World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:1, No:3, 2007

An Exact Solution to Support Vector Mixture

Monjed Ezzeddinne, Nicolas Lefebvre, and Régis Lengellé

***Abstract*—**This paper presents a new version of the SVM mixture algorithm initially proposed by Kwok for classification and regression problems. For both cases, a slight modification of the mixture model leads to a standard SVM training problem, to the existence of an exact solution and allows the direct use of well known decomposition and working set selection algorithms. Only the regression case is considered in this paper but classification has been addressed in a very similar way. This method has been successfully applied to engine pollutants emission modeling.

***Keywords*—**Identification, Learning systems, Mixture of Experts, Support Vector Machines.

I. Introduction

MIXTURE of Experts (MoE) was introduced in [3] and [5] to deal with functional approximation and interpolation problems, with the aim not only to improve the global accuracy by combining results of expert functions but also to overcome the curse of dimensionality.

Support Vector Machines (SVM), which are well known for their generalization abilities, that result from the structural risk minimisation principle [9][10], have been introduced by Kwok [6] into the mixture domain. He showed that the solution is obtained by solving a Quadratic Programming (QP) problem very similar to that of SVM, for both classification and regression applications.

A deeper look into Kwok's SVM mixture method and the classical SVM QP problem shows some slight differences between them. These differences show inexistence of a solution to the SVM mixture problem. This results from the use of the Least Squares method to compute a certain vector bias term.

Even if an approximate solution could be considered as satisfactory, inexistence of an exact solution prevents the correct use of training algorithms in the large dataset case, such as, for example, decomposition algorithms. This results from the impossibility to verify Karush Kuhn Tucker (KKT) conditions.

In order to translate the ill-posed problem into a well-posed one, we introduce in this paper a slight modification of the mixture model. Moreover we show that:

- the modified QP problem is exactly the same as classical regression SVM problem
- so an exact solution exists (KKT conditions can be fulfilled)
- any large dataset training algorithm directly applies, including, for example, Joachim's working set selection algorithm for decomposition methods.

This paper is organized as follows. In the next section we remind the optimization problem associated to SVM regression. In Section 3 we briefly present the SVM mixture method proposed by Kwok. Modifications to the mixture model, that ensure the existence of an exact solution to the optimization problem, are provided in section 4. Experiments on simulated data and on a real world application are presented in section 5. The last section gives some concluding remarks.

II. SVM Regression (SVR)

For Support Vector Regression, using the ε-insensitive cost function, the dual problem can be written as follows:

$$\min_{\boldsymbol{\alpha},\boldsymbol{\alpha}^*} \left.\begin{array}{l} \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left(\alpha_i-\alpha_i^*\right)\left(\alpha_j-\alpha_j^*\right)K(\mathbf{X}_i,\mathbf{X}_j)+ \\ \qquad \varepsilon\sum_{i=1}^{N}(\alpha_i+\alpha_i^*)+\sum_{i=1}^{N}y_i(\alpha_i-\alpha_i^*) \\ \text{s.t.}\begin{cases} 0\le\alpha_i,\alpha_i^*\le C,\ i\in\{1,...,N\} \\ \sum_{i=1}^{N}\left(\alpha_i-\alpha_i^*\right)=0 \end{cases} \end{array}\right\} DPR_1$$

For each input vector $\mathbf{X}_i, i=1...N$, two Lagrange parameters are introduced: α_i and α_i^*. They must verify $\alpha_i\alpha_i^*=0, \forall i$. ε is the error amplitude that the user tolerates.

A major difficulty in solving this quadratic problem appears to be the prohibitive amount of memory required to store the matrix **K** when the number *N* of observations is large. Among others, decomposition methods [1][4][7] have been proposed to overcome this difficulty by iteratively solving smaller QP sub-problems.

Manuscript received April 13, 2007. This work was supported by PSA Peugeot Citroen Department of Numerical Engineering.

Monjed Ezzeddine is with PSA Peugeot Citroen, 18 Rue des Fauvelles, La Garenne Colombes, 92256 France (phone: 0033-1-56474788; fax: 0033-1-56473131; e-mail: monjed.ezzeddinne@mpsa.com).

Nicolas Lefebvre is with PSA Peugeot Citroen, Route de Gisy, 78140 Velizy-Villacoublay, France (e-mail: Nicolas.lefebvre@mpsa.com).

