A Maximum K-Min Approach for Classification

Mingzhi Dong†, Liang Yin†, Weihong Deng†, Li Shang‡, Jun Guo†, Honggang Zhang†
†Beijing University of Posts and Telecommunications
‡Intel Labs China
mingzhidong@gmail.com, {yin,whdeng}@bupt.edu.cn, li.shang@intel.com, {guojun,zhhg}@bupt.edu.cn

Abstract
In this paper, a general Maximum K-Min approach for classification is proposed. With the physical meaning of optimizing the classification confidence of the K worst instances, Maximum K-Min Gain/Minimum K-Max Loss (MKM) criterion is introduced. To make the original optimization problem with combinational number of constraints computationally tractable, the optimization techniques are adopted and a general compact representation lemma for MKM Criterion is summarized. Based on the lemma, a Nonlinear Maximum K-Min (NMKM) classifier and a Semi-supervised Maximum K-Min (SMKM) classifier are presented for traditional classification task and semi-supervised classification task respectively. Based on the experiment results of publicly available datasets, our Maximum K-Min methods have achieved competitive performance when comparing against Hinge Loss classifiers.

Introduction
In the realm of classification, maximin approach, which pays strong attention to the worst situation, is widely adopted and it is regarded as one of the most elegant ideas. Hard-margin Support Vector Machine (SVM) (Vapnik 2000; Cristianini and Shawe-Taylor 2000) is the most renowned maximin classifier and it enjoys the intuition of margin maximization. Nevertheless, maximin methods based on the worst instance may be sensitive to noisy points/outliers near the boundary, as shown in Figure 1(a). Therefore, slack variables are introduced and Soft-margin SVM is proposed (Vapnik 2000; Cristianini and Shawe-Taylor 2000). By tuning the hyperparameter/hyperparameters, a balance between the margin and the Hinge Loss can be obtained. Satisfied classification performance has been reported in a large number of applications and numerous modified algorithms have been proposed for specified tasks, such as S3VM for semi-supervised classification (Bennett, Demiriz, and others 1999), MI-SVM for multi-instance classification (Andrews, Tschantaridis, and Hofmann 2002) and so on. But during the training process, the methods based on Hinge Loss can not control the number of worst instances to be considered exactly, in many applications, we may prefer to set a parameter K and focus on maximizing the gain obtained by the K worst-classified instances while ignoring the remaining ones, as exemplified in Figure 1(c)(d).

Therefore, after the previous work on a special case of naive linear classifier (Dong et al. 2012), in this paper, we propose a general Maximum K-Min approach for classification. With the physical meaning of optimizing the classification confidence of the K worst instances, Maximum K-Min Gain/Minimum K-Max Loss (MKM) criterion is firstly introduced. Then, the general compact representation lemma is summarized, which makes the original optimization problem with combinational number of constraints computationally tractable. To verify the performance of Maximum K-Min criterion, a Nonlinear Maximum K-Min (NMKM) classifier and a Semi-supervised Maximum K-Min (SMKM) classifier are presented for traditional classification task and semi-supervised classification task respectively. Based on the experiment results of publicly available datasets, our Maximum K-Min methods have achieved competitive performance when comparing against Hinge Loss classifiers.

In summary, the contributions of this paper are listed as follows
- MKM criterion is introduced, which can control the parameter of K directly and servers as an alternative of Hinge Loss;
- The general compact representation lemma of MKM criterion is summarized;
- NMKM and SMKM are presented for traditional classification and semi-supervised classification respectively;
- The performance of MKM criterion is verified via experiments.

This paper is organized as follows. Section 2 discusses SVM in the view of maximin. Then Section 3 describes the MKM criterion and the tractable representation. Section 4 proposes two MKM classifiers of NMKM and SMKM. Section 5 shows the experiment results and Section 6 draws the conclusion.

SVM in the View of Maximin
Firstly, we will discuss the maximin formula of SVM in this section.
Figure 1: A comparison of Maximin Approach (Hard-Margin SVM) and linear Maximum K-Min Approach (Support Vectors/Worst K Instances are marked ×). When selected as the support vectors, the outliers near the decision boundary have much influence in Maximin approach, as shown in (a). In contrast, with proper chosen K, Maximum K-Min approach will be more robust, as shown in (c)(d).

