Stefan Kramer and Neil Lawrence (Eds.)

Machine Learning in Systems Biology

Proceedings of the Fifth International Workshop
July 20-21, 2011
Vienna, Austria

Preface

Molecular biology and all the biomedical sciences are undergoing a true revolution as a result of the emergence and growing impact of a series of new disciplines/tools sharing the "-omics" suffix in their name. These include in particular genomics, transcriptomics, proteomics and metabolomics, devoted respectively to the examination of the entire systems of genes, transcripts, proteins and metabolites present in a given cell or tissue type.

The availability of these new, highly effective tools for biological exploration is dramatically changing the way one performs research in at least two respects. First, the amount of available experimental data is not a limiting factor any more; on the contrary, there is a plethora of it. Given the research question, the challenge has shifted towards identifying the relevant pieces of information and making sense out of it (a "data mining" issue). Second, rather than focus on components in isolation, we can now try to understand how biological systems behave as a result of the integration and interaction between the individual components that one can now monitor simultaneously (so called "systems biology").

Taking advantage of this wealth of "genomic" information has become a conditio sine qua non for whoever ambitions to remain competitive in molecular biology and in the biomedical sciences in general. Machine learning naturally appears as one of the main drivers of progress in this context, where most of the targets of interest deal with complex structured objects: sequences, 2D and 3D structures or interaction networks. At the same time bioinformatics and systems biology have already induced significant new developments of general interest in machine learning, for example in the context of learning with structured data, graph inference, semi-supervised learning, system identification, and novel combinations of optimization and learning algorithms.

This book contains the scientific contributions presented at the Fifth International Workshop on Machine Learning in Systems Biology (MLSB 2011), held in Vienna, Austria, from July 20 to 21, 2011. The workshop was organized as an official satellite meeting of the 19th Annual International Conference on Intelligent Systems for Molecular Biology and the 10th European Conference on Computational Biology (ISMB/ECCB 2011). The workshop was supported by the PASCAL2 Network of Excellence, under the IST programme of European Union, and by the City of Vienna. The aim of the workshop was to contribute to the cross-fertilization between the research in machine learning methods and their applications to systems biology (i.e., complex biological and medical questions) by bringing together method developers and experimentalists. A non-exhaustive list of the topics of interest to the workshop were:

Methods

- Machine Learning Algorithms
- Bayesian Methods

- Data integration/fusion
- Feature/subspace selection
- Clustering
- Biclustering/association rules
- Kernel Methods
- Probabilistic inference
- Structured output prediction
- Systems identification
- Graph inference, completion, smoothing
- Semi-supervised learning

Applications

- Sequence Annotation
- Gene Expression and post-transcriptional regulation
- Inference of gene regulation networks
- Gene prediction and whole genome association studies
- Metabolic pathway modeling
- Signaling networks
- Systems biology approaches to biomarker identification
- Rational drug design methods
- Metabolic reconstruction
- Protein function and structure prediction
- Protein-protein interaction networks
- Synthetic biology

The technical program of the workshop consisted of invited lectures and oral presentations. Invited lectures were given by Eleazar Eskin, Magnus Rattray and Robert Küffner. 20 oral presentations were given, with extended abstracts included in this booklet, each reviewed by at least two reviewers. We would like to thank all the people contributing to the technical programme, the scientific program committee, Steven Leard from ISCB (local organization and registration) and Jörg Wicker (website) for making the workshop possible.

July 2011

Stefan Kramer and Neil Lawrence
Program Chairs
MLSB 2011

Organization

Program Chairs

Stefan Kramer (TU München, Germany)
Neil Lawrence (The University of Sheffield, UK)

Program Committee

Florence d'Alché-Buc (University of Evry, France)
Hendrik Blockeel (Katholieke Universiteit Leuven, Belgium)
Sašo Džeroski (Jožef Stefan Institute, Slovenia)
Paolo Frasconi (Università degli Studi di Firenze, Italy)
Pierre Geurts (University of Liège, Belgium)
Dirk Husmeier (Biomathematics and Statistics Scotland, UK)
Lars Kaderali (University of Heidelberg, Germany)
Samuel Kaski (Helsinki University of Technology, Finland)
Ross King (Aberystwyth University, UK)
Stefan Kramer (TU München, Germany)
Neil Lawrence (The University of Sheffield, UK)
Elena Marchiori (Vrije Universiteit Amsterdam, The Netherlands)
Yves Moreau (Katholieke Universiteit Leuven, Belgium)
Sach Mukherjee (University of Warwick, UK)
Mahesan Niranjan (University of Southampton, UK)
John Pinney (Imperial College London, UK)
Gunnar Rätsch (Friedrich Miescher Laboratory of the Max Planck Society, Germany)
Magnus Rattray (The University of Sheffield, UK)
Simon Rogers (University of Glasgow, UK)
Juho Rousu (University of Helsinki, Finland)
Céline Rouveirol (University of Paris XIII, France)
Yvan Saeys (University of Gent, Belgium)
Guido Sanguinetti (University of Edinburgh, UK)
Peter Sykacek (BOKU University, Austria)
Fabian Theis (TU München, Germany)
Ljupco Todorovski (University of Ljubljana, Slovenia)
Koji Tsuda (National Institute of Advanced Industrial Science and Technology, Japan)
Jean-Philippe Vert (Ecole des Mines, France)
Louis Wehenkel (University of Liège, Belgium)
Filip Zelezny (Czech Technical University in Prague, Czech Republic)

Sponsors

PASCAL2 Network of Excellence
City of Vienna

Table of Contents

Invited Talks

Contributions

Known and Unknown Confounding in Genetic Studies

Eleazar Eskin

University of California, Los Angeles, USA

Abstract. Variation in human DNA sequences account for a significant amount of genetic risk factors for common disease such as hypertension, diabetes, Alzheimer's disease, and cancer. Identifying the human sequence variation that makes up the genetic basis of common disease will have a tremendous impact on medicine in many ways. Recent efforts to identify these genetic factors through large scale association studies which compare information on variation between a set of healthy and diseased individuals have been remarkably successful. However, despite the success of these initial studies, many challenges and open questions remain on how to design and analyze the results of association studies. As several recent studies have demonstrated, confounding factors such as batch effects, population structure, and measurement errors can complicate genetics analysis by causing many spurious associations. Yet little is understood about how these confounding factors affect analyses and how to correct for these factors. In this talk I will discuss several recently developed methods based on linear mixed models for correcting for both known and unknown confounding factors in genetic studies.

Modeling Gene Expression Time-Series with Bayesian Non-Parametrics

Magnus Rattray

The University of Sheffield, Sheffield, UK

Abstract. Bayesian non-parametric methods are a natural approach to fitting models with continuous parameters or unbounded parameter set cardinality. We are applying these methods to diverse models of time-series gene expression data. Example applications include differential equation models of transcriptional regulation, clustering data sampled at uneven times and phylogenetic models of gene expression change over evolutionary time. We use continuous-time Gaussian processes to model the time-evolution of gene expression and protein activation/concentration in time. Dirichlet processes can also be used to model an unbounded set of Gaussian process models. I will present results where we apply these methods to gene expression time-course data from embryonic development in Drosophila.

This is joint work with Neil Lawrence, Antti Honkela, Michalis Titsias and James Hensman.

Successful Strategies to Gene Regulatory Network Inference

Robert Küffner

Ludwig-Maximilians-Universität München, Munich, Germany

Abstract. The inference of gene regulatory networks from mRNA expression data is characterized by the development of many different approaches with their specific performances, data requirements, and inherent biases. Based on a recent community-wide challenge, the Dialogue on Reverse Engineering Assessment and Methods (DREAM), the so far largest assessment of inference approaches has been conducted. The accuracy of predictions was evaluated against experimentally supported interactions in the procaryote model organism E. coli, the eucaryote model organism S. cerevisiae and in silico target systems. Over thirty independently contributed methods were analyzed including well-known approaches based on lasso, mutual information and Bayesian networks but also a range of novel strategies. For instance, the novel algorithms based on random forests and ANOVA outperformed established tools significantly. Further analysis revealed not only which inference strategies are particularly successful but also which kind of specific information was utilized from the different types of experimental measurements. At the same time, the performance of the individual approaches varied in different network motifs or target systems. By integrating individual predictions into community predictions the performance improved overall and became markedly more robust. This principle is known as the wisdom of crowds: a solution derived from a community of independent decision makers will be better, on average, than any individual solution. Based on community predictions, we also constructed the first comprehensive gene regulatory model for the human pathogen Staphylococcus aureus.

Epistasis Detection in Subquadratic Time

Panagiotis Achlioptas[1], Bernhard Schölkopf[2], and Karsten M. Borgwardt[3]

[1] University of Crete, Greece
[2] Max Planck Institute for Intelligent Systems, Tübingen, Germany
[3] Max Planck Institutes Tübingen, Germany

Abstract. Genome-wide association studies (GWAS) have not been able to discover strong associations between many complex human diseases and single genetic loci. Mapping these phenotypes to pairs of genetic loci is hindered by the huge number of candidates leading to enormous computational and statistical problems. In GWAS on single nucleotide polymorphisms (SNPs), one has to consider in the order of 10^{10} to 10^{14} pairs, which is infeasible in practice. In this article, we give the first algorithm for 2-locus genome-wide association studies subquadratic in the number of SNPs n.

Core to this algorithm is the subquadratic lightbulb algorithm for detecting pairs of highly correlated binary random variables [3]. This algorithm differs in three aspects from epistasis detection: First, SNPs have typically three or more states and are not binary. Second, epistasis detection is interested in the pairs of SNPs that maximally differ between cases and controls, not SNP pairs correlated across the whole dataset. Third, epistasis detection is based on differences in Pearson's correlation coefficient, while Paturi *et al.* use a correlation score based on the probability of two random variables being in the same state.

We overcome these three problems by proposing a lightbulb algorithm that finds pairs of variables with maximum differences in correlation between cases and controls. We use Locality Sensitive Hashing [2] to binarize SNPs and to get an approximation of Pearson's correlation coefficient via the lightbulb algorithm. Through our contributions, the favourable subquadratic runtime of the lightbulb algorithm can be transferred to the problem of epistasis detection.

The running time of our algorithm is data-dependent, but large experiments over real genomic data suggest that it scales empirically as $n^{\frac{3}{2}}$. As a result, our algorithm can easily cope with $n \sim 10^7$, i.e., it can efficiently search all pairs of SNPs in the human genome. A manuscript describing our finding in full detail will be published at the 17th ACM SIGKDD Conference on Knowledge Discovery and Data Mining in San Diego (KDD 2011) [1].

References

1. P. Achlioptas, B. Schölkopf, and K. Borgwardt. Two-locus association mapping in subquadratic time. In *KDD*, 2011.
2. M. S. Charikar. Similarity estimation techniques from rounding algorithms. In *STOC*, 2002.
3. R. Paturi, S. Rajasekaran, and J. Reif. The light bulb problem. In *Proc. 2nd Annu. Workshop on Comput. Learning Theory*, pages 261 – 268, San Mateo, CA, 1989. Morgan Kaufmann.

A Study of Dynamic Time Warping for the Inference of Gene Regulatory Relationships

Matthias Böck, Constanze Schmitt, and Stefan Kramer

Technische Universität München,
Institut für Informatik Lehrstuhl I12 - Bioinformatik,
Boltzmannstr. 3, 85748 Garching b. München, Germany
{matthias.boeck,constanze.schmitt,stefan.kramer}@in.tum.de
http://wwwkramer.in.tum.de

Abstract. The paper assesses different variants of *Dynamic Time Warping* (*DTW*) for the inference of gene regulatory relationships. Apart from *DTW* on continuous time series, we present a novel angle-based discretization approach and a distance learning method that is combined with *DTW* to find new gene interactions. A positive influence of the distance optimization on the performance of the alignments of gene expression profiles could not yet be established. However, our results show that discretization can be important to the outcome of the alignments. The discretization is not only able to keep the important features of the time series, it is also able to perform better than regular *DTW* on the original data.

Keywords: Time series alignment, gene expression, Dynamic Time Warping, discretization

1 Introduction

The analysis of time series data is still one of the most challenging fields and occurs in many scientific disciplines. Steady state data can only give a snapshot of the actual dynamics while time series allow to study the processes over time and to capture the dependencies between the forces and protagonists. In this study we are focusing on gene expression data and how to infer the interactions and dependencies from it. We propose a slope based discretization of given microarray data and a new alignment approach, combining the ideas of *Dynamic Time Warping* (*DTW*) with *Stochastic Local Search* (*SLS*). Building of alignments of discretized profiles is supposed to be robust against noisy data and to overcome the assumption of strictly linear relationships between two interacting genes. A basic assumption for the alignment of time series is that co-regulated genes also show similar expression behaviors over time and hence similar amplitudes which can be aligned with suitable transformations. Testing and evaluation of the approach has been done with two biological data sets from Pramila *et al.* [6] and Tu *et al.* [8] following the cell cycles of *S. cerevisiae*. As a benchmark network the protein-protein interaction network from the STRING database (v8.3) [4] is

used. It is clear that the PPI network is only able to cover part of the gene regulatory processes but still, observations on this level can provide insight into the performance of the methods.

2 Method

Dynamic time warping (DTW) was introduced in the 1960s [2] and has been intensively used for speech recognition and other fields, like handwriting recognition systems, gesture recognition, signal processing and gene expression time series clustering [1]. The basic idea of this unsupervised learning approach is that a suitable distance measure, which is most generally the Euclidean distance, allows the algorithm to stretch (or compress) the time and expression rate axis to find the most suitable fit of two given time series. The DTW algorithm will be described briefly in the following. Consider two given sequences $S = s_1, ..., s_n$ and $T = t_1, ..., t_m$ and a given distance function $\delta(s_i, t_j)$ with $1 \leq i \leq n$ and $1 \leq j \leq m$, DTW tries then to minimize with the given δ over all possible warping paths between the two given sequences based on the cumulative distance for each path. This is solved by a recursive dynamic programming approach for each $i \in [1, ..., n]$ and $j \in [1, ..., m]$:

$$DTW(i,j) = \begin{cases} 0 & \text{for } i = j = 0 \\ min \begin{cases} DTW_{i-1,j-1} + 2 \cdot \delta(s_i, t_j) \\ DTW_{i-1,j} + \delta(s_i, t_j) \\ DTW_{i,j-1} + \delta(s_i, t_j) \end{cases} & \text{for } i, j > 0 \\ \infty & \text{otherwise} \end{cases} \quad (1)$$

$DTW[n, m]$ is the total distance $DTW(S, T)$ and can be calculated in $O(nm)$. The traceback through the matrix D gives the optimal warping of the aligned sequences.

We present in the following the framework for a discretized sequence alignment approach focusing on the analysis of gene expression time series. In contrast to other existing methods the approach deals also with anti-correlated time series and uses a supervised method to infer a data specific distance matrix for the alignment. The result is a scoring matrix for the pairwise distances between the measured genes.

We use a list of 1129 cell cycle dependent genes, suggested by Rowicka *et al.* [7], for the evaluation. Genes, for which no PPI data or probe sets in the microarray experiments exist, were discarded from our set. We use cubic smoothing splines to interpolate the time series for missing values and smoothen out smaller fluctuations from experimental or biological noise.

The discretization of each time series for each gene is done according to the steepness of the expression change $\delta\ exp$ between two consecutive time steps. This is done by calculating the angle: $\alpha =$ **atan** $\delta\ exp \cdot \frac{180}{\pi}$. The angles are then

discretized into positive and negative (increasing or decreasing) integer values according to a predefined threshold. Defining the threshold is done by dividing the possible range for increases or decreases of 180 degrees into n subsectors, with a range of $\frac{180}{n}$ degrees each. Each of this sectors represents a possible range of angles for the increase or decrease between two consecutive time points and has assigned a discrete value. For n sectors the range of these values would be $[-\frac{n}{2}, -\frac{n}{2}+1, ..., \frac{n}{2}-1, \frac{n}{2}]$.

A crucial point for the quality of the alignments is the choice of a suitable distance matrix which defines the distances between the discretized values of the time series. This motivates our supervised approach to use a set of already known interacting genes I to infer the distance matrix δ. These gene pairs are chosen randomly from the PPI data along with a further randomly chosen set of not interacting genes N. The resulting δ should minimize the distance for I and maximize the distance for N. Since DTW is not differentiable, we apply a combination of *Stochastic Local Search* (SLS) and simulated annealing for the stepwise improvement of δ. For a more detailed introduction to SLS, we refer to the work of Hoos and Stützle [3].

We imposed three constraints on the step-wise altering of the distance matrix δ to reduce the search space and to keep the basic distance structure between different bins of angles: $\delta(i,j) = 0$ for $i = j$, $\delta(i,j) = \delta(j,i)$ and $\delta(i,j) < \delta(i,j-1)$.

The resulting distance matrix is then used for the calculations of the alignments and the score defines the distance between each pair of genes. Additional alignments are done for each comparison with flipped signs for one of the time series to find anti-correlated pairs.

3 Evaluation

We compare the performance on the two biological data sets to the results with simple correlation, MRNET (mutual information), DTW and $DDTW$ (a modification of DTW which uses for the discretization the first derivative for each point) [5]. DTW_{disc} applies our discretization method with different numbers of sectors (n) and calculates the alignments with DTW. DTW_{SLS} additionally applies the distance optimization before the calculations. The evaluation is done based on ROC curves and the AUC. Interactions are undirected and hence only a two class problem considered, interaction predicted or not. We applied to both data sets smoothing splines and did a z-score transformation before the calculations. An excerpt of our results of AUC values is shown in Table 1.

	cor	MRNET	DTW	$DDTW$	DTW_{disc}	DTW_{SLS}
Pramila *et al.* (961 genes)	0.59	0.58	0.5	0.46	0.61 (n=7)	0.58 (n=7)
Tu *et al.* (944 genes)	0.54	0.54	0.62	0.62	0.59 (n=5)	0.60 (n=5)

Table 1. Comparison of the AUC values for the different methods evaluated with PPI data from STRING.

In general, the different DTW approaches perform better than correlation or MRNET, except for the case of regular DTW and $DDTW$ on the Pramila *et al.* data. The results of DTW_{disc} show that the discretization keeps the important features and even with a small number of sectors performs well. The approach of DTW_{SLS} seems, to this date, not to be able to improve the distance measure and achieves slightly smaller AUC values. The discretization method outperforms DTW and $DDTW$ on the Pramila *et al.* data and performs only slightly worse on the other data set.

4 Conclusion

In the paper, we investigated several variants of *Dynamic Time Warping* for the detection of gene regulatory relationships. While the supervised optimization of the distance matrix did not lead to improvements, a novel discretization approach seems, even with a small number of defined sectors, able to keep the main features and appears as a suitable qualitative transformation for time series alignments. On the biological data sets, our approach seems to be more stable compared to DTW and $DDTW$. In contrast to correlation-based methods, DTW is also able to infer the orientation of the time shift through the traceback and hence able to hint at possible causalities. We intend to make use of this information and further evaluate the robustness of the discretization method compared to DTW and $DDTW$.

References

1. J. Aach and G.M. Church. Aligning gene expression time series with time warping algorithms. *Bioinformatics*, 17(6):495–508, 2001.
2. R. Bellman and R. Kalaba. On adaptive control processes. *IRE Transactions on Automatic Control*, 4(2):1–9, November 1959.
3. H. H. Hoos and T Stützle. *Stochastic Local Search - Foundation and Applications*. Elsevier, 2005.
4. L. J. Jensen, M. Kuhn, M. Stark, S. Chaffron, C. Creevey, J. Muller, T. Doerks, P. Julien, A. Roth, M. Simonovic, P. Bork, and C. von Mering. STRING 8 : a global view on proteins and their functional interactions in 630 organisms. *Nucleic Acids Research*, 37(suppl 1):D412–D416, 2009.
5. E. J. Keogh and M. J. Pazzani. Derivative Dynamic Time Warping. *Proceedings of the SIAM International Conference on Data Mining*, pages 1–11, 2001.
6. T. Pramila, W. Wu, S. Miles, W.S. Noble, and L.L. Breeden. The Forkhead transcription factor Hcm1 regulates chromosome segregation genes and fills the S-phase gap in the transcriptional circuitry of the cell cycle. *Genes and Development*, 20(16):2266–2278, 2006.
7. M. Rowicka, A. Kudlicki, B. P. Tu, and Z. Otwinowski. High-resolution timing of cell cycle-regulated gene expression. *Proceedings of the National Academy of Sciences*, 104(43):16892–16897, 2007.
8. B. P. Tu, A. Kudlicki, M. Rowicka, and S. L. McKnight. Logic of the yeast metabolic cycle: temporal compartmentalization of cellular processes. *Science*, 310:1152–1158, 2005.

A New Theoretical Angle to Semi-supervised Output Kernel Regression for Protein-protein Interaction Network Inference

Céline Brouard[1], Florence d'Alché-Buc[1], and Marie Szafranski[2,1]

[1] IBISC, EA 4526, Université d'Évry Val d'Essonne, F–91025 Évry cedex, France
{celine.brouard, florence.dalche, marie.szafranski}@ibisc.fr
[2] ÉNSIIE, F–91025 Évry cedex, France

1 Background

Recent years have witnessed a surge of interest for network inference in biological networks. *In silico* prediction of protein-protein interaction (PPI) networks is motivated by the cost and the difficulty to experimentally detect physical interactions between proteins. The underlying hypothesis is that some input features relative to the proteins provide valuable information about the presence or the absence of a physical interaction. The main approaches devoted to this task fall into two families: supervised approaches, which aim at building pairwise classifiers able to predict if two proteins interact, from a dataset of labeled pairs of proteins [1–5], and matrix completion approaches that fits into an unsupervised setting with some constraints [6, 7] or directly into a semi-supervised framework [8, 9].

Let us define $\mathcal{O}$ the set of descriptions of the proteins we are interested in. In this paper, we have chosen to convert the binary pairwise classification task into an output kernel learning task as in [3, 4]. This is made possible by noticing that a Gram matrix K_{Y_ℓ} on the training data $\mathcal{O}_\ell$ can be defined from the adjacency matrix using any kernel that encodes the proximities of proteins in the network (for instance a diffusion kernel [10]). We assume that a positive definite kernel κ_y: $\mathcal{O} \times \mathcal{O} \to \mathbb{R}$ underlies this Gram matrix such that $\forall i, j \leq \ell, K_{Y_\ell}(i, j) = \kappa_y(o_i, o_j)$. Moreover, there exists an Hilbert space $\mathcal{F}_y$, called the feature space, and a feature map $y : \mathcal{O} \to \mathcal{F}_y$ such that $\forall (o, o') \in \mathcal{O}, \kappa_y(o, o') = \langle y(o), y(o') \rangle_{\mathcal{F}_y}$.
The assumption underlying output kernel learning is that an approximation of κ_y will provide valuable information about the proximity of proteins in terms of nodes in the interaction graph. This approximation is built from the inner product between the outputs of a single variable function $h : \mathcal{O} \to \mathcal{F}_y$: $\widehat{\kappa_y}(o, o') = \langle h(o), h(o') \rangle_{\mathcal{F}_y}$. This allows one to reduce the problem of learning from pairs to learning a single variable function with values in the output feature space. This supervised regression task is referred to as Output Kernel Regression (OKR). Once the output kernel is learnt, a classifier f_θ is defined from the approximation $\widehat{\kappa_y}$ by thresholding its output values:

$$f_\theta(o, o') = \mathrm{sgn}(\widehat{\kappa_y}(o, o') - \theta) \, .$$

2 RKHS for vector-valued functions for supervised and semi-supervised OKR

In the case of OKR, the function to be learnt is not real-valued but vector-valued in the output Hilbert space. If we want to benefit from the theoretical framework of Reproducing Hilbert Space theory (RKHS), well appropriate to regularization, we need to turn to the proper RKHS theory, devoted to vector-valued functions, which was introduced in [13] and developed in [14]. In this theory, kernels are operator-valued and applied to vectors in the given output Hilbert space. While being very powerful, this theory is still underused.

Supervised setting

In this work, the RKHS theory devoted to functions with values in a Hilbert space provides us with a general framework for OKR. Let $\mathcal{F}_y$ be an Hilbert space. Let $S_\ell = \{(o_i, \mathbf{y}_i)\}_{i=1}^{\ell} \subseteq \mathcal{O} \times \mathcal{F}_y$ be a set of labeled examples, and $\mathcal{H}$ be a RKHS with reproducing kernel $\mathcal{K}_x$. We focus here on the to penalized least square cost in the case of vector-valued functions:

$$\underset{h \in \mathcal{H}}{\operatorname{argmin}}\, J(h) = \sum_{i=1}^{\ell} \|h(o_i) - \mathbf{y}_i\|_{\mathcal{F}_y}^2 + \lambda_1 \|h\|_{\mathcal{H}}^2 \text{ , with } \lambda_1 > 0. \tag{1}$$

Michelli & Pontil [14] have shown that the minimizer of this problem admits an expansion $\hat{h}(\cdot) = \sum_{j=1}^{\ell} \mathcal{K}_x(o_j, \cdot)\mathbf{c}_j$, where the vectors $\mathbf{c}_j \in \mathcal{F}_y, j = \{1, \cdots, \ell\}$, satisfy the equations:

$$\mathbf{y}_j = \sum_{i=1}^{\ell} \mathcal{K}_x(o_i, o_j)\mathbf{c}_i + \lambda_1 \mathbf{c}_j \ . \tag{2}$$

To benefit from this theory, we must define a suitable input operator-valued kernel. OKR is extended to data described by some input scalar kernel. The training input set is now defined by an input Gram matrix K_{X_ℓ}, which encodes for the properties of the training objects $\mathcal{O}_\ell$. As in the output case, the coefficients of the Gram matrix are supposed to be defined from a positive definite input kernel function $\kappa_x : \mathcal{O} \times \mathcal{O} \to \mathbb{R}$, with $\forall i, j \leq \ell, K_{X_\ell}(i, j) = \kappa_x(o_i, o_j)$. We define an operator-valued kernel $\mathcal{K}_x$ from this scalar kernel:

$$\mathcal{K}_x(o, o') = \kappa_x(o, o') \times I_{\mathcal{F}_y} \ , \tag{3}$$

with $I_{\mathcal{F}_y}$, the identity matrix of size $dim(\mathcal{F}_y) \times dim(\mathcal{F}_y)$. The theorem from [13, 14] ensures that a RKHS can be built from it. Starting from the results existing in the supervised case for the penalized least-square cost, we show that with this choice of the operator-valued kernel, we can derive a closed-form solution.

Proposition 1. *When $\mathcal{K}_x$ is defined by mapping* (3)*, the solution of Problem* (1) *reads*

$$C = Y_\ell (K_{X_\ell} + \lambda_1 I_\ell)^{-1} \ , \tag{4}$$

where $Y_\ell = (\mathbf{y}_1, \cdots, \mathbf{y}_\ell)$, $C = (\mathbf{c}_1, \cdots, \mathbf{c}_\ell)$, and I_ℓ is the $\ell \times \ell$ identity matrix.

It is worth noting that we directly retrieve the extension of kernel ridge regression to output kernels proposed by [15].

Semi-supervised setting

In biology, it is much easier to get a detailed description of the properties of a protein compared to the cost of experimental methods used to detect physical interactions between two proteins. To benefit from the usually large amount of unlabeled data, we need to extend OKR to semi-supervised learning. A powerful approach is based on graph-based regularization that forces the prediction function to be smooth on the graph describing similarities between inputs. Enforcing smoothness of the function permits to propagate output labels over close inputs as shown in [11, 12]. [12] have proposed to explicitly embed such ideas into the framework of regularization within RKHS for real-valued functions.

Let $S_\ell = \{(o_i, \mathbf{y}_i)\}_{i=1}^{\ell}$ be a set of labeled examples and $S_u = \{o_i\}_{i=\ell+1}^{\ell+u}$ a set of unlabeled examples. Let $\mathcal{H}$ be a RKHS with reproducing kernel $\mathcal{K}_x$, and a symmetric matrix W with positive values measuring the similarity of objects in the input space. We consider the following optimization problem:

$$\underset{h\in\mathcal{H}}{\operatorname{argmin}}\, J(h) = \sum_{i=1}^{\ell} \|h(o_i) - \mathbf{y}_i\|^2_{\mathcal{F}_y} + \lambda_1 \|h\|^2_{\mathcal{H}} + \lambda_2 \sum_{i,j=1}^{\ell+u} W_{ij} \|h(o_i) - h(o_j)\|^2_{\mathcal{F}_y} \ , \quad (5)$$

with λ_1 and $\lambda_2 > 0$.

We state and prove a new representer theorem devoted to semi-supervised learning in RKHS with vector-valued functions:

Theorem 1. *The minimizer $\hat{h}$ of the optimization problem* (5) *admits an expansion $\hat{h}(\cdot) = \sum_{j=1}^{\ell+u} \mathcal{K}_x(o_j, \cdot)\mathbf{c}_j$, where the vectors $\mathbf{c}_j \in \mathcal{F}_y, j = \{1, \cdots, (\ell+u)\}$ satisfy the equations:*

$$V_j \mathbf{y}_j = V_j \sum_{i=1}^{\ell+u} \mathcal{K}_x(o_i, o_j)\mathbf{c}_i + \lambda_1 \mathbf{c}_j + 2\lambda_2 \sum_{i=1}^{\ell+u} L_{ij} \sum_{m=1}^{\ell+u} \mathcal{K}_x(o_m, o_i)\mathbf{c}_m \ . \quad (6)$$

The matrix V_j of dimension $dim(\mathcal{F}_y) \times dim(\mathcal{F}_y)$ is the identity matrix if $j \leq \ell$ and the null matrix if $\ell < j \leq (\ell + u)$. L is the $(\ell+u)\times(\ell+u)$ Laplacian matrix, given by $L = D - W$, where D is a diagonal matrix such that $D_{ii} = \sum_{j=1}^{\ell+u} W_{ij}$.

Using the operator-valued kernel defined previously leads us to define a new model, expressed as a closed-form solution.

