ECML 2007 PKDD
WARSAW POLAND

THE 18TH EUROPEAN CONFERENCE ON MACHINE LEARNING
AND
THE 11TH EUROPEAN CONFERENCE ON PRINCIPLES AND PRACTICE
OF KNOWLEDGE DISCOVERY IN DATABASES

PROCEEDINGS OF THE 2ND WORKSHOP IN DATA MINING IN FUNCTIONAL GENOMICS AND PROTEOMICS

DMFGP'07

September 17, 2007

Warsaw, Poland

Editors:
A. Fazel Famili
National Research Council, Canada
Xiaohui Liu
Brunel University, UK
José-María Peña
Universidad Politécnica de Madrid, Spain

Preface

Data Mining in Functional Genomics and Proteomics involves a close collaboration between researchers from a number of diverse areas, such as biology, medicine, genomics and proteomics to computer science, mathematics and statistics. This collaboration of disciplines has evolved because of the: (i) advances that have occurred in data production and acquisition facilities, such as the introduction of mircroarrays and high throughput genomics and proteomics, (ii) enormous amounts of data that is generated every day that cannot be analyzed using ordinary data mining tools and techniques, and (iii) strong interest from many groups (research institutes, hospitals, academia, pharmaceuticals, etc.) who want to benefit from this wealth of data. Many efforts to deal with these issues are being undertaken by researchers working in this field. The aim of this workshop was to bring together researchers working on different topics related to data mining in functional genomics and proteomics. In particular we were interested to focus on current trends and emphasize on what should be the future directions for generic and applied research in this field. The main topics addressed during the workshop are integration methodologies for functional genomics and proteomics and also issues related to structuring and disseminating all useful knowledge that increasingly becomes available in this field.

Our call for papers resulted in some very interesting papers that are the contents of these workshop proceedings. The workshop was organized as part of ECML/PKDD-2007 conference. We are grateful for the support that we received from the organizers of this conference. In particular we would like to thank Dr. Marzena Kryszkiewics, ECML/PKDD 2007 Workshops Chair for accepting our proposal, the program committee and additional reviewers (Drs. Amira Djebbari, Edwin Wang, and Youlian Pan) who helped us for the review.

Warsaw, September 2007

A. Fazel Famili (Chair)
Xiaohui Liu (Co-Chair)
José-María Peña (Co-Chair)

Workshop Organization

DMFGP Workshop Chairs

A. Fazel Famili (National Research Council – Canada) (chair)
Xiaohui Liu (Brunel University – UK) (co-chair)
José-María Peña (Universidad Politécnica de Madrid – Spain) (co-chair)

ECML/PKDD Workshop Chair

Marzena Kryszkiewicz (Warsaw University of Technology)

Workshop Program Committee

Guillaume Beslon (LIRIS, INSA–Lyon – France)
Henrik Bostrom (Royal Inst. Of Technology – Sweden)
Joaquin Dopazo (CIPF – Spain)
Ana Teresa Freitas (INESC-ID/IST – Portugal)
Samuel Kaski (Helsinki University of Technology -– Finland)
Alexander Schliep (Max Planck Institute — Germany)
Sergio Storari (Università di Ferrara – Italy)
Evgenii Vityaev (Russian Academy of Science – Russia)

Table of Contents

Combining APRIORI and Bootstrap Techniques for Marker Analysis

Giacomo Gamberoni[1], Evelina Lamma[1], Fabrizio Riguzzi[1], Chiara Scapoli[2], and Sergio Storari[1]

[1] ENDIF, University of Ferrara, Italy
{giacomo.gamberoni, evelina.lamma, fabrizio.riguzzi, sergio.storari}@unife.it
[2] Department of Biology, University of Ferrara, Italy, scc@unife.it

Abstract. In genetic studies, complex diseases are often analyzed searching for marker patterns that play a significant role in the susceptibility to the disease. In this paper we consider a dataset regarding periodontitis, that includes the analysis of nine genetic markers for 148 individuals. We analyze these data by using a novel subgroup discovering algorithm, named APRIORI-B, that is based on APRIORI and bootstrap techniques. This algorithm can use different metrics for rule selection. Experiments conducted by using as rule metrics novelty and confirmation, confirmed some previous results published on periodontitis.

1 Introduction

In classical genetics [1], diseases are divided into Mendelian disorders and complex traits. While the former are attributed to single gene mutations with a simple mode of inheritance, the latter are thought to result from interaction among multiple genes. The main task in the study of these polygenic diseases is obviously to find the genetic patterns that increase susceptibility to the diseases.

In machine learning, such task is faced by using subgroup discovery techniques. Their goal is to find subgroups, represented by rules, which describe subsets of the population that are sufficiently large and statistically unusual with respect to a target attribute. This task is at the intersection of predictive and descriptive induction, and has been formulated in [2], [3], [4]. The problem can be expressed as follows: given a population and a single property of the individuals, find population subgroups that are statistically "most interesting". For example, we may look for groups that are as large as possible and on which the property of interest has a distribution that is as different as possible with respect to the distribution over the whole population. In the literature, several algorithms have been proposed for subgroup discovery (e.g. Explora [2], MIDOS [3], APRIORI-SD [5], CN2-SD [6]) and for classification rule learning (e.g. CBA [7]).

In this paper, we present a novel algorithm, named APRIORI-B, that performs subgroup discovery by combining APRIORI [8] and bootstrap techniques (more precisely the randomization test).

Our method uses APRIORI for finding frequent itemsets, and then generates rules from them. In the rule selection post-processing phase, it sorts the generated rules by using a rule evaluation metric. Then the most significant rules are selected by using the randomization test [9].

We verified the suitability of APRIORI-B for marker analysis by applying it on real biological data. In the experiment, we analyzed a dataset used by biologists to investigate the relation between nine genetic markers and periodontitis. For this biological dataset we provide some subjective evaluations of the subgroups identified.

This paper is organized as follows: Section 2 presents background information on APRIORI algorithm and methods for rule evaluation. Section 3 describes our algorithm. Section 4 illustrates the chosen case study: the analysis of genetic markers. Section 5 reports the results of applying our algorithm the genetic dataset. Finally, Section 6, presents conclusions and perspectives for future works.

2 Background

In subgroup discovery, subgroups can be modeled by classification rules. In this section, we first present association rules and then one of their special case, represented by classification rules (Section 2.1). Then in Section 2.2, we briefly describe the APRIORI algorithm [8] for association rule mining.

2.1 Association and classification rules

Association rules. Consider a table D having only discrete attributes. If D has also numeric attributes, they are discretized. An *item* is a literal of the form $A = v$ where A is an attribute of D and v is a value in the domain of A. Let M be the set of all the possible items. An *itemset* X is a set of items, i.e. it is such that $X \subseteq M$. A k-itemset is an itemset with k elements. We say that a record r of D *contains* an itemset X if $X \subseteq r$ or, alternatively, if r satisfies all the items in X. Let $n(X)$ be the number of records of D that contain X. Let $n(\overline{X})$ be the number of records of D that do not contain X. Let N be the number of records of D. The *support* of an itemset X (indicated by $Sup(X)$) is the fraction of records in D that contain X. i.e., $Sup(X) = n(X)/N$. It is also equal to the probability of a record of D of satisfying X, i.e. $p(X) = Sup(X)$. When X and Y are two itemsets we use the shorthand notation $n(XY)$, $Sup(XY)$ and $p(XY)$ to mean, respectively, $n(X \cup Y)$, $Sup(X \cup Y)$ and $p(X \cup Y)$.

Association rules are of the form $B \rightarrow H$ where B and H are itemsets such that $B \cap H = \emptyset$. B and H are respectively called *body* and *head*.

Classification rules. Classification rules are association rules whose head is of the form $Class = c$ where $Class$ is a special attribute of D. In this case, the records of D are also called *examples* and a rule $B \rightarrow Class = c$ covers a record r if $B \subseteq r$ and correctly covers a record if $B \cup \{Class = c\} \subseteq r$.

Notice that, for classification rules, a contingency table is a generalization of a confusion matrix, which is the standard basis for computing rule evaluation measures in binary classification problems. In the confusion matrix notation, $n(H)$ is the number of positive examples, $n(\overline{H})$ the number of negative examples, $n(B)$ is the number of examples covered by the rule therefore predicted as positive, $n(\overline{B})$ is the number of the examples not covered by the rule and therefore predicted as negative, $n(BH) = TP$ is the number of true positives, $n(\overline{BH}) = TN$ is the number of true negatives, $n(B\overline{H}) = FP$ is the number of false positives, and $n(\overline{B}H) = FN$ is the number of false negatives.

Rule metrics For association and classification rules a number of quality metrics can be defined. All rule evaluation measures are defined in terms of frequencies from the *contingency table* only (see Table 1).

Table 1. A contingency table.

	Body		
Head	B	$\overline{B}$	
H	$n(HB)$	$n(H\overline{B})$	$n(H)$
$\overline{H}$	$n(\overline{H}B)$	$n(\overline{H}\overline{B})$	$n(\overline{H})$
	$n(B)$	$n(\overline{B})$	N

Given a rule $R = B \rightarrow H$, we define the following metrics:

- Support: $Sup(R) = p(BH) = Sup(BH) = \frac{n(BH)}{N}$
- Confidence: $Conf(R) = p(H|B) = \frac{Sup(BH)}{Sup(B)} = \frac{n(BH)}{n(B)}$
- Novelty: $Nov(R) = p(HB) - p(H)p(B)$
- Confirmation: $Confirmation(R) = \frac{p(BH)-p(B)p(H)}{\sqrt{p(B)p(H)p(\overline{B})p(\overline{H})}}$

Support and Confidence are classical association and classification rule metrics. Novelty [10] and Confirmation [11] are examples of more complex rule evaluation metrics [12], and we choose to focus the experiments described in this paper on them.

The definition of novelty states that we are only interested in high support if that could not be expected from the marginal probabilities, i.e., when $p(H)$ and/or $p(B)$ are relatively low. It can be demonstrated that $-0.25 \leq Nov(R) \leq 0.25$: a strongly positive value indicates a strong association between H and B, while a strongly negative value indicates a strong association between $\overline{H}$ and B.

2.2 APRIORI

The task of discovering association rules consists in finding all the association rules having a minimum support *minsup* and a minimum confidence *minconf*.

In order to discover such rules, the approach proposed in [8] first discovers all the itemsets with support higher than *minsup* and then finds the rules from them. The itemset with support above *minsup* are called *large*. The part of APRIORI that finds large itemsets is shown in Figure 1. Figure 2 shows function apriori-gen that is used by APRIORI.

Notation:L_k, set of large k-itemset

1. $L_1 = \{$ large 1-itemsets $\}$
2. for(k=2; $L_{k-1} \neq \emptyset$; $k++$) do begin
3. C_k=apriori-gen(L_{k-1}); // new candidates
4. forall records $r \in D$ do begin
5. C_r = subset(C_k, r); // candidates contained in r
6. forall candidates $c \in C_r$ do
7. $c.count++$
8. end
9. $L_k = \{c \in C_k | (c.count/size(D)) > minsup\}$
10. end
11. $Answer = L = \bigcup_k L_k$

Fig. 1. Algorithm APRIORI

// Phase 1
Insert into C_k
Select $p.\text{item}_1$,$p.\text{item}_2$,...,$p.\text{item}_{k-1}$,$q.\text{item}_{k-1}$
From L_{k-1} p, L_{k-1} q
Where ($p.\text{item}_1$=$q.\text{item}_1$) and ... and
($p.\text{item}_{k-2}$=$q.\text{item}_{k-2}$) and ($p.\text{item}_{k-1} < q.\text{item}_{k-1}$)
// Phase 2
forall itemset $c \in C_k$ do
forall $(k-1)$-subsets s of c do
if $s \notin L_{k-1}$ then
Delete c from C_k

Fig. 2. Function apriori-gen

APRIORI is based on the fact that $X \supseteq Y \rightarrow Sup(X) \leq Sup(Y)$. Therefore if $Sup(X) < minsup$ then $\forall Y \supseteq X,\ Sup(Y) < minsup$. So we can discard every itemset that has a non large subset.

3 APRIORI-B algorithm

APRIORI-B performs subgroup discovery by learning in several steps a set og classification rules. Given a dataset D, it:

1. removes the $Class$ attribute from D, obtaining $D_{noclass}$;
2. uses APRIORI (described in Section 2.2) on $D_{noclass}$, to obtain the set of large itemsets L;
3. for each itemset $B \in L$ and for each item $H = \{Class = c\}$ where c is a value of the $Class$ attribute, builds the rule $R = B \rightarrow H$;
4. for each rule, computes the rule score metric;
5. sorts rules (in descending order of the metric) and filters them (using a lower bound on the metric $minmetric$ and a maximum number of rules $maxrules$),
6. evaluates the p-value of each rule, by using the randomization test described in Section 3.1.
7. filters the rules, considering a p-value threshold, and obtains the final rule set RS.

APRIORI-B allows the use of several rule evaluation metrics. For the experiment performed in this paper, we used novelty and confirmation (defined in Section 2.1).

Our algorithm is very close to CBA [7] but while CBA uses APRIORI for identifying classification rules with a minimum support, APRIORI-B uses APRIORI in the first learning phase for finding itemsets with a minimum support that are then used as classification rule bodies. Another difference is the following: our algorithm does not aim to build a classifier. Its goal is to find a set of rules that can highlight relations between attributes and the class.

Moreover, one of the main distinguishing features of APRIORI-B is the use of randomization test. The main advantage of using this approach for rule selection is that we obtain immediately a p-value for each rule. This can be very useful to assess rules significance.

3.1 Randomization test

In order to select only the rules having a significant value for the considered metric, we performed a randomization test [9].

First of all, we generated 1000 shuffled dataset, starting from the original one, by independently shuffling the values inside each column. In this way we obtained datasets with the same probabilities for each attribute values but without relations between them. This step was performed before the dataset preparation described in Section 5.1.

We used APRIORI for obtaining the rules, and sorted them using the value of the metric. Then we re-computed the metric for each of the learned rules using all the 1000 shuffled dataset. In this way, for each rule, we obtained a statistical distribution of its metric (i.e. we computed the mean and standard deviation of the metric). By comparing the value of the metric computed by using the original dataset with this distribution, we can assess the significance value of a rule (we considered the values to have a normal distribution).

4 The case study: Marker Analysis

Most common diseases are complex genetic traits [1], where multiple genetic and environmental variables contribute to the observed traits. Because of the multifactorial nature of complex traits, each individual genetic variant (susceptibility allele[1]) generally has only a modest effect, and the interaction of genetic variants with each other or with environmental factors can potentially be quite important in determining the observed phenotype[2]. Genetic association studies, in which the allele or genotype[3] frequencies at markers are determined in affected individuals and compared with those of controls (case-control study design), may be an effective approach to detecting the effects of common susceptibility variants.

The most abundant source of genetic variation in the human genome is represented by single nucleotide polymorphisms (SNPs). SNPs can identify common, but minute, variations that occur when a single unit in a genome sequence (nucleotide) is altered. These variations can be used to track inheritance in families.

Eleven million SNPs of greater than 1% frequency are estimated to exist in the genome and the International HapMap Project has as a primary goal the identification of appropriate sets of tag SNPs that span the genome. These tag SNPs may be able to capture most of the common genetic variants contributing to complex human disease.

At the moment, studies and algorithms able to identify non-random correlations between alleles at a pair of SNPs, have been discussed as a general approach to determine multiple locus involved in human chronic diseases with a genetic component. Moreover, a quantity of "tagging" algorithms for selecting minimum informative subsets of SNPs has recently appeared in the literature.

4.1 Experimental Dataset

As an example of complex genetic trait, we choose Generalized Aggressive Periodontitis (GAP) as case study. Periodonditis is a dental disorder that results from progression of gingivitis, involving inflammation and infection of the ligaments and bones that support the teeth.

The dataset, provided by the Research Center for the Study of Periodontal Diseases, University of Ferrara, collects data from 46 GAP patients (16 males and 30 females) and 102 periodontally healthy control subjects. All subjects were chosen amongst current and permanent residents of the city of Ferrara area. Systemically healthy GAP patients were selected for study among those undergoing periodontal supportive therapy at the Research Center for the Study of Periodontal Diseases, University of Ferrara, and the diagnoses were confirmed by

[1] Allele: one of several alternative form of a gene or DNA sequence at a specific chromosomal location (locus). At each locus an individual possesses two alleles, one inherited from the father an one from the mother.

[2] Phenotype: the observable attribute(s) of a cell or an individual, brought about by the interaction of genotype and environment.

[3] Genotype: the specific allelic composition of an organism or cell.

the same clinician. The clinical diagnosis at the time of the initial visit was based on recent international classification [13]. The periodontally healthy control subjects were selected if they showed no interproximal attachment loss greater than 2 mm at any of the fully erupted teeth. Controls were matched by age and sex with GAP patients. All GAP patients and controls were Caucasian Italian. The study design was approved by the local ethical and written informed consent was provided by all participants in line with the Helsinki Declaration before inclusion in the study.

The following variants in the IL-1 gene cluster have been tested: IL-1α^{+4845} (recorded as $M1$), IL-1β^{+3953} ($M3$), IL-1β^{-511} ($M2$) and also the minisatellite of IL-1RN intron 2 ($M5$). Furthermore, it has been tested a new marker variant at the IL-1F5 ($M6$) gene as described in Scapoli et al. [14]. Besides polymorphisms at IL-1 cluster, other markers have been tested in different pro-inflammatory cytochine such as IL-6 (variant IL-6^{-174} ($M8$) and IL-6^{-622} ($M7$)) and TNF-A (variant TNF-α^{-308} ($M4$)). Finally also a polymorphism at the TNF-α receptor has been tested (TNFRSF1β^{+196} ($M9$)).

4.2 Related Studies

Several studies have shown a role for the involvement of interleukin-1 (IL) gene cluster polymorphisms in the risk of periodontal diseases. In [15] the authors tested polymorphisms, derived from genes of the IL1 cluster, for association with generalized aggressive periodontitis (GAP) through both allelic association and by constructing a Linkage Disequilibrium map of the 2q13-14 disease candidate region. For the IL-1RN intron 2 ($M5$), a statistically significant difference was found between patients and controls in the genotypic distribution, but no significant difference was found for allelic distribution. Authors also observed some evidence for an association between GAP and the IL-1β^{+3953} ($M3$) polymorphism.

For the other IL-1 Cluster polymorphisms, no significant differences were found between patients and controls for both genotypic and allelic frequencies.

Moreover, in [16], the authors showed that allele 1 of the IL-1β^{+3953} ($M3$) and allele 1 of the IL-1RN intron 2 ($M5$) in combination were significantly elevated in GAP as compared to controls.

5 Experiments

5.1 Results on GAP dataset

Dataset preparation The application of the algorithms for subgroup discovery on genetics dataset was performed by an examiner who was blinded as to the correspondence of the M1, M2,. . . ,M9 variables and the related polymorphisms, so that the examiner had not information on previous statistical analyses and on the expected results about IL-1β^{+3953} (M3), IL-1RN (M5) and TNFRSF1β^{+196} (M9) markers and the disease status.

Starting from the blinded dataset originated from the GAP study, we obtained a new dataset on which we ran the experiments. In the original dataset, each marker can assume three possible values: 11, 12 and 22. 11 and 22 are homozygote subjects while 12 define the heterozygote status. As an example, if there are two markers $(M1, M2)$ a possible record of the dataset is $(11, 12)$. In our analysis we consider the configuration of a single chromosome and we want to test, for each marker, whether the allele on that chromosome is 1 or 2. For heterozygote individuals, we do not know on which chromosomes lies the 1: in other words, the allelic configuration for the marker on the two chromosomes could be 12 or 21 with equal probability. The new dataset will contain, for each record from the original dataset all possible configurations of a single chromosome (haplotype) compatible with the record. Therefore, for each record in the original dataset, we generate 2^k tuples in the new dataset, where k is the number of marker analyzed. For example, in the case of the record above, the new dataset will contain the four tuples: $(1, 1)$, $(1, 2)$, $(1, 1)$ and $(1, 2)$.

Results: The dataset obtained (as described in the previous section) was analyzed by using APRIORI-B with two different rule metric, Novelty and Confirmation. The algorithm was configured with the following parameters: *minsup* set to 0.3, *minmetric* set to 0, *maxrule* set to 100 and p-value threshold set to 0.01 .

Rule learned by APRIORI-B using Novelty are shown in Table 2. For each learned rule, the table shows:

- Rule Body, the body of a learned rule containing a conjunction of $Marker = Allele$ tests ;
- State, the disease state associated to the conjunction of $Marker = Allele$ tests in the Rule Body;
- Novelty, the novelty metric value for the rule;
- Rand. Mean, the mean of the novelty values found in the 1000 randomized datasets for the classification rule under analysis;
- Rand. Std, the standard deviation of the Novelty values found in the 1000 randomized datasets for the classification rule under analysis;
- p-value, the rule p-value.

Rules learned by APRIORI-B using Confirmation have not been reported as they are the same learned in the experiment conducted with Novelty even if in a slightly different order.

Analyzing these results, we noticed that some of the rules are related to the two markers that have been reported in literature as involved in the pathology: M3 and M5. The expert confirmed that the correlation between the combination of M3 and M5 found in rule 3 is confirmed by literature [16]. The role of M9 and the combination between M8 and M9, and between M1 and M9 needs further biological investigations.

Table 2. Rule learned by APRIORI-B using Novelty

#	Rule Body	State	Novelty	Rand. Mean	Rand. Std	p-value
1	M9=1	GAP	0.0498	-0.0006	0.0154	0.000530
2	M5=1 M9=1	GAP	0.0408	-0.0006	0.0153	0.003379
3	M3=1 M5=1 M9=1	GAP	0.0383	0.0000	0.0127	0.001288
4	M3=1 M9=1	GAP	0.0377	-0.0001	0.0131	0.001953
5	M8=1 M9=1	GAP	0.0323	-0.0005	0.0127	0.005038
6	M1=1 M9=1	GAP	0.0310	-0.0001	0.0130	0.008699
7	M1=1 M5=1 M9=1	GAP	0.0305	-0.0001	0.0127	0.008153

6 Conclusion And Future Work

In this paper we described a novel algorithms for subgroup discovery named APRIORI-B. This algorithm is based on APRIORI for large itemset generation and randomization test for rule selection.

We developed this algorithm in order to study data obtained from marker analysis. APRIORI-B performance has been evaluated on a real dataset about generalized aggressive periodontitis, and the learned rules were judged interesting by the biologist.

Given this set of rules, further investigation could be made identifying the group of patients which present the marker combination specified by one of the rules. The comparison of the clinical state of these patient groups can be useful to conduct a more specific study of the disease (e.g. finding different disease phenotypes). This will be matter of future works. Moreover, a new dataset about sclerosis will be analyzed.

7 Acknowledgments

This work has been partially supported by NOEMALIFE under the “SPRING” regional PRRITT project, by the PRIN 2005 project “Specification and verification of agent interaction protocols” and by the FIRB project “TOCAI.IT”.

References

1. Lewin, B.: Genes VII. Oxford University Press, Oxford (2000)
2. Klösgen, W.: Explora: A multipattern and multistrategy discovery assistant. In: Advances in Knowledge Discovery and Data Mining. AAAI/MIT (1996) 249–271
3. Wrobel, S.: An algorithm for multi-relational discovery of subgroups. In Komorowski, H.J., Zytkow, J.M., eds.: PKDD. Volume 1263 of Lecture Notes in Computer Science., Springer (1997) 78–87
4. Wrobel, S.: Inductive logic programming for knowledge discovery in databases. In Dzeroski, S., Lavrac, N., eds.: Relational Data Mining. Springer (2001) 74–101

5. Kavsek, B., Lavrac, N., Jovanoski, V.: Apriori-sd: Adapting association rule learning to subgroup discovery. In Berthold, M.R., Lenz, H., Bradley, E., Kruse, R., Borgelt, C., eds.: IDA. Volume 2810 of Lecture Notes in Computer Science., Springer (2003) 230–241
6. Lavrač, N., Kavček, B., Flach, P., Todorovski, L.: Subgroup discovery with CN2-SD. Journal of Machine Learning Research **5** (2004) 153–188
7. Liu, B., Hsu, W., Ma, Y.M.: Integrating classification and association rule mining. In: KDD-98. (1998) 80–86
8. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules. In Bocca, J., Jarke, M., Zaniolo, C., eds.: Proceedings of the 20th International Conference on Very Large Data Bases (VLDB'94), Morgan Kaufmann (1994) 487–499
9. Efron, B., Tibshirani, R.: An Introduction to the Bootstrap. Chapman & Hall, London (1993)
10. Lavrac, N., Flach, P., Zupan, B.: Rule evaluation measures: A unifying view. In Dzeroski, S., Flach, P.A., eds.: ILP. Volume 1634 of Lecture Notes in Computer Science., Springer (1999) 174–185
11. Flach, P.A., Lachiche, N.: Confirmation-guided discovery of first-order rules with tertius. Machine Learning **42**(1/2) (2001) 61–95
12. Geng, L., Hamilton, H.J.: Interestingness measures for data mining: A survey. ACM Comput. Surv. **38**(3) (2006) 9
13. Tonetti, M.S., Mombelli, A.: Early-onset periodontitis. Ann Periodontol **4** (1999) 39–53
14. Scapoli, C., Tatakis, D., Mamolini, E., Trombelli, L.: Modulation of clinical expression of plaque-induced gingivitis: interleukin-1 gene cluster polymorphisms. J Periodontol **76** (2005) 49–56
15. Scapoli, C., Trombelli, L., Mamolini, E., Collins, A.: Linkage disequilibrium analysis of case-control data: an application to generalized aggressive periodontitis. Genes Immun **6** (2005) 44–52
16. Parkhill, J.M., H.B.C.I.H.P., Taylor, J.: Association of interleukin-1 gene polymorphisms with early-onset periodontitis. J Clin Periodontol **27** (2000) 682–689

Discovering informative genes from gene expression data: A multi-strategy approach

Fazel Famili[1], Sieu Phan[1], Ziying Liu[1], Youlian Pan[1], Amira Djebbari[1], Anne Lenferink[2], and Maureen O'Connor[2]

[1]Institute for Information Technology, NRC, Ottawa Ontario, K1A 0R6, Canada;
[2]Biotechnology Research Institute, NRC, Montreal, Quebec, H4P 2R2, Canada
{fazel.famili, sieu.phan, ziying.liu, youlian.pan, amira.djebbari, anne.lenferink, maureen.oconnor}@nrc-cnrc.gc.ca

Abstract. This paper discusses the issue of dealing with large volume of high throughput genomics data and applying unsupervised multi-strategy methods to identify differentially expressed genes. We introduce a novel method that consists of 8 steps. The approach is applied to a set of genomics data obtained from a cancer research study. We demonstrate the effectiveness of our method which includes some validation using biological experiments and literature search.

Keywords: Gene expression data analysis, Multi-strategy learning, Data mining and knowledge discovery.

1 Introduction

Discovering useful, and ideally, all previously unknown knowledge from historical or real-time data obtained from various sources, such as biological experiments or clinical information, is a complex and challenging task. This first requires an in-depth understanding of the domain and second the development of novel and appropriate strategies for data preprocessing and analysis. In high throughput genomics applications, knowledge discovery processes support various research and development activities. Two examples of these are: (*i*) discovering relationships between genes and their functions based on time-series data (such as drug response over time or developmental stages), and (*ii*) investigating gene responses to various treatments at one discrete time point. Many different data mining approaches have been developed and successfully applied to biological datasets. However, a method suitable for analyzing one dataset may not be successful when used in another dataset. It is well-recognized that different methods for the identification of differentially-expressed genes produce different lists of genes. This has motivated many researchers to apply several techniques, instead of one. Among the questions in that case would be: how to properly combine the results generated from all methods without losing any useful information. The objective of this paper is therefore to address this question.

In this paper we provide an overview of knowledge discovery in genomics and emphasize on multi-strategy approaches in which a number of unsupervised learning methods are applied to identify differentially-expressed genes from a given dataset. The following sections consist of a brief summary of some related works followed by the detail of the biological problem which motivated us to consider the multi-strategy approach. We then describe our method, provide the results of applying the multi-strategy method on a breast cancer related dataset and conclude.

2 Related work

To properly relate our work to previous research in machine learning and knowledge discovery in genomics, we have looked at several areas. First, this research is related to multi-strategy methods in both supervised and unsupervised learning, of which the ensemble approach is well-known [5]. There is also an extensive research on bagging and boosting [5] that is related to our research and is an example of this work. Second, it overlaps with feature selection based on multiple methods. And third, there are application specific papers that have some commonalities with our work. Following is a brief overview of some related works.

Supervised methods, which are mainly concept learners, generate hypotheses that are based on the original set of attributes. In many learning applications, the original learning space becomes inadequate. This inadequacy becomes evident through a high degree of irregularity in the distribution of instances and the models that are generated as output. Bloedorn *et al* [2] have developed a methodology to apply multiple learners and a range of strategies for an automated improvement of the knowledge representation space. A system like AQ-17 [2] has been able to significantly extend the machine learning capabilities as a multi-mechanism approach and produce a new generation of symbolic learning system. Hsu *et al* [8] proposed a high level optimization system (in the form of a wrapper) for relevance determination and constructive induction, and on integrating these wrappers with elicited knowledge on attribute relevance and synthesis. Their approach is based on using decision support systems when multi-strategy machine learning is applied. Similarly Geurts *et al* [6] proposed a new tree-based ensemble method for supervised classification and regression problems. This approach consists of randomizing both attribute and cut-point choice while splitting a tree node. In the extreme case, they build totally randomized trees whose structures are independent of the output values of the learning sample.

Similar efforts are seen in applying unsupervised methods for multi-strategy learning. Amershi and Conati [1] outline a user modeling framework that uses both unsupervised and supervised machine learning in order to reduce development costs of building user models, and facilitate transferability. They apply a framework to model student learning during interaction with the Adaptive Coach for Exploration (ACE) learning environment (using both interface and eye-tracking data). Learning from cluster examples (LCE) [10] is a hybrid task combining features of two common grouping tasks: learning from examples and clustering. In this approach, each training example is a partition of objects.

The objective is then to learn from a training set, a rule for partitioning unseen object sets. A general method for learning such partitioning rules is useful in any situation where explicit algorithms for deriving partitions are hard to formalize, while individual examples of correct partitions are easy to specify. In the past, clustering has been applied to such problems, despite being essentially unsuited. Multi-clustering is an example that has qualitative advantages over standard clustering when applied to vector-data images.

Of the most relevant research in feature selection based on biological data is the comparison and evaluation of ten different feature selection methods by Jeffery *et al* [9]. The authors applied all methods to nine microarray datasets where these methods returned dissimilar gene lists. From these datasets, only 8-12% of the genes listed by these methods were common. Along the same line Diaz-Uriarte and Alvarez de Andres [4] investigate the use of random forest for classification of nine microarray datasets and propose a new method for gene selection based on random trees and bootstrapping that produces relatively small gene lists. They also compare their approach with a number of other classification methods reported in the literature.

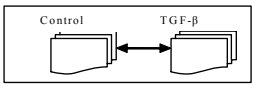
3 The Biological problem and data used for this study

The dataset used in this paper was generated by exposing a mouse mammary tumor cell line, the JM01 cell line [11], for 24 hours to a treatment with the Transforming Growth Factor (TGF-β). TGF-β induces an Epithelial-to-Mesenchymal Transition (EMT) in these cells, a phenomenon characterized by significant morphology and motility changes, which are thought to be critical for tumor progression. TGF-β can act both as a tumor suppressor and tumor promoter depending on the context in which it is expressed. Given the opposite actions of TGF-β and the multiplicity of effectors that this growth factor utilizes, it is essential to identify those mediators that are specific to its tumor promoting or tumor suppressive pathways. Elucidation of the genetic programs underlying this EMT should provide a better understanding of the molecular mechanisms involved in cancer development and progression.

The goal of this study is to identify the TGF-β modulated genes involved in the EMT process. This should result into identifying breast cancer specific biomarkers that can lead to a better screening and the development of novel drug and therapeutics targeted towards personalized medicine. Four experiments were performed each consisting of 4-6 replicates. The transcriptome changes by TGF-β were monitored using University Health Network (UHN) 15.6K mouse cDNA array platform. For demonstration of methodology, this paper is devoted to the analysis of one of the four experiments: TGF-β vs. Control (Fig. 1).

