Learning Algorithms using Chance-Constrained Programs

A Thesis

Submitted For the Degree of

Doctor of Philosophy

in the Faculty of Engineering

by

Saketha Nath Jagarlapudi

Computer Science and Automation

Indian Institute of Science

BANGALORE – 560 012

DECEMBER 2007

Acknowledgements

I would like to express sincere gratitude and thanks to my adviser, Dr. Chiranjib Bhattacharyya. With his interesting thoughts and ideas, inspiring ideals and friendly nature, he made sure I was filled with enthusiasm and interest to do research all through my PhD. He was always approachable and spent ample time with me and all my lab members for discussions, though he had a busy schedule. I also thank Prof. M. N. Murty, Dr. Samy Bengio (Google Labs, USA) and Prof. Aharon Ben-Tal (Technion, Israel) for their help and co-operation.

I am greatly in debt to my parents, wife and other family members for supporting and encouraging me all through the PhD years. I thank all my lab members and friends, especially Karthik Raman, Sourangshu, Rashmin, Krishnan and Sivaramakrishnan, for their useful discussions and comments.

I thank the Department of Science and Technology, India, for supporting me financially during the PhD work. I would also like to take this opportunity to thank all the people who directly and indirectly helped in finishing my thesis.

Publications based on this Thesis

1. J. Saketha Nath, Chiranjib Bhattacharyya and M. Narasimha Murty. Clustering Based Large Margin Classification: A Scalable Approach using SOCP Formulation. *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2006.*

2. J. Saketha Nath and Chiranjib Bhattacharyya. Maximum Margin Classifiers with Specified False Positive and False Negative Error Rates. *Proceedings of the SIAM International Conference on Data mining, 2007.*

3. Rashmin B, J. Saketha Nath, Krishnan S, Sivaramakrishnan, Chiranjib Bhattacharyya and M N Murty. Focused Crawling with Scalable Ordinal Regression Solvers. *Proceedings of the International Conference on Machine Learning, 2007.*

4. J. Saketha Nath, Aharon Ben-Tal and Chiranjib Bhattacharyya. Robust Classification for Large Datasets. *Submitted to Journal of Machine Learning Research.*

Abstract

This thesis explores Chance-Constrained Programming (CCP) in the context of learning. It is shown that chance-constraint approaches lead to improved algorithms for three important learning problems — classification with specified error rates, large dataset classification and Ordinal Regression (OR). Using moments of training data, the CCPs are posed as Second Order Cone Programs (SOCPs). Novel iterative algorithms for solving the resulting SOCPs are also derived. Borrowing ideas from robust optimization theory, the proposed formulations are made robust to moment estimation errors.

A maximum margin classifier with specified false positive and false negative rates is derived. The key idea is to employ chance-constraints for each class which imply that the actual misclassification rates do not exceed the specified. The formulation is applied to the case of biased classification.

The problems of large dataset classification and ordinal regression are addressed by deriving formulations which employ chance-constraints for clusters in training data rather than constraints for each data point. Since the number of clusters can be substantially smaller than the number of data points, the resulting formulation size and number of inequalities are very small. Hence the formulations scale well to large datasets.

The scalable classification and OR formulations are extended to feature spaces and the kernelized duals turn out to be instances of SOCPs with a single cone constraint. Exploiting this specialty, fast iterative solvers which outperform generic SOCP solvers, are proposed. Compared to state-of-the-art learners, the proposed algorithms achieve a speed up as high as 10000 times, when the specialized SOCP solvers are employed.

The proposed formulations involve second order moments of data and hence are

susceptible to moment estimation errors. A generic way of making the formulations robust to such estimation errors is illustrated. Two novel confidence sets for moments are derived and it is shown that when either of the confidence sets are employed, the robust formulations also yield SOCPs.

Contents

List of Tables

List of Figures

Notation and Abbreviations

CCP	—	Chance-Constrained Program
SOCP	—	Second Order Cone Program
$\mathcal{D}$	—	Training dataset
FP, FN	—	False Positive, False Negative error rates
m	—	Number of training data points
n	—	Dimensionality of training data
SVM	—	Support Vector Machine
$\mathbf{w}, b$	—	Parameters of the discriminating hyperplane, $\mathbf{w}^\top \mathbf{x} - b = 0$
$\|\cdot\|_2$	—	2-norm or Euclidean norm of a vector
pdf	—	Probability density function
$X \sim f_X$	—	The random variable X has the pdf f_X
OR	—	Ordinal Regression
URL	—	Universal Resource Locator
I_e	—	Indicator of occurrence of event 'e'. $I_e = 1$ if 'e' occurs and 0 otherwise
$\mathbf{x}$	—	Lower case bold-faced symbol represents a vector
X	—	Upper case letter represents a random variable
$\mathbf{X}$	—	Upper case bold-faced symbol represents a matrix

Chapter 1

Introduction

Abstract

This chapter introduces the main theme of thesis and discusses the main contributions. The chapter also provides a road-map to the thesis which serves as a reader's guide.

Many learning tasks can be understood as constructing a function f which predicts the label y of an observation $\mathbf{x}$. Usually, the only information available is a finite set of examples, $\mathcal{D} = \{(\mathbf{x}_i, y_i),\ i = 1, \ldots, m\}$, known as the training set. Real-world learning applications desire that the *generalization error*, which is the error in predicting labels of examples not necessarily in $\mathcal{D}$, is low and acceptable. It is challenging to construct such a function because the training set is finite and may not represent the underlying distribution in its entirety. From a pragmatic perspective, the other main challenge is to develop efficient algorithms for constructing the function f, which are viable even when the training set size, m, is large.

This thesis explores the possibility of employing Chance-Constrained Programs (CCPs) for addressing these issues with learning algorithms. The key contribution is to show that chance-constraint approaches lead to accurate, scalable and robust learning algorithms. The Chebyshev-Cantelli inequality, which uses the second order moment information, is exploited in order to pose the CCPs as tractable, convex, optimization problems. It is

shown that the optimization problems, which are instances of Second Order Cone Programs (SOCPs), have elegant geometric interpretations and hence can be solved using efficient iterative schemes. Further, in cases where the SOCPs derived involve a single cone constraint, novel algorithms which outperform generic SOCP solvers are proposed. Using robust optimization principles the formulations are made robust from moment estimation errors. It is shown that the robust formulations are also instances of SOCPs and hence are tractable.

In the past, chance-constraint approaches were employed for handling uncertainty in training data [44] and for constructing learners where bounds on error rates are known [32]. This thesis shows that chance-constraint approaches can also be employed for achieving scalability, enabling the learning algorithms to handle large datasets involving millions of examples. It is shown that the chance-constraint based learning algorithms, when compared to the state-of-the-art, give a speed-up as high as 10000 times in some cases. The specific learning problems discussed in this thesis and the proposed chance-constraint approaches for solving them are briefly outlined below.

Real world classification applications desire to obtain a classifier whose actual misclassification rate does not exceed the maximum tolerable limit. For instance, in case of medical diagnosis of cancer [30], it is required that the false negative (FN) rate is low. Whereas slightly high false positive (FP) rate may be tolerated. This is because the cost of misclassifying a cancer patient is very high compared to that of misclassifying a normal patient. In this thesis, a maximum margin classification formulation with specified false positive and false negative rates is proposed, which has potential to be applied in cases where classifiers with preferential bias towards a particular class are desired. The key idea is to employ chance-constraints for each class implying that the actual error rates do not exceed the specified. Using the Chebyshev-Cantelli inequality and second order moments of class conditional densities, the resulting CCP is posed as an SOCP.

Large-scale binary classification is another problem discussed in this thesis. Many real-world classification applications like Intrusion detection, web page classification and spam filtering involve analyzing millions of data points. However most of the existing

classification algorithms either require the training data to be in memory or make multiple passes of the dataset and hence are not attractive for large dataset classification. A similar problem is that of training an ordinal regressor on large datasets. Ordinal Regression (OR) problems frequently occur in the areas of Information retrieval, social science, personalized searches and other ranking based applications. The existing algorithms for OR do not scale well for reasons same as in case of large-scale classification.

The above mentioned large-scale learning problems are addressed by deriving formulations which employ chance-constraints for clusters in training data rather than constraints for each data point. Using the second order moments of clusters, the resulting CCPs are posed as SOCPs involving one cone constraint and one linear constraint per cluster. Thus the size of optimization problem which needs to be solved is small even when the training data size, m, is large. An online clustering algorithm like BIRCH [48] can be employed to estimate moments of clusters efficiently in $O(m)$ time. Training time with the new learning scheme, which is the sum of clustering and SOCP solving times, grows linearly with training set size, as the cone program size depends on number of clusters rather than on number of data points. The scalable classification and OR formulations are extended to feature spaces and the duals also turn out to be instances of single cone-constrained SOCPs. Exploiting this special structure, fast iterative solvers are proposed, which outperform generic SOCP solvers. It is shown that the training time with the chance-constraint based scalable formulations, solved using the novel iterative algorithms, is orders of magnitude less than the state-of-the-art.

The novel SOCP formulations briefed above involve second order moments of training data and hence are susceptible to moment estimation errors. Using the large dataset classification formulation as an example, a generic method for introducing robustness from estimation errors is presented. The method is based on two new confidence sets for moments — separate and joint confidence sets. It is shown that when either of the confidence sets is employed, the robust variant of the original formulation also is an SOCP and hence is tractable.

1.1 Main Contributions

The thesis concentrates on developing chance-constraint approaches for learning, leading to improved algorithms. The main contributions are as follows:

- A maximum margin classifier with specified false positive and false negative rates is proposed. The formulation is posed as an SOCP and is applied to the case of biased classification. It is shown that the dual turns out to be the problem of minimizing distance between ellipsoids and an iterative algorithm to solve the dual efficiently is presented. The formulation is extended to non-linear classifiers and it is shown that the non-linear, linear formulations have the same form.

- A scalable binary classification formulation which employs chance-constraints for clusters in training data is proposed. Using second order moments of clusters, the resulting CCP is posed as a single cone-constrained SOCP involving a small number of variables and linear inequalities. When an online clustering algorithm is employed for estimating moments of clusters, the overall training time, which is the sum of clustering and SOCP solving times, grows linearly with training data size, m. The key advantage is that the training time is comparable to that with the state-of-the-art SVM solvers even when the formulation is solved using generic SOCP solvers. The geometric interpretation of the formulation turns out to be that of doing a maximum margin classification of spheres centered at means of clusters and radii proportional to the variances.

- Using similar ideas, a scalable, chance-constraint based ordinal regression formulation for large datasets is also derived. Methodology of extending the scalable formulation to feature spaces is presented and it is shown that the overall training time remains to be $O(m)$. Maximum number of support vectors with the non-linear formulation turns out to be the number of clusters, making it suitable for situations where fast predictions are desired. Another contribution is to pose the problem of focused crawling [11] as a large scale ordinal regression problem and solve efficiently using the proposed OR formulation.

- Fast iterative algorithms for solving the scalable classification and OR formulations, which are instances of SOCPs with single cone-constraint, are derived. The new SOCP solvers scale very well when compared to generic SOCP solvers and are very simple to code. When such solvers are employed, the chance-constraint based learning schemes outperform the state-of-the-art in terms of training time.

- Using ideas from robust optimization theory, the proposed formulations are made robust from moment estimation errors. Two novel confidence sets — separate and joint confidence sets for moments are derived and it is shown that when either of the confidence sets are employed, the robust variant of the original formulation is also an SOCP.

1.2 Thesis Road-Map

This section shows the organization of remainder of the thesis and serves as a road-map to the reader.

The problem of classification with specified error rates is described in chapter 2. The chapter begins by motivating the need for employing such a classifier in applications like biased classification. Section 2.1 reviews past work and discusses issues with the existing methods. Main contribution of the chapter, maximum margin formulation with specified FP and FN rates, is presented in section 2.2. Using moments of class-conditional densities, the formulation is posed as an SOCP. Section 2.3 presents dual of the formulation. The dual SOCP turns out to be the problem of minimizing distance between ellipsoids. An iterative algorithm to solve the dual efficiently is presented in section 2.3.1. Section 2.4 presents non-linear extensions for the proposed formulation. It is shown that the non-linear formulation has the same form as the linear formulation. Numerical experiments comparing performance of the non-linear and linear formulations, solved using generic SOCP solvers and the new iterative algorithm (section 2.3.1), are presented in section 2.5. Experiments also compare the new formulations with the biased SVM formulation [3]. The chapter concludes with a brief summary in section 2.6.

The problem of large dataset classification is discussed in chapter 3. The chapter starts by discussing the importance of the problem of scalable classification and justifies the proposed approach of employing a chance-constraint based scheme. In section 3.1 past work done on large dataset classification is presented. The main contribution, maximum margin classification formulation using chance-constraints for clusters, is presented in section 3.2. The section also describes the overall classification scheme. In section 3.3 the dual of the clustering based classification formulation is presented. The geometrical interpretation turns out to be that of classifying spheres centered at means and radii proportional to variances of the clusters. Section 3.4 presents results of experiments on large synthetic and real world datasets comparing the proposed scheme and the state-of-the-art SVM solver, SVM-Perf [26]. The experiments confirm that the chance-constraint based method compares well, both in terms of training time and accuracy, with SVM-Perf. In cases where the datasets are very large and do not fit in memory, SVM-Perf fails to complete training whereas the proposed scheme remains a viable option. Section 3.5 concludes the chapter with a summary.

Chapter 4 discusses the problem of large dataset ordinal regression. Initial discussion of the chapter motivates the need for a scalable, chance-constraint based solution. The discussion also introduces the problem of focused crawling and suggests that the focused crawling problem can be posed as a large scale OR problem. In section 4.1 a brief review of the past work on ordinal regression and focused crawling is presented. Section 4.2 presents the scalable chance-constraint based OR formulation. The formulation is extended to non-linear feature spaces in section 4.3. This section also derives the dual of the new OR formulation. Detailed discussion on the focused crawling problem and the suggested OR based solution are described in section 4.4. In section 4.5, experiments showing scalability of the scalable OR formulation are discussed. The section also presents experiments comparing the performance of the OR based crawler and the baseline crawler [11]. The chapter ends with a summary in section 4.6.

In chapter 5, a fast iterative algorithm for solving the scalable OR formulation is presented. The chapter begins by motivating the need for such a fast solver. Section 5.1

presents the fast algorithm which exploits the fact that the formulation is an instance of SOCP with only one cone constraint. Derivations in the chapter can be reproduced for the scalable classification formulation, as both the formulations are similar in structure (see appendix B). In section 5.2, experiments which compare scalability of the new iterative SOCP solver and `SeDuMi` [45], a generic SOCP solver, are presented. Experiments also show that the scalable OR scheme outperforms the state-of-the-art, when the new SOCP solver is employed. The chapter concludes with a brief summary in section 5.3.

A common problem with the proposed learning formulations is susceptibility to moment estimation errors. Chapter 6 deals with the issue of making the formulations robust to such estimation errors. The methodology is illustrated using the scalable classification formulation as an example. Section 6.1 describes the generic idea of using confidence sets for moments in order to introduce robustness. In sections 6.1.1, 6.1.2, two novel confidence sets for means and variances are derived for the special case of normal distribution of clusters. The sections also present the main contribution of the chapter — proving that the robust variants of the formulation, when either confidence sets are employed, are also SOCPs. Experimental results presented in section 6.2 prove the working of the robust formulations. Experiments also show that such robust formulations are indeed required. Section 6.3 concludes the chapter with a brief summary.

Chapter 7 concludes the thesis by summarizing main contributions. The chapter also discusses related issues and directions for future work.

Chapter 2

Classification with Specified Error Rates

Abstract

This chapter presents a maximum margin classification formulation with specified false positive and false negative error rates[1]*. The key idea is to employ chance-constraints for each class which imply that the positive and negative misclassification rates do not exceed the specified limits. The formulation is posed as an SOCP and is applied to the case of biased classification. An iterative algorithm to solve the dual, which turns out be the problem of minimizing distance between ellipsoids, is presented. Using the kernel trick, the formulation is extended to feature spaces.*

Real world classification applications require that the misclassification error incurred by the classifier is less than the tolerable limit. Moreover, in case of applications like medical diagnosis of cancer [30], tolerable limits on the false-negatives and false-positives differ. Because the cost of misclassifying a cancer patient is far higher than that of misclassifying a normal patient, usually low false-negative rates and relatively high false-positive rates are tolerated in such applications. Hence there is need to design classifiers that have some bias towards a particular class. Also it is common in such applications that the number of patients with cancer is far less than those who are normal. Hence

[1] *This work was presented at the SIAM International Conference on Data Mining, 2007.*

the training data is highly unbalanced in these applications. Traditional classification methods like SVM [46] do not address these issues satisfactorily.

This chapter studies the problem in the context of two classes, when data is summarized by the moments of class conditional densities. It is assumed that the maximum tolerable false-negative and false-positive error rates (η_1, η_2 respectively) 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 one can model the bias towards the positive class. Employing chance-constraints for each class, which imply that the maximum false-negative, false-positive rates do not exceed η_1 and η_2 respectively, a maximum margin formulation is derived. Using the Chebyshev-Cantelli inequality cone constraints equivalent to the chance-constraints are derived and the formulation is posed as an SOCP. As a special case of convex non-linear optimization, SOCPs have gained much attention in recent times due to their occurrence in solving many practical problems [34]. The formulation can be solved using generic SOCP solvers like `SeDuMi` and has potential to be exploited in situations where maximum tolerable error rates are known. As a specific application, the proposed formulation can be used for classification with preferential bias towards a particular class.

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 to solve the dual, based on the approach of Lin and Han [33]. Using kernel methods, the original formulation can be extended to feature spaces. The kernelized formulation has same structure as its linear counterpart and hence can be solved using the iterative algorithm for finding distance between ellipsoids.

The chapter is organized as follows: section 2.1 presents a brief review of past work done on biased classification. The new maximum margin formulation with specified error rates is presented in section 2.2. In section 2.3, the dual and its fast iterative solver are presented. Section 2.4 presents feature space extensions for the proposed formulation. Experimental results with the proposed formulation are detailed in section 2.5. The chapter concludes with a brief summary in section 2.6.

2.1 Past Work

Several methods exist in literature which address the problem of classification with preferential bias towards a particular class. Simple sampling techniques which introduce bias by up-sampling the examples of the important class or down-sampling the less important class examples or both exist [12, 31]. However down-sampling will lose information, while up-sampling may introduce noise. Methods which adjust the costs or shift the decision boundary towards the preferred class also exist [3, 9, 41]. Although such methods work well in practice, it is usually hard to build direct quantitative connections to the biased classifier's performance. These methods therefore fail to provide a rigorous approach to the task of classification where preferential bias towards one class is needed.

Biased minimax probability machine [24] is a formulation designed specifically for asymmetric cost classification. 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). This is done so as to achieve high accuracy for the preferred class while not having high error on the other class. However, with this method, at the optimum, p_1 may turn out to be less than p_2. The present formulation avoids such issues by taking an alternative and novel approach of designing a classifier whose positive and negative misclassification rates do not exceed the maximum allowed.

2.2 New Classification Formulation

This section presents the novel classification formulation with specified false positive and false negative error rates. Let $\mathcal{D} = \{(\mathbf{x}_i, y_i) | \mathbf{x}_i \in \mathbb{R}^n, y_i = \{1, -1\}, i = 1, \ldots, m\}$ be the training dataset consisting of data points $\mathbf{x}_i$ and labels y_i. Suppose X_1 represents the random vector that generates examples of the positive class and X_2 represents that of the negative class. Let the mean and covariance of X_i be $\mu_i \in \mathbb{R}^n$ and $\mathbf{\Sigma}_i \in \mathbb{R}^{n \times n}$ respectively for $i = 1, 2$. Note that $\mathbf{\Sigma}_1, \mathbf{\Sigma}_2$ are symmetric positive semi-definite.

Assume that $\mathbf{w}^\top \mathbf{x} - b = 0$ denotes the discriminating hyperplane and the corresponding positive and negative half spaces are denoted by:

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

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

$$\begin{aligned} \min_{\mathbf{w},b} \quad & \tfrac{1}{2}\|\mathbf{w}\|_2^2 \\ \text{s.t.} \quad & Prob(X_1 \in \mathcal{H}_2) \leq \eta_1 \\ & Prob(X_2 \in \mathcal{H}_1) \leq \eta_2 \\ & X_1 \sim (\mu_1, \mathbf{\Sigma}_1)\ X_2 \sim (\mu_2, \mathbf{\Sigma}_2) \end{aligned} \tag{2.1}$$

The chance-constraints $Prob(X_1 \in \mathcal{H}_2) \leq \eta_1$ and $Prob(X_2 \in \mathcal{H}_1) \leq \eta_2$ ensure that the false-negative and false-positive rates do not exceed η_1 and η_2. As in case of SVMs, the objective implies minimizing $\|\mathbf{w}\|_2$, leading to good generalization [46]. The chance-constraints in (2.1) can be re-written as deterministic constraints using the following theorem:

THEOREM 2.1. *Let X be an n-dimensional random variable having mean and covariance $(\mu, \mathbf{\Sigma})$. Then the following is true for any $\mathbf{c} \in \mathbb{R}^n,\ d \in \mathbb{R},\ 0 \leq e \leq 1$:*

$$\mathbf{c}^\top \mu - d \ \geq\ \kappa \sqrt{\mathbf{c}^\top \mathbf{\Sigma}\ \mathbf{c}} \ \Rightarrow\ Prob(\mathbf{c}^\top X \geq d) \geq e \tag{2.2}$$

where $\kappa = \sqrt{\frac{e}{1-e}}$.

Further if X is multivariate normal, then $\kappa = \Phi^{-1}(e)$. Φ is the distribution function[2] *of uni-variate normal with mean 0 and variance 1.*

Refer appendix A for a proof of theorem 2.1. The proof is based on the multivariate

[2] $\Phi(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{z} \exp\left(-s^2/2\right)\ ds$

generalization of the Chebyshev-Cantelli inequality [36]. Applying the theorem (2.1) with $\mathbf{c} = \mathbf{w}, d = b, e = 1 - \eta_1$ and $X = X_1$ gives $Prob(X_1 \in \mathcal{H}_2) \leq \eta_1$ if $\mathbf{w}^\top \mu_1 - b \geq \kappa_1 \sqrt{\mathbf{w}^\top \mathbf{\Sigma}_1 \mathbf{w}}$ where $\kappa_1 = \sqrt{\frac{1-\eta_1}{\eta_1}}$. Similarly if $b - \mathbf{w}^\top \mu_2 \geq \kappa_2 \sqrt{\mathbf{w}^\top \mathbf{\Sigma}_2 \mathbf{w}}, \kappa_2 = \sqrt{\frac{1-\eta_2}{\eta_2}}$, then $Prob(X_2 \in \mathcal{H}_1) \leq \eta_2$.

