Available online at www.sciencedirect.com

ScienceDirect

Procedia Engineering 70 (2014) 791 – 799

Procedia Engineering

www.elsevier.com/locate/procedia

12th International Conference on Computing and Control for the Water Industry, CCWI2013

On-line learning of predictive kernel models for urban water demand in a smart city

M. Herrera[a,*], J. Izquierdo[b], R. Pérez-García[b], D. Ayala-Cabrera[b]

[a]BATir, Université libre de Bruxelles, Av. F. Roosevelt, 50 (CP 194/2) B-1050 Bruxelles (Belgium)
[b]Fluing - IMM, Universitat Politècnica de València, C. de Vera s/n, 46022 Valencia (Spain)

Abstract

This paper proposes a multiple kernel regression (MKr) to predict water demand in the presence of a continuous source of information. MKr extends the simple support vector regression (SVR) to a combination of kernels from as many distinct types as kinds of input data are available. In addition, two on-line learning methods to obtain real time predictions as new data arrives to the system are tested by a real-world case study. The accuracy and computational efficiency of the results indicate that our proposal is a suitable tool for making adequate management decisions in the smart cities environment.

Keywords: Smart cities; urban water demand; kernel regression; on-line learning

1. Introduction

Nowadays, a city can be understood as a platform where people live, work, and consume resources; as well as being a framework where companies and public administrations offer services to the society. The use of information and communication technology (ICT) techniques, thus, is a must to automatically and efficiently address the management of infrastructures and services. This is the origin of the concept of smart city, a concept based on urban mobility, energy efficiency, and sustainable management of resources. Thus, suitable management of its water supply system is one of the main targets to be achieved by a city today. Among the management infrastructures and urban facilities in a smart city, the work relating to water supply must be highlighted. Sensors, meters and GPS georeferenced devices offer data from water consumption, pressure levels and quality in real-time. Interpreting all this amount of information is not trivial and requires tools capable of handling large and/or complex databases.

This paper focuses on water demand prediction in the presence of a continuous source of information. Thus, water consumption data can be registered by flowmeters, and sent by radio-frequency to a central database for storage and posterior analysis. New paradigms introduced in data management within the smart city framework make that the model can take into consideration all the available information, and can be efficiently updated in real time. Our

[*] Coresponding author. Tel.: +32-2-650-2159.
E-mail address: mherrera@ulb.ac.be

doi:10.1016/j.proeng.2014.02.086

proposal is to approach this concept, firstly by the use of multiple kernel regression (MKr), Qiu and Lane (2005), that allows to manage heterogeneous data, extending the simple support vector regression (SVR), Schölkopf and Smola (2001), Smola and Schölkopf (2004), to a combination of kernels from as many distinct types as kinds of input data, Gönen and Alpaydin (2011). It represents an interesting additional expansion of the work by Herrera et al. (2010), where several methods for predicting hourly water demand were compared. Other outstanding works have used different sources of information. An et al. (1995) that proposed an extension of the rough sets methodology describing relationships between weather factors and water consumptions. Lertpalangsunti et al. (1999) developed hybrid intelligent forecasting systems, based on the concept of communication between different components, and compared them with different regression models. Jain and Ormsbee (2002) formulated demand levels as a function of climate variables (such as air temperature, volume, and the occurrence of rainfall) and previous water demand. Concerning the use of SVMs, Khan and Coulibaly (2006) conducted a comparative study between SVMs, neural networks, and the traditional seasonal autoregressive model in the forecasting of the water level of a lake. Msiza et al. (2007) described a similar study with the application to water demand of time series forecasting. Two other methods were proposed by Chen, Chen and Zhang (2006a,b), whereby hourly water demand is predicted using Bayesian and non-Bayesian least squares SVMs.

The majority of hydraulic modeling found in literature are done off-line, which do not represent well the current state of the water supply system for operational purposes, especially in emergency events, Machell et al. (2009). The other disadvantage of off-line modeling is that unknown parameters are updated by using short term sample of hydraulic data, Preis et al. (2009). In addition, we should be able to exploit the high amount of data generation in real-time within a smart city framework. Thus, two on-line learning MKr versions are proposed to update the current model to a more accurate one, Hoi et al. (2013), avoiding the computational efforts associated with re-calculating the whole process each time that new data are available, Herrera and Filomeno Coelho (2013). These approaches are by sliding (technique introduced by van Vaerenbergh et al. (2006)) and a modification of growing windows of data, updating the model just in the MKr kernel matrix and in their possible combination of single kernels. The aim of these on-line methods is to achieve an increase in the performance of the process proposed without increasing the original algorithm's computational time.

