Electrical Energy Consumption Forecasting in Oil Refining Industry Using Support Vector Machines and Particle Swarm Optimization

MILENA R. PETKOVIĆ
MILAN R. RAPAIĆ
BORIS B. JAKOVLJEVIĆ
Computing and Control Department
University of Novi Sad
Trg Dositeja Obradovića 6
SERBIA
milena5@uns.ac.rs http://ccd.ns.ac.rs

Abstract: - In this paper, Support Vector Machines (SVMs) are applied in predicting electrical energy consumption in the atmospheric distillation of oil refining at a particular oil refinery. During cross-validation process of the SVM training Particle Swarm Optimization (PSO) algorithm was utilized in selection of free SVM kernel parameters. Incorporation of PSO into SVM training process has greatly enhanced the quality of prediction. Furthermore, various (different) kernel functions were used and optimized in the process of forming the SVM models.

Key-Words: - Support Vector Machines (SVM), Kernel Functions, Particle Swarm Optimization (PSO), Electrical Energy Prediction, Oil Refining

1 Introduction

High quality prediction of consumption of fuels, including both electrical energy and fossil fuels (heating oil, natural and refined gas and steam) in the first phase of oil refining (atmospheric distillation) is vital for control and optimization of the oil refining process. Energy and fuel consumption are quantities of utmost importance, since they affect overall cost of the entire oil refining process, and consequently, the definite prices of all refined oil products. Fuel consumption analysis and predictions are therefore vital for production and an interesting field of research. In terms of prediction methods, Support Vector Machines (SVM) are attractive as relatively new, yet effective technique in modeling of complex functional correlations. Therefore, utilization of SVMs in estimating and predicting the consumption of various types of fuels for industrial systems is a promising approach, as stipulated by the present paper as well as by other authors [1].

The present paper advocates utilization of regression SVM model [2] with PSO algorithm incorporated in cross-validation phase of training. Support Vector Machines implement the principle of structural risk minimization in place of empirical risk minimization, which gives them excellent generalization ability in the situation of small training sample [3]. In addition, SVMs can change a nonlinear learning problem into a linear one, in order to reduce the algorithm complexity by using the kernel function idea (the "kernel trick"). At present, SVMs have been utilized in solving nonlinear regression estimation problems in financial time series forecasting [4], reliability prediction [5], power load forecasting [6] and many different problems[7, 8]. However, SVMs have rarely been applied to forecast fuel consumption in oil refining, though, it is the opinion of the authors, the technique has great potential in this area. A short account of SVM regression is given in section 2.

The main disadvantage of SVM is the necessity to set a number of parameters in advance. Standard procedure, known as cross-validation, is to make several consecutive trials with different parameter sets and then choose the set giving the best performance. In the present paper, the cross-validation procedure is conducted by PSO algorithm. The idea to use global optimization procedure in cross-validation process is not entirely new. Successful applications of genetic algorithm (GA) have previously been reported in literature [6]. PSO is novel optimization procedure, known for its efficiency, well adopted for solving non-convex, multimodal optimization problems. PSO is introduced in section 3. Application of PSO to cross-validation process is addressed in section 4. Results obtained in this study clearly demonstrate effectiveness of PSO-based cross-validation. In particular, prediction offset presented in the previous studies [10] was completely suppressed in the current one.

SVM models developed in the present paper were trained on a one year data-base consisting of 1) daily refining of oil, 2) daily usage of industrial units (in percents), 3) type of oil being refined, 4) the daily consumption of fuels (both electric energy and fossil fuels) and 5) climate conditions (season). The data concerns particular facility for atmospheric oil distillation, that we named facility A. This facility was selected based on analysis of the production process of the refinery and the fact that this facility uses a considerable fraction of the overall fuel consumed in the oil refining process. Results and conclusions are presented in section 5 and 6 respectively.

2 Support Vector Machines

The supervised learning algorithm attempts to learn the input-output relationship (dependency or function) f(x) by using a training data set {X = [x_i , y_i], i = 1, . . . , n} consisting of *n* pairs (x_1 , y_1), (x_2 , y_2), . . . (x_n, y_n), where the inputs x are m-dimensional vectors and the labels (or system responses) y are discrete (e.g.,Boolean) for classification problem and continuous values for regression tasks. Support Vector Machines (SVMs) and Artificial Neural Network (ANN) are two of the most popular techniques in this area [7, 8, 9, 11].

The learning task in regression is to find the underlying function between some m-dimensional input vectors x and scalar outputs y. The regression problems can also be found in many disciplines, including time-series analysis, control system, navigation and interest rates analysis in finance. There are two phases when applying supervised learning algorithms for problem-solving as shown in Figure 1. The first phase is the so-called learning phase where the learning algorithms design a mathematical model of a dependency, function or mapping (in an regression) or classifiers (in a classification i.e., pattern recognition) based on the training data given.

The second phase is the test and/or application phase. In this phase, the models developed by the learning algorithms are used to predict the outputs y of the data which are unseen by the learning algorithms in the learning phase. Before an actual application, the test phase is always carried out for checking the accuracy of the models developed in the first phase.

