Avoiding data overfitting in scientific discovery: Experiments in functional genomics

Dragan Gamberger[1] and **Nada Lavrač**[2 3]

Abstract. Functional genomics is a typical scientific discovery domain characterized by a very large number of attributes (genes) relative to the number of examples (observations). The danger of data overfitting is crucial in such domains. This work presents an approach which can help in avoiding data overfitting in supervised inductive learning of short rules that are appropriate for human interpretation. The approach is based on the subgroup discovery rule learning framework, enhanced by methods of restricting the hypothesis search space by exploiting the relevancy of features that enter the rule construction process as well as their combinations that form the rules. A multi-class functional genomics problem of classifying fourteen cancer types based on more than 16000 gene expression values is used to illustrate the methodology.

1 Introduction

Recent research in the construction of high dimensional classifiers as well as in combining different classifiers has enabled that good prediction quality can be obtained also for gene expression domains which are characterized by unproportionally many attributes compared to the number of available examples [3, 7, 11]. The obtained results are promising for the applications of functional genomics in the tasks like disease diagnosis, disease forecasting, or therapeutic decision making. But the constructed classifiers are not appropriate for human interpretation [1]. Construction of understandable and explainable models, also known as descriptive induction, is important for scientific discovery as well as for the generation of actionable knowledge. It is possible to extract the most informative features or attributes from complex classifiers (in [11] the attributes with this property are called disease markers) but logical connections among these features or attributes are missing which disables the construction and expert interpretation of models describing the target class. In contrast, short rules—despite being potentially less accurate than the complex classifiers—are much more appropriate for scientific discovery tasks in which the interpretability of induced models is of ultimate importance.

The problem with the induction of low dimensional non-redundant classifiers is that they are very sensitive to training set overfitting, an effect which denotes that a classifier has significantly lower prediction quality on unseen test sets than on the training set [12]. Although maximal prediction accuracy is not the main goal of scientific discovery and descriptive induction tasks, high generalization error or large difference in prediction quality for the training and the test set are a reliable sign that the induction of a classifier was not successful in finding really relevant relations between attribute values and the classes. This is the main reason of our interest in avoiding data overfitting. Selection of an appropriate hypothesis language as well as the reduction of the hypothesis search space are known methods for avoiding overfitting [2, 8, 12].

In this work an approach is presented which, although it can not guarantee that overfitting will not happen, may help in avoiding overfitting in inductive learning of simple rules which are appropriate for human interpretation. For illustrations and experiments a gene expression domain with more than 16000 attributes is used. The approach is based on methods for reducing the hypothesis search space so that the reduction is done by the elimination of those rule components (features) and those of their combinations (rules) suspected to be a result of statistically irrelevant artifacts. The problem with this approach is that with a strongly reduced hypothesis space it may be difficult to induce rules that cover all/many examples from the training set. However, the proposed subgroup discovery approach provides a much better framework for the application of the suggested methodology of feature and rule relevancy than the standard separate-and-conquer rule learning [4].

In Section 2 of this paper the subgroup discovery methodology is described, while feature relevancy and rule relevancy are presented in Sections 3 and 4, respectively. Section 5 presents the experiments using the proposed methodology on a gene expression domain.

2 Rule learning for subgroup discovery

Subgroup discovery is a form of supervised inductive learning of subgroup descriptions for the target class in a two class domain. The descriptions have the form of rules built as logical conjunctions of features. Features are logical conditions that have values true or false, depending on the values of attributes which describe the examples in the problem domain. Subgroup discovery rule learning is therefore a form of two-class propositional inductive rule learning. Multi-class problems can be solved as a series of two-class learning problems, so that each class is once selected as the target class while examples of all other classes are treated as non-target class examples.