Proposition 2. *When $\mathcal{K}_x$ is defined by mapping* (3)*, the solution of Problem* (5) *reads*

$$C = Y_\ell U (K_{X_{\ell+u}} U^T U + \lambda_1 I_{\ell+u} + 2\lambda_2 K_{X_{\ell+u}} L)^{-1}, \quad (7)$$

where $Y_\ell = (\mathbf{y}_1, \cdots, \mathbf{y}_\ell)$, $C = (\mathbf{c}_1, \cdots, \mathbf{c}_{\ell+u})$. *$U$ denotes a* $\ell \times (\ell+u)$ *matrix that contains an identity matrix of size* $\ell \times \ell$ *on the left hand side and a zero matrix of size* $\ell \times u$ *on the right hand side.* $K_{X_{\ell+u}}$ *is the Gram matrix of size* $(\ell+u)\times(\ell+u)$ *associated to kernel* κ_x. *Finally,* $I_{\ell+u}$ *is the identity matrix of size* $(\ell+u)$.

3 Experiments

We extensively studied the behavior of the provided models on transductive link prediction using artificial data and a protein-protein interaction network dataset.

Synthetic networks We illustrate our method on synthetic networks in order to measure the improvement brought by the semi-supervised method in extreme cases (i.e. for low percentage of labeled proteins) when the input kernel is a very good approximation of the output kernel. We produce the data by sampling random graphs from a Erdős-Renyi law with different probabilities of presence of edges. The input feature vectors have been obtained by applying Kernel PCA on the diffusion kernel associated with the graph. Finally, we use the components that capture 95% of the variance to define the input features. We observe from the results obtained that the semi-supervised approach improves upon the supervised one on Auc-Roc and Auc-Pr, especially for a small percentage of labeled data (up to 10%). Based on these results one can formulate the hypothesis that supervised link prediction is harder in the case of more dense networks and that the contribution of unlabeled data seems more helpful in this case. One can also assume that using unlabeled data increases the AUCs for low percentage of labeled data. But when enough information can be found in the labeled data, semi-supervised learning does not improve the performance.

Protein-protein interaction network We illustrate our method on a PPI network of the yeast *Saccharomyces Cerevisiae* composed of 984 proteins linked by 2438 interactions. To reconstruct the PPI network, we deal with usual input features that are gene expression data, phylogenetic profiles, protein localization and protein interaction data derived from yeast two-hybrid (see for instance [2–6] for a more complete description).

Table 1. Auc-roc and Auc-pr obtained for the reconstruction of the PPI network from the gene expression data in the supervised and the semi-supervised settings. The percentage values correspond to the proportions of labeled proteins.

Methods	**Auc-roc**			**Auc-pr**		
	5%	**10**%	**20**%	**5**%	**10**%	**20**%
Supervised	76.9 ± 4.3	80.3 ± 0.9	82.1 ± 0.6	5.4 ± 1.6	7.1 ± 1.1	8.1 ± 0.7
Semi-supervised	79.6 ± 0.9	80.7 ± 1.0	81.9 ± 0.7	6.6 ± 1.1	7.6 ± 0.8	8.4 ± 0.5

We experimented with our method in the semi-supervised setting and compared the results with those obtained in the supervised setting. For different

values of ℓ, that is the number of labeled proteins, we randomly sub-sampled a training set of proteins and considered all the remaining proteins for the test set. The interaction assumed to be known are those between two proteins from the training set. We ran each experiment ten times and tuned the hyperparameters by 5-fold cross-validation on the training set. Averaged and standard deviations of the Auc-roc and Auc-pr values when using gene expression data as input features are summarized in Table 1. It is worth noting that the semi-supervised method reaches better performances when the number of labeled proteins is small, which is usually the case in PPI network inference problems.

References

1. Ben-Hur, A., Noble, W.S.: Kernel methods for predicting protein–protein interactions. Bioinformatics, vol. 21, pp. 38–46 (2005)
2. Yamanishi, Y., Vert, J.-P., Kanehisa, M.: Protein network inference from multiple genomic data: a supervised approach. Bioinformatics, vol. 20, pp. 363–370 (2004)
3. Geurts, P., Wehenkel, L., d'Alché-Buc F.: Kernelizing the output of tree-based methods. In: Proc. of the 23th Intl. Conf. on Machine learning (2006)
4. Geurts, P., Touleimat, N., Dutreix, M., d'Alché-Buc, F.: Inferring biological networks with output kernel trees. BMC Bioinformatics, vol. 8 (2007)
5. Bleakley, K., Biau, G., Vert, J.-P.: Supervised reconstruction of biological networks with local models. Bioinformatics, vol. 23, pp. i57–i65 (2007)
6. Kato, T., Tsuda, K., Asai, K.: Selective integration of multiple biological data for supervised network inference. Bioinformatics, vol. 21, pp. 2488–2495 (2005)
7. Tsuda, K., Noble, W. S.: Learning kernels from biological networks by maximizing entropy. Bioinformatics, vol. 20, pp. 326–333 (2004)
8. Kashima, H., Yamanishi, Y., Kato, Ts., Sugiyama, M., Tsuda, K.: Simultaneous inference of biological networks of multiple species from genome-wide data and evolutionary information. Bioinformatics, vol. 25, pp. 2962–2968 (2009)
9. Yip, K. Y., Gerstein, M.: Training set expansion: an approach to improving the reconstruction of biological networks from limited and uneven reliable interactions. Bioinformatics, vol. 25, pp. 243–250 (2009)
10. Kondor, R. I., Lafferty, J. D.: Diffusion Kernels on Graphs and Other Discrete Input Spaces. In: Proc. of the 19th Intl. Conf. on Machine Learning (2002)
11. Zhou, D., Bousquet, O., Lal, T.N., Weston, J., Schölkopf, B.: Learning with Local and Global Consistency. In: Adv. in Neural Information Processing Systems 16 (2004)
12. Belkin, M., Niyogi, P., Sindhwani, V.: Manifold Regularization: A Geometric Framework for Learning from Labeled and Unlabeled Examples. J. Mach. Learn. Res., vol. 7, pp. 2399–2434 (2006)
13. Senkene, E., Tempel'man, A.: Hilbert Spaces of operator-valued functions. Lithuanian Mathematical Journal, vol. 13, pp. 665–670 (1973)
14. Micchelli, C.A., Pontil, M. A.: On Learning Vector-Valued Functions. Neural Computation, vol. 17, pp. 177–204 (2005)
15. Cortes, C.,Mohri, M., Weston, J.: A general regression technique for learning transductions. In: Proc. of the 22nd Intl. Conf. on Machine Learning, pp 153–160 (2005)

A machine-learning approach to hydrogenosomal protein identification in *Trichomonas vaginalis*

David Burstein[1], Sven Gould[2] Verena Zimorski[2], Thorsten Klösges[2], Fuat Kiosse[2], Peter Major[2], William Martin[2], Tal Pupko[1,3], and Tal Dagan[2]

[1] Department of Cell Research and Immunology, George S. Wise Faculty of Life Sciences, Tel Aviv University, Israel
[2] Institute of Botany III,Heinrich-Heine University Düsseldorf, Düsseldorf, Germany
[3] National Evolutionary Synthesis Center (NESCent) 2024 W. Main Street, Durham, NC 27705-4667, U.S.A

Abstract. The protozoan parasite *Trichomonas vaginalis* is the causative agent of the most prevalent non-viral sexually transmitted disease in humans. It possesses mitochondrion-related organelles, known as hydrogenosomes, that produce ATP under anaerobic conditions. To date only 37 hydrogenosomal proteins have been identified. We sought to identify new hydrogenosomal proteins within the 59,672 open reading frames (ORFs) of *T. vaginalis*, using a machine-learning approach. We applied Naïve Bayes, Bayesian networks, SVM classifiers on a training set consisting of all known hydrogenosomal and 576 non-hydrogenosomal proteins. For each ORF 57 features that measure various evolutionary, genomic, and biochemical traits of the proteins were taken into account. Ten high scoring predictions were experimentally validated by *in vivo* localization studies, yielding the identification of six new hydrogenosomal proteins.

1 Introduction

The parabasalian flagellate *Trichomonas vaginalis*, which infects the urogenital tract of hundreds of millions of people annually, can produces ATP under anaerobic conditions due to its hydrogenosomes. Hydrogenosomes are mitochondrion-like organelles that, unlike mitochondria, are devoid of a genome and translation machinery, making the import of hundreds of nuclear-encoded proteins from the cytosol mandatory. Hydrogenosomes share a common ancestor with the mitochondrion but their scattered distribution over the eukaryotic kingdoms (some fungi, parabasalids, amoeboflagellates, ciliates and at least one animal) suggests that the specialization of the organelle to the anaerobic lifestyle occurred several times in independent lineages during evolution [1–4]. Understanding the biochemistry and molecular evolution of hydrogenosomes is not only of medical importance. The evolution of mitochondria is directly linked with the emergence of eukaryotes, and comparative understanding of mitochondria and hydrogenosomes should shed light on the early evolutionary events in the endosymbiotic theory [1, 3, 4].

A first critical step towards studying hydrogenosome functions and furthermore analyzing its proteome in an evolutionary context is to generate the most reliable set of proteins imported into the hydrogenosome. In contrast to mitochondria, little is known about the targeting mechanism, let alone the components assembling the import machinery in hydrogenosomes. Import of precursors is ATP-dependent and early in vitro analyses suggested that correct targeting requires a hydrogenosomal targeting signal (HTS) on the N-terminus of the protein [5].

The genome of *T. vaginalis* contains 59,672 ORFs, 226 of which encode the canonical HTS defined by Carlton and colleagues as follows: 5'ML[S|T|A]X{1-15}R[N|F|E|XF], or 5'MSLX{1-15}R[N|F|XF] or 5'MLR[S|N|F] [5]. This value is significantly lower than the 500 proteins expected to be found in the hydrogenosome [6]. Thus, other protein characteristics, besides an HTS, may serve as potential targeting precursors to the hydrogenosomes. A recent localization of two important hydrogenosomal proteins that lack HTS [7] supports this assumption. Consequently, one can conclude that *T. vaginalis* encodes further hydrogenosomal proteins, which have not been identified due to their lack of a canonical HTS at their N-terminus.

Our study was aimed at identifying so far unrecognized proteins targeted to the hydrogenosome independent of the canonical HTS. We applied various classification tools to perform an unbiased screening of the entire *T. vaginalis* genome for potentially imported proteins. Individual proteins (high and low scoring) were then tested for their localization *in vivo* to verify (or reject) the bioinformatics predictions. Validated imported proteins were subsequently added to the positive learning set in a an additional learning phase to improve predictions. To the best of our knowledge, this is the first attempt to use machine learning tools for a genomic scale function predictions in *T. vaginalis*.

2 Classification

The training set of the first learning phase included the 37 experimentally validated imported proteins and 576 non-imported proteins that were chosen based on their annotation, indicating a strict cytosolic localization. A total of 57 features regarding the gene and protein sequence, protein function, evolutionary relationships, the existence of an import signal, and GO annotation were computed for each protein.

Three types of classifiers were used for the machine learning inference: (a) Naïve Bayes; (b) Bayesian networks with different network structure search algorithms: K2 search algorithm [8], and the tree augmented Bayes network (TAN) search algorithm [9]; (c) SVM with two alternative kernels: polynomial and radial basis function (RBF). These learning schemes were subjected to an inner feature selection procedure in order to identify the subset of the 57 features that performs best with each classifier. The feature selection was performed by applying a "Wrapper" [10] using the BestFirst search algorithm, a greedy hill-

climbing augmented with a backtracking facility. The machine learning analysis was implemented using the open-source package WEKA version 3.7.0 [11].

The performance of all learning schemes was evaluated by the area under the ROC (AUC). Ten-fold cross-validation was performed for choosing the best performing classifier. In order to maintain the independence between the training process and evaluation process, the features selection was performed separately for each one of the ten folds. The best performing classifier went through an additional step of feature selection and training, which was performed on the entire training set. The resulting trained classifier was used to produce the import scores for all *T. vaginalis* ORFs.

The unbalanced frequencies of imported and non-imported proteins included in the learning set (about 1:20), might render an overestimated AUC [12]. In order to provide comparable performance estimates despite the bias, the values of area under precision recall curve (AUPR) were calculated as well, using AUC-Calculator 0.2 [12]. The classification performance of the first phase was high, with AUC value of 0.978 and AUPR value of 0.845.

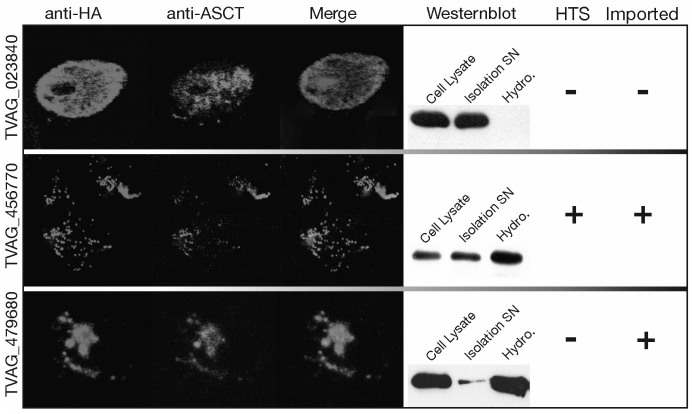


Fig. 1. Results of the *in vivo* localization of TVAG_456770, TVAG_479680 and TVAG_023840, together with the hydrogenosomal marker ASCT

3 Experimental Validations

We selected 14 proteins for experimental validation based on their predicted import scores. Four out of these 14 received low scores, and were predicted not to be localized to the hydrogenosome. The other ten were high-scoring proteins, predicted to be localized to the hydrogenosome. Out of these ten predictions, four include a canonical N-terminal import motif as described previously (Carlton et al. 2007). For the experimental validations, all proteins were tagged using

hemagglutinin (HA) at their C-terminus, and their import into the hydrogenosome was determined by cell subfractionation and subsequent western blot analysis. An additional verification was performed by an in situ immunolocalisation (Fig. 1). Altogether, our predictions were correct in ten of these 14 proteins. All four low scoring predictions were found not to localize to the hydrogenosome (true negative). Out of the ten high scoring predictions, we localized six novel proteins to the hydrogenosomes of *T. vaginalis*, two of which lack the canonical N-terminal HTS (e.g., TVAG_479680 in Fig. 1).

We expect that as more hydrogenosomal proteins are discovered the performances of the machine-learning prediction will improve, as new and more pronounced patterns in the data are likely to emerge. Indeed, when including the six proteins we had localized *in vivo*, the accuracy as calculated by the AUC and AUPR measures has improved (AUC = 0.992; AUPR = 0.956). Our final prediction scores provide fertile grounds for further research of the hydrogenosome and the parasites harbouring it.

References

[1] Müller, M.: The hydrogenosome. J. Gen. Microbiol. 139: 2879–2889 (1993)

[2] Finlay, B.J., Fenchel, T.: Hydrogenosomes in some anaerobic protozoa resemble mitochondria. FEMS. Microbiol. Lett. 65: 311–314 (1989)

[3] Embley, T.M. and Martin, W.: Eukaryotic evolution, changes and challenges. Nature. 440: 623–630 (2006)

[4] Hjort, K., Goldberg, A.V., Tsaousis, A.D., Hirt, R.P., Embley, T.M.: Diversity and reductive evolution of mitochondria among microbial eukaryotes. Philos. Trans. R. Soc. Lond. B. Biol. Sci. 365: 713–727 (2010)

[5] Carlton, J.M., Hirt, R.P., Silva, J.C., Delcher, A.L., Schatz, M., et al.: Draft genome sequence of the sexually transmitted pathogen *Trichomonas vaginalis*. Science. 315: 207–212 (2007)

[6] Shiflett, AM., Johnson, P.J.: Mitochondrion-related organelles in eukaryotic protists. Annu. Rev. Microbiol. 64: 409–429 (2010)

[7] Mentel, M., Zimorski, V., Haferkamp, P., Martin, W., Henze, K.: Protein import into hydrogenosomes of *Trichomonas vaginalis* involves both N-terminal and internal targeting signals: a case study of thioredoxin reductases Eukaryot. Cell. 7: 1750–1757 (2008)

[8] Cooper, G., Herskovits, E.: A Bayesian method for the induction of probabilistic networks from data. Mach. Learn. 9: 309–347 (1992)

[9] Friedman, N., Geiger, D., Goldszmidt, M.: Bayesian network classifiers. Mach. Learn. 29: 131–163 (1997)

[10] Kohavi, R. and John, G.H.: Wrappers for feature subset selection. Artif. Intell. 97: 273–324 (1997)

[11] Witten, I.H. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.(2009) The WEKA data mining software: an update. SIGKDD Explorations. 11: 10–18 (2009)

[12] Davis, J. and Goadrich, M.: The relationship between precision-recall and ROC curves. Proceedings of the 23rd international conference on Machine learning: 233–240 (2006)

An Empirical Analysis of Markov Blanket Filters for Feature Selection on Microarray Data

David Dernoncourt[1], Blaise Hanczar[2], and Jean-Daniel Zucker[1]

[1] INSERM, UMRS 872, Les Cordeliers, Eq. 7 Nutriomique,
15 rue de l'École de médecine, 75006 Paris, France
[2] LIPADE, Université Paris Descartes,
45 rue des Saint-Pères, 75006 Paris, France

Abstract. Feature selection is an important step when building a classifier on microarray data. Over the large number of methods proposed, Markov blanket filters are the *theorically* optimal solution to the feature selection problem. However, existing sound algorithms have not been tested on small sample microarray datasets, which represent the vast majority of microarray datasets. In this study we introduce a modified version of Markov blanket algorithm IPC-MB, then we present results about its performance on four small to medium sample microarray datasets.

Keywords: Markov blanket, variable selection, microarray

1 Introduction

Over the last years, advances in high-throughput technologies have allowed the production of large gene expression datasets. Those data can be used, for instance, to classify tumor types or to predict clinical outcome [1]. A particular characteristic of gene expression data, such as microarray data, is that the number n of samples is very small (usually around 100) compared to the number m of features (usually several thousands). This poses a challenge to classification techniques, since too many features or redundant features can decrease accuracy and increase the risk of overfitting and computation time of classifiers. Feature selection is a solution to this problem, but much work remains to achieve robust and optimal selection of genes.

Feature selection refers to the process of removing irrelevant or redundant features, so as to retain informative features useful for classification. Feature selection methods can be grouped in three categories: filter, wrapper and embedded methods [2]. Wrapper methods interact with the classifier to select a subset of features optimized specifically for this classifier. Embedded methods are embedded within the classifier so that the feature selection process takes place during the classifier's construction. Wrapper and embedded methods are thus computationally intensive and known to present a risk of overfitting. But because they take the classifier into account they can result in good prediction performances if the risk of overfitting is controlled. Filter methods select features independently from the classifier. They are faster than wrapper and embedded

methods, scale better to very high-dimensional datasets, and reduce the risk of overfitting. But the lack of interaction with the classifier may reduce prediction performances.

Filter methods can be further divided into univariate and multivariate methods. Univariate filter methods rank features according to a univariate statistic such as t-test, χ^2, correlation or information gain. Each feature is considered separately, ignoring feature dependencies, which results in great speed and scalability, but may select redundant features. Multivariate filter methods deal with this issue by taking feature dependencies into account. Examples of such methods are Zuber and Strimmer's shrinkage correlation-adjusted t-score (cat score), Golub's neighborhood analysis, correlation-based feature selection (CFS), min redundancy – max relevance (MRMR), and Markov blanket filters [3].

Tsamardinos and Aliferis [4] pointed out that wrappers and filters cannot provide a universally optimal solution to the feature selection problem, and that this theorically optimal solution is, as Koller and Sahami [3] showed, the Markov blanket (MB) of the class to predict T.

2 Markov Blanket Filters and the IPC-MB Algorithm

Let U be a set of features (genes), and $T \notin U$ the target variable of interest (the class to predict). The Markov blanket $MB(T)$ of T is the smallest subset of U that renders T statistically independent from all the remaining features:

$$T \perp S \mid MB(T),\ \forall S \subseteq (U \setminus MB(T))\ . \tag{1}$$

Less formally, the Markov blanket of T is the set containing all and only the features necessary to predict T.

A number of algorithms were proposed to induce $MB(T)$: KS (Koller and Sahami) [3], GS (Grow-Shrink), IAMB (Iterative Associative MB), and variants (Fast-IAMB...), MMPC/MB (Max-Min Parents and Children/MB), HITON-PC/MB and PCMB (parents and children based Markov boundary), but all of these were either too computationally expensive or data-inefficient to be applied on microarray data, or even unsound.

More recently, Fu [5] introduced a sound and efficient algorithm, IPC-MB (Iterative Parents-Children based search of MB) while Aliferis and al. [6] published an algorithmic framework, GLL-PC/MB (Generalized Local Learning - Parents and Children / MB), describing the general steps an algorithm should follow to efficiently and correctly learn Markov blankets, as well as improved versions of HITON-PC/MB following this framework. To our knowledge, as of today, none of those algorithms have been tested on microarray datasets consisting of fewer than 200 samples, which is the case of the vast majority of studies (only 1% of the microarray datasets of the Gene Expression Atlas contain such a high number of samples). The aim of this work is to study how this kind of algorithm performs on such small-sample microarray data. We chose to implement the IPC-MB algorithm because it is well described by Fu [5] and follows the GLL framework.

Like many other MB algorithms, IPC-MB is based on the Bayesian approach of Markov blankets: $MB(T)$ is the union of parents, children and spouses of T [5]. It can be divided into three phases:
1. Identify candidate parents and children (PC_T^C) (function "findCanPC")
2. Remove false positives (descendants farther than children)
3. Identify spouses (co-parents)

We implemented a modified version of IPC-MB, IPC-MBUVR (IPC-MB with uncached variable removal), containing a function findCanPCUVR modified so that features identified as independent are removed immediately after the independence test. In the initial algorithm those features were first cached and then removed in bulk, which would allow two features strongly correlated to each other to mutually eliminate themselves from the candidate parents and children even if they were strongly correlated to the target variable. We also replaced IPC-MB's validity check for the conditional independence test by GLL's *max-k* parameter, which places an absolute limit on the cardinality of conditioning sets in findCanPC.

3 Experimental Evaluation

To evaluate the algorithm, we used four publicly available microarray datasets, related to colon cancer (Alon, $n = 62, m = 2000$), leukemia (Golub, $n = 72, m = 7129$), lung cancer (CAMDA, $n = 203, m = 2000$), and breast cancer (van de Vijver, $n = 295, m = 2000$). Since we used the G-test to assess variable independence, we discretized the variables into binary variables using R package *mclust* before filtering them with IPC-MBUVR. Eleven other feature selection methods were used: t score (as well as Welch and Wilcoxon), cat score, Golub's criterion, componentwise boosting [7], one-step recursive feature elimination (RFE) [8], lasso [9], elastic net [10], and a correlation-based score and a fold change score implemented in R package *SlimPLS* [11]. Performance of classifiers was used as the measure of performance of feature selection algorithms. Six different general purpose classifiers were used: LDA (linear discriminant analysis), DDA (diagonal DA), kNN (k-nearest neighbors), SVM (support vector machine, with linear and radial kernel) and random forest. Classification performance was estimated by a 10 fold cross-validation. Although we did not perform an internal cross-validation to optimize filter parameters, we tried a number of different parameter values and retained, as the final result, the 20^{th} percentile of the average performance over the classifiers (30^{th} percentile for the Markov blanket filter).

Table 1 shows part of our experimental results. The Markov blanket filter based on IPC-MBUVR performed well on microarray data with a bit more than 200 observations. This is consistent with the results previously obtained by Aliferis and al. [6] on a number of real and artificial datasets of such a sample size. However, when applied to microarray data consisting of a smaller sample, the Markov blanket filter eliminated too many variables to remain efficient.

IPC-MBUVR's performance depends on the reliability of the multiple conditional independence (CI) tests it performs. On datasets containing too few

Table 1. Error rate (%) of classifiers depending on filters and datasets. *LDA*: linear discriminant analysis; *SVM*: support vector machine with radial kernel; *kNN*: k-nearest neighbors; *Avg*: average on the three classifiers; *IPC-MBUVR* and *findCanPCUVR*: our algorithms; dark background: best four average error rates for each dataset.

Filter	Colon cancer				Leukemia				Breast cancer				Lung cancer			
	LDA	SVM	kNN	Avg	LDA	SVM	kNN	Avg	LDA	SVM	kNN	Avg	LDA	SVM	kNN	Avg
findCanPCUVR	18.1	16.4	24.3	19.6	7.1	7.1	7.0	**7.1**	33.6	32.6	39.0	**35.1**	15.0	16.0	18.4	**16.4**
IPC-MBUVR	25.7	22.9	30.7	26.4	8.6	8.6	7.0	8.0	33.9	34.6	35.6	**34.7**	17.5	17.5	20.4	**18.4**
lasso	11.7	13.1	16.4	**13.7**	7.1	7.1	8.4	7.5	37.7	35.0	36.0	36.3	17.4	15.9	25.4	**19.6**
cat score	13.1	16.2	17.9	15.7	5.7	5.7	7.1	**6.2**	36.3	33.9	38.0	36.1	22.4	16.4	22.4	20.4
Welch	17.9	19.5	14.5	17.3	5.7	2.9	4.1	**4.2**	38.7	35.0	36.0	36.6	17.9	22.9	21.9	20.9
t-score	17.9	18.1	16.4	17.5	8.6	5.7	8.6	7.6	35.6	33.3	37.0	**35.3**	20.4	16.5	20.4	**19.1**
Golub	11.7	16.4	14.5	**14.2**	4.3	5.7	7.1	**5.7**	37.3	36.0	36.0	36.4	29.9	25.4	25.3	26.9
RFE	12.9	13.1	14.3	**13.4**	8.6	7.1	8.6	8.1	35.3	34.3	32.9	**34.1**	31.9	28.4	24.4	28.2
elastic net	11.7	16.2	17.9	**15.2**	11.4	12.7	12.7	12.3	36.6	34.3	37.7	36.2	20.9	21.4	21.8	21.4
fold-change	14.5	19.1	23.8	19.1	16.8	15.5	23.8	18.7	37.3	33.2	35.9	35.5	22.0	29.5	16.4	22.6

observations, the conditioning set (CS) has to remain small to preserve CI tests reliability, and it is not possible anymore to find a proper compromise between a CS small enough for a good CI test reliability and a CS large enough for a good IPC-MBUVR accuracy. FindCanPCUVR alone performs better on such datasets, probably because it performs far fewer (possibly inaccurate) CI tests and because IPC-MBUVR detects spouses with CI tests based on CS of greater cardinality (thus less reliable).

4 Conclusion

In this work we explored the potential of a sound and efficient Markov blanket filter for use on small sample microarray datasets. Our results suggest that although the Markov blanket filter is very efficient to filter microarray features given a large enough sample, it is not able to compensate the information loss caused by discretization when the sample size is too small. It is also penalized by the decreased reliability of conditional independence tests in such a setting. We are currently trying to improve IPC-MBUVR's performance on small samples by replacing the G-test with another CI test which wouldn't require a complete discretization of data. As future work, we would also like to explore more precisely the minimum sample size necessary for the Markov blanket filter to perform well.

References

1. MAQC Consortium: The MicroArray Quality Control (MAQC)-II study of common practices for the development and validation of microarray-based predictive models. Nat Biotech, 28, 827–838 (2010)
2. Saeys, Y., Inza, I., Larrañaga, P.: A review of feature selection techniques in bioinformatics. Bioinformatics 23, 2507-2517 (2007)
3. Koller, D., Sahami, M.: Toward Optimal Feature Selection. Technical Report, Stanford InfoLab (1996)
4. Tsamardinos, I., Aliferis, C. F.: Towards principled feature selection: relevancy, filters and wrappers. In: AISTATS 2003

5. Fu, S.: Efficient Learning of Markov Blanket and Markov Blanket Classifier. Thèse de doctorat, École Polytechnique de Montréal, Canada (2010)
6. Aliferis, C.F., Statnikov, A., Tsamardinos, I., Mani, S., Koutsoukos, X.D.: Local Causal and Markov Blanket Induction for Causal Discovery and Feature Selection for Classification Part I: Algorithms and Empirical Evaluation. J. Mach. Learn. Res. 11, 171–234 (2010)
7. Bühlmann, P., Yu, B.: Boosting with the L2 loss: Regression and Classification. J. Amer. Statist. Assoc. 98, 324–339 (2003)
8. Guyon, I., Weston, J., Barnhill, S., Vapnik, V.: Gene selection for cancer classi cation using support vector machines. Mach. Learn. 46, 389–422 (2002)
9. Friedman, J.H., Hastie, T., Tibshirani, R.: Regularization paths for generalized linear models via coordinate descent. J. Statist. Software 33, 1–22 (2010)
10. Zou, H., Hastie, T.: Regularization and variable selection via the elastic net. J. Royal Statist. Soc. 67, 301–320 (2005)
11. Gutkin, M., Lavi, O.: SlimPLS: SlimPLS multivariate feature selection, `http://cran.r-project.org/web/packages/SlimPLS`

Exploring signature multiplicity in microarray data using ensembles of randomized trees

Pierre Geurts[1] and Yvan Saeys[2]

[1] Department of EE and CS & GIGA-R, University of Liège, Belgium
[2] VIB-Ghent University, Ghent, Belgium
p.geurts@ulg.ac.be,yvan.saeys@ugent.be

Abstract. A challenging and novel direction for feature selection research in computational biology is the analysis of signature multiplicity. In this work, we propose to investigate the effect of signature multiplicity on feature importance scores derived from tree-based ensemble methods. We show that looking at individual tree rankings in an ensemble could highlight the existence of multiple signatures and we propose a simple post-processing method based on clustering that can return smaller signatures with better predictive performance than signatures derived from the global tree ranking at almost no additional cost.

