Big Data Classification Using the SVM Classifiers with the Modified Particle Swarm Optimization and the SVM Ensembles

Liliya Demidova
Moscow Technological Institute
Ryazan State Radio Engineering University
Moscow, Russia

Evgeny Nikulchev
Moscow Technological Institute
Moscow,
Russia

Yulia Sokolova
Ryazan State Radio Engineering University
Ryazan, Russia

***Abstract*—The problem with development of the support vector machine (SVM) classifiers using modified particle swarm optimization (PSO) algorithm and their ensembles has been considered. Solving this problem would allow fulfilling the high-precision data classification, especially Big Data classification, with the acceptable time expenditures. The modified PSO algorithm conducts a simultaneous search of the type of kernel functions, the parameters of the kernel function and the value of the regularization parameter for the SVM classifier. The idea of particles' «regeneration» served as the basis for the modified PSO algorithm. In the implementation of this algorithm, some particles change the type of their kernel function to the one which corresponds to the particle with the best value of the classification accuracy. The offered PSO algorithm allows reducing the time expenditures for the developed SVM classifiers, which is very important for Big Data classification problem. In most cases such SVM classifier provides the high quality of data classification. In exceptional cases the SVM ensembles based on the decorrelation maximization algorithm for the different strategies of the decision-making on the data classification and the majority vote rule can be used. Also, the two-level SVM classifier has been offered. This classifier works as the group of the SVM classifiers at the first level and as the SVM classifier on the base of the modified PSO algorithm at the second level. The results of experimental studies confirm the efficiency of the offered approaches for Big Data classification.**

***Keywords*—Big Data; classification; ensemble; SVM classifier; kernel function type; kernel function parameters; particle swarm optimization algorithm; regularization parameter; support vectors**

I. INTRODUCTION

Big Data is a term for data sets that are so large and/or complex that traditional data processing technologies are inadequate. They require technologies that can be used to store and process the exponentially increasing data sets which contain structured, semi structured and unstructured data. Volume, variety and velocity are three defining characteristics of Big Data. Volume refers to the huge amount of data, variety refers to the number of data types and velocity refers to the speed of data processing. The problems of the Big Data management result from the expansion of all three characteristics. The Big Data does not consist of only numbers and strings but also geospatial data, audio, video, web data, social files, etc. obtained from various sources such as sensors, mobile phones, cameras and so on.

The main purpose of the Big Data technologies is to provide the high quality of data processing and data analysis. Nowadays the Big Data technologies have been applied in many fields of science and engineering, including physical, biological and biomedical sciences. Also, they have been used in government agencies, financial corporations, large enterprises, etc.

The high volume of storage space, in particular, the cloud storage is needed to manage and reuse Big Data which can be useful for many purposes, for example, for hardware and software maintenances. It is therefore necessary to perform the analytical, retrieval and process operations, which are very complex and time consuming ones. In order to overcome these difficulties new Big Data technologies have been getting a lot of attention over the last few years. The Big Data processing improves the transfer speed of the data sets in comparison to the speed of the simple data exchanges. The Big Data mining tools are very useful to the end users when they solve their own actual problems.

Currently many efficient approaches must be implemented when dealing with the Big Data. In particular, the feature selection, clustering and classification plays an important role in the Big Data analysis, when it is necessary to retrieve, search or classify a data, using the Big Data sets. These approaches are useful for such spheres as pattern recognition, machine learning, bio-informatics, data mining, semantic ontology and so on. As there are many algorithms available for the feature selection, clustering and classification, it is necessary to find the appropriate algorithms which must be chosen properly for the problem of the Big Data analysis.

The machine learning algorithms can be considered along a spectrum of the supervised and unsupervised learning algorithms. In the strictly unsupervised learning, the problem is to find the structure such as clusters in the unlabeled data set. The supervised learning uses the training set of the classified data to construct classifier, which can be used to classify new data. In both cases, the Big Data applications demonstrate the growing number of features and the growing volume of the input data.

The Support Vector Machine (SVM) algorithm is the supervised machine learning algorithm. Currently, the SVM algorithm (one of the boundary classification algorithms [1, 2]) is used for different classification problems in various applications with great success.

The SVM classifiers based on the SVM algorithm have been applied for credit risk analysis [3], medical diagnostics [4], handwritten character recognition [5], text categorization [6], information extraction [7], pedestrian detection [8], face detection [9], Earth remote sensing [10], etc.

SVM classifier uses special kernel function to construct a hyperplane separating the classes of data. An example of the separating hyperplane in the 2D space is shown in Fig. 1.

The SVM classifier is used for training, testing, and classification. Satisfactory quality of training and testing allows using the resulting SVM classifier in the classification of new objects.

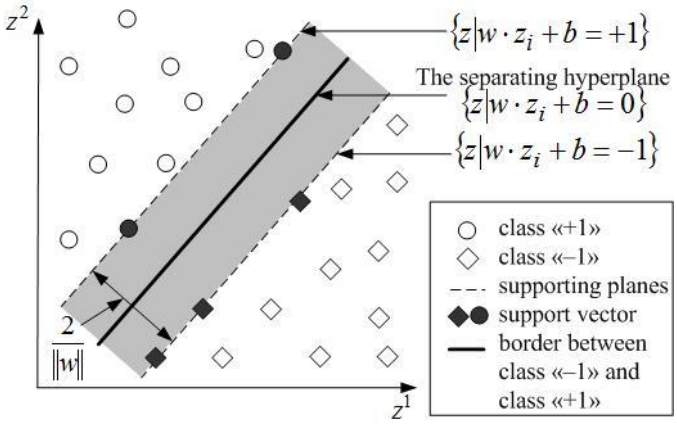


Fig. 1. Linear separation for two classes by the SVM classifier in the 2D space

SVM algorithms are well-known for their excellent performance in the sphere of the statistical classification. Still, the high computational cost due to the cubic runtime complexity is problematic for the Big Data sets: the training of the SVM classifier requires solving a quadratic optimization problem [1, 3]. Using a standard quadratic problem solver for the SVM classifier training would involve solving a big quadratic programming problem even for a moderate sized data set. This can limit the size of problems which can be solved with the application of the SVM classifier. Nowadays methods like SMO [11], chunking [12] and simple SVM [13], Pegasos [14] exist that iteratively compute the required solution and have a linear space complexity [15].

In recent years to mitigate the problem of the high computational cost the cascade SVM algorithm had been proposed [16]. In this algorithm the SVM classifier is iteratively trained on subsets of the original data set, acquired support vectors of the resulting models are combined to create new training sets. The general idea is to bound the sizes of all considered training sets and therefore obtain a significant speedup. This algorithm can easily be parallelized because the number of independent models has to be fitted during each stage of the cascade [17].

In the millennium of Big Data it is necessary to develop data mining algorithms which are suitable for the Big Data analysis. Several parallel algorithms have been developed using threads, MPI, MapReduce and so on [18]. Among all these techniques MapReduce is practically well suited for the Big Data analysis. One of the last trends in the Big Data processing and analysis is using the Hadoop framework for the SVM classifiers development [18]. Hadoop is an open-source software framework for the distributed storage and distributed processing of very large data sets on the computer clusters built from the commodity hardware. The Hadoop cluster is a special type of computational cluster designed specifically for storing and analyzing huge amounts of unstructured data in the distributed computing environment.

Therefore the use of the SVM algorithm is very perspective for the Big Data classification [19, 21].

Choosing optimal parameters for the SMV classifier is a significant problem at the moment. It is necessary to find the kernel function type, values of the kernel function parameters and value of the regularization parameter, which must be set by a user and shouldn't be changed [1, 2]. It is impossible to provide implementing of high-accuracy data classification with the use of the SVM classifier without adequate solution to this problem.

In the simplest case solution to this problem can be found by a search of the kernel function types, values of the kernel function parameters and value of the regularization parameter that demands significant computational expenses. For an assessment of classification quality, the indicators of classification accuracy, classification completeness, etc. can be used [3].

Usually, developing the binary classifiers requires working with the complex, multiextreme function, multi-parameter objective function.

Gradient methods are not suitable for search of the optimum of this objective function, but search algorithms of stochastic optimization, such as the genetic algorithm [22, 24], the artificial bee colony algorithm [25], the particle swarm algorithm [26-29], etc., have been used. Each of the optimal decision is carried out at once in all space of possible decisions.

The particle swarm algorithm (Particle Swarm Optimization, PSO algorithm), which is based on an idea of possibility to solve the optimization problems using modeling of animals' groups' behavior is the simplest algorithm of evolutionary programming because for its implementation it is necessary to be able to determine only value of the optimized function [26-29].

The traditional approach to application of the PSO algorithm consists of the repeated applications of the PSO algorithm for the fixed type of the kernel functions to choose optimal values of the kernel function parameters and value of the regularization parameter with the subsequent choice of the best type of the kernel function and values of the kernel function parameters and value of the regularization parameter corresponding to this kernel function type.

In a traditional approach to the application of the PSO algorithm it applied repeatedly to the fixed type of the kernel functions to find the optimal parameters. Whereas with a new approach the algorithm uses simultaneous search for the best type of the kernel function, values of the kernel function parameters and value of the regularization parameter. Hereafter, particle swarm algorithms corresponding to traditional and modified approaches will be called as the traditional PSO algorithm and the modified PSO algorithm respectively [30, 31].

It is necessary to say that the PSO algorithm and other nature inspired swarm optimization algorithms are very well suited for the distributed architecture and handling of high volume unstructured data in the Big Data analytics.

In recent years, much attention is paid to the question of increasing the accuracy of the models based on the machine learning algorithms. Therefore approaches dealing with the creation of the classifiers' ensembles for the accuracy increase of the classification solution have been investigated [3–5]. The training of the SVM ensemble is the training procedure of the finite set of the base (individual) classifiers: the individual solutions are combined to form the resulting classification decisions, based on the aggregated classifier. There are different approaches to choose the combination rules of the individual classifiers in the ensemble and the strategies for the creation of the resulting classification decisions [2].

The main purposes of this paper are the following: to create the modified PSO algorithm and compare it with the traditional one using the time required to find the optimal parameters of the SVM classifier and the classification accuracy of data; to improve the accuracy of the classification decisions using the SVM ensemble based on the decorrelation maximization algorithm for the different strategies of the decision-making on the data classification and the majority vote rule; to improve the accuracy of the classification decisions using the two-level SVM classifier.

The rest of this paper is structured as follows. Section II presents the main stages of the SVM classifier development. Section III details the proposed new approach for solving the problem of the simultaneous search of the kernel function type, values of the kernel function parameters and value of the regularization parameter for the SVM classifier. This approach is based on the application of the modified PSO algorithm, the main idea of which is the «regeneration» of particles: some particles change their kernel function type to the one which corresponds to the particle with the best value of the classification accuracy. Section IV is devoted to the problems of the development of the SVM ensembles on the base of the decorrelation maximization algorithm for the different strategies of the decision-making on the data classification and the majority vote rule. Section V details the two-level SVM classifier. This classifier works as the group of the SVM classifiers at the first level and as the SVM classifier on the base of the modified PSO algorithm at the second level. Experimental results follow in Section VI. Finally, conclusions are drawn in Section VII.

II. The Support Vector Machine Classifier

Let the experimental data set be a set in the form of $\{(z_1,y_1),\ldots,(z_S,y_S)\}$, in which each object $z_i \in Z$ ($i=\overline{1,s}$; s is the number of objects) is assigned to a number $y_i \in Y=\{+1;\ -1\}$ having a value of +1 or −1 depending on the class of the object z_i. It is assumed that every object z_i is mapped to q-dimensional vector of numerical values of characteristics $z_i=(z_i^1, z_i^2, \ldots, z_i^q)$ (typically normalized by values from the interval [0, 1]) where z_i^l is the numeric value of the l-th characteristic for the i-th object ($i=\overline{1,s}$, $l=\overline{1,q}$) [30], [31]. It is necessary to use the special function $\kappa(z_i, z_\tau)$, which is called the kernel, to build the classifier $F:\ Z \to Y$, which compares the class to the number from the set $Y=\{+1;\ -1\}$ or some object from the set Z.

To build «the best» SVM classifier it is necessary to implement the numerous repeated training (for the training data set with S elements) and testing (for the test data set $s-S$ elements, $S<s$) on the different randomly generated training and test sets with following determination of the best SVM classifier in terms of the highest possible classification quality provision. The test set contains the part of data from the experimental data set. The size of the test set must be equal to 1/10 – 1/3 of the experimental data set. The test set doesn't participate in controlling the parameters of the SVM-classifier. This set is used to measure classifier's accuracy. The SVM classifier with satisfactory training and testing results can be used to classify new objects [1–3].

