Journal of Global Optimization **11:** 433–462, 1997.

The Surgical Separation of Sets

JAMES E. FALK
Department of Operations Research, The George Washington University, Washington, DC 20052, U.S.A.

EMMA LOPEZ-CARDONA
Office of Environmental Management, Department of Energy, Washington, DC 20585, U.S.A.

(Received: 8 September 1994; accepted: 2 April 1997)

Abstract. Given a pair of finite disjoint sets A and B in R^n, a fundamental problem with many important applications is to efficiently determine a hyperplane $H(w, \lambda)$ which separates these sets when they are separable, or 'nearly' separates them when they are not. We seek a hyperplane which minimizes a natural error measure in the latter case, and so will 'surgically' separate the sets. When the sets are separable in a strong sense, we show that the problem is a convex program with a unique solution, which has been investigated by others. Using the KKT conditions, we improve on an existing algorithm. When the sets are not separable, the problem is nonconvex, generally with proper local solutions, and we solve an equivalent problem by Branch and Bound. Numerical results are presented.

Key words: Pattern recognition, non-convex optimization, set separation.

1. Introduction and Preview

The terms 'discriminant analysis', 'pattern classification', and 'separation of sets' are different expressions describing the common and practically important problem of partitioning R^n into t subsets, based on a given set of data points composed of t types of data. The applications include: medical diagnosis, scoring of credit card applications, biological specimen classification, digit and handprinted character recognition, and identification of tax returns for audit.

The general problem is simple to describe. We are given a finite set D of data points (vectors in R^n), with each data point belonging to exactly one of t types D^j. Based on this given set $D = D^1 \cup D^2 \cup \cdots \cup D^t$, we wish to associate *any* point in R^n with one of the t types. There are two quite different approaches to this problem, 'parametric' and 'non-parametric' where the former applies to methods wherein one assumes that the points of D are outcomes of random experiments whose underlying random variables are of a certain type with parameters to be estimated from D. We are interested in non-parametric approaches wherein no such statistical assumptions are made.

We thus seek to partition R^n into t subsets $S^1, S^2, \ldots, S^t$ (so $S^1 \cup S^2 \cup \cdots \cup S^t = R^n$, and $intS^i \cap intS^j = \emptyset$ for all $i \neq j$). In this manner, we can associate *any* point in R^n with one of the subsets S^i (ties are not addressed here, it is assumed that they

can be broken according to some unambiguous rule) and thus 'classify' all points. A simple (but potentially computationally expensive) scheme is to associate each point with the type of its closest Euclidean neighbor. In this way, R^n is partitioned into as many 'cells' as there are points in D — this is the so-called *Voronoi Partition* (see, e.g., Preparata [21], and we set S^j equal to the union of all cells of all points in D^j).

In this paper we address the most basic problem of separating *two* finite disjoint sets of points A and B. With some abuse of notation, we will use the same symbols to denote matrices whose rows are the points of A and B. Let $|A| = p$ and $|B| = q$, so that A is p by n and B is q by n. We are interested in *linearly* separating A and B whenever possible, i.e., we are interested in efficiently determining a hyperplane $H(w, \gamma)$ (so that $w \neq 0$) for which $A_i w \geq \gamma$ for $i = 1, 2, \ldots, p$ and $B_j w \leq \gamma$ for $j = 1, 2, \ldots q$ when possible, or minimizing an error function when no such hyperplane exists.

There are a number of reasons why we limit our attention to the two-set separation problem, and among them are:

- the multi-set problem can be solved by solving a sequence of two-set separation problems (e.g., see [24]),
- two (or more) disjoint sets can be completely separated by solving a sequence of linear separating problems (e.g., see [15]), and
- the two set linear separation problem has interest in its own right.

To illustrate the latter point, we briefly describe an application of the two-set separation problem to the detection of breast cancer (see [16] and [17] for a complete description of the model). We discuss the accuracy of our model on this data set in Section 7. Here the database D consists of 683 points (535 in an early study), each a vector in Euclidean nine-space, R^9. Each vector corresponds to a tissue sample – fine needle aspirate (FNA) of human breast tissue taken from a specific patient. Each of the nine components of the vector corresponds to a specific attribute of the sample*. The numbers in each component are integers ranging from 1 to 10, and represent a pathologist's judgement as to the degree that the given sample displays this attribute. A sample point is classified into set A (or set B) depending on the presence (or absence) of cancer in that sample. In the aforementioned database, 239 of the 683 samples were cancerous.

When a new patient's sample - say P - is evaluated, the problem is to associate it with one of the two sets A or B. In the event that these sets are separable by a hyperplane $H(w, \gamma)$, we then would associate it with the set A if $P \cdot w \geq \gamma$, and with the set B otherwise. In the event that the sample is classified with the malignant set, the patient would undergo a biopsy, otherwise a re-examination would be recommended to confirm the diagnosis. When A and B are not linearly separable, a possible approach is to construct a piecewise linear separator by

* The nine attributes are: clump thickness, uniformity of cell size, uniformity of cell shape, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli, and mitosis.

enclosing the ambiguous points in a band (a pair of parallel hyperplanes) according to some rule, eliminate the correctly classified points, then try to linearly separate the remaining ambiguous points. Several applications of this procedure might be needed to completely separate A and B — Mangasarian [15] describes a simple way of accomplishing this.

We now preview the contents of this paper. In the next section, we define weak and strong separation, and motivate and define the basic optimization model to be solved. In the same section we mention some of the previous models and work which has been done.

In Section 3 we relate the optimal value of the basic optimization problem and the nature of the level sets of its objective function to the nature of the separability of the sets A and B. The nature of this separability can be identified by solving a linear program. When the sets are separable, the basic optimization problem is a convex program, otherwise it is nonconvex.

Sections 4 and 5 contain algorithms for solving the basic optimization problem, with Section 4 concentrating on the strongly separable case, and Section 5 concentrating on the weakly separable and the nonseparable cases. The algorithm offered for the strongly separable case of Section 4 is an enhanced version of a cutting plane method originally proposed by Cavalier, Ignizio and Soyster [3]. A method to solve the nonconvex problem corresponding to the nonseparable case is described in Section 5. To set up the method, a sequence of $2 \cdot n$ linear programs must first be solved. If any of these problems is unbounded, the sets A and B are identified as being weakly separable, and a separating hyperplane is determined.

In Section 6 we describe some numerical experience that we have had with the method on a set of randomly generated problems. In Section 7, we apply the algorithm to the Breast Cancer Data described above in order to test the accuracy of the model's hyperplane on a set of actual data.

With the exception of using the same symbol (A and B) to denote both a set of vectors and a matrix whose rows and columns are the vectors, the notation used herein is standard. We will often use the symbol e (with no subscript) to denote a column vector of ones, with the number of rows determined by the context, so that when we write $A \cdot w \geq \gamma \cdot e$ we know that e has p rows, whereas in the inequalities $B \cdot w \leq \gamma \cdot e$, the vector e has q rows.

2. The Basic Optimization Model

We have assumed that the sets A and B are disjoint. Following Marlow [19], we say that these sets are **separable** if there is some hyperplane $H(w, \gamma)$ such that

$$A_i \cdot w \geq \gamma \text{ for all } i = 1, \ldots p \text{ and } B_j \cdot w \leq \gamma \text{ for all } j = 1, \ldots, q$$

where we can assume that $\|w\| = 1$ ($H(0, \gamma)$ is not a hyperplane and $H(kw, k\gamma)$ is the same hyperplane as $H(w, \gamma)$ for any $k > 0$.) If the sets are separable and there is a hyperplane that satisfies these inequalities in a strict sense, we say that these

sets are **strongly separable**, otherwise we say that the sets are **weakly separable**[⋆]**.** If no hyperplane exists that satisfies the above inequalities, the sets are simply **not separable**.

Note that a given pair of sets A and B satisfies exactly one of the three relations: they are either strongly separable, weakly separable, or not separable. Note also that when A and B are strongly separable there are an infinite number of distinct hyperplanes which separate them. If A and B are weakly separable, there may or may not be an infinity of separating hyperplanes (e.g., if $A = \{(-1, 0), (0, 1), (1, 0)\}$ and $B = \{(0, 0)\}$ in R^2, the only separating hyperplane is $w = (0, 1), \gamma = 0$, but if A and B are considered to be subsets of the plane $x_3 = 0$ in R^3, there is an infinite number of separating hyperplanes.)

When A and B are strongly separable, we are interested in finding a hyperplane which 'robustly' separates the sets (in which case we say that A and B are 'surgically' separated.) While any separating hyperplane will separate A and B, we imagine that these sets are simply finite representatives of larger sets $\hat{A}$ and $\hat{B}$ which are in fact the sets which we want to separate. If $\hat{A}$ and $\hat{B}$ are fairly close to A and B in size then the hyperplane which our model predicts is likely to also separate $\hat{A}$ and $\hat{B}$.

We also would like the separating hyperplane to be invariant under transformations that preserve congruence, i.e., if A and B undergo a transformation T which is a translation, reflection, rotation, scaling, or combination of these, then we would like our model to chose the hyperplane $T(H)$ separating $T(A)$ and $T(B)$ when our model would choose H separating A and B. Indeed, if one simply interchanges the labels A and B, our model will choose the hyperplane $(-w, -\gamma)$.

A final property of the model which we address is that it produces a *unique* hyperplane when the sets A and B are strongly separable. This has the pleasant implication that the hyperplane predicted by the model does not depend on the software used to solve the model.

When A and B are weakly separable, we are content to produce any weakly separating hyperplane in the event that a multiplicity of separating hyperplanes exists.

Finally, when A and B are not separable we want to produce a hyperplane which 'comes as close as possible' to separating A and B, i.e., a hyperplane which minimizes some measure of the 'error' associated with the attempted separation.

