Application of Machine Algorithms for Classification and Formation of the Optimal Plan

Nataliya Boyko and Rostyslav Hlynka

Lviv Polytechnic National University, Profesorska Street 1, Lviv, 79013, Ukraine

Abstract

The paper presents three methods for data classification and finding the optimal plan: the study of the quadratic programming problem, the double problem and the Support Vector Machine method. It is known that linear programming is used to solve resource allocation problems. Also, its purpose is widely used to determine the highest profit or lowest cost, inventory management, the formation of an optimal transportation plan or to determine research, and so on. An important approach to the application of linear programming problems is the use of the duality principle, which is methodologically related to the theory of systems of dependent inequalities. This aspect better explains the concept of duality in linear programming problems with general mathematical rigor.

Keywords 1

Data Mining, Mathematical Programming, Linear Programming, Nonlinear Programming, Quadratic Programming, Problem of the Quadratic Programming, Support Vector Machine

1. Introduction

Known methods of transition from a primal problem to a dual one are based on qualitative transformations and are meaningful. Formalization and proof of the correctness of the algorithm for constructing a dual problem for an arbitrary form of representation of a primal problem will make it easy to obtain correct pairs of known dual problems. The relevance of research is due to the requirements for simplification of solutions of linear programming problems based on the development of a formal algorithm for the transformation of a primal problem to a dual linear optimization problem [1, 6].

Quadratic programming is a area of mathematical programming devoted to the theory of solving problems characterized by a quadratic dependency between variables [2]. The usage of this method is relevant today, as the use of mathematical models is an important factor in improving the planning of the company. Mathematical representation of data allowed to create and model different options for choosing the optimal solution [3, 9, 11].

The paper considers the Support Vector Machine (SVM) method that is taught by examples and used to classify objects. It is established that SVM can be successfully used to control complex electromechanical systems, it can ensure the adaptability of control algorithms, perform the functions of an observer, an identifier of unknown parameters, a reference model, with its help you can control complex nonlinear objects, as well as objects with stochastic parameters [4, 10, 17].

The aim of the work is to solve a dual problem by SVM, the comparison with the primal problem and classification of the dataset.

Achieving this goal involves solving specific tasks:

- determine the problem of the method of SVM for the dual problem;
- compare dual SVM and primary;

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: nataliya.i.boyko@lpnu.ua (N. Boyko); hlynka1608@gmail.com (R. Hlynka)
ORCID: 0000-0002-6962-9363 (N. Boyko); 0000-0003-0827-3145 (R. Hlynka)

- analyze this method;
- apply it in practice.

The object of research is to solve a dual problem by the method of SVM.

The purpose of the study is to apply the problems of linear and nonlinear programming to study the properties of the studied problems, to determine their advantages and disadvantages.

2. Methods

Mathematical programming is an applied mathematical discipline that investigates the extremum of a function (maximum or minimum search problems) and develops methods for solving them. Such problems are also called optimization [5, 7].

The area of mathematical programming can be applied to the type of objective function and to the system of constraints. As a result, we obtain a division into:

- Linear programming - objective function and constraint functions included in the constraint system are linear (first order equation).
- Nonlinear programming - the objective function or one of the constraint functions included in the constraint system is nonlinear (higher order equations).
- Integer (discrete) programming - if at least one variable has an integer constaint.

Dynamic programming - if the parameters of the objective function and / or system of constraints change over time or the objective function has an additive / multiplicative form or the decision-making process itself is multi-step [8, 12].

Depending on that all the information about the process is known in advance, the field of mathematical programming is divided into:

- Stochastic programming - not all information about the process is known in advance: the parameters included in the objective function or in the constraint function are random or have to make decisions in conditions of risk.
- Deterministic programming - all information about the process is known in advance.

Depending on the number of objective functions, the tasks are divided into:

- Single-criteria;
- Multicriteria.

The optimization problem can be classified as follows: those problems that describe the properties of the constraint system and, accordingly, others that are determined by the objective function:

- Unconditional optimization problems or problems without restrictions - they do not impose restrictions on quantitative variables.
- Conditional optimization problems or constrained problems - in these problems, quantitative variables are constrained.
- Optimization problems for incomplete data - they have a goal function or a system of constraints depend on some parameter p (numerical, vector), the value of which is completely undefined at the time of solving the problem.

The first type includes optimization problems, the task of which is to minimize or maximize the quadratic function of several variables with linear constraints on these variables.

Quadratic programming problems include a special class of NP problems in which the objective function $f(x)$ is quadratic and concave (or convex), and all constraints are linear [13, 15].

Each linear programming problem can be matched to another that relates in some way to the original task. Such problems are called dual, or conjugate. Joint consideration of dual pairs of problems is very important in the economic analysis of the optimal plan. The correspondence between the original and dual problems is to build a dual problem on the basis of the first problem (as the source can be considered any of the conjugate pair of problems). Dual problems are symmetric and asymmetric.

