数理解析研究所講究録
1174巻 2000年 109-121

A Cutting Plane Algorithm for Semi-Definite Programming Problems with Applications to Failure Discrimination and Cancer Diagnosis

今野浩 (Hiroshi Konno)　後藤順哉 (Jun-ya Gotoh)　宇野毅明 (Takeaki Uno)
Department of Industrial Engineering and Management, Tokyo Institute of Technology

Abstract We will propose a new cutting plane algorithm for solving a class of semi-definite programming problems with a small number of variables and a large number of constraints. This type of problems appear when we try to classify a large number of multi-dimensional data into two groups by a hyper-ellipsoidal surface. Among such examples are cancer diagnosis and failure discrimination of enterprises.
We will show that this algorithm is much more efficient than the standard interior point algorithm for solving semi-definite programming problems.
Keywords : semi-definite programming, ellipsoidal separation, cutting plane method, failure discriminant analysis, cancer diagnosis

1 Introduction

Semi-definite programming problems (SDP) have been under intensive study in recent years. A number of efficient algorithms have been developed [6, 8] and used for solving various classes of combinatorial optimization problems and control problems [14, 15].

Recently, one of the authors applied semi-definite programming approach to failure discrimination of enterprises, where multi-dimensional financial data are classified into two groups, namely failure group and going group by a hyper-ellipsoid. This research was inspired by a remarkable success reported in Mangasarian et al. [12], where three dimensional physical data of suspected breast cancer patients were classified into benign group and malignant group by a hyperplane.

We applied the same idea to failure discrimination of enterprises, a very important and well studied subject in financial engineering. Unfortunately, however this method did not work well for financial data. This led us to extend hyperplane separation to quadratic separation. However, general quadratic separation often generates disconnected regions of discrimination, which is awkward from the nature of financial data. To obtain legitimate results, we need to impose a condition that the discriminant surface is an ellipsoid or paraboloid. Thus we have to solve a semi-definite programming problem.

Separation of multi-dimensional data by an ellipsoid was proposed by Rosen [13] in 1963. However, no one applied this method to real world problems since no efficient computational tools were available until recently. It is reported in [11] that ellipsoid separation performs much better than its counterpart, hyperplane separation. In fact, preliminary computation exhibits that the chance of wrong discrimination of ellipsoid separation is much less than hyperplane separation. In addition, ellipsoid separation may be used as a basis for rating enterprises. Those companies located very far (into a desirable direction) from the separating ellipsoid can be considered as an enterprise with smaller probability of bankruptcy. Thus the distance from the boundary may be used as the score of each enterprise. We compared the rating score announced by a leading rating company and the score calculated by ellipsoidal separation and found that they are reasonably well

correlated [11].

One disadvantage of ellipsoidal separation is that the computation time for solving the resulting SDP is much larger than solving a hyperplane separation problem. When the number k of enterprises is 455 and the dimension n of the data is 6, hyperplane separation problem can be solved in less than one second, while ellipsoid separation requires around 1000 seconds by SDPA, a well designed code for SDP's. Therefore, we need to have a more efficient algorithm when the number of enterprises is over a few thousand.

The purpose of this paper is to propose a more efficient algorithm for solving an SDP with the structure stated above. We will first formulate an SDP as a linear programming problem with an infinite number of linear constraints. We then propose a new algorithm where we solve a series of tighter relaxation problems. The constraint to be added is the mostly violated constraint among the infinite number of linear constraints. Problem to be solved at each step is a linear programming problem whose optimal solution can be recovered from an optimal basic solution of the previous step by applying a number of dual simplex iterations.

In section 2, we will briefly introduce alternative formulations of failure discrimination problem, *i.e.*, a hyperplane separation, quadratic separation and hyper ellipsoidal separation. Section 3 will be devoted to the description of a cutting plane algorithm for SDP.

In Section 4, we will present the result of numerical simulation using the financial data collected in Tokyo capital market. It will be shown that we can solve SDP much faster than SDPA software. Further, we will report the performance of our method to cancer diagnosis problem discussed in Mangasarian et al.[12, 4]. It will be shown that better solution can be obtained by ellipsoidal separation with slightly more computation time.

Section 5 will be devoted to a more detailed analysis of the cutting plane method. Due to the limitation of data availability, we solved problems up to $k = 569$ and $n = 6$. Computational results indicate that problems with k over a few thousand can be solved in a practical amount of time as long as n is less than ten.

2 Separation of Multi-dimensional Data by Mathematical Programming

2.1 Separation by a Hyperplane

Let A_i, $i = 1, \ldots, m$ be going enterprises and B_l, $l = 1, \ldots, h$ be enterprises which have undergone failure. Also, let $a_i \in \mathbb{R}^n$, $b_l \in \mathbb{R}^n$ be, respectively the vectors of financial data of A_i, B_l.