The separating hyperplane for the objects from the training set can be represented by equation $\langle w, z\rangle + b = 0$, where w is a vector-perpendicular to the separating hyperplane; b is a parameter which corresponds to the shortest distance from the origin of coordinates to the hyperplane; $\langle w, z\rangle$ is a scalar product of vectors w and z [1–3]. The condition $-1 < \langle w, z\rangle + b < 1$ specifies a strip that separates the classes. The wider the strip, the more confidently we can classify objects. The objects closest to the separating hyperplane, are exactly on the bounders of the strip.

If classes can be separated by the straight line, a hyperplane can be chosen so that no objects from the training set get between them and then maximize the distance between the hyperplanes (width of the strip) $2/\langle w, w\rangle$, which will solve the problem of quadratic optimization [1, 2]:

$$\begin{cases} \langle w, w\rangle \to \min, \\ y_i \cdot (\langle w, z_i\rangle + b) \ge 1, \quad i=\overline{1,S}. \end{cases} \tag{1}$$

Finding the separating hyperplane is basically the dual problem of searching a saddle point of the Lagrange function, which reduces to the problem of quadratic programming, containing only dual variables [1, 2]:

$$\begin{cases} -L(\lambda) = -\sum_{i=1}^{S}\lambda_i + \\ \quad + \frac{1}{2}\cdot\sum_{i=1}^{S}\sum_{\tau=1}^{S}\lambda_i\cdot\lambda_\tau\cdot y_i\cdot y_\tau\cdot\kappa(z_i,z_\tau)\to\min\limits_{\lambda}, \\ \quad \sum_{i=1}^{S}\lambda_i\cdot y_i = 0, \\ \quad 0\le\lambda_i\le C,\ i=\overline{1,S}, \end{cases} \quad (2)$$

where λ_i is a dual variable; z_i is the object of the training set; y_i is a number (+1 or −1), which characterize the class of the object z_i from the experimental data set; $\kappa(z_i,z_\tau)$ is a kernel function; C is a regularization parameter ($C>0$); S is a quantity of objects in the experimental data set; $i=\overline{1,S}$.

In training of the SVM classifier it is necessary to determine the kernel function type $\kappa(z_i,z_\tau)$, values of the kernel parameters and value of the regularization parameter C, which allows finding a compromise between maximizing of the gap separating the classes and minimizing of the total error. A herewith typically one of the following functions is used as the kernel function $\kappa(z_i,z_\tau)$ [1, 3, 32]:

- linear function: $\kappa(z_i,z_\tau)=\prec z_i,z_\tau\succ$;
- polynomial function: $\kappa(z_i,z_\tau)=(\prec z_i,z_\tau\succ+1)^d$;
- radial basis function:

$$\kappa(z_i,z_\tau)=exp(-\prec z_i-z_\tau,z_i-z_\tau\succ/(2\cdot\sigma^2));$$

- sigmoid function: $\kappa(z_i,z_\tau)=th(k_2+k_1\cdot\prec z_i,z_\tau\succ)$,

where $\prec z_i,z_\tau\succ$ is a scalar product of vectors z_i and z_τ; d [$d\in N$ (by default $d=3$)]; σ [$\sigma>0$ (by default $\sigma^2=1$)]; k_1 [$k_1>0$ (by default $k_1=1$)] and k_2 [$k_2<0$ (by default $k_2=-1$)] are some of parameters; th is a hyperbolic tangent.

These kernel functions allow dividing the objects from different classes.

As a result of the training, the classification function is determined in the following form [1], [3]:

$$f(z)=\sum_{i=1}^{S}\lambda_i\cdot y_i\cdot\kappa(z_i,z)+b. \quad (3)$$

The classification decision, associating the object z to the class −1 or +1, is adopted in accordance with the rule [1], [3]:

$$F(z)=sign(f(z))=sign\left(\sum_{i=1}^{S}\lambda_i\cdot y_i\cdot\kappa(z_i,z)+b\right). \quad (4)$$

The SVM classifier training results in determining the support vectors. Using the PSO algorithm provides better accuracy of classification by choosing the kernel function type, values of the kernel function parameters and value of the regularization parameter.

Quality of the SVM classifier can be measured by different classification quality indicators [3]. There are cross validation data indicator, accuracy indicator, classification completeness indicator and ROC curve analysis based indicator, etc.

III. The Modified Particle Swarm Optimization Algorithm

In the traditional PSO algorithm the n-dimensional search space (n is the number of parameters which are subject to optimization) is inhabited by a swarm of m agents-particles (elementary solutions). Position (location) of the i-th particle is determined by vector $x_i=(x_i^1,x_i^2,\ldots,x_i^n)$, which defines a set of values of optimization parameters. These parameters can be presented in an explicit form or even absent in the analytical record of the objective function $u(x)=u(x^1,x^2,\ldots,x^n)$ of the optimization algorithm (for example, the optimum is the minimum which must be achieved).

The particles must be situated randomly in the search space during the process of initialization. Each i-th particle ($i=\overline{1,m}$) has its own vector of speed $v_i\in R^n$ which influence i-th particle ($i=\overline{1,m}$) coordinates' values in every single moment of time corresponding to some iteration of the PSO algorithm.

The coordinates of the i-th particle ($i=\overline{1,m}$) in the n-dimensional search space uniquely determine the value of the objective function $u(x)=u(x^1,x^2,\ldots,x^n)$ which is a certain solution of the optimization problem [26–29].

For each position of the n-dimensional search space where the i-th particle ($i=\overline{1,m}$) was placed, the calculation of value of the objective function $u(x_i)$ is performed. A herewith each i-th particle remembers the best value of the objective function found personally as well as the coordinates of the position in the n-dimensional space corresponding to the value of the objective function. Moreover each i-th particle ($i=\overline{1,m}$) «knows» the best position (in terms of achieving the optimum of the objective function) among all positions that had been «explored» by particles (due to it the immediate exchange of information is replicated by all the particles). At each iteration particles correct their velocity to, on the one hand, move closer to the best position which was found by the particle independently and, on the other hand, to get closer to the position which is the best globally at the current moment. After a number of iterations particles must come close to the best position (globally the best for all iterations). However, it is possible that some particles will stay somewhere in the relatively good local optimum.

Convergence of the PSO algorithm depends on how velocity vector correction is performed. There are different approaches to implementation of velocity vector v_i correction for the i-th particle ($i=\overline{1,m}$) [26]. In the classical version of the PSO algorithm correction of each j-th coordinate of

velocity vector ($j=\overline{1,n}$) of the i-th particle ($i=\overline{1,m}$) is made in accordance with formula [26]:

$$v_i^j = v_i^j + \hat{\varphi}\cdot\hat{r}\cdot(\hat{x}_i^j - x_i^j) + \tilde{\varphi}\cdot\tilde{r}\cdot(\tilde{x}^j - x_i^j), \quad (5)$$

where v_i^j is the j-th coordinate of velocity vector of the i-th particle; x_i^j is the j-th coordinate of vector x_i, defining the position of the i-th particle; $\hat{x}_i^j$ is the j-th coordinate of the best position vector found by the i-th particle during its existence; $\tilde{x}^j$ is the j-th coordinate of the globally best position within the particles swarm in which the objective function has the optimal value; $\hat{r}$ and $\tilde{r}$ are random numbers in interval (0, 1), which introduce an element of randomness in the search process; $\hat{\varphi}$ and $\tilde{\varphi}$ are personal and global coefficients for particle acceleration which are constant and determine behavior and effectiveness of the PSO algorithm in general.

With personal and global acceleration coefficients in (5) random numbers $\hat{r}$ and $\tilde{r}$ must be scaled; the global acceleration coefficient $\tilde{\varphi}$ operates by the impact of the global best position on the speeds of all particles and the personal acceleration coefficient $\hat{\varphi}$ operates by the impact of the personal best position on the velocity of some particle.

Currently different versions of the traditional PSO algorithm are known. In one of the most known canonical version it is supposed to undertake the normalization of the acceleration coefficients $\hat{\varphi}$ and $\tilde{\varphi}$ to make the convergence of the algorithm not so much dependent on the choice of their values [26].

Correction of each j-th coordinate of the velocity vector ($j=\overline{1,n}$) of the i-th particle ($i=\overline{1,m}$) is performed in accordance with formula [26]:

$$v_i^j = \chi\cdot[v_i^j + \hat{\varphi}\cdot\hat{r}\cdot(\hat{x}_i^j - x_i^j) + \tilde{\varphi}\cdot\tilde{r}\cdot(\tilde{x}^j - x_i^j)], \quad (6)$$

where χ is a compression ratio;

$$\chi = 2\cdot K / \left|2 - \varphi - \sqrt{\varphi^2 - 4\cdot\varphi}\right|; \quad (7)$$

$$\varphi = \hat{\varphi} + \tilde{\varphi} \ (\varphi > 4); \quad (8)$$

K is some scaling coefficient, which takes values from the interval (0, 1).

When using formula (6) for correction of velocity vector the convergence of the PSO algorithm is guaranteed and there is no need to control the particle velocity explicitly [26].

Let the correction of velocity vector of the i-th particle ($i=\overline{1,m}$) is executed in accordance with one of the formulas (5) or (6). The correction of the j-th coordinate of the i-th particle ($i=\overline{1,m}$) can be executed in accordance with the formula:

$$x_i^j = x_i^j + v_i^j. \quad (9)$$

Then for each i-th particle ($i=\overline{1,m}$) the new value of the objective function $u(x_i)$ can be calculated and the following check must be performed: whether a new position with coordinates vector x_i became the best among all positions in which the i-th particle has previously been placed. If new position of the i-th particle is recognized to be the best at the current moment the information about it must be stored in a vector $\hat{x}_i$ ($i=\overline{1,m}$).

Value of the objective function $u(x_i)$ for this position must be remembered. Then among all new positions of the swarm particles the check of the globally best position must be carried out. If some new position is recognized as the best globally at the current moment, the information about it must be stored in vector $\tilde{x}$. Value of the objective function $u(x_i)$ for this position must be remembered.

In the case of the SVM classifier's development with the use of the PSO algorithm the swarm particles can be defined by vectors declaring their position in the search space and corded by the kernel function parameters and the regularization parameter: (x_i^1, x_i^2, C_i), where i is a number of particle ($i=\overline{1,m}$); x_i^1, x_i^2 are the kernel function parameters of the i-th particle, [parameter x_i^1 is equal to the kernel function parameters d, σ or k_2 (depending on the kernel function type which corresponds to a swamp particle); parameter x_i^2 is equal to the kernel function parameter k_1, if the swamp particle corresponds to the sigmoid type of the kernel function, otherwise this parameter is assumed to be zero]; C_i is the regularization parameter [30, 31].

After that to choose the optimal parameter values of the kernel function and the regularization parameter traditional approach to the application of the PSO algorithm is concluded numerous times for the fixed kernel function's type.

As a result for each type T of the kernel function, participating in the search, the particle with the optimal combination of the parameters values $(\tilde{x}^1, \tilde{x}^2, \tilde{C})$ providing high quality of classification will be defined [30, 31].

The best type and the best values of the required parameters are found using the results of the comparative analysis of the best particles received at realization of the PSO algorithm with the fixed kernel function type.

Along with the traditional approach to the application of the PSO algorithm in the development of the SVM classifier there is a new approach that implements a simultaneous search for the best kernel function type $\tilde{T}$, parameters' values $\tilde{x}^1$ and $\tilde{x}^2$ of the kernel function and value of the regularization parameter $\tilde{C}$ [30, 31]. At such approach each i-th particle in a swamp ($i=\overline{1,m}$) defined by a vector which describes particle's position in the search space: (T_i, x_i^1, x_i^2, C_i), where T_i is the number of the kernel function type (for example, 1, 2, 3 – for

polynomial, radial basis and sigmoid functions accordingly); parameters x_i^1, x_i^2, C_i are defined as in the previous case. It is possible to «regenerate» particle through changing its coordinate T_i on number of that kernel function type, for which particles show the highest quality of classification. In the case of particles' «regeneration» the parameters' values change so that they corresponded to new type of the kernel function (taking into account ranges of change of their values). Particles which didn't undergo «regeneration», carry out the movement in own space of search of some dimension.

