UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Departamento de Ingeniería de Sistemas y Automática

Tesis Doctoral / PhD Dissertation

Técnicas de segmentación dinámica aplicadas a curvas de carga de consumo eléctrico diario de clientes domésticos

Dynamic segmentation techniques applied to load profiles of electric energy consumption from domestic users

Autor:
Ignacio Javier Benítez Sánchez

Directores:
José Luis Díez Ruano
Alfredo Quijano López

December 15, 2015

Esta tesis está realizada con las horas que no dediqué a mi familia.
A mi familia, por tanto, está dedicada.
A mi mujer, M. Carmen, y a mis hijos, Claudia y Juan.

Agradecimientos/Acknowledgement

There are people.
There are stories.
The people think they shape the stories, but the reverse is often closer to the truth.
Stories shape the world. They exist independently of people, and in places quite devoid of man, there may yet be mythologies.

Alan Moore, Steve Bissette, John Totleben and Tatjana Wood
Down Amongst The Dead Men. Swamp Thing Annual, Vol.2. DC Comics. January 1985.

- When I dream, sometimes I remember how to fly. You just lift one leg, then you lift the other leg, and you're not standing on anything, and you can fly. And then when I wake up I can't remember how to do it any more.
- So?
- So what I want to know is, when I'm asleep, do I really remember how to fly? And forget how when I wake up? Or am I just dreaming I can fly?
- When you dream, sometimes you remember. When you wake, you always forget.
- But that's not fair...
- No.

Neil Gaiman, Jill Thompson, Vince Locke and Daniel Vozzo
The Sandman: Brief Lives. The Sandman, No. 43. DC Comics. September 1992.

Resulta complicado hacer memoria de todas las personas a las que he consultado y que desinteresadamente me han ayudado y brindado su apoyo durante estos años en los que se ha ido gestando el presente documento. Aún así, a riesgo de poder dejarme a alguien fuera, quiero aventurarme a expresar mi agradecimiento a una serie de personas que han marcado el devenir de este trabajo.

Comienzo expresando mi gratitud a mis directores de tesis, D. José Luis Díez Ruano, del Departamento de Ingeniería de Sistemas y Automática, y D. Alfredo Quijano López, del Departamento de Ingeniería Eléctrica, por su apoyo y su guía durante todo el proceso de investigación y redacción de tesis, aportando cada uno desde su área de conocimiento su propia visión y perspectiva para encauzar la investigación realizada.

Mi agradecimiento también a la empresa Iberdrola Distribución Eléctrica S.A.U., y en especial a Isabel Navalón Burgos, por facilitar los datos de consumos energéticos que han sido empleados en esta tesis como base para probar y validar los desarrollos realizados.

También quiero agradecer a D. Pedro Albertos Pérez, profesor emérito en el Departamento de Ingeniería de Sistemas y Automática, sus consejos y su ayuda todo este tiempo, desde el comienzo de los trabajos que posteriormente dieron lugar al proceso de investigación y desarrollo que se describe en este documento.

Mi agradecimiento también a mis compañeros en el Departamento de Ingeniería de Sistemas y Automática, en la antigua Salabecarios, por su ayuda desinteresada durante el tiempo que permanecí allí. Gracias a José Camacho por pasarme su tesis completa en formato LaTeX, la cual he aprovechado como plantilla para la redacción de este documento.

Gracias a Leopoldo Armesto, del Departamento de Ingeniería de Sistemas y Automática, por su apoyo y sus consejos.

Gracias a Luciano Azpiazu, por ponerme en la pista de la principal referencia en los primeros trabajos sobre definición y análisis de curvas de carga.

Quiero expresar mi especial agradecimiento hacia mis compañeros en el Departamento de Smart Grids del Instituto Tecnológico de la Energía (ITE), por toda la ayuda que me han prestado. Igualmente, quiero dar mi agradecimiento a la Dirección del ITE, por su comprensión y apoyo.

Mi agradecimento también para dos personas sin las cuales realizar esta tesis no habría sido posible. La primera, Laura Moreno, quien me apoyó y realizó las gestiones oportunas para disponer de los datos de consumos y la oportunidad para analizarlos. La segunda, Ignacio Delgado, quien recogió el testigo y me ha ayudado con sus revisiones y recomendaciones sobre el presente trabajo.

Mi agradecimiento a mis padres y hermano por su apoyo durante todos estos años.

Mi agradecimiento especial a mis suegros, por todo el apoyo que me han brindado y su inestimable ayuda para conseguir la mayor cantidad de tiempo posible para realizar este trabajo.

Finalmente, mi agradecimiento a mi mujer, por su ayuda, su comprensión y su supervisión, y a mis hijos, que se han aliado con su madre para poder dejarme todas las horas libres que han podido (“papá, a hacer tesis”).

Puntuales apoyos algunos, intensos y prolongados en el tiempo otros, todos han sido decisivos para permitirme realizar esta investigación, que llevo por fin a su punto y seguido con el presente documento.

En Valencia, a 25 de Agosto de 2015

Resumen

El sector eléctrico se halla actualmente sometido a un proceso de liberalización y separación de roles, que está siendo aplicado bajo los auspicios regulatorios de cada Estado Miembro de la Unión Europea y, por tanto, con distintas velocidades, perspectivas y objetivos que deben confluir en un horizonte común, en donde Europa se beneficiará de un mercado energético interconectado, en el cual productores y consumidores podrán participar en libre competencia.

Este proceso de liberalización y separación de roles conlleva dos consecuencias o, visto de otra manera, conlleva una consecuencia principal de la cual se deriva, como necesidad, otra consecuencia inmediata. La consecuencia principal es el aumento de la complejidad en la gestión y supervisión de un sistema, el eléctrico, cada vez más interconectado y participativo, con conexión de fuentes distribuidas de energía, muchas de ellas de origen renovable, a distintos niveles de tensión y con distinta capacidad de generación, en cualquier punto de la red. De esta situación se deriva la otra consecuencia, que es la necesidad de comunicar información entre los distintos agentes, de forma fiable, segura y rápida, y que esta información sea analizada de la forma más eficaz posible, para que forme parte de los procesos de toma de decisiones que mejoran la observabilidad y controlabilidad de un sistema cada vez más complejo y con más agentes involucrados.

Con el avance de las Tecnologías de Información y Comunicaciones (TIC), y las inversiones tanto en mejora de la infraestructura existente de medida y comunicaciones, como en llevar la obtención de medidas y la capacidad de actuación a un mayor número de puntos en redes de media y baja tensión, la disponibilidad de datos sobre el estado de la red es cada vez mayor y más completa. Todos estos sistemas forman parte de las llamadas *Smart Grids*, o redes inteligentes del futuro, un futuro ya no tan lejano.

Una de estas fuentes de información proviene de los consumos energéticos de los clientes, medidos de forma periódica (cada hora, media hora o cuarto de hora) y enviados hacia las Distribuidoras desde los contadores inteligentes o *Smart Meters*, mediante infraestructura avanzada de medida o *Advanced Metering Infrastructure* (AMI). De esta forma, cada vez se tiene una ma-

yor cantidad de información sobre los consumos energéticos de los clientes, almacenada en sistemas de *Big Data*. Esta cada vez mayor fuente de información demanda técnicas especializadas que sepan aprovecharla, extrayendo un conocimiento útil y resumido de la misma.

La presente Tesis doctoral versa sobre el uso de esta información de consumos energéticos de los contadores inteligentes, en concreto sobre la aplicación de técnicas de minería de datos (*data mining*) para obtener patrones temporales que caractericen a los usuarios de energía eléctrica, agrupándolos según estos mismos patrones en un número reducido de grupos o *clusters*, que permiten evaluar la forma en que los usuarios consumen la energía, tanto a lo largo del día como durante una secuencia de días, permitiendo evaluar tendencias y predecir escenarios futuros. Para ello se estudian las técnicas actuales y, comprobando que los trabajos actuales no cubren este objetivo, se desarrollan técnicas de *clustering* o segmentación dinámica aplicadas a curvas de carga de consumo eléctrico diario de clientes domésticos. Estas técnicas se prueban y validan sobre una base de datos de consumos energéticos horarios de una muestra de clientes residenciales en España durante los años 2008 y 2009. Los resultados permiten observar tanto la caracterización en consumos de los distintos tipos de consumidores energéticos residenciales, como su evolución en el tiempo, y permiten evaluar, por ejemplo, cómo influenciaron en los patrones temporales de consumos los cambios regulatorios que se produjeron en España en el sector eléctrico durante esos años.

Resum

El sector elèctric es troba actualment sotmès a un procés de liberalització i separació de rols, que s'està aplicant davall els auspicis reguladors de cada estat membre de la Unió Europea i, per tant, amb distintes velocitats, perspectives i objectius que han de confluir en un horitzó comú, on Europa es beneficiarà d'un mercat energètic interconnectat, en el qual productors i consumidors podran participar en lliure competència.

Aquest procés de liberalització i separació de rols comporta dues conseqüències o, vist d'una altra manera, comporta una conseqüència principal de la qual es deriva, com a necessitat, una altra conseqüència immediata. La conseqüència principal és l'augment de la complexitat en la gestió i supervisió d'un sistema, l'elèctric, cada vegada més interconnectat i participatiu, amb connexió de fonts distribuïdes d'energia, moltes d'aquestes d'origen renovable, a distints nivells de tensió i amb distinta capacitat de generació, en qualsevol punt de la xarxa. D'aquesta situació es deriva l'altra conseqüència, que és la necessitat de comunicar informació entre els distints agents, de forma fiable, segura i ràpida, i que aquesta informació siga analitzada de la manera més eficaç possible, perquè forme part dels processos de presa de decisions que milloren l'observabilitat i controlabilitat d'un sistema cada vegada més complex i amb més agents involucrats.

Amb l'avanç de les tecnologies de la informació i les comunicacions (TIC), i les inversions, tant en la millora de la infraestructura existent de mesura i comunicacions, com en el trasllat de l'obtenció de mesures i capacitat d'actuació a un nombre més gran de punts en xarxes de mitjana i baixa tensió, la disponibilitat de dades sobre l'estat de la xarxa és cada vegada major i més completa. Tots aquests sistemes formen part de les denominades *Smart Grids* o xarxes intel·ligents del futur, un futur ja no tan llunyà.

Una d'aquestes fonts d'informació prové dels consums energètics dels clients, mesurats de forma periòdica (cada hora, mitja hora o quart d'hora) i enviats cap a les distribuïdores des dels comptadors intel·ligents o *Smart Meters*, per mitjà d'infraestructura avançada de mesura o *Advanced Metering Infrastructure* (AMI). D'aquesta manera, cada vegada es té una major quantitat d'informació sobre els consums energètics dels clients, emmagatzemada en sistemes de *Big Data*. Aquesta cada vegada major font d'informació demanda tècniques especialitzades que sàpiguen aprofitar-la, extraient-ne un coneixement útil i resumit.

La present tesi doctoral versa sobre l'ús d'aquesta informació de consums energètics dels comptadors intel·ligents, en concret sobre l'aplicació de tècniques de mineria de dades (*data mining*) per a obtenir patrons temporals que caracteritzen els usuaris d'energia elèctrica, agrupant-los segons aquests mateixos patrons en una quantitat reduïda de grups o *clusters*, que permeten avaluar la forma en què els usuaris consumeixen l'energia, tant al llarg del dia com durant una seqüència de dies, i que permetent avaluar tendències i predir escenaris futurs. Amb aquesta finalitat, s'estudien les tècniques actuals i, en comprovar que els treballs actuals no cobreixen aquest objectiu, es desenvolupen tècniques de *clustering* o segmentació dinàmica aplicades a corbes de càrrega de consum elèctric diari de clients domèstics. Aquestes tècniques es proven i validen sobre una base de dades de consums energètics horaris d'una mostra de clients residencials a Espanya durant els anys 2008 i 2009. Els resultats permeten observar tant la caracterització en consums dels distints tipus de consumidors energètics residencials, com la seua evolució en el temps, i permeten avaluar, per exemple, com van influenciar en els patrons temporals de consums els canvis reguladors que es van produir a Espanya en el sector elèctric durant aquests anys.

Abstract

The electricity sector is currently undergoing a process of liberalization and separation of roles, which is being implemented under the regulatory auspices of each Member State of the European Union and, therefore, with different speeds, perspectives and objectives that must converge on a common horizon, where Europe will benefit from an interconnected energy market in which producers and consumers can participate in free competition.

This process of liberalization and separation of roles involves two consequences or, viewed another way, entails a major consequence from which other immediate consequence, as a necessity, is derived. The main consequence is the increased complexity in the management and supervision of a system, the electrical, increasingly interconnected and participatory, with connection of distributed energy sources, much of them from renewable sources, at different voltage levels and with different generation capacity at any point in the network. From this situation the other consequence is derived, which is the need to communicate information between agents, reliably, safely and quickly, and that this information is analyzed in the most effective way possible, to form part of the processes of decision taking that improve the observability and controllability of a system which is increasing in complexity and number of agents involved.

With the evolution of Information and Communication Technologies (ICT), and the investments both in improving existing measurement and communications infrastructure, and taking the measurement and actuation capacity to a greater number of points in medium and low voltage networks, the availability of data that informs of the state of the network is increasingly higher and more complete. All these systems are part of the so-called *Smart Grids*, or intelligent networks of the future, a future which is not so far.

One such source of information comes from the energy consumption of customers, measured on a regular basis (every hour, half hour or quarter-hour) and sent to the Distribution System Operators from the *Smart Meters* making use of *Advanced Metering Infrastructure* (AMI). This way, there is an increasingly amount of information on the energy consumption of customers, being stored in *Big Data* systems. This growing source of information demands specialized techniques which can take benefit from it, extracting a useful and summarized knowledge from it.

This thesis deals with the use of this information of energy consumption from Smart Meters, in particular on the application of *data mining* techniques to obtain temporal patterns that characterize the users of electrical energy, grouping them according to these patterns in a small number of groups or *clusters*, that allow evaluating how users consume energy, both during the day and during a sequence of days, allowing to assess trends and predict future scenarios. For this, the current techniques are studied and, proving that the current works do not cover this objective, *clustering* or dynamic segmentation techniques applied to load profiles of electric energy consumption from domestic users are developed. These techniques are tested and validated on a database of hourly energy consumption values for a sample of residential customers in Spain during years 2008 and 2009. The results allow to observe both the characterization in consumption patterns of the different types of residential energy consumers, and their evolution over time, and to assess, for example, how the regulatory changes that occurred in Spain in the electricity sector during those years influenced in the temporal patterns of energy consumption.

Contents

Part I The Smart Grid. Current scenario and future trends

Part II State of the art on data mining and clustering techniques

Part III Development of dynamic clustering techniques applied to load profiles time series

Part IV Conclusions

Part V Appendices

List of Figures

List of Tables

Motivation and objectives

0.1 Motivation

The development of the present thesis is mainly motivated by the works developed in two different research projects: the Profesion@l Project and the GAD project:

- The Profesion@l project, arranged under the auspice of the European Union (European Union EQUAL Initiative, code ES-ES20040550, 2004), aimed to study the issues behind the creation of gender stereotypes about the different careers and professions, starting from the days at school, until the students access to university. Through the use of a specifically designed software, information gathered from groups of students at different educational stages was analyzed, applying clustering algorithms and fuzzy logic inference, with the objective of developing a model able to characterize, predict and evaluate the different motivations for the selection of a professional career, and the influence of gender stereotypes in this decision.
- The GAD or "Active Demand Management" (in Spanish) project was a project supported by the Spanish Government, and participated by 14 different companies and 14 research centres. It was sponsored by the CDTI (Technological Development Centre of the Ministry of Science and Innovation of Spain), and financed by the INGENIO 2010 program. The objective of the GAD project was to investigate and develop solutions to optimize the electrical consumption in low and medium voltage users, by the research and development of new tools for the Demand Side Management. The data analyzed in this thesis has been provided by the Spanish Distribution System Operator Iberdrola Distribución Eléctrica S.A.U. and the GAD project, in form of a database with load profile registers from smart meters, gathered from the sample of Spanish users selected.

The Profesion@l project provided, through its developments and works, a first insight to the data mining techniques and the clustering algorithms and fuzzy inference. On the other hand, the GAD project was the initial scenario of data analysis and characterization of profiles of energy consumption from electrical energy customers. From the developments and results on load profiling that were obtained in the GAD project, and the knowledge gained from the Profesion@l project, the following step was to apply this knowledge to pursue the objective of characterizing customers and evaluating their evolution in patterns through time. These are the motivations and the initial steps that have evolved in the developments presented in this thesis.

0.2 Objectives

As will be discussed in this document, there is no specific work currently in the literature that addresses the dynamic clustering of load profiles as a daily time series data. As will be also described, this will be a need that companies in the sector of electrical energy will have to address, if they want to extract useful knowledge or information from huge volumes of data that are being increasingly measured and stored. This is, therefore, the objective of the present thesis: to study, develop, test and evaluate methods and techniques to perform clustering, obtain centroids or prototypes and visualize the results, on time series of daily load profiles.

0.3 Main Contributions

The main contributions that have been produced during the development of this thesis are the following:

- The development of a **common framework** to a dynamic cluster analysis of time series data of load profiles, applying different clustering techniques and distance measures, has been presented. This work has also been described in the following communication: “Ignacio Benítez, Alfredo Quijano, José-Luis Díez, Ignacio Delgado. *Clustering of Time Series Load Profiles for Grid Reliability.* International Conference on Condition Monitoring, Diagnosis and Maintenance 2015 (CMDM 2015). Bucharest, Romania, October 5th -8th, 2015”.
- A graphical method for the visualization and **representation of the clusters**, and a methodology for the **interpretation of the results**, by means of specific indices, have been described. This methodology is presented in the following article: “Benítez, I.; Quijano, A.; Díez, J.-L. and Delgado, I. *Dynamic clustering segmentation applied to load profiles of energy consumption from Spanish customers.* International Journal of Electrical Power & Energy Systems , 2014, 55, 437 - 448”, which also presents the first approach for dynamic clustering developed, along with the results from the first test described in this thesis.

- A selection and modification of **cluster validity indices** appropriate to evaluate the clustering results in this proposed framework has been performed.
- An approach to compare two load profiles time series as the **comparison between two 3D surfaces** has been presented.
- A proposal to obtain the distance between the two surfaces as the decomposition in a number of smaller linear surfaces and the application of a new **Hausdorff-based similarity measure** has been explained.

The three last contributions are described in a new article with the following title: “Dynamic Clustering of Residential Electricity Consumption Time Series Data Based on Hausdorff Distance”, by Ignacio Benítez, José-Luis Díez, Alfredo Quijano and Ignacio Delgado. This paper has been sent to the Electric Power Systems Research Journal for publication and is currently under review.

In order to reach the contributions listed above, two introductions to different fields of knowledge and two different states of the art have been produced. These are the following:

- An introduction to the changing environment of the current power systems and the Smart Grids is presented. The current scenario regarding the energy market and the regulatory situation in Spain and the European Commission is so dynamic that, probably while these lines are being written, part of the information described in this thesis is becoming obsolete. However, this introduction has been considered necessary in order to understand the needs of data analysis and the objectives approached in this thesis.
- An introduction to the field of Knowledge Discovery in Databases, and Data Mining as being part of this process and, more recently, the shift towards Big Data systems and Big Data Analytics, has also been described. Clustering and dynamic clustering techniques are presented, as a specific objective of the Data Mining.
- A state of the art on current algorithms and techniques on dynamic clustering has been performed, and is presented in this thesis.
- Complementing this review, a state of the art on clustering, classification and forecasting algorithms focused on the analysis of energy consumption data is also presented. Conclusions are obtained and discussed from both states of the art. As a result, the developments previously described have been obtained.

All the developments presented in this thesis have been programmed making use of the Matlab$^{\text{TM}}$ software, and its Database Toolbox, in order to access the data, stored in a MySQL database.

The sole exception regards the developments made for the paper on “Clients segmentation according to their domestic energy consumption by the use of Self-Organizing Maps”, cited in this thesis, where the SOM 2.0 toolbox provided by the Aalto University (http://www.cis.hut.fi/somtoolbox/) was used.

0.4 Publications

A number of publications have been produced through the development of this thesis. Some of them are a direct presentation of the works developed. Other publications are a result of the application of some of the developments done to a different field of knowledge or on data of a different nature than the ones presented here. Following, all these publications are introduced:

1. Ignacio Benítez, José Luis Díez, Pedro Albertos. *Applying Dynamic Mining on Multi-Agent Systems.* In Proceedings of the 17th World Congress of the International Federation of Automatic Control (IFAC). Seoul, Korea, July 6-11, 2008. This paper represents the foundational work on dynamic clustering from the authors, that is the basis for all the developments presented in the thesis. It approaches the dynamic analysis and segmentation of a group of agents with dynamical characteristics, i.e., a varying location.
2. I. Benítez Sánchez, I. Delgado Espinós, L. Moreno Sarrión, A. Quijano López, and I. Navalón Burgos. *Clients segmentation according to their domestic energy consumption by the use of Self-Organizing Maps.* In Proceedings of the 6th International Conference on the European Energy Market, EEM09, 2009. This is a work developed under the GAD project, which approached the static classification of energy consumption load profiles from residential users, in this case making use of Self-Organizing Maps (SOM). This initial work of classification was the basis to the extension to the dynamic analysis of the data proposed and developed in this thesis.
3. Benítez, I.; Quijano, A.; Díez, J.-L. and Delgado, I. *Dynamic clustering segmentation applied to load profiles of energy consumption from Spanish customers.* International Journal of Electrical Power & Energy Systems , 2014, 55, 437 - 448. This paper presents the first approach and conclusions developed in this thesis, for the dynamic clustering of time series load profiles.
4. Benítez, I.; Blasco, C.; Mocholí, A. and Quijano, A. *A Two-Step Process for Clustering Electric Vehicle Trajectories.* Proceedings of the IEEE International Electric Vehicle Conference (IEVC 2014), 2014. This paper makes use of the development of a two-step dynamic clustering proposed in this thesis, but to a different field or area of knowledge (Electro-mobility), and with different techniques applied, due to the different nature of the data. In the development presented in the thesis, a two-step dynamic clustering algorithm is proposed, which makes use of a Hausdorff-based distance to compute similarity between data objects. In this paper, the Hausdorff-based distance is replaced by a dynamic time warping distance,

and the objective is to cluster Electric Vehicle trajectories, given in sequences of spatial coordinates (longitude and latitude).

5. Ignacio Benítez, Alfredo Quijano, José-Luis Díez, Ignacio Delgado. *Clustering of Time Series Load Profiles for Grid Reliability.* International Conference on Condition Monitoring, Diagnosis and Maintenance 2015 (CMDM 2015). Bucharest, Romania, October 5th -8th, 2015. This paper presents the developments made in this thesis of a common framework for the Type 3 dynamic clustering of time series load profiles. In this case the analysis is oriented to the objective of serving as a predictive maintenance tool.
6. I. Benítez, A. Quijano, I. Delgado, J.L. Díez. Classification of customers based on temporal load profile patterns. CIGRÉ Session, Paris. 21-26 August, 2016. This paper, accepted for publication but not presented yet, describes the results of the development of a classifier of consumers based on their temporal patterns, obtained with the framework developed in this thesis, and the definition of some indicators regarding trends and shapes of temporal energy consumption in the resulting patterns.

0.5 Document structure

The present document focuses on the approach of data mining techniques, specifically dynamic clustering of time series data, to the analysis of energy consumption data. It is structured in five differentiated Parts, which are the following:

1. The first Part of this document presents the current scenario of today's power grids and their new management models, that have arisen as a result of an unbundling process, in order to convert the classical scenarios of oligopolies or State-owned electrical networks in frameworks to compete on equal terms, that guarantee the participation of private companies and third parties in the generation, transport, distribution and consumption of electrical energy. A main need will be identified in this Part, as a "side effect" that comes with the integration of the Smart Grids: **the adequate management of the information** from smart metering data, or how to properly manage and benefit from large amounts of data that are being available.
2. Part II describes which can be considered as the main path towards a solution: the structured and ordered analysis of the data, or, in other words, how to obtain the desired information from large sets of time series data. For this reason, the broad research field of Knowledge Discovery in Databases (KDD) is presented and, as a main component of KDD, Data Mining objectives and techniques are described. It is the cluster analysis and, more specifically, the cluster evolution analysis, or the **dynamic clustering of time series data**, the data mining objective which is

considered of relevance for the analysis of the Smart Metering data from energy consumption, arranged in daily vectors of 24 hours or dimensions. For this reason, a state of the art in clustering techniques are described in this Part. The main conclusion of all the reviews performed in this Part II is that, as indicated previously in Section 0.2 of Objectives, **no specific development has been found that addresses the dynamic clustering and visualization of energy consumption load profiles time series data.**

3. The development of these specific techniques, therefore, are presented in the following Part of this document, Part III. The development made, as will be explained, is a flexible framework that can make use of different clustering techniques for **raw data dynamic clustering** presented in the state of the art, extended **to process time series data of n dimensions**, where all the dimensions have the same magnitude (energy – kWh) and, as an innovation perspective, they are processed as a daily time series (for a week, or a month, or a year...). This development is presented and tested on a dataset of energy consumption load profiles from a sample of domestic users in Spain.
4. In Part IV, the results obtained in the previous Part are evaluated and the main conclusions are presented, along with future works that can be followed from the developments made and presented in this thesis.
5. Part V, finally, gathers the different Annexes produced and the references cited throughout this thesis.

Part I

The Smart Grid. Current scenario and future trends

The first Part of this document presents the current scenario of today's power grids and their new management models, that have arisen as a result of an unbundling process, in order to convert the classical scenarios of oligopolies or State-owned electrical networks in frameworks to compete on equal terms, that guarantee the participation of private companies and third parties in the generation, transport, distribution and consumption of electrical energy.

The creation and liberalization of the energy market and the free trading between agents is one of the key pillars of this structure. The supervision and maintenance of power quality and voltage levels among all the nodes in the network, dealing with a dynamic real-time balancing between generation and consumption, could be identified as the second key pillar of this structure. Finally, the need for a close interconnection between the technical management of the network and the trades of energy would be the third key pillar of this structure.

Information and Communication Technologies (ICT) are able to capture the state of the grid and much more related information in real time. Big Data and data warehousing systems are able to conveniently store and have the data available for future analysis and specific queries. The challenge arises, however, in how to adequately manage the huge amounts of data that are beginning to be available, in order to extract useful knowledge from them, that can be of profit for the different agents involved in the management of power systems and energy trading, providing a valuable return of the investment being done in the ICT and Advanced Metering Infrastructure (AMI). These three fields represent the future of the power grids or, in another words, the *Smart Grids* of the short-term and medium-term future.

An introduction Chapter (Chapter 1) first presents the current scenario in the shift towards the liberalization of the energy market, in the European Union and in Spain, regarding legislative actions. Chapter 2 then provides a view of the current electrical power grids, and the main generation or electricity production sources and their integration in the grid, specifically for renewable and distributed energy resources. Chapter 3 describes the new features that the Smart Grids bring to the management of the power systems, such as standardization and interoperability aspects, and the availability of information through ICT and AMI. New concepts that derive from the implementation of these technologies, such as Demand Response and Demand Side Management, are described. Finally, Chapter 4 identifies the benefits of the Smart Grids and also one of its main "drawbacks": the growing availability of information, in daily and hourly streams of data, which makes necessary the availability of systems able to properly manage the information first, and, following, to analyze it. A specific objective is identified, which can provide valuable knowledge for decision taking to the different agents: to evaluate trends in temporal patterns of energy consumption from end users in the electricity distribution network, grouping the users by their temporal electric signature, or the way the energy is used and its amount variation in time.

1

Introduction

The different sectors of generation, transportation and distribution (and commercialization) of electric energy are currently undergoing a slow process of transition towards a new paradigm of liberalized, competence-free market, where the power grid is managed by all the interested parties and agents in two ways: physical and virtual. The **physical** management is performed by all the agents that ensure that the electricity is generated, transported and distributed to the end users under the required conditions of power quality. This power grid of the (not so far) future is distributed in control areas at different aggregation levels (Wide Area Networks, Local Area Networks), managed locally by the different agents, from Transport Network Operators (TNO), to Distribution Network Operators (DNO) to the smaller owners of the low voltage electrical grids where the end users are connected (microgrid managers, neighborhoods, districts). The local management of the power grid in control areas of different size, by different agents or companies, makes necessary the standardization and deployment of robust, fast and interoperable communication protocols, to allow at the same time a local management of power networks, following local needs or directives, and a global, centralized supervision by a supervision agent, responsible of guaranteeing grid stability and balance between generated and consumed energy under real time constraints [1].

As the need for communication and management in real time rises, the need to monitor and measure the state of the grid with the smallest period of time also rises. The requirements for monitoring and metering of energy at all the voltage levels of the grid (High Voltage, Medium Voltage, Low Voltage) have progressed in parallel with the technology readiness level to comply with these. Besides, the local generation (and consumption) of electricity has gained interest in the recent years, particularly with the integration of generation from renewable energy resources (mainly photovoltaic, solar thermal and wind) of different sizes and installed power (from low voltage and a few kilowatts to medium voltage and Megawatts of power injected to the grid). The energy storage vector has also gained interest, as the technology provides advanced resources of smaller dimension and higher density devices to store and deliver energy from and to the grid, making use of electro-chemical principles or other forms of energy conversion [2]. The integration of these local resources imply the need for the development and validation of more complex, fast and robust management and control systems, able to integrate controllable and non-controllable energy sources, with different technologies, constraints and power connections to the grid, most of them through specific electronics equipment.

The visibility and controllability of power networks is gradually increasing at all levels. As the needs to monitor, supervise and control these energy flows increase, so does the complexity of the Information and Communication Technologies (ICT) infrastructure needed for this purpose, and the software architectures, systems and algorithms implemented of Information Systems and Data Analysis or Data Mining.

Regarding the **virtual** management of electricity, this is performed by all the agents that interact in a (regulated but free) market, where the electric energy is traded as an asset or good (a commodity), under the supervision of an agent or agents that ensure: 1) that the energy transactions are performed following specific regulations and 2) that the energy sold and purchased at all times fills the predicted energy demand for a defined window frame and falls under the technical requirements of energy balance and power quality (in voltage and frequency) either at global and local scale in the power grid. If this is not the case, specific mechanisms are activated to suppress this foreseen situation, such as the provision of extra demand or generation of electricity in intra-day or operation markets. These are *ancillary services* [3], needed to stabilize the grid.

The energy market behaves, as other markets do, as an offer-demand model, with a clear exception: the electricity demand is not elastic, i.e., it cannot be substituted by any other commodity (at least, not immediately) [4]. The energy is traded in two main scenarios or time scales: in the following period, or at medium or long term. The energy traded for the next period, typically for the 24 hours of the following day, is sold and purchased in the *spot* (i.e., "on the spot") market. The energy traded in medium or long terms is sold and purchased by means of forward contracts between seller and buyer, or at forward or Over-The-Counter (OTC) markets. These contracts are allegedly of a lower risk. In the spot markets, the energy *pool* model has been implemented: for each hour of the day, all the producers submit their bids of estimated generation capacity and production price per unit of energy. All the retailers submit the demand expected and the price they would pay for it. Both quantities are ranked: the generation in order of increasing price and the consumption in order of decreasing price. The point of intersection of the supply and demand curves sets the market clearing price and fixes the system marginal price for that hour. Figure 1.1 depicts an example of the energy pool market clearing price for the Iberian (Spanish and Portuguese) Energy Market Operator, OMIE (*Operador del Mercado Ibérico de Energía*).

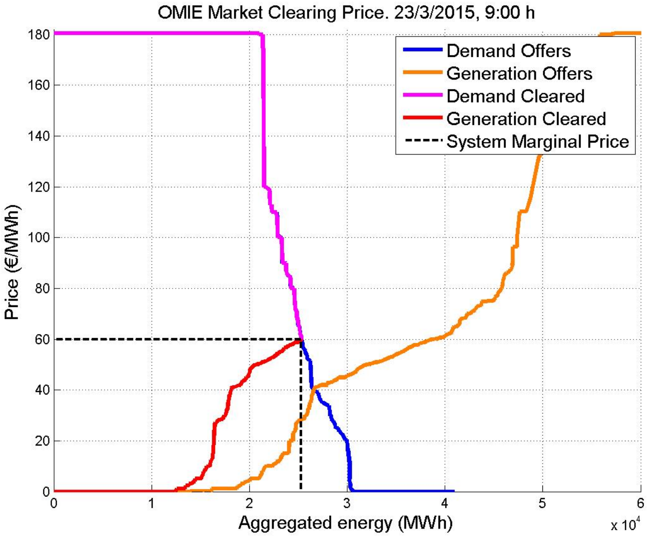


Fig. 1.1. Example of market clearing price in the Spanish Energy Market Pool. Source: OMIE (Operador del Mercado Ibérico de Energía).

The market clearing price, for the hour 9 of the day 23rd of March, 2015, displays four curves: the offered generation (in orange) and purchase bids (in blue), and the final cleared offer and demand curves (in red and magenta colors). The discrepancy between the offered and cleared generation is due to the aggregation of generation offer imposed by the System Operator to meet technical requirements.

1.1 Liberalization of the energy market in the European Union

This process of transition towards liberalized energy markets is progressing globally, although with different speeds, in the different countries worldwide. Regarding the deployment of Advanced Metering Infrastructure (AMI) in Europe, the current 28 Member States of the European Union are requested to deploy intelligent metering systems for at least the 80 % of consumers before 2020 [5], subject to a positive economical assessment of the long-term costs and benefits of the roll-out, performed by each country. The last report from the European Commission (EC) addressing the progress of this deployment states, among other conclusions, the following [6]:

- "16 Member States (Austria, Denmark, Estonia, Finland, France, Greece, Ireland, Italy, Luxemburg, Malta, Netherlands, Poland, Romania, Spain, Sweden and the UK) will proceed with large-scale roll-out of smart meters by 2020 or earlier, or have already done so. In two of them, namely in Poland and Romania, the Cost-Benefit Analyses (CBAs) yielded positive results but official decisions on roll-out are still pending;

- In seven Member States (Belgium, the Czech Republic, Germany, Latvia, Lithuania, Portugal, and Slovakia), the CBAs for large-scale roll-out by 2020 were negative or inconclusive, but in Germany, Latvia and Slovakia smart metering was found to be economically justified for particular groups of customers;
- For four Member States (Bulgaria, Cyprus, Hungary and Slovenia), the CBAs or roll-out plans were not available at the time of writing; and
- Legislation for electricity smart meters is in place in the majority of Member States, providing for a legal framework for deployment and/or regulating specific matters such as timeline of the roll-out, or setting technical specifications for the meters, etc. Only five Member States (Belgium, Bulgaria, Hungary, Latvia and Lithuania), have no such legislation in place."

The document emphasizes the need for a cybersecurity and data privacy framework [7] in the smart metering infrastructure being deployed, following EU legislation on this issue. It also recommends the use of available standards, to ensure technical and commercial interoperability.

Regarding the implementation of liberalized energy markets in Europe, the same EC Directive 2009/72 that encouraged the deployment of AMI, defined also the main rules for the energy market liberalization and the unbundling of the system in the European Union [5], therefore allowing both producers and consumers to participate in the energy market transactions. The EC encourages the Member States to ensure that an adequate "playing field" is set up for this purpose. For instance, its 35^{th} paragraph states:

> "In order to ensure effective market access for all market players, including new entrants, non-discriminatory and cost-reflective balancing mechanisms are necessary. As soon as the electricity market is sufficiently liquid, this should be achieved through the setting up of transparent market-based mechanisms for the supply and purchase of electricity, needed in the framework of balancing requirements. In the absence of such a liquid market, national regulatory authorities should play an active role to ensure that balancing tariffs are non-discriminatory and cost-reflective."

Each EU Member State has, therefore, a national regulatory authority that supervises and "watches" the adequate functioning of the markets. In Spain, for instance, the *Comisión Nacional de los Mercados y la Competencia* or National Commission for Markets and Competition (CNMC) performs this task.

Regarding the energy markets in the EU, there are 7 regional electricity wholesale markets operating in year 2014 [8]:

- Central Western Europe (CWE). Formed in 2010 as a result of the coupling of energy power exchange markets from Austria (EXAA), Belgium (Belpex), France (EPEX SPOT SE), Germany (EPEX SPOT SE), the Netherlands (APX) and Switzerland (EPEX SPOT SE).
- British Isles. Formed by two power exchange markets, APX Power UK (United Kingdom) and SEMO (Single Energy Market Operator for the Republic of Ireland and Northern Ireland).
- Northern Europe (NordPool). Formed by a single market (Nord Pool Spot AS), that performs power exchange trading for Denmark, Estonia, Finland, Latvia, Lithuania, Norway and Sweden since 1996.
- Apennine Peninsula. Formed by the Gestore del Mercato Elettrico S.p.A., or Italian Power Exchange, from Italy.
- Iberian Peninsula. Formed by the Operador del Mercado Ibérico - Polo Español S.A. / - Polo Portugués SGPS (OMIE/OMIP), which operate the spot market (OMIE) and forward and OTC market (OMIP) for Spain and Portugal since 2007.
- Central Eastern Europe. Formed by the PXE (Power Exchange Central Europe), which integrates the trading markets for Czech Republic, Hungary, Poland, Romania, Slovakia and Slovenia since 2007.
- South Eastern Europe. Formed by Lagie, the Operator of the Electricity Market from Greece.

The objective of the EC progresses towards a future common power exchange market, involving not only trading but also a physical interconnection between electrical systems. New power exchange connections are planned to accomplish this objective [9]. Regarding the power exchange trading, the initiative called Price Coupling of Regions or PCR has been launched in 2009 [10], with the objective of developing a single price coupling framework for all the power exchange markets in the European Union, by the development and application of a common algorithm, which has been called EUPHEMIA (Pan-European Hybrid Electricity Market Integration Algorithm). EUPHEMIA works on the principle that in a market coupling scenario, markets with lower prices can export electricity to the markets with the highest prices. The algorithm is based on the market coupling algorithm called COSMOS, developed by the CWE power exchange operator [11]. This algorithm uses branch-and-bound optimization techniques [12] to provide the most feasible solution of price coupling, given a specific Available Transmission Capacity or ATC defined by the TSOs.

1.2 Liberalization of the energy market in Spain

The liberalization of the energy market in Spain was formalized through the national regulation RD 54/1997, in year 1997 [13]. This Law declared the unbundling of the electrical power network in Spain, alienating it as a whole from the public service, and distributing the necessary competences among the following roles:

1. The Producers of electricity, or generating companies.
2. The Market Operator.
3. The System Operator.
4. The Transmission System Operator.
5. The Distribution System Operators.
6. The Retailers.
7. The Consumers.
8. The System Load Managers. These are entities that, being consumers of electricity, are allowed to trade with the supplied energy, for purposes of load charging or storage.

These roles are further described in the following Chapter. The System Operator has to guarantee that the electricity is supplied under conditions of continuity and power quality. Another of its obligations is to provide the correct coordination between production plants and the transmission network. Thus, the System Operator in the Spanish case is also the Transmission System Operator. In Spain this role is performed by the company Red Eléctrica de España S.A.U. (REE). The transmission network is formed, according to the Law, by cables, systems and installations which transport voltage equal or higher than 380 kV (primary transmission network), and by cables, systems and installations which transport voltage equal or higher than 220 kV (secondary transmission network), or below this value but they perform a transmission purpose. Interconnections with electrical systems from other countries or insular connections are also managed by the System Operator.

The structure of the power exchange markets are also stated in the RD 54/1997, being the following:

1. Forward.
2. Day-ahead.
3. Intra-day.
4. Resolution of technical constraints.
5. Ancillary services.
6. Deviations management.
7. Non-organized.

The first two (forward and day-ahead markets) are managed by the Market Operator, and the other ones due to resolution of technical constraints and real time imbalances in the power grid (intra-day, technical constraints, ancillary services and deviations management markets) are managed by the System Operator. In Spain, the Market Operator role is performed by the company OMIE (Operador del Mercado Ibérico de Energía).

Subsequent regulations have modified some articles from this Law, until year 2013, when a new Law, 24/2013 [14], has derogated the one from

1997. This Law on the electrical sector maintains the main definitions of the structure of the liberalized energy market and introduces new concepts such as the regulation of self-consumption. Other articles are complemented with new paragraphs, such as the structure of the power exchange market, where now bilateral and forward contracts are described.

1.3 Conclusions

It can be said that the liberalization of the energy market in Spain and the EU is today a reality; however there are yet difficulties and issues that need to be solved. Technical and regulatory limitations still may pose barriers for a truly accessible, interconnected and open European energy market. Recently the EC has launched a public consultation process on a new energy market design [15]. In this communication, the EC acknowledges that Europe's electricity system is "in the middle of a period of profound change". The following critical paths that need to be addressed are identified:

1. New models for electricity markets. With actions such as fostering the power interconnections, and allowing a better integration of renewables in the energy market.
2. Improving the cooperation among Regions and TSOs. Including a common alignment of national policies and regulations for the integration of energy markets.
3. Ensuring security of supply.

At least two of these issues have a special significance due to their complexity: the seamless integration of renewable generation in the grid, especially the one coming from distributed resources, and the common agreement on policies and regulations in the member States for the integration of the energy markets. As will be seen in the following Chapter, the injection of energy from distributed energy resources at different voltage levels, together with "classical" generation technologies in large production plants, in a network which was designed as a one way energy flow, is still a non-resolved issue [16].

2

The current power grid

2.1 The different roles involved in the management of Power Systems

With the unbundling of the management of power systems, in favor of liberalized, free competence markets, traditional electricity companies have had to adapt their organizational structure to embrace specific roles or purposes, which from now on become in most cases exclusive from other roles. These are mainly four: production, transmission, distribution and commercialization of electricity. The rules of the – now liberalized – markets do not allow that the same company, for instance, performs distribution and commercialization activities. A typical list of agents or roles in power systems, extracted from Kirschen and Strbac [4] is given below:

- "**Generating companies** produce and sell electrical energy. They may also sell services such as regulation, voltage control and reserve that the system operator needs to maintain the quality and security of the electricity supply. A generating company can own a single plant or a portfolio of plants of different technologies.
- **Distribution companies** own and operate distribution networks. Initially called Distribution Network Operators (DNO), their definition has shifted for the sake to embrace a broader scope to **Distribution System Operators or DSO**.
- **Retailers** buy electrical energy on the wholesale market and resell it to consumers who do not wish, or are not allowed, to participate in this wholesale market. Retailers do not have to own any power generation, transmission or distribution assets. Some retailers are subsidiaries of generation or distribution companies. All the customers of a retailer do not have to be connected to the network of the same distribution company.
- The **Independent System Operator (ISO)** has the primary responsibility of maintaining the security of the power system. It is called independent because in a competitive environment, the system must be operated

in a manner that does not favor or penalize one market participant over another. An ISO would normally own only the computing and communications assets required to monitor and control the power system. An ISO usually combines its system operation responsibility with the role of the operator of the market of last resort.

- A **Market Operator (MO)** typically runs a computer system that matches the bids and offers that buyers and sellers of electrical energy have submitted. It also takes care of the settlement of the accepted bids and offers. This means that it forwards payments from buyers to sellers following delivery of the energy. The independent system operator (ISO) is usually responsible for running the market of last resort, that is, the market in which load and generation are balanced in real time. Markets that close some time ahead of real time are typically run by independent for-profit market operators.
- **Transmission companies** own transmission assets such as lines, cables, transformers and reactive compensation devices. They operate this equipment according to the instructions of the ISO. In some cases the same company has both responsibilities; in this case they are called **Transmission System Operators or TSO**.
- The **Regulator** is the governmental body responsible for ensuring the fair and efficient operation of the electricity sector. It determines or approves the rules of the electricity market and investigates suspected cases of abuse of market power. The regulator also sets the prices for the products and services that are provided by monopolies.
- **Small consumers** buy electrical energy from a retailer and lease a connection to the power system from their local distribution company. Their participation in the electricity market usually amounts to no more than choosing one retailer among others when they have this option.
- **Large consumers**, on the other hand, will often take an active role in electricity markets by buying their electrical energy directly through the market. Some of them may offer their ability to control their load as a resource that the ISO can use to control the system. The largest consumers are sometimes connected directly to the transmission system."

However, this description may have become a simplistic view. The Harmonised Electricity Market Role Model [17] is a document developed and maintained by ebIX (European forum for energy Business Information eXchange), EFET (European Federation of Energy Traders) and ENTSO-E (European Network of Transmission System Operators for Electricity). It describes a model identifying all the roles that can be played for given domains within the electricity market, where a domain represents "a delimited area that is uniquely identified for a specific purpose and where energy consumption, production or trade may be determined". The roles are of a logical nature, and represent the "external intended behavior of a party". A role must represent a relatively autonomous function.

In its current version (year 2014), the complete role model defines 37 roles and 23 domains. However, the same party can play several of these roles. The full list of these roles and domains is described below. Regarding domains, the following are defined [17]:

1. "**Accounting Point**. An entity under balance responsibility where balance supplier change can take place and for which commercial business processes are defined. These entities are usually defined in a contract. Typical business processes where this would be used may be "compensation managemen", "calculation of energy volumes", etc.
2. **Allocated Capacity Area**. A market area where the transmission capacity between the Balance Areas is given to the Balance Responsible Parties according to rules carried out by a Transmission Capacity Allocator. Trade between balance areas is carried out on a bilateral or unilateral basis. Examples are France and Spain (the Pyrenees) and Portugal and Spain.
3. **Balance Group**. A collection of Metering Points for imbalance settlement within a Market Balance Area.
4. **Capacity Market Area**. A market area where the transmission capacity between the Market Balance Areas is given to the Balance Responsible Parties in a price based process separated from trading carried out by a Transmission Capacity Allocator. Trade between Market Balance Areas is carried out on a bilateral or unilateral basis.
5. **Certificate Area**. A Certificate Market Area where a common set of rules relative to taxes and pricing for defined types of energy production are applied.
6. **Common Capacity Area**. A Market Area where the available transmission capacity between the Market Balance Areas is given to the Balance Responsible Parties based on their bidding to the Market Operator. Trade between Market Balance Areas is carried out through the Market Operator.
7. **Control Area**. The composition of one or more Market Balance Areas under the same technical load frequency control responsibility.
8. **Control Block**. The composition of one or more Control Areas, working together to ensure the load frequency control on behalf of Regional Group Continental Europe (RGCE) System Operation Committee.
9. **Control Entity**. A geographic area consisting of one or more Metering Grid Areas with an energy delivery responsibility. Each area is synchronously connected to another area. In most cases such areas have a load frequency responsibility and therefore may have to report to a higher level control entity.
10. **Coordination Center Zone**. The composition of a number of Control Blocks under the responsibility of the same Coordination Center Operator.
11. **Functional Group**. A collection of Metering Points for consumption and generation within a Market Balance Area.

12. **ITC**. The Inter TSO Compensation (ITC) market is composed of a group of System Operators that accept a common set of rules for the invoicing of energy flows over the border.
13. **Local Market Area**. A Market Area where there is no transmission capacity restrictions between the Market Balance Areas.
14. **Market Area**. An area made up of several Market Balance Areas interconnected through AC or DC links. Trade is allowed between different Market Balance Areas with common market rules for trading across the interconnection.
15. **Market Balance Area**. A geographic area consisting of one or more Metering Grid Areas with common market rules for which the settlement responsible party carries out a balance settlement and which has the same price for imbalance. A Market Balance Area may also be defined due to bottlenecks.
16. **Meter**. A physical device containing one or more registers.
17. **Metering Grid Area**. A Metering Grid Area is a physical area where consumption, production and exchange can be metered. It is delimited by the placement of meters for period measurement for input to, and withdrawal from the area. It can be used to establish the sum of consumption and production with no period measurement and network losses.
18. **Metering point**. An entity where energy products are measured or computed.
19. **National Area**. An area covered by a single set of national electricity arrangements established at government level. This is not necessarily the same as the geographical boundaries of a nation.
20. **Register**. A physical or logical counter measuring energy products.
21. **Reserve Object**. A resource technically pre-qualified using a uniform set of standards to supply reserve capabilities to a System Operator associated with one or more Metering Points and tele-measuring devices.
22. **Resource Object**. An object that represents a grid asset, a consumption resource or a production resource related to the energy industry. A Resource Object can represent for example a generating unit, a consumption unit or a virtual power plant defined in a contract.
23. **RGCE Interconnected Group**. The composition of a number of coordination center zones, operating under RGCE rules, where the exchange and compensation programmes within the zone must sum up to zero."

Regarding roles, the following roles are described [17]:

1. "**Balance Responsible Party**. A party that has a contract proving financial security and identifying balance responsibility with the Imbalance Settlement Responsible of the Market Balance Area entitling the party to operate in the market. This is the only role allowing a party to nominate energy on a wholesale level. The meaning of the word "balance" in this context signifies that the quantity contracted to provide or to consume

must be equal to the quantity really provided or consumed. This term is equivalent to "market agent" in Spain.

2. **Balance Supplier**. A party that markets the difference between actual metered energy consumption and the energy bought with firm energy contracts by the Party Connected to the Grid. In addition the Balance Supplier markets any difference with the firm energy contract (of the Party Connected to the Grid) and the metered production. There is only one Balance Supplier for each Accounting Point.
3. **Billing Agent**. The party responsible for invoicing a concerned party.
4. **Block Energy Trader**. A party that is selling or buying energy on a firm basis (a fixed volume per market time period).
5. **Capacity Coordinator**. A party, acting on behalf of the System Operators involved, responsible for establishing a coordinated Offered Capacity and/or Net transfer capacity (NTC) and/or Available transfer capacity (ATC) between several Market Balance Areas.
6. **Capacity Trader**. A party that has a contract to participate in the Capacity Market to acquire capacity through a Transmission Capacity Allocator. Note: The capacity may be acquired on behalf of an Interconnection Trade Responsible or for sale on secondary capacity markets.
7. **Consumer**. A party that consumes electricity.
8. **Consumption Responsible Party**. A party who can be brought to rights, legally and financially, for any imbalance between energy nominated and consumed for all associated Accounting Points. This is a type of Balance Responsible Party.
9. **Control Area Operator**. Responsible for : 1. The coordination of exchange programs between its related Market Balance Areas and for the exchanges between its associated Control Areas. 2. The load frequency control for its own area. 3. The coordination of the correction of time deviations.
10. **Control Block Operator**. Responsible for : 1. The coordination of exchanges between its associated Control Blocks and the organisation of the coordination of exchange programs between its related Control Areas. 2. The load frequency control within its own block and ensuring that its Control Areas respect their obligations in respect to load frequency control and time deviation. 3. The organisation of the settlement and/or compensation between its Control Areas.
11. **Coordination Center Operator**. Responsible for : 1. The coordination of exchange programs between its related Control Blocks and for the exchanges between its associated Coordination Center Zones. 2. Ensuring that its Control Blocks respect their obligations in respect to load frequency control. 3. Calculating the time deviation in cooperation with the associated coordination centers. 4. Carrying out the settlement and/or compensation between its Control Blocks and against the other Coordination Center Zones.

12. **Data Provider**. A party that has a mandate to provide information to other parties in the energy market. For example, a data provider may be a Transmission System Operator or a third party agreed by a TSO.
13. **Grid Access Provider**. A party responsible for providing access to the grid through an Accounting Point and its use for energy consumption or production to the Party Connected to the Grid.
14. **Grid Operator**. A party that operates one or more grids.
15. **Imbalance Settlement Responsible**. A party that is responsible for settlement of the difference between the contracted quantities and the realised quantities of energy products for the Balance Responsible Parties in a Market Balance Area. The Imbalance Settlement Responsible has not the responsibility to invoice, it may delegate the invoicing responsibility to a more generic role such as a Billing Agent.
16. **Interconnection Trade Responsible**. He is a Balance Responsible Party or depends on one. Recognised by the Nomination Validator for the nomination of already allocated capacity.
17. **Market Information Aggregator**. A party that provides market related information that has been compiled from the figures supplied by different actors in the market. This information may also be published or distributed for general use. The Market Information Aggregator may receive information from any market participant that is relevant for publication or distribution.
18. **Market Operator**. The unique power exchange of trades for the actual delivery of energy that receives the bids from the Balance Responsible Parties that have a contract to bid. The Market Operator determines the market energy price for the Market Balance Area after applying technical constraints from the System Operator. It may also establish the price for the reconciliation within a Metering Grid Area.
19. **Meter Administrator**. A party responsible for keeping a database of meters.
20. **Meter Operator**. A party responsible for installing, maintaining, testing, certifying and decommissioning physical meters.
21. **Metered Data Aggregator**. A party responsible for the establishment and qualification of metered data from the Metered Data Responsible. This data is aggregated according to a defined set of market rules.
22. **Metered Data Collector**. A party responsible for meter reading and quality control of the reading.
23. **Metered Data Responsible**. A party responsible for the establishment and validation of metered data based on the collected data received from the Metered Data Collector. The party is responsible for the history of metered data for a Metering Point.
24. **Metering Point Administrator**. A party responsible for registering the parties linked to the metering points in a Metering Grid Area. He is also responsible for maintaining the Metering Point technical specifications. He is responsible for creating and terminating metering points.

25. **Merit Order List (MOL) Responsible**. Responsible for the management of the available tenders for all Acquiring System Operators to establish the order of the reserve capacity that can be activated.
26. **Nomination Validator**. Has the responsibility of ensuring that all capacity nominated is within the allowed limits and confirming all valid nominations to all involved parties. He informs the Interconnection Trade Responsible of the maximum nominated capacity allowed. Depending on market rules for a given interconnection the corresponding System Operators may appoint one Nomination Validator.
27. **Party Connected to the Grid**. A party that contracts for the right to consume or produce electricity at an Accounting Point.
28. **Producer**. A party that produces electricity. This is a type of Party Connected to the Grid.
29. **Production Responsible Party**. A party who can be brought to rights, legally and financially, for any imbalance between energy nominated and produced for all associated Accounting Points. This is a type of Balance Responsible Party.
30. **Reconciliation Accountable**. A party that is financially accountable for the reconciled volume of energy products for a profiled Accounting Point.
31. **Reconciliation Responsible**. A party that is responsible for reconciling, within a Metering Grid Area, the volumes used in the imbalance settlement process for profiled Accounting Points and the actual metered quantities. The Reconciliation Responsible may delegate the invoicing responsibility to a more generic role such as a Billing Agent.
32. **Reserve Allocator**. Informs the market of reserve requirements, receives tenders against the requirements and in compliance with the prequalification criteria, determines what tenders meet requirements and assigns tenders.
33. **Resource Provider**. A role that manages a resource object and provides the schedules for it.
34. **Scheduling Coordinator**. A party that is responsible for the schedule information and its exchange on behalf of a Balance Responsible Party.
35. **System Operator**. A party that is responsible for a stable power system operation (including the organisation of physical balance) through a transmission grid in a geographical area. The System Operator will also determine and be responsible for cross border capacity and exchanges. If necessary he may reduce allocated capacity to ensure operational stability. Transmission as mentioned above means "the transport of electricity on the extra high or high voltage network with a view to its delivery to final customers or to distributors. Operation of transmission includes as well the tasks of system operation concerning its management of energy flows, reliability of the system and availability of all necessary system services." (definition taken from the ENTSO-E RGCE Operation handbook

Glossary). Additional obligations may be imposed through local market rules.

36. **Trade Responsible Party**. A party who can be brought to rights, legally and financially, for any imbalance between energy nominated and consumed for all associated Accounting Points. A power exchange without any privileged responsibilities acts as a Trade Responsible Party. This is a type of Balance Responsible Party.
37. **Transmission Capacity Allocator**. Manages the allocation of transmission capacity for an Allocated Capacity Area. For explicit auctions: The Transmission Capacity Allocator manages, on behalf of the System Operators, the allocation of available transmission capacity for an Allocated capacity Area. He offers the available transmission capacity to the market, allocates the available transmission capacity to individual Capacity Traders and calculates the billing amount of already allocated capacities to the Capacity Traders."

It could be concluded that the process of unbundling has brought an increase in the complexity of the system. Probably some of the listed roles and domains were also present in the bundled systems, managed by oligopolies and State-owned companies. However the liberalization process has made clear that the interaction among all the different agents and roles must be defined, and the communication must be made making use of international standards that assure systems' interoperability.

2.2 The electrical energy network and the Smart Grid

The classical topology approach divides the electrical energy network, according to voltage levels, in the following distribution:

- Very high voltage network: voltages with more than 145 kV, from the central generation plants.
- High voltage network: with voltages between 36 kV and 145 kV.
- Medium voltage (MV) network: with voltages between 1 kV and 36 kV.
- Low voltage (LV) network: with voltages lower than 1 kV.

Very high and high voltage networks are typically associated to the transmission network, with the objective to transport the electricity from the sites where it is being generated to the electrical substations where the energy is transformed at MV level and dispatched through distribution networks to final users, such as industrial consumers, or at LV levels to residential, medium and small enterprises or services consumers. Figure 2.1 depicts as an example a diagram of the Spanish power distribution grid with the typical voltage values.

Transmission networks are usually meshed, i.e., with redundant physical connections, whereas distribution networks are typically operated in a radial way, where from each feeder from a primary substation the energy is distributed following a tree structure, in branches and nodes [18]. The typical large production plants that can be found in generation systems are the following:

- Hydroelectric generation.
- Thermal plants.
- Combined cycle plants.
- Nuclear thermal plants.
- Solar farms.
- Wind farms.

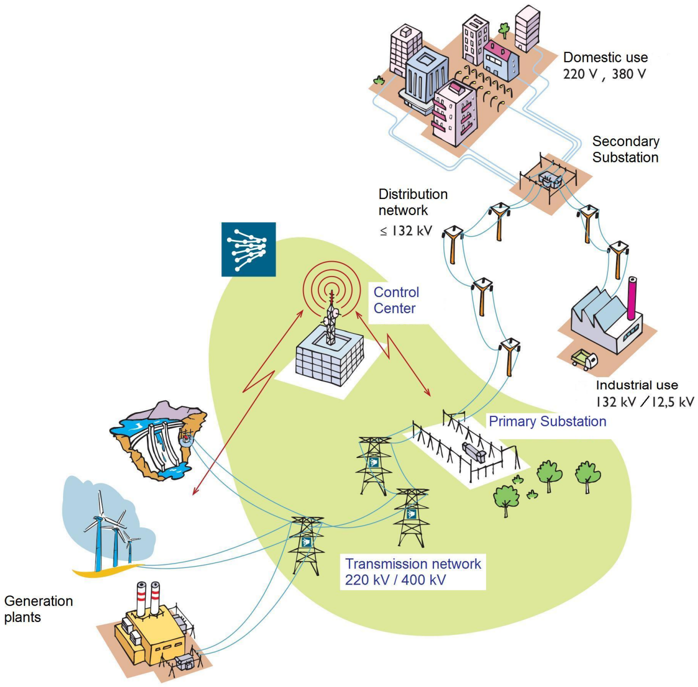


Fig. 2.1. Diagram of the Spanish power distribution, with the typical voltage levels in generation, transmission and distribution. Source: Red Eléctrica de España (REE), the Spanish TSO. Translated to English by Ignacio Benítez.

This scenario, however, is changing. Due to the increasing number of connections of distributed generation, mainly to the medium voltage network, with a range of different installed power magnitudes (typically from a few kW up to 50 MW), the power flow can no longer be restricted to just one direction (from central generation to final customers) but can be bidirectional, at least at a MV network level. This means that the power grids must be prepared for bidirectional generation and consumption flows. There is, therefore, a need to reinforce the existing infrastructure with protection components able to operate in both directions, and to integrate monitoring devices and measuring equipment in lower voltage levels of the grid. Information and Communication Technologies (ICT) become the backbone of this system of the future, and will be the base for new Information Technology (IT) systems, able to analyze large amounts of distributed data and provide advanced services, such as the prediction of events and voltage or power levels for given conditions; identification of patterns in energy data; or condition monitoring for the management of assets. All these systems, being integrated at generation, transmission and distribution levels, are the components of the future **Smart Grids** [19]. A Smart Grid does not currently have a unified description. Following, two descriptions are included. The first one is from the Electric Power Research Institute (EPRI), in the United States, and the second one is from the European Commission:

> “A Smart Grid is one that incorporates information and communications technology into every aspect of electricity generation, delivery and consumption in order to minimize environmental impact, enhance markets, improve reliability and service, and reduce costs and improve efficiency.” (http://smartgrid.epri.com/)

> “A Smart Grid is an electricity network that can cost efficiently integrate the behaviour and actions of all users connected to it – generators, consumers and those that do both – in order to ensure economically efficient, sustainable power system with low losses and high levels of quality and security of supply and safety.” (EC M/490)[20]

As can be seen in Fig. 2.2, the main applications provided by the Smart Grids, besides power quality and reliability, are the following [21]:

- Demand Response and Dynamic Pricing.
- Distributed Generation and Alternate Energy Sources.
- Self-Healing Wide Area Protection and Islanding.
- Asset Management and On Line Equipment Monitoring.
- Real-Time Simulation and Contingency Analysis.
- Participation in Energy Markets.

To achieve these objectives, The International Energy Agency defined, in year 2011, the following technology areas where the Smart Grids rely on:

- Wide Area Monitoring and Control.
- Integration of ICT.
- Integration of renewable and distributed generation.

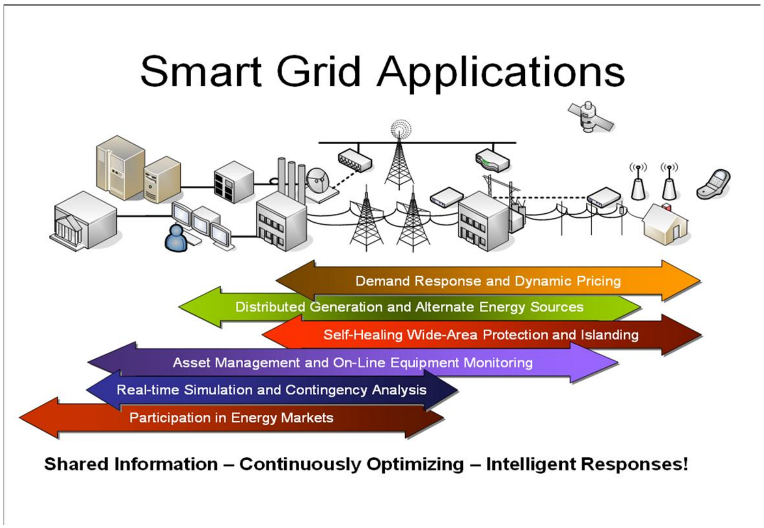

Smart Grid Applications
Demand Response and Dynamic Pricing
Distributed Generation and Alternate Energy Sources
Self-Healing Wide-Area Protection and Islanding
Asset Management and On-Line Equipment Monitoring
Real-time Simulation and Contingency Analysis
Participation in Energy Markets
Shared Information – Continuously Optimizing – Intelligent Responses!

Fig. 2.2. Smart Grid applications in the future schema of electricity generation, transmission and distribution. Source: MyISM3004 blog and Sean Dempsey, 2011.

- Transmission enhancement applications.
- Distribution grid management.
- Advanced Metering Infrastructure (AMI).
- Electric Vehicle (EV) charging infrastructure.
- Customer-side systems.

Figure 2.3 depicts these areas and the power grid level that they adhere to. Cybersecurity has also been added to these objectives in the last years [22], as a major and increasing concern, as more information and data from assets and consumers is being measured and managed.

2.3 Distributed Energy Resources (DER)

In recent years, distributed generation is gaining relevance, as the number of dispersed, medium power generation plants connected to the medium voltage network, increases. These plants obtain their generation mostly from renewable energy resources, such as wind, photovoltaic or solar energy. Therefore, a production uncertainty is associated to this generation, adding another issue of complexity to be dealt with.

Currently there is not a unique, established definition for distributed generation. Trebolle [23] gathers some definitions given by different authors and associations, such as the following:

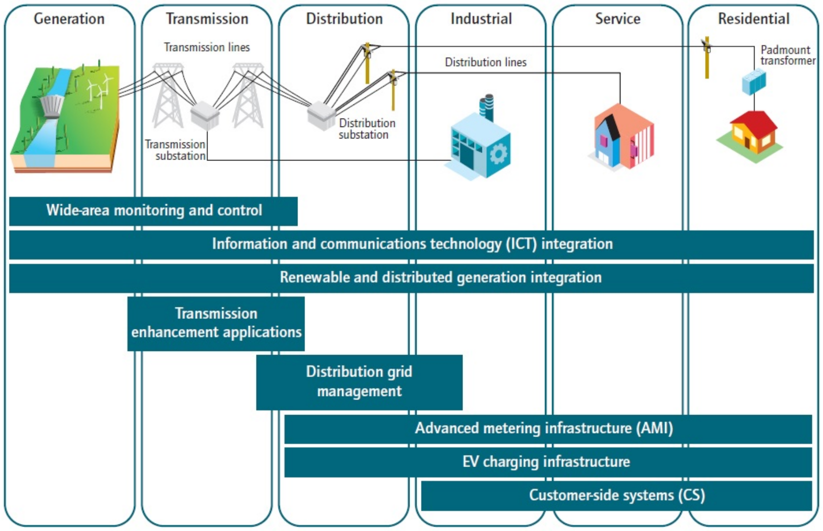

Generation
Transmission
Distribution
Industrial
Service
Residential
Transmission lines
Distribution lines
Padmount transformer
Distribution substation
Transmission substation
Wide-area monitoring and control
Information and communications technology (ICT) integration
Renewable and distributed generation integration
Transmission enhancement applications
Distribution grid management
Advanced metering infrastructure (AMI)
EV charging infrastructure
Customer-side systems (CS)

Fig. 2.3. Technology areas in the Smart Grids. Source: Smart Grids Roadmap, International Energy Agency, 2011.

> “Distributed generation is an electric power source connected directly to the distribution network or on the customer site of the meter.” [24]
>
> “Distributed generation is (a) generating plant serving a customer onsite or providing support to a distribution network, connected to the grid at distribution-level voltages.The technologies generally include engines, small (and micro) turbines, fuel cells, and photovoltaic systems. It generally excludes wind power, since that is mostly produced on wind farms rather than for on-site power requirements.”[25]
>
> “Some common attributes of embedded or dispersed generation may be listed as:
> - Not centrally planned (by the utility)
> - Not centrally dispatched
> - Normally smaller than 50 – 100 MW
> - Usually connected to the distribution system
>
> The distribution system is taken to be those networks to which customers are connected directly and which are typically of voltages from 230/400 V up to 145 kV.”[26]

Jenkins et al. [26] make use of the definitions *embedded* and *dispersed* generation, to indicate a generation that is embedded in the distribution network and does not come from a central generation.

As stated by Ackermann, Andersson and Söder [24], the environmental impact is not considered in the definition of distributed generation, therefore both renewable and fossil-fuel based resources are included. These authors establish categories for distributed generation according to the generation power. These categories are the following:

- Micro: up to 5 kW
- Small: from 5 kW to 5 MW
- Medium: from 5 MW to 50 MW
- Large: more than 50 MW

All these definitions have two characteristics in common: 1) they all refer to the distributed nature of the generation, and 2) the connection to the network is done at the distribution level. However, the definition and characteristics for distributed generation must be approached under the scope of the specific regulations and network configurations for each country. The following characteristics can be considered, however, as common issues of the distributed generation [23]:

- Usually connected to the distribution network.
- The generation can be either totally injected to the grid or partially consumed locally.
- There is no centralized management of the production nor dispatchability management.
- Power is usually less than 50 MW.

Regarding the different distributed energy resources, a distinction can be made in those classified as renewable and non-renewable energies. From a management and operation perspective, these resources can be classified in controllable resources and non-controllable ones. Controllable resources are those whose generation can be controlled. Non-renewable sources are all controllable. Non-controllable resources are typically of a renewable nature; their production although predictable to a variable extent, has an uncertainty associated, and cannot be directly controlled by an operator, unless other complementary resources, such as an energy storage source, is integrated.

The current DER that can be found in the literature [27] are listed and briefly described below. Their maturity or technology readiness level is also indicated:

- Commercially available technology:
  - Combustion engines.
  - Gas turbines.
  - Small Hydro.
  - Wind.
  - Solar thermal.
  - Photovoltaic.
- Commercially available, but not fully mature technology:
  - Biomass.
  - Microturbines.

  - Fuel Cells.
- Emergent technologies:
  - Marine.
  - Geothermal.

Combustion engines generate electricity based on the rotational movement powered by a thermal engine, typically fueled by diesel or gas. They are versatile and controllable generators, mostly used in industries for cogeneration or Combined Heat and Power (CHP). However, they are powered by fossil fuels and their combustion implies a significant impact in emissions and carbon footprint.

Gas turbines are rotational machines propelled by gas expansion as a result of the combustion of compressed air and gas. Their usual Coefficient Of Performance (COP) is around 40%. This value, however, can be improved if the turbine is enhanced with extra devices:

- CHP: the turbine retrieves heat from the exhaust gases. An improvement of up 60% in the COP can be obtained.
- Combined cycle: A steam turbine is added to the exit of the gas turbine, that makes use of the heat from the exhaust gases to generate more electricity with water steam. Usual COP of these combined cycles are between 70 and 90%.

Gas turbines are powered by fossil fuels and, even though they might improve the carbon footprint compared to combustion engines, they also have a significant impact in pollutant emissions.

Small Hydraulic generation, or mini-hydro generation, addresses stations of hydraulic generation of less than 10 MW of nominal power, being this power a function of the available volumetric flow of water and the available height or elevation.

The main components in mini-hydro generation plants do not greatly differ from those of high-power hydraulic plants. The generation is produced by the transmission of the potential energy from water to kinetic energy in a water turbine, characterized by the available flow and elevation, and the type of installation: a water reservoir or a water stream. Based on these data, there are mainly three types of turbines for small hydro generation:

- Pelton: suitable for high elevations.
- Francis: suitable for medium elevations.
- Kaplan: suitable for low elevations and high flow volumes.

The main process of hydraulic generation produces no pollutant emissions, therefore it can be considered of a renewable nature. One of the main considerations of hydraulic plants, however, is their environmental impact, which must be carefully studied.

Small Wind generation units, although being a mature technology, are not common nowadays due to a number of reasons. Some of these reasons are:

- They have a significant visual impact. In urban environments, the mounted poles on roofs or isolated spaces contribute to a negative view of this kind of technology.
- A minimum amount of wind is required. In urban environments, the wind usually encounters more obstacles to reach the generators, which makes the wind behave as a turbulent flow when it finally reaches the blades. Besides, being installed at lower elevations, the wind can too low in speed and availability, therefore the number of productive hours per year are reduced.

These disadvantages have allowed that solar photovoltaic and solar thermal distributed generation have gained more presence in urban environments than wind, since their integration in rooftops and facades allow a minor visual impact and the availability of sun is usually more constant than that of wind. The technology of vertical axis generation has allowed to mitigate this gap. This technology implements the blades directly on the vertical pole, which becomes rotational, removing the big blades present in horizontal axis generation.

Small wind generation, as the big wind power plants, are renewable energy sources, and therefore present no emissions nor carbon footprint.

Solar Thermal technology converts the energy from the Sun, by radiation, to heat transferred to a fluid, usually water. The hot water can be used for generation of electricity or for thermal uses. Their application as DER is mainly for this last purpose, to be used in domestic hot water supply or for HVAC (Heating, Ventilation and Air Conditioning) systems. These are called low temperature applications, below 100 ºC

Solar thermal energy used for production of electricity are large production plants with a capacity production that can vary between 80 and 200 MW and reach up to 800 ºC. In these systems, solar thermal power uses various means, such as parabolic trough, parabolic dish, or solar power tower to generate heat, with the water being converted into steam that will be used to drive a conventional steam turbine to produce electricity [2].

Photovoltaic energy is the most extended DER due to its ease of maintenance and its lower visual impact compared to other technologies such as wind generation, among other reasons. Photovoltaic energy makes use of the Sun's radiation to induce a differential of potential between nodes of a diode or solar cell. When these cells are connected on an electrical circuit wire, the energy is obtained, in form of Direct Current (DC) which must be converted to AC by means of inverters. The performance of photovoltaic systems is usually low (around 10-20%), and they require space for the solar panels to be mounted, facing to the Sun with the appropriate angle. Its use, however, is very extended due to the simplicity in maintenance and management and the availability of rooftops and facades for the installation, from domestic users to small and medium enterprises and big factories.

Biomass energy is the energy derived from organic matter. It is usually used in a DER level for CHP generation, or as a source of heat in domestic hot water or HVAC systems. Although biomass generates about the same amount of carbon dioxide emissions when burned as do fossil fuels, its net emission is considered to be zero since its origin is organic, from plants that have previously removed carbon dioxide from the atmosphere [2].

Microturbines are small size turbines with power generation levels that vary between 25 and 500 kW. Most of them use natural gas as the primary fuel. The main difference with gas turbines is their size and their design, suitable for small-scale applications of DER. Microturbines present a compact design which incorporates all the components and devices needed for the electrical generation, compressor, combustor, turbine, and generator, all along the same shaft. The most typical applications of Microturbines include [2]:

- Peak shaving and base load power (grid parallel).
- Combined heat and power.
- Stand-alone power.
- Backup/standby power.
- Ride-through connection.
- Primary power with grid as backup.
- Microgrid.

Due to their design, microturbines present very low emissions. However, these are turbines powered mostly by gas and, therefore, of a non-renewable nature.

Fuel cells produce electricity by an electro-chemical reaction (called inverse electrolysis) of a hydrogen-rich fuel and an oxidizer (such as oxygen or air). In the reaction, exhaust gases or fluids may be obtained (like water in an ideal reaction). There are different types of fuel cells, mainly classified by the type of electrolyte used between anode and cathode in the cell. Some of the main classes of fuel cells are the following [2]:

- Proton Exchange Membrane Fuel Cells (PEMFCs). Characterized by a solid polymer membrane which separates the anode and the cathode. Temperature of operation ranges between 70 and 90 ºC.
- Phosphoric Acid Fuel Cells (PAFCs). With a liquid electrolyte, the temperature of operation is around 220 ºC.
- Alkaline Fuel Cells (AFCs) were the first workable power unit used by NASA in the manned Apollo spaceship mission. With liquid electrolyte, their characteristics and operation temperature are similar to PEMFC.
- Molten Carbonate Fuel Cells (MCFCs) use as electrolyte a mixture of alkali carbonates of potassium and lithium. Their operating temperature is around 650 ºC.
- Solid Oxide Fuel Cells (SOFCs). uses yttria and zirconia as oxides to operate at high temperatures (around 1000 ºC). This cell has an entirely solid-state construction and a single side tube sealing.

- Direct Methanol Fuel Cells (DMFCs). In these cells, methanol is oxidized electrochemically by water at the anode to produce carbon dioxide and positive and negative ions, in contrast to what happens in a PEMFC, where the positive ions from the hydrogen are supplied directly to the anode. Their main advantage is the use of methanol as fuel, therefore the transport and storage of hydrogen is not required.

high-temperature operating cells are designed for large-scale electric power generation. low-temperature cells, such as PEMFCs, AFCs, DMFCs, are suitable for their use as DER and portable generation units, and also in transport, in fuel cell powered vehicles.

Marine technologies are emergent developments that make use of the kinetic and potential energy present in oceans and rivers to generate electricity. The main sources are the following:

- Tidal energy, which make use of tides and the gradient in water levels in a tide cycle.
- Marine flows, make use of the natural flows in seas and oceans.
- Waves energy, which make use of the waves.

Although some operational tidal energy plants can be found in the world, this technology is still in an early stage of development. A number of prototypes and R&D projects can be found regarding marine flows and waves generation.

Geothermal energy makes use of the latent heat that can be found in the inner layers beneath the Earth's surface. By circulating liquid such as water through these layers, the heat is transferred for either direct generation of electricity or for hot water systems, such as domestic hot water or HVAC. This is the most common case in geothermal DER, therefore its use as a direct generation source is at higher power levels.

2.4 Renewable energies and their integration in the grid

Regarding technical issues for connection of Renewable Energy Sources or RES, Trebolle [23], based on the Spanish Regulations, lists the following:

- The voltage level and network where the source will be connected depends of the installed power. The power limit connection to the LV network is established in 100 kVA; if the sum of net generators' power is more than this amount, the plant will have to be connected to the MV network.
- According to voltage level, the generator power and the shortcircuit power of the node where the source is being connected, there is a fixed voltage variation allowed ($\pm V(\%)$), which cannot be exceeded.
- A minimum of evacuation capacity is requested, due to network requirements, usually stated as a percentage of the line or transformer's nominal capacity ($RD2818/1998$).

- Regarding security and protection issues, little regulation is found among the different Spanish Orders. One is the $RD1663/2000$, which states that photovoltaic plants of up to 100 kVA connected to LV, should disconnect when no voltage is detected at the line (islanding situation). Concerning wind power, in order to avoid fluctuations in the network, the injected power is limited to a maximum of 5% of the connection point shortcircuit power.

Regarding losses in the electricity network due to the presence of DER, the following issues are described by Trebolle [23]:

Proximity of DER to consumption nodes. The proximity of the energy resources to the consumption reduces the losses due to transportation.

Topology and grid structure. The different topologies of HV, MV and LV networks highly define the integrity and robustness of the network. Typically, in MV or LV networks with a radial structure, the losses are influenced by the distance where the DER is found with respect to both consumption nodes and feeders.

Penetration level. The mathematical relation between energy losses and the DER penetration level follows a "U" shape: few DER at a network with no generation improve the losses; however as the amount of DER increase, the trend is inverted and the losses become higher. This is the case, for instance, with wind generation; as the available power increased with new turbines and more installations, the need for transportation increased and so the losses at a distribution level.

Type and technology of DER. The origin and controllability of the DER plays an important role on the demand management and the losses at the network. A good prediction of production is needed for the provision of the energy dispatch for the energy market, taking into account the expected consumption. The energy from a DER can be obtained on a demand basis, such as with CHP, or can be based on renewable, uncontrollable resources (photovoltaic, wind energy), where the production is associated to a certain level of uncertainty. In order for the generation plant not being disconnected to avoid overproduction, an optimized schema of production and demand side balance must be established. Energy storage has been taken into account as an interesting option to approach this challenge in recent years.

Distributed energy resources contribute to keep voltage levels within the range (±7 % for MV and ±3 % for LV). However, the voltage level at a node with a DER connected is strongly dependant of its generation. This situation can turn in an unpredictable voltage deviation at the node, and an increase in maintenance costs of the primary substation feeding that node, which should automatically adjust its tap changers to correct the voltage level variation through the day [23].

