A Parametric Optimization Method for Machine Learning

Article *in* Informs Journal on Computing · March 1995

DOI: 10.1287/ijoc.9.3.311 · Source: CiteSeer

2 authors:

Kristin P. Bennett
Rensselaer Polytechnic Institute
186 PUBLICATIONS **9,271** CITATIONS

SEE PROFILE

Erin J. Bredensteiner
University of Evansville
6 PUBLICATIONS **850** CITATIONS

SEE PROFILE

A Parametric Optimization Method for Machine Learning

Kristin P. Bennett
Erin J. Bredensteiner *

R.P.I. Math Report No. 217

January 31, 1995

Abstract

The classification problem of constructing a plane to separate the members of two sets can be formulated as a parametric bilinear program. This approach was originally created to minimize the number of points misclassified. However, a novel interpretation of the algorithm is that the subproblems represent alternative error functions of the misclassified points. Each subproblem identifies a specified number of outliers and minimizes the magnitude of the errors on the remaining points. A tuning set is used to select the best result amoung the subproblems. A parametric Frank-Wolfe method was used to solve the bilinear subproblems. Computational results on a number of datasets indicate that the results compare very favorably with linear programming and simulated annealing approaches. The algorithm can be used as part of a decision tree algorithm to create nonlinear classifiers.

1 Introduction

A fundamental problem in machine learning is the discrimination between the elements of two sets $\mathcal{A}$ and $\mathcal{B}$ in the n-dimensional real space R^n. In the simplest case, a linear function consisting of a linear combination of the input attributes can be used to separate the two sets. The linear function determines a separating plane. In practice, it is uncommon for the two given sets to be strictly linearly separable. Thus, it is important to find the linear function that discriminates best between the two sets according to some error minimization criterion. The linear function found can be utilized in a decision tree to attain nonlinear separation. In a decision tree, nonlinear separation can be achieved by recursively applying several linear functions or decisions that partition R^n into disjoint regions, each corresponding to set $\mathcal{A}$ or set $\mathcal{B}$. The goal is to find the linear separator that generalizes best, i.e. correctly classifies future points. For example, an approximate separating plane

*Department of Mathematical Sciences, Rensselaer Polytechnic Institute, Troy, NY 12180. Email bennek@rpi.edu, bredee@rpi.edu. This material is based on research supported by National Science Foundation Grant 949427.

can be such that it minimizes the distances of the misclassified points from the separating plane [BM92]. In misclassification minimization the problem is to minimize the number of misclassified points. For a given problem, different error functions may result in better (or worse) separators in terms of generalization. In Figure 1, the plane obtained by minimizing the distances of the misclassified points from the separating plane (Plane 1) is misleading. There is no clear underlying separation of the sets along Plane 1. Plane 2 was formed by minimizing the number of points misclassified. There is a clear division along Plane 2. In practice, the choice of error functions is not always clear. Thus we propose a hybrid approach (parametric misclassification minimization) that identifies the outliers and minimizes the distances of the remaining misclassified points. This parametric approach includes the linear program that minimizes the average misclassification error as a subproblem [BM92].

This research investigates mathematical programming methods for constructing decisions in decision trees. Classical decision tree algorithms such as CART [BFOS84] and ID3 [Qui84] use exhaustive search to find decisions based on a single input attribute. When the decisions are multivariate linear functions of the input attributes, exhaustive search is no longer feasible. Linear programming and perceptron algorithms have been used to construct decisions that minimize the distances of the misclassified points from the separating plane. Linear programming approaches [BM92, Ben92, Glo90] find optimal decisions by this criterion in polynomial time. Decision tree methods based on heuristic variants of perceptron algorithms [Utg89, BU92] have worked well in practice, but the algorithms may fail to converge and may not find optimal solutions. The problem of creating a linear function that minimizes the number of points misclassified is NP-complete [Hea92]. The algorithms CSADT [HKS93] and OC1 [MKSB93, MKS94] use simulated annealing to minimize functions of the number of misclassified points. Using mathematical programming, we have developed an algorithm that combines the two error criteria and exploits the mathematical structure of the underlying problem in order to find better solutions.

In Section 2, we investigate the parametric bilinear programming formulation of the misclassification minimization program, first proposed by Mangasarian [Man94]. We discuss a novel interpretation of the suboptimal solutions as an alternative error criterion. In Section 3, we propose an algorithm based on the Frank-Wolfe method discussed in [BM93] for solving the parametric bilinear programming problem. This algorithm is attractive because half of the subproblems have closed form solutions. Computational results on a number of practical problems are given in Section 4.

