

Maximum Margin Classifiers with Specified False Positive and False Negative Error Rates

J. Saketha Nath

C. Bhattacharyya

Abstract

This paper addresses the problem of maximum margin classification given the moments of class conditional densities and the false positive and false negative error rates. Using Chebyshev inequalities, the problem can be posed as a second order cone programming problem. The dual of the formulation leads to a geometric optimization problem, that of computing the distance between two ellipsoids, which is solved by an iterative algorithm. The formulation is extended to non-linear classifiers using kernel methods. The resultant classifiers are applied to the case of classification of unbalanced datasets with asymmetric costs for misclassification. Experimental results on benchmark datasets show the efficacy of the proposed method.

1 Introduction

In many classification tasks the cost of misclassification is different for each class. For instance, in case of medical diagnosis of cancer (Kononenko, 2001), the cost of misclassifying a normal patient is far less than that of misclassifying a cancer patient. Hence there is need to design classifiers that have some bias towards a particular class. Also, the number of patients with cancer is far less than those who are normal. Hence, in such situations, the training data will be highly unbalanced. Traditional classification methods like SVM (Vapnik, 1998) do not address these issues satisfactorily. In this paper we study this problem in the context of two classes, when data is summarized by the moments of class conditional densities. We also assume that the maximum false positive and false negative error rates (η_1, η_2 respectively) that can be tolerated are given. For instance, in the case of Medical diagnosis, one can allow a low η_1 and a relatively high η_2 . In this way we can model the bias towards the positive class.

Let $\mathcal{D} = \{(\mathbf{x}_i, y_i) | \mathbf{x}_i \in \mathbb{R}^n, y_i = \{1, -1\}, i = 1, \dots, m\}$ be the training dataset consisting of data points \mathbf{x}_i and labels y_i . Let \mathbf{X}_1 represent the random vector that generates examples of class 1 and \mathbf{X}_2 represent that of class -1 . Let the mean and covariance of \mathbf{X}_i be $\mu_i \in \mathbb{R}^n$ and $\Sigma_i \in \mathbb{R}^{n \times n}$ respectively for $i = 1, 2$. Σ_1, Σ_2 are symmetric positive semi-definite.

Previously (Lanckriet et al., 2002; Lanckriet et al., 2003) addressed the problem of classification given μ_1, μ_2, Σ_1 and Σ_2 in a minimax setting (minimax probability machine). In their approach, a Chebyshev inequality is used to bound the error of misclassification. Biased Minimax Probability Machine (Huang et al., 2004) extends the minimax probability machine to the case of classification with asymmetric costs. In this method, the probability of correctly classifying positive examples (p_1) is maximized, while keeping a lower bound on that for the negative class (p_2). However, with this method, at the optimum, p_1 may turn out to be less than p_2 . Other techniques such as the methods of sampling (Chawla et al., 2002; Kubat & Matwin, 1997), the methods of adapting the thresholds and adjusting costs (Bach et al., 2005; Provost, 2000; Cardie & Howe, 1997) can be used to incorporate a certain bias into the learning methods. However, the sampling methods favor the more important class by down-sampling (removing) some instances of the less important class or up-sampling (duplicating) some instances of the more important class. Down-sampling will lose information, while up-sampling may introduce noise. In the case of the methods that adjust costs, it is usually hard to build direct quantitative connections to the biased classifiers performance. These methods therefore fail to provide a rigorous approach to the task of classification where preferential bias towards one class is needed.

In this paper, we address the problem by designing maximum margin classifiers given $\mu_1, \mu_2, \Sigma_1, \Sigma_2, \eta_1$ and η_2 . Using the Chebyshev inequality, the problem is posed as a Second Order Cone Programming problem (SOCP). SOCPs are a special class of non-linear convex optimization problems, which can be efficiently solved by interior point codes (Lobo et al., 1998). Interestingly, the dual of the formulation leads to an elegant geometric optimization problem, that of computing the distance between two ellipsoids. This observation immediately leads to a fast iterative algorithm, based on the approach of (Lin & Han, 2002). Using kernel methods, the original formulation can be extended to the case of non-linear classifiers. This kernelized formulation turns

out to be similar to the original and hence can be cast as an SOCP or solved using the iterative algorithm for finding the distance between ellipsoids.

The paper is organized as follows. In section 2 we review the related past work on Large Margin Classification. The main contributions of the paper are in section 3. Experimental results for the formulations are shown in section 4. The concluding section 5 summarizes the main contributions and future directions of work.

2 Large Margin Classifiers

In this section we review the SVMs (section 2.1), which are the state-of-the-art classifiers. The derivation of the proposed formulation requires understanding of these SVM formulations. In section 2.2 we review the geometric interpretation of the SVMs. This will help in understanding the analogy between the SVMs and the proposed classifiers, whose geometric interpretation is of finding minimum distance between two ellipsoids. We conclude the section with a brief discussion on the variants of SVMs (section 2.3) that can be used in biased learning.

2.1 Review of SVMs Consider the problem of classification using hyperplanes. In such cases the classifier is defined as

$$(2.1) \quad f(\mathbf{x}) = \text{sign}(\mathbf{w}^\top \mathbf{x} - b).$$

The classifier essentially divides the space into two half-spaces, so that all datapoints belonging to the positive class lies in $\mathbf{w}^\top \mathbf{x} - b > 0$ and all datapoints lying in the negative class belongs to the other halfspace $\mathbf{w}^\top \mathbf{x} - b < 0$. Given a dataset the problem of computing the classifier is then equivalent to finding w, b so that the predicted labels ($f(\mathbf{x})$) is equal to the given labels y . This requirement can be enforced via the constraints $y_i(\mathbf{w}^\top \mathbf{x}_i - b) \geq 0$. The constraints ensure that the label prediction error of the hyperplane classifier on the training set is low. However, if (\mathbf{w}, b) is a solution that satisfies the constraints, then $(\gamma \mathbf{w}, \gamma b)$, for any $\gamma \in \mathbb{R}$, also satisfies the constraints. In order to handle this, the constraints can be modified as

$$(2.2) \quad y_i(\mathbf{w}^\top \mathbf{x}_i - b) \geq 1$$

The set of hyperplanes $\mathbf{w}^\top \mathbf{x} - b = -1$ and $\mathbf{w}^\top \mathbf{x} - b = 1$ are called the supporting hyperplanes. The distance between the set of supporting hyperplanes is termed as margin. It is easy to see that margin is $\frac{2}{\|\mathbf{w}\|_2}$. The generalization error for a classifier f is measured as $R(f) = P(Y \neq f(X))$. It can be shown (Vapnik,

1998) that

$$R(f) \leq R_{emp}(f) + \Phi\left(\frac{h}{m}\right)$$

holds with probability greater than $1 - \delta$ where

$$\Phi\left(\frac{h}{m}\right) = \sqrt{\frac{1}{m} \left(h \log\left(\frac{2m}{h}\right) + 1 + \log\left(\frac{1}{4\delta}\right) \right)}$$

and $R_{emp}f = \frac{1}{m} \sum_{i=1}^m I_{\{y_i \neq f(x_i)\}}$. The parameter h is the VC dimension for the set of classifiers, in the present context it is over all linear classifiers. For linear classifiers the VC dimension is related to the margin via the following theorem.

THEOREM 2.1. *Consider the dataset \mathcal{D} defined in the previous section and let $\max_i \|\mathbf{x}_i\| = R$. The VC dimension h of the classifiers as defined in (2.1), is given by the following relation*

$$h \leq \min(\lceil R^2 \|w\|^2 \rceil, d) + 1$$

To compute a classifier with low generalization error one needs to find $f(\cdot)$ so that both R_{emp} and $\Phi(\frac{h}{m})$ are minimized. It is to be noted that $\Phi(\cdot)$ is an increasing function of h and thus if $\|w\|$ is low, $\Phi(\cdot)$ is also low. Hence maximizing the margin, or equivalently minimizing $\|w\|$ leads to lower generalization error provided that R_{emp} is low.

The well known hard-margin SVM formulation (Vapnik, 1998) maximizes the margin subject to the constraints (2.2):

$$(2.3) \quad \begin{aligned} \min_{\mathbf{w}, b} \quad & \frac{1}{2} \|\mathbf{w}\|_2^2 \\ \text{s.t.} \quad & y_j(\mathbf{w}^\top \mathbf{x}_j - b) \geq 1, \quad j = 1, \dots, m \end{aligned}$$

The constraints ensure that R_{emp} is 0. However if the data is not linearly separable then it is not possible to obtain $R_{emp} = 0$. This problem is then approached by hunting for a classifier which has the lowest value of R_{emp} amongst the class of linear classifiers with low VC dimension. This problem turns out to be difficult. To alleviate this problem one considers the following loss function, which is called the hinge loss function

$$\frac{1}{m} \sum_{i=1}^m \max(1 - y_i(\mathbf{w}^\top \mathbf{x}_i + b), 0)$$

This is actually an upperbound on R_{emp} and can be easily implemented via linear inequalities. Consider now the following objective

$$(2.4) \quad \begin{aligned} \min_{\mathbf{w}, b, \xi_j} \quad & \sum_{j=1}^m \xi_j \\ \text{s.t.} \quad & y_j(\mathbf{w}^\top \mathbf{x}_j - b) - 1 + \xi_j \geq 0, \\ & \xi_j \geq 0, \quad j = 1, \dots, m, \|\mathbf{w}\|_2 \leq W \end{aligned}$$

where, W is some positive real number. The constraint $\|\mathbf{w}\|_2 \leq W$ is put in order that the hyperplane classifier achieves good generalization (Vapnik, 1998). Geometrically, this constraint puts a lower bound on the separation (margin) between the set of supporting hyperplanes.