1 Introduction

Feature selection is an important aspect of many machine learning applications in computational biology [4]. Traditionally, many standard feature selection algorithms assume the existence of a single set of "optimal" features. However, in reality, this need not necessarily be the case and there could be several, distinct or overlapping, (minimal) subsets of features that might all explain the output of interest equally well given a particular loss function. We will refer to these equivalent minimal subsets as *signatures*, and the occurence of multiple signatures as *signature multiplicity* [6]. This phenomenon arises naturally in the presence of correlated or redundant features on a pairwise basis, but multiplicity can also occur at the level of signatures of larger sizes. For some loss function, signature multiplicity can be related to the existence of multiple markov boundaries for the target variable [6]. The study of signature multiplicity, and its effect on feature selection is at the moment only in its childhood, and so far studies have mainly focused on the microarray domain [1,6].

As standard feature ranking methods are not designed to cope with multiple signatures, they often interleave the features from the different signatures. Thus, thresholding this ranking does not even ensure to give a single valid and/or minimal signature. Furthermore, signature multiplicity might have a detrimental effect on the stability of feature selection methods, as small perturbations on the training set can result in large deviations regarding the ranking of features.

In this work, we investigate the impact of signature multiplicity on tree-based ensemble methods and we propose a simple post-processing method based on clustering to retrieve multiple signatures from the individual rankings provided by individual trees in a randomized tree ensemble.

2 Exploring individual tree rankings

Classification and regression trees are non-parametric supervised learning methods that learn an input-output model in the form of a tree, combining elementary tests defined on the input features. Because of their high variance, they are typically exploited in the context of ensemble methods such as bagging or random forests. A feature importance measure can be derived in different ways from a tree. In this work, we restrict ourselves to the importance obtained by summing the impurity reduction score at each tree node where this feature is used to split[3]. These importance scores are then averaged over several trees to yield a more stable score.

While one is often interested only in the global ranking obtained by averaging the individual rankings, in the presence of multiple signatures, one can reasonably expect that each tree in an ensemble will highlight a distinct signature. Indeed, since each tree is built greedily in a top-down fashion, the selection of a feature, or group of features, in a tree branch will decrease the probability to select redundant features at deeper nodes, which will favor the appearance of features from only one signature in each tree. In addition, because of randomization, one can also expect the selected signature to be different from one tree to another.

To check this hypothesis, we carried out experiments on the TIED dataset, an artificial dataset, specifically designed to contain multiple signatures [5]. The TIED dataset was generated from a bayesian network containing 1000 discrete variables, including the four-valued target. By construction, each of the 72 signatures contains 5 variables and belongs to $\{9\} \times \{4, 8\} \times \{11, 12, 13\} \times \{18, 19, 20\} \times \{1, 2, 3, 10\}$. The upper left graph of Figure 1 shows a heatmap representing 1000 tree rankings obtained with bagging (x-axis) for the top 20 features (y-axis) in the global ranking. Features are ranked top-down according to their global importances and rankings have been ordered by hierarchical clustering (dendrogram not shown). This heatmap clearly highlights the existence of groups of rankings each corresponding to one of the signatures. While the global ranking introduces the redundant features by block (e.g., features 1,2,3, and 10 are the top 4 features which are redundant by construction), each individual ranking usually contains only one feature per group. We obtained similar results on other artificial datasets.

3 Towards an automatic identification of signatures

Assuming that we are looking for K signatures, the analysis in the previous section suggests a simple approach for retrieving the multiple signatures from T feature importance vectors; Use any clustering algorithm (k-means in our experiments) to determine K clusters of weight vectors. Then, average the weight vectors in each of the clusters, and rank the features according to their average weight. To evaluate the quality of a given signature, a model is rebuilt with any

[3] A feature not appearing in a tree receives a zero importance.

supervised learning method using the top m features in each cluster for increasing values of m. When there are multiple signatures, we expect that the model obtained from each cluster will be at least equally good as a model learned in the same manner from the global ranking, i.e. the ranking obtained by averaging over all trees, and not over the clustered ones. To determine the optimal value of the number of clusters, we propose to proceed as follows: several values of K are compared, and the one that maximizes the difference over all values of m between the error obtained from the global ranking and the average error over the clusters is considered as optimal.

We carried out experiments with this approach on the TIED dataset. T was fixed to 1000, and the values explored for K were $\{2, 3, 5, 10, 15\}$. Features were ranked using a bagged ensemble of trees and the evaluation was done using ensembles of (100) totally randomized trees [2]. The latter method is not robust to the introduction of irrelevant features and is thus appropriate to determine minimal signatures. For the evaluation of signatures, we used 20 repetitions of a 90%-10% split of data in training and test, with the feature ranking computed only on the training sample, so as to avoid any selection bias.

The bottom left graph of Figure 1 shows in red the evolution of the error with the number of features m taken in their order in the global ensemble ranking, and in green the average error over all cluster rankings, for the value of $K = 15$ selected as just described. Blue curves show for each value of m respectively the minimal and maximal error obtained over all clusters. This graph shows that the cluster signatures are all very good and much better than the global signature for small values of m.

4 Experiments with microarray data

We have applied the same approach on several microarray datasets related to two families of problems: biomarker discovery for disease classification and regulatory network inference [3]. We only report below the results obtained on one representative problem. The graphs on the right in Figure 1 were obtained from microarray data when trying to discover the regulators of gene `tyrP` of *E. coli* using the same procedure and dataset as in [3]. The protocol was exactly the same as for the experiments on the TIED dataset.

The heatmap clearly highlights the diversity and complexity of the signatures, with for example the top feature from the global ranking not being used in many single rankings. The optimal number of clusters as determined automatically is here 5 and it leads to five signatures that are all (slightly) better than the global one.

5 Conclusion and future works

The discovery of multiple signatures is a challenging topic in the context of feature selection. In this work, we investigate the effect of signature multiplicity on tree-based feature rankings. We show that looking at individual tree rankings in an ensemble could highlight the existence of multiple signatures and we propose

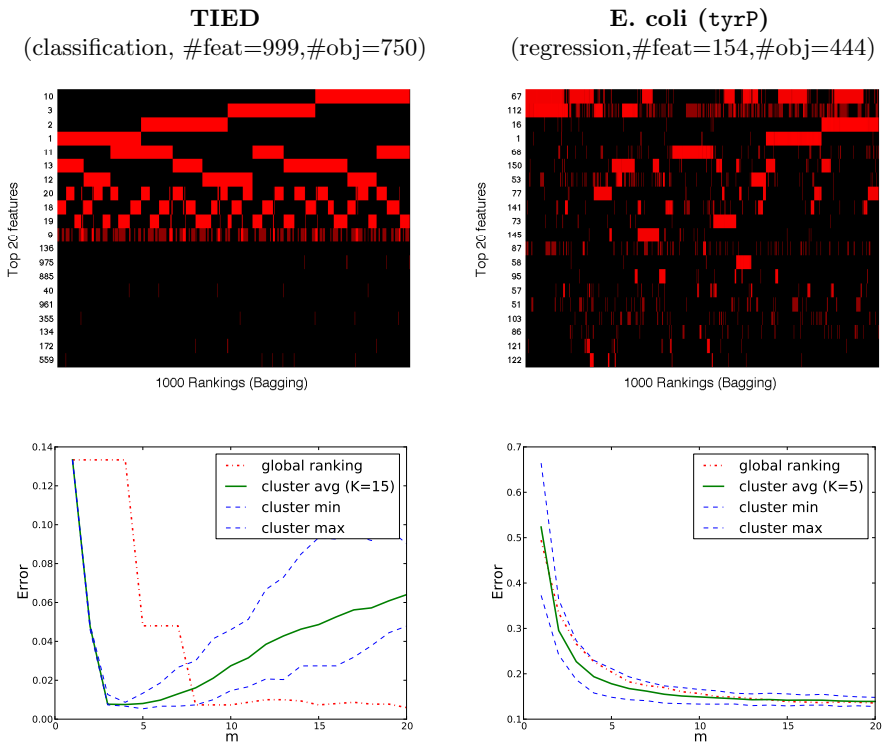


Fig. 1. Results on artificial and real datasets

a simple post-processing method based on clustering that can return smaller signatures with better predictive performance than signatures derived from the global tree ranking at almost no additional cost. In future work, we would like to explore alternative ways to extract multiple signatures from an ensemble of randomized feature rankers (not restricted to trees) and determine a measure of the multiplicity in a given dataset.

Acknowledgments

Pierre Geurts is a research associate with FNRS and Yvan Saeys is a postdoctoral fellow with FWO. This work is partially supported by the Interuniversity Attraction Poles Programme (IAP P6/25 BIOMAGNET), initiated by the Belgian State, Science Policy Office and by the European Network of Excellence, PASCAL2.

References

1. L. Ein-Dor, I. Kela, G. Getz, D. Givol, and E. Domany. Outcome signature genes in breast cancer: is there a unique set? *Bioinformatics*, 21(2):171, 2005.

2. P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomized trees. *Machine Learning*, 36(1):3–42, 2006.
3. V. A. Huynh-Thu, A. Irrthum, L. Wehenkel, and P. Geurts. Inferring regulatory networks from expression data using tree-based methods. *Plos ONE*, 5(9):e12776, sept 2010.
4. Y. Saeys, I. Inza, and P. Larrañaga. A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23(19):2507–2517, 2007.
5. A. Statnikov and C. Aliferis. Tied: An artificially simulated dataset with multiple markov boundaries. *Journal of Machine Learning Research Workshop Conference & Proceedings*, 2009.
6. A. Statnikov and C. Aliferis. Analysis and computational dissection of molecular signature multiplicity. *Plos Comput Biol*, 6(5):e1000790, May 2010.

An exact empirical Bayes approach for incorporating biological knowledge into network inference

Steven M. Hill[1,2] and Sach Mukherjee[2,1]

[1] Centre for Complexity Science, University of Warwick, UK
[2] Department of Statistics, University of Warwick, UK

1 Introduction

The elucidation of the topology of molecular networks remains a challenging problem in systems biology and continues to be an active area of research. Dynamic Bayesian networks (DBNs) [1] are probabilistic graphical models describing conditional independence relationships between variables, through time. DBNs have previously been employed to infer gene regulatory networks (GRNs) from time series data [2, 3].

A wealth of information regarding the structure of molecular networks is widely available due to ongoing development of online tools and databases. Such information can be a valuable resource when used to aid the inference process. Following [4], we carry out network inference within a Bayesian framework, thus enabling the incorporation of existing biology via an informative prior distribution on networks [3, 5]. The key question that we address here is how to weight such prior knowledge against experimental data.

We propose an empirical Bayes approach to determine the weighting of existing biology objectively. Moreover, we use exact model averaging to both perform empirical Bayes and calculate exact posterior edge scores. This results in a procedure which is simple from the user perspective, requiring no user-set parameters or MCMC convergence diagnostics. We also note that we use a continuous linear model with interaction terms. This avoids (lossy) data discretisation whilst retaining the ability to capture combinatorial interplay.

We apply our method to simulated data and to cancer protein signalling data. Inferred networks are used to generate testable hypotheses which can subsequently be validated in further experiments. Whilst we carry out this process in the setting of protein signalling networks, our proposed method can equally be applied to other applications (such as GRNs).

2 Methods

2.1 Model

Let p denote number of variables under study and T number of time points in the dataset. Let X_i^t be a random variable representing variable i at time

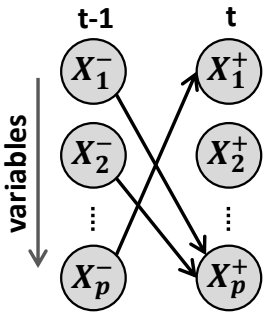


Fig. 1. Schematic of a DBN.

point t and $X^t = \left(X_1^t, \ldots, X_p^t\right)$ be the corresponding random vector at time t. Following previous authors we make first-order Markov and stationarity assumptions [6, 2] and permit only edges forward in time [2, 3]. Then the DBN consists of a graph G with two vertices for each variable, representing adjacent time points, with associated random vectors denoted by $X_i^- = \left(X_i^1, \ldots, X_i^{T-1}\right)^\mathsf{T}$ and $X_i^+ = \left(X_i^2, \ldots, X_i^T\right)^\mathsf{T}$ (see Fig. 1). The edge structure describes conditional independences between variables and is the object of inference here.

Under these assumptions we have the following likelihood:

$$p(\mathbf{X} \mid G, \{\theta_i\}) = \prod_{i=1}^{p} \prod_{t=2}^{T} p(X_i^t \mid X_{\pi_G(i)}^{t-1}, \theta_i) \tag{1}$$

where $\mathbf{X}$ denotes the complete data, $\pi_G(i) \subseteq \{1, \ldots, p\}$ is the set of parents of variable i in graph G, $X_{\pi_G(i)}^t = \left\{X_j^t \mid j \in \pi_G(i)\right\}$ is a corresponding data vector and $\{\theta_i\}$ are parameters associated with the conditional distributions.

The conditionals constituting the likelihood (1) are taken to be Gaussian with mean given by a linear combination of parents and all possible products of parents. Then

$$p(X_i^+ \mid X_{\pi_G(i)}^-, \theta_i) = \mathcal{N}(\mathbf{B}_i \beta_i, \sigma_i^2 I_n) \tag{2}$$

where $\mathbf{B}_i$ is a design matrix (with columns corresponding to parents of i and products of parents) and n is sample size. If data consists of m time courses each with T time points, then $n = m(T-1)$.

Following [7], we use the reference prior $p(\sigma_i^2) \propto \sigma_i^{-2}$ for variances and take $\beta_i \sim \text{Normal}(\mathbf{0}, n\sigma_i^2(\mathbf{B}_i{}^\mathsf{T}\mathbf{B}_i)^{-1})$. Then, integrating out parameters gives the closed-form marginal likelihood,

$$p(\mathbf{X}|G) \propto \prod_{i=1}^{p} (1+n)^{-(2^{|\pi_G(i)|}-1)/2} \left(X_i^{+\mathsf{T}} X_i^+ - \frac{n}{n+1} X_i^{+\mathsf{T}} \mathbf{B}_i \left(\mathbf{B}_i{}^\mathsf{T}\mathbf{B}_i\right)^{-1} \mathbf{B}_i{}^\mathsf{T} X_i^+\right)^{-\frac{n}{2}}$$

This formulation has attractive invariance properties under rescaling of the data and has no free user-set parameters.

2.2 Exact Inference

We are interested in calculating posterior probabilities of edges $e = (X_a^-, X_b^+)$ in the graph G. The posterior probability of the edge is calculated by averaging over the space of all possible graphs $\mathcal{G}$ [4],

$$P(e \mid \mathbf{X}) = \sum_{G \in \mathcal{G}} \mathbb{1}_{\{e \in G\}} P(G \mid \mathbf{X}). \tag{3}$$

where $P(G \mid \mathbf{X})$ is the posterior distribution over graphs and is given, up to proportionality, by $P(\mathbf{X} \mid G)P(G)$, where $P(\mathbf{X} \mid G)$ is the marginal likelihood above and $P(G)$ is a prior probability distribution on graph structures G.

For DBNs with p variables the size of the graph space is 2^{p^2}, hence growing super-exponentially with p. This precludes explicit enumeration of the sum in (3) for even small-to-moderate p. However, for the DBNs used here, it is possible to utilise a variable selection approach to efficiently calculate posterior edge probabilities exactly, thereby increasing confidence in results whilst avoiding the need for expensive convergence diagnostics. In particular, for each variable i, we score subsets of potential parents $\pi(i) \subseteq \{1, \ldots, p\}$. Model averaging is then carried out by averaging over subsets of parents rather than over full graphs,

$$P(e \mid X^-, X_b^+) = \sum_{\pi(b)} \mathbb{1}_{\{a \in \pi(b)\}} P(\pi(b) \mid X^-, X_b^+). \tag{4}$$

If the network prior $P(G)$ factorises into a product of local priors over parents sets for each variable $P(\pi_G(i))$, then posterior edge probabilities calculated by averaging over parent sets (4) equal those calculated by averaging over the (much larger) space of graphs (3). We note that, whilst this equivalence holds for edge probabilities, it does not hold for arbitrary graph features. Following previous work [2, 3], we restrict maximum number of parents to four. The size of the space of parent sets then becomes polynomial in p, enabling exact calculation of posterior edge probabilities via (4).

2.3 Network priors and empirical Bayes

We follow [5] and use a prior of the form $P(G) \propto \exp(\lambda f(G))$, where λ is a parameter controlling the strength of the prior. Let E^* be a set of *a priori* expected edges, generated from existing domain knowledge. Then $f(G) = -|E(G) \backslash E^*|$ where $E(G)$ is the set of edges contained in G and $|\cdot|$ denotes set cardinality. Therefore the prior does not actively promote any particular edge, but rather penalizes graphs according to the number of edges that are not contained in the prior graph.

The strength parameter λ is set using an objective, empirical Bayes approach by empirically maximising the quantity $p(\mathbf{X}|\lambda) = \mathbb{E}\left[p(\mathbf{X} \mid G)\right]_{P(G \mid \lambda)}$. This quantity can be calculated exactly using the variable selection framework described above.

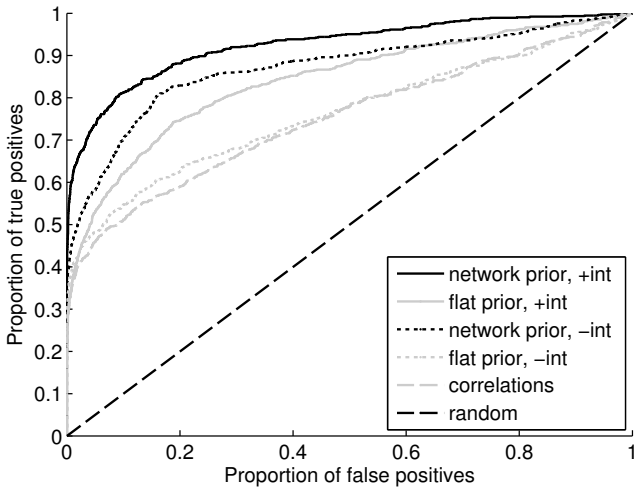


Fig. 2. Average ROC curves for simulation study. Legend - "network prior": DBN inference using an informative prior $P(G)$, weighted objectively by empirical Bayes; "flat prior": DBN inference using a flat prior over graph space; "+int/-int": with/without interaction terms; "correlations": absolute correlations between proteins at adjacent time points.

3 Results

3.1 Simulation

A simulation study was carried out using 20 variables, 8 time points and 4 time courses per variable. We first created 25 data-generating networks, using a random, Erdös-Renyi-like approach configured to ensure that the networks differed substantially from the prior graph used. Data were generated by ancestral sampling, using a Gaussian model, with individual terms in the model included with probability 0.5; this meant that some dependencies were strictly linear, whilst others included interactions. Empirical Bayes setting of prior strength parameter resulted in an average value of $\lambda = 3.54 \pm 0.34$ over the 25 experiments. Fig. 2 shows average ROC curves for edges called at varying thresholds. We see that an informative prior, weighted by empirical Bayes, provides significant gains in sensitivity and specificity, even though a non-trivial proportion of information in the prior is not in agreement with the data-generating model.

3.2 Cancer signalling

We have applied our method to two (unpublished) phospho-proteomic datasets from different breast cancer cell lines. Both cell lines are of the same basal subtype. As in our simulation, the datasets consisted of 20 variables, 8 time points and 4 time courses per variable. A prior network was formed from the literature

and weighted using empirical Bayes. We found evidence of striking differences in signaling network topology even though the cancers are closely related. We predicted a number of novel signaling links, specific to the individual cancer cell lines, which we are currently validating using independent experiments.

4 Conclusion

We have proposed an exact network inference method, free of user-set parameters, that integrates existing biology using informative network priors, objectively weighted by an empirical Bayes approach. We have illustrated our method on simulated data and applied it to protein signalling data. We note that whilst ODEs are bio-chemically more realistic, the continuous linear model used here leads to a computationally efficient procedure, allowing efficient exploration of large graph spaces.

References

1. Murphy K.P.: Dynamic Bayesian networks: representation, inference and learning. PhD thesis, Computer Science, University of California, Berkeley, CA (2002)
2. Husmeier D.: Sensitivity and specificity of inferring genetic regulatory interactions from microarray experiments with dynamic Bayesian networks. Bioinformatics 19:2271-2282 (2003)
3. Werhli A.V., Husmeier D.: Reconstructing gene regulatory networks with Bayesian networks by combining expression data with multiple sources of prior knowledge. Stat Appl Genet Mol Biol 6:15 (2007)
4. Madigan D., York J., Allard D.: Bayesian graphical models for discrete data. Int Stat Rev 63:215-232 (1995)
5. Mukherjee S., Speed T.P.: Network inference using informative priors. Proc Natl Acad Sci USA 105:14313-14318 (2008)
6. Friedman N., Murphy K., Russell S.: Learning the structure of dynamic probabilistic networks. In Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence (Morgan Kaufmann, San Francisco, CA), pp. 139-147 (1998)
7. Smith M., Kohn R.: Nonparametric regression using Bayesian variable selection. J Econometrics 75:317-343 (1996)

A Simple Approach to Ranking Differentially Expressed Gene Expression Time Courses through Gaussian Process Regression

Alfredo A. Kalaitzis and Neil D. Lawrence

The Sheffield Institute for Translational Neuroscience, University of Sheffield,
385A Glossop Road, Sheffield, S10 2HQ, United Kingdom
{A.Kalaitzis,N.Lawrence}@sheffield.ac.uk

Abstract. Two basic forms of analysis recur for gene expression time series: removing inactive (quiet) genes from the study and determining which genes are differentially expressed. Often these analysis stages are applied disregarding the fact that the data is drawn from a time series. In this paper we propose a simple model for accounting for the underlying temporal nature of the data based on a Gaussian process. We present [6] a simple approach for filtering quiet genes, or for the case of time series in the form of expression ratios, quantifying differential expression. We assess via ROC curves the rankings produced by our regression framework and compare them to a recently proposed hierarchical Bayesian model for the analysis of gene expression time-series (BATS). We compare on both simulated and experimental data showing that the proposed approach considerably outperforms the current state of the art.

1 Introduction

Gene expression profiles give a snapshot of mRNA concentration levels as encoded by the genes of an organism under given experimental conditions. With the decreasing cost of gene expression microarrays, long time-series experiments have become commonplace, giving a far broader picture of the gene regulation process. Such time series are often irregularly sampled and may involve differing numbers of replicates at each time point. The experimental conditions under which gene expression measurements are taken cannot be perfectly controlled leading the signals of interest to be corrupted by noise, either of biological origin or arising through the measurement process.

As opposed to methods targeted at static experiments (one timepoint), it would seem sensible to consider methods that can account for the special nature of time course data [1, 10, 11]. The analysis of gene expression microarray time-series has benefited the genome-wide identification of direct targets of transcription factors [4, 5] and the full reconstruction of gene regulatory networks [2]. A comprehensive review on the motivations and methods of analysis of time-course gene expression data can be found in [3].

1.1 Testing for Differential Expression

A primary stage of analysis is to characterize the activity of each gene in an experiment. Removing inactive or *quiet* genes (genes which show negligible changes in mRNA concentration levels in response to treatments) allows the focus to dwell on genes that have responded to treatment. Removing quiet genes will often have benign effects later in the processing pipeline. However, mistaken removal of profiles can clearly compromise any further downstream analysis. If the temporal nature of the data is ignored, our ability to detect such phenomena can be severely compromised.

This paper, as many other studies, uses data from a *one-sample* setup [1], in which the *control* and *treatment* cases are directly hybridized on a microarray and the measurements are normalized log fold-changes between the two output channels of the microarray, so the goal is to test for a non-zero signal.

A recent significant contribution in regards to the estimation and ranking of differential expression of time-series in a *one-sample* setup is a hierarchical Bayesian model for the analysis of gene expression time-series (BATS) [1], which offers fast computations through exact equations of Bayesian inference, while making a number of prior biological assumptions to accomplish this. In BATS each time-course profile is assumed to be generated from an underlying function, which is expanded on an orthonormal basis (Legendre or Fourier), plus noise. The number of bases and their coefficients, are estimated through analytic computations in a fully Bayesian manner. Thus the global estimand for every gene expression trajectory is the linear combination of some number of bases whose coefficients are estimated by a posterior distribution. In addition, the BATS framework allows for various types of non-Gaussian noise models to be used.

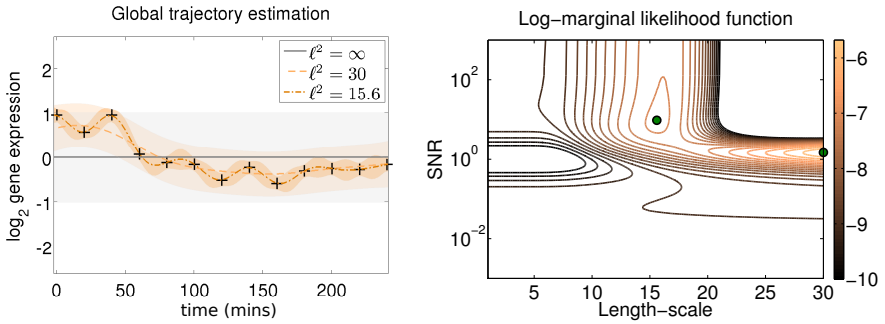
1.2 Gene Expression Analysis with Gaussian Processes

Gaussian processes(GP) [8] offer an easy to implement approach to quantifying the true signal and noise embedded in a gene expression time-series, and thus allow us to rank the differential expression of the gene profile. In this paper we use the *squared-exponential* covariance function (or RBF kernel). Figure 1 illustrates an example of fitting a GP with an RBF kernel on an experimental profile.

When using different types of models (e.g. with different number of hyperparameters), a Bayesian-standard way of comparing them is through Bayes factors [1, 9]

$$K = \frac{\int \mathrm{d}\boldsymbol{\theta}_1 \; p(\mathbf{y} \,|\, \mathbf{x},\, \boldsymbol{\theta}_1,\, \mathcal{H}_1)\, p(\boldsymbol{\theta}_1 \,|\, \mathcal{H}_1)}{\int \mathrm{d}\boldsymbol{\theta}_2 \; p(\mathbf{y} \,|\, \mathbf{x},\, \boldsymbol{\theta}_2,\, \mathcal{H}_2)\, p(\boldsymbol{\theta}_2 \,|\, \mathcal{H}_2)},$$

where $\mathcal{H}_1$ represents the hypothesis where the profile has a significant underlying signal and thus it is truly differentially expressed, and for $\mathcal{H}_2$ is for no underlying signal in the profile where the observed gene expression is just the effect of random noise.



(a) (b)

Fig. 1. (a) A GP fitted on the centred profile of the gene *Cyp1b1* (probeID 1416612_at in the *GSE10562* dataset) with different settings of the lengthscale hyperparameter ℓ^2. Crosses are zero-mean hybridised gene expression in time (log2 ratios between treatment and control), lines are mean predictions of the GP and shaded areas are the point-wise mean plus/minus two standard deviations (95% confidence region). When the mean function is constant as $\ell^2 \to \infty$ (0 inverse lengthscale) then all of the observed variance is attributed to noise (σ_n^2). When the lengthscale is set to a local-optimum large value ($\ell^2 = 30$), the mean function roughly fits the data-points and the observed variance is equally explained by signal (σ_f^2) and noise (σ_n^2). Additionally, the GP has a high uncertainty in its predictive curve. When the lengthscale is set to a local-optimum small value ($\ell^2 = 15.6$) then the mean function tightly fits the data-points with high certainty. The interpretation from the covariance function in this case is that the profile contains a minimal amount of noise and that most of the observed data variance is explained by the underlying signal (σ_f^2). **(b)** The contour of the corresponding LML function plotted through an exhaustive search of ℓ^2 and signal-to-noise-ratio (SNR) values. The two main local-optima are indicated by green dots and a third local optimum, that corresponds to the constant zero function, has a virtually flat vicinity in the contour, which encompasses the whole lengthscale axis for very small values of SNR (i.e. the lengthscale is trivial if SNR≈ 0).

Depending on the model $\mathcal{H}$, these integrals may be intractable. In this paper we present a simple approach to ranking the differential expression of a profile. Instead, we approximate the Bayes factor with a log-ratio of marginal likelihoods

$$K \approx \log\left(\frac{p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta}_2)}{p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta}_1)}\right),$$

with each likelihood being a function of different configurations of $\boldsymbol{\theta}$ — the hyperparameters of the RBF kernel. We still maintain hypotheses $\mathcal{H}_1$ and $\mathcal{H}_2$ that represent the same notions explained above, but in this case they differ simply by configurations of $\boldsymbol{\theta}$. Specifically, on $\mathcal{H}_2$ the hyperparameters are *fixed* to $\boldsymbol{\theta}_2 = (\infty, 0, \mathrm{var}(\mathbf{y}))^\top$ to encode a function constant in time ($\ell^2 \to \infty$), with no underlying signal ($\sigma_f^2 = 0$), which generates a time-series with a variance that can be solely explained by noise ($\sigma_n^2 = \mathrm{var}(\mathbf{y})$). Similarly, on $\mathcal{H}_1$ the hyperparameters $\boldsymbol{\theta}_1$ are *initialised* to encode a function that fluctuates in accordance

to a typical significant profile (e.g. $\ell^2 = 20$), with a distinct signal variance that solely explains the observed time-series variance ($\sigma_f^2 = \text{var}(\mathbf{y})$) and with no noise ($\sigma_n^2 = 0$). The log-marginal is then *optimised*, through *scaled conjugate gradients*, with respect to the hyperparameters. The ranking score of a profile is based on how likely $\mathcal{H}_1$ is in comparison to $\mathcal{H}_2$. This methodology is applied on every expression profile in our datasets.

A Gaussian process with an RBF kernel makes the reasonable assumption that the underlying signal in a profile is a *smooth* (infinitely differentiable) function. This property endows the GP with a large degree of flexibility in capturing the underlying signals without imposing strong modeling assumptions (e.g. number of basis functions in BATS) but may also allow it to erroneously pick up spurious patterns (false positives). For a GP approach on *two-sample* data (separate time-course profiles for each treatment), see the work in [9]. GP priors have also been used for modeling transcriptional regulation [7].