The following notation is used. Let $\mathcal{A}$ and $\mathcal{B}$ be two sets of points in the n-dimensional real space R^n with cardinality m and k respectively. Let A be a $m \times n$ matrix whose rows are the points in $\mathcal{A}$. Let B be a $k \times n$ matrix whose rows are the points in $\mathcal{B}$. The i^{th} point in $\mathcal{A}$ and the i^{th} row of A are both denoted A_i. Likewise, B_j is the j^{th} point in $\mathcal{B}$ and the j^{th} row in B. For two vectors in R^n, xy denotes the dot product. The set of minimizers of $f(x)$ on the set $\mathcal{S}$ is denoted by $arg\min_{x \in \mathcal{S}} f(x)$. For a vector x in R^n, x_+ will denote the vector in R^n with components $(x_+)_i := max\{x_i, 0\},\ i = 1, \ldots, n$. The step function x_* will denote the vector in $[0,1]^n$ with components $(x_*)_i := 0$ if $(x_*)_i \leq 0$ and $(x_*)_i := 1$ if $(x_*)_i > 0,\ i = 1, \ldots, n$.

2 Misclassification Minimization

In this section, we investigate the misclassification minimization problem [Man94] which minimizes the number of points misclassified by a plane and discuss its benefits and limitations. The primary limitations are that the problem is NP-complete [Hea92] and has infinitely many local minima [Man94]. Thus the problem may require extensive computational time. The parametric bilinear

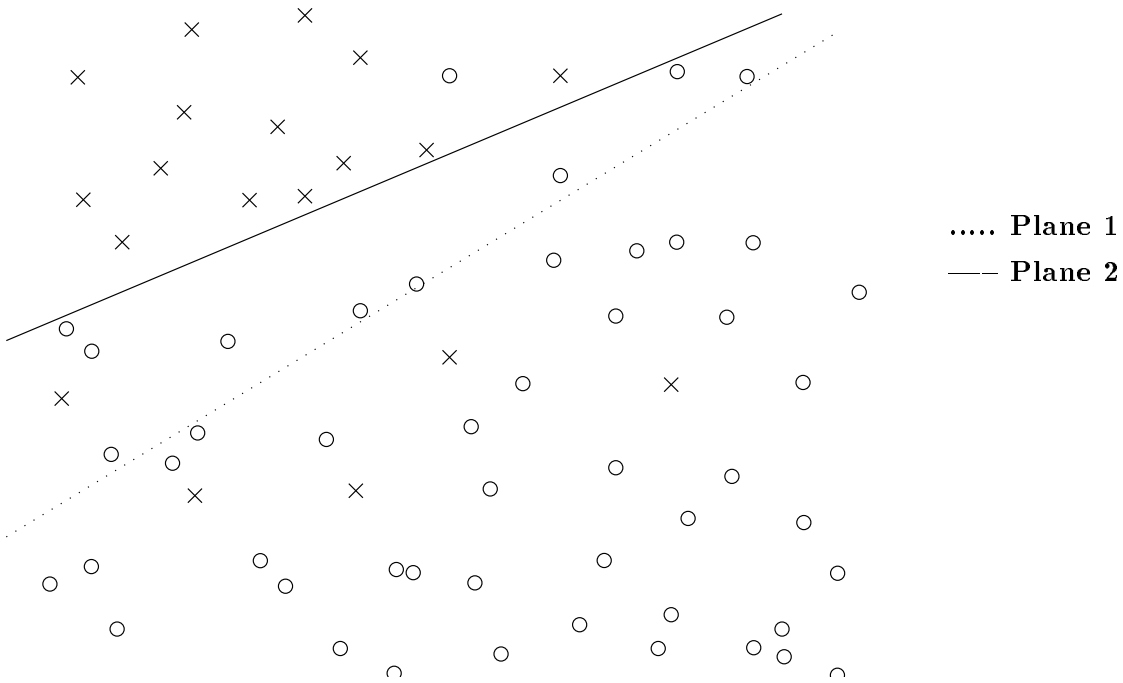

Figure 1: Planes were found by minimizing two different error functions. Plane 1 was found by minimizing the distance of the misclassified points from the plane. Plane 2 was found by minimizing the number of points misclassified.

programming formulation of the misclassification minimization program requires the solution of a series of subproblems. The subproblems produce several alternate solutions based on related but different error criteria. This is an attractive property since it is not known *a priori* which criterion will produce the plane that generalizes best for a given dataset. We begin with the definition of linear separability.

Definition 2.1 (Linear Separability) *Given two sets of points $\mathcal{A}$ and $\mathcal{B}$, the two sets are linearly separable if there exists some plane*

$$wx = \gamma \tag{1}$$

where $w \in R^n$ is the normal to the plane and γ determines the distance of the plane from the origin, such that

$$Aw > e\gamma \qquad e\gamma > Bw \tag{2}$$

where e is a vector of ones of the appropriate dimension. This becomes (upon normalization)

$$Aw - e\gamma - e \geq 0 \qquad -Bw + e\gamma - e \geq 0 \tag{3}$$

Let the step function be defined as $(\zeta)_*$:R$\longrightarrow$R as

$$(\zeta)_* = \begin{cases} 1 & \text{if } \zeta > 0 \\ 0 & \text{if } \zeta \leq 0 \end{cases}$$

