CMS (2006) 3:113–129
DOI 10.1007/s10287-005-0003-7

ORIGINAL PAPER

Fixed-size least squares support vector machines: a large scale application in electrical load forecasting

**Marcelo Espinoza · Johan A.K. Suykens ·
Bart De Moor**

Published online: 16 February 2006

Abstract Based on the Nyström approximation and the primal-dual formulation of the least squares support vector machines, it becomes possible to apply a nonlinear model to a large scale regression problem. This is done by using a sparse approximation of the nonlinear mapping induced by the kernel matrix, with an active selection of support vectors based on quadratic Renyi entropy criteria. The methodology is applied to the case of load forecasting as an example of a real-life large scale problem in industry. The forecasting performance, over ten different load series, shows satisfactory results when the sparse representation is built with less than 3% of the available sample.

Keywords Least squares support vector machines · Nyström approximation · Fixed-size LS-SVM · Kernel based methods · Sparseness · Primal space regression · Load forecasting · Time series

1. Introduction

Large databases available for data analysis are common in industry and business nowadays, e.g. banking, finance, process industry, etc. Building and estimating a model from a large dataset requires the algorithms to handle large datasets directly. In this paper we illustrate the performance of a large-scale kernel-based nonlinear regression technique, the Fixed-Size

* This work is supported by grants and projects for the Research Council K.U.Leuven (GOA-Mefisto 666, GOA-Ambiorics, several PhD/ Postdocs & fellow grants), the Flemish Government (FWO: PhD/ Postdocs grants, projects G.0211.05, G.0240.99, G.0407.02, G.0197.02, G.0141.03, G.0491.03, G.0120.03, G.0452.04, G.0499.04, ICCoS, ANMMM; AWI; IWT: PhD grants, GBOU (McKnow) Soft4s), the Belgian Federal Government (Belgian Federal Science Policy Office: IUAP V-22; PODO-II (CP/ 01/40), the EU (FP5- Quprodis; ERNSI, Eureka 2063- Impact; Eureka 2419- FLiTE) and Contracts Research / Agreements (ISMC /IPCOS, Data4s, TML, Elia, LMS, IPCOS, Mastercard). J. Suykens and B. De Moor are an associate professor and a full professor with K.U.Leuven, Belgium, respectively. The scientific responsibility is assumed by its authors.

M. Espinoza (✉) · J.A.K. Suykens · B. De Moor
ESAT/SISTA, Katholieke Universiteit Leuven. Kasteelpark Arenberg 10, 3000 Leuven, Belgium
E-mail: marcelo.espinoza@esat.kuleuven.be

least squares support vector machines (LS-SVM) (Suykens et al. 2002b), on a real-life large-scale modelling problem.

Kernel based estimation techniques, such as support vector machines (SVMs) and LS-SVMs have shown to be powerful nonlinear classification and regression methods (Poggio and Girosi 1990; Vapnik 1998; Williams 1998). Both techniques build a linear model in the so-called feature space where the inputs have been transformed by means of a (possibly infinite dimensional) nonlinear mapping $\boldsymbol{\varphi}$. This is converted to the dual space by means of the Mercer's theorem and the use of a positive definite kernel, without computing explicitly the mapping $\boldsymbol{\varphi}$. The SVM model solves a quadratic programming problem in dual space, obtaining a sparse solution (Cristianini and Shawe-Taylor 2000). The LS-SVM formulation, on the other hand, solves a linear system under a least squares cost function (Suykens and Vandewalle 1999), where the sparseness property can be obtained by sequentially pruning the support value spectrum (Suykens et al. 2002a). The LS-SVM training procedure involves a selection of the kernel parameter and the regularization parameter of the cost function, that usually can be done by cross-validation or by using Bayesian techniques (MacKay 1995). In this way, the solutions of the LS-SVM can be computed using an eventually infinite-dimensional $\boldsymbol{\varphi}$ based on a non-parametric estimation in the dual space.

Solving the LS-SVM in dual formulation requires the resolution of a linear system of dimension N (the number of datapoints). This is practical when working with large dimensional input spaces, or when the dimension of the input space is larger than the sample size. However, there is an obvious drawback when N is too large, and in such case the direct application of this method becomes prohibitive. In this case, the primal-dual structure of the LS-SVM can be exploited further. It is possible to compute an approximation of the nonlinear mapping $\boldsymbol{\varphi}$ in order to perform the estimations directly in primal space; furthermore, it is possible to compute a sparse approximation by using only a subsample of selected support vectors from the dataset. In this case, one can estimate a large scale nonlinear regression problem in primal space.

As an application to an interesting real-life problem, we study the case of the short-term load forecasting problem, which is an important area of quantitative research (Ramanathan et al. 1997; Lotufo and Minussi 1999; Fay et al. 2003; Bunn 2000; Espinoza et al. 2005). Within this context, the goal of the modelling task is to generate a model that can capture all the dynamics and interaction between possible explanatory variables to explain the behavior of the load in an hourly scale. Usually a load series shows important seasonal patterns (yearly, weekly, intra-daily patterns) that need to be taken into account in the modelling strategy (Hylleberg 1992). In our case, the data series comes from 10 local low voltage substations in Belgium, with each load series containing approximately 36,000 hourly values. This paper is structured as follows. The description of the LS-SVM is presented in section 2. In section 3, the methodology for working in primal space is described, with the particular application to a large scale problem. Section 4 presents the problem and describes the setting for the estimation, and the results are reported in section 5.

2. Function estimation using LS-SVM

The standard framework for LS-SVM estimation is based on a primal-dual formulation. Given the dataset $\{\boldsymbol{x}_i, y_i\}_{i=1}^{N}$ the goal is to estimate a model of the form

$$y_i = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{\varphi}(\boldsymbol{x}_i) + b + e_i, \tag{1}$$

where $\boldsymbol{x} \in \mathbb{R}^n$, $y \in \mathbb{R}$, and $\boldsymbol{\varphi}(\cdot) : \mathbb{R}^n \rightarrow \mathbb{R}^{n_h}$ is the mapping to a high dimensional (and possibly infinite dimensional) feature space, and the error terms e_i are assumed to be i.i.d. with zero mean and constant (and finite) variance.

The following optimization problem with a regularized cost function is formulated:

$$\min_{\boldsymbol{w},b,\boldsymbol{e}} \quad \frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w} + \gamma\frac{1}{2}\sum_{i=1}^{N} e_i^2, \tag{2}$$

$$\text{s.t.} \quad y_i = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{\varphi}(\boldsymbol{x}_i) + b + e_i, \quad i = 1, \ldots, N,$$

where γ is a regularization constant. The solution is formalized in the following lemma.

