# Chapter 2
# Generalized Eigenvalue Proximal Support Vector Machines

## 2.1 Introduction

In Chap. 1, we have presented a brief account of Proximal Support Vector Machines (PSVM) for the binary data classification problem. The PSVM formulation determines two parallel planes such that each plane is closest to one of the two data sets to be classified and the planes are as far apart as possible. The final separating hyperplane is then selected as the one which is midway between these two parallel hyperplanes. In a later development, Mangasarian and Wild [1] presented a new formulation where the two planes are no longer required to be parallel. Here, each plane is required to be as close as possible to one of the data sets and as far as possible from the other data set. They termed this new formulation as Generalized Eigenvalue Proximal Support Vector Machine (GEPSVM) because the required (non-parallel) planes are determined by solving a pair of generalized eigenvalue problems. We are discussing GEPSVM type models in this chapter because the basic GEPSVM model of Mangasarian and Wild [1] has been the major motivation for the introduction of Twin Support Vector Machines.

This chapter consists of three main sections, namely, GEPSVM for Classification: Mangasarian and Wild's Model, Variants of GEPSVM for Classification, and GEPSVR: Generalized Eigenvalue Proximal Support Vector Regression. Our presentation here is based on Mangasarian and Wild [1], Guarracino et al. [2], Shao et al. [3] and Khemchandani et al. [4].

## 2.2 GEPSVM for Classification

Let the training set for the binary data classification be given by

$$T_C = \{(x^{(i)}, y_i), i = 1, 2, ..., m\}, \tag{2.1}$$



Jayadeva et al., *Twin Support Vector Machines*, Studies in Computational Intelligence 659, DOI 10.1007/978-3-319-46186-1_2

where $x^{(i)} \in \mathbb{R}^n$ and $y_i \in \{-1, 1\}$. Let there be $m_1$ patterns having class label $+1$ and $m_2$ patterns having class label $-1$. We construct matrix A (respectively B) of order $(m_1 \times n)$ (respectively $(m_2 \times n)$) by taking the *i th* row of A (respectively B) as the *i th* pattern of class label $+1$ (respectively class label $-1$). Thus, $m_1 + m_2 = m$.

### 2.2.1 Linear GEPSVM Classifier

The linear GEPSVM classifier aims to determine two non parallel planes

$$x^T w_1 + b_1 = 0 \quad \text{and} \quad x^T w_2 + b_2 = 0, \tag{2.2}$$

such that the first plane is closest to the points of class $+1$ and farthest from the points of class $-1$, while the second plane is closest to the points in class $-1$ and farthest for the points in class $+1$. Here $w_1, w_2 \in \mathbb{R}^n$, and $b_1, b_2 \in \mathbb{R}$.

In order to determine the first plane $x^T w_1 + b_1 = 0$, we need to solve the following optimization problem

$$\underset{(w,b)\neq 0}{Min} \; \frac{\|Aw + eb\|^2 / \|(w, b)^T\|^2}{\|Bw + eb\|^2 / \|(w, b)^T\|^2}, \tag{2.3}$$

where $e$ is a vector of 'ones' of appropriate dimension and $\|.\|$ denotes the $L_2$-norm.

In (2.3), the numerator of the objective function is the sum of the squares of two-norm distances between each of the points of class $+1$ to the plane, and the denominator is the similar quantity between each of the points of class $-1$ to the plane. The way this objective function is constructed, it meets the stated goal of determining a plane which is closest to the points of class $+1$ and farthest to the points of class $-1$. Here, it may be remarked that ideally we would like to minimize the numerator and maximize the denominator, but as the same $(w, b)$ may not do the simultaneous optimization of both the numerator and denominator, we take the ratio and minimize the same. This seems to be a very natural motivation for introducing the optimization problem (2.3).

The problem (2.3) can be re-written as

$$\underset{(w,b)\neq 0}{\text{Min}} \; \frac{\|Aw + eb\|^2}{\|Bw + eb\|^2}, \tag{2.4}$$

where it is assumed that $(w, b) \neq 0$ implies that $(Bw + eb) \neq 0$. This makes the problem (2.4) well defined. Now, let

$$G = [A \quad e]^T [A \quad e],$$
$$H = [B \quad e]^T [B \quad e],$$

and $z = (w, b)^T$. Then (2.4) becomes

$$\underset{z \neq 0}{\text{Min}} \ \frac{z^T G z}{z^T H z} \ . \tag{2.5}$$

The objective function in (2.5) is the famous Rayleigh quotient of the generalized eigenvalue problem $Gz = \lambda Hz$, $z \neq 0$. When $H$ is positive definite, the Rayleigh quotient is bounded and its range is $[\lambda_{min}, \lambda_{max}]$ where $\lambda_{min}$ and $\lambda_{max}$ respectively denote the smallest and the largest eigenvalues. Here, $G$ and $H$ are symmetric matrices of order $(n+1) \times (n+1)$, and $H$ is positive definite under the assumption that columns of $[B \quad e]$ are linearly independent.

Now following similar arguments, we need to solve the following optimization problem to determine the second plane $x^T w_2 + b_2 = 0$

$$\underset{(w,b) \neq 0}{Min} \ \frac{\|Bw + eb\|^2}{\|Aw + eb\|^2}, \tag{2.6}$$

where we need to assume that $(w, b) \neq 0$ implies that $(Aw + eb) \neq 0$. We can again write (2.6) as

$$\underset{z \neq 0}{Min} \ \frac{z^T H z}{z^T G z} \ , \tag{2.7}$$

where $G$ can be assumed to be positive definite provided columns of $[A \quad e]$ are linearly independent.

But in general, there is always a possibility that the null spaces of $G$ and $H$ have a nontrivial intersection. In that case the generalized eigenvalue problems (2.5) and (2.7) become singular and hence require a regularization technique to render their solutions. In this context, we may refer to Parlett [5] where the symmetric eigenvalue problems are discussed in greater detail.

