The Extraction Method of DNA Microarray Features Based on Experimental *A* Statistics

Piotr Artiemjew

Department of Mathematics and Computer Sciences
University of Warmia and Mazury, Olsztyn, Poland
artem@matman.uwm.edu.pl

Abstract. The DNA microarray exploration topic is a really important area of research. Comparing samples of tissues we can find genes, which are characteristic of particular research problems. A number of researchers involved in bioinformatics are attempting to find effective gene extraction methods and classifiers, in order to predict particular medical problems. Even if we do not consider the ontological sense of genes, it is possible by information technology methods to find genes which are the most significant for a given research problem. An exemplary application of DNA microarrays can be the ability to detect some illnesses, personal identification, or distinguishing features of some organisms. In this work we describe our most effective (in the global sense) gene extraction method based on experimental *A* statistics, called SAM5. Next, we use the granular classifier based on weighed voting, which proved the best among those recently studied by Polkowski, and Artiemjew - 8_v1_w4 algorithm.

Keywords: rough sets, DNA microarrays, features extraction.

1 Introduction

This paper is dedicated to an application of the rough set methods - see [4,5], [6] - in the classification problems of DNA microarrays. The main motivation to use our rough set methods in the DNA microarray exploration was our participation in the recent DNA microarray data mining contest 2010 - see [9] - our algorithm based on modified Fisher method with our weighted voting classifier reached position eighteen on the basic track of this competition and was worse only by 3.5 per cent balanced accuracy than that of the winner. Since that time we have been carrying out intensive research on the new methods of gene extraction. The best result of our work was the idea of the DNA gene extraction methods based on experimental *A* statistics. *A* statistics is the modification of our approach, as suggested by Professor Polkowski. *A* statistics measure the separation ratio between pairs of decision classes by means of the difference in the indiscernibility ratio of descriptors and indiscernibility ratio descriptors and the average value of decision classes. The smaller the *A* statistics between decision classes for considered gene, the better the separation of decision classes.

J.T. Yao et al. (Eds): RSKT 2011, LNCS 6954, pp. 642–648, 2011.

In the first step we describe the weighted voting classifier - see [7] - based on residual rough inclusions - see [1].

2 Weighted Voting Classifier - 8_v1.4 Algorithm

The idea of our weighted voting classifiers is to decrease or increase weights depending on the case, as shown in [2], [7]. In this paper we use only the 8_v1.4 algorithm. This method reduces overfitting of classification by slight weakening of weights between descriptors.

The procedure is the following:

Step 1. The training decision system (U_{trn}, A, d), and the test decision system (U_{tst}, A, d) have been input.

Step 2. The maximal, and the minimal value of attribute a on the training set have been found, and marked as max_attr_a, and min_attr_a respectively.

Step 3. The attribute similarity degree ε has been input.

Step 4. The classification of the test objects is done as follows:

For all conditional attributes $a \in A$, training objects $v_p \in U_{trn}$, where $p \in \{1, ..., card\{U_{trn}\}$, and the test objects $u_q \in U_{tst}$, where $q \in \{1, ..., card\{U_{tst}\}$ we compute

(i) If $\frac{|a(u_q)-a(v_p)|}{max\_attr_a - min\_attr_a} \geq \varepsilon$, then

$$w(u_q, v_p) = w(u_q, v_p) + \frac{|a(u_q) - a(v_p)|}{(max\_attr_a - min\_attr_a) * \varepsilon + |a(u_q) - a(v_p)|} \quad (1)$$

(ii) If $\frac{|a(u_q)-a(v_p)|}{max\_attr_a - min\_attr_a} < \varepsilon$, then

$$w(u_q, v_p) = w(u_q, v_p) + \frac{|a(u_q) - a(v_p)|}{(max\_attr_a - min\_attr_a) * \varepsilon} \quad (2)$$

After weights are computed for a given test object u_q, and each training object v_p, the voting procedure consists of computing the parameter values,

$$Param(v_d) = \sum_{\{v_p \in U_{trn} : d(v_p) = v_d\}} w(u_q, v_p), \text{for } \forall \ c, \text{ decision classes.} \quad (3)$$

Finally the test object u_q is classified into the class v_d with the minimal value of $Param(v_d)$.