Lemma 1. *Given a positive definite kernel function $K : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$, the solution to (2) is given by the dual problem*

$$\left[\begin{array}{c|c} 0 & \mathbf{1}^{\mathrm{T}} \\ \hline \mathbf{1} & \boldsymbol{\Omega} + \gamma^{-1}\mathbf{I} \end{array}\right] \left[\begin{array}{c} b \\ \hline \boldsymbol{\alpha} \end{array}\right] = \left[\begin{array}{c} 0 \\ \boldsymbol{y} \end{array}\right], \tag{3}$$

with $\boldsymbol{y} = [y_1, \ldots, y_N]^{\mathrm{T}}, \boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_N]^{\mathrm{T}}$, and $\boldsymbol{\Omega}$ is the kernel matrix with $\boldsymbol{\Omega}_{i,j} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)\ \forall i, j = 1, \ldots, N$.

Proof. Consider the Lagrangian of problem (2)

$$\mathcal{L}(\boldsymbol{w}, b, e; \alpha) = \frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w} + \gamma\frac{1}{2}\sum_{i=1}^{N} e_i^2 - \sum_{i=1}^{N} \alpha_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\varphi}(\boldsymbol{x}_i) + b + e_i - y_i), \tag{4}$$

where $\alpha_i \in \mathbb{R}$ are the Lagrange multipliers. The conditions for optimality are given by

$$\begin{cases} \frac{\partial\mathcal{L}}{\partial\boldsymbol{w}} = 0 \rightarrow \boldsymbol{w} = \sum\limits_{j=1}^{N} \alpha_j\boldsymbol{\varphi}(x_j) \\ \frac{\partial\mathcal{L}}{\partial b} = 0 \rightarrow \sum\limits_{i=1}^{N} \alpha_i = 0 \\ \frac{\partial\mathcal{L}}{\partial e_j} = 0 \rightarrow \alpha_j = \gamma e_j, \quad i = 1, \ldots, N \\ \frac{\partial\mathcal{L}}{\partial\alpha_j} = 0 \rightarrow y_j = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{\varphi}(\boldsymbol{x}_j) + b + e_j, \end{cases} \tag{5}$$

with the application of Mercer's theorem (Vapnik 1998) $\boldsymbol{\varphi}(\boldsymbol{x}_i)^{\mathrm{T}}\boldsymbol{\varphi}(\boldsymbol{x}_j) = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ with a positive definite kernel K, we can eliminate $\boldsymbol{w}$ and e_i, obtaining $y_j = \sum_{i=1}^{N} \alpha_i K(\boldsymbol{x}_i, \boldsymbol{x}_j) + b + \frac{\alpha_j}{\gamma}$. Building the kernel matrix $\boldsymbol{\Omega}_{i,j} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ and writing the equations in matrix notation gives the final system (3) □

The final model is expressed in dual form

$$y(\boldsymbol{x}) = \sum_{i=1}^{N} \alpha_i K(\boldsymbol{x}_i, \boldsymbol{x}) + b. \tag{6}$$

With the application of the Mercer's theorem it is not required to compute explicitly the nonlinear mapping $\boldsymbol{\varphi}(\cdot)$ as this is done implicitly through the use of positive definite kernel functions K. For $K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ there are usually the following choices: $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{x}_j$ (linear kernel); $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{x}_j/c + 1)^d$ (polynomial of degree d, with c a tuning parameter); $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(-||\boldsymbol{x}_i - \boldsymbol{x}_j||_2^2/\sigma^2)$ radial basis function (RBF), where σ is a tuning parameter. Usually the training of the LS-SVM model involves an optimal selection of the tuning parameters σ (kernel parameter) and γ, which can be done using e.g. cross-validation techniques or Bayesian inference (MacKay 1995).

3. Estimation in primal space

In this section, the estimation in primal space is described in terms of the explicit approximation of the nonlinear mapping $\boldsymbol{\varphi}$, and the further implementation for a large scale problem.

3.1. Nyström approximation in primal space

Explicit expressions for $\boldsymbol{\varphi}$ can be obtained by means of an eigenvalue decomposition of the kernel matrix $\boldsymbol{\Omega}$ with entries $K(\boldsymbol{x}, \boldsymbol{x}_j)$. Given the integral equation

$$\int K(\boldsymbol{x}, \boldsymbol{x}_j)\phi_i(\boldsymbol{x})p(\boldsymbol{x})d\boldsymbol{x} = \lambda_i \phi_i(\boldsymbol{x}_j), \tag{7}$$

with solutions λ_i and ϕ_i for a variable $\boldsymbol{x}$ with probability density $p(\boldsymbol{x})$, we can write

$$\boldsymbol{\varphi} = \left[\sqrt{\lambda_1}\phi_1, \sqrt{\lambda_2}\phi_2, \ldots, \sqrt{\lambda_{n_h}}\phi_{n_h}\right]. \tag{8}$$

Given the dataset $\{\boldsymbol{x}_i, y_i\}_{i=1}^N$, it is possible to approximate the integral by a sample average (Williams and Seeger 2000a,b). This will lead to the eigenvalue problem (Nyström approximation)

$$\frac{1}{N}\sum_{k=1}^{N} K(\boldsymbol{x}_k, \boldsymbol{x}_j)u_i(\boldsymbol{x}_k) = \lambda_i^{(s)} u_i(\boldsymbol{x}_j), \tag{9}$$

where the eigenvalues λ_i and eigenfunctions ϕ_i from the continuous problem can be approximated by the sample eigenvalues $\lambda_i^{(s)}$ and eigenvectors u_i as

$$\hat{\lambda}_i = \frac{1}{N}\lambda_i^{(s)}, \quad \hat{\phi}_i = \sqrt{N}u_i. \tag{10}$$

Based on this approximation, it is possible to compute the eigendecomposition of the kernel matrix $\boldsymbol{\Omega}$ and use its eigenvalues and eigenvectors to compute the ith required component of any point $\hat{\boldsymbol{\varphi}}(\boldsymbol{x})$ by means of

$$\hat{\boldsymbol{\varphi}}_i(\boldsymbol{x}) = \frac{N}{\sqrt{\lambda_i^{(s)}}}\sum_{k=1}^{N} u_{ki} K\left(\boldsymbol{x}_k, \boldsymbol{x}^{(v)}\right). \tag{11}$$

This finite dimensional approximation $\hat{\boldsymbol{\varphi}}(\boldsymbol{x})$ can be used in the primal problem (2) to estimate $\boldsymbol{w}$ and b directly.

