Journal of Hydrology 387 (2010) 141–150

Contents lists available at ScienceDirect

Journal of Hydrology

journal homepage: www.elsevier.com/locate/jhydrol

Predictive models for forecasting hourly urban water demand

Manuel Herrera [a,*], Luís Torgo [b], Joaquín Izquierdo [a], Rafael Pérez-García [a]

[a] IMM-Universidad Politécnica de Valencia Valencia, C. de Vera s/n, Edificio 5C, 46022 Valencia, Spain
[b] LIAAD-INESC Porto LA, R. Ceuta 118 - 6., 4050-190 Porto, Portugal

ARTICLE INFO

Article history:
Received 29 September 2009
Received in revised form 1 January 2010
Accepted 3 April 2010

This manuscript was handled by G. Syme, Editor-in-Chief

Keywords:
Urban water demand
Non-linear time series
Predictive regression models
Machine learning algorithms
Monte Carlo simulations

SUMMARY

One of the goals of efficient water supply management is the regular supply of clean water at the pressure required by consumers. In this context, predicting water consumption in urban areas is of key importance for water supply management. This prediction is also relevant in processes for reviewing prices; as well as for operational management of a water network. In this paper, we describe and compare a series of predictive models for forecasting water demand. The models are obtained using time series data from water consumption in an urban area of a city in south-eastern Spain. This includes highly non-linear time series data, which has conditioned the type of models we have included in our study. Namely, we have considered artificial neural networks, projection pursuit regression, multivariate adaptive regression splines, random forests and support vector regression. Apart from these models, we also propose a simple model based on the weighted demand profile resulting from our exploratory analysis of the data.

In our comparative study, all predictive models were evaluated using an experimental methodology for hourly time series data that detailed water demand in a hydraulic sector of a water supply network in a city in south-eastern Spain. The accuracy of the obtained results, together with the medium size of the demand area, suggests that this was a suitable environment for making adequate management decisions.

1. Introduction

The most important factor in planning and operating a water distribution system is satisfying consumer demand. This means continually providing users with quality water in adequate volumes at reasonable pressure, and so ensuring a reliable water distribution system. Efficiently operating and managing a water-supply system requires short-term water demand forecasts; and the estimation of future municipal water demand is central to the planning of a regional water-supply system (Zhou et al., 2002). Water demand forecasting is becoming an essential tool for the design, operation, and management of water-supply systems in activities such as: planning new developments or system expansion; estimating the size and operation of reservoirs, pumping stations, and pipe capacities; and for urban water management issues (pricing policies and water use restrictions).

Long-term forecasting is required mainly for planning and design; while short-term forecasting is useful in operation and management. As mentioned by Bougadis et al. (2005), short-term demand projections help water managers make better informed water management decisions when balancing the needs of water supply, residential/industrial demands, and stream flows for fish and other habitats. Short-term demand forecasts help utilities plan and manage water demands for near-term events (Jain and Ormsbee, 2002).

In this paper we tackle this kind of temporal landscape, taking our predictive output and using it as input of a previously calibrated water model (i.e. in EPANET Rossman, 2000). The ready availability of hourly predictions of water demand into a calibrated mathematical model is crucial due to a number of reasons:

- From an operative point of view, it enables water managers to determine optimal regulation and pumping schemes to supply the predicted demand. The aim is to improve the energetic efficiency through lower pumping energy consumption.
- From the quality point of view, the more suitable combination of water sources to obtain a given standard in the supplied water may be selected.
- From the vulnerability point of view, the comparison between the predicted and the real flow measurements can help pinpoint possible network failures (water leaks and pipe bursts). This provides the first step of a procedure for establishing an early warning management.

In this paper, our analysis is based on hourly water demand data in a hydraulic sector in a city in south-eastern Spain. Water demand around the Mediterranean basin is growing at an alarming rate. Like many Mediterranean regions, south-eastern Spain is suffering from large increases in the use of ground-water from

* Corresponding author.
E-mail address: mahefe@gmmf.upv.es (M. Herrera).

0022-1694/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.jhydrol.2010.04.005

its major aquifers. This region is a prime example of an area where the aquifers are under pressure, as ground-water supplies essential water needs for agriculture and tourism.

The paper is organized as follows. Section 2 provides a review of the literature on water demand forecasting. In Section 3 a case study based on water demand historical hourly data from a hydraulic sector in a city is presented. Section 4 describes the predictive models used in our study. In this section we discuss the techniques used and propose a simple model based on the weighted demand profile. In Section 5 we present the evaluation methodology and the results of experiments using the data set described before. Section 6 summarizes the conclusions.

2. Literature review

Forecasting water demand is a very active field of study. Early works addressed this question by using mainly traditional statistical models. In recent years, more sophisticated modeling techniques have been applied to this problem. There is a very large set of tools that can be used in this forecasting problem, and so there is an effective need for choosing and understanding the best techniques for a given forecasting application. Perhaps the most frequently adopted methods for forecasting have been linear regression and time series analysis (Jain et al., 2001). In the 1980s Maidment et al. (1985) used short-term Box and Jenkins models for daily municipal water use, and these were modelled as a function of rainfall and air temperature. Once again, Maidment and Miaou (1986) applied this model to the water consumption of nine US cities. Smith (1988) developed time series models to forecast daily municipal water demand, which included day-of-week effects and a randomly varying mean, as these two factors were not included in the Maidment models. In 1995, An et al. (1995) 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.

In the first decade of this century, the technique of artificial neural networks (ANN) was used with the back-propagation algorithm for several civil engineering applications (Lingireddy and Ormsbee, 1973). Zhou et al. (2002) developed time series models for daily water consumption in Melbourne, Australia. ANN models have also been used to model weekly peak demand (Jain and Ormsbee, 2002). These demand levels can be formulated as a function of climate variables (such as air temperature, volume, and the occurrence of rainfall) and previous water demand. In addition to the ANNs, other machine learning methodologies have been applied in forecasting hydraulic time series. Shrestha and Solomatine (2006) presented in 2006 a regression methodology based on fuzzy clusters applied to estimate hydrological data sets. Khan and Coulibaly (2006), in 2006, conducted a comparative study between support vector machines (SVM), ANN, and the traditional seasonal autoregressive model (SAR) in the forecasting of the water level of a lake. In 2007, 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 and Zhang (2006b,a), whereby hourly water demand is predicted using Bayesian and non-Bayesian least squares SVMs. An optimization of pump-scheduling based on forecasting urban water demand for the city of Seoul was proposed by Kim et al. (2007). Other Machine Learning models, like projection pursuit regression (Dahl and Hylleberg, 2004; Friedman and Stueltze, 1981), multivariate adaptive regression splines (Hastie and Tibshirani, 1990; Moisen and Frescino, 2002) and random forests (Breiman, 2001), which are currently unusual in the water demand literature, have been used in this paper with good results.

3. The case study

The study site is located in a hydraulic sector (or a hydraulic zone) in a city in south-eastern Spain. It has a population of approximately 5000 consumers and an extension of nearly 8 km^2. The water demand average is 19 m^3/h with an associated standard deviation of approximately 8 m^3/h. The decision to use only one hydraulic sector for our study is motivated by the following: homogeneity and utility of the results; elimination of sources of bias; avoidance of the impact of a small set of consumers that may incorrectly bias the forecasts due to unusual consumption profiles.

