*Sašo Džeroski*

*Simon Rogers*

*Guido Sanguinetti*

*(editors)*

# Machine Learning in Systems Biology

*Proceedings of the*

*Fourth International Workshop*

*Edinburgh, Scotland*

*October 15-16, 2010*

VIA VERITAS VITA

THE UNIVERSITY OF EDINBURGH

PASCAL2
Pattern Analysis, Statistical Modelling and Computational Learning

Sašo Džeroski, Simon Rogers and Guido Sanguinetti (Eds.)

# Machine Learning
# in
# Systems Biology

Proceedings of The Fourth International Workshop<br>October 15-16, 2010<br>Edinburgh, Scotland

## Contact Information

Postal address:

School of Informatics
10 Crichton Street
Edinburgh Eh8 9AB
United Kingdom

URL: http://www.inf.ed.ac.uk

# 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 and 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'.

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 Fourth International Workshop on Machine Learning in Systems Biology (MLSB'2010), held in Edinburgh from October 15 to 16, 2010. The workshop was organized as a core event of the PASCAL2 Network of Excellence, under the IST programme of European Union. 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.

The technical program of the workshop consisted of invited lectures, oral presentations and poster presentations. Invited lectures were given by Nir Friedman,Ursula Kummer, Hans Lehrach, Florence d'Alché-Buc, and Vebjorn Ljosa. Sixteen oral presentations were given, for which extended abstracts are included in this book: these were selected from 24 submissions, each reviewed by three members of the scientific program committee. Twenty-five poster presentations were given, for which abstracts of varying length are included here. We would like to thank all the people contributing to the technical programme, the scientific program committee, the local organizers and the sponsors for making the workshop possible.

Edinburgh, October 2010 Sašo Džeroski, Simon Rogers and Guido Sanguinetti

## Program Chairs

Sašo Džeroski (Jožef Stefan Institute, Slovenia)
Simon Rogers (University of Glasgow, UK)
Guido Sanguinetti (University of Edinburgh, UK)

## Scientific Program Committee

Nigel Burroughs (University of Warwick, UK)
Theo Damoulas (Cornell University, USA)
Werner Dubitzky (University of Ulster, UK)
Sašo Džeroski (Jožef Stefan Institute, Slovenia)
Pierre Geurts (University of Liège, Belgium)
Dirk Husmeier (Biomathematics & Statistics Scotland, UK)
Samuel Kaski (Helsinki University of Technology, Finland)
Ross D. King (Aberystwyth University, UK)
Elena Marchiori (Vrije Universiteit Amsterdam, The Netherlands)
Sach Mukherjee (University of Warwick, UK)
Mahesan Niranjan (University of Southampton, UK)
John Pinney (Imperial College London , UK)
Magnus Rattray (University of Manchester, 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 Sheffield, UK)
Ljupčo Todorovski (University of Ljubljana, Slovenia)
Koji Tsuda (Max Planck Institute, Tuebingen)
Jean-Philippe Vert (Ecole des Mines, France)
Jean-Daniel Zucker (University of Paris XIII, France)
Blaž Zupan (University of Ljubljana, Slovenia)

## Organizing Commitee

Dragi Kocev (Jožef Stefan Institute, Slovenia)
Valentin Gjorgjioski (Jožef Stefan Institute, Slovenia)
Fiona Clark (University of Edinburgh, UK)
Andrea Ocone (University of Edinburgh, UK)
Shahzad Asif (University of Edinburgh, UK)

## Sponsors

PASCAL2 Network of Excellence, Core Event;
IST programme of the European Community, contract IST-2007-216886.
University of Glasgow, U.K.
University of Edinburgh, U.K.
Jožef Stefan Institute, Slovenia

# Table of Contents

## I Invited Lectures

## II Oral Presentations: Extended Abstracts

## III Poster Presentations: Abstracts

# Part I

# Invited Lectures

# Protein-protein network inference with regularized output and input kernel methods

Florence d'Alché-Buc

Université d'Evry-Val d'Essonne, Evry, France

**Abstract.** Prediction of a physical interaction between two proteins has been addressed in the context of supervised learning, unsupervised learning and more recently, semi-supervised learning using various sources of information (genomic, phylogenetic, protein localization and function). The problem can be seen as a kernel matrix completion task if one defines a kernel that encodes similarity between proteins as nodes in a graph or alternatively, as a binary supervised classification task where inputs are pairs of proteins. In this talk, we first make a review of existing works (matrix completion, SVM for pairs, metric learning, training set expansion), identifying the relevant features of each approach. Then we define the framework of output kernel regression (OKR) that uses the kernel trick in the output feature space. After recalling the results obtained so far with tree-based output kernel regression methods, we develop a new family of methods based on Kernel Ridge Regression that benefit from the use of kernels both in the input feature space and the output feature space. The main interest of such methods is that imposing various regularization constraints still leads to closed form solutions. We show especially how such an approach allows to handle unlabeled data in a transductive setting of the network inference problem and multiple networks in a multi-task like inference problem. New results on simulated data and yeast data illustrate the talk.

# Exploring transcription regulation through cell-to-cell variability

Nir Friedman

The Hebrew University of Jerusalem, Jerusalem, Israel

**Abstract.** The regulation of cellular protein levels is a complex process involving many regulatory mechanisms. These regulatory mechanisms introduce a cascade of stochastic events leading to variability of protein levels between cells. Previous studies have shown that perturbing genes involved in transcription regulation alters variability of protein levels, but to date, there has been no systematic characterization of these effects. Here we utilize single-cell expression levels of two fluorescent reporters under a wide range of genetic perturbations in Saccharomyces cerevisiae to identify proteins that affect expression variability. We introduce computational methodology to determine the variability introduced by each perturbation, and distinguish between global variability, affecting both reporters in a coordinated manner, and local variability, affecting individual reporters independently. Classifying genes by their variability phenotype identifies functionally coherent groups, which broadly correlate with the different stages of transcriptional regulation. Specifically, we find that perturbation of processes related to DNA maintenance, chromatin regulation and RNA synthesis affect local variability, while processes related to protein synthesis and transport, cell morphology and cell size affect global variability. In addition, we find that perturbations of many processes related to chromatin regulation affect both global and local variability. Finally, we demonstrate that the variability phenotypes of different protein complexes provide insights into their cellular functions. Our methodology provides tools for examining arising data on variability, and establishes the utility of this phenotype as a tool in dissecting the regulatory mechanisms involved in gene expression.

# Computational environments for modeling biochemical networks

Ursula Kummer

BIOQUANT, University of Heidelberg, Germany

**Abstract.** Computational modeling is an integral and crucial part of systems biology. It relies on accessible and user-friendly software to set up models, model management and model analysis. Here, two systems are presented that have been implemented for these needs. The first one, COPASI, has been around since 2004 and is a standalone software suite that encompasses many of the commonly used algorithms and approaches in computational modeling. Amongst others, it allows parameter estimation of model on the basis of experimental data sets with diverse methods. The second software is SYCAMORE which is a web based application designed to allow database driven modeling. Thus, it interacts directly with databases for enzymatic kinetics and with tools to estimate parameters based on protein structural data. Both systems are constantly refined and features added.

# Deep sequencing and systems biology: steps on the way to an individualised treatment of cancer patients

Hans Lehrach

Max Plank Institute of Molecular Genetics, Berlin, Germany

**Abstract.** Biological processes are driven by complex networks of interactions between molecular and cellular components. Predicting the outcome of potential disturbances is of prime importance to be able to prevent disease, as well as to identify possible therapies for diseases, which are already present. To predict the behaviour of such complex networks, we will have to develop general models of the processes involved, based on information on pathways derived from genetic and molecular approaches, to individualise these by applying genomics scale analysis techniques (e.g. genome and/or transcriptome analysis by next-gen sequencing techniques-genomics), and to explore the behaviour of these models computationally (systems biology). We are using a combination of high throughput sequencing of genome and transcriptome of both tumor and patient to establish predictive models (virtual patients), which ultimately will reflect the response of real patients to specific therapies in oncology and other areas of medicine.

# Automatic quantification of subtle cellular phenotypes in microscopy-based high-throughput experiments

Vebjorn Ljosa

The Broad Institute of MIT and Harvard, USA

**Abstract.** Microscopy-based high-throughput experiments can provide a view into biological responses and states at the resolution of singe cells. CellProfiler, our open-source image-analysis software, has become widely used by biologists to design custom analysis pipelines for complex high-throughput assays. I will discuss our work in progress to automatically quantify the prevalence of subtle cellular phenotypes in high-throughput samples of cultured cells I will also touch briedly on the use of machine learning to improve the accuracy and robustness of CellProfiler's image segmentation. Our classification tool, CellProfiler Analyst, enables a biologist to train a boosting classifier iteratively to detect rare, complex phenotypes, and its usefulness has been demonstrated in several high-throughput screens. Here, I will describe a method to learn phenotypes without requiring hand-labeled cells for training. Instead, a classifier is trained from negative and positive controls in the experiment, where the positives are known to be enriched in the phenotype of interest, even if only slightly (e.g., 55% vs. 45% penetrance). By nonlinearly projecting cells into a random feature space, we can use efficient linear methods but still benefit from nonlinear notions of similarity, and can overcome experimental noise by training on millions of cells. Using the resulting classifier to assign soft labels to each cell in the experiment, we can identify enriched samples (”hits”) nonparametrically. Furthermore, we are developing techniques to automatically identify relevant cellular phenotypes in large-scale chemical profiling experiments.

# Part II

# Oral Presentations: Extended Abstracts

# High Throughput Network Analysis

Sumeet Agarwal[1,2], Gabriel Villar[1,2,3], and Nick S Jones[2,4,5]

[1] Systems Biology Doctoral Training Centre, University of Oxford, Oxford OX1 3QD, United Kingdom
[2] Department of Physics, University of Oxford, Oxford OX1 3PU, United Kingdom
[3] Department of Chemistry, University of Oxford, Oxford OX1 3TA, United Kingdom
[4] Oxford Centre for Integrative Systems Biology, University of Oxford, OX1 3QU, United Kingdom
[5] CABDyN Complexity Centre, University of Oxford, Oxford OX1 1HP, United Kingdom

## Introduction

Gene regulatory systems, metabolic pathways, neuronal connections, food webs, social structures and the Internet are all naturally represented as networks; indeed, this may be said of any collection of distinct, interacting entities. Sometimes the value of this mathematical abstraction is clear; for instance, to minimise the spread of an epidemic it may be important to prioritise the immunisation of individuals with high centrality. In many cases, however, one may not know beforehand how a network representation could increase ones understanding of its real-world counterpart.

It may be that abstracting a real-world system as a network discards all of the relevant information, but this seems unlikely for such a high-dimensional representation. Here, we presume that there is some valuable information encoded in the network; the problem is simply to find it. One approach for doing so is to draw a full diagram of the network, since this can, if clearly drawn, contain all of the recorded information. However, an unambiguous diagram is only feasible for very small networks, in which case it is unlikely that the mathematical abstraction will return any surprising results. To learn about a network of any significant size it is therefore necessary to characterise it by summary descriptions, which we will refer to as *metrics*.

A great variety of metrics exist in the literature, but studies that aim to characterise a particular network typically employ a small subset of these, and the choice of metrics is not systematic. Similarly, when a new model for generating synthetic networks is presented, the synthetic networks are compared to real networks in only a few characteristics. This may be justified if one is interested only in the behaviour of a particular metric; but if the goal is to develop synthetic networks that are statistically indistinguishable from real networks, it is important to look at these networks in as many ways as possible. The same is true of exploratory network analysis. Finally, it is typical for a new metric to be introduced with a comparison to only a few existing metrics. The lack of a systematic comparison makes it difficult to tell which metrics give genuinely novel information about a network, and which pairs of metrics might be redundant or complementary.

Efforts to address this have recently been made [2], but it remains true that there is as yet no systematic program for characterising network structure [7] that can be used to compare the diverse ways in which networks are analysed. We introduce a more systematic framework, in the form of a matrix whose rows correspond to networks, and columns to metrics; we term this the *data matrix*. Each element of the data matrix contains the value of one metric as applied to one network. In this paper we show that this framework enables the systematic comparison of networks and metrics, and demonstrate its utility in the objective selection of metrics for a given purpose; in model fitting; in the analysis of evolving networks; and to determine the robustness of metrics to variations in network size, network damage and sampling effects.

### Networks

We collected approximately 1,200 real network data sets. These included several types of biological networks (such as trophic, brain connectivity, protein interaction and metabolic networks), social networks, computer networks and miscellaneous others (including word adjacency and transportation networks). In addition to these real networks, we generated synthetic networks from the Erdős-Rényi, Watts-Strogatz, Barabási-Albert, fitness and graphlet arrival models.

### Metrics

This study included approximately 60 base metrics taken from the literature. In order to obtain single numbers from metrics that return distributions (over nodes or links), we generated a number of summary statistics of these distributions, including measures of central tendency and skewness and also likelihoods of certain model fits. Additionally, we include graph clustering or community detection [3, 8] metrics, which return a partition of the network into subnetworks. We then summarise this in a number of ways, such as computing partition entropy and coarse-grained measures on the network of subnetworks.

## Selected Results

Given that a large number of metrics exist for describing a network, selecting appropriate subsets for particular tasks is important. Here we demonstrate two applications of feature selection in a supervised learning setting.

First, we consider two sets of networks from a study on metabolic networks [4]. The first set consists of 43 networks that each represent the full cellular network of an organism. The networks in the second set are subsets of the first, including only the metabolic part of each of the 43 networks. We used this classified network data to investigate how metabolic networks differ from whole-cellular networks. We performed sequential feature selection to optimise the linear discriminability between the metabolic networks from eukaryotes, archaea and bacteria. A 95% classification success rate was obtained by using just three metrics (Figure 1).

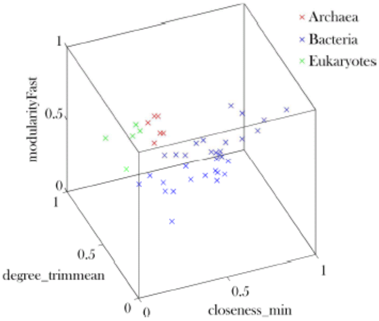
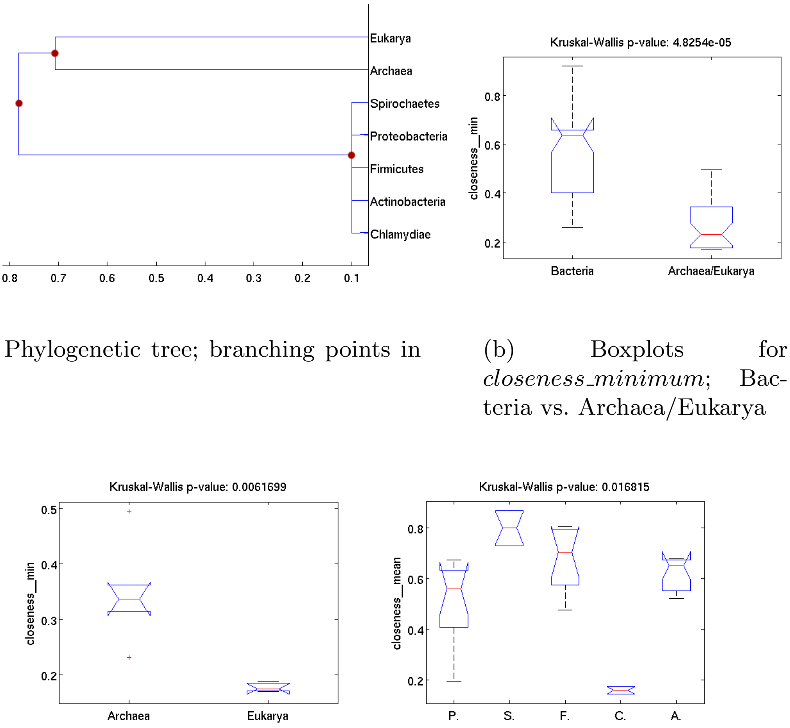


**Fig. 1.** Classification of metabolic networks of organisms from different kingdoms.

A natural extension of this approach is to look not only at a particular level of species classification, but instead to attempt to take into account the entire structure of evolutionary relationships between species, as represented by a phylogenetic tree. We are currently working on this using ideas from the area of phylogenetic comparative methods [1, 5, 6]: one can assume a certain statistical process (e.g., Brownian motion) underlying the variation in network characteristics along the branches of a phylogeny, and then estimate the extent to which different characteristics are constrained by the phylogenetic structure. As a rough preliminary step towards this, we have taken the 43 metabolic networks referred to above and grouped them at the leaves of a highly simplified phylogeny (Figure 2(a)). We represent each network by its feature vector of metrics, and then carry out feature selection based on information gain at each of the branching points in the phylogeny. Figure 2 shows that features based on *closeness*, a measure of node centrality, are found to be amongst the most informative ones at each of the 3 branching points. This suggests that this metric is capturing some biologically relevant network property, and it should be of interest to study this in greater detail using the approach described above.

As an example to demonstrate unsupervised learning on more varied data, we took a set of 192 networks from a wide range of disciplines and carried out principal component analysis (PCA), utilising a set of 433 metrics. The results are shown in Figure 3, with each data point representing a network's position along the two largest principal components and different colours depicting the different domains from which the networks are drawn. We see that certain kinds of networks fall into very cohesive groupings, such as financial, fungal and metabolic networks. On the other hand, some types of networks such as protein inter-



(a) Phylogenetic tree; branching points in red

(b) Boxplots for *closeness_minimum*; Bacteria vs. Archaea/Eukarya

(c) Boxplots for *closeness_minimum*; Archaea vs. Eukarya

(d) Boxplots for *closeness_mean*; for the 5 Bacterial phyla

**Fig. 2.** 43 metabolic networks [4] are grouped according to a simplified phylogeny **(a)**. Network features representing the closeness distribution of nodes are found to be significantly different in their distributions on either side of the 3 branching points **(b,c,d)**.

action, collaboration and social networks are much less well separated. We also attempted building a supervised classification tree for this set of networks, which resulted in a 10-fold cross-validation accuracy of nearly 80% and made use of only about 15 of the 433 features.

## Discussion

In some ways, the approach taken here is complementary to standard perspectives in network science. When a new metric is introduced in the networks lit-

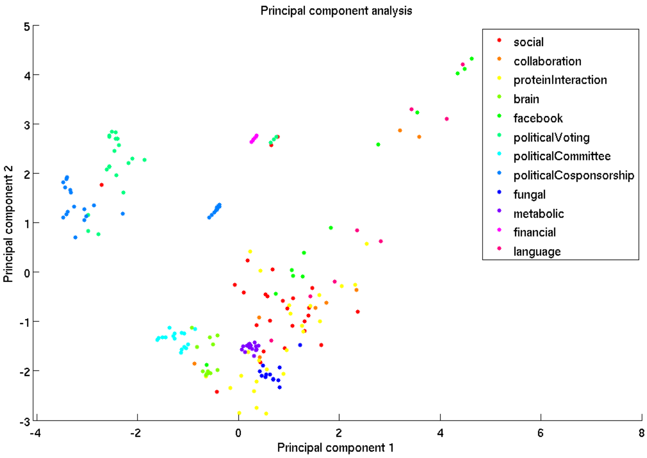


**Fig. 3.** Results of PCA on a set of 192 networks, using 433 features. The two largest principal components are shown.

erature, it may be motivated by an expectation of what aspects of a network it will capture, or by some distinguishing feature of its calculation. Similarly, new network models are assessed by how closely they match certain particular metrics. Here, we simply apply all of the available metrics to a set of networks, and use the resulting data structure to explore the networks or metrics in an unprejudiced manner. This framework as a way of systematically comparing metrics should be valuable for both explorative network analysis, and for finding the best way to answer a particular question in a data-driven manner. It continues to be work in progress, but we hope that once complete, public distribution of the software and database built for this project will benefit users and see new applications of the framework.

## References

1. Felsenstein, J.: Phylogenies and the comparative method. The American Naturalist 125(1), 1–15 (January 1985), http://dx.doi.org/10.1086/284325
2. Filkov, V., Saul, Z.M., Roy, S., D'Souza, R.M., Devanbu, P.T.: Modeling and verifying a broad array of network properties. EPL (Europhysics Letters) 86(2), 28003 (April 2009), http://dx.doi.org/10.1209/0295-5075/86/28003
3. Fortunato, S.: Community detection in graphs. Physics Reports 486(3-5), 75–174 (2010), http://www.sciencedirect.com/science/article/B6TVP-4XPYXF1-1/2/99061fac6435db4343b2374d26e64ac1
4. Jeong, H., Tombor, B., Albert, R., Oltvai, Z.N., Barabasi, A.L.: The large-scale organization of metabolic networks. Nature 407(6804), 651–654 (October 2000), http://dx.doi.org/10.1038/35036627

5. Macholán, M.: The mouse skull as a source of morphometric data for phylogeny inference. Zoologischer Anzeiger 247(4), 315–327 (October 2008), http://dx.doi.org/10.1016/j.jcz.2008.06.001
6. Martins, E.P.: Estimating the rate of phenotypic evolution from comparative data. The American Naturalist 144(2), 193–209 (August 1994), http://dx.doi.org/10.1086/285670
7. Newman, M.E.J.: The structure and function of complex networks. SIAM Review 45(2), 167–256 (2003), http://link.aip.org/link/?SIR/45/167/1
8. Porter, M.A., Onnela J-P, Mucha, P.J.: Communities in networks. Notices of the American Mathematical Society 56(9), 1082–1097, 1164–1166 (September 2009), http://arxiv.org/abs/0902.3788

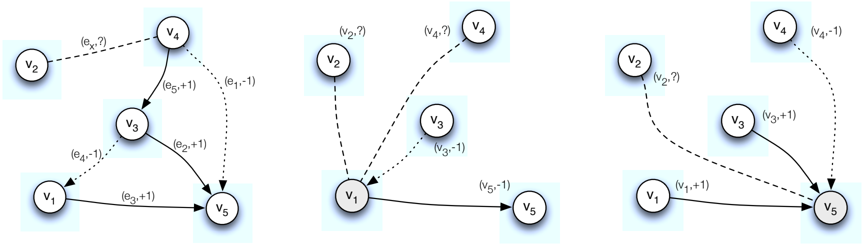
# On Learning Gene Regulatory Networks with Only Positive Examples

Luigi Cerulo and Michele Ceccarelli

DSBA, University of Sannio, Benevento, Italy
Institute of Genetic Research “Gaetano Salvatore”, Ariano Irpino (AV), Italy
{lcerulo, ceccarelli}@unisannio.it

**Abstract.** Learning with positive only examples occurs when the training set of a binary classifier is composed of examples known to be positive, and examples where the label category is unknown. Such a condition largely affects the task of learning gene regulatory networks as biologists does not aware the information whether two genes does not interact. We introduce the problem of learning new gene–gene interactions from positive and unlabeled data and propose a roadmap of possible approches.

