INTEGRATION OF TRANSCRIPTION FACTOR BINDING AND GENE EXPRESSION BY ASSOCIATIVE CLUSTERING

Janne Nikkilä[1,3]*, Christophe Roos*[2]*, and Samuel Kaski*[1,3]

[1]University of Helsinki, Department of Computer Science,
P.O. Box 68, FI-00014 University of Helsinki, Finland, janne.nikkila@hut.fi
[2]Medicel Oy, Huopalahdentie 24, FI-00350 Helsinki, Finland, christophe.roos@helsinki.fi.
[3]Helsinki University of Technology, Neural Networks Research Centre,
P.O. Box 5400, FI-02015 HUT, Finland, samuel.kaski@cs.helsinki.fi.

ABSTRACT

We integrate paired genomic data sets to reveal their dependencies. We suggest using a dependency-maximizing clustering method for the task. The recently introduced method *associative clustering (AC)* finds groupings of genes for which the two data sources are maximally dependent. The dependencies between data sources become represented as a contingency table, which is optimized to reveal the association between data sets, bypassing the possible incommensurability between the data sets. The method is applied to searching for regulatory interactions in yeast, by looking for dependencies between gene expression profiles and regulator binding patterns.

1. INTRODUCTION

Integration of the multiple sources of genomic data is an essential task in modern biology. The possible sources include gene expression, gene sequence, protein expression, and protein interaction data. The data types are heterogeneous, which prevents the most trivial integration methods. Probabilistic models [1] and kernel methods [2] have been applied to the problem earlier.

However, the incommensurability of the information sources may still be a problem, even when the type of the sources is the same. This holds for gene expression measurements made with different array platforms, or in fact for any information sources producing co-occurring multivariate real-valued data. The incommensurability makes for instance the simple concatenation (for joint distribution modeling) of the vector valued data sets suboptimal, since some variables from one source only may dominate all the other variables. More importantly, the dominating sources may hide the associations between the sources, which are the main interest in this work.

We propose to represent dependencies between two data sets by a contingency table formed by a set of clusters for each data sources. The contingency table is cross-tabulation of the clusters of the one source against the other, and the table cells contain the counts of the data items co-occurring in the respective marginal clusters. It can be interpreted as a coarse summary of the dependencies between the two sets of clusters, with an important property: data sets are now commensurable on the level of the contingency table. The problem is now to find such clusters in each data space that reveal maximally well the dependencies between the data sets. This can be solved with a recently developed method called *associative clustering (AC) [3]*.

The problem setting of AC is the following: Assume two data sets with *co-occurring* samples, that is, samples coming in pairs $(\mathbf{x}, \mathbf{y})$ where $\mathbf{x}$ belongs to the first set and $\mathbf{y}$ to the second. In this paper both $\mathbf{x}$ and $\mathbf{y}$ are multivariate real-valued genomic measurements about the same gene, but in principle the type of the data sets need not be the same. The general research problem is to find *common properties* in the set of pairs; statistically speaking, the goal is to find statistical dependencies between the pairs. Additionally, when one is not able or willing to postulate a detailed parametric model *a priori*, dependency modeling with non- or semiparametric methods (such as associative clustering) is a natural way of formalizing the search for commonalities in co-occurring data sets. Another nice property of the AC is that in data mining the search for dependencies between data sets is a considerably better-defined target than the common, unsupervised search for clusters and other regularities.

The standard unsupervised clustering methods, reviewed for gene expression clustering for instance in [4], aim at finding clusters where genes have similar expression profiles. The goal of AC is different: to cluster the $\mathbf{x}$ and the $\mathbf{y}$ separately such that the dependencies between the two clusterings capture as much of the statistical dependencies between two sets of clusters as possible. In this sense the clustering is *associative*; it finds associations between samples of different spaces. The research problem will be formalized in Section 2.

We apply associative clustering to genomic data sets to search for regulatory interactions between gene expression during cell-cycle and transcription factor binding patterns. More generally, we argue that once a research goal can be represented as a search for dependencies between data sets, our approach is a well-defined middle ground between purely hypothesis
-driven research, for which hypotheses must be available,

and purely exploratory research, where the task is often ill-defined.

For microarray data, the existing dependency -searching techniques, like Information Bottleneck methods [5], have two deficiencies. First, mutual information, the dependency measure that they maximize, is defined for probability distributions which in turn need to be estimated from samples. The separate estimation stage with its own optimality criteria will introduce errors to the models. The errors are negligible for asymptotically large data sets but non-negligible for many real-life sets. We will solve the problem by directly defining a dependency measure for data instead of distributions. It is justified by combinatorial and Bayesian arguments. For asymptotically large data sets the dependency measure becomes mutual information, and can therefore be viewed as a principled alternative to mutual information for finite data sets.