In this work, subgroup discovery is performed by the SD algorithm,[4] a relatively simple iterative beam search rule learning algorithm [5]. The input to SD consists of a set of examples E ($E = P \cup N$, P is the set of target class examples, N the set of non-target class examples) and a set of features F constructed for the given

[1] Rudjer Bošković Institute, Zagreb, Croatia email: dragan.gamberger@irb.hr
[2] Jožef Stefan Institute, Ljubljana, Slovenia email:nada.lavrac@ijs.si
[3] Politehnika Nova Gorica, Nova Gorica, Slovenia

[4] The approach has been implemented in the on-line Data Mining Server (DMS), publicly available at `http://dms.irb.hr`. DMS and its constituting subgroup discovery algorithm SD can be tested on user submitted domains with up to 250 examples and 50 attributes.

example set. For discrete (categorical) attributes, features have the form $Attribute = value$ or $Attribute \neq value$, while for continuous (numerical) attributes they have the form $Attribute > value$ or $Attribute \leq value$. The output of the SD algorithm is a set of rules with optimal covering properties on the given example set. As in classification rule learning, an induced rule (subgroup description) has the form of a (backwards) implication: $Class \leftarrow Cond$. In terms of rule learning, the property of interest for subgroup discovery is the target class ($Class$) that appears in the rule consequent, and the rule antecedent ($Cond$) is a conjunction of features (attribute-value pairs) selected from the features describing the training instances.

A rule with ideal covering properties is true for all target class examples and not true for all non-target class examples. Target class examples covered by a rule are also called *true positives, TP*, while non-target class examples covered by the rule are called *false positives, FP*. All remaining non-target class examples not covered by the rule are called *true negatives, TN*. An ideal rules has $TP = P$ and $TN = N$. In the proposed subgroup discovery approach [5], the following rule quality measure q_g is used in heuristic search of rules: $\frac{|TP|}{|FP|+g}$, where g is a user defined *generalization parameter*. High quality rules will cover many target class examples and a low number of non-target examples. The number of tolerated negative examples, relative to the number of covered target class cases, is determined by parameter g.

The flexibility of subgroup discovery is due to its search of rules that satisfy groups of examples of the target class, not necessary excluding all of the non-target examples. Sizes of subgroups are not defined in advance but the algorithm tends to make them as large as possible. Due to this flexibility the algorithm is able to incorporate different rule relevancy methods with the goal to prevent the construction of target class subgroup descriptions which do not have sufficient supportive evidence for being significantly different from non-target samples. An equally important part of the methodology for avoiding overfitting is that each feature that enters the subgroup discovery algorithm should itself be a relevant target class descriptor.

3 Relevancy of features

The relevancy of features is determined by a combination of methods for restricting the hypothesis search space and for eliminating features with low covering properties. The later methods based on absolute and relative relevancy (Section 3.2) are universally applicable to any domain and their use is suggested in all feature based inductive learning tasks. The restrictions of the hypothesis search space are related to the form of rules and to the properties of the domain. Section 3.1 presents an effective approach that can strongly reduce the number of features and its application is suggested for descriptive induction tasks in gene expression domains.

The features are restricted to simple forms only, as defined in the previous section, because their complex forms may enable that, despite testing feature covering properties, features with insufficient supportive evidence may enter the rule construction process. For example, for discrete attributes the simple features have the form $A_i = a$ or $A_i \neq a$. No complex logical forms like ($A_i = a \wedge A_j = b$) or ($A_i = a \vee A_j = b$) are acceptable. The first form is not needed as all potential conjunctions are tested by the beam search procedure of the subgroup discovery algorithm. The second form is dangerous because, for example, the feature $A_i = a$ may be relevant while the feature $A_j = b$ may be irrelevant. Their combination $A_i = a \vee A_j = b$ may be even more relevant than $A_i = a$ itself, which may cause that condition $A_j = b$ may be included into the finally constructed rules while its inclusion is not justified by its covering properties on the training set. Notice that if both conditions $A_i = a$ and $A_j = b$ are relevant, it does not mean that by restricting the form of used features some important logical combinations of features will be ignored. In the subgroup discovery approach both features can build separate subgroup descriptions and—if they are relevant—they both have a chance to appear in the final set of induced rules.