The problem (2.4) is an instance of Second Order Cone Programming problem. An SOCP problem is a convex optimization problem with a linear objective function and second order cone constraints (SOC). An SOC constraint on the variable $\mathbf{x} \in \mathbb{R}^n$ is of the form

$$(2.5) \quad \mathbf{c}^\top \mathbf{x} + d \geq \|\mathbf{Ax} + \mathbf{b}\|_2$$

where $\mathbf{b} \in \mathbb{R}^m$, $\mathbf{c} \in \mathbb{R}^n$, $\mathbf{A} \in \mathbb{R}^{m \times n}$ are given.

The problem (2.4) can be equivalently written as the following convex quadratic programming problem:

$$(2.6) \quad \begin{aligned} \min_{\mathbf{w}, \mathbf{b}, \xi_j} \quad & \frac{1}{2} \|\mathbf{w}\|_2^2 + C \sum_{j=1}^m \xi_j \\ \text{s.t.} \quad & y_j (\mathbf{w}^\top \mathbf{x}_j - b) - 1 + \xi_j \geq 0, \\ & \xi_j \geq 0, j = 1, \dots, m \end{aligned}$$

(2.4) is the famous SVM soft-margin formulation. The parameters C and W are related. However, W has the elegant geometric interpretation as a lower bound on the margin.

2.2 Geometric Interpretation of SVMs In the following text we discuss the geometrical interpretation of the above optimization problem (2.3), due to (Bennett & Bredensteiner, 2000). The Wolfe dual (Fletcher, 1989) of this primal is (α_i are the Lagrange multipliers):

$$(2.7) \quad \begin{aligned} \min_{\alpha_i} \quad & \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j x_i^\top x_j - \sum_i \alpha_i \\ \text{s.t.} \quad & \sum_i \alpha_i y_i = 0, \alpha_i \geq 0, \forall i \end{aligned}$$

Now suppose $\sum_{y_i=1} \alpha_i = \sum_{y_i=-1} \alpha_i = \lambda$. Define $\beta_i = \frac{\alpha_i}{\lambda}$. Then, the dual can be written as:

$$(2.8) \quad \begin{aligned} \min_{\beta_i, \lambda} \quad & \frac{\lambda^2}{2} \sum_{i,j} \beta_i \beta_j y_i y_j x_i^\top x_j - 2\lambda \\ \text{s.t.} \quad & 0 \leq \beta_i \leq 1, \sum_i \beta_i = 1, \lambda \geq 0 \end{aligned}$$

Since the above dual is a convex optimization problem one can find optimal value of λ by minimizing the objective keeping β_i fixed. Suppose $\sum_{i,j} \beta_i \beta_j y_i y_j x_i^\top x_j = a^2$. Then, clearly the objective is minimized at $\lambda = \frac{2}{a^2}$ and is equal to $\frac{-2}{a^2}$. Thus the dual can be expressed as below:

$$(2.9) \quad \begin{aligned} \min_{\beta_i} \quad & \sum_{i,j} \beta_i \beta_j y_i y_j x_i^\top x_j \\ \text{s.t.} \quad & 0 \leq \beta_i \leq 1, \sum_i \beta_i = 1 \end{aligned}$$

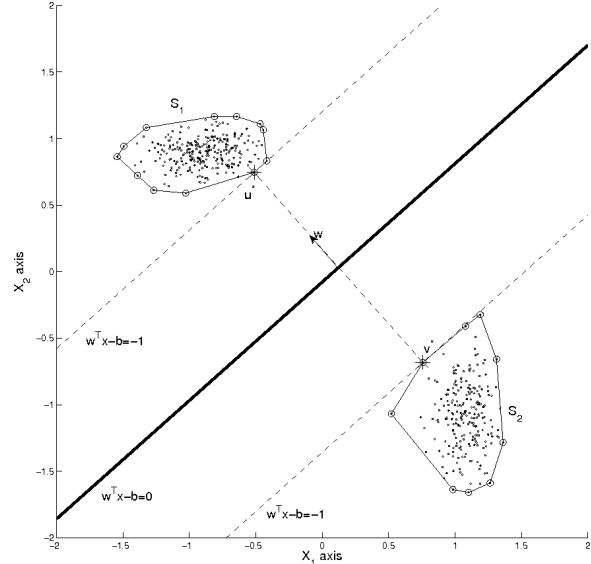


Figure 1: Illustration showing the geometric interpretation of the SVM formulation

Let $u = \sum_{y_i=1} \beta_i x_i$ and $v = \sum_{y_i=-1} \beta_i x_i$. Clearly $u \in S_1$ and $v \in S_2$ where, S_1 and S_2 are the convex hulls formed by the positive and negative labeled datapoints in \mathcal{D} . Thus the final form of dual is:

$$(2.10) \quad \begin{aligned} \min_{u,v} \quad & \|u - v\|_2^2 \\ \text{s.t.} \quad & u \in S_1, v \in S_2 \end{aligned}$$

Geometrically this means that, the optimization is equivalent to finding the closest points u^* and v^* on the convex hulls formed by the positive and negative labeled datapoints in \mathcal{D} . It can also be shown that $w = u^* - v^*$ i.e., the optimal hyperplane is perpendicular to the line joining the closest points in S_1 and S_2 (see figure 1). There are efficient algorithms that solve the dual. (Platt, 1999) proposes a Sequential Minimal Optimization (SMO) algorithm to solve dual (2.7). (Keerthi et al., 2000) proposes a fast iterative nearest point algorithm to solve dual (2.10).

The geometrical interpretation of the proposed classifier turns out to be that finding closest points of ellipsoids, whose centroids are the means and the shapes are described by the covariance matrices of the class conditional densities. This is the analogy between the proposed method and the SVM. As in the case of SVMs, a simple iterative scheme for solving the dual also exists in the case of the proposed classifiers.

2.3 Biased Classification using SVMs Previous researchers have proposed simple methods like penalizing the ξ_i in the SVM problem (2.6) with costs different

for different classes. For instance, the term $C \sum_i \xi_j$ in (2.6) is replaced with $C_+ \sum_{y_j=1} \xi_j + C_- \sum_{y_j=-1} \xi_j$ and the formulation can be rewritten as:

$$\begin{aligned} \min_{\mathbf{w}, b, \xi_j} \quad & \frac{1}{2} \|\mathbf{w}\|_2^2 + C_+ \sum_{y_j=1} \xi_j + C_- \sum_{y_j=-1} \xi_j \\ \text{s.t.} \quad & y_j(\mathbf{w}^\top \mathbf{x}_j - b) - 1 + \xi_j \geq 0, \\ (2.11) \quad & \xi_j \geq 0, j = 1, \dots, m \end{aligned}$$

By varying the values of C_+ and C_- , one can achieve different bias for the classes. Though such simple methods work well in practice, the theoretical connections between the values of C_+ , C_- chosen and the bias obtained towards a particular class are hard to obtain. On the contrary, the proposed method obtains a classifier that theoretically guarantees a maximum error rate of η_1 on the positive class and a maximum error rate of η_2 on the negative class.

3 A New Maximum Margin Formulation

In this section, we present a novel formulation we use to solve the above discussed classification problem. The dual of this formulation and its geometric interpretation are discussed in section 3.1. An iterative algorithm to solve the dual will be presented in section 3.2. In section 3.3, we extend the formulation to non-linear classifiers and suggest ways of solving the kernelized primal and dual problems.

Let $\mathbf{w}^\top \mathbf{x} - b = 0$ be the discriminating hyperplane. Let us denote the corresponding positive and negative half spaces by:

$$\mathcal{H}_1(\mathbf{w}, b) = \{\mathbf{x} | \mathbf{w}^\top \mathbf{x} > b\}, \mathcal{H}_2(\mathbf{w}, b) = \{\mathbf{x} | \mathbf{w}^\top \mathbf{x} < b\}$$

As mentioned above, we wish to design a maximum margin classifier such that the false positive and false negative error rates do not exceed η_1 and η_2 . To this end, consider the following problem:

$$\begin{aligned} \min_{\mathbf{w}, b} \quad & \frac{1}{2} \|\mathbf{w}\|_2^2 \\ \text{s.t.} \quad & \text{Prob}(\mathbf{X}_1 \in \mathcal{H}_2) \leq \eta_1 \\ & \text{Prob}(\mathbf{X}_2 \in \mathcal{H}_1) \leq \eta_2 \\ (3.12) \quad & \mathbf{X}_1 \sim (\mu_1, \Sigma_1) \quad \mathbf{X}_2 \sim (\mu_2, \Sigma_2) \end{aligned}$$

The constraints in the above formulation can be rewritten as deterministic constraints using the following multivariate generalization of Chebyshev-Cantelli inequality (Marshall & Olkin, 1960).

