Boolean Property Encoding for Local Set Pattern Discovery: An Application to Gene Expression Data Analysis

Ruggero G. Pensa and Jean-François Boulicaut

INSA Lyon
LIRIS CNRS UMR 5205
F-69621 Villeurbanne cedex, France
{Ruggero.Pensa,Jean-Francois.Boulicaut}@insa-lyon.fr

Abstract. In the domain of gene expression data analysis, several researchers have recently emphasized the promising application of local pattern (e.g., association rules, closed sets) discovery techniques from boolean matrices that encode gene properties. Detecting local patterns by means of complete constraint-based mining techniques turns to be an important complementary approach or invaluable counterpart to heuristic global model mining. To take the most from local set pattern mining approaches, a needed step concerns gene expression property encoding (e.g., over-expression). The impact of this preprocessing phase on both the quantity and the quality of the extracted patterns is crucial. In this paper, we study the impact of discretization techniques by a sound comparison between the dendrograms, i.e., trees that are generated by a hierarchical clustering algorithm on raw numerical expression data and its various derived boolean matrices. Thanks to a new similarity measure, we can select the boolean property encoding technique which preserves similarity structures holding in the raw data. The discussion relies on several experimental results for three gene expression data sets. We believe our framework is an interesting direction of work for the many application domains in which (a) local set patterns have been proved useful, and (b) Boolean properties have to be derived from raw numerical data.

1 Introduction

This volume is dedicated to local pattern detection. It has been motivated by the need for a better characterization of what is local pattern detection and what are the main research challenges in this area. We contribute to this objective by considering the exciting application domain of *transcription module discovery* from gene expression data. In this molecular biology context, the goal is to identify sets of genes which seem to be co-regulated, associated with the sets of biological situation which seems to trigger the co-regulation.

The state-of-the-art is that global patterns like partitions can provide some useful information and suggest some of the transcription modules. We are however interested by the intrinsic limitations of these approaches, e.g., their heuristic nature or the lack of unexpectedness of the findings. We strongly believe

K. Morik et al. (Eds.): Local Pattern Detection, LNAI 3539, pp. 115–134, 2005.

that complete extractions of local patterns which satisfy a given conjunction of constraints (e.g., a minimal frequency constraint or a maximality constraint) are an invaluable and complementary approach to suggest unexpected but relevant patterns, i.e., putative transcription modules.

Let us now introduce the application domain and our contribution. Thanks to a huge research effort and technological breakthroughs, one of the challenges for molecular biologists is to discover knowledge from data generated at very high throughput. For instance, different techniques (including microarray [1] and SAGE [2]) enable to study the simultaneous expression of (tens of) thousands of genes in various biological situations. The data generated by those experiments can be seen as expression matrices in which the expression level of genes (rows) is recorded in various biological situations (columns). A toy example of some microarray data is the matrix in Tab. 1a.

	1	2	3	4	5
a	-1	6	0	12	9
b	3	-2	3	-3	1
c	0	5	-1	6	6
d	4	-1	2	-2	-1
e	-3	9	1	10	6
f	5	-3	3	-6	0
g	4	-4	3	-7	0
h	-2	2	-2	8	5

(a)

	1	2	3	4	5
a	0	1	0	1	1
b	1	0	1	0	1
c	0	1	0	1	1
d	1	0	1	0	0
e	0	1	0	1	1
f	1	0	1	0	1
g	1	0	1	0	1
h	0	0	0	1	1

(b)

	1	2	3	4	5
a	0	0	0	1	0
b	1	0	1	0	0
c	0	0	0	1	1
d	1	0	0	0	0
e	0	0	0	1	0
f	1	0	0	0	0
g	1	0	0	0	0
h	0	0	0	1	0

(c)

Table 1. A gene expression matrix (a) with two derived boolean matrices (b and c)

Once large gene expression datasets are available, biologists have to drop the traditional one-to-one approach to gene expression data analysis and crucially need for Knowledge Discovery in Databases techniques (KDD). Among the classical KDD approaches, classification techniques (i.e., learning a classifier from data which, for example, can predict a cancer diagnosis according to individual gene expression profiles) have been intensively studied (see, e.g., [3] for a collection of recent contributions). In this paper, we do not consider such problems. We are interested in descriptive techniques which provides either global patterns like partitions (clustering) or local patterns like co-regulated sets of genes and/or sets of situations.

The use of hierarchical clustering (see, e.g., [4]) is indeed quite popular among practitioners. Genes are grouped together according to similar expression profiles. The same can be done on biological situations. Thanks to the appreciated vizualization component introduced with [4], biologists can identify some putative transcription modules. Practitioners do not use only hierarchical clustering but also most of the classical clustering techniques. A common characteristic of

these techniques is that global patterns like partitions are extracted by means of a heuristic search. They provide "global pictures" of similarity structures. Not only the heuristic nature can lead to different results for different experiments but also, the fact we get global patterns, i.e., which hold in the whole data, leads to rather expected findings. Our thesis is that unexpected patterns are a priori interesting and that they are typically local ones, i.e., they hold in only a part of the data. Therefore, looking for collections of local patterns in gene expression data appears as a promising and complementary approach. The last 5 years, a major research sub-domain in data mining has concerned the design of efficient and complete constraint-based mining tools on boolean data, also called transactional data by some authors. The completeness assumption means that every pattern from the pattern language which satisfies the defined constraints has to be returned (e.g., every frequent set, every closed set, every frequent and closed set which does not contain a given item). In general, and this is the case for our work, non heuristic methods are used.

To apply these techniques for gene expression data analysis, we have to encode boolean gene expression properties, e.g., over-expression, strong variation, co-regulation. Tab. 1b and Tab. 1c are two data sets derived from the toy microarray data from Tab. 1a. Once such boolean data sets are available, it is possible to look for putative synexpression groups (see [5]) by computing the popular frequent sets (frequent sets of situations in a matrix $Genes \times Situations$ and frequent sets of genes in its transposition). Given the number of genes, we can alternatively compute condensed representations of the frequent sets, e.g., the frequent closed sets [6,7,8]. Deriving association rules from synexpression groups has been studied as well [9,10]. Furthermore, putative transcription modules can be provided by computing the so-called formal concepts (see, e.g., [11,12,13]). Also, constraint-based mining of concepts has been considered [14,15]. Notice that the collection of every formal concept which can be extracted from large real gene expression matrices can be considered as a collection of overlapping clusters on either the genes or the situations. The global picture is not there but every locally strong association (associated closed sets, see Section 3) has been captured.

So far, very few studies have concerned the quality of gene expression property encoding, i.e., a kind of feature construction phase. This is a critical step because its impact on both the quantity and the quality of the extracted patterns is crucial.