The road-map of the paper is as follows. Section 2 introduces the MKr methodology for heterogeneous data types. The theoretical proposal of this work is completed by suggesting two on-line MKr methodologies to update MKr in real time in Section 3. Section 4 shows a number of results of these methodologies when tested in a real-world case study based on hourly water demand data. A conclusions section closes the paper.

2. Multiple kernel regression

Common kernel-based learning methods (Schölkopf and Smola (2001), Shawe-Taylor and Cristianini (2006)) use an implicit mapping of the input data into a high dimensional feature space defined by a kernel function, i.e. a function, K, returning the inner product $\langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x}') \rangle$ between the images of two data points $\boldsymbol{x}, \boldsymbol{x}'$ in the feature space (see Table 1). The choice of the map, ϕ, aims to convert the nonlinear relations into linear ones. The learning then takes place in the feature space and the learning algorithm can be expressed so that the data points only appear inside dot products with other points. This is often referred to as the "kernel trick", Schölkopf and Smola (2001), Schölkopf (2000).

Table 1. Short list of some common kernel functions.

Name	Expression
Gaussian	$K(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{\lVert\boldsymbol{x}-\boldsymbol{x}'\rVert^2}{2\sigma^2}\right)$
ANOVA	$K(\boldsymbol{x}, \boldsymbol{x}') = \sum \exp\left(-\sigma(x^k - x'^k)^2\right)^d$
Linear	$K(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{x}^T \boldsymbol{x}' + c$
Polynomial	$K(\boldsymbol{x}, \boldsymbol{x}') = (\alpha \boldsymbol{x}^T \boldsymbol{x}' + c)^d$
Rational Quadratic	$K(\boldsymbol{x}, \boldsymbol{x}') = 1 - \frac{\lVert\boldsymbol{x}-\boldsymbol{x}'\rVert^2}{\lVert\boldsymbol{x}-\boldsymbol{x}'\rVert^2+c}$

The use of kernel methods lends itself well to the problem of data integration as it enables multiple types of data to be converted into a common usable format. These can be combined eventually with a weighted summation and used as training data for a classical support vector regression (SVR) scheme, Sonnenburg et al. (2006). The principles of SVR are summarized below.

2.1. Introduction to support vector regression

The key characteristic of SVR is that it allows to specify a margin, ε, within which we are willing to accept errors in the sample data without they affecting the predictions quality. The SVR predictor is defined by those points which lie outside the region formed by the band of size $\pm\varepsilon$ around the regression (see Eq.(1)). Those vectors are the so-called *support vectors*.

$$\hat{f}(\boldsymbol{x}) = \langle \boldsymbol{w}, \phi(\boldsymbol{x}) \rangle + b \tag{1}$$

The goal is to find a function $\hat{f}(x)$ that at most deviates ε from the observed output, y_i, for the regression with the training data and at the same time minimizes the model complexity (see Eq.(2)).

$$\begin{aligned} &\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|^2 \\ &\text{s. to } y_i - \langle \boldsymbol{w}, \phi(x_i) \rangle - b \leq \varepsilon \\ &\qquad \langle \boldsymbol{w}, \phi(x_i) \rangle + b - y_i \leq \varepsilon \end{aligned} \tag{2}$$

The constraints of Eq.(2) assume that $\hat{f}(\boldsymbol{x})$ exists for all y_i with precision $\pm\varepsilon$. Nevertheless, the solution may actually not exist or it would be possible to achieve better predictions if outliers were allowed. Those are the reasons to include *slack variables* on the regression. Thus we have ξ^+ and ξ^- such that:

$$\xi^+ = \hat{f}(x_i) - y(x_i) > \varepsilon \tag{3}$$

$$\xi^- = y(x_i) - \hat{f}(x_i) > \varepsilon \tag{4}$$

and the objective function and constraints for SVR are

$$\begin{aligned} &\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\frac{1}{n}\sum_{i=1}^{n}(\xi_i^+ + \xi_i^-) \\ &\text{s. to } y_i - \langle \boldsymbol{w}, \phi(x_i) \rangle - b \leq \varepsilon + \xi_i^+, \\ &\qquad \langle \boldsymbol{w}, \phi(x_i) \rangle + b - y_i \leq \varepsilon + \xi_i^-, \\ &\qquad \xi_i^+, \xi_i^- \geq 0 \;\; i = 1, \ldots, n \end{aligned} \tag{5}$$

where n is the number of training patterns and C is a trade-off parameter between model complexity and training error. Additionally, ξ^+ and ξ^- are slack variables for exceeding the target value by more than ε and for being below the target value by more than ε, respectively. This method of tolerating errors is known as *ε-insensitive*, Schölkopf and Smola (2002), and its final expression, after solving the dual problem of Eq.(5), is done by Eq.(6).

$$\hat{f}(\boldsymbol{x}) = b + \sum_{i=1}^{n} \omega_i K(x_i, \boldsymbol{x}) \tag{6}$$

2.2. Multiple kernel regression

The SVR method uses a single mapping function ϕ, and hence a single kernel function K. If a data set has a locally varying distribution, using a single kernel may not catch up correctly the varying distribution. Kernel fusion can help to deal with this problem, Christmann and Hable (2012). Recent applications, such as we can found in Lanckriet et al. (2004), and developments based on support vector machines have shown that using multiple kernels instead of a single one can enhance interpretation of the decision function and improve classifier performance, Sonnenburg et al. (2006). By the use of different kernels we can approach problems from different data nature too. It also represents an advantage in the perspective of mixed variable programming, such is indicated in Hemker (2008), Abramson et al. (2004). The kernel fusion is straightforward using several mapping functions combined, instead of one single mapping function.

$$\Phi(\boldsymbol{x}) = [\phi_1(\boldsymbol{x}), \phi_2(\boldsymbol{x}), \ldots, \phi_M(\boldsymbol{x})] \tag{7}$$

We adopt the weighted sum fusion with the following mapping functions:

$$\Phi(\boldsymbol{x}) = [\,\sqrt{\mu_1}\phi_1(\boldsymbol{x}),\ \sqrt{\mu_2}\phi_2(\boldsymbol{x}), \ldots,\ \sqrt{\mu_M}\phi_M(\boldsymbol{x})] \tag{8}$$

where $\mu_1, \mu_2, \ldots, \mu_M$ are weights of component functions. Now, the regression problem includes the optimization of two parts. One part is the regression hyperplane $f(\boldsymbol{x})$ and the other part is the weight vector $\boldsymbol{\mu} = [\mu_1, \mu_2, \ldots, \mu_M]$. The idea is to approach this two parts of the optimization process in just one step, based on the parametric dependence idea.

The resulting multi-kernel, expressed by Eq.(9),

$$\begin{aligned}\tilde{K}(x_i, x_j) &= < \Phi(x_i), \Phi(x_j) > \\ &= \mu_1 < \phi_1(x_i), \phi_1(x_j) > + \mu_2 < \phi_2(x_i), \phi_2(x_j) > + \ldots \\ &+ \mu_M < \phi_M(x_i), \phi_M(x_j) > \\ &= \mu_1 K_1(x_i, x_j) + \mu_2 K_2(x_i, x_j) + \ldots + \mu_M K_M(x_i, x_j) \\ &= \sum_{s=1}^{M} \mu_s K_s(x_i, x_j)\end{aligned} \tag{9}$$

is the weighted sum of M kernel functions which will be another kernel function, Shawe-Taylor and Cristianini (2006). We can solve the regression hyperplane by plugging this multi-kernel on equation concerning the SVR regression surface, Smola and Schölkopf (2004), as Eq.(10) shows.

$$\hat{f}(\boldsymbol{x}) = b + \sum_{i=1}^{n} (\alpha_i^+ - \alpha_i^-) \tilde{K}(x_i, \boldsymbol{x}) \tag{10}$$

3. Windowing for on-line MKr

The aim of the windowing strategies for on-line MKr (see Fig.1) is to improve the performance of the process without increasing the original algorithm's computation time. The new windowing strategy of "worm-windows" is proposed in this work. This method has two "expand-shrink" phases: firstly, it allows increasing the kernel matrix as its size remains adequate to work and there is not any over-fitting issue. Shrinking the kernel matrix to the original size is proposed when its size tends to be not computationally efficient.

