Parsimonious Support Vector Regression using Orthogonal Forward Selection with the Generalized Kernel Model

Xunxian Wang†, Sheng Chen‡, David Brown†

† Computer Intelligence & Applications Research Group
Department of Creative Technologies, University of Portsmouth
Buckingham Building, Lion Terrace, Portsmouth, PO1 3HE, UK
Email: xunxian.wang@port.ac.uk

‡ School of Electronics and Computer Science
University of Southampton
Highfield, Southampton SO17 1BJ, U.K.
E-mail: sqc@ecs.soton.ac.uk

Abstract

Sparse regression modeling is addressed using a generalized kernel model in which kernel regressor has its individually tuned position (center) vector and diagonal covariance matrix. An orthogonal least squares forward selection procedure is employed to append regressors one by one. After the determination of the model structure, namely the selection certain number of regressors, the model weight parameters are calculated from the Lagrange dual problem of the regression problem with the regularized linear $\varepsilon-$insensitive loss function. Different from the support vector regression, this stage of the procedure involves neither reproducing kernel Hilbert nor Mercer decomposition concepts and thus the difficulties associated with selecting a mapping from the input space to the feature space, needed in the support vector machine methods, can be avoided. Moreover, as the regressors used here are not restricted to be positioned at training input points and each regressor has its own diagonal covariance matrix, sparser representation can be obtained. Experimental results involving one toy example and two data sets demonstrate the effectiveness of the proposed regression modeling approach.

Keywords: Regression, support vector machine, orthogonal least squares forward selection, generalized kernel model, sparse modeling

1. Introduction

Having good generalization ability and sparse representation are two key requirements in establishing a learning machine. Forward selection using the orthogonal least squares (OLS) algorithm [1] is a simple and efficient construction method that is capable of producing parsimonious linear-in-the-weights nonlinear models with excellent generalization performance. Alternatively, the state-of-art sparse kernel modeling techniques, such as the relevant vector machine and support vector machine (SVM) [2,3], have been gaining popularity in data modeling applications especially SVM algorithm. Originated from maximum margin linear classification problem, one of the main features of the SVM is to use hyper-plane to do both classification and regression. In classification, the hyper-plane will be adjusted to obtain the maximum classification margin. In regression, the gradient of the hyper-plane will be kept as small as possible. In a SVM type method, the training data are mapped to a high dimensional space where they can be approximated by a hyper-plane. The parameter of the hyper-plane is obtained by minimizing the cost consisted of the linear $\varepsilon-$insensitive loss function and the squared gradient of the hyper-plane. The successfully application of the SVM is heavily depended on the finding of the mapping that is not easy to find unfortunately. And then reproducing kernel Hilbert space theory is used through Mercer theorem.

Unlike SVM formulation, the method proposed in this paper minimizes the cost consists of linear $\varepsilon-$insensitive loss function and the squared weight of the regressors. This minimization problem allows the usage of the non-Mercer kernels. Specifically, the generalized kernel function can be used in which each kernel regressor has its own tunable center vector and diagonal covariance matrix. The support vectors are selected by the OLS criteria and the number of the regressors can be determined by using some criteria such as cross validation, but not controlled by the ε value like in SVM. Unlike the

standard OLS algorithm [1], in which only the regressor selection procedure is used, here the regerssor parameters will be optimized as well. In fact, at each stage of the selection, the optimization is used with respect to the kernel center vector and diagonal covariance matrix, and the determination of these kernel parameters is performed using a repeated weighted boosting search algorithm [4]. After the selection of a parsimonious model representation, the kernel weights are then calculated from the Lagrange dual of the minimization problem. This proposed generalized kernel regression modelling approach has the potential of improving modelling capacity and producing sparser final models, compared with the standard SVM algorithm. The advantages of the proposed method are illustrated using one toy example.