2 Results and Conclusions

We assume that each gene expression profile can be categorized as either quiet or differentially expressed. As a noisy ground truth, we use data from [4]. For that study, the TSNI algorithm (time-series network identification) was developed to infer the direct targets of TRP63. Furthermore, the direct targets inferred were biologically confirmed by correlation with ChIP-Seq binding regions.

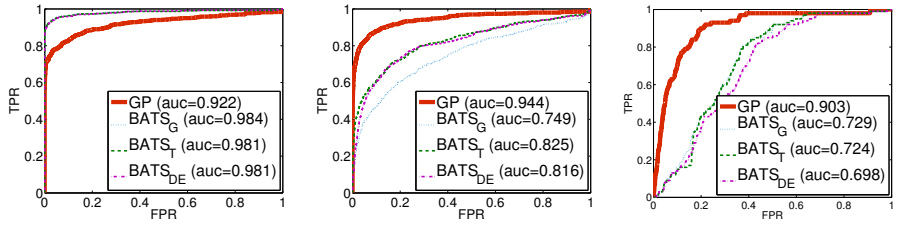
We apply standard GP regression and BATS on two in-silico datasets simulated by BATS and GPs (see Figures 2(a)(b)) and on the experimental data[1], where only the top 100 ranks of TSNI were labelled as *truly* differentially expressed in the ground truth (see Figure 2(c)). From the output of each model a ranking of differential expression is produced and assessed with ROC curves to quantify how well in accordance to the ground truth (BATS-sampled, GP-sampled, experimental) the method performs.

The experimental data are much more complex and apparently the robust-noise BATS variants now offer no increase in performance. Since the ground truth focuses on the 100 most differentially expressed genes (according to TSNI) with respect to the induction of the TRP63 transcription factor, these results indicate that the proposed approach of ranking indeed highlights differentially expressed genes and that it naturally displays an attractive degree of robustness (similar AUC) against different kinds of noise.

References

[1] Angelini, C., De Canditiis, D., Mutarelli, M., Pensky, M.: A Bayesian approach to estimation and testing in time-course microarray experiments. Stat Appl Genet Mol Biol 6, 24 (2007)

[1] Available on the GEO database, under accession number GSE10562. Ranking list of direct targets is available for download: `genome.cshlp.org/content/suppl/2008/05/05/gr.073601.107.DC1/DellaGatta_SupTable1.xls`



(a) (b) (c)

Fig. 2. One ROC curve for the GP method and three for BATS, using a different noise model (subscript "G" for Gaussian, "T" for Student's-t and "DE" for double exponential marginal distributions of error), followed by the area under the corresponding curve (AUC). **(a)** Data simulated by BATS, induced with Gaussian noise. Very similar results were acquired for simulated data induced with Student's-t with 5 degrees of freedom and 3 degrees of freedom. **(b)** On data simulated by GPs. **(c)** On experimental data from [4].

[2] Bansal, M., Gatta, G.D., Di Bernardo, D.: Inference of gene regulatory networks and compound mode of action from time course gene expression profiles. Bioinformatics 22(7), 815 (2006)

[3] Bar-Joseph, Z.: Analyzing time series gene expression data. Bioinformatics 20(16), 2493 (2004)

[4] Della Gatta, G., Bansal, M., Ambesi-Impiombato, A., Antonini, D., Missero, C., di Bernardo, D.: Direct targets of the TRP63 transcription factor revealed by a combination of gene expression profiling and reverse engineering. Genome research 18(6), 939 (2008)

[5] Honkela, A., Girardot, C., Gustafson, E.H., Liu, Y.H., Furlong, E.E.M., Lawrence, N.D., Rattray, M.: Model-based method for transcription factor target identification with limited data. Proceedings of the National Academy of Sciences 107(17), 7793 (2010)

[6] Kalaitzis, A.A., Lawrence, N.D.: A simple approach to ranking differentially expressed gene expression time courses through gaussian process regression. BMC Bioinformatics 12(180) (2011)

[7] Lawrence, N.D., Sanguinetti, G., Rattray, M.: Modelling transcriptional regulation using Gaussian processes. Advances in Neural Information Processing Systems 19, 785 (2007)

[8] Rasmussen, C.E., Williams, C.K.I.: Gaussian Processes for Machine Learning. MIT Press, Cambridge, MA (2006)

[9] Stegle, O., Denby, K.J., Cooke, E.J., Wild, D.L., Ghahramani, Z., Borgwardt, K.M.: A robust Bayesian two-sample test for detecting intervals of differential gene expression in microarray time series. Journal of Computational Biology 17(3), 355–367 (2010)

[10] Storey, J.D., Xiao, W., Leek, J.T., Tompkins, R.G., Davis, R.W.: Significance analysis of time course microarray experiments. Proceedings of the National Academy of Sciences of the United States of America 102(36), 12837 (2005)

[11] Tai, Y.C., Speed, T.P.: A multivariate empirical Bayes statistic for replicated microarray time course data. The Annals of Statistics 34(5), 2387–2412 (2006)

Linear Model for Network Inference using RNA interference Data

Bettina Knapp and Lars Kaderali

Heidelberg University, ViroQuant Research Group Modeling, BioQuant BQ26,
Im Neuenheimer Feld 267, 69120 Heidelberg, Germany
{bettina.knapp,lars.kaderali}@bioquant.uni-heidelberg.de
http://hades1.bioquant.uni-heidelberg.de/

Abstract. RNA interference in combination with gene expression measurements provides a useful tool not only for the identification of gene function but also for the inference of gene-gene interactions. However, learning the underlying network of a given biological process remains a challenging task since the problem space increases exponentially with network size and furthermore, the given data is in most cases noisy, incomplete or both. We present a model which infers signaling networks using a linear optimization program which can be solved efficiently even for larger network sizes. The model can easily deal with double or multiple knockdowns and integrate prior knowledge information. Using simulated data we show that our model performs much better than random guessing even for noisy or missing data. Furthermore, we outperform a recently proposed method on simulated and on real biological data studying the ERBB signaling where we could confirm several gene interactions as well as identify potential new ones.

Keywords: Network Inference, Linear Optimization, Linear Programming, RNAi Data

1 Introduction

Over the past years, RNA interference (RNAi) has been extensively used to perform high-throughput, high-content knockdown experiments which allow the functional characterization of genes in living cells. However, to study the behavior of complex biological systems it is necessary to understand how genes interact with each other in the underlying signaling network. Hence, the reconstruction of networks from RNAi data is a challenging problem. One of the problems is the exponentially increasing dimension space for increasing network elements. Thus, a complete enumeration of the solution space is not feasible for many network inference approaches when there are more than five to six genes [3, 4].
We formulate the network inference task as a linear optimization problem (LP) which can be solved efficiently even for large network sizes. The input of our method can be RNAi or any other gene perturbation data where the effects on the remaining elements can be quantified for example using additional expression

measurements. Ourfali et al. [5] proposed an integer programming formulation for the explanation of knockout effects on gene expression levels and the inference of an integrated network of protein-protein and protein-DNA interactions. In contrast to their method, our approach does not need a pre-given interaction network and can infer the signaling network purely from data. Furthermore, our LP model can easily handle incomplete data.

2 Linear Model for Network Inference

The model presented here is based on the assumption that after the knockdown of a gene its descendant genes in the underlying network show a phenotypic effect. Therefore, a protein a influences other proteins which are further down in the network topology (i.e. b), if there exists a path from a to b. Furthermore, we assume that the information flow within a network begins at one or several source nodes S and it is then propagated down through the network until it reaches one or several final nodes F. We classify all genes of a study whether they are *active* or *inactive* after respective gene perturbations. Nodes are active, if the sum of incoming edge weights from predecessor nodes are higher than a pre-given threshold and inactive otherwise. We formulate the network inference problem as an optimization problem which uses the observed data to find a network topology that minimizes absolute edge weights $w_{ij} \in \mathbb{R}$ between node i and j and fits the data best. The model is flexible since additional constraints can be easily formulated and only little restrictions (for example self-regulating edges are excluded) are requested on the network structures.

Given are n different genes, K different knockdowns of one or several genes at the same time and observations $x_{ik} \in \mathbb{R}_{\geq 0}$ for $i \in \{1, \ldots, n\}$ genes and $k \in \{1, \ldots, K\}$ knockdowns. Gene i is active after knockdown k if $x_{ik} \geq \delta_i$ and inactive otherwise, with δ_i being calculated from the data.
The LP is defined as:

$$\min z(w_{ji}^+, w_{ji}^-, w_i^0, \xi_l) := \left(\sum_{i,j} (w_{ji}^+ + w_{ji}^-) + \sum_i w_i^0 + \lambda \sum_l \xi_l \right) \tag{1}$$

$$\text{s.t. if } x_{ik} \geq \delta_i \text{ and } b_{ik} = 1: \quad w_i^0 + \sum_{j \neq i} (w_{ji}^+ - w_{ji}^-) x_{jk} \geq \delta_i - \xi_l \tag{2}$$

$$\text{if } x_{ik} < \delta_i \text{ and } b_{ik} = 1: \quad w_i^0 + \sum_{j \neq i} (w_{ji}^+ - w_{ji}^-) x_{jk} \leq 0 \tag{3}$$

$$\text{if } i \in V \setminus S: \quad \sum_{j \in V, j \neq i} (w_{ji}^+ + w_{ji}^-) \geq \delta_i \tag{4}$$

$$\text{if } i \in V \setminus F: \quad \sum_{j \in V, j \neq i} (w_{ij}^+ + w_{ij}^-) \geq \delta_i \tag{5}$$

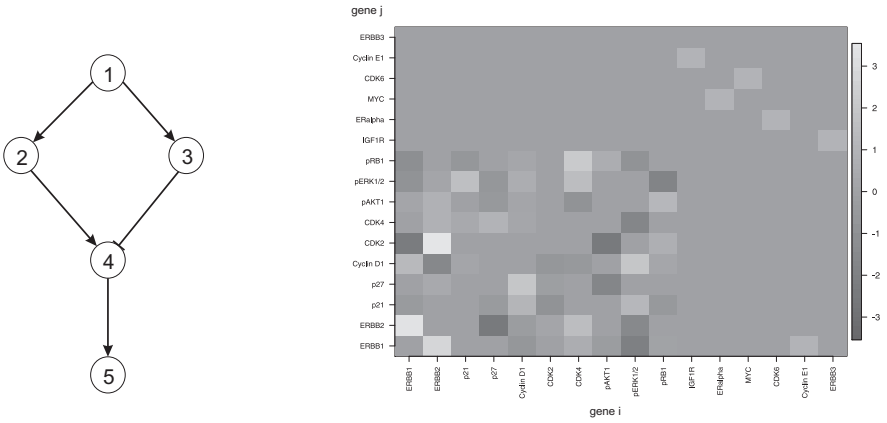
$$\text{if } w_{ij} \text{ is known to be} \geq m \in \mathbb{R}: \quad w_{ji} \geq m, \tag{6}$$

where the optimization function z in equation 1 is minimized over three terms. The first term accounts for the absolute edge weights $w_{ji} = w_{ij}^{+} + w_{ij}^{-}$ with $w_{ij}^{+}, w_{ij}^{-} \in \mathbb{R}_{\geq 0}$ representing activating and deactivating interactions, respectively. The second term optimizes offset variables $w_i^0 \in \mathbb{R}_{\geq 0}$ which denote the baseline activity of the genes and the third term allows to deal with noisy data by using *slack* variables ξ_l with $\xi_l \in \mathbb{R}_{\geq 0}$.
The information whether a gene has been silenced in a certain knockdown is denoted with parameter b_{ik} being equal to zero and b_{ik} being equal to one otherwise. Parameter $\lambda \in \mathbb{R}_{\geq 0}$ is defined as a non-negative penalty parameter to control the introduction of slack variables and thus, the trade-off between sparcity and connectedness of the network.
The constraints 2 and 3 specify the effects of the knockdowns. We assume that the activity of each gene i is given by the addition of its baseline activity (w_i^0) and the activity of preceding genes (x_{jk} with $j \neq i$) multiplied with the corresponding edge weights ($w_{ji}^{+} - w_{ji}^{-}$) after knockdown k. Thus, if gene i is observed to be activated after the knockdown k, so $x_{ik} \geq \delta_i$ (and it has not been silenced, so $b_{ik} = 1$), the sum of the incoming information flow and its baseline activity has to be greater or equal to $\delta_i - \xi_l$ and smaller or equal to zero otherwise (see constraints 2 and 3).
The inequalities given in 4 and 5, respectively, force each node which is not a start or end node to have at least one incoming and one outgoing edge, respectively. Both constraints are necessary to avoid *lose ends*. By lose ends we mean for example a node which is not a start node but has no incoming information flow, or a node which is not an end node, but has no outgoing information flow.
The last constraint (6) exemplifies how already known interactions from all kinds of biological prior knowledge can be integrated in the network inference model.

3 Results on Simulated Data and Real Data on ERBB Signaling

3.1 Simulated Data

Assume a toy network topology to be given like in Figure 1 (a). Assuming single knockdowns for each gene and a double-knockdown for gene 2 and 3 to be given, we simulated 100 times data based on this network topology using uniform random noise with $\mathcal{U}(0, sd)$ for $sd = \{0, \ldots, 0.5\}$. Moreover, we simulated missing data where at least one data point to maximally 80% are missing and noise has been generated with $\mathcal{U}(0, 0.1)$. We solved the corresponding LPs using $\delta = 0.5$, $\lambda = 1$ and calculated the area under the ROC- (AU-ROC) and the PR-curve (AU-PR). The resulting mean and standard deviation of AU-ROC$= 0.96 \pm 0.04$ and AU-PR$= 0.88 \pm 0.11$ for the noisy data and AU-ROC$= 0.68 \pm 0.11$ and AU-PR$= 0.44 \pm 0.15$ for the missing data show, that we are each time much better than random guessing values (AU-PR$= 0.56$, AU-ROC$= 0.23$ for three class-problems like given here). Not surprisingly, the overall performance of the model is decreasing the larger the noise and the more data points are missing



(a) Toy network (b) Imageplot representing inferred edges

Fig. 1. (a) Toy network topology with five nodes. Arrows indicate activation and between node 3 and 4 an inactivation is shown. (b) Imageplot representing inferred edges (gene i influences gene j) for all genes of the ERBB signaling data. The edge weights are coded in grey-scale with white corresponding to positive and black to negative interactions.

(data not shown).
We compared the performance of our LP-model with a recently proposed network inference method proposed by Froehlich *et al.* which is called deterministic effects propagation networks (DEPNs) [2]. Results of the DEPNs for the noisy data average to AU-ROC= 0.82 ± 0.02 and AU-PR= 0.7 ± 0.04, which clearly demonstrates that we are highly outperforming their proposed method.

3.2 Real Data

We tested our model on a real biological problem using normalized data from literature [2] where 16 genes (ERBB1, ERBB2, p21, p27, Cyclin D1, CDK2, CDK4, pAKT1, pERK1/2, pRB1, IGF1R, ERalpha, MYC, CDK6, Cyclin E1, ERBB3) of the ERBB signaling network have been measured using 16 knockdowns (including 3 double knockdowns) and MOCK (no stimulation). Each knockdown has been repeated in 4 technical and 3 biological replicates. Reverse Phase Protein Array measurements have been carried out for 10 network intermediates, each time before and twelve hours after EGF stimulation. We summarized replicates using mean and solved the corresponding LP with $\lambda = 4$ and $\delta_i = MOCK_i$ at time=0 for each gene i.
Results are shown in Figure 1 (b) where the inferred edge weights are coded in grey-scale with white indicating positive and black negative interactions, respectively. The inferred gene interactions which are most interesting (and

which have highest edge weights) are the activation of gene ERBB1 by ERBB2 (weight=2.8) and CDK2 by ERBB2 (weight=3.5). The most pronounced inactivation is learned between gene pAKT and CDK2 (weight=-2.4). All three edges have been already reported in literature and inferred using DEPNs [2].
Apart from the gene-interactions already inferred by Froehlich, we additionally learned the activation of ERBB2 through ERBB1 (weight=3.4), which is not surprising, since they are forming heterodimers [1]. Furthermore, our results indicate that pERK1/2 is inactivating ERBB1 (weight=-2.1). Although this interaction has not been explicitly reported in literature yet, Chen *et al.* showed that the ERBB response is silenced by negative feedback from active ERK [1] and thus, this is strongly supporting our results.

4 Discussion

We formulated the challenge of linear network inference as an LP, which can be solved efficiently even for large-scale problems. The model is able to include prior knowledge and can easily handle double or multiple gene knockdowns at the same time. Moreover, the method presented here determines whether a gene-gene interaction is activating or deactivating and both, discrete and continuous data can be processed. We showed on simulated data that the model is able to deal with missing and noisy data with performance significantly better than random guessing as well as a recently published approach. Furthermore, using our method on real biological data studying ERBB signaling we were able to confirm already known interactions given in the literature and additionally, identify new ones.

References

1. Chen, W.W., Schoeberl, B., Jasper, P.J., Niepel, M., Nielsen, U.B., Lauffenburger, D.A., Sorger, P. K.: Inputoutput behavior of ErbB signaling pathways as revealed by a mass action model trained against dynamic data. Mol Syst Biol., vol. 5, p 239 (2009)
2. Froehlich, H., Sahin, O., Arlt, D., Bender, C., Beissbarth, T.: Deterministic Effects Propagation Networks for reconstructing protein signaling networks from multiple interventions. BMC Bioinformatics, vol. 10, pp 322 (2009)
3. Kaderali, L., Dazert, E., Zeuge, U., Frese, M., Bartenschlager, R.: Reconstructing signaling pathways from RNAi data using probabilistic Boolean threshold networks. Bioinformatics, vol. 25, p 2229-2235 (2009)
4. Markowetz, F., Bloch, J., Spang, R.: Non-transcriptional pathway features reconstructed from secondary effects of RNA interference. Bioinformatics, vol. 21, p 4026-4032 (2005)
5. Ourfali, O., Shlomi, T., Ideker, T., Ruppin, E., Sharan, R.: SPINE: a framework for signaling-regulatory pathway inference from cause-effect experiments. Bioinformatics, vol 23, p i359-366 (2007)

Gaussian Logic for Proteomics and Genomics

Ondřej Kuželka, Andrea Szabóová, Matěj Holec, and Filip Železný

Czech Technical University, Prague, Czech Republic
{kuzelon2,szaboand,holecmat,zelezny}@fel.cvut.cz,

1 Introduction

We describe a statistical relational learning framework called Gaussian Logic capable to work efficiently with combinations of relational and numerical data. The framework assumes that, for a fixed relational structure, the numerical data can be modelled by a multivariate normal distribution. We show how the Gaussian Logic framework can be used to predict DNA-binding propensity of proteins and to find motifs describing novel gene sets which are then used in set-level classification of gene expression samples[1].

2 A Probabilistic Framework

We address the situation where training examples have both *structure* and *real parameters*. One example may e.g. describe a measurement of the expression of several genes; here the structure would describe functional relations between the genes and the parameters would describe their measured expressions. Note that we allow different structures in different examples. In the genomic example, a training set thus may consist of measurements pertaining to different gene sets, each giving rise to a different structure of mutual relations between the genes.

To describe such training examples as well as learned models, we use a conventional first-order logic language $\mathcal{L}$ whose alphabet contains a distinguished set of constants $\{\mathsf{r}_1, \mathsf{r}_2, \ldots \mathsf{r}_n\}$ and variables $\{R_1, R_2, \ldots R_m\}$. Any substitution in our framework must map variables (other than) R_i only to terms (other than) r_j. The structure of an example is described by a (Herbrand) interpretation H, in which the constants r_i represent uninstantiated real parameters. The parameter values are then determined by a real vector $\boldsymbol{\theta}$. Thus each example is a pair $(H, \boldsymbol{\theta})$. Examples are assumed to be sampled from the distribution

$$P\left(H, \Omega_H\right) = \int_{\Omega_H} f_H\left(\boldsymbol{\theta}|H\right) P\left(H\right) \mathrm{d}\boldsymbol{\theta}$$

which we want to learn (where $\Omega_H \subseteq R^n$). Here, $P(H)$ is a discrete probability distribution on the countable set of Herbrand interpretations of $\mathcal{L}$. $f_H\left(\boldsymbol{\theta}|H\right)$

[1] A longer version of this paper appears at the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD 2011) under the title: "Gaussian Logic for Predictive Classification".

are the conditional densities of the parameter values. The advantage of this definition is that it cleanly splits the possible-world probability into the discrete part $P(H)$ which can be modeled by state-of-the-art approaches such as Markov Logic Networks (MLN's) [3], and the continuous conditional densities $f_H(\boldsymbol{\theta}|H)$ which we elaborate here. In particular, we assume that $f(\boldsymbol{\theta}|H) = N(\boldsymbol{\mu}_H, \Sigma_H)$, i.e., $\boldsymbol{\theta}$ is normally distributed with mean vector $\boldsymbol{\mu}_H$ and covariance matrix Σ_H. The indexes H emphasize the dependence of the two parameters on the particular Herbrand interpretation that is parameterized by $\boldsymbol{\theta}$.

To learn $P(H, \Omega_H)$ from a sample E, we first discuss a strategy that suggests itself readily. We could rely on existing methods (such as MLN's) to learn $P(H)$ from the multi-set $\mathcal{H}$ of interpretations H occurring in E. Then, to obtain $f(\boldsymbol{\theta}|H)$ for each $H \in \mathcal{H}$, we would estimate $\boldsymbol{\mu}_H, \Sigma_H$ from the multi-set $\hat{\Omega}_H$ of parameter value vectors $\boldsymbol{\theta}$ associated with H in the training sample E. The problem of this approach is that, given a fixed size of the training sample, when $\mathcal{H}$ is large, the multi-sets $\hat{\Omega}_H, H \in \mathcal{H}$ will be small, and thus the estimates of $\boldsymbol{\mu}_H, \Sigma_H$ will be poor.

Our strategy is instead to discover *Gaussian features* of the training examples. A Gaussian feature is logic formula which, roughly said, extracts some of the parameter values for each example into a vector such that this vector is approximately normally distributed across the training sample. For example, the intentionally simplistic feature

$$\exists G_1, G_2 \quad \mathrm{expr}(G_1, R_1) \wedge \mathrm{expr}(G_2, R_2) \wedge \mathrm{regulates}(G_1, G_2)$$

contains two standard FOL variables G_1, G_2 and two distinguished variables R_1, R_2, and indicates that expressions of any two genes (G_1, G_2) in the regulation relation are co-distributed normally. The corresponding mean vector and covariance matrix are then estimated from all training examples whose structures contain one or more pairs of such related genes. The learned features then act as constraints on the target distribution $P(H, \Omega_H)$. By choosing the number of employed features, we are able to trade off between under- and over-constraining the target distribution model.

In general, the problem of estimating parameters of Gaussian features is an NP-hard problem. However, it is tractable for a class of features, *conjunctive tree-like features* for which we have devised also an efficient feature construction algorithm based on the feature-construction algorithm from [8]. It shares most of the favourable properties of the original algorithm like detection of redundant features.

3 Predictive Classi cation Applications

A straightforward application of the Gaussian-logic framework is in Bayesian classification. We address a case study involving an important problem from biology: prediction of DNA-binding propensity of proteins. Several computational approaches have been proposed for the prediction of DNA-binding function from protein structure. It has been shown that electrostatic properties of proteins are

good features for predictive classification (e.g. [1, 2]). A more recent approach is the method of Szilágyi and Skolnick [9] who created a logistic regression classifier based on 10 features also including electrostatic properties.

Here, we use Gaussian logic to create a model for capturing distributions of positively charged amino acids in protein sequences. We split each protein into consecutive non-overlapping *windows*, each containing l_w amino acids (possibly except for the last window which may contain less amino acids). For each window of a protein P we compute the value a_i^+/l_w where a_i^+ is the number of positively charged amino-acids in the window i. Then for each protein P we construct an example $e_P = (H_P, \boldsymbol{\theta}_P)$ where $\boldsymbol{\theta}_P = \left(a_1^+/l_w, a_2^+/l_w, \ldots, a_{n_P}^+/l_w\right)$ and $H_P = w(1, r_1)$, $next(1, 2)$, $\ldots$, $next(n_P - 1, n_P)$, $w(n_P, r_P)$. We constructed only one feature $F_{non} = w(A, R_1)$ for non-DNA-binding proteins since we do not expect this class of proteins to be very homogeneous. For DNA-binding proteins, we constructed a more complex model by selecting a set of features using a greedy search algorithm. The greedy search algorithm optimized classification error on training data. Classification was performed by comparing, for a tested protein, the likelihood-ratio of the two models (DNA-binding and non-DNA-binding) with a threshold selected on the training data. We estimated the accuracy of this method using 10-fold cross-validation (always learning parameters and structure of the models and selecting the threshold and window length l_w using only the data from training folds) on a dataset containing 138 DNA-binding proteins (PD138 [9]) and 110 non-DNA-binding proteins (NB110 [1]). The estimated accuracies (*Gaussian Logic*) are shown in Table 1. The method performs similarly well as the method of Szilagyi et al. [9] (in fact, it outperforms it slightly but the difference is rather negligible) but uses much less information. Next, we were interested in the question whether the machinery of Gaussian logic actually helped improve the predictive accuracy in our experiments or whether we could obtain the same or better results using only the very simple feature $F = w(A, R_1)$ also to model the DNA-binding proteins, thus ignoring any correlation between charges of different parts of a protein (*Baseline Gaussian Logic* in Table 1). Indeed, the machinery of Gaussian Logic appears to be helpful from these results.

Method	**Accuracy [%]**
Szilágyi et al.	81.4
Baseline Gaussian logic	78.7
Gaussian logic	**81.9**

Table 1. Accuracies estimated by 10-fold cross-validation on PD138/NB110.

It is interesting how well the Gaussian-logic model performed considering the fact that it used so little information (it completely ignored types of positively charged amino acids and it also ignored negative amino acids). The model that we presented here can be easily extended, e.g. by adding secondary-structure information. The splitting into consecutive windows used here is rather artificial

and it would be more natural to split the sequence into windows corresponding to secondary-structure units (helices, sheets, coils). The features could then distinguish between consecutive windows corresponding to different secondary-structure units.

Next, we used Gaussian logic to search for novel definitions of gene sets with high discriminative ability. This is useful in set-level classification methods for prediction from gene-expression data [5]. Set-level methods are based on aggregating values of gene expressions contained in pre-defined gene sets and then using these aggregated values as features for classification. We constructed examples $(H_S, \boldsymbol{\theta}_S)$ from gene-expression samples and KEGG pathways [7] as follows. For each gene g_i, we introduced a logical atom $g(g_i, r_i)$ to capture its expression level. Then we added all relations extracted from KEGG as logical atoms $relation(g_i,\ g_j,\ relationType)$. We also added a numerical indicator of class-label to each example as a logical atom $label(\pm 1)$ where $+1$ indicates a positive example and -1 a negative example. Finally, for each gene-expression sample S we constructed the vector of the gene-expression levels $\boldsymbol{\theta}_S$. Using our feature construction algorithm we constructed a large set of tree-like features involving exactly one atom $label(L)$, at least one atom $g(G_i, R_i)$ and relations *expression*, *repression*, *activation*, *inhibition*, *phosphorylation*, *dephosphorylation*, *state* and *binding/association*. After that we selected a subset of features according to the correlation of the average expression of the involved genes with the class label, which can be extracted from the estimated Gaussian-feature parameters.

Dataset	**GL**	**FCF**	**Dataset**	**GL**	**FCF**
Collitis	80.0	**89.4**	Pheochromocytoma	**64.0**	56.0
Pleural Mesothieloma	**94.4**	92.6	Prostate cancer	**85.0**	80.0
Parkinson 1	52.7	**54.5**	Squamus cell carcinoma	**95.5**	88.6
Parkinson 2	**66.7**	63.9	Testicular seminoma	58.3	**61.1**
Parkinson 3	62.7	**77.1**	**Wins**	**5**	4

Table 2. Accuracies of set-level-based classifiers with Gaussian-logic features and FCF-based features, estimated by leave-one-out cross-validation.

We have constructed features using a gene-expression dataset from [4] which we did not use in the subsequent predictive classification experiments. We have compared gene sets constructed by the outlined procedure with gene sets based on so called *fully-coupled fluxes (FCFs)* which are biologically-motivated gene sets used previously in the context of set-level classification [5]. We constructed the same number of gene sets for our features as was the number of FCFs. The accuracies of an SVM classifier (estimated by leave-one-out cross-validation) are shown in Table 2. We can notice that the gene sets constructed using our novel method performed equally well as the gene sets based on fully-coupled fluxes. Interestingly, our gene sets contained about half the number of genes as compared to FCFs and despite that they were able to perform equally well.

4 Conclusions and Future Work

We have introduced a novel relational learning system capable to work efficiently with combinations of relational and numerical data. The experiments gave us some very promising results, slightly outperforming methods based on features hand-crafted by biologists using only automatically constructed Gaussian features. Furthermore, there are other possible applications of Gaussian logic in predictive classification settings which were not discussed in this paper. For example, finding patterns that generally correspond to highly correlated sets (not necessarily correlated with the class) of genes may have applications with group-lasso based classification approaches [6].

Acknowledgement: We thank the anonymous reviewers of MLSB and ECML PKDD for their valuable comments. This work was supported by the Czech Grant Agency through project 201/09/1665 *Bridging the Gap between Systems Biology and Machine Learning* and project 103/11/2170 *Transferring ILP techniques to SRL*.