Then system (3), the definition of linear separability, is equivalent to

$$e(-Aw + e\gamma + e)_* + e(Bw - e\gamma + e)_* = 0 \tag{4}$$

For the case where $\mathcal{A}$ and $\mathcal{B}$ are not linearly separable, the problem becomes one of minimizing the left-hand side of (4), which merely counts the number of points misclassified by the current plane (w, γ). As shown by Mangasarian [Man94] this problem can be reformulated into a linear program with equilibrium constraints:

$$\begin{array}{lrr} \min\limits_{w,\gamma,r,u,s,v} & er + es & \\ & u + Aw - e\gamma - e \geq 0 & v - Bw + e\gamma - e \geq 0 \\ & r \geq 0 & s \geq 0 \\ & r(u + Aw - e\gamma - e) = 0 & s(v - Bw + e\gamma - e) = 0 \\ & -r + e \geq 0 & -s + e \geq 0 \\ & u \geq 0 & v \geq 0 \\ & u(-r + e) = 0 & v(-s + e) = 0 \end{array} \tag{5}$$

It has been demonstrated [Man94] that this linear program has a stationary point for almost every (w, γ). To avoid this problem we consider the following equivalent parametric bilinear program:

Find the non-negative integer $\bar{\delta}$ such that

$$\bar{\delta} = \min_{\delta \geq 0} \{\delta | f(\delta) = 0\} \tag{6}$$

where $f(\delta) =$

$$\begin{array}{ll} \min\limits_{w,\gamma,r,u,s,v} & \frac{1}{m}[r(u + Aw - e\gamma - e) + u(-r + e)] + \\ & \frac{1}{k}[s(v - Bw + e\gamma - e) + v(-s + e)] \\ & u + Aw - e\gamma - e \geq 0 \quad v - Bw + e\gamma - e \geq 0 \\ & u \geq 0 \qquad\qquad\qquad v \geq 0 \\ & 0 \leq r \leq e \qquad\qquad 0 \leq s \leq e \\ & er + es \leq \delta \qquad\qquad \delta \in [0, \infty) \end{array} \tag{7}$$

The parametric bilinear formulation (6) represents a combination of the two error functions mentioned previously: the distance of the misclassified points from the separating plane and the number of points misclassified. When $\delta = 0$ the subproblem (7) is precisely the following linear program [BM92] which minimizes the average magnitude of the misclassification errors, i.e.

$$\begin{array}{ll} \min\limits_{w,\gamma,u,v} & \frac{1}{m}eu + \frac{1}{k}ev \\ \textit{subject to} & u + Aw - e\gamma - e \geq 0 \\ & v - Bw + e\gamma - e \geq 0 \\ & u \geq 0, \quad v \geq 0 \end{array} \tag{8}$$

The parametric bilinear problem requires the solution of a series of subproblems for different values of δ. We will now show that each subproblem identifies at most δ "outliers", or difficult to classify points, and minimizes the average magnitude of the misclassification errors on the remaining points. The optimal $\bar{\delta}$ is exactly the fewest number of outliers removed such that the remaining points of the two classes are linearly separable. We begin by showing that at optimality the variables u and v measure the error in terms of the distance of the misclassified points from the optimal plane $xw = \gamma$. This is the error metric used in LP (8).

Theorem 2.1 (Characterization of optimal solution) *Let $\bar{w}, \bar{\gamma}, \bar{r}, \bar{u}, \bar{s}, \bar{v}$ be an optimal solution of the bilinear subproblem (7) for a given value of δ. Then*

$$\begin{array}{rcl} \bar{u} & = & (-A\bar{w} + e\bar{\gamma} + e)_+ \\ \bar{v} & = & (B\bar{w} - e\bar{\gamma} + e)_+ \end{array} \tag{9}$$

Proof. Let $\bar{l}^i \geq 0$, $i = 1 \ldots 9$ be the optimal dual variables for problem (7). The optimal $\bar{w}, \bar{\gamma}, \bar{r}, \bar{u}, \bar{s}, \bar{v}$ are feasible and satisfy the remaining optimality conditions [Lue84]:

$$\begin{array}{ll} l^1(u + Aw - e\gamma - e) = 0 & \\ l^2(v - Bw + e\gamma - e) = 0 & \\ l^3(u) = 0 & l^4(v) = 0 \\ l^5(r - e) = 0 & l^6(s - e) = 0 \\ l^7(r) = 0 & l^8(s) = 0 \\ l^9(er + es - \delta) = 0 & \\ \frac{1}{m}(rA) - \frac{1}{k}(sB) - l^1A + l^2B = 0 & \\ \frac{1}{m}(-re) + \frac{1}{k}(se) + l^1e - l^2e = 0 & \\ \frac{1}{m}e - l^1 - l^3 = 0 & \\ \frac{1}{k}e - l^2 - l^4 = 0 & \\ \frac{1}{m}(Aw - e\gamma - e) + l^5 - l^7 + el^9 = 0 & \\ \frac{1}{k}(-Bw + e\gamma - e) + l^6 - l^8 + el^9 = 0 & \end{array} \tag{10}$$

