Research Article

A Robust Probability Classifier Based on the Modified χ^2 -Distance

Yongzhi Wang,¹ Yuli Zhang,² Jining Yi,³ Honggang Qu,^{3,4} and Jinli Miu^{3,4}

¹ College of Instrumentation & Electrical Engineering, Jilin University, Changchun 130061, China

² Department of Automation, TNLIST, Tsinghua University, Beijing 100084, China

³ Development and Research Center of China Geological Survey, Beijing 100037, China

⁴ Key Laboratory of Geological Information Technology, Ministry of Land and Resources, Beijing 100037, China

Correspondence should be addressed to Yongzhi Wang; iamwangyongzhi@126.com

Received 9 January 2014; Revised 5 April 2014; Accepted 7 April 2014; Published 30 April 2014

Academic Editor: Hua-Peng Chen

Copyright © 2014 Yongzhi Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We propose a robust probability classifier model to address classification problems with data uncertainty. A class-conditional probability distributional set is constructed based on the modified χ^2 -distance. Based on a “linear combination assumption” for the posterior class-conditional probabilities, we consider a classification criterion using the weighted sum of the posterior probabilities. An optimal robust minimax classifier is defined as the one with the minimal worst-case absolute error loss function value over all possible distributions belonging to the constructed distributional set. Based on the conic duality theorem, we show that the resulted optimization problem can be reformulated into a second order cone programming problem which can be efficiently solved by interior algorithms. The robustness of the proposed model can avoid the “overlearning” phenomenon on training sets and thus keep a comparable accuracy on test sets. Numerical experiments validate the effectiveness of the proposed model and further show that it also provides promising results on multiple classification problems.

1. Introduction

Statistics classification has been extensively studied in the field of machine learning and statistics. A typical classification problem is to design a linear or nonlinear classifier based on a known training set such that a new observation can be assigned to one of the known classes. Many classification models have been proposed, such as the naive Bayes classifiers (NBC) [1, 2], artificial neural network [3], and support vector machines (SVM) [4].

In real-world classification problems, it is often the case that the data of training set are imprecise due to unavoidable observational noises in the process of data collection or data approximation from incomplete samples. One way to handle the data uncertainty is to design a robust classifier in the sense that it has the minimal worst-case misclassification probability for the training sets. The idea of robustness has been widely applied in many traditional machine learning and statistics techniques, such as robust Bayes classifiers [5],

robust support vector machines [6], and robust quadratic regressions [7]. Robust classifiers are highly related to the recently flourished research on robust optimization. For more recent developments on robust optimization, we refer the readers to the excellent book [8] and reviews [9, 10].

Recently [11, 12] have proposed a robust minimax approach called the minimax probability machine to design a binary classifier. Unlike the traditional methods, they make no assumption on the class-conditional distributions, but only the mean and covariance matrix of each class are assumed to be known. Under this assumption, the designed classifier is determined by minimizing the worst-case probability of misclassification under all possible choices of class-conditional distributions with the given mean and covariance matrix. By reformulating the classifier design problem into second order cone programming, they show that the computational complexity of the proposed approach is similar to that of SVM. Because of its computational advantage and competitive performance with other current methods, this

approach has been further extended to incorporating other features. El Ghaoui et al. [13] propose a robust classification model by minimizing the worst-case value of a given loss function over all possible choices of the data in a bounded hyperrectangles. Three loss functions from SVM, logistic regressions, and minimax probability machines are studied in [13]. Based on the same assumption of known mean and covariance matrix, [14, 15] propose the biased minimax probability machines to address the biased classification problem and further generalize it to obtain the minimum error minimax probability machines. Hoi and Lyu [16] study a quadratic classifier with positive definite covariance matrices and further consider the problem of finding a convex set to cover known sampled data in one class while minimizing the worst-case misclassification probability. The minimax probability machines have also been extended to solve multiple classification problems; see [17, 18].

In this paper, we propose a robust probability classifier (RPC) based on the modified χ^2 -distance. Specifically, for a given training set, we first estimate the probability of each sample belonging to each class based on a feature, which is called a nominal class-conditional distribution. Then a ϵ -confidence probability distributional set P_ϵ is constructed based on the nominal class-conditional distributions and the modified χ^2 -distance, where parameter ϵ controls the size of the constructed set. Unlike the “conditional independence assumption” in NBC, we introduce a “linear combination assumption” for the posterior class-conditional probabilities, and the proposed classifier takes a linear combination form of these probabilities based on different features and it will assign the sample to the class with the maximal posterior probability. To get a robust classifier, we minimize the worst-case loss function value over all possible choices of class-conditional distributions over the distributional set P_ϵ . The underlying assumption is that, due to observational noises, we cannot obtain the true probability distribution of each class, but it can be well estimated by the nominal distribution such that it belongs to the distributional set P_ϵ .

Our two major contributions are as follows. First in our model, the proposed distributional set P_ϵ is based on the nominal distribution and the modified χ^2 -distance. As pointed in [19], such distributional set can make use of more information conveyed in the training set compared with traditional robust approaches which only use the information of mean and covariance matrix. To the best of our knowledge, this is among the first study of classification models considering complex distribution information. Although [20] considers a ϵ -contaminated robust support vector machine model, its distributional set is defined by easily handled linear constraints and its analysis is highly dependent on characterization of the extreme points of this set. Here our proposed distributional set is defined by nonlinear quadratic function and is analyzed by the conic duality theorem. Second by taking the absolute error function as the loss function, we show how to transform our robust minimax optimization problem into computable second order cone programming. The absolute error function in the objective function also distinguishes our model from other existing models, such

as the soft-margin support vector machine which uses the Hinge loss function [21, 22] and the logistic regression which uses the negative log likelihood function [23]. Note that the absolute error function is essential in our model to obtain a tractable optimization problem for the proposed model. Numerical experiments on real-world application validate the effectiveness of the proposed classifier and further show that the proposed classifier also performs well for multiple classification problems.

The paper proceeds as follows. Section 2 introduces the proposed robust minimax probability classifier based on the modified χ^2 -distance and discusses how to construct the desired distributional set P_ϵ . Section 3 provides an equivalent reformulation by handling the robust constraints and robust objective separately. Numerical experiments on real-world data set are carried out to validate the effectiveness of the proposed classifier in Section 4. Section 5 concludes this paper and gives future research directions.

2. Classifier Models

In this section, a simple probability classifier is first presented and then it is extended to handle data uncertainty by introducing a distributional set P_ϵ . We also discuss how to construct this distributional set based on training data set.

