Olga VESELSKA[1], Oleksandr PETROV[2], Anton PETROV[3],
Ruslana ZIUBINA[4]

METODY I ALGORYTMY KLASYFIKACJI DANYCH

Streszczenie: W pierwszej części artykułu przeprowadzono szczegółowe badanie analizy dyskryminacyjnej, opartej na koncepcji odległości między elementami oraz na kryterium bliskości skonstruowanej na podstawie obliczenia cosinusu kąta między dwoma wektorami. Również wszystkie konstrukcje bazujące na podanym przykładzie klasyfikacji tekstów. W drugiej części artykułu zbadano metodę wektorów wsparcia (SVM - Support Vector Machines), zawartą w zbiorze algorytmów nazywanych "nauczaniem nadzorowanym". Są one skutecznie wykorzystywane także w problemach klasyfikacji jako metoda analizy dyskryminacyjnej. W związku z tym w artykule szczegółowo omówiono problem klasyfikacji danych.

Słowa kluczowe: klasyfikacja danych, algorytmy genetyczne, wektory wspierające, dane, przetwarzanie wstępne, funkcje dyskryminacyjne

METHODS AND ALGORITHMS OF DATA CLASSIFICATION

Summary: In the first part of the article conducted a detailed study of the discriminant analysis, based on the concept of distance between elements, and on the criterion of proximity constructed on calculating a cosine of the angle between two vectors. Also, all constructions relying on the example of texts classification provided. In the second part of the article investigated the method of support vectors (SVM - Support Vector Machines), included in the set of algorithms called as "supervised learning". They are effectively used in problems of classification as a method of discriminant analysis too. Thus, the article examined in detail the problem of data classification.

Keywords: data classification, genetic algorithms, support vectors, data, preprocessing, discriminant functions

1. Using of genetic algorithms for creation of the vector classifiers

In this paragraph we will consider the discriminatory analysis based not on the concept of distance between elements, and on the criterion of proximity constructed on

[1] University of Bielsko-Biala, Department of Computer Science and Automatics, oveselska@ath.edu.pl
[2] AGH University of Science and Technology, Krakow, Poland, asp1951@gmail.com
[3] Federal State Budgetary Educational Institution of Higher Education “Kuban State Agrarian University named after I.T. Trubilin”, Krasnodar, Russia, anton.a.petrov@gmail.com
[4] University of Bielsko-Biala, Department of Computer Science and Automatics, rziubina@ath.edu.pl

calculating a cosine of the angle between two vectors. We are going to provide all constructions relying on the example of texts classification [1-5].

The first step consists in preprocessing of data – creating sets of statistics for the available classes. For the creation of the set of statistics all sets of word forms b^ν, $\nu = 0,..., M-1$ are consistently processed, belonging to one class $B = \{b^\nu\}_{\nu=0}^{M-1}$. On the set of word forms of each processed text b^ν the set of the unique (not repeating) word forms and their counters is under construction - (w_i^ν, n_i^ν) $(i = 0,..., N^\nu - 1)$. Here N^ν- quantity of unique word forms for the text b^ν. After that data for each document separately are normalized in the following form

$$\bar{n}_i^\nu = \frac{n_i^\nu}{\sqrt{\sum_{j=0}^{N^\nu-1}(n_j^\nu)^2}} (i = 0,..., N^\nu - 1).$$

Then, we arrange all words for each document in the same order (the word order is not essential, the main thing that words in each of structures $(w_i^\nu, n_i^\nu)(i = 0,..., N^\nu - 1)$ went in the same order) and we find the sum of all vectors $n_i(B) = \sum_{j=0}^{M-1} \bar{n}_i^\nu$ $(i = 0,..., N(B))$ (where $N(B)$ – a quantity of unique word forms for the class B in general) also we normalize it by its unit as follows

$$\bar{n}_i(B) = \frac{n_i(B)}{\sqrt{\sum_{j=0}^{N(B)}(n_j(B))^2}}.$$

For the received central point of the class we create the set of statistics, writing down in it values $\left(w_i(B), \bar{n}_i(B)\right)$ $(i = 0,..., N(B))$.

For creating the central vector of classes $\{B^\mu\}_{\mu=0}^{K-1}$ where each class B^μ is described by the central vector $(w_i(B^\mu), \bar{n}_i(B^\mu))(i = 0,..., N(B^\mu))$, it is necessary to find their sum, having summed up all coordinates from all vectors for each value of the word form, that is for the word form ω we receive the coordinate

$$n(\omega) = \sum_{\mu=0}^{K-1}\{\bar{n}_i(B^\mu) | \omega_i(B^\mu) = \omega, \quad i = 0,...N(B^\mu)\}.$$

Therefore, it is necessary to make the list of unique word forms on all central vectors of classes $\{B^\mu\}_{\mu=0}^{K-1}$ and to sum up their coordinates. The set consisting their unique (not repeating) word forms and their coordinates can be result

$$\left(w_i\left(\{B^\mu\}_{\mu=0}^{K-1}\right), n_i\left(\{B^\mu\}_{\mu=0}^{K-1}\right)\right) \ (i = 0,..., N(\{B^\mu\}_{\mu=0}^{K-1})),$$

where $N(\{B^\mu\}_{\mu=0}^{K-1})$ is a quantity of unique word forms of the set of classes $\{B^\mu\}_{\mu=0}^{K-1}$. It is necessary to normalize the received coordinates

$$\bar{n}_i(\{B^\mu\}_{\mu=0}^{K-1}) = \frac{n_i(\{B^\mu\}_{\mu=0}^{K-1})}{\sqrt{\sum_{j=0}^{N(\{B^\mu\}_{\mu=0}^{K-1})}(n_j(\{B^\mu\}_{\mu=0}^{K-1}))^2}},$$

and, the received vector $\left(w_i\left(\{B^\mu\}_{\mu=0}^{K-1}\right), \hat{n}_i\left(\{B^\mu\}_{\mu=0}^{K-1}\right)\right)$ $(i = 0,..., N(\{B^\mu\}_{\mu=0}^{K-1}))$ can be the central vector of the set $\{B^\mu\}_{\mu=0}^{K-1}$.