References

1. Shandar Ahmad and Akinori Sarai. Moment-based prediction of dna-binding proteins. *Journal of Molecular Biology*, 341(1):65 – 71, 2004.
2. Nitin Bhardwaj, Robert E. Langlois, Guijun Zhao, and Hui Lu. Kernel-based machine learning protocol for predicting DNA-binding proteins. *Nucleic Acids Research*, 33(20):6486–6493.
3. Pedro Domingos, Stanley Kok, Daniel Lowd, Hoifung Poon, Matthew Richardson, and Parag Singla. Probabilistic inductive logic programming. chapter Markov logic, pages 92–117. Springer-Verlag, 2008.
4. William A Freije et al. Gene expression profiling of gliomas strongly predicts survival. *Cancer Res*, 64(18):6503–10, 2004.
5. Matěj Holec, Filip Železný, Jiří Kléma, and Jakub Tolar. Integrating multiple-platform expression data through gene set features. In *Proceedings of the 5th International Symposium on Bioinformatics Research and Applications*, ISBRA '09, pages 5–17. Springer-Verlag, 2009.
6. Laurent Jacob, Guillaume Obozinski, and Jean-Philippe Vert. Group lasso with overlap and graph lasso. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 433–440. ACM, 2009.
7. M. Kanehisa, S. Goto, S. Kawashima, Y. Okuno, and M. Hattori. The kegg resource for deciphering the genome. *Nucleic Acids Research*, 1, 2004.
8. O. Kuželka and F. Železný. Block-wise construction of tree-like relational features with monotone reducibility and redundancy. *Machine Learning*, 83:163–192, 2011.
9. András Szilágyi and Jeffrey Skolnick. Efficient prediction of nucleic acid binding function from low-resolution protein structures. *Journal of Molecular Biology*, 358(3):922 – 933, 2006.

Probabilistic dependency models for data integration in functional genomics

Leo Lahti (1) and Samuel Kaski (2)

(1) University of Helsinki, Department of Veterinary Bioscience, Finland (2) Aalto University and University of Helsinki, Helsinki Institute for Information Technology HIIT
leo.lahti@iki.fi, samuel.kaski@hiit.fi

Abstract. *Co-occurring genomic observations are increasingly available in biomedical studies, providing complementary views to genome function. Integrative analysis of these data sources can reveal dependencies and interactions that cannot be detected based on individual data sources. Prior information of the application domain can guide the search for novel multi-view biomarkers that have potential diagnostic, prognostic and clinical relevance. We propose an integrative analysis framework based on regularized probabilistic canonical correlation analysis with particular applications in cancer gene discovery and the analysis of human intestinal microbiota.*

Keywords: canonical correlation analysis, data integration, dependency modeling, functional genomics

1 Introduction

Complementary genomic observations of gene- and micro-RNA expression, DNA copy number, methylation status, and host-microbiome interactions are increasingly available in biomedical studies public repositories such as the Cancer Genome Atlas [1]. Analysis of statistical dependencies between different functional layers of the genome allows the discovery of regularities and interactions that are not seen in individual data sets. For instance, integrative analysis of gene expression and copy number measurements can reveal cancer-associated chromosomal regions with potential clinical relevance. Variants of probabilistic canonical correlation analysis (CCA) [2] provide a robust framework for data integration in functional genomics that can deal with the uncertainties associated with small sample sizes common in biomedical studies and provide tools to guide dependency modeling through Bayesian priors [3]. We apply these models to detect and characterize functionally active chromosomal changes in gastric cancer and discuss other biomedically relevant extensions of the model.

2 Regularized dependency detection framework

Dependency between two data sources can be modeled by decomposing the observations into shared and data set specific components. Let us consider two sets

of co-occurring genomic observations, X, Y. The shared effects are described by a shared latent variable $\mathbf{z}$ whose manifestation in each data set is characterized by linear transformations W_x and W_y, respectively. Independent data set-specific effects are denoted by $\varepsilon_x, \varepsilon_y$. This gives the model

$$\begin{aligned} X &\sim W_x\mathbf{z} + \varepsilon_x \\ Y &\sim W_y\mathbf{z} + \varepsilon_y \end{aligned} \tag{1}$$

In standard probabilistic CCA [2], the shared latent variable $\mathbf{z}$ follows standard multivariate normal distribution and the data-set specific effects are described by multivariate Gaussians with covariance matrices Ψ_x and Ψ_y, respectively. Biomedical screening studies often focus on particular types of regulation and unconstrained models easily lead to overfitting with small sample size. We incorporate domain-specific prior knowledge to focus on specific types of dependency. For instance, imposing particular structure on the marginal covariances could be used to data set specific prior information, and non-negativity constraints on W would focus on positive regulation. We show how constraining the relation between W_x and W_y helps to model spatial dependencies in chromosomally local gene neighborhoods [3].

3 Detecting functionally active DNA mutations

DNA alterations are a key mechanism in cancer development. An important task in cancer studies is to distinguish so-called *driver* mutations from the less active *passengers*. Driver mutations that affect expression levels of the associated genes will contribute to dependencies between gene copy number and expression and detecting such regions will reveal potential candidate genes for cancer studies. Such dependencies are spatially constrained: probes with small chromosomal distance are expected to show similar changes in both data sources. This is encoded by requiring that the transformations W_x, W_y are similar. To enforce this we use a symmetric prior $W_x \sim N(W, \Sigma_w)$, $W_y \sim N(W, \Sigma_w)$. Isotropic covariance matrix $\Sigma_w = \sigma I$, using σ to tune the similarity between W_x and W_y. With $\sigma \to \infty$ the transformations are uncoupled, yielding ordinary probabilistic CCA. Comparisons to another extreme, $\sigma \to 0$, which gives $W_x = W_y$ confirm that the regularized variant outperforms the unregularized model in cancer gene discovery.

To prioritize cancer-associated chromosomal regions, dependency is quantified within each gene neighborghood with a sliding window approach over the genome. The regions are sorted based on the dependency, which is quantified by the ratio of shared vs. data set-specific effects $\frac{Tr(WW^T)}{Tr(\Psi)}$, where $W = [W_x W_y]$ and Ψ is a block-diagonal matrix of the data set specific covariances Ψ_x, Ψ_y. A fixed dimensionality (window size around each gene) yields dependency scores that are directly comparable between the regions.

Figure 1**A** illustrates the dependencies across chromosome arm 17q in gastric cancer [5]. Genome-wide analysis of the dependencies confirms the overall cancer

gene detection performance of the model [3] and shows favorable performance when compared to other recently proposed integrative approaches for cancer gene discovery, including standard correlation- and regression-based alternatives such as DR-Correlate [7] (manuscript in preparation). The model parameters are directly interpretable: a ML-estimate of the shared latent variable $\mathbf{z}$ indicates signal strength in each sample while W highlights probes that capture the shared signal (Fig. 1**B**). The model can detect rare copy number events that are manifested only in a subset of probes, which is an important property for cancer studies.

Extensions of the model can be used to investigate cancer-associated changes on micro-RNA and epigenetic regulation [4, 6], or associations between intestinal microbiota and human physiology, which is of particular interest for understanding cancer development in the gastrointestinal tract - a causal link between *H. Pylori* infection and gastric cancer has been established but the role of microbial changes in other types of cancer in gastrointestinal tract remain poorly characterized.

4 Conclusion

Modeling of dependencies can reveal regularities and interactions that are not seen in individual data sets. Regularized variants of probabilistic CCA provide efficient tools to investigate statistical dependencies between complementary genomic observations and to guide dependency detection through Bayesian priors. Implementations of dependency detection models and application tools are available through CRAN[1] and BioConductor[2].

Acknowledgments. This work has been partially funded by TEKES (grant 40141/07)

References

1. The Cancer Genome Atlas Research Network. Comprehensive genomic characterization defines human glioblastoma genes and core pathways. *Nature*, 455:1061–1068, 2008.
2. F. R. Bach and M. I. Jordan. A probabilistic interpretation of canonical correlation analysis. Technical report, Department of Statistics, University of California, Berkeley, 2005.
3. L. Lahti, S. Myllykangas, S. Knuutila, and S. Kaski. Dependency detection with similarity constraints. In *Proc. MLSP'09 IEEE International Workshop on Machine Learning for Signal Processing XIX*, pages 89–94, Piscataway, NJ, 2009.

[1] http://dmt.r-forge.r-project.org/
[2] http://www.bioconductor.org/packages/devel/bioc/html/pint.html

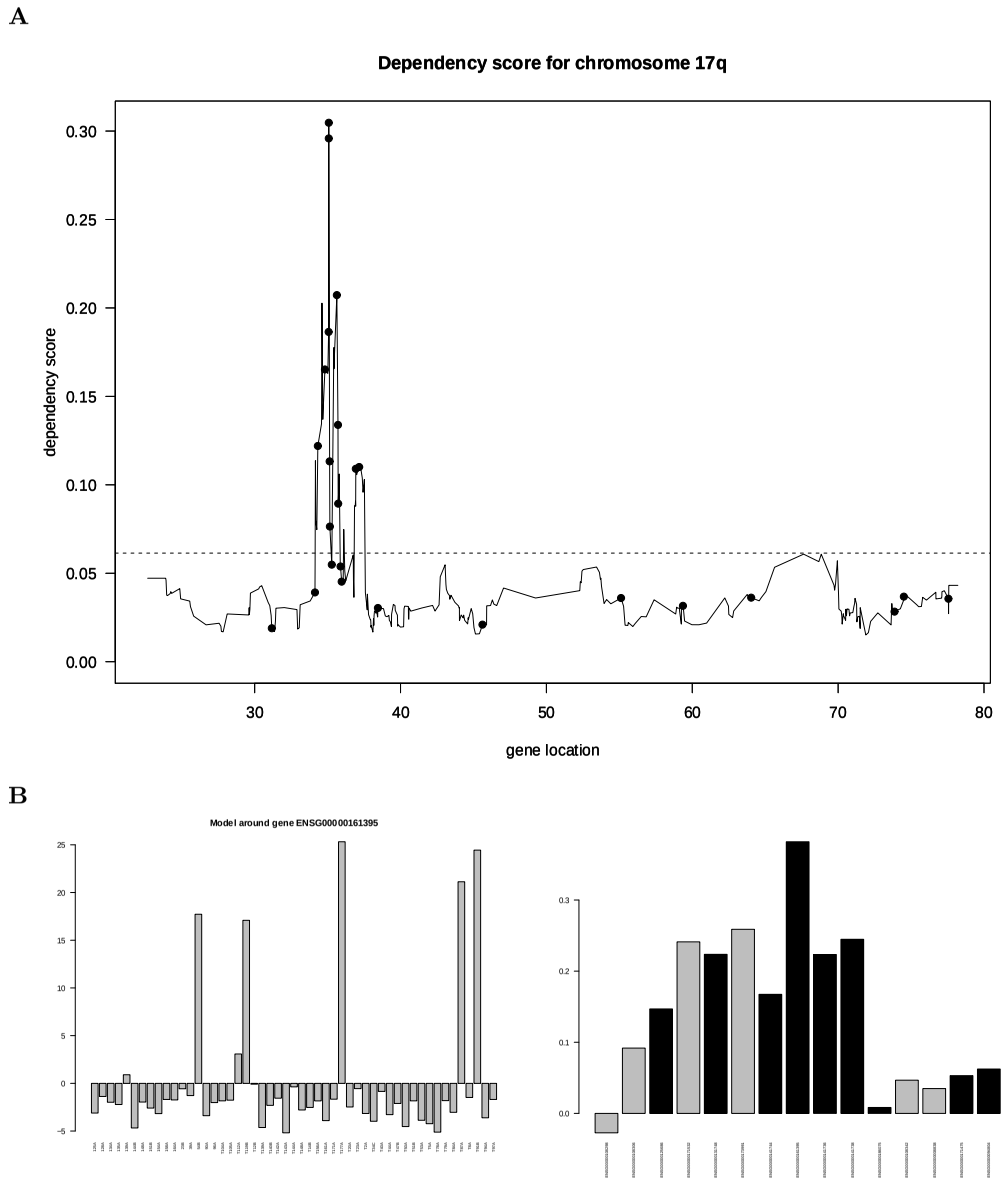


Fig. 1. A Dependency score between gene copy number and expression across chromosomal arm 17q in gastric cancer. The black dots highlight known gastric cancer associated genes. **B** Visualization of the model parameters around the gene with the highest dependency score: ML-estimate of the latent variable $\mathbf{z}$ and the estimated model parameter W quantify sample and variable contributions to the dependencies, respectively, highlighting the affected patients and genes in this region.

4. R. Louhimo, V. Aittomäki, A. Faisal, M. Laakso, P. Chen, K. Ovaska, E. Valo, L. Lahti, V. Rogojin, S. Kaski and S. Hautaniemi. Systematic Use of Computational Methods Allows Stratifying Treatment Responders in Glioblastoma Multiforme. *Critical Assessment of Massive Data Analysis (CAMDA) workshop.* ISMB, Vienna, Austria, July 2011.
5. S. Myllykangas, S. Junnila, A. Kokkola, R. Autio, I. Scheinin, T. Kiviluoto, M.-L. Karjalainen-Lindsberg, J. Hollmén, S. Knuutila, P. Puolakkainen, and O. Monni. Integrated gene copy number and expression microarray analysis of gastric cancer highlights potential target genes. *International Journal of Cancer*, 123:817–825, 2008.
6. P. Nymark, M. Guled, I. Borze, A. Faisal, L. Lahti, K. Salmenkivi, E. Kettunen, S. Anttila and S. Knuutila. Integrative Analysis of microRNA, mRNA and aCGH Data Reveals Asbestos- and Histology-Related Changes in Lung Cancer. *Genes, Chromosomes and Cancer*. In press.
7. K. Salari, R. Tibshirani, and J. R. Pollack DR-Integrator: a new analytic tool for integrating DNA copy number and gene expression data. *Bioinformatics* 26:414–416.

Bayesian Experimental Design for the Inference of Gene Regulatory Networks

Johanna Mazur and Lars Kaderali

Viroquant Research Group Modeling, University of Heidelberg,
Bioquant BQ26, INF 267, 69120 Heidelberg, Germany
johanna.mazur@bioquant.uni-heidelberg.de,
lars.kaderali@bioquant.uni-heidelberg.de

Abstract. The inference of gene regulatory networks from time-series data is an interesting and challenging task. The challenges arise due to the extensive parameter space, non-identifiability and sloppiness of parameters and noisy data. Bayesian parameter estimation methods have proven to work for this problem, as they provide distributions over parameters instead of point estimates. These distributions can be used to perform Bayesian experimental design, which is essential because data is limited and experiments are expensive. We implemented an efficient sequential Bayesian experimental design framework based on maximum entropy sampling on a high-dimensional non-linear ordinary differential equations model for the inference of gene regulatory networks. As results we show that our method outperforms random experiments by a high degree.

1 Challenges for the Inference of Gene Regulatory Networks

Right from the beginning as the field of systems biology began to emerge, one of the most interesting but also challenging topics was, and still is, the inference of gene regulatory networks (GRNs). Although of high interest, this inference is a difficult task due to several reasons. Firstly, the amount of data is limited and additionally contains a lot of intrinsic and extrinsic noise. Secondly, for increasing number of genes under consideration the number of possible gene network topologies grows exponentially, i.e., the parameter space to be searched is extensive. In general, in parameter estimation procedures one has also to consider non-identifiability [10] and sloppiness [3] of parameters.

All these problems can successfully be addressed in a Bayesian context [8] and several approaches and tools have been proposed (see e.g. [13]). Recently, we applied a Bayesian parameter estimation procedure to an ordinary differential equation (ODE) model for GRNs. We showed the reliability of our approach on simulated data and outperformed the best submitted results on the DREAM2 Challenge #3 data [9]. The distributions of the parameters obtained in [9] still need to be analyzed further to look whether and how different gene network topologies may explain the data in a similar way. To distinguish between these

different topologies, new experiments have to be performed. Nevertheless, since biological experiments need a lot of resources, experimental design is a crucial step to obtain the most informative data for more reliable parameter values in the next round of parameter estimation.

Most often classical experimental design [1] is used to perform optimal experiments. However, it only works for linear problems and for non-linear models the problem is linearized around a point estimate. Thus, not the whole parameter distribution is considered. To reflect the whole distribution, Bayesian experimental design (BED) can be used [2]. Although, in the non-linear and non-Gaussian case it is not analytically tractable and thus computationally demanding and consequently it is a problem far from being solved. In the field of systems biology, several approaches have been proposed for BED. Steinke *et al.* [12] used BED for the reconstruction of GRNs dealing with linear models and it is not obvious how their method can be expanded to non-linear models. Kramer and Radde [7] considered for the inference of dynamic network models only the steady states of the underlying dynamic models and proposed the perturbation experiment where the entropy of the posterior distribution over the model parameters is minimized.

In contrast, we propose in this work an efficient framework for Bayesian experimental design being applicable to high-dimensional non-linear ODE models for parameter estimation and consider the whole dynamic behavior and not only the steady states. We show the pertinence of our method to an ODE model for a five gene network.

2 Bayesian Experimental Design by means of Maximum Entropy Sampling

To perform BED for parameter estimation purposes, one needs two main ingredients. The first one is the *predictive distribution* for future data d for experiments e of interest. This we obtain with a model $M(\omega)$, dependent on the parameters of interest ω. Together with a prior knowledge of ω, the predictive distribution is

$$p(d \mid M_e) = \int p(d \mid \omega, M_e) p(\omega)\, d\omega.$$

The second ingredient is a *utility function* $U(d, e)$ to make the decision which experiment to perform next. The utility function depends on the experiments we consider as possible experiments to be performed and on the data these experiments will generate. Since, of course, this data is not known before the experiment is performed, one chooses the experiment where the *expected utility*

$$EU(e) = \int U(d, e) p(d \mid M_e)\, dd$$

is maximal. As utility function, the *information*

$$I(X) = \int f(x) \ln\big(f(x)\big)\, dx = -\operatorname{Ent}(X)$$

of the new posterior distribution is used, where $\mathrm{Ent}(X)$ denotes the *entropy* of X and $f(\cdot)$ denotes the density of the continuous random variable X. For realistic biological models, i.e., non-linear models, BED has to be solved with Markov chain Monte Carlo methods (MCMC). However, this is computationally intractable, as triple integrals would have to be solved. To avoid this, Shewry and Wynn [11] introduced *maximum entropy sampling* for BED. For many problems, BED with triple integrals can be reformulated into a problem, where we want to find the experiment e where $\mathrm{Ent}\big(p(d \mid \omega, M_e)\big)$ is maximal. To express this in a different way, one can say, that the experiment has to be performed where we have the most uncertainty in the dynamical behavior of the system.

Thus, our method takes into account the suggestion of Gutenkunst *et al.* [3] to look more at the dynamical behavior of the system and less on the perfect parameter estimates because of the universal property of sloppiness inherent in a lot of non-linear ODE models describing biological effects.

3 Results for a 5 gene network

As a model $M(\omega)$ for gene regulatory networks we use non-linear differential equations which are explained in detail in [9]. This ODE model does not only provide parameter values for the connection of the underlying network but offers detailed information about the dynamics of all gene products present. As experiments e we took the measurement of additional time points of gene product concentration for all genes, where we started first with two time points and performed sequential experiments with 21 time points at the end. The probability for data d is

$$p(d \mid \omega, M_e) = \mathcal{N}\big(M_e(\omega), \sigma\big)$$

where $M_e(\omega)$ denotes the values of the data one would obtain, if the parameters ω are the correct ones, i.e., the differential equations have to be numerically integrated. As entropy estimator we use a histogram-based estimator proposed by Györfi and van den Meulen [4]. To get samples for the model parameters needed for the calculation of $\mathrm{Ent}\big(p(d \mid \omega, M_e)\big)$ we use a population-based MCMC algorithm [5] and sample from the distribution

$$p(\omega) = \frac{1}{\sqrt{2\pi\sigma^2}} \prod_{i=1}^{5} \prod_{\tau=0}^{T} \mathrm{e}^{-\frac{1}{2\sigma^2}\left(d_{i\tau} - M(\omega)\right)^2}$$

where $d_{i\tau}$ is the measured data and T is the number of data points we have for every gene. We set $\sigma = 0.01$. Furthermore, to speed up the sampling procedure of the high-dimensional space, we used pMatlab and MatlabMPI [6] to parallelize the sampling procedure. Population-based MCMC algorithms are ideal to do so, because the communication between different subpopulations is very limited, although not completely omitted. We run 20 chains split up in 4 subpopulations.

In Figure 1 the results are depicted. On the x-axis the round of the experiment is denoted and on the y-axis one finds the information of the distribution $p(\omega)$. We ran 10 independent runs of our experimental design procedure and plot

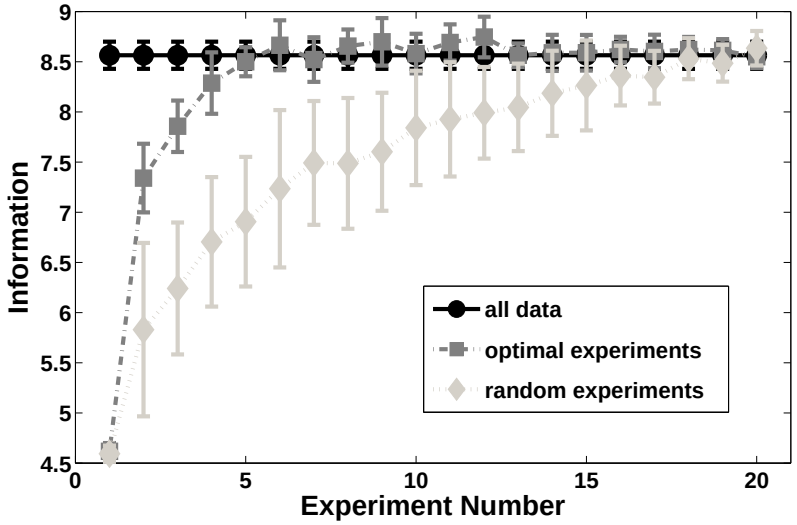


Fig. 1. Results for an ODE model containing 46 parameters for gene network reconstruction. For 10 independent runs the mean and the standard deviation is depicted. The circled line represents the runs with the maximal amount of data available. The diamond line represents random experiments and the squared line illustrates optimal experiments.

the mean together with the standard deviation of these runs. The circled line denotes the runs where we just sampled from the distribution $p(\omega)$, where all 21 data points were given and no experimental design was performed. This reflects the maximal possible information contained in the data and thus in the model parameters. The circled bars depicted for the other experiment rounds are just shown for better comparison to the results of random and optimal experiments. The squared line depicts the optimal experiments and the diamond one depicts the random experiments. It can clearly be seen, that after performing 4 optimal experiments the information in the parameters is as high as if all data points were given, whereas for random experiments one has to add almost all data points to obtain the same information content in the distribution of the parameters.

4 Discussion and Conclusion

An efficient method for sequential optimum Bayesian experimental design by means of maximum entropy sampling was proposed and applied to a high-dimensional non-linear ODE model for gene regulatory networks. We see that it outperforms random experiments and works excellently for high-dimensional non-linear models for the goal of parameter estimation. Moreover, it takes into account that the whole dynamics of the system are captured properly and not only the steady states of the ODE system.

In the future it remains to test the method on real biological data and compare it with the performance of other experimental design procedures. Furthermore, maximum entropy sampling needs to be applied to other experimental design frameworks like perturbation experiments, e.g., gene knockouts or knockdowns. Focusing in this work on the parameter estimation of the underlying ODE model describing the gene regulatory network, we considered only a small network. To deal with larger networks the MCMC sampling has to be sped up, e.g., by using splines to approximate the dynamics of the system as was done for parameter estimation in [9] to avoid numerical integration.

References

1. A. C. Atkinson, A. N. Donev, and R. D. Tobias. *Optimum Experimental Designs, with SAS*, volume 34 of *Oxford Statistical Science Series*. Oxford University Press, New York, USA, 2007.
2. Kathryn Chaloner and Isabella Verdinelli. Bayesian experimental design: A review. *Statistical Science*, 10(3):273–304, 1995.
3. Ryan N. Gutenkunst, Josua J. Waterfall, Fergal P. Casey, Kevin S. Brown, Christopher R. Myers, and James P. Sethna. Universally sloppy parameter sensitivities in systems biology models. *PLoS Computational Biology*, 3(10):e189, 2007.
4. László Györfi and Edward C. van der Meulen. Density-free convergence properties of various estimators of entropy. *Computational Statistics & Data Analysis*, 5:425–436, 1987.
5. Bo Hu and Kam-Wah Tsui. Distributed evolutionary Monte Carlo for Bayesian computing. *Computational Statistics & Data Analysis*, 54(3):688–697, 2010.
6. Jeremy Kepner. *Parallel MATLAB for Multicore and Multinode Computers*. Society for Industrial Mathematics, Philadelphia, 2009.
7. Andrei Kramer and Nicole Radde. Towards experimental design using a Bayesian framework for parameter identification in dynamic intracellular network models. *Procedia Computer Science*, 1:1645–1653, 2010.
8. Johanna Mazur and Lars Kaderali. The importance and challenges of Bayesian parameter learning in systems biology. In H. G. Bock, T. Carraro, W. Jäger, S. Körkel, R. Rannacher, and J. P. Schlöder, editors, *Model Based Parameter Estimation: Theory and Applications*, volume 3 of *Contributions in Mathematical and Computational Sciences*. Springer, 2011. in press.
9. Johanna Mazur, Daniel Ritter, Gerhard Reinelt, and Lars Kaderali. Reconstructing nonlinear dynamic models of gene regulation using stochastic sampling. *BMC Bioinformatics*, 10:448, 2009.
10. A. Raue, C. Kreutz, T. Maiwald, J. Bachmann, M. Schilling, U. Klingmüller, and J. Timmer. Structural and practical identifiability analysis of partially observed dynamical models by exploiting the profile likelihood. *Bioinformatics*, 25(15):1923–1929, 2009.
11. M. C. Shewry and H. P. Wynn. Maximum entropy sampling. *Journal of Applied Statistics*, 14(2):165–170, 1987.
12. Florian Steinke, Matthias Seeger, and Koji Tsuda. Experimental design for efficient identification of gene regulatory networks using sparse Bayesian models. *BMC Systems Biology*, 1:51, 2007.
13. Vladislav Vyshemirsky and Mark Girolami. BioBayes: A software package for Bayesian inference in systems biology. *Bioinformatics*, 24(17):1933–1934, 2008.

Machine learning approaches for network-based gene prioritization from expression data

Daniela Nitsch[1] , Léon-Charles Tranchevent [1], Yves Moreau[1]

[1] Department of Electrical Engineering (ESAT-SCD), Katholieke Universiteit Leuven, 3001 Leuven, Belgium
{Daniela.Nitsch, Leon-Charles.Tranchevent, Yves.Moreau}@esat.kuleuven.be

Abstract. Discovering novel disease genes is challenging for diseases for which no prior knowledge is available. We have proposed a method that replaces prior knowledge about the biological process by experimental data. Our approach propagates the expression data over the protein-protein interaction network using distinct machine learning approaches. It relies on the assumption that strong candidate genes tend to be surrounded by many differentially expressed neighboring genes in a protein interaction network.

Keywords: gene prioritization, network analysis, random walk, ridge regression

1 Introduction

Discovering novel disease genes is challenging for diseases for which no prior knowledge - such as known disease genes or disease-related pathways - is available. Performing genetic studies frequently result in large lists of candidate genes of which only few can be followed up for further investigation. In the past couple of years, several gene prioritization methods have been proposed, such as Endeavour [1], SUSPECT [2], GeneWanderer [3], etc. They are using a guilt-by-association concept (candidate genes that are similar to the already confirmed disease genes are considered promising), and are therefore not applicable when little is known about the phenotype or when no confirmed disease genes are available beforehand.

We have proposed a method that overcomes this limitation by replacing prior knowledge about the biological process by experimental data on differential gene expression between affected and healthy individuals [4]. At the core of the method are a protein interaction network and disease-specific expression data. Candidate genes are ranked based on the differential expression of their network neighborhood. Our method relies on the assumption that strong candidate genes tend to be surrounded by many differentially expressed neighboring genes in a protein interaction network. This allows the detection of a strong signal for a candidate even if its own differential expression value is too small to be detected by a standard analysis, as long as its interacting partners are highly differentially expressed.

Recently, we have proposed a prioritization method applying different machine learning approaches that identify promising candidate genes by determining whether a gene is surrounded by highly differentially expressed genes in a protein-protein interaction network [5] which we have made freely available to the community as a web server for gene prioritization [6].

2 Methods

We have applied different machine learning approaches to prioritize candidate genes based on network analysis of differential expression to determine whether a gene is surrounded by highly differentially expressed genes in a protein-protein interaction network: initially, we have applied a kernel method, namely the Laplacian exponential diffusion kernel, aggregating the differential expression of neighbors weighted as a function of distance [4]. In a recent study [5], we have proposed advanced machine learning algorithms: first, we have smoothed a candidate gene's differential expression levels through kernel ridge regression [7-8]. Second, we have applied network diffusion by using the heat kernel algorithm [9] to our problem of disease candidate gene prioritization. Third, we have carried out network diffusion by applying the Arnoldi algorithm based on a Kyrlov Space method [10], approximating the Laplacian exponential diffusion kernel. Fourth, we have ranked the candidate genes by combining their differential expression levels with the average of the differential expression levels among their direct neighbors in a protein-protein interaction network (this straightforward approach for scoring candidates represents a naïve strategy for network analysis of differential expression).

Finally, we have implemented further random walk approaches (such as HITS with priors and k-step Markov) and made this method freely available to the community as a web server for gene prioritization, namely PINTA [6]. In doing that, we provide a large variety of machine learning approaches for gene prioritization to the user.

3 Evaluation

To assess the performance of our implemented strategies, we have set up a benchmark on 40 publicly available data sets originated from Affymetrix chips on which mice with (simple) knockout genes were tested against controls. We have preprocessed and normalized each data set using RMA [11] and computed log2 ratios as differential expression levels for each gene in the network based on the expression in the knockout experiment versus the expression in the control for each data set.

As the underlying network, we have applied a functional protein association network for mouse derived from STRING [12], a database of known and predicted protein-protein associations derived from heterogeneous data sources and different organisms including both physical interactions and functional associations.

For each data set we have selected a set of 100 candidate genes, including the knockout gene. For getting the candidates, we have chosen the knockout genes' nearest 100 genes on the chromosome which we have then prioritized.