3.1. Exploratory analysis of the data

In this section, we carry out some exploratory analysis of the available data with the goal of providing a better understanding of the modeling problem.

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 or not have 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.

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).

All these factors are connected with the water demand behavior. Temperature is the more relevant because it directly influences multiple sources of water consumption such as showers, water for gardens. But water consumers respond too to the occurrence of rainfall and other climate variables (An et al., 1995). Regarding this connexion, Maidment and Miaou (1986) are critical about the linearity between water demand and weather variables. They suggest that rainfall has a dynamic effect in the sense that it reduces water demand initially, but the effect diminishes over time. These nonlinearities are also of concern to many authors, among others Arbués et al. (2003) and Gato et al. (2007).

Previous analysis of data similar to our case included a distinction between weekdays and weekends. In the present study, we have instead used the day of the week. The justification for this change lies in the observation that there is a clear difference in the demand profile for the different days; namely, Saturdays are clearly different from Sundays. This daily profile can be observed in Fig. 1. It shows the average water consumption for the 24 h of each day. Each average value was calculated from all the available data (i.e. an 8 week period).

The curves show a similar behavior during the early morning. All curves grow from 6:00 am till 10:00 am. From 10:00 am to 4:00 pm, the behaviors are very different depending on the day. In the afternoon, all the curves have the same trend: first decreasing and then increasing (except on weekends). The maximum water demand values (Fig. 2) may be useful for pumping and other water management actions. This, together with the information of the slopes in different time windows is important to add to the forecasts.

In this way, the model based on the pattern curve (Section 4.6), may increase the sensitivity in detecting peaks in water demand. We have also carried out an analysis of the relationship between the weather information and the water demand. Fig. 3 provides

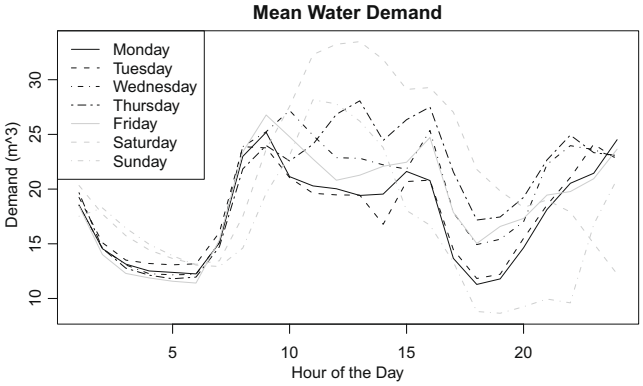

Fig. 1. Evolution of the mean water demand in the 24 h of each day.

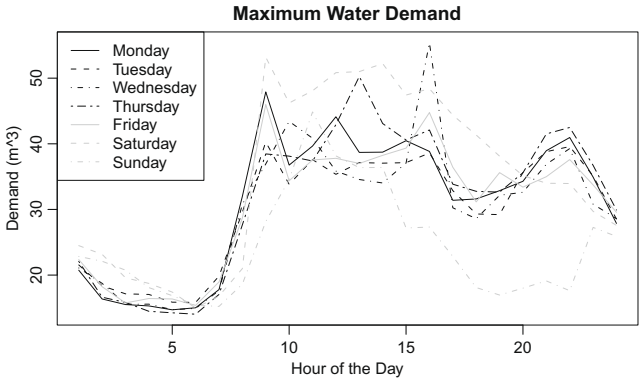

Fig. 2. Maximum water demand per hour.

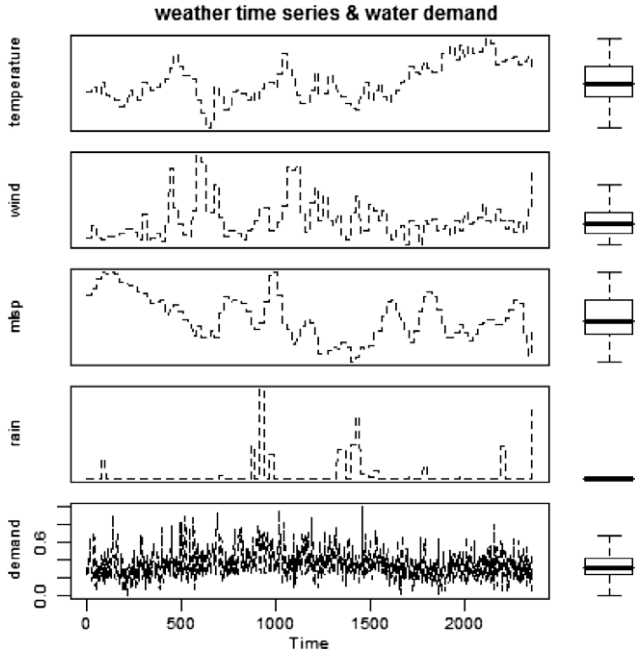

Fig. 3. Visualization of the impact of weather variables in water demand.

such analysis in the form of paired graphs for several weather-related variables and also water demand. The left side graphs show the time plots of the respective time series, while the right side graph is a box plot of the values of the variable across the period under analysis. This last graph provides an idea of the distribution of the values of the variables.

The box plots in Fig. 3 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.

Given the multivariate time series database described previously, it is possible to use a multiple linear regression method to obtain a model that predicts the value of the future water demand from the past observed demand. From the point of view of multivariate time series theory, the first method of approaching the problem is to use vector AR (VAR) models, or state-space models; (Durbin and Koopman, 2001; Wagner, 1999; Gilbert, 2006) are good references for the fundamentals of these methods. Nevertheless, the behavior of the autocorrelation function of filtered and differenced time series shows clear non-linearities. In this context, we will approach the problem using non-linear forecasting methods, which will also allow to develop more flexible regression models, including for instance, the weather time series.

3.2. The prediction task

In this section we describe the concrete prediction task we have selected to compare a set of alternative models with the goal of forecasting future water demand. Namely, we describe the predictor variables and the target variable that were selected.

With respect to the target variable we have selected the water demand in the next hour as our forecasting goal. This means that at time i we will ask our candidate models to forecast the water demand for time $i+1$ h. Regards the predictor variables used to forecast this future water demand we have used our exploratory analysis described in the previous section to guide our selection. In this context, we have selected the current hour water demand, the previous hour water demand, the water demand for our target hour on the previous week, given the day of the week regularities that were observed, and also the current weather time series values. More formally, we are trying to approximate the unknown multiple regression function,

$$W_{i+1} = f(W_i, W_{i-1}, W_{i-7\times24+1}, t, vv, pnm, pt) \quad (1)$$

where W_i is value of the hourly water demand time series at hour i, and t, vv, pnm and pt (cf. Section 3.1) are the last known values of the weather time series: temperature, wind velocity, atmospheric pressure and rain, respectively (Fig. 3).