The quadratic programming includes the SVM method. He constructs a model in the form of points in space using a binary linear classifier. This model goes through a series of iterations in which new patterns are displayed in a given space and determine the side of the gap. On the basis of these data the forecast of belonging of samples to a certain category is made.

3. Observation and analysis of the existing methods and means

Each linear programming problem corresponds to a dual, formed by certain rules directly from the condition of the primal problem. Comparing these two formulated problems, we conclude that the dual linear programming problem is formed from a primal problem by the following rules:

1. Each constraint of a primal problem corresponds to a variable of a dual problem. The number of unknowns of a dual problem is equal to the number of constraints of the primal problem.
2. For the primary problem, a certain variable corresponds to the specified constraint of the double problem and, accordingly, their number determines the number of unknowns in the primary problem.
3. If the objective function of the primary problem goes to max, then the objective function of the double problem goes to min, and vice versa.
4. In the objective function of a double problem, the coefficients of the variables are free values of the system of constraints for the primary problem.
5. The column of free members of the double problem is the coefficients for the variables in the objective function of the primary problem.
6. The coefficients of variables in the system of constraints of the primary problem are written in the matrix and accordingly it is transposed to determine the coefficients of constraints for the double problem.

As a result of intensive research in the field of machine learning, aimed at improving the quality of classifiers, a new generation of methods appeared, in particular – SVM. This method refers to machine learning methods based on vector spatial models, the purpose of which is to find dividing surfaces between classes as far as possible from all points of the study population (perhaps ignoring some points such as emissions or noise) [14, 16].

If the training set contains two classes of data that allow linear division, then there are a large number of linear classifiers with which you can divide this data. It is intuitively clear that a dividing surface passing through the middle of a strip separating two classes. For example, the perceptron allows you to find at least one linear, other methods, such as the naive Bayesian method, find the best linear separator using a certain criterion. In particular, SVM necessarily assumes that the decisive function is completely determined by a subset of data that affect the position of the delimiter. In vector space, a point can be considered as a vector passing through the origin. Consider a dataset of n points in the form $(x_1, y_1),...,(x_n, y_n)$, where $y_1 \in \{-1,1\}$ identifies to which class the point x belongs.

Each point is a p-dimensional real vector. SVM means to find the maximum hyperplane that separates groups of points belonging to $y = 1$ from $y = -1$.

In Equation 1 we write a hyperplane through many points that satisfy the condition:

$$w * x - b = 0, \qquad (1)$$

where w is a optional vector to the hyperplane. Parameter $\frac{b}{\|w\|}$ determines the displacement of the hyperplane from the origin on the normal vector (Figure 1).

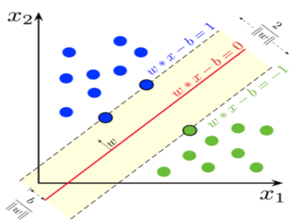

Figure 1: Visual representation of SVM

In Equation 2 we define the data that are not linearly separated:

$$\max(0, 1 - y_i(wx_i - b)). \qquad (2)$$

In Equation 3, we minimize the function:

$$\left[\frac{1}{n}\sum_{i=1}^{n}\max(0,1-y_i(wx_i-b))\right]+\gamma\|w\|^2, \quad (3)$$

where γ determines the trade-off between the size of the margin and the guarantee that the point lies on the correct side of the margin. Hence, if γ is very small, the second operand becomes insignificant, and the function will behave as with a hard margin.

The calculation of the soft margin classifier is to minimize the expression of the form (Equation 4):

$$\left[\frac{1}{n}\sum_{i=1}^{n}\max(0,1-y_i(wx_i-b))\right]+\gamma\|w\|^2. \quad (4)$$

Therefore, in a further study, we will consider a classifier with a bounded boundary.

3.1. Primal problem

Equation 4 presents the minimization of a bounded optimization problem with a differentiated objective function. For each i we enter a variable e_i - the least positive number that satisfies $y_i(w*x_i-b)\geq e^i$ and $e_i\geq 0$.

Equation 5 presents the problem of optimization taking into account additions.

$$\text{minimize}\ \frac{1}{\text{n}}\sum_{i=1}^{n}e_i+\gamma\|w\|^2. \quad (5)$$

Solving the primary problem for the dual Lagrange, we obtain a simplified problem (Equation 6):

$$\text{maxmize} f(c_1,...,c_n)=\sum_{i=1}^{n}c_i-\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}y_ic_i(x_i*x_j)y_jc_j, \quad (6)$$

if: $\sum_{j=1}^{n}y_ic_i=0$ and $0\leq c_i\leq\frac{1}{2n\gamma}$.