Maximin Formula of Hard-Margin SVM

In traditional binary classification, the task is to estimate the parameters $w$ of a classification function $f$ according to a set of training instances whose categories are known. Under linear assumption, the classification function can be expressed as $f(x, w) = w^T x + b$. Then, the category $t$ of a new instance $x$ will be determined as follows

$$
\begin{align*}
    f(x, w) \geq 0 & \quad \rightarrow t = 1; \\
    f(x, w) < 0 & \quad \rightarrow t = -1.
\end{align*}
$$

(1)

In linear separable case, there will exist $w$ such that

$$
g_n = t_n f_n = t_n f(x_n, w) \geq 0, \quad n = 1, \ldots, N \quad (2)
$$

where $f_n = f(x_n, w)$, $x_n$ indicates the $n$th training instances, $t_n$ indicates the corresponding label, $N$ indicates the number of training instances.

Hard-margin SVM defines margin as the smallest distance between the decision boundary and any of the training instances. A set of parameters which maximizes the margin will be obtained during the training process. The original objective function of linear Hard margin SVM can be expressed as

$$
\max \left\{ \min_{w, b} \frac{t_n f(x_n, w)}{\|w\|} \right\}, \quad n = 1, \ldots, N.
$$

(3)

Therefore, it is obvious that Hard-margin SVM can be interpreted as a special case of maximin (Bishop 2006), which tries to find a hyperplane best classifying the worst instance/instances.

To solve Formula 3, by assigning a lower bound to the numerator and minimizing the denominator, the following Quadratic Programming (QP) formula of SVM (Vapnik 2000; Cristianini and Shawe-Taylor 2000) can be obtained

$$
\begin{align*}
    \min_{w, b} & \quad w^T w; \\
    \text{s.t.} & \quad t_n(w^T x_n + b) \geq 1; \\
    & \quad n = 1, \ldots, N.
\end{align*}
$$

(4)

Different from the above formula, we can also assign an upper bound to the denominator and maximize the numerator. Then the original objective function can be reformulated into the following Quadratically Constrained Linear Programming (QCLP) formula

$$
\begin{align*}
    \max_{w, b} & \quad \min_n \{t_n(w^T x_n + b)\}; \\
    \text{s.t.} & \quad w^T w \leq 1; \\
    & \quad n = 1, \ldots, N.
\end{align*}
$$

(5)

In the following derivation of our methods, a similar reformulation strategy will be adopted.

Soft-Margin SVM with Hinge Loss

It is well known that hard-margin SVM cannot deal with non-linearly separable problems efficiently. In Soft-margin SVM (Vapnik 2000; Cristianini and Shawe-Taylor 2000), by
The parameters $w$ of a classifier is obtained via the minimization of K-Max Loss $\Theta_K$ (Dong et al. 2012)

$$\min_w \Theta_K = \sum_{n=1}^{K} \theta_{[n]}.$$ (9)

Therefore, according to the above definition, the MKM can be expressed as the following optimization problem

$$\min_w \ s,$$ (10)

$$s.t. \quad l_{n_1} + \cdots + l_{n_K} \leq s; \quad \text{where} \quad 1 \leq n_1 \leq \cdots \leq n_K \leq N.$$

### Tractable Representation

However, it’s prohibitive to solve an optimization problem with $C_N^K$ inequality constrains. We need to summarize a compact formula with the optimization techniques (Boyd and Vandenberghe 2004). Firstly, we introduce the following lemma.

**Lemma 1** For a fixed integer $K$, the sum of $K$ largest elements of a vector $l$, is equivalent to the optimal value of a linear programming problem as follows,

$$\max_z l^T z;$$ (11)

$$s.t. \quad 0 \leq z_n \leq 1; \quad \sum_n z_n = K;$$

$$\text{where} \quad n = 1, \ldots, N;$$

$$z = (z_1, \ldots, z_N)^T.$$ (11)

**Proof** According to the definition, $\theta_{[1]} \geq \cdots \geq \theta_{[n]} \geq \cdots \geq \theta_{[N]}$ denote the elements of $l$ sorted in decreasing order.

With a fixed integer $K$, the optimal value of $\max l^T z$ under the constraints is obviously $\theta_{[1]} + \cdots + \theta_{[K]}$, which is the same as the sum of $K$ largest elements of $y$, i.e. $\Theta_K(y) = \theta_{[1]} + \cdots + \theta_{[K]}$.

