Improving the Performance of the Support Vector Machine: Two Geometrical Scaling Methods

P. Williams, S. Wu, and J. Feng

Department of Informatics, University of Sussex, Falmer, Brighton BN1 9QH, UK

Abstract. In this chapter, we discuss two possible ways of improving the performance of the SVM, using geometric methods. The first adapts the kernel by magnifying the Riemannian metric in the neighborhood of the boundary, thereby increasing separation between the classes. The second method is concerned with optimal location of the separating boundary, given that the distributions of data on either side may have different scales.

Key words: kernel transformation, adaptive kernel, geometric scaling, generalization error, non-symmetric SVM

1 Introduction

The support vector machine (SVM) is a general method for pattern classification and regression proposed by Vapnik and co-authors [10]. It consists of two essential ideas, namely:

- to use a kernel function to map the original input data into a high-dimensional space so that two classes of data become linearly separable;
- to set the discrimination hyperplane in the middle of two classes.

Theoretical and experimental studies have proved that SVM methods can outperform conventional statistical approaches in term of minimizing the generalization error (see e.g. [3, 8]). In this chapter we review two geometrical scaling methods which attempt to improve the performance of the SVM further. These two methods concern two different ideas of scaling the SVM in order to reduce the generalization error.

The first approach concerns the scaling of the kernel function. From the geometrical point of view, the kernel mapping induces a Riemannian metric in the original input space [1, 2, 9]. Hence a good kernel should be one that can enlarge the separation between the two classes. To implement this idea, Amari

P. Williams, S. Wu, and J. Feng: *Improving the Performance of the Support Vector Machine: Two Geometrical Scaling Methods*, StudFuzz **177**, 205–218 (2005)
www.springerlink.com

and Wu [1, 9] propose a strategy which optimizes the kernel in a two-step procedure. In the first step of training, a primary kernel is used, whose training result provides information about where the separating boundary is roughly located. In the second step, the primary kernel is conformally scaled to magnify the Riemannian metric around the boundary, and hence the separation between the classes. In the original method proposed in [1, 9], the kernel is enlarged at the positions of the support vectors, which takes into account the fact that support vectors are in the vicinity of the boundary. This method, however, is susceptible to the distribution of data points. In the present study, we propose a different way for scaling kernel that directly acts on the distance to the boundary. Simulation shows that the new method works robustly.

The second approach to be reviewed concerns the optimal position for the discriminating hyperplane. The standard form of SVM chooses the separating boundary to be in the middle of two classes (more exactly, in the middle of the support vectors). By using extremal value theory in statistics, Feng and Williams [5] calculate the exact value of the generalization error in a one-dimensional separable case, and find that the optimal position is not necessarily to be at the mid-point, but instead it depends on the scales of the distances of the two classes of data with respect to the separating boundary. They further suggest how to use this knowledge to rescale SVM in order to achieve better generalization performance.

2 Scaling the Kernel Function

The SVM solution to a binary classification problem is given by a discriminant function of the form

$$f(\mathbf{x}) = \sum_{s \in SV} \alpha_s y_s K(\mathbf{x}_s, \mathbf{x}) + b \tag{1}$$

A new out-of-sample case is classified according to the sign of $f(\mathbf{x})$.[1]

The support vectors are, by definition, those $\mathbf{x}_i$ for which $\alpha_i > 0$. For separable problems each support vector $\mathbf{x}_s$ satisfies

$$f(\mathbf{x}_s) = y_s = \pm 1 \, .$$

In general, when the problem is not separable or is judged too costly to separate, a solution can always be found by bounding the multipliers α_i by the condition $\alpha_i \leq C$, for some (usually large) positive constant C. There are then two classes of support vector which satisfy the following distinguishing conditions:

$$\begin{aligned} &\text{I}: \quad y_s f(\mathbf{x}_s) = 1 \, , \quad 0 < \alpha_s < C \, ; \\ &\text{II}: \quad y_s f(\mathbf{x}_s) < 1 \, , \quad \alpha_s = C \, . \end{aligned}$$

[1] The significance of including or excluding a constant b term is discussed in [7].

Support vectors in the first class lie on the appropriate separating margin. Those in the second class lie on the wrong side (though they may be correctly classified in the sense that $\text{sign} f(\mathbf{x}_s) = y_s$). We shall call support vectors in the first class *true* support vectors and the others, by contrast, *bound*.

