In: Workshop en Inteligencia Artificial, 2004, Arica. In: Jornadas Chilenas de Computacion. Santiago. Chile.

VARIABLE ORDERING APPLIED TO FEATURE SELECTION TASK UTILIZING ESTIMATIVE OF DISTRIBUTION ALGORITHM

Rodrigo Traleski[1] Aurora Trinidad Ramirez Pozo[2]

Abstract

The initial population of the evolutionary algorithm has a large importance on the performance and efficacy of its applications. The prior knowledge about the problem in point can change the procedure of this population creation. On this project, the importance of each feature obtained from the Qui-square Statistic test has been used as priors knowledge. With an initial population inducted on this test, the project intends to find the best searching space solution, through one forecast distribution algorithm, with a lower number of generations. The results obtained, when not the best ones, are statistically close to the results obtained with the algorithm by initial population randomly generated.

Keywords: Feature selection, evolutionary computation, estimate of distribution algorithm, variable ordering.

1. Introduction

The classification task is one, among several available tasks, used in data mining. Its aim is to find any kind of relationship between predictive attributes and the class attribute, discovering a knowledge that can be used to foresee the class of a not classified instance. Predictive attributes and one objective attribute or class composes an instance or register. The predictive attributes can be of two kinds: discreet or continuous. Otherwise, the class attribute must be discreet, assuming just one value from the available values on the set. One quality procedure used on the classification is the accuracy. The accuracy allows discovering the right grade of the submitted instances to the set. The higher is the right choice grade, the better is the accuracy and therefore better is the classification. Inside the predictive attributes can exist redundant or irrelevant attributes for a specific classification. The use of these attributes can interfere on the classification accuracy.

To execute the relevant attributes selection task, also known as feature selection, evolutionary algorithms have been used successfully [1]. The set of selected feature can reduce the computing cost to create the classifier model and enlarge the classification accuracy. A smaller set of feature can also improve the classification comprehension.

The Estimative of Distribution Algorithms (EDAs) are an example of evolutionary algorithms and has been used [2] on the feature selection task. The EDAs herd characteristics from the traditional simple genetic algorithms (sGA). The difference is on how the populations evolutes. Instead of using crossing and mutation, EDAs use a statistic model from individuals that had survived the selection to create new individuals. The individuals generation based in the statistic model preserves the existing relationships among the population genes.

This approach is considered by the scientific community as an important step to solve the problem of breaking the hypothesis constructor's blocks [3]. The BOA algorithm (Bayesian Optimization Algorithm) [4] is an example of EDA algorithm. BOA uses a Bayesian net to create the population probabilistic model. The creation process of a Bayesian net on BOA algorithm is more efficient when prior information about the problem is available.

The main target of this job is to introduce the prior knowledge on the BOA algorithm, based on the groups submitted to the feature selection. The variables ordering using the qui-square statistic test returns a list sorted by the importance of the predictive attributes in relation to the class attribute. This ordination will be introduced as prior knowledge, modifying the procedure of the initial population generation. The methodology described in [2] was used on the experiments to validate the insertion of prior knowledge. Two experiments were done: the first one was realized with BOA using the randomly generation of initial population. The second experiment used the BOA version with variables ordering with prior knowledge. The results got from these two experiments allow the analysis of the performance and quality of the obtained characteristics groups.

The based methods used on this project will be described on chapters two, three and four. On chapter five, the detailed description of the experiment and its results will be presented. The conclusions on chapter six.

2. Feature Selection

Feature selection methods are important in many situations where a huge set of characteristics is available and it's needed to select an appropriate subset to maximize the instances classification accuracy.

[1] Federal University of Paraná, rodrigo@inf.ufpr.br
[2] Federal University of Paraná, aurora@inf.ufpr.br

In a domain where objects are described by *d* characteristics, it's possible to obtain 2^d of possible subsets [5].

Two approaches for feature selection are possible. On the selection by filter, the characteristics are grabbed according to priorities considered good characteristics such as ortogonality and high content of information. Although this could be fast in one way, this approach by filter can produce results considered disappointing, because it ignores completely the introduction algorithm.

On the wrapper approach for characteristics selection, the key idea is to consider the inductor algorithm as a black box and use it as an algorithm for heuristic searching to evaluate each set of candidate characteristics [4]. After finding the set with the best evaluation, it's applied the inductor algorithm. The major disadvantage of this approach is the computing cost needed.

With the insertion of knowledge a prior on the set of data submitted to the characteristics selection, the learning process of the probabilistic model by the BOA algorithm can be optimized requiring a lower computing effort on this process. The computing cost of the wrapper approach is a problem faced by several proposed jobs for the task of characteristic selection, and its minimization is one of the main aims of this project.

