A Survey on Pre-processing Techniques: relevant issues in the Context of Environmental Data Mining

Karina Gibert
Knowledge Engineering and Machine Learning Group. Department of Statistics and Operation Research, Universitat Politècnica de Catalunya-BarcelonaTech, Barcelona, Catalonia, Spain. e-mail: karina.gibert@upc.edu

Miquel Sànchez–Marrè
Knowledge Engineering and Machine Learning Group. Computer Science Department, Universitat Politècnica de Catalunya-BarcelonaTech, Barcelona, Catalonia, Spain

Joaquín Izquierdo
Fluing-IMM Universitat Politècnica de València, Valencia, Spain

One of the important issues related with all types of data analysis, either statistical data analysis, machine learning, data mining, data science or whatever form of data-driven modeling, is data quality. The more complex the reality to be analyzed is, the higher the risk of getting low quality data. Unfortunately real data often contain noise, uncertainty, errors, redundancies or even irrelevant information. Useless models will be obtained when built over incorrect or incomplete data. As a consequence, the quality of decisions made over these models, also depends on data quality. This is why pre-processing is one of the most critical steps of data analysis in any of its forms. However, pre-processing has not been properly systematized yet, and little research is focused on this. In this paper a survey on most popular pre-processing steps required in environmental data analysis is presented, together with a proposal to systematize it. Rather than providing technical details on specific pre-processing techniques, the paper focus on providing general ideas to a non-expert user, who, after reading them, can decide which one is the more suitable technique required to solve his/her problem.

Keywords: Pre-processing, Data Quality, Data Mining, Knowledge Discovery from Databases, Multidisciplinary approach, Environmental Systems

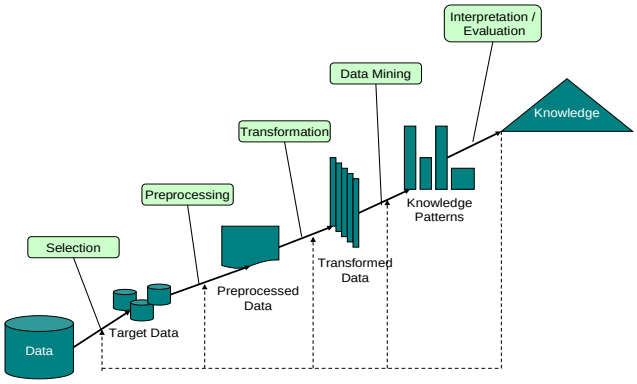
AI Communications
ISSN 0921-7126, IOS Press.

1. Introduction

Environmental systems (ESs) typically contain many interrelated components and processes, which may be biological, physical, geological, climatic, chemical, or social. Whenever we attempt to analyze ESs and related problems, we are immediately confronted with complexity stemming from various sources. Thus, there is a great need for data analysis, modeling of ESs and development of decision support systems in order to improve the understanding of ESs behavior and the management of these complex systems (specially under abnormal situations). As stated in [90], the special features of environmental processes demand a careful approach to improve the analysis to be better known, modeled and consequently better managed or controlled.

Knowledge Discovery of Data (KDD) [45] [228] appeared in 1989 [67] referring to high level applications including particular methods of *Data Mining* (DM, see Fig. 1), oriented to extract useful and understandable knowledge from complex data. Thus, KDD is a specifically appealing umbrella to analyze environmental data. Providing highly useful information from data bases is per se very important, although KDD commonly is a preparatory activity for an environmental software system development. Also, the KDD approach facilitates the integration of different knowledge sources and fields of expertise and the involvement of end-user (domain expert) criteria and stakeholders' points of view in algorithm design and result interpretation, which bridges the gap between data and real effective decision-making levels. Thus, in the last years an increasing interest to apply DM to environmental data has been observed.

Fayyad's proposal marked the beginning of a new paradigm in KDD research, considering prior and posterior analysis as much important as the application of DM techniques themselves [67]:



- Understanding goals
- Data Selection
- Data cleaning
- Data reduction and projection (transformation)
- Choosing the DM task
- Selecting the DM algorithm/s and parameters
- Data mining
- Interpreting mined patterns (post-processing)
- Reporting and using discovered knowledge

Fig. 1. Outline of the Knowledge Discovery from Data process as originally defined by Fayyad (Fayyad 1996)

Most previous work on KDD has focused on [...] DM step. However, the other steps are of considerable importance for the successful application of KDD in practice

In fact, both prior and posterior analyses require great effort when dealing with real applications [83] [84] [85]. Prior analysis is critical, mainly owing to two reasons:

- Real data sets tend to be imperfect, contain errors, outliers, missing data, extra noise. Tools either for detecting or correcting them are required.
- Application of a certain data mining technique may require specific conditions for the data set (only binary variables, centered data, normality, only qualitative variables, etc.). In this case, tools for verifying that those conditions hold, or eventually transform data in the appropriate way to meet these conditions, are required.

In environmental data, measurement errors (from automatic or manual monitoring), uncertainty, imprecision, multi-scalarity, non-linearities, non-stacionarity, non-normality, heterogeneity, etc., are frequent. Also, redundant variables, irrelevant, or even contradictory, are found. Systematic data pre-processing including objective exploration, visualization and data transformation is particularly critical for:

- Better understanding the data set
- Detecting imperfections in the data and managing them in the proper way to guarantee a correct analysis
- Correctly preparing data for the selected DM technique/s, if the required assumptions do not hold
- The correctness of the DM itself, which critically depends on the quality of the data and wrong or poor pre-processing may lead to incorrect results.

Also, particular efforts in post-processing the results provided by a DM technique are important in this context, in order to make these results directly understandable by an environmental scientist, who has to make real decisions upon them, ensuring a real impact to the environmental system behavior [34] [85] [47] [81] [80]. In fact, it can be said that the quality of the decisions will depend, not only on the quality of the data mining results themselves, but also on the capacity of the system to communicate the relevant results to the decision-maker as understandably as possible, and, in a first level, on the data quality itself (Fig. 2) [187].

Indeed, *data cleaning* [163], also known as *data preparation* [187] or *pre-processing* [63], are often time consuming and difficult; mainly because the approaches taken should be tailored to each specific application, and human interaction is required. In fact, once pre-processing is finished and data is ready for the analysis, the application of DM algorithms becomes quick (requiring some parameter settings in the appropriate software), and can be often automated. In fact, a small proportion of the time devoted to the whole KDD process is spent in the DM step. In real applications, the time devoted to pre-processing is rarely below 70% of the time-life for the whole KDD process [126] [192]. Anyway, a serious pre-processing is essential for obtaining good quality results, and useful new knowledge from it. Data miners should become conscious of the importance of performing very careful and rigorous pre-processing, and allocate sufficient time to this activity accordingly.

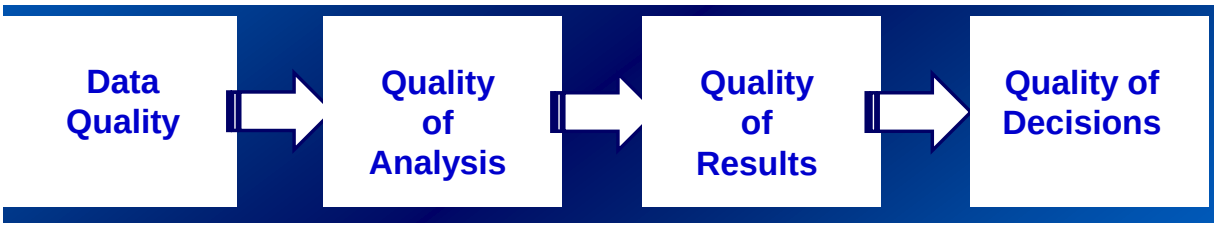


Fig. 2. Propagation of lacks of quality: From data to decisions

Most part of the research in Data Mining has been basically focused on providing better data mining algorithms, but few works specifically focus on both pre and post-processing, despite both are recognized as critical steps for successful KDD processes by the scientific community. In this work we focus on the very first step of the process, related to pre-processing in environmental applications. A clear methodology for tackling pre-processing within the KDD process has not been well-established yet. This paper does not intend to be exhaustive, but to provide environmental scientists with tools for addressing the most common problems arisen in pre-processing as well as to provide a specific methodological proposal. Authors expect this paper can contribute to make pre-processing fully available to both data miners and environmental scientists and to help them to better pre-process their data, getting better KDD results, and as a consequence, better knowledge from their environmental systems.

This paper tries to provide a reference material of what to do in practice, although work is still required to provide systematic decision criteria in pre-processing.

1.1. Structure of the paper

The structure of the paper is the following: Section 2 identifies the main pre-processing issues. Section 3 provides a brief introduction on getting the original data. Section 4 discusses about how domain knowledge is used to filter objects and select a first set of variables to build the first working data matrix. Section 5 gives a basic introduction on visualization tools useful for pre-processing. Section 6 deals with outliers. Section 7 is on error detection. Section 8 on missing data. Section 9 on relevance and redundancy detection and dimensionality reduction. Section 10 introduces most used transformations in pre-processing for the data mining purpose. Section 11 discusses on some interesting scenarios where creation of new variables is useful for data mining. Section 12 contains the conclusions and future lines.

2. The main pre-processing issues

As previously stated, pre-processing or data cleaning is a fundamental aspect, too often neglected. Mining data with imperfections can have dramatic consequences on the results of the analysis. Detecting and properly treating them is a very complex task which requires a lot of expertise and is highly time-consuming.

Classically, Statistics provided a wide set of possibilities to pre-process data. The global process of transforming a raw dataset to a correct one ready for analysis is called *data pre-processing* or *data cleaning* or *data preparation* and makes an intensive use of basic descriptive statistics and basic univariate and bivariate graphical representations of data. Sometimes some multivariate descriptive techniques (Principal Component Analysis) are also useful to detect high-order outliers or some non-linearities. Also, inductive techniques, coming from Artificial Intelligence, are useful in some tasks, like determining variables' relevance. Visualization techniques, more or less sophisticated, play a crucial role in the detection of many important issues for pre-processing.

No well-established pre-processing methodologies have been formalized yet, but some clear guidelines can be provided. Pre-processing ranges from the simplest descriptive techniques to the more sophisticated data analysis methods, depending on the nature of data and the goals of the analysis itself. The authors belief that most of the operations performed in a pre-processing step can be reduced to a preliminary step for entering the data and four main families of techniques, which are required, or not, depending on the original data formats received for the analysis:

- **Introducing data in the pre-processing tool**: oriented to read the data from the software as well as metadata supported by the software to create the correct context for a proper data pre-processing
- **Visualization techniques**: oriented to show characteristics of data that will require attention before the analysis

– **Detection techniques**: Those oriented to detect imperfections in datasets or to verify the accomplishment of required assumptions for a particular analysis. These are often associated with some kind of diagnoses on data, and consequently, to some decisions related to pre-processing:
  * Outlier detection
  * Data errors detection
  * Missing data detection
  * Relevance or redundancy detection, feature weighting
  * Independence assessment
  * Detection of influent observations
  * Normality assessment
  * Linearity assessment
– **Transforming techniques**: Those oriented to perform transformations in the dataset in order to correct the imperfections detected before, or to achieve the technical conditions to apply a certain analysis technique.
  * Determining the active set of data matrix rows and columns (expert-based objects and variables selection, including filtering)
  * Outlier treatment
  * Error data treatment
  * Missing data treatment
  * Treatment of relevance and redundancy and dimensionality reduction techniques
    * Instance Selection and resampling
    * Feature Selection
    * Factorial methods
  * Transforming variables
    * Homogenization
    * Differences and ratios
    * Compositional data
    * Functional transforms (inverse, logarithmic, quadratic, Box-Cox, etc., transformations)
    * Recodification
    * Discretization
    * Centering, standardization and normalization
    * Fuzzyfication
    * Imbalanced datasets
  * Creation of new variables
    * Aggregation
    * Feature Extraction
    * Building indicators
    * Multivalued variables

When data is geospatial, additional smoothing or noise reduction operations can be required. This issue is out of the scope of this paper and will not be specifically treated (see [69] for an overview). In some cases, images are pre-processed to extract relevant features that can be placed in standard data matrices, eventually georeferenced, for further joint mining ([70]).

Based on our experience in mining real data sets from more than 20 years, we would say that the main flow of this process can be synthesized in Fig. 3.

In the following sections we provide information for all the topics stated above. For some of them detection and treatment are strongly related and both aspects are treated in the same section, as is the case of outliers treatment and detection.

3. Building the original data matrix

As said before, many different sources of information can be involved in the observation of an ES. In the recent years, data coming from smart sensors or images are quite usual. Today, the web is still not a common information source for ES analysis, but it might come in the near future as a relevant provider of both structured and unstructured data.

This means that when an ES is analyzed, different data sources might be combined and some hard preprocessing might be required even before getting the very first original data matrix suitable for analysis.

Thus, when information sources are images, and a data mining approach is to be adopted, some feature extraction methods [169] need to be applied to transform the image into a vector of indicators identifying the image itself, which can be properly represented in a classical two-dimensional data matrix. For example, in a land use application, satellite images are often used, and they might be synthesized in a vector of indicators like *existence of river or not in the image, surface of the urban area, surface of forest, surface of crop area*, etc. Feature extraction processes often are domain-dependent and require enough contextual knowledge to determine which are the relevant features to be extracted from the image, according also to the goals of the problem.

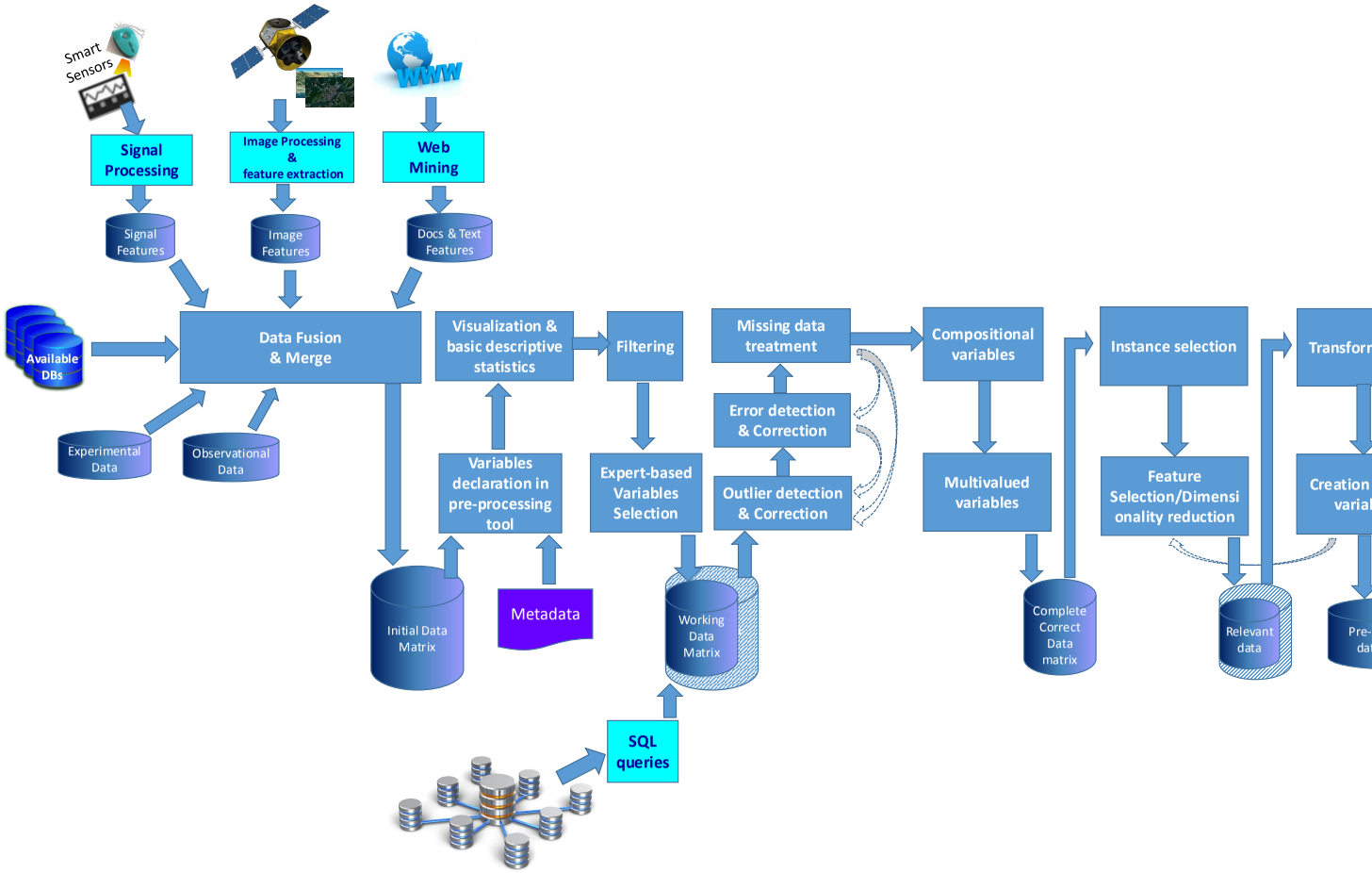


Fig. 3. Fig3 A proposal for pre-processing process

Also, data might come from smart sensors, which are on-line, monitoring several parameters of an ES, like a Waste Water Treatment Plant (WWTP) for example. In that case, data corresponds to a certain instant time, and normally each observation has a time stamp, expressed in one of the variables of the data matrix. For instance, the data matrix can include biochemical parameters of the WWTP along time (like concentration of organic matter, suspended solids, etc.), providing, for example, daily means. In most applications, each parameter (or variable) will represent a column of the data matrix and the timestamp will be an additional variable. Sometimes, some pre-processing is required to integrate the information of several sensors in a single indicator (like mean water level of a bioreactor for example, measured by several sensors in different points) and the imprecision or uncertainty associated to the measurements must be properly treated at pre-processing level, in particular the noise associated to the signals [168]. Regarding imprecision, georadar, or GPS systems, for example, provide a region where the target can be located with a certain probability, but are not able to provide exact positions of the target instances. SCADA provides the minimum and maximum values of a signal in a certain interval. Strictly, despite being able to measure a magnitude, its exact value will be never known; it is only known that the measurement is somewhere in a certain range, bounded by the precision of the measurement instrument or measurement procedure itself [221]. In applications with these kinds of inputs, it makes sense to include the uncertainty associated to the measure itself in the representation of the values. Fuzzy sets [233] or rough sets [177] are among the most commonly used representation models for these situations, but require specific data mining methods accepting this kind of fuzzy or rough variables. In the first case, linguistic labels are used to represent the uncertainty associated to the values of the variables for every observation [217]. Fuzzy datasets can directly come from a monitoring system, like radar data, which intrinsically provides areas of localization. Also, original crisp data can be artificially fuzzified for specific analysis, taking into account the precision of the measurement instruments and using specific fuzzification algorithms (see [20] [124] [173] [137] [66] [117]. In this paper, we will provide preprocessing methods oriented to crisp representation of data, which is the most used for simplicity in current real applications. However, the reader has to keep in mind the possibility of dealing with these uncertainties naturally associated with raw data, by using specific uncertainty formalisms which are less known from a practical point of view.