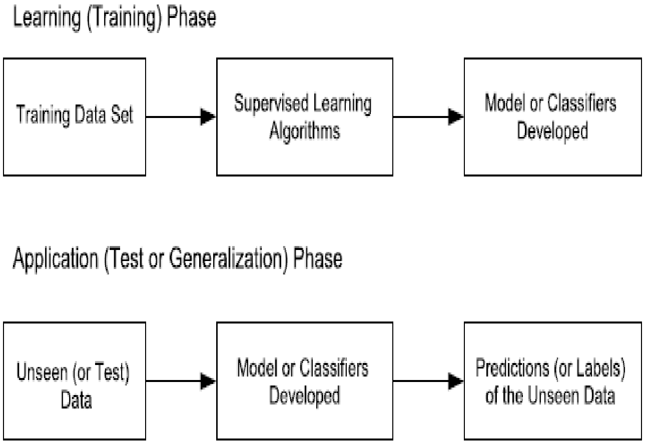

Learning (Training) Phase
Training Data Set
Supervised Learning Algorithms
Model or Classifiers Developed
Application (Test or Generalization) Phase
Unseen (or Test) Data
Model or Classifiers Developed
Predictions (or Labels) of the Unseen Data

Fig 1. *Two Phases of Supervised Learning Algorithms*

2.1 Support Vector Regression

The general regression learning problem is set as follows: the learning machine is given *n* training data pairs from which it attempts to learn the input-output relationship $y = f(x)$. A training data set $X = \{(x_i, y_i), i = 1,..., n\}$ consists of *n* training pairs. The inputs $\boldsymbol{x}$ are m- dimensional vectors, while the target outputs *y* are real valued scalars. We introduce all the relevant and necessary concepts of SVM regression starting with a linear regression hyperplane $\hat{y} = f(x, w, b)$ given as

$$\hat{y} = f(x, w, b) = w^T x + b \qquad (1)$$

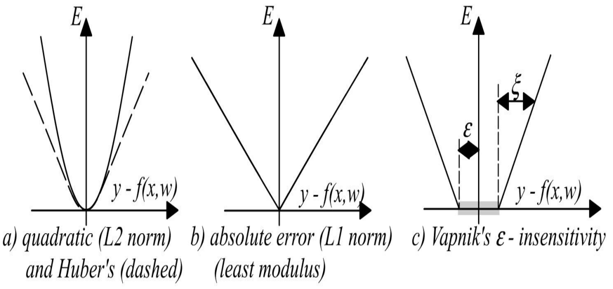
where $\hat{y}$ is predicted output, x is input pattern, w is weight vector and *b* is bias [12, 13]. Both weight and bias are set during the training process.

The most important difference of SVM with respect to classical regression techniques is the use of a novel loss (error) function [2] – Vapnik's linear loss function with ε-insensitivity zone, defined as

$$\begin{aligned} E(x, y, f) &= |y - f(x, w)|_\varepsilon = \\ &= \max(0, |y - f(x, w)| - \varepsilon) \end{aligned} \qquad (2)$$

Thus, the loss is equal to zero if the difference between the predicted $f(x, w)$ and the measured value *y* is less than ε. In contrast, if the difference is larger than ε, this difference is used as the error. In other words, Vapnik's error (loss) function (2) defines a “ε tube” as shown in Fig 3. If the predicted value is within the tube, the loss is zero; for all other predicted points outside the tube, the loss equals the magnitude of the difference between the predicted value and the radius ε of the tube.

The two classic error functions are: a square error, i.e., L2 norm $(y - f)^2$, as well as an absolute error, i.e., L1 norm, least modulus $|y - f|$ introduced by Rudjer Boskovic in 18th century [12]. The latter error function is related to Huber’s error function. An application of Huber’s error function results in a robust regression. It is the most reliable technique if nothing specific is known about the model of a noise. We do not present Huber’s loss function here in analytic form. Instead, we show it by a dashed curve in Figure 2.a. In addition, Figure 2. shows typical shapes of all mentioned error (loss) functions above.



a) quadratic (L2 norm) and Huber's (dashed) b) absolute error (L1 norm) (least modulus) c) Vapnik's ε - insensitivity

Fig 2. *Loss (Error) functions*

It can be shown [14] that generalization ability of the SVM depends on the magnitude of the weight vector: the smaller the magnitude $\|w\|$ the greater the generalization ability of the SVM becomes. Therefore, linear regression hyperplane is constructed by minimizing

$$R = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\left|y - f(x,w)\right|_{\varepsilon} \tag{3}$$

where C is a positive constant (regularization parameter) [14].

From (2) and Fig 1 it follows that for all training patterns one can define positive quantities known as slack variables

$$\begin{aligned} \xi &= \max(y - f(x,w) - \varepsilon, 0) \\ \xi^* &= \max(f(x,w) - y - \varepsilon, 0) \end{aligned} \tag{4}$$

Notice that at least one of these quantities is equal to zero for each training pattern. For patterns inside the tube, both of them are zero. Thus, the minimization of the risk R above equals the minimization of

$$R_{\xi} = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\left(\xi + \xi^*\right) \tag{5}$$

under constraints

$$y_i - w^T x_i - b \le \varepsilon + \xi_i \tag{6}$$

$$-y_i + w^T x_i + b \le \varepsilon + \xi_i^* \tag{7}$$

$$\xi_i \ge 0, \xi_i^* \ge 0 \tag{8}$$

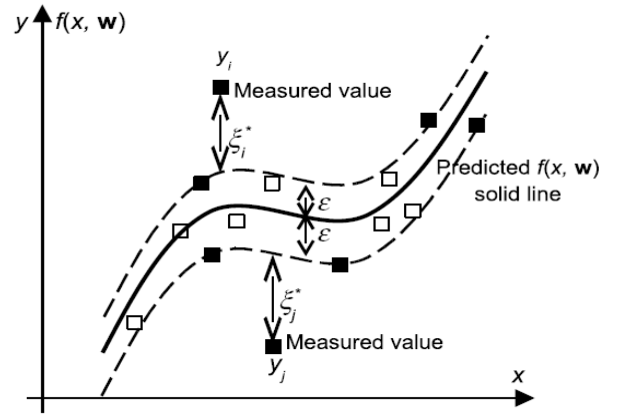


Fig 3. The parameters used in (1-D) Support Vector regression. Filed data are support vectors

