STUDIA INFORMATICA 2014
Volume 35 Number 4 (118)

Katarzyna STĄPOR
Silesian University of Technology, Institute of Informatics

USING TABU SEARCH FOR FEATURE SELECTION IN DISCRIMINANT ANALYSIS

Summary. The well known statistical software packages like STATISTICA [11] continue to use classic variable selection methods in stepwise Discriminant Analysis such as the sequential forward/backward ones. Such stepwise procedures suffer from the nesting effect. Moreover, due to the criterion used for evaluation of variable subsets they are designed for descriptive purposes, not for predictive ones. We propose the new solution to the mentioned problems, the feature selection algorithm based on metaheuristic tabu search. After performing some tests it is found that our tabu search-based algorithm obtains significantly better results than stepwise procedures of statistical package.

Keywords: stepwise discriminant analysis, feature selection, metaheuristic, tabu search

SELEKCJA CECH Z WYKORZYSTANIEM PRZESZUKIWANIA Z TABU W ANALIZIE DYSKRYMINACYJNEJ

Streszczenie. W znanych szeroko pakietach do obliczeń statystycznych (np. STATISTICA [11]) selekcja zmiennych wejściowych w module krokowej Analizy Dyskryminacyjnej wykonywana jest z wykorzystaniem klasycznych metod sekwencyjnych w przód/w tył, których wadą jest efekt zagnieżdżania. Również kryterium ewaluacyjne w tychże metodach jest dostosowane do celów deskryptywnych, a nie predyktywnych. Artykuł proponuje nowe rozwiązania wspomnianych problemów – algorytm selekcji z wykorzystaniem metaheurystyki przeszukiwania z tabu. Wykonane, wstępne testy wykazały znacznie lepszą sprawność klasyfikacji w porównaniu z metodami krokowymi.

Słowa kluczowe: krokowa analiza dyskryminacyjna, selekcja cech, metaheurystyka, przeszukiwanie z tabu

1. Introduction

The aim in *classification problem* is to classify the unkown instances characterized by a set of attributes or variables, i.e. to determine which class (group) those instances belong to. Based on a set of examples whose class is known, a set of rules are designed and generalized to classify the set of instances with the lowest error possible.

There are several methodologies for dealing with this problem, for example: classic discriminant analysis, logistic regression, instance-based learning, neural networks, support vector machines, decision trees, etc. [4].

Classical *linear discriminant analysis* (LDA) [8] is a multivariate technique to classify study instances into groups (predictive discriminant analysis, PDA) and/or describe group differences (descriptive discriminant analysis, DDA). The main advantage of LDA is its strong intuitive appeal to managers. A linear discriminant function is easy to calculate and all the manager has to do is to measure a small number of variables, multiple them with the appropriate discrimination coefficients, add them together and compare to the critical Z-score.

Moreover, the use of linear functions enables better interpretation of the results, for example the significance of each variable in instance classification.

LDA is widely used in many areas such as biomedical studies, banking environment (for credit evaluation), financial management, bankrupcy prediction, marketing, and many others.

LDA computes an optimal transformation (projection) by minimizing the within–class distance and maximizing the between–class distance simultaneously, thus achieving maximum class discrimination. The optimal transformation in LDA can be readily computed by applying an eigendecomposition on the so-called scatter matrices.

One of the main problems in classification task in general (and in discriminant analysis as a special case) is variable or *feature selection problem*, i.e. when there are many variables involved, only those variables that are really required should be selected. That is, the first step is to eliminate the less significant variables from the analysis. Extensive research into variable selection problem has been carried out over the past four decades.

Stepwise procedures are a common analytic procedure used in discriminant analysis to reduce the number of variables. However, the use of stepwise procedures (implemented in many commercial statistical software packages like STATISTICA [11]) entails the number of problems which can lead to misleading and inaccurate results.

This work proposes the new method for variable selection in discriminant analysis which can cope with the problems of stepwise procedures. The proposed method is based on the metaheuristic strategy tabu search. It explicitly uses classification performance as a subset selection criteria and because of its more effective search strategy is better than simply forward/backward sequential search strategies implemented in the well-known, commercial statistical packages like STATISTICA [11] for doing stepwise discriminant analysis.