Even though the web data is not very popular yet in the current environmental applications, the web provides documents and pages that in some near future might become relevant for the environmental data mining. In that case, text mining methods need to be used to extract relevant characteristics of these documents that might also be represented in a classical two-dimensional data matrix, and can be used as ordinary inputs of classical data mining methods. There is a good chance to discover new findings and conclusions using modern methods of text mining supported by ontologies and semantic patterns. A survey of these methods is given in [49]. Text mining technology plays an important role in detecting epidemics (*epidemic intelligence*) based on event alerting from unstructured web pages and social networks [46]. A sample of the interest of environmental authorities in the monitoring of social networks for environmental purposes, or emergency management is the challenge on prediction of timing and intensity of seasonal climate events like hurricanes or earthquakes, among others [72][164]. Also, sentiment analysis is currently providing a significant added value by analyzing publications either in social networks or web pages [176] [222]. Some public resources are available to process this kind of textual information and obtain new variables, like the intensity of the earthquake [14].

Thus, building the original data matrix means to make *data fusion* [50] [63] with all information coming from the different sources, and eventually merge several independent data matrices into a single one containing all available information for each data row [192]. Finally, one of the current problems in environmental data representation is the lack of a standard environmental data format, despite some proposals and efforts of standardization made. Environmental data sharing is classically based on sharing data format, measurement methods, analysis methods and instrumentations. On the other hand, the standardization of the environmental data aims at sharing the data structure in order to share information among various bodies. Since environmental data possess coordinates on the globe, it can be treated as a form of *geographical information*. Then data structuring definition work is done by the ISO/TC211 [115] [114]. If this is applied, then environmental data may be standardized in the same way as other geographical information. The ISO/TC211 considers that geographical information comes from geographical feature data and meta-data [113].

One of the most promising directions has been the use of XML files to encode environmental data. XML is becoming an increasingly common format for data representation in data mining domains due to its expressiveness, flexibility, and cross-platform nature. An example is Extended Environments Markup Language (EEML), which is a protocol for sharing sensor data between remote responsive environments, both physical and virtual, initially aimed for construction purposes [61].

As an example, in the analysis of river's retention [153], one could get different databases. One containing morphologic characteristics of the river, built by experts; another containing biochemical characteristics of the rivers at different timestamps, provided by a biochemical laboratory; another containing information on microbiological diversity, provided by a set of biologists, and based on microscope observation of samples; another with some satellite pictures to learn land uses around the river; and another with some sensor data about flows along time. The very first step of preprocessing is to properly transform all this information into a single data matrix where rivers are in the rows and all flow, land use, morphologic, biochemical and microbiological, characteristics are expressed as columns in the data matrix with values at the different timestamps (one row per timestamp).

This single data matrix is the one used as *original data matrix* to pass to the next step of pre-processing.

3.1. Structure of original data matrix

In the classical data mining applications, the original data matrix is a two-dimensional table containing n rows representing the set I of n observations (also named individuals or instances) and K columns, which are the variables used to describe the observations. In cell $(i,k), i = 1:n, k = 1:K$, the value of variable X_k observed for object i is registered, x_{ik}. Usually, these data matrices can be represented in plain text files or CSV formats. Variables are in columns, and can be numerical (like pressures, concentrations of pollutants, depths, etc.), ordinal (level of risk of fire in a forest), binary (presence of certain microorganism in water, dangerous levels of a certain pollutant in air, etc.) or qualitative with more than two modalities (color of algae, type or eruption in a volcano, etc.).

Along this paper, the terminology used to refer to both the *variables* and *observations* of the usual two-dimensional data matrix changes according to the most usual term used in the concrete field of study providing the different pre-processing techniques (i.e., statistics, machine learning, etc.). Therefore, in the text, we refer to a *variable* with the following words: *feature, attribute, property* or *data column*. Likewise, we refer to an observation with the following words: *instance, example, object* or *data row*.

Eventually, when observations are taken along the time, one of the variables is the timestamp, which can be expressed in any date format, and requires attention to ensure it is properly interpreted by the software.

Also, when observations are linked to a specific location, some variables giving geographical information may appear, like Cs radioactivity measures in points of a surrounding area of a nuclear plant. They could refer to 2D or 3D spatial information. In 2D situations, each geographical position is determined by a pair of values, like latitude and longitude values, a pair of UTM coordinates, etc. [30]. In the case of 3D, in addition to the pair of coordinates, an additional elevation or deep value is provided. This data is relevant for territorial graphical models where geographical areas are identified with the same coordinates. Then, data can be visualized over digital maps, by using this information as pointers to particular map areas, to visualize or induce some spatial predictive model.

In this paper two special cases of subsets of related variables that require special pre-processing are also considered: multivalued qualitative variables and compositional data, both common in ESs.

Multivalued qualitative variables are qualitative variables that can have several values simultaneously. This is the case, for example, of the *water access technologies* available in cities. Imagine a data matrix containing one city per row. The qualitative variable water access technology can take a set of possible values (for example, gravity water service, pump water services, well water services and call water services) [205] [121]. As the same city can have different districts with different access technologies, this variable is taking a list of values (technologies) in each city. Formally, it corresponds to the case where the corresponding cell of the data matrix is a vector of modalities. As this would not be manageable in a classical data mining software, it is usual to represent each of the modalities as a binary variable and report a Boolean for each city, by indicating whereas the city has households with gravity water services or not, pump water services or not and so on. Thus, for a multivalued variable with s modalities, a set of s binary variables will be used, which is different from dummy variables, as here, more than one can be true simultaneously.

Finally, *compositional data* are sets of s strictly positive real variables with constant sum $Z > 0$ [178]. Z is generally 100 (percentage) or 1 (proportion). This implies the existence of a linear relationship among the s variables, with implications in the results of the data mining process, particularly when methods assuming independence of variables are used. This is the case, for example, of geochemical composition of rocks, composition of sediments in rivers, sources of pollution in a lake, land use composition, waste/wastewater composition, among others [5].

3.2. Introducing the data into the pre-processing tool

It is always important to verify that the data matrix read by the software really contain the expected rows and columns, with proper contents. Here, format issues can be responsible of reading numeric columns as alphanumeric, or to merge several columns in one and so on. It is essential to detect these issues at the very first step of the data pre-processing. As an example, the format of a csv file generated by Excel if it has Spanish language activated, uses "," as decimal indicator and ";" as column separator instead of the standard "." and ",". This means that importing a csv file generated by a Spanish version of Excel in pre-processing software expecting an *English csv file* will not work.

Data should always incorporate extra information describing the variables, like source of information, type of the variable, unit of measurement, degree of uncertainty, etc. This is the *meta-data* [155]. It is crucial for correct interpretation of results and also for proper data pre-processing and right data mining technique choice. For example, a dataset containing a column entitled Age with values 10, 20, 30, 40... can be wrongly interpreted as a numerical variable, but it can also be the encoding for something like *Children, Young, Adult*... In this case, the values of the variable are not numbers, in a mathematical sense, but simple codes of a *qualitative* variable that do not admit, for instance, standard deviation computation. Moreover, this can be the current age of the individual, but also the age at which he was diagnosed for a certain illness. Only the proper additional meta-data makes it possible to be sure that data is properly understood, and that a proper treatment will be provided.

Unfortunately, most software tools do not support meta-data, or cannot process it, and it is too frequent to get datasets with incomplete meta-data, that can propagate wrong assumptions (like assuming the same measurement units in the whole sample for a pollutant concentration, where different laboratories have been involved and results are expressed in different concentration units, according to each laboratory). Meta-data must be collected and properly understood for a proper data analysis and interpretation. Also, it must be transferred to the pre-processing software tool at the highest possible level. For example, nowadays, most of the software tools admit some declaration of qualitative variables that prevent for certain wrong manipulations, as wrong interpretation of numerical codes, wrong computation of standard deviations, or even wrong introduction into classical linear regression models by mistake. Of course, this is software-depending and it is the responsibility of the analyst to manage each variable in the proper way, with or without the support of the software.

4. Determining the working data matrix

Once all available information is expressed in the *original data matrix*, the data really used for the analysis is going to be determined. We name this step *building the working data matrix*, and basically consists in answering two main questions:

- Which data matrix rows (instances) are going to be kept for the analysis
- Which data matrix columns (variables) are going to be considered in the analysis

4.1. Determining the data matrix rows

This is basically related with defining the target population. Consider a dataset with information of a WWTP for the whole year. Whereas we are interested in modeling the plant in the summer period, it is useless to consider the rows corresponding to winter days. *Filtering* is mainly devoted to selection of subsamples from the main data matrix, with several purposes:

- Restricting the scope of the analysis to a local subdomain: only summer measurements, as said before; also, given 5 years of air quality measurements, reduce the analysis to those days where emission of pollutants are higher than some legal threshold to identify emission sources.
- Eliminating observations coming from other domains not targeted in the analysis. As an example, eliminate pieces of land in the mountain, wrongly included in a research regarding agricultural land-uses in a valley.

Thus, filters will be required to select the rows relevant for the analysis. Formally, filters are Boolean expressions defined over individuals that will evaluate to TRUE when the row must be kept or FALSE when must be excluded for the analysis.

In medicine, defining the target population for a clinical trial is a very sensitive task and the experts commonly talk about *defining inclusion/exclusion criteria* that determine the target population. *Inclusion/exclusion criteria* is a term non-commonly used in environmental systems, whereas it can be perfectly introduced into the picture, as this is exactly the mission of filters: to clearly express the conditions that must be held by the rows kept for the analysis, and those held by the rows excluded for the analysis.

When different filters are managed together, the main caution required is to be sure, at every step of the analysis, that the results only apply to the exact subpopulation targeted. Making an extensive use of intermittent filters can produce confusions and cross-results, which are extremely difficult to discover. If many local analyses are required to complete the entire analysis, sometimes it is better to run the filters once at the beginning of the study and physically produce the required subsamples in separate databases. Then, local analyses can be performed directly on the correct data matrix thus decreasing the risk of errors. If this methodology is adopted, subsamples must be rebuilt every time a change happens in the master data matrix.

The target population defined for the analysis is framing the conclusions of the mined data and it is absolutely relevant to report it explicitly in order to make clear the scope of the conclusions and to which extent they can be generalized

4.2. Determining the data matrix columns

The second issue is to determine the subset of columns of the data matrix to be considered for the analysis. This is related to the goals of the analysis and the kind of questions that have to be answered with the analysis, or the decisions to be made afterwards. Datasets may contain irrelevant variables regarding to these goals, questions or decisions as well as redundant variables [84]. As previously stated, the quality of discovered knowledge usually depends on the quality of the data used. Also, the success of some learning schemes, in their attempts to construct models of data, often relies on the identification of a small set of highly predictive variables. The inclusion of irrelevant, redundant and noisy variables in the model can result in poor predictive performance and increased computation. This is the reason why a previous work for determining the variables to be kept is required.

At this stage of the analysis an expert-guided variable selection process is to be conducted. This means that, according to the goals of the analysis, the existing knowledge on the targeted Environmental System and the meaning of the available variables, a first selection is performed. This first step is interactive with the experts. For example, given a database with information about water infrastructures in a set of villages [121], with an interest in improving drinking water access to households, all variables describing the status of the sewer network infrastructures can be dismissed.

In case of doubt, it is preferable to be maximalist and keep a variable that will be disregarded later along the analysis, than to find during the data mining step that some variables initially dismissed become required. The pre-processing can imply several transformations on the data matrix (like selection of some rows) that may complicate the addition of other variables in an intermediate point of the modeling process. Re-processing a certain data mining model without some variable has a much lower computational cost than re-pre-processing the whole database.

4.3. Practicum

Most of the commercial packages or data management products provide very sophisticated tools to perform filters over data, as well as to select subsets of variables from the data matrix, most of them using Boolean expressions (or logical rules) to specify the inclusion-exclusion criteria. Nowadays, it is quite usual that the system permits to activate and inactivate several filters intermittently.

Sometimes, when data has to be retrieved from a big information system, the original data matrix is the whole reference data warehouse, and the filter is expressed in an SQL query, already providing the working data matrix with the required rows and columns to be considered in the analysis.

When data has to be acquired from scratch, determining both the inclusion/exclusion criteria and the subset of relevant variables is the first step of the data acquisition process. Then, other disciplines become relevant: sample theory will help in observational studies, when measurements only require system observation. Experimental design theory will help when specific control conditions of the ES must be artificially created to make the measurements. This issues are out

of the scope of the paper and regard the data acquisition step, but will determine in which form data comes to the study, how the working data matrix is built and which pre-processing tasks are required to get it. Of course, selecting data from a pre-existing data matrix, a pre-existing data warehouse or requiring to perform the experimental or observational measurements from scratch implies very different costs. It is then relevant to consider which data is already available in previous data bases at the beginning of the study.

5. Data Visualization

Visualization is a powerful strategy for leveraging the visual orientation of sighted human beings. Sighted humans are extraordinarily good at recognizing visual patterns, trends and anomalies; these skills are valuable at all stages of the KDD [229] [156]. Many of the pre-processing detection issues, like outlier detection, normality assessment or linearities, randomness of missing values or multimodalities, can be assessed through visualization techniques [198].

Graphs commonly used for classical exploratory visualization, like boxplots, histograms, time series plots or scatter plots perform poorly considering the great number of variables involved in ESs datasets, along with their complex interrelations or spatio-temporal references. Thus, more sophisticated visualization methods are required, as for example:

- Distributional plots,
- Three, four, and five dimensional plots (color and symbols may be used to represent the higher dimensions),
- Density plots
- Dimension scaling, like log scales
- Rotary frames,
- Animation and interactive graphs,
- Geo-referenced visualizations and maps.

Most data mining packages, such as Weka [98], TABLEAU [103], QlikTech [157], include visualization tools, while more advanced features are provided with wide-spread tools such as Matlab, a dedicated data language such as IDL [64] or the CommonGIS tool [9]. Dedicated interactive visualization tools are also available, such as GGobi [212] or GODIVA2 [27]. Visual representations are extremely effective at pre-processing stage to identify the basic pre-processing issues to deal with. For example, some data errors or existence of outliers or structural missing data might be visible on certain visualizations.

Also, some KDD methods, mainly those coming from classical statistics, usually involve some technical assumptions that must held on the target domain to guarantee the validity of the model. This is the case of linear regression, for example, in which independent observations, non-correlated variables, conditional normality of the response variable and linear relationship between the response variable and the explanatory variables are required. Some of these properties can be assessed on data just before the analysis and can help to decide, for instance, whether the linear regression is the proper approach for the target model or other alternatives like artificial neural nets can perform better on our data. Part of these properties can be assessed using visualization tools, like assessing normality by means of histograms or Q-Q plots, or the normal probability plot and the Henri line [214]; or Scatter plots can help to see independence or linearities.

This is not exclusive for statistical methods. Inductive machine learning methods, like Decision Trees for example, require balanced data sets to provide reliable results, or certain clustering methods require constant data density. These kinds of issues are easily observable over some charts or scatterplots and become extremely difficult to evaluate in a numerical way.

Although visual assessment often is very efficient in these cases, when possible, tests can be introduced in automatic repetitive KDD procedures, since they do not require human interaction to decide: the χ^2-tests for independence, or the Shapiro-Wilks for normality [40], the assessment of multivariate normality it is not so easy, but Mardia test can be used [194].

However, these tests usually have their own technical assumptions in turn and it is important to ensure these last hold to guarantee the reliability of the significance test results

Thus, by making previous visualization of data, one can easily point out which kind of pre-processing operations are required either related to intrinsic data quality, or to make available the application of some particular Data Mining method.

6. Outliers and influential observations: detection and treatment

Outliers are instances with very extreme values in one or more variables [17]. Some data mining methods are robust to the presence of outliers, like hierar-

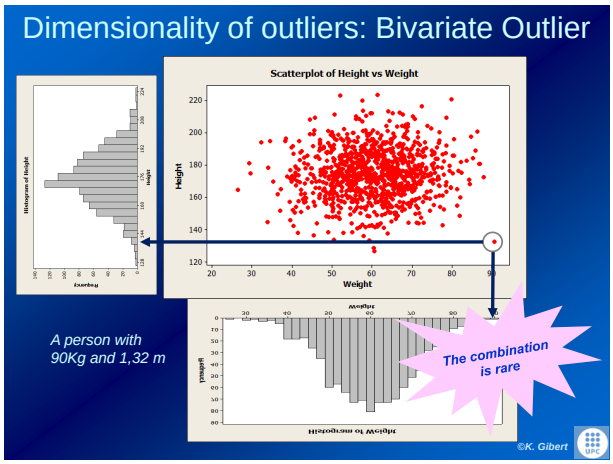


Fig. 4. Dimensionality of outliers

chical clustering methods, while others, like regression or density estimation methods, can provide highly disturbed results when outliers are present in the dataset. For non-robust methods, it is crucial to detect outliers and to properly treat them in order to get a model that fits well the reality and not too much distorted by the outlier itself.

6.1. *Dimensionality of outliers*

Something often ignored is that outliers have a certain dimensionality. This means that sometimes an observation appears as an outlier regarding a certain single variable, but sometimes it appears as outlier in a subspace of higher dimensionality. To clarify, a two-dimensional example is shown in Fig. 4. The same situation can also appear in higher dimensional spaces, although we are not able to visualize in a clear way yet. In Fig. 4, a scatter plot for COD (Chemical Oxygen Demand) and BOD (Biochemical Oxygen Demand) of a WWTP is shown. In one day, the WWTP inflow has a value of COD concentration equal to 1279 mg/l, and BOD concentration equals 198 mg/l. None of these values are particularly extreme (see the marginal histograms) [86]. This means that this day is not considered as an outlier, neither regarding the COD, nor the BOD by themselves. However, looking at the plot, things change. In the bivariate space formed by BOD and COD the point (1279, 198) becomes an outlier. The *relationship* between the two values is the rarity. Thus, individuals appear as outliers or not, according to the subset of variables considered together.

Outliers become dangerous only when they perform as *influential observations* in the analysis. An influential observation is an observation that strongly determines the results of a certain analysis. In the top-right corner of Fig. 4, a second outlier is much far from the general data cloud than the one marked by the circle. However, the former aligns better with the general association between BOD and COD. Thus the one with the circle in Fig. 4 is more influent. Influent observations are well-known in the context of regression analysis, but they can also distort other non-robust models. This means that they may spoil modeling. When the outlier is far from other observations, but follows the same model, it is not dangerous.

