Chapter 18

FAILURE DISCRIMINATION BY SEMI-DEFINITE PROGRAMMING

Hiroshi Konno
Department of Industrial and Systems Engineering,
Chuo University

Jun-ya Gotoh
Institute of Policy and Planning Sciences,
University of Tsukuba

Stanislav Uryasev
Department of Industrial and Systems Engineering,
University of Florida

Atsushi Yuki
The Toyo Trust and Banking,Co.

Abstract This paper addresses itself to a new approach for failure discriminant analysis, a classical and yet very actively studied problem in financial engineering. The basic idea of the new method is to separate multi-dimensional financial data corresponding to ongoing and failed enterprises by an ellipsoidal surface which enjoys a good mathematical property as well as a clear financial interpretation.

We will apply a new cutting plane algorithm for solving a resulting semi-definite programming problem and show that it can generate an optimal solution in a much more efficient way than standard interior point algorithms. Computational results using financial data of Japanese enterprises show that the ellipsoidal separation leads to significantly better results than the hyperplane separation. Also it performs better than the separation by a general quadratic surface, a well used method in support vector machine approach.

Keywords: failure discriminant analysis, semi-definite programming, ellipsoidal separation, cutting plane algorithm, data-mining, support vector machine

1. Introduction

Quantitative analysis of credit risk attracted more attention of researchers as well as practitioners in recent years due to a number of failures of medium to large scale enterprises during the last decade. This trend is expected to be further intensified due to the new BIS regulation to be implemented in the year 2004, by which banks are required to evaluate the credit risk of enterprises in a more reliable way.

In this paper, we will propose a new approach for failure discrimination, one of the major areas in credit risk analyses. Failure discriminant analysis has a long history since the pioneering works of Altman [1, 2] in 1960's, where he assumed that the financial data of failure group and ongoing group are both normally distributed and separated them by a hyperplane. In subsequent "statistical" discriminant analysis, some kind of assumption on the distribution of financial data plays an essential role.

Recently, in accordance with an increasing concerns about failures, a number of new approaches evolved supported by a rapid progress in computing technologies. Among them are various methods developed in artificial intelligence and data mining. Significant differences of these approaches from the statistical discriminant analysis is that no assumptions are imposed on the distribution of financial data.

One of the authors, encouraged by the remarkable success of linear programming based data mining approach to cancer diagnosis [12], applied the same idea to failure discrimination. But the computational results were not convincing enough.

There are two reasons why linear separation does not work well in this case. First, many if not all financial indexes are correlated to each other, which linear model cannot take into account. Second, although most indexes have monotonic property in the sense that larger (or smaller) value is associated with better performance, some indexes have the "mid-value property", namely that enterprises are considered to perform well when its value is in some (unknown) interval.

Natural extension of hyperplane separation is quadratic separation, which is common in support vector machine approach (SVM). However, general quadratic surface may generate disconnected regions for each group, which is not desirable due to the basic property of financial indexes explained above.

To restrict the discriminant region of each group to be connected, we need to impose a condition that the surface is either ellipsoid or paraboloid. Separation of multi-dimensional data by an ellipsoid was first proposed by Rosen [14] in as early as 1965. However, no one pursued this approach since no efficient method to generate such an ellipsoid was known until mid 1990's, when a class of primal-dual interior point of algorithms were developed for semi-definite programming problems (SDP).

It is reported in [10] that ellipsoid separation performs better than hyperplane separation for failure discrimination problems. However, only 7 out of 455 test data belong to failure group, so that the result was inconclusive. Also, one significant disadvantage of ellipsoidal separation is that the computation time for solving a resulting SDP is much larger than that for solving a hyperplane or quadratic separation problem. When the number of data is 455 and the dimension n of the data is 6, hyperplane separation problem can be solved in less than 0.1 second, while ellipsoidal separation requires around 1000 seconds by SDPA5.0, a well designed code for SDP's.

The purpose of this paper is to apply a more efficient cutting plane algorithm recently developed by the authors [11] and demonstrate that ellipsoidal separation performs better than its counterpart, hyperplane separation and quadratic separation.