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 $\eta_i = 1$. In other words, $\mathbf{w}^\top \mu_1 - b \geq 1$ and $b - \mathbf{w}^\top \mu_2 \geq 1$. The problem (2.1) can now be stated as the following deterministic optimization problem.

$$\begin{aligned} \min_{\mathbf{w},b} \quad & \tfrac{1}{2}\|\mathbf{w}\|_2^2 \\ \text{s.t.} \quad & \mathbf{w}^\top \mu_1 - b \geq 1 + \kappa_1 \sqrt{\mathbf{w}^\top \mathbf{\Sigma}_1 \mathbf{w}} \\ & b - \mathbf{w}^\top \mu_2 \geq 1 + \kappa_2 \sqrt{\mathbf{w}^\top \mathbf{\Sigma}_2 \mathbf{w}} \end{aligned} \tag{2.3}$$

Since the matrices $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$ are symmetric positive semi-definite, there exist square matrices $\mathbf{C}_1$ and $\mathbf{C}_2$ such that $\mathbf{\Sigma}_1 = \mathbf{C}_1 \mathbf{C}_1^\top$ and $\mathbf{\Sigma}_2 = \mathbf{C}_2 \mathbf{C}_2^\top$. Now, (2.3) can be written as:

$$\begin{aligned} \min_{\mathbf{w},b} \quad & \tfrac{1}{2}\|\mathbf{w}\|_2^2 \\ \text{s.t.} \quad & \mathbf{w}^\top \mu_1 - b \geq 1 + \kappa_1 \|\mathbf{C}_1^\top \mathbf{w}\|_2 \\ & b - \mathbf{w}^\top \mu_2 \geq 1 + \kappa_2 \|\mathbf{C}_2^\top \mathbf{w}\|_2 \end{aligned} \tag{2.4}$$

Clearly, the optimization problem (2.4) 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 (2.4) imply that the hyperplane $\mathbf{w}^\top \mathbf{x} - b = 0$ must separate the means μ_1 and μ_2. Thus, whenever the means are not coinciding, the problem (2.4) can be made feasible by choosing appropriate values for η_1 and η_2. The non-linear constraints in (2.4) are known as second order cone constraints.

The formulation (2.4) can be written in the following standard SOCP form:

$$\begin{aligned} \min_{\mathbf{w},b,t} \quad & t \\ \text{s.t.} \quad & t \geq \|\mathbf{w}\|_2 \\ & \mathbf{w}^\top \mu_1 - b \geq 1 + \kappa_1 \|\mathbf{C}_1^\top \mathbf{w}\|_2 \\ & b - \mathbf{w}^\top \mu_2 \geq 1 + \kappa_2 \|\mathbf{C}_2^\top \mathbf{w}\|_2 \end{aligned} \tag{2.5}$$

SOCP problems can be efficiently solved by interior point methods for convex non-linear optimization [39]. For a discussion on further efficient algorithms and applications of SOCP see [34]. Once the formulation is solved for $\mathbf{w}$ and b, the decision function given in (2.6) can be used to classify a new data point $\mathbf{x}$.

$$f(\mathbf{x}) = \text{sign}\left(\mathbf{w}^\top \mathbf{x} - b\right) \tag{2.6}$$

By varying values of the parameters η_1 and η_2, bias can be introduced into the classifier in a controlled way. The proposed classifier also has potential to be exploited in applications where the maximum tolerable error rates are specified.

2.3 Dual and its Iterative Solver

Constraints in the new classification formulation (2.4) have an elegant geometric interpretation. In order to see this, consider the following problem. Suppose

$$B(\mu, \mathbf{C}, \kappa) = \{\mathbf{x} \mid \mathbf{x} = \mu - \kappa \mathbf{C}\mathbf{u},\ \|\mathbf{u}\|_2 \leq 1\} \tag{2.7}$$

represents an ellipsoid centered at μ, whose shape is determined by $\mathbf{C}, (\mathbf{\Sigma} = \mathbf{C}\mathbf{C}^\top)$ and size by κ. Also assume that $\mathbf{C}$ is square full-rank, in which case

$$B(\mu, \mathbf{C}, \kappa) = \left\{\mathbf{x} \mid (\mathbf{x} - \mu)^\top \mathbf{\Sigma}^{-1} (\mathbf{x} - \mu) \leq \kappa^2\right\}, \tag{2.8}$$

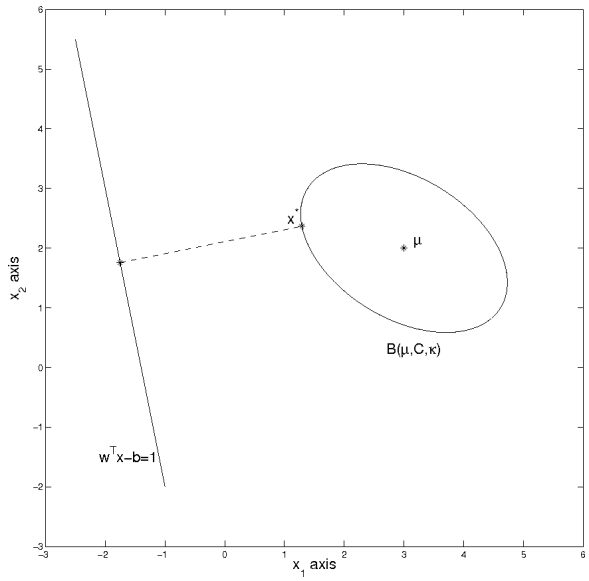

Figure 2.1: Illustration showing the geometric intuition behind the constraints of the proposed formulation

Now consider the problem of constraining all points in the ellipsoid to lie in the positive half-space, $\mathbf{w}^\top \mathbf{x} - b \geq 1$ (assume that the hyperplane does not intersect the ellipsoid). Mathematically, this can be written as:

$$\mathbf{w}^\top \mathbf{x} - b \geq 1 \; \forall \; \mathbf{x} \in B(\mu, \mathbf{C}, \kappa) \tag{2.9}$$

Though this is a set of infinite constraints, one can satisfy them by finding the point in ellipsoid closest to the hyperplane and then constraining the point to lie in the positive half-space, $\mathbf{w}^\top \mathbf{x} - b \geq 1$. Finding the closest point, $\mathbf{x}^*$, is easy because of the special form of the set $B(\mu, \mathbf{C}, \kappa)$. Firstly, the point must lie on the boundary of the ellipsoid and secondly, the direction of normal at $\mathbf{x}^*$ must be opposite to $\mathbf{w}$ (see figure 2.1):

$$2\mathbf{\Sigma}^{-1}\mathbf{x}^* - 2\mathbf{\Sigma}^{-1}\mu = \rho\mathbf{w} \tag{2.10}$$

where ρ is some negative constant. The value of ρ is obtained by using the fact that $\mathbf{x}^*$

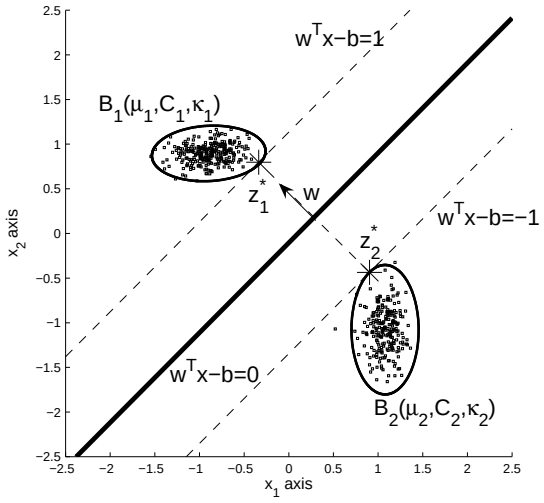

Figure 2.2: Illustration showing the geometric interpretation of the proposed formulation

lies on the boundary of ellipsoid:

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

Using this, one can get the value of $\mathbf{x}^*$:

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

As mentioned earlier, it is enough to constrain that $\mathbf{w}^\top \mathbf{x}^* - b \geq 1$, in order to satisfy the infinite constraints in (2.9). In other words, $\mathbf{w}^\top \mu - b \geq 1 + \kappa \|\mathbf{C}^\top \mathbf{w}\|_2$. Note that this is similar in form to the constraints in (2.4).

Thus geometrical interpretation of the proposed formulation is to find a maximum margin hyperplane which separates ellipsoids whose centers are the means, shapes are parametrized by the covariance matrices and sizes depend on the parameters κ_1 and κ_2 (see figure 2.2). In the following text, this is derived more rigorously using Duality theory. The dual norm characterization gives

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

Using this, the formulation (2.4) can be re-written as:

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

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

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

with the constraints $\|\mathbf{u}_1\|_2 \leq 1,\ \|\mathbf{u}_2\|_2 \leq 1,\ \lambda_1 \geq 0$ and $\lambda_2 \geq 0$. From Karush-Kuhn-Tucker (KKT) conditions [19], 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

$$\mathbf{w} = \lambda(\mu_1 - \kappa_1 \mathbf{C}_1 \mathbf{u}_1 - \mu_2 - \kappa_2 \mathbf{C}_2 \mathbf{u}_2) \tag{2.11}$$

The dual formulation is obtained by maximizing $\mathcal{L}$ with respect to dual variables $\lambda \geq 0, \mathbf{u}_1 \leq 1$ and $\mathbf{u}_2 \leq 1$, subject to the constraints $\frac{\partial \mathcal{L}}{\partial b} = 0$, $\nabla_{\mathbf{w}} \mathcal{L} = 0$:

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

The objective is maximized when

$$\lambda = \frac{2}{\|\mathbf{z}_1 - \mathbf{z}_2\|_2^2}. \tag{2.12}$$

and the maximized value is $\frac{2}{\|\mathbf{z}_1-\mathbf{z}_2\|_2^2}$. Since it is assumed that the ellipsoids are non-intersecting, $z_1 - z_2 \neq 0$ at optimality. Using (2.12), the dual can be re-written as follows:

$$\begin{aligned} \min_{\mathbf{z}_1,\mathbf{z}_2} \quad & \|\mathbf{z}_1 - \mathbf{z}_2\|_2^2 \\ & \mathbf{z}_1 \in B_1(\mu_1, \mathbf{C}_1, \kappa_1),\ \mathbf{z}_2 \in B_2(\mu_2, \mathbf{C}_2, \kappa_2) \end{aligned} \tag{2.13}$$

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 optimization problem (2.13) has an 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 parametrized by the matrices $\mathbf{C}_1$ and $\mathbf{C}_2$ respectively. Thus the dual optimization problem minimizes distance between two ellipsoids. The value of $\mathbf{w}$ can be obtained by using:

$$\mathbf{w} = 2\frac{\mathbf{z}_1^* - \mathbf{z}_2^*}{\|\mathbf{z}_1^* - \mathbf{z}_2^*\|_2^2} \tag{2.14}$$

where, $\mathbf{z}_1^*, \mathbf{z}_2^*$ are the optimal variables of (2.13). The KKT conditions of the dual can be summarized as follows

$$\begin{aligned} -\kappa_1 \mathbf{C}_1^\top(\mathbf{z}_1 - \mathbf{z}_2) + \gamma_1 \mathbf{u}_1 = 0, \quad & \gamma_1(\|\mathbf{u}_1\|_2 - 1) = 0, \\ -\kappa_2 \mathbf{C}_2^\top(\mathbf{z}_1 - \mathbf{z}_2) + \gamma_2 \mathbf{u}_2 = 0, \quad & \gamma_2(\|\mathbf{u}_2\|_2 - 1) = 0, \\ \|\mathbf{u}_1\|_2 \leq 1,\ \|\mathbf{u}_2\|_2 \leq 1, \quad & \gamma_1 \geq 0, \gamma_2 \geq 0 \end{aligned} \tag{2.15}$$

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 $\theta(\mathbf{u}, \mathbf{v}) = \arccos\left(\frac{\mathbf{u}^\top \mathbf{v}}{\|\mathbf{u}\|_2 \|\mathbf{v}\|_2}\right)$. Then, at optimality:

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

If $\mathbf{\Sigma}_1, \mathbf{\Sigma}_2$ are positive definite or more specifically 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

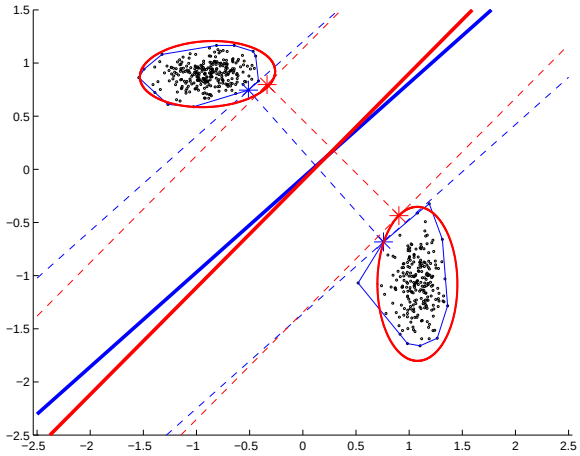

Figure 2.3: Illustration comparing the classifiers obtained with SVM and present method. The SVM solution is shown in blue, whereas that of the present method is shown in red ($\eta_1 = \eta_2 = 0.1$).

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 (2.12), we have $\lambda > 0$, which implies that both the constraints in (2.4) 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

$$b = 2\frac{\mathbf{z}_1^\top (\mathbf{z}_1 - \mathbf{z}_2)}{\|\mathbf{z}_1 - \mathbf{z}_2\|_2^2} - 1 \tag{2.16}$$

It is interesting to note the analogy between SVMs [7] and the proposed formulation. In case of SVM, the dual turns out to be the problem of minimizing distance between two convex hulls, whereas in the present case, the dual minimizes distance between ellipsoids. Figure 2.3 shows the optimal hyperplane as obtained with the formulation (2.4) 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 SVM was introduced. However it involves an additional regularization parameter C. The figure also confirms the equivalence of the primal (2.4) and dual (2.13).

2.3.1 Iterative Algorithm for Solving Dual

The geometrical 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 and Han [33] provide an iterative, provably convergent algorithm for this geometric optimization problem. In the following text, application of the same algorithm for solving (2.13) is presented. Suppose the matrices $\mathbf{\Sigma}_i$ 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

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

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

Input μ_i, $\mathbf{\Sigma}_i$ and κ_i

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

Initialization Compute the following:

1. $\mathbf{A}_i, \mathbf{b}_i$ and ρ_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 $t_i,\ i = 1, 2$:

$$\begin{aligned}
&\frac{1}{2}t_i^2 \left\{(\mathbf{c}_1 - \mathbf{c}_2)^\top \mathbf{A}_i (\mathbf{c}_1 - \mathbf{c}_2)\right\} - \\
&\quad t_i \left\{(\mathbf{c}_1^\top - \mathbf{c}_2^\top)(\mathbf{A}_i c_1 + \mathbf{b}_i)\right\} + \\
&\left\{\frac{1}{2}\mathbf{c}_1^\top \mathbf{A}_i \mathbf{c}_1 + \mathbf{b}_i^\top \mathbf{c}_1 + \rho_i\right\} = 0
\end{aligned}$$

(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 (2.13). The outline of the proof of convergence is provided here (refer [33] for details), assuming that the ellipsoids are separated. The KKT optimality conditions for (2.13) are (in terms of the ellipsoids in standard form):

$$\mathbf{z}_1^* \in \Omega(B_1), \qquad \mathbf{z}_2^* \in \Omega(B_2),$$
$$\theta(\mathbf{z}_2^* - \mathbf{z}_1^*, \mathbf{A}_1\mathbf{z}_1^* + \mathbf{b}_1) = 0, \quad \theta(\mathbf{z}_1^* - \mathbf{z}_2^*, \mathbf{A}_2\mathbf{z}_2^* + \mathbf{b}_2) = 0,$$

where, $\Omega(B_i)$ represents the boundary of the ellipsoid B_i. These optimality conditions say that the optimal $(\mathbf{z}_1^*,\ \mathbf{z}_2^*)$ 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 and regular, KKT conditions are necessary and sufficient. Note that these conditions are equivalent to those given in (2.15). 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,

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

Triangle inequality gives:

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

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

$$\|\mathbf{z}_1^k - \mathbf{z}_2^k\| \geq \|\mathbf{z}_1^{k+1} - \mathbf{z}_2^{k+1}\|$$

Therefore, the sequence of distances $\{\|\mathbf{z}_1^k - \mathbf{z}_2^k\|\}$ is monotone and hence converges. Now one can also prove that for such a sequence,

$$\lim_{k\to\infty} \theta(\mathbf{z}_2^* - \mathbf{z}_1^*, \mathbf{A}_1\mathbf{z}_1^* + \mathbf{b}_1) = 0,$$

$$\lim_{k\to\infty} \theta(\mathbf{z}_1^* - \mathbf{z}_2^*, \mathbf{A}_2\mathbf{z}_2^* + \mathbf{b}_2) = 0,$$

proving that $(\mathbf{z}_1^k,\ \mathbf{z}_2^k)$ converges to $(\mathbf{z}_1^*,\ \mathbf{z}_2^*)$.

Note that at every step of iteration, two one-dimensional quadratic equations are solved. However, the initialization cost is high, 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.

2.4 Non-linear Classifiers

The formulation (2.3) provides a linear classifier and hence cannot deal with non-linearly separable data. In the following text, the formulation is extended to feature spaces in order to handle such data. Let $\mathbf{T}_1$ be a $n \times m_1$ matrix, where each column of $\mathbf{T}_1$ is a positive training data point. Similarly, let $\mathbf{T}_2$ be a $n \times m_2$ data matrix for the other class. Let $[\mathbf{M}_1, \mathbf{M}_2]$ and $[\mathbf{M}_1; \mathbf{M}_2]$ denote the horizontal and vertical concatenation of the matrices $\mathbf{M}_1$ and $\mathbf{M}_2$ respectively. The empirical estimates of the mean and covariance can be written as:

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

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

and similarly

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

where, $\mathbf{e}_i$ is a vector of ones of dimension m_i and $\mathbf{I}_i$ is an identity matrix of dimensions $m_i \times m_i$.

Since $\mathbf{w}$ is a vector in n dimensional space, it can be written as a linear combination of the training data points and other points in $\mathbb{R}^n$ which are orthogonal to the training data points. Mathematically, this can be written as $\mathbf{w} = [\mathbf{T}_1, \mathbf{T}_2]\mathbf{s} + \mathbf{M}\mathbf{r}$ where $\mathbf{M}$ is a matrix whose columns are vectors orthogonal to the 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 span entire $\mathbb{R}^n$. Now, the terms involving $\mathbf{w}$ in the constraints of (2.3) can be written as

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

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

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

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

and

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

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

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 0. In other words, the optimal $\mathbf{w}$ is a linear combination of the training data points only. Using this, the formulation (2.3) can be written as:

$$\begin{aligned} \min_{\mathbf{s},b} \quad & \tfrac{1}{2}\mathbf{s}^\top\mathbf{K}\mathbf{s} \\ \text{s.t.} \quad & \mathbf{s}^\top\mathbf{g}_1 - b \geq 1 + \kappa_1\sqrt{\mathbf{s}^\top\mathbf{G}_1\mathbf{s}}, \\ & b - \mathbf{s}^\top\mathbf{g}_2 \geq 1 + \kappa_2\sqrt{\mathbf{s}^\top\mathbf{G}_2\mathbf{s}} \end{aligned} \tag{2.17}$$

where, $\mathbf{K} = [\mathbf{K}_{11}, \mathbf{K}_{12}; \mathbf{K}_{21}, \mathbf{K}_{22}]$. Note that, in order to solve (2.17), 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 products is by using kernel functions which satisfy Mercer conditions [37]. Assuming that such a kernel function, $k : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$, is available, the quantities $\mathbf{g}_1, \mathbf{g}_2, \mathbf{G}_1, \mathbf{G}_2$ and $\mathbf{K}$ can be calculated in any feature space. Suppose $\mathbf{K}$ is positive definite, in which case $\mathbf{K} = \mathbf{L}^\top\mathbf{L}$, $\mathbf{L}$ is a full rank square matrix. Now the formulation (2.17) can be re-written as:

$$\begin{aligned} \min_{\mathbf{v},b} \quad & \tfrac{1}{2}\|\mathbf{v}\|_2^2 \\ \text{s.t.} \quad & \mathbf{v}^\top\mathbf{h}_1 - b \geq 1 + \kappa_1\sqrt{\mathbf{v}^\top\mathbf{H}_1\mathbf{v}}, \\ & b - \mathbf{v}^\top\mathbf{h}_2 \geq 1 + \kappa_2\sqrt{\mathbf{v}^\top\mathbf{H}_2\mathbf{v}} \end{aligned} \tag{2.18}$$

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 (2.3). Again, $\mathbf{H}_i$, being positive semi-definite, can be written as $\mathbf{H}_i = \mathbf{D}_i\mathbf{D}_i^\top$. (2.18) can be solved using interior point methods when cast into the following standard SOCP form:

$$\begin{aligned} \min_{\mathbf{v},b,t} \quad & t \\ \text{s.t.} \quad & t \geq \|\mathbf{v}\|_2 \\ & \mathbf{v}^\top\mathbf{h}_1 - b \geq 1 + \kappa_1\|\mathbf{D}_1^\top\mathbf{v}\|_2, \\ & b - \mathbf{v}^\top\mathbf{h}_2 \geq 1 + \kappa_2\|\mathbf{D}_2^\top\mathbf{v}\|_2 \end{aligned} \tag{2.19}$$

Using the arguments presented in section 2.3, the dual of (2.18) can be written as:

$$\begin{aligned} \min_{\mathbf{z}_1,\mathbf{z}_2} \quad & \|\mathbf{z}_1 - \mathbf{z}_2\|_2^2 \\ & \mathbf{z}_1 \in B_1(\mathbf{h}_1, \mathbf{D}_1, \kappa_1),\ \mathbf{z}_2 \in B_2(\mathbf{h}_2, \mathbf{D}_2, \kappa_2) \end{aligned} \tag{2.20}$$

and can be solved using the iterative geometric algorithm presented in section 2.3.1. Once the optimum value of $\mathbf{v}$ and b are obtained either by solving the SOCP problem (2.19) or by the iterative algorithm, one can classify a new data point $\mathbf{x}$ using the following decision function.

$$f(\mathbf{x}) \equiv \text{sign}(\mathbf{w}^\top\mathbf{x} - b) = \text{sign}(\mathbf{s}^\top\mathbf{K}_\mathbf{x} - b) \tag{2.21}$$

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

In practical experiments, it may well happen that the positive/negative error rate computed on the test set is greater than η_1/η_2. This is because estimated moments are employed in cone constraints instead of the true, unknown moments. Validity of the cone constraint depends on how accurate the estimates are. Chapter 6 discusses this issue and suggests methods for introducing robustness to moment estimation errors.

Table 2.1: Results on benchmark datasets, comparing the performance of **NL-SOCP**, **NL-ITER**, **L-SOCP**, **L-ITER** algorithms.