4 The multi-strategy approach

This section provides an overview of the multi-strategy approach (Fig. 2). Microarray data are first passed through a basic data preprocessing stage such as normalization, data



	TGF-β vs. Control at 24h					
probe	R1	R2	R3	R4	R5	R6
1						
2						
...						
15600						

Fig. 1. The structure of our data and the biological experiment

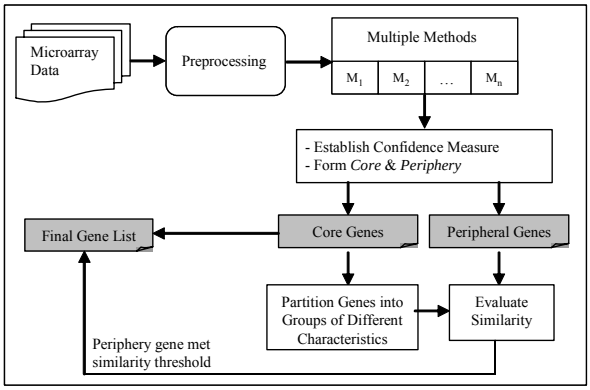


Fig. 2. Microarray data analysis and multi-strategy approach

filtering, and missing data handling. Certain steps in data preprocessing require domain expertise and additional research. Data preprocessing helps us to better envision the scope of knowledge discovery and to ascertain whether or not the experiments have been performed properly. The next step in multi-strategy approach is to apply as many data analysis methods as desired to obtain the best possible lists of the most informative genes for the biological experiment under consideration. The gene lists obtained from all methods are then consolidated, based on a novel algorithm that is one of our contributions in this study.

The overall consolidation algorithm is summarized in Fig. 3. After obtaining the gene lists from different analysis methods, the first step is to establish a confidence measure to select from these gene lists a set of genes to form the *core* of our final selection. The remainder of the genes forms the *periphery* which is subject to exclusion or inclusion into the final selection as described below. Depending on the context of the problem under study, there is a variety of ways to define the confidence measure. A simple confidence measure could be defined as some kind of voting scheme. Under a unanimity voting

scheme, the *core* consists of genes that are identified by all methods. One could also opt to define a less stringent voting strategy by selecting the *core* as the genes that are selected by more than one method.

1) Apply n methods that generate n lists of genes: $L_1, L_2, ...L_n$
2) Define confidence measure, CM
3) Form *core*, L_{core}, according to CM. For CM defined as the less stringent voting strategy, we have
$$L_{core} = \bigcup_{i=1}^{n-1} \bigcup_{j=i+1}^{n} (L_i \bigcap L_j)$$
4) Form *periphery*, $L_{periphery}$, according to CM:
$$L_{periphery} = \bigcup_{i=1}^{n} L_i - L_{core}$$
5) Partition L_{core} into different characteristic groups $G = \{G_1, G_2, ...G_m\}$
6) Evaluate the similarity between genes in $L_{periphery}$ and each group in G
7) Recruit genes from $L_{periphery}$ into $L_{recruited}$ if the genes in $L_{periphery}$ meet the similarity criteria
8) Build final gene set $L_{final} = L_{core} \bigcup L_{recruited}$

Fig. 3. Consolidation algorithm

The next step is recruitment of similar genes in the *periphery* into *core*. This is done based on the principle of characteristic similarity such as gene co-regulation, pathways involved, and Gene Ontology (GO) [15] annotation. In order to achieve this, we first partition the genes in *core* into different characteristic groups. The genes in each group could:

i. participate in the same biological pathway (based on, for example, KEGG database [16]),
ii. have the same biological function (based on GO annotation), or
iii. be regulated by the same mechanism (based on common transcription factors)

We then evaluate the similarity of the genes in *periphery* to the characteristics of each group in *core*. If a gene in the peripheral region passes the pre-established similarity threshold, this gene is recruited into the final gene list.

The effectiveness of the proposed methodology is demonstrated through its application to the JM01 dataset (Section 3). After data preprocessing, a set of data analysis methods are applied to search for informative genes through various significance measures in gene

expression profiles. Each method produces a different gene list. In this paper, we applied Rank Products (RP) [3], SAM [13] and t-test [9] to identify our list of differentially expressed genes from the JM01 dataset. In our experiments for RP, the expected RP-values and False Discovery Rate (FDR) were calculated using 100 random experiments (number of permutations) of the same size of the original dataset. We selected genes based on the 5% of false discovery rate. As for SAM, a one-class response was applied to identify the genes which were highly over- or under-expressed in TGF- treatment vs. control. The false discovery rate for SAM was 5% and the analysis was based on 100 random permutations. For t-test, the cut point is $p \leq 0.05$. To form the *core*, we selected genes that were identified by more than one method. The remaining genes that were identified by only one individual method fall into *periphery*. We used the DAVID annotation tool [7] to partition the genes in the *core* based on similar characteristics, such as biological pathway, biological function, sub-cellular location, protein domain, and gene regulation mechanism. A $p \leq 0.05$ threshold was applied to each annotation in the enrichment analysis. We used 0.35 for the grouping similarity threshold (*S*) and evaluated the similarity between the genes in the *periphery* and each characteristic group in the core region. If the similarity was $\geq S$, we recruit the genes into the final gene list.

5 Results

Fig. 4 shows the summary of the list of differentially expressed genes identified by SAM, t-test and Rank Products for up and down modulation, *core* and *periphery*. There are 554 and 359 genes in *core* and *periphery*, respectively. The results of partitioning the *core* and the recruitment are given in Table 1. We should mention that the recruitment is based on known genes only. There are 164 known genes in the *core* which belong to 13

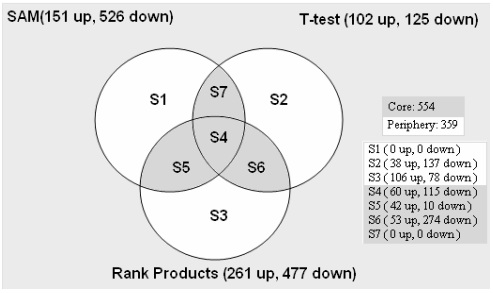


Fig. 4. Summary of number of genes identified by each method for up and down modulation, and number of genes in *core* (S4+S5+S6+S7) and *periphery* (S1+S2+S3).

Table 1. Summary of functional groups and genes recruited into final set from *periphery* that meet the similarity criteria ($\boldsymbol{p} \leq 0.05$, $\boldsymbol{S} \geq 0.35$) into the final gene set.

Functional group IDs	Description	Core*	Periphery*
1	Grown inhibition	7	2
2	Cell structure	8	4
3 & 12	Cell Metabolism	6+13	0
4 & 11	ATP	24+4	15+0
5	Mitochondrial	7	9
6 & 10	DNA binding &Replication	48+25	45+10
7	Phosphatases	5	4
8 & 9	Peptidases and Proteases	5+4	21+0
13	Membrane	8	17
Sum		**164**	**127**

* The + sign corresponds to the & in the first column, i.e. for DNA binding and replication, 48 genes form group 6 and 25 gene form group 10. Similarly, 45 and 10 genes are recruited based on characteristic of group 6 and 10 from periphery, respectively.

functional groups. From the *periphery,* 127 known genes meet the similarity threshold of the functional groups in the *core* and have been recruited into the final gene list.

Our results indicate that this consolidation method performs better than any single participating method and provides a stronger confidence in the results. This method is capable of identifying additional genes that would have been missed if only one method were used without incurring too much false identification. For example, the proposed methodology has helped us to uncover an important breast cancer gene, FGFR2 that would have been missed if the multi-strategy approach were not applied. FGFR2 has recently been identified as a biomarker in a genome-wide study conducted by several international teams [14]. Exploration on how well the consolidated gene list performs in constructing gene networks and by comparing to the gene lists produced by the individual methods is under study. The experimental validation of the identified genes is our next task in this research.

Table 2 shows the final gene list for up and down modulated genes identified by multi-strategy approach. Initially, there were 155 up, and 399 down modulated genes in the *core.* Through the recruitment from the *periphery*, 23 up and 40 down modulated

Table 2. Final gene list for up and down modulated genes.

Identified genes	TGF-β vs. Control	
	Up	Down
Core (# of genes)	155	399
# of added genes (pathway)	23	40
# of added genes (annotation)	27	65
Final gene list (not including overlapped genes)	**204**	**480**

genes were brought in according to functional annotation clustering. In addition, 27 up and 65 down modulated genes were brought in according to pathway information. The final list contains 204 up and 480 down modulated genes, which did not include the overlapped genes.

To demonstrate the effectiveness of our recruitment strategy, Table 3 and 4 show the genes that are recruited from *periphery* and their importance based on biological evidence. These genes are either related to morphology and mortality changes, e.g. beta-catenin (Ctnnb1) and CXCR4, or alterations in the cell cycle, e.g. Cyclin I (Ccni) and Cyclin D2 (Ccnd2), which are important in cancer development and progression. Some of the genes are known as breast cancer related or involved in EMT/TGF-β signaling pathway such as

Table 3. Important genes that are brought in from *periphery*: morphology and motility related.

Gene	Relevance
Up-modulated genes	
Actg1	Reorganized during EMT; cytoskeletal
Ctnnb1	Translocated in EMT; cytoskeletal
Cxcr4	Important role in chemotaxis cancer cells and tumour metastasis
Flna	Cytoskeletal function
Flnb	Cytoskeletal function
Fn1	Cell adhesion, extracellular matrix
Itgb1	Reported to be involved in TGF-β induced EMT
Itm2b, Ralb	
Jarid1b	Embryonic development
Map1lc3b	Cytoskeletal reorganization
Msn	Involved in actin filament/plasma membrane interaction that is regulated by Rho
Pard3	Reported to be involved in TGF-β induced EMT NMuMG cells
Pdgfc	Structurally more similar to VEGF-A than to PDGF-B
Rhob	Cytoskeletal reorganization
Sdc3	Cell-cell interaction regulating heparan sulfate proteoglycans
Sgce, Tmem59	
Sgk	Close homolog Akt; phosphorylates Forkhead
Zyx	Influences integrin-dependent cell motility and actin stress fiber remodeling
Down-modulated genes	
Tcof1	
Tubb5	Putative function; cytoskeleton and motility
Rdx	Cytoskeletal protein
Dmpk	Role in myogenic differentiation
Mylk	Phosphorylates 20-kDa myosin light chains in a Ca2+/calmodulin-
Diap3	Binds to Cdc42 and remodels the actin cytoskeleton

Table 4. Important genes that are brought in from *periphery*: cell cycle associated.

Gene	Relevance
Up-modulated genes	
Clk1, Ccni	
Down-modulated genes	
Ccnd2	Overexpression in transgenic mice induces thymic and epidermal hyperplasia
Cdc20	A key regulator of the mitotic anaphase-promoting
Cdk8, Pa2g4, Prim1, Topbp1	
Chaf1a	Essential for chromatin assembly in eukaryotes
Pak2	Growth Inhibition TGF-β
Plk4	Marker for cellular proliferation.
Ybx1	Transcription factor

Table 5. Important genes that are brought in from *periphery*: known to be related to breast cancer or TGF-β treatment.

Gene	Relevance
Up-modulated genes	
Pdgfc	Structurally more similar to VEGF-A than to PDGF-B
Gadd45g	Clinicopathological significance in human familial breast carcinoma.
Down-modulated genes	
Cdc20	A key regulator of the mitotic anaphase-promoting complex
Dapk1	TGF-β induces apoptosis through Smad -mediated expression of DAP-kinase
Egr1	Capable of stimulating the activity of the murine TbetaR-II promoter
Fgfr2	Transforming potential of alternatively spliced variants in human mammary epithelial cells.
Fosl1	Transcription factor through which TGF-β regulates Clusterin (amongst other genes)
Ly6e	Established protooncogene in T cell
Map3k12	Regulates radial cell migration via microtubule-based events
Mylk	Phosphorylates 20-kDa myosin light chains in a Ca2+/calmodulin-dependent manner
Pkm2	Role in glycolysis
Ppp1cc	Enhances cellular glycogen levels
Prkx	Highly distinctive expression pattern during neuronal development
Rps6ka3	Regulator Eralpha phosphorylation, docking and transcriptional activation
Senp2	Promotes nuclear accumulation and metabolic stability of tumor suppressor Smad4
Shmt1	Catalyzes the reversible conversion of serine and tetrahydrofolate to glycine and methylenetetrahydrofolate
Suv39h1	Role in myogenic differentiation
Ybx1	Transcription factor

PDGFC and GADD45g (Table 5). The real value of this gene selection process in terms of important genes becomes known when extensive biological validation is performed.

A Gene Set Enrichment Analysis was conducted using GSEA software [12]. Given an *a priori* defined set of genes *Z* (e.g. genes encoding products in a metabolic pathway, located in the same cytogenetic band, or sharing the same GO category), GSEA determines whether the members of gene set *Z* are randomly distributed throughout the input gene list *L* or primarily found at the top or bottom. Functional sets were applied in this study. A summary of the results from GSEA are given in Table 6 which shows that the up-modulated gene list generated through the multi-strategy method is much more enriched in gene sets that are related to breast cancer, EMT or the TGF-β signaling pathway compared to the other three methods. For the down-modulated gene lists, the multi-strategy method performed equally to the RP, slightly better than SAM and much better than the t-test.

Reverse transcriptase polymerase chain reaction (RT-PCR) and quantitative RT-PCR experiments confirmed some of the genes selected through our multi-strategy method. Table 7 shows all of the genes that were confirmed and which were modulated by TGF-β. These results show that if only SAM were applied, two genes would have been missed. The number of missed genes would have been four if only t-test were applied and one if only RP were applied. If we only consider the core set, then three genes will be missed. However, multi-strategy approach helped us to identify many important genes. Union set of all the three methods will contain too many false positive.

Table 6. Summary of GSEA results (only a small subset selected).

Gene set enriched for up modulated genes (p< 0.01 & FDR < 0.25)	Multi-strategy method	*SAM*	*t-test*	*RP*
BREAST_CANCER_ ESTROGEN_SIGNALING	x			x
TGFBETA_EARLY_UP	x			x
…	…	…	…	…
# of gene set (which are related to breast cancer, or EMT/TGF-β signaling pathway) enrichment to each gene lists	**9**	**2**	**1**	**7**

Gene set enriched for down modulated genes (Top20 & p<0.001)	*Multi-strategy method*	*SAM*	*T-Test*	*RP*
BREAST_CANCER_ ESTROGEN_SIGNALING	x			x
BRCA_ER_NEG			x	
…	…	…	…	…
# of gene set (which are related to breast cancer, or EMT/TGF-β signaling pathway) enrichment to each gene lists	**10**	**9**	**6**	**10**

Table 7. Biological validation by RT-PCR or SQ-RT-PCR on certain selected genes

CloneID	Gene Name	TGF-β modulated, selected by different method					Biological confirmation	
		SAM	T-Test	RP	Core	Multi-Strategy	Real-time PCR	SQ-RT-PCR
H3108A04	Clusterin	x	x	x	x	x	x	x
H3003A10	CTLA-2	x	x	x	x	x	x	
H3112A08	Matrilin	x	x	x	x	x	x	
H3124A01	Tenascin	x	x	x	x	x	x	
H3017E12	Syndecan-3			x		x	x	
H3054C02	Gadd45γ			x		x		x
H3089D06	Caveolin-1	x	x	x	x	x		x
H3099E11	Makorin	x				x		x
H3118G02	Ptpn13	x		x	x	x		x
H3151A04	Integrin α 6	x	x	x	x	x		x

6 Conclusions

Many methods have been developed to identify lists of differentially expressed genes when comparing stages, treatments, etc. of a biological system. It is well known that relying on the results that were obtained through a single analysis method is highly risky. We therefore propose a multi-strategy analysis method that takes into account and integrates the output of all participating methods. Applying our multi-strategy method to the JM01 microarray data indicates promising results, and has encouraged us to apply this method to other biological data, including perhaps, a proteomics data set. The main challenge in all multi-strategy methods is to consolidate and interpret the results. We strongly believe that, given the objectives of this research and the biological problem, applying multiple methods has produced better results than any single method. The two most important questions arising from this analysis are, given a biological dataset of this type, and having the same objectives that we had in this research, and knowing that different datasets normally contain different characteristics, (i) what are the most appropriate methods to search for informative genes that would lead us to identifying biomarkers, and (ii) how to combine the results. We would like to pursue this research by applying this approach to more datasets, and also consider other comparison methods. The novelty of our approach is in employing data mining concepts, refining them and combining them with other methods to discover useful information from biological data. Our contribution is in the area of unsupervised learning.

References

1. Amershi S. and Conati C., User modeling: Unsupervised and supervised machine learning in user modeling for intelligent learning environments Proceedings of the 12th international conference on Intelligent user interfaces IUI '07, January 2007.
2. Bloedorn E., Michalski R.S., and Wnek J., Multistrategy constructive Induction: AQ17-MCI, Proceedings of the 2nd International workshop on Multi-strategy learning, May 1993.
3. Breitling R., Armengaud P. Amtmann A., and Herzyk P., (2004) Rank products: a simple, yet powerful, new method to detect differentially regulated genes in replicated microarray experiments, FEBS letters 573, pp 83-92. 3. Cleveland, W.S. (1979) Robust locally.
4. Diaz-Uriarte R. and Alvarez de Andres S., Gene selection and classification of microarray data using random forest, BMC Bioinformatics, 7(3) January 2006.
5. Dietterich T.G., "An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting and randomization," Machine Learning, vol. 40, no. 2, pp. 1-19, 2000
6. Geurts P., Ernst D., and Wehenkel L., Extremely randomized trees Source**,** Machine Learning, Volume 63 , Issue 1, April 2006.
7. Glynn Dennis Jr*, Brad T Sherman*, Douglas A Hosack*, Jun Yang*, Wei Gao*, H Clifford Lane† and Richard A Lempicki* Software DAVID: Database for Annotation, Visualization, and Integrated Discovery. *Genome **Biology*** 2003, **4:**R60.
8. Hsu, W.H., Welge M., Redman T., and Clutter D., High performance commercial data mining: A multi-strategy machine learning application, Data Mining and Knowledge Discovery Journal, Vol, 6(4) October 2002.
9. Jeffery I.B, Higgins D.G. and Culhane A.C., Comparison and evaluation of methods for generating differentially expressed gene lists from microarray data, Bioinformatics, 7:359, July 2006.
10. Kamishima T. and Motoyoshi F., Learning from Cluster Examples, Machine Learning, Volume 53, Number 3, 2003 , pp. 199-233.
11. Lenferink, A. EG, Magoon, J., Cantin, C., and O'Connor-McCourt, M. D.: Investigation of three new mouse mammary tumor cell lines as models for transforming growth factor (TGF)-β and Neu pathway signaling studies: identification of a novel model for TGF-β-induced epithelial-to-mesenchymal transition. Breast Cancer Res. 2004; 6(5): R514–R530.
12. Subramanian, A., Tamayo, P., Mootha, V.K., Mukherjee, S., Ebert, B. L., Gillette, M. A., Paulovich, A., Pomeroy, S.L., Golub, T.R., Lander, E.S., and Mesirov, J.P.: Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. PNAS October 25, 2005. vol. 102 no. 43 15545–15550.
13. Tusher V. G., Tibshirani R. and Chu G. (2001) Significance analysis of microarrays applied to the ionizing radiation response, PNAS, Vol.98 (9), 5116-5121.
14. http://www.nature.com/news/2007/070521/full/070521-13.html
15. The Gene Ontology Consortium: Gene ontology: tool for the unification of biology, *Nature Genetics*, **25**, 25-29
16. KEGG: Kyoto Encyclopedia of Genes and genomes, http://www.genome.jp/kegg/

Breast Cancer Biomarker Selection Using Multiple Offspring Sampling

A. LaTorre[1], J.M. Peña[1], S. González[1], O. Cubo[1], F. Famili[2]

[1] Computer Architecture Department, Universidad Politécnica de Madrid
Boadilla del Monte, Madrid, 28660, Spain
{atorre, jmpena, sgonzalez, ocubo}@fi.upm.es
[2] Institute for Information Technology, National Research Council,
Ottawa Ontario, K1A 0R6, Canada;
fazel.famili@nrc-cnrc.gc.ca

Abstract. Biomarkers are biochemical facets that can be used to measure different aspects of a disease. In the last years, there has been much interest in biomarkers of different cancer variants for predicting future patterns of disease. However, DNA Biomarker selection is a difficult task as it involves dealing with a special type of datasets, microarrays, that consists of a large number of features with small number of samples. This paper proposes a new approach for biomarkers selection by means of an innovative parallel evolutionary algorithm that performs wrapper feature selection from thousands of genes to achieve a small set of most relevant ones. To test our method, the well known Van't Veer dataset on Breast Cancer [1] has been considered. Preliminary results outperform those reported by Van't Veer both in accuracy and the number of genes selected.

1 Introduction

Biomarkers are biochemical facets or features that can be used to measure different aspects of a disease, like the risk to develop it, its progress or the effects of particular treatments. Disease markers can be studied at many molecular levels, ranged from genomic, epigenomic, proteomics, cellular and morphologic, to genetic factors. These factors predispose patients to the disease or indicate its occurrence. In particular, genetic biomarkers are DNA subsequences that have biological significance, in terms of disease evolution, drug tolerance or response to specific treatments.

There has been much interest in biomarkers of cancer variants in predicting future patterns of disease, especially as cancer treatment has made such positive strides in the last few years. The hope that prognosis and disease treatment could be predicted using these information patterns pushes forward the research in this particular field.

During the last few years, early cancer diagnosis has been based on the concentration of serum antigens, like CEA (Carcinoembryonic antigen), in blood [2]. CEA and other antigens are nonspecific for cancer and can be produced by normal organs as well. Their application is restricted in use and no treatment is ever based solely on a CEA. Usually, alterations above normal can spur further diagnostic testing to catch the disease at an early stage. These serum biomarkers can be partially effective in preliminary diagnosis or, additionally, as a way of determining the adequacy of postoperative therapy [3].

As an alternative to serum antigens, DNA biomarkers could provide predictive capabilities in the evaluation of the evolution of the disease and prognosis [4]. In addition, and even more important, they could lead us to the development of effective treatments using appropriate drugs and therapies.

DNA biomarker selection is difficult to perform. The machine learning analogy for biomarkers selection is feature subset selection (FSS) on microrrays. Microarrays are datasets with the problem of curse of dimensionality (large number of features with small number of samples). FSS approaches are divided into wrapper and filter methods. Wrapper methods provide better results but they have two major issues to be considered: (i) a robust and coherent validation method should be applied to ensure quality and fairness of the internal classifier, and (ii) the size of the search space grows exponentially according to the number of features.

In this paper, a new approach is presented on biomarkers selection. This approach is based on an innovative parallel evolutionary algorithm that performs wrapper feature selection from thousands of genes to achieve a small set of most relevant ones, keeping the best prediction quality. This new technique is a two-stage method (depicted on figure 1). Preliminary feature filtering and data preprocessing is followed by the actual biomarkers selection using Multiple Offspring Sampling (MOS). This method has been tested using the well known Van't Veer dataset on Breast Cancer [1]. The selection obtained includes fewer genes than the ones reported by Van't Veer getting better prediction results.

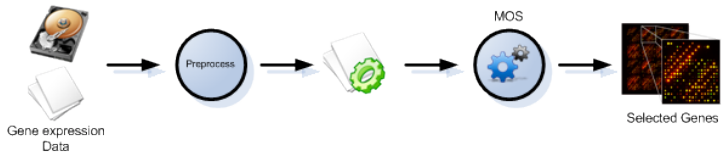


Fig. 1. Overview of MOS applied to Biomarker Selection

2 Feature Subset Selection (FSS)

The FSS problem [5,6,7] deals with the search of the best subset of variables to train a classifier. This is a very important issue in several areas of knowledge discovery such as machine learning, optimization, pattern recognition and statistics. The goal behind FSS is the appropriate selection of a relevant subset of features upon which to focus the attention of a classification algorithm, while ignoring the rest. The FSS problem is based on the fact that the inclusion of more variables in a training dataset does not necessarily improve the performance of the model. We can distinguish two different kinds of variables:

Irrelevant features. This variable has no relation with the target of the classifier.

Redundant features. There exist subsets of variables with variables whose information can be deduced from other variables from that subset. The inclusion of all variables within one of those sets will not improve the final model.

2.1 Classical Solutions

The literature describes several approaches to solve this problem. To achieve the best possible performance with a particular learning algorithm on a particular training set, a feature subset selection method should consider how the algorithm and the training set interact. There are two alternatives to consider this interaction which have been called *filters* or *wrappers*, respectively.

Filter methods [8,9] are mathematical expressions that evaluate each available feature. We can sort all features using this evaluation to obtain a ranking of features and cut this rank when desired. Correlation between each feature and the target variable is a classical example of a filter measure. Filter FSS is based on an estimation of the performance of the algorithm (without its actual execution) based on statistical or information-based relationships among the selected features, including the classification label.

Wrapper methods [7,10] use the induction algorithm itself to evaluate the performance of each subset. We train the model with each candidate subset of features and use any resulting quality measure to evaluate each candidate feature selection. In the wrapper approach, FSS becomes an optimization problem for finding the best set of features, using the induction algorithm as a black box.

Filter methods are, in practice, faster than wrapper ones and obtain good enough results in some datasets. Wrapper methods potentially achieve better feature selections but their computational cost is higher. There are two main aspects that deeply influence the computational cost of these techniques: (i) the optimization algorithm could be more or less exhaustive. For example, forward selection, backward elimination, and their stepwise variants can be viewed as simple hill-climbing techniques in the space of feature subsets; (ii) the robustness of the validation method applied to evaluate the quality of the results obtained by each candidate selection. It includes the measure to use, but also the validation schema (leave-one-out, cross-validation, bootstrap, ...). These validation methods behave differently in terms of variance, bias, and complexity.

An accurate FSS technique based on wrapper approaches that combines both a powerful search method and a robust validation approach is still a challenge, particularly in high dimensional datasets. An appropriate alternative is using a *hybrid approach*. The most common one is the use of a filter to reduce the number of features (features are ranked based on their representativness and the worst are removed), and a wrapper to perform the final selection. This represents a balance between the number of features to make the wrapper technique reasonable in computational time and the number of features included in the optimal subset selection.

2.2 Heuristic Wrapper Approaches

As it has been said before, wrapper methods use the final model as an internal evaluation step for feature selection. The wrapper trains a model and uses the accuracy of the model

as the fitness value of the subset used for training. These methods need a search schema that guides the generation and selection of subsets of features.

An exhaustive search generates and evaluates all possible subsets of features and so the algorithm always finds the best subset. The main disadvantage of this approach is its complexity. The application of this algorithm is unfeasible even with small-medium size datasets.

Most common approaches are greedy algorithms due to its low computational cost and good results in general. Four greedy approaches can be distinguished [11,12]:

Sequential Forward Selection. The search starts with an empty subset and adds the best feature in each step until no improvement can be done.

Sequential Backward Elimination. This approach starts with all features selected and deletes the worst one. The deletion of variables stops when no improvement can be done.

Sequential Floating Forward Selection. The algorithm starts with an empty subset and the best feature is added in each step (the same as SFS). After adding the variables it tries to delete one of the previously selected ones (a backward step) if this improves the current solution.

Sequential Floating Backward Elimination. It starts with all features and deletes the worst. After deleting each variable, the algorithm tries to add one of the previously discarded ones.

Genetic Algorithms [13,14] have been proposed as an alternative to FSS in regular datasets. Although it is a more powerful explorative method, the results with standard datasets are similar to the greedy alternatives. However, these algorithms may behave differently with horizontal datasets (e.g. microarrays).

2.3 FSS applied to Microarray Analysis

The analysis of gene expression using microarray data has become popular in the past few years. Microarrays are applied to a wide variety of problems in life and medical sciences. An important issue is patients' diagnosis for some specific disease. Because of the cost and effort required to gather this information, microarray datasets have only a low number of samples or observations (10-100). However, each sample has a large number of numerical expression levels of genes (10000-30000). This extreme asymmetry, referred as the "curse of dimensionality" [15], is the typical property of most microarray datasets, and needs modified computational techniques to be analyzed.

An important task in classification is to reduce the high dimensionality feature space, that is, for example, applying dimensionality reduction or feature subset selection techniques.

Feature selection applied to microarray data has primarily been studied in a supervised learning context, where predictive accuracy is commonly used to evaluate feature subsets. Specifically, (penalized or non-penalized) logistic regression algorithms were used by [16,17]. Even new algorithms based on logistic regression (Recursive Feature Elimination) were proposed [18] to obtain the best genes selection. Other supervised methods have also been considered [19,1] for cancer diseases.

Considering both wrapper and filter feature selection, Inza and Larrañaga present a comparison between both models in DNA microarray domains [20]. Different methods using both models have been proposed [21,22,1] trying to exploit benefits from both approaches with significative results.

3 Breast cancer dataset description

Van't Veer dataset [1] on Breast Cancer [3] has been considered to validate our approach. As we know, Van't Veer researches were approved by FDA (Food and Drug Administration) and were applied in a genetic test, named MammaPrint, that predicts whether patients will suffer breast cancer relapse or not.

Data is divided into two groups, learning and validation instances. The training data consists of 78 patients, 34 of which are patients that developed distance metastases within 5 years (poor prognosis). The rest of the dataset (44 patients) are the ones who remained healthy from the disease after their initial diagnosis for an interval of 5 years (good prognosis). The second group of patients (validation dataset) consists of 19 patients, 12 patients with poor prognosis and 7 with good prognosis.

DNA microarray analysis was used to determine the mRNA expression levels of approximately 24500 genes for each patient. All the tumours were hybridized against a reference pool made by pooling equal amounts of RNA from each patient.

3.1 Preprocessing

Obviously, the original data contains many redundancies and also incorrect or missing values, depending on some factors. So, as a first step, certain preprocessing was performed in order to clean up and prepare the data. Variables with low internal variance or low correlation with outcome were also discarded.

Several preprocessing algorithms have been carried out through the training data. Firstly, replicated genes are discarded. Next, patients with more than 80% of missing gene values are also discarded. All data have been background corrected, normalized and log-transformed using Lowess Normalization [23]. Missing values were estimated using a 15-weighted nearest neighbours algorithm [24] (kNN Impute).

3.2 Preliminary Filtering

Filter scoring tries to identify genes that are differentially expressed in the categories of the problem. The first step of the filter procedure is to rank the features in terms of the values of the used univariate scoring metric. In a second step, the d features with the highest scoring metric are chosen to induce the LR model. For this contribution, Pearson measure has been selected.

$$r(j) = \frac{\sum_{i=1}^{N}(x_{ij} - \overline{x}_j) \cdot (y_i - \overline{y})}{(n-1) \cdot s_j \cdot s_y} \tag{1}$$

[3] available at `http://www.rii.com/publications/2002/vantveer.html`

where $\overline{x}_i$ is the mean value, s_j is the standard deviation of expression levels, and y_i and $\overline{y}$ are the class value and class mean respectively.

Next, a ranking list ordered by Pearson correlation is generated. With this list, a group of 1000 best genes has been selected. A large number of pre-candidate genes has been selected to provide enough alternatives to the wrapper search in the second stage. As mentioned before, this means that the search space in the wrapper method is large and potentially very complex. The proposed wrapper method to select biomarkers from a 1000 candidate genes is based on an innovative evolutionary technique that allows optimal values to be found on complex and large search spaces.

4 MOS: Multiple Offspring Sampling

Multiple Offspring Sampling is introduced as a variant of classic population-based evolutionary algorithms. This new approach proposes the simultaneous use of different *techniques* (a proper definition of technique in the context of *MOS* will be given in subsection 4.2) to create new individuals (candidate solutions).

To show how *MOS* modifies the behaviour of classic Evolutionary Algorithms (EA), we should first present a general schema of EA functioning, which will be given in the next subsection. Afterwards, Multiple Offspring Sampling will be presented.

4.1 Evolutionary Algorithms

Evolutionary algorithms (like Genetic Algorithms (GAs)), in a general schema, are divided into different phases:

① Creation of the initial population P_0.
② Evaluation of the initial population P_0.
③ Checking of the algorithm termination (convergence or generation limit), if so then finish, otherwise continue.
④ Generation, using some individuals from P_i, of new individuals for the next generation, called offspring population O_i.
⑤ Evaluation of the new individuals in O_i.
⑥ Combination of offspring and previous population to define the next population P_{i+1}.
⑦ Go back to ③.

Based on this schema, different evolutionary algorithms and approaches have been developed. For example, in step ⑥ classical GAs take the offspring as the next population ($P_{i+1} = O_i$). Other approaches, like steady state algorithms generate only one offspring individual that replaces the worst individual in P_i, and intermediate approaches, based on elitism, take the best individuals from both O_i and P_i to generate P_{i+1}.

In step ④, there have been also many different approaches in the literature. Some examples are based on selecting different genetic operators, or using statistical approaches for modelling the population and later sampling the offspring (e.g. estimation of distribution algorithms by [25]).

4.2 Multiple Offspring Basics

We introduce Multiple Offspring Sampling (MOS) approach as a combined alternative in the way steps ④ and ⑥ are performed. MOS proposes the definition of multiple mechanisms to generate new individuals, and make them compete during the evolution process. Each mechanism creates its own offspring $O_i^{(j)}$ (i is the generation and j is the mechanism).

