XIII Balkan Conference on Operational Research

ONE APPROACH FOR THE GROUP SYNTHESIS OF RECOGNITION AND CLASSIFICATION TASKS

KALIMOLDAYEV MAKSAT[1], AMIRGALIYEV YEDILKHAN[1,5], BERIKOV VLADIMIR[2,3] , CHERICBAYEVA LYAILA[4], LATUTA KONSTANTIN[5], KALYBEK UULU BEKTURGAN[6]

[1]Institute of Information and Computational Technologies, SC MES RK, Almaty, mnk@ipic.kz, amir_ed@mail.ru
[2] Sobolev Institute of Mathematics, SB RAS, Novosibirsk, berikov@math.nsc.ru
[3]Novosibirsk State University, berikov@math.nsc.ru
[4]Alfarabi Kazakh National University, Almaty, lyailya_sh@mail.ru
[5]Suleyman Demirel University, Almaty, konstantin.latuta@sdu.edu.kz
[6]Institute of Automation and Information Technology of the NAN KR, Kyrgyzstan, ukuevb@gmail.com

---

Abstract: *In this work semi-supervised learning was considered. To solve the problem of semi-supervised learning CASVM and CANN algorithms were developed. The algorithms are based on combination of collective cluster analysis and kernel methods. Probabilistic model of classification with use of cluster ensemble was proposed. Within the model, error probability of CANN was studied. Assumptions that make probability of error converge to zero were formulated. The proposed algorithms were experimentally tested on a hyper spectral image. It was shown that CASVM is more noise resistant than standard SVM.*

Keywords: *Recognition, Classification, Hyper spectral image, Semi-supervised learning.*

1. INTRODUCTION

At present, a sufficiently large number of algorithms for cluster analysis have been developed (Berikov 2013, Amirgaliev & Mukhamedgaliev 1985, Aidarkhanov et al. 2001) . The problem of cluster analysis can be formulated as follows. There are many objects described by a set of some variables (or a distance matrix). These objects are to be broken down into a relatively small number of clusters (groups, classes) so that the grouping criterion would take its "best" value. The number of clusters can be either selected in advance or not specified at all (in the latter case, the optimal number of clusters must be determined automatically). A quality criterion usually means a certain function, depending on the scatter of objects within the group and the distances between groups (Berikov 2013, Amirgaliev & Mukhamedgaliev 1985) .

Recently cluster analysis has been actively developing an approach based on collective decision-making. It is known that algorithms of cluster analysis are not universal: each algorithm has its own specific area of application: for example, some algorithms can better cope with problems in which objects of each cluster are described by "spherical" regions of multidimensional space; other algorithms are designed to search for "tape" clusters, etc. In the case when the data are of a heterogeneous nature, it is advisable to use not one algorithm but a set of different algorithms to allocate clusters. The collective (ensemble) approach also makes it possible to reduce the dependence of grouping results on the choice of parameters of the algorithm, to obtain more stable solutions in the conditions of "noisy" data, if there are "omissions" in them (Berikov, 2013, Amirgaliev & Mukhamedgaliev 1985, Aidarkhanov et al. 2001) .

In classification problems, group methods are widely used. They consist in the synthesis of results obtained by applying different algorithms to a given source information, or in selection of optimal, in some sense, algorithms from a given set. There are various ways of defining group classifications. The formation of recognition as an independent scientific theory is characterized by the following stages:

- The appearance of a large number of various incorrect (heuristic) methods and algorithms to solve practical problems, oftentimes applied without any serious justification.
- The construction and research of collective (group) methods, providing a solution to the problem of recognition based on the results.
- processing of initial information by separate algorithms (Joydeep & Ayan, 2011) .

Ensemble approach allows improving the quality of clustering. There are several main directions in the methods of constructing ensemble solutions of cluster analysis: based on the consensus distribution, on the co-associative matrices, on the models of the mixture of distributions, graph methods, and so on. as well as the main methods for obtaining collective cluster solutions: the use of a pairwise similarity/difference matrix;

maximization of the degree of consistency of decisions (normalized mutual information, corrected Rand index, etc.) Each cluster analysis algorithm has some input parameters, for example, the number of clusters, the boundary distance, etc. In some cases, it is not known what parameters of the algorithm work best. It is advisable to apply the algorithm with several different parameters rather than one specific parameter.