3.1 Domain specific restrictions for functional genomics domains

Gene expression scanners measure signal intensity as continuous values which form an appropriate input for data analysis. The problem is that for continuous valued attributes there can be potentially many boundary values separating the classes, resulting in many different features for a single attribute. Another possibility is to use presence call (signal specificity) values computed from measured signal intensity values by the Affymetrix GENECHIP software. The presence call has discrete values A (absent), P (present), and M (marginal). The M value can be interpreted as a *'do not know state'*, while for values A and P it holds that feature $Attribute = A$ is identical to $Attribute \neq P$. Consequently, for every attribute there are only two distinct features $Attribute = A$ and $Attribute = P$ generated for each attribute.

The presented subgroup discovery algorithm as well as the filtering based on feature and rule relevancy are applicable both when using the signal intensity or the presence call attribute values. Typically signal intensity values are used [10] because they impose less restrictions on the classifier construction process and because the results do not depend on the GENECHIP software presence call computation. For descriptive induction tasks we prefer the later approach based on presence call values. The reason is that features presented by conditions like A_i is true (A_i is present) or A_j is false (A_j is absent) are very natural for human interpretation. Although present GENECHIP software presence call computation is perhaps not ideal, some expert evaluation results demonstrate that it can enable induction of very interesting rules both because of the ease of their interpretation and because of their predictive quality.

A more important reason for using presence call values is that the approach can help in avoiding overfitting, as the feature space is very strongly restricted: instead of many features per attribute we have only two. Also, as the measured gene expression values are not completely reliable (which is reflected by the fact that for the same sample measured values may change from one measurement to another), some robustness of constructed rules is welcome. To some extent, this can be achieved by treating the marginal presence call attribute value M as a 'do not know state'. The value can neither be used to support the relevancy of a feature or a rule, nor it can be used for prediction purposes. In this way it additionally restricts the hypothesis search space.

The domain specific restrictions presented in this section are characteristic for the functional genomics machine learning problems but similar approaches can be defined also for other domains and their use is recommended.

3.2 Absolute and relative feature relevancy

In order for a feature to be acceptable as a building block of rules representing some genuine dependencies between classes and attribute values, the feature itself must have at least some quality which is measured by its covering properties on the available training set.

Definition 1: absolute irrelevancy
A feature that has either $|TP| < min\_tp$ or $|TN| < min\_tn$ is absolutely irrelevant, for min_tp and min_tn being user defined constants.

A feature with $|TP| < min\_tp$ is true for a small number of target class examples and a feature with $|TN| < min\_tn$ is false for a small number of non-target class examples. It is assumed that such small numbers may be the result of statistical chance so that it seems reasonable not to use features with either of these properties in the rule construction process. If a feature has $|TP| = 0$ or $|TN| = 0$ it is totally irrelevant because it is absolutely of no use in building rules that distinguish one example class from the other.

By conjunctive connection of features, the generated rule will have $|TP|$ equal or smaller than the smallest $|TP|$ value of the features forming a conjunctive subgroup description. In contrast, the $|TN|$ value of a rule will be at least as large as the largest $|TN|$ of the used features. This is the reason why min_tp is typically selected higher than min_tn and it can be as large as the minimal estimated number of examples that must be covered by any acceptably good subgroup for the domain. The problem with absolute irrelevancy is that both min_tp and min_tn are user defined constraints and that any value, regardless how low it is, can not guarantee that a feature is actually relevant. The optimal values for these constants may significantly change from one application to another. A practical suggestion is to start with their small values and after that to experiment with larger values. The optimal point is just before significant decrease of the covering properties of induced rules can be noticed. A good starting values for gene expression domains are $min\_tp = |P|/2$ and $min\_tn = \sqrt{|N|}$ which have been used in all the experiments reported in Section 5.

While the aim of using absolute relevancy is to ensure minimal quality that must be satisfied by every feature, relative relevancy aims to ensure that only the best among the available features can enter the rule construction process.

Definition 2: relative irrelevancy
A feature f is irrelevant if there exists another feature f_{rel} such that true positives of f are a subset of true positives of f_{rel} and true negatives of f are a subset of true negatives of f_{rel}.

