A COMPARATIVE STUDY OF LS-SVM'S APPLIED TO THE SILVER BOX IDENTIFICATION PROBLEM

Marcelo Espinoza, Kristiaan Pelckmans, Luc Hoegaerts, Johan Suykens, Bart De Moor

K.U. Leuven, ESAT-SCD-SISTA
Kasteelpark Arenberg 10, B-3001 Leuven, Belgium
Email: marcelo.espinoza@esat.kuleuven.ac.be

Abstract: Within the context of nonlinear system identification, different variants of LS-SVM are applied to the Silver Box dataset. Starting from the dual representation of the LS-SVM, and using Nyström techniques, it is possible to compute an approximation for the nonlinear mapping to be used in the primal space. In this way, primal space based techniques as Ordinary Least Squares (OLS), Ridge Regression (RR) and Partial Least Squares (PLS) are applied to the same dataset together with the dual version of LS-SVM. We obtain mean squared error values of the order of 10^{-7} using iterative prediction on a pre-defined test set.

Keywords: Fixed-Size LS-SVM, Nyström Approximation, Kernel Methods, Sparseness, Primal Space, Nonlinear Identification.

1. INTRODUCTION

For the task of nonlinear system identification, one approach is to apply a black-box technique. In this way, it is possible to define a regression vector from a set of inputs (Sjöberg *et al.*, 1995) and a nonlinear mapping in order to finally estimate a model suitable for prediction or control. Kernel based estimation techniques, such as Support Vector Machines (SVMs) and Least Squares Support Vector Machines (LS-SVMs) have shown to be powerful nonlinear regression methods (Poggio, 1990; Vapnik, 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 φ. 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 φ. 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 in dual space under a least-squares cost function (Suykens, 1999), where the sparseness property can be obtained by sequentially pruning the support value spectrum (Suykens *et al.*, 2002). 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 e.g. by cross-validation or by using Bayesian techniques (MacKay, 1995).

Although the LS-SVM system is solved on its dual form, the problem can be formulated directly in primal space by means of an explicit approximation for the nonlinear mapping φ. Furthermore, it is possible to compute a sparse approximation by using only a subsample of selected Support Vectors from the dataset in order to estimate a large-scale nonlinear regression problem in primal space. Primal-dual formulations in the LS-SVM context have been given also for kernel-Fisher's discriminant (kFDA), kernel-Partial Least Squares (kPLS), kernel-Canonical Correlation (kCCA) analysis (Suykens *et al.*, 2002b). Working in primal space gives enough flexibility to apply different techniques from statistics or the traditional system identification framework (Ljung, 1999).

In this paper we apply a battery of different variants of the LS-SVM in dual and primal space to a nonlinear identification problem known as the "Silver Box" (Schoukens *et al.*, 2003). According to the definition of the data, it is an example of a nonlinear dynamic system with a dominant linear behavior. This paper is structured as follows. The basic description of the LS-SVM is presented in Section II. In Section III, the methodology for working in primal space for the different variants of the LS-SVM are described. Section IV presents the problem and describes the overall setting for the working procedure, and the results are reported in Section V.

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

$$y = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}) + b \tag{1}$$

where $\boldsymbol{x} \in \mathbb{R}^n$, $y \in \mathbb{R}$ and $\boldsymbol{\varphi}(\cdot) : \mathbb{R}^n \to \mathbb{R}^{n_h}$ is the mapping to a high dimensional (and possibly infinite dimensional) feature space. The following optimization problem is formulated:

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

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

With the application of the Mercer's theorem on the kernel matrix $\boldsymbol{\Omega}$ as $\boldsymbol{\Omega}_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{\varphi}(\boldsymbol{x}_i)^T\boldsymbol{\varphi}(\boldsymbol{x}_j), i, j = 1, \ldots, N$ 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: $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{x}_i^T\boldsymbol{x}_j$ (linear kernel); $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i^T\boldsymbol{x}_j + c)^d$ (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.