3.1. On-line MKr by sliding windows

The sliding window approach consists in only taking the last N pairs of the stream to perform the multi-kernel regression. When we obtain a new observed pair $\{x_{n+1}, y_{n+1}\}$, we first down-size the kernel matrix, $K_j^{(n)}$, by extracting the contribution from x_{n-N} (see Eq.(11))

$$\check{K}_j^{(n)} = \begin{pmatrix} K_j^{(n)}(2,2) & \cdots & K_j^{(n)}(2,N) \\ \vdots & \ddots & \vdots \\ K_j^{(n)}(N,2) & \cdots & K_j^{(n)}(N,N) \end{pmatrix} \tag{11}$$

and then we augment again the $K_j^{(n)}$ dimension by importing the data input x_{n+1} to obtain the kernel expressed in Eq.(12).

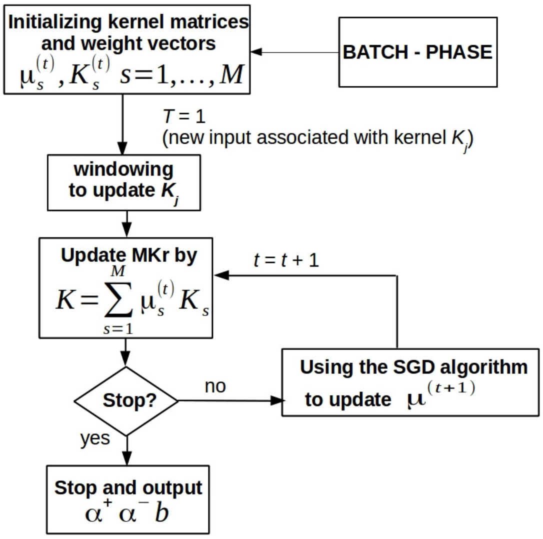
Initializing kernel matrices and weight vectors
$\mu_s^{(t)}, K_s^{(t)}\ s=1,\ldots,M$
BATCH - PHASE
$T = 1$
(new input associated with kernel K_j)
windowing to update K_j
Update MKr by
$K=\sum_{s=1}^{M}\mu_s^{(t)}K_s$
$t = t + 1$
Stop?
no
Using the SGD algorithm to update $\mu^{(t+1)}$
yes
Stop and output
$\alpha^+\ \alpha^-\ b$

Fig. 1. Online multiple kernel regression.

$$K_j^{(n+1)} = \begin{pmatrix} \check{K}_j^{(n)} & K_j(\boldsymbol{X}_n, x_{n+1}) \\ K_j(x_{n+1}, \boldsymbol{X}_n) & K_j(x_{n+1}, x_{n+1}) + \lambda \end{pmatrix} \tag{12}$$

where $\boldsymbol{X}_n = (x_{n-N+1}, \ldots, x_n)^T$ and λ is a correction factor.

Next, the kernel matrices are summed again (see Fig.1) and their weights, μ, should be updated too. As it is a particular case of the calculation of weights corresponding to the batch phase of the overall process, the proposal is to follow a Stochastic Gradient Descent (SGD) algorithm, Kivinen et al. (2004), Karatzoglou (2006).

3.2. On-line MKr by “worm” windows

The so-called *worm* window approach consists in augmenting the kernel matrix size when new data become available. A shrink to the original size is proposed when its performance falls below a certain tolerance limit. Then, the last n data are taken into account. The performance of the first growing phase of the algorithm should be checked after the first iteration; simulating its computational efficiency with random data and establishing a maximum size. Besides, over-fitting issues should be considered in order to shrink the kernel matrix.

The *worm* windows alternative should offer a major stability in their predictions as consequence of always considering a number of data equal or greater than sliding windows. On the other hand, the sliding alternative requires a lower computational efforts and will take a major proportion of new data. Thus, depending on the nature of the database, its variability and the targets of the analysis, we could choose one of these two options for the on-line learning.