If there exists a vector $(c, c_0) \in \mathbb{R}^{n+1}$ such that

$$c^T a_i > c_0, \quad i = 1, \ldots, m, \tag{1}$$

$$c^T b_l < c_0, \quad l = 1, \ldots, h, \tag{2}$$

we will call

$$H(c, c_0) = \{ x \in \mathbb{R}^n \mid c^T x = c_0 \},$$

a discriminant hyperplane. (Figure 1-(a))

Discriminant hyperplane does not exist in general. (Figure 1-(b))

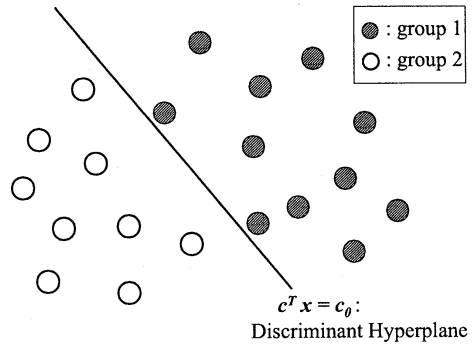

Figure 1-(a) : Discriminant Hyperplane

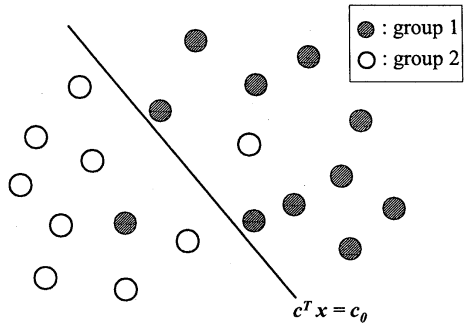

Figure 1-(b) : No Discriminant Hyperplane

Upon normalization, condition (1) and (2) are equivalent to

$$c^T a_i \geq c_0 + 1, \quad i = 1, \ldots, m, \tag{3}$$
$$c^T b_l \leq c_0 - 1, \quad l = 1, \ldots, h. \tag{4}$$

Let us define subspaces :

$$H_+(c, c_0) = \{\ x \in \mathbb{R}^n \mid c^T x \geq c_0\ \}, \tag{5}$$
$$H_-(c, c_0) = \{\ x \in \mathbb{R}^n \mid c^T x \leq c_0\ \}. \tag{6}$$

An enterprise A_i such that $a_i \notin H_+(c, c_0)$ will be called a misclassified enterprise of the first kind. Also, B_l such that $b_l \notin H_-(c, c_0)$ will be called a misclassified enterprise of the second kind.

Let y_i, z_l be, respectively the distance of $a_i \notin H_+(c, c_0)$ and $b_l \notin H_-(c, c_0)$ from the hyperplane $H(c, c_0)$ and let us try to minimize the weighted sum of y_i's and z_l's. This problem can be formulated as the following linear programming problem :

$$\left|\begin{array}{ll} \text{minimize} & (1-\lambda)\dfrac{1}{m}\displaystyle\sum_{i=1}^{m} y_i + \lambda\dfrac{1}{h}\sum_{l=1}^{h} z_l \\ \text{subject to} & c^T a_i + y_i \geq c_0 + 1, \quad i = 1, \ldots, m, \\ & c^T b_l - z_l \leq c_0 - 1, \quad l = 1, \ldots, h, \\ & y_i \geq 0, \quad i = 1, \ldots, m, \\ & z_l \geq 0, \quad l = 1, \ldots, h, \end{array}\right. \tag{7}$$

where $\lambda \in [0, 1]$ is a constant representing the relative importance of the cost associated with misclassification of the first and the second kind.

Let us note that the problem (7) is feasible. Also, the objective function is bounded below. Therefore, it has an optimal solution $(c^*, c_0^*, y_1^* \ldots y_m^*, z_1^* \ldots z_h^*)$ [4]. Mangasarian et al.[12] applied this method to breast cancer diagnosis to classify 569 patients into benign and malignant groups by using 30 dimensional physical data. According to their report, 97.5% of the 569 patients are classified correctly by using only three data carefully chosen from 30 available data.

2.2 Separation by a Quadratic Surface

Discrimination of financial data is more difficult than physical data such as breast cancer data. For one thing, financial data are much less reliable than physical data since they are

calculated from less precise numbers. Even worse, they are sometimes subject to window-dressing procedure. Also, financial data are highly correlated to each other, so that linear (hyperplane) separation is doomed to be of limited power for failure discrimination.

Hyperplane separation method was applied to failure discrimination by Konno-Kobayashi [11], where they chose six dimensional financial data compiled from balance sheets of 170 companies. However, only 92% of the companies were correctly classified, which is not at all satisfactory from the practical point of view.