A recent trend in computational biology aims at using supervised approaches to reconstruct large biological networks from genomic data [9]. In this paper we focus on gene regulatory networks, the network of transcription dependences among genes, known as *transcription factors*, and their binding site. A gene regulatory network is modeled as a directed graph where vertices, $V = (v_1, v_2, v_3, \ldots, v_n)$, represent genes of an organism and edges their interactions. Each vertex is assumed to have a description in term of a feature vector $\phi(v) \in \mathrm{R}^n$, where $\phi(v)$ is a vector of expression levels of gene $v$ in a set of $p$ different DNA microarray experimental conditions. The problem is to reconstruct the set of edges, $E \subset V \times V$, that represent the interactions among genes. Such a problem can be formalized as a binary classification problem [9, 5], which is widely studied in machine learning [10]. It requires a training set of edge examples, $T = \{(\phi(e_1), l_1), (\phi(e_2), l_2), \ldots, (\phi(e_N), l_N)\}$, where $\phi(e_i) \in \mathrm{R}^n$ is an n-dimesional feature vector of the edge $e_i \in E$, and, $l_i \in \{-1, +1\}$, is a binary label representing the information that the pair of genes belonging to the edge interacts $(+1)$ or not $(-1)$. The goal is to infer a function $f(e_x) : \mathrm{R}^n \rightarrow \{-1, +1\}$ that is able to predict the binary label of any new edge $e_x \in \mathrm{R}^n$. In such a learning scheme the feature vector of an edge, $e_{ij} = (v_i, v_j)$, is built from the feature vector of its gene pair components $\phi(e_{ij}) = \psi(\phi(v_i), \phi(v_j))$. The basic principle is to use the natural inductive reasoning to predict new regulations: if a gene $v_1$ having expression profile $\phi(v_1)$ is known to regulate a gene $v_2$ with expression profile $\phi(v_2)$, then all other couples of genes $v_i$ and $v_j$, having respectively expression profiles similar to $\phi(v_1)$ and $\phi(v_2)$ are likely to interact in a similar manner. Different approaches have been proposed to build ad edge feature vector, as for example vector concatenation, direct product, and tensor product [9]. Despite the completeness of such a *global* learning scheme, in real applications, it reveals expensive in term of representation dimensionality and computational time. An alternative, based on *local* models, has been proposed to ovecome such a limitation [9]. It consists to partition the original problem into $n$ sub–problems, one for each vertex in the graph. A local classifier is built for each vertex in the graph to discriminate the vertices that are connected and those that are non connected to it. The final list of predicted edge can be obtained by combining the edge predicted by each vertex classifier. The main advantage is that the dimensionality of each classifier is drastically reduced because vertex feature vectors, $\phi(v_i)$, are used in place of edge feature vectors, $\phi(e_i)$. Hovewer, with local models there is no way to predict connection between free vertex, those with no known connections, because no training data is available for those vertices.



(a) global model (b) vertex $v_1$ local model (c) vertex $v_5$ local model

**Fig. 1.** Global and local models in learning gene regulatory networks

Figure 1 summarizes both global and local learning scheme of a gene regulatory network. Several learning algorithms have been proposed in literature, neural networks, decision tree, logistic models, and Support Vector Machines (SVM) [10]. Among all SVM gave promising results in the reconstruction of biological networks [7, 1, 11]. A crucial point in a binary classifier is that it needs both positive and negative examples to learn effectively. This condition does not hold in the context of gene regulatory network as biologists not aware the information whether two genes does not interact. Databases, such as RegulonDB (http://regulondb.ccg.unam.mx), report only whether a gene regulate another gene, not the contrary. Thus the problem of learning gene regulatory neworks fall into the problem of learning with positive and unlabeled data as the overall dataset is composed by two types of data: positive examples, i.e. known gene–gene interactions, and unlabeled examples which could be both positive and negative. The goal is to predict the unknown gene–gene interactions in the unlabeled data. In literature can be distinguished approaches that depends on a starting selection of reliable negative examples [12]; and approaches that does not need labeled negative examples and basically tries to adjust the probability of being positive estimated by a traditional classifier trained with labeled and unlabeled examples [3]. We focus in particular on a class of approaches aiming at selecting reliable negatives from the unlabeled set in order to have a two–class training set for a binary classifier. The main problem is that some of the selected negative examples could in fact be positives embedded in the unlabeled data and then affect negatively the binary classifier. The key success of such an approach is to generate a sufficiently large negative training set without positive contamination. We experimented the extend to which a negative selection heuristic could improve the performance of an SVM classifier by assuming an ideal heuristic that is able to select candidate negatives with a prefixed fraction of positive contamination $\lambda \in [0, 1]$. We simulated such an heuristic with a local learning scheme on the experimental data of *Escherichia coli* made publicly available by [4], consisting of 445 different microarray experimental conditions for 4345 genes and 3293 experimentally confirmed regulations between 154 transcription factors and 1211 genes (RegulonDB (version 5) [8]).

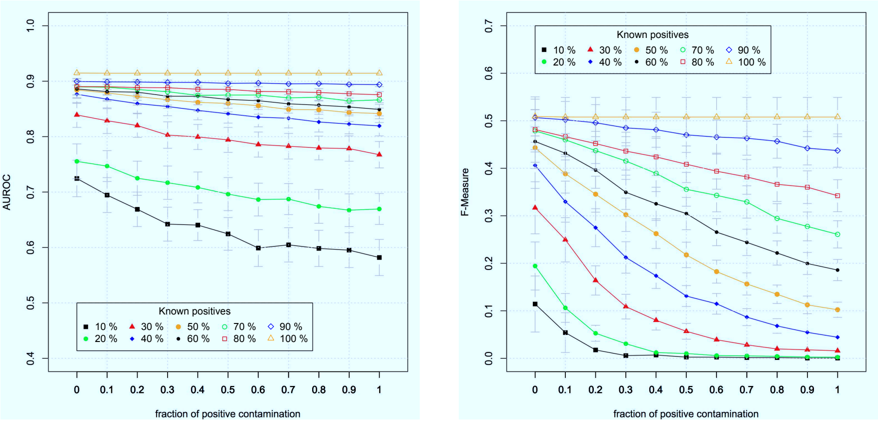


**Fig. 2.** The performances of an SVM classifier (95% confidence interval) trained with a set contamined with a fraction of positives assumed as negatives.

Figure 2 shows the results of such an experiment. The SVM classifier is trained with a percentage of known positive examples and unlabeled data assumed as negative examples but contamined with a fraction of positive examples. Results can be interpreted as an upper bound for a negative selection heuristic. The figure shows two measures of prediction accuracy, AUROC (Area Under the ROC curve) and F-Measure, obtained with a ten fold cross validation. Both performance measures grow when the percentage of known positives increases. This is expectable as when more positives are known less unlabeled examples could be positive. Instead, the effect of positive contamination on AUROC and F-Measure is similar but with different effect size: F-Measure decreases quickly when the fractions of positive contamination increases; while, AUROC significantly decreases when the percentage of known positives is low. We believe that this is an encouraging result to investigate for new negative selection heuristics that could improve the quality of the training set of a binary classifier. In [2] we proposed a method that selects negative examples by exploiting the known network topology. It is based over the assumption that a regulatory network has no or few cycles and candidate negatives could be those given by the union of the transitive closure of the known network Another approach could exploit the over presence of network motifs, i.e. feed-forward loops, bi-fan clusters, and single input modules. Network motifs are small connected subnetworks that a network exhibits in significantly higher occurrences than would be expected just by chance in a network with the same number of edges [6]. Therefore, candidate negative edges could be those that may affect the overpresence of certain motifs, if assumed as positive interactions.

## References

1. J. R. Bock and D. A. Gough. Predicting protein-protein interactions from primary structure. *Bioinformatics*, 17(5):455–460, 2001.
2. M. Ceccarelli and L. Cerulo. Selection of negative examples in learning gene regulatory networks. In *Bioinformatics and Biomedicine Workshop, 2009. BIBMW 2009. IEEE International Conference on*, pages 56–61, Nov. 2009.
3. L. Cerulo, C. Elkan, and M. Ceccarelli. Learning gene regulatory networks from only positive and unlabeled data. *BMC Bioinformatics*, 11(1):228, 2010.
4. J. J. Faith, B. Hayete, J. T. Thaden, I. Mogno, J. Wierzbowski, G. Cottarel, S. Kasif, J. J. Collins, and T. S. Gardner. Large-scale mapping and validation of Escherichia coli transcriptional regulation from a compendium of expression profiles. *PLoS Biol*, 5(1):e8, 01 2007.
5. M. Grzegorczyk, D. Husmeier, and A. V. Werhli. Reverse engineering gene regulatory networks with various machine learning methods. *Analysis of Microarray Data*, 2008.
6. R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon1. Network motifs: Simple building blocks of complex networks. *Science*, 298(5594):824–827, October 2002.
7. F. Mordelet and J.-P. Vert. SIRENE: supervised inference of regulatory networks. *Bioinformatics*, 24(16):i76–82, 2008.
8. H. Salgado and et al. Regulondb (version 5.0): Escherichia coli k-12 transcriptional regulatory network, operon organization, and growth conditions. *Nucleic Acids Res*, 34(Database issue), January 2006.
9. J.-P. Vert. *Reconstruction of Biological Networks by Supervised Machine Learning Approaches*, pages 163–188. Wiley, 2010.
10. I. H. Witten and E. Frank. *Data mining : practical machine learning tools and techniques.* Morgan Kaufmann series in data management systems. Morgan Kaufman, June 2005.
11. Y. Yamanishi, J.-P. Vert, and M. Kanehisa. Supervised enzyme network inference from the integration of genomic data and chemical information. *Bioinformatics*, 21:468–477, 2005.
12. H. Yu, J. Han, and K. C. chuan Chang. Pebl: Web page classification without negative examples. *IEEE Transactions on Knowledge and Data Engineering*, 16:70–81, 2004.

# An Integrated Generative and Discriminative Bayesian Model for Binary Classification

Keith Harris and Mark Girolami

Inference Group, Department of Computing Science, University of Glasgow, UK
{keithh,girolami}@dcs.gla.ac.uk

## 1 Introduction

High dimensional data sets typically consist of several thousand covariates and a much smaller number of samples. Analysing such data is statistically challenging, as the covariates are highly correlated, which results in unstable parameter estimates and inaccurate prediction. To alleviate this problem, we have developed a statistical model which uses a small number of meta-covariates inferred from the data through a Gaussian mixture model, rather than all the original covariates, to classify samples via a probit regression model. A graphical overview of our model is presented in Figure 1 below.

The novelty of our approach is that our meta-covariates are formed considering predictor-outcome correlations as well as inter-predictor correlations. This idea was partly inspired by recent empirical research that has shown that optimum predictive performance often corresponds to an intermediate trade-off between the purely generative and purely discriminative approaches to classification [2]. The main advantage over using a sparse classification model [1] is that we can extract a much larger subset of covariates with essential predictive power and partition this subset into groups, within which the covariates are similar.

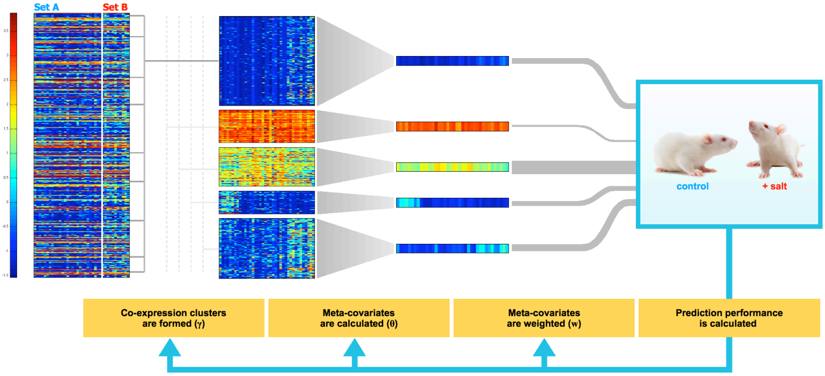

Set A
Set B
3.5
3
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
control
+ salt
Co-expression clusters are formed (γ)
Meta-covariates are calculated (θ)
Meta-covariates are weighted (w)
Prediction performance is calculated

**Fig. 1.** The meta-covariate method applied to gene expression data. Co-expression clusters are identified and represented by a cluster mean. Each cluster mean is assigned a weight according to its ability to distinguish between set A and set B data. Predictive performance is used to iteratively update the clustering structure and the weights.

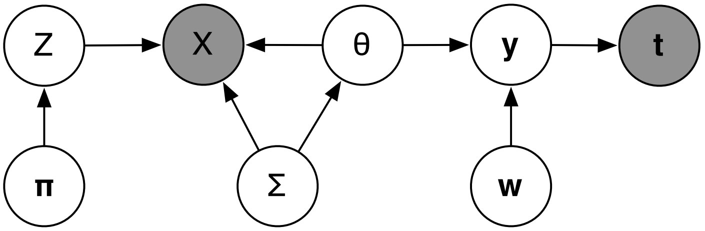


**Fig. 2.** Graphical representation of the conditional dependencies within the meta-covariate binary classification model.

## 2 Model details

In the following discussion, we will denote the $N \times D$ design matrix as $X = [\mathbf{x}_1, \ldots, \mathbf{x}_D]$ and the $N \times 1$ vector of associated response values as $\mathbf{t}$ where each element $t_n \in \{0, 1\}$. The $K \times N$ matrix of clustering mean parameters $\theta_{kn}$ is denoted by $\theta$, the $K \times N$ matrix of clustering variance parameters $\sigma^2_{kn}$ by $\Sigma$ and the $K \times 1$ vector of mixing coefficients $\pi_k$ by $\boldsymbol{\pi}$. We represent the $K \times 1$-dimensional columns of $\theta$ by $\boldsymbol{\theta}_n$ and the corresponding $N \times 1$-dimensional rows of $\theta$ by $\boldsymbol{\theta}_k$. The $D \times K$ matrix of clustering latent variables $z_{dk}$ is represented as $Z$. The $K \times 1$ vector of regression coefficients $w_k$ is denoted by $\mathbf{w}$. Finally, we denote the $N \times 1$ vector of classification auxiliary variables $y_n$ by $\mathbf{y}$.

The graphical representation of the conditional dependency structure in the meta-covariate classification model is shown in Figure 2. From Figure 2 we see that the joint distribution of the meta-covariate classification model is given by

$$p(\mathbf{t}, \mathbf{y}, X, Z, \boldsymbol{\pi}, \theta, \Sigma, \mathbf{w}) = p(\mathbf{t}, \mathbf{y}|\theta, \mathbf{w})p(X, Z|\boldsymbol{\pi}, \theta, \Sigma)p(\boldsymbol{\pi})p(\theta|\Sigma)p(\Sigma)p(\mathbf{w}).$$

The distribution $p(X, Z|\boldsymbol{\pi}, \theta, \Sigma)$ is the likelihood contribution from our clustering model, which we chose to be a normal mixture model with unequal weights and diagonal covariance matrices, that is,

$$p(X, Z|\boldsymbol{\pi}, \theta, \Sigma) = \prod_{d=1}^{D} \prod_{k=1}^{K} \pi_k^{z_{dk}} \mathcal{N}(\mathbf{x}_d|\boldsymbol{\theta}_k, \Sigma_k)^{z_{dk}},$$

where $\Sigma_k = \operatorname{diag}(\sigma^2_{k1}, \ldots, \sigma^2_{kN})$. Similarly, $p(\mathbf{t}, \mathbf{y}|\boldsymbol{\theta}, \mathbf{w})$ is the likelihood contribution from our classification model, which we chose to be a binary probit regression model whose covariates are the means of each cluster, that is, $\boldsymbol{\theta}_k$, $k = 1, \ldots, K$. Thus,

$$p(\mathbf{t}, \mathbf{y}|\boldsymbol{\theta}, \mathbf{w}) = \prod_{n=1}^{N} p(t_n|y_n)p(y_n|\boldsymbol{\theta}_n, \mathbf{w}),$$

where

$$p(t_n|y_n) = \begin{cases} \delta(y_n > 0) & \text{if } t_n = 1 \\ \delta(y_n \leq 0) & \text{otherwise} \end{cases} \text{ and } p(y_n|\boldsymbol{\theta}_n, \mathbf{w}) = \mathcal{N}(y_n|\mathbf{w}^T\boldsymbol{\theta}_n, 1).$$

Finally, the model was completed by specifying a uniform prior for $\boldsymbol{\pi}$, vague inverse Gamma priors for $\sigma^2_{kn}$, and vague normal priors for $\theta$ and $\mathbf{w}$. Thus,

$$p(\theta|\Sigma) = \prod_{k=1}^{K} \mathcal{N}(\boldsymbol{\theta}_k|\boldsymbol{\theta}_0, h\Sigma_k), \quad p(\Sigma) = \prod_{k=1}^{K}\prod_{n=1}^{N} \text{Inv-Gamma}\left(\sigma^2_{kn}\middle|\, \nu, \xi\right),$$

and

$$p(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mathbf{0}, lI),$$

where the hyperparameters $\boldsymbol{\theta}_0$, $h$, $\nu$, $\xi$ and $l$ are chosen such that weak prior information is specified.

## 3 Summary of our inference approach

Given the number of clusters $K$, we would like to infer the full posterior distribution of the parameters. In our previously published research we derived an EM algorithm that allowed us to maximise the joint distribution with respect to the parameters and successfully applied this algorithm to a renal gene expression dataset in a rat model of salt-sensitive hypertension [4]. Here, we use this EM algorithm to initialise the following Gibbs sampler:

1. Sample $\boldsymbol{\pi}$ from Dirichlet $(D_1 + 1, \cdots, D_K + 1)$, where $D_k = \sum_{d=1}^{D} z_{dk}$.
2. Sample $\theta_{kn}$ from $\mathcal{N}((e_n w_k + m_{kn})v_{kn}, v_{kn})$, where:

$$e_n = y_n - \sum_{k' \neq k} w_{k'}\theta_{k'n}, \; m_{kn} = \frac{1}{\sigma^2_{kn}}\left(\sum_{d=1}^{D} z_{dk}x_{nd} + \frac{\theta_{0n}}{h}\right),$$

and

$$v_{kn} = \left[w_k^2 + \frac{1}{\sigma^2_{kn}}\left(D_k + \frac{1}{h}\right)\right]^{-1}.$$

3. Sample $\sigma^2_{kn}$ from:

$$\text{Inv-Gamma}\left(\frac{1}{2}D_k + \nu + \frac{1}{2}, \frac{1}{2}\sum_{d=1}^{D} z_{dk}(x_{nd} - \theta_{kn})^2 + \frac{1}{2h}(\theta_{kn} - \theta_{0n})^2 + \xi\right). \tag{1}$$

4. Sample $\mathbf{w}$ from:

$$\mathcal{N}\left(\left(\theta\theta^T + l^{-1}I\right)^{-1}\theta\mathbf{y}, \left(\theta\theta^T + l^{-1}I\right)^{-1}\right).$$

Note that the first component of $\mathbf{w}$ is set to 1, so that the model is identifiable.
5. Sample $\mathbf{z}_d$ from Multinomial$(n_{\text{trials}}, p_1, \ldots, p_K)$, where $n_{\text{trials}} = 1$ and

$$p_k = E(z_{dk}) = \frac{\pi_k\left(\prod_n \sigma^2_{kn}\right)^{-1/2} \exp\left\{-\frac{1}{2}\sum_n \frac{(x_{nd}-\theta_{kn})^2}{\sigma^2_{kn}}\right\}}{\sum_j \pi_j\left(\prod_n \sigma^2_{jn}\right)^{-1/2} \exp\left\{-\frac{1}{2}\sum_n \frac{(x_{nd}-\theta_{jn})^2}{\sigma^2_{jn}}\right\}}.$$

6. Sample $y_n$ from:

$$p(y_n|\mathbf{y}_{-n}, \boldsymbol{\pi}, \theta, \Sigma, \mathbf{w}, \mathbf{t}, X, Z) \propto \begin{cases} \delta(y_n > 0)\mathcal{N}(y_n|\mathbf{w}^T\boldsymbol{\theta}_n, 1) & \text{if } t_n = 1 \\ \delta(y_n \leq 0)\mathcal{N}(y_n|\mathbf{w}^T\boldsymbol{\theta}_n, 1) & \text{otherwise.} \end{cases}$$

7. We obtain the predictive classification of a new observation $t^*$, conditioning on the test point $\mathbf{x}^*$, using the Monte-Carlo estimate:

$$P(t^* = 1|\mathbf{x}^*, \mathbf{t}, X) \approx \frac{1}{I}\sum_{i=1}^{I} \Phi(\mathbf{w}_i^T\boldsymbol{\theta}_i^*),$$

where $\mathbf{w}_i$ and $\boldsymbol{\theta}_i^*$ are the MCMC samples of $\mathbf{w}$ and $\boldsymbol{\theta}^*$ from their full conditional distributions. Thus, we also need to sample $\theta_k^*$ from $\mathcal{N}(m_k^* v_k^*, v_k^*)$, where:

$$m_k^* = \frac{1}{\sigma_k^{*2}}\left(\sum_{d=1}^{D} z_{dk}x_d^* + \frac{\theta_0^*}{h}\right) \text{ and } v_k^* = \left[\frac{1}{\sigma_k^{*2}}\left(D_k + \frac{1}{h}\right)\right]^{-1},$$

and sample $\sigma_k^{*2}$ from equation (1).

## 4 Application of gene selection

We apply our method to a publicly available breast cancer dataset [3] from patients carrying mutations in the predisposing genes, BRCA1 or BRCA2, and from patients not expected to carry either of these hereditary predisposing mutations. This dataset contains 22 breast tumour samples: 7 BRCA1, 8 BRCA2 and 7 sporadic. There are 3,226 genes for each tumour sample. We use our method to classify BRCA1 versus the others and compare our method to a Bayesian sparse probit regression model [1]. We run the Gibbs samplers of both methods for 100,000 iterations and discard the first half of each chain as burn-in. We compared the methods using leave-one-out cross validation. Our results indicate that our Gibbs sampling approach of inferring meta-covariates in classification has competitive performance with Bayesian sparse probit regression.

## References

1. Bae, K., Mallick, B.K.: Gene selection using a two-level hierarchical Bayesian model. Bioinformatics 20(18), 3423–3430 (July 2004)
2. Bishop, C.M., Lasserre, J.: Generative or discriminative? Getting the best of both worlds. In: Bayesian Statistics. vol. 8, pp. 3–24. Oxford University Press (June 2007)
3. Hedenfalk, I., Duggan, D., Chen, Y.D., Radmacher, M., Bittner, M., Simon, R., Meltzer, P., Gusterson, B., Esteller, M., Kallioniemi, O.P., Wilfond, B., Borg, A., Trent, J.: Gene-expression profiles in hereditary breast cancer. New England Journal of Medicine 344(8), 539–548 (February 2001)
4. Hopcroft, L.E.M., McBride, M.W., Harris, K.J., Sampson, A., McClure, J.D., Graham, D., Young, G., Holyoake, T.L., Girolami, M.A., Dominiczak, A.F.: Predictive response-relevant clustering of expression data provides insights into disease processes. Nucleic Acid Research (June 2010)

# On the stability and interpretability of prognosis signatures in breast cancer

Anne-Claire Haury and Jean-Philippe Vert

Mines ParisTech - CBIO / Institut Curie / INSERM U900, Paris, France

## 1 Introduction

In recent years, several genome-wide expression profiles studies lead to the identification of prognosis molecular signatures to predict breast cancer outcome from gene expression measurements [5, 2, 10, 11, 9]. These signatures, which are meant to have direct bearing on the therapy choice, typically consist in a few tens of genes which have been selected by various feature selection methods. In addition to their predictive power, the selection of a few prognosis genes may lead to the identification of new therapeutic targets and the elucidation of biological pathways involved in metastatic progression. However, it has been observed that signatures obtained from different studies show very low overlap, raising questions on the capacity of these methods to retrieve biologically relevant genes and processes.

In this work we wish to answer the questions: (i) how much can we trust the list of genes and the biological functions found in a predictive signature, and (ii) how do common feature selection methods compare to each other in this regard? We propose a rigorous framework to assess the accuracy, the stability, and the interpretability of a feature selection method, and compare 8 common feature selection methods as well as ensemble feature selection variants on three breast cancer datasets. Results highlight the very low robustness of most existing methods, including ensemble methods, and raise a warning about the over-interpretation of published signatures in terms of genes and biological processes.

