TERMINATION OF SIMULATED ANNEALING ALGORITHM SOLVING SEMI-SUPERVISED LINEAR SVMS PROBLEMS

Vaida Bartkute-Norkuniene
Vilnius University, Institute of Informatics and Mathematics, Vilnius, Lithuania
Utena University of Applied Sciences, Utena, Lithuania

Keywords: Order Statistics, Continuous Optimization, Simulated Annealing, Semi-supervised SVMs Classification.

Abstract: In creating heuristic search algorithms one has to deal with the practical problem of terminating and optimality testing. To solve these problems, we can use information gained from the set of the best function values (order statistics) provided during optimization. In this paper, we consider the application of order statistics to establish the optimality in heuristic optimization algorithms and to stop the Simulated Annealing algorithm when the confidence interval of the minimum becomes less than admissible value. The accuracy of the solution achieved during optimization and the termination criterion of the algorithm are introduced in a statistical way. We build a method for the estimation of confidence intervals of the minimum using order statistics, which is implemented for optimality testing and terminating in Simulated Annealing algorithm. A termination criterion - length of the confidence interval of the extreme value of the objective function - is introduced. The efficiency of this approach is discussed using the results of computer modelling. One test function and two semi-supervised SVMs linear classification problems illustrate the applicability of the method proposed.

1 INTRODUCTION

The termination problem is topical in stochastic and heuristic optimization algorithms. Note, values of the objective function provided during optimization contain important information on the optimum of the function and, thus, might be applied to algorithm termination. Mockus (Mockus, 1967), Zilinskas and Zhigljavsky (Zilinskas & Zhigljavsky, 1991) were the first who proposed statistical inferences for optimality testing in optimization algorithms using theory of order statistics. These inferences were studied by computer simulation (see, Bartkute et al, 2005, Bartkute & Sakalauskas, 2009), which confirmed theoretical assumption about the distribution of order statistics with respect to extreme value distribution. Thus, the estimate of extremum value of the objective function and its confidence interval were proposed following to latter assumption. Besides, in Bartkute & Sakalauskas (Bartkute & Sakalauskas, 2009a) it was proposed to terminate the stochastic optimization algorithm, when the confidence interval of the extremum becomes less than prescribed value. Since theoretical analysis of the optimal decision making algorithm is complicated, computer modelling becomes an important research method that enable us to test and study hypotheses arising from the problem discussed above. Semi-supervised SVMs linear classification problems as an examples illustrate the applicability of the method proposed.

2 METHOD FOR TESTING THE OPTIMALITY

Assume, the optimization problem is (minimization)

$$f(x)\to\min \tag{1}$$

where $f:\Re^n\to\Re$ is a function bounded from below, $\min\limits_{x\in\Re^n} f(x)=f\left(x^*\right)=A>-\infty$, $\left|x^*\right|<\infty$. Let this problem be solved by the Markov type algorithm providing a sample $\mathrm{H}=\{\eta_1,\dots,\eta_\mathrm{N}\}$, whose elements are function values $\eta_k=f(x_k)$. Our approach is grounded by the assumption on the asymptotic distribution of order statistics according to the Weibull (Weibull, 1951) distribution

Bartkute-Norkuniene V..
TERMINATION OF SIMULATED ANNEALING ALGORITHM SOLVING SEMI-SUPERVISED LINEAR SVMS PROBLEMS.
DOI: 10.5220/0003759301500156
In *Proceedings of the 1st International Conference on Operations Research and Enterprise Systems* (ICORES-2012), pages 150-156
ISBN: 978-989-8425-97-3

$W(x,\alpha,c,A)=1-e^{-c\cdot(x-A)^{\alpha}}$, $\alpha>0,\ x\geq A,\ c>0,$ where c, A and α denote the scale, location and shape parameters, respectively (see for details Bartkute & Sakalauskas, 2009a). The Weibull distribution is one of the extreme-value distributions which is applied also in optimality testing of Markov type optimization algorithms. Although this limit distribution of extreme values is studied mostly for i.i.d. values, it also might be often used in the absence of the assumption of independence (Galambosh, 1984).