Mangasarian and Wild [1] proposed the introduction of a Tikhonov type regularization (Tikhonov and Arsen [6]) in problems (2.4) and (2.6) to get

$$\underset{(w,b) \neq 0}{\text{Min}} \ \frac{\|Aw + eb\|^2 + \delta \|(w, b)^T\|^2}{\|Bw + eb\|^2}, \tag{2.8}$$

and

$$\underset{(w,b) \neq 0}{\text{Min}} \ \frac{\|Bw + eb\|^2 + \delta \|(w, b)^T\|^2}{\|Aw + eb\|^2}, \tag{2.9}$$

respectively. Here $\delta > 0$. Further a solution of (2.8) gives the first plane $x^T w_1 + b_1 = 0$ while that of (2.9) gives the second plane $x^T w_2 + b_2 = 0$.

Let us now introduce the following notations

$$\begin{aligned} P &= [A \quad e]^T [A \quad e] + \delta I, \\ Q &= [B \quad e]^T [B \quad e], \\ R &= [B \quad e]^T [B \quad e] + \delta I, \\ S &= [A \quad e]^T [A \quad e], \\ z^T &= (w, b), \end{aligned} \tag{2.10}$$

and rewrite problems (2.8) and (2.9) as

$$\underset{z \neq 0}{Min} \ \frac{z^T P z}{z^T Q z}, \tag{2.11}$$

and

$$\underset{z \neq 0}{Min} \ \frac{z^T R z}{z^T S z}, \tag{2.12}$$

respectively.

We now have the following theorem of Parlett [5].

**Theorem 2.2.1** (Parlett [5])

*Let $E$ and $F$ be arbitrary $(n \times n)$ real symmetric matrices and $F$ be positive definite. Let for $u \neq 0$, $\gamma(u) = ((u^T Eu)/(u^T Fu))$ be the Rayleigh quotient and $Eu = \theta Fu$ be the corresponding generalized eigenvalue problem. Then*

1. *$\gamma(u)$ ranges over $[\theta_{min}, \theta_{max}]$ where $\theta_{min}$ and $\theta max$ are the smallest and the largest eigenvalues of the generalized eigenvalue problem $Eu = \theta Fu$.*
2. *If $\overline{u}$ is a stationary point of $\gamma(u)$, then $\overline{u}$ is an eigenvector of the corresponding generalized eigenvalue problem $Eu = \theta Fu$ and conversely.*

In view of Theorem 2.2.1, solving the optimization problem (2.11) amounts to finding the eigenvector corresponding to the smallest eigenvalue of the generalized eigenvalue problem

$$Pz = \lambda Qz, \quad z \neq 0. \tag{2.13}$$

In a similar manner, to solve the optimization problem (2.12) we need to get the eigenvector corresponding to the smallest eigenvalue of the generalized eigenvalue problem

$$Rz = \mu Sz, \quad z \neq 0. \tag{2.14}$$

We can summarize the above discussion in the form of the following theorem.

**Theorem 2.2.2** (Mangasarian and Wild [1])

*Let $A$ and $B$ be $(m_1 \times n)$ and $(m_2 \times n)$ matrices corresponding to data sets of class $+1$ and class $-1$ respectively. Let $P$, $Q$, $R$ and $S$ be defined as at (2.10). Let columns of $(A \quad e)$ and $(B \quad e)$ be linearly independent. Let $z_1$ be the eigenvector corresponding to smallest eigenvalue of the generalized eigenvalue problem (2.13). Also, let $z_2$ be the eigenvector corresponding to the smallest eigenvector of the generalized eigenvalue problem (2.14). Then, $z_1 = col(w_1, b_1)$ and $z_2 = col(w_2, b_2)$.*

*Remark 2.2.1* $z_1$ and $z_2$ determine the planes $x^T w_1 + b_1 = 0$ and $x^T w_2 + b_2 = 0$. Further, the generalized eigenvalue problems (2.13) and (2.14) can be solved very easily by employing two MATLAB commands: $eig(P, Q)$ and $eig(R, S)$ respectively.

*Remark 2.2.2* A new point $x \in \mathbb{R}^n$ is assigned to the class $i$, $(i = 1, 2)$ depending on which of the two planes this point is closest to, i.e. class $= \underset{i=1,2}{arg\ min}\{|x^T w_i + b_i|/\|w_i\|^2\}$.

*Remark 2.2.3* The requirement that columns of $(A \quad e)$ and $(B \quad e)$ are linearly independent, is only a sufficient condition for the determination of $z_1$ and $z_2$. It is not a necessary condition as may be verified for the XOR example. Also this linear independence condition may not be too restrictive if $m_1$ and $m_2$ are much larger than $n$.

We now discuss a couple of examples for illustration purposes.

*Example 2.2.1* (*XOR Problem*)

The training set for the XOR problem is given by

$$T_C = \{((0, 0), +1), ((1, 1), +1), ((1, 0), -1), ((0, 1), -1).\}$$

Therefore

$$A = \begin{bmatrix} 0 & 0 \\ 1 & 1 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Hence

$$P = S = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 2 \end{bmatrix},$$

and

$$Q = R = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 2 \end{bmatrix}.$$

Then the generalized eigenvalue problem (2.13) has the solution $\lambda_{min} = 0$ and $z_1 = (-1 \quad 1 \quad 0)$. Therefore, the first plane is given by $-x_1 + x_2 + 0 = 0$, i.e., $x_1 - x_2 = 0$. In the same way, the generalized eigenvalue problem (2.14) has the solution $\mu_{min} = 0$ and $z_2 = (-1 \quad -1 \quad 1)$, which gives the second plane $-x_1 - x_2 + 1 = 0$, i.e., $x_1 + x_2 = 1$.

Here we observe that neither the columns of $(A \quad e)$ nor that of $(B \quad e)$ are linearly independent but the problems (2.13) and (2.14) have solutions $z_1$ and $z_2$ respectively.

The XOR example also illustrates that proximal separability does not imply linear separability. In fact these are two different concepts and therefore the converse is also not true. It is also possible that two sets be both proximally and linearly separable, e.g. the AND problem.

*Example 2.2.2* (*Cross Planes Example : Mangasarian and Wild* [1])

Mangasarian and Wild [1] constructed the famous Cross Planes example, where the data consists of points that are close to one of the two intersecting cross planes in $\mathbb{R}^2$. The below given Fig. 2.1 depicts the Cross Planes data sets in $\mathbb{R}^2$. Here it can be observed that for this data set, the training set correctness of GEPSVM is 100 %, while it is only 80 % for the linear PSVM classifier.