	η_1	η_2	**NL-SOCP**		**NL-ITER**		**L-SOCP**		**L-ITER**	
% err			+ve	-ve	+ve	-ve	+ve	-ve	+ve	-ve
Breast Cancer	0.9	0.3	12.74	00.56	12.74	00.56	16.98	00.00	16.04	00.84
$m = 569, n = 30,$	0.7	0.3	10.85	01.12	10.85	01.12	13.68	00.00	13.21	01.40
$m_1 = 212, m_2 = 357,$	0.5	0.3	04.72	01.96	04.72	01.96	05.19	00.84	07.55	02.24
$\zeta = 0.032$	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	0.9	0.7	30.14	31.94	30.14	31.94	×	×	×	×
$m = 400, n = 2,$	0.7	0.7	20.57	36.65	20.57	36.65	×	×	×	×
$m_1 = 209, m_2 = 191,$	0.5	0.7	12.44	41.89	12.92	41.36	×	×	×	×
$\zeta = 3$	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	0.9	0.3	10.15	01.28	10.53	01.28	09.02	00.43	09.02	00.43
$m = 500, n = 2,$	0.7	0.3	06.39	01.71	07.52	01.71	06.77	00.43	06.77	00.43
$m_1 = 266, m_2 = 234,$	0.5	0.3	05.26	02.56	06.02	02.56	04.51	01.28	04.51	01.28
$\zeta = 20$	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	0.9	0.9	14.60	22.50	14.60	22.50	18.99	14.38	18.99	14.38
$m = 297, n = 13,$	0.7	0.9	13.14	27.50	13.14	28.13	17.52	17.50	17.52	17.50
$m_1 = 137, m_2 = 160,$	0.5	0.9	11.68	32.50	11.68	32.50	13.14	21.88	13.14	21.88
$\zeta = 0.16$	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	×	×	×	×

2.5 Numerical Experiments

This section presents experimental results comparing the performance of the proposed non-linear (2.19) and linear (2.5) classification formulations, solved using `SeDuMi` software (denoted by **NL-SOCP** and **L-SOCP** respectively) and the iterative algorithm for dual (denoted by **NL-ITER** and **L-ITER** respectively). Recall that the iterative algorithm required that the matrices $\mathbf{\Sigma}_i$ and $\mathbf{H}_i$ to be positive definite (section 2.3.1). However, in practice, $\mathbf{\Sigma}_i$ or $\mathbf{H}_i$ can be ill-conditioned. To handle such cases, regularization $\mathbf{\Sigma}_i = \mathbf{\Sigma}_i + \epsilon\mathbf{I}$ 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 $k(\mathbf{x}_i, \mathbf{x}_j) = \exp\{-\zeta\|\mathbf{x}_i - \mathbf{x}_j\|_2^2\}$, with parameter ζ, is used to evaluate dot products in the feature space. The first set of experiments have been done to show that:

- Varying η_1 and η_2 different classifiers are obtained, whose false positive and false negative error rates vary accordingly.

Table 2.2: Results on benchmark datasets, comparing the errors obtained with **L-SVM** and **L-SOCP**.

Dataset	**Method**	$\mathbf{C_+/\eta_1}$	$\mathbf{C_-/\eta_2}$	% 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 2.3: Results on benchmark datasets, comparing the risk obtained with **L-SVM** and **L-SOCP**.

Dataset	**Method**	$\mathbf{C_+/\eta_1}$	$\mathbf{C_-/\eta_2}$	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

- 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 2.1 shows the results on some benchmark datasets. The Breast Cancer and Heart Disease datasets have been acquired from UCI-ML repository [8]. 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[3]. The table shows the dimensions of each dataset and the value of ζ used for non-linear classifiers. For each dataset, the actual false-negative and false-positive rates obtained, with all the 4 classifiers, are shown. The error values represent the 3-fold cross validation errors averaged over 3 cross validation experiments. The value '$\times$' 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.

[3]available at `http://www.cs.toronto.edu/~delve/data/datasets.html`

Observe that, in general, as η_1 value is decreased the false-negative error decreases, the false-positive error increases and the errors are less than the specified limits. 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 [3]. Such variants of the traditional SVMs replace the term $C\sum_i \xi_i$ in the objective by $C_+ \sum_{y_i=1} \xi_i + C_- \sum_{y_i=-1} \xi_i$, allowing for biased classification. Experiments on two Datasets are shown: 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 2.2 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 2.3 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 results are shown for the tuned set of parameters only. Results confirm that the proposed classifier achieves performance comparable to that of SVMs. The advantage being that the error rates are now guaranteed to be less than the tolerable limits.

2.6 Summary

A maximum margin classifier whose probability of misclassification, on each of the two classes, is bounded above by a specified value was presented. Chance-constraints for each class were employed to achieve the task. Using the Chebyshev's inequality and

second order moments of class-conditional densities, the resulting CCP was posed as an SOCP. The dual problem turns out to be that of finding the distance between two ellipsoids and the optimal hyperplane is perpendicular to the line joining the minimum distant points. An iterative algorithm to solve the dual was presented which avoids use of any optimization routines. An extension of the original formulation for feature spaces using kernel methods was also presented. As in the linear classifier case, the feature space classifier can be solved by posing as an SOCP or by using the iterative algorithm. Experimental results on benchmark datasets show that the false-negative and false-positive rates are less than the specified and in general achieve similar generalization as the SVMs.

Chapter 3

Scalable Maximum Margin Classification

Abstract

In this chapter we propose a novel chance-constraint based SOCP formulation[1] *which scales well for large datasets. The key idea is to derive a maximum margin classification formulation which minimizes training error by employing second order moment based chance constraints for clusters rather than constraints for individual data points. An online clustering algorithm is used to estimate the moments of clusters. It is shown that the training time is comparable with the state-of-the-art linear time SVM solvers even when generic SOCP solvers are used to solve the formulation. Further improvement in training time can be achieved by employing fast solvers for the new SOCP formulation.*

Recent advances in technology have enabled efficient generation, collection and storage of huge amounts of data. As a result many of the real-world binary classification applications involve analyzing millions of data points. Intrusion detection, web page classification and spam filtering applications are a few of them. Most of the existing classification algorithms either require the training data to be in memory or make multiple passes of the dataset and hence are not attractive for large dataset classification.

[1] *This work was presented at the* 12^{th} *ACM SIGKDD conference, 2006.*

Support Vector Machines are one of the most successful classifiers that achieve good generalization in practice. SVMs owe their success to the fact that they perform maximum margin classification of data points. SVMs (soft-margin SVMs) pose the classification problem as a convex quadratic optimization problem of size $m+n+1$, where m is the number of training data points and n is their dimensionality. The optimization problem has a quadratic objective function and $O(m)$ linear inequalities. Though problem size in case of the SVMs scales with m, they have emerged as useful tools for classification in practice primarily because of the availability of efficient algorithms like SMO [40] and chunking [25], which solve the dual of the SVM formulation. However, these algorithms are known to be at least $O(m^2)$ in running time (see [40, 47]) and hence not scalable to large datasets.

The key idea in this work is to perform maximum margin classification as in case of the SVMs, however, a novel convex optimization formulation, whose size is independent of the training set size, is employed to achieve the task. The class conditional densities are assumed to be modeled using mixture models with spherical covariances. An online clustering algorithm is employed to estimate the second order moments of components of the mixture models, $(\mu_j, \sigma_j^2\mathbf{I})$. The proposed formulation minimizes training error by employing chance constraints for clusters rather than constraints for individual data points. Using the Chebyshev's inequality and second order moments of clusters, cone constraints equivalent to the chance constraints are derived. A maximum margin classification formulation, similar in spirit to the SVMs, is then proposed using the derived cone constraints. The formulation is posed as an SOCP problem of size k (number of clusters/components), having k linear constraints, one cone constraint and can be solved using generic SOCP solvers like `SeDuMi`. Since the formulation size is independent of the training dataset size, the proposed classification scheme can be employed for large datasets.

Estimation of the moments of the component distributions can be done using an efficient clustering scheme, such as BIRCH [48]. BIRCH, in a single pass over the data constructs a CF-tree (Cluster Feature tree), given a limited amount of resources. CF-tree

consists of the sufficient statistics for the hierarchy of clusters in the data. BIRCH also handles outliers effectively as a by-product of clustering. The experiments presented in the chapter show that the overall training time, which is sum of the clustering and SOCP solving times, is comparable to the that of the state-of-the-art linear time SVM solvers even when generic SOCP solvers are used to solve the new classification formulation. This is because the problem size for the new formulation is number of clusters and that for the SVMs is number of data points. By employing fast solvers which are tuned for the SOCP formulation, further improvement in training time can be achieved (see chapter 5).

The remainder of the chapter is organized as follows. In section 3.1 we present a brief review of the past work on large dataset classification. The chance-constraint based, scalable classification scheme is presented in section 3.2. The geometric interpretation and dual of the proposed formulation are presented in section 3.3. Section 3.4 presents the experiments on synthetic and real world datasets which show the scalability of the new classification scheme. Section 3.5 concludes the chapter with a brief summary.

3.1 Past Work on Large Dataset Classification

As discussed in the previous section, SVMs achieve good generalization in practice, but their utility in large dataset classification is limited by non-availability of scalable solvers. Hence most of the past work on large dataset classification concentrated on developing fast SVM solvers [18, 20, 26, 28, 35]. We take an orthogonal approach of making the formulation itself scalable, while still performing maximum margin classification. Further scalability can be achieved by employing fast algorithms for solving the proposed formulation.

Clustering before computing the classifier is an interesting strategy for large scale problems. CB-SVM [47] is an iterative, hierarchical clustering based SVM algorithm, which handles large datasets. The algorithm thrives on the fact that the SVM optimization solution depends only on a small set of data points called support vectors that

lie near the optimal classification boundary. The authors, in their paper, show that the algorithm gives accuracies comparable to SMO with a very small run-time. The proposed classification method also uses clustering as a pre-processing step for classification. However, the method does not proceed in an iterative fashion and does not require hierarchical clustering of the training set. It uses both mean and variance of clusters in order to build the classifier, which is in contrast to CB-SVM as it uses the mean information only.

SVM-Perf [26] is a linear time solver for a formulation equivalent to the SVM. The authors in their paper show that the algorithm achieves generalization comparable to that of SVMs but with very less training time. However the SVM-Perf algorithm is not online in nature and needs to store the training dataset in memory, hence making it unsuitable for very large datasets. The experiments presented in this chapter confirm that the proposed clustering based scheme is comparable to SVM-Perf, both in training time as well as in accuracy, and is also a viable option in cases where datasets do not fit in memory.

3.2 Scalable Classification Formulation

This section presents the novel SOCP formulation for large-scale classification. Let Z_1 and Z_2 represent the random variables that generate the data points of the positive and negative classes respectively. Assume that the distributions of Z_1 and Z_2 can be modeled using mixture models, with component distributions having spherical covariances. Let k_1 be the number of components in the mixture model of positive class and k_2 be that in the negative class. $k = k_1 + k_2$ is the total number of clusters. Let $X_j,\ j = 1, \ldots, k_1$ represent the random variable generating the j^{th} component of the positive class and $X_j,\ j = k_1 + 1, \ldots, k$ represent that generating the $(j - k_1)^{th}$ component of the negative class. Let X_j have the second order moments $(\mu_j, \sigma_j^2 \mathbf{I})$. The probability density functions (pdfs) of Z_1 and Z_2 can be written as $f_{Z_1}(\mathbf{z}) = \sum_{j=1}^{k_1} \rho_j f_{X_j}(\mathbf{z}),\ f_{Z_2}(\mathbf{z}) = \sum_{j=k_1+1}^{k} \rho_j f_{X_j}(\mathbf{z})$ where, ρ_j are the mixing probabilities ($\rho_j \geq 0$, $\sum_{j=1}^{j=k_1} \rho_j = 1$ and $\sum_{j=k_1+1}^{j=k} \rho_j = 1$).

Any good clustering algorithm will correctly estimate the second order moments of the components. BIRCH is one such clustering algorithm, that scales well for large datasets. Given these estimates of second order moments, an optimal classifier that generalizes well must be built.

Let $\mathbf{w}^\top \mathbf{x} - b = 0$ be the discriminating hyperplane and $\mathbf{w}^\top \mathbf{x} - b = 1$, $\mathbf{w}^\top \mathbf{x} - b = -1$ be the corresponding set of supporting hyperplanes. The constraints $\mathbf{w}^\top Z_1 - b \geq 1$ and $\mathbf{w}^\top Z_2 - b \leq -1$ ensure that the training set error is low. Since Z_1 and Z_2 are random variables, the constraints cannot be satisfied always. Thus, we ensure that with high probability, the events $\mathbf{w}^\top Z_1 - b \geq 1$ and $\mathbf{w}^\top Z_2 - b \leq -1$ occur:[2]:

$$\begin{aligned} P(\mathbf{w}^\top Z_1 - b \geq 1) \geq \eta, \quad P(\mathbf{w}^\top Z_2 - b \leq -1) \geq \eta \\ Z_1 \sim f_{Z_1}, \qquad\qquad Z_2 \sim f_{Z_2} \end{aligned} \tag{3.1}$$

where, η is a user defined parameter. η lower bounds the classification accuracy. Since the distribution of Z_i is a mixture model, in order to satisfy (3.1), it is sufficient that each of the components/clusters satisfy the following chance-constraints:

$$\begin{aligned} P(\mathbf{w}^\top X_j - b \geq 1) \geq \eta, \quad & j = 1, \ldots, k_1 \\ P(\mathbf{w}^\top X_j - b \leq -1) \geq \eta, \quad & j = k_1 + 1, \ldots, k \\ X_j \sim f_{X_j}, \quad & j = 1, \ldots, k \end{aligned} \tag{3.2}$$

For $\eta \neq 0$ the constraints (3.2) are consistent only if the means of the components are linearly separable. Thus, in order to handle the case of outliers and almost linearly separable datasets, the chance-constraints in (3.2) can be relaxed using some slack variables (ξ_i) and suitably penalizing the relaxation. This leads to the following maximum margin classification formulation similar in spirit to SVMs:

[2] $Z_i \sim f_{Z_i}$ denotes Z_i has the pdf f_{Z_i}

$$\begin{aligned} \min_{\mathbf{w},b,\xi_j} \quad & \tfrac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{j=1}^{k}\xi_j \\ \text{s.t.} \quad & P(\mathbf{w}^\top X_j - b \geq 1 - \xi_j) \geq \eta,\ j = 1,\ldots,k_1, \\ & P(\mathbf{w}^\top X_j - b \leq -1 + \xi_j) \geq \eta,\ j = k_1+1,\ldots,k, \\ & \xi_j \geq 0,\ X_j \sim f_{X_j},\ j = 1,\ldots,k \end{aligned} \tag{3.3}$$

The objective function (3.3) minimizes $\|\mathbf{w}\|_2$ in order to achieve good generalization. C is a user defined regularization parameter. The constraints in the optimization problem (3.3) are chance-constraints and hence need to be written as deterministic constraints in order to be able to solve the formulation. Using theorem 2.1, it can be shown that the chance-constraints for positive class are satisfied if the following constraints hold:

$$\mathbf{w}^\top \mu_j - b \geq 1 - \xi_j + \kappa\ \sigma_j \|\mathbf{w}\|_2 \tag{3.4}$$

where, $\kappa = \sqrt{\frac{\eta}{1-\eta}}$. Similarly the set of constraints on the negative class can be obtained.

Let $y_j, j = 1,\ldots,k$ represent the labels of the components (clusters). Thus $y_j = 1$ for $j = 1,\ldots,k_1$ and $y_j = -1$ for $j = k_1+1,\ldots,k$. Using this notation, (3.3) can be written as the following deterministic optimization problem:

$$\begin{aligned} \min_{\mathbf{w},b,\xi_j} \quad & \tfrac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{j=1}^{k}\xi_j \\ \text{s.t.} \quad & y_j(\mathbf{w}^\top \mu_j - b) \geq 1 - \xi_j + \kappa\ \sigma_j\|\mathbf{w}\|_2,\ \xi_j \geq 0,\ j = 1,\ldots,k \end{aligned} \tag{3.5}$$

The formulation in (3.5) can be written in the following equivalent form:

$$\begin{aligned} \min_{\mathbf{w},b,\xi_j,t} \quad & \tfrac{1}{2}t^2 + C\sum_{j=1}^{k}\xi_j \\ \text{s.t.} \quad & y_j(\mathbf{w}^\top \mu_j - b) \geq 1 - \xi_j + \kappa\ \sigma_j\|\mathbf{w}\|_2,\ \xi_j \geq 0,\ j = 1,\ldots,k,\ \|\mathbf{w}\|_2 \leq t \end{aligned} \tag{3.6}$$

Now the constraints in (3.6) which involve $\|\mathbf{w}\|_2$ can be written as:

$$\begin{aligned} \frac{y_j(\mathbf{w}^\top \mu_j - b) - 1 + \xi_j}{\kappa\ \sigma_j} &\geq \|\mathbf{w}\|_2,\ j = 1, \ldots, k \\ t &\geq \|\mathbf{w}\|_2 \end{aligned}$$

Thus, the optimization problem (3.5) can be written in the following final form:

$$\begin{aligned} \min_{\mathbf{w}, b, \xi_j, t} \quad & \tfrac{1}{2}t^2 + C \textstyle\sum_{j=1}^{k} \xi_j \\ \text{s.t.} \quad & y_j(\mathbf{w}^\top \mu_j - b) \geq 1 - \xi_j + \kappa\ \sigma_j t,\ \xi_j \geq 0,\ j = 1, \ldots, k,\ \|\mathbf{w}\|_2 \leq t \end{aligned} \tag{3.7}$$

The scalable classification formulation (3.7), henceforth denoted by CBC-SOCP, is an SOCP with only one cone constraint. This problem can be solved using open source SOCP solvers like `SeDuMi` to obtain the optimal values of $\mathbf{w}$ and b. The overall classification algorithm can be summarized as follows:

- Using a scalable clustering algorithm cluster the positive and negative data points.
- Estimate the second order moments of all the clusters.
- Solve the optimization problem (3.7), using SOCP solvers. This gives optimum values of $\mathbf{w}$ and b.
- The label of a new data point $\mathbf{x}$ is given by $\text{sign}(\mathbf{w}^\top \mathbf{x} - b)$.

Observe that when $\sigma_{ij} = 0$, the standard SVM formulation and the present formulation are same. In other words, if each data point is considered to be a cluster, then both the formulations are exactly the same. However, the SVM formulation involves $2m$ linear inequalities whereas the proposed formulation involves only $2k$ inequalities. Thus, the new formulation is expected to scale very well to large datasets. The time-complexity of clustering algorithm like BIRCH is $O(m)$ and that of the optimization is independent of m. Thus, the overall algorithm is expected to have a training time of $O(m)$. Another key advantage is that the new classification scheme does not require to store training data in the memory.

3.3 Dual and Geometrical Interpretation

The constraints in (3.7) have an elegant geometric interpretation. In order to see this, consider the problem of classifying the points lying in the sphere centered at μ, with radius $\kappa\sigma$ (denote sphere by $B(\mu, \kappa\sigma)$) onto the positive side of the $\mathbf{w}^\top\mathbf{x}-b=1$ hyperplane (allowing for slack variables):

$$\mathbf{w}^\top\mathbf{x} - b \geq 1 - \xi,\ \forall\ \mathbf{x}\ \in\ B(\mu, \kappa\sigma) \tag{3.8}$$

THEOREM 3.1. *The constraints (3.8) are equivalent to the following cone constraint:*

$$\mathbf{w}^\top\mu - b \geq 1 - \xi + \kappa\ \sigma\|\mathbf{w}\|_2 \tag{3.9}$$

Proof. Geometrically, (3.8) says that all points that belong to $B(\mu, \kappa\sigma)$ must lie on the positive half space of the hyperplane $\mathbf{w}^\top\mathbf{x} - b = 1 - \xi$. This geometric picture (also see [44]) immediately shows that the constraint (3.8) can be satisfied just by ensuring that the point in $B(\mu, \kappa\sigma)$ which is nearest to the hyperplane $\mathbf{w}^\top\mathbf{x}-b=1-\xi$ lies on the positive half space. Thus (3.8), which actually represents infinite number of constraints, can be written as the following single constraint:

$$z \geq 1 - \xi,\ z = \min_{\mathbf{x}\ \in\ B(\mu,\kappa\sigma)} \mathbf{w}^\top\mathbf{x} - b \tag{3.10}$$

Finding the minimum distant point on a sphere to a given hyperplane is simple. Drop a perpendicular to the hyperplane from the sphere's center. The point at which the perpendicular intersects the sphere gives the minimum distant point ($\mathbf{x}^*$). Note that $\mathbf{x}^*$ is the optimum solution of (3.10). Using this geometrical argument, $\mathbf{x}^*$ can be calculated using: $\mathbf{x}^* - \mu = -\rho\mathbf{w},\ \mathbf{x}^*\ \in\ B(\mu, \kappa\sigma)$. This gives $\mathbf{x}^* = \mu - \frac{\kappa\sigma\mathbf{w}}{\|\mathbf{w}\|_2}$. Now, (3.8) is satisfied if $\mathbf{w}^\top\mathbf{x}^* - b \geq 1 - \xi$. Substituting the value of $\mathbf{x}^*$, (3.9) is got. This completes the proof. □

Observe that (3.9) has the same form as (3.4). Hence, geometrical interpretation of the constraints in (3.7) is to restrict the discriminating hyperplane to lie such that most of the spheres $B(\mu_j, \kappa\sigma_j)$ are classified correctly. Figure 3.1 shows this geometric picture. All the spheres in the figure expect the one at $(5, 5)$ satisfy the constraint with $\xi_j = 0$.