If $\mathcal{S}$ denotes the set of biological situations and $\mathcal{P}$ denotes the set of genes, the expression properties can be encoded into $\mathbf{r} \subseteq \mathcal{P} \times \mathcal{S}$. $(g_i, o_j) \in \mathbf{r}$ denotes that gene i has the encoded expression property in situation j. Different expression properties might be considered. Without loss of generality, we consider that only one expression property is encoded for each gene, which means that we can talk indifferently of genes or gene expression properties. Generally, encoding is performed according to some discretization operators that, given user-defined parameters, transform each numerical value from raw gene expression data into one boolean value per gene property. Many operators can be used that typically

compute thresholds from which it is possible to decide wether the true or the false value must be assigned. For instance, in Tab. 1b, an over-expression property has been encoded and, e.g., Genes a, c, and e are over-expressed together in Situations 2, 4 and 5.

In [16], we have proposed a method which supports the choice for a discretization technique and an informed decision about its parameters. The idea was to study the impact of discretization by a sound comparison between the dendrograms (i.e., binary trees) that are generated by the same hierarchical clustering algorithm applied to both the raw expression data and various derived boolean matrices. This paper is a significant extension of [16]. The framework has been revisited and the experimental validation have been considerably extended.

In Section 2, we refine the similarity measure introduced in [16]. It is level independent, and it depends for each node on its subtree structure. It can be applied on gene and/or situation dendrograms and we introduce an aggregated measure for considering both simultaneously. Section 3 is dedicated to the use of this similarity measure on three real gene expression data sets in order to select an adequate discretization technique. The robustness of the approach is also emphasized by an a posteriori analysis of the extracted patterns in the various boolean contexts. For this purpose, we adapt the similarity measure between collections of patterns introduced in [17]. In Section 4, we study further the robustness of our approach by comparing several clustering results in the raw data. Section 5 is a short conclusion.

2 Boolean Encoding Assessment

2.1 Comparing Binary Trees

The problem of tree comparison has motivated a lot of research. Designing similarity measures between trees is difficult because it has to be defined according to the semantics of trees and similarities which are generally application domain dependant. For instance, considering the analysis of phylogenies, distance measures between both rooted and unrooted trees have been designed to compare different phylogenetic trees concerning the same set of individuals (e.g., different species of animals having a common ancestor). Various distance metrics between trees have been proposed. The `nni` (nearest neighbor interchange) and the `mast` (maximum agreement subtree) are two of the most used metrics. `nni` has been introduced independently in [18] and [19] and its NP-completeness has been recently proved [20,21]. `mast` has been proposed in [22], and [23] describes an efficient algorithm for computing this metrics on binary trees. These two approaches are tailored for the problem of comparing phylogenies where the goal is to measure some degree of isomorphism between two dendrograms representing the same species of biological organisms.

In our data mining problem, we have sets of objects (vectors of expression values for genes in various biological situations), that we want to process with a hierarchical clustering algorithm. Depending on the different discretization operations on raw expression data, a same clustering algorithm working on encoded

boolean gene expression data can return (very) different results. We are looking for a method that supports the comparison of these various gene and/or situation dendrograms obtained on boolean data w.r.t. the common reference dendrogram that has been computed from the raw data. We need to measure both the degree of similarity of their structures and the similarity between the contents of their associated collections of clusters. We introduced in [16] a simple measure which is also easy to compute. Intuitively, it depends on the number of matching nodes between the two trees we have to compare.

2.2 Definition of Similarity Scores

Let $\mathcal{O} = \{o_1, \ldots, o_n\}$ denote a set of n objects. Let T denote a binary tree built on $\mathcal{O}$. Let $\mathcal{L} = \{l_1, \ldots, l_n\}$ denote the set of n leaves of T associated to $\mathcal{O}$ for which, $\forall i \in [1 \ldots n]$, $l_i \equiv o_i$. Let $\mathcal{B} = \{b_1 \ldots b_{n-1}\}$ denote the set of the $n-1$ nodes of T generated by a hierarchical clustering algorithm starting from $\mathcal{L}$. By construction, we consider $b_{n-1} = r$, where r denotes the root of T. We define the two sets:

$$\delta(b_i) = \{b_j \in \mathcal{B} \mid b_j \textit{ is a descendent of } b_i\},$$

$$\tau(b_i) = \{l_j \in \mathcal{L} \mid l_j \textit{ is a descendent of } b_i\}.$$

An example of a tree for the genes from Tab. 1a is given in Fig. 1. Here, $\tau(b_3) = \{b, d, f, g\}$ and $\delta(b_3) = \{b_1, b_2\}$.

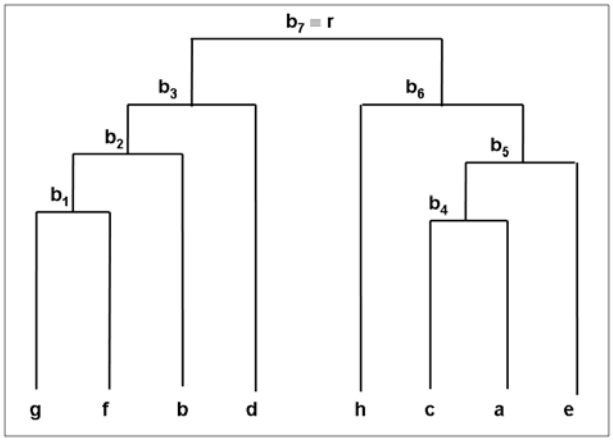


Fig. 1. An example of binary tree

We want to measure the similarity between a tree T and a reference tree T_{ref} built on the same set of objects $\mathcal{O}$. For each node b_i of T, we define the following score (denoted S_B and called **BScore**):

$$S_B(b_i, T_{ref}) = \sum_{b_j \in \delta(b_i)} a_j$$

$$a_j = \begin{cases} \frac{1}{|\tau(b_j)|}, & if \\ & \exists b_k \in T_{ref} \mid \tau(b_j) = \tau(b_k) \\ 0, & otherwise \end{cases} \quad (1)$$

In other terms, for a node b in T, its score depends both on the number of its matching nodes in T_{ref} ($b_k \in T_{ref}$ is a matching node for b if $\tau(b) = \tau(b_k)$) and $|\tau(b)|$. To obtain the similarity score of T w.r.t. T_{ref} (denoted S_T and called **TScore**), we consider the **BScore** value on the root, i.e.:

$$S_T(T, T_{ref}) = S_B(r, T_{ref}) \quad (2)$$

As usually, it is interesting to normalize the measure to get a score between 0 (for a tree which is totally different from the reference) and 1 (for a tree which is equal to the reference). For the **TScore** measure, since its maximal value depends on the tree morphology, we can normalize by $S_T(T_{ref}, T_{ref})$:

$$\overline{S_T}(T, T_{ref}) = \frac{S_T(T, T_{ref})}{S_T(T_{ref}, T_{ref})} \quad (3)$$

$\overline{S_T}(T, T_{ref}) = 0$ means that T is totally different from T_{ref}, i.e., there are no matching node between T and T_{ref}. Indeed, $\overline{S_T}(T, T_{ref}) = 1$ means that T is totally similar to T_{ref}, i.e., every node in T matches with a node in T_{ref}. Given two trees T_1 and T_2 and a reference T_{ref}, if $\overline{S_T}(T_1, T_{ref}) < \overline{S_T}(T_2, T_{ref})$, then T_2 is said to be more similar to T_{ref} than T_1 according to **TScore**.