To estimate confidence intervals for the minimum A of the objective function, it suffices to choose from sample H only $k+1$ the best function values $\eta_{0,N},\dots,\eta_{k,N}$, from the ordered sample $\eta_{0,N}\leq\eta_{1,N}\leq,\dots,\leq\eta_{k,N}\leq\dots\leq\eta_{N,N}$, where $k=k(N)$, $\frac{k^2}{N}\to 0,\ N\to+\infty$ (Zilinskas & Zhigljavsky, 1991, Bartkute & Sakalauskas, 2009a). Then the linear estimators for A can be as follows:

$$A_{N,k}=\eta_{0,N}-c_k\cdot\left(\eta_{k,N}-\eta(0)\right) \qquad (2)$$

where coefficient c_k can be estimated as $c_k=1\Big/\left(\prod_{i=1}^{k}\left(1+\frac{1}{i\cdot\alpha}\right)-1\right)$, α is the shape parameter of distribution of the extreme values, $\alpha=\frac{n}{\beta}$, β is the parameter of homogeneity of the function $f(x)$ in the neighbourhood of the point of minimum: $\left|f(x)-f\left(x^*\right)\right|=O\left(\left\|x-x^*\right\|^{\beta}\right)$ (Zilinskas & Zhigljavsky, 1991, Bartkute & Sakalauskas, 2009a).

The one-side confidence interval of the minimum of the objective function is as follows:

$$[\eta_{0,N}-r_{k,\gamma}\cdot\left(\eta_{k,N}-\eta_{0,N}\right),\eta_{0,N}] \qquad (3)$$

where $r_{k,\gamma}=\left(1-(1-\delta)^{\frac{1}{k}}\right)^{\frac{1}{\alpha}}\Big/\left(1-\left(1-(1-\delta)^{\frac{1}{k}}\right)^{\frac{1}{\alpha}}\right)$, γ is the confidence level.

The estimates introduced here might be used to create the termination criterion for the stochastic and heuristic optimization algorithms, namely, the algorithm stops, when the length of the confidence interval becomes less than prescribed value $\varepsilon>0$.

3 DESCRIPTION OF SIMULATED ANNEALING ALGORITHM

Let us consider an application of this approach to continuous global optimization by the Simulated Annealing algorithm (SA). This is a well-known Markov type algorithm for random optimization. Simulated Annealing (SA) is widely applied in multiextremal problems. Conditions of global convergence of SA are studied by many authors (Granville *et al*., 1994, Yang, 2000, etc.). We use the modification of SA, developed by Yang (2000), where the function regulating the neighbourhood depth of solution is introduced together with the temperature regulation function. The procedure of the SA algorithm consists of the following steps:

Step 1. Choose an initial point $x^0\in D\subset\Re^n$, an initial temperature value $T_0>0$, a kind of temperature-dependent generation probability density function, a corresponding temperature updating function, and a sequence $\{\rho_t,\ t\geq 0\}$ of monotonically decreasing positive numbers, describing the neighboring states. Calculate $f(x^0)$. Set $t=0$.

Step 2. Generate a random vector z^t by using the generation probability density function. If there exists i such that $\left\|z_i^t\right\|<\rho_t$, $1\leq i\leq n$, where z_i^t is the i^{th} component of the vector z^t, repeat Step 2. Otherwise, generate a new trial point y^t by adding the random vector z^t to the current iteration point x^t,

$$y^t=x^t+z^t \qquad (4)$$

If $y^t\notin D$, repeat Step 2; otherwise, calculate $f(y^t)$.

Step 3. Use the Metropolis acceptance criterion to determine a new iteration point x^{t+1} [10]. Specifically, generate a random number κ with the uniform distribution over [0,1], and then calculate the probability $P\left(y^t,x^t,T_t\right)$ of accepting the trial point y^t as the new itteration point x^{t+1}, given x^t and T_t, $P(y^t,x^t,T_t)=\min\left\{1,\ \exp\left(\frac{f(x^t)-f(y^t)}{T_t}\right)\right\}$.