Consider a multiclass multifeature classification problem in which each sample contains $|L|$ features, and there are $|J|$ classes and $|I|$ samples. Specifically, given a training set $(X, Y) \in \mathbb{R}^{|I| \times |L|} \times \{0, 1\}^{|I| \times |J|}$, where $x_{i,l}$ denotes the l th feature of the i th sample and $y_{i,j} = 1$ if the i th sample belongs to j th class; otherwise, $y_{i,j} = 0$. In the following context, we will also use the term x^i to denote the i th sample; that is, $x^i = (x_{i,1}, \dots, x_{i,|L|})$.

2.1. Probability Classifier. Bayes classifiers assign an observation x to the $j^*(x)$ th class which has the maximal posterior probability; that is,

$$j^*(x) = \arg \max_{j \in J} P(j | x), \quad (1)$$

and $P(j | x)$ is the posterior probability function, that is, the conditional probability that the sample belongs to the j th class, given that we know it has feature vector x .

Using Bayes' theorem, we have

$$P(j | x) = \frac{P(j) P(x | j)}{P(x)} \propto P(j) P(x | j), \quad (2)$$

where $P(j)$ is the prior probability of the j th class, $P(x | j)$ is the conditional probability for the j th class, and $P(x)$ is the probability that a sample has feature vector x . Note that $P(x)$ is a constant if the values of the feature variables are known and thus can be omitted. To design an effective Bayes classifier, the key issue is estimating the class-conditional probability $P(x | j)$ or the joint probability $P(x, j)$. Theoretically, using the chain rule, we have

$$P(x, j) = P(j) P(x_1 | j) P(x_2 | j, x_1) \cdots P(x_{|L|} | j, x_1, \dots, x_{|L|-1}). \quad (3)$$

However such estimating method leads to the problem of “dimension disaster.”

To address this issue, the naive Bayes classifier makes the following “conditional independence assumption”:

$$P(x | j) = \prod_{l=1}^{|L|} p_j^l(x), \quad (4)$$

where $p_j^l(x) := P(x_l | j)$ is the class-conditional probability that the observation x belongs to the j th class based on the l th feature. Here we introduce another “linear combination assumption” for the class-conditional probability:

$$P(x | j) = \sum_{l=1}^{|L|} \beta_j^l p_j^l(x), \quad (5)$$

where β_j^l is a coefficient. Compared with the “conditional independence assumption,” which uses the probabilistic information in terms of multiplication, the proposed “linear combination assumption” uses the probabilistic information in terms of weighted sum. We will further discuss the rationality of this assumption at the end of this subsection.

Under this assumption, we have

$$P(j | x) \propto P(j) P(x | j) = P(j) \sum_{l=1}^{|L|} \beta_j^l p_j^l(x) = \sum_{l=1}^{|L|} \alpha_j^l p_j^l(x), \quad (6)$$

where $\alpha_j^l := P(j)\beta_j^l$ denotes the probability weight of the l th feature for the j th class.

To obtain the optimal probability classifier based on the “linear combination assumption,” it is natural to consider the following optimization problem:

$$\min_{\alpha \in \Theta} \sum_{j \in J} \sum_{i \in I} L(P(j | x^i), y_{i,j}), \quad (7)$$

where $L(\cdot, \cdot) : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}_+$ is a prespecified loss function. In the following context, we will take the absolute error function as our loss function; that is, $L(x, y) = |x - y|$. In view of its probability property, it is straightforward to impose the following constraints on the posterior probability:

$$0 \leq f(j | x^i) \leq 1, \quad \forall i \in I, j \in J. \quad (8)$$

Under such constraints, we have that

$$\begin{aligned} & \sum_{j \in J} \sum_{i \in I} L(f(j | x^i), y_{i,j}) \\ &= \sum_{j \in J} \sum_{i \in I} |f(j | x^i) - y_{i,j}| \end{aligned}$$

$$\begin{aligned} &= \sum_{j \in J} \sum_{i \in I} \{y_{i,j} (1 - f(j | x^i)) + (1 - y_{i,j}) f(j | x^i)\} \\ &= \sum_{j \in J} \sum_{i \in I} (1 - 2y_{i,j}) f(j | x^i) + |I|, \end{aligned} \quad (9)$$

where $|I| = \sum_{j \in J} \sum_{i \in I} y_{i,j}$.

Thus the optimal probability classifier (PC) problem can be formulated as follows:

$$\begin{aligned} \text{(PC)} \quad & \min \sum_{j \in J} \sum_{i \in I} (1 - 2y_{i,j}) \sum_{l \in L} \alpha_j^l p_{i,j}^l + |I| \\ \text{s.t.} \quad & 0 \leq \sum_{l \in L} \alpha_j^l p_{i,j}^l \leq 1, \quad \forall i \in I, j \in J. \end{aligned} \quad (10)$$

It is no doubt that the “linear combination assumption” may not work sometimes. However, we justify the proposed classifier by the following facts.

- (1) As an intuitive interpretation, note that $p_j^l(x)$ estimates the probability of the observation x belonging to the j th class only based on the l th feature; thus it provides partial probabilistic information of the sample. Hence we can interpret the weight α_j^l as certain degree of trust on the information, and in this sense, the “linear combination assumption” is a way of combining evidence from different sources. Similar ideas can also be found in the theory of evidence; see the Dempster-Shafer theory [24, 25].
- (2) In terms of the classification performance, in the worst case, the proposed classifier may put all weight on one feature; thus in such case, it is equivalent to a Bayes classifier based on a well-selected feature. If each class has its “typical” feature which can distinguish it from other classes, the proposed classifier has the ability to learn this property by putting different weights on different features for different classes and thus provides better classification performance. A real-life application on lithology classification problems also validates its classification performance by comparison with support vector machines and the naive Bayes classifier.
- (3) Another advantage of the proposed classifier is its high computability. As we show in Section 3, the proposed classifier and its robust counterpart problems can be reformulated as second order cone programming problems and thus can be solved by interior algorithms in polynomial time.

2.2. Robust Probability Classifier. Due to observational noises, the true class-conditional probability distribution is often difficult to obtain. Instead we can construct a confidence distributional set which contains the true distribution. Unlike the traditional distributional sets in minimax probability machines, which only utilize mean and covariance matrix, we construct our class-conditional probability distributional set based on the modified χ^2 -distance which uses more information from the samples.