The second shortcoming has been that the models are not applicable to symmetric dependency clustering of *continuous* data. While a trivial extension of the existing continuous-data methods may seem sufficient, a conceptual change is actually required. Existing finite-data formulations either maximize the likelihood $p(y|\mathbf{x})$ of one data set, say y, given $\mathbf{x}$, or maximize the symmetric joint likelihood for $p(\mathbf{x}, \mathbf{y})$. Neither of these approaches is dependency modeling: Conditional models are asymmetric, while joint density models represent all variation in $\mathbf{x}$ and $\mathbf{y}$ instead of common variation, and therefore do not even asymptotically reduce to mutual information. A solution implemented in associative clustering is to use a hypothesis comparison approach which translates to a Bayes factor cost function.

From the biological perspective, the advantages of clustering by maximizing dependency between two sources of genomic information are at least two-fold. First, the new problem setting makes it possible to formulate new kinds of hypotheses about the dependency of the sources, not possible with conventional one-source clusterings. Second, mining for regularities in the common properties of two data sets is a more constrained problem that mining for any kinds of regularities within either of them. Hence, assuming the sets are chosen cleverly, the results are potentially better targeted. Our hypothesis is that there will be less false positives in the discovered regulatory interactions when expression and transcription factor binding are combined in a dependency maximizing way, compared to one-source clusterings. The previous studies support this hypothesis, see for example [6]. We will study the interactions in Section 5.

2. ASSOCIATIVE CLUSTERING

Given two feature sets, $\mathbf{x}$ and $\mathbf{y}$, for one set of objects, the associative clustering [3] clusters each feature set separately, in such a way that (i) the clusterings capture as much as possible of the dependencies between pairs of data samples $(\mathbf{x}, \mathbf{y})$, and (ii) the clusters contain similar data points.

More formally, for paired data $\{(\mathbf{x}_k, \mathbf{y}_k)\}$ of real vectors $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{d_x} \times \mathbb{R}^{d_y}$, we search for partitionings $\{V_i^{(x)}\}$ for $\mathbf{x}$ and $\{V_j^{(y)}\}$ for $\mathbf{y}$. The partitions can be interpreted as clusters in the same sense as in K-means; they are Voronoi regions parameterized by their prototype vectors $\mathbf{m}_i$. The $\mathbf{x}$ belongs to $V_i^{(x)}$ if $\|\mathbf{x} - \mathbf{m}_i\| \leq \|\mathbf{x} - \mathbf{m}_k\|$ for all k, and correspondingly for $\mathbf{y}$.

2.1. Measuring and maximizing dependency

If the joint probability distribution p_{ij} of two cluster sets indexed with i and j was given, mutual information could be used to measure the dependency between them. However, if only the co-occurrence frequencies n_{ij} computed from a finite data set are available, mutual information computed from this empirical distribution would be a biased estimate.

Co-occurrences of data in two sets of clusters can be interpreted as a *contingency table*. For finite data in contingency tables, *Bayes factors* have classically been used as dependency measures (see, e.g., [7, 8]). They operate by comparing a model of dependent margins to another model for independent margins, and are asymptotically equivalent to mutual information.

The novelty in AC is that the Bayes factor is *optimized* instead of only being used to measure dependency in a fixed table. Similarly to the classical case, also in AC the Bayes factor compares two alternative models for the cross-tabulation data: one describing a table where the margins are dependent and the other a table with independent margins. However, in AC the clusters in each data space now correspond the categorical variables defining the rows and columns of the contingency table. The clusters, in turn, are defined by the Voronoi regions parameterized by the prototypes vectors. In the optimization, the prototypes are tuned to make the dependent model describe the contingency table data better than the independent model, which can be interpreted as maximizing the dependency between the clusters.

In associative clustering, the frequencies over the cells of a contingency table are assumed to be multinomially distributed. The model M_I of *independent clusters* assumes that the multinomial parameters over cells are outer products of posterior parameters at the margins: $\theta_{ij} = \theta_i \theta_j$. The model M_D of *dependent clusters* ignores the structure of the cells as a two-dimensional table and samples cell-wise frequencies directly from a table-wide multinomial distribution θ_{ij}. Dirichlet priors are set for both the margin and the table-wide multinomials.

AC is optimized by maximizing the following Bayes factor

$$BF = \frac{p(\{n_{ij}\}|M_D)}{p(\{n_{ij}\}|M_I)} \tag{1}$$

with respect to the margin clusters parameterized by prototypes $\mathbf{m}^{(x)} \in \mathbb{R}^{d_x}, \mathbf{m}^{(y)} \in \mathbb{R}^{d_y}$. This results in a contingency table where the margin clusters are maximally dependent, that is, the table is as far from the product of independent margins as possible. Note that the counts (cluster memberships) are determined by the parameters of the

Voronoi regions in Eq. (1) in the sense of normal vector quantization explained in the beginning of the section.