Results were evaluated by retrieving the position of the knockout gene in the ranking list and by calculating the corresponding AUC value. Ideally, the knockout gene should appear in the top of the ranking list based on the hypothesis that this gene is causing all the disruption in the expression of the genes in the network.

4 Results

The performance was assessed against results obtained using a standard procedure in genetics (here: our baseline) that ranks candidate genes based solely on their differential expression levels. The aim was to show that our machine learning based approaches could outperform this standard procedure by ranking the knockout genes on higher ranking positions.

Strategy	**Average ranking position (out of 100)**	**AUC**	**Error reduction relative to baseline**
Baseline: standard procedure in genetics	17	83.7%	
Kernel ridge regression ranking	14	86.8%	19.0%
Heat kernel ranking	8	92.3%	52.8%
Arnoldi diffusion ranking	13	87.4%	22.7%
Average expression ranking	12	88.0%	26.4%

Table 1: Performance of distinct algorithms compared to our baseline using the STRING network and based on a benchmark consisting of 40 publicly available data sets on which mice with (simple) knockout genes were tested against controls.

Table 1 presents an overview of the performance of the ranking strategies based on the benchmark as described above. Results show that our machine learning approaches clearly outperformed our baseline, and that the best results were obtained using the network diffusion based on the heat kernel algorithm leading to an average ranking position of 8 out of 100 genes, an AUC value of 92.3% and an error reduction of 52.8% relative to our baseline which ranked the knockout gene in average at position 17 with an AUC value of 83.7% [5].

5 Conclusion

In this work, we have proposed a method that replaces prior knowledge about the biological process by experimental data, assuming that strong candidate genes tend to be surrounded by many differentially expressed neighboring genes in a protein interaction network. The results of our benchmark showed that we could identify promising candidate genes using network-based machine learning approaches even if no knowledge is available about the disease or phenotype.

References

1. Aerts S et al. Gene prioritization through genomic data fusion. Nature Biotechnology, 24:537–544, 2006.
2. Adie EA et al. Suspects: enabling fast and effective prioritization of positional candidates. Bioinformatics, 22:773–774, 2006.
3. Köhler et al. Walking the interactome for prioritization of candidate disease genes. Am J Hum Genet, 2008.
4. Nitsch D et al. Network analysis of differential expression for the identification of disease-causing genes, PLoS One 4(5), 2009.
5. Nitsch D et al. Candidate Gene Prioritization by Network Analysis of Differential Expression using Machine Learning Approaches, BMC Bioinformatics 11(460), 2010.
6. Nitsch D et al. PINTA - A web server for network-based gene prioritization from expression data, Nucleic Acids Research, first published online May 20, 2011, doi:10.1093/nar/gkr289.
7. Saunders C et al. Ridge Regression Learning Algorithm in Dual Variables. In Proceedings of the 15th International Conference on Machine Learning (ICML): 24-27 July 1998; Madison. Edited by Shavlik JW: Morgan Kaufmann Publishers; 1998.
8. Cawley GC et al. Estimating Predictive Variances with Kernel Ridge Regression. In Proceedings of Machine Learning Challenges: First PASCAL Machine Learning Challenges Workshop (MLCW): 11-13 July 2005; Southampton. Edited by Quinonero-Candela J. Dagan I, Magnini B, D'Alché-Buc F: Springer Verlag 2006:56-77.
9. Chung F et al. Coverings, heat kernels and spanning trees. Electronic Journal of Combinatorics 6, 1999.
10. Saad Y. Analysis of some Krylov subspace approximations to the matrix exponential operator. SIAM Journal on Numerical Analysis (SINUM) 29(1): 209-228, 1992.
11. Irizarry RA et al. Exploration, normalization, and summaries of high density oligonucleotide array probe level data. Biostatistics 2003, 4(2):249-64.
12. Jensen LJ et al. STRING 8 – a global view on proteins and their functional interactions in 630 organisms. Nucleic Acids Res 2009, 37 Database: D412-6.

Assessing Noise Models for Microarray Data Analysis

Alexandra Posekany[1], Klaus Felsenstein[2], and Peter Sykacek[1]

[1] Chair of Bioinformatics, University of Natural Resources and Life Sciences, Vienna
[2] Department of Statistics, Vienna University of Technology, Vienna

Abstract. Comparing Gaussian and Student-t noise models for microarray data analysis provides substantial evidence that the appropriate choice of noise distribution significantly affects data analysis and thus the biological conclusions drawn from such an analysis. An investigation of thirteen publicly available microarray experiments reveals that heavy-tailed noise provides a by far better fit than a standard Gaussian for all data sets. This observation was found independent of the chosen model organism, measurement platform and data preprocessing. We may therefore conclude that non-parametric methods or approaches which allow for heavy-tailed noise are preferred for reliably analysing microarray data.

Keywords: microarray, student's t distribution, robust analysis, Bayesian modelling, ANOVA, Markov Chain Monte Carlo, Gene Ontologies, noise model

1 Introduction

The importance of microarrays for biological research triggered the development of many analysis methods, such as t-tests, linear and probabilistic models which commonly assume Gaussian noise. Giles and Kipling [4] provided justification for this assumption by finding that microarray data follow a normal distribution. Doubt has however been cast on the Gaussian assumption with investigations having resulted in different findings. Hardin and Wilson [6] tested microarray data for normality and concluded that it does not follow a normal distribution. Evaluations by Novak et al. [8] revealed that approximately $5 - 15\%$ of the data violate the Gaussian assumption. These conflicting findings suggest a detailed investigation of the problem of including outliers properly in the analysis. Since measurements are costly and we cannot rule out that biological effects cause over-dispersion, outlying observations must be included in the analysis appropriately.

Using non-parametric statistics is the standard practice for avoiding Gaussian assumptions (cf. [3, 7]). Such approaches suffer however from low power and thus not providing significant findings, when analysing small sample sets [10]. In such situations, robust parametric approaches, for example based on a Student-t noise model [5] are suitable alternatives.

Statistical methods have gained considerable importance for machine learning approaches in the field of systems biology. High-throughput techniques,

among them microarrays, provide some of the data which computational approaches in systems biology integrate into more complex models. Therefore, understanding the possible sources of noise becomes essential in order to handle this type of data adequately.

In our work, we rely on a hierarchical Bayesian model to compare the goodness of fit of Gaussian and heavy tailed Student-t noise models in microarray data analysis. We perform inference applying a hybrid Markov chain Monte Carlo (MCMC) algorithm which "jumps" between the Gaussian and the Student-t distribution and thus directly compares the models by Bayesian means. The proposed application allows for a two step investigation: step one infers the most probable noise model and step two compares analysis results of the "optimal" noise distribution with a Gaussian alternative. This approach allows assessing the implications of unsuitable noise distributions by comparing gene and Gene Ontology (GO) term lists obtained with the different choices. We chose GOs as an example for the propagation of errors when higher-order analyses are based on incorrect gene lists. Our investigations, thus, yielded strong evidence that inappropriately chosen noise models will generate misleading leads for subsequent biological research.

2 Results

Bayesian ANOVA model with adjustable noise: To investigate whether Gaussian distributions can be used for microarray data analysis, we designed a robust Bayesian ANOVA model. The ANOVA model is based on a linear relationship between observations and the gene-wise mean expression values. To allow inference over noise characteristics, we do not fix the likelihood function, but design a model with variable noise by considering a finite class of distributions. In order to consider a wide range of characteristics, we include the Gaussian distribution and Student-t distributions with degrees of freedom ν between 1 (the extremely heavy-tailed Cauchy distribution) and a maximum ν_{max} (which is reasonably close to the Gaussian distribution).

To infer this complex model we implemented a hybrid MCMC sampler, consisting of Gibbs, Metropolis-Hastings and Reversible Jump updates. As the model's hyperparameters direct the computational inference, we conducted a detailed sensitivity analysis on artificial datasets to avoid choosing influential hyperparameters (cf. [9]). Knowing the ground truth about the sample noise, we could verify that the algorithm identified the best-fitting noise model accurately and precisely. Using artificial and spike-in data, we could also show that the algorithm's sensitivity and specificity was at least comparable to commonly applied methods (cf. [2]). Furthermore, we could demonstrate that the distance between the included degrees of freedom ν of the considered Student-t models would influence convergence behaviour of the Markov chains. Switching from a larger to a smaller model grid size during run time instead of keeping it fixed improved mixing of the chains.

Table 1. Overview of data sets. Experiments are identified by their GEO ID (CAMDA08 refers to the Endothelial Apoptosis contest data set, spike to the golden spike experiment [2]), the preprocessing applied to the data ('Preproc.'), the posterior mean degrees of freedom ('dfs') $\overline{\nu}$ we obtained for three different preprocessing methods (vsn, loess and quantile normalisation) and the noise dependencies of gene and GO term lists ('diff./common')

GEO ID	Preproc.	mean dfs ($\overline{\nu}$)			diff./common	diff./common
		vsn	loess	quant.	genes	GO terms
GDS3216	MAS5.0	5	2	1	150/1176	78/111
GDS3225	MAS5.0	6	1	1	290/832	21/161
GDS1404	PathStat	14	1	1	136/1776	14/11
GDS1686	RMA	4	3	3	174/136	96/11
CAMDA 08	CLSS4.1	4	1	1	304/400	67/26
GDS1375	MAS5.0	3	1	1	3561/6861	316/160
GDS810	MAS5.0	4	1	1	135/72	51/9
GDS2960	RPG3.0	4	3	3	166/318	2/51
GDS3221	RMA	4	3	3	119/180	52/108
GDS3162	MAS5.0	4	1	1	446/797	66/112
GDS1555	MAS5.0	4	1	1	183/131	110/24
GDS2946	MAS5.0	5	2	2	157/146	306/14
GDS972	MAS5.0	5	1	1	163/369	71/94
spike	MAS5.0	4	1	1	1748/401	-/-

Biological Consequences: For a systematic investigation, we chose the 13 data sets summarised in table 1, which cover a variety of experimental settings. To assure that our findings are independent of a chosen normalisation method (cf. [1]), we used three different preprocessing methods (vsn, loess, quantile) for our investigations. Our assessment revealed the compelling result that heavy-tailed Student-t distributions provide a better fit than the Gaussian for all the data sets we analysed (cf. table 1). Depending on the normalisation, the optimal degrees of freedom of the Student-t density were between 1 and 3 (loess, quantile) or 4 and 14 (vsn).

In addition, our investigations showed that biological inference depends substantially on the chosen noise model. To quantify the implication of unsuitably chosen noise distributions, we compared the lists of genes and Gene Ontology (GO) terms inferred conditionally on the optimal Student-t model with gene and GO term lists we obtained by relying on Gaussian noise. For vsn normalised data, we found between 119 and 3561 genes and between 14 and 316 GO terms with a noise model dependent assessment (cf. table 1). We chose Gene Ontologies as an example for a higher-order biological analysis based on microarray data or gene rankings inferred by models on the data. As already many top-ranked genes differ between the results of the approaches with different noise, the discrepancy is propagated when applying additional methods on the outcome of the models.

These numbers together with the overwhelming evidence we obtain for Student-t noise models with small degrees of freedom suggest that the Gaussian noise

model leads to a large number of wrong assessments. Figure 1 shows that among these problematic cases, we find both false positives and false negatives. We assume that false positives occur, because the Gaussian distribution incorrectly assigns differential expression due to one or a few far outlying measurements, which the Student-t model can include more appropriately. Whereas false negatives may result from grave overestimation of the variance by the Gaussian approach, while the Student-t distribution estimates a better-fitting error model. As the noise behaviour of the spike-in data is similar to the biological data sets, we can assume that the measurement process is partially responsible. However, there is no way to exclude the possibiblity that biological effects cause the observed noise behaviour. Our findings therefore provide strong evidence that Gaussian noise models are unsuitable for microarray data analysis, even if according to Novak et al. [8] only $5 - 15\%$ of genes show outlying behaviour.

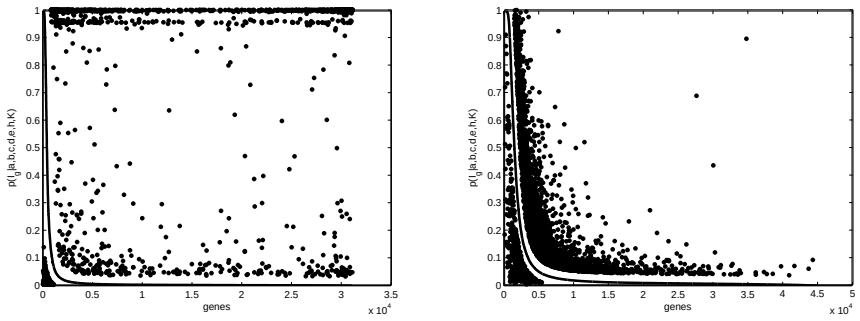


Fig. 1. Marginal posterior probabilities of genes being differentially expressed for the Gaussian and the robust noise model. The genes are ranked regarding the probabilities obtained with Gaussian noise (black line), the dots mark posterior probabilities obtained from the optimal Student-t noise model. Dots to the left of the line below the cut-off 0.85 mark false positive genes which are misclassified by the Gaussian model. Dots to the right of the line above the cut-off 0.85 are those which the Gaussian distribution would have overlooked in these data sets (false negatives). The left figure shows results for the GDS2946 data set, the right for the GDS3162 data set.

To compare the robust Student-t model with more general approaches, we inferred differentially expressed genes with two non-parametric approaches: the Kruskal-Wallis permutation test [7] and a non-parametric robust ANOVA [3]. In agreement with previous investigations (cf. [10]), we find that non-parametric approaches fail in detecting significant findings in small sample scenarios. However, if the non-parametric approach produces results, both non-parametric methods agree better with the optimal Student-t noise model ($76\% - 86\%$ shared genes) than with the Gaussian approach ($71\% - 84\%$ shared genes). We may therefore conclude that non-parametric approaches are good choices for analysing microar-

ray experiments with large sample numbers, with robust parametric methods being a more generally applicable alternative.

3 Conclusion

An investigation of the robustness levels required for microarray data analysis suggests that heavy tailed Student-t noise models or non-parametric methods should be preferred over Gaussian noise distributions. The proposed evaluation adopted a two stage strategy using a hierarchical Bayesian ANOVA model for inference. We first inferred the optimal class of noise model, which in all cases resulted to be a heavy-tailed Student-t density. A comparison of inferring biological leads, once with the optimal noise model and another time with a Gaussian density, showed a strong dependency of inferred genes and Gene Ontology terms on the chosen noise model. These findings showed not only that results differ when deviating from the optimal noise, but also that these errors are propagated when performing additional analyses. Furthermore, we found that inference results obtained with optimal Student-t noise models are in good agreement with analysis results obtained with robust non-parametric methods, as long as sample sizes permit the application of the latter methods. We may therefore conclude that microarray data analysis should avoid Gaussian noise assumptions and rather rely on non-parametric (cf. [3, 7]) or robust parametric approaches (cf. [5, 9]).

References

1. Bolstad,B., Irizarry, R., Astrand, M., Speed, T.: A comparison of normalization methods for high density oligonucleotide array data based on bias and variance. Bioinformatics, 19, 185–193. (2003)
2. Choe,S. et al. Preferred analysis methods for affymetrix genechips revealed by a wholly defined control dataset. Genome Biol., 6, R16. (2005)
3. de Haan,J., Bauerschmidt, S., van Schaik, R., Piek, E., Buydens, L., Wehrens, R. : Robust anova for microarray data. Chemometr. Intell. Lab. Syst., 98, 38–44. (2009)
4. Giles,P., Kipling,D.: Normality of oligonucleotide microarray data and implications for parametric statistical analyses. Bioinformatics, 19, 2254–2262. (2003)
5. Gottardo,R., Raftery, A., Yeung, K., Bumgarner, R.: Bayesian robust inference for differential gene expression in microarrays with multiple samples. Biometrics, 62, 10–18. (2006)
6. Hardin,J., Wilson,J.: A note on oligonucleotide expression values not being normally distributed. Biostatistics, 10, 446–450. (2009)
7. Lee,M., Whitmore G., Björbacka, H., Freeman, M.: Nonparametric methods for microarray data based on exchangeability and borrowed power. J. Biopharm. Stat., 15, 783–797. (2005)
8. Novak,J. et al.: Generalization of DNA microarray dispersion properties: microarray equivalent of t-distribution. Biol. Direct, 1, 27. (2006)
9. Posekany, A., Felsenstein, K., Sykacek, P.: Biological assessment of robust noise models in microarray data analysis. Bioinformatics, 27: 807–814. (2011)
10. Whitley,E., Ball,J.: Statistics review 6: nonparametric methods. Crit. Care, 6, 509–513. (2002)

Interaction-based Feature Selection for Predicting Cancer-Related Proteins in Protein-Protein Interaction Networks

Hossein Rahmani[1], Hendrik Blockeel[1,2], and Andreas Bender[3]

[1] Leiden Institute of Advanced Computer Science, Universiteit Leiden,
Niels Bohrweg 1, 2333 CA Leiden, The Netherlands
hrahmani@liacs.nl,blockeel@liacs.nl
[2] Department of Computer Science, Katholieke Universiteit Leuven,
Celestijnenlaan 200A, 3001 Leuven, Belgium
[3] Unilever Centre for Molecular Science Informatics,
Department of Chemistry, University of Cambridge,
Lensfield Road, Cambridge CB2 1EW, United Kingdom
ab454@cam.ac.uk

1 Introduction

The task of predicting in a protein-protein-interaction (PPI) network which proteins are involved in certain diseases, such as cancer, has received a significant amount of attention in the literature [1, 4]. Multiple approaches haven been proposed, some based on graph algorithms, some on standard machine learning approaches. Machine learning approaches such as Milenkovic et al.[5], Furney et al. [1], Li et al. [4], Furney et al. [2] and Kar et al. [3] typically use a feature-based representation of proteins as input, and their success depends strongly on the relevance of the selected features. In earlier work it has been shown that the Gene Ontology (GO) annotations of a protein have high relevance. For instance, Li et al. [4] found predictive performance to depend only slightly on the chosen machine learning method, but strongly on the chosen features, and among many features considered, GO annotations turned out to be particularly important.

In previous work, when a protein p is to be classified as disease-related or not, the GO annotations used for that prediction are usually those of p itself. In this paper, we present a new type of GO-based features. These features are based not on the GO annotation ("function") of a single protein, but on pairs of functions that occur on both sides of an edge in the PPI network. We call them *interaction-based features*.

2 Interaction-based feature selection

A PPI network is a graph where nodes are proteins and an edge between two nodes indicates that those two proteins are known to interact. In our application, proteins in the training set are also labeled as cancer-related or not (supervised learning). Additionally, each protein p is annotated with a vector $FS(p)$ that

indicates the functions that p has according to the Gene Ontology. Let $F = \{f_1, \ldots, f_{|F|}\}$ be the set of all functions in GO. $FS(p)$ is then an $|F|$-dimensional vector with $FS_i(p) = 1$ if protein p has function f_i, and $FS_i(p) = 0$ otherwise.

Several authors [1, 4] propose to use a χ^2-based feature selection method to select the most relevant GO terms. Let C and $\bar{C}$ be the set of proteins that are cancer-related (C) or not ($\bar{C}$), and let, for each f_i, P_i be the set of proteins annotated with f_i and $\bar{P}_i$ the set of proteins not annotated with it. With $a = |C \cap P_i|$, $b = |C \cap \bar{P}_i|$, $c = |\bar{C} \cap P_i|$ and $d = |\bar{C} \cap \bar{P}_i|$, we have

$$\chi^2(f_i) = \frac{(ad - bc)^2 * (a + b + c + d)}{(a + b)(c + d)(b + d)(a + c)} \quad (1)$$

Selecting individual discriminative functions based on equation 1 does not consider the network topology and the way different functions interact with each other in the network. Recent approach by Rahmani et al. [8] showed that considering Collaborative Functions: Pairs of functions that frequently interface with each other in different interacting proteins, improves the prediction of proteins functions. For the task of predicting cancer-related proteins, it is not impossible that a function f_i does not correlate itself with cancer-involvement, but when a protein with function f_i interacts with a protein with function f_j, this interaction may be an indication of the former protein being involved in a cancer.

To be able to take into account the information in the interactions, we here define new features f_{ij}. These do not describe nodes, but directed edges between nodes. Although edges in a PPI network are undirected, we can see them as pairs of directed edges. A directed edge $p \rightarrow q$ is considered positive if p is a cancer-related protein, and negative otherwise. By definition, $f_{ij}(p \rightarrow q) = 1$ if $FS_i(p) = 1$ and $FS_j(q) = 1$, and 0 otherwise. If C is the set of positive edges, $\bar{C}$ the set of negative edges, and for each feature f_{ij}, P_{ij} is the set of edges for which $f_{ij} = 1$ and $\bar{P}_{ij}$ is the set of edges for which $f_{ij} = 0$, then the χ^2 value of f_{ij} can be defined exactly as above (substituting f_{ij} and P_{ij} for f_i and P_i in the formulas for a, b, c, d and χ^2). Intuitively, an f_{ij} with high χ^2-value is relevant for the class of the protein on the i-side.

The f_{ij} features describe edges, but we need instead features that describe proteins. Therefore, we define features F_{ij} as follows: $F_{ij}(p) = \sum_q f_{ij}(p \rightarrow q)$ if $FS_i(p) = 1$, and $F_{ij}(p) = -1$ otherwise. Note that by introducing -1 as a separate value indicating that $FS_i(p) = 0$, each F_{ij} encodes implicitly the corresponding f_i feature.

In this work we compare how well cancer-involvement can be predicted from: (1) a limited number of f_i features, when those features are selected according to their χ^2 value as defined above, and (2) the same number of F_{ij} features, when those features are selected according to the following score, which combines the overall relevance of f_i, f_j, and f_{ij}:

$$score(F_{ij}) = \chi^2(f_i) + \chi^2(f_j) + \chi^2(f_{ij}).$$

In the following we will call the f_i individual-based features, and the F_{ij} interaction-based features.

3 Results

We evaluate our methods on the dataset used by Milenkovic et al. [5]. This dataset is the union of three human PPI datasets: HPRD [6], BIOGRID [9] and the dataset used by Radivojac et al. [7]. Milenkovic et al. provide details on the construction of the integrated network; some statistical information is shown in Table 1.

We divided the dataset into a training set containing 90%, and a test set containing the remaining 10%, of the proteins. We used information in the train set to select the $K(= 100, 200, 300, 400, 500)$ highest scoring individual-based, respectively interaction-based, features. Then, we described each protein in the test set based on the selected features and finally, we applied the Naive Bayes classifier for predicting cancer-related proteins.

Number of proteins	10,282
Average Degree	9.201
Min Degree	1
Max Degree	272
Number of Cancer Genes	939

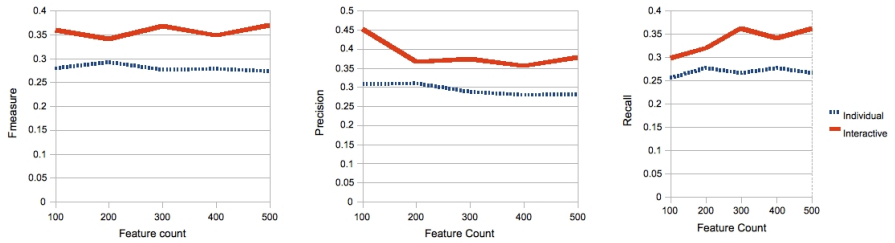
Table 1. Statistical information of union of three human PPI datasets: HPRD [6], BIOGRID [9] and Radivojac et al. [7].

Figure 1 compares our interaction-based features with the individual-based features with respect to the Fmeasure, Precision and Recall metrics. Our proposed method outperforms the individual-based method with 7.8%, on average, with respect to Fmeasure. This confirms our assumption about the usefulness of considering network interactions in feature selection. Table 2 lists five high-ranked function pairs; it shows that the functions in these pairs are not necessarily among the highest ranking functions with respect to their own χ^2.

f_i	f_j	$Rank(\chi^2(f_i))$	$Rank(\chi^2(f_j))$	$Rank(score(f_i, f_j))$
GO-0005515	GO-0003700	5	6	1
GO-0005515	GO-0007165	5	46	2
GO-0060571	GO-0001656	175	17	3
GO-0060571	GO-0001823	175	105	4
GO-0060571	GO-0050768	175	170	5

Table 2. Five high-score interactive function pairs. Function members of interactive pairs are not necessarily among the functions with high chi-score value.

What is interesting about Table 2 is that terms from two of the ontologies used, namely Molecular Function as well as Biological Process, are selected using



(a) Fmeasure (b) Precision (c) Recall

Fig. 1. Comparing interaction-based feature selection with protein-based feature selection with respect to the Fmeasure, Precision and Recall metrics. Interaction-based feature selection outperforms the protein-based method with 7.8%, on average, with respect to Fmeasure.

our feature selection method. This is the case both for pairs of terms from the same ontology, as well as for pairs of terms taken from both ontologies. More explicitly, GO terms 5515 and 3700 relate to protein amino acid binding and DNA binding transcription factor activity, and are hence related to cellular replication (first entry in Table 2). Subsequent entries have slightly different character though, such as relating protein binding (GO term 5515) to events such as signal transduction (GO term 7165), and they are hence alerting to the particular kinds of proteins that are often involved in cancer, namely kinases (such as EGFR) involved in a large number of signaling processes in the cell. It is interesting that GO terms 60571, and also 1823 and 1656 are returned by our analysis, the former relating to morphogenesis of an epithelial fold, and the latter two to different stages of kidney development. Hence, some of the terms returned can also be seen as tissue-specific as well as organ-specific, and in this way a more subtle differentiation of ontology annotations can be achieved than by using single terms alone.

4 Conclusions

Earlier work showed that Gene Ontology annotations of a protein are relevant for predicting whether it is involved in cancer. In this work we have shown that predictive accuracy can be improved significantly by combining this information with the information contained in the topology of a PPI network. Although the combination of GO-based features and features based on network topology has been considered before, the idea of attributing GO-based features to edges, rather than nodes, is novel, and is shown here to substantially improve predictive accuracy, and to identify functional interactions for which the involved functions would not normally be found relevant by themselves.

Acknowledgements

This research was funded by the Netherlands Organisation for Scientific Research (NWO) through a Vidi grant.

References

1. Simon Furney, Desmond Higgins, Christos Ouzounis, and Nuria Lopez-Bigas. Structural and functional properties of genes involved in human cancer. *BMC Genomics*, 7(1):3, 2006.
2. Simon J. Furney, Borja Calvo, Pedro Larraaga, Jose A. Lozano, and Nuria Lopez-Bigas. Prioritization of candidate cancer genesan aid to oncogenomic studies. *Nucleic Acids Research*, 36(18):e115, 2008.
3. Gozde Kar, Attila Gursoy, and Ozlem Keskin. Human cancer protein-protein interaction network: A structural perspective. *PLoS Comput Biol*, 5(12):e1000601+, December 2009.
4. Li Li, Kangyu Zhang, James Lee, Shaun Cordes, David Davis, and Zhijun Tang. Discovering cancer genes by integrating network and functional properties. *BMC Medical Genomics*, 2(1):61, 2009.
5. Tijana Milenkovic, Vesna Memisevic, Anand K. Ganesan, and Natasa Przulj. Systems-level cancer gene identification from protein interaction network topology applied to melanogenesis-related functional genomics data. *Journal of the Royal Society, Interface / the Royal Society*, 7(44):423–437, March 2010.
6. S. Peri, J. D. Navarro, T. Z. Kristiansen, R. Amanchy, V. Surendranath, B. Muthusamy, T. K. Gandhi, K. N. Chandrika, N. Deshpande, S. Suresh, B. P. Rashmi, K. Shanker, N. Padma, V. Niranjan, H. C. Harsha, N. Talreja, B. M. Vrushabendra, M. A. Ramya, A. J. Yatish, M. Joy, H. N. Shivashankar, M. P. Kavitha, M. Menezes, D. R. Choudhury, N. Ghosh, R. Saravana, S. Chandran, S. Mohan, C. K. Jonnalagadda, C. K. Prasad, C. Kumar-Sinha, K. S. Deshpande, and A. Pandey. Human protein reference database as a discovery resource for proteomics. *Nucleic Acids Res*, 32(Database issue), January 2004.
7. Predrag Radivojac, Kang Peng, Wyatt T. Clark, Brandon J. Peters, Amrita Mohan, Sean M. Boyle, and Sean D. Mooney. An integrated approach to inferring gene-disease associations in humans. *Proteins*, 72(3):1030–1037, August 2008.
8. Hossein Rahmani, Hendrik Blockeel, and Andreas Bender. Collaboration-based function prediction in protein-protein interaction networks. In Saso Dzeroski, Simon Rogers, and Guido Sanguinetti, editors, *Machine Learning in Systems Biology, Proceedings of the Fourth International Workshop, Edinburgh, Scotland, October 15-16, 2010,*, pages 55–59, 2010.
9. Chris Stark, Bobby-Joe Breitkreutz, Teresa Reguly, Lorrie Boucher, Ashton Breitkreutz, and Mike Tyers. Biogrid: a general repository for interaction datasets. *Nucleic Acids Research*, 34(Database-Issue):535–539, 2006.

Sparse Canonical Correlation Analysis for Biomarker Discovery: A Case Study in Tuberculosis

Juho Rousu[1], Daniel D. Agranoff[2], John Shawe-Taylor[3], and Delmiro Fernandez-Reyes[4]

[1] Department of Computer Science, P.O. Box 68 (Gustaf Hällströmin katu 2b), FI-00014 University of Helsinki, Finland, juho.rousu (at) cs.helsinki.fi, WWW home page: http://www.cs.helsinki.fi/group/sysfys
[2] Department of Medicine, Imperial College London, Exhibition Road London SW7 2AZ, United Kingdom, d.agranoff (at) imperial.ac.uk
[3] Department of Computer Science, University College London, Gower Street, London WC1E 6BT, United Kingdom J.Shawe-Taylor (at) cs.ucl.ac.uk
[4] MRC National Institute for Medical Research, The Ridgeway, Mill Hill, London, NW7 1AA, United Kingdom, dfernan (at) nimr.mrc.ac.uk

Abstract. Biomarker discovery from 'omics data is a challenging task due to the high dimensionality of data and the relative scarcity of samples. Here we explore the potential of canonical correlation analysis, a family of methods that finds correlated components in two views. In particular we use the recently introduced technique of sparse canonical correlation analysis that finds a projection directions that are primally sparse in one of the views and dually sparse in the other view. Our experiments show that the method is able to discover meaningful feature combinations that may have use as biomarkers for tuberculosis.