These MOS mechanisms, or techniques, as they are named at the beginning of section 4, could be defined as a mechanism to create new individuals, i.e., (a) a particular evolutionary algorithm model, (b) with an appropriate coding, (c) using specific operators (if required) and (d) configured with its necessary parameters.

According to the above definition we can consider different parameters and thus divide MOS into several categories. A rough taxonomy of how MOS can be divided could be:

- Algorithm-based MOS: different algorithms (GAs, EDAs) are used to create new individuals.
- Coding-based MOS: different codings (genotypes) can be used to represent one candidate solution (phenotype) of the problem.
- Operator-based MOS: for a single coding of candidate solutions there could exist different genetic operators (if working with GAs) that could be used simultaneously.
- Parameter-based MOS: different values for evolutionary parameters (crossover and mutation ratios, selection mechanisms, etc.) are used within each technique.
- Hybrid MOS: a combination of any of the previous.

In the particular case of the experimentation performed for this study, two different genotype encodings are considered.

As a solution, the phenotype, can participate in multiple genotype recombinations, a group of functions is required to transform genotypes between two different encodings.

Once the offspring population is created by each of the techniques being used, the quality of these populations is evaluated by means of several possible measures. The most obvious of these measures is the average fitness of the population, but more sophisticated measures could be proposed to take into account not only the current performance of the technique but its capability.

Finally, in phase ⑥, previous population P_i and all the offsprings $O_i^{(j)}$ are merged to produce the next population P_{i+1}. This process is usually done by using an elitist population merge function.

The calculation of the amount of new individuals created in each generation, for n different offspring sampling methods, is obtained using a Participation Function (***PF***). Different functions have been proposed in other scenarios by [26], where the first approach to Algorithm-MOS was introduced under the combination of two different Evolutionary Algorithms: GAs and EDAs. From these functions, a dynamic one was selected for being used in our studies. This function dynamically adjusts the participation of each technique according to the quality of the offspring populations calculated before.

4.3 MOS for Biomarker Selection

Previous subsections have introduced MOS as an innovative parallel genetic algorithm that is able to exploit the benefits of using different techniques to produce a new offspring based on current population. In the case of this study, two different codings were used.

First coding is simply a binary vector of length the number of features in the learning dataset. Each of these binary values tells if that feature will or will not be selected by the algorithm.

Second coding is a condensed version of the first one, consisting of a vector of integer numbers where each number represents a gene being selected by the algorithm. This messy coding was firstly introduced by [27] and since then reliably applied to a wide range of optimization problems [28,29].

These two codings coexist all along the evolutionary process, each of them taking more participation in different phases of the execution of the genetic algorithm and helping the GA to outperform itself when using just a single genetic representation (coding).

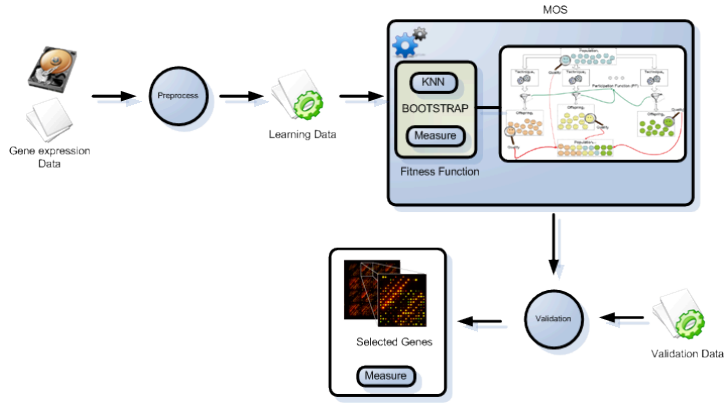


Fig. 2. A detailed view of MOS applied to Biomarker Selection

5 Experimentation scenario, results and discussion

This section provides an overview of the whole process followed in this research along with the results obtained and a discussion about these results.

Figure 4.3 clearly depicts the followed process:

① Firstly, a preprocessing phase is performed to select the best 1000 genes considering their position in a ranked list ordered by Pearson correlation, as explained in section 3.2.
② Then, a MOS algorithm is executed to select most relevant genes from these 1000 genes previously selected. MOS will evaluate each generated individual with a bootstrap (200 iterations) using a KNN algorithm, trying to optimize a fitness "measure", the AUC in this experimentation, as it has empirically demonstrated to behave quite well for this problem. For KNN, two different distance measures have been considered: traditional Euclidean distance and Chebyshev distance. First one has been selected because it has been widely used in previous works and that lets us to fairly compare our approach with others. Second one has been used due to its capability to penalize selections of genes with large distance among only few of them (even between just two of them), a characteristic we wanted to exploit in our experiments.
③ Finally, an external validation process is performed considering only selecting genes to learn a KNN algorithm and a validation dataset different from that used to learn and not seen by the algorithm until now.

The experiments were executed on a 13 dual Xeon cluster at 2.40 GHz, using a parallel asynchronous genetic algorithm implemented in GAEDALib coded by [30] with the configuration described in table 1(b).

Table 1. Experimental scenario

(a) GA configuration

(Global) Pop. size	390
Termination	Pop. convergence
Convergence %	98 %
Individuals selection	Roulette wheel
Crossover %	90 %
Mutation %	1 %

(b) Parallel configuration

Paradigm	islands model
Model	asynchronous
Topology	mesh
Migration rate	10 gens.
Migration pop.	Top 20 %
Nodes	26

Table 2 summarizes the results obtained in this experimentation. Fourteen different configurations were tested. For each of the two distance measures considered, seven different fitness functions were tested. First function only tried to maximize the AUC, regardless of the number of selected genes. With such a great degree of freedom the algorithm tends to select a huge number of genes. For this reason, a new fitness function was introduced (see equation 2) that tries to avoid this problem. This fitness function tries to lower the number of variables as much as possible but not more than the pivot value that acts as a center of gravity for the number of variables. Then, six new configurations were executed, with the only difference being in the pivot value.

$$\textit{fitness} = AUC * \frac{1}{abs(\#genes - pivot) + 1} \tag{2}$$

Results, in general, outperform those of Van't Veer both in prediction accuracy and smaller number of genes selected. Best results are achieved with Chebyshev distance and the penalized fitness function with pivot equal to 40, although there are not great differences among all the configurations with penalized fitness function regardless of the distance measure used. This makes us think that the optimal number of genes must be within this range ([20, 60]).

Table 2 presents the average results of ten executions for each configuration. Several executions of the algorithm (with different configurations) returned a selection of genes with an impressive 94% of accuracy in external validation and with just 20 genes selected in the best case.

From table 2 we can also observe that there exists a strong correlation between the optimization measure (AUC) and the validation measure (accuracy) (0.92 for Pearson correlation). This property is quite desirable for an optimization measure when training an algorithm that will be validated with unknown data.

Finally, the penalizing method appears to be very restrictive and makes the algorithm to adjust perfectly to the selected value of pivot. This behaviour must be studied and some modifications may be introduced to allow a certain level of flexibility for the number of features selected.

Table 2. Summary of results: all reported values are the average of ten executions

	AUC	**Accuracy**	**Size**
Chebyshev Distance - Not Penalized	0.75	0.79	317.70
Chebyshev Distance - Penalized (centered on 0)	0.60	0.71	2.45
Chebyshev Distance - Penalized (centered on 20)	0.76	0.81	20.00
Chebyshev Distance - Penalized (centered on 30)	0.73	0.79	30.00
Chebyshev Distance - Penalized (centered on 40)	0.76	**0.84**	40.00
Chebyshev Distance - Penalized (centered on 50)	0.75	0.81	50.00
Chebyshev Distance - Penalized (centered on 60)	0.75	0.81	60.00
Euclidean Distance - Not Penalized	0.74	0.77	131.25
Euclidean Distance - Penalized (centered on 0)	0.66	0.73	2.35
Euclidean Distance - Penalized (centered on 20)	0.76	0.80	20.00
Euclidean Distance - Penalized (centered on 30)	0.75	0.82	30.00
Euclidean Distance - Penalized (centered on 40)	0.75	0.81	40.00
Euclidean Distance - Penalized (centered on 50)	0.80	0.82	50.00
Euclidean Distance - Penalized (centered on 60)	0.76	0.81	60.00

6 Conclusions and future work

This paper introduces an innovative and robust method to perform FSS on large microarray data sets (1000 features or more).

It also presents a validation mechanism that consists of: (i) an internal validation process to avoid overfitting to learning data (bootstrap with 200 iterations in this ex-

perimentation) and (ii) an external validation to evaluate the quality of the selection of genes.

Results demonstrate the effectiveness of this method, with an average accuracy of 84% in the best configuration, and several selections of genes with an accuracy of 94%. The number of genes selected is also fewer than those reported by Van't Veer, which makes this approach outperform previous works both in accuracy and selections of genes' size.

Future works will include analysis of the relations among different selections of genes with similar performance and a study of the behaviour of the algorithm when learning with measures others than AUC.

Acknowledgements

This research project is funded by the Spanish Ministry of Science TIN2007-67148.

References

1. van 't Veer, L.J., Dai, H., van de Vijver, M.J., He, Y.D., Hart, A.A., Mao, M., Peterse, H.L., van der Koõy, K., Marton, M.J., Witteveen, A.T., Schreiber, G.J., Kerkhoven, R.M., Roberts, C., Linsley, P.S., Bernards, R., Frieñd, S.H.: Gene expression profiling predicts clinical outcome of breast cancer. Nature **415**(6871) (2002) 530–536
2. Wang, D., Knyba, R., Bulbrook, R., Millis, R., Hayward, J.: Serum carcinoembryonic antigen in the diagnosis and prognosis of women with breast cancer. Eur J Cancer Clin Oncology **1**(20) (1984) 25–56
3. et al., J.M.T.: Serum markers and prognosis in locally advanced breast cancer. Tumori **6**(91) (2005) 522–552
4. Hartwell, L., Mankoff, D., Paulovich, A., Ramsey, S., Swisher, E.: Cancer biomarkers: a systems approach. Nature Biotechnology **8**(24) (2006) 905–913
5. Almuallim, H., Dietterich, T.: Learning with many irrelevant features. In: Proceedings of the Ninth National Conference on Artificial Intelligence (AAAI-91). Volume 2., Anaheim, California, AAAI Press (1991) 547–552
6. Blum, A., Langley, P.: Selection of relevant features and examples in machine learning. Artificial Intelligence **97**(1–2) (1997) 245–271
7. John, G., Kohavi, R., Pfleger, K.: Irrelevant features and the subset selection problem. In: International Conference on Machine Learning. (1994) 121–129 Journal version in AIJ.
8. Ben-Bassat, M.: Use of distance measure, information measures, and error bounds on feature evaluation. In Krishnaiah, P.R., Kanal, L.N., eds.: Classification, Pattern Recognition and Reduction of Dimensionality. North–Holland Publishing Company, Amsterdam (1987) 773–791
9. Jeffery, I.B., Higgins, D.G., Culhane, A.C.: Comparison and evaluation of methods for generating differentially expressed gene lists from microarray data. BMC Bioinformatics **7** (2006) 359+
10. Kohavi, R., John, G.: Wrappers for feature subset selection. Artificial Intelligence **97**(1–2) (1997) 273–324
11. Kittler, J.: Feature set search algorithms. Pattern Recognition and Signal Processing (1978) 41–60
12. Somol, P., Pudil, P., Novovicová, J., Paclík, P.: Adaptive floating search methods in feature selection. Pattern Recognition Letters **20**(11–13) (1999) 1157–1163

13. Inza, I., Larrañaga, P., Sierra, B.: Feature subset selection by bayesian networks: a comparison with genetic and sequential algorithms. International Journal of Approximate Reasoning **27**(2) (2001) 143–164
14. Jain, A., Zongker, D.: Feature selection: Evaluation, application, and small sample performance. IEEE Trans. Pattern Anal. Mach. Intell. **19**(2) (1997) 153–158
15. usa H. Asyali, Dilek Colak, O.D., Inan, M.S.: Gene expression profile classification: A review. Current Bioinformatics **1**(1) (2006) 55–73
16. Shevade, S.K., Keerthi, S.S.: A simple and efficient algorithm for gene selection using sparse logistic regression. Bioinformatics **19**(17) (2003) 2246–2253
17. Weber, G., Vinterbo, S.A., Ohno-Machado, L.: Multivariate selection of genetic markers in diagnostic classification. Artificial Intelligence in Medicine **31**(2) (2004) 155–167
18. Shen, L., Tan, E.C.: Dimension reduction-based penalized logistic regression for cancer classification using microarray data. IEEE/ACM Trans. Comput. Biol. Bioinformatics **2**(2) (2005) 166–175
19. Golub, T., Slonim, D., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J., Coller, H., Loh, M., Downing, J., Caliguri, M., Bloomfield, C., Lander, E.: Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science **286** (1999) 531–537
20. Inza, I., Larrañaga, P., Blanco, R., Cerrolaza, A.J.: Filter versus wrapper gene selection approaches in dna microarray domains. Artificial Intelligence in Medicine **31**(2) (2004) 91–103
21. Mamitsuka, H.: Selecting features in microarray classification using roc curves. Pattern Recogn. **39**(12) (2006) 2393–2404
22. Statnikov, A., Tsamardinos, I., Dosbayev, Y., Aliferis, C.F.: Gems: a system for automated cancer diagnosis and biomarker discovery from microarray gene expression data. Int J Med Inform **74**(7-8) (2005) 491–503
23. Quackenbush, J.: (Microarray data normalization and transformation - nature genetics)
24. Troyanskaya, O., Cantor, M., Sherlock, G., Brown, P., Hastie, T., Tibshirani, R., Botstein, D., Altman, R.B..: Missing value estimation methods for dna microarrays. Bioinformatics **17**(6) (2001) 520–525
25. Larrañaga, P., Lozano, J.: Estimation of Distribution Algorithms. A New Tool for Evolutionary Computation. Kluwer Academic Publisher (2002)
26. Robles, V., Peña, J., Larrañaga, P., Pérez, M., Herves, V.: GA-EDA: A New Hybrid Cooperative Search Evolutionary Algorithm. In: Towards a New Evolutionary Computation. Advances in Estimation of Distribution Algorithms. Volume 192 of Studies in Fuzziness and Soft Computing. Springer (2006) 187–220
27. Goldberg, D.E., Deb, K., Kargupta, H., Harik, G.: Rapid accurate optimization of difficult problems using fast messy genetic algorithms. In Forrest, S., ed.: Proceedings of the Fifth International Conference on Genetic Algorithms, San Mateo, CA (1993) 56–64
28. Watson, R.A., Hornby, G.S., Pollack, J.B.: When food is better than sex: Messy variations on the ga. In: Proceedings of the 5th International Conference of the Society for Adaptive Behavior (SAB'98), University of Zurich, Switzerland (1998)
29. Fenton, P., Walsh, P.: A comparison of messy ga and permutation based ga for job shop scheduling. In Beyer, H., O'Reilly, U., eds.: Proceedings of the Genetic and Evolutionary Computation Conference, GECCO 2005, Washington DC, USA, ACM Press (2005) 1593–1594
30. Díaz, P.: Diseño e implementación de una librería de algoritmos evolutivos paralelos. Master's thesis, Facultad de Informática, Universidad Politécnica de Madrid (2005)

Classification of Microarrays with kNN: Comparison of Dimensionality Reduction Methods

Sampath Deegalla[1] and Henrik Boström[2]

[1] Dept. of Computer and Systems Sciences,
Stockholm University and Royal Institute of Technology,
Forum 100, SE-164 40 Kista, Sweden
si-sap@dsv.su.se
[2] School of Humanities and Informatics,
University of Skövde,
P.O. Box 408, SE-541 28, Skövde, Sweden
henrik.bostrom@his.se

Abstract. Dimensionality reduction can often improve the performance of the k-nearest neighbor classifier (kNN) for high-dimensional data sets, such as microarrays. The effect of the choice of dimensionality reduction method on the predictive performance of kNN for classifying microarray data is an open issue, and four common dimensionality reduction methods, Principal Component Analysis (PCA), Random Projection (RP), Partial Least Squares (PLS) and Information Gain(IG), are compared on eight microarray data sets. It is observed that all dimensionality reduction methods result in more accurate classifiers than what is obtained from using the raw attributes. Furthermore, it is observed that both PCA and PLS reach their best accuracies with fewer components than the other two methods, and that RP needs far more components than the others to outperform kNN on the non-reduced dataset. None of the dimensionality reduction methods can be concluded to generally outperform the others, although PLS is shown to be superior on all four binary classification tasks, but the main conclusion from the study is that the choice of dimensionality reduction method can be of major importance when classifying microarrays using kNN.

1 Introduction

Microarray gene-expression technology has spread across the research community with immense speed during the last decade [1]. Being able to effectively learn from data generated through this technology is important for many reasons, including allowing for early accurate diagnoses which might lead to proper choice of treatments and therapies [2, 3]. On the other hand, this type of high-dimensional data, often involving thousands of attributes, creates challenges for many learning algorithms, including the well-known k-nearest neighbor classifier (kNN) [4].

The kNN has a very simple strategy as a learner: instead of generating an explicit model, it keeps all training instances. A classification is made by measuring the distances from the test instance to all training instances, most commonly using the Euclidean distance. Finally, the majority class among the k nearest instances is assigned to the test instance. This simple form of kNN can however be both inefficient and ineffective for high-dimensional data sets due to presence of irrelevant and redundant attributes. Therefore the classification accuracy of kNN often decreases with an increase in dimensionality. One possible remedy to this problem that earlier has shown to be successful is to use dimensionality reduction [5].

The kNN has earlier been demonstrated to allow for successful classification of microarrays [2] and it has also been shown that dimensionality reduction can further improve the performance of kNN for this task [5]. However, it is an open question if the choice of dimensionality reduction technique has any impact of the performance, and for this purpose, four commonly employed dimensionality reduction methods are compared in this study when used in conjunction with kNN for microarray classification.

The organization of the paper is as follows. In the next section, we briefly present the four dimensionality reduction methods used in the study. In section 3, details of the experimental setup are provided, and the results of the comparison on eight microarray data sets are given. Finally, we give some concluding remarks and outline directions for future work.

2 Dimensionality Reduction

2.1 Principal Component Analysis (PCA)

PCA uses a linear transformation to obtain a simplified data set retaining the characteristics of the original data set.

Assume that the original matrix contains d dimensions and n observations and that one wants to reduce the matrix into a k dimensional subspace. This transformation can be given by[6]:

$$Y = E^T X \tag{1}$$

where $E_{d\times k}$ is the projection matrix containing k eigen vectors corresponding to the k highest eigen values, and $X_{d\times n}$ is the mean centered data matrix.

2.2 Random Projection (RP)

By RP, the original data set is transformed into a lower dimensional subspace by using a random matrix [7, 8].

Assume that one wants to reduce the d dimensional data set into a k dimensional set where number of instances are n. The transformation is then given by:

$$Y = R\,X \tag{2}$$

where $R_{k\times d}$ is the random matrix and $X_{d\times n}$ is the original data matrix. The original idea behind the RP is based on the Johnson-Lindenstrauss lemma (JL) [9] which states that n points can be projected from $R^d \rightarrow R^k$ while preserving the Euclidean distance between the points within an arbitrarily small factor. For more details on the method, see [8].

This random matrix can be created in several ways and the one we have used is introduced by Achlioptas [10], by which the random matrix is generated as follows.

$$r_{ij} = \begin{cases} +\sqrt{3} \text{ with } P_r = \frac{1}{6}; \\ 0 \text{ with } P_r = \frac{2}{3}; \\ -\sqrt{3} \text{ with } P_r = \frac{1}{6}. \end{cases} \tag{3}$$

2.3 Partial Least Squares (PLS)

PLS was originally developed within the social sciences and has later been used extensively in chemometrics as a regression method[11]. It seeks for a linear combination of attributes whose correlation with the class attribute is maximized. In PLS regression the task is to build a linear model, $\bar{Y} = BX + E$, where B is the matrix of regression coefficients and E is the matrix of error coefficients. In PLS, this is done via the factor score matrix $Y = WX$ with an appropriate weight matrix W. Then it considers the linear model, $\bar{Y} = QY + E$, where Q is the matrix of regression coefficients for Y. Computation of Q will yield $\bar{Y} = BX + E$, where $B = WQ$. However, we are interested in dimensionality reduction using PLS and used the SIMPLS algorithm [12, 13]. In SIMPLS, the weights are calculated by maximizing the covariance of the score vectors y_a and $\bar{y_a}$ where $a = 1, \ldots, A$ (where A is the selected numbers of PLS components) under some conditions. For more details of the method and its use, see [12, 14]

2.4 Information Gain (IG)

Information Gain (IG) can be used to measure the information content in a feature [15], and is commonly used for decision tree induction. Maximizing IG is equivalent to minimizing:

$$\sum_{i=1}^{V} \frac{n_i}{N} \sum_{j=1}^{K} -\frac{n_{ij}}{n_i} \log_2 \frac{n_{ij}}{n_i}$$

where K is the number of classes, V is the number of values of the attribute, N is the total number of examples, n_i is the number of examples having the ith value of the attribute and n_{ij} is the number of examples in the latter group belonging to the jth class.

3 Empirical Study

3.1 Data Sets

The following eight micorarray data sets are used in this study:

- Colon Tumor [16], which consists of 40 tumor and 22 normal colon samples.
- Leukemia [17], which contains 72 samples of two types of leukemia: 25 acute myeloid leukemia (AML) and 47 acute lymphoblastic leukemia (ALL).
- Central Nervous System [18], which consists of 60 patient samples of survivors (39) and failures (21) after treatment of the medulloblastomas tumor (This is data set C from [18]).
- SRBCT [3], which contains four diagnostic categories of small, round blue-cell tumors as neuroblastoma (NB), rhabdomyosarcoma (RMS), non-Hodgkin lymphoma (NHL) and the Ewing family of tumors (EWS).
- Lymphoma [19], which contains 42 samples of diffuse large B-cell lymphoma (DLBCL), 9 follicular lymphoma (FL) and 11 chronic lymphocytic leukemia (CLL).
- Brain [18] contains 42 patient samples of five different brain tumor types: medulloblastomas (10), malignant gliomas (10), AT/RTs (10), PNETs (8) and normal cerebella (4). (This is the data set A from [18].)
- NCI60 [20], which contains eight different tumor types. These are breast, central nervous system, colon, leukemia, melanoma, non-small lung carcinoma, ovarian and renal tumors.
- Prostate [2], which consists of 52 prostate tumor and 50 normal specimens.

The first three data sets come from Kent Ridge Bio-medical Data Set Repository[21] and the remaining five from [22]. The data sets are summarized in Table 1.

Table 1. Description of data

Data set	Attributes	Instances	# of Classes
Colon Tumor	2000	62	2
Leukemia	7129	38	2
Central Nervous	7129	60	2
SRBCT	2308	63	4
Lymphoma	4026	62	3
Brain	5597	42	5
NCI60	5244	61	8
Prostate	6033	102	2

3.2 Experimental Setup

We have used Matlab to transform raw attributes to both PLS and PCA components. The PCA transformation is performed using the Matlab's Statistics

Toolbox whereas the PLS transformation is performed using the BDK-SOMPLS toolbox[23, 24], which uses the SIMPLS algorithm. The WEKA data mining toolkit [15] is used for the RP and IG methods, as well as for the actual nearest neighbor classification. Default parameters are used with kNN, i.e. distance weighting is not considered in voting.

Both PLS and IG are supervised methods which uses class information for their transformations. Therefore, to generate the PLS components for test sets, the weight matrix generated for the training set has to be used. For IG, attributes in the training set is ranked based on the information content in a decreasing manner and the same attributes are selected for the test set. As earlier explained, attributes generated using RP are of a random nature since a random matrix is used for the transformation. For this reason, we have averaged results of RP from 30 runs to reduce the variance.

The optimal number of neighbors (i.e., k) could be specific to different data sets and dimensionality reduction methods. Therefore, we have investigated the effect of different values of k, namely 1, 3, 5, 7 and 9.

Stratified 10-fold cross validation[15] is employed to obtain measures of accuracy, which has been chosen as the performance measure in this study. For PCA, PLS and IG same training and testing sets are generated with the same seed.

3.3 Experimental Results

The results are summarized in Fig. 1 and Fig. 2. It can be observed that both PLS and PCA obtain their best classification accuracies with relatively few dimensions, while more dimensions are required for IG and many more for RP.

None of the methods turns out as a clear winner, except perhaps PLS on the binary classification tasks. However, all methods outperform not using dimensionality reduction, and the difference in performance between the best and worst method can vary greatly for a particular dataset, leading to the conclusion that the choice of dimensionality reduction to be used in conjunction with kNN for microarray classification can be of major importance.

In most of the cases, simply setting $k = 1$ gives the best result. However, for IG it seems that one should consider choosing higher values for k which improves the classification accuracy by at least 1% for 5 out of 8 datasets. For PCA, the choice of a higher k value yields at least a 1% improvement for 3 out of 8 data sets whereas for PLS, an improvement of at least 1% is obtained for 4 out of 8 datasets.

4 Concluding Remarks

Four dimensionality reduction methods are compared for classifying microarrays with the nearest neighbor classifier. Experiments with eight microarray datasets show that dimensionality reduction indeed is effective for nearest neighbor classification.

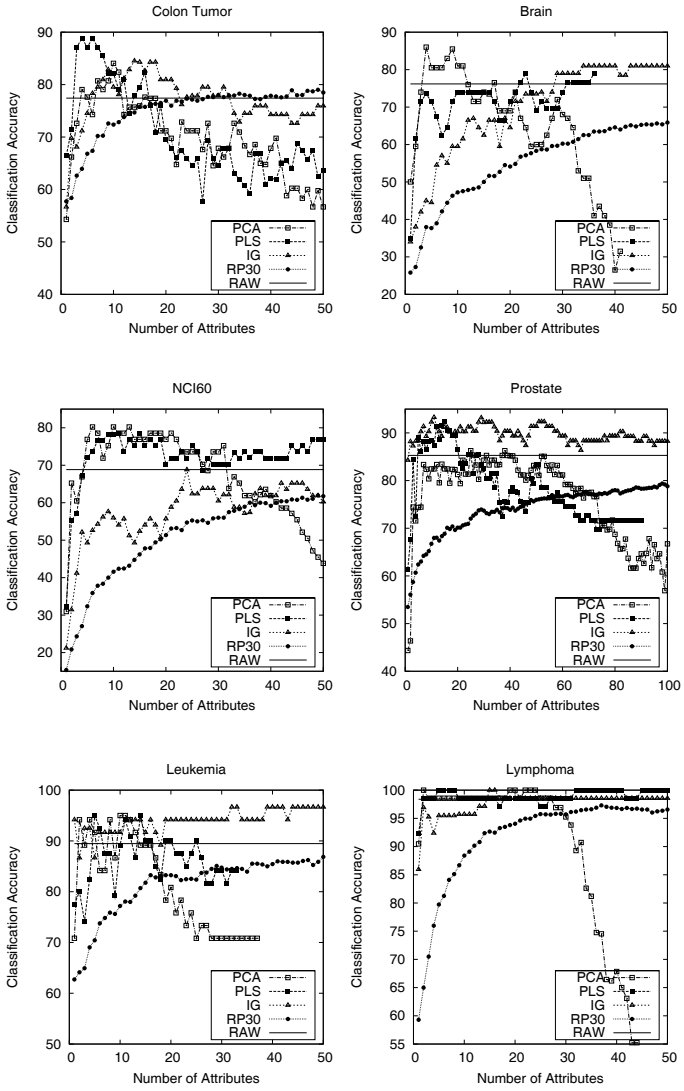


Fig. 1. Predictive performance with the change of numbers of dimensions using PCA, PLS, RP and IG with Nearest Neighbor (IB1) for Colon Tumor, Brain, NCI60, Prostate, Leukemia and Lymphoma data sets.

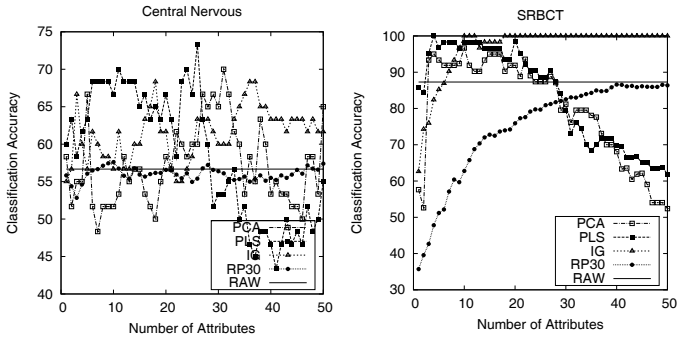


Fig. 2. Predictive performance with the change of numbers of dimensions using PCA, PLS, RP and IG with Nearest Neighbor (IB1) for Central Nervous and SRBCT data sets.

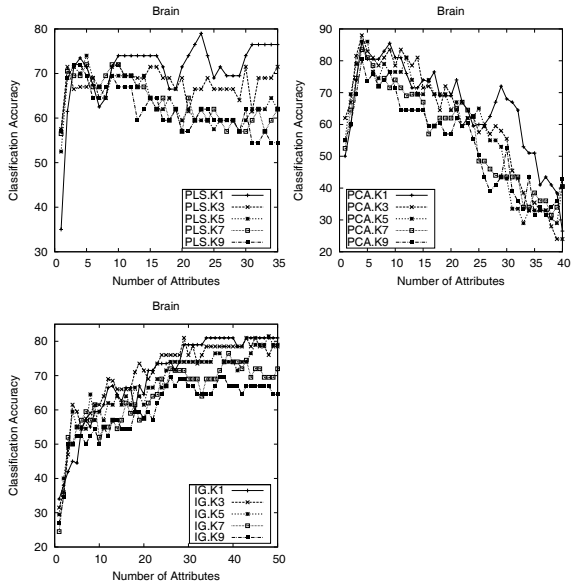


Fig. 3. Predictive accuracy with different k values for nearest neighbor classifier for Brain dataset

Table 2. Order of k values w.r.t averaged accuracy

	Decreasing order of accuracy		
	IG	PCA	PLS
Colon Tumor	7,5,9,3,1	5,9,7,3,1	7,9,5,3,1
Leukemia	1,3,5,7,9	1,3,5,7,9	3,1,5,7,9
Central Nervous	7,9,5,1,3	3,7,9,5,1	9,7,5,3,1
SRBCT	3,5,1,7,9	1,9,3,7,5	9,7,5,3,1
Lymphoma	5,9,1,7,5	1,3,5,7,9	1,3,5,7,9
Brain	3,1,5,7,9	1,3,5,7,9	1,3,5,7,9
NCI60	9,7,1,5,3	1,3,5,7,9	1,3,5,7,9
Prostate	3,7,9,5,1	9,5,7,3,1	9,3,1,7,5

However, none of the methods used in the study consistently gives the best accuracy on all data sets. Generally, both PCA and PLS results in the highest accuracy for few dimensions whereas RP and IG require more dimensions. Compared to the other three methods, PCA is shown to be more sensitive to the choice of dimensionality, and typically gives poor results in higher dimensions. It can be observed that PLS outperforms the other methods for binary classification problems (Colon, Leukemia, Central Nervous and Prostate).

We have also investigated the accuracy of kNN for different values of k. Generally, k=1 seems to be the best choice for PCA and PLS, while higher values are required for IG.

There are a number of issues that need further exploration. First, additional binary microarray classification tasks could be investigated to test the finding that PLS appears to be superior in these cases. Second, further characterizations of the situations in which the different dimensionality reduction methods are successful could be identified. Furthermore, the possibility of combining several reduced features sets generated by different reduction methods could also be investigated.

Acknowledgements

Financial support from SIDA/SAREC for the first author is greatly acknowledged.

