ANL/RA/CP-100231

*The paper submitted for presentation at the ANS International Topical Meeting on "Advances in Reactor Physics and Mathematics and Computation into the Next Millennium", May 7-11, 2000, Pittsburgh, Pennsylvania,USA

Support Vector Machines for Nuclear Reactor State Estimation

by

Nela Zavaljevski and Kenny C. Gross

Reactor Analysis Division
Argonne National Laboratory
9700 S. Cass Avenue
Argonne, IL 60439

*Work supported by the U. S. Department of Energy, Nuclear Energy Programs under Contract W-31-109-Eng-38

DISCLAIMER

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

SUPPORT VECTOR MACHINES FOR NUCLEAR REACTOR STATE ESTIMATION

N. Zavaljevski and K. C. Gross

Reactor Analysis Division, Argonne National Laboratory
9700 South Cass Avenue, Argonne, IL 60439
nelaz@ra.anl.gov, gross@anl.gov

ABSTRACT

Validation of nuclear power reactor signals is often performed by comparing signal prototypes with the actual reactor signals. The signal prototypes are often computed based on empirical data. The implementation of an estimation algorithm which can make predictions on limited data is an important issue. A new machine learning algorithm called support vector machines (SVMs) recently developed by Vladimir Vapnik and his coworkers enables a high level of generalization with finite high-dimensional data. The improved generalization in comparison with standard methods like neural networks is due mainly to the following characteristics of the method. The input data space is transformed into a high-dimensional feature space using a kernel function, and the learning problem is formulated as a convex quadratic programming problem with a unique solution. In this paper we have applied the SVM method for data-based state estimation in nuclear power reactors. In particular, we implemented and tested kernels developed at Argonne National Laboratory for the Multivariate State Estimation Technique (MSET), a nonlinear, nonparametric estimation technique with a wide range of applications in nuclear reactors. The methodology has been applied to three data sets from experimental and commercial nuclear power reactor applications. The results are promising. The combination of MSET kernels with the SVM method has better noise reduction and generalization properties than the standard MSET algorithm.

1. INTRODUCTION

One approach for validation of nuclear power plant signals makes use of pattern recognition techniques. This approach often assumes that there is a set of signal prototypes that are continuously compared with the actual sensor signals. These signal prototypes are often computed based on empirical models with little or no knowledge about physical processes. A common problem with all data-based models is their limited ability to make predictions on the basis of available training data. Another problem is related to suboptimal training algorithms.

Both of these potential shortcomings with conventional approaches to signal validation and sensor operability validation are successfully resolved by adopting a recently proposed learning paradigm called the Support Vector Machine (SVM)[1]. The work presented here is a novel adaptation of the SVM paradigm for data-based modeling of system state variables in a nuclear power plant, integrated with a nonlinear, nonparametric technique called the Multivariate State Estimation Technique (MSET), an algorithm developed at Argonne National Laboratory for a wide range of nuclear plant applications.[2]

2. BACKGROUND ON MSET

State estimation in MSET is achieved by first organizing the data into a "process memory" matrix $\vec{\vec{D}}$. The number of columns of this matrix is equal to the number of observations, and the number of rows is equal to the number of sensors. If a new observation is made and the sensor measurements from this matrix represent correlated phenomena, then it can be assumed that the estimate of the new state is related to the data matrix in the following way:

$$\vec{X}_{est} = \vec{\vec{D}} \cdot \vec{W} \tag{1}$$

The weight vector $\vec{W}$ is computed in MSET using a proprietary set of nonlinear operators. In a general operator form, the solution for the weight matrix in MSET is given by the following expression:

$$\vec{W} = (\vec{\vec{D}}^T \otimes \vec{\vec{D}})^{-1} (\vec{\vec{D}}^T \otimes \vec{X}_{obs}) \equiv \vec{\vec{K}}^{-1} \vec{A} \tag{2}$$

where symbol $\otimes$ represents a nonlinear operator applied to the input data. The goal of this operation is to transform the input data space into another space, called the feature space, which reveals the similarity between new states and previous states that are stored in the process memory matrix. The matrix $\vec{\vec{K}} = \vec{\vec{D}}^T \otimes \vec{\vec{D}}$ has components $K_{ij} = \vec{X}_i^T \otimes \vec{X}_j$ and is designated as the similarity matrix. Several nonlinear operators have been invented and implemented in the MSET algorithm for a variety of successful applications in the areas of nuclear power plant signal validation and fault identification.[2]

To improve numerical accuracy and stability, Tikhonov regularization[3] has recently been implemented in MSET[4]. The Tikhonov regularized solution $\vec{W}_\lambda$ is obtained as the solution to the following minimization problem:

$$\min_w \{ \| \vec{\vec{K}} \vec{W}_\lambda - \vec{A} \|^2 + \lambda \| \vec{\vec{L}} \vec{W}_\lambda \|^2 \} \tag{3}$$

where λ is the regularization parameter and $\vec{\vec{L}}$ is a convenient regularization matrix that controls the smoothness of the solution. Testing with ill-conditioned problems in MSET has demonstrated

that the generalized cross validation (GCV)[5] procedure provides a good method for optimization of the regularization parameter.