3. Estimative of Distribution Algorithm

Evolutionary Algorithms, as the genetic algorithm (GA), depend on a large set of associated parameters, such as operators and crossing and mutation probabilities, population size, reproduction index and number of generations. The needed experience to the right use of these algorithms is a key factor for its application success. Thus the task of selecting the best values to these parameters is an optimization problem [6]. Besides, the use of crossing and mutation operators, from GAs applied in problems of simple searching, does not assure the preservation of hypothesis constructor's blocks. This problem occurs due to related genes in chromosomes with high value of fitness are split by the crossing and mutation operator and by the change realized by the mutation. More details can be found in [12].

The reasons above have motivated the creation of a new kind of algorithm classified as Estimative Distribution Algorithm (EDA) [7]. Besides, to reduce the number of associated parameters, the EDAs use the algorithms of searching data in population, based on probabilistic modulation of promissory solutions, associated with the simulation of a new induction model used to guide the search.

On the EDAs the new individuals population are created without the use of crossing and mutation operators. The new individuals are created from distributions of estimated probabilities of database containing only selected individuals from previous generations. On EDAs the relationships among variables that represent the individuals are explicitly expressed through the distributions of jointed probabilities, associated with selected individuals in each generation.

The following pseudo code shows a generic scheme of the EDA approach, which one follows essentially the following steps:

1. An initial population D_0 is randomly created with R individuals.
2. To create the $m+1$ population, D_m evolutes to the next D_{m+1} a number N $(N < R)$ of selected individuals from D_m according to a criterion. We call D^N_{m+1} the series of N selected individuals from generation m.
3. The probabilistic model of n-dimensions that better represents the relationships among the n variables is inducted. This kind of step is known as the *learning* procedure, and it's considered fundamental, once the dependencies among the variables interfere on the individuals evolution.
4. Finally, the new population D_{m+1} made by R individuals is obtained through the simulation of the distribution of probability learnt on the previous step.

The complexity to create a statistic model from the population is influenced by the number of relationships among the variables of the problem. Specifically, there are models where no relation were found among the variables, and the model is the simplest of all. In a second model the variables relate in pares, and the model became capable to represent some classes of problems where the relationships are equal or higher than two among the variables. The algorithms from this model are classified as NP-complete. The BOA algorithm is inside this classification.

The BOA uses a Bayesian Theory to create the probabilistic model that represents the relationships among the variables. The K2 [13, 14] algorithm, from the learning and punctuation approach, is used to learn the Bayesian net on BOA. The K2 uses a greedy searching algorithm to find a position where it could be inserted an arc between two variables of the problem. On each inserted arc, the Dirichlet metric is used to evaluate the quality of the net. As soon as there are no insertions that improve the net quality the algorithm is finished. The K2 algorithm is considered the most representative algorithm of the search and punctuation approach.

As well as the simple genetic algorithm (sGA), BOA needs to specify some needed parameters to its execution. The population size, selection and offspring method can be fixed in the same way than in the sGAs. Besides the traditional parameters of the sGAs, BOA has some specific parameters of the probabilistic models that will represent the population, as for instance, the number of arcs allowed to one variant. On this project the BOA algorithm initial parameters have not been

modified, and the truncation selection method has been used with a 50% selection index, also 50% substitution index.

4. The statistical qui-square test χ^2

The qui-square test is a statistical test that verifies if observed frequencies of variables are sufficiently close to the expected [8]. In data mining the qui-square test is widely applied, as in attributes discretization [9] and feature selection. In this work, the importance between the predictive attribute and the class attribute is determined by this test. The resulting order from this test allows us to grant privilege to the most important attributes to generate the initial population of the BOA.

The test allows verifying if two variables differ in relation to a specific characteristic, through the comparison of the relative frequency that each variable occurrence fits into one of the several categories [8]. This way it is possible to verify the independence between two variables using this test, which can be define as:

$$\chi^2 = \sum_{i=1}^{2} \sum_{j=1}^{k} \frac{(Oij - Eij)^2}{Eij}$$

Where, *Oij* are the observed frequencies, and *Eij* are the expected frequencies. A test application example is shown in [15].

5. Experiment descriptions

This work assumes prior knowledge of the data used on the experiment and the impact of this knowledge on the feature selection using the estimative distribution algorithms. The concept of variables ordering was used to determine the importance of the predictive attributes in relation to the class attribute. The order obtained allows us to fix the relative frequency of each gene, so the initial characteristics of the population are more interesting than when the population was selected randomly.

The next step is the description of the tools used, the data sets, and the methodology applied on this work.

5.1. Data set and algorithms

The data sets used on this work are listed on table 1. All data sets are from the UCI repository found on the internet. The data sets that had continuous predictive attributes were discretized using the Gauss normalization. This transformation was made using a function from the VFML [11] library. The discretization is necessary due to the use of the Naivebayes algorithm, also from the VFML library. The Naivebayes algorithm from this library has the limitations of working only with discrete data sets. This algorithm was chosen because it is fast and reliable when classifying using characteristics sets. The whole VFML library was developed on C and is also available on the internet.