4. Case study

The complete water supply network under study has been divided into hydraulic zones. Starting at a treatment plant, a water main distributes water to the sectors. Each sector has one or two sources and may have or not an output. A number of control valves isolating or communicating each zone with the whole network are essential. Water consumption in each sector is registered by flowmeters, and registered data are sent by radio-frequency to a central database for storage and posterior analysis. For this work, field measurements were collected from January 2005 through April 2005 on a hourly basis. The data is divided as follows: the 1000 first data are used for training and validation and the next 300 data are used for testing the predictive models (the on-line case will be explained later).

In addition to water consumption values, we also have information concerning daily values of climate variables: temperature in Celsius, wind velocity in km/h, millimeters of rain, and atmospheric pressure (mean sea level pressure, measured in millibars). The box plots in Fig.2 show a very limited variability of the rain values, as well as the asymmetry in the distribution of the wind velocity values. The time plots of the various variables reveal different trends and enable us to observe, for example, the inverse association between the atmospheric pressure and the temperature for these months. We also observe that there is little water demand in the beginning of this period, coinciding with low temperatures. There is then a growth in the consumption followed by a growing variability period (coinciding with the irregular weather of the first days of spring). From these graphs it seems obvious that there is some influence of the temperature on water demand, as well as of the wind velocity and the volume of rain (these influences are shown significant after applying a Spearman rank correlation test). The influence of a rainy day in water demand is of especial interest: just the day that it rains this demand increases on average and goes down this level the next day. This may be attributed to the fact that people tend to stay longer at home during rainy days.

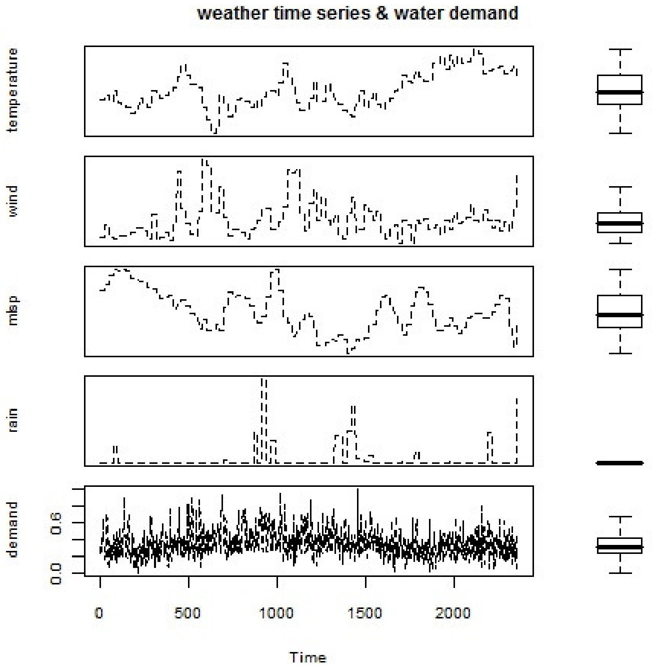

Fig. 2. Visualization of the impact of weather variables in water demand

Given the multivariate time series database described previously, our target is to predict the hourly water demand by MKr method (using Gaussian kernel for the continuous data and the Linear one for the categorical part - day of week and hour of day -). A comparison, in terms of root mean squared error (RMSE), between MKr and SVR with a single (Gaussian) kernel, is shown in Table 2. The parameters were tuned by a Grid Search algorithm in both cases.

The MKr's predicted values vs. (new) observed data is represented in Fig.3.

Despite these good results obtained by MKr and SVR, by new data arrival the models tend to be slightly obsolete until reaching a number of new data from which is necessary recalculate those models. The windowing techniques introduced in Section 3 attempt to achieve an on-line updating of the model without major computational efforts. This allows to control the augmenting error of the process in real-time. Fig.4 shows a comparison of the error behavior by each different on-line method: 'worm' windows (starting with the last 1000 data and shrinking the growing every 150 new data to the last 1000 again), sliding windows (of 1000 fixed last data), and the model without any updates.

In Fig.4 we can observe how the error grows with the 1000 new data available in both model without updates and in the on-line sliding windows. Nevertheless, the 'worm' on-line MKr can control this error proposing a stable approach.

Table 2. MKr vs. SVR comparison via RMSE.