6.2. *Causes of outliers*

Outliers might be produced by different reasons. Understanding them helps to diagnose and deciding the proper treatment:

- Error: COD=1279 mg/l is not compatible with low BODand low suspended solids.
- Informative point: a single point from a missing part of the population (a single observation referring a sensor failure, etc.)
- Member of another population
- Intrinsic extreme value: i.e. the case of COD=1579 mg/l with BOD=987 mg/l
- Missing code: 99999 usually appears as an outlier for the univariate distribution of the variables (Fig. **??**)

6.3. *Outlier detection*

Using domain-related additional information provided by the experts, like the normal range of values of the variables, supervised outlier detection can be conducted [1]. When this information is totally, or partially, available [2], it is convenient to take advantage of it. In front of a lack of this domain-related information, unsupervised outlier detection must be used.

Graphical techniques were once the most common method for unsupervised outlier detection, but increases in database sizes and dimensionality have led to a variety of automated techniques [206]. The use of standard deviations is possible when considering variables following symmetric distribution; but outliers may also take the form of unusual combinations of two or more variables, as said before, and then, they only manifest in the proper dimensionality subspace. This means that graphical representation will not be useful over more than three dimensions. In that case, hierarchical clustering, where outliers form singletons

[144] or other methods [31] might help. Also, some specific statistical tests exist [22] but they are only implemented in some software tools, and usually assume some specific univariate distributions.

In any case, the data point should be analyzed as a whole to understand the nature of the outlier and multivariate approach is required. See [41] for a survey in the field. Determining whereas an outlier is performing as an influent observation can be quantified by the Cook's D coefficient (Cook 1979) and it is important to determine if the observation belongs to the target population or not.

6.4. Outlier Treatment

The treatment will depend on the nature of the outlier. As said before, the outliers become dangerous when they perform as influent observations. If not, they do not need to be treated. The influence of outliers can dramatically affect the results or certain methods, and should feature the choice of tools used throughout the rest of the process. See [163] for an interesting discussion on the dangers of simply eliminating rows with outliers:

> In 1985 British scientists reported a hole in the ozone layer of the Earth's atmosphere over the South Pole. [...] The British report was at first disregarded, since it was based on ground instruments looking up. More comprehensive observations from satellite instruments looking down had shown nothing unusual. Then, examination of the satellite data revealed that the South Pole ozone readings were so low that the computer software [...] had automatically suppressed these values as erroneous outliers! Readings dating back to 1979 were reanalyzed and showed a large and growing hole in the ozone layer [...] suppressing an outlier without investigating it can keep valuable out of sight.

When the outlier is a mistake, the best option, if possible, is to come back to the original observation and correct it. When it is no more available, then the observation must be substituted by a missing code and conveniently treated.

When an outlier belongs to a segment of population that has not been properly represented in the data sample, it is convenient to enlarge the DB with more observations in the same subpopulation. If this is no possible, then the scope of the analysis should be restricted to the subpopulations sufficiently observed. For example, in the analysis of biodiversity in a river, some days may correspond to a scenario in which forbidden industrial discharge occurred, with consequences in biodiversity. As this is, fortunately, infrequent, only few rows of data matrix are involved. Getting a global model for biodiversity requires enlarging the data sample to more days with industrial discharges. Otherwise, the study must be restricted to normal days. Another example is to design the analysis on air pollutants in a certain urban area involving the past five years and, provided that measurements for the 1^{st} and 2^{nd} year are too scarce in some cities to get a reliable analysis, a restriction to the last three years would be convenient.

When the outlier is a member of another population, included in the data sample by mistake, it is convenient to treat it apart, and properly report in an explicit way that the general model is obtained without considering it.

When the outlier is the natural skew of the distribution it often will not perform as influent observation and will follow the general model (Fig.4). Then, it is convenient to keep it for the analysis.

When the outlier is a numerical code used for representing missing data (i.e. 999999), it is convenient to substitute it by the missing code and proceed with missing data treatment (imputation).

It is of utmost importance to report any decision made over outliers in a clear and explicit way that permits the end user to properly evaluate if something is wrong with the data or outlier treatment decisions must be revisited.

7. Error detection and treatment

Corrupt data can come from sensor failure, data transmission, improper data entry, etc. Often, it produces values out of range [63].

Corrupt data can also be identified by evaluating consistency rules over the data, representing well-known relationships among the variables. As an example, in Figure 4, the mistake in BOD=1279 can be identified by applying specific domain knowledge: In fact, BOD and COD are correlated and it is not possible that one is high and the other one is low simultaneously. To see which of both is wrong, other variables help, like suspended solids, also positively correlated with them.

To identify corrupt data, specific domain knowledge is required together with intensive interaction with the expert.

When an error is detected, they are basically two ways of treating it:

- Correction: This means to come back to the original data source and try to retrieve the correct data. This is the preferable option, when available.
- Missing production: When correction is not available, the data must be substituted by a missing value and conveniently imputed.

A special task in this topic is terminology normalization for qualitative variables. More and more, qualitative variables play a bigger role in data mining. It often happens that a same modality is expressed with different strings, which are not properly recognized by the software as equal, thus significantly altering the data mining results. For example, data coming from different countries can have modalities expressed in different languages (a qualitative variable indicating *risk of fire* can have *Alt, Alto, High, Elevé*, all of them meaning *high* in different languages; they will be treated as different in the analysis if they are not properly preprocessed).

In a more specific situation, as common Thesaurus are not extensively used yet to fill-in qualitative variables, slight differences in the words spelling might have the same consequences (accents, blank characters, capital letters, special characters like dashes, slashes or points are the biggest enemies of qualitative variables' treatment). As an example, in the dataset containing the measurements from Hipparcos satellite, one of the variables is the *Spectrum* of the measured star [82]. The file contains measurements over 87475 stars, and the descriptive analysis says that they are 3655 different Spectrum values. Some of the stars show a Spectrum type *"M2 :"*, while other show *"M2 : "* or even *"M2:"*, which in fact are the same, but codified in different strings, and the software do not identify as equal. Also *"M3 J"* and *"M3 J"* The difference is subtle, only some blank spaces distinguishing them, but the effect over data mining results is important. Also *"A0p (Si)"*, or *"A0p Si"* and also *"A5 IV-V"* or *"A5IV/V"*. After standardizing the labels of all modalities of the variable, the number of modalities reduces to some hundreds. So, very careful editing is required to normalize terminology before starting the analysis.

Any decision made for data error handling must be transparently reported.

8. Missing Data

Often, a number of cells are missing from the data matrix. In fact, the existence of missing data is not negligible in real applications. A missing data may occur by many different reasons and may have different natures. First, missing data must be detected. After, diagnosed. Accordingly, proper treatment can be determined.

8.1. Types of Missing data

Most of the references found in the literature classify missing data into missing completely at random, missing at random and missing not at random [8] [145]. However, the origin of the missing data is analyzed in depth, so that the user can get criteria to understand. From the application point of view there are basically two main families of missing data: random (completely or not) and non-random, and it is crucial to distinguish between them.

- Random missing data (include missing completely at random and missing at random, usually referred in the literature) are randomly produced and do not follow any particular pattern. This means that, even if unobserved, it is correct to assume that they follow the same distribution of observed data. Thus, observed data can provide useful information about them. They might appear by chance:
  - an employee is copying a set of manual measurements from paper to computer and incidentally skips one of them
  - a sensor is temporarily loosing connection with the server and one reading is lost
  - a forced missing value often is also random. It results of detecting a wrong observation. If it cannot be properly corrected, then it has to be transformed in a missing.
- Non-random missing values are produced by identifiable causes; normally they come from a specific subpopulation; and there are differences in their behavior with respect to the observed data. In this case, observed data is not providing useful information about them. They might appear from many different reasons:
  - Because it is deliberately hidden: Too high concentration of nitrates in the outflow of a wastewater treatment plant in a concrete day can be deliberately substituted by a missing to avoid a fine. Some industries do not provide emissions of certain pollutants to avoid extra taxes.

- Because data is explicitly not provided: In surveys on water consumption carried out by water companies, some users do not explicitly declare sensible variables to prevent eventual requalification of the household profile, often with consequences in the bill. Some of these variables are the existence of garden or pool on the property, the use of devices saving water, the number of occupants of the dwelling or the level of family income.
- Because it corresponds to a special value. Presence of filamentous bacteria in a WWTP plant might indicate a further bulking and problems with the normal operation of the plant, providing effluent with too high concentration of organic matter, over the permitted limits.
- It is not possible to obtain the measurement, by some reasons:
  - Because the technology is not available: in the case of Hurricane IKE, occurred in Cuba in 2008 [23], anemometers in Pinar del Río (western Cuba) ceased to measure *wind speed* over 250 km/h, since they simply broke, thus producing missing measurements after the moment in which wind reached this speed. In a volcano eruption, sensors are used to measure the toxics emitted by the eruption, but no measures are available in the middle of the crater, because the heat fudges the sensors. Also, water age in potable water distribution networks, relevant to predict biofilm development [191]. It is no available under conditions far from the steady state, since it depends on water quality and some consumption parameters that cannot be determined under conditions far from the steady state;
  - Because of a lack of privileges to get it: measurements of air quality in a grid of points of a certain mountain region including a military area will produce missing values in the points falling inside military zone.
- Lost: The measurement has been taken correctly, and lost. This can happen when digital data bases corrupt after time, and past data disappears; in migration of software versions, sometimes this might also occur. Another example is the study of migration of birds, in which GPS trackers are used to mark the migratory birds. Some of the measures can be lost, just when visibility from the satellite is temporarily lost [223] [26].
- Structural: data unavailable for a certain part of the population: As an example, the variable *form of the beak* of a mammal, in a database on animal species leaving in a certain ecosystem, like a natural park or a forest. *Number of legs* of a fish is another example. These missing values often appear as a consequence of a wrong data encoding policy at data collection step. Meaningful values (like *lackOfBeak*) are a better option to represent structural missing values instead of missing values, this requiring a proper design of the data collection process.

8.2. Missing data representation

It is important to get precise information about how the missing data are represented in data matrix, as part of the meta-data information.

Sometimes, missing cells are marked as a *, ?, NA (Not Available), blank space or other special character or special numeric code such as 99999.

Blank space can be problematic when imputing the data matrix in some specific software, as it might happen that the software does not detect the blank and shifts values of the row one position to the right.

Numerical codes assigned to missing data like 99999 can produce severe mistakes in calculations if not properly treated, as they can be wrongly incorporated in the analysis.

Missing code automatically recognized by the software ("*" in Minitab, "?" in Weka, "NA" in R package, etc.) can also be problematic, if the user is not aware of the implicit assumptions that the software is making over data, and missing treatment.

8.3. Detection

For missing data represented by means of the missing code properly recognized by the pre-processing software tool ("*", "?", "NA", etc.), simple basic descriptive statistics of the variables will directly report it.

When they are represented as other kind of numerical codes, basic visualization of the variables (in a histogram, for example) will show a missing data pattern, as a very extreme value of the variable with certain prevalence. In Fig 5, the concentration of suspended solids at the inflow of a WWTP is shown. The variable is measured in mg/l and 9999 is a non-possible value

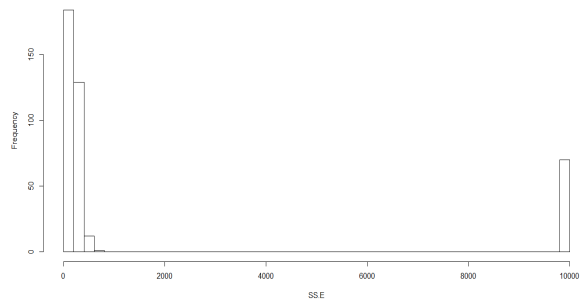


Fig. 5. Pattern of missing codified as numerical code

for the variable, indeed representing the missing value code. Understanding if this is a real extreme value of the variable or a value out of range, used as a numerical code for the missing value, requires interaction with the expert and some domain knowledge.

8.4. Diagnoses

The main issue is to identify whereas the missing data of a certain variable are random or not.

When a variable contains a block of missing data, it is useful to check together with the expert if they might be structural missing data that corresponds to particular conditions (as said before, mammals do not have beak).

In the general case, the Little's test [145] can provide a technical outcome about the randomness of the missing data. It is a statistical test based on checking differences in the multivariate distribution of data between the set of observation with a certain missing data pattern and the others. When differences are big, missings are non-random. The R package implements it. See [183] [179] for details on tests on missing data.

8.5. Treatment

Nowadays, most of the data mining softwares claim that they can deal with missing values. Apparently, this seems to mean that one can forget about pre-processing them. However, one needs to know well which kind of missing data treatment is going to be automatically performed by the software in each different data mining method and properly evaluate whether this treatment fits well the data and goals of the analysis or not. In most cases, the pre-processing software tool is using the *listwise deletion* strategy or a *complete case analysis* strategy, both meaning that rows of data matrix containing missing data are *excluded* from the analysis. Removing rows with missing cells from a dataset may cause serious problems if the missing values are not randomly distributed [125]. As this decision is transparently taken by the software, when the missing data are non-random, this means, in practice, that the analysis is performed without a certain subpopulation. For example, in the database with animals leaving on a certain ecosystem, all registers describing a mammal will be ignored in the analysis, just because they have a missing in the *form of the beak* variable. The first consequence is that the resulting description of life in the ecosystem obtained is implicitly ignoring all things happening with mammals. As this decision is transparently taken by the software, the risks to assume that results describe well the complete ecosystem is very high, with potential dramatic consequences. So, using the default missing data treatment provided by the preprocessing tool software directly impacts on the scope of the analysis results, and restricts the conclusions to the smaller part of the population really analyzed by the software. In some cases, particularly when a big number of variables are considered, this produces software errors: it may happen that none of the data rows contains values in none of the remaining variables, or that some of them become constant, and then the software will intend to analyze an empty dataset or one with null variance. Also, this might produce loss of significant information (imagine for example, excluding from the analysis data row that lacks one out of 100 variables).

Also, some systems assume substitution by the variable global mean. This permits to take advantage of the whole data rows, and it has the property of keeping the variable means invariant after imputation. However, this reduces the variance of the variable, having consequences in any inference operation and distorting the relationship with other variables, with effects in coefficients like covariances, and biases in inducted models.

Thus, leaving the missing data treatment to the software requires a good knowledge of what is the software doing, and evaluating if it is correct for the particular case analyzed. Otherwise, this kind of automatic treatment can seriously damage the analysis in a transparent way.

A better option is *imputation*. Imputation [201] is a complex process for converting missing data into useful data using estimation techniques.

For qualitative variables, sometimes it is useful to use a specific modality to encode missings, like *UnkownValue*. This is treated as an ordinary modality by

the software pre-processing tool and the variable can be entered in the models. Impact of the qualitative missing in modeling could be then better evaluated, and decisions at the data mining step made accordingly.

Structural missing data: the experts might provide rules to manually impute them, i.e. substitute *form of the beak* of all mammals by *lackOfBeak* or 0 legs for fishes.

For temporal data, *interpolation* might be a good option. When data with different granularities are represented in a data matrix, the slower (or higher level) variables (measured less frequently) will produce lots of *false missing data*. They can be interpolated between useful data in case of temporal data, or properly propagated to the lower abstraction levels in case of spatial data, by taking advantage of some implicit assumptions, like constant behavior inside the interval (whether spatial or temporal), this being the reason for saving measurements at certain intervals (or granularity levels). As an example [86], consider the levels of BOD, measured in a WWTP, once a week, while pH in the bioreactor is measured daily. Since BOD evolves very slowly and require a laboratory test, they are measured less frequently. If the dataset contains one daily register, this situation is producing 6 weekly missings for BOD when, in fact, it can be assumed that it behaves constantly from week to week. Monotonous interpolation or constant imputation will be used according to the knowledge on the variables provided by the expert. For example, in water resources management, different levels of aggregation regarding a water consumption model cannot be disconnected between them and from the global information available on the resources [60] [105]. This coexistence of different granularities in both time and space, in geospatial phenomena, characteristic in ESs, is a reason why learning global models for environmental phenomena is so difficult [16].

In a most precise approach, *model-based imputation* builds a predictive model for each variable to be imputed. Many different predictive models has been used to this purpose, like regression [213], local least square imputation [127], Support Vector Machines [107]. In [12] a decision tree has been used to impute the type of cooking habits of a household, and an artificial neural net to impute the PM10 emissions of their traditional stoves. However, model-based imputation is criticized by many authors for being time consuming [94].

A more sophisticated proposal consists in performing the missing imputation by preserving the relationships among variables in the entire dataset. This is the case of Maximum Likelihood Estimation (MLE method) and related methods [145] [219], which performs well, with good theoretical properties, provided that a sound statistical model is available for the target problem, which is not always the case, especially in ESs. Also, building a MLE from scratch at every single application is often difficult and too time-consuming. The EM algorithm is also well founded, and involves bootstrapping to find standard errors for imputation, but requires high expertise to be properly used.

Multiple imputation [7] [15] [111] uses a distributional model for every variable and estimates several complete imputed data matrices by repeatedly making random trials from the reference distributional model. Then, all the imputed data matrices are mined and a consensus model is built, combining the results obtained for each of the matrices. It requires realistic statistical assumptions for the variables, only specialized software performs it, and it is no trivial to get the consensus final model.

In ([57] [203] surveys of missing imputation data can be found, some of them with a very practical approach [93]. The most important aspect is to avoid false assumptions when considering imputation methods, which may have a significant effect on the results. All the methods have pros and cons, and the choice must be made with care.

The most sophisticated a method is, the more precise are the imputations provided, but the more expertise is required to apply properly, the most time consuming is. Considering that missing imputation is a specific part of the pre-processing step, which in turn is only the first step of the KDD process, excessive time consuming methods are not the best option in real applications, where often answers are required in a very limited time.

In the following, some intermediate solutions from the context of data mining that try to provide acceptable imputation in short time are mentioned.

Distance-based methods can be used. The most basic approach in this line is using the k-NN [18]. The idea is to use distances to identify the closest instance in the data base to the one containing the missing, and to use the value of the nearest neighbor for imputation. This can be extended to consider the mean value of the first k nearest neighbors as imputation value. The method is quick and performs well, provided that the variables used for the k-NN are complete, i.e. do not contain missing values. Otherwise, distances that can deal with missing values in turn are required, and stan-

dard softwares do not implement them. Other distance-based works are [189].