The multinomial parameters can be marginalized out, and the Bayes factor takes the form [9]

$$BF = \frac{\prod_{ij} \Gamma(n_{ij} + n^{(d)})}{\prod_i \Gamma(n_{i\cdot} + n^{(x)}) \prod_j \Gamma(n_{\cdot j} + n^{(y)})} , \quad (2)$$

where $n_{i\cdot} = \sum_j n_{ij}$ and $n_{\cdot j} = \sum_i n_{ij}$ express the margins. The hyperparameters $n^{(d)}$, $n^{(x)}$, and $n^{(y)}$ arise from Dirichlet priors. We have set all three hyperparameters to unity, which makes the *BF* equivalent to the hypergeometric probability classically used as a dependency measure of contingency tables. It can be shown that for large data set sizes N the logarithmic Bayes factor approaches mutual information $I(I, J)$ between the categorical variables I and J having cluster indices as their values.

2.2. Optimization of AC

In associative clustering the Bayes factor (2) is maximized with respect to the Voronoi prototypes $\{\mathbf{m}^{(x)}\}, \{\mathbf{m}^{(y)}\}$. Because the optimization problem would be combinatorial for hard clusters, the clusters are smoothed enabling the use of gradient methods. Additionally, the denominator of the Bayes factor is given extra weight by introducing constants $\lambda^{(\cdot)}$ that improve the optimization [10]. A choice of $\lambda^{(\cdot)} > 1$ in general favors solutions with uniform margin distributions, i.e. with equal cluster sizes.

Summarizing, the optimization of AC proceeds as follows: (i) Prototypes $\{\mathbf{m}^{(x)}\}$ and $\{\mathbf{m}^{(y)}\}$ are independently initialized by choosing the best of three K-means runs initialized randomly. (ii) We choose $\lambda^{(\cdot)} = 1.2$. (iii) $\sigma_{(\cdot)}$ are chosen by running the algorithm for half of the data and testing on the rest. (iv) The $\{\mathbf{m}^{(x)}\}$ and $\{\mathbf{m}^{(y)}\}$ are optimized with a standard conjugate gradients algorithm, using $\log$ *BF'* as the target function [9]. The reported results are from cross-validation runs.

Note that smoothing is for optimization only: Results are evaluated with *BF*, which translates to having crisp clusters.

2.3. Bootstrapping AC

In AC we do not test any hypotheses , cf. [7], but maximize the Bayes factor to explicitly find dependencies. Since the maximization finds a point estimate for the maximal dependency of the contingency table, it leaves the uncertainty of the solution open.

A widely used "light-weight" method to take into account the uncertainty in clustering is bootstrap [11, 12]. As in [13], we use bootstrap to produce several perturbed clusterings. We wish to find cross clusters (contingency table cells) that signify dependencies between the data sets and are reproducible.

Reproducibility of the found dependencies will be estimated from the bootstrap clusterings as follows.

First, we define what we mean by a significantly dependent cross cluster within a given AC-clustering. The optimized AC model provides a way of estimating how unlikely a cross cluster is, *given that the margins are independent*. For this purpose several (1000 or more) data sets of the same size as the observed one are generated from the marginals of the contingency table (i.e., under the null hypothesis of independence). The cross clusters with the observed amount of data more extreme than that observed by chance with probability 0.01 or less (Bonferroni corrected with the number of cross clusters), are defined to be *significantly dependent cross clusters*.

Next, the two criteria, dependency and reproducibility, will be combined by evaluating how likely it is for each gene pair to occur within the same significantly dependent cross cluster in bootstrap (this is analogous to [13]). This similarity matrix will finally be summarized by hierarchical clustering.

Summarizing, the final AC gene clusters are the bootstrapped, most dependent cross clusters.

2.4. Interpreting the cross clusters

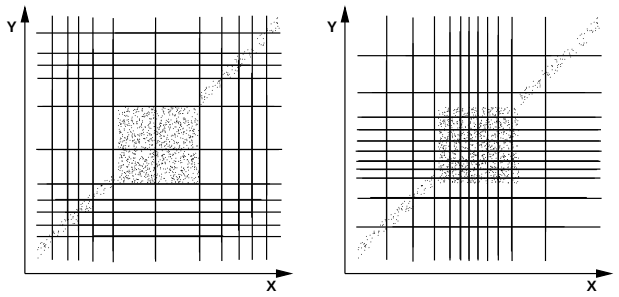
We evaluate which cross clusters are exceptional by their expression or TF binding profile. For determining the exceptionality of the observed cross cluster, for each of them 10,000 random sets of genes were first sampled, each of the same size as the cluster under study. The within-cluster average profiles were then computed for the observed cluster as well as for the simulated ones. A part of the observed average profile was denoted as extreme if it was lower or higher in value than all the simulations.