2. Standard kernel regression modelling

The task of kernel regression modelling is to construct a kernel model from the given training data set $\{x_i, y_i\}_{i=1}^N$, where x_i is the ith training input vector of dimension m, y_i is the desired output with single dimension for the input x_i and N the number of training data. The SVM method solves the problem by using the following strategy.

2.1 Support vector machine regression problem

In dual space, SVM regression problem can be stated as below:

$$\max imize\ L = -\varepsilon\sum_{i=1}^{N}(\alpha_i^* + \alpha_i) + \sum_{i=1}^{N} y_i(\alpha_i^* - \alpha_i) - \frac{1}{2}\sum_{i,j=1}^{N}(\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)k(x_i, x_j) \tag{1}$$

$$\sum_{i=1}^{N}(\alpha_i^* - \alpha_i) = 0$$

$$subject\ \text{to}\ 0 \le \alpha_i^* \le C, i = 1, \cdots, N \tag{2}$$

$$0 \le \alpha_i \le C, i = 1, \cdots, N$$

Where $k_{i,j} = k(x_i, x_j) = \langle \varphi(x_i), \varphi(x_j) \rangle$, $\varphi(x)$ is the selected mapping from the input space to a high-dimensional (feature) space, $y = W^T\varphi(x) + b$ is the regression linear function (hyperplane) in the high dimensional space, W is the gradient of the hyperplane, C is the regularization parameter.
After obtaining $\alpha_i, \alpha_i^*, i = 1, \cdots, N$ and b, the regression model can be given by

$$\hat{y} = \sum_{i=1}^{N}(\alpha_i^* - \alpha_i)k(x_i, x) + b \tag{3}$$

One of the most common choices of kernel function is the Gaussian function of the form:

$$k(x_i, x) = \exp\left(-\frac{\|x - x_i\|^2}{2\sigma^2}\right) \tag{4}$$

The common kernel variance σ^2 is not provided by the algorithm and has to be determined by other means, such as via cross validation.

2.2 The dual of the minimization problem of linear $\varepsilon-$insensitive loss function with squared regressor weights

The proposed algorithm uses the system model of general OLS problem [1] defined by

$$\hat{y} = \sum_{i=1}^{M} w_i h_i(x) + b \tag{5}$$

where $h_i(x), i = 1, \cdots, M$ are the regression functions. $w_i, i = 1, \cdots, M$ are the regression weights. If define $W = [w_1 \quad w_2 \quad \cdots \quad w_M]^T$, the following minimization problem can be established

$$\text{Minimize}\ J(w, \xi^*, \xi) = \frac{1}{2}W^T W + C\left(\sum_{i=1}^{N}\xi_i^* + \sum_{i=1}^{N}\xi_i\right) \tag{6}$$

$$
\begin{aligned}
& y_i - \sum_{j=1}^{M} w_j h_j(x_i) - b \le \varepsilon + \xi_i^* \\
\text{Subject to } & \sum_{j=1}^{M} w_j h_j(x_i) + b - y_i \le \varepsilon + \xi_i \quad \text{for } 1 \le i \le N \\
& \xi_i^* \ge 0 \\
& \xi_i \ge 0
\end{aligned} \tag{7}
$$

Define $h(x) = [h_1(x) \quad h_2(x) \quad \cdots \quad h_M(x)]^T$, the dual problem of equations (6),(7) can be obtained as

Maximize

$$
D(\alpha, \alpha^*) = -\frac{1}{2}\sum_{i,j}^{N}(\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)h^T(x_i)h(x_j) - \varepsilon\sum_{i=1}^{N}(\alpha_i^* + \alpha_i) + \sum_{i=1}^{N}(\alpha_i^* - \alpha_i)y_i \tag{9}
$$

$$
\begin{aligned}
& \sum_{i=1}^{N}(\alpha_i^* - \alpha_i) = 0 \\
\text{Subject to } & 0 \le \alpha_i^* \le C, \quad i = 1, \cdots, N \\
& 0 \le \alpha_i \le C, \quad i = 1, \cdots, N
\end{aligned} \tag{10}
$$

