# Chapter XIII
# Gene Expression Mining Guided by Background Knowledge

**Jiří Kléma**
*Czech Technical University in Prague, Czech Republic*

**Filip Železný**
*Czech Technical University in Prague, Czech Republic*

**Igor Trajkovski**
*Jozef Stefan Institute, Slovenia*

**Filip Karel**
*Czech Technical University in Prague, Czech Republic*

**Bruno Cremilleux**
*Universite de Caen, France*

**Jakub Tolar**
*University of Minnesota, USA*

## ABSTRACT

*This chapter points out the role of genomic background knowledge in gene expression data mining. The authors demonstrate its application in several tasks such as relational descriptive analysis, constraint-based knowledge discovery, feature selection and construction or quantitative association rule mining. The chapter also accentuates diversity of background knowledge. In genomics, it can be stored in formats such as free texts, ontologies, pathways, links among biological entities, and many others. The authors hope that understanding of automated integration of heterogeneous data sources helps researchers to reach compact and transparent as well as biologically valid and plausible results of their gene-expression data analysis.*

## INTRODUCTION

High-throughput technologies like microrarrays or SAGE are at the center of a revolution in biotechnology, allowing researchers to simultaneously monitor the expression of tens of thousands of genes. However, gene-expression data analysis represents a difficult task as the data usually show an inconveniently low ratio of samples (biological situations) against variables (genes). Datasets are often noisy and they contain a great part of variables irrelevant in the context under consideration. Independent of the platform and the analysis methods used, the result of a gene-expression experiment should be driven, annotated or at least verified against genomic background knowledge (BK).

As an example, let us consider a list of genes found to be differentially expressed in different types of tissues. A common challenge faced by the researchers is to translate such gene lists into a better understanding of the underlying biological phenomena. Manual or semi-automated analysis of large-scale biological data sets typically requires biological experts with vast knowledge of many genes, to decipher the known biology accounting for genes with correlated experimental patterns. The goal is to identify the relevant "functions", or the global cellular activities, at work in the experiment. Experts routinely scan gene expression clusters to see if any of the clusters are explained by a known biological function. Efficient interpretation of this data is challenging because the number and diversity of genes exceed the ability of any single researcher to track the complex relationships hidden in the data sets. However, much of the information relevant to the data is contained in publicly available gene ontologies and annotations. Including this additional data as a direct knowledge source for any algorithmic strategy may greatly facilitate the analysis.

This chapter gives a summary of our recent experience in mining of transcriptomic data. The chapter accentuates the potential of genomic background knowledge stored in various formats such as free texts, ontologies, pathways, links among biological entities, etc. It shows the ways in which heterogeneous background knowledge can be preprocessed and subsequently applied to improve various learning and data mining techniques. In particular, the chapter demonstrates an application of background knowledge in the following tasks:

- Relational descriptive analysis
- Constraint-based knowledge discovery
- Feature selection and construction (and its impact on classification accuracy)
- Quantitative association rule mining

The chapter starts with an overview of genomic datasets and accompanying background knowledge analyzed in the text. Section on relational descriptive analysis presents a method to identify groups of differentially expressed genes that have functional similarity in background knowledge. Section on genomic classification focuses on methods helping to increase accuracy and understandability of classifiers by incorporation of background knowledge into the learning process. Section on constraint-based knowledge discovery presents and discusses several background knowledge representations enabling effective mining of meaningful over-expression patterns representing intrinsic associations among genes and biological situations. Section on association rule mining briefly introduces a quantitative algorithm suitable for real-valued expression data and demonstrates utilization of background knowledge for pruning of its output ruleset. Conclusion summarizes the chapter content and gives our future plans in further integration of the presented techniques.

# GENE-EXPRESSION DATASETS AND BACKGROUND KNOWLEDGE

The following paragraphs give a brief overview of information resources used in the chapter. The primary role of background knowledge is to functionally describe individual genes and to quantify their similarity.

## Gene-Expression (Transcriptome) Datasets

The process of transcribing a gene's DNA sequence into the RNA that serves as a template for protein production is known as gene expression. A gene's expression level indicates an approximate number of copies of the gene's RNA produced in a cell. This is considered to be correlated with the amount of corresponding protein made.

**Expression chips (DNA chips, microarrays)**, manufactured using technologies derived from computer-chip production, can now measure the expression of thousands of genes simultaneously, under different conditions. A typical gene expression data set is a matrix, with each column representing a gene and each row representing a class labeled sample, e.g. a patient diagnosed having a specific sort of cancer. The value at each position in the matrix represents the expression of a gene for the given sample (see Figure 1). The particular problem used as an example in this chapter aims at distinguishing between acute lymphoblastic leukemia (ALL) and acute myeloid leukemia (AML) (Golub, 1999). The gene expression profiles were obtained by the Affymetrix HU6800 microarray chip, containing probes for 7129 genes, the data contains 72 class-labeled samples of expression vectors. 47 samples belong to the ALL class (65%) as opposed to 25 samples annotated as AML (35%).

**SAGE (Serial Analysis of Gene Expression)** is another technique that aims to measure the expression levels of genes in a cell population (Velculescu, 1995). It is performed by sequencing tags (short sequences of 14 to 21 base pairs (bps) which are specific of each mRNA). A SAGE library is a list of transcripts expressed at one given time point in one given biological situation. Both the identity (assessed through a tag-to-gene complex process, (Keime, 2004)) and the amount of each transcript is recorded. SAGE, as a data source, has been largely under-exploited as of today, in spite of its important advantage over microarrays. In fact, SAGE can produce datasets that can be directly compared between libraries without the need for external normalization. The human transcriptome can be seen as a set of libraries that would ideally be collected in each biologically relevant situation in the human body. This is clearly out of reach at the moment, and we deal in the present work with 207 different situations ranging from embryonic stem cells to foreskin primary fibroblast cells. Unambiguous tags (those that enable unequivocal gene identification) were selected leaving a set of 11082 tags/genes. A 207x11082 gene expression matrix was built.

*Figure 1. The outcome of a microarray or SAGE experiment*

| | gene 1 | gene 2 | ... | gene n | target |
|---|---|---|---|---|---|
| sample/situation 1 | Values of gene expression (binary, symbolic, integer or real)<br>Sample expression signatures in rows, gene expression profiles in columns | | | | $T_1$ |
| sample/situation 2 | | | | | $T_2$ |
| ... | | | | | ... |
| sample/situation m | | | | | $T_m$ |

The biological situations embody various tissues (brain, prostate, breast, kidney or heart) stricken by various possible diseases (mainly cancer, but also HIV and healthy tissues). As the main observed disorder is carcinoma, a target binary attribute Cancer was introduced by the domain expert. The class value is 0 for all the healthy tissues and also the tissues suffering by other diseases than cancer (77 situations in total, 37.2%). It is equal to 1 for all the cancerous tissues (130 situations, 62.8%). The dataset was also binarized to encode the over-expression of each gene using the MidRange method described in (Becquet, 2002). For each gene it takes its highest value (max), the lowest value (min), and calculates the mid-range as (max-min)/2. Values above the threshold are given a boolean value of 1; all others are given a value of 0.

## Background Knowledge

In this chapter, the term genomic background knowledge refers to any information that is not available in a gene-expression dataset but it is related to the genes or situations contained in this dataset. The richest body of background knowledge is available for genes. Gene datatabases such as Entrez Gene (NCBI website: http://www.ncbi.nlm.nih.gov/) offer a large scale of gene data – general information including a short textual summary of gene function, cellular location, bibliography, interactions and further links with other genes, memberships in pathways, referential sequences and many other pieces of information. Having a list of genes (i.e. colums in Figure 1), the information about all of the genes from the list can be collected automatically via services such as Entrez Utils (NCBI website: http://www.ncbi.nlm.nih.gov/). Similarly, annotations of the biological samples (i.e. rows in Figure 1) contained in the gene-expression dataset are available. In the simplest case, there is at least a brief description of the aim of the experiment where the sample was used.

Two forms of external knowledge require special attention during data pre-processing. These are freetexts and gene ontologies (GOs). We use them in two principal ways. The first way of utilization extracts all the relevant keywords for each gene, the main purpose is to **annotate**. In the second way we aim to **link** the genes. We introduce a quantitative notion of gene similarity that later on contributes to the cost-efficient reduction of computational costs across various learning and data mining algorithms. In the area of freetexts we have been inspired mainly by (Chaussabel, 2002; Glenisson, 2003). Both of them deal with the term-frequency vector representation which is a simple however prevailing representation of texts. This representation allows for an annotation of a gene group as well as a straightforward definition of gene similarity. In the area of gene ontologies we mostly rely on (Martin, 2004), the gene similarity results from the genes' positions in the molecular functional, biological process or cellular component ontology.