The use of tabu search for feature selection in classification has already been reported, for example in [13,14], but there are very few key references on the selection of variables for their use in discriminant analysis, for example [10] in which the selected metaheuristics are used to guide the search for the best feature subset in discriminant analysis.

In comparison with [10], in our feature selection method based on tabu search we use different representation of a solution as well as different definition of a neighborhood, the two important concepts of tabu search metaheuristic. We also implemented simple, but effective intensification procedure to improve the search process, which differs our method from the ones described in [10, 13].

The remainder of this paper is organized as follows. A short overview of discriminant analysis and stepwise procedures is presented in the 2-nd section. In section 3 the problem of stepwise discriminant analyzis is modeled as a feature selection problem and a short overview of this problem is presented. Section 4 presents our solution to the problems of stepwise discriminant analysis – the feature selection method based on metaheuristic tabu search. The results of the tests on the proposed method are presented in section 5, and a short conclusions follows in section 6.

2. Overview of discriminant analysis and stepwise procedures

Discriminant Analysis (**DA**) [8] is a multivariate statistical method for separating two or more groups of populations. At the basis of observations (i.e. features, input variables) with known group memberships the so-called *discriminant variables* are constructed, aiming at separating the groups as much as possible. DA is broken into a two-step process:

1) computation of a set of discriminant variables and testing their significance,
2) classification.

Suppose we have given observations of a multivariate random variable $X = (X_1, ..., X_d)^T$ coming from *c* populations $G_1, ..., G_c$. Let π_j be the prior probability that an observation to classify belongs to group G_j for *j = 1, …, c*. The population means are denoted by $\mu_1, ..., \mu_c$ and the population covariance matrices by $\Sigma_1, ..., \Sigma_c$. We define the overall weighted mean by $\overline{\mu} = \sum_j \pi_j \cdot \mu_j$. Then the *covariance matrix B* describing the variation *between the groups* is defined as:

$$B = \sum_{j=1}^{c} \pi_j (\mu_j - \overline{\mu})(\mu_j - \overline{\mu})^T \tag{1}$$

The *within groups covariance matrix W* is given by:

$$W = \sum_{j=1}^{c} \pi_j \Sigma_j \tag{2}$$

We consider the linear combinations $Y = a^T X$ where $a \neq 0$. Assuming that the group covariance matrices are all equal, we can form the ratio:

$$\frac{a^T B a}{a^T W a} \tag{3}$$

which measures the variability between the groups of *Y* values relative to the variability within the groups, and maximizing this expression with respect to *a* corresponds to maximizing the separation of the group centers.

It can be shown [8] that the solution for *a* to maximize the above ratio are the eigenvectors $v_1,...,v_s$ of $W^{-1}B$ (scaled so that $v_i^T W v_i = 1$ $i = 1,...,s$). Here *s* is the number of strictly positive eigenvalues of $W^{-1}B$, and it can be shown that $s \leq \min(c-1, d)$. Using the notation $V = (v_1,...,v_s)$, it is easy to see that $\mathrm{cov}(V^T X) = I_s$, meaning that the components of the *new discriminant space* are uncorrelated and have unit variance.

There are several tests of significance of discriminant variables, i.e. their discriminatory power. The multivariate Wilk's lambda test statistic is used the most frequently. For a given set of discriminant variables, *Wilk's lambda test statistic* is a ratio of within-group/total sum of squares used in multivariate analysis of variance to test the null hypothesis that a group means are equal (for details see for example [8]).

Although DA method can be used solely for *descriptive* purposes (descriptive DA, **DDA**), it is more usefully employed as a *predictive* method (predictive DA, **PDA**), i.e. for classification.

For a new observation *x* to classify, the linear combinations $y_i = v_i^T x$ are called the values of the *i*-th *Fisher linear discriminant variables* (*i=1,…,s*).