In Section 2, we introduce three alternative schemes, *i.e.*, *hyperplane separation, quadratic separation* and *ellipsoidal separation* and discuss mathematical and geometric properties of each method in detail. Section 3 will be devoted to the description of a cutting plane algorithm for solving an SDP formulated in Section 2.

In Section 4, we will present the result of numerical simulation using up to 9 dimensional financial data of small to medium scale Japanese enterprises. It will be shown that the quality of ellipsoidal separation is better than its counterparts. Also, we will show that cutting plane algorithm can generate an optimal solution much faster than SDPA5.0.

Finally, in Section 5, we will summarize the paper and discuss future direction of research.

2. Mathematical Formulation of Hyperplane, Quadratic and Ellipsoidal Separation

2.1. Separation by a Hyperplane

Let A_i, $i = 1, \dots, m$ be going enterprises and B_l, $l = 1, \dots, h$ 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

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

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

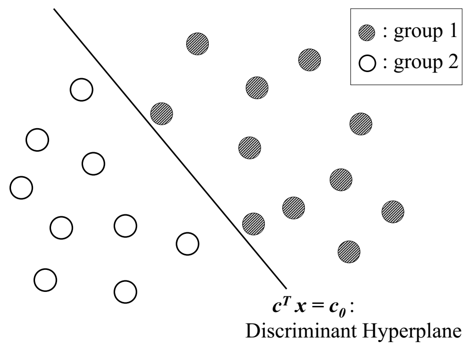

Figure 18.1. Discriminant Hyperplane

then we will call

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

a discriminant hyperplane. (Figure 18.1) Upon normalization, condition (1) and (2) are equivalent to the following (See [12].):

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

Discriminant hyperplane may not exist in general. In such a case, an enterprise A_i such that $c^T a_i < c_0 + 1$ will be called a misclassified enterprises of the first kind. Also, B_l such that $c^T b_l > c_0 - 1$ will be called the misclassified enterprise of the second kind. Associated with misclassified data a_i, let y_i be the distance of a_i from the hyperplane $c^T x = c_0 + 1$. Also, for those misclassified data b_l, let z_l be the distance from the hyperplane $c^T x = c_0 - 1$ (See Figure 18.2). To minimize the weighted sum of these errors, we consider the following linear programming problem:

$$\left| \begin{array}{ll} \text{minimize} & (1-\lambda)\dfrac{1}{m}\displaystyle\sum_{i=1}^{m} y_i + \lambda \dfrac{1}{h}\sum_{l=1}^{h} z_l \\ \text{subject to} & c^T a_i + y_i \geq c_0 + 1, i = 1, \ldots, m, \\ & c^T b_l - z_l \leq c_0 - 1, l = 1, \ldots, h, \\ & y_i \geq 0, i = 1, \ldots, m, \\ & z_l \geq 0, l = 1, \ldots, h, \end{array} \right. \tag{5}$$

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.

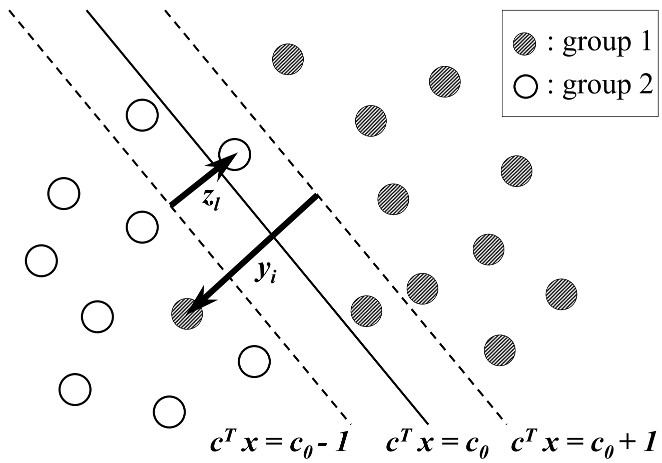

Figure 18.2. Hyperplane by (5)

Let us note that the problem (5) 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 3 dimensional physical data. According to their report, 97.5% of the 569 patients are classified correctly.