Obviously, we are not claiming this is the best setup for this type of problems. Our selection is solely based on our exploratory study of this concrete data set. Nevertheless, all models will be tested and compared on the same predictive task and we have no reasons to believe that different conclusions would be reached if slightly different variables would have been selected for this comparative study.

4. The used predictive models

This section describes the modeling tools we have used in the experimental comparisons performed with our hour demand data set. All experiments were carried out in the R environment for statistical computing (R Development Core Team, 2009).

4.1. Artificial neural networks (ANN)

An artificial neural network (ANN) is an interconnected group of artificial neurons. Each neuron executes a non-linear computation based on the input values and the resulting value is fed to other neurons. Neurons are usually arranged as a series of interconnected layers. Based on the data presented to the network, an algorithm (usually back-propagation) is used to iteratively adjust the neuron connection weights in such a way that the predictive performance of the network (Bishop, 2005) is improved. The steps involved in developing an ANN model are detailed in Maier and Dandy (2000), among other references.

The most common ANN network is the feed-forward network, which uses the back-propagation algorithm for training (Bougadis et al., 2005). Obtaining this type of network is an iterative process in which each sample case is presented several times to the input neurons of the ANN.

Usually, a typical three-layer feed-forward model is used for forecasting purposes (Lingireddy and Ormsbee, 1973). Hidden nodes (h in the next equation) with appropriate non-linear transfer functions are used to process the information received by the p input nodes, each associated with one of the predictors. Finally, the model can be written as (Zhang and Qi, 2005):

$$Y_t = \alpha_0 + \sum_{j=1}^{p} \alpha_j f\left(\sum_{i=1}^{h} \beta_{ij} y_{t-j} + \beta_{0j}\right) + \epsilon_t \tag{2}$$

where p is the number of input nodes, h is the number of hidden nodes, f is a sigmoid transfer function; α_j, with $j = 0,1,\ldots,h$, is the vector of the weights from the hidden to the output nodes and β_{ij}, with $i = 0,1,\ldots,p$ and $j = 1,\ldots,h$, are the weights from the input to hidden nodes. α_0 and β_{0j} are the weights of the arcs leaving from the bias terms.

4.1.1. Tuning ANN

In our comparative study, we have used feed-forward neural networks with one hidden layer and the back-propagation learning algorithm (Zealand et al., 2005, 2006). Specifically, we have used the implementation of this type of ANN available in the `nnet` package (Venables and Ripley, 2002) of the R statistical environment. The input is normalized by subtracting each column of the data set by its mean value and dividing by the standard deviation, to obtain data in the same scale and in agreement with the support domain of the activation functions. In terms of the different parameters of the R function used to obtain the ANN models, we have considered nine different alternatives by varying the number of hidden nodes (parameter `size`) between 3, 5, and 7; and also varying the learning rate (parameter `decay`) between 0.0001, 0.001, and 0.1.

4.2. Projection pursuit regression (PPR)

Projection pursuit regression (PPR) is a powerful, non-parametric regression method proposed by Friedman and Stueltze (1981). Recently, Dahl and Hylleberg (2004) included the method in their comparative study of flexible regression models for industrial purposes. Storlie and Helton (2008a) describe PPR embedded in a sensitivity analysis of multiple predictor smoothing methods. In Storlie and Helton (2008b), the authors revise these techniques in various application environments.

PPR explains the target variable as a sum of spline functions of projections of the input variables. For many practical problems, the data is usually of high dimension. The most common practice is the use of dimension-reduction transformations, such as linear projections, to project the original high dimensional data into a lower-dimensional space, in an attempt to find the intrinsic structure for visual inspection. The PPR model can be written as:

$$y_t = \mu_{\text{ppr}}(x_t, Q) + \epsilon_t \tag{3}$$

where

$$\mu_{\text{ppr}}(x_t, Q) = x\prime_t \beta + \sum_{j=1}^{v} \omega_j \phi_j(x\prime_t \phi_j) \tag{4}$$

and

$$Q = (\beta, \omega_1, \ldots, \omega_v, \phi_1, \ldots, \phi_v) \tag{5}$$

Parameters ϕ_j define the projection of the input vector x_t onto a set of planes indicated by j. These projections are transformed by the non-linear activation functions, noted $\phi_j(\cdot)$, and these, in turn, are linearly combined with weights ω_j and added to the linear part, $x'_t\beta$, to form the output variable y_t.

Friedman and Stueltze (1981) proposed an initial algorithm to obtain an estimate of Q. This algorithm consists of two components: a PP index and a PP algorithm. A PP index, $I(\alpha)$, is the objective function computed on the projected data set, and measures the "interestingness" of the projection α (the possible directions). $I(\alpha)$ is an estimate of the distance between the distribution of the projected data and an uninteresting distribution (implicitly, depending on the data). The larger the index value, the more interesting is the projection and thus the method tries to maximize the value. A PP algorithm is a numerical optimization algorithm that varies the projection direction so as to find the optimal projections.

4.2.1. Tuning PPR

In our experiments, we have again used an implementation of PPR available in R, through the function `ppr()` of package `stats` (R Development Core Team, 2009). In this function, the parameter `nterms` sets the number of linear combinations that will be included in the projection pursuit model. With the use of parameter `max.terms` it is possible to set a maximum number of terms that will be tried. After adding this maximum number of terms, the "worse" terms will be iteratively removed by back-fitting until a model with `nterms` is reached. The levels of optimization (argument `optlevel`) differ in how thoroughly the models are re-fitted during this process. At level 0, the existing ridge terms are not re-fitted. At level 1, the projection directions are not re-fitted, but ridge functions and regression coefficients are fitted. Level 2 re-fits all the terms. In our experiments, we have tried four different levels, together with four values of regression smoothing parameter `bass` (0, 2, 5, 7). This last parameter will increase the regression smoothing as the values grow. In total, 12 PPR variants are checked.

4.3. Multivariate adaptive regression splines (MARS)

Multivariate adaptive regression splines (MARS) were first introduced by Friedman and Stueltze (1981) in an attempt to overcome some of the limitations of regression trees. This procedure generalizes recursive partitioning methods such as classification and regression trees (CART), while sharing their ability to capture high-order interactions. However, the procedure has more power and flexibility to model additive relationships (Hastie and Tibshirani, 1990). MARS is a regression method applicable to high

dimensional data, and is usually considered as an excellent example of a modern statistical approach to regression.

MARS models the target variable using a linear combination of splines, which are automatically built (matching the boundaries of each region) from an increasing set of piecewise-defined linear basic functions (Moisen and Frescino, 2002). The model takes the form of an expansion in product spline basis functions, where the number of these basis functions and the parameters associated with each are determined by the data. The model can be represented in a form that separately identifies the additive contributions and those associated with different multivariate interactions. To avoid overfitting, it is possible to subsequently apply a pruning mechanism to reduce model complexity. One form of writing the MARS model is the following,

$$\hat{f}(x) = a_0 + \sum_{k_m=1} f_i(x_i) + \sum_{k_m=2} f_{ij}(x_i, x_j) + \sum_{k_m=3} f_{ijk}(x_i, x_j, x_k) + \cdots \quad (6)$$

The functions in the first sum are defined as,

$$f_i(x_i) = \sum_{k_m=1/i\in V(m)} a_m B_m(x_i) \quad (7)$$

where $V(m)$ is the variable associated with the mth basis function, B_m, that survives backward selection strategies.