If $\kappa \le P\left(y^t, x^t, T_t\right)$, set $x^{t+1} = y^t$ and $f\left(x^{t+1}\right) = f\left(y^t\right)$; otherwise, set $x^{t+1} = x^t$ and $f\left(x^{t+1}\right) = f\left(x^t\right)$.

Step 4. If the prescribed termination condition is satisfied, then stop; otherwise, update the value of the temperature by means of the temperature updating function, and then go back to Step 2.

Thus, by applying the generation mechanism and the Metropolis acceptance criterion, the SA algorithm produces two sequences of random points. These are the sequence $\left\{y^t, t \ge 0\right\}$ of trial points generated by (4) and the sequence $\left\{x^t, t \ge 0\right\}$ of iteration points determined by applying the Metropolis acceptance criterion as described in Step 3. These two sequences of random variables are all dependent on the temperature sequence $\{T_t, t \ge 0\}$ determined by the temperature updating function, the state neighbouring sequence $\{\rho_t, t \ge 0\}$, and the approach of random vector generation.

The sequence $\{\rho_t, t \ge 0\}$ of positive numbers specified in Step 1 of the above SA algorithm is used to impose a lower bound on the random vector, generated at the each iteration, for obtaining the random trial point. This lower bound should be small enough and monotonically decreasing as the annealing proceeds. Since the temperature-dependent generation probability density function is used to generate random trial points and since only one trial point is generated at each temperature value the SA algorithm considered is characterized by a nonhomogeneous continuous-state Markov chain.

The convergence conditions of the SA were studied by Yang (Yang, 2000) and several updating functions for the method parameters were given, which ensure convergence of the method. We applied the next updating functions in testing our approach.

Let $r \in \Re^n$, with component $r_i = \max_{x,y \in D} |x_i - y_i|$, $1 \le i \le n$, $d > 1$, $u > 1$, $0 < \lambda < u$, $0 < \rho_0 < \min_{1 \le i \le n} r_i$, $\rho_t = \rho_0 \cdot t^{-\frac{\lambda}{u \cdot n}}$ for all $t \ge 1$, where $\{\rho_t, t \ge 0\}$ is the sequence used to impose lower bounds on the random vectors generated in the SA algorithm. Let the temperature-dependent generation probability density function $p(\cdot, T_t)$ be given by

$$p(z, T_t) = \prod_{i=1}^{n} \frac{(a-1)}{2T_t} \cdot \left(\frac{|z_i|}{T_t} + 1\right) \left(\log\left(\frac{|z_i|}{T_t} + 1\right)\right)^d, \; z \in \Re^n.$$

Then, for any initial point $x^0 \in D$, the sequence $\left\{f(x^t); t \ge 0\right\}$ of objective function values converges in probability to the global minimum f^*, if the temperature sequence $\{T_t, t \ge 0\}$ determined by the temperature updating function satisfies the following condition:

$$T_t = T_0 \cdot \exp\left(-l \cdot t^{\frac{1}{d \cdot n}}\right), \; i = 1, 2, ...,$$

where $T_0 > 0$ is the initial temperature value and $l > 0$ is a given real number (Yang, 2000). Typically a different form of the temperature updating function has to be used with respect to a different kind of the generation probability density function in order to ensure the global convergence of the corresponding SA algorithm. Furthermore, the flatter is the tail of the generation probability function, the faster is the decrement of the temperature sequence determined by the temperature updating function.

4 SVM CLASSIFICATION

Data classification is a common problem in science and engineering. Support Vector Machines (SVMs) are powerful tools for classifying data that are often used in data mining operations.