THEOREM 3.1. *Let \mathbf{X} be an n dimensional random vector. The mean and covariance of \mathbf{X} be $\mu \in \mathbb{R}^n$ and $\Sigma \in \mathbb{R}^{n \times n}$. Given $\mathbf{w} \in \mathbb{R}^n, \mathbf{w} \neq 0$ and $b \in \mathbb{R}$, let*

$$\mathcal{H}(\mathbf{w}, b) = \{\mathbf{z} | \mathbf{w}^\top \mathbf{z} < b, \mathbf{z} \in \mathbb{R}^n\}$$

be a half space. Then

$$(3.13) \quad \text{Prob}(\mathbf{X} \in \mathcal{H}) \geq \frac{s^2}{s^2 + \mathbf{w}^\top \Sigma \mathbf{w}}$$

where $s = (b - \mathbf{w}^\top \mu)_+, (x)_+ = \max(x, 0)$.

Applying theorem 3.1 with $\mathbf{X} = \mathbf{X}_1$ and $\mathcal{H} = \mathcal{H}_1$ (see also (Lanckriet et al., 2003)), the constraint for class 1 can be handled by setting

$$\text{Prob}(\mathbf{X}_1 \in \mathcal{H}_2) \leq \frac{\mathbf{w}^\top \Sigma_1 \mathbf{w}}{(\mathbf{w}^\top \mu_1 - b)_+^2 + \mathbf{w}^\top \Sigma_1 \mathbf{w}} \leq \eta_1$$

which result in two constraints

$$\mathbf{w}^\top \mu_1 - b \geq \sqrt{\frac{1 - \eta_1}{\eta_1}} \sqrt{\mathbf{w}^\top \Sigma_1 \mathbf{w}}, \quad \mathbf{w}^\top \mu_1 - b \geq 0$$

Similarly applying theorem 3.1 to the other constraint, two more constraints are obtained. Note that the constraints are positively homogeneous. That is, if \mathbf{w}, b satisfy the constraints then $c\mathbf{w}, cb$ also satisfy the constraints, for any positive c . To deal with this extra degree of freedom, one can impose the constraint that the classifier should separate the means even if $\eta_i = 1$. This can be imposed by requiring that $\mathbf{w}^\top \mu_1 - b \geq 1$ and $b - \mathbf{w}^\top \mu_2 \geq 1$. Let

$$\kappa_1 = \sqrt{\frac{1 - \eta_1}{\eta_1}}, \quad \kappa_2 = \sqrt{\frac{1 - \eta_2}{\eta_2}}$$

The problem (3.12) can now be stated as a deterministic optimization problem.

$$\begin{aligned} \min_{\mathbf{w}, b} \quad & \frac{1}{2} \|\mathbf{w}\|_2^2 \\ \text{s.t.} \quad & \mathbf{w}^\top \mu_1 - b \geq 1 + \kappa_1 \sqrt{\mathbf{w}^\top \Sigma_1 \mathbf{w}} \\ (3.14) \quad & b - \mathbf{w}^\top \mu_2 \geq 1 + \kappa_2 \sqrt{\mathbf{w}^\top \Sigma_2 \mathbf{w}} \end{aligned}$$

Since both the matrices Σ_1 and Σ_2 are symmetric positive semi-definite, there exist square matrices \mathbf{C}_1 and \mathbf{C}_2 such that $\Sigma_1 = \mathbf{C}_1 \mathbf{C}_1^\top$ and $\Sigma_2 = \mathbf{C}_2 \mathbf{C}_2^\top$. Now, (3.14) can be written as:

$$\begin{aligned} \min_{\mathbf{w}, b} \quad & \frac{1}{2} \|\mathbf{w}\|_2^2 \\ \text{s.t.} \quad & \mathbf{w}^\top \mu_1 - b \geq 1 + \kappa_1 \|\mathbf{C}_1^\top \mathbf{w}\|_2 \\ (3.15) \quad & b - \mathbf{w}^\top \mu_2 \geq 1 + \kappa_2 \|\mathbf{C}_2^\top \mathbf{w}\|_2 \end{aligned}$$

Clearly, the optimization problem (3.15) is feasible whenever $\mu_1 \neq \mu_2$. This is because one can choose $\eta_1 = \eta_2 = 1$, in which case the constraints in (3.15) imply that the hyperplane $\mathbf{w}^\top \mathbf{x} - b = 0$ must separate the means μ_1, μ_2 . Thus whenever the means are not coinciding, the problem (3.15) can be made feasible by choosing

appropriate values for η_1, η_2 . Note that the formulation is a convex programming problem, with a strict convex quadratic objective function (Hessian is the identity matrix). Hence, whenever feasible, the problem has a unique global solution (Fletcher, 1989). The non-linear constraints are called Second Order Cone (SOC) constraints (2.5). The formulation (3.15) can be written in the following standard SOCP form, by noting that minimizing $\frac{1}{2}\|\mathbf{w}\|_2^2$ is equivalent to minimizing $\|\mathbf{w}\|_2$.

$$(3.16) \quad \begin{aligned} \min_{\mathbf{w}, b, t} \quad & t \\ \text{s.t.} \quad & t \geq \|\mathbf{w}\|_2 \\ & \mathbf{w}^\top \mu_1 - b \geq 1 + \kappa_1 \|\mathbf{C}_1^\top \mathbf{w}\|_2 \\ & b - \mathbf{w}^\top \mu_2 \geq 1 + \kappa_2 \|\mathbf{C}_2^\top \mathbf{w}\|_2 \end{aligned}$$

SOCP problems can be efficiently solved by interior point methods for convex non-linear optimization (Nesterov & Nemirovskii, 1993). As a special case of convex non-linear optimization, SOCPs have gained much attention in recent times. For a discussion of further efficient algorithms and applications of SOCP see (Lobo et al., 1998). Once the formulation is solved for \mathbf{w} and b , the decision function given in (2.1) can be used to classify a new data point \mathbf{x} .

3.1 The Dual Formulation and Geometric Interpretation The constraints in (3.15) have an elegant geometric interpretation. In order to see this, consider the following problem. Suppose

$$(3.17) \quad B(\mu, \mathbf{C}, \kappa) = \{\mathbf{x} | \mathbf{x} = \mu - \kappa \mathbf{C} \mathbf{u}, \|\mathbf{u}\|_2 \leq 1\}$$

represents an ellipsoid centered at μ , whose shape is determined by \mathbf{C} and size by κ . We also assume that \mathbf{C} is square full-rank, in which case

$$(3.18) \quad B(\mu, \mathbf{C}, \kappa) = \{\mathbf{x} | (\mathbf{x} - \mu)^\top \Sigma^{-1} (\mathbf{x} - \mu) \leq \kappa^2\},$$

where $\Sigma = \mathbf{C} \mathbf{C}^\top$.

Now suppose we wish to put a constraint that all points in this ellipsoid must lie on the positive half-space of the hyperplane $\mathbf{w}^\top \mathbf{x} - b = 1$ (we assume that the given hyperplane does not intersect the ellipsoid). More formally the constraint can be written as:

$$(3.19) \quad \mathbf{w}^\top \mathbf{x} - b \geq 1 \quad \forall \mathbf{x} \in B(\mu, \mathbf{C}, \kappa)$$

Though this is a set of infinite constraints, one can satisfy them by doing the following: find the point that lies in the ellipsoid which is nearest to the hyperplane and then put a constraint that this point must lie on the positive half-space of $\mathbf{w}^\top \mathbf{x} - b = 1$. Now finding the point nearest (\mathbf{x}^*) to the hyperplane is easy because of the special form of the set $B(\mu, \mathbf{C}, \kappa)$. First of all this