However, alternative sources can also be used, e.g., (Sevon, 2006) suggests an approach to discover links between entities in biological databases. Information extracted from available databases is represented as a graph, where vertices correspond to entities and edges represent annotated relationships among entities. A link is manifested as a path or a sub-graph connecting the corresponding vertices. Link goodness is based on edge reliability, relevance and rarity. Obviously, the graph itself or a corresponding similarity matrix based on the link goodness can serve as an external knowledge source.

### Free Texts and Their Preprocessing

To access the gene annotation data for every gene or tag considered, probe identifiers (in the case microarrays) or Reference Sequence (RefSeq) identifiers (for SAGE) were translated into Entrez Gene

Identifiers (Entrez Ids) using the web-tool MatchMiner (http://discover.nci.nih.gov/ matchminer/). The mapping approached 1 to 1 relationship. Knowing the gene identifiers, the annotations were automatically accessed through hypertext queries to the EntrezGene database and sequentially parsed (Klema, 2006). Non-trivial textual records were obtained for the majority of the total amount of unique ids. The gene textual annotations were converted into the vector space model. A single gene corresponds to a single vector, whose components correspond to the frequency of a single vocabulary term in the text. This representation is often referred to as bag-of-words (Salton, 1988). The particular vocabulary consisted of all stemmed terms (Porter stemmer, http://www.tartarus.org/~martin/ PorterStemmer/) that appear in 5 different gene records at least. The most frequent terms were manually checked and insufficiently precise terms (such as gene, protein, human etc.) were removed. The resulting vocabulary consisted of 17122 (ALL/AML), respectively 19373 terms (SAGE). The similarity between genes was defined as the cosine of the angle between the corresponding term-frequency inverse-document-frequency (TFIDF) (Salton, 1988) vectors. The TFIDF representation statistically considers how important a term is to a gene record.

A similarity matrix *s* for all the genes was generated (see Figure 2). Each field of the triangular matrix $s_{ij}\in\langle 0,1\rangle$ gives a similarity measure between the genes *i* and *j*. The underlying idea is that a high value of two vectors' cosine (which means a low angle among two vectors and thus a similar occurrence of the terms) indicates a semantic connection between the corresponding gene records and consequently their presumable connection. This model is known to generate false positive relations (as it does not consider context) as well as false negative relations (mainly because of synonyms). Despite this inaccuracy, bag-of-words format corresponds to the commonly used representation of text documents. It enables efficient execution of algorithms such as clustering, learning, classification or visualization, often with surprisingly faithful results (Scheffer, 2002).

## Gene Ontology

One of the most important tools for the representation and processing of information about gene products and functions is the Gene Ontology (GO). It provides a controlled vocabulary of terms for the description of cellular components, molecular functions, and biological processes. The ontology also identifies those pairs of terms where one **is a** special case of the other. Similarly, term pairs are identified where

*Figure 2. Gene similarity matrix – the similarity values lie in a range from 0 (total mismatch between gene descriptions) to 1 (pertfect match), n/a value suggests that at least one of the gene tuple has no knowledge attached*

| | gene 1 | gene 2 | gene 3 | gene 4 | ... | gene n |
|---|---|---|---|---|---|---|
| gene 1 | 1 | 0.05 | n/a | n/a | ... | 0.63 |
| gene 2 | | 1 | 0.01 | 0.33 | ... | 0.12 |
| gene 3 | | | 1 | n/a | ... | n/a |
| gene 4 | | | | 1 | ... | n/a |
| ... | ... | ... | ... | ... | ... | ... |
| gene n | | | | | | 1 |

one term refers to a **part of** the other. Formally this knowledge is reflected by the binary relations "**is a"** and "**part of**".

For each gene we extracted its ontological annotation, that is, the set of ontology terms relevant to the gene. This information was transformed into the gene's background knowledge encoded in relational logic in the form of Prolog facts. For example, part of the knowledge for particular gene SRC, whose EntrezId is 6714, is as follows:

```
function(6714,'ATP binding').
function(6714,'receptor activity').
process(6714,'signal complex formation').
process(6714,'protein kinase cascade').
component(6714,'integral to membrane').
...
```

Next, using GO, in the gene's background knowledge we also included the gene's generalized annotations in the sense of the "**is a**" relation described above. For example, if one gene is functionally annotated as: "zinc ion binding", in the background knowledge we also included its more general functional annotations such as e.g. transition metal ion binding or metal ion binding.

The genes can also be functionally related on the basis of their GO terms. Intuitively, the more GO terms the genes share, and the more specific the terms are, the more likely the genes are to be functionally related. (Martin, 2004) defines a distance based on the Czekanowski-Dice formula, the methodology is implemented within the GOProxy tool of GOToolBox http://crfb.univ-mrs.fr/GOToolBox/). A similarity matrix of the same structure as shown in Figure 2 can be generated.

### Gene Interactions

The similarity matrix described in the previous paragraphs is one specific way to represent putative gene interactions. Besides, public databases also offer the information about pairs of genes for which there is traced experimental evidence of mutual interaction. In this case we use a crisp declarative representation of the interaction, in the form of a Prolog fact. The following example represents an interaction between gene SRC (EntrezId 6714) and genes ADRB3 (EntrezId 155) and E2F4 (EntrezId 1874):

```
interaction(6714,155).
interaction(6714,1874).
```

## RELATIONAL DESCRIPTIVE ANALYSIS

This section presents a method to identify groups of differentially expressed genes that have functional similarity in background knowledge formally represented by gene annotation terms from the gene ontology (Trajkovski, 2006). The input to the algorithm is a multidimensional numerical data set, representing the expression of the genes under different conditions (that define the classes of examples), and an ontology used for producing background knowledge about these genes. The output is a set of gene groups whose expression is significantly different for one class compared to the other classes. The distinguishing property of the method is that the discovered gene groups are described in a rich, yet human-readable language. Specifically, each such group is defined in terms of a logical conjunction of features, that each member of the group possesses. The features are again logical statements that describe gene properties using gene ontology terms and interactions with other genes.

Medical experts are usually not satisfied with a separate description of every important gene, but want to know the processes that are controlled by these genes. The presented algorithm enables to find these processes and the cellular components where they are "executed", indicating the genes from the pre-selected list of differentially expressed genes which are included in these processes.

These goals are achieved by using the methodology of Relational Subgroup Discovery (RSD) (Lavrac, 2002). RSD is able to induce sets of rules characterizing the differentially expressed genes in terms of functional knowledge extracted from the gene ontology and information about gene interactions.

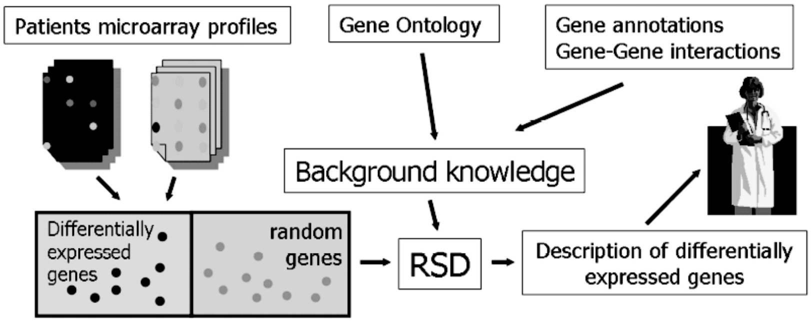
## Fundamental Idea

The fundamental idea of learning relational descriptions of differentially expressed gene groups is outlined in Figure 3 (Trajkovski 2008). First, a set of differentially expressed genes, $G_C(c)$, is constructed for every class $c \in C$ (e.g. types of cancer). These sets can be constructed in several ways. For example: $G_C(c)$ can be the set of k ($k > 0$) most correlated genes with class c, for instance computed by Pearson's correlation. $G_C(c)$ can also be the set of best k single gene predictors, using the recall values from a microarray/SAGE experiment (absent/present/marginal) as the expression value of the gene. These predictors can acquire the form such as:

If $gene_i$ = present Then class = c

In our experiments, $G_C(c)$ was constructed using a modified version of the t-test statistics. The modification lies in an additional condition ensuring that each selected gene has at least twofold difference in its average expression for the given class with respect to the rest of the samples. The second step aims at improving the interpretability of $G_C$. Informally, we do this by identifying subgroups of genes in $G_C(c)$ (for each $c \in C$) which can be summarized in a compact way. Put differently, for each $c_i \in C$ we search for

*Figure 3. An outline of the process of gene-expression data analysis using RSD*


Patients microarray profiles
Gene Ontology
Gene annotations
Gene-Gene interactions
Background knowledge
Differentially expressed genes
random genes
RSD
Description of differentially expressed genes

*(© 2008 IEEE Transactions on Systems, Man, and Cybernetics: Part C. Used with permission.)*

compact descriptions of gene subgroups with expression strongly correlating (positively or negatively) with $c_i$ and weakly with all $c_j \in C$; $j \neq i$.