The first term in (5) is weight decay, which is used to regularize weight size and penalize large weights. The second term is the empirical error(risk) which is scaled by ε- insensitive loss function(2). Parameter C is the regularization constant determing the compromise between the empirical error and the regularized term. Both C and ε need to be chosen empirically.

By introducing Lagrange multipliers and kernel function, the optimal regression function (1) is obtained in the following explicit form:

$$f(\mathbf{x}) = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)K(\mathbf{x_i}, \mathbf{x}) + b \tag{9}$$

where α_i and α_i^* are the Lagrange multipliers satisfying $\alpha_i\, \alpha_i^*$=0, $\alpha_i \ge$ 0, $\alpha_i^* \ge$ 0. Based on Karush- Kuhn- Tucker(KKT) condition only a certain number of coefficients $(\alpha_i - \alpha_i^*)$ are non-zero. The data pairs corresponding to these non-zero coefficients are named support vectors. $K(\mathbf{x}_i, \mathbf{x})$ is the symmetric kernel function(a symmetric function is a kernel if it fulfills Mercer's theorem[14]) used to avoid the computation of the nonlinear mapping.

SVM is a kernel-based algorithm. A kernel is a function that transforms the input data to a high-dimensional space where the problem is solved. Kernel functions can be linear or nonlinear.

We generated and simulated the processes of SVMs with the use of LIBSVM software[15] - library of functions needed for creating models of support vector machines - which also includes the implementation of libraries of functions for resolution of regression problems. In order to do this it is required to prepare the data to be adequate for training of the SVM models and then modify it back in the form recognizable for the LIBSVM software.

3 PSO Algorithm

Particle Swarm Optimization (PSO) algorithm has been introduced for the first time by Kennedy and Eberhart [16] as a new population based optimization technique inspired by animal social behavior. The algorithm investigates solution space using a set of vectors, usually referred to as "particles". Each particle is a potential solution, and the entire set is referred to as the "population", or sometimes as the "swarm". A particle is described by its position (x) and speed (v), and is able to memorize the position with the highest fitness value it has achieved so far (p), the so called personal best position.

Initially, the swarm is randomly dispersed within the search space, and random velocity is assigned to each particle. Particles interact by sharing information. Although different patterns of interactions have been investigated in literature, we focus our efforts to the so-called "star topology", also known as the "gBest PSO model" [17]. In this setting, the swarm as a whole memorizes the best position achieved so far by any of its particles (g), the so called global best position. At each step a particle caries over a portion of its previous speed, and is, in addition, simultaneously accelerated towards its personal best position and the best position found by any other particle in the swarm. Therefore, each particle explores the search space according to its current state (represented by the current position and current velocity) and its own memory (represented by personal best position), but also according to the collective knowledge of the entire swarm (represented by the global best position).

Dynamics of each particle is, therefore, determined by the following set of equations

$$\begin{aligned} v[k] &= w \cdot v[k-1] + cp \cdot rp[k] \cdot (p[k] - x[k]) + \\ &+ cg \cdot rg[k] \cdot (g[k] - x[k]) \end{aligned} \tag{10}$$

$$x[k+1] = x[k] + v[k] \tag{11}$$

Positive parameter *w*, the "inertia weight", was introduced by Shi and Eberhart [18] in an attempt to control diversity of the swarm during the optimization process. It is generally true, in population-based optimization methods, that high diversity is necessary in early stages of the search in order to fully investigate the search space and reduce possibility of being trapped in local optimum. On the other hand, in later stages, algorithm should focus on fine-tuning good solutions already found, so reduced diversity is desirable. Shi and Eberhart found that considerable improvements in performance of the original PSO are achieved by linear decreasing inertia over the generations from 0.9 to 0.4. Positive coefficients cp and cg are usually called the "acceleration factors" [17]. Random values *rp* and *rg* are mutually independent and uniformly distributed in range [0, 1]. Factor cp is sometimes referred to as the "cognitive" parameter, while cg is referred to as "social" parameter [17]. Due to original work of Kennedy and Eberhart, it is common choice to set both acceleration factors equal to 2. However, it is known that relatively high cognitive component enhances exploration, while relatively high social component forces particles to cluster.

Acknowledging this fact, Ratnaweera *et al.* suggested [19] that time-varying acceleration coefficients may further improve performance of the optimizer. They reported improvements for most of the benchmarks when decreasing *cp* form 2.5 to 0.5, and simultaneously increasing *cg* from 0.5 to 2.5. This variant of the PSO algorithm, known as the Time-Varying Acceleration Coefficients PSO (TVAC-PSO) is utilized in this paper. It allows each particle within the swarm to investigate the search space freely in the early stages of the search. This is due to the fact that each particle is affected more by its personal than by the global knowledge at the first few iterations. Later, the effect of the global knowledge prevails, and the swarm as a whole focuses on fine-tuning a number of good solution found previously. At the very end, the entire swarm converges to a very small region of the search space.