If for a feature f there exists another feature f_{rel} with the property that if in any rule f is substituted by f_{rel}, the rule quality measured by the number of correct classifications $|TP|$ and $|TN|$ does not decrease, then it means that f_{rel} can be always used instead of f, and that we actually do not need f. The definition of relative irrelevancy is very important because it does not depend on user defined constants and its usage is suggested for all machine learning applications [9]. Interesting and important relations between absolute and relative feature relevancy for real and randomly generated domains are described in Section 5.

If all features generated for an attribute are detected as relatively or absolutely irrelevant then it means that the complete attribute is actually irrelevant for the domain. This property means that feature relevancy can be used also as a preprocessing filter for attribute based learners.

4 Relevancy of rules

As in the case of feature relevancy, relevancy of rules may be both domain related and general.

Domain related relevancy typically means that some combinations of features are unacceptable, or which is more often, that some features are preferred as rule building blocks. The subgroup discovery algorithm enables the inclusion of various forms of expert knowledge or preferences into the rule construction process. In gene expression domains there is not a lot of available expert knowledge and the only applied strategy, also suggested for other descriptive induction tasks, is to limit the maximal complexity of rules. This complexity is determined by the number of features used in any rule and it can be easily bounded by the maximal number of iterations in the main loop of the subgroup discovery algorithm. This approach is suggested because it also restricts the hypothesis search space.

The domain unrelated rule relevancy consists of two main parts. The first is related to the form of induced rules. As mentioned earlier, the subgroup discovery algorithm constructs only rules in the form of conjunctions of features. Disjunctions are not allowed because they can enable that some parts of the rule are actually irrelevant even when the complete rule seems relevant. The reasoning is the same as for features. The restriction does not disable the detection of some relevant dependencies because different relevant disjunctive parts may still be detected as distinct subgroups.

The second part of conditions for general rule relevancy is based on rule covering properties. It can be compared with absolute and relative relevancy of features. Absolute rule relevancy is ensured by absolute relevancy of features that are used in the rule construction process. As conjunctions of features may reduce the number of target class examples covered by the rule to fall below an acceptable level, an additional absolute condition based on minimal rule support level is used. As the support level is defined by the relation $|TP|/|E|$ this condition ensures minimal covering of target class examples that must be satisfied by any rule. The good news is that this condition must be satisfied also by any subrule of a rule that must satisfy this condition. Consequently, this condition can be incorporated in the subgroup discovery algorithm to prevent any combination of features below the given support level to enter the beam. The minimal acceptable support level is a user defined constant, whose default value $|P|/2|E|$ was used in the described experiments.

The quality measure to be maximized in the beam search for best rules $q_g = \frac{|TP|}{|FP|+g}$ serves as the relative rule relevancy criterion. In ensures that rules covering many target class examples and no or a few of non-target class examples are preferred to those covering a small number of target class examples and many non-target class ones. The problem with this quality measure is that, in contrast to the relative relevancy of features, relative rule relevancy is conditional because it depends on a user selected generalization parameter g. Typically the user must experiment with different values of the parameter.[5]

5 Experiments with overfitting in a gene expression database

The gene expression domain, described in [11, 6] and used in our experiments, is a typical scientific discovery domain characterised by very many attributes compared to the number of available examples. It is a domain with 14 different cancer classes and 144 training examples in total. Eleven classes have 8 examples each, two classes have 16 examples and only one has 24 examples. The examples are described by 16063 attributes presenting gene expression values. As mentioned in Section 3.1 in all experiments we used only the presence call values A, P, and M. The domain can be downloaded from http://www-genome.wi.mit.edu/cgi-bin/cancer/datasets.cgi . There is also an independent test set with 54 examples.

[5] Suggested g values in the SD algorithm in the Data Mining Server are in the range between 0.1 and 100, for analyzing data sets of up to 250 examples.

The experiments were performed separately for each cancer class so that a two-class learning problem was formulated where the selected cancer class was the target class and the examples of all other classes formed non-target class examples. In this way the domain was transformed into 14 inductive learning problems, each with the total of 144 training examples and with between 8 and 24 target class examples. For each of these tasks a complete procedure consisting of feature construction, elimination of irrelevant features, and induction of subgroup descriptions in the form of rules was repeated. Finally, using the SD subgroup discovery algorithm [5], for each class a single rule with maximal q_g value has been selected, for $q_g = \frac{|TP|}{|FP|+g}$ being the heuristic of the SD algorithm, and g being equal 5 in all experiments presented in this work. The rules for all 14 tasks consisted of 2–4 features. The induced rules were tested on the independent example set. The results are presented in Table 1.