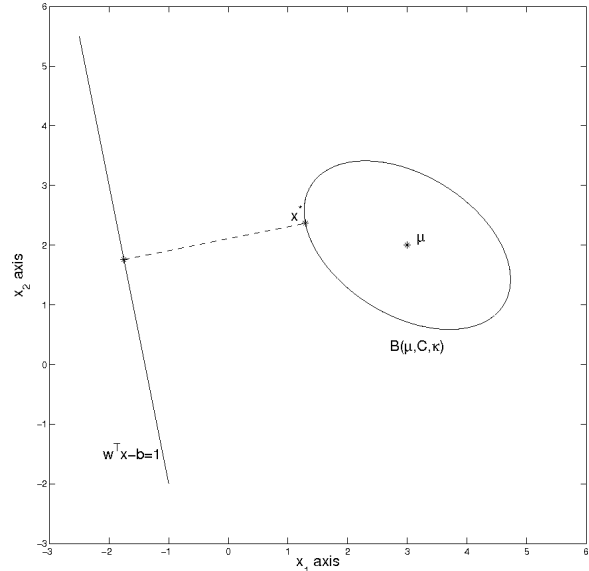


Figure 2: Illustration showing the geometric intuition behind the constraints of the proposed formulation

point must lie on the boundary of the ellipsoid since we assumed that the hyperplane does not intersect the ellipsoid. Next, the direction of normal at \mathbf{x}^* for the ellipsoid must be opposite to \mathbf{w} (see figure 2). In other words:

$$(3.20) \quad 2\Sigma^{-1}\mathbf{x}^* - 2\Sigma^{-1}\mu = \alpha \mathbf{w}$$

where α is some negative constant. Now substituting this into the boundary of ellipsoid in (3.18), one obtains the value

$$\alpha = \frac{-2\kappa}{\|\mathbf{C}^\top \mathbf{w}\|_2}$$

Using this value of α and (3.20), one can get the value of \mathbf{x}^* as

$$\mathbf{x}^* = \mu - \frac{\kappa \Sigma \mathbf{w}}{\|\mathbf{C}^\top \mathbf{w}\|_2}$$

As told earlier, it is enough to put the constraint that $\mathbf{w}^\top \mathbf{x}^* - b \geq 1$ in order to satisfy the infinite constraints in (3.19). In other words, $\mathbf{w}^\top \mu - b \geq 1 + \kappa \|\mathbf{C}^\top \mathbf{w}\|_2$, which is similar in form as the constraints in the proposed formulation (3.15).

Thus the geometrical interpretation of the proposed formulation is to find a maximum margin hyperplane that separates ellipsoids whose centers are the means, shape is parameterized by the covariance matrices of the class conditional densities and the size by the parameters κ_1 and κ_2 (see figure 3). In the following text we derive this more rigorously using the Duality theory.

Using the dual norm characterization

$$\|\mathbf{v}\|_2 = \sup_{\|\mathbf{u}\|_2 \leq 1} \mathbf{u}^\top \mathbf{v},$$

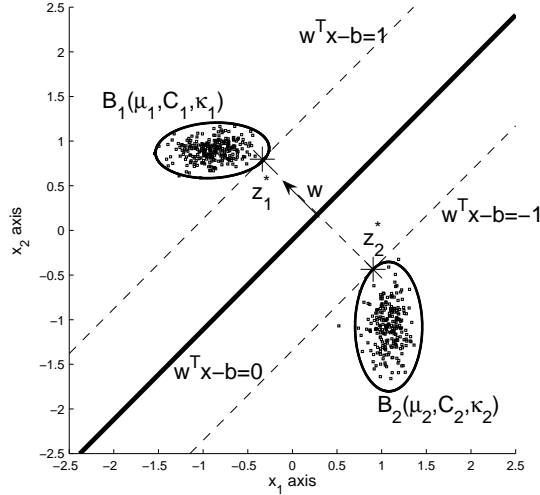


Figure 3: Illustration showing the geometric interpretation of the proposed formulation

the formulation (3.15) can be rewritten as:

$$\begin{aligned} \min_{\mathbf{w}, b, \mathbf{u}_1, \mathbf{u}_2} \quad & \frac{1}{2} \|\mathbf{w}\|_2^2 \\ \text{s.t.} \quad & \mathbf{w}^\top \mu_1 - b \geq 1 + \kappa_1 \mathbf{u}_1^\top \mathbf{C}_1^\top \mathbf{w}, \\ & b - \mathbf{w}^\top \mu_2 \geq 1 + \kappa_2 \mathbf{u}_2^\top \mathbf{C}_2^\top \mathbf{w}, \\ & \|\mathbf{u}_1\|_2 \leq 1, \|\mathbf{u}_2\|_2 \leq 1 \end{aligned}$$

Then the Lagrangian of this problem is given by $\mathcal{L}(\mathbf{w}, b, \lambda_1, \lambda_2, \mathbf{u}_1, \mathbf{u}_2) \equiv$

$$\begin{aligned} \frac{1}{2} \|\mathbf{w}\|_2^2 & - \lambda_1 (\mathbf{w}^\top \mu_1 - b - 1 - \kappa_1 \mathbf{u}_1^\top \mathbf{C}_1^\top \mathbf{w}) \\ & - \lambda_2 (b - \mathbf{w}^\top \mu_2 - 1 - \kappa_2 \mathbf{u}_2^\top \mathbf{C}_2^\top \mathbf{w}) \end{aligned}$$

with the constraints $\|\mathbf{u}_1\|_2 \leq 1$, $\|\mathbf{u}_2\|_2 \leq 1$, $\lambda_1 \geq 0$ and $\lambda_2 \geq 0$. At optimality, from Karush-Kuhn-Tucker (KKT) conditions (Fletcher, 1989), we have $\frac{\partial \mathcal{L}}{\partial b} = 0$ which implies that $\lambda_1 = \lambda_2 = \lambda$, where $\lambda \geq 0$ is a Lagrange variable. The optimal \mathbf{w} satisfies $\nabla_{\mathbf{w}} \mathcal{L} = 0$ giving

$$(3.21) \quad \mathbf{w} = \lambda(\mu_1 - \kappa_1 \mathbf{C}_1 \mathbf{u}_1 - \mu_2 - \kappa_2 \mathbf{C}_2 \mathbf{u}_2)$$

The dual formulation is obtained by maximizing \mathcal{L} with respect to the dual variables $\lambda \geq 0$, \mathbf{u}_1 and \mathbf{u}_2 , subject to the constraints $\frac{\partial \mathcal{L}}{\partial b} = 0$, $\nabla_{\mathbf{w}} \mathcal{L} = 0$ and the vectors \mathbf{u}_1 and \mathbf{u}_2 lying in a unit ball. This gives

$$\begin{aligned} \max_{\lambda, \mathbf{u}_1, \mathbf{u}_2} \quad & -\frac{1}{2} \lambda^2 \|\mathbf{z}_1 - \mathbf{z}_2\|_2^2 + 2\lambda \\ \mathbf{z}_1 = \mu_1 - \kappa_1 \mathbf{C}_1 \mathbf{u}_1, \quad & \mathbf{z}_2 = \mu_2 + \kappa_2 \mathbf{C}_2 \mathbf{u}_2 \\ \|\mathbf{u}_1\|_2 \leq 1, \quad & \|\mathbf{u}_2\|_2 \leq 1, \quad \lambda \geq 0 \end{aligned}$$

The objective is maximized when

$$(3.22) \quad \lambda = \frac{2}{\|\mathbf{z}_1 - \mathbf{z}_2\|_2^2},$$

and the maximized value is $\frac{2}{\|\mathbf{z}_1 - \mathbf{z}_2\|_2^2}$. Noting that maximizing such an objective is equivalent to minimizing its reciprocal, the dual can be stated as follows:

$$(3.23) \quad \min_{\mathbf{u}_1, \mathbf{u}_2} \|\mathbf{z}_1 - \mathbf{z}_2\|_2^2$$

$$\mathbf{z}_1 \in B_1(\mu_1, \mathbf{C}_1, \kappa_1), \quad \mathbf{z}_2 \in B_2(\mu_2, \mathbf{C}_2, \kappa_2)$$

where,

$$B_i(\mu_i, \mathbf{C}_i, \kappa_i) = \{\mathbf{z}_i | \mathbf{z}_i = \mu_i - \kappa_i \mathbf{C}_i \mathbf{u}_i, \|\mathbf{u}_i\|_2 \leq 1\}$$

The above optimization problem has a very elegant geometric interpretation. The sets $B_1(\mu_1, \mathbf{C}_1, \kappa_1)$ and $B_2(\mu_2, \mathbf{C}_2, \kappa_2)$ are ellipsoids centered at μ_1 and μ_2 and the parameterized by the matrices \mathbf{C}_1 and \mathbf{C}_2 respectively. The dual optimization problem can be thus seen as finding the minimum distance between the two ellipsoids. The optimal \mathbf{z}_1^* and \mathbf{z}_2^* can be used to compute λ by (3.22) and consequently using (3.21) one can find

$$(3.24) \quad \mathbf{w} = 2 \frac{\mathbf{z}_1^* - \mathbf{z}_2^*}{\|\mathbf{z}_1^* - \mathbf{z}_2^*\|_2^2}$$