A number of authors have studied the problem of using linear or nonlinear programming to find hyperplanes which separate, or 'nearly separate', given sets A and B. One of the earliest papers, due to Mangasarian [15] formulated a linear programming model quite similar to the one which we address, but used the L_∞ norm on the weight vector w, and so produced a hyperplane for the strongly

[⋆] This convention is not universally accepted, especially among papers dealing with pattern recognition (e.g. [15]) where the concepts of 'strong separablility' and 'separability' are considered interchangeable and weak separability is not addressed. The reason for this is probably that in most applications, the sets are either strongly separable or they are not separable.

separable case which is not invariant under rotations. In fact, his solution will be the first LP solution of a sequence of LP solutions which our model will produce.

In the same year, Rosen [22] pointed out that the problem of identifying a separating hyperplane for the strongly separable case can be formulated as the convex quadratic problem of minimizing the distance between the convex hulls $C(A)$ and $C(B)$ of the sets A and B. Let $x_A \in C(A)$ and $x_B \in C(B)$ so that the segment $[x_A, x_B]$ 'connects' $C(A)$ and $C(B)$. If the length of $[x_A, x_B]$ is minimal over all such connecting line segments, then the hyperplane bisecting $[x_A, x_B]$ with normal $(x_A - x_B)/\|x_A - x_B\|$ is, in fact, the hyperplane which we will produce via sequential linear programming when A and B are strongly separable. A disadvantage of this approach is that no useful information is available from the solution of the quadratic program when A and B are not strongly separable.

Another early effort is due to Smith [23] who formulated a linear program whose objective is to minimize a measure of the distance of the points to the hyperplane. This model did indeed provide a separating hyperplane for the strongly separable case, but did not yield any useful information for the other cases. The method was later generalized by Bennett and Mangasarian [1] who modified the objective function by including weights which not only provided a separating hyperplane for the strongly separable case, but also provided a hyperplane minimizing an average error measure for the non-separable case. Neither of these methods produce a hyperplane invariant under conguence preserving transformations.

During the '80's and early '90's, an entire sequence of papers by and/or referencing Freed and Glover ([5–12]) appeared in the journal *Decision Sciences*. All of this work is essentially independent of the aforementioned references, and is directed at establishing linear programming models whose solutions will separate (or nearly separate) a pair of sets A and B. Some of the models only work under special conditions, and some are not invariant under all congruence preserving transformations.

The model that we will address was most recently considered by Cavalier et al, [3], who suggested a heuristic method for its solution in the strongly separable case. We will now motivate this model from several different perspectives. Let $H(w, \gamma)$ be any hyperplane, and p be any point in R^n. The Euclidean distance between p and $H(w, \gamma)$ is $|z \cdot w - \gamma| \,/\, \|w\|$ or simply $|z \cdot w - \gamma|$ if $\|w\| = 1$. Now if $H(w, \gamma)$ separates the sets A and B, then the closest point in $A \cup B$ is of distance equal to the minimum of the non-negative numbers $A_i w - \gamma$ $(i = 1, ..., p)$ and $\gamma - B_j w$ $(j = 1, ..., q)$. In the strongly separable case, we wish to maximize this quantity, i.e., we wish to solve *Problem P*:

$$\max_{\|w\|=1,\gamma} \min_{\substack{i=1,...,p \\ j=1,...,q}} \{A_i w - \gamma, \gamma - B_j w\} \tag{1}$$

Note that this problem has a piecewise linear and concave objective function, but is a non-convex program by virtue of the constraint $\|w\| = 1$. In this formulation,

we are seeking a separating hyperplane which maximizes the minimum distance between it and the data points.

Because we are using Euclidean distance as a measure here, our solution will be invariant under transformations which preserve distance.

Before addressing other interpretations of the model, it is convenient to eliminate γ from *Problem P*. For any given weight vector w, it is easy to see that the γ which minimizes the above objective function is

$$\gamma = \left(\frac{1}{2}\right)\left(\min_{i=1,\ldots,p}\{A_i w\} + \max_{j=1,\ldots,q}\{B_j w\}\right) \tag{2}$$

and, eliminating γ from the objective function of *Problem P*, we obtain the new version

$$\text{Find } V^* = \left(\frac{1}{2}\right)\max_{\|w\|=1}\left\{\min_{i=1,\ldots,p}\{A_i w\} - \max_{j=1,\ldots,q}\{B_j w\}\right\}. \tag{3}$$

In this form it is clear that *Problem P* must have a solution.

We now consider a slightly different way of interpreting *Problem P*. Let $H(w, \gamma)$ be any separating hyperplane, and let

$$\begin{aligned} s &= \min_{i=1,\ldots,p}\{A_i w\} \\ t &= \max_{j=1,\ldots,q}\{B_j w\}. \end{aligned}$$

Because the hyperplane separates A and B, $t \le \gamma \le s$, and the quantity $(s - \gamma)$ is the distance of the closest point in A to H, and $(\gamma - t)$ is the distance of the closest point in B to H. The set $B(H) = \{x \in R^n : t \le xw \le s\}$ is a 'band' of width $s - t$, whose interior contains no points of either set A or B. Lambert [13] refers to $B(H)$ as the 'dead zone'. *Problem P seeks to find a hyperplane whose dead zone is of maximal width.* Figure 1 shows two strongly separated sets, a separating hyperplane, and the dead zone. Note that the hyperplane shown does not produce the maximal dead zone.

As a final interpretation of *Problem P*, consider the related problem:

$$\min_{x \in C(A), y \in C(B)} \|x - y\|$$

where $C(A)$ and $C(B)$ are the convex hulls of the sets A and B respectively. Here we are seeking a line segment $[x^*, y^*]$ of minimallength joining the convex hulls of the sets A and B. This is the problem addressed by Rosen, [22]. It turns out that there is an intimate relationship between a solution of this problem, and a solution of *Problem P*, which we will summarize in the following statement. We will not include a proof here but one can be realized by comparing the KKT conditions for the above problem with a problem equivalent to *Problem P* (one based on duality theory can be found in [14].)

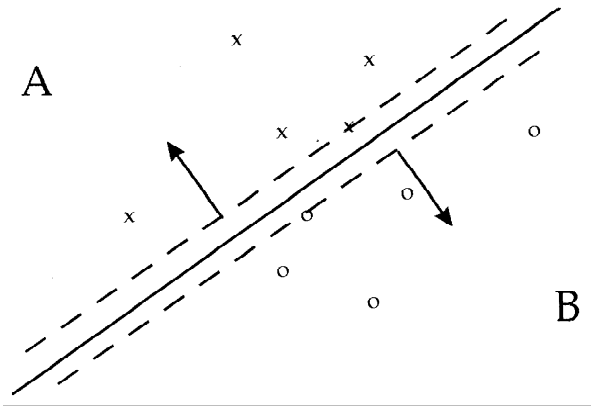

Figure 1. A dead zone for two strongly separable sets.

THEOREM 1. *Assume the sets A and B are strongly separable, and let (x^*, y^*) denote a solution to the above problem. Then the hyperplane $H(w, \gamma)$ with $w = (x^* - y^*)/\|x^* - y^*\|$ and $\gamma = (1/2) \cdot (\|x^*\|^2 - \|y^*\|^2)$ is a solution to Problem P.*

In connection with this last interpretation of *Problem P*, note that:

- The above problem must have a solution, and
- The solution of the above problem might not be unique (e.g., let $A = \{(0,0),(1,0)\}$ and $B = \{(0,1),(1,1)\}$ in R^2.
- When A and B are weakly separable, a solution of the above problem offers no separating hyperplane since in this case $x^* = y^*$, so that $w = 0$.

We now turn to the case where the sets A and B are not linearly separable. In this case, for any hyperplane $H(w, \gamma)$, at least one of the numbers $(A_i w - \gamma)$ or $(\gamma - B_j w)$ must be negative, with the most negative values corresponding to the worst misclassified points. If *we choose to seek a hyperplane whose worst misclassified points are as near as possible to being correctly classified* (as measured by the Euclidean norm), we are led to maximize the smallest of the above numbers, i.e., we again seek a solution to *Problem P.*

As above, let

$$s = \min_{i=1,\ldots,p} \{A_i w\}$$
$$t = \max_{j=1,\ldots,q} \{B_j w\}.$$

so that $Aw \geq s \cdot e$ and $Bw \leq t \cdot e$. Since we are assuming that the sets A and B are not separable, $s < t$, for otherwise with w given, there is some value of $\gamma^* \in [t, s]$ such that the hyperplane $H(w, \gamma^*)$ separates A and B. Note that all points of A such that $A_i w \geq t \cdot e$ are correctly classified, and all points of B such that $B_j w \leq s \cdot e$ are correctly classified by any hyperplane $H(w, \gamma)$ where $s \leq \gamma \leq t$. However, the band of points $B(H) = \{x : s \leq xw \leq t\}$ contains both points in A and points of

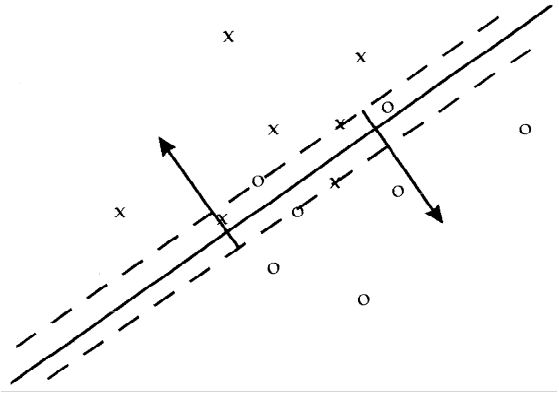

Figure 2. An ambiguous strip for two nonseparable sets.

B, and is the narrowest band of such points for the given weight vector w, with a width of $(t - s)$. Lambert [13] refers to this as the 'negative dead zone' but we will refer to it as the 'ambiguous zone'. Figure 2 exhibits it for a pair of non-separable sets. Note that the ambiguous zone shown is not minimal. *We seek to determine w (and the corresponding γ via equation (2.2)) which minimizes the width of the ambiguous zone*, i.e., one which maximizes $(s - t)$. Thus again we are led to seek a solution of *Problem P*.