The MICE method [15] solves in an iterative process the existence of missings in the other variables. Each variable is imputed based on a regression model built with the remaining variables. All missing data of regressors need to be previously imputed. They are initially imputed by the mean, and all variables imputed iteratively until convergence.

The MIMMI method [79] is based on performing a previous clustering with a subset of complete variables and to impute missing values of the remaining variables with the conditional means of discovered clusters. Such an approach may even help to identify errors in the dataset (when some observation is placed in a wrong cluster and the reasons are investigated).

Another method proposed in [13] was to use qualitative indicators (i.e. by assigning quality labels instead of predicting values) for substituting missing or erroneous values. Similar work has been done for substituting missing or erroneous values with fuzzy sets, probability distributions or confidence intervals.

Sometimes, the best imputation method depends on the particular data mining method to be applied afterwards [151] [65].

It is important to impute missing data before using the corresponding variable to create other variables or indicators, in order not to propagate the missings to the new variables.

It is of outmost importance to report the missing imputation method used explicitly for allowing the end user to evaluate if the missing imputation decisions are acceptable or not for each specific application.

9. Relevance or redundancy detection and dimensionality reduction

In section 4 some strategies for an expert-based selection of rows and columns from the original data matrix are presented. In this section, methods for a selection of both rows and columns from a more technical perspective are presented. These methods can only be applied after errors in database have been corrected and outliers and missing values have been properly treated. The main aim is to detect irrelevant rows or columns from the working data matrix or those redundant that inefficiently increase the dimensionality of the dataset without providing any additional information. Building the models without those irrelevant or redundant observations or variables is expected to not significantly decrease the model quality.

9.1. Instance selection

Sometimes, redundancy of instances can produce unnecessary big size of the dataset. It also can produce imbalanced datasets, which can disturb the data mining process.

Detection and treatment of redundant instances is relevant in the pre-processing step to guarantee proper performance of the data mining. The aim of the instance selection techniques [146] [193] [118] is to reduce the number of examples of the dataset with a minimum loss of information (predictive accuracy, clustering accuracy, etc.); thus retaining the relevant instances and removing the redundant ones, which probably are not useful for modeling.

Instance selection techniques obtain a subset S, $S \subseteq T$, of the original training dataset $T \subseteq I$, such that S does not contain redundant or superfluous instances, and the accuracy in the models mined (usually predictive or clustering models) is nearly equivalent using S or T as the training set.

Instance selection techniques can be organized according to the processing way of how S is computed:

- Incremental methods: start with an initial empty new set of instances ($S = \emptyset$), and proceed by incorporating only the relevant instances.
- Decremental methods: start with $S = T$, and proceed by removing the redundant instances until getting the final set S.
- Mixed methods: use a mixed incremental or decremental way of the selection process. In some step of the algorithm, they proceed incrementally by adding new instances or decrementally by removing instances, and at a further step, they proceed in the opposite way.

According to the bias dimension, like in feature selection methods, the instance selection methods can be divided into *filter* methods and *wrapper* methods.

9.1.1. Filter instance selection methods

These methods use several selection criteria, but none is based on the feedback of a mined model from the data (classifier, clustering, etc.). There are some special filtering methods, sometimes noted as *exhaustive* methods, which mainly use the size or some percentage of instances to make the selection process:

- Random methods: take randomly just the percentage of instances that the system is able to process (sampling).

– Supervised random methods: take randomly the same percentage of instances within each class group (stratified sampling).

Random filters are repeatedly used as the first step of iterations in resampling methods [92], like Bootstrap, Jackknife or Monte Carlo methods [230] and ensemble methods [56] [25]. Resampling methods are nowadays becoming very popular in the context of big data, since they belong to a *divide and conquer* approach that makes dealing with huge datasets possible. Also, resampling techniques, and in consequence, repeated random filters can be involved in some validation procedures as cross-validation [129].

Among the *filter methods* in the literature, we can find methods based on *k*-d Trees [166], where a binary tree is constructed. Some authors [36] [208] have proposed the idea of using clustering for the selection process, taking the centers (or nearby) of the clusters as the selected instances. Some methods following those ideas are the GCM (Generalized-Modified Chang Algorithm) method [162], the NSB (Nearest Sub-class Classifier) method [218], the CLU (Clustering) instance selection method [152], and the OSC (Object Selection by Clustering) [174]. Other authors suggested the assignment of weights to the instances and selecting only the relevant instances like the WP (Weighting Prototypes) method (Paredes 2000) and the PSR (Prototype Selection by Relevance) method [175].

Other filter methods are based on other semantic criteria, based on some properties about the instances. Instances are defined to be *border instances* or *inner instances*. A border instance for a class C is the nearest neighbor of an instance from another class C'. Some filter methods based on selecting border instances are the POP (Pattern by Ordered Projections) method [196], which discards inner instances and selects some border instances, and POC-NN (Pair Opposite Class-NearestNeighbor) method [190], which selects border instances.

9.1.2. Wrapper Instance Selection

The selection criterion is based on the accuracy obtained by a model mined with the considered instances. Usually, those instances not contributing to the accuracy of the model are the candidates to be discarded from the training set.

Most of the wrapper methods proposed in the literature are based on the *k*-NN classifier [77]. This is the case of Condensed Nearest Neighbor (CNN) [104], one of the oldest, the SNN (Selective Nearest Neighbor rule) method [197], and the Generalized Condensed Nearest Neighbor rule (GCNN) [44]. Other early instance selection methods focused on discarding noisy instances in the training set, like the Edited Nearest Neighbor method [226] and some variations like the *all k-NN method* [215] and the *Multiedit* method [55]. The IB2 and IB3 (Instance Based) methods were proposed by [4], which are incremental methods. Other methods related to k-NN are those proposed by [227]: DROP1, DROP2, DROP5 (Decremental Reduction Optimization Procedure), which are based on the concept of *associates*. The associates of an instance *i* are those instances such that *i* is one of their *k* nearest neighbors. Another method related to the associates concept was the Iterative Case Filtering algorithm (ICF) proposed by [32]. In addition, *evolutionary algorithms* have been used for instance selection [54] [74] [37] [24][138]. In [76] it was proposed a *memetic algorithm*, combining evolutionary algorithms and local search in the evolutive process, and [73] proposed the Clonal selection Algorithm (CSA), which is based on the artificial immune system approach. Finally, some authors proposed the taboo search for the selection of the instances [234] [39].

A special wrapper method is the *Windowing method* [188] [122], which uses a percentage of random instances to make the initial selection process, followed by feedback guidance; the selected data is mined (under a classifier/predictive model) and the inducted model is applied on the remaining data. Those which have not been correctly predicted are added to the selected dataset and the process is repeated. It could save up to 20% of the data.

9.2. Relevance of variables and dimensionality reduction

A major problem in data mining is to find out which are the relevant variables or features to be taken into account. When experts are available in a particular domain or application, experts could give their advice. When there is no expertise available, or the selection provided by experts wants to be technically refined, some automatic methods should be used.

Feature weighting methods quantify the relevance of the variables. *Feature selection* identifies the subset of relevant variables. Factorial methods build a smaller set of new synthetic variables conserving most information from the working data matrix.

When the number of variables is too high to deal with in a reasonable way, which is not unusual in data mining context, a data reduction method can be ap-

plied. Either Factorial methods or Feature Weighting or Feature Selection are suitable possibilities in these cases.

Next subsections give details.

9.2.1. Feature weighting

Feature weighting [3] [170] techniques provide a ranking (weight) of the attributes (or variables) according to their degree of relevance.

The degree of relevance of a variable X_k is expressed by means of a weight (w_k), usually in the interval [0,1]. They assign high weights to more important variables, and low weights to irrelevant or redundant variables. This way, it is possible to decide which the important variables in a dataset are. These techniques are useful only when the importance of the variables for the dataset can be taken into account in the analysis. For instance, feature weight assignment is frequently used to denote the relevance of attributes in *similarity* or *distance-based methods*, like clustering or some inductive classification rule methods, allowing to emphasize the relevant variables in the distance/similarities.

One of the problems in the feature weighting methods is how to decide when a set of weights is better than another. They must be evaluated in terms of the performance of a task:

- In supervised domains, a classification task could measure the accuracy of the label predictions for unlabeled instances.
- Unsupervised weighting methods assign weights to variables without any knowledge about class labels, so this task is presumed more difficult [51] [109]. In fact, they use alternative measures like significant changes in similarity or distances to evaluate the goodness of a set of weights.

In [225] a conceptual framework for the classification of weight assignment methods is presented. This framework consists of five dimensions: *Bias* feedback, preset, *Weight space* continuous, binary, *Representation* Given, Transformed, *Generality* Global, Local and *Knowledge* Poor, Intensive. Most important dimensions are the Bias, the Weight Space and the Generality.

The *Bias* dimension refers to whether the weight learning bias is guided by feedback from the performance algorithm (i.e. the classifier), or whether it is instead, a preset bias (i.e. maximize intra-class similarity and minimize inter-class similarity) that does not incorporate performance feedback.

Bias is useful to separate those algorithms that use feedback from those that do not use it. The first ones are known as *wrappers* [131]. The second ones are named as *filters*. Presumably, the wrappers have an advantage. Their search for attribute weight assignment is guided by how well those assignments perform. Thus, there should be no mismatch between the biases of the weighting and performance algorithms.

The *Weight Space* dimension distinguishes *feature weighting* from *feature selection* algorithms. The latter are a proper subset of feature weighting algorithms because they only employ binary weights (i.e. 0 or 1), meaning that the attribute is either deleted or retained. Although weight assignment improve accuracy in classification and retrieval tasks, feature selection is vital to reduce the dimensionality in learning tasks, eliminating irrelevant attributes. In general, feature weighting is more appropriate for tasks where features vary in their relevance, but such methods search larger spaces of weight assignments. Feature selection algorithms perform better when the features are either highly correlated with the class label or completely irrelevant.

Feature weighting algorithms can also be distinguished by their *Generality*. While most algorithms learn settings for a single set of weights that are employed globally (i.e., over the entire instance space), other algorithms assume different weights among local regions of the instance space.

- The assumption of global weighting methods that attribute relevance is invariant over the whole set of instances I is constraining, and often inappropriate. Other algorithms assume that the relevance of attributes is not necessarily the same in the whole domain.
- Two types of local weighting schemes are popular. The first assigns a different weight to each qualitative value of the attribute. Although this allows feature relevance to vary over the values of the feature, it still constrains weights to be identical for all instances with the same qualitative feature value. The second local weighting scheme removes these constraints by allowing feature weights to vary as a function of the instance and their belonging to a class.

In recent years, many researchers are focusing on *supervised feature weighting*, i.e. where a response variable is available and relevance towards it is evaluated:

- Some research in *filter global methods* has been done like the Mutual Information technique (MI) [110] [225], the QM2 method by [160], and [48] about their introduced Cross-Category Feature importance (CCF) method. On the other hand, *fil-*

ter local weighting methods have been proposed in the literature such as the value difference metric (VDM) of [209], the Class Distribution Weighting method (CDW) [108], and the Per-Category Feature importance criterion (PCF) [48].
- Some *wrapper methods* are: the RELIEF-F method [135] [128] and the DIET algorithm [132], and some approaches based on the use of Genetic Algorithms [91] [112].

However, less work has been done in the field on *unsupervised feature weighting*, based on the sensitivity of instance similarities to that variable. Unsupervised scenario is clearly the required one when facing a new unknown database, which we want to mine to discover new knowledge, and usually no reference classification is available. In the literature, one of the few works is [204] on a Gradient Descent technique (GD) and feature selection approach on unsupervised entropy-based method [51]. In [59] a feature weighting method for supervised learning is presented. Derived from it, in [171] two new unsupervised feature weighting methods were proposed (UEB-1 and UEB-2) with promising results.

Empirical works [225] and theoretical ones [141], suggest that the learning complexity is exponential regarding the number of irrelevant variables. Therefore, the failures in the data mining process could be related to a similarity model, and in particular, with an incorrect weight assignment methodology. A comprehensive review of feature weighting can be found in [170].

Indeed, the use of pure feature weighting is quite marginal in real applications. It is more frequent to use them as a previous step of feature selection by choosing a relevance threshold to keep a variable for the analysis.

9.2.2. Feature selection

Feature selection is a specialization of Feature Weighting, where all the weights get binary values: 0 or 1. If the weight of a variable is 0, that means it is discarded, if the weight is 1, the variable is used for the analysis. For a survey of common feature selection techniques, see [42] [28] [147] [202] [161].

Feature selectors are algorithms that attempt to identify and remove as much irrelevant and redundant information as possible prior to learning or knowledge discovery. It is important to remark that most of the feature selection methods perform with respect to a certain response variable that has to be modeled, although some research has been done in the non-supervised scenario [159] [172] [171] [59]. Feature selection can result in enhanced performance, a reduced hypothesis search space, and, in some cases, reduced storage requirement. Usually, analyzing the feature subset selection provides better results than analyzing the complete set of variables [101] [101].

Feature selection can be performed by applying a cutting threshold over the results of a feature weighting process. However, automated techniques for identifying and removing unhelpful or redundant variables are extensively used. Some can use statistical criteria to rank the variables [100] [33], others are mutual-information criteria based [181] [149] or more general concepts like instance consistency [52] [11] [10].

Feature selection methods usually take one of the three forms [101]:

- Filter Feature Selection Methods: Direct examination of the relevance of candidate variables, independently of the machine learning model to be used for the data mining, usually a predictive method. Usually we can distinguish two types of filter methods [200] [142] [53] [21] [38] [133]:
  - Feature ranking methods: they rank features by a metric and eliminates all features bellow an adequate score. Often these methods do not consider potential interactions among features.
  - Subset selection methods: they search for the optimal subset of features. Some of them can take into account the interaction among features
  - Mixed approaches: they can sequentially interleave some feature ranking operations with some subset selection operations, to try to capture all the benefits from both approaches.
- Wrapper Feature Selection methods: Searching the best combination of variables in terms of model performance and feedback. They utilize the ML model of interest as a black box to score subsets of feature according to their predictive power [182]:
  - Brute-force methods: they explore all the possible subsets of combinations of features and get the optimal one [130] [167].
  - Forward methods: starting with an empty set of features, they add one feature at each step until getting the optimal subset of features [195] [185].
  - Backward methods: starting from a set containing all the features, they discard one feature at each step until getting the optimal set of features [165] [207] [210].

  - Random methods: they can try several subsets of features in a random way, trying to avoid to be trapped in local optima (i.e. anytime algorithms) [211] [6] [148] [232] [123] [186].

- An intermediate type of methods are the *embedded methods*: they perform feature selection in the process of training of the ML technique and are usually specific, and included within the given ML techniques [58], such as in decision trees or classification problems [140] and artificial neural networks techniques [158]. Some works give common frames for different modeling methods [150].

Most recent works tackle the problem of computational cost in big data scenarios [235] [120]. As an example, in [191] up to 15 variables associated with biofilm development in water distribution systems were available, including sampling and incubation methods and physical, hydraulic and physicochemical characteristics of water. However, a random wrapper suggested an adequate model using only eight of those variables.

9.2.3. Factorial methods and related

Reducing the original set of variables of a dataset to a smaller set of equivalent variables is an interest topic of Multivariate Statistics from the beginning of XXth century [180]. The main goal of factorial methods is to substitute the original list of variables by a subset of factors that keep the same information as the original dataset. The approach is radically different from the one followed in feature selection. Here, an algebraic metaphor (classical in statistics), in which each row of the data matrix is associated with a K-dimensional point in a vector space (K being the number of variables), is used to find subspaces of smaller dimension where the projection of the original data cloud conserves the adjacency relationships among both variables and individuals. These methods are mainly based in matrix rotations and diagonalizations and several techniques are available according to the kind of information contained in the dataset. The Principal Components Analysis (PCA) [97] is one of the best known techniques of this group and it is suitable for datasets with only numerical variables. Each principal component is a linear combination of the original variables, and the aim is to work with a reduced set of these, such that the loss of information is not relevant. Thus, PCA is suitable for synthesizing an original set of numerical variables into a small number of fictitious variables conserving as much information as possible from the original dataset. Equivalent techniques are available for qualitative data, like simple or multiple correspondence analysis [143]. A general formulation including all factorial techniques as particular cases is the canonical analysis, in which other more general methods are embraced, like non-linear multivariate analysis [134] and methods based on oblique rotations [194]. In the context of finding a relevant set of descriptors from images, non-linear generalizations of PCA based on artificial neural networks and deep learning methods are used [106].

In these methods, the information retained in each of the factors is quantified and the number of significant factors to be retained can be evaluated. However, it has to be taken into account that, in most cases, interpretation of the new variables (or factors) may not be clear. If the factors are later used to build data mining models with them, implications over understandability of the final results might appear.

An appropriate alternative for the case when the interpretation of the significant factors is difficult is to identify the subset of original variables having higher contribution to that factors, and use this subset of original variables in the data mining. This is an intermediate alternative producing a subset of original variables bigger than the set of significant factors, but keeping the data mining in the original space, thus maintaining the interpretability of the obtained models.

10. Transformations

Sometimes transformation of variables may be required. We can distinguish three main reasons: one related to data quality, the second oriented to achieve the technical assumptions required by some specific data mining method, and the third one regarding data interpretability.

10.1. Transform to improve data quality

These are mandatory transformations and required, because when skipped, wrong data mining results will be obtained. Most of the times they are difficult to identify and require good knowledge of the variables' meaning and sometimes details about how they were measured. Three basic scenarios are considered here:

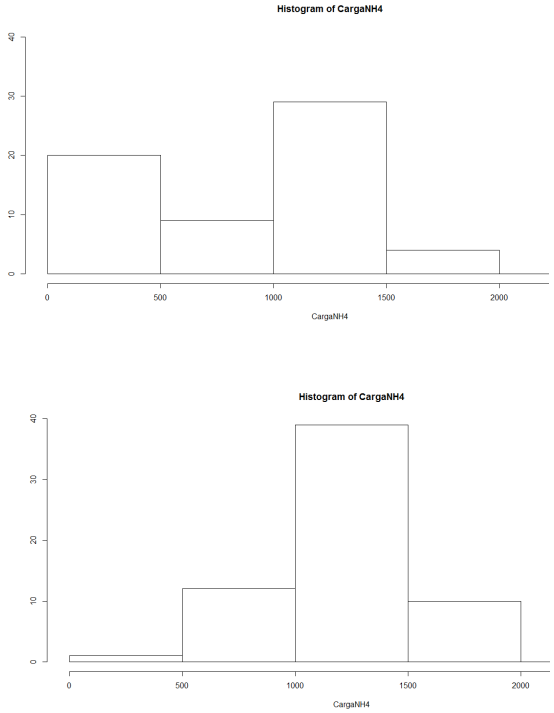


Fig. 6. Daily load of NH4 in a WWTP (up: row data; down: pre-processed data).