Searching for these groups of genes, together with their description, is defined as a supervised machine learning task. We refer to it as the secondary mining task, as it aims to mine from the outputs of the primary learning process in which differentially expressed genes are searched. This secondary task is, in a way, orthogonal to the primary discovery process in that the original attributes (genes) now become training examples, each of which has a class label "differentially expressed" and "not differentially expressed". Using the gene ontology information, gene annotation and gene interaction data, we produce background knowledge for differentially expressed genes on one hand, and randomly chosen genes on the other hand. The background knowledge is represented in the form of Prolog facts. Next, the RSD algorithm finds characteristic descriptions of the differentially expressed genes. Finally, the discovered descriptions can be straightforwardly interpreted and exploited by medical experts.

## Relational Subgroup Discovery

The RSD algorithm proceeds in two steps. First, it constructs a set of relational features in the form of first-order logic atom conjunctions. The entire set of features is then viewed as an attribute set, where an attribute has the value true for a gene (example) if the gene has the feature corresponding to the attribute. As a result, by means of relational feature construction we achieve the conversion of relational data into attribute-value descriptions. In the second step, interesting gene subgroups are searched, such that each subgroup is represented as a conjunction of selected features. The subgroup discovery algorithm employed in this second step is an adaptation of the popular propositional rule learning algorithm CN2 (Clark, 1989).

The feature construction component of RSD aims at generating a set of relational features in the form of relational logic atom conjunctions. For example, the informal feature *"gene g interacts with another gene whose functions include protein binding"* has the relational logic form:

interaction(g,B), function(B,'protein binding')

where upper cases denote variables, and a comma between two logical literals denotes a conjunction. The user specifies mode declarations which syntactically constrain the resulting set of constructed features and restrict the feature search space. Furthermore, the maximum length of a feature (number of contained literals) is declared. RSD proceeds to produce an exhaustive set of features satisfying the declarations. Technically, this is implemented as an exhaustive depth-first backtrack search in the space of all feature descriptions, equipped with certain pruning mechanisms. Finally, to evaluate the truth value of each feature for each example for generating the attribute-value representation of the relational data, the first-order logic resolution procedure is used, provided by a standard Prolog language interpreter.

Subgroup discovery aims at finding population subgroups that are statistically "most interesting", e.g., are as large as possible and have the most unusual statistical characteristics with respect to the target class. To discover interesting subgroups of genes defined in terms of the constructed features, RSD follows a strategy stemming from the popular rule learner CN2. See (Zelezny, 2006) for details on this procedure.

## Experiments

In ALL, RSD has identified a group of 23 genes, described as a conjunction of two features: component(G,'nucleus') AND interaction(G,B),process(B,'regulation of transcription, DNA-dependent'). The products of these genes, proteins, are located in the nucleus of the cell, and they interact with genes that are included in the process of regulation of transcription. In AML, RSD has identified several groups of overexpressed genes, located in the membrane, that interact with genes that have 'metal ion transport' as one of their function.

Subtypes of ALL and AML can also be distinguished, in a separate subgroup discovery process where classes are redefined to correspond to the respective disease subtypes. For example, two subgroups were found with unusually high frequency of the BCR (TEL, respectively) subtype of ALL. The natural language description of BCR class derived from the automatically constructed subgroup relational description is the following: *genes coding for proteins located in the integral to membrane cell component, whose functions include receptor activity*. This description indeed appears plausible, since BCR is a classic example of a leukemia driven by spurious expression of a fusion protein expressed as a continuously active kinase protein on the *membrane* of leukemic cells. Similarly, the natural language description for the TEL class is: genes coding for proteins located in the nucleus whose functions include protein binding and whose related processes include transcription. Here again, by contrast to BCR, the TEL leukemia is driven by expression of a protein, which is a transcription factor active in the *nucleus*.

A statistical validation of the proposed methodology for discovering descriptions of differentially expressed gene groups was also carried out. The analysis determined if the high descriptive capacity pertaining to the incorporation of the expressive relational logic language incurs a risk of descriptive overfitting, i.e., a risk of discovering subgroups whose bias toward differential expression is only due to chance. The discrepancy of the quality of discovered subgroups on the training data set on one hand and an independent test set on the other hand was measured. It was done through the standard 10-fold stratified cross-validation regime. The specific qualities measured for each set of subgroups produced for a given class are average precision (PRE), recall (REC) and area under ROC (AUC) values among all subgroups in the subgroup set. In ALL/AML dataset, RSD showed PRE 100(±0)%, REC 16% and AUC 65% in training data and PRE 85(±6)%, REC 13% and AUC 60% in independent testing data. The results demonstrate an acceptable decay from the training to the testing set in terms of both PRE and REC, suggesting that the discovered subgroup descriptions indeed capture the relevant gene properties. In terms of total coverage, in average, RSD covered more then 2/3 of the preselected differentially expressed genes, while 1/3 of the preselected genes were not included in any group. A possible interpretation is that they are not functionally connected with the other genes and their initial selection through the t-test was due to chance. This information can evidently be back-translated into the gene selection procedure and used as a gene selection heuristic.

## GENOMIC CLASSIFICATION WITH BACKGROUND KNOWLEDGE

Traditional attribute-value classification searches for a mapping from attribute value tuples, which characterize instances, to a discrete set whose elements correspond to classes. When dealing with a large number of attributes and a small number of instances, the resulting classifier is likely to fit the training data solely by chance, rather than by capturing genuine underlying trends. Datasets are often noisy and they contain a great part of variables irrelevant in the context of desired classification.

In order to increase the predictive power of the classifier and its understandability, it is advisable to incorporate background knowledge into the learning process. In this section we study and test several simple ways to improve a genomic classifier constructed from gene expression data as well as textual and gene ontology annotations available both for the genes and the biological situations.

## Motivation

Decision-tree learners, rule-based classifiers or neural networks are known to often overfit gene expression data, i.e., identify many false connections. A principal means to combat the risk of overfitting is feature selection (FS); a process aiming to filter irrelevant variables (genes) from the dataset prior to the actual construction of a classifier. Families of classifiers are available, that are more tolerant to abundance of irrelevant attributes than the above mentioned traditional methods. Random forests (Breiman, 2001; Diaz-Uriarte, 2006) or support vector machines (Furey, 2000; Lee, 2003) exemplify the most popular ones. Still, feature selection remains helpful in most gene expression classification analyses. Survey studies (such as (Lee, 2005)) stress that the choice of feature selection methods has much effect on the performance of the subsequently applied classification methods.

In the gene expression domain, feature selection corresponds to the task of finding a limited set of genes that still contains most of the information relevant to the biological situations in question. Many gene selection approaches create rankings of gene relevance regardless of any knowledge of the classification algorithm to be used. These approaches are referred to as filter methods. Besides general filter ranking methods (different modifications of the t-test, information gain, mutual information), various specific gene-selection methods were published. The signal-to-noise (S2N) ratio was introduced in (Golub, 1999), significance analysis of microarrays (SAM) appeared in (Tusher, 2001). (Tibshirani, 2002) proposed and tested nearest shrunken centroids (NSC). The wrapper methods can be viewed as gene selection methods which directly employ classifiers. Gene selection is then guided by analyzing the embedded classifier's performance as well as its result (e.g. to detect which variables proved important for classification). Recursive Feature Elimination (RFE) based on absolute magnitude of the hyperplane elements in a support vector machine is discussed in (Guyon, 2002). (Uriarte, 2006) selects genes according to the decrease of the random forest classification accuracy when values of the gene are permuted randomly.

Here we consider feature selection techniques in a different perspective. All of the above-mentioned methods rely on gene expression data itself. No matter whether they apply a single-variate or multivariate selection criteria, they disregard any potential prior knowledge on gene functions and its true or potential interactions with other genes, diseases or other biological entities. Our principal aim is to exploit background knowledge such as literature and ontologies concerning genes or biological situations as a form of evidence of the genes' relevance to the classification task.