According to Lemma 1, we can obtain an equivalent representation of $\Theta_K$ with $2N + 1$ constrains. Nevertheless, $l$ is multiplied by $z$, this can be solved by deriving the dual of Formula 11

$$\min_{s,u} Ks + \sum_n u_n;$$ (12)

$$s.t. \quad s + u_n \geq l_n; \quad u_n \geq 0;$$

$$\text{where} \quad n = 1, \ldots, N; \quad u = (u_1, \ldots, u_N)^T.$$ (12)

According to the strong duality theory, the optimal value of Formula 12 is equivalent to the optimal value of Formula 11. To make the above conclusion more general, we introduce the following Equivalent Representation Lemma.

**Lemma 2** $\Theta_K$ can be equivalently represented as follows

(1)In the objective Function:

$$\min \Theta_K.$$ (13)
During the training process of NMKM, the similarity be-
taking weighted linear combinations of the training set target
tance during classification. The prediction of
ition, the local evidence is weighted more strongly than dis-
ences in the task of classification. Thus, under this assump-

\[ \min_{s, u} \quad Ks + \sum_{n} u_n; \]
\[ \text{s.t.} \quad s + u_n \geq l_n; \]
\[ u_n \geq 0; \]
\[ n = 1, \ldots, N; \]
\[ u = (u_1, \ldots, u_N)^T. \]
\[ (14) \]

(2) In the constraints:

\[ l \text{ satisfies } \Theta_K \leq \alpha \]
\[ (15) \]

is equivalent to

There exists \( s, u \) which satisfy

\[ Ks + \sum_{n} u_n \leq \alpha; \]
\[ s + u_n \geq l_n; \]
\[ u_n \geq 0; \]
\[ n = 1, \ldots, N; \]
\[ u = (u_1, \ldots, u_N)^T. \]
\[ (16) \]

Maximum K-Min Classifiers

In this section, we will adopt Lemma 2 to design Maximum K-Min Classifiers.

MKM Classifier for Traditional Classification

First, we will design a Nonlinear Maximum K-Min Classi-
fier (NMKM) for traditional binary classification task.

Linear Representation Assumption  To utilize kernel in
NMKM approach, we make the following linear representa-
tion assumption directly

\[ f(x, a, b) = \sum_{i=1}^{N} a_i s(x, x_i) t_i + b \]
\[ (17) \]

where \( s(x, x_i) \) denotes certain kind of similarity measure-
ment between \( x \) and \( x_i \). Kernel function is certainly a good
choice. \( a = \{a_1, \ldots, a_N\} \) denotes the weight of training in-
stances in the task of classification. Thus, under this assump-
tion, the local evidence is weighted more strongly than dis-
tant evidence and different points also show different impor-
tance during classification. The prediction of \( x \) is made by
taking weighted linear combinations of the training set target
values, where the weight is the product of \( a_i \) and \( s(x, x_i) \).
During the training process of NMKM, the similarity be-
tween \( x_n \) and itself will not be considered. Therefore, the
term of \( s(x_n, x_n) \) is deleted and we have the following \( f_n \)

\[ f_n = \sum_{i=1, n, n+1, \ldots, N} a_i s(x_n, x_i) t_i + b. \]
\[ (18) \]

Different from Equation 8, there’s no constraint of \( a_i \) in
our model assumption. Thus during the learning process, we
may obtain \( a_i \) with negative value, which indicates that \( x_i \)
may be mistakenly labeled. For a dataset carefully labeled
without mistakes, the constrains of \( a_i \geq 0 \) can be added as
prior knowledge manually.

KNN (Cover and Hart 1967) in binary classification can be
regarded as a special case of the above assumption with
\( a_i = 1, i = 1, \ldots, N \) and \( s(x, x_i) \) of the following formula

\[ s(x, x_i) = \begin{cases} 1, & \text{if } x_i \text{ is the k-nearest neighbor of } x. \\ 0, & \text{else}. \end{cases} \]

A test point \( x \) can be classified by KNN according to the
sign of \( f(x) \).

In Bayesian Linear Regression (Bishop 2006), the mean
of the predicated variable can be obtained according to the
following formula

\[ \mathbf{m}(x) = \sum_{i=1}^{N} k(x, x_i) t_i. \]

This form can also be considered as a special case of
assumption Equation 17, where \( s(x, x') = k(x, x') = \beta \phi(x)T S_N \phi(x') \) is known as the equivalent kernel.