References

1. Quackenbush, J.: Microarray analysis and tumor classification. The New England Journal of Medicine **354**(23) (2006) 2463–2472
2. Singh, D., Febbo, P.G., Ross, K., Jackson, D.G., Manola, J., Ladd, C., Tamayo, P., Renshaw, A.A., D'Amico, A.V., Richie, J.P., Lander, E.S., Loda, M., Kantoff, P.W., Golub, T.R., Sellers, W.R.: Gene expression correlates of clinical prostate cancer behavior. Cancer Cell **1** (2002) 203–209
3. Kahn, J., Wei, J.S., Ringnér, M., Saal, L.H., Ladanyi, M., Westermann, F., Berthold, F., Schwab, M., Antonescu, C., Peterson, C., Meltzer, P.: Classification

and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. Nature Medicine **7** (2001) 673–679
4. Aha, D.W., Kiblear, D., Albert, M.K.: Instance based learning algorithm. Machine Learning **6** (1991) 37–66
5. Deegalla, S., Bostrom, H.: Reducing high-dimensional data by principal component analysis vs. random projection for nearest neighbor classification. In: ICMLA '06: Proceedings of the 5th International Conference on Machine Learning and Applications, Washington, DC, USA, IEEE Computer Society (2006) 245–250
6. Shlens, J.: (A tutorial on principal component analysis) URL: http://www.snl.salk.edu/ shlens/pub/notes/pca.pdf.
7. Bingham, E., Mannila, H.: Random projection in dimensionality reduction: applications to image and text data. In: KDD '01: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining. (2001) 245–250
8. Fradkin, D., Madigan, D.: Experiments with random projections for machine learning. In: KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining. (2003) 517–522
9. Dasgupta, S., Gupta, A.: An elementary proof of the Johnson-Lindenstrauss lemma. Technical Report TR-99-006, International Computer Science Institute, Berkeley, California, USA (1999)
10. Achlioptas, D.: Database-friendly random projections. In: ACM Symposium on the Principles of Database Systems. (2001) 274–281
11. Abdi, H.: Partial least squares (pls) regression. (2003)
12. de Jong, S.: SIMPLS: An alternative approach to partial least squares regression. Chemometrics and Intelligent Laboratory Systems (1993)
13. StatSoft Inc.: Electronic statistics textbook (2006) URL: http://www.statsoft.com/textbook/stathome.html.
14. Boulesteix, A.L.: Pls dimension reduction for classification with microarray data. Statistical Applications in Genetics and Molecular Biology (2004)
15. Witten, I.H., Frank, E.: Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann, San Francisco (2005)
16. Alon, U., Barkai, N., Notterman, D., Gish, K., Ybarra, S., Mack, D., Levine, A.J.: Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. In: Proc. Natl. Acad. Sci. USA. Volume 96. (1999) 6745– 6750
17. Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M., Downing, J.R., Caligiuri, M.A., Bloomfield, C.D., Lander, E.S.: Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. Science **286** (1999) 531–537
18. Pomeroy, S.L., Tamayo, P., Gassenbeek, M., Sturla, L.M., Angelo, M., McLaughlin, M.E., Kim, J.Y., Goumnerova, L.C., Black, P.M., Lau, C., Allen, J.C., Zagzag, D., Olson, J.M., Curran, T., Wetmore, C., Biegel, J.A., Poggio, T., Mukherjee, S., Rifkin, R., Califano, A., Stolovitzky, G., Louis, D.N., Mesirov, J.P., Lander, E.S., Golub, T.R.: Prediction of central nervous system embryonal tumour outcome based on gene expression. Nature **415** (2002) 436–442
19. Alizadeh, A.A., Eisen, M.B., Davis, R.E., Ma, C., Lossos, I.S., Rosenwald, A., Boldrick, J.C., Sabet, H., Tran, T., Yu, X., Powell, J.I., Yang, L., Marti, G.E., Moore, T., Hudson Jr, J., Lu, L., Lewis, D.B., Tibshirani, R., Sherlock, G., Chan, W.C., Greiner, T.C., Weisenburger, D.D., Armitage, J.O., Warnke, R., Levy, R., Wilson, W., Grever, M.R., Byrd, J.C., Botstein, D., Brown, P.O., Staudt, L.M.:

Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. Nature **403** (2000) 503–511
20. Ross, D.T., Scherf, U., Eisen, M.B., Perou, C.M., Rees, C., Spellman, P., Iyer, V., Jeffrey, S.S., de Rijn, M.V., Waltham, M., Pergamenschikov, A., Lee, J.C., Lashkari, D., Shalon, D., Myers, T.G., Weinstein, J.N., Botstein, D., Brown, P.O.: Systematic variation in gene expression patterns in human cancer cell lines. Nature Genetics **24**(3) (2000) 227–235
21. Kent Ridge Bio-medical Data Set Repository URL: http://sdmc.lit.org.sg/GEDatasets/Datasets.html.
22. Díaz-Uriarte, R., de Andrés, S.A.: Gene selection and classification of microarray data using random forest. Bioinformatics **7**(3) (2006) URL: http://ligarto.org/rdiaz/Papers/rfVS/randomForestVarSel.html.
23. Melssen, W., Wehrens, R., Buydens, L.: Supervised kohonen networks for classification problems. Chemometrics and Intelligent Laboratory Systems **83** (2006) 99–113
24. Melssen, W., Üstün, B., Buydens, L.: Sompls: a supervised self-organising map - partial least squares algorithm. Chemometrics and Intelligent Laboratory Systems **86**(1) (2006) 102–120

Knowledge Discovery in Neuroblastoma-related Biological Data

Edwin van de Koppel[1], Ivica Slavkov[2], Kathy Astrahantseff[3], Alexander Schramm[3], Johannes Schulte[3], Jo Vandesompele[4], Edwin de Jong[1], Sašo Džeroski[2], Arno Knobbe[1]

Utrecht University, P.O. box 80 089, NL-3508 TB Utrecht, The Netherlands[1], Department of Knowledge Technologies, **Jožef Stefan Institute**, Jamova 39, SI-1000 Ljubljana, Slovenia[2], Center of Pediatric Oncology, **University Children's Hospital of Essen**, Hufelandstrasse 55, 45122 Essen, Germany[3], **University of Ghent**, Centre for Medical Genetics, De Pintelaan185, 9000, Gent, Belgium[4]
contact: a.knobbe@kiminkii.com

Abstract In this paper, we provide initial Data Mining results on four sets of genetic data, collected in the context of the new European Embryonal Tumour Pipeline project. These data sets provide different views on the genetic processes involved in the genesis and development of a specific type of tumour, known as neuroblastoma. Although the project involves other types of tumours as well, with potentially similar underlying causal processes, neuroblastoma is currently the only disease for which sufficient data has been collected to analyse. We provide results on this data using systems developed at two Data Mining groups in Europe, with the aim of introducing the different Data Mining challenges involved, and outlining the approach we intend to apply throughout the project. Our descriptions focus on the analysis of individual data sets, stemming from separate analysis platforms (e.g. Affymetrix microarrays). Additionally, we provide some pointers for doing cross-platform analysis in the future.

1 Introduction

In this paper we give an overview of the many Data Mining challenges involved in a new EU-funded project called the European Embryonal Tumour Pipeline (EET Pipeline in short). The EET Pipeline attempts to improve treatment of a group of cancers affecting infants and small children through channelling information extracted from high-throughput molecular profiling of these tumours into pipelines to validate targets for novel therapy and diagnostic development. As cancer is the second cause of deaths (after accidents) among children in Europe, this is an important goal, and Data Mining will play a crucial role in the extraction of knowledge from the large quantities of data produced in this project. As the project has started only recently, we do not intend to give a complete report of the results obtained, but rather provide insights in the intended approach, and show promising initial results. The aim of this paper is to outline the types of data available, the Data Mining challenges that result

from this data, and some of the techniques we are employing to deal with these challenges. Specifically, we are considering the embryonal tumour neuroblastoma (reviewed in [3]), for which extensive molecular data is already available within the project. In Section 5 of this paper, we provide initial results for this illness. As the remaining tumour types will involve similar types of data, the results reported here will give a good indication of the activities that will be performed throughout the project. However, data for these other tumour types will only become available in later stages of the project.

One of the main characteristics of the project is its integrated approach towards embryonal tumours. This integration takes the form of a unified approach across tumour types. Furthermore, for all tumours, we are gathering data through a range of high-throughput analysis platforms, providing multiple views on the biological processes involved in the development of tumours. The analysis platforms include microarrays for gene and microRNA expression, ArrayCGH for chromosomal deletion and multiplication, and mass-spectrometry for proteomics. The diverse nature of these different data sources, in terms of data structure, is a first challenge, which we address in this paper by demonstrating how our analysis techniques can be applied to individual data sources. Further challenges lie of course in the integrated analysis of data across analytic platforms, either by combining data sources into rich unified descriptions of patients and tumour tissue, or by integrating the knowledge that is extracted using platform-specific techniques. We provide some pointers as to these activities in Section 6.

The two analysis systems we are employing are the results of years of Data Mining research at Utrecht University and the Jožef Stefan Institute (JSI) respectively (affiliations 1 and 2 in the author list). Both flavours of Data Mining can be characterised by a strong emphasis on interpretability of the models created. We are focusing on results that make sense to a domain expert and that lead to new insights about the underlying genetic processes, rather than on inducing a black box with high predictive accuracy per se. Both systems are generic Data Ming tools, with broader application than just biology. The first system, Safarii [7, 11], was developed at Utrecht University and Kiminkii, a Dutch company owned by the last author. It is based on the discovery of patterns such as rules and interesting subgroups, and combining these patterns into classifiers using a number of techniques such as Pattern Teams [8] or Support Vector Machines. A specific forte of Safarii is the support for Multi-Relational Data Mining, a technique that allows the integration of data from different sources. The second system, developed in cooperation between the Katholieke Universiteit Leuven and the JSI, implements a tree-based approach known as Predictive Clustering Trees (PCTs) [1]. Such trees combine the benefits of clustering with those of tree-based classification methods. Of specific interest in this context is the ability to induce multi-target PCTs, trees that are optimised for multiple targets (e.g. tumour subtype and developmental stage) at the same time. We describe and demonstrate both systems in Section 4.

2 Neuroblastoma

Neuroblastoma is the most common extracranial solid tumour of childhood, and 88% of neuroblastoma patients are 5 years or younger. Neuroblastoma demonstrates

many features of common interest to cancer, such as spreading of the cancer and the development of resistance to chemotherapy. However, due to its manifestation early in life, it presents an excellent model to study genetically based changes leading to cancer, relatively free from the influence of environmental factors. Additionally, the embryonal tumours, to which neuroblastoma belongs, also are unique in the high incidence of spontaneous regression and differentiation of the tumours. The understanding of how this "self-cure" mechanism works may also be applicable to develop new treatment strategies for other cancers. Treatment of neuroblastomas with polychemotherapy provokes good initial response, regardless of tumour stage. However, two major problems of the current treatment regimen exist. Disseminated (cancer spread throughout the body) stage 4 tumours frequently relapse due to minimal residual disease arising from a few resistant tumour cells, resulting in poor overall survival rates (<35%). On the other hand, overtreatment of *MYCN*-nonamplified stage 2 or 3 tumours causes most of the surviving patients to suffer from significant organ toxicity or develop secondary malignancies later in life, reducing their quality of life. Novel strategies to more precisely diagnose and treat neuroblastoma are urgently needed to improve this situation. With the recent advent of high-throughput technologies, it is now possible to assess the tumour at multiple biological levels, including the genome, transcriptome and proteome. The large amounts of molecular information resulting from these analyses holds the promise of not only a better understanding of neuroblastoma biology and progression, but also the identification of molecules that can be targeted for therapy and used to better tailor treatment for a personalised diagnosis.

3 Data Sources

For neuroblastoma tumour samples and patient serum, a total of four data sets have been collected (being ArrayCGH, Affymetrix microarray, MicroRNA and SELDI Mass Spectrometry data). In this section, we will discuss the characteristics of each of these separately, and assess their potential and problems. First, we will give a description of the target concepts that we want to investigate. Then we will address the characteristics of each data set.

Target concepts for Investigation

The data from the EET Pipeline project leads to a range of potentially interesting target concepts for Data Mining. With space limitations in mind, we have selected two targets for this paper that are of interest to the domain experts: clinical course (*NBstatus*) and neuroblastoma stage (*Stage*).

Clinical Course: Domain experts rate this as being one of the most interesting target concepts for investigation. The clinical course *NBstatus* lists the patients last recorded follow up status, being either ‘alive without event’, ‘alive with relapse/primary tumour’ or ‘deceased’. Since only deceased patients in the data who died as result of a relapse or primary tumour were chosen for analysis here, we can make a binary comparison by testing ‘alive without event’ versus the rest.

If Data Mining can succeed in showing correlations between, for example, gene expression levels in the tumour or protein levels in the blood and relapse, an ‘early

warning system' can be constructed, identifying patients with a high risk of relapse before they actually suffer from it.

Stage: The INSS staging system developed for neuroblastoma tumours is the standard in Europe, the U.S. and Japan [4]. It categorises tumours into several stages based on clinical characteristics, numbered 1 through 4 with 4 being the most severe. All tumours from children under one year of age but limited metastases to liver, bone marrow or skin (never bone) are classified into a special stage, known as 4s. These patients have a very good prognosis for recovery. The majority of tumours from this patient subset undergo spontaneous regression even with little or no chemotherapy treatment. Patients diagnosed with stage 4 tumours, however, often succumb to their disease despite aggressive multimodal therapy. We attempt to determine whether Data Mining can deliver more information about molecular characteristics specific for certain clinical subgroups of neuroblastoma. As a starting point, we will only consider the task of distinguishing less severe neuroblastoma subgroups (stages 1, 2, 3 and 4s) from stage 4.

Data Sets

Affymetrix Expression Profiling Affymetrix is one type of array platform to conduct expression profiling. The probes on the microarray recognise one or more short areas of a specific gene transcript. The signals measured give information about how many RNA transcripts of which genes are present in the sample, which is a measure of gene activity. Expression was analysed in 63 primary neuroblastomas using the Affymetrix U95Av2 oligonucleotide microarrays. These data are included in the 68 patients analysed in [12]. This array measures the expression levels for a total of 12625 probes (genes).

ArrayCGH Array-based Comparative Genomic Hybridization (ArrayCGH) analyses the status of the whole genome of a tissue sample. It is known that certain segments of the DNA in the chromosomes are often altered in neuroblastomas [14, 9]. Possible genomic alterations include amplifications or deletions in distinct areas of certain chromosomes (including several genes) and even multiple copies of the complete chromosome complement in the cell (*trisomy*). ArrayCGH utilizes DNA probes of varying sizes to represent all areas of the genome in different levels of detail. These probes are Bacterial Artificial Chromosomes (BAC's), and analysis detects the number of copies of the DNA region corresponding to a BAC that is present in the tumour sample relative to the normal DNA complement of two copies. The data is represented as negative or positive real numbers, showing deletion or amplification, respectively.

Our data set includes ArrayCGH analysis of 19 primary neuroblastomas. These 19 tumours were among the 23 analysed in [14]. Unfortunately, four patients had to be disregarded in the current analysis, since *Stage* and *NBstatus* information could not be obtained. For each tumour there are 6228 attributes (the BAC's). However, the data contain many missing values. Specifically, data for certain BAC's are missing for all tumours analysed. Removing those, we end up with only 4820 attributes that have a value for at least one patient.

MicroRNA Expression Profiling The expression of small, non-coding regulatory RNAs, or microRNAs (miRNAs), can also be analysed using a microarray platform. MicroRNAs inhibit the expression of specific groups of genes via sequence specific

binding of the mRNA molecule, inhibiting translation into the protein. The probes on these types of array measure the expression of miRNAs, which are short RNA molecules (about 21-23 nucleotides long).

The data set contains measurements from 25 primary neuroblastomas. The tumours were analysed on a 2-channel cDNA array with probes for 384 miRNAs [13]. Two records come from different tissue samples from the same tumour (so there are 24 unique patients). For all patients we have the *Stage* information. Unfortunately, there is *NBstatus* information available for only 13 patients. Each patient is characterised by 384 attributes (miRNAs) indicating the deviation in activity from the average case.

SELDI Mass Spectrometry Surface-Enhanced, Laser Desorption/Ionisation Mass Spectrometry (SELDI MS) data is a different type of data. The mass spectrometer measures the amount and size (in Daltons) of all proteins in a complex protein mixture using time-of-flight (TOF) detection. The serum from 43 neuroblastoma patients at the time of diagnosis were fractionated on anion-exchange columns and profiled on metal-binding arrays (IMAC-Cu^{++}) using SELDI-MS. Both *Stage* and *NBstatus* information were available for these patients. Data from this analysis is expressed as mass-to-charge ratios (m/z). Mapping these m/z data to a specific protein identity is a non-trivial task requiring further chemical purifications and analyses of a larger sample amount. Only data produced from serum fraction 1 were used here.

4 Methods

Predictive Clustering Trees

Predictive modelling aims at constructing models that can predict a target property of an object from a description of the object. Predictive models are learned from sets of examples, where each example has the form (D, T), with D being an object description (or set of attributes of that object) and T a target property value. While a variety of representations ranging from propositional to first order logic have been used for D, T is almost always considered to consist of a single target attribute called the class, which is either discrete (classification problem) or continuous (regression problem).

Clustering, on the other hand, is concerned with grouping objects into subsets of objects (called clusters) that are similar with respect to their description D. There is no target property defined in clustering tasks. In conventional clustering, the notion of a distance (or conversely, similarity) is crucial: examples are considered to be points in a metric space and clusters are constructed such that examples in the same cluster are close according to a particular distance metric.

Predictive clustering [1], the analysis paradigm of our interest, combines elements from both prediction and clustering. As in clustering, we seek clusters of examples that are similar to each other, but in general taking both the descriptive part and the target property into account. In addition, a predictive model must be associated to each cluster. The predictive model assigns new instances to clusters based on their description D and provides a prediction for the target property T. It should be noted

that in this predictive clustering setting, the target T is not necessarily a single value, but rather a set of target attributes.

Also a distinction is made between the target attributes T and clustering attributes C. The distance measure is calculated on $C \cup T$, i.e., we produce models that are trying to correctly predict the attributes in both T and C. The difference between the T and C attributes is purely in the semantic for the end-user. The user is interested in the accuracy of the target attributes T, while the clustering attributes are included in the model building process in order to improve it. That is why in the results section we only report the accuracy of the obtained models for the target attributes T.

A well-known type of model which is used for the predictive clustering paradigm is a decision tree [10]. A decision tree that is used for predictive clustering is called a predictive clustering tree (PCT). Each node of a PCT represents a cluster. The conjunction of conditions on the path from the root to that node gives a description of the cluster. Essentially, each cluster has a symbolic description in the form of a rule (IF conjunction of conditions THEN cluster), while the tree structure represents the hierarchy of clusters.

A generic system for constructing PCTs is available in the Clus system, which can be obtained at "http://www.cs.kuleuven.be/~dtai/clus".

Safarii

Safarii [11] is a Multi-Relational Data Mining system that has been developed over the last year at Utrecht University and Kiminkii, primarily by the last author and colleagues. It includes a range of Data Mining techniques, as well as general facilities for dealing with large (multi-relational) data stored in relational databases. The primary approach for data analysis that is relevant to the domain at hand is centred around the discovery of regularities such as rules or interesting subgroups, which we will refer to in general as patterns [5, 12]. Such patterns may capture interesting, but possibly incomplete, knowledge concerning the influence of specific genes on a selected target (e.g. neuroblastoma vs. healthy), or the interaction of two or more genes, to name but a few examples. After such patterns have been discovered, they can be combined into more ambitious models of the biological processes that involve multiple patterns. Such global models can be used as classifiers in a black-box setting, for example to aid the diagnosis of tissue from new (suspected) patients. More importantly, by focussing on fairly simple and understandable patterns and the interaction between them, our approach aims to produce useful insights into the dynamics of the domain.

For combining patterns into global models, Safarii offers a number of reasonably well-known classifiers, notably Support Vector Machines (SVM) and Decision Table Majority (DTM) classifiers [8]. It is important to note that we are applying these classifiers not directly to the original data, but rather to the set of patterns that was previously discovered. In a sense, the patterns are treated as new constructed features, which are guaranteed to be predictive because they are the result of a mining operation themselves. The benefit of this approach is that the classifiers are constructed of pieces of knowledge that are intelligible and informative, compared to, for example, the application of SVMs to the data directly, which produces classifiers that are notoriously hard to interpret.

A possible downside of the pattern discovery approach is the potentially large number of patterns reported. Especially in genetic data, where it is not uncommon for many genes to be correlated, many possible patterns may be found, involving a range of genes that essentially capture the same aspect of the biological process. Safarii offers substantial facilities for dealing with this redundancy in sets of patterns. A technique known as Pattern Teams [8] selects out of the original large set of patterns, a small but informative subset of patterns, where each pattern adds something unique to the team.

Due to the small volumes of data, we are forced to work with fairly simple patterns, typically only including a single gene or location in the mass-spectrum. With larger data sets, and therefore less risk of overfitting, there is nothing that would prevent us from discovering more complex patterns. Note that possible interactions between genes are also captured during the combination into classifiers or teams, reducing the need for finding these interactions immediately. As a further limit on the complexity and expressiveness of our models, we will build Pattern Teams involving only few patterns. Small teams have the further advantage that they can be easily visualised, aiding the understanding and communication of findings.

5 Results

We have analysed all four dataset with the two systems at our disposal. In the interest of space however, we only demonstrate the results for two arbitrarily selected datasets per analysis technique: MicroRNA and SELDI-MS in the case of Safarii, and Affymetrix microarray and ArrayCGH in the case of Predictive Clustering Trees. Predictive models were built for *NBstatus* and *Stage* attributes. Additionally, we utilised the ability of PCTs for multi-target prediction and constructed predictive models which take into account other patient information (e.g. MYCN amplification). Comparisons were made between the single and multi-target prediction models.

Affymetrix (PCTs) When analysing the Affymetrix microarray data, two target attributes were taken into account: *NBstatus* and *Stage*. As it can be seen in Table 1 and Table 2, when trying to do a single target prediction for *NBstatus* and *Stage*, the accuracy obtained from the ten-fold cross-validation was a little better (for *NBstatus*) or worse (for *Stage*) than the default distribution.

In order to improve the performance when building PCTs, we included as clustering attributes other patient information which was previously shown [9] to be connected to the outcome of the disease. Those attributes were *deletion of the 1p chromosome region* and *amplification of the MYCN gene*. Figure 1 shows a PCT which is built when considering *NBstatus* as target and *1p deletion* as a clustering attribute. As any decision tree model, a PCT can be easily interpreted. The first node of the tree, with attribute *40235_at* (TNK2, 'tyrosine kinase, non-receptor, 2'), splits the samples into two groups. In the first group there are patients with 'alive without event' and 'no deletion' of the 1p chromosome region. The remaining group is split by a node (*34480_at*, CDH16, 'cadherin 16, KSP-cadherin') of the PCT that essentially distinguishes between patients that have/do not have a 1p deletion. The last node (*g32415_at*, IFNA5, 'interferon, alpha 5') further differentiates between the patients with 1p deletion that had a relapse (i.e., 'alive with relapse/primary tumour' or 'deceased') or are 'alive without event'.

From Table 1 it can be seen that including *1p deletion* and *MYCN amplification* as clustering attributes significantly improved the predictive performance of the constructed PCTs. The results in Table 2 show that for *Stage*, it is extremely difficult to build a predictive model which will surpass the default distribution (probability of the majority class), except for the last case when as a clustering attribute *NBstatus* is included. Considering the initial distribution, which is skewed, and the few Stage 4 cases, the learning of a predictive model is a difficult task.

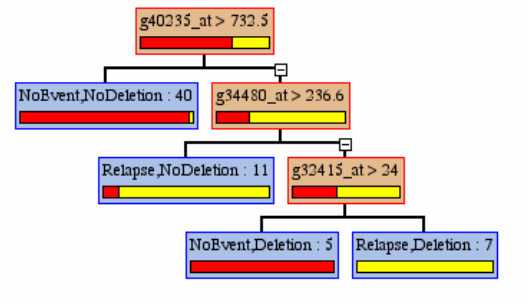


Figure 1. PCT constructed for *T = NB* status and *C = 1p*

Table 1. Results from a 10-fold cross-validation for *NBstatus*

Target and Clustering attributes	default acc. (%)	PCTs acc. (%)
T = NBstatus	71.4	74.6
T = NBstatus, C = 1p	71.4	90.5
T = NBstatus, C = MYCN	71.4	84.1
T = NBstatus, C = 1p, MYCN	71.4	74.6
T = NBstatus, C = Stage	71.4	82.5

Table 2. Results from a 10-fold cross-validation for *Stage*

Target and Clustering attributes	default acc. (%)	PCTs acc. (%)
T = Stage	79.3	77.7
T = Stage, C = 1p	79.3	73.0
T = Stage, C = MYCN	79.3	74.6
T = Stage, C = 1p, MYCN	79.3	77.7
T = Stage, C = NBstatus	79.3	80.9

ArrayCGH (PCTs) For the ArrayCGH data, a similar analysis was performed. The same target and clustering attributes were taken into account. As is evident from the results in Table 3 and Table 4, it proved to be very difficult to build PCTs with accuracy higher than the default. Including multiple attributes did not significantly improve the accuracy. The small sample size (19) and the initial class distribution

(only 3 “Stage4” samples) of this particular dataset make building accurate PCTs and predictive models difficult.

Table 3 Results from a 10 fold cross validation for *NBstatus*

Target and Clustering attributes	default acc. (%)	PCTs acc. (%)
T = NBstatus	73.6	73.6
T = NBstatus, C = 1p	73.6	78.9
T = NBstatus, C = MYCN	73.6	73.6
T = NBstatus, C = 1p, MYCN	73.6	73.6

Table 4 Results from a 10-fold cross-validation for *Stage*

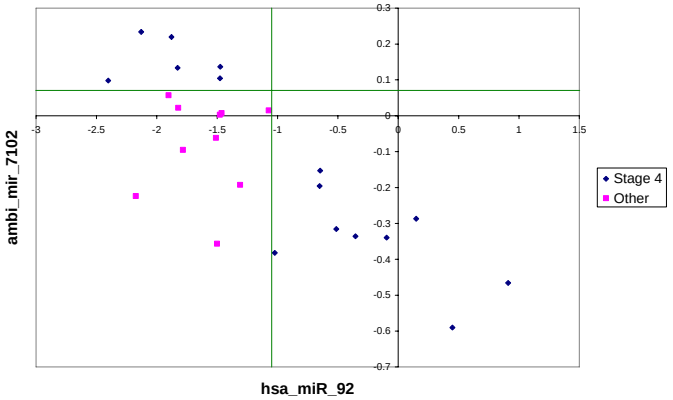
Target and Clustering attributes	default acc. (%)	PCTs acc. (%)
T = Stage	84.2	89.4
T = Stage, C = 1p	84.2	89.4
T = Stage, C = MYCN	84.2	84.2
T = Stage, C = MYCN, 1p	84.2	84.2
T = Stage, C = NBstatus	84.2	84.2

MicroRNA (Safarii) As a demonstration of the kind of knowledge that can be discovered with Safarii, we show some results for *Stage*. For the MicroRNA data, the top 100 patterns (in this case most differentially expressed probes) were identified using Safarii’s Subgroup Discovery algorithm. The resulting patterns are ranked according to the *novelty* measure (a.k.a ‘weighted relative accuracy’) [5, 7]. A minimum coverage of 6 patients was applied. For reducing the redundancy, we then applied the Pattern Team technique to the 100 patterns, producing a team of two essential probes. It reports a combination of the 2nd and 96th pattern:

Pattern Rank	Coverage	Novelty	Condition list
2	9	0.14	hsa-mir-92 ≥ -1.04
96	6	0.096	ambi-mir-7102 ≥ 0.07

A Pattern Team of size two can be easily visualised in a scatter plot, as demonstrated below. The two thresholds for the patterns involved are shown as the horizontal and vertical lines. Clearly, the lines separate the patients into three distinct clusters that appear to coincide with the target concept specified. This plot clearly demonstrates how the selected approach finds multivariate interactions that are relevant to this tumour type. Analysis of these array results using the SAM algorithm also identified hsa-mir-92 part as the most important miRNA associated with MYCN-amplified neuroblastomas (submitted). This miRNA was also identified as the first "oncomir", or miRNA which can act as an oncogene to potentially induce several tumour types [6].

Analogous results can of course be obtained for the *NBstatus*, our second target concept.

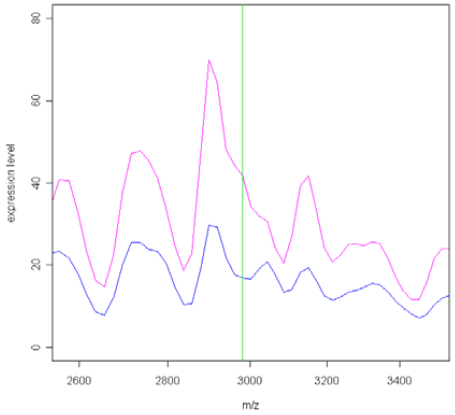


SELDI-MS (Safarii) For the SELDI-MS data some extra pre-processing was required. To reduce the effects of noise in the data as a consequence of the measurement process, a data smoothing procedure was applied, based on a Gaussian Kernel. We used the approach given by [2]. For our analyses, the kernel width was set to 101 data points (50 below the data point we want to smooth, the point itself, and 50 above). After that we reduced the resolution of the total spectrum, since it has around 56000 data points for each patient. We did this by selecting every 25th data point from the smoothed spectrum, resulting in a little over 2200 data points for each serum.

Again, the Subgroup Discovery algorithm was run with the same settings as for the MicroRNA data, creating 100 patterns for *Stage* 4 versus other stages. In the figure below, we show part of the (pre-processed) spectrum in the area of one of the patterns discovered, as an example. The two curves represent the averages of the stage 4 group (the lower line) and the remaining stages (the upper line). The vertical line corresponds to the second pattern found:

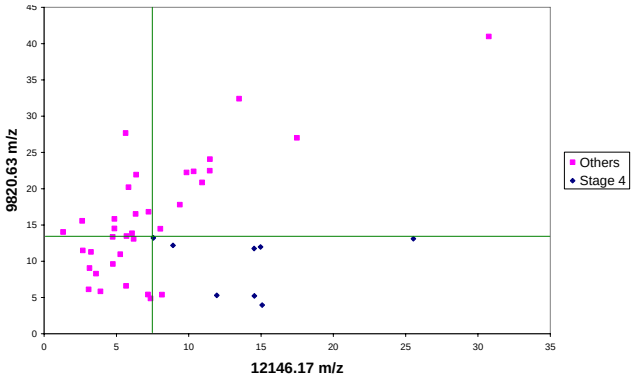
Pattern Rank	Coverage	Novelty	Condition list
2	13	0.11	2981.60 m/z ≤ 15.0

It is interesting to observe that our method does not necessarily select locations corresponding to peaks in the spectrum. Although peaks obviously correspond to specific proteins, some of which may be related to the difference between stages, apparently the exact optimum of such a peak is not guaranteed to be the most informative. As the figure demonstrates, peaks overlap to some degree, and subtle peaks may therefore not appear as actual optimums in the spectrum. The upper line in this figure shows a bump between the two adjacent peaks which is clearly missing in the stage 4 patients. Although this location does not seem promising at first hand, our method is able to identify such cases. This is in contrast to other methods (statistical and modified SVM) that have been used to analyse these data, which were incapable of analysing differences between neuroblastoma subtypes and could only be used to analyse neuroblastoma vs. healthy or related tumour patients (different targets).



We would like to add that simply taking the average over a group (as is done in the figure) does not necessarily give good insight into the distribution of individual values. As an alternative, we again show a scatter plot for a predictive pair of patterns:

Pattern Rank	Coverage	Novelty	Condition list
3	19	0.11	12146.17 m/z ≥ 7.4
24	21	0.10	9820.63 m/z ≤ 13.28



6 Conclusion and Future Developments

We have presented initial Data Mining results for a number of data sets related to neuroblastoma, in the context of the EET Pipeline project. For some of these, the methods that we used were able to construct good predictive models for the targets of

interest. For others, the small sample size and the prior distribution made the task of constructing good predictive models challenging. At this stage, we have only considered the analysis of data sets separately. The ultimate goal of the project is to combine data sets and thus obtain knowledge that spans different biological levels. As was demonstrated, there is still a considerable mismatch between the patient sets used for the different analysis platforms. This not only hinders the analysis of individual data sets, as data samples are often small, but also the integration of data sets, because the intersection of samples is even smaller. Still, with data sets becoming more complete as the project continues, integrated analysis will become important. An obvious way of integrating is to simply join data sets (over patient or tissue identifiers). Apart from scalability problems, this will be a straightforward step that will ideally lead to findings that involve cross-platform combinations of patterns. An alternative approach involves integration on the level of discovered knowledge rather than on the data level. For example, all data sets, except the SELDI-MS data, in some way map to loci on the genome. This means that if multiple data sets independently produce patterns involving the same locus, this will improve the evidence for this locus being involved in the biological process under investigation.