Geometrically, the Cross Planes data set is a perturbed generalization of the XOR example. Similar to XOR, it serves as a test example for the efficacy of typical linear classifiers, be it linear SVM, linear PSVM, linear GEPSVM or others to be studied in the sequel. An obvious reason for the poor performance of linear PSVM on Cross Planes data set is that in PSVM we have insisted on the requirement that the proximal planes should be parallel, which is not the case with linear GEPSVM.

We next discuss the kernel version of GEPSVM to get the nonlinear GEPSVM classifier.

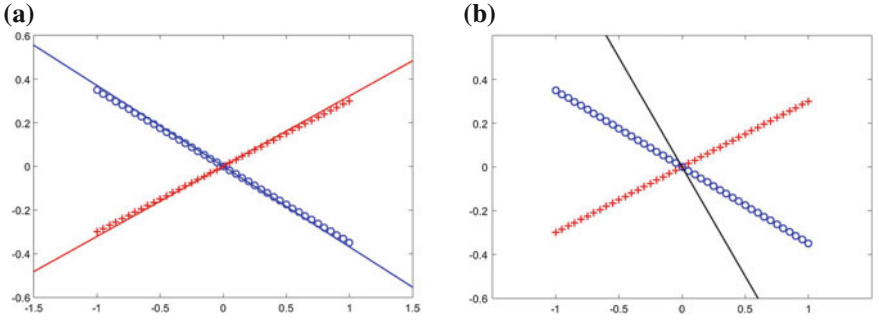


**Fig. 2.1** Cross planes dataset: **a** GEPSVM classifier **b** SVM classifier

### 2.2.2 Nonlinear GEPSVM Classifier

Let $C^T = [A \quad B]^T$ and $K$ be an appropriate chosen kernel. We wish to construct the following two kernel generated surfaces

$$K(x^T, C^T)u_1 + b_1 = 0, \quad \text{and} \quad K(x^T, C^T)u_2 + b_2 = 0. \tag{2.15}$$

Here we note that (2.15) are nonlinear surfaces rather than planes, but serve the same purpose as (2.2). Thus, the first (respectively second) surface is closest to data points in class $+1$ (respectively class $-1$) and farthest from the data points in class $-1$ (respectively class $+1$).

If we take the linear kernel $K(x^T, C^T) = x^T C$ and define $w_1 = C^T u_1, w_2 = C^T u_2$, then (2.15) reduces to (2.2). Therefore to generate surfaces (2.15) we can generalize our earlier arguments and get the following two optimization problems

$$\underset{(u,b)\neq 0}{Min} \frac{\|K(A, C^T)u + eb\|^2 + \delta\|(u, b)^T\|^2}{\|K(B, C^T)u + eb\|^2}, \tag{2.16}$$

and

$$\underset{(u,b)\neq 0}{Min} \frac{\|K(B, C^T)u + eb\|^2 + \delta\|(u, b)^T\|^2}{\|K(A, C^T)u + eb\|^2}. \tag{2.17}$$

If we now define

$$\begin{aligned} P_1 &= [K(A, C^T) \quad e]^T[K(A, C^T) \quad e] + \delta I, \\ Q_1 &= [K(B, C^T) \quad e]^T[K(B, C^T) \quad e], \\ R_1 &= [K(B, C^T) \quad e]^T[K(B, C^T) \quad e] + \delta I, \\ S_1 &= [K(A, C^T) \quad e]^T[K(A, C^T) \quad e], \\ z^T &= (u, b), \end{aligned} \tag{2.18}$$

then the optimization problem (2.16) and (2.17) reduces to generalized eigenvalue problems

$$\underset{z\neq 0}{Min} \frac{z^T P_1 z}{z^T Q_1 z}, \tag{2.19}$$

and

$$\underset{z\neq 0}{Min} \frac{z^T R_1 z}{z^T S_1 z}, \tag{2.20}$$

respectively.

Now we can state a theorem similar to Theorem (2.2.2) and make use of the same to generate the required surfaces (2.15). Specifically, let $z_1$ (respectively $z_2$) be the eigenvector corresponding to the smallest eigenvalue of the generalized eigenvalue problem (2.19) and (respectively 2.20), the $z_1 = (u_1, b_1)$ (respectively $z_2 = (u_2, b_2)$) gives the desired surface $K(x^T, C^T)u_1 + b_1 = 0$ (respectively $K(x^T, C^T)u_2 + b_2 = 0$.). For a new point $x \in \mathbb{R}^n$, let

$$\text{distance}(x, S_i) = \frac{|K(x^T, C^T)u_i + b_i|}{\sqrt{(u_i)^T K(C, C^T)u_i}},$$

where $S_i$ is the non linear surface $K(x^T, C^T)u_i + b_i = 0$ $(i = 1, 2)$. Then the class $i$ $(i = 1, 2)$ for this new point $x$ is assigned as per the following rule

$$\text{class} = \underset{i=1,2}{arg\,Min} \frac{|K(x^T, C^T)u_i + b_i|}{\sqrt{(u_i)^T K(C, C^T)u_i}}. \tag{2.21}$$

Mangasarian and Wild [1] implemented their linear and nonlinear GEPSVM classifiers extensively on artificial as well as real world data sets. On Cross Planes data set in $(300 \times 7)$ dimension (i.e. $m = 300, n = 7$) the linear classifier gives 10-fold testing correctness of 98 % where as linear PSVM and linear SVM give the correctness of 55.3 % and 45.7 % respectively. Also on the Galaxy Bright data set, the GEPSVM linear classifier does significantly better than PSVM. Thus, Cross Planes as well as Galaxy Bright data sets indicate that allowing the proximal planes to be non-parallel allows the classifier to better represent the data set when needed. A similar experience is also reported with the nonlinear GEPSVM classifier on various other data sets.