In the standard binary classification problem, a set of training data $\left(u^i, y^i\right)$, … , $\left(u^m, y^m\right)$ is observed, where the input set of points is $u^i \in U \subset \Re^n$, the y^i is either +1 or −1, indicating the class to which the point u^i belongs, $y^i \in \{+1, -1\}$. The learning task is to create the classification rule $f : U \to \{+1, -1\}$ that will be used to predict the labels for new inputs. The basic idea of SVMs classification is to find a maximal margin separating hyperplane between two classes. It was first described by Cortes and Vapnik (Cortes & Vapnik, 1995). The standard binary SVM classification problem is shown visually in Figure 1.

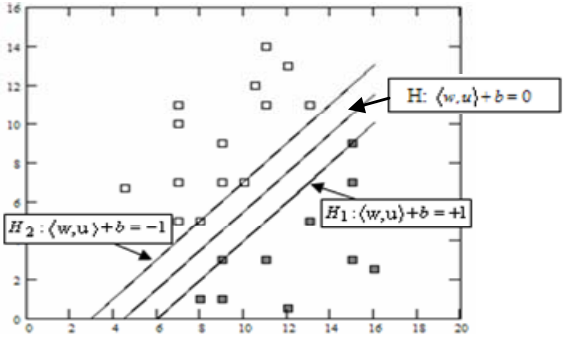

Figure 1: Linear separating hyperplanes for a separable case.

4.1 Semi-supervised Linear SVMs

There are a lot of classification problems where data labeling is hard or expensive, while unlabeled data is often abundant and cheap to collect. The typical areas where this happens is the speech processing, text categorization, webpage classification, business risk identification, credit scoring and, finally, a bioinformatics area where it is usually both expensive and slow to label huge number of data produced. When data points consist of exactly two sets: one set that has been labeled by a decision maker and the other that is not classified, but belongs to one known category we have a traditional semi-supervised classification problem (Bennett & Demiriz (1999), Huang & Kecman (2004)). The goal of semi-supervised classification is to use unlabeled data to improve the performance of standard supervised learning algorithms. In semi-supervised learning the data set $U = \{u^i\}_{i=1}^{n}$ can be divided into two parts: the training set consists of p labelled examples $\{(u^i, y^i)\}_{i=1}^{p}$, $y^i = \pm 1$, and of m unlabeled examples $\{u^i\}_{i=p+1}^{n}$, with $n = p + m$. The learning task is to create the classification rule $f : U \to \{+1, -1\}$ that will be used to predict the labels for new inputs. To solve that problem we may rewrite standard binary classification problem (Cortes & Vapnik, 1995) in the following unconstrained form (Astorino & Fuduli, 2007, Bartkute-Norkuniene, 2009b):

$$\min_{w \in \Re^n, b \in \Re} f(w,b),$$

where

$$f(w,b) = \frac{\|w\|^2}{2} + C_1 \cdot \sum_{i=1}^{p} L\left(y^i \cdot \left(w^T \cdot u^i + b\right)\right) + C_2 \cdot \sum_{i=p+1}^{m+p} L\left(\left|w^T \cdot u^i + b\right|\right)$$

w and b are both the hyperplane parameters, $L(t) = max(0, 1-t)$, $L(|t|) = max(0, 1-|t|)$ are the loss functions, $C_1 \geq C_2 \geq 0$ are certain penalty coefficients, p is the size of training set, and m is the size of testing set. The first two terms in the objective function $f(w,b)$ define the standard SVM, and the third one incorporates unlabelled (testing) data. The error over labelled and unlabelled examples is weighted by two parameters C_1 and C_2. This form seems advantageous especially when the input dataset is very large.

5 COMPUTER MODELLING

The empirical evidence of our approach, using two test functions, synthetic and real datasets, is provided and discussed in this Section. To evaluate the performance of our proposed algorithm in practice, we analyze two machine learning datasets.