After α_i, α_i^* are obtained, W can be calculated as

$$
W = \sum_{i=1}^{N}(\alpha_i^* - \alpha_i)h(x_i) \tag{11}
$$

2.3 Construction of sparse kernel models

Different from SVM, which can give a sparse system model, normally the value W obtained from equation (11) is not sparse. To obtain a sparse model, number M as well as the M kernel functions should be determined by some criteria before the equations (9-11) are solved. To obtain a sparse model, we proposed first to use the OLS algorithm [1] to select a parsimonious subset model from the full regression model with M items defined as $G = [g_1, g_2, \cdots, g_M]^T$, where $g_i = [h_i(x_1), h_i(x_2), \cdots, h_i(x_N)]^T$. In selecting the regressors, we will assume the bias term b=0 in the model and use the criteria in [1] which can be stated as

$$
J_M = Y^T Y - \sum_{j=1}^{M} \frac{(Y^T p_j)^2}{p_j^T p_j} \tag{12}
$$

Where $Y = [y_1, y_2, \cdots, y_N]^T$ and $p_j, j = 1, \cdots, M$ are the orthogonalized regressors [1]

Based on this error reduction criterion, a subset model can be obtained in a forward selection procedure [1]. At the lth selection stage, a model term is selected from the remaining candidates $p_j, l \le j \le M$ as the lth model term in the subset model, if it maximizes the error reduction criterion ER_j. The details of the selection algorithm are readily available in [1]-[5] and, therefore, will not be repeated here.

It should be stated that although two different cost functions are used in problem (9),(10) and the standard OLS problem, the usage of the OLS regressor selection is reasonable. Actually, the equation (29) can be rewritten as

$$
ER_j = (Y^T Y)\frac{(Y^T p_j)^2}{(p_j^T p_j)(Y^T Y)} \tag{13}
$$

With the same $(Y^T Y)$ for all the candidate regerssors $p_j, l \le j \le M$, the selection of the regressor is really based on the squared correlation between the training data and the regressor.

In the standard kernel regression modelling (both of SVM and OLS), each kernel regressor is positioned at a training input data point and a single common kernel variance σ^2 is used for every regerssors. Using the OLS forward selection procedure described above, we first obtain a sparse representation containing M_s kernel regressors. The corresponding kernel weights are then calculated using the ESVM method of section 2.2. We will referred to this approach of constructing sparse kernel models as the sparse extended SVM (SESVM) method.

3. Generalized Gaussian kernel regression model

In section 2.2, the deduction of the dual problem does not assume the concept of reproducing kernel Hilbert space and Mercer kernel. Therefore, we are not restricted to Mercer kernel. For example, we will allow a kernel function to take position other than the training input data points and to have an individually tunable diagonal covariance matrix. This leads the generalized kernel regression modelling. Specifically, we consider the regressors which take the forms of generalized Gaussian kernels:

$$g_j(x;\mu_j,\Sigma_j)=\exp\left(-\frac{1}{2}(x-\mu_j)^T\Sigma_j^{-1}(x-\mu_j)\right) \tag{14}$$

for $1\le j\le M$,where μ_j is the mean vector of the jth kernel and $\Sigma_j=diag\{\sigma_{j,1}^2,\sigma_{j,2}^2,\cdots,\sigma_{j,m}^2\}$ its diagonal covariance matrix.

In this section, we develop an incremental construction procedure for obtaining sparse generalized kernel models. We will adopt an orthogonal forward selection to append the kernels one by one. At the *lth* stage of model construction, the *lth* regressor is determined by maximizing the following error reduction criterion

$$ER_l(\mu_l,\Sigma_l)=\frac{(Y^T p_l)^2}{p_l^T p_l} \tag{15}$$

By using the method proposed in [4], a number of regressors with mean and covariance as their parameters can be obtained. After a certain number of kernels are selected, the dual problem will be solved to obtain the weight. We call this algorithm generalized sparse extended SVM (GSESVM).