An important property (missing from [16]) is the following:

Property 1. The measure 1 is asymmetric, i.e. given a reference tree T_{ref}, $\exists T$ such that $\overline{S_T}(T, T_{ref}) \neq \overline{S_T}(T_{ref}, T)$.

As a consequence of this property, such a measure makes sense when one wants to compare different binary trees with the same reference. If a symmetric measure is needed, one can consider the mean of the two possible measures for a couple of trees:

$$\frac{\overline{S_T}(T_1, T_2) + \overline{S_T}(T_2, T_1)}{2}.$$

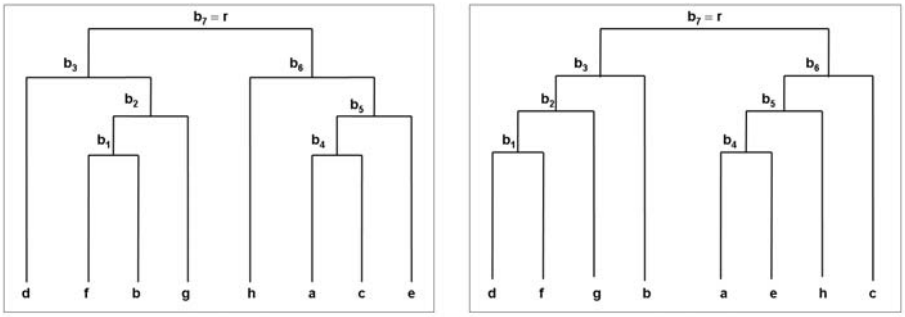
2.3 Comparison Between Gene Dendrograms

Tab. 1a is a toy example of a gene expression matrix. Each row represents a gene vector, and each column represents a biological sample vector. Each cell contains an expression value for a given gene and a given sample. In this example, we have $\mathcal{O} = \{a, b, c, d, e, f, g, h\}$. A hierarchical clustering using the Pearson's correlation coefficient and the average linkage method (see, e.g., [4]) on the data from Tab. 1a leads to the dendrogram in Fig. 1.

Assume now that we discretize the expression matrix by applying two different methods used for over-expression encoding [9]. The first one, the so-called

"Mid-Ranged" method, considers the mean between the maximal and minimal values for each gene vector. Values which are greater than the average value are set to 1, 0 otherwise (Tab. 1b). A second method, the so-called "Max - X% Max" method, takes into account the maximal value for each gene vector. Values that are greater than $(100 - X)\%$ of the maximal value are set to 1, 0 otherwise. We set X to 10 deriving the matrix in Tab. 1c.

Assume now that we use the same clustering algorithm on the two derived boolean data sets. The resulting dendrograms are shown in Fig. 2. Fig. 2a (resp. Fig. 2b) represents the gene dendrogram obtained by clustering the boolean matrix in Tab. 1b (resp. Tab. 1c).



a) b)

Fig. 2. Gene trees built on two differently discretized matrices

We can now use the similarity score and decide which discretization is better for this gene expression data set, i.e., the one for which $\overline{S_T}(T, T_{ref})$ has the largest value. The common reference (T_{ref}) is the tree in Fig. 1. Let T_a and T_b denote the trees in Fig. 2a and 2b respectively. Using Equation 3, we obtain:

$$\overline{S_T}(T_a, T_{ref}) = 0.77 \qquad \overline{S_T}(T_b, T_{ref}) = 0.23.$$

Since $\overline{S_T}(T_a, T_{ref}) > \overline{S_T}(T_b, T_{ref})$, the first discretization method is considered better for this data set w.r.t. the performed hierarchical clustering. In fact, in T_a, only node b_1 does not match (i.e., it does not share the same set of leaves) with any node in T_{ref}, while in T_b, there are only two nodes (b_3 and b_6) that match with some nodes in T_{ref}.

The same process can be applied to situation dendrograms by considering now that the objects are the situations. In practice, we perform both processes to support the choice of a discretization technique as illustrated in the next section.

2.4 Average Similarity Score

When we compare both situation and gene trees, we have different results for each comparison. According to our practice of gene expression data analysis, we

often have thousands genes and a few tens or hundreds of situations. It means that, the similarity scores computed for situations tree are usually greater than those computed for gene dendrograms. This can be explained by the fact that situation dendrograms have more probabilities to be identical, since they contains less leaves, and the correlation coefficients (during the hierarchical clustering process) are computed on vectors of thousands components (the genes whose expression is measured in each situation). As a result, if we compare differently discretized gene expression matrix, the discretization thresholds for which we get the highest similarity score can be different for gene and situation dendrograms.

If we are interested in a unique similarity score, different solutions can be adopted. For example, we can consider the average between the gene and the situation similarity scores. A problem is that if one of the trees is totally dissimilar from the reference (relative score is equal to zero), the average value will not be zero. We can solve this problem by simply considering the square root of the product between the two similarity scores:

$$\overline{S_{AT}}\,(T, T_{ref}) = \sqrt{\overline{S_{GT}}\,(T, T_{ref}) \cdot \overline{S_{ST}}\,(T, T_{ref})} \tag{4}$$

where $\overline{S_{GT}}$ and $\overline{S_{ST}}$ and denote respectively the normalized similarity score for genes and situation, and $\overline{S_{AT}}$ denotes the average similarity score.

Following this definition, $\overline{S_{AT}}$ is always between the gene and the situation similarity score values. Furthermore, when at least one of the two similarity scores is equal to zero, also the average similarity score is zero.

3 Using Similarity Scores

Many discretization techniques can be used to encode gene expression properties from expression values that are either integer values (case for SAGE data [2]) or real values (case for microarray data [1]). In this paper, we consider for our experimental study only three techniques that have been used for encoding the over-expression of genes in [9]:

- "Mid-Ranged". The highest and lowest expression values are identified for each gene and the mid-range value is defined. For a given gene, all expression values that are strictly above the mid-range value give rise to value 1, 0 otherwise.
- "Max - X% Max". The cut off is fixed w.r.t. the maximal expression value observed for each gene. From this value, we remove a percentage X of this value. All expression values that are greater than the $(100 - X)\%$ of the Max value give rise to value 1, 0 otherwise.
- "X% Max". For each gene, we consider the situations in which its level of expression is in X% of the highest values. These genes are assigned to value 1, 0 otherwise.

We want to evaluate the relevancy of a discretization algorithm and its parameters according to the preserved properties w.r.t. a hierarchical clustering of

the raw data. So, we have to compare the dendrograms obtained from the three different boolean matrices with the reference dendrogram.