The solution of the regularized problem is

$$\vec{\mathbf{X}}_{est} = \vec{\mathbf{D}}\vec{\mathbf{W}}_{\lambda} = \vec{\mathbf{D}}(\vec{\mathbf{K}} + \lambda\mathbf{I})^{-1}\vec{\mathbf{K}}(\vec{\mathbf{X}}_{obs}) \equiv \vec{\gamma}\ \vec{\mathbf{K}}(\vec{\mathbf{X}}_{obs}) \tag{4}$$

Here $\vec{\mathbf{K}}(\vec{\mathbf{X}}_{obs})$ is a column vector. Each component $(\vec{\mathbf{K}}(\vec{\mathbf{X}}_{obs}))_i \equiv \mathbf{K}(\vec{\mathbf{X}}_{obs}, \vec{\mathbf{X}}_i) = \vec{\mathbf{X}}_{obs}{}^{T} \otimes \vec{\mathbf{X}}_i$ can be regarded as a kernel evaluated at the points $\vec{\mathbf{X}}_{obs}$ and $\vec{\mathbf{X}}$ in the input space.

The estimation problem in the input space (first equality in equation (4)) has been transformed into an estimation problem in the feature space (last equlity). Nonlinear features are defined using kernel **K**. Under some regularity conditions (see, e.g., Girosi[6]), the kernel can be represented as a functional expansion in a set of basis functions, which define a feature space. The ability of the learning algorithm to select appropriate features is the main factor in controlling the generalization of the learning algorithm. An overly rich set of features may lead to overfitting, while a sparse set may lead to large estimation errors.

3. SUPPORT VECTOR MACHINES

Most standard learning methods control the generalization error by keeping the number of features small. For example, in feedforward neural networks, the number of hidden units corresponds to the dimensionality of the feature space. Various heuristic approaches can be used for selecting a small number of features. Keeping the dimensionality of the feature space small effectively controls the model complexity.

The SVM approach is based on a different principle. The feature space is high dimensional (for some kernels the set of basis functions may be infinite), but complexity control is performed in the input space. This is very similar to complexity control in the standard versions of MSET. Although the performance of standard MSET selection procedures was quite good in practice, there was no guarantee that the selected training dataset would represent an optimal tradeoff between compactness and estimation accuracy. The main advantage of the SVM algorithm is that an optimal and unique training dataset can be found.

To exploit an SVM approach, the first quadratic error term in (3) should be replaced with Vapnik's ε-insensitive error function[1] of the form

$$\mathbf{V} = |\vec{\mathbf{K}}\vec{\mathbf{W}} - \vec{\mathbf{A}}|_e = \begin{cases} 0 \text{ if } |\vec{\mathbf{K}}\vec{\mathbf{W}} - \vec{\mathbf{A}}| < \varepsilon \\ |\vec{\mathbf{K}}\vec{\mathbf{W}} - \vec{\mathbf{A}}| - \varepsilon \text{ otherwise} \end{cases} \tag{5}$$

The intuitive motivation for this choice is robustness: errors less than a specified threshold ε are ignored.

In order to be able to approximate a variety of experimental data, one should use a large class of approximating function. One such class of function belongs to a functional space called the Reproducing Kernel Hilbert Space (RKHS). This specific choice of functional space is used both in functional analysis[7] and in the theory of stochastic processes[5]. To each RKHS corresponds a reproducing kernel named so because of the reproducing property defined by

$$< f(\cdot), K(\cdot, x) >_H = f(x) \tag{6}$$

for every function f belonging to that functional space. Here $< , >_H$ denotes the inner product in RKHS.