The *Fisher classifier* is defined as follows: the new observation *x* is assigned to the population G_k if:

$$D_k(x) = \min_{j=1,...,c} D_j(x) \tag{4}$$

with the so-called Fisher discriminant scores:

$$\begin{aligned} D_j^2(x) &= [V^T(x-\mu_j)]^T [V^T(x-\mu_j)] - 2\log \pi_j = \\ &= \sum_{i=1}^{s} (y_i - \mu_{jY_i})^2 - 2\log \pi_j \end{aligned} \tag{5}$$

where μ_{jY_i} is the i-th component of the j-th group center in the discriminant space. The j-th discriminant score measures the (Euclidean) distance of the observation *x* to the j-th group center in the discriminant space. Moreover, since $s \le \min(c-1, d)$, DA method allows for a reduction of a dimensionality.

Stepwise discriminant analysis (**SDA**) [11] is concerned with selecting input variables by their contribution to the separation between the groups whilst retaining the highest discrimination power possible. The process of selecting a smaller number of variables is often necessary for a variety number of reasons. Removing variables which are redundant or are measuring the same aspect of group differences results in smaller number of variables that may be easier to interpret and provide a more simply solution than a larger number of variables. Variable reduction may also be necessary due to the cost of administering the large number of instruments.

In the existing statistical software packages (for example one of the most popular – STATISTICA [11]), SDA is realized as a *sequential forward* or *backward method* while looking at the Wilk's lambda for each variable, i.e. variables are entered in a stepwise fashion using Wilk's lambda criterion. In the first step of forward stepwise procedures, each variable is entered into a separate analysis, and the variable with the best univariate discrimination (lowest Wilk's lambda) is selected. Next, each remaining variable is paired with the first and entered into a separate analysis. The variable which, when paired with the first provides the best multivariate discrimination (again, the lowest Wilk's lambda) is selected next. The third step matches each remaining variable with the first two, and so on. This process is continued until either all variables are selected or the decrease in Wilk's lambda is insufficient to warrant further variable selection, as determined by the F-ratio.

Stepwise procedures can also be used in a reverse (i.e. backward) manner, to a similar effect.

Despite the frequency of their use, the SDA procedures entails a number of problems which can lead to misleading and inaccurate results, especially for predictive purposes [5]. The following group of problems have been revealed:

1) **Variable selection procedures**. Stepwise procedures do not always select the best subset of variables of a given size. By entering variables one at a time, stepwise procedures do not include all of the information supplied jointly by two or more variables not already included in the analysis. For example, some best subsets of a given size will never be considered.
2) **Capitalization on sampling error**. Stepwise procedures are especially suspect to sampling error. This is due to the fact that stepwise procedures select the variable with the lowest Wilk's lambda to be entered, no matter how small the difference.

3) **Selection criteria**. Stepwise procedures on common statistical packages are designed not for PDA, but for DDA. PDA is only concerned with hit rates, the number of cases correctly classified and does not utilize tests of statistical significance such as Wilk's lambda. This distinction is important, because Wilk's lambda cannot be adversely affected (made higher) by adding variables to a DDA, hit rates can be made worse. In DDA a completely worthless variable would be given a weight zero and its impact is essentially removed from the analysis. In PDA however, the same worthless variable would contribute "noise" to the prediction analysis, making group prediction less accurate.

From the above mentioned problems it follows, that using DDA stepwise selection procedures (implemented in commercial statistical packages) to receive results for PDA is both inaccurate and inappropriate.

3. Modelling the problem

Stepwise discriminant analysis can be formulated as a *feature selection problem* in pattern recognition [7,12] which is the process of selecting a subset of relevant features for their use in the model (i.e. classifier) construction. When the number of initial features becomes too large, the performance of the designed classifier (i.e. its error rate will not be guaranteed in the case of small sample size. As a consequence, for a given amount of samples, reducing number of features may result in improving classifier's performance.

Let F denotes the initial set of features with cardinality d, S is a subset of F *(a solution)*, $|S|$ - subset size, i.e., the number of inclusive features, p – the required subset size, t - the tolerable threshold, *err(S)* be the relevant error rate or some other measures of performance such as class separability or an error rate of a classifier, *f(S)* – the objective function to be minimized.

