Linear programming measures for solving the problem of a linear division of convex multiplayers.

Sarmad H. Ali Osamah A. Ali Samir C. Ajmi

Electronic Computer Center, Al-Muthanna University

sarmad@mu.edu.iq osama@mu.edu.iq Samir@mu.edu.iq

Received: 11\2\2018 Revised: 29\3\2018 Accepted: 2\4\2018

Available online: 10/6/2018

DOI: 10.29304/jqcm.2018.10.2.395

Abstract
In this research, we are trying to solve Simplex methods which are used for successively improving solution and finding the optimal solution, by using different types of methods Linear, the concept of linear separation is widely used in the study of machine learning, through this study we will find the optimal method to solve by comparing the time consumed by both Quadric and Fisher methods.

Keyword
simplex method, linear programming problem, Quadratic simplex method, separating hyperplane
Introduction
The creation of computers and the associated accelerated development of mathematical theories, including mathematical cybernetics and discrete mathematics, as well as information technologies, made it possible to set and solve on the computer new tasks that until recently were exclusively within the competence of man. One of such fundamental problems is the solution of multidimensional problems of linear separation of convex polyhedra. The concept of linear separation is widely used in the study of machine learning. It is also used for the preparation of support vector methods, by means of which problems of complex systems are solved. In this study, some methods for testing the linear separation between two classes of data will be presented[10]. In general form, this problem can be formulated as follows: two subsets (two classes) X and Y of R^d are considered. They are linearly separable if there exists a hyperplane P of R^d and if the elements of X and Y lie on opposite sides of each other. i.e.

\[ P(w, \gamma) = \{z \in R^d : <w, z> = \gamma \}. \]

\[ <w, x> \leq \gamma \ \forall \ x \in X. \]

\[ <w, y> \leq \gamma \ \forall \ y \in Y. \]

\[ P(w, \gamma) \]

Squares and circles represent the two classes. (X and Y)

In Figure 1 (a) is represented by the linearly separable case, as in Figure 1 (b) is not separable case.

Purpose and objectives of the study
The purpose of this study is to develop a software package that numerically realizes and compare the solution of the problem of linear separation of convex polyhedral by various computational means: 1) By solving the linear programming problem by the simplex method; 2) By reducing to the problem of quadratic programming; 3) Through the application of the Fisher’s method. To achieve this goal, it was necessary to solve the following tasks:
- describe a general approach to solving the problem of linear separability of two classes on the basis of the listed methods;
- Design and implement a software package for solving the separation problem, using the developed methods and algorithms. Conduct computational experiments to analyze the effectiveness of the proposed approaches.
Theoretical and practical value
The theoretical value of the study is that it gives a formal description of methods for solving the separation problem.
The practical value of the study lies in the fact that the proposed methods are implemented in the form of a software complex for the computer, which allows to effectively solve the problems of linear separability, in particular, used for dimension equal to 2, for the implementation of visual graphical interpretation.

In conclusion, summarizes the main results of the study, brief conclusions have presented the work.

1.1. The solution of the linear programming problem, Simplex - method.
When solving problems of linear programming, one of the most popular methods is the simplex method. The simplex method is a method of successively improving the solution and finding the optimal solution[1], at the beginning if there is any admissible basic solution corresponding to one of the corner points of the solution polyhedron, and then this solution is purposefully improved by moving to a new basic solution at the neighboring corner point at which the value of the objective function does not decrease to the maximum , until an optimal solution is obtained. This method is universal, with the help of which it is possible to solve any linear programming problem.[2]
The simplex method is intended to solve linear programming problems in canonical form. Depending on the nature of the limits, linear programming problems can be solved with a natural and artificial basis.
If the limits are given by inequalities of the type "≤", then the problem is solved with a natural basis if the limits are given by inequalities of the type "≥" or the equalities "=" , then the solution is conducted with an artificial basis.
If there are three ways to solve the task :
1) The model is solvable.
2) The model is inadmissible.
3) The model is unlimited.
In this study, we use the simplex method to solve the problem of linear separability between two classes. Algorithms for solving the objective function are presented in Tables I and II. The first algorithm makes it possible to find the values of q and p, shows the obtaining of the optimal solution, or vice versa, shows that the problem does not have a solution.