To improve precision of discrimination, Konno and Kobayashi [11] proposed a quadratic separation. Let $D = (d_{ij}) \in \mathbb{R}^{n \times n}$ be a symmetric matrix and define a quadratic surface :

$$Q(D, c, c_0) = \{ x \in \mathbb{R}^n \mid x^T D x + c^T x = c_0 \}, \tag{8}$$

and consider the following minimization problem :

$$\left| \begin{array}{ll} \text{minimize} & (1-\lambda)\dfrac{1}{m}\displaystyle\sum_{i=1}^{m} y_i + \lambda \dfrac{1}{h}\sum_{l=1}^{h} z_l \\ \text{subject to} & a_i^T D a_i + a_i^T c + y_i \geq c_0 + 1, \quad i = 1, \ldots, m, \\ & b_l^T D b_l + b_l^T c - z_l \leq c_0 - 1, \quad l = 1, \ldots, h, \\ & y_i \geq 0, \quad i = 1, \ldots, m, \\ & z_l \geq 0, \quad l = 1, \ldots, h, \end{array} \right. \tag{9}$$

where $D \in \mathbb{R}^{n \times n}$, $c \in \mathbb{R}^n$, $c_0 \in \mathbb{R}^1$, $y_i \in \mathbb{R}^1$, $i = 1, \ldots, m$; $z_l \in \mathbb{R}^1$, $l = 1, \ldots, h$ are variables to be determined. Variables y_i and z_l represent, respectively the distance of the misclassified enterprises from the quadratic surface $Q(D, c, c_0)$.

The problem (9) is a linear programming problem. It is easy to see that this problem has an optimal solution (D^*, c^*, c_0^*). Since this problem contains $n(n+1)/2$ additional variables compared with (7), the value of objective function should be much better than that of hyperplane separation. On the other hand, the configuration of separating surface $Q(D, c, c_0)$ can be very complicated since we do not impose any condition on the matrix D. In particular, if the surface is hyperbolic, then either one or both of the discriminant regions :

$$Q_+(D^*, c^*, c_0^*) = \{ x \in \mathbb{R}^n \mid x^T D^* x + (c^*)^T x \geq c_0^* \}, \tag{10}$$

$$Q_-(D^*, c^*, c_0^*) = \{ x \in \mathbb{R}^n \mid x^T D^* x + (c^*)^T x \leq c_0^* \}, \tag{11}$$

can be disconnected. This is awkward since financial data are monotonic in the sense that a larger (or smaller) value is associated with better performance of enterprises.

It is reported in [11] that we usually obtain 100% correct separation. However, the discriminant region is disconnected, and the company score exhibits almost no correlation to rating data. This means that quadratic separation leads to overfitting of the model to data, as is usually observed in neural network approach.

2.3 Separation by an Ellipsoid

To avoid the difficulty associated with disconnected discriminant region, Konno and Kobayashi [11] requires $D \in \mathbb{R}^{n \times n}$ to be a symmetric positive semi-definite matrix. This

means that the surface $Q(D, c, c_0)$ becomes a hyper-ellipsoid or hyperparoboloid. This leads us to solve the following minimization problem:

$$\left|\begin{array}{ll} \text{minimize} & (1-\lambda)\dfrac{1}{m}\displaystyle\sum_{i=1}^{m} y_i + \lambda\dfrac{1}{h}\sum_{l=1}^{k} z_l \\ \text{subject to} & a_i^T D a_i + a_i^T c + y_i \geq c_0 + 1, \quad i = 1, \ldots, m, \\ & b_l^T D b_l + b_l^T c - z_l \leq c_0 - 1, \quad l = 1, \ldots, h, \\ & y_i \geq 0, \quad i = 1, \ldots, m, \\ & z_l \geq 0, \quad l = 1, \ldots, h, \\ & x^T D x \geq 0, \quad \forall x \in B_n, \end{array}\right. \tag{12}$$

where $B_n = \{\, x \in \mathbb{R}^n \mid \|x\| = 1\}$.

This is a "semi-definite programming problem". Many practical algorithms [6, 14] have been proposed when n is not too large. By requiring D to be positive semi-definite, we can obtain more persuasive results at the expense of more computation time and misclassification cost compared with general quadratic separation.

Ellipsoidal separation can be used for the rating of enterprises. As reported in [11] scores of individual enterprises calculated by using their distance from the separating ellipsoid exhibits a good correlation with the result of rating reported by a leading rating company. Therefore, these scores may be used as a basis for rating a large number of enterprises in an automatic way.

Unfortunately, however the computation time for solving an SDP (12) is over 1000 times more than that for solving the associated linear program (9) when $n = 6$ and $m+h = 455$. The problem to be solved in practice is much larger, *i.e.*, $m + h$ is over a few thousand while n remains small. Therefore, we need to develop a more efficient algorithm for solving (12) using its special structure.

3 A Cutting Plane Algorithm for SDP

Let us first define $y = (y_1, \ldots, y_m)^T$, $z = (z_1, \ldots, z_h)^T$, $e_m = (1/m, \ldots, 1/m)^T \in \mathbb{R}^m$, $e_h = (1/h, \ldots, 1/h)^T \in \mathbb{R}^h$ and let

$$\mathcal{F}_0 = \left\{ (D, c, c_0, y, z) \;\middle|\; \begin{array}{l} a_i^T D a_i + a_i^T c + y_i \geq c_0 + 1, \quad i = 1, \ldots, m, \\ b_l^T D b_l + b_l^T c - z_l \leq c_0 - 1, \quad l = 1, \ldots, h, \\ y_i \geq 0, \quad i = 1, \ldots, m, \quad z_l \geq 0, \quad l = 1, \ldots, h, \\ d_{jj} \geq 0, \quad j = 1, \ldots, n. \end{array} \right\} \tag{13}$$

and denote (12) in a compact form as follows :

$$\text{(P)} \left|\begin{array}{ll} \text{minimize} & (1-\lambda) e_m^T y + \lambda e_h^T z \\ \text{subject to} & (D, c, c_0, y, z) \in \mathcal{F}_0 \\ & x^T D x \geq 0, \quad \forall x \in B_n. \end{array}\right. \tag{14}$$

An important observation is that this belongs to a class of infinite programming problem, a linear programming problem with an infinite number of constraints.