1.1. Ensemble cluster analysis

The group ensemble consists of different partitions. Such partitions can be obtained from several applications of any one algorithm with different parameters, or from applying different algorithms to one data set. The proposals of cluster ensembles solve the problems inherent in clustering: they can provide more reliable and stable solutions using consensus on several clustering results. The orthogonal problem associated with clustering is a large dimension. Large data is a complex task for the clustering process. Different algorithms of clustering can process data of low dimensionality, but as data dimension increases, these algorithms tend to collapse. In large-size spaces, it is very likely that for any given pair of points within one cluster, there are at least several dimensions at which the points are far apart. As a consequence, distance functions that are using equally all input functions can be inefficient. The technique of a group ensemble is characterized by two components: a mechanism for creating a variety of sections, as well as a consensus function for combining the input of partitions into finite clustering. A variety of partitions are usually generated using different clustering algorithms, or by applying a single algorithm with different parameters, possibly in combination with a selection of data or functions. One popular methodology for constructing a consensus function uses a co-associated matrix. Such a matrix can be considered as a similarity matrix, so it can be used with any clustering algorithm that works directly on similarities.

Let set S = {x_1, x_2, ..., x_n} has n points. An ensemble is a collection of m clustering solutions: G = {G_1, G_2, ..., G_m}. Each solution G_L for L = 1, ..., m, is a partitioning of set S, that is, $G_L = \{G_L^1, G_L^2, \ldots, G_L^{K_l}\}$, where $\cup_K G_L^K$=S. Given the set of cluster solutions C and the desired number of clusters k, the goal is to combine different solution clusters and calculate a new partition of S into k disjoint clusters. The task of cluster ensembles is to develop an appropriate consensus function that integrates cluster component solutions into "improved" final clustering.

In this work semi-supervised learning is considered. In semi-supervised learning the classes are known only for a subset of objects in the sample. The problem of semi-supervised learning is important for the following reasons:

- unlabeled data is cheap
- labeled data may be difficult to obtain
- using unlabeled data along with some labeled data may increase the quality of learning

There are many algorithms and approaches to solve the problem of semi-supervised learning (Joydeep & Ayan, 2011) . The goal of the work is to devise and test a novel approach to semi-supervised learning. The novelty lies in the combination of algorithms of collective cluster analysis (Domeniconi & Al-Razgan 2009, Berikov 2014) and kernel methods (support vector machines *SVM* (Berikov & Pestunov, 2017) and nearest neighbor *NN*), as well as in theoretical analysis of the error of the proposed method. In the coming sections a more formal problem statement will be given, some cluster analysis and kernel methods will be reviewed, the proposed methods will be described and its theoretical and experimental ground will be provided.

Cluster ensembles combine multiple clusters of a set of objects into one consolidated clustering, often called a consensus solution.

2. FORMAL PROBLEM STATEMENT OF SEMI-SUPERVISED LEARNING

Suppose we have a set of objects X to classify and finite set of class labels Y . Features describe all the objects. By a feature of an object we mean the following mapping $f : X \to D_f$, where D_f — set of values of a feature.

Depending on D_f features can be of the following types: Binary features: D_f ={0,1}; Numerical features: $D_f = R$; Nominal features: D_f — finite set; Ordered features: D_f — finite ordered set.

For a given feature vector $f_1,.., f_m$, vector $x = (f_1(\alpha),.., f_m(\alpha))$ is called feature descriptor of object $\alpha \in X$. Further, in the text we do not distinguish between an object and its feature descriptor. In the problem of semi-supervised learning at the input we have a sample X_N = { $x_1,.., x_N$ } of objects from X .