In addition, they conducted out of sample test (prediction) by using the following method:

(i) Split all data in K groups D_k, $k = 1, \ldots, K$, randomly.

(ii) Choose D_k and generate a discriminant hyperplane using data contained in D_k, $k \neq i$ and count the number of correct prediction using D_i.

(iii) Repeat this procedure for each i and calculate the average.

According to [12], 97.5% of the data were correctly predicted, which is comparable to the precision of diagnoses by experts in this field. (INFORMS's Lanchester Prize of the year 2000 was awarded to O.Mangasarian for this remarkable accomplishments.)

One of the authors, inspired by this success, applied the same method to failure discrimination, using six dimensional financial data of 455 enterprises. However, the result was not convincing. In fact, only 92% of the data were

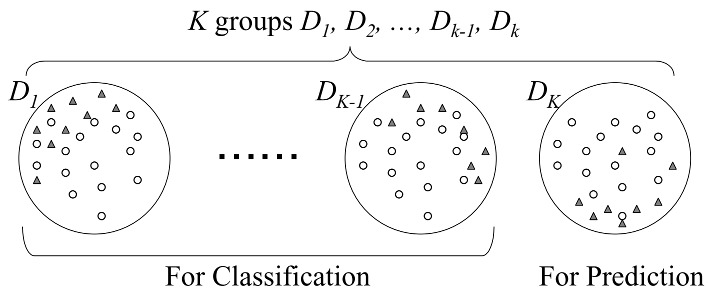

Figure 18.3. Cross Validation

correctly classified and prediction procedure could not be conducted because the failure group consists of only 7 data.

Discrimination of financial data is admittedly more difficult than physical data. For one thing, financial data are less reliable since they are calculated from less precise data. Even worse, financial data of small companies are often subject to imprecise or wrong procedure if not window-dressing procedure.

2.2. Separation by Quadratic Surface

To improve precision of discrimination, let us consider the separation by a quadratic surface. Let $D \in \mathbb{R}^{n\times n}$ be a symmetric matrix and consider the following minimization problem:

$$\left|\begin{array}{ll} \text{minimize} & (1-\lambda)\frac{1}{m}\sum_{i=1}^{m} y_i + \lambda\frac{1}{h}\sum_{l=1}^{h} z_l \\ \text{subject to} & a_i^T D a_i + a_i^T c + y_i \geq c_0 + 1, i = 1,\ldots,m, \\ & b_l^T D b_l + b_l^T c - z_l \leq c_0 - 1, l = 1,\ldots,h, \\ & y_i \geq 0, i = 1,\ldots,m, \\ & z_l \geq 0, l = 1,\ldots,h, \end{array}\right. \tag{6}$$

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. Here, a discriminant quadratic surface $Q(D,c,c_0)$ is defined by

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

Variables y_i and z_l represent, respectively the distance of the misclassified data a_i and b_l from the quadratic surfaces $x^T Dx + c^T x = c_0 \pm 1$.

The problem (6) is a linear programming problem. It is easy to see that this problem has an optimal solution. Since this problem contains $n(n+1)/2$ additional variables compared with (5), the value of objective function should be much smaller than that of hyperplane separation. On the other hand, the configuration of separating surface can be very complicated. In particular, if D is indefinite, then we will obtain a disconnected region of discrimination. This is not desirable since financial data satisfy either monotonic or mid-value property discussed in Introduction. Furthermore, this is why we do not apply the SVM with standard non-linear kernels.

It is reported in [10] that we usually obtain 100% correct classification by this method. However, it often results in an overfitting, so that the quality of prediction would deteriorate.

2.3. Separation by an Ellipsoid

In order to generate two connected regions of discrimination and to avoid overfitting, we will restrict the separating surface to be convex. This implies that D is either positive or negative semi-definite. Therefore, we have to solve the following "semi-definite" programming problem:

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

where $D \succeq O$ denotes that D is positive semi-definite. The constraint $D \succeq O$ generates an ellipsoid (or paraboloid) containing b_l, $l = 1,\ldots,h$ within the ellipsoid, while $-D \succeq O$ generates an ellipsoid containing a_i, $i = 1,\ldots,m$ within an ellipsoid. Let us note that ellipsoidal separation imposes a tight restriction on the component of D.