There are two forms of feature selection:

1) **Non-constrained optimization problem**: to find an optimal subset having a predefined number p of features and yield the lowest error rate of a classifier

$$\min \quad f(S) = err(S) \quad S \subset F \quad p = |S| < d \tag{6}$$

2) **Constrained optimization problem**: to seek the smallest subset of features for which error rate is below a given threshold.

$$\min \quad f(S) = |S| \quad S \subset F \quad err(S) < t \tag{7}$$

Feature selection problem is an example of special case of optimization problems, namely *combinatorial optimization* (**co**) problems [1] in which solutions are encoded with discrete variables. In co problems we are looking for an object from a finite or possibly countable infinite set, for example a subset of features.

An exhaustive approach to feature selection problem would require examining all $\binom{d}{p}$ possible p-subsets of the feature set *F*. The number of possibilities grows exponentially. Both forms of feature selection problem are *NP-hard* (Non Polynomial time) problems, so the optimal solution cannot be guaranteed except for doing exhaustive search in the solution space [6]. For a bigger values of *n*, the explosive computational cost makes the exhaustive search impracticable. On the assumption that the objective function is monotonic, the *branch and bound* (BB) algorithm is potentially capable of examining all feasible solutions, so it is optimal. But for large values of *n* it is still unavailable. Moreover, monotonic condition is seldom satisfied. Thus, the main stream of feature selection research was directed towards *suboptimal*, but efficient methods [6,7]. The early approaches of feature selection were based on probabilistic measures of class separability and on entropies. In some methods the independence of features was assumed and the features were selected on the basis of their individual merits. Such methods ignore the interactions among features, and the selected subsets are not satisfactory. As was pointed out in [6], the best two independent features do not have to be the two best. The *sequential forward selection method* (SFS), *sequential backward selection method* (SBS) belong to greedy algorithms. These algorithms begin with a feature subset and sequentially add or remove features until some termination criterion is met. But in these algorithms features selected cannot be removed later, and features discarded cannot be reselected (*the nesting effect*). Since these algorithms do not examine all possible feature subsets, they are not guaranteed to produce the optimal results. The *plus l take away r method* was proposed to prevent the nesting effect. But there is no theoretical guidance to determine the appropriate values of r and *l*. The sequential methods have been improved by introducing *sequential forward floating selection* (SFFS) and *sequential backward floating selection method* (SBFS) which avoid a problem of predefining *l* and *r* parameters in plus *l* take away *r* method. They achieve results comparable to the optimal BB algorithm, but are faster than BB [6].

To conclude, despite some progress has been obtained, the available feature selection techniques for large feature sets are not yet completely satisfactory. They are either computationally feasible but far from optimal, or they are optimal or almost optimal but cannot cope with the computational complexity of feature selection problems of realistic size.

Recently, there has been a resurgence of interest in applying feature selection methods due to the application needs. In the application of information fusion of multiple sensor's data, integration of multiple models and data mining, the number of features is usually quite large, in some cases it may be over 100 !

It is necessary to research more powerful methods for feature selection, which should give very good results and should be computationally more efficient.

4. The solution approach for stepwise discriminant analysis: tabu search algorithm

In this paper, we introduce the use of metaheuristic tabu search method for feature selection in discriminant analysis. As found in other optimization problems, metaheuristic techniques have proved to be superior methodologies. Tabu search [2] is one such example.

The tabu search (**TS**) proposed in [2] is a local search-based metaheuristic and is among the most cited and used metaheuristics for combinatorial optimization problems. Local search algorithms start from some initial solution and iteratively try to replace the current solution by a better one in an approximately defined neighborhood of the current solution, where the neighborhood is formally defined as follows: a *neighborhood structure* is a function $N: S \rightarrow 2^S$ that assigns to every $s \in S$ a set of neighbors $N(s) \subseteq S$, where $N(s)$ is called the *neighborhood of s.* The concept of neighborhood structure enables us to define the concept of *locally minimal solutions,* i.e. with respect to a neighborhood structure.

Metaheuristics are a new kind of approximate algorithms that try to combine basic heuristic methods in higher level frameworks aimed at efficiently and effectively exploring a search space. The term *metaheuristic* derives from the composition of two Greek words: *heuristic* derives from the verb *heuriskein* which means "to find", while the suffix *meta* means "beyond, in upper level". Up to now there is no commonly accepted definition for the term metaheuristic. We quote one of them [1]:

"A *metaheuristic* is formally defined as an iterative generation process which guides the subordinate heuristic by combining intelligently different concepts for exploring and exploiting the search space, learning strategies are used to structure information in order to find efficiently near-optimal solutions".