There are two types of objects in the sample:

- $X_c = \{x_1, .., x_k\}$ - labeled objects with the classes they belong to: $Y_c = \{y_1, .., y_k\}$
- $X_u = \{x_{k+1}, .., x_N\}$ - unlabeled objects

Conduct so-called inductive learning — build a classification algorithm $a : X \to Y$, which minimizes probability of error and match objects to their X_u, and new objects to X_{test}, which were unavailable at the time of building of the algorithm.

The second is so-called transductive learning. Here we get labels only for objects from X_u with minimal error. In this work, we consider the second variant of problem statement.

The following example shows how semi-supervised learning differs from a supervised learning.

Example: Label objects are given at the input $X_c = \{x_1, .., x_k\}$ with their respective classes $Y_c = \{y_1, .., y_k\}$, where $y_i \in \{0,1\}, i = 1, .., k$. The objects have two features and their distribution is shown in Figure 1 a). Unlabeled data is also given $X_u = \{x_{k+1}, .., x_N\}$ as shown in Figure 1b).

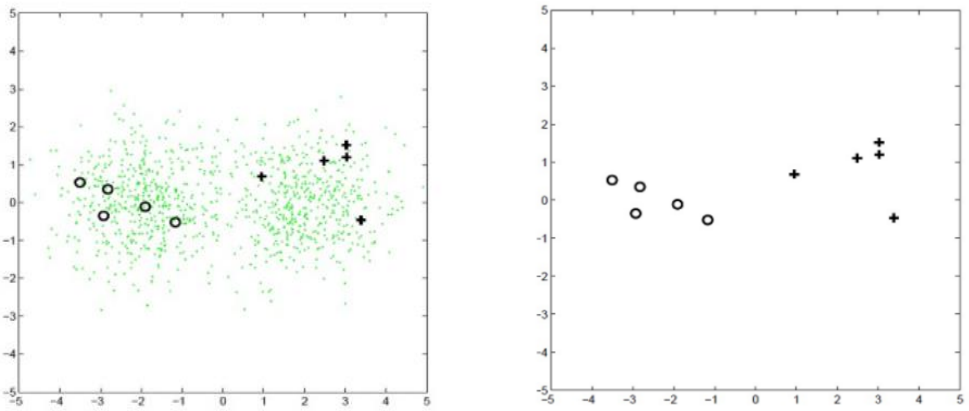

Figure 1: a) Features of objects; b) Labeled objects X_c with unlabeled objects X_u

Suppose that a sample from a mixture of normal distributions is given. Let's estimate the density of the classes throughout the data set at only on the labeled data, after which we construct the separating curves. Then, from Figure 2 it can be seen that the quality of the classification using the full set of data is higher.

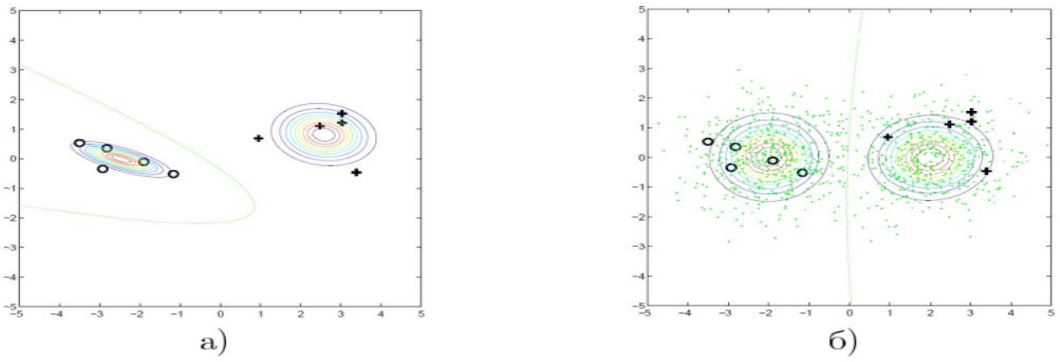

Figure 2: Obtained class densities: a) by labeled data; b) by unlabeled data

3. COLLECTIVE SOLUTIONS IN CLUSTER ANALYSIS

3.1. On the reasons for the development of the collective approach

The task of cluster analysis is to split the sample into disjoint subsets, called clusters, so that each cluster represents a group of similar objects, and objects in different clusters differ significantly. The solution of the clustering problem may be ambiguous for several reasons:

- There is no best criterion for the quality of clustering. A large number of reasonable heuristic criteria and algorithms are known that do not have an explicitly defined criterion, but quite a decent clustering;
- The number of clusters is very often unknown in advance and is set either manually or during the operation of the algorithm;
- The results of clustering depend very much on the metric, which is chosen by the expert and the specifics of the application domain.

3.2. The matrix of average differences

To construct a matrix of average differences, clustering of all available objects $X = \{x_1, .., x_N\}$ is done by an ensemble of several different algorithms $\mu_1, .., \mu_M$. Each algorithm gives L_m variants of partition, $m = 1, .., M$. Based on the results of the algorithms, a matrix H of average differences is built for objects of X. The matrix elements are equal to:

$$h(i, j) = \sum_{m=1}^{M} \alpha_m \frac{1}{L_m} \sum_{l=1}^{L_m} h_{lm}(i, j), \quad (1)$$

where $i, j \in \{1, .., N\}$ - objects' numbers $(i \neq j), \alpha_m \geq 0$ - initial weights so that $\sum_{m=1}^{M} \alpha_m = 1; h_{lm}(i, j) = 0$, if pair (i, j) belong to different clusters in l-th variant of partition, given by algorithms μ_m and 1, if it belongs to the same cluster.

Weights α_m may be same or, for example, may be set with respect to quality of each clustering algorithm. The selection of optimal weights is researched in Domeniconi & Al-Razgan (2009) .

4. KERNEL METHODS OF CLASSIFICATION

To solve the classification problem, kernel methods are widely used, based on the so-called "kernel trick". To demonstrate the essence of this "trick", consider the support vector machine method (SVM) - the most popular kernel method of classification. SVM is a binary classifier, although there are ways to refine it for multiclassification.

4.1. Binary classification with SVM

In the problem of dividing into two classes (the problem of binary classification), a training sample of objects $X = \{x_1, .., x_n\}$ is at the input with classes $Y = \{y_1, .., y_n\}$, $y_i \in \{+1, -1\}$, for $i = 1, .., n$, where object are points in m-dimensional space of feature descriptors. We are to divide the points by hyperplane of dimension $(m-1)$. In the case of linear class separability, there exist an infinite number of separating hyperplanes. It is reasonable to choose a hyperplane, the distance from which to both classes is maximized. An optimal separating hyperplane is a hyperplane that maximizes the width of the dividing strip between classes. The problem of the support vector machine method consists in constructing an optimal separating hyperplane. The points lying on the edge of the dividing strip are called support vectors.

A hyperplane can be represented as $\langle w, x \rangle + b = 0$, where $\langle , \rangle$ — scalar product, w — vector perpendicular to separating hyperplane, and b — an auxiliary parameter. Support vector method builds decision function in in the form of

$$F(x) = sign(\sum_{i=1}^{n} \lambda_i c_i \langle x_i, x \rangle + b)$$

It is important to note that the summation goes only along support vectors for which $\lambda_i \neq 0$. Objects $x \in X$ with $F(x) = 1$ will be assigned one class, and objects with $F(x) = 0$ another.

With linear inseparability of classes, one can perform a transformation $\varphi : X \to G$ of object space X to a new space G of a higher dimension. The new space is called is called "rectifying", because the objects in the space can already be linearly separable.

Decision function $F(x)$ depends on scalar products of objects, rather that the objects themselves. That is why scalar products $\langle x, x' \rangle$ can be substituted by products of $\langle \varphi(x), \varphi(x') \rangle$ kind in the space G. In this case the decision function $F(x)$ will look like this:

$$F(x) = sign(\sum_{i=1}^{n} \lambda_i c_i \langle \varphi(x_i), \varphi(x) \rangle + b)$$

Function $K(x, x') = \langle \varphi(x), \varphi(x') \rangle$ is called kernel. The transition from scalar products to arbitrary kernels the "kernel trick". Selection of the kernel determines the rectifying space and allows to use linear algorithms (like SVM) to linearly non-separable data.