The modified χ^2 -distance $d(\cdot, \cdot) : \mathbb{R}^m \times \mathbb{R}^m \rightarrow \mathbb{R}$ is used to measure the distance between two discrete probability distribution vectors in statistics. For given $p = (p_1, \dots, p_m)^T$ and $q = (q_1, \dots, q_m)^T$, it is defined as

$$d(q, p) = \sum_{j=1}^m \frac{(q_j - p_j)^2}{p_j}. \quad (11)$$

Based on the modified χ^2 -distance, we present the following class-conditional probability distributional set:

$$P_\epsilon = \left\{ q_{i,j}^l : \sum_j q_{i,j}^l = 1, q_{i,j}^l \geq 0, \sum_{j \in J} \frac{(q_{i,j}^l - p_{i,j}^l)^2}{p_{i,j}^l} \leq \epsilon, \right. \\ \left. \forall i \in I, l \in L, j \in J \right\}, \quad (12)$$

where $p_{i,j}^l$ is the nominal class-conditional distribution probability for the i th sample belonging to the j th class based on the l th feature and the prespecified parameter ϵ is used to control the size of the set.

To design a robust classifier, we need to consider the effect of data uncertainty on the objective function and constraints. The robust objective function is to minimize the worst-case loss function value over all the possible distributions in the distributional set P_ϵ ; the robust constraints ensure that all the original constraints should also be satisfied for any distribution in P_ϵ . Thus the robust probability classifier problem is of the following form:

$$\begin{aligned} \text{(RPC)} \quad \min \quad & \left\{ \max_{j \in J} \sum_{i \in I} (1 - 2y_{i,j}) \sum_{l \in L} \alpha_j^l q_{i,j}^l \right. \\ & \left. + |I| : \{q_{i,j}^l\} \in P_\epsilon \right\} \\ \text{s.t.} \quad & 0 \leq \sum_{l \in L} \alpha_j^l q_{i,j}^l \leq 1, \quad \forall \{q_{i,j}^l\} \in P_\epsilon \forall i, j. \end{aligned} \quad (13)$$

Note that the above optimization problem has an infinite number of robust constraints and its objective function is also an embedded subproblem. We will show how to solve such minimax optimization problem in Section 3.

2.3. Construct the Distributional Set. To get the distributional set P_ϵ , we need to define the parameter ϵ and the nominal probability $p_{i,j}^l$. The selection of parameter ϵ is application based and we will discuss this issue in the numerical experiment section; next we will provide a procedure to calculate $p_{i,j}^l$.

For the l th feature, the following procedure takes an integer K_l indicating the number of data intervals as an input and will output the estimated probability $p_{i,j}^l$ of the i th sample belonging to the j th class.

- (1) Sort samples in the increased order and divide them into K_l intervals such that each interval has at least $\lfloor |I|/K_l \rfloor$ number of samples. Denote the k th interval by $\Delta_{l,k}$.
- (2) Calculate the total number of samples in the j -class, N_j , the total number of samples in the k th interval, $N_{l,k}$, and the total number of samples belonging to the j -class in the k th interval, $N_{l,k,j}$.
- (3) For the i th sample, if it falls into the k th interval, the class-conditional probability $p_{i,j}^l$ is calculated by

$$\begin{aligned} p_{i,j}^l &= \text{Prob}(i \in j \mid x_{i,l} \in \Delta_{l,k}) \\ &= \frac{\text{Prob}(i \in j, x_{i,l} \in \Delta_{l,k})}{\text{Prob}(x_{i,l} \in \Delta_{l,k})} \\ &= \frac{\text{Prob}(i \in j) \text{Prob}(x_{i,l} \in \Delta_{l,k} \mid i \in j)}{\sum_{j' \in J} \text{Prob}(i \in j') \text{Prob}(x_{i,l} \in \Delta_{l,k} \mid i \in j')} \\ &= \frac{(N_j / |I|) \cdot (N_{l,k,j} / N_j)}{\sum_{j' \in J} (N_{j'} / |I|) \cdot (N_{l,k,j'} / N_{j'})} \\ &= \frac{N_{l,k,j}}{N_{l,k}}. \end{aligned} \quad (14)$$

Note that from the definition of P_ϵ , we easily compute the upper bound $\bar{q}_{i,j}^l$ and lower bound $\underline{q}_{i,j}^l$ for the true class-conditional probability $q_{i,j}^l$ as follows:

$$\begin{aligned} \bar{q}_{i,j}^l &= \max \left\{ q_{i,j}^l : \sum_s q_{i,s}^l = 1, \right. \\ & \quad \left. \sum_{s \in J} \frac{(q_{i,s}^l - p_{i,s}^l)^2}{p_{i,s}^l} \leq \epsilon, q_{i,s}^l \geq 0, \forall s \in J \right\}, \\ \underline{q}_{i,j}^l &= \min \left\{ q_{i,j}^l : \sum_s q_{i,s}^l = 1, \right. \\ & \quad \left. \sum_{s \in J} \frac{(q_{i,s}^l - p_{i,s}^l)^2}{p_{i,s}^l} \leq \epsilon, q_{i,s}^l \geq 0, \forall s \in J \right\}. \end{aligned} \quad (15)$$

The above problems can be efficiently solved by a second order cone solver such as SeDuMi [26] or SDPT3 [27].

3. Solution Methods for RPC

In this section, we first reduce the infinite number of robust constraints to a finite set of linear constraints and then transform the inner robust objective function into a minimization problem by the conic duality theorem. At last, we obtain an equivalent computable second order cone programming for the RPC problem. The following analysis is based on the strong duality result in [8].

Consider a conic program of the following form:

$$\begin{aligned}
 \text{(CP)} \quad & \min \quad c^T x \\
 \text{s.t.} \quad & A_i x - b_i \in C_i, \quad \forall i = 1, \dots, m, \\
 & Ax = b,
 \end{aligned} \quad (17)$$

and its dual problem

$$\begin{aligned}
 \text{(DP)} \quad & \max \quad b^T z + \sum_{i=1}^m b_i^T y_i \\
 \text{s.t.} \quad & A^* z + \sum_{i=1}^m A_i^* y_i = c, \\
 & y_i \in C_i^*, \quad \forall i = 1, \dots, m,
 \end{aligned} \quad (18)$$

where C_i is a cone in $\mathbb{R}^{n_i}$ and C_i^* is its dual cone defined by

$$C_i^* = \{y \in \mathbb{R}^{n_i} : y^T x \geq 0, \forall x \in C_i\}. \quad (19)$$

A conic program is called strictly feasible if it admits a feasible solution $\bar{x}$ such that $A_i \bar{x} - b_i \in \text{int } C_i$, $\forall i = 1, \dots, m$, where $\text{int } C_i$ denotes the interior point set of C_i .