Besides, specific issues in safety have been reported in MV networks with photovoltaic generation [28], turning in undesired islanding situations. In this case, the grid was disconnected from the primary substation feeder for maintenance operations, finding that voltage was still measured in different points of the MV network for periods of up to 40 minutes. A further analysis of this situation pointed out that, even with a power mismatch between generation and load in the network, there are some main factors that may negatively affect the functionality of the active anti-islanding systems of the inverters, making them inoperative [29]. These factors are, among others, the following:

- The variation of the protection delay trigger related to the interaction among the inverters.
- The presence of dynamic loads, such as asynchronous motors and voltage-dependent devices.
- The proper configuration of the gain factors (K) of the Sandia Frequency Shift (SFS) Islanding Detection Method and the Phase Locked Loop (PLL) systems in the inverters.
- The configuration and parametrization of the P-Q power control loop of the inverters.

2.5 Description of ancillary services

The Federal Energy Regulatory Commission (FERC) of the United States defined the **ancillary services** in 1995 as "those services necessary to support the transmission of electric power from seller to purchaser given the obligations of control areas[1] and transmitting utilities within those control areas to maintain reliable operations of the interconnected transmission system" [31]. The six ancillary services described by the FERC were:

- Reactive power and voltage control.
- Loss compensation.
- Scheduling and dispatch.
- Load following.
- System protection.
- Energy imbalance.

In 1996, authors Hirst and Kirby stated that the overall cost of ancillary services could roughly represent from 6 to 20% of the total generation and

[1] Control Area, as defined, for instance, by the Alberta Electric System Operator (Canada), means "a geographic area comprised of an electric system or systems, bounded by interconnection metering and telemetry, capable of controlling generation to maintain its interchange schedule with other control areas, and contributing to frequency regulation of the interconnection, all in accordance with the requirements of the WECC (Western Electricity Coordinating Council)." [30]

transmission costs in the United States [31]. These authors reviewed the number and characteristics of ancillary services and defined a list of seven basic ancillary services [3], from where more specific services, concerning management and control issues, can be extracted. The seven basic ancillary services, as described by Hirst and Kirby, are:

- Scheduling and dispatch.
- Load following.
- Operating reserves.
- Energy imbalance.
- Real power-loss replacement.
- Voltage control.
- Other services.

Scheduling and dispatch refer to the need of planning and adequately dispatching generated power to the end users or customers. The necessary operations to achieve this objective must include a real-time supervision of the system, conveniently scaled, along with real-time control of generation and transmission resources available at any time. Scheduling and dispatch are services associated to the system operator. Scheduling can have different time windows, such as a one week ahead basis, one day ahead, or just a few minutes ahead. Dispatching includes not only the control of all the available generation and transmission resources, but also the control strategies to keep their reliability within the control area.

Load following reserve service keeps instantaneous balance between load and resources by assuring a sufficient capacity of generation at the control area, at any time, plus a certain level of generating reserve, able to be used on the event of an increase in the load. Since variations in loads affect the frequency of the generators, frequency control, to keep the electric signal to a constant value, is one of the ancillary services related to this service. Other services that are closely related to load following are:

- Regulation.
- Energy imbalance.
- Spinning reserve.

Operating reserves are those generation units which can be used to balance the energy demand when an unexpected shortage in the connected generation occurs. These generation units can be divided in two, according to their response time:

- Reliability reserves: generation that can be used immediately and is fully available within the first ten minutes after the shortage. Generation reserves that are online are also referred as *spinning reserve*, whereas offline reserve, which can be connected and fully available within the first ten minutes, is called *non-spinning reserve*. The term "spinning" refers to the state of the generation being connected and, therefore, synchronized with the network. Their respond is usually based on an automated frequency control, responding to a variation in generation setpoints. The non-spinning reserve also applies to interruptible loads, which can be disconnected from the network within the first ten minutes due to balancing issues.

- Supplemental operating reserves: generation units that can provide power after ten minutes, and are fully available within the first thirty minutes after the generation shortage. These reserves supply energy for a longer term than reliability reserves.

Additionally, a third reserve, called *replacement reserve*, is also considered, whose function is to replace reliability and supplemental operating reserves for a longer term durability, until the contingency has been solved. Replacement reserve allow reliability and supplemental operating reserves to restore to their pre-contingency status [32].

Typically, operating reserves (and other ancillary services, such as load following and voltage control) are dimensioned for each Control Area, defined by the Independent System Operator (ISO) as a variable percentage of the current generation. Whereas load following reserve is considered as a continuous regulation and balancing of small variations in energy consumption, operating reserves are considered as a respond to more infrequent, but usually larger, energy shortages, due to generation and transmission failures.

Energy imbalance refers to the difference between the scheduled generation and the final consumed energy within a specific time frame, typically one hour. This imbalance must be provided by the System Operator, therefore an energy imbalance tariff is applied, to both generators and loads, that penalize deviations of energy consumption from the scheduled generation, thus promoting a better scheduling. Bandwidths may vary according to country legislation. For instance, in 2007, the FERC established a 10% penalty for energy imbalance that is outside the 1.5% deviation band [33]. The energy imbalance service is critical in the presence of renewable, non-controllable energy sources, such as wind and photovoltaics.

Real power-loss replacement refers to the the differences between generated real power and the real power delivered to customers [3]. These losses are mainly due to the transmission losses in the cables that cover the distance between the generator locations, either big production plants or DER, and the customer locations and demands. The values of the losses along the electrical network are usually estimated by the use of computer based models of the grid, and verified by distributed measurements. With the implementation of ICT and sensors at lower voltage levels of distribution, such as secondary substation and LV nodes, losses are becoming more accurately computed. These losses are dynamic and vary during the day, as a function of the energy demand and other factors such as weather conditions. Since the main losses are from the transmission network, the System Operator has to have enough available generation to balance the losses in real time.

Voltage control, or reactive power management, refers to the critical need of maintaining all the voltage levels at their pre-specified values (within an allowed tolerance) along the electrical network, from production to consumption. This is accomplished by the management of the injection and absorption of reactive power by generators and electronic equipments distributed along the network, known as FACTS (Flexible AC Transmission Systems). One typical FACTS equipment is the SVC (Static Var Compensator), able to store and dispatch reactive power automatically to maintain voltage at a reference level.

As indicated by Hirst and Kirby [3], "Enough reactive-power capacity must be available to meet expected demands plus a reserve margin for contingencies. Voltage drops are predominantly caused by the inductance of the lines and transformers (rather than the resistance), and can be compensated by supplying reactive power (conversely, too much reactive compensation can produce excessively high voltages). Because of the high inductance of lines and transformers, reactive power does not travel well through the transmission system, so reactive support must be provided much closer to reactive loads than real power needs to be provided to real loads."

Other services. The authors include here services that they consider beyond the basic services of energy supply, due to a number of reasons. These services include the following:

- Black start. This service addresses the situation of restarting power supply after a collapse, from generators that do not need to draw power from the grid to start supplying.
- Time correction. This service deals with the need to monitor and synchronize the generators frequency-based clocks, since small variations in frequency can incrementally drift from the real time.

Concerning **Spain**, ancillary services managed by the Spanish System Operator, Red Eléctrica de España (REE), are the following [34]:

- **Primary regulation**. This service deals with instantaneous balance between generation and load, within the first 30 seconds after an imbalance is measured. It is, therefore, a load following service, supplied by the velocity controls of the generators, by an adjustment of the setpoints.
- **Secondary regulation**. This service has as its objective to maintain the energy balance beyond the 30 seconds horizon, up to 15 minutes. This service is offered and managed by chosen generators that are available and with enough capacity at given time, grouped by control areas, therefore it is an operating reserve service, suited to already connected, fast-response generators, i.e., spinning reserves.
- **Tertiary regulation**. This service is designed for a longer term supply of energy, letting secondary regulation to be available again for future events

or imbalances. Its time frame comprehends from 15 minutes to at least 2 hours, therefore, this is a supplemental operating reserve service.

- **Deviation management**. The deviation management market is a service designed as a tool for REE to deal with prolonged energy imbalances which occur between two sessions of the intradiary market. It is intended as a service that allows REE to overcome an imbalance between energy and generation without having to extend the use of tertiary and secondary regulation, posing a risk on their availability.
- **Voltage control**. Its objective is to keep voltage control at transportation nodes, to assure signal quality, by the use of generators with more than 30 MW of nominal capacity or other regulated production units.
- **Black restart**. On the event of a national or regional system shutdown, the black restart service applies a sequence of generators injection to the grid, in order to restore it to its previous state.

2.6 Conclusions

As stated at the beginning of this Chapter, it seems clear that the liberalization process has brought to light the need to implement appropriate communication channels for the interaction among all the different agents, making use of international standards that assure systems' interoperability. Some of these standardization efforts are described in the following Chapter (Chapter 3).

The Smart Grids have arisen, as a new concept that gathers Information and Communication Technologies (ICT) as the backbone of the system, and Information Technology (IT) systems, able to analyze large amounts of distributed data and provide advanced services, such as the prediction of events and voltage or power levels for given conditions; identification of patterns in energy data; or condition monitoring for the management of assets.

One of the main challenges of the Smart Grids is the integration of Distributed Energy Resources (DER), especially Renewable Energy Sources (RES) in the power grid. Due to the uncertainty in energy generation and the dynamics of energy flows that can be given in the Points of Common Coupling (PCC) for the existing generation technologies, there is a need to increase the controllability of the network, which implies to increase the observability too, to measure (or estimate) the current network state. The integration of these energy sources have been associated in the literature to safety issues, due to unintentional islanding, and to an added complexity in the regulation of voltage and frequency and the provision of ancillary services.

3

The impact of Advanced Metering Infrastructure and the Smart Grids

3.1 Standardisation in the European Smart Grids

From the publication of the EC Directive regarding the liberalization of the energy markets in the European Union [5], as part of what has been called the "Third Energy Package", The European Commission, concerned with the lack of normalization and the use of common standards for truly interoperability in power systems and the Smart Grids of the future, has issued different mandates on standardisation efforts to European standardisation organisations (ESOs). The main three mandates are the following:

- M/441: Standardisation Mandate to CEN, CENELEC and ETSI in the field of measuring instruments for the development of an open architecture for utility meters involving communication protocols enabling interoperability [35].
- M/468: Standardisation Mandate to CEN, CENELEC and ETSI concerning the charging of electric vehicles [36].
- M/490: Standardization Mandate to European Standardisation Organisations (ESOs) to support European Smart Grid deployment [20].

The objective of the **M/441 Mandate** [35] is to define European standards that will enable interoperability of utility meters (electricity, water, gas and heat), either using existing standards as a base, or describing new ones. The mandate requires specifically the following:

1. A standard with a description on software and hardware open architectures for utility meters (known as Smart Meters) to support secure bidirectional communication flows through standardised interfaces and data exchange formats, allowing advanced management, monitoring and control services (i.e., allowing and Advanced Metering Infrastructure or AMI).
2. Other standards containing harmonised solutions for additional functionalities within an interoperable framework using where needed the abovementioned open architecture for communication protocols.

The European Standards Organisations (CEN, CENELEC, ETSI) formed the Smart Metering Coordination Group to work on this mandate. This work is still ongoing. A report was published in 2011 describing a proposal of functional reference architecture for communications in smart metering [37]. In parallel, the SMCG has been working on reviewing and selecting standards for smart metering in Europe; the same report lists existing communication standards to be used for smart metering and identifies the possible gaps where new standards should be developed. Above all of them, the DLMS - COSEM (Device Language Message Specification - COmpanion Specification for Energy Metering) and its standard specification, IEC 62056 [38], emerges as the main standard for smart metering object modelling, and it is currently the one being adopted by the PowerLine Communication (PLC) protocols developed for smart metering communication by the different alliances of manufacturers, vendors and DSOs. Such is the case, for instance, with PRIME (PoweRline Intelligent Metering Evolution) protocol [39], G3 [40], or, more recently, Meters & More [41].

The **M/468 Mandate** [36] encourages the Standardisation Organisations in Europe to develop or review existing standards to ensure interoperability and connectivity between the electricity supply point and the charger, and between the charger and the electric vehicle. Following this mandate, the eMobility Coordination Group (eM-CG) was creaated. Since then the eM-CG has been reporting the development of standards for electric mobility in Europe, such as the IEC 61851 [42], the IEC 62196 [43] and the ISO/IEC 15118 [44].

The IEC 61851 standard [42] describes four charging modes for electric vehicles:

- Mode 1: The installation requires earth leakage and circuit breaker protection. The power supply is of a maximum of 16 A , 230 V AC per phase, using standard common sockets.
- Mode 2: The installation requires earth leakage and circuit breaker protection, and a special device with a pilot control function. The power supply is of a maximum of 32 A , 230 V AC per phase.
- Mode 3: A specific Electric Vehicle Supply Equipment (EVSE) is needed, and also specific cable and connectors with dedicated communication pins. The power supply can reach higher values than modes 1 and 2, such as 64 A, 380 V AC.
- Mode 4: the charge is in Direct Current (DC), therefore specific components, such as cables, connectors and power converters are required.

The standard describes basic schemas for the implementation of the different modes and the hardware architecture for a basic control pilot function communication between the Electric Vehicle (EV) and the EVSE, based on Pulse Width Modulation (PWM). This pilot function allows the EVSE to signal the available electric power to the vehicle, thus enabling simple load control by an external energy controller. This function acts as a handshake mechanism prior to start supplying power.

The IEC 62196 standard [43] refers to the electrical connectors needed for each of the charging modes described in the IEC 61851, describing the electrical requirements. Connectors that fit with the descriptions in the standard have quickly become adopted by manufacturers, such as the Schuko model, the Yazaki connector or the Mennekes plug.

The ISO/IEC 15118 [44] standard specifies the communication between the Electric Vehicle (EV) and the EVSE, for smart charging purposes. This standard is beyond the simple handshaking mechanism of the control pilot function, allowing vehicle-to-grid capabilities and the EV demand management. The complete definition of the standard is still being developed. The last part published, Part 3, describes the physical and data link layer requirements. Part 2 describes the network and application protocol requirements, and Part 1 the general information and description of use cases.

The eM-CG has recognized the need to coordinate the standardisation requirements with Smart Grids Coordination Group (SGCG), created after the M/490 Mandate. A technical report issued by both groups [45] highlights the current gaps in communication standards between charging stations and central management systems, and among different management systems and e-mobility services providers. The development of standards based on web services data exchange is proposed in this report, such as the OCPP (Open Charge Point Protocol) [46].

The objective of the **M/490 Mandate** [20] is to "develop or update a set of consistent standards within a common European framework that integrating a variety of digital computing and communication technologies and electrical architectures, and associated processes and services, that will achieve interoperability and will enable or facilitate the implementation in Europe of the different high level Smart Grid services and functionalities as defined by the Smart Grid Task Force that will be flexible enough to accommodate future developments." The Mandate also states that there must be a close coordination with the outcomes of M/441 and M/468.

The European Standardisation Organizations (CEN, CENELEC, ETSI) formed the Smart Grid Coordination Group (SGCG) to address the requirements of this Mandate. Different works have been done by this Group, being the main ones the definition of a common framework for Smart Grid Use Cases, known as the SGAM, or Smart Grid Architecture Model [47], and the elaboration of a roadmap of standards for the Smart Grids, where the adoption of international standards, such as CIM [48] or IEC 61850 [49], is encouraged. These standards and the SGAM are briefly described next.

3.1.1 The Smart Grid Architecture Model (SGAM)

In year 2012, following the mandate of the European Commission EC-M/490 to European Standardisation Organizations, to support the development of the European Smart Grid [20], the SGCG described the Smart Grid Architecture Model (SGAM), with the objective to "present the design of smart grid use cases in an architectural but solution and technology neutral manner [50]". The model defines five layers, as can be seen in Fig. 3.1: component layer, communications layer, information layer, function layer and business layer. Each layer is spanned by smart grid domains and zones.

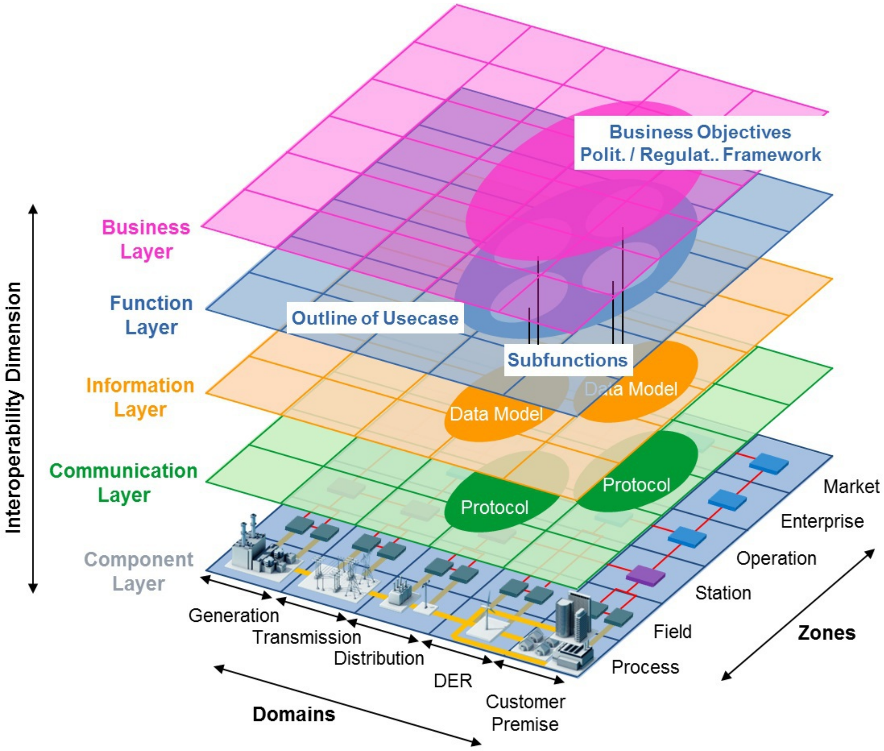


Fig. 3.1. The SGAM (Smart Grid Architecture Model). Source: Heise.de.

The *domains* are physically related to the electrical grid generation, transmission and distribution, including the presence of DER and the "Customer Premises" [50]. These last domains can include generation too.

The SGAM *zones* represent the hierarchical levels of power system management according to the standard IEC 62357-1 [51]. These zones reflect a hierarchical model that considers the concept of aggregation and functional separation in power system management. The SGAM zones, as defined in the SGAM User Manual [47], are the following:

1. "Process. Includes the physical, chemical or spatial transformations of energy (electricity, solar, heat, water, wind) and the physical equipment directly involved (e.g. generators, transformers, circuit breakers, overhead lines, cables, electrical loads, any kind of sensors and actuators which are part or directly connected to the process).
2. Field. Includes the equipment to protect, control and monitor the process of the power system, e.g. protection relays, bay controller, any kind of intelligent electronic devices which acquire and use process data from the power system.
3. Station. Representing the areal aggregation level for field level, e.g. for data concentration, functional aggregation, substation automation, local SCADA systems, plant supervision.
4. Operation. Hosting power system control operation in the respective domain, e.g. distribution management systems (DMS), energy management systems (EMS) in generation and transmission systems, microgrid management systems, virtual power plant management systems (aggregating several DER), electric vehicle (EV) fleet charging management systems.
5. Enterprise. Including commercial and organizational processes, services and infrastructures for enterprises (utilities, service providers, energy traders), e.g. asset management, logistics, work force management, staff training, customer relation management, billing and procurement.
6. Market. Reflecting the market operations possible along the energy conversion chain, e.g. energy trading, retail market."

For interoperability between systems or components, the SGAM consists of five *layers*, defined in the SGAM User Manual as the following [47]:

- "Business. The business layer represents the business view on the information exchange related to smart grids. SGAM can be used to map regulatory and economic (market) structures (using harmonized roles and responsibilities) and policies, business models and use cases, business portfolios (products & services) of market parties involved. Also business capabilities, use cases and business processes can be represented in this layer.
- Function. The function layer describes system use cases, functions and services including their relationships from an architectural viewpoint. The functions are represented independent from actors and physical implementations in applications, systems and components. The functions are derived by extracting the use case functionality that is independent from actors.
- Information. The information layer describes the information that is being used and exchanged between functions, services and components. It contains information objects and the underlying canonical data models. These information objects and canonical data models represent the common semantics for functions and services in order to allow an interoperable information exchange via communication means.
- Communication. The emphasis of the communication layer is to describe protocols and mechanisms for the interoperable exchange of information

between components in the context of the underlying use case, function or service and related information objects or data models.

- Component. The emphasis of the component layer is the physical distribution of all participating components in the smart grid context. This includes system & device actors, power system equipment (typically located at process and field level), protection and tele-control devices, network infrastructure (wired / wireless communication connections, routers, switches, servers) and any kind of computers."

The SGAM has been defined with the purpose to serve as a tool in three dimensions (layers, domains, zones) to test whether the Smart Grid use cases are supported by the existing standards, allowing to identify possible gaps. Mapping a system onto the SGAM will consist of:

1. The definition of the set of generic use cases the considered system can/-may support.
2. The drawing of the typical architecture and components used by this system (component layer).
3. A list of standards to be considered for interfacing each components within this system.

3.1.2 The IEC-61850 standard

The international standard IEC 61850 [49] originally described the general requirements for communications networks and systems in substations. However, its scope has broadened to include communication among power system elements under substation automation requirements, such as DER (Distributed Energy Resources) or ES (Energy Storage). The standard describes communication among Intelligent Electronic Devices (IED), over broadband Ethernet. The IEC 61850 abstracts the definition of data objects and services, therefore the models defined are independent of any underlying communication protocols [52].

However the standard is mapped to specific protocols, to comply with security and real time constraints, assuring interoperability in equipments from different vendors or manufacturers. Each IED may include different functionalities or services. The data objects or classes are named Logical Nodes or LN. An IED is formed by aggregating LNs for specific purposes, such as measurement, control and protection. The abstract services are defined in the ACSI (Abstract Common Services Interface) model. This model defines a set of services and the responses to those services that enables all IEDs to behave in an identical manner from the network behavior perspective [52]. The standard defines three hierarchical levels for substation automation:

- Process: the lowest level, integrates sensors, voltage and current transformers and actuators, needed to operate and monitor the substations.

- Bay: the intermediate level, includes control and protection systems with a higher abstraction level by the aggregation of processes.
- Substation: the highest level, where Human Machine Interfaces (HMI) and gateways to control centers are located, for supervision and management.

The following protocols are defined in the standard, to cover different functionalities and services:

- GOOSE/GSSE (Generic Object Oriented Substation Event / Generic Substation Status Event) at the station bus, between bay and substation levels.
- SV (Sampled Values) at the process bus, between process and bay levels.
- MMS (Manufacturing Message Specification), used to map the data object and services model to the manufacturing ISO/IEC standard 9506 [53], resulting in a unique and unambiguous reference for each data object and LN.

IEC 61850 defines also an XML based schema for configuration of the elements to be integrated in the network. This schema is called SCL (Substation Configuration Language).

3.1.3 The Common Information Model (CIM)

The Common Information Model (CIM) is an abstract information model that can be used to model an electrical network and the various equipment used on the network [54]. This model covers three primary objectives [48]:

1. To facilitate the exchange of power system network data between organizations.
2. To allow the exchange of data between applications within an organization.
3. To exchange market data between organizations.

The model definition is currently covered in two IEC standards:

- IEC 61970 [55]: describes the Common Information Model from the perspective of Energy Management Systems (EMS).
- IEC 61968 [56]: describes specific applications of CIM for Distribution Management Systems, Outage Management Systems and Work Management Systems.

An information model is an abstract and formal representation of objects, their attributes, their associations to other objects, and the behavior and operations that can be performed on them. The modeled objects may be physical objects, such as devices on an electrical network, or they may themselves be abstract, such as the objects used in a customer information system [54]. In the CIM, the formal representation is made by the use of UML or Unified Modelling Language, a descriptive language which makes use of diagrams to model software systems.

CIM does not describe the contents of a model, but instead defines how to represent, in a structured way, the interaction among components in a power system. For this reason, CIM is suitable for Distribution Management Systems and SCADA of power networks.

3.2 The use of Big Data and Big Data Analytics in Power Systems

The current deployment of ICT solutions for the use of DMS or Distribution Management Systems for electrical transmission and distribution grid, and the integration of AMI in a growing number of customer premises, has brought to light the necessity of the development of adequate systems to gather, process and store huge amounts of data. For instance, one single smart meter can collect 96 quarter-hourly measures per day, 365 days per year, which yields an amount of 35040 values of energy per year, only for one client. Besides, data from different sources have different requirements in processing time and storing period. For instance, DMS data used in the operation of the power grid is required in real time, whereas consumption data from smart meters can be sent to warehouse systems once a day or a week [57]. The systems that are designed to manage big amounts of data, either in real time or off-line, are called "Big Data" systems, and the companion software systems that deal with the processing and analysis of these data are called "Big Data Analytics" systems [58]. Big Data and Big Data Analytics systems, as commented, must face three main technical challenges, as indicated by Hu et al. [59]:

- "How to collect and integrate the data from distributed locations.
- How to store and manage the gathered massive and heterogeneous datasets, while provide function and performance guarantee, in terms of fast retrieval, scalability, and privacy protection.
- How to effectively mine massive datasets at different levels in real time or near real time, including modeling, visualization, prediction, and optimization functionalities, that provided an added value to the manager and help in decision making."

Traditional data management and analysis systems, mainly based on Relational Database Management Systems (RDBMS), are not appropriate for Big Data, for the following main two reasons, also indicated by Hu et al. [59]:

1. "From the perspective of data structure, RDBMSs can only support structured data, but offer little support for semi-structured or unstructured data.
2. From the perspective of scalability, RDBMSs scale up with expensive hardware and cannot scale out with commodity hardware in parallel, which is unsuitable to cope with the ever growing data volume."

For these reasons, Big Data systems are built on new architecture paradigms, such as cloud computing, and database systems such as NOSQL (Not Only Structured Query Language) [60], which supports the management of unstructured data. Big Data Analytics systems rely on the Big Data systems to gather, process and analyze the data, which can be obtained from distributed, scalable resources, or from the cloud, either in real time (i.e. "Stream" data) or off-line, even though stream mining systems are still in an initial phase of development [61]. Big Data Analytics systems include three main functionalities [59]:

- "Data visualization. In general, charts and maps help people understand information easily and quickly. However, as the data volume grows to the level of big data, traditional spreadsheets cannot handle the enormous volume of data. Therefore, new techniques and visualizations are needed to handle the description of the Big Data analysis.
- Statistical analysis. Statistical analysis can serve two purposes for large data sets: description and inference. Descriptive statistical analysis can summarize or describe a collection of data, whereas inferential statistical analysis can be used to draw inferences about the process.
- Data mining. Described as the computational process of discovering knowledge in large data sets, as will be seen in the following chapters of this thesis. Usual algorithms for the processing of large data sets include classification, clustering, regression, statistical learning, association analysis and link mining."

The data can be processed, stored and analyzed in three different locations [62]:

- In dedicated servers, owned by the Utility or the TSO.
- In external servers in "the Cloud", with distributed cloud computing functionalities [63].
- In the sensors and local equipment, providing the devices with local intelligence able to store and process the data, and send only the necessary information, alleviating the congestion in the communication channel (and the storage of superfluous data).

Upon these physical locations, Big Data Analytics software apply their specific analyses by partitioning the data and the algorithms in clusters of data and tasks distributed among the available processors in commodity (i.e. rack-mounting) servers, and later gathering the results by making use of advanced file management systems [62]. These systems are easily scalable, depending on the number of available servers. Currently the most known and worldwide implemented systems are the ones based on Hadoop, supported by the Apache Software Foundation, and MapReduce, initially developed by Google [61]. MapReduce is a software tool for parallel processing of data; it initially *maps* the data across the available nodes, applies the analysis code, and then collates and *reduces* together the results. Hadoop was built on MapReduce and offers a framework for distributed and cloud computing by the advanced functionalities of its file management system, called HDFS (Hadoop Distributed File System), which is able to partition files in blocks and store them in different servers, assuring redundancy and coherence of the data [64].

3.3 Demand Side Management and Demand Response

Demand Side Management (DSM) and Demand Response (DR) are examples of the direct application of the AMI and ICT and the Smart Grids use cases. These techniques refer to the capability of altering the consumption patterns of a consumer through the day as a function of either internal or external setpoints or orders, serving for different purposes, such as avoiding consumption peaks on the most expensive hours of the day (thus saving money in the electricity bill) or as a response from an external agent (DSO or TSO) to balance energy consumption at transmission or distribution nodes, either by increasing or decreasing energy consumption. The first example fits in the use case of DSM, whereas the second one (the response to external orders from other agents) describes the objectives of DR.

The benefits of both DR and DSM have been reported in previous works, such as [65], and have been piloted in a great variety of scenarios with different tariffs and incentives [66][67]. To ensure DSM and DR functionalities, a number of systems must be implemented at different levels:

1. Appropriate monitoring and control devices must be installed, that allow to measure the energy flow, and to control in real time the production or consumption of energy from the specific components whose consumption can be managed.
2. The information must be sent and received in real time. Therefore specific communication architectures and standards must be implemented at the different levels: process, bay, system, etc.
3. Intelligent systems and interfaces at higher management levels must be deployed, that must have communication with DMS to monitor the state of the network, and to produce the appropriate signals or orders. Data mining techniques can also be needed to identify and classify customers based on the amount of power that can be managed and the expected response.

Besides these systems, DR devices that participate in the balance of the energy flow should have a presence in the energy market as an asset, which can be traded. However, the penetration of DR in energy markets in Europe has very different velocities, as described in the 2014/2015 DR Map in Europe Report prepared by the Smart Energy Demand Coalition (SEDC) [68]: from totally integrated in the market in France, to currently illegal in Spain.

3.4 Conclusions and identification of needs

The European Technology Platform on Smart Grids (ETP SG) has issued a report [69] on the research and development needs foreseen by the platform for the EC Horizon 2020 Research and Innovation Programme [70], for the years 2016 and 2017. One of the main challenges identified by the ETP SG is the **utilization of smart metering data**, from two perspectives: from the side of the consumer, who becomes actively involved in the management of the energy consumption, becoming, therefore, what is called a “prosumer”; and from the perspective of the grid manager, which incorporates the smart metering data analysis to the Distribution Management Systems to increase the observability and controllability of the network. According to the ETP SG, “a very large amount of data is being collected whose potential has been untapped”.

Data Mining techniques, as will be seen in Part II, are an effective tool to unlock this potential, and benefit from the analysis and knowledge extraction from large sets of data of smart metering. Within this functionality, a specific objective is also indicated as a short-term challenge by the ETP SG: to develop models of **segmentation** of the customers based on their flexibility or availability for demand side management programs. The identification of end users suitable for these programs must meet specific requirements in the shape of their energy consumption through the day: a minimum amount of average energy consumption per day; the presence of manageable loads, such as equipment or appliances; a great peak/valley relation, etc. These users can be found by applying segmentation techniques on the smart metering data, and selecting the most representative of the resulting groups, based on the values of specific indices [71].

There is, therefore, an identified need to segment the end users as a function of their shapes of daily energy consumption, also called **load profiles**, and to obtain **patterns** that allow to classify the users in a group based on how they consume the energy. The analysis of these patterns of use can provide an added knowledge for the development of future systems to improve the efficiency in the use of the energy and the reliability of the grid. The power system can benefit, for instance, from Demand Side Management or Demand Response programs. Besides, the inclusion of the end user (being in some cases an active user or *prosumer*) habits of energy use allows the study and development of specific Use Cases for the appropriate management of energy flows in the context of the future Smart Grids. As has been described, an standardization effort is in process in this sense, with the definition of the SGAM and the communication standards for power systems and a common data model to exchange information through the different layers and domains.

4

Conclusions and future trends

The previous Chapters have depicted a current scenario of transition, from a stationary perspective of the electrical energy network, towards a future scenario of totally unbundled systems, where a number of different agents interact daily in order to manage (and benefit from) the dispatch of energy, from production to consumption. In this changing environment, the Smart Grid paradigm is presented as the tool to serve these different purposes and objectives to be achieved. Smart Grids, as have been seen, refer not only to the deployment of the necessary ICT among all the levels of the electrical network, but to integrate information management and data analysis systems that can process and provide value from huge amounts of hourly available data to operators, managers, and all the other agents involved.

It has been highlighted in Chapter 1 that there are still some technical and regulatory limitations that may pose barriers for a truly accessible, interconnected and open European energy market. Specific issues regarding the fully integration of DER and RES and their operation in the grid are realized by the EC in its last communication on a public consultation process on a new energy market design [15]. The need for a common regulatory framework among all the member States has also been identified.

In Chapter 2, the complex scenarios derived from the liberalization process and the definition of different roles, and the operation and management of power grids with an increasing presence of DER and RES have been presented. The benefits of the Smart Grids have been described, as a new concept that gathers Information and Communication Technologies (ICT) as the backbone of the system, and Information Technology (IT) systems, able to analyze large amounts of distributed data and provide advanced services, such as the prediction of events and voltage or power levels for given conditions; identification of patterns in energy data; or condition monitoring for the management of assets. As a direct benefit, the Smart Grids increase the observability of the grid, therefore increasing its controllability, and allowing a better integration of the dispersed generation and RES.

Finally, in Chapter 3, a main need has been identified, as a "side effect" that comes with the integration of the Smart Grids: the adequate management of the information. As indicated by the European Technology Platform on Smart Grids (ETP SG), one of the main challenges in the short term is the **utilization of smart metering data** [69], or how to properly manage and benefit from large amounts of data that are being available. Data Mining techniques, as will be seen in Part II, are an effective tool to unlock this potential, and benefit from the analysis and knowledge extraction from large sets of data of smart metering. Within this functionality, a specific objective is also indicated as a short-term challenge by the ETP SG: to develop models of **segmentation** of the customers based on their flexibility or availability for demand side management programs. These users can be found by applying segmentation techniques on the smart metering data, and selecting the most representative of the resulting groups, based on the values of specific indices [71].

A need has been identified, to segment the end users as a function of their shapes of daily energy consumption, also called **load profiles**, and to obtain **patterns** that allow to classify the users in a group based on how they consume the energy. However, this analysis is limited to a single day. Since the smart metering data are time series formed by sequential measurements of energy through each hour or quarter of hour of the day, but also through each day, it becomes clear that the analysis of the data needs to be extended to consider the **dynamic evolution of the consumption patterns** through days, weeks, months, seasons, and even years. This is precisely the objective of this thesis, as will be comprehensively described in the last Part of this document (Part III).

Big Data and Big Data Analytics systems are being implemented by DSOs and TSOs for the purpose of data management [58]. As has been described in this Chapter, three are the main objectives of the Big Data Analytics systems: data visualization, statistical analysis and data mining. The present thesis addresses the visualization and data mining objectives and, more specifically, how to provide useful and valuable information in form of the classification of a high number of consumers in a reduced number of temporal patterns. This objective fulfills the demand to identify and classify consumers according to the way they are consuming the energy, taking into account the different variations in energy consumption habits, like the different weekdays or the season.

The following Chapters are devoted to this objective. First, a state of the art on data mining and clustering techniques is described. Then, a compilation of some of the main works found in the literature regarding data mining techniques applied to the analysis of load profiles, or daily energy consumption patterns, is provided. Finally, the development of this thesis is presented, in the form of specific techniques and proposals for data visualization and clustering of time series energy consumption data.

Part II

State of the art on data mining and clustering techniques

Having described the scenario and highlighted the main advantages of Big Data systems but also their main threats (i.e., the certainty of becoming lost immersed in a pool of data), the following Part describes which can be considered as the main path towards a solution: the structured and ordered analysis of the data, or, in other words, how to obtain the desired information from large sets of time series data (and how this desired information would look like).

For this reason, the broad research field of Knowledge Discovery in Databases (KDD) is presented in Chapter 5, and, as a main component of KDD, Data Mining objectives and techniques are described. The dynamic analysis of data, in form of sequences of items or time series data, is one of the objectives of the data mining, therefore a review of the objectives and techniques for dynamic data mining is also included in this Chapter.

In Part I, the specific need for the analysis of load profiles of energy consumption is identified and described: the need to evaluate trends in temporal patterns of energy consumption from end users in the electricity distribution network, grouping the users by their temporal electric signature, or the way the energy is used and its amount variation in time. This objective can be accomplished by the use of the **cluster** analysis, one of the objectives described in the data mining field, where the data are grouped according to mathematical similarity with no *a priori* knowledge of the resulting patterns nor how they should behave. Clustering techniques are extensively described and reviewed in Chapter 6. Since the data analyzed are time series, and the objective is to obtain dynamic or temporal patterns, **dynamic or time series clustering** techniques must be applied for this analysis. A current state of the art in these techniques is described in Chapter 7.

This review will help, as will be seen in the last Part, to address the specific need identified and where the techniques for the obtention of the patterns would fit in the classification of the data mining objectives and the current clustering algorithms found in the literature. As described in Chapter 7 in the conclusions Section, the current algorithms and techniques found in the literature do not address this need properly, therefore specific developments, built upon some of the techniques found, will be performed and described in the last Part of this document, Part III.

5

An introduction to data mining

The following chapter classifies and describes algorithms and techniques that deal with the analysis of large amounts of multidimensional objects or observations, with one generic objective: to extract useful information or relations that can be used to obtain further conclusions on the nature of the data. These relations or information from the data can pursue different hypothesis or methods, as well as the conclusions, but they all have one thing in common: in most of the cases, this information is hidden underneath the large sets of data, undetectable at first sight. The appropriate choice of the method or methods to be applied will draw solutions that will tell the expert whether his/her assumptions were right or not, and to what extent.

5.1 Data Mining and Knowledge Discovery in Databases

The term "data mining" can be found in the literature under a number of different definitions, all of them pointing to the same target, which is the analysis and extraction of useful information from large sets of data. The following descriptions can be fitted to this objective:

> "Data mining is the analysis of (often large) observational data sets to find unsuspected relationships and to summarize the data in novel ways that are both understandable and useful to the data owner [72]."

> "Simply stated, data mining refers to extracting or "mining" knowledge from large amounts of data [73]."

Data mining techniques have been described as the intermediate step within a bigger process, called *Knowledge Discovery in Databases* or KDD, described as *"the nontrivial process of identifying valid, novel, potentially useful and ultimately understandable patterns in data"* [74]. This process covers the whole sequence of knowledge extraction in large data sets or, in recent years, in large amounts of data from the Internet of Things (IoT) [75] and from Big Data systems [76].

The KDD process comprehends the following steps [73]:

1. Initially, the expert or data analyst studies the data set and makes an assumption or states a hypothesis regarding the objective of the knowledge discovery process, or the model or relationship that may be found among the different variables or features of the data objects. The subsequent analysis on the data will be based on these initial hypothesis, which would at the end be validated upon the results obtained, or not. If the results are not satisfactory, the expert will perform new assumptions or hypothesis, therefore starting the KDD process again.
2. Then, the *Data warehousing* process [77] is performed. This step of the analysis comprises all the techniques and procedures to conveniently process erroneous or missing values in the data, filter noise, produce specific queries to access the data, and rearrange the information in the desired format in order to be analyzed.
3. Following, the selected *Data mining* technique or techniques are applied, with the objective of extracting useful relations or information from the data, in form of *patterns*, *models*, *association rules* or other.
4. The next step is the *Visualization of results*, along with numerical and/or categorical indicators and information to help the expert to identify and evaluate the results.
5. Finally, the expert has to perform an *Interpretation of results*. In this step, the expert will validate the initial hypothesis made concerning the underneath models or patterns able to summarize the data, obtaining conclusions and extracting knowledge from the results. In the case these results are not satisfactory, new hypotheses can be formulated, which will imply the need for new data mining analyses, therefore starting the KDD process again.

Normalization is usually one method used in the data warehousing process to adequately scale the data being analyzed. A description of the most common techniques that are used for normalization can be found in Section 6.4. Concerning the treatment of missing information, Jain and Dubes [78] gather some simple methods to cope with missing data in objects from a database. These are the following:

1. Simply delete the objects with missing data.
2. Replace the missing value x_{ij} with the average of the jth feature of the k nearest neighbors of object x_i.
3. Compute the distance d_j between feature j as indicated in (5.1), and then compute the distance between objects as indicated in (5.2).

$$d_j = \begin{cases} 0 & \text{if } x_{ij} \text{ or } x_{kj} \text{ is missing} \\ x_{ij} - x_{kj} & \text{otherwise} \end{cases} \tag{5.1}$$

$$d(i,k) = \frac{d}{d - d_0} \sum d_j^2 \tag{5.2}$$

where d_0 is the number of features missing in x_i, x_j, or both, and d is the total number of features or dimensions. If there are no values missing, $d(i,k)$ is the Euclidean distance.

4. Compute the distance d_j between feature j as indicated in (5.3), where $\bar{d_j}$ is defined as the average distance between all pairs of objects along the feature j, indicated in (5.4).

$$d_j = \begin{cases} \bar{d_j} & \text{if } x_{ij} \text{ or } x_{kj} \text{ is missing} \\ x_{ij} - x_{kj} & \text{otherwise} \end{cases} \tag{5.3}$$

$$\bar{d_j} = \frac{2}{n(n-1)} \sum_{i=2}^{n} \sum_{k=1}^{i-1} \|x_{ij} - x_{kj}\| \tag{5.4}$$

Following, compute the distance between objects as given in (5.5).

$$d(i,k) = \sum d_j^2 \tag{5.5}$$

The KDD process makes use of many different algorithms and techniques from different research fields: pattern recognition and classification, statistics, data visualization, machine learning and artificial intelligence [79]. According to Mitra et al. [74], the issue that makes KDD different from other areas is that it is focused on knowledge discovery from large sets of data, including the overall process of *"storage and accessing such data, scaling of algorithms to massive data sets, interpretation and visualization of results, and the modeling and support of the overall human machine interaction"*. The following Sections in this Chapter describe the main techniques and algorithms used for data mining found in the literature.

In recent years, the research field on KDD and data mining has started to move towards the analysis of data in Big Data systems by means of Big Data Analytics tools [80]. The objectives are kept the same, but the techniques have been adopted and gone through an evolution process to be able to manage large amounts of unordered data from distributed sources, and especially in the case of stream data mining, where the amount of available data per unit of time makes unavailable the traditional schema of loading the data and then performing a centralized analysis on them [80].

5.2 Data mining objectives

A data mining algorithm is typically formed by three elements [74]:

- **A model**, which is the structure that has to be adjusted as the objective of the data mining, determined by the data being used.
- **A preference criterion** or cost function, which states the conditions for the algorithm to stop.
- **A search algorithm**, that defines the step to be followed to adjust the model, based on the preference criterion and the data.

The model is given by the data that has to be clustered, and some initial parameters like the number of clusters to be found. The cost function can be a linear combination of a number of variables, however most algorithms of quantitative data clustering are variations of a cost function that optimizes distance among centroids, applying Minkowski metrics (see Section 6.3.1). The search algorithms are iterative algorithms with stop conditions, like the number of iterations or a convergence margin or tolerance reached.

According to Mitra et al. [74], two are the main tasks performed by data mining algorithms: **descriptive** and **predictive**. The first deals with the description of the data, by means of patterns for example, and the other with obtaining models that establish input – output relations among variables and predict future outcomes given the initial conditions and values of inputs.

Beyond this division, a further classification can be established, according to Han and Kamber [73], based on the type of data to be mined (and the type of database), and the specific objectives of the data mining process, or the kind of knowledge being mined. Figure 5.1 depicts the classification of the data mining tasks according to their objectives, and the different types of databases that can be the objective of the analysis. A detailed description of the types of databases can be found in the book by Han and Kamber [73]. These databases are the following:

- Relational, which follows a structure based on tables, with attributes or variables, and objects stored as *tuples* or records, where each object is usually identified by a unique key.
- Data warehouses, repositories or collections of data from multiple sources, stored at one site.
- Transactional, where each record represents a transaction, with a unique ID.
- Object relational, with a structure based on object-oriented programming, where each new entity is considered an object, with methods and attributes.
- Temporal, sequence, and time-series databases, which store relational data that include time-related attributes, such as values obtained over repeated measurements of time, in the case of time-series.
- Spatial and spatiotemporal databases, which store spatial-related information, such as geographic data.
- Text and multimedia databases, with word descriptions for objects and other multimedia information.

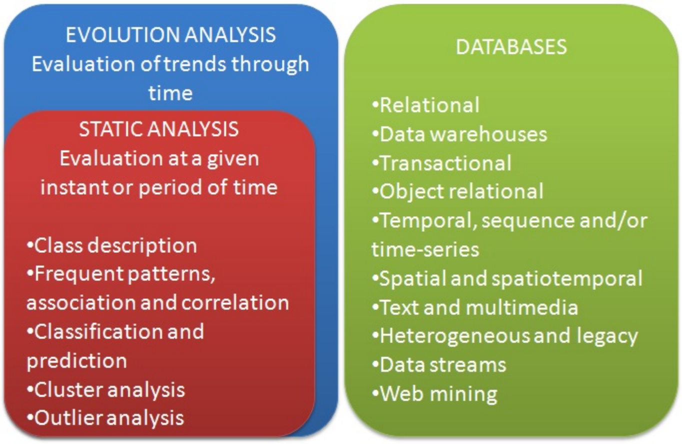

EVOLUTION ANALYSIS
Evaluation of trends through time
STATIC ANALYSIS
Evaluation at a given instant or period of time
•Class description
•Frequent patterns, association and correlation
•Classification and prediction
•Cluster analysis
•Outlier analysis
DATABASES
•Relational
•Data warehouses
•Transactional
•Object relational
•Temporal, sequence and/or time-series
•Spatial and spatiotemporal
•Text and multimedia
•Heterogeneous and legacy
•Data streams
•Web mining

Fig. 5.1. Classification of Data mining techniques according to the knowledge mined.

- Heterogeneous and legacy databases, formed by a set of interconnected, autonomous component databases that share information; a group of heterogenous databases that combines different kinds of data systems, forms a legacy database.
- Data streams, consisting in application where a (usually) huge information volume flows in and out of an observation window dynamically.
- Web mining, containing analysis of use, access and pattern searching among web pages and servers.

Regarding the objectives, the main are, according to Han and Kamber, the following [73]:

- Static analysis. Comprehends techniques aimed at the analysis of static data, or a set of data from a given time or time frame, or a set of data whose time stamp value is not being taken into account as an objective of the analysis. The possible objectives of this analysis are:
  - Descriptive:
    - Class description.
    - Frequent Pattern Mining.
    - Classification.
    - Cluster analysis.
    - Outlier analysis.
  - Predictive:
    - Association.
    - Correlation.
    - Prediction.

- Evolution analysis. In the evolution analysis, the trend of the series and the temporal evolution of the data is a key factor in the objective of the analysis. Most of the objectives described for the static analysis can be extended in the evolution analysis. However, the inclusion of a new dimension (time) as an objective of the analysis implies the modification of the techniques and algorithms developed for a static analysis.

Following, techniques and algorithms used for these objectives are briefly outlined.

5.2.1 Descriptive data summarization

Descriptive data summarization techniques or statistical analysis, although described by Han and Kamber [73] as being part of the processing of the data in the Data Warehousing step, are included in this list to be taken into account as an initial analysis that allows the expert to make hypothesis and select the subsequent data mining techniques to be applied on the data. This analysis can be used to "identify the typical properties of the data and highlight which data values should be treated as noise or outliers [73]". Examples of data summarization analysis include central tendency (mean, median, mode) and dispersion of the data (quartiles, interquartile range (IQR), variance, outliers, or boxplot graphs) [73].

5.2.2 Class description

The *concept description*, as described by Han and Kamber [73] generates "descriptions for the characterization and comparison of the data. It is sometimes called class description, when the concept to be described refers to a class of objects. Characterization provides a concise and succinct summarization of the given collection of data, while concept or class comparison (also known as discrimination) provides descriptions comparing two or more collections of data."

Data summarization is one of the techniques applied for class or concept description. It provides the user with comprehensive global descriptions of data sets, allowing to grasp the essence from a large amount of information in a database [74]. According to Raschia and Mouaddib [81], a *summary* z is a synthetic description of a subset of data from the database $\sigma_z(\Re)$. This subset is called the *extent* of the summary, whereas the summarized description z is called the *intent*, and describes similar features or attributes of *tuples* (data objects from the subset) . These descriptions can be either quantitative or qualitative. Global conclusions that describe the entire database are extracted and arranged in linguistic summaries of the type "most of the clients from city A buy product d". One of its applications is the extraction of knowledge in text mining [82].

5.2.3 Frequent pattern mining

Frequent patterns are patterns that appear in a data set frequently [73]. Han and Kamber describe different types of patterns:

> "A set of items that that appear frequently together in a transaction data set is a frequent itemset. A subsequence, or an ordered sequence of objects, such as the purchase of goods for example, if it occurs frequently in a shopping history database, is a (frequent) sequential pattern. A substructure can refer to different structural forms, such as subgraphs, subtrees, or sublattices, which may be combined with itemsets or subsequences. If a substructure occurs frequently, it is called a (frequent) structured pattern. [73]"

One typical example of frequent itemset mining is market basket analysis. This process analyzes customer buying habits by finding associations between the different items that customers place in their shopping baskets. Frequent itemset mining leads to the discovery of associations and correlations among items in large transactional or relational data sets [73]. These associations and correlations can be expressed in form of association rules. Association rules describe interesting and usually hidden relationships among attributes or variables, such as a dependency relation [83]. The efficiency of a rule is measured by two indicators: the *support* and the *confidence*. The support reflects the usefulness of the rules, or the the number of times the rule is produced, from the total set of rules. The confidence reflects the certainty of the rule, i.e., a percentage or probability of truth, that when the rule is given, it is given in the form it was declared.

In general, association rule mining can be viewed as a two-step process [73]:

1. Find all frequent itemsets: By definition, each of these itemsets will occur at least as frequently as a predetermined minimum support count.
2. Generate strong association rules from the frequent itemsets: By definition, these rules must satisfy minimum support and minimum confidence.

5.2.4 Classification

The classification process has as objective to classify objects within a range of values or categories. This classification is usually a two-step process: first, the classifier has to be modeled or trained, making use of a selection of the data whose classification is known, called the training data set. Once the classifier is trained, the second step is the classification process itself.

The classifier reliability is, therefore, based on the match between the training and the classification data sets. The training process is called *supervised* learning, because the classifier learns from pre-classified data, adjusting its parameters accordingly. This is the main difference with the cluster analysis, where the data are grouped according to similarity with no previous knowledge about their membership or class; this is known as *unsupervised* learning.

Examples of classification techniques are the Self Organizing Maps, such as the Kohonen networks [84], artificial neural networks that classify objects based on a reinforced learning algorithm [71]. Other examples are the algorithms that yield classification rules, applying neural networks and / or fuzzy sets, such as in [85]. Mutilayer artificial neural networks are also used as classifiers, having an output layer with as many neurons as different classes or categories [86]. Support Vector Machines (SVM) are also classifiers that separate the data making use of n-dimensional hyperplanes. The points nearest to the separating hyperplane are called the Support Vectors [87]. Decision trees induction are also a common technique for classification [88] [86]. A decision tree is, as described by Han and Kamber, "a flowchart-like tree structure,where each internal node (nonleaf node) denotes a test on an attribute, each branch represents an outcome of the test, and each leaf node (or terminal node) holds a class label [73]."

Bayesian networks are also used for classification. These are graphical representations of probability distributions, derived from co-occurrence counts in the set of data items. A Bayesian Belief Network or BBN [89] is a network graph formed by nodes and edges, where the nodes represent attribute variables and the edges represent probabilistic dependencies between the attribute variables. Associated with each node are conditional probability distributions that describe the relationships between the node and its parents . Naïve Bayesian classifiers, unlike BBN. assume that the effect of an attribute value on a given class is independent of the values of the other attributes. This assumption is called *class conditional independence*. It is made to simplify the computations involved and, in this sense, is considered "naïve" [73].

Classifiers can also be designed as a combination of classification algorithms; such is the case in the *bagging* method, for instance, which uses as output for the classification the average value from a set of pre-trained classifiers.

5.2.5 Cluster analysis

The *cluster analysis* is an unsupervised learning classification technique, whose aim is to obtain clusters of data, grouped by similarity, where similarity is computed as a mathematical function of distance between two objects [90]. Typically, the clustering is a classification process done on groups of objects from a database. The objects are entities that are formed by a number n of characteristics or dimensions, thus the objects are n - dimensional. The resulting clusters are therefore represented by a number c of centroids or pattern prototypes, usually one per cluster, which are also n - dimensional. Figure 5.2 displays an example of clustering in a set of $2D$ objects. The centroids usually coincide with the center of gravity of the cluster, however this may vary, according to the algorithms used and the data types being classified.

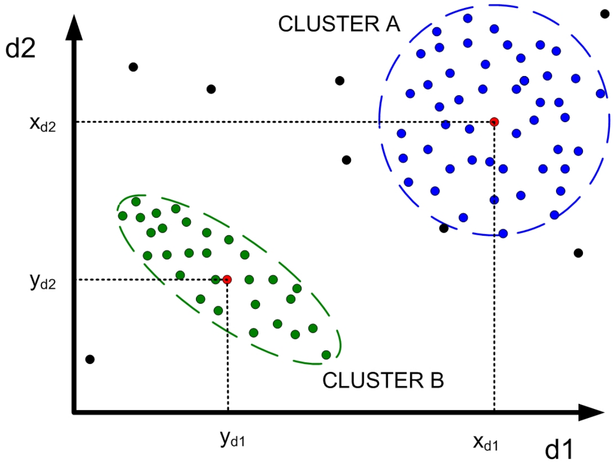


Fig. 5.2. Example of two clusters (A and B) and their respective centroids, (x_1, y_1) and (x_2, y_2), on a 2D data set.

Clustering algorithms and techniques are explained in detail in Chapter 6.

5.2.6 Outlier analysis

Besides a descriptive summarization of the data with a statistical analysis, the identification and classification of outliers, or data values out of the scope of the expected or typical values in the data set, can be a challenging task and it is one of the most common first steps in the data warehousing process, in order to remove outliers or noise from the data (noise is defined as the data with erroneous value, mostly due to an error in the data monitoring process). If these data are not removed, subsequent data analysis patterns or models can be greatly modified or altered.

The outlier mining problem can be viewed as two subproblems, according to Han and Kamber [73]:

1. “Define what data can be considered as inconsistent (or outlier) in a given data set.
2. Find an efficient method to mine the outliers so defined.”

Scatter plots are a proper visualization analysis to compare two dimensions or characteristics of the data objects and quickly identify linear or non-linear dependencies, and possible outliers or noise. However, as the dimensions or characteristics of the data objects increase, data mining techniques are needed to identify possible outliers in a data set. One example is the use of wavelets [91] to identify region outliers in multidimensional meteorological data [92]. Other approach makes use of density-based cluster analysis to obtain clusters of data and outliers [93].

As indicated by Han and Kamber [73], the methods for outlier detection can be categorized into four approaches, which are the statistical approach, the distance-based approach, the density-based local outlier approach, and the deviation-based approach:

- The statistical distribution-based approach assumes a distribution or probability model for the given data set (e.g. a normal or Poisson distribution) and then identifies outliers with respect to the model using a *discordancy* test.
- The distance-based outliers are those objects that do not have "enough" neighbors, where neighbors are defined based on distance from the given object.
- In the density-based local outlier approach, an object is a local outlier if it is outlying relative to its local neighborhood, particulary with respect to the *density* of the neighborhood. unlike previous methods, it does not consider being an outlier as a binary property. Instead, it assesses the degree to which an object is an outlier.
- The deviation-based approach identifies outliers by examining the main characteristics of objects in a group. Objects "that deviate" from this description are considered outliers.

5.2.7 Correlation

The correlation analysis provides a measure of how related two attributes are to each other. This fact, however, does not necessary imply that one attribute is modifying the behavior of the other. A further study is needed to determine which of the two variables is the antecedent and which is the consequent; or it may happen that there is no causality between the two variables, but both of them are correlated to other attributes.

For numerical variables, the most common measure of linear correlation is the correlation coefficient, also called *Pearson product moment coefficient* [73], shown in (5.6).

$$d(j,r) = \frac{\frac{1}{N}\sum_{i=1}^{N}(x_{ij} - \bar{x}_j)(x_{ir} - \bar{x}_r)}{s_j s_r} \tag{5.6}$$

Where $\bar{x}$ is the mean and s is the standard deviation of each dimension or each variable j and r (5.7).

$$s = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \bar{x})^2}{N}} \tag{5.7}$$

The value of the correlation coefficient varies between -1 and 1. The higher the value, the most correlated the two variables are. If the coefficient has a negative value, it indicates a negative correlation: as one variable increases in value, for instance, the other decreases at same rate. If the correlation coefficient has a value of 0, then the two variables are linearly independent and are not correlated.

For qualitative or categorical data, the chi-square (χ^2) test, also called *Pearson statistic* χ^2, is also a common correlation coefficient between to categorical series [73]. Its expression is shown in (5.8).

$$\chi^2 = \sum_{i=1}^{c} \sum_{j=1}^{r} \frac{(o_{ij} - e_{ij})^2}{e_i j} \tag{5.8}$$

Where the two attributes, x_a and x_b, can have different values, from x_{a1} to x_{ac}, and from x_{b1} to $x_b r$. The term o_{ij} is the *observed frequency*, i.e., the actual count of a given combination (x_{ai}, x_{bj}). The term e_{ij} is the *expected frequency*, or the maximum amount of combinations that can be given, computed as can be seen in (5.9).

$$e_{ij} = \frac{(count x_a = x_{ai}) x (count x_b = x_{bj})}{N} \tag{5.9}$$

The χ^2 statistic tests the hypothesis that the two variables are independent. If the hypothesis can be rejected, then it can be said that the two attributes are statistically related or associated. A correlation measure can be used to augment the support-confidence framework for association rules [73]. However, the Pearson coefficient quantifies the linear relationship, and therefore it is not a reliable measure whenever there is a non-linear relationship between two variables. Scatter plots can be used as a first approach to identify these situations and visualize the correlation (and the dispersion) of the data.

5.2.8 Prediction

Data prediction is a two step process, similar to that of data classification. In this case, the attribute for which values are being predicted is continuous-valued (ordered) rather than categorical (discrete-valued and unordered)[73].

The most common techniques in prediction are regression and Artificial Neural Networks (ANN) [94]. Regression models make use of chosen variables from the data set to predict future values of other variables or characteristics from the same data set [74], with the objective of being able to predict values from new inputs. In other words, “regression analysis can be used to model the relationship between one or more independent or predictor variables and a dependent or response variable (which is continuous-valued) [73]”.

Regression models can be linear or non-linear. The most common technique to obtain linear regression models is the linear least square method, which is described in Annex C of the present document. Regarding non-linear models, polynomial fitting is also a common technique for regression. ANNs are briefly described in the following Section.

Prediction models are the basis for forecasting systems. In the present document, a comprehensive review of techniques for forecasting values of energy demand and load profiles has been included in Chapter 10.

5.3 Techniques and algorithms for mining data

The main techniques used in data mining for the previously described objectives, according to Mitra et al. [74], are the following:

- Fuzzy sets.
- Artificial Neural networks (ANNs).
- Rough sets.
- Genetic algorithms.
- Case-Based reasoning.
- Decision trees.

These techniques are briefly described next.

5.3.1 Fuzzy sets

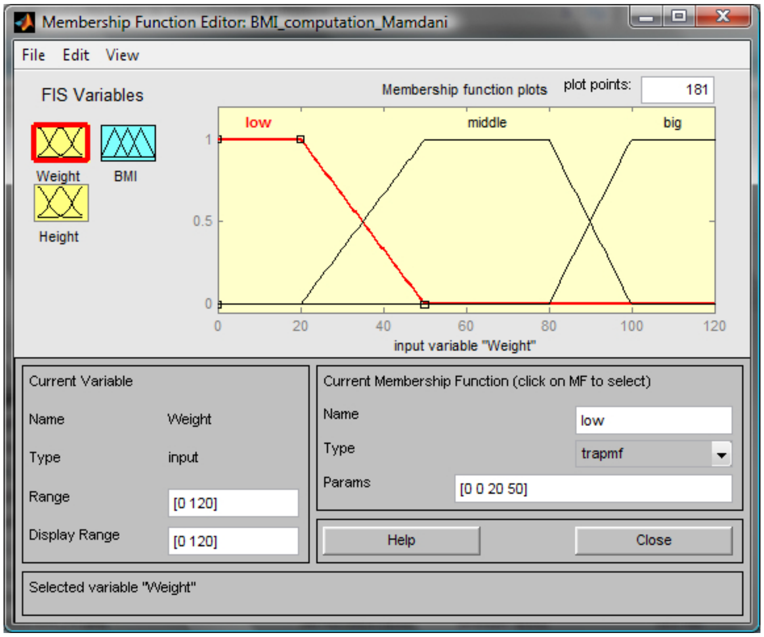
Fuzzy sets [95] allow modeling imprecise and qualitative knowledge, in the form of categorical labels that indicate states with an unclear bound, such as in human reasoning.

The two most important properties of fuzzy sets are: to allow modeling uncertainty and dealing with it in robust systems, and to capture and work with categorical labels of variables of different nature (qualitative, quantitative), with soft transitions among the different labels that gather all the possible scope of values for a variable.

Databases may contain data with dimensions or variables of different magnitude and/or nature, data with uncertainties or with a qualitative attribute (low, high, etc). Fuzzy sets can represent all this information, as groups of data with a value or degree of membership (not probability) attached, ranging between zero and one. Each group has a categorical or qualitative description. In words of Zadeh [95]:

> "A fuzzy set A in X is characterized by a membership (characteristic) function $f_A(x)$ which associates with each point in X a real number in the interval $[0, 1]$ with the value of $f_A(x)$ at x representing the grade of membership of x in A. Thus, the nearer the value of $f_A(x)$ to unity, the higher the grade of membership of x in A."

This is the *fuzzification* process. An example can be seen in Fig. 5.3 with the fuzzification of the variable "weight" in three trapezoidal membership functions: "low", "middle" and "big".



plot points: 181

Current Variable

Name Weight

Type input

Range [0 120]

Display Range [0 120]

Current Membership Function (click on MF to select)

Name low

Type trapmf

Params [0 0 20 50]

Selected variable "Weight"

Fig. 5.3. Example of fuzzification process: three membership functions are defined for the variable "weight".

From all the fuzzified input variables, a *fuzzy inference* system is defined, in order to yield fuzzy output variables as a function of the fuzzy input values and the inference rules or models. These rules are mainly *IF–THEN* or antecedent – consequent rules, and AND or OR conditions or operators [96]. The mathematical interpretation of the AND and OR logical operators may vary. For instance, an OR operator can be interpreted as a sum of two values, or just the maximum of both. The use of *hedges* (very, many, too, a few, etc) can also modify the shape of the membership functions (i.e. the slope) and therefore have an impact on the inference result [97][98][99] as can be seen in Fig. 5.4.

There are mainly two approaches for fuzzy inference: the **Mamdani** type [100] and the **Takagi-Sugeno** type [101]. The Mamdani type defines the output variables as fuzzy sets, whose fuzzy value is a function of the fuzzy inference or, more specifically, a combination of the active or *fired* rules in the inference system, as a function of the input values.

An example can be seen in Fig. 5.5, where three fuzzy membership functions for the output variable Body Mass Index (BMI) have been defined.

Then, some very simple inference rules, relating inputs (weight and height) with the output (BMI) have been defined (Fig. 5.6).

The *defuzzification* process is the final conversion of the result from a fuzzy value to a final datum or label. The most used methods are the center of

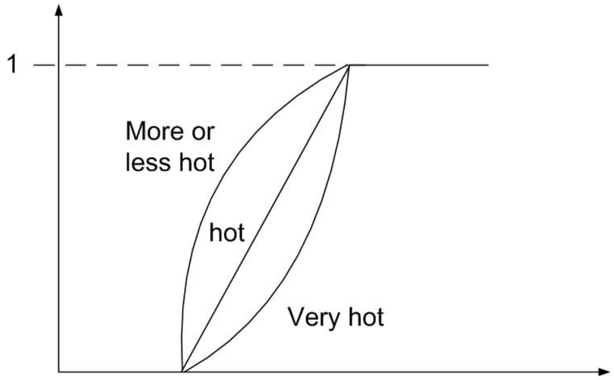
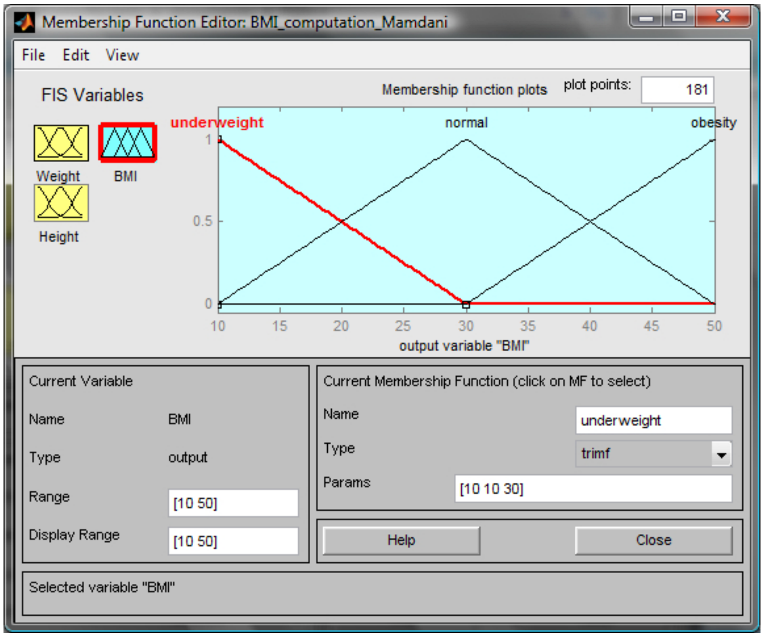


Fig. 5.4. Example of application of hedges "very" and "more or less" on membership function "hot".



plot points: 181

Current Variable

Name BMI

Type output

Range [10 50]

Display Range [10 50]

Current Membership Function (click on MF to select)

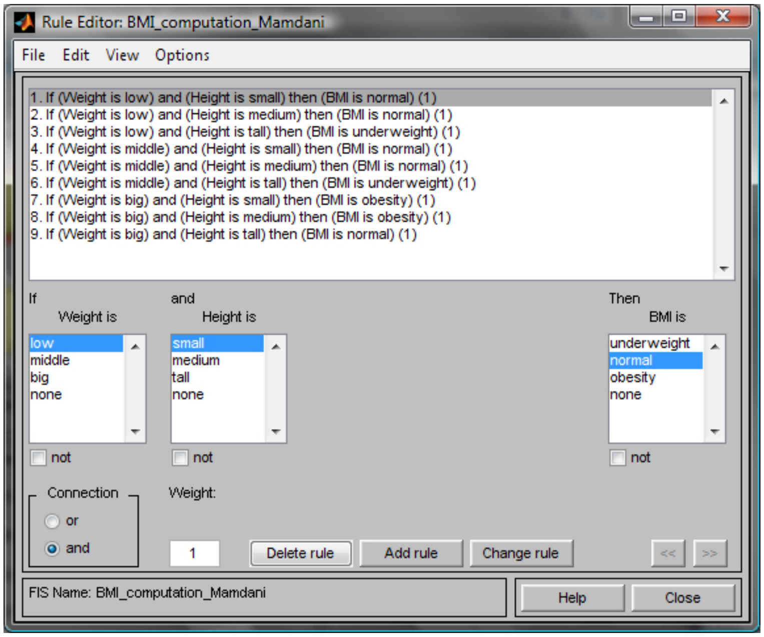
Name underweight

Type trimf

Params [10 10 30]

Selected variable "BMI"

Fig. 5.5. Example of Mamdani inference: three membership functions are defined for the output variable "BMI".



1. If (Weight is low) and (Height is small) then (BMI is normal) (1)
2. If (Weight is low) and (Height is medium) then (BMI is normal) (1)
3. If (Weight is low) and (Height is tall) then (BMI is underweight) (1)
4. If (Weight is middle) and (Height is small) then (BMI is normal) (1)
5. If (Weight is middle) and (Height is medium) then (BMI is normal) (1)
6. If (Weight is middle) and (Height is tall) then (BMI is underweight) (1)
7. If (Weight is big) and (Height is small) then (BMI is obesity) (1)
8. If (Weight is big) and (Height is medium) then (BMI is obesity) (1)
9. If (Weight is big) and (Height is tall) then (BMI is normal) (1)

Weight is	Height is	BMI is
low	small	underweight
middle	medium	normal
big	tall	obesity
none	none	none

Weight: 1

FIS Name: BMI_computation_Mamdani

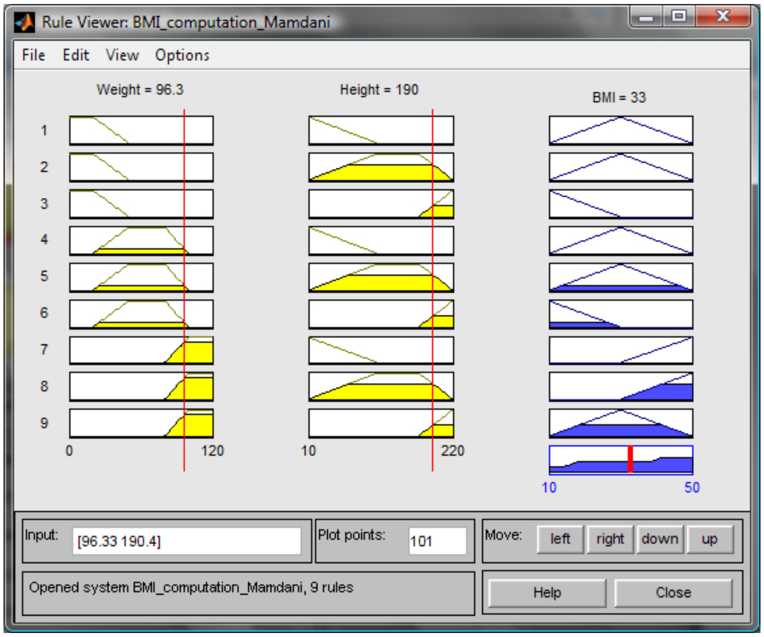
Fig. 5.6. Example of Mamdani inference: description of inference rules.

gravity or a weighted sum of function outputs. Figure 5.7 depicts an example of Mamdani defuzzification process applying the center of gravity method: the resulting numerical output value is the one that results from adding the activated or *fired* output fuzzy sets from the inference system, and computing the middle point value of the resulting area (i.e. the center of gravity).

In the Takagi-Sugeno inference the membership functions of the output variables are linear functions or constant values, instead of linguistic variables. The numerical output or defuzzification process is the weighted sum of all the resulting values from the inference system. An example can be seen in Fig. 5.8, where the output variable BMI is defined by three membership functions which are *singleton* functions or constant values.

According to Mitra et al. [74], the main applications of fuzzy sets in data mining are the following:

- *Clustering*. Fuzzy clustering algorithms, such as the ones described by Gustafson and Kessel [102]; Gath and Geva [103]; Pedrycz and Waletzky [104]; Tamas et al. [105]; Oliveira and Pedrycz [90] and Dunn [106] are widely used in clustering analysis.
- *Association rules*. Fuzzy association rules allow the affinity of rule-based inference with human reasoning and knowledge representation. Developments of mining fuzzy association rules from databases are found, for instance, in [107].

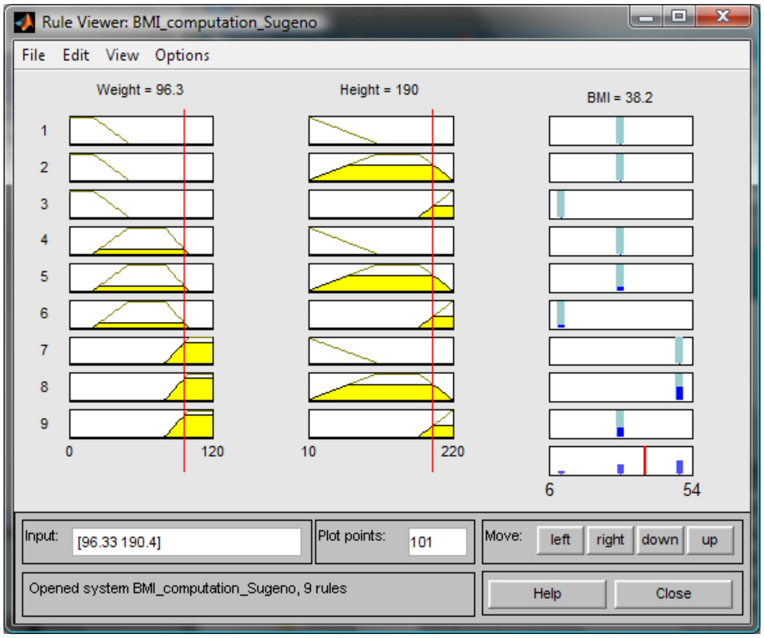


Input: [96.33 190.4]

Plot points: 101

Opened system BMI_computation_Mamdani, 9 rules

Fig. 5.7. Example of Mamdani inference: defuzzification process.



Input: [96.33 190.4]

Plot points: 101

Opened system BMI_computation_Sugeno, 9 rules

Fig. 5.8. Example of Sugeno inference.

- *Functional dependencies.* A functional dependency [108] defines a constraint between two sets of attributes from a relational database [109]. Given two attributes X and Y, attribute Y is functionally dependent of attribute X (denoted as $X \rightarrow Y$) if and only if both attributes are a subset of a relational scheme r ($X, Y \subseteq r$) and for each value of attribute X at a tuple t ($X(t_1)$), there is exactly one value of attribute Y ($Y(t_1)$). For instance, attribute *Age* in a database is functionally dependent of attribute *Name*, since each person (or name) listed has a unique age associated (the contrary does not apply, more than one person can have the same age). Equation (5.10) describes the conditions for a functional dependency, stating that if two tuples have the same value for attribute X, then the value for attribute Y has also to be the same at the two tuples.

$$\forall t_1, t_1 \in r, If X(t_1) = X(t_2), then Y(t_1) = Y(t_2) \tag{5.10}$$

  Fuzzy logic can be applied to design inference systems based on the use of functional dependencies as linguistic variables. The resulting fuzzy functional dependencies allow to detect smooth relationships between attributes with respect to a resemblance criterion [108][81].
- *Data summarization.* Fuzzy sets give data summarization the possibility to either assign membership of clusters of data to different linguistic categories or values of an attribute, or to build summaries with fuzzy rules. For instance, Raschia and Mouaddib apply fuzzy sets to deal with imprecise and vague information to build hierarchical summary trees [81].
- *Web application.* Fuzzy logic is applied to the objective of mining user profiles and *URL* associations, by means of defining rules.
- *Image retrieval.* Fuzzy memebership functions are applied in the area called Content-Based Image Retrieval or CBIR. These systems retrieve images by similarity measures concerning visual information such as colors, texture, shape, main shapes, etc.

Most of the fuzzy sets developed and implemented in different applications are of one dimension, called Type-1 fuzzy sets. Zadeh, however, defined the concept of *linguistic variable* with the possibility to have n dimensions or fuzzy sets [110]. In this sense, Type-2 fuzzy clustering algorithms have been developed and can be found in the literature [111], with a fuzzy membership grade in two dimensions (with the definition of primary and secondary membership functions).

5.3.2 Neural networks

Artificial neural networks or ANN are inspired in a model of parallel computation of the human brain [112][79]. A biological neuron performs communication with other neurons by *synapsis* (see Fig. 5.9): electrochemical signals are produced and dispersed by a neuron to its neighbors. Each neuron receives a number of signals in the terminations called *dendrites*, and produce an output accordingly, which is a function of the inputs. This output is sent forward through the *axon*.

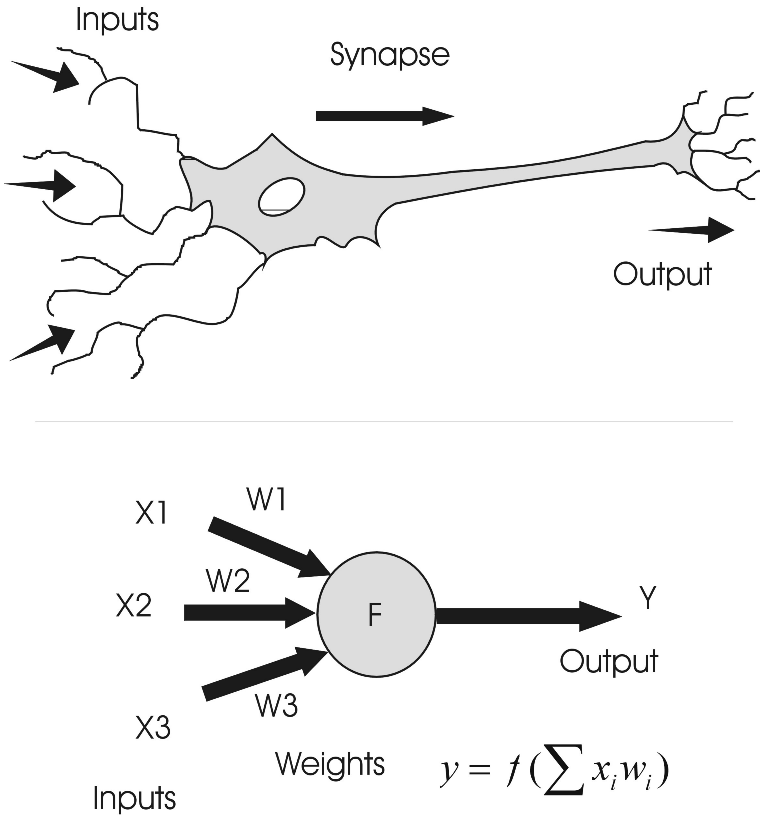


Fig. 5.9. Mathematical representation of a biological neuron and synapse process.