It is interesting to study the dual of the formulation (3.7). Using the dual norm definition, $\|\mathbf{w}\|_2 = \sup_{\|\mathbf{u}\|_2 \leq 1} \mathbf{u}^\top \mathbf{w}$ and Lagrange multiplier theory, the dual can be written as:

$$\begin{array}{ll} \min\limits_{\alpha_j, \lambda} & \frac{1}{2}(\lambda - \sum_j \kappa\sigma_j\alpha_j)^2 - \sum_j \alpha_j \\ \text{s.t.} & \|\sum_j \alpha_j y_j \mu_j\|_2 \leq \lambda,\ \sum_j \alpha_j y_j = 0,\ 0 \leq \alpha_j \leq C \end{array} \tag{3.11}$$

and the necessary and sufficient Karush-Kuhn-Tucker (KKT) conditions can be written as:

$$\sum_j \alpha_j y_j \mu_j = \lambda \mathbf{u},\quad \sum_j \alpha_j y_j = 0,\quad \alpha_j + \beta_j = C,\ t + \sum_j \kappa\sigma_j\alpha_j = \lambda$$

$$\alpha_j(1 - \xi_j + \kappa\sigma_j t - y_j(\mathbf{w}^\top \mu_j - b)) = 0,\ \beta_j\xi_j = 0$$

$$\lambda(\mathbf{w}^\top \mathbf{u} - t) = 0,\ \alpha_j \geq 0,\ \beta_j \geq 0,\ \lambda \geq 0,\ \|\mathbf{u}\|_2 \leq 1 \tag{3.12}$$

where $\alpha_j, \beta_j, \lambda$ are the Lagrange multipliers. Suppose $0 < \alpha_j < 1$ and $\lambda > 0$ then, from the KKT conditions it can be seen that $\xi_j = 0$, $\|\mathbf{w}\|_2 = t$ and $y_j(\mathbf{w}^\top \mu_j - b) = 1 + \kappa\sigma_j\|\mathbf{w}\|_2$. These conditions imply that the supporting hyperplanes are tangent to $B(\mu_j, \kappa\sigma_j)$. Extending the terminology used in case of SVMs, such spheres may be called as non-bound support spheres. Similarly the bounded support spheres can be defined as the spheres with $\alpha_j = 1$. Also, note that $\alpha_j = 1 \Rightarrow \xi_j > 0$. In figure 3.1, the spheres marked with 'o' are non-bound support spheres and hence are tangent to the supporting hyperplanes. Note that the dual involves only dot products of data points. This is because,

$$\|\sum_j \alpha_j y_j \mu_j\|_2 = \sqrt{\left(\sum_i \sum_j \alpha_i \alpha_j y_i y_j \mu_i^\top \mu_j\right)}$$

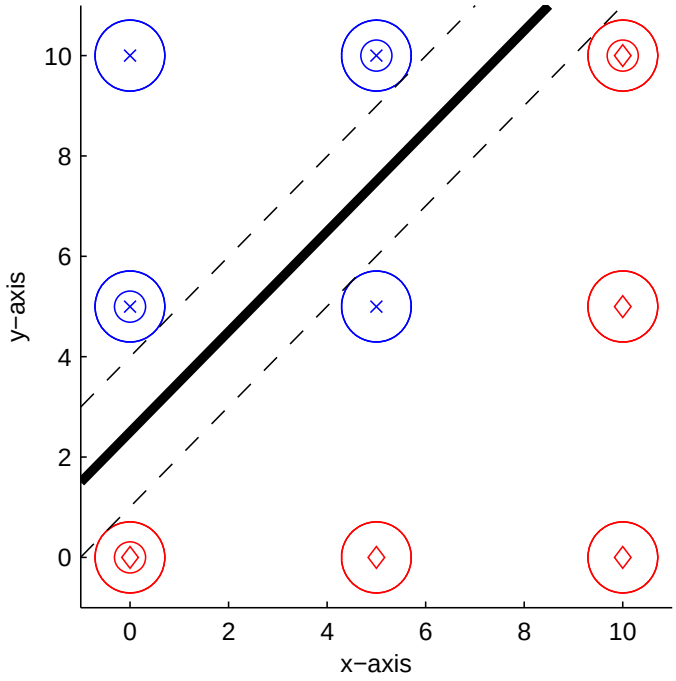

Figure 3.1: Illustration showing geometric interpretation of the constraints. Clusters marked with '×' have positive labels and those marked '◊' have negative labels. The radii of spheres are proportional to $\kappa\sigma_j$.

and the estimate of σ_j^2 is $\frac{1}{m_j}\sum_{k=1}^{m_j}(\mathbf{x}_k - \mu_j)^\top(\mathbf{x}_k - \mu_j)$ where, $\mathbf{x}_k$ are the m_j data points that belong to j^{th} cluster. Since the formulation (3.11) involves only the dot products of the data points, it can be extended to arbitrary feature spaces using Mercer kernels (see appendix B for details).

For linearly separable datasets, the following hard-margin variant of (3.7) can be written:

$$\begin{aligned} \min_{\mathbf{w},b} \quad & \tfrac{1}{2}\|\mathbf{w}\|_2^2, \\ \text{s.t.} \quad & y_j(\mathbf{w}^\top\mathbf{x}_j - b) \geq 1 + \kappa\sigma_j\|\mathbf{w}\|_2 \;\forall\; j \end{aligned} \tag{3.13}$$

Interestingly, the dual of the problem (3.13) turns out to be that of finding distance between the convex hulls formed by the negative and positive spheres ($B(\mu_j, \kappa\sigma_j)$). This is analogous to the case of SVMs, where dual is the problem of finding distance between the convex hulls formed by the negative and positive data points [7].

3.4 Numerical Experiments

This section presents experiments comparing testset accuracy and training time with the proposed classification scheme (denoted by **CBC-SeDuMi**[3]) and **SVM-Perf** [26], which is state-of-the-art linear time SVM algorithm. Datasets used in experiments are:

$\mathcal{D}_1$ Synthetic dataset with $m = 4,500,000$ and $n = 2$. Generated using 9 Gaussian distributions with $\sigma = 0.5$ and centers on a 3×3 square grid (fig. 3.1). Equal number of points $(500,000)$ were generated from each cluster. A testset ($m = 450,000$) was also generated using the same Gaussian distributions.

$\mathcal{D}_2$ Synthetic dataset with $m = 4,500,000$ and $n = 38$, such that projection of $\mathcal{D}_2$ onto the first two dimensions gives $\mathcal{D}_1$. Testset for $\mathcal{D}_2$ was also generated independently.

IDS KDD 1999 Intrusion Detection Dataset[4], $m = 4,898,430, n = 41$. The classification task is to build a network intrusion detector, a predictive model capable of distinguishing between "bad" connections, called intrusions or attacks, and "good" normal connections. This dataset has 7 categorical features and 3 of them take string values. Since the proposed classifier and the SVMs work for numerical data, these three features were removed from the training data. Hence, the final training data has 38 dimensions.

Web-Page This dataset[5] has $m = 49,749$ data points in $n = 300$ dimensions. The classification task is "Text categorization": classifying whether a web page belongs to a category or not.

IJCNN1 IJCNN1 dataset[6] has $m = 49,990$ data points in $n = 22$ dimensions.

All experiments were carried on Athlon64 Dual Core 4200 machines with 2GB memory. The results are shown in table 3.1 for the tuned set of parameters which gave

[3]Implementation can be downloaded from `http://mllab.csa.iisc.ernet.in/downloads/cbclassifier.html`.
[4]Training and testset available at `http://www.ics.uci.edu/~kdd/databases/kddcup99/kddcup99.html`
[5]Training and testset available at `http://research.microsoft.com/~jplatt/smo.html`
[6]Training and testset available at `http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary.html`

Table 3.1: Results on some large datasets, comparing the performance of **CBC-SeDuMi** and **SVM-Perf**.

Dataset	m	**Accuracy** %		**Total Time (sec)**	
		CBC-SeDuMi	**SVM-Perf**	**CBC-SeDuMi**	**SVM-Perf**
Web-page	49,749	97.40	98.77	5(2)	4(4)
IJCNN1	35,000	90.50	91.60	1(1)	3(2)
IDS	4,898,430	92.00	91.89	63(1)	100(30)
$\mathcal{D}_\mathbf{1}$	4,500,000	88.88	88.85	21(1)	43(23)
$\mathcal{D}_\mathbf{2}$	4,500,000	88.88	×	56(1)	×

the best testset accuracy for the respective classifier when trained with the full training data. A '×' mark in a table represents the failure of the corresponding classifier to complete training due to lack of memory (thrashing). The column "Total Time" shows the sum of loading, pre-processing, formulation solving times. Thus for **CBC-SeDuMi** $T_{total} = T_{clust} + T_{SOCP}$ and for **SVM-Perf** $T_{total} = T_{load} + T_{QP}$. The figures in the brackets represent the formulation solving time alone i.e. T_{SOCP} for **CBC-SeDuMi** and T_{QP} for **SVM-Perf**. The table shows that, in all cases, **CBC-SeDuMi** and **SVM-Perf** are comparable both in terms of training time and testset accuracy. However, since **CBC-SeDuMi** employs an online clustering algorithm for pre-processing, it makes only a single pass of the dataset and does not store the training data in memory. Thus though **SVM-Perf** failed to complete training with $\mathcal{D}_\mathbf{2}$, **CBC-SeDuMi** successfully completed training. The Figures 3.2, 3.3 and 3.4 summarize the scaling experiments done on the $\mathcal{D}_\mathbf{1}$, $\mathcal{D}_\mathbf{2}$ and **IDS** datasets. The figures show that both schemes scale almost linearly with training data size.

3.5 Summary

A classification method which is scalable to very large datasets has been proposed, using SOCP formulations. Assuming that the class conditional densities of positive and negative data points can be modeled using mixture models, the second order moments of the components of mixture are estimated using a scalable clustering algorithm like BIRCH. Using the second order moments, an SOCP formulation is proposed which ensures that

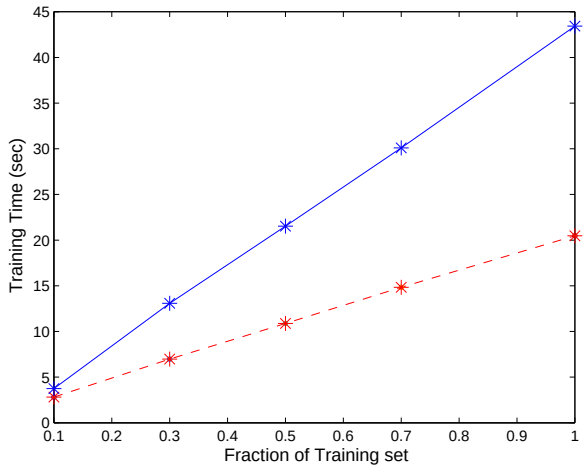

Figure 3.2: Graph showing scaling results on $\mathcal{D}_1$. Solid line represents **SVM-Perf** and dashed line **CBC-SeDuMi**.

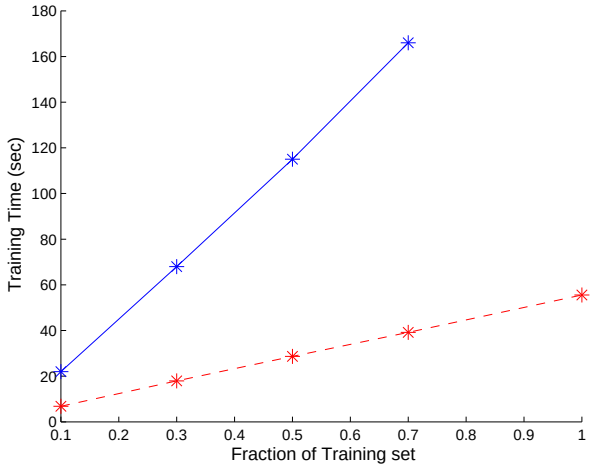

Figure 3.3: Graph showing scaling results on $\mathcal{D}_2$. Solid line represents **SVM-Perf** and dashed line **CBC-SeDuMi**.

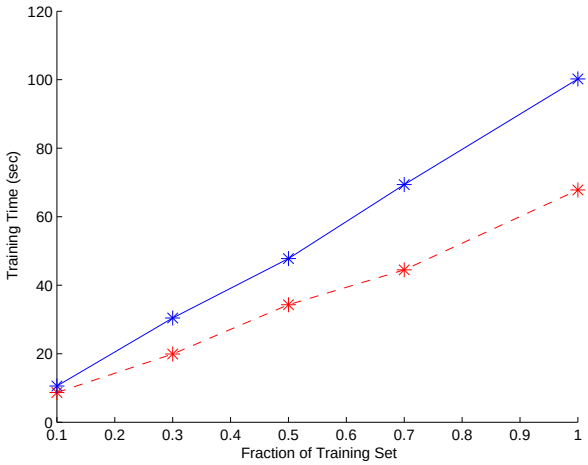

Figure 3.4: Graph showing scaling results on **IDS**. Solid line represents **SVM-Perf** and dashed line **CBC-SeDuMi**.

most of the clusters are classified correctly. The geometric interpretation of the formulation, is to classify spheres $B(\mu_j, \kappa\sigma_j)$ with as little error as possible. Experiments on synthetic and real world datasets show that the proposed method achieves good accuracy with $O(m)$ training time.

As pointed in section 3.3, the optimization formulation can be extended to feature spaces (appendix B provides the details). The appendix chapter also presents a fast iterative solver for the scalable SOCP formulation (3.7). Employing such a solver will result in further decrease of training time. Also, the formulation (3.7) uses estimated moments of clusters instead of the unknown, true moments, and hence is susceptible to estimation errors. The issue of making the formulation robust from such errors is discussed in chapter 6.

Chapter 4

Large-Scale Ordinal Regression and Focused Crawling

Abstract

This chapter extends the chance-constraint based, scalable classification approach to the important problem of large-scale ordinal regression[1]*. The chapter also shows how the formulation can be extended to feature spaces using the kernel trick. As in case of classification, the new OR scheme scales linearly with the number of training data points. Experiments on non-linear benchmark datasets show working of the kernelized formulation. Another contribution of the chapter is to pose focused crawling as a large-scale OR problem, solved efficiently using the new scalable scheme. This removes inefficiencies of the existing focused crawlers.*

Ordinal regression problems frequently occur in the areas of information retrieval, social science, personalized searches etc [14, 22, 23, 43]. Given a training dataset labeled with a set of ranks, the task of Ordinal Regression (OR) is to construct a function that predicts the rank of new data points. In contrast to metric regression problems, these ranks are of finite types and the metric distances between the ranks are not defined. Also the existence of the ordering information among the classes makes an OR problem different from a multiclass classification problem. Because of its wide applicability in

[1] *This work was presented at the* 24^{th} *International Conference on Machine Learning, 2007.*

ranking, there is considerable interest in solving large-scale OR problems. In this work, we consider the large margin formulation given by [13] as the baseline OR formulation.

Due to the increasing usage of Internet and growth of web, the need for fast training and prediction algorithms in the domains of Information retrieval and personalized search is increasing. Existing OR formulations require that the number of constraints in the formulation grow with the number of data points, m. In this chapter, a formulation which minimizes training error by employing chance-constraints for clusters rather than constraints for individual data points is presented. Since the number of clusters could be substantially smaller than the number of data points, the proposed formulation has better scaling properties. The moments of clusters can be estimated efficiently using an online algorithm like BIRCH and the formulation can be solved using generic SOCP solvers like `SeDuMi`. Experiments show that training time for the new OR scheme is far less than that with the baseline, even when generic SOCP solvers are employed. Scalability of the proposed OR scheme can be further improved by employing the fast solver presented in chapter 5.

A simple way of extending the proposed large-scale OR scheme to feature spaces is also presented. The number of support vectors with the kernelized formulation can at the maximum be k, the number of clusters, whereas that for the baseline is m. Another key advantage is that the kernelized scheme also scales linearly with m, even though it works with feature spaces. Hence the chance-constraint based OR scheme has potential to be exploited in cases where fast training and fast predictions are desired.

Focused crawling [11] is an efficient mechanism for finding web-pages relevant to a particular topic. The idea is to traverse and retrieve a part of the web which is relevant to a particular topic, starting from a seed set of topic relevant pages. Focused crawlers make efficient usage of network bandwidth, storage capacity and hence provide a viable mechanism for frequent updation of search engine indexes. They have also been useful in applications like distributed processing of the web.

In the past, various learning formulations for the problem of focused crawling were proposed and have shown good performance. However, the major drawback with them is

that they require topic taxonomy for constructing a set of topic-irrelevant (negative set) web-pages for training. This chapter shows that posing focused crawling as a large-scale ordinal regression problem avoids such problems and leads to efficient crawling schemes.

The outline of this chapter is as follows: in section 4.1, brief review of the past work on ordinal regression and focused crawling is presented. Section 4.2 presents the chance-constraint based OR formulation. In section 4.3, the proposed OR scheme is extended to feature spaces using the kernel trick. The methodology of focused crawling using OR is described in section 4.4. Section 4.5 details the experiments done on non-linear benchmark datasets comparing performance of the chance-constraint based and baseline OR schemes. The section also presents crawling experiments to show benefits of the OR-based focused crawler. Section 4.6 concludes the chapter with a brief summary.

4.1 Past Work

This section presents a brief review of the past work on ordinal regression (section 4.1.1) and the problem of focused crawling (section 4.1.2).

4.1.1 Ordinal Regression

Given a data set $\mathcal{D} = \{(\mathbf{x}_i{}^j, y_i) \mid \mathbf{x}_i^j \in \mathbb{R}^n, i = 1, \cdots, r,\ j = 1, \cdots, m_i\}$, where y_i is the rank of the data point, r is the number of ranks, m_i is the number of data points having rank 'i' and $m = \sum_{i=1}^{r} m_i$ is the total number of data points, the task of ordinal regression is to construct a function $f : \mathbb{R}^n \rightarrow \{1, \ldots, r\}$ such that $f(\mathbf{x}_i^j) = y_i$. Such formulations find ready appeal in many ranking applications [23]. The OR problem is very similar to the problem of multiclass classification. However the main difference is that the classes in case of OR are ranked. Hence a data point belonging to class '1' and misclassified as '2' is penalized less than when it is misclassified as class '3'. In case of multiclass classification all misclassifications are treated equally. Because of the ordinal relation among classes, several evaluation metrics exist for OR problems. Two of the important evaluation metrics are: mean absolute error (MAE) and mean zero-one

error (MZE). MAE is the average deviation of the predicted class from the true class, assuming the ordinal classes are consecutive integers. $MAE = \frac{1}{M}\sum_{i=1}^{M}|\hat{y}_j - y_j|$, where $\hat{y}_j$ are the predicted classes and y_j are the true classes. MZE is the fraction of incorrect predictions. In this work we use the MZE evaluation measure.

Several approaches exist to solve the problem of ordinal regression. However, in this thesis, we restrict ourselves to maximum margin approaches, which in general achieve good generalization. The work by Herbrich et.al [23] is one such approach which applies the maximum margin theory. However the formulation size is a quadratic function of the training set size, m. Shashua and Levin [43] extended the support vector formulation for ordinal regression by representing the r ordered classes as r consecutive intervals on the real line. However the problem with their formulation is that there are no specific constraints which imply the ordering among the classes. This omission may lead to wrong results in some unfortunate cases. Including the ordinal constraints on classes, Chu and Keerthi [13] proposed a large margin formulation which we consider as the baseline OR formulation in this thesis. In the following text, the baseline formulation is described in brief:

The baseline OR formulation finds a set of hyperplanes $\mathbf{w}^\top\mathbf{x} - b_i = 0,\ i = 1,\ldots,r-1$, which separate the data points belonging to the r ordinal classes in a maximum margin sense. One can define $b_0 = -\infty, b_r = \infty$ and constrain that the data points of class 'i' must lie between $\mathbf{w}^\top\mathbf{x} - b_{i-1} = 0$ and $\mathbf{w}^\top\mathbf{x} - b_i = 0$. In addition, constraints on the thresholds, b_i, are put in order to specify the ordinal relation among the classes ($b_i - b_{i-1} \geq 0,\ i = 2,\ldots,r-1$). The baseline OR formulation hence can be written as:

$$\begin{aligned}
\min_{\mathbf{w},\mathbf{b},\xi_i^j,\xi_i^{*j}} \quad & \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{i=1}^{r}\sum_{j=1}^{m_i}\ \xi_i^j + \xi_i^{*j} \\
\text{s.t.} \quad & \mathbf{w}^\top\mathbf{x}_i^j - b_i \leq -1 + \xi_i^j,\ \xi_i^j \geq 0, \\
& \mathbf{w}^\top\mathbf{x}_i^j - b_{i-1} \geq 1 - \xi_i^{*j},\ \xi_i^{*j} \geq 0,\ \forall\ i,j \\
& b_i - b_{i-1} > 0,\ i = 2,\ldots,r-1
\end{aligned} \tag{4.1}$$

where $\mathbf{b}$ is the vector containing $b_i, i = 1,\ldots,r-1$.

The above formulation can be shown to be equivalent to:

$$\begin{aligned} \min_{\mathbf{w},\mathbf{b},\xi_i^j,\xi_i^{*j}} \quad & \sum_{i=1}^{r}\sum_{j=1}^{m_i} \xi_i^j + \xi_i^{*j} \\ \text{s.t.} \quad & \mathbf{w}^\top \mathbf{x}_i^j - b_i \leq -1 + \xi_i^j,\ \xi_i^j \geq 0,\ \xi_i^{*j} \geq 0, \\ & \mathbf{w}^\top \mathbf{x}_i^j - b_{i-1} \geq 1 - \xi_i^{*j},\ \forall\ i, j,\ \|\mathbf{w}\|_2 \leq W, \\ & b_i - b_{i-1} > 0, i = 2, \ldots, r-1 \end{aligned} \tag{4.2}$$

The Quadratic Program (QP) in (4.1) is similar to the SVM formulation and the SOCP (4.2) is similar to the generalized optimal hyperplane formulation [46]. However both the formulations are equivalent. The standard SVM formulation has gained popularity over its SOCP counterpart primarily because of the existence of fast SVM solvers like SMO. The key advantage of the SOCP formulation is interpretation of the parameter W. $\frac{2}{W}$ is the lower bound on margin. However such an interpretation for the parameter C in case of the QP formulation does not exist. In this work, the SOCP in (4.2) is considered as the baseline OR formulation.

The baseline formulation size is $2m+n+r-1$ and has $O(m+r)$ linear inequalities. The problem size and number of inequalities can be drastically reduced by employing chance-constraints for clusters in training data rather than having constraints for individual data points. Exploiting this, a novel chance-constrained based, large-scale OR scheme is proposed. The proposed scheme can also be extended to non-linear feature spaces using the kernel trick.

4.1.2 Focused Crawling

Focused crawling is an efficient resource discovery system for the web [11]. The aim of any focused crawler is to start from a *seed* set of pages relevant to a given topic and traverse specific links to collect pages about the topic without fetching pages unrelated to the topic. The fraction of relevant pages fetched is called the harvest rate [10]. In other words, if N denotes the total number of pages crawled and N_R the number of relevant

pages, then the harvest rate is defined as $\frac{N_R}{N}$. Higher the harvest rate, better the focused crawler. The first focused crawler, developed by Chakrabarti et.al [11], had three main components – web crawler, classifier and distiller. The classifier and distiller are used to define the strategy of the crawl. An existing document taxonomy is employed to define the topics of interest and irrelevant topics. Classifiers are learned at each internal node of the tree, which give the probability of a web-page belonging to a particular topic. Since probability is assigned to a web-page and all URLs in the page have equal priority in the crawl. The URLs to be crawled are queued and fetched according to their priority. The distiller improves performance of the crawl by prioritizing the URLs in hubs [29]. An improved variant was proposed in [10] which prioritizes the URLs within a web-page by using a classifier called the apprentice. Intelligent crawling [1] is a method that allows users to specify arbitrary predicates for measuring relevance and uses reinforcement based learning.

The major drawback with existing focused crawlers is the use of topic taxonomy. As mentioned above, classifiers are trained at each internal node of the topic taxonomy, using the corresponding topic, sub-topic documents as positive examples and the others as negative examples. Because of this, the negative class is too diverse. In this work, we take an alternate approach where the link structure in the web is exploited to define the degree of topic relevance. Focused crawling is posed as an OR problem, where ordinal classes represent the degree of relevance. This avoids the need for topic taxonomy and related issues.

Diligenti et.al. [16] proposed a related work, where a context graph is created using link information in the web. However, the problem of focused crawling is posed as a multiclass classification problem. During a crawl, the web-pages assigned to a particular class are preferred over the web-pages assigned to other classes. Thus inherently it is assumed that the classes are ranked. But the ranking information is not considered during training of the multiclass classifier. The present method attempts to pose the problem as a ranking problem, and OR is employed; it being more suited for ranking than multiclass classification.

4.2 Large-Scale OR Formulation

This section presents the scalable, chance-constraint based OR scheme. Let Z_i be the random variable that generates the data points of rank 'i'. Assume that the distributions of Z_i can be modeled using mixture models. Let k_i be the number of components of Z_i where each component distribution has spherical covariance. Let $X_i^j,\ j = 1, \cdots, k_i$ be the random variable generating the j^{th} component of Z_i whose second order moments are given by $(\mu_i^j, {\sigma_i^j}^2 I)$. Given these estimates of second order moments, an optimal regressor that generalizes well needs to be built.