Table I. Algorithm of the objective function
- Data: an array A of size M × N is given, containing the constraints of linear expressions, where the rotation column and rows are represented as p and q.
- Result: the solution is, if the limits of the linear expression are solvable, and otherwise there is no solution.

```
Repea
  q := 0;
  Repeat
    q := q + 1
    Until (q = M + 1) Or (a(0, q) < 0);
    p := 0;
  Repeat
    p := p + 1
    Until (p = N + 1) Or (a(p, q) > 0);
    For i := p + 1 To N Do
      If a[i, q] > 0
        Then
          If ((a[i, M + 1]/a[i, q]) < (a[p, M + 1]/a[p, q]))
            Then
              p := i;
            If (q < M + 1) And (p < N + 1)
            Then
              pivot(p, q)
        Until(q = M + 1) Or (p = N + 1)
```

Table II. Procedure of turning (p, q)
Data: the string p and the column q of the array M × N are given, containing the linear expression constraints that are used to perform the rotation

```
Begin
  For j := 0 To N Do
    For k := M + 1 DownTo 1 Do
      If (j <> p) And (k <> q)
        Then
          a[j, k] := a[j, k] - a[p, k] * a[j, q]/a[p, q];
    For j := 0 To N Do
      If j <> p
        Then
          a[j, q] := 0;
    For k := 0 To M + 1 Do
      If k <> q
        Then
          a[p, k] := a[p, k]/a[p, q];
    a[p, q] := 1;
End
```

These algorithms can be used to solve the jective function.
We illustrate the simplex method, showing Linear programming through some task Z[7].

Task Z: Let X = {(1,1)} and Y = {(0,0), (1,0), (0,1)} be the initial data for two classes that define the problem Z. We check X || Y?
Adj: is the operation of adding a value, in this case, we add 1 if the point belongs to class 1, otherwise -1 and the coordinates of points from 2 classes change the sign $S = \text{Adj}(X, 1) \cup -\text{Adj}(Y, 1)\{1,1,1,0,0,0,-1,-1,0,-1,-1,-1\}$. 

So, to find out $X \parallel Y$, we need to find a set of values for weights $w_1 + w_{1+1}$, and threshold $t$, so that it minimizes one of the indicated inequalities.

On Condition

\[
\begin{align*}
\begin{cases}
  w_1 + w_2 + t > 0 \\
  t > 0 \\
  w_2 - t > 0 \\
  w_1 - t > 0
\end{cases}
\end{align*}
\]

(1.1)

Since the Simplex method limits the values of variables that are $\geq 0$, and the weight value can be either positive or negative, we transform each of our original variables as a difference of two variables. That is, we reduce it to the canonical form:

\[
\begin{align*}
  w_1 &= w_{1,aux} - w_{1,aux} \\
  w_2 &= w_{2,aux} - w_{2,aux} \\
  t &= t_{aux} - t_{aux}
\end{align*}
\]

(1.2)

Since the simplex method does not take a strict inequality, we change our sign $> 0$ by $\geq 1$.

Using the above transformations, our new set of constraints becomes:

\[
\begin{align*}
\begin{cases}
  (w_{1,aux} - w_{2,aux}) + (w_{2,aux} - w_{2,aux}) + (t_{aux} - t_{aux}) \geq 1 \\
  (-t_{aux} - t_{aux}) \geq 1 \\
  (-w_{1,aux} + w_{2,aux}) - (t_{aux} + t_{aux}) \geq 1 \\
  (-w_{2,aux} + w_{2,aux}) - (t_{aux} + t_{aux}) \geq 1
\end{cases}
\end{align*}
\]

(1.3)

Applying the simplex method next, we obtain an admissible solution which gives the following result:

\[
\begin{align*}
  w_{1,aux} &= 2 \\
  w_{2,aux} &= 0 \\
  w_{2,aux} &= 2 \\
  t_{aux} &= 0 \\
  t_{2,aux} &= 3
\end{align*}
\]

(1.4)

Initial data from us $x_i \in R^n$, $i = 1, ..., l$ and $c_i, c \in \{-1, 1\}$.