The reproducing kernel can be expanded in independent functions, called features

$$K(\vec{x}, \vec{z}) = \sum_{n=1}^{N} \beta_n \phi_n(\vec{x}) \phi_n(\vec{z}) \tag{7}$$

Any function from the RKHS defined by ϕ_n can be represented as

$$y(\vec{x}) = \sum_{n=1}^{N} c_n \phi_n(\vec{x}) \tag{8}$$

A natural form for the regularized functional in the RKHS is

$$R_{reg} = \sum_{i=1}^{m} | y_i - \sum_{n=1}^{N} c_n \phi_n(\vec{x}_i) |_\varepsilon + \lambda \sum_{n=1}^{N} \frac{c_n^2}{\beta_n} \tag{9}$$

The second term is a norm in the RKHS defined by functions ϕ_n. Here λ is the regularization parameter, β_n is a positive decreasing sequence.

We have implemented features that are based on the Hermite polynomials

$$H_k(x) = \mu_k P_k(x) e^{-x^2} \tag{10}$$

where

$$P_k(x) = (-1)^k e^{x^2} \left(\frac{d}{dx} \right)^k e^{-x^2} \tag{11}$$

and μ_k are normalization constants.

For these polynomials one can obtain the kernels (Vapnik, 1998)

$$K(x, y) = \sum_{i=0}^{\infty} q^i H_i(x) H_i(y) = \frac{1}{\sqrt{\pi(1-q^2)}} \exp\left\{\frac{2xyq}{1+q} - \frac{(x-y)^2 q^2}{1-q^2}\right\} \tag{12}$$

where $1 \geq q \geq 0$ is a regularization parameter. This kernel has a closed form expression for a multivariate case. The tensor product of a d-dimensional kernel again has the same form [1]

$$K(\vec{x}, \vec{y}) = C \exp(2\delta(\vec{x} \cdot \vec{y})) \exp(-|\vec{x} - \vec{y}|^2 \sigma^2) \tag{13}$$

This form controls both global (scalar product term) and local (Gaussian term) approximation. The reason to use the Hermite polynomials is deeper than the mathematical convenience of the closed expression given by (12). We cannot expect that all our variables will be Gaussian processes. Any sufficiently regular non-linear transformation of a Gaussian process will produce a non-Gaussian process, which can be expanded in Hermite polynomials of the original Gaussian process[8].

The formulated optimization problem for the regularized functional (9) is more difficult than the problem for the quadratic functional (3). When the objective function is convex, this primal problem has a corresponding dual problem insofar as a minimum of the primal objective function is a maximum of the dual objective function. Using the technique of Lagrange multipliers it can be proved[6] that the minimum of the functional (9) is equivalent to the solution obtained using SVM[1].

The SVM-based approximation scheme for any new variable y takes on a form similar to the last equality in equation (4), except that a constant term b is added for numerical stability

$$y = \sum_{i=1}^{m} \gamma_i \mathbf{K}(\vec{x}, \vec{x}_i) + b = \sum_{i=1}^{m} (\alpha_i^* - \alpha_i) \mathbf{K}(\vec{x}, \vec{x}_i) + b \tag{14}$$

The expansion coefficients α_i and α_i^* are the solutions of the following quadratic programming (QP) problem

$$\textbf{min } \mathbf{R}(\alpha^*, \alpha) = \varepsilon \sum_{i=1}^{m} (\alpha_i^* + \alpha_i) - \sum_{i=1}^{m} y_i (\alpha_i^* - \alpha_i) + \frac{1}{2} \sum_{i,j=1}^{m} (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j) \mathbf{K}(\vec{x}_i, \vec{x}_j) \tag{15}$$

subject to the constraints:

$$0 \le \alpha_i^*, \alpha_i \le C = 1/\lambda$$
$$\sum_{i=1}^{m} (\alpha_i^* - \alpha_i) = 0 \qquad (16)$$
$$\alpha_i \alpha_i^* = 0$$

The order of the QP problem is 2m, where m is the number of observations. Due to the nature of this QP problem, only a number of the coefficients α_i^*-α_i will be different from zero, and the input data points associated with those that are nonzero are called support vectors. The number of support vectors depends on C and ε. The parameter C weighs the data term with respect to the smoothness term, and is related to the amount of noise in the input reactor signals. This parameter can be selected based on the GCV criterion. The parameter ε controls the tolerance level in the objective function error. A larger tolerance improves the generalization ability, but values that are too large can lead to unacceptable bias. The constant term b is obtained from the necessary and sufficient conditions for the global minimum of the regularized functional[6].

Once the kernel function is selected, the training of the support vector machines is reduced to a convex quadratic programming problem. The selection of an appropriate optimization approach is important since the number of variables can be enormous, and standard algorithms for QP are not appropriate. Since the number of variables is equal to the number of data points, and the quadratic form is completely dense, memory requirements grow with the square of the number of data points. Therefore, decomposition and scalability become important issues. In the present version we implemented a modification of Sequential Minimal Optimization[9], a convenient decomposition algorithm that does not require storage of the complete quadratic form.