4 Modeling examples

Two hundred points of training data $\{x, y\}$ were generated from the scaled sinc function corrupted by an observation noise shown below

$$y(x)=\frac{5\sin x}{x}+\varepsilon \tag{16}$$

where the equally spaced input $x\in[-10,10]$ and ε denotes the Gaussian white noise process with unit variance. Two hundred points of noise-free data were also generated as the test data set for possible model validation. For the Gaussian kernel modeling, the common kernel variance was set to $\sigma^2=1$. The parameter used in the repeated weighted boosting search algorithm for the generalized Gaussian kernel modeling were chosen to be $P_S=17$ and $M_R=20$.

Experimental comparison is used to show the advantage of the proposed algorithm over SVM. First, we randomly select $\varepsilon=0.5$ and obtain the modelling performance by using mean squares error (MSE) as the function of the regularization parameter C and the results are depicted in Fig.1. It can be seen that when C=0.6, the MSE of SVM for the testing set reaches its minimum and we use this C in the following experiments. In practice, however, the noise-free testing set was unavailable. But to show the advantage of the proposed method, we give some bias to SVM by using this C. When fixed C=0.6, and change the value of ε , the relationships between MSE and the ε can be obtained, which is shown in Fig.2. When $\varepsilon=0.6$, the test MSE of SVM reaches its minimum. Then $\varepsilon=0.6$ and C=0.6 are used and the system models for SVM and ESVM can be obtained, which are depicted in Fig.3 and Fig.4 alternatively. The MSE of SVM for the training set is 0.9405, for the noise-free test set is 0.0508; they are 0.8907 and 0.0556 for ESVM relatively. It should be pointed out that the system model given by ESVM is not sparse. For the two sparse methods, the selected first 16 support vectors for SESVM and GSESVM are listed in Table 1 and 2 separately and Fig.5 shows modelling performance as function of the selected subset model size. From Fig.5, it appears that when the numbers of the support vectors (SVs) equal to 7 and 13 for the standard Gaussian kernel model (SESVM), the training set MSE have a significantly reduction and after words there are long flat periods. For the general Gaussian kernel model (GSESVM), this item values are 5-terms and 13-terms alternatively. If we use SVs=7 and 13 for SESVM and SVs=5 and 13 for GSESVM to establish system models and evaluate the modelling performance as the function of ε , the results shown in Fig.6 can be obtained. To make comparison easy, 13-items models for both SESVM and GSESVM are constructed by set C=0.6. By changing ε value to obtain different MSE for SVM, SESVM and GSESVM, the obtained results can be summarised in Table 3. It is obvious that both the SESVM and the GSESVM methods can give more sparse models than SVM when similar MSE is required, especially in the situation when ε is small. The two sample sparse models with 13 terms constructed by the SESVM and GSESVM are shown in

Fig. 7 and 8, respectively. The MSE for SESVM are 0.9393 for the training set and 0.0319 for the noise-free test set, and 0.9325 and 0.0298 for GSESVM respectively. Because the support vectors of GSESVM does not belong to the training set, the weight of the relative regressors are used as the y value to depict the picture.

5 Conclusion

The contributions of this paper are threefold. Firstly, we have considered an alternative SVM formulation, referred to as the ESVM, which does not assume the reproducing kernel Hilbert space and can be applied to non-Mercer kernels. Secondly, a sparse kernel model construction algorithm, called the SESVM, has been proposed. In this approach a parsimonious representation is selected using the standard OLS forward selection procedure and the corresponding model weights are then computed using the ESVM formulation. Thirdly, which is a major contribution of our work, the generalized kernel modeling has been derived where each kernel regressor has its tunable center vector and diagonal covariance matrix. An orthogonal forward selection procedure has been proposed to incrementally construct a sparse generalized kernel model representation. At each model construction stage, a kernel regressor is optimized using a guided random search optimization algorithm. Again the corresponding model weights are then calculated using the ESVM formulation. Our modeling experimental results have clearly demonstrated the advantage of this proposed novel modeling technique to produce very sparse models that generalize well.