References

1. Blockeel, H., De Raedt, L., Ramon, J., *Top-down induction of clustering trees*. In Proceedings of ICML '98, p. 55-63, 1998
2. Brett, M., *An Introduction to Smoothing,* http://imaging.mrc-cbu.cam.ac.uk/imaging/PrinciplesSmoothing, 2006
3. Brodeur, G.M., *Neuroblastoma: Biological insights into a clinical enigma*, Nat. Rev. Cancer 3:203-216, 2003
4. Evans, A.E., D'Angio, G.J., Sather, H.N., *et al.*, *A comparison of four staging systems for localized and regional neuroblastoma: a report from the Childrens Cancer Study Group*, J. Clin. Oncol. 8:678-688, 1990
5. Fürnkranz, J., Flach, P., *ROC 'n' Rule Learning – Towards a Better Understanding of Covering Algorithms*, Machine Learning, 58, 39–77, Springer, 2005
6. He, L., Thomson, J., Hemann, M., Hernando-Monge, E., Mu, D., Goodson, S., *et al. A microRNA polycistron as a potential human oncogene*. Nature 435:828-833, 2005
7. Knobbe, A.J., *Multi-Relational Data Mining*, Ph.D. dissertation, 2004, http://www.kiminkii.com/thesis.pdf
8. Knobbe, A.J., Ho, E.K.Y., *Pattern Teams*, in Proceedings PKDD 2006, 2006
9. Maris, J.M., *The biologic basis for neuroblastoma heterogeneity and risk stratification*, Curr. Opin. Pediatr. 17:7-13, 2005
10. Quinlan, J.R., *C4.5: Programs for Machine Learning*. Morgan Kaufmann, 1993
11. *Safarii Multi-Relational Data Mining Environment*, http://www.kiminkii.com/safarii.html, 2006
12. Schramm, A., Schulte, J.H., Klein-Hitpass, L., Havers, W., Sieverts, H., Berwanger, B., Christiansen, H., *et al. Prediction of clinical outcome and biological characterization of neuroblastoma by expression profiling*, Oncogene 24:7902-7912, 2005
13. Shingara, J., Keiger, K., Shelton, J., Laosinchai-Wolf, W., Powers, P., Conrad, R., Brown, D., Labourier, E., *An optimized isolation and labeling platform for accurate microRNA expression profiling*. RNA 11:1461-1470, 2005
14. Vandesompele, J., Baudis, M., De Preter, K., Van Roy, N., *et al.*, *Unequivocal delineation of clinicogenetic subgroups and development of a new model for improved outcome prediction in neuroblastoma*, J. Clin. Oncol. 23:2280-2299, 2005

Partially-supervised context-specific independence mixture modeling

Benjamin Georgi and Alexander Schliep

Max Planck Institute for Molecular Genetics, Dept. of Computational Molecular Biology, Ihnestrasse 73, 14195 Berlin, Germany

Abstract. Partially supervised or semi-supervised learning refers to machine learning methods which fall between clustering and classification. In the context of clustering, labels can specify *link* and *do-not-link* constraints between data points in different ways and constrain the resulting clustering solutions. This is a very natural framework for many biological applications as some labels are often available and even very few labels greatly improve clustering results.
Context-specific independence models constitute a framework for simultaneous mixture estimation and model structure determination to obtain meaningful models for high-dimensional data with many, possibly uninformative, variables. Here we present the first approach for partial learning of CSI models and demonstrate the effectiveness of modest amounts of labels for simulated data and for protein sub-family determination.

1 Introduction

Historically, clustering and classification or learning from unlabeled data and learning from labeled data were considered distinct tasks in machine learning with little common ground. For several application areas however, problems occupy a middle ground between them: we will focus on examples from molecular biology and on improving clustering approaches. For example, disease sub-types are often defined by clustering patients based on clinical data; clusters and their representatives are subsequently used for predicting disease outcome or choosing optimal treatment strategies (e.g., [1]). A pure unsupervised approach has to ignore information about known sub-types, which otherwise, even if incomplete, at least provides a lower bound on the number of sub-types. Moreover, it will violate known *positive* links between patients diagnosed and confirmed to suffer from the same sub-type and *negative* links between patients diagnosed and confirmed to be afflicted by distinct sub-types. The incomplete set of sub-type labels provides constraints which should not be violated in the final clustering solution.
The same general considerations about clustering and partial information apply, if we replace patients by genes and disease sub-type by cell cycle phase [2], or if we replace patients by proteins and disease sub-type by functionally related sub-group [3]. Generally speaking, pretending complete ignorance about cluster

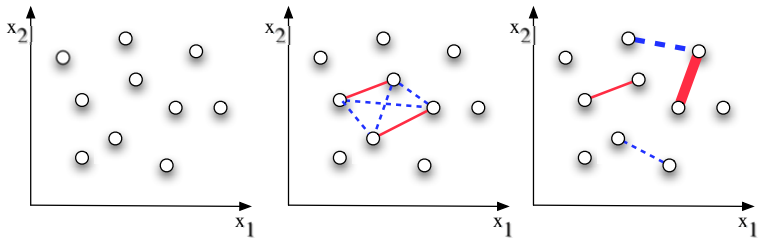


Fig. 1. Variants of partially-supervised clustering: The clustering instance of bivariate data (left) becomes easier once labels are introduced (middle). Here data points connected with a red solid line (positive constraint) share the same label. Negative constraints, indicated by dashed blue lines, result implicitly from positive constraints. A more flexible formulation (right) allows *explicit* specification of positive and negative constraints and allows to specify weights, indicated by edge weights, for the pair-wise constraints.

structure is not reflective of the availability of unlabeled mass data and sparse, labeled high quality data for a wide range of biological settings.

A recent book [4] presents a nice overview of semi-supervised learning. A lot of the literature concentrates on improving classification motivated by the observation that decrease in classification error is exponential in the proportion of labeled data [5]. Since then, a number of approaches followed the same general idea. They range from classifying text documents by constructing weighted graphs [6], partitioning graphs by min-cuts controlled by labeled examples [7], or inferring the (minimal) sub-manifold from labeled and unlabeled data and using the labeled samples for classification [8]. Cozman [9] studied how supervised mixtures get corrupted by unlabeled examples, which can also be interpreted in the framework of transductive learning [10]. More recently, a framework for integrating labeled data when learning Hidden Markov Random Fields [11] was introduced.

For clustering several variants under several names—partially supervised, semi-supervised learning, respectively constrained clustering—have been proposed. We will concentrate on clustering with mixture models [12], as mixtures have been identified as the model of choice for complex data such as gene-expression time-courses [13] and provide a sound statistical framework for extensions. The first bioinformatics application for which partial learning was proposed was concerned with improving clustering of gene expression time-courses [14]. A mixture with hidden Markov model components was trained with a variant of the expectation-maximization (EM) algorithm which essentially implemented a hard assignment of genes to clusters. The two steps of the EM are, first, computing posterior probabilities for component models given the data based on current model parameter estimates and second, estimating updated parameters from

the data where the posteriors specify the influence a particular data point has in the estimation of the parameters (see [15] for details). Recall that unlike the k-means algorithm all the data points contribute to the estimation of every component; the weighting by posterior means that ill-fitting data points contribute less. The label can be effectively used in the EM by *setting* the posterior of data points with the same label to unity for the same designated component. These explicit positive constraints (i.e., link these data points, cf. Fig. 1) do not say anything about the parameters of the designated component, they just make sure that the labeled points assigned contribute maximally to the estimation of its parameters. While data points can have distinct labels, each label corresponding to one specific component, negative constraints only arise implicitly between all pairs of data points with distinct labels. For example, it is not possible to specify two negative constraints between two pairs of data points. The advantages are an easy implementation and that the local convergence results of the EM still apply [14]. Noteworthy is the very large positive effect on clustering quality even for small quantities (less than 1%) of labels. Here, clustering quality is with respect to classification error of the data subgroups defined by the clusters and the true subgroups present in the data.

The hard assignment can be relaxed to soft assignment by specifying posterior distributions which do not put all the mass on one component. Both implementation and theory remain unchanged. However, even for the soft assignment, it is not possible to directly use information about pair-wise similarity or dissimilarity of data points, a type of information often abundant in bioinformatics, in the EM. In other words the constraints are not weighted and a reformulation in terms of the posteriors is likely cumbersome. Recently [16, 17] a new approach was proposed to use additional soft constraints for observations in the form of pair-wise positive (link) respectively negative (do-not-link) constraints w_{ij}^{+} respectively $w_{ij}^{-} \in [0, 1]$, which reflect the degree of linking for each pair of observations; cf. Fig. 1 (right).

In parallel to this development several approaches and many applications were introduced which essentially combine mixture estimation and model structure determination to improve learning on instances with many, possibly uninformative variables, with sparse data and, ultimately, arrive at more meaningful models for high-dimensional data. The central idea of these approaches is to automatically adapt model complexity to the degree of variability present in a given data set. This notion of *context-specific independence* (CSI) arose in the Bayesian network community [18–20] and has been successfully applied in mixture model framework for application such as clustering of gene expression data [21], transcription factor binding site detection [22], subtype discovery in complex genetic disease data [23] or clustering and functional annotation of protein families [3].

In the following we propose the first approach to combine CSI structure learning with the integration of prior knowledge in a partially supervised learning setup, using hard constraints on the component posteriors for labeled data.

2 Methods

2.1 CSI Mixture Models

Let $X_1, ..., X_p$ be random variables. Given a data set D with N samples, $D = x_1, ..., x_N$ with $x_i = (x_{i1}, ..., x_{ip})$ a conventional mixture density is defined as

$$P(x_i) = \sum_{k=1}^{K} \pi_k \ f_k(x_i|\theta_k), \tag{1}$$

the non-negative π_k are the mixture coefficients, $\sum_{k=1}^{K} \pi_k = 1$ and each component distribution f_k is a product of distributions over each of the X_i,(i = 1, ..., p) parameterized by parameters $\theta_k = (\theta_{k1}, ..., \theta_{kp})$,

$$f_k(x_i|\theta_k) = \prod_{j=1}^{p} P_j(x_{ij}|\theta_{kj}). \tag{2}$$

The full parameterization of the mixture is then given by $\theta = (\pi, \theta_1, ..., \theta_k)$.
For a data set D of N samples the likelihood under mixture M is simply the product of the mixture densities of each sample

$$P(D|M) = \prod_{i=1}^{N} P(x_i). \tag{3}$$

The central idea of the CSI extension to the mixture framework is that it is unnecessary to have unique parameters θ_{kj} for *all* components in *each* feature. Rather the number of parameters should be adapted to the degree of variability observed in the data. This means that multiple components share parameters for features where there is no discriminatory information for the induced grouping of the data. The CSI principle is visualized in Fig. 2. On the left side the model structure of a conventional mixture is visualized. Each cell of the matrix represents an uniquely parameterized distribution and there is a unique distribution for each component in each feature. The matrix on the right shows one possible CSI structure. Here cells spanning multiple rows represent which components share parameters in each feature. For instance for feature X_1 and X_3 components C_4 and C_5 share parameters, for feature X_2, C_1 is uniquely parameterized and for feature X_4 all components share a parameterization.
Formally the CSI mixture model is defined as follows: For the set of K component indexes $\mathcal{C} = \{1, .., K\}$ and features $X_1, ..., X_p$ let $G = \{g_j\}_{(j=1,...,p)}$ be the CSI structure of the model M. Then $g_j = (g_{j1}, ...g_{jZ_j})$ with Z_j given by the number of subgroups for X_j and each $g_{jr}, r = 1, ..., Z_j$ is a subset of component indexes from $\mathcal{C}$. That means, each g_j is a partition of $\mathcal{C}$ into disjunct subsets where each g_{jr} represents a subgroup of components with the same distribution for X_j. The CSI mixture distribution is then obtained by replacing $f_{kj}(x_{ij}; \theta_{kj})$ with $f_{kj}(x_{ij}; \theta_{g_j(k)j})$ in (2) where $g_j(k) = r$ such that $k \in g_{jr}$. Accordingly $\theta_M = (\pi, \theta_{X_1|g_{1r}}, ..., \theta_{X_p|g_{pr}})$ is the full model parameterization and $\theta_{X_j|g_{jr}}$ denotes the different parameter sets in the structure for feature j. The complete CSI model M is then given by $M = (G, \theta_M)$.

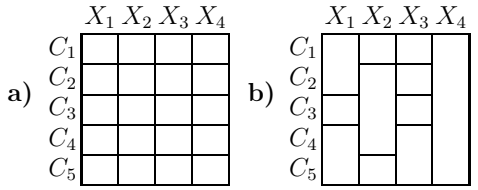


Fig. 2. Model structure matrices for a) conventional mixture model with five components over four features and b) corresponding CSI mixture model.

2.2 Partially supervised learning

The learning task in the CSI setup consist of inferring the parameterization of the mixture Θ and the CSI structure G. For the former, the standard technique is the *Expectation Maximization* (EM) algorithm [24], for the latter we apply a Bayesian approach in the structural EM framework [25, 22]. One central quantity for both of these algorithms is the posterior of component membership given by

$$\tau_{ik} = \frac{\pi_k \, f_k(x_i|\theta_k)}{\sum_{k=1}^{K} \pi_k \;\; f_k(x_i|\theta_k)}, \tag{4}$$

i.e., τ_{ik} is the probability that a sample x_i was generated by component k. In the EM algorithm the posterior is essentially a weight that determines the contribution of a sample to the parameters of a component. In the structure learning the posterior is used to compute the expected sufficient statistics of candidate structures, which then can be evaluated by the model posterior in an efficient manner (see [22] for details).
For the partially supervised case, a number of samples is assigned to components *a priori* by the labels. For a labeled sample x_i with label l this means $\tau_{ik} = 1$ for $k = l$ and 0 for all other k. This binds the contribution of the sample to parameter estimation and structure learning to a specific component. In the same way that this modification of the posterior implements partially supervised learning for the parametric EM, it gives rise to the partially supervised Structural EM algorithm in the CSI structure learning framework [21, 25, 22].

3 Results

3.1 Simulation study

In order to demonstrate the impact of a small number of labels on parameter estimation and structure learning we compared the performance of models trained with and without labels on simulated data. The generating model G was a Gaussian mixture with uniform weights on the three components over 12 features. The first two features were informative for the discrimination of the components, the remaining ten were uninformative with equal randomly chosen parameters for

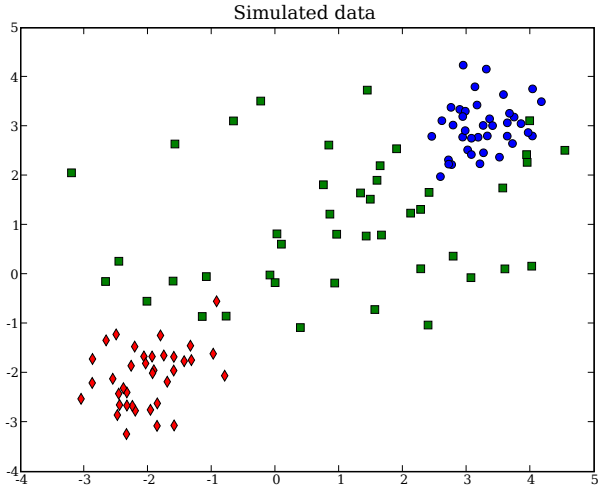


Fig. 3. Example simulated data for the first two informative features. The distinct classes are indicated by carets, rectangles and circles respectively.

all components. An example data set for the informative components is shown in Fig. 3. Two components were rather compact with diagonal covariance matrices and diagonal entries 0.5, the other component was more spread out (diagonal covariance with diagonal entries 1.5). The components with smaller variance each overlapped to a degree with the central large-variance component. The ten uninformative features provided the opportunity for the structure learning to adapt model complexity in the learned models.

We sampled 30 data sets of size 120 from G and trained CSI mixtures with and without labels. For the former three labels were used for each component. The average performance of the models over the 30 data sets with respect to the true component labels is summarized in Tab. 1

	Unlabeled	Partially Labeled
Sensitivity	92.76% SD 3.96%	92.10% SD 4.13%
Specificity	71.47% SD 15.13%	91.56% SD 4.34%

Table 1. Average sensitivity and specificity for labeled and unlabeled data over 30 simulated data sets.

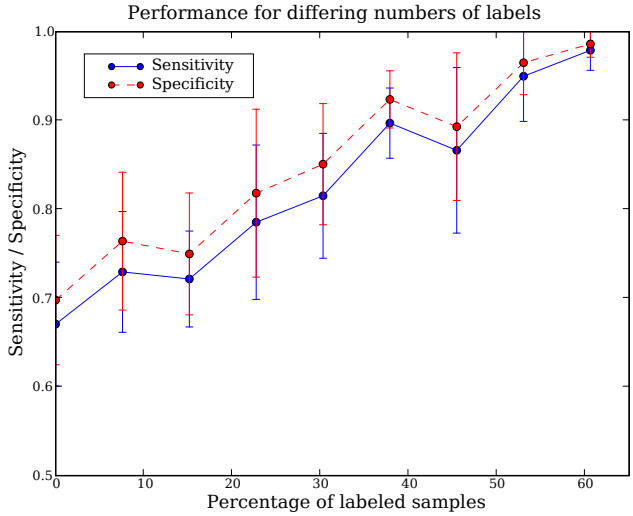


Fig. 4. Average sensitivity and specificity of clustering of the nucleotidyl cyclase data for different numbers of labels. Standard deviations are shown by error bars.

It can be seen that the addition of three labels for each components yields a considerable increase in specificity of the trained models. To assess the impact of the labeling on the structure learning we considered the edit distance of the learned structures to the true structure in G with respect to merge/split operations in the structure matrix. For instance the edit distance of Fig. 2a) to 2b) is nine since nine merges are needed to convert a) into b) (the same holds for splits in the other direction). The average edit distance of the models based on unlabeled data was 6.3 (SD 4.71), the labeled data yielded an average distance of 0.17 (SD 0.38). This indicates a greatly increased precision in the structure learning for the labeled data.

3.2 Protein sequence data

In order to examine the effect of labels in the data on a true data set we applied CSI mixture models on a multiple sequence alignment of nucleotidyl cyclase family protein sequences. We used the model extensions previously introduced for CSI for protein data [3]. The 132 sequences fall into biological subgroups of guanylyl cyclases (GC) and adenylyl cyclases (AC). We used the true classification into these subgroups as labels for the partially supervised learning. Labels were chosen randomly in equal numbers for GC and AC subgroups in steps of

five in the range 5–40, i.e. we considered data sets with 5, 10, 15, ..., 40 labels for each class. The average sensitivity and specificity for the different numbers of labels is shown in Fig. 4. It can be seen that qualitatively both sensitivity and specificity increase with the amount of prior knowledge considered, i.e. the number of labels assigned to the data set. It is noteworthy that for 60 labels (45% of the data set labeled) there is a drop in performance. This can probably be attributed to the random choice of labels. If by chance the selection of labels is poor, for instance only labels from one boundary region of a cluster, the partially supervised approach may actually mislead the parameter estimation.

4 Discussion

The results on the simulated data indicate that a partially supervised setup even for a small number of labels greatly increases the clustering performance. While sensitivity was similar for unlabeled and labeled data, the addition of labels yielded greatly increased specificity. This was the expected result from the literature on partially supervised learning. A more interesting question was how much the CSI structure learning would be impacted by the labels. The vastly smaller structure edit distance to the true CSI structure of the generating model we observed for the partially supervised case, indicates that the structure learning can also greatly benefit from the addition of labels. While this is encouraging, in the future a more in-depth evaluation using different generating models, data set sizes and number of labels will be required.
When applying the partially supervised learning on protein data the picture was somewhat more noisy, though the advantage of the labeling could still be seen. The rather high variance in results we observed can probably be attributed to the inherent noisiness of the data and the random choice of labels. Taken together the results suggest that the partially supervised learning can bring considerable improvement to both the parameter estimates and the learned CSI structure but one should be aware that in order to fulfill its potential the approach requires high-quality labels.
There are several open questions regarding the objective formulation for partially supervised learning of CSI models, in particular if pair-wise constraints need to be included, as the CSI structure controls cluster membership only indirectly and, more importantly, *not* variable-wise but rather by all variables simultaneously. This suggests that pair-wise constraints could negate the computational advantages gained by the independence assumption between variables. Nevertheless, the bioinformatics applications directly drive the need for partially supervised learning and our results show that non-trivial improvements can be realized on realistic instances from applications.

References

1. Hummel, M., Bentink, S., Berger, H., Klapper, W., Wessendorf, S., Barth, T.F.E., Bernd, H.W., Cogliatti, S.B., Dierlamm, J., Feller, A.C., Hansmann, M.L., Haralambieva, E., Harder, L., Hasenclever, D., Kuhn, M., Lenze, D., Lichter, P.,

Martin-Subero, J.I., Moller, P., Muller-Hermelink, H.K., Ott, G., Parwaresch, R.M., Pott, C., Rosenwald, A., Rosolowski, M., Schwaenen, C., Sturzenhofecker, B., Szczepanowski, M., Trautmann, H., Wacker, H.H., Spang, R., Loeffler, M., Trumper, L., Stein, H., Siebert, R.: A biologic definition of Burkitt's lymphoma from transcriptional and genomic profiling. N Engl J Med **354**(23) (Jun 2006) 2419–2430
2. Schliep, A., Costa, I.G., Steinhoff, C., Schönhuth, A.: Analyzing gene expression time-courses. IEEE/ACM Transactions on Computational Biology and Bioinformatics **2**(3) (2005) 179–193
3. Georgi, B., Schultz, J., Schliep, A.: Context-specific independence mixture modelling for protein families. In: Proceedings of the European Conference on Principles and Practice of Knowledge Discovery in Databases, Springer (2007)
4. Chapelle, O., Schölkopf, B., Zien, A., eds.: Semi-Supervised Learning. MIT Press, Cambridge, MA (2006)
5. Castelli, V., Cover, T.M.: On the exponential value of labeled samples. Pattern Recognition Letters **16** (1994) 105–111
6. Szummer, M., Jaakkola, T.: Partially labeled classification with Markov random walks. Neural Information Processing Systems (NIPS) **14** (2002)
7. Blum, A., Chawla, S.: Learning from labeled and unlabeled data using graph mincuts. In: ICML. (2001)
8. Belkin, M.: Problems of learning on manifolds. PhD thesis, University of Chicago (2003)
9. Cozman, F.G., Cohen, I., Cirelo, M.C.: Semi-supervised learning of mixture models. In: Proceedings of the Twentieth International Conference on Machine Learning (ICML 2003), Washington DC, 2003. (2003)
10. Vapnik, V.: The Nature of Statistical Learning Theory. Wiley (1998)
11. Basu, S., Bilenko, M., Mooney, R.J.: A probabilistic framework for semi-supervised clustering. In: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2004), Seattle WA, August 2004. (2004)
12. McLachlan, G., Peel, D.: Finite Mixture Models. Wiley Series in Probability and Statistics. Wiley, New York (2000)
13. Bar-Joseph, Z.: Analyzing time series gene expression data. Bioinformatics **20**(16) (Nov 2004) 2493–503
14. Schliep, A., Schönhuth, A., Steinhoff, C.: Using Hidden Markov models to analyze gene expression time course data. Bioinformatics **19 Suppl 1** (Jul 2003) I255–I263
15. McLachlan, G., Krishnan, T.: The EM Algorithm and Extensions. Wiley, New York (1997)
16. Lange, T., Law, M.H.C., Jain, A.K., Buhmann, J.M.: Learning with constrained and unlabelled data. In: CVPR (1), IEEE Computer Society (2005) 731–738
17. Lu, Z., Leen, T.: Semi-supervised learning with penalized probabilistic clustering. In Saul, L.K., Weiss, Y., Bottou, L., eds.: Advances in Neural Information Processing Systems 17. MIT Press (2005) 849–856
18. Boutilier, C., Friedman, N., Goldszmidt, M., Koller, D.: Context-specific independence in Bayesian networks. In: Uncertainty in Artificial Intelligence. (1996) 115–123
19. Chickering, D.M., Heckerman, D.: Efficient approximations for the marginal likelihood of bayesian networks with hidden variables. Mach. Learn. **29**(2-3) (1997) 181–212

20. Friedman, N., Goldszmidt, M.: Learning bayesian networks with local structure. In: Proceedings of the NATO Advanced Study Institute on Learning in graphical models, Norwell, MA, USA, Kluwer Academic Publishers (1998) 421–459
21. Barash, Y., Friedman, N.: Context-specific bayesian clustering for gene expression data. J Comput Biol **9**(2) (2002) 169–91
22. Georgi, B., Schliep, A.: Context-specific independence mixture modeling for positional weight matrices. Bioinformatics **22**(14) (2006) e166–73
23. Georgi, B., Spence, M., Flodman, P., Schliep, A.: Mixture model based group inference in fused genotype and phenotype data. In: Studies in Classification, Data Analysis, and Knowledge Organization, Springer (2007)
24. Dempster, A., Laird, N., Rubin, D.: Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society, Series B (1977) 1–38
25. Friedman, N.: The Bayesian structural EM algorithm. Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence (1998) 129–138

Using Symmetric Causal Independence Models to Predict Gene Expression from Sequence Data

Rasa Jurgelenaite[1], Tom Heskes[1], and Tjeerd Dijkstra[2]

[1] Institute for Computing and Information Sciences, Radboud University Nijmegen, PO Box 9010, 6500 GL Nijmegen, The Netherlands
{rasa, tomh}@cs.ru.nl
[2] Department of Parasitology, Leiden University Medical Center, PO Box 9600, 2300 RC Leiden, The Netherlands
t.dijkstra@lumc.nl

Abstract. We present an approach for inferring transcriptional regulatory modules from genome sequence and gene expression data. Our method, which is based on symmetric causal independence models, is both able to model the logic behind transcriptional regulation and to incorporate uncertainty about the functionality of putative transcription factor binding sites. Applying our approach to the deadliest species of human malaria parasite, Plasmodium falciparum, we obtain several striking results that deserve further (biological) investigation.

Key words: Transcriptional regulatory networks, symmetric causal independence models, Plasmodium falciparum

1 Introduction

One of the major challenges facing biologists is to understand the transcriptional regulation of genes, which is critical for the development, complexity and homeostasis of all living organisms. The introduction of DNA microarray technology [26], which enables researchers to simultaneously measure the concentration of RNA transcripts from a single sample of cells or tissues, has offered the possibility to infer large-scale transcriptional regulatory networks for various organisms. The algorithms developed for this purpose can be grouped into two general strategies: an influence strategy, which seeks to identify regulatory influences between RNA transcripts, and a physical strategy, which seeks to identify the proteins that regulate transcription and the DNA motifs to which the proteins bind [11]. In this paper, we propose a method following the latter strategy, which has the advantage of being able to combine genome sequence data and RNA expression data to enhance the specificity and sensitivity of predicted interactions.

The physical strategy methods that make use of both RNA expression data and genome sequence data rely on the assumption that genes with similar expression profiles share common regulatory mechanisms. Based on the way in which the two sources of data are related, we can distinguish three groups of these methods. The first group includes the methods that first cluster genes on

the basis of their expression patterns and then search for putative motifs in the upstream regions of the genes in each cluster. The early methods following this approach searched for individual transcription factor binding site patterns in upstream regions of the coexpressed genes (see e.g. [5, 8, 28]), while the more recent algorithms search for DNA target sites for cooperatively binding transcription factors [12, 18]. The methods in the second group work in the opposite direction, first identifying a set of candidate motifs and then trying to explain RNA expression using these motifs [7, 15, 22]. Finally, the algorithms in the last group use both sources of data together. These methods use one or more iterations of the following procedure: first, genes are clustered or grouped according to their expression data, then the search for motifs in the upstream regions of the coexpressed genes is performed, and, finally, the motifs identified are used to build models that predict the expression pattern of the gene (see e.g. [2, 27]).

A key feature of transcriptional regulation of gene expression in eukaryotes is that genes are often regulated by more than one transcription factor [30]. A number of approaches have been proposed to address the combinatorial nature of transcriptional regulation. One approach is based on the assumption that the influence of different transcription factors on gene expression is additive. The studies based on this approach use a simple linear regression to relate transcription factor binding sites to gene expression values [7, 17]. A probabilistic model by Segal et al. [27] assumes that genes are partitioned into modules, which determine the gene expression profile. The strength of the association of a gene with a module is the sum of its weighted motifs, where each weight specifies the extent to which the motif plays a regulatory role in the module. These approaches, however, cannot identify synergistic motif combinations that control gene expression patterns. Algorithms have been developed to model the synergy between two transcription factors that bind to sites located anywhere in the upstream region [22] or sites that are spatially close to each other [7, 12]. Beer and Tavazoie [2] present an approach which utilizes AND, OR and NOT logic to capture combinatorial effects of transcription factors in the regulation of gene expression. This method is not only able to infer combinatorial rules that involve more than two transcription factors, but it also includes constraints on motif strength, orientation and relative position. A similar method has been reported by Hvidsten et al. [15]. To link transcription factor binding site combinations to genes with particular expression profiles, the method extracts IF-THEN rules which correspond to AND logic.

Although the methods that model combinatorial effects of the motifs have appealing properties, their drawback is their inability to cope with uncertainty in the transcription factor binding sites that are identified. The robustness of the method in the face of uncertainty is important, as non-functional transcription factor binding sites can be readily found throughout the genome, including promoters [31]. We present an approach which is both able to model the logic behind transcriptional regulation and to incorporate uncertainty about the functionality of putative transcription factor binding sites. Our probabilistic method, which is based on symmetric causal independence models, extends the earlier methods

that infer combinatorial rules in two important directions. First, we use a broad class of Boolean functions, symmetric Boolean functions, to capture combinatorial effects of transcription factors in the regulation of gene expression. Second, the motifs contribute to the regulation of a gene through hidden variables; thus, the method is able to cope with non-functional transcription factor binding sites.

In this paper, we apply our method to Plasmodium falciparum, which is the deadliest species of the parasite that causes malaria in humans. Human malaria infects between 300 and 500 million people and causes up to 2.7 million deaths annually, mostly among young children in Sub-Saharan Africa [6]. In many endemic countries, malaria is also responsible for economic stagnation [23]. A good understanding of transcriptional regulation in this organism is important for devising new ways to disrupt the parasite's life cycle.

2 Methodology

In this section, we present our approach based on symmetric causal independence models for inferring transcriptional regulatory modules from genome sequence and gene expression data. The underlying assumption in this approach is that genes in the different clusters share common regulatory mechanisms. When trying to separate the genes in one cluster from all others, we aim to find motifs and their interactions that are specific to specific regulatory mechanisms. We start our method (Figure 1) with a 'data pre-processing' step, where we use a motif-finding algorithm to identify putative transcription factor binding motifs and we cluster genes according to their expression profiles. Then, for each cluster of genes that exhibited significant changes, we learn a symmetric causal independence model, which, given the binding sites of a gene, classifies the gene as belonging to the cluster or not. Finally, we analyze the results of experiments and identify potential transcription factors binding to the motifs that play a regulatory role in gene expression. All these steps are described in detail further in this section.

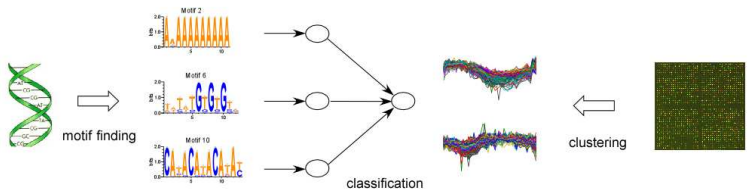


Fig. 1. Overview of the proposed approach.

2.1 Finding transcription factor binding motifs

We extracted the DNA sequence 1000 bp upstream from the initiation codon of each of 5404 Plasmodium falciparum genes using PlasmoDB release 5.2. In instances where the upstream regulatory region overlapped with another open reading frame, we extracted only the sequence between the open reading frames. To find over-represented motifs, the extracted sequences were analyzed using the AlignACE program [13]. We set the GC background parameter to 0.13 (the fractional GC background for these regions), the number of columns to align to 10 and the number of expected sites to 5.

2.2 Clustering of the RNA expression data

We used a Plasmodium falciparum 3D7 strain RNA expression data set [4]. We downloaded data that were normalized and median-centered and we only used data for those oligonucleotides that have a corresponding open reading frame assigned from PlasmoDB. We discarded the genes for which more than 20% of measurements were missing. A number of open reading frames had more than one oligonucleotide measured; we averaged the measurements of these open reading frames. After the data had been $\log_2$ transformed, we imputed missing values using the weighted K-nearest neighbours method. We chose to use this data imputation method as it has been shown to provide a more robust and sensitive missing value estimation in microarray data than a singular value decomposition based method or the commonly used row average method [29]. The weighted K-nearest neighbours method uses a weighted average of values from the K genes closest to the gene of interest as an estimate for the missing value. Based on the results reported in [29], we chose the value of K to be 15 and Euclidean distance as a metrics for gene similarity.

We used the K-means algorithm [19] with random initializations to cluster the genes according to their RNA expression data. Since the K-means algorithm is known to sometimes get stuck in a local optimum, we ran the algorithm 10 times for each number of clusters. To select the optimal number of clusters we used the so-called C-index [14], which has been shown to outperform 13 other indices for determining the number of clusters in binary data sets when the data are clustered using the K-means algorithm [10].