## 2 Feature selection methods

We compared the 8 feature selection methods listed below, which span a wide range of approaches in feature selection [6]. For each of them, we can control the number of feature selected by a method-specific parameter.

**Filters.** T-test, Chernoff Bound, Wilcoxon rank-sum test, KL divergence
**Wrappers.** Greedy forward selection, SVM RFE
**Embedded.** Lasso, Elastic Net

In addition to these "single-run" feature selection methods, we consider for each of them an "ensemble" feature selection variant defined as follows. We bootstrap the original dataset $B$ times, and apply the single-run feature selection method on each bootstrap sample to get $B$ rankings $(r^1...r^B)$ of all genes. We then compute, for each gene $j$, the score $S_j = \sum_{b=1}^{B} \exp(-r_j^b/50)$, where $r_j^b$ is the rank of gene $j$ for bootstrap $b$, and rank the genes by decreasing score. A signature of size $L$ is obtained by taking the top $L$ genes in this list. Intuitively, a gene that often appears in the top 50 genes for a given bootstrap sample will have a good final score. Ensemble feature selection were recently proposed to improve the performance and stability of feature selection methods [1, 7], and we wish here to systematically compare single-run and ensemble methods.

## 3 Evaluating a feature selection method

We borrowed 3 breast cancer expression datasets from public repositories : the *Van de Vijver* dataset (24,496 genes, 295 samples, 101 metastatic) [10], the *Wang* dataset (22,215 genes, 286 samples, 107 metastatic) [11], and the *Sotiriou* dataset (7,650 genes, 99 samples, 45 metastatic). The processing of these sets was the one of the original studies.

For each dataset, we evaluate the accuracy and stability of genes and biological functions by $k$-fold cross validation, for $k = 2, 5, 10, 20$ : the samples of each dataset are split in $k$ non-overlapping groups, and $k$ signatures are estimated by $k-1$ groups by leaving apart each group in turn. We set a fixed number of 50 boostrap replications when ensemble methods were used.

**Accuracy.** The accuracy measures how well a signature trained on $k-1$ groups predicts the metastatic status of samples in the $k$-th group, as measured by the balanced accuracy (1/2(sensitivity + specificity)) of a nearest centroid classifier trained on the signature genes [10]. Although embedded and wrapper feature selection methods produce a classifier with their signatures, we checked that the performance of the nearest centroid classifier trained on the signature was not significantly different from the performance of the native classifier. The balanced accuracy is averaged over the $k$ folds.

**Stability.** The similarity between two signatures $S_1$ and $S_2$ is computed with the Tanimoto coefficient $T(S_1, S_2) = \frac{|S_1 \cap S_2|}{|S_1 \cup S_2|}$ which ranges between 0 (signatures have no gene in common) to 1 (signatures are the same). The stability of a feature selection method is defined as the average Tanimoto similarity between the $k(k-1)/2$ pairs of signatures.

**Interpretatability.** For a given signature, we build a biological interpretation by extracting the list of significant Gene Ontology (GO) terms corresponding to biological processes, at a false discovery rate of 5% for a hypergeometrical test with correction for multiple test [3]. We compare the biological interpretation of two signatures by the Tanimoto coefficient of the corresponding lists of GO terms, and assess the intepretation stability of a method as the average Tanimoto similarity between the $k(k-1)/2$ pairs of signatures.

While $k$-fold cross-validation is a well-established method for accuracy estimation, it must be pointed out that it is positively biased for stability estimation for $k > 2$. More precisely, for a total of $n$ samples two different folds of $(k-1)n/k$ samples each have on average $(k-2)n/k$ samples in common. The case $k = 2$ corresponds to a "hard" perturbation, where we assess the similarity of two signatures estimated on completely different samples (but within a given study). The cases $k > 2$ correspond to a "soft perturbation", where we observe how robust a signature is when only a few samples change. Although the later is often used to assess stability [1], the former (k=2) is more adequate to quantify the ability of a signature to capture some intrinsic biological information.

## 4 Results and discussion

In this section we summarize some of the main findings of this study. First, the predictive balanced accuracy is overall in the range $60 - 70\%$ for most methods in all datasets, confirming the difficulty to estimate very precise molecular signatures for breast cancer prognosis [8]. Second, no method performs significantly better than the random drawing of a given number of genes (when at least 10 genes are selected), as already observed by [4]. From these results, it is clear that accuracy should not be the only criterion to use in order to prefer a signature. Moreover, we did not notice any significant effect of the signature size on accuracy, in the range $10 \sim 100$ genes. Figure 1 ranks the methods for a signature of 100 genes: an arrow from method 1 to method 2 means that method 1 performs significantly better than method 2 (Wilcoxon signed-rank test across folds, at 5% significance level).

Regarding stability, we observe overall that filter methods outperform wrapper and embedded methods (Figure 2). However, while the Tanimoto coefficient can be in the range $40 \sim 50\%$ in 10-fold cross-validation, it decreases when less overlap exist between the training samples (Figure 1) and drops to very small values (at most 5%) in 2-fold cross-validation, i.e., when signatures are estimated on non-overlapping samples.

Surprisingly, ensemble feature selection methods barely improve the situation. While we observe like [1] that the stability of SVM RFE (as well as that of GFS) significantly benefits from this technique, other methods do not, and the stability of SVM RFE with ensemble feature selection remains much below that of simpler filter methods (Figure 1 and 2).

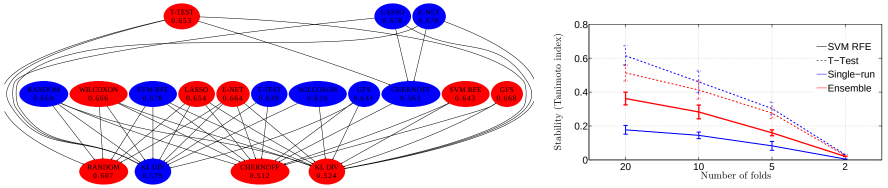


Fig. 1: Van't Veer dataset, signature of size 100. Left : Comparison of performance. An arrow means 'significantly more performant than'. Blue nodes refer to single-run algorithms, red nodes to ensemble methods. Right: Stability of SVM RFE and T-test (single-run and ensemble) in $k$-fold cross-validation, for different $k$.

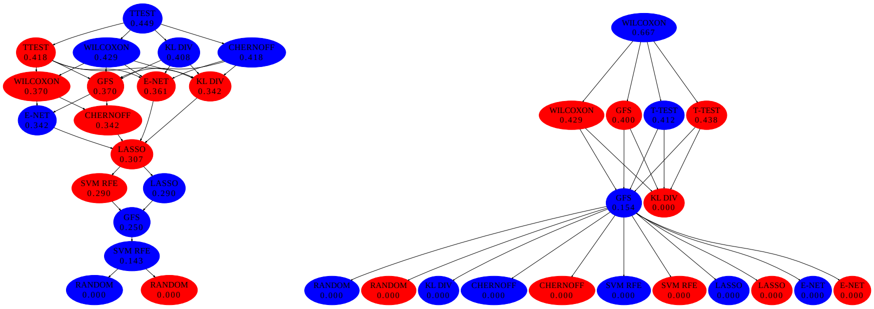


Fig. 2: Van't Veer dataset, signature of size 100. Left : Comparison of algorithms stability. Right : Stability of the interpretability, i.e. annotation stability.

Finally, the stability of biological interpretation across methods follows a pattern similar to that of gene stability (Figure 2). While it is worth noting that there are many ways to obtain a biological interpretation that could yield different results, e.g. other types of pathways or testing procedures, it appears again that, overall, univariate filters do best, that stability values remain low both on soft and hard perturbation settings, and again that ensemble methods do not bring clear benefits.

In conclusion, in this empirical study focused on breast cancer prognosis, simple filter methods provide the best accuracy/stability compromise, outperforming more complex wrapper or embedded methods. Although wrapper and embedded methods benefit from the ensemble setting, they do not outperform simple filters. The stability of the best methods trained on non-overlapping training samples remains not significantly better than random gene selection, both when genes and annotations are considered. This highlights the difficulty to extract potential drug targets and biological insight from most existing signatures, and calls for new methods to capture biological information from gene expression data in a robust manner.

## References

[1] T. Abeel et al. *Bioinformatics*, 26(3):392, 2009.
[2] A. A. Alizadeh et al. *Nature*, 403(6769):503–511, 2000.
[3] Y. Benjamini and Y. Hochberg. *J. R. Stat. Soc. Ser. B*, 57(1):289–300, 1995.
[4] L. Ein-Dor et al. *Bioinformatics*, 21(2):171, 2005.
[5] T. R. Golub et al. *Science*, 286(5439):531, 1999.
[6] I. Guyon and A. Elisseeff. *J. Mach. Learn. Res.*, 3:1157–1182, 2003.
[7] N. Meinshausen and P. Buhlmann. *Preprint, arXiv*, 809, 2009.
[8] S. Michiels et al. *The Lancet*, 365(9458):488–492, 2005.
[9] C. Sotiriou et al. *Proc. Natl. Acad. Sci. USA*, 100(18):10393–10398, 2003.
[10] M. J. van de Vijver et al. *N. Engl. J. Med.*, 347(25):1999–2009, 2002.
[11] Y. Wang et al. *The Lancet*, 365(9460):671–679, 2005.

# Inferring Regulatory Networks from Expression Data using Tree-based Methods

Vân Anh Huynh-Thu[1,2], Alexandre Irrthum[1,2], Louis Wehenkel[1,2], and Pierre Geurts[1,2]

[1] Department of Electrical Engineering and Computer Science, Systems and Modeling, University of Liège, Liège, Belgium
[2] GIGA-Research, Bioinformatics and Modeling, University of Liège, Liège, Belgium

## 1 Introduction

One of the pressing open problems of computational systems biology is the elucidation of the topology of genetic regulatory networks (GRNs) using high throughput genomic data, in particular microarray gene expression data. The Dialogue for Reverse Engineering Assessments and Methods (DREAM) challenge aims to evaluate the success of GRN inference algorithms on benchmarks of simulated data [11–13]. In this article, we present a new algorithm for the inference of GRNs that was best performer in the DREAM4 *In Silico Multifactorial* challenge[3]. In addition, we show that the algorithm compares favorably with existing algorithms to decipher the genetic regulatory network of *Escherichia coli*. It doesn't make any assumption about the nature of gene regulation, can deal with combinatorial and non-linear interactions, produces directed GRNs, and is fast and scalable.

An extended version of this works appears in [7]. Our software is freely available from `http://www.montefiore.ulg.ac.be/~huynh-thu/software.html`.

### 1.1 Network Inference with Tree-based methods

Our GRN inference algorithm decomposes the prediction of a regulatory network between $p$ genes into $p$ different regression problems. In each of the regression problems, the expression pattern of one of the genes (target gene) is predicted from the expression patterns of all the other genes (input genes), using tree-based ensemble methods Random Forests [1] or Extra-Trees [6].

One of the most interesting characteristics of tree-based methods is that it is possible to compute from a tree a variable importance measure that allows to rank the input features according to their relevance for predicting the target. Several variable importance measures have been proposed in the literature for tree-based methods. In our experiment, we consider a measure which at each test node $\mathcal{N}$ computes the total reduction of the variance of the output variable due to the split, defined by [2]:

$$I(\mathcal{N}) = \#S\mathrm{Var}(S) - \#S_t\mathrm{Var}(S_t) - \#S_f\mathrm{Var}(S_f), \qquad (1)$$

[3] `http://wiki.c2b2.columbia.edu/dream09/index.php/D4c2`

where $S$ denotes the set of samples that reach node $\mathcal{N}$, $S_t$ (resp. $S_f$) denotes its subset for which the test is true (resp. false), Var(.) is the variance of the output variable in a subset, and # denotes the cardinality of a set of samples. For a single tree, the overall importance of one variable is then computed by summing the $I$ values of all tree nodes where this variable is used to split. For an ensemble of trees, the importances are averaged over all individual trees.

We thus exploit the embedded feature ranking mechanism of a tree-based ensemble method to solve each of the $p$ regression problem. The importance of an input gene in the prediction of the target gene expression pattern is taken as an indication of a putative regulatory link. The $p$ individual gene rankings are then aggregated to get a global ranking of all putative regulatory links.

## 2 Results

### 2.1 Results on the DREAM4 multifactorial data

We report here our results on the DREAM4 competition, where one challenge concerned the inference of five *in silico* regulatory network of $p = 100$ genes, each from multifactorial perturbation data. Multifactorial data are defined as static steady-state expression profiles resulting from slight perturbations of all genes simultaneously. In total, the number of expression profiles for each network was set to 100.

We took part in this challenge and submitted the rankings obtained by our procedure using the Random Forests algorithm as tree-based method. Among twelve challengers, our tree-based procedure got the highest areas under the precision-recall curve (AUPR) and the receiver operating characteristic curve (AUROC) on all networks. Table 1 shows the AUPR and AUROC values of our predictions (RF) and those of the first runner-up (2nd best) as a comparison.

**Table 1.** AUPR and AUROC scores for DREAM4 Multifactorial challenge.

| | Method | NET1 | NET2 | NET3 | NET4 | NET5 |
|---|---|---|---|---|---|---|
| AUPR | RF | 0.154 | 0.155 | 0.231 | 0.208 | 0.197 |
| | 2nd best | 0.108 | 0.147 | 0.185 | 0.161 | 0.111 |
| AUROC | RF | 0.745 | 0.733 | 0.775 | 0.791 | 0.798 |
| | 2nd best | 0.739 | 0.694 | 0.748 | 0.736 | 0.745 |

### 2.2 Performance on *Escherichia coli* dataset

In addition, we carried out experiments with our method on the inference of the regulatory network of *Escherichia coli*, which has been used by several authors as a benchmark. The dataset of expression profiles we used was retrieved from the Many Microbe Microarrays ($\mathrm{M}^{3\mathrm{D}}$) database [3] (version 4 build 6). It contains

907 *E. coli* microarray expression profiles of 4297 genes collected from different experiments at steady-state level. To validate the network predictions we used 3433 experimentally confirmed regulatory interactions among 1471 genes that have been curated in RegulonDB version 6.4 [5].

We adopted the same evaluation protocol as in [4] that assumes that we have prior knowledge about which genes of the gold standard (i.e. the experimentally confirmed interactions curated in RegulonDB) are transcription factors. Figure 1 compares our procedure using Random Forests with three methods, CLR [4], ARACNE [9], and MRNET [10], using exactly the same protocol. The predictions obtained using our procedure outperform those obtained from ARACNE and MRNET, and give a precision-recall curve very similar to CLR.

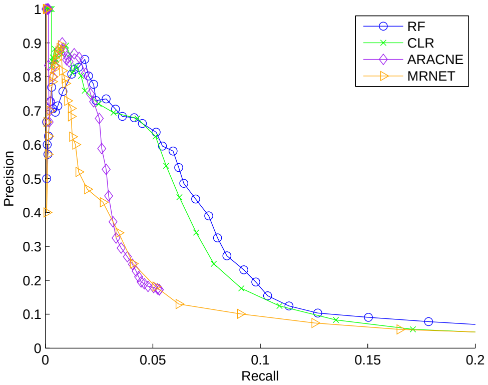


**Fig. 1.** Precision-recall curves for the *E. coli* network.

## 3 Conclusions

We propose a new algorithm for GRN inference based on tree-based ensemble methods that performs well on both synthetic and real gene expression data. The algorithm is simple and generic, making it adaptable to other types of genomic data and interactions.

So far, we focused on providing a ranking of the regulatory interactions. In some practical applications however, one would like to determine a threshold on this ranking to obtain a practical predicted network. As future work, we would like to extend the technique developed in [8] to better assess the significance of the predicted regulatory links and thus help determining a threshold.

**Acknowledgments**

This work was partially funded 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. VAHT is recipient of a F.R.I.A. fellowship and PG is Research Associate of the F.R.S.-FNRS, Belgium. The scientific responsibility rests with the authors.

## References

1. L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.
2. L. Breiman, J. H. Friedman, R. A. Olsen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth International (California), 1984.
3. J. J. Faith, M. E. Driscoll, V. A. Fusaro, E. J. Cosgrove, B. Hayete, F. S. Juhn, Schneider S. J., and Gardner T. S. Many microbe microarrays database: uniformly normalized affymetrix compendia with structured experimental metadata. *Nucleic Acids Research*, 36 (Database issue):D866–70, 2008.
4. J. J. Faith, B. Hayete, J. T. Thaden, I. Mogno, J. Wierzbowski, G. Cottarel, S. Kasif, J. J. Collins, and T. S. Gardner. Large-scale mapping and validation of *Escherichia coli* transcriptional regulation from a compendium of expression profiles. *PLoS Biology*, 5(1):e8, 2007.
5. S. Gama-Castro, V. Jimnez-Jacinto, M. Peralta-Gil, A. Santos-Zavaleta, M. I. Pealoza-Spinola, B. Contreras-Moreira, J. Segura-Salazar, L. Muiz-Rascado, I. Martnez-Flores, H. Salgado, C. Bonavides-Martnez, C. Abreu-Goodger, C. Rodrguez-Penagos, J. Miranda-Ros, E. Morett, E. Merino, A. M. Huerta, L. Trevio-Quintanilla, and J. Collado-Vides. RegulonDB (version 6.0): gene regulation model of Escherichia coli K-12 beyond transcription, active (experimental) annotated promoters and Textpresso navigation. *Nucleic Acids Research*, 36 (Database issue):D120–4, 2008.
6. P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomized trees. *Machine Learning*, 36(1):3–42, 2006.
7. V. A. Huynh-Thu, A. Irrthum, L. Wehenkel, and P. Geurts. Inferring regulatory networks from expression data using tree-based methods. Submitted to a journal.
8. V. A. Huynh-Thu, L. Wehenkel, and P. Geurts. Exploiting tree-based variable importances to selectively identify relevant variables. *JMLR: Workshop and Conference proceedings*, 4:60–73, 2008.
9. A. A. Margolin, K. Wang, W. K Lim, M Kustagi, I Nemenman, and A Califano. Reverse engineering cellular networks. *Nature Protocols*, 1(2):663–672, 2006.
10. P. E. Meyer, K. Kontos, F. Lafitte, and G. Bontempi. Information-theoretic inference of large transcriptional regulatory networks. *EURASIP J Bioinform Syst Biol*, 2007:79879, 2007.
11. R. J. Prill, D. Marbach, J. Saez-Rodriguez, P. K. Sorger, L. G. Alexopoulos, X. Xue, N. D. Clarke, G. Altan-Bonnet, and G. Stolovitzky. Towards a rigorous assessment of systems biology models: The dream3 challenges. *PLoS ONE*, 5(2):e9202, 02 2010.
12. G. Stolovitzky, D. Monroe, and A. Califano. Dialogue on reverse-engineering assessment and methods: The DREAM of high-throughput pathway inference. *Annals of the New York Academy of Sciences*, 1115:11–22, 2007.
13. G. Stolovitzky, R. J. Prill, and A. Califano. Lessons from the dream2 challenges. *Annals of the New York Academy of Sciences*, 1158:159–95, 2009.

# Equation Discovery for Whole-Body Metabolism Modelling

Marvin Meeng[1], Arno Knobbe[1], Arne Koopman[1], Jan Bert van Klinken[2], and Sjoerd A. A. van den Berg[2]

[1] LIACS, Leiden University, the Netherlands, meeng@liacs.nl
[2] LUMC, Leiden University, the Netherlands

This paper is concerned with the modelling of whole-body metabolism. The analysis is based on data collected from a range of experiments involving mice in *metabolic cages*, whose food consumption, activity and respiration has been monitored around the clock. Our aim is to model the dependencies between these different variables by means of (differential) equations, as is customary in systems biology. However, compared to the common setting of modelling on the cellular level, where changes in concentrations are mostly instantaneous, in whole-body metabolism we need to take into account the relatively slow process of food digestion. As a result, the effects of eating will only be visible in the activity and body-heat variables with a certain delay. To further complicate the modelling, the digestive delay depends on the different rates of metabolism of carbohydrates and fat. We accommodate for these (varying) delays in digestion, by adding different time-shifted versions of the primary variables to the data, and applying different levels of smoothing. These newly constructed variables can be interpreted as representations of available blood sugars, with different hypothetic rates of digestions. The Lagramge tool [2] was used to induce ordinary and differential equations that model the enriched data. Lagramge is an equation discovery tool that finds equations of arbitrary (configurable) complexity, and subsequently performs the parameter fitting to the data.

## 1 The metabolic cage

Our data was gathered at the LUMC in a project concerning the Metabolic Syndrome. In that study, 16 genetically identical mice were divided into two equal-sized groups, one was put on a low (LFD), the other on a high fat diet (HFD). During a 3-day period, various variables were recorded every 7.5 minutes while the mice were in a metabolic cage. Such a cage creates a closed environment in which the amount of oxygen and carbon dioxide can be controlled.

For the experiments below, the following variables are used: $VO_2$ (oxygen consumption), $VCO_2$ (carbon dioxide production), RER (respiratory exchange ratio), HEAT (aka. energy expenditure), $F$ (food consumed) and $X$ (total X-activity). The activity was measured using a number of infrared beams in the cage. RER and HEAT are calculated as follows:

$$\begin{aligned} RER &= VCO_2/VO_2 \\ HEAT &= 3.185 \cdot VO_2 + 1.232 \cdot VCO_2 \end{aligned}$$

Though just a simple ratio, RER is very useful, as it gives direct insight into the energy source an organism digested to fuel its activity. Digestion of pure carbohydrates would result in a RER of 1.00, pure fat in 0.707, and a 50/50 diet would result in a RER of 0.85 [1]. This allows to differentiate between the two diet groups.

## 2 Equation Discovery with Lagramge

The equation discovery tool Lagramge was used to generate equations that might capture the essential variables involved during the various stages of metabolism, along with their interplay. Lagramge is capable of discovering both ordinary (OE) and differential (DE) equations. To restrict the search space, the structure of equations can be defined through a context free grammar, which also allows domain specific knowledge to be included, in the form of formulas. Such formulas, then need no longer be discovered, but are available to be included into new candidate equations right from the start. Three different grammars were tested, a *Linear*, *Universal* and *Metabolic Cage* (*MC*) grammar. Because of space limitations, we only present results for the *MC* grammar (shown below), which is somewhat inspired by the *Universal* [2] grammar, but includes the information of the RER and HEAT equations above.

$$\begin{array}{l}
\hline
\mathrm{E} \rightarrow \mathrm{E} + \mathrm{F} \mid \mathrm{E} - \mathrm{F} \mid \mathrm{E} \cdot \mathrm{F} \mid \mathrm{E} \,/\, \mathrm{F} \mid \mathrm{const} \\
\mathrm{F} \rightarrow \mathrm{RER} \mid \mathrm{HEAT} \mid \mathrm{V} \\
\mathrm{RER} \rightarrow \mathrm{V} \,/\, \mathrm{V} \\
\mathrm{HEAT} \rightarrow \mathrm{const} \cdot \mathrm{V} + \mathrm{const} \cdot \mathrm{V} \\
\hline
\end{array}$$

## 3 Experiments

Three variables were chosen as targets for separate experiments: RER, HEAT and X. Both OEs and DEs were sought using an exhaustive search setting of depth 4, as this turned out to be a good trade-off between formula complexity and computation time. Note that depth refers to the number of refinements by means of one of the rules in the grammar.