Having described the classification method, we can return to the main point of our paper - the gene extraction method.

3 Feature Extraction Method Based on A Statistics - SAM5

First of all, we make the following assumption. We let,

$$train_{C_i^a, C_j^a} = max\_attr_{C_i^a, C_j^a} - min\_attr_{C_i^a, C_j^a} \quad (4)$$

where $max\_attr_{C_i^a,C_j^a}$, $min\_attr_{C_i^a,C_j^a}$ is respectively a maximal, and a minimal value of the attribute a in both training decision classes C_i^a, and C_j^a.

For the decision system (U, B, d), where $U = \{u_1, u_2, ..., u_n\}$, $B = \{a_1, a_2, ..., a_m\}$, $d \notin B$, classes of d: $c_1, c_2, ..., c_k$, we propose to obtain the rate of the separation of the gene $a \in B$ for a pair of decision classes c_i, c_j, where $i, j = 1, 2, ..., k$, and $i < j$ in the following way. We let,

$$A_{c_i,c_j}(a) = C_i^a \wedge_\varepsilon C_j^a \tag{5}$$

$$C_k^a = \{a(u) : u \in U \text{ and } d(u) = c_k\}, \overline{C}_k^a = \frac{\{\sum a(u) : u \in U \text{ and } d(u) = c_k\}}{card\{C_k^a\}}, k = i, j. \tag{6}$$

where,

$$C_i^a \wedge_\varepsilon C_j^a = \frac{card\{a(u) \in C_i^a : \exists a(v) \in C_j^a; \frac{|a(u)-a(v)|}{train_{C_i^a,C_j^a}} \leq \varepsilon\}}{card\{C_i^a\} + card\{C_j^a\}} +$$

$$\frac{card\{a(v) \in C_j^a; \exists a(u) \in C_i^a; \frac{|a(v)-a(u)|}{train_{C_i^a,C_j^a}} \leq \varepsilon\}}{card\{C_i^a\} + card\{C_j^a\}} - \tag{7}$$

$$\frac{card\{a(u) \in C_i^a : \frac{|a(u)-\overline{C}_j^a|}{train_{C_i^a,C_j^a}} \leq \varepsilon\} + card\{a(v) \in C_j^a : \frac{|a(v)-\overline{C}_i^a|}{train_{C_i^a,C_j^a}} \leq \varepsilon\}}{card\{C_i^a\} + card\{C_j^a\}}$$

After the rate of separation $A_{c_i,c_j}(a)$ is computed for all the genes $a \in B$, as well as all of the pairs of decision classes c_i, c_j, where $i < j$, the genes are sorted by increasing order of , $A_{c_i,c_j}(a)$

$A^1_{c_{i_1},c_{i_2}} < A^2_{c_{i_1},c_{i_2}} < ... < A^{card\{B\}}_{c_{i_1},c_{i_2}}$, where $i_1 \in \{1, 2, ..., k-1\}$ and $i_2 \in \{i_1+1, ..., k\}$

Finally, we choose for experimentation a fixed number of genes from the sorted list by means of the procedure,

Procedure
Input data
$B' \leftarrow \emptyset$
$iter \leftarrow 0$
for $i = 1, 2, ..., card\{B\}$ **do**
  for $j_1 = 1, 2, ..., k-1$ **do**
    for $j_2 = j_1 + 1, ..., k$ **do**
      if $A_{c_{j_1},c_{j_2}}(a) = A^i_{c_{j_1},c_{j_2}}(a)$ and $a \notin B'$ **then**
        $B' \leftarrow a$

```
        iter ← iter + 1
        if iter = fixed number of the best genes then
          BREAK
        end if
      end if
    end for
    if iter = fixed number of the best genes then
      BREAK
    end if
  end for
  if iter = fixed number of the best genes then
    BREAK
  end if
end for
return B'
```

The results for our algorithm with a real DNA microarray - see Tab. 1 - from the Advanced Track of Discovery Challenge - see [8], and [9] - are reported in the next section.