We find the inverse matrix by the Gauss method, for the convenience of the definition we denote $S^{-1} = A$ and $S^{-1} = X$, then we use the equation $A^*X = E$, where: $X$ is the inverse matrix of the matrix $A$, and $E$ is the identity matrix.

We reduce the matrix $A$ to triangular matrix using the Gauss algorithm“Direct motion”.

In parallel, we perform similar operations with the unit matrix $E$.
We finally represent the obtained equation in its expanded form (fig 1.2)

The method of support vectors should find a solution to the minimization problem:

\[ \frac{1}{2} ||w||^2 \]

On Condition \( c_i (w^T x_i + b) \geq 1 \), Where \( C > 0 \) represents the penalty parameter.

We solve the following minimization problem:

\[ \frac{1}{2} ||w||^2 \] (1.5)

Formal description of the task:

We believe that the points have the form:

\[ (x_1, c_1), (x_2, c_2), \ldots, (x_n, c_n) \]

Where \( c_i \) takes the values 0 or 1, depending on which class the point \( x_i \) belongs to. Each \( x_i \) is a \( p \)-dimensional real vector, usually normalized by the values [0,1]. If the points are not normalized, the point with large deviations from the average coordinates of the points will affect the classifier too much. We can treat this as a learning collection, in which the class to which it belongs is already assigned to each element. We want the algorithm of the support vector method to classify them in the same way. To do this, we construct a separating hyperplane that looks like this:

\[ w^T x - b = 0; \] (1.6)

Where \( w, b \) are unknown, they need to be found.

The vector \( w \) is the perpendicular to the separating plane (hyperplane);

\[ \frac{b}{||w||} \] - the distance from the hyperplane to the origin;

If the parameter \( b \) is zero, the hyperplane passes through the origin, which limits the solution.

\[ ||w|| = \sqrt{\sum_{i=1}^{n} w_i^2} \] - is the length of the vector (1.7)

Since we are interested in the optimal separation, we are interested in support vectors and hyperplanes parallel to the optimal and closest to the supporting vectors of two classes. It can be shown that these parallel hyperplanes can be described by the following equations (up to normalization).

\[ w^T x - b = 1, \] (1.8)

\[ w^T x - b = -1. \] (1.9)

If the training sample is linearly separable, then we can choose hyperplanes in such a way that no points of the training sample lie between them and then maximize the distance between the hyperplanes. The width of the strip between them is easy to find from considerations of geometry, it is \( \frac{2}{||w||} \), so our task is to minimize \( ||w|| \).

To exclude all points from the strip, we must verify for all \( i \) that

\[ w \ast x_i - b \leq -1, c_i = -1 \]

\[ w \ast x_i - b \geq 1, c_i = 1 \] (1.10)

This can also be written in the form:

\[ c_i (w \ast x_i - b) \geq 1; \quad 1 \leq i \leq n \] (1.11)

The problem of constructing an optimal separating hyperplane reduces to minimizing \( ||w|| \). Under the condition (1.11). This is the problem of quadratic optimization, which has the form:

\[ \text{min } ||w^2|| \]

\[ c_i (w \ast x_i - b) \geq 1; \quad 1 \leq i \leq n \] (1.12)

By the Kuhn-Tucker theorem (necessary conditions for solving the problem of non-linear programming), this problem is equivalent to the dual problem of finding the saddle point of the Lagrange function.

\[ L(w, b, h) = \frac{1}{2} ||w^2|| - \sum_{i=1}^{n} h_i (c_i (w \ast x_i - b) - 1) \]

\[ \lambda_i \geq 0, 1 \leq i \leq n \] (1.13)

\[ \lambda = (\lambda_1, \ldots, \lambda_n) \] - a vector of dual variables.