## 2.3 Some Variants of GEPSVM for Classification

There are two main variants of original GEPSVM. These are due to Guarracino et al. [2] and Shao et al. [3]. Guarracino et al. [2] proposed a new regularization technique which results in solving a single generalized eigenvalue problem. This formulation is termed as Regularized GEPSVM and is denoted as ReGEPSVM. The formulation of Shao et al. [3] is based on the difference measure, rather than the ratio measure of GEPSVM, and therefore results in solving two eigenvalue problems, unlike the two generalized eigenvalue problems in GEPSVM. Shao et al. [3] model is termed as Improved GEPSVM and is denoted as IGEPSVM. Both of these variants are attractive as they seem to be superior to the classical GEPSVM in terms of classification accuracy as well as in computation time. This has been reported in [2] and [3] by implementing and experimenting these algorithms on several artificial and benchmark data sets.

### 2.3.1 ReGEPSVM Formulation

Let us refer to the problems (2.8) and (2.9) where Tikhonov type regularization term is introduced in the basic formulation of (2.4) and (2.6). Let us also recall the following theorem from Saad [7] in this regard.

**Theorem 2.3.1** *Let $\tau_1$, $\tau_2$, $\delta_1$ and $\delta_2$ be scalars such that $(\tau_1\tau_2 - \delta_1\delta_2) \neq 0$. Let $G^* = (\tau_1 G - \delta_1 H)$ and $H^* = (\tau_2 H - \delta_2 G)$. Then the generalized eigenvalue problems $G^*x = \lambda H^*x$ and $Gx = \mu Hx$ are related as follows*

1. *if $\overline{\lambda}$ is an eigenvalue of $G^*x = \lambda H^*x$ and $\overline{\mu}$ is an eigenvalue of $Gx = \mu Hx$, then*

$$\overline{\mu} = \frac{\tau_2\overline{\lambda} + \delta_1}{\tau_1 + \delta_2\overline{\lambda}},$$

2. *both generalized eigenvalue problems have the same eigenvectors.*

We now consider the following optimization problem

$$\underset{(w,b)\neq 0}{Min} \quad \frac{\|Aw + eb\|^2 + \hat{\delta_1}\|Bw + eb\|^2}{\|Bw + eb\|^2 + \hat{\delta_2}\|Aw + eb\|^2}, \tag{2.22}$$

which is similar to (2.8) (and also to (2.9)) but with a different regularization term. Now if we take $\tau_1 = \tau_2 = 1$, $\hat{\delta_1} = -\delta_1 > 0$, $\hat{\delta_2} = -\delta_2 > 0$ with the condition that $\tau_1\tau_2 - \delta_1\delta_2 \neq 0$, then Theorem (2.3.1) becomes applicable for the generalized eigenvalue problem corresponding to the optimization problem (2.22). Let

$$\begin{aligned} U &= [A \quad e]^T[A \quad e] + \hat{\delta_1}[B \quad e]^T[B \quad e], \\ V &= [B \quad e]^T[B \quad e] + \hat{\delta_2}[A \quad e]^T[A \quad e]. \end{aligned}$$

Then, the generalized eigenvalue problem corresponding to (2.22) is

$$Uz = \lambda Vz, \qquad z \neq 0. \tag{2.23}$$

Further, the smallest eigenvalue of the original problem (2.5) becomes the largest eigenvalue of (2.23), and the largest eigenvalue of (2.5) becomes the smallest eigenvalue of (2.23). This is because of Theorem (2.3.1) which asserts that the spectrum of the transformed eigenvalue problem $G^*x = \lambda H^*x$ gets shifted and inverted. Also, the eigenvalues of (2.5) and (2.7) are reciprocal to each other with the same eigenvectors. Therefore, to determine the smallest eigenvalues of (2.5) and (2.7) respectively, we need to determine the largest and the smallest eigenvalues of (2.23). Let $z_1 = col(w_1, b_1)$ and $z_2 = col(w_2, b_2)$ be the corresponding eigenvectors. Then, the respective planes are $x^T w_1 + b_1 = 0$ and $x^T w_2 + b_2 = 0$.

For the nonlinear case, we need to solve the analogous optimization problem

$$\underset{(u,b)\neq 0}{Min} \frac{\|K(A, C^T)u + eb\|^2 + \hat{\delta}_1 \|K(B, C^T)u + eb\|^2}{\|K(B, C^T)u + eb\|^2 + \hat{\delta}_2 \|K(A, C^T)u + eb\|^2}, \tag{2.24}$$

via the related generalized eigenvalue problem $G^*x = \lambda H^*x$. Its solution will give two proximal surfaces

$$K(x^T, C^T)u_1 + b_1 = 0 \quad \text{and} \quad K(x^T, C^T)u_2 + b_2 = 0,$$

and the classification for a new point $x \in \mathbb{R}^n$ is done similar to nonlinear GEPSVM classifier.

### 2.3.2 *Improved GEPSVM Formulation*

It has been remarked in Sect. 2.1 that the determination of the first plane $x^T w_1 + b_1 = 0$ (respectively the second plane $x^T w_2 + b_2 = 0$) requires the minimization (respectively maximization) of $(\|Aw + eb\|^2/\|(w, b)^T\|^2)$ and the maximization (respectively minimization) of $(\|Bw + eb\|^2/\|(w, b)^T\|^2)$. Since the same $(w, b)$ may not perform this simultaneous optimization, Mangasarian and Wild [1] proposed a ratio measure to construct the optimization problem (2.4) (respectively (2.6)) and thereby obtained the generalized eigenvalue problem (2.13) (respectively (2.14)) for determining the first (respectively second) plane.