3.2. Sparse approximations and large scale problems

It is important to emphasize that the use of the entire training sample of size N to compute the approximation of $\boldsymbol{\varphi}$ will produce a vector $\hat{\boldsymbol{\varphi}}(\boldsymbol{x})$ having N components, each one of which can be computed by (11) for all $\boldsymbol{x} \in \{\boldsymbol{x}_i\}_{i=1}^N$. However, for a large scale problem, it has been motivated in (Suykens et al. 2002b) to use of a subsample of $M \ll N$ datapoints to compute $\hat{\boldsymbol{\varphi}}$. In this case, up to M components will be computed. The selection of the subsample of size M, the initial support vectors, is done prior to the estimation of the model, and the final performance of the model can depend on the quality of the initial selection. It is possible to take a random selection of M datapoints and use them to build the approximation of the nonlinear mapping $\boldsymbol{\varphi}$, or it is possible to use a more optimal selection. External criteria such as entropy maximization can be applied for an optimal selection of the subsample. In this case, given a fixed-size M, the aim is to select the support vectors that maximize the quadratic Renyi entropy (Suykens et al. 2002b; Girolami 2003)

$$H_R = -\log \int p(x)^2 dx, \tag{12}$$

that can be approximated by

$$\int \hat{p}(x)^2 dx = \frac{1}{N^2}\mathbf{1}^{\mathrm{T}}\boldsymbol{\Omega}\mathbf{1}. \tag{13}$$

The use of this active selection procedure can be quite important for large scale problems, as it is related to the underlying density distribution of the sample. In this sense, the optimality of this selection is related to the final accuracy that can be obtained in the modelling exercise. It is important to stress out that the difference between the performance of a model having an initial random selection and a model having an initial entropy-based selection will depend on the characteristics of the dataset itself. A rather simple dataset may be well approximated by both methods; whereas in a more complex dataset, the models can have different performances. Intuitively, the initial selection should contain some important regions of the dataset, as it was shown in Espinoza et al. (2003) for the case of the Santa Fe Laser example (Weigend and Gershenfeld 1994).

It is interesting to note that the equation (9) is related to applying kernel PCA in feature space (Shawe-Taylor and Williams 2003). However, in our case the conceptual aim is to obtain a finite approximation of the mapping $\boldsymbol{\varphi}$ on feature space as good as possible. If we use the entire sample of size N, then only equations (11) are to be computed and therefore the components of $\hat{\boldsymbol{\varphi}}$ are directly the eigenvectors of the kernel matrix $\boldsymbol{\Omega}$. In the application of this paper, it is required to define the number M prior to the modelling exercise. Each fixed-size sample will lead to an approximation of the nonlinear mapping for the entire sample of size N.

3.3. Fixed-size LS-SVM

Based on the explicit approximation $\hat{\boldsymbol{\varphi}}$ that can be computed from an initial sample of M datapoints from the given the dataset $\{\boldsymbol{x}_i, y_i\}_{i=1}^N$, the fixed-size LS-SVM (FS-LSSVM) nonlinear regression can be formulated as follows:

$$\min_{\boldsymbol{w},b,\boldsymbol{e}} \quad \frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w} + \gamma\frac{1}{2}\sum_{i=1}^{N} e_i^2 \tag{14}$$

$$\text{s.t.} \quad y_i = \boldsymbol{w}^{\mathrm{T}}\hat{\boldsymbol{\varphi}}(\boldsymbol{x}_i) + b + e_i, \quad i = 1, \ldots, N.$$

where γ is a regularization constant. It is important to stress out that this model is nonlinear in the inputs, but linear in the parameters. Therefore, it can be solved by traditional linear regression techniques.

3.4. Hyperparameters selection

When working in dual space, i.e. without computing explicitly an approximation to the nonlinear mapping, then the solution is given by (3) for a given kernel and regularization constant. Usually the training of the LS-SVM model in dual space involves an optimal selection of the kernel parameter(s) and the regularization term, which can be done using e.g. cross-validation techniques or Bayesian inference (MacKay 1995). However, working with the explicit expression of $\hat{\boldsymbol{\varphi}}$ makes the problem (14) a linear least-squares problem, in which the solution is given by the estimates of $\boldsymbol{w}$ and b for a given kernel. Solving the regression problem (14) can be done with traditional statistical techniques. Using $\gamma > 0$ is equivalent to ridge-regression; using $\gamma = \infty$ is equivalent to ordinary least squares (OLS). Within the linear ridge-regression context, it is known that the use of regularization is motivated by the existence of collinearity between some regressors (Hoerl and Kennard 1970), in which case the small bias produced by the inclusion of a regularization term is compensated with a decrease in the variance of the parameters. However, in the case of fixed-size LS-SVM, the regressors to be included in the linear regression (14) do not show collinearity problems as they are derived from the orthogonal eigenvectors of the small kernel matrix $\boldsymbol{\Omega}_M$. This is the reason why in this paper the model (14) is estimated with OLS, which is equivalent to $\gamma = \infty$, as motivated in Espinoza et al. (2003). For a discussion about the use of a regularization term and its properties in linear regression, the reader is referred to Björkstrom and Sundberg (1999), Frank and Friedman (1993), Hoerl and Kennard (1970), Stone and Brooks (1990), and Sundberg (1993).

3.5. Implementation of the FS-LSSVM

The algorithm for the final implementation can be described through the following steps:

1. Consider the dataset $\{\boldsymbol{x}_i, y_i\}_{i=1}^N$
2. Select a subsample of size M of the training points $\{\boldsymbol{x}_i\}_{i=1}^N$ using maximization of the quadratic Renyi entropy (13)
3. Use the selected subsample of size M to build a small kernel matrix $\boldsymbol{\Omega}_M$
4. Compute the eigenvectors u_i and eigenvalues $\lambda_i^{(s)}$ of $\boldsymbol{\Omega}_M$
5. Compute the approximation of the nonlinear mapping $\hat{\boldsymbol{\varphi}}(\boldsymbol{x}_i)$ using (11) for all points $i = 1, \ldots, N$
6. Solve the linear least-squares regression problem (14)

4. Practical example: short-term load forecasting

In this section, the practical application is described, in term of the problem context, methodological issues and results.