As mentioned above, the data points that belong to class 'i' must lie between the hyperplanes $\mathbf{w}^\top \mathbf{x} - b_{i-1} = 0$ and $\mathbf{w}^\top \mathbf{x} - b_i = 0$ with high probability. This can be mathematically expressed as: $P(\mathbf{w}^\top Z_i - b_i \leq -1 + \xi_i^j) \geq \eta,\ P(\mathbf{w}^\top Z_i - b_{i-1} \geq 1 - \xi_i^{*j}) \geq \eta$ where η is user defined parameter. η lower bounds the classification accuracy. Following the arguments given in section 3.2, and using the formulation (4.2), one can derive the following large margin OR formulation (henceforth denoted by CBOR-SOCP):

$$\begin{aligned}
\min_{\mathbf{w},\mathbf{b},\xi_i^j,\xi_i^{*j}} \quad & \sum_{i=1}^{r}\sum_{j=1}^{k_i} \xi_i^j + \xi_i^{*j} \\
\text{s.t.} \quad & \mathbf{w}^\top \mu_i^j - b_i \leq -1 + \xi_i^j - \kappa \sigma_i^j W, \\
& \mathbf{w}^\top \mu_i^j - b_{i-1} \geq 1 - \xi_i^{*j} + \kappa \sigma_i^j W, \\
& \xi_i^j \geq 0,\ \xi_i^{*j} \geq 0,\ \forall\ i, j,\ \|\mathbf{w}\|_2 \leq W, \\
& b_i - b_{i-1} > 0,\ i = 2, \ldots, r-1
\end{aligned} \tag{4.3}$$

where $\kappa = \sqrt{\frac{\eta}{1-\eta}}$. The ordinal regression formulation (4.3) is an SOCP and can can be solved using generic SOCP solvers like `SeDuMi` to obtain the optimal values of $\mathbf{w}$ and $\mathbf{b}$. Note that the number of constraints for each rank 'i' in (4.3) is k_i, compared to m_i in (4.1). Thus the CBOR-SOCP formulation scales better than the baseline formulation (4.1). The overall training scheme, **CBOR-SeDuMi**, is to cluster the data points using any online clustering algorithm like BIRCH, which provides second order moment estimates of clusters and then solve the SOCP problem (4.3) using `SeDuMi`.

4.3 Feature Space Extension

This section extends the proposed OR formulation (4.3) to feature spaces. As discussed in section 3.3, the geometric interpretation of the inequalities in (4.3) is that of separating spheres centered at μ_i^j and radius $\kappa\sigma_i^j$, in a maximum margin sense. Now suppose that a non-linear mapping, ϕ, maps the means μ_i^j to $\phi(\mu_i^j)$. Assume the mapping has the property that "closer data points remain close and farther data points remain far". The mapping implicitly achieved by Gaussian kernel $K(\mathbf{x}_1, \mathbf{x}_2) = \exp\{-\zeta\|\mathbf{x}_1 - \mathbf{x}_2\|_2^2\}$, where ζ is the Gaussian kernel parameter, is in fact one such mapping. Though the following discussion holds for any such mapping, we restrict ourselves to the case of Gaussian kernel, in order to keep the equations simple. One can easily verify that if $\|\mathbf{x} - \mu_i^j\|_2 \leq \kappa\sigma_i^j$, then $\|\phi(\mathbf{x}) - \phi(\mu_i^j)\|_2 \leq r_i^j$ where $r_i^j = \sqrt{2\left(1 - \exp\left\{-\zeta\left(\kappa\sigma_i^j\right)^2\right\}\right)}$. Using this, one can rewrite the OR formulation (4.3) as:

$$\begin{aligned}
\min_{\mathbf{w},\mathbf{b},\xi_i^j,\xi_i^{*j}} \quad & \sum_{i=1}^{r}\sum_{j=1}^{m_i} \xi_i^j + \xi_i^{*j} \\
\text{s.t.} \quad & \mathbf{w}^\top\phi(\mu_i^j) - b_i \leq -1 + \xi_i^j - r_i^j W, \\
& \mathbf{w}^\top\phi(\mu_i^j) - b_{i-1} \geq 1 - \xi_i^{*j} + r_i^j W, \\
& \xi_i^j \geq 0,\ \xi_i^{*j} \geq 0,\ \forall\ i, j,\ \|\mathbf{w}\|_2 \leq W, \\
& b_i - b_{i-1} > 0,\ i = 2, \ldots, r-1
\end{aligned} \tag{4.4}$$

The dual of the above formulation turns out to be

$$\begin{aligned}
\max_{\alpha,\alpha^*,\rho} \quad & \mathbf{d}^\top(\alpha + \alpha^*) - \rho W \\
\text{s.t.} \quad & \sqrt{(\alpha^* - \alpha)^\top\mathbf{K}(\alpha^* - \alpha)} \leq \rho, \\
& 0 \leq \alpha \leq 1, 0 \leq \alpha^* \leq 1, \\
& s_i^* \leq s_i,\ \forall\ i = 1, \ldots, r-2, s_{r-1}^* = s_{r-1}
\end{aligned} \tag{4.5}$$

where, $\alpha = \{\alpha_1^1, \ldots, \alpha_1^{m_1}, \ldots, \alpha_r^1, \ldots, \alpha_r^{m_r}\}^\top$ and α_i^j are the Lagrange multipliers for the inequalities $\mathbf{w}^\top\phi(\mu_i^j) - b_i \leq -1 + \xi_i^j - r_i^j W$. Similarly α^* is the vector containing the

Lagrange multipliers α_i^{*j} for the inequalities $\mathbf{w}^\top\phi(\mu_i^j) - b_{i-1} \geq 1 - \xi_i^{*j} + r_i^j W$ and ρ is the Lagrange multiplier for the inequality $\|\mathbf{w}\|_2 \leq W$. $\mathbf{d}$ is the vector containing $1 + r_i^j W$ as its entries, $\mathbf{K}$ is the matrix containing dot products of $\phi(\mu_i^j)$ with each other, $s_i = \sum_{k=1}^{i}\sum_{j=1}^{m_k}\alpha_k^j$ and $s_i^* = \sum_{k=2}^{i+1}\sum_{j=1}^{m_k}\alpha_k^{*j}$ (please refer Appendix C for a short derivation of the dual).

Parameters of the formulation (4.5) are $\mathbf{K}, \mathbf{d}, W$. The parameter W, which is an upper bound on $\|\mathbf{w}\|_2$, is user given. Parameters $\mathbf{K}, \mathbf{d}$ can be computed using the dot products of means, variances in input space and the Gaussian kernel parameter, ζ. The i^{th} decision function $f_i(\mathbf{x}) = \mathbf{w}^\top\mathbf{x} - b_i$ can be written as (assuming non-trivial case $\rho \neq 0$):

$$f_i(\mathbf{x}) \equiv \mathbf{w}^\top\mathbf{x} - b_i = \frac{W}{\rho}\mathbf{K}_{\mathbf{x}}^\top(\alpha^* - \alpha) - b_i \tag{4.6}$$

The class to which a new test example $\mathbf{x}$ belongs to is given by $\text{argmax}_{i=0,\ldots,r-1}\left(I_{\{f_i(\mathbf{x})>0\}}\right)+1$. The discussion shows that solving (4.5) and predicting labels of new examples involve dot products of means and variance information only. Thus training and prediction of labels with the proposed OR scheme can be done in any feature space using the kernel trick. The key advantage is that the training time still scales linearly with the number of data points.

Equation (4.6) shows that the decision functions depend on the set of training examples for which $\alpha_i^j - \alpha_i^{*j} \neq 0$. Extending the terminology of SVMs, such data points can be named as "support vectors". Clearly the maximum number of support vectors for the new OR formulation is the no. clusters, whereas that for the baseline OR formulation (4.2) is the no. training data points. Thus the proposed OR formulation can be applied in cases where fast prediction algorithms are desired.

4.4 Focused Crawling as an OR problem

In this section, we pose focused crawling as an OR problem and discuss its merits. A focused crawler usually consists of the following basic components: a page fetcher, a priority queue and a scoring function. The fetcher gets the web-page pointed to by

the URL at the head of the queue. Once the page is fetched, the scoring function determines the relevance of the web-page and the likelihood/probability of the links on the page leading to a web-page of interest. It then inserts these links in the priority queue based on this likelihood/probability. Most of the existing focused crawlers use a topic hierarchy to define the topics of interest and irrelevant topics. Classifiers are learned at each internal node of the hierarchy, which give the probability of the web-page belonging to the node (topic). Also, with such a model, for any topic, the negative training set becomes very large and diverse, which makes the classifiers difficult to construct.

To overcome these problems, the proposed crawling strategy uses the inherent link structure of the web for constructing a training set from the given seed set of relevant pages. Any web-page is semantically closer to web-pages hyperlinked with it, than to web-pages which are not [15, 21]. For instance, pages which are one link away are semantically closer to seed pages than pages that are two or three links away. Thus the web is modeled as a layered graph, with the pages relevant to the seed pages/topic forming the first layer. Similarly, pages which have links to topic pages form the second layer, pages having links to these second layer pages forming the third layer and so on. The idea is to crawl the links in seed set and rank the fetched web-pages based on their link distance to the seed pages. Using this set as training data, an ordinal regressor is trained, which would predict the degree of relevance of a web-page to the given topic. The overall crawling strategy is same as the baseline, except that an ordinal regressor is trained in place of a classifier. During the actual crawl, links on level i pages are given higher priority than those on level $i + 1$. That is, links on a page are prioritized depending on how quickly these links would lead to topic page, where time is measured in terms of number of links that need to be crawled.

4.5 Numerical Experiments

The section presents two sets of experiments: the first set of experiments look at the performance and scalability of the **CBOR-SeDuMi** scheme (section 4.2). The second

set of experiments compare the performance of the proposed OR based crawler and the baseline crawler.

4.5.1 Scalability of CBOR-SeDuMi

In this section, scaling experiment results on two large, non-linear benchmark datasets, comparing scalability of **CBOR-SeDuMi** and the baseline solved using the SMO algorithm [13] (denoted by **SMO-OR**), are presented. The two benchmark datasets used are California Housing dataset[2] and Census dataset[3]. California Housing dataset has $20,640$ data points in 8 dimensions and Census dataset has $22,784$ data points in 16 dimensions. The benchmark datasets were randomly partitioned into training and test datasets of different sizes in order to compare scalability of the algorithms. In all cases the results shown are for the tuned parameters that gave best accuracy on the test set. In case of **CBOR-SeDuMi** and **SMO-OR**, the Gaussian kernel always performed better than the linear kernel since the benchmark datasets are essentially non-linear. Table 5.1 summarizes the scaling experiment results. Note that the training time for **CBOR-SeDuMi** is $T_{CBOR-SeDuMi} = T_{clust} + T_{SOCP}$, where T_{clust} is the time required to cluster the data using BIRCH and T_{SOCP} is the time required to solve (4.5) using `SeDuMi`. A '$\times$' in the table implies that the corresponding algorithm failed to complete training, due to lack of memory. Results show that the average error rate on the test sets, in case of both datasets, with **CBOR-SeDuMi** and **SMO-OR** are comparable. In all the cases where `SeDuMi` completed training, **CBOR-SeDuMi** has very less training time than **SMO-OR**. This can be attributed to the fact that size of the optimization problem solved by **CBOR-SeDuMi** is very small when compared to **SMO-OR**. Hence even though `SeDuMi` is a generic solver and SMO is specialized to solve (4.2), the training time for **CBOR-SeDuMi** is less. However, in cases where the number of clusters themselves is large (around 2000), `SeDuMi` failed to converge due to lack of memory. In chapter 5, a specialized solver for the OR formulation (4.5), which outperforms `SeDuMi` in terms of

[2] `http://lib.stat.cmu.edu/datasets/`

[3] `http://www.liacc.up.pt/~ltorgo/Regression/DataSets.html`

Table 4.1: Comparison of training times (in sec) with **CBOR-SeDuMi** and **SMO-OR** on benchmark datasets. The test set error rate is given in brackets. (CH-California Housing, CS-Census datasets).

	S-Size	**SMO-OR**	**CBOR-SeDuMi**
		sec (err)	sec (err)
CS	5,690	893 (.128)	20.4 (.109)
	11,393	5281.6 (.107)	108.8 (.112)
	15,191	9997.5 (.107)	271.1 (.108)
	22,331	×	435.7 (.119)
CH	10,320	551.9 (.619)	112 (.623)
	13,762	1033.2 (.616)	768.8 (.634)
	15,482	1142 (.617)	×
	17,202	1410 (.617)	×

Table 4.2: Datasets: Categories and training set sizes

Category	**Seed**	**1**	**2**	**3**	**4**
NASCAR	1705	1944	1747	1464	1177
Soccer	119	750	1109	1542	3149
Cancer	138	760	895	858	660
Mutual Funds	371	395	540	813	1059

solving time, is presented. Employing such a solver boosts the scalability of the proposed OR scheme.

4.5.2 Performance of Focused Crawler

This section presents experiments comparing harvest rates of the baseline and OR-based crawlers. Nalanda iVia crawler[4], which implements the baseline [11], was used in the experiments. It employs a logistic regression based binary classifier.

The topics chosen for crawl were mutual funds, NASCAR, soccer, and cancer. These topics posed challenge to the baseline crawler and are considered to be 'difficult'. A set of seed pages was collected for each topic, from sources like Wikipedia and links returned by general purpose search engines. Using the seed pages, training set for the

[4] `http://ivia.ucr.edu/`

Table 4.3: Table comparing harvest rates of **BL-Crawl** and **OR-Crawl**. #R(#I) is the number of relevant (irrelevant) web pages crawled by **BL-Crawl**. It indicates the difficulty in crawling these categories.

Dataset	#R/#I	**BL-Crawl**	**OR-Crawl**
NASCAR	11530/19646	0.3698	0.6977
Soccer	10167/9131	0.3400	0.4952
Cancer	6616/12397	0.4714	0.5800
Mutual Fund	9960/10992	0.5260	0.5969

ordinal regressor, with 5 levels of relevance, was created (see section 4.4). Each web-page was represented by a vector of 4000 features. The sizes of seed and training sets are shown in the Table 4.2. Level '0' (seed set pages) represents the most relevant pages and level '4' represents the least relevant pages for the given topic.

The **CBOR-SeDuMi** scheme was then used to train the ordinal regressor efficiently. The parameters η, ζ and W were tuned using grid search on a subset of the training set (validation set). During the crawling phase, for each newly crawled web-page, the relevance level was predicted using the ordinal regressor. If a page was marked as level 0, 1, 2, or 3, the links from the pages were added to the priority queue; any page marked as level 4 was discarded.

Table 4.3 compares harvest rates with the proposed OR-based focused crawler (**OR-Crawl**) and the baseline crawler (**BL-Crawl**). For harvest rate calculation, the number of relevant pages crawled, N_R, was taken to be the number of web-pages belonging to levels 0, 1, or 2. As seen, **OR-Crawl** performs better than **BL-Crawl** even on categories which are considered to be difficult.

4.6 Summary

A novel chance-constraint based OR scheme, which works with feature spaces and scales linearly with the training set size was presented. The proposed formulation involves few support vectors and hence has potential to be employed in applications where fast

predictions are desired. An ordinal regression formulation for the focused crawling problem which removes the need for a topic taxonomy was presented. The chance-constraint based OR scheme was used to solve the formulation efficiently. As seen from experimental results, the OR-based crawler performs better than the baseline even on topics which are considered difficult.

Chapter 5

Fast Solver for Scalable Classification and OR Formulations

Abstract

This chapter presents a fast iterative algorithm to solve the large-scale OR formulation (4.3)[1]*. The algorithm exploits the formulation's special structure, that it is a single cone constrained SOCP, and performs a projected gradient descent. The iterative algorithm scales very well when compared to generic SOCP solvers and is very easy to implement. The algorithm can also be derived for the scalable classification formulation (3.7), since it is again an instance of single cone constrained SOCP.*

In chapters 3, 4 it is shown that the chance-constraint based learning formulations scale better than traditional large margin formulations and also achieve good generalization. The scalability of such schemes can be owed to the fact that the no. clusters in the training data will be very less compared to the no. training data points. However the experiments in section 4.5.1 showed that when the no. clusters themselves is large, generic SOCP solvers like `SeDuMi` fail. The main contribution of this chapter is a fast iterative algorithm, **CBOR-Iter**, that can efficiently handle large number of clusters and in-turn very large number of data points. **CBOR-Iter** exploits the fact that the large-scale OR formulation (4.3) has only one cone constraint and efficiently solves the

[1] *This work was presented at the* 24^{th} *International Conference on Machine Learning, 2007.*

dual formulation. This removes the necessity of employing generic optimization software like `SeDuMi`.

Erdougan and Iyengar [17] show that algorithms which exploit the specialty of having single cone constraint perform better than generic SOCP solvers. The authors in their paper derive an active set method which is more efficient than `SeDuMi` in solving single cone constrained SOCPs. In this work, we further exploit the special structure of the formulation (4.3) and derive a fast, easy to implement, projected gradient based algorithm, which is similar in spirit to the SMO algorithm. Experimental results show that the fast algorithm outperforms `SeDuMi` in solving (4.3).

The fast iterative algorithm presented here can also be derived for the scalable classification formulation (see appendix B). This is because both the formulations are similar in structure.

5.1 Large-Scale OR Solver

This section presents a fast iterative solver for the dual of the chance-constraint based OR formulation (4.5). The constraints in (4.5) imply a lower bound on ρ and objective implies minimizing ρ. Hence at optimality, $\rho = \sqrt{(\alpha^* - \alpha)^\top \mathbf{K}(\alpha^* - \alpha)}$. Using this condition, the dual can be re-written as:

$$\begin{aligned} \min_{\alpha,\alpha^*} \quad & W\sqrt{(\alpha^* - \alpha)^\top \mathbf{K}(\alpha^* - \alpha)} - \mathbf{d}^\top (\alpha + \alpha^*) \\ \text{s.t.} \quad & 0 \leq \alpha \leq 1,\ 0 \leq \alpha^* \leq 1 \\ & s_i^* \leq s_i,\ \forall\ i = 1, \ldots, r-2,\ s_{r-1}^* = s_{r-1} \end{aligned} \tag{5.1}$$

From the KKT conditions (C.2) and the optimal value of ρ (assuming $\rho \neq 0$), one can calculate the value of $\mathbf{w}$ as $\frac{W}{\rho}\sum_{i=1}^{r}\sum_{j=1}^{m_i}\left(\alpha_i^{*j} - \alpha_i^j\right)\phi(\mu_i^j)$. The decision function can be written as:

$$f(\mathbf{x}) \equiv \mathbf{w}^\top \mathbf{x} - b = g(\mathbf{x}) - b,\ g(\mathbf{x}) = \frac{W}{\rho}\mathbf{K}_{\mathbf{x}}^\top (\alpha^* - \alpha) \tag{5.2}$$

where $\mathbf{K_x}$ is the vector of dot products of $\phi(\mathbf{x})$ and $\phi(\mu_i^j)$. Thus neither for solving the dual (4.5), nor for calculating $f(\mathbf{x})$, ϕ is explicitly needed; dot products are enough. The optimal values of $b_1, \ldots, b_{r-1}$ can be computed using the KKT conditions (C.2):

$$\begin{aligned}
\alpha_i^j = 0 \quad & g(\mu_i^j) + 1 + r_i^j W \leq b_i \\
0 < \alpha_i^j < 1 \quad & g(\mu_i^j) + 1 + r_i^j W = b_i \\
\alpha_i^j = 1 \quad & g(\mu_i^j) + 1 + r_i^j W \geq b_i \\
\alpha_{i+1}^{*j} = 0 \quad & g(\mu_{i+1}^j) - 1 - r_i^j W \geq b_i \\
0 < \alpha_{i+1}^{*j} < 1 \quad & g(\mu_{i+1}^j) - 1 - r_i^j W = b_i \\
\alpha_{i+1}^{*j} = 1 \quad & g(\mu_{i+1}^j) - 1 - r_i^j W \leq b_i
\end{aligned} \tag{5.3}$$

Let ${b_{low}}^i, {b_{up}}^i$ denote the greatest lower bound, least upper bound on b_i. Hence we have the conditions ${b_{low}}^i \leq b_i \leq {b_{up}}^i, i = 1, \ldots, r-1$. The KKT conditions (C.2) also indicate that for $i = 2, \ldots, r-1$, $b_{i-1} \leq b_i$ and if $s_{i-1} > s_{i-1}^*$ then $b_{i-1} = b_i$. Thus the overall optimality conditions can be written as ${B_{low}}^i \leq b_i \leq {B_{up}}^i$ where

$${B_{low}}^i = \begin{cases} \tilde{B}_{low}^{i+1} & \text{if } s_i > s_i^* \\ \tilde{B}_{low}^i & \text{otherwise} \end{cases} \tag{5.4}$$

and

$${B_{up}}^i = \begin{cases} \tilde{B}_{up}^{i-1} & \text{if } s_{i-1} > s_{i-1}^* \\ \tilde{B}_{up}^i & \text{otherwise} \end{cases} \tag{5.5}$$

and $\tilde{B}_{low}^i = \max\{{b_{low}}^k : k = 1, \ldots, i\}$, $\tilde{B}_{up}^i = \min\{{b_{up}}^k : k = i, \ldots, r-1\}$. Note that b_{low}, b_{up} represent the conditions at every hyperplane due to neighboring class data points; whereas B_{low}, B_{up} represent the conditions over all hyperplanes.

The proposed **CBOR-Iter** algorithm starts with some feasible solution. Then at every iteration, ${B_{low}}^i, {B_{up}}^i \ \forall \ i$ are calculated. If ${B_{low}}^i \leq {B_{up}}^i \ \forall \ i$, then the optimal solution is found and the algorithm terminates. Else the index i for which ${B_{low}}^i \leq {B_{up}}^i$ is most violated is calculated: $I = \arg\max_i\{i : {B_{low}}^i - {B_{up}}^i > 0\}$. Using (5.4) and

(5.5) the maximum KKT violating pair can be calculated. Now the following cases exist: **Case 1** The most violating pair is α_p^{jp} and α_q^{jq}, **Case 2** The most violating pair is α_p^{*jp} and α_q^{*jq}, **Case 3** The most violating pair is α_p^{jp} and α_q^{*jq}, **Case 4** The most violating pair is α_p^{*jp} and α_q^{jq}, where $p \leq q$. The equality constraint $s_{r-1} = s_{r-1}^*$ must hold. So for Case 1,2 one can update the variables by adding $\Delta\alpha$ to jp^{th} variable and subtracting $\Delta\alpha$ from jq^{th} variable. In Case 3,4 both variables must be incremented by $\Delta\alpha$ ($\Delta\alpha$ can also take negative values).