It has been shown that TVAC-PSO performs very well in a number of cases.

4 PSO - Based Cross Validation

The selection of SVM parameters, namely C, ε and parameters of kernel function, is important for the forecasting accuracy. Selecting appropriate values of these parameters is crucial in gaining excellent forecasting performance. In this paper, we chose Radial Basis Function for a kernel function and it was necessary to select its widths.

However, it is not known beforehand what values of the parameters are appropriate. Therefore, PSO is used to optimize parameters in the proposed SVM model. The schematic diagram of PSO-based cross-validation is presented in Fig 4.

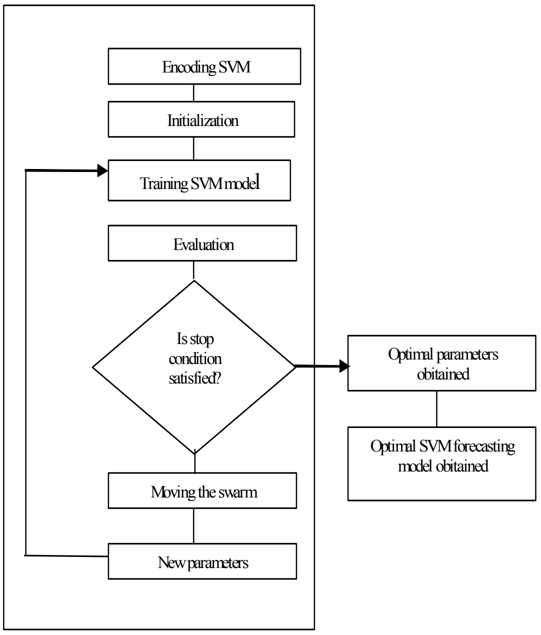

Encoding SVM
Initialization
Training SVM model
Evaluation
Is stop condition satisfied?
Optimal parameters obitained
Optimal SVM forecasting model obitained
Moving the swarm
New parameters

Fig 4. *Schematic diagram of PSO- based cross validation*

5 Results

5.1 PSO vs. Classical Cross- Validation (RBF Kernel Function)

In following examples, output is the amount of electrical energy consumed (in kW), while the inputs are: the amount of refined oil (in tons), the type of oil, plant utilization (in %), consumption of other fuels (heating oil, natural and refined gas and steam), as well as critical events such are plant shut-down and restart.

Kernel function for both examples is Radial Basis Function(RBF)

$$K(x_i, x_j) = \exp(-\gamma \left\| x_i - x_j \right\|^2) \qquad (12)$$

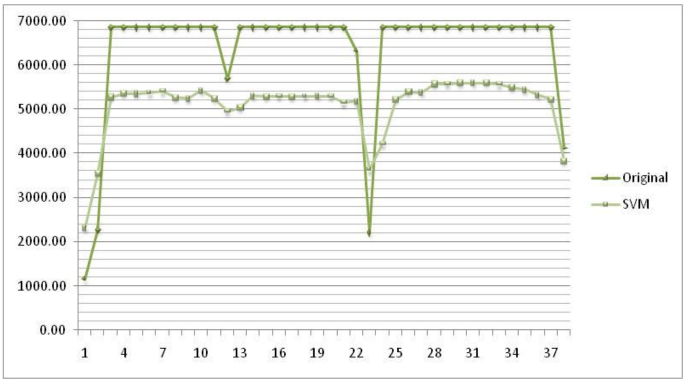


Fig 5. *Test-set prediction results with classical cross-validation (LIBSVM implementation). The ordinate depicts kW of power consumed.*

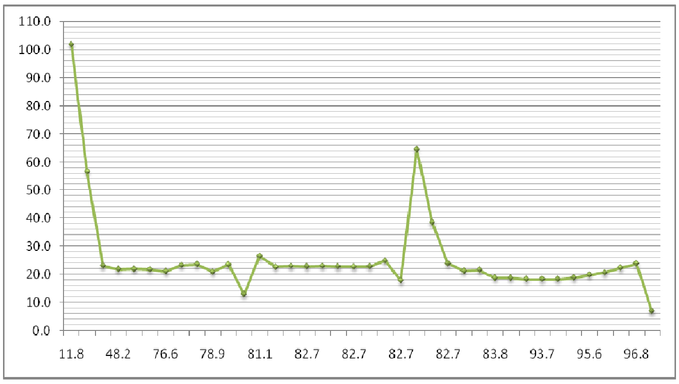


Fig 6. *Relative test-set prediction errors (in %) with classical cross-validation (LIBSVM implementation) as a function of plant utilization (in %).*

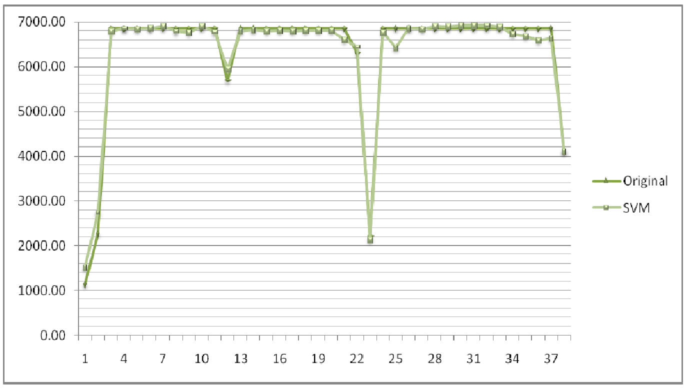


Fig 7. *Test-set prediction results with PSO-based cross-validation. The ordinate depicts kW of power consumed.*