We have considered three gene expression data sets: two microarray data sets and a SAGE data set. The first data set (CAMDA [24]) concerns the transcriptome of the intraerythrocytic developmental cycle of the plasmodium falciparum, a parasite that is responsible for a very frequent form of malaria. We have the expression values for 3 719 genes in 46 different time points, i.e., biological situations. The second data set (Drosophila [25]) concerns the gene expression of drosophila melanogaster during its life cycle. We have the expression values for 3 030 genes and 81 biological situations. The third one (human SAGE data from NCBI, see also [26,13]) contains the expression values for 5 327 human genes in 90 different cancerous and not cancerous cellular samples belonging to different human organs.

One indicator of the differences between derived boolean contexts is their density, i.e., the number of true values divided by the total number of cells in the matrices. In Fig. 3, we provide the density curves for the three data sets and depending on different thresholds for the "Max - X% Max" method. Notice that densities for the "X% Max" method are equal to X.

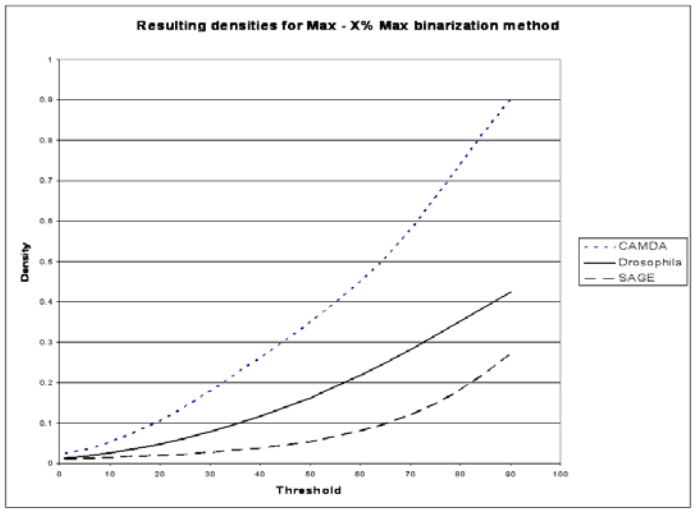


Fig. 3. Density values for different "Max - X% Max" thresholds

We processed all the computed boolean matrices with a hierarchical clustering algorithm based on the centered Pearson's correlation coefficient and the average linkage method. The same algorithm with the same options has been applied to the three original matrices. Finally, for each data set, we have compared all the genes and situations trees derived from the boolean matrices with the reference trees. The results in terms of **TScore** (Equation 1) for the "Mid-Ranged" method, are summarized in Fig. 4.

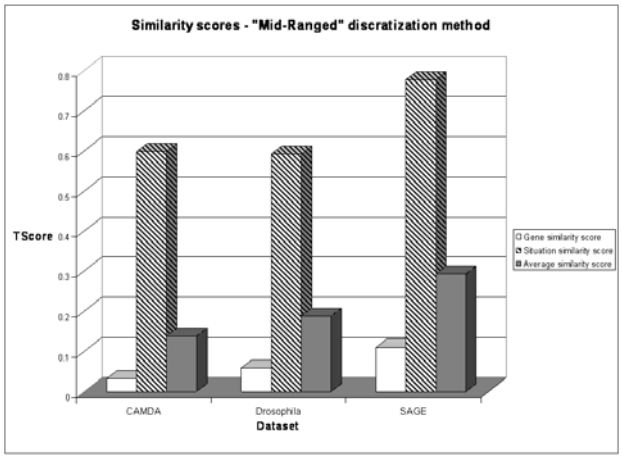


Fig. 4. Similarity scores for clustering trees on Mid-Ranged discretized matrices

For the "Max - X% Max" and "X% Max" methods, we summarize the results depending on the variation of the threshold X for the gene dendrograms in Fig. 5a and Fig. 6c, for the situation dendrograms in Fig. 5b and Fig. 6d. It is important to observe that, for each data set, we obtained the highest values of similarity scores for both the genes and the situations for almost the same discretization thresholds.

We have used the definition of average similarity score (Equation 4), to identify a unique measure of similarity for each boolean context. Results are summarized in Fig. 7.

We have also applied the same clustering algorithm on various randomly generated boolean matrices based on the same sets of objects. Then, we have compared the resulting dendrograms with the reference. In the first two data sets (CAMDA and Drosophila), the similarity scores of the randomly generated boolean matrices are always very low or equal to 0. In the SAGE data set, given a density value, the gene scores resulting from randomly generated matrices are always lower than the ones obtained by any discretization method (while the situation scores are always negligible). One explanation could be that the discretized matrices are here very sparse compared to the ones we derive from the first two data sets (see Fig. 3). Using a low threshold to discretize such a matrix does not make sense: obtained scores are similar to the scores which are computed on random boolean matrices. Moreover, using a high threshold value X for the "X% Max" discretization method leads to similarity scores that are close to those obtained for randomly generated matrices, though still higher. We can observe the behavior of this particular SAGE data set in Fig. 8.

As we can see, each discretization method has a set of threshold values for which it produce relatively high results in terms of similarity scores. Obviously, depending on the analysis task, one method can be more adapted than the

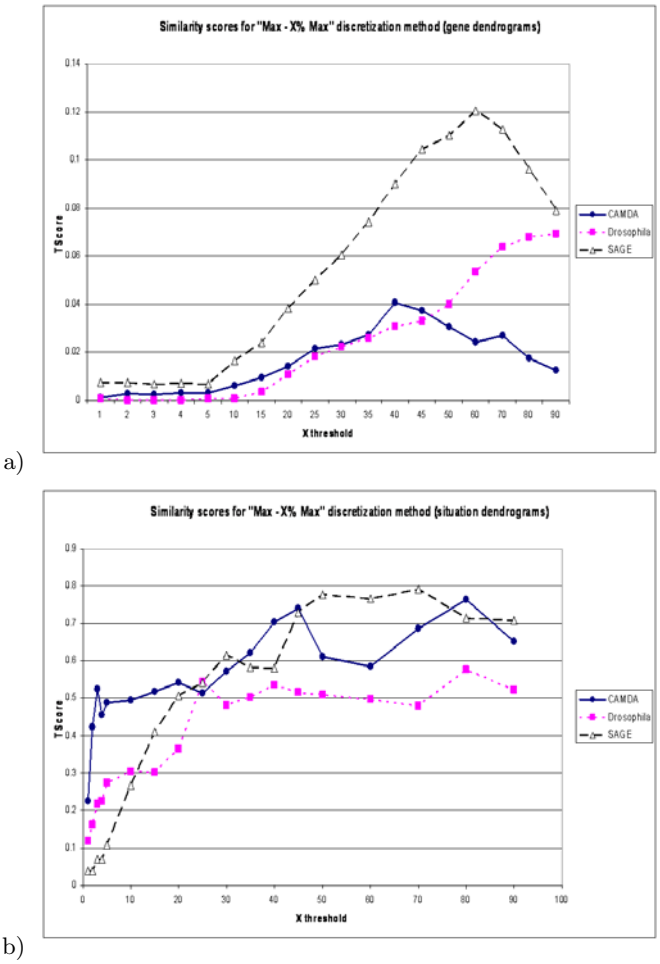
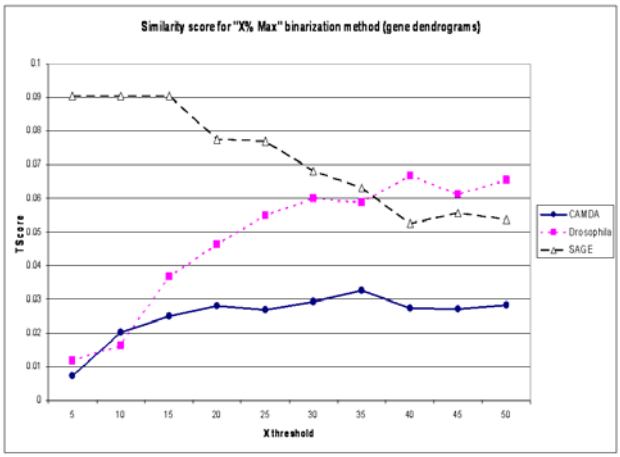