For $\bar{u}_i,\ i = 1, \ldots, m$, if $\bar{u}_i > 0$ then

$$0 = \bar{l}_i^3 = \frac{1}{m} - \bar{l}_i^1 \quad \Rightarrow \quad \bar{l}_i^1 = \frac{1}{m}$$

By complementarity $(\bar{u}_i + A_i\bar{w} - \bar{\gamma} - 1) = 0$. Thus $\bar{u}_i = (-A_i\bar{w} + \bar{\gamma} + 1)_+$. If $\bar{u}_i = 0$ then $A_i\bar{w} - \bar{\gamma} - 1 \geq 0$. Hence $\bar{u}_i = (-A_i\bar{w} + \bar{\gamma} + 1)_+ = 0$. The same argument holds for $\bar{v}_j,\ j = 1, \ldots, k$ and points in set $\mathcal{B}$. □

For each bilinear subproblem (7) with $\delta \leq \bar{\delta}$, there exist optimal r and s such that exactly δ of the components of r and s are equal to 1 and the remaining components are equal to 0. This is proved in the next theorem.

Theorem 2.2 (Existence of an integer solution for r and s.) *Let $\bar{w}, \bar{\gamma}, \bar{u}, \bar{v}$ be an optimal solution of the bilinear subproblem (7) for a given integer value of δ, $0 < \delta \leq \bar{\delta}$, then there exists $\bar{r} \in [0,1]^m$ and $\bar{s} \in [0,1]^k$ such that $\bar{w}, \bar{\gamma}, \bar{r}, \bar{u}, \bar{s}, \bar{v}$ is an optimal solution of $f(\delta)$ with $e\bar{r} + e\bar{s} = \delta$.*

Proof. We know by Theorem 2.1 that $\bar{u} = (-A\bar{w} + e\bar{\gamma} + e)_+$ and $\bar{v} = (B\bar{w} - e\bar{\gamma} + e)_+$. For $i = 1, \ldots, m$ if $\bar{u}_i = 0$ then $\bar{r}_i = 0$ is always feasible and optimal. If $\bar{u}_i > 0$ then $\bar{u}_i + A_i\bar{w} - \bar{\gamma} - 1 = 0$. Similarly for $j = 1, \ldots, k$, if $\bar{v}_j = 0$ then $\bar{s}_j = 0$ is optimal, and if $\bar{v}_j > 0$ then $\bar{v}_j - B_j\bar{w} + \bar{\gamma} - 1 = 0$. Thus for fixed $\bar{w}, \bar{\gamma}, \bar{u}, \bar{v}$ the problem simplifies to

$$\begin{array}{ll} \min\limits_{r,s} & \frac{1}{m}\bar{u}(-r + e) + \frac{1}{k}\bar{v}(-s + e) \\ \textit{subject to} & er + es \leq \delta \\ & 0 \leq r \leq e \quad 0 \leq s \leq \delta \end{array} \quad . \tag{11}$$

Let $Q = \left\{ \bar{u}_i \text{ and } \bar{v}_j |\ i \text{ and } j \text{ with } \delta \text{ largest values of } \frac{1}{m}\bar{u}_i,\ i = 1, \ldots, m, \text{ and } \frac{1}{k}\bar{v}_j,\ j = 1, \ldots, k \right\}$. The optimal solution of (11) is $r_i = 1$ for $\bar{u}_i \in Q$, $r_i = 0$ otherwise, and $s_j = 1$ for $\bar{v}_j \in Q$, $s_j = 0$ otherwise. □

We now show that for $\delta \leq \bar{\delta}$ the nonzero components of r and s identify the δ outlying points in classes $\mathcal{A}$ and $\mathcal{B}$. The plane $xw = \gamma$ is chosen to minimize the average distance of the remaining misclassified points from the plane.

Theorem 2.3 (Optimality of alternative error functions) *Let $\bar{w}, \bar{\gamma}, \bar{r}, \bar{u}, \bar{s}, \bar{v}$ with dual variables $\bar{l}^i,\ i = 1, \ldots, 9$ be an optimal solution of $f(\delta)$, for $\delta \leq \bar{\delta}$, and $\bar{r}, \bar{s}$ are binary vectors. Let*

$\hat{A}$ *be the matrix formed from A by removing the rows A_i where $\bar{r}_i = 1,\ i = 1, \ldots, m$*

$\hat{B}$ *be the matrix formed from B by removing the rows B_j where $\bar{s}_j = 1,\ j = 1, \ldots, k$*

$\hat{u}$ *be the vector formed from $\bar{u}$ by removing the components $\bar{u}_i$ where $\bar{r}_i = 1,\ i = 1, \ldots, m$*