The main difference between the works of Mangasarian and Wild [1], and that of Shao et al. [3] is that the former considers a ratio measure whereas the latter considers a difference measure to formulate their respective optimization problems. Conceptually there is a bi-objective optimization problem which requires the minimization of $(\|Aw + eb\|^2/\|(w, b)^T\|^2)$ and maximization of $(\|Bw + eb\|^2/\|(w, b)^T\|^2)$, i.e. minimization of -$(\|Bw + eb\|^2/\|(w, b)^T\|^2)$. Thus the relevant optimization problem is the following bi-objective optimization problem

$$\underset{(w,b)\neq 0}{Min} \left( \frac{\|Aw + eb\|^2}{\|(w, b)^T\|^2}, -\frac{\|Bw + eb\|^2}{\|(w, b)^T\|^2} \right). \tag{2.25}$$

Let $\gamma > 0$ be the weighting factor which determines the trade-off between the two objectives in (2.25). Then, the bi-objective optimization problem (2.25) is equivalent to the following scalar optimization problem

$$\underset{(w,b)\neq 0}{Min} \left( \frac{\|Aw + eb\|^2}{\|(w, b)^T\|^2} - \gamma \frac{\|Bw + eb\|^2}{\|(w, b)^T\|^2} \right). \tag{2.26}$$

Let $z = (w, b)^T$, $G = [A \quad e]^T[A \quad e]$ and $H = [B \quad e]^T[B \quad e]$. Then, the optimization problem (2.26) becomes

$$\underset{z\neq 0}{Min}\ \frac{z^T(G-\gamma H)z}{z^T I z}. \tag{2.27}$$

Now similar to GEPSVM, we can also introduce a Tikhonov type regularization term (2.27) to get

$$\underset{z\neq 0}{Min}\ \frac{z^T(G-\gamma H)z+\delta\|z\|^2}{z^T I z},$$

i.e.

$$\underset{z\neq 0}{Min}\ \frac{z^T(G+\delta I-\gamma H)z}{z^T I z}. \tag{2.28}$$

The above problem (2.28) is exactly the minimization of the Rayleigh quotient whose global optimum solution can be obtained by solving the following eigenvalue problem

$$(G+\delta I-\gamma H)z=\lambda z, \qquad z\neq 0. \tag{2.29}$$

By determining the eigenvector corresponding to the smallest eigenvalue of (2.29) we get $z_1=(w_1,b_1)^T$ and hence the desired first plane $x^T w_1+b_1=0$.

In a similar manner, the second plane $x^T w_2+b_2=0$ is obtained by determining the eigenvector $z_2=(w_2,b_2)^T$ corresponding to the smallest eigenvalue for the eigenvalue problem

$$(H+\delta I-\gamma G)z=\mu z, \qquad z\neq 0. \tag{2.30}$$

In conclusion, the two desired planes $x^T w_i+b_i=0$, $(i=1,2)$ can be obtained by solving two eigenvalue problems (2.29) and (2.30). Then as before, a new point $x\in\mathbb{R}^n$ is assigned to the class $i$ $(i=1,2)$, depending on which of the two hyperplanes it is closer to, i.e.,

$$\text{class(x)}=\underset{i=1,2}{arg\,Min}\frac{\|x^T w_i+b_i\|}{\|w_i\|}.$$

The above results can easily be extended to the nonlinear case by considering the eigenvalue problems

$$(M+\delta I-\gamma N)z=\lambda z, \qquad z\neq 0, \tag{2.31}$$

and

$$(N+\delta I-\gamma M)z=\mu z, \qquad z\neq 0, \tag{2.32}$$

where

$$\begin{aligned} M &= [K(A, C^T) \quad e]^T [K(A, C^T) \quad e], \\ N &= [K(B, C^T) \quad e]^T [K(B, C^T) \quad e], \\ C^T &= [A \quad B]^T, \end{aligned}$$

and I is an identity matrix of appropriate dimension.

Let $z_1 = (u_1, b_1)^T$ (respectively $z_2 = (u_2, b_2)^T$) be the eigenvector corresponding to the smallest eigenvalue of (2.31) (respectively (2.32)) then the desired surfaces are $K(x^T, C^T)u_1 + b_1 = 0$ and $K(x^T, C^T)u_2 + b_2 = 0$ respectively. For a new point $x \in \mathbb{R}^n$ we again decide class label based on the following decision rule

$$\text{class(x)} = \underset{i=1,2}{arg\,Min} \frac{|K(x^T, C^T)u_i + b_i|}{\sqrt{(u_i)^T K(C, C^T)u_i}}. \tag{2.33}$$

Shao et al. [3] termed their model as Improved Generalized Eigenvalue Proximal Support Vector Machine (IGEPSVM). It is known that the linear IGEPSVM needs only to solve two eigenvalue problems with computational time complexity of $\mathcal{O}(n^2)$, where $n$ is the dimensionality of data points. In contrast, the linear GEPSVM requires two generalized eigenvalue problems whose complexity is $\mathcal{O}(n^3)$. For the nonlinear case, the computational complexity is $\mathcal{O}(m^2)$ for Shao et al. model [3] and is $\mathcal{O}(m^3)$ for Mangasarian and Wild model. Here, $m$ is the number of training data points. This probably explains that in numerical implementations, IGEPSVM takes much less time than GEPSVM. For details of these numerical experiments we shall refer to Shao et al. [3]. Recently, Saigal and Khemchandani [8] presented comparison of various nonparallel algorithms including variations of GEPSVM along with TWSVM for multicategory classification.

## 2.4 GEPSVR: Generalized Eigenvalue Proximal Support Vector Regression

The aim of this section is to present the regression problem in the setting of the generalized eigenvalue problem. Here, we present two models for the regression problem. The first is termed as the GEPSVR which is in the spirit of GEPSVM and requires the solution of two generalized eigenvalue problems. The second formulation is in the spirit of ReGEPSVM and is termed as the Regularized Generalized Eigenvalue Support Vector Regressor (ReGEPSVR). The formulation of ReGEPSVR requires the solution of a single regularized eigenvalue problem and it reduces the execution time to half as compared to GEPSVR.

Earlier Bi and Bennett [9] made a very significant theoretical contribution to the theory of support vector regression. They (Bi and Bennett [9]) showed that the problem of support vector regression can be regarded as a classification problem in

the dual space and maximizing the margin corresponds to shrinking of effective $\epsilon$-tube. From application point of view, this result is of utmost importance because this allows to look for other classification algorithms and study their regression analogues. We have already demonstrated this duality aspect in the context of SVR in Chap. 1, where it was derived via SVM approach. We shall continue to take this approach in this section as well as in some of the later chapters where regression problem is further studied.