The number of particles taking part in «regeneration» must be determined before start of algorithm. This number must be equal to 15% – 25% of the initial number of particles. It will allow particles to investigate the space of search. A herewith they won't be located in it for a long time if their indicators of accuracy are the worst.

The offered modified PSO algorithm can be presented by the following consequence of steps [30].

Step 1. To determine parameters of the PSO algorithm: number m of particles in a swamp, velocity coefficient K, personal and global velocity coefficients $\hat{\varphi}$ and $\tilde{\varphi}$, maximum iterations number $N_{\max}$ of the PSO algorithm. To determine types T of kernel functions, which take part in the search ($T=1$ – polynomial function, $T=2$ – radial basis function, $T=3$ – sigmoid function) and ranges boundaries of the kernel function parameters and the regularization parameter C for the chosen kernel functions' types T: $x_{\min}^{1T}$, $x_{\max}^{1T}$, $x_{\min}^{2T}$, $x_{\max}^{2T}$, $C_{\min}^T$, $C_{\max}^T$ ($x_{\min}^{2T}=0$ and $x_{\max}^{2T}=0$ for $T=1$ and $T=2$). To determine the particles' «regeneration» percentage p.

Step 2. To define equal number of particles for each kernel type function T, included in search, to initialize coordinate T_i for each i-th particle ($i=\overline{1,m}$) (every kernel function type must be corresponded by equal number of particles), other coordinates of the i-th particle ($i=\overline{1,m}$) must be generated randomly from the corresponding ranges: $x_i^1 \in [x_{\min}^{1T},\ x_{\max}^{1T}]$, $x_i^2 \in [x_{\min}^{2T},\ x_{\max}^{2T}]$ ($x_i^2=0$ under $T=1$ and $T=2$), $C_i \in [C_{\min}^T,\ C_{\max}^T]$. To initialize random velocity vector $v_i(v_i^1, v_i^2, v_i^3)$ of the i-th particle ($i=\overline{1,m}$) ($v_i^2=0$ under $T=1$ and $T=2$). To establish initial position of the i-th particle ($i=\overline{1,m}$) as its best known position $(\hat{T}_i, \hat{x}_i^1, \hat{x}_i^2, \hat{C}_i)$, to determine the best particle with coordinates' vector $(\tilde{T}, \tilde{x}^1, \tilde{x}^2, \tilde{C})$ from all the m particles, and to determine the best particle for each kernel function type T, including in a search, with coordinates' vector $(\overline{T}, \bar{x}^{1T}, \bar{x}^{2T}, \overline{C}^T)$. Number of executed iterations must be considered as 1.

Step 3. To execute while the number of iterations is less than the fixed number $N_{\max}$:

– «regeneration» of particles: to choose p% of particles which represent the lowest quality of classification from particles with coordinate $T_i \neq \tilde{T}$ ($i=\overline{1,m}$); to change coordinate T_i (with the kernel function type) on $\tilde{T}$; to change values of the parameters x_i^1, x_i^2, C_i of «regenerated» particles to let them correspond to a new kernel function type $\tilde{T}$ (within the scope of the corresponding ranges);

– correction of velocity vector $v_i(v_i^1, v_i^2, v_i^3)$ and position (x_i^1, x_i^2, C_i) of the i-th particle ($i=\overline{1,m}$) using formulas:

$$v_i^j = \begin{cases} \chi \cdot [v_i^j + \hat{\varphi} \cdot \hat{r} \cdot (\hat{x}_i^j - x_i^j) + \tilde{\varphi} \cdot \tilde{r} \cdot (\bar{x}^{jT} - x_i^j)], & j=1,\ 2, \\ \chi \cdot [v_i^j + \hat{\varphi} \cdot \hat{r} \cdot (\hat{C}_i - C_i) + \tilde{\varphi} \cdot \tilde{r} \cdot (\overline{C}^T - C_i)], & j=3, \end{cases} \quad (10)$$

$$x_i^j = x_i^j + v_i^j \text{ for } j=1,\ 2, \quad (11)$$

$$C_i = C_i + v_i^3, \quad (12)$$

– where $\hat{r}$ and $\tilde{r}$ are random numbers in interval (0, 1), χ is a compression ratio calculated using the formula (7); formula (10) is the modification of formula (6): the coordinates' values $\bar{x}^{1T}, \bar{x}^{2T}, \overline{C}^T$ are used instead of the coordinates' values $\tilde{x}^1, \tilde{x}^2, \tilde{C}$ of the globally best particle;

– accuracy calculation of the SVM classifier with parameters' values (T_i, x_i^1, x_i^2, C_i) ($i=\overline{1,m}$) with aim to find the optimal combination $(\tilde{T}, \tilde{x}^1, \tilde{x}^2, \tilde{C})$, which will provide high quality of classification;

– increase of iterations number on 1.

The particle with the optimal combination of the parameters' values $(\tilde{T}, \tilde{x}^1, \tilde{x}^2, \tilde{C})$ which provides the highest quality of classification on chosen the function types will be defined after execution of the offered algorithm.

After executing of the modified PSO algorithm it can be found out that all particles will be situated in the search space which corresponds to the kernel function with the highest classification quality because some particles in the modified PSO algorithm changed their coordinate, which is responsible for number of the kernel function. A herewith all other search spaces will turn out to be empty because all particles will «regenerate» their coordinate with number of the kernel function type. In some cases (for small value $N_{\max}$ and for small value p) some particles will not «regenerate» their kernel function type and will stay in their initial search space.

The modified PSO algorithm allows reducing the time expenditures for development of the SVM classifier.

IV. The Support Vector Machine Ensemble

In most cases SVM classifier based on the modified PSO algorithm provides high quality of data classification. In

exceptional cases the SVM ensembles can be used to increase the classification accuracy. The using of the SVM ensemble allows fulfilling the high-precision data classification, especially Big Data classification, with the acceptable time expenditures.

After training, each classifier generates its own (individual) classification decisions, same or different from the actual results of classification. Accordingly, the different individual SVM classifiers correspond to the different classification accuracy. The quality of the received classification decisions can be improved on the base of ensembles of the SVM classifiers [3], [33–36]. In this case, the finite set of individually trained classifiers must be learned. Then the classification decisions of these classifiers are combined. The resulting solution is based on the aggregated classifier. The majority vote method and the vote method based on the degree of reliability can be used as the rules (strategies) of the definition of the aggregated solutions.

The majority vote method is one of the most common and frequently used methods for combining decisions in the ensemble of classifiers. But this method does not fully use the information about the reliability of each individual SVM classifier. For example, suppose that the SVM classifier ensemble aggregates the results of five individual SVM classifiers, where values of the function $f(z)$ of the object z (3) obtained from the three individual SVM classifiers, are negative (class –1), but very close to the neutral position, and values of the function $f(z)$ of the other two SVM classifiers are strongly positive (class +1), i.e. very far away from the neutral position. Then the result of the aggregated decision of the ensemble on the basis of «one classifier – one vote» is following: the object z belongs to the negative class (majority vote), although it is obvious, that the best and more appropriate choice for the object z is a positive class. Despite the good potential of the majority vote method for combining of the group of decisions, it is recommended to use other methods to increase the accuracy of classification.

Vote method based on the degree of reliability uses value of the function $f(z)$ for the object z obtained by each individual SVM classifier. The greater the positive value of $f(z)$ in (3) returned by the SVM classifier, the more precisely the object z is determined in class +1, and the less negative value $f(z)$, the more precisely the object z is defined in class –1. Values «–1» and «+1» for $f(z)$ indicate that the object z is situated on the boundary of the negative and positive classes, respectively.

When using an ensemble of classifiers for solving classification problems special attention should be paid to the methods of forming a set of individual classifiers, which can later be used in the development of the final SVM classifier. It is experimentally confirmed [3], [33–37], that the ensemble of classifiers shows better accuracy than any of its individual members, if individual classifiers are accurate and varied. Therefore, the formation of the set of the individual SVM classifiers is required: 1) to use the various kernel functions; 2) to build classifiers in the different ranges of change of the kernel parameters and regularization parameter; 3) to use various sets of training and test data. To select the appropriate members of the ensemble in the set of the trained SVM classifiers it is recommended to use the principle of maximum decorrelation. In this case the correlation between the selected classifications should be as small as possible. After training, each private j-th classifier from the k trained classifier will correspond to a certain array of errors: $e_{ij} = | y_{ij} - \tilde{y}_{ij} |$, where e_{ij} is the error of j-th classifier at i-th row of the experimental data set ($i = \overline{1,s}$; $j = \overline{1,k}$); y_{ij} is the classification decision (−1 or +1) of j-th classifier at i-th row of the experimental data set; $\tilde{y}_{ij}$ is the real meaning of a class (−1 or +1), for which the i-th object is belong to.

The SVM classifiers not permitting an error on the experimental data set should be excluded from further consideration and from the remaining quantity of the SVM classifiersIt is necessary to select an appropriate number of individual SVM classifiers with maximal variety. To solve this problem decorrelation maximization algorithm can be used. This algorithm provides a variety of individual SVM classifiers, being used in the construction of the ensemble [3]. If the correlation between the selected classifiers is small, then the decorrelation is maximal.

Let there be an error matrix E of set of individual SVM classifiers with size $s \times k$:

$$E = \begin{bmatrix} e_{11} & e_{12} & \dots & e_{1k} \\ e_{21} & e_{22} & \dots & e_{2k} \\ \dots & \dots & \dots & \dots \\ e_{s1} & e_{s2} & \dots & e_{sk} \end{bmatrix}, \quad (13)$$

where e_{ij} is the error of the j-th classifier at the i-th row of the experimental data set ($i = \overline{1,s}$; $j = \overline{1,k}$).

On the basis of the error matrix E (13) the following assessments can be calculated [3]:

– mean:

$$\bar{e}_j = \frac{1}{s} \sum_{i=1}^{s} e_{ij} \ \ (j = \overline{1,k}); \quad (14)$$

– variance:

$$\mathrm{V}_{jj} = \frac{1}{s} \sum_{i=1}^{s} (e_{ij} - \bar{e}_j)^2 \ \ (j = \overline{1,k}); \quad (15)$$

– covariance:

$$\mathrm{V}_{tj} = \frac{1}{s} \sum_{i=1}^{s} (e_{ij} - \bar{e}_j) \cdot (e_{it} - \bar{e}_t) \ \ (j = \overline{1,k}, t = \overline{1,k}); \quad (16)$$

Then the elements r_{tj} of the correlation matrix with size $k \times k$ are calculated as:

$$r_{tj} = V_{tj} \Big/ \sqrt{V_{tt} \cdot V_{jj}} \ ; \quad (17)$$

where r_{tj} is the correlation coefficient, representing the degree of correlation of t-th and j-th classifiers ($j=\overline{1,k}$; $t=\overline{1,k}$); $r_{jj}=1$ ($j=\overline{1,k}$).

Using the correlation matrix R it is possible for each individual j-th classifier to calculate the plural-correlation coefficient ρ_j, which characterizes the degree of correlation of j-th and all other $(k-1)$ classifiers with numbers t ($t=\overline{1,k}$; $t\neq j$) [3]:

$$\rho_j=\sqrt{1-|R|/R_{jj}}\ (\ j=\overline{1,k}\ ), \tag{18}$$

where $|R|$ is the determinant of the correlation matrix R; R_{jj} is the cofactor of the element r_{jj} of the correlation matrix R.

A quantity ρ_j^2 called the coefficient of determination. It shows the proportion of the variation of the analyzed variable, which is explained by variation of the other variables. The coefficient of determination ρ_j^2 can take values from 0 to 1. The closer the coefficient to 1, the stronger the relationship between the analyzed variables (in this case, between individual classifiers) [3]. It is believed that there is a dependency, if the coefficient of determination is not less than 0.5. If the coefficient of determination greater than 0.8, it is assumed that high dependence exists.

For selection of individual SVM classifiers for integration into the ensemble it is necessary to determine the threshold θ. Thus, the j-th individual classifier must be removed from the list of classifiers if the coefficient of determination ρ_j^2 satisfies to condition $\rho_j^2>\theta$ ($j=\overline{1,k}$). If it is necessary to identify the most various classifiers, generating decisions with the most different arrays of errors on the experimental data set, thresholds θ, satisfying to condition $\theta<0.7$ should be selected. The additional considerations can be also taken into account to avoid the exclusion of insufficient or excessive number of individual SVM classifiers.

The decorrelation maximization algorithm can be summarized into the following steps [3].

Step 1. To calculate the matrix V and the correlation matrix R with formulas (15), (16) and (17) respectively.