References

[1]. S. Chen, S.A. Billings and W. Luo, "Orthogonal least squares methods and their application to non-linear system identification," *Int. J. Control*, Vol.50, No.5, pp.1873–1896, 1989.

[2]. V. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer-Verlag, 1995.

[3]. B. Scholkopf, K.K. Sung, C.J.C. Burges, F. Girosi, P. Niyogi, T. Poggio and V. Vapnik, "Comparing support vector machines with Gaussian kernels to radial basis function classifiers," *IEEE Trans. Signal Processing*, Vol.45, No.11, pp.2758–2765, 1997.

[4]. S. Chen, X. Wang, D. Brown. Orthogonal least square with boosting for regression. Ideal04, 25-27, Aug. 2004, Exeter, UK. Lecture Notes in Computer Science 3177.

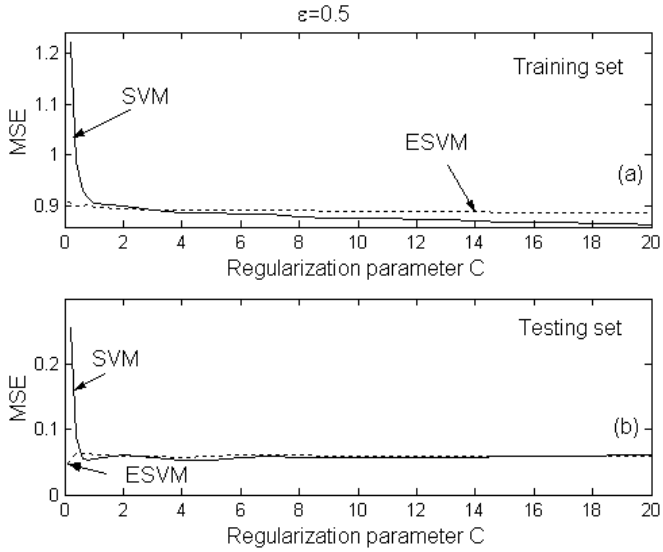

Figure 1: Influence of the regularization parameter C to the performance of the SVM and ESVM for the toy example: (a) over the noise training set, and (b) over the noise-free test set. The kernel variance $\sigma^2 = 1$, the error band parameter $\varepsilon = 0.5$.

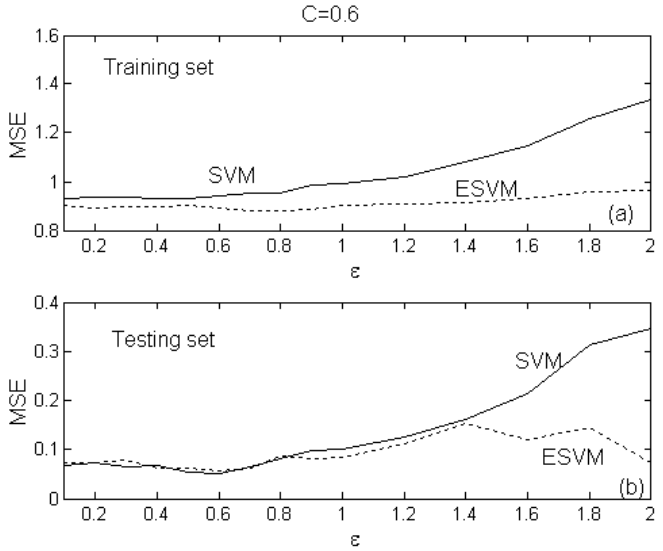

Figure 2: Influence of the error band parameter ε to the performance of the SVM and ESVM for the toy example: (a) over the noise training set, and (b) over the noise-free test set. The kernel variance $\sigma^2 = 1$ and the regularization parameter C=0.6.