### 2.4.1 GEPSVR Formulation

For GEPSVR, our objective is to find the two non-parallel $\epsilon$-insensitive bounding regressors. The two non-parallel regressors around the data points are derived by solving two generalized eigenvalue problems. We first discuss the case of linear GEPSVR, the case of nonlinear GEPSVR may be developed analogously.

Let the training set for the regression problem be given by

$$T_R = \{(x^{(i)}, y_i), i = 1, 2, ..., m\},$$

for some $\epsilon > 0$ where $x^{(i)} \in \mathbb{R}^n$ and $y_i \in \mathbb{R}$.

Following Bi and Bennett [9], we consider the associated classification problem with the training set $T_C$ in $\mathbb{R}^{n+1}$ given by

$$T_C = \{((x^{(i)}, y_i + \epsilon), +1), ((x^{(i)}, y_i - \epsilon), -1), i = 1, 2, ..., m\}, \text{ where } \epsilon > 0.$$

The GEPSVR algorithm finds two functions $f_1(x) = x^T w_1 + b_1$ and $f_2(x) = x^T w_2 + b_2$ such that each one determines the $\epsilon$-insensitive bounded regressor.

Let $A$ be an $(m \times n)$ matrix whose $i\,th$ row is the vector $(x^{(i)})^T$. Let $(Y + \epsilon e) = (y_1 + \epsilon, ..., y_m + \epsilon)$ and $(Y - \epsilon e) = (y_1 - \epsilon, ..., y_m - \epsilon)$. Then $f_1(x) = x^T w_1 + b_1$ may be identified as a hyperplane in $\mathbb{R}^{n+1}$. The GEPSVR formulation treats the given regression problem as a classification problem in $\mathbb{R}^{n+1}$ with the training data set as $T_C$. Thus, $f_1(x)$ is determined so as to minimize the Euclidean distance of the hyperplane $f_1(x) = x^T w_1 + b_1$ in $\mathbb{R}^{n+1}$ from the set of points $(A, Y - \epsilon e)$ and maximize its Euclidean distance from the set of points $(A, Y + \epsilon e)$. Likewise $f_2(x)$ is determined so as to minimize the Euclidean distance of the hyperplane $f_2(x) = x^T w_2 + b_2$ in $\mathbb{R}^{n+1}$ from the set of points $(A, Y + \epsilon e)$ and maximize its distance from $(A, Y + \epsilon e)$.

This leads to the following optimization problem for determining $f_1(x)$, the optimization problem for determining $f_2(x)$ can be written on similar lines.

$$\underset{(w,b)\neq 0}{Min} \; \frac{\|Aw + eb - (Y - e\epsilon)\|^2 / \|(w, b)^T\|^2}{\|Aw + eb - (Y + e\epsilon)\|^2 / \|(w, b)^T\|^2}. \tag{2.34}$$

Here, it is implicitly assumed that $(w, b) \neq 0$ implies that $(Aw + eb - (Y + e\epsilon)) \neq 0$. In that case (2.34) can be simplified as

$$\underset{(w,b)\neq 0}{Min} \frac{\|Aw + eb - (Y - e\epsilon)\|^2}{\|Aw + eb - (Y + e\epsilon)\|^2}. \tag{2.35}$$

Now the optimization problem (2.35) can be regularized as

$$\underset{(w,b)\neq 0}{Min} \frac{\|Aw + eb - (Y - e\epsilon)\|^2 + \delta\|(w \quad b \quad -1)^T\|^2}{\|Aw + eb - (Y + e\epsilon)\|^2}, \tag{2.36}$$

where $\delta > 0$ is the regularization coefficient. Let $\mu = (w \quad b \quad -1)^T$ and

$$R = [A \quad e \quad (Y - e\epsilon)]^T[A \quad e \quad (Y - e\epsilon)] + \delta I$$
$$S = [A \quad e \quad (Y + e\epsilon)]^T[A \quad e \quad (Y + e\epsilon)].$$

Then, the solution of regularized optimization problem (2.36) can be obtained by solving the following generalized eigenvalue problem

$$Ru = \eta Su, \qquad u \neq 0. \tag{2.37}$$

Let $u_1$ denote the eigenvector corresponding to the smallest eigenvalue $\eta_{min}$ of (2.37). To obtain $w_1$ and $b_1$ from $u_1$, we normalize $u_1$ by the negative of the $(n+2)^{th}$ element of $u_1$ so as to force a $(-1)$ at the $(n+2)^{th}$ position of $u_1$. Let this normalized representation of $u_1$ be $u_1$ with $(-1)$ at the end, such that $u_1^{new} = [w_1 \quad b_1 \quad -1]^T$. Then, $w_1$ and $b_1$ determine an $\epsilon$-insensitive bounding regressor as $f_1(x) = x^T w_1 + b_1$.

Similarly for determining the second bounding regressor $f_2(x)$ we consider the regularized optimization problem

$$\underset{(w,b)\neq 0}{Min} \frac{\|Aw + eb - (Y + e\epsilon)\|^2 + \delta\|(w \quad b \quad -1)^T\|^2}{\|Aw + eb - (Y - e\epsilon)\|^2}, \tag{2.38}$$

where $(w, b) \neq 0$ implies $(Aw + eb - (Y + e\epsilon)) \neq 0$. Let $u = (w \quad b \quad -1)^T$ and

$$P = [A \quad e \quad (Y + e\epsilon)]^T[A \quad e \quad (Y + e\epsilon)] + \delta I,$$
$$Q = [A \quad e \quad (Y - e\epsilon)]^T[A \quad e \quad (Y - e\epsilon)].$$

Then, the optimization problem (2.38) is equivalent to the generalized eigenvalue problem

$$Pu = \nu Qu, \qquad u \neq 0. \tag{2.39}$$

Now as before, finding minimum eigenvalue $\nu_{min}$ of (2.37) and having determined the corresponding eigenvector $u_2$, we obtain $w_2$ and $b_2$ by the normalizing procedure explained earlier. This gives the other $\epsilon$-insensitive regressor $f_2(x) = x^T w_2 + b_2$.