Régis Lengellé is with the University of Technology of Troyes, 12 Rue Marie Curie, 10000 Troyes, France (e-mail: Regis.lengelle@utt.fr).

III. SVM Mixture

In this section, we investigate the SVM mixture model proposed by Kwok [6], which is given by:

$$f(\mathbf{X}) = \sum_{k=1}^{P} \pi_k(\mathbf{X})\, \pi'_k(\mathbf{X})$$

where P is the number of experts, $\pi_k(\mathbf{X})$ is the output of expert k and is assumed to be previously determined, $\pi'_k(\mathbf{X})$ is the activation level function that has to be determined.

To be compatible with the SVM formalism, $\pi'_k(\mathbf{X})$ can be written $\pi'_k(\mathbf{X}) = <\boldsymbol{\omega}_k, \boldsymbol{\varphi}(\mathbf{X})> + \beta_k$, so the weighted output is given [6] by:

$$f(\mathbf{X}) = \sum_{k=1}^{P} \pi_k(\mathbf{X})\,(<\boldsymbol{\omega}_k, \boldsymbol{\varphi}(\mathbf{X})> + \beta_k) \qquad E_1$$

where $\boldsymbol{\omega}_k$, β_k are parameters to be determined during the training process. In the SVM mixture context [6], the criterion to be optimized is the sum of an error term $C\sum_{i=1}^{N}(\xi_i + \xi_i^*)$ and a penalty term $\frac{1}{2}\sum_{k=1}^{P}\|\boldsymbol{\omega}_k\|^2$. In the following, we remind the SVM mixture algorithm for regression problems.

As developed in [6], training of SVM mixture for regression problems leads to the following dual problem:

$$\left.\begin{array}{l} \min\limits_{\boldsymbol{\alpha},\boldsymbol{\alpha}^*} \frac{1}{2}\sum\limits_{i=1}^{N}\sum\limits_{j=1}^{N}\left(\alpha_i-\alpha_i^*\right)\left(\alpha_j-\alpha_j^*\right)H(\mathbf{X}_i,\mathbf{X}_j) \\ \quad +\varepsilon\sum\limits_{i=1}^{N}\left(\alpha_i+\alpha_i^*\right) + \sum\limits_{i=1}^{N}\left(\alpha_i-\alpha_i^*\right)y_i \\ \text{s.t.} \begin{cases} 0 \le \alpha_i, \alpha_i^* \le C, \quad i \in \{1,\ldots,N\} \\ \sum\limits_{i=1}^{N}\left(\alpha_i-\alpha_i^*\right)\pi_k(\mathbf{X}_i) = 0, k \in \{1,\ldots,P\} \end{cases} \end{array}\right\} DPR_2$$

where $H(\mathbf{X}_i,\mathbf{X}_j) = \sum_{k=1}^{P}\pi_k(\mathbf{X}_i)\pi_k(\mathbf{X}_j)K(\mathbf{X}_i,\mathbf{X}_j)$.

DPR_2 is a quadratic problem similar to DPR_1.

We have now to determine the β_k that no longer appear in DPR_2 but are variables in E_1, by exploiting KKT conditions. KKT conditions indicate that for SV verifying $0 < \alpha_i^{(*)} < C$, $(y_i - \mathrm{f}(\mathbf{X}_i)) = \pm\varepsilon$, the $\pm$ sign depending on $\alpha_i = 0$ or $\alpha^*_i = 0$. Introducing E_1 into this equation generally leads to a non invertible linear system from which the β_k can be obtained as a least squares solution [6]. This solution cannot be exact so KKT conditions cannot be verified.

Another drawback induced by the impossibility to verify KKT conditions is that decomposition methods cannot be used. Furthermore, most of efficient working set selection algorithms do not apply. For example Joachims' algorithm [4] does not apply because of the constraints expression in the SVM mixture QP problem that differs from the usual SVM QP problem. Other algorithms based on heuristics that randomly select observations violating KKT conditions are also disturbed by the approximate value of the bias. This is the case of Osuna's algorithm [8].