The presented framework uses the well-known CN2 (Clark, 1989) rule learning algorithm. In fact, rule-based classification exhibits a particular weakness when it comes to gene expression data classification. This is due to their small resistance to overfitting, as commented above. As such, a rule learning algorithm is a perfect candidate to evaluate the possible assets of background knowledge. Thus, the main goal is not to develop the best possible classifier in terms of absolute accuracy. Rather, we aim to assess the relative gains obtained by integrating prior knowledge. The evaluated gains pertain to classification accuracy, but also to the comprehensibility of the resulting models.

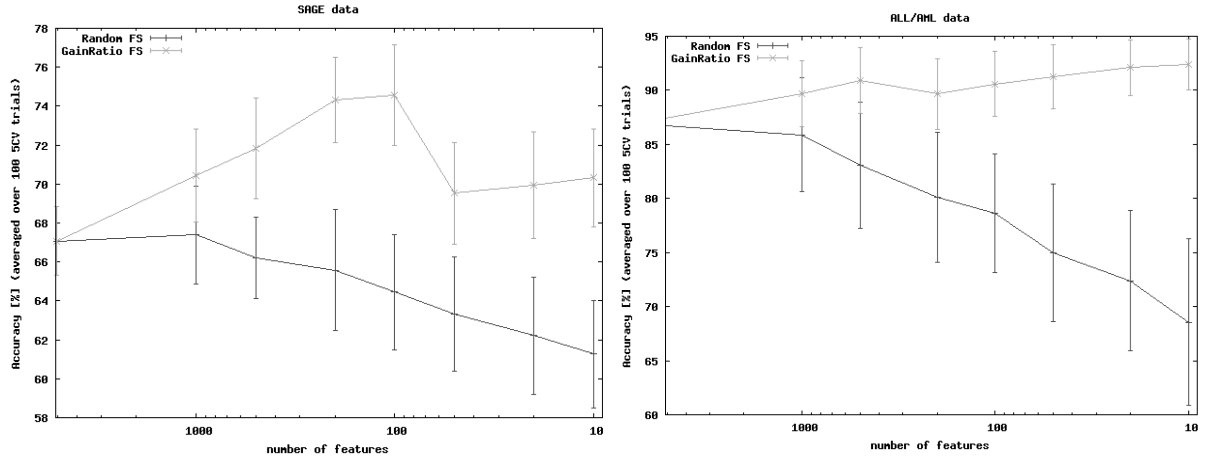
## Feature Selection

We first consider the widely accepted dogma that feature selection helps improve classification accuracy and test it in the gene expression domain. A single-variate gain ratio (GR) (Quinlan, 1986) evaluation criterion was used. The criterion is information-based and disregards apriori knowledge. The graphs in Figure 4 show that indeed: 1) FS improves classification accuracy (SAGE dataset – the average accuracy grows from 67.1% for 5052 features to 72.8% for 50 features, ALL/AML dataset – the average accuracy grows from 86.9% for 7129 features to 92.4% for 10 features), 2) informed FS outperforms the random one.

Next, we want to design a mechanism which could guide feature selection using available apriori knowledge. The main idea is to promote the genes whose description contains critical keywords relevant to the classification objective. For example, SAGE classification tries to distinguish among cancerous and non-cancerous tissues. Consequently, the genes that are known to be active in cancerous tissues may prove to be more important than they seem to according to their mutual information with the target (expressed in terms of entropy, gain ratio or mutual information itself). These genes should be promoted into the subset of selected features. Auxiliary experiments proved that there are large gene groups whose mutual information with the target differs only slightly. As a consequence, even an insignificant difference then may decide whether the gene gets selected. To avoid this threshold curse, one may favor multi-criteria gene ranking followed by gene filtering.

The way in which we rank genes with respect to their textual and/or ontological description depends on the amount of information available for biological situations. In the SAGE dataset, each situation contains a brief textual annotation. The frequent words from these annotations serve to create a list of relevant keywords. In the ALL/AML dataset, there are descriptions of the individual classes and the list of keywords is made of the words that characterize these classes. In order to calculate gene importance, the list of keywords is matched with the bag-of-words that characterizes the individual genes. A gene is rated higher if its description contains a higher proportion of situation keywords. Let us show the following simple example:

*Figure 4. Gain ratio - development of classification accuracy with decreasing number of features/ genes*



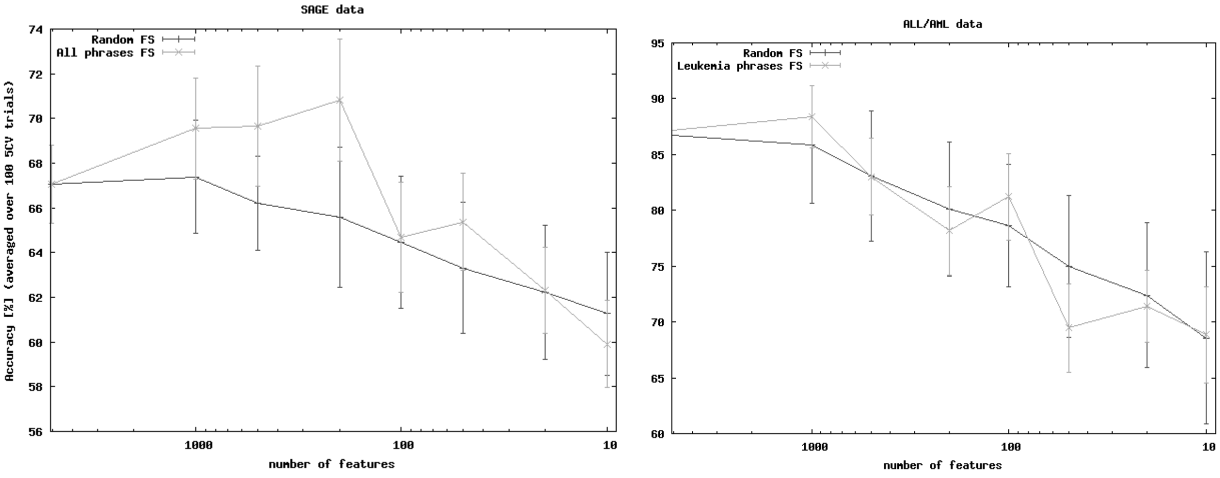
Keywords (characterize the domain): carcinoma, cancer, glioblastoma
Bag of words (characterize the gene): bioactive, cancer, framework, glioblastoma
gene1: 1, 3, 4, 2, gene2: 0, 0, 2, 0 (the word bioactive appears 3 times in gene1 annotations etc.)
gene1 scores (1+4)/(1+3+4+2)=0.5, gene2 scores 0/2=0

We refer to this process as the apriori-based FS. The graphs in Figure 5 compare the apriori-based FS with the random one. In the SAGE dataset, the list of apriori genes is better than random, although the margin is not as distinct as for the information-based criterion used in Figure 4. In the ALL/AML dataset, the apriori-based genes proved to have similar predictive power as randomly selected genes. A likely explanation for this is that the list of keywords was too short. The gene ranking was too rough to correlate with the real gene importance. A great portion of genes scored 0 as they never co-occur with any keyword.

We next tackle the question whether one can cross-fertilize the information-based and apriori-based FS. Two different FS procedures were implemented – conditioning and combination. Conditioning FS keeps the gain ratio ranking but removes all the genes scoring less than a threshold on the apriori-based ranking scale. When asked for X best genes, it takes the X top genes from the reduced list. Combination FS takes the best genes from top of both the lists. When asked for X best genes it takes X/2 top genes from the gain ratio list and X/2 top genes from the apriori list. The result is shown in Figure 6. In spite of better than random quality of apriori-based FS in SAGE dataset, neither conditioning nor combination outperforms gain ratio. The apriori list seems to bring no additional strong predictors. In the ALL/AML dataset, conditioning gives the best performance. It can be explained by the good informativeness of the set of 1000 top genes from the apriori list, which enriches the original gain-ratio list.

In general, the experiments proved that usability of apriori-based FS strongly depends on the domain and the target of classification. The amount of available keywords and their relevance make the crucial issue.