3. REFERENCE METHODS

First of all, we are interested in comparing AC to the conventional clustering. The main reason here is that it is a way to estimate whether there are any reproducibly dependent subsets in the data. If there are not, the conventional clustering should perform equally well in the dependency maximization task. The reference method should be as similar to AC as possible in other respects. In this work the baseline method will be independent K-means clusterings in both data spaces, since both AC and K-means are prototype-based clustering methods for continuous data like AC.

The main reference method is the information bottleneck (IB) [5, 14]. Although IB works on nominal-valued data, whereas in our setting is the data is continuous, it can be modified for the task. We discretize the continuous data first by K-means, which results in a new algorithm called here K-IB. For discrete data, the closest alternative to AC among information bottleneck methods is the symmetric two-way IB [14]. Our sequential implementation is based on [15].

Note that the final partitions from K-IB are very flexible (they are not constrained to be continuous in the data spaces), and therefore the method is expected to model the dependencies of the margin variables well. As a natural drawback, interpretation of the clusters may be difficult. This is due the final margin clusters consisting of many atomic Voronoi regions, which cannot be summarized easily as needed for example in the determination of



AC K-means

Figure 1. Associative clustering applied on artificial data demonstrating how the dependent subsets of the data become modelled more accurately in AC. Here both margin spaces, denoted by **X** and **Y**, are 1-dimensional, and the figure shows a scatterplot of the data (dots on the plane where **X** and **Y** are the axes). Cluster borders in the **X**-space are shown with the vertical lines and cluster borders in the **Y**-space with horizontal lines. The resulting grid of so-called cross clusters then corresponds to the contingency table; the number of dots within each grid cell gives the amount of data in a contingency table cell. The AC cells are sparse in the bulk of independent data in the middle and denser on the sides where the **X** and **Y** are dependent. K-means, in contrast, focuses on modeling the bulk of the data in the middle.

the extremity of the TF binding or expression. The empirical results support both the good performance of K-IB and the non-localness of the resulting clusters.

4. DEMONSTRATION WITH ARTIFICIAL DATA

One key property of associative clustering is its ability to focus the resources only on the relevant parts of the data spaces. Figure 1 demonstrates this with as simple artificial data sets as possible.

The clusters focus on modeling those regions of the margin data spaces, that is, those subsets of data, where the co-occurring pairs $\mathbf{x}$ and $\mathbf{y}$ are dependent. This is clearly visible as the concentration of the cluster borders on the extremes of both margin spaces in Figure 1.

5. CASE STUDY: DEPENDENCIES BETWEEN GENE EXPRESSION AND TRANSCRIPTION FACTOR BINDING

One of the most common model organisms used in biology is the baker's yeast, *Saccharomyces cerevisiae*. The main reasons for its popularity are the generalizability of its genetic regulation to other eukaryotes an its easy experimental handling.

Gene expression regulation represents a crucial machinery in cell's functionality. It operates on several levels, of which perhaps the most important is transcriptional control. A set of regulatory proteins called *transcription factors (TFs)* bind to DNA in the gene regulatory (promoter) region and can either enhance or suppress the gene's expression, and are the core component in transcriptional regulation. In most cases TFs interact with each other to make up macromolecular complexes before binding to the regulatory regions of DNA. Since TFs are manufactured by expressing the relevant genes, they are the key components of gene interaction networks. We focus on the dependencies between the TFs and gene expression, that is, on the gene regulatory network.

One possibility to study regulatory interactions is by measuring genome-wide expression with microarrays in time series experiments. In time series experiments the goal is to infer causality in the gene regulatory network based on the sequential changes in expression levels. However, since the interaction network between the genes is complicated, discerning the direct change of expression in a time series from noise and the mass of second-order effects can be very difficult, if not impossible. At least a comprehensive, very expensive high resolution time-series experiment with numerous replications would be required. Alternative approaches are thus worth exploring to complement these experiments.

Microarray-based chromatin immunoprecipitation (ChIP) allows measuring the binding strength of the transcription factor proteins on any gene's promoter region [16]. This reveals which TFs are able to bind the specific gene's promoter and are thus potential regulators. But many TFs bind numerous gene promoter regions and are still not operational regulators. Thus the number of false positives, i.e. the number of false regulator relations between TFs and genes, can be very high. Inferring the regulatory relationships based on the binding information alone requires the use of strict statistical criteria, which in turn results in high false negative rate, i.e. the number of real regulatory relations missed may be high. This phenomenon has been noted for example in [6].

We make an assumption that the effects common to both TF binding and gene expression are more likely to be true functional regulatory interactions than those present only in either of the data sets. We combine the functional information (gene expression) and the potential regulator information (TF binding) with associative clustering. Note that the potential incommensurability problems are by-passed by handling the dependencies on contingency level.