From the Lagrangian $\mathcal{L}(\boldsymbol{w}, b, e; \alpha) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \gamma\frac{1}{2}\sum_{i=1}^N e_i^2 - \sum_{i=1}^N \alpha_i(\boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_i) + b + e_i - y_i)$, where $\alpha_i \in \mathbb{R}$ are the Lagrange multipliers, the conditions for optimality are given by:

$$\begin{cases} \dfrac{\partial\mathcal{L}}{\partial\boldsymbol{w}} = 0 \to \boldsymbol{w} = \displaystyle\sum_{i=1}^N \alpha_i\boldsymbol{\varphi}(x_i) \\ \dfrac{\partial\mathcal{L}}{\partial b} = 0 \to \displaystyle\sum_{i=1}^N \alpha_i = 0 \\ \dfrac{\partial\mathcal{L}}{\partial e_i} = 0 \to \alpha_i = \gamma_i e_i, \qquad i = 1, \ldots, N \\ \dfrac{\partial\mathcal{L}}{\partial\alpha_i} = 0 \to y_i = \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_i) + b + e_i, \end{cases} \tag{3}$$

By elimination of $\boldsymbol{w}$ and e_i, the following linear system is obtained:

$$\left[\begin{array}{c|c} 0 & \mathbf{1}^T \\ \hline \mathbf{1} & \boldsymbol{\Omega} + \gamma^{-1}\mathbf{I} \end{array}\right] \left[\begin{array}{c} b \\ \hline \boldsymbol{\alpha} \end{array}\right] = \left[\begin{array}{c} 0 \\ \hline \boldsymbol{y} \end{array}\right], \tag{4}$$

with $\boldsymbol{y} = [y_1, \ldots, y_N]^T, \boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_N]^T$. The resulting LS-SVM model in dual space becomes

$$y(\boldsymbol{x}) = \sum_{i=1}^N \alpha_i K(\boldsymbol{x}, \boldsymbol{x}_i) + b. \tag{5}$$

Usually the training of the LS-SVM model involves an optimal selection of the kernel parameters and the regularization parameter, which can be done using e.g. cross-validation techniques or Bayesian inference (MacKay, 1995).

3. ESTIMATION IN PRIMAL SPACE

In order to work in the primal space, it is required to compute an explicit approximation of the nonlinear mapping $\boldsymbol{\varphi}$. Then, the final estimation of the model can be done using different techniques. In this paper, we apply Ordinary Least Squares (OLS), Ridge Regression (RR) and Partial Least Squares (PLS) in primal space.

3.1 Nonlinear 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 $K(\boldsymbol{x}_i, \boldsymbol{x}_j)$. Given the integral equation $\int K(\boldsymbol{x}, \boldsymbol{x}_j)\phi_i(\boldsymbol{x})p(\boldsymbol{x})d\boldsymbol{x} = \lambda_i\phi_i(\boldsymbol{x}_j)$, with solutions λ_i and ϕ_i for a variable $\boldsymbol{x}$ with probability density $p(\boldsymbol{x})$, we can write

$$\boldsymbol{\varphi} = [\sqrt{\lambda_1}\phi_1, \sqrt{\lambda_2}\phi_2, \ldots, \sqrt{\lambda}_{n_h}\phi_{n_h}]. \tag{6}$$

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, 2000). This will lead to the eigenvalue problem (Nyström approximation)

$$\frac{1}{N}\sum_{k=1}^N K(\boldsymbol{x}_k, \boldsymbol{x}_j)u_i(\boldsymbol{x}_k) = \lambda_i^{(s)}u_i(\boldsymbol{x}_j), \tag{7}$$

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

$$\hat{\lambda}_i = \frac{1}{N}\lambda_i^{(s)}, \hat{\phi}_i = \sqrt{N}u_i. \tag{8}$$