5. PROPOSED METHOD

The idea of the method is to construct a similarity matrix (1) of all objects from the input sample X. The matrix will be compiled by applying different clustering algorithms to X. The more a pair of objects are classified as belonging to one class the more similar they will be. Two possible variants of prediction for unlabeled classes X_u will be proposed using similarity matrix. Further the idea of the algorithms will be described in detail. The following theorem holds:

Theorem 1. Let $\mu_1, .., \mu_M$ — be algorithms of clustering analysis , each algorithm gives L_m variants of partition, $m = 1, .., M$, $h_{lm}(x, x') = 0$, if a pair of objects (x, x') belongs to different clusters in l-th variant of partition, given by algorithm μ_m and 1, if it belongs to the same cluster. $\alpha_m \geq 0$ - initial weights such that $\sum_{m=1}^{M} \alpha_m = 1$. Then function $H(x, x') = \sum_{m=1}^{M} \alpha_m \frac{1}{L_m} \sum_{l=1}^{L_m} h_{lm}(x, x')$ satisfies the condition of Mercer theorem.

Proof. It is obvious that function $H(x, x')$ symmetric. Let C_r^{lm} - be the set of indices of objects that belong to r-th cluster, given by m-th algorithmin l-th variant of partition. Let's show that $H(x, x')$ nonnegatively defined.

Let take arbitrary $z \in R^p$ and show that $z^T H z \geq 0$

$$z^T H z = \sum_{i,j=1}^{p} \sum_{m=1}^{M} \alpha_m \frac{1}{L_m} \sum_{l=1}^{L_m} h_{lm}(i, j) z_i z_j = \sum_{m=1}^{M} \alpha_m \frac{1}{L_m} \sum_{l=1}^{L_m} \sum_{i,j=1}^{p} h_{lm}(i, j) z_i z_j =$$

$$\sum s_{m=1}^{M} \alpha_m \frac{1}{L_m} \sum_{l=1}^{L_m} (\sum_{i,j \in C_1^{lm}} z_i z_j + ... + \sum_{i,j \in C_{K_{lm}}^{lm}} z_i z_j) = \sum_{m=1}^{M} \alpha_m \frac{1}{L_m} \sum_{l=1}^{L_m} ((\sum_{i \in C_1^{lm}} z_i)^2 + ... + (\sum_{i \in C_{K_{lm}}^{lm}} z_i)^2) \geq 0.$$

Thus, function $H(x, x')$ can be used as a kernel in kernel methods of classification. For instance, in support vector machines (SVM) and in nearest neighbor method (NN). Further, the two variants of the algorithm that implement the proposed method are described:

Algorithm CASVM

Input: objects X_c with their classes Y_c and objects X_u, number of clustering algorithms M, number of clustering L_m by each algorithm $\mu_m, m = 1, ..., M$.

Output: classes of objects X_u.

1. Cluster objects $X_c \cup X_u$ by algorithms $\mu_1, .., \mu_M$, and get L_m variants of partitions from each algorithm $\mu_m, m = 1, ..., M$.

2. Computer matrix H for $X_c \cup X_u$ by formula (1).

3. Train *SVM* with labeled data X_c, using matrix H as kernel.

4. By means of *SVM* predict classes of unlabeled data X_u.

End of algorithm

Algorithm CANN

Input: objects X_c with given classes Y_c and objects X_u, number of clustering algorithms M, number for clusters L_m by each algorithm $\mu_m, m = 1,...,M$.

Output: classes of objects X_u.

1. Cluster objects $X_c \cup X_u$ by algorithms $\mu_1,..,\mu_M$, get L_m variants of partitions from each algorithm $\mu_m, m = 1,...,M$.