2.1 Kernel Geometry

It has been observed that the kernel $K(\mathbf{x}, \mathbf{x}')$ induces a Riemannian metric in the input space S [1, 9]. The metric tensor induced by K at $\mathbf{x} \in S$ is

$$g_{ij}(\mathbf{x}) = \frac{\partial}{\partial x_i} \frac{\partial}{\partial x'_j} K(\mathbf{x}, \mathbf{x}') \Big|_{\mathbf{x}'=\mathbf{x}}. \tag{2}$$

This arises by considering K to correspond to the inner product

$$K(\mathbf{x}, \mathbf{x}') = \boldsymbol{\phi}(\mathbf{x}) \cdot \boldsymbol{\phi}(\mathbf{x}') \tag{3}$$

in some higher dimensional feature space H, where $\boldsymbol{\phi}$ is a mapping of S into H (for further details see [4, p. 35]). The inner product metric in H then induces the Riemannian metric (2) in S via the mapping $\boldsymbol{\phi}$.

The volume element in S with respect to this metric is given by

$$dV = \sqrt{g(\mathbf{x})}\, dx_1 \cdots dx_n \tag{4}$$

where $g(\mathbf{x})$ is the determinant of the matrix whose (i, j)th element is $g_{ij}(\mathbf{x})$. The factor $\sqrt{g(\mathbf{x})}$, which we call the *magnification* factor, expresses how a local volume is expanded or contracted under the mapping $\boldsymbol{\phi}$. Amari and Wu [1, 9] suggest that it may be beneficial to increase the separation between sample points in S which are close to the separating boundary, by using a kernel $\tilde{K}$, whose corresponding mapping $\tilde{\boldsymbol{\phi}}$ provides increased separation in H between such samples.

The problem is that the location of the boundary is initially unknown. Amari and Wu therefore suggest that the problem should first be solved in a standard way using some initial kernel K. It should then be solved a second time using a conformal transformation $\tilde{K}$ of the original kernel given by

$$\tilde{K}(\mathbf{x}, \mathbf{x}') = D(\mathbf{x}) K(\mathbf{x}, \mathbf{x}') D(\mathbf{x}') \tag{5}$$

for a suitably chosen positive function $D(\mathbf{x})$. It follows from (2) and (5) that the metric $\tilde{g}_{ij}(\mathbf{x})$ induced by $\tilde{K}$ is related to the original $g_{ij}(\mathbf{x})$ by

$$\begin{aligned} \tilde{g}_{ij}(\mathbf{x}) = {} & D(\mathbf{x})^2 g_{ij}(\mathbf{x}) + D_i(\mathbf{x}) K(\mathbf{x}, \mathbf{x}) D_j(\mathbf{x}) \\ & + D(\mathbf{x}) \{ K_i(\mathbf{x}, \mathbf{x}) D_j(\mathbf{x}) + K_j(\mathbf{x}, \mathbf{x}) D_i(\mathbf{x}) \} \end{aligned} \tag{6}$$

where $D_i(\mathbf{x}) = \partial D(\mathbf{x}) / \partial x_i$ and $K_i(\mathbf{x}, \mathbf{x}) = \partial K(\mathbf{x}, \mathbf{x}') / \partial x_i \,|_{\mathbf{x}'=\mathbf{x}}$. If $g_{ij}(\mathbf{x})$ is to be enlarged in the region of the class boundary, $D(\mathbf{x})$ needs to be largest in

that vicinity, and its gradient needs to be small far away. Note that if D is chosen in this way, the resulting kernel $\tilde{K}$ becomes *data dependent*.

Amari and Wu consider the function

$$D(\mathbf{x}) = \sum_{i \in SV} e^{-\kappa_i \|\mathbf{x}-\mathbf{x}_i\|^2} \tag{7}$$

where κ_i are positive constants. The idea is that support vectors should normally be found close to the boundary, so that a magnification in the vicinity of support vectors should implement a magnification around the boundary. A possible difficulty is that, whilst this is correct for true support vectors, it need not be correct for bound ones.[2] Rather than attempt further refinement of the method embodied in (7), we shall describe here a more direct way of achieving the desired magnification.

2.2 New Approach

The idea here is to choose D so that it decays directly with distance, suitably measured, from the boundary determined by the first-pass solution using K. Specifically we consider

$$D(\mathbf{x}) = e^{-\kappa f(\mathbf{x})^2} \tag{8}$$

where f is given by (1) and κ is a positive constant. This takes its maximum value on the separating surface where $f(\mathbf{x}) = 0$, and decays to $e^{-\kappa}$ at the margins of the separating region where $f(\mathbf{x}) = \pm 1$, This is where the true support vectors lie. In the case where K is the simple inner product in S, the level sets of f and hence of D are just hyperplanes parallel to the separating hyperplane. In that case $|f(\mathbf{x})|$ measures perpendicular distance to the separating hyperplane, taking as unit the common distance of true support vectors from that hyperplane. In the general case the level sets are curved non-intersecting hypersurfaces.