4. RESULTS

Insofar as SVM is a newly emerging paradigm, many issues remain unresolved. One important component of the SVM approach is the kernel function, and optimal kernel selection remains a topic of intense investigation. For the present investigation, two kernels have been implemented and applied to nuclear power plant state estimation.

Most of the MSET kernels are empirical functions, and it is difficult to prove positive definiteness of the resulting similarity matrices. After extensive experimenting we selected for comparison the MSET kernel called the Vector Similarity Estimation Technique (VSET). For a sufficiently large number of explanatory variables with significant correlation, we have never observed cases where this kernel was not positive definite. The analytical form of this kernel is close to that of radial kernels, but with additional components that enable anisotropic correlation modeling in a multidimensional vector space.

To compare generalization errors for the standard MSET algorithm and the SVM algorithm we first used the SMORN VII Neural Network (NN) Benchmark[10]. Generated electric power in the

Borssele nuclear power plant in the Netherlands is estimated based on sensor readings for 11 power plant variables. For the training set of 600 patterns and the testing set of 942 patterns, the results are given in Figure 1.

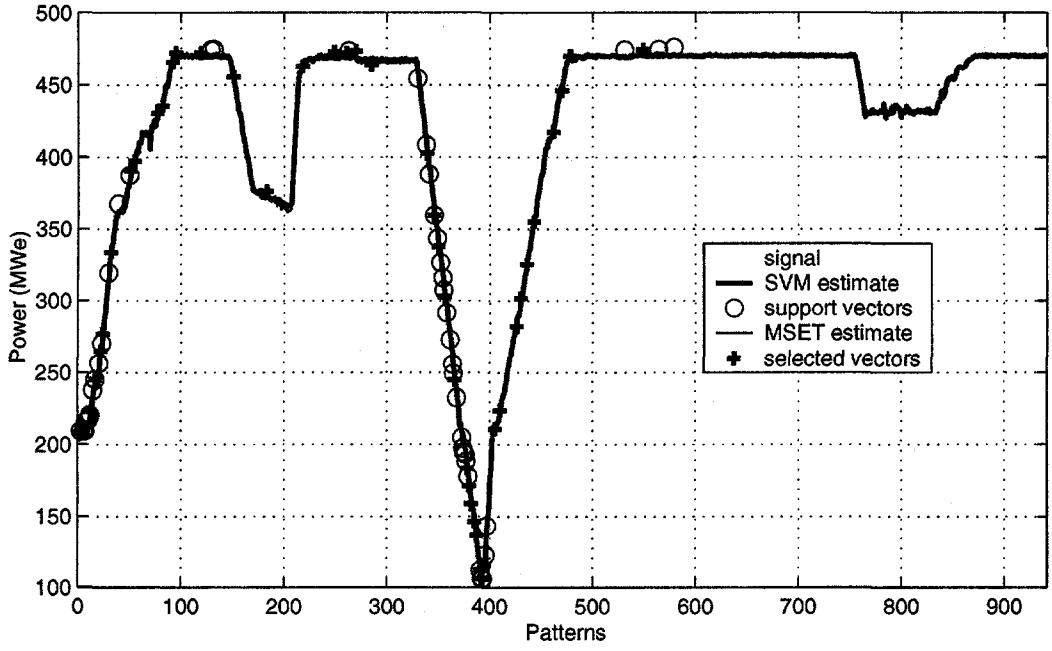

Fig 1. Comparison of SVM estimation and standard MSET for Neural Network Benchmark

After some experimenting, the value of the parameter C in the SVM algorithm was set to 480, a value slightly larger than the maximum of the data. Smaller values of C would sometimes cause artificial "clipping" of the data and introduce bias. Only 45 support vectors are needed to estimate the data with the accuracy around 1% of the maximum value (tolerance level $\varepsilon = 5$). With the same number of training vectors in MSET, selected using an unsupervised learning procedure called Vector Ordering[2], the estimation results are practically indistinguishable.

An extended data set from the NN Benchmark has been used to compare training and testing errors. The training set has 942 patterns, while the testing set has 1169. The results are given in Table I.

Training and testing errors are similar to the reported errors for artificial neural networks[10]. The improvement in generalization error in the SVM algorithm over the standard MSET with the same number of training vectors selected is modest. However, in Vector Ordering, the number of training vectors must be prescribed by the user *ad hoc*, while in SVM the support vectors are obtained automatically based on the optimization of the regularized functional.