Step 2. To calculate the multiple correlation coefficients ρ_j ($j=\overline{1,k}$) with (18) for all classifiers.

Step 3. To remove classifiers, for which $\rho_j^2>\theta$ ($j=\overline{1,k}$), from the list of classifiers.

Step 4. To repeat iteratively steps 1 – 3 for the remaining classifiers in the list until for all classifiers the condition $\rho_j^2\leq\theta$ ($j=\overline{1,k}$) will not satisfied.

As a result, the list of classifiers used to form the ensemble will consist of m ($m\leq k$) individual classifiers.

For classifiers selected in the ensemble, it is necessary to carry out:

- the rationing of degrees of the reliability;
- the strategy search for the integration of members of the ensemble;
- the calculation of the aggregated decision of the ensemble.

Value of the reliability $f_j(z)$, which is defined for the object z by the j-th classifier, falls into the interval (-∞, +∞). The main drawback of such values is that in the ensemble the individual classifiers with large absolute value are often dominated in the final decision of the ensemble. To overcome this drawback, the rationing is carried out: the transformation of values of degrees of reliability in the interval [0; 1] is fulfilled. In the case of binary classification in the rationalization for the object z the values of the reliability of its membership to positive class (labeled +1) $g_j^+(z)$ and to negative class $g_j^-(z)$ are determined. These values can be determined by the formulas [3]:

$$g_j^+(z)=\frac{1}{1+e^{-f_j(z)}}, \tag{19}$$

$$g_j^-(z)=1-g_j^+(z). \tag{20}$$

The selected individual classifiers are combined into the ensemble using $g_j^+(z)$ and $g_j^-(z)$ ($j=\overline{1,m}$) in accordance with one of the following five strategies [3].

1) Maximum strategy:

$$A(z)=\begin{cases}1, & \text{if } \max\limits_{j=1,m} g_j^+(z)\geq \max\limits_{j=1,m} g_j^-(z),\\ -1, & \text{otherwise.}\end{cases} \tag{21}$$

2) Minimum strategy:

$$A(z)=\begin{cases}1, & \text{if } \min\limits_{j=1,m} g_j^+(z)\geq \min\limits_{j=1,m} g_j^-(z),\\ -1, & \text{otherwise.}\end{cases} \tag{22}$$

3) Median strategy:

$$A(z)=\begin{cases}1, & \text{if } \frac{1}{m}\sum\limits_{j=1}^{m} g_j^+(z)\geq \frac{1}{m}\sum\limits_{j=1}^{m} g_j^-(z),\\ -1, & \text{otherwise.}\end{cases} \tag{23}$$

4) Mean strategy:

$$A(z)=\begin{cases}1, & \text{if } \sum\limits_{j=1}^{m} g_j^+(z)\geq \sum\limits_{j=1}^{m} g_j^-(z),\\ -1, & \text{otherwise.}\end{cases} \tag{24}$$

5) Product strategy:

$$A(z)=\begin{cases}1, & \text{if } \prod\limits_{j=1}^{m} g_j^+(z)\geq \prod\limits_{j=1}^{m} g_j^-(z),\\ -1, & \text{otherwise.}\end{cases} \tag{25}$$

The value $A(z)$ is an aggregated measure of the reliability's value of the SVM classifier ensemble. It can be used to integrate the members of the ensemble [3].

The learning algorithm of the ensemble of the SVM classifiers can be summarized into the following steps.

Step 1. To divide the experimental data set into k training data sets: TR_1, …, TR_k.

Step 2. To learn k individual SVM classifiers with the different training data sets TR_1, …, TR_k and to obtain k individual SVM classifiers (ensemble members).

Step 3. To select m ($m \le k$) SVM classifiers from k classifiers using the decorrelation maximization algorithm.

Step 4. To determine values of m classification functions for each selected individual SVM classifier: $f_1(z)$,…, $f_m(z)$.

Step 5. To transform values of degrees of reliability, using (19) and (20), for the positive class $g_1^+(z)$,…, $g_m^+(z)$ and for the negative class $g_1^-(z)$, … , $g_m^-(z)$.

Step 6. To determine the aggregated value $A(z)$ of the reliability of the SVM classifier ensemble using (21) – (25).

This algorithm, used for the weak SVM classifiers, will provide a better quality of the classification accuracy than accuracy of any single individual classifier used for aggregation.

The problem of choosing of the threshold θ is very important. Value θ for which all five rules of classification (21) – (25) show stable improvement of the classification quality must be chosen as the threshold value θ^* ($\theta^* < 0.7$). Thus the use of each of the five rules leads to improvement of the classification quality resulting in the reduction of the number of erroneous decisions, when the smaller number of individual classifiers, corresponding to the threshold value θ^*, is applied. Such stable improvement of the classification quality isn't observed for all examined values θ' (for which $\theta' > \theta^*$).

It should be noted, that the majority vote rule may be used for decisions, obtained using the classification rules (21) – (25), to determine the required threshold value θ^*.

V. Two-Level SVM Classifier

The main problem which limits the use of the PSO algorithm is associated with quite a lot of time required to search for the optimal parameters of the SVM classifier (the kernel function type, the values of the kernel function parameters and the value of the regularization parameter). The search time can be partly reduced by using a small number of particles in the swarm and a small number of iterations of the PSO algorithm. But in this case we limit the number of the generated and compared SVM classifiers, and will probably find the worst decision.

One approach to reducing the search time is associated with the reduction in the size of the training data set. A herewith those objects that will not affect the classification results shouldn't be considered. This approach is based on the following theoretical fact of the development of the SVM classifier: the classification function (3) performs the summation only for the support vectors for which $\lambda_i \neq 0$. These vectors contain all the information about the objects division and play the main role in the construction of the hyperplanes separating the classes.

Therefore the two-level SVM classifier has been developed. This SVM classifier works as the group of the SVM classifiers at the first level and as the SVM classifier on the base of the modified PSO algorithm at the second level.

This two-level SVM classifier is iteratively trained on subsets of the original experimental data set at the the first level. Then support vectors of the obtained SVM classifiers are combined to create the new training set for the SVM classifier on the base of the modified PSO algorithm.

The proposed approach to can be described by the following consequence of steps.

1) To train k SVM classifiers on the original experimental data set using different training data sets TR_1, TR_2 , …, TR_k at the first level of the two-level SVM classifier. A herewith it is necessary to use the different kernel functions types, the different values of the kernel function parameters and the regularization parameter.

2) To obtain the support vectors sets SV_1, SV_2, …, SV_k from the trained SVM classifiers and form the set SV as the union of the support vectors sets SV_1, SV_2, …, SV_k. Let this set SV consists of t objects ($t \le s$, where s is the number of objects in the experimental data set).

3) To select from the set SV the subset SV^+, consisting of T ($T \le t$) objects (support vectors), which have been correctly classified by the SVM classifiers. It is necessary to ensure that false data is not participated in the training of the SVM classifier on the base of the modified PSO algorithm at the second level of the two-level SVM classifier. The rest objects (support vectors) from the set SV will form the subset SV^- with $t-T$ objects. The subset SV^+ will be used for the training and the subset SV^- will be used for the testing of the SVM classifier on the base of the modified PSO algorithm.

4) To develop the SVM classifier on the base of the modified PSO algorithm.

5) To classify objects (from the original experimental data set) which not included in the sets SV^+ and SV^-.

The using of the two-level SVM classifier also allows carrying out the high-precision data classification, especially Big Data classification, with the acceptable time expenditures.

VI. Experimental Studies

The assessment of the offered approaches for the development of the SVM classifiers and their ensembles has been carried out by test and real data.

In the first experiments for the particular data set the traditional PSO algorithm and the modified PSO algorithm have been applied. Comparison between these algorithms was executed using the found optimal parameters values of the SVM classifier, classification accuracy and spent time. All data sets used in the experimental researches we taken from the Statlog project and from the UCI library for machine learning.

Particularly, we used two data sets for medical diagnostics, two data sets for credit scoring and one data set for the creation of the predictive model of the spam recognition on the base of the e-mails' data set:

- Breast cancer data set of The Department of Surgery at the University of Wisconsin, in which the total number of instances is 569 including 212 cases with the diagnosed cancer (class 1) and 357 cases without such diagnosis (class 2); a herewith each patient is described by 30 characteristics ($q = 30$) and all information was obtained with the use of digital images (WDBC data set in the Tabl. I, the source is http://archive.ics.uci. edu/ml/machine-learning-databases/breast-cancer-wisconsin/);
- Heart disease data set, in which the total number of instances is 270 including 150 cases with the diagnosed heart disease (class 1) and 120 cases without such diagnosis (class 2); a herewith each patient is described by 13 characteristics ($q = 13$) (Heart data set in the Tabl. I, the source is http://archive.ics.uci.edu/ ml/machine-learning-databases/statlog/heart/; a herewith desease was found for 150 patients (class 1) and desease was not found for 120 patients (class 2));
- Australian consumer credit data set, in which the total number of instances is 690 including 382 creditworthy cases (class 1) and 308 default cases (class 2); a herewith each applicant is described by 14 characteristics ($q = 14$) (Australian data set in the Tabl. I, the source is http://archive.ics.uci.edu/ ml/machine-learning-databases/ statlog/australian/);
- German credit data set, in which the total number of instances is 1000 including 700 creditworthy cases (class 1) and 300 default cases (class 2); a herewith each applicant is described by 24 characteristics ($q = 24$) (German data set in the Tabl. I; the source is http://archive.ics.uci.edu/ml/machine-learning-databases/statlog/german/);
- Spam data set, in which in which the total number of instances is 4601 including 1813 cases with the spam (class 1), that is equal to 39.4% of the data set size, and 2788 cases without the spam (class 2); a herewith each e-mail is described by 57 characteristics ($q = 57$) (Spam data set in the Tabl. I; the source is https://archive.ics.uci.edu/ml/machine-learning-databases/spambase/).

The Spam data set we consider as an example of the Big Data. It is logical, especially, if we plan to use the developed SVM-classifier for the identification of the new spam patterns in a data flow.

Also, we used two test data sets Test1 (Test1 data set in the Tabl. I) and Test2 (Test2 data set in the Tabl. I; the source is http://machinelearning.ru/wiki/images/b/b2/MOTP12_svm_example.rar) [26].

For all data sets binary classification has been performed.

Experimental calculations were made on the base of PC under the Microsoft Windows 7 for x64-based Operating System with the random access memory of 3 GB and the four-nuclear Intel® Core™ i3 processor with the kernels' clock frequency of 2.53 GHz. The SVM algorithm from the software package MATLAB 7.12.0.635 was applied for the modeling.

For development of the SVM classifier the traditional and the modified PSO algorithms were used, meaning that the choice of the optimal values of the SVM classifier parameters was conducted. The kernels with polynomial (#1), radial basis (#2) and sigmoid (#3) functions were included in the search and the identical values of the PSO algorithm parameters and the identical ranges of values' change of the required SVM classifier parameters were established.

The short description of characteristics of each data set is provided in the Table I. Here the search results of the optimal values of parameters of the SVM classifier with the application of the traditional PSO algorithm and the modified PSO algorithm are presented (in the identical ranges of parameters' change and at the identical PSO algorithm parameters), number of error made during the training and testing of the SVM classifier and search time.

TABLE I. THE SEARCH RESULTS BY MEANS OF THE TRADITIONAL PSO ALGORITHM AND THE MODIFIED PSO ALGORITHM

Data set	**Number of objects**	**Number of characteristics**	**PSO algorithm type**	**Found parameters**				**Errors**		**Number of the support vectors**	**Accuracy (%)**	**Search time (sec.)**
				Kernel number	C	x_1	x_2	***At the training***	***At the testing***			
Test1	300	2	traditional	1	7.97	3	-	0 of 240	0 of 60	6	100	1478
			modified	1	2.83	8	-	0 of 240	0 of 60	6	100	382
Heart	270	13	traditional	2	9.6	3.34	-	7 of 230	7 of 40	106	94.81	2276
			modified	2	6.01	2.99		6 of 230	3 of 40	131	96.67	876
WDBC	569	30	traditional	2	9.36	2.89	-	0 of 427	3 of 142	113	99.47	3919
			modified	2	9.84	3.99	-	0 of 427	2 of 142	79	99.65	1464
Australian	690	14	traditional	2	9.28	2.51	-	23 of 518	25 of 172	248	93.04	9086
			modified	2	4.45	2.22		26 of 518	20 of 172	258	93.33	2745
German	1000	24	traditional	1	1.58	3	-	0 of 850	42 of 150	438	95.8	14779
			modified	1	5.53	4	-	0 of 850	42 of 150	546	95.8	5766

Test2	400	2	traditional	2	6.31	0.19	-	7 of 340	8 of 60	156	96.25	15632
			modified	2	5.69	0.22	-	11 of 340	4 of 60	146	96.25	7146
Spam	4601	57	traditional	2	7.82	2.47	-	40 of 3681	56 of 920	1634	97.91	92645
			modified	2	8.57	2.45	-	36 of 3681	60 of 920	1659	97.91	44933

For example, for the WDBC data set with the use of the traditional and the modified PSO algorithms the kernel with radial basis function (#2) was determined as the optimal. For the traditional PSO algorithm the optimal values of the kernel parameter and the regularization parameter are equal to $\sigma = 2.89$ and $C = 9.36$ accordingly. For the modified PSO algorithm the optimal values of the kernel parameter and the regularization parameter are equal to $\sigma = 3.99$ and $C = 9.84$ accordingly.