IV. SVM Mixture with Scalar Bias

The core of the SVM mixture method is the way it combines experts. We can modify this combination rule as follows:

$$f(\mathbf{X}) = \sum_{k=1}^{P} \pi_k(\mathbf{X})\, \pi'_k(\mathbf{X}) + b \qquad E_2$$

where $\pi_k(\mathbf{X})$ is an expert that is a regression function here, $\pi'_k(\mathbf{X})$ is the activation level function, now modified to become unbiased, and b is a global scalar bias. The combination rule translates to:

$$f(\mathbf{X}) = \left(\sum_{k=1}^{P} \pi_k(\mathbf{X})\ <\boldsymbol{\omega}_k, \boldsymbol{\varphi}(\mathbf{X})>\right) + b$$

Introducing $H(\mathbf{X}_i,\mathbf{X}_j) = \sum_{k=1}^{P}\pi_k(\mathbf{X}_i)\pi_k(\mathbf{X}_j)K(\mathbf{X}_i,\mathbf{X}_j)$, and after some analytical manipulations, the dual optimization problem becomes:

$$\left.\begin{array}{l} \min\limits_{\boldsymbol{\alpha},\boldsymbol{\alpha}^*} \frac{1}{2}\sum\limits_{i=1}^{N}\sum\limits_{j=1}^{N}\left(\alpha_i-\alpha_i^*\right)\left(\alpha_j-\alpha_j^*\right)H(\mathbf{X}_i,\mathbf{X}_j) \\ \quad +\varepsilon\sum\limits_{i=1}^{N}\left(\alpha_i+\alpha_i^*\right) + \sum\limits_{i=1}^{N}\left(\alpha_i-\alpha_i^*\right)y_i \\ \text{s.t.} \begin{cases} 0 \le \alpha_i, \alpha_i^* \le C, \ i \in \{1,\ldots,N\} \\ \sum\limits_{i=1}^{N}\left(\alpha_i-\alpha_i^*\right) = 0 \end{cases} \end{array}\right\} DPR_3$$

Now, DPR_3 has exactly the same expression as DPR_1. As usual, we can now determine the scalar bias b by exploiting observations corresponding to $0 < \alpha_i^{(*)} < C$ that must verify $(y_i - \mathrm{f}(\mathbf{X}_i)) = \pm\varepsilon$ (KKT conditions), as usual. Existence of an exact solution to the primal optimization problem is now shown and any large dataset training algorithm can be directly used.

V. Experiments

A. Input Space Partitioning

Because of the particularities of the real world application considered here (see §C), but without loss of generality, we are now concerned with mixtures where each model is trained only on a subset of the initial data. In addition, input space partition is performed in order to reduce the dimension of the input space E. The input vector $\mathbf{X}$ is partitioned into primary input variables $\mathbf{x}$ and secondary input variables $\boldsymbol{\theta}$:

$\mathbf{X} = (\mathbf{x}, \boldsymbol{\theta}) \in E \subset R^l \times R^m = R^d$. The training set $I = (\mathbf{x}_i, \boldsymbol{\theta}_i)$, $i = 1...n$ is divided into P subsets I_k, each corresponding to a fixed $\boldsymbol{\theta}_k \in R^m$, $I_k = \{(\mathbf{x}_i, \boldsymbol{\theta}_i); \boldsymbol{\theta}_i = \boldsymbol{\theta}_k\}$.

Input vector partitioning appears to be natural in many practical applications and industrial engineering problems. Consider, for example, a dynamical system where the input/output model depends on the time varying set point $\boldsymbol{\theta}$ and on the control parameters $\mathbf{x}$ (see §C).

In this case, the SVM mixture model is given by: $f(\mathbf{X}) = \sum_{k=1}^{P} \pi_k(\mathbf{x}) \pi'_k(\boldsymbol{\theta})$, where $\boldsymbol{\theta}$ represents the current set point vector, P denotes the number of local experts, $\pi_k(\mathbf{x})$ is the output (assumed to be known) of the k^{th} expert (i.e. corresponding to $\boldsymbol{\theta} = \boldsymbol{\theta}_k$) and $\pi'_k(\boldsymbol{\theta})$ is the activation level function. This is a particular case of the previous SVM mixture problem, where $\pi_k(\mathbf{X}) = \pi_k(\mathbf{x})$ and $\pi'_k(\mathbf{X}) = \pi'_k(\boldsymbol{\theta})$. Application of the SVM mixture method on this particular situation leads to the same problem as DPR_3 with only a slight modification in the kernel function. The QP problem becomes:

$$\left.\begin{aligned} \min_{\boldsymbol{\alpha}, \boldsymbol{\alpha}^*} & \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*) \sum_{k=1}^{P} \pi_k(\mathbf{x}_i) \pi_k(\mathbf{x}_j) K(\boldsymbol{\theta}_i, \boldsymbol{\theta}_j) \\ & + \varepsilon \sum_{i=1}^{N} (\alpha_i + \alpha_i^*) + \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) y_i \\ \text{s.t.} & \begin{cases} 0 \le \alpha_i, \alpha_i^* \le C, \quad i \in \{1, \ldots, N\} \\ \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) = 0 \end{cases} \end{aligned}\right\} DPR_4$$