The quadratic function solves double maximization problems. its results satisfy linear constraints. Equations 7-8 determine the variables c_i to determine the second problem.

$$\text{maxmize} f(c_1,...,c_n)=\sum_{i=1}^{n}c_i-\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}y_ic_i(x_i*x_j)y_jc_j. \quad (7)$$

$$w=\sum_{i=1}^{n}y_ic_ix_i. \quad (8)$$

Moreover, $c_i=0$ just when the point lies on the right side of the field and $0\leq c_i\leq\frac{1}{2n\gamma}$, when lying on the edge of the field. Equation 9 shows a linear combination of reference vectors, which determines the offset through a point on the field boundary.

$$y_i(wx_i-b)=1\geq b=wx_i-y_i. \quad (9)$$

3.2. Comparison of problems

Equation (Formula 8) gives the optimal value of w in terms of c. Suppose we have adjusted the parameters of our model to the training set, and now we want to make a prediction for the new point x. Then we will calculate w^Tx+b and forecast $y=1$, if and only if this value is greater than zero. But, using (Formula 3), this value can also be written as (Equation 10):

$$w^Tx+b=\sum_{i=1}^{n}(y_ic_i(x_i))^Tx+b=\sum_{i=1}^{n}y_ic_i(x_i,x)+b. \quad (10)$$

So, if we find c_i to make a prediction, we have to calculate a value that depends only on the internal product between x. Moreover, we have previously seen that c_i will be equal to all but zero support vectors. Thus, many terms in the above sum will be zero, and we really only need to find the internal products between x and the reference vectors (which are often only a small number) to calculate (Formula 9) and make our prediction.

Considering the dual form of the optimization problem, we got a good idea of the structure of the problem, and we can write the whole algorithm in terms of only the internal products between the vectors of the input functions. This property is important to apply kernels to our classification problem. The obtained algorithm, supporting vector machines, will be able to effectively learn in spaces with high dimensions.

Also dual SVM requires fewer kernel estimates than the primary. Therefore, it gives a more stable result in less computational time (Table 1).

Table 1
Comparison between primal and dual SVMs on the different datasets

SVM	Data	SVs	Iterations	Kernels	Rec.	Time
Dual	Thyroid	98	1439	2,660,742,540	97.81 (100)	220
	Blood cell	188	13	86,324,885	93.58 (97.19)	10
	H-50	77	20	386,391,968	99.28 (100)	50
	H-13	39	83	2,343,986,783	99.55 (100)	282
	H-105	91	22	812,259,183	100 (100)	147
	Satimage	1001	28	600,106,924	91.70 (100)	157
	USPS	597	19	593,529,638	95.47 (100)	114
Primal	Thyroid			No convergence		
	Blood cell	203	10	445,319,582	93.61 (97.19)	21
	H-50	70 (78)	15	605,637,116	99.28 (100)	52
	H-13	99	12	749,712,635	99.70 (99.96)	93
	H-105	111	13	907,360,383	100 (100)	140
	Satimage	1006	25	26,125,955,619	91.70 (99.71)	1258
	USPS	604	16	8,116,273,966	95.47 (99.99)	412

Table 1 shows the results of the primal and dual SVM, using mark datasets.

Contains 7 columns:

- SVM – the type of problem of the used method of support vectors;
- Data – the name of the used dataset;
- SVs – solution function to determine the number of vectors;
- Iterations – number of steps;
- Kernels – the number of calls to the kernel;
- Rec. – recognition accuracy;
- Time – training time.

In terms of stable convergence and learning speed, a dual SVM is better than a basic SVM.

4. Experiments

The study requires solving a double problem by the method of reference vectors, comparison with the primary problem and classification of the data set.

Achieving this goal involves solving specific tasks:

- determine the problem of the method of support vectors for the dual problem;
- compare dual SVM and primal;
- analyze the algorithm of the method;
- apply it in practice.

The Iris dataset was chosen for the implementation of the support vector method. This is a well-known set of data used in the area of machine learning.

Dataset attribute information:

1. Sepallength.
2. Sepalwidth.
3. Petallength.
4. Petalwidth.
5. Classes:

- Iris Setosa;
- Iris Versicolour;
- Iris Virginica.

The best way to analyze a large data set is to visualize it. Data visualization refers to the approaches used to understand data through visual representation.

The main purpose of visualization – interpretation of a large data set into visual graphics to easily understand complex data relationships and quickly get an imagination of the dataset.

The histogram shown in Figure 2 shows the frequencies of the data set, which can be used to understand the trend of length / width of the petals for each type of plant.