The classification accuracy by the traditional PSO algorithm is equal to 99.47%, and the classification accuracy by the modified PSO algorithm is equal to 99.65%. The search time came to 3919 and 1464 seconds accordingly.

For the Spam data set with the use of the traditional and the modified PSO algorithms the kernel with radial basis function (#2) also was determined as the optimal. For the traditional PSO algorithm the optimal values of the kernel parameter and the regularization parameter are equal to $\sigma = 2.47$ and $C = 7.82$ accordingly. For the modified PSO algorithm the optimal values of the kernel parameter and the regularization parameter are equal to $\sigma = 2.45$ and $C = 8.57$ accordingly.

The classification accuracy by the traditional and modified PSO algorithm is equal to 97.91%. The search time came to 92645 and 44933 seconds accordingly.

Fig. 2–4 show for the Spam data set the location examples of the particles swarm in the D-2 search spaces and in the D-3 search space at the initialization, at the 3-rd iteration and at the 12-th iteration. These locations of the particles in the swamp were obtained with the use of the modified PSO algorithm.

The kernels with polynomial, radial basis and sigmoid functions were included in the search. A herewith the following change ranges of values' parameters were set: $3 \le d \le 8$, $d \in N$ (for the polynomial function); $0.1 \le \sigma \le 10$ (for the radial basis function); $-10 \le k_2 \le -0.1$ and $0.1 \le k_1 \le 10$ (for the sigmoid function).

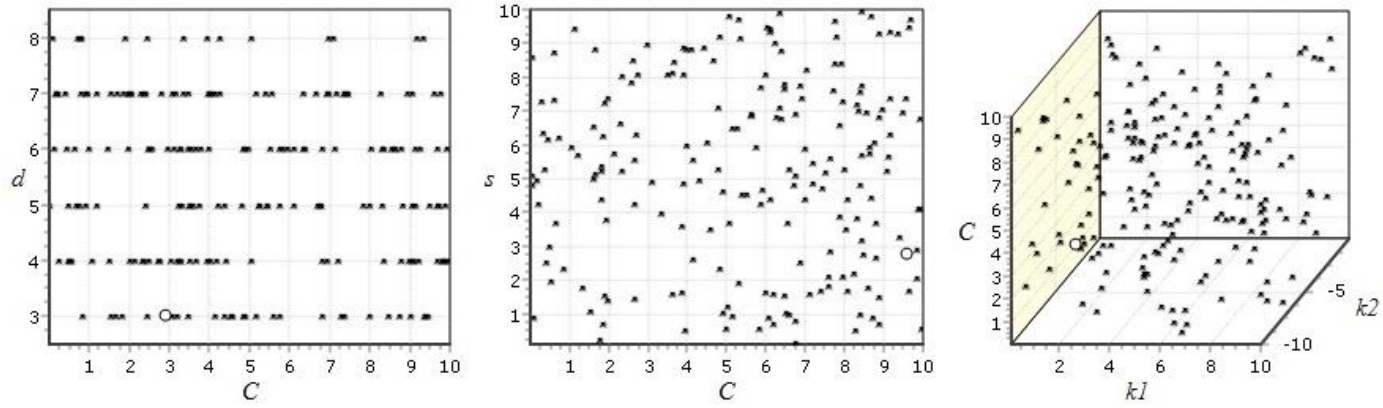


Fig. 2. Location of the particles in the swamp at the initialization (polynomial kernel function is on the left, radial basis is in the middle, sigmoid is on the right)

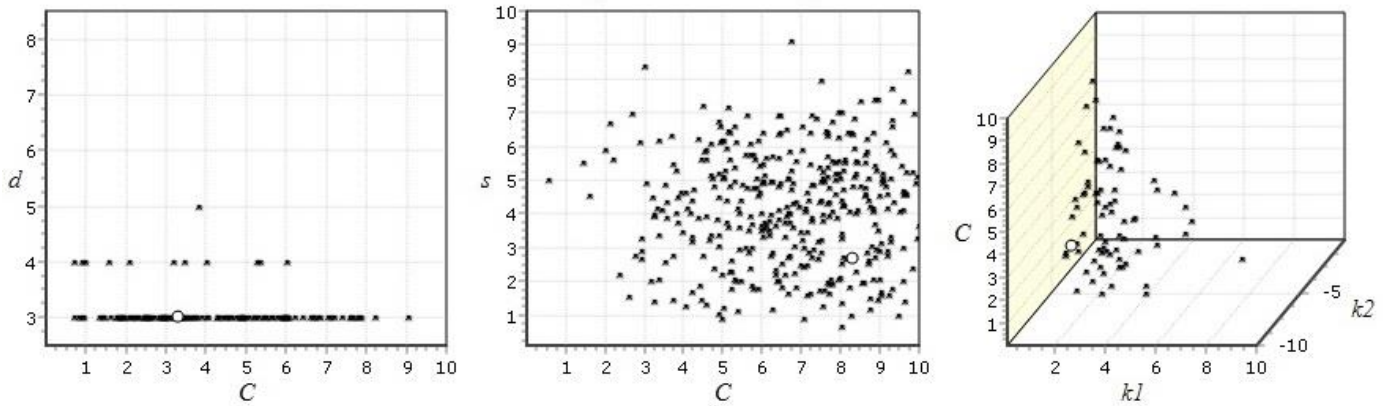


Fig. 3. Location of the particles in the swamp at the 3-rd iteration (polynomial kernel function is on the left, radial basis is in the middle, sigmoid is on the right)

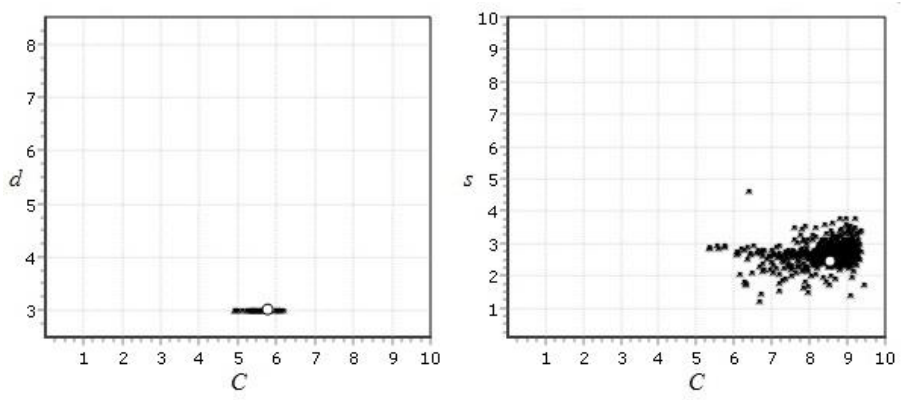


Fig. 4. Location of the particles in the swamp at the 12-th iteration (polynomial kernel function is on the left, radial basis is on the right)

Change range for the regularization parameter C was determined as: $0.1 \le C \le 10$. Moreover, the following values of parameters of the PSO algorithm were set: number m of particles in a swarm equal to 600 (200 per each kernel function type); iterations' number $N_{\max} = 20$; personal and global velocity coefficients equal to $\hat{\varphi} = 2$ and $\tilde{\varphi} = 5$ accordingly; the scaling coefficient $K = 0.3$; «regeneration» coefficient of particles $p = 20\%$.

Particles are marked by asterisk bullets in the search spaces and the best position from the search space is marked by white round bullet. During realization of the modified PSO algorithm the swamp particles moves towards the best (optimal) position for the current iteration in the search space and demonstrate collective search of the optimal position. A herewith velocity and direction of each particle are corrected. Moreover «regeneration» of particles takes place: some particles change own search space to space, in which particles show the best quality of classification.

Thus, at the realization of the modified PSO algorithm there is a change of the particles' coordinates, which are responsible for the parameters of the kernel function $\kappa(z_i, z_\tau)$ and the regularization parameter C. Besides, the type of the kernel function also changes. As a result the particles moves towards the united search space (in this case – the space corresponding to the radial basis kernel function) leaving the space where they were initialized.

In the reviewed example only 7 particles didn't change their kernel function type after 20 iterations. Other particles situated near the best position responsible for the optimal solution in the search space (Fig. 5).

Fig. 6 shows the location examples of the particles swarm in the D-2 search space at the initialization and at the 2-nd, the 7-th and the 10-th iterations of the traditional PSO algorithm for the radial basis kernel function. The best particle has been found at the 8-th iteration, though 20 iterations have been executed.

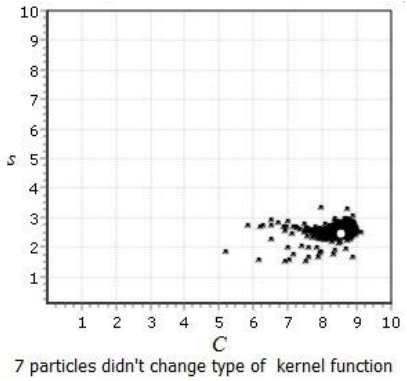
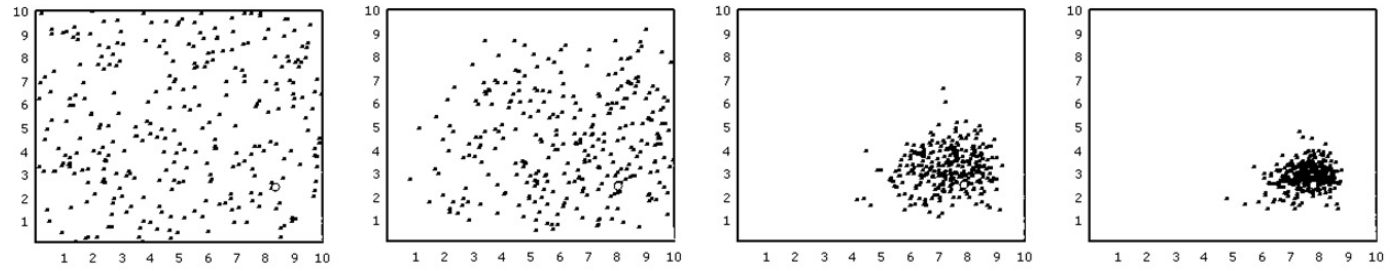


Fig. 5. Location of particles after the 20-th iteration



(a) (b) (c) (d)

Fig. 6. Location of the particles in the swamp: a) at the initialization; b) at the 2-nd iteration; c) at the 7-th iteration; d) at the 10-th iteration (the horizontal axis corresponds to the regularization parameter C, the vertical axis corresponds to the parameter σ)

TABLE II. THE CHARACTERISTICS OF THE BEST CLASSIFIER AT THE REALIZATION OF THE PSO ALGORITHM

Traditional PSO algorithm (with the radial basis kernel function)					**Modified PSO algorithm**				
Performance stage	**Errors**		**Total quantity of error**	**Number of the support vectors**	**Performance stage**	**Errors**		**Total quantity of error**	**Number of the support vectors**
	at the training	***at the testing***				***at the training***	***at the testing***		
Initialization of the swarm	45	85	130	1483	Initialization, the 1-st and 2-nd iterations	45	56	101	1494
The 1-st iteration	41	88	129	1648	The 3-d iterations	43	57	100	1582
The 2-nd iteration	56	73	129	1264	The 4-th iterations	38	61	99	1592
The 3-d and the 4-th iterations	54	73	127	1240	The 5-th and the 6-th iterations	38	61	99	1584
The 5-th iteration	54	73	127	1240	The 7-th iteration	37	61	98	1671
The 6-th iteration	40	58	98	1645	The 8-th iteration	37	61	98	1589
The 7-th iteration	35	61	96	1686	The 9-th , the 10-th and 11-th iterations	38	59	97	1602
From the 8-th to 20-th iteration	40	56	96	1634	From the 12-th to 20-th iteration	36	60	96	1659

The kernels with polynomial, radial basis and sigmoid functions were included in the search. The following change ranges of values' parameters were set: $3 \le d \le 8$, $d \in N$ (for the polynomial function); $0.1 \le \sigma \le 10$ (for the radial basis function); $-10 \le k_2 \le -0.1$ and $0.1 \le k_1 \le 10$ (for the sigmoid function.