The first step of our algorithm is to solve a linear programming problem :

$$(\mathrm{Q}_0) \left|\begin{array}{ll} \text{minimize} & (1-\lambda) e_m^T y + \lambda e_h^T z \\ \text{subject to} & (D, c, c_0, y, z) \in \mathcal{F}_0 \end{array}\right. \tag{15}$$

by relaxing the last constraint of (14). Let us note that this program is feasible and the objective function is bounded below. Therefore, it has an optimal solution, $(D^o, c^o, c_0^o, y^o, z^o)$ [5].

If D^o is positive semi-definite, then $(D^o, c^o, c_0^o, y^o, z^o)$ is obviously an optimal solution of (12). If D^o is not positive semi-definite, then there exists $x \in B_n$ such that $x^T D^o x < 0$.

Let us consider the quadratic program :

$$(\pi_0) \left| \begin{array}{ll} \text{minimize} & x^T D^o x \\ \text{subject to} & x \in B_n. \end{array} \right. \tag{16}$$

LEMMA 3.1 *Let λ_0 and x^0 be, respectively the smallest eigenvalue and associated eigenvector of D^0. Then the minimal value of the problem (16) is attained at x^o and $(x^o)^T D^o x^o = \lambda_0$.*

Proof See Gantmacher [7]. ∎

Let us define a new set

$$\mathcal{F}_1 = \mathcal{F}_0 \cap \left\{ D \mid (x^o)^T D x^o \geq 0 \right\} \tag{17}$$

and define a tighter linear program

$$(Q_1) \left| \begin{array}{ll} \text{minimize} & (1-\lambda) e_m^T y + \lambda e_h^T z \\ \text{subject to} & (D, c, c_0, y, z) \in \mathcal{F}_1 \end{array} \right. \tag{18}$$

In the $k(\geq 1)$ th step, let us consider the linear program

$$(Q_k) \left| \begin{array}{ll} \text{minimize} & (1-\lambda) e_m^T y + \lambda e_h^T z \\ \text{subject to} & (D, c, c_0, y, z) \in \mathcal{F}_k \end{array} \right. \tag{19}$$

where

$$\mathcal{F}_k = \mathcal{F}_{k-1} \cap \left\{ D \mid (x^{k-1})^T D x^{k-1} \geq 0 \right\} \tag{20}$$

Let $(D^k, c^k, c_0^k, y^k, z^k)$ be an optimal solution and solve

$$(\pi_k) \left| \begin{array}{ll} \text{minimize} & x^T D^k x \\ \text{subject to} & x \in B_n. \end{array} \right. \tag{21}$$

Let x^k be its solution, for which the objective value of (π_k) is the minimal eigenvalue by lemma 1. If D^k is positive semi-definite, then we are done. Otherwise, repeat the k-th step replacing k with $k+1$.

Cutting Plane (CP) Algorithm

Initialization Let $\varepsilon > 0$ be a tolerance and set $\mathcal{F}_0$ such as (13) and $k = 0$.

General Step k Solve a linear program (Q_k) and let $(D^k, c^k, c_0^k, y^k, z^k)$ be its optimal solution. Let $\alpha_k, \tilde{D}^k$, $\tilde{c}^k$ and $\tilde{c}_0^k$ be, respectively, a constant, a matrix, a vector, a scalar satisfying that $\|\tilde{D}^k\|^2 + \|\tilde{c}^k\|^2 + \|\tilde{c}_0^k\|^2 = 1, \alpha_k > 0, \alpha^k \tilde{D}^k = D^k$, $\alpha_k \tilde{c}^k = c^k$ and $\alpha_k \tilde{c}_0^k = c_0^k$.[1]

[1]Note that $\alpha_k, \tilde{D}^k$, $\tilde{c}^k$, $\tilde{c}_0^k$ are uniquely defined for a set of (D^k, c^k, c_0^k).

Case 1 $(x^k)^T D^k x^k \geq -\varepsilon$. Then $(D^k, c^k, c_0^k, y^k, z^k)$ is an optimal solution of (P)

Case 2 $\tilde{D}^k$ and $\tilde{c}^k$ satisfy

$a_i^T \tilde{D}^k a_i + a_i^T \tilde{c}^k - \tilde{c}_0^k \geq 0, \quad i = 1, \ldots, m,$

$b_i^T \tilde{D}^k b_l + b_l^T \tilde{c}^k - \tilde{c}_0^k \leq 0, \quad l = 1, \ldots, h.$

Then $Q(\tilde{D}^k, \tilde{c}^k, \tilde{c}_0^k)$ is a separating ellipsoidal surface.