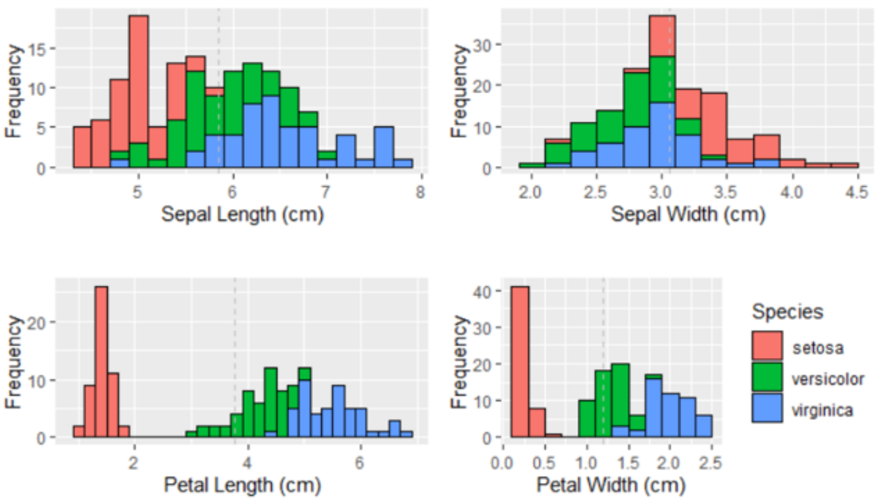

Figure 2: Histogram of frequencies

From Figure 2 it is seen that the type of petal virginica is much larger than others and sepals are mostly too, setosa has the smallest size and small range of values, versicolor - medium in size.

The box plot (Figure 3) shows the distribution of data by quartiles, the average values and statistical emissions are highlighted. The vertical lines (whiskers) drawn to the rectangles reflect the variability of the values outside the upper and lower quartiles, any point on these lines is considered a statistical ejection. The other two species are larger and have larger tendrils, which characterize a large scope.

The heatmap (Figure 4) shows the levels of correlation between attributes and their linear dependencies:

- (-0.09; 0.0)(0.0; 0.09) - linear independence;
- (-0.3; -0.1)(0.1; 0.3) - low linear dependence;
- (-0.5; -0.3) (0.3; 0.5) – medium linear dependence;
- (-1.0; -0.5) (0.5; 1.0) - high linear dependence.

The values of the length of the sepals depend entirely on the values of its width, or vice versa. In the case of a petal, the length and width are independent of each other. It is also interesting that the length of the petal depends on the length and width of the sepals.

In Figure 5 you can see the linear separation between classes.

Setosa is linearly separated from Versicolor and Virginica, and Versicolor and Virginica are not linearly separated. It means that in the first case it is necessary to apply SVM with a hard margin, and in the second - with soft.