Example 1: test function (Zhigljavsky & Zilinskas, 2007)

$$f_{(s,l)}(x) = \begin{cases} 1 - \frac{1}{2}\left(\sin\frac{ls\pi x}{(s-1)(l-1)}\right)^2 & for\ x \in \left[0, \frac{(s-1)(l-1)}{sl}\right] \\ 1 - \left(\sin\frac{ls\pi x}{l-1}\right)^2 & for\ x \in \left[\frac{(s-1)(l-1)}{sl}, \frac{l-1}{l}\right] \\ 1 - \frac{1}{2}(\sin l\pi x)^2 & for\ x \in \left[\frac{l-1}{l}, 1\right] \end{cases}$$

For all integer $s, l \geq 2$, the functions $f_{(s,l)}(x)$ are continuously differentiable in the set $[0,1]$ and have three local minima. These local minima are achieved at the points:

$$x_1 = \frac{(s-1)(l-1)}{2sl},\ x_2 = \frac{(2s-1)(l-1)}{2sl},\ x_3 = 1 - \frac{1}{2l}.$$

Global minimum is at the point x_2 and equal to 0. Despite the fact that the functions $f_{(s,l)}(x)$ are continuously differentiable, the problem of finding the minimum point is very difficult when k is large.

Example 2: The Rastrigin function

$f(x) = 10n + \sum_{i=1}^{n}\left(x_i^2 - 10\cos(2\pi \cdot x_i)\right)$, search domain is $-5.12 \leq x_i \leq 5.12$, $n = 2$, the minimum is 0.

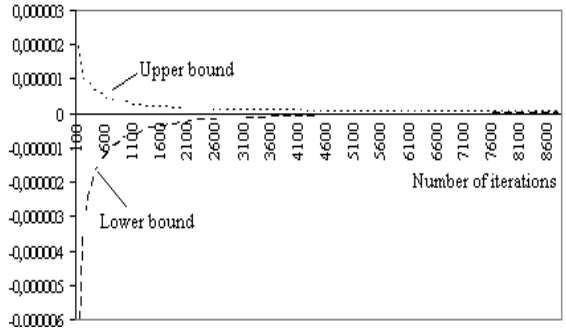

Figure 2: Confidence bounds of the minimum (Example 1, s=12, l=5).

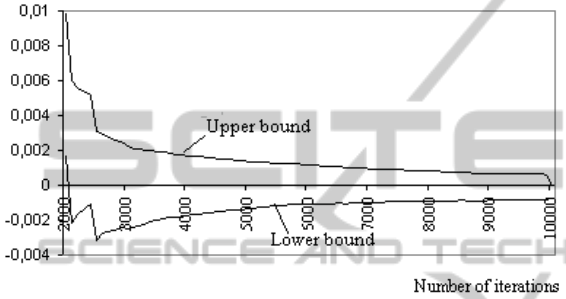

Figure 3: Confidence bounds of the minimum (Example 2).

Test functions were minimized, with the number of iterations N =10000 and the number of trials M=500, starting from points randomly distributed in the search domain. Results of the estimate (2) of the test functions minimum value $A_{N,k}$ and the estimate (3) of the confidence interval are presented in Table 1 and Figures 2 and 3. These results show that the proposed estimates approximate the confidence interval of the objective function minimum value rather well, and that the length of the confidence interval decreases when the number of iterations increases.

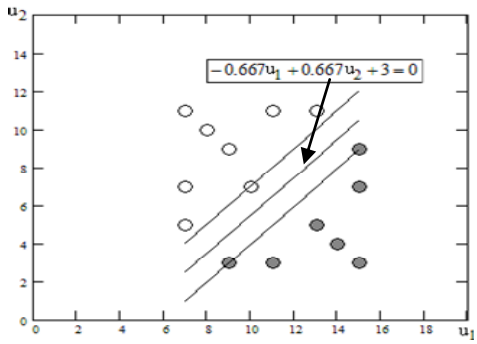

Figure 4: Linear separating hyperplanes of training data.

Example 3: linear example (V. Bartkute-Norkuniene (2009). The linear separating hyperplanes of training data are demonstrated in Figure 4. Figure 5 illustrates that the SA classifier for training and testing datasets is close to an optimal decision boundary.