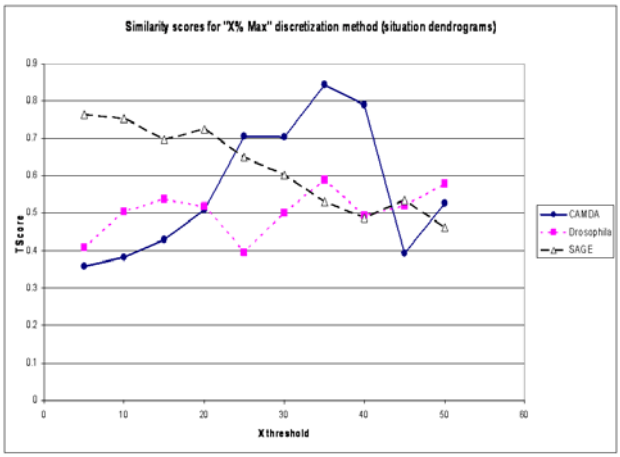
Fig. 5. Similarity scores w.r.t. different thresholds for "Max - X%Max"

other ones. For instance, even if both the "Max - X% Max" and the "X% Max" methods encode over-expression, the first one produces a boolean context whose density is strictly dependent on the maximal expression value for each gene. Instead, with the second method, we are sure that the density of the resulting boolean context is near to the X threshold. Does it mean that we are able to extract different kinds of patterns?

Clearly, the collections of patterns we can extract when using two different discretization techniques for over-expression encoding, will be different. We consider however that if we extract in proximity of the thresholds which produced the highest similarity scores for both methods, the intersection between the extracted collections will have a significant size. Patterns belonging to this intersection will also inform about rather strong associations.



c)



d)

Fig. 6. Similarity scores w.r.t. different thresholds for "X%Max"

We have analyzed such intersections between different collections of formal concepts ([11]) which have been extracted from the boolean SAGE data set.

Definition 1. *$(G, T) \in \mathcal{P} \times \mathcal{S}$ is a formal concept in $\mathbf{r} \subseteq \mathcal{P} \times \mathcal{S}$ when $T = \psi(G, \mathbf{r})$ and $G = \phi(T, \mathbf{r})$. ψ and ϕ are the classical Galois operators, i.e., we have $\phi(T, \mathbf{r}) = \{g \in \mathcal{P} \mid \forall o \in S, (g, o) \in \mathbf{r}\}$ and $\psi(G, \mathbf{r}) = \{o \in \mathcal{S} \mid \forall g \in G, (g, o) \in \mathbf{r}\}$. (ϕ, ψ) is the so-called Galois connection between $\mathcal{S}$ and $\mathcal{P}$. Notice that, by construction, when (G, T) is a formal concept, G and T are closed sets.*

We used the D-Miner algorithm [14] to extract formal concepts under constraints: to avoid problems with outliers, we have considered formal concepts with at least 2 biological situations and at least 10 genes (i.e., $|G| \geq 10$ and

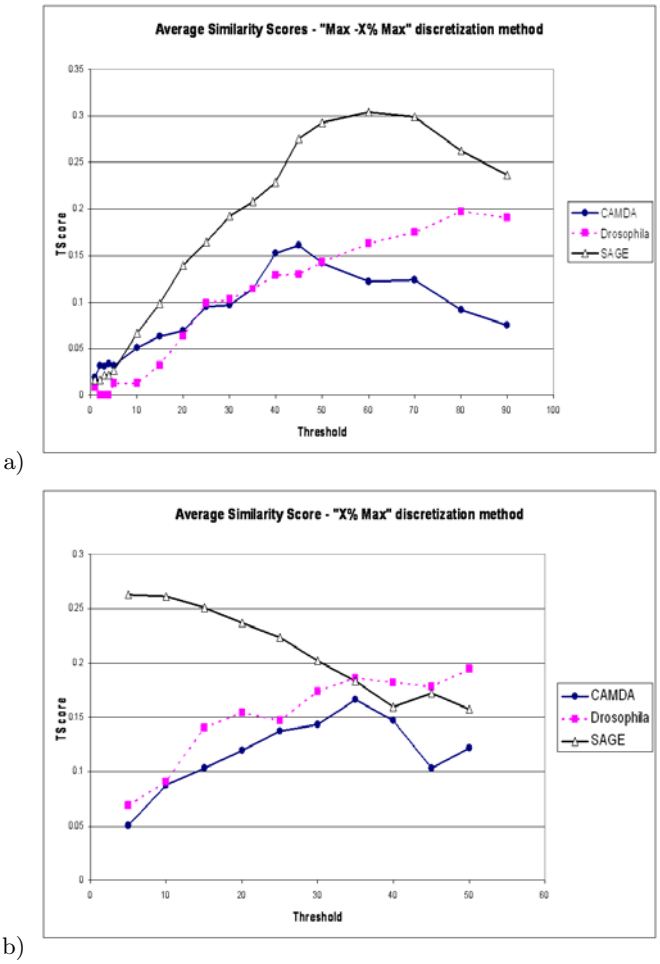


Fig. 7. Average similarity scores w.r.t. different thresholds for "Max - X%Max" (a) and "X%Max" (b)

$|T| \geq 2$). The mined boolean contexts have been obtained by the "Max - X% Max" and the "X% Max" over-expression encoding methods. We used the X threshold values which have produced the highest similarity scores (see Fig. 7). Then we compared all the collections extracted from each boolean context obtained with the first method, with all the collections related to the second method.

To compare pattern collections, we adapted the interactive self-similarity metrics introduced in [17]. Such a measure has been studied for comparing two collections of frequent itemsets extracted from two samples of a same data set. We modified it to work on formal concepts extracted from different boolean instances of a same data set.