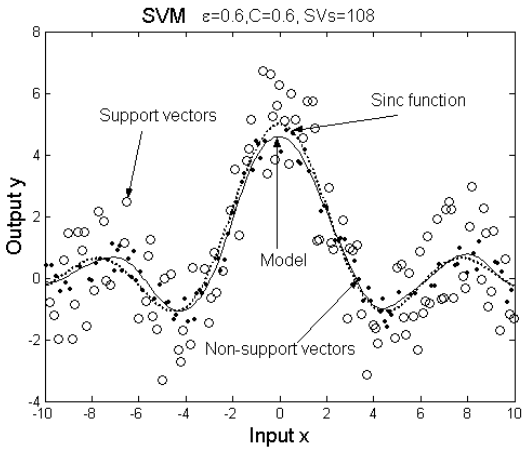

Figure 3: The experiment result of SVM for the toy example. Both the dots and circles are noisy training data. While the circles are support vectors, the dots are not. The dot curve denotes the sinc function and the solid curve indicates the kernel model. The regularization parameter C=0.6, the kernel variance $\sigma^2 = 1$ and $\varepsilon = 0.6$

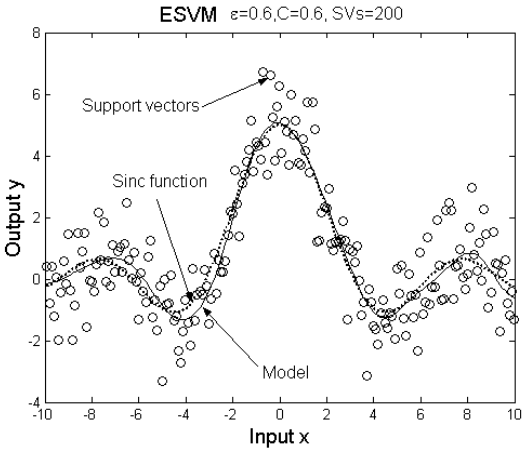

Figure 4: The experiment result of ESVM for the toy example. The circles are both training data and support vectors. The dot curve denotes the sinc function and the solid curve indicates the kernel model. The regularization parameter C=0.6, the kernel variance $\sigma^2 = 1$ and $\varepsilon = 0.6$

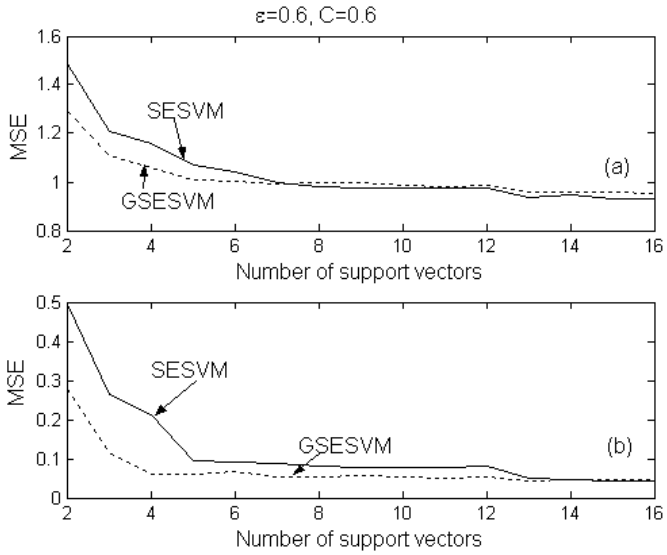

Figure 5: Modeling performance over the noisy training set (a) and the noise-free test set (b) as function of the selected model size. For the SESVM, standard Gaussian kernel model is used with $\sigma^2 = 1$ while for the GSESVM, generalized Gaussian kernel model with tunable means and variances is used. The error band parameter $\varepsilon = 0.6$, the regularization parameter C=0.6.