- Homogenization of instances: sometimes, especially when the original data matrix is the result of the fusion of different data sources, it may happen that the measurement units of some groups of observations differ. It seems obvious that all observations must be transformed to a single measurement unit. But unless a complete meta-data set for the original data matrix arrives, this information might be masked with dramatic consequences for the further data mining.
  As an example, a dataset contains rows with daily information about a WWTP for a complete year. Among them the NH4 load in the inflow (Fig.6(up)). However, when looking at the metainformation, it is reported that data from March, April and May is reported in Kg/day, whereas it is in g/day for the rest. As all days are properly transformed to g/day one gets a complete different image (Fig. 6 (down)).
- Homogenization of columns: Let us suppose a dataset on National Parks with a National Park in each row and a variable indicating the number of members of a certain protected species, like *lynx pardinus*. If the size of the parks is highly variable all along the database, the number of specimens per park is not directly comparable among parks and the variable needs to be transformed in a density of specimens per surface like specimens/Km^2 of the park.
- Elimination of individual variability: This is typically required in pre-post analysis scenarios, that is, when repeated measurements are taken before and after some intervention in the domain, or the occurrence of some event. As an example, consider measuring air quality before and after using a certain pesticide on some pieces of land. Among others, the data base will contain a couple of columns with the concentration of organometallic components in the atmosphere before (X_0) and after (X_1) pesticide distribution. If a piece of land is in a place with high toxic components in the air, probably it will register also high values of X_1. The effect of distributing the pesticide in the air quality in each piece of land, can be evaluated as the difference in the two concentrations ($X = X_1 - X_0$). Using X for the data mining instead of both X_0 and X_1, as two independent variables, the individual characteristics of each piece of land are eliminated. Thus, for paired observations, differences have to be built before the data mining. For the case of variables with very small values, ratios can behave better than differences, because too many values close to 0 might give numerical instability.
- Compositional data treatment: (introduced in section 3.1). Let us suppose a dataset on rivers including information about the sediments. Suppose these are classified in *boulders, cobbles, pebbles, gravel, sand, silt* and *organic mud*. The dataset has a subset of 7 variables containing the proportion of that material found in the sediments. And the total of those 7 columns is always 1, for all the rivers; this establishes a linear relation among the 7 variables. These are compositional data, and cannot enter as independent variables in the data mining models. In [178] a theoretical explanation is given. The log-ratio transformation usually works well, particularly for those data mining method requiring normality, like classical regression models. Thus, given a subset of s compositional variables, it is enough to choose any of them as a referential variable $X_j, (j = 1 : s)$, build $X'_k = log(X_k/X_j)$, for $(k = 1 : s)$ and use X'_k in modeling. In [35] a nice introduction to the compositional data treatment is given and alternative transformations which make sense in some

specific situations are discussed. Note that sometimes, the compositional data are *hidden* in the dataset. This means that the components appear in non-contiguous columns; sometimes they are presented as concentrations or absolute numbers and they are not directly expressed as proportions. For example, in a microscopic observation of a water sample, it is usual to account for the number of specimens of different families of microorganisms. Thus, the dataset contains some columns with counts of individuals for different families, which need to be transformed into proportions with respect to the total number of specimens counted and treated as compositional data.

10.2. Transformations required to increase interpretability

These are convenient transformations done before the data mining step, oriented to better interpret the resulting models. However, the data mining models would not be incorrect if the transformation would be skipped. In this group, several operations appear: recoding, discretization, functional univariate transformations.

Consider as an example a dataset about car models in a study about air pollution caused by traffic, and suppose some variables with technical characteristics of the car. Among them the X= *miles per gallon*. Building a new variable $Y = 1/X$, the consumption of the car is obtained, much easy to interpret by itself, but also linearly related with pollutant emissions in a more natural way. Even if it is not strictly required, making these kinds of transformations may linearize the relationships with the response variable.

In a second scenario, consider qualitative variables with a large number of modalities; they can be grouped according to expert knowledge (*recoding*). As an example, consider a variable containing the *type of dominant microorganism* in a microbiological sample of wastewater. Several hundreds of species can be identified in microscopical analyses. Thus, we are here in front of a qualitative variable with hundreds of modalities. Nevertheless, this information performs as noise when introduced in the data mining process, because the variable contains too many modalities, each with low prevalence. In these cases it seems reasonable to group the species by Genera. In [78][**?**], 24 Genera related with WWTP are mentioned (*Pseudomonas, Achromobacter, Bacillus, Alcaligenes, Flavobacterium, Arthrobacter, Zooglea, Acinetobacter, Citromonas, Bacillus, Nitrosomonas, Nitrobacter, Nitrospirillum, Vorticella, Aspicidica, Paramecium, Nocardia, Microthrix, Spaerotilus fungi, snails, Phosphate accumulating organisms, algae (lagoons), Viruses, yeast, pathogens (include Salmonella, Giardia, etc.)*). Even more, those can be grouped according to their role in the treatment process. Also according to [**?**]: *Floc forming, Nitrifying, Predators, Nuisance bacteria and eukaryotes, Specialty populations, other*. In that case, a qualitative variable with only 6 modalities is obtained, much easy to work with. The kind of recoding and the granularity to which recoding is required depend on the goals of the analysis. The authors dissuade the reader for recoding operations as a general principle, since this is changing the structure of the original dataset space, by collapsing sets of values in a single one, and relevant information can be missed. However, when the qualitative variable has so much detail that perspective is lost, this kind of recodifications oriented to increase interpretability and provide structure can work well.

Another situation quite common in pre-processing is *discretization* of numerical variables. Here, the authors keep a similar position as for recodifications: better avoid. However, sometimes, interpretability increases by dealing with the qualitative version of the variable. Discretization may work, provided that the cutpoints are non-arbitrary. As an example, a variable accounting for the *level of radioactivity* can be discretized according to the associated risks for humans: < 1000 mSv, *innocuous*; $[1000, 2000)$ mSv, *Minor Nausea, Headache*; $[3000, 6000)$ mSv, illness (*Mod Vomiting, loose of Hair, diarrheas, Mild Headache, Mod Fever, Cognitive Impairment*); $[6000, 8000)$ mSv, severity (*Vomiting, Mod Headache, High Fever, Cognitive Impairment, Hemorrhage, Infection*); $[8000, 20000)$ mSv, lethal in some weeks (*Severe Vomiting, Severe Headache, Severe Fever, Incapacitated*); $[20000, 30000)$ mSV, *lethal in some hours*; $> 30,000$ mSv, *lethal with immediate effect*). In most cases a discretization operation transforms a numerical variable in an *ordinal one*, but sometimes, the result of a discretization may produce a binary variable, or also a qualitative one. For example, grouping all abnormal values of a sensor (either high or low) in a single modality, to send and alarm or not, makes sense in intelligent decision support systems, or control systems and produces a binary variable as a result.

Most of the software tools provide automatic methods to discretize numerical variables (by equal width bins, or equal frequency groups, given the number of

desired groups) [192]. However, despite this transform can increase model performance, the meaning of the resulting beans often becomes difficult to understand and the data mining loses interpretability. In fact, in real applications it is quite common to globally discretize any numerical attribute before applying learning algorithms to datasets, since a number of them cannot handle numerical variables directly. Some data mining techniques increase performance with prior discretization, like decision trees [71]. However, the reader must know that discretization is changing the inner structure of the problem and the results provided by the discretized data can be significantly different from those provided by the original data, even if the intervals are well justified and consensual. This is the case of [89], where *concentration of hormones in blood* are discretized to *Low, Normal* and *High* to identify profiles of thyroids dysfunctions and it is shown that the models obtained with the original quantitative concentrations are better, even if the discretization is based on WHO reference values.

Noise is often a critical issue, and especially with environmental data, some bias may exist that can be removed with a filter [63]. Transformations should always be justified and documented, and the biases that may be introduced noted.

It is strongly recommended to keep the interpretability of transformed variables.

10.3. Transformations required to fit technical assumptions of data mining methods

Some data mining methods involve technical assumptions that must be hold to guarantee the validity of the results. For example, linear regression or ANOVA require normality, homo-chedasticity, and non-colinearity. Also, inductive machine learning classifiers perform badly in front of imbalanced datasets.

Thus, in front of non-normal, correlated or imbalanced datasets, transformations are apparently mandatory previous to the data mining step.

The vision of the authors is to better change another data mining method well adapted to the nature of the data, rather than changing data to adapt the method [88][87]. This is mainly because, in the context of data mining, this may lead to good models, with uncertain usefulness, made of non-understandable transformed variables. Nowadays, there are so many data mining methods, that we presume this technical transformations could be complete dispensable in the preprocessing. As an example, it is enough to change linear regression by ANN to avoid normality, linearity and homochedasticity assumptions and get better predictions for non-normal, non-linear or heterochedastic data. However, we provide here details on the most popular technical transformations, just for the cases in which they make really sense to use them.

Centering: Some methods, like PCA gain properties when operating over centered data. This means to have a dataset with a center of gravity in the origin of coordinates. Each variable X_k is transformed in $X'_k = X_k - \bar{x}_k$, being $\bar{x}_k$ the arithmetic mean of X_k. Centering guarantees that $\bar{x}_k = 0$. Performing PCA with centered data enables to interpret the angles formed by the vectors projecting the variables over the factorial planes as a visualization of their linear correlation, such that 90^o means uncorrelated variables, and 0^o means linearly associated variables.

Standardizing and normalizing: This is often used to avoid variables of different magnitudes coexisting in the same data matrix, which can bias the analysis, and also invalidate comparisons among them. Also, it is used to meet technical requirements of some data mining methods that need the variables to be properly scaled to avoid numerical ill conditions (like some distance-based methods) [192]. All these operations move the original variable to the interval [-1,1]. In the classical standardization transformation named Z-score standardization in [102], the variable is centered and scaled to have null mean and variance equal to 1. The standardization transformation is $X'_k = \frac{X_k - \bar{x}_k}{s_k}, \bar{x}_k$ being the arithmetic mean of X_k and s_k the standard deviation. With this transformation X'_k always has mean equal to 0 and variance equal to 1. Other normalization transformations can also be used to move to [-1,1] intervals, but they do not guarantee variance equal to 1. Normalization based on the range of the variable [99], also referred by some authors as min-max normalization [102] [192]: $X'_k = \frac{X_k - min(X_k)}{max(X_k) - min(X_k)}$; if it is generalized to the transformation $X'_k = a + \frac{(X_k - min(X_k))(b-a)}{max(X_k) - min(X_k)}$, then X'_k moves to the interval [a,b] [99]. Decimal scaling normalization [102], where $X'_k = \frac{X_k}{10^r)}$, where r is the smaller integer such that $max(|X_k|) < 1$; this is what happens when measurement expressed in m are transformed to Km, in this case with $r = 3$. In [192], a transformation to achieve the normal distribution of the new variable is proposed, based on first ranking observations and second substituting original values of the variable by the z-score corresponding to the Z distribution percentile equal to the rank value.

Logarithmic transformation: Useful to achieve homochedasticity. When the variable has values too close to 0, it is important to previously translate it horizontally to avoid numerical instability.

Quadratic transformation: catching quadratic relationships with the response variable. They are useful to linearize the quadratic terms in regression models for example. The same concept extends to high order transforms Y=Xr, r¿1.

In [29], the Box-Cox transformation is formulated as a family of transforms depending on a parameter r in the form $X'_k = \frac{X_k^r - 1}{r}, r > 0$, and $X'_k = log(X_k), r = 0$. For $r = 1$ it simply makes a horizontal translation of the variable. In statistical modeling they are often used to eliminate skewness and normalize asymmetric distributions. See [139] for details. As it can be seen, the purpose of Box-Cox transforms are strictly technical and the interpretation of X'_k is usualy missed.

The *rank transformation* [192] is based on sorting the observations of the variable and enumerating them. The transformed variable accounts for the relative position of the object in this list and becomes an ordinal variable. This is useful when the ordering among individuals is important, but the distance between the individuals in consecutive positions has no particular meaning. For example, when evaluating enterprises to get the contract to manage a WWTP in the next five years, and some evaluation is obtained from the candidates, the differences in the final score are not important, but the ranking is. Also, for distance-based data mining methods, ordinal variables (like *risk of eruption in a volcano*: low, moderate, high, sure) can be first recodified to ordinal encoding (1,2,3,4) and the resulting encoded variable ranked and used in the distance-based methods (like clustering) instead of the ordinal one, for better exploiting the ordinal structure of the variable in the data mining process.

As said before, sometimes data is *fuzzified* to better include uncertainty intrinsic from data into the data mining model [75]. This means to transform a crisp value in a fuzzy number and requires specific fuzzy algorithms to work with them. This process is particularly natural when data coming from certain technological devices provide uncertain punctual measurements together with some evaluation of the uncertainty itself, like variability, error bounds, occurrence regions, and so on. Radar data, GPS data or SCADA data are particularly suitable for this kind of treatment [221].

10.3.1. Transformations associated to imbalanced datasets

As said before, some data mining methods require balanced datasets. However, imbalanced sets appear in many real applications. They appear where the number of instances of one subpopulation (or class, usually the general class representing the usual scenario) is much bigger than the number of instances of some other classes (usually abnormal classes), which frequently are the most informative and valuable ones. These often are the object of interest in the specific problem. Among others, imbalanced datasets typically appear in the identification of anomalies in certain systems designed to work under steady conditions. Some examples are: Water Distribution Systems [116], the detection of oil spills from satellite images [136], the identification of power distribution fault causes [231] and the prediction of pre-term births [95]. This issue is growing in importance since it appears more and more in most real domains, especially in systems where data from the usual scenario are abundant while data from abnormal ones are scarce.

Most classical inductive machine learning algorithms generally perform poorly on imbalanced datasets, because they are designed to minimize the global error rate [119], thus biasing toward the majority class. Classification algorithms tend to classify almost all instances within the majority class, poorly classifying the minority class. Using normal and abnormal data in a water distribution system as they are produced, without discrimination, produces a tolerance to failure unable to ensure capacity redundancy, and network operation under the failure of one component cannot be ensured [154].

To deal with imbalanced datasets several approaches are suitable:

- Pre-processing: trying to balance data by oversampling the minority classes, under-sampling the majority one or a combination of both [19] [43] [96]. Resampling for building several balanced subsets of data and using ensemble methods policies are also used in these cases, to gain robustness. Also, introduction of synthetic data for abnormal situations, simulated on the basis of the real one is also used by some authors to balance the dataset and proceed normally. As an example, in [116] a complex hybrid model, which combines a calibrated classical model of a Water Supply System and a neuro-fuzzy technique, is used to identify leaks and other anomalies in the system. Since data on abnormal situations are

scarce, the calibrated classical model of the hydraulic network is used to simulate abnormal data, thus producing a dataset with enough fuzzy examples to correctly train the artificial neural network. However, in [220] it is shown that over-sampling the abnormal class might produce an unacceptable rate of false positives that might carefully be analyzed.
- Solutions at the algorithmic level: modifying the cost per class [184], adjusting the probability estimation by establishing a bias towards the minority classes [224], etc.

In [68], different pre-processing mechanisms are used in conjunction with a Fuzzy Rule Classification System to deal with imbalanced datasets.

10.3.2. Practicum

In the context of data mining, avoidance of unnecessary transformations is recommended, especially if the transformation decreases interpretability. For example, $Y = log(stream\_flow)$, can produce a variable Y compatible with the normal distribution and suitable to enter in a regression model; however, the logarithm of a flow is something difficult to understand. It is non-recommended to strongly transform data with the unique purpose of getting a better fitting of the model. Indeed, hard transformations can provide high quality models from a technical point of view. However, these are often models built over variables with difficult interpretation. Then, usability of the model will decrease.

If transformations are definitely required, some bias is often introduced into the results. Thus, it is convenient to minimize arbitrariness of the transformation as much as possible (in recoding *Age*, it is always difficult to justify why the modality *Adult* encodes ages from 18 to 65 or from 20 to 70), and this implies that the goals of the analysis must also be taken into account. For arithmetic transformations, imputation of missing data before building the transformed variables is strongly recommended in order not to propagate the missing values to the transformed variable, since any operation with a missing value produces a missing.

Finally, instead of strongly transform the data till it fits on the technical assumptions of the preferred data mining technique (in spite of losing interpretability, or introducing arbitrariness), a better practice is to select a data mining technique fitting well on the target phenomenon and the type and structure of available data [81]. This means that, when data is far from normal distribution, it is better to use ANNs rather than classical multivariate regression, or that an alternative to ID3 method should be used to find groups over data if it is numeric, like clustering, instead of forcing discretization.

11. Creating new variables

Finally, in KDD it is quite usual to build new variables by combining some others included in the dataset. Here, expert knowhow and domain knowledge is usually the guide. Often new variables are nonlinear combinations of the measured variables, thus providing new elements for the modeling step that can significantly increase model performances. When the purpose of these transformations is to make explicit some of the decisional variables that the user really has in mind for decision-making, this is a good practice, bringing significantly added value to the dataset, and opening the way to better data mining models. There are different scenarios:

Creation of aggregates: Like totals. As an example, consider the concentrations of several pollutants in atmosphere, like CO2, nitrogen, arsenic, selenium, antimony in atmosphere, it makes sense to build a new variable with total concentration of heavy metals, by adding concentrations of arsenic, selenium and antimony. This might increase interpretability of the data mining models obtained. However, care is required, as this decreases the granularity of information, by collapsing different profiles of air quality; whereas decisions associated with exceeding of arsenic should be different of exceeding of cadmium, using a global variable for all heavy metals together is masking the possibility to understand what is happening in detail. The same happens if all pollutants found in the outflow of a WWTP are added together in a variable *concentration of pollutants* [86]. In that case, a high level of this variable do not permit to understand if the plant has abnormal operations in the settler (producing too high suspended solids concentrations) or in the bioreactor (too high organic matter concentrations).