Now let $G_1 \equiv W\sqrt{(\alpha^* - \alpha)^\top \mathbf{K}(\alpha^* - \alpha)} - \mathbf{d}^\top(\alpha + \alpha^*)$ denote the dual objective with current values of α, α^*. Let G_2 denote the value of dual objective after appropriately incrementing the variables α, α^* with $\Delta\alpha$. We wish to find that value of $\Delta\alpha$ for which $G_2 - G_1$ is minimized. This can be written as the following 1-d minimization problem:

$$\begin{aligned} \min_{\Delta\alpha} \quad & \sqrt{a(\Delta\alpha)^2 + 2b(\Delta\alpha) + c} - e\Delta\alpha \\ \text{s.t.} \quad & lb \leq \Delta\alpha \leq ub \end{aligned} \tag{5.6}$$

where $a = W^2(\mathbf{K}(jp, jp) - 2\mathbf{K}(jp, jq) + \mathbf{K}(jq, jq))$, $b = W^2 l_1((\mathbf{K}_{jq} - \mathbf{K}_{jp})^\top(\alpha^* - \alpha))$, $c = W^2(\alpha^* - \alpha)^\top \mathbf{K}(\alpha^* - \alpha)$ and $e = (\mathbf{d}(jp) - l_2\mathbf{d}(jq))$. The values of l_1, l_2 depend on the Case to which update belongs to. $l_1 = 1$ for Case 1,3 and $l_1 = -1$ for Case 2,4. $l_2 = 1$ for Case 1,2 and $l_2 = -1$ for Case 3,4. lb, ub denote the tightest lower and upper bounds on $\Delta\alpha$ got from the inequality constraints in (5.1).

The optimum value of $\Delta\alpha$ that minimizes (5.6) is given by

$$\Delta\alpha = \begin{cases} \left[\dfrac{e\sqrt{\frac{ac-b^2}{a-e^2}} - b}{a}\right]_{lb}^{ub} & \text{if } ac - b^2 > 0, a - e^2 > 0 \\ \left[\frac{-b}{a}\right]_{lb}^{ub} & \text{if } ac - b^2 = 0, a - e^2 > 0 \\ ub & \text{if } e - \sqrt{a} \geq 0 \\ lb & \text{if } e + \sqrt{a} \leq 0 \end{cases}$$

where $\Delta\alpha]_{lb}^{ub}$ denotes $\max(lb, \min(ub, \Delta\alpha))$. Once the optimum value of $\Delta\alpha$ is calculated, then the values of α and α^* are updated accordingly and the procedure is repeated in the next iteration.

The **CBOR-Iter** algorithm can be summarized as follows:

1. Initialize α and α^* with some non-trivial, feasible values.
2. Calculate $B^i_{low}, B^i_{up}\ \forall\ i$. If KKT conditions are satisfied i.e., $B^i_{low} \leq B^i_{up}\ \forall\ i$ then terminate, else continue.
3. Identify the maximum KKT violating pair using (5.4) and (5.5).
4. Solve (5.6) to get the optimal value of $\Delta\alpha$. Update Lagrange multipliers of the maximum KKT violating pair and repeat step 2.

5.2 Numerical Experiments

This section supplements the experiments presented in section 4.5. The previous experiments compared **CBOR-SeDuMi** and **SMO-OR**. Here we present results with the clustering based OR formulation (5.1) solved using the fast iterative solver, **CBOR-Iter**. Note that the training time for **CBOR-Iter** is $T_{CBOR-Iter} = T_{clust} + T_{SMO}$ where T_{clust} is the time required to cluster the data using BIRCH and T_{SMO} is the time required to solve the dual (5.1). Table 5.1 compares the three methods **CBOR-Iter**, **SMO-OR** and **CBOR-SeDuMi**. A '$\times$' in the table implies that the corresponding algorithm failed to complete training, due to lack of memory. The results clearly show that the training time with **CBOR-Iter** is very less when compared to **SMO-OR** and **CBOR-SeDuMi**. Also the average error rate on the test sets with **CBOR-Iter** on California Housing dataset is 0.6221 and on Census dataset is 0.1122. These are comparable to the average error rates, 0.6184 and 0.1172, given by **SMO-OR** on the benchmark datasets. This shows that the **CBOR-Iter** algorithm achieves similar generalization as **SMO-OR**, but requires very less training time.

In order to show that the **CBOR-Iter** algorithm can scale up to very large datasets containing millions of data points, we present scaling results on a large synthetic OR dataset in 2 dimensions having 5 classes. The data points of each class were generated

Table 5.1: Comparison of training times (in sec) with **CBOR-Iter**, **SMO-OR** and **CBOR-SeDuMi** on benchmark datasets. The test set error rate is given in brackets. (CH-California Housing, CS-Census datasets).

	S-Size	**CBOR-Iter**	**SMO-OR**	**CBOR-SeDuMi**
		sec (err)	sec (err)	sec
CH	10,320	.5 (.623)	551.9 (.619)	112
	13,762	1.5 (.634)	1033.2 (.616)	768.8
	15,482	8.4 (.618)	1142 (.617)	×
	17,202	14.3 (.621)	1410 (.617)	×
	20,230	10.4 (.62)	1838.5 (.62)	×
CS	5,690	.3 (.109)	893 (.128)	20.4
	11,393	.7 (.112)	5281.6 (.107)	108.8
	15,191	1 (.108)	9997.5 (.107)	271.1
	22,331	1.5 (.119)	×	435.7

Table 5.2: Comparison of training times in sec with **CBOR-Iter** and **SMO-OR** on synthetic dataset.

S-Rate	**S-Size**	**CBOR-Iter**	**SMO-OR**
0.002	10,000	1	182
0.0025	12,500	1	260
0.003	15,000	1	340
0.3	1,500,000	9	×
1	5,000,000	36	×

using a GMM with 5 components. Thus the size of problem for **CBOR-Iter** is 25, whereas that for **SMO-OR** it is the size of the whole training set. Table 5.2 presents results of the scaling experiment. As shown in the table, **CBOR-Iter** scales well even for datasets containing millions of data points. Figure 5.1 summarizes experiments comparing the scalability of **CBOR-Iter** and **SeDuMi**. The experiments were done on a synthetic ordinal regression dataset with 5 class, assuming each data point is a cluster. As the figure shows, **CBOR-Iter** algorithm solves the SOCP with a run time under 1 minute even with few thousands of clusters; whereas **SeDuMi** fails if the number of clusters are more than around 2000.

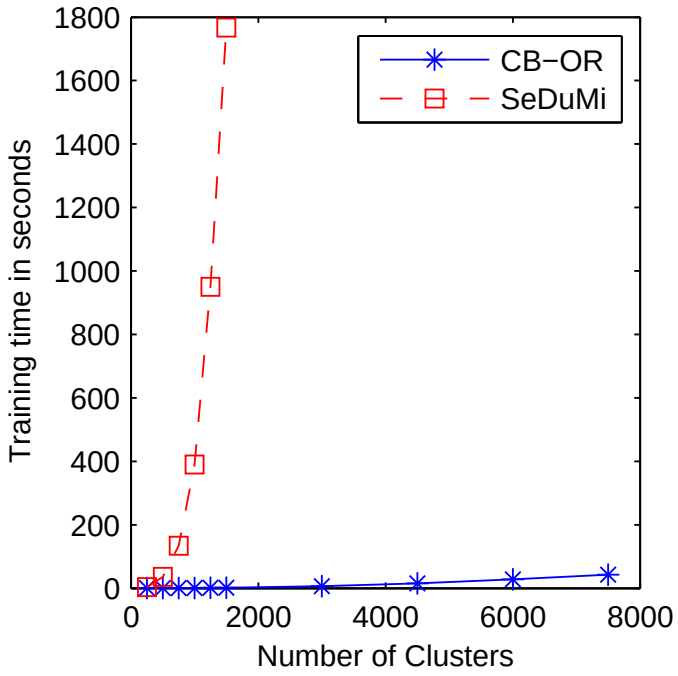

Figure 5.1: Dashed line represents training time with **CBOR-SeDuMi** and continuous line that with **CBOR-Iter** on a synthetic dataset.

5.3 Summary

This chapter presented a fast, easy to implement, iterative solver for the chance-constraint based, large-scale OR formulation. The iterative solver scales very well when compared to generic SOCP solvers like `SeDuMi` and further improves scalability of the large-scale OR scheme. The solver can also be developed for the scalable classification formulation, as both formulations can be posed as SOCPs having similar structure.

Chapter 6

Robustness to Moment Estimation Errors

Abstract

This chapter deals with the important issue of handling moment estimation errors[1]*. As shown in the chapter, validity of the constraints/formulations derived in previous chapters critically depend on how close the estimated moments are to the unknown, true moments. Using the scalable classification formulation as an example, generic procedure of making the formulations robust to moment estimation errors is presented. The main contribution is to show that robust variants of the formulation, built using two novel confidence sets, are also SOCPs and hence are tractable.*

The discussion in chapters 3 and 4 showed how summarizing the data using clusters and then employing chance-constraints for clusters can lead to very scalable classification and ordinal regression formulations. In both cases, online clustering algorithms like BIRCH was used to estimate moments of clusters. However, the moments estimated may not be exact, and as a result the maximum misclassification probability incurred by the separating hyperplane may well be less than the required probability, $1 - \eta$. To protect against this dangerous consequence of inexact moments estimation we employ here the robust optimization methodology of [4, 5] and references therein. In this methodology,

[1] *This work was submitted to the Journal of Machine Learning Research.*

the uncertain (inexact) parameters of an optimization problem are assumed to lie in a bounded convex uncertainty set and the constraints are required to hold for all possible realizations of the parameters in the uncertainty set. Two such novel uncertainty sets (confidence sets) are presented in this chapter for the case of multivariate normal distribution for the components: 1) confidence set for mean and variance is derived as the Cartesian product of individual confidence sets for the moments, 2) an asymptotic approximate joint confidence set for both mean and variance together. A new constraint is derived which when satisfied implies that the original cone constraint is satisfied for all values of moments in the confidence set. We illustrate the methodology using the cone constraints in (3.7) and the corresponding SOCP formulation (denoted by CBC-SOCP). Similar methodologies can be developed for various cone constraints presented in this thesis.

One of the key results presented in this chapter is to show that, when either confidence sets are employed, the robust variant of the original cone constraint is also a cone constraint (see theorems 6.1, 6.2). Using these robust cone constraints, variants of the CBC-SOCP formulation are then proposed (RCBC1-SOCP and RCBC2-SOCP). Experimental results show that in most cases, test set error exceeds $1-\eta$ when the original cone constraints are employed, whereas it does not when the robust variants are employed. The experiments also show that the cone constraint derived using separate confidence set is more pessimistic (conservative) than the one derived using joint confidence set.

The organization of this chapter is as follows: the robust variants of CBC-SOCP are presented in section 6.1. Section 6.2 presents experiments comparing the robust, non-robust cone constraints and formulations. The chapter concludes with a brief summary in section 6.3.

6.1 Robust Classifiers for Large Datasets

As discussed earlier, the moments in the CBC-SOCP formulation are estimated using fast clustering algorithms like BIRCH. Hence the moment estimates are prone to be

erroneous. Assuming that the estimated moments of the clusters are $(\hat{\mu}, \hat{\sigma}^2\mathbf{I})$, the CBC-SOCP formulation (3.7) can be written as follows:

$$\begin{aligned} \min_{\mathbf{w},b,\xi_j,t} \quad & \tfrac{1}{2}t^2 + C\textstyle\sum_{j=1}^{k}\xi_j \\ \text{s.t.} \quad & y_j(\mathbf{w}^\top\hat{\mu}_j - b) \geq 1 - \xi_j + \kappa\ \hat{\sigma}_j t,\ \xi_j \geq 0,\ j = 1,\ldots,k, \\ & \|\mathbf{w}\|_2 \leq t \end{aligned} \tag{6.1}$$

As stated previously, if $X \sim (\mu, \sigma^2\mathbf{I})$ is the random vector generating a cluster, the cone constraint $y(\mathbf{w}^\top\mu - b) \geq 1 - \xi + \kappa\ \sigma\|\mathbf{w}\|_2$ implies $P(y(\mathbf{w}^\top X - b) \geq 1 - \xi) \geq \eta$ where $\kappa = \Phi^{-1}(\eta)$ if components are Gaussian and $\kappa = \sqrt{\frac{\eta}{1-\eta}}$ otherwise. However the CBC-SOCP formulation (6.1) uses estimates of moments $(\hat{\mu}, \hat{\sigma}^2\mathbf{I})$, in place of true moments $(\mu, \sigma^2\mathbf{I})$. Hence validity of the cone constraints will be in question if estimated moments are not accurate.

Suppose the actual moments, (μ_j, σ_j^2), with confidence $c = 1-\delta$, lie in a set $R((\hat{\mu}_j, \hat{\sigma}_j^2), \delta)$. Then the robust CBC-SOCP formulation can be written as follows:

$$\begin{aligned} \min_{\mathbf{w},b,\xi_j,t} \quad & \tfrac{1}{2}t^2 + C\textstyle\sum_{j=1}^{k}\xi_j \\ \text{s.t.} \quad & y_j(\mathbf{w}^\top\mu_j - b) \geq 1 - \xi_j + \kappa\ \sigma_j t,\ j = 1,\ldots,k,\ \forall\ (\mu_j, \sigma_j^2) \in R((\hat{\mu}_j, \hat{\sigma}_j^2), \delta) \\ & \|\mathbf{w}\|_2 \leq t,\ \xi_j \geq 0,\ j = 1,\ldots,k, \end{aligned} \tag{6.2}$$

In the following text, for the special case of multivariate spherical normal distribution for the clusters, expressions for $R((\hat{\mu_j}, \hat{\sigma_j}^2), \delta)$ are derived and later it is shown that the robust variants of the CBC-SOCP formulation also turn out to be SOCPs.

6.1.1 Separate Confidence Sets for Moments

In this section, the robust cone constraint developed using separate confidence sets for the moments is presented. Suppose the unbiased estimates of $(\mu_j, \sigma_j^2\mathbf{I})$ are $(\hat{\mu}_j, \hat{\sigma}_j^2\mathbf{I})$ and that the clusters are normally distributed. Let m_j, n denote the number of training data points belonging to j^{th} cluster and the dimension of training data respectively. Hotelling's

T^2 statistic is the standard tool for inference about the mean of a multivariate normal distribution (refer [27], page 227). According to this statistic, with confidence $c = 1 - \delta$,

$$\|\mu_j - \hat{\mu}_j\|_2^2 \leq p_j^2(c),\ p_j^2(c) = \frac{(m_j - 1)n}{(m_j - n)m_j}\hat{\sigma}_j^2 F_{n,m_j-n}(c) \tag{6.3}$$

where $F_{n,m_j-n}(c)$ is the value at $c = 1 - \delta$ of the inverse cumulative distribution function of the standard F distribution with $n, m_j - n$ degrees of freedom for the χ^2 distributions in the numerator and denominator of the F-distribution respectively. The confidence set for variance can be obtained using Cochran's theorem (refer [42], page 419), according to which $(m_j - 1)\frac{\hat{\sigma}_j^2}{\sigma_j^2} \sim \chi^2_{m_j-1}$. Hence the confidence interval for variance turns out to be

$$q_j^2(c) \leq \sigma_j^2 \leq r_j^2(c)),\ q_j^2(c) = \frac{m_j - 1}{\chi^2_{m_j-1}(1 - \frac{1-c}{2})}\hat{\sigma}_j^2,\ r_j^2(c) = \frac{m_j - 1}{\chi^2_{m_j-1}(\frac{1-c}{2})}\hat{\sigma}_j^2 \tag{6.4}$$

From these individual confidence sets the following confidence set for both the moments can be derived:

$$R_1\left((\mu_j, \sigma_j^2), \delta\right) = \left\{(\mu, \sigma^2) \mid \|\mu - \hat{\mu}_j\|_2^2 \leq p_j^2(\sqrt{c}),\ \sigma^2 \in (q_j^2(\sqrt{c}), r_j^2(\sqrt{c}))\right\} \tag{6.5}$$

The original cone constraints (3.4) can be made robust to moment estimation errors if $y_j(\mathbf{w}^\top \mu_j - b) \geq 1 - \xi_j + \kappa\sigma_j\|\mathbf{w}\|_2$ for all (μ_j, σ_j^2) lying in the confidence set described by (6.5). To this end consider the following theorem:

THEOREM 6.1. *The constraints (6.6) and (6.7) are equivalent to each other:*

$$y(\mathbf{w}^\top \mu - b) \geq 1 - \xi + \kappa\sigma\|\mathbf{w}\|_2,\ (\mu, \sigma^2) \in R_1((\hat{\mu}, \hat{\sigma}^2), \delta) \tag{6.6}$$

$$y(\mathbf{w}^\top \hat{\mu} - b) \geq 1 - \xi + \left(p(\sqrt{c}) + \kappa r(\sqrt{c})\right)\|\mathbf{w}\|_2 \tag{6.7}$$

where the values of p, q, r are given by (6.3) and (6.4).

Proof. It is easy to see that (6.6) is satisfied for all $(\mu, \sigma^2) \in R_1((\hat{\mu}, \hat{\sigma}^2), \delta)$ iff:

$$\begin{aligned} & \left(\min_{\|\mu-\hat{\mu}\|_2^2 \leq p^2(\sqrt{c}),\ \sigma^2 \in (q^2(\sqrt{c}), r^2(\sqrt{c}))} y\mathbf{w}^\top \mu - \kappa\sigma\|\mathbf{w}\|_2\right) \geq yb + 1 - \xi \\ \Leftrightarrow \quad & \left(\min_{\|\mu-\hat{\mu}\|_2^2 \leq p^2(\sqrt{c})} y\mathbf{w}^\top(\mu - \hat{\mu})\right) + y\mathbf{w}^\top\hat{\mu} - \kappa r(\sqrt{c})\|\mathbf{w}\|_2 \geq yb + 1 - \xi \\ \Leftrightarrow \quad & -p(\sqrt{c})\|\mathbf{w}\|_2 + y\mathbf{w}^\top\hat{\mu} - \kappa r(\sqrt{c})\|\mathbf{w}\|_2 \geq yb + 1 - \xi \end{aligned}$$

which is same as (6.7). This completes the proof. □

Interestingly (6.7) is also a cone constraint. Using theorem 6.1, a robust scalable SOCP formulation (RCBC1-SOCP), similar in spirit to CBC-SOCP (3.7), can be derived:

$$\begin{aligned} \min_{\mathbf{w},b,\xi_j,t} \quad & \tfrac{1}{2}t^2 + C\textstyle\sum_{j=1}^k \xi_j \\ \text{s.t.} \quad & y_j(\mathbf{w}^\top\hat{\mu}_j - b) \geq 1 - \xi_j + (p_j(\sqrt{c}) + \kappa r_j(\sqrt{c}))\, t,\ j = 1, \ldots, k \\ & \|\mathbf{w}\|_2 \leq t,\ \xi_j \geq 0,\ j = 1, \ldots, k \end{aligned} \tag{6.8}$$

6.1.2 Joint Confidence Sets for Moments

In this section, the robust cone constraint developed using joint confidence sets for the moments is presented. It is well known (see, for example, [38], page 211) that the multivariate Maximum Likelihood Estimate (MLE), $\hat{\theta}_{(m)}$, for θ, a $(n \times 1)$ vector of parameters, is asymptotically normal in the sense that

$$\hat{\theta}_{(m)} \approx N^{(n)}\left(\theta, \frac{1}{m}\Sigma(\theta)\right) \tag{6.9}$$

where m is number of samples used for estimation, $\Sigma^{-1}(\theta)$ is the matrix containing the entries $\sigma_{kl}(\theta) \equiv -E\left[\frac{\partial^2}{\partial\theta_k \partial\theta_l}\left(\log f_{\mathbf{X}}(\mathbf{x};\theta)\right)\right]$. Here $f_{\mathbf{X}}(\mathbf{x};\theta)$ denotes the pdf of the data distribution with the actual parameters θ.

For the case of spherical normal distribution for clusters, the vector of parameters of the j^{th} cluster, θ_j, is $\begin{bmatrix} \mu_j \\ {\sigma_j}^2 \end{bmatrix}$ and the pdf, $f_{\mathbf{X}}(\mathbf{x};\theta_j)$, is $N^n(\mu_j, \sigma_j^2\mathbf{I})$. Now the entries

$\sigma_{kl}(\theta_j)$ can be written as follows:

$$\sigma_{kl}(\theta_j) = \begin{cases} \frac{1}{\sigma_j^2} & \text{if } k = l,\ l = 1, \ldots, n \\ 0 & \text{if } k \neq l,\ l = 1, \ldots, n \\ 0 & \text{if } k \neq l,\ k = n+1 \text{ or } l = n+1 \\ \frac{n}{2\sigma_j^4} & \text{if } k = l = n+1 \end{cases} \tag{6.10}$$

Recall that m_j, n denote the number of training data points belonging to j^{th} cluster and the dimension of training data respectively. With a slight abuse of notation let $(\hat{\mu}_j, \hat{\sigma}_j^2\mathbf{I})$ denote the MLE moment estimates of the j^{th} cluster. Using (6.10), for the case of spherical normal distribution for clusters, (6.9) can be re-written as follows:

$$\begin{bmatrix} \hat{\mu_j} \\ \hat{\sigma_j}^2 \end{bmatrix} \approx N^{n+1}\left(\begin{bmatrix} \mu_j \\ \sigma_j^2 \end{bmatrix}, \frac{1}{m_j} \begin{bmatrix} \text{diag}_n(\sigma_j^2) & 0_{n\times 1} \\ 0_{1\times n} & \frac{2\sigma_j^4}{n} \end{bmatrix} \right) \tag{6.11}$$

Now since $(\hat{\mu_j}, \hat{\sigma_j}^2)$ approximately follows Normal distribution,

$$U_j = \frac{m_j}{\sigma_j^2}\|\mu_j - \hat{\mu_j}\|_2^2 + \frac{nm_j}{2\sigma_j^4}(\sigma_j^2 - \hat{\sigma_j}^2)^2$$

will approximately follow chi-square distribution. Thus, for large m_j, the set $R((\hat{\mu}_j, \hat{\sigma}_j^2), \delta) = \{(\mu_j, \sigma_j^2) : U_j < \chi_{n+1}^2(1-\delta)\}$ is an approximate $1-\delta$ confidence set for (μ_j, σ_j^2). Here $\chi_{n+1}^2(1-\delta)$ denotes the inverse of the chi-square cumulative distribution function with $n+1$ degrees of freedom and at the value $1-\delta$.

As discussed in [2], for large enough m_j,

$$V_j = \frac{m_j}{\hat{\sigma_j}^2}\|\mu_j - \hat{\mu_j}\|_2^2 + \frac{nm_j}{2\hat{\sigma_j}^4}(\sigma_j^2 - \hat{\sigma_j}^2)^2 \approx \chi_{n+1}^2 \tag{6.12}$$

Thus for large m_j, the set $R((\hat{\mu}_j, \hat{\sigma}_j^2), \delta) = \{(\mu_j, \sigma_j^2) : V_j < \chi_{n+1}^2(1-\delta)\}$, which is an ellipse in $\|\mu_j - \hat{\mu_j}\|_2$ and $(\sigma_j^2 - \hat{\sigma_j}^2)$, is also an approximate $1-\delta$ confidence set for (μ_j, σ_j^2). In fact, the authors in their experiments show that this approximate confidence set performs well even in cases where the normal distribution assumption fails.