The second sum is over all basis functions that involve two variables,

$$f_i(x_i, x_j) = \sum_{k_m=2/i,j\in V(m)} a_m B_m(x_i, x_j) \quad (8)$$

The third sum is over all basis functions that involve three variables, and so on.

4.3.1. Tuning MARS

MARS models are implemented in several R in packages. In our experiments we have used the implementation in package `earth` (Milborrow, 2009), which is a re-implementation of the original MARS code by Trevor Hastie and Robert Tibshirani (Friedman, 1991) with similar, but not identical, results. We have considered 12 variants of these models formed by different combinations of the parameter `nk` that sets the maximum number of term before pruning (values 10 and 17); two variants (1 and 2) of the parameter `degree` that sets the maximum degree of interaction; and three variants (0.01, 0.001 and 0.0005) of the parameter `thresh` specifying the forward stepwise stopping threshold.

4.4. Support vector regression (SVR)

Smola and Schölkopf (1998, 2004) published a fundamental tutorial giving an overview of the basic ideas underlying SVM for function estimation. Vapnik (1995, 1998), and Shawe-Taylor and Cristianini (2000) are two of the essential references for SVM. These are complemented with the works of Karatzoglou (2006), Karatzoglou et al. (2006) for implementing a SVM and kernel method environment in R Language; or Canu et al. (2003) for developing a MatLab toolbox.

In Support Vector Regression (SVR) the basic idea is to map the data x into a high dimensional feature space F via a non-linear mapping Φ and obtain a linear regression model in this new space:

$$f(x) = (\omega \cdot \Phi(x)) + b \quad (9)$$

with Φ: $R^n \to F$, $\omega \in F$, where b is a threshold.

Thus, linear regression in a high dimensional (feature) space corresponds to non-linear regression in the low dimensional input space R^n. Since Φ is fixed, ω is determined from the data by minimizing the sum of empirical risk $R_{emp}[f]$ and a complexity term $\|\omega\|^2$, which enforces flatness in the feature space:

$$R_{reg}[f] = R_{emp}[f] + \lambda\|\omega\|^2 = \sum_{i=1}^{l} C(f(x_i) - y_i) + \lambda\|\omega\|^2 \quad (10)$$

where l denotes the sample size, $C(\cdot)$ is a cost function (e.g. Vapnik's ϵ-insensitive loss function) and λ is a regularization constant.

For a large set of cost functions, the previous equation can be minimized by solving a quadratic programming problem, which is uniquely solvable. It is possible to write the vector ω in terms of the data points:

$$f(x) = \sum_{i=1}^{l} (\alpha_i - \alpha_i^*)(\Phi(x_i)) \quad (11)$$

with α_i, α_i^* being the solution of the afore-mentioned quadratic programming problem.

The problem may be rewritten as products in the low dimensional space:

$$\omega = \sum_{i=1}^{l} (\alpha_i - \alpha_i^*)(\Phi(x_i) \cdot \Phi(x)) + b = \sum_{i=1}^{l} (\alpha_i - \alpha_i^*) K(x_i, x) + b \quad (12)$$

In Eq. (12) the kernel function is introduced: $K(x_i,x_j) = (\Phi(x_i)\cdot\Phi(x_j))$. It can be shown that any asymmetric kernel function, K, satisfying Mercer's condition, corresponds to a dot product in some feature space. A common kernel is a Radial Basis Function (RBF) kernel:

$$K(x_i, x_j) = \exp\left(-\gamma \times \|x_i - x_j\|^2\right) \quad (13)$$

being γ a user-defined parameter.

4.4.1. Tuning SVR

The R package `e1071` (Dimitriadou et al., 2009), contains the `svm()` function that is mostly programmed in R but uses the optimizers found in *libsvm* (Chang and Lin, 2001), which provide a very efficient C++ version of the sequential minimization optimization (SMO). SVM is an excellent tool for classification, novelty detection, and regression.

In our comparative study, we have focused on variants of the γ parameter of the RBF kernel (`gamma` parameter in the `svm()` function). Namely, we have tried the values 0.01, 0.001 and 0.0005. Different values of `cost` parameter, namely values 10, 150 and 200. In summary, we have considered nine variants of SVR models.

4.5. Random forests

Random forests (Breiman, 2001) are formed by an ensemble of tree-based models (Breiman et al., 1984). They can be used for classification tasks in which the base models are classification trees, or regression tasks where the base models are regression trees. The particularity of random forests when compared to other ensemble strategies lies on the process by which the trees are built. Namely, at each split in a tree within the forest, the test is chosen from a randomly selected sub-set of the independent variables. Moreover, the obtained trees are not pruned.

Random forests have been proving to be outstanding predictive models in many classification and regression tasks. These methods can also be used for estimating the variable importance and also for outlier detection. They are reasonably fast to obtain and can be easily parallelized if more speed is required.

4.5.1. Tuning random forests

The R package `randomForest` (Liaw and Wiener, 2002) implements Breiman's random forest algorithm (based on Breiman and Cutler's original Fortran code) for classification and regression. In our experiments we have considered three variants of the parameter `ntree`, which controls the number of trees within the ensem-

bles. For this parameter we have considered the values 250, 500 and 750.

4.6. Weighted pattern-based model for water demand forecasting

The final option that we have considered in our experiments was a simple and heuristic model based on observations from our exploratory analysis presented in Section 3.1. The main objective of this simple model is to serve as a kind of baseline from which we will compare the other more sophisticated models presented in the previous sections. This method is based on the pattern of demand, namely, its seasonal properties. There are some relationships with the method proposed by Alvisi et al. (2007). The basic idea is to explore the similarities in the hourly demand that were observed previously (c.f. Fig. 1), for each of the days of the week. Namely, a first term of the model includes the weighed average of the water demand for the hour and day of week for which a prediction is required. This average is calculated using past values of the water demand for the same hour and day of the week. This average value is then adjusted by a second term of the model in order to correct it for some specificities of the day in question (e.g. special weather conditions). This adjustment is carried out again by a weighted average of the errors committed by the strategy of the first term in the most recent hours for which a prediction was carried out. This means, for instance, that if today the water demand is consistently above what was on the past for the same hours and day of the week (which is captured by the average on the first term), then there will be a consistent error of this first term under-estimating the water demand. This error will then enter in the second term of the model so as to compensate the first term for the specificity of the current day.

From a theoretical point of view, this proposal can be seen as sharing some ideas with partially linear models (Härdle et al., 2000). As partial linear models, this proposal contains two components: an initial part that reflects the typical behavior of the system; and a second part that corrects/adjusts this initial forecast to account for the specifics of the context for which a prediction is being obtained.

Formally, our the prediction of this model for the water demand at time $i+1$ is given by,

$$W_{i+1} = f_1(\{W_{i+1-k\times 24\times 7}\}_{k=0}^{p}) + f_2(\{e_{i-l}\}_{l=0}^{q}) \quad (14)$$

where p and q are two parameters controlling the amount of memory the model has, while $f_1()$ and $f_2()$ are two averaging functions that can either be standard averages or any form of weighted average that gives for instance more relevance to more recent values.