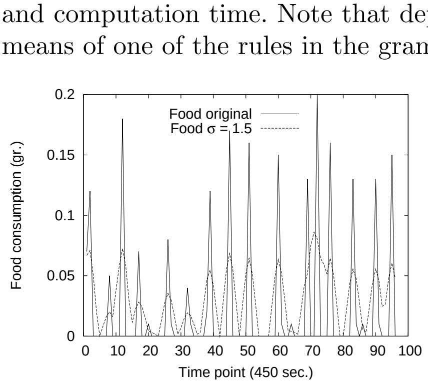


**Fig. 1.** $F$ Smoothed using $\mathcal{G}\,(0, 1.5)$

Data was preprocessed in two ways. First, for all variables but $X$, the data was modestly smoothed using the standard Gaussian kernel, $\mathcal{G}\,(\mu, \sigma)$, with $\mu = 0$, and $\sigma = 1$. As time points are relatively far apart, some smoothing was deemed necessary to compensate for boundary effects. $X$ was left out of this procedure as, compared to eg. food digestion, this is the most abrupt process, and any spread in dependencies between $X$ and other variables is already achieved by their smoothing. Furthermore, for the $F$ variable, data was additionally smoothed using $\mathcal{G}(0, 0.5)$ and $\mathcal{G}(0, 1.5)$ to account for any gradual effects the consumption of food may have.

More than any of the other variables, the energetic effects of food consumption depend on time. As a second step, we therefore added four versions of each $F$ variable, to accommodate potential different rates of metabolism. The time delays were 15, 30, 60 and 90 minutes, which resulted in 15 different $F$ variables in total, five for each kernel version. These were all simultaneously present in the data file, the rationale being that Lagramge might include the important variables from concurrent time scales all in a single equation. Here, the concurrent time scales are related to the various rates and stages of carbohydrate and fat metabolism. Figure 1 shows a smoothing of the $F$ variable for a HFD mouse (first 12 hours are depicted only).

### 3.1 Results

Table 1 shows some of the best equations found using the MC grammar in an exhaustive search of depth 4. For OEs and DEs the target is denoted like $T_d$ and $T'_d$ respectively, where $d$ indicates the diet group. For the food variable ($F$), the $\sigma$ subscript denotes the sigma used for the smoothing kernel and $M$ denotes the shift in minutes. A superficial scan of these results shows that a variety of equation syntaxes are used, with linear equations dominating the $RER$ results. Furthermore, most equations involve at least one delayed $F$ variable, with only a single equation being based on the (slightly smoothed) direct $F$ variable. This clearly illustrates the effect that absorbed nutrients have on the mechanisms by which fuel selection is regulated. Also, Table 1 shows that the activity X is a major determinant of energy expenditure, as would be expected.

**Table 1.** Best equations found for each setting and diet type.

| Target | Equation |
|---|---|
| $RER_{LFD}$ | $0.827 + F_{\sigma 1.5,M30} + 1.645 \cdot F_{\sigma 1.5,M15} - 2.094 \cdot F_{\sigma 1.5,M60}$ |
| $RER_{HFD}$ | $0.790 + F_{\sigma 1.5,M30} + 0.761 \cdot F_{\sigma 1.5,M15} - 0.318 \cdot F_{\sigma 1.5,M60}$ |
| $RER'_{LFD}$ | $-0.004 \cdot F_{\sigma 1.5,M90} + 0.005 \cdot F_{\sigma 0.5,M0} + 0.181 \cdot F_{\sigma 0.5,M90}$ |
| $RER'_{HFD}$ | $-0.008 \cdot F_{\sigma 1.5,M30} + 0.010 \cdot F_{\sigma 0.5,M15} + 0.627 \cdot F_{\sigma 0.5,M30}$ |
| $HEAT_{LFD}$ | $0.441 + 1.218 \cdot 10^{-4} \cdot RER - 0.405 \cdot X$ |
| $HEAT_{HFD}$ | $0.376 + F_{\sigma 1.5,M30} + 2.146 \cdot 10^{-4} \cdot F_{\sigma 1.5,M60} + 0.348 \cdot X$ |
| $HEAT'_{LFD}$ | $(-0.024 + F_{\sigma 0.5,M15}) \cdot (0.015 \cdot F_{\sigma 0.5,M15} - 6.906 \cdot F_{\sigma 0.5,M90})$ |
| $HEAT'_{HFD}$ | $(0.872 - RER) \cdot (-0.007 \cdot F_{\sigma 0.5,M30} + 77.451 \cdot F_{\sigma 1.0,M30})$ |
| $X_{LFD}$ | $(-0.221 + VO_2) \cdot (17217.2 \cdot F_{\sigma 0.5,M0} - 16822.8 \cdot VO_2)$ |
| $X_{HFD}$ | $(-0.206 + VO_2) \cdot (6075.65 \cdot HEAT - 784.577 \cdot VO_2)$ |
| $X'_{LFD}$ | $-0.011 - F_{\sigma 0.5,M15} + F_{\sigma 0.5,M0}/HEAT$ |
| $X'_{HFD}$ | $-0.706 \cdot HEAT + 1.400 \cdot F_{\sigma 0.5,M90} - 9.480 \cdot VCO_2$ |

For all three targets, figure 2 shows the number of occurrences of each version of the $F_{\sigma 0.5}$ variable in the top 1,000 DEs for each diet group. Here we can clearly see a different pattern for the two groups. Compared to the HFL group there are

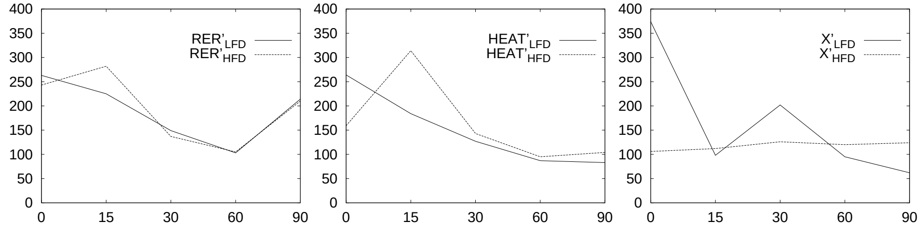


**Fig. 2.** Histograms of three target variables, each for two diets.

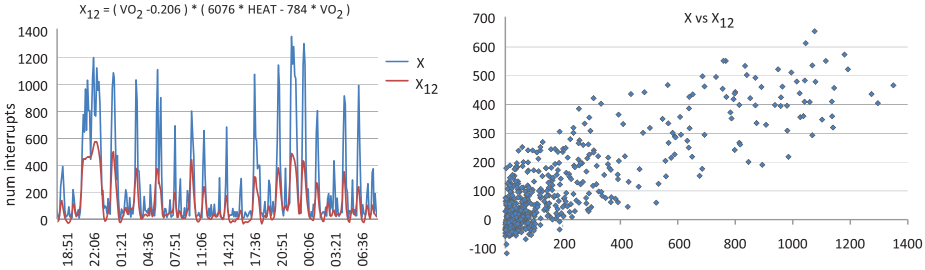


**Fig. 3.** Signal fit and scatter

many more occurrences of the non-shifted variable $F_{\sigma 0.5,M0}$ in the LFD group for $HEAT'$ and $X'$, while $F_{\sigma 0.5,M15}$ is much more frequent for the HFD group for $RER'$ and $HEAT'$. This indicates that the energy from high-carb nutrition is available in the blood stream quicker than for the high-fat diet.

Finally, for target $X$, figure 3 (left) shows an example of one of the found equations (HFD group) compared to the original data, as a function of time. The right figure shows the fit between the actual measurement and its model, for the same equation. The linear correlation between these two functions is $r = 0.84$.

## 4 Conclusion

The experiments reported in this paper demonstrate that Lagramge can be an important tool for modelling in systems biology. It allows the induction of relatively elaborate algebraic and differential equations, including the fitting of parameters, without requiring excessive computation times. Especially where modelling of whole-body metabolism is concerned, the use of various time-shifted variants of the primary data is essential, in order to account for different metabolic processes that have an inherent delay, the details of which may not directly be measurable in the system. The experiments show that the difference in metabolic rates of the two diets considered can be recognized from the difference in time shifts that occur in the respective equations.

## References

1. McLean & Tobin, Animal and Human Calorimetry, Cambridge University Press 1987, ISBNO-521-30905-0.
2. Todorovski & Džeroski, Declarative Bias in Equation Discovery, ICML 1997.

# A fast algorithm for structured gene selection

Sofia Mosci[1], Sivia Villa[1,2], Alessandro Verri[1], and Lorenzo Rosasco[3]

[1] DISI, Università di Genova, Italy
[2] DIMA, Università di Genova, Italy
[3] CBCL, McGovern Institute, Artificial Intelligence Lab, BCS, MIT

**Abstract.** We deal with the problem of gene selection when genes must be selected group-wise, where the groups, defined a priori and representing functional families, may overlap. We propose a new optimization procedure for solving the regularization problem proposed in [4], where the group lasso penalty is generalized to overlapping groups. While in [4] the proposed implementation requires replication of genes belonging to more than one group, our iterative procedure, provides a scalable alternative with no need for data duplication. This scalability property allows avoiding the otherwise necessary pre-processing for dimensionality reduction, which is at risk of discarding relevant biological information, and leads to improved prediction performances and higher selection stability.

## 1 Introduction

The analysis of microarray gene expression data has gained a central role in the process of understanding biological processes. Among the many machine learning algorithms proposed for gene selection, $\ell_1$ regularization proved to be a powerful approach. Nevertheless, when the number of samples is not sufficient to guarantee accurate model estimation, one can exploit the prior knowledge encoded in online repositories. Toward this end *structured sparsity* techniques (see [10,5] and references therein) combine $\ell_1$ regularization with available a priori information, restricting the admissible sparsity patterns of the solution. A promising structured sparsity penalty is proposed in [4], which restricts the support of the solution to be a union of groups defined a priori. A straightforward solution to the minimization problem underlying the method proposed in [4] is to apply state-of-the-art techniques for group lasso in an expanded space, built by duplicating variables that belong to more than one group. Though very simple, such an implementation does not scale to large datasets, with significant group overlap. For this reason we propose an alternative optimization that does not requires gene replication and is thus more appropriate for dealing with high dimensional problems. Our approach is based on a combination of proximal methods (see for example [1]) and constrained Newton method [2], where the latter is used to solve the dual problem associated to the proximity operator of the regularization term. We empirically show that our scheme significantly outperforms state-of-the-art algorithms with data duplication, and has improved prediction and selection performance when applied to microarray data.

## 2 The GLO-pridu Algorithm

Given a response vector $y = (y_1, y_2, ..., y_n)$, a $n \times d$ gene expression matrix $X$, and $B$ subsets of genes $\mathcal{G} = \{G_r\}_{r=1}^{B}$ with $G_r \subset \{1, \dots, d\}$, we consider the minimization of the functional $\mathcal{E}_\tau(\beta) := \frac{1}{n} \left\|X\beta - y\right\|^2 + 2\tau \Omega^{\mathcal{G}}(\beta)$ with

$$\Omega^{\mathcal{G}}(\beta) = \inf\{\sum_{r=1}^{B} \|v_r\| \;:\; v_r \in \mathbb{R}^d, \mathrm{supp}(v_r) \subset G_r, \sum_{r=1}^{B} v_r = \beta\}.$$

The functional $\Omega^{\mathcal{G}}$ was introduced in [4] as a generalization of the group lasso penalty to allow overlapping groups, while maintaining the group lasso property of enforcing sparse solutions which support is a *union of groups*.

If one needs to minimize $\mathcal{E}_\tau$ for high dimensional data, the use of standard second-order methods such as interior-point methods is precluded, since they need to solve large systems of linear equations. On the other hand, accelearated first order methods based on proximal methods [1] are accurate, and already proved to be a computationally efficient alternative in many machine learning applications [3,7]. A proximal algorithm for minimizing the sum $F + \Omega$ of a differentiable functional $F$ and a not differentiable penalty $\Omega$ combines the forward gradient descent step on $F$ with the evaluation of the proximity operator of $\Omega$

$$\beta^p = \mathrm{prox}_{\tau/\sigma\Omega^{\mathcal{G}}} \left(\beta^{p-1} - (\sigma)^{-1}\nabla F(\beta^{p-1})\right) \tag{1}$$

for a suitable choice of $\sigma$. Due to one-homogeneity of $\Omega^{\mathcal{G}}$, its proximity operator reduces to the identity minus the projection onto the convex set $K = \{v \in \mathbb{R}^d, \|v\|_G \leq 1\ \forall G \in \mathcal{G}\}$, with $\|\beta\|_G = (\sum_{j\in G} \beta_j^2)^{1/2}$ . While in group lasso (without overlap, i.e. $G_r \cap G_s = \emptyset, \forall r \neq s$) the projection can be computed group-wise, so that the proximity operator resolves to group-wise soft-thresholding

$$\left(\mathbf{S}_{\tau/\sigma}(\beta)\right)_j = (||\beta||_{G_r} - \tau/\sigma)_+ \beta_j, \qquad \text{for } j \in G_r, \qquad \text{for } r = 1, \dots, B, \tag{2}$$

with general overlap the proximity operator has not a closed a form and must be computed approximatively as in the following theorem.

**Theorem 1.** *Given $\beta \in \mathbb{R}^d$, $\mathcal{G} = \{G_r\}_{r=1}^{B}$ with $G_r \subset \{1, \dots, d\}$, the projection onto $\tau K$ with $K = \{v \in \mathbb{R}^d, \|v\|_{G_r} \leq \tau$ for $r = 1, \dots, B\}$ is given by*

$$[\pi_{\tau K}(\beta)]_j = \beta_j (1+\sum_{r=1}^{\hat{B}} \lambda_r^* \mathbf{1}_{r,j})^{-1} \;\; \textit{with} \;\; \lambda^* = \underset{\lambda\in\mathbb{R}^{\hat{B}}_+}{\mathrm{argmax}} -\sum_{j=1}^{d} \beta_j^2 (1+\sum_{r=1}^{\hat{B}} \mathbf{1}_{r,j}\lambda_r)^{-1} - \sum_{r=1}^{\hat{B}} \lambda_r \tau^2,$$

*$\hat{\mathcal{G}} = \{G \in \mathcal{G}, \|\beta\|_G \geq \tau\} := \{\hat{G}_1, \dots, \hat{G}_{\hat{B}}\}$, and $\mathbf{1}_{r,j}$ is $1$ if $j \in \hat{G}_r$ and $0$ otherwise.*

The above maximization problem is the dual problem associated to the projection onto $\hat{K}(\beta) = \{v \in \mathbb{R}^d, \|v\|_G \leq 1 \forall G \in \hat{\mathcal{G}}\} \supset K$, which involves only the $\hat{B} \leq B$ active constraints. In order to solve it efficiently we employ Bertsekas' constrained Newton method [2]. In Algorithm 1 we report our scheme for computing the regularization path for problem $\beta(\tau) = \mathrm{argmin}\, \mathcal{E}_\tau(\beta)$, i.e. the set of solutions corresponding to different values of the parameter $\tau_1 > \ldots > \tau_T$. The proximal algorithm used in 1 is an acceleration of (1) inspired to [6].

**Algorithm 1** GLO-pridu Algorithm

```
Given: $\tau_1 > \tau_2 > \cdots > \tau_T, \mathcal{G}, \eta \in (0,1), \delta \in (0,1/2), \epsilon_0 > 0, \nu > 0$
Let: $\sigma = ||\Psi^T \Psi||/n$, $\beta(\tau_0) = 0$
for $t = 1, \ldots, T$ do
    Initialize: $\beta^0 = \beta(\tau_{t-1}), \lambda_0^* = 0$
    while $||\beta^p - \beta^{p-1}|| > \nu||\beta^{p-1}||$ do
        • Set $w = h^p - (n\sigma)^{-1}\Psi^T(\Psi h^p - y)$
        • Compute $\beta^p = \pi_{\tau/\sigma K}(w)$ as in Th. (1) via Bertsekas' Algorithm
        • Update $c_p = (1 - t_p)c_{p-1}, \quad t_{p+1} = 1/4(-c_p + \sqrt{c_p^2 + 8c_p})$,
                    $h^{p+1} = \beta^p(1 - t_{p+1} + t_{p+1}/t_p) + \beta_{p-1}(t_p - 1)t_{p+1}/t_p$
    end while
    $\beta(\tau_t) = \beta^p$
end for
return $\beta(\tau_1), \ldots, \beta(\tau_T)$
```

## 3 Numerical Experiments

**Projection vs duplication** In [4] the authors show that minimizing $\mathcal{E}_\tau$ is equivalent to minimizing the standard group lasso functional in an expanded space built by replicating variables belonging to more than one group. Such a formulation allows using any state-of-the-art algorithms for group lasso. In terms of proximal methods, a solution is given by substituting in Algorithm 1 the proximity operator $I - \pi_K$ with the group-wise soft-thresholding of Eq.(2) (we refer to this algorithm as GL-prox). In the following we compare the performances of GLO-pridu and GL-prox in terms of computing time on a set of synthetic data. Note that we consider only the computing performance and not the prediction and selection performance, since the two algorithms lead the same solution. The input variables $x = (x_1, \ldots, x_d)$ are uniformly drawn from $[-1, 1]^d$. The labels $y$ are given by $y = c\beta \cdot x + w$, where $\beta$ is equal to 1 on the first 240 variables and 0 otherwise, $w \sim N(0, 1)$, and $c$ sets the signal to noise ratio to 5:1. We define $G_1 = [1, \ldots, 100]$, $G_2 = [81, \ldots, 180]$, and $G_3 = [1, \ldots, 20, 161, \ldots, 240]$ (20% pairwise overlap). The remaining $B - 3$ groups are built by randomly drawing sets of 100 indexes. We let $n = 2400$, and vary $d$ and $B = \alpha d/100$, where $\alpha$ can be thought of as the average number of groups a single gene belongs to. We then evaluate the running time for computing the entire regularization path for GL-prox and GLO-pridu, repeat 20 times for each pair $(d, \alpha)$, and report the results in Tab.1. When $\alpha$ gets significant there is a clear advantage in using GLO-pridu.

**Microarray data** We consider the microarray experiment presented in [4] where the breast cancer dataset compiled by [9] (8141 genes for 295 tumors)

**Table 1.** Running time (mean ± standard deviation) in seconds. For each $d$ and $\alpha$, the left and right side correspond to GLO-pridu, and GL-prox, respectively.

| | $\alpha = 1.2$ | | $\alpha = 2$ | | $\alpha = 5$ | |
|---|---|---|---|---|---|---|
| $d$=1000 | $11.7 \pm 0.4$ | $24.1 \pm 2.5$ | $11.6 \pm 0.4$ | $42 \pm 4$ | $13.5 \pm 0.7$ | $1467 \pm 13$ |
| $d$=5000 | $31 \pm 13$ | $38 \pm 15$ | $90 \pm 5$ | $335 \pm 21$ | $85 \pm 3$ | $1110 \pm 80$ |
| $d$=10000 | $16.6 \pm 2.1$ | $13 \pm 3$ | $90 \pm 30$ | $270 \pm 120$ | $296 \pm 16$ | $> 12$h |

is analyzed with the group lasso with overlap penalty and the 637 gene groups corresponding to the MSigDB pathways [8]. In [4] the accuracy of a logistic regression is estimated via 3-fold cross validation (CV). On each split the 300 genes most correlated with the output are selected and the optimal $\tau$ is chosen via CV. 6, 5 and 78 pathways are selected with a $0.36 \pm 0.03$ CV error. We applied GLO-pridu to the entire data set with two loops of k-fold CV (k$=3$ for testing). The obtained CV error is $0.33 \pm 0.05$ and $0.30 \pm 0.06$, with k$=3$ and k$=10$ for validation, respectively. In both cases the number of selected groups is $2, 3$, and 4, with 1 group in 3, and 3 pathways selected in 2 out of 3 splits. Not only the CV is lower, but also the number of selected groups is much more stable when avoiding the correlation-based filtering. Note that the improved CV error might be due to the second optimization step (RLS). The computing time for running the entire framework for GLO-pridu (comprising data and pathways loading, recentering, selection via GLO-pridu, regression via RLS on the selected genes, and testing) is 850s (k=3) and 3387s (k=10).

## 4 Discussion

We presented an efficient optimization scheme, whose convergence is theoretically guaranteed, for selecting genes from microarray data according to biological priors. Our procedure allows computing the solution of group lasso with overlap, even in high dimensional problems and large groups overlap, and has a great computational advantage with respect to state-of-the-art algorithms. When applied to microarray data, it has improved prediction and selection performance, since a possibly dimensionality reduction step is no longer needed.

## References

1. A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.*, 2(1):183–202, 2009.
2. D. Bertsekas. Projected newton methods for optimization problems with simple constraints. *SIAM Journal on Control and Optimization*, 20(2), 1982.
3. J. Duchi and Y. Singer. Efficient online and batch learning using forward backward splitting. *Journal of Machine Learning Research*, 10:28992934, December 2009.
4. L. Jacob, G. Obozinski, and J.-P. Vert. Group lasso with overlap and graph lasso. In *Proceedings of ICML 2009*, 2009.
5. R. Jenatton, J. Mairal, G. Obozinski, and F. Bach. Proximal methods for sparse hierarchical dictionary learning. In *Proceeding of ICML 2010*, 2010.
6. Y. Nesterov. A method for unconstrained convex minimization problem with the rate of convergence $o(1/k^2)$. *Doklady AN SSSR*, 269(3):543–547, 1983.
7. L. Rosasco, S. Mosci, M. Santoro, A. Verri, and S. Villa. Proximal methods for structured sparsity regularization. *LNAI, Springer (to appear)*, 2010.
8. A. Subramanian and et al. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *PNAS*, 102(43), 2005.
9. L. van 't Veer et al. Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, 415(6871), 2002.
10. P. Zhao, G. Rocha, and B. Yu. The composite absolute penalties family for grouped and hierarchical variable selection. *Annals of Statistics*, 37(6A):3468–3497, 2009.

# Parameter-Free Clustering of Protein-Protein Interaction Graphs

Axel-Cyrille Ngonga Ngomo

AKSW Group
Department of Computer Science
Johannisgasse 26, Room 5-22
04103 Leipzig, Germany
{ngonga}@informatik.uni-leipzig.de
http://bis.uni-leipzig.de/AxelNgonga

## 1 Introduction

Protein-protein interactions govern most processes at cellular level [4, 6]. Significant efforts are currently being invested in the extraction of molecular complexes from protein-protein interactions (PPI) networks. The correct identification of these complexes (i.e., of groups of proteins which display co-regulation patterns) is one of the main building blocks for understanding cellular processes and biological functions. Graph clustering techniques are commonly used as a device to direct the discovery of these complexes. Yet, finding the optimal parameter configuration for processing a given PPI network is highly time-consuming and dataset-dependent. For example, experiments reported in [1] required the analysis of 2,916 parameter combinations to compute the optimal parameterization of the Restricted Neighborhood Search Clustering algorithm (RNSC) [2].

We present the BorderFlow algorithm, a parameter-free approach to the clustering of weighted directed graphs. We evaluate our approach on the results of six high-throughput experiments against the Markov Clustering algorithm (MCL, [5]), one of the leading clustering algorithm for clustering PPI graphs [1]. We show that our algorithm does not only attain state-of-the-art accuracy but that it outperforms MCL in separation.

## 2 The BorderFlow Algorithm

BorderFlow is a general-purpose local graph-clustering algorithm for directed and undirected weighted graphs. It was designed initially for the computation of semantic classes out of large term-similarity graphs [3]. The algorithm computes a soft and complete clustering of the input graph, i.e., BorderFlow assigns each node to one or more clusters.

BorderFlow implements a seed-based approach. The default setting for the seeds consists of taking all nodes in the input graph as seeds. For each seed $v$, the algorithm begins with an initial cluster $X$ containing only $v$. Then, it expands $X$ iteratively by adding nodes from the direct neighborhood of $X$ to $X$

until $X$ is node-maximal with respect to a function called the border flow ratio. The same procedure is repeated over all seeds. As different seeds can lead to the same cluster, identical clusters (i.e., clusters containing exactly the same nodes) that resulted from different seeds are subsequently collapsed to one cluster. The set of collapsed clusters and the mapping between each cluster and its seeds are returned as result.