Ideally created classification of the vector method is such set of classes $\{B^\mu\}_{\mu=0}^{K-1}$, for which the following condition is satisfied: $\forall b \in B^\mu, \mu = 0,..., K-1$ the inequality takes place

$$\langle \bar{n}(b), \bar{n}(B^{\mu}) \rangle < \langle \bar{n}(b), \bar{n}(B^{\nu}) \rangle, \quad \nu \neq \mu. \tag{1}$$

Let's consider the vector Λ (control vector) of dimension $N(B^{\mu})$, which coordinates accept only the one or the other admissible values (zero)

$$\lambda_i = \begin{cases} 0 \\ 1 \end{cases}.$$

Through Λb let's designate the direct product of vectors Λ and b, that is

$$\Lambda b = (\lambda_0 \bar{n}_0(b), \lambda_1 \bar{n}_1(b), \ldots, \lambda_{N(B^{\mu})} \bar{n}_{N(B^{\mu})}(b)).$$

The control Λ let's call admissible on the class $B^{\mu} = \{b^k\}_{k=0}^{M-1}$, if the condition (2) is satisfied

$$\langle \overline{\Lambda \bar{n}}(b^k), \overline{\Lambda \bar{n}}(B^{\mu}) \rangle < \langle \overline{\Lambda \bar{n}}(b^k), \bar{n}(B^{\nu}) \rangle, \ \nu \neq \mu, \ k = 0,1,\ldots,M-1. \tag{2}$$

Admissible control vector Λ for which this inequality (2) and at the same time which is fulfilling the condition $\sum_{k=0}^{M-1} (\Lambda b^k)^2 \rightarrow \max$, is called the optimum.
If for $\nu \neq \mu$ the set of admissible controls is degenerated, the class $B^{\mu} = \{b^k\}_{k=0}^{M-1}$ is defined incorrectly, i.e. it is inseparable from the class B^{ν} .
The problem of finding the optimum control by classical methods is rather difficult therefore we will apply genetic algorithms to its decision [1]).
For that matter, the single-point crossing over (single-point crossover) is used. It is modeled as follows: Let there are two parent individuals with chromosomes $X = \{x_i, i \in \{0,\ldots,L\}\}$ and $Y = \{y_i, i \in \{0,\ldots,L\}\}$. In a random way the point in the chromosome is defined (discontinuity point) in which both chromosomes are divided into two parts and exchange them. After processing reproduction we can get mutation. It is reached because accidentally chosen gene in the chromosome changes.

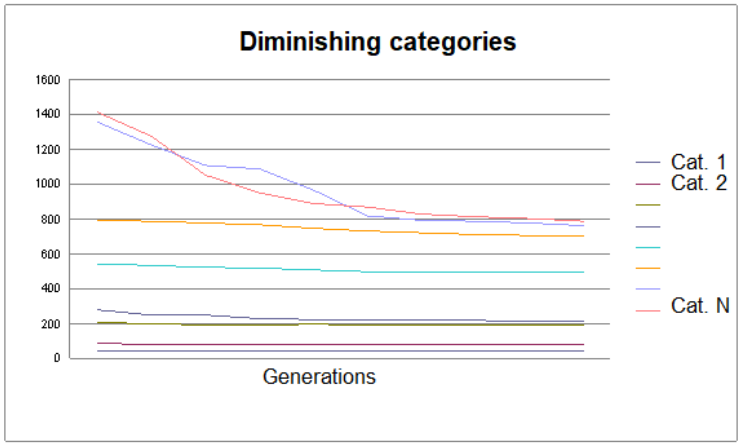

Figure 1. The chart of reduction of dimension of categories when using genetic algorithms

For creating new population, we used the elite selection. Intermediate population which includes both parents, and their descendants are created. Members of this population are evaluated, and behind that N of the best get out of them (suitable) which will enter the next generation.

The result of applicating the genetic algorithm to the problem of reducing class dimension, is given in the figure 1.

Let's notice that the vector method as a criterion of quality uses the size of the scalar product of basis vectors, thus, the class of unit vectors (documents) is limited on the sphere by the circle with the center at the end of the central vector of the class. As sphere cuts on the circle cannot densely pack all surface of the single sphere, there is the point set (basis vectors) which cannot essentially get to one class. Thus, there is the need to break the point set on the single sphere so that elements of this splitting densely packed all surface of the single sphere, that is allowed to classify any document unambiguously.