4.1. Description and objective

Our objective is to perform the application of the fixed-size LS-SVM technique to the real-life problem of short-term load forecasting. The modelling and forecasting of the load is currently an important area of quantitative research. In order to deal with the everyday process of planning, scheduling and unit-commitment, the need for accurate short-term forecasts has led to the development of a wide range of models based on different techniques. Some interesting examples are related to periodic time series (Espinoza et al. 2005), traditional time series analysis (Ramanathan et al. 1997), neural networks applications (Steinherz et al. 2001), and Support Vector Machines (Chen et al. 2002). The main goal is to generate a model that can capture all the dynamics and interactions between possible explanatory variables for the load. Short-term is usually referred to one hour ahead, up to one day ahead, and it is a task that is used on a daily basis on every major dispatch center or by grid managers (Mariani and Murthy 1997). For this task, there is a broad consensus about possible explanatory variables: past values of the load, weather information, calendar information, and possibly some

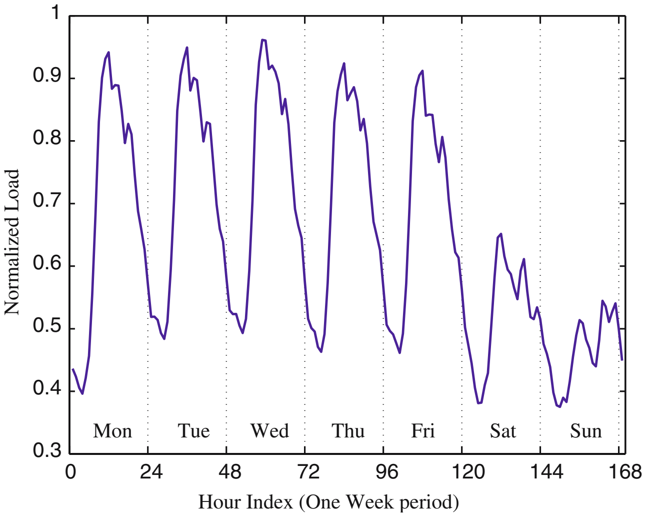

Fig. 1 Example of a load series within a week. Daily cycles are visible, as well as the weekend effects. Also visible are the intra-day phenomena, such as the peaks (morning, noon, evening) and the night hours

past-errors correction mechanisms. Forecasting the load is not straightforward, particularly due to the presence of multiple seasonal patterns in the load series (monthly, weekly, intra-daily). Figure 1 shows an example of a load series in a week, at hourly values starting at 00:00 hours on Monday, until 24:00 hours on Sunday. In the literature, it is often found that some local models of the load are used to produce short-term forecasts; the local models are selected in order to isolate a seasonal pattern (working only with winter, summer, evenings, working-days, etc).

4.2. Data and methodology

The dataset consists of ten time series, each containing hourly load values from a HV-LV substation within the Belgian grid, for a period of approximately 5 years (from January 1998 until September 2002). The ten load series differ in their behavior as they represent different types of underlying customers (residential, business, industrial, etc.). We use a sample of 1500 days (36,000 h) for training the models. A first linear regression containing only a linear trend is estimated for each substation, to remove any growth trend present in the sample. Finally, the series were normalized using the maximum observed value in order to scale all the series to a range between 0 and 1.

The nonlinear model formulation to be used is a nonlinear ARX specification, with the following structure:

- An autoregressive part of 48 lagged load values (i.e. the last 2 days) (Espinoza et al. 2005).
- Temperature-related variables measuring the effect of temperature on cooling and heating requirements (Engle et al. 1986).
- Calendar information in the form of dummy variables for month of the year, day of the week and hour of the day (Espinoza et al. 2005).

This leads to a set of 97 explanatory variables. To illustrate the technique outlined in the algorithm described in the previous section, we use different sizes for the initial subsample.

Each time, the RBF kernel function is used. As indicated previously, the linear least-squares problem (14) is solved by OLS, thus considering $\gamma = \infty$. Tuning of the hyperparameter σ is performed by tenfold cross validation in the training sample. We keep the value of σ that minimizes the out-of-sample mean squared error (MSE).

To illustrate the effect of increasing sizes of M, the above methodology is tested for sizes of M=200, 400, 600, 800 and M=1,000 support vectors, selected with the quadratic entropy criterion. It is important to stress out that we are using between 0.5 and 3% of the dataset to build the nonlinear mapping for the entire sample. In this sense, we see that it is possible to build a nonlinear forecasting model using the fixed-size LS-SVM using a relatively few datapoints as support vectors, which in turn depends on the data generating process (DGP) underlying the current sample. Values of M larger than 1,000 are possible, which will finally depend on the computer capacity to store a $M \times M$ matrix in memory and perform its eigendecomposition. In practice, there is a trade-off between accuracy of the model and computational time (and storage resources), and for our purposes $M = 1{,}000$ is the largest number of support vectors being tested.

The fixed-size LS-SVM is compared with a linear model Verbeek (2000) and Ljung (1999) estimated with the same initial set of variables. In addition, a traditional LS-SVM is estimated using only the last 1,000 datapoints on the sample. In this way, it is possible to compare the difference in performance between two nonlinear models in the following two cases: when the full sample is taken into account (fixed-size LS-SVM) or only when the most recent 1,000 h (last 42 days) are considered.

The forecasting performance is assessed as follows. The simplest scheme is to forecast the first out-of-sample load value using all information available, then wait 1-h until the true value of this forecast has been observed, and then forecast the next value again using all available information (1-h-ahead prediction). However, planning engineers require forecasts with a longer time horizon, at least a full day in advance. In this case, it is required to predict the first out-of-sample value using all the working sample, then predict the second value out-of-sample using this first prediction, and so on (iterative simulation). In practice, it is reasonable to stop this iterative process after 24-h and update the information with actual observations. The methods are compared on a test data set (not using during training/estimation) that consists of 15 d after the last sample point. The performance is assessed via the MSE for the one-step-ahead prediction and the 24-h-ahead-simulation with updates at 00:00 hours of each day.

5. Results

In this section, the results of the fixed-size LS-SVM methodology applied to the load modelling problem are reported, for the training procedure, the selection of support vectors and the out of sample performance.

5.1. Training performance

The above procedure is applied for M=200,400,600,800 and M=1,000. Training using tenfold crossvalidation is performed for each case, looking for an optimal value of the hyperparameter σ in the RBF kernel. Figure 2 shows the evolution of the MSE in the tenfold crossvalidation training procedure for the cases of $M = 200$ and $M = 400$ in one of the load series, where it can be seen that the optimal value is $\sigma = 2.01$. For the cases $M = 600$, $M = 800$, and M=1,000 we perform the crossvalidation process using only the selected σ. The results for

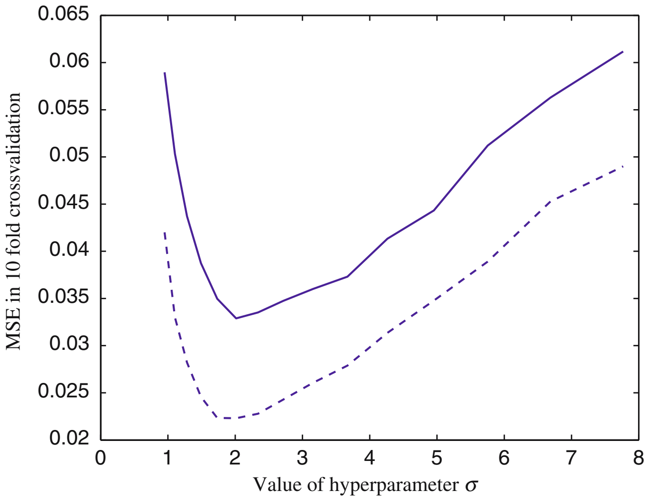

Fig. 2 Performance evolution in the training procedure. The lines show the evolution of the MSE in a tenfold crossvalidation for the cases $M = 200$ (*full line*) and $M = 400$ (*dashed line*). The optimal value for the σ hyperparameter is 2.01

Table 1 Performance of the fixed-size LS-SVM models

Estimation	Mean squared error (CV)
Linear	0.043
M=200	0.032
M=400	0.022
M=600	0.017
M=800	0.016
M=1000	0.015

The nonlinear mapping approximation has been built with M support vectors, on a crossvalidation basis using the optimal σ

the computed MSE in a crossvalidation basis, and the equivalent result for the linear model, are shown in Table 1 using the selected σ.