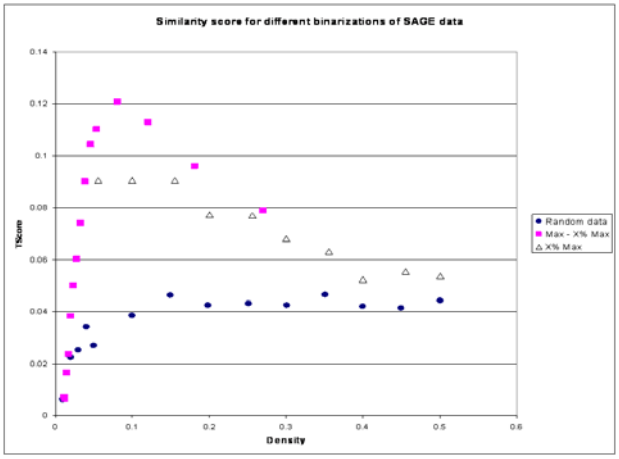


Fig. 8. Similarity scores w.r.t. density for "Max - X%Max", "X%Max" and random discretization methods on SAGE data

Given two boolean contexts $\mathbf{r}_1$ and $\mathbf{r}_2$ our pattern collection similarity measure is defined as follows:

$$Sim\left(\mathcal{C}_1, \mathcal{C}_2\right) = \frac{\sum_{x \in \{T_1\} \cap \{T_2\}} \frac{|\phi(x, \mathbf{r}_1) \cap \phi(x, \mathbf{r}_2)|}{|\phi(x, \mathbf{r}_1) \cup \phi(x, \mathbf{r}_2)|}}{|\{T_1\} \cup \{T_2\}|} \tag{5}$$

where $\mathcal{C}_1 = \{(G_1, T_1) \mid (G_1, T_1) \text{ is a concept}\}$ and $\mathcal{C}_2 = \{(G_2, T_2) \mid (G_2, T_2) \text{ is a concept}\}$ are the collection of concepts extracted respectively from $\mathbf{r}_1$ and $\mathbf{r}_2$.

To better understand the meaning of this measure, we can see a toy example based on the tables Tab. 1b and Tab. 1b. Let $\mathcal{C}_b$ and $\mathcal{C}_c$ denote the collection of formal concepts extracted respectively from the boolean matrices in Tab. 1b and Tab. 1c (with a non empty set of genes and a non empty set of situations). The list of concepts contained in the two collections is:

$\mathcal{C}_b$	$\mathcal{C}_c$
$(G_{b1}, T_{b1}) = \{a, c, e\}, \{2, 4, 5\}$	$(G_{c1}, T_{c1}) = \{c\}, \{4, 5\}$
$(G_{b2}, T_{b2}) = \{b, f, g\}, \{1, 3, 5\}$	$(G_{c2}, T_{c2}) = \{b\}, \{1, 3\}$
$(G_{b3}, T_{b3}) = \{b, d, f, g\}, \{1, 3\}$	$(G_{c3}, T_{c3}) = \{b, d, f, g\}, \{1\}$
$(G_{b4}, T_{b4}) = \{a, c, e, h\}, \{4, 5\}$	$(G_{c4}, T_{c4}) = \{a, c, e, h\}, \{4\}$
$(G_{b5}, T_{b5}) = \{a, b, c, e, f, g, h\}, \{5\}$	

Clearly, only two sets of situations are shared by the two collections. They are $T_{b3} = T_{c2} = \{1, 3\}$ and $T_{b4} = T_{c1} = \{4, 5\}$. We get the following result:

$$Sim\left(\mathcal{C}_b, \mathcal{C}_c\right) = \frac{\frac{|G_{b3} \cap G_{c2}|}{|G_{b3} \cup G_{c2}|} + \frac{|G_{b4} \cap G_{c1}|}{|G_{b4} \cup G_{c1}|}}{7} = \frac{\frac{1}{4} + \frac{1}{4}}{7} = 0.07$$

Applying such a measure to our different collections gives the results collected in Tab. 2.

	Max -X%Max					
X %Max	*40*	*45*	*50*	*55*	*60*	*65*
2	0.009456	0.004353	0.001392	0.000412	0.000095	0.000016
5	0.147644	0.082908	0.028939	0.008899	0.002057	0.000334
8	0.093602	0.149451	0.146705	0.062045	0.017565	0.003033
10	0.033129	0.0663	0.131817	0.10268	0.039822	0.007915
15	0.003442	0.008034	0.026383	0.06342	0.097521	0.03868
20	0.000337	0.000792	0.0028	0.009689	0.035462	0.082248

Table 2. Self-similarity measures on different collections of concepts in SAGE data

Interestingly, the self-similarity values are relatively high in the intersection between the X values for which the "X% Max" method takes the highest similarity scores (**TScore**), and the X values for which the "Max -X% Max" method has the same behavior (see Fig. 7). We notice how the measures are usually very low (the highest one is about 0.15). It emphasizes the impact of the choice of a relevant discretization method. The relevancy of the extracted patterns is not only related to the preservation of some properties of the raw data set, but also tightly related to the specific biological problem at hand.

Comparing dendrograms resulting from the clustering of different types of derived boolean matrices enables to choose the "best" discretization method and parameters for a given data set. When looking at the average similarity scores for "Max - X% Max" and "X% Max" methods (see Fig. 7), we observe either an optimal value or an asymptotic behavior. It could mean that the best choice for the discretization threshold is a trade-off between the value for which we get the best similarity score and the value for which the data mining tasks remain tractable.

4 Robustness of the Measure

In Section 2, we proposed a method to assess gene expression property encoding. We refined the measure presented in [16] by defining an average similarity score which can take into account both gene and situation similarity scores. We now discuss the choice of the reference tree, and thus the choice of the clustering algorithm. Our idea is simple. If we apply a clustering algorithm with different parameters to the same gene expression matrix, and then compare all the resulting dendrograms using our method, the measures should be quite similar.

Even if there are methods that produce very similar results, and others that produce totally different results, the overall behavior of the measures should be identical, i.e., for each particular configuration of the clustering algorithm, the mean of the similarity scores obtained by comparing its resulting dendrogram with the dendrograms related to all the other configuration, should be high and should not differ too much from the means computed in the same way for the other configurations.

To perform the experiments, we have used the three datasets described in Section 2.3. Hierarchical clustering has been performed with the free software HCE 2.0 (Hierarchical Clustering Explorer) available on-line on the site of the Human-Computer Interaction Laboratory (University of Maryland)[1]. The used clustering metrics have been the classical Euclidean distance and the centered/uncentered Pearson's coefficients ([4]). Moreover, we used the four classical linkage methods (i.e., single, complete, average, average group linkage) and Shneiderman's 1-by-1 linkage method as well. For each data set, once the clustering process was completed, we have compared each of the resulting dendrograms with all the other dendrograms. This has been done for both gene and situation dendrograms. Due to space limitations, we provide only the average similarity scores for the Pearson's uncentered coefficient in the SAGE data set (see Tab. 3).