3 Geometry and Magnification

To proceed further we need to consider specific forms for the kernel K.

3.1 RBF Kernels

Consider the Gaussian radial basis function kernel

$$K(\mathbf{x}, \mathbf{x}') = e^{-\|\mathbf{x}-\mathbf{x}'\|^2/2\sigma^2} \ . \tag{9}$$

[2] The method of choosing the κ_i in [9] attempts to meet this difficulty by making the decay rate roughly proportional to the local density of support vectors. Thus isolated support vectors are associated with a low decay rate, so that their influence is minimized.

This is of the general type where $K(\mathbf{x}, \mathbf{x}')$ depends on $\mathbf{x}$ and $\mathbf{x}'$ only through the norm their separation so that

$$K(\mathbf{x}, \mathbf{x}') = k\left(\|\mathbf{x} - \mathbf{x}'\|^2\right) \ . \tag{10}$$

Referring back to (2) it is straightforward to show that the induced metric is Euclidean with

$$g_{ij}(\mathbf{x}) = -2k'(0)\,\delta_{ij} \ . \tag{11}$$

In particular for the Gaussian kernel (9) where $k(\xi) = e^{-\xi/2\sigma^2}$ we have

$$g_{ij}(\mathbf{x}) = \frac{1}{\sigma^2}\,\delta_{ij} \tag{12}$$

so that $g(\mathbf{x}) = \det\{g_{ij}(\mathbf{x})\} = 1/\sigma^{2n}$ and hence the volume magnification is the constant

$$\sqrt{g(\mathbf{x})} = \frac{1}{\sigma^n} \ . \tag{13}$$

3.2 Inner Product Kernels

For another class of kernel, $K(\mathbf{x}, \mathbf{x}')$ depends on $\mathbf{x}$ and $\mathbf{x}'$ only through their inner product so that

$$K(\mathbf{x}, \mathbf{x}') = k(\mathbf{x} \cdot \mathbf{x}') \ . \tag{14}$$

A well known example is the inhomogeneous polynomial kernel

$$K(\mathbf{x}, \mathbf{x}') = (1 + \mathbf{x} \cdot \mathbf{x}')^d \tag{15}$$

for some positive integer d. For kernels of this type, it follows from (2) that the induced metric is

$$g_{ij}(\mathbf{x}) = k'(\|\mathbf{x}\|^2)\,\delta_{ij} + k''(\|\mathbf{x}\|^2)\,x_i x_j \ . \tag{16}$$

To evaluate the magnification factor, we need the following:

Lemma 1. *Suppose that* $\mathbf{a} = (a_1, \ldots, a_n)$ *is a vector and that the components* A_{ij} *of a matrix* $\mathbf{A}$ *are of the form* $A_{ij} = \alpha\delta_{ij} + \beta a_i a_j$. *Then* $\det \mathbf{A} = \alpha^{n-1}\left(\alpha + \beta\|\mathbf{a}\|^2\right)$.

It follows that, for kernels of the type (14), the magnification factor is

$$\sqrt{g(\mathbf{x})} = \sqrt{k'(\|\mathbf{x}\|^2)^n \left(1 + \frac{k''(\|\mathbf{x}\|^2)}{k'(\|\mathbf{x}\|^2)}\,\|\mathbf{x}\|^2\right)} \tag{17}$$

so that for the inhomogeneous polynomial kernel (15), where $k(\xi) = (1+\xi)^d$,

$$\sqrt{g(\mathbf{x})} \ = \ \sqrt{d^n\,(1 + \|\mathbf{x}\|^2)^{n(d-1)-1}\,(1 + d\|\mathbf{x}\|^2)} \ . \tag{18}$$

For $d > 1$, the magnification factor (18) is a radial function, taking its minimum value at the origin and increasing, for $\|\mathbf{x}\| \gg 1$, as $\|\mathbf{x}\|^{n(d-1)}$. This suggests it might be most suitable, for binary classification, when one the classes forms a bounded cluster centered on the origin.

3.3 Conformal Kernel Transformations

To demonstrate the approach, we consider the case where the initial kernel K in (5) is the Gaussian RBF kernel (9). For illustration, consider the binary classification problem shown in Fig. 1, where 100 points have been selected at random in the square as a training set, and classified according to whether they fall above or below the curved boundary, which has been chosen as e^{-4x^2} up to a linear transform.