The KKT conditions of the dual can be summarized as follows

$$(3.25) \quad \begin{aligned} -\kappa_1 \mathbf{C}_1^\top (\mathbf{z}_1 - \mathbf{z}_2) + \gamma_1 \mathbf{u}_1 &= 0, \quad \gamma_1 (\|\mathbf{u}_1\|_2 - 1) = 0, \\ -\kappa_2 \mathbf{C}_2^\top (\mathbf{z}_1 - \mathbf{z}_2) + \gamma_2 \mathbf{u}_2 &= 0, \quad \gamma_2 (\|\mathbf{u}_2\|_2 - 1) = 0, \\ \|\mathbf{u}_1\|_2 \leq 1, \quad \|\mathbf{u}_2\|_2 \leq 1, \quad & \gamma_1 \geq 0, \gamma_2 \geq 0 \end{aligned}$$

Thus, at optimality, $\mathbf{C}_1^\top (\mathbf{z}_1 - \mathbf{z}_2)$ is parallel to \mathbf{u}_1 and $\mathbf{C}_2^\top (\mathbf{z}_1 - \mathbf{z}_2)$ is parallel to \mathbf{u}_2 . Define

$$(3.26) \quad \theta(\mathbf{u}, \mathbf{v}) = \arccos \left(\frac{\mathbf{u}^\top \mathbf{v}}{\|\mathbf{u}\|_2 \|\mathbf{v}\|_2} \right)$$

Then,

$$\theta(\mathbf{C}_1^\top (\mathbf{z}_1 - \mathbf{z}_2), \mathbf{u}_1) = \theta(\mathbf{C}_2^\top (\mathbf{z}_1 - \mathbf{z}_2), \mathbf{u}_2) = 0$$

at optimality. In this paper, we are interested in the case of non-intersecting ellipsoids, in which case the original problem (3.15) is feasible. In such a scenario, since B_1 and B_2 are disjoint, $\mathbf{z}_1^* - \mathbf{z}_2^*$ is not zero. Furthermore, if $\mathbf{z}_1^* - \mathbf{z}_2^*$ does not lie in the null space of \mathbf{C}_1^\top and \mathbf{C}_2^\top , the Lagrange variables γ_1 and γ_2 are strictly positive, which gives the conditions $\|\mathbf{u}_1\|_2 = 1$ and $\|\mathbf{u}_2\|_2 = 1$ at optimality. This implies that the optimal \mathbf{z}_1^* and \mathbf{z}_2^* are at the boundary of the ellipsoids B_1 and B_2 respectively. By (3.22), we have $\lambda > 0$, which implies

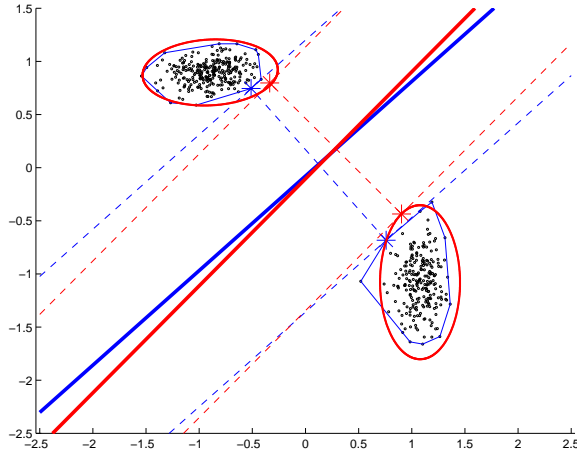


Figure 4: Illustration comparing the classifiers obtained with SVM and present method. The SVM solution is shown in blue, whereas that of the present method is shown in red ($\eta_1 = \eta_2 = 0.1$).

that both the constraints in (3.15) are active, giving two conditions $\mathbf{w}^\top \mathbf{z}_1^* - b = 1$ and $\mathbf{w}^\top \mathbf{z}_2^* - b = -1$. This geometrically means that the hyperplanes $\mathbf{w}^\top \mathbf{x} - b = 1$ and $\mathbf{w}^\top \mathbf{x} - b = -1$ are tangents to the ellipsoids B_1 and B_2 respectively. Using any of these conditions, one can compute b and more precisely

$$(3.27) \quad b = 2 \frac{\mathbf{z}_1^\top (\mathbf{z}_1 - \mathbf{z}_2)}{\|\mathbf{z}_1 - \mathbf{z}_2\|_2^2} - 1$$

It is interesting to note the analogy between the geometrical interpretation of the SVM dual (Bennett & Bredensteiner, 2000) and that of the present formulation. In case of SVM, the dual turns out to be the problem of finding distance between two convex hulls, whereas in the present case, the dual turns out to be distance between ellipsoids. Figure 4 shows the optimal hyperplane as obtained with the formulation (3.15) and that with SVM on a synthetic dataset. In general, one can observe that if the training data has small number of noisy examples, then the convex hull solution is more effected than the ellipsoid solution. To circumvent this problem, the soft-margin SVMs are introduced. However they involve an additional regularization parameter C . The figure also confirms the equivalence of the primal (3.15) and dual (3.23).

3.2 An Iterative Algorithm for Solving the Dual The geometric insight presented in the previous section gives us a way of solving the formulation using a simple iterative scheme for finding the distance between two ellipsoids. (Lin & Han, 2002) provide an iterative, provably convergent algorithm for finding the minimum

distance between two ellipsoids. We provide here the application of the same algorithm to our case. Suppose the matrices Σ_i are positive definite, in which case \mathbf{C}_i can be chosen to be square matrices of full rank. Then, the equation of the ellipsoid $B_i(\mu_i, \mathbf{C}_i, \kappa_i)$ in the standard form is

$$q_i(\mathbf{z}_i) \equiv \frac{1}{2} \mathbf{z}_i^\top \mathbf{A}_i \mathbf{z}_i + \mathbf{b}_i^\top \mathbf{z}_i + \alpha_i \leq 0,$$

where $\mathbf{A}_i = 2\Sigma_i^{-1}$, $\mathbf{b}_i^\top = -2\mu_i^\top \Sigma_i^{-1}$ and $\alpha_i = \mu_i^\top \Sigma_i^{-1} \mu_i - \kappa_i^2$. Once this is done, the following iterative algorithm can be used to solve the dual:

Input μ_i, Σ_i and κ_i

Output \mathbf{z}_1^* and \mathbf{z}_2^*

Initialization Compute the following:

1. $\mathbf{A}_i, \mathbf{b}_i$ and α_i
2. $\mathbf{c}_1 = \mu_1$ and $\mathbf{c}_2 = \mu_2$ — two interior points in the ellipsoids

General Steps At the k^{th} iteration, having an interior point \mathbf{c}_1 of B_1 and \mathbf{c}_2 of B_2 ,

1. Find points of intersection of line segment joining \mathbf{c}_1 and \mathbf{c}_2 with the ellipsoids:
 - (a) Represent the line segment using $(1-t)\mathbf{c}_1 + t\mathbf{c}_2$, $0 \leq t \leq 1$
 - (b) Solve for $q_i((1-t_i)\mathbf{c}_1 + t_i\mathbf{c}_2) = 0$, to get t_i , $i = 1, 2$:

$$\begin{aligned} & \frac{1}{2} t_i^2 \{(\mathbf{c}_1 - \mathbf{c}_2)^\top \mathbf{A}_i (\mathbf{c}_1 - \mathbf{c}_2)\} - \\ & t_i \{(\mathbf{c}_1^\top - \mathbf{c}_2^\top)(\mathbf{A}_i \mathbf{c}_1 + \mathbf{b}_i)\} + \\ & \left\{ \frac{1}{2} \mathbf{c}_1^\top \mathbf{A}_i \mathbf{c}_1 + \mathbf{b}_i^\top \mathbf{c}_1 + \alpha_i \right\} = 0 \end{aligned}$$

- (c) Solve for roots of quadratic, such that $0 \leq t_i \leq 1$ and calculate $\mathbf{z}_i^k = (1-t_i)\mathbf{c}_1 + t_i\mathbf{c}_2$, the points of intersection
- (d) If $t_1 > t_2$, then the problem is infeasible. Terminate giving an error.
2. If the line segment joining the centers is normal at the points \mathbf{z}_1^k and \mathbf{z}_2^k , then optimal achieved:
 - (a) Compute $\theta_1 = \theta(\mathbf{z}_2^k - \mathbf{z}_1^k, \mathbf{A}_1 \mathbf{z}_1^k + \mathbf{b}_1)$ and $\theta_2 = \theta(\mathbf{z}_1^k - \mathbf{z}_2^k, \mathbf{A}_2 \mathbf{z}_2^k + \mathbf{b}_2)$
 - (b) If $\theta_1 = \theta_2 = 0$, then terminate indicating convergence to optimality
3. Else, compute new interior points $\bar{\mathbf{c}}_1$ and $\bar{\mathbf{c}}_2$, as centers of spheres that entirely lie inside the corresponding ellipsoids and touch the ellipsoids at \mathbf{z}_1^k and \mathbf{z}_2^k :

- (a) Use $\bar{\mathbf{c}}_i = \mathbf{z}_i^k - \delta_i(\mathbf{A}_i \mathbf{z}_i^k + \mathbf{b}_i)$
(b) $\delta_i = \frac{1}{\|\mathbf{A}_i\|_2}$