Case 3 Otherwise, set $k \leftarrow k+1$ and

$$\mathcal{F}_{k+1} = \mathcal{F}_k \cap \left\{ D \mid (x^k)^T D x^k \geq 0 \right\}$$

and repeat **General Step *k*.**

Note that any (Q_k) has an optimal solution since (Q_k) has a feasible solution and the objective function can not be negative for feasible solutions. Now let v^* be the optimal value of (P).

Theorem 1 *$(D^k, c^k, c_0^k, y^k, z^k)$ converges to an ε-optimal solution of (P), or $Q(\tilde{D}^k, \tilde{c}^k, \tilde{c}_0^k)$ converges to a separating ellipsoidal surface.*

Proof To prove the theorem, we show that "if $(D^k, c^k, c_0^k, y^k, z^k)$ does not converge to any optimal solution of (P), then $Q(\tilde{D}^k, \tilde{c}^k, \tilde{c}_0^k)$ converges to a separating ellipsoidal surface".

Suppose that $(D^k, c^k, c_0^k, y^k, z^k)$ does not converge to any optimal solution of (P). Then, the algorithm generates an infinite sequence $\{(D^1, c^1, c_0^1, y^1, z^1), (D^2, c^2, c_0^2, y^2, z^2), \ldots\}$ satisfying that $(x^k)^T D^k x^k < -\delta$ for a positive constant δ.

The vector composed of the elements of $\tilde{D}^k$, $\tilde{c}^k$ and $\tilde{c}_0^k$ is on the surface of the unit sphere of $\mathbb{R}^{(n+1)(n+2)/2}$. Since the sphere is a compact set, there is an infinite subsequence $\{j_1, j_2, \ldots\} \subseteq \{1, 2, \ldots\}$ such that

$$(\tilde{D}^{j_k}, \tilde{c}^{j_k}, \tilde{c}_0^{j_k}) \rightarrow (\tilde{D}^\infty, \tilde{c}^\infty, \tilde{c}_0^\infty)$$

as $k \rightarrow \infty$. Therefore, for any $\varepsilon > 0$, there exists a constant K such that

$$\left| (x^{j_k})^T \tilde{D}^{j_{k+1}} x^{j_k} - (x^{j_k})^T \tilde{D}^{j_k} x^{j_k} \right| < \varepsilon, \quad {}^\forall k > K.$$

However, since $(x^{j_k})^T \tilde{D}^{j_{k+1}} x^{j_k} \geq 0$, we have

$$(x^{j_k})^T \tilde{D}^{j_k} x^{j_k} > (x^{j_k})^T \tilde{D}^{j_{k+1}} x^{j_k} - \varepsilon \geq -\varepsilon, \quad {}^\forall k > K.$$

Therefore, $\tilde{D}^\infty$ is a semi-definite matrix. Since we assumed that $\alpha_k((x^k)^T \tilde{D}^k x^k) = (x^k)^T D^k x^k < -\delta$, α_{j_k} diverges to ∞ as $k \rightarrow \infty$.

Since the optimal value of (Q_k) is not greater than v^*, there exists a constant $\overline{y}$ such that for any i, $y_i^k \leq \overline{y}$. Then,

$$\alpha_{j_k}(a_i^T \tilde{D}^{j_k} a_i + a_i^T \tilde{c}^{j_k} - \tilde{c}_0^{j_k}) \geq 1 - y_i^k \geq 1 - \overline{y}$$

for any $i, 1 \leq i \leq m$. Since $\alpha_{j_k} \rightarrow \infty$, for any $\varepsilon > 0$, there exists a constant K such that $a_i^T \tilde{D}^{j_k} a_i + a_i^T \tilde{c}^{j_k} - \tilde{c}_0^{j_k} \geq -\varepsilon$ for any $k > K$. Therefore, for any $i, 1 \leq i \leq m$,

$$a_i^T \tilde{D}^\infty a_i + a_i^T \tilde{c}^\infty - \tilde{c}_0^\infty \geq 0.$$

Similarly, for any l, $1 \leq l \leq h$, we have

$$b_l^T \tilde{D}^\infty b_l + b_l^T \tilde{c}^\infty - \tilde{c}_0^\infty \leq 0.$$

Here we obtain that $Q(\tilde{D}^\infty, \tilde{c}^\infty, \tilde{c}_0^\infty)$ is a separating ellipsoidal surface. ∎

4 Quality of Separation

We compared three alternative separation schemes *i.e.*, (i) hyperplane separation, (ii) quadratic separation and (iii) ellipsoidal separation using (a) financial data of enterprises used in [11] and (b) WDBC data used for breast cancer diagnosis [12]. All the experiments were conducted on PentiumIII Processor (500MHz) using C/C++. Also linear programming subproblems were solved by CPLEX6.5.