Another operation that merits attention in pre-processing is what we can group under the term *knowhow-based feature extraction*, where new variables will appear on the database by combining several original variables in specific ways indicated by the experts. As said before, explicit creation of new variables that approach the decisional criteria used by the experts might provide a powerful set of variables that improve performance of data mining models. This practice often increases also usability, as providing models in terms

of the reasoning variables used by experts, and approaches better the understanding of the expert and their trust on the model. As an example, consider a study on water quality in rivers, where several parameters of the water are measured, among them concentrations on suspended solids, organic matter, biochemical organic matter, ammonium, nitrates, phosphor, also conductivity, temperature, flow, etc. Experts tend to evaluate the water quality based on some standardized indexes that can be computed and added to the dataset as new variables. As an example, the ISQA (used in Spain) is a Simplified Index for Water Quality is $ISQA = T * (COD + SS + DO + C)$ where T is a function of the temperature of the water ($T = 1$ if the temperature $t \leq 20$ ^{o}C, $T = 1 - 0,0125(t - 20)$ otherwise), COD a function of the chemical oxygen demand ($COD = 30 - cod$ if $cod \leq 10$ mg/l, $COD = 0$ if $cod > 60$ mg/l, $COD = 21 - (0,35 - cod)$ otherwise), SS a function of the suspended solids ($SS = 25 - (0,15 * sst)$, if $sst \leq 100$ mg/l, $SS = 0$ if $sst > 250$ mg/l, $SS = 17 - (0,07 * sst)$, otherwise), DO a function of the dissolved oxygen ($DO = 2,5 * do$, if $do < 10$ mg/l, $DO = 25$ otherwise), C a function of the conductivity ($C = 15,4 * (3,6 - logc)$, if $c \leq 4000$ μS/cm, $D = 0$ otherwise) [216] and, as it can be seen, ISQA is a non-linear relationship among these parameters. Introducing it as an additional variable in data mining modeling can improve the performance and interpretability of the model. Another example is to build the atmospheric clearness index as the ratio between two other variables that can be directly measured: the solar radiation and ground level to extra-terrestrial solar radiation.

Creation of binary indicators: evaluating to TRUE or FALSE according to a certain Boolean condition expressed over some subset of observed variables. For example, a variable *Abnormal operation of the bioreactor* in a WWTP, that will evaluate to TRUE if the effluent of the plant has concentrations of organic matter over the thresholds permitted by the law or there was bulking or foaming in the bioreactor. Or *lethal radiation* that will evaluate to TRUE if the radiation is *lethal in some weeks or lethal in some hours or lethal with immediate effect*.

Another type of operation is related to creation of new variables by split or concatenation of the original ones:

- Splitting a variable: It creates a vector of variables X_k by decomposing a variable Y in parts. As an example, the EWC [62] is an international European code for encoding waste, which has a format that reflects the taxonomy in behind. Thus, the semantics of the several groups of digits of the code correspond to the categories of the waste from a more general classification to a more specific. As an example, code 050105 has the following meaning: first two digits 05 indicates a group of sources generating similar kinds of waste, in this case *Wastes from petroleum refining, natural gas purification and pyrolytic treatment of coal*. Third and fourth digits indicate a more specific sector inside the main source. In the example 01 means *wastes from petroleum refining*. Finally, the last two digits give the specific waste. The code 05 corresponds to *oil spills*. Thus, the code could be transformed in three new variables regarding the three levels of waste categories and one of them used in the data mining modeling, according to the level of granularity appropriate for the goals of the analysis. A date can be as well decomposed in three new columns with day, month and year, which can be used for different purposes, like filtering all weekend days (day equal to Saturday or Sunday) to build specific pollution models in big cities for weekends, or entering the month into data mining modeling as a seasonal term. Of course all these operations are strongly linked with a good knowledge of the semantics of the variables and are absolutely domain-dependent. Dedicated parsers are usually required to this purpose.
- Creating a new variable Y by concatenation of two (or more) pre-existent variables X_k, $k \in S$, $S \subseteq (1 : K)$. When X_k are qualitative, Y is in the space of the Cartesian product of all X_k used. From a practical point of view, this corresponds to creating a crossing variable. This is the case, for example, of having a dataset with volcanic eruptions in the rows and, among others, a set of 15 binary variables indicating whereas the eruption generated or not different kinds of products (liquid lava, glassy lava, lava bombs, chunk of lava, rubbly lava, lobes of lava, viscous magma, fragmented rocks, scoria, tephra, ash, stem, pyroclastic density current, gas). Whereas the 15 variables become concatenated, a variable indicating the whole list of products generated in the eruption will produce values like *LiquidLavaLavaBombsFragmentedRocksScoriaAshGas, just liquidLava, or viscousMagmaTephraAshPiroclasticDensityCurrent* and the resulting variable can perform better in classifying the type of eruption than the 15 bi-

nary variables. If the X_k are represented by numerical codes of d digits or less, an efficient form of obtaining Y is by the transformation $Y = \prod_{k\in\{1:card(S)\}} X_(s_k) \times d^k$.

Special attention is required for multivalued variables, introduced in section 3.1. From the informational point of view, the s binary variables, representing the modalities of the multivalued variable, are providing information of a single aspect of the object described in each data row. An example is the variable *technologies available in a city to provide access to water* expressed by means of $s = 3$ binary variables in [12]. However, if the multivalued variable is introduced in the data mining analysis as a set of s independent binary variables, this artificially increases the dimensionality of the data matrix by dedicating s dimensions to a single variable. This often biases the analysis. In the example, it is like if the water access technologies were more important in the description of the cities than other variables, like population density or type of climate, both explicable in a single column each. To avoid this excess of weight of the multivalued variable, some possibilities are available: one is to build the concatenation of all binary components, as explained before. The new variable, in the example, will contain the list of combinations of water access technologies available in each city. This enables also to analyze the kind of combinations really used or not and some analysis on the interactions between water access technologies. Also, some expert-based feature extraction can be done to substitute the multivalued variable by a vector of more powerful variables describing it. In the example, a possibility is building the *number of water access technologies available in the city*, and, for example, an indicator for the existence of rudimentary systems or most modern ones, etc. This clearly depends on the goal of the analysis and must be carefully designed using specific domain-knowledge.

In section 9.2.3 the factorial methods have been presented as a method to reduce data dimensionality. The retained factors are in fact new variables, which correspond to (usually linear) combinations of the original variables and can be used instead of them in the data mining. It is important to ensure that the factors can be properly interpreted, to get meaningful data mining models.

Finally, when serial measurements are obtained, from sensors for example, it is dangerous to introduce the series itself in the analysis as it is, for the same reasons discussed in the multivalued variables case. Imagine a dataset describing some pieces of land, with their surface, characteristics of the soil, climate, proximity to the see, type of culture, agricultural technology used (including type of fertilizers, pesticides, etc.), obtained production, etc., and including a set of 365 variables with the daily mean temperatures along a complete year, another with the minimum daily temperatures, and another with the maximum temperatures. Building data mining models to analyze the relationship between production and the other factors, cannot include directly the 365 temperature variables as they are. In this case, expert-based feature extraction can be used to substitute the temperature series to a subset of meaningful variables describing the series. For example, *mean temperature in the period of growing the vegetable, max temperature in the same period, min temperature in the same period*, some synthetic descriptors or indicators, as for example, *was the temperature below 0 degrees in between November and February?*. In [199] it is shown that reducing the series to a single mean significantly decreases the quality of the data mining models.

12. Conclusions and Challenges for pre-processing in Environmental Data Mining

Environmental processes have some intrinsic complexities that make the data analysis difficult and often the classical data analysis methods do not perform well in environmental applications. KDD is a good support to analyze environmental domains, defining a new paradigm where, apart from the data mining step itself, pre-processing of data and post-processing of results are included in the methodology itself, and the prior expert knowledge is also considered to discover new useful results.

The main focus of this paper is to provide a general overview of the pre-processing techniques regarding its use within Environmental Systems providing guidelines to the end user from a practical perspective.

Few works are found on pre-processing in the literature, but none of them is specific for environmental data, and none provides methodological recommendations for the pre-processing process as a whole.

This paper proposes a general pre-processing methodology. Usually, the pre-processing tasks are a mixture of different techniques that have not been analyzed and structured. Most data scientists use some pre-processing techniques taken from an unstructured pool of techniques, but without a reasonable and right ordering of the pre-processing steps. Sometimes, for-

getting to check some data issues gives problems during the data mining step and requires costly backtracking to the pre-processing step thus forcing to rebuild the whole analysis several times.

In this paper, the most relevant steps involved in pre-processing with environmental data are identified and reviewed, and a general framework and methodology are proposed for dealing with the pre-processing from an applied perspective in the Data Mining process, considering that pre-processing is highly related with data quality issues and it is of paramount importance to avoid propagation of errors within the data mining models and consequent decision-making based on them. As it can be seen, the pre-processing step is strongly multidisciplinary, and requires integrated tools from many different origins, including basic Statistics, multivariate analysis, machine learning, information management, visualization and GIS, among others.

The most relevant aspects of each issue in pre-processing have been reviewed and structured in a way that the reader can effectively use them to make decisions on how to pre-process a real dataset.

As a result of the state-of-the-art review performed in this paper, and the efforts in conceptualization of the different sections, like the one about outliers or missing data, some guidelines for pre-processing tasks are synthetized here, as a first approach to provide a global guide to perform correct KDD processes as a whole.

The main contribution of this paper is to provide a deep analysis of the steps composing the pre-processing task. It can be said that, in general, most of the pre-processing techniques are devoted to: visualization of the data, detection of imperfections or verification of the accomplishment of some properties of the dataset, transformation of data for correcting imperfections or achieving certain properties for better analysis. This include a number of different operations that make sense in different scenarios suitable in environmental data mining, which often are related with the nature of data itself, but also with the goals of the analysis and the related domain knowledge.

Finally, as a last contribution of the paper, the enormous effort in reviewing the work done in the literature in pre-processing, allowed to identify the hot issues and challenging aspects in and of the interdisciplinary field of environmental DM in coming years. The achievement of some specific goals, non-well established yet, at the level of both end-users and developers or researchers would increase the benefits of a proper pre-processing process and improve the quality of the further data mining modeling step.

At the level of end-users the following issues are important:

- Get acquainted with available methods and technical assumptions of pre-processing methods, even before data collection, if possible.
- Try to clarify the relevant questions to be answered by the KDD process to orient an adequate Data Mining. Do not lose perspective of final usefulness.
- Consider all data available which might be relevant for the KDD goal, independently of their original format (images, text, etc.) and use data fusion to get the initial data matrix. Try to minimize the missing data collected by properly designing the structural missing data management. Consider the collection of qualitative variables without internal tedious numerical encodings when possible.
- Collaboration between the expert and the data miner becomes crucial in most of the decisions made during the pre-processing step.
- Be careful and rigorous on meta-data collection. It is important to store it in an available format and to clearly understand it for proper analysis decisions, and proper data and results interpretation.
- Reserve a relevant time in your projects for data pre-processing and devote the required time to it, by performing it carefully enough to guarantee the quality of the data mining models results.
- Carefully verify that missing data is properly imputed in the pre-processing tool, and transfer to the system as much meta-information as possible.
- Select relevant data for the working data matrix before starting the pre-processing analysis itself. To this purpose, KDD goals, and priori specific domain knowledge has to be taken into account. In case of doubt, it is preferable to keep a variable and make decisions at modeling step.
- Use visualization to decide which pre-processing aspects need to be treated.
- Try to change your data as little as possible; make the minimum pre-processing operations required to improve data quality and guarantee better interpretability of the final data mining results. Try to avoid, as much as possible, transformations that generate variables without a meaning that can make data mining models understandable. Instead of adapting data to the data mining method technical requirements, try to find a method that fits well with the intrinsic structure of your dataset.

- However, spend all required time in terminology standardization.
- Pay special attention to imbalanced datasets, compositional data, and multivalued variables. Consider the available possibilities to deal with data noise and uncertainty.
- Whenever possible, include in the dataset synthetic variables approaching the decisional indicators used by experts to reason.
- Use both instance selection and/or feature selection to detect redundant or irrelevant variables or instances, reducing the dimension of the dataset by keeping the quality of the data mined models.
- When appropriate, use feature weighting to estimate the relevance of the variables of the data matrix, and make further decisions accordingly.
- Try resampling to check stability of results and to process too big datasets with a particular limited method. Use the classical statistical sampling principles, traditionally used to get samples from real populations, to sample over a big dataset.
- Ensure that conclusions are well framed into the subpopulations determined by the pre-processing step really analyzed.
- Always report in a transparent way all pre-processing assumptions done, and all pre-processing decisions made, particularly those related to elimination of some rows or variables, or modifications owing to error correction.

At the level of researchers, the following issues merit serious attention in the near future, in order to advance the state of the art in pre-processing in the challenging directions that will make it easier, more powerful and reliable in real applications:

- The development of meta-data management support tools that permit to standardize the meta-data representation models, and the way they can be used in the pre-processing is of main interest, since there is a lack of standards and all the responsibility relies nowadays to the end user, this increasing the risks of disregarding important details. Also, to develop mechanisms to introduce these standards into the data mining system is relevant, since currently only basic issues like declaring qualitative variables is available in a reduced number of data mining software tools. This will permit automation of some decisions, increasing efficiency of the pre-processing step.
- Elaborate protocols to enhance data sharing and reuse, particularly oriented to build data matrices to be mined, by combining different data sources from different formats. In particular, the potential of web/text mining needs to be further explored.
- There is a great need of developing methods for data fusion able to deal simultaneously with data coming from different sources, with different natures, scales, granularities and formats. Particular attention on getting data from smart sensors, images or web documents is suggested.
- Particular attention is required to explore the challenges for properly pre-processing incremental data and also the specific considerations required for pre-processing in big data.
- More research is still required to better understand the complexity of pre-processing tasks in the context of KDD processes. The methodological proposal presented in this paper is a first attempt to systematize the pre-processing process, which can be used to guide the flow of pre-processing operation in data mining software tools. However, additional research is required to stabilize a complete and exhaustive methodology for pre-processing in environmental systems.
- As it has been extensively illustrated along the paper, it is very relevant to take into account the expertise of the end-user as well as the specific domain knowledge to guide many decisions and operations required in the pre-processing step. For this reason, a methodology that considers the involvement of both prior domain knowledge and interaction with the expert is essential for the progress in the pre-processing field.
- As many Data Mining softwares provide GUIs to design the workflow for a complete KDD process, collect KDD experiences in a workflow library and develop Data Mining strategies to mine the workflows themselves for improving the pre-processing, data mining and post-processing recommenders under an evidence-based approach are much needed.
- Development of standard procedures (benchmarks) for experimental testing and validation of new pre-processing tools.
- There is a lack of tools for explicit representation and handling of discovered knowledge for greater understandability. Development of tools to bridge the gap between modeling and effective decision-making [84] [83] [85] [80] can be enormously useful at all levels of data mining models, to guarantee the effective impact of the data mining step at an environmental decision-making level. As in

ESs it is usual to meet decision-makers with no technical skills, development of this kind of tools is more needed than in other more technical fields.
- Finally, there is a need of moving towards the development of integrated KDD-systems, with sufficient intelligence to pursue integral approaches, covering the whole KDD process, from problem formulation to discovered knowledge production, including interpretation of results. This requires KDD systems including data fusion, meta-data management, data cleaning and the whole range of pre-processing techniques, taking into account prior expert knowledge, data mining method recommendation, and post-processing, and involving the expert in both the problem formulation, preferences on kind of results and interpretation.

Regarding the support provided to pre-processing by the various available KDD software packages, we can say that, although they often include some available tools to perform some pre-processing steps, current packages seem still to be far from being such a complete intelligent system that could support both a well systematized pre-processing and an integral KDD, as stated above. The complete pre-processing process is normally designed from scratch for every application and only lists of unconnected techniques for pre-processing are provided in most softwares. We strongly believe that efforts should be made in the near future in this direction to provide a general framework for pre-processing the data in a reliable way, and any advance on the state of the art in this direction will have benefits not only in the quality of results in data mining projects, but also in the time required for them.

References

[1] C. C. Aggarwal. *Supervised Outlier Detection*. Arfken and Weber, 2012.

[2] C. C. Aggarwal. Outlier analysis. In *Data Mining*, pages 237–263. Springer, 2015.

[3] D. W. Aha. Feature weighting for lazy learning algorithms. In *Feature Extraction, Construction and Selection*, pages 13–32. Springer, 1998.

[4] D. W. Aha, D. Kibler, and M. K. Albert. Instance-based learning algorithms. *Machine learning*, 6(1):37–66, 1991.

[5] J. Aitchison. Principles of compositional data analysis. *Lecture Notes-Monograph Series*, pages 73–81, 1994.

[6] A. Alexandridis, P. Patrinos, H. Sarimveis, and G. Tsekouras. A two-stage evolutionary algorithm for variable selection in the development of rbf neural network models. *Chemometrics and Intelligent Laboratory Systems*, 75(2):149–162, 2005.

[7] P. D. Allison. Multiple imputation for missing data: A cautionary tale, 1999.

[8] P. D. Allison. *Missing data*, volume 136. Sage publications, 2001.

[9] G. Andrienko and A. Andrienko. Research on visual analysis of spatio-temporal data at fraunhofer ais: An overview of history and functionality of commongis. In *Proceedings of the Knowledge-Based Services for the Public Services Symposium, Workshop III: Knowledge Discovery for Environmental Managment*, pages 26–31, 2004.

[10] A. Arauzo-Azofra, J. L. Aznarte, and J. M. Benítez. Empirical study of feature selection methods based on individual feature evaluation for classification problems. *Expert Systems with Applications*, 38(7):8170–8177, 2011.

[11] A. Arauzo-Azofra, J. M. Benitez, and J. L. Castro. Consistency measures for feature selection. *Journal of Intelligent Information Systems*, 30(3):273–292, 2008.

[12] I. Arregui, A. Balaguer, and et al. Learning on the relationships between respiratory disease and the use of traditional stoves in bangladesh households. In iEMSs, editor, *Procs iEMSs'2016*, volume 3, 2016.

[13] I. N. Athanasiadis, V. G. Kaburlasos, P. A. Mitkas, and V. Petridis. Applying machine learning techniques on air quality data for real-time decision support. In *First international NAISO symposium on information technologies in environmental engineering (ITEE-2003), Gdansk, Poland*. Citeseer, 2003.

[14] J. Atserias and et al. Syntactic and semantic services in an open-source nlp library. In *Procs LREC*, volume 6, 2006.

[15] M. J. Azur, E. A. Stuart, C. Frangakis, and P. J. Leaf. Multiple imputation by chained equations: what is it and how does it work? *International journal of methods in psychiatric research*, 20(1):40–49, 2011.

[16] A. Bargiela and W. Pedrycz. *Granular computing: an introduction*, volume 717. Springer Science & Business Media, 2012.

[17] V. Barnett, V. Barnett, and T. Lewis. *Outliers in statistical data*. Wiley, 1978.

[18] G. E. Batista, M. C. Monard, et al. A study of k-nearest neighbour as an imputation method. *HIS*, 87(251-260):48, 2002.

[19] G. E. Batista, R. C. Prati, and M. C. Monard. A study of the behavior of several methods for balancing machine learning training data. *ACM Sigkdd Explorations Newsletter*, 6(1):20–29, 2004.