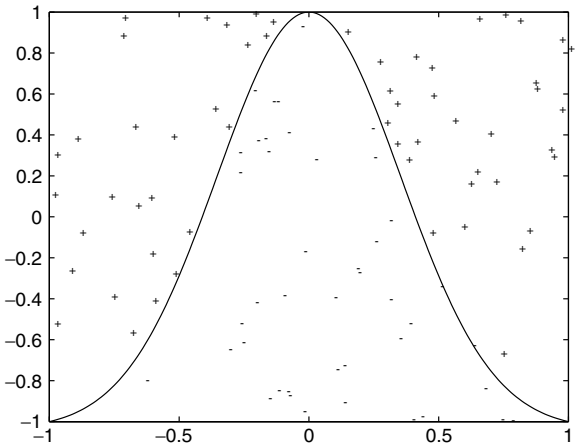

Fig. 1. A training set of 100 random points classified according to whether they lie above (+) or below (−) the Gaussian boundary shown

Our approach requires a first-pass solution using conventional methods. Using a Gaussian radial basis kernel with width 0.5 and soft-margin parameter $C = 10$, we obtain the solution shown in Fig. 2. This plots contours of the discriminant function f, which is of the form (1). For sufficiently large samples, the zero contour in Fig. 2 should coincide with the curve in Fig. 1.

To proceed with the second-pass we need to use the modified kernel given by (5) where K is given by (9) and D is given by (8). It is interesting first to calculate the general metric tensor $\tilde{g}_{ij}(\mathbf{x})$ when K is the Gaussian RBF kernel (9) and $\tilde{K}$ is derived from K by (5). Substituting in (6), and observing that in this case $K(\mathbf{x}, \mathbf{x}) = 1$ while $K_i(\mathbf{x}, \mathbf{x}) = K_j(\mathbf{x}, \mathbf{x}) = 0$, we obtain

$$\tilde{g}_{ij}(\mathbf{x}) = \frac{D(\mathbf{x})^2}{\sigma^2}\,\delta_{ij} + D_i(\mathbf{x})D_j(\mathbf{x})\ . \tag{19}$$

The $\tilde{g}_{ij}(\mathbf{x})$ in (19) are of the form considered in Lemma 1. Observing that $D_i(\mathbf{x})$ are the components of $\nabla D(\mathbf{x}) = D(\mathbf{x})\nabla \log D(\mathbf{x})$, it follows that the ratio of the new to the old magnification factors is given by

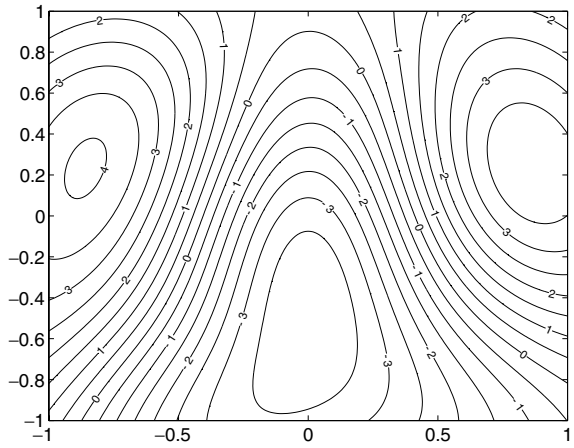

Fig. 2. First-pass SVM solution to the problem in Fig. 1 using a Gaussian kernel. The contours show the level sets of the discriminant function f defined by (1)

$$\sqrt{\frac{\tilde{g}(\mathbf{x})}{g(\mathbf{x})}} \;=\; D(\mathbf{x})^n \sqrt{1+\sigma^2 \|\nabla \log D(\mathbf{x})\|^2}\;. \tag{20}$$

This is true for any positive scalar function $D(\mathbf{x})$. Let us now use the function given by (8) for which

$$\log D(\mathbf{x}) \;=\; -\kappa f(\mathbf{x})^2 \tag{21}$$

where f is the first-pass solution given by (1) and shown, for example, in Fig. 2. This gives

$$\sqrt{\frac{\tilde{g}(\mathbf{x})}{g(\mathbf{x})}} \;=\; \exp\left\{-n\kappa f(\mathbf{x})^2\right\} \sqrt{1+4\kappa^2\sigma^2 f(\mathbf{x})^2 \|\nabla f(\mathbf{x})\|^2}\;. \tag{22}$$

This means that

1. *the magnification is constant on the separating surface* $f(\mathbf{x}) = 0$*;*
2. *along contours of constant* $f(\mathbf{x}) \neq 0$*, the magnification is greatest where the contours are closest.*

The latter is because of the occurrence of $\|\nabla f(\mathbf{x})\|^2$ in (22). The gradient points uphill orthogonally to the local contour, hence in the direction of steepest ascent; the larger its magnitude, the steeper is the ascent, and hence the closer are the local contours. This character is illustrated in Fig. 3 which shows the magnification factor for the modified kernel based on the solution of Fig. 2. Notice that the magnification is low at distances remote from the boundary.