1 Introduction

In biomarker discovery from 'omics data one's aim is to find small sets of measurements that correlate with the phenotype of interest, in many cases a disease. Given the high-dimensionality and typically small sample size, one faces a challenging feature selection task, for which many approaches have been developed over the years. In supervised feature selection, the aim typically is to pick a small set of features that give a model with a high classifcation accuracy [2]. An alternative approach to crisp feature selection is to regularize the feature weights by ℓ_1-norm, which favours models with a small set of features having a non-negligible weight [8].

In this paper we consider unsupervised feature selection, where at learning time we do not posses the class (diagnostic) labels, but the data comes in two independent views, namely a proteomics expression profile, and a set of clinical data (patient history, symptoms, etc.) of controls and cases in different stages of tuberculosis. Machine learning from proteomics data has previously been shown to result in accurate diagnostic predictions in tuberculosis [1].

Our method is based on sparse canonical correlation analysis [4] that finds paired combinations of features in the two views that have good correlation in our data. In addition, to facilitate biomarker discovery, the method uses 1-norm regularization of feature weights in one (proteomics) view, while the other view is regularized in the dual representation, giving projection directions that are defined by a small number of examples.

2 Sparse canonical correlation analysis

Canonical correlation analysis (CCA) is a family of statistical method designed to situations where there are two available views (two independent sets of measurements) of the same phenomenon, and the goal is to find latent variables that explain the both views ('the generating model') [5].

Given data in two views, $\Phi_a = (\phi_a(x_i))$ and $\Phi_b = (\phi_b(x_i))$, CCA aims to find projection directions w_a and w_b that maximize the correlation of the projected data $x_a = \mathbf{w}_a^T \phi_a(x)$, and $x_b = \mathbf{w}_b^T \phi_b(x)$ in the two views,

$$\rho = \frac{\mathbf{w}_a^T C_{ab} \mathbf{w}_b}{||\mathbf{w}_a^T C_{aa} \mathbf{w}_a|| \left|\left|\mathbf{w}_b^T C_{bb} \mathbf{w}_b\right|\right|} = \frac{\alpha^T K_a K_b \beta}{\sqrt{\alpha^T K_a^2 \alpha \beta^T K_b^2 \beta}} \tag{1}$$

where the first expression gives the primal representation with explicit feature vectors, where $C_{ab} = \Phi_a \Phi_b^T$ is the empirical covariance matrix of the views over the sample. The second expression gives the dual view with implicit, kernelized representation, with $K_a = \Phi_a^T \Phi_a$ and $K_b = \Phi_b^T \Phi_b$

The basic formulation of CCA shares a property with principal component analysis (PCA) in that the projection directions are non-sparse, typically putting non-zero weight to all variables which hinders finding the most important variables. To overcome this problem, sparse variants of CCA have been developed [4, 6, 7]. In particular in the approach of [4], primal sparsity—aiming to have a small set of non-zero feature weights—is is applied to one of the views only while dual sparsity—aiming to have a small set of contributing examples—is applied to the other view. The SCCA optimization problem is given by

$$\min_{\mathbf{w},e} \left|\left|x_a^T \mathbf{w} - K_b \mathbf{e}\right|\right|^2 + \mu \left|\left|\mathbf{w}\right|\right|_1 + \gamma \left|\left|\mathbf{e}\right|\right|_1, s.t \left|\left|\mathbf{e}\right|\right|_\infty = 1 \tag{2}$$

where the first term of the objective aims to make the two views to align, the second term penalizes the feature weights in the first view by 1-norm (imposing sparsity), and the final term penalizes the dual variables by 1-norm, while the constraint $||\mathbf{e}||_\infty = 1$ ensures that at least one example will have non-zero dual coefficient. The hyperparameters μ and γ control the balance between primal and dual sparsity in the respective views.

For our biomarker application, the SCCA formulation is intuitively a good fit: we want to find a small set of proteins that correlate with clinical measurements, and ultimately with the diagnosis. The sparsity in the output view corresponds to a kind of clustering in the space of clinical profiles: a small set of mutually coherent samples are used for each model.

3 Materials and methods

Data and preprocessing. The data consists of 412 samples with three components: serum proteomics profiles measured by mass-spectrometry (270 variables), clinical data (19 variables) and diagnostic classes (Active TB, Symptomatic control, Asymptomatic control). The proteomics and clinical variables were standardized by substracting the mean and dividing by standard deviation. Then, the proteomics and clinical profiles were converted to unit length by dividing by the euclidean norm of the feature vector.

Learning parameter setup. The SCCA algorithm requires two user-defined parameters as input: a set of seed examples for output components and a scaling factor $s = \mu/\gamma$ controlling the balance between primal (input) and dual (output) sparsity. We chose the seed examples by k-means clustering of the clinical profiles and choosing the cluster centers as the seeds. The value $k = 3$ was used for number of clusters. The scaling factor was kept as its default value ($s = 1$).

Randomization. Statistical significance of the results were estimated using randomization tests. In randomization, a background data distribution consistent the null hypothesis is generated by simulation, where the statistical connection to be tested has been broken, but the data distribution is otherwise kept close to the original data. We used randomization in two tasks: assessing the statistical significance of the canonical correlation values and assessing the significance of class enrichment in the score space of the model.

4 Results

The sparse canonical correlation analysis (2) extracted a model with correlation coefficient of 0.79, which is statistically very significant (99.9% confidence level) according to the randomization test where the input and output views were randomly recoupled and the best sparse canonical correlation model was computed for the randomized data. The randomization test effectively conducts a hypothesis test where the null hypothesis is "the two views are independently generated". Thus the high confidence level strongly indicates that the views are not statistically independent. Note that the randomization setup used here also automatically corrects for a possible multiple testing bias.

Figure 1 depicts the scatter plot of the data when the input ($\phi_a(x)$) and output ($\phi_b(x)$) features of each data point ares projected to the respective SCCA components $x_a = \mathbf{w}_a^T \phi_a(x)$ (Proteomics score), and $x_b = \mathbf{w}_b^T \phi_b(x)$ (Clinical Profile Score). In addition, we have labeled the data points based on the diagnostic classes (Active, Symptomatic Control, Asymptomatic Control). By visual inspection the clusters are relatively tight and do not overlap significantly, which suggests that the SCCA model could have diagnostic value.

The statistical significance of the label enrichment was tested by randomizing the labels of the data points whilst keeping the positions of the data points intact,

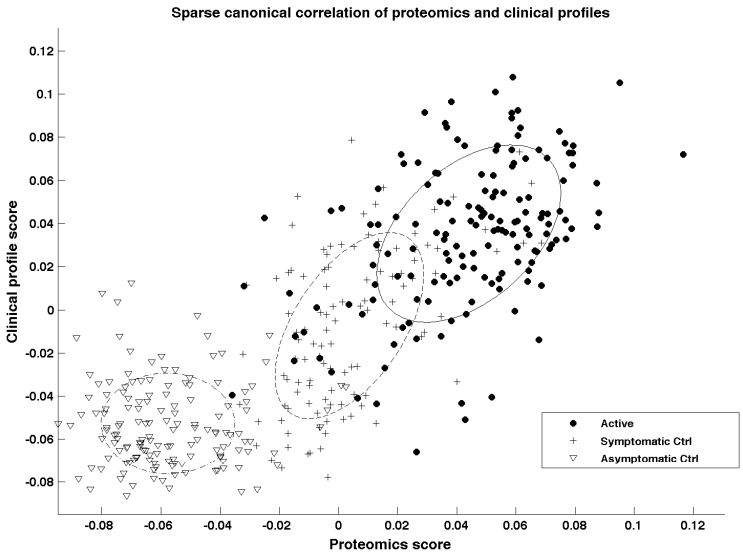


Fig. 1. Sparse canonical correlation between proteomics and clinical profiles in three classes (Active TB, Symptomatic Control, Asymptomatic Control). Ellipses denote the mean and covariance of the class clusters.

and counting for how many points the nearest neighbor has the same class label. All three class clusters were statistically very significant (p-value $< 0.1\%$).

The model included 12 variables with non-negligible coeffients selected out of 271, whilst 13 out of 18 clinical variables had non-negligible coefficients. Comparing to results of analyzing the same data with non-sparse CCA (1), we observed similar level correlation, but with a high number of non-zero proteomics weights, i.e. no feature selection effect (data not shown). Note that SCCA only enforces primal sparsity among the input variables, which is seen in our results in that significant feature selection effect is noticeable among the inputs but not among the outputs.

5 Discussion and Future Work

We analysed a set of data consisting of mass spectrometry data of serum proteome and clinical profiles. Our sparse canonical correlation analysis discovered a model with statistically very significant canonical correlation, according to a randomization test. Sparsity of the extracted model is shown by the fact that less than 4% proteomics variables had non-negligible coefficients. The clustering of three diagnostic classes was also found to be statistically significant, indicating that the set of proteomics variables indeed could function as biomarkers for tuberculosis.

Future work includes analysis of the proteomic features extracted by SCCA to get biological insight on the model. In method development, we will study the benefits of the SCCA approach over other biomarker discovery methods, in particular, the relative performance of these methods compared to SVM with recursive feature elimination [3], the LASSO regression methods i.e. sparse learning from single view containing all data [8] as well as the CCA methods [6, 7] that are sparse in both views.

Acknowledgements

This work was financially supported by Academy of Finland grant 118653 (ALGODAN), and in part by the IST Programme of the European Community, under the PASCAL2 Network of Excellence, ICT-2007-216886. This publication only reflects the authors' views.

References

1. Agranoff, D., Fernandez-Reyes, D., Papadopoulos, M., Rojas, S., Herbster, M., Loosemore, A., Tarelli, E., Sheldon, J., Schwenk, A., Pollok, R., et al.: Identification of diagnostic markers for tuberculosis by proteomic fingerprinting of serum. The Lancet 368(9540), 1012–1021 (2006)
2. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. The Journal of Machine Learning Research 3, 1157–1182 (2003)
3. Guyon, I., Weston, J., Barnhill, S., Vapnik, V.: Gene selection for cancer classification using support vector machines. Machine learning 46(1), 389–422 (2002)
4. Hardoon, D., Shawe-Taylor, J.: Sparse canonical correlation analysis. Machine Learning 83, 331–353 (2011)
5. Hardoon, D., Szedmak, S., Shawe-Taylor, J.: Canonical correlation analysis: An overview with application to learning methods. Neural Computation 16(12), 2639–2664 (2004)
6. Parkhomenko, E., Tritchler, D., Beyene, J.: Sparse canonical correlation analysis with application to genomic data integration. Statistical Applications in Genetics and Molecular Biology 8(1), 1 (2009)
7. Witten, D., Tibshirani, R., Hastie, T.: A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. Biostatistics 10(3), 515 (2009)
8. Zhu, J., Rosset, S., Hastie, T., Tibshirani, R.: 1-norm support vector machines. In: Advances in Neural Information Processing Systems 16: Proceedings of the 2003 Conference (2003)

SOM Biclustering of Gene Expression Data

Constanze Schmitt, Matthias Böck, and Stefan Kramer

Technische Universität München,
Institut für Informatik Lehrstuhl I12 - Bioinformatik,
85748 Garching b. München, Germany
{constanze.schmitt,matthias.boeck,stefan.kramer}@in.tum.de
http://wwwkramer.in.tum.de

Abstract. Self-Organising Maps (SOMs) are an unsupervised learning mechanism chiefly used for dimensionality reduction in high-dimensional data. This makes them particularly appropriate when dealing with gene expression microarray data, where they are invaluable for exploratory data analysis, such as cluster identification. The classical SOM approach performs clustering in only one dimension. However, with multiple gene expression chips describing different experimental conditions or individuals, subspace clustering is far more adapted to detect patterns of co-expressed genes present in only a subset of the samples. So far, Self-Organising Maps have been very little employed in the biclustering context. This paper describes a probabilistic extension of a SOM-Biclustering approach by Cottrel *et al.* [4] and assesses its performance with regard to both synthetic and biological data.

1 Introduction and Related Work

The analysis of gene expression data can benefit from biclustering approaches by allowing to identify local patterns of gene expression. This can be especially useful for detecting markers in different disease stages. Recently, many biclustering approaches, using several definitions of biclusters like constant, additive or multiplicative biclusters, have been described. A survey of biclustering methods is provided by Madeira *et al.* [9] and Prélic *et al.* [10]. SOMs, first introduced by Kohonen [8], are especially useful for high-dimensional data as they are very fast in computation, even without previous feature selection or filtering. They have already been successfully used for gene expression data, e.g. by Golub *et al.* [6] for the classification of cancer. But still, until now, few adaptations of Self-Organising Maps to 2-dimensional data have been proposed and little is known of their performance compared to standard biclustering-algorithms. This work's aim is to describe an extension of a SOM-based biclustering algorithm and evaluate its performance in comparison to some well-known biclustering algorithms. It is based on an approach by Cottrel *et al.* [4], who use Correspondence Analysis [1] on an input matrix to define a deterministic association of rows to columns to obtain an extended input, which is afterwards used for a slightly adapted SOM learning procedure. The proposed extension makes this association less

deterministic by allowing the algorithm to choose between the top-k association partners. In our opinion, taking only the maximum places a restriction on the algorithm as genes may be upregulated in more samples, but not necessarily in the same degree; taking the k-best generalizes the concept and thus allows bigger biclusters in the subsequent backmapping. In the following section, both the original Korresp algorithm and the extension are described.

1.1 The Korresp Algorithm

A Self-Organising Map is a usually two-dimensional grid of nodes ("cluster centers"), which each have an associated weight vector of the same dimension as the input. During learning, an input vector x is randomly sampled and mapped to the closest (in terms of a predefined distance metric) weight vector, the corresponding grid node is called the *best matching unit* (*BMU*). Node weights in the *BMU*'s neighbourhood are updated according to $w_j = w_j + \delta\ \alpha$(x-$w_j$), α being the learning rate and δ the neighbourhood radius in which updates are effected (both α and δ decrease with time). The Korresp algorithm [4] uses a preprocessing step based on Correspondance Analysis [1] to relate the two dimensions of the input and then perform SOM learning on the extended input.
Basically, Korresp takes an input matrix of dimension m×n, and extends it to $(m + n) \times (n + m)$ by first adjoining the most probable (according to Correspondance Analysis) column $c_{j|i}$ (transposed) to each row r_i and then doing the reverse for columns . The extended input matrix consists of two types of rows of the form $(r_i, {c_{j|i}}^T)$ and $(r_{i|j}, {c_j}^T)$. Learning is then done using the SOM-algorithm [8] on input vectors alternately drawn from the first and second type. Determination of the *BMU* is done with respect to r_i in the first type, but the neighbour updates use the whole weight vector; in the second row type, c_j is used for the matching to the SOM and equally, the whole vector for the neighbourhood update.

1.2 Extension of the Korresp Algorithm

In an extension of the Korresp algorithm, we propose to use not only the maximum for input connection, but also non-unique measures like top-k. This allows the algorithm to be more flexible in the learning phase by reconnecting the input in every iteration (by sampling out of the top-k) and thus admit less restricted patterns in the gene expression space.

2 Validation

2.1 Biological and Artificial Data

We studied the performance of SOM-based biclustering using the same test protocol as the comparative paper by Prélic *et al.* [10]. Validation is conducted on both synthetic and biological data. While synthetic data makes verification

easier and allows to test robustness against noise, biological data are used to assess the biological relevance of biclustering results. Synthetic data are generated in the following way: artificially implanted biclusters are constant valued and non-overlapping, and noise is added in increasing levels. A gene expression data set by Gasch *et al.* [5] (2993 genes under 173 different stress conditions in *Saccharomyces cerevisiae*) is used to perform gene set enrichment analysis as provided in the Gene Ontology Consortium [3]. For each method, the percentage of enriched biclusters is calculated. In both settings, biclusters are only evaluated in the "gene dimension", which has the advantage of available annotation (a grouping of conditions is a lot more difficult to verify, at least in the biological context) and also makes the approach comparable to one-dimensional clustering algorithms like hierarchical clustering. Three well-known biclustering algorithms were used for comparison along with the original Korresp algorithm [4]: BiMax [10],the Iterative Signature Algorithm (ISA) *et al.* [7], Cheng and Church [2], all were run with the advised parameter settings and the implementation was taken from "The Biclustering Analysis Toolboox" BicAT by Prélic *et al.* [10].

3 Results

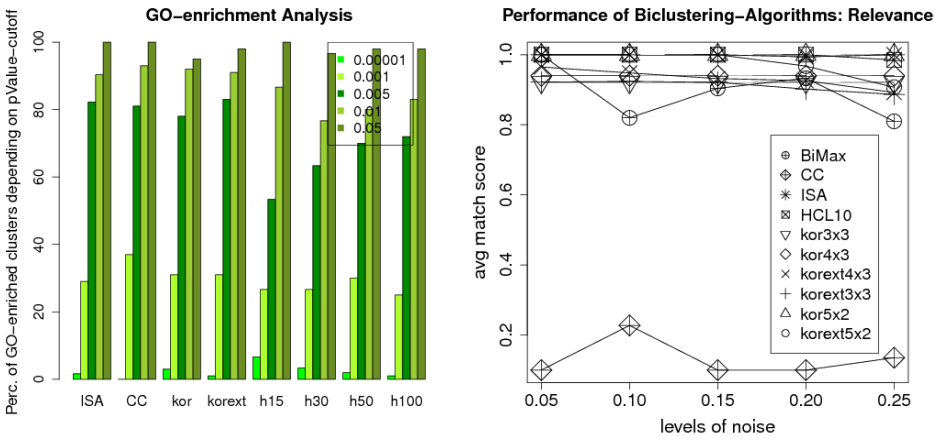


Fig. 1. Gene set-enrichment analysis based on "GO Biological Process" and evaluation on the synthetic data set

The left-hand side of the figure describes the results of the evaluation on the yeast data. Both Korresp and its extended version were run with a 10x10 grid, for the extension, k was set to 5. Results of the extended Korresp are within the range of the original algorithm. The right-hand side of the figure describes to what extent results represent true biclusters, for the score calculation, see

[10]. Both the original and extended Korresp (k=5) are run with three different grid-shapes: a 3×3, 5×2 and 4×3 grid (whose results are filtered to 10, which corresponds to the number of implanted biclusters). Korresp performs well, the extended version slightly less so, but both are dependent on the grid shape. Hierarchical clustering (HCL10) also performs well in the artificial setting, as the implanted biclusters are non-overlapping.

4 Conclusion

The application of Korresp and its extension and the comparison to other algorithms shows that Korresp shows promising results even with standard settings. It outperforms the Cheng and Church algorithm in the artificial setting and performs well compared to the other algorithms. The use of SOMs for biclustering seems to be worthwile and should be explored more deeply, as SOMs also offer the advantage of short running times and the possibility of intuitive visualization.

References

1. J.P. Benzécri. Analyse des données. In *Tome 2: Analyse des correspondances, Dunod, Paris*, 1973.
2. Y. Cheng and G. M. Church. Biclustering of Expression Data. *International Conference on Intelligent Systems for Molecular Biology*, 8:93–103, 2000.
3. The Gene Ontology Consortium. Gene Ontology: Tool for the Unification of Biology. *Nature Genetics*, 25:93–103, 2000.
4. M. Cottrell and P. Letrémy. Classification et analyse des correspondances au moyen de l' algorithme de kohonen: application à l'étude de données socio-économiques. In *Proceedings of Neuro-Nimes '94*, 1994.
5. A. P. Gasch, P. T. Spellman, C. M. Kao, O. Carmel-Harel, M. B. Eisen, G. Storz, D. Botstein, P. O. Brown, and P. A. Silver. Genomic Expression Programs in the Response of Yeast Cells to Environmental Changes. *Molecular Biology of the Cell*, 11:4241–4257, 2000.
6. T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, and C. D. Bloomfield. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, 286:531–537, 1999.
7. J. Ihmels, G. Friedlander, S. Bergmann, O. Sarig, Y. Ziv, and N. Barkai. Revealing Modular Organization in the Yeast Transcriptional Network. *Nature Genetics*, 31:370–377, 2002.
8. T. Kohonen. The Self-Organizing Map. *Proceedings of the IEEE*, 78(9), 1990.
9. S. C. Madeira and A. L. Oliveira. Biclustering algorithms for biological data analysis: a survey. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 1:24–45, 2004.
10. A. Prélic, S. Bleuler, P. Zimmermann, A. Wille, P. Bühlmann, W. Gruissem, L. Hennig, L. Thiele, and E. Zitzler. A Systematic Comparison and Evaluation of Biclustering Methods for Gene Expression Data. *Bioinformatics*, 22(9):1122–1129, 2006.

An Evolutionary Measure for Studying the Re-wiring of Protein-Protein Interactions.

Ryan Topping[1] and John W. Pinney[1]

Centre for Bioinformatics, Division of Molecular Biosciences, Imperial College London, London, United Kingdom

1 Introduction

The phenotypes of a living cell are the result of networks of interaction between molecules. Proteins are the class of cellular molecules responsible for the majority of cellular function and the majority of proteins need to interact with others to perform a function [1]. One of the central aims of systems biology is to explain how these interactions give rise to complex phenotypes. Protein interactions are gained and lost over evolutionary time and the results of these 're-wiring' events are seen in the differences in protein interactions between species. Study of the evolution of protein interactions is therefore essential for understanding the emergence of complex behaviour in biological systems [2].

Machine learning methods have already been applied in some studies of the evolution of protein interaction networks. A convenient conceptual framework is given by the *interaction tree* [3–5], in which nodes represent potential interactions between pairs of proteins and edges represent evolution between potential interactions, as shown in Fig. 1. This tree can be used as the basis for a Bayesian network, with a binary state at each node corresponding to a present or absent interaction. Then, given a suitable conditional probability function relating evolution on the interaction tree to the probability of losing or gaining an interaction, belief-propagation algorithms [6] can be used to calculate posterior probabilities for the presence of an interaction at each node in the tree, conditioned on the available data [3]. This allows the prediction of interaction states for both ancestral [3] and unobserved present day [7] protein pairs. So far, applications of interaction trees have used either uniform post-duplication probabilities of loss and gain of interaction or a simple conditional probability function based on protein sequence alone. Probabilities based on sequence evolution have proven successful in predicting interactions in the bZIP transcription factor family, whose proteins dimerise using a simple coiled-coil interface [3]. However, it is not clear whether functions based on protein sequence alone will successfully predict changes in more complex domain-domain interactions, or whether a more detailed function is required, for example taking into account the structure of the interface. This work focuses on defining a suitable conditional probability function for such inference tasks in the context of interactions between globular domains.

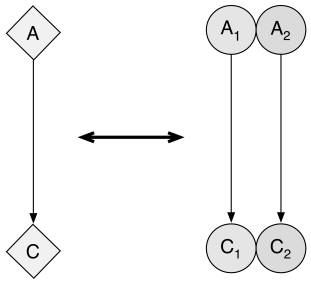


Fig. 1. Protein interaction evolution: on the left is one branch of an interaction tree showing evolution from an ancestral interaction to a child interaction, either of which may be present or absent. This is equivalent to the diagram on the right in which we consider evolution between two pairs of proteins, which may or may not interact in the ancestral and child case.

2 Methods

We consider two protein families, between which we know that protein-protein interactions are possible (i.e. protein X from family 1 has been observed to interact with protein Y from family 2). These two families span a number of species for which we have a species phylogeny. For each family we then construct a gene phylogeny, which is reconciled with the species phylogeny in order to assign ancestral genes to ancestral species. This allows us to identify every possible interaction between proteins of each family in each ancestral species, along with the evolutionary paths between possible interactions as shown in Fig. 1. We now construct an interaction tree, in which nodes represent possible interactions and edges the evolution between them. If we can then define a conditional probability function that relates evolution on this tree to the probability of gain or loss of interaction, the resulting structure is a Bayesian network that can be used to predict both ancestral and present day interactions from the observed interaction data. To define such a function we need to identify a measure of evolution along a branch of the interaction tree that is related to the gain and loss of interactions.

We consider three such measures of protein interaction evolution. Firstly, for an interaction between two protein domains we define the interface as the set of all residues from either domain that are within 4.5Å of a residue from the other domain. Then for each protein domain we define an interaction face as all residues in the interface belonging to that domain. Now we can define the first measure, the *interface distance*,

$$D_{\text{dis}}(A, C) = E(A_1, C_1) + E(A_2, C_2), \tag{1}$$

where $E(i, j)$ is the distance between protein sequences i and j, restricted to the relevant interaction face, under a Jones-Taylor-Thornton model of amino acid substitution and proteins are labelled as in Fig. 1. This sequence based measure

has been used successfully to predict interactions in a family of transcription factors [3].

The second measure, the *difference of face distances*, is then defined as

$$D_{\text{dif}}(A, C) = \left| \frac{E(A_1, C_1) - E(A_2, C_2)}{E(A_1, C_1) + E(A_2, C_2)} \right|. \tag{2}$$

This measures the similarity in branch length of the two gene phylogeny branches shown on the right hand side of Fig. 1. Use of this measure is motivated by existing methods for predicting protein interactions that rely on similarity of branch length as an indicator of interaction (e.g. [8]).

The final distance measure considered makes use of the *complementary fraction* adapted from the SCOTCH [9] method for scoring docked protein models. To calculate the complementary fraction, we first divide the 20 amino acids into 4 groups; (GLY, ALA, VAL, LEU, ILE, MET, CYS, PHE, PRO, TRP, TYR), (SER, THR, ASN, GLN), (LYS, ARG, HIS), (ASP, GLU). These are the hydrophobic, polar, positively charged and negatively charged residues respectively. We define two amino acids to be complementary if they are both hydrophobic, both polar or one positively and one negatively charged. The complementary fraction at an interface is then the fraction of contacting residue pairs that are complementary, where we allow complementarity to be maintained by nearby residues at the interface. Allowing nearby residues to account for complementarity recognises the fact that the residues responsible for maintaining an interaction can change during evolution. It has previously been shown that true protein-protein interfaces have higher complementary fraction than decoys [9], reflecting the ability of this measure to detect the effect of maintaining an interaction on the residue-residue contacts at an interface. This motivates the definition of our final distance measure,

$$D_{\text{com}}(A, B) = F(C_1, C_2) - F(A_1, A_2), \tag{3}$$

where $F(i, j)$ is the complementary fraction at the interface between protein i and protein j.

2.1 Test Data

In order to test the three measures' abilities to predict changes of interaction state, we require a test set of ancestor proteins and their extant child proteins, as shown in Fig. 1, for which we know whether either pair interacted/interacts. However, it is very difficult to construct a training set from any real world interaction trees as it is not possible to observe ancestral interactions directly. To circumvent this problem, we construct branches in which the parent and child interaction nodes represent possible present-day interactions and the branch between them represents hypothetical evolution between these interactions.

To construct a set of these hypothetical branches, we start with a set of n present-day protein pairs, for which we know that some interact and the rest do not interact. We then construct a hypothetical branch from each pair to every

other pair in turn, to generate a training set of n^2 branches for which we know the state of the 'ancestor' and 'child' interaction node.

For our initial training set, we use the interactions in the *Saccharomyces cerevisiae* 20s proteasome as extracted from the x-ray crystal structure 1RYP. This protein complex is composed of 28 subunits which consist of 14 unique single-domain protein chains (each chain appears twice), giving 196 (14^2) possible interactions, which can be paired together to construct a set of 38,416 (196^2) hypothetical paths of interaction evolution. Each of the 196 possible interaction nodes can then be assigned a state as follows: if a pair of proteins have a heavy atom from each chain within 4.5Å of each other, we assign the corresponding interaction node as present. All other interaction nodes are marked as absent. This results in a large test set of interaction tree branches for which we know whether the ancestral and child protein pairs interact. This test set has been chosen as the proteasome structure is well studied and scenarios for the evolution of the complex have been proposed. (e.g. [10])

3 Results

We test the ability of each measure to predict the interaction state of a pair of proteins, given the interaction state of their ancestral proteins and the evolution occurring between the pairs. We assume a simple threshold model for each measure to produce ROC curves, allowing comparison of the effectiveness of each in predicting interaction gain or loss, as shown in Fig. 2. We find that D_{com} outperforms the sequence based measures, as shown by the AUC statistics, particularly in predicting gains of interaction.

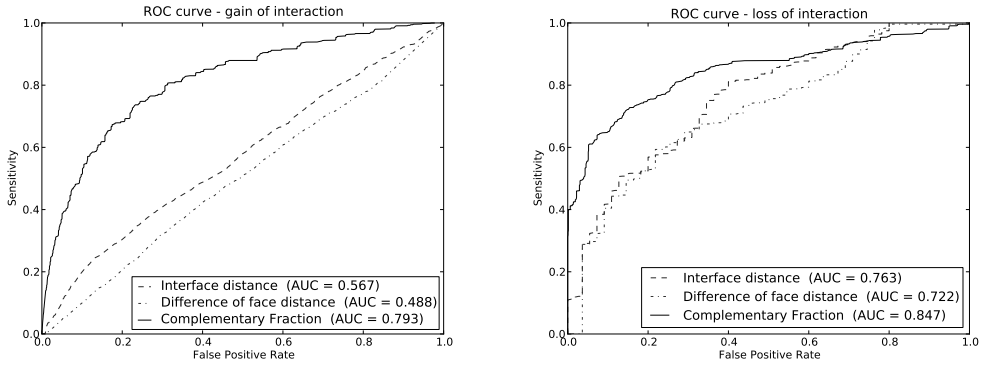


Fig. 2. ROC curves comparing the ability of the three measures to predict gains (left) and losses (right) of interaction. See legend for Area Under Curve statistics.