*Figure 5. Apriori-based feature selection - development of classification accuracy with decreasing number of features/genes*



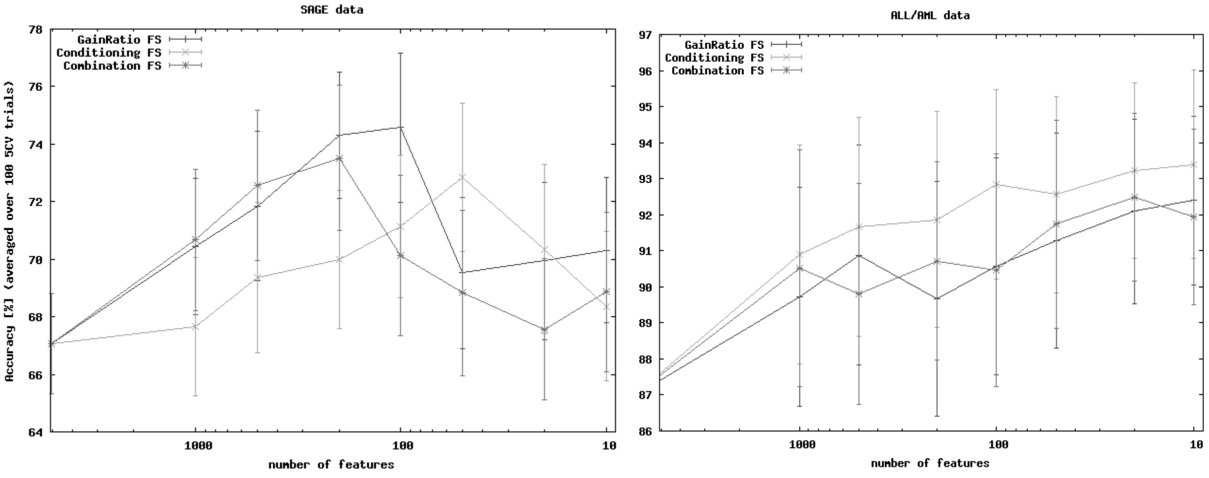
## Feature Extraction

The curse of feature space dimensionality can also be overcome or in the least reduced by feature extraction (FE). It is a procedure that transforms the original feature space by building new features from the existing ones. (Hanczar, 2003) proposed a prototype-based feature extraction that consists of two simple steps: 1) identify equivalence classes inside the feature space, 2) extract feature prototypes that represent the classes invented in step 1. In practice, the features are clustered and each cluster is represented by its mean vector – the prototype. The prototypes are used to learn a classifier and to classify new biological situations.

An interesting fact is that equivalence classes can be derived from the gene expression profiles as well as from the known gene functions or any other biologically relevant criteria. The gene similarity matrix based on gene-expression profiles can be combined with the gene-similarity matrices inferred from the background knowledge. Although the prototypes did not prove to increase classification accuracy either in the ALL/AML or the SAGE task, the prototypes can increase understandability of the resulting classifier. The classifier does not treat the individual genes but it reports the equivalence classes whose interpretability is higher as they are likely to contain "similar" genes.

Another idea is to inject background knowledge into the learning algorithm itself. In case of CN2, the algorithm implements a laplacian heuristic that drives rule construction. As mentioned earlier, the algorithm is likely to overfit the data as it searches a large feature space, verifies a large number of simple conditions and randomly finds a rule with a satisfactory heuristic value. Background knowledge can complement the laplacian criteria in the following way: 1) promote short rules containing genes with apriori relevance to the target (a kind of late feature *selection* conditioned by rule length and heuristic value), 2) promote the rules with interacting genes (a kind of late feature *extraction* with the same conditioning). This form of background knowledge injection was implemented and evaluated in (Trna, 2007). The main benefit of this method is the understandability of the resulting classifier.

*Figure 6. Combined feature selection - development of classification accuracy with decreasing number of features/genes*



## CONSTRAINT-BASED KNOWLEDGE DISCOVERY

Current gene co-expression analyses are often based on global approaches such as clustering or bi-clustering. An alternative way is to employ local methods and search for patterns – sets of genes displaying specific expression properties in a set of situations. The main bottleneck of this type of analysis is twofold – computational costs and an overwhelming number of candidate patterns which can hardly be further exploited by a human. A timely application of background knowledge can help to focus on the most plausible patterns only. This section discusses various representations of BK that enables the effective mining and representation of meaningful over-expression patterns representing intrinsic associations among genes and biological situations.

### Constraints Inferred from Background Knowledge

Details on knowledge discovery from local patterns are given in another chapter of this book (Cremilleux, 2008). This section focuses on processing, representation and utilization of BK within the constraint-based framework presented ibid. In the domain of constraint-based mining, the constraints should effectively link different datasets and knowledge types. For instance, in the domain of genomics, biologists are interested in constraints both on co-expression groups and common characteristics of the genes and/or biological situations concerned. Such constraints require to tackle transcriptome data (often provided in a transactional format) and external databases. This section provides examples of a declarative language enabling the user to set varied and meaningful constraints defined on transcriptome data, similarity matrices and textual resources.

In our framework, a constraint is a logical conjunction of propositions. A proposition is an arithmetic test such as $C > t$ where $t$ is a number and $C$ denotes a *primitive* or a *compound*. A primitive is one of a small set of predefined simple functions evaluated on the data. Such primitives may further be assembled into compounds.

We illustrate the construction of a constraint through an example. A textual dataset provides a description of genes. Each row contains a list of phrases that characterize the given gene. The phrases can be taken from gene ontology or they can represent frequent relevant keywords from gene bibliography:

Gene 1: ‘metal ion binding’ ‘transcription factor activity’ ‘zinc ion binding’
Gene 2: ‘hydrolase activity’ ‘serine esterase activity’ ‘cytoplasmic membrane-bound vesicle’
...
Gene n: ‘serine-type peptidase activity’ ‘proteolysis’ ‘signal peptide processing’

In reference to the textual data, *regexp(X,RE)* returns the items among *X* whose phrase matches the regular expression *RE*.

As concerns the similarity matrices, we deal with primitives such as *sumsim(X)* denoting the similarity sum over the set of items *X* or *insim(X,min,max)* for the number of item pairs whose similarity lies between *min* and *max*. As we may deal with a certain portion of items without any information, there are primitives that distinguish between *zero* similarity and *missing value* of similarity. The primitive *svsim(X)* gives the number of item pairs belonging to *X* whose mutual similarity is valid and *mvsim(X)* stands for its counterpart, i.e., the missing interactions when one of the items has an empty record within the given similarity representation. The primitives can make compounds. Among many others, *sumsim(X)/svsim(X)* makes the average similarity, *insim(X,thres,1)/svsim(X)* gives a proportion of the strong interactions (similarity higher than the threshold) within the set of items, *svsim(X)/(svsim(X)+mvsim(X))* can avoid patterns with prevailing items of an unknown function.

Relational and logical operators as well as other built in functions enable to create the final constraint, e.g., $C_1 \geq thres_1$ and $C_2 \neq thres_2$ where $C_i$ stands for an arbitrary compound or primitive. Constraints can also be simultaneously derived from different datasets. Then, the dataset makes another parameter of the primitive. For example, the constraint *length(regexp(X,'*ribosom*', TEXT))>1* returns all the patterns that contain at least 2 items involving "ribosom" in any of their characteristic phrases within the TEXT dataset.

## Internal and External Constraints to Reach a Meaningful Limited Pattern Set

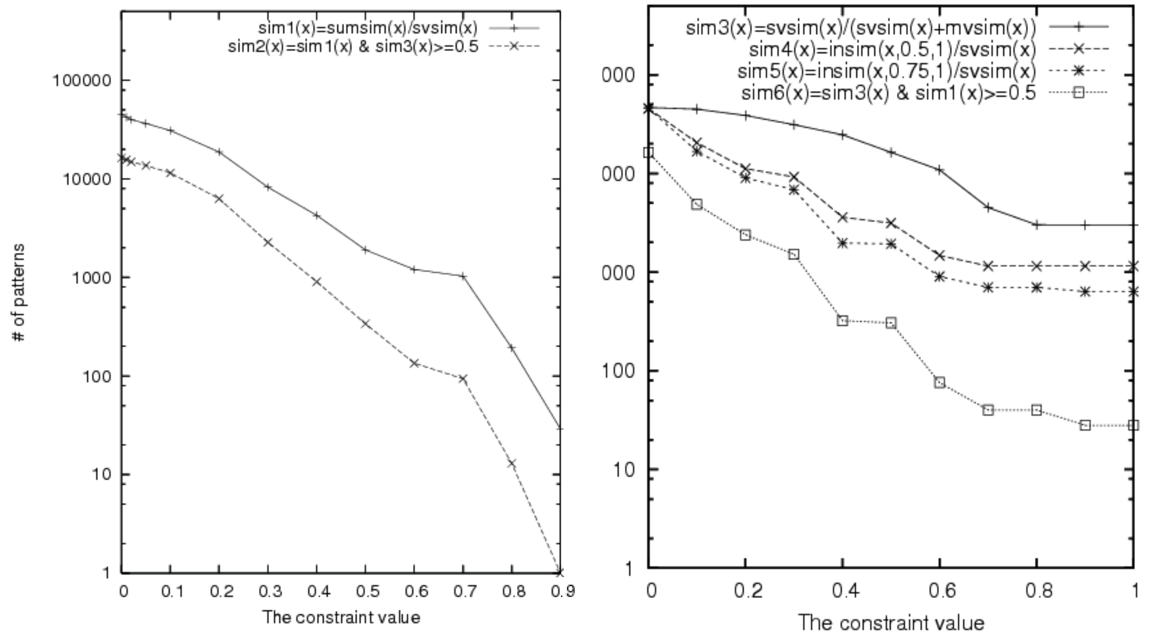
Traditional pattern mining deals with constraints that we refer to as internal. Truth values of such constraints are fully determined by the transcriptome dataset. The most meaningful internal constraints usually are the *area*, i.e. the product of the number of genes in the pattern (gene set), and the frequency of the pattern (number of transactions where the set is contained). This is because the main goal is usually to identify large gene sets that tend to co-occur frequently. For these constraints to apply, one must consider a *binarized* expression dataset enabling to state whether or not a gene is expressed in a given situation. Verifying the area constraints means checking whether the area is larger than a certain threshold.