Having determined $f_1(x)$ and $f_2(x)$ from $u_1$ and $u_2$, the final regressor $f(x)$ is constructed by taking the average, i.e.,

$$f(x) = \frac{1}{2}(f_1(x) + f_2(x)) = \frac{1}{2}x^T(w_1 + w_2) + \frac{1}{2}(b_1 + b_2).$$

### 2.4.2 Regularized GEPSVR Formulation

Working on similar lines as that of ReGEPSVM, we intend to use regularization technique of Guarracino et al. [2] for finding a single regularized eigenvalue problem whose smallest and largest eigenvalue $\mu_{min}$ and $\mu_{max}$ would provide $(w_1, b_1)^T$ and $(w_2, b_2)^T$ from their corresponding eigenvectors. These solutions would then correspond to the two $\epsilon$-insensitive bounding regressors $f_1(x) = x^T w_1 + b_1$ and $f_2(x) = x^T w_2 + b_2$ respectively. Let $\hat{\delta}_1 > 0$ and $\hat{\delta}_2 > 0$. We consider the following regularized optimization problem

$$\underset{(w,b)\neq 0}{Min} \frac{\|Aw + eb - (Y - e\epsilon)\|^2 + \hat{\delta}_1\|Aw + eb - (Y + e\epsilon)\|^2}{\|Aw + eb - (Y + e\epsilon)\|^2 + \hat{\delta}_2\|Aw + eb - (Y - e\epsilon)\|^2}. \tag{2.40}$$

Let $t = [w \quad b \quad -1]^T$ and $U$ and $V$ be defined as

$$U = [A \quad e \quad (Y - e\epsilon)]^T[A \quad e \quad (Y - e\epsilon)] + \hat{\delta}_1[A \quad e \quad (Y + e\epsilon)]^T[A \quad e \quad (Y + e\epsilon)],$$
$$V = [A \quad e \quad (Y + e\epsilon)]^T[A \quad e \quad (Y + e\epsilon)] + \hat{\delta}_2[A \quad e \quad (Y - e\epsilon)]^T[A \quad e \quad (Y - e\epsilon)].$$

Now using the earlier discussed properties of Rayleigh quotient, the optimization problem (2.40) is equivalent to the following generalized eigenvalue problem

$$Ut = \nu Vt, \qquad t \neq 0. \tag{2.41}$$

This yields the eigenvector $t_1$ corresponding to largest eigenvalue $\nu_{max}$ of (2.41), and eigenvector $t_2$ corresponding to the smallest eigenvalue $\nu_{min}$ of (2.41). To obtain $w_1$ and $b_1$ from $t_1$ and $w_2$ and $b_2$ from $t_2$, we follow the usual normalization procedure of Sect. 2.4.1 and get $t_1^{new} = (w_1 \quad b_1 \quad -1)^T$ and $t_2^{new} = (w_2 \quad b_2 \quad -1)^T$. This yield the $\epsilon$-insensitive bounding regressors $f_1(x) = x^T w_1 + b_1$ and $f_2(x) = x^T w_2 + b_2$ from $t_1^{new}$ and $t_2^{new}$. For a new point $x \in \mathbb{R}^n$, the regressed value $f(x)$ is given by

$$f(x) = \frac{1}{2}(f_1(x) + f_2(x)) = \frac{1}{2}x^T(w_1 + w_2) + \frac{1}{2}(b_1 + b_2).$$

For extending our results to the nonlinear case, we consider the following kernel generated functions instead of linear functions

$$F_1(x) = K(x^T, A^T)w_1^\phi + b_1^\phi \quad \text{and} \quad F_2(x) = K(x^T, A^T)w_2^\phi + b_2^\phi, \tag{2.42}$$

where $K$ is the chosen Kernel function and $w_1^\phi$, $w_2^\phi$, $b_1^\phi$ and $b_2^\phi$ are defined in the kernel spaces.
Let $t^\phi = [w^\phi \quad b^\phi \quad -1]^T$ and

$$\begin{aligned} E &= [K(A, A^T) \quad e \quad (Y - e\epsilon)]^T[K(A, A^T) \quad e \quad (Y - e\epsilon)] + \\ &\quad \hat{\delta_1}[K(A, A^T) \quad e \quad (Y + e\epsilon)]^T[K(A, A^T) \quad e \quad (Y + e\epsilon)], \\ F &= [K(A, A^T) \quad e \quad (Y + e\epsilon)]^T[K(A, A^T) \quad e \quad (Y + e\epsilon)] + \\ &\quad \hat{\delta_2}[K(A, A^T) \quad e \quad (Y - e\epsilon)]^T[K(A, A^T) \quad e \quad (Y - e\epsilon)]. \end{aligned}$$

We have the following generalized eigenvalue problem

$$Et^\phi = \beta Ft^\phi, \qquad t^\phi \neq 0. \tag{2.43}$$

This yields the eigenvector $t_1^\phi$ corresponding to the largest eigenvalue $\beta_{max}$ of (2.43), and $t_2^\phi$ corresponding to the smallest eigenvalue $\beta_{min}$ of (2.43). We do the usual normalization of $t_1^\phi$ and $t_2^\phi$ to get $t_{new}^\phi = [w_1^\phi \quad b_1^\phi \quad -1]^T$, and $t_{new}^\phi = [w_2^\phi \quad b_2^\phi \quad -1]^T$. This gives the $\epsilon$-insensitive bounding regressor $F_1(x) = K(x^T, A^T)w_1^\phi + b_1^\phi$ and $F_2(x) = K(x^T, A^T)w_2^\phi + b_2^\phi$. For a new input pattern $x \in \mathbb{R}^n$, the regressed value is given by

$$F(x) = \frac{1}{2}(w_1^\phi + w_2^\phi)K(x^T, A^T) + \frac{1}{2}(b_1^\phi + b_2^\phi).$$

To test the performance of ReGEPSVR, Khemchandani et al. [4] implemented their model on several data sets including UCI, financial time series data and four two dimensional functions considered by Lázaro et al. [10]. Here, we summarize some of the conclusions reported in Khemchandani et al. [4].