An ANN mimics this behavior: each neuron receives a number of inputs and processes this information through a specific function. The input to this activation function is the weighted sum of all the inputs. The output can be a linear combination, such as the weighted sum of the inputs, or other functions, such as gaussian functions, or the so called *activation* functions. These are functions that allow a smooth change of the output between two states (0 and 1). The most used of the activation functions is the sigmoid function (5.11), whose behavior can be observed in Fig. 5.10.

$$y = \frac{1}{1 + e^{-x}} \tag{5.11}$$

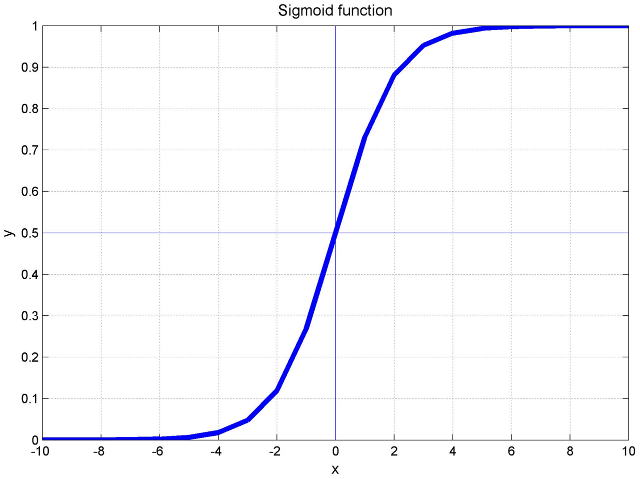


Fig. 5.10. Sigmoid function.

The first mathematical models of a biological neuron were developed in the decade of 1940 [113] by different authors, such as McCullock and Pitts [114] and Hebb [115]. Initially, the first models focused on a single neuron, being the **Perceptron** [116] the most used, but rapidly the advantages of networks of neurons with different arrangements were seen and presented in subsequent works [117] [118] [119].

The union of neurons in different structures provides the resulting ANNs with capabilities to approximate and represent complex and non-linear behaviors. ANNs are widely used as non-linear regression models [120], or as classifiers [121], or to extract behavioral patterns from multivariable data [84]. Figure 5.11 depicts an example of an ANN used as a prediction model, to estimate photovoltaic generation as a function of weather conditions and irradiance [122].

One of the most used configurations of ANN is the **multilayer** network (see Fig. 5.12). The ANN is arranged in layers, having one input layer, a variable number of hidden layers, and one output layer. This configuration of ANN is used mainly in prediction and classification models. Hidden layers are formed by activation functions, and output layers by linear functions, such as the sum of weighted inputs.

These ANNs are supervised learning algorithms, i.e., they need a training data set in order to adjust the weights and biases of the neurons, to obtain the expected values at the outputs. The most common technique for the supervised learning is the *backpropagation* of the error [123]: values at the output layer (o_i) are compared with the expected p outputs of the training dataset (t_i), and the resulting errors are propagated backwards to the precedent layers and used to adjust the weights and biases of the neurons (w_{ij}). The procedure

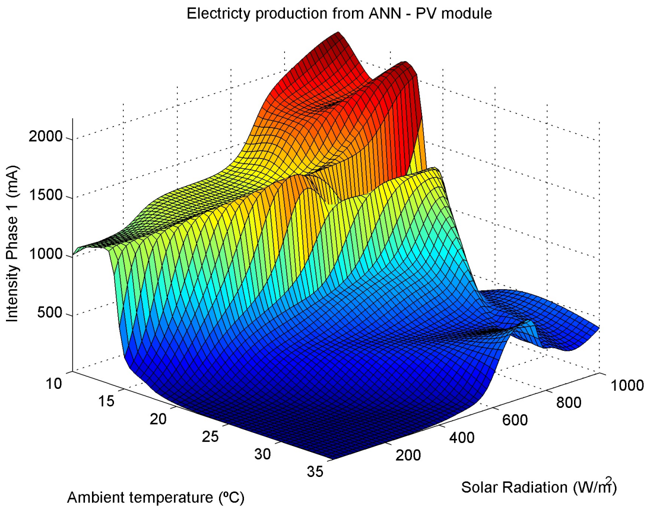


Fig. 5.11. Example of prediction from multilayer ANN.

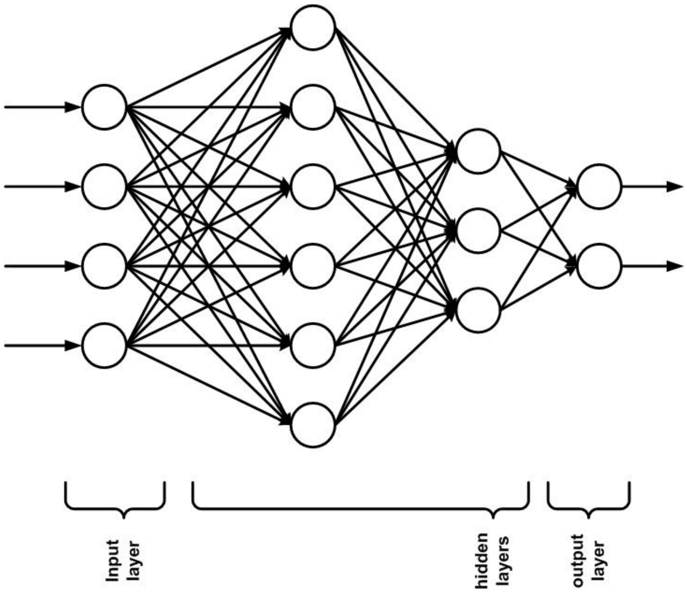


Fig. 5.12. Multilayer ANN.

for this is a learning rule or algorithm obtained as the result of the minimization of the objective function E, which is the squared sum of errors, as seen in (5.12) and (5.13). The error at each iteration is the difference between the obtained output o_i and the expected value t_i from a training dataset, for all the p neurons at the output layer. This algorithm includes a parameter (γ), which is a learning rate, usually a constant value, used to define the step (i.e. the speed) in the search for the local minima.

$$E = \frac{1}{2} \sum_{i=1}^{p} \|o_i - t_i\|^2 \tag{5.12}$$

$$\Delta w_{ij} = -\gamma \frac{\partial E}{\partial w_{ij}} \tag{5.13}$$

Other learning techniques can be applied. For instance, Huang et al. [124] suggest extreme learning machine (ELM) as the best option to train single-hidden layer feedforward neural networks. The ELM randomly chooses hidden nodes and analytically determines the output weights.

Self-Organizing Maps or SOM are other kind of ANNs, used mainly for classification, data reduction and visualization, and pattern recognition purposes [84]. The versatility of SOM relies mainly on an easier interpretation of the classification, based on a 2D grid projection of multidimensional objects. The SOM training algorithm makes use of unsupervised competitive learning, a type of learning procedure where the input object is compared with the weights or codebook of all neurons, choosing the one with the highest similarity to the object. This “winning” neuron is called the Best Matching Unit or BMU. The similarity is usually computed as the minimum Euclidean distance, although other configurations and similarity measures might be used.

A learning rule is then applied, which updates the weights of the BMU, and also the weights of its closest neighbors, following a distribution function (usually Gaussian) which decreases as the distance between the neuron and the BMU gets higher. Examples of a SOM used to classify 24 hour load profiles of energy consumption are shown in Figures 5.13 and 5.14.

Other configurations of ANNs have been described in the literature, such as RBF (Radial Basis Function) networks [125], Hopfield networks [119], and recurrent ANNs [126].

Because of their *black-box* nature, neural networks were initially discarded for data mining [74]. However, their potential to model symbolic rules has spread their usage. According to Mitra et al. [74], neural networks are used in data mining for the following objectives:

- Rule extraction. Many algorithms of rule extraction start with a trained network, from where rules are extracted based on the values at specific nodes or neurons from the output or hidden layers. Evaluation indices are then needed to compute the performance of the rules, which can be removed or modified as stated by a pruning algorithm [127][128].

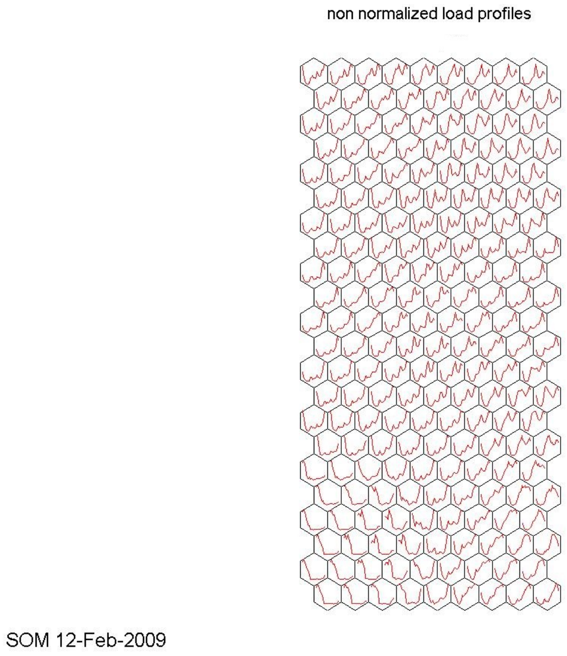


Fig. 5.13. Spatial assignment of energy consumption load profiles in SOM neurons based on similarity.

- Clustering and Classification. Self-organizing or Kohonen maps [84], also called SOM , are used with different objectives: exploratory data analysis, data visualization, dimensionality reduction, pattern recognition and classification [129] [130] [79].
- Regression. Neural networks are used as regression models, to predict future values given new inputs, based on a training dataset [131] [132] [126].

5.3.3 Rough sets

Rough sets, first described by Zdislaw Pawlak in 1991 [133], are closely related to *granular computing* theory, in the sense that "information is allowed to be imperfect; i.e., it may be imprecise, uncertain, incomplete, conflicting, or partially true [134]."

Rough set theory associates some information to an object, regarding its nature and the description of the object according to the class it belongs to and the knowledge about the object that might be available. A rough set is formed by all the objects which share similar information. However, this information cannot be precise in all the cases, therefore the vague or imprecise regions of information are defined by two *perception* regions: the *lower* and

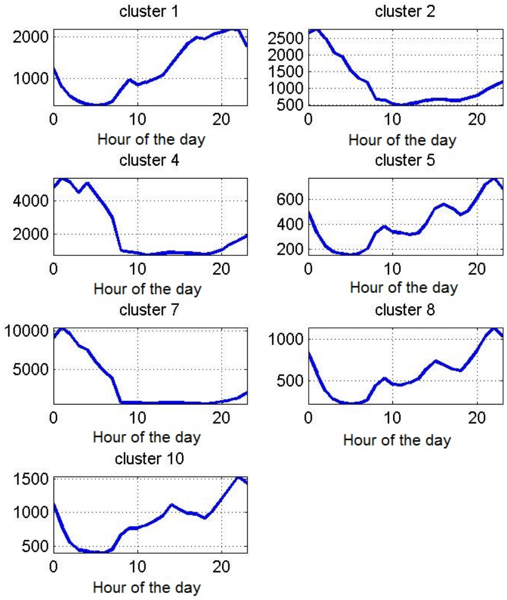


Fig. 5.14. Resulting prototypes after applying the K-means clustering algorithm on the SOM neuron codebooks.

the *upper* approximations. The lower approximation consists of all objects which *surely* belong to the concept and the upper approximation contains all objects which *possibly* belong to the perception [134].

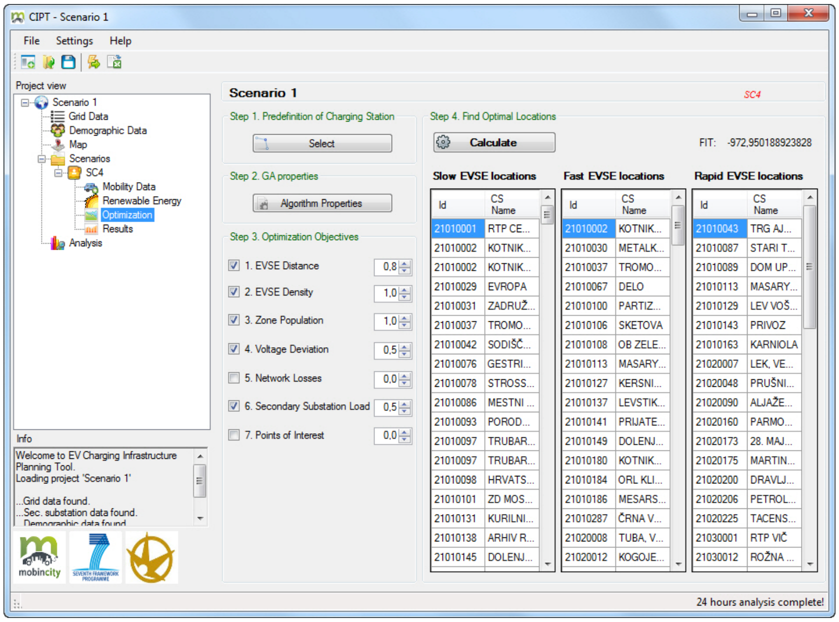
According to Mitra et al. [74], the main applications of rough sets to data mining are the following:

- Rule induction. Rogh sets are used to induce association rules which can be "true", if only objects from the lower approximation of the set are used, or can be "possible", if objects from the upper approximation are used [133][135].
- Data filtration by template generation. This mainly involves extracting elementary blocks from data based on an equivalence relation [136].

5.3.4 Genetic and evolutionary algorithms

Genetic algorithms or GA [137] search for global optimization of a *fitness* or objective function by imitating the biological evolution following a process of natural selection: the solutions with the best fitness value (the best *genes*) are chosen and they *reproduce* by combination of their characteristics to produce a new generation, which may improve the results of the fitness function. Random *mutations* and crossover are introduced in the algorithm to enrich the possibilities of finding a global optimal solution and to avoid a rapid decaying of the search algorithm by focusing on a specific region of the search space.

GA are suited for multi-objective optimization problems, such as optimal route finding or allocation of resources. For instance, Fig. 5.15 displays a software for the optimal allocation of chargers for electric vehicles, taking into consideration diverse factors such as population, mobility areas and congestion of the power grid [138].



Welcome to EV Charging Infrastructure Planning Tool.
Loading project 'Scenario 1'

...Grid data found.
...Sec. substation data found.
...Demographic data found.

Step 1. Predefinition of Charging Station

Step 2. GA properties

Step 3. Optimization Objectives

☑ 1. EVSE Distance 0,8
☑ 2. EVSE Density 1,0
☑ 3. Zone Population 1,0
☑ 4. Voltage Deviation 0,5
☐ 5. Network Losses 0,0
☑ 6. Secondary Substation Load 0,5
☐ 7. Points of Interest 0,0

Step 4. Find Optimal Locations

FIT: -972,950188923828

Slow EVSE locations

Id	CS Name
21010001	RTP CE...
21010002	KOTNIK...
21010002	KOTNIK...
21010029	EVROPA
21010031	ZADRUŽ...
21010037	TROMO...
21010042	SODIŠČ...
21010076	GESTRI...
21010078	STROSS...
21010086	MESTNI ...
21010093	POROD...
21010097	TRUBAR...
21010097	TRUBAR...
21010098	HRVATS...
21010101	ZD MOS...
21010131	KURILNI...
21010138	ARHIV R...
21010145	DOLENJ...

Fast EVSE locations

Id	CS Name
21010002	KOTNIK...
21010030	METALK...
21010037	TROMO...
21010067	DELO
21010100	PARTIZ...
21010106	SKETOVA
21010108	OB ZELE...
21010113	MASARY...
21010127	KERSNI...
21010137	LEVSTIK...
21010141	PRIJATE...
21010149	DOLENJ...
21010180	KOTNIK...
21010184	ORL KLI...
21010186	MESARS...
21010287	ČRNA V...
21020008	TUBA, V...
21020012	KOGOJE...

Rapid EVSE locations

Id	CS Name
21010043	TRG AJ...
21010087	STARI T...
21010089	DOM UP...
21010113	MASARY...
21010129	LEV VOŠ...
21010143	PRIVOZ
21010163	KARNIOLA
21020007	LEK, VE...
21020048	PRUŠNI...
21020090	ALJAŽE...
21020160	PARMO...
21020173	28. MAJ...
21020175	MARTIN...
21020200	DRAVLJ...
21020206	PETROL...
21020225	TACENS...
21030001	RTP VIČ
21030012	ROŽNA ...

24 hours analysis complete!

Fig. 5.15. CIPT (Charging Infrastructure Planning Tool). FP7 Project MOBINCITY (Smart Mobility in Smart City), grant agreement No. 314328.

According to Mitra et al. [74], the main applications of GA in data mining are the following:

- Regression. GA can be used to model MIMO systems, based on the relations among all the variables assigned to the search space [139][140].
- Association rules. GA are used to identify and improve association rules by finding the range of values that optimizes rules' *accuracy* and *coverage*. The accuracy of a rule measures its degree of confidence, whereas its coverage is interpreted as the comprehensive inclusion of all the records that satisfy the rule [141][142].

5.3.5 Case-based reasoning

Case-based reasoning or CBR is an Artificial Intelligence (AI) method based on gathering knowledge from all the cumulated previous experiences. CBR "is able to utilize the specific knowledge of previously problem situations or cases. New problems are solved by finding a similar past case, and reusing it in the new problem situation [143]". The new experience is retained each time a problem has been solved, making it immediately available for future problems. A general CBR cycle may be described by the sequence of the following four processes [143]:

1. Retrieve the most similar case or cases.
2. Reuse the information and knowledge in that case to solve the problem.
3. Revise the proposed solution.
4. Retain the parts of this experience likely to be useful for future problem solving.

A new problem is solved by retrieving one or more previously experienced cases, reusing the case in one way or another, revising the solution based on reusing a previous case, and retaining the new experience by incorporating it into the existing knowledge-base (case-base).

CBR techniques in data mining are used to reinforce the outcomes of other methods such as statistical analysis, for applications such as the induction of association rules, for instance [144][145].

5.3.6 Decision trees

Decision trees are search structures formed by hierarchical nodes and branches. Each time a node is reached, a decision is made regarding the creation of new branches, where the data are mapped, following some specific criteria. The tree is hierarchically arranged in abstraction levels, following a *top-down*, or a *bottom-up*, structure [146].

Common applications of decision trees for data mining are clustering [147] and classification [86][88].

5.3.7 Other methods

Graph-based theory

Graphs are used in data mining to unveil hidden *topological* relations between data objects. A graph is defined as an object G formed by a number of vertices (V), a group of segments connecting pairs of vertices (E), and a mapping function f for the data objects [148].

Bayes networks

Also called *Belief Networks*, the Bayes networks are graphical representations of *conditional* probabilistic distributions and dependencies between objects' dimensions. The nodes in the networks represent objects dimensions or attributes, and the links between nodes represent the dependencies, conditioned by probabilistic functions [149][150][151].

Wavelets

Wavelets, a name derived from the term *small wave*, are functions that separate and analyze time series data in different components, similar to a Fourier decomposition, but in different levels or scales of granularity, based on the data frequency domain components [152][91]. Wavelets are mainly used for the filtering and selection of data and dimensionality reduction with the objectives of regression, clustering [153], classification and data visualization.

5.4 Introduction to dynamic data mining

Clustering and other data mining techniques are usually performed on static data, bounded within a specific range of time when the data were acquired, and therefore valid for that specific range. A sample time consistency is expected, meaning that all the data being analyzed should share a predefined value of acquisition time.

Dynamic data mining techniques deal with the analysis of dynamic, time-dependent changing data, such as time series or data streams. These data can be differentiated, according to its nature, purpose and mining objectives, in three types [73]:

- **Data Streams**. This definition typically refers to massive amounts of multidimensional data, being continuously obtained from any source, in chronological order, each measure with its time stamp. Examples of data streams might be data from communication networks, data of measured values of distributed energy generation, or data from meteorological stations. Data streams have the following characteristics: they are massive, temporarily ordered, fast changing and potentially infinite. Data mining techniques for data streams require an online processing, due to the huge amount of data and the complexity of storing it all and processing it off line [154].
- **Time-series database**. These databases comprise sequences of data with time stamp, which have been obtained over repeated measurements of time. A typical example of time-series data are stock market prices. A number of developments in the literature can be found, which apply data mining techniques on this type of data, in order to discover patterns and build prediction models.

- **Sequence database**. A sequence database is obtained from sequence measurements, which can be related to time or to other variables. *Sequence pattern mining techniques* have as their objective to find repeated patterns among these data. A typical example of sequence pattern mining is the market basket analysis, with the objective to extract association rules from sequential itemsets purchased by customers in supermarkets or shopping malls [155][156].

The techniques developed for mining temporal data mainly include, according to Weber [90], ANNs, decision trees, association rules and clustering. Following, methods and techniques for mining the three types of data are briefly described.

5.4.1 Data stream mining

As previously mentioned, data streams are characterized by being "temporally ordered, fast changing, massive, and potentially infinite [73]". Efforts in research in the area of data stream mining are concentrated on fast, online algorithms able to perform data mining on continuous data streams on a single scan, by applying one, or a combination, of the methods which will be described next. Due to the high amount of data, usually a faster, approximate answer is preferred, with an estimated error associated, rather than exhaustive, time-consuming analyses.

Data stream mining needs powerful systems that perform end-user queries, and store and visualize the results. Such systems are called DSMS or *Data Stream Management Systems*, and comprise three elements: end user interface, query processing and scratch space, to store temporal results and visualize them.

Methods for handling data streams

Synopses are methods to summarize data, or a subset of information from scanned chunks of data [73]. The term *synopsis data structure* applies to any summarization that, as a result, provides a smaller structure than the base data set. Some of these synopses methods, as described by Han and Kamber [73], are the following:

Random sampling. A random sampling method is applied to choose an unbiased population of the online data stream. In most cases, approximation techniques are applied to choose the population, since its size and characteristics are unknown. Different techniques can be applied, like *reservoir sampling* [73], where a set of candidates is maintained, being its population replaced with new candidates with similar characteristics.

Sliding window. The selected data to analyze is given by a fixed length of n-data samples, typically the most recent ones. This way, the resulting models are continuously updated with the most recent information. The sliding window can be complemented with a sample-variable weighting factor that gives the highest priority to the most recent samples within the window length, and the lowest weight or priority to the oldest values, thus varying the importance of the data being analyzed in the sliding window.

Histograms. A histogram partitions the data into a set of contiguous *buckets*, approximating the frequency distribution of element values in a data stream. However, as histograms give a view of the variance in data stream values, there are some issues to be considered to use this information as analysis, such as how to choose the length or range of the buckets, and how to update the computation of the histograms (which period to apply, which temporal data size to use).

Tilted time frame. Other approach to select the data to be analyzed from a data stream is to gradually modify the granularity as the time instants evolve: recent data is gathered at the finest granularity, and older data at coarser ones. This approach has the name of *tilted time frame.* Three examples of tilted time frames [73] are the following:

- Natural tilted time frame. the time frame is structured in multiple granularities based on the "natural" or usual time scale: the most recent 4 quarters (15 minutes), followed by the last 24 hours, then 31 days, and then 12 months.
- Logarithmic tilted time frame. The time frame is structured in multiple granularities according to a logarithmic scale.
- Progressive tilted time frame. The time frame is divided in different granularity levels, different than the *natural* time frame.

Critical layers. In cases when the computational cost of analyzing data cubes are too high, one approach that can be applied is to divide the data stream cube in smaller cuboids, either in time samples or time frame, or in the data dimensions, assigning different priorities. The data to be analyzed is the n partitioned in layers according to the priority to be analyzed: the first, critical layer, gathers the data that is strictly needed to obtain a result of the analysis. Following layers gather the rest of the data.

Stream data mining techniques

The techniques for stream data mining are classified, according to Han and Kamber [73], in three main categories, as a function of the mining objective: **frequent-pattern mining**, **classification** and **clustering**. The three of them are briefly commented below:

- **Frequent-pattern mining** finds a set of patterns that occur frequently in a data set, where a pattern can be a set of items (called an itemset), a subsequence, or a substructure [73]. In data streams, the scan and analysis of the selected data frame will probably be performed only once. The algorithms developed for frequent pattern mining in data streams yield an *approximate* set of patterns at each time step. One of these algorithms is the **Lossy Counting** algorithm. This algorithm, defined by Manku and Motwani [157] works by dividing the incoming data stream into sequential buckets of width w. The items in each new bucket are identified and a frequency list of all the items is updated. For each item, the list maintains f, the approximate frequency count, and Δ, the maximum possible error of f. Whenever the stream length N reaches the bucket boundary ($N = nw$, being n an integer), the frequency and error of all the items is reviewed and updated, in a way that the resulting frequency count stored for each item will either be the true frequency of the item or an underestimate of it. Let b be the current bucket number. An item entry is deleted if, for that entry, $f + \Delta \leq b$. This way, the frequency list is prevented from growing excessively.

- **Classification** models for data streams have to face the challenge of the *concept drift* [73], i.e., the variation in the target used as data reference to build the classifier, as new data comes in. Some classification algorithms developed to deal with data streams are the following:
  - The **Hoeffding Tree Algorithm** [158] makes use of the Hoeffding bound (or additive Chernoff bound) to decide at each node of a decision tree the smallest number, N, of examples when selecting a splitting attribute. This attribute would be the same as that chosen using infinite examples. Consider a real-valued random variable r whose range is R. Suppose that n independent observations of r have been made, and their mean $\bar{r}$ is computed. The Hoeffding bound states that, with probability $1 - \delta$, the true mean of the variable is at least $r - \epsilon$, where $\epsilon = \sqrt{\frac{R^2 ln(1/\delta)}{2n}}$. The Hoeffding bound has the property that it is independent of the probability distribution generating the observations. The price of this generality is that the bound is more conservative than distribution-dependent ones (i.e., it will take more observations to reach the same δ and ϵ). As more data comes in, the tree will continue to grow incrementally. One drawback of the Hoeffding tree, however, is that it cannot handle the concept drift: once a node is created, it does not change
  - The **Very Fast Decision Tree (VFDT)** [158] and **Concept - adapting Very Fast Decision Tree (CVFDT)** [159] are improvements of the Hoeffding tree in dealing with very fast data streams, with a varying structure in the case of the CVFDT, to cope with the concept drift. Some of the improvements implemented are: deactivating the least promising leaves whenever memory is running low, dropping poor splitting attributes, and improving the initialization method. CVFDT works by constructing alternate subtrees at the nodes, with the new best splitting attribute at the root. As new examples stream in, the alternate subtree will continue to develop, without yet being used for

  classification. Once the alternate subtree becomes more accurate than the existing one, the old subtree is replaced.
  - A **classifier ensemble** is a different approach to be used. Instead of applying a single classifier, a group of them is defined (CVFDT and naïve Bayes classifier, for example); at each iteration, the decisions are then based on the weighted votes of the classifiers.

- **Clustering** data streams implies to group the incoming data in clusters in one single pass. Moreover, the forming clusters may evolve and change at each iteration. Different techniques have been developed to manage effectively the clustering of data streams. Some of them, as described by Han and Kamber [73], are the following:
  - Compute and store summaries of past data, in order to analyze statistics in the resulting clusters.
  - Divide data streams into chunks based on order of arrival, compute summaries for these chunks, and then merge the summaries.
  - Perform incremental clustering of incoming data streams.
  - Perform both microclustering and macroclustering analyses. Apply a first clustering analysis, and then perform a second clustering step to group and fuse the previously obtained clusters.

  Examples of clustering algorithms for data streams are the following:
  - **STREAM**. The STREAM algorithm is a one-pass K-means algorithm for data streams [160] [161]. The core idea is to break the stream into chunks, each of which is of manageable size and fits into main memory. the STREAM algorithm processes data streams in buckets (or batches) of m points. The data in each bucket is clustered and then summarized in one single point or *centroid* per cluster, discarding the rest of the data. Each centroid is, however, weighted according to the number of elements belonging to the cluster. Subsequent chunks are clustered in the same way, until when the number of clusters exceeds m. Then, a second level of clustering is applied to the set of weighted centroids. A major limitation of the STREAM algorithm is that it is not particularly sensitive to evolution in the underlying data stream.
  - **CluStream**. The CluStream algorithm [162] divides the clustering process into "on-line and off-line components. The on-line component computes and stores summary statistics about the data stream using microclusters, and performs incremental on-line computation and maintenance of the microclusters. The off-line component does macroclustering and answers various user questions using the stored summary statistics, which are based on the tilted time frame model [73]."

5.4.2 Time series data mining

The three main analyses defined as objectives for the mining of time series data are: trend analysis, similarity search and clustering. The dynamic clustering of time series data applied to load profiles of electric energy consumption is the core objective of this thesis; therefore, its description in objectives, methods and techniques is detailed in the following Chapters. The trend analysis and similarity search are briefly commented next.

Trend analysis

The trend analysis can focus, according to Han and Kamber [73], in four different approaches or perspectives:

- Trend or long-term movements: These indicate the general direction in which a time series graph is moving over a long interval of time.
- Cyclic movements or cyclic variations: These refer to the cycles, that is, the long-term oscillations about a trend line or curve, which may or may not be periodic.
- Seasonal movements or seasonal variations.
- Irregular or random variations.

The techniques mostly applied to perform trend analysis are the development of prediction models, either by linear regression, or by neural networks. However, pure regression analysis cannot capture all of the four movements described. Complementary techniques can be applied, such as the computation of a *seasonal index*, or the implementation of a moving average [73]. The concept of seasonal index is introduced as a set of numbers showing the relative values of a variable during the months of a year. If the original monthly data are divided by the corresponding seasonal index numbers, the resulting data are said to be *deseasonalized*, or adjusted for seasonal variations. A moving average smooths the data evolution, therefore it removes fluctuations and high frequency oscillations, allowing a better observation of the trend.

Similarity search

There are two types of similarity searches: subsequence matching and whole sequence matching [73]. In both cases, A similarity measure is computed between the two sequences, typically by applying the Euclidean distance. There are two concepts that must be taken into account when performing this similarity search: the *baseline*, or offset of the signal or the data, and the *scale* or amplitude. One of the methods to solve differences in baseline and scale is the normalization of the data. The gaps in the sequence must also be taken into account. Two subsequences are considered similar and can be matched if “one lies within an envelope of ϵ width around the other, ignoring outliers [73]”. Two sequences are similar if “they have enough non-overlapping time-ordered pairs of similar subsequences [73]”.

5.4.3 Sequence data pattern mining

In Sequence Pattern Mining, the objective is the mining of frequently occurring events in a sequence of data, which do not necessarily need to be an ordered temporal sequence. A sequence is formed by an ordered list of *events* [73]. Each event can be formed by an unordered set of items or *itemset* such as, for instance, a list of items bought in a supermarket. Each sequence can be formed by a set of subsequences, if the same subsequence is found in other sequences of the database.

Han and Kamber [73] describe algorithms for Sequential Pattern Mining, some of them being the following:

- The **GSP** (Generalized Sequential Patterns) algorithm [163] scales linearly with the number of data sequences. The algorithm works by performing scans of the data, where in each pass new frequent sequences are obtained from the initial *candidate* sequences, or the frequent sequences found in the previous iteration. With each scan, the number of frequent sequences to be found is augmented, until no more frequent patterns are found.
- **SPADE** (Sequential PAttern Discovery using Equivalent classes) [164] improves the GSP algorithm by applying a vertical ID list database format, instead of a horizontal one. This approach reduces the number of scans of the sequence database. However, the breadth-first strategy for searching patterns that both algorithms apply makes to generate large sets of candidates at each iteration.
- The **PrefixSpan** algorithm [165] improves the performance of the GSP algorithm by building prefix patterns, which it concatenates with suffix patterns to find frequent patterns, avoiding candidate generation. From each prefix patterns, a projected database with suffix patterns is formed. This method generates no candidate sequences in the mining process. However, it may generate many projected databases, one for each frequent prefix subsequence [73].

5.5 Conclusions

The data mining objectives and techniques have been presented in this Chapter. As has been previously discussed in this document (see Chapter 4), there is a need of data analysis and extraction of useful information from daily load profiles of energy consumption. The specific objective is the following: to obtain knowledge from large amounts of energy consumption data. These data are being available by the increasing implementation of AMI in households, as has been described in Part I. There will be a need in the near future of how to process these data to obtain interesting and useful information in form of *patterns*, that group similar consumptions in *clusters* and inform about the evolution of the load profiles variation in time, through a sequence of days.

From all the data mining objectives described in this Chapter, it is the cluster analysis and, more specifically, the cluster evolution analysis, or the dynamic clustering of time series data, the data mining objective which is considered of relevance for the analysis of these data from energy consumption, arranged in daily vectors of 24 hours or dimensions. This objective of data mining described in the literature fulfills the need of data analysis described in Chapter 4, and therefore, it is proposed in this thesis as the solution to be adopted to analyze the data. For this reason, in the following Chapters a state of the art in clustering techniques (Chapter 6) and clustering of time series data (Chapter 7) have been described.

6

Description of clustering techniques

6.1 Introduction

A complete definition of clustering can be found in the book by Oliveira and Pedrycz[90], where clustering is defined as the following:

> "Clustering is an unsupervised learning task that aims at decomposing a given set of *objects* into subgroups or *clusters* based on similarity. The goal is to divide the data set in such a way that objects (or example cases) belonging to the same cluster are as similar as possible, whereas objects belonging to different clusters are as dissimilar as possible [...]. Cluster analysis is primarily a tool for discovering previously hidden structure in a set of unordered objects [...]. By arranging similar objects into clusters one tries to reconstruct the unknown structure in the hope that every cluster found represents an actual type or category of objects. Clustering methods can also be used for data reduction purposes."

Clustering algorithms, although referred as one of the methods for data mining and knowledge discovery in databases, have been applied in other research areas with different purposes. For instance Díez [166] applies clustering algorithms for systems identification in the obtention of SISO models from I/O data.

In this Chapter, clustering algorithms are described. The following Sections describe the definition of data objects, patterns and classes; the similarity and dissimilarity measures described in the literature to compare clusters; the different criteria used to classify clustering algorithms; and a review of the current state of the art on clustering algorithms found in the literature.

6.2 Definition and types of data objects

An **object** can be defined as an element represented by a number of attributes, characteristics or dimensions. These characteristics are usually expressed in vector form, as can be seen in (6.1), where each element, from 1 to d, represents a different characteristic or dimension.

$$x = [x_1 x_2 \dots x_d] \tag{6.1}$$

Databases contain many objects, usually arranged in tables, in matrix form, like in (6.2), where each row is a different object and the number of columns must match the number of dimensions or characteristics of all the objects. This configuration implies that all the objects have the same number of dimensions; however objects may have attributes with no values, or with erroneous values due to noise or failure during the data acquisition process.

$$X = [x_1 x_2 \dots x_n]^T = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1d} \\ x_{21} & x_{22} & \dots & x_{2d} \\ \vdots & \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nd} \end{pmatrix} \tag{6.2}$$

According to Mitra *et al.* [74], **data sets** are a series of objects or facts that form a group. A **pattern** is “an expression E, in a language L, that is used to describe the facts of a subgroup F_E of the group F, in a way that describing subgroup F_E with the pattern E is easier and faster than the enumeration of all the objects belonging to subgroup F_E [74].”

Clusters or classes, are groups of objects, formed under some specified similarity criterion. Each cluster has usually an object that is representative or prototype of the objects pertaining to the cluster; this object is called the centroid. Figure 5.2 displays an example of clustering in a set of $2D$ objects. The centroids usually coincide with the center of gravity of the cluster, however this may vary, according to the algorithms used and the data types being classified. Cluster centroids are usually denoted with letter c or z, depending on the authors, whereas objects of the data set are usually denoted with letter x_i.

According to the description given by Jain and Dubes [78], classification of data can be made based on two approaches: the **type** and the **scale** of the data. The type refers to the “degree of quantization” of the data [78], or the number of values a variable can have. For instance, *binary* variables can only have two values. *Discrete* variables are those variables whose possible values can be enumerated in a finite sequence of, usually small, number of options. For instance age, number of children or ordinal numbers are considered discrete variables. On the other hand, those variables which can have any value within a specified range are *continuous* variables. Sensor measures usually are, for instance, continuous variables.

The data scale classifies the variables according to the significance of the values that the data may have. The data are classified in *quantitative* or *qualitative*. Qualitative refers to data that have a meaning only when referred to other qualitative data, whereas quantitative data have a meaning by themselves. For instance, values that are used as labels or categories in nominal scales, like the Likert scale [167], or a colors palette, are qualitative, since their whole universe of data values has to be seen in order to fully understand their meaning; on the contrary, values like age, height, volume, temperature, etc, are continuous variables, since they have a meaning by themselves.

6.3 Cluster distance measures

Distance measures between clusters can be classified in two generic groups: the ones that measure similarity, and the ones that measure dissimilarity. Both groups of indices are complementary (i.e., the higher the similarity, the lower the dissimilarity between two given clusters, A and B). Distance measure indices are usually denoted with the expression $d(i,k)$, and must meet conditions (6.3), (6.4) and (6.5) [78].

$$d(i,i)\begin{cases} = 0, \forall\ i & \text{(dissimilarity)} \\ \geq max_k d(i,k), \forall\ i & \text{(similarity)} \end{cases} \tag{6.3}$$

$$d(i,k) = d(k,i), \forall\ (i,k) \tag{6.4}$$

$$d(i,k) \geq 0, \forall\ (i,k) \tag{6.5}$$

Following, some of the most common distance measures used in clustering algorithms are included.

6.3.1 The Minkowski metric

It is a dissimilarity measure suitable for quantitative values. Its definition is given in (6.6).

$$d(i,k) = \left(\sum_{j=1}^{d} |x_{ij} - x_{kj}|^r \right)^{\frac{1}{r}} \text{, where } r \geq 1 \tag{6.6}$$

The Minkowski metric is different as a function of the r parameter. The most used Minkowski metrics are three, described below: the *Manhattan* distance, the *Euclidean* distance and the *Chebyshev* distance (see Figure 6.1).

Manhattan or *taxicab* distance

When $r = 1$, the Minkowski metric turns into the $norm - 1$ metric (6.7):

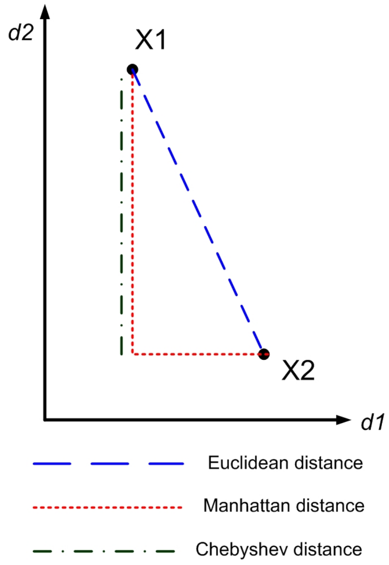


Fig. 6.1. Minkowski metrics.

$$d(i,k) = \sum_{j=1}^{d} |x_{ij} - x_{kj}| \tag{6.7}$$

This distance metric is called Manhattan or *taxicab* distance, because of its similarity to a taxi driving through the square blocks of Manhattan island, NY. In the case of having all the variables or dimensions being binary, the resulting distance is called the *Hamming* distance [168], which is a direct measure of the differing number of attributes or dimensions between two binary objects. The Hamming code is widely used in telecommunications, to check errors in the transmission of binary signals [169].

Euclidean distance

When $r = 2$, the $norm - 2$ distance is obtained (6.8):

$$d(i,k) = \left(\sum_{j=1}^{d} |x_{ij} - x_{kj}|^2 \right)^{\frac{1}{2}} = \sqrt{(x_i - x_k)^T (x_i - x_k)} = \|x_i - x_k\| \tag{6.8}$$

This metric is the *Euclidean distance*, defined as the shortest quantitative distance between two objects. The Euclidean distance is the most used in clustering algorithms for quantitative data.

Chebyshev or supremum norm distance

When r is assigned an infinite value ($r \to \infty$) (6.9), the resulting distance is the maximum among all the characteristics or dimensions of both objects (6.10).

$$\lim_{r\to\infty} \left(\sum_{j=1}^{d} |x_{ij} - x_{kj}|^r \right)^{\frac{1}{r}} \tag{6.9}$$

$$d(i,k) = max_{1\leq j\leq d}|x_{ij} - x_{kj}| \tag{6.10}$$

This distance, based on the supremum or infinite norm, is known as the *Chebyshev* distance.

6.3.2 Mahalanobis distance

The Mahalanobis distance incorporates the correlation between features and standardizes each feature to zero mean and unit variance [78]. Its expression can be seen in (6.11).

$$d(i,k) = (x_i - x_k)^T \varphi^{-1}(x_i - x_k) \tag{6.11}$$

Where φ is the covariance matrix (6.12) among the different features or characteristics.

$$\varphi = \frac{1}{N} \sum_{k=1}^{N} (x_k - \bar{x})(x_k - \bar{x})^T \tag{6.12}$$

If φ^{-1} is assigned the identity matrix value (I), the Mahalanobis distance is the squared euclidean distance ($\|x_i - x_k\|^2$).

6.3.3 Sample correlation coefficient

The sample correlation or Pearson coefficient is a quantitative similarity measure, used to determine the linear relationship between sequences of data, like time series. Correlation is one of the objectives of the data mining and, as such, has been described in the previous Chapter, in Section 5.2.7.

6.3.4 Matching coefficients

Matching coefficients are proximity indices for nominal and binary data. The proximity is measured by counting the number of matchings per characteristic between two objects. Three of the most common matching coefficients are the following: the *simple matching* coefficient, the *Jaccard* coefficient and the *overlap* metric.

Since they are addressed to qualitative data, matching coefficients take into account the meaning of the value: if two objects have at their characteristic a value of 1, their matching is **positive**, because 1 would be in this case equivalent to TRUE or “present”. On the contrary, if the two objects have at the specific characteristic a value of 0, the matching is **negative**, because it is equivalent to a NULL or “void” significance.

Simple matching coefficient

The simple matching coefficient counts the number of matchings, either positive or negative. Its expression is given in (6.13).

$$d(i,k) = \frac{\sum \text{positive matches} + \sum \text{negative matches}}{\sum \text{positive matches} + \sum \text{negative matches} + \sum \text{non matches}} \tag{6.13}$$

Jaccard coefficient

The Jaccard coefficient takes only into account the number of positive matches (6.14).

$$d(i,k) = \frac{\sum \text{positive matches}}{\sum \text{positive matches} + \sum \text{non matches}} \tag{6.14}$$

Overlap metric

The overlap metric is a dissimilarity index. It counts the number of non-matches between two objects. Its expression is given in (6.15).

$$\begin{gathered} d(i,k) = \textstyle\sum_{j=1}^{d} \delta_{x_{ij},x_{kj}} \\ \text{where } \delta_{x_{ij},x_{kj}} = \begin{cases} 0 \text{ if } x_{ij} = x_{kj} \\ 1 \text{ if } x_{ij} \neq x_{kj} \end{cases} \end{gathered} \tag{6.15}$$

6.3.5 Entropy

Despite its formulation by the second law of thermodynamics, the measure of entropy for clustering algorithms is a measure of the uncertainty of a random variable [170]. Given a random variable X, the range of possible values $S(X)$, and its associated probability function $p(x)$, the entropy $E(X)$ is defined as in the expression (6.16).

$$E(X) = -\sum_{x \in S(X)} p(x) log(p(x)) \tag{6.16}$$

Entropy in clustering algorithms yields a measure of diversity or *heterogeneity* among objects, with the objective to group them in *homogeneous* clusters [171][172][173].

6.3.6 Kullback-Leibler distance

The Kullback-Leibler distance, also called relative entropy, is a measure of the divergence between two probability distributions [174]. Given two distributions, p and q, from a series T, the relative entropy is defined by the expression in (6.17).

$$D_{KL}[p\|q] = \sum_{t \in T} p(t) log \frac{p(t)}{q(t)} \tag{6.17}$$

Kullback-Leibler distance is used to estimate the gain in divergence when combining two clusters, for instance for distributed data clustering [175].

6.3.7 Gowda and Diday distance

This dissimilarity measure, proposed in [176], defines three different measures for two variables, A_k and B_k: *position*, *span* and *content*. The summation of these three yields the Gowda and Diday distance. Given the following definitions:

$$\begin{cases} al = \text{ lower limit from } A_k \\ bl = \text{ lower limit from } B_k \\ a\mu = \text{ upper limit from } A_k \\ b\mu = \text{ upper limit from } B_k \\ inters = \text{ intersection distance between } A_k \text{and } B_k \\ ls = span = |max(a\mu, b\mu) - min(al, bl)| \\ U_k = \text{ difference between the highest and the lowest values from the} \\ \text{k dimension or characteristic for all the data objects} \\ l_a = |a\mu - al| \\ l_b = |b\mu - bl| \end{cases}$$

Position, span and content distances are obtained as shown in expressions (6.18), (6.19) and (6.20).

$$D_p(A_k, B_k) = \frac{|al - bl|}{U_k} \tag{6.18}$$

$$D_s(A_k, B_k) = \frac{|l_a - l_b|}{l_s} \tag{6.19}$$

$$D_c(A_k, B_k) = \frac{|l_a + l_b - 2inters|}{l_s} \tag{6.20}$$

6.4 Normalization

The normalization of the object features is usually recommended [166][78]. An example of this situation is given by the Euclidean distance, one of the most similarity measure used for continuous data: when using Euclidean distance, features with larger ranges are assigned more weighting than those with smaller ranges of data values.

However, some exceptions are considered in the literature where normalization is not necessary, like in cases where all the features or dimensions of the object have the same nature, e.g. a 24-hours energy consumption pattern [84].

Different techniques of normalization have been described in the literature. The most common are the following:

1. In case the maximum value is known, and all the compared characteristics or dimensions have the same magnitude, one direct normalization method is to divide all the values by the maximum of the series, normalizing all the data to equivalent values between 0 and 1 (which would be the maximum).
2. Substract the mean or average value at each feature j, as indicated in expressions (6.21) and (6.22).

$$\hat{x_{ij}} = x_{ij} - m_j \tag{6.21}$$

$$m_j = \frac{1}{n}\sum_{k=1}^{n} x_{kj} \tag{6.22}$$

3. Scale all the features to zero mean and unit variance (See expressions (6.23) and (6.24)).

$$\hat{x_{ij}} = \frac{x_{ij} - m_j}{s_j} \tag{6.23}$$

$$s_j^2 = \frac{1}{n}\sum_{k=1}^{n}(x_{kj} - m_j)^2 \tag{6.24}$$

6.5 Classification of clustering algorithms

Clustering algorithms can be classified according to different criteria [174][166][78], based on how the clustering is performed (the model applied), or which kind of data are being clustered. The most common criteria are the following:

- **Partitional vs. hierarchical clustering**. Partitional algorithms cluster the data at a single level, by optimizing a defined cost index, whereas hierarchical algorithms perform a nested sequence of partitions of the data, forming trees of hierarchical clusters which are called *dendograms*. Examples of partitional algorithms are the *K-means* [177], the *PAM* (Partitioning Around Medoids) [178] or the *CLARA*(Clustering LARge Applications) [178]. Hierarchical clustering algorithms work in both ways, *top-down*, starting from one, single cluster, which splits to lower hierarchical levels, or *bottom-up*, by defining a number of clusters, which are being combined in different levels until no partition is found. Hierarchical algorithms cluster objects at different levels, giving as a result a tree or *dendogram* where each branch represents a different cluster. Examples of hierarchical clustering algorithms are *BIRCH* [147] or *CURE* [179].

- **Exclusive vs. non-exclusive membership**. This classification separates algorithms into those where an object can only belong to a unique cluster, or those where objects belong to more than one cluster, with different grades of membership. Typical non-exclusive membership algorithms are those based on fuzzy logic [95], such as the *fuzzy c-means* (FCM) algorithm [180][106] or the GK or Gustafson-Kesel algorithm [102]; or those based on possibilistic membership, such as the *possibilistic c-means* or *PCM* algorithm [181].
- **Quantitative vs. non-quantitative data**. Quantitative clustering algorithms base their calculation of similarity among objects by applying quantitative distance measures, such as the Euclidean distance. Examples of quantitative algorithms are the *K-means* [177], FCM [180][106] or Gustafson-Kesel (*GK*) algorithms [102]. However, usually objects stored in a database are composed of variables of different nature, i.e., numeric, ordinals, qualitative descriptions or selections, etc. A specific branch of clustering algorithms has been developed in the last years devoted to the cluster analysis of non-quantitative data, or data with quantitative and qualitative variables or dimensions. Examples of these algorithms are *K-modes* [182], ROCK (*RObust Clustering using linKs*) [183] [184], CACTUS (*Clustering Categorical Data Using Summaries*) [185] [184] or COOLCAT [171], among others.

Other classifications of clustering algorithms can be found in the literature, referred mainly to specific purpose clustering, such as the following:

- **Density-based vs. proximity-based clustering**. Algorithms based on density group objects in clusters according to the objects proximity and cluster volume relation, i.e. a density measure. Examples of clustering algorithms based on density are DBSCAN (*Density-Based Spatial Clustering of Applications with Noise*) [186] or DENCLUE (*DENsity-based CLUstEring*) [187].
- **Spatial, geographical and grid-based clustering**. These algorithms are designed to cluster spatial and geographical objects, by applying different techniques. For instance, grid-based algorithms partition the space and search similar objects at the resulting cells. Examples of grid-based

algorithms are STING (*STatistical INformation Grid*) [188], *WaveCluster* [153] and CLIQUE (*CLustering In QUEst*) [189]. Spatial clustering algorithms cluster objects according to spatial properties, like geometrical features, distance to centers, area and radius, etc. Examples of spatial clustering algorithms are GDBSCAN (*Generalized Density Based Spatial Clustering of Applications with Noise*) [190] and GRAVIclust (Gravity Based Spatial Clustering) [191].

- **Distributed data clustering**. These algorithms are designed to cluster objects that are distributed among different repositories or large databases in networked systems, such as the internet. Pattern recognition and clustering techniques can be useful to detect local faults, or to perform a global diagnosis based on the activity of decentralized data mining agents on distributed nodes [192]. Clustering can be based on a global analysis of clusters in the network, performed sequentially node by node, or it can be done in an asynchronous way, by combining incoming, local information from distributed nodes. Privacy and data encryption are key issues in these kind of algorithms [175][186][193][194].
- **Model-based data clustering.** These algorithms apply clustering on parameter values for predefined mathematical models [195]. A representative example of these kind of algorithms is the EM *Expectation-Maximization* algorithm [196].

Following, the main algorithms found in the literature are described, attending to their main characteristic or differentiation criterion previously described. It must be taken into account, however, that most of the algorithms can belong to more than one classification. For instance, the FCM algorithm is of fuzzy clustering type, but it is also a partitional algorithm.

Since the objective of this thesis is the clustering of load profiles of energy consumption time series, the algorithms described are mostly used to cluster quantitative data. However, clustering algorithms for data objects with quantitative and qualitative information have also been included.

6.6 Partitional clustering algorithms

6.6.1 Chain-map clustering

Chain–map algorithm [166] [197] is one of the simplest clustering algorithms. Its procedure is as follows:

1. Select an object from the data set. This will be the first object of the chain.
2. Sort the rest of elements according to similarity to the first object, in ascending order. The most similar objects will be the closest to the first object in the chain (6.25). Similarity is measured with the Euclidean distance.

3. The sequence of objects with their Euclidean distance below a predefined threshold are assigned to a cluster.

$$z_i(0), z_i(1), \ldots, z_i(N-1) \tag{6.25}$$

In the chain map algorithm, the number of clusters is not an input parameter, but the threshold or maximum Euclidean distance above which the beginning of a new cluster is considered. Therefore, two are the key issues that affect the efficiency of the algorithm: the threshold value and the sorting of the data.

This is a very simple clustering algorithm which may be used as a fast initial analysis to determine the number of clusters, or as an initialization procedure for other clustering algorithms. It has been also used as very fast way to identify points of interest or change of trend in a time series.

6.6.2 Max-min

Max–min algorithm [166] does not need to define an initial number of clusters either. The procedure of the algorithm is the following:

1. Select an initial object at random, which will be assigned the prototype of cluster 1.
2. Compute all the Euclidean distances of the remaining objects to the prototype of cluster 1.
3. The object with the highest distance is selected as the prototype of cluster 2.
4. Compute all the Euclidean distances of the remaining objects to the prototype of cluster 2.
5. From the two obtained distances for each of the remaining objects, select the smallest. From the resulting group of distances, select the object with the highest distance.
6. If the selected distance is higher than an f–factor weighted distance between prototypes of clusters 1 and 2 (see (6.26)), then a new cluster is created (cluster 3), being the selected object the prototype of the new cluster.
7. The procedure is repeated, until the highest of the smallest distances among the remaining objects is no longer higher than the average of the f–factor weighted distances between all the clusters' prototypes. Then the Euclidean distances of all the remaining objects to all the created prototypes are computed again, and each object is assigned to the cluster whose distance to prototype is the smallest.

$$d_{max} > f \ d(z_1, z_2), \ 0 < f < 1 \tag{6.26}$$

This algorithm has as an inconvenient the arbitrary selection of the value of *f*, which highly influences the resulting number of clusters and prototypes.

6.6.3 K-means

The K-means algorithm [166][177] is one of the most known and used clustering algorithm, due to its efficiency and robustness. The name refers to the number k of clusters to be found, defined as an initial parameter. The procedure of the algorithm is the following:

1. Select k objects and set them as the initial prototypes of the k clusters that are to be found. This can be done at random from the data set. However, this choice can delay the execution time of the algorithm and affect its performance. Therefore other options are suggested instead, such as setting the initial prototypes based on heuristics or knowledge of the data to be clustered; or to apply specific algorithms to initialize centroids, such as the *K-means* ++ algorithm [198].
2. Compute all the Euclidean distances of the remaining objects to the k prototypes. Assign each object to the cluster with the smallest distance.
3. Compute clusters prototypes or centroids as the average or mean value from all the objects that belong to the cluster, with the objective to minimize the function shown in (6.27). This objective function is a summation of all the k summations of the distances from all the objects to the centroid of each cluster. The minimization of the objective function is described in Appendix B.
4. Proceed with the two previous steps until a termination condition is reached, like the variation in centroids falling under a predefined limit.

$$J = \sum_{i=1}^{k} \left(\sum_{j, z_j \in A_i} \|z_j - c_i\|^2 \right) \tag{6.27}$$

The efficiency of the K-means algorithm highly relies on the parameter of the number of clusters (k) to be found, its adequacy to the real number of clusters, and the initialization of prototypes. Moreover, if the initialization prototypes are not adequate, the algorithm may converge to a local optimum rather than a global one [79].

6.6.4 PAM

PAM (*Partitioning Around Medoids*) [184] [178] is an extension of the K-means algorithm, where each cluster is represented by the most centric element of the cluster; this element is called the *medoid*, and it does not have to coincide with the centroid in other clustering algorithms, which is computed as the average of all the elements pertaining to the cluster. The medoid is equivalent, therefore, to the statistical *median* of a data set (i.e. the middle point in a series).

The procedure of the algorithm is the same as the K-means, only differing in the way to compute the medoids at each iteration (the median instead of the average).

6.6.5 CLARA

The CLARA (*Clustering Large Applications*) algorithm [184] [178] divides the original database in samples of equal size, and then applying the PAM algorithm on the resulting sets, selecting the best result from all the resulting partitions. The objective of this algorithm is to provide a fast response when clustering large databases, minimizing computational cost (however losing reliability in the results).

6.7 Hierarchical clustering algorithms

6.7.1 BIRCH

the BIRCH (*Balanced Iterative Reducing and Clustering using Hierarchies*) algorithm [184] [147] generates for each cluster three data: the number of objects in the cluster, the summation of all the values from all the dimensions of all the data objects belonging to the cluster, and the summation of the squared values from all the dimensions from all the objects belonging to the cluster. With this information, BIRCH builds a tree of clusters, called *CF-tree* (*Cluster Features tree*), following the same procedure:

- Generate an initial CF-tree, assigning the objects to different branches based on a threshold distance value.
- Adjust the number of branches by modifying the threshold distance value, until an adequate number is found.
- Apply a clustering algorithm, such as K-means, on the data from each branch, generating a seconfd level of clusters.
- Redistribute centroids among the branches if necessary, in the case that similar clusters are found in different branches.

6.7.2 CURE

CURE (*Clustering Using REpresentatives*) [184] [179] is an agglomerative hierarchical algorithm that follows a bottom-up approach, by considering all the objects initially as clusters, and then grouping them iteratively. From each new cluster, CURE gathers its most extreme values and moves them towards the cluster centroid, a distance which is the average value of all the objects in the cluster (i.e. the centroid).

6.8 Fuzzy clustering algorithms

6.8.1 Definition of fuzzy partition

Fuzzy clustering algorithms are based on a **non-exclusive partition** of the membership, i.e., an object can belong to more than one cluster, having a certain degree of membership to each one. In an exclusive partition, all the objects can belong exclusively to one group or cluster, and must follow the properties indicated in (6.28), (6.29) and (6.30).

$$A_i \bigcap A_j = \varnothing,\ 1 \leq i \neq j \leq c \tag{6.28}$$

$$\varnothing \subset A_i \subset Z,\ 1 \leq i \leq c \tag{6.29}$$

$$\bigcup_{i=1}^{c} A_i = Z \tag{6.30}$$

Expression (6.28) indicates that the intersection between two groups yields no common values (being c the number of groups); expressions (6.29) and (6.30) indicate that there are no empty groups, and that the union among all the groups must yield the complete data set of objects, Z.

The **membership matrix** U is defined as the matrix $U = [\mu_{ik}]$ of dimensions c x N which indicates, for each object, its degree of membership to all the c clusters. In an exclusive partition, the resulting partition is defined in expression (6.31), where the term "hc" stands for *hard* or exclusive clustering.

$$M_{hc} = \left\{ U \in R^{cxN} | \mu_{ik} \in \{0,1\}, \forall i, k; \sum_{i=1}^{c} \mu_{ik} = 1, \forall k; 0 < \sum_{k=1}^{N} \mu_{ik} < N, \forall i \right\} \tag{6.31}$$

As can be seen in the expression (6.31), membership degrees in an exclusive partition can only have two values, 0 or 1. In a **fuzzy partition**, however [199][166], degrees of membership can acquire any value between 0 and 1, as can be seen by the definition given in (6.32), where the term "fc" stands for *fuzzy* clustering.

$$M_{fc} = \left\{ U \in R^{cxN} | \mu_{ik} \in [0,1], \forall i, k; \sum_{i=1}^{c} \mu_{ik} = 1, \forall k; 0 < \sum_{k=1}^{N} \mu_{ik} < N, \forall i \right\} \tag{6.32}$$

In a fuzzy partition, the clusters become fuzzy membership zones or areas that cover the full range of data values, in resemblance with the fuzzy membership functions described in Section 5.3.1. Each data belongs to more than one cluster with a degree value that varies between 0 and 1, in accordance, again, with the *fuzzification* process of the input data described in Section 5.3.1. Fuzzy clustering allows a soft transition of the data through membership to the different clusters or, in other words, allows to describe an object by the combination of all the clusters the object belongs to, with different *weights* or membership values. This approach poses advantages, and also some inconveniences. The main advantage is that it enriches the description of the data, as being related to more than one cluster, and therefore captures the singularities of objects whose characteristics may place them on the bound between two different clusters. Allowing to assign a number of objects to all the different clusters and their resulting centroids or prototypes unveils, however, the main inconvenience or disadvantage of fuzzy clustering, which is its unclear definition of the membership of each object to a unique cluster and prototype. If an object belongs to all the clusters with more or less the same value of membership, this means that the clustering process is losing definition. If the same situation is given for a significant percentage of the data, then the resulting clusters will very much look the same, and therefore the cluster analysis will not be able to identify clear patterns and their main differences.

In both partitions (fuzzy and hard), a condition is defined that all the degrees of membership for each object must sum 1. This is a **non–possibilistic** constraint. The possibilistic partition can also be given, and is used on possibilistic clustering algorithms, such as the PCM [181], or the possibilistic FCM algorithm [200][201]. In a possibilistic partition, the condition is that each object must have at least one of its degrees of membership to clusters greater than 0.

6.8.2 Fuzzy C-means (FCM)

The FCM or fuzzy C-means algorithm [166] is based on the application of fuzzy sets, first described by L. Zadeh [95] to the K-means algorithm. The FCM algorithm was developed by J. Bezdek and Dunn [180][106]. The algorithm is based on the minimization of the performance index given in (6.33), a weighted sum of the quadratic error obtained when defining c_i objects as centroids or prototypes of the c clusters, subject to the constraint that the summation of membership values of each object to all the clusters must equal one (6.34).

$$J(Z;U,C) = \sum_{i=1}^{c}\sum_{k=1}^{N}(\mu_{ik})^m \|z_k - c_i\|_B^2 \tag{6.33}$$

$$\sum_{i=1}^{c}\mu_{ik} = 1 \tag{6.34}$$

The variables defined in the index are: Z, which is the number of objects; the membership matrix U, whose elements μ_{ik} have a power factor m equivalent to the “fuzziness” level, and must have a value greater than one; and the matrix of centroids C, representing the centroid for each cluster. The expression $\|z_k - c_i\|_B^2$ indicates a measure of the distance between each centroid and the datum, as can be seen in (6.35).

$$\|z_k - c_i\|_B^2 = (z_k - c_i)^T B(z_k - c_i) = D_{ikB}^2 \tag{6.35}$$

Matrix B is the norm of the distance, and determines the geometry of the clusters. When B is equal to the identity matrix, then the Euclidean distance is obtained; whereas if B is equal to the inverse of the covariance matrix, the result is the Mahalanobis distance (6.3.2).

The minimization of the index (described in Appendix B), taking into account the constraint included as a Lagrange multiplier, yields two expressions, (6.36) and (6.37), used to obtain the values of centroids and membership matrix in an iterative process, described in the following steps:

1. Initialize membership matrix U with random values.
2. Compute the cluster centroids according to Eq. (6.37).
3. Obtain all the distances of the objects to the centroids of the clusters (6.35).
4. Recompute membership matrix U applying Eq. (6.36) when $D^2_{ikB} > 0$ for any i, k, or applying the expression displayed in (6.38) in any other case.
5. Verifiy if termination conditions are met. If not, start again from the second step of this sequence. Termination conditions are usually stated as a threshold value ε of variation in the membership matrix compared to the last iteration; if the threshold value is not reached, the sequence stops. This condition is expressed in Eq. (6.39).

$$\mu_{ik} = \frac{1}{\sum_{j=1}^{c}\left(\frac{D^2_{ikB}}{D^2_{jkB}}\right)^{\frac{2}{m-1}}}, \; 1 \leq i \leq c, \; 1 \leq k \leq N \tag{6.36}$$

$$c_i = \frac{\sum_{k=1}^{N}(\mu_{ik})^m z_k}{\sum_{k=1}^{N}(\mu_{ik})^m}, \; 1 \leq i \leq c \tag{6.37}$$

$$\begin{gathered} \mu_{ik} = 0, \text{ if } D_{ikB} > 0 \\ \mu_{ik} \in [0,1], \text{ with } \sum_{i=1}^{c}\mu_{ik} = 1 \text{ for the rest} \end{gathered} \tag{6.38}$$

$$\|U_{(k)} - U_{(k-1)}\| < \varepsilon \tag{6.39}$$

6.8.3 Gustafson-Kessel or GK

The GK algorithm is a variation of the FCM, proposed by Gustafson and Kessel in 1979 [102][166]. The algorithm assigns different norms B to the different clusters, allowing the formation of clusters with different shapes.

Having a vector B which contains c norms, the objective function from FCM is modified as indicated in (6.40).

$$J(Z;U,C,B) = \sum_{i=1}^{c}\sum_{k=1}^{N}(\mu_{ik})^m \|z_k - c_i\|^2_{B_i} \tag{6.40}$$

The scope of possible values for the B_i elements is limited by assigning a fixed value to its determinant (6.41).

$$|B_i| = \rho_i, \quad \rho > 0 \tag{6.41}$$

The result from minimizing the objective function is a new expression to compute the norms (6.42), where the F_i variable is the *covariance* matrix from class i, obtained in (6.43). The formula to compute the centroids is similar to the FCM (6.37), but the computation of membership to clusters includes a norm that may be different for each group (6.44).

$$B_i = [\rho_i det(F_i)]^{\frac{1}{n}} F_i^{-1} \tag{6.42}$$

$$F_i = \frac{\sum_{k=1}^{N} (\mu_{ik})^m (z_k - c_i)(z_k - c_i)^T}{\sum_{k=1}^{N} (\mu_{ik})^m} \tag{6.43}$$

$$\mu_{ik} = \frac{1}{\sum_{j=1}^{c} \left(\frac{D_{ikB_i}^2}{D_{jkB_i}^2} \right)^{\frac{2}{m-1}}}, \ 1 \le i \le c, \ 1 \le k \le N \tag{6.44}$$

The steps followed by the algorithm are the following:

1. Initialize membership matrix U with random values.
2. Compute centroids, according to (6.37).
3. Compute covariance matrix in each cluster (6.43).
4. Compute new distances, applying the norm in each case, according to (6.42) and (6.35).
5. Compute new membership values (6.44).
6. Verify stopping condition, or go back to step 2.

6.8.4 Fuzzy Maximum Likelihood (FMLE)

The (*Fuzzy Maximum Likelihood*) algorithm [103][166] is another extension of the FCM that includes a norm with an exponential term, computed as indicated in expressions (6.45), (6.46) and (6.47). The procedure is similar to the GK algorithm.

$$D_{ikG_i}^2 = \frac{\sqrt{det(G_i)}}{P_i} \ exp \left[\frac{1}{2} (z_k - c_i)^T G_i^{-1} (z_k - c_i) \right] \tag{6.45}$$

$$G_i = \frac{\sum_{k=1}^{N} \mu_{ik} (z_k - c_i)(z_k - c_i)^T}{\sum_{k=1}^{N} \mu_{ik}} \tag{6.46}$$

$$P_i = \frac{1}{N} \sum_{k=1}^{N} \mu_{ik} \tag{6.47}$$

6.9 Density-based clustering algorithms

6.9.1 GDBSCAN

The GDBSCAN *(Generalized Density Based Spatial Clustering of Applications with Noise)* algorithm [190] is based on the density of the resulting clusters. To identify an aggregation of data as a cluster, the measured density for the group must reach a defined threshold. The density is computed as the relation between the number of objects and the geographical area they occupy. This allows the formation of clusters with very different and irregular shapes, and has as advantage the removal of outliers from the resulting clusters.

6.9.2 DENCLUE

The DENCLUE (*DENsity-based CLUstEring*) algorithm [187] applies the concept of *functions of influence* to address the influence that each object has to the other objects nearby. These functions of influence are similar to the activation functions used in ANNs: upon a certain threshold of distance, the output changes state, usually from inactive (0) to active (1). Gaussian or sigmoid functions are commonly used as functions of influence.

The density is then computed as the sum of the outputs from all the functions of influence for each object. The objects with the maximum densities become the *attractors* of the other objects, and therefore the main characteristic objects from the resulting clusters.

6.10 Grid-based clustering algorithms

6.10.1 STING

STING (*STatistical INformation Grid*) [184] [188] partitions the space in levels, in a number of cells with a hierarchical structure. From each cell the algorithm extracts statistical information from its data (mean, median, variance, distribution function of the objects within). Then the cells are partitioned again in the following level, forming a hierarchical tree, similar to the BIRCH algorithm [147]. When the search has finished, the resulting clusters are formed by grouping the most similar cells with each other.

6.10.2 CLIQUE

CLIQUE (*CLustering In QUEst*) [189] [184] also performs partitions in different levels, however in this case each new level is one of the dimensions of the data objects. The partitions are performed until the n dimensions of the data are reached. The partitions are in form of hyper-rectangles. From the first dimension, the algorithm looks for the most dense regions. Once these initial clusters are found, a new dimension is added and the space is partitioned in rectangles, looking for the most dense ones. The third dimension partitions the space in 3D cubes. The process is repeated until the number of dimensions of the data is reached.

6.11 Spatial or geographical clustering algorithms

6.11.1 GRAVIclust

GRAVIclust (Gravity Based Spatial Clustering) [191] searches groups of data geographically defined by area, center, radius and density. The resulting clusters have, therefore, a circular shape and are formed by the most dense geographical areas of the data space. The Euclidean distance is used as the similarity measure between objects. The objective or cost function that the algorithm optimizes is the one expressed in (6.48).

$$J = \sum_{i=1}^{k} \sum_{p \in C_i} d(p, L_i) \tag{6.48}$$

Where k is the number of clusters, p is an object that belongs to cluster C_i, and d is the distance of p to the center of gravity of cluster C_i, denoted as L_i. Each object belongs only to one cluster (the one whose distance d is the lowest). The center of gravity of each cluster becomes its centroid or most representative element.

6.12 Clustering algorithms for distributed data

6.12.1 Collective Principal Component Analysis

Kargupta et al. [194] propose a clustering algorithm for data distributed in different nodes of a network (such as distributed servers or repositories in the internet). The authors propose to use Principal Component Analysis or PCA [202], whose objective is to provide a reduced representation of the data by the linear combination of the dimensions or characteristics of the data in a new reduced set of dimensions or Principal Components.

The resulting algorithm is the Collective Principal Component Analysis or CPCA. Each node in the network performs the algorithm individually; the results are collected and sent to a central node. The clustering algorithm is performed again on all the results received from the nodes, and the resulting prototypes are sent back to all the nodes as the patterns or prototypes. A new clustering process is done at all the nodes taking these patterns as the initial centroids. The results are sent again to the central node and combined.

6.12.2 RACHET

RACHET (*Recursive Agglomeration of Clustering Hierarchies by Encircling Tactic*) [193] is a hierarchical clustering algorithm designed for distributed data sets in different nodes of a network. In each node, a *dendogram* or clustering tree is built. Following, all the dendograms are sent to a central node in form of summaries with statistical information, such as number of objects in each branch and average Euclidean distance. The central node combines the information from all the nodes to build the resulting tree.

6.13 Quantitative and qualitative data clustering algorithms

Clustering algorithms for quantitative and qualitative data offer an alternative to the common procedure of assigning an ordinal (a number) to a qualitative label and computing the quantitative distance (such as the Euclidean distance) between the values. This procedure can lead to erroneous results, as stated by Andritsos [174]. One of the main reasons is the resulting distance between qualitative labels, which, although being a quantitative measure, refers to a difference between two qualitative values, and therefore can lead to interpretation errors. Other reason that can be mentioned is that the result of the clustering process can group data objects that have nothing in common.

An example is provided by Guha et al. [183]. Given four objects with binary dimensions, as shown in (6.49), the objective is to cluster them in groups, taking into account that the value 0 means no value and, therefore, the clustering algorithm should group together objects with 1s in their dimensions or attributes.

$$X = \begin{pmatrix} 1\,1\,1\,0\,1\,0 \\ 0\,1\,1\,1\,1\,0 \\ 1\,0\,0\,1\,0\,0 \\ 0\,0\,0\,0\,0\,1 \end{pmatrix} \tag{6.49}$$

If the objects are grouped using the Euclidean distance, for instance applying the K-means algorithm, the result would be to group objects 1 and 2 in one cluster and 3 and 4 in another. Even though the first cluster might seem appropriate, it can be seen that the second cluster is formed by objects, 3 and 4, which have nothing in common.

Following, some algorithms are presented that perform clustering on quantitative and qualitative data by applying similarity measures that are different from the Euclidean distance and its most common variations.

6.13.1 K-modes and K-prototypes

K-modes algorithm [182] is an extension from K-means, where the mean or average distances are replaced by the *modes* or modal values of each cluster, and the similarity distance used is the *proximity*, which computes the number of dissimilarities between data attributes or dimensions (i.e. the overlap metric) (6.15).

The K-modes algorithm is designed for data with qualitative data only. For data with a combination of qualitative and quantitative information, the K-prototypes clustering algorithm is proposed [203]. In this algorithm, quantitative and qualitative dimensions are processed separately, and the total distance is a weighted sum of qualitative (or categorical) and quantitative distances between objects, as seen in (6.50), where γ is a weighting factor used to mixed both distances together.

$$s^T = s^{num} + \gamma s^{cat} \tag{6.50}$$

6.13.2 ROCK

ROCK (*RObust Clustering using linKs*) is a hierarchical clustering algorithm designed for qualitative data [184] [183], based on establishing *links* between neighbors, or objects that are near from each other, exceeding a proximity threshold value. The proximity distance can be any of the matching coefficients measure (see Section 6.3.4), being the threshold value θ defined by the user as an input parameter.

A *link* between two objects $\hat{x}$ and $\hat{y}$ is defined as the common number of other objects which are similar to them. The objective function (6.51) maximizes the sum of all the links between objects in the same cluster and minimizes the sum of links between objects in different clusters. The term C_i represents cluster i of size n_i.

$$J = \sum_{i=1}^{k} n_i \sum_{\hat{x},\hat{y} \in C_i} \frac{link(\hat{x},\hat{y})}{n_i^{1+2f(\theta)}} \tag{6.51}$$

The algorithm starts by considering all the objects as clusters, and then combines the objects at each iteration by the optimization of the objective function, following the value of a given *goodness* measure between clusters, which determines whether two clusters are fused together or not.

6.13.3 COOLCAT

COOLCAT [171] is a clustering algorithm for qualitative data based on the measure of entropy as a similarity measure (see Section 6.3.5). The objective of COOLCAT is to group object sin clusters achieving a minimization of the global entropy of the data. The objective function (6.52) is a summation of all the entropies from the resulting clusters, where D represents the total number of objects.

$$\bar{E}(\check{C}) = \sum_{k} \left(\frac{|C_k|}{|D|} E(C_k) \right) \tag{6.52}$$

The procedure of the algorithm is similar to the Max-min clustering (Section 6.6.2), but computing entropies instead of Euclidean distances. Starting from two initial clusters, the objects are iteratively grouped in clusters minimizing the global entropy, therefore the resulting clusters are formed by the most homogeneous objects found.

6.13.4 Mixed-type variable fuzzy c-means (MVFCM)

The MVFCM clustering algorithm [204] is an extension of the FCM that deals with quantitative and / or qualitative or categorical data by performing what is called *fuzzy clustering of fuzzy data*: the objects' attributes or dimensions are first fuzzified, and then the cluster analysis is performed. Different fuzzification approaches are defined as a function of the type of data (quantitative, qualitative or fuzzy), and a distance between each pair of dimensions is computed for the two objects, A and B. The final similarity measure is computed as a weighted sum (6.53), where α is the weight factor for each dimension.

$$D(A,B) = \sum_{k=1}^{d} \alpha_k D(A_k, B_k) \tag{6.53}$$

6.14 Visualization and dimensionality reduction

The visualization of clustering results is of critical importance, in order to provide to the expert a comprehensive information, that allow to obtain conclusions and extract knowledge from the data set. Graphics that display the clusters as volumetric figures, in 2D and 3D, and statistical summaries, are two of the most common visualization tools. Optionally, dimensionality reduction techniques can be applied on the data, in order to visualize the main characteristics of the results. According to Abonyi and Feil [90]:

> "One of the approaches applied to the visualization of high-dimensional spaces is the distance preserving mapping of the higher-dimensional space into a lower, usually two-dimensional map. Two general approaches for dimensionality reduction are: (i) feature extraction, transforming the existing features into a lower-dimensional space, and (ii) feature selection, selecting a subset of the existing features without a transformation."

In this sense, the following classification of dimensionality reduction and visualization techniques are gathered by Oliveira and Pedrycz [90]:

$$\text{Dimensionality reduction methods}\begin{cases}\text{Linear} & \begin{cases}\text{PCA / Factor Analysis}\\ \text{Discriminant analysis}\end{cases}\\ \\ \text{Non} - \text{linear} & \begin{cases}\text{M-PCA}\\ \text{Sammon}\\ \text{Self Organizing Maps (SOM)}\\ \text{Koontz}\end{cases}\end{cases}$$

The type of linear dimensionality reduction method used is influenced by the availability of categorical labels for the resulting classes or prototypes: if no categorical labels are available, the PCA or Principal Component Analysis is performed. In the case that categorical labels are available, the discriminant analysis is applied [205][87].

When linear methods cannot capture the data structure, non-linear methods are used. Self Organizing Maps (SOM) [84] and Sammon mapping [206] are two techniques used to visualize higher dimensional data on a 2D grid. While PCA tries to preserve variance of the data during the mapping, Sammon's mapping tries to preserve the interpattern distances [90], by minimizing the sum of the errors between the original high dimensional distances between objects (d^*_{ij}) and the reduced or projected ones (d_{ij}) (6.54).

$$E = \frac{1}{\sum_{i<j} d^*_{ij}} \sum_{i<j} \frac{(d_{ij} - d^*_{ij})^2}{d^*_{ij}} \tag{6.54}$$

Another option for visualization is the use of *data cubes*, very common in On-Line Analytical Processing (OLAP) systems [73]. A **data cube** is the graphical representation and modelling of multidimensional data sets by the use of different dimensions or *perspectives*, or "entities with respect to which an organization wants to keep records [73]". Each dimension has a table of facts or values for the different objects stored. The data cube is n-dimensional, and allows the precomputation and fast accessing and visualization of summarized data, by the selection of the main variable or perspective, and the ones that accompany the main one in the visualization. A cube is the visual representation of a 3D set; for more dimensions, different arrangements can be defined.

6.15 Cluster validity indices

Cluster validity indices have been defined in the literature to assign a numerical score to the results of clustering algorithms, since the "ideal" solution is not known (due to its unsupervised learning nature). Different cluster validity indices can be found, such as the Davies-Boulding (DB) [207] or Xie-Beni (XB) [208] indices. A selection of the most common is described in Annex D of the present document.

Despite their original use, as a measure of validity of the clusters obtained, these indices can also be used to compare the results of different clustering algorithms on the same data set [209][210]. This is the methodology applied in this thesis, along with a description of the expected results based on heuristics, as will be seen in the next chapters of this document.

6.16 Conclusions

Figure 6.2 summarizes the main classification of clustering algorithms and most representative algorithms of each kind. This result has also been summarized in form of table (Table 6.1), where the different similarity measures used are indicated.

From this state of the art, the next Chapter focuses on a more detailed review of clustering algorithms for time series data, which are the objective of this thesis. These studies have allowed to detect, so far, the following issues:

1. Visualization techniques have to be specifically addressed in more research and development works, especially regarding high dimensional data.
2. There are very few generic purpose clustering (and data mining) algorithms. Most of the algorithms described have been designed for specific kind of data. This fact does not pose, however, a significant drawback, but may add complexity to data mining and Big Data solutions.
3. The analysis of time series or dynamic data is approached as a separate field within data mining analysis. The inclusion of the time dimension in the classification and prediction results adds a new factor of complexity to the analysis and, therefore, to the algorithms developed for this purpose, as will be seen in the following Chapters.