Table 1 shows that there is a marginal improvement on accuracy when increasing the number of support vectors from $M = 800$ to M=1,000. This suggests that the current problem can be modelled based on a subsample of M=1,000 support vectors, which may not be the case for other regression problems. In principle, the number of required support vectors is related to the DGP underlying the current sample, and they are not necessarily related to the size of the sample. Intuitively, the data generated by a more complex phenomena will require more support vectors than a more simple one, regardless of the available sample size. An example of fixed-size LS-SVM on another large scale problem, in the context of nonlinear system identification, is detailed in Espinoza et al. (2004) in which a model trained with $M = 500$ was the best entry on the benchmark study of nonlinear identification techniques based on a dataset of 130,000 points (thus, using only 0.4% of the available sample). However, a definition of the optimal number M based on theoretical considerations (not only practical one) is still under research.

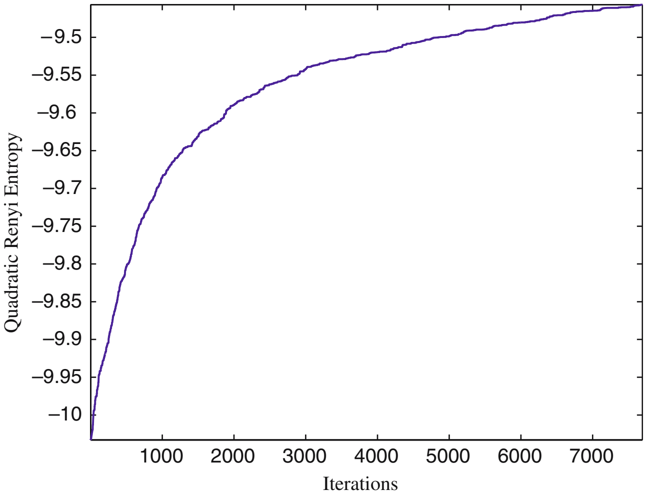

Fig. 3 The evolution of the quadratic Renyi entropy within the iterative search for support vectors for the case $M = 400$

5.2. Support vector selection

The initial set of M support vectors has been selected by maximizing the quadratic Renyi entropy. Starting from a random sample of size M, it is possible to replace elements of the selected sample by elements of the remaining sample if the entropy is maximized, and iterate this procedure until convergence. In this way, it is possible to obtain a selection of those M points that converge to a maximum value of the quadratic entropy. Figure 3 shows the evolution of the entropy value within this iterative process (for a selected load series), for the $M = 400$ case. Figure 4 shows the position (time index) of the first element of the selected support vectors for the $M = 400$ case. It is interesting to see how the selected support vectors are those for which the output series is located in the regions of high load values (winter times), some in the lower values (summer times) and almost none of them in spring seasons. It is also clear that the output at the selected support vectors position is going through some “critical” regions.

5.3. Effect of selection method

It is possible to compare the performance between a model estimated with a random selection of support vectors versus the same model estimated with a quadratic Renyi entropy selection starting from the same random selection. In other words, one can generate a random selection of support vectors and either perform a quadratic entropy selection using the random selection as initial position for the iterations (and later approximate the nonlinear model with the entropy-selected support vectors), or one can just use the random selection for the nonlinear mapping approximation immediately. Both models can be compared in terms of performance on the same test set. Table 2 and Figure 5 show the comparison for the results after 20 random initial selections, in which the model is either estimated after quadratic entropy selection taking the random selection as the initial starting point (Case I), or it is estimated directly (Case II). In all tests it has been used $M = 200$.

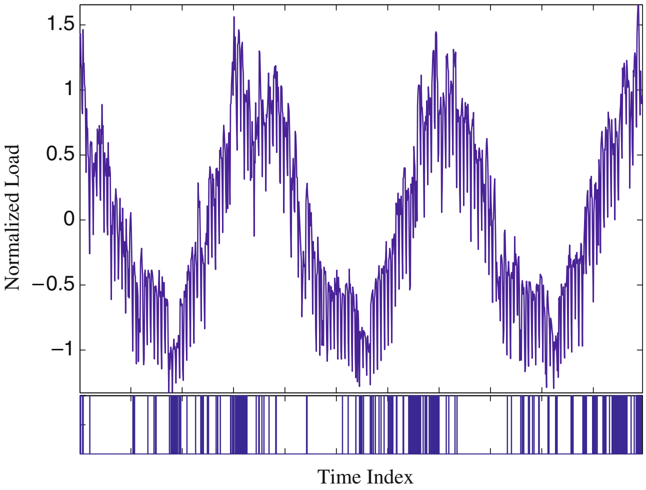

Fig. 4 The normalized load from series 1 used as training sample (*Top*), shown here only as daily averages rather than hourly values. The position of the selected support vectors corresponding to the load sample output is represented by *dark bars* at the *bottom*, showing the time index position of the first element of the support vector

Table 2 Comparison of the mean and standard deviation of the MSE for a test set performance using $M = 200$ over 20 randomizations

Support vector selection	Average MSE	Standard deviation MSE
Entropy-based selection (Case I)	0.0311	0.0016
Random-based selection (Case II)	0.0317	0.0025

Case I refers to the random selection of support vectors and immediate estimation of the model. Case II starts from the same random selection, performs quadratic-based selection using the random sample as starting point, and then the model is estimated