3. The Separability State of the Sets A and B

From the preceding section, we see that in all cases we are led to seek a solution of *Problem P.* Note that the objective function of *Problem P* measures:

- the width of the dead zone when the sets A and B are separable, and
- the negative of the width of the ambiguous zone when the sets are not separable.

We can summarize this as follows:

THEOREM 2. *The optimal value V^* of Problem P is:*

- positive if and only if the sets A and B are strongly separable,
- zero if and only if the sets A and B are weakly separable, and
- negative if and only if the sets A and B are not separable.

We now look at a geometric interpretation of *Problem P*.

Let

$$F(w) = \min_{i=1,\ldots,p} \{A_i w\} \; - \max_{j=1,\ldots,q} \{B_j w\} \tag{4}$$

which is just the objective function of *Problem P*, ignoring the constant (1/2). Note that F is piecewise linear and concave, and that $F(0) = 0$. Note also that $F(\lambda w) = \lambda F(w)$ whenever $\lambda \geq 0$. The level sets

$$L(F; \sigma) = \{w : F(w) \geq \sigma\}$$

are convex polyhedra and contain the origin $w = 0$ when σ is non-positive. Now when the sets A and B are separable, $V^* \geq 0$, there is some feasible point w^* such that $F(w^*) \geq 0$, which implies that $F(\lambda w^*) \geq 0$ for all $\lambda \geq 0$. This means that the level set $L(F; 0)$ is unbounded, which in turn implies that all non-empty level sets $L(F; \sigma)$ are unbounded (see, e.g., Panik [20]). When the sets A and B are weakly separable, the highest value of F over the feasible region $\|w\| = 1$ is zero, so that, while $L(F; 0)$ is non-empty and unbounded, all of the level sets $L(F; \sigma)$ are empty for $\sigma > 0$.

On the other hand, when the sets A and B are not separable, the highest value of F over the feasible region is negative. In this case, the level set $L(F; 0) = \{0\}$, which implies that all non-empty level sets of F are bounded. We summarize this information in the following theorem, and exhibit the three cases in Figure 3.

THEOREM 3. *The sets A and B are separable if and only if $L(F; 0)$ is unbounded. In this case, the sets A and B are:*

- strongly separable if and only if $L(F; \sigma)$ is non-empty for all $\sigma > 0$.
- weakly separable if and only if $L(F; \sigma)$ is empty for some $\sigma > 0$.

These figures also suggest other facts about the strongly separable case.

THEOREM 4. *When the sets A and B are strongly separable, the solution of Problem P is equivalent to the convex program which results by replacing the constraint $\|w\| = 1$ by the constraint $\|w\| \leq 1$.*

Proof. The maximum value of F over $\|w\| = 1$ is positive, so that the maximum value of F over $\|w\| \leq 1$ is positive. The solution of the latter problem cannot occur at an interior point, because $F(\lambda w) = \lambda F(w)$ whenever $\lambda \geq 0$. □

THEOREM 5. *When the sets A and B are strongly separable, the solution of Problem P is unique.*

Proof. Assume the opposite and pick two optimal points to *Problem P*. The optimal value of F is positive. The midpoint on the line segment joining the points gives F at least as high a value as the two points give to F, and because $F(\lambda w) = \lambda F(w)$ whenever $\lambda \geq 0$, one gets a higher value over $\|w\| = 1$ than the assumed solution value. □

In the weakly separable case, F is maximized over $\|w\| = 1$ at some point which gives a zero value to F, as does $w = 0$. Thus the counterpart of Theorem 4 above is also true for this case, but there are multiple points (at least an entire line segment) where $F(w) = F(0) = 0$.

THEOREM 6. *When the sets A and B are weakly separable, the solution of Problem P is equivalent to the convex program which results by replacing the constraint $\|w\| = 1$ by the constraint $\|w\| \leq 1$. In this case, the origin $w = 0$ is a solution as well as at least one point on the boundary of this constraint.*

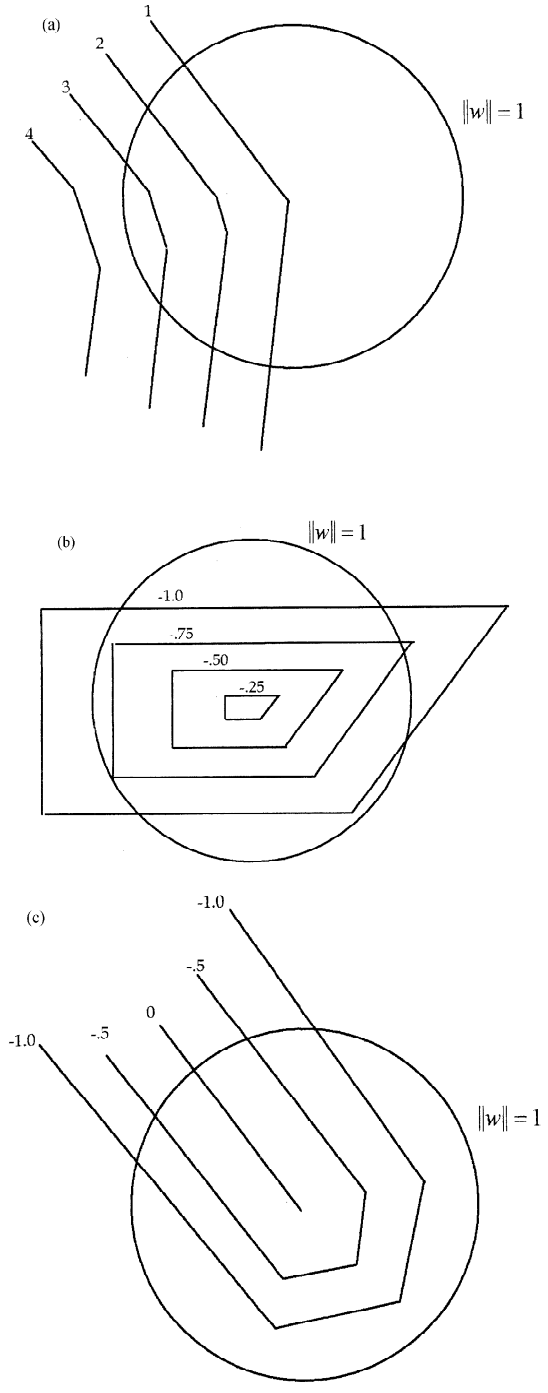

Figure 3. Level sets of F. (a): Unbounded level sets, A and B strongly separable; (b): bounded level sets A and B nonseparable; (c): unbounded level sets, A and B weakly separable.

Finally, in the non-separable case (when the level sets of F are bounded and $L(F,0) = \{0\}$), geometrically we are seeking a level set with a value as high as possible, which still intersects with the constraint $\|w\| = 1$. While the following theorem is computationally useless, it emphasizes the difficult nature of *Problem P* in the non-separable case.

THEOREM 7. *When the sets A and B are not separable, the solution of Problem P is equivalent to the non-convex program which results by replacing the constraint $\|w\| = 1$ by the constraint $\|w\| \geq 1$.*

Indeed, in this case, the unconstrained maximization of F over R^n is zero at $w = 0$.

To summarize, when the sets A and B are separable, a convex program can be solved to locate a solution, but in the weakly separable case one must be careful to select one of the multiple solutions on the boundary of the constraint region. In the non-separable case, we are forced to solve a non-convex program. Algorithms for these cases will be suggested in the next two sections. We conclude this section by identifying a linear program whose solution will reveal the nature of the separability of the sets A and B . It is the same LP that Mangasarian [15] had used earlier to produce a separating hyperplane in the strongly separable case.

First we note that the set $B = \{w : -1 \leq w_k \leq 1 \ (k = 1, \ldots, n)\}$ contains the set $S = \{w : \|w\| \leq 1\}$. If the sets A and B are strongly separable, the maximum of F over S and the maximum of F over B are both positive. Likewise if F is constantly zero over some ray emanating from $w = 0$ (the weakly separable case), the maximum of F over both B and S will be zero, with some point on the boundary of B giving a value of zero to F. Finally, if the sets A and B are not separable, F is maximized over both B and S at $w = 0$, and this is the unique maximizer of F.

Thus to determine the separability status of the sets A and B, we are led to seek the solution of the *Problem P^+* :

$$\text{Find } V^+ = \left(\frac{1}{2}\right) \max_{\|w\|_\infty = 1} \left\{ \min_{i=1,\ldots,p} \{A_i w\} - \max_{j=1,\ldots,q} \{B_j w\} \right\}. \tag{5}$$

which is equivalent to the linear program *Problem P^+* :

$$\text{Find } V^+ = \left\{ \begin{array}{ll} \max & s - t \\ s.t. & s \cdot e \leq Aw, \ t \cdot e \geq Bw, \ -e \leq w \leq e \end{array} \right. .$$

Note that the counterpart of Theorem 2, with P^+ replacing P and V^+ replacing V^* holds.

In the strongly separable case, the solution of this problem has a usefulness beyond the determination of the separability status of the sets A and B, as it is the first of a sequence of LP's whose solutions will ultimately yield the solution of *Problem P*.