We reduce this problem to an equivalent quadratic programming problem, which contains only dual variables:

\[ -L(\lambda) = - \sum_{i=1}^{n} \lambda_i + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j c_i c_j (x_i x_j) \rightarrow \text{min } \lambda_i \]

\[ \lambda_i \geq 0, 1 \leq i \leq n \]

\[ \sum_{i=1}^{n} \lambda_i c_i = 0 \] (1.14)
All variables are $\lambda$.

We single out the linear and quadratic parts as follows:

$$-L(\lambda) = - \sum_{i=1}^{n} \lambda_i + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j c_{ij} (x_i x_j) \rightarrow \min_{\lambda_i}$$

$$CX + X^TDX$$

(1.15)

Next, we make a system check (1.10). Namely, a quadratic function having a nonnegative definite quadratic form is minimized, hence, the objective function is convex. An admissible region is a polyhedron, which means that we are faced with the problem of quadratic programming.

Hence we can apply the Kuhn-Tucker optimality theorem.

We have a task: $CX + \frac{1}{2}X^TDX \rightarrow \min_{(1.16)}$

Under the constraints: $AX \leq b, X \geq 0$.

Next, we apply the Kuhn-Tucker theorem. The convexity conditions of the objective function and the admissible domain are allowed to use this theorem.

We record the Kuhn-Tucker system:

$$\begin{align*}
Ax &+ Y = b \\
-Dx - A^{T}[u] + V = C \\
V^{x} & = 0, \quad U^{y} = 0 \\
X, Y, U, V & \geq 0
\end{align*}$$

(1.17)

Hence we have a system of equations.

$V^{x} = 0, \quad U^{y} = 0$— is called the complementary non-rigidity condition.

That is, if we introduce vectors $y = (y_1, \ldots, y_n)$ and $v = (v_1, \ldots, v_m)$, then relation

$$(c + 2x D + uA)x^T = 0$$

$$x A^T - b \leq 0$$ (1.18)

$$(x A^T - b)u^T = 0$$

Will look like:

$$c + 2x D + uA - \varepsilon = 0 \quad (1.19)$$

$$x A^T - b + v = 0 \quad (1.20)$$

$$\varepsilon x^T T - v u^T = 0 \quad (1.21)$$

The system (1.19) - (1.20) consists of $n + m$ linear equations relatively $2(m + n)$ unknown $x_j$, $y_j$ ($j = 1, \ldots, n$), $u_i, v_i$ ($i = 1, \ldots, m$).

In addition, as follows from conditions (1.18), there should be:

If $x_j > 0$, then $y_j = 0$ (1.22)

If $y_j > 0$, then $x_j = 0$ (1.23)

If $u_i > 0$, then $v_i = 0$ (1.24)

If $v_i > 0$, then $u_i = 0$ (1.25)

Consequently, the desired solution of the system (1.19) - (1.20) can be an arbitrary inseparable basic solution of it, but such that the variables $x_j$ and $y_j$ (and also $u_i$ and $v_i$) with the same indices cannot be basic at the same time. To find such a solution, one can apply any of the known methods of LP, in particular, the artificial base method.

To this goal, we write the system (1.19) - (1.20) in the form

$$2x D + uA - b = -c \quad (1.26)$$

$$x A^T + v = b \quad (1.27)$$

Without limiting universality, we will assume that the right-hand parts of this system are inalienable.

According to the artificial basis method, in every equation of the system (1.26) — (1.27), which does not contain a basic variable, we introduce an artificial variable. Since the variables $v_i$ ($i = 1, \ldots, m$) can be considered basic, then the artificial variables $z$ ($z_1, \ldots, z_m$) We introduce only Eq. (1.26) and consider the auxiliary CLRF.

$$z^T \min \quad (1.28)$$

$$\begin{align*}
2x D + uA - b + z = -c \\
x A^T + v = b
\end{align*}$$

(1.30)
Where \( i = (1,1,...,1) — n \) Dimensional unit vector.