### 2.1 Formal Specification

Let $G = (V, E, \omega)$ be a weighted directed graph with a set of vertices $V$, a set of edges $E$ and a weight function $\omega$, which assigns a positive weight $\omega(e) \in \mathbb{R}^+$ to each edge $e \in E$. Non-existing edges $e$ are considered to be edges such that $\omega(e) = 0$. Let $X \subseteq V$ be a set of nodes. We define the set $i(X)$ of inner nodes, $b(X)$ of border nodes and $n(X)$ of direct neighbors of $X$ as follows:

$$\begin{aligned} i(X) &= \{x \in X | \neg(\exists y \in V \backslash X : \omega(xy) > 0)\}, \\ b(X) &= \{x \in X | \exists y \in V \backslash X : \omega(xy) > 0\}, \\ n(X) &= \{y \in V \backslash X \ | \exists x \in X : \omega(xy) > 0\}. \end{aligned} \tag{1}$$

For two subsets $X$ and $Y$ of $V$, we define $\Omega(X, Y)$ as the total weight of the edges from $X$ to $Y$ (i.e., the flow between $X$ and $Y$), i.e. $\Omega(X, Y) = \sum\limits_{x \in X, y \in Y} \omega(xy)$. The border flow ratio $F(X)$ of $X \subseteq V$ is then defined as follows:

$$F(X) = \frac{\Omega\big(b(X), X\big)}{\Omega\big(b(X), V \backslash X\big)} = \frac{\Omega\big(b(X), X\big)}{\Omega\big(b(X), n(X)\big)}. \tag{2}$$

The aim of BorderFlow is to compute *non-trivial local maximums*[1] of $F()$. Accordingly, for each node $v \in V$, BorderFlow computes a set $X$ of nodes that maximize the ratio $F(X)$ with respect to the following maximality criterion:[2]

$$\begin{aligned} \forall X \subseteq V, F(X) \text{ maximal} \Leftrightarrow \forall X' \subseteq V \ \forall v \in n(X), \\ X' = X + v \Rightarrow F(X') < F(X). \end{aligned} \tag{3}$$

The computation of the cluster $X$ for $v \in V$ begins with $X = \{v\}$. Then, $X$ is expanded iteratively. Each of these iterations is carried out in two steps. During step 1, the set $C(X)$ of candidates $u \in n(X)$ which maximize $F(X + u)$ is computed as follows: $C(X) := \underset{u \in n(X)}{\arg\max} \ F(X + u)$.

In step 2, BorderFlow picks the candidates $u \in C(X)$ which maximize the flow $\Omega(u, n(X))$. The final set of candidates $C_f(X)$ is thus

$$C_f(X) := \underset{u \in C(X)}{\arg\max} \ \Omega(u, n(X)). \tag{4}$$

[1] One trivial solution to this equation would be to put all nodes in one cluster, leading to an infinite border flow ratio.

[2] For the sake of brevity, we shall utilize the notation $X + c$ to denote the addition of a single element c to a set $X$. Furthermore, singletons will be denoted by the element they contain, i.e., $\{v\} \equiv v$.

The elements of $C_f(X)$ are added to $X$ if and only if the condition $F(X \cup C_f(X)) \geq F(X)$ is satisfied. The insertion of nodes by the means of steps 1 and 2 is iterated until $C(X) = \emptyset$. $X$ is then returned as the cluster for $v$. The clustering procedure is repeated over all nodes of $V$.

### 2.2 A heuristic for maximizing the border flow ratio

The implementation proposed above demands the simulation of the inclusion of each node in $n(X)$ into the cluster $X$ for computing $F(X)$. Such an implementation can be time-consuming as nodes in PPI graphs can have a high number of neighbors. We can show that for a node $v \in n(X)$, maximizing $\Delta F(X, v) = F(X + v) - F(X)$ can be approximated by maximizing:

$$f(X, v) = \frac{\Omega(b(X), v)}{\Omega(v, V \backslash X)}. \tag{5}$$

By setting $C(X) := \underset{u \in n(X)}{\arg\min}\ 1/f(X, u)$, BorderFlow can be implemented efficiently.

### 2.3 Hardening

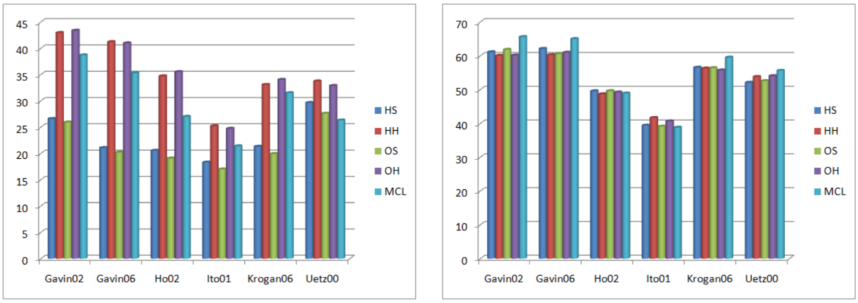
A drawback of BorderFlow is its tendency to generate many overlapping clusters. We can address this drawback by using a simple hardening approach to post-process BorderFlow's results. Let $C_1$ ... $C_\eta$ be the clusters computed by BorderFlow. The hardening of the results of BorderFlow is as follows:

1. Discard all clusters $C_i$ such that $\exists C_j : C_j \subset C_i$.
2. Order all remaining $C_j$ into a list $L = \{\lambda_1, ..., \lambda_m\}$ in descending order with respect to the number of seeds that led to their computation.
3. Discard all $\lambda_z \in L$ with $z > k$, with $k$ being the smallest index such that the union of all $\lambda_i$ with $i \leq k$ equals $V$
4. Re-assign each $v$ to the cluster $C$ such that $\Omega(v, C)$ is maximal.
5. Return the new clusters.

Since our hardening approach can be applied to both the node-optimal and the heuristic version of the algorithm, we will distinguish the following four versions of BorderFlow in the remainder of this paper: *OS* (*O*ptimal, *S*oft), *OH* (*O*ptimal, *H*ard), *HS* (*H*euristic, *S*oft) and *HH* (*H*euristic, *H*ard).

## 3 Results and Discussion

We evaluated our approach using exactly the same data sets and reference data utilized in [1] against MCL, one of the leading clustering algorithm for PPI graphs [6]. As the graphs were undirected and unweighted, we set all edges to be symmetric and set their weight to 1. As evaluation metric we used the separation as defined in [1]. The results of our evaluation as shown in Fig. 1(a) show that the hardened versions of the BorderFlow results outperform the soft versions: $HH$ outperforms $HS$ by 12.24% while $OH$ improves upon $OS$ by 13.63%. Furthermore, $OH$ is 0.1% better than the $HH$ w.r.t. separation. Yet, the runtimes of $OH$ are between 2 and 3 orders of magnitude greater than those of $HH$. Thus, $HH$ should be used for processing large graphs such as PPI graphs. In addition to not necessitating any form of parametrization, the hardened results of our algorithm outperform MCL on all data sets by an average of 5.22% ($OH$). Consequently, $OH$ and $HH$ are superior to the algorithms presented in [1, 6].



(a) Separation

(b) Accuracy

**Fig. 1.** Comparison of the separation and accuracy of BorderFlow and MCL in %.

For the sake of completeness, we also compared BorderFlow and MCL w.r.t. accuracy (see Figure 1(b)). Although there is no statistically significant difference between BorderFlow and MCL (t-test, confidence level = 95%), $HH$ is 2.12% less accurate than MCL in average . Yet, it has been pointed out in previous work [1] that the accuracy metric does not adequately reflect the quality of a clustering, as a clustering containing one large cluster and many small clusters can lead to high accuracy values without reflecting the reference data.

## References

1. S. Brohee and J. van Helden. Evaluation of clustering algorithms for protein-protein interaction networks. *BMC Bioinformatics*, 7:488–506, 2006.
2. A. D. King, N. Przulj, and I. Jurisica. Protein complex prediction via cost-based clustering. *Bioinformatics*, 20(17):3013–3020, November 2004.
3. A.-C. Ngonga Ngomo and F. Schumacher. Borderflow: A local graph clustering algorithm for natural language processing. In *CICLing*, pages 547–558, 2009.
4. Y. Qi, F. Balem, C. Faloutos, J. Klein-Seetharaman, and Z. Bar-Joseph. Protein complex identification by supervised graph local clustering. *Bioinformatics*, 24(13):250–268, 2008.
5. S. van Dongen. *Graph Clustering by Flow Simulation.* PhD thesis, University of Utrecht, May 2000.
6. J. Vlasblom and S. Wodak. Markov clustering versus affinity propagation for the partitioning of protein interaction graphs. *BMC Bioinformatics*, 10, 2009.

# Inference in hierarchical transcriptional network motifs

Andrea Ocone and Guido Sanguinetti

School of Informatics, University of Edinburgh
10 Crichton Street, EH8 9AB Edinburgh, United Kingdom
a.ocone@sms.ed.ac.uk
gsanguin@inf.ed.ac.uk

**Abstract.** We present a novel inference methodology to reverse engineer the dynamics of transcription factors (TFs) in hierarchical network motifs such as feed-forward loops. The approach we present is based on a continuous time representation of the system where the high level master TF is represented as a two state Markov jump process driving a system of differential equations. We present an approximate variational inference algorithm and show promising preliminary results on a realistic simulated data set.

**Keywords:** transcriptional regulation, stochastic process, Bayesian inference

## 1 Introduction

Transcription factor (TF) proteins play a fundamental role in mediating environmental signals in cells. Despite their importance, experimental techniques for measuring their activation state are hampered by several technical problems: TFs are often low expressed, and they are designed to transit fast between active and inactive states through post-translational modifications. For this reasons, an idea that has gained considerable attraction in the machine learning community is to treat TF activity as latent variables in models of gene expression, to be inferred from mRNA levels of target genes. In particular, in recent years there has been considerable interest in using realistic ODE models of transcription, placing a stochastic process prior over TF activities [1–4]. While this approach holds much promise, it has so far been restricted to simple motifs with one (or recently more [5]) TF directly controlling a number of targets.

In this work we extend the work of [3] to hierarchical motifs such as feed-forward loops (FFL). This type of motif is frequently encountered in transcriptional regulatory networks due to its important function in biological signal processing [6]. We use the Bayesian framework and a variational approximation in order to solve the inference problem. Initial results on a simulated data set show the promise of the approach and the identifiability of the model.

## 2 Mathematical methods

### 2.1 Feed-Forward Loop model

We consider an OR gate FFL consisting of a master TF (whose binary activity state is denoted as $\mu$), a slave TF (whose protein expression we denote as $x$) and a target gene whose mRNA expression we denote as $y$. A graphical representation of the network is given in Figure 1. In order to model the FFL, we assume that the activation of the master TF is triggered by a fast post-translational modification (e.g. a phosphorylation). In contrast, we assume that regulation of the target gene by the slave TF is governed by a logical function. Mathematically, the model is described by the following equations:

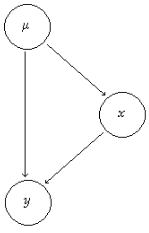


**Fig. 1.** Feed-forward loop (FFL) network motif.

$$\frac{\mathrm{d}x(t)}{\mathrm{d}t} = A_1\mu(t) + b_1 - \lambda_1 x(t) \tag{1}$$

$$\frac{\mathrm{d}y(t)}{\mathrm{d}t} = A\mu(t) + b - \lambda y(t) + A_2\Theta[x(t) - c]. \tag{2}$$

Here $\Theta$ represents the Heaviside step function and $c$ represents a critical threshold for the effect of the slave TF to become felt. Following Sanguinetti et al. [3], the prior distribution for the master TF is modeled by a two-states Markov jump process (the *telegraph process*), which is a continuous time stochastic process that switches with certain transition rates $f_{\pm}(t)$ between an ON state and an OFF state. The first equation refers to the regulation of $x$ by the master TF. As $\mu$ is a binary variable, it represents a logical approximation to a Michaelis-Menten model of transcription [7] where: $A_1$ is the sensitivity of the gene encoding $x$ for the master TF; $b_1$ is the basal transcription rate and $\lambda_1$ is the decay rate of the mRNA. The second equation is similar to the first but it contains an additional term which takes into account the regulation of the target gene by the slave TF.

Assuming that the expression level of the target gene and the slave TF are observed, we want to infer the posterior distribution over the activity of the master TF.

### 2.2 Variational inference

We assume that observations are obtained from the true values by corruption with additive i.i.d. Gaussian noise $p(\hat{y}|y) = \mathcal{N}(\hat{y}|y, \sigma_y)$. Combining the likelihood of the observations with the prior distribution over the process $p(\mu_{0:T})$ enables to find the posterior distribution according to the Bayes' theorem:

$$p_{post}(\mu_{0:T}|\hat{y}) = \frac{1}{Z} p(\hat{y}|\mu_{0:T}) p_{prior}(\mu_{0:T}) \tag{3}$$

where $\mu_{0:T}$ denotes the whole *trajectory* in continuous time of the master TF. The inference problem is solved by approximating the posterior distribution $p_{post}(\mu_{0:T})$ with another distribution $q(\mu_{0:T})$ that is a telegraph process with transition rates $g_{\pm}(t)$. The approximating solution is obtained minimising the *Kullback-Leibler* (KL) *divergence* [8] between the posterior process and the approximating Marvok process:

$$KL\left[q||p_{post}\right] = \int dq \, \log \frac{q}{p_{post}} = KL\left[q||p_{prior}\right] + \ln Z - E_q[\ln \prod_{i=1}^{N} p(\hat{x}_j|x_j)] \tag{4}$$

In this sense, the KL divergence becomes a function of the transition rate for the approximating process $g_{\pm}$, and the problem turns into an optimisation problem. The first term on the right hand side of (4) represents the KL divergence between the prior distribution, which is a telegraph process with prior transition rates $f_{\pm}$, and the approximating posterior distribution, which is another telegraph process with transition rates $g_{\pm}$ [9]. The key technical difficulty is the estimation of the expectation of the Heaviside step under the approximate posterior of the $\mu$ process. To overcome this, we use a Laplace-type approximation:

$$\langle \Theta[x(t) - c] \rangle = P(x(t) > c) \sim \int_{c}^{\infty} \mathcal{N}(x| \langle x(t) \rangle, \langle x(t)^2 \rangle - \langle x(t) \rangle^2) \mathrm{d}x(t) \tag{5}$$

This allows us to compute the functional derivatives of the KL divergence with respect to the rate functions and to solve the optimisation problem by gradient descent. Tha same strategy can be used in order to compute gradients with respect to parameters, but we have not implemented it yet. Details of the algorithm are omitted for space reasons and will be given elsewhere.

## 3 Results

We tested our model on a simulated data set. Observations are given by adding Gaussian noise with SD of 0.03 to 10 discrete time points drawn from the model with a given TF activity (input) and known parameters. The inferred posterior TF activity compared with the true input is showed in Figure 2(A). Figures 2(B-C) show the posterior first moment of $x$ and $y$, with observations.

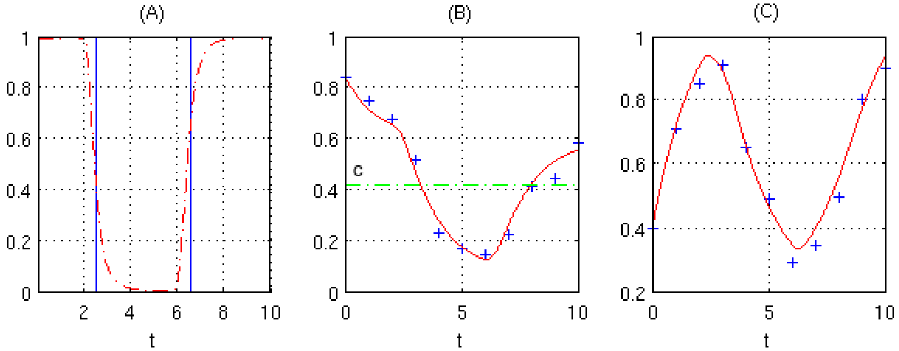


**Fig. 2.** Results on simulated data. (A) Inferred posterior mean activity (*dashed red*) versus true input impulse (*solid blue*). (B) Posterior first moment of $x$ (*solid red*), observations of $x$ (*blue cross*) and critical threshold $c$ (*dashed-dotted green*). (C) Posterior first moment of $y$ (*solid red*) and observations of $y$ (*blue cross*). The parameters of the model were chosen as: $A_1 = 3.7 \cdot 10^{-3}$, $b_1 = 5 \cdot 10^{-4}$, $\lambda_1 = 7 \cdot 10^{-4}$, $A = 2.7 \cdot 10^{-3}$, $b = 8 \cdot 10^{-4}$, $\lambda = 5 \cdot 10^{-4}$, $A_2 = 2.5 \cdot 10^{-3}$, $c = 4.2 \cdot 10^{-1}$.

## 4 Conclusion

The preliminary results shown here indicate that inference in stochastic models of hierarchical network motifs is in principle feasible. Further steps will include optimising model parameters and considering AND gate FFLs, as well as applying to real data from stress response experiments in bacteria.

## References

1. Lawrence, N.D., Sanguinetti, G., Rattray, M.: Modelling transcriptional regulation using gaussian processes. In: Advances in Neural Information Processing Systems 19, pp. 785–792. MIT Press, Cambridge, MA (2007)
2. Gao, P., Honkela, A., Rattray, M., Lawrence, N.D.: Gaussian process modelling of latent chemical species: applications to inferring transcription factor activities. Bioinformatics. 24(16), i70–i75 (2008)
3. Sanguinetti, G., Ruttor, A., Opper, M., Archambeau, C.: Switching regulatory models of cellular stress response. Bioinformatics. 25(10): 1280–1286 (2009)
4. Barenco, M., Tomescu, D., Brewer, D., Callard, R., Stark, J., Hubank, M.: Ranked prediction of p53 targets using hidden variable dynamic modeling. Genome Biology. 7:R25 (2006)
5. Opper, M., Sanguinetti, G.: Learning combinatorial transcriptional dynamics from gene expression data. Bioinformatics. 26(13): 1623–1629 (2010)
6. Mangan, S., Alon, U.: Structure and function of the feed-forward loop network motif. Proc. Natl. Acad. Sci. USA. 100(21): 11980–11985 (2003)
7. Alon, U.: An introduction to systems biology. Chapman and Hall, London (2006)
8. Bishop, C.M.: Pattern recognition and machine learning. Springer, Singapore (2006)
9. Opper, M., Sanguinetti, G.: Variational inference for Markov jump processes. In: Advances in Neural Information Processing Systems 20. MIT Press, Cambridge, MA (2008)

# A note on inference for reaction kinetics with monomolecular reactions

Manfred Opper and Andreas Ruttor[1]

Computer Science, TU Berlin

**Abstract.** We develop a variational lower bound to the free energy for stochastic reaction models with monomolecular reactions which can be used for approximate inference. This bound is based on simpler partition function which can be evaluated efficiently.

## 1 Monomolecular reactions

The problem of probabilistic inference for stochastic reaction models in systems biology has attracted considerable interest, see e.g. [1]. A variety of inference techniques like sampling approaches, variational bounds and weak noise approximations have been considered in order to solve the inference problems efficiently. While these methods are usually general enough to be applicable to arbitrary reaction models, one might ask the question whether one can work out approximations for specific models which allow for an exact treatment of a nontrivial part of the problem. In this submission we will show that specific types of auxiliary likelihoods can be efficiently computed for models with monomolecular reactions. These can be used to compute a variational lower bound to the true free energy.

The state of reaction models is described by a vector $\mathbf{n} = (n_1, \ldots, n_M)$, where $n_i$ is the number of molecules of species $i$. The stochastic dynamics is assumed to be a Markov jump process (MJP) defined by a rate function $f(\mathbf{n}'|\mathbf{n})$ which determines the temporal change of transition probabilities via $P(\mathbf{n}', t + \Delta t|\mathbf{n}, t) \simeq \delta_{\mathbf{n}',\mathbf{n}} + \Delta t\, f(\mathbf{n}'|\mathbf{n})$ for $\Delta t \to 0$. The rate $f(\mathbf{n}'|\mathbf{n})$ is the sum of the rates for all individual processes which lead from $\mathbf{n}$ to $\mathbf{n}'$.

*Monomolecular* reaction systems are defined by three possible types of processes which have rates and corresponding changes of states $\mathbf{n}' \to \mathbf{n}$ given by

$$\begin{aligned} c_{jk} n_j &\text{ with } n'_l = n_l + \delta_{lk} - \delta_{lj} \\ c_{0j} &\text{ with } n'_l = n_l + \delta_{lk} \\ c_{j0} n_j &\text{ with } n'_l = n_l - \delta_{lj}\ . \end{aligned} \tag{1}$$

The rates are at most linear in the number of molecules/species. In the first reaction, molecules are converted from type $j$ into type $k$. The second reaction describes a creation of molecules of type $j$ and the third the corresponding deletion or degradation process. Only a single component or a pair of components change by an amount ±1. In the latter case the changes are of opposite signs.

Although such processes do not contain *chemical* reactions, they have been applied to stochastic *reaction-diffusion* systems e.g. for the *Bicoid* protein evolution in *Drosophila.* Approximate inference for this model has been discussed within a variational mean field approach [3] and a weak noise approximation [4].

If we assume that there are typically about $N$ molecules/species, then the *Master equation*, which governs the temporal evolution of the marginal probability $P_t(\mathbf{n})$ is a system of linear equations of the size $O(N^M)$. Remarkably, at least in principle, an exact solution can be given in terms of convolutions of a multivariate Poisson distribution and $M$ multinomial distributions [2]. If $M$ grows large, e.g. in applications to reaction diffusion models, the practical use of such an exact solution may be questioned.

If we consider the evolution of the first moment $\mathbf{m}(t) = E[\mathbf{n}(t)]$ instead, the situation is much simpler. For monomolecular reactions one simply obtains the "classical" linear (!) rate equations

$$\frac{dm_i}{dt} = c_{0i} + \sum_{j=1}^{M} c_{ji} m_j - \sum_{j=0}^{M} c_{ij} m_i \tag{2}$$

which are of the size $M$ rather than $O(N^M)$.

## 2 Inference

In order to estimate rate constants from a set of noisy observations $\mathbf{y}_k$, $k = 1, \ldots, K$ taken at discrete times $t_k$ one can apply a likelihood based approach, such as a Bayesian one. In this case, the partition function which equals the probability of the observations

$$Z = E\left[\prod_{k=1}^{K} P(\mathbf{y}_k|\mathbf{n}(t_k))|\mathbf{n}(0) = \mathbf{n}\right] \tag{3}$$

given all parameters is required. Here $E[\ldots]$ denotes an expectation over all paths of the process with given initial condition $\mathbf{n}$ at $t = 0$.

A typical form of the conditional likelihood $P(\mathbf{y}_k|\mathbf{n}(t_k))$ could be

$$P(\mathbf{y}_k|\mathbf{n}(t_k)) \propto \exp\left[-\frac{1}{2\sigma^2}\sum_k ||\mathbf{y}_k - \mathbf{L}[\mathbf{n}(t_k)]||^2\right] \tag{4}$$

which describes a noisy measurement of a linearly transformed vector $\mathbf{n}$. The linear transformation might describe a summation of all molecule numbers in a certain window of many "neighbouring" species (cells in a compartment model) by the measuring process. We may then assume that the dimensionality of $\mathbf{y}$ stays finite even when the number of species $M$ grows large.