The e_i values in the model are calculated as follows,

$$e_i = f_1(\{W_{i-k\times 24\times 7}\}_{k=0}^{p}) - W_i \quad (15)$$

In our experiments we have tried several values for the different parameters of the model. Namely, we have used for the averaging functions the median of the values and also an exponential averaging function. For the value of p we have also used the maximum value allowed by the training set size, i.e. we calculate the average using all water demand values of the same day and hour. Finally, for the value of q we have tried the values 10, 18 and 24.

5. Experimental study

In this section we present the results of a series of experiments using the data presented in Section 3, namely for the predictive task described in Section 3.2. The main objective of these experiments is to compare different variants of the models described in Section 4 on the task of forecasting the next hour water demand (W_{i+1}) for our case study.

Given the time dependency between the observed values of water demand, we have been particularly careful with the experimental methodology we have used to estimate the predictive performance of the different models. In effect, for this type of data, estimation processes that involve a random resampling step that changes the order of the observations, provide biased estimates of the predictive performance. These processes are based on resampling procedures that change the (time) order of the data due to their random nature. This may lead to models being obtained with data that is more recent (in terms of the time line) than the data on which they are tested. As the basic assumption of our prediction task is that it is possible to predict the future water demand looking at the past patterns of this variable, the use of the models with these setups will inevitably lead to over-optimistic estimates of the predictive performance of the models (Torgo, 2010). In this context, we have followed a different estimation procedure in our experiments. Namely, we have used a Monte Carlo simulation designed to estimate the predictive performance of a model obtained on a set of data $D_{i\ldots j}$, on another set of data occurring later, i.e. $D_{m\ldots n}$, with $m = j+1$. Giannella et al. (2003) and Lin et al. (2005) are two good references in the evaluation of models based on time granularity. Full details on the used procedure are given on the next sections.

5.1. Monte Carlo estimates

Monte Carlo methods are a class of computational algorithms that can be used, among others, to obtain estimates of any variable by random repetition of a simulation experiment. In our concrete application we are interested in estimating the value of certain predictive performance evaluation statistics (c.f. Section 5.2) that should be measured on a set of test data using models obtained on another set of data that occurred before in time.

In our experiments we have used the following experimental methodology to obtain our estimates. We start by randomly selecting a point in time, t, within the period for which we have data available.[1] We then use the previous $t-1344$ data points to construct a training set with which the candidate models will be obtained. This means that we will use a training window of 8 weeks (1344 h) of data. The obtained models will then be tested on the subsequent 2 weeks, i.e. till point $t+336$. This evaluation process is repeated 20 times to obtain 20 different estimates of the evaluation statistics for each model, each obtained at different points t in time to ensure a certain robustness of the estimates. The Monte Carlo estimates are obtained by averaging these 20 point estimates. To test the statistical significance of the observed differences among these averages we have carried out paired non-parametric Wilcoxon tests (cfr. Table 1).

5.2. The evaluation statistics

In terms of measures to assert the predictive accuracy of our candidate models we have selected the following,

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(O_i - P_i)^2} \quad (16)$$

where n is the number of test cases, O_i the true value of the target variable for the case i, and P_i the respective prediction of the model for the same case, and

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|O_i - P_i| \quad (17)$$

[1] We actually have to reserve some initial and final periods of the data, to ensure that there is enough back data for training the models and also enough future data for testing purposes. Moreover, we have reserved the final week of our data set for our final out of the sample test of the best model according to the Monte Carlo estimates.

Table 1
Difference between the best growing and sliding approaches.

		wPatt	NNET	MARS	RF	SVR	PPR
First	Strat.	Slide	Slide	Grow	Slide	Slide	Slide
	Up.Freq.	1	12	1	1	1	1
	MAE	9.31	6.24	4.34	4.37	4.33	4.34
Second	Strat.	Grow	Grow	Slide	Grow	Grow	Grow
	Up.Freq.	1	24	1	1	1	1
	MAE	9.62	6.3	4.37	4.37	4.33	4.34
	Signif.	++		+			

Both these metrics are expressed in the same units as the target variable. We have also used two non-dimensional evaluation metrics: the Nash–Sutcliffe efficiency (Eq. (18)), and a modification of Nash–Sutcliffe (Eq. (19)), which are more sensitive to systematic errors (Krause et al., 2005), and so may offer more information on the systematic and dynamic errors present in the models.

$$E_j = \frac{\sum_{i=1}^{n} |O_i - P_i|^j}{\sum_{i=1}^{n} |O_i - \overline{O}|^j} \tag{18}$$

$$d_j = \frac{\sum_{i=1}^{n} |O_i - P_i|^j}{\sum_{i=1}^{n} (|P_i - \overline{O}| + |O_i - \overline{O}|)^j} \tag{19}$$

with $j \in \aleph$.

The range of these criteria lies between 0 (perfect fit) and ∞. An efficiency of more than one indicates that the mean (median) value of the observed time series would have been a better predictor than the model. In particular, for $j = 1$, the overestimation of the demand water peaks is reduced significantly – and results in a better overall evaluation (Krause et al., 2005).

5.3. Model building

For each trial of a model in the Monte Carlo process described in Section 5.1, we are given a training set formed by 8 weeks of data and asked to make predictions for the next 2 weeks. In this context, the natural approach would be to obtain a model with the 8 weeks and then use it for obtaining the predictions for all data in the 2 test weeks. This procedure is acceptable in applications where we have no reason to suspect that the dynamics of the target time series varies with time. If that is the case, the model should maintain its accuracy during the 2 weeks. On problems with clear changes in regime (caused by several factors like weather or seasonal effects), this standard approach might not be the most adequate. In Section 3 we have observed that our target problem has clear changes of regime along the time. In this context we follow different approaches to obtain the models.

The growing window strategy (see top graph in Fig. 4) consists in obtaining the model using the initially available data (8 weeks) and then use it for making predictions during a limited time window. When this window ends the test cases within this period are added to the initially available data and a new model is obtained with this larger data set. This new model is used to obtain predictions for another limited time window of the same size, and so on, till predictions for all test period are obtained. This means that this model building algorithm in effect develops several models (with an increasing number of observations), to obtain predictions for the test period. In the limit each time a single prediction is made, a new model is obtained.

The sliding window strategy (bottom graph in Fig. 4) again uses several models, with each model being used for predictions only during a limited time window. The difference for the growing window strategy lies in the fact that all models are obtained with a training window of the same size. So fresher data is added but at the same time the older data is being removed at the same proportion so that a constant size training window of 8 weeks is used to obtain all models. The motivation is to obtain models using only the most recent data.