Note that in the algorithm, the standard form of ellipsoids is used. Hence, \mathbf{C}_i need not be calculated explicitly. Also, for all values of δ_i , the spheres with center $\bar{\mathbf{c}}_i$ and radius δ_i touch the ellipsoids at \mathbf{z}_i^k . But only for values of $\delta_i \leq \frac{1}{\|\mathbf{A}_i\|_2}$, the spheres will entirely lie inside the ellipsoids. Hence, we choose $\delta_i = \frac{1}{\|\mathbf{A}_i\|_2}$ to get maximum possible iterative step size. The algorithm given above will converge to the optimal solution of (3.23). The outline of the proof of convergence is provided here (refer (Lin & Han, 2002) for details), assuming that the ellipsoids are separated. The KKT optimality conditions for (3.23) are (in terms of the ellipsoids in standard form):

$$\begin{aligned} z_1^* &\in \Omega(B_1), & z_2^* &\in \Omega(B_2), \\ \theta(z_1^* - z_1^*, \mathbf{A}_1 z_1^* + b_1) &= 0, & \theta(z_2^* - z_2^*, \mathbf{A}_2 z_2^* + b_2) &= 0, \end{aligned}$$

where, $\Omega(B_i)$ represents the boundary of the ellipsoid B_i . These optimality conditions say that the optimal (z_1^*, z_2^*) lie on the boundaries of corresponding ellipsoids and the line segments joining the optimal points are the normals at those points. Since the problem is convex, KKT conditions are necessary and sufficient. Note that these conditions are equivalent to those given in (3.25). This argument justifies step 2 of the above algorithm. In case of finding distance between two spheres, one can get the optimal points as the points of intersection of the line segment joining the centers with the spheres. Thus, this algorithm can be viewed as if the two ellipsoids were being iteratively approximated locally by spheres. Using the notation given in the algorithm,

$$\|\bar{\mathbf{c}}_1 - \bar{\mathbf{c}}_2\| \geq \delta_1 + \delta_2 + \|z_1^{k+1} - z_2^{k+1}\|$$

Using triangle inequality, we have

$$\begin{aligned} \|\bar{\mathbf{c}}_1 - \bar{\mathbf{c}}_2\| &\leq \|\bar{\mathbf{c}}_1 - z_1^k\| + \|z_1^k - z_2^k\| + \|z_2^k - \bar{\mathbf{c}}_2\| \\ &\leq \delta_1 + \delta_2 + \|z_1^k - z_2^k\| \end{aligned}$$

Using these inequalities, we have the following monotonicity property that every step:

$$\|z_1^k - z_2^k\| \geq \|z_1^{k+1} - z_2^{k+1}\|$$

Therefore, the sequence of distances $\{\|z_1^k - z_2^k\|\}$ is monotone and hence converges. Now one can also prove that for such a sequence,

$$\lim_{k \rightarrow \infty} \theta(z_1^* - z_1^*, \mathbf{A}_1 z_1^* + b_1) = 0,$$

$$\lim_{k \rightarrow \infty} \theta(z_2^* - z_2^*, \mathbf{A}_2 z_2^* + b_2) = 0,$$

proving that (z_1^k, z_2^k) converges to (z_1^*, z_2^*) .

Note that at every step of iteration, two one-dimensional quadratic equations are solved. The SMO algorithm (Platt, 1999), which is used to solve the dual of the SVM, also solves a one-dimensional optimization problem that has a closed form solution. Thus, the iterative algorithm of solving the dual (3.23) and the SMO algorithm are analogous. However, the initialization cost is high for the above iterative algorithm, due to inversion of matrices, which is of $O(n^3)$ time complexity (n is the dimension of \mathbf{z}_i). In addition to this, at each step of iteration, the coefficients of the two quadratic expressions need to be computed. This is of $O(n^2)$ time complexity.

3.3 Non-linear Classifiers The formulation (3.14) gives a linear classifier and hence cannot deal with non-linearly separable data. In the following text, we extend the formulation to such data. Let \mathbf{T}_1 be a $n \times m_1$, data matrix for one class, say with label $y = 1$, where each column of \mathbf{T}_1 is a datapoint with positive label. Similarly, let \mathbf{T}_2 be a $n \times m_2$ data matrix for the other class having the label $y = -1$. More formally let,

$$\mathbf{T}_1 = [\mathbf{x}_{11}, \dots, \mathbf{x}_{1m_1}], \mathbf{T}_2 = [\mathbf{x}_{21}, \dots, \mathbf{x}_{2m_2}]$$

Let $[\mathbf{M}_1, \mathbf{M}_2]$ and $[\mathbf{M}_1; \mathbf{M}_2]$ represent the horizontal and vertical concatenation of the matrices \mathbf{M}_1 and \mathbf{M}_2 respectively. The empirical estimates of the mean and covariance are given as:

$$\mu_1 = \frac{1}{m_1} \mathbf{T}_1 \mathbf{e}_1, \mu_2 = \frac{1}{m_2} \mathbf{T}_2 \mathbf{e}_2,$$

$$\begin{aligned} \Sigma_1 &= \frac{1}{m_1} (\mathbf{T}_1 - \mu_1 \mathbf{e}_1^\top) (\mathbf{T}_1^\top - \mathbf{e}_1 \mu_1^\top) \\ &= \frac{1}{m_1} \mathbf{T}_1 (\mathbf{I}_1 - \frac{\mathbf{e}_1 \mathbf{e}_1^\top}{m_1})^2 \mathbf{T}_1^\top, \end{aligned}$$

and similarly

$$\Sigma_2 = \frac{1}{m_2} \mathbf{T}_2 (\mathbf{I}_2 - \frac{\mathbf{e}_2 \mathbf{e}_2^\top}{m_2})^2 \mathbf{T}_2^\top$$

where, \mathbf{e}_i is a vector of ones of dimension m_i and \mathbf{I}_i is an identity matrix of dimensions $m_i \times m_i$. \mathbf{w} , being a vector in n dimensional space, can be written as a linear combination of the training data points and some other points in \mathbb{R}^n , which are orthogonal to the training data points, such that together the set of data points span the whole of \mathbb{R}^n .

Mathematically, $\mathbf{w} = [\mathbf{T}_1, \mathbf{T}_2] \mathbf{s} + \mathbf{M} \mathbf{r}$ where \mathbf{M} is a matrix with its columns as vectors orthogonal to training data points and \mathbf{s}, \mathbf{r} are vectors of combining coefficients. The columns of $\mathbf{T}_1, \mathbf{T}_2$ and \mathbf{M} together

Table 1: Results on benchmark datasets, comparing the performance of the four algorithms.

Dataset	η_1	η_2	NL-SOCP % err		NL-ITER % err		L-SOCP % err		L-ITER % err	
			+ve	-ve	+ve	-ve	+ve	-ve	+ve	-ve
Breast Cancer $m = 569, n = 30,$ $m_1 = 212, m_2 = 357,$ $\sigma = 0.032$	0.9	0.3	12.74	00.56	12.74	00.56	16.98	00.00	16.04	00.84
	0.7	0.3	10.85	01.12	10.85	01.12	13.68	00.00	13.21	01.40
	0.5	0.3	04.72	01.96	04.72	01.96	05.19	00.84	07.55	02.24
	0.3	0.3	03.30	02.24	03.30	02.24	03.77	02.80	03.77	03.08
	0.1	0.3	02.36	04.20	02.36	04.48	×	×	×	×
Ring Norm $m = 400, n = 2,$ $m_1 = 209, m_2 = 191,$ $\sigma = 3$	0.9	0.7	30.14	31.94	30.14	31.94	×	×	×	×
	0.7	0.7	20.57	36.65	20.57	36.65	×	×	×	×
	0.5	0.7	12.44	41.89	12.92	41.36	×	×	×	×
	0.3	0.7	10.05	46.07	10.53	45.03	×	×	×	×
	0.1	0.7	07.66	47.12	07.66	48.17	×	×	×	×
Two Norm $m = 500, n = 2,$ $m_1 = 266, m_2 = 234,$ $\sigma = 20$	0.9	0.3	10.15	01.28	10.53	01.28	09.02	00.43	09.02	00.43
	0.7	0.3	06.39	01.71	07.52	01.71	06.77	00.43	06.77	00.43
	0.5	0.3	05.26	02.56	06.02	02.56	04.51	01.28	04.51	01.28
	0.3	0.3	05.64	04.27	05.64	04.27	03.38	01.71	03.38	01.71
	0.1	0.3	07.52	05.98	05.26	07.26	×	×	×	×
Heart Disease $m = 297, n = 13,$ $m_1 = 137, m_2 = 160,$ $\sigma = 0.16$	0.9	0.9	14.60	22.50	14.60	22.50	18.99	14.38	18.99	14.38
	0.7	0.9	13.14	27.50	13.14	28.13	17.52	17.50	17.52	17.50
	0.5	0.9	11.68	32.50	11.68	32.50	13.14	21.88	13.14	21.88
	0.3	0.9	10.95	30.00	11.69	34.38	10.22	36.25	10.22	36.25
	0.1	0.9	10.95	30.00	10.95	36.25	×	×	×	×