As reported in [10], 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 (7) is over 1000 times more than that for solving the associated linear program (6) when $n = 6$ and $N \equiv m + h = 455$ [10]. The problem to be solved in practice is much larger, *i.e.*, N is over a few thousand while n remains small. Therefore, we need to develop a more efficient algorithm for solving (7) using its special structure.

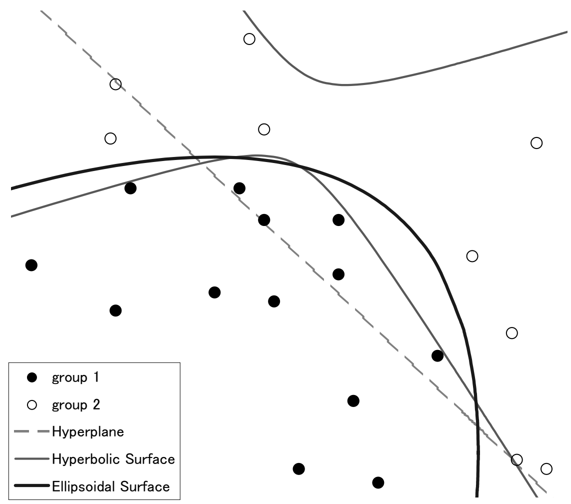

Figure 18.4. Hyperplane, Hyperbolic Surface, Ellipsoidal Surface.

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

$$\mathcal{F}_0 = \left\{ (D, c, c_0, y, z) \;\middle|\; \begin{array}{l} a_i^T D a_i + a_i^T c + y_i \geq c_0 + 1, i = 1, \ldots, m, \\ b_l^T D b_l + b_l^T c - z_l \leq c_0 - 1, l = 1, \ldots, h, \\ y_i \geq 0, i = 1, \ldots, m, z_l \geq 0, l = 1, \ldots, h, \\ d_{jj} \geq 0, j = 1, \ldots, n. \end{array} \right\} \tag{8}$$

and denote (7) in a compact form as follows :

$$\text{(P)} \left| \begin{array}{ll} \text{minimize} & (1-\lambda) e_m^T y + \lambda e_h^T z \\ \text{subject to} & (D, c, c_0, y, z) \in \mathcal{F}_0 \\ & x^T D x \geq 0, \forall x \in \mathcal{B}_n, \end{array} \right. \tag{9}$$

where $\mathcal{B}_n$ is an n-dimensional unit ball.

An important observation is that this belongs to a class of semi-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 :

$$(\mathrm{Q}_0) \left| \begin{array}{ll} \text{minimize} & (1-\lambda)e_m^T y + \lambda e_h^T z \\ \text{subject to} & (D, c, c_0, y, z) \in \mathcal{F}_0 \end{array} \right. \qquad (10)$$

by relaxing the last constraint of (9). 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 (7). If D^o is not positive semi-definite, then there exists $x \in \mathcal{B}_n$ such that $x^T D^o x < 0$.

Let us consider the quadratic program :

$$(\pi_0) \left| \begin{array}{ll} \text{minimize} & x^T D^o x \\ \text{subject to} & x \in \mathcal{B}_n. \end{array} \right. \qquad (11)$$

Lemma 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 (11) 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

$$\mathcal{F}_1 = \mathcal{F}_0 \cap \{ \, D \mid (x^o)^T D x^o \geq 0 \, \} \qquad (12)$$

and define a tighter linear program

$$(\mathrm{Q}_1) \left| \begin{array}{ll} \text{minimize} & (1-\lambda)e_m^T y + \lambda e_h^T z \\ \text{subject to} & (D, c, c_0, y, z) \in \mathcal{F}_1 \end{array} \right. \qquad (13)$$