$\hat{v}$ *be the vector formed from $\bar{v}$ by removing the components $\bar{v}_j$ where $\bar{s}_j = 1,\ j = 1, \ldots, k$*

then $\bar{w},\ \bar{\gamma},\ \hat{u},\ \hat{v}$ are the optimal solutions of the LP

$$\begin{array}{ll} \min\limits_{w,\gamma,u,v} & \frac{1}{m}eu + \frac{1}{k}ev \\ \textit{subject to} & u + \hat{A}w - e\gamma - e \geq 0 \\ & v - \hat{B}w + e\gamma - e \geq 0 \\ & u \geq 0, \quad v \geq 0 \end{array} \tag{12}$$

Proof. Consider the optimality conditions [Lue84] of the bilinear program (7) and LP (12). Clearly the primal constraints of LP (12) are satisfied since they are a subset of the constraints of (7). The dual variables $\bar{l}^i \geq 0,\ i = 1, \ldots, m$ satisfy the optimality conditions given in (10). Let $\hat{l}^1$ be the vector formed from $\bar{l}^1$ by removing the components $\bar{l}^1_i$ where $\bar{r}_i = 1, i = 1, \ldots, m$ and let $\hat{l}^2$ be the vector formed from $\bar{l}^2$ by removing the components $\bar{l}^2_j$ where $\bar{s}_j = 1, j = 1, \ldots, k$. Note that $\hat{l}^1 \geq 0$ and $\hat{l}^2 \geq 0$. Complementary slackness for LP (12) holds since $\bar{l}^1(\bar{u} + A\bar{w} - e\bar{\gamma} - e) = 0$ and $\bar{l}^2(\bar{v} - B\bar{w} + e\bar{\gamma} - e) = 0$. Thus all that remains to be shown is $\hat{l}^1\hat{A} = \hat{l}^2\hat{B}$ and $\hat{l}^1 e = \hat{l}^2 e$. As shown in Theorem 2.2, if $\bar{r}_i = 1$ then $\bar{u}_i > 0$. Hence

$$0 = \bar{l}^3_i = \frac{1}{m} - \bar{l}^1_i \quad \Rightarrow \quad \bar{l}^1_i = \frac{1}{m} \quad \Rightarrow \quad (\frac{1}{m}\bar{r}_i - \bar{l}^1_i) = 0.$$

If $\bar{s}_j = 1$ then $\bar{v}_j > 0$ and $\bar{l}^2_j = \frac{1}{k} \ \Rightarrow (\frac{1}{k}\bar{s}_j - \bar{l}^2_j) = 0$. Thus $(\frac{1}{m}\bar{r} - \bar{l}^1)A = (\frac{1}{k}\bar{s} - \bar{l}^2)B \ \Rightarrow \ \hat{l}^1\hat{A} = \hat{l}^2\hat{B}$. Similarly $(\frac{1}{m}\bar{r} - \bar{l}^1)e = (\frac{1}{k}\bar{s} - \bar{l}^2)e \ \Rightarrow \ \hat{l}^1 e = \hat{l}^2 e$. □

3 Misclassification Minimization Algorithms

In this section, we provide the MISMIN and MISMIN-P algorithms. We begin by describing the common parts of both approaches. First subproblem (7) is solved for $\delta = 0$. This corresponds to solving the LP (8) to find the plane that minimizes the average distance of misclassified points to the plane. For $\delta > 0$, a Frank-Wolfe type algorithm [BM93] is used to solve bilinear subproblem (7). This algorithm has the beneficial property that it decomposes the problem into two linear programs one of which has a closed form integer solution. Note that for $\delta > 0$ the bilinear subproblem is nonconvex and may have local minima. By careful choice of the sequence of δ and by using the solution of one subproblem as the starting point for the next subproblem, we found better solutions and reduced the computation time. The secant method proposed but not implemented in [Man94] was used to select the values of δ.

MISMIN-P is a parametric misclassification minimization algorithm that selects the best bilinear subproblem (7) solved within the MISMIN algorithm by using a tuning set. The plane that performs best on a reserved set of points is selected as the final plane. This is not necessarily the plane corresponding to the lowest value of δ such that $f(\delta) = 0$. Details of all the algorithms are given below.

3.1 Bilinear Subproblems

The parametric bilinear programming formulation (7) is an uncoupled bilinear program. The Frank-Wolfe algorithm applied to an uncoupled bilinear program will converge to a global solution or a stationary point [BM93]. Applying this Frank-Wolfe algorithm to problem (7) we obtain the following algorithm:

Algorithm 3.1 (Frank-Wolfe algorithm for uncoupled bilinear programs) *For fixed δ, Step 1:* $(w^{i+1}, \gamma^{i+1}, u^{i+1}, v^{i+1}) \in$

$$\begin{array}{lll} arg\min\limits_{w,\gamma,u,v} & \frac{1}{m}[eu + r^i(Aw - e\gamma - e)] + \frac{1}{k}[ev + s^i(-Bw + e\gamma - e)] & \\ & u + Aw - e\gamma - e \geq 0 & v - Bw + e\gamma - e \geq 0 \\ & u \geq 0 & v \geq 0 \end{array}$$

Step 2: $(r^{i+1}, s^{i+1}) \in$

$$arg\min_{r,s} \quad -\tfrac{1}{m} r u^{i+1} - \tfrac{1}{k} s v^{i+1}$$
$$0 \le r \le e \quad 0 \le s \le e \quad er + es \le \delta$$

Step 3: Repeat until no improvement in objective.