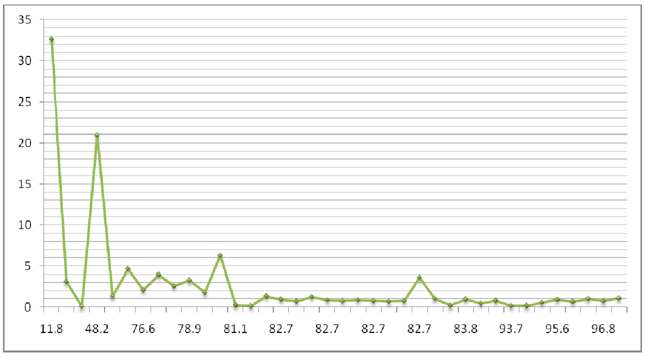


Fig 8. *Relative test-set prediction errors (in %) with PSO-based cross-validation as a function of plant utilization (in %).*

Fig 5. and Fig 7. depict true and predicted power consumption using classical and PSO-based cross-validation, respectively. It is clear that, when applying classical cross-validation there is a problem with offset, as well as with peaks occurring during critical events. These errors were also reported in previous studies [10]. PSO-based SVM regression proposed in the current paper is not prone to such errors. Relative errors presented in Fig 6 and Fig 8 only confirm the effectiveness of our method. However, conclusions presented in [10] concerning the importance of sufficient plant utilization for high quality prediction remain.

5.2 Different Kernel Functions

The kernel function as well as parameters selection is a very important problem in the research of support vector machine.

Many kernel mapping functions can be used – probably an infinite number. But a few kernel functions have been found to work well in for a wide variety of applications. The default and recommended kernel function is the Radial Basis Function (RBF) we used in 5.1.

5.2.1 Linear Kernel Function

In following examples, output is the amount of electrical energy consumed (in kW), while the inputs are: the amount of refined oil (in tons), the type of oil, plant utilization (in %), consumption of other fuels (heating oil, natural and refined gas and steam), as well as critical events such are plant shut-down and restart.

Kernel function is linear

$$K(x_i, x_j) = x_i^T x_j \qquad (13)$$

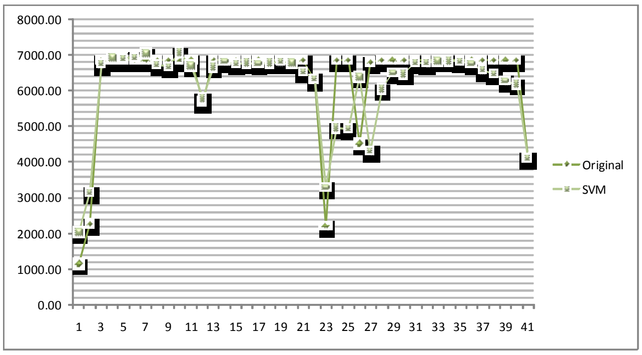


Fig 9. *Test-set prediction results with PSO-based cross-validation and Linear kernel function. The ordinate depicts kW of power consumed.*

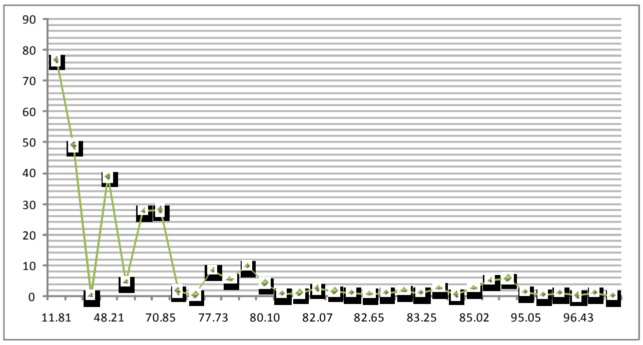


Fig 10. *Relative test-set prediction errors (in %) with PSO-based cross-validation(Linear kernel function) as a function of plant utilization (in %).*

5.2.2 Polynomial Kernel Function

In following example, output is the amount of electrical energy consumed (in kW), while the inputs are: the amount of refined oil (in tons), the type of oil, plant utilization (in %), consumption of other fuels (heating oil, natural and refined gas and steam), as well as critical events such are plant shut-down and restart.