Lemma 1 (see [8]). *If one of the problems (CP) and (DP) is strictly feasible and bounded, then the other problem is solvable, and (CP) = (DP) in the sense that both have the same optimal objective function value.*

3.1. Robust Constraints. The following lemma provides an equivalent characterization for the infinite number of robust constraints in terms of a finite set of linear constraints which can be solved efficiently.

Lemma 2. *For given i, j , the robust constraint*

$$0 \leq \sum_{l \in L} \alpha_j^l p_{i,j}^l \leq 1, \quad \forall \{q_{i,j}^l\} \in P_\epsilon \quad (20)$$

is equal to the following constraints:

$$\begin{aligned}
 & \sum_{l \in L} \left(q_{i,j}^l u_{i,j}^{l,0} - \bar{q}_{i,j}^l v_{i,j}^{l,0} \right) \geq 0, \\
 & \alpha_{i,j}^l - u_{i,j}^{l,0} + v_{i,j}^{l,0} \geq 0, \quad u_{i,j}^{l,0}, v_{i,j}^{l,0} \geq 0, \quad \forall l \in L, \\
 & 1 + \sum_{l \in L} \left(\underline{q}_{i,j}^l u_{i,j}^{l,1} - \bar{q}_{i,j}^l v_{i,j}^{l,1} \right) \geq 0, \\
 & v_{i,j}^{l,1} - \alpha_{i,j}^l - u_{i,j}^{l,1} \geq 0, \quad u_{i,j}^{l,1}, v_{i,j}^{l,1} \geq 0, \quad \forall l \in L.
 \end{aligned} \quad (21)$$

Proof. First note that the distributional set $P_{\epsilon,i}$ can be represented as the Cartesian product of a series of projected subsets

$$P_\epsilon = \prod_{i \in I} P_{\epsilon,i}, \quad (22)$$

where the projected subset on index i is defined by

$$P_{\epsilon,i} = \left\{ q_{i,j}^l : \sum_j q_{i,j}^l = 1, q_{i,j}^l \geq 0, \sum_{j \in J} \frac{(q_{i,j}^l - p_{i,j}^l)^2}{p_{i,j}^l} \leq \epsilon, \forall l \in L, j \in J \right\}. \quad (23)$$

Then for given i, j , since the robust constraint is only associated with variables $q_{i,j}^l, l \in L$, we can further split the projected subset $P_{\epsilon,i}$ into $|J|$ subsets

$$P_{\epsilon,i} = \prod_{j \in J} P_{\epsilon,i,j} = \prod_{j \in J} \left\{ q_{i,j}^l : \underline{q}_{i,j}^l \leq q_{i,j}^l \leq \bar{q}_{i,j}^l, \forall l \in L \right\}, \quad (24)$$

where $\bar{q}_{i,j}^l$ and $\underline{q}_{i,j}^l$ are computed by (15) and (16), respectively.

For constraint $\sum_{l \in L} \alpha_j^l p_{i,j}^l \geq 0, \forall \{q_{i,j}^l\} \in P_\epsilon$, it is equal to the following constraint:

$$\begin{aligned}
 & \sum_{l \in L} \alpha_j^l p_{i,j}^l \geq 0, \quad \forall \{q_{i,j}^l\} \in P_{\epsilon,i} \\
 \iff & \sum_{l \in L} \alpha_j^l p_{i,j}^l \geq 0, \quad \forall \{q_{i,j}^l\} \in P_{\epsilon,i,j} \\
 \iff & \min \left\{ \sum_{l \in L} \alpha_j^l p_{i,j}^l : \underline{q}_{i,j}^l \leq q_{i,j}^l \leq \bar{q}_{i,j}^l, \forall l \in L \right\} \geq 0 \\
 \iff & \max \left\{ \sum_{l \in L} \left(\underline{q}_{i,j}^l u_{i,j}^{l,0} - \bar{q}_{i,j}^l v_{i,j}^{l,0} \right) : \right. \\
 & \left. \alpha_{i,j}^l - u_{i,j}^{l,0} + v_{i,j}^{l,0} \geq 0, u_{i,j}^{l,0}, v_{i,j}^{l,0} \geq 0, \forall l \in L \right\} \geq 0 \\
 \iff & \sum_{l \in L} \left(\underline{q}_{i,j}^l u_{i,j}^{l,0} - \bar{q}_{i,j}^l v_{i,j}^{l,0} \right) \geq 0, \\
 & \alpha_{i,j}^l - u_{i,j}^{l,0} + v_{i,j}^{l,0} \geq 0, \quad u_{i,j}^{l,0}, v_{i,j}^{l,0} \geq 0, \quad \forall l \in L,
 \end{aligned} \quad (25)$$

where the last equivalence comes from the strong duality between these two linear programs.

For the constraint $\sum_{l \in L} \alpha_j^l p_{i,j}^l \leq 1, \forall \{q_{i,j}^l\} \in P_\epsilon$, the same technique applies; thus we complete the proof. $\square$

3.2. Robust Objective Function. In the RPC problem, the robust objective function is defined by an inner maximization problem. The following proposition shows that it can be transformed into a minimization problem over second order cones. To prove the following result, we utilize the concept of conjugate function d^* of the modified χ^2 -distance:

$$d^*(s) = \sup_{t \geq 0} \{st - d(t)\} = \frac{[s+2]_+^2}{4} - 1, \quad (26)$$

where the function $[\cdot]_+$ is defined as $[x]_+ = x$ if $x \geq 0$; otherwise $[x]_+ = 0$. For more details about conjugate functions, see [28].

Proposition 3. *The following inner maximization problem*

$$\max \sum_{j \in J} \sum_{i \in I} (1 - 2y_{i,j}) \sum_{l \in L} \alpha_j^l q_{i,j}^l + |I| : \{q_{i,j}^l\} \in P_\epsilon \quad (27)$$

is equivalent to a second order cone programming

$$\begin{aligned} \min \quad & \sum_{i \in I} \sum_{l \in L} (\epsilon \lambda_i^l - \theta_i^l) + \sum_{i \in I} \sum_{l \in L} \sum_{j \in J} p_{i,j}^l w_{i,j}^l + |I| \\ \text{s.t.} \quad & \begin{pmatrix} w_{i,j}^l \\ z_{i,j}^l \\ 2\lambda_i^l + w_{i,j}^l \end{pmatrix} \in L^3, \quad \forall i \in I, j \in J, l \in L, \\ & u_{i,j}^l = \alpha_j^l (1 - 2I_{i,j}) + \theta_i^l, \quad \forall i \in I, l \in L, j \in J. \\ & z_{i,j}^l \geq r_{i,j}^l + 2\lambda_i^l, \quad \lambda_{i,j}^l, z_{i,j}^l \geq 0 \quad \forall i \in I, j \in J, l \in L, \end{aligned} \quad (28)$$

where a second order cone L^{n+1} is defined as

$$L^{n+1} = \left\{ x \in \mathbb{R}^{n+1} : x_{n+1} \geq \sqrt{\sum_{i=1}^n x_i^2} \right\}. \quad (29)$$