RMSE (test phase)	Model	Model Parameters
0.10	MKr	- general parameters $C = 638$; $\varepsilon = 0.12$ - combining kernels: $\mu_1 = 1.58$; $\mu_2 = 1.17$ - for Gaussian and linear kernels: $\sigma = 1.37$; $d = 3.55$
0.13	SVR	$C = 800$; $\varepsilon = 0.01$ $\sigma = 0.95$

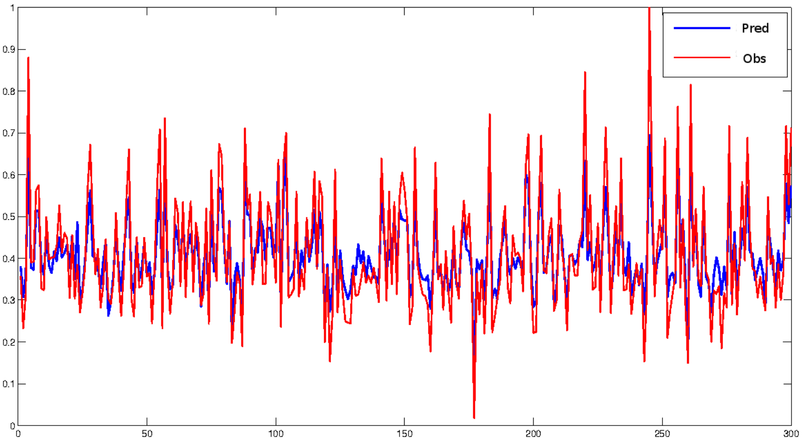

Fig. 3. Observed vs. (MKr) predicted water demand

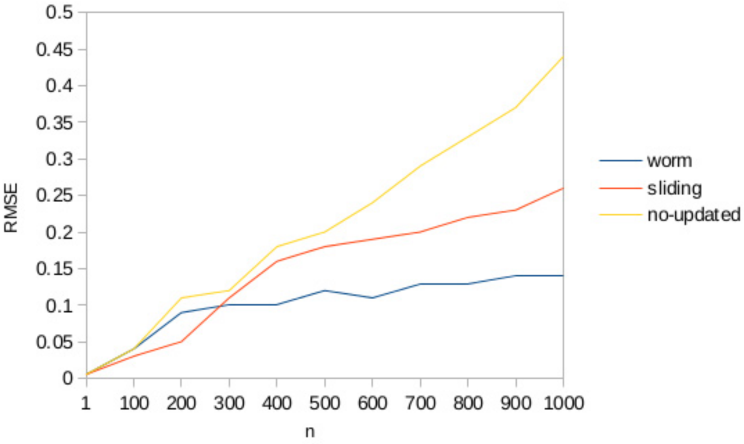

Fig. 4. On-line techniques comparison

5. Conclusions

A smart city is a concept in a continuously expansion. New perspectives in the management of cities resources are necessary as well as real time response to operational scenarios should use a maximum of the high amount of available information. Knowledge of water demand behavior is a plus in supply management and decision making of a water distribution system. Methods based on Machine Learning, such as MKr and SVR introduced here, have emerged as an attractive option for prediction and classification in water systems due to their fast execution time and their easy adaptation to any novelty in the working scenario. An on-line tool such the presented on-line windowing MKr takes into account all the information that can generate a city nowadays. The response is immediate because it does not need to replicate the model each time that we use new information and it looses a lower quantity of information than the no-updated model. Tuning the windowing parameters (the optimum size of the window or the best time to shrink in the case of the 'worm' alternative) and combining the use of these on-line tools with methods to simplify the input are new challenges to approach in further works.

Acknowledgements

This work has been supported by project IDAWAS, DPI2009-11591, of the Dirección General de Investigación of the Ministerio de Ciencia e Innovación of Spain, and ACOMP/2011/188 of the Conselleria d'Educació of the Generalitat Valenciana.

References

Abramson, M., Audet, C., Dennis, D.E.J., 2004. Filter pattern search algorithms for mixed variable constrained optimization problems. SIAM Journal on Optimization 11, 573–594.

An, A., Shan, N., C., C., Cercone, N., Ziarko, W., 1995. Discovering rules form data for water demand prediction, in: Workshop on Machine Learning and Expert System - IJCAI'95, pp. 187–202.