4 Conclusion

A measure of protein interaction evolution taking into account the structure of the interface outperforms two sequence based methods in prediction of interaction rewiring events in globular proteins. In further work, we have been able to show that this relationship is generalisable over a non-redundant set of protein complexes. This D_{com} measure is therefore suitable to be used in the construction of conditional probability functions for belief-propagation algorithms that can infer ancestral protein interactions, integrate the available protein interaction data and predict novel interactions.

References

1. Alberts, B.: The cell as a collection of protein machines: preparing the next generation of molecular biologists. Cell **92** (1998) 291–4
2. Lovell, S.C., Robertson, D.L.: An integrated view of molecular coevolution in protein-protein interactions. Molecular Biology and Evolution **27** (2010) 2567–2575
3. Pinney, J., Amoutzias, G., Rattray, M., Robertson, D.: Reconstruction of ancestral protein interaction networks for the bzip transcription factors. Proceedings of the National Academy of Sciences **104** (2007) 20449
4. Gibson, T.A., Goldberg, D.S.: Reverse engineering the evolution of protein interaction networks. Pacific Symposium on Biocomputing **14** (2009) 190–202
5. Dutkowski, J., Tiuryn, J.: Identification of functional modules from conserved ancestral protein protein interactions. Bioinformatics **23** (2007) i149–i158
6. Pearl, J.: Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kaufmann (1988)
7. Dutkowski, J., Tiuryn, J.: Phylogeny-guided interaction mapping in seven eukaryotes. BMC bioinformatics **10** (2009) 393
8. Pazos, F., Ranea, J.A.G., Juan, D., Sternberg, M.J.E.: Assessing protein coevolution in the context of the tree of life assists in the prediction of the interactome. Journal of molecular biology **352** (2005) 1002–15
9. Madaoui, H., Guerois, R.: Coevolution at protein complex interfaces can be detected by the complementarity trace with important impact for predictive docking. Proc Natl Acad Sci USA **105** (2008) 7708–13
10. Gille, C., Goede, A., Schlöetelburg, C.: A comprehensive view on proteasomal sequences: implications for the evolution of the proteasome. Journal of molecular biology **326** (2003) 1437–1448

Identification of chemogenomic features from drug-target interaction networks by sparse canonical correspondence analysis

Yoshihiro Yamanishi[1,2,3⋆], Edouard Pauwels[1,2,3], Hiroto Saigo[4], and Véronique Stoven[1,2,3]

[1]Mines ParisTech, Centre for Computational Biology, 35 rue Saint-Honore, F-77305 Fontainebleau Cedex, France, [2]Institut Curie, F-75248, Paris, France, [3]INSERM U900, F-75248, Paris, France, and [4]Department of Bioscience and Bioinformatics, Kyushu Institute of Technology, 680-4 Kawazu, Iizuka, Fukuoka, Japan

Abstract. The identification of rules governing molecular recognition between drug chemical substructures and protein functional sites is a challenging issue at many stages of the drug development. In this study we develop a novel method to extract sets of drug chemical substructures and protein domains that govern drug-target interactions on a genome-wide scale. This is made possible using sparse canonical correspondence analysis (SCCA) for analyzing drug substructure profiles and protein domain profiles simultaneously. In the results we show the usefulness of the extracted chemical substructures and protein domains for predicting new drug-target interactions and addressing the problem of ligand specificity in chemogenomics.

Keywords: chemogenomics, drug-target interaction network, feature extraction, sparsity, canonical correspondence analysis

1 Introduction

Most drugs are small chemical compounds which interfere with the biological behavior of their target proteins, therefore identification of interactions between ligand compounds and target proteins is a key area in drug discovery. A traditional approach to analyze and predict ligand-protein interactions is docking, but docking requires the information about protein 3D structures, which limits its use on a genome-wide scale. The importance of chemogenomic approach is growing fast in recent years [1], and a variety of statistical methods based on chemical and genomic information have been proposed to predict drug-target or more generally, ligand-protein interactions. Examples are support vector machine with pairwise kernels for ligand-protein pairs [2, 4], and the supervised bipartite graph inference with distance learning [6].

Ligand-protein interactions are often due to common chemical structures (the pharmacophore) that are usually shared by the ligands of a given protein. Ligand-protein interactions are also due to functional sites of proteins (e.g., domains,

⋆ to whom correspondence should be addressed.

motifs). The relevant question is how to relate ligand chemical substructures with protein functional sites in terms of ligand-protein interactions.

In this study we develop a novel method to extract sets of drug chemical substructures and protein domains that govern drug-target interactions. We develop an extension of the CCA algorithm by incorporating sparsity for easier interpretation, which we call sparse canonical correspondence analysis (SCCA). The originality of the proposed method is that it correlates protein domains to chemical substructures expected to be present in their ligands, based on a learning dataset. In other words, the method identifies pharmacophores automatically, explaining why a given molecule binds to a given protein domain.

2 Methods

2.1 Ordinary canonical correspondence analysis (OCCA)

We want to extract drug chemical substructures and protein domains which tend to jointly appear in the interaction pairs of drugs and target proteins, and to disappear in the other pairs. A possible statistical approach for achieving this goal is the canonical correspondence analysis (CCA) [3].

Suppose that we have a set of n_x drugs with p substructure features, a set of n_y target proteins with q domain features, and information about interactions between the drug set and the target protein set. Note that $n_x \neq n_y$. Each drug is represented by a p-dimensional feature vector $\mathbf{x} = (x_1, \cdots, x_p)^T$, and each target protein is represented by a q-dimensional feature vector $\mathbf{y} = (y_1, \cdots, y_q)^T$.

Consider two linear combinations for drugs and proteins as $u_i = \boldsymbol{\alpha}^T \mathbf{x}_i$ $(i = 1, 2, \cdots, n_x)$ and $v_j = \boldsymbol{\beta}^T \mathbf{y}_j$ $(j = 1, 2, \cdots, n_y)$, respectively, where $\boldsymbol{\alpha} = (\alpha_1, \cdots, \alpha_p)^T$ and $\boldsymbol{\beta} = (\beta_1, \cdots, \beta_q)^T$ are weight vectors. The goal of CCA is to find $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ which maximize the following canonical correlation coefficient:

$$corr(u, v) = \frac{\sum_{i,j} I(\mathbf{x}_i, \mathbf{y}_j) \boldsymbol{\alpha}^T \mathbf{x}_i \cdot \boldsymbol{\beta}^T \mathbf{y}_j}{\sqrt{\sum_i d_{x_i} (\boldsymbol{\alpha}^T \mathbf{x}_i)^2} \quad \sqrt{\sum_j d_{y_j} (\boldsymbol{\beta}^T \mathbf{y}_j)^2}}, \tag{1}$$

where $I(\cdot, \cdot)$ is an indicator function which returns 1 if drug $\mathbf{x}_i$ and protein $\mathbf{y}_j$ interact or 0 otherwise, d_{x_i} (resp. d_{y_j}) is the degree of $\mathbf{x}_i$ (resp. $\mathbf{y}_j$), $\sum_i u_i = 0$ (resp. $\sum_j v_j = 0$) is assumed, and u (resp. v) is called *canonical components* for $\mathbf{x}$ (resp. $\mathbf{y}$). This maximization problem can be written as follows:

$$\max\{\boldsymbol{\alpha}^T X^T A Y \boldsymbol{\beta}\} \quad \text{subject to} \quad ||\boldsymbol{\alpha}||_2^2 \leq 1, \quad ||\boldsymbol{\beta}||_2^2 \leq 1, \tag{2}$$

where $||\cdot||_2$ is L_2 norm, A is an $n_x \times n_y$ adjacency matrix A where element $(A)_{ij}$ is equal to 1 (resp. 0) if drug $\mathbf{x}_i$ and protein $\mathbf{y}_j$ are connected (resp. disconnected), and X denotes the $n_x \times p$ matrix defined as $X = [\mathbf{x}_1, \cdots, \mathbf{x}_{n_x}]^T$, and Y denotes the $n_y \times q$ matrix defined as $Y = \left[\mathbf{y}_1, \cdots, \mathbf{y}_{n_y}\right]^T$.

2.2 Sparse canonical correspondence analysis (SCCA)

In the OCCA, the weight vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are not unique if p exceeds n_x or q exceeds n_y. In addition, it is difficult to interpret the results when there are many non-zero elements in the weight vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. To impose the sparsity on $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ for easier interpretation, we propose to consider the following optimization problem with some additional L_1 penalty terms:

$$\max\{\boldsymbol{\alpha}^T X^T A Y \boldsymbol{\beta}\} \text{ subject to } ||\boldsymbol{\alpha}||_2^2 \leq 1,\ ||\boldsymbol{\beta}||_2^2 \leq 1,\ ||\boldsymbol{\alpha}||_1 \leq c_1\sqrt{p},\ ||\boldsymbol{\beta}||_1 \leq c_2\sqrt{q}, \tag{3}$$

where $||\cdot||_1$ is L_1 norm (the sum of absolute values in the vector), c_1 and c_2 are parameters to control the sparsity and restricted to ranges $0 < c_1 \leq 1$ and $0 < c_2 \leq 1$. The sparse version of CCA is referred to as sparse CCA (SCCA).

The optimization problem in SCCA can be regarded as the problem of penalized matrix decomposition of the matrix $Z = X^T AY$. Recently, a useful algorithm for solving the penalized matrix decomposition (PMD) problem has been proposed [5]. In order to obtain the solutions of SCCA, we propose to apply the PMD algorithm to the matrix $Z = X^T AY$. Here the criterion (to be maximized) is denoted as $\rho = \boldsymbol{\alpha}^T Z \boldsymbol{\beta}$ and is referred to as the singular value.

In order to obtain multiple canonical components, we propose to iterate the maximization of the above criterion repeatedly, each time using the Z matrix as the residuals obtained by subtracting from the matrix the previous factors found (deflation), that is, we recursively estimate the k-th weight vectors $\boldsymbol{\alpha}_k$ and $\boldsymbol{\beta}_k$ for $k = 1, 2, \cdots, m$. Substructures and domains with non-zero weights in each component are considered important in terms of drug-target interactions.

Here we consider predicting new drug-target interactions, based on the extracted chemical substructures and protein domains. Suppose that we are given a compound $\mathbf{x}$ and a protein $\mathbf{y}$, and we want to predict unknown interactions involving the compound and protein. We propose the following prediction score for any given pair of compound $\mathbf{x}$ and protein $\mathbf{y}$:

$$s(\mathbf{x}, \mathbf{y}) = \sum_{k=1}^{m} u_k \rho_k v_k = \sum_{k=1}^{m} \mathbf{x}^T \boldsymbol{\alpha}_k \rho_k \boldsymbol{\beta}_k^T \mathbf{y}, \tag{4}$$

where m is the number of canonical components and ρ_k is the k-th singular value. If $s(\mathbf{x}, \mathbf{y})$ is higher than a threshold, compound $\mathbf{x}$ and protein $\mathbf{y}$ are predicted to interact with each other.

3 Results and Discussion

Drug-target interactions were obtained from DrugBank, which led to build a protein-drug dataset containing 4809 interactions involving 1554 proteins and 1862 drugs. Each drug was represented by an 881 dimensional binary vector whose elements encode for the presence or absence of each PubChem substructure by 1 or 0, respectively. Each target protein was represented by a 876 dimensional binary vector whose elements encode for the presence or absence of each of the retained PFAM domain by 1 or 0, respectively.

The proposed SCCA method extracted 50 components, each of which contains a limited number of chemical substructures and protein domains. Interestingly, it successfully clusters protein domains that may be evolutionary unrelated, but that bind a common set of chemical substructures. Table 1 shows some examples of extracted chemical substructures (SMILE-like format in PubChem) and protein domains (PFAM IDs), and high scoring drugs (DrugBank IDs) and target proteins (UniProt IDs) in the first four canonical components (CCs): CC1, CC2, CC3 and CC4.

Table 1. Examples of extracted sets of protein domains, drug chemical substructures, and high scoring target proteins and drugs in canonical components 1, 2, 3 and 4

CC1	Drug substructures	CC1CC(O)CC1; CC1C(O)CCC1; saturated or aromatic carbon-only ring size 9; CC1C(C)CCC1; ...
	Protein domains	PF02159 (Oestrogen receptor); PF02155 (Glucocorticoid receptor); PF00191 (Annexin); ...
	Drugs	DB00443 (Betamethasone); DB00823 (Ethynodiol Diacetate); DB00663 (Flumethasone Pivalate)); ...
	Target proteins	ESR1_HUMAN (Estrogen receptor); GCR_HUMAN (Glucocorticoid receptor); ...
CC2	Drug substructures	SC1CC(S)CCC1; Sc1cc(S)ccc1; Sc1c(Cl)cccc1; SC1C(Cl)CCCC1; N-S-C:C; N-S; ...
	Protein domains	PF00194 (Carbonic anhydrase); PF08403; PF02254; PF03493 (potassium channel);...
	Drugs	DB00562 (benzthiazide); DB00232 (Methyclothiazide); DB01324 (Polythiazide); ...
	Target proteins	KCMA1_HUMAN (Calcium-activated potassium channel); CAH12_HUMAN (Carbonic anhydrase 12); ...
CC3	Drug substructures	C(H)(:C)(:C); C:C-C-C; C-C-C-C:C; C:C-C-C-C; C-C:C-C-C; C-C-C:C-C; ...
	Protein domains	PPF00001 (transmembrane receptor); PF03491 (Serotonin neurotransmitter transporter); ...
	Drugs	DB01654 (Thiorphan); DB00743 (gadobenic acid); DB03788 (GC-24); ...
	Target proteins	TOP2A_HUMAN (DNA topoisomerase); SC6A4_HUMAN (Sodium-dependent serotonin transporter); ...
CC4	Drug substructures	C(C)(C)(C)(C); C-C(C)(C)-C-C; unsaturated non-aromatic carbon-only ring size 6; ...
	Protein domains	PF00105 (Zinc finger); PF00104; PF02159 (Oestrogen receptor); PF00191 (Annexin); ...
	Drugs	DB00596 (halobetasol); DB01234 (Dexamethasone); DB00620 (Triamcinolone); ...
	Target proteins	ESR1_HUMAN (Estrogen receptor); GCR_HUMAN (Glucocorticoid receptor); ...

For example, Annexin domains and two ligand-binding domains of estrogen-receptors are all associated with some extracted chemical substructures in some components. The method extracted and classified Annexin-specific substructures, estrogen-receptor specific substructures and common core substructures into different components. We take the example of drug DB00823 (or PubChem ID 9270) that binds the estrogen receptor but not the annexin domains, according to the DrugBank database. The first 3 components where the estrogen receptor has a high score are CC1, CC4, and CC12. All chemical substructures with high scores in the CC1, CC4, and CC12 components are present in DB00823. Figure 1 show that the high scoring substructures of the low order components CC1 and CC4 allow to build the chemical scaffold of this drug, while those of the higher order component CC12 encode chemical groups bound to this drug's molecular scaffold.

It is difficult to evaluate the performance of a feature extraction method in a direct manner. However, if the extracted chemical substructures and proteins domains are biologically meaningful and capture relevant information with respect to protein-ligand interactions, one would expect that they present good generalization properties. We tested the ability of the method to reconstruct known drug-target interactions by performing the 5-fold cross-validation and evaluating the AUC (area under the ROC curve) score. We compared the reconstruction performance with other possible drug-target interaction prediction methods: nearest neighbor (NN), pairwise SVM (P-SVM), using the same data

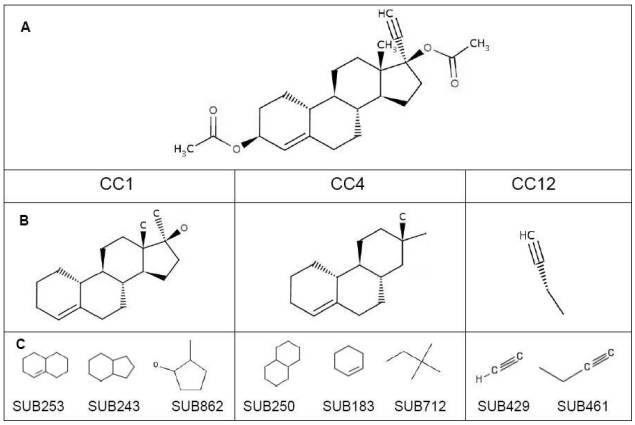


Fig. 1. An illustration of extracted drug chemical fragments. (A) Ethynodiol diacetate (DB00823). (B) Part of the molecular structure of ethynodiol diacetate that can be built using high scoring substructures of components CC1, CC4 and CC12. (C) Some high scoring substructures of components CC1, CC4 and CC12 that can be used to build the above partial structure.

descriptors, where all parameters in each method were optimized with AUC as an objective function. The resulting AUC scores for NN, P-SVM, OCCA and SCCA are 0.5892, 0.7504, 0.7377 and 0.7497, respectively. The accuracy of the proposed SCCA method was better than or close to that of other methods. It should be pointed out that NN, P-SVM and OCCA do not provide any biological interpretation since they only predict interactions, and they do not extract any information about important molecular features for these interactions.

The proposed method constitutes a contribution to the recent field of chemogenomics that aims to connect the chemical space with the biological space, and could be of interest in various ways in the drug development process.

References

1. Dobson, C.: Chemical space and biology. Nature 432, 824–828 (2004)
2. Faulon, J., Misra, M., Martin, S., Sale, K., Sapra, R.: Genome scale enzyme-metabolite and drug-target interaction predictions using the signature molecular descriptor. Bioinformatics 24, 225–233 (2008)
3. Greenacre, M.: Theory and applications of correspondence analysis. Academic Press (1984)
4. Jacob, L., Vert, J.P.: Protein-ligand interaction prediction: an improved chemogenomics approach. Bioinformatics 24, 2149–2156 (2008)
5. Witten, D., Tibshirani, R., Hastie, T.: A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. Biostatistics 10(3), 515–534 (2009)
6. Yamanishi, Y., Araki, M., Gutteridge, A., Honda, W., Kanehisa, M.: Prediction of drug-target interaction networks from the integration of chemical and genomic spaces. Bioinformatics 24, i232–i240 (2008)

A kernel based framework for cross-species candidate gene prioritization

Shi Yu[1], Léon-Charles Tranchevent[1], Sonia M. Leach[1], Bart De Moor[1], and Yves Moreau[1]

[1]Department of Electrical Engineering ESAT-SCD, Katholieke Universiteit Leuven, Leuven, Belgium.

Abstract. In biology, there is often the need to prioritize large list of candidate genes to further assay only the most promising candidate genes with respect to a biological process of interest. In the recent years, many computational approaches have been developed to tackle this problem efficiently by merging multiple genomic data sources. We present a gene prioritization method based on the use of kernel methods and prove that it outperforms our previous method based on order statistics. In addition, the method supports data integration over multiple related species. We have also developed a web based interface termed 'MerKator' that implements this strategy and proposes candidate gene prioritization for 5 species. Our cross-species approach has been benchmarked and case studies demonstrate that human prioritizations can benefit from model organism data.

1 Introduction

In modern biology, the use of high-throughput technologies allows researchers and clinicians to quickly and efficiently screen the genome in order to identify the genetic factors underlying a given disorder. However these techniques are often generating large lists of candidate genes among which only one or a few are actually associated to the disease of interest. Since the individual validation of all these candidate genes is often too costly and time consuming, only the most promising genes are experimentally assayed. This process is termed gene prioritization and several methods have been developed in the last decade to tackle that problem. Most of them combine genomic knowledge with pure experimental data to leverage the effect between reliability and novelty and rely on the 'guilt-by-association' concept.

Although numerous existing approaches are restricted to integrating information in a single species, people have recently started to collect evidence among multiple species to facilitate the prioritization of candidate genes. Chen *et al.* proposed ToppGene that performs human gene prioritization using human and mouse data [3]. Hutz *et al.* have developed CANDID, an algorithm that combines cross-species conservation measures and other genomic data sources to rank candidate genes that are relevant to complex human diseases [4]. Liu *et*

al. have investigated the effect of adjusting gene prioritization results through cross-species comparison in Drosophila [5].

We introduce MerKator, whose main feature is the cross-species prioritization through genomic data fusion over multiple data sources and multiple species. This software is developed on the Endeavour data sources [1,10] and a kernel fusion novelty detection methodology [2]. Our approach is different from previous approaches since our cross-species integration scheme is not limited to a single data source nor to a single species. At the contrary, MerKator can integrate 14 genomic data sources over 5 species (H. sapiens, R. norvegicus, M. musculus, D. melanogaster and C. elegans). We also present a benchmark analysis, through leave-one-out cross-validation, that shows the efficiency of the cross-species approach.

2 Methods

The inputs are genes from the main species (for instance human genes in our benchmark). The first input of the method is a set of training genes that will be used to model the biological process under study $\{T_{0,1}, ..., T_{0,n_0}\}$. The second input is a set of candidate genes to be prioritized $\{C_{0,1}, ..., C_{0,m_0}\}$. The single output is a ranking of these candidate genes from the most promising on top to the less promising at the bottom (based on the final score f_{cs}).

We use k species beside the main species. For each species i, we define the prioritization problem as a MKL task, each kernel is a normalized linear kernel and corresponds to a single genomic data source. The optimization task is then solved using a one class SVM (1–SVM) algorithm [8,9] as described in De Bie *et al.* [2]. Basically, the training genes are used to model the biological process under study (*i.e.*, to define the separating hyperplane), the candidate genes are then scored based on their distance to this hyperplane.

The prioritization is performed independently for each species i using the homologous genes of the training and candidate genes (respectively $\{T_{i,1}, ..., T_{i,n_i}\}$ and $\{C_{i,1}, ..., C_{i,m_i}\}$) according to the HomoloGene database [7]. The species specific scores (f_i) are first normalized to be in the range of $[0, +1]$ (with 0 being the best). They are then integrated through a Noisy–Or like model [6]. This is motivated by the fact that an excellent prioritization score obtained in one species should be enough to obtain an overall excellent score.

The following section describes in more details the computation of the score for a single candidate gene $C_{0,1}$ since each candidate is scored independently. We first denote hg() as the function that returns the HomoloGene score of two given genes. We then define the strength of the homology (h_i) as follows.

$$h_i = \min \begin{cases} \text{hg}(C_{0,1}, C_{i,1}) \\ \text{median}(\text{hg}(T_{0,1}, T_{i,1}), ..., \text{hg}(T_{0,n_0}, T_{i,n_i})) \end{cases} \quad (1)$$

The *adj* coefficient combines information from multiple species by a Noisy–Or like model. Like for f_i, a smaller *adj* value is better.

$$adj = \{\prod_{i=1}^{k}(1 - h_i(1 - f_i))\}^{\frac{1}{k}} \tag{2}$$

Ultimately, the final score f_{cs} is computed; f_0 is the score for the main species.

$$f_{\text{cs}} = \frac{f_0 * (1 + adj)}{2}. \tag{3}$$

The kernel methods underlying MerKator are usually computationally intensive. The complexity can be reduced by performing part of the computation offline (*i.e.*, kernel computation and decomposition). Similarly, kernel centering is achieved on submatrices, which reduces the computing time while keeping the estimation error reasonable.

3 Results

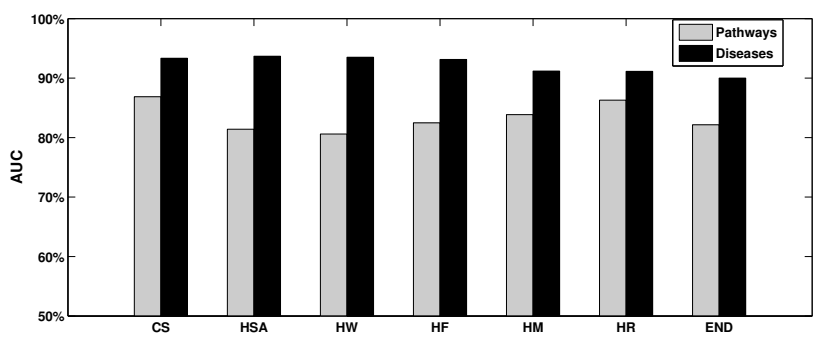


Fig. 1. Benchmark results on 14 pathways and 28 diseases. The AUC is displayed for the complete cross-species model (CS), the human only model (HSA), the two species models (HW: human and worm, HF: human and fly, HM: human and mouse, HR: human and rat), as well as for the order statistics based model (END).

As a proof of concept, we have benchmarked MerKator with 14 biological pathways and 28 diseases using a leave-one-out cross-validation procedure (described in [1]). The 14 pathway sets were derived from Gene Ontology and contain a total of 99 genes, the 28 disease sets were derived from OMIM and contain 487 genes. The cross-validation was performed using all possible data sources (Gene Ontology was excluded for the pathway benchmark). A single prioritization run results in a ranking of the candidate genes, including the position

of the left-out gene. For a disease with n_0 training genes, the result is a set of n_0 rankings. Using a hard threshold on these rankings, it is then possible to compute the sensitivity and specificity. Varying that threshold allows us to build a complete Receiver Operating Characteristic (ROC) curve. The Area Under the ROC Curve (AUC) is then used as an indicator of the performance.

For the pathway based benchmark, we obtained a global AUC of 86.87% for the cross-species model, while the model based on human data alone obtains a smaller AUC of 81.40%. In addition, the performance of every two species model is lower than the performance of our cross-species model (HW, HF, HM, and HR in Figure 1). These results do not entirely stand for the disease benchmark. In fact, the performance of the human model is already very high (93,68%), which makes improvement more difficult (93,34% for the cross-species model). This might be because the human data contain explicit disease information, which makes the identification of disease causing gene easier. In both cases, our kernel based method outperforms our method based on order statistics (END). Altogether, results indicate that our cross-species model is conceptually valid, and that combining genomic data cross-species can enhance the gene prioritization performance.

4 Conclusion

This paper presents MerKator, a software that combines cross-species information and multiple genomic data sources to prioritize candidate genes. The software is developed using the same databases adopted in Endeavour, but is equipped with a kernel fusion technique and a cross-species integration model. The issue of multiple species prioritization is complicated, which may involve many factors. It is therefore difficult to make statistical hypothesis, or estimate the data model for the final prioritization score. Our approach alternatively avoids the assumption about the data model of prioritization scores and calculates it using support vector machines. The performance of kernel-based algorithms is strongly affected by the selection of hyper-parameters, that should be determined by cross-validation, which may not be always feasible for a software oriented for biologists and medical researchers. The overall performance does however not rely on a single kernel parameter, so even when the optimal parameter is not involved, the fusion procedure can still leverage among several near optimal parameters and provides a near optimal result. For real applications, the 1% difference of performance is not so critical to the end users; the speed of solution is usually much preferred than the very optimality of the parameter or the model.

References

1. Stein Aerts, Diether Lambrechts, Sunit Maity, Peter Van Loo, Bert Coessens, Frederik De Smet, Léon-Charles Tranchevent, Bart De Moor, Peter Marynen, Bassem Hassan, Peter Carmeliet, and Yves Moreau. Gene prioritization through genomic data fusion. *Nature Biotechnology*, 24(5):537–544, May 2006. PMID: 16680138.

2. Tijl De Bie, Léon-Charles Tranchevent, Liesbeth M M van Oeffelen, and Yves Moreau. Kernel-based data fusion for gene prioritization. *Bioinformatics (Oxford, England)*, 23(13):i125–132, July 2007. PMID: 17646288.
3. Jing Chen, Huan Xu, Bruce J Aronow, and Anil G Jegga. Improved human disease candidate gene prioritization using mouse phenotype. *BMC Bioinformatics*, 8:392, 2007. PMID: 17939863.
4. Janna E Hutz, Aldi T Kraja, Howard L McLeod, and Michael A Province. CANDID: a flexible method for prioritizing candidate genes for complex human traits. *Genetic Epidemiology*, 32(8):779–790, December 2008. PMID: 18613097.
5. Qian Liu, Koby Crammer, Fernando C N Pereira, and David S Roos. Reranking candidate gene models with cross-species comparison for improved gene prediction. *BMC Bioinformatics*, 9:433, 2008. PMID: 18854050.
6. Judea Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann, 1988.
7. Eric W Sayers, Tanya Barrett, Dennis A Benson, Evan Bolton, Stephen H Bryant, Kathi Canese, Vyacheslav Chetvernin, Deanna M Church, Michael Dicuccio, Scott Federhen, Michael Feolo, Lewis Y Geer, Wolfgang Helmberg, Yuri Kapustin, David Landsman, David J Lipman, Zhiyong Lu, Thomas L Madden, Tom Madej, Donna R Maglott, Aron Marchler-Bauer, Vadim Miller, Ilene Mizrachi, James Ostell, Anna Panchenko, Kim D Pruitt, Gregory D Schuler, Edwin Sequeira, Stephen T Sherry, Martin Shumway, Karl Sirotkin, Douglas Slotta, Alexandre Souvorov, Grigory Starchenko, Tatiana A Tatusova, Lukas Wagner, Yanli Wang, W John Wilbur, Eugene Yaschenko, and Jian Ye. Database resources of the national center for biotechnology information. *Nucleic Acids Research*, 38(Database issue):D5–16, January 2010. PMID: 19910364.
8. B Schölkopf, JC Platt, J Shawe-Taylor, AJ Smola, and RC Williamson. Estimating the support of a high-dimensional distribution. *Neural Computation*, 13:1443–1471, 2001.
9. DMJ Tax and RPW Duin. Support vector domain description. *Pattern Recognition Letter*, 20:1191–1199, 1999.
10. Léon-Charles Tranchevent, Roland Barriot, Shi Yu, Steven Van Vooren, Peter Van Loo, Bert Coessens, Bart De Moor, Stein Aerts, and Yves Moreau. ENDEAVOUR update: a web resource for gene prioritization in multiple species. *Nucleic Acids Research*, 36(Web Server issue):W377–384, July 2008. PMID: 18508807.