Solving the original problem again, but now using the modified kernel $\tilde{K}$, we obtain the solution shown in Fig. 4. Comparing this with the first-pass

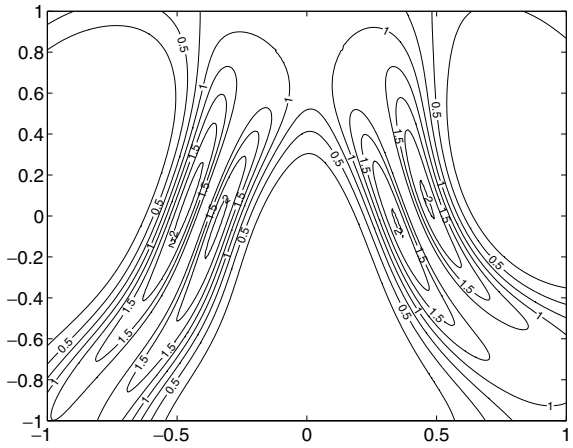

Fig. 3. Contours of the magnification factor (22) for the modified kernel using $D(\mathbf{x}) = \exp\{-\kappa f(\mathbf{x})^2\}$ with f defined by the solution of Fig. 2

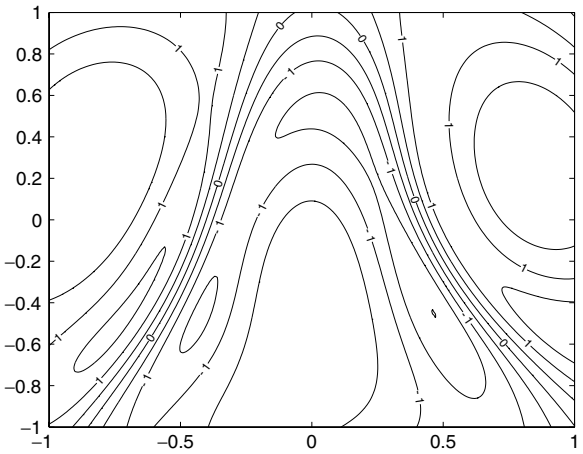

Fig. 4. Second-pass solution using the modified kernel

solution of Fig. 2, notice the steeper gradient in the vicinity of the boundary, and the relatively flat areas remote from the boundary.

In this instance the classification provided by the modified solution is little improvement on the original classification. This an accident of the choice of the training set shown in Fig. 1. We have repeated the experiment 10000 times, with a different choice of 100 training sites and 1000 test sites on each occasion, and have found an average of 14.5% improvement in classification

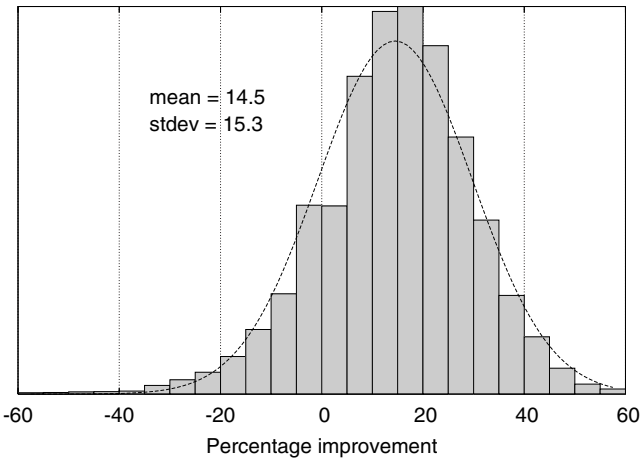

Fig. 5. Histogram of the percentage improvement in classification, over 10000 experiments, together with a normal curve with the same mean and standard deviation

performance.[3] A histogram of the percentage improvement, over the 10000 experiments, together with a normal curve with the same mean and standard deviation, is shown in Fig. 5.

3.4 Choice of κ

A presently unresolved issue is how best to make a systematic choice of κ. It is clear that κ is dimensionless, in the sense of being scale invariant. Suppose all input dimensions in the input space S are multiplied by a positive scalar a. To obtain the same results for the first-pass solution, a new $\sigma_a = a\sigma$ must be used in the Gaussian kernel (9). This leads to the first-pass solution f_a where $f_a(a\mathbf{x}) = f(\mathbf{x})$ with f being the initial solution using σ. It then follows from (5) and (8) that *provided κ is left unchanged* the rescaled second-pass solution automatically satisfies the corresponding covariance relation $\tilde{f}_a(a\mathbf{x}) = \tilde{f}(\mathbf{x})$ where $\tilde{f}$ was the original second-pass solution using σ.