Kernel function is polynomial

$$K(x_i x_j) = (\gamma x_i^T x_j + r)^d \qquad (14)$$

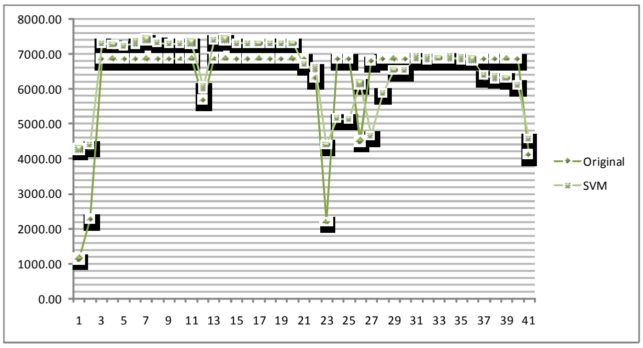


Fig 11. *Test-set prediction results with PSO-based cross-validation and Polynomial kernel function. The ordinate depicts kW of power consumed.*

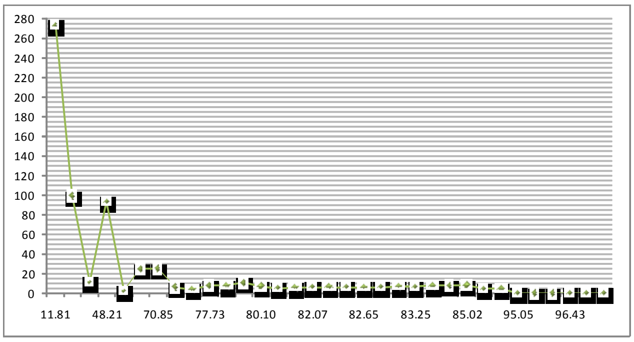


Fig 12. *Relative test-set prediction errors (in %) with PSO-based cross-validation(Polynomial kernel function) as a function of plant utilization (in %).*

5.2.3 Sigmoid Kernel Function

In this example, output is the amount of electrical energy consumed (in kW), while the inputs are: the amount of refined oil (in tons), the type of oil, plant utilization (in %), consumption of other fuels (heating oil, natural and refined gas and steam), as well as critical events such are plant shut-down and restart.

Kernel function is sigmoid

$$K(x_i, x_j) = \tanh(\gamma x_i^T x_j + r) \qquad (15)$$

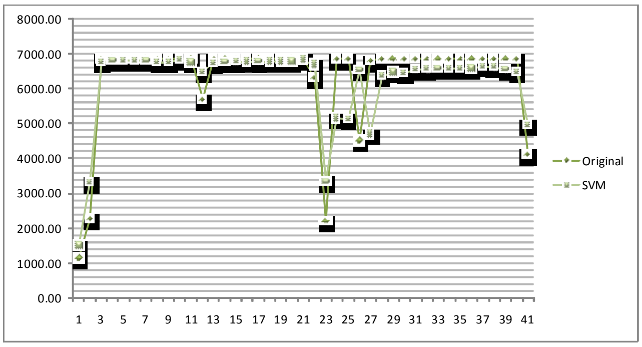


Fig 13. *Test-set prediction results with PSO-based cross-validation and Sigmoid kernel function. The ordinate depicts kW of power consumed*

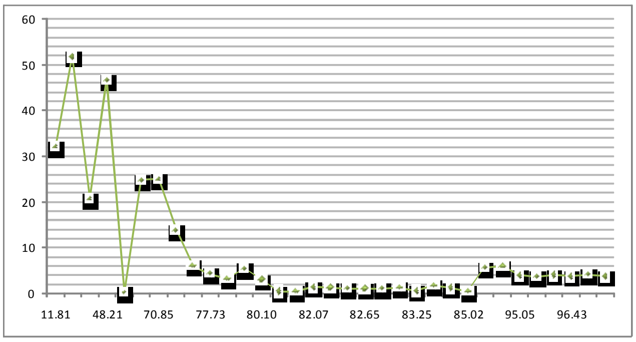


Fig 14. *Relative test-set prediction errors (in %) with PSO-based cross-validation (Sigmoid kernel function) as a function of plant utilization (in %).*

5.3 Standard Refinery Fuel

Commonly, the consumption of fuels in oil refining processes is expressed in tons of Standard Refinery Fuel [20,21].

Standard Refinery Fuel (SRF) is a reference fuel whose lower heating value is 9673 kcal/kg.

Formulas for conversion of consumption of different fuels in tones of SRF are shown in Table 1.

Table 1. *Different fuels in tones of Standard Refinery Fuel*

Fuel	Energy value	Tones of SRF
Heating oil	40500 MJ/kg	1
Natural gas	39000 MJ/kg	0.9629
Refined gas	48000 MJ/kg	1.1851
Steam	2382 MJ/kg	0.0588

In following examples, output is the amount of electrical energy consumed (in kW), while the inputs are: the amount of refined oil (in tons), the type of oil, plant utilization (in %), sum of consumptions of all other fuels (heating oil, natural and refined gas and steam) expressed in tons of SRF, as well as critical events such are plant shut-down and restart.

5.3.1 Kernel function is Radial Basis Function

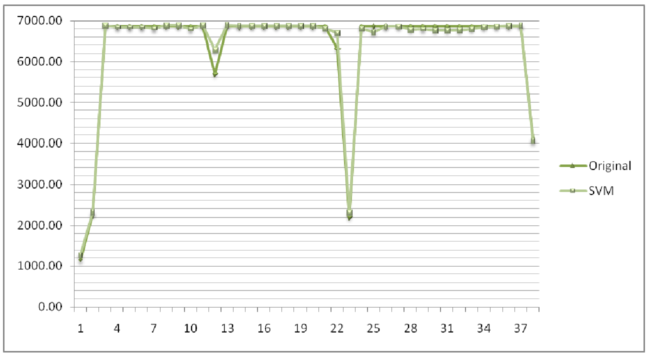


Fig 15. *Test-set prediction results with PSO-based cross-validation and fuel consumptions expressed in tons of SRF(RBF Kernel). The ordinate depicts kW of power consumed.*