2. Compute H for $X_c \cup X_u$ by formula (1).

3. Use *NN*: for each unlabeled object $x \in X_u = \{x_{k+1},..,x_N\}$ assign the most similar class in sense $H(x,x')$ of labeled object $x' \in X_c = \{x_1,..,x_k\}$.

Formally written: $x_i = \underset{j=1,..,k}{\arg\max} H(x_i, x_j),\ i = k+1,..,N$.

End of algorithm

Note that in the proposed algorithms there is no need to store matrix H in memory $N \times N$ entirely: it is enough to store the clustering matrix of size $N \times L$, where $L = \sum_{l=1}^{M} L_m$, in this case H can be computed dynamically. In practice, $L << N$, for example, when working with image pixels.

6. THEOETICAL ANALYSIS OF CANN

Let's recall the problem statement: At the input we have sample of objects $X_N = \{x_1,..,x_N\}$. There are two types of objects in the sample:

- $X_c = \{x_1,..,x_k\}$ - labeled objects with classes $Y_c = \{y_1,..,y_k\}$, $I_c = \{1,...,k\}$ - object indices
- $X_u = \{x_{k+1},..,x_N\}$ - unlabeled objects, $I_u = \{k+1,...,N\}$ - indices of the objects

For simplicity assume that he classes do not intersect, run L clustering by one algorithm μ with random parameters $\Omega_1,...,\Omega_L$.

Let's introduce the following notations for $i, j \in I_u$:

$$h_l(x_i, x_j) = \begin{cases} 1, & algorithm\ \mu in\ variant\ l\ combined\ pair\ of\ x_i, \quad x_j\ into\ one\ cluster \\ 0, & otherwise \end{cases}$$

and $L_1(i,j) = \sum_{l=1}^{L} h_l(x_i, x_j)$, $L_0(i,j) = L - L_1(i,j)$, which represent the number of variants of clustering, in which algorithms has voted to combine x_i, x_j, or against, respectively .

Let $Y(x)$ - hidden true labels of the classes of unlabeled objects $x \in X_u$. Let's introduce a quantity for $i, j \in I_u$:

$$z(x_i, x_j) = \begin{cases} 1, if\ Y(x_i) = Y(x_j) \\ 0, if\ Y(x_i) \neq Y(x_j). \end{cases}$$

In the nearest neighbor methods (NN) for all $i \in I_u$ we assign label y_i a values y', where y' - class of object, $x' = \underset{x_j \in X_c}{arg\ max}\ H(x_i, x_j)$

The following theorem holds

Theorem 1. Let $\forall l \in \{1,...,L\}$, $P[h_l(x_i,x_j)=1 \mid z(x_i,x_j)=1] > \frac{1}{2}$, 1and $L_0(i,j)=const \quad \forall i,j \in I_u$. Then in the algorithm CANN for object $x_i \in X_u$ the probability of incorrect classification $P_{er}(x_i)=P[Y(x_i)\neq y'] \to 0$ for $L \to \infty$.

The theorem shows that the probability of a classification error by the CANN algorithm tends to zero under the assumptions that the classes of objects do not intersect and that the algorithms of cluster analysis that are used correctly classify pairs of objects to one or different clusters with probability more $1/2$, that is, they do not act at random.

7. EXPERIMENTAL ANALYSIS

A typical RGB image contains three channels: the intensity values for each of the three colors. In some cases, this is not enough to get complete information about the characteristics of the object being shot. To obtain data on the properties of objects that are indistinguishable by the human eye, hyper spectral images are used.

For an experimental analysis of the developed algorithm, we used a picture of Pavia University scene with size of 610 x 340 pixels, which contains 103 spectral channels. The spatial resolution of the image is 1.3 m. Figure 5a shows the RGB composite images (channels 40, 50 and 70), and in Figure 5b) the standard image partition into thematic classes is given.

а) *б)*

Рис. 4. Гиперспектральное изображение Pavia University scene (RGB композит) (а) и размеченные данные (б).