4. An Algorithm for Solving Problem P - Strongly Separable Case

When the sets A and B are determined to be strongly separable (because the solution value V^+ of *Problem P*$^+$ is positive), we resort to Theorem 4 and seek the unique solution of the convex program:

$$\textit{Problem } P^{vex}: \text{ Find } V^* = (1/2) \cdot \max_{\|w\| \le 1} \{ \min_{i=1,\ldots,p} \{A_i w\} - \max_{j=1,\ldots,q} \{B_j w\} \} \tag{6}$$

which is equivalent to the problem:

$$\textit{Problem } P^{vex}: \text{ Find } 2V^* = \left\{ \begin{array}{ll} \max & s - t \\ s.t. & s \cdot e \le Aw, \ t \cdot e \ge Bw, \ \|w\| \le 1 \end{array} \right. . \tag{7}$$

This convex program had been previously addressed by Cavalier et al [3] who had suggested the following cutting plane method to handle the non-linear constraint $\|w\| \le 1$:

0) initialize: set $k = 0$, and solve *Problem P*^{vex} with the constraint $\|w\| \le 1$ relaxed to the linear constraints $S^0 = \{w : -e \le w \le e\}$. Note that this is precisely the *Problem P*$^+$ which was solved to determine the separability status of the sets A and B. Let w^0 denote a solution of this problem.

k) check for optimality: with $w^k \in S^k$ given, compute $\left\|w^k\right\|$. This will be ≥ 1. If it is equal to one, w^k is feasible to *Problem P* and we are done. Otherwise set $S^{k+1} = S^k \cup \{w : w \cdot (w^k / \left\|w^k\right\|) \le 1\}$ (i.e., add the 'cut' $w \cdot (w^k / \left\|w^k\right\|) \le 1$) and solve *Problem P*$^{k+1}$ with this new feasible region, and continue to iterate with $k = k + 1$.

This is a standard cutting plane procedure, with the cuts $w \cdot (w^k / \left\|w^k\right\|) \le 1$ continually added as needed to iteratively shave the initial box S^0 towards the desired feasible region of *Problem P*. Cavalier et al. [3] present the method as a 'heuristic' but, in fact, it can be shown to converge (in the strongly separable case) to the unique solution (see, [2]). Convergence, however, might not be finite. Figure 4 exhibits the level sets of F for the example when $A = \{(3, 2)\}$ and $B = \{(1, 1)\}$. In this example, the points w^k alternate among newly created vertices of the sets S^k, and approach the optimal solution $w^* = (2/\sqrt{5}, 1/\sqrt{5})$ $\gamma^* = 13/(2\sqrt{5})$ only in the limit.

We are thus motivated to seek an enhancement to this basic cutting plane algorithm. To that end, we note that the Karush-Kuhn-Tucker conditions for *Problem P* (now ignoring the constant (1/2) in the objective function) are:

$$\begin{array}{lll} \lambda A - \mu B = 2\nu w & \lambda e = 1 & \mu e = 1 \\ \lambda \ge 0 & Aw - se \ge 0 & \lambda(Aw - se) = 0 \\ \mu \ge 0 & Bw - te \le 0 & \mu(Bw - te) = 0 \\ \nu \ge 0 & \|w\|^2 \le 1 & v(\|w\|^2 - 1) = 0 \end{array} . \tag{8}$$

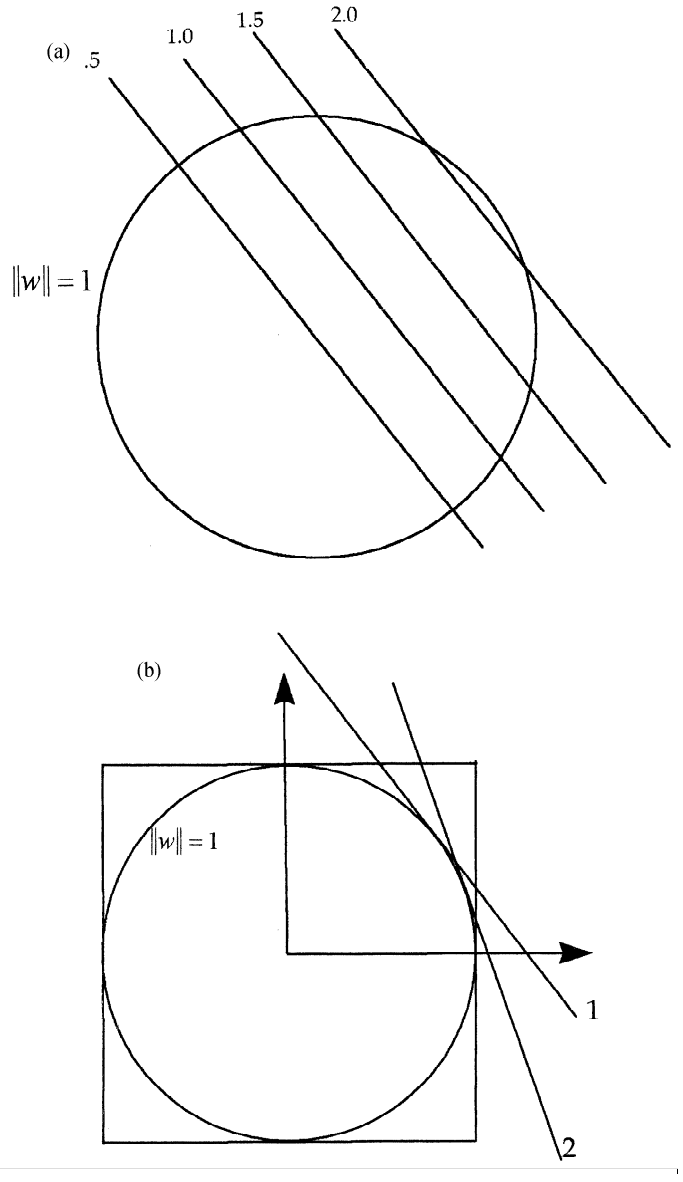

Figure 4. Level sets and cuts for the example. (a): Level sets; (b): two cuts for the example.

There are 12 expressions in Relations (8), arranged in the form of a 4 by 3 matrix. We will use the convention that expression (i, j) is the expression in row i, column j of Relations (8).

In the strongly separable case, no convex combination of the rows of A can equal a convex combination of the rows of B. From expressions $(1, 1)$ and $(4, 1)$

of Relations (8), we conclude that $\nu > 0$. This then, with expression $(4,3)$ implies that $\|w\|^2 = 1$. Also we know that $s - t > 0$ in the strongly separable case.

Rosen [22] had shown that generally at most $n+1$ of the constraints $(2,2)$ and $(3,2)$ of Relations (8) are binding, i.e., at most $n+1$ of the constraints

$$A_i w - s \geq 0 \ \ (i = 1, ..., p) \ \text{ and } \ B_j w - t \leq 0 \ \ (j = 1, \ldots, q)$$

occur as equalities at the optimal solution. In degenerate cases, one could have more than $n+1$ of these constraints binding, and the example illustrated in Figure 4 has less than $n+1$ constraints binding, but the usual case has exactly $n+1$ binding constraints. Assume first that exactly $n+1$ constraints are binding, and denote the binding constraints with a subscript b, i.e., assume that

$$A_b w - s e_b = 0 \ \text{ and } \ B_b w - t e_b = 0\,.$$

(As before, the notation e_b denotes a vector of ones of appropriate size.) These equations are homogenous so we can normalize a solution of them. Since we know that $s - t > 0$, we can assume that $s - t = 1$, and seek a solution of the matrix system:

$$\begin{pmatrix} A_b & -e_b & 0 \\ -B_b & 0 & e_b \\ 0 & 1 & -1 \end{pmatrix} \begin{pmatrix} w \\ s \\ t \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

representing $n+2$ equations in the $n+2$ unknowns w, s, t. If the system has no solution, we can simply revert to the usual cutting plane method discussed above. If this system has a solution (and it will if the matrix is non-singular), we can test it for optimality. To do this, we first normalize w, update the values s and t accordingly, and then determine the dual variables λ, μ, and ν. This can be done in the following way.

First multiply equation $(1,1)$ of Relations (8) by the normalized w and use relations $(1,2)$, $(1,3)$, $(2,3)$ and $(3,3)$ to get $s - t = 2\nu$. which determines ν. Now rewrite equations $(1,1)$, $(1,2)$, and $(1,3)$ of Relations (8) in the matrix form

$$(\ \lambda\ \ \mu\ \ 0\ ) \begin{pmatrix} A_b & -e_b & 0 \\ -B_b & 0 & e_b \\ 0 & 1 & -1 \end{pmatrix} = (\ 2\nu w\ \ -1\ \ 1\ )$$

and note that the inverse of the matrix had been determined earlier. Thus all of the relevant quantities have been determined, and the optimality conditions can be checked. If they are satisfied, we are done. Else add the cut as before, and continue.

When there are fewer than $n+1$ points of the sets A and B determining the current trial solution, the solution of the system (8) is somewhat more complicated. The equations $(1,1)$ in matrix form are

$$(\lambda\ \mu) \begin{pmatrix} A_b \\ -B_b \end{pmatrix} = 2\nu w$$

(note that we are not distinguishing between row and column vectors with a transpose, but by context, so here the vector w must be a row vector) where the number of rows of the matrix is at most n. If the matrix has less than full row rank, we revert to the normal cutting plane procedure. Otherwise, we can multiply on the right by the matrix transposed, to get:

$$(\lambda\ \mu)\begin{pmatrix} A_b \\ -B_b \end{pmatrix}\begin{pmatrix} A_b^T & -B_b^T \end{pmatrix} = 2\nu\begin{pmatrix} wA_b^T & -wB_b^T \end{pmatrix}.$$

Let

$$\hat{A} = \begin{pmatrix} A_b \\ -B_b \end{pmatrix}\begin{pmatrix} A_b^T & -B_b^T \end{pmatrix}$$

and set $\bar{A} = \left(\hat{A}\right)^{-1}$ so that the above equations, expressions (2,2) and (3,2) of Relations (8) become

$$(\lambda\ \mu) = 2\nu\,(se_A\ - te_B)\,\bar{A} \tag{9}$$

where we are using the subscripts on the vectors e to indicate their size (i.e., e_A pertains to those constraints $A_i w \geq \gamma$ which are binding at the current solution w^k).