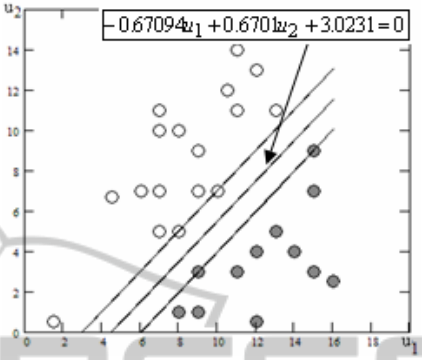

Figure 5: Linear separating hyperplanes of the training and testing data.

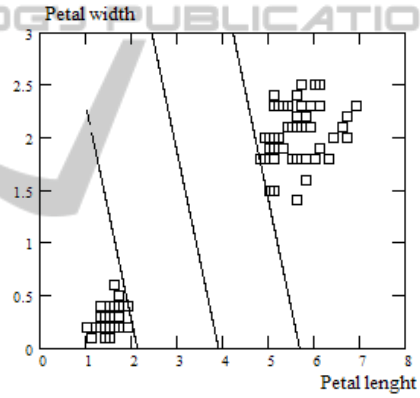

Figure 6: Linear separating hyperplanes for two dimensional Iris Plant data, b= 2.1830, w_1=-0.5625, w_2= -0.2741.

Example 4: dataset of Iris Plants (Asuncion & Newman, 2007). The dataset contains 3 classes of 50 instances each, where each class refers to a type of iris plant. One class is linearly separable from the other two, the latter are not linearly separable from each other. In our approach for the binary classification we use only two classes of iris plant: iris Setosa (the class +1) and iris Virginica (the class -1).

Linear separating hyperplanes for two-dimensional Iris Plant data are illustrated in Figure 6. These results illustrate the applicability of SA algorithm for Semi-supervised SVM classification.

In Figure 7, we can see histograms of the number of iterations after termination of the SA algorithm depending on the length of the confidence interval.

Table 1: Computer modelling results of the minimum value and the confidence interval.

Confidence probability	$A_{N,k}$	Confidence interval		p	Confidence interval of the hitting probability p	
		Lower bound	Upper bound		Lower bound	Upper bound
Example 1						
$\delta = 0.9$	-0.0000000307	-0.000000483	0.0000002275	0.91	0.8614377	0.94498488
$\delta = 0.95$	0.0000000005	-0.00000002	0.0000000072	0.95	0.89763031	0.98009752
$\delta = 0.975$	-0.000000031	-0.00000151	0.0000002275	0.98	0.92955759	0.9975685
$\delta = 0.99$	-0.000000031	-0.00000239	0.0000002275	0.98	0.91852038	0.99850762
Example 2						
$\delta = 0.9$	0.0000478328	-0.00077633	0.000620020	0.886	0.90384692	0.86549069
$\delta = 0.95$	0.0000478328	-0.00122913	0.000620020	0.948	0.92806921	0.96283961
$\delta = 0.975$	0.0000478328	-0.00169791	0.000620020	0.97	0.95099096	0.98311659
$\delta = 0.99$	0.0000478328	-0.00234306	0.000620020	0.984	0.96551508	0.99416328

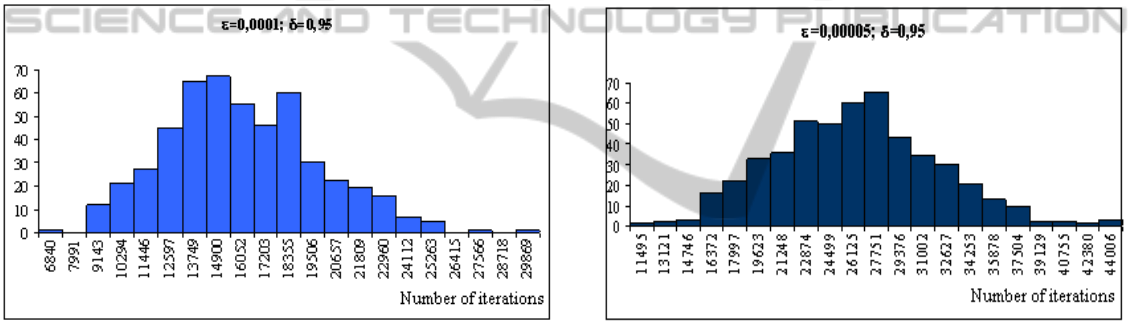

Figure 7: The number of iterations after termination of the algorithm (two dimensional Iris Plant data).