Figure 3: Hyper spectral image of Pavia University scene a) RGB composite images b) Marked data

Note that the image has unmapped pixels that are not assigned to any of the nine classes. These pixels were excluded from consideration in the analysis.

In an experimental analysis of the algorithm, 1% of the pixels selected at random for each class made up the labeled sample; the remaining ones were included in the unlabeled set.

To study the effect of noise on the quality of the algorithm, randomly selected r% of the spectral brightness values of the pixels in different channels were subjected to a distorting effect: the corresponding value was replaced by a random variable from interval

To study the effect of noise on the quality of the algorithm, randomly selected r % of the spectral brightness values of the pixels in different channels were subjected to a distorting effect: the corresponding value x was replaced by a random variable from interval $[x(1-p), x(1+p)]$, where r, p - initial parameters. The noisy data table containing the spectral brightness values of the pixels across all channels was fed to the input of the CASVM algorithm, and the K-means algorithm was chosen as the basic algorithm for constructing the cluster ensemble. Different variants of partitioning were obtained by varying the number of clusters in the interval $[30, 30+L]$, where L was equal to 120.

In addition, for the construction of each variant of the solution, channels were randomly chosen, the number of which was set to 2. To speed up the operation of the K-means algorithm and to obtain more diverse groupings, the number of iterations was limited to 1.

Since the proposed algorithm implements the idea of distance metric learning, it would be natural to compare it with a similar algorithm (SVM method), which uses the standard Euclidean metric, under similar conditions (the algorithm parameters recommended by default in Matlab environment).

Table 1 shows the accuracy values of the classification of the unlabeled pixels of the Pavia University scene for some values of the noise parameters. The running time of the algorithm was about 2 minutes on a dual-core Intel Core i5 processor with a clock speed of 2.8 GHz and 4 GB of RAM. As it is shown in the table, CASVM algorithm has better noise resistance than SVM algorithm.

Table 1: Accuracy of CASVM and SVM under various noise values

Noise parameters r, p	0%, 0	10%, 0.1	20%, 0.2	30%, 0.3
CASVM	0.82	0.80	0.78	0.77
SVM	0.83	0.75	0.66	0.64

8. CONCLUSION

The paper considers one of the variants of the problem of pattern recognition - the task of semi-supervised learning. The algorithms CASVM and CANN were developed to solve this problem. They are based on a combination of methods of collective cluster analysis and kernel classification methods.

A probabilistic classification model using a cluster ensemble was proposed. Within the model, the behavior of error probability of the CANN algorithm was analyzed. The assumptions are formulated, in which the error probability tends to zero.

An experimental study of the proposed algorithm on a hyperspectral image was performed. It was shown that the CASVM algorithm is more noise-resistant than the standard method of SVM.

REFERENCES

Aidarkhanov, M.B., Amirgaliev, E.N., La, L.L. (2001). Correctness of algebraic extensions of models of classification algorithms. In *Cybernetics and Systems Analysis*, 37(5), 777-781.

Amirgaliev, E.N., Mukhamedgaliev, A.F. (1985). On optimization model of classification algorithms. In *USSR Computational Mathematics and Mathematical Physics*, 25(6), 95-98.

Berikov, V.B., Lbov, G.S. Current trends in cluster analysis. http://biocomparison.ucoz.ru/_ld/0/49_berikov_lbov.pdf.

Berikov, V.B. (2013). A team of algorithms with weights in a cluster analysis of heterogeneous data. *Bulletin of Tomsk State University*, 2 (23), 22-31.

Berikov, V.B. (2014) Weighted ensemble of algorithms for complex data clustering.*Pattern Recognition Letters*, 38, 99-106.

Berikov, V., Pestunov, I. (2017). Ensemble clustering based on weighted co-association matrices: Error bound and convergence properties. *Pattern Recognition*, 63, 427-436.

Domeniconi, C., Al-Razgan, M. (2009). Weighted cluster ensembles: Methods and analysis. *ACM Transactions on Knowledge Discovery from Data*, 2(4):17.

Joydeep, G., Ayan, A. (2011). Cluster ensembles. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(4), 305–315.