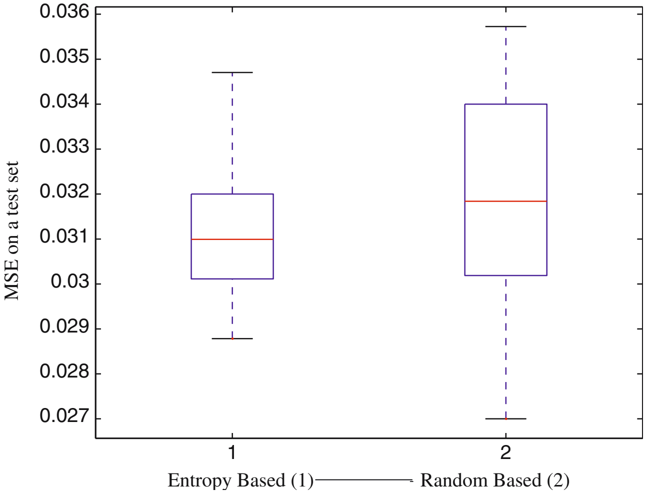

Fig. 5 Box-plot of the MSE in a test set for models estimated with entropy-based (1) and random (2) selection of support vectors. Results for 20 repetitions

Table 3 Model performance on the test set for different forecasting modes, for series 1–5

Series	Mode	Performance	LS-SVM (%)	FS-LSSVM (%)	Linear (%)
1	1-step-ahead	MSE	2.2	0.6	1.4
		MAPE	2.8	1.5	2.5
	24-steps-ahead	MSE	5.0	2.7	9.5
		MAPE	4.3	3.1	5.9
2	1-step-ahead	MSE	3.4	2.3	3.0
		MAPE	4.3	3.4	3.9
	24-steps-ahead	MSE	20.2	11.5	11.9
		MAPE	10.6	7.4	7.9
3	1-step-ahead	MSE	9.7	6.7	10.2
		MAPE	29.4	17.7	24.9
	24-steps-ahead	MSE	15.1	9.4	15.0
		MAPE	30.1	23.1	29.7
4	1-step-ahead	MSE	4.9	4.0	7.4
		MAPE	12.6	10.5	16.2
	24-steps-ahead	MSE	10.1	6.0	14.7
		MAPE	20.7	14.5	22.3
5	1-step-ahead	MSE	2.2	0.9	1.7
		MAPE	2.6	1.7	2.2
	24-steps-ahead	MSE	9.0	3.8	6.7
		MAPE	5.5	3.4	4.4

Table 4 Model performance on the test set for different forecasting modes, for series 6–10

Series	Mode	Performance	LS-SVM (%)	FS-LSSVM (%)	Linear (%)
6	1-step-ahead	MSE	0.8	0.3	1.1
		MAPE	2.3	1.4	2.2
	24-steps-ahead	MSE	3.9	2.6	7.5
		MAPE	5.1	4.4	7.1
7	1-step-ahead	MSE	2.6	1.6	3.0
		MAPE	2.9	2.2	3.1
	24-steps-ahead	MSE	5.7	3.8	6.8
		MAPE	4.5	3.5	4.7
8	1-step-ahead	MSE	2.4	1.5	2.2
		MAPE	3.0	2.4	2.8
	24-steps-ahead	MSE	9.8	5.3	7.7
		MAPE	7.3	4.4	5.3
9	1-step-ahead	MSE	0.9	0.5	1.3
		MAPE	1.8	1.3	2.0
	24-steps-ahead	MSE	3.2	2.1	6.9
		MAPE	3.4	2.8	5.3
10	1-step-ahead	MSE	2.8	2.3	3.5
		MAPE	5.7	4.9	6.0
	24-steps-ahead	MSE	9.9	8.2	12.7
		MAPE	11.0	10.9	13.4

The existence of the standard deviation in Case I accounts for the fact that the convergence of the entropy selection is not unique, specially for a selection of 200 points out of 36,000 possible samples. However, starting from different random selections, the entropy-based selection yields lower dispersion in the errors. For this dataset, and after 20 repetitions, the average MSE are quite similar, but there is no guarantee that the random-selection will perform like this for a more complex dataset.

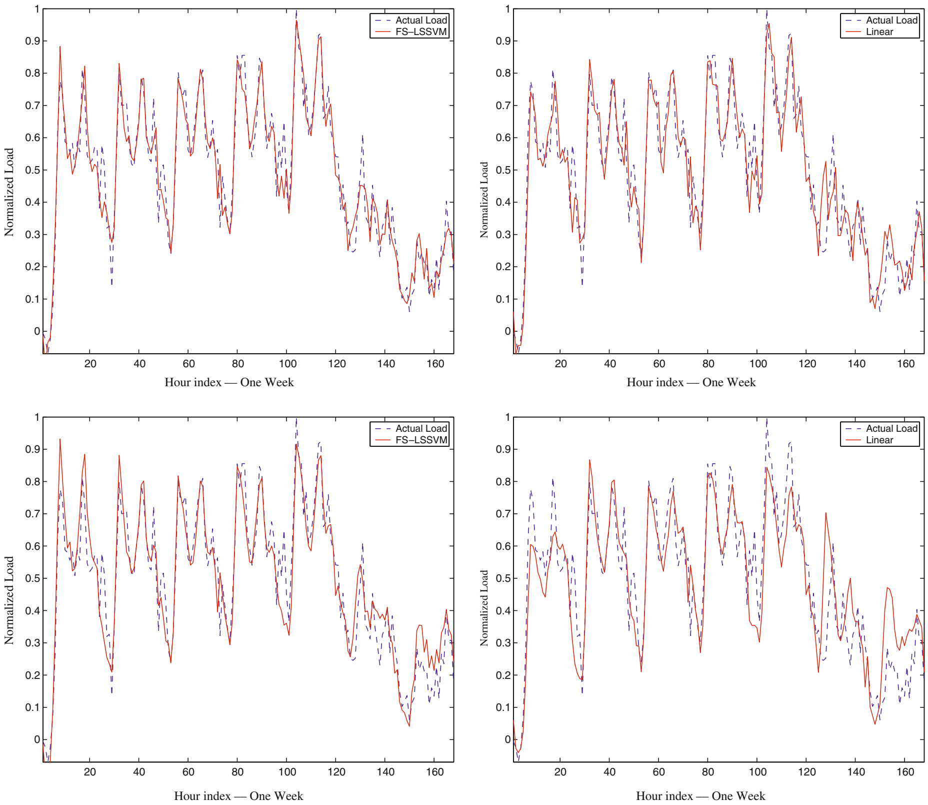

Fig. 6 Forecasts comparison. FS-LSSVM and Linear 1-h-ahead predictions (*Top-left* and *Top-right*, respectively). FS-LSSVM and Linear 24-h-ahead predictions (*Bottom-left* and *Bottom-right*, respectively), for a full week (series 3)

5.4. Out of sample performance

The models are compared on a test data set that consists of 15 days after the last sample point. The performance is assessed over 2 forecasting modes: one-hour-ahead prediction, and 24-h-ahead-simulation with updates at 00:00 hrs. of each day. The performance is measured by the MSE and the mean absolute percentage error (MAPE). As indicated above, three models are estimated for each load series: the fixed-size LS-SVM (FS-LSSVM) estimated using the entire sample, the standard LS-SVM in dual version estimated with the last 1,000 datapoints of the training sample, and a linear model estimated with the same variables as the FS-LSSVM.