span entire \mathbb{R}^n . Now, the terms involving \mathbf{w} in the constraints of (3.14) can be written as

$$\mathbf{w}^\top \mu_1 = \mathbf{s}^\top \mathbf{g}_1, \quad \mathbf{g}_1 = \left[\frac{\mathbf{K}_{11} \mathbf{e}_1}{m_1}; \frac{\mathbf{K}_{21} \mathbf{e}_1}{m_1} \right],$$

$$\mathbf{w}^\top \mu_2 = \mathbf{s}^\top \mathbf{g}_2, \quad \mathbf{g}_2 = \left[\frac{\mathbf{K}_{12} \mathbf{e}_2}{m_2}; \frac{\mathbf{K}_{22} \mathbf{e}_2}{m_2} \right],$$

$$\mathbf{w}^\top \Sigma_1 \mathbf{w} = \mathbf{s}^\top \mathbf{G}_1 \mathbf{s},$$

$$\mathbf{G}_1 = \frac{1}{m_1} [\mathbf{K}_{11}; \mathbf{K}_{21}] (\mathbf{I}_1 - \frac{\mathbf{e}_1 \mathbf{e}_1^\top}{m_1})^2 [\mathbf{K}_{11}, \mathbf{K}_{12}]$$

and

$$\mathbf{w}^\top \Sigma_2 \mathbf{w} = \mathbf{s}^\top \mathbf{G}_2 \mathbf{s},$$

$$\mathbf{G}_2 = \frac{1}{m_2} [\mathbf{K}_{12}; \mathbf{K}_{22}] (\mathbf{I}_2 - \frac{\mathbf{e}_2 \mathbf{e}_2^\top}{m_2})^2 [\mathbf{K}_{21}, \mathbf{K}_{22}]$$

where the matrices $\mathbf{K}_{11} = \mathbf{T}_1^\top \mathbf{T}_1$, $\mathbf{K}_{12} = \mathbf{T}_1^\top \mathbf{T}_2$, $\mathbf{K}_{22} = \mathbf{T}_2^\top \mathbf{T}_2$ consist of elements which are dot products of data points, more precisely the i^{th} row j^{th} column entry for the matrix $\mathbf{K}_{12}(i, j) = \mathbf{x}_{1i}^\top \mathbf{x}_{2j}$. Note that the constraints are independent of the matrix \mathbf{M} and the objective to be minimized is $\frac{1}{2} \|\mathbf{w}\|_2^2$. Hence, the entries in \mathbf{r} for the optimal \mathbf{w} must be all 0. In other words, at optimality, $\mathbf{w} = [\mathbf{T}_1, \mathbf{T}_2] \mathbf{s}$. Thus, the formulation (3.14)

can be written as:

$$\begin{aligned} \min_{\mathbf{s}, b} \quad & \frac{1}{2} \mathbf{s}^\top \mathbf{K} \mathbf{s} \\ \text{s.t.} \quad & \mathbf{s}^\top \mathbf{g}_1 - b \geq 1 + \kappa_1 \sqrt{\mathbf{s}^\top \mathbf{G}_1 \mathbf{s}}, \\ & b - \mathbf{s}^\top \mathbf{g}_2 \geq 1 + \kappa_2 \sqrt{\mathbf{s}^\top \mathbf{G}_2 \mathbf{s}} \end{aligned} \quad (3.28)$$

where, $\mathbf{K} = [\mathbf{K}_{11}, \mathbf{K}_{12}; \mathbf{K}_{21}, \mathbf{K}_{22}]$. Note that to solve the above problem, one needs to know only the dot products of training data points. Thus, one can solve the above problem in any feature space as long as the dot products in that space are available. One way of specifying dot product is by kernel functions satisfying positive definite conditions (Mercer, 1909). Assuming that we have such a kernel function $k : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$, the quantities $\mathbf{g}_1, \mathbf{g}_2, \mathbf{G}_1, \mathbf{G}_2$ and \mathbf{K} can be calculated. Suppose \mathbf{K} is positive definite, in which case write $\mathbf{K} = \mathbf{L}^\top \mathbf{L}$, where \mathbf{L} is a full rank square matrix. Now re-parameterize the formulation (3.28), in terms of a new variable $\mathbf{v} = \mathbf{L} \mathbf{s}$.

$$\begin{aligned} \min_{\mathbf{v}, b} \quad & \frac{1}{2} \|\mathbf{v}\|_2^2 \\ \text{s.t.} \quad & \mathbf{v}^\top \mathbf{h}_1 - b \geq 1 + \kappa_1 \sqrt{\mathbf{v}^\top \mathbf{H}_1 \mathbf{v}}, \\ & b - \mathbf{v}^\top \mathbf{h}_2 \geq 1 + \kappa_2 \sqrt{\mathbf{v}^\top \mathbf{H}_2 \mathbf{v}} \end{aligned} \quad (3.29)$$

where, $\mathbf{h}_i = \mathbf{L}^{-T} \mathbf{g}_i$ and $\mathbf{H}_i = \mathbf{L}^{-T} \mathbf{G}_i \mathbf{L}^{-1}$.

Note that the above formulation is similar to the original formulation (3.14). Again, \mathbf{H}_i , being positive

semi-definite, can be written as $\mathbf{H}_i = \mathbf{D}_i \mathbf{D}_i^\top$. (3.29) can be solved using interior point methods when cast into the following standard SOCP form.

$$(3.30) \quad \begin{aligned} \min_{\mathbf{v}, b, t} \quad & t \\ \text{s.t.} \quad & t \geq \|\mathbf{v}\|_2 \\ & \mathbf{v}^\top \mathbf{h}_1 - b \geq 1 + \kappa_1 \|\mathbf{D}_1^\top \mathbf{v}\|_2, \\ & b - \mathbf{v}^\top \mathbf{h}_2 \geq 1 + \kappa_2 \|\mathbf{D}_2^\top \mathbf{v}\|_2 \end{aligned}$$

Using the arguments presented in section 3.1, the dual of (3.29) is

$$(3.31) \quad \begin{aligned} \min_{\mathbf{u}_1, \mathbf{u}_2} \quad & \|\mathbf{z}_1 - \mathbf{z}_2\|_2^2 \\ \text{s.t.} \quad & \mathbf{z}_1 \in B_1(\mathbf{h}_1, \mathbf{D}_1, \kappa_1), \mathbf{z}_2 \in B_2(\mathbf{h}_2, \mathbf{D}_2, \kappa_2) \end{aligned}$$

and can be solved using the iterative geometric algorithm presented in section 3.2. Once the optimum value of \mathbf{v} and b are obtained either by solving the SOCP problem or by the iterative algorithm, one can classify a new data point \mathbf{x} using the following decision function.

$$(3.32) \quad f(\mathbf{x}) \equiv \text{sign}(\mathbf{w}^\top \mathbf{x} - b) = \text{sign}(\mathbf{s}^\top k(:, \mathbf{x}) - b)$$

where, $\mathbf{s} = \mathbf{L}^{-1} \mathbf{v}$ and $k(:, \mathbf{x})$ is the vector of kernel values of all training data points with the new data point \mathbf{x} .

Finally we conclude this section with a note on estimating the moments of class conditional densities. In practical experiments, it may well happen that the positive/negative data error rate computed on the test set is greater than η_1/η_2 . This seems to contradict the previous claim that η_1/η_2 is an upper bound on the misclassification error on positive/negative data. This is because the bounds are valid under the assumption that the moments of the class conditional densities are accurate. In practice, since we do not know the mean and covariance a priori, they need to be estimated from the data. The validity of the bound depends on how good the estimate is. This is an issue especially in the "small sample" case. One can use methods suggested in (Lanckriet et al., 2003), where the authors propose ways of introducing robustness into the classifiers by handling the uncertainties in moment estimations.

4 Experimental Results

The SOCP formulations for non-linear classifier (3.30) and for linear classifier (3.16) are solved using publicly available `SeDuMi` software (Sturm, 1999). Let **NL-SOCP** and **L-SOCP** denote these classifiers respectively. The dual problems for non-linear classifier (3.31) and for linear classifier (3.23) are solved using the iterative algorithm presented in section 3.2. Let **NL-ITER** and **L-ITER** denote these classifiers respectively. Recall that the iterative algorithm required that the matrices Σ_i and \mathbf{H}_i to be positive definite (section 3.2).

Table 2: Results on benchmark datasets, comparing the errors obtained with **L-SVM** and **L-SOCP**.