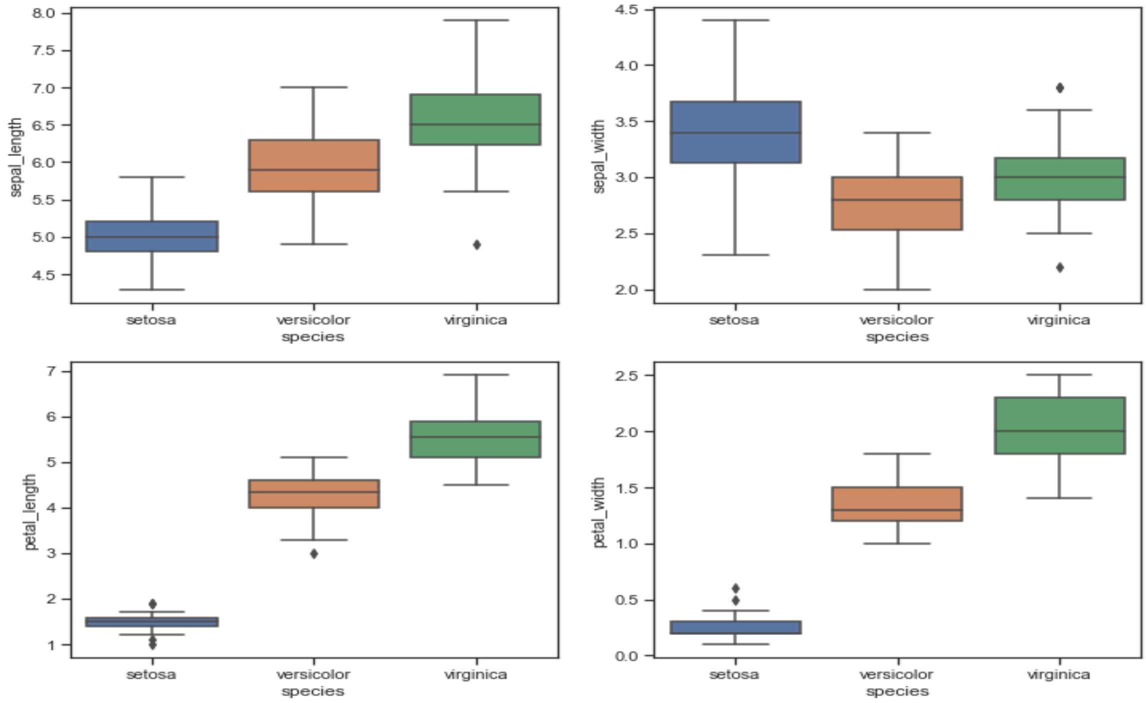

Figure 3: Box plot

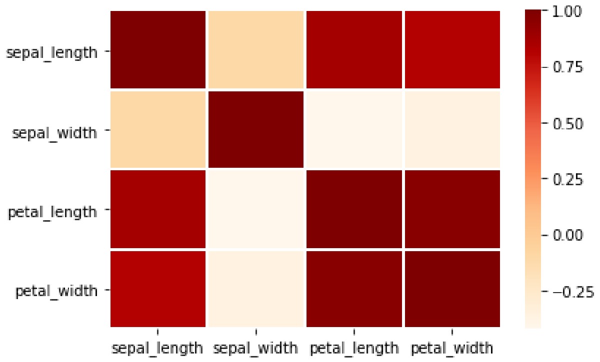

Figure 4: Heatmap

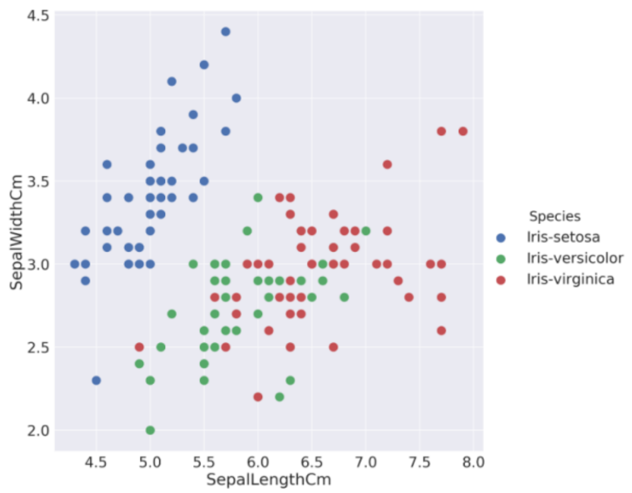

Figure 5: Scatter plot

4.1. Statistical analysis

Table 2
Statistic of the iris dataset

	Formula	sepallength	sepalwidth	petallength	petalwidth
Amount	n	150	150	150	150
Mean value	$\frac{\sum_{i=1}^{n} x_i}{n}$	5.84	3.057	3.758	1.19
Dispersion	$\frac{\sum_{i=1}^{n} (x_i - \overline{x}_j)^2}{n}$	0.6724	0.19	3.115	0.578
Standard deviation	$\sqrt{\frac{\sum_{i=1}^{n} (x_i - \overline{x}_j)^2}{n}}$	0.828	0.436	1.765	0.762

From this statistical in the Table 2 it is seen: that the size of the sepals mostly is larger than the size of the petal. Also, since the variance and standard deviation characterize the scattering of values around the distribution center, it can be concluded that the variation of petal size values between plant species is much larger than for the sepals.

5. Results

Two arbitrary points from the iris dataset are selected for analysis, and the optimal hyperplane is calculated.

Let first arbitrary point – $A(12,4.8)$, second – $B(100,6.3)$, where x – index in the dataset, y – length of the sepal (Figure 6).

The point A will be considered negative, B – positive, thus $yA = 1$, $yB = -1$.

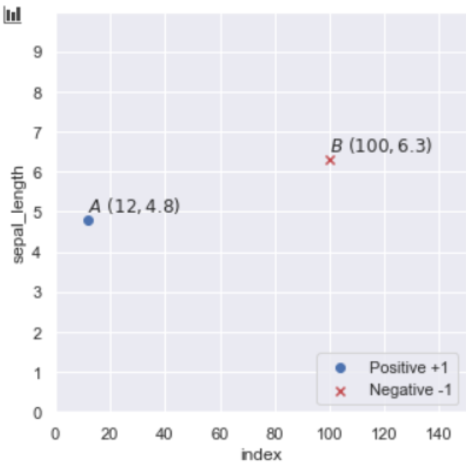

Figure 6: Visualization of selected points

You need to find the optimal separate hyperplane.

It is known that any hyperplane can be described as (equation 11):

$$wx + b = 0, \tag{11}$$

where w – normal vector to the hyperplane and $\frac{b}{\|w\|}$ - perpendicular distance from the hyperplane to the origin.