Table II shows the information on the best SVM classifier at the different iterations of the traditional PSO (for the radial basis kernel function, which was defined as the best kernel function) and modified PSO algorithm (for three kernel functions) for the Spam data set.

It is visible from the Table I, that as a result of the search for the reviewed data sets both algorithms determined identical kernel function type as the optimal, similar values of the kernel function parameter and the regularization parameter, and also similar accuracy values of training and testing of the SVM classifier.

But the modified PSO algorithm is more effective, because it took less (more than 2 – 3 times) time for search compared to the traditional one.

At the determination of the optimal parameters' values of the SVM classifier with use of the traditional or modified PSO algorithm in the chosen search space we must create the huge number of the SVM classifiers to figure out, which shows the maximum classification accuracy under the minimum number of the support vectors. Therefore at the implementation of the PSO algorithm with 600 particles under 20 iterations of the PSO algorithm it is necessary to build and compare 12000 SVM classifiers.

If the average time of the training and testing of the SVM classifier equal to 5 seconds, then the time expenditures for the search of the optimal parameters' values of the SVM classifier will be $600 \times 20 \times 5 = 60000$ seconds or about 16.67 hours, that considerably surpasses the time expenditures for the development of 18 SVM classifiers (90 seconds) under the SVM ensemble development.

The experimental studies show, that the search time is defined: a) by the own parameters of the PSO algorithm (the speed coefficients, the quantity of the kernel functions and the types of the kernel functions, the search ranges, etc.); b) by properties of the experimental data set used for the training and testing of the SVM classifier (in particular, by the size of the data set and the number of characteristics). The lesser search time of the modified PSO algorithm in comparison with the search time of the traditional PSO algorithm is explained by the fact that some particles "regenerate" from the one search space (with the one kernel function type) to another search space (with the another kernel function type). The time expenditures for the SVM classifier development for the first kernel function type are more expensive than for the second kernel function type (in particular, the most expensive on time is the polynomial kernel function).

It should be noted that the SVM classifier for the German data set doesn't have really good classification accuracy assessment (in comparison with the SVM classifiers for other data sets). The attempt of the SVM classifier training in this case leads to the SVM classifier with not really high classification accuracy or to the retraining of the SVM classifier when the number of error for the test set is significantly more, than for the training set (with the acceptable classification accuracy for the experimental data set in general). Therefore, it is expedient to try to use other approaches to the classifier development, in particular, the approach based on the SVM ensemble development.

One more reason to use SVM ensemble is the realization of the PSO algorithm, which deals with the high time expenditures: to increase the classification accuracy we need to increase the number of the PSO algorithm iterations or/and number of particles in the swarm, however it doesn't guarantee that the expected high classification accuracy will be obtained. Therefore it is necessary to try to develop the SVM ensemble on the base of the individual SVM classifiers with not really high classification accuracy. The classification accuracy of the SVM ensemble should have higher classification accuracy than the classification accuracy of the used individual SVM classifiers.

In the last experiments the usefulness of the SVM ensembles was confirmed with application of test and real data sets.

Several individual SVM classifiers using different types of the kernel function, different values of the kernel function functions of the kernel parameters and different values of the regularization parameter were learned in experiments for the particular data sets. The different training and test sets randomly generated from the original data set were used. Then the decorrelation maximization algorithm for the different

strategies of the decision-making on the data classification and the majority vote rule were applied.

For example, for the German data set we developed 18 individual SVM classifiers with use of various input parameters.

At the testing it was found, that the individual classifiers indicate the classification accuracy in range from 83.5% to 93.2%, and the initial values of the determination coefficient (if $\theta^* = 1$), calculated for all 18 individual classifiers, are in the range from 0.049 to 0.534. As a result, the threshold values θ were examined from the range [0.1; 0.55] with step 0.05. Values of the classification parameters corresponding to the different threshold values θ are given in the Table III.

The optimal threshold value θ^* for the reviewed example is 0.3, since for $\theta^* = 0.3$ all five classification rules (strategies) (21) – (25) give the stable improvement of the classification quality when the number of classifiers reduces to the number corresponding to the threshold value $\theta^* = 0.3$. The finite number of classifiers in the SVM ensemble proved equal to 6. A further decrease in the number of classifiers is not feasible (due to a further sharp decrease in their number and a substantial reduction of their variety).

The use of the median strategy (or sum strategy) with $\theta^* = 0.3$ allowed classifying correctly 98.29% of the objects of the initial data set. At the same time, the maximum classification accuracy of one of the individual SVM classifiers, used in the SVM ensemble, was equal to 93.2%, and the accuracy reached with use of the majority vote rule was equal to 96.8%.

Thus, the use of the SVM ensemble allowed increasing the classification accuracy by more than 5% compared to the maximum classification accuracy of one of the individual classifiers in the SVM ensemble.

TABLE III. VALUES OF CLASSIFICATION PARAMETERS AT THE DIFFERENT THRESHOLD VALUES OF THE DETERMINATION COEFFICIENT (GERMAN DATA SET)

Value of classification	**Strategy**	**The threshold value of the determination coefficient**							
		0.55	**0.5**	**0.45/0.4**	**0.35**	**0.3**	**0.25**	**0.2**	**0.15/0.1**
Overall accuracy (%)	Majority vote	96.80	96.80	96.80	96.80	96.80	96.80	96.80	96.80
	Maximum and minimum	79.90	80.10	81.10	83.80	90.30	91.20	92.30	93.50
	Median and sum	95.20	96.20	95.60	97.20	98.29	97.70	97.80	97.60
	Product	87.40	89.10	89.00	90.90	97.10	97.30	97.00	96.10
Sensitivity (%)	Majority vote	98.00	98.00	98.00	98.00	98.00	98.00	98.00	98.00
	Maximum and minimum	84.71	84.86	85.86	88.00	94.57	95.57	96.29	96.71
	Median and sum	96.14	97.43	96.86	98.57	99.29	99.14	99.43	98.57
	Product	89.14	90.86	90.86	92.29	99.00	99.00	98.86	97.71
Specificity (%)	Majority vote	94.00	94.00	94.00	94.00	94.00	94.00	94.00	94.00
	Maximum and minimum	68.67	69.00	70.00	74.00	80.33	81.00	83.00	86.00
	Median and sum	93.00	93.33	92.67	94.00	95.67	94.33	94.00	95.33
	Product	83.33	85.00	84.67	87.67	92.67	93.33	92.67	92.33
Number of errors of the 1-st type	Majority vote	14	14	14	14	14	14	14	14
	Maximum and minimum	107	106	99	84	38	31	26	23
	Median and sum	27	18	22	10	5	6	4	10
	Product	76	64	64	54	7	7	8	16
Number of errors of the 2-nd type	Majority vote	18	18	18	18	18	18	18	18
	Maximum and minimum	94	93	90	78	59	57	51	42
	Median and sum	21	20	22	18	13	17	18	14
	Product	50	45	46	37	22	20	22	23
Number of classifiers in the ensemble		18	15	13	8	6	5	4	3

TABLE IV. THE PARAMETERS AND CHARACTERISTICS OF THE INDIVIDUAL CLASSIFIERS (SPAM DATA SET)

Number of the individual SVM classifier	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Kernel function type	1	1	1	1	1	2	2	2	2	2	2	2	2	3	3	3	3	4
Regularization parameter	1.2	3.1	2	1.3	2.5	4	0.5	4	10	15	12	10	0.25	0.5	0.5	0.2	3.4	1.2
Kernel parameters	3	3	4	4	3	0.2	0.4	0.6	5	10	10	15	0.5	0.15/ -1.00	0.50/ -1.50	0.20/ -2.20	0.30/ -0.80	-
Overall accuracy (%)	83.33	91.33	86.72	85.59	90.70	93.44	93.68	94.04	83.89	94.63	92.98	92.87	90.33	87.00	83.53	82.29	82.59	91.72
Number of the support vectors	578	606	618	549	516	3099	3156	2891	646	686	754	836	3444	585	385	1160	628	874
Size of the training set	3681	3681	3911	3681	3451	3451	3451	3451	3681	3221	3681	3681	3681	3911	3911	3681	3451	3911
Quantity of errors at the training	574	286	482	509	273	5	13	11	581	151	253	263	18	510	652	659	613	316
Size of the test set	920	920	690	920	1150	1150	1150	1150	920	1380	920	920	920	690	690	920	1150	690
Quantity of errors at the testing	193	113	129	154	155	297	278	263	160	96	70	65	427	88	106	156	188	65
Sensitivity of the classifier (%)	58.52	79.54	69.39	66.30	78.60	83.45	84.11	85.11	59.85	90.95	85.22	86.05	98.95	81.69	68.34	74.30	81.69	84.28

Specificity of the classifier (%)	99.46	99.00	97.99	98.13	98.57	99.93	99.89	99.86	99.53	97.02	98.03	97.31	84.72	90.46	93.40	87.48	83.18	96.56
Number of errors of the 1-st type	752	371	555	611	388	300	288	270	728	164	268	253	19	332	574	466	332	285
Number of errors of the 2-nd type	15	28	56	52	40	2	3	4	13	83	55	75	426	266	184	349	469	96
Initial determination coefficient	0.474	0.385	0.312	0.311	0.377	0.106	0.134	0.121	0.383	0.63	0.711	0.757	0.025	0.473	0.486	0.276	0.308	0.572
Development time, s	5	6	16	5	5	7	8	7	3	2	3	3	8	2	2	4	2	2

For the Spam data set we also developed 18 individual SVM classifiers with use of various input parameters.

The parameters and characteristics of 18 individual classifiers have been shown in the Table IV. The kernels with polynomial (#1), radial basis (#2), sigmoid (#3) and linear (#4) functions were included in the search. In the Table IV for the sigmoid kernel function the first number is k_1, the second number is k_2.

Also Table IV shows information on the time expenditures for the training of each individual SVM-classifier. The total time of the training is 90 seconds. At the training for each individual SVM-classifier the training set was formed in a random way on the base of the initial experimental data set of the e-mails. The number of instances in the test set was equal to 10%–25% of the initial number of instances in the initial experimental data set.

At the testing it was found, that the individual classifiers indicate the classification accuracy ranged from 82.29% to 94.63%, and the initial values of the determination coefficient (if $\theta^* = 1$), calculated for all 18 individual classifiers, are in the range from 0.025 to 0.757. As a result, the threshold values θ were examined from the range $[0.15;\ 0.8]$ with step 0.05. Values of the classification parameters corresponding to the different threshold values θ are given in the Table V.

The optimal threshold value θ^* for the reviewed example belongs to the range $[0.15;\ 0.25]$, since for the threshold values from this range all five classification rules (21) – (25) give the stable improvement of the classification quality when the number of classifiers reduces to the number corresponding to the threshold value θ^* from the range $[0.15;\ 0.25]$. The finite number of classifiers in the SVM ensemble proved is equal to 4. A further decrease in the number of classifiers is not feasible (due to a further sharp decrease in their number and a substantial reduction of their variety).

Table VI shows information on the characteristics of the individual SVM-classifier, which take a part in the SVM ensemble. This ensemble was created on the base of the strategies (21) – (25) for the threshold values θ^* from the range $[0.15;\ 0.25]$. Also Table VI shows information on the characteristics of the best SVM-classifier on the base of the traditional PSO algorithm and the modified PSO algorithm.

Use of the maximum (minimum) strategy allowed classifying correctly 98.59% of the objects in the initial data set. At the same time, the maximum classification accuracy of one of the individual SVM classifiers, used in the SVM ensemble, was equal to 94.04% (for the 13-th SVM classifier), and the accuracy reached with use of the majority vote rule was equal to 96.8%.