Table 1. Covering properties on the training and on the independent test set for rules induced for 14 classes. Sensitivity is $|TP|/|P|$, specificity is $|TN|/|N|$, while precision is defined as $|TP|/(|TP| + |FP|)$.

Cancer	Training set			Test set		
	Sens.	Spec.	Prec.	Sens.	Spec.	Prec.
breast	5/8	136/136	100%	0/4	49/50	0%
prostate	7/8	136/136	100%	0/6	45/48	0%
lung	7/8	136/136	100%	1/4	47/50	25%
colorectal	7/8	136/136	100%	4/4	49/50	80%
lymphoma	16/16	128/128	100%	5/6	48/48	100%
bladder	7/8	136/136	100%	0/3	49/51	0%
melanoma	5/8	136/136	100%	0/2	50/52	0%
uterus_adeno	7/8	136/136	100%	1/2	49/52	25%
leukemia	23/24	120/120	100%	4/6	47/48	80%
renal	7/8	136/136	100%	0/3	48/51	0%
pancreas	7/8	136/136	100%	0/3	45/51	0%
ovary	7/8	136/136	100%	0/4	47/50	0%
mesothelioma	7/8	136/136	100%	3/3	51/51	100%
CNS	16/16	128/128	100%	3/4	50/50	100%

The table presents measured covering properties both on the training set and on the test set. Although the obtained covering values on the training sets are very good, the measured prediction quality on the test sets is for many classes very low, significantly lower than those reported in [11]. For 7 out of 14 classes the measured precision on the test sets is 0%. But from the table an interesting and important relationship between prediction results on the test set and the number of target class examples in the training set can be noticed. There are very large differences among the results on the test sets for various classes (diseases) and the precision higher than 50% has been obtained for only 5 out of 14 classes. There are only three classes (lymphoma, leukemia, and CNS) with more than 8 training cases and all of them are among those with high precision on the test set, while for only two out of eleven classes with 8 training cases (colorectal and mesothelioma) high precision was achieved. The classification properties of rules induced for classes with 16 and 24 target class examples (lymphoma, leukemia and CNS) are comparable to those reported in [11], while the results on eight small example sets with 8 target examples were poor.

An obvious conclusion is that the use of the subgroup discovery algorithm is not appropriate for problems with a very small number of examples because overfitting can not be avoided in spite of the heuristics used in the SD algorithm and the additional domain-specific techniques used to restrict the hypothesis search space. But for larger training sets the subgroup discovery methodology enabled effective construction of relevant knowledge. The result, illustrated in Figure 1, demonstrates that mean values of rule sensitivity and precision are significantly higher for three tasks with 16 and 24 target class examples than for eleven tasks with only 8 target class examples. The mean values for the specificity are also higher but they were over 95% already for small target class sets. The induced rules for lymphoma, leukemia and CNS were evaluated by a domain expert and most of features used in them were recognized as known disease markers for the target class cancers [6]. Expert evaluation, which is out of scope of this work, proved the relevancy of induced rules. Both good prediction results on an independent test set as well as expert interpretation of induced rules prove the effectiveness of described methods for avoiding overfitting in scientific discovery tasks. Mostly bad results for tasks with only 8 target class examples demonstrate that the methods can not be successful in all situations, especially those with a very small number of examples.

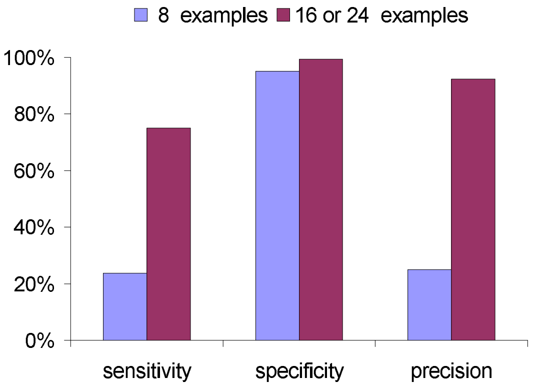

Figure 1. Mean values of sensitivity, specificity, and precision measured on the independent test set versus the number of target class cases in the training set.