Nonlinear Maximum K-Min Classifier  Under the as-
sumption of Formula 17, the original optimization Formula
of NMKM is proposed as follows

\[ \min_{a, b, \beta} \quad \sum_{K\text{-largest}} \beta; \]
\[ \text{s.t.} \quad \left\{ \sum_{i=1, n, n+1, \ldots, N} a_i s(x_n, x_i) t_i + b \right\} t_n \leq \beta_n; \]
\[ a^T a \leq 1; \]
\[ n = 1, \ldots, N; \]
\[ \mathbf{a} = (a_1, \ldots, a_N)^T; \]
\[ \mathbf{\beta} = (\beta_1, \ldots, \beta_N)^T. \]
\[ (19) \]

In the above optimization problem, \( \sum_{K\text{-largest}} \beta \) indicates the
K-largest elements of vector \( \mathbf{\beta} \). The regularization of param-
eters \( \mathbf{a} \) is performed via a similar way as Formula 5. We can
also add a \( l_1 \) or \( l_2 \) regularization term in the objective func-
tion, but in this way we have to deal with one more hyper-
parameter.

Tractable Formula of NMKM  According to Lemma 2,
we can obtain a tractable representation for NMKM as fol-
lows

\[ \min_{a, b, u, s} \quad Ks + \sum_{n=1}^{N} u_n; \]
\[ \text{s.t.} \quad \left\{ \sum_{i=1, n, n+1, \ldots, N} a_i s(x_n, x_i) t_i + b \right\} t_n \leq s + u_n; \]
\[ a^T a \leq 1; \]
\[ u_i \geq 0 \]
\[ n = 1, \ldots, N; \]
\[ \mathbf{a} = (a_1, \ldots, a_N)^T; \]
\[ \mathbf{u} = (u_1, \ldots, u_N)^T. \]
\[ (20) \]

Thus the original optimization problem with \( C_{nk}^K + 1 \) con-
straints is reformulated into a convex problem with \( 2N + 1 \)
constraints. As a Quadratically Constrained Linear Pro-
gramming (QCLP) problem, standard convex optimization
methods, such as interior-point methods (Boyd and Vand-
bergh 2004) (Wright 1997), can also be adopted to solve the
above problem efficiently and a global maximum solution is
guaranteed.
MKM Classifier for Semi-supervised Classification

Semi-supervised Classification and S3VM In practical classification problems, since the labeling process is always expensive and laborious, semi-supervised Learning is proposed and tries to improve the classification performance with unlabeled training sets (Chapelle et al. 2006). Therefore, semi-supervised classifiers make use of both labeled and unlabeled data for training and fall between unsupervised classifier (without any labeled training data) and supervised classifier (with completely labeled training data).

As illustrated in (Bennett, Demiriz, and others 1999), Semi-Supervised SVM(S3VM) is a natural extension of SVM in semi-supervised problems and the objective function of S3VM can be written as

\[
\min w, b, \eta, \xi, z, d \quad \sum_{i=1}^{L} \eta_{i} + C \sum_{j=L+1}^{L+U} (\xi_{i} + z_{j}) + w^{T} w;
\]

s.t. \[
y_{i}(w^{T} x_{i} + b) + \eta_{i} \geq 1;
\]
\[
w^{T} x_{j} + b + \xi_{j} + M(1-d_{j}) \geq 1;
\]
\[
-w^{T} x_{j} + b + z_{j} + M d_{j} \geq 1;
\]
\[
\eta_{i} \geq 0; \quad \xi_{j} \geq 0; \quad z_{j} \geq 0;
\]

where \( M > 0 \) is a sufficiently large constant;

\[
i = 1, \ldots, L; \quad j = L + 1, \ldots, L + U;
\]
\[
\eta = (\eta_{1}, \ldots, \eta_{L})^{T};
\]
\[
\xi = (\xi_{L+1}, \ldots, \xi_{L+U})^{T};
\]
\[
z = (z_{L+1}, \ldots, z_{L+U})^{T};
\]
\[
d = (d_{L+1}, \ldots, d_{L+U})^{T};
\]
\[
d_{j} \in \{0, 1\}.
\]

In the above formula, \( L \) indicates the size of labeled training set; \( U \) indicates the size of unlabeled training set; \( d_{j} \) is a binary variable which indicates the predicted category of the unlabeled training instances; \( \sum_{i=1}^{L} \eta_{i} \) measures the Hinge Loss of the labeled training instances and \( \sum_{j=L+1}^{L+U} (\xi_{i} + z_{j}) \) measures the Hinge Loss of the unlabeled training instances. Since the optimization problem has both binary variables and continuous variables, it is a Mixed-Integer Quadratic Programming (MIQP) problem and the globally optimal solution can be solved via commercial solvers, such as Gurobi (Optimization 2012) and Cplex (Cplex 2009).