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

$$(\mathrm{Q}_k) \left| \begin{array}{ll} \text{minimize} & (1-\lambda)e_m^T y + \lambda e_h^T z \\ \text{subject to} & (D, c, c_0, y, z) \in \mathcal{F}_k \end{array} \right. \qquad (14)$$

where

$$\mathcal{F}_k = \mathcal{F}_{k-1} \cap \left\{ \, D \mid (x^{k-1})^T D x^{k-1} \geq 0 \, \right\} \qquad (15)$$

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

$$(\pi_k) \left| \begin{array}{ll} \text{minimize} & x^T D^k x \\ \text{subject to} & x \in \mathcal{B}_n. \end{array} \right. \qquad (16)$$

Let x^k be its optimal solution, for which the objective value of (π_k) is the minimal eigenvalue of D^k 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 (8) 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 α_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]

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 ε-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

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

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 2 $(D^k, c^k, c_0^k, y^k, z^k)$ *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.

[1]Note that α_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).

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

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

as $k \to \infty$. 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 $k \to \infty$.

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 $\alpha_{j_k} \to \infty$, 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 Discrimination

We will compare the performance of four separation schemes:

(a) hyperplane separation (equation (5))
(b) quadratic separation (equation (6))
(c) ellipsoidal separation enclosing failure group (equation (7) with $D \succeq O$)
(d) ellipsoidal separation enclosing ongoing group (equation (7) with $-D \succeq O$)

using two sets of data. Data set 1 consists of up to 9 dimensional financial data of 428 small to medium scale Japanese enterprises supplied by Shoko Research, Co., among which 40 data belongs to failure group. (We excluded small companies whose number of employees are less than 50 or greater than 1000.) The size of this data may not be large enough to derive convincing results, but failure data are rarely available in the public domain contrary to physical data used for cancer diagnosis [12]. Most of these data are kept confidential as a top secret of banks.

Table 18.1. Attributes Set No.1

(1)	ratio of net worth (equity ratio)
(2)	ratio of fixed assets to long term-capital
(3)	current liability
(4)	interest coverage
(5)	net profit margin
(6)	liquidity in hand
(7)	cash flow
(8)	ROA (rate of return on asset)
(9)	ratio of operating cash flow to current liabilities

Table 18.2. Optimal Value of Each Method

	n : # of attributes		
	3	6	9
(a) Hyperplane	0.637	0.619	0.599
(b) Quadratic	0.527	0.268	0.119
(c) Ellipsoid1	0.529	0.305	0.192
(d) Ellipsoid2	0.636	0.619	0.597

Following are the financial indexes (attributes) used in our analysis:
We used the following procedure to choose these indexes.

Step 1. Calculate 18 potentially meaningful indexes using the arithmetic mean of balance sheet data of 428 companies of the year 1997 and 1998.

Step 2. For each k ($k = 3 \sim 9$), generate all possible combinations of k indexes out of 18 indexes and choose the best one in terms of the precision of classification using ellipsoidal surface (for 428 companies).

It turned out that the best combination for $k-1$ indexes is the subset of the best combination for k indexes. The reason why we limited k up to 9 is twofold: The first reason is that the amount of computation required to solve semi-definite programming problem sharply increases as we increase k beyond 10. The second reason is that using too many indexes for 428 available data is expected to result in overfitting.

Table 18.2 shows the value of the objective function associated with optimal solutions when $\lambda = 0.5$. Each column corresponds to the computation using first 3, 6, 9 indexes listed above.

As expected, the optimal value of quadratic separation (b) is the smallest. Ellipsoidal separation (c) is a little worse but is much better than hyperplane. We see a significant improvement of the solution for (b) and (c) as we increase n, while that of (a) is very marginal.

Table 18.3 shows the percentage of classification errors. We see from this table that (c) is consistently better than (d). Hence, we eliminate scheme (d) from the subsequent comparison for this data set.

Table 18.4 shows the number of misprediction. Both (b) and (c) are better than (a). The difference becomes larger as we increase the number of indexes. On the other hand, the difference between (b) and (c) are not very significant.