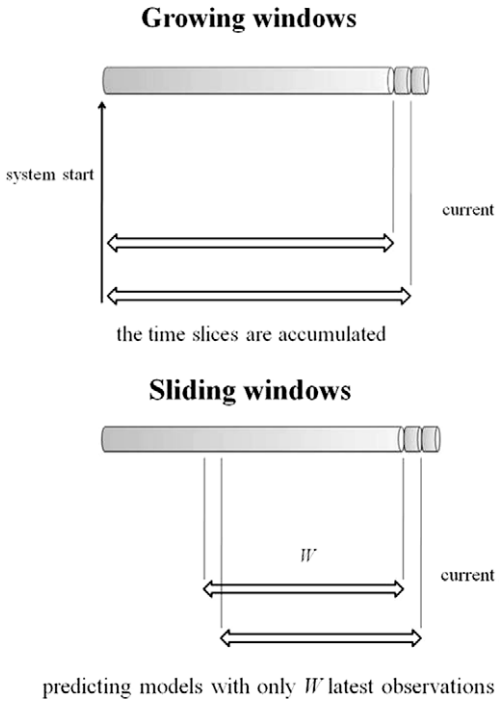

Fig. 4. Two different approaches to model building for time series prediction.

In our experimental comparisons every model variant was obtained using different model building strategies. Namely, we have experimented with three different growing window approaches with the size of the test window for which the model is kept the same set to 1, 12 or 24, i.e. 1 h, half day or a full day. With sliding windows we have followed an equal approach in terms of timings for model updating, but for these methods the training window was always kept with the same size (8 weeks).

5.4. Results of the Monte Carlo comparisons

We first analyze the results of the Monte Carlo experimental comparison of all model variants described in Section 4. We provide a summary of our experimental results in the next sections looking at them from different perspectives.

5.4.1. Comparing the model building strategies

In this section we address the question of which of the six different model building strategies that were considered, is more adequate for each of the modeling techniques. The six variants differ on the algorithm used (growing or sliding) and on the model updating frequency (1, 12 or 24 h).

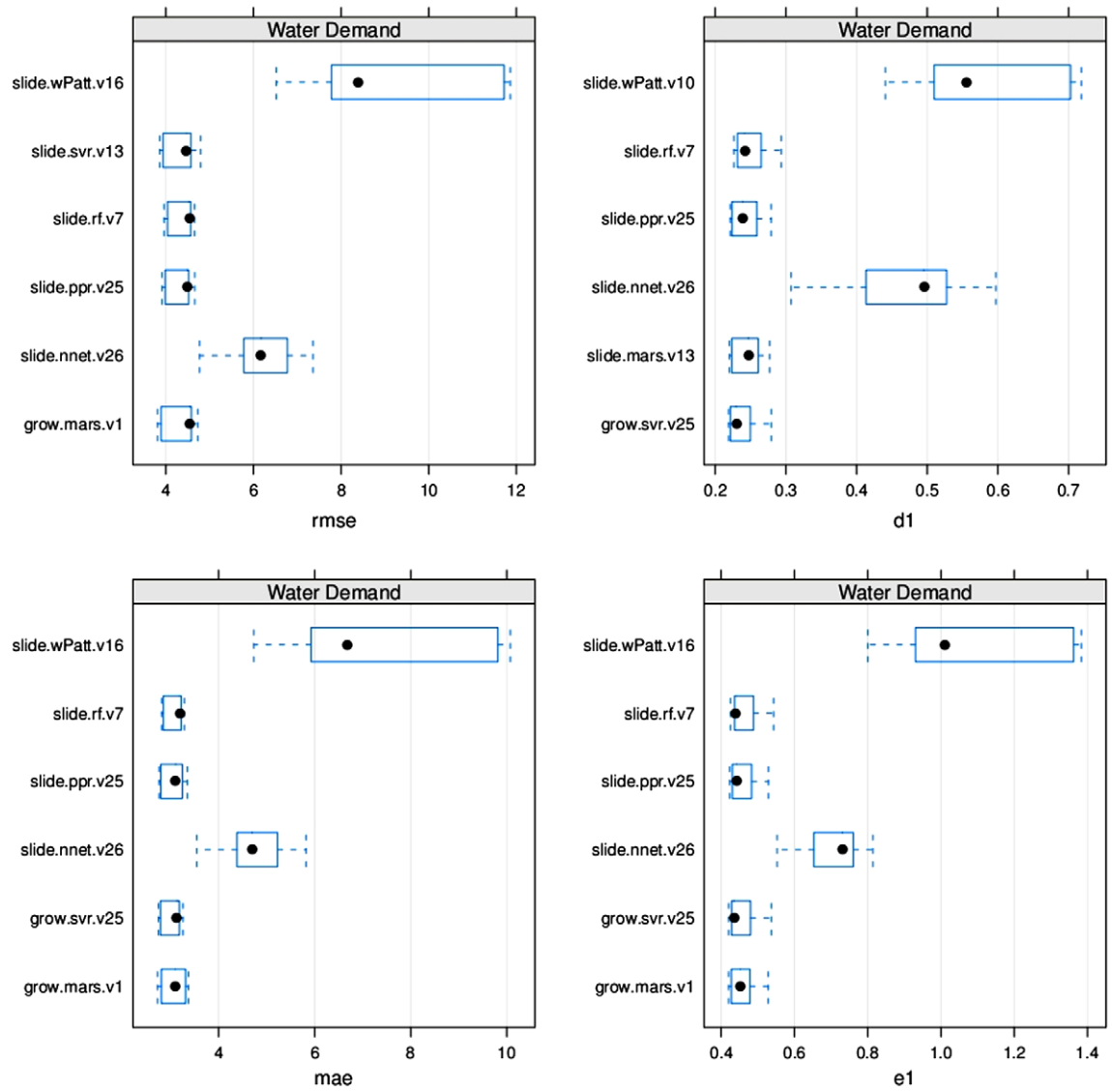

Fig. 5. The best model variants for each evaluation statistic.

Table 1 shows a comparison in terms of *MAE* of the two top models for each of the building strategies across all learning algorithms. The final line indicates the statistical significance of the difference between the two best variants. Two signs represent a difference significant with the confidence level of 0.01, one sign represents a level of 0.05, while no sign represents lowest levels of significance.

Only two of the observed differences are statistically significant. Nevertheless, we can observe that with a single exception (MARS) all models obtain the best score using a sliding window approach.[2] Moreover, again with a single exception (NNET) all best scores are achieved using a model updating frequency of 1 h. This is a clear indication of the existence of fast regime changes in the water demand time series of our problem.

In summary, the model building algorithm only seems relevant for the weighted pattern-based and MARS algorithms, although with opposite conclusions regards which is the best. On the contrary, with respect to the model updating frequency, our results consistently indicate advantages on using a fast pace, with the best results almost always being obtained for 1 h.

5.4.2. The best model variants for each evaluation metric

We now consider the question of which are the best variants in terms of the four used evaluation metrics. Fig. 5 includes the distribution of the values on these statistics obtained by the best model variants of each algorithm, on the 20 repetitions of the Monte Carlo experiment. This distribution is presented by means of box plots.

The results on this figure show a group of models (SVR, Random Forests, PPR and MARS) as clearly better than both NNET and the weighted pattern-based model. Moreover, these differences are statistically significant with 99% confidence. It is also interesting to remark that these top models achieve rather consistent results across the 20 repetitions as it can be seen by the tight boxes in the box plots.