Proof. For given feasible α satisfying the robust constraints, it is straightforward to show that the inner maximum problem is equal to the following minimization problem (MP):

$$\begin{aligned} \text{(MP)} \quad \min \quad & t \\ \text{s.t.} \quad & t \geq \sum_{j \in J} \sum_{i \in I} \left\{ (1 - 2y_{i,j}) \sum_{l \in L} \alpha_j^l q_{i,j}^l \right\} + |I|, \\ & \forall \{q_{i,j}^l\} \in P_\epsilon. \end{aligned} \quad (30)$$

The above constraint can be further reduced to the following constraint:

$$\begin{aligned} \max \quad & \left\{ \sum_{j \in J} \sum_{i \in I} \left\{ (1 - 2y_{i,j}) \sum_{l \in L} \alpha_j^l q_{i,j}^l \right\} \right. \\ & \left. + |I| - t : \forall \{q_{i,j}^l\} \in P_\epsilon \right\} \leq 0. \end{aligned} \quad (31)$$

By assigning Lagrange multipliers $\theta_i^l \in \mathbb{R}$ and $\lambda_i^l \in \mathbb{R}_+$ to the constraints in the left optimization problem, we obtain the following Lagrange function:

$$\begin{aligned} L(q, \theta, \lambda) = & \sum_{i \in I} \sum_{l \in L} (\epsilon \lambda_i^l - \theta_i^l) \\ & + \sum_{i \in I} \sum_{l \in L} \sum_{j \in J} \left(r_{i,j}^l q_{i,j}^l - \lambda_i^l \frac{(q_{i,j}^l - p_{i,j}^l)^2}{p_{i,j}^l} \right) \\ & + |I| - t, \end{aligned} \quad (32)$$

where $r_{i,j}^l = \alpha_j^l (1 - 2y_{i,j}) + \theta_i^l$. Its dual function is given as

$$\begin{aligned} D(\theta, \lambda) = & \max_{q \geq 0} L(t, q, \theta, \lambda) \\ = & \sum_{i \in I} \sum_{l \in L} (\epsilon \lambda_i^l - \theta_i^l) \\ & + \sum_{i \in I} \sum_{l \in L} \sum_{j \in J} \max_{q_{i,j}^l \geq 0} \left(r_{i,j}^l q_{i,j}^l - \lambda_i^l p_{i,j}^l \left(\frac{q_{i,j}^l - p_{i,j}^l}{p_{i,j}^l} \right)^2 \right) \\ & + |I| - t, \\ = & \sum_{i \in I} \sum_{l \in L} (\epsilon \lambda_i^l - \theta_i^l) \\ & + \sum_{i \in I} \sum_{l \in L} \sum_{j \in J} p_{i,j}^l \max_{t \geq 0} \left(r_{i,j}^l t - \lambda_i^l (t - 1)^2 \right) + |I| - t, \\ = & \sum_{i \in I} \sum_{l \in L} (\epsilon \lambda_i^l - \theta_i^l) \\ & + \sum_{i \in I} \sum_{l \in L} \sum_{j \in J} p_{i,j}^l \lambda_i^l \max_{t \geq 0} \left(\frac{r_{i,j}^l}{\lambda_i^l} t - (t - 1)^2 \right) + |I| - t, \\ = & \sum_{i \in I} \sum_{l \in L} (\epsilon \lambda_i^l - \theta_i^l) \\ & + \sum_{i \in I} \sum_{l \in L} \sum_{j \in J} p_{i,j}^l \lambda_i^l d^* \left(\frac{r_{i,j}^l}{\lambda_i^l} \right) + |I| - t. \end{aligned} \quad (33)$$

Note that, for any feasible α , the primal maximization problem (31) is bounded and has a strictly feasible solution $\{p_{i,j}^l\}$; thus there is no duality gap between (31) and the following dual problem:

$$\begin{aligned} \min \quad & D(\theta, \lambda) : \theta_i^l \in \mathbb{R}, \lambda_i^l \in \mathbb{R}_+, \forall i \in I, l \in L \\ \Leftrightarrow \quad & \left\{ \begin{array}{l} \min \quad \sum_{i \in I} \sum_{l \in L} (\epsilon \lambda_i^l - \theta_i^l) + \sum_{i \in I} \sum_{l \in L} \sum_{j \in J} p_{i,j}^l w_{i,j}^l + |I| - t \\ \text{s.t.} \quad w_{i,j}^l \geq \lambda_i^l d^* \left(\frac{r_{i,j}^l}{\lambda_i^l} \right), \quad \forall i \in I, l \in L, j \in J, \\ \theta_i^l \in \mathbb{R}, \quad \lambda_i^l \in \mathbb{R}_+, \quad \forall i \in I, l \in L \end{array} \right\}. \end{aligned} \quad (34)$$

Next we show that the constraint about the conjugate function can be represented by second order cone constraints:

$$\begin{aligned} \lambda_i^l d^* \left(\frac{r_{i,j}^l}{\lambda_i^l} \right) \leq w_{i,j}^l & \Leftrightarrow \lambda_i^l \left(-1 + \frac{1}{4} \left[\frac{r_{i,j}^l}{\lambda_i^l} + 2 \right]_+^2 \right) \leq w_{i,j}^l \\ & \Leftrightarrow 4\lambda_i^l (\lambda_i^l + w_{i,j}^l) \geq [r_{i,j}^l + 2\lambda_i^l]_+^2 \\ & \Leftrightarrow 4\lambda_i^l (\lambda_i^l + w_{i,j}^l) \geq (z_{i,j}^l)^2, \\ & z_{i,j}^l \geq 0, \quad z_{i,j}^l \geq r_{i,j}^l + 2\lambda_i^l, \end{aligned}$$

$$\begin{aligned} &\Leftrightarrow \begin{pmatrix} w_{i,j}^l \\ z_{i,j}^l \\ 2\lambda_i^l + w_{i,j}^l \end{pmatrix} \in L^3, \\ &z_{i,j}^l \geq 0, \quad z_{i,j}^l \geq r_{i,j}^l + 2\lambda_i^l. \end{aligned} \quad (35)$$