Now partition the matrix $\bar{A}$ in conformance with the vectors e_A and e_B. Thus we set

$$\bar{A} = \begin{pmatrix} \bar{A}_{11} & \bar{A}_{12} \\ \bar{A}_{21} & \bar{A}_{22} \end{pmatrix},\ E = \begin{pmatrix} e_A & 0 \\ 0 & e_B \end{pmatrix}$$

and let

$$\begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix} = E\begin{pmatrix} \bar{A}_{11} & \bar{A}_{12} \\ \bar{A}_{21} & \bar{A}_{22} \end{pmatrix}E^T.$$

Note that c_{ij} is simply the sum of all entries in the matrix $\bar{A}_{ij}$ and that $c_{ij} = c_{ji}$. Now multiply Equation (9) on the right by the vector $\begin{pmatrix} e_A^T \\ 0 \end{pmatrix}$ and use the expression (1,2) to get

$$1 = 2\nu\,(sc_{11} - tc_{12}).$$

Similarly, multiply Equation (9) on the right by the vector $\begin{pmatrix} 0 \\ e_B^T \end{pmatrix}$ and expression (1,3) of Relation (8) to get

$$1 = 2\nu\,(sc_{12} - tc_{22}).$$

These last two equations imply that

$$\begin{pmatrix} s \\ t \end{pmatrix} = \begin{pmatrix} c_{12} - c_{22} \\ c_{11} - c_{12} \end{pmatrix}\alpha \tag{10}$$

for some α.

At the same time, using the facts that $\|w\| = 1, \nu > 0$, expression (1,1) of Relations (8) and Equation (9), we get the quadratic equation

$$s^2 c_{11} - 2stc_{12} + t^2 c_{22} = 1$$

which, in view of (10) above, reduces to a quadratic equation in α. While there are generally two roots of this equation, we choose the one which gives the larger value to the quantity $s - t$. With these values of s and t, we get $\nu = (s - t)/2$ and through Equation (9), the vectors λ and μ. Finally, through expression (1,1), we get w.

As before, if the KKT conditions are satisfied, we are done and otherwise we resort to the standard cutting plane procedure.

The process is finite, as eventually w^k must identify the correct binding constraints. As we will point out in Section 6, this enhancement dramatically improved convergence on a set of randomly generated problems.

While each subproblem solved is an LP (and is therefore solvable in polynomial time by any one of several interior point methods), we cannot claim that *Problem P* is also polynomial solvable by our enhanced cutting plane method.

5. Algorithms for Solving Problem P – Weakly Separable and Non-separable Cases

When the optimal value of *Problem* P^+ is zero, we know that the sets A and B are not strongly separable. Indeed, depending on the computer code used to solve *Problem* P^+, we may even know more. In the weakly separable case, the vector $w = 0$ is a solution, as well as at least one point on the boundary of the constraints $-1 \leq w_k \leq 1\ (k = 1, \ldots, n)$. If the code happens to produce a solution point other than $w = 0$, we know the sets are weakly separable and the normalized solution produced is a solution to *Problem P*. This is not to be expected, however.

If the code indicates that the solution point $w = 0$ is unique, we know that the sets are not separable. Some codes (such as GAUSS) offer the possibility of producing other solutions when multiple optimal solutions are indicated. However, we cannot depend on this for two reasons: a) many codes do not allow a user to request an alternate solution if one exists, and b) even codes which do allow a user to request an alternate solution are not necessarily reliable. The reason for the latter is simple — the indication for multiple optimal solutions is a zero reduced cost on a non-basic variable. That means the variable can be made basic without changing the value of the objective function. However, in degenerate cases, such a change in basis might not produce a different solution if the incoming variable cannot be brought in at a positive value. This is especially true for our problem, as the null solution will be exhibited by an entirely degenerate optimal tableau, so that a pivot in any nonbasic column will only result in a change of basis and not in a new optimal solution. Thus we need to look for a more reliable way of producing alternate solutions if, indeed, any exist.

To that end, assume $V^+ = 0$ and $w = 0$ are the optimal outputs of a computer implementation of *Problem P*. Then either the sets A and B are weakly separable, or they are not separable. But by Theorem 3, this is equivalent to determining if the level sets $L(F;\sigma)$ are unbounded for $\sigma \leq 0$ or not. We can set $\sigma = -1$, and solve the 2n problems:

For $k = 1, \ldots, n$, solve:

$$\textit{Find } u_k = \begin{cases} \max & w_k \\ s.t. & F(w) \geq -1 \end{cases}$$

$$\textit{Find } l_k = \begin{cases} \min & w_k \\ s.t. & F(w) \geq -1 \end{cases}$$

or, equivalently, for $k = 1, \ldots, n$, solve the linear programs:

$$\textit{Find } u_k = \begin{cases} \max & w_k \\ s.t. & se \leq Aw \\ & te \geq Bw \\ & s - t \geq -1 \end{cases}$$

$$\textit{Find } l_k = \begin{cases} \min & w_k \\ s.t. & se \leq Aw \\ & te \geq Bw \\ & s - t \geq -1 \end{cases}$$

If any of these problems is unbounded, the sets A and B are weakly separable, and a weakly separating hyperplane can be constructed by setting the unbounded variable equal to an appropriate constant (e.g., +1 if $w_k = \infty$ and -1 if $l_k = -\infty$), resolving P^+ with the new constraints and setting $s - t = 0$.

If all of these problems are bounded, then we will have: a) determined that the sets A and B are not separable, and b) constructed vectors u (of upper bounds) and l (of lower bounds) such that $\{w : F(w) \leq -1\} \subset \{w : l_k \leq w_k \leq u_k\}$. These bounds are necessary to begin the solution of *Problem P* in the nonseparable case.

In the nonseparable case, we will need to solve a nonconvex program. We intend to solve this by applying the Branch and Bound algorithm introduced by Falk and Soland [4]. This procedure requires a separable objective function and (for a practical implementation) linear constraints. As it stands, *Problem P* is not of this form. However, the following result allows us to address an equivalent problem. The result may be motivated by examining Figure 6, wherein it is clear that finding the level set $L(F;\sigma)$ with the highest value of σ which intersects the feasible region of *Problem P* defined by the equation $\|w\| = 1$, is equivalent to finding the largest value of the function $G(w) = \|w\|^2$ over some non-empty level set of F, say $L(F;-1)$.

Define

$$\textit{Problem Q:} \begin{cases} \max & G(w) = \|w\|^2 \\ s.t. & F(w) \geq -1 \end{cases}$$

i.e.,

$$\textit{Problem Q:} \begin{cases} \max & G(w) = \|w\|^2 \\ s.t. & se \leq Aw \\ & te \geq Bw \\ & s - t \geq -1 \end{cases}$$

Note that the feasible region here is known to be a nonempty bounded polyhedral set contained within the hyper-rectangle defined by the vectors of lower and upper bounds l and u.

THEOREM 8. *Any solution of Problem Q, when normalized, is a solution of Problem P.*

Proof. *Problem Q* involves the maximization of a continuous function over a compact set, and so has a solution – let w^q denote such a point (it need not be unique). This point is not zero, as we have eliminated the weakly separable case. Then its normalized point w^Q is feasible to *Problem P*. Let w^P be a global solution of *Problem P*. We know that $F(w^P) < 0$ because the sets are not separable. Therefore,

$$\begin{aligned} \left\| w^P/(-F(w^P)) \right\| &= -1/(F(w^P)), \text{ and} \\ F(w^P/(-F(w^P)) &= F(w^P)/(-F(w^P)) = -1 \end{aligned}$$

so that the point $w^P/(-F(w^P))$ is a nonzero feasible solution of *Problem Q*. □

Since w^q is a solution to *Problem Q*, $F(w^q) \geq -1$ and so

$$\begin{aligned} \|w^q\| &\geq \left\| w^p/(-F(w^P) \right\| \\ &= 1/(-F(w^P)) \\ &> 0. \end{aligned}$$

This implies that

$$\|w^q\| F(w^P) \leq -1 \tag{11}$$

At the same time, since w^P is a solution to *Problem P*,

$$\begin{aligned} F(w^P) &\geq F(w^q/\|w^q\|) \\ &= F(w^q)/\|w^q\| \end{aligned}$$

from which we see that

$$\begin{aligned} \|w^q\| F(w^P) &\geq F(w^q) \\ &\geq -1. \end{aligned} \tag{12}$$

Taking (11) and (12) together, we get

$$\|w^q\| F(w^P) = -1. \tag{13}$$

Now if the normalized point $w^q / \|w^q\|$ were not a solution of *Problem P*, we would have $F(w^P) > F(w^q / \|w^q\|)$ which implies $\|w^q\| F(w^P) > F(w^q) \geq -1$, in contradiction to (11).

The objective function of *Problem Q* is separable, and the constraints define a bounded linear polytope. Finite upper and lower bounds u and l describing a hyper-rectangle enclosing the feasible region had been determined above, so that we can now apply the Falk-Soland Algorithm [4]. This is a Branch and Bound method that sets up and solves a finite sequence of linear programs whose solutions ultimately will produce a global solution w^* to *Problem P*. We summarize the algorithm here.

Problem P^1 is defined as follows. For each term w_k^2 in the objective function, we obtain its concave envelope $h_k^1(w_k)$ over the interval $[l_k, u_k]$. This is easily determined to be the linear function $h_k^1(w_k) = (l_k + u_k) \cdot w_k - l_k u_k$ (see Figure 5). Then define

$$\textit{Problem } P^1\textit{:} \begin{cases} \max & H^1(w) = \sum_{k=1}^n h_k^1(w_k) \\ s.t. & se \leq Aw \\ & te \geq Bw \\ & s - t \geq -1. \end{cases}$$

Let w^1 denote an optimal solution of this problem and let ub^1 denote the optimal value.

Since $H^1(w)$ overestimates the objective function $\|w\|^2$ of *Problem Q* over their common feasible region, the number ub^1 must be an upper bound on the optimal value of *Problem Q*. Moreover, since w^1 is feasible to *Problem Q* the number $lb^1 = \|w^1\|^2$ must server as a lower bound on the optimal value of *Problem Q*. In the event that $lb^1 = ub^1$ we are done and w^1 is globally optimal for *Problem Q*. In general however, $lb^1 < ub^1$ and we must continue.