Dataset	Method	C_+/η_1	C_-/η_2	% err
PIMA	L-SVM	5.5	4.5	22.53
	L-SOCP	0.1	0.5	23.44
B. Cancer	L-SVM	5	5	5.1
	L-SOCP	0.76	0.76	2.99

Table 3: Results on benchmark datasets, comparing the risk obtained with **L-SVM** and **L-SOCP**.

Dataset	Method	C_+/η_1	C_-/η_2	risk
PIMA	L-SVM	13.333	6.667	255
	L-SOCP	0.085	0.515	256
B. Cancer	L-SVM	5	5	45
	L-SOCP	0.76	0.76	26

However, in practice, Σ_i or \mathbf{H}_i can be ill-conditioned. To handle such cases, regularization $\Sigma_i = \Sigma_i + \epsilon \mathbf{I}$ and $\mathbf{H}_i = \mathbf{H}_i + \epsilon \mathbf{I}$ has been used. ϵ being a small positive quantity, regularization does not effect the final classifier much. Gaussian kernel $k(\mathbf{x}_i, \mathbf{x}_j) = \exp\{-\sigma \|\mathbf{x}_i - \mathbf{x}_j\|_2^2\}$, with parameter σ , is used to evaluate dot products in the feature space.

The first set of experiments have been done to show that:

- By varying η_1 and η_2 different classifiers are obtained, whose false positive and false negative error rates vary accordingly.
- The classifiers **NL-SOCP**, **NL-ITER** are equivalent. Similarly **L-SOCP**, **L-ITER** are equivalent.
- If data is non-linearly separable, then non-linear classifiers perform better than the linear classifiers.

Table 1 shows the results of building the four classifiers on some benchmark datasets. The Breast Cancer and Heart Disease datasets have been acquired from UCI-ML repository (Blake & Merz, 1998). These datasets are unbalanced and the cost of misclassifying positive examples is higher than cost of misclassifying negative examples. The Two Norm and Ring Norm datasets have been generated using the standard dataset generation scripts got from Delve-Benchmark repository (available at <http://www.cs.toronto.edu/~delve/data/datasets.html>). The table shows the dimensions of each dataset and the value of σ used for non-linear classifiers. For each dataset, the % error on class 1 (+ve) and class -1 (-ve) data points obtained with all the 4 classifiers is shown, which is the 3-fold cross validation error averaged over 3 cross validation

experiments. The value ‘×’ in the table represents infeasibility of the problem. In order to show the nature of dependency of % error on the values of η_i used for each dataset, the value of η_2 is kept constant and η_1 is varied. Observe that, in general, as η_1 value is decreased the % +ve error decreases and % -ve error increases. This shows the potential of the proposed classifiers in classification applications with asymmetric costs for misclassification. The Ring Norm dataset is not linearly separable, in fact, $\mu_1 \approx \mu_2$. Hence, for all values of η_i shown, the classifiers **L-SOCP** and **L-ITER** fail. However, the classifiers **NL-SOCP** and **NL-ITER** work well.

The second set of experiments have been done to show that the proposed classifiers achieve accuracies and risks comparable to the state of the art classifiers, SVMs where C_+, C_- are varied (2.11). These are variants of the traditional SVMs where the term $C \sum_i \xi_i$ in the objective of the SVMs is replaced by $C_+ \sum_{y_i=1} \xi_i + C_- \sum_{y_i=-1} \xi_i$. Thus allowing for biased classification. We show experiments on two Datasets: Pima Indian Diabetes dataset ($m = 768, n = 8, m_1 = 268, m_2 = 500$) and Breast Cancer dataset from UCI-ML repository. These datasets are highly unbalanced, the cost of misclassifying the positive class is higher than the other and in general, linear classifiers work well on them. Table 2 summarizes the results that compare the cross validation error obtained with Linear SVMs (**L-SVM**) and the proposed classifier **L-SOCP**. The values of parameters are chosen to be the tuned set of parameters that gave the least cross validation error. Table 3 summarizes the results that compare the risk of **L-SOCP** and **L-SVM**. The risk shown in the table is computed assuming that the cost of misclassifying positive class is twice that of the other. Thus if e_+ and e_- are the cross validation errors, then the risk is $2e_+ + e_-$. Again we show the results for tuned set of parameters only. The results show that in the case of both datasets, the proposed classifier achieves performance comparable to that of SVMs.

5 Conclusions

We have presented a maximum margin classifier whose probability of misclassification, for each of the two classes, is bounded above by a specified value. Assuming that the training data is summarized by moments of class conditional densities, using Chebyshev-Cantelli inequality, the problem is cast as an SOCP. The dual problem of the original formulation turns out to be that of finding the distance between two ellipsoids. The optimal hyperplane is perpendicular to the line joining the minimum distant points on the ellipsoids. An iterative algorithm to solve the dual was presented. An extension of the original formulation for non-linear

classifiers using kernel methods, assuming empirical estimates of moments of class conditional densities, was presented. As in the linear classifier case, the non-linear extension can be solved by casting as an SOCP or by using the iterative algorithm. Experiments on some benchmark datasets were done to show the working of the classifiers. Experiments also confirm the equivalence of the primal and dual, in the case of both linear and non-linear classifiers.

In the future, we would like to explore fast and efficient algorithms for solving the problem of minimizing distance between ellipsoids. The theoretical benefits of the proposed classifier have potential to be exploited in real world applications, where false positive and false negative error rates that can be tolerated are specified.

Acknowledgments

The first author is supported by DST (Department of Science and Technology, Government of India) project DSTO/ECA/CB/660.

References

- Bach, F. R., Heckerman, D., & Horvitz, E. (2005). On the path to an ideal roc curve: considering cost asymmetry in learning classifiers. *Tenth International Workshop on Artificial Intelligence and Statistics (AISTATS)*.
- Bennett, K. P., & Bredensteiner, E. J. (2000). Duality and geometry in SVM classifiers. *Proceedings of the International Conference on Machine Learning* (pp. 57–64). San Francisco, California: Morgan Kaufmann Publishers.
- Blake, C., & Merz, C. (1998). UCI repository of machine learning databases. <http://www.ics.uci.edu/~mllearn/MLRepository.html>.
- Cardie, C., & Howe, N. (1997). Improving minority class prediction using case specific feature weights. *Proceedings of the 14th International Conference on Machine Learning* (pp. 57–65). Morgan Kaufmann.
- Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence and Research*, 16, 321–357.
- Fletcher, R. (1989). *Practical methods of optimization*. New York: John Wiley and Sons.
- Huang, K., Yang, H., King, I., Lyu, M., & Chan, L. (2004). Biased minimax probability machine for medical diagnosis. *8th International Symposium on Artificial Intelligence and Mathematics*.

- Keerthi, S. S., Shevade, S. K., Bhattacharyya, C., & Murthy, K. R. K. (2000). A fast iterative nearest point algorithm for support vector machine classifier design. *IEEE Transactions on Neural Networks*, *11*, 124–136.
- Kononenko, I. (2001). Machine learning for medical diagnosis: history, state of the art and perspective. *Artificial Intelligence in Medicine*, *23*, 89–109.
- Kubat, M., & Matwin, S. (1997). Addressing the curse of imbalanced training sets: one-sided selection. *Proceedings of the 14th International Conference on Machine Learning* (pp. 179–186). Morgan Kaufmann.
- Lanckriet, G. R., Ghaoui, L. E., Bhattacharyya, C., & Jordan, M. I. (2002). Minimax probability machine. *Advances in Neural Information Processing Systems*. MIT Press.
- Lanckriet, G. R., Ghaoui, L. E., Bhattacharyya, C., & Jordan, M. I. (2003). A robust minimax approach to classification. *Journal of Machine Learning Research*, *3*, 555–582.
- Lin, A., & Han, S.-P. (2002). On the distance between two ellipsoids. *SIAM Journal of Optimization*, *13*, 298–308.
- Lobo, M., Vandenberghe, L., Boyd, S., & Lebet, H. (1998). Applications of second-order cone programming. *Linear Algebra and its Applications*, *284*, 193–228.
- Marshall, A. W., & Olkin, I. (1960). Multivariate chebyshev inequalities. *Annals of Mathematical Statistics*, *31*, 1001–1014.
- Mercer, J. (1909). Functions of positive and negative type and their connection with the theory of integral equations. *Philosophical Transactions of the Royal Society, London, A* *209*, 415–446.
- Nesterov, Y., & Nemirovskii, A. (1993). *Interior point algorithms in convex programming*. No. 13 in Studies in Applied Mathematics. Philadelphia: SIAM.
- Platt, J. (1999). Fast training of support vector machines using sequential minimal optimization. *Advances in Kernel Methods—Support Vector Learning* (pp. 185–208). Cambridge, MA: MIT Press.
- Provost, F. (2000). Learning from imbalanced data sets. *Proceedings of the seventeenth National Conference on Artificial Intelligence (AAAI-2000)*.
- Sturm, J. (1999). Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, *11–12*, 625–653. Special issue on Interior Point Methods.
- Vapnik, V. (1998). *Statistical learning theory*. New York: John Wiley and Sons.