This optimization problem DPR_4 translates to DPR_3 where $H(\mathbf{X}_i, \mathbf{X}_j) = \sum_{k=1}^{P} \pi_k(\mathbf{x}_i) \pi_k(\mathbf{x}_j) K(\boldsymbol{\theta}_i, \boldsymbol{\theta}_j)$, that remains a kernel [2]. We now apply this method to a simulated data regression problem.

B. Experiment on Simulated Data

The proposed SVM mixture method consists in determining the global regression function as a combination of the different experts $\pi_k(\mathbf{x})$ weighted by the activation level function $\pi'_k(\boldsymbol{\theta})$. The function to be reconstructed is depicted in figure 1. This function has been sampled on a bidimensional grid (Fig. 2). A global SVM regression was performed with optimized parameters; results are presented in Fig. 3. The training set, represented in Fig. 2, has been partitioned into subsets w.r.t. the variable $\boldsymbol{\theta}$ (Fig. 4), where an expert function has been estimated by SVM for each subset (Fig. 5). Fig. 6 shows the results of our SVM mixture method. The determination coefficient R_2 between the mixture output and the initial function equals 0.995 for our SVM mixture algorithm, to be compared with 0.808 for the global SVM regression depicted in Fig. 3.

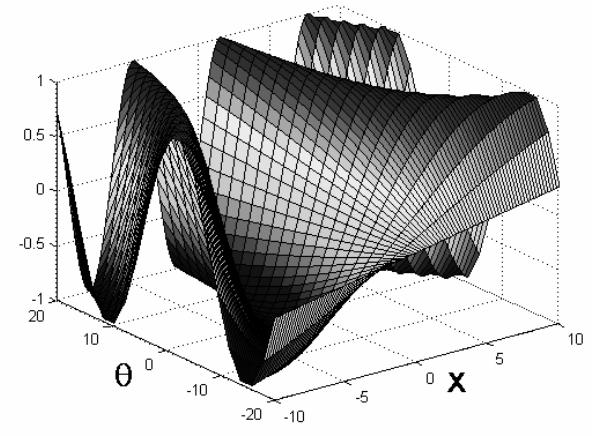

Fig. 1 Initial function

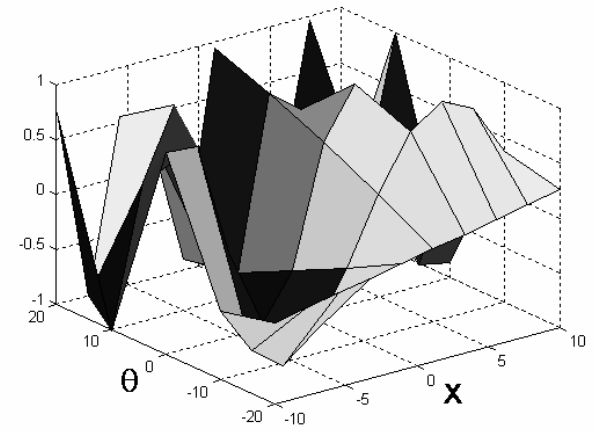

Fig. 2 Training Set

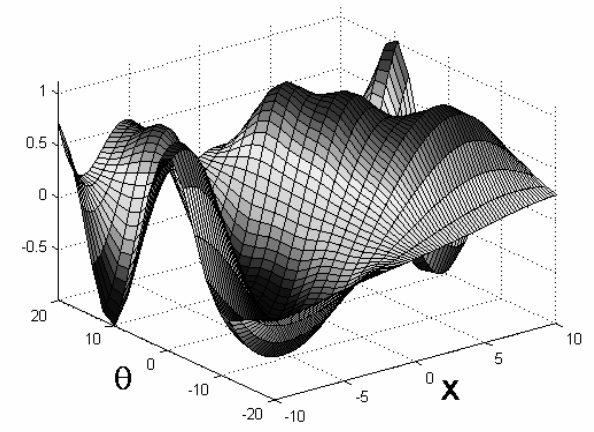

Fig. 3 Global SVM

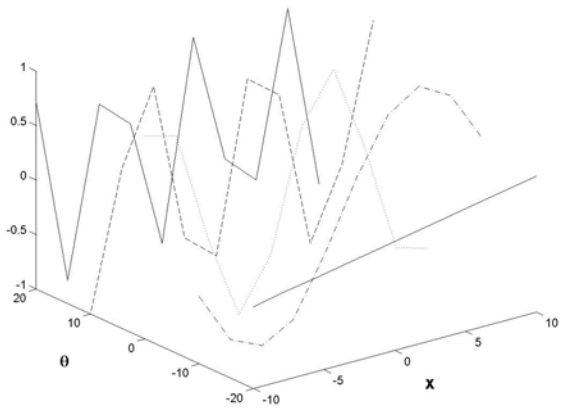

Fig. 4 Partitioned Training Set

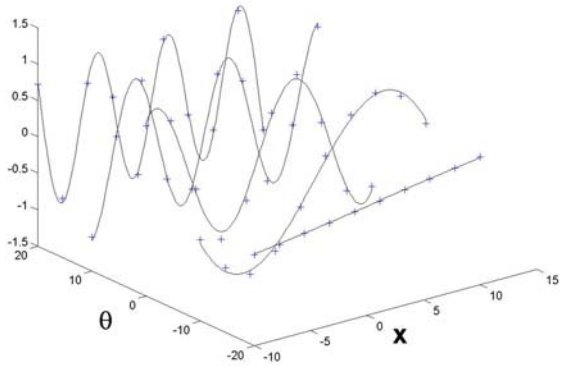

Fig. 5 Estimated Experts