To find $\|w\|$ and b dual form should be introduced. It contains a quadratic objective function with constraints (equation 12).

$$\max_a L_D = \sum_{i=1}^{L} a_i - \frac{1}{2}\sum_{i,j} a_i a_j y_i y_j \langle x_i, x_j \rangle, \qquad (12)$$

if $\sum_{i=1}^{L} a_i y_i = 0, \quad a_i \geq 0 \quad \forall \quad i$

After data substitution:

$$\max_a L_D = \sum_{i=1}^{L} a_i - \frac{1}{2}\sum_{i,j} a_i a_j y_i y_j \langle x_i, x_j \rangle = a_1 + a_2 - \frac{1}{2}(a_1 a_1 * 1 * 1 * \left\langle \begin{pmatrix} 12 \\ 4.8 \end{pmatrix}, \begin{pmatrix} 12 \\ 4.8 \end{pmatrix} \right\rangle +$$

$$+ 2 * a_1 a_2 * 1 * (-1) * \left\langle \begin{pmatrix} 12 \\ 4.8 \end{pmatrix}, \begin{pmatrix} 100 \\ 6.3 \end{pmatrix} \right\rangle + a_2 a_2 * (-1) * (-1) * \left\langle \begin{pmatrix} 100 \\ 6.3 \end{pmatrix}, \begin{pmatrix} 100 \\ 6.3 \end{pmatrix} \right\rangle) =$$

$$= a_1 + a_2 - \frac{1}{2}(167.04 a_1^2 - 2460.48 a_1 a_2 + 10039.69 a_2^2)$$

Using the Lagrange equation we can solve this problem:

$$L(X, \gamma) = x_1 + x_2 - \frac{1}{2}(167.04 x_1^2 - 2460.48 x_1 x_2 + 10039.69 x_2^2 + \gamma * (x_1 - x_2 - 0).$$

Under the condition of the extremum of the Lagrange function, we equate the partial derivatives to zero.

Built system:

$$\begin{cases} \frac{\partial L}{\partial x_1} = -167.04 x_1 + 1230.24 x_2 + \gamma + 1 = 0 \\ \frac{\partial L}{\partial x_2} = 1230.24 x_1 - 10039.69 x_2 - \gamma + 1 = 0 \\ \frac{\partial L}{\partial \gamma} = -x_2 = 0 \end{cases}$$

Solving this system of equations by the Gaussian method:

$$\begin{pmatrix} -167 & 1230 & 1 \\ 1230 & -10040 & -1 \\ 1 & -1 & 0 \end{pmatrix} = \begin{pmatrix} -1 \\ -1 \\ 0 \end{pmatrix}$$

- multiply the first row by 1230;
- multiply the second row by 167;
- add the second row to the first.

$$\begin{pmatrix} 0 & -163780 & 1063 \\ 1230 & -10040 & -1 \\ 1 & -1 & 0 \end{pmatrix} = \begin{pmatrix} -1397 \\ -1 \\ 0 \end{pmatrix}$$

- multiply the third row by (-1230);
- add the third row to the second.

$$\begin{pmatrix} 0 & -163780 & 1063 \\ 0 & -8810 & -1 \\ 1 & -1 & 0 \end{pmatrix} = \begin{pmatrix} -1397 \\ -1 \\ 0 \end{pmatrix}$$

- multiply the first row by (-0.0538);
- add the second row to the first.

$$\begin{pmatrix} 0 & 0 & -58.181 \\ 0 & -8810 & -1 \\ 1 & -1 & 0 \end{pmatrix} = \begin{pmatrix} 74.147 \\ -1 \\ 0 \end{pmatrix}$$

This system can now be written as:

$$\begin{cases} -58.181\gamma = 74.147 \\ -8810x_2 - \gamma = -1 \\ x_1 - x_2 = 0 \end{cases}$$

Hence:

$$\gamma = \frac{74.147}{-58.181} = -1.274$$

$$x_2 = \frac{-1-(-1)*(-1.274)}{-8810} = \frac{-2.274}{-8810} = 0.000258$$

$$x_1 = \frac{0-(-1)*0.000258}{1} = \frac{0.000258}{1} = 0.000258$$

So the Lagrange multipliers are equal to:

$$a_1 = x_1 = 0.000258 = \frac{25}{96837}$$

$$a_2 = x_2 = 0.000258 = \frac{25}{96837}$$

Calculating w and b:

$$w = \sum_{i=1}^{L} a_i y_i x_i = \frac{25}{96837} *1* \begin{pmatrix} 12 \\ 4.8 \end{pmatrix} + \frac{25}{96837} *(-1)* \begin{pmatrix} 100 \\ 6.3 \end{pmatrix} = \begin{pmatrix} \frac{300}{96837} \\ \frac{120}{96837} \end{pmatrix} - \begin{pmatrix} \frac{2500}{96837} \\ \frac{157.5}{96837} \end{pmatrix} = \begin{pmatrix} \frac{2200}{96837} \\ \frac{37.5}{96837} \end{pmatrix}$$

$$b = y_s - \sum_{m \in S}^{L} a_m y_m \langle x_m, x_s \rangle = 1 - (\frac{25}{96837} * \left\langle \begin{pmatrix} 12 \\ 4.8 \end{pmatrix}, \begin{pmatrix} 12 \\ 4.8 \end{pmatrix} \right\rangle - \frac{25}{96837} * \left\langle \begin{pmatrix} 12 \\ 4.8 \end{pmatrix}, \begin{pmatrix} 100 \\ 6.3 \end{pmatrix} \right\rangle) =$$

$$= \frac{96837}{96837} - \left(\frac{4176}{96837} - \frac{30756}{96837} \right) = \frac{96837}{96837} + \frac{26580}{96837} = \frac{123417}{96837}$$