### 2.4.3 Experimental Results

To test the performance of the proposed ReGEPSVR, Khemchandani et al. [4] compared it with SVR on several datasets. The performance of these regression algorithms on the above-mentioned datasets largely depends on the choice of initial parameters. Hence, optimal parameters for these algorithms for UCI datasets [11] are selected by using a cross-validation set [12] comprising of 10 percent of the dataset, picked up randomly. Further, RBF kernel is taken as the choice for kernel function in all

the implementations, adding to another parameter, i.e. $\sigma_{kernel}$ to be tuned. Hence, for SVR, the parameters to be tuned are $C$ and $\sigma_{kernel}$ respectively. Similarly, the parameters to be tuned for ReGEPSVR are $\hat{\delta_1}$, $\hat{\delta_2}$ and $\sigma_{kernel}$ respectively. Further, normalization is applied on the input features of UCI datasets.

For comparing results over UCI datasets, we consider the standard tenfold cross-validation methodology. This yields 10 results corresponding to each fold, of which the mean and variance values are computed and compared. The algorithms on all these datasets are evaluated on the basis of their:

- NMSE: (Normalized Mean Square Error)
  Without loss of generality, let $m$ be the number of testing samples, $y_i$ be the real output value of sample $x_i$, $\hat{y_i}$ be the predicted value of sample $x_i$, and $\bar{y} = \frac{1}{m}\sum_i y_i$ be the average value of $y_1, ....y_m$. Then NMSE is defined as

$$NMSE = \frac{(m-1)}{(m)} \times \frac{\sum_{i=1}^{m}(y_i - \hat{y_i})^2}{\sum_{i=1}^{m}(y_i - \bar{y})^2}. \tag{2.44}$$

  A small NMSE value means good agreement between estimations and real-values.

We consider four benchmark datasets, including the Boston Housing, Machine CPU, Servo, and Auto-price datasets, obtained from the UCI repository.

The Boston Housing dataset consists of 506 samples. Each sample has thirteen features which designate the quantities that influence the price of a house in Boston suburb and an output feature which is the house-price in thousands of dollars. The Machine CPU dataset concerns Relative CPU performance data. It consists of 209 cases, with seven continuous features, which are MYCT, MMIN, MMAX, CACH, CHMIN, CHMAX, PRP (output). The Servo dataset consists of 167 samples and covers an extremely nonlinear phenomenon-predicting the rise time of a servo mechanism in terms of two continuous gain settings and two discrete choices of mechanical linkages. The Auto price dataset consists of 159 samples with fifteen features.

Results for comparison between ReGEPSVR and SVR for the four UCI datasets are given in Table 2.1.

**Table 2.1** UCI datasets: NMSE comparisons

| Dataset | NMSE | NMSE |
|---|---|---|
| | TSVR | ReGEPSVR |
| Boston Housing | $0.154 \pm 0.053$ | $0.140 \pm 0.002$ |
| Machine CPU | $0.253 \pm 0.029$ | $0.198 \pm 0.023$ |
| Servo | $0.185 \pm 0.125$ | $0.180 \pm 0.027$ |
| Auto Price | $0.510 \pm 0.142$ | $0.223 \pm 0.022$ |

## 2.5 Conclusions

This chapter presents the GEPSVM formulation of Mangasarian and Wild [1] for binary data classification and discusses its advantages over the traditional SVM formulations. We also discuss two variants of the basic GEPSVM formulation which seem to reduce overall computational effort over that of GEPSVM. These are ReGEPSVM formulation of Guarracino et al. [2], and Improved GEPSVM formulation of Shao et al. [3]. In ReGEPSVM we get only a single generalized eigenvalue problem, while Improved GEPSVM deals with two simple eigenvalue problems. Taking motivation from Bi and Bennett [9], a regression analogue of GEPSVM is also discussed. This regression formulation is due to Khemchandani et al. [4] and is termed as Generalized Eigenvalue Proximal Support Vector Regression (GEPSVR). A natural variant of GEPSVR, namely ReGEPSVR is also presented here.

## References

1. Mangasarian, O. L., & Wild, E. W. (2006). Multisurface proximal support vector machine classification via generalized eigenvalues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *28*(1), 69–74.
2. Guarracino, M. R., Cifarelli, C., Seref, O., & Pardalos, P. M. (2007). A classification method based on generalized eigenvalue problems. *Optimization Methods and Software*, *22*(1), 73–81.
3. Shao, Y.-H., Deng, N.-Y., Chen, W.-J., & Wang, Z. (2013). Improved generalized eigenvalue proximal support vector machine. *IEEE Signal Processing Letters*, *20*(3), 213–216.
4. Khemchandani, R., Karpatne, A., & Chandra, S. (2011). Generalized eigenvalue proximal support vector regressor. *Expert Systems with Applications*, *38*, 13136–13142.
5. Parlett, B. N. (1998). *The symmetric eigenvalue problem: Classics in applied mathematics* (Vol. 20). Philadelphia: SIAM.
6. Tikhonov, A. N., & Arsenin, V. Y. (1977). *Solutions of Ill-posed problems*. New York: Wiley.
7. Saad, Y. (1992). *Numerical methods for large eigenvalue problems*. New York: Halsted Press.
8. Saigal, P., & Khemchandani, R. (2015) Nonparallel hyperplane classifiers for multi-category classification. In *IEEE Workshop on Computational Intelligence: Theories, Applications and Future Directions (WCI)*. Indian Institude of Technology, Kanpur.
9. Bi, J., & Bennett, K. P. (2003). A geometric approach to support vector regression. *Neurocomputing*, *55*, 79–108.
10. Lázarao, M., Santamaŕia, I., Péreze-Cruz, F., & Artés-Rodŕiguez, A. (2005). Support vector regression for the simultaneous learning of a multivariate function and its derivative. *Neurocomputing*, *69*, 42–61.
11. Alpaydin, E., & Kaynak, C. (1998). UCI Machine Learning Repository, Irvine, CA: University of California, Department of Information and Computer Sciences. http://archive.ics.uci.edu/ml.
12. Duda, R., Hart, P., & Stork, D. (2001). *Pattern classification*. New York: Wiley.

http://www.springer.com/978-3-319-46184-7

Twin Support Vector Machines
Models, Extensions and Applications
Jayadeva; Khemchandani, R.; Chandra, S.
2017, XIV, 211 p. 21 illus., 20 illus. in color., Hardcover
ISBN: 978-3-319-46184-7