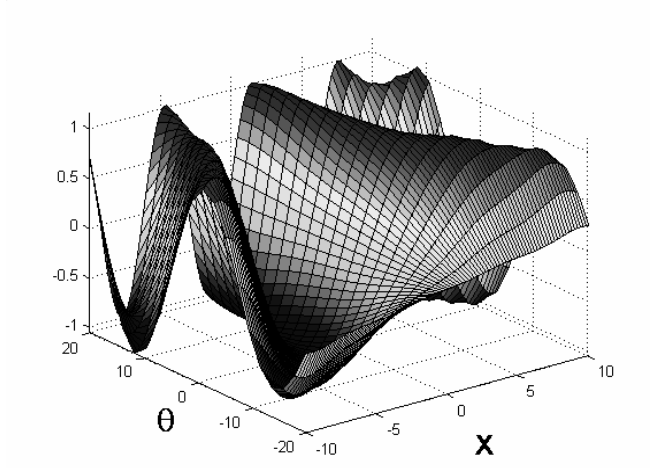

Fig. 6 SVM mixture

C. Engine Exhaust Pollutants Emission Modeling

In order to comply with the future EURO 5 emission standards, car manufacturers have to optimize the engine control parameters cartographies. While the driver selects torque and speed (that are considered as the time varying set point), engine control parameters (main and pilot injection quantities, injection timings, boost and rail pressures, …) have to be tuned in order to minimize a cost function of the emitted pollutants (e.g. NOx, CO, HC, particles…). The first step of this optimization process is to build up efficient models for estimating each considered pollutant quantity as a function of the control parameters for every location in the (torque, rpm) space. Because of the cost of experiments, modeling is generally achieved on a very few points in the (torque, rpm) space. Furthermore, the input output relationship is highly nonlinear. The *P* local models experimentally obtained correspond to our local experts. Interpolation to every other point of the (torque, rpm) space is a very challenging task.

We proposed to perform the mixture of the different local models using our algorithm. Here, the vector $\boldsymbol{\theta}$ denotes torque and rpm and the vector **x** denotes the vector of engine control parameters. The results obtained have shown that the SVM mixture algorithm outperforms classical interpolation methods usually used within the context of engine control. Fig. 7 depicts the sampling of the (torque, rpm) space where were trained the local models.