The subproblem contained in step 2 has a closed form integer solution as shown in Theorem 2.2.

3.2 The MISMIN Bilinear Program

The parametric programming algorithm (6) searches for the smallest integer value of δ such that $f(\delta) = 0$. Since the subproblem for a given δ may have local minima, the choices of δ affect the final solution. One ordering of δ values may drive the solution to a local minimum, while another ordering will find the global optimizer. For each choice of δ several linear programs may need to be solved, thus making a large number of guesses at the true value of $\bar{\delta}$ results in computational inefficiency. Figure 2 demonstrates how the function value, $f(\delta)$, in the cancer problem (described in the computational results section) changes as δ increases. This figure shows that a variation of the secant method will be an excellent approximator for $\delta = \bar{\delta}$. Thus in practice we employ the following algorithm:

Algorithm 3.2 (Misclassification Minimization) *Let δ_{max} denote the fewest number of points misclassified by any plane at a given time. Let δ_{min} denote the largest δ value* **attempted** *so far in Algorithm 3.1 such that $f(\delta) > 0$.*

Step 0: Let $\delta = 0$ and solve bilinear subproblem (7) using Algorithm 3.1.
Let δ_{max}= the number of points misclassified by the new plane.
Let $\delta = \frac{2}{3}\delta_{max}$.

Step 1: Solve bilinear subproblem (7) using Algorithm 3.1.

Step 2: Let δ_{max} = minimum of δ_{max} and number of points misclassified by the current plane.

Step 3: If $f(\delta) = 0$
then let $\delta = \frac{1}{2}(\delta_{min} + \delta_{max})$
else calculate secant method update

$$p = \delta - f(\delta)\frac{(\delta - \delta_{min})}{(f(\delta) - f(\delta_{min}))}$$

Let $\delta_{min} = \delta$
If $p \in (\delta_{min}, \delta_{max})$
then let $\delta = p$
else let $\delta = \frac{1}{2}(\delta_{min} + \delta_{max})$

Step 4: If $\delta_{max} > \delta_{min} + 1$ Go to Step 1.

3.3 Parametric Misclassification

As shown in Theorem 2.3, each of the optimal solutions of the bilinear subproblem where $f(\delta) > 0$ corresponds to minimizing the problem using an error function that selects δ outliers and minimizes

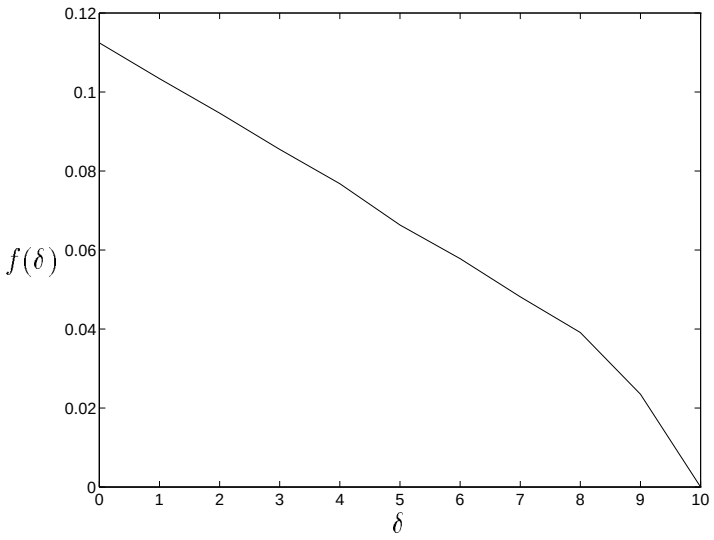

Figure 2: Function values for varying values of δ on the Wisconsin Breast Cancer Data

the average distance from the separating plane of the remaining misclassified points. We would like to select the plane that minimizes the classification error on future points. To estimate the accuracy of the plane on future points we use a tuning set. Points are reserved from the training data and not used in the bilinear programs. These points are used to evaluate the optimal planes found by solving subproblem (7) for each value of δ. The plane that minimizes the testing set error is returned as the best solution. Note that both the plane that minimizes the number of points misclassified and the plane that minimizes average magnitude of the errors are candidate solutions. So MISMIN-P should do at least as well as MISMIN and LP (8) alone.