The fixed-size LS-SVM models are computed using M=1,000 initial support vectors. The different performance levels across series is due to the different behavior of each particular load series. Tables 3 and 4 show the comparison between the models for the different forecasting modes over the ten load series. It is clear that the FS-LSSVM improves over the traditional LS-SVM by using the entire datasample available, rather than just using the last 1,000 datapoints. In the context of load-forecasting, the existence of important seasonal variations makes it important to consider as much datapoints as possible into the model. On the other hand, the linear model shows good performance in some series, but it is always outperformed

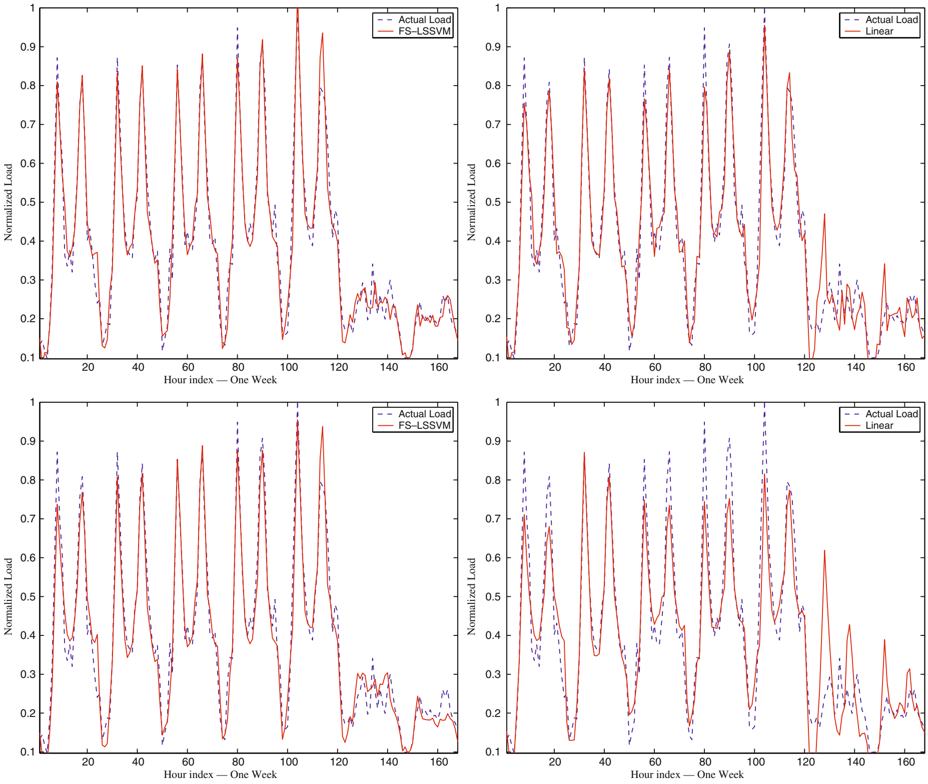

Fig. 7 Forecasts comparison. FS-LSSVM and Linear 1-h-ahead predictions (*Top-left* and *Top-right*, respectively). FS-LSSVM and Linear 24-h-ahead predictions (*Bottom-left* and *Bottom-right*, respectively), for a full week (series 4)

by the fixed-size LS-SVM. Linear models for load forecasting have to be designed in more detail to improve its performance, through the explicit incorporation of seasonal variations across weeks and days into the model [e.g. periodic linear autoregressions (Espinoza et al. 2005)]. The nonlinear model requires less effort from the user in the definition of the model, and the whole procedure can be automatized.

The comparison between the forecasts obtained with the fixed-size LS-SVM and the linear model are shown on Figures 6, 7 and 8 for series 3, 4, and 9, respectively. On each figure, the top panels show the performance using one-hour-ahead forecasts. The bottom panels show the comparison using 24-h-ahead simulation. Each plot shows the first 7 days of the test set, starting with 00:00 h on Monday. It is clearly visible that the fixed-size LS-SVM model provides better forecasts, particularly for the case of 24-h-ahead prediction. It is also interesting to note the different behavior of each load series.

6. Conclusion

This paper illustrates the application of a large-scale nonlinear regression technique to a real-life modelling problem. We have shown that it is possible to build a large scale nonlinear

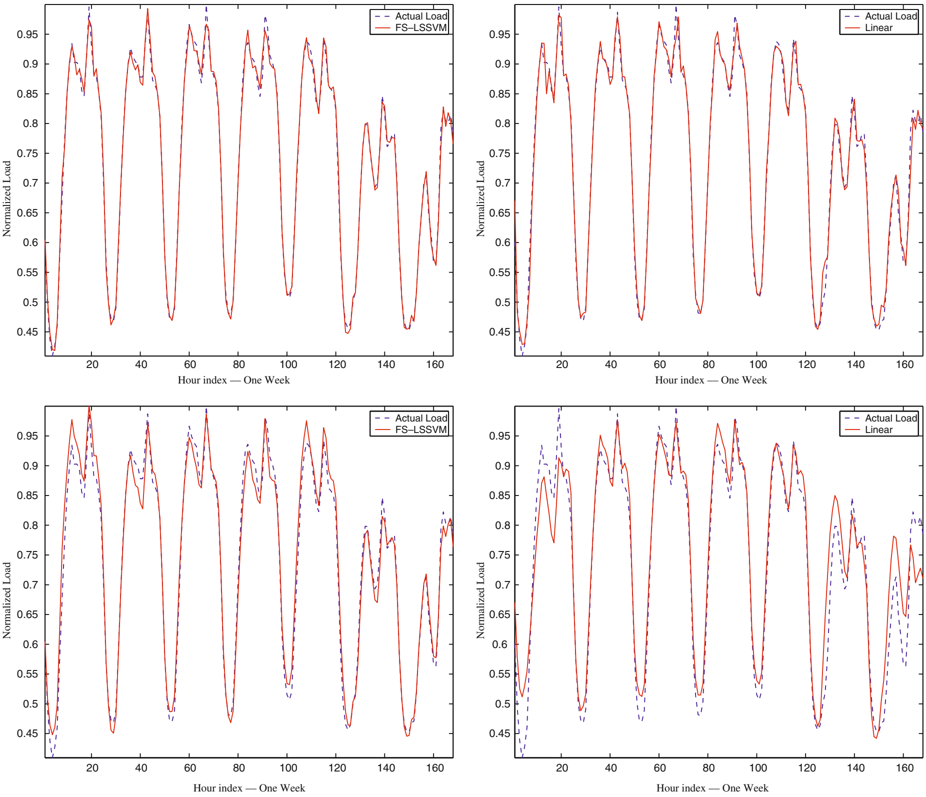

Fig. 8 Forecasts comparison. FS-LSSVM and Linear 1-h-ahead predictions (*Top-left* and *Top-right*, respectively). FS-LSSVM and Linear 24-h-ahead predictions (*Bottom-left* and *Bottom-right*, respectively), for a full week (series 9)