Table I. Training and Testing Errors for NN Benchmark

SVM estimation				MSET Estimation		
Parameter ε (%)	Number of SV	Training Error (%)	Testing Error (%)	Number of Vectors	Training Error (%)	Testing Error (%)
0.5	130	0.45	2.09	130	0.58	2.16
1.0	50	0.74	1.89	50	0.61	2.01

As a second example we use 10 archive signals from the Experimental Breeder Reactor-II (EBR-II). We compare the VSET kernel and the Hermitian kernel (13) written in the following form

$$K(\vec{x},\vec{y}) = \exp(\alpha(\vec{x}\cdot\vec{y}))\exp(-\alpha\|\vec{x}-\vec{y}\|^2/2\cdot\sigma^2) \qquad (17)$$

to explicitly include a data-dependent normalization factor α and a free parameter σ that controls the width of the Gaussian. This parameter has effect on smoothing properties, as in standard kernel regression. Better adaptation to data could be obtained using a properly selected parameter σ. This parameter should follow the input data variability, but in practice it is not easy to find the proper value. In this paper we used a trial and error method, but some formal paramater selection procedures have been developed[11] and will be implemented in the future.

For the EBR II data we found that $\sigma = 0.1$ is appropriate. Comparison of the training and testing errors for VSET and the Hermite kernel is given in Table II.

Table II. Training and Testing Errors for EBR II data

SVM estimation with the VSET kernel				SVM Estimation with the Hermitian kernel		
Parameter ε	Number of SV	Training Error	Testing Error	Number of SV	Training Error	Testing Error
0.25	239	0.299	0.346	61	0.134	0.153
0.5	90	0.331	0.349	28	0.241	0.237

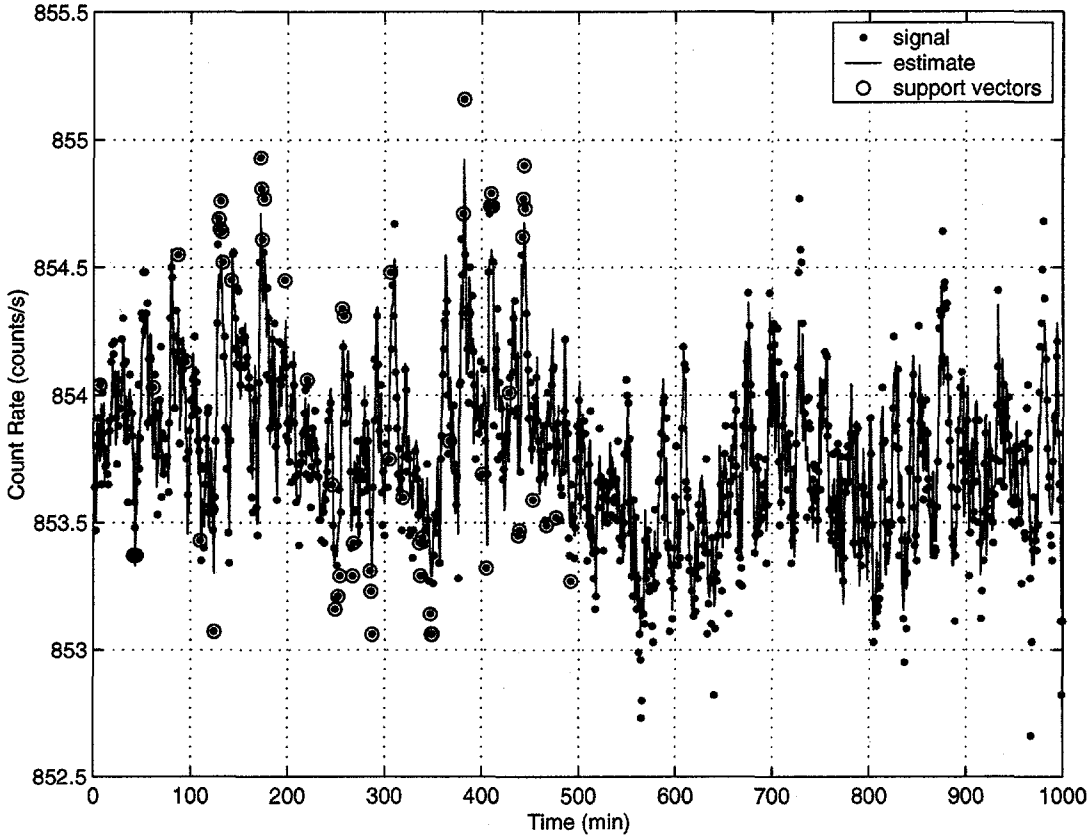

Figure 2. A signal from EBR II estimated using SVM with the Hermite kernel (ε =0.25)

In this case the Hermitian kernel outperforms the VSET kernel. Using a much smaller number of support vectors, the testing error is smaller for both values of ε selected.