It may appear that there is a relationship between κ and σ in the expression (22) for the magnification ratio. Using a corresponding notation, however, it is straightforward to show that the required covariance $\tilde{g}_a(a\mathbf{x})/g_a(a\mathbf{x}) = \tilde{g}(\mathbf{x})/g(\mathbf{x})$ also holds provided κ is left unchanged. The reason is that $\sigma\|\nabla f(\mathbf{x})\|$ is invariant under rescaling since a multiplies σ and divides $\nabla f(\mathbf{x})$.

Possibly κ should depend on the dimension n of the input space. This has not yet been investigated. In the trials reported above, it was found that a

[3] If there are 50 errors in 1000 for the original solution and 40 errors for the modified solution, we call this a 20% improvement. If there are 60 errors for the modified solution, we call it a −20% improvement.

suitable choice was $\kappa = 0.25$. We note that this is approximately the reciprocal of the maximum value obtained by f in the first pass solution.

In the following we introduce the second approach which concerns how to scale the optimal position of the discriminating hyperplane.

4 Scaling the Position of the Discriminating Hyperplane

The original motivation of the SVM relates to maximizing the margin (the distance from the separating hyperplane to the nearest example). The essence of the SVM is to rely on the set of examples which take extreme values, the so-called support vectors. But from the statistics of extreme values, we know that the disadvantage of such an approach is that information contained in most samples (not extreme) values is lost, so that such an approach would be expected to be less efficient than one which takes into account the lost information. These ideas were explored in [5]. We give here a summary of results.

To introduce the model, consider for simplicity a one-dimensional classification problem. Suppose that we have two populations, one of positive variables x and one of negative variables y, and that we observe t positive examples $x(1), \ldots, x(t) > 0$ and t negative examples $y(1), \ldots, y(t) < 0$. Since this case is separable, the SVM will use the threshold

$$z(t) = \frac{1}{2}\,\underline{x}(t) + \frac{1}{2}\,\overline{y}(t) \tag{23}$$

for classifying future cases, where

$$\underline{x}(t) = \min\{x(i) : i = 1, \ldots, t\}$$

is the minimum of the positive examples and

$$\overline{y}(t) = \max\{y(i) : i = 1, \ldots, t\}$$

is the maximum of the negative examples. A newly observed ξ will be classified as belonging to the x or y populations, depending on whether $\xi > z(t)$ or $\xi < z(t)$. This is pictured in Fig. 6.

4.1 Generalization Error

If a new ξ is observed, which may belong to either the x or y populations, an error occurs if ξ lies in the region between the dashed and solid lines shown in Fig. 6. The dashed line is fixed at the origin, but the solid line is located at the threshold $z(t)$ which, like ξ, is a random variable. A misclassification will occur if either $0 < \xi < z(t)$ or $z(t) < \xi < 0$.

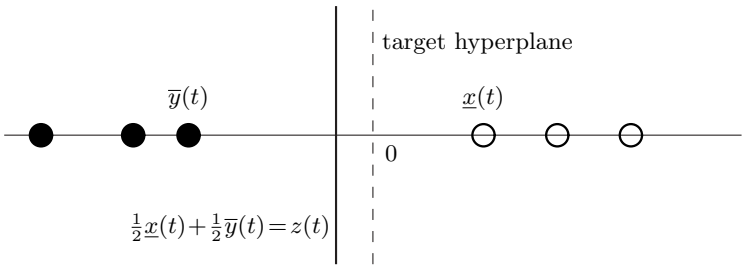

Fig. 6. Schematic representation of the one-dimensional support vector machine. The task is to separate the disks (*filled*) from the circles (*hollow*). The true separation is assumed to be given by the *dashed vertical line*. After learning t examples, the separating hyperplane for the support vector machine is at $z(t) = \frac{1}{2}\underline{x}(t) + \frac{1}{2}\overline{y}(t)$. The error region is then the region between the *dashed line* and the *solid line*