The application of other strategies also leads to increasing of the classification accuracy in comparison to the classification accuracy of the individual SVM classifiers, the classification accuracy on the base of the majority vote rule and the classification accuracy of the SVM classifier on the base of the PSO algorithm.

TABLE V. VALUES OF CLASSIFICATION PARAMETERS AT THE DIFFERENT THRESHOLD VALUES OF THE DETERMINATION COEFFICIENT (SPAM DATA SET)

Value of classification	**Strategy**	**The threshold value of the determination coefficient**							
		0.8	**0.7**	**0.6**	**0.5**	**0.4**	**0.35**	**0.3**	**0.25-0.15**
Overall accuracy (%)	Majority vote	95.44	95.44	95.44	95.44	95.44	95.44	95.44	95.44
	Maximum and minimum	84.35	84.35	84.35	84.35	84.22	84.13	88.05	98.59
	Median and sum	94.31	94.28	94.02	93.89	95.26	95.31	97.09	98.44
	Product	86.55	86.50	86.39	86.29	85.87	85.09	93.26	98.44
Sensitivity (%)	Majority vote	89.30	89.30	89.30	89.30	89.30	89.30	89.30	89.30
	Maximum and minimum	83.78	83.78	83.78	83.78	83.95	83.89	82.24	96.47
	Median and sum	86.82	86.43	85.77	85.22	88.36	89.41	94.04	96.14
	Product	84.28	84.34	84.17	84.00	84.56	84.17	91.01	96.14
Specificity (%)	Majority vote	99.43	99.43	99.43	99.43	99.43	99.43	99.43	99.43
	Maximum and minimum	84.72	84.72	84.72	84.72	84.44	84.29	91.82	99.96
	Median and sum	99.18	99.39	99.39	99.53	99.75	99.14	99.07	99.93
	Product	88.02	87.91	87.84	87.77	86.73	85.69	94.73	99.93
Number of errors of the 1-st type	Majority vote	194	194	194	194	194	194	194	194
	Maximum and minimum	294	294	294	294	291	292	322	64
	Median and sum	239	246	258	268	211	192	108	70
	Product	285	284	287	290	280	287	163	70
Number of errors of the 2-nd type	Majority vote	16	16	16	16	16	16	16	16
	Maximum and minimum	426	426	426	426	435	438	228	1
	Median and sum	23	17	17	13	7	24	26	2
	Product	334	337	339	341	370	399	147	2

Number of classifiers in the ensemble	18	16	15	14	11	8	5	4

TABLE VI. THE CLASSIFICATION RESULTS ON THE BASE OF THE INDIVIDUAL SVM CLASSIFIERS AND THEIR SVM ENSEMBLE

The classifier characteristics	**The SVM classifier on the base**		**The classifier number, which take a part in the SVM ensemble**				**Strategy**			
	of the traditional PSO algorithm	***of the modified PSO algorithm***	***6***	***7***	***8***	***13***	***Maximum and minimum***	***Median and sum***	***Product***	***Majority vote***
Overall accuracy (%)	97.91	97.91	93.44	93.68	94.04	90.33	98.59	98.44	98.44	95.44
Sensitivity of the classifier (%)	96.47	96.14	83.45	84.11	85.11	98.95	96.47	96.14	96.14	89.30
Specificity of the classifier (%)	98.85	99.07	99.93	99.89	99.86	84.72	99.96	99.93	99.93	99.43
Number of errors of the 1-st type	64	70	300	288	270	19	64	70	70	194
Number of errors of the 2-nd type	32	26	2	3	4	426	1	2	2	16
Determination coefficient (θ)	-	-	0.078	0.104	0.091	0.004	$\theta^* \in [0.15;\ 0.25]$			-

Thus, the use of the SVM ensemble allowed increasing the classification accuracy almost by 5% compared to the maximum classification accuracy of one of the individual classifiers in the SVM ensemble.

The SVM ensemble with 98.59% classification accuracy doesn't concede to the SVM classifier on the base of the modified PSO algorithm with 97.91% classification accuracy and strongly surpasses it at the minimization of the time expenditures.

The proposed two-level SVM classifier was used for the Test2 data set classification (Table I). It is evident that, despite the small volume ($s = 400$) and the number of characteristics ($q = 2$), the PSO algorithm finds the optimal parameters for the SVM classifier in quite a long time (longer than, for example, for the WDBC data set of 569 objects with 30 characteristics). This is due to the data being hard to separate. Fig. 7 shows the location of the data in the 2D space and its division into two classes. Objects of the first class are marked by asterisk bullets, objects of the second class are marked by plus bullets. It is obviously that it is very difficultly to draw the curve separating the classes.

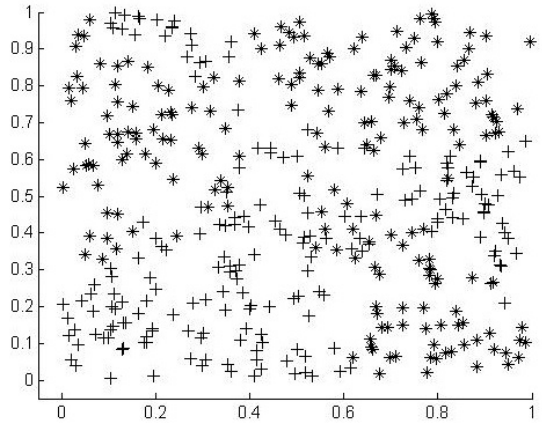


Fig. 7. Representation of the data set Test2 in 2D space

TABLE VII. THE PARAMETERS AND CHARACTERISTICS OF THE INDIVIDUAL CLASSIFIERS (TEST2 DATA SET)

Number of the individual SVM classifier	1	2	3	4	5	6	7	8	9	**Result**
Kernel function type	1	1	1	2	2	2	3	3	3	**2**
Regularization parameter	1	0.2	1.2	1.2	1.7	4.5	8.4	9	6.6	**8.5**
Kernel parameters	3	3	3	0.5	0.6	0.6	0.80; -3.00	0.50; -2.20	0.90; -2.50	**0.25**
Overall accuracy (%)	90.25	89.75	90.75	91.5	90.5	91	86.25	88.5	85.75	**96.75**
Number of the support vectors	84	112	96	121	106	87	150	150	103	**101**
Size of the training set	300	320	340	340	320	300	340	320	300	**204**
Quantity of errors at the training	27	29	30	25	27	23	46	35	40	**9**
Size of the test set	100	80	60	60	80	100	60	80	100	**11**
Quantity of errors at the testing	12	12	7	9	11	13	9	11	17	**4**
Size of the classified set	-	-	-	-	-	-	-	-	-	**185**
Quantity of errors at the classification	-	-	-	-	-	-	-	-	-	**0**
Sensitivity of the classifier (%)	84.39	85.37	86.34	90.24	91.22	91.22	79.51	83.9	79.02	**97.07**
Specificity of the classifier (%)	96.41	94.36	95.38	92.82	89.74	90.77	93.33	93.33	92.82	**96.41**
Number of errors of the 1-st type	7	11	9	14	20	18	13	13	14	**7**
Number of errors of the 2-nd type	32	30	28	20	18	18	42	33	43	**5**
Development time, s	<1	<1	<1	1	1	1	1	1	1	**2465**

For this data set the group of 9 SVM classifiers was trained (Table VII). Three kernel functions were included in the search: polynomial (# 1), radial basis (# 2) and sigmoidal (# 3). In the Table VII for the sigmoid kernel function the first number is k_1, the second number is k_2.

At the first level of the two-level SVM classifier 215 objects were selected from the initial 400 objects. These 215 objects have been identified by the group of the SVM classifiers as the support vectors. Noteworthy, 204 objects were classified correctly and entered in the training set SV^+, and 11 objects were incorrectly classified and entered in the test set SV^-. The time used for the development of one individual SVM classifier is on average less than 1 second.

At the second level of the two-level SVM classifier the SVM classifier on the base of the modified PSO algorithm has been created. A herewith we used the training set SV^+ and the test set SV^-. The search time for optimal parameters

amounted to 2465 seconds, that almost 3 times less than the search time for the original experimental data set (7146 seconds).

The remaining 185 objects (more than 46%) were not used in the development of the SVM classifier and compiled the classifying data set. These objects were correctly classified by the two-level SVM classifier.

Fig. 8 shows the classification results of the Test2 data set: on the left – the part of the objects (support vectors) and their separating curve; on the right – the original experimental data set (after the classification of the remaining 185 objects).

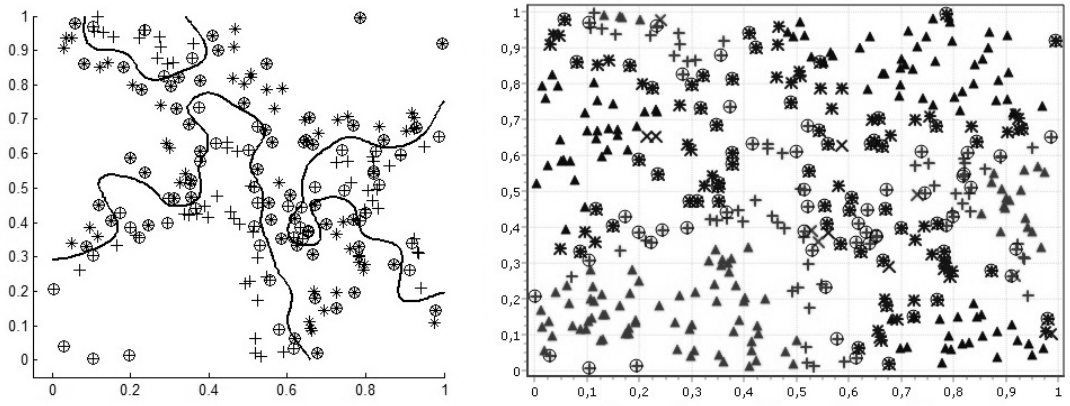


Fig. 8. The classification results of the Test2 data set

TABLE VIII. THE PARAMETERS AND CHARACTERISTICS OF THE INDIVIDUAL CLASSIFIERS (SPAM DATA SET)

Number of the individual SVM classifier	1	2	3	4	5	6	7	8	9	10	**Result**
Kernel function type	1	1	1	2	2	2	2	3	3	3	**1**
Regularization parameter	1	2	2.5	10	12	10	10	0.5	0.5	3.4	**0.1**
Kernel parameters	3	4	3	15	10	15	8.00	0.15; -1.00	0.50; -1.50	0.30; -0.80	**3**
Overall accuracy (%)	89.15	86.66	84.7	94.09	94.39	93.96	91.28	86.09	84.24	81.85	**97.26**
Number of the support vectors	579	561	576	752	760	859	723	515	584	638	**717**
Size of the training set	3681	3911	3451	3221	3681	3681	3681	3451	3911	3451	**1834**
Quantity of errors at the training	364	497	505	178	189	214	315	484	624	625	**27**
Size of the test set	920	690	1150	1380	920	920	920	1150	690	1150	**221**
Quantity of errors at the testing	135	117	199	94	69	64	86	156	101	210	**26**
Size of the classified set	-	-	-	-	-	-	-	-	-	-	**2546**
Quantity of errors at the classification	-	-	-	-	-	-	-	-	-	-	**73**
Sensitivity of the classifier (%)	73.97	69.11	62.82	92.11	89.74	91.89	80.14	76.78	77.88	81.96	**95.37**
Specificity of the classifier (%)	99.03	98.06	98.92	95.37	97.42	95.3	98.53	92.14	88.38	81.78	**98.49**
Number of errors of the 1-st type	27	54	30	129	72	131	41	219	324	508	**42**
Number of errors of the 2-nd type	472	560	674	143	186	147	360	421	401	327	**84**
Development time, s	4	8	7	4	3	3	3	2	2	3	**19599**

During the experiments it was found that the individual classifiers show the accuracy of ranging from 85.75% to 91.5%. The accuracy of the two-level SVM classifier amounted to 96.75%. Thus, using the two-level SVM classifier has improved the classification accuracy by more than 5% compared to the maximum precision of one of the SVM classifiers. The number of objects used in the training and testing of the SVM classifier was reduced from 400 to 215.

Besides, the offered two-level SVM classifier has been used for Spam data set classification. For this data set the group of 10 SVM classifiers was trained (Table VIII). Three kernel functions were included in the search: polynomial (# 1), radial basis (# 2) and sigmoidal (# 3). In the Table VIII for the sigmoid kernel function the first number is k_1, the second number is k_2. A herewith we used the SVM classifiers which show the acceptable classification accuracy (more than 80%) under the small number of the support vectors (till 1000).