Next, we solve this system, applying the rule, the transition from one basis to another, which is used in the simplex method.

We have, that \( V_i x_i = U_j y_j \) For all \( i,j \). The fulfillment of this condition means that if the variable \( V_i \) in the basic solution takes a positive value, then the variable \( U_j \) can not be a basic variable and take a positive value. Similarly, the variables \( x_i \) and \( y_j \) cannot simultaneously take positive values. That is, this rule is also applied in the simplex method.

### Quadratic simplex method

If the found admissible basic solution of this problem satisfies the conditions of complementary non-rigidity, then it determines the optimal solution of the original. Otherwise, we must pass to a new admissible basic solution. In this case, a new variable with a zero estimate is included in the basic variables. The simplex method with the conditions (1.22) - (1.25) for the solution of the auxiliary CLPP (1.28) - (1.30), constructed on the basis of the problem of convex quadratic programming (1.16), is called the quadratic simplex method (the algorithm of the ordinary simplex method)[9].

If in the optimal solution of the auxiliary CPRF (1.28) - (1.30) all the artificial variables \( z_j \) (j=1,...,n) take zero values, then discarding them, we obtain the DBR of the system (1.26) - (1.27). That part of it that corresponds to the variables of the initial problem of convex quadratic programming (1.16) is its optimal solution.

If the value of at least one of the artificial variables is different from zero in the optimal solution of the auxiliary CLRP (1.28) - (1.30), then the system (1.26) - (1.27) has no solutions, and consequently the set of saddle points of the Lagrange function of the initial problem is empty.

Let's say we solved the problem and found \( \lambda = (\lambda_1,..,\lambda_n) \) and now we can find a hyperplane that separates two classes:

\[
w = \sum_{i=1}^{n} \lambda_i c_i x_i, \quad (1.31) \\
b = \omega x - c_i, \quad \lambda_i > 0 \quad (1.32)
\]

The summation does not go over the entire sample, but only over reference ones, for which \( \lambda_i \neq 0 \).

The advantage of support vector methods is that the problem of convex quadratic programming has a unique solution. And also the support vector method is the best method of linear classification.

1.3. The solution of the problem of linear separation by the Fisher method

Fisher's linear discriminant in the original value is a method that determines the distance between the distributions of two different classes of objects or events. It can be used in problems of machine learning with a statistical (Bayesian) approach to solving classification problems[3].

Suppose that the training sample satisfies the following hypotheses in addition to the basic hypotheses of the Bayesian classifier:

Classes are distributed according to the normal law

Covariance matrices of classes are equal

Such a case corresponds to the best division of classes by the Fisher discriminator (in the original value). Then the statistical approach leads to a linear discriminant, and it is this classification algorithm that is now often understood as the linear discriminant of Fisher[4].

Under certain general assumptions, the Bayesian classifier reduces to the formula:

\[
a(x) = \arg \max_{y \in Y} \lambda_y \phi_y(x) \quad (1.33)
\]

Where \( Y \) is the set of answers (classes), \( x \) belongs to the set of objects \( X \), \( \phi_y \) is the a priori probability of class \( y \), \( \lambda_y \) is the weighting factor (error price on object of class \( Y \)).

In the extension of all the above hypotheses, in addition to the hypothesis of the equality of covariance matrices, this formula has the form:

\[
a(x) = \arg \max_{y \in Y} (\ln(\lambda_y) \sigma_y^{-1}(x - \mu_y) - \frac{1}{2}(x - \mu_y)^T \sigma_y^{-1}(x - \mu_y)) \quad (1.34)
\]

Where \( \mu_y = \frac{1}{l_y} \sum_{x_i \in Y} x_i, \sigma_y = \frac{1}{l_y} \sum_{x_i \in Y} (x_i - \mu_y)(x_i - \mu_y)^T \) Approximations of the expectation vector and the covariance matrix of class \( Y \), obtained as maximum likelihood estimates, \( l \) is the length of the training sample, \( l_y \) is the number of objects of class \( Y \) in the training sample \( x \in R^n \).
This classification algorithm is A quadratic discriminant. It has a number of drawbacks, one of the most important of which is the poor conditionality or degeneracy of the covariance matrix $\Sigma_Y$ with a small number of training elements of class $Y$, which can result in a strongly distorted result when the given matrix $\Sigma_Y^{-1}$ in the past. And the entire classification algorithm is unstable, it will work poorly (it is also possible that the inverse matrix $\Sigma_Y^{-1}$ does not exist at all).