By reinjecting the above constraints into (MP), the robust objective function is equivalent to the following problem:

$$\begin{aligned} \min \quad & t \\ \text{s.t.} \quad & \sum_{i \in I} \sum_{l \in L} (\epsilon \lambda_i^l - \theta_i^l) + \sum_{i \in I} \sum_{l \in L} \sum_{j \in J} p_{i,j}^l w_{i,j}^l + |I| \leq t, \\ & \begin{pmatrix} w_{i,j}^l \\ z_{i,j}^l \\ 2\lambda_i^l + w_{i,j}^l \end{pmatrix} \in L^3, \\ & z_{i,j}^l \geq r_{i,j}^l + 2\lambda_i^l, \quad z_{i,j}^l, \lambda_{i,j}^l \geq 0, \quad \forall i \in I, j \in J, l \in L, \\ & u_{i,j}^l = \alpha_j^l (1 - 2I_{i,j}) + \theta_i^l, \quad \forall i \in I, l \in L, j \in J. \end{aligned} \quad (36)$$

By eliminating variable t , we complete the proof. $\square$

Based on the Lemma 2 and Proposition 3, we obtain our main result.

Proposition 4. *The RPC problem can be solved as the following second order cone programming:*

$$\begin{aligned} \min \quad & \sum_{i \in I} \sum_{l \in L} (\epsilon \lambda_i^l - \theta_i^l) + \sum_{i \in I} \sum_{l \in L} \sum_{j \in J} p_{i,j}^l w_{i,j}^l + |I| \\ \text{s.t.} \quad & \begin{pmatrix} w_{i,j}^l \\ z_{i,j}^l \\ 2\lambda_i^l + w_{i,j}^l \end{pmatrix} \in L^3, \quad \forall i \in I, j \in J, l \in L, \\ & r_{i,j}^l = \alpha_j^l (1 - 2I_{i,j}) + \theta_i^l, \quad \forall i \in I, j \in J, l \in L, \\ & z_{i,j}^l \geq r_{i,j}^l + 2\lambda_i^l, \quad \forall i \in I, j \in J, l \in L, \\ & \sum_{l \in L} (q_{i,j}^l u_{i,j}^{l,0} - \bar{q}_{i,j}^l v_{i,j}^{l,0}) \geq 0, \quad \forall i \in I, j \in J, \\ & 1 + \sum_{l \in L} (q_{i,j}^l u_{i,j}^{l,1} - \bar{q}_{i,j}^l v_{i,j}^{l,1}) \geq 0, \quad \forall i \in I, j \in J, \\ & \alpha_{i,j}^l - u_{i,j}^{l,0} + v_{i,j}^{l,0} \geq 0, \quad \forall i \in I, j \in J, l \in L, \\ & v_{i,j}^{l,1} - \alpha_{i,j}^l - u_{i,j}^{l,1} \geq 0, \quad \forall i \in I, j \in J, l \in L, \\ & \lambda_{i,j}^l, z_{i,j}^l, u_{i,j}^{l,1}, v_{i,j}^{l,1}, u_{i,j}^{l,0}, v_{i,j}^{l,0} \geq 0, \quad \forall i \in I, j \in J, l \in L, \\ & r_{i,j}^l, \theta_i^l, w_{i,j}^l, \alpha_{i,j}^l \in \mathbb{R}, \quad \forall i \in I, j \in J, l \in L. \end{aligned} \quad (37)$$

4. Numerical Experiments on Real-World Applications

In this section, numerical experiments on real-world applications are carried out to verify the effectiveness of the proposed robust probability classifier model. Specifically we consider lithology classification data sets from our practical application. We compare our model with the regularized SVM (RSVM) and the naive Bayes classifier (NBC) on both binary and multiple classification problems.

All the numerical experiments are implemented in Matlab 7.7.0 and run on Intel(R) Core(TM) i5-4570 CPU. SDPT3 solver [27] is called to solve the second order cone programs in our proposed method and the regularized SVM.

4.1. Data Sets. Lithology classification is one of the basic tasks for geological investigation. To discriminate the lithology of the underground strata, various electromagnetic techniques are applied to the same strata to obtain different features, such as Gamma coefficients, acoustic wave, striation, density, and fusibility.

Here numerical experiments are carried out on a series of data sets: the borehole T1, Y4, Y5, and Y6. All boreholes are located in Tarim Basin, China. In total, there are 12 data sets used for binary classification problems and 8 data sets used for multiple classification problems. For each data set, based on a prespecified training rate $\gamma \in [0, 1]$, it is randomly partitioned into two subsets: a training set and a test set, such that the size of training set accounts for γ of the total number of samples.

4.2. Experiment Design. The parameters in our models are chosen based on the size of data set. The parameter ϵ depends on the number of the classes and defined as $\epsilon = \delta^2/|J|$, where $\delta \in (0, 1)$. The choice of ϵ can be explained in this way: if there are $|J|$ classes and the training data are uniformly distributed, then, for each probability $p_{i,j}^l = 1/|J|$, its maximal variation range is between $p_{i,j}^l(1 - \delta)$ and $p_{i,j}^l(1 + \delta)$. The number of data intervals K_l is defined as $K_l = |I|/(|J| \times K)$ such that if the training data are uniformly distributed, then in each data interval there are K samples in each class. In the following context, we set $\delta = 0.2$ and $K = 8$.

We compare the performances of the proposed RPC model with the following regularized support vector machine model [6] (take the j th class for example):

$$\begin{aligned} \text{(RSVM)} \quad \min \quad & \sum_{i \in I} \xi_{i,j} + \lambda_j \|w_j\| \\ \text{s.t.} \quad & \bar{y}_{i,j} \left(\sum_{l \in L} w_j^l x_i^l + b_j \right) \geq 1 - \xi_{i,j}, \quad i \in I, \\ & \xi_{i,j} \geq 0, \quad i \in I, \end{aligned} \quad (38)$$

where $\bar{y}_{i,j} = 2y_{i,j} - 1$ and $\lambda_j \geq 0$ is a regularization parameter. As pointed by [8], $\lambda_j \geq 0$ represents a trade-off between the number of training set errors and the amount of robustness

TABLE 1: Performances of RSVM, NBC, and RPC for binary classification problems on Y5 data set.

tr (%)	RSVM		NBC		RPC	
	Train (%)	Test (%)	Train (%)	Test (%)	Train (%)	Test (%)
50	90.7	88.2	63.9	66.2	88.4	90.5*
55	89.9	88.6	69.1	72.8	89.5	89.9*
60	89.0	85.0	70.3	72.1	91.3	86.4*
65	86.3	85.9	72.1	72.8	88.0	92.5*
70	92.3	84.1	70.3	75.7	90.8	86.3*
75	88.8	87.9	74.2	74.6	88.7	91.6*
80	88.7	93.8*	90.0	87.5	88.3	93.3
85	89.5	89.3	93.4	89.6	89.2	91.0*
90	89.5	88.4	93.3	95.8*	89.2	92.6