2.3 Learning symmetric causal independence models

The global structure of a *symmetric causal independence model* is shown in Figure 2; it expresses the idea that causes $C_1, \ldots, C_n$ influence a given common effect E through hidden variables $H_1, \ldots, H_n$ and a symmetric Boolean function f. All variables in this model are binary; the hidden variable H_i is considered to be a contribution of the cause variable C_i to the common effect E. The function f represents in which way the hidden effects H_i, and indirectly also the causes C_i, interact to yield the final effect E. To learn more about symmetric causal independence models and learning them, see [16].

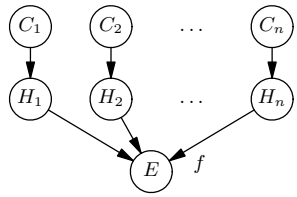


Fig. 2. Symmetric causal independence model

In this paper, we use symmetric causal independence models as a technique to model combinatorial effects of transcription factor binding motifs in the regulation of gene expression. Transcription factor binding sites are causes in this model, where the positive state of this variable is presence or absence of the motif, depending on the motif's effect on expression of genes in the cluster. The positive state of the effect variable represents gene belonging to the cluster, and the negative state represents gene belonging to any other cluster.

We used a greedy approach to select the motifs whose absence or presence contributes to the difference between the expression of genes belonging to a given cluster and the expression of the other genes. First, we ranked all motifs based on their mutual information scores, where the mutual information measures the mutual dependence of the variable M that represents a motif and the class variable C and is defined as:

$$I(M;C) = \sum_{m \in M} \sum_{c \in C} \Pr(m,c) \log \frac{\Pr(m,c)}{\Pr(m)\Pr(c)}.$$

Then, we built a model from the h highest ranked motifs. We started from a model containing only a leaky cause, then iteratively added the next highest ranked motif and evaluated the model thus obtained. If the new model did not have a higher score than the previous model, the motif was removed from the model. Since there are 2^{n+1} symmetric Boolean functions for a model with n variables that represent motifs, evaluating all the resulting models is too expensive computationally. Therefore, we restricted the interaction function space to the Boolean threshold functions. This restriction means that for every added motif we only had to evaluate two models, a gene model with the interaction function τ_k and a gene model with the interaction function τ_{k+1}, where τ_k is the interaction function from the model with the highest score. We evaluated each model using the classification accuracy on the validation set.

To solve the problem of unbalanced data (different class size, see Table 1), we added as many copies of every gene from the smaller class as was needed for this class to amount for at least half of the genes. To learn the parameters of the gene model, we ran 25 iterations of the EM algorithm, computed the classification accuracy on the validation set after each iteration and chose those parameters that provided the highest score.

2.4 Evaluation of the results

We used two error estimation methods, cross-validation and bootstrap, to evaluate the models learned. The cross-validation scheme was used to examine the predictive performance of the models, whereas the bootstrap approach was used to evaluate the reliability of the model parameters. For both methods, we performed 100 runs, and the data was split into training, validation and test sets. The validation set was used to choose the number of iterations of the EM algorithm and the threshold function; the results reported were obtained using an independent test set.

We used the results of the bootstrap approach to test for potential synergistic motif pairs. From the results of the bootstrap approach, we estimated $\hat{\boldsymbol{\theta}} = (\theta_1, \theta_2, \ldots)$, where θ_i is the probability that motif M_i will be chosen as a feature in the model. We introduce a variable X_{jk} that specifies four possible combinations of occurrence of the motifs M_j and M_k. Our null hypothesis was that X_{jk} follows multinomial distribution, with each trial resulting in one of 4 possible outcomes with probabilities $p_1 = (1-\hat{\theta}_j)(1-\hat{\theta}_k), p_2 = \hat{\theta}_j(1-\hat{\theta}_k), p_3 = (1-\hat{\theta}_j)\hat{\theta}_k, p_4 = \hat{\theta}_j\hat{\theta}_k$, and the number of trials n being equal 100. To measure the discrepancy between the observed and expected counts, we used Pearson's chi-square statistic:

$$X^2 = \sum_i \frac{(O_i - E_i)^2}{E_i}$$

where i is a possible outcome and the expected count $E_i = np_i$.

To compare our classifier to a classifier which assigns all genes to the biggest class, we used a binomial test described in [24]. The test uses the number of cases n for which the two classifiers produce a different output, and the number of cases s where the output of the examined classifier was correct, while the output of the reference classifier was wrong. Under the null hypothesis that the two classifiers perform equally well, we compute:

$$p = 2\sum_{i=s}^{n} \frac{n!}{i!(n-i)!} 0.5^n.$$

2.5 Identifying potential transcription factors binding to the motifs

To identify potential transcription factors binding to the motifs, we used comparative genome analysis, which is based on the fact that sequence similarity might reflect functional similarity. Identification, which was done separately for each motif, involves three steps. Firstly, we used STAMP [20], a web tool for exploring DNA-binding motif similarities, to find a number of the closest matches for a given motif in 13 supported databases. Secondly, for each match found, we checked whether the database where the motif is stored reports a transcription factor binding to it. Finally, if the transcription factor is known, we used BLAST [1, 25] to find the most similar protein sequences from the Plasmodium falciparum protein database.

3 Experimental Results

3.1 Transcription factor binding motifs found and clusters obtained

AlignACE found 100 transcription factor binding motifs in the given upstream sequences. The motifs that were found to be the most important features for classifying the genes will be discussed later in this section.

We chose the number of clusters to be 5, as the C-index curve had an 'elbow' at this value. Figure 3 presents the clusters obtained, which are comparable to the four characteristic stages of intraerythrocytic parasite morphology discussed by Bozdech et al. [4], as the vast majority of genes induced in every one of the stages belong to one of four clusters. Cluster 5 is a cluster of genes whose expression did not show a significant change. The correspondence among the characteristic stages and the clusters and the cluster sizes are given in Table 1.

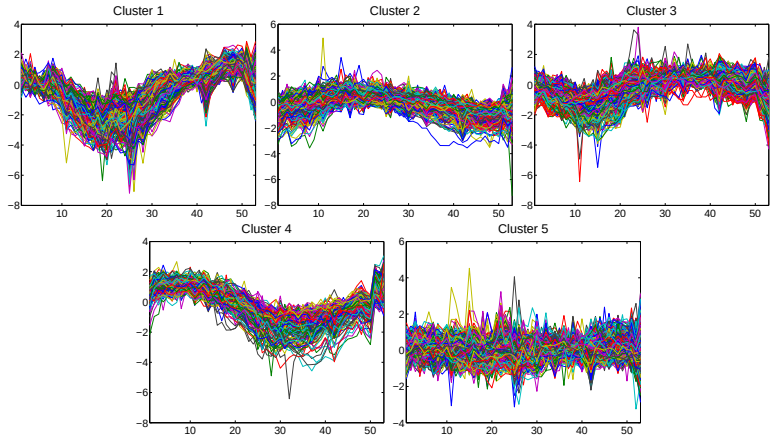


Fig. 3. Clusters of Plasmodium falciparum RNA expression data.

3.2 Models learned

We learned the models for the first four clusters, i.e. the clusters of genes whose expression changed throughout the intraerythrocytic stage.

The classification accuracy of the gene models learned using the cross-validation procedure explained in 2.4 is reported in Table 1. The p-values for the null hypothesis that the gene models perform equally well as a classifier which assigns all genes to a bigger class are less than 10^{-10}.

Table 2 lists the motifs that were most often selected as features of the gene model. Due to space limitations, we report only those motifs that were selected as

Table 1. A brief description of the clusters, the number of the genes assigned, the corresponding characteristic stage of intraerythrocytic parasite morphology; and classification accuracy obtained using the cross-validation procedure.

Cluster	Number of genes	Corresponding stage	Accuracy obtained (%)	Baseline accuracy (%)
1	329	schizont	60.48	50.79
2	1033	ring/early trophozoite	61.52	52.52
3	985	trophozoite/early schizont	59.16	50.90
4	144	early ring	63.21	51.30
5	1344	-	-	-

features of the model in more than 50 bootstrap runs. Some of the motifs appear in more than one cluster; however, their weighting is different (not shown) and they can be either 'present' or 'absent' (the presence or absence is a positive state of the corresponding variable in the model). Sequence logos of the motifs, which were generated using the WebLogo program [9], are shown in Figure 4. A study of the positive states of the variables representing the motifs selected as features of the model in more than 20 bootstrap runs reveals a distinct pattern. The variables in models for cluster 2 and cluster 4 represent the absence of the motifs, while the variables in models for cluster 1 and cluster 3 mainly represent the presence of the motifs. Even though there are 6 motifs that break this pattern in clusters 1 and 3, these motifs are found in a very small number of genes (from 1 to 5 % of genes); the other motifs selected are much more common in genes. The summary of these results is presented in Table 3.

Table 2. Motifs that were selected as features of the model in more than half of the bootstrap runs; the number of runs the motif was selected is given in parentheses. 'Present' motifs are written in roman, 'absent' motifs are written in roman.

Cluster	Motifs selected more than 50 times
1	Motif 38 (100), Motif 37 (95), Motif 6 (65), Motif 59 (65), Motif 11 (63), Motif 21 (55)
2	**Motif 6** (98), **Motif 35** (93), **Motif 37** (89), **Motif 38** (89), **Motif 11** (75), **Motif 21** (68), **Motif 59** (68), **Motif 7** (64), **Motif 80** (59)
3	Motif 6 (100), **Motif 88** (82), Motif 38 (67), Motif 59 (60), Motif 21 (51)
4	**Motif 6** (99), **Motif 11** (93), **Motif 4** (55)

Interpretation of the probabilities of the hidden variables is somewhat tricky as they highly depend on the number of input variables and the interaction function in the model, which currently vary a lot from one bootstrap run to another. Nevertheless, there is a pattern which suggests that probabilities of

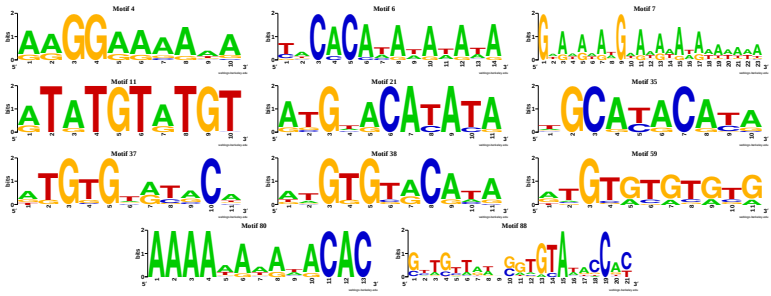


Fig. 4. Sequence logos of the motifs that were selected as features of the model in more than half of the bootstrap runs.

Table 3. Positive states of variables representing the motifs that were selected as features of the model in at least 20 bootstrap runs.

Cluster	Motifs selected	Positive state: absence	Positive state: presence
1	14	2	12
2	14	13	1
3	10	4	6
4	15	15	0

hidden variables contain information about functionality of putative transcription factor binding sites. The pattern emerges when we compare probabilities of hidden variables with the average of probabilities of the other hidden variables in the model. The probabilities in clusters of 'absent' motifs were almost the same, while the probabilities in clusters of 'present' motifs differed much more and the majority of motifs had the tendency to have corresponding probabilities below or above the average.

To find statistically significant occurrences of motif pairs, we tested all possible pairs of the motifs selected as causes in the model in at least 20 bootstrap runs (see 2.4 for the description of the method). We rejected the null hypothesis at the significance level of 0.05 (corrected for multiple testing) for two motif pairs from cluster 4: for the pair of motifs 70 and 72 (with p-value of 0.0055), and the pair of motifs 4 and 5 (with p-value of 0.0174). These motifs were selected together to be features in the model more often than it would be expected. Sequence logos for potential synergistic motif pairs are shown in Figure 5.

3.3 Potential transcription factors binding to the motifs

We present the most significant findings for the motifs reported in Figure 4.

Motifs 6, 11 and 35 have the same closest match - the binding site of fruit fly's transcription factor Topoisomerase 2, reported in FlyReg database [3]. The most

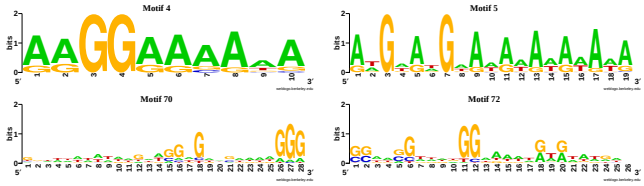


Fig. 5. Sequence logos of potential synergistic motif pairs.

significant alignment in Plasmodium falciparum is PF14_0316, putative DNA topoisomerase 2, whose protein sequence is nearly identical (E value of 0.0).

Another gene of Plasmodium falciparum that is a potential transcription factor binding to at least two of the motifs discussed is PF14_0175, which is annotated as a hypothetical protein in PlasmoDB. One of the closest matches for motif 7 is MCM1+SFF_M01051 reported in TRANSFAC database [21]. The most significant alignment for MCM1, which is yeast transcription factor involved in cell-type-specific transcription and pheromone response and plays a central role in the formation of both repressor and activator complexes, is PF14_0175 (E value of 10^{-5}). Another motif to which this transcription factor could bind is motif 80; this possible connection was found through a different transcription factor in a different organism. Motif FOXP1_M00987 reported in TRANSFAC is a close match to motif 80. Mouse transcription factor FOXP1 which binds to this motif is thought to repress expression of epithelial genes in the lung and reduce expression from promoters of mouse CC10 gene G002818. The most significant alignment for variants T04812 and T04813 of FOXP1 in Plasmodium falciparum is PF14_0175 (E value of 10^{-8}).

A gene which is found as potential transcription factor for one third of the motifs analyzed is PFL0465c, zinc finger transcription finger (krox1). For motif 4, the connection was found through motif Helios_M01004 reported in TRANSFAC and mouse transcription factor IKAROS family zinc finger 2, Helios, whose functions include zinc ion binding, DNA binding and nucleic acid binding (E value of $7 \cdot 10^{-6}$). For motif 21, the connection was found through motif CF2-II_M00012 reported in TRANSFAC and fruit fly transcription factor CF2-II, a late activator in follicle cells during chorion formation (E value of 10^{-6}).

4 Discussion and Future Work

We have presented an approach which is both able to model the logic behind transcriptional regulation and to incorporate uncertainty about the functionality of putative transcription factor binding sites. Another advantage of our technique is that it does not require other biological knowledge than genome sequence data and RNA expression data to validate the results. Since we do not use expression data while searching for putative regulatory motifs, the accuracy of the models

in predicting gene expression pattern is an unbiased measure of the soundness of the models learned.

Experimental results revealed the lack of consistency in the properties of the models learned. This inconsistency could be caused by the lack of additional constraints on the motifs, such as position relative to the translation start, orientation and functional depth. Therefore, the next step in our research is to implement normal and binomial approximations to Poisson binomial distribution, which will help to reduce computational complexity of the EM algorithm. Reduced computational complexity will enable us to test more interaction functions and to examine the additional constraints on the motifs.

We will also continue our discussions with biologists to find the explanation to the experimental results, especially, the pattern of clusters of 'present' and 'absent' motifs, and potential transcription factors binding to the motifs.

Acknowledgments. We would like to thank Michael A. Beer, Saeed Tavazoie, Zbynek Bozdech and Mahony Shaun for helpful discussions.

References

1. Altschul, S.F., Madden, T.L., Schäffer, A.A., Zhang, J., Zhang, Z., Miller, W., Lipman, D.J.: Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Research **25** (1997) 3389–3402
2. Beer, M.A., Tavazoie, S.: Predicting gene expression from sequence. Cell **117** (2004) 185–198
3. Bergman, C.M., J.W Carlson, S.E. Celniker: Drosophila DNase I footprint database: A systematic genome annotation of transcription factor binding sites in the fruitfly, D. melanogaster. Bioinformatics **21** (2005) 1747–1749
4. Bozdech, Z., Llinas, M., Pulliam, B., Wong, E.D., Zhu, J., DeRisi, J.: The transcriptome of the intraerythrocytic developmental cycle of Plasmodium falciparum. PLoS Biology **1** (2003) 85–100
5. Brazma, A., Jonassen, I., Vilo, J., Ukkonen E.: Predicting gene regulatory elements in silico on a genomic scale. Genome Research **8** (1998) 1202–1215
6. Bremen, J.: The ears of the hippopotamus: manifestations, determinants, and estimates of the malaria burden. American Journal of Tropical Medicine and Hygiene **64** (2001) 1–11
7. Bussemaker, H.J., Li, H. and Siggia, E.D. Regulatory element detection using correlation with expression. Nature Genetics **27** (2001) 167–171
8. Chu, S., DeRisi, J., Eisen, M., Mulholland, J., Botstein, D., Brown, P.O., Herskowitz, I.: The transcriptional program of sporulation in budding yeast. Science **282** (1998) 699–705
9. Crooks G.E., Hon G., Chandonia J.M., Brenner S.E.: WebLogo: A sequence logo generator. Genome Research **14** (2004) 1188–1190
10. Dimitriadou, E., Dolničar, S. Weingessel, A.: An examination of indexes for determining the number of clusters in binary data sets. Psychometrika **67** (2002) 137–160
11. Gardner, T.S., Faith, J.: Reverse-engineering transcription control networks. Physics of Life Reviews **2** (2005) 65–88
12. GuhaThakurta, D., Stormo, G.: Identifying target sites for cooperatively binding factors. Bioinformatics **17** (2001) 608–621

13. Hughes, J.D., Estep, P.W., Tavazoie S., Church, G.M.: Computational identification of cis-regulatory elements associated with groups of functionally related genes in Saccharomyces cerevisiae. Journal of Molecular Biology **296** (2000) 1205–1214
14. Hubert, L.J., Levin, J.R.: A general statistical framework for accessing categorical clustering in free recall. Psychological Bulletin **83** (1976) 1072–1082
15. Hvidsten, T.R., Wilczyński, B., Kryshtafovych, A., Tiuryn, J., Komorowski, J., Fidelis K.: Discovering regulatory binding-site modules using rule-based learning. Genome Research **15** (2005) 856–866
16. Jurgelenaite, R., Heskes, T.: EM algorithm for symmetric causal independence models. Proceedings of the Seventeenth European Conference on Machine Learning (2006) 234–245
17. Keleş, S., van der Laan, M., Eisen, M.B.: Identification of regulatory elements using a feature selection method. Bioinformatics **18** (2002) 1167–1175
18. Liu, X., Brutlag, D., Liu, J.S.: BioProspector: discovering conserved DNA motifs in upstream regulatory regions of coexpressed genes. Pacific Symposium on Biocomputing (2001) 127–138
19. MacQueen, J.B.: Some methods for classification and analysis of multivariate observations. Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability **1** (1967) 281–297
20. Mahony, S., Benos, P.V.: STAMP: a web tool for exploring DNA-binding motif similarities. Nucleic Acids Research (2007) in press
21. Matys, V., Kel-Margoulis, O.V., Fricke, E., Liebich, I., Land, S., Barre-Dirrie, A., Reuter, I., Chekmenev, D., Krull, M., Hornischer, K., Voss, N., Stegmaier, P., Lewicki-Potapov, B., Saxel, H., Kel, A.E., Wingender, E.: TRANSFAC and its module TRANSCompel: transcriptional gene regulation in eukaryotes. Nucleic Acids Research **34** (2006) D108–110
22. Pilpel, Y., Sudarsanam, P., Church, G.M.: Identifying regulatory networks by combinatorial analysis of promoter elements. Nature Genetics **29** (2001) 153–159
23. Sachs, J.D., Malaney, P.: The economic and social burden of malaria. Nature **415** (2002) 680–685
24. S. Salzberg: On comparing classifiers: pitfalls to avoid and a recommended approach, Data Mining and Knowledge Discovery **1** (1997) 317-327
25. Schäffer, A.A., Aravind, L., Madden, T.L., Shavirin, S., Spouge, J.L., Wolf, Y.I., Koonin, E.V., Altschul, S.F.: Improving the accuracy of PSI-BLAST protein database searches with composition-based statistics and other refinements. Nucleic Acids Research **29** (2001) 2994–3005
26. Schena, M., Shalon, D., Davis, R.W., Brown, P.O.: Quantitative monitoring of gene expression patterns with a complementary DNA microarray. Science **270** (1995) 467–470
27. Segal, E., Yelensky, R., Koller, D.: Genome-wide discovery of transcriptional modules from DNA sequence and gene expression. Bioinformatics **19** (2003) 1273–1282
28. Tavazoie, S., Hughes, J.D., Campbell, M.J., Cho, R.J., Church, G.M.: Systematic determination of genetic network architecture. Nature Genetics **22** (1999) 281–285
29. Troyanskaya O., Cantor M., Sherlock G., Brown P., Hastie T., Tibshirani R., Botstein D., Altman R.B.: Missing value estimation methods for DNA microarrays. Bioinformatics **17** (2001) 520–525
30. Wagner, A.: Genes regulated cooperatively by one or more transcription factors and their identification in whole eukaryotic genomes. Bioinformatics **15** (1999) 776–784
31. Werner, T.: Models for prediction and recognition of eukaryotic promoters. Mammalian Genome **10** (1999) 168-175

Identification of cooperative mechanisms in transcription regulatory networks using non-supervised learning techniques

Ana T. Freitas[1], Ana P. Ramalho[1], Carlos A. Oliveira[1], Christian S. Nogueira[1], Miguel C. Teixeira[2], Isabel Sá-Correia[2], Arlindo L. Oliveira[1]

[1] INESC-ID/IST, Lisboa, Portugal.
[2] IBB-CEBQ/IST, Lisboa, Portugal

Abstract. Accurate identification of biological processes and gene regulatory mechanisms remains an open problem that needs to be addressed, since it represents a key stepping stone in our path to the goal of systems biology. In this work, we propose a new methodology for the identification of putative cooperative regulatory mechanisms in transcription regulatory networks.
Our approach is based on the identification of biclusters in a binary regulation matrix, where each row corresponds to a given transcription factor, each column to a target gene and each matrix entry contains the value one if there is a documented (or potential) regulation between the transcription factor and the gene that correspond to that entry.
We show that a perfectly homogeneous bicluster in this matrix identifies a set of transcription factors that jointly regulate a group of genes. Less than perfectly homogeneous biclusters may also provide new clues about still unknown regulation processes. Our methodology was tested using data from a microarray analysis of the early response of the eukaryotic model *Saccharomyces cerevisiae* to the widely used herbicide 2,4-D dichlorophenoxyacetic acid.
Results obtained with this validation show that the approach has the ability to uncover groups of genes and transcription factors that are jointly involved in a number of relevant biological processes. More significantly, our method has the potential to identify a number of possibly interesting new hypotheses of regulatory mechanisms, that may be validated experimentally.

1 Introduction

The combinatorial transcriptional regulation of gene expression in eukaryotes is highly complex and often occurs through the coordinated action of multiple transcription factors (TFs) [1]. Groups of transcription factors often cooperate in order to regulate genes controlling different cellular functions under different conditions [2, 3]. These groups of TFs can be viewed as modules, i.e., functional units that perform specific functions. Different modules will contain different sets of TFs, but, in general, there will exist significant overlap between modules.

The identification and study of regulatory modules is very important from a variety of viewpoints. On one hand, the function of each regulatory module is important

in itself, and it may be worthwhile to devote a significant effort to understand the cooperative mechanisms involved in the coordinated regulation of genetic expression. On the other hand, modules may be preserved across species, and may even be viewed as basic building blocks of biological systems.

Regrettably, the complex physical and biochemical mechanisms involved in coordinated regulation make it difficult to identify regulatory modules directly from first principles, and even sophisticated experimental methods are difficult to apply directly to the identification of cooperative regulatory mechanisms.

For these reasons, computational biology approaches that help researchers identify and analyze cooperative regulatory mechanisms are essential, and will represent a fundamental tool in our quest for the understanding of organisms as biological systems.

In this work, we propose a method that accepts as its input a regulation matrix, which is a concise way to represent known and potential regulatory associations between TFs and genes. The regulation matrix is a binary matrix where rows represent TFs and columns represent genes. A regulatory association between a TF and a gene is represented by 1 in the corresponding entry of the matrix. If no association exists, the entry contains the value 0.

The key observation that supports our method is that submatrices (of the regulation matrix) containing only entries with the value 1 correspond to a set of transcription factors that jointly regulate a specific set of genes. Submatrices of this type have been extensively studied and are usually called biclusters, although many other designations have been proposed in the literature, such as *formal concepts*, co-clusters and subspace clusters [4].

In our experiments, we have used regulation matrices for the yeast *S. cerevisiae*, obtained from the information in a publicly available database dedicated to the transcription regulatory associations in this organism, YEASTRACT [5].

2 Related Work

Understanding the complex interactions involved in the regulation of genes is crucial, if one wishes to understand and model regulatory mechanisms and gene regulatory networks. It is particularly important to be able to identify regulatory modules, i.e., closely interconnected sets of transcription factors and target genes that are commonly active under specific conditions and that are, in many case, reused and even conserved across species.

2.1 Computational Approaches for the Identification of Regulatory Mechanisms

Two computational approaches that are essentially independent and orthogonal, but also complementary, have been pursued to date in order to understand the modular organization of the transcriptional regulatory networks in different organisms [6–9].

The first approach aims at identifying regulatory mechanisms by analyzing directly the patterns of gene expression obtained using high throughput techniques,

such as microarrays. Clustering [10] and blustering [11, 4] have been extensively used to analyze microarray gene expression data, in an attempt to gain insight into complex regulatory mechanisms.

The second approach aims at identifying regulatory mechanisms by directly analyzing data obtained from direct analysis of promoter regions, or from know regulation mechanisms, identified using, for instance, ChIP-chip methods [1]. Computational methods that look directly at sequence data aim at identifying over-represented motifs in the promoter regions of the genes of interest [12–15]. By identifying which transcription factors recognize a specific motif in the promoter regions of genes, it is possible to identify potential regulatory mechanisms, that can then be biologically validated. However, this methodology is somewhat limited in its ability to identify cooperative regulatory mechanisms since it can only tell, at best, whether a specific pairwise interaction exists between one transcription factor and one gene, and not whether a set of transcription factors interact to regulate a gene.

Recent research in data collection and analysis [9] has shown that the combination of these two approaches leds to gain new insights into the network and a better definition of transcriptional modules.

Our approach uses known information about the interaction of a given transcription factor and a gene. However, it goes further than existing methods because it identifies potential cooperative, joint or alternative regulation mechanisms, by looking not at pairwise interactions between transcription factors and target genes, but at interactions between groups of transcription factors and groups of target genes.

2.2 Biclustering Algorithms

A large number of approaches has been proposed for the identification of biclusters in matrices. Biclustering is a non-supervised approach that performs simultaneous clustering on the row and column dimensions of a data matrix. The concept of biclustering can be traced back to the seventies [16] and was applied to several domains. The first application of biclustering to computational biology was the work of Cheng and Church [11]. Since, in its most general setting, biclustering is an NP-complete problem, heuristic approaches are used to obtain suboptimal solutions using reasonable computational resources.

Biclustering algorithms may identify one or several biclusters at a time. Several approaches identify only one bicluster at a time [11, 17], but then mask it with random noise and repeat the procedure in order to find other biclusters.

Other methods discover several biclusters at once [16, 18], or even all of them in parallel [19, 20]. In our approach we are interested in the identification of several, potentially overlapping, biclusters in a binary matrix. For this task an algorithm called TRYBO (Transcriptional RegulatorY Bicluster IndentifcatiOn) was developed.

3 Methods and Algorithms

Our approach is based on the idea that regulatory modules will correspond to specific structures in the regulation matrix.

We describe the approach by first defining the regulation matrix and the concept of biclusters in this matrix, and then showing why these biclusters will correspond to regulatory mechanisms of interest.

3.1 Biclusters in the Regulation Matrix

Consider an n by m binary (0-1) regulation matrix A, with the set of rows $X = \{x_1, .., x_i, .., x_n\}$ and the set of columns $Y = \{y_1, .., y_j, .., y_m\}$. Each row i represents a TF and each column j represents a gene. When TF i regulates gene j, i.e., when there is a regulatory association, element a_{ij} takes the value 1 and it takes the value 0 otherwise.

We use (X, Y) to denote matrix A. Considering that $I \subseteq X$ and $J \subseteq Y$ are subsets of rows and columns, respectively, $A_{IJ} = (I, J)$ denotes the sub-matrix of A that contains only the elements a_{ij} belonging to the sub-matrix with the set of rows I and the columns J.

The data matrix A can be viewed as a representation of a bipartite graph. A graph $G = (V, E)$ where V is the set of vertexes and E is the set of edges, is said to be bipartite if its vertexes can be partitioned into two sets L and R such that each edge of E has one end on L and the other in R. We can say that L is the set of TF, R is the set of genes and E is the set of regulatory associations between TFs and genes.

We define a bicluster as a subset of TF and a subset of genes that have regulatory associations between them. Given a data matrix, A, we want to identify a set of maximal biclusters $B_k = (I_k, J_k)$. Each maximal bicluster corresponds directly to a biclique[1] in the bipartite graph. Therefore, the problem of finding a maximal bicluster in a binary matrix is equivalent to the problem of finding a biclique with the maximal number of edges, which is a known NP-complete problem.

3.2 Biclusters and Cooperative Regulation Mechanisms

Each bicluster, in the regulation matrix, identifies a subset of transcription factors that regulate a subset of genes, and this can provide evidence that either there exists a cooperative or coordinated regulation mechanism.

Biclusters in Documented Regulation Matrices We first consider the case where the regulation matrix is obtained from documented evidence of pairwise regulations. In this case, there is a 1 in entry i, j of matrix A if there is a known and documented regulation relationship between $TF\ i$ and target gene j. It is straightforward to verify that each submatrix of matrix A that consists solely of ones corresponds to a group of transcription factors that regulate a given group of genes.

Given what we know of regulation mechanisms and the randomness of evolutionary changes, such a joint regulation will correspond to either a) some form of a cooperative regulatory mechanism, corresponding to a regulatory module or b) the

[1] A biclique is a complete bipartite graph where every vertexes of the first set is connected to every vertex of the second set.

replication of a structure in the promoter region, such as a binding site, that is used by more than one transcription factor to regulate the target genes.

In many cases, other phenomena, including, possibly, RNA interference or post-transcriptional regulation, may lead to a change in this regulatory module, that will suppress a particular regulation between a transcription factor and a gene. This means that not all regulatory modules will correspond to perfect biclusters. Some will correspond to imperfect biclusters that also represent important regulatory mechanisms.

Such imperfect biclusters may also occur in the regulation matrix by another important reason, namely that the missing regulation exists but is not yet known or has not yet been documented.

Finding (slightly) imperfect biclusters may, therefore, lead to important insights in the cooperative gene regulation mechanisms, either by leading researchers to find novel and previously unknown regulations, or by pointing to pre or post-transcriptional phenomena that explain why a given regulation that was expected is not in fact present in the regulation matrix.

Biclusters in Potential Regulation Matrices The situation is somewhat different when one is analyzing potential regulation matrices. In a potential regulation matrix, there is an entry with value 1 in position (i, j) of matrix A if there is a potential regulation between transcription factor i and target gene j. The data is obtained by scanning the promoter regions of the target genes and checking for the existence of a structure that indicates a possible binding site for a given transcription factor.

The difference between this case and the case with documented regulation matrices arises because available models for the process that leads a transcription factor to bind to the promoter region are not precise enough to actually determine, with any significant precision, whether binding will take place. A number of models has been proposed [12] but they fall mainly into two categories: consensus based and probabilistic.

Consensus based models define a consensus, i.e., a partially specified sequence of DNA bases that describes possible binding sites for a given transcription factor. Probabilistic models use, instead, a Position Weight Matrix (PWM) that describes the binding site and consider that a binding may take place if the match between the PWM and the actual DNA sequence in the promoter region is higher than a given threshold.

In practice, both models are relatively poor predictors of actual bindings, and they usually err on the optimistic side, predicting that a binding will take place when, in fact, no such binding exists. This means that, in general, there will exist many entries in the potential regulation matrix that do not correspond to the actual existence of regulation between a transcription factor and a gene.

3.3 An Heuristic Approach to Biclustering: TRYBO

As described above, our objective is the identification of (possibly imperfect) biclusters in the regulation matrix. Although a number of approaches could have been used, the biclustering algorithm selected to support this methodology was TRYBO (Trancriptional RegulatorY Bicluster identificatOn), inspired in the FLOC method

[19]. FLOC is an heuristic approach that identifies of several overlapping coherent biclusters in matrices of real values. Although TRYBO is a FLOC like algorithm, a new metric was defined to deal with binary matrices. The metric was developed to direct the algorithm towards the identification of large, close to perfect, biclusters. TRYBO is also based on the bicluster definition used by Cheng and Church [11], where a bicluster is a submatrix of constant values. In this context most entries must be 1. Unlike Cheng and Church method, TRYBO can discover a set of k, possibly overlapping, biclusters, simultaneously. For these reasons, it was not possible to use an existing biclustering algorithm and TRYBO was developed specifically for this purpose.