Tabu search differs from the local search technique in the sense that tabu search allows moving to a new solution which makes the objective function worse in the hope that it will not trap in the local optimum solutions. Tabu search explicitly uses the history of the search by using *a short-term memory*, both to escape from local minima, to avoid cycles and to implement an explorative strategy.

In our algorithm the short-term memory is implemented as a *tabu list* to record and guide the process of the search, i.e. that keeps track of the most recently visited solutions and forbids moves toward them. At each iteration the best solution from the allowed set is chosen as the new current solution. The solution that is picked at certain iteration is put in the tabu-list, so that it is not allowed to be reversed in the next l iterations, i.e. this solution is *tabu* (the l is the size of tabu list, the parameter). When the length of a tabu list reaches that size, then the first solution on a tabu list is freed from being tabu and a new solution enters that list. The process continues.

Each *solution* (i.e. the feature subset of size p) in our algorithm is represented as a vector of length d with the 0/1 element in a position i, indicating that a feature i ($i = 1,2,\ldots,d$) is not/is included in a subset. The *neighbourhood* $N(x)$ of a solution x is a set of solutions which are generated through adding randomly one feature on x meanwhile removing one feature. The neighborhood of the current solution is restricted to the solutions that do not belong to the tabu list. The *initial solution* is generated randomly, but it must have exactly the required number of features. The *objective function value* $f(x)$ of a solution s (i.e. a feature subset) is defined as a percentage of hits on a given dataset obtained through the features of s with Fisher's classifier. Termination condition is a predefined number of iterations.

The functioning of our complete tabu search algorithm for feature selection is outlined as follows:

***

Tabu search algorithm

(1) *Initialize*

Generate an initial solution x. Let $Sb = x$, $k = 1$, $TL = \emptyset$

/* Sb – the best solution obtained so far */

(2) *Generate neighborhood*

Generate neighborhood $N(x)$ of x

(3) *Move*

a) If $N(x) = \emptyset$ go to step (2), otherwise find out the best solution y in $N(x)$.

b) If y is in tabu list and $f(y)$ is not better than $f(Sb)$, let $N(x) = N(x) - \{y\}$, go to 3a), otherwise let $x = y$, $Sb = y$ if y is better than Sb

(4) *Output*

If termination condition is reached, stop and output Sb, otherwise add the new solution x to the tail of tabu list and if the length of the list exceeds a predefined size, remove the head of a list, let $k = k+1$, go to (2).

***

To explore in more detail the regions where the best solutions have been found up, we perform simple *intensification* of the search after each iteration of the described above tabu search procedure. The underlying idea is that in the path of two good solutions there should be a solution of similar quality, in some cases even better. Thus, based on the ideas presented in [3], we propose the following method for performing intensification.. A list *best_solutions* of the best k (k=6 in our case) solutions, i.e. the best subsets of features found up to the moment is maintained together with the list *best_performances* of performances corresponding to those solutions. At the beginning, the *best_solution* list is initialized with 6 best solutions found during first iterations of tabu search procedure. We then combine each pair of the solutions in the list to generate new solutions by building a "path", i.e. a chain of intermediate solutions between them. A path between two solutions is constructed in the following way: starting with the first solution we add or remove subsequent features until the second solution is reached. At each step there could be more than one possibilities – we choose the feature which results in the best overall performance of the Fisher classifier. After building the paths, we have created the set of solutions from which we choose the 3 best solutions and replace the 3 worse solution in the list *best_solutions* and check if the best solution *z* in the list *best_solutions* is better than the solution *Sb*. In such a case we replace *Sb* with *z*.

5. Experiments on benchmark datasets

5.1. Data description

To check the efficacy of the proposed feature selection method in discriminant analysis an experiment was run with the selected datasets. These datasets can be found in the well-known data repository of the University of California, UCI ([9]). Table 1 presents the short characteristics of the selected datasets. The meaning of the subsequent columns are as follows: *attr. type* – attribute type: *c* (categorical), *i* (integer), *r* (real); *#instances, attributes, classes* – nr of instances, attributes and classes in a dataset.