The following additional assumptions motivate the application of AC: First, it is assumed that the genes are co-expressed in groups that are unknown, cf. [17, 18]. Second, it is sensible to assume that a common set of transcription factors binds to the co-expressed genes. Otherwise groupwise expression would be very unlikely. This is of course an oversimplification, but it has some biological justification. To be more realistic, we do not assume that all the genes are regulated in such a manner: instead we assume that only *subsets* of genes behave this way, only *a subset* of transcription factors need to be the same, and co-expression needs to take place only in *a subset* of time points.

Associative clustering, when applied to expression and TF binding data, makes precisely these assumptions, and we now aim to find subsets of genes whose expression is

maximally dependent on their transcription factor binding profiles. These sets then act as hypotheses for co-regulation of gene expression, and additionally the potential regulators can be inferred from them.

5.1. Time series gene expression and TF binding

We use the expression data originally published in two different papers [19, 20] (`http://genome-www.stanford.edu/cellcycle/links.html`) and measured during yeast cell cycle. The data consisted of 77 timepoints in total. The transcription factor binding data used here is the updated (2003) version of [16] for 106 transcription factors. In this case study the missing values were imputed with the k-nearest neighbor method ($k = 10$) [21] and logarithms were taken from both of the data sets. Including only the genes present in both data sets resulted in a total of 5618 genes. The amount of clusters were 30 in the expression space and 20 in the TF-binding space.

5.1.1. Numerical results

We first used this data to validate the performance of AC in the two tasks it addresses: maximizing the dependency and keeping the clusters homogeneous. These were measured in 10-fold crossvalidation runs with pre-validated σ for AC and pre-validated number of K-means clusters for K-IB. Pre-validation was analogous for both methods: the data was divided into two equally sized parts, and several parameter values were tried from three different random initializations. Of these the parameter value giving the best AC cost was chosen. The final cross-validation runs were also started from three different random initializations.

The differences in dependency modeling between all the methods were statistically significant for this data pair (10-fold cross-validation, paired t-test; d.f. = 9; $p < 0.001$). Natural logarithmic Bayes factor for AC was 32.27, for IB -13.17, and for K-means -92.30, implying that AC found a very strong dependency between the data sets.

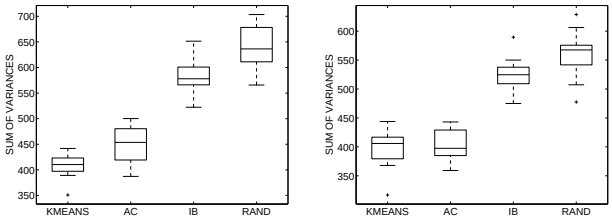
The measure of cluster homogeneity, or actually dispersion, was the sum of the componentwise variances. For this data pair AC produced significantly (10-fold cross-validation, paired t-test; d.f.=9; $p < 0.001$) less dispersed cross clusters and margin clusters than IB. Figure 2 visualizes the cross cluster dispersion for all methods.

5.1.2. Biological results

We sought for biological findings from the bootstrapped AC clusters. The clusters with average distance smaller than 60 (times in the same dependent cross cluster out of 100) and with more than 2 genes were chosen. This resulted in a total of 16 clusters with 307 genes.

Gene ontology classes were enriched statistically significantly in 13 of the 16 clusters (GO::TermFinder [22]; Fisher's exact test, Bonferroni corrected; $p < 0.05$), proving that the clusters are biologically meaningful.

Next, AC results were contrasted to the work presented in the literature by studing the distribution of cell cycle associated genes defined in [19] into AC clusters. This will



Expression space TF binding space

Figure 2. Cross cluster dispersion for all methods in cell-cycle experiments, demonstrating that AC produces clusters that are almost as compact as K-means clusters, whereas IB produces significantly more dispersed clusters. RAND is a kind of an upper limit for cluster dispersion, obtained by randomly assigning samples to clusters.

characterize the differences between the analysis focusing on cyclic behavior presented in [19] and the analysis based on similarities and dependencies defined by AC. In our initial data set we could find 642 cell cycle associated genes. The total number of genes in the reliable clusters was 307 of which 107 were cell cycle gene (Fisher exact test; p-value $< 2.2e-16$). Closer study showed that the cell-cycle genes were statistically significantly enriched in 6 out of 16 of AC clusters (Fisher exact test; Bonferroni corrected p-value < 0.05). This hints that AC finds a part of cell cycle machinery revealed also by other analyses, but additionally suggesting regulators for it, and also produces completely new groups of genes being expressed and regulated similarly during cell cycle. We describe below a representative sample of four clusters, including both cell cycle associated and new clusters.

The first, most notable cluster is a large set of about one hundred genes that all code for ribosomal proteins. These genes are known to be expressed often very homogeneously, and they can also often be found in usual cluster analyses, cf. [23, 24]. The salience of this cluster is not unique to the AC method and it is not cell cycle specific, but it emphasizes that also previously known co-regulated sets of genes are readily detected.