		Average Similarity Scores - Pearson's Uncentered				
Metrics	**Linkage**	*Average*	*Avg.Group*	*Complete*	*Single*	*Shneid.*
Pearson's Uncentered	*Average*	1	0.67314383	0.67284944	0.80330149	0.73423766
	Average Group	0.52915848	1	0.46868204	0.74420618	0.57334442
	Complete	0.72280557	0.64047742	1	0.76910562	0.65782797
	Single	0.37379950	0.44053048	0.33315260	1	0.38401040
	Shneiderman	0.69095298	0.68635693	0.57626332	0.77659575	1
Pearson's Centered	*Average*	0.73765387	0.63184005	0.57791403	0.76935144	0.63659514
	Average Group	0.51583859	0.71440599	0.71471718	0.73727445	0.58849284
	Complete	0.63213575	0.60668335	0.71471718	0.73727445	0.58849284
	Single	0.34417977	0.40501553	0.30670888	0.84271541	0.35339934
	Shneiderman	0.60198327	0.64004493	0.51441451	0.75143098	0.7112918
Euclidean	*Average*	0.22302538	0.26032110	0.21204947	0.34910165	0.22825697
	Average Group	0.22822833	0.26402535	0.20887047	0.34531201	0.23794469
	Complete	0.30246296	0.33102761	0.29277610	0.39226471	0.29859425
	Single	0.15444260	0.18272967	0.14310903	0.28929635	0.15970716
	Shneiderman	0.02970444	0.03884745	0.02795342	0.07367044	0.03641223

Table 3. Average similarity scores for clustering using Pearson's uncentered coefficient

Obviously, obtained values can be quite different. As expected, comparisons between "Pearson's coefficient" and "Euclidean distance" lead to rather low similarity scores. It is interesting to notice that comparisons with the single linkage method as reference leads to very high similarity scores. The same linkage method, compared with other references, give rise to rather low similarity scores. Our measure is indeed asymmetric.

We can compute the mean of the similarity scores obtained for each reference (see Fig. 9). The scores are always higher than the computed scores we got when comparing the dendrograms from the boolean matrices (see Section 2.3).

[1] http://www.cs.umd.edu/hcil/hce/

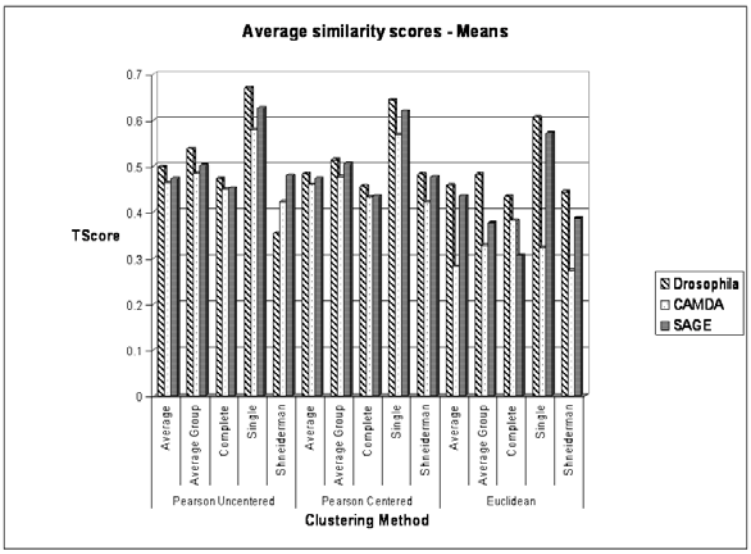


Fig. 9. Computed means of the average similarity scores for the three datasets

Finally, we have also considered the robustness of our metrics by looking at the overall behavior. For each data set, we have computed the mean of the measures shown in Fig. 9. To explain the content of this figure, let us remind all the steps of our analysis. First we have computed the similarity scores between all couples of computed dendrograms. Let T_i denotes the dendrogram resulting of a particular combination of clustering parameters ($i = 1..15$). Let S_{ij} denotes the similarity score computed between each couple of dendrograms T_i and T_j (T_i being the reference). Notice that in general $S_{ij} \neq S_{ji}$. In Fig. 9 we have the following values:

$$\overline{S}_i = \frac{\sum_{j=1}^{15} S_{ij}}{15}.$$

Let $\overline{S}_i^p$ denote the mean computed only on the dendrograms obtained by using the two Pearson's coefficient, i.e.,

$$\overline{S}_i^p = \frac{\sum_{j=1}^{10} S_{ij}}{10}.$$

For each data set, we are interested in the following measures:

$$\overline{S} = \frac{\sum_{i=1}^{15} \overline{S}_i}{15} \text{ and } \overline{S}^p = \frac{\sum_{i=1}^{10} \overline{S}_i^p}{10}.$$

Finally, we need to compute the standard deviations of the $\overline{S}_i$ and $\overline{S}_i^p$ values:

$$\sigma = \sqrt{\frac{\sum_{i=1}^{15} \left(\overline{S}_i - \overline{S}\right)^2}{15}} \text{ and } \sigma^p = \sqrt{\frac{\sum_{i=1}^{10} \left(\overline{S}_i^p - \overline{S}^p\right)^2}{10}}$$

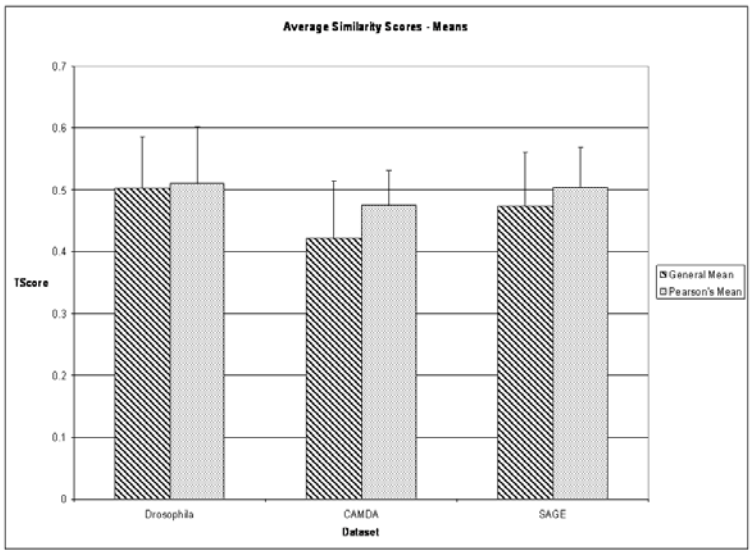


Fig. 10. Values of $\overline{S}$ and $\overline{S}^p$ and related standard deviations σ and σ^p

The final results are summarized in Fig. 10. Notice that the values of the means ($\overline{S}$ and $\overline{S}^p$) are near to 0.5 for every dataset, while the standard deviation is generally small. Both these observation make us conclude that the dendrogram resulting from the hierarchical clustering algorithm is a valid reference for our problem of comparing different method of gene expression property encoding. Moreover, the choice of the Pearson's correlation coefficient for the execution of the comparison (see Section 2.3), is shown to be adequate by the fact that the means computed only on the dendrograms obtained through this metrics ($\overline{S}^p$) are greater than the general means ($\overline{S}$), while the related standard deviations (σ^p) are similar or smaller than the general ones (σ).