## 3 Exact solution for an artificial likelihood

In principle, the partition function $Z$ could be calculated recursively by a backward type of algorithm. Again, the size of the corresponding system of equations is expected to be too large to be of practical use. However, for the case of monomolecular reactions exact solutions are possible if we restrict ourselves to a family of artificial "likelihoods" which are simpler compared to (4). The corresponding partition functions are defined by

$$Z_0 \doteq E\left[\exp\left\{\sum_{k=1}^{K} \mathbf{u}^\top(t_k)\mathbf{n}(t_k)\right\} |\mathbf{n}(0) = \mathbf{n}\right] \tag{5}$$

where the "log - likelihood" $\mathbf{u}^\top(t_k)\mathbf{n}(t_k)$ is linear rather than quadratic in $\mathbf{n}$. To compute $Z_0$, we consider the function

$$\psi_t(\mathbf{n}) \doteq E\left[\exp\left\{\int_t^T \mathbf{v}^\top(s)\mathbf{n}(s)\, dt\right\} |\mathbf{n}_t = \mathbf{n}\right] , \tag{6}$$

where $\mathbf{v}(s) = \sum_k \delta(s-t_k)\mathbf{u}(s)$. It is easy to show that $\psi_t(\mathbf{n})$ fulfils the backward type equation

$$\frac{d}{dt}\psi_t(\mathbf{n}) = \sum_{\mathbf{n}'\neq\mathbf{n}} f(\mathbf{n}'|\mathbf{n})\, [\psi_t(\mathbf{n}) - \psi_t(\mathbf{n}')] - \mathbf{v}^\top(t)\mathbf{n}(t)\psi_t(\mathbf{n}) \tag{7}$$

which must solved backwards in time with $\psi_T(\mathbf{n}) \equiv 1$. This is again of the large dimensionality $O(N^M)$. Using the form of the rates (2), we can show that the solution of (7) is of the form $\psi_t(\mathbf{n}) = a(t)e^{\mathbf{b}(t)^\top \mathbf{n}}$ where the functions $a(t)$ and $r_i(t) \doteq \ln b_i(t)$ obey the systems of equations

$$\frac{dr_i}{dt} = -\sum_{k\neq 0} c_{ik}(r_k - 1)\ ; \qquad i = 1, \ldots, M \qquad \frac{da}{dt} = -a\sum_{k\neq 0} c_{0k}(r_k - 1) \tag{8}$$

Here we have included a diagonal element for the matrix $c_{ij}$ which is defined by $c_{ii} \doteq -\sum_{j=0, j\neq i} c_{ij}$. The first equation holds for times $t$ *between two "observations"*. $r_i(t)$ jumps at the times $t_k$ by $r_i(t_k^-) = r_i(t_k^+)e^{u_i(t_k)}$ for $k = 1, \ldots, K$.

Note, that equations (8) are of *small size* $M$ compared to (7). For reaction diffusion problems it is even possible to take the limit $M \to \infty$ which results in a linear partial differential equation. For simple geometries one can solve these easily using e.g. Fourier methods.

## 4 Variational lower bound to the free energy

Unfortunately, $Z_0$ in (5) does not correspond to a real measurement model. It is just a multivariate, multi - time *generating function* for the Markov process. In principle, this could be used in order to (approximately) recover the multivariate distribution of arbitrary functions of the process using e.g. saddle - point methods. In this contribution, we will use a simpler variational approach, which unlike the "standard" variational method (based on the Kullback - Leibler divergence) yields a *lower bound* to the free energy $-\ln Z$. Using the convex duality transformation $\frac{1}{2\sigma^2}||\mathbf{a}||^2 = \max_{\boldsymbol{\phi}} \left\{ -\frac{\sigma^2}{2}||\boldsymbol{\phi}||^2 + \boldsymbol{\phi}^\top \mathbf{a} \right\}$ . to represent (4) and exchanging the max with the expectation in (3) yields the bound $-\ln Z \geq \max_{\{\boldsymbol{\phi}\}_{k=1}^K} F(\boldsymbol{\phi})$ where

$$F(\boldsymbol{\phi}) = -\frac{\sigma^2}{2}\sum_k ||\boldsymbol{\phi}_k||^2 + \sum_k \boldsymbol{\phi}_k^\top \mathbf{y} - \ln E\left[\exp\left(\sum_k \boldsymbol{\phi}_k^\top \mathbf{L}(\mathbf{n}(t_k))\right)|\mathbf{n}(0) = \mathbf{n}\right] \quad (9)$$

For any $\boldsymbol{\phi}$, the expectation in the last term is of the form of (5) and can be computed efficiently. The optimisation w.r.t. to $\boldsymbol{\phi}$ could be based on a gradient ascent method. The gradient of the last term

$$\nabla_{\boldsymbol{\phi}_k} \ln E\left[\exp\left(\sum_l \boldsymbol{\phi}_l L(\mathbf{n}(t_l)\right)\right] = \langle L(\mathbf{n}(t_k))\rangle \quad (10)$$

can be written as an average $\langle \ldots \rangle$ over the "posterior" process obtained from the auxiliary likelihood. For a similar approach to a different model, see [5]. Such a posterior is also Markov but an inhomogeneous one with a time dependent rate function that is given by $g_t(\mathbf{n}'|\mathbf{n}) = f(\mathbf{n}'|\mathbf{n})\frac{\psi_t(\mathbf{n}')}{\psi_t(\mathbf{n})}$ . In the case of monomolecular reactions, the posterior process is also monomolecular and we can show that the prior reaction rates are replaced by

$$c_{jk} \to c_{jk}e^{b_k - b_j} \qquad c_{j0} \to c_{j0}e^{-b_j} \qquad c_{0k} \to c_{0k}e^{b_k} \ .$$

Inserting these rates into (2) and solving this system of linear ODEs would give an efficient approach for computing the gradient (10).

We expect to have results and comparisons with other methods ready by the time of the workshop.

## References

1. Neil D. Lawrence, Mark Girolami, Magnus Rattray and Guido Sanguinetti. Learning and Inference in Computational Systems Biology, *The MIT Press*, 2009.
2. Tobias Jahnke and Wilhelm Huisinga. Solving the chemical master equation for monmolecular reaction systems analytically. *J. Math. Biol.*, 2006.
3. Michael A Dewar, Visakan Kadirkamanathan, Manfred Opper, and Guido Sanguinetti. Parameter estimation and inference for stochastic reaction-diffusion systems: application to morphogenesis in d. melanogaster. *BMC Systems Biology*, 4:21, 2010.
4. Andreas Ruttor and Manfred Opper. Approximate parameter inference in a stochastic reaction-diffusion model. *Proceedings of The Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS) 2010*, pp. 273-280, editors: Y. W. Teh and M. Titterington, JMLR: W&CP 9 (2010).
5. Manfred Opper and Guido Sanguinetti. Learning combinatorial transcriptional dynamics from gene expression data. *Bioinformatics*, 26:1623–1629, 2010.

# Collaboration-based Function Prediction 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

**Abstract.** We consider the problem of predicting the functions of individual proteins in protein-protein interaction (PPI) networks. Existing techniques assume that proteins that are topologically close in the network tend to have similar functions. We hypothesize that better predictive accuracy can be obtained by generalizing this assumption. We call two functions collaborative if proteins with one function often interact with proteins performing the other function. Our hypothesis is that techniques that extract such function collaboration information from networks, and exploit it, can yield better predictions. We propose and evaluate two such techniques. A comparative evaluation on three *S. cerevisiae* interaction networks, at different levels of detail, shows that the new techniques consistently improve over state of the art function prediction techniques, with improvements in F-measure ranging from 3% to 17%.

## 1 Methods

The PPI network is represented by protein set $P$ and interaction set $E$. Each $e_{pq} \in E$ shows an interaction between two proteins $p \in P$ and $q \in P$. Let $F$ be the set of all the functions that occur in the PPI network. Each classified protein $p \in P$ is annotated with an $|F|$-dimensional vector $FS_p$ that indicates the functions of this protein: $FS_p(f_i)$ is 1 if $f_i \in F$ is a function of protein $p$, and 0 otherwise. $FS_p$ can also be seen as the set of all functions $f_i$ for which $FS_p(f_i) = 1$. Similarly, the $|F|$-dimensional vector $NB_p$ describes how often each function occurs in the neighborhood of protein $p$. $NB_p(f_i) = n$ means that among all the proteins that interact with $p$, $n$ have function $f_i$.

In this section we discuss two methods for the task of function prediction in PPI networks. They both predict functions based on function collaboration.

### 1.1 A Reinforcement Based Function Predictor

In this method, we try to quantify how strongly two functions $f_i$ and $f_j$ collaborate, in the following way. Let $FuncColVal(f_i, f_j)$ denote the strength of collaboration between $f_i$ and $f_j$. We consider each classified protein $p \in P$ in turn. If function $f_j$ occurs in the neighborhood of protein $p$ (i.e., $NB_p(f_j) > 0$) then we increase the collaboration value between function $f_j$ and all the functions in $FS_p$:

$$\forall f_i \in FS_p : FuncColVal(f_i, f_j) {+}{=} \frac{NB_p(f_j) * R}{support(f_j)}$$

If function $f_j$ does not occur in the neighborhood of $p$ ($NB_p(f_j) = 0$), we decrease the collaboration value between function $f_j$ and all the functions belonging to $FS_p$:

$$\forall f_i \in FS_p : FuncColVal(f_i, f_j) {-}{=} \frac{P}{support(f_j)}$$

$support(f_j)$ is the total number of times that function $f_j$ appears on the side of an edge $e_{pq}$ in the network. $R$ and $P$ are "Reward" and "Punish" coefficients determined by the user. Next, we determine the candidate functions for an unclassified protein $p$ and rank them based on how well they collaborate with the neighborhood of protein $p$. As an example of a candidate functions strategy, consider Majority Rule: this method nominates all functions that appear in the direct neighborhood of the unclassied protein (and among these, will select the most frequently occurring ones). After selecting candidate functions, we rank them based on how well they collaborate with the neighborhood of unclassified protein $p$. Formula (1) assigns a collaboration score to each candidate function $f_c$:

$$Score(f_c) = \sum_{\forall f_j \in F} NB_p(f_j) * FuncColVal(f_j, f_c) \tag{1}$$

High score candidate function(s) collaborates better with the neighborhood of $p$ and are predicted as its functions. We call the above method the "Reinforcement based function predictor", as it is based on reinforcing collaboration values between functions as they are observed.

### 1.2 SOM Based Function Predictor

The second approach presented in this work employs a Self Organizing Map (SOM) for the task of function prediction in PPI networks. We map the PPI network to a SOM as follows:

- Input Layer: The number of input neurons equals the number of functions in the PPI network. So, if $inputNeurons$ is the set of all neurons in the input layer then $|inputNeurons| = |F|$. The values we put in the input layer are extracted from the neighborhood function vector of the protein: if $inputNeuron(i)$ is the $i$'th neuron in the input layer then $inputNeuron(i) = NB_p(f_i)$.

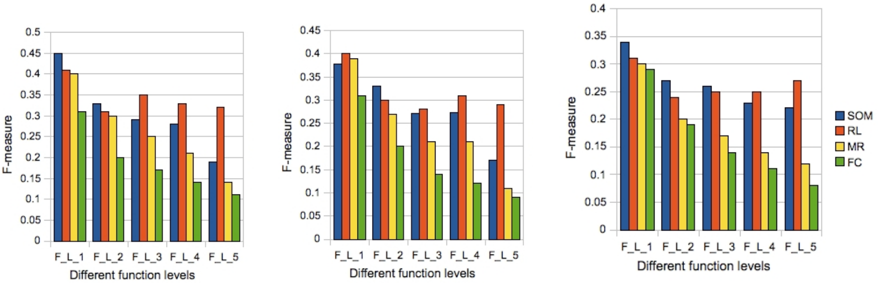
- Output Layer: The number of output neurons equals to number of functions in the PPI network ($|outputNeurons| = |F|$). The values we put in the output layer are extracted from the function vector of the protein: if $outputNeuron(i)$ is the $i$'th neuron in the output layer then $outputNeuron(i) = FS_p(f_i)$.
- Network Initialization: Weights of the neurons can be initialized to small random values; in our implementation we initialized all the weights to zero.
- Adaption: Weights of winner neurons and neurons close to them in the SOM lattice should be adjusted towards the input vector. The magnitude of the change decreases with time and with distance from the winner neuron. Here, we take some new parameters into consideration which are $LearningRate(LR)$, $DecreasingLearningRate(DecLR)$ and $TerminateCriteria(TC)$ parameters. $LR$ is the change rate of the weights toward the input vector and $DecLR$ determines the change rate of $LR$ in different iterations. $TC$ is the criteria in which the learning phase of SOM will terminate. Here, we think of TC as the minimum amount of change required in one iteration: when there is less change, the training procedure stops. We use Formula (2) for updating weights of output neurons.

$$W_{ij,New} = W_{i,j,Current} + LR * (NB_p(j) - W_{i,j,Current}) \tag{2}$$

- Testing: For each protein $p$ in the PPI network that we did not use in the training phase, we find the Euclidean distance between $NB_p$ and the weight vectors. We select the output neurons which have the shortest Euclidean distance to $NB_p$ and predict them as the functions of protein $p$. The number of predicted functions is fixed and determined by the user.

## 2 Evaluation

We compare our collaboration based methods (i.e., collaborative-RL and SOM) with similarity based methods (i.e., Majority Rule [3] and Functional Clustering [2, 1]) on the Krogan, VonMering and DIP-Core datasets, using average F-measure as the evaluation criterion. We predict 3 functions for each unclassified protein in all methods and then we compare the F-measure of different methods. Figure (1), compares SOM and Collaborative-RL (or RL in short) with function similarity based methods on the Krogan, DIP-Core and VonMering datasets respectively. We compare the methods on five different function levels. For example, two functions 11.02.01 (rRNA synthesis) and 11.02.03 (mRNA synthesis) are considered the same up to the second function level (i.e., 11.02 = RNA synthesis), but not on deeper levels. In all three datasets, collaboration based methods predict functions more accurately than similarity based methods. As we consider more detailed function levels, the difference between their performance increases.



(a) DIP Core (b) Von Mering (c) Korgan

**Fig. 1.** Compare Collaborative based methods (SOM and RL) with function similarity based methods (MR and FC) at five different function levels in Krogan, Von Mering and DIP Core dataset. In all function levels, collaboration based methods predict functions more accurately than similarity based methods.

## 3 Conclusion

To our knowledge, this is the first study that considers function collaboration for the task of function prediction in PPI networks. We view biological process as an aggregation of each individual protein functions and our hypothesis is that topologically close proteins have collaborative functions. We proposed two methods based on this assumption. The first method rewards the collaboration value of two functions if they interface with each other in two sides of one interaction and punishes the collaboration value if just one of the functions occurs on either side of an interaction. At prediction time, this method ranks candidate functions base on how well they collaborate with the neighborhood of unclassified protein. The second method uses Self Organizing Map (SOM) for the task of function prediction. We selected two methods, Majority Rule and Functional Clustering, as representatives of the similarity based approaches. We compared our collaboration based methods with these similarity based methods on three interaction datasets: Krogan, DIP-Core and VonMering. We examined up to five different function levels and we found classication performance according to F-measure values indeed improved, sometimes by up to 17 percent, over the benchmark methods employed.

## References

1. Christine Brun, Carl Herrmann, and Alain Guénoche. Clustering proteins from interaction networks for the prediction of cellular functions. *BMC Bioinformatics*, 5:95, 2004.
2. A. D. King, Natasa Przulj, and Igor Jurisica. Protein complex prediction via cost-based clustering. *Bioinformatics*, 20(17):3013–3020, 2004.
3. Benno Schwikowski, Peter Uetz, and Stanley Fields. A network of protein-protein interactions in yeast. *Nat Biotechnol*, 18(12):1257–1261, December 2000.

# Automated Detection of Chaotic and Oscillatory Regimes

Daniel Silk[1,2], Paul D.W. Kirk[1,2], Christopher Barnes[1,2], and Michael P.H. Stumpf[1,2,3]

[1]Centre for Bioinformatics, Imperial College London
[2]Institute of Mathematical Sciences, Imperial College London
[3]Centre for Integrative Systems Biology at Imperial College London

**Abstract.** Qualitative parameter estimation algorithms are a promising but underdeveloped class of inference methods [1]. They offer the ability to infer parameters when only qualitative features of the system are known or need to be specified. Here we adapt the unscented Kalman filter for identification of model parameters that give rise to a desired Lyapunov spectrum. This novel approach extends the reach of these techniques, allowing detection of the most complex and elusive dynamical behaviours. We demonstrate our method on three ODE models, including a simple model of the Hes1 regulatory system.

**Key words:** Lyapunov exponents, unscented Kalman filter, qualitative inference, system design, chaos, oscillations.

## 1 Introduction

Even very simple dynamical systems can exhibit rich and complex spectrums of dynamical behaviour. This is perhaps most prominently exemplified by the logistic map,

$$x_{t+1} = \lambda x_t(1 - x_t)$$

(with $1 < \lambda \leq 4$) for which, in a landmark paper in 1972 [2], the bewildering complexity of the possible dynamics was first discussed. From these early beginnings the notion of chaos in dynamical systems rapidly spread across the physical, biological and social sciences as a *deterministic* description of seemingly random and unpredictable behaviours. An alternative to stochastic interpretations, chaos theory offers the potential for control, if not long-term predictability, of these observed phenomena, and with applications to such popular fields as climate prediction [3], stock market forecasting [4] and medical research [5], the "butterfly effect" remains an idea that captures the general public's imagination.

Characterising the dynamics of simple systems such as the logistic map is a relatively straightforward task. However for more complicated models, finding regimes of complex (or simple) dynamics poses a serious challenge, with analytical solutions rarely available and simulation-based searches prohibitively expensive (in terms of computational time required). Here we shall consider different dynamical systems of the general form

$$\frac{dy(t)}{dt} = f(y(t), y_0; \theta)$$

where $y(t)$ denotes the state of the system, $f$ is the gradient field characterised by the vector valued parameter $\theta$, $t$ is time and $y_0 = y(0)$ are the initial conditions. The aim of our analysis is to provide values of $\theta$ that lead to certain desired (or avoid certain undesired) types of dynamical behaviour. To this end we introduce a powerful and flexible new approach that, given a dynamical system, and a set of initial guesses for the model parameters, converges to a parameter regime that exhibits the target qualitative behaviour (e.g. stationary state, limit cycle or chaos), if it is compatible with the system.

As a tool for fitting models, the approach outlined below has numerous strengths. Firstly, it provides a method for inferring parameters when, as is often the case, only qualitative features (eg. stable attraction to steady state) of the system are known. Secondly, by driving the parameter inference in this way, parameter identifiability, sloppiness and related concepts may be linked directly to features of interest, thus informing our intuition about the system under study. We believe this has huge potential, and for example, could help in providing treatments for epilepsy, where chaotic regimes are thought to be desired and regular oscillatory behaviour avoided, and heart arrhythmia, where the opposite holds [6]. Thirdly, our approach offers an elegant solution to the notoriously difficult problem of inferring parameters for oscillatory behaviour, and further, more complex behaviours such as chaos may be treated in the same way, offering a novel means of chaos control/anti-control for autonomous systems. Finally, moving away from the traditional aims of parameter inference, our approach suggests a natural formulation and solution for the problem of system design, namely (i) we encode the desired dynamical behaviour in a suitable manner, and (ii) we search parameter space for parameters that generate the target dynamics.

It is within this framework that we introduce our method below, and then present example applications to the classic Lorenz oscillator, the simplest biochemical model capable of a Hopf bifurcation [7, 8] and finally to a model of the Hes1 gene-regulatory system.

## 2 Encoding Dynamics through Lyapunov Exponents

A central concept of dynamical systems theory, Lyapunov exponents may be used to discriminate between qualitatively different orbit types. They may be understood as the rate of exponential divergence of trajectories starting off sufficiently close to one another, thus determining the long term evolution of initially small perturbations to the system. Under some widely applicable assumptions [9] , the Lyapunov exponents, $\lambda_i$, may be derived as the logarithms of the eigenvalues of the matrix

$$L(y_0) = \lim_{t \to \infty} \left( J(t) J^T(t) \right)^{1/2t} , \quad (1)$$

where $J(t)$ is the Jacobian matrix of the system evaluated at $y_0$.

The Lyapunov exponents, $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$, encode the qualitative behaviour of a system in the following way: For $\lambda_1 < 0$ the system will attain a stable stationary state; for $\lambda = 0$ the system's attractor is characterised by stable oscillations; finally, for $\lambda_1 > 0$, initially close trajectories will diverge exponentially over time and we refer to this regime as chaotic. So rather than trying to specify the orbits of a system we will characterise the desired attractor via its Lyapunov spectrum.

In general, non-linear system equations and the asymptotic nature of $L$, preclude the analytic evaluation of expression (1). Instead, numerical approximations of the Lyapunov exponents may be calculated. We here employ the approach detailed in [10].

## 3 Filtering as a Design Tool

Unlike in the case for linear systems, where identifying suitable parameters that produce observed or desired dynamics is trivial, inference for highly non-linear systems is far from straightforward. Given that exact inferences are prohibitively expensive for even small to moderate systems a host of different approximation approaches have been proposed [11–13]. Here we take a Bayesian approach, seeking to approximate the posterior distribution over parameters, conditioned on the desired behaviour (as specified through the LE or spectrum). We adopt the perspective that an approximation to the probability distribution is easier to justify or control than an approximation to the non-linear dynamics ever will be [14].

In brief, we begin with an initial set of parameters $\theta_0$, and apply the unscented Kalman filter to the following dynamical state space model:

$$\begin{aligned} \theta_{k+1} &= \theta_k + v_k \\ \lambda_{\text{target}} &= g(\theta_k, y_0; f) + u_k , \end{aligned}$$

where $g$ is a function of the parameters (here a numerical routine to calculate the Lyapunov exponents), $\lambda_{target}$ is a constant target vector of Lyapunov exponents, $v_k \sim \mathcal{N}(0, v^2)$ and $u_k \sim \mathcal{N}(0, u^2)$ are the process and measurement noise, respectively, $y_0$ denotes the initial conditions and $f$ is the dynamical system under investigation (with parameters $\theta$).

The approach allows us to infer a set of parameters, $\theta$, that give rise to the desired behaviour or attractor structure.

Sometimes we may wish to constrain a search to particular regions of parameter space. In the context of modelling a real world process, this may be based upon the physical impossibility of certain parameter combinations (e.g. negative chemical reaction rates). More generally we may wish to avoid "badly behaving" regions of parameter space where the model is, for example, unbounded.

Instead of constraining the filter algorithm, we write a new observation function $g^* = g \circ p$, where $p$ maps the input parameters onto the region of interest. For example, in order to avoid negative chemical reaction rates, $p$ may output the absolute value of the parameters (and the unchanged model).

## 4 Results

In this section we present the results of applying our method to three different ordinary differential equation (ODE) systems. For each system we form the appropriate dynamical state space model, specify a target Lyapunov spectrum, $\Lambda$, and then employ the unscented Kalman filter to home in on suitable parameters. In all equations given below, a dot above a variable indicates the time differential.