TRYBO has three phases. In the first phase, k initial biclusters are generated. For each initial bicluster, each row/column of the matrix is added to the bicluster with independent probability ρ.

The second phase of the algorithm is an iterative process that continuously improves the quality of these biclusters. During each iteration, each row and column is examined to determine the best action that can be taken in order to reduce the average score of biclusters. For each row/column there are k possible actions to take, that correspond to the change of membership of the row/column in each bicluster: a row can be added to the bicluster if it is not yet included in it, or can be removed if it already belongs to it. Each action has a gain associated. The gain of an action is defined as the difference between the current score of the bicluster and the score of the new bicluster that is obtained by executing the action. Actions are then tried in a random order, and the action that leads to a larger gain is selected and applied.

The third and last phase starts when no improvements are observed after a given number of iterations. The algorithm then enters a final greedy optimization phase, where each action is tested and the action that exhibits more gain is applied. This greedy phase continues until no more improvements are observed.

Extensive experimentation has shown that this combination or random action selection, in phase two, coupled with the greedy method, in phase three, leads to a more effective method than what is obtained by pursuing a strictly greedy approach.

To direct the algorithm towards the identification of nearly constant biclusters, we used the following merit function, M, to evaluate the value of a given set of biclusters:

$$M = \sum_k m_k, \tag{1}$$

where m_k is the value of each individual bicluster k, given by

$$m_k = \sum_{i \in I_k, j \in J_k} a_{ij} - \frac{V_k \sigma_k}{\mu_k}, \tag{2}$$

where I_k and J_k are, respectively, the rows and columns in bicluster k, V_k is the volume (number of rows times number of columns in bicluster k) of the bicluster, σ_k is the standard deviation of the elements in the bicluster, and μ_k is the average value of the elements in bicluster k,

$$\mu_k = \frac{\sum_{i \in I_k, j \in J_k} a_{ij}}{V_k}. \tag{3}$$

This metric takes a maximum value of $\sum_{i \in I_k, j \in J_k} a_{ij}$ for a perfect bicluster where all entries in (I_k, J_k) are 1. This maximum value increases with the size of the bicluster and decreases with added variation within the bicluster, directing the algorithm towards large, close to perfect, biclusters.

3.4 Statistical Significance of Biclusters

Since the method may generate a large number of biclusters, it is important to be able to assess the relative importance of each bicluster. For each bicluster, we computed a p-value, by computing the probability that, under the null hypothesis of independent entries in the regulation matrix, a bicluster of that size is generated. The Bonferroni's correction for multiple testing was used.

For this, we compute p,

$$p = \sum_{i=1}^{|I_k|} \frac{G_i}{N}, \tag{4}$$

where G_i is number of genes regulated by transcription factor i and N the total number of genes, in the matrix A, and obtain the p-value by computing the tail of the binomial distribution,

$$p - value = \sum_{j=|J_k|+1}^{N} \binom{N}{j} p^j (1-p)^{N-j}. \tag{5}$$

4 Experiments and Results

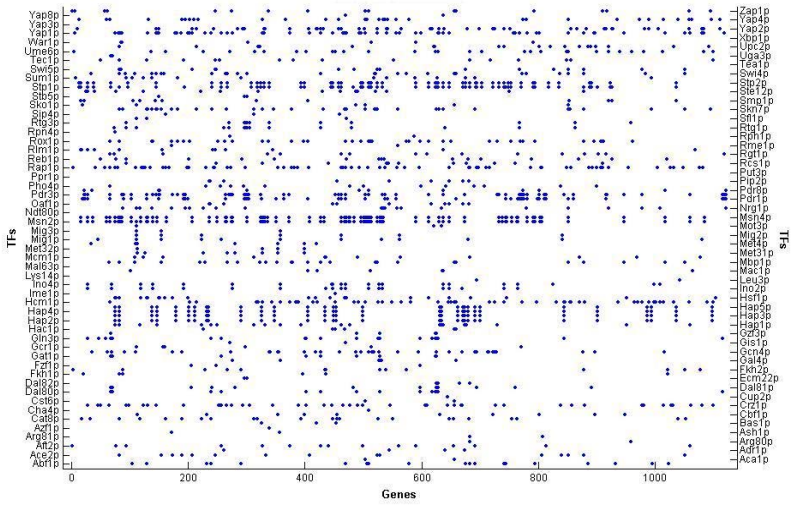
This section describes the application of the proposed methodology to data obtained from a microarray analysis of the early response of *Saccharomyces cerevisiae* to a sudden aggression with toxic concentrations of the widely used herbicide 2,4-dichlorophenoxyacetic acid (2,4-D) [21]. This environmental genomics approach has provided mechanistic insights into the toxicity and resistance to this widely used herbicide [22].

A dataset was created from the treated microarray data, including the yeast genes whose transcript level increased at least 1.5 fold in response to stress induced by 2,4-D. This dataset includes 1126 genes and 102 transcription factors. To generate the documented and potential matrices, we used the utility *Generate Regulation Matrix* from the YEASTRACT database [5].

4.1 Analysis of the Biclusters in the Documented Matrix

The documented matrix contains 102 rows, corresponding to all the characterized transcription factors listed in the YEASTRACT database, and 1126 columns, corresponding to the genes that have shown significant response to the herbicide 2,4-D. Figure 1 illustrates the documented regulation matrix obtained.

Fig. 1. Illustration of the documented regulation matrix for the 2,4-D aggression experimental data.



The TRYBO algorithm has identified 64 biclusters[2]. These biclusters were the object of analysis, with the aim of identifying relevant phenomena, either new or known.

The biclusters obtained with higher scores (lower p-values), group homologous transcription factors and the corresponding shared target genes. One of these biclusters (bicluster 1) includes the transcription regulators Hap2p, Hap3p, Hap4p and Hap5p which are subunits of a single operational transcription factor and, thus, regulate precisely the same target genes. The remaining high score bi-clusters group pairs of homologue transcription factors (e.g. Msn2p and Msn4p, Pdr1p and Pdr3p, Nrg1p and Nrg2p or Stp1p and Stp2p), whose target genes are mostly overlapping. Although expected, these high score biclusters validate the accuracy and efficacy of the approach proposed in this work and suggest that these transcription factors do play a role in the yeast response to 2,4-D. Interestingly, the main regulators of the Environmental Stress Response (ESR) program, Msn2p and Msn4p, were shown to play a role in the resistance to 2,4-D, controlling the expression of approximately 20% of the up-regulated genes, most of them encoding general stress responsive genes, heat shock proteins, chaperones and antioxidant enzymes [23]. The homologous transcription factors Pdr1p and Pdr3p, implicated in Pleiotropic Drug Resistance (PDR) and determinants of yeast resistance to 2,4-D, promote the transcription of the *TPO1*

[2] The complete list of biclusters obtained is available as supplementary material, at `http://tahoe.inesc-id.pt/ecml07/documented.html`.

and *PDR5* genes. These encode two multidrug resistance (MDR) transporters of the Major Facilitator and ATP-Binding Cassette superfamilies, postulated to actively export the herbicide counterion, thus reducing the intracellular concentration of the toxicant [24]. Stp1p and Stp2p are involved in yeast response to amino acid availability that is strongly limited in yeast cells challenged with the herbicide [25]. The pair of homologous transcription factors Nrg1p and Nrg2p and the transcription factor complex comprised of Hap2p, Hap3p, Hap4p and Hap5p are involved in the response to glucose exhaustion, a phenomenon that appears to be sensed by 2,4-D challenged cells.

A very interesting set of biclusters are those clustering the two pairs of transcription factors, Yap1p and Hsf1p and Yap1p and Rpn4p. Yap1p is associated with the oxidative stress response, Hsf1p regulates the heat shock response and Rpn4p controls the biosynthesis of subunits of the proteosome and may regulate multidrug resistance via the proteosome protein genes, which is consistent with conceivable protein inactivation by 2,4-D, with the consequent increased protein degradation via the 26S proteosome. Although a significant cooperation between these transcription factors could not be anticipated, the application of TRYBO led to the suggestion that Yap1p may interplay with Hsf1p and Rpn4p in the yeast response to 2,4-D. Yap1p and Hsf1p are pointed out as co-regulators of 17 of the 2,4-D-induced genes, most of them encoding heat shock proteins with chaperone activity. These proteins play an important role in protein protection and renaturation both in the heat shock response, mediated by Hsf1p, and in the oxidative stress response, mediated by Yap1p. The appearance of *RPN4* and *UBI4*, encoding the ubiquitin which binds to proteins marking them for selective degradation, among the genes co-regulated by Yap1p and Hsf1p is also noteworthy, since it assigns to Yap1p and Hsf1 a cooperative role in protein degradation through the proteosome. Yap1p to Rpn4p were clustered together with 19 of the 2,4-D-induced genes, including 2 involved in pleiotropic drug resistance (*CIN5* and *TPO4*) and 3 involved in protein degradation (*LAP4*, *RPN4* and *UBI4*). This group also includes a large number of uncharacterized ORFs, whose functional analysis may profit from all these indications.

In agreement with the indications provided by the biclustering data, the association of Yap1p and Hsf1p was described for the first time during 2006, in a study focused on the participation of both transcription factors in the regulation of multidrug resistance and protein degradation through the transcription factors Pdr3p and Rpn4p [3]. The same study also associates Yap1p with Rpn4p, in agreement with this biclustering analysis.

4.2 Analysis of the Biclusters in the Potential Matrix

We have also obtained the potential regulation matrix for this group of genes and transcription factors, and performed a (necessarily limited in scope) analysis of the biclusters obtained. To generate this matrix, we considered that a transcription factor regulates a gene if the (known) consensus factor of that transcription factor appears at least twice in the promoter region of a target gene. This matrix was also generated using the facility available in the YEASTRACT database.

The analysis of the 94 biclusters obtained by applying TRYBO to the potential regulatory matrix[3] revealed a particularly interesting case, bicluster 7, with a low p-value of 1.5E-15, grouping the transcription factors Pdr1p, Pdr3p and Pdr8p with 19 potential target genes. The three transcription factors are members of the Zn2Cys6 family of transcription regulators and play an important role in the control of the Multiple or Pleiotropic Drug Resistance (PDR) phenomenon in yeast [26, 27]. Based on microarrays analysis, Pdr1p and Pdr3p are documented regulators of over one hundred genes, while the more recently described Pdr8p appears to regulate a narrow range of target genes [27]. Among the 19 target genes emerging from this analysis only two, *PDR15* and *YGR035C*, are documented targets of the 3 transcription factors [27], while 12 are documented targets of both Pdr1p and Pdr3p. There are four other documented targets of the three transcription factors [27] that failed detection through our analysis, due to the fact that they do not fit in the more restricted definition of potential target genes used in this work. Indeed, in order to decrease the false positives in the potential regulation matrix and increase the reliability of the obtained biclusters, the definition of potential target genes was restricted to those presenting at least two copies of each transcription factor binding site in their promoter region and the missing genes only have one binding site in their promoter regions. This indicates that the imposed restriction, although considered necessary, may lead to false negatives. The functional role of the 19 target genes is also consistent with the biological role of the associated transcription factors. One half of the obtained target genes of known function are involved in MDR, including two transcription factors (*PDR3* – known to be autoregulated - and *RPN4*), three multidrug transporters of the ABC (ATP-Binding Cassette) Superfamily, (*PDR5*, *PDR10* and *PDR15*), and two genes involved in phospholipid transport, (*PDR16* and *RSB1*).

Although a number of the clustered genes appears to have cellular functions not directly related with the PDR phenomenon, a possible connection was recently established for Rpn4p. This transcription factor, described as a regulator of proteosome biogenesis, was recently shown to be, itself, under the regulation of Pdr1p and Pdr3p, thus relating proteosomal activity with drug resistance [28]. Guided by this biclustering analysis, it may be postulated that the biological function of 7 of the 19 genes in this bicluster, still classified as "of unknown function", may also be related with the PDR phenomenon.

5 Conclusions and Future Directions

In this work, we proposed a new method for the identification of cooperative regulatory mechanisms, that uses a non-supervised learning approach based in biclustering techniques to find sets of genes that are cooperatively or jointly regulated by sets of transcription factors. The results have shown that the application of this methodology to data obtained from microarray analysis of the early response of the eukaryotic model *Saccharomyces cerevisiae* to a widely used herbicide 2,4-D dichlorophenoxyacetic acid was able to discover interesting patterns, that, when analyzed, are mean-

[3] The complete list of biclusters is available as supplementary material at `http://tahoe.inesc-id.pt/ecml07/potential.html`.

ingful and related with interesting phenomena, both well known or recently discovered. Other patterns are likely to correspond to still unknown regulatory mechanisms, and deserve further experimental analysis.

The most interesting direction for future work is related with the integration of biological knowledge with sequence and expression data, in order to identify, with high accuracy and good coverage, regulatory modules. From a biological standpoint, we also plan to pursue a more detailed analysis of some of the hypotheses that were raised by the application of the methodology proposed in this work.

6 Acknowledgements

This work was supported by FEDER, FCT and the POSI and POCTI programs (projects POSI/EIA/57398/2004, POCTI/BIO/56838/2004).

References

1. Lee, T.I., Rinaldi, N.J., Robert, F., Odom, D.T., Bar-Joseph, Z., Gerber, G.K., Hannett, N.M., Harbison, C.T., Thompson, C.M., Simon, I., Zeitlinger, J., Jennings, E.G., Murray, H.L., Gordon, D.B., Ren, B., Wyrick, J.J., Tagne, J.B., Volkert, T.L., Fraenkel, E., Gifford, D.K., Young, R.A.: Transcriptional regulatory networks in *Saccharomyces cerevisiae*. Science **298**(5594) (2002) 799–804
2. Akache, B., MacPherson, S., Sylvain, M., Turcotte, B.: Complex Interplay Among Regulators of Drug Resistance Genes in *Saccharomyces cerevisiae*. Journal of Biological Chemistry **279**(27) (2004) 27855–27860
3. Hahn, J., Neef, D., Thiele, D.: A stress regulatory network for co-ordinated activation of proteasome expression mediated by yeast heat shock transcription factor. Molecular Microbiology **60**(1) (2006) 240–251
4. Madeira, S.C., Oliveira, A.L.: Biclustering algorithms for biological data analysis: a survey. IEEE/ACM Transactions on Computational Biology and Bioinformatics **1**(1) (2004) 24–45
5. Teixeira, M., Monteiro, P., Jain, P., Tenreiro, S., Fernandes, A., Mira, N., Alenquer, M., Lourenço, A., Freitas, A., Oliveira, A., Sá-Correia, I.: The YEASTRACT database: a tool for the analysis of transcription regulatory associations in *Saccharomyces cerevisiae*. Nucleic Acids Research **34** (2006) D446–D451
6. Ihmels, J., Bergmann, S., Barkai, N.: Defining transcription modules using large-scale gene expression data. Bioinformatics **20**(13) (2004) 1993–2003
7. Segal, E., Shapira, M., Regev, A., PeÇer, D., Botstein, D., Koller, D., Friedman, N.: Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. Nature Genetics **34**(2) (2003) 166–76
8. Manke, T., Dieterich, C., Vingron, M.: Detecting functional modules of transcription factor binding sites in the human genome. In: Regulatory Genomics. (2004) 14–21
9. Lemmens, K., Dhollander, T., De Bie, T., Monsieurs, P., Engelen, K., Smets, B., Winderickx, J., De Moor, B., Marchal, K.: Inferring transcriptional modules from ChIP-chip, motif and microarray data. Genome Biology **7**(R37) (2006)
10. Eisen, M.B., Spellman, P., Brown, P., Botsteim, D.: Cluster analysis and display of genome-wide expression patterns. Proceedings of the National Academy of Sciences **25**(95) (1998) 14863–14868

11. Cheng, Y., Church, G.M.: Biclustering of expression data. In: Proceedings of the 8th International Conference on Intelligent Systems for Molecular Biology. (2000) 93–103
12. Sandve, G., Drablos, F.: A survey of motif discovery methods in an integrated framework. Biology Direct **1** (2006) 11
13. Bailey, T.L., Elkan, C.: Unsupervised learning of multiple motifs in biopolymers using expectation maximization. Machine Learning **21**(1-2) (1995) 51–80
14. Carvalho, A.M., Freitas, A.T., Oliveira, A.L., Sagot, M.F.: An efficient algorithm for the identification of structured motifs in DNA promoter sequences. IEEE/ACM Transactions on Computational Biology and Bioinformatics **3**(2) (2006) 126–140
15. Mendes, N., Casimiro, A., Santos, P.M., Sá-Correia, I., Oliveira, A.L., Freitas, A.T.: Musa: a parameter free algorithm for the identification of biologically significant motifs. Bioinformatics **22**(24) (2006) 2996–3002
16. Hartigan, J.A.: Direct clustering of a data matrix. Journal of the American Statistical Association **67**(337) (1972) 123–129
17. Sheng, Q., Moreau, Y., Moor, B.D.: Biclustering micrarray data by Gibbs sampling. Bioinformatics **19 (Suppl. 2)** (2003) 196–205
18. Getz, G., Levine, E., Domany, E.: Coupled two-way clustering analysis of gene microarray data. Proceedings of the National Academy of Sciences **97**(22) (2000) 12079–12084
19. Yang, J., Wang, W., Wang, H., Yu, P.: δ-clusters: Capturing subspace correlation in a large data set. In: Proceedings of the 18th IEEE International Conference on Data Engineering. (2002) 517–528
20. Yang, J., Wang, W., Wang, H., Yu, P.: Enhanced biclustering on expression data. In: Proceedings of the 3rd IEEE Conference on Bioinformatics and Bioengineering. (2003) 321–327
21. Teixeira, M., Fernandes, A., Mira, N., Becker, J., Sá-Correia, I.: Early transcriptional response of *Saccharomyces cerevisiae* to stress imposed by the herbicide 2, 4-dichlorophenoxyacetic acid. FEMS Yeast Research **6**(2) (2006) 230–48
22. Teixeira, M., Duque, P., Sá-Correia, I.: Environmental genomics: mechanistic insights into toxicity and resistance to the herbicide 2,4-d. Trends in Biotechnology **DOI: 10.1016/j.tibtech.2007.06.002** (2007) in press
23. Simões, T., Teixeira, M., Fernandes, A., Sá-Correia, I.: *Saccharomyces cerevisiae* adaptation to the herbicide 2, 4-dichlorophenoxyacetic acid mediated by Msn2p/Msn4p-regulated genes: role of SPI1. Applied Environmental Microbioly **69** (2003) 4019–4028
24. Teixeira, M., Sá-Correia, I.: *Saccharomyces cerevisiae* resistance to chlorinated phenoxyacetic acid herbicides involves Pdr1p-mediated transcriptional activation of TPO1 and PDR5 Genes. Biochemical and Biophysical Research Communications **292**(2) (2002) 530–537
25. Teixeira, M., Santos, P., Fernandes, A., Sá-Correia, I.: A proteome analysis of the yeast response to the herbicide 2, 4-dichlorophenoxyacetic acid. Proteomics **5**(7) (2005) 1889–901
26. Balzi, E., Goffeau, A.: Yeast multidrug resistance: The PDR network. Journal of Bioenergetics and Biomembranes **27**(1) (1995) 71–76
27. Hikkel, I., Lucau-Danila, A., Delaveau, T., Marc, P., Devaux, F., Jacq, C.: A general strategy to uncover transcription factor properties identifies a new Regulator of drug resistance in yeast. Journal of Biological Chemistry **278**(13) (2003) 11427–11432
28. Owsianik, G., Balzi, E., Ghislain, M.: Control of 26 S proteasome expression by transcription factors regulating multidrug resistance in *Saccharomyces cerevisiae*. Molecular Microbiology **43**(5) (2002) 1295–1308

Generating Data from the Evolution of Artificial Regulatory Networks

Yolanda Sánchez-Dehesa[1], José-María Peña[2], and Guillaume Beslon[1]

[1] Laboratoire d'InfoRmatique en Images et Systémes d'information
UMR 5205 CNRS / INSA de Lyon/Université Claude Bernard Lyon 1 / Université Lumiére Lyon 2 / Ecole Centrale de Lyon, France
[2] DATSI, Universidad Politécnica de Madrid, Spain

Abstract. Existing regulatory network models attempt to copy the "in vivo" regulatory principles by reproducing founded biological results "in silico". These models sometimes don't reflect the biological principal of protein regulation and they don't take into account the organisms evolution. An innovative approach is presented on this contribution, based on the analyze of the existing models. Biological principles of regulatory networks have been considered. In fact, this new model will provide the tools to study regulatory networks emergency and evolution and to acquire knowledge from generated time series. Studying networks in silico provide us the tools for controlling the environment and a better behavior analysis.

Key words: evolutionary algorithm, regulatory networks, protein transcription, network emergency, dynamical properties, artificial data generation

1 Introduction

Genetic regulatory networks are part of many cellular processes and the study of these networks and their evolution over the time can provide us a tool to understand cellular mechanisms. However, these networks cannot be easily studied by "in vitro" tools, due to their kinetic aspect and the impossibility of studying evolution in living systems. So, many authors have estimated the evolutionary mechanism by using bio-informatic tools and models [1,2].

Most of these models treat directly the question of genetic networks evolution, separately from cellular evolution and kinetic. Yet, the Genetic Network is neither directly selected nor directly subjected to mutations: It doesn't evolve by itself. Genetic networks can only evolve as part of an individual and as an inferred system from genome (subjected to mutations).

Functional parts of particular regions of the genome are known to have complex relationships with different biological processes (for example, different pathways). In general, the construction of regulatory networks to represent transcription/translation influence of genes is hard to study on living organisms.

As a required step to deal with the most complex regulatory networks, for example those related to disease evolution or treatment response, it is necessary to understand how these networks arise and evolve. Some authors have worked about genetic networks artificial generation for studied topology [3,4]. They create random networks following some desirable characteristics already known. Unfortunately, generated data are not issued by evolution itself and they are biased to the chosen properties.

In this paper we propose an integrated model of Regulatory Networks to take into account these two aspects in order to study the evolution of Genetic Networks. Our model integrates a genome, who's going to subject mutations, a regulatory network (which evolution will be studied) and an evaluation process based on phenotype characteristics. This model will provide us data to study the genome structure evolution by applying data mining principles.

2 Genetic Regulation

2.1 Principles

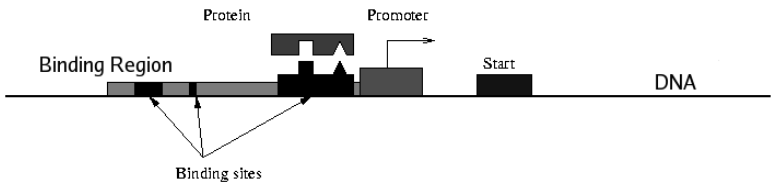


Fig. 1. Example about protein binding over regulatory sites

As it has been formulated by F. Crick in the 1960's, (in the "Central dogma of molecular biology") the DNA decoding is described as a two steps process: transcription (from DNA to RNA) and translation (from RNA to amino-acid sequence). However this process has quickly been extraordinarily complexified by the discovery of the regulatory principles (Jacob, Monod): some proteins can bind on specific DNA locations (Fig 1), interact with them and with the transcription complex, and finally modify the genes transcription close to the binding region.

The regulation activity of the transcription factor also depends on the DNA sequence on the binding sites. Not all the proteins have the same binding site affinity and the transcription level of genes depends on this. The more affinity between transcription factor and binding site, the more gene transcription corresponding to the gene binded site, in the case of enhancer proteins, and in the case of inhibitor proteins,the more affinity the less transcription. Thus, the overall transcription process can be represented as a network (Fig 2. Each

node represents a protein and each arc represents the influence of a protein over the transcription of another one. The weight in the arcs define the regulatory level (enhancing or inhibition) between a given protein and the regulated one. Gene networks are conceptual models of genetic regulation where each gene is considered to be directly affected by a number of other genes [4].

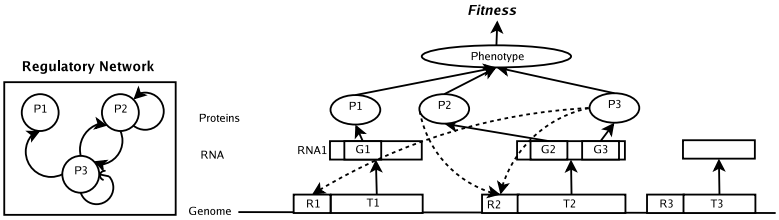


Fig. 2. Regulation in Artificial Evolution. Protein P could be binded in a binding region R, to regulate the transcription of genes G in the T-region

The dynamic of gene network is supposed to explain most of the cellular behaviors as cell differentiation, cell response to stress or cell proliferation in cancer. Regulation activity is the core of cell behavior and it could become a key element in many research lines, either in specific cells/organisms or in more general way (e.g. by studying small world properties of the network structure [5])

Regulatory networks, like many other biological networks, are not randomly connected: they follow a power-law distribution [5]. These networks show clustering and modularity phenomena [6] but don't show random distribution. In these networks sometimes motifs emergency have been detected [7,8]. This connectivity is probably fundamental for the cell activity. Thus, the understanding of both causes and consequences of these specific wiring schemata, is one of the main questions of contemporary biology.

2.2 Evolution of Genetic Networks

As Dobzhansky [9] said 30 years ago: "Nothing in biology makes sense except in the light of evolution". Following this idea, we wonder how networks have evolved to their current topology i.e. what is the evolutionary origin of network structure. It is known that the wiring process has a strong influence on the final network structure and dynamic (See for example preferential attachment from [5]), but the formation process of genetic networks remains mainly unknown.

There exist multiple open issues in the literature of this domain: what's the origin of regulatory networks? Why regulatory networks appear during evolution? How networks evolve over time? Studying the inclusion of new nodes in

already existing regulatory networks, and studying the development of new regulatory networks could help us to answer some of these questions thus providing us a better understanding on their evolution.

In some specific conditions the modularity phenomenon can appear in evolved regulatory networks [6,10]. The main question is what are the conditions to make networks evolve? We can also ask what's the origin of these patterns, or if they play a role during evolution.

However, for studying network evolution and emergency, we cannot tests our hypothesis on living systems. The solution is to develop a biological model "in silico" that will provide us tools to infer knowledge from artificial network evolution, and to explain how networks evolve.

2.3 Models of Regulatory Networks

There is an impressive amount of literature on the question of regulatory networks modeling. However most of this activity relates to networks kinetic or the inference of genetic regulatory networks from transcriptomic analysis of organisms. As long as evolution of regulatory networks is concerned, authors have mainly focus their works on the question of topology evolution [7,8,6], robustness [11,12,13] and evolution of artificial functions [14,15,16,17]. Nevertheless we claim that most of these models are not realistic enough because they consider direct evolution of the genetic network (i.e. direct encoding of the network) or direct selection on the basis of the network property.

Other authors have focused their work on the study of network evolution from the kinetic point of view [18,10,19,14]. They obtain mathematical equations from analytical models which defines the kinetic behavior of regulatory network, after that an analysis of network evolution has been performed: convergence to steady states, oscillations, ...

When we want to study evolution in biological systems "in silico", we use models which gather the main basis of evolution: mutations, genotype-phenotype mapping, and selection. Moreover if the goal is to study evolution of regulatory networks, we should take into account that mutations don't affect directly the network obviously, mutation occurs on the genome, thus modifying indirectly the genome network. As a consequence the genetic network is not submitted to a direct selection pressure (e.g. for connectivity of activity properties). The network *contributes* to the cell phenotype. So it is indirectly selected if it produces sufficient effect on phenotype. While many evolutionary studies directly select the individuals for its topological (e.g. small world or scale free) or kinetic properties (e.g. stability or robustness) it is not clear whether such properties have a real impact on the phenotype. Maybe, they are not selected at all.

If we want to study evolution in regulatory networks, we need to create an integrated model that will include evolution of genome-phenotype mapping and that it will provide us tools to study and analyze the evolution of a genetic networks inside such a system. This biological model should also collects the main biological bases about regulation. It should be stratified on different organization levels from genome to organism's phenotype and it should be compliant with

some minimal characteristics of real regulatory networks: different activation levels, auto-regulation, different degrees of protein production, different binding levels, or genotype-phenotype differentiation.

3 Regulation in Artificial Evolution, the Raevol Model

In order to build such a model we use an integrative evolutionary model previously defined in our team: the Aevol Model [20,21,22].

This model is well suited for our study because it integrates a genotype-phenotype mapping, different codification between genes and proteins, and production levels depending the promoter affinity. It integrates all these features and managing different individual's evolution in a population, there is an evolutionary process: a population with a set of artificial organisms subjecting to a selection process. This process handles individuals in population to achieve a set of metabolic processes.

Aevol has been initially developed to study robustness and evolvability in artificial organisms. However for the sake of simplicity it doesn't include a regulation level.

In Aevol model, the artificial genome is made of a variable number of genes separated by non-coding sequences. The genome is organized as a circular double-strand binary string. Mutations can then change the genome structure. Protein production levels are not predefined, and they are non-dependent on the gene position in the sequence. Even initial protein concentration is determined by the basal level. This level is calculated from the matching level between protein promoter and a reference promoter pattern. The phenotype is the result from the interactions of basic functional elements encoded by genes.

For studying evolution of regulatory networks a regulatory system has been included to Aevol, developing in this way the Regulatory-Aevol Model (Raevol).

In Raevol model, protein's production level is calculated from the matching level between protein promoter and a promoter pattern. The main actor in our model is the protein. It can behave in two different roles: after transcription the genetic code will determine its regulatory capacity, being able to modify the protein's production level, whereas after translation the functional code will define the protein function in metabolic processes.The phenotype is the result from the interactions of basic functional elements encoded by genes (Fig. 3).

3.1 From Genome to Phenotype

The first step is the genome decoding. The genome sequence is parsed to find out promoter position. Once we have localized the promoter regions and the Start protein codon, we extract the binary code of protein. This binary code is translated into a protein sequence by using a functional code. This protein sequence has two tasks: first of all being translated into a protein function, contributing to the phenotype, secondly, compute the regulatory activity of the protein (see Fig 3).

Each protein can be considered as a fuzzy function, represented by a triangle described by three parameters: the mean m, the width w and the maximal height H. From gene sequence and using the genetic code, we can determine a Gray binary value associated to the different parameters of this fuzzy logic. Protein achieve a subset of metabolic processes [21].

When we have translated all proteins in the genome, we have to calculate the interactions between proteins and binding regions to obtain the regulatory graph and calculate the protein production rate (Section 3.2).

Organisms must reach an objective function thanks to an evolutionary algorithm to simulate organisms evolution, so we have to define a phenotype and a fitness function. The **phenotype** is computed as a fuzzy set of processes [21]. If we represent the set of activation functions as A_i and the set of inhibition functions (negative h) as I_j the phenotype can be represented as $P = (\cup_i A_i) \cap (\cup_j I_j)$. For this purpose we have used the Lukasiewicz functions, $W(A_1(x), A_2(x)) = \max(0, A_1(x)+A_2(x)-1)$ for the junction and $W'(A_1(x), A_2(x)) = \min(1, A_1(x)+A_2(x))$ for the intersection.$A_i(x)$ represents the concentration of protein i in the last time step (See Section 3.2)

Fitness: the goal is to compute the individual adaptation to the environment. The objective function is also represented as a set of fuzzy functions. This set can be seen as a set of process E(x) [21]. We measure the gap g between the objective function and the individual phenotype, more gap is found less offspring the organism will have.

3.2 Regulation in genome transcription

To obtain the protein production rate, we should calculate the regulatory capacity of each protein in the individual. By comparing the functional sequence and the binding site (using a regulatory code) we can obtain the protein influence over the regulated protein production (See Fig. 3).

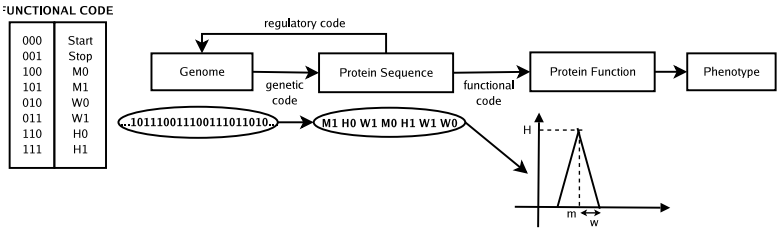


Fig. 3. General schema about steps in DNA-Transcription-Translation and protein behavior levels

To know exactly how a protein influences the other proteins production, we compare the protein sequence with the DNA regulatory region associated with each promoter. Comparing the functional sequence, and the binding site by using a comparison table [3] we can compute the protein affinity with the binding site, thus extracting its influence over the regulated protein concentration.