We start an ordered list call it 'LIST'. The items on the list are solved linear programs, ordered according to the optimal values of the LP's with the largest values at the top of the list. The feasible region of each problem on LIST contains a potentially global solution of *Problem Q*. Initially the only problem on LIST is *Problem* P^1. Each problem P^t on LIST is characterized by:

- an interval I^t consisting of lower and upper bounds on the variables w_k, where initially $I^1 = [l, u]$ - the lower and upper bounds determined at the beginning of this section. The *Problem* P^t is similar to *Problem* P^1 except that

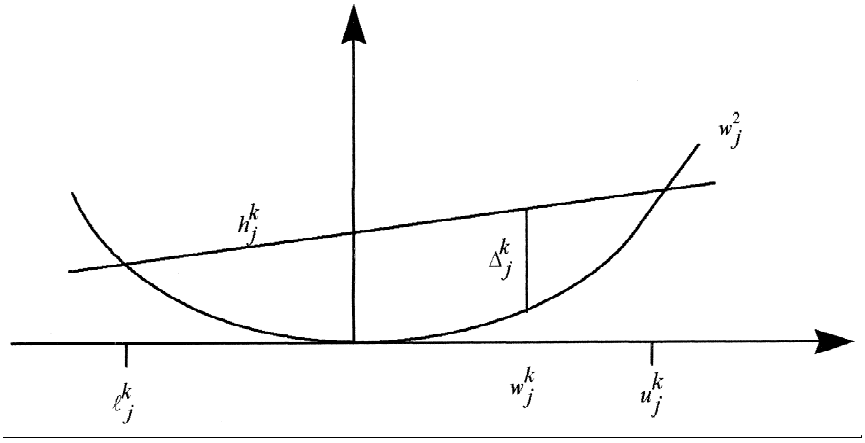

Figure 5. Convex envelope of $\mathcal{W}_k^2$.

the feasible region of P^t includes the additional constraints $w \in I^t$, and the objective function of *Problem* P^t has the form

$$H^t(w) = \sum_{k=1}^{n} h_k^t(w_k)$$

where each function h_k^t is the concave envelope of w_k^2 taken over the interval $[l_k^t, u_k^t]$,

- the solution w^t of *Problem* P^t,
- an upper bound ub^t equal to the optimal value of the linear program *Problem* P^t,and
- a lower bound lb^t equal to $\|w^t\|^2$.

All items on LIST will have $lb^t < ub^t$ indicating that the feasible region of *Problem* P^t does contain a feasible point with (original) objective function value $lb^t = \|w^t\|^2$, and may contain a feasible point with an objective function value as high as ub^t.

Along with LIST, there will be an 'incumbent' solution w^{inc} with an objective function value val^{inc} corresponding to the most promising of all solutions found to date. Initially, $w^{inc} = w^1$ and $val^{inc} = lb^1$. All items kept on LIST will have $ub^t > val^{inc}$ indicating that the feasible regions of the problems on LIST may contain feasible points with higher values than the best one found so far.

The algorithm proceeds in stages, with stage one consisting of the solution of *Problem* P^1. With stage T complete, we set up stage $T+1$ by selecting (and removing from the list) the problem at the top of the list (the '*parent problem*') – call it *Problem* P^t - and creating two new problems (the '*offspring problems*'). The parent problem has the largest objective function value of all problems still on the list, and so corresponds to the 'most promising' problem on the list. To create the

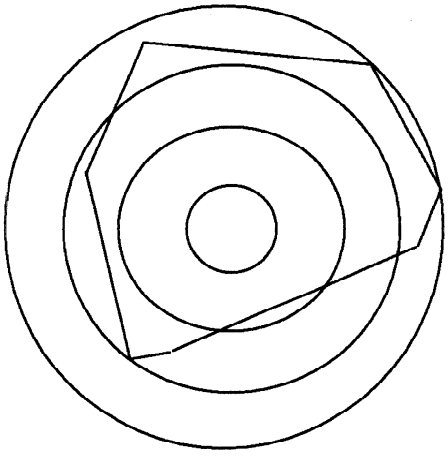

Figure 6. Level sets of F – Example.

offspring problems, we first determine a '*branching variable*'. Since $lb^t < ub^t$, we must have

$$\left(w_k^t\right)^2 < h_k^t(w_k^t)$$

for at least one k. For the branching variable, we choose any index $k(t)$ for which this difference is maximal, and create the two offspring problems $P^{t(1)}$ and $P^{t(2)}$ by modifying the defining intervals $I^{t(1)}$ and $I^{t(2)}$. Both of these intervals are the same as the parent interval I^t except in the component $k(t)$, where the interval $[l_{k(t)}^t, u_{k(t)}^t]$ of the parent is replaced by the interval $[l_{k(t)}^t, w_{k(t)}^t]$ for the first offspring problem, and by the interval $[w_{k(t)}^t, u_{k(t)}^t]$ for the second offspring problem.

The objective functions of the offspring problems are modified in variable $k(t)$ only, by computing new concave envelopes over the new intervals. We then solve the new problems (with the new intervals added to the constraints of the parent problem) to obtain solutions $w^{t(1)}$ and $w^{t(2)}$ with new objective function values $ub^{t(1)}$ and $ub^{t(2)}$ and new lower bounds $lb^{t(1)}$ and $lb^{t(2)}$ on the global optimal value of the original objective function. Clearly

$$ub^t \geq \max\{ub^{t(1)}, ub^{t(2)}\}.$$

We now update the incumbent solution if a more promising point has been found, i.e., if

$$\max\{lb^{t(1)}, lb^{t(2)}\} > val^{inc}.$$

If, in fact, the incumbent is modified, all problems currently on LIST with inferior upper bounds (i.e., $ub^r \leq val^{inc}$) are now deleted from LIST as their feasible regions cannot contain any points better than the current incumbent point.

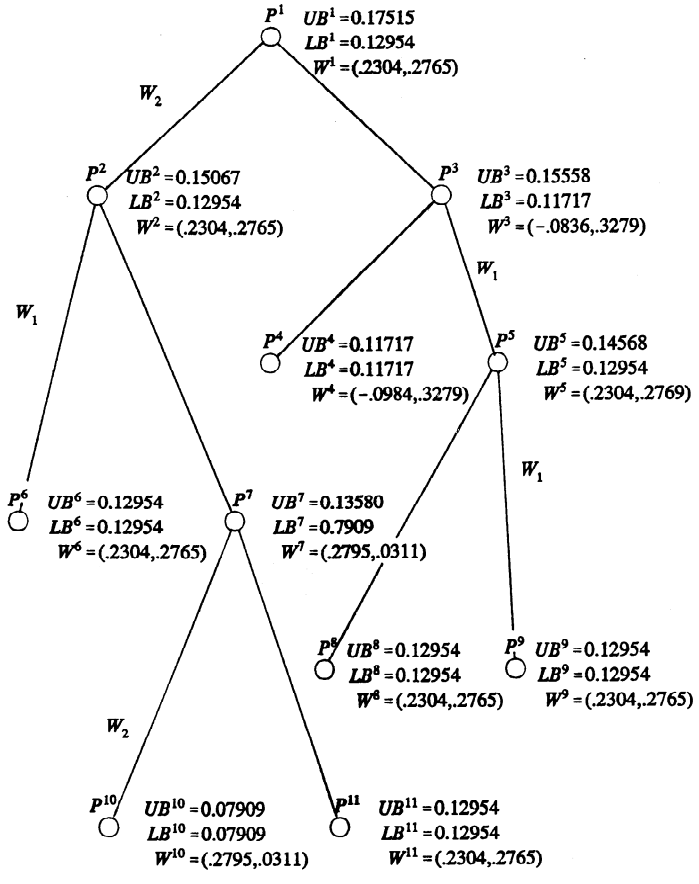

Figure 7. Branch and bound tree – Example.

In any event, we now consider adding the newly solved offspring problems to LIST. In particular, *Problem* $P^{t(i)}$ is added to LIST only if $ub^{t(i)} > val^{inc}$ (the feasible region of *Problem* $P^{t(i)}$ may contain a point superior to the current incumbent).

If LIST is now empty, we are done - the incumbent solution is the global solution of *Problem Q*. Otherwise we continue as described earlier.

The process is finite (see Falk and Soland [4]). However, as it is basically a non-convex problem apparantly requiring some variation of a Branch and Bound approach, it is almost certainly NP-hard.

To illustrate the method, the following (badly) non-separable example was constructed. The level sets of F are shown in Figure 6 and the Branch and Bound Tree is exhibited in Figure 7. Note that the optimal solution is actually found as the solution of *Problem* P^1 but not recognized as such until after eleven LP's are solved. Note that there are proper local solutions in this example.

Example 1:

$$A = \begin{pmatrix} 0.5 & 1.5 \\ 3.5 & -1.0 \\ 5.0 & 1.0 \\ 2.0 & 1.0 \\ 0.0 & 6.0 \end{pmatrix} \text{ and } B = \begin{pmatrix} 4.0 & 2.2 \\ 0.0 & 1.0 \\ 4.0 & -2.0 \end{pmatrix}.$$

6. Computational Results

The algorithms described in Section 4 and 5 were implemented using GAUSS, the linear programming package provided therein, and ran on a 486/33N personal computer. Two types of computational results are presented, one randomly generated and the other taken from an established database. In this section we address the randomly generated problems.

To test the efficiency of the methodology for the strongly separable case, we first generated a random hyperplane $H(w,\gamma)$ in R^n, and a random m by n matrix M (the integers m and n are inputs of the user). All random numbers were realized by a random number generator from a uniform distribution over [0,1). The matrix M was then devided into two matrices A and B where the rows A_i of A are those rows of M that satisfy $A_i w \geq \gamma$. In the (unlikely) event that A or B had no rows, we would have disregarded the data, but this never happened.