However, the area constraint is not a panacea for distinction between meaningful patterns and spurious ones, i.e., the patterns occurring randomly. Indeed, the largest area patterns often tend to be trivial, bringing no new knowledge. In SAGE, the increase of the area threshold in order to get a reasonable number of patterns leads to a small but uniform set that is flooded by the ribosomal genes which represent the most frequently over-expressed genes in the dataset. On the other hand, if the area threshold is decreased, the explosion of patterns may occur. It has been experimentally proven that the number of potentially large patterns is so high that they cannot be effectively surveyed by a human expert.

The described deficiency may be healed by augmenting internal constraints by further constraints, called *external*. An external constraint is one whose truth value is determined exclusive of the transcriptome dataset. Such constraints are for example *interestingness or expressiveness*, i.e., the future interpretability by a biologist. The interesting patterns are those exhibiting a general characteristic common for the genes and/or samples concerned (or at least their sub-sets). The more internal functional links in the pattern the more interesting the pattern.

Selectivity of selected external constraints in SAGE dataset is shown in Figure 7. The constraints capture the amount of similarity in given patterns through the measurement of the similarity of all gene pairs within that given pattern as well as they can avoid patterns with prevailing tags of an unknown function. The pruning starts with 46671 patterns that are larger than 3 genes and more frequent than 5 samples. The graphs depict that if both similarity (sumsim or insim) and existence (svsim) are thresholded, very compact sets of patterns can be reached. (Klema, 2006) gives a demonstration that these sets also gather biologically meaningful patterns.

*Figure 7. Pattern pruning by the external constraints - simultaneous application of internal and external constraints helps to arbitrarily reduce the number of patterns while attempting to conserve the potentially interesting ones. The figures show the decreasing number of patterns with increasing threshold of selected external constraints. The effect of six different constraints of various complexity is shown*



*(© 2006 IEEE Computer Society Press. Used with permission.)*

## QUANTITATIVE ASSOCIATION RULE MINING IN GENOMICS USING BACKGROUND KNOWLEDGE

Clustering is one of the most often used methods of genomic data mining. The genes with the most similar profiles are found so that the similarity among genes in one group (cluster) is maximized and similarity among particular groups (clusters) is minimized. While clustering arguably is an elegant approach to provide effective insight into data, it does have drawbacks as well, of which we name three (Becquet, 2002):

1. One gene has to be clustered in one and only one group, although it functions in numerous physiological pathways.
2. No relationship can be inferred between the different members of a group. That is, a gene and its target genes will be co-clustered, but the type of relationship cannot be rendered explicit by the algorithm.
3. Most clustering algorithms will make comparisons between the gene expression patterns in all the conditions examined. They will therefore miss a gene grouping that only arises in a subset of cells or conditions.

Admittedly, drawback 1 is tackled by soft-clustering and drawback 2 is tackled by conceptual clustering. We are not aware of a clustering algorithm void of all the three deficiencies.

**Association rule (AR) mining** can overcome these drawbacks, however transcriptomic data represent a difficult mining context for association rules. First, the data are high-dimensional (typically contain several thousands of attributes), which asks for an algorithm scalable in the number of attributes. Second, expression values are typically quantitative variables. This variable type further increases computational demands and moreover may result in an output with a prohibitive number of redundant rules. Third, the data are often noisy which may also cause a large number of rules of little significance. In this section we discuss the above-mentioned bottlenecks and present results of mining association rules using an alternative approach to quantitative association rule mining. We also demonstrate a way in which background genomic knowledge can be used to prune the search space and reduce the amount of derived rules.

## Related Work

One of the first thorough studies of AR mining on genomic data sets was provided in (Becquet, 2002). To validate the general feasibility of association rule mining in this data domain, the authors of (Becquet, 2002) have applied it to a freely available data set of human serial analysis of gene expression (SAGE). The SAGE data was first normalized and binarized as to contain only zeros and ones. These values stand for underexpression and overexpression of a given gene in a given situation, respectively. The authors selected 822 genes in 72 human cell types and generated all frequent and valid rules in the form of 'when gene a and gene b are overexpressed within a situation, then often gene c is over expressed too'.

To avoid this discretization step, authors in (Georgii, 2005) investigate the use of *quantitative association rules*, i.e., association rules that operate directly on numeric data and can represent the cumulative effects of variables. Quantitative association rules have the following form:

*If the weighted sum of some variables is greater than a threshold, then, with high probability, a different weighted sum of variables is greater than second threshold.*

An example of such rule can be:

$0.99 \times \text{gene1} - 0.11 \times \text{gene2} > 0.062 \rightarrow 1.00 \times \text{gene3} > -0.032.$

This approach naturally overcomes the discretization problem; on the other hand it is quite hard to understand the meaning of the rule. This algorithm does not exhaustively enumerate all valid and strong association rules present in the data, it uses an optimization approach.

An analysis of a microarray data-set is presented in (Carmona-Saez, 2006). The authors bring external biological knowledge to the AR mining by setting a specific language bias In particular, only gene ontology terms are allowed to appear in the antecedent part of the rule. Annotated gene expression data sets can thus be mined for rules such as:

cell cycle → [+]condition1, [+]condition2, [+]condition3, [−]condition6 which means that a significant number of the genes annotated as 'cell cycle' are over-expressed in condition 1, 2 and 3 and under-expressed in condition 6, where the conditions here correspond to time interval <T1..T7>. A proviso for this method is, of course, that ontology annotations are available for all genes in question.

## Time Complexity of Association Rule Mining

Time complexity is a serious issue in association rule mining, as it is an exponential function of sample dimensionality. To get a glimpse of the problem size, consider a binarized gene-expression dataset. Here, the number of possible itemsets is $2^{1000} \approx 10^{300}$ Although algorithms such as APRIORI use effective pruning techniques to dramatically reduce the search space traversed, the time complexity bound remains exponential.

With quantitative association rules, things get even worse. Consider the discretization into three bins, when each gene takes three values: 1 – gene is underexpressed, 2 – gene is averagely expressed, 3 – gene is overexpressed. Number of possible conditions (itemsets) grows to $5^{1000} \approx 10^{700}$, because now there are **five** possibilities for gene's value {1; 2; 3; 1..2; 2..3}. Any complete search-based algorithm becomes unfeasible even if it is completed by pruning or puts restrictions on the number of attributes on the left-hand and right-hand side (LHS, RHS). Clearly, the strong restrictions mentioned above have to be complemented by other instruments.

## Background Knowledge Experiments in Association Rule Mining

In order to increase noise robustness, focus and speed up the search, it is vital to have a mechanism to exploit BK during AR generation. In the following we employ BK in the form of a similarity matrix as defined earlier. In particular, the similarity matrix describes how likely the genes are functionally related based on the GO terms they share. The experiments are carried out in the frame of the SAGE dataset.

BK is employed in pruning. The pruning takes a following form: generate a rule only if the similarity of the genes contained in the rule is above some defined threshold'. Similarly to constraint-based learning, this condition reduces the search space and helps to speed up the algorithm. It also provides us with results, which could be better semantically explained and/or annotated.

The QAR mining algorithm presented in (Karel, 2006) was used for experiments on SAGE dataset. The QAR algorithm uses a modified procedure of rule generation – it constructs compound conditions using simple mathematical operations. Then it identifies areas of increased association between LHS and RHS. Finally, rules are extracted from these areas of increased association. The procedure is incomplete as it does not guarantee that all the rules satisfying the input conditions are reported. Although the algorithm differs in principle from traditional AR mining, it outputs association rules in the classical form.