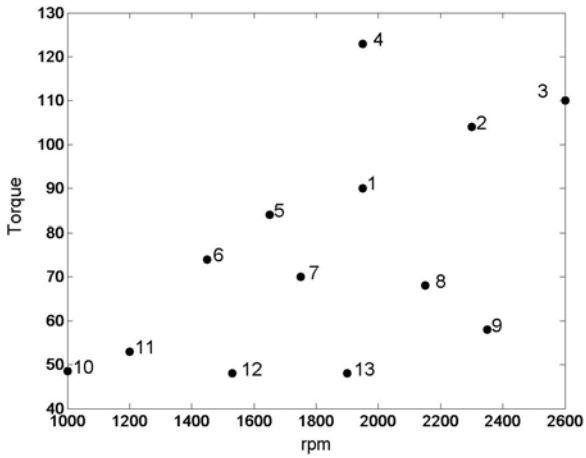

Fig. 7 (torque, rpm) Space sampling

Performances of our algorithm were evaluated by a leaving one out approach. Every setpoint in the (torque, rpm) space and all the corresponding control parameters were iteratively eliminated from the training data. The mixture model was optimized on the remaining observations and eliminated data were used as a test set. Figure 8 represents the mixture output as a function of the actual output for the pollutant NOx, when the setpoint #2 of the (torque, rpm) space was eliminated and used as a test set. As can be seen, both training and test data are very close to the first bisector, which indicates a very good prediction.

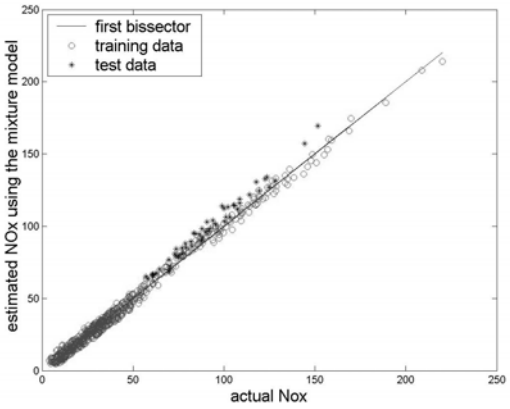

Fig. 8 Estimated NOx as a function of NOx

Global performance has been judged satisfactory by PSA Peugeot-Citroën.

VI. Conclusion

In this paper, we have presented an adaptation of the SVM mixture algorithm initially proposed by Kwok. Both for the regression and classification cases, the modification of the mixture model allows to translate the SVM mixture problem to a standard SVM training problem with only a slight modification of the kernel. The main results are:

- classical SVM training algorithms directly apply
- large data sets training algorithms can be used
- for decomposition training methods, the efficient Joachim's working set selection algorithm applies without any modification.

The proposed method has been tested on a simulated data regression example. The results obtained have shown that the SVM mixture outperforms a global SVM approach. This method has been successfully applied on a large scale real world application: engine exhausts pollutants emission modeling.

References

[1] Collobert R. and Bengio S., SVMTorch: "Support Vector Machine for Large-Scale Regression and Classification Problems", *Journal of Machine Learning Research*, 1: pp. 143-160, 2001.

[2] Genton M., "Classes of Kernels for Machine Learning: A statistics Perspective", *Journal of Machine Learning Research* 2: pp. 299-312, 2001.

[3] Jacobs R., Jordan M., Nowlan S., and Hinton G., "Adaptative Mixtures of Local Experts", *Neural Computation*, 3(1): pp. 79 -87, 1991.

[4] Joachims T., Making "Large-Scale SVM Learning Practical", *Advances in Kernel Methods – Support Vector Learning*, ch. 11, pp. 169-184 , MIT Press, 1999.

[5] Jordan M., Jacobs R., "Hierarchical Mixtures of Experts and the EM Algorithm", *Neural Computation*, 6(2): pp. 181-214, 1994.

[6] Kwok J., "Support Vector Mixture for Classification and Regression Problems", *Proceedings of the International Conference on Pattern Recognition*, pp. 255-258, 1998.

[7] Lin C. J., "On the Convergence of the Decomposition Method for Support Vector Machines", *IEEE Transactions on Neural Networks*, 12 (2001), pp. 1288-1298.

[8] Osuna E., Freund R. and Girosi F., "An Improved Training Algorithm for Support Vector Machines", *Proceedings of IEEE Workshop on Neural Networks for Signal Processing* pp. 276-285, 1997.

[9] *Vapnik V.,* The Nature of Statistical Learning Theory, *Springer, New York, 1995.*

[10] Vapnik V., Statistical Learning Theory, A Wiley-Interscience Publication, New York, 1998.