The second cluster of 25 genes (15 associated to cell cycle in [19]) contains many genes (about two thirds) of unknown molecular function. Even the biological process they contribute to may be unknown. The known genes map to such GO categories as "nuclear organisation and biogenesis" and the most reliable transcription factor associated to genes in this cluster was YAP5p/YIR018Wp. This transcription factor is known to be activated by the main regulators (SBF and MBF [25]) of the START of the cell cycle, a time just before DNA replication. This clearly refers to cell cycle regulation and to organization of the nucleus prior to replication. The known genes in this cluster therefore seem to form a knowledge base from which biological experiments can be designed in order to solve the function of the unknown genes in the same cluster.

The third cluster (Fig. 3) contains a significantly high

number (25/33) of genes involved in cell cycle regulation, and, more specifically, at the stage of entry into the mitotic cell cycle (9/33). The main regulator identified in this module is SIP4p/YJL089Wp which is possibly involved in the transcriptional activation regulated by SNF1p/YDR477Wp. This latter signaling factor is required for transcription in response to glucose limitation. Interestingly, a DNA-binding domain of SIP4p/YJL089Wp is similar to that of GAL4p/YPL248Cp transcription factor, involved in galactose response, another route in energy metabolism. Taken together, this cluster contains some clear references to cell cycle regulation on one hand and energy metabolism on the other, and proposes a set of genes that can bridge and connect these two biological processes. Thereby AC offers the possibility to hypothesize on a relation between biological functions, also offering some clues on what genes could be involved.

The fourth cluster contains 9 genes of unknown molecular function or associated biological process, and without a known association to cell cycle. The transcription factor ACE2p/YLR131Cp is associated to this cluster and is known to activate expression of early G1-specific genes, localizes to daughter cell nuclei after cytokinesis and there delays G1 progression in the daughters. Based on this data, the 9 genes can be predicted to act during the G1 phase of the cell-cycle, thus specifying what kind of targeted experiments are needed to establish their function.

6. CONCLUSION

We have proposed a recently developed associative clustering (AC) method [3] for the integration of the possibly incommensurable data sources. AC searches for the dependencies between data sets consisting of co-occurring samples data sets, by maximizing the dependency of two sets of clusters.

AC summarizes dependencies between data sets as clusters of similar samples having similar dependencies. It bypasses the problems of incommensurability between the data sources by parameterizing the data sets separately and representing the dependencies between them with the common data items between the clusters. Such a method is particularly needed for mining functional genomics data where measurements are available about different aspects of the same set of functioning genes. Then a key challenge is to find commonalities between the measurements. The answer should reveal characteristics of the genes, not only characteristics of the measurement setups.

Associative clustering is a general-purpose semiparametric model which learns to fit a new data set instead of being manually tailored. As a result, it is probably not as accurate as more specific models, but it can be expected to be faster and easier to apply to new problems. Its main intended application area is in exploratory data analysis, "looking at the dependencies in the data" in the first stages of a research project.

AC was applied to functional genomics data sets here. Regulatory interactions between gene expression and transcription factor binding were explored. Both trivial (expected) and unexpected findings were made: known regularities, outliers, and hints about unexpected regularities. The distribution of the known cell-cycle genes revealed that AC had captured the statistically significant portion of them into clusters, but not all. This is certainly due to different analysis techniques, but also possibly due to fact that not all the transcription factors regulating the genes during the cell cycle are among public TF data.

Several research directions concerning AC are possible in the future. Most importantly, its applicability to heterogeneous data will be tested. This is important since, while the parameterization and optimization of the clusters in vector-valued data is straightforward, it requires some consideration for example with sequence data. Still, the principle of dependency maximizing clustering itself is naturally applicable whenever it is sensible to form groups of data items. In addition, the amount clusters and their parameterization should be investigated further.

Dependency-searching methods may potentially overfit the data, which is well-known for canonical correlation analysis and can be avoided by regularization. Analogously, also AC can be regularized. We have developed two regularization methods for an earlier method corresponding AC with one fixed margin. "Entropy regularization" was used here because it is easier to apply in practice and has not been shown to be worse than the alternative [10]. In the present case bootstrap also helped. Another related question is which kinds of priors to use for the distributional parameters. The simple constant Dirichlet priors used in this work may be too informative.

We expect that exploratory models of the type introduced here are viable as complementary methods for gathering the necessary prior knowledge for the more specific models.

7. ACKNOWLEDGMENTS

The authors would like to thank Jaakko Peltonen for the code for the sequential symmetric IB. This work has been supported by the Academy of Finland, decisions #79017 and #207467, and by National Technology Agency of Finland through NeoBio-program.

8. REFERENCES

[1] Eran Segal, Ben Taskar, Audrey Gasch, Nir Friedman, and Daphne Koller, "Rich probabilistic models for gene expression," *Bioinformatics*, vol. 17, no. Suppl 1, pp. 243–252, 2003.