The linear discriminant of Fisher solves this problem

The main idea of the algorithm

When accepting the hypothesis of equality between covariance matrices, the classification algorithm takes the form:

$$a(x)=\arg\max_{y\in Y}[\ln(p_y) - \frac{1}{2} \mu_y^T \Sigma_Y^{-1} \mu_y + x^T \Sigma_Y^{-1} \mu_y], \quad OR \quad a(x)=\arg\max_{y\in Y}(\beta_y + x^T \alpha_y) \quad (1.35)$$

Simplicity of classification by the linear discriminant Fisher -

One of the advantages of the algorithm: in the case of two classes in the two-dimensional characteristic space, the separating surface will be a straight line. If there are more than two classes, then the dividing surface will be piecewise-linear. But the main advantage of the algorithm in comparison with the quadratic discriminant is the decrease in the effect of the poor conditionality of the covariance matrix with insufficient data.

For small $l_y$ approximations

$$\Sigma_Y = \frac{1}{l_y} \sum_{y=1}^{l_y} (x_i - \mu_y)(x_i - \mu_y)^T \quad (1.36)$$

Will give a bad result, therefore, even in those problems where it is known that classes have different forms, sometimes it is advantageous to use the heuristic of the Fisher discriminator and to consider the covariance matrices of all classes to be equal. This will allow us to calculate some "average" covariance matrix using the entire sample[6]:

$$\sum = \frac{1}{l_y} \sum_{i=1}^{l_y}(x_i - \mu_Y)(x_i - \mu_Y)^T \quad (1.37)$$

The use of which in most cases will make the classification algorithm more stable.

Formulation of the problem:

Let $X_1$ and $X_2$ represent a certain problem of classifying points into two classes. And let $X_1 = \{x_1^1, x_1^2, \ldots, x_1^n\}$ Set of points belonging to class $X$

and $X_2 = \{x_2^1, x_2^2, \ldots, x_2^n\}, x_i^j \in R^d$, Belong to the class $Y$.

By Fisher's method, we solve the problem posed and determine whether the points are separable or not. The linear Fisher discriminator is given by the vector $w$, which maximizes

$$J(w) = w^T S_w w / w^T S_w w^T$$

Where $S_w$ Selective covariance matrix[8]

The covariance matrix in probability theory is a matrix composed of the pairwise covariances of the elements of one or two random vectors. The covariance matrix of a random vector is a square symmetric non-negative definite matrix on the diagonal of which the variances of the vector components are located, and the off-diagonal elements are the covariance between the components. The covariance matrix of a random vector is a multidimensional analog of the variance of a random variable for random vectors. The covariance matrix of two random vectors is a multidimensional analog of the covariance between two random variables.

It is given by the following formulas:

$$S_w = (m_1 - m_2)(m_1 - m_2)^T \quad (1.38)$$

$$S_w = \sum_{i\in(1,2)} \sum_{j=1}^{l_i} (x_i^j - m_i)(x_i^j - m_i)^T \quad (1.39)$$

$$m_i = l_i^{-1} \sum_{j=1}^{l_i} x_i^j \quad (1.40)$$

$l_i$ - Number of elements in a class $l$.