From each comparison we obtain a value who means how this base can potentially influence the production of regulated protein. Once obtained all the comparison results, mean is calculated. The obtained value enables us to compute how the regulated protein is going to be enhanced or inhibited.

Once we have obtained all binding values, we can build the regulatory graph, and simulate the network behavior. We compute the network concentrations in the cell during twenty time steps. To determine the protein concentration in every time step we use the same equations as in [15]. They provide us a way to reflect the different enhancing/inhibition levels. In these equations protein production is determined by the influence of all proteins over it.

$e_i = \frac{1}{N} \sum_j c_j e^{\beta(u_j^+ - u_{max}^+)}$

$in_i = \frac{1}{N} \sum_j c_j e^{\beta(u_j^- - u_{max}^-)}$

where $i = 1 \ldots n$, is the protein number and N the maximum number of proteins. u_{max}^+ and u_{max}^- are the maximum match achievable between protein and binding region, for enhancer and inhibitor proteins respectively. u_j represents the matching value (or binding value) between protein j and the binding region of protein i.

Giving this equations, the produced protein i follows the equation:

$\frac{\partial c_i}{\partial t} = basal\_level(1 + (e_i - in_i) \cdot c_i) - \phi$

Where ϕ is the protein degradation factor.

The initial concentration of each protein is given by its basal level. It corresponds to the matching level between the promoter of the protein and the promoter matching model. For more details see [21].

To calculate individual's phenotype fitness, we take the protein concentration levels for each individual to obtain the metabolic functions achieved by the individual (organism phenotype) and we compare them with the objective function. With comparison value we will be able to select the best individuals in population (See Section 3.1).

3.3 Evolutionary Algorithm

As in classical genetic algorithms population evolves by individual mutation. Individual fitness is computed and individuals are selected for reproduction using sampling with replacement. When new individuals are created by parents crossover, they replace parents in population.

Achieved mutations can affect only a few bits (local mutations) or a huge genomic segment. Three types of local mutations can be applied: switching a

[3] this table gather the protein codon-DNA base affinity. It has been created following a random normal function and filling half cases with null values

position (from 0 to 1 or vice-versa) , inserting a few bases (from 1 to 6) in a position, and deleting from 1 to 6 bases. Mutations applied over huge genomic segments can be deletion, duplication , translocation or inversion of long strings [21]. These mutations affect only genome, so they will modify indirectly the protein network topology.

4 Proposed Scenario

To study network emergence, we have simulated the bacterial behavior in a steady state. Any external protein or influence is going to modify the bacterial metabolic functions. Organisms must reach a set of biological processes(objective function).

We have simulated the evolution in a population of 1000 individuals with a mutation rate of $1e-5$ base/genome, different seeds, and the same parameters used in [14]. These parameters have been chosen to obtain protein concentration stabilization in the last time steps in bacterial life.

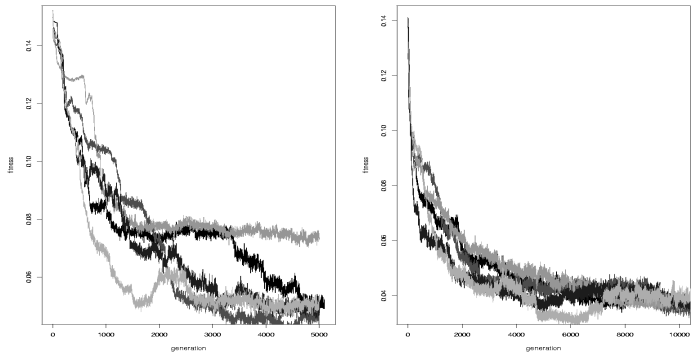


Fig. 4. Left: mean fitness curve of all individuals in the population over 5000 generations. Right: mean fitness curve of all individuals in the population over 10000 generations, following the external protein influence

After 5000 generation we can see in Figure 4(left) how individuals in different simulations reach the objective function. In the Raevol model, the lower the fitness, the more adapted the individuals. Here we see that the fitness value is continuously decreasing, showing that individuals improves their achieving task.

Obtained networks are highly and completely connected (See in Fig. 5, aver. active links) and the network size increases as number of genes does (See in Fig. 5,average number of genes per genome). However, the links value's histogram

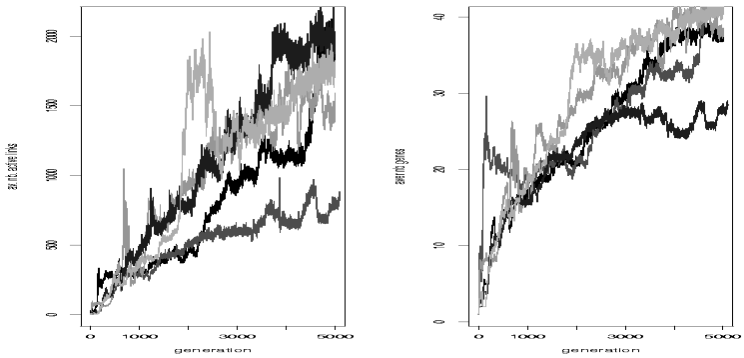


Fig. 5. On the left, average number of active links during generations. On the right, average number of genes per genome. 5 simulations with 5 different seeds during 5000 generations.

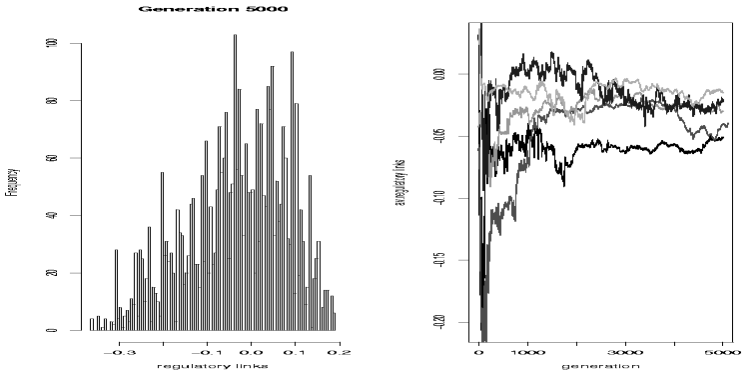


Fig. 6. On the left: Histogram about links values in regulatory network for the best individual at generation 5000. On the right: Evolution of average links values in regulatory network during generations for five simulations with five different seeds.

(Fig 6) shows that most of these links are close to zero and can thus be considered as inactive links in the network. This is probably due to the stable fitness function we used in this experiment: since the organisms only need to achieve a unique metabolic function, they evolve a weakly connected network.

In these simulations, individuals have to reach a steady function. They develop mainly negative links (See histogram Fig 6) and they converge to a steady

negative value (about -0.025) near to zero (See graphic about mean links in Fig 6).

We have tested the algorithm in a new context. The individual must achieve a set of biological processes, but when an external protein arrives to the nucleus, the organisms must change its behavior and stop achieving a subset of biological processes. This external protein should have the inhibitor role in the regulatory network created by the individual. In this case, individuals must adapt themselves not only to develop the first set of processes but they must also adapt their binding sites to the new protein to inhibit the necessary biological functions. More precisely, first of all individuals must reach a set of metabolic functions, during its life period, an external protein arrive to the cells, modifying the objective function.The new external protein should have a repression activity. For this scenario, we have simulated during 10000 generations to wait for stabilization, maintaining the mutation rate and individuals population.

As we can see in Fig. 4 the algorithm reach the objective functions decreasing it fitness curve. The approximation is as good as the first scenario, when any external protein influence the cell behavior.

For this scenario, obtained networks are also highly and completely connected (See in Fig. 7, aver. active links) and the network size increases as number of genes does (See in Fig. 7,average number of genes per genome). Contrary to the first experiment, we can see how genome size and number of links in the graph tends to stabilize to a reasonable size after some increasing generations (Fig.7). In first generations, algorithm search the best solution in the fitness landscape, exploring all possibilities by adding genes to the genome. Selection will erase the non adapted features to reach the objective function.

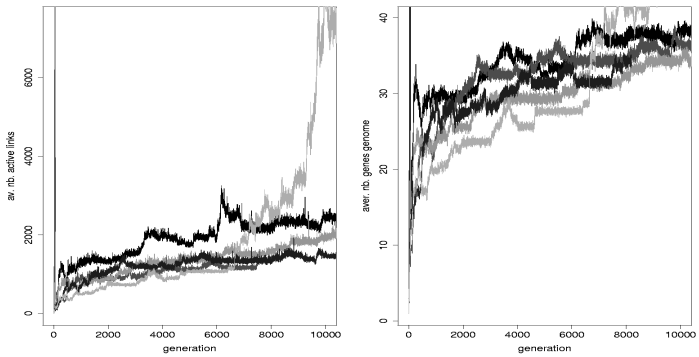


Fig. 7. On the left, average number of active links during generations. On the right, average number of genes per genome. 5 simulations with 5 different seeds during 5000 generations.

Individuals should adapt their regulatory network to inhibit the set of process that differs from the first set of metabolic functions. For this purpose, individuals regulatory network must contain negative links who should inhibit the subset of metabolic processes not covered by the second objective function. In Fig. 8 we can see observe the regulatory values distribution. Most of values are between [-0.1,0.1], these leak values cannot have a big influence in network, but as we can see in histogram, a set of inhibitory links appear (between [-0.3,-0.1]). This means that individuals have been able to adapt themselves to the new objective function, inhibiting the set of metabolic processes that differs from the first function.

Looking at evolution of mean links values, we see it tends to zero (Fig. 8). Taking into account above conclusions, it means that enhancer proteins will bind the inhibitor protein sites, activating in this way the repression. This could explain the big number of positive links (activators) appeared in the histogram (contrary to expected).

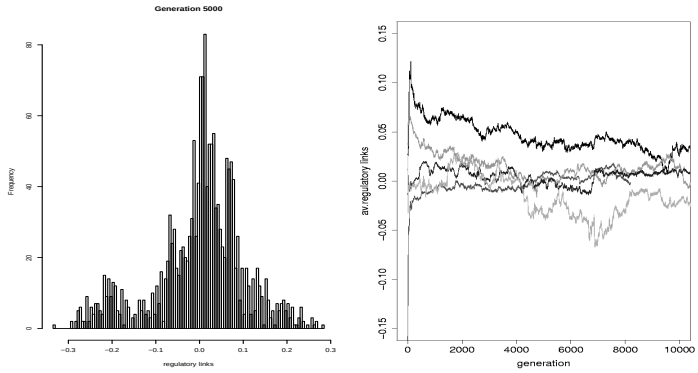


Fig. 8. On the left: Histogram about links values in regulatory network for the best individual at generation 5000. On the right: Evolution of average links values in regulatory network during generations for five simulations with five different seeds, when external protein influences behavior.

5 Open issues and Future Works

Although preliminary, these results show that evolution of regulatory networks is not fully random: some emerged features are repetitive over the different simulations. We can also conclude that our model works and that he's able to adapt himself and it regulatory network to functions requested. These are a very inter-

esting result because networks are not directly selected and it opens a big study field.

Data from real laboratory experiments are biased by noise and by the fact that phenomena observation can change the observed phenomena. In silico experiments are a good solution for easy data generation, noise management and controlled experiment environment.

The model will provide data about the genome structure during different steps in evolution. Studying these data, we could localize the emerged patterns, and how they evolve: together, deviating and later derivation after duplication, ... [2]

In this model we have realized that the mechanisms used in first stages of evolution is duplication. We would like to know if for this exploratory phase, promoters and genes evolve together and if they are duplicated together. Only comparing the different binary genome structures during time, we could be able to answer to these questions.

Studying the genome's structure applying data mining principles can answer the questions about genome structure emergency and evolution. We should also find a system to study more precisely the topology of emerged networks.

Although "in silico" evolution provides controlled environments, it requires a simplified version of the actual biological model. We should be aware that a mimicry for biological models should be realistic enough to make extracted results worthy. Previous results were based on very simple models and then limited in the scope of their results. Our proposal would look further and it dare also propose more ambitious goals.

Further experiences will test the influence of mutation rate over the network emergency and evolution. They should also modify the network topology. A mathematical study is foreseen. We'll analyze if mathematical equations that define the concentrations levels, are the most adapted for our purpose.

References

1. Teichmann, S.A., Babu, M.M.: Gene regulatory network growth by duplication. Nat Genet **36**(5) (2004) 492–496
2. Madan Babu, M., Teichmann, S.A.: Evolution of transcription factors and the gene regulatory network in escherichia coli. Nucleic Acids Res **31**(4) (2003) 1234–1244
3. Medina, A., Lakhina, A., Matta, I., Byers, J.: BRITE: Universal topology generation from a user's perspective. Technical Report 2001-003 (2001)
4. Mendes, P., Sha, W., Ye, K.: Artificial gene networks for objective comparison of analysis algorithms. Bioinformatics **19 Suppl 2** (2003)
5. Barabasi, A., Oltvai, Z.N.: Network biology: Understanding the cell's functional organization. Nat. Rev. Genet. **5** (2004) 101–113
6. Kashtan, N., Alon, U.: Spontaneous evolution of modularity and network motifs. PNAS **102**(39) (2005) 13773–13778
7. Kashtan, N., Itzskovitx, S., Milo, R., Alon, U.: Efficient sampling algorithm for estimating subgraph concentrations and detecting network motifs. Oxford Bioinformatics **20**(11) (2004) 1746–1758

8. Yeger-Lotem, E., Sattath, S., Kashtan, N., Itzkovitz, S., Milo, R., Pinter, R.Y., Alon, U., Margalit, H.: Network motifs in integrated cellular networks of transcription-regulation and protein-protein interaction. PNAS **101**(16) (2004) 5934–5939
9. Dobzhansky, T.: Nothing in biology makes sense except in the light of evolution. The American Biology Teacher (35) (1973) 125–129
10. Hallinan, J., Willes, J.: Evolving genetic regulatory networks using an artificial genome. 2nd Asia-Pacific Bioinformatics Conference **29** (2004)
11. Ciliberti, S., O.C.Martin, Wagner, A.: Robustness can evolve gradually in complex regulatory gene networks with varying topology. PLoS Computational Biology **3**(2) (2007) e15
12. Aldana, M., Balleza, E., Kauffman, S., Resendiz, O.: Robustness and evolvability in genetic regulatory networks. J Theor Biol (2006)
13. Soyer, O.S.S., S.Bonhoeffer: Evolution of complexity in signaling pathways. Proc Natl Acad Sci U S A (2006)
14. Kuo, P.D., Leier, A., Banzhaf, W.: Evolving dynamics in an artificial regulatory network model. In Yao, X., Burke, E.K., Lozano, J.A., Smith, J., Guervós, J.J.M., Bullinaria, J.A., Rowe, J.E., Tiño, P., Kabán, A., Schwefel, H.P., eds.: Parallel Problem Solving from Nature - PPSN VIII, 8th Int. Conf., Birmingham, UK, Sept. 18-22, 2004, Proceedings. Volume 3242 of LNCS., Springer (2004) 571–580
15. Banzhaf, W.: On evolutionary design, embodiment and artificial regulatory networks. Embodied Artificial Intelligence (2004) 284–292
16. Banzhaf, W.: Artificial regulatory networks and genetic programming. In Riolo, R.L., Worzel, B., eds.: Genetic Programming Theory and Practice. Kluwer (2003) 43–62
17. P.Francois, Hakim, V.: Design of genetic networks with specified functions by evolution in silico. PNAS. USA **101**(2) (2004) 580–585
18. de Jong, H., Page, M., Hernandez, C., Geiselmann, J.: Qualitative simulation of genetic regulatory networks: Method and application. In: IJCAI. (2001) 67–73
19. Krul, T., Kaandorp, J.A., Blom, J.G.: (Modelling developmental regulatory networks)
20. Knibbe, C.: Structuration des génomes par sĺection indirecte de la variabilité mutationnelle, une approche de modlisation et de simulation. PhD thesis, INSA - Lyon (2006)
21. Knibbe, C., Beslon, G., Lefort, V., Chaudier, F., Fayard, J.M.: Self-adaptation of genome size in artificial organisms. In: Proc. of the 8th European Conference, ECAL 2005. Volume 3630., Springer (2005) 423–432
22. Knibbe, C., Mazet, O., Chaudier, F., Fayard, J.M., Beslon, G.: Evolutionary coupling between the deleteriousness of gene mutations and the amount of non-coding sequences. JTB **244**(4) (2007) 621–630

Transcription Factor Binding Site Discovery by the Probabilistic Rules

Irina Khomicheva[1], Alexander Demin[2], and Evgeny Vityaev[3]

[1] Institute of Cytology and Genetics SB RAS, Lavrentyev aven., 10
630090 Novosibirsk, Russia
[2] A.P.Yershov Institute of Information Systems SD RAS, Lavrentyev aven., 6,
630090 Novosibirsk, Russia
[3] Sobolev Institute of Mathematics, Koptyug aven. 4,
630090 Novosibirsk, Russia
{khomicheva@bionet.nsc.ru, alexandredemin@yandex.ru, vityaev@math.nsc.ru}

Abstract. Control of gene expression at the level of transcription is achieved by nuclear factors that bind to regulatory elements, short DNA sequence motifs, called transcription factor binding sites. The development of reliable methods for binding site recognition is an important step in large-scale genome analysis. The Data Mining approaches adapted to bioinformatics tasks show high efficiency. Yet the specificity of the regulatory region analysis task consists in the high false-positive rates. In this paper the program system 'Discovery' was applied to tasks of binding site recognition. 'Discovery' makes a semantic probabilistic inference and finds the statistically significant probabilistic rules. The hypothesis class is defined by the expert in dialog mode. In this paper we demonstrate that 'Discovery' is consistently more accurate than the traditional weight matrices in binding site prediction task, as was established for three families of transcription factors.

Keywords: semantic probabilistic inference, probabilistic rule, transcription factor binding sites, prediction.

1 Introduction

Analysis of gene transcription regulatory regions is of great importance for understanding molecular mechanisms of transcription.

The task of transcription factor binding sites (TFBS) prediction is methodologically difficult due to a high variety of DNA binding proteins and the degeneracy of TFBSs conferred on them by the tissue- and stage-specific regulatory mechanisms. These sequences vary in length, position, redundancy, orientation in the DNA chain, and bases. As a direct consequence, the problem of a large number of false-positives necessarily takes place manifesting itself in the poor predictive performance of the corresponding software.

The problem of regulatory region analysis challenges the Data Mining and Machine Learning approaches. Machine Learning algorithms intended for addressing bioinformatics tasks are: hidden markov models, decision trees, neural network, genetic algorithms, etc. [1].

The traditional approach to predict TFBSs is the positional weight matrix, PWM, a very powerful tool, but still with some drawbacks and limitations [2]. PWM and consensus-based methods involve an explicit assumption that the contribution of each nucleotide position to the binding affinity is independent and the effect produced on the binding strength is cumulative. In PWM, elements simply correspond to the probabilities of observing each nucleotide at each position. Numerous works [3, 4, 5, 6] indicate that the nucleotides of TFBSs cannot be treated independently. This assumption is invalid and contradicts the processes underlying the biological model. Predictions can be further improved by taking into account the sequence context in which a predicted site is located.

Despite an evident importance of noncoding sequences to gene regulation, our ability to describe and properly localize them is extremely limited. The known approaches are rather restricted as they are confronted with the lack of sufficient training data and the degeneracy of the biological objects under analysis.

In this paper we applied 'Discovery' system to knowledge acquisition tasks on DNA sequences. Unlike PWM and consensus methods, 'Discovery' reveals mutual interdependences among the nucleotides which, in the general case, are rather distant from one another.

In this paper we demonstrate that 'Discovery' is consistently more accurate than the traditional weight matrices in TFBS prediction tasks, as was established for three families of binding factors.

2 'Discovery' as Implemented for Bioinformatics

As the training data for 'Discovery' system we used the samples of nucleotide sequences, putative TFBSs, that were organized in the data table. Each table row contained the binding site name and its nucleotide sequence. For example,

>S1916	gtccgtgggt
>S4809	ttgggggcga
>S6067	gaggggcgg
>S6069	gcgggggcgg
>S5824	acggaggcgg

The 'Discovery' system reveals the probabilistic rules of the form:

$$(\boldsymbol{Pos}_1 = \boldsymbol{C}_1)^{\varepsilon 1} \,\&\, (\boldsymbol{Pos}_2 = \boldsymbol{C}_2)^{\varepsilon 2} \,\&\, \ldots \,\&\, (\boldsymbol{Pos}_k = \boldsymbol{C}_k)^{\varepsilon k} \rightarrow (\boldsymbol{Class} = \boldsymbol{1}) \,. \qquad (1)$$

where $(\boldsymbol{Pos}_i = \boldsymbol{C})^{\boldsymbol{\varepsilon}}$ – is a predicate, denoting that the sequence position $\boldsymbol{i}$ contains (if $\boldsymbol{\varepsilon} = \boldsymbol{1}$) or doesn't contain (if $\boldsymbol{\varepsilon} = \boldsymbol{0}$) the symbol $C \in \{a, t, g, c\}$; $(\boldsymbol{Class} = \boldsymbol{1})$ – target predicate, which implies that the nucleotide sequence is one of the binding sites of a particular transcription factor.

In general, 'Discovery' system makes a semantic probabilistic inference [7, 8], which allows the user to find all statistically significant rules $\boldsymbol{P_1} \& ... \& \boldsymbol{P_m} \to \boldsymbol{P_0}$ that predict the predicate $\boldsymbol{P_0}$ with the highest probability.

The semantic probabilistic inference is based on the definition of the probabilistic rule, which is as follows. The rule $\boldsymbol{P_1} \& ... \& \boldsymbol{P_m} \to \boldsymbol{P_0}$ is a probabilistic rule, if for any rule $\boldsymbol{P_{i1}} \& ... \& \boldsymbol{P_{ik}} \to \boldsymbol{P_0}$ such that $\{\boldsymbol{P_{i1}}, ..., \boldsymbol{P_{ik}}\} \subset \{\boldsymbol{P_1}, ..., \boldsymbol{P_m}\}$ the conditional probability satisfies the inequality $\boldsymbol{p}(\boldsymbol{P_0} \mid \boldsymbol{P_{i1}} \& ... \& \boldsymbol{P_{ik}}) < \boldsymbol{p}(\boldsymbol{P_0} \mid \boldsymbol{P_1} \& ... \& \boldsymbol{P_m})$.

Let us introduce the *correction* to the rule. The rule $\boldsymbol{P_1} \& ... \& \boldsymbol{P_m} \& \boldsymbol{P_{m+1}} \to \boldsymbol{P_0}$ is a *correction* to the rule $\boldsymbol{P_1} \& ... \& \boldsymbol{P_m} \to \boldsymbol{P_0}$, if the former rule was created by adding an any predicate $\boldsymbol{P_{m+1}}$ to the statement of the rule $\boldsymbol{P_1} \& ... \& \boldsymbol{P_m} \to \boldsymbol{P_0}$.

Consider the algorithm of making semantic probabilistic inference.

The algorithm provides a successive introduction of corrections to the rules and a consistent check for conformity to the criterion of being probabilistic rule. Checking all possible rules is a difficult, time-consuming computational task, and for that reason semantic probabilistic inference in practice involves two successive checks: the *basic check* and the *advanced check.*

The algorithm of making semantic probabilistic inference.

1. At the first step, a set of probabilistic rules $\boldsymbol{REG_1}$ is generated by the exhaustive search of all the rules $\boldsymbol{P_1} \& ... \& \boldsymbol{P_m} \to \boldsymbol{P_0}$ for $\boldsymbol{1} \le \boldsymbol{m} \le \boldsymbol{d}$ and checking for conformity to the criterion of being probabilistic. That is, $\boldsymbol{REG_1} = \{\boldsymbol{R_i}\}$, where $\boldsymbol{i} \in \boldsymbol{I_1}$, $\boldsymbol{R_i} = \boldsymbol{P_1} \& ... \& \boldsymbol{P_m} \to \boldsymbol{P_0}$, $\boldsymbol{1} \le \boldsymbol{m} \le \boldsymbol{d}$, $\boldsymbol{R_i}$ is the probabilistic rule. This step is referred to as a *basic check*, and the value *d* is referred to as the *depth of the basic check.*
2. At the *k*-th ($k > 1$) step, a set of probabilistic rules $\boldsymbol{REG_k}$ is generated by correcting all the rules that were found at the previous step and checking for conformity to the criterion of being probabilistic rule. That is, $\boldsymbol{REG_k} = \{\boldsymbol{R_i}\}$, $\boldsymbol{i} \in \boldsymbol{I_k}$, $\boldsymbol{R_i} = \boldsymbol{P_1} \& ... \& \boldsymbol{P_m} \to \boldsymbol{P_0}$, $\boldsymbol{m} = \boldsymbol{d} + (\boldsymbol{k} - \boldsymbol{1})$, $\boldsymbol{R_i}$ is a probabilistic rule, $\boldsymbol{R_i} \in \boldsymbol{Spec}(\boldsymbol{REG_{k-1}})$, where $\boldsymbol{Spec}(\boldsymbol{REG_{k-1}})$ is the set of all corrections to the rules in $\boldsymbol{REG_{k-1}}$.
3. The algorithm stops when no further correction to any rule is possible. The ultimate set of all rules $\boldsymbol{REG}$ is equal to the union of all $\boldsymbol{REG_k}$: $\boldsymbol{REG} = \bigcup_{\boldsymbol{k}} \boldsymbol{REG_k}$.

The depth of the basic check, that is, the maximum length of the rules that will be subject to the basic check, is specified by the user. In practice, the depth of the basic check is normally equal to 2 or 3.

As the algorithm proceeds, the conditional probability of the rules is assessed using a training set. To prevent the rules that fail to achieve significance, additional criteria are normally applied to assess statistical significance among the rules. We used the exact Fisher criterion applied to contingency tables [9]. The rules that fail to meet criteria are to be discarded even if they prove highly accurate on the training set.

The calculation of object score is critical for decision making if the nucleotide sequence is one of the binding sites of a particular transcription factor. 'Discovery' supports several procedures for calculating the score. For TFBS recognition purposes, we used the following procedure:

$$\boldsymbol{score} = \frac{\sum_{\boldsymbol{R}\in\boldsymbol{TR}} \boldsymbol{p}(\boldsymbol{R})}{\sum_{\boldsymbol{R}\in\boldsymbol{AR}} \boldsymbol{p}(\boldsymbol{R})}. \quad (2)$$

where $\boldsymbol{p(R)}$ – conditional probability of the rule $\boldsymbol{R}$, $\boldsymbol{AR}$ – all the rules discovered by the system, $\boldsymbol{TR}$ – all the rules that are applicable to the object.

Further the sequence score is compared to the threshold $\boldsymbol{\delta} \in [\boldsymbol{0},\boldsymbol{1}]$. If the object score is higher than the threshold, then the object is one of the TFBSs. We defined the false positive (FP) rates based on false negative (FN) rate using the standard jackknife procedure.

3 Experimental Data Analysis

We analyzed the DNA targets of three protein families: sterol regulatory element binding protein (SREBP), early growth response factor 1 (EGR1), and Hepatocyte nuclear factor 4 (HNF4). The training data sets (sequences of TFBSs with flanks) were retrieved from the TRRD database [10].

We performed the accuracy comparison of 'Discovery' and PWM according to the standard jackknife procedure [11]. Totally, the data sets contained 38 sequences of SREBP binding sites, 22 (EGR1), 30 (HNF4) (Table 1). First of all, we tried PWM on different sequence lengths to reach the highest PWM recognition accuracy. When the optimal sequence lengths for PWM were found to be 18 nucleotides for SREBP data, 10 for EGR1 and 13 for HNF4, we prepared positive training sets containing sequences of binding sites 18, 10, and 13 nucleotides in length. The negative training set consisted of randomly generated sequences with the same frequencies as in the positive set.

Table 1. Samples of nucleotide sequences, recognition accuracy of the 'Discovery' system and PWM for TFBS SREBP, EGR1 and HNF4. False positive rates at the stringent threshold are defined by the false negative rate equal to 50%.

TFBS	Power of TFBSs Set	Sequence length, nt	FP rate PWM	FP rate 'Discovery'

SREBP	38	18	4.70E-04	3.90E-04
EGR1	22	10	4.06E-03	2.39E-03
HNF4	30	13	2.14E-04	7.00E-05

During each jackknife iteration methods were trained on the data set of sequences leaving exactly one for the control. The trained methods were applied to the rest sequence, estimating the FP error; the control negative sample was randomly generated with the nucleotide frequencies as in the positive samples and contained 100 000 sequences. Totally the number of jackknife iterations in each case was equal to the number of sequences in the data sets. We arranged the control sites according to the FP rates. Figure 1 depicts the correlations between the true positive (TP) and FP rates for the HNF4 binding sites data. 'Discovery' outperforms PWM at any error cutoff.

The obtained result for the rest families of transcription factors (EGR1 and SREBP) is analogous to the HNF4, 'Discovery' favorably competes with the PWM. We produce the FP rates for both algorithms at the stringent threshold defined by the FN rate equal to 50% (table 1).

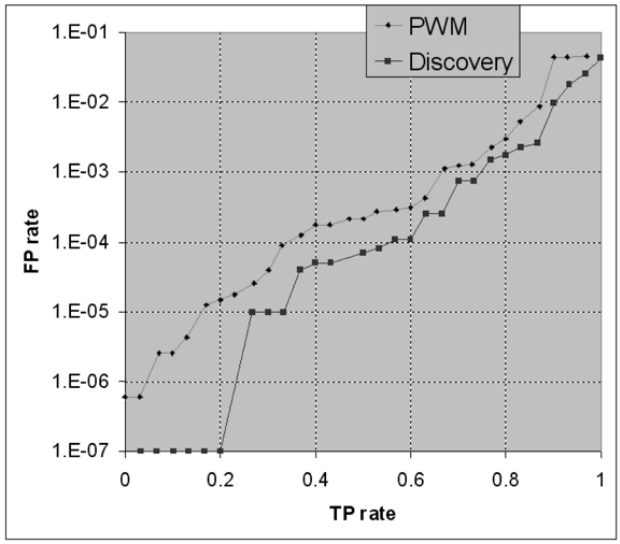


Fig. 1. Recognition performance of 'Discovery' system and PWM for HNF4 binding sites. The best six HNF4 binding sites are recognized at the FP rate lower then 1.E-07.

Acknowledgments. Siberian Branch of the Russian Academy of Sciences (integration project no. 115). The work was in part supported by the President of the Russian Federation, Scientific Schools grant 4413.2006.1.

References

1. Tan, A.C., Gilbert, D.: An empirical comparison of supervised machine learning techniques in bioinformatics. Proceedings of First Asia Pacific Bioinformatics Conference (APBC) (2003)
2. Stormo, G.D.: DNA binding sites: representation and discovery. Bioinformatics, 16, 16--23 (2000)
3. Benos, P.V., Bulyk, M.L., Stormo, G.D.: Additivity in protein-DNA interactions: how good an approximation is it? Nucleic Acids Res, 30, pp. 4442--4451 (2002)
4. Man, T.K., Stormo, G.D.: Non-independence of Mnt repressoroperator interaction determined by a new quantitative multiple fluorescence relative affinity (QuMFRA) assay, Nucleic Acids Res., 29, pp. 2471--2478 (2001)
5. Barash, Y., Elidan, G., Friedman, F., Kaplan, T.: Modeling dependencies in protein-DNA binding sites. RECOMB, pp. 28--37 (2003)
6. Udalova, I.A., Mott, R., Field, D., Kwiatkowski, D.: Quantitative prediction of NF-kB DNA–protein interactions. Proc. Natl. Acad. Sci. USA, 99, pp. 8167--8172 (2002)
7. Vityaev, E., Data mining. Computational intelligence. Cognitive models. Novosibirsk State University, Novosibirsk, p. 293 (2006)
8. Vityaev, E.E.: Semantic approach to knowledge base creating. Semantic probabilistic inference of the best for prediction PROLOG-programs by a probability model of data Logic and Semantic Programming. Computational Systems, 146, Novosibirsk, pp.19--49 (1992)
9. Kendall, M.G., Stuart A.: The advanced theory of statistics, 4th ed., v.3. Charles Griffin & Co LTD, London (1977)
10. Kolchanov, N.A., Ignatieva, E.V., Ananko, E.A., Podkolodnaya, O.A., Stepanenko, I.L., Merkulova, T.I., Pozdnyakov, M.A., Podkolodny, N.L., Naumochkin, A.N., Romashchenko, A.G.: Transcription Regulatory Regions Database, (TRRD): its status in 2002. Nucleic Acid Res., 30, pp. 312--317 (2002)
11. Efron, B., Gong, G.: A leisurely look at the bootstrap the jackknife and resampling. American Statistician, 37, pp. 36--48 (1983)

Author Index