(a) Failure Discriminant Analysis

Failure discrimination has a long history since the pioneering work of Altman [1, 2] and is one of the most actively studied field in financial optimization [9, 10]. In [11], 455 companies were classified into two groups, *i.e.*, ongoing group and failure group using following six dimensional data

1. I_1=cash flow(million yen)
2. I_2=(debt/cash flow)$\times$ 100
3. I_3=capital adequacy ratio (%)
   =(amount of net capital/amount of total capital)$\times$ 100
4. I_4=ROE(rate of return on equity)
   =(operation profit + received interest/dividend)/(total capital)$\times$ 100
5. I_5=instant coverage
   =(operation profit + received interest/dividend)/(paid interest + discount charge)
6. I_6=operating profit rate
   =(operating income + non-operating income)/(operating expense + non-operating expense)$\times$ 100

These indexes are those which are used for rating enterprises by R&I, Co, the largest rating company of Japan.

Table 1 shows the results of each method.

Table 1 : Misclassification of Failure Discrimination

	(a) Hyperplane	(b) Indefinite	(c) Ellipsoid	(d) Reverse
total	37/455 (8.13%)	0/455 (0.00%)	0/455 (0.00%)	23/455 (5.05%)
1st	0/7 (0.00%)	0/7 (0.00%)	0/7 (0.00%)	0/7 (0.00%)
2nd	37/448 (8.26%)	0/448 (0.00%)	0/448 (0.00%)	23/448 (5.13%)

The column named "(b) Indefinite" shows the result by (9). "(d) Reverse" shows the result of (12) so that the ongoing enterprises would be enclosed within the ellipsoid, while "(c) Ellipsoid" tried to enclose the failed enterprises within the ellipsoid.

Here misclassification of the 1st and 2nd kind refer to the case that an ongoing enterprise is classified into the failure group and vice versa. Also "reverse" means that the group of ongoing enterprises are enclosed within the ellipsoid, contrary to the standard case where failure group is enclosed. We found that there exists a separating ellipsoid when we used the standard method. However, a large number of ongoing companies are misclassified into failure group (misclassification of the 1st kind) when we employed reverse separation. This means that the decision as to which group should be enclosed is very important.

These results contradict the earlier result reported in [11], where 2 out of 7 failed companies are misclassified into the going group in standard ellipsoidal separation. This is due to the fact that computation in [11] was terminated before convergence.

Table 2 shows the computation time of each method. We see that the cutting plane (CP) method is slower than hyperplane or quadratic separation. However, it is much faster than SDPA. (See Section 5.)

Table 2 : Computation Time

				original data		normalized data	
		#var	#const	time[sec.]	#pivot	time[sec.]	#pivot
(a)	Hyperplane	462	455	0.07	25	0.08	19
(b)	Indefinite	483	455	0.30	444	0.27	219
(c)	Ellipsoid	483	455	4.37	1376	29.66	4183
(d)	Reverse	483	455	4.13	953	6.83	1000

(b) Cancer Diagnosis : Ex-Ante Performance

Table 3 shows the results of separation of breast cancer data reported in [12], where three dimensional data were classified into benign and malignant groups. The result is surprisingly good. In fact, it is comparable to the results of professional physician.

We see that the quality of ellipsoidal separation is slightly better than hyperplane separation.

Table 3 : Misclassification of Cancer Diagnosis

	(a) Hyperplane	(b) Indefinite	(c) Ellipsoid	(d) Reverse
total	14/569 (2.46%)	13/569 (2.28%)	12/569 (2.11%)	15/569 (2.64%)
1st.	7/212 (3.30%)	7/212 (3.30%)	7/212 (3.30%)	8/212 (3.77%)
2nd.	7/357 (1.96%)	6/357 (1.68%)	5/357 (1.40%)	7/357 (1.96%)

(c) Cancer Diagnosis : Ex-post Performance

Let us compare the quality of hyperplane separation and ellipsoidal separation in more detail. Following Mangasarian et al.[12], we partitioned all sample data into 10 disjoint groups. We calculate separating hyperplane/(indefinite) quadratic surface/ellipsoids using 9 data groups by removing one group, say p-th group and check the quality of prediction using the p-th data group. We repeated this process ten times for all p ($p = 1, \ldots, 10$).

Table 4 shows the total number of misprediction, average and standard deviation of misclassification (within 9 data groups) and the average and standard deviation of misprediction.

Table 4 : Ex-post Analysis

	total # of misprediction			average misclassification			average misprediction		
	total	1st.	2nd.	total	1st.	2nd.	total	1st.	2nd.
(a) Hyperplane	15/569	8/212	7/357	2.52%	3.41%	1.99%	2.64%	3.81%	1.96%
	(2.64%)	(3.77%)	(1.96%)	(0.22%)	(0.42%)	(0.21%)	(2.28%)	(3.56%)	(2.20%)
(b) Indefinite	17/569	9/212	8/357	2.25%	3.35%	1.59%	2.99%	4.29%	2.23%
	(2.99%)	(4.25%)	(2.24%)	(0.29%)	(0.35%)	(0.38%)	(1.58%)	(3.33%)	(1.67%)
(c) Ellipsoid	16/569	8/212	8/357	2.32%	3.46%	1.65%	2.99%	3.81%	2.23%
	(2.81%)	(3.77%)	(2.24%)	(0.37%)	(0.42%)	(0.44%)	(1.57%)	(2.86%)	(2.08%)
(d) Reverse	15/569	8/212	7/357	2.64%	3.72%	1.99%	2.64%	3.81%	1.96%
	(2.64%)	(3.77%)	(1.96%)	(0.28%)	(0.55%)	(0.21%)	(2.28%)	(3.56%)	(2.20%)

The values in brackets in the left part of the table shows the percentage of misclassification of 10 trials, while those in the center and the right parts show the values of standard deviation of misclassification and misprediction of 10 trials, respectively.