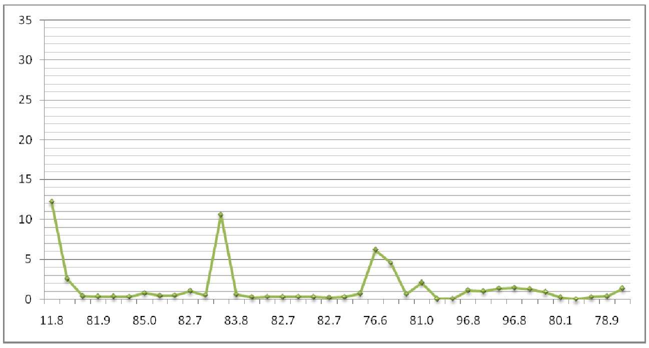


Fig 16. *Relative test-set prediction errors (in %) with PSO-based cross-validation (RBF kernel) and fuel consumptions expressed in tons of SRF as a function of plant utilization (in %).*

Fig. 15 and Fig. 16 show that the modeling of input parameters of fuel consumption over SRF additionally enhances the quality of prediction of energy consumption in oil refining processes.

5.3.2 Different Kernel Functions

For different kernel function we give relative test-set prediction errors (in %) with PSO-based cross-validation and fuel consumptions expressed in tons of SRF as a function of plant utilization.

1) SVM with Linear Kernel Function

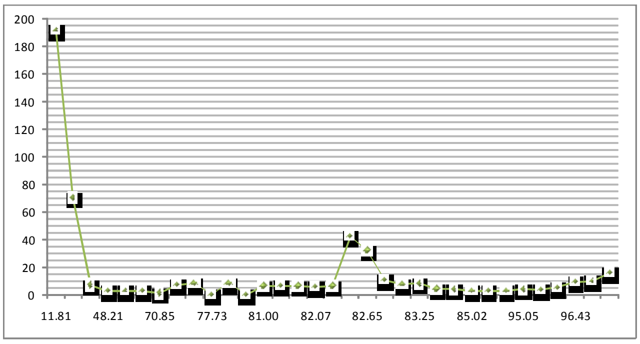


Fig 17. *Relative test-set prediction errors (in %) with PSO-based cross-validation(Linear kernel) and fuel consumptions expressed in tons of SRF as a function of plant utilization (in %).*

2) SVM with Polynomial Kernel Function

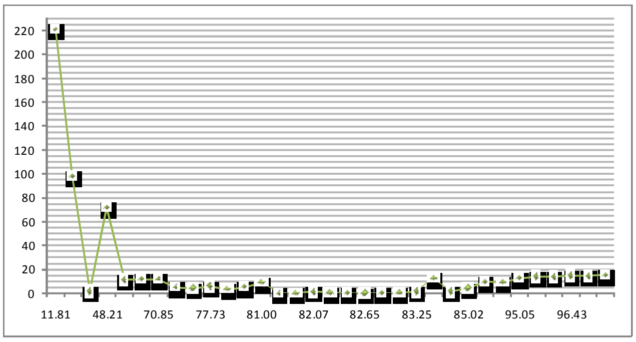


Fig 18. *Relative test-set prediction errors (in %) with PSO-based cross-validation (Polynomial kernel) and fuel consumptions expressed in tons of SRF as a function of plant utilization (in %).*

3) SVM with Sigmoid Kernel Function

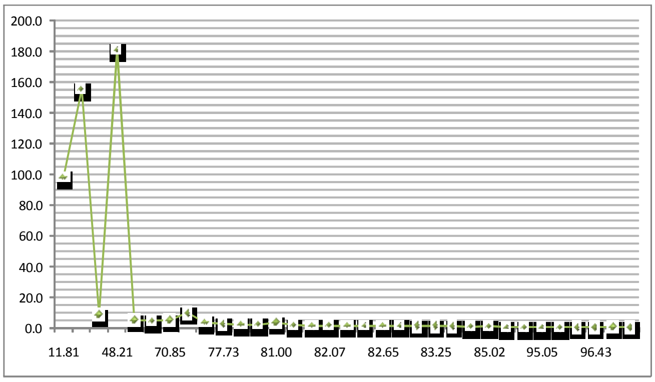


Fig 19. *Relative test-set prediction errors (in %) with PSO-based cross-validation (Sigmoid kernel) and fuel consumptions expressed in tons of SRF as a function of plant utilization (in %).*

5.4 Accuracy of SVM models

Validation of the models was required to test the accuracy of the methods as well as to enable comparison between them. In this study, each data set was divided into training set for model development and test set for external prediction.

The construction of the test set was accomplished by insisting that members of the test set be representative of all members of the training set in terms of the ranges of experimental values. Initially, all the predictive model underwent a leave-one-out (LOO) procedure.

Mean Absolute Percentage Error (MAPE) is a method for measuring the accuracy of a forecast by summing the absolute percentage error. It's commonly used in quantitative forecasting methods because it produces a measure of relative overall fit.

$$\text{MAPE}=\frac{1}{n}\sum_{i=1}^{n}\frac{y_i-\hat{y}_i}{y_i},\ \ i=1,\ldots,n \qquad (16)$$

where y_i are actual values, $\hat{y}_i$ are predicted values and n is number of data points.

In the Table 2 we show Mean Absolute Percentage Error (MAPE) for all SVM models that we simulated.