To summarize, the discussion shows that the actual moments of j^{th} cluster, with confidence $1-\delta$, lie in an ellipse described by $V_j \leq \chi^2_{n+1}(1-\delta)(= f, \text{ say})$. Hence the joint confidence interval can be written as:

$$R_2\left((\hat{\mu}_j, \hat{\sigma}_j^2), \delta\right) = \left\{(\mu_j, \sigma_j^2) : \frac{m_j}{\hat{\sigma_j}^2}\|\mu_j - \hat{\mu_j}\|_2^2 + \frac{nm_j}{2\hat{\sigma_j}^4}(\sigma_j^2 - \hat{\sigma_j}^2)^2 \leq f\right\} \tag{6.13}$$

Now coming back to the classification problem, one needs to ensure that cone constraints of the form $y(\mathbf{w}^\top\mu - b) \geq 1 - \xi + \kappa\sigma\|\mathbf{w}\|_2$ are satisfied for all values of (μ, σ^2) lying in the ellipse described by $V \leq f$. To this end, consider the following theorem:

THEOREM 6.2. *The constraints (6.14) and (6.15) are equivalent to each other:*

$$y(\mathbf{w}^\top\mu - b) \geq 1 - \xi + \kappa\sigma\|\mathbf{w}\|_2, \ (\mu, \sigma^2) \ \in \ R_2((\hat{\mu}, \hat{\sigma}^2), \delta) \tag{6.14}$$

$$y(\mathbf{w}^\top\hat{\mu} - b) \geq 1 - \xi + (\kappa\sigma^* + g^*\hat{\sigma})\|\mathbf{w}\|_2 \tag{6.15}$$

where, $g^* = \sqrt{\frac{2nf\sigma^{*2}}{m(2n\sigma^{*2} + \kappa^2\hat{\sigma}^2)}}$ *and* σ^{*2} *is a particular root of the cubic (6.20)*

Proof. In order to satisfy the cone constraint $y(\mathbf{w}^\top\mu - b) \geq 1 - \xi + \kappa\sigma\|\mathbf{w}\|_2$ where (μ, σ^2) lies in the ellipse described by $V \leq f$, it is enough to constrain that $y(\mathbf{w}^\top\mu^* - b) \geq 1 - \xi + \kappa\sigma^*\|\mathbf{w}\|_2$, where (μ^*, σ^{*2}) is the solution of the following minimization problem:

$$\begin{aligned} \min_{(\mu,\sigma^2)} \quad & y\mathbf{w}^\top\mu - \kappa\sigma\|\mathbf{w}\|_2 \\ \text{s.t.} \quad & \tfrac{m}{\hat{\sigma}^2}\|\mu - \hat{\mu}\|_2^2 + \tfrac{nm}{2\hat{\sigma}^4}(\sigma^2 - \hat{\sigma}^2)^2 \leq f \end{aligned} \tag{6.16}$$

The Lagrangian of (6.16) turns out to be $\mathcal{L} = y\mathbf{w}^\top\mu - \kappa\sigma\|\mathbf{w}\|_2 + \lambda(V - f), \lambda \geq 0$ where λ is the Lagrange multiplier. KKT conditions are as follows:

$$\nabla_\mu\mathcal{L} = 0 \ \Rightarrow \ y\mathbf{w} + \frac{2\lambda m}{\hat{\sigma}^2}(\mu - \hat{\mu}) = 0 \Rightarrow \lambda \neq 0 \tag{6.17}$$

$$\frac{\partial\mathcal{L}}{\partial\sigma^2} = 0 \ \Rightarrow \ \frac{-\kappa\|\mathbf{w}\|_2}{2\sigma} + \frac{\lambda mn}{\hat{\sigma}^4}(\sigma^2 - \hat{\sigma}^2) = 0 \Rightarrow \sigma^2 \geq \hat{\sigma}^2 \tag{6.18}$$

$$\lambda(V - f) = 0, \lambda \neq 0 \ \Rightarrow \ \frac{m}{\hat{\sigma}^2}\|\mu - \hat{\mu}\|_2^2 + \frac{nm}{2\hat{\sigma}^4}(\sigma^2 - \hat{\sigma}^2)^2 - f = 0 \tag{6.19}$$

Substituting the values of $(\mu - \hat{\mu})$ from (6.17) and $(\sigma^2 - \hat{\sigma}^2)$ from 6.18 in (6.19), we

get

$$\lambda = \frac{\hat{\sigma}\|\mathbf{w}\|_2}{2\sqrt{fm}}\sqrt{\left(1+\frac{\kappa^2\hat{\sigma}^2}{2n\sigma^2}\right)}$$

Substituting this λ expression in (6.18), we get the following cubic in $z = \sigma^2$:

$$(2mn^2)z^3 + mn\hat{\sigma}^2(\kappa^2 - 4n)z^2 + 2mn\hat{\sigma}^4(n-\kappa^2)z + \kappa^2\hat{\sigma}^6(nm-2f) = 0 \tag{6.20}$$

CLAIM 1. *Given $\kappa, f > 0$, $\exists$ a unique root z^* of cubic (6.20) in the interval $(\hat{\sigma}^2, \infty)$.*

Proof. Let the cubic in (6.20) be represented by $p(z)$. The following observations are true:

- $p(\hat{\sigma}^2) = -2\kappa^2 f < 0$.
- $p^{'}(\hat{\sigma}^2) = 0$.
- $p^{'}(z) = mn^2\left[6(z-\hat{\sigma}^2)^2 + 4\hat{\sigma}^2(z-\hat{\sigma}^2)\right] + 2mn\kappa^2\hat{\sigma}^2(z-\hat{\sigma}^2)$. Implying $p^{'}(z) > 0,\ \forall\ z > \hat{\sigma}^2$. So $p(z)$ is strictly increasing in the interval $[\hat{\sigma}^2, \infty)$.

The above three properties prove the claim. □

Hence the cubic (6.20) can be solved for the particular root, $z^* = \sigma^{*2}$, which is $\geq \hat{\sigma}^2$. Now

$$y\mathbf{w}^\top\mu^* = y\mathbf{w}^\top\hat{\mu} - g^*\hat{\sigma}\|\mathbf{w}\|_2$$

where $g^* = \sqrt{\frac{2nf\sigma^{*2}}{m(2n\sigma^{*2}+\kappa^2\hat{\sigma}^2)}}$. Thus the final constraint turns out to be (6.15). This completes the proof. □

Using theorem 6.2, a robust scalable SOCP formulation (RCBC2-SOCP), similar in spirit to CBC-SOCP (3.7), can be derived:

$$\begin{aligned}
\min_{\mathbf{w},b,\xi_j,t} \quad & \tfrac{1}{2}t^2 + C\textstyle\sum_{j=1}^{k}\xi_j \\
\text{s.t.} \quad & y_j(\mathbf{w}^\top\hat{\mu}_j - b) \geq 1 - \xi_j + (\kappa\sigma_j^* + g_j^*\hat{\sigma}_j)t,\ j = 1,\ldots,k \\
& \|\mathbf{w}\|_2 \leq t,\ \xi_j \geq 0,\ j = 1,\ldots,k
\end{aligned} \tag{6.21}$$

Note that RCBC2-SOCP is indeed an SOCP formulation, with the form exactly same as its non-robust counterpart, CBC-SOCP. Geometrically, the only difference in the CBC-SOCP and RCBC2-SOCP formulations is the size of spheres (refer section 3.3). For a fixed η, in case of CBC-SOCP, the sphere sizes are proportional to $\hat{\sigma}_j$, whereas, in case of RCBC2-SOCP, the sizes are proportional to $\kappa\sigma_j^* + g_j^*\hat{\sigma}_j$. For RCBC1-SOCP, the sphere sizes are proportional to $p_j(\sqrt{c}) + \kappa r_j(\sqrt{c})$. Experiments show that the cone constraints derived using separate confidence sets are more conservative than the ones derived using joint confidence sets, which is as expected.

6.2 Numerical Experiments

This section presents experiments that compare performance of the cone constraints derived in sections 3.2, 6.1. Comparison is done for the following four cone constraints, all of which imply $P(y(\mathbf{w}^\top X - b) \geq 0) \geq \eta$ where $X \sim (\mu, \sigma^2\mathbf{I})$ and y is its label (see theorem 2.1):

$$y(\mathbf{w}^\top \mu - b) \geq \kappa\sigma\|\mathbf{w}\|_2 \tag{6.22}$$

$$y(\mathbf{w}^\top \hat{\mu} - b) \geq \kappa\hat{\sigma}\|\mathbf{w}\|_2 \tag{6.23}$$

$$y(\mathbf{w}^\top \hat{\mu} - b) \geq \left(p(\sqrt{c}) + \kappa r(\sqrt{c})\right)\|\mathbf{w}\|_2 \tag{6.24}$$

$$y(\mathbf{w}^\top \hat{\mu} - b) \geq (\kappa\sigma^* + g^*\hat{\sigma})\|\mathbf{w}\|_2 \tag{6.25}$$

(6.22), (6.23) were presented in section 3.2 and represent the standard cone constraint with true and estimated moments respectively. Let these cone constraints be denoted by **T-SOC** and **E-SOC** respectively. (6.24), (6.25) were derived in section 6.1 and represent the robust cone constraints derived using separate confidence intervals and joint confidence intervals for moments respectively. Let these cone constraints be denoted by **R1-SOC** and **R2-SOC** respectively. The experiments in section 6.2.2 compare the objective value and averaged test set accuracy for the CBC-SOCP and RCBC2-SOCP formulations.

6.2.1 Comparison of the Cone Constraints

The discussion in the previous sections showed that if (6.22) is satisfied, then with probability η, the data generated by the moments $(\mu, \sigma^2\mathbf{I})$ will lie on the correct side of the separating hyperplane, $\mathbf{w}^\top\mathbf{x} - b = 0$. If there are no or very less estimation errors, then (6.23) also implies the same as (6.22). However if the estimation errors are not negligible, then the cone constraints need to be robust. (6.25) and (6.24) are two such robust variants which imply that the data generated by the moments $(\mu, \sigma^2\mathbf{I})$ will lie on the correct side of the separating hyperplane, $\mathbf{w}^\top\mathbf{x} - b = 0$, even if the estimation errors are not negligible. This section presents results that compare the four cone constraints: **T-SOC, E-SOC, R1-SOC, R2-SOC**.

In order to illustrate the effect of error in moment estimation on the correctness of the cone constraints, the following experiment was done: in each run of the experiment, 100 training examples and 10000 test examples were generated from a normal distribution with fixed moments $\left(\begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right)$. $\mathbf{w}$ was chosen to be $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$. As per (6.23), if b is chosen as $\mathbf{w}^\top\hat{\mu} - \kappa\hat{\sigma}\|\mathbf{w}\|_2$, where the moments were estimated using the 100 training examples, then on any testset not more than $1-\eta$ fraction of the examples will lie in the negative half space of $\mathbf{w}^\top\mathbf{x} - b = 0$. Similarly, as per (6.25), b needs to be chosen as $\mathbf{w}^\top\hat{\mu} - (\kappa\sigma^* + g^*\hat{\sigma})\|\mathbf{w}\|_2$ and for (6.24), $b = \mathbf{w}^\top\hat{\mu} - (p(\sqrt{c}) + \kappa r(\sqrt{c}))\,\|\mathbf{w}\|_2$. For each run of the experiment, the same training and test set was used to compare the four cone constraints. Of the 20000 runs of the experiment, the % of runs in which the testset error was greater than $1-\eta$ (denoted by p_η), for various values of η, is shown in table 6.1. The values e_η, m_η represent the average and maximum test set error in the 20000 runs. Ideally, p_η must be 0 and e_η, m_η must be equal to $1-\eta$. Also, lesser their value, more is the robustness towards moment estimation errors. In order to have a baseline for comparison, the case $b = \mathbf{w}^\top\mu - \kappa\sigma\|\mathbf{w}\|_2$, where (μ, σ^2) are the true moments, is also reported in the table under the column **T-SOC**.

Firstly, $p_\eta \gg 0$ and $m_\eta \gg 1-\eta$ for **E-SOC** showing that validity of the original cone constraint (3.4) is indeed in question when moment estimation errors are present.

Secondly, the values of p_η, e_η, m_η are less for **R1-SOC** and **R2-SOC**, showing that the corresponding cone constraints are robust to moment estimation errors. The fact that values of m_η for **R1-SOC** are less than $1-\eta$ and less than those for **R2-SOC** show that (6.24) unnecessarily ensures a tighter constraint than required in order to make the constraint robust. Thus **R2-SOC** is robust and also not very pessimistic, and hence has more practical utility. Figure 6.1 is a histogram for $\eta = 0.9$ and $\delta = 0.9$. A group of 3 bars at x represents the number of experiment runs where testset error was between $(x-0.5)\%$ and $(x+0.5)\%$ with **E-SOC, T-SOC, R1-SOC, R2-SOC**. Ideally, the histogram must be a single peak at $x = 10\%$ and zero elsewhere. **T-SOC** behaves nearest to the ideal since true moments are used in the cone constraint. Also **R1-SOC, R2-SOC** have least number of experiments with testset error greater than $1-\eta$ ($= 10\%$ here) showing that they are robust towards moment estimation errors. The figure again confirms that **R1-SOC** is more pessimistic than **R2-SOC**. The histogram also shows that in almost half the number of experiment runs **E-SOC** violated the misclassification error bound of $1-\eta$.

Table 6.2 compares the values of p_η, e_η, m_η for **E-SOC** and **R2-SOC** with $\eta = 0.9, \delta = 0.9$ for synthetic datasets generated from various distributions other than normal. **U, B** represent datasets generated from distributions where each dimension is generated from independent uniform, beta (parameters $\alpha, \beta = 2$)[2] random variables respectively and shifted appropriately to have $\mu = [1\,0]^\top$. **E, G** represent datasets with double exponential distribution ($\lambda = 1, \mu = [1\,0]^\top$)[3] and double gamma distribution ($k = 2, \theta = 2, \mu = [1\,0]^\top$)[4] respectively for each dimension. The means are shifted appropriately. The table clearly shows that **R2-SOC** performs better than **E-SOC** even when the distribution assumption of normal is not valid.

[2]Beta: $f(x) = \frac{1}{B(\alpha,\beta)} x^{\alpha-1}(1-x)^{\beta-1},\ 0 \le x \le 1$

[3]Double Exponential: $f(x) = \frac{\lambda}{2}\exp\left(-\lambda|x-\mu|\right)$

[4]Double Gamma: $f(x) = \frac{|x-\mu|^{k-1}}{2} \frac{\exp -\left(\frac{|x-\mu|}{\theta}\right)}{\theta^k \Gamma(k)}$

Table 6.1: Results on synthetic data, comparing **E-SOC, T-SOC, R1-SOC, R2-SOC**.

η	e_η			
	E-SOC	**T-SOC**	**R1-SOC**	**R2-SOC**
0.9	0.10302	0.09999	0.04257	0.06020
0.8	0.20267	0.20001	0.11148	0.13601
0.7	0.30164	0.29994	0.19662	0.21927
0.6	0.40108	0.39999	0.29442	0.30867
η	m_η			
	E-SOC	**T-SOC**	**R1-SOC**	**R2-SOC**
0.9	0.21220	0.11140	0.08520	0.14270
0.8	0.35730	0.21410	0.16220	0.27740
0.7	0.45240	0.31800	0.26080	0.35920
0.6	0.55320	0.42160	0.33790	0.46600
η	$p_\eta\%$			
	E-SOC	**T-SOC**	**R1-SOC**	**R2-SOC**
0.9	52.93	50.41	0.00	1.25
0.8	51.78	50.40	0.00	0.99
0.7	50.33	49.28	0.00	0.96
0.6	50.74	49.46	0.00	0.83

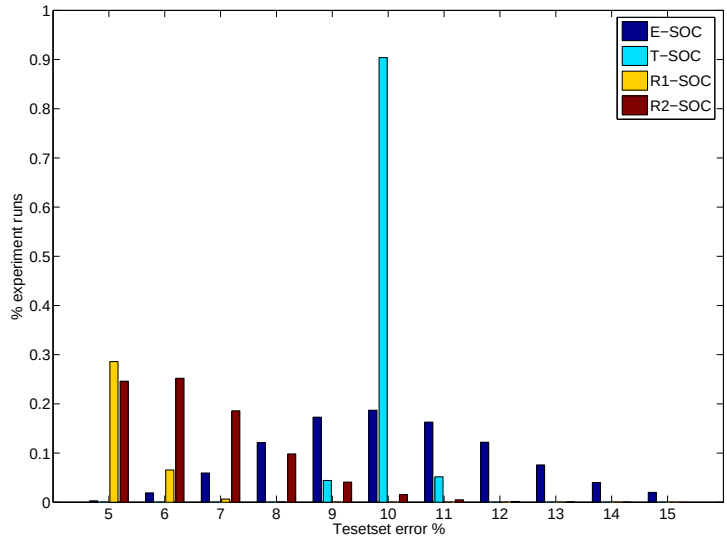

Figure 6.1: Histogram of % experiment runs vs. testset error % at $\eta = 0.9$ for **E-SOC, T-SOC, R1-SOC, R2-SOC**.

Table 6.2: Results on synthetic data generated from **U, B, E, G** distributions comparing **E-SOC, T-SOC, R2-SOC** at $\eta = 0.9$.

	e_η		m_η		$p_\eta\%$	
	E-SOC	**R2-SOC**	**E-SOC**	**R2-SOC**	**E-SOC**	**R2-SOC**
U	0.13215	0.04855	0.26090	0.18150	84.64	5.48
B	0.11998	0.06193	0.26080	0.18910	77.77	4.24
E	0.08441	0.05615	0.16890	0.12300	17.36	0.17
G	0.09108	0.05766	0.17560	0.12270	29.93	0.44

6.2.2 Comparison of CBC-SOCP and RCBC2-SOCP

This section presents results comparing the CBC-SOCP formulation (3.7) and its robust counterpart, RCBC2-SOCP (6.21). Two synthetic dataset templates were constructed with fixed number of clusters, means and variances. Training and test samples were generated from these templates 1000 times and in each case, the moments were estimated using the training data. Since the true moments are known, the 1000 experiments can be arranged according to the value of V in (6.12) i.e. according to how much the true and estimated moments differed. The top few experiments where the true moments and estimated moments differed the maximum are considered and summarized in table 6.3 below. This was done to show that in such cases, where really the moments are erroneously estimated, the robust counterpart performs better though its objective value is higher. The objective value of RCBC2-SOCP is higher because the constraints are more tighter. One can also observe from the table that as $1 - \delta$ value decreases, the objective values of RCBC2-SOCP and CBC-SOCP become closer, which is expected.

6.3 Summary

Learning algorithms which use the estimated means and variances are prone to infeasibility i.e., the portion of misclassified data points may be larger than the required one, and the chance of this happening can be as high as 50%. The robust classification scheme, which uses ellipsoidal confidence sets centered around the estimated mean and variance, indeed succeed to achieve feasibility with high fidelity. The joint confidence set performs

Table 6.3: Results on synthetic datasets, comparing the performance of CBC-SOCP and RCBC2-SOCP.

	η	$1-\delta$	**CBC-SOCP**		**RCBC2-SOCP**	
			Acc.(%)	Obj.	Acc.(%)	Obj.
DS-1	0.9	0.9	98.39	0.918	98.39	2.733
	0.9	0.5	98.15	0.938	98.21	1.652
	0.9	0.1	98.26	0.977	98.28	1.274
	0.7	0.9	97.99	0.298	98.21	0.438
	0.7	0.5	97.97	0.309	98.14	0.382
	0.7	0.1	98.18	0.279	98.27	0.307
	0.52	0.9	98.00	0.180	98.16	0.231
	0.52	0.5	98.03	0.191	98.12	0.223
	0.52	0.1	97.90	0.189	97.92	0.204
DS-2	0.9	0.9	98.47	0.534	98.81	0.850
	0.9	0.5	98.65	0.646	98.85	0.888
	0.9	0.1	98.67	0.646	98.77	0.752
	0.7	0.9	98.28	0.263	98.67	0.350
	0.7	0.5	98.01	0.237	98.29	0.278
	0.7	0.1	98.40	0.268	98.52	0.291
	0.52	0.9	98.06	0.180	98.47	0.227
	0.52	0.5	97.85	0.172	98.17	0.196
	0.52	0.1	98.03	0.177	98.18	0.189

better than the one built using individual confidence sets for mean and variance, and hence expected to give better values of the objective function. We also note that for large enough datasets, good results are obtained by the robust classifiers, even when the underlying distribution is not Normal.

In this thesis, bounds on the misclassification probability for the case of non-normal distributions are based on the Chebyshev's inequality. Better bounds may be used in the future, which are based on partial information of the underlying distribution (see for e.g. [6]). This will result in less conservative classification schemes, but perhaps at the cost of less tractable optimization problems.

Chapter 7

Conclusions

Abstract

In this final chapter we summarize the main contributions and discuss related issues, open problems and possible directions for future work.

This thesis presented ideas to leverage existing learning algorithms using chance-constrained programs. Traditionally, chance-constraint approaches were employed for handling uncertainty in training data. A key idea presented in the thesis was to employ chance-constraints for developing scalable learning formulations. It was shown that CCP based approaches lead to accurate, fast, as well as robust, learning algorithms. The proposed CCP based formulations give insights into important quantities like generalization error and also are tractable. Using second order moment information, the CCPs were posed as SOCPs, which are well studied convex optimization problems. It was shown that the duals turn out to be geometric optimization problems involving ellipsoids and spheres. The thesis also presented simple iterative solvers which further increase scalability of the large-scale classification and OR formulations. The methodology for handling moment estimation errors was also discussed.

The problem of classification with specified error rates was solved by employing chance-constraints for each class which ensure that the actual false-negative and false-positive rates do not exceed the specified limits. Using second order moments of class

conditional densities, the resulting CCP was posed as an SOCP. An efficient algorithm to solve the dual SOCP, which is the problem of minimizing distance between ellipsoids, was also presented. The formulation when extended to feature spaces also yields an SOCP. Important problems like medical diagnosis, fault detection and other classification problems where preferential bias towards a particular class is desired, can be efficiently solved with the novel formulation. The formulation achieves generalization comparable to that with existing biased classification methods and additionally guarantees that the generalization error is less than the specified limit.

Employing chance-constraints for clusters in training data, scalable maximum margin formulations for classification and OR were developed. Using second order moments of clusters, the CCPs were posed as SOCPs involving one cone constraint and one linear constraint per cluster. Since the SOCPs involve substantially smaller number of variables and constraints than the corresponding baseline formulations, the training times are comparable to those for the state-of-the-art solvers, even when generic solvers are employed to solve the SOCPs. The scalability of the proposed training schemes can further be improved by employing novel projected co-ordinate descent based algorithms for solving the SOCPs. The speed-up achieved with such solvers is shown to be as high as 10000 times when compared to the state-of-the-art. Thus the thesis also throws light on the importance of solving some special cases of SOCP, like SOCPs with a single cone constraint, more efficiently. The scalable classification and OR formulations were extended to feature spaces using the kernel trick. It was shown that the training times grow linearly with the training set size even though the formulations work in feature spaces. Large-scale classification problems like intrusion detection, spam filtering, web-page classification and large ranking problems like focused crawling, personalized searches, information retrieval can be efficiently solved with the scalable learning algorithms.