[20] B. Bazartseren, G. Hildebrandt, and K.-P. Holz. Short-term water level prediction using neural networks and neuro-fuzzy approach. *Neurocomputing*, 55(3):439–450, 2003.

[21] R. Bekkerman, R. El-Yaniv, N. Tishby, and Y. Winter. Distributional word clusters vs. words for text categorization. *Journal of Machine Learning Research*, 3(Mar):1183–1208, 2003.

[22] D. A. Belsley, E. Kuh, and R. E. Welsch. *Regression diagnostics: Identifying influential data and sources of collinearity*, volume 571. John Wiley & Sons, 2005.

[23] R. Berg. Hurricane ike (al092008) 1–14 september 2008. *National Hurricane Center Tropical Cyclone Rep*, 2009.

[24] J. C. Bezdek and L. I. Kuncheva. Nearest prototype classifier designs: An experimental study. *International Journal of Intelligent Systems*, 16(12):1445–1473, 2001.

[25] A. Bifet, G. Holmes, B. Pfahringer, R. Kirkby, and R. Gavaldà. New ensemble methods for evolving data streams. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 139–148. ACM, 2009.

[26] R. Bischof, L. E. Loe, E. L. Meisingset, B. Zimmermann, B. Van Moorter, and A. Mysterud. A migratory northern ungulate in the pursuit of spring: jumping or surfing the green wave? *The American Naturalist*, 180(4):407–424, 2012.

[27] J. D. Blower, K. Haines, A. Santokhee, and C. L. Liu. Godiva2: interactive visualization of environmental data on the web. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 367(1890):1035–1039, 2009.

[28] V. Bolón-Canedo, N. Sánchez-Maroño, and A. Alonso-Betanzos. A review of feature selection methods on synthetic data. *Knowledge and information systems*, 34(3):483–519, 2013.

[29] G. E. Box and D. R. Cox. An analysis of transformations. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 211–252, 1964.

[30] G. Bretana. Admiralty manual of navigation. *Volume*, 1:227, 1987.

[31] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander. Lof: identifying density-based local outliers. In *ACM sigmod record*, volume 29 (2), pages 93–104. ACM, 2000.

[32] H. Brighton and C. Mellish. Advances in instance selection for instance-based learning algorithms. *Data mining and knowledge discovery*, 6(2):153–172, 2002.

[33] G. Brown, A. Pocock, M.-J. Zhao, and M. Luján. Conditional likelihood maximisation: a unifying framework for information theoretic feature selection. *Journal of Machine Learning Research*, 13(Jan):27–66, 2012.

[34] I. Bruha and A. Famili. Postprocessing in machine learning and data mining. *ACM SIGKDD Explorations Newsletter*, 2(2):110–114, 2000.

[35] A. Butler and C. Glasbey. A latent gaussian model for compositional data with zeros. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 57(5):505–520, 2008.

[36] Y. Caises, A. González, E. Leyva, and R. Pérez. Scis: combining instance selection methods to increase their effectiveness over a wide range of domains. In *International Conference on Intelligent Data Engineering and Automated Learning*, pages 17–24. Springer, 2009.

[37] J. R. Cano, F. Herrera, and M. Lozano. Using evolutionary algorithms as instance selection for data reduction in kdd: an experimental study. *IEEE Transactions on Evolutionary Computation*, 7(6):561–575, 2003.

[38] R. Caruana and V. R. d. Sa. Benefitting from the variables that variable selection discards. *Journal of machine learning research*, 3(Mar):1245–1264, 2003.

[39] V. Cerveron and F. J. Ferri. Another move toward the minimum consistent subset: a tabu search approach to the condensed nearest neighbor rule. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 31(3):408–413, 2001.

[40] J. M. Chambers. *Graphical methods for data analysis*. Wadsworth, 1983.

[41] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 41(3):15, 2009.

[42] G. Chandrashekar and F. Sahin. A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):16–28, 2014.

[43] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.

[44] C.-H. Chou, B.-H. Kuo, and F. Chang. The generalized condensed nearest neighbor rule as a data reduction method. In *18th International Conference on Pattern Recognition (ICPR'06)*, volume 2, pages 556–559. IEEE, 2006.

[45] K. Cios, W. Pedrycz, R. W. Swiniarski, and L. A. Kurgan. *Data Mining: A Knowledge Discovery Approach*. Springer, 2007.

[46] N. Collier. Uncovering text mining: A survey of current work on web-based epidemic intelligence. *Global public health*, 7(7):731–749, 2012.

[47] P. Cortez and M. J. Embrechts. Using sensitivity analysis and visualization techniques to open black box data mining models. *Information Sciences*, 225:1–17, 2013.

[48] R. H. Creecy, B. M. Masand, S. J. Smith, and D. L. Waltz. Trading mips and memory for knowledge engineering. *Communications of the ACM*, 35(8):48–64, 1992.

[49] H.-J. Dai, C.-H. Huang, J. Y.-W. Lin, P.-H. Chou, R. T.-H. Tsai, and W.-L. Hsu. A survey of state of the art biomedical text mining techniques for semantic analysis. In *Sensor Networks, Ubiquitous and Trustworthy Computing, 2008. SUTC'08. IEEE International Conference on*, pages 410–417. IEEE, 2008.

[50] S. K. Das. *High-level data fusion*. Artech House, 2008.

[51] M. Dash and H. Liu. Handling large unsupervised data via dimensionality reduction. In *1999 ACM SIGMOD workshop on research issues in data mining and knowledge discovery*, 1999.

[52] M. Dash and H. Liu. Consistency-based search in feature selection. *Artificial intelligence*, 151(1):155–176, 2003.

[53] J. L. Davidson and J. Jalan. Feature selection for steganalysis using the mahalanobis distance. In *IS&T/SPIE Electronic Imaging*, pages 754104–754104. International Society for Optics and Photonics, 2010.

[54] J. Derrac, S. García, and F. Herrera. A survey on evolutionary instance selection and generation. *Int'l J. Applied Metaheuristic Computing*, 1(1):60–92, 2010.

[55] P. A. Devijver and J. Kittler. On the edited nearest neighbor rule. In *Proc. 5th Int. Conf. on Pattern Recognition*, pages 72–80, 1980.

[56] T. G. Dietterich. Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pages 1–15. Springer, 2000.

[57] Y. Dong and C.-Y. J. Peng. Principled missing data methods for researchers. *SpringerPlus*, 2(1):1, 2013.

[58] B. Duval, J.-K. Hao, and J. C. Hernandez Hernandez. A memetic algorithm for gene selection and molecular classification of cancer. In *Proceedings of the 11th Annual conference on Genetic and evolutionary computation*, pages 201–208. ACM, 2009.

[59] J. G. Dy and C. E. Brodley. Feature selection for unsupervised learning. *Journal of machine learning research*, 5(Aug):845–889, 2004.

[60] M. Edwards, N. Ferrand, F. Goreaud, and S. Huet. The relevance of aggregating a water consumption model cannot be disconnected from the choice of information available on the resource. *Simulation Modelling Practice and Theory*, 13(4):287–307, 2005.

[61] EEML. http://www.eeml.org, 2008.

[62] EPA. *European Waste Catalogue and Hazardous Waste list*. European Environmental Protection Agengy, 2002.

[63] F. Famili, W.-M. Shen, R. Weber, and E. Simoudis. Data preprocessing and intelligent data analysis. *International Journal on Intelligent Data Analysis*, 1(1), 1997.

[64] D. W. Fanning. *IDL programming techniques*. Fanning software consulting, 2000.

[65] A. Farhangfar, L. Kurgan, and J. Dy. Impact of imputation of missing values on classification error for discrete data. *Pattern Recognition*, 41(12):3692–3705, 2008.

[66] R. M. Faye, S. Sawadogo, C. Lishou, and F. Mora-Camino. Long-term fuzzy management of water resource systems. *Applied Mathematics and Computation*, 137(2):459–475, 2003.

[67] U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy. *Advances in knowledge discovery and data mining*, volume 21. AAAI press Menlo Park, 1996.

[68] A. Fernández, S. García, M. J. del Jesus, and F. Herrera. A study of the behaviour of linguistic fuzzy rule based classification systems in the framework of imbalanced data-sets. *Fuzzy Sets and Systems*, 159(18):2378–2398, 2008.

[69] W. Förstner. Image preprocessing for feature extraction in digital intensity, color and range images. In *Geomatic Method for the Analysis of Data in the Earth Sciences*, pages 165–189. Springer, 2000.

[70] P. G. Foschi, D. Kolippakkam, H. Liu, and A. Mandvikar. Feature extraction for image mining. In *Multimedia Information Systems*, pages 103–109, 2002.

[71] E. Frank and I. H. Witten. Making better use of global discretization. In *Proc. of the Sixteenth International Conference on Machine Learning*, 1999.

[72] H. Gao, G. Barbier, R. Goolsby, and D. Zeng. Harnessing the crowdsourcing power of social media for disaster relief. Technical report, DTIC Document, 2011.

[73] U. Garain. Prototype reduction using an artificial immune model. *Pattern analysis and applications*, 11(3-4):353–363, 2008.

[74] N. GarcÃa-Pedrajas. Evolutionary computation for training set selection. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(6):512–523, 2011.

[75] M. Garcia, E. López, V. Kumar, and A. Valls. A multicriteria fuzzy decision system to sort contaminated soils. In *International Conference on Modeling Decisions for Artificial Intelligence*, pages 105–116. Springer, 2006.

[76] S. García, J. R. Cano, and F. Herrera. A memetic algorithm for evolutionary prototype selection: A scaling up approach. *Pattern Recognition*, 41(8):2693–2709, 2008.

[77] S. Garcia, J. Derrac, J. Cano, and F. Herrera. Prototype selection for nearest neighbor classification: Taxonomy and empirical study. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(3):417–435, 2012.

[78] M. H. Gerardi. *Wastewater bacteria*, volume 5. John Wiley & Sons, 2006.

[79] K. Gibert. Mixed intelligent-multivariate missing imputation. *International Journal of Computer Mathematics*, 91(1):85–96, 2014.

[80] K. Gibert and D. Conti. atlp: A color-based model of uncertainty to evaluate the risk of decisions based on prototypes. *AI Communications*, 28(1):113–126, 2015.

[81] K. Gibert and D. Conti. On the understanding of profiles by means of post-processing techniques: an application to financial assets. *International Journal of Computer Mathematics*, 93(5):807–820, 2016.

[82] K. Gibert and U. Cortés. Clustering based on rules and knowledge discovery in ill-structured domains. *Revista Computación y Sistemas*, 1(4):213–227, 1998.

[83] K. Gibert, A. García-Rudolph, and G. Rodríguez-Silva. The role of kdd support-interpretation tools in the conceptualization of medical profiles: An application to neurorehabilitation. *Acta Informatica Medica*, 16(4):178, 2008.

[84] K. Gibert, J. Izquierdo, G. Holmes, I. Athanasiadis, J. Comas, and M. Sànchez-Marrè. On the role of pre and post-processing in environmental data mining. In *Proceedings of iEMSs 2008 International Congress on Environmental Modeling and Software*, pages 1937–1958. iEMSs, 2008.

[85] K. Gibert, G. Rodríguez-Silva, and R. Annicchiarico. Post-processing: Bridging the gap between modelling and effective decision-support. the profile assessment grid in human behaviour. *Mathematical and Computer Modelling*, 57(7):1633–1639, 2013.

[86] K. Gibert, G. Rodríguez-Silva, and I. Rodríguez-Roda. Knowledge discovery with clustering based on rules by states: A water treatment application. *Environmental Modelling & Software*, 25(6):712–723, 2010.

[87] K. Gibert and M. Sànchez-Marrè. Improving ontological knowledge with reinforcement in recommending the data mining method for real problems. In *Procs of CAEPIA 2015 (TAMIDA)*, pages 769–778. CEDI, 2015.

[88] K. Gibert, M. SÃ¨nchez-MarrÃ¨, and V. Codina. Choosing the right data mining technique: classification of methods and intelligent recommendation. In *Proceedings of iEMSs 2010 International Congress on Environmental Modeling and Software*, pages 2448–2453. iEMSs, 2010.

[89] K. Gibert and Z. Sonicki. Classification based on rules and medical research. *Journal of Applied Stochastic Models and Data Analysis, formerly JAMSDA*, 15(3):319–24, 1999.

[90] K. Gibert, J. Spate, M. Sànchez-Marrè, I. N. Athanasiadis, and J. Comas. Chapter twelve data mining for environmental systems. *Developments in Integrated Environmental Assessment*, 3:205–228, 2008.

[91] E. Golobardes, X. Llora, J. M. Garrell, D. Vernet, and J. Bacardit. Genetic classifier system as a heuristic weighting method for a case-based classifier system. *Butlletı de l'Associació Catalana d'Intel.ligencia Artificial*, 22:132–141, 2000.

[92] P. I. Good and P. Good. *Resampling methods: A practical guide to data analysis*. Springer Science & Business Media, 2013.

[93] J. W. Graham. Missing data analysis: Making it work in the real world. *Annual review of psychology*, 60:549–576, 2009.

[94] J. W. Graham, P. E. Cumsille, and E. Elek-Fisk. Methods for handling missing data. *Handbook of psychology*, 2003.

[95] J. W. Grzymala-Busse, L. K. Goodwin, and X. Zhang. Increasing sensitivity of preterm birth by changing rule strengths. *Pattern Recognition Letters*, 24(6):903–910, 2003.

[96] H. Guo and H. L. Viktor. Learning from imbalanced data sets with boosting and data generation: the databoost-im approach. *ACM SIGKDD Explorations Newsletter*, 6(1):30–39, 2004.

[97] J. F. Hair. *Multivariate data analysis*. Kennesaw State University, 2009.

[98] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.

[99] M. Hall, I. Witten, and E. Frank. Data mining: Practical machine learning tools and techniques. *Kaufmann, Burlington*, 2011.

[100] M. A. Hall. Correlation-based feature selection for discrete and numeric class machine learning. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 359–366. Morgan Kaufmann Publishers Inc., 2000.

[101] M. A. Hall and L. A. Smith. *Practical feature subset selection for machine learning*. Springer, 1998.

[102] J. Han and M. Kamber. *Data mining: Concepts and Techniques*. Morgan Kaufmann, 2006.

[103] P. Hanrahan. Tableau software white paper-visual thinking for business intelligence. *Tableau Software, Seattle, WA*, 2003.

[104] P. Hart. The condensed nearest neighbor rule. In *IEEE Trans. Inform. Theory (Corresp.)*, volume IT-14, pages 515–516, 1968.

[105] M. Herrera, J. Izquierdo, R. Pérez-García, and I. Montalvo. Multi-agent adaptive boosting on semi-supervised water supply clusters. *Advances in Engineering Software*, 50:131–136, 2012.

[106] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

[107] F. Honghai, C. Guoshun, Y. Cheng, Y. Bingru, and C. Yumei. A svm regression based approach to filling in missing values. In *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*, pages 581–587. Springer, 2005.

[108] N. Howe and C. Cardie. Examining locally varying weights for nearest neighbor algorithms. In *International Conference on Case-Based Reasoning*, pages 455–466. Springer, 1997.

[109] N. Howe and C. Cardie. Feature subset selection and order identification for unsupervised learning. In *In proceedings of 17th International Conference on Machine Learning*. Morgan Kaufmann, 2000.

[110] M. Y. Huh. Incremental subset selection for complex data. *Proceedings, COMPSTAT2006, Rome, Italy*, 2006.

[111] L. Ingsrisawang and D. Potawee. Multiple imputation for missing data in repeated measurements using mcmc and copulas. In *Proceedings of the International MultiConference of Engineers and computer scients (IMECs), Hong kong*, 2012.

[112] N. Ishii and Y. Wang. Learning feature weights for similarity using genetic algorithms. In *Intelligence and Systems, 1998. Proceedings., IEEE International Joint Symposia on*, pages 27–33. IEEE, 1998.

[113] T. Ishiwata, M. Muroi, T. Harada, H. Nakajima, and I. Ogasawara. Establishing an environmental data platform for promoting coastal zone environmental management. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XXXVII:25–30, 2008.

[114] ISO/TC211. Iso19115 geographic information - metadata, 2003.

[115] ISO/TC211. Iso19136 geographic information - geomatics, 2007.

[116] J. Izquierdo, P. López, F. Martínez, and R. Pérez. Fault detection in water supply systems using hybrid (theory and data-driven) modelling. *Mathematical and Computer Modelling*, 46(3):341–350, 2007.

[117] J. Izquierdo, R. Pérez, P. López, and P. Iglesias. Neural identification of fuzzy anomalies in pressurized water systems. In *Proceedings of the 3rd Biennial meeting of the International Environmental Modeling and Software Society, Burlington, VT, USA*. iEMSs, 2006.

[118] N. Jankowski and M. Grochowski. Comparison of instances seletion algorithms i. algorithms survey. In *International conference on artificial intelligence and soft computing*, pages 598–603. Springer, 2004.

[119] N. Japkowicz and S. Stephen. The class imbalance problem: A systematic study. *Intelligent data analysis*, 6(5):429–449, 2002.

[120] K. Javed, H. A. Babri, and M. Saeed. Feature selection based on class-dependent densities for high-dimensional binary data. *IEEE Transactions on Knowledge and Data Engineering*, 24(3):465–477, 2012.

[121] A. Jiménez and A. Pérez-Foguet. Improving water access indicators in developing countries: a proposal using water point mapping methodology. *Water Science and Technology: Water Supply*, 8(3):279–287, 2008.

[122] G. H. John and P. Langley. Static versus dynamic sampling for data mining. In *KDD*, volume 96, pages 367–370, 1996.

[123] D. Jouan-Rimbaud, D.-L. Massart, R. Leardi, and O. E. De Noord. Genetic algorithms as a tool for wavelength selection in multivariate calibration. *Analytical Chemistry*, 67(23):4295–4301, 1995.

[124] C.-F. Juang. Temporal problems solved by dynamic fuzzy network based on genetic algorithm with variable-length chromosomes. *Fuzzy Sets and Systems*, 142(2):199–219, 2004.

[125] M. Juhola and J. Laurikkala. Missing values: how many can they be to preserve classification reliability? *Artificial Intelligence Review*, 40(3):231–245, 2013.

[126] KDNuggets. Data preparation. website, 2003.

[127] H. Kim, G. H. Golub, and H. Park. Missing value estimation for dna microarray gene expression data: local least squares imputation. *Bioinformatics*, 21(2):187–198, 2005.

[128] K. Kira and L. A. Rendell. A practical approach to feature selection. In *Proceedings of the ninth international workshop on Machine learning*, pages 249–256, 1992.

[129] R. Kohavi et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14(2), pages 1137–1145, 1995.

[130] R. Kohavi and G. H. John. Wrappers for feature subset selection. *Artificial intelligence*, 97(1):273–324, 1997.