Table 2. *Mean Absolute Percentage Error for different SVM models*

Kernel	Parameters Selection Algorithm	MAPE
Linear	Cross-validation	14,5247
Polynomial	Cross-validation	28,6505
RBF	Cross-validation	25,7000
Sigmoid	Cross-validation	30,7587
Linear	PSO	9,4693
Polynomial	PSO	10,7361
RBF	PSO	2,70611
Sigmoid	PSO	8,8362
Linear	PSO +SRF inputs modeling	16,1513
Polynomial	PSO +SRF inputs modeling	16.4626
RBF	PSO +SRF inputs modeling	1,4646
Sigmoid	PSO +SRF inputs modeling	14,2482

6 Conclusion

This paper is dedicated to the problem of predicting the energy consumption in oil refining process using SVM regression with cross-validation based on PSO algorithm. We used the SVM method in which the parameters were determined by the PSO algorithm, a new method for solving this type of problems. The results of this paper clearly demonstrate the effectiveness of the proposed procedure in comparison to the classical cross-validation.

Several number of different kernel functions were used. It has been shown that PSO based parameter selection outperforms classical cross-validation in all of the considered cases. Especially good results were obtained with radial basis function kernel.

Prediction based standard refinery fuel was also demonstrated with different kernel function. In this case, also, the parameters of kernel functions were selected using PSO based procedure. The obtained results demonstrate, once again, the effectiveness of PSO based cross-validation strategy.

References:

[1] S. Živković, "Primena Support vector machines u predikciji potrošnje električne energije", (in Serbian) MSc Thesis, Faculty of Technical Sciences, Novi Sad, Serbia, 2006.

[2] A. Smola and B. Schölkopf, *"A Tutorial on Support Vector Regression"*, Statistics and Computing 14: 199-222, (2004), Kluwer Academic Publishers. Manufactured in The Netherlands, (2004)

[3] H. Shin, S. Cho, Response modeling with support vector machines, *Expert Syst. Appl.* 30 (4) (2006) 746–760.

[4] K.-j. Kim, Financial time series forecasting using support vector machines, *Neurocomputing* 55 (1–2) (2003) 307–319.

[5] X. P. Zhang, R. Gu, Electrical energy consumption forecasting based on cointegration and a support vector machine in China, *WSEAS Transactions on Mathematics,* Issue 12, Vol. 6, 2007, pp. 878-883.

[6] P.-F. Pai,W.-C. Hong, Forecasting regional electricity load based on recurrent support vector machines with genetic algorithms, *Electric Power Syst. Res.* 74 (3) (2005) 417–425.

[7] G. Pajares, Support vector machines for shade identification in urban areas, *WSEAS Transactions on Information Science and Applications*, Vol. 2, No. 1, 2005, pp. 38-41

[8] Wei Gao, Ning Wang, Ocean Ambient Noise Prediction Based on Support Vector Machine, *WSEAS Transactions on Information Science and Applications*, Vol. 4, No. 2, 2007, pp. 1341-1345.

[9] Maher I. Rajab, Khalid A. Al-Hindi, Analysis of Neural Network Edge Pattern Detectors in Terms of Domain Functions, *WSEAS Transactions on Information Science and Applications*, Vol. 5, No. 2, 2008, pp. 1341-1345.

[10] Milena Petrujkić, Marijana Bobar, Olivera Papić „Predikcija potrošnje energenata u primarnoj preradi nafte primenom Support Vector Machines", (in Serbian) ETRAN, Herceg Novi 2007.

[11] S. Chitwong, S. Witthayapradit, S. Intajag, Multispectral Image Classification Using Back Propogation Neural Network in Pca domain, pp 489-511, *proceeding of WSEAS International Multi Conference, Izmir, Turkey, Sep. 13-16, 2004.*

[12] B. Schölkopf, A. Smola, *Learning with Kernels, Supprt Vector Machines*, Regularization, Optimization and Beyond, MIT Press, Cambridge, MA, 2002.

[13] Kecman V., High Dimensional Function Approximation (Regression, Hypersurface Fitting) by an Active Set Least Squares Learning Algorithm*, School of Engineering Report 643*, The University of Auckland, Auckland, NZ, (53 p.), 2006

[14] V. Kecman, *Learning and Soft Computing, Support vector machines, Neural Networks and Fuzzy Logic*, The MIT Press, Cambridge, MA, (2001)

[15] C. C. Chang and C. J. Lin (2001) LIBSVM: A Library for Support Vector Machines [Online]. Available: http://www.csie.ntu.edu.tw/~cjlin/libsvm

[16] J. Kennedy, R.C. Eberhart, "Particle Swarm Optimization", *Proc. of IEEE Int. Conf. on Neural Networks, Perth, Australia* (1995) 1942-1948.

[17] F. van den Bergh, "An analysis of particle swarm optimizers", *PhD Thesis*, University of Pretoria, 2001.

[18] Y. Shi, R.C. Eberhart, "Empirical study of particle swarm optimization", *Proc. IEEE Int. Congr. Evolutionary Computation*, vol 3, 1999, 101-106

[19] A. Ratnaweera, et al, "Self-Organizing Hierarchical Particle Swarm Optimizer With Time-Varying Acceleration Coefficients*", IEEE Transactions on Evolutionary Computation,* vol. 8, no. 3, June 2004, 240-255

[20] G. G. Rayan, *Practical energy efficiency optimization*, ISBN: 978-1-59370-051-5

[21] http://www.shell.com