Table 1

The description of the datasets used in the experiment

dataset	attr. type	# instances	# attributes	# classes
hepatitis	c,i,r	155	19	2
indian-diabetes	i,r	768	8	2
liver-disorders	c,i,r	345	7	2
spectf-heart	i	267	44	2
spambase	i,r	4601	57	2

5.2. Data preprocessing

There are four types of data values: continuous, binary, ordered, and categorical. Discriminant analysis is originally designed for continuous valued datasets. However, with simple preprocessing, it can be used on any type of data sets, too. Values of a binary valued attribute can be translated into 0 and 1. Values of an ordered valued attribute can be translated into natural numbers according to their order. A categorical valued attribute can be replaced with the same number of binary attributes as its cardinality, each of which represents whether a value belongs to the corresponding category of the original attribute. For example, suppose that an attribute takes values from set $\{A,B,C\}$. It is replaced with three binary attributes, named *A,B,C*. Attribute *A* takes value *1* if the original attribute takes value *A* and 0 otherwise.

Many datasets contain missing values, discriminant analysis can be easily extended to handle such datasets. For our experiments, a missing value was replaced with the mean of the existent attribute values in the same class.

5.3. Results analysis

The conducted experiment consisted of comparing the performance of the Fisher classifier on a subset of features selected by the proposed tabu search-based feature selection algorithm with the performance obtained using stepwise procedures as implemented in the well known statistical package STATISTICA [11]. The *stepwise forward* procedure in the *Discriminant Analysis* module of STATISTICA package was selected, as *backward* procedure seems to work similar or, in some cases a bit worse than forward one.

Our tabu search-based algorithm for feature selection in discriminant analysis is implemented for the 1-st form of the mentioned feature selection problem, i.e. the non-constrained combinatorial optimization problem. This means that the proposed feature selection algorithm was run for the predefined values of the dimensionality p, starting with $p=1$ until the number d of features in a given dataset. For each value of p we have noted down the best performance of the Fisher classifier obtained through the predefined number of iterations, in our case 100. Tabu length list was set to $l=30$.

The classification performance of the Fisher classifier in the module Discriminant Analysis in STATISTICA package is implemented as the ratio of correctly classified to all cases on **the train dataset**. To be comparable, the performance of the Fisher classifier with our tabu search-based feature selection algorithm was computed in the same way, i.e. on the same train dataset. From statistical learning theory [4], it is known that these performances would be overestimated (this fact is important for predictive purposes).

In Table 2 we can see the comparison of the obtained performances (in %) of the Fisher classifier with our tabu search and forward stepwise/STATISTICA feature selection methods respectively, on the described above train datasets. The column (perf.all) presents the classification performance of Fisher classifier obtained with all features, while column (perf.stepwise) - the performance of the best solution obtained with the proposed feature selection algorithm, the column (#best subset) – gives the associated number of features comprising the best subset.

Table 2

The comparison of our tabu search and forward stepwise/STATISTICA methods on the selected train datasets: perf.all/perf.tabu – performances of the Fisher classifier (%) with all/selected by tabu subset of features, #subset – number of features in a subset selected by tabu search

data set	perf. all	perf. stepwise	best perf. tabu	# best subset
hepatitis	23.33	74,32	80.03	11
indian-diabetes	74.68	75,12	78.32	5
liver-disorders	59.42	66,02	70.26	5
spectf-heart	50.03	71,41	76.91	22
spambase	59.26	78,0	84.89	27

From Table 2 it can be seen that our feature selection algorithm improves the solutions obtained with stepwise procedures for feature selection implemented in software package STATISTICA for any case. Further improvements would be possible by modifying basic tabu search procedure to enable better exploration of a solution space. Also, the poor performance of the Fisher classifier on all features in a dataset shows the necessity of selecting feature subset to obtain better classifier performance.

6. Conclusions and future work

This work approaches the problem of variable selection in discriminant analysis. In fact, the most well known statistical packages continue to use classic selection methods like sequential forward/backward suffering from the nesting effect. Moreover, due to the criterion used for the evaluation of feature subsets – Wilk's lambda, they are designed for descriptive discriminant analysis only, not for predictive one.