Table 2 shows the top models on each statistic according to the Monte Carlo estimates. They are all variants of the SVR algorithm.

In terms of statistical significance, the difference between the top models shown on Table 2 and the remaining best model variants shown in Fig. 5 is most of the times significant with a confidence of 99%. The exceptions are the following. For the *MAE*

[2] Some results seem equal due to rounding effects, but only for SVR that really happens.

Table 2
The best overall results.

Stat.	Score	Model ID	Model parameters
MAE	3.0284422	grow.svr.v25	GrowWind.; Upd.Freq. = 1; Cost = 200; Gamma = 0.0005
RMSE	4.3314537	slide.svr.v13	SlideWind.; Upd.Freq. = 1; Cost = 150; Gamma = 0.001
E1	0.4537398	grow.svr.v25	GrowWind.; Upd.Freq. = 1; Cost = 200; Gamma = 0.0005
d1	0.2365621	grow.svr.v25	GrowWind.; Upd.Freq. = 1; Cost = 200; Gamma = 0.0005

Table 3
The results of the best model for the final week of data.

Train size	*MAE*	*RMSE*	*e*1	*d*1
8 weeks	3.2555276	4.3831834	0.5551210	0.2938535
All data	3.3156837	4.4407355	0.5666435	0.2999423

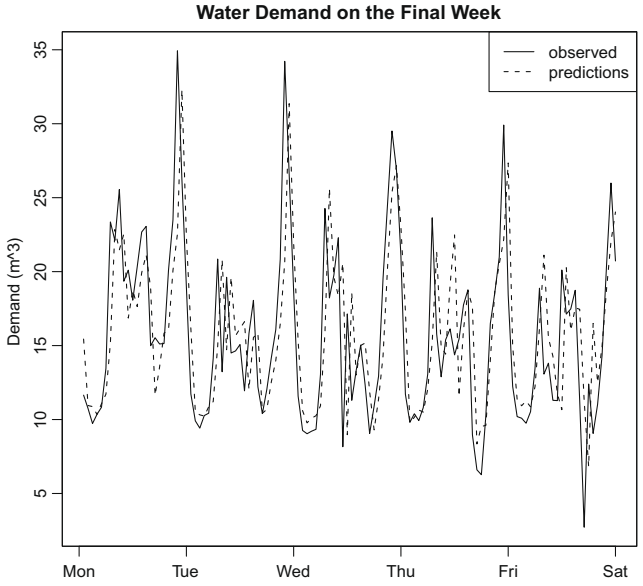

Fig. 6. Difference between forecasted and observed water demand values for the last week.

statistic there is lack of statistical significance to the scores obtained by the two MARS variants and also the PPR variant. With respect to the *RMSE* metric the results of the best model are only significantly different from the results of the weighted pattern-based models and the NNET. In terms of the *e*1 statistic the conclusion is the same as for *MAE*. Finally, for the *d*1 metric the differences are always statistically significant (although only with 95% confidence with respect to the other SVR variant).

Summarizing our results we can conclude that the SVR model seems clearly more adequate to this problem of forecasting the next hour water demand. Still, competitive results are obtained by MARS and PPR on this problem, as well as Random Forests.

5.5. Using the best model

We have intentionally left the last week of data out of our Monte Carlo experimental comparison. The goal is to use the model selected as the best in our comparisons to obtain predictions for this last week. This section describes this small experiment with the "grow.svr.v25" model.

Using again the previous 8 weeks of data (1344 h) we have obtained a SVR model with the parameters *cost* = 200 and *gamma* = 0.0005. Then, using a growing window approach with an updating frequency of 1 h we have obtained the predictions of these models for the last week of data. In terms of the evaluation statistics the results are shown on the first line of Table 3. These scores are slightly worse than the estimates obtained by the Monte Carlo estimation method. Still, they are comparable. We then carried out the same experiment but instead of using only the 8 previous weeks of data, we have decided to include all previous data available. The results of this model are shown on the second line of the same table. As we may observe, the inclusion of more data made the results worse. This again confirms the existence of clear regime shifts on this data set, which makes the inclusion of too old data harmful for the prediction models. This result also provides evidence of the utility of the model building strategies we have used in our study.

On Fig. 6 we plot the true values of the demand for this last week and also the predictions of the best model according to our experimental comparison. With few exceptions the predicted values seem to follow the trend of the observed water demand.

6. Conclusions

In this work we have addressed the task of trying to predict the future water demand on an urban area of a city in south-eastern Spain. We have considered a large number of alternative machine learning methods to solve this prediction task. We have briefly described these techniques and carried out a through experimental comparison of several variants of these models using our case study data. The results of this comparison have identified support vector regression models (SVR) as the most accurate models, closely followed by MARS, PPR and Random Forests. The experiments have also revealed a disappointing performance of the variants of neural networks that were considered. Finally, a heuristic model based on the empirical analysis of the regularities of the time series has shown its limitations when compared to these more sophisticated modelling approaches.

This paper has also addressed the problem of correctly estimating the prediction performance of different models when facing time series data with clear changes of dynamics across the time. We have used a Monte Carlo simulation experiment that ensures unbiased estimates of the selected evaluation metrics, and have also considered and compared different model building strategies to handle the regime shifts of the data. Our experiments have confirmed the advantages of these learning approaches and have identified that little differences exist between the performance of growing and sliding windows. However, for this data set, we have also found that the model updating frequency should be as high as possible (in our case every model is updated as new data is arriving on a hourly basis). In addition, when feeding the EPANET calibrated model with the obtained results, it can be shown that the hydraulic targets listed in Section 1 are clearly met.

All our analysis was carried out using the R programming language environment (free software from R Development Core Team, 2009). This environment provides high quality tools and moreover, being free open source software, ensures an easy replication of our results to other researchers.

Further work in this line of study will include improving the weighted pattern-based method, perhaps with the inclusion of different time support partitions. We also plan to develop a software tool to help in incorporating these modelling tools in a production environment that can help in the water supply management.

Acknowledgments

This work has been supported with the support of the project IDAWAS, DPI2009-11591, of the Dirección General de Investigación of the Ministerio de Ciencia e Innovación of Spain and by Project MORWAQ (PTDC/EIA/68489/2006) financed by the Portuguese Science Foudantion (FCT).

We wish to acknowledge help received from "Aguas de Murcia" in providing data. The revision of this paper was funded by the Universidad Politécnica de Valencia, Spain.

References

Alvisi, S., Franchini, M., Marinelli, A., 2007. A short-term, pattern-based model for water-demand forecasting. Journal of Hydroinformatics 9 (1), 39–50.

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

Arbués, F., García-Valiñas, M.A., Martínez-Espiñeira, R., 2003. Estimation of residential water demand: a estate-of-the-art review. The Journal of Socio-Economics 32, 81–102.

Bishop, C.M., 2005. Neural Networks for Pattern Recognition. Oxford University Press.

Bougadis, J., Adamowski, K., Diduch, R., 2005. Short-term municipal water demand forecasting. Hydrological Processes 19, 137–148.

Breiman, L., 2001. Random forests. Machine Learning 45, 5–32.