For any center of system $\{A\}$ it is possible to specify area of space which all points are closer to this center, than to any other center of system. Such area is called the Voronoi polyhedron or Voronoi diagram. The Voronoi diagram usually carry to the polyhedron also its outer surface. In three-dimensional space the Voronoi diagram for any i center of systems $\{A\}$ is the convex polyhedron, in two-dimensional space it is the convex polygon. Formally Voronoi polygons T_i in R^2 are defined as follows:

$$T_i = \{x \in R^2 : d(x, x_i) < d(x, x_j)\ \forall j \neq i\}.$$

where d is a distance function.

Creating an approximation relies on fundamental property for randomly selected n set points on the plane S. For any node from n on the plane, there is the great number of natural neighbors N. The concept of natural neighbors is closely connected with splitting the Voronoi diagram cells. For the nonempty Voronoi cell $V(R)$, where $R \subset S$ the natural neighbors for a vertex of Delon's triangles $r \in R$, are points incidental to $V(R)$.

The two-dimensional Voronoi polyhedron (polygon on the plane) is shown on the (Fig. 2). The lines Voronoi which generated edges at the polygon are called the forming lines and the relevant centers of system are the geometrical neighbors of this center A. Among geometrical neighbors (natural) we can distinguish two kinds of them. For the first kind - the bisecting point of a segment connecting it to the central node lies on the verge of the Voronoi polyhedron. For the second kind the bisection point is out of the edge and, therefore, out of the polyhedron.

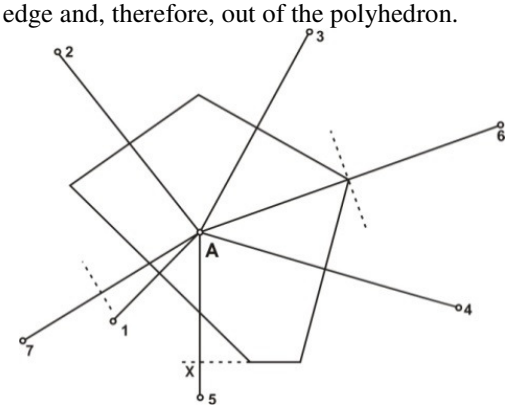

Figure 2. The Voronoi polyhedron (polygon) for the center A in two-dimensional system

Polyhedrons Voronoi, the systems constructed for each center $\{A\}$, give the mosaic of polyhedrons - splitting Voronoi points (see Fig.3). Polyhedrons Voronoi systems $\{A\}$ do not enter each other and fill the space, being adjacent on the whole edges. Splitting space into Voronoi polyhedrons unambiguously is defined by system $\{A\}$ and converse uniquely defines it.

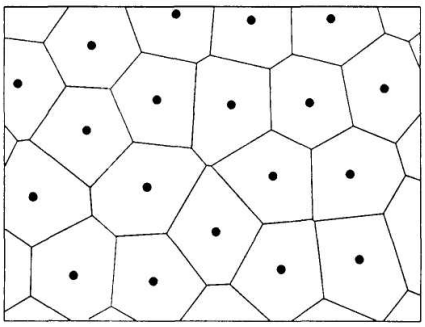

Figure 3. The Voronoi chart on the plane

Using the design of Voronoi diagrams in relation to points on the multidimensional single sphere, we receive splitting all basis vectors of documents into natural classes. Borders of classes will be the hyper planes dividing spherical Voronoi polyhedrons. Points on the single sphere, which in relation to all hyperplanes limiting this class lie on the same side of sphere, as the central vector of this class, will belong to one class. Let classes of documents be checked for the splitting correctness C_ν and C_μ. For corresponding basis vector (the central vectors) $\hat{C}_\nu$, $\hat{C}_\mu$, we build the difference vector

$$\vec{\Delta}_{\nu,\mu} = \hat{C}_\nu - \hat{C}_\mu = \{\hat{n}^\nu(w_i) - \hat{n}^\mu(w_i)\}$$

and sum vector

$$\vec{\Xi}_{\nu,\mu} = \frac{1}{2}(\hat{C}_\nu + \hat{C}_\mu) = \frac{1}{2}\{\hat{n}^\nu(w_i) + \hat{n}^\mu(w_i)\}.$$

The half-sum of vectors begin coordinates of this vector in number. Let's designate it through $\Xi_{\nu,\mu}$. Let's carry out through the point $\Sigma_{\nu,\mu}$ the plane with the normal vector $\vec{\Delta}_{\nu,\mu}$

$$\Omega_{\nu,\mu} = \langle\vec{\Delta}_{\nu,\mu} \cdot (P - \Xi_{\nu,\mu})\rangle = 0\ . \tag{3}$$

This plane splits classes. In order to the method correctly splits classes, it is necessary that all points (documents) of one class are on the one side of the planes, that is if $b \in C_\nu$, then

$$\langle\vec{\Delta}_{\nu,\mu} \cdot (\hat{C}_\nu - \Xi_{\nu,\mu})\rangle\langle\vec{\Delta}_{\nu,\mu} \cdot (\hat{b} - \Xi_{\nu,\mu})\rangle \geq 0.$$

Points in which this condition is not satisfied need to be considered relating to belonging to category C_μ.