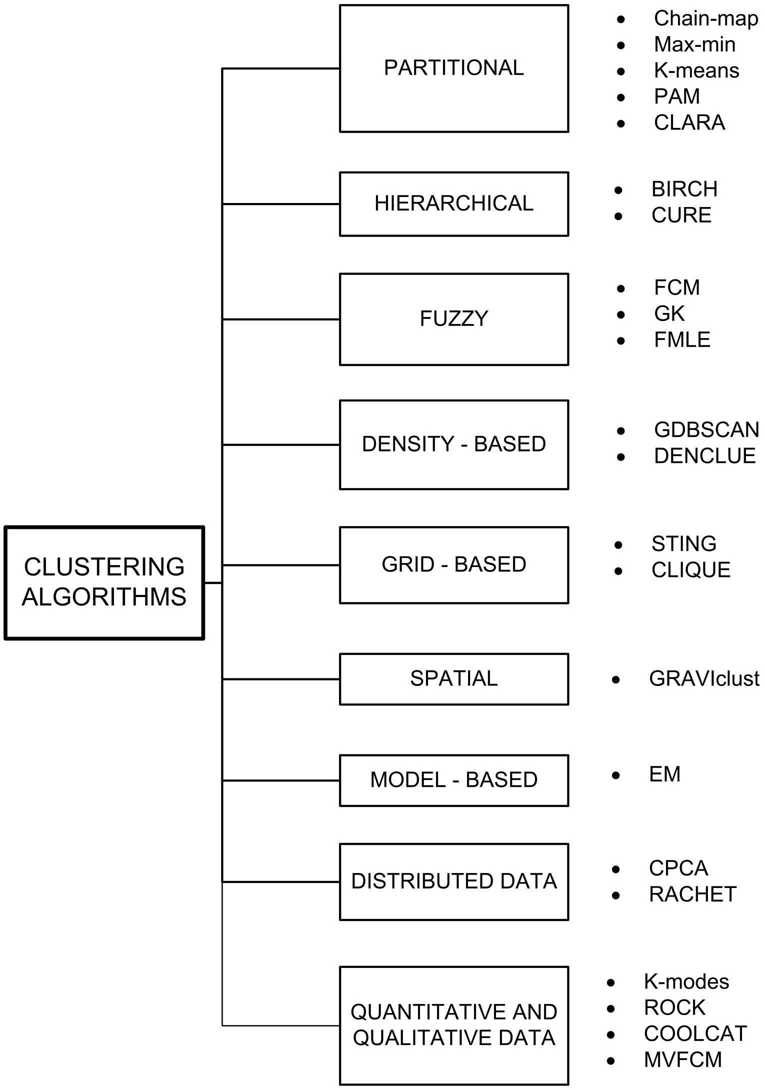


Fig. 6.2. Classification of clustering algorithms.

Algorithm	**Classification**	**Similarity measure**
Chain-map	Partitional	Euclidean distance
Max-min	Partitional	Euclidean distance
K-means	Partitional	Euclidean distance
PAM	Partitional	Median
CLARA	Partitional	Median
BIRCH	Hierarchical	Euclidean distance
CURE	Hierarchical	Euclidean distance
FCM	Fuzzy	Euclidean distance
GK	Fuzzy	Euclidean distance with norm
FMLE	Fuzzy	Euclidean distance with exponential norm
GDBSCAN	Density-based	Density relation
DENCLUE	Density-based	Function of influence
STING	Grid-based	Statistical information
CLIQUE	Grid-based	Density in hyper cubes
GRAVIclust	Spatial	Euclidean distance
CPCA	Distributed	PCA
RACHET	Distributed	Euclidean distance
K-modes and K-prototypes	Quantitative and qualitative	Overlap metric
ROCK	Quantitative and qualitative	Number of links
COOLCAT	Quantitative and qualitative	Entropy
MVFCM	Quantitative and qualitative	Fuzzy inference

Table 6.1. Clustering algorithms by main characteristic and similarity measure used.

7

State of the art on clustering algorithms for time series data

7.1 Introduction

Concerning the dynamic clustering, the objective of performing dynamic segmentation on a time series database is to obtain dynamic centroids, i.e., patterns that represent a number of objects whose features may vary with time, but remain similar enough to pertain to the same cluster. The objective, therefore, is to obtain a set of patterns that depict the full evolution of the data through time.

Liao [211] highlights the following issues that must be considered when performing clustering on time series data:

- The difference in length in the data objects to be compared.
- The errors in the data, or noise.
- The sampling rate and if this sampling is even or not.

The clustering technique used must deal with objects that may have time-series discontinuities, i.e., gaps between time measures, which can vary among the different objects in the database. This is significantly true for daily load curves: measures from the meters can be unavailable for some days, due to unknown reasons, such as a malfunction or maintenance. These discontinuities do not happen with the same frequency, nor at the same days for all the clients, therefore occasional, unpredicted discontinuities appear at the data, which does not necessarily mean that a client has been removed from the database, simply that there are not available measures. Clients in this situation should be kept at their last state within a cluster, until a new measure is analyzed.

7.2 Distance or similarity measures for time series data

A review of the most used similarity measures for clustering has been described in Chapter 6. However, there are other similarity measures and distances which have been used in the literature to compare time series data (although sometimes this was not their original purpose). Serrà and Arcos [212], for instance, have performed an empirical evaluation of seven groups of similarity measures applied on 45 time series data sets. Their conclusion is that Euclidean distance and Dynamic Time Warping (DTW) are good similarity measures for generic time series, but the TWED (Time Warp Edit Distance), which is a combination of DTW with Edit Distance, outperforms these two, becoming, according to the authors, the best similarity measure for time series. These similarity metrics, and others, are described next.

7.2.1 Dynamic Time Warping (DTW)

Dynamic Time Warping or DTW [213] [214] [215] is a well-known technique used to compare time series which can be unequal in length and/or in the data sampling rates. The Euclidean distance and similar metrics compare the time series sequentially point by point. As a result, the computed distance can be high, even when the two series are similar in shape, but delayed in time, or they have been sampled at different rates.

The DTW is defined as the optimal alignment of two temporal series. This optimal alignment is the optimal *warping path* between the two series. Given two time series, R and S, of length n and m respectively, the matrix $d(nxm)$ of distances is defined. Each element $d(i, j)$ in this matrix corresponds to the distance between R_i and S_j, typically the Euclidean distance, although other measures could be used. The warping path $W = w_1, w_2, ...w_k$ is any alignment between R and S, subject to the following constraints or conditions, described by Capitani and Ciaccia [214]:

- **Boundary conditions**: $w_1 = [1, 1]$ and $w_k = [n, m]$. This condition sets the beginning and end of the path at the vertices of the distance matrix.
- **Continuity**: given $w_{k-1} = [i_{k-1}, j_{k-1}]$ and $w_k = [i_k, j_k]$, then $i_k - i_{k-1} \leq 1$ and $j_k - j_{k-1} \leq 1$. This condition ensures that the elements of the warping path are adjacent.
- **Monotonicity**: given $w_{k-1} = [i_{k-1}, j_{k-1}]$ and $w_k = [i_k, j_k]$, then $i_k - i_{k-1} \geq 0$ and $j_k - j_{k-1} \geq 0$, with at least one strict inequality. This forces W to progress over time.

The optimal DTW path p is the warping path with the minimal cost or sum of squared distances for the complete path, i.e., the *optimal* warping path (7.1).

$$DTW(R, S)^2 = min_W \{ \sum_{i_k, j_k \epsilon W} d(i_k, j_k)^2 \} \tag{7.1}$$

The DTW distance can be recursively computed using a dynamic programming approach that fills the cells of a *cumulative distance matrix D* [214] using the recurrence relation displayed in (7.2). The total DTW distance is obtained as $DTW(R, S) = \sqrt{D[n, m]}$.

$$\begin{cases} D[i,1] = \sum_{k=1}^{i} d(r_k, s_1) \\ D[1,j] = \sum_{k=1}^{j} d(r_1, s_k) \\ D[i,j] = d[i,j] + min \begin{cases} D[i-1,j-1] \\ D[i-1,j] \\ D[i,j-1] \end{cases} \end{cases} \tag{7.2}$$

Once the D matrix has been computed, the warping path points $p = (p_1, ...p_L)$ are obtained in reverse order, starting from the last point of the path (p_L) going back to p_1 following the procedure in Eq. (7.3)

$$\begin{aligned} p_L &= (N, M) \\ p_{l-1} &= \begin{cases} (1, m-1) & ifn = 1 \\ (n-1, 1) & ifm = 1 \\ argmin \begin{cases} D[n-1, m-1] \\ D[n-1, m] \\ D[n, m-1] \end{cases} & otherwise \end{cases} \end{aligned} \tag{7.3}$$

Warping paths can be subject to some global constraints, in order to prevent pathological alignments. The most commonly used of such constraints is the Sakoe - Chiba band [216] of width b, that forces warping paths to deviate no more than b steps from the matrix diagonal [214].

The previous conditions can also be relaxed in some applications. One method is to relax the initial and/or final warping points, allowing the DTW to start and end in different points rather than the vertices of the d matrix. Other method is to relax the adjacency condition, allowing an adjacency margin of 2, 3 or more points.

Different similarity measures based on DTW have been developed by different authors in the last years. One of them, for instance, is the CBDTW (Correlation - based DTW), proposed by Bankó and Abonyi [217], which is a combination of DTW and PCA. Another example is the global averaging measure for time series, based on DTW, proposed by Petitjean and Gançarski [218]. The DTW distance is also used for template matching of time series by Niennattrakul et al. [219].

7.2.2 Hausdorff distance

The Hausdorff distance was described by Felix Hausdorff in his foundational book on Set theory [220]. The definition of this distance is the following [221]: given an operator d of distance, such as the Euclidean distance, the distance from a generic point x to a closed subset A, both x and A belonging to the p-dimensional subset of the closed subsets in $\Re$, is defined as the minimum of the distances of x to all the points that belong to A (7.4).

$$d(x, A) = min_{\tilde{a} \in A}(d(x, \tilde{a})) \tag{7.4}$$

The Hausdorff metric between two non-empty closed subsets, A and B, is defined as the maximum of all possible distances $d(\tilde{a}, B)$, as seen in (7.5). Since this metric is not necessarily symmetric, the Hausdorff distance $d_H(A, B)$ between the two subsets is obtained as the maximum of their two Hausdorff metrics, $h(A, B)$ and $h(B, A)$ (7.6).

$$h(A, B) = max_{\tilde{a} \in A}(d(\tilde{a}, B)) \tag{7.5}$$

$$d_H(A, B) = max(h(A, B), h(B, A)) \tag{7.6}$$

The Hausdorff distance is mostly used in clustering to compare similarity between geometrical objects [222], time series data [223], trajectories [224] and interval or symbolic data [225][226][227].

7.2.3 Edit Distance on Real sequences (EDR)

Developed by Chen et al. [228], the Edit Distance on Real sequences is specifically designed to compute similarity between time series with two or more dimensions, with noise and outliers in the data, such as moving object trajectories defined by spatial coordinates and the time stamp. The Edit Distance is a similarity measure between two strings, computed as the number of *insert*, *delete*, or *replace* operations that are needed to convert one string in the other. The same definition is applied to compare sequences of spatial coordinates from two moving objects: the EDR is defined as the number of *insert*, *delete*, or *replace* operations that are needed to change one trajectory *R*, of length *n*, into another trajectory *S*, of length *m*. The EDR computation is described in Eq. (7.7)

$$EDR(R, S) = \begin{cases} n & if m = 0 \\ m & if n = 0 \\ min \begin{cases} EDR(Rest(R), Rest(S)) + subcost \\ EDR(Rest(R), S) + 1 \\ EDR(R, Rest(S)) + 1 \end{cases} & otherwise \end{cases} \tag{7.7}$$

where $subcost = 0$ if $match(r_i, s_j) = true$ and $subcost = 1$ otherwise. The two sequences match if their absolute difference falls below a specified threshold for each dimension. This matching operation minimizes the effect of noise and outliers in the computation, since the distance is quantized.

7.2.4 Time Warp Edit Distance (TWED)

Presented by Marteau [229], the TWED is based on the EDR, substituting the *insert*, *delete* and *match* operations by the *delete A*, *delete B* and *match* operations, which are described as follows:

- The *delete* A operation consists in removing a point in the sequence A and placing it on the previous sample, with a total cost associated computed as the length difference between the removed sample and the previous one.
- The *delete* B operation is the same process but applied on sequence B.
- The *match* operation is defined as the process of moving one segment of A and placing it on the equivalent segment on B. The total cost associated to this operation can be computed as the sum of the length differences between segment vertices.

7.3 Classification of clustering algorithms for time series data

In the same way as the classification of clustering algorithms, there are also different criteria when classifying clustering algorithms for time series data. The three main approaches or criteria are the following:

- According to the dynamic nature of data and the clustering algorithms.
- According to the way the time series data is processed.
- According to the number of features or characteristics of the data.

These three approaches are described next.

7.3.1 Classification according to the dynamic nature of data and the clustering algorithms

The dynamic data is formed by objects with dynamic features, represented by feature trajectories in time. The static data, on the contrary, is represented by objects with a features vector of values captured at a given time. The dynamic clustering algorithm performs segmentation, therefore, on a dynamic data set, augmenting the possibilities of clustering beyond a static "picture" of clusters at a given time or period. Table 7.1, extracted from Weber [90], classifies the cluster analysis in four types or categories, according to the dynamic nature of the data and the clusters:

1. The data, although is time series, is treated as static and the clustering process is also static. This is the case when the static clustering is applied, and corresponds to the example seen in Figure 5.2.
2. The data, although is time series, is treated as static. The number of clusters, however, is not fixed, and may vary at each new computation. This is the case when the database is partitioned in batches and processed sequentially. For each batch or sequence, the number of clusters may vary, indicating a variation in the data values or trends. This clustering process implies, in case a relationship among clusters is to be established through time, the need of a pattern matching or recognition system, able to associate the new clusters with the ones from previous steps[230]. In this

system, issues such as clusters formation, collapse, split or fusion must be considered.

3. The data is treated as dynamic, evolving through time, therefore the time series become trajectories of the different data features or dimensions through time. The number of clusters is fixed. The objects are clustered taking into account the evolution or trajectories of the object features and, therefore, the resulting centroids or patterns are defined by feature trajectories that are representative of the data evolution in the different resulting clusters. The present thesis approaches this type of dynamic clustering analysis.
4. The data is also treated as dynamic, as in type 3, becoming feature trajectories that evolve through time, and the number of clusters varies dynamically at each iteration. Clusters and patterns can, therefore, as in type 2, merge or split.

Table 7.1. Types of clustering according to dynamic nature of data and classes

No.	**Data objects**	**Classes**	**Type of clustering**
1	Static	Static	Classical, static clustering
2	Static	Dynamic	Dynamic. Prototypes and classes are updated at each cycle
3	Dynamic	Static	Dynamic. Prototypes represented by feature trajectories. Fixed classes
4	Dynamic	Dynamic	Dynamic. Prototypes represented by feature trajectories, classes updated iteratively

Dynamic clustering in general and, more specifically, Types 3 and 4 in the Table, consider the analysis of the evolution of the features or characteristics of the data object, therefore evaluating and clustering feature *trajectories*, not real-valued vectors [90]. This type of clustering is also called *temporal* data mining, and is the objective of the present thesis. The dynamic clustering is applied on time series data from daily load profiles of energy consumption. The objective is to evaluate the dynamic evolution through the days of the 24 hours or features of a daily load profile, therefore the 24 feature trajectories are to be evaluated.

The techniques studied and developed in this thesis correspond to the Type 3 dynamic clustering (dynamic data, but a fixed number of classes or clusters). Type 4 dynamic clustering is considered as a further step of development from the techniques developed for Type 3; this development, however, is out of the scope of the present thesis. Its approach and possible steps for development are described in the Chapter 15 of future works.

7.3.2 Classification according to the way the time series data is processed

Regarding the clustering of time series data, Liao [211] makes a differentiation of clustering types for time series data based on three main approaches: clustering on the **raw** time series data, clustering on a **feature-based** transformation of the time series data, and clustering on a **model-based** transformation of the time series data. The raw data clustering takes the data as they are; however feature-based and model-based clustering algorithms perform an initial step where the data are transformed. In the case of the features, by obtaining a reduced dimensionality version of the original data. In the case of the models, by obtaining different representations of the data which can be handled by the algorithm. The three of them are described in the following Sections of this Chapter.

7.3.3 Classification according to the number of features or characteristics of the data

Time-series or dynamic data clustering can be applied on a one dimensional or feature data series (such as ECG or electrocardiogram), or on n-dimensional data, such as the daily vector of 24 hour measures of energy consumption from a household, which is the objective of this thesis. Based on this main difference, the clustering algorithms applied are different, as will be seen in the following Sections.

7.4 State of the art on clustering algorithms for time series data

The main criterion selected to classify and present the different algorithms is according to the way the data is processed. However, each algorithm described is also classified to the other two criteria previously defined. The following information has been obtained mainly from surveys of clustering algorithms for time series data found in the literature, such as the ones by Liao [211]; Rani and Sikka [231]; Esling and Agon [232]; and Fu [233].

7.4.1 Raw time series data clustering

The clustering techniques or algorithms that perform the clustering on the raw time series data set are commonly modifications of the static clustering techniques, where the distance or similarity function is replaced by another measure suitable for time series data. In this case, the time series data can be of equal or unequal length.

In the case of dynamic clustering, or trajectories clustering with fixed classes (type 3 in Table 7.1), Weber [90] describes the algorithm called **Functional Fuzzy C-means or FFCM**. This algorithm is presented as a generalization of the static fuzzy c-means or FCM [180]. The distance d between two trajectories, f and g, is computed in three steps:

1. The fuzzy Membership Function (MF) "approximately zero" is defined ($\mu(f(x))$). This is a symmetric MF, with the maximum membership (1) at zero value, that can have any shape. Weber defines a gaussian MF, which rapidly decreases as the value of x increases.
2. Compute the *similarity* function $s(f, g)$ between two trajectories f and g as the fuzzification of the difference between the two trajectories, i.e., $s(f, g) = \mu(f(x) - g(x))$.
3. Compute the distance between the two trajectories f and g as the inverse of the similarity: $d(f, g) = (1/s(f, g)) - 1$.

In particular, The FFCM algorithm presents a modified calculation of the membership value of object i to cluster j at each iteration, indicated in Eq. 7.8.

$$\mu_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{d(X_i, V_j)}{d(X_i, V_k)} \right)^{\frac{2}{m-1}}} \tag{7.8}$$

Where X_i represents the feature vector of object i; Vj and Vk are the class centers of the classes j and k, respectively; c is the number of classes; and m is a parameter determining the degree of fuzziness of the generated clusters (usually, with a value higher than 1). The d operator represents the distance function based on fuzzy inference. In the same description of the FFCM algorithm, Weber also indicates that:

1. If the trajectories are defined by sets of real values, a numerical distance, such as the Euclidean distance, can be applied.
2. The method described can be applied to any other data mining technique.

Kosmelj and Batagelj [234] apply the **Generalized Ward criterion** [235] to develop a *relocation clustering procedure* for time series objects, performed at each time step iteration t. From an initial partition, the objects are reassigned to the clusters applying any pre-defined dissimilarity metric, with the objective to minimize the Ward function ($p(C)$, being p any function that reflects the adequacy of the clusters set assignment to objects). This process is repeated between features for each time step, having therefore a matrix of dissimilarities between features for all the time steps of a given period T. The resulting dissimilarity value between the two objects is computed as a weighted sum of all the dissimilarities in the matrix. This cross-sectional approach is also applied by Liao et al. [236], by means of different clustering techniques, such as K-means, fuzzy c-means and genetic clustering with the objective to form a discrete number of battle states from multivariate battle simulation time series data of unequal length.

Meesrikamolkul et al. [237] propose a modified K-means clustering algorithm for time series, the **Shape-Based Clustering**, where the Euclidean distance is replaced by a Dynamic Time Warping (DTW) alignment between time series. Since the computational cost of the DTW is high, the authors present a method to obtain the representative element from each cluster called Ranking Shape-based Template Matching Framework (RSTMF), which can average a group of time series effectively but take as much as 400 times less computational time than that of a previous method called STMF (Shape-based Template Matching Framework).

The same approach is presented by Izakian et al. [238]. The "classical" static FCM and Fuzzy-C-Medoids clustering algorithms [239], and a third hybrid technique that uses both, are modified, and a **DTW** distance is used as the similarity measure to cluster one-dimensional time series data. The authors indicate that the use of the DTW is appropriate only in those cases when the similarity is sought between the *shapes* of the data objects, and depending on the nature of the data themselves.

The same authors present a clustering algorithm for clustering spatiotemporal data [240], where the Euclidean distance is replaced by an **"augmented"** distance, which is the weighted sum of two Euclidean distances: the comparison of the spatial components and the comparison of the temporal features. This modification is later extended by Izakian and Pedrycz [241] to n features or dimensions with the concept of **blocks** or groups of similar features, computing a weighted sum of Euclidean distances where the different weights are obtained by Particle Swarm Optimization (**PSO**) [242].

Yin et al. [243] perform a hierarchical clustering of traffic flow time series data by applying an **Encoded-Bitmap based similarity** distance. This distance, defined in [244], assigns a time series u to one of two clusters, A or B, depending on the value of the *linkage difference* of u to A and B. The linkage difference (Eq. (7.9)) is computed as the difference of the average linkages of u to A and B. The average linkage is computed as in Eq. (7.10), where $W(A,u)$ is the summation of all the distances among the A and u elements (Eq. (7.11).

$$\Delta D(u,A,B) = D(A,u) - D(B,u) \tag{7.9}$$

$$D(A,u) = \frac{W(A,u)}{|A|} \tag{7.10}$$

$$W(A,u) = \sum_{a_i \epsilon A, u_j \epsilon u} |a_i - u_j| \tag{7.11}$$

Kakizawa et al. [245] make use of two similarity distances, the **Kullback-Leibler** and the **Chernoff** measure, to classify among different patterns in multivariate time series. Shumway [205] applies also Kullback-Leibler divergence as a discriminant measure to differentiate explosions from earthquakes in the spectral time series for both events.

Keogh et al. [246] propose a hierarchical clustering of time series based on a parameter-free dissimilarity distance, the **Compression-based Dissimilarity Measure** or CDM, obtained from the sizes of the compressed strings or time series ($C(x)$) after applying a compression engine. The formula for the CDM is seen in Eq. (7.12), where $C(xy)$ is the size of compressed string x by first training the compression on string y.

$$CDM(x, y) = \frac{C(xy)}{C(x) + C(y)} \tag{7.12}$$

Möller-Levet et al. [247] define the *Short Time Series* (STS) distance, as a similarity measure able to capture differences in shapes, and develop the **fuzzy short time series** (FSTS) algorithm as a modification of the standard fuzzy c-means algorithm (FCM), with the STS distance. This distance is defined as the squared difference between the slopes of two intervals of the time series being compared, therefore it could also be considered a measure of the difference in the two time series changing rate.

Kumar et al. [248] develop a specific distance to compare and cluster time series data with noise, represented by **seasonalities**. Seasonality is defined as the normalized underlying demand of a group of similar merchandize as a function of time of the year after taking into account other factors that impact sales such as discounts, inventory, promotions and random effects. The data samples in each seasonality are represented by a Gaussian noise, with average value and standard deviation. The resulting distance is equivalent to a weighted Euclidean distance: values with lower standard deviation have a higher weight in the computation.

Golay et al. [249] present a modified FCM clustering where the distance metric used is the **Pearson correlation** between two fMRI (functional magnetic resonance image) temporal series.

Other authors have developed specific algorithms or modifications on static clustering techniques for time series clustering. Wissmüller et al.[250], for instance, propose to apply free energy **Vector Quantization** (VQ) to obtain clusters from magnetic resonance time-series images. The algorithm, similar to a Self Organizing Map, obtains representative patterns of the pixels data gray scales evolution, called the Pixel Time Course (PTC). These corresponding Codebook Vectors (CVs) represent prototypical PTCs sharing similar temporal characteristics. As a result, each PTC can be assigned to a specific CV according to a minimal distance criterion.

Liao [251] proposes a **two-step clustering** method: "the first step applies the k-means or fuzzy c-means clustering algorithm to time stripped data in order to convert multivariate real-valued time series into univariate discrete-valued time series. The converted variable is interpreted as state variable process. The second step employs the k-means or FCM algorithm again to group the converted univariate time series, expressed as transition probability matrices, into a number of clusters. The Euclidean distance is used in the first step, whereas various distance measures such as the Dynamic Time Warping, or the symmetric version of Kullback–Liebler distance, are employed in the second step [211]."

Policker and Geva [252] present a model for time series based on **temporal clustering**. The model generator is computed based on the clustering of past values of the time series in segments or time clusters. The model is then built from the sum of the resulting clusters.

Ji et al. [253] present also a modification of the FCM algorithm for time series data, defining the **key points** in a time series as the group of points that describe, implicitly, how a series changes in a certain time.These points usually represent specific events, such as the start or the end in a tendency of upward or downward, the peak or the bottom of the series. The key points reveal the dynamic aspect of a given time series. The authors present a study of the dynamic switching of the time series among a fixed number of clusters, over time, based on the membership coefficient at each step.

van Wijk and van Selow [254] perform **hierarchical clustering** to identify patterns and trends on multiple time scales (days, weeks, seasons) on univariate time series data. The analysis is performed on raw time series data from load curves of electric energy consumption from research facility. The average Euclidean distance is used to compute similarities among load curves and merge day into clusters in a bottom-up approach. The analysis is performed to cluster similar days for a single facility.

Computational Verb Theory [255] is applied by Zhang and Yang [256] to cluster stock market series data. Three computational verbs are defined: increase, decrease and stay. The distances among the series data and these three verbs are computed and clustered.

7.4.2 Feature-based time series data clustering

As stated by Fu [233], one of the objectives of time series processing is the transformation of the data to a reduced dimensionality or to a summary or reduced amount of representative values, prior to the clustering process. Guo et al. [257], for instance, perform a dimensionality reduction of the raw time series data by applying the **Independent Component Analysis** (ICA), a statistical method similar to Principal Component Analysis (PCA) [258] and Factor Analysis. This similarity measure is then used in a modified K-means algorithm.

With the objective of classification, Fulcher and Jones [259] develop **summaries** from the analysis of the data objects features applying a suite of techniques: correlation, entropy, etc, to later perform a selection of the most representative features for the classification.

Irpino et al. [260] also compute summaries of the time series data by obtaining histogram representations of each feature or dimension. The data is then compared by applying a specific similarity measure for histograms, based on the **Wasserstein distance** [261]. the Wasserstein distance between two univariate distribution functions can be expressed as in Eq. 7.13.

$$d_m(y_i, y_j) = \left[\int_0^1 |F_i^{-1}(t) - F_j^{-1}(t)|^m dt\right]^{\frac{1}{m}} \tag{7.13}$$

Where F_i and F_j are the cumulative distribution functions (cdfs) associated to the ith and jth histogram and the F_i^{-1} and F_j^{-1} are the corresponding quantile functions (qfs).

The **autocorrelation** function coefficients are used by D'Urso and Maharaj [262] to determine similarity between time series data, computing the Euclidean distance between these coefficients. A modified fuzzy c-means is the applied iteratively to dynamically assign the cluster membership of the time series data.

Wilpon and Rabiner [263] obtain clusters of spoken sentences by the combination of word patterns. The clusters evolve combining different patterns. Different techniques to obtain the cluster centers are presented, such as minmax computation or averaging. A modified k-means algorithm is presented, where the similarity between time series is computed applying the **log likelihood** distance proposed by Itakura [264], based on autocorrelation coefficients.

Goutte et al. [265] perform feature-based clustering on fMRI data, based on the **cross-correlation** function, presented in a previous work in [266]. The raw data is reduced in size by a selection process of only relevant data from the magnetic resonance that can be representative to form the different clusters. Cross-correlation is applied between the fMRI data and a function defined as the "excitation paradigm". The resulting correlation indices are used for the clustering process, along with features or indices such as the *strength* of activation or the response delay.

Fu et al. [267] compress the input patterns by a **perceptually important point** (PIP) identification algorithm, replacing the original data by a group of PIPs, thus reducing dimensionality. Following, the authors identify patterns making use of SOM applied on data sequences segmented by a sliding window, with the objective to mine frequent patterns in the temporal series.

Owsley et al. [268] perform clustering on vibration sensor time measures from industrial machine tools, to serve as a predictive maintenance diagnosis. First, the time transients are identified from the raw data, by comparison with normal use steady state patterns. Then, this information is used to compute **Self Organizing Feature Maps** (SOFM).

Vlachos et al. [269] apply dimensionality reduction on the time series data making use of the **Haar wavelet** transform. The authors present a modified incremental k-means clustering, where the Haar wavelet is applied on the data, and the centers from the clustering result are used as inputs for another clustering at a finer resolution level, proceeding this way until the termination criterion is met. The same approach, although with different techniques, is presented by Aghabozorgi et al. [270]: an **incremental clustering** for time series is presented, based on the identification of the Longest Common Subpaths (LCSS) from dimensionality reduced time series picked randomly. Then, a fuzzy clustering, such as the FCM is applied, to dynamically create new classes, modify existing ones, and/or remove unchanged clusters.

Li and Guo [271] propose the **Piecewise Cloud Approximation** function (PWCA) to reduce the dimensionality of time series. The time series is first segmented in frames, and each frame is transformed into a cloud model representation, defined by three variables: Ex (the Expected value, or mean of the segment), En, which is the Entropy, or uncertainty in the measurements, and Hex, which is a hyper-entropy, or an uncertainty measurement of the entropy En, and reflects the thickness of the cloud.

Lin et al. [272] present the multi-solution piecewise aggregate approximation (**MPAA**) transform. The technique is an iterative segmentation of a time series in lower dimension sequences of the average values. In each level of dimensionality reduction, a k-means clustering is applied and the intra-cluster error is compared with the Hoeffding bound [273], as a stopping condition. The clustering algorithm is then used in this work for data stream clustering, by applying a nearest neighbor assignment of new incoming sequences to existing clusters, and then re-computing the MPAA clustering process only on the affected cluster.

Chandrakala and Sekhar [93] propose to cluster time series by a **kernel density-based clustering**. The kernel based clustering method comprises two parts: a nonlinear transformation that maps data points from a low-dimensional input space to a high dimensional feature space induced by an inner product kernel or a Mercer kernel, and a learning algorithm to find linear patterns in that feature space. The use of a density-based clustering algorithm, such as DBSCAN (Density Based Spatial Clustering of Applications with Noise) [274] avoids the need of defining the number of clusters to be found as an input parameter.

Lai et al. [275] propose a clustering process in **two levels of granularity** analysis. In a first level, the data is processed and summarized by applying the **Symbolic Aggregate Approximation (SAX)** representation, developed by Keogh et al. [276], in which a time series is summarized by a list of “breakpoints”, or statistically representative data points from the time series. Following, these summaries are clustered, by applying the density-based CAST clustering algorithm [277]. On the resulting clusters, a second CAST clustering is then applied, making use of the DTW similarity to compare sequences of unequal length. A similar approach is presented in [278] to cluster vehicle trajectories in two steps: first by start-destination coordinates, and then applying DTW to cluster the entire trajectories by similarity.

Lin and Li [279] make use of the SAX representation to compute summaries of the time series in form of **histograms** of SAX values, inspired by the “bag-of-words” summarization of text documents [280]. The resulting summaries are then clustered by a hierarchical clustering algorithm.

7.4.3 Model-based time series data clustering

Li and Prakash [281] perform clustering of trajectories by the development of ***complex-valued* linear dynamical systems** (CLDS) to handle time lags and joint dynamics among trajectories. The system parameters are estimated following a procedure similar to the Expectation-Maximization (EM) algorithm.

Xiong and Yeung [282] perform a classification of time series data into patterns using **mixtures** of autoregressive moving average (ARMA) models. Both mixing coefficients and model parameters are computed applying the Expectation-Maximization (EM) algorithm, by defining the probabilities of a time series being a mixture of a number of models.

Bagnall and Janacek [283] compare also the coefficients from ARMA models of the time series data, but transforming first the time series into a discretised **Clipped data** representation, where the new values are equal to 1 if the original time series is above the median, and 0 otherwise. The authors indicate that this discretisation removes outliers from the data, thus resulting in better clusters.

Baragona [284] applies clustering based on distance computed from **cross-correlation** indices among the time series data. Meta-heuristic methods are applied for clustering: Simulated Annealing, Tabu Search and Genetic algorithms.

Maharaj [285] fits the two time series to be compared to **Auto-Regressive** (AR) models of order k, and then compares the difference between the resulting AR models. An agglomerative hierarchical clustering algorithm is applied on the AR estimates to group them by similarity.

Vilar et al. [286] make use of **sieve bootstraps**, or non-parametric models of unidimensional time series, to obtain forecasts of the data up to a certain instant of time $T + b$, given by the problem objective. Then, the estimated density forecasts are hierarchically clustered based on two specific distance measures: L^1 and L^2, which compare the distance between pairs of estimated density functions $f(i)_{X_{T+b}}$ (see Eq. (7.14)).

$$D_{1,ij} = \int |f(i)_{X_{T+b}} - f(j)_{X_{T+b}}| dx \tag{7.14}$$

Zhang et al. [287] model gene expressions as **splines**, thus reducing the size of the time series data and filtering noise and loss of information, and then apply an **energy based similarity measure**, called SimilB, described in [288]. This distance measures the non-linear interaction between two real time functions taking into account their first and second derivatives. Its formula is described in Eqs. (7.15) and (7.16), given two curves $f(t)$ and $g(t)$.

$$SimilB(f,g) = \frac{\sqrt{2}\int \psi_B(f,g)dt}{\int \sqrt{\psi_B(f,f)^2 + \psi_B(g,g)^2} dt} \tag{7.15}$$

$$\psi_B(f,g) = \frac{df(t)}{dt}\frac{dg(t)}{dt} - \frac{1}{2}\left(f(t)\frac{d^2g(t)}{dt^2} + g(t)\frac{d^2f(t)}{dt^2}\right) \tag{7.16}$$

A larger value indicates more similarity between the time series, which reaches its largest value of 1 when $f(t) = g(t)$.

Ramoni et al. [149] present a Bayesian method for clustering dynamic processes. The method models dynamics as **Markov chains** and then applies an agglomerative clustering procedure to discover the most probable set of clusters capturing different dynamics. To increase efficiency, the method uses an entropy-based heuristic search strategy. The authors have also developed a multivariate clustering dynamics [289], applying an average of the Kullback-Leibler distance between transition states to compute similarity.

Savvides et al. [290] propose to analyse the similarity between time series based on their ***cepstral* coefficients**, obtained from the spectral density functions of the equivalent ARMA models of two time series. Kalpakis et al. [291] perform clustering on time series by computing the similarity between parameters of the ARIMA models obtained of the data making use of a Linear Predictive Coding (LPC) cepstrum distance. The cepstrum is defined as the inverse Fourier transform of the short-time logarithmic amplitude spectrum. The dissimilarity between time series is measured using the Euclidean distance between their LPC cepstral coefficients. The Partitioning Around Medoids (PAM) clustering algorithm is used. The results show that cepstral coefficients have a high discriminatory power to separate time series generated by different models. A similar approach is presented by Maharaj and D'Urso [292], where the cepstral coefficients are obtained and the fuzzy c-means is applied.

Corduas [293] performs similarity between time series data by computing the **Mahalanobis distance** between regression coefficients and the Euclidean distance between Autoregressive weights. This metric is applied on time series hydrology observations from different locations.

Li and Biswas [294] apply **Hidden Markov models** (HMM) for clustering temporal data. The HMM clustering method proposed by the authors can be summarized in terms of four levels of nested searches:

1. The number of clusters in a partition.
2. The structure for a given partition size.
3. The HMM structure for each cluster, and
4. The parameters for each HMM structure.

The process goes from the inner level of search to the outer levels, following a maximum likelihood approach to obtain the different models.

Oates et al. [295] also model each cluster as a HMM, able to generate the temporal sequences that belong to that cluster; the first partition of the data (and also the number of clusters) is obtained by applying a hierarchical clustering algorithm which applies DTW as the dissimilarity distance. The result from this partition is used as the initialization to train the HMM.

Tran and Wagner [296] develop a fuzzy c-means clustering based **normalization** method that is used to improve the speaker's verification, by comparing an original speech record with another one that might be of unequal length. The resulting score determines the acceptance or rejection of the speech.

7.5 Conclusions

Tables 7.2, 7.3 and 7.4 summarize the present state of the art on dynamic or time series clustering algorithms described in this Chapter. From these tables, the following conclusions can be drawn:

- Most of the algorithms are designed for one - dimensional time series, such as a traffic flow, stock market data, or measures from medical equipment, such as fMRI (functional Magnetic Resonance Image) or ECGM temporal data.
- Most of the algorithms are designed for time series, or dynamic data, but a fixed number of classes or clusters (i.e. Type 3 from Table 7.1). Type 4 algorithms (dynamic data and dynamic classes) are also present, mainly in feature-based and model-based algorithms.

As will be seen in the following Chapters, the load profile time series data from energy consumption are formed by daily temporal data with typical values of either 24 or 96 dimensions (24 hour energy measures or 96 quarter-hour energy measures per day). Therefore, the development presented in this thesis is designed for energy consumption n-dimensional data, where all the dimensions have the same magnitude (energy – kWh) and, as an innovation perspective, they are processed as a daily time series (for a week, or a month, or a year...).

The development made, as will be explained, is a flexible framework that can make use of different clustering techniques presented in the state of the art, extended to process time series data of n dimensions. The resulting clustering process is of Type 3, applied on the raw data on energy consumption with 24 features or dimensions.

The resulting similarity measure that will be described can be seen as an "augmented" distance, in the sense described by Izakian et al. [240] or Izakian and Pedrycz [241], since the similarity in static or Type 1 clustering is augmented to process time instants in n features or dimensions. It can also be seen as a development based on the description of the membership function of a time series object to a class made by Weber in the FFCM algorithm [90]. The distance function operator d is replaced by specific distance functions able to compare two time series and yield a value of similarity. None of the previous works presented, however, have been developed to analyze and visualize the resulting clusters in the form of n or, in this case, 24 dimensions of dynamic data objects. The development presented in this thesis takes as an initial

Algorithm	**Way to process**	**Type of data**	**No. of dimensions**	**Technique applied**
FFCM	Raw	3	1	Fuzzy sets
Kosmelj and Batagelj, 1990	Raw	3	n	Generalized Ward criterion
Liao et al., 2002	Raw	3	n	Euclidean distance
Shape-Based Clustering	Raw	3	n	DTW
FCM, Fuzzy-C-Medoids, hybrid algorithm	Raw	3	1	DTW
Izakian et al., 2013	Raw	3	2	Augmented Euclidean distance
Izakian and Pedrycz, 2014	Raw	3	n	Weighted sum of blocks
Yin et al., 2006	Raw	3	1	Encoded-Bitmap based similarity
Kakizawa et al., 1998	Raw	3	n	Kullback-Leibler and Chernoff measures
Shumway, 2006	Raw	3	1	Kullback-Leibler measure
Keogh et al., 2007	Raw	3	1	CDM
FSTS	Raw	3	1	STS distance
Kumar et al.,2002	Raw	3	1	seasonalities
Golay et al., 1998	Raw	3	1	Pearson correlation
Golay et al., 1998	Raw	3	1	Pearson correlation
Wissmüller et al., 2002	Raw	3	1	Vector Quantization
Liao, 2007	Raw	4	n	FCM and DTW
Policker and Geva, 2000	Raw	4	1	Temporal clustering
Ji et al., 2014	Raw	3	1	Key points
van Wijk and van Selow, 1999	Raw	3	1	Euclidean distance
Zhang and Yang, 2010	Raw	3	1	Computational Verb Theory

Table 7.2. Summary of raw data dynamic clustering algorithms found in the literature.

Algorithm	**Way to process**	**Type of data**	**No. of dimensions**	**Technique applied**
Guo et al., 2008	Feature-based	3	n	ICA
Fulcher and Jones, 2014	Feature-based	3	n	Summaries
Irpino et al., 2014	Feature-based	3	n	Wasserstein distance
D'Urso and Maharaj, 2009	Feature-based	3	1	Autocorrelation function
Wilpon and Rabiner, 1985	Feature-based	2	1	Log likelihood
Goutte et al., 2001	Feature-based	3	1	Cross-correlation
Fu et al., 2001	Feature-based	4	1	PIP
Owsley et al., 1997	Feature-based	3	1	SOFM
Vlachos et al., 2003	Feature-based	3	1	Haar wavelet
Aghabozorgi et al., 2012	Feature-based	4	1	Incremental clustering
Li and Guo, 2011	Feature-based	3	1	PWCA
Lin et al., 2005	Feature-based	4	n	MPAA
Chandrakala and Sekhar., 2008	Feature-based	4	n	Density-based clustering
Lai et al., 2010	Feature-based	3	1	CAST
Lin and Li, 2009	Feature-based	3	1	SAX

Table 7.3. Summary of feature-based dynamic clustering algorithms found in the literature.

stage the concepts presented by the different authors in their developments of specific-purpose algorithms for time series or sequence data, and presents, as a further step, a development which is able to use different similarity measures to obtain temporal patterns from n dimensional data objects. This development will be presented in the next Part (Part III) of the present document.

Algorithm	**Way to process**	**Type of data**	**No. of dimensions**	**Technique applied**
Li and Prakash, 2011	Model-based	3	1	CLDS
Xiong and Yeung, 2004	Model-based	3	1	ARMA mixtures
Bagnall and Janacek, 2004	Model-based	3	1	Clipped data
Baragona, 2001	Model-based	3	1	Cross-correlation
Maharaj, 2002	Model-based	4	1	AR models
Vilar et al., 2010	Model-based	4	1	Sieve bootstraps
Zhang et al., 2010	Model-based	3	1	Splines and SimilB
Ramoni et al., 2002	Model-based	4	n	Bayes
Ramoni et al., 2000	Model-based	3	n	Kullback-Leibler distance
Savvides et al., 2008	Model-based	3	1	Cepstral coefficients
Kalpakis et al., 2001	Model-based	3	1	ARIMA models
Maharaj and D'Urso, 2011	Model-based	3	1	Cepstral coefficients
Corduas, 2011	Model-based	3	1	Mahalanobis and Euclidean distances
Li and Biswas, 1999	Model-based	3	1	HMM
Oates et al., 1999	Model-based	4	1	HMM
Tran and Wagner, 2002	Model-based	3	1	FCM

Table 7.4. Summary of model-based dynamic clustering algorithms found in the literature.

8

Conclusions and proposal of development

In the present Part, given the current scenario of the Smart Grids and power systems described in Part I, and a main need identified in the sense of a growing demand of dynamic clustering techniques to extract useful information from AMI data, a number of subsequent steps have conducted to the introduction, identification and selection of the path to follow towards the development of a solution for the specific problem of dynamic clustering and visualization of energy consumption load profiles.

As has been previously discussed (see Chapter 4), there is a need of data analysis and extraction of useful information from daily load profiles of energy consumption. The specific objective is the following: to obtain knowledge from large amounts of energy consumption data. These data are being available by the increasing implementation of AMI in households, as has been described in Part I. There will be a need in the near future of how to process these data to obtain interesting and useful information in form of *patterns*, that group similar consumptions in *clusters* and inform about the evolution of the load profiles variation in time, through a sequence of days.

In Chapter 5, the **data mining** objectives and techniques have been presented. From all the data mining objectives described in this Chapter, it is the cluster analysis and, more specifically, the cluster evolution analysis, or the **dynamic clustering of time series data**, the data mining objective which is considered of relevance for the analysis of the AMI data from energy consumption, arranged in daily vectors of 24 hours or dimensions.

For this reason, a state of the art in clustering techniques (Chapter 6) and clustering of time series data (Chapter 7) have been described. The following conclusions are derived from both literature reviews:

- Visualization techniques have to be specifically addressed in more research and development works, especially regarding high dimensional data.
- There are very few generic purpose clustering (and data mining) algorithms. Most of the algorithms described have been designed for specific

kind of data. This fact does not pose, however, a significant drawback, but may add complexity to data mining and Big Data solutions.

- The analysis of time series or dynamic data is approached as a separate field within data mining analysis. The inclusion of the time dimension in the classification and prediction results adds a new factor of complexity to the analysis and, therefore, to the algorithms developed for this purpose.
- Most of the dynamic clustering algorithms reviewed are designed for one - dimensional time series, such as a traffic flow, stock market data, or measures from medical equipment, such as fMRI (functional Magnetic Resonance Image) or ECGM temporal data.
- Most of the dynamic clustering algorithms reviewed are designed for time series, or dynamic data, but a fixed number of classes or clusters (i.e. Type 3 from Table 7.1). Type 4 algorithms (dynamic data and dynamic classes) are also present, mainly in feature-based and model-based algorithms.

These previous conclusions add to the main conclusion of all the reviews performed in this Part II: that **the current literature on dynamic clustering does not address the obtention and visualization of temporal patterns from load profiles time series**. Therefore, the development presented in this thesis is designed for energy consumption n-dimensional data, where all the dimensions have the same magnitude (energy – kWh) and, as an innovation perspective, they are processed as a daily time series (for a week, or a month, or a year...).

The similarity measure that will be developed can be seen as an "augmented" distance, in the sense described by Izakian et al. [240] or Izakian and Pedrycz [241], since the similarity in static or Type 1 clustering is augmented to process time instants in n features or dimensions. It can also be seen as a development based on the description of the membership function of a time series object to a class made by Weber in the FFCM algorithm [90]. The distance function operator d is replaced by specific distance functions able to compare two time series and yield a value of similarity. None of the previous works presented, however, have been developed to analyze and visualize the resulting clusters in the form of n or, in this case, 24 dimensions of dynamic data objects. The development presented in this thesis takes as an initial stage the concepts presented by the different authors in their developments of specific-purpose algorithms for raw time series or sequence data, and presents, as a further step, a development which is able to use different similarity measures to obtain temporal patterns from n dimensional data objects. This development will be presented in the next Part (Part III) of the present document.

The development made, as will be explained, is a flexible framework that can make use of different clustering techniques for **raw data dynamic clustering** presented in the state of the art, extended to process time series data of n dimensions. The resulting clustering process is of Type 3 dynamic clustering (dynamic data, but a fixed number of classes or clusters), applied on the raw data on energy consumption with 24 features or dimensions. Type 4 dynamic clustering is considered as a further step of development from the techniques developed for Type 3; this development, however, is out of the scope of the present thesis. Its approach and possible steps for development are described in the Chapter 15 of future works.

Part III

Development of dynamic clustering techniques applied to load profiles time series

From the conclusions of the Part I, a main need of data analysis is identified. There is a need to segment the end users as a function of their shapes of daily energy consumption, also called **load profiles**, and to obtain **patterns** that allow to classify the users in a group based on how they consume the energy. However, this analysis is limited to a single day. Since the smart metering data are time series formed by sequential measurements of energy through each hour or quarter of hour of the day, but also through each day, thanks to the implementation of AMI and the Smart Grids technologies, it becomes clear that the analysis of the data needs to be extended to consider the **dynamic evolution of the consumption patterns** through days, weeks, months, seasons, and even years. This is precisely the objective of this thesis.

In Part II a complete review and state of the art in possible techniques and solutions has been made. In Chapter 5, the **data mining** objectives and techniques have been presented. From all the data mining objectives described in this Chapter, it is the cluster analysis and, more specifically, the cluster evolution analysis, or the **dynamic clustering of time series data**, the data mining objective which is considered of relevance for the analysis of the AMI data from energy consumption, arranged in daily vectors of 24 hours or dimensions. The development of these specific techniques, therefore, are presented in the following Part of this document, Part III.

These developments are based on previous works found in the literature. There are different criteria when classifying clustering algorithms for time series data. The three main approaches or criteria are the following:

- According to the dynamic nature of data and the clustering algorithms.
- According to the way the time series data is processed.
- According to the number of features or characteristics of the data.

With respect to the dynamic nature of the data and the cluster analysis, Table 7.1, extracted from Weber [90], classifies the cluster analysis in four types or categories, according to the dynamic nature of the data and the clusters:

1. The data is treated as static and the clustering process is also static.
2. The data is treated as static. The number of clusters is not fixed, and may vary at each new computation. In this system, issues such as clusters formation, collapse, split or fusion must be considered.
3. The data is treated as dynamic, evolving through time, therefore the time series become trajectories of the different data features or dimensions through time. The number of clusters is fixed. The objects are clustered taking into account the evolution or trajectories of the object features and, therefore, the resulting centroids or patterns are defined by feature trajectories that are representative of the data evolution in the different resulting clusters. The present thesis approaches this type of dynamic clustering analysis.

4. The data is also treated as dynamic, as in type 3, becoming feature trajectories that evolve through time, and the number of clusters varies dynamically at each iteration. Clusters and patterns can, therefore, as in type 2, merge or split.

Dynamic clustering in general, and, more specifically, Types 3 and 4 in the Table, considers the analysis of the evolution of the features or characteristics of the data object, therefore evaluating and clustering feature *trajectories*, not real-valued vectors [90]. This type of clustering is also called *temporal* data mining, and is the objective of the present thesis. The dynamic clustering is applied on time series data from daily load profiles of energy consumption. The objective is to evaluate the dynamic evolution through the days of the 24 hours or features of a daily load profile, therefore the feature trajectories are to be evaluated.

Regarding the classification made according to the way the time series data is processed, Liao [211] makes a differentiation of clustering types for time series data based on three main approaches: clustering on the **raw** time series data, clustering on a **feature-based** transformation of the time series data, and clustering on a **model-based** transformation of the time series data. The raw data clustering takes the data as they are; feature-based and model-based clustering algorithms perform an initial step where the data are transformed. In the case of the features, by obtaining a reduced dimensionality version of the original data. In the case of the models, by obtaining different representations of the data which can be handled by the algorithm.

Finally, a third classification of the algorithms cab be performed based on the number of dimensions of the data they are able to process. Most of the algorithms found in the literature are designed for one - dimensional time series, such as a traffic flow, stock market data, or measures from medical equipment, such as fMRI (functional Magnetic Resonance Image) or ECGM temporal data.

The development made, as will be explained, is a flexible framework that can make use of different clustering techniques for **raw data dynamic clustering** presented in the state of the art, extended **to process time series data of n dimensions**, where all the dimensions have the same magnitude (energy – kWh) and, as an innovation perspective, they are processed as a daily time series (for a week, or a month, or a year...). The resulting clustering process is of Type 3 dynamic clustering (dynamic data, but a fixed number of classes or clusters), applied on the raw data on energy consumption with 24 features or dimensions. Type 4 dynamic clustering is considered as a further step of development from the techniques developed for Type 3; this development, however, is out of the scope of the present thesis.

The distance function operator d described by different authors such as, for instance, Izakian and Pedrycz [241], or Weber [90] is replaced by specific distance functions able to compare two time series and yield a value of similarity. None of the previous works presented, however, have been developed to analyze and visualize the resulting clusters in the form of n or, in this case, 24 dimensions of dynamic data objects. The development presented in this thesis takes as an initial stage the concepts presented by the different authors in their developments of specific-purpose algorithms for raw time series or sequence data, and presents, as a further step, a development which is able to use different similarity measures to obtain temporal patterns from n dimensional data objects.

Following, this development is presented and tested on a dataset of energy consumption load profiles from a sample of domestic users in Spain.

9

Introduction

In this Chapter, first brief definitions of Energy and Load profiles are included. Although already mentioned many times in the previous Chapters, these definitions will help to clarify the objective of the data analysis pursued in this thesis.

Following, a study is included that recalls the data mining objectives described in previous Sections, and the main roles of the different agents in the management of power systems, and performs a match between roles and data analysis objectives, which will serve to highlight what has been identified as the main need and the solution sought with the development that is presented in the following Chapters.

9.1 Energy measures

Energy is the measure of the *capacity* of work or **power** that can be delivered (either generated or consumed) by a device. When referred to electrical energy, this measure of power delivered is expressed in Watts (W) or kiloWatts (kW) per hour (h), i.e. Wh or kWh, indicating the total amount of power delivered in one hour. As explained in Annex A the power, defined as the work performed by time unit, is computed as the product of the voltage drop (v) in the device and the current flow (i) passing through it (A.7).

9.2 Definition of load profile

Willis [297] describes the **electrical load curve or profile** as the plot of "electric consumption as a function of time", being this time the hour of the day, mainly. In the same article, the concept of **demand** is also described:

> "Demand is the average value of load over a period of time known as the demand interval. Often, demand is measured on an hourly or

> quarter-hour basis, but it can be measured on any interval - seven seconds, one minute, 30 minutes, daily, monthly, annually. The average value of power, p(t) during the demand interval is found by dividing the kilowatt-hours accumulated during the interval by the number of hours in the interval."

The demand is, therefore, the energy, or the integral of the power consumed during the last period of time defined as the interval, usually fifteen minutes or one hour. The measure of the demand must not be mixed with a representation of the load curve performed by instantaneous periodical measures of power. This type of representation, although interesting from the point of view of power quality measurements, does not reflect the total demand during the day.

In his article, Willis also emphasizes the need for large electrical companies to focus on the end-user: their needs and the ways they consume energy, not only as a maximum, aggregated unique value per day, computed at substation level, but going downstream in deeper levels of electrical energy distribution. Willis also points to the main differences in the consumption or load shape among domestic users, being these the climate conditions and the appliances at home, i.e., geographical and socio-economical variables.

The new specifications that arise in the energy market make necessary an approach to an effective measurement and management of the end-user energy consumption and trends, not only concerning the traditionally supervised large consumption customer, but also the medium and high energy consumption residential user, whose consumption profile depicts unbalanced patterns of peaks of energy consumption, and valley or peak–off regions where the energy demand remains unsolicited.

9.3 Data mining objectives on load profiles

Currently, more and more data from energy consumption are becoming stored and available for analysis, by means of Big Data systems and Big Data Analytics tools. These data are measured in form of hour or quarter - hour energy consumption from households or end-user facilities thanks to the massive integration of smart meters and AMI in the grid infrastructure.

The question arises on what are the next steps or, in other words, how these large amounts of data can be used in a way to be profitable to an interested party or agent. KDD and data mining, as has been described in previous Chapters, are a tool for knowledge extraction, but there must be an objective or a final use for the extracted knowledge.

To try to answer this question, in this Section the main objectives of data mining are reviewed and analyzed from the perspective of an agent in power systems and the energy market. As described in Chapter 5, the main objectives of the data mining analysis are the following:

- Static analysis. Comprehends techniques aimed at the analysis of static data, or a set of data from a given time or time frame, or a set of data whose time stamp value is not being taken into account as an objective of the analysis. The possible objectives of this analysis are:
  - Descriptive:
    - Class description.
    - Frequent Pattern Mining.
    - Classification.
    - Cluster analysis.
    - Outlier analysis.
  - Predictive:
    - Association.
    - Correlation.
    - Prediction.
- Evolution analysis. In the evolution analysis, the trend of the series and the temporal evolution of the data is a key factor in the objective of the analysis. Most of the objectives described for the static analysis can be extended in the evolution analysis. However, the inclusion of a new dimension (time) as an objective of the analysis implies the modification of the techniques and algorithms developed for a static analysis.

Regarding the **static** analysis, the information unit in this case is a daily load profile formed by 24 or 96 energy demand values from day i. Each day has 24/96 measures or dimensions (energy consumption per hour or quarter hour), the time stamp, and the ID of the client. Each object in the database corresponds, therefore, to a day with 26 dimensions (in the case of hourly measures of energy demand). To apply static clustering techniques on these data, one of the common procedures [298] is to divide the data according to the season of the year or the type of day (labour / non-labour), since the influence of this temporality in the energy patterns is already known [299]. The main objectives of knowledge to be extracted from these data can be the following:

- **Prediction**. Also called *load forecasting*, is one of the main applications in research in the data mining field on load profiles data [125][300][301]. A state of the art on load forecasting algorithms for load profiles is described in the following Chapter.
- **Cluster analysis**. The identification of patterns that represent the summarized behavior in electrical consumption of a big number of users is another of the main applications of data mining in the cluster analysis of load profiles [302][71].
- **Outlier analysis**. The detection and processing of outliers and other anomalous data, such as noise, is of critical importance in the analysis of load profiles (or in the analysis of any kind of data). Different works in the literature address this issue, either as an initial step in the analysis or data warehousing [73], or as part of the data mining technique itself [303].

- **Classification**. Given the identification of representative patterns in the behavior of electric energy consumption from a group of consumers, the following step is the development of systems that make use of these patterns to classify new measures of energy consumption that may be received daily [304][129]. A state of the art on clustering, pattern recognition and classification algorithms for load profiles is described in the following Chapter.

Regarding the **dynamic** or evolution analysis, the information unit to be analyzed becomes a sequence of n days of daily load demand of 24 or 96 energy values. The data object can adopt the form of a matrix with n rows and 24 or 96 columns. The objective of the data mining is to capture the evolution in time of the load profiles. It allows the obtention of patterns that evolve through time, in a time frame defined by the expert. This allows an interpretation of the results that depicts the full dynamic behavior of all the objects, therefore providing a much more complete (and also complex) information, from where conclusions can be obtained and actions can be determined, to fulfill the purposes of the data mining process. Issues such as identifying specific groups with special trends or shapes in time, or comparing clusters' differences according to their entire behavior in the time frame, can now be considered in the data mining objectives. From this perspective, there are two main objectives of the data mining considered of relevance:

- **Cluster analysis**. The objective is to segment the end users into a number of clusters or groups, as a function of the way they consume the energy in two meta - dimensions of time: *along the day and along the days.* Each hour of the day becomes a feature or dimension trajectory, which propagates along the full sequence of days, for a given time frame of n days, which can be a week, a month, a season, or a year.
- **Classification analysis**. Given a known set of categories or type of clients as a function of how they consume dynamically the energy, classifiers can be developed over these known patterns, to serve as a fast classification and assignment of new customers, or to detect trends and variations in the habits of energy consumption.

The cluster analysis of these kind of temporal data is the main objective of this thesis, as has been emphasized in the different Chapters of this document, from the description of the current scenario and the integration of Smart Grid technologies and ICT in the power grid, to the different states of the art and description of KDD and data mining objectives and techniques.

Following, the data mining objectives that may be of interest on load profiles are analyzed from the perspective of the agent of the energy market. To do this, a summary of the agents' list is recalled from Chapter 2. The list of agents or roles in power systems is the following [4]:

- **Generating companies**. Produce and sell electrical energy.
- **Distribution System Operators or DSO**. Own and operate distribution networks.

- **Retailers**. Buy electrical energy on the wholesale market and resell it to consumers who do not wish, or are not allowed, to participate in this wholesale market.
- **Transmission System Operator or TSO**. It has the primary responsibility of maintaining the security of the power system. It normally owns transmission assets such as lines, cables, transformers and reactive compensation devices, and also the computing and communications assets required to monitor and control the transmission power system. It also usually combines its system operation responsibility with the role of the operator of the market of last resort.
- **Market Operator (MO)**. It typically runs a computer system that matches the bids and offers that buyers and sellers of electrical energy have submitted. It also takes care of the settlement of the accepted bids and offers.
- **Regulator**. It is the governmental body responsible for ensuring the fair and efficient operation of the electricity sector.
- **Small consumers**. Buy electrical energy from a retailer and lease a connection to the power system from their local distribution company.
- **Large consumers**. Often take an active role in electricity markets by buying their electrical energy directly through the market. The largest consumers are sometimes connected directly to the transmission system.

From this list, the following suggestions on data mining objectives can be made:

- **Prediction** techniques are critical for large **Producers** of energy from renewable resources, such as wind turbines. The accuracy of the predictions may affect the energy balance in the power grid directly, having therefore a technical and economical impact for the producer and the other agents involved.
- **Prediction, classification and cluster analysis** can be of great help for **Transmission and Distribution System Operators**, which have to either plan future investments in the grid, supervise the state of assets and equipment and balance produced and consumed energy in near real time, for all the nodes and lines in the power grid, at all the voltage levels.
- **Prediction** techniques can be a valuable support for **Retailers** to help in decision making. Stock market prediction techniques can be applied to the prediction of the energy price, and load forecasting techniques can be used by retailers to know in advance the aggregated estimation of consumption from their clients.
- **Prediction and cluster analysis** are also of interest for the **Consumers**, either large or small. By the identification of the way the energy is consumed, further energy efficiency techniques or demand side management options can be explored, with the objective of reducing the electricity bill.

9.4 Conclusions

Following, a state of the art on the main data mining objectives selected (clustering, classification and prediction) for electrical energy consumption load profiles is presented. All these works refer to the **static** analysis of load profiles since, as has been indicated in Chapter 8, no previous works nor developments have been found regarding the clustering and identification of temporal patterns in load profiles time series. This is the objective of the present thesis and the results are presented in the following Chapters. The next Chapter, first, is devoted to a state of the art on static classification and prediction techniques applied on load profiles. Following, the new perspective of analyzing temporal load profiles will be described and the developments made for this thesis will be presented.

10

Assessment of data mining techniques for the analysis of load profiles

In this Chapter, a state of the art on the main data mining objectives previously identified for the analysis of load profiles is presented. This state of the art is divided in two main Sections: on one hand, cluster and classification developments are presented. On the other hand, load forecasting works are reviewed. As will be seen, the interest on load forecasting and the development of accurate prediction models dates back in the literature to the 1980s and before, whereas the first works in clustering and classification of load profiles start gaining relevance in the 1990s. There is a gap, therefore, in the two research fields, even when sometimes the classification methods are used as inputs for forecasting models. It can be said that currently, forecasting receives more attention than clustering and classification of load profiles. However, as has been explained in the first Chapters of this document, this situation may change in the following years.

10.1 Techniques and algorithms for clustering and classification of load profiles

In the latest years, an increasing number of data mining techniques for classification and pattern recognition have been applied to databases of load profiles, or energy consumption measures, mainly hourly measures. The available data is of different nature, from energy meters in some cases, to secondary and primary substations in others. The results of the analysis have been often used to assign patterns of consumption to the clients' profile associated to the power distribution line, and therefore to extract conclusions on the relation between a consumer's specific characteristics and the profile of its energy consumption.

Following, works found in the literature in this sense are described. All the works found treat the data of load profiles as **static** information from a given period of time and, therefore, the analyses performed are of a static nature.

As will be seen, although divided in two main Subsections, clustering and classification, the two research fields are closely related since, in most cases, the cluster analysis is used as a first step of pattern recognition from a sample of data, and later these patterns are used as the labels or categories where new load profiles and customers are classified. Most of the classification systems found in the literature follow this two-step procedure: first, a cluster analysis, or an ensemble of them, is applied. Then, the results are used as the input for classifiers, developed by applying other artificial intelligence techniques, such as ANN, fuzzy logic or expert systems. Following, the main developments found in the literature are described.

10.1.1 Cluster analysis and pattern recognition

Concerning the analysis of load curves of energy consumption, a number of pattern recognition and segmentation studies have been done. One of the initial works in this subject is the work presented by Pitt and Kirschen in 1999 [305]. The authors build a decision tree clustering algorithm to cluster in hierarchical levels, in a top–bottom approach, a dataset of half-hour energy consumption measures of 500 small businesses over a period of seven months, by binary splitting of each node in two branches, according to one predictor variable at each level. Tariff contracted, season of the year and load factors are variables used, among others, in the splitting process.

Chen et al. [306] obtain the number of classes to characterize the loads from the tariff structure, which is directly related to the voltage consumption level and the usage made (residential, industrial, commercial) by the customers. From this information, a stratified random sampling is applied, choosing carefully the sample sizes to match a specific confidence level and standard deviation. In another work, Chen et al. [307] extend the analysis to other regions or districts, thus adding the geographical dimension to the study, allowing to infer the power system demand by aggregating typical patterns from customers, tailored for each district.

Apetrei et al. [308] perform a segmentation of load profiles aggregated at distributor level for the eight regions of Romania. A clustering algorithm has been run several times for each region, with different numbers of clusters. Ozveren et al. [309] perform a segmentation procedure on a dataset of load profiles from years 1994 and 1995 in the UK. The authors develop a fuzzy inference system that characterizes loads based on the definition of specific similarity measures.

Chicco et al. [129] perform a comparison of clustering techniques (hierarchical tree, K-means, FCM) with a modified "follow-the-leader" [310] clustering procedure and one-dimensional Self Organizing Map (SOM) [84] to segment a sample of 234 non-residential customers. The authors conclude that the modified "follow-the-leader" and the hierarchical clustering run with the average distance linkage criterion provide the best results of intra-clusters dispersion, i.e., the resulting clusters have clear well defined and different centroids or patterns. A similar work is presented in [298], extending the analysis to data dimensionality reduction techniques. An extension to this approach is also presented by Chicco et al. [311] focusing in the analysis of the "follow-the-leader"clustering procedure and the SOM for load characterization.

Akperi and Matthews [312] perform another comparison of static clustering techniques applied to load profiles, but from measurements at a primary substation at the distribution level. Their results indicate that K-means clustering shows the best performance in generating unique, well-populated cluster groups. To evaluate the resulting clusters, the authors use two cluster validity indices, the Davies - Bouldin or DB index [207] and the Xie-Beni or XB index [208], and also a classification of the substations based on a PCA of the load profiles, in rural/urban, and commercial–industrial/residential uses.

Chicco et al. [121] develop a complete comparative study of classification techniques, including clustering methods. Identification of the loading conditions (e.g., season influence and a partition of days in working and non-working days) is suggested in this work, as a pre-treatment stage. A further work from Chicco [209] addresses a survey of clustering methods for segmentation of load profiles and the validity indices for clustering performance assessment. The following techniques for segmentation are reviewed: adaptive vector quantization (AVQ), entropy-based, follow-the-leader (FDL), fuzzy logic, fuzzy and ARIMA models, FCM, hierarchical clustering, iterative refinement clustering (IRC), K-means, min–max neuro-fuzzy (MMNF), multivariate statistics (MANOVA), probabilistic neural network (PNN), SOM, support vector clustering (SVC) and weighted evidence accumulation clustering (WEACS).

Tsekouras et al. [304] also perform a comparative method of different classification techniques, validating the results with adequacy measures and indicators for the resulting partitions, such as the Davies - Bouldin or DB index [207]. In another work, Tsekouras et al. [313] develop a classification framework for MV customers, which makes use of an initial step of load segmentation, by applying and comparing different clustering methods and SOM by the use of adequacy indicators, such as the cluster dispersion and the DB index.

Zhou et al. [314] present a classification framework based on clustering. Different techniques are presented and tested: K-means, FCM, hierarchical clustering and SOM. The authors indicate three main applications for a clustering and classification analysis: identification of erroneous measurements, load forecasting based on a clustering initial step, and tariff setting based on load classification.

Haben et al. [315] study the characteristics of load profiles at household level, identifying four key periods through the day, and apply model-based clustering and bootstrap techniques to characterize residential customers in ten representative patterns.

Chaouch [316] presents a work where the cluster analysis is used as an initial step for load forecasting of household energy consumption. A hierarchical clustering algorithm is applied on historical records of load profiles to obtain the M typical patterns of past energy consumption. Then, each resulting cluster can be related to a number of segments of load profile records. This information is used as an input by the forecasting algorithm, which estimates a future segment of energy consumption values as a function of the weighted sum of the segment of the day before and all the most similar past segments, i.e. all the segments that belong to the same cluster than that of the day before. Similarity is computed as the Euclidean distance between the discrete wavelet coefficients of two segments.

Viegas et al. [317] present a work with the same objective but applying different techniques. The FCM algorithm is used to extract the typical load profile patterns from a dataset of half-hour energy consumption readings from 4232 Irish households over a period of 1 and a half year. Then, the patterns are used as the knowledge base for different prediction models, such as fuzzy inference, RBF networks or Support Vector Machine (SVM) models [318].

Kwac et al. [319] present a complete framework for customer characterization based on clustering. The measurements from load profiles are divided in two variables: the total consumption per day and the normalized shape or daily profile. Two clustering procedures are developed to analyze and segment the normalized shapes: first, an adaptive K-means clustering performs clustering varying the number of clusters, until a suitable number is found. Then, a hierarchical clustering algorithm can combine clusters to better adjust the results. From the resulting clusters, a statistical analysis is derived to characterize all the customers that are assigned to each cluster.

10.1.2 Classification

Sforna [320] presents one of the initial works in classification of clients applying data mining techniques. A SOM is trained with historical records from a set of clients of a DSO. Sixteen variables, obtained from the monthly energy consumption through a year, are used as inputs. From the resulting patterns on the SOM, a fuzzy inference system is designed, with a number of rules whose final objective is to detect anomalous situations of energy consumption, which may lead to uncover fraud situations or measurement errors from metering equipment.

Gerbec et al. [321] develop a classification framework based on a decision tree and fuzzy inference. The classification of new load profiles to existing patterns is done by a probabilistic neural network. The system is tested with a sample of customers from different DSOs, comparing the resulting classes with the customers' economic activity code.

Valero et al. [130] apply SOM on a dataset of load profiles of industrial, institutional and small residential loads, all of them previously classified. The authors produce variables from the data set in the time domain and in the frequency domain, building classification maps with different input variables and comparing the results. A classification success ratio is of 100 % achieved in the three scenarios: hourly load profile, time domain indices and Discrete Fourier Transform (DFT).

In another work, Valero et al. [303] study the adequacy of SOM for classification. Different SOM are trained, with time domain and frequency domain data, and then they are used to classify new customers to one of the resulting patterns from the training step.The results show that frequency domain data allow a better classification of load profiles from new customers, although further research is needed to validate these conclusions.

In the work developed by Benítez et al. [71], SOM are used as the tool that allows classification of users according to their load profiles, by treating each daily load profile as a vector of 24 dimensions or hourly energy measures (expressed in Wh). The SOM algorithm allows an easy identification and group formation over a 2D grid displaying the complete scenario of load profiles based on the dataset used for training, comprised of daily energy consumption profiles from a sample of 625 monitored clients of the Spanish DSO Iberdrola Distribución Eléctrica S.A.U., along the year 2008. Each measure includes the hourly load profile, and also a number of specific indices that define the consumption pattern and the characteristics of the client. The use of these indices and the corresponding values of the prototypes of each group, allows using the trained SOM to both classify new users according to their load profiles, and estimate their energy consumption characteristics. The data are analyzed separately based on season of the year and working or non-working day, therefore 8 SOM are trained and used for classification per year.

The so trained SOM classifier allows a correspondence between new users' load profiles, and one of the obtained patterns, thus defining the users' energy consumption based on a single measure. The information of a classified client is related to the indices attached to the pattern that his consumption best fits to. The values of the indices for each obtained pattern or prototype are computed as the average or modal values of the indices' values from all the users assigned to that cluster.

The final step of the analysis is the classification of the total subset of data with the trained SOM, yielding a membership to one of the obtained clusters for each sample or load profile. The assignment of a client to a pattern or prototype is made choosing the cluster with the highest number of the user's load profiles assigned.

Ten clusters, each with a prototype and a set of indices attached, have been identified in each SOM. The ten clusters can be located on the SOM, as can be seen in Fig. 10.1, whereas the prototypes that define the different energy consumption patterns can be seen in Fig. 10.2. As can be seen, the classification made by the SOM is influenced by the shape of the load profile and the level of energy consumption (the data are not normalized, since all the dimensions have the same magnitude). This classification, therefore, allows to identify groups of consumers which are of great interest from the point of view of the demand side management objectives, i.e., groups of consumers with a high level of energy consumption, a high number of peaks of energy consumption per day, a high value of the peak/valley relationship, and a high value of the estimated index of potentially manageable consumption. In this analysis of the Winter season, working days, for instance, prototypes 3 and

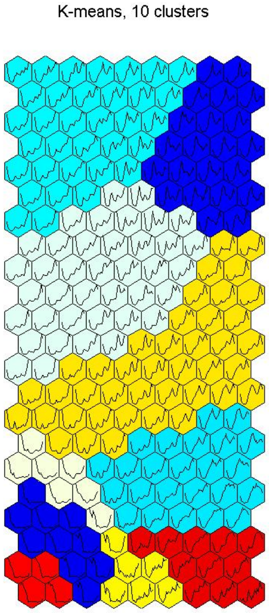


Fig. 10.1. Clustering algorithm applied on the trained SOM.

10, and the clients assigned to each group, are the groups which best fit with these objectives.

Figueiredo et al. [86] present a framework for consumer characterization based on a combination of unsupervised and supervised learning techniques. The authors use SOM and K-means clustering [177][166], as the basis for a classifier of load profiles from electricity customers, based on decision trees or Artificial Neural Networks (ANN) [112]. The data are initially separated in groups according to season of the year and working and non-working days. Chang and Lu [322] present a similar framework for classification, that uses FCM to obtain the classes, and a multilayer ANN for the classification of new customers.

Varga and Czinege [323] present a classification framework that uses as inputs the *weekly* load profiles from customers, instead of days. First, the characteristic load profiles are obtained applying clustering techniques and SOM. Then, a classification system is built using a feed-forward three-layer ANN. The dimension of the weekly load profile data is reduced by testing different techniques, such as PCA and Genetic Algorithms.

Chicco et al. [324] define the indices used for characterization and classification of load profiles in two categories: *a priori* indices, derived from the contractual nature (e.g. contracted power), and *field* indices, which calculated

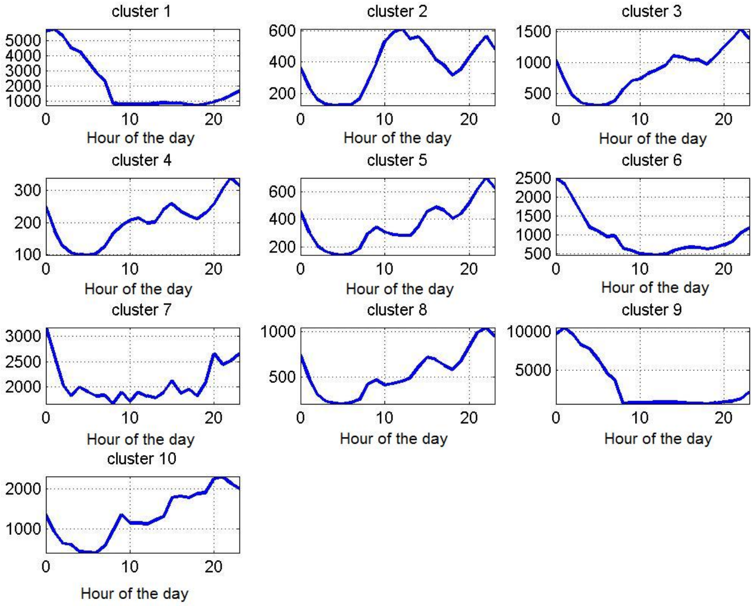


Fig. 10.2. Prototypes obtained from the SOM classification.

from the load profiles (such as, for instance, the peak/valley relation). A classification based on these indices is presented for a sample of 471 non-residential customers, and the representative load profiles from the resulting classes are analyzed, with the objective to define new tariff schemes.

Ramos et al. [325] present a classification system of MV customers based on a previous step where the main patterns or consumption prototypes are obtained from an ensemble of clustering algorithms on a selected sample of consumers. A set of validity indices is used to select the most suitable number of clusters by an expert; the resulting patterns are used as the knowledge base for a classification tree. Specific indices for load characterization are defined by the authors to help in the classification procedure.

10.2 Techniques and algorithms for forecasting of load profiles

According to the distinction of techniques gathered by García [326], there has been an evolution in load forecasting models and techniques through time, indicating the following four main groups of analysis:

1. Models until the 1980s, mainly based in adaptive filters and regression models. The formulas developed during this period made use of linear models based on Fourier analysis, regression techniques, and load models as transfer functions. Works that make use of probability estimations, such as Markov processes, are also found.
2. Models developed during the 1980s years, make use of autoregressive algorithms and techniques such as the aggregation of load patterns. Works based on statistical techniques and Markov processes were also developed during this period. Initial works based on non-linear models are also found.
3. During the 90s, models that combine electric variables with categorical or qualitative variables are developed. As an example, Quijano [327] develops a prediction model of loads for medium voltage lines, which makes use of variables such as an hourly activity index, a classification of the clients activity, day and season, and climate conditions (ambient temperature).
4. Finally, from the year 2000 onwards, soft computing methods have been increasingly applied. Examples of techniques used are Artificial Neural Networks (ANN), Fuzzy logic and hybrid fuzzy-neural models.

Hernandez et al. [328] describe two main criteria to classify forecasting models for electric loads:

- According to the prediction horizon: the typical objectives are Very Short-Term Load Forecasting (VSTLF), from seconds or minutes to several hours; Short-Term Load Forecasting (STLF), from hours to weeks, and Medium-Term and Long-Term Load Forecasting (MTLF and LTLF), from months to years.
- According to the aim of the forecast: Two main groups exist. The first group is formed by those that forecast only one value (next hour's load, next day's peak load, next days total load, etc.); the second group consists of forecasts with multiples values, such as next hours, peak load plus another parameter (for example, aggregated load) or even next days hourly forecast- the so-called *load profile.*

the main techniques applied in different works are described next.

10.2.1 Mathematical models based on quantitative and qualitative variables

Abou-Hussien et al. [329] present a STLF model designed for power system operation, where the hourly load j from day i is computed as a function of the weekly load pattern for the day i and hour j, plus the influence of a weather sensitive component.

Willis and Brooks [330] propose a forecast model for a complete year (8760 hours) based purely on the aggregation of past load values for the same hour of the same day of the different classes of customers connected to the same node in the electricity grid.

Walker and Pokoski [331] propose an innovative approach to STLF in the sense that the model developed for residential loads is a direct influence of the human behavior. The paper describes the concepts of an "availability" function which statistically estimates the number of people in a household available to use an appliance, and a "proclivity" function which gives the probability that an individual will use that appliance at any given time of day. These functions are then used to drive "physical" models of the various appliances and these are incorporated into a "combined model" which is used to estimate the load on a time of day basis.

Lu et al. [332] propose a Hammerstein nonlinear system to model the relationship between load and temperature for short-term load forecasts. An adaptive algorithm with an orthogonal escalator structure and with a lattice structure for joint processes is introduced to update the load model and predict total system load.

Rahman and Shrestha [333] present a forecasting model for short term based on the priority vector technique. The priority vector based load forecasting technique uses pairwise comparisons to extract relationships from presorted historical hourly load and weather records. the technique is a blend of expert system and statistical techniques. It uses expert knowledge to filter the historical data to form a subset which is most pertinent to the forecasting environment. Then, it transforms the individual data set in this subset to a relative scale to reflect the overall characteristics of the subset. Statistical technique is then applied to this data set for forecasting purposes.

Quijano [327] proposes in his PhD thesis a model for the estimation of the load in Medium Voltage (MV) networks, defined by the partition of the daily load curve in hourly activity groups, for different types of customers. The magnitude used is the power unit (p.u.), referred to the secondary substation rated power.