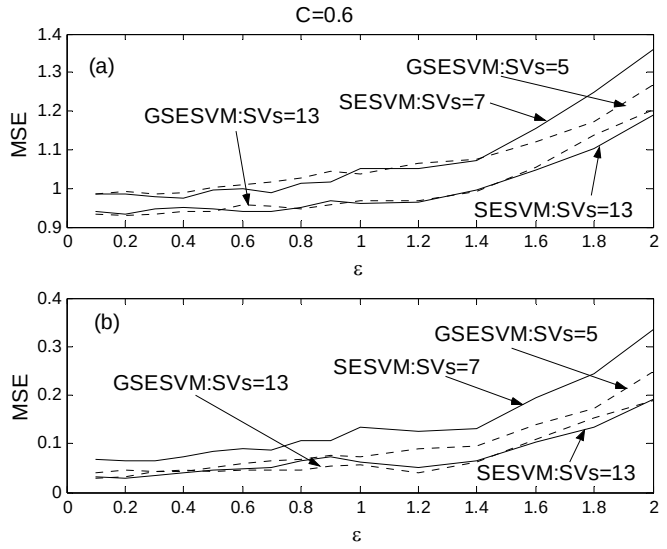

Figure 6: Influence of the error band parameter ε to the performance of the SESVM and GSESVM for the toy example: (a) over the noise training set, and (b) over the noise-free test set. The regularization parameter C=0.6. Standard Gaussian kernel variance $\sigma^2 = 1$.

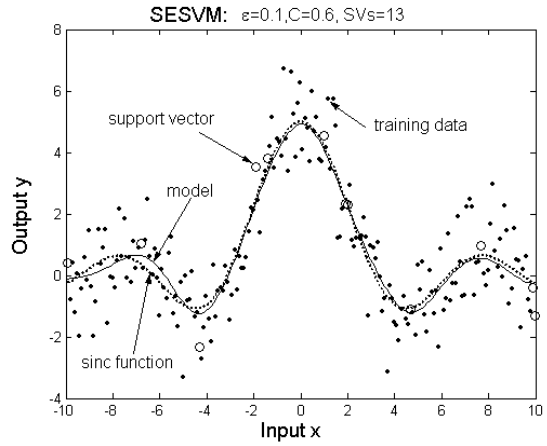

Figure 7: The experiment result of SESVM for the toy example. Both the dots and circles are noisy training data. While the circles are support vectors, the dots are not. The dot curve denotes the sinc function and the solid curve indicates the kernel model. The regularization parameter C=0.6, the kernel variance $\sigma^2 = 1$ and $\varepsilon = 0.1$

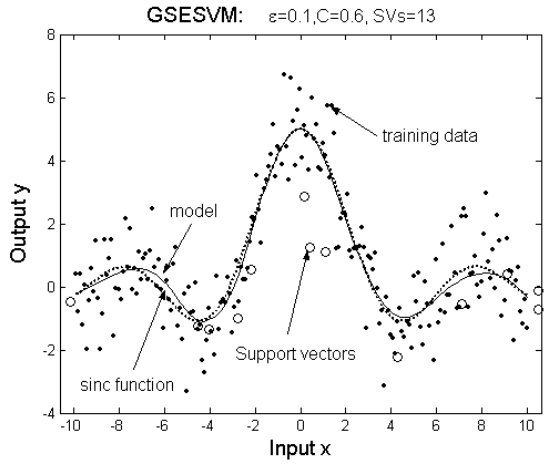

Figure 8: The experiment result of GSESVM for the toy example. The dots are noisy training data, the circles are added support vectors while the y value is the weight of this SV. The dot curve denotes the sinc function and the solid curve indicates the kernel model. The regularization parameter C=0.6, and $\varepsilon = 0.1$