At the first level of the two-level SVM classifier 2055 objects (that is equal to about 45% of the original experimental data set) were selected from the initial 4601 objects. These 2055 objects have been identified by the group of the SVM classifiers as the support vectors. Noteworthy, 1834 objects were classified correctly and entered in the training set SV^+, and 221 objects were incorrectly classified and entered in the test set SV^-. The time used for the development of one individual SVM classifier is on average less than 4 second.

At the second level of the two-level SVM classifier we found the best SVM classifier on the base of the modified PSO algorithm with the polynomial kernel function. A herewith $d = 3$ and $C = 0.1$. The search time for optimal parameters amounted to 19566 seconds, that almost 2 times less than the search time for the original experimental data set (44933 seconds).

During the experiments it was found that the individual classifiers show the accuracy of ranging from 85.75% to

91.5%. The accuracy of the two-level SVM classifier amounted to 97.26%. Thus, the two-level SVM classifier improved the classification accuracy by almost 3% compared to the maximum accuracy of one of the SVM classifiers. The number of objects used in the training and testing of the SVM classifier was reduced from 4601 to 2055 (i.e. more than twice).

Thus, the results of experimental studies confirm the efficiency of the offered approaches for Big Data classification.

VII. Conclusion

The efficiency of the suggested approaches has been confirmed by the results of experimental studies.

The SVM classifiers on the base of the modified PSO algorithm allow classifying data with the high classification accuracy.

The modified PSO algorithm allows choosing the best kernel function type, values of the kernel function parameters and value of the regularization parameter within appropriate time expenditures, which turned out to be significantly less than when using the traditional PSO algorithm. The main feature of the modified PSO algorithm is using the «regeneration» of the particles.

The SVM ensembles based on the decorrelation maximization algorithm for the different strategies of the decision-making on the data classification and the majority vote rule allow reducing the accident classification decision received by one classifier, and help to improve the classification accuracy. The shortcomings of some classifiers are compensated by strengths of others classifiers thanks to combination of their results. Classifiers counterbalance the results' accident of each other, finding the most plausible output classification decision. It allows finding the best classification result with minimum classification error.

The two-level SVM classifier also allows improving the classification accuracy within appropriate time expenditures.

Further researches will be devoted to the development of recommendations on the application of the SVM classifiers based on the modified PSO algorithm and their ensembles for the solution of the practical problems, especially for the Big Data classification problems. It is necessary to say that the PSO algorithm and other nature inspired swarm optimization algorithms are very well suited for the distributed architecture and handling of high volume unstructured data in the Big Data analytics.

References

[1] O. Chapelle, V. Vapnik, O. Bousquet, and S. Mukherjee, "Choosing Multiple Parameters for Support Vector Machine," Machine Learning, vol. 46 (1–3), pp. 131–159, 2002. http://dx.doi.org/10.1023/A:1012450327387

[2] V. Vapnik, "Statistical Learning Theory," Wiley, New York, 1998.

[3] Lean Yu, Shouyang Wang, Kin Keung Lai, and Ligang Zhou, "Bio-Inspired Credit Risk Analysis. Computational Intelligence with Support Vector Machines," Springer-Verlag Berlin Heidelberg, p. 244, 2008.

[4] J.S. Raikwal, and K. Saxena, "Performance Evaluation of SVM and K-Nearest Neighbor Algorithm over Medical Data set," International Journal of Computer Applications, vol. 50, no. 14, pp. 35–39, 2012. http://research.ijcaonline.org/volume50/number14/pxc3881055.pdf

[5] Y. LeCun, L.D. Jackel, L. Bottou, C. Cortes, at al., "Learning Algorithms for Classification: A Comparison on Handwritten Digit Recognition," Neural Networks: The Statistical Mechanics Perspective, Oh, J. H., Kwon, C. and Cho, S. (Ed.), World Scientific, pp. 261–276, 1995.

[6] T. Joachims, "Text Categorization with Support Vector Machines: Learning with Many Relevant Features," Lecture Notes in Computer Science, vol. 1398, pp. 137–142, 2005. http://dx.doi.org/10.1007/BFb0026683

[7] Y. Li, K. Bontcheva, and H. Cunningham, "SVM Based Learning System For Information Extraction," Lecture Notes in Computer Science, vol. 3635, pp. 319–339, 2005. http://dx.doi.org/10.1007/11559887_19

[8] M. Oren, C. Papageorgious, P. Sinha, E. Osuna, and T. Poggio, "Pedestrian Detection Using Wavelet Templates," 1997 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 193–199, 1997. http://dx.doi.org/10.1109/CVPR.1997.609319

[9] E. Osuna, R. Freund, and F. Girosi, "Training Support Vector Machines: An Application to Face Detection," 1977 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 130–136, 1997. http://dx.doi.org/10.1109/CVPR.1997.609310

[10] I. Saha, U. Maulik, S. Bandyopadhyay, and D. Plewczynski. SVMeFC: SVM Ensemble Fuzzy Clustering for Satellite Image Segmentation // IEEE Geoscience and Remote Sensing Letters, 2012, vol. 9, no. 1, pp. 52–55.

[11] S.K. Shevade, S.S. Keerthi, C. Bhattacharyya, and K.R.K. Murthy, "Improvements to the SMO Algorithm for SVM Regression," IEEE Transactions on Neural Networks, vol. 11, no. 5, pp. 1188–1193, 2000. http://dx.doi.org/10.1109/72.870050

[12] E. Osuna, R. Freund, and F. Girosi, "Improved Training Algorithm for Support Vector Machines," 1997 IEEE Workshop Neural Networks for Signal Processing, pp. 24–26, 1997. http://dx.doi.org/10.1109/NNSP.1997.622408

[13] S.V.N. Vishwanathan, A. Smola, and N. Murty, "SSVM: a simple SVM algorithm," Proceedings of the 2002 International Joint Conference on Neural Networks, vol. 3, pp. 2393-2398, 2002. http://dx.doi.org/10.1109/IJCNN.2002.1007516

[14] S. Shalev-Shwartz, Y. Singer, N. Srebro, and A. Cotter, "Pegasos: Primal Estimated sub-Gradient Solver for SVM," Mathematical Programming, vol. 127, no. 1, pp. 3–30, 2011. http://dx.doi.org/10.1007/s10107-010-0420-4

[15] L. Bottou, and C.-J. Lin, "Support Vector Machine Solvers," MIT Press, pp. 1–28, 2007.

[16] H.P. Graf, E. Cosatto, L. Bottou, I. Durdanovic, and V. Vapnik, "Parallel Support Vector Machines: The Cascade SVM," Advances in Neural Information Processing Systems, 17, 521–528, 2005.

[17] O. Meyer, B. Bischl, and C. Weihs, "Support Vector Machines on Large Data Sets: Simple Parallel Approaches," In M. Spiliopoulou, L. Schmidt-Thieme, and R. Janning, editors, Data Analysis, Machine Learning and Knowledge Discovery, Studies in Classification, Data Analysis, and Knowledge Organization, pp. 87–95, 2014.

[18] A. Priyadarshini, and S. Agarwal, "A Map Reduce based Support Vector Machine for Big Data Classification," International Journal of Database Theory and Application, vol.8, no.5 (2015), pp. 77–98. doi: 10.14257/ijdta.2015.8.5.07

[19] G. Cavallaro, M. Riedel, M. Richerzhagen, J. A. Benediktsson, and Antonio Plaza, "On Understanding Big Data Impacts in Remotely Sensed Image Classification Using Support Vector Machine Methods," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 8 , issue: 10, pp. 4634–4646, 2015. doi: 10.1109/JSTARS.2015.2458855

[20] P. Yasodha, and N.R. Anathanarayanan, "Analysing Big Data to Build Knowledge Based System for Early Detection of Ovarian Cancer," Indian Journal of Science and Technology, vol 8(14), 2015. doi: 10.17485/ijst/2015/v8i14/65745

[21] P. Rebentrost, M. Masoud, and L. Seth, "Quantum Support Vector Machine for Big Data Classification," Phys. Rev. Lett. 113, 130503, 2014. doi: 10.1103/PhysRevLett.113.130503

[22] D.E. Goldberg, B. Korb, and K. Deb, "Messy genetic algorithms: Motivation, analysis, and first results," Complex Systems, vol. 3, no. 5, pp. 493–530, 1989. http://www.complex-systems.com/abstracts/v03_i05_a05.html

[23] D.R. Eads, D. Hill, S. Davis, S.J. Perkins, J. Ma, R.B. Porter, and J.P Theiler, "Genetic algorithms and support vector machines for time series classification," Proc. SPIE 4787, Applications and Science of Neural Networks, Fuzzy Systems, and Evolutionary Computation, p. 74, 2002. http://dx.doi.org/10.1117/12.453526

[24] S. Lessmann, R. Stahlbock, and S.F. Crone, "Genetic algorithms for support vector machine model selection," 2006 IJCNN'06. International Joint Conference on Neural Networks, pp. 3063–3069, 2006. http://dx.doi.org/10.1109/IJCNN.2006.247266

[25] D. Karaboga, and B. Basturk, "Artificial Bee Colony (ABC) Optimization Algorithm for Solving Constrained Optimization Problems," Proceeding IFSA '07 Proceedings of the 12th international Fuzzy Systems Association world congress on Foundations of Fuzzy Logic and Soft Computing, pp. 789–798, 2007.

[26] Jun Sun, Choi-Hong Lai, and Xiao-Jun Wu, "Particle Swarm Optimisation: Classical and Quantum Perspectives," CRC Press, p. 419, 2011.

[27] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization," Swarm Intelligence, vol. 1, no. 1, pp. 33–57, 2007. http://dx.doi.org/10.1007/s11721-007-0002-0

[28] W. Xun, Y.B. An, and R. Jie, "Application of Parallel Particle Swarm Optimize Support Vector Machine Model Based on Hadoop Framework in the Analysis of Railway Passenger Flow Data In China," Chemical Engineering Transactions, vol. 46, pp. 367–372, 2015. doi: 10.3303/CET1546062

[29] P.S. Duggal, S. Paul, and P. Tiwari, "Analytics for the Quality of Fertility Data using Particle Swarm Optimization," International Journal of Bio-Science and Bio-Technology, vol. 7, no.1, pp. 39-50, 2015. doi: 10.14257/ijbsbt.2015.7.1.05

[30] L. Demidova, and Yu. Sokolova, "Modification Of Particle Swarm Algorithm For The Problem Of The SVM Classifier Development," 2015 International Conference "Stability and Control Processes" in Memory of V.I. Zubov (SCP). pp. 623–627, 2015.

[31] L. Demidova, E. Nikulchev, and Yu. Sokolova, "The SVM Classifier Based on the Modified Particle Swarm Optimization," International Journal of Advanced Computer Science and Applications, vol. 7, no. 2, pp. 16-24, 2016.

[32] A. Karatzoglou, D. Meyer, and K. Hornik, "Support vector machines in R," Research Report, WU Vienna University of Economics and Business, Vienna, 2005. http://epub.wu.ac.at/id/eprint/1500

[33] L. Demidova, Yu. Sokolova and E. Nikulchev, "Use of Fuzzy Clustering Algorithms' Ensemble for SVM classifier Development," International Review on Modelling and Simulations (IREMOS), vol. 8, no. 4, pp. 446–457, 2015. http://dx.doi.org/10.15866/iremos.v8i4.6825

[34] L. Demidova, and Yu. Sokolova, "SVM-Classifier Development With Use Of Fuzzy Clustering Algorithms' Ensemble On The Base Of Clusters' Tags' Vectors' Similarity Matrixes," 16th International Symposium on Advanced Intelligent Systems, pp. 889–906, 2015.

[35] L. Demidova, and Yu. Sokolova, "Development of the SVM Classifier Ensemble for the Classification Accuracy Increase," 6-th Seminar on Industrial Control Systems: Analisys, Modeling and Computing, 2016

[36] L. Demidova, and Yu. Sokolova, "Training Set Forming For SVM Algorithm With Use Of The Fuzzy Clustering Algorithms Ensemble On Base Of Cluster Tags Vectors Similarity Matrices," 2015 International Conference "Stability and Control Processes" in Memory of V.I. Zubov (SCP), pp. 619–622, 2015.

[37] M.S. Eastaff., and P. Premalatha, "Analysis of Big Data Based On Ensemble Classification," Proceedings of the UGC Sponsored National Conference on Advanced Networking and Applications (27th March 2015), pp. 191–193, 2015.