To evaluate the performance of the algorithm, we generated 50 problems for each of 32 (m,n) pairs. Each problem had n variables, and the same total number $m\,(=p+q)$ of points. For each problem, we counted the number of linear programs solved with and without the KKT enhancement, and recorded the average number of LP's solved. A tolerance of 10^{-6} between successive iterates w^k was used as a convergence criterion for the unenhanced version. We increased n in increments of 3 from 3 to 12, and m in increments of 5 and 10 from 5 points to 50 points. The results are tabulated in Table I.

The top entry in each cell corresponds to the average number of LP's required for the unenhanced version, and the bottom entry corresponds to the enhanced version. The case $n = 12$ and $m = 5$ is not displayed for the unenhanced version as it was requiring an excessive amount of computing time. Indeed the averages for the case $n = 9, m = 5$ is only based on 15 problems for the same reason.

We note the following from Table I:

- For a fixed dimension n, problems with a low number of points were somewhat harder to solve. This is probably because fewer than $n + 1$ points were determining the separating hyperplane, resulting in a number of poor predictions of binding constraints by the trial iterates w^k.
- The KKT enhancement generally cut down the total number of required LP's by at least a factor of 3, and as much as a factor of 41. The most dramatic

Table I. Average No. LP's Required, Without and With Enhancement

n\(p+q)	5	10	15	20	25	30	40	50
3	11.22 1.42	5.58 1.32	4.08 1.16	3.56 1.08	3.54 1.14	3.82 1.12	3.12 1.00	3.02 1.02
6	37.30 2.32	17.42 3.42	10.44 2.42	7.90 1.98	5.86 1.56	5.20 1.66	4.54 1.32	4.12 1.24
9	99.20 2.40	47.50 6.12	28.68 5.32	16.70 4.04	12.94 3.36	8.90 2.68	7.56 2.22	5.44 1.92
12	2.26	95.10 5.86	52.20 7.74	35.74 7.62	26.48 6.42	18.92 5.48	12.16 4.04	9.04 2.98

Table II. Ratio of Entries of Table I

n\(p+q)	5	10	15	20	25	30	40	50
3	7.90	4.23	3.52	3.30	3.10	3.41	3.12	2.96
6	16.08	5.09	4.31	3.99	3.76	3.13	3.44	3.32
9	41.33	7.76	5.39	4.13	3.85	3.32	3.40	2.83
12		16.23	6.74	4.69	4.12	3.45	3.01	3.03

improvements were found where there were relatively few points and a relatively high dimension. The effect of the enhancement generally decreased as $m = p + q$ increased for fixed n. Indeed, the ratios of the average number of LP's required for the two versions seemed to be decreasing to something below 3.0, prompting the question (posed by an astute referee) as to whether there is a limit greater than 1. Table II exhibits these ratios.

- For a fixed dimension n, the average number of LP's needed to solve the examples generally *decreased* as the number of points increased, both for the enhanced and the unenhanced versions. This is probably due to the fact that more data points would generally guarantee that exactly $n + 1$ points would determine the optimum.

The behavior of the algorithm for the nonseparable case is more difficult to evaluate as Branch and Bound is an implicit enumeration method and one would expect wide variations of results for randomly generated problems of the same size. We would, however, expect that the method would perform well on problems which were 'nearly' separable and also that the width of the ambiguous strip would decrease as the degree of inseparability decreased. To test these conjectures, we took a pair of nonseparable sets and gradually 'separated' them by adding constant increments to one of the sets until the translated set separated from the other. Consider the example of Section 5 (Example 2).

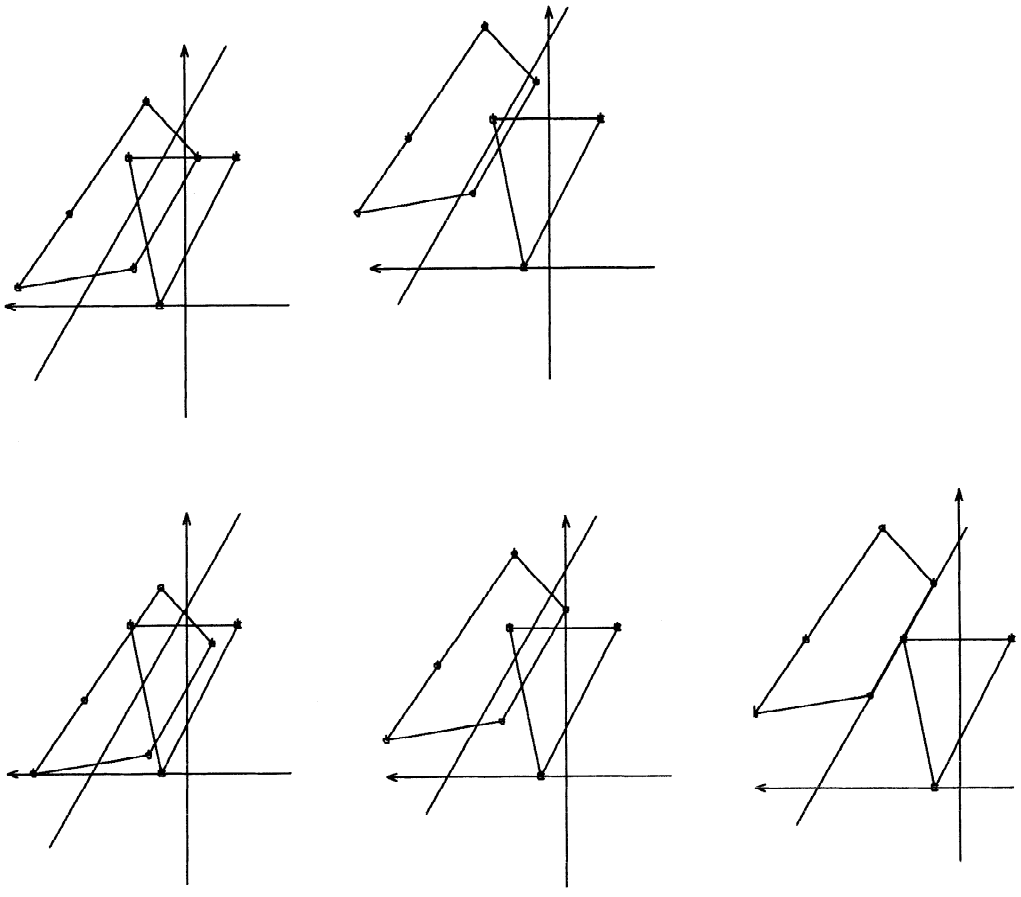

Figure 8. Nonseparable A and B, and four translations of A.

Example 2:

$$A = \begin{pmatrix} 0.5 & 1.5 \\ 3.5 & -1.0 \\ 5.0 & 1.0 \\ 2.0 & 1.0 \\ 0.0 & 6.0 \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} 4.0 & 2.2 \\ 0.0 & 1.0 \\ 4.0 & -2.0 \end{pmatrix}.$$

The solution of this problem took 11 problems (6 stages) as is illustrated in Figure 7. The example is badly nonseparable as indicated in Figure 8a.

Now begin to increase each entry of A by multiples of 0.5. This amounts to shifting the set A to the northeast direction. Eventually the shifted A will be separated from B. Figure 8 exhibits this shifting. As expected, both the number of LP's required, as well as the width of the ambiguous strip eventually decreased as A and B tended more to separate, with the former decreasing from 11 to 3 to 1

and 1, and the latter decreasing from 2.7784 to 2.0742 to 1.3700 to 0.6658 before becoming separable.

In the next example we generated two matrices A and B (again from U(0,1)) of 10 rows and 4 columns each, and increased the entries of A successively in increments of 0.1. The actual matrices generated were:

$$A = \begin{pmatrix} .77416 & .57563 & .88177 & .26804 \\ .78414 & .48796 & .97274 & .28300 \\ .60458 & .44884 & .34724 & .60052 \\ .01751 & .08588 & .55000 & .84631 \\ .40625 & .54202 & .66676 & .48591 \\ .83884 & .96729 & .83151 & .05376 \\ .84716 & .93729 & .83446 & .50611 \\ .10919 & .73131 & .89678 & .04628 \\ .89560 & .18773 & .22152 & .06302 \\ .99138 & .63610 & .48169 & .16690 \end{pmatrix}$$

and

$$B = \begin{pmatrix} .44535 & .89996 & .81787 & .61715 \\ .23264 & .58236 & .74107 & .31872 \\ .41808 & .33384 & .32166 & .76504 \\ .35443 & .48940 & .51111 & .24722 \\ .00331 & .77013 & .91097 & .40096 \\ .43340 & .26992 & .41054 & .79773 \\ .79571 & .28602 & .86998 & .87883 \\ .85539 & .31589 & .60069 & .26846 \\ .56903 & .24379 & .77924 & .89591 \\ .44381 & .19153 & .45194 & .62555 \end{pmatrix}$$

For the matrix A with a translation of 0.1, the method took 75 LP's with an ambiguous strip of 0.33926. With a translation of 0.2, it took 15 LP's, with an ambiguous strip of 0.22822. With a translation of 0.3 it took 3 LP's with an ambiguous strip of 0.04054. With a translation of 0.4, the sets became strongly separable. Figure 9 is a graph of the number of LP's required as plotted vs. the size of the ambiguous strip for the same example with smaller increments of 0.05.

In all of these examples, as with others that we ran, it is clear that the Branch and Bound Algorithm is most effective when the sets A and B are close to being separable. What is not exhibited (but not surprising either) is that the method generally found the correct hyperplane in the first subproblem P^1, with the other LP's required to confirm global optimality.