The numbers of rules as well as the numbers of candidate rule verifications were examined during the experiments, since the number of rules quantifies the output we are interested in and the number of verifications determinates time complexity of the algorithm.

The SAGE dataset is sparse – a great portion of gene-expression values equal to zero. The distribution of zeroes among genes is very uneven. So called housekeeping genes are expressed (nearly) in all the tissues; however there is a reasonable amount of genes having zero values in almost all situations.

A total of 306 genes having more than 80% non-zero values were used in the experiment. The raw data were preprocessed and discretized into three bins using K-means discretization.

While the right hand side of rules can take arbitrary forms within the language bias, we do fix it to only refer to the target variable *cancer*, as this variable is of primary interest. Additional restrictions needed to be introduced to keep time complexity in reasonable limits. The maximum number of LHS genes was bounded. The results of experiments are summarized in Table 1.

The theoretical number of verifications is computed without considering a *min_supp* pruning, because it is hard to estimate the reached reduction. Numbers of rules and verifications using background knowledge depend on the BK pruning threshold.

A vector *gene_appearance* was generated for the purpose of overall analysis of the results; the value $gene\_appearance_i$ is equal to the number of corresponding gene appearances in generated rules. Spearman rank correlation coefficient among *gene_appearance* vectors of all results was computed, see Table 2.

As we can see using background knowledge we receive most similar rules. Surprising is negative correlation between results with 2 LHS genes with background knowledge and 3 LHS genes without background knowledge. Background knowledge influences not only number of rules generated but also the character of the rules. Some concrete examples of generated rules can be found in Table 3.

*Table 1. The number of rules and verifications for 2 and 3 antecedent genes. The settings were following: 2 gene thresholds: min_supp = 0.3, min_conf = 0.7 and min_lift = 1.3, 3 gene thresholds: min_supp = 0.15, min_conf = 0.8 and min_lift = 1.3*

| **algorithm** | number of LHS genes | number of rules | number of verifications |
|---|---|---|---|
| complete search (theoretical) | 2 | n/a | 2 318 000 |
| QAR algorithm (without BK) | | 530 | 76 747 |
| QAR algorithm (with BK) | | 92 | 12 770 |
| complete search (theoretical) | 3 | n/a | 591 090 000 |
| QAR algorithm (without BK) | | 7 509 | 14 921 537 |
| QAR algorithm (with BK) | | 243 | 699 444 |

*Table 2. Spearman rank correlation coefficients for vectors describing number of genes' appearances in generated rules*

| | 2-ant with BK | 2-ant without BK | 3-ant with BK | 3-ant without BK |
|---|---|---|---|---|
| 2-ant with BK | 1 | 0.04 | 0.29 | -0.25 |
| 2-ant without BK | 0.04 | 1 | 0.09 | 0.17 |
| 3-ant with BK | 0.29 | 0.09 | 1 | 0.26 |
| 3-ant without BK | -0.25 | 0.17 | 0.26 | 1 |

*Table 3. Examples of generated association rules. For gene expression levels it holds 1 – underexpressed, 2 – averagely expressed, 3 – overexpressed. Consequent condition stands for binary class cancer (0 – cancer did not occur, 1 – cancer did occur).*

| nr. | antecedent genes and their values | antecedent genes full name | cons. condition | conf | supp | lift |
|---|---|---|---|---|---|---|
| 1 | RPL31 = 1..2<br>NONO = 2..3 | ribosomal protein L31<br>non-POU domain containing, octamer-binding | 1 | 0.83 | 0.35 | 1.32 |
| 2 | NONO = 2..3<br>FKBP8 = 1 | non-POU domain containing, octamer-binding<br>FK506 binding protein 8, 38kDa | 1 | 0.81 | 0.31 | 1.29 |
| 3 | MIF = 1..2<br>CDC42 = 2..3 | macrophage migration inhibitory factor (glycosylation-inhibiting factor)<br>cell division cycle 42 (GTP binding protein, 25kDa) | 1 | 0.79 | 0.30 | 1.25 |
| 4 | PHB2 = 2<br>PGD = 1<br>LGALS1 = 1 | prohibitin 2<br>phosphogluconate dehydrogenase<br>lectin, galactoside-binding, soluble, 1 (galectin 1) | 1 | 0.94 | 0.15 | 1.50 |
| 5 | COPA = 1..2<br>CDC42 = 2..3<br>NDUFS3 = 2..3 | coatomer protein complex, subunit alpha<br>cell division cycle 42 (GTP binding protein, 25kDa)<br>NADH dehydrogenase (ubiquinone) Fe-S protein 3, 30kDa (NADH-coenzyme Q reductase) | 1 | 0.90 | 0.17 | 1.43 |
| 6 | PCBP1 = 2..3<br>ZYX = 1..1<br>ATP5B = 1..1 | poly(rC) binding protein 1<br>zyxin<br>ATP synthase, H+ transporting, mitochondrial F1 complex, beta polypeptide | 1 | 0.88 | 0.18 | 1.40 |

## Discussion

A heuristic QAR approach reduces the number of verifications and thus time costs. The usage of AR mining is extended beyond boolean data and can be applied on genomic data sets, although the number of attributes in the conditions has still to be restricted. The number of generated rules was also reduced by other means – there were at most two rules for each gene tuple. Consequently, the output is not flooded by quantities of rules containing the same genes having only small changes in their values.

Background knowledge was incorporated into QAR mining. BK provides a principled means to significantly reduce the search space and focus on plausible rules only. In general, the genes with prevalence of 'n/a' values in the similarity matrices are discriminated from the rules when using BK. However, a gene without annotation can still appear in a neighbourhood of 'a strong functional cluster' of other genes. This occurrence then signifies its possible functional relationship with the given group of genes and it can initiate its early annotation. On the other hand, the genes with extensive relationships to the other genes may increase their occurrence in the rules inferred with BK.

## CONCLUSION

The discovery of biologically interpretable knowledge from gene expression data is one of the hottest contemporary genomic challenges. As massive volumes of expression data are being generated, intelligent analysis tools are called for. The main bottleneck of this type of analysis is twofold – computational costs and an overwhelming number of candidate hypotheses which can hardly be further post-processed exploited by a human expert. A timely application of background knowledge available in literature, databases, biological ontologies and other sources, can help to focus on the most plausible candidates only. We illustrated a few particular ways how background knowledge can be exploited for this purpose.

Admittedly, the presented approaches to exploiting background knowledge in gene expression data mining were mutually rather isolated, despite their common reliance on the same sources of external genomic knowledge. Intuition suggests that most effective results could be obtained by their pragmatic combination. For example, gene-gene similarity has so far been computed on the sole basis of gene ontology or textual term occurrences in the respective annotations. This definition admittedly may be overly shallow. Here, the RSD mechanism of constructing non-trivial relational logic features of genes may instead be used for computing similarity: two genes would be deemed functionally similar if they shared a sufficient number of the relational logic features referring to gene functions. The inverse look at the problem yields yet another suggestion for combining the methods. The similarity matrix computed from gene ontology term occurrences can be used as a part of background knowledge which RSD uses to construct features. Technically, a new predicate *similar(A,B)* would be introduced into the feature language, while its semantics for two genes *A* and *B* would be determined in the obvious way from the precomputed similarity matrix. These ideas form grounds for our future explorations.

## ACKNOWLEDGMENT

Jiří Kléma, Filip Železný and Filip Karel have been supported by the grant 1ET101210513 ”Relational Machine Learning for Analysis of Biomedical Data” funded by the Czech Academy of Sciences. The Czech-French travels were covered by Czech-French PHC Barrande project “Fusion de données hétérogenes pour la découverte de connaissances en génomique’’. The Czech-USA travels were covered by Czech Ministry of Education through project ME910 “Using Gene Ontologies and Annotations for the Interpretation of Gene Expression Data through Relational Machine Learning Algorithms”. We thank Olivier Gandrillon and his team at the Centre de Génétique Moléculaire et Cellulaire for helpful and stimulating biological discussions and providing us with the SAGE dataset. We are grateful to Arnaud Soulet for providing us with his tool MUSIC applied in constraint-based knowledge discovery.

## REFERENCES

Becquet, C., Blachon, S., Jeudy, B., Boulicaut, J. F. & Gandrillon O. (2002). Strong Association Rule Mining for Large Gene Expression Data Analysis: A Case Study on Human SAGE Data. *Genome Biology***,** 3(12):531-537.

Breiman, L. (2001) Random Forests. *Machine Learning*, 45(1), 5–32.