Quality of separation depends upon choice of financial indexes. Various statistical methods are now being applied to find a better set of indexes. Following are the ones proposed by one of our colleagues Ms.D.Wu.

We compared four classification schemes (a), (b), (c), (d) using data set 2 compiled from balance sheets of 1701 companies including those small companies (whose employees are less than 50), which were excluded in the previous simulation.

Table 18.6 and 18.7 show, respectively the number of misclassification and misprediction. We see from this that scheme (d) which encloses ongoing companies within the ellipsoid performs best in terms of both classification and prediction. In fact, (d) consistently dominates (a). The number of serious errors, *i.e.*, the number of significant misclassified and mispredicted companies of scheme (d), are much less than others.

Best performance of classification is achieved by (b), but (d) outperforms (b) in prediction. The overall quality of prediction is worse than the results for data set 1, but we have more stable results by data set 2. We conclude from this that the ellipsoidal separation is subject to smaller risk of overfitting compared with data set 1, although the precision of classification and prediction is worse. This is due to the fact that we used all available data including those less reliable ones associated with smaller companies.

Which of the two schemes (c) or (d) is better is yet to be tested by more extensive simulation. Table 18.8 shows the relation between the level of tolerance ε and the CPU time to calculate an ε-optimal solution for data set 1 and 2.

Table 18.3. Number of Misclassification

	$n = 3$			$n = 6$			$n = 9$		
	total	ongoing	failure	total	ongoing	failure	total	ongoing	failure
(a)	132 (30.99%)	126 (32.64%)	6 (15.00%)	135 (31.69%)	133 (34.46%)	2 (5.00%)	136 (31.92%)	131 (33.94%)	5 (12.50%)
(b)	121 (28.40%)	118 (30.57%)	3 (7.50%)	66 (15.49%)	64 (16.58%)	2 (5.00%)	26 (6.10%)	25 (6.48%)	1 (2.50%)
(c)	122 (28.64%)	118 (30.57%)	4 (10.00%)	76 (17.84%)	74 (19.17%)	2 (5.00%)	46 (10.80%)	44 (11.40%)	2 (5.00%)
(d)	132 (30.99%)	126 (32.64%)	6 (15.00%)	135 (31.69%)	133 (34.46%)	2 (5.00%)	137 (32.16%)	132 (34.20%)	5 (12.50%)

The number of samples are as follows: ongoing : 386, failure : 40.

Table 18.4. Number of Misprediction

	$n = 3$			$n = 6$			$n = 9$		
	total	ongoing	failure	total	ongoing	failure	total	ongoing	failure
(a)	140 (32.86%)	133 (34.46%)	7 (17.50%)	142 (33.33%)	135 (34.97%)	7 (17.50%)	145 (34.04%)	137 (35.49%)	8 (20.00%)
(b)	126 (29.58%)	120 (31.09%)	6 (15.00%)	76 (17.84%)	64 (16.58%)	12 (30.00%)	82 (19.25%)	59 (15.28%)	23 (57.50%)
(c)	128 (30.05%)	122 (31.61%)	6 (15.00%)	86 (20.19%)	77 (19.95%)	9 (22.50%)	77 (18.08%)	60 (15.54%)	17 (42.50%)

The number of samples are as follows: ongoing : 386, failure : 40.

Table 18.5. Attribute Set No.2

(1)	turnover of discounted and negotiable bills
(2)	acid ratio (quick ratio)
(3)	ratio of working capital to total assets
(4)	earnings per person (earnings per employee)
(5)	ratio of sales to total capital
(6)	average interest rate on debt
(7)	rate of net worth (equity ratio)

Table 18.6. Number of Misclassification

	total		ongoing		failure		average
(a)	376	(22.10%)	355	(22.44%)	21	(17.65%)	20.05%
(b)	353	(20.75%)	341	(21.56%)	12	(10.08%)	15.82%
(c)	441	(25.93%)	424	(26.80%)	17	(14.29%)	20.55%
(d)	359	(21.11%)	341	(21.56%)	18	(15.13%)	18.35%
#samples	1701		1582		119		