4 The Results of Experiments for Exemplary DNA Microarray Data

The examined data - see Tab. 1 - can be interpreted without any context, because all the decision classes are classified in the general sense as one big decision system. The decision classes of examined DNA microarrays are unbalanced - see Tab. 1. For this reason we evaluate the results by balanced accuracy, average accuracy from all decision classes.

In this paper we apply our best classification algorithm 8_v1.4 based on weighted voting with the fixed parameter $\varepsilon = 0.01$, and our SAM5 feature

Table 1. An information table of the examined data sets - see [8]; data1 = anthracyclineTaxaneChemotherapy, data2 = BurkittLymphoma, data3 = HepatitisC, data4 = mouseType, data5 = ovarianTumour, data6 = variousCancers_final

Data	*No.attr*	*No.obj*	*No.class*	*The.dec.class.details*
*data*1	61359	159	2	1(59.7%), 2(40.2%)
*data*2	22283	220	3	3(58.1%), 2(20%), 1(21.8%)
*data*3	22277	123	4	2(13.8%), 4(15.4%), 1(33.3%), 3(37.3%)
data4	45101	214	7	3(9.8%), 2(32.2%), 7(7.4%), 6(18.2%), 5(16.3%), 4(9.8%), 1(6%)
*data*5	54621	283	3	3(86.5%), 1(6.3%), 2(7%)
data6	54675	383	9	3(6.2%), 2(40.4%), 4(10.1%), 7(5.2%), 5(12.2%), 6(10.9%), 8(4.1%), 9(4.6%), 10(5.7%)

Table 2. Leave One Out; The average balanced accuracy of the classification for the implemented methods; Examined data sets: all from Tab. 1; No.of.genes = the number of classified genes, method = method's name

*method**No.of.genes*	10	20	50	100	200	500	1000
*SAM*5	0.718	0.77	0.815	**0.841**	**0.84**	**0.846**	**0.833**

Table 3. Leave One Out; The balanced accuracy of the classification for particular decision systems for method SAM5 at 500 genes; Examined data sets: all from Tab. 1

*data*1	*data*2	*data*3	*data*4	*data*5	*data*6	*Average balanced accuracy*
0.866	0.932	0.91	0.581	0.913	0.876	**0.846**

Table 4. Leave One Out; Average balanced accuracy; A statistics vs F statistics [1], and the Advanced Track results of the Discovery Challenge [9]; Examined data sets: all from Tab. 1; in case * a result for 500 genes, in case ** a result for 50 genes

method	*Balanced Accuracy*
*SAM*5*	**0.846**
*MSF*6** [1]	**0.789**
RoughBoy [9]	0.75661
ChenZe [9]	0.75180
wulala [9]	0.75168

extraction method, with the result evaluated by means of the Leave One Out algorithm (LOO). The motivation to use the LOO method is to be found, among other places, in [3], and [9]. It's obvious that the LOO method does not produce the best result for all kinds of data, but it did work very well during the above-mentioned contest - see the winning solution [9].

4.1 The Results of SAM5 Gene Extraction Method

The SAM5 DNA microarray gene separation method based on experimental A statistics produces the best average results, in the global sense, among all of the methods that we studied. On the basis of the average results - see Tab. 2 - we can conclude that for 100, 200, 500, and 1000 genes it works best. The best balanced accuracy 0.846 for all examined data was obtained with 500 genes - see Tab. 3. In the Tab. 4 we can see the comparison of our results, and the Advanced Track RSCTC'2010 discovery challenge winners' results. The results show

that our method is fully comparable to other methods. Additionally, in the Tab. 4, we can see the results for our A statistics, compared with our (MSF6) algorithm - see [1] - based on modified F statistics.