Figure 2 presents the summary of results obtained by different experiments in eliminating irrelevant features. The experiments started with three domains with 16 and 24 target class examples for which successful induction of descriptive rules was possible. For these tasks both concepts of absolute and relative relevancy were very effective in reducing the number of features. About 60% of all features were detected as absolutely irrelevant while relative irrelevancy was even more effective as it managed to eliminate up to 75% of all the features. Their combination resulted in the elimination of 75% to 85% of all the features. These results are presented in the leftmost part of Figure 2. The set of all features in these experiments is generated so that for each attribute two features are constructed ($Att = A$ and $Att = P$) but so that totally irrelevant features (with $|TP| = 0$ or $|TN| = 0$) are eliminated.

Next, another domain with 16063 completely randomly generated attribute values was constructed. The same experiments were repeated as for the real gene expression domain. The results of experiments (repeated with 5 different randomly generated attribute sets) were significantly different: there have been only about 40% of absolutely irrelevant features and practically no relatively irrelevant features. The average results are presented in the second part of Figure 2. Comparing the results for the real and for the randomly generated domain, especially large differences can be noticed in the performance of relative relevancy. It is the consequence of the fact that in the real domain there are some features that are really relevant. They cover many target class examples and a few non target class examples and in this way they make many other features relatively irrelevant. The results prove the importance of relative relevancy for domains in which strong and relevant dependencies between classes and attribute values exist.

The experiments with feature relevancy continued with two domains with 32126 attributes. The first was completely random while the second was the combination of two previous domains with 16063 attributes, one of them being the real and the other the randomly generated. The results for the first domain are presented in the third part of Figure 2. The results were expected in the sense that they repeated the results for the domain with 16063 random attributes. It means that both absolute and relative relevancy do not become more effective when the number of random attributes increases. In this respect the results for the second domain presented in the rightmost part of Figure 2 are more important. After the elimination of absolutely irrelevant features the number of features is equal to the sum of features that remained in the independent domains with 16063 attributes. In contrast, relative relevancy was much more effective. Besides eliminating many features from the real attribute part it was now possible to eliminate also a significant part (more than 50%) of features constructed from randomly generated attributes.

From the previous analysis it is obvious that the elimination of features is the most effective for real domains. The same result was confirmed in experiments with domains with only 8 target class examples. In contrast, the approach is not effective for randomly generated domains. But it is important that for domains which are combinations of real and random attributes the methodology is effective. So it was possible that for three tasks with 16 and 24 target class examples there remained less features when there were 32126 attributes, including 16063 randomly generated, than when there were only 16063 random attributes. This proves that the presented methodology, especially relative irrelevancy, can be very useful in reducing the hypothesis search space by eliminating non-significant dependencies between attribute values and classes. This property is important because it may be assumed that among 16063 real attributes there are many of them which are also completely irrelevant with respect to the target class.