For the solving this problem the following method can be taken into account. Let's consider categories C_ν and C_μ. Let's split them by the plane (3), and all points lying on the one side, we will collect in new two categories C_ν^* and C_μ^*.
Let

$$d(B,\Omega_{\nu,\mu}) = \frac{|\langle\vec{\Delta}_{\nu,\mu}\cdot(B-\Xi_{\nu,\mu})\rangle|}{|\vec{\Delta}_{\nu,\mu}|}$$

be a distance from the point $B=\{b_i\}$ to the plane $\Omega_{\nu,\mu}$.
If the condition is satisfied (that is, after cutting off of data both categories are removed from each other)

$$\begin{cases} d(C_\nu^*,\Omega_{\nu,\mu}) - d(C_\nu,\Omega_{\nu,\mu}) > 0, \\ d(C_\mu^*,\Omega_{\nu,\mu}) - d(C_\mu,\Omega_{\nu,\mu}) > 0, \end{cases}$$

that categories C_ν and C_μ have nonempty crossing $\tilde{C}$, which can be defined as follows, $b \in \tilde{C}$ if $b \in C_\nu$ and at the same time

$$\langle\vec{\Delta}_{\nu,\mu}\cdot(\hat{C}_\nu-\Xi_{\nu,\mu})\rangle\langle\vec{\Delta}_{\nu,\mu}\cdot(\hat{b}-\Xi_{\nu,\mu})\rangle < 0,$$

or, if $b \in C_\mu$,

$$\langle\vec{\Delta}_{\nu,\mu}\cdot(\hat{C}_\mu-\Xi_{\nu,\mu})\rangle\langle\vec{\Delta}_{\nu,\mu}\cdot(\hat{b}-\Xi_{\nu,\mu})\rangle < 0.$$

It is natural that the problem of classes dimension reduction is also urgent for the method constructed on Voronoi diagrams.
Comparative analysis of application of different discriminatory analyses to test base of documents [2] is given in the following figures (see fig.4; 5; 6).

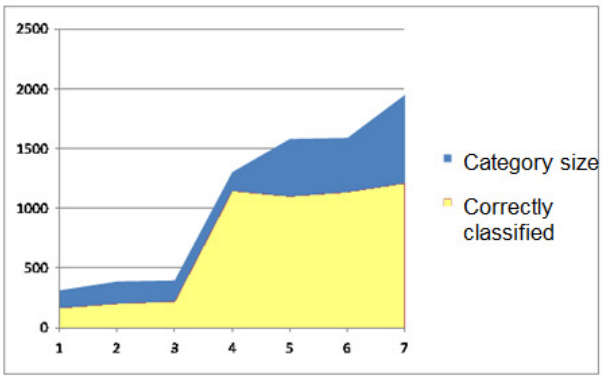

Figure 4. The result of applicating the algorithm of Bayes

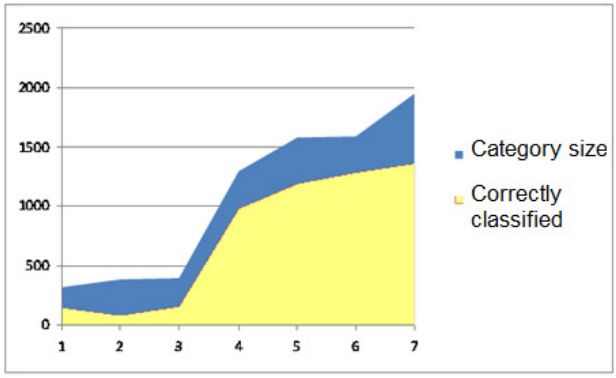

Figure 5. The result of applicating the vector algorithm

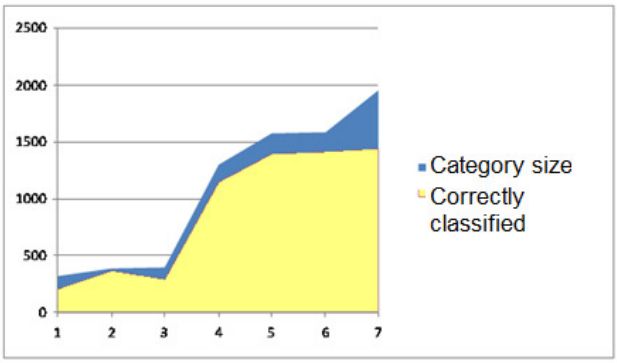

Figure 6. The result of applicating the algorithm based on Voronoi diagrams

Thus, for given test base on condition of hit in the class not less than 90% of documents, dimension of classes from 10% was succeeded to reduce to 50%.

2. Support vector machines

The method of support vectors (SVM) is included in the set of algorithms called as "supervised learning". They are effectively used in problems of classification. In 1963 [6] proposed an algorithm which if the data are linearly separable then finds the maximal margin between the fixed boundary and the nearest points of each class. The next work by is the cornerstone of SVM [6-9]. SVM method belongs to family of linear qualifiers [10].

In the previous paragraph we considered the simplest discriminant functions realizing the linear qualifier (see Fig. 7). It can be written down in the form $g(x) = w^t x + w_0$, where

$$g(x) > 0 \Rightarrow x \in \text{Class}[1] \text{and } g(x) < 0 \Rightarrow x \in \text{Class}[2].$$

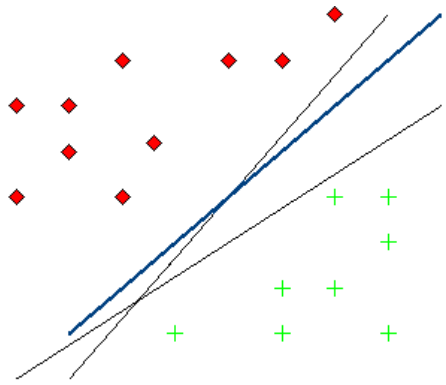

Figure 7. Symmetric determinant functions

Thus, the discriminant function is described by the equation $g(x)=0$. Distance between the point x and the point of dividing function $g(x) = 0$ is equal to $\frac{|w^t x+w_0|}{\|w\|}$.