Chen, L., Zhang, T., 2006a. Hourly water demand forecast model based on bayesian least squares support vector machine. Tianjin University Science and Technology 39, 1037–1042.

Chen, L., Zhang, T., 2006b. Hourly water demand forecast model based on least squares support vector machine. Tianjin University Science and Technology 38, 1528–1530.

Christmann, A., Hable, R., 2012. Consistency of support vector machines using additive kernels for additive models. Computational Statistics & Data Analysis 56, 854–873.

Gönen, M., Alpaydin, E., 2011. Multiple kernel learning algorithms. Machine Learning Research 12, 2211–2268.

Hemker, T., 2008. Derivative free surrogate optimization for mixed-integer nonlinear black box problems in engineering. Ph.D. thesis. Technisen Universität Darmstad, Germany.

Herrera, M., Filomeno Coelho, R., 2013. Windowing strategies for on-line multiple kernel regression, in: International Workshop on Advances in Regularization, Optimization, Kernel Methods and Support Vector Machines, ROKS-2013, Leuven, Belgium. pp. 105–106.

Herrera, M., Torgo, L., Izquierdo, J., Pérez-García, R., 2010. Predictive models for forecasting hourly urban water demand. Hydrology 387, 141–150.

Hoi, S.C., Jin, R., Zhao, P., Yang, T., 2013. Online multiple kernel classification. Machine Learning 90, 289–316.

Jain, A., Ormsbee, L.E., 2002. Short-term water demand forecasting modelling techniques-conventional versus AI. AWWA 94, 64–72.

Karatzoglou, A., 2006. Kernel methods software, algorithms and applications. Ph.D. thesis.

Khan, M., Coulibaly, P., 2006. Application of support vector machine in lake water level prediction. Hydrological Engineering 11, 199–205.

Kivinen, J., Smola, A., Williamson, R.C., 2004. Online learning with kernels. IEEE Transactions on Signal Processing 52(8), 2165–2176.

Lanckriet, G., Cristianini, N., Barlett, P., El-Ghaoui, L., Jordan, M.I., 2004. Learning the kernel matrix with semi-definite programming. Machine Learning Research , 27–72.

Lertpalangsunti, N., Chan, C., Mason, R., Tontiwachwuthikul, P., 1999. A tool set for construction of hybrid intelligent forecasting systems: application for water demand prediction. Artificial Intelligence in Engineering 13, 21–42.

Machell, J., Mounce, S.R., Boxall, J.B., 2009. Online modelling of water distribution systems: a UK case study. Drinking Water, Engineering and Science 2, 279–294.

Msiza, I., Nelwamondo, F., Marwala, T., 2007. Artificial neural networks and support vector machines for water demand time series forecasting, in: IEEE International Conference on Systems, Man and Cybernetics, pp. 638–643.

Preis, A., Whittle, A., Ostfeld, A., 2009. Online hydraulic state prediction for water distribution systems, in: World Environmental and Water Resources Congress, American Society of Civil Engineers (ASCE).

Qiu, S., Lane, T., 2005. Multiple kernel learning for support vector regression. Technical Report. Computer Science Department, University of New Mexico, Albuquerque, NM, USA.

Schölkopf, B., 2000. The kernel trick for distances. Technical Report. Microsoft Research.

Schölkopf, B., Smola, A.J., 2001. Learning with Kernels: Support Vector Ma- chines, Regularization, Optimization, and Beyond. MIT Press, Cambridge, MA, USA.

Schölkopf, B., Smola, A.J., 2002. Learning with kernels. MIT Press.

Shawe-Taylor, J., Cristianini, N., 2006. Kernel Methods for Pattern Analysis. Cambridge University Press.

Smola, A., Schölkopf, B., 2004. A tutorial on support vector regression. Statistics and Computing 14, 199–222.

Sonnenburg, S., Rätsch, G., Schäfer, C., 2006. A general and efficient multiple kernel learning algorithm, in: Weiss, Y., Schölkopf, B., Platt, J. (Eds.), Advances in Neural Information Processing Systems 2006, MIT Press, Cambridge, MA. pp. 1273–1280.

van Vaerenbergh, S., Vía, J., Santamaría, I., 2006. A sliding-window kernel RLS algorithm and its application to nonlinear channel identification, in: IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP 2006, pp. 789–792.