Table 18.7. Number of Misprediction

	total		ongoing		failure		average
(a)	398	(23.40%)	373	(23.58%)	25	(21.01%)	22.29%
(b)	374	(21.99%)	350	(22.12%)	24	(20.17%)	21.15%
(c)	450	(26.46%)	423	(26.74%)	27	(22.69%)	24.71%
(d)	371	(21.81%)	350	(22.12%)	21	(17.65%)	19.89%
#samples	1701		1582		119		

Table 18.8. CPU time [sec.] of CPA

		Data Set 1			Data Set 2	
# of attributes		n=3	n=6	n=9	n=7 (c)	n=7 (d)
(1st.itr.)		0.05	1.20	1.41	1.58	1.77
ε	10^{-3}	0.11	2.22	16.68	19.92	20.60
	10^{-4}	0.12	2.71	23.04	25.04	24.92
	10^{-5}	0.14	3.17	29.57	31.11	30.22
	10^{-6}	0.16	3.66	36.87	36.48	34.90
	10^{-7}	0.19	4.12	-	41.17	42.27

All computation were conducted on PentiumIII Processor (500MHz) using C/C++. Also, linear programming subproblems were solved by CPLEX6.5. We see from this table that a problem with $n = 7$ and $N \equiv m + h = 1701$ can be solved in less than 40 seconds. We believe that more elaborate implementation of the cutting plane algorithm would be able to solve problems such as $n = 9$ and N is over a few thousand within a practical amount of time.

5. Concluding Remarks and Future Direction of Research

In this paper, we compared four schemes of failure discriminant analysis. According to simulation using financial data of Japanese enterprises, separation by ellipsoidal surface performs best among them. We also showed that the precision of classification improves as we increase the number of enterprises N $(= m + h)$ and number of financial indexes n, Also, the precision of prediction improves as we increase N. However, it need not improve or even deteriorate as we increase n beyond some bound.

The choice of appropriate indexes is crucial for this kind of analysis as observed through comparison of results for data set 1 and data set 2. In particular, those indexes which work good for some category of enterprises may not work well for other categories. The detailed statistical analysis on the good choice of indexes is now under way and will be reported in the subsequent paper.

Finally, let us remark that the result of separation may be used for rating the enterprises. Those enterprises located very far from the boundary into the ongoing direction are very unlikely to fail in the near future. Therefore, they are entitled to have a high rating score. On the other hand, those enterprises located near the boundary or in the failure side will be rated poorly because they are more likely to fail. Therefore, we can rate the ongoing enterprises according to the distance from the discriminant surface.

Precise rating requires a significant amount of time and cost, so that only large scale enterprises can be subject to full scale rating procedure. Therefore, some kind of cheap and quick method is required to rate thousands of small to medium scale enterprises. We believe that the score generated by ellipsoidal separation can be used for this purpose.

Further extension of our method is the separation of enterprises by multi-layer ellipsoidal surface. We may first classify enterprises into two groups by the method proposed above and then classify the ongoing group into two groups by another ellipsoid and so on.

Let us add that we need to conduct more extensive simulation to establish the superiority of ellipsoidal separation over the other existing methods such as decision trees and neural networks.

Acknowledgments

The research of the first author was supported in part by the Grant-in-Aid for Scientific Research B(2)12480105 of the Ministry of Education, Science and Culture and the Hitachi Research Institute, Hitachi Co.. Research of the second author was supported by JSPS Research Fellowships for Young Scientists. Also, the authors acknowledge the generous support of IBJ-DL Financial Technologies, Inc. and the Toyo Trust and Banking, Co..

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] 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.

[10] Konno, H. and Kobayashi, H.(2000), "Failure Discrimination and Rating of Enterprises by Semi-Definite Programming", to appear in *Asia-Pacific Financial Markets*, **7**

[11] Konno, H., Gotoh, J. and Uno, T.(2000), "A Cutting Plane Algorithm for Semi-Definite Programming Problems with Applications", CRAFT Working Paper 00-05, Center for Research in Advanced Financial Technologies, Tokyo Institute of Technology, submitted to *Optimization and Engineering*.

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

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

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

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

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