We observe that there is no significant difference among two methods. Let us note that the quality of prediction of standard ellipsoidal separation is a little bit worse than hyper plane separation to our disappointment. The quality of prediction of reverse ellipsoidal separation is better than standard ellipsoidal separation, which implies the need for more detailed analysis.

5 Efficiency of Cutting Plane Algorithm

We compared the performance of the cutting plane algorithm and the standard interior point algorithm for SDP's using the software SDPA5.0 [6] as a benchmark.

We employed two alternative parameter setting method, *i.e.*, (i) stable_but_slow (SS) and (ii) unstable_but_fast (UF)

Table 5 : Computational Performance of SDPA5.0 for Failure Discrimination

parameter & data	CPU-time [sec.]	#itr	solution status	misclassification		
				total	1st	2nd
SS with orig	1142.92	60	pdFEAS	0/455	0/7	0/448
UF with orig	1423.45	75	pdFEAS	0/455	0/7	0/448
SS with norm	866.19	45	pdFEAS	3/455	0/7	3/448
UF with norm	923.04	48	pdFEAS	3/455	0/7	3/448

"orig" in the left column means that original data was used, while "norm" means that normalized data was used.
"pdFEAS" stands for "primal-dual feasible", which indicates optimal solution was not attained.

Table 6 : Computational Performance of SDPA5.0 for Cancer Diagnosis

parameter & data	CPU-time [sec.]	#itr	solution status	misclassification		
				total	1st	2nd
SS with orig	767.84	34	pFEAS	12/569	7/212	5/357
UF with orig	1030.81	46	pdFEAS	12/569	7/212	5/357
SS with norm	899.95	40	pdFEAS	12/569	7/212	5/357
UF with orig	832.92	37	pdFEAS	12/569	7/212	5/357

"pFEAS" stands for "primal feasible", which indicates optimal solution was not attained.

We see from Table 5 that the normalized data leads to faster convergence. Also, SS shows better performance than UF, contrary to our expectation. We observe that SDPA terminates at pdFEAS instead of pdOPT, which means that this optimality is not yet achieved although the solution is expected to be close to optimal. In any case, cutting plane algorithm is much more efficient than SDPA. The primary reason for this difference is that SDPA has to handle a semi-definite matrix of the size $(m+h+1) \times 2 + n$, while our method works on a semi-definite matrix of size n. Note that n is very small (less than 6) and $n+m+h$ is one hundred times larger.

In addition, an optimal solution of the subproblems of CP algorithm can be obtained very fast by a few dual simplex iterations due to the special structure of the problem. However, the efficiency of CP algorithm is expected to deteriorate as n increases as is usually observed for various classes of cutting plane or outer approximation algorithm. Also, as discussed later, convergence is slower when we can completely separate data in two sets. In such case, however we usually obtain perfect separation by a hyperplane, so that ellipsoidal separation is not required.

Next, we will discuss the performance of cutting plane algorithm for more general class of problems. The results of separation should be the same regardless of the scaling and/or the choice of origin. However, the speed of convergence is affected by this transformation. Table 7 shows the results for the failure discrimination for original data and normalized data. We see that the normalization results an larger computation time.

Table 7 : Computational Performance of CP algorithm for Failure Discrimination

tolerance	CPU-time[sec.]		#itr		#pivot		max_eeigenvalue	
(ε)	orig	norm	orig	norm	orig	norm	orig	norm
(1st itr.)	0.26	0.19	1	1	485	93	0.08501	271.645
-1.00E-03	2.58	24.91	55	472	1277	4053	0.41508	22373.7
-1.00E-04	3.08	25.96	70	490	1321	4086	0.45886	22851.8
-1.00E-05	3.50	27.05	83	509	1340	4110	0.46148	22827.9
-1.00E-06	4.00	28.12	98	527	1362	4137	0.46453	22881.1
-1.00E-07	4.34	29.63	108	551	1376	4183	0.46324	22850.7

Table 8 shows the result of cancer diagnosis, where ellipsoidal separation cannot completely separate benign and malignant groups.

We see from this that convergence is very fast although the size of the problem is a bit larger. Also, CPU time (as well as the number of iteration) is not sensitive to the level of tolerance.

Table 8 : Computational Performance of CP algorithm for Cancer Diagnosis

tolerance	CPU-time[sec.]		#itr		#pivot		max_eeigenvalue	
(ε)	orig	norm	orig	norm	orig	norm	orig	norm
(1st.itr.)	0.06	0.07	1	1	26	28	0.4354	1.03036
-1.00E-03	0.17	0.18	8	7	45	61	2.5671	1.28427
-1.00E-04	0.25	0.20	13	8	60	62	15.7846	1.30669
-1.00E-05	0.33	0.23	18	10	87	64	61.3253	1.30031
-1.00E-06	0.40	0.26	22	12	107	66	62.0419	1.29587
-1.00E-07	0.45	0.28	25	14	114	68	71.1464	1.29623

Finally, Table 9 shows the result of computation for randomly generated problems. These data are hardly separable by hyperplane or ellipsoidal, so that convergence is expected to be faster than real world problems.