6 CONCLUSIONS

A linear estimator and confidence bounds for the minimum value of the function have been proposed, using order statistics of the function values provided by SA algorithm, which were studied in an experimental way. These estimators are simple and depend only on the parameter of the extreme value distribution α. The latter parameter α is easily estimated, using the parameter of homogeneity of the objective function or in a statistical way. Theoretical considerations and computer examples have shown that the confidence interval of the function minimum can be estimated with an admissible accuracy, when the number of iterations is increased. Empirical study of the statistical hypothesis on order statistics have shown that function values lead us to a conclusion that the estimates proposed can be applied in optimality testing and termination of the SA algorithm. The estimates introduced here can be used to create the termination criterion for SA algorithm, namely, the algorithm stops, when the length of the confidence interval becomes smaller than prescribed value $\varepsilon > 0$.

REFERENCES

Astorino, A., Fuduli, A., 2007. Nonsmooth Optimization Techniques for Semisupervised Classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, No. 12, p. 2135-2142.

Asuncion, A., Newman, D. J., 2007. *UCI Machine Learning Repository. School of Information and Computer Science*, University of California, Irvine, CA. (http://www.ics.uci.edu/˜mlearn/ MLRepository. html)

Bartkutė, V., Sakalauskas, L., 2004. Order statistics for testing optimality in stochastic optimization. *Proceedings of the 7th International Conference Computer data analysis and Modelling"*, Minsk, p. 128-131

Bartkute, V., Sakalauskas, L., 2009a. Statistical Inferences for Termination of Markov Type Random Search Algorithms. *Journal of Optimization Theory and Applications*, vol. 141, p. 475-493.

Bartkutė-Norkuniene V., 2009b. Stochastic Optimization Algorithms for Support Vector Machines Classification. *Informatica*, vol. 20, No. 2, p. 173–186.

Bennett, K. P., Demiriz, A., 1999. Semi-supervised support vector machines. In M. S. Kearns, S. A. Solla, and D. A. Cohn, editors, *NIPS*, vol. 11, p. 368–374.

Cortes, C., Vapnik, V., 1995. Support-vector networks. *Machine Learning*, vol. 20, No. 3, p. 273-297.

Galambosh, Y., 1984. *Asymptotic Theory of Extremal Order Statistics*, Nauka, Moscow (in Russian).

Granville, V., Krivanek, M., Rasson, J. P., 1994. Simulated annealing: a proof of convergence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol 16, No. 6, p. 652–656.

Hall, P., 1982. On estimating the endpoint of a distribution. *Annals of Statistic,* vol. 10, p. 556-568.

Huang, T. M., Kecman, V., 2004. Semi-supervised Learning from Unbalanced Labeled Data – An Improvement, in 'Knowledge Based and Emergent Technologies Relied Intelligent Information and Engineering Systems', Eds. Negoita, M. Gh., at al., *Lecture Notes on Computer Science*, vol. 3215, p. 765-771.

Mockus, J., 1967. *Multi-Extremal Problems in Engineering Design*. Nauka, Moscow, (in Russian).

Yang, R. L., 2000. Convergence of the simulated annealing algorithm for continuous global optimization. *Journal of Optimization Theory and Applications*, vol. 104, No. 3, p. 691–716.

Zilinskas, A., Zhigljavsky, A., 1991. *Methods of the global extreme searching*. Nauka, Moscow, (in Russian).

Zilinskas, A., Zhigljavsky, A., 2007. *Stochastic global optimization*. Springer.