Up to seven hourly activity groups are proposed by the author, for different types of customers: residential, industry, small and medium enterprise. The values of consumption for each activity group are defined empirically by the author, as average values or linear increasing or decreasing trends, from real measures obtained at secondary substations. The total demand is obtained by a weighted aggregation of pattern loads at each secondary substation, considering the percentages of type of customers connected. The effect of the temperature is also recorded and modeled, as a variation in the demanded power.

García [326] presents in her PhD thesis a prediction model for the loads in low voltage (LV) networks which is, in fact, a characterization of the entire low voltage network from the perspective of the energy consumption habits of the customers connected to it. The system requires as inputs specific information from the LV network topology, from the secondary substation to the connection to the protection panels where the LV customers are connected. The detailed information from all the customers' electricity billing (and at which physical switch they are connected from the panel) is also needed, along with measures of daily load curves at secondary substation level and from a representative sample of the LV customers. The modeling process is the following:

First, a characterization process of all the LV customers is performed, in order to obtain representative patterns from socio-economic information. This process is done in two steps: a factorial analysis is performed on the customers billing information dataset, which comprehends variables such as contracted power, energy consumed, location, electricity tariff, economic activity, etc. The factorial analysis yields a reduced number of variables which are linear combination of the original ones, therefore this is a dimensionality reduction step. Following a hierarchical fuzzy clustering is applied, obtaining a set of LV customer patterns attending to socio-economic factors.

A prototype load curve is then assigned to each of the customer types obtained before. To do this, a sufficient sample of LV customers with measurable load curves must be present in each of the previous resultant clusters. The prototype load curve is computed from the load curve measures of a sample of customers belonging to each cluster.

All the estimated load curves per customer type are then aggregated at two levels, protection panel and secondary substation, making use of the convolution operator as a moving average weighted by the contracted power per customer. The statistical density functions are computed from the resulting aggregated loads at the secondary substation. An estimation error is then computed by comparing the results with measured loads at the secondary substation level. Different models are trained with this procedure, according to the season of the year, the type of day, and other scenarios.

10.2.2 Autoregressive models

Vemuri et al. [301] present a work on electric load forecasting by ARMA (AutoRegressive Moving Average) model, in comparison with other approaches, such as AR (Autoregressive), MA (Moving Average) or Box-Jenkins. The authors develop STLF adaptive on-line models to forecast the hourly load demand on the system.

Cho et al. [334] develop a STLF ARIMA (AutoRegressive Integrated Moving Average) model which includes temperature as an input. For four types of customer in the Taiwan power system (residential load, commercial load, office load and industrial load customers) the summer ARIMA model transfer function models are derived and tested with real data from one week.

Pappas et al. [335] perform a study on the modeling of load curves by ARMA models to de-seasonalized load curves from the aggregated power demand from a Greek DSO. The paper studies first whether the electricity loads in the Hellenic power market can be modeled by an ARMA process and secondly on a comparison of ARMA model order selection criteria under the presence of noise. The current study also presents a new method for multivariate ARMA model order selection and parameter estimation based on the adaptive multi-model partitioning theory.

Cancelo et al. [299] present the building process and models used by Red Eléctrica de España (REE), the Spanish system operator, in short-term electricity load forecasting. The forecasting system implemented in REE consists of one daily model for forecasting the daily load up to ten days ahead, and 24 hourly models for computing hourly predictions for horizons up to three days. The daily model is aimed at producing forecasts for network outage planning, while the hourly models are used to derive forecasts for the next-day hourly dispatch. The hourly load forecast is estimated as a function of four components: the normal (or baseline) load, the weather (and seasonal) sensitive part, the influence of special events and a random component. ARIMA models are used to estimate the different variables used in this function.

Tso and Yau [131] develop and compare forecasting models for electricity consumption in Hong Kong by three different techniques: regression models, decision trees and neural networks. Different models are develop for Summer and Winter seasons, and the results are compared.

Espinoza et al. [302] perform time series regression on 245 series of four-year measurements of hourly electricity consumption at primary substations, provided by the Belgian National Grid Operator ELIA. The load is predicted at each hour, as a function of the load consumption at the 48 previous hours, plus a seasonality component and an error term. Exogenous variables such as temperature and weekday are also included. Ordinary Least Squares are used to identify the parameters in the forecasting models.

10.2.3 Artificial Neural Networks (ANN) models

The inputs to prediction models based on ANNs in all the works reviewed are mainly the past load values and a forecast of weather variables [132]. One of the initial works concerning the use of ANN can be found in [120], where a multilayer ANN is developed for the prediction of active power hourly values from a Taiwanese power system. A multilayer neural network with an adaptive learning algorithm is presented, for short term load forecasting, in order to predict the 24 hourly loads for the next day. The system uses as inputs the forecast of weather for the same day and predefined patterns of normalized 24 hourly loads per type of day, computed from historical records. The adaptive learning is achieved by defining a variable momentum learning rate, with the objective to speed up convergence.

Mohammed et al. [336] present also an adaptive learning multilayer ANN, developed for the Florida Power and Light Company. In this case, the system works in real time and thus it has to adapt to the changing environment and conditions. It also implements a specific behavior to deal with the prediction in abnormal situations, such as days of extreme temperatures, with do not have enough precedence in the historical records. The overall system consists of two modules. One is the system forecast initialization (SFINIT) module and the other is the real time module. The SFINIT mode is used to train various neural network architectures and obtain weights for different years, seasons or months. This process is done off-line. The real time mode utilizes the ANN weights obtained using the SFINIT module and allows the user to interactively obtain a forecast on-line. Forecasts for 1 hour to 7 days can be obtained based on forecast temperature.

Beccali et al. [337] present a multilayer ANN for STLF, with the additional feature of developing a first step classification of the historical loads and weather records by applying a Self Organizing Map (SOM). The system is, therefore, a forecasting model for load curves in a sequence of supervised and unsupervised learning: first, a SOM is used to classify in a number of clusters the loads along with the weather data, obtaining a pattern or prototype for each cluster. Then, this classification is provided as input to the ANN, which will produce the 24 hours forecast for the next day based on the information on the resulting clusters from the SOM on the load profiles from the past two days.

Banda and Folly [338] present a multilayer ANN for STLF with similar inputs as in the work from Cancelo et al.: the normal (or baseline) load, the weather sensitive part (temperature), the influence of special events (holidays) and a random component. A second model is developed which includes rainfall as input, having a better performance as claimed by the authors.

Xia et al. [125] develop Radial Basis Function (RBF) ANNs for STLF, MTLF and LTLF. The inputs are historical records of electricity consumption and weather variables, such as temperature, wind and pluviometry.

10.2.4 Self Organizing maps (SOM) models

Although they are classified as ANN, SOM or Kohonen networks are a particular design for model learning. Lendasse et al. [339] develop a STLF model for the prediction of the daily load profile. On the trained SOM, probability transitions in the Voronoi areas among centroids are computed. The authors choose the prediction for the next day as the load profile with the highest transition probability rate from the current pattern load.

Carpinteiro and Reis [340] present a STLF model developed as a hierarchical model with two self-organizing maps, one on top of the other. The input at the bottom layer is a normalized vector of seven components: the load at the current hour, the load at the hour immediately before, the load at twenty-four hours behind, at one week behind, and at one week and twenty-four hours behind the hour whose load is to be predicted. The sixth and seventh units represent a trigonometric coding for the hour to be forecast. The input to the second, top layer SOM, are the distances of the trained units in the bottom SOM.

10.2.5 Fuzzy inference models and expert systems

Rahman and Bhatnagar [341] develop an expert system for the prediction of one-to-six hour and 24 hour ahead forecasts from a power utility. The expert system considers the logical and syntactical relationships between weather (mainly temperature) and load, and the prevailing daily load shapes. The seasonal variation and the loads' thermal inertia are also modeled.

Kuo and Hsu [342] propose the use of fuzzy sets to estimate the loads in a distribution system and to devise a proper service restoration plan following a fault. First, typical load patterns for commercial, industrial and residential customers are obtained as the average of historical records from these three types of customers, for each type of day to be considered (weekday, weekend). The patterns are normalized to the rated transformer capacity at each node. The resulting patterns are also divided according to the level of hourly normalized energy consumption in five levels: very small, small, medium, large and very large consumption. This way, the 24 hourly load patterns can be expressed in 24 values of these five linguistic descriptions, which are then defined as fuzzy membership functions. From the measure of the total power at the feeder in the primary substation, the loads at each node of the branch are estimated by obtaining the load types connected to the node for the type of day specified, normalized by the transformer capacity, and fuzzified.

10.2.6 Statistical and probabilistic models

Seppala [343] presents a study on the adequacy of statistical distribution functions to customer hourly load profiles. From a historical record of load curves from LV customers for 10 years, and an initial classification of the customers in 46 different classes, the authors studies how well the distribution functions represent the behavior of each hourly load for each different customer class. Four different distribution functions are tested: normal, log-normal, log-normal estimated by percentiles, and a modified log-normal distribution function, being this last one considered as the most interesting by the author. A further research in this sense is presented by Stephen et al. [344]. The authors develop models for load profiling of residential customers based on mixtures of Gaussian models and Factor Analyzers.

Belzer and Kellogg [345] develop Monte Carlo simulations to analyze sources of uncertainty in forecasting annual peak power loads. The authors use historical records of power demand and weather conditions, and apply Monte Carlo analysis to incorporate the uncertainty of the disturbances in the estimated daily load model. An Extreme Value Distribution (EVD) is used for each Monte Carlo simulation and then the estimated EVDs are used to derive a composite distribution. A daily peak model is presented, as a function of the minimum daily temperature, a binary variable that defines the type of day, and an error component.

10.2.7 Hybrid models

Mori and Kobayashi [346] present a fuzzy inference system for short term load forecasting which implements an initial adaptive learning phase to tune the fuzzy membership function parameters. The fuzzy inference is of the Takagi-Sugeno-Kang type, where the output from each rule is a constant value, being the result from the fuzzy system the weighted sum of the fired rules. Triangle fuzzy membership functions are used, whose parameters are adjusted by a simulated annealing optimization algorithm.

Kim et al. [347] develop a hybrid model for short-term load forecast that integrates artificial neural networks and a fuzzy expert system. First, the hourly load curve is predicted by an ANN, trained with values from the hourly loads from past days and weeks. Then, a fuzzy expert system modifies the estimated load by taking into account variations due to the influence of temperature and holidays.

Srinivasan et al. [348] present also a short term load forecasting model built from the combination of neural networks with fuzzy inference. The inputs to the forecasting model are the daily minimum and maximum temperatures, rain indication, seasonal variation, day-of-the-week and special day effects, in addition to the historical load data. All the inputs are fuzzified by triangular membership functions and a fuzzy knowledge base is built with rules based on the experience of the distribution operators. The set of inputs and outputs from the fuzzy system is used to train a three layer backpropagation ANN, which provides the fuzzy degree of membership to the load membership functions. The final step is the defuzzification of this output, to provide the 24 values of load consumption for the next day.

Khan and Abraham [300] perform a comparative study of STLF models for predicting the aggregated 48 hourly (2 day ahead) power demand from a Czech utility, using six different techniques: multilayer ANN, Elman recurrent neural network, radial basis function network, Hopfield model, fuzzy inference system and hybrid fuzzy neural network. The models are trained and tested using the hourly load data obtained from the Czech Electric Power Utility (CEZ) for seven years (January 1994 – December 2000). Their results indicate that hybrid fuzzy neural network and radial basis function networks are the best candidates for the analysis and forecasting of electricity demand.

10.3 Conclusions. Extending analysis to dynamic clustering

The present Chapter has described previous works found in the literature that analyze data of load profiles with three objectives: clustering, classification and prediction (forecasting). The objective of the present thesis, as has been stated in different Sections, such as Section 0.2 and Chapter 4, is the dynamic clustering of load profiles time series and the visualization and analysis of the resulting patterns. The result of the state of the art performed in the present Chapter aligns with the conclusions obtained from the previous review of the state of the art on dynamic clustering techniques (as discussed in Chapter 8): **the objective sought is not addressed in any of the works reviewed**. There is a need, therefore, for the development of specific techniques that can provide as a result temporal patterns that display the evolution and trends on energy consumption daily profile through time. This is the objective of the last Chapters of this document, where these techniques are described, developed and tested, and results and conclusions are produced.

11

Development of algorithms and techniques to perform dynamic clustering on load profiles time series

11.1 Introduction

Concerning the dynamic clustering, the objective of performing dynamic segmentation on a time series database is to obtain dynamic centroids, i.e., patterns that represent a number of objects whose features may vary with time, but remain similar enough to pertain to the same cluster. The objective, therefore, is to obtain a set of patterns that depict the full evolution of the data through time.

There are four issues, concerning the data characteristics and visualization of time series data, that must be considered. These are the following:

- **Data visualization**. Adequate visualization options should be designed or defined in order to properly visualize the evolution of the clusters along time.
- **Data temporal milestones**. Milestones in clusters evolution should be automatically recorded, concerning the number of clusters, and information concerning their sizes and shapes.
- **Data granularity**. The data from energy consumption are available as hourly or quarter-hourly energy measures, through a day, for a number of n sequential days. Since it is an objective of the analysis to observe the evolution of the load profile through time, the 24 hours are preserved; however, the daily time measure can be variable, in days, weeks, or months, according to data variability. Moreover, it might be adequate to condense the information of load profiles per weeks, in order to remove the effect of labor days variation in energy consumption, since it would affect the dynamic clustering in an undesired level of detail. The energy information could be provided, therefore, as the average of hourly energy consumption per week, or the aggregated load per week.
- **Traceability**. It is important to study the trace of each client along the subsequent analyses, observing when and where there is a change in cluster's belonging and behavioral trend.

These possibilities will be explored and the solutions adopted will be described. The clustering technique must deal with objects that may have time-series discontinuities, i.e., gaps between time measures, which can vary among the different objects that form the database. This is significantly true for daily load curves: measures from the meters can be unavailable for some days, due to unknown reasons, such as a malfunction or maintenance. These discontinuities do not happen with the same frequency nor at the same days for all the clients, therefore occasional, unpredicted discontinuities appear at the data, which does not necessarily mean that a client has been removed from the database, simply that there are not available measures. Clients in this situation should be kept at their last state within a cluster, until a new measure is analyzed. Other factors, such as the presence of noise and outliers, must be taken into account in a data warehousing process, to conveniently filter the load profiles data before the cluster analysis.

As has been seen in previous Chapters (Chapter 8 and Chapter 10), the presented techniques so far do not fulfill the objectives defined in this thesis. Therefore, something else is needed. A new framework is presented in this Chapter that addresses the dynamic clustering, visualization and identification of temporal patterns in load profiles time series, fulfilling the detected gap in this area. The present development is a generic framework that allows the clustering and visualization of load profiles time series or, more specifically, of n-dimensional time series data where the dimensions have the same magnitude (such as load profiles).

The developed framework allows the use, as an extension for dynamic clustering, of virtually any kind of static clustering technique and similarity measure. The most common in the literature have been used, and also a new similarity measure, specifically for this kind of data, has been described and tested. More specifically, the following developments have been made for this thesis:

1. The description of the **data model**, or data format, for the time series data, to be analyzed in a context of dynamic clustering of time series (Section 11.2).
2. The development of a **data warehousing** process for the filtering and rearrangement of the data in a format suitable for the posterior analysis (Section 11.3).
3. The development of specific **validity indices for dynamic clustering**, obtained as extensions of common indices used in the literature for static clustering (Section 11.4).
4. The development of a **common framework for the dynamic clustering and visualization of daily load profile time series**, suited to be used by most of the static clustering techniques and similarity measures found in the literature (Section 11.5).

5. The development of a new **similarity measure for the dynamic clustering of n-dimensional time series data**, such as load profiles time series (Section 11.6).

Following, these developments made are being presented.

11.2 Data Model

When addressing the dynamic segmentation of the daily load profiles database, different options of granularity can be applied, according to the objectives of the segmentation. Some of these options are, for instance, the following:

1. As n sequential daily load curves.
2. As cumulated or average n weekly or monthly load curves, to avoid weekly variations due to working days.
3. As n sequential daily load curves arranged by type of day, to avoid weekly variations due to working days (e.g., all the weekend daily curves only, all the working day curves only, etc.).

The three options have been implemented and tested, as will be seen in the following Chapter.

Regarding the data model for the analysis, the matrix form is chosen due to its computational possibilities, and for a visualization of the results as 3D surfaces. The matrices used and obtained will have the following dimensions:

- Inputs:
  - The processed **data** matrix of objects to be clustered (Fig. 11.1). The size of this matrix is the following: number of rows = (number of clients or objects) x (maximum number of samples or observations per client or object data); number of columns = the object data dimensions, either qualitative and quantitative, as stated for instance in Table 12.1.
  - The initial matrix of **centroids** (c_{ini}) (Fig. 11.2). The size of this matrix is the following: number of rows = (number of clusters to be found) x (maximum number of samples or observations per client or object data); number of columns = the centroid data dimensions, with all the dimensions that are being clustered (i.e., load profiles of 24 or 96 measures). In this case, the 24 hourly energy demand from customers in the analyzed time frame.
- Outputs:
  - Resulting matrix of **centroids** (c). This matrix has the same size as c_{ini} (Fig. 11.2).
  - **Membership** matrix (u) (Fig. 11.3). This matrix indicates, for each data object, its membership value to all the clusters defined. Therefore, the number of rows is equal to the number of clusters to be found x the maximum number of samples or observations per client or data object,

in a similar way to the centroids matrix. This information is obtained for every object of the data set, thus the number of columns equals the number of objects.

– **Exclusive cluster membership** (*pertain*) (Fig. 11.4). Obtained from the u matrix, *pertain* indicates, for each object, the cluster number it belongs to, i.e., the one to which the membership value is the highest from all the clusters. Therefore, each sample of each object is assigned a value, from zero (no cluster or no data) to the the number c of clusters that have been found. The number of rows equals the number of objects, and the number of columns equals the maximum number of samples.

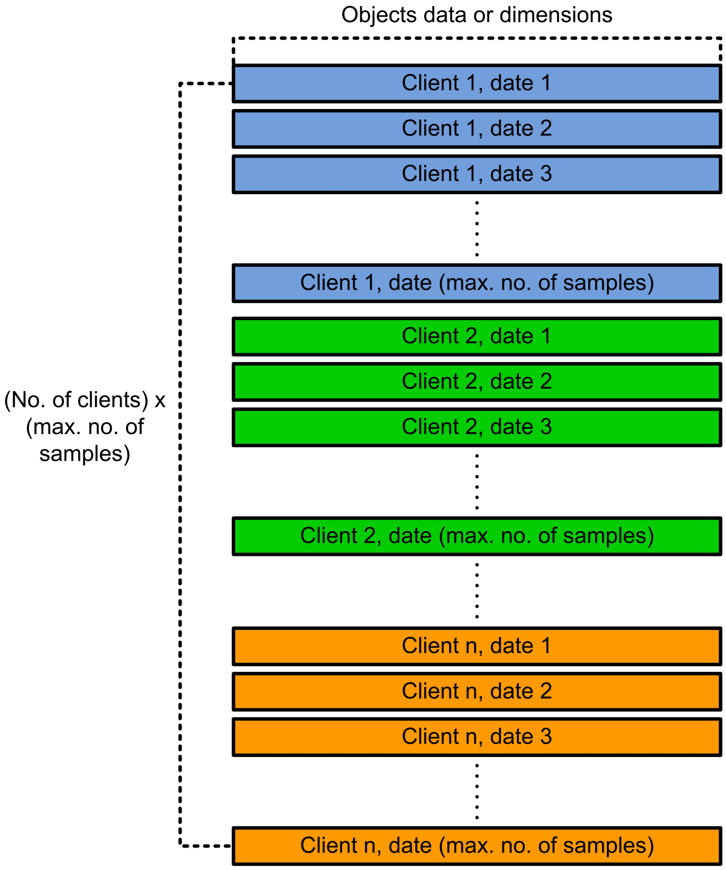

Objects data or dimensions
Client 1, date 1
Client 1, date 2
Client 1, date 3
Client 1, date (max. no. of samples)
Client 2, date 1
Client 2, date 2
Client 2, date 3
Client 2, date (max. no. of samples)
Client n, date 1
Client n, date 2
Client n, date 3
Client n, date (max. no. of samples)
(No. of clients) x (max. no. of samples)

Fig. 11.1. Data cell structure after preprocessing.

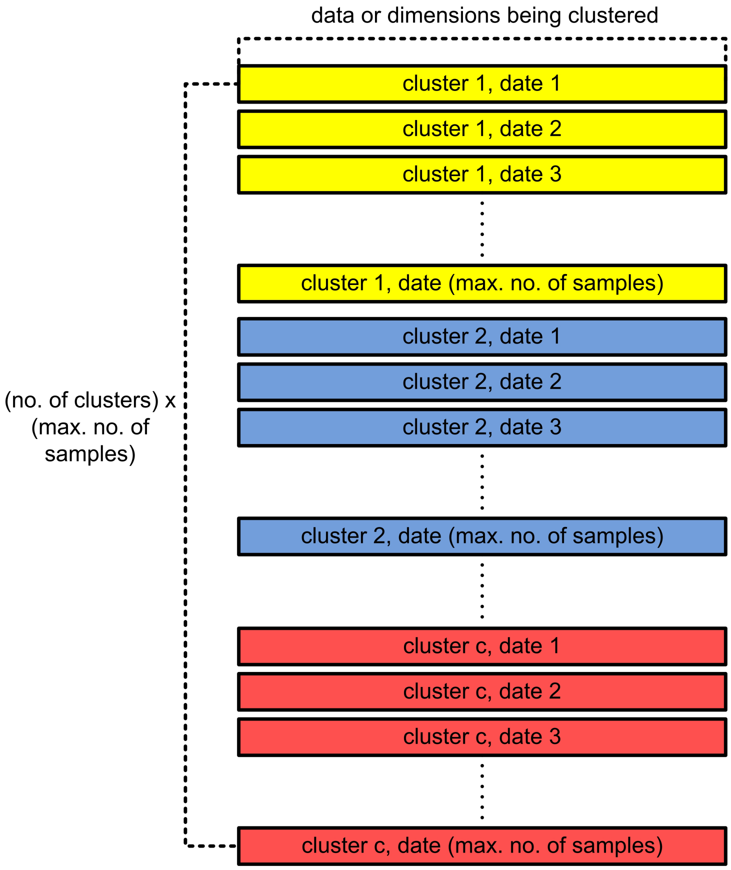

data or dimensions being clustered
cluster 1, date 1
cluster 1, date 2
cluster 1, date 3
cluster 1, date (max. no. of samples)
cluster 2, date 1
cluster 2, date 2
cluster 2, date 3
cluster 2, date (max. no. of samples)
cluster c, date 1
cluster c, date 2
cluster c, date 3
cluster c, date (max. no. of samples)
(no. of clusters) x (max. no. of samples)

Fig. 11.2. Matrix of centroids (c, c_{ini}).

11.3 Development of a data warehousing procedure for the pre-processing of time series data

As stated in previous Chapters of this document, typically the cluster analysis is done in two steps:

1. A data warehousing or preprocessing step is performed, to arrange the data in a cell format, sorted by client and date, and filter noise or erroneous data.
2. The resulting data set is used as input to the selected dynamic clustering algorithm.

A preprocessing stage must sort all the data per client and date, fulfilling blank or lost dates with zeros or other values, such as the average. In this stage a filtering of the data is also applied, to remove noise and outliers. Figure 11.5

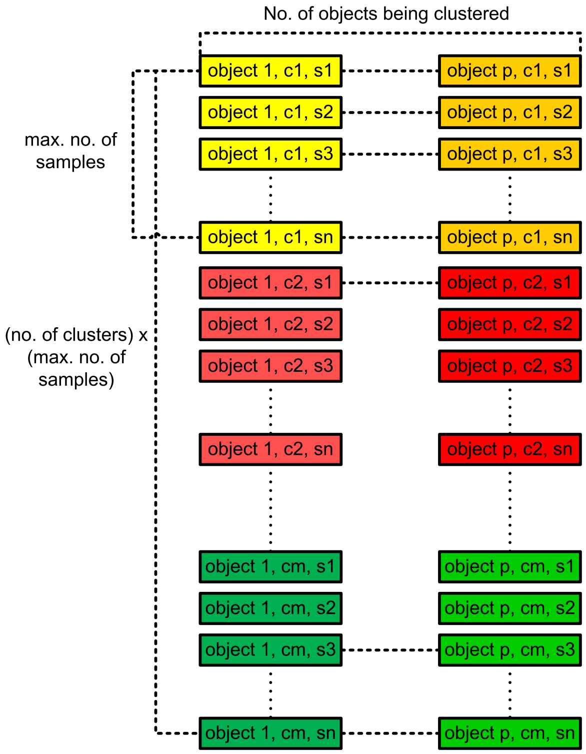


Fig. 11.3. Membership matrix (u).

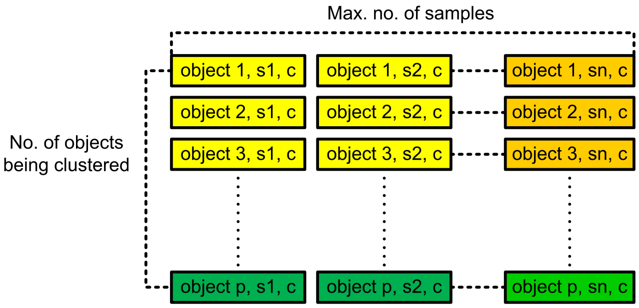


Fig. 11.4. Exclusive membership matrix (*pertain*).

displays the graph of this sequence, in this case fulfilling empty values with zeros. Other techniques have been used in different tests, such as the following:

- Fill the empty dates with the average of the values with dates.
- Fill the empty date with the average from the precedent and consequent dates with values.
- Completely remove objects from the data set if the number of empty dates exceeds a given threshold of samples (such as one third of the maximum number of samples, for instance).

Normalization of the data can also be applied in this stage. The normalization has been applied depending on the similarity measure used. In the case that the Euclidean distance is used, since the 24 dimensions have the same magnitude (energy in Wh), normalization is not considered.

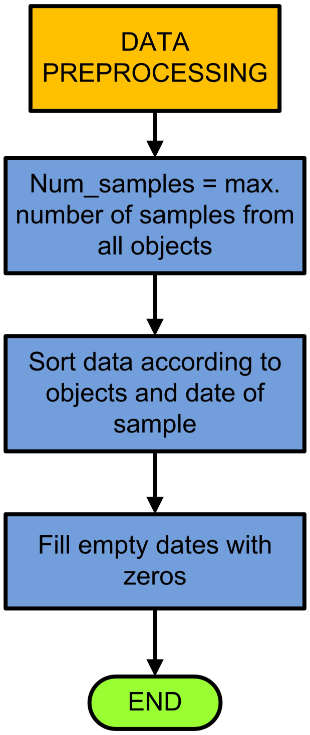

DATA PREPROCESSING
Num_samples = max. number of samples from all objects
Sort data according to objects and date of sample
Fill empty dates with zeros
END

Fig. 11.5. Data preprocessing stage.

11.4 Development of cluster validity indices for the evaluation of dynamic clustering on time series n-dimensional data

Following, a number of cluster validity indices are described, for the comparison of the results of dynamic clustering algorithms on time series data. The indices described are extensions for the dynamic analysis of common static clustering validity indices. Annex D of the present document includes a complete description of these indices found in the literature. Most of them, as indicated in the Annex, are variations that take into account the dispersion or variance within the resulting clusters (intra-cluster dispersion) and the separation among the cluster centroids. A good segmentation process is one that produces well distinct (separated) cluster centroids and, at the same time, assures that the dispersion of data in all the clusters is minimized. This description, however, is not aligned with the objectives of a fuzzy partition, for instance, where all the objects can belong to all the existing clusters. For this reason, some validity indices are defined specifically for fuzzy clustering algorithms.

The clustering validity indices were initially described to assess the initial selection of a number of clusters for partitional clustering algorithms. In his PhD thesis, Díez [166], for instance, gathers some of these validity indices (Dunn, DB and FS), and cites other ones, such as the Hubert index, as part of a review of methods to determine the best number of classes for partitional clustering. Different authors, however, as stated in Section 6.15, have used them to compare the results of different clustering algorithms on the same data sets, to evaluate the performance of the different algorithms. They will also be used in this thesis for this purpose.

Following, a brief description of the indices is included, and their modification for the evaluation of the dynamic clustering results is described. Since fuzzy clustering techniques are included in the developments for this thesis, all the indices described have been modified to evaluate dynamic fuzzy partitions. Table D.1 in Annex D gathers the static clustering validity indices described in this Section along with more indices described in the literature, and indicates the expected values that would imply a good selection of the number of clusters.

11.4.1 DB index

This index was described by Davies and Bouldin in 1979 [207]. It defines the similarity measure R_{ij} between two clusters i and j as in (11.1), where s_i and s_j are measures of dispersion of the clusters (11.2) and d_{ij} is the distance between the two clusters, which is typically obtained as the Euclidean distance between the two cluster prototypes or centroids. C_i is the size of the cluster i, obtained from the number of elements that belong to the cluster.

$$R_{ij} = \frac{s_i + s_j}{d_{ij}} \tag{11.1}$$

$$s_i = \frac{1}{\mid C_i \mid} \sum_{x \epsilon C_i} \parallel x - c_i \parallel \tag{11.2}$$

The expression R_{ij} indicates, therefore, a relation between the compactness of the clusters and the distances among them. The Davies-Bouldin or DB index is defined in (11.3), where N_c is the number of clusters and R_i is defined in (11.4).

$$DB = \frac{1}{N_c} \sum_{i=1}^{N_c} R_i \tag{11.3}$$

$$R_i = max_{j=1...N_c, i \neq j} R_{ij}, i = 1...N_c \tag{11.4}$$

Since the objective is to obtain clusters with the highest separation among them and the lowest dispersion, it is expected that the lower the value of the DB index, the better the results of the clustering will be.

Modification of the DBI to analyze dynamic fuzzy clustering

Concerning the fuzziness of the resulting clusters, in this case mainly two options could be studied:

1. A weighting factor could be used when computing the dispersion (s_i) values between clusters, and the size of each one ($\mid C_i \mid$). This weighting factor would be no other than the fuzzy membership value of each object to each cluster.
2. A simpler method would consist in converting fuzzy to non-fuzzy clusters, by assigning each object to the cluster with the highest fuzzy membership value.

Concerning the modification of the DBI to analyze the results of dynamic fuzzy clustering, it can be observed that the only term affected by the dynamic clustering is the computation of distances, either between objects and centroids, and also between centroids. The needed modification would affect, therefore, Equations (11.1) and (11.2), which would be rewritten as in (11.5) and (11.6).

$$R_{ij} = \frac{s_i + s_j}{d_{ij}} \tag{11.5}$$

$$s_i = \frac{1}{\mid C_i \mid} \sum_{x \epsilon C_i} d_i \tag{11.6}$$

Being now d the distance between dynamic objects, which can be any of the similarity measures described for the dynamic cluster analysis (see Chapter 7).

11.4.2 SD index

The SD validity index [349][210] is based on a relation between distance among clusters and the clusters' dispersion or scattering. The formula proposed by the authors for the average scattering of the clusters is given in (11.7), where $\sigma_{(c_i)}$ is the variance of each cluster (see (11.8)), and $\sigma_{(X)}$ is the variance of the whole data set (11.9).

$$Scatt = \frac{1}{N_c} \sum_{i=1}^{N_c} \frac{\|\sigma_{(c_i)}\|}{\|\sigma_{(X)}\|} \tag{11.7}$$

$$\sigma_{(c_{ij})} = \frac{1}{\|C_i\|} \sum_{k=1}^{n_{ij}} (x_{kj} - c_{ij})^2 \tag{11.8}$$

$$\sigma_{(x_j)} = \frac{1}{N} \sum_{k=1}^{N} (x_{kj} - \overline{x_j})^2 \tag{11.9}$$

The formula proposed to define the total separation or distance among clusters is expressed in (11.10).

$$Dis = \frac{max_{i,j=1 \ldots N_c}(\|c_i - c_j\|)}{min_{i,j=1 \ldots N_c}(\|c_i - c_j\|)} \sum_{k=1}^{N_c} \left(\sum_{j=1, i \neq j}^{N_c} \|c_i - c_j\| \right)^{-1} \tag{11.10}$$

Combining the two expressions, the SD index is obtained as indicated in (11.11). The parameter α is a weighting factor, needed in order to normalize the range of *Scatt* and *Dis* expressions. Lower values of the SD index indicate better results, since the compactness of the clusters is bigger and also the separation among them.

$$SD = \alpha Scatt + Dis \tag{11.11}$$

Modification of the SD index to analyze dynamic fuzzy clustering

Concerning the fuzziness of the resulting clusters, the same options mentioned before could be studied:

1. A weighting factor could be used when computing the clusters' dispersion or scattering, affecting the size of each cluster ($\|C_i\|$) and its variance. This weighting factor would be the fuzzy membership value of each object to each cluster.
2. A simpler method would consist in converting fuzzy to non-fuzzy clusters, by assigning each object to the cluster with the highest fuzzy membership value.

this conversion affects the scattering or dispersion of the results. The modification needed to analyze the results of a dynamic clustering involves, moreover, modifications in the computation of the scattering and the separation among clusters, since the distance between centroids and objects, and between centroids, must be replaced by an operator d that computes distance between two dynamic objects. The resulting Equations would be (11.12), (11.13) and (11.14).

$$\sigma_{(c_{ij})} = \frac{1}{\|C_i\|} \sum_{k=1}^{n_{ij}} d_{kij}^2 \tag{11.12}$$

$$\sigma_{(x_j)} = \frac{1}{N} \sum_{k=1}^{N} d_{kj}^2 \tag{11.13}$$

$$Dis = \frac{max_{i,j=1 \ldots N_c}(d_{ij})}{min_{i,j=1 \ldots N_c}(d_{ij})} \sum_{k=1}^{N_c} \left(\sum_{j=1, i \neq j}^{N_c} d_{ij} \right)^{-1} \tag{11.14}$$

11.4.3 PC index

The *partition coefficient* (PC) index was proposed by Bezdek [199]. This index, expressed in (11.15), computes the level of fuzziness of the resulting clusters. The value of the PC index ranges between $1/N_c$ and 1, indicating a value of 1 that the partition is totally non-fuzzy, and a value of $1/N_c$ that the partition is very fuzzified, i.e., most of the objects belong to all the clusters in similar percentages. This result can be due to a non-appropriate selection of the number of clusters, or due to the nature of the data, which do not clearly adhere to a group of different patterns, or these patterns are very similar.

$$PC = \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{N_c} \mu_{ij}^2 \tag{11.15}$$

Therefore, a value of the PC index closer to 1 would be a better result than a value closer to 0. A value of 1, however, is indicating that this is an exclusive partition. The numeric result of this index needs to be evaluated along with the nature of the data and their variability.

Modification of the PC index to analyze dynamic fuzzy clustering

Since the index is computed from the fuzzy membership values of each object to each cluster, it is not necessary to introduce any further change in order to evaluate the results of dynamic fuzzy clustering.

11.4.4 XB index

The Xie-Beni (XB) index [208] involves the fuzzy partitions and the data set. The index relates the compactness of the clusters with the separation among them, as can be seen in (11.16).

$$XB = \frac{\sum_{i=1}^{N_c} \sum_{j=1}^{N} \mu_{ij}^m \|x_j - c_i\|^2}{N \min_{i,k} \|c_i - c_k\|^2} \tag{11.16}$$

Since the objective is to obtain clusters with the highest separation among them and the lowest dispersion, it is expected that the lower the value of the XB index, the better the results of the clustering will be.

Modification of the XB index to analyze dynamic fuzzy clustering

The XB index can be seen as a fuzzy equivalent of the DBI, therefore the index is suitable for fuzzy clustering. Concerning the dynamic clustering, the distances between centroids and objects, and between centroids, must be replaced by an operator d that computes distance between two dynamic objects. Equation (11.16) must be replaced by the expression in (11.17).

$$XB = \frac{\sum_{i=1}^{N_c} \sum_{j=1}^{N} \mu_{ij}^m d_{ij}^2}{N \min_{i,k} d_{ik}^2} \tag{11.17}$$

11.4.5 FS index

The Fukuyama - Sugeno (FS) index [350] combines a measure of fuzziness of the clusters with a measure of fuzziness among the different clusters, as expressed in (11.18) and (11.19).

$$FS = \sum_{i=1}^{N_c} \sum_{j=1}^{N} \mu_{ij}^m \|x_j - c_i\|^2 - \sum_{i=1}^{N_c} \sum_{j=1}^{N} \mu_{ij}^m \|c_i - \overline{c}\|^2 \tag{11.18}$$

$$\overline{c} = \frac{1}{N_c} \sum_{i=1}^{N_c} c_i \tag{11.19}$$

For a good selection of the number of clusters, the value of the FS index should be as low as possible.

Modification of the FS index to analyze dynamic fuzzy clustering

The index is suitable for fuzzy clustering. Concerning the dynamic clustering, the distances between centroids and objects, and between centroids and centroid average, must be replaced by an operator d that computes distance between two dynamic objects. Equation (11.18) must be replaced by expression (11.20).

$$FS = \sum_{i=1}^{N_c}\sum_{j=1}^{N} \mu_{ij}^{m} d_{ij}^{2} - \sum_{i=1}^{N_c}\sum_{j=1}^{N} \mu_{ij}^{m} d_{i}^{2} \qquad (11.20)$$

11.4.6 Summary of the dynamic distances needed

The distance between two dynamic objects to be computed, from all the indices described, can be summarized in the following:

- A distance between a centroid i and all the objects belonging to that cluster i.
- A distance between two cluster centroids, i and j.
- A distance between all the objects of the data set and the objects' average value.
- A distance between all the centroids and the centroids' average value.

For the computation of these distances, any of the similarity measures described for the dynamic cluster analysis (see Chapter 7) could be used, or new similarity measures could be defined.

11.5 Development of a common framework for the dynamic clustering and visualization of daily load profile time series

Following, two developments are presented. The first one is the extension of a K-means static clustering to the dynamic clustering of n 24 hours daily load profiles, by comparing the objects' similarity at each time instant (in this case, each day). A different distance between two objects is computed for each day, and then a final distance is obtained as the average of the n measures.

The second approach applies the concept of Type 3 dynamic clustering described in previous Chapters. In this case the feature or dimension trajectories of the objects are clustered. The dimensions of two objects are compared as sequences of n samples, and a final distance is obtained as the average of all the comparisons of features at the 24 dimensions. Although the two approaches may deal similar mathematical results, they are quite different and, depending on the operators and similarity measures used, may produce very different outcomes. The first approach can be seen as a succession of Type 1 static clustering calculated for n times and clustered together. The second approach is designed as a Type 3 dynamic clustering of dynamic trajectories through time, with a fixed number of classes.

In this development, average values are used as the final similarity measures between dynamic objects. Other similarity measures, however, could be used. For instance, objects can be treated as temporal surfaces and compared geographically, obtaining not only one, but a set of characteristic distances that defines the similarity between the surfaces. The following Section presents a development in this sense, based on the Hausdorff distance. Although the Hausdorff distance has been used in clustering to compare similarity between geometrical objects [222], time series data [223], trajectories [224] and interval or symbolic data [226][227], none of these previous works, however, implement its use for the clustering of time series in the way it is used in the development presented.

11.5.1 Approach 1: comparing dimensions by the same time instant or day sample

A dynamic K-means clustering algorithm has been developed, by modifying the static K-means algorithm to obtain the similarity distances among objects taking into account all the Euclidean distances between each pair of objects from their coincident time stamps. The process diagram of this computation is illustrated in Fig. 11.6.

The steps for the static K-means clustering algorithm to evaluate the data are the following:

1. Select k objects and set them as the initial prototypes of the k clusters that are to be found.
2. Compute all the Euclidean distances of the remaining objects to the k prototypes. Assign each object to the cluster with the smallest distance.
3. Compute clusters prototypes or centroids as the average or mean value from all the objects that belong to the cluster, with the objective to minimize the cost index shown in (11.21). This index is a summation of all the k summations of the distances from all the objects to the centroid of each cluster.
4. Proceed with the two previous steps until a termination condition is reached, like the variation in centroids falling under a predefined limit.

$$J = \sum_{i=1}^{k} \left(\sum_{j, z_j \in A_i} \|z_j - c_i\| \right) \tag{11.21}$$

A number of analyses has been performed applying this algorithm, varying the data to be clustered. The results are described in the following Chapter (Chapter 12).

11.5.2 Approach 2: evaluating dimensions as dynamic features

The objective is the development of dynamic clustering algorithms for time series databases with n dimensions of the same length and nature (i.e., magnitude). A general framework for this analysis is defined, from where these four

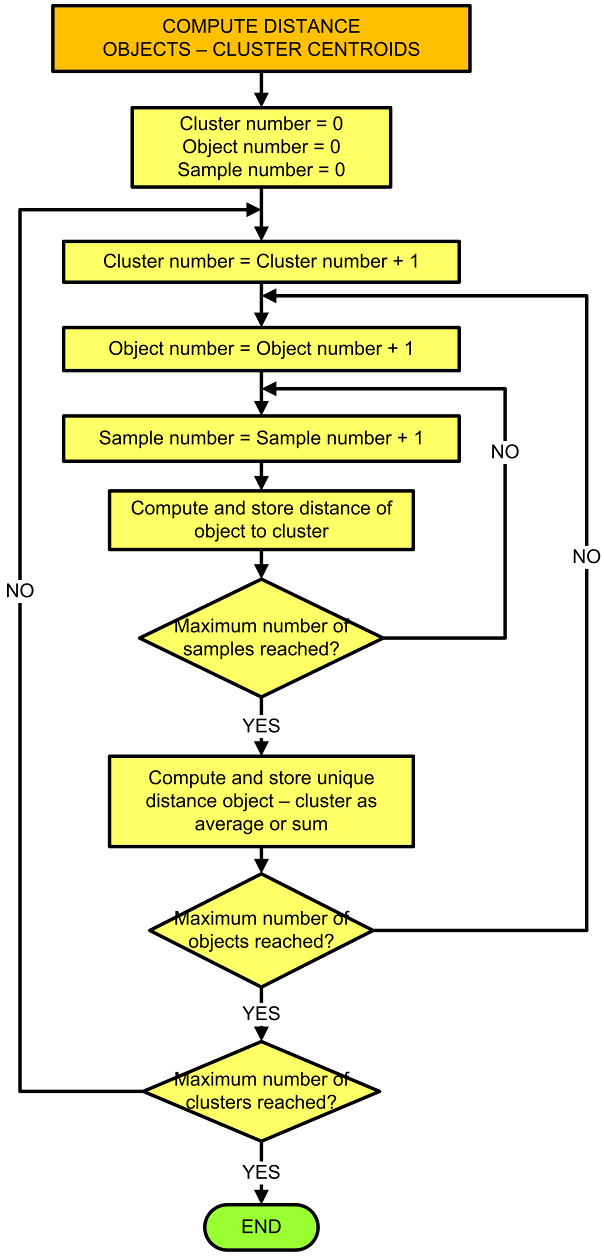

COMPUTE DISTANCE
OBJECTS – CLUSTER CENTROIDS
Cluster number = 0
Object number = 0
Sample number = 0
Cluster number = Cluster number + 1
Object number = Object number + 1
Sample number = Sample number + 1
Compute and store distance of object to cluster
Maximum number of samples reached?
NO
YES
Compute and store unique distance object – cluster as average or sum
Maximum number of objects reached?
NO
YES
Maximum number of clusters reached?
NO
YES
END

Fig. 11.6. Computation of distances between objects and clusters.

dynamic clustering algorithms are developed, allowing the development and test of more dynamic clustering algorithms by the combination of different techniques and similarity measures.

The objective is to obtain a set of patterns that depict the full evolution of the data through time, i.e., patterns that represent a number of objects whose features may vary with time, but remain similar enough to pertain to the same cluster. The time series data analyzed must have the following characteristics:

- The features or dimensions of the data must have the same length, i.e., the same number of samples or time stamps. Therefore a preprocessing stage is necessary, in order to harmonize the same number of samples for all the objects, filter noise data, and fill in the missing data values.
- The features or dimensions of the data must have the same nature or magnitude. This is a limitation in the scope of the analysis, motivated by the nature of the data being analyzed, which is the daily load profiles of energy consumption. These data objects present 24 dimensions (the 24 hours of the day) and a similar magnitude (energy consumption). The similarity between two time series objects is computed, as is explained next, as the average of distances between the objects' dimensions or feature trajectories, therefore all the dimensions must be of the same nature or magnitude, or the similarity computation would be distorted, since the data is not normalized. A further improvement in this framework can consider the time series clustering of objects with dimensions of different nature and magnitudes.

A general framework is presented for this analysis, called the **Equal N - Dimensional (END) Time series Clustering Framework**. Within this framework, this work presents the development of partitional dynamic clustering algorithms, obtained as an extension of the classical, static techniques, where the $2D$ data and patterns are extended to $3D$ time series data and patterns. The way to perform this extension is based on the description of the FFCM by Weber (see Chapter 7), where other similarity measures have been used instead of defining fuzzy inference to compute the distances. For doing so, the END framework developed envelops the partitional clustering algorithm and modifies it in order to obtain the similarity distances among objects taking into account the distances between their feature or dimension trajectories. Applying this method, two different static clustering techniques, K-means and Fuzzy C-means or FCM, have been extended to dynamic clustering. Two different similarity measures, based on the Euclidean distance and on the correlation measure, have been used. The resulting dynamic clustering techniques have been called: **END-FCME** (END FCM Euclidean-based), **END-FCMC** (END FCM Correlation-based), **END-KME** (END K-Means Euclidean-based) and **END-KMC** (END K-Means Correlation-based). Table 11.1 indicates the combination of techniques used by Weber, and the ones developed and presented in this work. This method can be extended to most of the clustering techniques found in the literature.

Table 11.1. Formation of different Type 3 dynamic clustering algorithms based on Weber description.

Static clustering technique	dynamic objects similarity measure	**Resulting dynamic clustering**
FCM	Fuzzy membership functions	**FFCM (described by Weber)**
FCM	Euclidean distance	**(END-FCME) END FCM with Euclidean-based distance (present work)**
FCM	Correlation	**(END-FCMC) END FCM with correlation-based distance (present work)**
K-means	Euclidean distance	**(END-KME) END K-means with Euclidean-based distance (present work)**
K-means	Correlation	**(END-KMC) END K-means with correlation-based distance (present work)**

All the objects are assigned the same number of time samples or instants. The missing samples in this case have been filled with the average of the preceding and forthcoming values. Once all the data objects are harmonized, the distance between each cluster and the object is computed between feature trajectories. This is the global value used in the computation of membership values, in (7.8). Two different techniques for obtaining this distance have been applied: Euclidean distance and correlation. In the case of Euclidean distance, the computation is shown in (11.22), where n is the number of features or characteristics of the data, and B is the norm. Since the Euclidean distance is used, the identity matrix has been applied.

$$d(X_i, V_j) = \frac{1}{n}\sum_{k=1}^{n} \|X_{i_k} - V_{j_k}\|_B^2 = \frac{1}{n}\sum_{k=1}^{n}((X_{i_k} - V_{j_k})^T B(X_{i_k} - V_{j_k})) \quad (11.22)$$

In the case of correlation, the *Pearson* correlation coefficient between two series is computed between each pair of feature trajectories, as can be seen in (11.23) and (11.24), where p is the total number of time samples or instants. The result is an index between $[-1, 0, +1]$, which yields the linear relationship degree between the two series, which is later conveniently transformed to a distance measure, applying the equation expressed in (11.25).

$$corr(X_i, V_j) = \frac{1}{n} \sum_{k=1}^{n} corr(X_{i_k}, V_{j_k}) \tag{11.23}$$

$$corr(X_{i_k}, V_{j_k}) = \frac{\sum_{m=1}^{p} (X_{i_{k_m}} - \bar{X_{i_k}})(V_{j_{k_m}} - \bar{V_{j_k}})}{\sqrt{\sum_{m=1}^{p} (X_{i_{k_m}} - \bar{X_{i_k}})^2} \sqrt{\sum_{m=1}^{p} (V_{j_{k_m}} - \bar{V_{j_k}})^2}} \tag{11.24}$$

$$d(X_i, V_j) = \frac{1 - corr(X_i, V_j)}{2} \tag{11.25}$$

The procedure to perform the dynamic clustering with the END framework, as described in the present work, can be summarized in the following steps:

1. Initialize the C matrix of centroids with random values, or other methods.
2. Obtain all the distances of the objects to the centroids of the clusters, by the formula indicated in (11.22) for Euclidean distance, or (11.25) for correlation-based distance, or any other suitable for time series.
3. Compute membership matrix U, applying (7.8), in the case of END FCM, or assigning each object to the cluster with the smallest distance, in the case of END K-means, or any other method (See Fig. 11.7).
4. Compute the cluster centroids according to the formulas for K-means or FCM in each case.
5. Repeat steps 2 to 4 until a termination condition is met, such as reaching a maximum number of iterations.

In the following Chapter (Chapter 12) these described dynamic clustering algorithms, along with the FFCM, are applied on different time series data sets, describing the results obtained.

11.6 Development of a two-step time series clustering algorithm with a Hausdorff-based similarity distance for the dynamic clustering of daily load profile time series

A specific development is presented in this document, being a dynamic clustering algorithm which applies a similarity measure to compare two energy consumption load profiles time series, as two dynamic surfaces, based on a two-step sequence: the decomposition of the objects in smaller linear surfaces, and the computation of Hausdorff distances [220] between the surfaces.

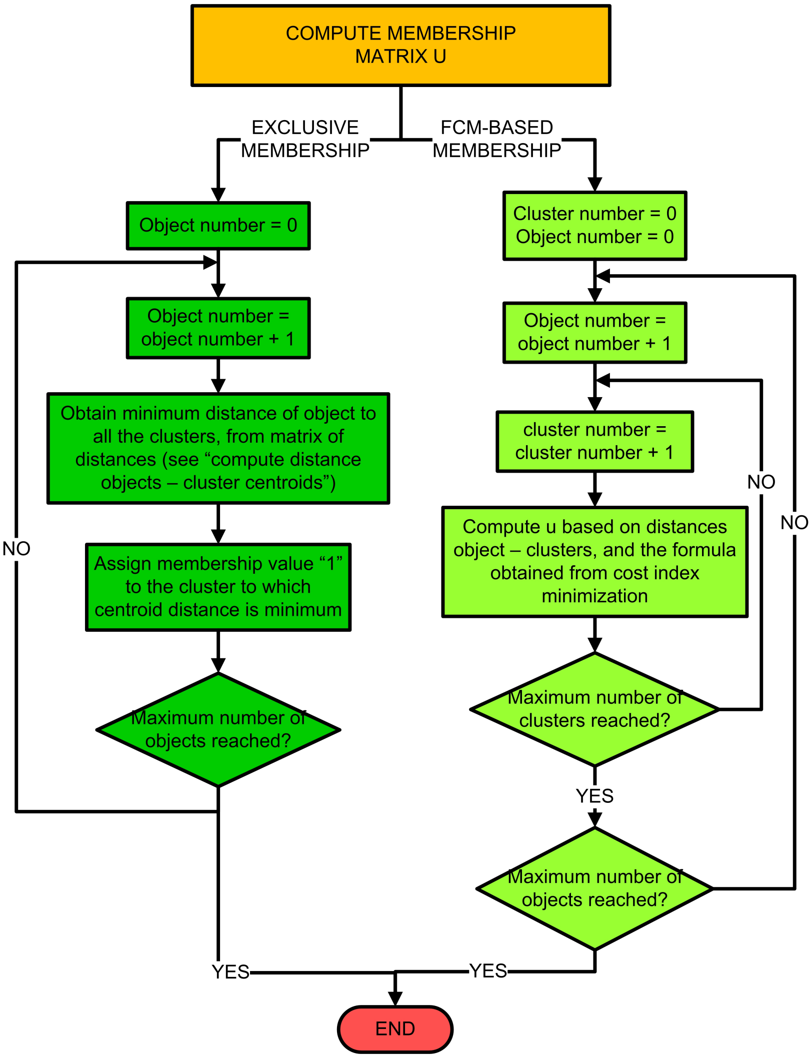

COMPUTE MEMBERSHIP MATRIX U
EXCLUSIVE MEMBERSHIP
FCM-BASED MEMBERSHIP
Object number = 0
Object number = object number + 1
Obtain minimum distance of object to all the clusters, from matrix of distances (see "compute distance objects – cluster centroids")
Assign membership value "1" to the cluster to which centroid distance is minimum
Maximum number of objects reached?
NO
YES
Cluster number = 0
Object number = 0
Object number = object number + 1
cluster number = cluster number + 1
Compute u based on distances object – clusters, and the formula obtained from cost index minimization
Maximum number of clusters reached?
NO
YES
Maximum number of objects reached?
NO
YES
END

Fig. 11.7. Computation of membership matrix u.

11.6.1 Description of the two-step time series clustering algorithm

The energy consumption profiles from residential users are seen as 3D surfaces, defined by the 24 hours load profile, where each hour is considered as a feature or dimension of the data; and the number of days (one year in the present work). The time series clustering algorithm needs to define a measure of similarity between each client's profile and all the resulting centroids at each iteration. The Euclidean distance is usually used for this in most works.

The proposed solution in this work is a two-step process, which is described next. First, all the shapes are decomposed in a number of linear surfaces, by applying least squares regression. The number of surfaces and the vertices is predefined, based on the expert's knowledge of the typical behavior from residential users regarding energy consumption. Then, the resulting surfaces are compared by computing the Hausdorff distance between them, and a global similarity value is obtained, given by the average value of all the Hausdorff distances between the different surfaces.

11.6.2 Decomposition of shapes in smaller linear surfaces

As indicated, the shapes of energy consumption are decomposed in a number of linear surfaces, applying least squares regression. Given a number of observations $(z_1, z_2, ..., z_n)$, each value can be written as a function defined by a number of independent variables (x), their coefficients (w), and a residual or error value (ε_i), which is an error measure between the observed value and the expected one. This formulation for the equation of a surface can be seen in (11.26) and, summarized in matrix form, in (11.28).

$$z_i = w_0 + w_1 x_{i1} + w_2 x_{i2} + \varepsilon_i \tag{11.26}$$

Expression (11.26) can be arranged as a matrix for the number n of observations, yielding expressions (11.27) and, in simplified form, (11.28).

$$\begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{pmatrix} = \begin{pmatrix} 1 \; x_{11} \;\; x_{12} \\ 1 \; x_{21} \;\; x_{22} \\ \vdots \\ 1 \; x_{n1} \;\; x_{n2} \end{pmatrix} \begin{pmatrix} w_0 \\ w_1 \\ w_2 \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{pmatrix} \tag{11.27}$$

$$Z = XW^t + \varepsilon \tag{11.28}$$

In order to obtain the coefficients that fit the observations to the desired function, the Least Squares method minimizes the Error function, which is defined as the sum of squared residuals, or the difference between the observations and the function outcomes, as can be seen in (11.29).

$$Error = \frac{1}{2} \sum_{i=1}^{n} (z_i - f(x_i, w))^2 \tag{11.29}$$

From these expressions, the formula to compute the coefficients of the linear surface that best fits the observations, as a batch process, is obtained, as can be seen in (11.30).

$$W = (X^t X)^{-1} X^t Z \tag{11.30}$$

The linear least squares method is described in more detail in Annex C.

11.6.3 Similarity measure between surfaces based on the Hausdorff distance

The Hausdorff distance [220] has been described in Chapter 7. The Hausdorff metric between two non-empty closed subsets, A and B, is defined as the maximum of all possible distances $d(\tilde{a}, B)$, as can be seen in (7.5). The Hausdorff distance $d_H(A, B)$ between the two subsets is obtained as the maximum of their two Hausdorff metrics, $h(A, B)$ and $h(B, A)$ (11.31).

$$d_H(A, B) = max(h(A, B), h(B, A)) \tag{11.31}$$

11.6.4 Pseudocode

Summarizing, the pseudocode of the two-step time series clustering algorithm previously described is the following:

```
main loop
  compute centroids
  for each object and centroid shapes
    Step 1: partition the shapes
      in n x m linear surfaces
    Step 2: compute the n x m
      Hausdorff distances
    Compute the mean of the n x m
      Hausdorff distances.
    Use it as the similarity measure
  end loop
Compute membership values of objects
  to clusters based on similarity
end main loop
```

11.7 Conclusions

Three different developments have been presented in this thesis. The first one is an initial approach for the dynamic clustering of time series data objects. The second one is a generic framework for Type 3 dynamic clustering of n dimensional data. The main difference between the first and second approach have been described. In this generic framework, called END framework, any type of static partitional clustering techniques from the literature could be applied, since the framework extends the cluster analysis for the segmentation of dynamic objects of n dimensions (24 in this case). Although in this development it is required that the dimensions have the same magnitude, a further research can be made to extend this analysis to n dimensional objects of any nature and magnitude, by applying a normalization step of all the dimensions, or a weighted sum of the resulting similarity measuere for each pair of feature trajectories.

The third development makes use of this framework, but goes a step further in considering the dynamic objects as surfaces that evolve through time. A two-step dynamic clustering algorithm is described, which decomposes these surfaces in smaller linear ones, and computes a similarity measure used in the literature to compute distances between surfaces: the Hausdorff distance.

In the following Chapter, these developments are tested with datasets from energy consumption measures from a sample of residential customers in Spain. The results obtained will be validated by the application of specifically developed validity indices, and the overall results will be analyzed.

12

Application of the dynamic clustering framework to load profile time series from residential customers and results

12.1 Introduction. The GAD project

The different analyses and algorithms developed have been tested on a database of load profiles provided by the Spanish DSO Iberdrola Distribución Eléctrica S.A.U. for the GAD project. This project had as the main objective to optimize energy consumption among medium and low voltage users by the research and development of new tools for the DSM, smoothing the peaks of energy demand, therefore enhancing the conditions of transport and distribution networks and the quality of energy delivery and service. The GAD acronym stands for the term "Active Demand Management" in Spanish.

The project had a duration of four years (2007 - 2010) and was supported by the CDTI (Technological Development Centre of the Ministry of Science and Innovation of Spain), and financed by the INGENIO 2010 program, under the CENIT research call. The consortium of the project had Iberdrola Distribución Eléctrica S.A.U. as the leader partner, and the following companies: Red Eléctrica de España (REE), Unión Fenosa Distribución, Unión Fenosa Metra, Iberdrola, Orbis Tecnología Eléctrica, ZIV Media, DIMAT, Siemens, Fagor Electrodomésticos, BSH Electrodomésticos España, Ericsson España, GTD Sistemas de Información, Grupo Foresis - Acceda Mundo Digital and Airzone.

As part of the developments addressed in the project, one of the objectives, as part of the tasks defined in the description of work, was the classification of users according to their patterns of daily energy load profiles, with the aim to detect different consumption patterns which may be of interest from the point of view of the GAD (and the DSM) objectives, i.e.: the average consumption per day (low, medium, high), the number of consumption peak periods and when these are produced, and an estimation of potentially manageable energy consumption.

12.2 Database description

The database of load profiles of energy consumption provided by the GAD project is formed by tables with 24 measures of hourly energy consumption per day, expressed in Wh, for a sample of clients that have been monitored during more than two years (2008 and 2009). These data have been divided in subsets according to seasonality (four seasons) and the type of day (working / non-working day), obtaining eight subsets of data per year. Each daily load profile is structured as a row with 24 measures or dimensions of energy consumption, with a set of qualitative and quantitative indices added. These indices describe the characteristics of the user and the load profile, and can be divided in three categories:

- Information on the client's contract and geographical situation. These variables comprise: a unique code to identify the user, the climate area (from a number of eight different climate areas in Spain), and five qualitative variables, computed from the consumption values of the electricity bill. Three of them determine the level of energy consumption throughout the year (High consumption, Medium consumption and Low consumption), and two qualitative variables define the increment of energy consumption in Winter (Winter peak) or in Summer (Summer peak). The five variables take only two possible values: zero (not applicable) or one (applicable).
- Quantitative information on the load profile. A number of indices is computed and added to each sample. The information describes the load profile in a set of variables: daily average consumption per hour (in Wh), maximum and minimum values of energy consumption per hour, the time of the maximum consumption, the peak/valley mathematical relation, the number of consumption peaks, and an index of peaks pattern, obtained as a six-bits digit where each bit stands for a different section of the day (early morning – breakfast time – morning – lunchtime – afternoon – dinner / evening). If a bit is set to one, it means that at least one consumption peak period is given at that section of the day.
- Quantitative and qualitative information obtained from questionnaires. The sample of clients were requested to answer a questionnaire regarding the type of home, number of residents, appliances owned, and consumption habits. The results have been computed to assign values to certain indices. These indices comprise the active management possibilities (percentage of high, medium and low penetration appliances at home, estimation of potentially manageable power, expressed in kWh per week); the familiar and residential characteristics (number of children, characteristics of the building); and the concern with environmental issues, the renewable energies and the knowledge about the current fee of electrical consumption.

The database comprises hourly energy consumptions taken from smart meters of a number of selected customers in diverse regions of Spain. The data contain errors, noise and blank dates, which will need to be conveniently filtered and processed.

The main values used in this work are defined in Table 12.1. These data comprise the user ID, the 24 hourly energy consumptions per day, and season and day indices. Other information is also available, such as the climate area for each client, and a set of qualitative and quantitative indices added. These indices describe the characteristics of the user and the load profile, and were obtained by means of questionnaires, submitted to the population sample. These questionnaires ask the sample of clients about their home, type of building, number of residents, appliances owned, and consumption habits. The results have been computed to assign values to certain indices. These indices comprise the active management possibilities (percentage of high, medium and low penetration appliances at home, estimation of potentially manageable power, expressed in kWh per week); the familiar and residential characteristics (number of children, characteristics of the building); and the concern with environmental issues, the renewable energies and the knowledge about the current fee of electrical consumption. Although these data have not been used in the clustering process, they are available and can be obtained by the user ID, allowing further analyses regarding the users' status, geography and habits and the relation with the pattern of load profile. This work can be found in other articles, such as [71].

Table 12.1. Objects data set. Description of columns or variables.

Column number	**Description**	**Format**
1	User or client unique ID	String
2	Date	Date
3 to 26	Hourly energy consumption (Wh)	Numeric
27	Season (Summer or Winter only)	Numeric
28	Weekday	String
29	Weekday	Numeric (1 = Sunday)
30	Working day	Numeric (1 = working day, 2 = non-working day)

12.3 Sample of residential users used

The sample of users represents typical residential load curves from Spanish consumers, in the three representative groups of customers of residential energy consumption: the so called "type a", "type b" and "type c" clients. These categories have been defined in the Spanish legislation, and are divided according to the type of contracted tariff and expected load profile:

- The "type a" clients are those who have contracted 2.0 A or 2.1 A tariffs. These are low voltage ($<$ 1000 V) tariffs with no time-of-use (TOU) pricing. The 2.0 A tariff is for clients with a power consumption below or equal to 10 kW, and the 2.1 A tariff is for clients with a power consumption higher than 10 kW and below or equal to 15 kW.
- The "type b" clients are those who have contracted 2.0 DHA or 2.1 DHA tariffs. These are low voltage tariffs with time-of-use (TOU) pricing, in two daily periods: peak and valley. The 2.0 DHA tariff is for clients with a power consumption below or equal to 10 kW, and the 2.1 DHA tariff is for clients with a power consumption higher than 10 kW and below or equal to 15 kW. These TOU tariffs begun to be applied in Spain on the first of January, 2008.
- The "type c" clients are those who have contracted 3.0 A or 3.1 with low voltage measure tariffs. These are TOU tariffs for clients with a contracted power higher than 15 kW. Tariff 3.0 A is for low voltage clients ($<$ 1000 V) and tariff 3.1 is for high voltage clients ($\geq$ 1000 V). The defined periods for pricing are three: peak, valley and flat.

According to the information from the Spanish National Commission of Energy (CNE) from the year 2009, type a, type b and type c customer numbers in Spain were the following:

- Type a customers: 24.835.412 ($92,98\%$ from total)
- Type b customers: 1.165.001 ($4,36\%$ from total)
- Type c customers: 709.276 ($2,66\%$ from total)

Of a total of 26.709.689 customers. Adequate samples that represent in percentage the total population [351] can be computed applying the formula described in (12.1), where n, N, t, p and e stand for:

- n: is the sample size, that represents the population percentage.
- N: is the total population number, for a known, finite size.
- t: confidence interval parameter, obtained as a given value of confidence α. A usual value of α is $0,05$, or the 95% of confidence. For this value, the value of the t parameter equals $1,96$.
- p: is the expected proportion in the sample, i.e., $92,98\%$ in the case of type a, $4,36\%$ in the case of type b, and $2,66\%$ in the case of type c.
- e: is the expected error in the sample. A usual value is to assume an error in the sample of 3%, therefore a value of $e = 0.03$ has been used.

$$n = \frac{Nt^2p(1-p)}{(N-1)e^2 + t^2p(1-p)} \tag{12.1}$$

Annex E of this document describes the mathematical methods to determine an adequate sample size. Assuming the preceding values, the resulting sample sizes for the three types of clients are:

- Type *a*: sample of 279 clients.
- Type *b*: sample of 178 clients.
- Type *c*: sample of 111 clients.

The database analyzed in the present work is comprised of load profiles from 759 clients, being 711 of them of type *a*, 44 of type *b* and 4 of type *c*. From these results it can be concluded that the sample is adequate for clients of type *a*, and inadequate for clients of types *b* and *c*, if no other stratification variables are taken into account, such as the presence of sufficient samples from the different climate regions of Spain. However, this information is not used for the segmentation, but to extract conclusions from the results.

12.4 Selection of the number of clusters to be found

Regarding the number of clusters to be found, the present analysis has been performed using a number of 10 clusters. This choice for the number of clusters is based on a previous work from Benítez et al. [71], where clustering and classification techniques are applied on a data set of energy consumption daily load profiles and the *DB index* [207] is computed for a scope of cluster numbers.

The methodology applied is the following: given a data set of measurements of daily load profiles for the year 2008 from a sample of residential customers in Spain, the data are analyzed separately based on season of the year and working or non-working day, therefore $4x2 = 8$ different data sets are produced and analyzed. The values of the DB indices are computed on the raw, non-normalized data, for an incremental value of clusters, until the 8 sets of data converge to a similar value of the DB index. This is the resulting value of 10 clusters chosen by the authors, as can be seen in Fig. 12.1.

The value of 10 clusters is seen by the authors as a good representative value to capture the main groups of energy consumption users and also some small groups of customers with unexpected or unusual energy consumption profiles. However, further studies could be made in this sense, comparing the values from different cluster validity indices. For instance 6, 8, or 12 clusters may also be a proper selection. Since the objective of this thesis is to develop and compare different partitional clustering methods, a value of 10 clusters has been selected for the segmentation in all the experiments.

12.5 Application of the developments made

Following, the developments presented in the previous Chapter are tested with the data from energy consumption load profiles described.

Three different tests have been designed. The first test compares the results of applying a dynamic clustering algorithm on time series data with different

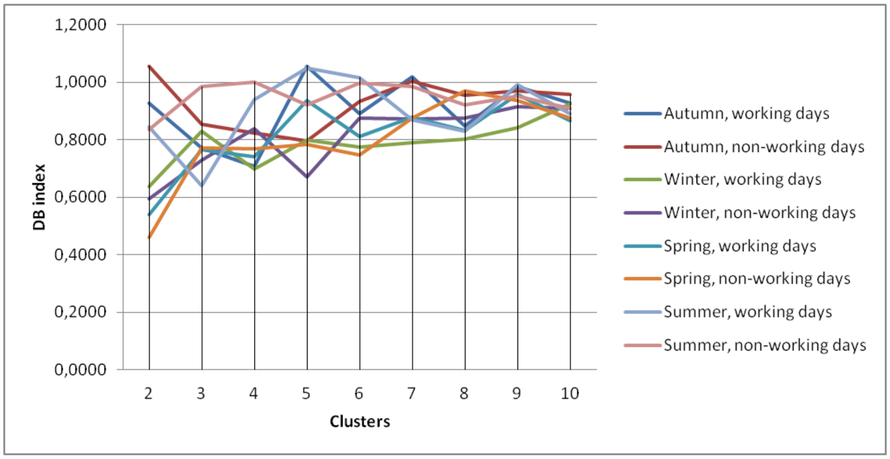


Fig. 12.1. DB indices for daily load profiles form year 2008

granularity. The dynamic K-means algorithm described in the first approach (Section 11.5.1) has been used for this analysis.

In the second test, the common framework for dynamic clustering developed and presented in Section 11.5.2 are tested and evaluated with two different time series data sets, one from energy consumption load profile values, and another with values generated synthetically. The objective of this analysis is to validate that, although developed specifically for the dynamic segmentation of load profiles, the work presented can be applied also to other data sets of different nature, given some conditions.

Finally the third test compares the results of the three developments presented in this thesis on the same data set of energy consumption values from two consecutive years. The dynamic K-means algorithm described in the first approach (Section 11.5.1), the common framework for dynamic clustering described in the second approach (Section 11.5.2) and the two-step dynamic clustering algorithm with a Hausdorff-based similarity distance described in Section 11.6, are applied on the same data set from time series load profiles from a sample of residential customers in Spain during the years 2008 and 2009.

12.5.1 First test: granularity options and dynamic k-means clustering. Analysis applied and results

The results from the first approach developed (Section 11.5.1) are presented in this Section. A dynamic clustering algorithm is applied on a database of daily load profiles from 759 clients during years 2008 and 2009. The objective of this analysis is twofold: on one hand, to classify the Spanish residential users by their dynamic daily load profile, evaluating the influence of the changes in normative and laws in the Spanish energy market that were produced in years 2008 and 2009. On the other hand, this work wants to present the possibilities of the dynamic clustering analysis, which can be a very useful tool that can be used by experts to classify groups of clients at a glance, detecting abnormalities in the energy load profile, evaluating the trends, and selecting target groups for DSM actions.

When addressing the dynamic segmentation of the daily load profiles database, different options of granularity can be applied, according to the objectives of the segmentation. Some of these options are, for instance, the following:

1. As n sequential daily load curves.
2. As cumulated or average n weekly or monthly load curves, to avoid weekly variations due to working days.
3. As n sequential daily load curves arranged by type of day, to avoid weekly variations due to working days (e.g., all the weekend daily curves only, all the working day curves only, etc.)

The three different arrangements of the load profiles' data have been performed and tested. Four different segmentation results are described, which are the following:

- Total sequence of daily load profiles through years 2008 and 2009.
- Sequence of working days daily load profiles through years 2008 and 2009.
- Sequence of non-working days daily load profiles through years 2008 and 2009.
- Sequence of cumulated monthly daily load profiles through years 2008 and 2009.

The non-working days data set in this analysis is comprised by all the Saturdays and Sundays of years 2008 and 2009, plus the holidays according to the Spanish calendar (e.g., the 1^{st} of May). The data set of working days includes all the other days.

Dynamic clustering on sequence of daily load profiles through years 2008 and 2009

The first analysis is the dynamic clustering performed on the sequential daily load profiles of the full sample of 759 clients through years 2008 and 2009, therefore the resulting clusters include 759 clients x (365 + 364) = 554.829 load profiles (the days March, the 30^{th}, 2008 and March, the 29^{th}, 2009 were lost due to the pre-treatment process of the data). The resulting centroids are depicted in Fig. 12.2.

Table 12.2 highlights some of the main values that can be obtained by observation of the resulting clusters. This information can also be easily extracted by means of mathematical functions or processes. The dynamic clustering indicates that most of the clients belong to a group of a low level of energy consumption (maximum consumption of around 1000 Wh) and an expectable pattern of valley and peak hours through the day (clusters 1, 3 and 10). Some clients present a higher level of consumption (clusters 2 and 5), and other clients are grouped under patterns of high energy consumption during the night, at the beginning of 2008, that have experienced a shift towards the first hours of the morning in 2009 (clusters 4 and 6). The reason for this shift might be mainly originated by the extinction of the nocturnal tariff in 2008, which would be used by clients to cumulate energy during the night, to be consumed during the day. The other clusters group other users with uncommon consumption patterns, such as clusters 7, 8 and 9. These patterns and the customers assigned to them should be object of a further study, in order to determine the nature of the consumption data, i.e. to discard outliers or erroneous measures as a first step, and then to study other possible reasons for the resulting pattern. For instance, clusters 7 and 8 present incomplete patterns of data, probably due to the absence of metering data during one of the two years analyzed. Besides, the consumption pattern of cluster 8 presents a more or less flat shape of elevated consumption during the day, indicating that the two consumers belonging to the cluster are probably a commerce or small enterprise.

Table 12.2. Clusters obtained from dynamic clustering on sequence of daily load profiles.

Cluster no.	**No. of clients**	**Maximum energy value (Wh)**	**time of maximum energy consumption (hours)**
1	175	$\simeq 1000$	$22-23$
2	3	$\simeq 5000$	$20-21$
3	476	$\simeq 500$	$22-23$
4	5	$\simeq 14000$	$00-01$
5	14	$\simeq 3500$	$22-23$
6	12	$\simeq 5000$	$00-01$
7	4	$\simeq 8000$	$21-22$
8	2	$\simeq 7000$	$14-15$
9	21	$\simeq 2500$	$23-00$
10	47	$\simeq 1300$	$22-23$

Dynamic clustering on sequence of daily load profiles through years 2008 and 2009, working days

The following analysis has been performed on the sequential daily load profiles, only for the working days through years 2008 and 2009, therefore the resulting clusters include 759 clients x 516 working days = 391.644 load profiles. The resulting centroids are depicted in Fig. 12.3.

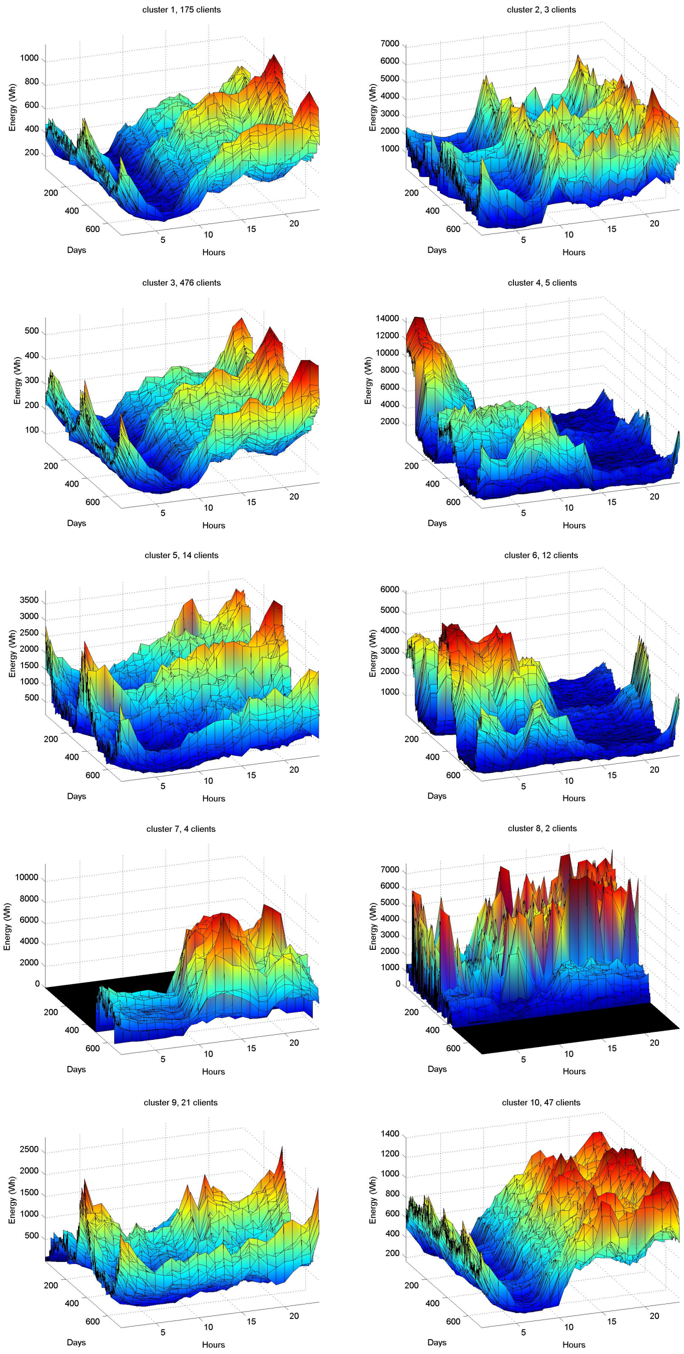


Fig. 12.2. Cluster prototypes from dynamic clustering on sequence of daily load profiles.

Table 12.3 indicates some of the main values from the resulting clusters, obtained as a first view of the cluster centroids. Again, this information could also be easily extracted by means of mathematical functions or processes. The dynamic clustering on the separate groups of load profiles, in working days and non-working days, has been performed in order to remove the known distinction in load profiles patterns between these two types of days. This way, the analysis of the differences found in patterns will not include the variability due to the sequence of working and non-working days through the year. As initial results from the observation of the centroids, similar results to the previous analysis that includes all the daily load profiles can be found. However, since no variability due to non-working days is present, the analysis yields also more clear differences in characteristic load profile patterns among the sample of clients: most of them belong to the group of low level of energy consumption and the usual load profile for residential users (cluster 7). Another group with a lower number of clients depicts the same usual pattern of load profile but a higher level of energy consumption, therefore being an appropriate objective for DSM actions (cluster 4). There are also groups with high energy consumption at nighttime (clusters 3, 9 and 10), and also clients with an unusual pattern of load profile and level of energy consumption (clusters 2 and 5). The seasonality effect is observed among all the patterns. Clusters 1 and 8 have grouped four clients with incomplete metering data. The reasons for this lack of data should be further investigated, in order to discard a failure in the metering infrastructure. In this sense, the development presented allows to quickly identify the AMI that should be reviewed for maintenance. However, the inclusion of users with lack of measurements represents also a source of noise for the dynamic clustering algorithm: since their pattern are very distinctive, the algorithm quickly assigns a new cluster for them, where they remain isolated through the algorithm's iterations. For this reason, in the following tests the users with lack of information have been removed, or the empty days have been filled with an average consumption obtained from the days with measurements.

Dynamic clustering on sequence of daily load profiles through years 2008 and 2009, non-working days

The following analysis has been performed on the sequential daily load profiles, only for the non-working days through years 2008 and 2009, therefore the resulting clusters include 759 clients x 213 non-working days = 161.667 load profiles. The resulting centroids are depicted in Fig. 12.4.