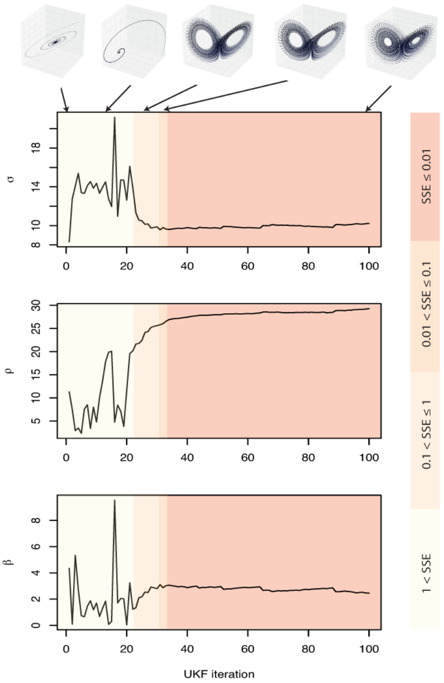


**Fig. 1.** Detecting chaos. Plots showing the estimated parameters for the Lorenz system at successive iterations of a single run of the unscented Kalman filter. Snapshots of the developing attractor are shown above each plot. Colours indicate the sum of squares error between $\Lambda$ and the Lyapunov exponents displayed for each parameter vector. After only 22 iterations, the characteristic "butterfly" attractor emerges. The final parameters and Lyapunov exponents are $\sigma = 10.2$, $\rho = 29.2$, $\beta = 2.45$ and $(0.899, 2.74e - 4, -14.6)$.

### 4.1 The Lorenz Oscillator

Originally used to model weather and climate phenomena, Lorenz's oscillator [15] was an early example of how sensitivity to initial conditions can give rise to unpredictable behaviour.

Defined by the system of ODEs,

$$\begin{aligned}\dot{x} &= \sigma(y - x)\\ \dot{y} &= x(\rho - z) - y\\ \dot{z} &= xy - \beta z\ ,\end{aligned}$$

the model is known to exhibit a chaotic regime with Lyapunov exponents, $\Lambda^* = (0.906, 0, -14.57)$, for parameter vector $(\sigma, \rho, \beta) = (10, 28, 8/3)$. As a proof of concept example, we attempt to infer back these parameters, starting from different positions in parameter space, by setting our target Lyapunov spectrum to $\Lambda = \Lambda^*$. If we restrict the parameter search to the region $[0, 30]^3$, we are able to do this reliably from random starting positions. The parameter trajectories and evolving attractor of one such inference is shown in Fig. 1, where after the 100th iteration, the sum of squares error is less than 8e-5. However, without restrictions, the inference is able to converge to different parameter combinations that display very similar Lyapunov exponents.

### 4.2 Detecting Oscillations in two Biological Systems

We now consider two biological examples, searching each for oscillations – a feature that is ubiquitous in nature, yet elusive to parameter inference techniques. The first example is a mathematical construct representing the simplest biochemical reaction system that permits a Hopf bifurcation [7, 8]. It is known that this system, described by,

$$\begin{aligned}\dot{x} &= (Ak_1 - k_4)x - k_2xy\\ \dot{y} &= -k_3y + k_5z\\ \dot{z} &= k_4x - k_5z\ ,\end{aligned}$$

where, $x$, $y$, $z$, represent the concentrations of three reactants, $k_i$, are the reaction rates, and, $A$, is the fixed concentration of a fourth reactant, displays a limit cycle for $Ak_1 = k_3 + k_4 + k_5$.

Oscillations in expression levels of the transcription factor Hes1 have been observed *in vitro* in mouse cell lines, and reproduced using various modelling approaches including continuous deterministic delay and discrete stochastic delay models. Here we investigate a simple three component ODE model of the regulatory dynamics with mRNA transcription modelled by a Hill function, and given by,

$$\begin{aligned}\dot{M} &= -k_{deg}M + 1/(1 + (P_2/P_0)^h)\\ \dot{P}_1 &= -k_{deg}P_1 + \nu M - k_1 P_1\\ \dot{P}_2 &= -k_{deg}P_2 + k_1 P_1\ ,\end{aligned}$$

where state variables $[M] = M/V$, $[P_1] = P_1/V$, $[P_2] = P_2/V$, are the molecular concentrations of Hes1 mRNA, cytoplasmic and nuclear proteins respectively, and $V$ is the assumed constant cell volume. $k_{deg}$ is the Hes1 protein degradation rate which we assume to be the same for both cytoplasmic and nuclear proteins, $k_1$ is the rate of transport of Hes1 protein to the nucleus, $P_0$ is the amount of Hes1 protein in the nucleus when the rate of transcription of Hes1 mRNA is at half its maximal value, $\nu$ is the rate of translation of Hes1 mRNA, and $h$ is the Hill coefficient. For the inference we take, $k_1$, to be the experimentally determined value of 0.03 $min^{-1}$ [16]. All other parameters are left free for the inference.

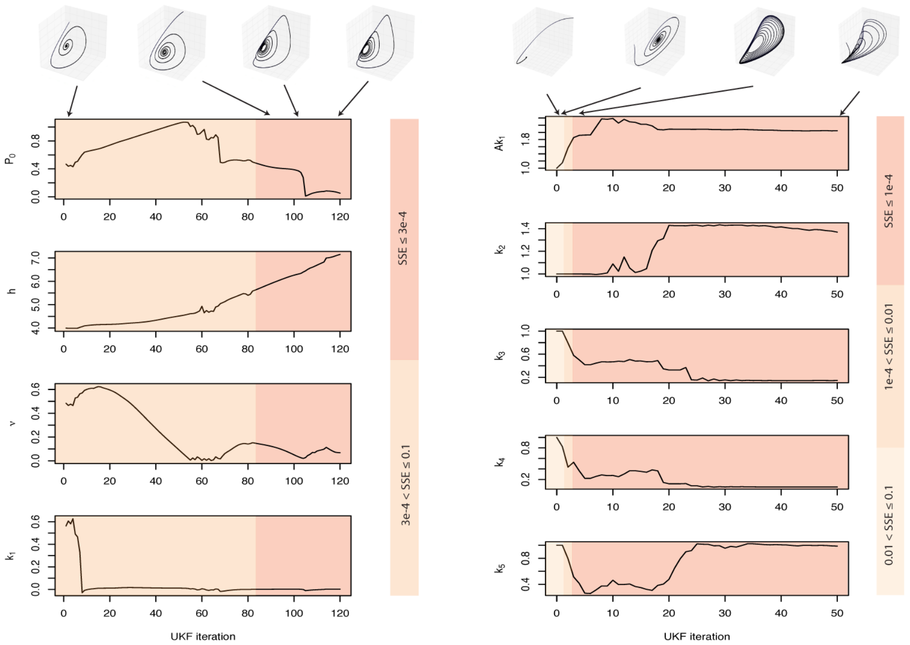


**Fig. 2.** Detecting oscillations. Plots showing parameter trajectories for (*left*) a simple model of the Hes1 regulatory system and (*right*) a Hopf bifurcating system through successive iterations of the unscented Kalman filter. For both systems we are able to meet the stated design objectives.

For both biological systems, we are able to find limit cycles within 30 iterations (see Fig. 2). Observe that the inferred parameters for the simple biochemical system obey the mathematically derived relationship between parameters for oscillations given above. In contrast to the parameters of this system, only parameter , $k_1$, of the Hes1 regulatory model seems strongly constrained by the demand for oscillatory behaviour. We are thus able to use the *qualitative* nature of our inference to hypothesise that oscillations of Hes1 protein and mRNA levels are strongly dependent upon maintaining a low rate of transport of Hes1 protein into the nucleus, and that the dependence on other system parameters is less strong.

## References

1. Endler, L., Rodriguez, N., Juty, N., Chelliah, V., Laibe, C., Li, C., Novère, N.L.: Designing and encoding models for synthetic biology. Journal of The Royal Society Interface 6, S405 (2009)
2. May, R.: Simple mathematical models with very complicated dynamics. Nature 261, 459–467 (1976)
3. Shukla, J.: Predictability in the midst of chaos: A scientific basis for climate forecasting. Science, (1998)
4. Vaga, T.: Profiting from chaos. McGraw-Hill, (1994)
5. Mackey, M., Glass, L.: Oscillation and chaos in physiological control systems. Science, (1977)
6. Ditto, W., Munakata, T.: Principles and applications of chaotic systems. Communications of the ACM, (1995)
7. Kirk, P., Toni, T., Stumpf, M.: Parameter inference for biochemical systems that undergo a hopf bifurcation. Biophysical journal 95, 540–549 (2008)
8. Wilhelm, T., Heinrich, R.: Smallest chemical reaction system with hopf bifurcation. Journal of Mathematical Chemistry, (1995)
9. Oseledec: A multiplicative ergodic theorem. lyapunov characteristic exponents for dynamical systems. Trans. Moscow Math. Soc. 19, 197 (1968)
10. Wolf, A., Swift, J., Swinney, H., Vastano, J.: Determining lyapunov exponents from a time series. Physica D: Nonlinear . . . , (1985)
11. Toni, T., Stumpf, M.: Simulation-based model selection for dynamical systems in systems and population biology. Bioinformatics 26, 104–110 (2010)
12. Moral, P.D., Doucet, A., Jasra, A.: An adaptive sequential monte carlo method for approximate bayesian computation. Annals of Applied Statistics, (2009)
13. Newton, M., Raftery, A.: Approximate bayesian inference with the weighted likelihood bootstrap. Journal of the Royal Statistical Society. Series B ( . . . , (1994)
14. Julier, S., Uhlmann, J.: A general method for approximating nonlinear transformations of probability . . . . Dept. of Engineering Science, (1996)
15. Lorenz, E.: Deterministic nonperiodic flow1. Atmos. Sci, (1963)
16. Hirata, H., Yoshiura, S., Ohtsuka, T., Bessho, Y., Harada, T., Yoshikawa, K., Kageyama, R.: Oscillatory expression of the bhlh factor hes1 regulated by a negative feedback loop. Science 298, 840–3 (2002)

# Multilabel Prediction of Drug Activity

Hongyu Su, Markus Heinonen, and Juho Rousu

Department of Computer Science
PO Box 68, 00014 University of Helsinki, Finland
{hongyu.su,markus.heinonen,juho.rousu}@cs.helsinki.fi
http://www.cs.helsinki.fi/group/sysfys

## 1 Introduction

Machine learning has become increasingly important in drug discovery where viable molecular structures are searched or designed for therapeutic efficacy. In particular, the costly pre-clinical *in vitro* and *in vivo* testing of drug candidates can be focused to the most promising molecules, if accurate *in silico* models are available [7]. During the last decade kernel methods [3, 7, 2, 1, 10] have emerged as an effective way for modelling the activity of candidate drug molecules.

However, classification methods focusing on a single target variable at a time are not optimally suited to drug screening applications where a large number of target cell lines are to be handled. In this paper we propose, to our knowledge, the first multilabel learning approach for molecular classification. Our method belongs to the structured output prediction family [6, 8, 4, 5], where graphical models and kernels have been successfully married in recent years. In our approach, the drug targets (cancer cell lines) are organized in a network, drug molecules are represented by kernels and discriminative max-margin training is used to learn the parameters. We demonstrate the benefits of the multilabel classification approach on a dataset of 60 cancer cell lines and 4554 candidate molecules.

## 2 MMCRF algorithm

The multilabel classification model used here is an instantiation of the structured output prediction framework MMCRF of [5, 4] for associative Markov networks. MMCRF takes as input a kernel matrix $K = (k(x_i, x_j))_{i,j=1}^m$ between the training patterns, which in our case are potential drug molecules, and a label matrix $Y = (\mathbf{y}_i)_{i=1}^m$ containing the multilabels $\mathbf{y}_i = (y_1, \ldots, y_k)$ of the training patterns. The components $y_j \in \{-1, +1\}$ of the multilabel are called microlabels and in our case correspond to different cancer cell lines. In addition, the algorithm assumes an associative network $G = (V, E)$ to be given, where node $j \in V$ corresponds to the $j$'th component of the multilabel and the edges $e = (j, j') \in E$ correspond to a microlabel dependency structure. A joint feature map $\varphi_e(x, \mathbf{y}) = \phi(x) \otimes \psi_e(\mathbf{y}_e)$ for an edge is composed via tensor product of input $\phi(x)$ and output feature maps $\psi_e(u) = ([\![u = (-1, -1)]\!], \ldots, [\![u = (1, 1)]\!])^T$, containing all pairs of an input feature and a labeling for an edge. Corresponding weights are denoted by $\mathbf{w}_e$. The parameters are learned by maximizing the minimum loss-scaled margin between the correct training examples $(x_i, \mathbf{y}_i)$ and incorrect pseudo-examples $(x_i, \mathbf{y}), \mathbf{y} \neq \mathbf{y}_i$ (c.f [5, 4]):

$$\begin{aligned} \underset{\mathbf{w}}{\text{minimize}}\ & \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{m} \xi_i \qquad (1) \\ \text{s.t. } & \mathbf{w}^T\varphi(x_i, \mathbf{y}_i) - \mathbf{w}^T\varphi(x_i, \mathbf{y})) \geq \ell(\mathbf{y}_i, \mathbf{y}) - \xi_i, \\ & \text{for all } i \text{ and } \mathbf{y}, \end{aligned}$$

where $\ell(\mathbf{y}_i, \mathbf{y})$ denotes the Hamming loss between multilabels, and $\xi_i$ denotes slack allotted to example $x_i$. The MMCRF algorithm uses kernels to represent high-dimensional inputs, and optimizes the model in the so called marginal dual form that gives a polynomial-size of the optimization problem. Efficient optimization is achieved via the conditional gradient algorithm with feasible ascent directions found by loopy belief propagation over the Markov network $G$ [5].

## 3 Experiments

We present comparison between the MMCRF multilabel classification model and the support vector machine (SVM) predicting each cell line in isolation, which is considered state of the art.

*Data and preprocessing.* We use the NCI-Cancer dataset obtained through PubChem Bioassay[1] [9] data repository. The dataset contains bioactivity information of large number of molecules against 60 human cancer cell lines in 9 different tissue types. For each molecule tested against a certain cell line, the dataset provides a bioactivity outcome that we use as the classes (active, inactive). The dataset used here contains 4554 molecules.

*Kernel of drug molecules.* Based on preliminary studies, we decided to use the Tanimoto kernel [3]

$$k(fp_1, fp_2) = \frac{N_{fp_1,fp_2}}{N_{fp_1} + N_{fp_2} - N_{fp_1,fp_2}},$$

that is computed from two molecule fingerprints (pre-defined molecule substructure features) by checking the fraction of features that occur in both fingerprints of all features. Above, $N_{fp_1}$ is the number of 1-bits in fingerprint $fp_1$, $N_{fp_2}$ is the number of 1-bits in fingerprint $fp_2$, and $N_{fp_1,fp_2}$ is the number of 1-bits in both of the fingerprints.

[1] http://pubchem.ncbi.nlm.nih.gov

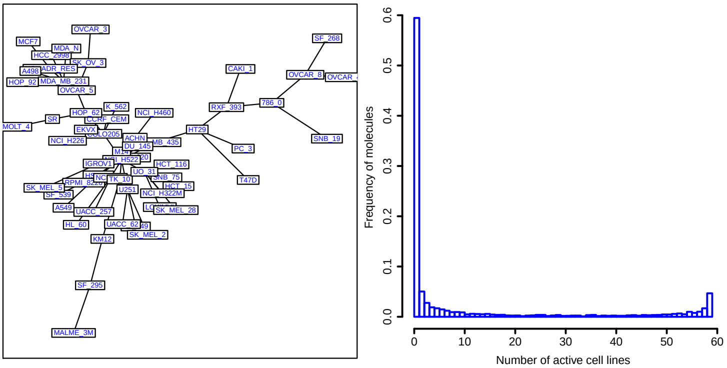


**Fig. 1.** Network constructed for cell lines (left) and the multilabel distribution of molecules (right).

*Network for cell lines* required by MMCRF is constructed as follows. Each node corresponds to a cell line and edges denote potential statistical dependencies. To extract the edges, we used auxiliary data (RNA radiation microarray data) available on the cancer cell lines from NCI database[2]. We built a correlation matrix betwen the pairs of cell lines, and extracted the edges by finding the minimum spanning tree of maximum weight from the correlation matrix of cell lines. The resulting network is shown in Figure 1, left.

*Results.* The label distribution of the dataset (Figure 1, right) turned out to be very skewed: over half of the molecules are inactive against all cell lines and the average number of active cell lines per molecule is small. Because of this skewness, we use F1 score (harmonic mean of precision and recall) instead of accuracy to compare the methods. From a single matrix of predictions $\hat{Y}$, F1 computed from one column gives a score for each cell line, F1 computed from a single row gives a score for each molecule. We used 5-fold cross-validation to evaluate the models' performance.

Figure 2 on the left shows the F1 score of MMCRF versus SVM for each cell line. A sign test shows a statistically significant difference (p=0.009). On the right, the F1 scores of different molecules are grouped based on the number of cell lines they are active against. MMCRF is consistently more accurate for molecules that are active against many cell lines.

## 4 Conclusions

We presented a multilabel classification approach to drug activity classification using the Max-Margin Conditional Random Field (MMCRF) algorithm. By utilizing the statistical dependencies between the cell lines, the MMCRF approach

[2] http://discover.nci.nih.gov/cellminer/home.do

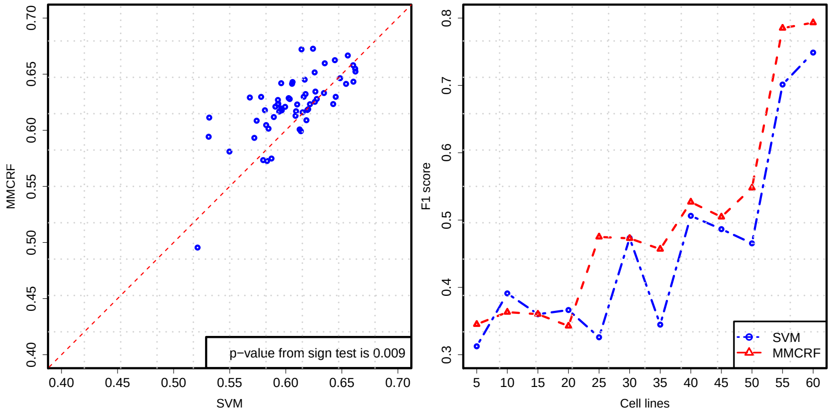


**Fig. 2.** MMCRF against SVM F1 score for each cell line (left) and average F1 score of molecules grouped by the number of active cell lines (right).

is able to significantly outperform SVM on a dataset comprising of a large set of cancer cell lines.

## References

1. Byvatov, E., Fechner, U., Sadowski, J., Schneider, G.: Comparison of support vector machine and artificial neural network systems for drug/nondrug classification. J. Chem. Inf. Comput. Sci. 43, 1882–1889 (2003)
2. Ceroni, A., Costa, F., Frasconi, P.: Classification of small molecules by two- and three-dimensional decomposition kernels. Bioinformatics 23, 2038–2045 (2007)
3. Ralaivola, L., Swamidass, S., Saigo, H., Baldi, P.: Graph kernels for chemical informatics. Neural Networks 18, 1093–1110 (2005)
4. Rousu, J., Saunders, C., Szedmak, S., Shawe-Taylor, J.: Kernel-Based Learning of Hierarchical Multilabel Classification Models. JMLR 7, 1601–1626 (2006)
5. Rousu, J., Saunders, C., Szedmak, S., Shawe-Taylor, J.: Efficient algorithms for max-margin structured classification. Predicting Structured Data pp. 105–129 (2007)
6. Taskar, B., Guestrin, C., Koller, D.: Max-margin markov networks. In: Neural Information Processing Systems 2003 (2003)
7. Trotter, M., Buxton, M., Holden, S.: Drug design by machine learning: support vector machines for pharmaceutical data analysis. Comp. and Chem. 26, 1–20 (2001)
8. Tsochantaridis, I., Hofmann, T., Joachims, T., Altun, Y.: Support vector machine learning for interdependent and structured output spaces. ICML'04 pp. 823–830
9. Wang, Y., Bolton, E., Dracheva, S., et al.: An overview of the pubchem bioassay resource. Nucleic Acids Research 38, D255–D266 (2009)
10. Zernov, V., Balakin, K., Ivaschenko, A., Savchuk, N., Pletnev, I.: Drug discovery using support vector machines. The case studies of drug-likeness, agrochemical-likeness, and enzyme inhibition predictions. J. Chem. Inf. Comput. Sci. 43, 2048–2056 (2003)

# Prediction of DNA-Binding Proteins from Structural Features

Andrea Szabóová[1], Ondřej Kuželka[1], Filip Železný[1], and Jakub Tolar[2]

[1] Czech Technical University, Prague, Czech Republic
[2] University of Minnesota, Minneapolis, USA

**Abstract.** We use logic-based machine learning to distinguish DNA-binding proteins from non-binding proteins. We combine previously suggested coarse-grained features (such as the dipole moment) with automatically constructed structural (spatial) features. Prediction based only on structural features already improves on the state-of-the-art predictive accuracies achieved in previous work with coarse-grained features. Accuracies are further improved when the combination of both feature categories is used. An important factor contributing to accurate prediction is that structural features are not Boolean but rather interpreted by counting the number of their occurences in a learning example.

## 1 Introduction

The process of protein-DNA interaction has been an important subject of recent bioinformatics research, however, it has not been completely understood yet. DNA-binding proteins have a vital role in the biological processing of genetic information like DNA transcription, replication, maintenance and the regulation of gene expression. Several computational approaches have recently been proposed for the prediction of DNA-binding function from protein structure.

Stawiski et al. investigated positively charged patches on the surface of DNA-binding proteins. They used a neural network with 12 features like patch size, hydrogen-bonding potential, the fraction of evolutionarily conserved positively charged residues and other properties of the protein [1]. Ahmad and Sarai trained a neural network based on the net charge and the electric dipole and quadrupole moments of the protein [2]. Bhardwaj et al. examined the sizes of positively charged patches on the surface of DNA-binding proteins. They trained a support vector machine classifier using the protein's overall charge and its overall and surface amino acid composition [3]. Szilágyi and Skolnick created a logistic regression classifier based on the amino acid composition, the asymmetry of the spatial distribution of specific residues and the dipole moment of the protein [4].

In the present work, we combine two categories of features to predict the DNA-binding function of proteins. The first category contains the above mentioned *coarse-grained features* which enabled [4] to achieve state-of-the-art predictive accuracies. The second category contains *structural* features representing characteristic spatial patterns in the unbound conformations of the protein residues. These features are formally described in first-order logic [5] and automatically discovered by our algorithm [7].

Nassif et al. [6] have previously used a first-order logic based approach in a similar context, in particular to classify hexose-binding proteins. The main differences of our approach from [6] are as follows. First, our fast feature-construction algorithm [7] enables us to produce features by inspecting much larger structures (up to tens of thousands of entries in a learning example) than those considered in [6] using the standard learning system Aleph. Second, our structural features acquire values equal to the number of occurrences of the corresponding spatial patterns, whereas [6] only distinguished the presence of a pattern in a learning example from its absence. Our results indicate that occurrence-counting indeed substantially lifts predictive accuracy. Third, rather than proposing an alternative classification method to state-of-the-art approaches, we elaborate its *augmentation* by the use of the structural features. Lastly, the approach of [6] resulted in classifiers that are more easily interpretable than state-of-the-art classifiers and comparable in predictive accuracy. Here we maintain the interpretability advantage but actually improve on the state-of-the-art predictive accuracies both by a purely structural approach (without the coarse-grained features) and even more so through the combination of structural and coarse-grained features.

## 2 Materials and Methods

*Data.* Both the protein and the DNA can alter their conformation during the process of binding. This conformational change can involve small changes in side-chain location, and also local refolding, in case of the proteins. Predicting DNA-binding propensity from a structural model of a protein makes sense if the available structure is not a protein-DNA complex, i.e. it does not contain a bound nucleic acid molecule. We decided to work with a positive data set (UD54) of 54 protein sequences in unbound conformation obtained from [4]. As a negative data set (NB110) we used a set of 110 non-DNA-binding proteins created by [2]. From the structural description of each protein we extracted the list of all contained residues with information on their type and the list of pairwise spatial distances among all residues. As for the coarse-grained features, we followed [4] and extracted features indicating the respective proportions of the Arg, Lys, Asp, Ala and Gly residues, the spatial asymmetry of Arg, Gly, Asn and Ser, and the dipole moment of the protein.