Returning to the representation of the hyperplane and substitute the numbers:

$$wx + b = 0 \Rightarrow w_1 x + w_2 y + b = 0$$

$$-\frac{2200}{96837} x - \frac{37.5}{96837} y + \frac{123417}{96837} = 0$$

Figure 7 shows the hyperplane that best classifies our data, and theoretically all positive points will be on the left and negative points on the right. Dark blue color indicates the hyperplane itself, dotted lines form a margin.

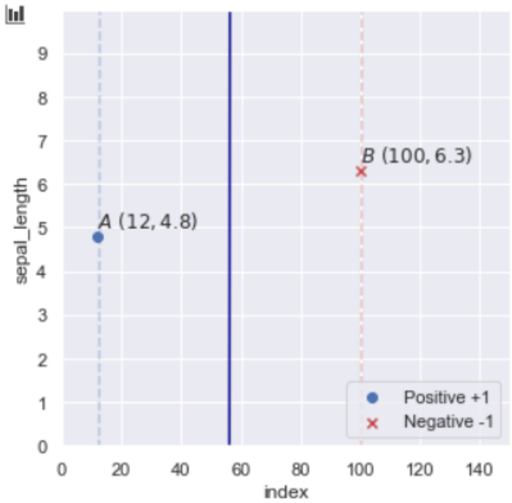

Figure 7: Visualization of the hyperplane

Comparing the result of the program and using the SVM of the sklearn library, the following result of the program is given:

$w = [[-0.02272067 \quad -0.00038728]]$

$b = [1.27450707]$

Indices of support vectors $= [1 \quad 0]$;

Support vectors $= [[100 \quad 6.3][12 \quad 4.8]]$;

Number of support vectors for each class $= [1 \quad 1]$;

Coefficients of the support vector in the decision function $= [[0.00025819 \quad 0.00025819]]$.

The values of w and b are calculated manually and coincide with the help of the program, so the calculations are correct. Also, since we have only two points in this case, they act as reference vectors, one for each class. The coefficients of these reference vectors in the objective function correspond to Lagrange factors calculated manually.

Figure 8 shows the calculated results using the program for the entire dataset:

Figure 8: Visualization of the dataset

Figure 8 shows the sepal length of two linearly separated plant species of the iris dataset - y, which must be classified using the SVM algorithm programmatically. We see that the data are linearly separated and easy to classify. Consider the problem is to find the optimal solution.

The result of the program:

$w = [[0.03919022 \quad 0.00099897]]$

The value of the normal vector to the hyperplane

$b = [-2.9253159]$

Bias value

Indices of support vectors $= [49 \quad 50]$;

Indices of support vectors

Support vectors $= [[49 \quad 5][100 \quad 6.3]]$;

Support vectors

Number of support vectors for each class $= [1 \quad 1]$;

Number of support vectors for each class

Coefficients of the support vector in the decision function $= [[0.00076844 \quad 0.00076844]]$.

Values of Lagrange multipliers.

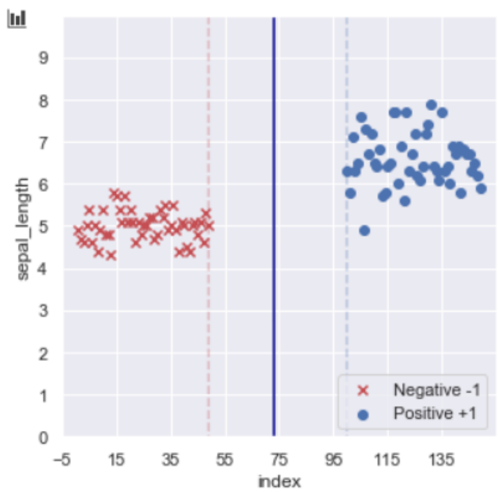

Figure 9: Visualization of the hyperplane

In Figure 9 two types of plants are classified by the hyperplane calculated programmatically. Since one of the parameters of the classification is the index, and iris dataset takes into account the indexation, namely (setosa - (1; 50), virginica - (50; 100), versicolor - (100; 150)), so the support vectors and their number for each class are found correctly. Plants with an index less than 75 will belong to setosa, and more - to versicolor. But it is necessary to remember about possible cases where the sepal_length value will be considered.

6. Conclusion

The dual method of support vectors is to solve the Lagrange problem, with found Lagrange multipliers it is easy to calculate the normal vector w and draw a separate hyperplane.