TABLE 2: Performances of RSVM, NBC, and RPC for binary classification problems on T1 data set.

tr (%)	RSVM		NBC		RPC	
	Train (%)	Test (%)	Train (%)	Test (%)	Train (%)	Test (%)
50	91.4	84.8	76.5	68.9	91.3	87.5*
55	92.5	86.6	68.0	77.0	92.0	90.3*
60	89.8	86.1	72.9	73.8	88.9	90.9*
65	91.0	82.3	80.5	81.6	89.8	92.9*
70	86.8	95.5*	83.4	89.8	88.4	93.7
75	89.4	85.2	85.9	79.5	89.7	93.5*
80	91.8	80.8	88.1	79.9	89.7	91.1*
85	88.3	89.9	89.9	92.8	90.8	97.1*
90	88.5	90.2	88.8	94.2	90.9	97.2*

with respect to spherical perturbations of the data points. To make a fair comparison, in the following experiments we will test a series of λ values and choose the one with best performance. Note that if $\lambda_j = 0$, we refer to this model as the classic support vector machine (SVM). See also [6] for more details on RSVM and its applications to multiple classification problems.

4.3. Test on Binary Classification. In this subsection, RSVM, NBC, and RPC are implemented on 12 data sets for the binary classification problems using the cross-validation methods. To improve the performances of RSVM, we transform the original data by the popularly used polynomial kernels [6].

Tables 1 and 2 show the averaged classification performances of RSVM, NBC, and the proposed RPC (over 10 randomly generated instances) for binary classification problems on Y5 and T1 data sets, respectively. For each data set, we randomly partition it into a training set and a test set based on the parameter tr which varies from 0.5 to 0.9. The highest classification accuracy on a training set among these three methods is highlighted in bold while the best classification accuracy on a test set is marked with an asterisk

Tables 1 and 2 validate the effectiveness of the proposed RPC for binary classification problems compared with NBC and RSVM. Specifically, for most of the cases, RSVM has the highest classification accuracy on training sets but its performance on test sets is unsatisfactory. For most of the cases, the proposed RPC provides the highest classification

accuracy on test sets. NBC provides better performances on test sets as the training rate increases. The experimental results also show that for given training rate, RPC can provide better performances on test sets than that on training sets; thus it can avoid the “overlearning” phenomenon.

To further validate the effectiveness of the proposed RPC, we test it on additional 10 data sets, that is, T41–T45 and T61–T65. Table 3 reports the averaged performances of three methods over 10 randomly generated instances when the training rate is set to 70%. Except for data sets T45, T63, and T64, RPC provides the highest accuracy on the test sets, and, for all the data sets, its accuracy is higher than 80%. As shown in Tables 1 and 2, the robustness of the proposed RPC guarantees its scalability on the test sets.

4.4. Test on Multiple Classification. In this subsection, we test the performances of on multiple classification problems by comparison with RSVM and NBC. Since the performance of RSVM is determined by its regularization parameter λ , we run a set of RSVM with λ varying from 0 to a big enough number and select the one with the best performance on test sets.

Figures 1 and 3 plot the performances of three methods on Y5 and T1 training sets, respectively. Unlike the case of binary classification problems, we can see that RPC provides a competitive performance even on the training sets. One explanation is that RSVM can outperform the proposed RPC on training sets by finding the optimal separation hyperplane

TABLE 3: Performances of RSVM, NBC, and RPC for binary classification problems on other data sets when $tr = 70\%$.

Data set	RSVM		NBC		RPC	
	Train (%)	Test (%)	Train (%)	Test (%)	Train (%)	Test (%)
T41	62.0	59.7	82.4	78.5	77.9	83.5*
T42	87.0	82.2	84.1	83.1	80.5	85.3*
T43	68.0	61.2	80.2	75.4	85.5	86.9*
T44	91.3	83.9	77.9	86.8	88.8	90.5*
T45	86.5	87.0	93.2	91.0*	84.0	89.1
T61	80.6	79.0	80.5	83.0	83.6	87.8*
T62	71.4	66.5	86.9	85.4*	86.3	85.4*
T63	63.7	69.5	89.6	89.1*	82.2	84.4
T64	88.2	86.7	97.0	96.9*	93.4	95.5
T65	75.0	63.4	79.7	81.5	90.5	92.9*

TABLE 4: Performances of RSVM, NBC, and RPC for multiple classification problems on T1 data set.

Data set	RSVM		NBC		RPC	
	Train (%)	Test (%)	Train (%)	Test (%)	Train (%)	Test (%)
M1	65.4	68.2	72.7	73.7	79.1	77.4*
M2	76.9	75.3	82.6	74.8	81.7	80.9*
M3	57.9	69.9	74.8	87.4	95.4	92.0*
M4	70.4	64.1	97.1	92.3	95.4	92.3*
M5	77.4	71.3	89.4	88.1*	92.0	88.0
M6	75.7	70.5	74.1	79.4	86.4	80.8*

FIGURE 1: Performances of RSVM, NBC, and RPC on Y5 training set.

for binary classification problem S while RPC is more robust to extend to solve multiple classification problems since it uses the nonlinear probability information of the data sets. The accuracy of NBC on the training sets also improves as the training rate increases.

Figures 2 and 4 show the performances of both methods on Y5 and T1 test sets, respectively. We can see that, for most

FIGURE 2: Performances of RSVM, NBC, and RPC on Y5 test set.

of the cases, RPC provides the highest accuracy among three methods. The accuracy of RSVM outperforms that of NBC on Y5 test set while the latter outperforms the former on the T1 test set.

To further test the performance of PRC on multiple classification problems, we carry out more experiments on data sets M1–M6. Table 4 reports the averaged performances of three methods on these data sets when the training rate is set to 70%. Except for the M5 data set, PRC always

FIGURE 3: Performances of RSVM, NBC, and RPC on T1 training set.

FIGURE 4: Performances of RSVM, NBC, and RPC on T1 test set.

provides the highest classification performances among three methods, and even for the M5 data set, its accuracy (88.0%) is very close to the best one (88.1%).

From the tested real-life application, we conclude that the proposed RPC has the robustness to provide better performance for both binary and multiple classification problems compared with RSVM and NBC. The robustness of PRC enables it to avoid the “overlearning” phenomenon, especially for the binary classification problems.