The distribution estimative algorithm used was the BOA. It was developed on C++ and it can receive new fitness functions. The algorithm ends when it does not observe characteristics improvements between the last generation and the new one. The initial parameters of the BOA were not modified except for the fitness conditions that were specifically developed for the feature selection task.

For the variable ordering was used the WEKA library developed on Java and available on the internet for academic purposes. The variable ordering occurs using the data sets already discreticized. The resulting order was included on BOA to generate an initial population induced by the importance of the predictive attributes. That means the most important attributes will have the relative frequency of the equivalent gene closer to 1 (one), or 100% of the gene filled with one. As this works uses small data sets, the ordering was made using a 10 folds cross validation. On k-folds the data are randomly divided in k parts without overlapping, D1,...,Dk. On each *i* iteration (from 1 to k), the network is trained with D – Di and tested with Di. Cross validation is widely used when the data sets are small, where it is impossible to divided the training and evaluation set.

5.2. Fitness measurements

The BOA algorithm works with the binary representation of the population chromosomes. As for this work, if the gene has value 1 (one), it means that the feature was included on the data set; if it does not the feature was not included.

In order to evaluate the characteristics chosen the Naivebayes algorithm was used. The accuracy of the group of characteristics was verified through the cross validation classification method using 10 (ten) folds. For every group, the method described was applied a maximum of 5 (five) times or until a 1% of the standard deviation, between two obtained results. The final result is an average of the accuracy from the cross validation iterations. This fitness evaluation methodology was also used on [1] and [2].

The described methodology was used on both experiments described below.

5.3. Evaluation methodology and results

The BOA algorithm is evaluated with 5 (five) iterations of cross validation 2 folds. That means BOA was executed 10 times and in each iteration the data set was divided in half. One half was used to find the best subset of features using the fitness evaluation. The best subset of features is then used to determine the classification final accuracy. This final evaluation uses both parts of the data set; the first part is used for the

training and the second part for the evaluation. As well as in the fitness evaluation the Naivebayes algorithm was used.

To evaluate the changes that occur on the accuracy and on the number of generations, with an initial population created based on the variable ordering, the whole procedures used on the initial experiment were used. The main goal is to determine the relative frequency of the gene according to its position on the ordering list obtained. The more to the left the gene is on the list, the greater is its relative frequency. This way we try to guarantee that the important genes from the ordering will be on the final features subset. The pseudo code below shows the methodology use to define the relative frequency of each gene on the initial population. As an example, consider a data se with five characteristics.

```
Begin
    Order: vector[4, 2, 1, 3, 5] integer;
    Limit: vector[1..5] integer;

    K = Population size / Number of genes;

    For i = 1 until i <= Number of genes do
        Limit[Order[i]] = K * (Number genes–1);
    End-for
End
```

Table 1. Data set used on the experiment.

Domain	Instances	Classes	Numeric features	Discrete features	Noise
Ionosphere	351	2	34	-	N
Segmentation	2310	7	19	-	N
Sick Euthyroid	3163	2	7	18	S
Soybean Large	683	19	-	35	S

Table 2. Comparison between the accuracy and generation average. Best results in bold.

Domain	Random population	Generations	Population χ^2	Generations
Ionosphere	90,140567 ± 0,59	2,2	**90,365937 ± 0,64**	**2,0**
Segmentation	**91,353441 ± 0,14**	2,4	91,161112 ± 0,21	**2,3**
Sick Euthyroid	**97,102690 ± 0,05**	**2,1**	96,852201 ± 0,08	2,2
Soybean Large	88,320687 ± 0,45	2,3	**90,034229 ± 0,35**	**2,2**

Table 3. Accuracy obtained with and without the characteristics subset. Best results in bold.

Domain	With subset	Nº features	Without subset	Nº features
Ionosphere	**90,365937 ± 0,64**	20	89,179811 ± 0,35	34
Segmentation	**91,161112 ± 0,21**	12	89,477948 ± 0,16	19
Sick Euthyroid	**96,852201 ± 0,08**	17	96,034457 ± 0,09	25
Soybean Large	**90,034229 ± 0,35**	23	89,149294 ± 0,25	35

The *Order* vector contains the order given for the qui-square algorithm implemented on the WEKA library. *K* holds the ratio between the population size and the number of genes. That is equivalent to a proportional distribution of the population size and the number of genes. The For section is used to determine the limit, that is, the relative frequency of each gene inside the population.