Carmona-Saez, P., Chagoyen, M., Rodriguez, A., Trelles, O., Carazo, J. M., & Pascual-Montano, A. (2006). Integrated analysis of gene expression by association rules discovery. BMC *Bioinformatics*, *7*, 54.

Chaussabel, D., & Sher, A. (2002). Mining microarray expression data by literature profiling. *Genome Biology, 3***.**

Clark, P., & Niblett, T. (1989). The CN2 induction algorithm. *Machine Learning*, 261–283.

Cremilleux, B., Soulet, A., Klema, J., Hebert, C., & Gandrillon, O. (2009). Discovering Knowledge from Local Patterns in SAGE data. In P. Berka, J. Rauch and D. J. Zighed (Eds.), *Data mining and medical knowledge management: Cases and applications*. Hershey, PA: IGI Global.

Diaz-Uriarte, R., & Alvarez de Andres, S. (2006). Gene selection and classification of microarray data using random forest. *BMC Bioinformatics*, *7*(3).

Furey, T.S., Cristianini, N., Duffy, N., Bednarski, D.W., Schummer, M. & Haussler, D. (2000). Support vector machine classifcation and validation of cancer tissue samples using microarray expression data. *Bioinformatics*, *16*, 906–914.

Georgii, E., Richter, L., Ruckert, U., & Kramer S. (2005) Analyzing Microarray Data Using Quantitative Association Rules. *Bioinformatics*, 21(Suppl. 2), ii123–ii129.

Glenisson, P., Mathys, J., & Moor, B. D. (2003) Meta-clustering of gene expression data and literature-based information. *SIGKDD Explor. Newsl*. 5(2), 101–112.

Golub, T., Slonim, D., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J., Coller, H., Loh, M., Downing, J., Caligiuri, M., Bloomfield, C., & Lander, E. (1999). Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 531–537.

Hanczar, B., Courtine, M., Benis, A., Hennegar, C., Clement, C. & Zucker, J. D. (2003). Improving classification of microarray data using prototype-based feature selection. *SIGKDD Explor. Newsl.*, *5*(2), 23--30. ACM, NY, USA.

Karel, F. (2006) Quantitative and ordinal association rules mining (QAR mining). In *Knowledge-Based Intelligent Information and Engineering Systems*, 4251, 195–202. Springer LNAI.

Karel, F., & Klema, J. (2007). Quantitative Association Rule Mining in Genomics Using Apriori Knowledge. In Berendt, B., Svatek, V. Zelezny, F. (eds.), Proc. of The *ECML/PKDD Workshop On Prior Conceptual Knowledge in Machine Learning and Data Mining*. University of Warsaw, Poland, (pp. 53-64).

Keime, C., Damiola, F., Mouchiroud, D., Duret, L. & Gandrillon, O. (2004). Identitag, a relational database for SAGE tag identification and interspecies comparison of SAGE libraries. *BMC Bioinformatics*, *5*(143).

Klema, J., Soulet, A., Cremilleux, B., Blachon, S., & Gandrilon, O. (2006). Mining Plausible Patterns from Genomic Data. *Proceedings of Nineteenth IEEE International Symposium on Computer-Based Medical Systems*, Los Alamitos: IEEE Computer Society Press, 183-188.

Klema, J., Soulet, A., Cremilleux, B., Blachon, S., & Gandrilon, O. (submitted). Constraint-Based Knowledge Discovery from SAGE Data. Submitted to *In Silico Biology.*

Lavrac, N., Zelezny, F., & Flach, P. (2002). RSD: Relational subgroup discovery through first-order feature construction. In *Proceedings of the 12th International Conference on Inductive Logic Programming*, 149–165.

Lee, J. W., Lee, J. B., Park, M., & Song, S. H. (2005). An extensive evaluation of recent classification tools applied to microarray data. *Computation Statistics and Data Analysis*, *48*, 869-885.

Lee, Y., & Lee, Ch. K. (2003) Classification of Multiple Cancer Types by Multicategory Support Vector Machines Using Gene Expression Data. *Bioinformatics*, *19*(9), 1132-1139.

Martin, D., Brun, C., Remy, E., Mouren, P., Thieffry, D., & Jacq, B. (2004). GOToolBox: functional investigation of gene datasets based on Gene Ontology. *Genome Biology 5*(12), R101.

Quinlan, J.R. (1986) Induction of decision trees. *Machine Learning*, *1*, 81-106.

Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing Management, 24*(5), 513–523.

Scheffer, T., & Wrobel, S. (2002). Text Classification Beyond the Bag-of-Words Representation. Proceedings of the *International Conference on Machine Learning (ICML) Workshop on Text Learning.*

Sevon, P., Eronen, L., Hintsanen, P., Kulovesi, K., & Toivonen, H. (2006). Link discovery in graphs derived from biological databases. In *3rd International Workshop on Data Integration in the Life Sciences* (DILS'06), Hinxton, UK.

Soulet, A., Klema J., & Cremilleux, B. (2007). Efficient Mining Under Rich Constraints Derived from Various Datasets. In Džeroski, S., Struyf, J. (eds.), *Knowledge Discovery in Inductive Databases*, LNCS,4747, 223-239. Springer Berlin / Heidelberg.

Tibshirani, R., Hastie, T., Narasimhan, B., & Chu, G. (2002). Diagnosis of multiple cancer types by shrunken centroids of gene expression. Proc. *Natl Acad. Sci*., 99(10), 6567-6572.

Trajkovski, I., Zelezny, F., Lavrac, N., & Tolar, J. (in press). Learning Relational Descriptions of Differentially Expressed Gene Groups. *IEEE Trans. Sys Man Cyb C, spec. issue on Intelligent Computation for Bioinformatics.*

Trajkovski, I., Zelezny, F., Tolar, J., & Lavrac, N. (2006) Relational Subgroup Discovery for Descriptive Analysis of Microarray Data. In *Procs 2nd Int Sympos on Computational Life Science*, Cambridge, UK 9/06. Springer Lecture Notes on Bioinformatics / LNCS.

Trna, M. (2007) *Klasifikace s apriorni znalosti*. CTU Bachelor's Thesis, In Czech.

Tusher, V.G., Tibshirani, R. & Chu G. (2001) Significance analysis of microarrays applied to the ionizing radiation response. Proc. *Natl Acad. Sci*., 98(9). 5116–5121.

Velculescu, V., Zhang, L., Vogelstein, B. & Kinzler, K. (1995). Serial Analysis of Gene Expression. *Science, 270*, 484–7.

Zelezny, F., & Lavrac, N. (2006) Propositionalization-Based Relational Subgroup Discovery with RSD. *Machine Learning, 62*(1-2), 33-63.

## KEY TERMS

**ALL, AML:** Leukemia is a form of cancer that begins in the blood-forming cells of the bone marrow, acute leukemias usually develop suddenly (whereas some chronic varieties may exist for years before they are diagnosed), acute myeloid leukemia (AML) is the most frequently reported form of leukemia in adults while acute lymphoblastic leukemia (ALL) is largely a pediatric disease.

**Association Rule:** A rule, such as implication or correlation, which relates elements co-occurring within a dataset.

**Background Knowledge:** Information that is essential to understanding a situation or problem, knowledge acquired through study or experience or instruction that can be used to improve the learning process.

**Classifier:** A mapping from unlabeled instances (a discrete or continuous feature space X) to discrete classes (a discrete set of labels Y), a decision system which accepts values of some features or characteristics of a situation as an input and produces a discrete label as an output.

**Constraint:** A restriction that defines the focus of search, it can express allowed feature values or any other user's interest.

**DNA, RNA, mRNA:** Deoxyribonucleic acid (DNA) is a nucleic acid that contains the genetic instructions used in the development and functioning of all known living organisms, ribonucleic acid (RNA) is transcribed from DNA by enzymes, messenger RNA (mRNA) is the RNA that carries information from DNA to the ribosome sites of protein synthesis (translation) in the cell, the coding sequence of the mRNA determines the amino acid sequence in the protein that is produced.

**Functional Genomics:** A field of molecular biology that attempts to make use of the vast wealth of data produced by genomic projects to describe gene and protein functions and interactions.

**Gene Expression:** The process of transcribing a gene's DNA sequence into the RNA that serves as a template for protein production.

**Gene Ontology:** A controlled vocabulary to describe gene and gene product attributes in any organism.

**Knowledge Discovery:** The nontrivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data.

**RefSeq:** Non-redundant curated data representing current knowledge of a known gene.

**Relational Data Mining:** Knowledge discovery in databases when the database has information about several types of objects.

**SRC:** V-src sarcoma (Schmidt-Ruppin A-2) viral oncogene homolog (avian) – a randomly taken gene to illustrate knowledge representation format.