The parameter of the linear classifier is given by the vector $\widehat{W}$:

$$q(x) = 0, \quad for \ \widehat{W}^T \times \widehat{x} + b \geq 0 \quad (1.40)$$

$$q(x) = 1, \quad for \ \widehat{W}^T \times \widehat{x} + b < 0 \quad (1.41)$$
Example 1. Let the points belong to the classes X and Y.

\[ X = \{-1; 3\}; \{-1; -2\}; \{-3; 2\} \]
\[ Y = \{-1; 2\}; \{2; 1\} \]

\( m_1 \) - Mean point value \( x_i \);
\( m_2 \) - Mean point value \( y_i \);

For each class, calculate the mean of the points:

\[ m_1 = (-1, 6666; -1) \]
\[ m_2 = (0,5; 1,5) \]

\( S_1 \) Matrix of selective covariance of points
\( S_2 \) Matrix of selective covariance of points \( y_i \)

\[ S_1[i, j] = \text{Cov}(x_i, x_j) \] (1.42)
\[ S_2[i, j] = \text{Cov}(y_i, y_j) \] (1.43)

\( S_W \) - Sum of two matrices

\[ S_W = S_1 + S_2 \Rightarrow S_W[i, j] = \text{Cov}(x_i, x_j) + \text{Cov}(y_i, y_j); \] (1.44)

\( S_W^{-1} \) - inverse matrix

The inverse matrix is a matrix \( S_W^{-1} \) multiplied by which the original matrix \( E \) gives the unit matrix \( E \):

\[ S_W^{-1} \times S_W = E \] (1.45)

We find the inverse matrix by the Gauss method, for the convenience of the definition we denote \( S_W = A \) and \( S_W^{-1} = X \), we use the equation \( A \times X = E \), where: \( X \) is the inverse matrix of the matrix \( A \), and \( E \) is the identity matrix. We reduce the matrix \( A \) to a triangular matrix using the Gauss algorithm "direct motion". In parallel, we perform similar operations with the unit matrix \( E \). We represent the obtained equation in its expanded form (see Figure 1.2)[9].

\[ X_{ji} = \frac{1}{A_{jj}} (E_{jj} - \sum_{k=n} A_{jk} X_{ki}) \] (1.46)

\( i=1...n, j=n...1 \)

\( W \) - The normal of the hyperplane:

\[ W = S_W^{-1}(m_1 - m_2) \] (1.47)

Finally, we calculate the hyperplane shift:

\[ b = (w \times m_1 - w \times m_2) / 2 \] (1.48)

\[ \text{If} \ w, x > -b \geq 0 \]
\[ \text{If} \ w, x \geq b \]

Known Known \( b = 0,42337 \)
The heuristic of the *linear Fisher discriminator* is in some way a simplification of the quadratic discriminant. It is used to obtain a more stable classification algorithm. It is most appropriate to use the *linear discriminant of Fisher* when Data for training is not enough. Owing to the main hypothesis on which the algorithm is based, it is most successfully solved by simple classification problems in which the forms are "similar" to each other in forms. The classification process by the linear Fisher discriminator can be described by the following scheme:

- Estimation of mathematical expectations
- Calculation of the general covariance matrix and its inversion

Our sample of data

(Fig 1) Feeding this data to the program

We start on calculation and we see the following result:

In this case, the points are linearly separable. And they are separated from each other by a separating hyperplane. Where $b$ is the hyperplane shift, $w$ is the normal of the hyperplane, $\min_X$ is the minimal scalar product of the vectors $z_i$ by the normal, is calculated by the formula:

$$\min_{i=1}^{m} < w, z_i >$$

$max_Y$ is the maximum scalar product of the vectors $z_i$ by the normal, is calculated by the formula:

$$\max_{j=1}^{m} < w, p_j >$$

$$< w, x > = b \quad \text{Hyperplane shift.}$$

The thickness of the hyperplane $\varepsilon_w$ in separable cases is calculated as follows:

$$\overline{\varepsilon} = \frac{w}{||w||} \left( \frac{w}{||w||} \right) = \left| \min_{i=1}^{m} < z_i, \frac{w}{||w||} > \right|$$

$$- \frac{w}{||w||} < p_j \left| t(\overline{\varepsilon}) \right| \left| \min_{i=1}^{m} < z_i, \overline{\varepsilon} > \overline{\varepsilon} - \frac{w}{||w||} < p_j, \overline{\varepsilon} > \right|$$
The quadratic method and see the following result in practice:

Time consumed 0.0840048 Sec

The Fisher method in practice:

Time consumption 0.01560089 Sec

Results

<table>
<thead>
<tr>
<th>METHOD</th>
<th>TIME CONSUMPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXAMPLE 1</td>
<td></td>
</tr>
<tr>
<td>Quadric</td>
<td>0.0840048</td>
</tr>
<tr>
<td>Fisher</td>
<td>0.01560089</td>
</tr>
<tr>
<td>EXAMPLE 2</td>
<td></td>
</tr>
<tr>
<td>Quadric</td>
<td>0.0890051</td>
</tr>
<tr>
<td>Fisher</td>
<td>0.1480084</td>
</tr>
<tr>
<td>EXAMPLE 3</td>
<td></td>
</tr>
<tr>
<td>Quadric</td>
<td>0.0320019</td>
</tr>
<tr>
<td>Fisher</td>
<td>0.0890051</td>
</tr>
<tr>
<td>EXAMPLE 4</td>
<td></td>
</tr>
<tr>
<td>Quadric</td>
<td>0.0350002</td>
</tr>
<tr>
<td>Fisher</td>
<td>0.0920052</td>
</tr>
</tbody>
</table>

Another side view where there is no solution

The both methods did not find solution.

Conclusion

In the study, the problems related to the methods of solving the linear separation problem were considered. A software package was developed that numerically realizes the solution of the problem of linear separation of convex polyhedral by various computational means:

1) By solving the linear programming problem by the simplex method;
2) By reducing to the problem of quadratic programming;
3) Through the application of the Fisher method.

A general approach to solving the problem of linear separability of two data classes based on the listed methods was described. Computational experiments were conducted to analyze the effectiveness of the proposed approaches. A comparison of the effectiveness of the approaches considered was carried out for problems of different dimensionality of space and the number of points defining separable sets with respect to the following indicators:

1) The time for solving the problem,
2) The thickness of the separation layer (formed between two hyperplanes supporting partially shared sets) of two classes of points.

Comparison of methods

The task was to compare methods and determine which one is effective. When comparing, it is necessary to take into the complexity of the task, the time of performance of work and the value of the measure of the thickness of the compartment.
1. When input data of dimension \( n = 2 \), from the point of view of the thickness of separation by a hyperplane, the Simplex method and the quadratic programming method are effective. In most cases, they have the same thickness of separation \( t_w \). The results obtained (for dimension \( n = 2 \)) can also be compared by the time of the work. After doing a lot of experiments, we saw that the quadratic programming method is optimal for the time of execution of the work.

2. If the input data has the dimension \( n = 3, \ldots n = 10 \), etc., then the best method is the method based on quadratic programming. The effectiveness of the method of quadratic programming does not depend on the dimension. When comparing the thickness of the compartment, in these tests quadratic programming provided the greatest thickness.

And by the time of the work, the optimal is the simplex method.

References


سرماد حمزة علي
اسامة عبد الجليل علي
سمير قيصر عجمي
مركز الحاسوب الالكتروني / جامعة المثنى
sarmad@mu.edu.iq
osama@mu.edu.iq
Samir@mu.edu.iq

المستخلص:
في هذا البحث، نحاول حل العادلات البسيطة التي تستخدم في إيجاد الحل الأمثل، وذلك باستخدام أنواع مختلفة من المعادلات الخطية. إن استخدام مفهوم تعلم الآلة (Machine Learning) الآن يعتبر من أهم المواضع على نطاق العالم حيث استخداماته واسعة تصل حتى لقيادة السيارة. ولهذا الغرض نستخدم مفهوم التعلم الخطي لقياس جودة نموذج تعلم الآلة، من خلال هذه البحث، سوف نجد الطريقة المثلى لحل هذه المعضلة من خلال مقارنة المثل والمثل، من خلال هذه البحث، سوف نجد

91