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 i^{th} required component of $\hat{\boldsymbol{\varphi}}(\boldsymbol{x})$ simply by applying (6) if $\boldsymbol{x} \in \{\boldsymbol{x}_i\}_{i=1}^N$ (is a training point), or for any point $\boldsymbol{x}^{(v)}$ by means of

$$\hat{\boldsymbol{\varphi}}_i(\boldsymbol{x}^{(v)}) \propto \frac{1}{\sqrt{\lambda_i^{(s)}}}\sum_{k=1}^N u_{ki}K(\boldsymbol{x}_k, \boldsymbol{x}^{(v)}). \tag{9}$$

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

3.2 Sparseness 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 yield at most N components, each one of which can be computed by (8) for all $\boldsymbol{x} \in \{\boldsymbol{x}_i\}_{i=1}^N$. However, if we have a large scale problem, it has been motivated (Suykens *et al.*, 2002b) to use of a subsample of $M \ll N$ datapoints to compute the $\hat{\boldsymbol{\varphi}}$. In this case, up to M components will be computed. External criteria such as entropy maximization can be applied for an optimal selection of the subsample: given a

fixed-size M, the aim is to select the support vectors that maximize the quadratic Renyi entropy (Girolami, 2003)

$$H_R = -\log \int p(x)^2 dx \tag{10}$$

that can be approximated by

$$\int \hat{p}(x)^2 dx = \frac{1}{M^2} \mathbf{1}^T \mathbf{\Omega} \mathbf{1}. \tag{11}$$

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 modeling exercise.

3.3 Estimation Techniques

Once the nonlinear mapping has been computed (either using the full sample or using a sparse approximation based on a subsample), the model has to be estimated in primal space. Let us denote by $\boldsymbol{z}_k = \hat{\boldsymbol{\varphi}}(\boldsymbol{x}_k)$ for $k = 1, \ldots N$ and consider the new $\boldsymbol{z}_k \in \mathbb{R}^m$ as the observations for the linear regression

$$\boldsymbol{y} = \boldsymbol{Z}\boldsymbol{\beta} + b\mathbf{1} + \boldsymbol{\varepsilon} \tag{12}$$

with $\boldsymbol{\varepsilon} = [\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_N]^T \in \mathbb{R}^{N\times 1}$ $\boldsymbol{y} = [y_1, y_2, \ldots, y_N]^T \in \mathbb{R}^{N\times 1}$ and $\boldsymbol{Z} = [\boldsymbol{z}_1^T; \boldsymbol{z}_2^T; \boldsymbol{z}_3^T; \ldots; \boldsymbol{z}_N^T] \in \mathbb{R}^{N\times m}$. The quantity m is the number of components of $\hat{\boldsymbol{\varphi}}$ that are going to be considered in the estimation, with $m \leq M$. For ease of notation, consider the matrix of full regressors $\boldsymbol{Z_F} = [\mathbf{1}\boldsymbol{Z}]$, and the vector of full coefficients $\boldsymbol{\beta_F} = [b, \boldsymbol{\beta}]^T$. The regression (12) can be written as:

$$\boldsymbol{y} = \boldsymbol{Z_F}\boldsymbol{\beta_F} + \boldsymbol{\varepsilon}. \tag{13}$$

3.3.1. Ridge Regression The LS-SVM formulation is originally defined as a ridge-regression problem in feature space, also called Kernel Ridge Regression in some contexts (Cristianini and Shawe-Taylor, 2000). Then, the most direct approach is to reproduce the same problem (2) using the approximation $\hat{\boldsymbol{\varphi}}$ in primal space. In this case, traditional ridge-regression (Hoerl and Kennard, 1970) is applied, and the regularization parameter γ needs to be tuned accordingly. In the case of the regression (13), the ridge-regression solution for $\hat{\boldsymbol{\beta}}_{\boldsymbol{F}}$ is given by

$$\hat{\boldsymbol{\beta}}_{\boldsymbol{F}\mathrm{RR}} = (\boldsymbol{Z_F}^T \boldsymbol{Z_F} + \frac{I}{\gamma})^{-1} \boldsymbol{Z_F}^T \boldsymbol{y} \tag{14}$$

where I is the identity matrix of dimension $m+1$. For this case, m can be equal to M and thus include all the computed components for $\hat{\boldsymbol{\varphi}}$, as the introduction of γ will reduce any collinearity problem between them.