5 Conclusion

We defined a new pre-processing technique that supports the evaluation and assessment of different discretization techniques for a given gene expression data set. The evaluation is based on the comparison of dendrograms obtained by clustering various derived boolean matrices with the one obtained on the raw matrix. The defined metrics is simple and we have validated its relevancy on different real data sets. A validation on a biological problem has been considered in [16]. This is a step towards a better understanding of a crucial pre-processing step when we want to apply the very efficient techniques based on set pattern mining from boolean data. Thanks to the exhaustive search for every pattern which satisfy the user-defined constraints, set pattern mining techniques like constraint-based mining of formal concepts appear to be complementary approaches to global pattern heuristic mining techniques like clustering.

Acknowledgements. The authors want to thank Céline Robardet, Sylvain Blachon and Olivier Gandrillon for the pre-processing of the SAGE data set, and Sophie Rome for her participation to microarray data preparation. Furthermore, we thank Claire Leschi and Jérémy Besson for their contribution to a preliminary version of this paper. Finally, this research is partially funded by ACI MD 46 (CNRS STIC 2004-2007) Bingo (Bases de Données Inductives pour la Génomique).

References

1. DeRisi, J., Iyer, V., Brown, P.: Exploring the metabolic and genetic control of gene expression on a genomic scale. Science **278** (1997) 680–686
2. Velculescu, V., Zhang, L., Vogelstein, B., Kinzler, K.: Serial analysis of gene expression. Science **270** (1995) 484–487
3. Piatetsky-Shapiro, G., Tamayo, P., eds.: Special issue on microrray data mining. SIGKDD Explorations, Volume 5, Issue 2 (2003)
4. Eisen, M., Spellman, P., Brown, P., Botstein, D.: Cluster analysis and display of genome-wide expression patterns. Proc. Natl. Acad. Sci. USA **95** (1998) 14863–14868
5. Niehrs, C., Pollet, N.: Synexpression groups in eukaryotes. Nature **402** (1999) 483–487
6. Boulicaut, J.F., Bykowski, A.: Frequent closures as a concise representation for binary data mining. In: Proceedings PAKDD'00. Volume 1805 of LNAI., Kyoto, JP, Springer-Verlag (2000) 62–73
7. Pei, J., Han, J., Mao, R.: CLOSET an efficient algorithm for mining frequent closed itemsets. In: Proceedings ACM SIGMOD Workshop DMKD'00, Dallas, USA (2000) 21–30
8. Zaki, M.J., Hsiao, C.J.: CHARM: An efficient algorithm for closed itemset mining. In: Proccedings SIAM DM'02, Arlington, USA (2002)
9. Becquet, C., Blachon, S., Jeudy, B., Boulicaut, J.F., Gandrillon, O.: Strong-association-rule mining for large-scale gene-expression data analysis: a case study on human sage data. Genome Biology **12** (2002)
10. Creighton, C., Hanash, S.: Mining gene expression databases for association rules. Bioinformatics **19** (2003) 79 – 86
11. Wille, R.: Restructuring lattice theory: an approach based on hierarchies of concepts. In Rival, I., ed.: Ordered sets. Reidel (1982) 445–470
12. Rioult, F., Boulicaut, J.F., Crémilleux, B., Besson, J.: Using transposition for pattern discovery from microarray data. In: Proceedings ACM SIGMOD Workshop DMKD'03, San Diego (USA) (2003) 73–79
13. Rioult, F., Robardet, C., Blachon, S., Crémilleux, B., Gandrillon, O., Boulicaut, J.F.: Mining concepts from large sage gene expression matrices. In: Proceedings KDID'03 co-located with ECML-PKDD 2003, Catvat-Dubrovnik (Croatia) (2003) 107–118
14. Besson, J., Robardet, C., Boulicaut, J.F.: Constraint-based mining of formal concepts in transactional data. In: Proceedings PAKDD'04. Volume 3056 of LNAI., Sydney (Australia), Springer-Verlag (2004) 615–624
15. Besson, J., Robardet, C., Boulicaut, J.F., Rome, S.: Constraint-based concept mining and its application to microarray data analysis. Intelligent Data Analysis journal **9** (2004) To appear.

16. Pensa, R.G., Leschi, C., Besson, J., Boulicaut, J.F.: Assessment of discretization techniques for relevant pattern discovery from gene expression data. In: Proceedings ACM BIOKDD'04 co-located with SIGKDD'04, Seattle, USA (2004) 24–30
17. Parthasarathy, S.: Efficient progressive sampling for association rules. In: Proceedings IEEE ICDM'02, Maebashi City, Japan (2002) 354–361
18. Moore, G.W., Goodman, M., Barnabas, J.: An iterative approach from the standpoint of the additive hypothesis to the dendrogram problem posed by molecular data sets. Journal of Theoretical Biology **38** (1973) 423–457
19. Robinsons, D.F.: Comparison of labeled trees with valency three. Journal of Combinatorial Theory, Series B **11** (1971) 105–119
20. DasGupta, B., He, X., Jiang, T., Li, M., Tromp, J., Zhang, L.: On distances between phylogenetic trees. In: Proceedings ACM-SIAM SODA'97. Volume 55. (1997) 427–436
21. DasGupta, B., He, X., Jiang, T., Li, M., Tromp, J., Zhang, L.: On computing the nearest neighbor interchange distance. In: Discrete mathematical problems with medical applications (New Brunswick, NJ, 1999), Providence, RI, Amer. Math. Soc. (2000) 125–143
22. Finden, C., Gordon, A.: Obtaining common pruned trees. Journal of Classification **2** (1985) 255–276
23. Cole, R., Hariharan, R.: An o(n log n) algorithm for the maximum agreement subtree problem for binary trees. In: Proceedings of the 7th annual ACM-SIAM symposium on Discrete algorithms, Atlanta, Georgia, United States (1996) 323–332
24. Bozdech, Z., Llinás, M., Pulliam, B.L., Wong, E., Zhu, J., DeRisi, J.: The transcriptome of the intraerythrocytic developmental cycle of plasmodium falciparum. PLoS Biology **1** (2003) 1–16
25. Arbeitman, M., Furlong, E., Imam, F., Johnson, E., Null, B., Baker, B., Krasnow, M., Scott, M., Davis, R., White, K.: Gene expression during the life cycle of drosophila melanogaster. Science **297** (2002) 2270–2275
26. Lash, A., Tolstoshev, C., Wagner, L., Schuler, G., Strausberg, R., Riggins, G., Altschul, S.: SAGEmap: A public gene expression resource. Genome Research **10** (2000) 1051–1060