4 Computational Results

In this section, we present results of computational experiments performed using MISMIN-P on four real world data sets: Cleveland Heart Disease Database [DJS+89], Wisconsin Breast Cancer Database [WM90] and Star/Galaxy Dim and Bright data sets [OSP+92]. MISMIN was implemented in AMPL [FGK93], a mathematical programming software package, utilizing the CPLEX 3.0 [CPL94] solver. We present results for LP (8), MISMIN, and MISMIN-P. Computational results were tabulated for each of these choices. For MISMIN-P we used the testing set for the tuning set in order to see what the best answer would be. In practice the tuning set must not include the testing data, so this should be regarded as an optimistic estimate.

OC1 [MKSB93], a simulated annealing algorithm, was applied to make comparisons on the dependability of the MISMIN results. OC1 is an algorithm that generates multivariate decision trees based on deterministic and randomized procedures. The results obtained for OC1 represent the accuracy of the root hyperplane of the decision tree constructed by this algorithm. The defaults were chosen for all parameters in OC1 except no pruning portion was used and 50 iterations were chosen (the value used in [MKSB93]). In correspondence with misclassification minimization the sum minority impurity measure was applied.

Training and testing set accuracies were measured for each dataset. The training and testing

	Cancer		Heart		Star/Galaxy Dim		Star/Galaxy Bright	
	Training	Testing	Training	Testing	Training	Testing	Training	Testing
MISMIN	98.5	96.3	90.1	81.1	96.5	95.5	99.9	99.2
OC1	97.6	95.6	84.1	73.8	91.5	91.2	98.9	98.5
LP	97.7	97.2	85.1	83.5	95.7	95.6	99.7	99.3
MISMIN-P	98.2	97.5	87.7	84.5	96.1	95.9	99.9	99.4

Table 1: Average Accuracy (%)

set accuracies were calculated using 10-fold cross validation. In training 9/10 of the points are used to find the hyperplane. Then in testing the remaining 1/10 of the points are used to test how well the hyperplane generalizes on unseen points. This process is repeated 10 times, one for each 1/10 partition, to establish an average testing and training set accuracy. The same partitions were used for all four methods.

Cleveland Heart Disease Database The Cleveland Heart Disease Database has 297 patients listed with 13 numeric attributes. Each patient is classified as to whether there is presence or absence of heart disease. There are 137 patients who have a presence of heart disease. This data set is available via anonymous file transfer protocol (ftp) from the University of California Irvine UCI Repository Of Machine Learning Databases [MA92].

Wisconsin Breast Cancer Database This data set is used to classify a set of 682 patients with breast cancer. Each patient is represented by nine integral attributes ranging in value from 1 to 10. The two classes represented are benign and malignant: 442 of the patients are benign while 240 are malignant. This data set is also available via anonymous ftp from the University of California Irvine UCI Repository Of Machine Learning Databases [MA92].

Star/Galaxy Databases The Star/Galaxy Database consists of two data sets: dim and bright. The dim data set has 4192 examples and the bright data set has 2462 examples. Each example represents a star or a galaxy and is described by 14 numeric attributes. The bright data set is nearly linearly separable, while the dim data set is significantly more difficult. These two data sets are generated from a large set of star and galaxy images collected by Odewahn [OSP+92] at the University of Minnesota.

Table 1 shows that MISMIN performs substantially better than OC1. Not all the differences are statistically significant, but the trends are clear. Both this version of the OC1 algorithm and MISMIN are optimizing the same error function: the number of misclassified points. MISMIN achieved greater training and testing set accuracies on all four datasets. This indicates that the simulated annealing algorithm used in OC1 is prematurely stopping at local minima. MISMIN found superior solutions. It is possible that through adjustment of the many paramenters, OC1 could obtain better solutions. One advantage of MISMIN is that no such parameter adjustment is needed. As expected, the training set accuracy of the LP (8) was lower than that of MISMIN because the LP minimizes the magnitudes of the misclassification errors instead of the number of points misclassified. But surprisingly the LP had better training set accuracy than OC1. In addition, the testing set accuracy of the LP was higher than both OC1 and MISMIN. MISMIN-P, the parametric error approach, provides further improvement over the LP results. This shows that *a priori* we do not know the error function most appropriate for a given data set.

Data	Method	Average Accuracy (%)	Average # of Decisions
Cancer	MISMIN-P	97.5	1.0
	OC1	97.4	2.4
Star/Galaxy (Dim)	MISMIN-P	95.9	1.0
	OC1	95.8	36.0
Star/Galaxy (Bright)	MISMIN-P	99.4	1.0
	OC1	99.2	15.6

Table 2: Comparison of MISMIN-P with best reported OC1 results

Additional testing is needed to explore the choices of error functions and optimization algorithms for decision tree construction. The very promising results in Table 1 are for a single plane. OC1 is designed to construct decision trees with many decisions. In [MKS94, MKSB93] OC1 was found to construct simpler trees that generalized better than those of other univariate decison tree approaches. In Table 2, we compare the best results reported in [MKSB93] for OC1 with those for MISMIN-P. Once again 10-fold cross validation was used to estimate the the average testing set accuracy and number of decisions used. There is no significant difference in the testing set accuracies for the two methods. However, OC1 used many planes (an average of 36 for the star/galaxy dim data), while MISMIN-P required only one. Thus MISMIN-P produced dramatically simpler trees. Certainly, MISMIN-P will not perform better using a single plane than the decision tree OC1 algorithm on all datasets. But these results do indicate that the simulated annealing approach is not searching the space of possible decisions as effectively as MISMIN-P. We believe that MISMIN-P will be a superior approach for decision tree construction since it selects among alternate error functions and finds better solutions with respect to these error functions.