Solving the primary problem, it is obtained the optimal w, but nothing about the Lagrange multipliers. To classify the point x, it is needed to clearly calculate the scalar product $w^T x$, which can be very expensive. Solving the dual problem: obtained Lagrange factors (where $a_i = 0$ for all but a few points - support vectors). This problem is very efficiently calculated if there are few support vectors. Also, with a scalar product that includes only data vectors, it is possible to use kernel trick for nonlinear problems.

Dual SVM in nonlinear problems is more stable and faster than the primary because it performs fewer kernel estimates.

Using the Lagrange function helps to distribute the data linearly. The kernel trick is used to separate data in different ways, but not line.

SVM can be successfully used to control complex electromechanical systems, it can ensure the adaptability of control algorithms, perform the functions of an observer, an identifier of unknown parameters, a reference model, it can be used to control complex nonlinear objects.

7. References

[1] V. Estivill-Castro, I. Lee, Amoeba: Hierarchical clustering based on spatial proximity using Delaunay diagram, in: 9th Intern. Symp. on spatial data handling, Beijing, China, 2000, pp. 26–41.

[2] C. Boehm, K. Kailing, H. Kriegel, P. Kroeger, Density connected clustering with local subspace preferences, IEEE Computer Society, Proc. of the 4th IEEE Intern. conf. on data mining, Los Alamitos, 2004, pp. 27–34.

[3] N. Boyko, K. Shakhovska, Information system of catering selection by using clustering analysis, in: 2018 IEEE Ukraine Student, Young Professional and Women in Engineering Congress (UKRSYW), Kyiv, Ukraine, 2018, pp.7-13.

[4] D. Harel, Y. Koren, Clustering spatial data using random walks, in: Proc. of the 7th ACM SIGKDD Intern. conf. on knowledge discovery and data mining, San Francisco, California, 200, pp. 281–286.

[5] A.K. Tung, J. Hou, J. Han, Spatial clustering in the presence of obstacles, in: The 17th Intern. conf. on data engineering (ICDE'01), Heidelberg, 2001, pp. 359–367.

[6] N. Boyko, A. Bronetskyi, N. Shakhovska, Application of Artificial Intelligence Algorithms for Image Processing, in: CEUR. Workshop Proceedings of the 8th International Conference on "Mathematics. Information Technologies. Education", MoMLeT&DS-2019, Vol-2386 urn: nbn: de: 0074-2386-1, Shatsk, Ukraine, June 2-4, 2019, pp. 194-211.

[7] R. Agrawal, J. Gehrke, D. Gunopulos, P. Raghava, Automatic sub-space clustering of high dimensional data, Data mining knowledge discovery, vol. 11(1), 2005, pp. 5–33.

[8] M. Ankerst, M. Ester, H.-P. Kriegel, Towards an effective cooperation of the user and the computer for classification, in: Proc. of the 6th ACM SIGKDD Intern. conf. on knowledge discovery and data mining, Boston, Massachusetts, USA, 2000, pp. 179–188.

[9] C. Zhang, Y. Murayama, Testing local spatial autocorrelation using. Intern. J. of Geogr. Inform. Science, vol. 14, 2000, pp. 681–692.

[10] V. Estivill-Castro, I. Lee, Amoeba: Hierarchical clustering based on spatial proximity using Delaunay diagram, in: 9th Intern. Symp. on spatial data handling, Beijing, China, 2000, pp. 26–41.

[11] D. Guo, D.J. Peuquet, M. Gahegan, ICEAGE: Interactive clustering and exploration of large and high-dimensional geodata, Geoinformatica, vol. 3, N. 7, 2003, pp. 229–253.

[12] N. Boyko, O. Basystiuk, Comparison Of Machine Learning Libraries Performance Used For Machine Translation Based On Recurrent Neural Networks, in: 2018 IEEE Ukraine Student, Young Professional and Women in Engineering Congress (UKRSYW), Kyiv, Ukraine, 2018, pp.78-82.

[13] C. Aggarwal, P. Yu, Finding generalized projected clusters in high dimensional spaces, in: ACM SIGMOD Intern. conf. on management of data, 2000, pp. 70–81.

[14] R. Thanki, S. Borra, Application of Machine Learning Algorithms for Classification and Security of Diagnostic Images, Machine Learning in Bio-Signal Analysis and Diagnostic Imaging, 2019, pp. 273-292.

[15] D.J. Peuquet, "Representations of space and time". N. Y.: Guilford Press (2000)

[16] C.M. Procopiuc, M. Jones, P.K. Agarwal, T.M. Murali, A Monte Carlo algorithm for fast projective clustering, in: Intern. conf. on management of data, ACM SIGMOD, Madison, Wisconsin, USA, 2002, pp. 418–427.

[17] K. Chitra, Dr. D.Maheswari, A Comparative Study of Various Clustering Algorithms in Data Mining. International Journal of Computer Science and Mobile Computing, Vol.6 Issue.8, August 2017, pp. 109 –115.