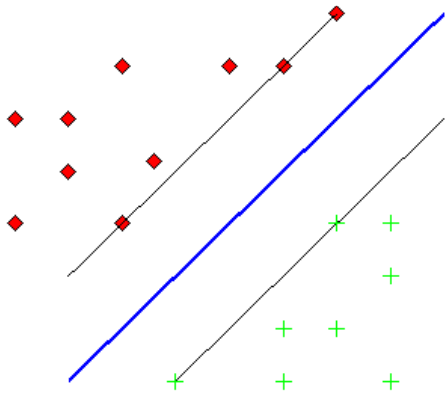

Figure 8. The maximum dividing corridor

Let x_i lie on short circuit of border, that is $|w^t x_i + w_0| = 1$. Border width of the dividing margin, is chosen as wide as possible (see Fig. 8). Considering that short strip of border meets the condition $|w^t x_i + w_0| = 1$, then distance from x_i to $g(x) = 0$ is

$$\frac{|w^t x+w_0|}{\|w\|} = \frac{1}{\|w\|}, \tag{4}$$

thus, width of the dividing strip is equal $\frac{2}{\|w\|}$ (see Fig. 9).

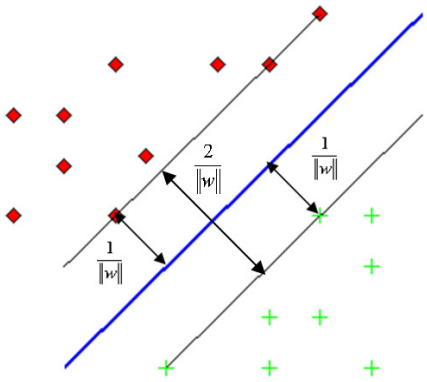

Figure 9. Illustration of creating support vectors

To exclude points from the dividing margin, we will write out the condition of belonging to classes

$$\begin{cases} w^t x_i + w_0 \geq 1 & \text{if } x_i \text{ belongs to class 1} \\ w^t x_i + w_0 \leq -1 & \text{if } x_i \text{ belongs to class 2} \end{cases}$$

Let's enter the index function

$$\begin{cases} u_i = 1 & \text{if } x_i \text{ belongs to class 1} \\ u_i = -1 & \text{if } x_i \text{ belongs to class 2} \end{cases}$$

Thus, the problem of estimating the dividing function generating the corridor of the greatest width is possible to write down as a problem of minimizing in the following form

$$J(w) = \frac{1}{2}\|w\|^2 \rightarrow \min \tag{5}$$

under the condition $u_i(w^t x_i + w_0) \geq 1$ for all i.
As the objective function is a square function, so this task has the only one resolve.
According to Kuhn-Tucker's theorem the condition (5) is equivalent to the following task

$$L(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j u_i u_j x_i^T x_j \rightarrow \max \tag{6}$$

provided that $\alpha \geq 0$ for all i and $\sum_{i=1}^{n} \alpha_i u_i = 0$, where $\alpha = \{\alpha_1,\ldots,\alpha_n\}$ are new variables. Let's rewrite $L(\alpha)$ in the matrix form

$$L(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix}^T H \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix},$$

Where coefficients of the matrix H are calculated as follows

$$H_{i,j} = u_i u_j x_i^T x_j \ .$$

The task $L(\alpha) \rightarrow \max$ is solved by methods of quadratic programming.
After finding the optimum $\alpha = \{\alpha_1,\ldots,\alpha_n\}$ for each i it is verified the conditions

- $\alpha_i = 0$ (it corresponds i is not a support vector);
- $\alpha_i \neq 0$ and $u_i(w^t x_i + w_0 - 1) = 0$ (it corresponds i is a support vector);

Then w from the ratio (4) can be found $w = \sum_{i=1}^{n} \alpha_i u_i x_i$ and the value w_0 is calculated considering that for any $\alpha_i > 0$ and $\alpha_i[u_i(w^t x_i + w_0) - 1] = 0$

$$w_0 = \frac{1}{u_i} - w^t x_i.$$

Then, at last, the discriminant function is received

$$g(x) = (\sum\{\alpha_i u_i x_i \mid x_i \in S\})^T x + w_0.$$

Notice that summing is carried out not on all vectors but only on the set S which represents the set of support vectors i.e. $S = \{x_i | \alpha_i \neq 0\}$.

Unfortunately, the described above algorithm is implementable only for linearly separable sets In practice this sets are not met frequently. In 1995 [6-7, 10] proposed modified algorithm for solving the problem for nonlinear separable sets [10-12]. Let's give the modernization of the algorithm for the case of nonlinearly separable sets.

In order to allow for misclassification in the model, it is entered additional variables ξ_i, which characterize the mistake size on each object of x_i. In the objective function, the penalty for the aggregate error is introduced in the following form:

$$\begin{cases} \frac{1}{2}\|w\|^2 + \lambda \sum_{i=1}^{n} \xi_i \rightarrow \min, \\ u_i(w^t x_i + w_0) \geq 1 - \xi_i, \ i = 1,\ldots,n, \\ \xi_i \geq 0, \ i = 1,\ldots,n, \end{cases} \tag{7}$$

here λ is the parameter specifying the cost of misclassifications. It allows to govern the relation between maximizing width of the dividing strip and minimization of the aggregate error [12-13].