5 Conclusion

The results of the research show, beyond doubt, an advantage of the SAM5 method compared with the gene separation, and classification methods that we saw at Advanced Track. Our results have been confirmed by the average balanced accuracy results. It turns out that the SAM5 method works best for a high number of the genes in the range of 100, 200, 500, and 1000, making use of lowering the value of the product of a pair of the decision classes by the indiscernibility degree of the elements of a given class from the average value of the paired decision class - which is a characteristic element of the SAM5 method. The essential element of the SAM5 method is the way of choosing the best genes after their calculation for particular pairs.

In future work we will search for the explanation of the effectiveness of our gene extraction methods based on A statistics, and for the extension of their theoretical description.

Acknowledgements. The research has been supported by grant 1309-802 from the Ministry of Science and Higher Education of the Republic of Poland.

References

1. Artiemjew, P.: The Extraction Method of DNA Microarray Features Based on Modified F Statistics vs. Classifier Based on Rough Mereology. In: Kryszkiewicz, M., Rybinski, H., Skowron, A., Raś, Z.W. (eds.) ISMIS 2011. LNCS(LNAI), vol. 6804, pp. 33–42. Springer, Heidelberg (2011)
2. Artiemjew, P.: On strategies of knowledge granulation and applications to decision systems. PhD Dissertation, Polish Japanese institute of Information Technology, Polkowski, L (Supervisor), Warsaw (2009)
3. Molinaro, A.M., Simon, R., Pfeiffer, R.M.: Prediction error estimation: a comparison of resampling methods. Bioinformatics 21(15), 3301–3307 (2005)
4. Polkowski, L.: Toward rough set foundations. Mereological approach (a plenary lecture). In: Tsumoto, S., Słowiński, R., Komorowski, J., Grzymała-Busse, J.W. (eds.) RSCTC 2004. LNCS (LNAI), vol. 3066, pp. 8–25. Springer, Heidelberg (2004)
5. Polkowski, L.: Formal granular calculi based on rough inclusions (a feature talk). In: IEEE International Confrence on Granular Computing, GrC 2005, pp. 57–62. IEEE Press, Los Alamitos (2005)
6. Polkowski, L.: A Unified Approach to Granulation of Knowledge and Granular Computing Based on Rough Mereology: A Survey. In: Pedrycz, W., Skowron, A., Kreinovich, V. (eds.) Handbook of Granular Computing, pp. 375–401. John Wiley & Sons, New York (2008)
7. Polkowski, L., Artiemjew, P.: On classifying mappings induced by granular structures. In: Peters, J.F., Skowron, A., Rybiński, H. (eds.) Transactions on Rough Sets IX. LNCS, vol. 5390, pp. 264–286. Springer, Heidelberg (2008)

8. Tuned IT platform, http://tunedit.org/repo/RSCTC/2010/A
9. Wojnarski, M., Janusz, A., Nguyen, H.S., Bazan, J., Luo, C., Chen, Z., Hu, F., Wang, G., Guan, L., Luo, H., Gao, J., Shen, Y., Nikulin, V., Huang, T.-H., McLachlan, G.J., Bošnjak, M., Gamberger, D.: RSCTC 2010 Discovery Challenge: Mining DNA Microarray Data for Medical Diagnosis and Treatment. In: Szczuka, M., Kryszkiewicz, M., Ramanna, S., Jensen, R., Hu, Q. (eds.) RSCTC 2010. LNCS(LNAI), vol. 6086, pp. 4–19. Springer, Heidelberg (2010)
10. Zadeh, L.A.: Fuzzy sets and information granularity. In: Gupta, M., Ragade, R., Yager, R.R. (eds.) Advances in Fuzzy Set Theory and Applications, pp. 3–18. North Holland, Amsterdam (1979)