5 Conclusions

We have investigated the parametric bilinear programming formulation proposed by Mangasarian which minimizes the number of points misclassified by a hyperplane in R^n. MISMIN uses a secant method to select the bilinear subproblems. Each subproblem was solved using a Frank-Wolfe method involving a sequence of uncoupled linear programs. A computationally useful result is that half of the linear programs have a closed form solution. This misclassification minimization problem is closely related to minimizing the average sum of the distances of the misclassified points from the separating plane. Each MISMIN subproblem identifies outliers of the given data set, removes them from the problem, and then minimizes the average sum of the distances of the remaining misclassified points from the separating plane. The evaluation of the series of subproblems leads naturally to the parametric misclassification minimization program, MISMIN-P. The computational results demonstrate that MISMIN performs better than the simulated annealing algorithm, and MISMIN-P provides an improvement over the linear programming solution. Thus no single error metric was always best, and the combination of metrics used in MISMIN-P led to better results.

References

[Ben92] K. P. Bennett. Decision tree construction via linear programming. In M. Evans, editor, *Proceedings of the 4th Midwest Artificial Intelligence and Cognitive Science Society Conference*, pages 97–101, Utica, Illinois, 1992.

[BFOS84] L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees.* Wadsworth International, California, 1984.

[BM92] K. P. Bennett and O. L. Mangasarian. Neural network training via linear programming. In P. M. Pardalos, editor, *Advances in Optimization and Parallel Computing*, pages 56–67, Amsterdam, 1992. North Holland.

[BM93] K. P. Bennett and O. L. Mangasarian. Bilinear separation of two sets in n-space. *Computational Optimization and Applications*, 2:207–227, 1993.

[BU92] C. E. Brodley and P. E. Utgoff. Multivariate decision trees. COINS Technical Report 92-83, University of Massachusetts, Amherst, Massachusetts, 1992. To appear in *Machine Learning.*

[CPL94] CPLEX Optimization Incorporated, Incline Village, Nevada. *Using the CPLEX Callable Library*, 1994.

[DJS+89] R. Detrano, A. Janosi, W. Steinbrunn, M. Pfisterer, J. Schmid, S. Sandhu, K. Guppy, S. Lee, and V. Froelicher. International application of a new probability algorithm for the diagnosis of coronary artery disease. *American Journal of Cardiology*, 64:304–310, 1989.

[FGK93] R. Fourer, D. Gay, and B. Kernighan. *AMPL A Modeling Language for Mathematical Programming.* Boyd and Frazer, Danvers, Massachusetts, 1993.

[Glo90] F. Glover. Improved linear programming models for discriminant analysis. *Decision Sciences*, 21:771–785, 1990.

[Hea92] David Heath. *A geometric framework for machine learning.* PhD thesis, The John Hopkins University, 1992.

[HKS93] D. Heath, S. Kasif, and S. Salzberg. Learning oblique decision trees. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence*, pages 1002–1007, Chambery, France, 1993. Morgan Kaufmann.

[Lue84] D. Luenberger. *Linear and Nonlinear Programming.* Addison-Wesley, Reading, Massachusetts, 1984.

[MA92] P.M. Murphy and D.W. Aha. UCI repository of machine learning databases. Department of Information and Computer Science, University of California, Irvine, California, 1992.

[Man94] O. L. Mangasarian. Misclassification minimization. *Journal of Global Optimization*, 5:309–232, 1994.

[MKS94] S. Murthy, S. Kasif, and S. Salzberg. A system for induction of oblique decision trees. *Journal of Artificial Intelligence Research*, 2:1–32, 1994.

[MKSB93] S. Murthy, S. Kasif, S. Salzberg, and R. Beigel. OC1: Randomized induction of oblique decision trees. In *Proceedings of the Eleventh National Conference on Artificial Intelligence*, pages 322–327, Boston, MA, 1993. MIT Press.

[OSP+92] S. Odewahn, E. Stockwell, R. Pennington, R Humphreys, and W Zumach. Automated star/galaxy discrimination with neural networks. *Astronomical Journal*, 103(1):318–331, 1992.

[Qui84] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1:81–106, 1984.

[Utg89] P. E. Utgoff. Perceptron trees: A case study in hybrid concept representations. *Connection Science*, 1(4):377–391, 1989.

[WM90] W. H. Wolberg and O.L. Mangasarian. Multisurface method of pattern separation for medical diagnosis applied to breast cytology. *Proceedings of the National Academy of Sciences, U.S.A.*, 87:9193–9196, 1990.