Penalty size ξ_i for the corresponding object x_i depends on the arrangement of the object in dividing strip. So, if x_i lies on the opposite side of discriminant function, then the penalty size is $\xi_i > 1$. If x_i lies in the dividing strip, but on the same side of discriminant function as the class, then the corresponding weight can take a value $0 < \xi_i < 1$. For the ideal separable case the penalty size is taken as $\xi_i < 0$ (see Fig. 8.4).

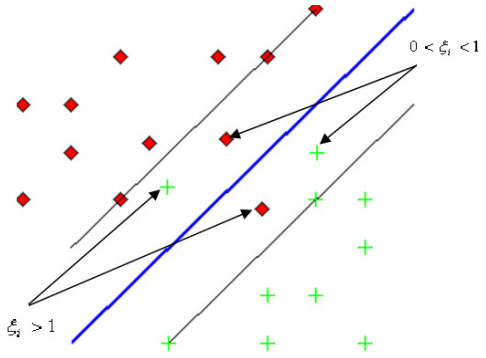

Figure 10. Points to which penalties are applied

Then the task (7) can be rewritten in the form below [14]

$$J(w,\xi_1,\ldots,\xi_n) = \frac{1}{2}\|w\|^2 + \beta\sum_{i=1}^{n} I(\xi_i > 0) \;\rightarrow\; \min\,, \tag{8}$$

that is in the way of minimization elements which do not represent the ideal case participate. Here

$$I(\xi_i > 0) = \begin{cases} 1, & \xi_i > 0, \\ 0, & \xi_i \le 0, \end{cases}$$

when fulfilling conditions $u_i(w^t x_i + w_0) \ge 1 - \xi_i$ and $\xi_i \ge 0$. In the formula 8.5 the constant β is the weight considering the bandwidth. If β is not enough, then we allow to arrange relatively many elements in the imperfect position, that is, in the dividing strip. If β is big, then we demand existence of small quantity of elements in the imperfect position, that is, in the dividing strip.
Unfortunately, in (8), the problem of minimization is rather difficult, in view of discontinuity $I(\xi_i)$. Instead we will consider a value minimization $J(w,\xi_1,\ldots,\xi_n) = \frac{1}{2}\|w\|^2 + \beta\sum_{i=1}^{n}\xi_i$ with restrictions for all *i* in the following form

$$\begin{cases} u_i(w^t x_i + w_0) \ge 1 - \xi_i \\ \xi_i \ge 0 \end{cases}.$$

Using Kuhn-Tucker's theorem, from here we receive

$$L(\alpha) = \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j u_i u_j x_i^T x_j \;\rightarrow\; \max \tag{9}$$

Provided that $0 \le \alpha_i \le \beta,\ \forall i$ and $\sum_{i=1}^{n}\alpha_i u_i = 0$.
From the ratio (7) we can find $w = \sum_{i=1}^{n}\alpha_i u_i x_i$. The value w_0 it is also possible to find, considering that for all $i\ \ 0 \le \alpha_i \le \beta$ and $\alpha_i[u_i(w^t x_i + w_0) - 1] = 0$.
The other idea of the SVM method (in the case when a linear division of classes is impossible), is transition to space of bigger dimension in which such division is possible [10, 12, 15-16]).
For solving the nonlinear classification problem by the linear qualifier, it is necessary:

- to design data x in space of higher dimension by means of transformation φ(x).
- to find a symmetric discriminant function for data φ(x).
- the received nonlinear discriminant function can be written down in the following form

$$g(x) = w^t\varphi(x) + w_0.$$

The received symmetric discriminant function for two-dimensional data X can take a form:

$$g\begin{pmatrix} x^{(1)} \\ x^{(2)} \end{pmatrix} = [w_1 \quad w_1]\begin{bmatrix} x^{(1)} \\ x^{(2)} \end{bmatrix} + w_0.$$

The one-dimensional discriminant function for nonlinear separable data using the function $\varphi(x) = (x, x^2)$ is written as follows:

$$g(x) = w_1 x + w_2 x^2 + w_0.$$

The example is shown on Fig. 8.5. For transferring data in space of higher dimension it is used so-called kernel functions.

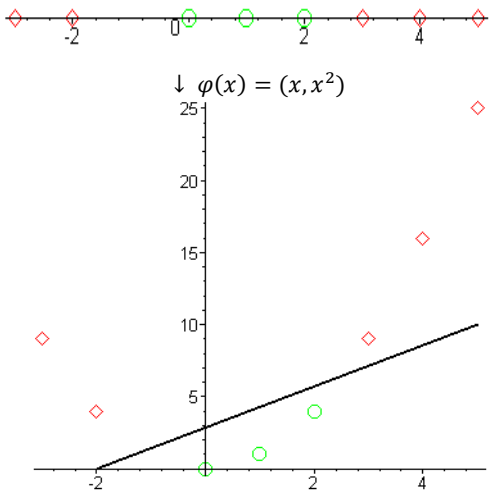

Figure 11. An example of linear division of sets upon transition to space of higher dimension