[2] G.R.G. Lanckriet, T.D. Bie, N. Cristianini, M.I. Jordan, and W.S. Noble, "A statistical framework for genomic data fusion," *Bioinformatics*, vol. 20, no. 16, pp. 2626–2635, 2004.

[3] Janne Sinkkonen, Janne Nikkilä, Leo Lahti, and Samuel Kaski, "Associative clustering," in *Machine Learning: ECML2004 (Proceedings of the ECML'04, 15th European Conference on Machine Learning)*, Boulicaut, Esposito, Giannotti, and Pedreschi, Eds., pp. 396–406. Springer, Berlin, 2004.

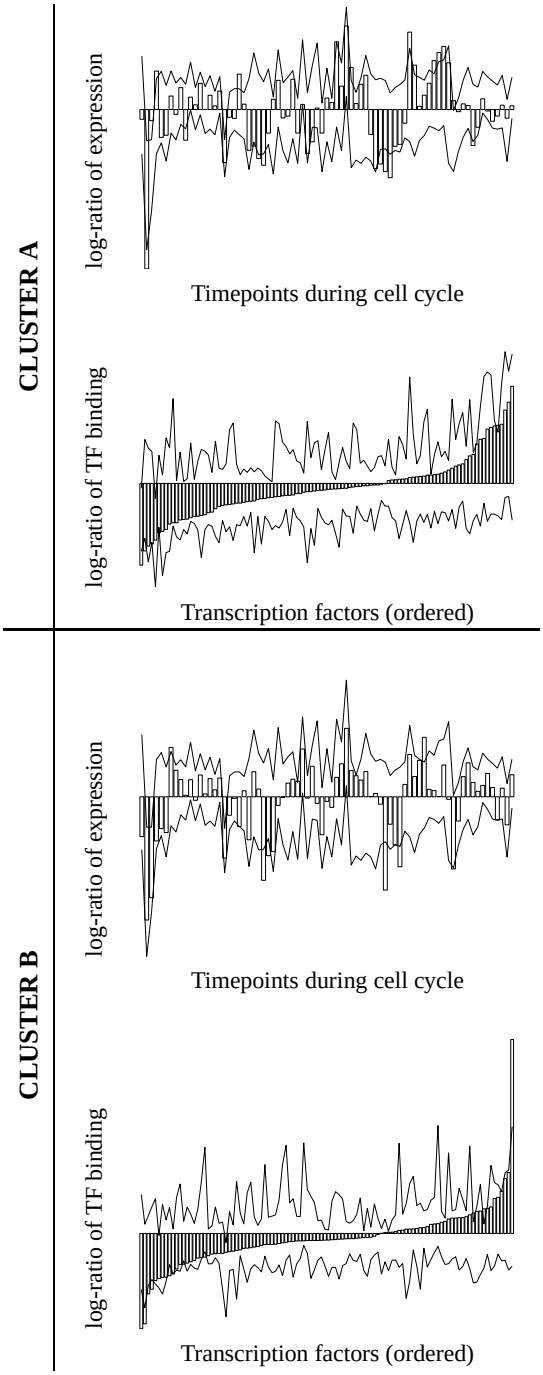


Figure 3. Two examples of bootstrapped cross clusters, associated to cell cycle, that reveal both known and novel dependencies between gene expression and TF binding. The upper figures: the average expression profiles (bars) with confidence intervals (curves). The periodicity of the cell cycle in the expression is visible. The lower figures: the average TF-binding profile with confidence intervals. TFs topping the confidence intervals are considered reliably extreme. **Cluster A:** There was only one reliable TF binding, SIP4, and that is known to regulate cell-cycle. **Cluster B:** SIP4 binds the genes also in this cluster, but additionally there is one extremely strongly binding TF, SFL1 (the rightmost bar). Its putative regulation of the gene cluster during cell cycle is a new finding.

[4] G. J. McLachlan, Kim-Anh Do, and Christophe Ambroise, *Analyzing microarray gene expression data*, Wiley, New York, 2004.

[5] Naftali Tishby, Fernando C. Pereira, and William Bialek, "The information bottleneck method," in *Proceedings of The 37th Annual Allerton Conference on Communication, Control, and Computing*, Bruce Hajek and R. S. Sreenivas, Eds., pp. 368–377. University of Illinois, Urbana, Illinois, 1999.

[6] Z. Bar-Joseph, G.K. Gerber, T.I. Lee, N.J. Rinaldi, J.Y. Yoo, F. Robert, D.B. Gordon nd E. Fraenkel, T.S. Jaakkola, and D.K. Gifford R.A. Young, "Computational discovery of gene modules and regulatory networks," *Nature Biotechnology*, vol. 21, pp. 1337–1342, 2003.