Semi-supervised Maximum K-Min Classifier We will adopt MKM Criterion to measure the loss of unlabeled training set and we can obtain the following formula

\[
\min w, b, \eta, \xi, z, d, \alpha, \beta \quad \sum_{i=1}^{L} \eta_{i} + C \sum_{K\text{-largest}} \{\alpha; \beta\};
\]

s.t. \[
y_{i}(w^{T} x_{i} + b) + \eta_{i} \geq 1;
\]
\[
w^{T} x_{j} + b + M(1-d_{j}) \leq \alpha_{j};
\]
\[
-w^{T} x_{j} + b + M d_{j} \leq \beta_{j};
\]
\[
w^{T} w \leq 1; \quad \eta_{i} \geq 0;
\]

where \( M > 0 \) is a sufficiently large constant;

\[
i = 1, \ldots, L; \quad j = L + 1, \ldots, L + U;
\]
\[
\eta = (\eta_{1}, \ldots, \eta_{L})^{T};
\]
\[
\alpha = (\alpha_{L+1}, \ldots, \alpha_{L+U})^{T};
\]
\[
\beta = (\beta_{L+1}, \ldots, \beta_{L+U})^{T};
\]
\[
d = (d_{L+1}, \ldots, d_{L+U})^{T};
\]
\[
d_{j} \in \{0, 1\}.
\]

In the above formula, \( \sum_{K\text{-largest}} \{\alpha; \beta\} \) indicates the K-largest elements of a set \( \gamma \) which contains all elements of \( \alpha \) and \( \beta \), i.e. \( \gamma = (\alpha_{L+1}, \ldots, \alpha_{L+U}, \beta_{L+1}, \ldots, \beta_{L+U})^{T} \).

Tractable Formula of SMKM According to Lemma 2, we can obtain a tractable representation for SMKM as follows

\[
\min w, b, \eta, d, s, u, \xi \quad \sum_{i=1}^{L} \eta_{i} + C(Ks + \sum_{n=1}^{2U} u_{n});
\]

s.t. \[
y_{i}(w^{T} x_{i} + b) + \eta_{i} \geq 1;
\]
\[
w^{T} x_{j} + b + M(1-d_{j}) - s - u_{t} \leq 0;
\]
\[
-w^{T} x_{j} + b + M d_{j} - s - u_{t+U} \leq 0;
\]
\[
w^{T} w \leq 1; \quad \eta \geq 0; \quad u_{t} \geq 0;
\]

where \( M > 0 \) is a sufficiently large constant;

\[
i = 1, \ldots, L; \quad t = 1, \ldots, U;
\]
\[
j = L + 1, \ldots, L + U;
\]
\[
\eta = (\eta_{1}, \ldots, \eta_{L})^{T};
\]
\[
u = (u_{1}, \ldots, u_{2U})^{T};
\]
\[
d = (d_{L+1}, \ldots, d_{L+U})^{T};
\]
\[
d_{j} \in \{0, 1\}.
\]

The above formula is a Mixed-Integer Quadratically Constrained Programming (MIQCP) problem and the globally optimal solution can be obtained via optimization solvers, such as Gurobi (Optimization 2012) and Cplex (Cplex 2009).

Experiment

Traditional Classification Experiment

In the experiment of traditional classification, NMKM with Radical Basis Function (RBF) kernel matrix is compared to SVM with RBF kernel and L2-regularized Logistic Regression (LR). NMKM is implemented using cvx toolbox1 (CVX Research ; Grant and Boyd 2008) in matlab environment with the solver of SeDuMi (Sturm 1999). LR is implemented using libliner toolbox2 and kernel SVM is implemented using libsvm toolbox3 (Chang and Lin 2011).