We define the *generalization error* $\varepsilon(t)$ to be the *probability* of misclassification. The generalization error $\varepsilon(t)$ is therefore a random variable whose distribution depends on the distributions of the x and y variables. In [5] it is shown that if there is an equal prior probability of ξ belonging to the x or y populations then, under a wide class of distributions for the x and y variables, the mean and variance of $\varepsilon(t)$, when defined in terms of the symmetric threshold (23), in the limit as $t \to \infty$ have the values

$$E\big(\varepsilon(t)\big) = 1/4t \tag{24}$$

$$\mathrm{var}\big(\varepsilon(t)\big) = 1/16t^2\ . \tag{25}$$

For example, suppose that the $x(t)$ are independent and uniformly distributed on the positive unit interval $[0, 1]$ and that the $y(t)$ are similarly distributed on the negative unit interval $[-1, 0]$. Then the exact value for the mean of the generalization error, for any $t > 0$, is in fact $t/(t + 1)(4t + 2) \approx 1/4t$. If the $x(t)$ have positive exponential distributions and the $y(t)$ have negative exponential distributions, the exact value for the mean is $1/(4t + 2) \approx 1/4t$. The generality of the limiting expressions (24) and (25) derives from results of extreme value theory [6, Chap. 1]. It is worth pointing out that results, such as (25), for the variance of the generalization error of the SVM have not previously been widely reported.

4.2 The Non-symmetric SVM

The threshold (23) follows the usual SVM practice of choosing the mid-point of the margin to separate the positive and negative examples. But if positive and negative examples are scaled differently in terms of their distance from the separating hyperplane, the mid-point may not be optimal. Let us therefore consider the general threshold

$$z(t) = \lambda\underline{x}(t) + \mu\overline{y}(t) \qquad (\lambda + \mu = 1)\ . \tag{26}$$

In separable cases (26) will correctly classify the observed examples for any $0 \leq \lambda \leq 1$. The symmetric SVM0 corresponds to $\lambda = 1/2$. The cases $\lambda = 0$ and $\lambda = 1$ were said in [5] to correspond to the "worst learning machine". We now calculate the distribution of the generalization error for the general threshold (26).

Note that the generalization error can be written as

$$\varepsilon_\lambda(t) = P\big(0 < \xi < z(t)\big)\, I(z(t) > 0) + P\big(z(t) < \xi < 0\big)\, I(z(t) < 0) \tag{27}$$

where $I(A)$ is the $\{0,1\}$-valued indicator function of the event A. To calculate the distribution of $\varepsilon_\lambda(t)$ we need to know the distributions of ξ and $z(t)$. To be specific, assume that each $x(i)$ has a positive exponential distribution with scale parameter a and each $y(i)$ has a negative exponential distribution with scale parameter b. It is then straightforward to show that $z(t)$ defined by (26) has an asymmetric Laplace distribution such that

$$P(z(t) > \zeta) = \left(\frac{\lambda a}{\lambda a + \mu b}\right) e^{-(t/\lambda a)\zeta} \qquad (\zeta > 0) \tag{28}$$

$$P(z(t) < \zeta) = \left(\frac{\mu b}{\lambda a + \mu b}\right) e^{(t/\mu b)\zeta} \qquad (\zeta < 0)\,. \tag{29}$$

Let us assume furthermore that a newly observed ξ has probability $1/2$ of having the same distribution as either $x(i)$ or $y(i)$. In that case ξ also has an asymmetric Laplace distribution and (27) becomes

$$\varepsilon_\lambda(t) = \frac{1}{2}\left\{1 - e^{-z(t)/a}\right\} I(z(t) > 0) + \frac{1}{2}\left\{1 - e^{z(t)/b}\right\} I(z(t) < 0)\,. \tag{30}$$

Making use of (28) and (29) for the distribution of $z(t)$, it follows that for any $0 \leq p \leq 1$,

$$P\big(2\varepsilon_\lambda(t) > p\big) = \left(\frac{\lambda a}{\lambda a + \mu b}\right)(1-p)^{t/\lambda} + \left(\frac{\mu b}{\lambda a + \mu b}\right)(1-p)^{t/\mu} \tag{31}$$

which implies that $2\varepsilon_\lambda(t)$ has a mixture of $\mathrm{Beta}(1, t/\lambda)$ and $\mathrm{Beta}(1, t/\mu)$ distributions.[4] It follows that the mean of $2\varepsilon_\lambda(t)$ is

$$\left(\frac{\lambda a}{\lambda a + \mu b}\right)\left(\frac{\lambda}{t + \lambda}\right) + \left(\frac{\mu b}{\lambda a + \mu b}\right)\left(\frac{\mu}{t + \mu}\right) \tag{32}$$

so that for large t, since $\lambda, \mu \leq 1$, the expected generalization error has the limiting value

$$E\big(\varepsilon_\lambda(t)\big) = \frac{1}{2t}\left\{\frac{\lambda^2 a + \mu^2 b}{\lambda a + \mu b}\right\}. \tag{33}$$

[4]The error region always lies wholly to one side or other of the origin so that, under present assumptions, the probability that ξ lies in this region, and hence the value of the generalization error $\varepsilon_\lambda(t)$, is never more than $1/2$.