Let's go back to written above (9) the extremum problem of the method of support vectors in the following form

$$L(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i\alpha_j u_i u_j x_i^T x_j \rightarrow \max .$$

Let's notice that the optimization depends on the formula $x_i^T x_j$. If we transfer x_i to space of higher dimension using the display function $\varphi(x)$, then it is necessary to calculate the similar formula in space of higher dimension $\varphi(x_i)^T \varphi(x_j)$.
The idea of the method consists that it is necessary to find kernel function $K(x_i, x_j) = \varphi(x_i)^T \varphi(x_j)$ and to maximize the following objective function

$$L(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i\alpha_j u_i u_j K(x_i, x_j) \rightarrow \max.$$

Let's review the new example and take kernel function in the form$K(\mathrm{x,y}) = (x^T y)^2$. It is simple to find out the display $\varphi(x)$ corresponding to the kernel function.

$$K(\mathrm{x,\ y}) = (x^T\ y)^2 = ([x^{(1)} \quad x^{(2)}]\begin{bmatrix} y^{(1)} \\ y^{(2)} \end{bmatrix})^2 = (x^{(1)}y^{(1)} + x^{(2)}y^{(2)})^2 =$$

$$= (x^{(1)}y^{(1)})^2 + 2(x^{(1)}y^{(1)})(x^{(2)}y^{(2)}) + (x^{(2)}y^{(2)})^2 =$$

$$= [(x^{(1)})^2, \quad \sqrt{2}x^{(1)}x^{(2)}, \quad (x^{(2)})^2][(y^{(1)})^2, \quad \sqrt{2}y^{(1)}y^{(2)}, \quad (y^{(2)})^2]^T.$$

Thus, the display function can be written in the following form $\varphi(x) = [(x^{(1)})^2, \quad \sqrt{2}x^{(1)}x^{(2)}, \quad (x^{(2)})^2]$.

It is important to noticed that the choice of kernel function is rather difficult.

Let's review the example [8, 9].

Class [1]: x_1=[1,-1], x_2=[-1,1].

Class [2]: x_3=[1,1], x_4=[-1,-1].

It is illustrated on Fig. 12.

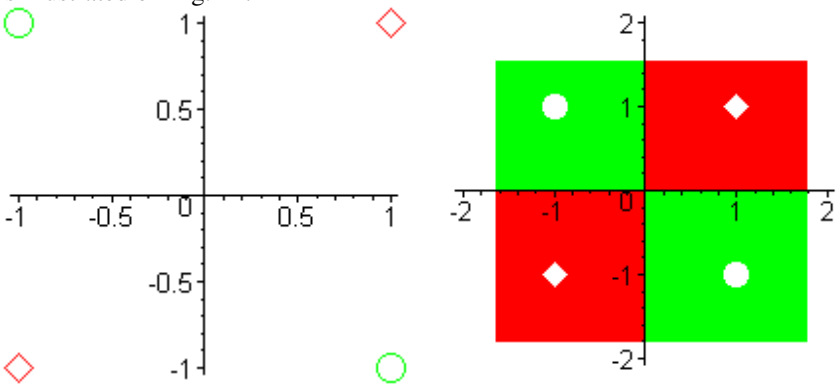

Figure 12. An example of linearly inseparable sets

For creation of nonlinear discriminant function, we use kernel function in the following form

$$K(x_i, x_j) = (x_i^T x_j + 1)^2.$$

The display function φ corresponding to the kernel function can be written as follows

$$\varphi(x) = [1, \quad \sqrt{2}x^{(1)}, \quad \sqrt{2}x^{(2)}, \quad \sqrt{2}x^{(1)}x^{(2)}, \quad (x^{(1)})^2, \quad (x^{(2)})^2].$$

Further it is necessary to maximize the objective function

$$L(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j u_i u_j (x_i^T x_j + 1)^2 \rightarrow \max$$

with restrictions

$$\alpha_i \geq 0, \; \alpha_1 + \alpha_2 - \alpha_3 - \alpha_4 = 0.$$

Let's rewrite the task in the following form

$$L(\alpha) = \sum_{i=1}^{4} \alpha_i - \frac{1}{4}\alpha^T H \alpha \, ,$$

where $\alpha = [\alpha_1 \quad \alpha_2 \quad \alpha_3 \quad \alpha_4]^T$ and $H = \begin{pmatrix} 9 & 1 & -1 & -1 \\ 1 & 9 & -1 & -1 \\ -1 & -1 & 9 & 1 \\ -1 & -1 & 1 & 9 \end{pmatrix}$.

For finding the maximum, we can calculate the partial derivatives with regards to unknown parameters α_i and equate zero these derivatives to zero. Then, we will find values of unknown on which the maximum of the objective function is reached.

$$\frac{d}{d\alpha}L(\alpha) = \begin{pmatrix}1\\1\\1\\1\end{pmatrix} - \begin{pmatrix}9 & 1 & -1 & -1\\1 & 9 & -1 & -1\\-1 & -1 & 9 & 1\\-1 & -1 & 1 & 9\end{pmatrix}\alpha = 0.$$

Solving the system equations, we receive $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = \frac{1}{4}$, and

$$w = \sum_{i=1}^{4} \alpha_i u_i \varphi(x_i) = \frac{1}{4}(\varphi(x_i) + \varphi(x_2) - \varphi(x_3) - \varphi(x_4))$$
$$= [0 \quad 0 \quad 0 \quad -\sqrt{2} \quad 0 \quad 0]$$

and, at last, the nonlinear discriminant function can take the following form

$$g(x) = w\varphi(x) = \sum_{i=1}^{6} w_i \varphi_i(x) = -\sqrt{2}(\sqrt{2}x^{(1)}x^{(2)}) = -2x^{(1)}x^{(2)}\ .$$

The result is shown on Fig. 8.