Ten publicly available binary classification datasets are adopted in the experiment. The detailed features are shown in Table 1. The parameters of NMKM and SVM are chosen implemented using libliner toolbox2 and kernel SVM is implemented using libsvm toolbox3 (Chang and Lin 2011).
Table 1: Detailed Features of the Binary Classification Datasets in NMKM Experiment

<table>
<thead>
<tr>
<th>datasets</th>
<th>sources</th>
<th>features</th>
<th>instances(training)</th>
<th>instances(testing)</th>
<th>instances(total)</th>
</tr>
</thead>
<tbody>
<tr>
<td>australian</td>
<td>Statlog / Australian</td>
<td>14</td>
<td>414</td>
<td>276</td>
<td>690</td>
</tr>
<tr>
<td>breast-cancer</td>
<td>UCI / Wisconsin Breast Cancer</td>
<td>10</td>
<td>410</td>
<td>273</td>
<td>683</td>
</tr>
<tr>
<td>colon-cancer</td>
<td>paper (Alon et al. 1999)</td>
<td>62</td>
<td>37</td>
<td>25</td>
<td>62</td>
</tr>
<tr>
<td>diabetes</td>
<td>UCI / Pima Indians Diabetes</td>
<td>8</td>
<td>460</td>
<td>308</td>
<td>768</td>
</tr>
<tr>
<td>fourclass</td>
<td>paper (Ho and Kleinberg 1996)</td>
<td>2</td>
<td>517</td>
<td>345</td>
<td>862</td>
</tr>
<tr>
<td>german.numer</td>
<td>Statlog / German</td>
<td>24</td>
<td>600</td>
<td>400</td>
<td>1000</td>
</tr>
<tr>
<td>heart</td>
<td>Statlog / Heart</td>
<td>13</td>
<td>162</td>
<td>108</td>
<td>270</td>
</tr>
<tr>
<td>ionosphere</td>
<td>UCI / Ionosphere</td>
<td>34</td>
<td>210</td>
<td>141</td>
<td>351</td>
</tr>
<tr>
<td>liver-disorders</td>
<td>UCI / Liver-disorders</td>
<td>6</td>
<td>207</td>
<td>138</td>
<td>345</td>
</tr>
<tr>
<td>splice</td>
<td>Delve / splice</td>
<td>60</td>
<td>1,000</td>
<td>2,175</td>
<td>3,175</td>
</tr>
</tbody>
</table>

Figure 2: Experiment Results of SMKM

'Semi-supervised Classification Experiment

In the experiment of semi-supervised classification, SMKM is compared with S3VM. Both SMKM and S3VM are implemented using cvx toolbox (CVX Research; Grant and Boyd 2008) in matlab environment with the solver of Gurobi (Optimization 2012). Three publicly available UCI datasets of 'Cancer', 'Heart' and 'Pima' are selected for comparison. All datasets are randomly splitted into labeled training set (50 instances), unlabeled training set (50 instances) and the testing set (all other instances). The hyperparameters of NMKM (C, K) and SVM (C1, C2) are chosen via 10 fold cross-validation during the training stage. 11 values for C, C1, C2 ranging from 2−5 to 25 are tested. 10 values for K ranging from 1 to 20 are tested. As shown in Figure 2, SMKM performs better in the datasets of 'Cancer' and 'Heart', while S3VM performs better in the dataset of 'Pima'. Therefore, SMKM has obtained competitive performance when comparing against S3VM in semi-supervised experiment.

**Conclusion**

In this paper, a general Maximum K-Min approach for classification is proposed. By reformulating the original objective function into a compact representation, the optimization of MKM Criterion becomes tractable. To verify the performance of MKM methods, a Nonlinear Maximum K-Min (NMKM) classifier and a Semi-supervised Maximum K-Min (SMKM) classifier are presented for traditional classification task and semi-supervised classification task respectively. As shown in the experiments, the classification performance of Maximum K-Min classifiers is competitive with Hinge Loss classifiers.

**Acknowledge**

We would like to thank the anonymous reviewers for their insight and constructive comments, which helped improve the presentation of this paper. Meanwhile, this work was partially supported by National Natural Science Foundation of China under Grant No.61005025, 61002051, 61273217, 61175011 and 61171193, the 111 project under Grant No.B08004 and the Fundamental Research Funds for the Central Universities.
References


CVX Research, I. CVX: Matlab software for disciplined convex programming, version 2.0 beta.