3.3.2. Ordinary Least Squares Another approach is to use directly OLS in primal space; but in order to do so it is necessary to only use $m < M$ components of $\hat{\boldsymbol{\varphi}}$, selected by looking at the eigenspectrum of the $M \times M$ kernel matrix $\mathbf{\Omega}$ (Espinoza *et al.*, 2003). Near collinearity is thus avoided, and there should be no need for the regularization parameter γ as in the original dual formulation of the LS-SVM. In this case, the estimated model will yield

$$\hat{\boldsymbol{\beta}}_{\boldsymbol{F}\mathrm{OLS}} = (\boldsymbol{Z_F}^T \boldsymbol{Z_F})^{-1} \boldsymbol{Z_F}^T \boldsymbol{y} \tag{15}$$

where $\boldsymbol{Z_F}$ is built using only $m < M$ columns of the nonlinear approximation.

3.3.3. Partial Least Squares This case involves an explicit construction of the set of regressors to be included in the $\boldsymbol{Z_F}$ matrix in order to take into account the information on the dependent variable and its correlation with the features. For this, we employ the method of Partial Least Squares in feature space (or kernel Partial Least Squares, kPLS (Rosipal, 2001)). Here we use not only the kernel $\mathbf{\Omega} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$, but also the linear kernel $\mathbf{\Omega}_{\boldsymbol{y}}$ computed upon the dependent variable $\boldsymbol{y}$. Therefore, in kPLS one iteratively computes the first eigenvectors of $\mathbf{\Omega}\mathbf{\Omega}_{\boldsymbol{y}}$ (related to the covariance between $\boldsymbol{x}$ and $\boldsymbol{y}$), each time deflating the $\mathbf{\Omega}$ and $\mathbf{\Omega}_{\boldsymbol{y}}$ matrices on the previously found component to obtain orthogonal components on each step. Finally, ridge-regression is applied on the constructed set of regressors.

4. IMPLEMENTATION FOR THE SILVER BOX PROBLEM

The definition of the training and validation strategy using the dataset at hand and the accuracy measurements to be reported are described in this section.

4.1 Techniques to be applied

The different variants of LS-SVM to be applied in this case, as described in previous sections, are:

- LS-SVM in dual space (LS-SVM). For this method, a subsample of size 1000 is used for training, as using the full training sample is prohibitive.
- Fixed Size LS-SVM in primal space with OLS (FS-OLS), RR (FS-RR) and PLS (FS-PLS). For these methods, different numbers of support vectors are selected to build the nonlinear mapping approximation. All subsamples are selected by maximization of the quadratic entropy criterion.

The general model structure is a NARX specification of the form $y_t = \boldsymbol{\varphi}(y_{t-1}, \ldots, y_{t-p}, u_{t-1}, \ldots, u_{t-p}) + \varepsilon_t$. Exploratory analysis for estimating the order p is based on the training-validation framework.

4.2 Data Description and Training Procedure

The data contains samples for input u_i and output y_i, with $i = 1, \ldots, N$, with $N = 131,072$ datapoints. An initial plot of the output (the "arrow")

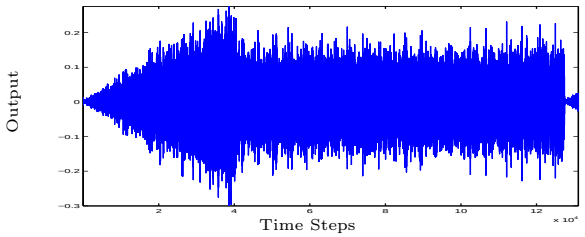


Fig. 1. Available output data series for the Silver Box identification problem.

is given in Figure 1.The working strategy for using the data in term of training, validation and testing goes as follows:

- Training Sample: The first half of the "body of the arrow", namely datapoints 40,001 to 85,000. Models are estimated using this part of the data. The mean squared error (MSE) for a one-step-ahead prediction can be computed directly using this training sample.
- Validation Sample: The second half of the "body of the arrow", datapoints 85,001 to the end. Having estimated the model parameters using the training sample, now the model is validated using new datapoints. A MSE on the validation set is computed on a one-step-ahead basis. Model selection is based on the validation MSE.
- Test Sample: The "head of the arrow", datapoints 1 to 40,000. After defined the optimal model (using the validation MSE), the prediction for the test set is done. In this case, an iterative prediction is computed for the entire test set (each time using past predictions as inputs, using the estimated model in simulation mode). The MSE on the test set is computed.

4.3 Hyperparameter Definitions

The LS-SVM formulation requires to use a kernel matrix $\boldsymbol{\Omega}_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$. In our experiments, as the nonlinear system is known to have a dominant linear behavior, we implemented not only the RBF kernel $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(-||\boldsymbol{x}_i - \boldsymbol{x}_j||_2^2/\sigma^2)$, but also the polynomial kernel $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i^T \boldsymbol{x}_j + c)^d$. Parameters σ, d, c and the regularization parameter γ are also tuned based on the training-validation scheme. From the machine-learning point of view, this procedure is not optimal as we are using only one training-validation setup; normally multiple crossvalidation is performed to achieve optimal results. In this sense, the results of this paper can even be further improved.

5. RESULTS

In this section we show the main results for the iterative prediction obtained with each method, and also some intermediate results related to specific definitions of the model.

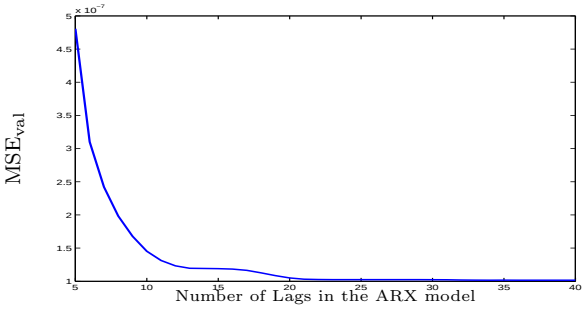


Fig. 2. The error in the validation set using a linear ARX model with increasing number of lags.

Method	γ	Kernel	p	$\text{MSE}_{\text{train}}$	MSE_{val}
LS-SVM	10	Poly	5	2.63×10^{-9}	4.55×10^{-9}
	10	RBF	5	5.78×10^{-8}	7.66×10^{-8}
FS-OLS	-	Poly	7	1.33×10^{-7}	1.23×10^{-7}
	-	RBF	7	3.57×10^{-7}	1.72×10^{-6}
FS-RR	1000	Poly	10	5.28×10^{-8}	5.05×10^{-8}
	1000	RBF	10	3.61×10^{-7}	2.93×10^{-7}
FS-PLS	1000	Poly	10	5.34×10^{-8}	5.09×10^{-8}
	1000	RBF	10	1.16×10^{-6}	1.00×10^{-6}

Table 1. Best models, based on the MSE_{val}. For all cases shown, $\sigma = 5.19p^{-1/2}$ with p=number of lags (for RBF kernel); $d = 3, c = 11$ (for Polynomial kernel).

5.1 Estimation and Model Selection

Using the definition of training and validation data described above, we start checking different lag orders and general parameters. Each time the model is estimated using the training set and then evaluated in the validation set, always on a one-step-ahead basis. We select the best model based on the lowest MSE on the validation set (MSE_{val}).

An initial analysis using a linear ARX model with increasing lags of inputs and outputs, using the same training/validation scheme, shows that the MSE for the validation set can easily reach levels of 1×10^{-7}. Figure 2 shows the MSE_{val} obtained when the number of lags varies from 5 to 40. However small, this error level obtained for high lags can be a symptom of overfitting.