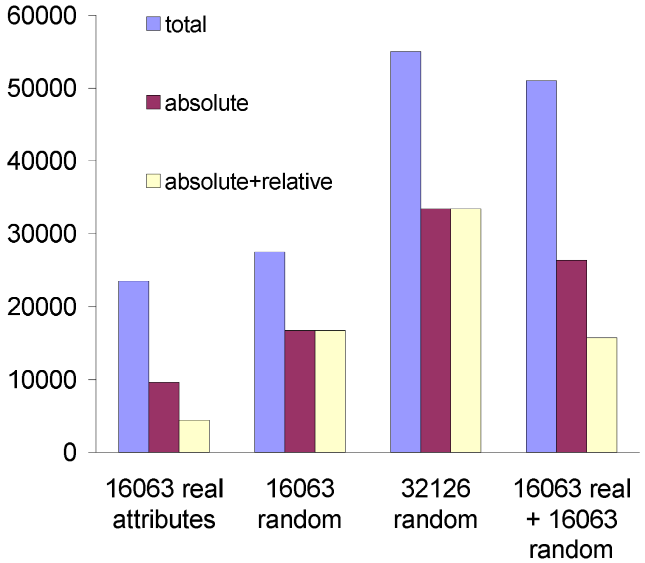

Figure 2. Mean numbers of features for three domains (lymphoma, leukemia, and CNS) after the elimination of totally irrelevant attributes (total), after the elimination of absolutely irrelevant features (absolute), and after the elimination of absolutely and relatively irrelevant features (absolute + relative). These three values are shown for the following training sets: real training set with 16063 attributes of gene expression activity values, a randomly generated set with 16063 attributes, a randomly generated set with 32126 attributes, and a set which is a combination of 16063 real and 16063 random attributes. The set of *all* features is generated so that for each attribute two features are constructed ($Att = A$ and $Att = P$).

Conclusions

The work confirms the known fact that restrictions of the hypothesis search space may help in avoiding overfitting of the training set. We have implemented both domain dependent restrictions by using discrete instead of continuous attribute values, and domain independent restrictions by the elimination of irrelevant features and rules. Interpretation of marginal gene values as a 'do not know state' helps also in reducing the feature space but more importantly it ensures robustness of the induced rules. Subgroup discovery proved to be a useful framework for the implementation of different relevancy conditions and an appropriate tool for descriptive induction.

Although we have tried to strictly realize the concept of restricted hypothesis space with the intention to prevent data overfitting, the results show that data overfitting in inductive learning of short rules can not be completely avoided, especially for domains and target classes with a very small number of samples. The very good news is that the obtained prediction quality of the induced rules grows very fast with the size of the training set. The results demonstrate that in the domain with more than 16000 attributes already for target classes with 16 or 24 examples and the total number of 144 examples it was possible to detect potentially new and relevant knowledge in form of dependencies between gene expression values and disease classes. This result may be interesting also for other scientific discovery applications.

REFERENCES

[1] Breiman, L. (2001) Random forests. *Machine Learning*, 45, 5–32.
[2] Domingos, P. (1999) The role of Occam's razor in knowledge discovery. *Data Mining and Knowledge Discovery*, 3: 409–425.
[3] Dudoit, S., Fridlyand J. & Speed T. (2000) Comparison of discrimination methods for the classification of tumors using gene expression data. Tech Report 576, University of California, Berkeley http://stat-www.berkeley.edu/ sandrine/tecrep/576.pdf
[4] Fürnkranz J. (1999) Separate-and-conquer rule learning. *Artificial Intelligence Review*, 13: 3–54.
[5] Gamberger, D. & Lavrač, N. (2002) Expert-guided subgroup discovery: Methodology and application. *Journal of Artificial Intelligence Research*, 17: 501–527.
[6] Gamberger, D., Lavrač, N., Železný, F. & Tolar, J. (2004) Induction of comprehensible models for gene expression datasets by subgroup discovery methodology. *Journal of Bioinformatics* (to appear).
[7] Golub, T.R. et al. (1999) Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 286: 531–537.
[8] Langley, P. & Sage, S. (1997) Scaling to domains with irrelevant features. In Petsche, T. and Greiner, S., editors *Computational Learning Theory and Natural Learning Systems*, Vol IV, MIT Press.
[9] Lavrač, N., Gamberger, D. & Turney, P. (1997) A relevancy filter for constructive induction. *IEEE Intelligent Systems & Their Applications*, 13: 50–56.
[10] Li, J. & Wong, L. (2002) Geography of differences between two classes of data. In *Proc. of 6th European Conference on Principles of Data Mining and Knowledge Discovery (PKDD2002)*, 325–337, Springer.
[11] Ramaswamy, S. et al. (2001) Multiclass cancer diagnosis using tumor gene expression signitures. In *Proc. Natl. Acad. Sci USA*, 98(26): 15149–15154.
[12] Schaffer, C. (1993) Overfitting avoidance as bias. *Machine Learning*, 10:153–178.