[131] R. Kohavi and G. H. John. The wrapper approach. In *Feature extraction, construction and selection*, pages 33–50. Springer, 1998.

[132] R. Kohavi, P. Langley, and Y. Yun. The utility of feature weighting in nearest-neighbor algorithms. In *Proceedings of the Ninth European Conference on Machine Learning*, pages 85–92, 1997.

[133] D. Koller and M. Sahami. Toward optimal feature selection. In *In 13th International Conference on Machine Learning*, 1995.

[134] S. Konishi. *Introduction to Multivariate Analysis: Linear and Nonlinear Modeling*. CRC Press, 2014.

[135] I. Kononenko. Estimating attributes: analysis and extensions of relief. In *European conference on machine learning*, pages 171–182. Springer, 1994.

[136] M. Kubat, R. C. Holte, and S. Matwin. Machine learning for the detection of oil spills in satellite radar images. *Machine learning*, 30(2-3):195–215, 1998.

[137] L. Kuncheva, J. Wrench, L. C. Jain, and A. Al-Zaidan. A fuzzy model of heavy metal loadings in liverpool bay. *Environmental Modelling & Software*, 15(2):161–167, 2000.

[138] L. I. Kuncheva. Fitness functions in editing k-nn reference set by genetic algorithms. *Pattern Recognition*, 30(6):1041–1049, 1997.

[139] M. H. Kutner, C. J. Nachtsheim, J. Neter, W. Li, et al. *Applied linear statistical models*, volume 103. McGraw-Hill Irwin New York, 2005.

[140] N. Kwak and C.-H. Choi. Input feature selection for classification problems. *IEEE Transactions on Neural Networks*, 13(1):143–159, 2002.

[141] P. Langley and W. Iba. Average-case analysis of a nearest neighbor algorithm. In *IJCAI*, pages 889–894. Citeseer, 1993.

[142] C. Lazar, J. Taminau, S. Meganck, D. Steenhoff, A. Coletta, C. Molter, V. de Schaetzen, R. Duque, H. Bersini, and A. Nowe. A survey on filter techniques for feature selection in gene expression microarray analysis. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 9(4):1106–1119, 2012.

[143] L. Lebart. Correspondence analysis. In *Data Science, Classification, and Related Methods: Proceedings of the Fifth Conference of the International Federation of Classification Societies (IFCS-96), Kobe, Japan, March 27–30, 1996*, page 423. Springer Science & Business Media, 2013.

[144] K. Leung and C. Leckie. Unsupervised anomaly detection in network intrusion detection using clusters. In *Proceedings of the Twenty-eighth Australasian conference on Computer Science-Volume 38*, pages 333–342. Australian Computer Society, Inc., 2005.

[145] R. J. Little and D. B. Rubin. *Statistical analysis with missing data*. John Wiley & Sons, 2014.

[146] H. Liu and H. Motoda. Data reduction via instance selection. In *Instance selection and construction for data mining*, pages 3–20. Springer, 2001.

[147] H. Liu and H. Motoda. *Computational methods of feature selection*. CRC Press, 2007.

[148] H. Liu and H. Motoda. *Feature selection for knowledge discovery and data mining*, volume 454. Springer Science & Business Media, 2012.

[149] H. Liu, J. Sun, L. Liu, and H. Zhang. Feature selection with dynamic mutual information. *Pattern Recognition*, 42(7):1330–1339, 2009.

[150] H. Liu and L. Yu. Toward integrating feature selection algorithms for classification and clustering. *IEEE Transactions on knowledge and data engineering*, 17(4):491–502, 2005.

[151] J. Luengo, S. García, and F. Herrera. On the choice of the best imputation methods for missing values considering three groups of classification methods. *Knowledge and information systems*, 32(1):77–108, 2012.

[152] A. Lumini and L. Nanni. A clustering method for automatic biometric template selection. *Pattern Recognition*, 39(3):495–497, 2006.

[153] P. Marmonier, G. Archambaud, N. Belaidi, N. Bougon, P. Breil, E. Chauvet, C. Claret, J. Cornut, T. Datry, M.-J. Dole-Olivier, et al. The role of organisms in hyporheic processes: gaps in current knowledge, needs for future research and applications. In *Annales de Limnologie-International Journal of Limnology*, volume 48(3), pages 253–266. EDP Sciences, 2012.

[154] J. B. Martínez-Rodríguez, I. Montalvo, J. Izquierdo, and R. Pérez-García. Reliability and tolerance comparison in water supply networks. *Water resources management*, 25(5):1437–1448, 2011.

[155] I. Mierswa, M. Wurst, R. Klinkenberg, M. Scholz, and T. Euler. Yale: Rapid prototyping for complex data mining tasks. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 935–940. ACM, 2006.

[156] H. J. Miller and J. Han. *Geographic data mining and knowledge discovery*. CRC Press, 2009.

[157] M. Minelli, M. Chambers, and A. Dhiraj. *Big data, big analytics: emerging business intelligence and analytic trends for today's businesses*. John Wiley & Sons, 2012.

[158] T. M. Mitchell. Generalization as search. *Artificial intelligence*, 18(2):203–226, 1982.

[159] P. Mitra, C. Murthy, and S. K. Pal. Unsupervised feature selection using feature similarity. *IEEE transactions on pattern analysis and machine intelligence*, 24(3):301–312, 2002.

[160] T. Mohri and H. Tanaka. An optimal weighting criterion of case indexing for both numeric and symbolic attributes. In *AAAI-94 Workshop Program: Case-Based Reasoning, Working Notes*, pages 123–127, 1994.

[161] L. C. Molina, L. Belanche, and À. Nebot. Feature selection algorithms: a survey and experimental evaluation. In *Data Mining, 2002. ICDM 2003. Proceedings. 2002 IEEE International Conference on*, pages 306–313. IEEE, 2002.

[162] R. A. Mollineda, F. J. Ferri, and E. Vidal. An efficient prototype merging strategy for the condensed 1-nn rule through class-conditional hierarchical clustering. *Pattern Recognition*, 35(12):2771–2782, 2002.

[163] D. S. Moore, G. P. McCabe, and M. J. Evans. *Introduction to the practice of statistics Minitab manual and Minitab version 14*. WH Freeman & Co., 2005.

[164] A. Murakami and T. Nasukawa. Tweeting about the tsunami?: mining twitter for information on the tohoku earthquake and tsunami. In *Proceedings of the 21st International Conference on World Wide Web*, pages 709–710. ACM, 2012.

[165] S. Nakariyakul and D. P. Casasent. An improvement on floating search algorithms for feature subset selection. *Pattern Recognition*, 42(9):1932–1940, 2009.

[166] B. L. Narayan, C. Murthy, and S. K. Pal. Maxdiff kd-trees for data condensation. *Pattern recognition letters*, 27(3):187–200, 2006.

[167] P. M. Narendra and K. Fukunaga. A branch and bound algorithm for feature subset selection. *IEEE Transactions on Computers*, 100(9):917–922, 1977.

[168] D. F. Nettleton, A. Orriols-Puig, and A. Fornells. A study of the effect of different types of noise on the precision of supervised learning techniques. *Artificial intelligence review*, 33(4):275–306, 2010.

[169] M. Nixon. *Feature extraction & image processing*. Academic Press, 2008.

[170] H. Núñez. *Feature weighting in plain case-based reasoning*. PhD thesis, Ph. D. Thesis. Universitat Politècnica de Catalunya, 2004.

[171] H. Núñez and M. Sànchez-Marrè. Instance-based learning techniques of unsupervised feature weighting do not perform so badly! In *ECAI*, volume 16, pages 102–106, 2004.

[172] H. Núñez, M. Sànchez-Marrè, and U. Cortés. Improving similarity assessment with entropy-based local weighting. In *International Conference on Case-Based Reasoning*, pages 377–391. Springer, 2003.

[173] S.-K. Oh and W. Pedrycz. Self-organizing polynomial neural networks based on polynomial and fuzzy polynomial neurons: analysis and design. *Fuzzy sets and systems*, 142(2):163–198, 2004.

[174] J. A. Olvera-López, J. A. Carrasco-Ochoa, and J. F. Martínez-Trinidad. Object selection based on clustering and border objects. In *Computer Recognition Systems 2*, pages 27–34. Springer, 2007.

[175] J. A. Olvera-López, J. A. Carrasco-Ochoa, and J. F. Martínez-Trinidad. Prototype selection via prototype relevance. In *Iberoamerican Congress on Pattern Recognition*, pages 153–160. Springer, 2008.

[176] B. Pang and L. Lee. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135, 2008.

[177] Z. Pawlak. *Rough sets: Theoretical aspects of reasoning about data*, volume 9. Springer Science & Business Media, 2012.

[178] V. Pawlowsky-Glahn and A. Buccianti. *Compositional data analysis: Theory and applications*. John Wiley & Sons, 2011.

[179] J. Pearl and K. Mohan. Recoverability and testability of missing data: Introduction and summary of results. *Available at SSRN 2343873*, 2013.

[180] K. Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901.

[181] H. Peng, F. Long, and C. Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on pattern analysis and machine intelligence*, 27(8):1226–1238, 2005.

[182] T. M. Phuong, Z. Lin, and R. B. Altman. Choosing snps using feature selection. *Journal of bioinformatics and computational biology*, 4(02):241–257, 2006.

[183] R. F. Potthoff, G. E. Tudor, K. S. Pieper, and V. Hasselblad. Can one assess whether missing data are missing at random in medical studies? *Statistical methods in medical research*, 15(3):213–234, 2006.

[184] F. Provost and T. Fawcett. Robust classification for imprecise environments. *Machine learning*, 42(3):203–231, 2001.

[185] P. Pudil, J. Novovičová, and J. Kittler. Floating search methods in feature selection. *Pattern recognition letters*, 15(11):1119–1125, 1994.

[186] W. F. Punch III, E. D. Goodman, M. Pei, L. Chia-Shun, P. D. Hovland, and R. J. Enbody. Further research on feature selection and classification using genetic algorithms. In *ICGA*, pages 557–564, 1993.

[187] D. Pyle. *Data preparation for data mining*, volume 1. Morgan Kaufmann, 1999.

[188] J. R. Quinlan. *C4. 5: programs for machine learning*. Elsevier, 2014.

[189] M. G. Rahman and M. Z. Islam. Fimus: A framework for imputing missing values using co-appearance, correlation and similarity analysis. *Knowledge-Based Systems*, 56:311–327, 2014.

[190] T. Raicharoen and C. Lursinsap. A divide-and-conquer approach to the pairwise opposite class-nearest neighbor (pocnn) algorithm. *Pattern recognition letters*, 26(10):1554–1567, 2005.

[191] E. Ramos-Martinez, A. M. Herrera Fernandez, J. Izquierdo, and R. Perez-Garcia. A multi-disciplinary procedure to ascertain biofilm formation in drinking water pipes. In *International Congress on Environmental Modelling and Software*. iEMSs, 2016.

[192] M. Refaat. *Data preparation for data mining using SAS*. Morgan Kaufmann, 2010.

[193] T. Reinartz. A unifying view on instance selection. *Data Mining and Knowledge Discovery*, 6(2):191–210, 2002.

[194] A. C. Rencher. *Methods of multivariate analysis*, volume 492. John Wiley & Sons, 2003.

[195] J. Reunanen. Overfitting in making comparisons between variable selection methods. *Journal of Machine Learning Research*, 3(Mar):1371–1382, 2003.

[196] J. C. Riquelme, J. S. Aguilar-Ruiz, and M. Toro. Finding representative patterns with ordered projections. *Pattern Recognition*, 36(4):1009–1018, 2003.

[197] G. Ritter, H. Woodruff, S. Lowry, and T. Isenhour. An algorithm for a selective nearest neighbor decision rule. *IEEE Transactions on Information Theory*, 21(6):665–669, 1975.

[198] J. C. Roberts. State of the art: Coordinated & multiple views in exploratory visualization. In *Coordinated and Multiple Views in Exploratory Visualization, 2007. CMV'07. Fifth International Conference on*, pages 61–71. IEEE, 2007.

[199] J. Rodas, K. Gibert, and J. E. Rojo. Kdsm methodology for knowledge discovery from ill-structured domains presenting very short and repeated serial measures with blocking factor. In *Topics in Artificial Intelligence*, pages 228–238. Springer, 2002.

[200] G. Roffo, S. Melzi, and M. Cristani. Infinite feature selection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4202–4210, 2015.

[201] D. B. Rubin. Inference and missing data. *Biometrika*, 63(3):581–592, 1976.

[202] Y. Saeys, I. Inza, and P. Larrañaga. A review of feature selection techniques in bioinformatics. *bioinformatics*, 23(19):2507–2517, 2007.

[203] J. L. Schafer and J. W. Graham. Missing data: our view of the state of the art. *Psychological methods*, 7(2):147, 2002.

[204] S. C. Shiu, D. S. Yeung, C. H. Sun, and X. Z. Wang. Transferring case knowledge to adaptation knowledge: An approach for case-base maintenance. *Computational Intelligence*, 17(2):295–314, 2001.

[205] SIASAR. http://siasar.org/reportes/resumen_regional_sistema_distrito/resumen_regional_sistema_distrito.php, 2016.

[206] K. Singh and S. Upadhyaya. Outlier detection: applications and techniques. *International Journal of Computer Science Issues*, 9(1):307–323, 2012.

[207] P. Somol, P. Pudil, J. Novovičová, and P. Paclık. Adaptive floating search methods in feature selection. *Pattern recognition letters*, 20(11):1157–1163, 1999.

[208] B. Spillmann, M. Neuhaus, H. Bunke, E. Pekalska, and R. P. Duin. Transforming strings to vector spaces using prototype selection. In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, pages 287–296. Springer, 2006.

[209] C. Stanfill and D. Waltz. Toward memory-based reasoning. *Communications of the ACM*, 29(12):1213–1228, 1986.

[210] S. D. Stearns. On selecting features for pattern classifiers. In *Proceedings of the 3rd International Joint Conference on Pattern Recognition*, pages 71–75, 1976.

[211] Y. Sun, C. Babbs, and E. Delp. A comparison of feature selection methods for the detection of breast cancers in mammograms: adaptive sequential floating search vs. genetic algorithm. In *2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, pages 6532–6535. IEEE, 2006.

[212] D. F. Swayne, D. Cook, and A. Buja. Xgobi: Interactive dynamic data visualization in the x window system. *Journal of Computational and Graphical Statistics*, 7(1):113–130, 1998.

[213] M. Templ, A. Kowarik, and P. Filzmoser. Iterative stepwise regression imputation using standard and robust methods. *Computational Statistics & Data Analysis*, 55(10):2793–2806, 2011.

[214] H. C. Thode Jr. Testing for normality, vol. 164 of statistics: Textbooks and monographs, 2002.

[215] I. Tomek. An experiment with the edited nearest-neighbor rule. *IEEE Transactions on systems, Man, and Cybernetics*, 6:448–452, 1976.

[216] P. Torres, C. H. Cruz, and P. J. Patiño. Índices de calidad de agua en fuentes superficiales utilizadas en la producción de agua para consumo humano: Una revisión crítica. *Revista Ingenierías Universidad de Medellín*, 8(15):79–94, 2009.

[217] A. Valls, J. Pijuan, M. Schuhmacher, A. Passuello, M. Nadal, and J. Sierra. Preference assessment for the management of sewage sludge application on agricultural soils. *International Journal of Multicriteria Decision Making*, 1(1):4–24, 2010.

[218] C. J. Veenman and M. J. Reinders. The nearest subclass classifier: A compromise between the nearest mean and nearest neighbor classifier. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(9):1417–1429, 2005.

[219] A. Vellido. Missing data imputation through gtm as a mixture of t-distributions. *Neural Networks*, 19(10):1624–1635, 2006.

[220] A. Vicente. Mineria de datos aplicada a la detección de fraude con tarjetas de crédito. Master's thesis, Degree on Statistics. Universitat Politècnica de Catalunya., 2016.

[221] J. Villar, A. Otero, J. Otero, and L. Sánchez. Taximeter verification with gps and soft computing techniques. *Soft Computing*, 14(4):405–418, 2010.

[222] S. E. Wakefield, S. J. Elliott, D. C. Cole, and J. D. Eyles. Environmental risk and (re) action: air quality, health, and civic involvement in an urban industrial neighbourhood. *Health & Place*, 7(3):163–177, 2001.

[223] G.-R. Walther, E. Post, P. Convey, A. Menzel, C. Parmesan, T. J. Beebee, J.-M. Fromentin, O. Hoegh-Guldberg, and F. Bairlein. Ecological responses to recent climate change. *Nature*, 416(6879):389–395, 2002.

[224] G. M. Weiss and F. Provost. Learning when training data are costly: the effect of class distribution on tree induction. *Journal of Artificial Intelligence Research*, 19:315–354, 2003.

[225] D. Wettschereck, D. W. Aha, and T. Mohri. A review and empirical evaluation of feature weighting methods for a class of lazy learning algorithms. *Artificial Intelligence Review*, 11(1-5):273–314, 1997.

[226] D. L. Wilson. Asymptotic properties of nearest neighbor rules using edited data. *IEEE Transactions on Systems, Man, and Cybernetics*, 3:408–421, 1972.

[227] D. R. Wilson and T. R. Martinez. Reduction techniques for instance-based learning algorithms. *Machine learning*, 38(3):257–286, 2000.

[228] I. H. Witten, E. Frank, and M. A. Hall. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Publishers Inc., 2011.

[229] P. C. Wong. Visual data mining. *IEEE Computer Graphics and Applications*, 19(5):20–21, 1999.

[230] C.-F. J. Wu. Jackknife, bootstrap and other resampling methods in regression analysis. *the Annals of Statistics*, pages 1261–1295, 1986.

[231] L. Xu, M.-Y. Chow, and L. S. Taylor. Power distribution fault cause identification with imbalanced data using the data mining-based fuzzy classification e-algorithm. *IEEE Transactions on Power Systems*, 22(1):164–171, 2007.

[232] J. Yang and V. G. Honavar. Feature subset selection using a genetic algorithm. *IEEE Intelligent Systems*, 13(2):44–49, 1998.

[233] L. A. Zadeh. Discussion: Probability theory and fuzzy logic are complementary rather than competitive. *Technometrics*, 37(3):271–276, 1995.

[234] H. Zhang and G. Sun. Optimal reference subset selection for nearest neighbor classification by tabu search. *Pattern Recognition*, 35(7):1481–1490, 2002.

[235] Z. Zhao, R. Zhang, J. Cox, D. Duling, and W. Sarle. Massively parallel feature selection: an approach based on variance preservation. *Machine learning*, 92(1):195–220, 2013.