For the NARX models, Table 1 shows the best results achieved for each one of the different techniques. It is important to remember that all the techniques based on the fixed-size primal space version make use of all the training/validation set; whereas the LS-SVM in dual space is limited to a subsample of 1000 points for training and validation. All MSE figures are expressed in the original units of the data.

It is clear that for all cases the polynomial kernel outperforms the RBF one, up to 2 orders of magnitude. Although the RBF kernel is widely used, the dominant linear behavior of the data is better captured here by the polynomial specification. Additionally, the performance of the FS-RR and FS-PLS models with polynomial kernel is much better than the obtained with the FS-OLS in the

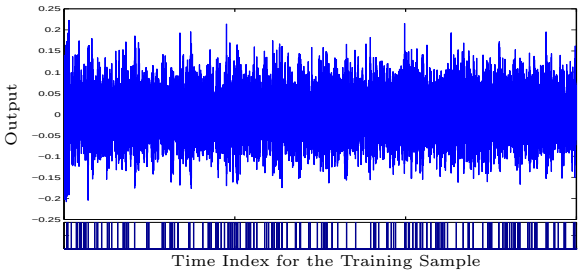


Fig. 3. (Top) The output training sample; (Bottom) The position, as time index, of the 500 selected support vectors is represented by dark bars

Number of Support Vectors M	$\text{MSE}_{\text{train}}$	MSE_{val}
100	7.91×10^{-6}	6.12×10^{-6}
200	2.11×10^{-7}	1.94×10^{-7}
300	1.57×10^{-7}	1.46×10^{-7}
400	1.42×10^{-7}	1.34×10^{-7}
500	1.33×10^{-7}	1.23×10^{-7}
1000	1.27×10^{-7}	1.19×10^{-7}
1500	1.23×10^{-7}	1.15×10^{-7}

Table 2. Effect of M on the performance of the FS-OLS estimator.

training/validation scheme. This means that there exists helpful information in the last components of the eigendecomposition of the $M-$sized kernel matrix $\boldsymbol{\Omega}$, which are just dropped out in the case of FS-OLS but not in FS-RR or FS-PLS.

The effect of selecting different numbers M of initial support vectors on the validation performance is reported in Table 2, for the FS-OLS version with polynomial kernel, where it is clear that the performance is improving marginally for $M > 500$. Therefore, taking into account practical considerations, we chose to keep $M = 500$ for the whole modeling exercise. The position of the selected 500 support vectors can be visualized in terms of the corresponding position of the output data $\boldsymbol{y}$. Figure 3 shows the dependent variable $\boldsymbol{y}$ in the training set, and the dark bars in the bottom represent the position of the selected support vectors. The quite uniform distribution shows that $\boldsymbol{y}$ do not have critical transition regions or critical zones. Finally, the effect of the inclusion of different lags was tested for the NARX models, using lags from 2 to 10. Figure 4 shows the evolution of the MSE in the validation set for FS-RR (full line), FS-OLS (dash-dot) and FS-PLS (dashed).

5.2 Final Results on Test Data Set

After selecting the order of the models and the parameters involved, each one of the estimated models is used to build an iterative prediction (simulation mode, using only past predictions and input information) for the first 40,000 datapoints (the "head of the arrow"). As this is a completely unseen dataset, from the point of view of the modeling strategy, we can expect two types of error sources: the first one, due to the iterative nature of the simulations, so past errors can

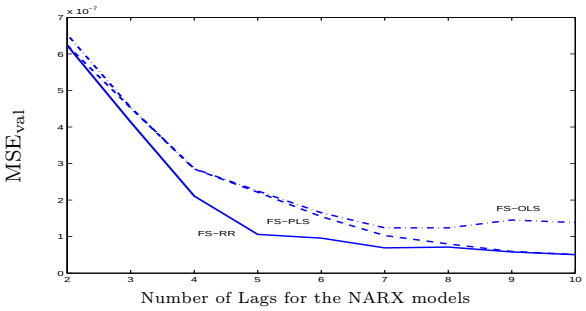


Fig. 4. MSE on the validation set obtained for FS-RR (full line), FS-PLS (dashed) and FS-OLS (dash-dot) using different number of lags.