5. Conclusion

In this paper, we propose a robust probability classifier model to address the data uncertainty in classification problems.

To quantitatively describe the data uncertainty, a class-conditional distributional set is constructed based on the modified χ^2 -distance. We assume that the true distribution lies in the constructed distributional set centered in the nominal probability distribution. Based on the “linear combination assumption” for the posterior class-conditional probabilities, we consider a classification criterion using the weighted sum of the posterior probabilities. The optimal robust probability classifier is determined by minimizing the worst-case absolute error value over all the possible distributions belonging to the distributional set.

Our proposed model introduces the recently developed distributionally robust optimization method into the classifier design problems. To obtain a computable model, we transform the resulted optimization problem into an equivalent second order cone programming based on conic duality theorem. Thus our model has the same computational complexity as the classic support vector machine and numerical experiments on real-life application validate its effectiveness. On the one hand, the proposed robust probability classifier provides a higher accuracy compared with RSVM and NBC by avoiding overlearning on training sets for binary classification problems; on the other hand, it also has a promising performance for multiple classification problems.

There are still many important extensions in our model. Other forms of loss function, such as the mean squared error function and Hinge loss functions, should be studied to obtain tractable reformulations and the resulted models may provide better performances. Probability models considering joint probability distribution information are also interesting research directions.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*, John Wiley & Sons, New York, NY, USA, 1973.
- [2] P. Langley, W. Iba, and K. Thompson, “An analysis of Bayesian classifiers,” in *Proceedings of the 10th National Conference on Artificial Intelligence (AAAI '92)*, vol. 90, pp. 223–228, AAAI Press, Menlo Park, Calif, USA, July 1992.
- [3] B. D. Ripley, *Pattern Recognition and Neural Networks*, Cambridge University Press, Cambridge, UK, 2007.
- [4] V. Vapnik, *The Nature of Statistical Learning Theory*, Springer, Berlin, Germany, 2000.
- [5] M. Ramoni and P. Sebastiani, “Robust Bayes classifiers,” *Artificial Intelligence*, vol. 125, no. 1-2, pp. 209–226, 2001.
- [6] Y. Shi, Y. Tian, G. Kou, and Y. Peng, “Robust support vector machines,” in *Optimization Based Data Mining: Theory and Applications*, Springer, London, UK, 2011.
- [7] Y. Z. Wang, Y. L. Zhang, F. L. Zhang, and J. N. Yi, “Robust quadratic regression and its application to energy-growth consumption problem,” *Mathematical Problems in Engineering*, vol. 2013, Article ID 210510, 10 pages, 2013.

- [8] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski, *Robust Optimization*, Princeton University Press, Princeton, NJ, USA, 2009.
- [9] A. Ben-Tal and A. Nemirovski, "Robust optimization—methodology and applications," *Mathematical Programming*, vol. 92, no. 3, pp. 453–480, 2002.
- [10] D. Bertsimas, D. B. Brown, and C. Caramanis, "Theory and applications of robust optimization," *SIAM Review*, vol. 53, no. 3, pp. 464–501, 2011.
- [11] G. R. G. Lanckriet, L. E. Ghaoui, C. Bhattacharyya, and M. I. Jordan, "Minimax probability machine," in *Advances in Neural Information Processing Systems*, pp. 801–807, 2001.
- [12] G. R. G. Lanckriet, L. El Ghaoui, C. Bhattacharyya, and M. I. Jordan, "A robust minimax approach to classification," *Journal of Machine Learning Research*, vol. 3, no. 3, pp. 555–582, 2003.
- [13] L. El Ghaoui, G. R. G. Lanckriet, and G. Natsoulis, "Robust classification with interval data," Tech. Rep. UCB/CSD-03-1279, Computer Science Division, University of California, 2003.
- [14] K. Huang, H. Yang, I. King, and M. R. Lyu, "Learning classifiers from imbalanced data based on biased minimax probability machine," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '04)*, vol. 2, pp. 558–563, IEEE, July 2004.
- [15] K. Huang, H. Yang, I. King, M. R. Lyu, and L. Chan, "The minimum error minimax probability machine," *The Journal of Machine Learning Research*, vol. 5, pp. 1253–1286, 2004.
- [16] C.-H. Hoi and M. R. Lyu, "Robust face recognition using minimax probability machine," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME '04)*, vol. 2, pp. 1175–1178, June 2004.
- [17] T. Kitahara, S. Mizuno, and K. Nakata, "Quadratic and convex minimax classification problems," *Journal of the Operations Research Society of Japan*, vol. 51, no. 2, pp. 191–201, 2008.
- [18] T. Kitahara, S. Mizuno, and K. Nakata, "An extension of a minimax approach to multiple classification," *Journal of the Operations Research Society of Japan*, vol. 50, no. 2, pp. 123–136, 2007.
- [19] D. Klabjan, D. Simchi-Levi, and M. Song, "Robust stochastic lot-sizing by means of histograms," *Production and Operations Management*, vol. 22, no. 3, pp. 691–710, 2013.
- [20] L. V. Utkin, "A framework for imprecise robust one-class classification models," *International Journal of Machine Learning and Cybernetics*, 2012.
- [21] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*, Cambridge University Press, Cambridge, UK, 2000.
- [22] B. Scholkopf and A. J. Smola, *Learning with Kernels*, The MIT Press, Cambridge, UK, 2002.
- [23] T. Hastie, R. Tibshirani, and J. J. H. Friedman, *The Elements of Statistical Learning*, Springer, New York, NY, USA, 2001.
- [24] L. A. Zadeh, "A simple view of the Dempster-Shafer theory of evidence and its implication for the rule of combination," *AI Magazine*, vol. 7, no. 2, pp. 85–90, 1986.
- [25] R. Yager, M. Fedrizzi, and J. Kacprzyk, *Advances in the Dempster-Shafer Theory of Evidence*, John Wiley & Sons, New York, NY, USA, 1994.
- [26] J. F. Sturm, "Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones," *Optimization Methods and Software*, vol. 11, no. 1, pp. 625–653, 1999.
- [27] K. C. Toh, R. H. T. Tütünü, and M. J. Todd, "On the implementation and usage of SDPT3C—a Matlab software package for semidefinite quadratic linear programming, version 4.0," 2006, <http://www.math.nus.edu.sg/~mattohkc/sdpt3/guide4-0-draft.pdf>.
- [28] A. Ben-Tal, D. D. Hertog, A. D. Waegenaere, B. Melenberg, and G. Rennen, "Robust solutions of optimization problems affected by uncertain probabilities," *Management Science*, vol. 59, no. 2, pp. 341–357, 2013.