regression model, using the fixed-size LS-SVM, from a dataset consisting of N=36,000 datapoints. This is done by selecting an initial subsample of size $M \ll N$, that provides a sparse representation of the nonlinear mapping. The results show that the nonlinear regressions in primal space improve their accuracy with larger values of M. The maximum value of M to be used depends on the computational resources at hand, and it also depends on the underlying distributional properties of the dataset. In this context, it was shown that quadratic entropy active selection of support vectors leads to performances which are less disperse as those obtained by random selection of support vectors.

The forecasting performance, assessed for ten different load series, is very satisfactory. The MSE levels are below 3% in most cases. Not only the model estimated with fixed-size LS-SVM produces better results than a linear model estimated with the same variables, but also it produces better results than a standard LS-SVM in dual space estimated using only the last 1,000 datapoints. Furthermore, the good performance of the fixed-size LS-SVM is obtained based on a subsample of M=1,000 initial support vectors, which represent less than 3% of the available sample. Further research on a more dedicated definition of the initial input variables (e.g. incorporation of external variables to reflect industrial activity, use of explicit seasonal information, etc.) should lead to further improvements.

References

Björkstrom A, Sundberg R (1999) A Generalized view on continuum regression. Scand J Stat 26:17–30

Bunn D (2000) Forecasting load and prices in competitive power markets. Invited paper, Proc IEEE 88(2): 163–169

Chen BJ, Chang MW, Lin CJ (2002) Load forecasting using support vector machines: a study on EUNITE competition 2001. Technical report, Department of computer science and information engineering, National Taiwan University, Taipei, Taiwan

Cristianini N, Shawe-Taylor J (2000) An introduction to support vector machines. Cambridge University Press, London

Engle R, Granger CJ, Rice J, Weiss A (1986) Semiparametric estimates of the relation between weather and electricity sales. J Am Stat Assoc 81:394, 310–320

Espinoza M, Suykens JAK, De Moor B (2003) Least squares support vector machines and primal space estimation. In: Proceedings of the IEEE 42nd conference on decision and control, Maui, USA, pp 3451–3456

Espinoza M, Pelckmans K, Hoegaerts L, Suykens JAK, De Moor B (2004) A comparative study of LS-SVM applied to the silverbox Identification problem. In: Proceedings of the 6th IFAC conference on nonlinear control systems (NOLCOS), Stuttgart, Germany

Espinoza M, Joye C, Belmans R, De Moor B (2005) Short term load forecasting, profile identification and customer segmentation: a methodology based on periodic time series. IEEE Trans Power Syst 20(3):1622–1630

Fay D, Ringwood J, Condon M, Kelly M (2003) 24-h electrical load data – a sequential or partitioned time series?. Neurocomputing 55:469–498

Frank I, Friedman J (1993) A statistical view of some chemometrics regression tools. Technometrics 35:109–148

Girolami M (2003) Orthogonal series density estimation and the kernel eigenvalue problem. Neural Comput 14(3):669–688

Girosi F (1998) An equivalence between sparse approximation and support vector machines. Neural Comput 10(6):1455–1480

Hylleberg S (1992) Modelling Seasonality. Oxford University Press, New York

Hoerl AE, Kennard RW (1970) Ridge regression: biased estimation for non orthogonal problems. Technometrics 8:27–51

Ljung L (1999) Systems identification: theory for the user. 2nd Edn, Prentice Hall, New Jersey

Lotufo ADP, Minussi CR (1999) Electric power systems load forecasting: a survey. IEEE Power Tech Conference, Budapest, Hungary

MacKay DJC (1995) Probable networks and plausible predictions – a review of practical Bayesian methods for supervised neural networks. Netw Comput Neural Syst 6:469–505

Mariani E, Murthy SS (1997) Advanced load dispatch for power systems. Advances in industrial control, Springer, Berlin Heidelberg New York

Poggio T, Girosi F (1990) Networks for approximation and learning. Proc IEEE 78(9):1481–1497

Ramanathan R, Engle R, Granger CWJ, Vahid-Aragui F, Brace C (1997) Short-run forecasts of electricity load and peaks. Int J Forecast 13:161–174

Shawe-Taylor J, Williams CKI (2003) The stability of kernel principal components analysis and its relation to the process eigenspectrum. In: Advances in neural information processing systems Vol 15, MIT Press, cambridge

Steinherz H, Pedreira C, Castro R (2001) Neural networks for short-term load forecasting: a review and evaluation. IEEE trans power syst 16(1):69–96

Stone M, Brooks RJ (1990) Continuum regression: cross-validated sequentially contructed prediction embracing ordinary least squares, partial least squares and principal components regression. J R Stat Soc B 52:237–269

Sundberg R (1993) Continuum Regression and ridge regression. J R Stat Soc B 55:653–659

Suykens JAK, Vandewalle J (1999) Least squares support vector machines classifiers. Neural Process Lett 9:293–300

Suykens JAK, De Brabanter J, Lukas L, Vandewalle J (2002a) Weighted least squares support vector machines: robustness and sparse approximation. Neurocomputing 48(1–4):85–105 (Special issue on fundamental and information processing aspects of neurocomputing)

Suykens JAK, Van Gestel T, De Brabanter J, De Moor B, Vandewalle J (2002b) Least squares support vector machines. World Scientific, Singapore

Vapnik V (1998) Statistical learning theory. Wiley, New York

Verbeek M (2000) A guide to modern econometrics. Wiley, New York

Weigend AS, Gershenfeld NA (eds) (1994) Time series predictions: forecasting the future and understanding the past. Addison-Wesley, Reading

Williams CKI (1998) Prediction with gaussian processes: from linear regression to linear prediction and beyond. In: Jordan MI (ed) Learning and inference in graphical models. Kluwer, Dordrecht

Williams CKI, Seeger M (2000a) The effect of the input density distribution on kernel-based classifiers. In: Langley (ed) Proceedings of the 17th international conference on machine learning (ICML 2000), Morgan Kauffmann, San Fransisco

Williams CKI, Seeger M (2000b) Using the Nyström method to speed up kernel machines. In: Leen T, Dietterich T, Tresp V (eds) Proceedings NIPS'2000, vol 13, MIT press, Cambridge