Table 12.4 indicates some of the main values from the resulting clusters, obtained as a first view of the cluster centroids. Load profile patterns from non-working days differ from working days in levels and hours of energy consumption. It can be seen that the main group of clients (cluster 5) displays a pattern of low level of energy consumption and a profile of peak and valley hours slightly different from that of working days. The second group with

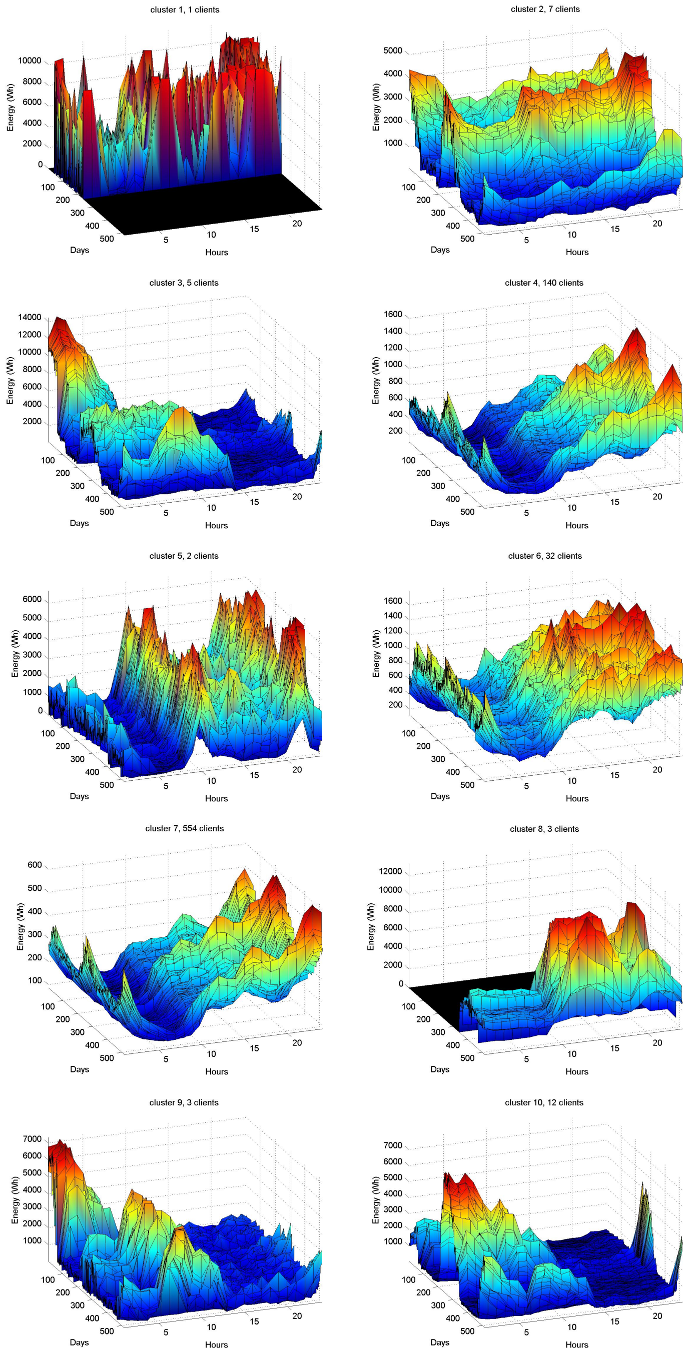


Fig. 12.3. Cluster prototypes from dynamic clustering on sequence of daily load profiles, working days.

Table 12.3. Clusters obtained from dynamic clustering on sequence of daily load profiles, working days.

Cluster no.	No. of clients	Maximum energy value (Wh)	time of maximum energy consumption (hours)
1	1	$\simeq 10000$	$01, 11, 22 - 23$
2	7	$\simeq 5000$	$22 - 23$
3	5	$\simeq 14000$	$00 - 01$
4	140	$\simeq 1500$	$22 - 23$
5	2	$\simeq 5000$	$10, 22 - 23$
6	32	$\simeq 1600$	$22 - 23$
7	554	$\simeq 600$	$22 - 23$
8	3	$\simeq 11000$	$11 - 16$
9	3	$\simeq 7000$	$00 - 01$
10	12	$\simeq 6000$	$00 - 01$

the highest number of clients (cluster 10) has a higher consumption of energy through the day, and the peak hours after lunch approximate in energy consumption those of the energy consumed after dinner. The group of users with a high level of energy consumption at night is also observed (cluster 8), but also some patterns are obtained, of clients that consume energy at an approximately constant rate through the day (clusters 3, 4, 6 and 9).

Table 12.4. Clusters obtained from dynamic clustering on sequence of daily load profiles, non-working days.

Cluster no.	No. of clients	Maximum energy value (Wh)	time of maximum energy consumption (hours)
1	34	$\simeq 1500$	$22 - 23$
2	9	$\simeq 2000$	$21 - 22$
3	10	$\simeq 4000$	$00 - 01$
4	9	$\simeq 4000$	$22 - 23$
5	499	$\simeq 600$	$22 - 23$
6	3	$\simeq 5000$	$22 - 23$
7	53	$\simeq 1400$	$22 - 23$
8	11	$\simeq 8000$	$00 - 01$
9	1	$\simeq 10000$	$11 - 16$
10	130	$\simeq 1000$	$22 - 23$

Dynamic clustering on sequence of cumulated monthly daily load profiles through years 2008 and 2009

The last of the analyses performed has been done on the aggregated or cumulated hourly energy used at the end of each month, from all the customers, for

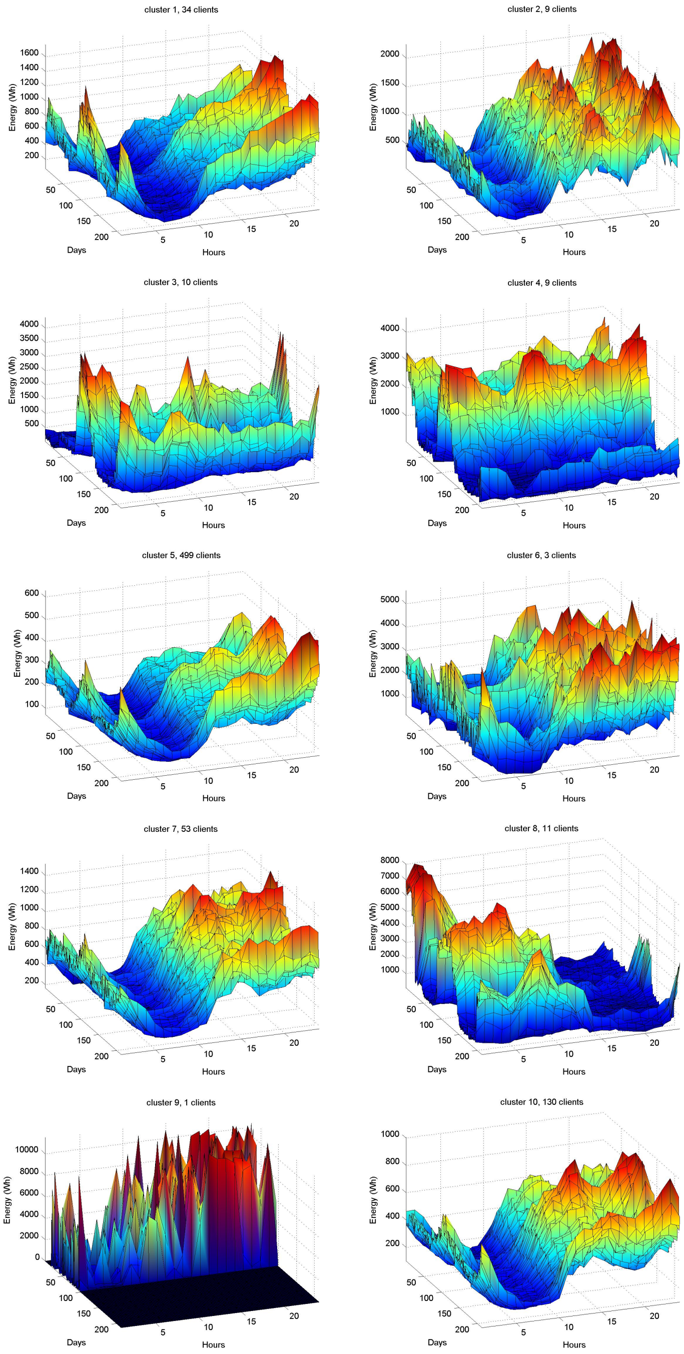


Fig. 12.4. Cluster prototypes from dynamic clustering on sequence of daily load profiles, non-working days.

years 2008 and 2009, therefore the resulting clusters include 759 clients $x24$ months = 18.216 load profiles. The resulting centroids are depicted in Fig. 12.5.

Table 12.5 indicates some of the main values from the resulting clusters, obtained as a first view of the cluster centroids.

From the results it can be seen that there are 2 clusters with the majority of clients that display the typical load profile of peaks and valley hours, with slight differences among them, and different levels of average energy consumption. These would correspond to clusters 7 and 8. In clusters 4 and 6, other type of electricity consumption patterns can be observed, with a more clear trend on an increase in consumption through the day. Clusters 2 and 9 gather a reduced group of users (10) with a high rate of energy consumption during the night. Besides, a shifting trend can be observed in energy consumption in the group of users from cluster 9, from a peak in the first hours of the day in the first months of 2008, to a more distributed consumption pattern in the last months of 2009. Finally, clusters 1, 5 and 10 represent 45 users with different patterns of energy consumption. A further analysis by an expert could be made to determine the nature for the obtained patterns.

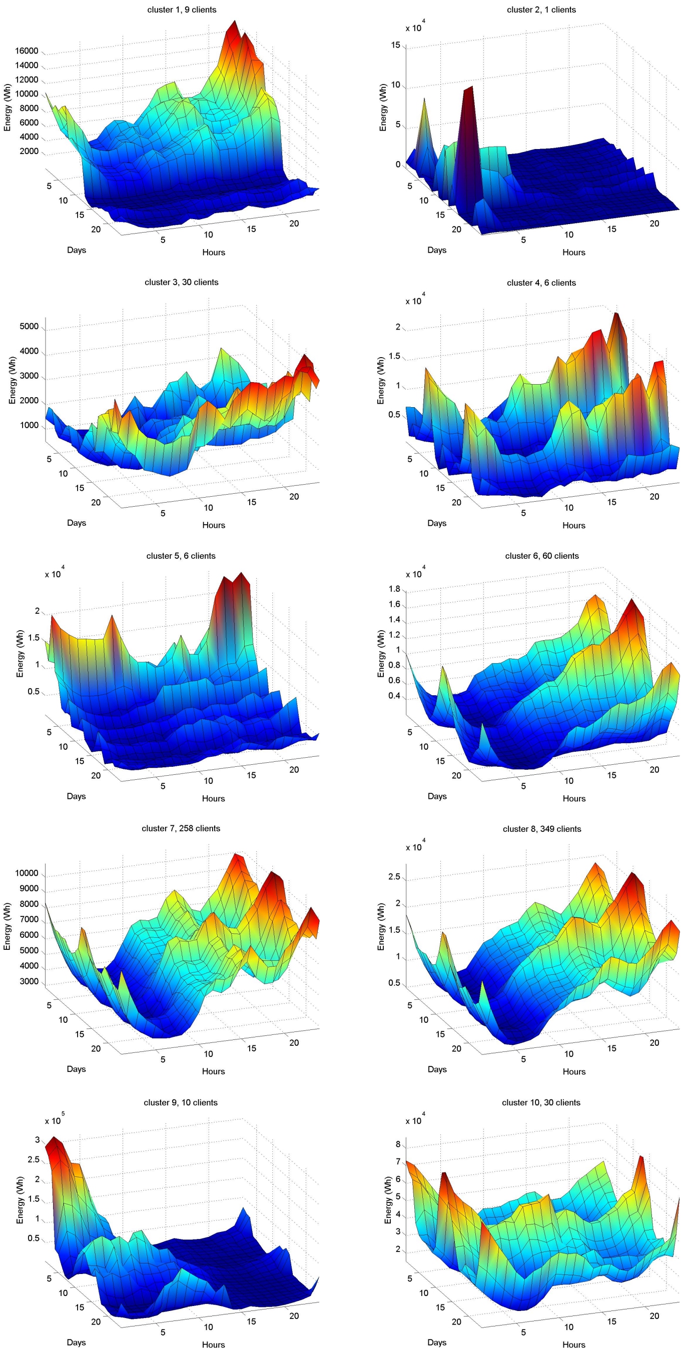
Table 12.5. Clusters obtained from dynamic clustering on sequence of cumulated monthly daily load profiles.

Cluster no.	**No. of clients**	**Maximum energy value (MWh) x 1 month**	**time of maximum energy consumption (hours)**
1	9	$\simeq 16$	$21-22$
2	1	$\simeq 150$	$01-02$
3	30	$\simeq 5$	$21-22$
4	6	$\simeq 20$	$22-23$
5	6	$\simeq 20$	$22-23$
6	60	$\simeq 18$	$21-22$
7	258	$\simeq 18$	$22-23$
8	349	$\simeq 25$	$22-23$
9	10	$\simeq 300$	$00-01$
10	30	$\simeq 300$	$23-00$

Conclusions

Regarding the first approach (Section 11.5.1), the same dynamic clustering analysis, based on the K-means algorithm, has been performed on 4 data sets of the same time series, a database of electricity consumption from residential consumers in Spain. The following conclusions can be drawn from these analyses:

- The results obtained in the four cases allow a fast identification of the main types of energy consumption patterns in the group of Spanish residential


cluster 1, 9 clients

cluster 2, 1 clients

cluster 3, 30 clients

cluster 4, 6 clients

cluster 5, 6 clients

cluster 6, 60 clients

cluster 7, 258 clients

cluster 8, 349 clients

cluster 9, 10 clients

cluster 10, 30 clients

Fig. 12.5. Cluster prototypes from dynamic clustering on sequence of cumulated monthly daily load profiles.

users of electric energy. This distinction can be more easily observed in the first three analyses, rather than in the cumulated monthly values of the energy consumed by each hour. Three main types of energy consumption users have been identified, which are explained next.

- The first type of client gathers the majority of the users in the sample (around 700 clients), and also represent the common pattern of energy consumption in domestic or residential users in Spain. It is represented by $2-4$ clusters in each of the four analysis, which represent a daily profile with three ascending peaks of energy consumption: one in the morning (around 8), another one at lunch (around 15 h.) and the highest one at night, around 22 h. The dynamic clustering groups these clients according to the shape of these peaks and the level of energy consumption, typically representing clients of low and medium energy consumption ($500-1.500$ Wh maximum). These results can be observed in the four analyses, but better in the first three. In the first analysis, for instance, this group is represented by clusters 1, 3 and 10.
- The second type of clients represents a minority of users with a high level of energy consumption through the day. These clusters can also be observed in the four analyses, however it is best observed in the first three. The clusters obtained have two main different shapes: one with the typical shape of energy consumption, described above, but with higher energy levels (from 2.500 to 7.000 Wh), for instance clusters 2, 5, 7 and 8 in the first analysis; and another group of users that present a (more or less) flat shape of energy consumption through the day (including the night hours), for instance cluster 9 in the first analysis.
- The third type comprehends clients with a higher consumption of energy at night. Examples of these clusters in the first analysis are clusters 5 and 6.
- Finally, in the first three analysis, there can be observed that the clustering process has identified 1 client with an anomalous pattern of energy consumption for residential use, with values of energy consumption that reach up to 10.000 kWh through daylight. In the first analysis, for instance, this client can be found in cluster 8. The dynamic clustering allows, therefore, a fast identification of anomalous or unexpected patterns of energy consumption.

The dynamic clustering analysis would be an efficient tool for clients' classification and trend behavior. The objective of DSM is one of the direct applications of this technique, however not the only one. Fraud detection could also be another possibility: a quick clustering of consumers in patterns will detect and highlight specific groups of clients whose level and profile of energy consumption may not suit the terms of their contracted power and energy tariffs.

The results of the analysis could be sufficiently representative of the type *a* residential customers in Spain, if no other strata differentiation are taken into account. If other stratifications are to be taken into account, however, such as for instance, the different climate regions in Spain, the idoneity of the clients analyzed to the stratified sample must be appropriately addressed.

From this analysis, an expert or operator should identify and classify objective clients. These decisions could be supported by decision support systems and an automated analysis of the resulting clusters, performing evaluation of trends, detection of anomalous behaviors, and automatically suggesting groups of clients for specific actions, such as commercial offers, or DSM orders for energy reduction. Prediction or load forecasting may also be combined with the dynamic clustering, to provide helpful estimation of patterns behaviors in medium term.

Regarding the granularity of the data, two conclusions can be obtained:

- The analysis of the raw data separated in working and non-working days may provide interesting information to the experts; however there is a loss of continuity when separating the analysis of the weeks in working days and non-working ones (mainly weekends). This loss of continuity provokes that the information regarding consumption habits of energy consumption for groups of customers is not complete. Therefore the clustering of the whole continuous sequence of days of raw data is preferred in this case.
- It is observed that when applying the dynamic cluster analysis on the cumulated monthly values there is a loss in the information of how the hourly energy consumption is produced, since the values are aggregated and the resulting profiles are smoothed. In this case also, the analysis of the raw data in form of hourly measures of energy consumption is preferred.

12.5.2 Second test: common framework for dynamic clustering. Analysis applied and results

Following, the results of the second approach developed (Section 11.5.2) are presented. The dynamic, or time series clustering algorithms with a fixed number of clusters, are obtained as extensions of some of the most used static clustering algorithms, such as the K-means [177] and Fuzzy C-means clustering [199][74], based on the dynamic clustering algorithm FFCM or Functional Fuzzy C-means, described by Oliveira and Pedrycz [90]. With these developments, time series of objects can be analyzed, and centroids are obtained that display the evolution in time of the objects they represent.

Following, this work is presented. The new algorithms developed have been tested in two different time series databases. One is the Synthetic Control Chart Series, a synthetic dataset obtained by Alcock and Manolopoulos [352], designed to test clustering techniques, and available at the University of California, Irvine (UCI) Machine Learning Repository; and the other is a record of a full year of daily energy consumption profiles from a number of residential customers from the GAD project database [66].

Example 1: applying dynamic clustering on synthetic time series data

The Synthetic Control Chart Series is a synthetic dataset obtained by Alcock and Manolopoulos [352], designed to test clustering and classification algorithms. It is available from the University of California, Irvine (UCI) Machine Learning Repository. This dataset contains 600 examples of control charts synthetically generated. There are six different classes or clusters of control charts:

1. Normal.
2. Cyclic.
3. Increasing trend.
4. Decreasing trend.
5. Upward shift.
6. Downward shift.

The dataset contains 100 examples of each class, each one comprised of 60 values of a time series process output. The six classes and the examples can be seen in Fig. 12.6. To perform the dynamic clustering on this dataset, the data has been rearranged, forming each object as a set of 10 features or dimensions, each with 60 time series values. The resulting six classes must, therefore, be formed of 10 objects each, with 10 features or dimensions, and 60 time instants. This rearrangement can be appreciated in Fig. 12.7.

Following, the results obtained when applying each different technique of dynamic clustering to this dataset are presented and commented. All the tests have been performed with an initialization of the centroids with random values.

Figure 12.8 depicts the resulting patterns and the number of objects from the dataset in each cluster, when applying the FFCM algorithm. The assignment in clusters does not match the expected result (ten objects per cluster). In particular, a mess of objects with different trends is observed in clusters 1 and 6. The other 4 clusters represent ascending trend (clusters 2 and 5) and descending trend (clusters 3 and 4). Even though there are some normal and cyclic objects present in the 4 clusters, the difference between the trend (clusters 2 and 4) and the shift (clusters 3 and 5) can still be appreciated.

Figure 12.9 depicts the resulting patterns and the number of objects from the dataset in each cluster, when applying the dynamic K-means with Euclidean distance algorithm (END-KME). In this case, two issues can be highlighted:

1. The algorithm has identified two clusters with a cyclic pattern (clusters 1 and 6). Clusters 1 and 5 have only one object.
2. The cluster of the decreasing trend objects has disappeared. Instead, both decreasing patterns have merged into one cluster (cluster 3), with 20 objects.

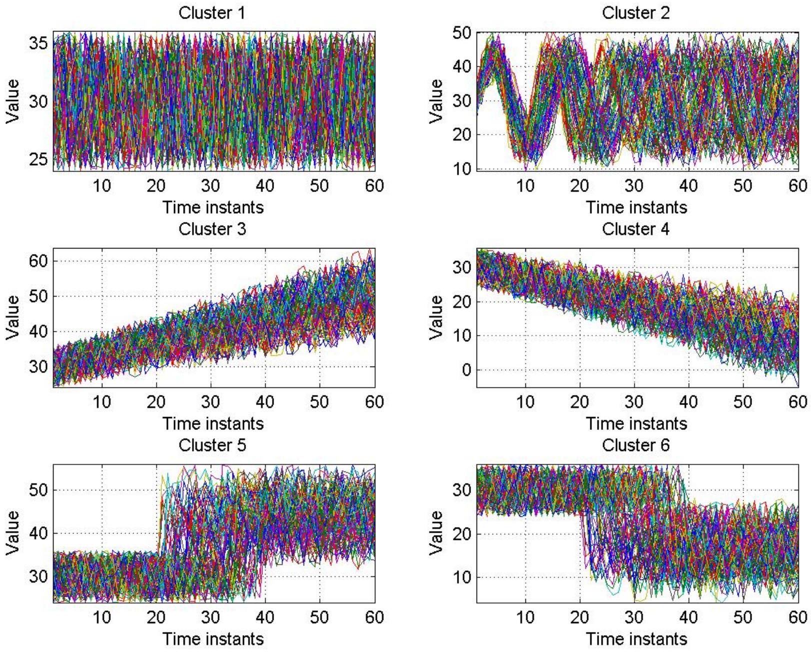


Fig. 12.6. The six classes in the UCI Synthetic Control Chart Series.

In this case one single object with an unexpected pattern (cluster 1) has, therefore, altered the partition of the data.

In Fig. 12.10 the END K-means with correlation-based distance algorithm (END-KMC) has been applied. In this case, similar results to that of the END-KME has been obtained.There are two clusters with a cyclic pattern (clusters 4 and 5) and only one with a decreasing trend (cluster 2), with 20 objects assigned.

Concerning the dynamic clustering based on the FCM, Fig. 12.11 depicts the results when applying the END FCM with correlation-based distance (END-FCMC). In this case, the six patterns are also clearly differentiated, however the number of objects assigned to each one have a worse score than in the previous one. Cluster 1 for instance has 18 objects assigned, whereas cluster 4 has 6.

Finally, Fig. 12.12 depicts the results when applying the END-FCM with Euclidean distance (END-FCME). The best match of the preceding algorithms is obtained, as can be seen from the resulting patterns and the number of objects in each cluster. The six expected patterns (normal, cyclic, ascending trend, descending trend, upward shift, downward shift) can be clearly differ-

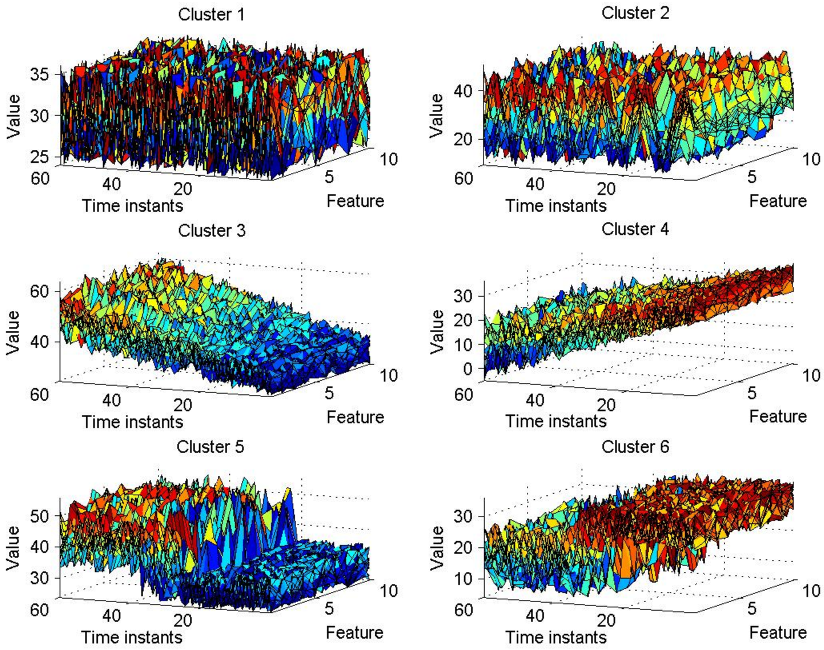


Fig. 12.7. Rearrangement of the UCI Synthetic Control Chart Series as a dataset with six classes of 10 objects each, with 60 time instants.

entiated, having each cluster 10 objects except clusters 4 (upward shift) and 6 (ascending trend), with 9 and 11 objects respectively.

As a conclusion from this analysis, it can be observed that the END-FCME has the best results of the five algorithms tested (see the summary of results in Table 12.6). However, these results should be analyzed in detail, and more tests should be performed. Following, the results on a completely different dataset are described.

Example 2: applying dynamic clustering on time series data from hourly electric energy measures from residential users

The database comprises hourly energy consumptions taken from smart meters of a number of 759 selected residential customers in Spain during the year 2009, with the objective to become a representative sample of the entire population of Spanish residential energy consumers. These data comprise the user ID, the 24 hourly energy consumptions per day, and season and day indices. The data to be clustered is comprised, therefore, of 759 objects, each

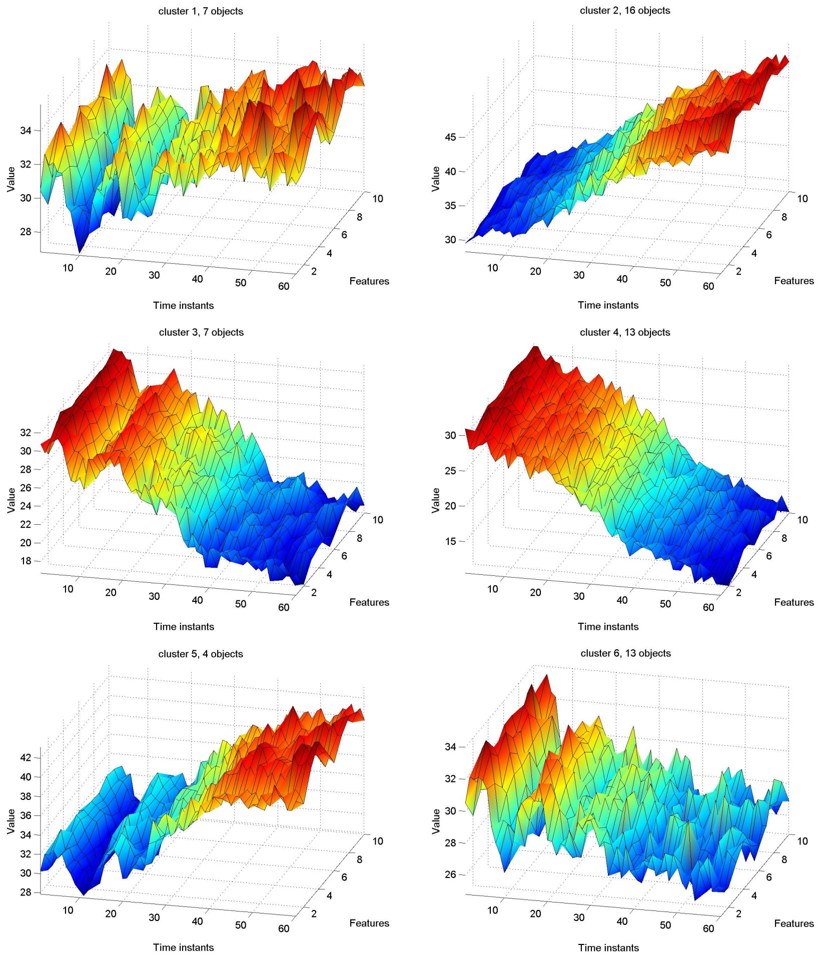


Fig. 12.8. FFCM cluster prototypes on the UCI Synthetic Control Chart Series.

one having 24 features or dimensions, and 364 daily measures (the day March, the 29^{th} was lost due to the pre-treatment process of the data).

In this case, the number of resulting clusters and how the partition should be is unknown. The expected results, however, should follow previous experiences concerning the classification of load profiles from residential or domestic electric energy users in Spain, such as the ones from the previous Section, or the classification results developed by Verdú et al. [130] and Benítez et al. [71], among others. The results should fit with the following description of the habits of consumption of this electricity consumption sector in Spain:

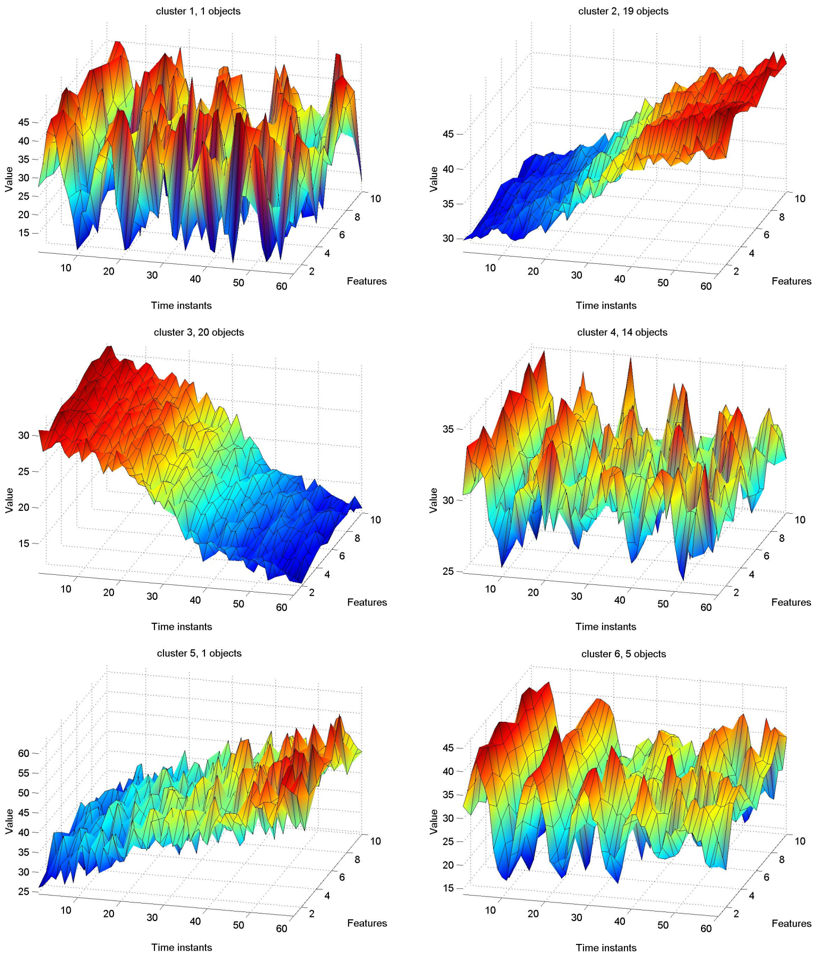
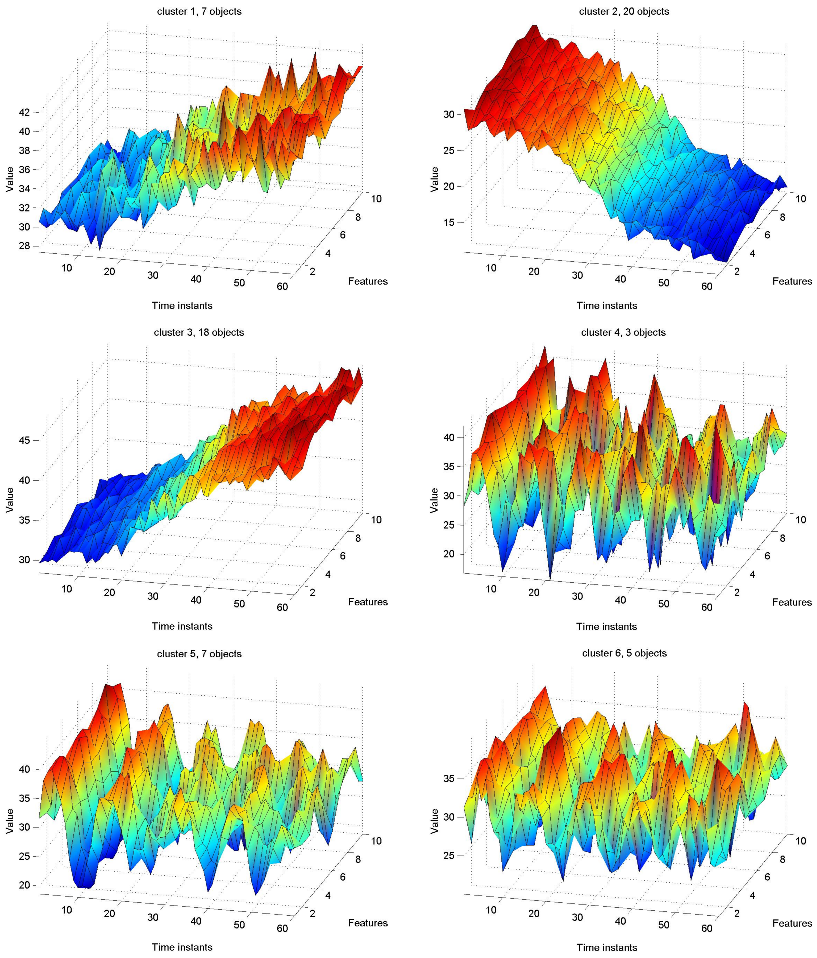


Fig. 12.9. END-KME clustering. Cluster prototypes on the UCI Synthetic Control Chart Series.

- The first type of clients represents the majority of energy consumption domestic or residential users in Spain. It is represented by a daily profile of energy consumption with three ascending peaks of energy consumption: one in the morning (around 8), another one at lunch (around 15 h.) and the highest one at night, around 22 h. The dynamic clustering groups these clients according to the shape of these peaks and the level of energy consumption, therefore this type of clients can be represented by two or more clusters that group clients of low and medium energy consumption (500 – 1.500 Wh maximum).



cluster 1, 7 objects

cluster 2, 20 objects

cluster 3, 18 objects

cluster 4, 3 objects

cluster 5, 7 objects

cluster 6, 5 objects

Fig. 12.10. END-KMC clustering. Cluster prototypes on the UCI Synthetic Control Chart Series.

- The second type of clients represents a minority of users with a high level of energy consumption through the day. There are two different patterns in this type of clients: one with the typical shape of energy consumption, described above, but with higher energy levels (from 2.500 to 7.000 Wh), and another group of users that have a (more or less) flat shape of energy consumption through the day, or other non-typical patterns of energy use.
- The third type comprehends a small group of clients with a higher consumption of energy at night, due to thermal energy accumulators that are used mainly at night, or in valley hours where the price of the energy is cheaper.

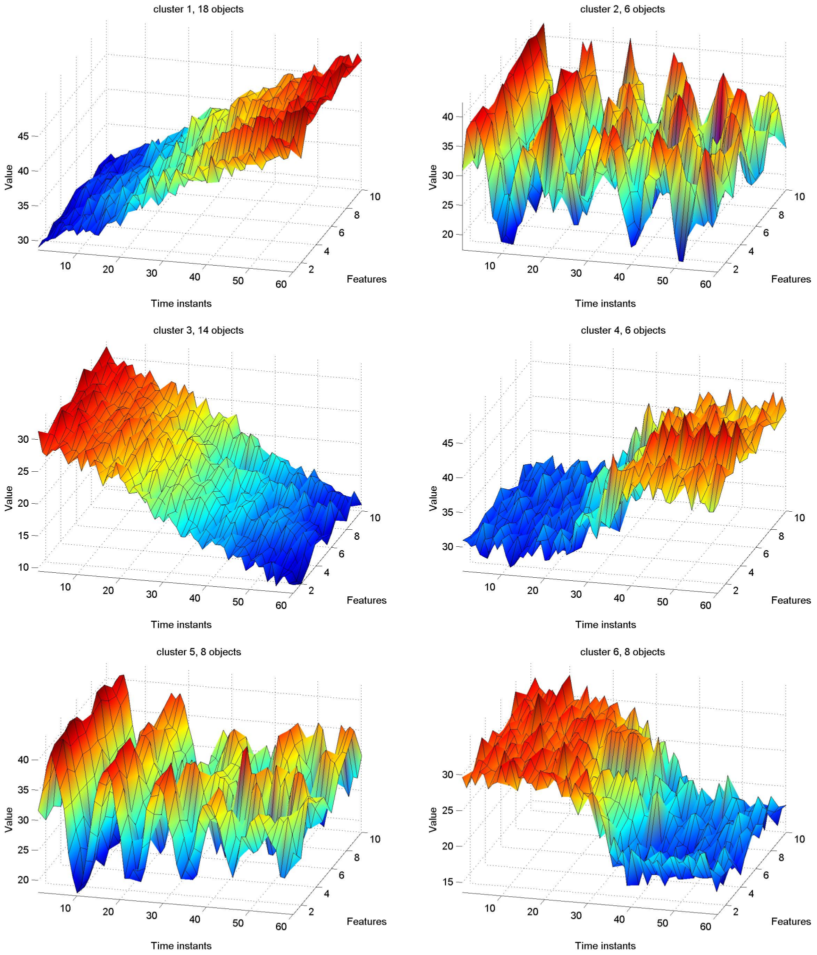


Fig. 12.11. END-FCMC. Cluster prototypes on the UCI Synthetic Control Chart Series.

Following, the results obtained when applying each different technique of dynamic clustering to this dataset are presented and commented. Regarding the number of clusters to be found, the present analysis has been performed using a number of 10 clusters, as explained before.

Figure 12.13 depicts the resulting clusters and clients assigned to each pattern of load profile when applying the FFCM. In this case, there are two predominant clusters with most of the clients assigned: cluster 9, which gathers 487 clients with a typical load profile pattern and a low level of energy consumption (up to 500 Wh), and cluster 5, which represents 125 clients with a typical load profile pattern and a higher (but yet low) level of energy con-

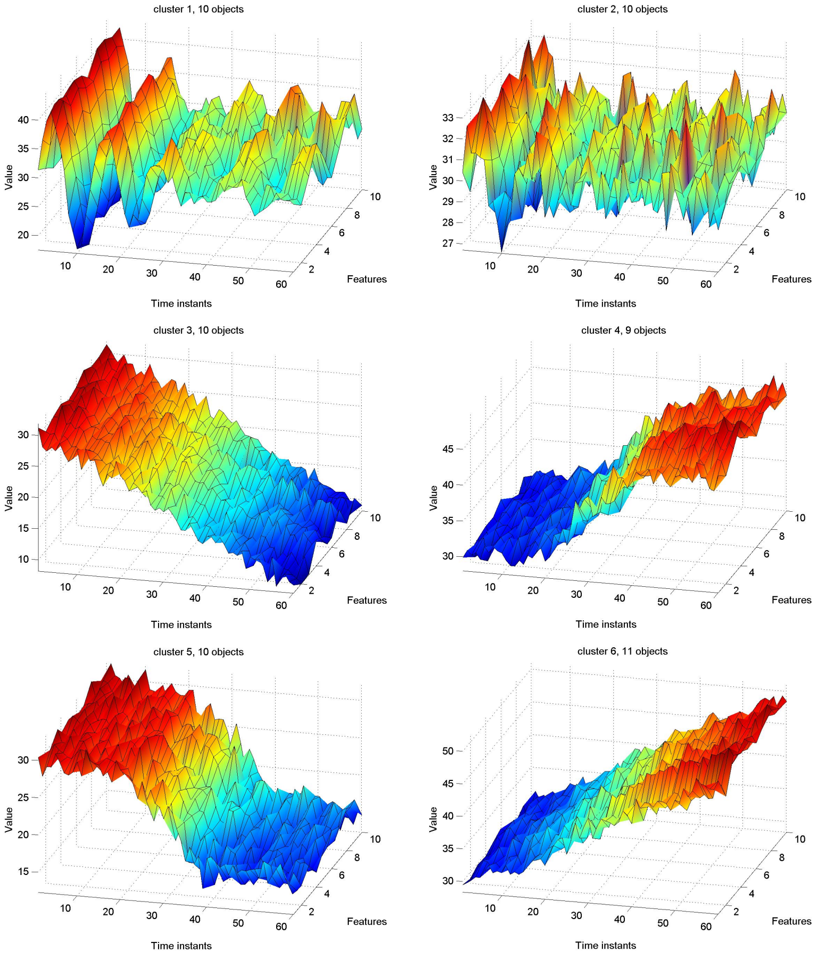


Fig. 12.12. END-FCME clustering. Cluster prototypes on the UCI Synthetic Control Chart Series.

sumption (up to 700 Wh). Clusters 1, 2, 6, 7, 8 and 10 gather clients with the typical load profile and a medium level of energy consumption (a total of 92 clients). Finally, two more clusters have been found. The two of them represent clients with a non-typical shape of load profile pattern, with a medium level of energy in one case (cluster 4) and with a high level of energy consumption in the second case (cluster 6). No clusters have been identified representing clients with a high level of energy consumption at night.

The last analysis is the END-FCMC clustering. The results can be observed in Fig. 12.14. In this case the results are visibly worse, since the

Table 12.6. Summary of results of dynamic clustering techniques applied on the UCI Synthetic Control Chart Series.

Algorithm	Objects in clusters	Comments
FFCM	$7-16-7-13-4-13$	mixed trends in some classes
END-KME	$1-19-20-14-1-5$	not proper partition
END-KMC	$7-20-18-3-7-5$	not proper partition
END-FCMC	$18-6-14-6-8-8$	Better results than the preceding ones
END-FCME	$10-10-10-9-10-11$	Best results obtained

partition depicts very similar patterns in the 10 clusters, with no relevant differences among any of them.

In Fig. 12.15, the results are displayed for the END-KMC clustering. This partition appears to be worse than the preceding one made by applying Euclidean distance. There is one main cluster that gather approximately half of the clients from the data set (cluster 4, 347 clients), displaying a pattern of typical load profile and a low level of energy consumption (700 Wh). The rest of the nine clusters gather (and mix) different types of clients, with resulting patterns of typical load profiles and other unusual shapes, all of them with medium and high energy consumption values. This result would be considered as a non-representative partition, since clients with different patterns and levels of energy consumption are mixed in the different clusters.

Figure 12.16 depicts the results of the END-KME clustering. In this case, the two main clusters have differentiated between low level of energy consumption (489 clients, cluster 10) and medium level of energy consumption (222 clients, cluster 1). Following, there is one cluster with clients with a typical pattern of energy consumption, but a higher level of energy use (cluster 3, 6 clients), and a group of clusters that gather unexpected patterns of load profiles and high or very high level of energy consumption. These are clusters 4, 5, 6, 7 and 8, with a total of 6 clients.Finally, there are two clusters that represent 19 clients with a high level of energy consumption at night, or at valley hours (clusters 2 and 9).

The results when applying the END-FCME clustering are depicted in Fig. 12.17. In this case, two main clusters can be observed in the results, clusters 1 and 2, representing the majority of the clients (695 in total), differentiated by the level of energy consumption (low in cluster 1 and medium in cluster 2), both clusters depicting the typical shape of the energy consumption load profile. Following, there are 5 clusters that group clients with high levels and unusual patterns of energy consumption. These clusters are 3, 5, 6, 7 and 8. Finally, three clusters that represent clients with energy consumption at night or at valley hours have been obtained (clusters 4, 9 and 10). The results of the clustering are similar to those obtained by the dynamic K-means clustering with Euclidean distance. The interpretation of the results is the prime key, therefore, to determine which method performs a better partition.

Table 12.7 summarizes these results. One of the first conclusions obtained from these analyses is that the correlation-based distance is not appropriate to measure similarity among irregular, non-linear objects in a database, therefore other similarity measures should be applied in this case. Concerning the idoneity of the method used, in both data sets the END-FCME and END-KME seem to perform better than the FFCM, being the END-FCME the algorithm that results in better, clearly differentiated patterns. The following Section, of conclusions and future work, describes the main findings from this work and the future tasks to be developed.

Table 12.7. Summary of results of dynamic clustering techniques applied on the dataset of daily load profiles from residential customers.

Algorithm	Objects in clusters	Comments
FFCM	$6-22-19-19-125-52-4-3-487-5$	mixed clients in some classes. Lack of classes
END-FCMC	$24-31-23-73-27-17-315-32-139-61$	not proper partition
END-KMC	$39-5-21-347-49-101-34-58-76-12$	not proper partition
END-KME	$222-16-6-1-1-1-2-1-3-489$	Better results than the preceding ones
END-FCME	$535-160-20-13-5-1-1-2-4-1$	Best results obtained

Conclusions

Going a step forward, concerning the results of the development presented in the second approach (Section 11.5.2), the best results are obtained with the END-FCM in both data sets (with Euclidean distance), then the END-K-means, and the last one being FFCM. However, the results highly depend on the similarity measure applied. The FFCM uses a fuzzy membership function, called “approximately zero”, to compute the similarity between two trajectories as the fuzzification of their difference. This function has configurable parameters that could be adjusted for each case, to test for a better performance of the clustering. The best results have been obtained, however, applying the Euclidean distance between pairs of feature trajectories of the dynamic objects. The correlation - based distance seems adequate to identify and classify objects with linear trends (such as in the UCI Synthetic Control Series data set), but does not work properly when classifying electricity load profiles, with non-linear shapes and different energy consumption levels.

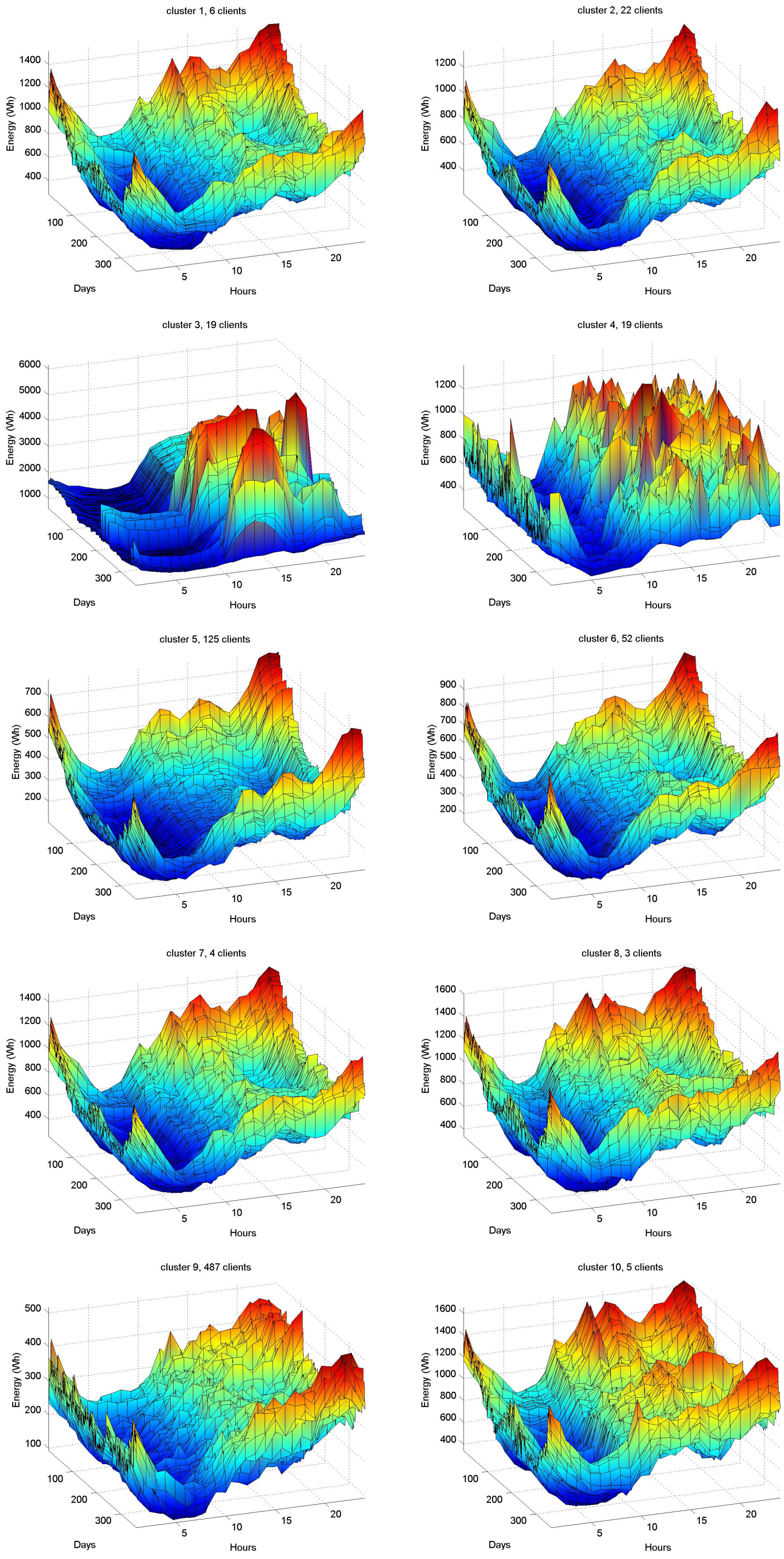


Fig. 12.13. FFCM cluster prototypes on the dataset of daily load profiles.

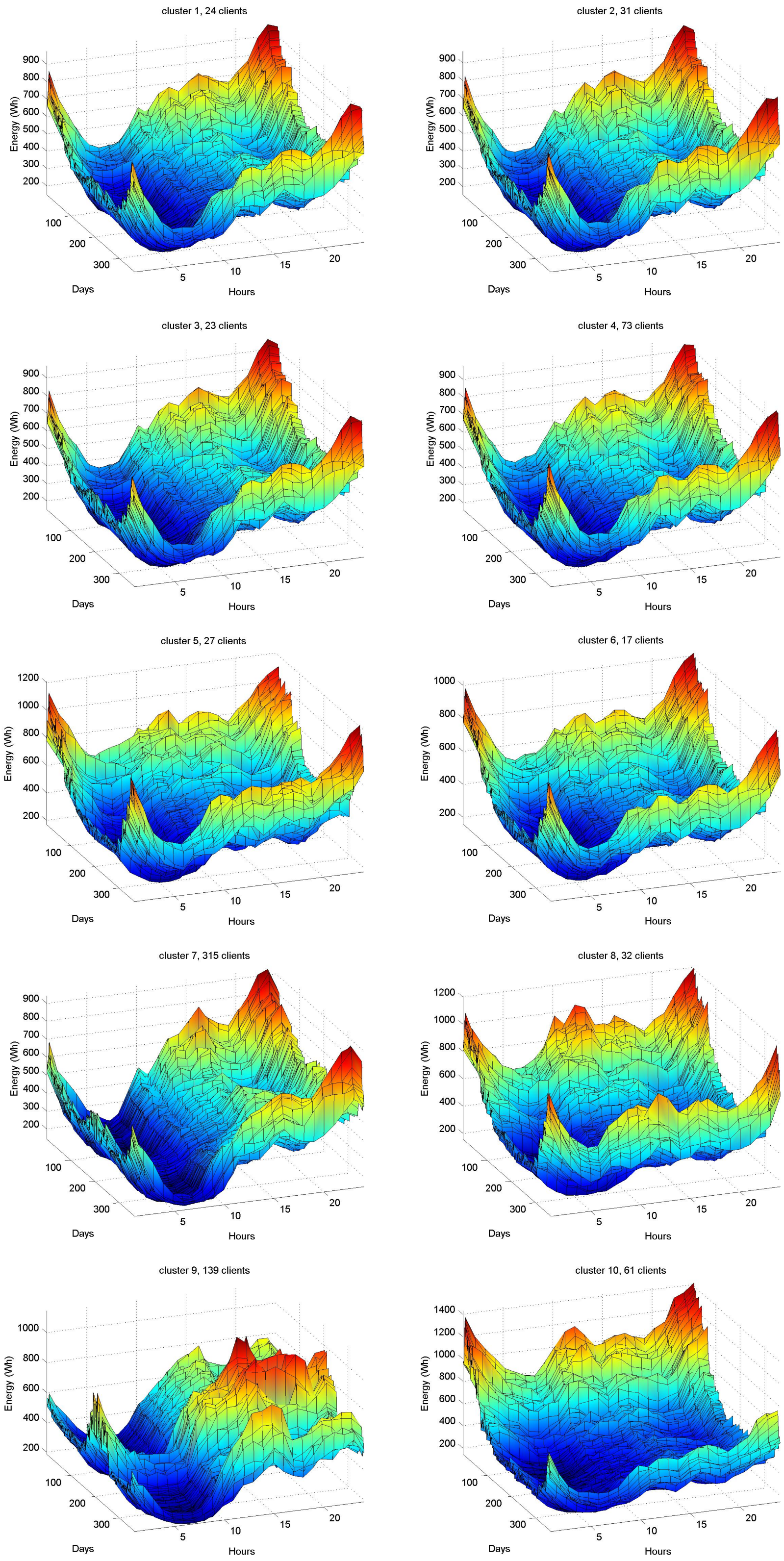


Fig. 12.14. END-FCMC clustering. Cluster prototypes on the dataset of daily load profiles.

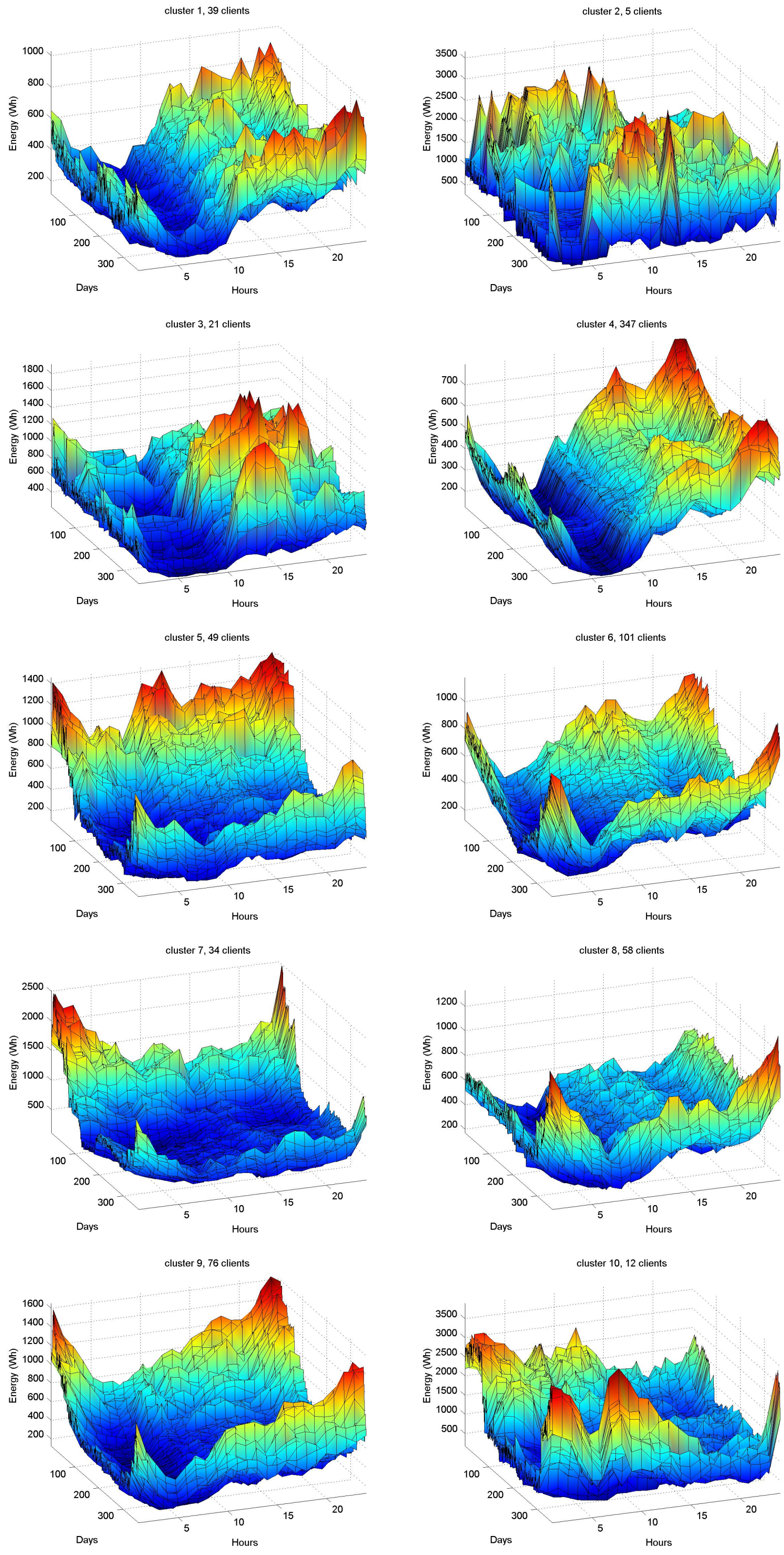


Fig. 12.15. END-KMC clustering. Cluster prototypes on the dataset of daily load profiles.

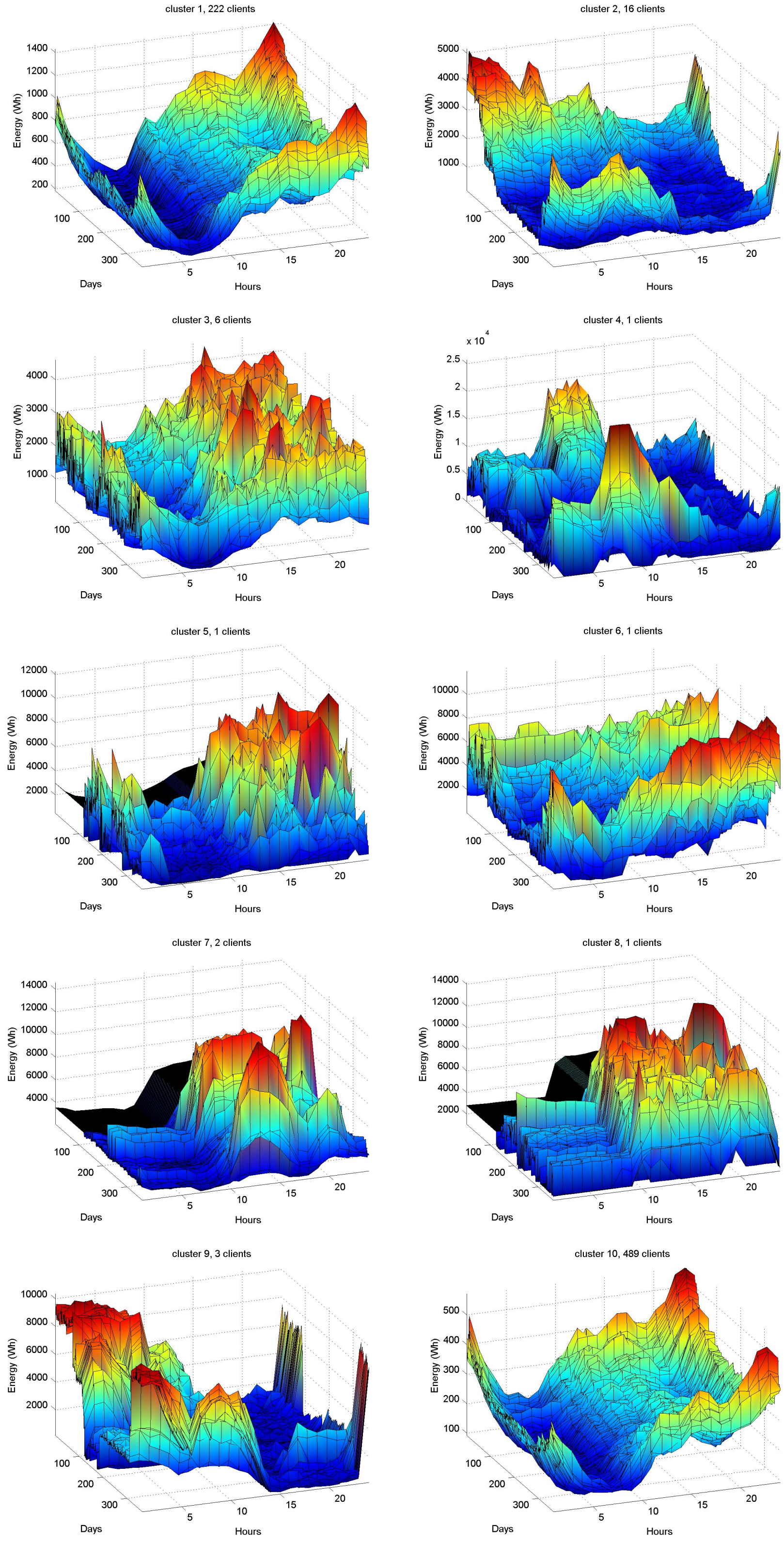


Fig. 12.16. END-KME clustering. Cluster prototypes on the dataset of daily load profiles.

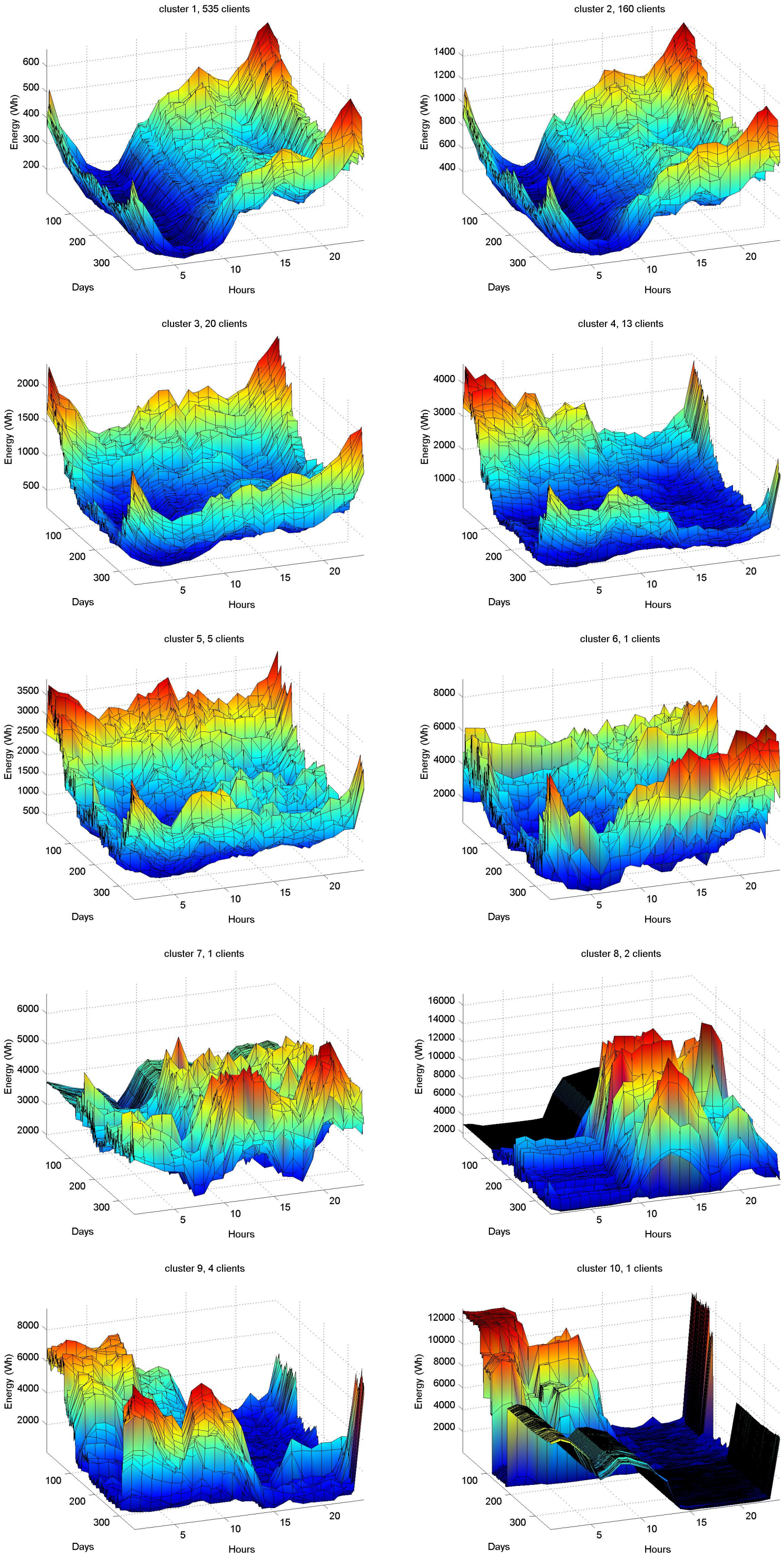


Fig. 12.17. END-FCME clustering. Cluster prototypes on the dataset of daily load profiles.

The idoneity of the resulting clusters is not easy to evaluate, due to the non-supervised learning nature of the clustering process. A further study is needed concerning the definition and test of validity indices for multidimensional time-evolving patterns, with time series variables or feature trajectories, allowing to compare different techniques for dynamic clustering.

The presented framework for time series clustering allows the evaluation of trends and cyclic behaviors. For instance, in the electric energy consumption dataset the seasonality effects can be observed. A pattern recognition procedure or expert system could automatically detect and distinguish the seasonal trends, for instance, and also detect abnormal situations, or to automatically classify and select the most appropriate objects for specific purposes.

In the data set of electric energy consumption, the clustering process has identified clients with an anomalous pattern of energy consumption for residential use, with values of energy consumption that reach up to 10.000 Wh through daylight. The END time series clustering framework allows, therefore, a fast identification of anomalous or unexpected patterns of energy consumption.

The practical applications of the dynamic clustering must be accompanied by a subsequent analysis of the clusters obtained, either by an expert or by a decision support system. Properly implemented in industrial processes, for instance, dynamic clustering can help in predictive maintenance tasks. In the case of electric energy consumption, the example presented can be a first segmentation analysis for Short Term Load Forecasting (STLF) ([327]) or to a Demand Side Management selection of client segments, as has been previously stated.

12.5.3 Third test: first approach of dynamic k-means clustering, common framework for dynamic clustering, and Hausdorff-based similarity measure dynamic clustering algorithm. Comparison and results

Following, all the three developments presented in the previous Chapter, in Sections 11.5.1, 11.5.2 and 11.6 are tested on the same data and the results are compared and evaluated by means of the clustering validity indices described in Section 11.4. The dynamic K-means algorithm described in the first approach (Section 11.5.1), the common framework for dynamic clustering described in the second approach (Section 11.5.2) and the two-step dynamic clustering algorithm with a Hausdorff-based similarity distance described in Section 11.6, are applied on the same data set from time series load profiles from a sample of residential customers in Spain during the years 2008 and 2009.

Data set used

The data set used in this analysis is obtained from the database of the GAD project described in Section 12.2. Two consecutive years of data, 2008 and

2009, have been used. After applying the data warehousing procedure described in Section 11.3, and eliminating those load profiles with noise and erroneous (disproportionately high) values, a final data set with 516132 load profiles from 708 clients is analyzed (i.e., hourly smart metering energy consumption values from 708 clients during 729 days). Figure 12.18 depicts these raw data in form of 24 hour load profiles.

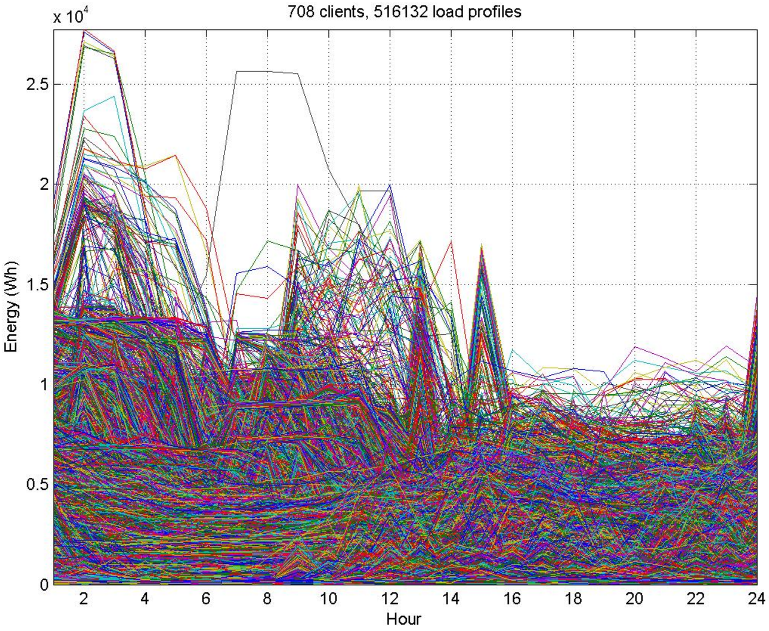


Fig. 12.18. Raw data being analyzed.

Selection of validity indices

The dynamic clustering validity indices described in Section 11.4, obtained as a modification of the original indices found in the literature, are the following: DB (Davies-Bouldin), SD (Scattering-Dispersion), PC (Partition Coefficient), XB (Xie-Beni) and FS (Fukuyama-Sugeno).

The five of them have been implemented in this analysis, and their resulting values are compared and evaluated. The SD index includes a parameter α as a weighting factor, whose value has been set to 0.5.

Algorithms applied and test procedure

As indicated before, The dynamic K-means algorithm described in the first approach (Section 11.5.1), the common framework for dynamic clustering described in the second approach (Section 11.5.2) and the two-step dynamic clustering algorithm with a Hausdorff-based similarity distance described in Section 11.6 are implemented and tested. As a result, the combination of the dynamic clustering algorithms described in Table 12.8 is tested on the same data set.

Table 12.8. Type 3 dynamic clustering algorithms tested.

No.	Static clustering technique	Dynamic objects similarity measure	Dynamic clustering algorithm
1	K-means	Euclidean distance	END-KME (present work)
2	K-means	Correlation	END-KMC (present work)
3	K-means	Hausdorff distance	END-KMH (present work)
4	K-means	Euclidean distance by the same time instant	Extended static clustering (present work, first approach)
5	FCM	Euclidean distance	END-FCME (present work)
6	FCM	Correlation	END-FCMC (present work)
7	FCM	Hausdorff distance	END-FCMH (present work)
8	FCM	Fuzzy membership functions	FFCM (described by Weber)

These 8 algorithms have been run 10 times each, and all the resulting clustering validity indices' values have been recorded. The number of clusters to be found is set to 10, as commented in a previous Section. The maximum number of iterations that each dynamic clustering algorithm is running until it stops (if no other convergence criterion is reached) has been also set to 10. The results are presented next.

Regarding the Hausdorff distance-based algorithms, the number of regions or surfaces to decompose the energy consumption shapes have been chosen based on previous knowledge of the behavior of the residential load profiles in Spain. The daily load profile has been divided in seven regions, according to the expected trends in a typical consumption profile of one day for the low voltage residential consumer: the first region is from 0:00 to 5:00 hours; the second from 5:00 to 8:00 hours, when the first peak of the morning is expected (getting up for going to work); the third from 8:00 to 10:00 hours, when the consumption decreases; the fourth from 10:00 to 15:00 hours, when the second peak of energy consumption is expected (due to lunchtime in Spain); the fifth is from 15:00 to 18:00 hours; when the energy consumption decreases again; the sixth is from 18:00 to 22:00 hours, when the third (and maximum) peak of energy consumption in Spanish dwellings is reached; and finally the seventh is from 22:00 to 24:00 hours. A typical profile with this behavior can be observed in the resulting cluster prototypes from the present analyses, for instance in

Fig. 12.17, cluster 1. The time axis has been divided in 24 sections equal in length, which would approximately correspond to the 24 months during two consecutive years.

Results

The following Tables display the results obtained for the clustering validity indices DB (Table 12.9), SD (Table 12.10), PC (Table 12.11), XB (Table 12.12) and FS (Table 12.13).

Table 12.9. Test results, DB modified index.

Cycle	END-KME	END-KMC	END-KMH	Extended K-means	END-FCME	END-FCMC	END-FCMH	FFCM
1	2.348	5.918	3.204	NaN	4.131	NaN	6.181	8.114
2	3.151	5.650	2.680	2.285	3.848	NaN	7.449	7.050
3	2.314	5.836	3.039	1.594	2.792	NaN	5.201	NaN
4	1.894	5.674	3.759	NaN	5.801	NaN	5.103	NaN
5	2.967	5.674	3.136	2.421	3.699	NaN	4.184	NaN
6	2.589	5.267	3.049	NaN	3.175	NaN	6.444	6.527
7	2.336	5.951	3.497	2.183	8.750	NaN	6.552	NaN
8	2.501	5.854	3.202	NaN	4.143	NaN	5.881	NaN
9	2.483	5.758	3.204	1.390	4.866	NaN	16.863	9.270
10	2.689	5.878	2.995	NaN	2.909	NaN	4.738	NaN

Table 12.10. Test results, SD modified index.

Cycle	END-KME	END-KMC	END-KMH	Extended K-means	END-FCME	END-FCMC	END-FCMH	FFCM
1	27.813	34.255	28.254	29.268	28.238	36.058	28.495	49.621
2	27.850	34.826	28.308	28.578	28.100	36.768	28.586	33.791
3	28.274	34.762	28.396	28.648	28.207	36.150	28.435	32.367
4	28.244	34.629	28.427	29.128	28.326	36.030	28.377	39.182
5	28.113	34.433	28.404	28.679	28.020	36.658	28.367	33.779
6	27.939	34.351	28.421	28.795	28.174	36.300	28.441	33.654
7	27.984	34.445	28.311	28.238	28.186	36.441	28.516	31.954
8	27.811	34.688	28.466	29.262	28.205	36.042	28.498	32.694
9	28.411	34.587	28.551	28.630	28.460	36.500	28.520	32.690
10	28.134	34.701	28.151	28.435	28.260	36.064	28.405	33.387

As described in Section 11.4, good values for a segmentation should be as low as possible for the DB, SD, XB and FS indices. For the PC index, a good partition would have a value closer to 1.

Table 12.11. Test results, PC index.

Cycle	**END-KME**	**END-KMC**	**END-KMH**	**Extended K-means**	**END-FCME**	**END-FCMC**	**END-FCMH**	**FFCM**
1	1.000	1.000	1.000	2.435	0.616	9.999	0.663	0.758
2	1.000	1.000	1.000	1.000	0.638	9.999	0.517	0.826
3	1.000	1.000	1.000	1.000	0.725	9.999	0.626	0.931
4	1.000	1.000	1.000	2.364	0.607	9.999	0.565	0.943
5	1.000	1.000	1.000	1.000	0.591	9.999	0.653	0.788
6	1.000	1.000	1.000	1.723	0.704	9.999	0.585	0.914
7	1.000	1.000	1.000	1.000	0.643	9.999	0.521	0.952
8	1.000	1.000	1.000	1.761	0.649	10.000	0.678	0.755
9	1.000	1.000	1.000	1.000	0.756	9.999	0.648	0.935
10	1.000	1.000	1.000	1.559	0.731	10.000	0.618	0.968

Table 12.12. Test results, XB modified index.

Cycle	**END-KME**	**END-KMC**	**END-KMH**	**Extended K-means**	**END-FCME**	**END-FCMC**	**END-FCMH**	**FFCM**
1	2.397	4.969	3.050	2.467	1.450	7783.078	1.645	4.637
2	2.650	5.049	2.609	1.355	1.657	16183.279	4.601	6.271
3	1.832	5.105	3.349	1.158	1.041	8734.723	2.226	3.004
4	1.478	5.261	4.362	2.665	2.266	7452.204	3.382	2.939
5	2.573	5.288	3.279	1.404	1.236	14888.034	2.253	4.250
6	2.565	4.180	3.399	1.768	1.061	10649.358	2.540	3.477
7	1.948	4.882	3.641	1.730	4.333	12319.869	4.034	1.763
8	2.765	4.425	4.569	1.494	1.825	7430.730	2.040	2.529
9	1.843	4.787	4.290	0.956	2.441	13012.834	12.000	3.067
10	2.593	5.670	3.631	2.552	1.185	7856.863	2.360	2.037

Table 12.13. Test results, FS modified index.

C.	**END-KME**	**END-KMC**	**END-KMH**	**Extended K-means**	**END-FCME**	**END-FCMC**	**END-FCMH**	**FFCM**
1	-12772470	4126311	-5209931	-36261842	-9073076	76304633	-4191469	-1802182
2	-11092386	3644888	-9591464	-12942781	-10473813	76308813	-1227209	-5699821
3	-14295512	4697885	-6526723	-20958214	-12694782	76306871	-3254040	-6290015
4	-15464305	4473610	-3964876	-34402772	-9492322	76299108	-2268683	-5705396
5	-12476157	3630603	-5754756	-10842871	-8789668	76307909	-3723585	-7980813
6	-9450688	1993840	-6763694	-31657597	-12254085	76308175	-2804311	-6285484
7	-12147622	4061213	-5397405	-15908953	-10585746	76304332	-1310244	-11532535
8	-11096014	4633430	-3561036	-32091774	-10463479	76307112	-4345782	-8207675
9	-11445730	4250248	-1188110	-21232201	-13191617	76309445	-3983810	-4838135
10	-11281278	4570829	-6895594	-23248168	-12765613	76306020	-3078600	-7342507

From the **DB index** values' Table (Table 12.9) it can be observed that the best results are obtained by the END-KME and the END-KMH algorithms, in all the 10 cycles. The worst result is obtained for the END-FCMC algorithm, where a NaN value in all the cycles indicates a division by zero or an error in numerical precision, probably due to the inability of the algorithm to produce well defined and separated clusters. The FCM-based algorithms, END-FCME and END-FCMH algorithms provide worse results than K-means. Finally, the FFCM and the Extended K-means yield NaN values in some cycles, therefore their proficiency would be less trustworthy than the clustering algorithms with no NaN values. It can be concluded that, regarding the DB index, K-means-based dynamic clustering algorithms with Euclidean or Hausdorff-based distances provide the best results.

Regarding the **SD index**, the results from Table 12.10 indicate, in accordance with results from the DB index, that the best partitions are provided by dynamic clustering algorithms with Euclidean or Hausdorff-based distances, whereas correlation-based algorithms provide bad results. This result may be induced by the non-linearity in the shapes of the load profiles, as has been indicated in the previous test. The FFCM algorithm provides a bad result too, with values in the same range as the correlation-based algorithms. In this case, however, it must be noted that both K-means and FCM based algorithms provide similar results, having therefore more influence the similarity distance used, rather than the clustering technique.