Technique	Lags	MSE_{test}
Linear	30	0.0718
LS-SVM	5	3.90×10^{-7}
FS-OLS	7	3.67×10^{-7}
FS-RR	10	1.08×10^{-7}
FS-PLS	10	1.01×10^{-7}

Table 3. MSE with the final iterative prediction (simulation mode) on the test data.

propagate to the next predictions; and the second one, due to the fact that there are datapoints that are located beyond the range on which the models were trained, namely the wider zone of the "head of the arrow". The iterated prediction series is compared with the true values, and then we compute this MSE on the test set (MSE_{test}).

Table 3 shows the results obtained with the iterative prediction, for all models, including the linear ARX model for comparison. The result for the linear model shows its lack of generalization ability for this example. The NARX models show a satisfactory performance, where FS-PLS and FS-RR obtain quite the same level of performance. As these two techniques do not drop off a priori any component of the $\boldsymbol{\varphi}$ nonlinear mapping, it can be expected to find that the performance of LS-OLS is lower than the other 2 techniques. Finally, LS-SVM with a direct subsampling for the computation of the model in dual space, obtains a MSE level in the test set within the same order of magnitude (10^{-7})), but almost 4 times the one obtained by FS-PLS or FS-RR.

Figure 5 shows the residuals of the iterative prediction (simulation mode), where it can be seen that the error remains within a stable zone, with the exception of very few peaks close to the wider zone of the "head of the arrow". In any case, the larger peak represents a 5% absolute error with respect to the level of the output series in that point.

6. CONCLUSION

The application of the LS-SVM methodology in a large scale nonlinear identification problem im-

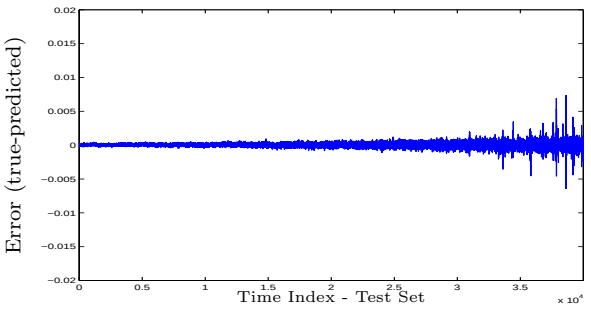


Fig. 5. The errors of the iterative prediction (simulation mode) in the test set. Only few peaks with larger errors are visible.

plies the challenge of working with a large number of datapoints. In this case, the application of the LS-SVM methodology directly requires a subsampling of the data and it is not possible to take into account the information contained in the entire training set. On the contrary, fixed size variants of the LS-SVM developed to work in primal space, do rely on approximations of the nonlinear mapping φ. But these techniques have the advantage that traditional tools available for regression can be applied successfully. In this paper, we have applied LS-SVM in dual space and 3 variants in primal space (fixed size - ordinary least squares, FS-OLS; fized size ridge regression, FS-RR; and fized size partial least squares, FS-PLS) to the identification of a nonlinear dynamical system using the "Silver Box" data.

The results show that the methods that rely on regularization and active construction of a regression in primal space (FS-RR and FS-PLS) obtain the best performance in the iterative prediction exercise. FS-OLS obtains a lower performance, but still better than the one obtained by the LS-SVM method with a direct sampling. The best performance yields an MSE on the test set of 1.0×10^{-7}.

The above results were obtained with models under a suboptimal strategy for model validation and selection. Usually crossvalidation in multiple training/validation sets are done, and in this case we only used one dataset for training and one for validation. Although this is a practical decision, mainly related to the number of datapoints available, results could be improved with a more optimal strategy.