[7] I. J. Good, "On the application of symmetric Dirichlet distributions and their mixtures to contingency tables," *Annals of Statistics*, vol. 4, no. 6, pp. 1159–1189, 1976.

[8] Adrian E. Kass, Robert E.; Raftery, "Bayes factors," *Journal of the American Statistical Association*, vol. 90, pp. 773–795, 1995.

[9] J. Sinkkonen, S. Kaski, J. Nikkilä, and L. Lahti, "Associative Clustering (AC): technical details," Tech. Rep. A84, Helsinki University of Technology, Laboratory of Computer and Information Science, Espoo, Finland, 2005.

[10] S. Kaski, J. Sinkkonen, and A. Klami, "Discriminative clustering," *Neurocomputing*, 2005, Accepted for publication.

[11] B. Efron and R. Tibshirani, *An Introduction to the Bootstrap*, Chapman&Hall, New York, 1993.

[12] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, Springer, New York, 2001.

[13] M. K. Kerr and G. A. Churchill, "Bootstrapping cluster analysis: Assessing the reliability of conclusions from microarray experiments," *Proceedings of the National Academy of Sciences*, vol. 98, pp. 8961–8965, 2001.

[14] Nir Friedman, Ori Mosenzon, Noam Slonim, and Naftali Tishby, "Multivariate information bottleneck," in *Proceedings of UAI'01, The Seventeenth Conference on Uncertainty in Artificial Intelligence*, pp. 152–161. Morgan Kaufmann Publishers, San Francisco, CA, 2001.

[15] Noam Slonim, *The information bottleneck: theory and applications*, Ph.D. thesis, Hebrew University, Jerusalem, 2002.

[16] T. I. Lee, N. J. Rinaldi, F. Robert, D. T. Odom, Z. Bar-Joseph, G. K. Gerber, N. M. Hannett, C. T. Harbison, C. M. Tomphson, I. Simon, J. Zeitlinger, E. G. Jennings, H. L. Murray, D. B. Gordon, B. Ren, J. J. Wyrick, J.-B. Tagne, T. L. Volkert, E. Fraenkel, D. K. Gifford, and R. A. Young, "Transcriptional regulatory networks in Saccharomyces cerevisiae," *Science*, vol. 298, pp. 799–804, 2002.

[17] Eran Segal, Michael Shapira, Aviv Regev, Dana Pe'er, David Botstein, Daphne Koller, and Nir Friedman, "Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data.," *Nature genetics*, vol. 34, pp. 166–176, 2003.

[18] N. Friedman, M. Linial, I. Nachman, and D. Pe'er, "Using Bayesian networks to analyze expression data," *Journal of Computational Biology*, vol. 7, pp. 559–584, 2000.

[19] P. T. Spellman, G. Sherlock, M. Q. Zhang, V. R. Iyer, K. Anders, M. B. Eisen, P. O. Brown, D. Botstein, and B. Futcher, "Comprehensive identification of cell cycle-regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization," *Molecular Biology of the Cell*, vol. 9, pp. 3273–97, 1998.

[20] R. J. Cho, M. J. Campbell, E. A. Winzeler, L. Steinmetz, A. Conway, L. Wodickaa, T. G. Wolfsberg, A. E. Gabrielian, D. Landsman, D. J. Lockhart, and R. W. Davis, "A genome-wide transcriptional analysis of the mitotic cell cycle.," *Molecular Cell*, vol. 2, pp. 65–73, 1998.

[21] Olga Troyanskaya, Michael Cantor, Gavin Sherlock, Pat Brown, Trevor Hastie, Robert Tibshirani, David Botstein, and Russ. B. Altman, "Missing value estimation methods for dna microarrays," *Bioinformatics*, vol. 17, pp. 520–5, 2001.

[22] E.I Boyle, S. Weng, J. Gollub, H. Jin, D. Botstein, J.M. Cherry, and G. Sherlock, "Go::termfinder - open source software for accessing gene ontology information and finding significantly enriched gene ontology terms associated with a list of genes.," *Bioinformatics*, vol. 20, pp. 3710–3715, 2004.

[23] Janne Nikkilä, Petri Törönen, Samuel Kaski, Jarkko Venna, Eero Castrén, and Garry Wong, "Analysis and visualization of gene expression data using self-organizing maps," *Neural Networks*, vol. 15, no. 8-9, pp. 953–966, 2002, Special issue on New Developments on Self-Organizing Maps.

[24] M.A. Beer and S. Tavazoie, "Predicting gene expression from sequence," *Cell*, vol. 117, pp. 185–198, 2004.

[25] C. E. Horak, N. M. Luscombe, J. Qian, P. Bertone, S. Piccirrillo, M. Gerstein, and M. Snyder, "Complex transcriptional circuitry at the G1/S transition in Saccharomyces cerevisiae," *Genes and Development*, vol. 16, pp. 3017–3033, 2002.