The Table with the results from the **PC index** (Table 12.11) provides expected results in most of the cases, and cannot be used to compare the validity of the different clustering algorithms. As can be seen from the Table, the K-means-based algorithms yield a value of 1 in all the cycles (values greater than 1 in the case of the Extended K-means algorithm are due to a NaN-prevention procedure, and should not be taken into account). Regarding the fuzzy-based clustering algorithms, in this case is the FFCM the one with the best results (closest to 1 values). Anomalous values can be observed in the case of the END-FCMC algorithm, which are a result of a similar NaN-prevention procedure implemented in the algorithm, and therefore the results from this algorithm cannot be evaluated.

Table 12.12, with the results from the modified **XB index**, display similar results: the best values are obtained with K-means or FCM Euclidean or Hausdorff-based distances. The FFCM the END-KMC provide worse results, and the END-FCMC is the worst of all. These results support the hypothesis, previously stated, that correlation is not a good similarity measure for the dynamic clustering of load profiles time series.

Finally, the results from the modified **FS index** (Table 12.13) are also aligned with the previous results. The worst values (positive values) are obtained by END-KMC and END-FCMC. Algorithms END-KMH, END.FCMH and FFCM provide better results (with negative values), and the best results are obtained by the END-KME, Extended K-means and the END-FCME algorithms.

The main conclusion that can be derived from the quantitative analysis of the different clustering validity indices, is that the best results are obtained with K-means or FCM based and Euclidean or Hausdorff-based distances (END-KME, END-KMH, END-FCME, END-FCMH). Following, the resulting patterns for each dynamic clustering algorithm are analyzed. In each case, the patterns with the best FS modified index from the 10 cycles have been chosen, however, as has been seen from the previous Tables, values from the validity indices obtained do not differ much along the 10 cycles for each algorithm, therefore any of the 10 cycles could have been used.

The resulting clusters and how the partition should be is unknown. The expected results, however, should follow previous experiences concerning the classification of load profiles from residential or domestic electric energy users in Spain [130][71]. There are mainly three types of energy consumption users according to their load profile patterns or prototypes, which are explained next.

- The first type of client represents the majority of energy consumption residential users in Spain. It is represented by a daily profile of energy consumption with three ascending peaks of energy consumption: one in the morning (around 8 h.), another one at lunchtime (around 15 h.) and the highest one at night, around 22 h. The clustering technique groups these clients according to the shape of these peaks and the level of energy consumption, therefore this type of clients can be represented by two or more clusters that group clients of low and medium energy consumption (500 - 1500 Wh maximum).
- The second type of clients represents a minority of users with a high level of energy consumption through the day. There are two different patterns in this type of clients: one with the typical shape of energy consumption, described above, but with higher energy levels (from 2500 to 7000 Wh), and another group of users that present a (more or less) flat shape of elevated energy consumption through the day, or other non-typical patterns of energy use.
- The third type comprehends a small group of clients with a higher consumption of energy at night, due to thermal energy accumulators that are used mainly at night, or in valley hours where the price of the energy is cheaper.

Following, the results from all the 8 dynamic clustering algorithms tested are displayed, in Figs. 12.19, 12.20, 12.21, 12.22, 12.23, 12.24, 12.25 and 12.26. In all the resulting patterns for each dynamic clustering algorithm, a study is performed in order to match the clusters obtained to one of the types mentioned. To do this, first the described types are categorized in the following five groups or labels:

1. Common profile of residential energy consumption, with low average of daily consumption (around 500 Wh).

2. Common profile of residential energy consumption, with medium average of daily consumption (around 1500 Wh).
3. Uncommon profile, with the typical shape but with elevated average or maximum of daily energy consumption (from 2500 to 7000 Wh).
4. Uncommon profile, more or less flat through the day (or other non-typical shape) and with elevated average of daily energy consumption.
5. Peak consumption mainly at night, or shifted to other valley hours.

Tables 12.14, 12.15, 12.16, 12.17, 12.18, 12.19, 12.20 and 12.21 display this assignment. Either the results from the Tables and the visualization of the obtained patterns support the conclusions obtained in the analysis of the clustering validity indices: K-means or FCM - based, with Euclidean or Hausdorff - based distances provide the best defined and well-balanced clusters and patterns. As can be seen in the results from Table 12.19, the END-FCMC algorithm is not able to discriminate users' profiles by their correlation measure, therefore only one cluster with a fuzzy membership of 100% from all the users has been created.

Table 12.14. Assignment of clusters from END-KME to expected groups.

Group	No. of clusters	No. of clients
1	1	439
2	2	239
3	1	5
4	3	5
5	3	20

Table 12.15. Assignment of clusters from END-KMC to expected groups.

Group	No. of clusters	No. of clients
1	3	473
2	1	121
3	0	0
4	4	96
5	2	8

Conclusions

The development made has been tested with a data set of two consecutive years of hourly measures of energy consumption. The results show that a selection of the algorithms developed provide an appropriate segmentation in temporal patterns, either visually and numerically. The quantitative analysis,

Table 12.16. Assignment of clusters from END-KMH to expected groups.

Group	No. of clusters	No. of clients
1	2	408
2	2	225
3	1	37
4	2	5
5	3	33

Table 12.17. Assignment of clusters from Extended K-means to expected groups.

Group	No. of clusters	No. of clients
1	1	528
2	1	144
3	1	8
4	3	4
5	4	24

Table 12.18. Assignment of clusters from END-FCME to expected groups.

Group	No. of clusters	No. of clients
1	1	473
2	3	210
3	0	0
4	1	3
5	5	22

Table 12.19. Assignment of clusters from END-FCMC to expected groups.

Group	No. of clusters	No. of clients
1	1	708
2	0	0
3	0	0
4	0	0
5	0	0

Table 12.20. Assignment of clusters from END-FCMH to expected groups.

Group	No. of clusters	No. of clients
1	2	430
2	3	215
3	2	34
4	0	0
5	3	29

Table 12.21. Assignment of clusters from FFCM to expected groups.

Group	No. of clusters	No. of clients
1	2	657
2	4	43
3	2	4
4	1	1
5	1	3

by means of specifically modified clustering validity indices, and a qualitative study, by observing the resulting patterns ad assigning them to one of five groups defined for typical profiles of residential energy consumption in Spain, are coincident in their results: the K-means or FCM - based algorithms, with Euclidean or Hausdorff - based distances, provide the best defined and well-balanced clusters and patterns.

The Extended K-means algorithm, however, has not provided a similar performance in all the cycles and does not provide a good segmentation of the customers. As can be seen from the resulting patterns, the algorithm is mixing profiles with different trends, since only the values from each dimension are compared at each instant of time. The resulting features are not dynamic trajectories, but an alignment of independently computed distances. Therefore this algorithm, as stated previously, is not a good option for dynamic clustering.

Correlation is not a good similarity measure for the dynamic clustering of load profiles time series either, as stated previously, probably due to the non-linearity in the features or dimensions' trends. This measure, however, can be appropriate for other data sets, as has been described in he second test.

The FFCM algorithm obtains worse results than Euclidean or Hausdorff-based distances. The reasons for these results could be in the definition of the membership function used to define the proximity between features.

12.6 Conclusions from the tests

Seasonality effects can be clearly observed, with generalized higher consumptions in winter, lower energy consumptions in summer, and the lowest values from autumn and spring. It is also interesting to analyze whether changes in the Spanish legislation of the energy market have affected the way residential clients make use of the energy; these changes would be reflected along the dynamic patterns of the resulting centroids. However, no significant change is observed. From July, the first, 2009, low voltage users with contracted power equal or below 10 kW were given the choice to either be charged under a specific, fixed tariff (TUR), or to contract the energy with an authorized retailer in a liberalized energy market. Low voltage users with contracted power higher than 10 kW and every high voltage customer had to contract their energy tariffs in the liberalized energy market. The change in legislation has mainly influenced the behavior of the clients from the disappeared nocturnal tariff, which has been switched to mostly type b users. These are the clusters of clients with high energy consumption at night hours. It can be observed from the results that in the patterns from some clusters the loads have shifted from night hours to morning or midday hours.

It can also be observed that groups of clients with higher energy consumption than the average are grouped at the same clusters. These users should be the main objective for DSM actions, since it is more likely that manageable equipment and appliances can be found at these residences.

The dynamic clustering allows capturing the trend of groups of users at a glance. As can be seen in the results, the clients are clustered by level of energy consumption and by the form of their load profiles, thus allowing a classification of these users according to the way they consume energy (when and how much).

the K-means or FCM - based algorithms, with Euclidean or Hausdorff - based distances, provide the best defined and well-balanced clusters and patterns. Worse results have been obtained with the FFCM algorithm and correlation - based clustering.

Regarding the granularity of the data, a possible loss of information has been noticed when partitioning the sequences and analyzing them separately (working and non-working days) or when aggregating data (cumulated values). Therefore, the analysis of the whole sequence of raw data is preferred.

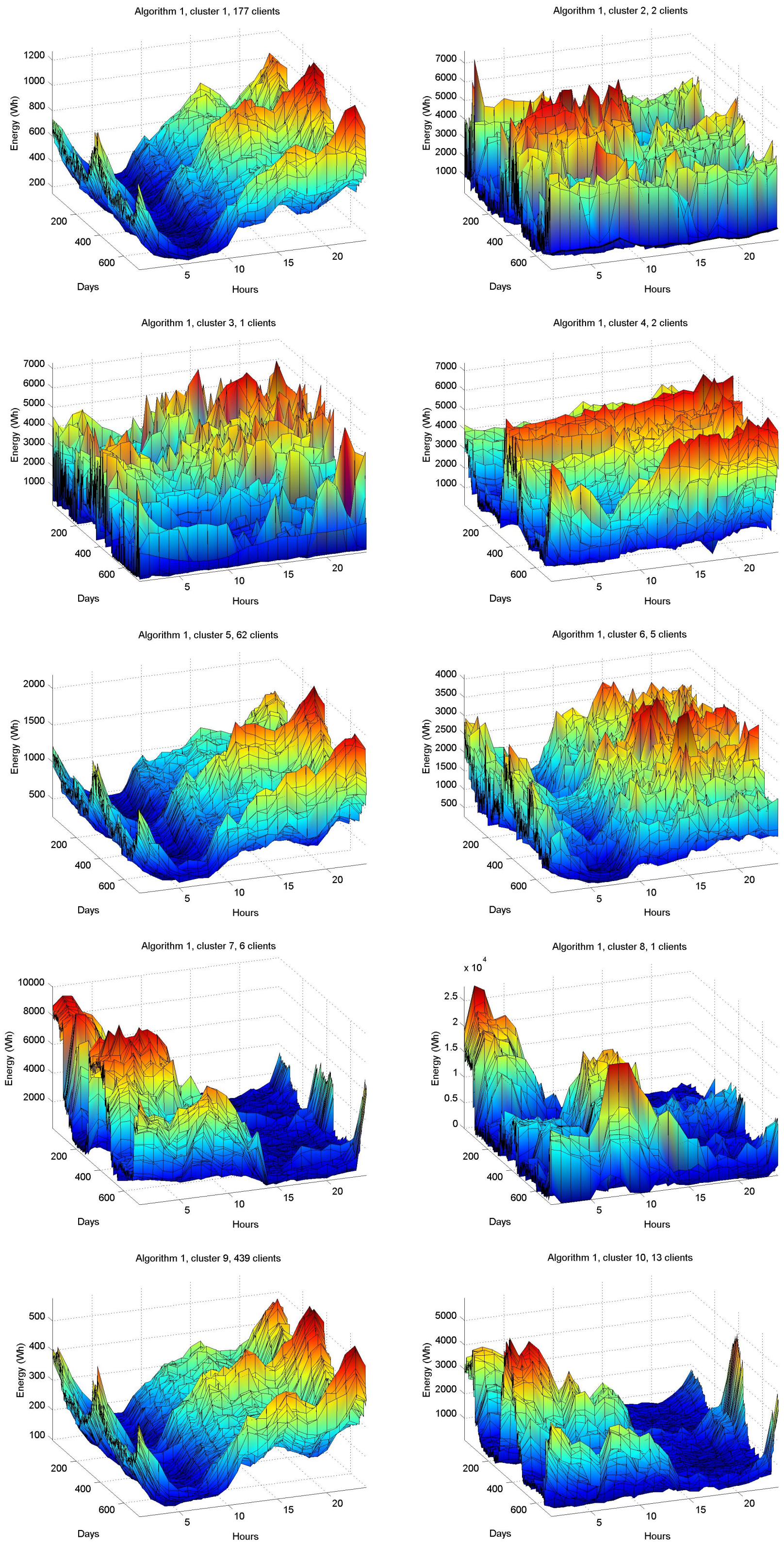
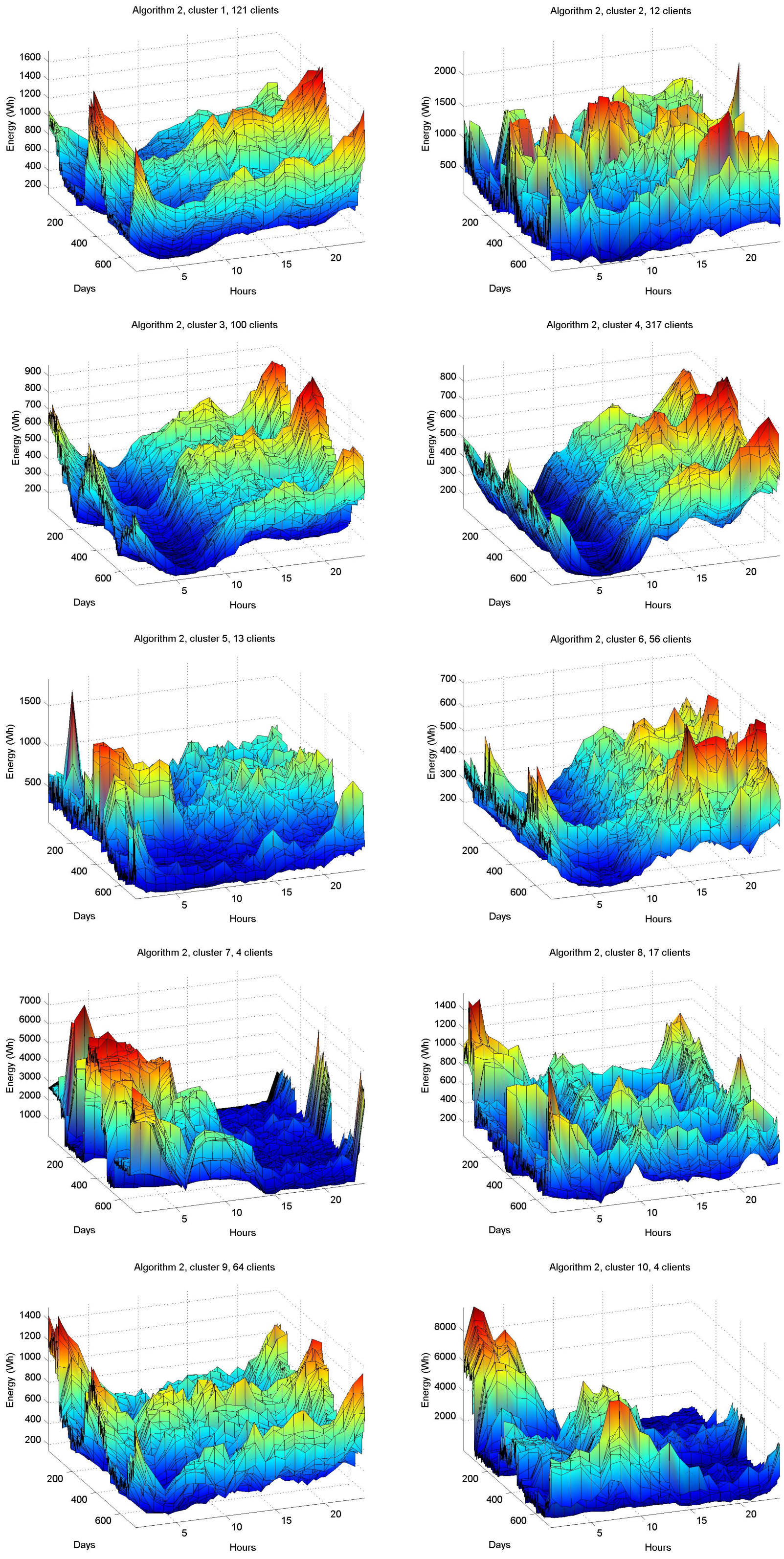


Fig. 12.19. Results from END-KME dynamic clustering algorithm.



Algorithm 2, cluster 1, 121 clients

Algorithm 2, cluster 2, 12 clients

Algorithm 2, cluster 3, 100 clients

Algorithm 2, cluster 4, 317 clients

Algorithm 2, cluster 5, 13 clients

Algorithm 2, cluster 6, 56 clients

Algorithm 2, cluster 7, 4 clients

Algorithm 2, cluster 8, 17 clients

Algorithm 2, cluster 9, 64 clients

Algorithm 2, cluster 10, 4 clients

Fig. 12.20. Results from END-KMC dynamic clustering algorithm.

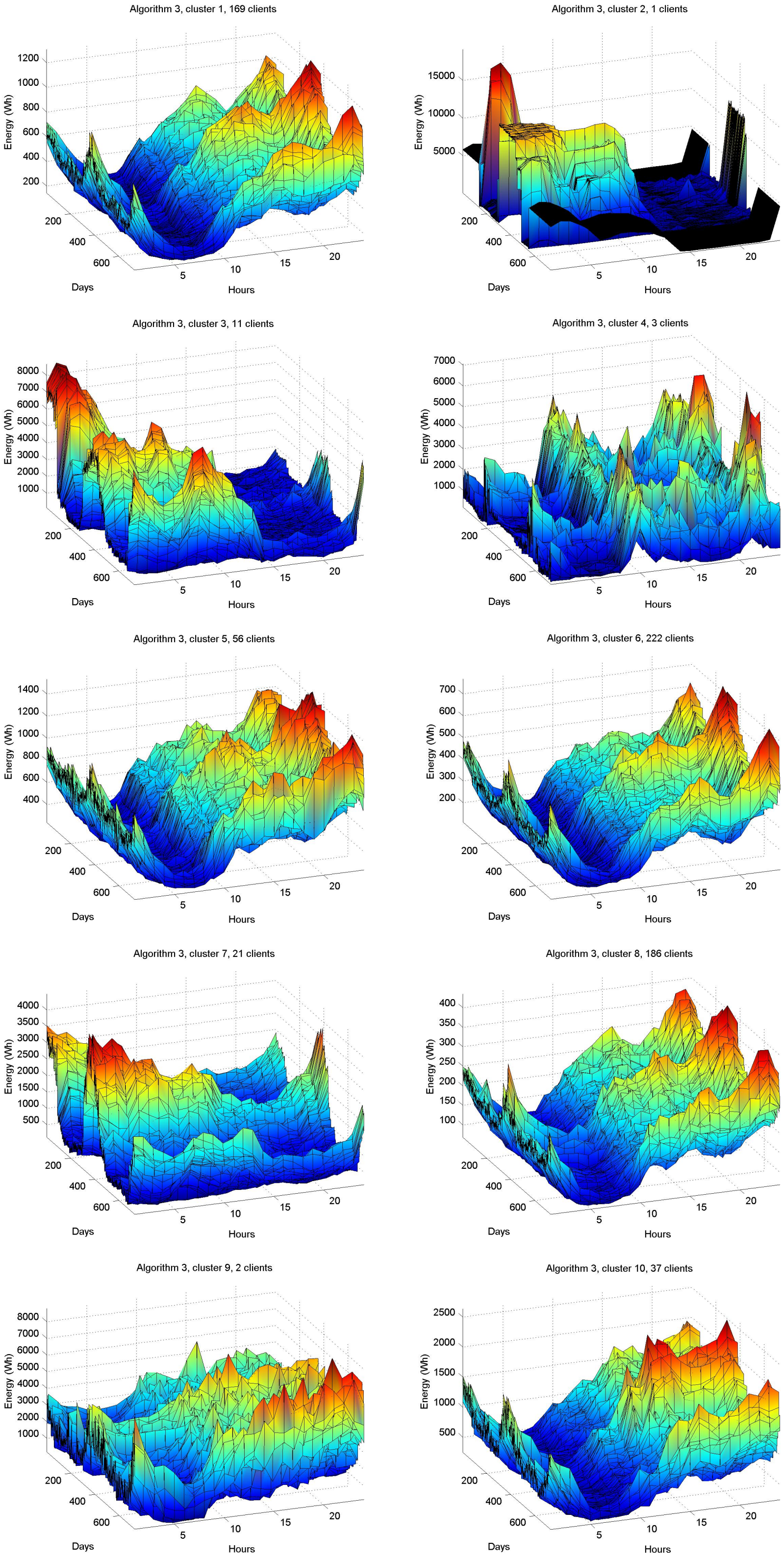
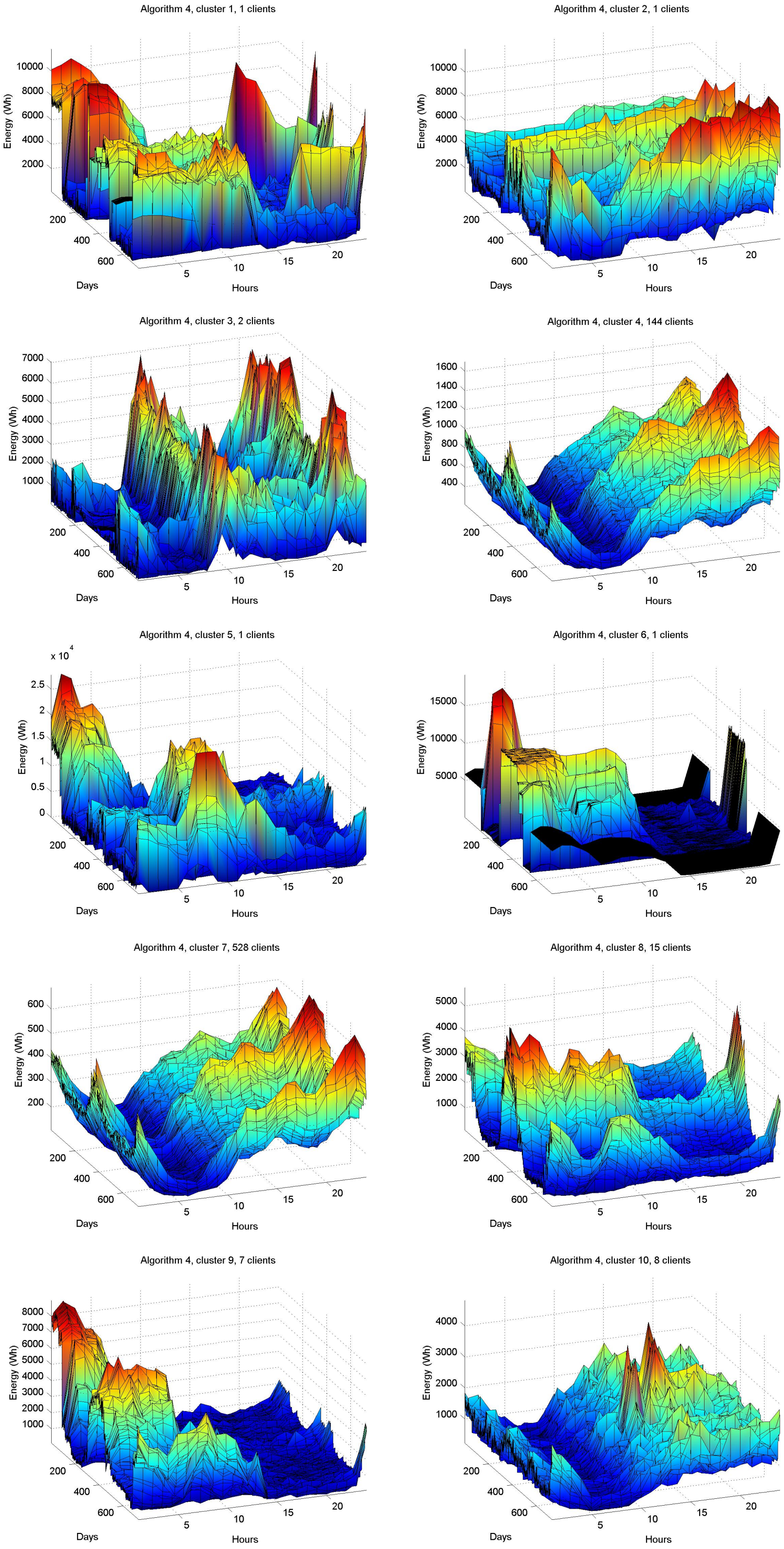


Fig. 12.21. Results from END-KMH dynamic clustering algorithm.



Algorithm 4, cluster 1, 1 clients

Algorithm 4, cluster 2, 1 clients

Algorithm 4, cluster 3, 2 clients

Algorithm 4, cluster 4, 144 clients

Algorithm 4, cluster 5, 1 clients

Algorithm 4, cluster 6, 1 clients

Algorithm 4, cluster 7, 528 clients

Algorithm 4, cluster 8, 15 clients

Algorithm 4, cluster 9, 7 clients

Algorithm 4, cluster 10, 8 clients

Fig. 12.22. Results from Extended K-means clustering algorithm.

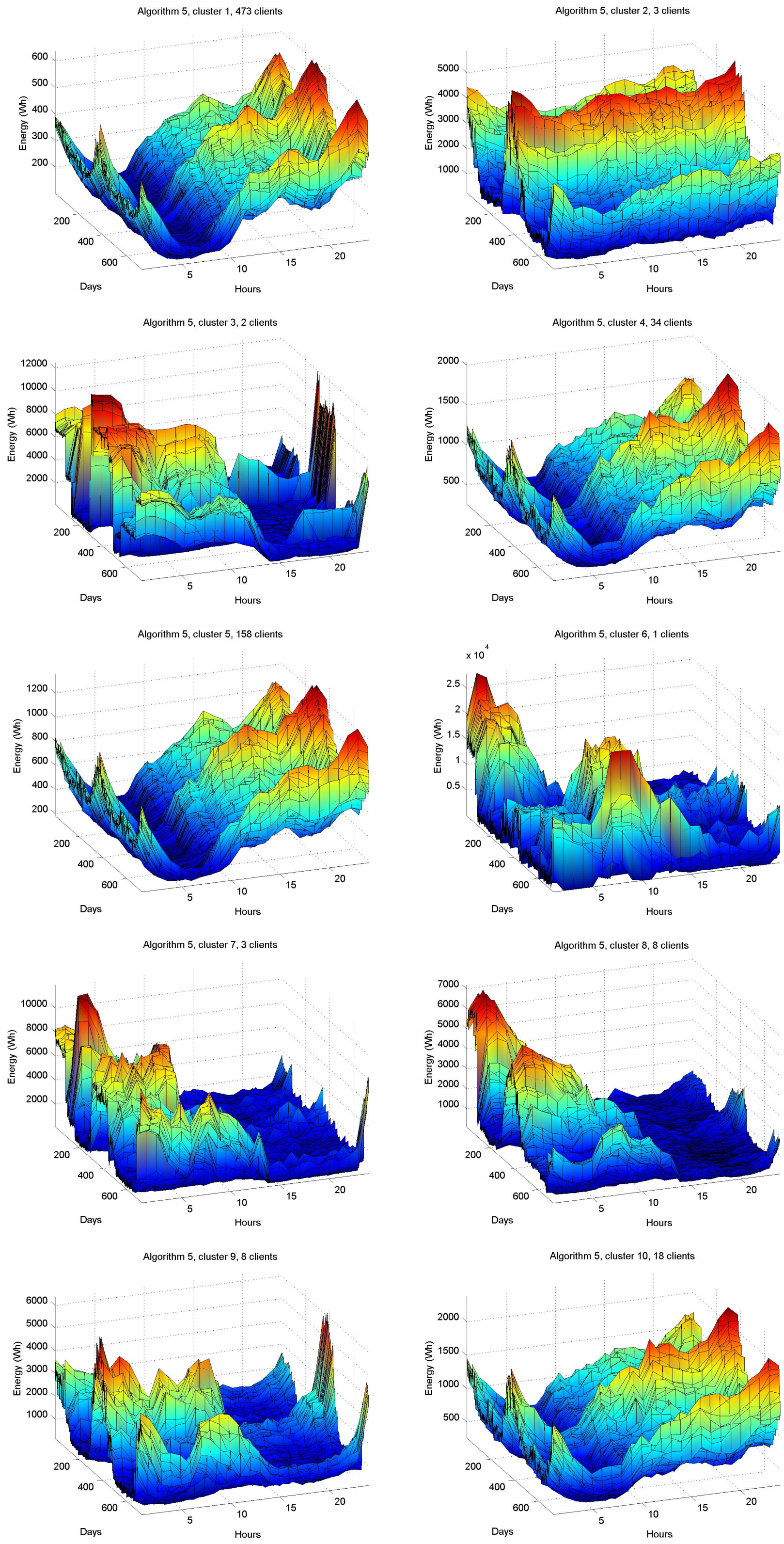
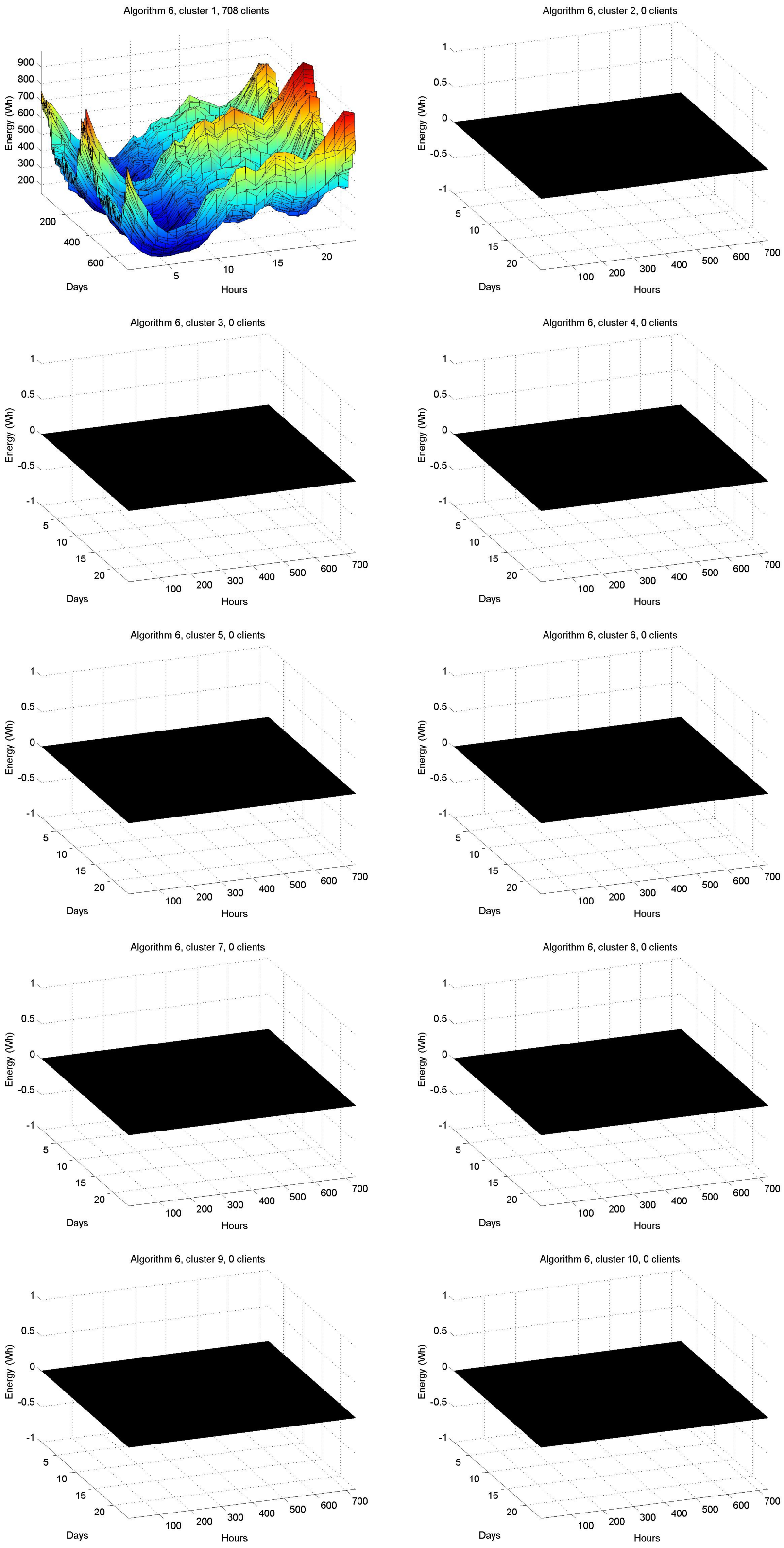


Fig. 12.23. Results from END-FCME dynamic clustering algorithm.


Algorithm 6, cluster 1, 708 clients

Algorithm 6, cluster 2, 0 clients

Algorithm 6, cluster 3, 0 clients

Algorithm 6, cluster 4, 0 clients

Algorithm 6, cluster 5, 0 clients

Algorithm 6, cluster 6, 0 clients

Algorithm 6, cluster 7, 0 clients

Algorithm 6, cluster 8, 0 clients

Algorithm 6, cluster 9, 0 clients

Algorithm 6, cluster 10, 0 clients

Fig. 12.24. Results from END-FCMC dynamic clustering algorithm.

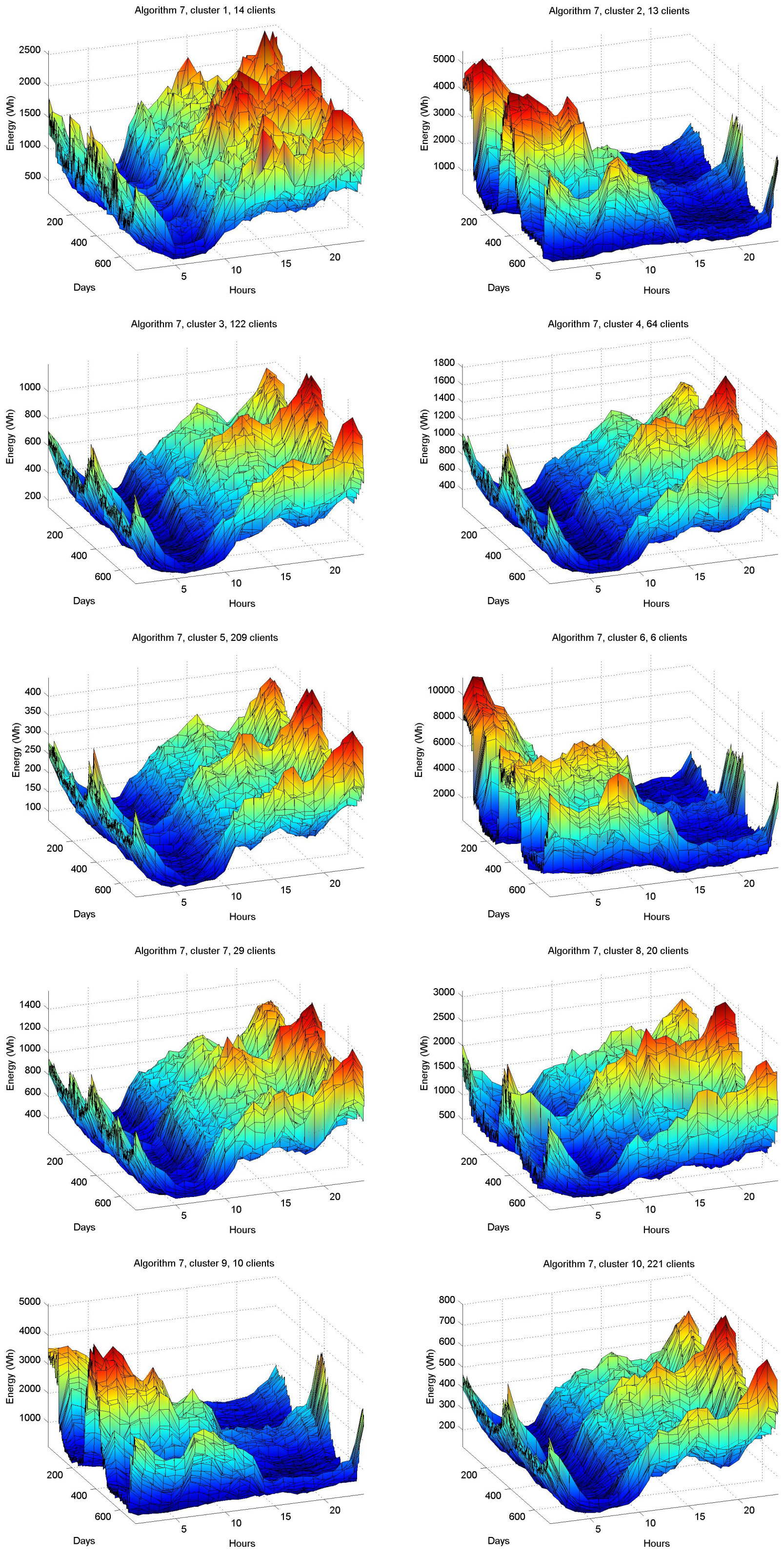


Fig. 12.25. Results from END-FCMH dynamic clustering algorithm.

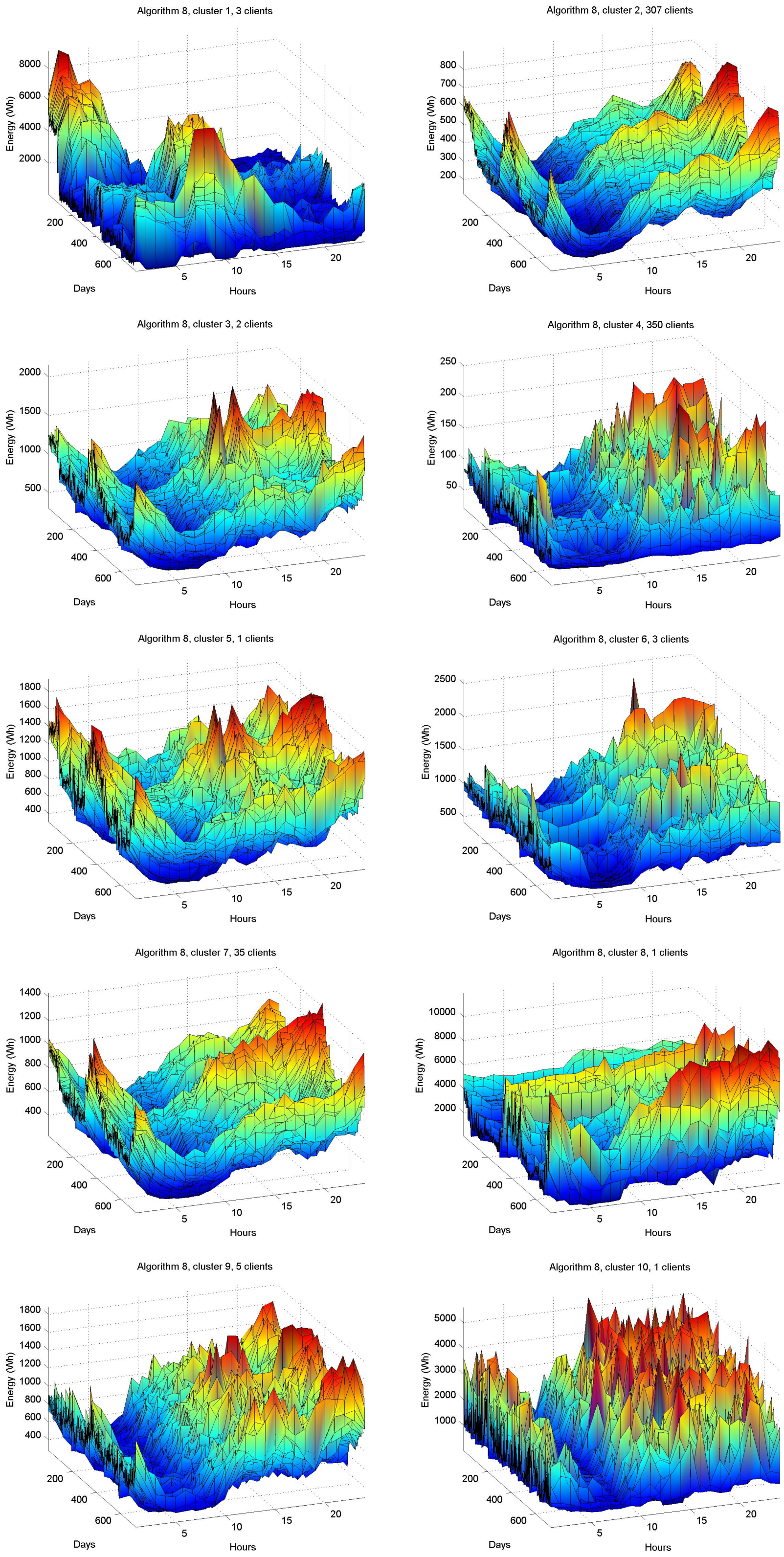


Fig. 12.26. Results from FFCM dynamic clustering algorithm.

13

Conclusions from developments and tests

13.1 Data mining field of knowledge

The present thesis describes a common framework for the dynamic clustering (Type 3) of objects with n dimensions of the same magnitude. It has been designed in a way that most of the partitional Type 1 (i.e. static) clustering algorithms can be used with it, once adapted and integrated in the framework.

No previous work in this sense has been found in the State of the Art. Some algorithms and concepts have been found that could be the first approach to the developments presented in this document; however, these works lacked a generic view and a broader scope, beyond the specific data types they were designed for.

Three tests have been performed and up to 8 different algorithms have been compared. As has been stated in the previous Section (Section 12.5.3), "the quantitative analysis, by means of specifically modified clustering validity indices, and a qualitative study, by observing the resulting patterns ad assigning them to one of five groups defined for typical profiles of residential energy consumption in Spain, are coincident in their results: the K-means or FCM - based algorithms, with Euclidean or Hausdorff - based distances, provide the best defined and well-balanced clusters and patterns." These algorithms are therefore indicated as the best option for the dynamic clustering of load profiles time series. However, depending on the nature of the data, other techniques and similarity measures may work better, as indicated in the results from the second test, with the analysis of the synthetic data set from the UCI repository (Section 12.5.2).

Regarding the analysis of load profiles time series, a more detailed analysis of the results indicates that the END-KMH algorithm provides the most balanced clustering results, in number of clusters and clients within each cluster per each of the five identified groups of residential customers of electrical energy. This can be seen by rearranging Tables 12.14 to 12.21 in two Tables where the number of clusters per group and the total number of clients per group are shown (Tables 13.1 and 13.2).

Table 13.1. Assignment of clusters to expected groups, results from Test 3.

Group	END-KME	END-KMC	END-KMH	Extended K-means	END-FCME	END-FCMC	END-FCMH	FFCM
1	1	3	2	1	1	1	2	2
2	2	1	2	1	3	0	3	4
3	1	0	1	1	0	0	2	2
4	3	4	2	3	1	0	0	1
5	3	2	3	4	5	0	3	1

Table 13.2. Assignment of clients to expected groups, results from Test 3.

Group	END-KME	END-KMC	END-KMH	Extended K-means	END-FCME	END-FCMC	END-FCMH	FFCM
1	439	473	408	528	473	708	430	657
2	239	121	225	144	210	0	215	43
3	5	0	37	8	0	0	34	4
4	5	96	5	4	3	0	0	1
5	20	8	33	24	22	0	29	3

As can be seen, the END-KMH algorithm provides a well-balanced assignment in clusters (2-2-1-2-3), better than the other 7 algorithms. The assignment of clients in each group is also well-balanced, taking into into account the characteristics of the five groups of customers.

It can also be seen from the Tables that the K-means-based algorithms (END-KME and END-KMH) are the ones that provide the best well differentiated patterns in the five groups, from the 8 algorithms compared. The FCM-based algorithms are not able to identify customers from one group, being group 3 in the case of END-FCME and group 4 in the case of the END-FCMH algorithm.

Regarding the similarity measures used, it can be observed that the Hausdorff-based distance works well to identify group three clients (the ones with a typical shape but an elevated average level of daily energy consumption). This can be observed in the results from END-KMH (37 clients) and END-FCMH (34 clients). The Hausdorff similarity, therefore, is able to differentiate objects with similar shape but different levels or offsets of energy consumption. The END-FCMH algorithm, however, has not been able to identify clients from the group 4 (flat or non-typical shape, with an elevated daily average of energy consumption). In this case the END-FCME algorithm has presented a better performance.

13.2 Power systems field of knowledge

The present thesis develops a technique to summary a number of customers and their smart metering data in a reduced number of temporal patterns, able to display trends in shape and level of energy consumption during the day, and their evolution through time. Very interesting information and conclusions can be obtained from this analysis by an expert in distribution networks and electrical customers.

As has been indicated in previous Chapters, the objectives of extracting knowledge from load profiles data and, more specifically, the segmentation, classification and forecasting, are identified needs by stakeholders and experts in data management systems from the electrical sector. As has been also indicated, no previous work similar to the developments presented in this thesis has been found in the literature, regarding the cluster analysis of load profiles time series. The results from the developments made have been compared and evaluated, and contrasted with the expected five groups of residential energy customers in Spain according to the way the electrical energy is consumed. These five groups have been identified based on the analysis of the data, and represent the different profiles of residential energy consumption in Spain. The results provide well defined temporal patterns and clusters, that fit to the five different groups of customers expected.

The results obtained provide a feasible and valuable analysis for the different experts and agents involved in the management of power systems, and can serve for different purposes, such as predictive maintenance, evaluation of consumption trends, detection of non-typical patterns of consumption, or identification of groups of customers with specific characteristics for the provision of energy.

Section 3.4 of the present thesis cites the report on the research and development needs [69] foreseen by the European Technology Platform on Smart Grids (ETP SG) for the EC Horizon 2020 Research and Innovation Programme [70], for the years 2016 and 2017. This report identifies as two of the main challenges in the near future the "**utilization** of smart metering data", and the development of "models of **segmentation** of the customers based on their flexibility or availability for demand side management programs". The developments presented in this thesis fulfill both objectives, therefore providing a step forward in the implementation of this kind of analyses in Big Data systems in the environment of the future Smart Grids, able to capture all the available knowledge from the data and to benefit from it.

Part IV

Conclusions

14

Main conclusions

The present thesis has been designed, from its first Chapters until the beginning of Part III, as a comprehensive guide of the different steps to help in the understanding of the current scenario of the power systems and the possibilities of applying the data mining techniques, specifically for one objective of the data mining considered of relevance and, as has been seen, yet not developed, to extract knowledge from the information of energy demand load profiles, that may be of interest for the different agents involved in the management of the power grids.

First, a brief introduction to the current situation of power grids and their transitional state towards the grids of the future or Smart Grids has been presented. From these Chapters two main conclusions are identified (see Chapter 4):

- Large amounts of data are being increasingly available, as more equipment and AMI are being installed.
- There is a growing need and interest to define specific analyses for these data, to make them really useful and profitable.

These Chapters have emphasized on the picture of the situation in the present period regarding the transition towards the smart grids and liberalized markets, in Europe and Spain (Chapter 1); the current configuration of power grids and power systems (Chapter 2); and the impact of the implementation of AMI and the implications of the added complexity of the systems to adequately handle huge amounts of data (Big Data and Big Data Analytics) (Chapter 3).

Then, the data mining objectives and techniques have been described, first as a group of techniques with different objectives, which are part of a bigger process of extracting knowledge from databases (KDD) (Chapter 5), and then focusing on the cluster analysis, either for static and for dynamic or evolution analysis of time series data (Chapters 6 and 7). Finally, the application of data mining objectives and techniques to the analysis of load profiles has been reviewed, mainly in classification and prediction techniques (Chapter 10).

The last Chapters of this thesis present the developments made to achieve its main objective (described mainly in Chapter 4): to study, develop, test and evaluate methods and techniques to perform clustering, obtain centroids or prototypes and visualize and analyze the results, on time series of daily load profiles. Having this description as the main objective, in the present thesis, the following achievements have been made through the different developments presented (Chapter 11):

- The development of a **common framework** to a dynamic cluster analysis of time series data of load profiles, applying different clustering techniques and distance measures, has been presented. This work has also been described in the following communication: "Ignacio Benítez, Alfredo Quijano, José-Luis Díez, Ignacio Delgado. *Clustering of Time Series Load Profiles for Grid Reliability.* International Conference on Condition Monitoring, Diagnosis and Maintenance 2015 (CMDM 2015). Bucharest, Romania, October 5th -8th, 2015".
- A graphical method for the visualization and **representation of the clusters**, and a methodology for the **interpretation of the results**, by means of specific indices, have been described. This methodology is presented in the following article: "Benítez, I.; Quijano, A.; Díez, J.-L. and Delgado, I. *Dynamic clustering segmentation applied to load profiles of energy consumption from Spanish customers.* International Journal of Electrical Power & Energy Systems , 2014, 55, 437 - 448", which also presents the first approach for dynamic clustering developed, along with the results from the first test described in this thesis.
- A selection and modification of **cluster validity indices** appropriate to evaluate the clustering results in this proposed framework has been performed.
- An approach to compare two load profiles time series as the **comparison between two 3D surfaces** has been presented.
- A proposal to obtain the distance between the two surfaces as the decomposition in a number of smaller linear surfaces and the application of a new **Hausdorff-based similarity measure** has been explained.

The three last contributions are described in a new article with the following title: "Dynamic Clustering of Residential Electricity Consumption Time Series Data Based on Hausdorff Distance", by Ignacio Benítez, José-Luis Díez, Alfredo Quijano and Ignacio Delgado. This paper has been sent to the Electric Power Systems Research Journal for publication and is currently under review.

Specific tests have been designed and performed to validate these developments, and the results have been evaluated, either in form of quantitative indices and also based on heuristics or the previous knowledge on the data sets and the expected clusters. These results have been described and discussed in the last Chapter (Chapter 12).

15

Future Works

The developments presented in this thesis can be seen as a first step in the field of dynamic clustering of dynamic load profiles. Given this first step, there are countless further steps that can benefit from the developments presented in this thesis. Following, some of these possible future works are proposed.

15.1 Type 4 dynamic clustering and batch processing of the time series data

The developments presented can be seen as a one-batch clustering of the full length of the data objects, i.e., for the total number of samples. However, new developments could be made to allow the extension of the Type 3 clustering presented in this thesis (static classes or clusters and dynamic data) to be of the Type 4 clustering. The Type 4, as indicated in Chapter 7, has the following description:

> The data is also treated as dynamic, as in type 3, becoming feature trajectories that evolve through time, and the number of clusters varies dynamically at each iteration. Clusters and patterns can, therefore, as in type 2, merge or split.

The dynamic clustering framework developed could be extended, therefore, to a sequential batch processing, segmenting the data objects in a number of samples, obtaining the clusters for each batch, and then applying a pattern matching procedure [230]. The development presented is prepared for this purpose, by the definition of two parameters, *horizon* and *lag*:

- An *horizon* parameter, is the number of days or **samples** that are to be clustered, and form the current batch (15.1).
- A *lag* parameter, indicates the **lag** or delay factor or number of samples from the previous batch processing iteration being analyzed in the new batch (15.2).

$$1 \leq horizon \leq \text{number of samples} \tag{15.1}$$

$$1 \leq lag \leq \text{number of samples} \tag{15.2}$$

These two parameters allow a flexible clustering process of temporal sequences in sequential batches. The size of the resulting centroids will be obtained as a function of the number of samples, the lag and the horizon, as can be seen in (15.3).

$$\text{centroid size} = min(horizon, lag) * \left| \frac{\text{number of samples}}{lag} \right| \tag{15.3}$$

This development, however, has not been done. Currently the size of the data objects is determined by the maximum number of samples available for each data object or client.

Regarding cluster merging techniques, inclusion indices [90], for instance, can be applied, which give a measure of clusters' overlapping. Other technique to be applied is the participatory learning [90], where objects are assigned to the cluster with the highest compatibility. If a threshold is not reached, then a new cluster is created.

15.2 Possibilistic clustering

The possibilistic clustering (see Chapter 6) is a kind of non-exclusive membership clustering, such as the fuzzy clustering, but in a possibilistic partition, the condition is that each object must have at least one of its degrees of membership to clusters greater than 0.

The developments presented in this thesis make use of common static clustering techniques of the exclusive and fuzzy kinds, such as K-means and FCM. The possibilistic clustering, such as the PCM [181], has not been implemented.

15.3 Feature weighting or discrimination

Another possibility to be implemented and tested is the discrimination of features or dimensions, implementing clustering algorithms with feature weighting or feature discrimination [90], in order to discover the dimensions which yield best results. In the case of daily load profiles, the hours of the day that provide the highest variability would be identified.

15.4 Decomposition of the load profiles data objects in a variable number of smaller surfaces

In Section 11.6, a specific development is presented, being a similarity measure to compare two energy consumption patterns, based on a two-step sequence: the decomposition of the patterns in linear surfaces, and the computation of Hausdorff distances [220] between the surfaces. The smaller surfaces are obtained partitioning the day in specific regions based on experience of how the load profiles evolve through the day. However, the definition of the smaller surfaces boundaries could be given by an algorithm, based on the slopes and variations in trends of the object data. In this sense, the work presented by Díez [166] of clustering algorithms for the identification of local models, such as the AFCRC, could be implemented to automatically detect and partition the surface in smaller surfaces with a similar behavior.

The dynamic clustering of objects seen as dynamic surfaces is also another trend for future research. In this thesis a comparison of surfaces has been presented based on the Hausdorff distance. Other methods to compare surfaces can be developed and tested for dynamic clustering. Also, as indicated in Section 11.5, the similarity measure between two surfaces could be expressed by not only one value, but a set of characteristic variables.

15.5 Definition of indices for the trend of clusters

One of the main advantages of the developments presented in this thesis is capability to observe the trend or evolution in consumption in form of temporal patterns. This knowledge can be quantitatively evaluated by the definition and use of specific temporal indices for the patterns obtained. These indices can inform on the evolution of the pattern, in terms of slope (incremental, decremental or flat consumption, for instance) or variability of the pattern through the days.

15.6 Dynamic clustering of quarter-hour energy demand measures

All the developed works have been applied on a database with daily measures of 24 hours of energy demand. However, the smart meters being implemented in households are able to measure energy consumption per quarter-hour, having there daily load profiles with 96 measures or dimensions. These data have not been tested in the present thesis.

15.7 Application of dynamic clustering to the development of prediction models

The dynamic clustering techniques developed can be used as the initial step to reach a different data mining objective, such as load forecasting. As has been seen in the review made on load forecasting techniques (Chapter 10), clustering is used as an initial step to obtain patterns that are the input for prediction models. In a similar approach, the dynamic clustering could obtain the patterns for local models for identification purposes, such as the developments described by Díez in his thesis [166], where static clustering algorithms are developed and applied with this purpose.

Part V

Appendices

A

Work, energy and power

A.1 Work

The *work* performed by a force between two points is defined as the curvilinear integral of the force along the trajectory followed between the two points, A and B [353].

$$W_{AB} = \oint_A^B \mathbf{F} d\mathbf{r} \tag{A.1}$$

If the force $\mathbf{F}$ is kept constant along the trajectory, (A.1) can be written as (A.2).

$$W_{AB} = \oint_A^B \mathbf{F} d\mathbf{r} = \mathbf{F} \oint_A^B d\mathbf{r} = \mathbf{F}(\mathbf{r_B} - \mathbf{r_A}) = \mathbf{F}\bar{AB} \tag{A.2}$$

If the constant force keeps a constant angle with the trajectory, (A.2) becomes (A.3).

$$W_{AB} = \mathbf{F}\bar{AB} = |\mathbf{F}||\bar{AB}|\cos\theta = Fd\cos\theta \tag{A.3}$$

In the case that the force is applied in the same direction as the movement, (A.3) can be written as the product of a constant force F along distance d (A.4).

$$W_{AB} = Fd\cos\theta = Fd\cos 0 = Fd \tag{A.4}$$

The work is measured in Joules (J). A Joule is defined as the work performed by a force of 1 Newton (N) applied in the same direction of the movement along 1 meter (m).

A.2 Power

Power is defined as the work performed by time unit (A.5).

$$P = \frac{dW}{dt} \tag{A.5}$$

The power is an instantaneous value, whereas the work is the power along a specific interval (A.6).

$$dW = Pdt \rightarrow W = \int Pdt \tag{A.6}$$

The power is measured in Watts (W), being 1 Watt equal to $1J/1s$. The electrical power associated with the charge flow through an element [354] can be seen in (A.7), which defines the relation among power, voltage and current.

$$P = \frac{dW}{dt} = \frac{dW}{dq}\frac{dq}{dt} = vi \tag{A.7}$$

A.3 Energy

the *Energy* is a measure of the *capacity* of work that can be delivered. Although the unit of energy is also the Joule, usually the Watts per hour (Wh) unit of measure is used instead, which is also a measure of work (power x time), as can be seen in (A.8).

$$W = \int Pdt = 1[Watt]x1[h] = 1[Watt]x3600[s] = \frac{1[J]}{1[s]}x3600[s] = 3600[J] \tag{A.8}$$

B

Minimization of objective functions of clustering algorithms

When defining clustering algorithms, a critical issue is the definition of the objective function whose minimization yields the iterative formulas that define each different partitional algorithm found in the literature.

B.1 Minimization of the K-means objective function

The objective function is described in (B.1), where N is the number of observations, K is the number of clusters, x_n are the objects or observations, c_k are the prototypes or centroids, and r_{nk} is a binary indicator, used to indicate to which cluster the data object belongs to. This indicator has a value of 1 if the object n belongs to the cluster k, and a value of 0 otherwise.

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|x_n - c_k\|^2 \tag{B.1}$$

The procedure to minimize this objective function is given as described by Bishop [79] and follows the approach of the Expectation-Maximization (EM) algorithm [355]. The optimization is performed in two iterative steps. First, J is minimized with respect to r_{nk}, keeping c_k fixed. Second, J is minimized with respect to c_k, keeping r_{nk} fixed. These two steps correspond respectively to the Expectation (E) and Maximization (M) steps of the EM algorithm.

The minimization in the E step is performed by simply assigning the value of 1 to the r_{nk} whose object is closest to the prototype k, i.e., to the r_{nk} with the minimum value of distance $\|x_n - c_k\|^2$. The value will be zero otherwise. With this assignment, each object of the observations is assigned to only one prototype.

The minimization in the M step is performed by deriving the objective function J with respect to c_k and then making the expression equal to zero. This minimization is expressed in (B.2) and (B.3).

$$\frac{\partial J}{\partial c_k} = 2\sum_{n=1}^{N} r_{nk}(x_n - c_k) = 0 \tag{B.2}$$

$$\sum_n r_{nk}x_n - \sum_n r_{nk}c_k = 0 \tag{B.3}$$

From this minimization, the expression for the computation of c_k is obtained, as indicated in (B.4). From this expression it can be seen that the prototypes or centroids are calculated as the mean value of all the objects from each cluster.

$$c_k = \frac{\sum_n r_{nk}x_n}{\sum_n r_{nk}} \tag{B.4}$$

B.2 Minimization of the FCM objective function

The following procedure to minimize the objective function for the FCM was described by Bezdek in 1980 [356]. The FCM objective function is the summation of all the distances of objects to the resulting prototypes (B.5), weighted by a membership coefficient μ_{ik}. As indicated in Expression (B.6), there is a constraint to the value of the coefficients, which is that the sum of all the membership values of a given object to all the clusters must equal 1.

$$J(Z;U,C) = \sum_{i=1}^{c}\sum_{k=1}^{N}(\mu_{ik})^m \|z_k - c_i\|_B^2 \tag{B.5}$$

$$\sum_{i=1}^{c}\mu_{ik} = 1 \tag{B.6}$$

The first step is to rearrange the objective function, including the defined constraint by means of Lagrange multipliers. The new objective function, expressed in (B.7), now has as variables the vectors W (membership coefficients) and α (Lagrange multipliers). In order to minimize this objective function, it is assumed that all the prototypes are fixed, and thus all the distances $\|z_k - c_i\|_B^2$ are substituted by the term d_{ik}. The membership coefficient μ_{ik} is substituted by the variable w_{ik}^2.

$$\Phi(W,\alpha) = \sum_{k=1}^{n}\sum_{i=1}^{c}(w_{ik})^{2m}d_{ik} + \sum_{k=1}^{n}\alpha_k(\sum_{i=1}^{c}w_{ik}^2 - 1) \tag{B.7}$$

The minimization of the objective function is performed by obtaining the partial derivatives, with respect to W and α, and making them equal to zero. The derivatives, expressed in (B.8), are obtained with respect to one of the vectors' variables, just one value of i and k, being l and p respectively, assuming that $1 \leq l \leq c$ and $1 \leq p \leq n$.

$$\begin{cases} \frac{\partial \Phi}{\partial w_{lp}} = 2m(w_{lp})^{2m-1} d_{lp} + 2w_{lp}\alpha_p = 0 \\ \frac{\partial \Phi}{\partial \alpha_j} = \sum_{i=1}^{c} w_{ij}^2 - 1 = 0 \end{cases} \tag{B.8}$$

From the $\frac{\partial \Phi}{\partial w_{lp}}$ partial derivative, and substituting w_{lp}^2 by μ_{lp}, the Expression in (B.9) for the membership coefficient is obtained, as a function of the Lagrange multiplier α_p.

$$\mu_{lp} = \left(\frac{-\alpha_p}{m d_{lp}} \right)^{\frac{1}{m-1}} \tag{B.9}$$

By summing all the membership coefficients over l for the c clusters, and applying the result of the $\frac{\partial \Phi}{\partial \alpha_j}$ partial derivative ($\sum_{i=1}^{c} w_{ij}^2 = 1$), the Expression in (B.10) for each α_p is obtained.

$$-\alpha_p^{\frac{1}{m-1}} = \frac{1}{\sum_{i=1}^{c} \left(\frac{1}{m d_{lp}} \right)^{\frac{1}{m-1}}} \tag{B.10}$$

The Expression for α_p (B.10) is then substituted in the Expression for μ_{lp} (B.9). The result, which can be seen in (B.11), is the Expression used to compute each μ_{lp} at each iteration of the FCM algorithm.

$$\mu_{lp} = \frac{1}{\left(\frac{d_l p}{\sum_{i=1}^{c} d_{ip}} \right)^{\frac{1}{m-1}}} \tag{B.11}$$

The expression to obtain the c prototypes at each iteration can be derived in a similar way as Expression (B.4) for the K-means, and it is equivalent to the weighted mean of all the objects that belong to each cluster, weighted by their respective membership coefficient μ_{ik}.

C

Linear Least Squares

The explanation of least squares for regression can be found in the literature, in most of the works that describe regression techniques, for instance in [79]. It must be noted that the regression model can be either simple or multiple, or linear or non-linear.

Multiple linear regression is addressed in this case. Let us consider the equation of a surface, as given in (C.1).

$$z = w_0 + w_1 x + w_2 y \tag{C.1}$$

Given a number of observations $(z_1, z_2, ..., z_n)$, each value can be written as a function defined by a number of independent variables (x), their coefficients (w), and a residual or error value (ε_i), which is an error measure between the observed value and the expected one. This formulation can be seen in (C.2) and, suited to the equation of a surface, in (C.3).

$$z_i = f(x_i, w) + \varepsilon_i \tag{C.2}$$

$$z_i = w_0 + w_1 x_{i1} + w_2 x_{i2} + \varepsilon_i \tag{C.3}$$

Expression (C.3) can be arranged as a matrix for the number n of observations, yielding expressions (C.4) and, in simplified form, in (C.5).

$$\begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{pmatrix} = \begin{pmatrix} 1 & x_{11} & x_{12} \\ 1 & x_{21} & x_{22} \\ & \vdots & \\ 1 & x_{n1} & x_{n2} \end{pmatrix} \begin{pmatrix} w_0 \\ w_1 \\ w_2 \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{pmatrix} \tag{C.4}$$

$$Z = XW^t + \varepsilon \tag{C.5}$$

In order to obtain the coefficients that fit the observations to the desired function, the Least Squares method minimizes the Error function, which is defined as the sum of squared residuals, or the difference between the observations and the function outcomes, as can be seen in (C.6).

$$Error = \frac{1}{2}\sum_{i=1}^{n}(z_i - f(x_i, w))^2 \tag{C.6}$$

From the result of this minimization, the formula to compute the coefficients, as a batch process, is obtained. This formula can be seen in (C.7).

$$W = (X^t X)^{-1} X^t Z \tag{C.7}$$

The following example illustrates the result obtained when applying Least Squares on a dynamic load curve, in order to compute the equivalent surface. Figure C.1 depicts the original load curve and the surface obtained from multiple linear regression applying Least Squares. Figure C.2 plots the histogram of the residuals or error values, between the observations (the load curve) and the equivalent points on the resulting surface.

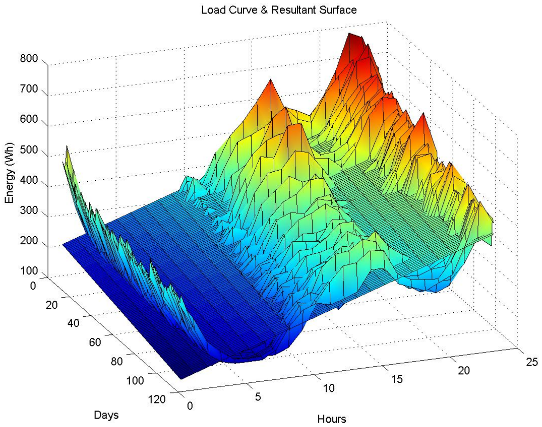


Fig. C.1. Example of Least Squares applied on a dynamic Load Curve.

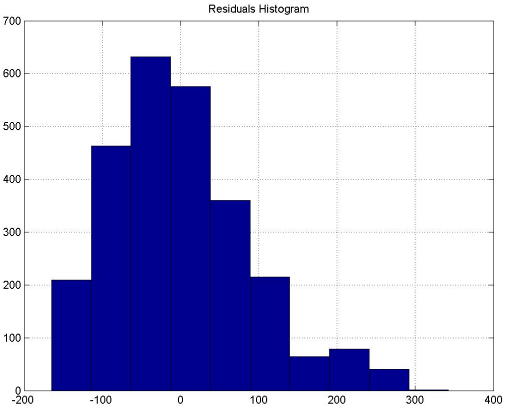


Fig. C.2. Example of Least Squares applied on a dynamic Load Curve. Histogram of residuals.

D

Cluster Validity Indices for Static Clustering

The following appendix is a selection of some of the most common cluster validity indices found in the literature to be applied on static clustering techniques, particularly on partitional, fuzzy and non-fuzzy algorithms. There are, of course, other indices based on different techniques, such as statistical methods, or designed for other clustering methods, like hierarchical clustering.

All the following indices have been extracted from previous state of the art works and studies from Díez [166], Halkidi et al. [210] and Wang and Zhang [357].

D.1 Partitional non-fuzzy clustering

D.1.1 Davies-Bouldin index

This index was described by Davies and Bouldin in 1979 [207]. It defines the similarity measure R_{ij} between two clusters i and j as in (D.1), where s_i and s_j are measures of dispersion of the clusters (see (D.2)) and d_{ij} is the distance between the two clusters, which is typically obtained as the Euclidean distance between the two cluster prototypes or centroids. C_i is the size of the cluster i, obtained from the number of elements that belong to the cluster.

$$R_{ij} = \frac{s_i + s_j}{d_{ij}} \tag{D.1}$$

$$s_i = \frac{1}{\mid C_i \mid} \sum_{x \epsilon C_i} \parallel x - c_i \parallel \tag{D.2}$$

The expression R_{ij} indicates, therefore, a relation between the compactness of the clusters and the distances among them. The Davies-Bouldin or DB index is defined in (D.3), where N_c is the number of clusters and R_i is defined in (D.4).

$$DB = \frac{1}{N_c} \sum_{i=1}^{N_c} R_i \tag{D.3}$$

$$R_i = max_{j=1...N_c, i \neq j} R_{ij}, i = 1...N_c \tag{D.4}$$

Since the objective is to obtain clusters with the highest separation among them and the lowest dispersion, it is expected that the lower the value of the DB index, the better the selection of the number of clusters will be.

D.1.2 Dunn and Dunn-like indices

This index was proposed by Dunn in 1974 [358]. It defines the similarity between two clusters as indicated in (D.5), (D.6) and (D.7).

$$D = \min_{i=1...N_c} \{ \min_{j=i+1...N_c} \frac{d(c_i, c_j)}{\max_{k=1...N_c} diam(C_k)} \} \tag{D.5}$$

$$d(c_i, c_j) = \min_{x \epsilon C_i, y \epsilon C_j} d(x, y) \tag{D.6}$$

$$diam(C_k) = \max_{x, y \epsilon C_k} d(x, y) \tag{D.7}$$

This index yields the relationship between the distance among the clusters and their diameter. The objective is to have compact (i.e. low diameter), well-separated clusters, therefore the objective is to have as large values of the Dunn index as possible.

The Dunn index presents two known drawbacks. These are:

- Considerable amount of time required for computation.
- Very sensitive to noise, due to the diameter computation.

For these reasons, Pal and Biswas proposed in 1997 [359] three variations of the Dunn index, more robust to noise, which are generally referred to as the Dunn-like indices. Their main difference refers to the obtention of the *diam* function, which is obtained applying graph theory principles, and it is based in one of the following three criteria: the minimum spanning tree (MST), the relative neighborhood graph (RNG) and the Gabriel graph.

D.1.3 RMSSDT and RS indices

These indices are used together in hierarchical clustering, in order to evaluate the formation of new branches and groups at each step. However, they can also be used in partitional clustering, since the RMSSDT (*Root Mean Squared Standard Deviation*) index computes the standard deviation of the cluster (see (D.8)), and the RS (*R Squared*) index is a measure of dissimilarity between clusters, obtained as the degree of homogeneity between two groups (see (D.9), (D.10) and (D.11)).

$$RMSSTD = \sqrt{\frac{\sum_{i=1\dots N_c, j=1\dots d} \sum_{k=1}^{n_{ij}} (x_k - \overline{x_j})^2}{\sum_{i=1\dots N_c, j=1\dots d} (n_{ij} - 1)}} \tag{D.8}$$

$$RS = \frac{SS_t - SS_w}{SS_t} \tag{D.9}$$

$$SS_t = \sum_{j=1}^{d} \sum_{k=1}^{n_j} (x_k - \overline{x_j})^2 \tag{D.10}$$

$$SS_w = \sum_{i=1\dots N_c, j=1\dots d} \sum_{k=1}^{n_{ij}} (x_k - \overline{x_j})^2 \tag{D.11}$$

N_c is the number of clusters, d is the number of variables or characteristics (data dimension), n_j is the number of data values of the j dimension, and n_{ij} is the number of data values of the j dimension that belong to cluster i. For the specific case where the number of objects is equal for each dimension, it can be observed that expressions (D.2) and (D.8) become the same measure of dispersion of a cluster.

The optimal number of clusters is selected based on an observation of the values of both indices. The lower the variance in each cluster, the better the clustering formation is considered, therefore the RMSSDT should be as low as possible. On the other hand, the value of the RS index varies between 0 and 1, indicating a 0 that there is no difference between the clusters, and a value of 1 that the clusters are very different. Therefore, the RS index should be as high as possible.

D.1.4 SD validity index

The SD validity index [349][210] is based also on a relation between distance among clusters and the clusters' dispersion or scattering. The formula proposed by the authors for the average scattering of the clusters is given in (D.12), where $\sigma_{(c_i)}$ is the variance of each cluster (see (D.13)), and $\sigma_{(X)}$ is the variance of the whole data set (see (D.14)). It can be seen that expressions (D.2), (D.8) and (D.13) are equivalent measures of cluster dispersion (variance and standard deviation).

$$Scatt = \frac{1}{N_c} \sum_{i=1}^{N_c} \frac{\|\sigma_{(c_i)}\|}{\|\sigma_{(X)}\|} \tag{D.12}$$

$$\sigma_{(c_{ij})} = \frac{1}{\|C_i\|} \sum_{k=1}^{n_{ij}} (x_{kj} - c_{ij})^2 \tag{D.13}$$

$$\sigma_{(x_j)} = \frac{1}{N}\sum_{k=1}^{N}(x_{kj} - \overline{x_j})^2 \tag{D.14}$$

The formula proposed to define the total separation or distance among clusters is expressed in (**??**).

$$Dis = \frac{max_{i,j=1\ldots N_c}(\|c_i - c_j\|)}{min_{i,j=1\ldots N_c}(\|c_i - c_j\|)} \sum_{k=1}^{N_c}\left(\sum_{j=1,i\neq j}^{N_c}\|c_i - c_j\|\right)^{-1} \tag{D.15}$$

Combining the two expressions, the SD index is obtained as indicated in (**??**). The parameter α is a weighting factor, needed in order to normalize the range of $Scatt$ and Dis expressions. Lower values of the SD index indicate better selection of number of clusters, since the compactness of the clusters is bigger and also the separation among them.

$$SD = \alpha Scatt + Dis \tag{D.16}$$

D.2 Partitional fuzzy clustering

D.2.1 PC index

The *partition coefficient* (PC) index was proposed by Bezdek [199]. This index, expressed in (D.17), computes the level of fuzziness of the resulting clusters. The value of the PC index ranges between $1/N_c$ and 1, indicating a value of 1 that the partition is totally non-fuzzy, and a value of $1/N_c$ that the partition is very fuzzified, i.e., most of the objects belong to all the clusters in similar percentages. This result can be due to a non-appropriate selection of the number of clusters, or due to the nature of the data, which do not clearly adhere to a group of different patterns, or these patterns are very similar.

$$PC = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{N_c}\mu_{ij}^2 \tag{D.17}$$

D.2.2 PE index

The *partition entropy* (PE) index was also proposed by Bezdek [199]. This index, expressed in (D.18), measures the level of entropy or fuzziness of a given partition U, where the parameter a is the base of the logarithm.

$$PE = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{N_c}\mu_{ij}\log_a \mu_{ij} \tag{D.18}$$

The PE index ranges between 0 and $\log_a N_c$, indicating a value of 0 that the clusters are well formed and the entropy is minimum, therefore a low value of the PE index is sought.

The PE and PC indices present three known drawbacks, which are the following:

- Their monotonous dependency on the number of clusters. For this reason, the optimal number of clusters is chosen based on variations of the plotted values, rather than on the values themselves.
- Their sensitivity to the fuzzifier value m.
- The lack of a direct connection to the geometry of the data, since the data set is not used in the computation of the indices.

D.2.3 XB index

The Xie-Beni (XB) index [208] involves the fuzzy partitions and the data set. The index relates the compactness of the clusters with the separation among them, as can be seen in (D.19).

$$XB = \frac{\sum_{i=1}^{N_c} \sum_{j=1}^{N} \mu_{ij}^m \|x_j - c_i\|^2}{N \min_{i,k} \|c_i - c_k\|^2} \tag{D.19}$$

The XB index can be seen as a fuzzy equivalent of the DB index, as it is observed when comparing expression (D.19) with (D.1), (D.2), (D.3) and (D.4).

D.2.4 Fukuyama - Sugeno index

The Fukuyama - Sugeno (FS) index [350] combines a measure of fuzziness of the clusters with a measure of fuzziness among the different clusters, as expressed in (D.20) and (D.21).

$$FS = \sum_{i=1}^{N_c} \sum_{j=1}^{N} \mu_{ij}^m \|x_j - c_i\|^2 - \sum_{i=1}^{N_c} \sum_{j=1}^{N} \mu_{ij}^m \|c_i - \overline{c}\|^2 \tag{D.20}$$

$$\overline{c} = \frac{1}{N_c} \sum_{i=1}^{N_c} c_i \tag{D.21}$$

For a good selection of the number of clusters, the values of the FS index should be as low as possible.

D.2.5 Gath and Geva indices

Three different indices were proposed by Gath and Geva in 1989 [103]: the *Fuzzy Hypervolume Validity* index (FHV), defined in (D.22) and (D.23), the *Average Partition Density* (APD), defined in (D.24) and (D.25), and the *Partition Density* index (PD), expressed in (D.26).

$$FHV = \sum_{i=1}^{N_c} \left[det(F_i)\right]^{1/2} \tag{D.22}$$

$$F_i = \frac{\sum_{j=1}^{N} \mu_{ij}^m (x_j - c_i)(x_j - c_i)^T}{\sum_{j=1}^{N} \mu_{ij}^m} \tag{D.23}$$

$$APD = \frac{1}{N_c} \sum_{i=1}^{N_c} \frac{S_i}{\left[det(F_i)\right]^{1/2}} \tag{D.24}$$

$$S_j = \sum_{x \in X_j} \mu_{ij} \tag{D.25}$$

$$PD = \frac{\sum_{i=1}^{N_c} S_i}{FHV} \tag{D.26}$$

The matrix F_i, defined in (D.23), is the fuzzy covariance matrix of cluster i. S_j, where X_j denotes the data set that belongs to cluster j, is the sum of the *central* members of cluster j. Small values of the FHV index indicate compact clusters (low covariances among clusters). APD and PD, on the contrary, should be high to indicate a good selection of the number of clusters.

D.3 Collection of indices

As can be seen, the indices described, either fuzzy and non-fuzzy, give a measure of validity of the partition based on different expressions that measure the dispersion of data inside the clusters and the separation or distances among them. Table D.1 gathers the different indices and indicates the expected values that would imply a good selection of the number of clusters.

Table D.1. Cluster validity indices, for fuzzy and non-fuzzy partitions.