We generated 500 quasi-random data in a hyperrectangle $[0, 10]^5$ and $[0, 10]^{10}$, and try to separate them to 2 sets of 250 data each. We see that we can obtain an optimal solution very fast even when n as large as 10.

Table 9 : Computational Performance of CP algorithm for Randomly Generated Data

tolerance	CPU-time[sec.]		#iter		#pivot	
(ε)	n=5	n=10	n=5	n=10	n=5	n=10
(1st.itr.)	0.10	0.80	1	1	64	231
-1.00E-03	0.29	2.58	4	8	167	548
-1.00E-04	0.42	3.87	8	18	195	642
-1.00E-05	0.59	5.31	14	31	210	683
-1.00E-06	0.81	6.96	22	46	224	712
-1.00E-07	0.89	9.47	25	68	227	763

6 Concluding Remarks

We discussed a new cutting plane algorithm for solving SDP's for separating a large number of low dimensional data into two groups by an ellipsoidal surface.

Standard algorithms based upon primal-dual interior point approach are efficient and stable for general class of SDP's. However, it is not efficient enough for solving a class of problems stated above, because we have to convert them into standard form of SDPA's.

Our algorithm, on the other hand, exploits the special structure of the problem. It is based upon a classical relaxation/cutting plane approach successfully applied to a problem with large number of structured linear constraints. One of the advantages of our approach is that we can employ an efficient dual simplex procedure to solve a tighter relaxation problem with one more linear constraint.

We showed in this paper that the new algorithm can solve failure discrimination problem and cancer diagnosis problem reported in earlier papers [11] very fast. The efficiency of the algorithm depends upon the degree of separation. In fact, we can solve randomly generated problem very fast, where two sets of data are located randomly. On the other hand, those problems where two sets of data can be completely separated into two groups are harder to solve.

We believe that our algorithm can be applied to other class of problems such as option pricing [3] where there are relatively small number of semidefinite constraints.

Acknowledgements : Research of the first and the third author were supported in part by the Grant-in-Aid for Scientific Research B (Extended Research) 11558046. Also, the first author acknowledges the generous support of the IBJ-DL Financial Technologies, Inc. and Toyo Trust and Banking, Co. Ltd. The second author is supported by JSPS Research Fellowships for Young Scientists.

References

[1] Altman, E.I. and Nelson, A.D.(1968), "Financial Ratios, Discriminant Analysis and the Prediction of Corporate Bankruptcy", *Journal of Finance*, **23**, 589-609.

[2] Altman, E.I.(1984), *Corporate Financial Distress*, John Wiley & Sons.

[3] Bertsimas, D. and Popescu, I.(1999), "On The Relation Between Option and Stock Prices : A Convex Optimization Approach"

[4] Bradley, P.S., Fayyad, U.M. and Mangasarian, O.L.(1999), "Mathematical Programming for Data Mining: Formulations and Challenges", *INFORMS J. on Computing*, **11**, 217-238.

[5] Chvátal, V.(1983), *Linear Programming*, Freeman and Co.

[6] Fujisawa, K., Kojima, M. and Nakata, K.(1999), "SDPA (Semidefinite Programming Algorithm) User's Manual - Version 5.00", Research Reports on Mathematical and Computing Sciences, Tokyo Institute of Technology.

[7] Gantmacher, F.R.(1959), *The Theory of Matrices*, Chelsea Pub. Co.

[8] Helmberg, C., Rendl, F., Vanderbei, R.J. and Wolkowicz, H.(1996), "An Interior-Point Method for Semidefinite Programming", *SIAM Journal on Optimization*, **6**, 342-361.

[9] J.P.Morgan(1997), *Credit Metrics*^{TM}

[10] Johnsen, T. and Melicher, R.W.(1994), "Predicting Corporate Bankruptcy and Financial Distress : Information Value Added by Multinomial Logit Models", *J.of Economics and Business*, **46**, 269-286.

[11] Konno, H. and Kobayashi, H.(2000), "Failure Discrimination and Rating of Enterprises by Semi-Definite Programming", to appear in *Asia-Pacific Financial Markets*

[12] Mangasarian, O., Street, W. and Wolberg, W.(1995), "Breast Cancer Diagnosis and Prognosis Via Linear Programming", *Operations Research*, **43**, 570-577.

[13] Rosen, J.B.(1965), "Pattern Separation by Convex Programming", *J.of Mathematical Analysis and Applications*, **10**, 123-134.

[14] Vandenberghe, L. and Boyd, S.(1996), "Semi-Definite Programming", *SIAM Review*, **38**, 49-95.

[15] Wolkowicz, H., Saigal, R. and Vandenberghe, L.(2000), *Handbook of Semidefinite Programming - Theory, Algorithms, and Applications*, Kluwer Academic Publishers