Finally, we used a set of 10 correlated signals taken from a commercial nuclear power plant to show how to control smoothness using both parameters ε and σ. Figure 3 shows the results of the standard MSET estimation with VSET kernel and 18 training vectors (optimal number of support vectors for VSET in this case). Figure 4 shows the corresponding results of SVM estimation with the VSET kernel.

The testing error for standard MSET estimation is 0.96, while for MSET integrated with SVM the testing error is 0.92. Here, SVM generalization is not significantly better. It is evident from the figure that the solution obtained using SVM has smaller variance than the original signal (signified by dots in the figure).

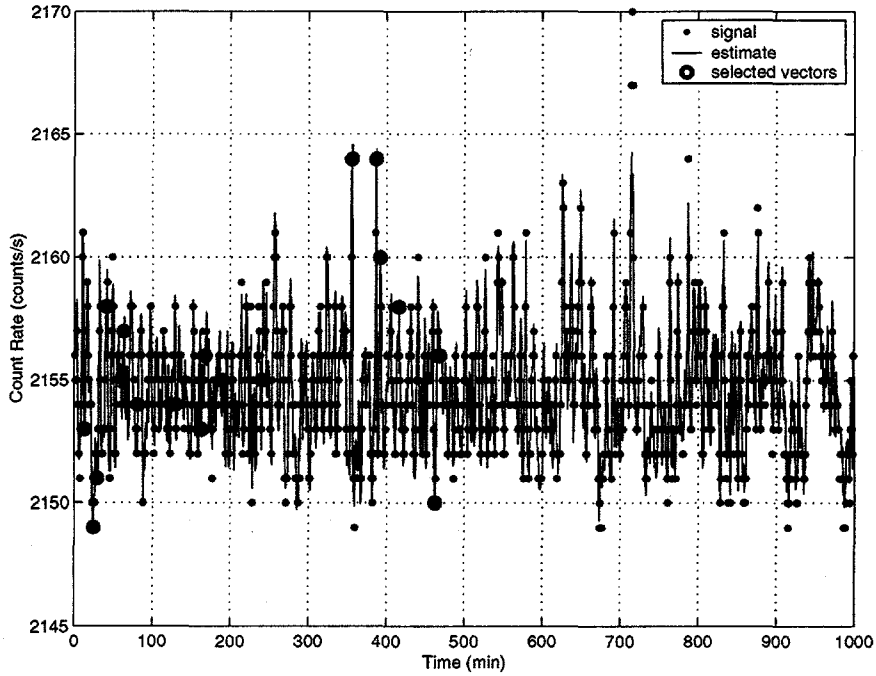

Figure 3 A commercial PWR signal estimated using standard MSET with a VSET kernel