ACKNOWLEDGMENTS

This work was supported by grants and projects for the Research Council K.U.L (GOA-Mefisto 666, IDO, PhD/Postdocs & fellow grants), the Flemish Government (FWO: PhD/Postdocs grants, projects G.0240.99, G.0407.02, G.0197.02, G.0141.03, G.0491.03, G.0120.03, ICCoS, ANMMM; AWI;IWT:PhD grants, Soft4s), the Belgian Federal Government (DWTC: IUAP IV-02, IUAP V-22; PODO-II CP/40), the EU(CAGE, ERNSI, Eureka 2063-Impact;Eureka 2419-FLiTE) and Contracts Research/Agreements (Data4s, Electrabel, Elia, LMS, IPCOS, VIB). J.Suykens and Bart De Moor are an associated professor and a full professor (respectively) at the K.U.Leuven, Belgium. The scientific responsibility is assumed by its authors.

REFERENCES

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

Espinoza, M., Suykens, J.A.K. and De Moor B.L.R. "Least Squares Support Vector Machines and Primal Space Estimation," In *Proc. of the 42nd IEEE Conference on Decision and Control*, Maui, USA, pp. 3451-3456, 2003.

Girolami, M. "Orthogonal Series Density Estimation and the Kernel Eigenvalue Problem," *Neural Computation* 14(3), 669-688, 2003.

Girosi, F. "An Equivalence Between Sparse Approximation and Support Vector Machines," *Neural Computation*, 10(6),1455-1480, 1998.

Hoerl, A.E. and Kennard, R.W. "Ridge Regression: Biased Estimation for Non Orthogonal Problems," *Technometrics* 8, 27-51, 1970.

Ljung, L. *System Identification: Theory for the User*. 2nd Edition, Prentice Hall, New Jersey, 1999.

MacKay, D.J.C. "Probable Networks and Plausible Predictions - A Review of Practical Bayesian Methods for Supervised Neural Networks," *Networks: Computations in Neural Systems*,6, 469-505, 1995.

Poggio, T. and Girosi, F. "Networks for Approximation and Learning," *Proceedings of the IEEE*, 78(9), 1481-1497, 1990.

Rosipal R. and Trejo, J. "Kernel Partial Least Squares Regression in Reproducing Kernel Hilbert Space," *Journal of Machine Learning Research*, 2, 97-123, 2001.

Shawe-Taylor, J. and Williams, C.K.I. "The Stability of Kernel Principal Components Analysis and its Relation to the Process Eigenspectrum," in *Advances in Neural Information Processing Systems* 15, MIT Press, 2003.

Schoukens J., Nemeth G., Crama P., Rolain Y., Pintelon R. "Fast Approximate Identification of Nonlinear Systems," *Automatica* 39(7), 2003.

Sjöberg, J., Zhang, Q., Ljung, L,. Benveniste, A., Delyon, B., Glorennec, P.-Y., Hjalmarsson H., Juditsky, A., "Nonlinear Black-Box Modelling in Systems Identification: A Unified Overview," *Automatica* 31(12), 1691-1724, 1995.

Suykens, J.A.K., Vandewalle, J. "Least Squares Support Vector Machines Classifiers," *Neural Processing Letters* 9, 293-300, (1999).

Suykens J.A.K., De Brabanter J., Lukas, L. ,Vandewalle J., "Weighted Least Squares Support Vector Machines: Robustness and Sparse Approximation," *Neurocomputing*, vol. 48, no. 1-4, pp. 85-105, 2002.

Suykens J.A.K.,Van Gestel T., De Brabanter J., De Moor B.,Vandewalle J., *Least Squares Support Vector Machines*. World Scientific, 2002, Singapore.

Vapnik, V. *Statistical Learning Theory*. John Wiley & Sons, New York, 1998.

Williams, C.K.I and Seeger, M. "Using the Nyström Method to Speed Up Kernel Machines," in T.Leen, T.Dietterich, V.Tresp (Eds.), *Proc. NIPS'2000*, vol 13., MIT press.