The summarized steps for the evaluation of the experiment are:

1. The priority order of the data set characteristics described on table 1 are found using the qui-square algorithm from the WEKA library.
2. The data base is split in half, one part is used for training and the other is used for evaluation.
3. The BOA algorithm is executed with the first part of the data set divided before. The initial population is generated using the order defined on the first step.
4. The final characteristic group is evaluated with the second part from step 2.

5. Steps 2, 3 and 4, are executed 5 times, always starting on step 2 and stopping on step 4.
6. The average obtain from the 5 iterations is the final accuracy obtained with the experiment.

On table 2 are presented the results obtained with the BOA algorithm using the random generation method for the initial population and the results after the modification on the process for the generation of the initial population.

It is possible to observe that the proposed method sometimes does not reflects on a higher accuracy or a better number of generations, but it is not statistically inferior. The soybean and ionosphere groups obtained better result for both, even though ionosphere did not show a significant improvement. The sick and segmentation groups did not obtain better results for accuracy, but the difference was not significant between the two experiments. In general the proposed method obtained a smaller number of iterations. Despite of the addition of the qui-square algorithm on the initial step of the process, the results showed a reduction on the computational efforts, because the computational cost of the qui-square algorithm is linear.

6. Conclusions and Future Projects

This article presents an evaluation of utilization of previous knowledge in Estimative of Distribution Algorithm, when used on characteristics selection task. By the selection of the characteristics set it's possible to improve the classification accuracy and minimize the needed computing effort. The wrapper approach is more efficient on this kind of task. However, the needed computing cost is too high which makes this application not feasible many times. Several evolutionary algorithms have been used on this task, such as the EDAs.

With the selected evaluation methodology, it was not possible to find any evidence, which could support or refuse the use of prior knowledge in Estimative of Distribution Algorithms, when used on feature selection task. When the results are not the best, they are at least very close to the initial experiment.

Tests on other bases with similar features are needed in order to better evaluate the variable ordination application comportment on this task. On this project it has been used bases with high diversity features, which could have interfered on the final result.

The use of another variable ordering algorithm can also be applied together with the EDA algorithms. The use of another classifier algorithm, as the ID3, can be studied and evaluated along with this methodology.

7. References

[1] Inza, I., Larrañaga, P., Etxeberria, R., and Sierra, B. "Feature subset selection bye Bayesian networks based on optimization", *Artificial Inteligence*, 1999, pp. 157-184.

[2] Cantú-Paz, E., "Feature subset selection by estimation of distribution algorithms", *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2002)*, Morgan Kaufmann Publishers, San Francisco, 2002, pp. 303-310.

[3] Thierens, D., "Scalability problems of simple genetic algorithms", *Evolutionary Computation*, 1999, pp. 331-352.

[4] Pelikan, M., Goldberg, D. E., and Cantú-Paz, E., "The bayesian optimization algorithm", *Proceedings of the Genetic and Evolutionary Computation Conference*, Morgan Kaufmann Publishers, San Francisco, 199, pp. 525-532.

[5] John, G., Kohavi, R., and Phleger, K., "Irrelevant features and the feature subset problem", *In Proceedings of the 11th International Conference on Machine Learning*, Morgan Kaufmann Publishers, San Francisco, 1994, pp. 121-129.

[6] Grefenstette, J. J., "Optimization of control parameters for genetic algorithms", *IEEE Transactions on Systems, Man, and Cybernetics*, IEEE Press, 1986, pp. 122-128.

[7] Larrañaga, P., and Lozano, J. A., "Estimation of distribution algorithm. A new tool for evolutionary computation", Kluwer Academic Publishers, 2001.

[8] Degroot, M. H., "Probability and statistics (2nd ed.)", Addison-Wesley, 1986.

[9] Liu, H. and Setiono, R., "Chi2: Features Selection and discretization of Numeric Attributes". *Proceedings of the Seventh IEEE International Conference on Tools with Artificial Intelligence*, 1995, pp. 388-391.

[10] Witten, I. H. and Frank, E., "Data mining: Practical machine learning tools and techniques with Java implementations". Morgan Kaufmann, San Francisco, CA, 2000.

[11] Hulten, G. and Domingos, P. "VFML - A toolkit for mining high-speed time-changing data streams", http://www.cs.washington.edu/dm/vfml, 2003.

[12] Holland, J. H., "Adaptation in natural and artificial systems", The University of Michigan Press, 1975.

[13] Heckermann, D., Geiger, D. and Chickering, D. M., "Learning bayesian networks: The combination of knowledge and statistical data", *In KDD Workshop*, 1994, pp. 85-96.

[14] Cooper, G. F. and Herskovits, E., "A bayesian method for the induction of probabilistic networks from data", *In machine learning, vol. 9*, 1992, pp. 309-347.

[15] Hruschka, E. R. J., "Variable ordering for Bayesian networks learning from data", COPPE/Federal University of Rio de Janeiro, 2003.