Index	**Value for a good selection**
Fuzzy	
PC	high, close to 1
PE	low, close to 0
XB	low
FS	low
FHV	low
APD	high
PD	high
Non - fuzzy	
DB	low
Dunn	high
RMSSDT	low
RS	high, close to 1
SD	low

E

Methods to Determine Sample Size

E.1 Combination of samples from a given population

The maximum number of combinations of samples from a given population N **with replacement**, i.e., once an individual is chosen, it can be chosen again, is obtained as n (sample size) times the power of N, as expressed in (E.1).

$$C_{N,n} = N^n \tag{E.1}$$

In the case where there is **no replacement**, the maximum number of combinations can be obtained applying the formula described in (E.2), which is the formula for an unordered combination with no replacement.

$$C_{N,n} = \binom{N}{n} = \frac{N!}{n!(N-n)!} = \frac{N(N-1)...(N-n+1)}{n(n-1)...1} \tag{E.2}$$

E.2 Definition of confidence margin

The confidence margin of an estimator of a specific population parameter θ is defined as the region of confidence where an error margin or precision ϵ of the estimation $(\hat{\theta})$ is obtained, with a probability of $(1-\alpha)$, where $(0 < \alpha < 1)$. The mathematical description is expressed in (E.3).

$$P[\hat{\theta} - \epsilon \leq \theta \leq \hat{\theta} + \epsilon] = 1 - \alpha \tag{E.3}$$

In the case that the estimator is unbiased, and is considered to have a normal distribution, Expression (E.3) is reformulated and becomes as expressed in (E.4) and (E.5).

$$P[-Z_{1-\alpha/2} \leq Z \leq Z_{1-\alpha/2}] = 1 - \alpha \tag{E.4}$$

$$Z = \frac{\hat{\theta} - E(\hat{\theta})}{\sigma(\hat{\theta})} = \frac{\hat{\theta} - \theta}{\sigma(\hat{\theta})} \tag{E.5}$$

The term $Z_{1-\alpha/2}\sigma(\hat{\theta})$ is known as the *error term*, and depends of the sample chosen. This term is related to the area of the normal distribution that covers the confidence margin. Usual values of confidence margin are 90%, 95% and 99%, which correspond to the confidence parameter α values of 0.1, 0.05 and 0.01. The resulting values of the confidence interval Z are 1.64, 1.96 and 2.56 respectively.

E.3 Population parameters

There are four parameters which are the most commonly estimated values when defining samples. These are: total (E.6), mean (E.7), class (E.8) and percentage class (E.9).

$$X = \theta(X_1, X_2 ... X_N) = \sum_{i=1}^{N} X_i \tag{E.6}$$

$$\bar{X} = \theta(X_1, X_2 ... X_N) = \frac{1}{N} \sum_{i=1}^{N} X_i \tag{E.7}$$

$$A = \theta(A_1, A_2 ... A_N) = \sum_{i=1}^{N} A_i \tag{E.8}$$

$$P = \theta(A_1, A_2 ... A_N) = \frac{1}{N} \sum_{i=1}^{N} A_i \tag{E.9}$$

E.4 Sample parameters

The probability of first inclusion of an object of the population in the sample, i.e., the probability to belong to the sample, is indicated in Eq. (E.10).

$$\pi_i = P(u_i \epsilon s) = \frac{n}{N} \tag{E.10}$$

Given the population parameters for estimation, the estimation errors from a sample can be obtained from the variance, as indicated in Section E.2. For each of the four estimators, the estimated values from a sample are expressed in Eqs. (E.11) to (E.14) and their respective variances are expressed in Eqs. (E.15) to (E.18).

$$\hat{X} = \sum_{i=1}^{n} \frac{X_i}{\pi_i} = \frac{N}{n} \sum_{i=1}^{n} X_i = N\hat{\bar{X}} \tag{E.11}$$

$$\hat{\bar{X}} = \sum_{i=1}^{n} \frac{X_i}{nN/N} = \frac{1}{n} \sum_{i=1}^{n} X_i \tag{E.12}$$

$$\hat{A} = \sum_{i=1}^{n} \frac{A_i}{n/N} = N \frac{1}{n} \sum_{i=1}^{n} A_i \tag{E.13}$$

$$\hat{P} = \sum_{i=1}^{n} \frac{A_i/N}{n/N} = \frac{1}{n} \sum_{i=1}^{n} A_i \tag{E.14}$$

$$Var(\hat{X}) = N^2(1-f)\frac{S^2}{n} \tag{E.15}$$

$$Var(\hat{\bar{X}}) = (1-f)\frac{S^2}{n} \tag{E.16}$$

$$Var(\hat{A}) = \frac{N^3}{N-1}\frac{1}{n}(1-f)PQ \tag{E.17}$$

$$Var(\hat{P}) = \frac{N}{N-1}\frac{1}{n}(1-f)PQ \tag{E.18}$$

Where S^2 is the *quasi-variance* of the population (E.19), f is the sample fraction (E.20), P is the class proportion (E.9) and Q is the total proportion of the other classes $(1-P)$.

$$S^2 = \frac{1}{N-1} \sum_{i=1}^{N} (X_i - \bar{X})^2 \tag{E.19}$$

$$f = \frac{n}{N} \tag{E.20}$$

E.5 Sample size with sampling error and confidence margin

To determine the sample size, the formula for the error term in a normal distribution is considered, $e_\alpha = Z_{1-\alpha/2}\sigma(\hat{\theta})$, as seen in Section E.2. From this formula, the value of n can be obtained for all the estimators, taking into account that the term $\sigma(\hat{\theta})$ refers to the standard deviation of the estimated values from a sample, i.e., $\sigma(\hat{\theta}) = \sqrt{Var(\hat{\theta}}$. The resulting formulas for the four estimators considered can be find below: total population (E.21), mean (E.22), total class (E.23) and percentage class (E.24).

$$n = \frac{Z^2_{1-\alpha/2} N^2 S^2}{e^2_\alpha + Z^2_{1-\alpha/2} S^2 N} \tag{E.21}$$

$$n = \frac{Z^2_{1-\alpha/2} N S^2}{N e^2_\alpha + Z^2_{1-\alpha/2} S^2} \tag{E.22}$$

$$n = \frac{Z^2_{1-\alpha/2} N^3 PQ}{(N-1) e^2_\alpha + Z^2_{1-\alpha/2} N^2 PQ} \tag{E.23}$$

$$n = \frac{Z^2_{1-\alpha/2} NPQ}{(N-1) e^2_\alpha + Z^2_{1-\alpha/2} PQ} \tag{E.24}$$

E.6 Stratified sample

A stratified sample is obtained by dividing the sample in subgroups, following the same proportion of the population, as expressed in (E.25).

$$n_i = \frac{N_i}{N} n \tag{E.25}$$

E.7 Example of sample sizes: database of electric energy consumption

According to the most recent information from the Spanish National Commission of Energy (CNE), type a and type b customer numbers in Spain are the following:

- Type *a* customers: 24.835.412 ($92,98\%$ from total)
- Type *b* customers: 1.165.001 ($4,36\%$ from total)
- Type *c* customers: 709.276 ($2,66\%$ from total)

Of a total of 26.709.689 customers. The database analyzed in the present work is comprised of load profiles from 759 clients, being 711 of them of type *a*, 44 of type *b* and 4 of type *c*.

Adequate samples that represent in percentage the total population [351][360] can be computed applying the formula described in Eq. (E.26), where n, N, t, p and e stand for:

- n: is the sample size, that represents the population percentage.
- N: is the total population number, for a known, finite size.
- t: confidence interval parameter, obtained as a given value of confidence α. A usual value of α is $0,05$, or the 95% of confidence. For this value, the value of the t parameter equals $1,96$.

- p: is the expected proportion in the sample, i.e., $92,98\%$ in the case of type a, $4,36\%$ in the case of type b, and $2,66\%$ in the case of type c.
- e: is the expected error in the sample. A usual value is to assume an error in the sample of 3%, therefore a value of $e = 0.03$ has been used.

$$n = \frac{Nt^2p(1-p)}{(N-1)e^2 + t^2p(1-p)} \tag{E.26}$$

Assuming the preceding values, the resulting sample sizes for the three types of clients are:

- Type a: sample of 279 clients.
- Type b: sample of 178 clients.
- Type c: sample of 111 clients.

From these results it can be concluded that the sample is adequate for clients of type a, and inadequate for clients of types b and c, if no other stratification variables are taken into account, such as the presence of sufficient samples from the different climate regions of Spain. However, this information is not used for the segmentation, but to extract conclusions from its results.

References

[1] MIT Energy Initiative. The Future of the Electric Grid. Technical report, MIT (Massachusets Institute of Technology), 2001.

[2] Felix A. Farret and M. Godoy Simoes. *Integration of Alternative Sources of Energy*. Wiley Interscience, 2006.

[3] Eric Hirst and Brendan Kirby. Electric Power Ancillary Services. Technical report, Oak Ridge National Laboratory, 1996.

[4] Daniel Kirschen and Goran Strbac. *Fundamentals of Power System Economics*. John Wiley & Sons, Ltd, 2004.

[5] EC. Directive 2009/72/EC of the European Parliament and of the Council of 13 July 2009 concerning common rules for the internal market in electricity and repealing Directive 2003/54/EC. Official Journal of the European Union, August 2009.

[6] Benchmarking smart metering deployment in the EU-27 with a focus on electricity. Technical report, European Commission, June 2014.

[7] Wenye Wang and Zhuo Lu. Cyber Security in the Smart Grid: Survey and Challenges. *Comput. Netw.*, 57(5):1344–1371, April 2013.

[8] EC. Quarterly Report on European Electricity Markets. Technical report, Market Observatory for Energy – DG Energy, 2014.

[9] Red Eléctrica de España. Electricity interconnections: a step forward towards a single integrated European energy market. Technical report, 2012.

[10] Anna Creti, Eileen Fumagalli, and Elena Fumagalli. Integration of electricity markets in Europe: Relevant issues for Italy. *Energy Policy*, 38(11):6966 – 6976, 2010.

[11] Rouquia Djabali, Joel Hoeksema, and Yves Langer. COSMOS description – CWE Market Coupling algorithm. Technical report, CWE, 2011.

[12] Stephen Boyd, Arpita Ghosh, and Alessandro Magnani. Branch and Bound Methods. Stanford University. Lecture Notes, November 2003.

[13] Spain. Ley 54/1997, de 27 de noviembre, del Sector Eléctrico. BOE, November 1997. No. 285, pages 35097 – 35126.

[14] Spain. Ley 24/2013, de 26 de diciembre, del Sector Eléctrico. BOE, December 2013. No. 310, pages 105198 – 105294.

[15] EC. Communication from the Commission to the European Parliament, the Council, the European Economic and Social Committee and the Committee of the Regions. Launching the public consultation process on a new energy market design. Communication, July 2015.

[16] Accomodating High Levels of Variable Generation. Technical report, NERC (North American Electric Reliability Corporation), 2009.

[17] The Harmonised Electricity Market Role Model. Technical report, ENTSO-E, 2014.

[18] M. C. Alvarez-Herault, N. N'Doye, C. Gandioli, N. Hadjsaid, and P. Tixador. Meshed distribution network vs reinforcement to increase the distributed generation connection. *Sustainable Energy, Grids and Networks*, 1:20 – 27, 2015.

[19] Xi Fang, Satyajayant Misra, Guoliang Xue, and Dejun Yang. Smart Grid – The New and Improved Power Grid: A Survey. *Communications Surveys Tutorials, IEEE*, 14(4):944–980, Fourth 2012.

[20] EC. Standardization Mandate to European Standardisation Organisations (ESOs) to support European Smart Grid deployment. M/490., March 2011.

[21] Sean Dempsey. Myism3004 blog. Smart Grids. Weblog, 2011.

[22] Working Group D2.31. Security architecture principles for digital systems in Electric Power Utilities. Technical report, CIGRÉ (International Council on Large Electric Systems), 2015.

[23] David Trebolle Trebolle. La Generación Distribuida en España. Master's thesis, Universidad Pontificia Comillas, 2006. In Spanish.

[24] Thomas Ackermann, Göran Andersson, and Lennart Söder. Distributed generation: a definition. *Electric Power Systems Research*, 57(3):195 – 204, 2001.

[25] IEA. Distributed Generation in Liberalised Energy Markets. Technical report, International Energy Agency (IEA), 2002.

[26] P. Crossley D. Kirschen N. Jenkins, R. Allan and G. Strbac. *Embedded Generation*. The Institution of Engineering and Technology, 2000.

[27] Irene Aguado. Impacto de la Generación Distribuida en Redes de Distribución de MT. Master's thesis, Universitat Politècnica de Vaència, 2013. In Spanish.

[28] Francisco José Pazos. Operational Experience and Field Tests on Islanding Events Caused by Large Photovoltaic Plants. In *Proceedings of the 21st International Conference on Electricity Distribution - CIRED 2011*, 2011.

[29] A. Quijano, J. Monreal, I. Benítez, and I. Delgado. Assessing the reasons for islanding in rural networks with dispersed photovoltaic generation. In *CIGRE Lisbon Symposium*, 2013.

[30] Glossary of Terms for Technical Requirements for Ancillary Services. Technical report, AESO (Alberta Electric System Operator), 2004.

[31] E. Hirst and B. Kirby. Ancillary Services. In *American Power Conference*, Chicago, Illinois., 1996.

[32] Brendan J. Kirby. Frequency Regulation Basics and Trends. Technical report, Oak Ridge National Laboratory, 2004.

[33] M. Milligan Y. Wan and B. Kirby. Impact of Energy Imbalance Tariff on Wind Energy. In *AWEA (American Wind Energy Association) WindPower 2007 Conference*, 2007.

[34] Alberto Carbajo. Los Mercados Eléctricos y los Servicios de Ajuste del Sistema. *Economía Industrial*, 364:55–62, 2007. In Spanish.

[35] EC. Standardisation Mandate to CEN, CENELEC and ETSI in the field of measuring instruments for the development of an open architecture for utility meters involving communication protocols enabling interoperability. M/441., March 2009.

[36] EC. Standardisation Mandate to CEN, CENELEC and ETSI concerning the charging of electric vehicles. M/468., June 2010.

[37] Functional reference architecture for communications in smart metering systems. Technical report, CEN, CENELEC and ETSI, 2011.

[38] IEC (International Electrotechnical Commission). IEC 62056. Electricity metering data exchange - the DLMS/COSEM suite, 2014.

[39] Specification for PoweRline Intelligent Metering Evolution (PRIME). Technical report, PRIME Alliance Technical Working Group, 2014.

[40] Van Laere Aurélien, Diakiese Tresor, Bette Sébastien, and Moeyaert Véronique. PERFORMANCE OF THE G3-PLC COMMUNICATION LINKS. In *Proceedings of CIRED Workshop*, 2014.

[41] Marco Cotti and Rocío Millán. Cervantes Project and Meters and More: The State of the Art of Smart Metering Implementation in Europe. In *Proceedings of the 2011 CIRED Conference*, 2011.

[42] IEC (International Electrotechnical Commission). IEC 61851-1:2010. Electric vehicle conductive charging system - Part 1: General requirements, 2010.

[43] IEC (International Electrotechnical Commission). IEC 62196-1:2014. Plugs, socket-outlets, vehicle connectors and vehicle inlets - Conductive charging of electric vehicles - Part 1: General requirements, 2014.

[44] IEC / ISO 15118-1:2013. Road vehicles – Vehicle to grid communication interface – Part 1: General information and use-case definition, 2013.

[45] eMobilityCG and SmartGridCG. E - Mobility Smart Charging. Report about Smart Charging of Electric Vehicles in relation to Smart Grids. Technical report, CEN, CENELEC, ETSI, 2013.

[46] Open Charge Point Protocol 2.0. Interface description between Charge Point and Central System. Technical report, Open Charge Alliance (OCA) Technology Working Group, 2014.

[47] SGAM User Manual – Applying, testing & refining the Smart Grid Architecture Model (SGAM). Technical report, CEN-CENELEC-ETSI Smart Grid Coordination Group, 2014.

[48] Common Information Model Primer: First Edition. Technical report, Electric Power Research Institute (EPRI), 2011.

[49] IEC (International Electrotechnical Commission). IEC 61850:2015 Series. Communication networks and systems for power utility automation - ALL PARTS, 2015.

[50] Smart Grid Coordination Group (SGCG) Report on Reference Architecture for Smart Grid. Technical report, CEN, CENELEC, ETSI, 2012.

[51] IEC (International Electrotechnical Commission). IEC TR 62357-1:2012. Power systems management and associated information exchange - Part 1: Reference architecture, 2012.

[52] R. Mackiewicz. Overview of IEC 61850 and benefits. *Power Engineering Society General Meeting*, 2006.

[53] ISO (International Organization for Standardization). ISO 9506-1:2003. Industrial automation systems – Manufacturing Message Specification – Part 1: Service definition, 2003.

[54] The Common Information Model for Distribution: An Introduction to the CIM for Integrating Distribution Applications and Systems. Technical report, Electric Power Research Institute (EPRI), 2008.

[55] IEC (International Electrotechnical Commission). IEC 61970-1:2005. Energy management system application program interface (EMS-API) - Part 1: Guidelines and general requirements, 2005.

[56] IEC (International Electrotechnical Commission). IEC 61968-1:2012. Application integration at electric utilities - System interfaces for distribution management - Part 1: Interface architecture and general recommendations, 2012.

[57] Dragan S. Markovic, Dejan Zivkovic, Irina Branovic, Ranko Popovic, and Dragan Cvetkovic. Smart power grid and cloud computing. *Renewable and Sustainable Energy Reviews*, 24:566 – 577, 2013.

[58] C.L. Philip Chen and Chun-Yang Zhang. Data-intensive applications, challenges, techniques and technologies: A survey on Big Data. *Information Sciences*, 275:314 – 347, 2014.

[59] Han Hu, Yonggang Wen, Tat-Seng Chua, and Xuelong Li. Toward Scalable Systems for Big Data Analytics: A Technology Tutorial. *Access, IEEE*, 2:652–687, 2014.

[60] Enrico Barbierato, Marco Gribaudo, and Mauro Iacono. Performance evaluation of NoSQL big-data applications using multi-formalism models. *Future Generation Computer Systems*, 37:345 – 353, 2014.

[61] Karthik Kambatla, Giorgos Kollias, Vipin Kumar, and Ananth Grama. Trends in Big Data analytics. *Journal of Parallel and Distributed Computing*, 74(7):2561 – 2573, 2014. Special Issue on Perspectives on Parallel and Distributed Processing.

[62] Brett Sargent. Big Data = Big Challenges for Big Substations and Big Assets. In *Proceedings of the 2013 CIGRÉ Canada Conference*, 2013.

[63] Ibrahim Abaker Targio Hashem, Ibrar Yaqoob, Nor Badrul Anuar, Salimah Mokhtar, Abdullah Gani, and Samee Ullah Khan. The rise of Big

Data on cloud computing: Review and open research issues. *Information Systems*, 47:98 – 115, 2015.

[64] Mike Olson. HADOOP: Scalable, Flexible Data Storage and Analysis. *IQT Quarterly*, 1(3):14–18, 2010.

[65] Benefits of Demand Response in Electricity Markets and Recommendations for Achieving them. Technical report, U.S. Department of Energy, 2006.

[66] I. Navalón, S. Bañares, L. Moreno, and A. Quijano. DSM in Spain. GAD Project. Aims, Developments and Initial Results. In *CIGRÉ SESSION*, 2010.

[67] Implementing Agreement on Demand-Side Management Technologies and Programmes. Annual Report. Technical report, International Energy Agency - IEA DSM, 2014.

[68] Demand Response Map in Europe 2014/2015. Technical report, Smart Energy Demand Coalition, 2015.

[69] Consolidated View of the ETP SG (European Technology Platform on Smart Grids) on Research, Development & Demonstration Needs in the Horizon 2020 Work Programme 2016–2017. Technical report, ETP SG, 2015.

[70] EC. Communication from the Commission to the European Parliament, the Council, the European Economic and Social Committee and the Committee of the Regions. Horizon 2020 - the Framework Programme for Research and Innovation. Communication, November 2011.

[71] I. Benítez Sánchez, I. Delgado Espinós, L. Moreno Sarrión, A. Quijano López, and I. Navalón Burgos. Clients segmentation according to their domestic energy consumption by the use of self-organizing maps. In *EEM 2009. 6th International Conference on the European Energy Market, 2009.*, pages 1–6, 2009.

[72] David Hand, Heikki Mannila, and Padhraic Smyth. *Principles of Data Mining*. The MIT Press, 2001.

[73] Jiawei Han and Micheline Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann, 2006.

[74] Sushmita Mitra, Sankar K. Pal, and Pabitra Mitra. Data Mining in Soft Computing Framework: A Survey. *IEEE Transactions on Neural Networks*, 13(1):3–14, January 2002.

[75] Chun-Wei Tsai, Chin-Feng Lai, Ming-Chao Chiang, and L.T. Yang. Data Mining for Internet of Things: A Survey. *Communications Surveys Tutorials, IEEE*, 16(1):77–97, First 2014.

[76] Matthew Herland, Taghi M. Khoshgoftaar, and Randall Wald. A review of data mining using Big Data in health informatics. *Journal Of Big Data*, 1(2), 2014.

[77] Surajit Chaudhuri and Umeshwar Dayal. An overview of data warehousing and OLAP technology. *SIGMOD Rec.*, 26(1):65–74, 1997.

[78] Anil K. Jain and Richard C. Dubes. *Algorithms for clustering data.* Prentice Hall, New Jersey, 1988.

[79] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics).* Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[80] Xindong Wu, Xingquan Zhu, Gong-Qing Wu, and Wei Ding. Data mining with Big Data. *Knowledge and Data Engineering, IEEE Transactions on*, 26(1):97–107, Jan 2014.

[81] G. Raschia and N. Mouaddib. SAINTETIQ: a fuzzy set-based approach to database summarization. *Fuzzy Sets Syst.*, 129(2):137–162, 2002.

[82] Minqing Hu and Bing Liu. Mining and summarizing customer reviews. In *KDD '04: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177, New York, NY, USA, 2004. ACM Press.

[83] Rakesh Agrawal, Tomasz Imieli, and Arun Swami. Mining association rules between sets of items in large databases. In *SIGMOD '93: Proceedings of the 1993 ACM SIGMOD international conference on Management of data*, pages 207–216, New York, NY, USA, 1993. ACM Press.

[84] T. Kohonen. *Self-Organizing Maps.* Springer, 2001.

[85] Y. Q. Zhang, M. D. Fraser, R. A. Gagliano, and A. Kandel. Granular Neural Networks for Numerical-Linguistic Data Fusion and Knowledge Discovery. *IEEE Transactions on Neural Networks*, 11(3):658–667, May 2000.

[86] V. Figueiredo, F. Rodrigues, Z. Vale, and J.B. Gouveia. An electric energy consumer characterization framework based on data mining techniques. *Power Systems, IEEE Transactions on*, 20(2):596–602, 2005.

[87] Florian Markowetz, Lutz Edler, and Martin Vingron. Support Vector Machines for Protein Fold Class Prediction. *Biometrical Journal*, 45(3):377–389, 2003.

[88] I-Jen Chiang and Jane Yung jen Hsu. Fuzzy classification trees for data analysis. *Fuzzy Sets Syst.*, 130(1):87–99, 2002.

[89] Michael Goebel and Le Gruenwald. A survey of data mining and knowledge discovery software tools. *SIGKDD Explor. Newsl.*, 1(1):20–33, 1999.

[90] Jose Valente de Oliveira and Witold Pedrycz, editors. *Advances in Fuzzy Clustering and its Applications.* John Wiley & Sons, Ltd., 2007.

[91] Tao Li, Qi Li, Shenghuo Zhu, and Mitsunori Ogihara. A survey on wavelet applications in data mining. *SIGKDD Explor. Newsl.*, 4(2):49–68, 2002.

[92] Chang-Tien Lu and Lily R. Liang. Wavelet fuzzy classification for detecting and tracking region outliers in meteorological data. In *GIS '04: Proceedings of the 12th annual ACM international workshop on Geographic information systems*, pages 258–265, New York, NY, USA, 2004. ACM.

[93] S. Chandrakala and C.C. Sekhar. A density based method for multivariate time series clustering in kernel feature space. In *Neural Networks, 2008. IJCNN 2008. (IEEE World Congress on Computational Intelli-*

gence). IEEE International Joint Conference on, pages 1885–1890, June 2008.

[94] D.V.; Wunsch D.C. II Saad, E.W.; Prokhorov. Comparative study of stock trend prediction using time delay, recurrent and probabilistic neural networks. *Neural Networks, IEEE Transactions on*, 9(6):1456–1470, 1998.

[95] Lofti Zadeh. Fuzzy Sets. *Journal of Information and Control*, 8:338–353, 1965.

[96] Lotfi A. Zadeh. Outline of a New Approach to the Analysis of Complex Systems and Decision Processes. *Systems, Man and Cybernetics, IEEE Transactions on*, SMC-3(1):28–44, Jan 1973.

[97] Chun-Yueh Huang, Chuen-Yau Chen, and Bin-Da Liu. Current-Mode Fuzzy Linguistic Hedge Circuits. *Analog Integr. Circuits Signal Process.*, 19(3):255–278, 1999.

[98] Tariq Rashid. Clustering of Fuzzy Image Features. Technical report, University of Bristol, 2003.

[99] Pang-Ning Tan and Rong Jin. Ordering patterns by combining opinions from multiple sources. In *KDD '04: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 695–700, New York, NY, USA, 2004. ACM Press.

[100] E.H. Mamdani and S. Assilian. An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-Machine Studies*, 7(1):1 – 13, 1975.

[101] Michio Sugeno. *Industrial Applications of Fuzzy Control.* Elsevier Science Inc., New York, NY, USA, 1985.

[102] E. E. Gustafson and W. C. Kessel. Fuzzy clustering with a fuzzy covariance matrix. In *Proc. of the IEEE Conference on Decision and Control*, pages 761–766, San Diego, CA, 1979.

[103] I. Gath and A.B. Geva. Unsupervised optimal fuzzy clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11:773–781, 1989.

[104] W. Pedrycz and J. Waletzky. Fuzzy clustering with partial supervision. *Systems, Man and Cybernetics, Part B, IEEE Transactions on*, 27(5):787–795, 1997.

[105] Tamas F.D., Abonyi J., Borszeki J., and Halmos P. Trace elements in clinker - II. Qualitative identification by fuzzy clustering. *Cement and Concrete Research*, 32:1325–1330, 2002.

[106] J. Dunn. A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters. *Journal of Cybernetics*, 3:32–57, 1974.

[107] Keith C. C. Chan and Wai-Ho Au. Mining fuzzy association rules. In *CIKM '97: Proceedings of the sixth international conference on Information and knowledge management*, pages 209–215, New York, NY, USA, 1997. ACM Press.

[108] J. C. Cubero, J. M. Medina, O. Pons, and M. A. Vila. Data summarization in relational databases through fuzzy dependencies. *Inf. Sci.*, 121(3-4):233–270, 1999.

[109] E. F. Codd. A Relational Model of Data for Large Shared Data Banks. *Commun. ACM*, 13(6):377–387, June 1970.

[110] L.A. Zadeh. The concept of a linguistic variable and its application to approximate reasoning. *Information Sciences*, 8(3):199 – 249, 1975.

[111] Cunyong Qiu, Jian Xiao, Lu Han, and Muhammad Naveed Iqbal. Enhanced interval type-2 fuzzy c-means algorithm with improved initial center. *Pattern Recognition Letters*, 38(0):86 – 92, 2014.

[112] S. Haykin. *Neural Networks, a Comprehensive Foundation.* MacMillan, 1994.

[113] Lakhmi C. Jain and N. M. Martin. *Fusion of Neural Networks, Fuzzy Sets, and Genetic Algorithms: Industrial Applications.* CRC Press, Inc., Boca Raton, FL, USA, 1st edition, 1998.

[114] WarrenS. McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943.

[115] Donald Hebb. *The Organization of Behavior.* 1949.

[116] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, November 1958.

[117] John von Neumann. *The Computer and the Brain.* Yale University Press, New Haven, CT, USA, 1958.

[118] Bernard Widrow and Marcian E. Hoff. Adaptive Switching Circuits. In *1960 IRE WESCON Convention Record, Part 4*, pages 96–104, New York, 1960. IRE.

[119] J. J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. In *Proceedings of the National Academy of Sciences of the United States of America*, volume 79, pages 2554 – 2558, 1982.

[120] Ku-Long Ho, Yuan-Yih Hsu, and Chien-Chuen Yang. Short term load forecasting using a multilayer neural network with an adaptive learning algorithm. *Power Systems, IEEE Transactions on*, 7(1):141–149, Feb 1992.

[121] G. Chicco, R. Napoli, F. Piglione, P. Postolache, M. Scutariu, and C. Toader. Emergent electricity customer classification. *Generation, Transmission and Distribution, IEEE Proceedings-*, 152(2):164–172, 2005.

[122] Ignacio Benítez, Laura Moreno, Andrés Lluna, and Alfredo Quijano. Prediction of Optimal Meteorological Conditions for Electrical Energy Generation by the Use of Artificial Neural Networks. In *3rd International Conference on Integration of Renewable and Distributed Energy Resources. Conference Abstracts*, 2008.

[123] Raúl Rojas. *Neural Networks: A Systematic Introduction.* Springer-Verlag New York, Inc., New York, NY, USA, 1996.

[124] Guang-Bin Huang, Qin-Yu Zhu, and Chee-Kheong Siew. Extreme learning machine: Theory and applications. *Neurocomputing*, 70:489 – 501, 2006. Neural Networks Selected Papers from the 7th Brazilian Symposium on Neural Networks (SBRN '04) 7th Brazilian Symposium on Neural Networks.

[125] Changhao Xia, Jian Wang, and Karen McMenemy. Short, medium and long term load forecasting model and virtual load forecaster based on radial basis function neural networks. *International Journal of Electrical Power & Energy Systems*, 32(7):743 – 750, 2010.

[126] Danilo P. Mandic and Jonathon Chambers. *Recurrent Neural Networks for Prediction: Learning Algorithms,Architectures and Stability.* John Wiley &amp; Sons, Inc., New York, NY, USA, 2001.

[127] Robert Andrews, Joachim Diederich, and Alan B. Tickle. Survey and critique of techniques for extracting rules from trained artificial neural networks. *Knowledge-Based Systems*, 8(6):373 – 389, 1995. Knowledge-based neural networks.

[128] A.B. Tickle, R. Andrews, M. Golea, and J. Diederich. The truth will come to light: directions and challenges in extracting the knowledge embedded within trained artificial neural networks. *Neural Networks, IEEE Transactions on*, 9(6):1057–1068, Nov 1998.

[129] G. Chicco, Roberto Napoli, and Federico Piglione. Application of clustering algorithms and self organising maps to classify electricity customers. In *Power Tech Conference Proceedings, 2003 IEEE Bologna*, volume 1, pages 7 pp. Vol.1–, 2003.

[130] S.V. Verdu, M.O. Garcia, F.J.G. Franco, N. Encinas, A.G. Marin, A. Molina, and E.G. Lazaro. Characterization and identification of electrical customers through the use of self-organizing maps and daily load parameters. In *Power Systems Conference and Exposition, 2004. IEEE PES*, pages 899–906 vol.2, 2004.

[131] Geoffrey K.F. Tso and Kelvin K.W. Yau. Predicting electricity energy consumption: A comparison of regression analysis, decision tree and neural networks. *Energy*, 32(9):1761 – 1768, 2007.

[132] H.S. Hippert, C.E. Pedreira, and R.C. Souza. Neural networks for short-term load forecasting: a review and evaluation. *Power Systems, IEEE Transactions on*, 16(1):44–55, Feb 2001.

[133] Zdzislaw Pawlak, Jerzy Grzymala-Busse, Roman Slowinski, and Wojciech Ziarko. Rough sets. *Commun. ACM*, 38(11):88–95, 1995.

[134] Witold Pedrycz, Andrzej Skowron, and Vladik Kreinovich. *Handbook of Granular Computing.* Wiley-Interscience, New York, NY, USA, 2008.

[135] Shusaku Tsumoto. Incremental Rule Induction Based on Rough Set Theory. In Marzena Kryszkiewicz, Henryk Rybinski, Andrzej Skowron, and ZbigniewW. R., editors, *Foundations of Intelligent Systems*, volume

6804 of *Lecture Notes in Computer Science*, pages 70–79. Springer Berlin Heidelberg, 2011.

[136] Piotr Synak. Temporal Templates and Analysis of Time Related Data. In Wojciech Ziarko and Yiyu Yao, editors, *Rough Sets and Current Trends in Computing*, volume 2005 of *Lecture Notes in Computer Science*, pages 420–427. Springer Berlin Heidelberg, 2001.

[137] John H. Holland. *Adaptation in Natural and Artificial Systems.* MIT Press, Cambridge, MA, USA, 1992.

[138] Amparo Mocholí, Carlos Blasco, Irene Aguado, and Vicente Fuster. Software for the Optimal Allocation of EV Chargers in the Power Distribution Grid. In *Proceedings of the 23rd International Conference on Electricity Distribution - CIRED*, 2015.

[139] Kebin Xu, Zhenyuan Wang, and Kwong-Sak Leung. Using a new type of nonlinear integral for multi-regression: an application of evolutionary algorithms in data mining. In *Systems, Man, and Cybernetics, 1998. 1998 IEEE International Conference on*, volume 3, pages 2326–2331 vol.3, Oct 1998.

[140] E. Noda, A.A. Freitas, and H.S. Lopes. Discovering interesting prediction rules with a genetic algorithm. In *Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on*, volume 2, pages –1329 Vol. 2, 1999.

[141] P.P. Wakabi-Waiswa, V. Baryamureeba, and K. Sarukesi. Optimized Association Rule Mining with genetic algorithms. In *Natural Computation (ICNC), 2011 Seventh International Conference on*, volume 2, pages 1116–1120, July 2011.

[142] B. Minaei-Bidgoli, R. Barmaki, and M. Nasiri. Mining numerical association rules via multi-objective genetic algorithms. *Information Sciences*, 233:15 – 24, 2013.

[143] Agnar Aamodt and Enric Plaza. Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI communications*, 7(1):39–59, 1994.

[144] Kai Bartlmae. Optimizing Data-Mining Processes: A CBR Based Experience Factory for Data Mining. In LucasChiKwong Hui and Dik-Lun Lee, editors, *Internet Applications*, volume 1749 of *Lecture Notes in Computer Science*, pages 21–30. Springer Berlin Heidelberg, 1999.

[145] Zoe Y. Zhuang, Leonid Churilov, Frada Burstein, and Ken Sikaris. Combining data mining and case-based reasoning for intelligent decision support for pathology ordering by general practitioners. *European Journal of Operational Research*, 195(3):662 – 675, 2009.

[146] Ross Quinlan and J.R. Quinlan. *C4.5: Programs for Machine Learning.* Morgan Kaufmann, 1992.

[147] Tian Zhang, Raghu Ramakrishnan, and Miron Livny. BIRCH: an efficient data clustering method for very large databases. In *SIGMOD '96: Proceedings of the 1996 ACM SIGMOD international conference*

on Management of data, pages 103–114, New York, NY, USA, 1996. ACM Press.

[148] Takashi Washio and Hiroshi Motoda. State of the art of graph-based data mining. *SIGKDD Explor. Newsl.*, 5(1):59–68, 2003.

[149] Marco Ramoni, Paola Sebastiani, and Paul Cohen. Bayesian Clustering by Dynamics. *Mach. Learn.*, 47(1):91–121, April 2002.

[150] V. Mihajlovic and Petkovic. Dynamic Bayesian Networks: A State of the Art. Technical report, University of Twente, 2001.

[151] Christian Robert, Tobias Ryden, and D. M. Titterington. Bayesian inference in hidden Markov models through the reversible Markov chain Monte Carlo method. *Royal statistical society*, 2000.

[152] Amara Graps. An introduction to Wavelets. *IEEE Computational Science and Engineering*, 8(2), 1995.

[153] Gholamhosein Sheikholeslami, Surojit Chatterjee, and Aidong Zhang. WaveCluster: a wavelet-based clustering approach for spatial data in very large databases. *The VLDB Journal*, 8(3-4):289–304, 2000.

[154] M.S.B. PhridviRaj and C.V. GuruRao. Data Mining – Past, Present and Future – A Typical Survey on Data Streams. *Procedia Technology*, 12:255 – 263, 2014. The 7th International Conference Interdisciplinarity in Engineering, INTER-ENG 2013, 10-11 October 2013, Petru Maior University of Tirgu Mures, Romania.

[155] Ivan F. Videla-Cavieres and Sebastián A. Ríos. Extending market basket analysis with graph mining techniques: A real case. *Expert Systems with Applications*, 41(4, Part 2):1928 – 1936, 2014.

[156] N. Gupta and M.L. Yadav. An implementation and analysis of DSR using market basket analysis to improve the sales of business. In *Confluence The Next Generation Information Technology Summit (Confluence), 2014 5th International Conference* -, pages 82–86, Sept 2014.

[157] Gurmeet Singh Manku and Rajeev Motwani. Approximate Frequency Counts over Data Streams. In *Proceedings of the 28th International Conference on Very Large Data Bases*, VLDB '02, pages 346–357. VLDB Endowment, 2002.

[158] Pedro Domingos and Geoff Hulten. Mining High-speed Data Streams. In *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '00, pages 71–80, New York, NY, USA, 2000. ACM.

[159] Geoff Hulten, Laurie Spencer, and Pedro Domingos. Mining Time-changing Data Streams. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '01, pages 97–106, New York, NY, USA, 2001. ACM.

[160] Sudipto Guha, Adam Meyerson, Nina Mishra, Rajeev Motwani, and Liadan O'Callaghan. Clustering Data Streams: Theory and Practice. *IEEE Trans. on Knowl. and Data Eng.*, 15(3):515–528, March 2003.

[161] Charu C. Aggarwal. A Survey of Stream Clustering Algorithms. In *Data Clustering: Algorithms and Applications*, pages 231–258. 2013.

[162] Charu C. Aggarwal, Jiawei Han, Jianyong Wang, and Philip S. Yu. A Framework for Clustering Evolving Data Streams. In *Proceedings of the 29th International Conference on Very Large Data Bases - Volume 29*, VLDB '03, pages 81–92. VLDB Endowment, 2003.

[163] Ramakrishnan Srikant and Rakesh Agrawal. Mining Sequential Patterns: Generalizations and Performance Improvements. In *Proceedings of the 5th International Conference on Extending Database Technology: Advances in Database Technology*, EDBT '96, pages 3–17, London, UK, UK, 1996. Springer-Verlag.

[164] Mohammed J. Zaki. SPADE: An Efficient Algorithm for Mining Frequent Sequences. *Mach. Learn.*, 42(1-2):31–60, January 2001.

[165] Jian Pei, Jiawei Han, B. Mortazavi-Asl, H. Pinto, Qiming Chen, U. Dayal, and Mei-Chun Hsu. PrefixSpan: Mining Sequential Patterns Efficiently by Prefix-Projected Pattern Growth. In *Proceedings of the 17th International Conference on Data Engineering*, ICDE '01, pages 215–224, Washington, DC, USA, 2001. IEEE Computer Society.

[166] José Luis Díez. *Técnicas de agrupamiento para identificación y control por modelos locales*. PhD thesis, Universidad Politécnica de Valencia, Julio 2003. In Spanish.

[167] R. Likert. A technique for the measurement of attitudes. *Archives of Psychology*, 22(140):1–55, 1932.

[168] Nikos Mamoulis, David W. Cheung, and Wang Lian. Similarity Search in Sets and Categorical Data Using the Signature Tree. In *Proceedings of the 19th International Conference on Data Engineering (ICDE 03)*, pages 75–86. IEEE, 2003.

[169] R.W. Hamming. Error Detecting and Error Correcting Codes. *Bell System Technical Journal*, 26(2):147–160, 1950.

[170] C.E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, pages 379–423, 1948.

[171] Daniel Barbará, Yi Li, and Julia Couto. COOLCAT: an entropy-based algorithm for categorical clustering. In *CIKM '02: Proceedings of the eleventh international conference on Information and knowledge management*, pages 582–589, New York, NY, USA, 2002. ACM Press.

[172] Chun-Hung Cheng, Ada Waichee Fu, and Yi Zhang. Entropy-based subspace clustering for mining numerical data. In *KDD '99: Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 84–93, New York, NY, USA, 1999. ACM Press.

[173] M. Zarinbal, M.H. Fazel Zarandi, and I.B. Turksen. Relative entropy fuzzy c-means clustering. *Information Sciences*, 260(0):74 – 97, 2014.

[174] Periklis Andritsos. *Scalable Clustering of Categorical Data and Applications*. PhD thesis, University of Toronto, 2004.

[175] Sally McClean, Bryan Scotney, Kieran Greer, and Ronan Páircéir. Conceptual Clustering of Heterogeneous Distributed Databases. In *Workshop on Distributed and Parallel Knowledge Discovery*, 2000.

[176] K. Chidananda Gowda and E. Diday. Symbolic clustering using a new dissimilarity measure. *Pattern Recogn.*, 24(6):567–578, 1991.

[177] J. B. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297, Berkeley, CA, 1967. University of California.

[178] Leonard Kaufman and Peter J. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis.* John Wiley & Sons, 1990.

[179] Sudipto Guha, Rajeev Rastogi, and Kyuseok Shim. CURE: an efficient clustering algorithm for large databases. In *SIGMOD '98: Proceedings of the 1998 ACM SIGMOD international conference on Management of data*, pages 73–84, New York, NY, USA, 1998. ACM Press.

[180] James C. Bezdek. *Fuzzy mathematics in pattern classification.* PhD thesis, Faculty of the Gradual School of Cornell University, Ithaca, NY, 1973.

[181] R. Krishnapuram and J.M. Keller. A possibilistic approach to clustering. *IEEE Transactions on Fuzzy Systems*, 1:98–110, 1993.

[182] Zhexue Huang and Michael K. Ng. A Fuzzy k-Modes Algorithm for Clustering Categorical Data. *IEEE Transactions on Fuzzy Systems*, 7(4):446–452, August 1999.

[183] Sudipto Guha, Rajeev Rastogi, and Kyuseok Shim. Rock: A robust clustering algorithm for categorical attributes. *Information Systems*, 25(5):345 – 366, 2000.

[184] Periklis Andritsos. Data Clustering Techniques. Qualifying oral examination paper, University of Toronto, March 2002.

[185] Venkatesh Ganti, Johannes Gehrke, and Raghu Ramakrishnan. CACTUS - clustering categorical data using summaries. In *KDD '99: Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 73–83, New York, NY, USA, 1999. ACM Press.

[186] Eshref Januzaj, Hans-Peter Kriegel, and Martin Pfeifle. Scalable density-based distributed clustering. In *PKDD '04: Proceedings of the 8th European Conference on Principles and Practice of Knowledge Discovery in Databases*, pages 231–244, New York, NY, USA, 2004. Springer-Verlag New York, Inc.

[187] A. Hinneburg and D. Keim. An efficient approach to clustering in large multimedia databases with noise. In *Proceedings of the 1998 International Conference on Knowledge Discovery and Data Mining*, pages 58–65, 1998.

[188] Wei Wang, Jiong Yang, and Richard R. Muntz. STING: A Statistical Information Grid Approach to Spatial Data Mining. In *Proceedings of the 23rd International Conference onVery Large Data Bases (VLDB)*, pages 186–195, Athens, Greece, 1997. Morgan Kaufmann Publishers.

[189] Rakesh Agrawal, Johannes Gehrke, Dimitrios Gunopulos, and Prabhakar Raghavan. Automatic Subspace Clustering of High Dimensional Data. *Data Min. Knowl. Discov.*, 11(1):5–33, 2005.

[190] Martin Ester, Hans-Peter Kriegel, and Jörg Sander. *Geographic Data Mining and Knowledge Discovery*, chapter Algorithms and Applications for Spatial Data Mining. Taylor & Francis, 2001.

[191] M. Indulska and M. E. Orlowska. Gravity based spatial clustering. In *GIS '02: Proceedings of the 10th ACM international symposium on Advances in geographic information systems*, pages 125–130, New York, NY, USA, 2002. ACM Press.

[192] Josenildo da Silva, Chris Giannella, Ruchita Bhargava, Hillol Kargupta, and Matthias Klusch. Distributed data mining and agents. *Engineering Applications of Artificial Intelligence*, 18:791–807, 2005.

[193] Nagiza F. Samatova, George Ostrouchov, Al Geist, and Anatoli V. Melechko. RACHET: An Efficient Cover-Based Merging of Clustering Hierarchies from Distributed Datasets. *Distrib. Parallel Databases*, 11(2):157–180, 2002.

[194] Hillol Kargupta, Weiyun Huang, Krishnamoorthy Sivakumar, and Erik Johnson. Distributed Clustering Using Collective Principal Component Analysis. *Knowledge and Information Systems*, 3:422–448, 2001.

[195] Charles Bouveyron and Camille Brunet-Saumard. Model-based clustering of high-dimensional data: A review. *Computational Statistics & Data Analysis*, 71:52 – 78, 2014.

[196] Paul S. Bradley, Usama Fayyad, and Cory Reina. Scaling Clustering Algorithms to Large Databases. In *Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 9–15, New York, NY, USA, 1998.

[197] E. A. Patrick. *Fundamentals of Pattern Recognition.* Prentice Hall, 1972.

[198] David Arthur and Sergei Vassilvitskii. k-means++: the advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, SODA '07, pages 1027–1035, Philadelphia, PA, USA, 2007. Society for Industrial and Applied Mathematics.

[199] J. C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms.* Plenum Press, 1981.

[200] N.R. Pal, K. Pal, J.M. Keller, and J.C. Bezdek. A Possibilistic Fuzzy c-Means Clustering Algorithm. *Fuzzy Systems, IEEE Transactions on*, 13(4):517–530, Aug 2005.

[201] Zexuan Ji, Yong Xia, Quansen Sun, and Guo Cao. Interval-valued possibilistic fuzzy C-means clustering algorithm. *Fuzzy Sets and Systems*, 253:138 – 156, 2014. Theme: Fuzzy Modeling and Clustering.

[202] J. Edward Jackson. *A User's Guide to Principal Components.* Wiley, 2003.

[203] Michael K. Ng and Joyce C. Wong. Clustering categorical data sets using tabu search techniques. *Pattern Recognition*, 35:2783–2790, 2002.

[204] Miin-Shen Yang, Pei-Yuan Hwang, and De-Hua Chen. Fuzzy clustering algorithms for mixed feature variables. *Fuzzy Sets and Systems*, 141(2):301–317, 2004.

[205] Robert H. Shumway. Time-frequency clustering and discriminant analysis. *Statistics & Probability Letters*, 63(3):307 – 314, 2003.

[206] J.W. Sammon. A Nonlinear Mapping for Data Structure Analysis. *Computers, IEEE Transactions on*, C-18(5):401–409, May 1969.

[207] D. L. Davies and D. W. Bouldin. Cluster Separation Measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1:95–104, 1979.

[208] X. L. Xie and G. Beni. A validity measure for fuzzy clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(8):841–847, 1991.

[209] Gianfranco Chicco. Overview and performance assessment of the clustering methods for electrical load pattern grouping. *Energy*, 42(1):68 – 80, 2012. 8th World Energy System Conference, {WESC} 2010.

[210] Maria Halkidi, Yannis Batistakis, and Michalis Vazirgiannis. On Clustering Validation Techniques. *J. Intell. Inf. Syst.*, 17(2-3):107–145, December 2001.

[211] T. Warren Liao. Clustering of time series data - a survey. *Pattern Recognition*, 38(11):1857 – 1874, 2005.

[212] Joan Serrà and Josep Ll. Arcos. An empirical evaluation of similarity measures for time series classification. *Knowledge-Based Systems*, 67(0):305 – 314, 2014.

[213] D.J. Berndt and J. Clifford. Using dynamic time warping to find patterns in time series. In *AAAI 1994 Workshop on Knowledge Discovery in Databases*, 1994.

[214] Paolo Capitani and Paolo Ciaccia. Warping the time on data streams. *Data & Knowledge Engineering*, 62(3):438 – 458, 2007. Including special issue: 20th Brazilian Symposium on Databases (SBBD 2005).

[215] Tomasz Górecki and Maciej Luczak. Non-isometric transforms in time series classification using {DTW}. *Knowledge-Based Systems*, 61(0):98 – 108, 2014.

[216] H. Sakoe and S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 26(1):43–49, Feb 1978.

[217] Zoltan Bankó and János Abonyi. Correlation based dynamic time warping of multivariate time series. *Expert Systems with Applications*, 39(17):12814 – 12823, 2012.

[218] François Petitjean and Pierre Gançarski. Summarizing a set of time series by averaging: From Steiner sequence to compact multiple alignment. *Theoretical Computer Science*, 414(1):76 – 91, 2012.

[219] Vit Niennattrakul, Dararat Srisai, and Chotirat Ann Ratanamahatana. Shape-based template matching for time series data. *Knowledge-Based Systems*, 26:1 – 8, 2012.

[220] Felix Hausdorff. *Grundzüge der Mengenlehre.* Veit and Company, Leipzig, 1914.

[221] Francesco Palumbo and Antonio Irpino. Multidimensional interval-data: metrics and factorial analysis. *Proceedings of the ASMDA*, pages 689–698, 2005.

[222] Marie Chavent. A Hausdorff Distance Between Hyper-Rectangles for Clustering Interval Data. In David Banks, FrederickR. McMorris, Phipps Arabie, and Wolfgang Gaul, editors, *Classification, Clustering, and Data Mining Applications*, Studies in Classification, Data Analysis, and Knowledge Organisation, pages 333–339. Springer Berlin Heidelberg, 2004.

[223] Nicolas Basalto, Roberto Bellotti, Francesco De Carlo, Paolo Facchi, Ester Pantaleo, and Saverio Pascazio. Hausdorff clustering. *Phys. Rev. E*, 78:046112, Oct 2008.

[224] Jinyang Chen, Rangding Wang, Liangxu Liu, and Jiatao Song. Clustering of trajectories based on Hausdorff distance. In *Electronics, Communications and Control (ICECC), 2011 International Conference on*, pages 1940–1944, Sept 2011.

[225] Francisco de A.T. de Carvalho, Renata M.C.R. de Souza, Marie Chavent, and Yves Lechevallier. Adaptive Hausdorff distances and dynamic clustering of symbolic interval data. *Pattern Recognition Letters*, 27(3):167 – 179, 2006.

[226] Francisco de A.T. de Carvalho. Fuzzy c-means clustering methods for symbolic interval data. *Pattern Recognition Letters*, 28(4):423 – 437, 2007.

[227] Chen-Chia Chuang, Jin-Tsong Jeng, and Sheng-Chieh Chang. Hausdorff distance measure based interval fuzzy possibilistic c-means clustering algorithm. *International Journal of Fuzzy Systems*, 15(4):471, 2013.

[228] Lei Chen, M. Tamer Özsu, and Vincent Oria. Robust and Fast Similarity Search for Moving Object Trajectories. In *Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data*, SIGMOD '05, pages 491–502, New York, NY, USA, 2005. ACM.

[229] P.-F. Marteau. Time Warp Edit Distance with Stiffness Adjustment for Time Series Matching. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(2):306–318, Feb 2009.

[230] Ignacio Benítez, José Luis Díez, and Pedro Albertos. Applying Dynamic Mining on Multi-Agent Systems. In *Proceedings of the 17th World Congress. The International Federation of Automatic Control (IFAC). Seoul, Korea, July 6-11, 2008*, 2008.

[231] Sangeeta Rani and Geeta Sikka. Recent Techniques of Clustering of Time Series Data: A Survey. *International Journal of Computer Applications*, 52(15):1–9, August 2012.

[232] Philippe Esling and Carlos Agon. Time-series Data Mining. *ACM Comput. Surv.*, 45(1):12:1–12:34, December 2012.

[233] Tak chung Fu. A review on time series data mining. *Engineering Applications of Artificial Intelligence*, 24(1):164 – 181, 2011.

[234] Katarina Kosmelj and Vladimir Batagelj. Cross-sectional approach for clustering time varying data. *Journal of Classification*, 7(1):99–109, 1990.

[235] Vladimir Batagelj. *Generalized Ward and Related Clustering Problems*, pages 67 – 74. 1988.

[236] T.W. Liao, B. Bolt, J. Forester, E. Hailman, C. Hansen, R.C. Kaste, and J. O' May. Understanding and projecting the battle state. In *23rd Army Science Conference*, 2002.

[237] Warissara Meesrikamolkul, Vit Niennattrakul, and ChotiratAnn Ratanamahatana. Shape-Based Clustering for Time Series Data. In Pang-Ning Tan, Sanjay Chawla, ChinKuan Ho, and James Bailey, editors, *Advances in Knowledge Discovery and Data Mining*, volume 7301 of *Lecture Notes in Computer Science*, pages 530–541. Springer Berlin Heidelberg, 2012.

[238] Hesam Izakian, Witold Pedrycz, and Iqbal Jamal. Fuzzy clustering of time series data using dynamic time warping distance. *Engineering Applications of Artificial Intelligence*, 39:235 – 244, 2015.

[239] R. Krishnapuram, A. Joshi, O. Nasraoui, and L. Yi. Low-complexity fuzzy relational clustering algorithms for web mining. *Fuzzy Systems, IEEE Transactions on*, 9(4):595–607, Aug 2001.

[240] H. Izakian, W. Pedrycz, and I. Jamal. Clustering Spatiotemporal Data: An Augmented Fuzzy C-Means. *Fuzzy Systems, IEEE Transactions on*, 21(5):855–868, Oct 2013.

[241] Hesam Izakian and Witold Pedrycz. Agreement-based fuzzy C-means for clustering data with blocks of features. *Neurocomputing*, 127:266 – 280, 2014. Selected papers from the 2012 Brazilian Symposium on Neural Networks (SBRN 2012).

[242] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Neural Networks, 1995. Proceedings., IEEE International Conference on*, volume 4, pages 1942–1948 vol.4, Nov 1995.

[243] Jian Yin, Duanning Zhou, and Qiong-Qiong Xie. A Clustering Algorithm for Time Series Data. In *Parallel and Distributed Computing, Applications and Technologies, 2006. PDCAT '06. Seventh International Conference on*, pages 119–122, Dec 2006.

[244] Jong P. Yoon, Jieun Lee, and Sung-Rim Kim. Trend similarity and prediction in time-series databases. volume 4057, pages 201–212, 2000.

[245] Y. Kakizawa, R.H. Shumway, and M. Taniguchi. Discrimination and Clustering for Multivariate Time Series. *Journal of the American Statistical Association*, 93:328–340, 1998.

[246] Eamonn Keogh, Stefano Lonardi, ChotiratAnn Ratanamahatana, Li Wei, Sang-Hee Lee, and John Handley. Compression-based data min-

ing of sequential data. *Data Mining and Knowledge Discovery*, 14(1):99–129, 2007.

[247] Carla S. Möller-Levet, Frank Klawonn, Kwang-Hyun Cho, and Olaf Wolkenhauer. Fuzzy Clustering of Short Time-Series and Unevenly Distributed Sampling Points. In Michael R. Berthold, Hans-Joachim Lenz, Elizabeth Bradley, Rudolf Kruse, and Christian Borgelt, editors, *Advances in Intelligent Data Analysis V*, volume 2810 of *Lecture Notes in Computer Science*, pages 330–340. Springer Berlin Heidelberg, 2003.

[248] Mahesh Kumar, Nitin R. Patel, and Jonathan Woo. Clustering Seasonality Patterns in the Presence of Errors. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '02, pages 557–563, New York, NY, USA, 2002. ACM.

[249] Xavier Golay, Spyros Kollias, Gautier Stoll, Dieter Meier, Anton Valavanis, and Peter Boesiger. A new correlation-based fuzzy logic clustering algorithm for fmri. *Magnetic Resonance in Medicine*, 40(2):249–260, 1998.

[250] Axel Wismüller, Oliver Lange, DominikR. Dersch, GerdaL. Leinsinger, Klaus Hahn, Benno PÃ¼tz, and Dorothee Auer. Cluster Analysis of Biomedical Image Time-Series. *International Journal of Computer Vision*, 46(2):103–128, 2002.

[251] T. Warren Liao. A clustering procedure for exploratory mining of vector time series. *Pattern Recognition*, 40(9):2550 – 2562, 2007.

[252] S. Policker and A.B. Geva. Nonstationary time series analysis by temporal clustering. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 30(2):339–343, Apr 2000.

[253] Min Ji, Fuding Xie, and Yu Ping. A Dynamic Fuzzy Cluster Algorithm for Time Series, 2013.

[254] J.J. van Wijk and E.R. van Selow. Cluster and calendar based visualization of time series data. In *Information Visualization, 1999. (Info Vis '99) Proceedings. 1999 IEEE Symposium on*, pages 4–9, 140, 1999.

[255] T. Yang. *Computational Verb Theory: From Engineering, Dynamic Systems to Physical Linguistics.* YangSky.com (Yang's). Yang's Scientific Research Institute, 2002.

[256] Mengfan Zhang and Tao Yang. Application of computational verb theory to analysis of stock market data. In *Anti-Counterfeiting Security and Identification in Communication (ASID), 2010 International Conference on*, pages 261–264, July 2010.

[257] Chonghui Guo, Hongfeng Jia, and Na Zhang. Time Series Clustering Based on ICA for Stock Data Analysis. In *Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08. 4th International Conference on*, pages 1–4, Oct 2008.

[258] Karl Pearson. LIII. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine Series 6*, 2(11):559–572, 1901.

[259] B.D. Fulcher and N.S. Jones. Highly comparative feature-based time-series classification. *Knowledge and Data Engineering, IEEE Transactions on*, PP(99):1–1, 2014.

[260] Antonio Irpino, Rosanna Verde, and Francisco de A.T. De Carvalho. Dynamic clustering of histogram data based on adaptive squared wasserstein distances. *Expert Systems with Applications*, 41(7):3351 – 3366, 2014.

[261] Ludger Rüschendorf. The Wasserstein distance and approximation theorems. *Zeitschrift fÃ¼r Wahrscheinlichkeitstheorie und Verwandte Gebiete*, 70(1):117–129, 1985.

[262] Pierpaolo D'Urso and Elizabeth Ann Maharaj. Autocorrelation-based fuzzy clustering of time series. *Fuzzy Sets and Systems*, 160(24):3565 – 3589, 2009. Theme: Non-Linear Systems and Fuzzy Clustering.

[263] J.G. Wilpon and L. Rabiner. A modified K-means clustering algorithm for use in isolated word recognition. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 33(3):587–594, Jun 1985.

[264] F. Itakura. Minimum prediction residual principle applied to speech recognition. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 23(1):67–72, Feb 1975.

[265] C. Goutte, L.K. Hansen, M.G. Liptrot, and E. Rostrup. Feature-space clustering for fMRI meta-analysis. *Human Brain Ma*, 13(1065-9471 (Linking)):165–183, 2001.

[266] C. Goutte, P. Toft, E. Rostrup, F. Nielsen, and L.K. Hansen. On clustering fMRI time series. *Neuroimage*, 9(1053-8119 (Linking)):298–310, 1999.

[267] T.C. Fu, F.L. Chung, V. Ng, and R. Luk. Pattern Discovery from Stock Time Series Using Self-Organizing Maps. In *KDD 2001 Workshop on Temporal Data Mining*, pages 27–37, 2001.

[268] L.M.D. Owsley, L.E. Atlas, and G.D. Bernard. Self-organizing feature maps and hidden Markov models for machine-tool monitoring. *Signal Processing, IEEE Transactions on*, 45(11):2787–2798, Nov 1997.

[269] M. Vlachos, J. Lin, E. Keogh, and D. Gunopulos. A wavelet based anytime algorithm for k-means clustering of time series. In *Proceedings of the Third SIAM International Conference on Data Mining*, 2003.

[270] Saeed Aghabozorgi, Mahmoud Reza Saybani, and Teh Ying Wah. Incremental clustering of time-series by fuzzy clustering. *Journal of Information Science and Engineering*, 28(4):671–688, 2012.

[271] Hailin Li and Chonghui Guo. Piecewise cloud approximation for time series mining. *Knowledge-Based Systems*, 24(4):492 – 500, 2011.

[272] Jessica Lin, Michai Vlachos, Eamonn Keogh, Dimitrios Gunopulos, Jianwei Liu, Shoujian Yu, and Jiajin Le. A MPAA-Based Iterative Clustering Algorithm Augmented by Nearest Neighbors Search for Time-Series Data Streams. In TuBao Ho, David Cheung, and Huan Liu, editors, *Advances in Knowledge Discovery and Data Mining*, volume 3518 of

Lecture Notes in Computer Science, pages 333–342. Springer Berlin Heidelberg, 2005.

[273] Wassily Hoeffding. Probability Inequalities for Sums of Bounded Random Variables. *Journal of the American Statistical Association*, 58(301):13–30, March 1963.

[274] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In *KDD'96*, pages 226–231, 1996.

[275] Cheng-Ping Lai, Pau-Choo Chung, and Vincent S. Tseng. A novel two-level clustering method for time series data analysis. *Expert Systems with Applications*, 37(9):6319 – 6326, 2010.

[276] E. Keogh, J. Lin, and W. Truppel. Clustering of time series subsequences is meaningless: implications for previous and future research. In *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on*, pages 115–122, Nov 2003.

[277] Amir Ben-Dor, Ron Shamir, and Zohar Yakhini. Clustering gene expression patterns. *Journal of computational biology*, 6(3-4):281–297, 1999.

[278] Ignacio Benítez, Carlos Blasco, Amparo Mocholí, and Alfredo Quijano. A Two-Step Process for Clustering Electric Vehicle Trajectories. In *Proceedings of the IEEE International Electric Vehicle Conference (IEVC 2014)*, 2014.

[279] Jessica Lin and Yuan Li. Finding Structural Similarity in Time Series Data Using Bag-of-Patterns Representation. In Marianne Winslett, editor, *Scientific and Statistical Database Management*, volume 5566 of *Lecture Notes in Computer Science*, pages 461–477. Springer Berlin Heidelberg, 2009.

[280] G. Salton, A. Wong, and C. S. Yang. A Vector Space Model for Automatic Indexing. *Commun. ACM*, 18(11):613–620, November 1975.

[281] Lei Li and B. Aditya Prakash. Time Series Clustering: Complex is Simpler! In Lise Getoor and Tobias Scheffer, editors, *ICML*, pages 185–192. Omnipress, 2011.

[282] Yimin Xiong and Dit-Yan Yeung. Time series clustering with ARMA mixtures. *Pattern Recognition*, 37(8):1675 – 1689, 2004.

[283] A. J. Bagnall and G. J. Janacek. Clustering Time Series from ARMA Models with Clipped Data. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pages 49–58, New York, NY, USA, 2004. ACM.

[284] R. Baragona. A simulation study on clustering time series with metaheuristic methods. *Quaderni di Statistica*, 3, 2001.

[285] E.A. Maharaj. Cluster of Time Series. *Journal of Classification*, 17(2):297–314, 2000.

[286] J. A. Vilar, A. M. Alonso, and J. M. Vilar. Non-linear Time Series Clustering Based on Non-parametric Forecast Densities. *Comput. Stat. Data Anal.*, 54(11):2850–2865, November 2010.

[287] Wei-Feng Zhang, Chao-Chun Liu, and Hong Yan. Clustering of Temporal Gene Expression Data by Regularized Spline Regression and an Energy Based Similarity Measure. *Pattern Recogn.*, 43(12):3969–3976, December 2010.

[288] Abdel-Ouahab Boudraa, Jean-Christophe Cexus, Mathieu Groussat, and Pierre Brunagel. An Energy-based Similarity Measure for Time Series. *EURASIP J. Adv. Signal Process*, 2008, January 2008.

[289] M. Ramoni, P. Sebastiani, and P. Cohen. Multivariate clustering by dynamics. In *Proceedings of the 2000 National Conference on Artificial Intelligence (AAAI 2000)*, pages 633–638, 2000.

[290] Alexios Savvides, Vasilis J. Promponas, and Konstantinos Fokianos. Clustering of biological time series by cepstral coefficients based distances. *Pattern Recognition*, 41(7):2398 – 2412, 2008.

[291] K. Kalpakis, D. Gada, and V. Puttagunta. Distance measures for effective clustering of arima time-series. In *Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on*, pages 273–280, 2001.

[292] Elizabeth Ann Maharaj and Pierpaolo D'Urso. Fuzzy clustering of time series in the frequency domain. *Information Sciences*, 181(7):1187 – 1211, 2011.

[293] Marcella Corduas. Clustering streamflow time series for regional classification. *Journal of Hydrology*, 407:73 – 80, 2011.

[294] Cen Li and Gautam Biswas. Temporal Pattern Generation Using Hidden Markov Model Based Unsupervised Classification. In *Proceedings of the Third International Symposium on Advances in Intelligent Data Analysis*, IDA '99, pages 245–256, London, UK, UK, 1999. Springer-Verlag.

[295] T. Oates, L. Firoiu, and P.R. Cohen. Clustering time series with hidden Markov models and dynamic time warping. In *Proceedings of the IJCAI-99 Workshop on Neural, Symbolic and Reinforcement Learning Methods for Sequence Learning*, 1999.

[296] Dat Tran and Michael Wagner. Fuzzy C-Means Clustering-Based Speaker Verification. In *Proceedings of the 2002 AFSS International Conference on Fuzzy Systems. Calcutta: Advances in Soft Computing*, AFSS '02, pages 318–324, London, UK, UK, 2002. Springer-Verlag.

[297] H. Lee Willis. *Electrical Transmission & Distribution Reference Book*, chapter 24: Characteristics of Distribution Loads, pages 784–808. ABB Power Company, 1997.

[298] G. Chicco, R. Napoli, and F. Piglione. Comparisons Among Clustering Techniques for Electricity Customer Classification. *IEEE Transactions on Power Systems*, 21(2):933, 2006.

[299] José Ramón Cancelo, Antoni Espasa, and Rosmarie Grafe. Forecasting the electricity load from one day to one week ahead for the spanish system operator. *International Journal of Forecasting*, 24(4):588–602, 2008.

[300] M.R. Khan and A. Abraham. Short Term Load Forecasting Models in Czech Republic Using Soft Computing Paradigms. *Archive preprint cs.AI/0405051*, 2004.

[301] S. Vemuri, Wen Liang Huang, and D.J. Nelson. On-Line Algorithms for Forecasting Hourly Loads of an Electric Utility. *Power Apparatus and Systems, IEEE Transactions on*, PAS-100(8):3775–3784, Aug 1981.

[302] M. Espinoza, C. Joye, R. Belmans, and B. De Moor. Short-Term Load Forecasting, Profile Identification, and Customer Segmentation: A Methodology Based on Periodic Time Series. *Power Systems, IEEE Transactions on*, 20(3):1622–1630, Aug 2005.

[303] S.V. Verdu, M.O. Garcia, C. Senabre, A.G. Marin, and F.J.G. Franco. Classification, Filtering, and Identification of Electrical Customer Load Patterns Through the Use of Self-Organizing Maps. *Power Systems, IEEE Transactions on*, 21(4):1672–1682, 2006.

[304] G.J. Tsekouras, N.D. Hatziargyriou, and E.N. Dialynas. Two-Stage Pattern Recognition of Load Curves for Classification of Electricity Customers. *Power Systems, IEEE Transactions on*, 22(3):1120–1128, 2007.

[305] B.D. Pitt and D.S. Kitschen. Application of data mining techniques to load profiling. In *Power Industry Computer Applications, 1999. PICA '99. Proceedings of the 21st 1999 IEEE International Conference*, pages 131–136, Jul 1999.

[306] C-S Chen, J. C Hwang, and C.W. Huang. Application of load survey systems to proper tariff design. *Power Systems, IEEE Transactions on*, 12(4):1746–1751, 1997.

[307] C.S Chen, M.S Kang, J.C Hwang, and C.W Huang. Synthesis of power system load profiles by class load study. *International Journal of Electrical Power & Energy Systems*, 22(5):325 – 330, 2000.

[308] Dan Apetrei, Ion Lungu, Florentin Batrinu, Gianfranco Chicco, Radu Porumb, and Petru Postolache. Load Pattern Classification and Profiling for a Large Supply Company. In *Proceedings of CIRED – 19th International Conference on Electricity Distribution*, 2007.

[309] C.S. Ozveren, C. Vechakanjana, and A.P. Birch. Fuzzy classification of electrical load demand profiles-a case study. In *Power System Management and Control, 2002. Fifth International Conference on (Conf. Publ. No. 488)*, pages 353–358, April 2002.

[310] Y.-H. Pao and D.J. Sobajic. Combined use of unsupervised and supervised learning for dynamic security assessment. In *Power Industry Computer Application Conference, 1991. Conference Proceedings*, pages 278–284, May 1991.

[311] G. Chicco, Roberto Napoli, Federico Piglione, P. Postolache, M. Scutariu, and C. Toader. Load pattern-based classification of electricity customers. *Power Systems, IEEE Transactions on*, 19(2):1232–1239, May 2004.

[312] B. Akperi and P. Matthews. Analysis of clustering techniques on load profiles for electrical distribution. In *2014 International Conference on Power System Technology (POWERCON)*, pages 1142–1149, Oct 2014.

[313] G.J. Tsekouras, P.B. Kotoulas, C.D. Tsirekis, E.N. Dialynas, and N.D. Hatziargyriou. A pattern recognition methodology for evaluation of load profiles and typical days of large electricity customers. *Electric Power Systems Research*, 78(9):1494 – 1510, 2008.

[314] Kai le Zhou, Shan lin Yang, and Chao Shen. A review of electric load classification in smart grid environment. *Renewable and Sustainable Energy Reviews*, 24:103 – 110, 2013.

[315] S. Haben, C. Singleton, and P. Grindrod. Analysis and Clustering of Residential Customers Energy Behavioral Demand Using Smart Meter Data. *Smart Grid, IEEE Transactions on*, PP(99):1–1, 2015.

[316] M. Chaouch. Clustering-Based Improvement of Nonparametric Functional Time Series Forecasting: Application to Intra-Day Household-Level Load Curves. *Smart Grid, IEEE Transactions on*, 5(1):411–419, Jan 2014.

[317] Joaquim L Viegas, Susana M Vieira, and João MC Sousa. Fuzzy clustering and prediction of electricity demand based on household characteristics. In *Proceedings of the 2015 Conference of the International Fuzzy Systems Association and the European Society for Fuzzy Logic and Technology (IFSA-EUSFLAT-15)*, 2015.

[318] Corinna Cortes and Vladimir Vapnik. Support-Vector Networks. *Machine Learning*, 20(3):273–297, 1995.

[319] Jungsuk Kwac, J. Flora, and R. Rajagopal. Household Energy Consumption Segmentation Using Hourly Data. *Smart Grid, IEEE Transactions on*, 5(1):420–430, Jan 2014.

[320] M Sforna. Data mining in a power company customer database. *Electric Power Systems Research*, 55(3):201 – 209, 2000.

[321] D. Gerbec, S. Gasperic, and F. Gubina. Determination and allocation of typical load profiles to the eligible consumers. In *Power Tech Conference Proceedings, 2003 IEEE Bologna*, volume 1, pages 5 pp. Vol.1–, 2003.

[322] R.F. Chang and C.N. Lu. Load profile assignment of low voltage customers for power retail market applications. *Generation, Transmission and Distribution, IEE Proceedings-*, 150(3):263–267, 2003.

[323] L. Varga and K. Czinege. Electricity consumer characterization in liberalized market based on data mining techniques. In *Universities Power Engineering Conference, 2007. UPEC 2007. 42nd International*, pages 78–82, Sept 2007.

[324] G. Chicco, Roberto Napoli, P. Postolache, M. Scutariu, and C. Toader. Customer characterization options for improving the tariff offer. *Power Systems, IEEE Transactions on*, 18(1):381–387, Feb 2003.

[325] Sergio Ramos, Joäo M. Duarte, F. Jorge Duarte, and Zita Vale. A data-mining-based methodology to support MV electricity customers characterization. *Energy and Buildings*, 91:16 – 25, 2015.

[326] Marta García. *Modelo de Carga para la Determinación de la Demanda Eléctrica en Baja Tensión y Centros de Transformación MT/BT*. PhD thesis, Departamento de Ingeniería Eléctrica. Universidad Politécnica de Valencia, 2001. In Spanish.

[327] Alfredo Quijano. *Modelos de Carga en Sistemas Eléctricos de Distribución*. PhD thesis, Departamento de Ingeniería Eléctrica. Universidad politécnica de Valencia, 1992. In Spanish.

[328] L. Hernandez, C. Baladron, J.M. Aguiar, B. Carro, A.J. Sanchez-Esguevillas, J. Lloret, and J. Massana. A Survey on Electric Power Demand Forecasting: Future Trends in Smart Grids, Microgrids and Smart Buildings. *Communications Surveys Tutorials, IEEE*, 16(3):1460–1495, Third 2014.

[329] M.S. Abou-Hussien, M.S. Kandlil, M.A. Tantawy, and S.A. Farghal. An Accurate Model for Short-Term Load Forecasting. *Power Apparatus and Systems, IEEE Transactions on*, PAS-100(9):4158–4165, Sept 1981.

[330] H.L. Willis and C.L. Brooks. An Interactive End-Use Electric Load Model for Microcomputer Implementation. *Power Apparatus and Systems, IEEE Transactions on*, PAS-102(11):3693–3700, Nov 1983.

[331] C.F. Walker and J.L. Pokoski. Residential Load Shape Modelling Based on Customer Behavior. *Power Apparatus and Systems, IEEE Transactions on*, PAS-104(7):1703–1711, July 1985.

[332] Q.-C. Lu, W.M. Grady, M.M. Crawford, and G.M. Anderson. An adaptive nonlinear predictor with orthogonal escalator structure for short-term load forecasting. *Power Systems, IEEE Transactions on*, 4(1):158–164, Feb 1989.

[333] S. Rahman and G. Shrestha. A priority vector based technique for load forecasting. *Power Systems, IEEE Transactions on*, 6(4):1459–1465, Nov 1991.

[334] M.Y. Cho, J.C. Hwang, and C-S Chen. Customer short term load forecasting by using ARIMA transfer function model. In *Energy Management and Power Delivery, 1995. Proceedings of EMPD '95., 1995 International Conference on*, volume 1, pages 317–322 vol.1, Nov 1995.

[335] S.Sp. Pappas, L. Ekonomou, D.Ch. Karamousantas, G.E. Chatzarakis, S.K. Katsikas, and P. Liatsis. Electricity demand loads modeling using AutoRegressive Moving Average (ARMA) models. *Energy*, 33(9):1353 – 1360, 2008.

[336] O. Mohammed, D. Park, R. Merchant, T. Dinh, C. Tong, A. Azeem, J. Farah, and C. Drake. Practical experiences with an adaptive neural network short-term load forecasting system. *Power Systems, IEEE Transactions on*, 10(1):254–265, Feb 1995.

[337] M. Beccali, M. Cellura, V. Lo Brano, and A. Marvuglia. Forecasting daily urban electric load profiles using artificial neural networks. *Energy Conversion and Management*, 45:2879 – 2900, 2004.

[338] E. Banda and K.A. Folly. Short Term Load Forecasting Using Artificial Neural Network. In *Power Tech, 2007 IEEE Lausanne*, pages 108–112, July 2007.

[339] A. Lendasse, M. Cottrell, V. Wertz, and M. Verleysen. Prediction of electric load using Kohonen maps - Application to the Polish electricity consumption. In *American Control Conference, 2002. Proceedings of the 2002*, volume 5, 2002.

[340] O.A.S. Carpinteiro and A.J.R. Reis. A SOM-based hierarchical model to short-term load forecasting. In *Power Tech, 2005 IEEE Russia*, pages 1–6, June 2005.

[341] S. Rahman and R. Bhatnagar. An expert system based algorithm for short term load forecast. *Power Systems, IEEE Transactions on*, 3(2):392–399, May 1988.

[342] Han-Ching Kuo and Yuan-Yih Hsu. Distribution system load estimation and service restoration using a fuzzy set approach. *Power Delivery, IEEE Transactions on*, 8(4):1950–1957, Oct 1993.

[343] A. Seppala. Statistical distribution of customer load profiles. In *Energy Management and Power Delivery, 1995. Proceedings of EMPD '95., 1995 International Conference on*, volume 2, pages 696–701 vol.2, Nov 1995.

[344] B. Stephen, A.J. Mutanen, S. Galloway, G. Burt, and P. Jarventausta. Enhanced Load Profiling for Residential Network Customers. *Power Delivery, IEEE Transactions on*, 29(1):88–96, Feb 2014.

[345] D.B. Belzer and M.A. Kellogg. Incorporating sources of uncertainty in forecasting peak power loads - a Monte Carlo analysis using the extreme value distribution. *Power Systems, IEEE Transactions on*, 8(2):730–737, May 1993.

[346] H. Mori and H. Kobayashi. Optimal fuzzy inference for short-term load forecasting. *Power Systems, IEEE Transactions on*, 11(1):390–396, Feb 1996.

[347] Kwang-Ho Kim, Jong-Keun Park, Kab-Ju Hwang, and Sung-Hak Kim. Implementation of hybrid short-term load forecasting system using artificial neural networks and fuzzy expert systems. *Power Systems, IEEE Transactions on*, 10(3):1534–1539, Aug 1995.

[348] D. Srinivasan, C.S. Chang, and A.C. Liew. Demand forecasting using fuzzy neural computation, with special emphasis on weekend and public holiday forecasting. *Power Systems, IEEE Transactions on*, 10(4):1897–1903, Nov 1995.

[349] M. Ramze Rezaee, B.P.F. Lelieveldt, and J.H.C. Reiber. A new cluster validity index for the fuzzy c-mean. *Pattern Recognition Letters*, 19(3 – 4):237 – 246, 1998.

[350] Y. Fukuyama and M. Sugeno. A new method of choosing the number of clusters for fuzzy c-means method. In *Proceedings of the 5th Fuzzy Systems Symposium*, 1989.

[351] Sharon L. Lohr. *Sampling: Design and Analysis.* 2000.

[352] Alcock R.J. and Manolopoulos Y. Time-Series Similarity Queries Employing a Feature-Based Approach. In *7th Hellenic Conference on Informatics*, 1999.

[353] A. Vidaurre, M. H. Giménez, and J. Riera. *Fundamentos Físicos de la Ingeniería II.* Universitat Politècnica de València, 1996. In Spanish.

[354] James A. Svoboda and Richard C. Dorf. *Introduction to Electric Circuits.* Wiley, 2013.

[355] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.

[356] J.C. Bezdek. A Convergence Theorem for the Fuzzy ISODATA Clustering Algorithms. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-2(1):1–8, Jan 1980.

[357] Weina Wang and Yunjie Zhang. On fuzzy cluster validity indices. *Fuzzy Sets and Systems*, 158(19):2095 – 2117, 2007. ¡ce:title¿Theme: Data Analysis¡/ce:title¿.

[358] J. C. Dunn. Well separated clusters and optimal fuzzy-partitions. *Journal of Cybernetics*, 4:95–104, 1974.

[359] N. R. Pal and J. Biswas. Cluster validation using graph theoretic concepts. *Pattern Recognition*, 30(6):847–857, 1997.

[360] Inmaculada Torres Castro. Sampling concepts. Lecture Notes. In Spanish.