We proposed the new feature selection algorithm based on metaheuristic tabu search that could be used instead of stepwise procedures for selecting input variables in discriminant analysis modules of the existing statistical packages.

After performing some tests, it is found that our tabu search-based feature selection algorithm obtained better results than stepwise forward/backward procedures implemented in STATISTICA package for stepwise discriminant analysis.

The presented feature selection algorithm based on tabu search could be further improved, for example by using more elaborated intensification for exploring the regions where the best solutions have been found up to this moment as well as a diversification of the search, i.e. directing the search towards unexplored regions. This will be the subject of the future research.

BIBLIOGRAPHY

1. Blum Ch., Roli A.: Metaheuristics in combinatorial optimization: overview and conceptual comparison, *ACM Computing Surveys*, vol. 35, 3, 2003, p. 268÷308.
2. Glover F.: Tabu Search. Part I, *ORSA Journal of Computing*, v.1, 1989, p. 190÷206.
3. Glover F. et al.: Fundamentals of scatter search and path relinking, *Control and Cybernetics*, v.29, 2000, p. 653÷684.
4. Hastie T., Tibshirani R., Friedman J.: *The elements of statistical learning*, Springer, New York, 2008.
5. Huberty C.J.: Problems with stepwise methods – better alternatives, in: Thompson B. (ed.): *Advances in Social Science Methodology*, vol.1, Greenwich, CI: JAI Press, 1989, p. 43÷70.
6. Jain A., Zonkger D.: Feature selection: evaluation, application, and small sample performance, *IEEE Trans. PAMI*, 19(2) , 1997, p. 153÷158.
7. Kohavi R., John G.: Wrappers for feature subset selection. *Artificial Intelligence*, vol. 97, 1-2, 1997, p. 234÷273.
8. Johnson R.A., Wichern, D.W.: *Applied multivariate statistical analysis*. Prentice-Hall, New Jersey 1998.
9. Murphy P.M., Aha D.W.: UCI repository of machine learning. University of California, Department of Information and Computer Science, [@:] http://www.ics.uci.edu/-~mlearn/MLRepository.html, 1994.
10. Pacheco J. et al.: Analysis of new variable selection methods in discriminant analysis, *Computational Statistics&Data Analysis*, v.51, 3, 2006, p. 1463÷1478.
11. STATISTICA – package documentation, StatSoft Inc., 2005.
12. Stąpor K.: *Classification methods in computer vision*. PWN, Warszawa 2011 (in Polish).
13. Tahir M.A., et al.: Feature selection using tabu search for improving the classification rate of prostate needle biopsis. *IEEE Proc. of the Int. Conf. on Pattern Recognition*, Cambridge 2004.
14. Zhang H, Sun G.: Feature selection using tabu search method, *Pattern Recognition*, 35, 2002, p. 701÷711.

Omówienie

Selekcja cech w module krokowej analizy dyskryminacyjnej w znanych szeroko pakietach do obliczeń statystycznych (np. STATISTICA [11]) wykonywana jest z wykorzystaniem

klasycznych metod sekwencyjnych w przód/w tył, których wadą jest efekt zagnieżdżania. Kryterium ewaluacyjne w tychże metodach – lambda Wilksa – jest dostosowane do celów deskryptywnych, a nie predyktywnych. Artykuł proponuje nowe rozwiązania wspomnianych problemów – algorytm selekcji cech z wykorzystaniem metaheurystyki przeszukiwania z tabu. W charakterze funkcji ewaluacyjnej zastosowano sprawność klasyfikatora Fishera. Algorytm selekcji składa się z fazy inicjalizacji – losowego rozwiązania początkowego, które w kolejnych iteracjach jest „ulepszane" poprzez poszukiwanie alternatywnych rozwiązań w jego sąsiedztwie z wykorzystaniem specjalnej struktury – listy tabu – aby zapobiec osiadaniu w minimach lokalnych. Wykonane wstępne testy wskazują na lepszą sprawność z użyciem proponowanej metody niż metod sekwencyjnych, zaimplementowanych w pakiecie STATISTICA.

Adresses

Katarzyna STĄPOR: Silesian University of Technology, Institute of Informatics, ul. Akademicka 16, 44-100 Gliwice, Polska, Katarzyna.Stapor@polsl.pl.