Breiman, L., Friedman, J., Olshen, R., Stone, C., 1984. Classification and Regression Trees. Statistics/Probability Series. Wadsworth & Brooks/Cole Advanced Books & Software.

Canu, S., Grandvalet, Y., Rakotomamonjy, A., 2003. SVM and Kernel Methods Matlab Toolbox. Perception Systemes et Information. <http://asi.insa-rouen.fr/arakotom/toolbox/index> (accessed on October 2008).

Chang, C., Lin, C., 2001. Libsvm: A Library for Support Vector Machines. <http://www.csie.ntu.edu.tw/cjlin/libsvm> (accessed on October 2008).

Chen, L., Zhang, T., 2006a. Hourly water demand forecast model based on bayesian least squares support vector machine. Journal of Tianjin University Science and Technology 39 (9), 1037–1042.

Chen, L., Zhang, T., 2006b. Hourly water demand forecast model based on least squares support vector machine. Journal of Tianjin University Science and Technology 38 (9), 1528–1530.

Dahl, C., Hylleberg, S., 2004. Flexible regression models and relative forecast performance. International Journal of Forecasting 20, 201–217.

Dimitriadou, E., Hornik, K., Leisch, F., Meyer, D., Weingessel, A., 2009. e1071: Misc Functions of the Department of Statistics (e1071), TU Wien. R Package Version 1.5-19.

Durbin, J., Koopman, S., 2001. Time Series Analysis by State Space Methods. Oxford University Press.

Friedman, J., Stueltze, W., 1981. Projection pursuit regression. Journal of American Statistical Association 76, 817–823.

Friedman, J., 1991. Multivariate adaptive regression splines (with discussion). Annals of Statistics 19 (1), 1–141.

Gato, S., Jayasuriya, N., Roberts, P., 2007. Temperature and rainfall thresholds for base use urban water demand modelling. Journal of Hydrology 337 (3–4), 364–376.

Giannella, C., Han, J., Pei, J., Yan, X., Yu, P., 2003. Mining frequent patterns in data streams at multiple time granularities. In: Kargupta, H., Joshi, A., Sivakumar, K., Yesha, Y. (Eds.).

Gilbert, P., 2006. Brief user's Guide: Dynamic System Estimation (DSE). R Foundation for Statistical Computing. <http://cran.r-project.org> (accessed on October 2008).

Hastie, T., Tibshirani, R., 1990. Generalized Additive Models. Chapman & Hall.

Härdle, W., Liang, H., Gao, J., 2000. Partially linear models. Physica-Verlag Heidelberg.

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

Jain, A., Varshney, K., Joshi, U., 2001. Short-term water demand forecast modelling at IIT Kanpur using artificial neural networks. IEE Transactions on Water Resources Management 15 (1), 299–321.

Karatzoglou, A., 2006. Kernel Methods Software, Algorithms and Applications – Ph.D. Dissertation. Ph.D. Thesis, Technischen Universitat Wien, Austria.

Karatzoglou, A., Meyer, D., Hornik, K., 2006. Support Vector Machines in R. Journal of Statistical Software 15(9). <http://www.jstatsoft.org/v15/i09> (accessed on January 2009).

Khan, M., Coulibaly, P., 2006. Application of support vector machine in lake water level prediction. Journal of Hydrological Engineering 11 (3), 199–205.

Kim, S., Koo, J., Kim, H., Choi, Y., 2007. Optimisation of pumping schedule based on forecasting hourly water demand in seoul. Water Science & Technology: Water Supply 7, 85–93.

Krause, P., Boyle, D., Bäse, F., 2005. Comparison of different efficiency criteria for hydrological model assessment. Advances in Geosciences 5, 89–97.

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.

Liaw, A., Wiener, M., 2002. Classification and regression by randomforest. R News 2 (3), 18–22. <http://CRAN.R-project.org/doc/Rnews/>.

Lin, H., Chiu, D., Wu, Y., Chen, A., 2005. Mining frequent itemset from data streams with a time-sensitive sliding window. In: Proceedings of the 2005 SIAM International Conference on Data Mining.

Lingireddy, S., Ormsbee, L.E., 1973. Neural Networks in Optimal Calibration of Water Distribution Systems, vol. 3. Flood and N. Kartam (ASCE).

Maidment, D., Miaou, S., 1986. Daily water use in nine cities. Water Resources Research 22 (6), 845–851.

Maidment, D., Miaou, S., Crawford, M., 1985. Transfer function models of daily urban water use. Water Resources Research 21 (4), 425–432.

Maier, H.R., Dandy, G.C., 2000. Application of artificial neural networks to forecasting of surface water quality variables: issues, applications and challenges. Environmental Modelling and Software, 15.

Milborrow, S., 2009. Earth: Multivariate Adaptive Regression Spline Models, Derived from mda:mars by Trevor Hastie and Rob Tibshirani.

Moisen, G., Frescino, T., 2002. Comparing five modelling techniques for predicting forest characteristics. Ecological Modelling 157 (2), 209–225.

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.

R Development Core Team, 2009. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria, ISBN: 3-900051-07-0.

Rossman, L., 2000. EPANET – User's Manual. United States Environmental Protection Agency (EPA), Cincinnati, OH.

Shawe-Taylor, J., Cristianini, N., 2000. An Introduction to Support Vector Machines. Cambridge University Press.

Shrestha, D., Solomatine, D., 2006. Machines learning approach for estimation of prediction interval for the model output. Neural Networks 19 (2), 225–236.

Smith, J., 1988. A model of daily municipal water use for short-term forecasting. Water Resources Research 24 (2), 201–206.

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

Smola, A.J., Schölkopf, B., 1998. A tutorial on support vector regression. In: NeuroCOLT Technical Report TR-98-030.

Storlie, C., Helton, J., 2008a. Multiple predictor smoothing methods for sensitive analysis: description of techniques. Reliability Engineering & System Safety 93, 28–54.

Storlie, C., Helton, J., 2008b. Multiple predictor smoothing methods for sensitive analysis: example results. Reliability Engineering & System Safety 93, 55–57.

Torgo, L., 2010. Data Mining with R: Learning by Case Studies. CRC Press.

Vapnik, V., 1995. The Nature of Statistical Learning Theory. Springer.

Vapnik, V., 1998. Statistical Learning Theory. John Wiley and Sons.

Venables, W., Ripley, B.D., 2002. Modern Applied Statistics with S, forth ed. Springer (ISBN 0-387-95457-0).

Wagner, M., 1999. VAR Cointegration in VARMA Models, Institute for Advanced Studies.

Wang, W., Gelder, P.V., Vrijling, J.K., Ma, J., 2006. Forecasting daily streamflow using hybrid ann models. Journal of Hydrology 324, 383–399.

Zealand, C.M., Burn, D.H., Simonovic, S., 2005. Short term streamflow using artificial neural networks. Journal of Hydrology 214, 32–48.

Zhang, G., Qi, M., 2005. Neural network forecasting for seasonal and trend time series. European Journal of Operational Research 160, 501–514.

Zhou, S.L., McMahon, T.A., Walton, A., Lewis, J., 2002. Forecasting operational demand for an urban water supply zone. Journal of Hydrology 259, 189–202.