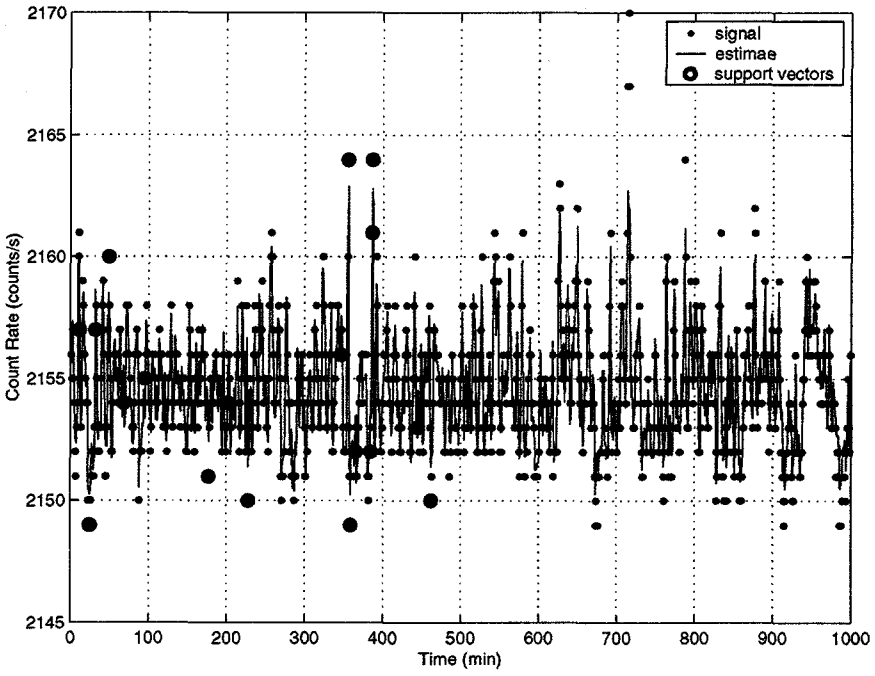

Figure 4. A commercial PWR signal estimated using SVM with a VSET kernel

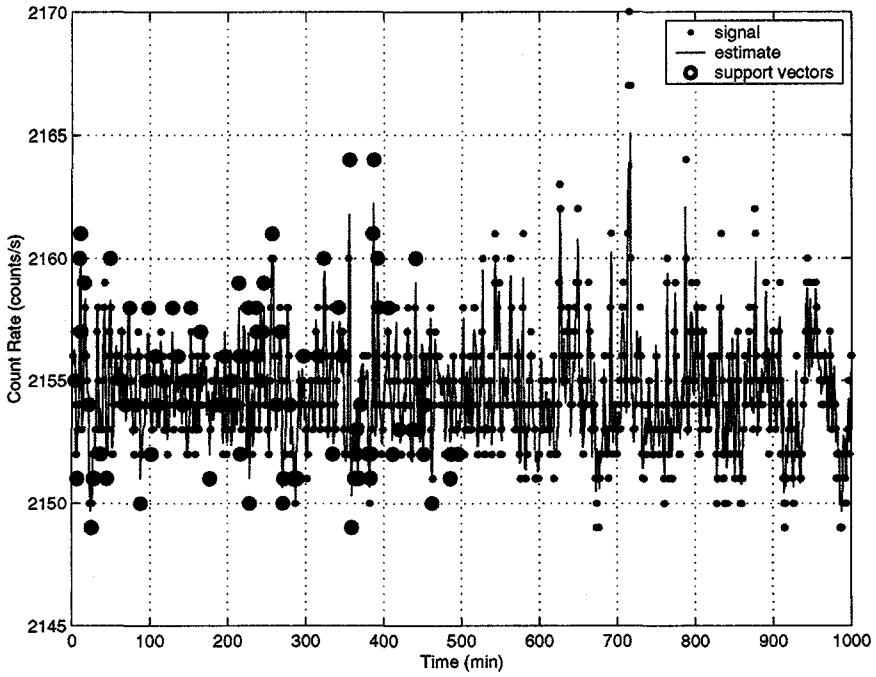

Figure 5. A commercial PWR signal estimated using SVM with the Hermitian kernel (σ=0.5)

Finally, Figure 5 shows the results of SVM estimation with the Hermitian kernel and $\sigma = 0.5$. This value was selected on the basis of trial and error. The effect of the parameter σ on the training data (first 500 observations) and the testing data can be observed from Figure 6. For $\sigma = 0.2$ the training error is the smallest, but the testing error is the largest. As σ increases, the training error increases. However, the increase in the training error is slower, and for $\sigma = 0.5$ both training and testing errors are small. Although the errors are close, or even better than SVM with the VSET kernel, the Hermitian kernel in this case requires a much larger number of training vectors.

The presented results are preliminary. Nevertheless, the examples presented here from the Neural Network Benchmark database, from EBR-II, and from an operating commercial PWR show an improvement of MSET when integrated with SVM as compared with the standard version of MSET. The SVM method results in variance reduction, near optimal selection of a compact subset of training vectors, and some improvement in generalization.