A corresponding, though lengthier, expression can be found for the variance. Note that the limiting form (33) holds for a wide variety of distributions, for example if each $x(i)$ is uniformly distributed on $[0, a]$ and each $y(i)$ is uniformly distributed on $[-b, 0]$, compare [5].

Optimal Value of λ

What is the optimal value of λ if the aim is to minimize the expected generalization error given by (33)? The usual symmetric SVM chooses $\lambda = 1/2$. In that case we have

$$E\left(\varepsilon_{\frac{1}{2}}(t)\right) = \frac{1}{4t} \tag{34}$$

$$\mathrm{var}\left(\varepsilon_{\frac{1}{2}}(t)\right) = \frac{1}{16t^2} \tag{35}$$

as previously in (24) and (25). Interestingly, this shows that those results are independent of the scaling of the input distributions. However, if $a \neq b$, an improvement may be possible.

An alternative, which comes readily to mind, is to divide the margin in the inverse ratio of the two scales by using

$$\lambda^\dagger = \frac{b}{a+b}\,. \tag{36}$$

We then have

$$E(\varepsilon_{\lambda^\dagger}(t)) = \frac{1}{4t} \tag{37}$$

$$\mathrm{var}(\varepsilon_{\lambda^\dagger}(t)) = \frac{1}{16t^2}\left\{1 + \left(\frac{a-b}{a+b}\right)^2\right\}. \tag{38}$$

Notice that, for $\lambda = \lambda^\dagger$, the expected generalization error is unchanged, but the variance is increased, unless $a = b$.

It is easy to verify, however, that the minimum of (33) in fact occurs at

$$\lambda^* = \frac{\sqrt{b}}{\sqrt{a}+\sqrt{b}} \tag{39}$$

for which

$$E(\varepsilon_{\lambda^*}(t)) = \frac{1}{4t}\left\{1 - \left(\frac{\sqrt{a}-\sqrt{b}}{\sqrt{a}+\sqrt{b}}\right)^2\right\} \tag{40}$$

$$\mathrm{var}(\varepsilon_{\lambda^*}(t)) = \frac{1}{16t^2}\left\{1 - \left(\frac{\sqrt{a}-\sqrt{b}}{\sqrt{a}+\sqrt{b}}\right)^4\right\} \tag{41}$$

showing that both mean and variance are reduced for $\lambda = \lambda^*$ compared with $\lambda = 1/2$ or $\lambda = \lambda^\dagger$.

5 Conclusions

In this chapter we have introduced two methods for improving the performance of the SVM. One method is geometry-oriented, which concerns a data-dependent way to scale the kernel function so that the separation between two classes is enlarged. The other is statistics-motivated, which concerns how to optimize the position of the discriminating hyperplane based on the different scales of the two classes of data. Both methods have proved to be effective for reducing the generalization error of SVM. Combining the two methods together, we would expect a further reduction on the generalization error. This is currently under investigation.

Acknowledgement

Partially supported by grants from UK EPSRC (GR/R54569), (GR/S20574) and (GR/S30443).

References

1. Amari S, Si Wu (1999) Improving Support Vector Machine classifiers by modifying kernel functions. Neural Networks 12:783–789 205, 206, 207
2. Burges CJC (1999) Geometry and invariance in kernel based methods In: Burges C, Schölkopf B, Smola A (eds) Advances in Kernel Methods—Support Vector Learning, MIT Press, 89–116 205
3. Cristanini N, Shawe-Taylor J (2000) An Introduction to Support Vector Machines. Cambridge University Press, Cambridge, UK 205
4. Cucker F, Smale S (2001) On the mathematical foundations of learning. Bulletin of the AMS 39(1):1–49 207
5. Feng J, Williams P (2001) The generalization error of the symmetric and scaled Support Vector Machines. IEEE Transactions on Neural Networks 12(5):1255–1260 206, 214, 215, 216, 217
6. Leadbetter MR, Lindgren G, Rootzén H (1983) Extremes and Related Properties of Random Sequences and Processes. Springer-Verlag, New York 215
7. Poggio T, Mukherjee S, Rifkin R, Raklin A, Verri A (2002) B. In: Winkler J, Niranjan M (eds) Uncertainty in Geometric Computations. Kluwer Academic Publishers, 131–141 206
8. Schölkopf B, Smola A (2002) Learning with Kernels. MIT Press, UK 205
9. Si Wu, Amari S (2001) Conformal transformation of kernel functions: a data dependent way to improve Support Vector Machine classifiers. Neural Processing Letters 15:59–67 205, 206, 207, 208
10. Vapnik V (1995) The Nature of Statistical Learning Theory. Springer, NY 205