The thesis also throws light on the issue of making the learning formulations robust to estimation errors. Experiments were detailed showing that in as high as 50% cases the constraints can actually be violated if estimate moments are employed instead of

the true moments. Using joint confidence sets for moments a robust, non-conservative and tractable formulation for the large-scale classification problem was derived. It was shown that constraints in the robust formulation imply feasibility even when estimated moments are erroneous.

In this thesis the Chebyshev-Cantelli inequality, which is based on second order moment information, was exploited in order to pose the CCPs as convex optimization problems. However Chebyshev's inequality models the worst-case behaviour — it is valid for all distributions having the specified moments. Therefore alternate concentration inequalities which are less conservative need to be explored. Also in cases where the second order moment information is not available, for e.g., interval data, or in cases where the second order moments are hard to estimate, for e.g., micro-array data where low samples of high dimensional vectors are available, the Chebyshev's inequality cannot be employed. The work by Ben-Tal and Nemirovski [6] is a good manuscript which discusses such situations. However the question whether such inequalities lead to tractable algorithms needs to be answered. Thus an important direction of future research is to explore various ways of modeling the chance-constraints leading to non-conservative and tractable formulations.

The dual of biased classification formulation (2.4) turns out to be the problem of minimizing distance between ellipsoids and that of the scalable classification formulation (3.7) is minimizing distance between convex hulls of spheres. It is easy to see that such geometric problems involving spheres and ellipsoids arise due to the cone-constraints. This throws light on the importance of developing scalable algorithms for solving geometric optimization problems involving spheres and ellipsoids. Efficient algorithms for such problems would further enhance the SOCP-based learning algorithms.

As shown in the experimental results (section 6.2), the formulations need to be robust from moment estimation errors. Two such robust variants were presented in the thesis and it was shown that the variant which employs the joint confidence set is robust and less conservative. Another important direction of future work is to derive confidence sets which are non-conservative and also lead to tractable learning algorithms.

Appendix A

Casting Chance-Constraint as Cone-Constraint

This section presents the proof of theorem 2.1. To this end consider the Chebyshev-Cantelli inequality [36]:

THEOREM A.1. *Let Z be a random vector and (μ, σ^2) its second order moments. Then for any $t > 0$,*

$$Prob(Z - \mu \geq t) \leq \frac{\sigma^2}{\sigma^2 + t^2}$$

Now let X be an n-dimensional random variable with moments (μ, Σ). Applying theorem A.1 to the random variable $-\mathbf{c}^\top X,\ \mathbf{c} \in \mathbb{R}^n$, which has moments $\left(-\mathbf{c}^\top \mu, \mathbf{c}^\top \Sigma \mathbf{c}\right)$ and with $t = \mathbf{c}^\top \mu - d$, we get

$$Prob(-\mathbf{c}^\top X \geq -d) \leq \frac{\mathbf{c}^\top \Sigma \mathbf{c}}{\mathbf{c}^\top \Sigma \mathbf{c} + (\mathbf{c}^\top \mu - d)^2} \tag{A.1}$$

As per theorem 2.1, we need to ensure $Prob(\mathbf{c}^\top X \geq d) \geq e$. In other words, we need to ensure $Prob(-\mathbf{c}^\top X \geq -d) \leq 1 - e$. Using (A.1), it is easy to see that this constraint is ensured if:

$$\frac{\mathbf{c}^\top \Sigma \mathbf{c}}{\mathbf{c}^\top \Sigma \mathbf{c} + (\mathbf{c}^\top \mu - d)^2} \leq 1 - e$$

Re-arranging terms in the above inequality gives (2.2).

Now if X is multivariate normal and Φ is the distribution function of univariate normal with 0 mean and unit variance, then

$$Prob(\mathbf{c}^\top X \geq d) = \Phi(\frac{\mathbf{c}^\top \mu - d}{\sqrt{\mathbf{c}^\top \Sigma \mathbf{c}}}) \geq e$$

leading to the inequality $\mathbf{c}^\top \mu - d \geq \Phi^{-1}(e)\sqrt{\mathbf{c}^\top \Sigma \mathbf{c}}$. This completes the proof.

Appendix B

Fast Solver for Scalable Classification Formulation

This section derives fast algorithm for solving the chance-constraint based scalable classification formulation (3.7). We begin by re-writing the formulation in the following equivalent form [46]:

$$\begin{aligned} \min_{\mathbf{w},b,\xi_j} \quad & \textstyle\sum_{j=1}^{k} \xi_j \\ \text{s.t.} \quad & y_j(\mathbf{w}^\top \mu_j - b) \geq 1 - \xi_j + \kappa\, \sigma_j W,\ j = 1, \ldots, k \\ & W \geq \|\mathbf{w}\|_2,\ \xi_j \geq 0,\ j = 1, \ldots, k \end{aligned} \tag{B.1}$$

The parameters C in (3.7) and W in (B.1) are related (see section 4.1.1 a for discussion). Using the arguments presented in section 4.3, the formulation (B.1) can be extended to feature spaces:

$$\begin{aligned} \min_{\mathbf{w},b,\xi_j} \quad & \textstyle\sum_{j=1}^{k} \xi_j \\ \text{s.t.} \quad & y_j(\mathbf{w}^\top \phi(\mu_j) - b) \geq 1 - \xi_j + r_j W,\ j = 1, \ldots, k \\ & W \geq \|\mathbf{w}\|_2,\ \xi_j \geq 0,\ j = 1, \ldots, k \end{aligned}$$

where $r_j = \sqrt{2\left(1 - \exp\left\{-\zeta\left(\kappa\sigma_j\right)^2\right\}\right)}$ and ζ is the Gaussian kernel parameter. The dual of the above formulation can be written as:

$$\begin{aligned} \min_{\alpha} \quad & W\sqrt{\alpha^\top \mathbf{Q}\alpha} - \mathbf{d}^\top \alpha \\ \text{s.t.} \quad & 0 \leq \alpha \leq 1, \mathbf{y}^\top \alpha = 0 \end{aligned} \tag{B.2}$$

where α is a vector of k Lagrange multipliers (one for each inequality in (B.1)), $\mathbf{Q}$ is the matrix whose $(i,j)^{th}$ element is $y_i y_j K(i,j)$ (K is the kernel function), $\mathbf{d}$ is the vector containing entries $1 + \kappa\sigma_j W$ and $\mathbf{y}$ denotes the vector containing the cluster labels y_j. The decision function $f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} - b$ can be written in the following form:

$$f(\mathbf{x}) = g(\mathbf{x}) - b, \;\; g(\mathbf{x}) = \frac{W}{\sqrt{\alpha^\top \mathbf{Q}\alpha}} \mathbf{Q}_{\mathbf{x}}^\top \alpha \tag{B.3}$$

where $\mathbf{Q}_{\mathbf{x}}$ represents the vector of dot products $\phi(\mathbf{x})^\top \phi(\mu_j)$. Note that both the dual (B.2) and $f(\mathbf{x})$ involve only dot products of the means of clusters. Hence classification can be done in any feature space using the kernel trick. The necessary and sufficient KKT conditions can be summarized as follows:

$$\begin{aligned} \alpha_j = 0, y_j = 1 \quad & g(\mu_j) - 1 - r_j W \geq b \\ \alpha_j = 0, y_j = -1 \quad & g(\mu_j) + 1 + r_j W \leq b \\ 0 < \alpha_j < 1, y_j = 1 \quad & g(\mu_j) - 1 - r_j W = b \\ 0 < \alpha_j < 1, y_j = -1 \quad & g(\mu_j) + 1 + r_j W = b \\ \alpha_j = 1, y_j = 1 \quad & g(\mu_j) - 1 - r_j W \leq b \\ \alpha_j = 1, y_j = -1 \quad & g(\mu_j) + 1 + r_j W \geq b \end{aligned} \tag{B.4}$$

Using these conditions, one can easy compute b_{low}, b_{up}, which are the greatest lower bound and the least upper bound on b. The proposed projected co-ordinate descent based algorithm starts with some set of feasible α_j. At every iteration b_{low}, b_{up} are

calculated. If $b_{low} \leq b_{up}$, then the KKT conditions are satisfied and hence the algorithm terminates. Else the maximum KKT violating pair, (l, m) is chosen and the values of α_l, α_m are updated such that the updation results in maximum decrease of the objective function: if $y_l = y_m$ then both α_l, α_m need to be incremented by $\Delta\alpha$ else α_l is incremented by $\Delta\alpha$ and α_m is incremented by $-\Delta\alpha$ in order to satisfy $\mathbf{y}^\top\alpha = 0$. The constraints $0 \leq \alpha \leq 1$ give bounds on $\Delta\alpha$ i.e., $\exists\ lb, ub\ \ni\ lb \leq \Delta\alpha \leq ub$. As mentioned earlier, $\Delta\alpha$ is chosen such that we get maximum decrease in objective function. This can be written as the following 1-d minimization problem:

$$\begin{aligned} \min_{\Delta\alpha} \quad & \sqrt{a(\Delta\alpha)^2 + 2b(\Delta\alpha) + c} - e\Delta\alpha \\ \text{s.t.} \quad & lb \leq \Delta\alpha \leq ub \end{aligned} \tag{B.5}$$

where $a = W^2(\mathbf{Q}(l,l) + 2s\mathbf{Q}(l,m) + \mathbf{Q}(m,m))$, $b = W^2\alpha^\top(\mathbf{Q}_l + s\mathbf{Q}_m)$, $c = W^2\alpha^\top\mathbf{Q}\alpha$ and $e = d_l + sd_m$. $s = 1$ if $y_l = y_m$ and $s = -1$ otherwise. As shown in section 5.1, the minimization problem has an analytic solution. Once the optimum value of $\Delta\alpha$ is calculated, α is updated accordingly and the procedure is repeated in the next iteration.

The iterative algorithm can be summarized as follows:

1. Initialize α with some feasible values.
2. Calculate b_{low}, b_{up}. If KKT conditions are satisfied i.e., $b_{low} \leq b_{up}$ then terminate, else continue.
3. Identify the maximum KKT violating pair (l, m).
4. Solve (B.5) to get the optimal value of $\Delta\alpha$. Update Lagrange multipliers of the maximum KKT violating pair and repeat step 2.

Appendix C

Dual of Large-Scale OR Formulation

This section derives dual of the primal formulation (4.4). Using the dual norm $\|\mathbf{w}\|_2 = \sup_{\|\mathbf{u}\|_2 \leq 1} \mathbf{w}^\top \mathbf{u}$, the Lagrangian function can be written as:

$$\begin{aligned} \mathcal{L} &= \sum_{i=1}^{r} \sum_{j=1}^{m_i} \left\{ \xi_i^j + \xi_i^{*j} + \alpha_i^j C_i^j \right. \\ &+ \left. \alpha_i^{*j} C_i^{*j} - \beta_i^j \xi_i^j - \beta_i^{*j} \xi_i^{*j} \right\} \\ &- \sum_{i=1}^{r} \gamma_i (b_i - b_{i-1}) + \rho(\mathbf{w}^\top \mathbf{u} - W) \end{aligned} \tag{C.1}$$

where the Lagrange multipliers satisfy $\alpha_i^j \geq 0, \alpha_i^{*j} \geq 0,\ \beta_i^j \geq 0, \beta_i^{*j} \geq 0, \gamma_i \geq 0, \rho \geq 0, \|\mathbf{u}\|_2 \leq 1$ and $C_i^j = \mathbf{w}^\top \phi(\mu_i^j) - b_i + 1 - \xi_i^j + r_i^j W, C_i^{*j} = 1 - \xi_i^{*j} + r_i^j W + b_{i-1} - \mathbf{w}^\top \phi(\mu_i^j)$. The KKT conditions for optimality can be summarized as follows:

$$\begin{aligned} \nabla_{\mathbf{w}} \mathcal{L} = 0 \ &\Rightarrow\ \rho \mathbf{u} = \sum_{i=1}^{r} \sum_{j=1}^{m_i} \left(\alpha_i^{*j} - \alpha_i^j \right) \phi(\mu_i^j) \\ \frac{\partial \mathcal{L}}{\partial b_{i:1 \leq i \leq r-1}} = 0 \ &\Rightarrow\ \sum_{j=1}^{m_{i+1}} \alpha_{i+1}^{*j} + \gamma_{i+1} = \sum_{j=1}^{m_i} \alpha_i^j + \gamma_i \\ \frac{\partial \mathcal{L}}{\partial \xi_i^j} = 0, \frac{\partial \mathcal{L}}{\partial \xi_i^{*j}} = 0 \ &\Rightarrow\ \alpha_i^j + \beta_i^j = 1, \alpha_i^{*j} + \beta_i^{*j} = 1 \end{aligned}$$

$$\begin{aligned} \text{Complimentary Slackness} \quad &\Rightarrow \quad \alpha_i^j C_i^j = 0,\ \alpha_i^{*j} C_i^{*j} = 0 \\ &\Rightarrow \quad \beta_i^j \xi_i^j = 0,\ \beta_i^{*j} \xi_i^{*j} = 0 \\ &\Rightarrow \quad \gamma_i (b_i - b_{i-1}), \rho(\mathbf{w}^\top \mathbf{u} - W) \end{aligned} \tag{C.2}$$

Since $b_0 = -\infty, b_r = \infty$, the complimentary slackness conditions immediately show that at optimality $\alpha_1^{*j} = \alpha_r^j = \gamma_1 = \gamma_r = 0\ \forall\ j$. With these boundary conditions, one can easily eliminate the γ_i multipliers from the KKT conditions using $\frac{\partial \mathcal{L}}{\partial b_i} = 0$, giving the following conditions: $s_i^* \leq s_i,\ \forall\ i = 1, \ldots, r-2,\ s_{r-1}^* = s_{r-1}$ where $s_i = \sum_{k=1}^{i} \sum_{j=1}^{m_k} \alpha_i^k$ and $s_i^* = \sum_{k=2}^{i+1} \sum_{j=1}^{m_k} \alpha_i^{*k}$. Now let us denote the column vector containing the α_i^j by α and that containing α_i^{*j} by α^*. We once again note that the entries corresponding to $i = 1$ are zero in α^* and those corresponding to $i = r$ are zero in α. Also let us denote the vector containing $1 + r_i^j W$ with $\mathbf{d}$ and the matrix containing the dot products of centers $\phi(\mu_i^j)$ with each other as $\mathbf{K}$. Using this notation, one can write the dual of the clustering based OR formulation as given in (4.5).

Bibliography

[1] Charu C. Aggarwal, Fatima Al-Garawi, and Philip S. Yu. Intelligent Crawling on the World Wide Web with Arbitrary Predicates. In *Proceedings of the* 10^{th} *International Conference on World Wide Web*, pages 96–105, 2001.

[2] Barry C. Arnold and Robert M. Shavelle. Joint Confidence Sets for the Mean and Variance of a Normal Distribution. *The American Statistician*, 52(2):133–140, 1998.

[3] Francis R. Bach, David Heckerman, and Eric Horvitz. On the Path to an Ideal ROC Curve: Considering Cost Asymmetry in Learning Classifiers. In *Proceedings of the* 10^{th} *International Workshop on Artificial Intelligence and Statistics*, 2005.

[4] A. Ben-Tal and A. Nemirovski. Robust Convex Optimization. *Mathematics of Operations Research*, 23:769–805, 1988.

[5] A. Ben-Tal and A. Nemirovski. Robust Optimization — Methodology and Applications. *Math. Programming*, 92:453–480, 2002.

[6] A. Ben-Tal and A. Nemirovski. On Safe Tractable Approximations of Chance Constrained Linear Matrix Inequalities, 2006. Available online at: `http://www.optimization-online.org/DB_HTML/2006/10/1484.html`.

[7] Kristin P. Bennett and Erin J. Bredensteiner. Duality and Geometry in SVM Classifiers. In *Proceedings of the* 17^{th} *International Conference on Machine Learning*, pages 57–64. Morgan Kaufmann, San Francisco, CA, 2000.

[8] C. L. Blake and C. J. Merz. UCI Repository of Machine Learning Databases, 1998. http://www.ics.uci.edu/∼mlearn/MLRepository.html.

[9] C. Cardie and N. Howe. Improving Minority Class Prediction using Case Specific Feature Weights. In *Proceedings of the* 14^{th} *International Conference on Machine Learning*, pages 57–65. Morgan Kaufmann, 1997.

[10] S. Chakrabarti, K. Punera, and M. Subramanyam. Accelerated Focused Crawling through Online Relevance Feedback. In *Proceedings of the* 11^{th} *International Conference on World Wide Web*, pages 148–159, 2002.

[11] Soumen Chakrabarti, Martin van den Berg, and Byron Dom. Focused Crawling: A New Approach to Topic-specific Web Resource Discovery. *Computer Networks*, 31(11–16):1623–1640, 1999.

[12] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence and Research*, 16:321–357, 2002.

[13] Wei Chu and S. Sathiya Keerthi. New approaches to support vector ordinal regression. In *Proceedings of the* 22^{nd} *International Conference on Machine Learning*, pages 145–152, 2005.

[14] K. Crammer and Y. Singer. Pranking with Ranking. In *Advances in Neural Information Processing Systems*, volume 14, 2002.

[15] Brian D. Davison. Topical Locality in the Web. In *Proceedings of the* 23^{rd} *Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 272–279, 2000.

[16] Michelangelo Diligenti, Frans Coetzee, Steve Lawrence, C. Lee Giles, and Marco Gori. Focused Crawling using Context Graphs. In *Proceedings of the* 26^{th} *International Conference on Very Large Databases*, pages 527–534, 2000.

[17] E. Erdougan and G. Iyengar. An Active Set Method for Single-Cone Second-Order Cone Programs. *SIAM Journal on Optimization*, 17(2):459–484, 2006.

[18] M. Ferris and T. Munson. Interior-point Methods for Massive Support Vector Machines. *Journal of Optimization*, 13(3):783–804, 2003.

[19] R. Fletcher. *Practical Methods of Optimization*. John Wiley and Sons, New York, 1989.

[20] G. Fung and O. Mangasarian. Proximal Support Vector Classifiers. In *Proceedings of the* 7^{th} *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2001.

[21] D. Grangier and S. Bengio. Exploiting Hyperlinks to Learn a Retrieval Model. In *NIPS Workshop on Learning to Rank*, 2005.

[22] S. Har-Peled, D. Roth, and D. Zimak. Constraint Classification: A New Approach to Multiclass Classification and Ranking. In *Advances in Neural Information Processing Systems*, volume 15, 2002.

[23] R. Herbrich, T. Graepel, and K. Obermayer. Large Margin Rank Boundaries for Ordinal Regression. In *Advances in Large Margin Classifiers*, pages 115–132, 2000.

[24] K. Huang, H. Yang, I. King, M.R. Lyu, and L Chan. Biased Minimax Probability Machine for Medical Diagnosis. In *Proceedings of the* 8^{th} *International Symposium on Artificial Intelligence and Mathematics*, 2004.

[25] T. Joachims. Making Large-Scale SVM Learning Practical. In B. Schölkopf, C. J. C. Burges, and A. J. Smola, editors, *Advances in Kernel Methods—Support Vector Learning*, pages 169–184, Cambridge, MA, 1999. MIT Press.

[26] Thorsten Joachims. Training Linear SVMs in Linear Time. In *Proceedings of the* 12^{th} *ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 217–226, New York, NY, USA, 2006. ACM Press.

[27] Richard A. Johnson and Dean W. Wichern. *Applied Multivariate Statistical Analysis*. Prentice Hall; 5 edition, 2002.

[28] S. Keerthi and D. DeCoste. A Modified Finite Newton Method for Fast Solution of Large Scale Linear SVMs. *Journal of Machine Learning Research*, 6:341–361, 2005.

[29] J.M. Kleinberg. Authoritative Sources in a Hyperlinked Environment. *Journal of the ACM*, 46(5):604–632, 1999.

[30] I. Kononenko. Machine Learning for Medical Diagnosis: History, State of the Art and Perspective. *Artificial Intelligence in Medicine*, 23:89–109, 2001.

[31] Miroslav Kubat and Stan Matwin. Addressing the Curse of Imbalanced Training Sets: One-sided Selection. In *Proceedings of the* 14^{th} *International Conference on Machine Learning*, pages 179–186. Morgan Kaufmann, 1997.

[32] G. R Lanckriet, L. E. Ghaoui, C. Bhattacharyya, and M. I. Jordan. A Robust Minimax Approach to Classification. *Journal of Machine Learning Research*, 3:555–582, 2003.

[33] Anhua Lin and Shih-Ping Han. On the Distance Between Two Ellipsoids. *SIAM Journal of Optimization*, 13:298–308, 2002.

[34] M.S. Lobo, L. Vandenberghe, S. Boyd, and H. Lebret. Applications of Second-Order Cone Programming. *Linear Algebra and its Applications*, 284(1–3):193–228, 1998.

[35] O. Mangasarian and D. Musicant. Lagrangian Support Vector Machines. *Journal of Machine Learning Research*, 1:161–177, 2001.

[36] A. W. Marshall and I. Olkin. Multivariate Chebychev Inequalities. *Annals of Mathematical Statistics*, 31(4):1001–1014, 1960.

[37] J. Mercer. Functions of Positive and Negative Type and their Connection with the Theory of Integral Equations. In *Proceedings of the Royal Society of London. Series*

A, Containing Papers of a Mathematical and Physical Character, volume 83, pages 69–70, 1909.

[38] A. M. Mood. *Introduction to the Theory of Statistics*. McGraw-Hill Book Co., 1950.

[39] Y. Nesterov and A. Nemirovskii. Interior Point Algorithms in Convex Programming. *Studies in Applied Mathematics, SIAM*, (13), 1993.

[40] J. Platt. Fast Training of Support Vector Machines using Sequential Minimal Optimization. In *Advances in Kernel Methods—Support Vector Learning*, pages 185–208, 1999.

[41] Foster Provost. Learning from Imbalanced Data Sets. In *Proceedings of the* 17^{th} *National Conference on Artificial Intelligence*, 2000.

[42] Henry Scheffé. *The Analysis of Variance*. Wiley-IEEE, 1959.

[43] A. Shashua and A. Levin. Ranking with Large Margin Principle: Two Approaches. In *Advances in Neural Information Processing Systems*, volume 15, 2003.

[44] P. K. Shivaswamy, C. Bhattacharyya, and A. J. Smola. Second Order Cone Programming Approaches for Handling Missing and Uncertain Data. *Journal of Machine Learning Research*, 7:1283–1314, 2006.

[45] J.F. Sturm. Using SeDuMi 1.02, A MATLAB Toolbox for Optimization over Symmetric Cones. *Optimization Methods and Software*, 11–12:625–653, 1999.

[46] V. Vapnik. *Statistical Learning Theory*. John Wiley and Sons, New York, 1998.

[47] H. Yu, J. Yang, and J. Han. Classifying Large Data Sets using SVM with Hierarchical Clusters. In *Proceedings of the* 9^{th} *ACM SIGKDD international conference on Knowledge discovery and data mining*, 2003.

[48] T. Zhang, R. Ramakrishnan, and M. Livny. BIRCH: An Efficient Data Clustering Method for Very Large Databases. In *Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data*, pages 103–114, 1996.