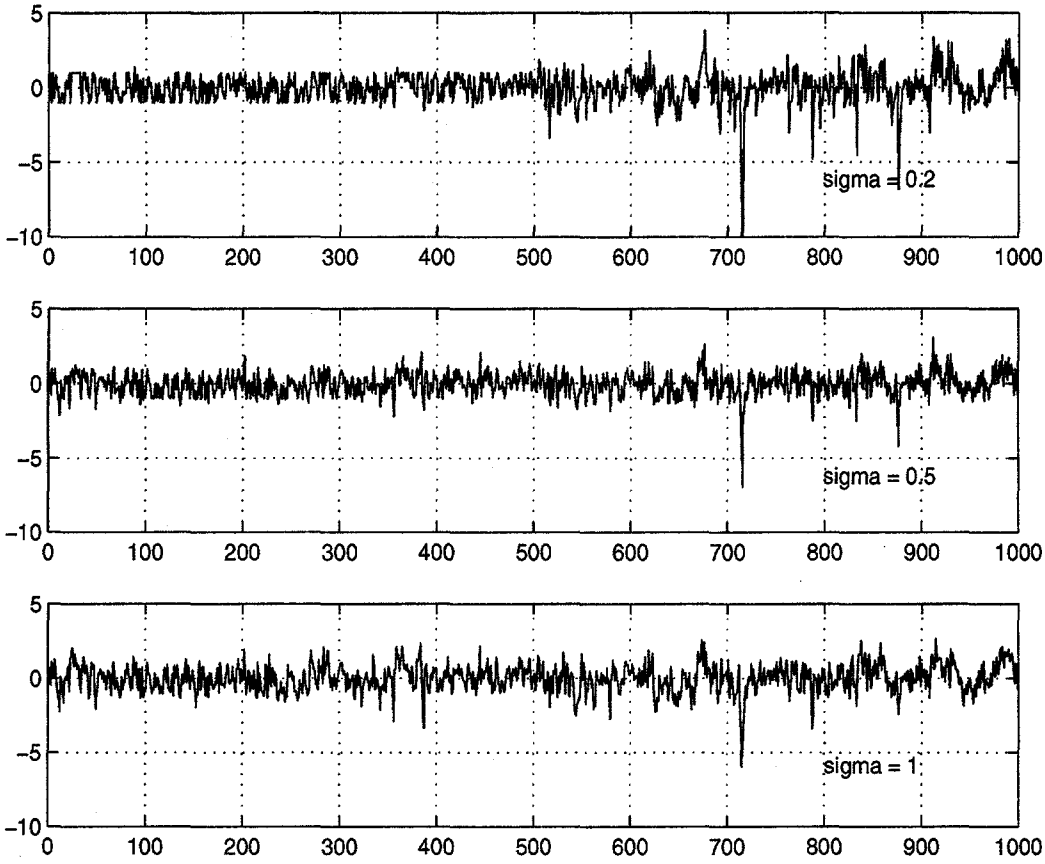

Figure 6. Dependence of the estimation error on the parameter σ

CONCLUSIONS

A recently proposed learning paradigm called the Support Vector Machine (SVM) has been applied to nuclear reactor state estimation. The generalization error of this methodology is improved by transformation of the input data into another space, called feature space, using an appropriately defined kernel function. In addition, optimal training is achieved using a convex quadratic programming approach that results in a unique solution. The selection of proper kernels is an open research problem. In this paper we investigated two kernels. The first kernel is a nonparametric function called the Vector State Estimation Technique (VSET) and is a component of the Multivariate State Estimation Technique (MSET) developed at Argonne National Laboratory. The other kernel is based on the Hermite polynomials and contains data-based parameters. We tested three real data sets from experimental and commercial nuclear power

reactors. For all cases investigated, application of the SVM algorithm shows improvement over the standard MSET algorithm.

REFERENCES

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

2. K. C. Gross, R. M. Singer, S. W. Wegerich, J. P. Herzog, R. Van Alstine, and F. Blockhorst, "Application of a Model-Based Fault Detection System to Nuclear Plant Signals," Proc. 9th Intl. Conf. On Intelligent Systems Application to Power Systems, Seoul, Korea, pp. 60-65 (1997).

3. A. N. Tikhonov and Y. Y. Arsenin, *Solutions of Ill-Posed Problems,* W.H. Winston, Washington (1977).

4. N. Zavaljevski, K. C. Gross and S. Wegerich, "Regularization Methods for the Multivariate State Estimation Technique (MSET)," Proc. MC'99 Conference on Mathematics and Computation, Reactor Physics and Environmental Analysis in Nuclear Applications, Madrid, Spain (1999).

5. G. Wahba, *Spline Methods for Observational Data,* SIAM, Philadelphia (1990).

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

7. F. Riesz and B. Sz.-Nagy, *Functional Analysis,* Ungar, New York (1955).

8. M. Grigoriu, *Applied Non-Gaussian Processes*, PTR Prentice Hall, Englewood Cliffs, NJ (1995).

9. J. Platt, "Fast Training of Support Vector Machines Using Sequential Minimal Optimization," in *Advances in Kernel Methods - Support Vector Learning*, ed. B. Sholkopf et al, MIT Press (1999).

10. O. Ciftioglu and E. Turkcan, "Neural Network Benchmark for SMORN-VII," OECD Report NEA/NSC/DOC(96)29 (1996).

11. N. Christianini, C. Campbell, J. Shawe-Taylor, "Dynamically Adaptive Kernels in Support Vector Machines," Neuro COLT Technical Report TR-1998-017, Royal Holloway College (1999).