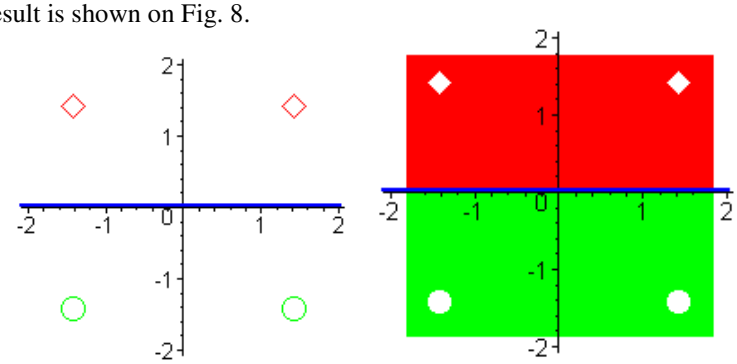

Figure 13. An example of linearly inseparable sets after using the kernel function

In conclusion we give a few of the most widespread kernel functions used for division of classes:

- The polynomial homogeneous kernel $K(x_i, x_j) = (x_i^T x_j)^d$.
- The polynomial heterogeneous kernel $K(x_i, x_j) = (x_i^T x_j + 1)^d$.
- The radial basis function (RBF kernel) $K(x_i, x_j) = \exp(-\frac{\|x_i - x_j\|^2}{2\sigma^2})$.
- The sigmoid kernel $K(x_i, x_j) = \tanh\big((x_i^T x_j + 1)\big)$

Conclusions

Thus, the classification problem was considered in the article — a task in which there are many objects, divided in some way into classes as well as methods and algorithms for solving it. Algorithms are analyzed that are able to classify an arbitrary object from the original set.

REFERENCES

1. RANA S.: Examining the Role of Local Optima and Schema Processing in Genetic Search, PhD thesis, Colorado State University, Colorado (1999).
2. Reuters-21578 text categorization test collection, (url: *http://www.daviddlewis.com/resources/testcollections/reuters21578/* date accessed: 20.03.2018)
3. SALTON G., BUCKLEY C.: Term weighting approaches in automatic text retrieval, In: Information Processing and Management, 24(1988)5, 513–523.
4. SHUMEYKO A.A., SOTNIK S.L.: Using of Genetic Algorithms for Text Classification Problems, In: Annals. Computer Science Series, 7(2009)(fasc. 1), 325-340
5. SHUMEYKO A.A., SOTNIK S.L.: Use of the agglomerative clustering for the automatic rubrication of texts, System technologies, 3(2011)74, 131-137
6. VAPNIK, V., CHERVONENKIS, A.: Theory of Pattern Recognition, Nauka, Moscow (in Russian); German translation: Theorie der Zeichenerkennung, Akademie Verlag, Berlin 1979, (Google Scholar url: *http://www.citeulike.org/group/1938/article/1055538*, date accessed: 18.04.2018).
7. VAPNIK V., LERNER A.: Pattern Recognition Using Generalized Portraits, Automation and Remote Control, 24(1963), 709–715.
8. JOACHIMS T.: Retrospective on Transductive Inference for Text Classification using Support Vector Machines, In: Proceedings of the International Conference on Machine Learning (ICML), 2009.
9. JOACHIMS T.: SVM-light Support Vector Machine, (url: *http://svmlight.joachims.org*/; date accessed: 20.03.2018)
10. CORTES C.; VAPNIK V.: Support-vector networks, Machine Learning, 20(1995)3, 273–297, [DOI:10.1007/BF00994018].
11. CRISTIANINI N., SHAWE-TAYLOR J.: An Introduction to Support Vector Machines and Other Kernel-based Learning Methods, Cambridge University Press New York, NY, USA 2000, 204p., ISBN:0-521-78019-5.
12. HAMEL L.: Knowledge Discovery With Support Vector Machines, Wiley Series on Methods and Applications in Data Mining, Hoboken, New Jersey, John Wiley and Sons Inc 2009, 262p., ISBN-13: 978-0470371923.
13. BORDES A., ERTEKIN S., WESTON J., BOTTOU L.: Fast kernel classifiers with online and active learning, Journal of Machine Learning Research, 6(2005), 1579–1619.
14. BOSER B.E., GUYON I.M., VAPNIK V.N.: A Training Algorithm for Optimal Margin Classifiers, In: D. Haussler, (Eds.), Proceedings of the Fifth Annual ACM Workshop on Computational learning theory COLT. Pittsburgh, Pennsylvania, USA, pp. 144-152, Proceedings of the fifth annual workshop on Computational

learning theory – COLT '1992, p. 144, [DOI:10.1145/130385.130401], ISBN:089791497X.

15. ROSSI F., VILLA N.: Support vector machine for functional data classification, Neurocomputing, 69(2006)7–9, 730-742
(url: https://doi.org/10.1016/j.neucom.2005.12.010, date accessed: 20.03.2018).