7. The Wisconsin Breast Cancer Database

The previous section addressed random problems of small and moderate size in order to get some idea of the amount of extra work (in terms of number of LP's)

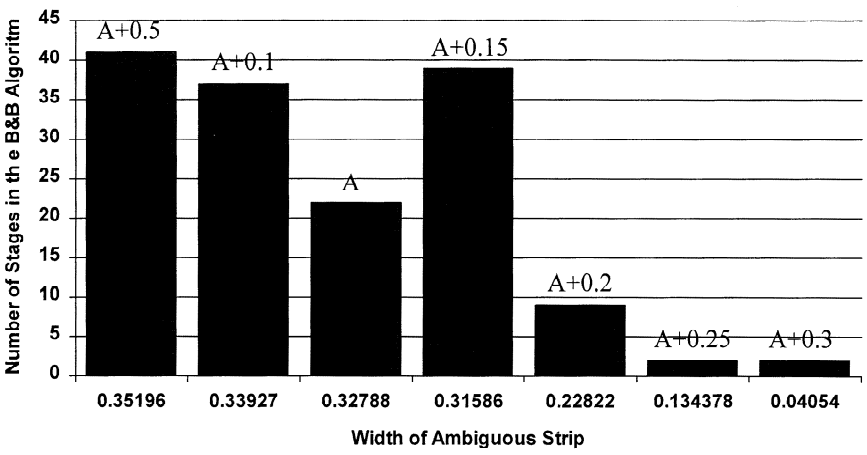

Figure 9. Effect of Translations on A.

Table III. Description of the Wisconsin Breast Cancer Database

Group	# of Points	# Benign	# Malignant	Separable?	Date entered
1	353	188	165	no	01/89
2	69	56	13	strong	10/89
3	31	22	9	strong	02/90
4	17	14	3	strong	04/90
5	48	36	12	strong	08/90
6	49	40	9	strong	01/91
7	31	17	14	strong	06/91
8	85	71	14	strong	11/91
3,4,5	96	72	24	strong	1990
6,7,8	165	128	37	strong	1991
3,4,5,6,7,8	261	200	61	strong	90-91
2,3,4,5,6,7,8	330	256	74	no	89-91

that is required to solve the model. In this section, we focus on a 'real world' set of data in order to get some idea of the accuracy of the model. We choose to use the Wisconsin Breast Cancer Database [17] which is available through the internet at http://www.ics.uci.edu/AI/ML/MLDBRepository.html.

The data in the Wisconsin Breast Cancer Database was added sequentially, and it is convenient to refer to the group number corresponding to when it was added. Each data point consisted of 9 attributes and each component is an integer between 1 and 10.

We are interested in seeing how well the hyperplanes generated by our model would do on this data set. While the criterion driving the models addressed herein attempts to maximize the width of the dead zone in the separable case, and minimize

Table IV. Prediction Accuracy with the Wisconsin Breast Cancer Database

Training Group/# points	# LP's	Testing Group/# points	Accuracy
100 distinct points of Group 1 / 100	151	{1,2,...,8}˜Training Set / 583	93.96%
2,3,4,5,6,7,8 / 330	195	1 / 353	91.22%
3,4,5 / 96	1	6,7,8 / 165	97.58%
3,4,5 / 96	1	6 / 49	95.92%
3,4,5,6 / 145	1	7 / 31	100%
3,4,5,6,7 / 176	3	8 / 85	97.65%
2,3,4,5 / 165	4	6 / 49	95.92%
6,7,8 / 165	12	1,2,3,4,5 / 518	94.79%
2,3,4,5 / 165	4	6,7,8 / 165	96.97%
3,4,5 / 96	1	1,2 / 422	94.76%

the width of the ambiguous strip in the non-separable case, we can, nevertheless, measure the 'efficiency' of our method by computing *the percentage of correctly classified points*. It turns out that the hyperplane produced by our method does, in fact, separate quite well using this criteria*.

We first applied our method to each of the 8 subgroups of Table III to identify its 'state of separability'. Group 1 is the only group which is not separable, all other groups are strongly separable. The separability status of the unions of some individual subgroups was also noted.

Using the first 100 distinct points of group 1 as a training set, we found them to be non-separable. Using the Branch and Bound method of Section 5, we obtained a hyperplane which minimized the width of the ambiguous strip of those 100 points. Using the remaining database as the test set, we attained an accuracy of 93.967%. This is reported in the top row of Table IV.

We (rather arbitrarily) selected various other combinations of subgroups to act as the training set, and others as the test set. The results of these tests are tabulated in Table IV.

For example, we used the 1990 data (of sets 3, 4, and 5) to predict the nature of the 1991 data (given by sets 6, 7, and 8) and found that the accuracy of prediction was 97.58% (row 3 of Table IV). Various other combinations are displayed in Table IV.

Note that the number of LP's required to solve the problems was as high as 195 – this is not surprising as groups 2 through 8 form a non-separable set and the problem is non-convex.

All of the above tests were performed with the GAUSS program running on a 486 Versa NEC docking station with 20 megabytes of memory and a 340 MB hard disk drive.

* It might be noted that a recent paper addresses this very objective function [18], and describres a heuristic method for the solution of the resulting non-convex program.

More testing (including comparative testing) should clearly take place, with other data and in tandem with other methods. However, the above results indicate that the models addressed in this paper are producing accuracies which are quite reasonable.

8. Summary

We began the paper by writing down an optimization problem whose solution would produce the separating hyperplane associated with a dead zone of maximal width when A and B are separable, and a separating hyperplane associated with an ambiguous region of minimal width when A and B are not separable. It turns out that the appropriate optimization model is exactly the same in any case, but the nature of the problem is quite different:

$$\begin{aligned} A \text{ and } B \text{ are separable} &\iff \text{the appropriate model is a convex program} \\ A \text{ and } B \text{ are not separable} &\iff \text{the appropriate model is a non-convex program.} \end{aligned}$$

A geometric interpretation of the models was offered in terms of the level sets of the objective function of the model.

In any case, the models had been previously posed by a number of authors.

In the strongly separable case, we pointed out that a previously suggested 'heuristic' cutting plane method is, in fact, guaranteed to converge. But we were able to improve convergence of the method by addressing the optimality conditions and attempting to satisfy them on the way to optimality.

In the non-separable case, we introduced a Branch and Bound method to globally solve the non-convex program.

Finally, we offered some computational evidence that

- the computational enhancement for convergence in the strongly separable case is useful,
- the solution of the model in the non-separable case is do-able (albeit with the high computational expense normally associated with Branch and Bound), and
- in either case, the models were able to forecast with a reasonable accuracy on a real-world data set.

References

1. K.P. Bennett and O.L. Mangasarian, Robust linear programming discrimination of two linearly inseparable sets. *Optimization Methods and Software* **19**(2) (1992), 23–24.
2. J.E. Blankenship and J.E. Falk, Infinitely constrained optimization problems. *Journal of Optimization Theory and Applications* **19** (1976), 261–281.
3. T.M. Cavalier, J.P. Ignizio, and A.L. Soyster, Discriminant analysis via mathematical programming: Certain problems and their causes. *Computers and Operations Research* **16**(4), (1989), 353–362.

4. J.E. Falk and R.M. Soland. An algorithm for separable nonconvex programming problems. *Management Science* **15** (1969), 550–569.
5. N. Freed and F. Glover, A linear programming approach to the discriminant problem. *Decisions Sciences* **12** (1981), 68–74.
6. N. Freed and F. Glover, Evaluating alternative linear programming models to solve the two-group discriminant problem. *Decisions Sciences* **17** (1986a), 151–162.
7. N. Freed and F. Glover, Resolving certain difficulties and improving the classification power of lp discriminant analysis formulations. *Decisions Sciences* **17** (1986b), 589–595.
8. L.W. Glorfeld and N. Gaither, On using linear programming in discriminant problems. *Decision Sciences* **13** (1982), 167–182.
9. F. Glover, Improved linear programming for discriminant analysis. *Decision Sciences* **21** (1990), 771–785.
10. F. Glover, S. Keene, and B. Duea. A new class of models for the discriminant problem. *Decision Sciences* **19** (1988), 269–280.
11. G.J. Koehler, Characterization of unacceptable solutions in lp discriminant analysis. *Decision Sciences* **20** (1989a), 239–257.
12. G.J. Koehler, Unacceptable solutions and the hybrid discriminant model. *Decision Sciences* **20** (1989b), 844–848.
13. P.F. Lambert, *Methodologies of Pattern Recognition*. Academic Press, New York, 1969.
14. E. Lopez-Cardona, *Surgical Separation of Finite Sets*. PhD thesis, The George Washington University, Washington, DC, 1994.
15. O.L. Mangasarian, Linear and nonlinear separation of patterns by linear programming. *Operations Research* **13** (1965), 444–452.
16. O.L. Mangasarian, R. Setiono and W.H. Wolberg, Pattern recognition via linear programming. In: T.F. Coleman and Y. Li (eds.), *Large-Scale Numerical Optimization*, pp. 22–30. SIAM, Philadelphia, 1990.
17. O.L. Mangasarian and W.H. Wolberg, Cancer diagnosis via linear programming. *SIAM NEWS* **23**(5), 1990.
18. M.G. Marcotte, P. and G. Savard, A new implicit enumeration scheme for the discriminant analysis problem. *Computers and Operations Research* **22**(6) (1995), 625–639.
19. W.H. Marlow, *Mathematics for Operations Research*. Wiley, New York, 1978.
20. M.J. Panik, *Fundamentals of Convex Analysis – Duality, Separation, Representation, and Resolution*. Kluwer, Amsterdam, 1993.
21. F.P. Preparata, *Computational Geometry, An Introduction*. Springer-Verlag, Paris, 1990.
22. J.B. Rosen, Pattern separation by convex programming, *Journal of Mathematical Analysis and Application* **10** (1965), 123–134.
23. F.W. Smith, Pattern classification design by linear programming. *IEEE Transactions on Computers, C-17*, **4** (1968), 367–372.
24. J. Ullman, *Pattern Recognition Techniques*. Crane, London, 1973.