*Method.* We experimented with 7 state-of-the-art attribute-value classifier types listed in Table 1. The attributes correspond to the coarse-grained features as listed above and to the structural features constructed as follows. The feature construction method assumes that proteins are described by means of formal-logic assertions. For example, the assertion res('1AJY', r1, 'CYS') denotes that the protein 1AJY contains a residue r1, which is a cysteine. Similarly, the assertion dist(r1,r2,10) denotes that the distance between residues r1 and r2 is (*approximately*) 10 angstroms. A complete description of a protein is a logical conjunction of such statements, pertaining to all involved residues, and their all pairwise spatial distances that do not exceed 40 Angstroms (computed from coordinates of *alpha carbons*). The full description of a real protein corresponds to a conjunction containing up to tens of thousands of literals.

A *feature* $F$ is a conjunction of first order literals. For a protein $p$ and a feature $F$ we define the *value* of feature $F$ to be the number of groundings $\theta$ such that $p \models F\theta$. In other words, the *value* of a feature is the number of possible ways to match the feature against a given protein. For example, a feature $F =$ res(P, R, 'CYS') counts the number of cysteines in a protein P. An example of a more complicated feature is the following feature

$$F = \mathsf{res(P,R1,'CYS'),\ res(P,R2,'HIS'),\ dist(R1,R2,8)}$$

which counts the number of pairs cystein-histidine, which are 8 angstroms apart from each other. Once we have a sufficiently rich set of features, we may feed the features into any attribute-value learning algorithm.A detailed description of the computational procedures used to accomplish the feature construction task is beyond the scope of this paper. In brief, we rely on the framework of inductive logic programming [5]. In particular, we employ our recently published algorithm [7] since it can scale to rather large structures corresponding to proteins, which would be prohibitively large for mainstream inductive logic programming algorithms. This feature construction algorithm exhaustively constructs a set of features which are not *redundant*, comply with a user-defined language bias and have frequency higher than a given threshold.

## 3 Results

As a result of structural pattern searching we obtained about 1500 patterns present in 54 unbounded DNA-binding proteins. We made two sets of trainings (accuracies are shown in Tab. 1): i) considering just the occurrence of the structural patterns - columns marked with (NC), ii) considering also the number of the occurrence of each pattern - columns marked with (C). We compare classifiers based on our structural patterns (F2) with classifiers based on 10 features (F1) from Szilágyi et al. [4]. We also trained classifiers based on both our features and features from Szilágyi et al. (F1+2). As we can see, we get better results for classifiers considering the number of the occurrence of each pattern. For the most classifiers the accuracy is higher when they are based on our features than on features of Szilágyi et al. However, we get the best results with combination of the two feature-sets. We show here three examples of unbounded DNA-binding proteins with the residues of the pattern which is the most informative according to the $\chi^2$ criterion (Fig. 1).

## 4 Conclusion and Future Work

We have improved on the state-of-the-art accuracies in predicting DNA-binding proteins by combining previously used coarse-grained features with logic-based

| Classifier | F1 | F2(NC) | F1+2(NC) | F2(C) | F1+2(C) |
|---|---|---|---|---|---|
| Linear SVM | 84.0 (2) | 77.5 (5) | 78.1 (4) | 83.0 (3) | **84.2 (1)** |
| SVM with RBK | 81.6 (3) | 67.1 (4-5) | 67.1 (4-5) | 83.0 (2) | **85.4 (1)** |
| Simple log. regr. | 81.6 (3) | 73.9 (5) | 78.8 (4) | **87.6 (1)** | 82.3 (2) |
| $L_2$-regularized log. regr. | 84.0 (2) | 78.7 (5) | 80.5 (4) | 82.4 (3) | **84.2 (1)** |
| Ada-boost | 77.4 (4) | 73.2 (5) | 83.0 (2) | 79.3 (3) | **84.7 (1)** |
| Random forest | 78.6 (4) | 76.8 (5) | **83.6 (1)** | 80.5 (2) | 79.9 (3) |
| J48 decision tree | 75.0 (3) | 70.7 (4) | 75.6 (2) | 68.1 (5) | **76.2 (1)** |
| **Average ranking:** | 3 | 4.79 | 3.07 | 2.71 | **1.43** |

**Table 1.** Accuracies obtained by stratified 10-fold crossvalidation using features of Szilágyi et al. (F1), our structural pattern features (F2) and combination of both of them (F1+2). The numbers in parentheses correspond to ranking w.r.t. the obtained accuracies.

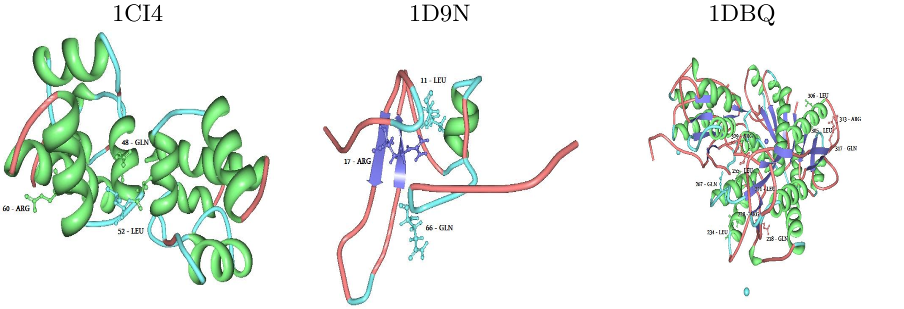


**Fig. 1.** Example proteins containing one discovered pattern shown using the protein viewer software [8]. Residues assumed by the pattern are indicated.

spatial protein features. It turns out that an important factor contributing to the high predictive accuracies is that the latter features are not Boolean but rather are assigned values counting the occurrences of the corresponding spatial pattern in the example protein. We are currently trying to further improve the predictions by incorporating further background knowledge.

Acknowledgement: This work was supported by project ME10047 granted by the Czech Ministry of Education and the Czech Technical University internal grant #10-801940.

## References

1. Stawiski, Gregoret, and Mandel-Gutfreund. Annotating nucleic acid-binding function based on protein structure. *J Mol Biol. 2003*
2. Ahmad, Shandar, and Akinori Sarai. Moment-based prediction of DNA-binding proteins. *Journal of Molecular Biology* 341, no. 1 (July 30, 2004): 65-71.
3. Bhardwaj et al. Kernel-based machine learning protocol for predicting DNA-binding proteins. *Nuc. Acids Res. 2005*
4. Szilágyi A., Skolnick J.. Efficient Prediction of Nucleic Acid Binding Function from Low-resolution Protein Structures *Journal of Molecular Biology.* 2006; 358:922–933.
5. De Raedt L.. *Logical and Relational Learning.* Springer 2008.
6. Houssam N., Hassan A.-A., Sawsan K., Walid K., and Page D. An Inductive Logic Programming Approach to Validate Hexose Binding Biochemical Knowledge. *ILP 2009: The 19th Int. Conf. on Inductive Logic Programming*
7. Kuželka O., Železný F.. Block-wise construction of acyclic relational features with monotone irreducibility and relevancy properties in *ICML '09: 26th International Conference on Machine Learning* 2009.
8. J.L. Moreland and A.Gramada and O.V. Buzko and Qing Zhang and P.E. Bourne. The Molecular Biology Toolkit (MBT): A Modular Platform for Developing Molecular Visualization Applications. *BMC Bioinformatics.* 2005.

# Parameter Estimation in an Endocytosis Model

Katerina Taškova[1], Peter Korošec[1], Jurij Šilc[1]
, Ljupčo Todorovski[2], and Sašo Džeroski[1]

[1] Jožef Stefan Institute, Ljubljana, Slovenia
[2] Faculty of Administration, University of Ljubljana, Slovenia

## 1 Introduction

The task of parameter estimation is regularly encountered in the context of constructing systems biology models. These models often focus on the dynamics of biological systems and have the form of ordinary differential equations (ODEs). The dynamics modeled is typically highly nonlinear and constrained and thus the corresponding parameter optimization problems are hard for traditional local search optimization methods The problem becomes even worse when we attempt to model the dynamics in an automated fashion, using approaches like the machine learning approach of equation discovery [3], which consider both different equation structures and different parameter values. The effectiveness and efficiency of the parameter optimization method used then becomes paramount.

In this context, we investigate the effectiveness and efficiency of different optimization methods for the task of estimating the parameters of ODE models in systems biology. We consider three optimization methods: the local search method Algorithm 717(ALG717) [1] and two meta-heuristic approaches: Differential Evolution (DE) [5] and Differential Ant-Stygmergy Algorithm (DASA) [4]. We compare them on the parameter estimation task in a nonlinear dynamic model of an important endocytotic regulatory system that switches between cargo transport and maturation in early, respectively late endosomes [2]. Both artificial and real data are used in the comparison, which shows that the recent DASA approach is the method of choice: It is both effective (in terms of the quality of the solutions) and efficient (in terms of the speed of convergence).

## 2 Materials

**The Endocytosis Model.** The model we study captures the cellular mechanisms of endosome maturation and cargo transport. It is based on the interactions of proteins from the Rab5 and Rab7 domains. The theoretical and experimental approach undertaken to model the endocytosis rely on the mutually exclusiveness of the Rab5 and Rab7 domains. It has been shown that a cut-out switch model best fits the biological observations [2].

The model is defined by four ODEs and 18 kinetic parameters, describing the behavior of four variables (species), that is the active (GTP-bound) and inactive (GDP-bound) forms of the Rab5 and Rab7 proteins. The corresponding variables are $r_5$ (Rab5-GDP), $R_5$ (Rab5-GTP), $r_7$ (Rab7-GDP), and $R_7$ (Rab7-GTP). The

ODEs are given below, where $v_1, \ldots, v_{10}$ denote different biochemical reactions in which the observed Rab5 and Rab7 proteins take part, while $c_1, \ldots, c_{18}$ are the kinetic rates that are to be estimated.

$$\begin{array}{lll}
v_1 = c_1 & v_2 = \frac{c_2\ r_5\ t}{(100+t)(1+e^{(c_3-R_5)\ c_4})} & \frac{d}{dt}r_5 = v_1 + v_7 + v_9 - v_2 - v_3 \\
v_3 = c_5\ r_5 & v_4 = c_6 & \frac{d}{dt}R_5 = v_2 - v_7 - v_3 \\
v_5 = \frac{c_7\ r_7\ R_7^{c_8}}{c_9+R_7^{c_8}} & v_6 = \frac{c_{10}\ r_7}{1+e^{(c_{11}-R_5)\ c_{12}}} & \frac{d}{dt}r_7 = v_4 + v_{10} - v_5 - v_6 - v_7 \\
v_7 = \frac{c_{13}\ R_5}{1+e^{(c_{14}-R_7)\ c_{15}}} & v_8 = c_{16}\ r_7 & \frac{d}{dt}R_7 = v_5 + v_6 - v_{10} \\
v_9 = c_{17}\ R_5 & v_{10} = c_{18}\ R_7 &
\end{array}$$

The above system is not completely observed in the available measurements: These represent the overall concentration of the two Rab proteins, i.e., the sums of the active and passive form of the Rab5 and Rab7 proteins: $\hat{Y}_1(t) = r_5(t) + R_5(t)$ and $\hat{Y}_2(t) = r_7(t) + R_7(t)$.

**The Data**. Artificial (pseudo-experimental) data were generated by simulating the model for 2 782 time points inside the interval $[0, 1\ 600]sec$ with the parameters values and initial conditions suggested by Del Conte-Zerial et al. [2].

Unlike real-experimental data, the simulated data are exact, i.e., contain no noise. To be more realistic, we also added normal Gaussian noise $(N(0,1))$ to the data: The noise was added relatively to the exact data in a quantity defined by the percentage factor $s$ ($s = 20\%$): $Y_{\text{noisy}} = Y\ (1 + s\ N(0,1))$. We used a resampling procedure for handling the noise, where the "true" model measurements were taken as the mean value calculated from 10 different values of the observed output $Y_{\text{noisy}}$.

We also used real time-course data [2], measured at 10 571 time points in the interval $[-5, 330]sec$ from several independent experiments, three for Rab5 (23 endosomes) and one for Rab7 (15 endosomes). As explained in the original study [2], the data were shifted in time so that (Rab5-to-Rab7) conversion events were synchronized around the time point 0, which explains the negative time points in the observed interval. The data were averaged across all tracked time courses and normalized over the range of measured values.

## 3 Methods

This section describes the optimization problem addressed in parameter estimation of the described model, the optimization methods used and the setup of the experiments investigating the relative performance of the optimization methods.

**Problem Statement.** The main focus of our work is parameter estimation within the Rab5-to-Rab7 conversion model. Given the model structure $m(c)$ (in this case, ODE model) described with a set of adjustable parameters $c = \{c_1, \ldots, c_D\}$, and a set of observation data $d$, the task of parameter estimation is to find values for the model parameters so that the model reproduces the observed data in the best possible way. This is performed by minimizing a cost function that measures the goodness of fit.

The parameter estimation problem is a 22-dimensional optimization problem with $c_i \in (0,4], 0 \leq i \leq 18$, and initial conditions of the species taken as additional parameters to be estimated, $c_i \in (0,2], 19 \leq i \leq 22$. Under the assumption of normally distributed and independent observations with constant variance, this is formulated as a non-linear least-squares problem, where we minimize the sum of squared errors between the observed and predicted (simulated) values of the system outputs.

**Parameter Estimation Methods.** We use three parameter estimation methods: one gradient-descent based (ALG717) and two meta-heuristic approaches (Differential Evolution, DE and Differential Ant-Stigmergy Algorithm, DASA). The first two are well-known methods, DASA is a recent and promising one.

ALG717 [1] is a set of modules for solving the parameter estimation problem in non-linear regression models. The algorithm is a variation of the Newton's method, which uses a model/trust-region technique for computing trial steps along with adaptive choice of the Hessian model. Since ALG717 is a local method, we wrapped it in a loop of restarts with randomly chosen initial points, providing in some way a simple global search. In the experiments, the number of restarts was set to 20.000 (25 evaluations/restart) and additionally, the DGLGB module with user-supplied derivatives of the cost function was used.

DE [5] is a simple and efficient evolutionary population-based heuristic for numerical optimization, which combines a differential mutation strategy and a uniform crossover operation over candidate solutions. The parameters of DE were set as follows [6]: the strategy "DE/rand-to-best/1/exp" was used, population size was $NP = 200$, weight factor $F = 0.85$, and crossover factor $CR = 1.0$.

DASA [4] is a recent ant-colony optimization method for numerical optimization. Based on a fine-grained discretization of the continuous domain, the problem is transformed into a graph-search problem. Parameters' deviations assigned to the graph vertices are used to navigate through the search space. Like DE, DASA setup was also adopted from the study [6], where the number of ants was set to $m = 8$, the pheromone evaporation factor to $\rho = 0.2$, the maximum parameter precision to $\epsilon = 10^{-15}$, the discrete base to $b = 10$, the global scale increase factor to $s_+ = 0.07$, and the global scale decrease factor to $s_- = 0.02$.

**Comparison Methodology.** We apply the above three methods to parameter estimation in the Rab5-to-Rab7 conversion model described above. As input, we use both noise-free and noisy artificial data, as well as measurement data. All parameter estimation experiments were repeated 25 times and limited to 500 000 function evaluations per run.

We compare the methods according to the *sum of squared errors* (SSE). We also consider the *convergence curves* based on the average results over the 25 runs. In addition, we use the *root mean squared error* (RMSE) of the models and the *Correlation coefficient* (R), standardly used to determine how well the predictive model fits the given data in terms of linear dependence.

## 4 Results

Experimental results from parameter estimation of the Rab5-to-Rab7 conversion model with ALG717, DE and DASA using pseudo-experimental and real-

**Table 1.** Experimental results from parameter estimation of the Rab5-to-Rab7 model

| | | SSE | | | RMSE | | | $R^2$ | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $s$ [%] | | alg717 | DE | DASA | alg717 | DE | DASA | alg717 | DE | DASA |
| | | pseudo-experimental data | | | | | | | | |
| 0[1] | Best | 573.60 | 122.63 | 11.99 | 0.45 | 0.21 | 0.07 | 0.499 | 0.942 | 0.993 |
| | Mean | 1105.35 | 245.12 | 51.24 | 0.63 | 0.29 | 0.13 | 0.305 | 0.867 | 0.974 |
| | Std | 243.55 | 70.43 | 36.92 | 0.08 | 0.04 | 0.05 | 0.119 | 0.048 | 0.022 |
| 20[2] | Best | 984.75 | 128.68 | 53.55 | 0.60 | 0.22 | 0.14 | 0.431 | 0.923 | 0.966 |
| | Mean | 1239.02 | 256.83 | 81.31 | 0.67 | 0.30 | 0.17 | 0.296 | 0.859 | 0.952 |
| | Std | 107.44 | 69.04 | 25.81 | 0.03 | 0.04 | 0.03 | 0.113 | 0.048 | 0.015 |
| | | real-experimental data | | | | | | | | |
| | Best | 651.24 | 59.14 | 45.72 | 0.25 | 0.07 | 0.07 | 0.366 | 0.938 | 0.952 |
| | Mean | 800.40 | 67.68 | 54.18 | 0.27 | 0.08 | 0.07 | 0.358 | 0.930 | 0.943 |
| | Std | 71.28 | 3.88 | 4.97 | 0.01 | 0.00 | 0.00 | 0.160 | 0.004 | 0.005 |

[1] optimal case SSE = 0, $R^2 = 1$ [2] optimal SSE = 44.93, $R^2 = 0.944$

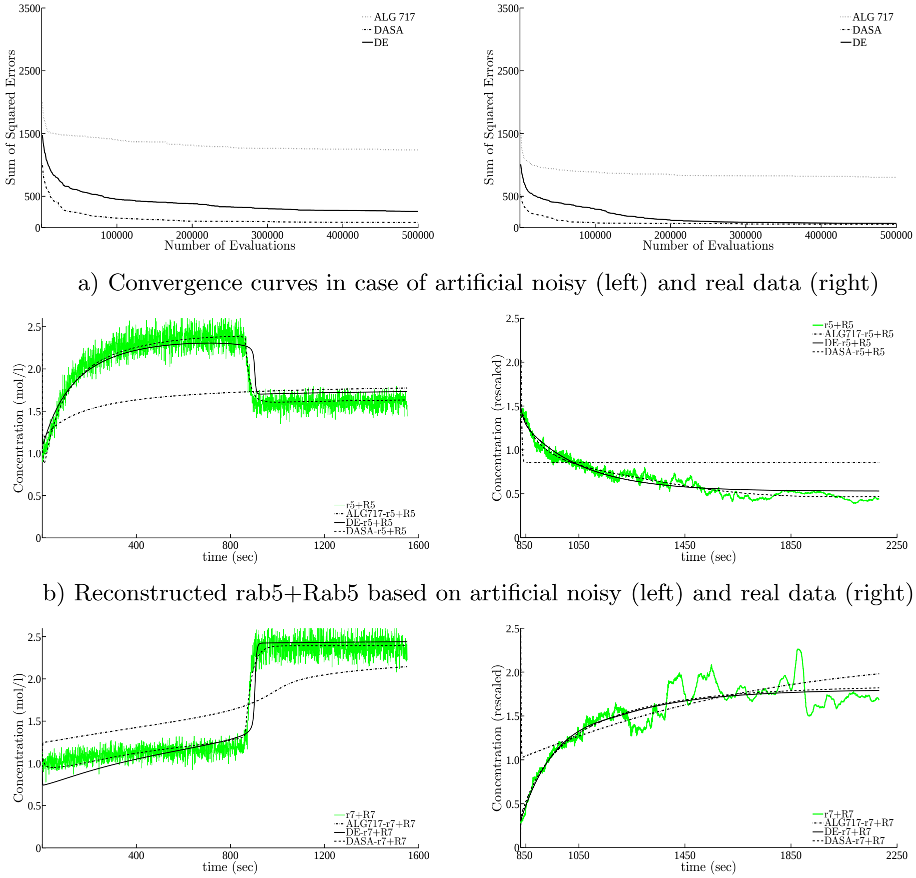
experimental data are presented in Table 1. The row Best represents the values of the measures (SSE, RMSE and $R^2$) for the best solution found with respect to SSE, while the Mean and Std rows outline the average value and standard deviation of the corresponding measures over all 25 runs.

Note that in the case of noise the optimum is not zero anymore, as given in Table 1. As the simulated data are artificially generated, including the noise added in the data, the optimum can be calculated as the sum of squared errors of the noisy observations with respect to the non-noisy observations.

According to Table 1, DASA is closest to the optimum, as confirmed by the high correlation in the case of simulated data (non-noisy and noisy). The simulated reconstruction of the output shows that the best DASA solutions are better than the one obtained by DE and ALG717. Both DE and DASA outperform ALG717 based on the statistics in Table 1: the presence of the measurement noise does not influence the ALG717 performance in a visible way (ALG717 is so far from the optimum that noise does not influence the SSE noticeably). A similar outcome is evident in the case of real data, where DASA and DE are far better than ALG717. Overall, DASA obtains smaller errors than DE.

The graphs in Figure 1a represent the convergence curves of the algorithms for the specific dataset, based on the mean of the best value of the cost function from 25 runs over the number of evaluations. Based on the convergence performance, DE and DASA outperform ALG717 in all cases. When compared to each other in the case of artificial data (Fig. 1a left) and in the case of real data (Fig. 1b right), DASA has visibly faster convergence than DE. The convergence curves on non-noisy artificial data are omitted, as they are quite similar to the noisy data case.

As our main goal was to reconstruct the dynamic of the Rab5-to-Rab7 conversion model, we visualized the predicted model output to validate qualitatively the results from Table 1. Figure 1b gives a comparison of the algorithms on



a) Convergence curves in case of artificial noisy (left) and real data (right)

b) Reconstructed rab5+Rab5 based on artificial noisy (left) and real data (right)

c) Reconstructed rab7+Rab7 based on artificial noisy (left) and real data (right)

**Fig. 1.** Results from parameter estimation in the Rab5-to-Rab7 conversion model on a) convergence performance; b) & c) reconstructed model outputs using the best estimates; based on pseudo-experimental data with 20% noise (left) and real data (right).

predicting the behavior of the model output $Y_1 = r_5 + R_5$ with the best estimated parameters using data perturbed with 20% noise, and real data respectively. Likewise, Figure 1c visualizes the reconstructed output $Y_2 = r_7 + R_7$ from the best run. Since the time scale and concentration scales of both datasets are different, we scaled the time $t$ in the real case with respect to the artificial case by $t \leftarrow (t + 850) \times 4$ and the outputs by $Y_i \leftarrow (Y_i - 10\ 000)/20\ 000 + 0.6$. It is evident that there is a very good correlation between the pseudo-experimental data and the predicted data for DASA and DE. While DE slightly overfits the simulated noise, DASA is more noise resistant. Moreover, DE and DASA capture the trend and shape of the real measurements very well, successfully dealing with the (visible) noise in the real data (Fig. 1b left and Fig. 1c left). The predicted dynamics in the noise-free data case is quite similar and is not included here.

Based on the mean (and median) values of SSE, we performed multiple comparisons using the Holm test, to prove the statistical significance of the results. For a 5% significance level, both DE and DASA are significantly better than ALG717 and there is no significant difference between DE and DASA. However, DASA converges much faster.

## 5 Conclusions

We have considered the task of parameter estimation in a practically relevant model of endocytosis. We have used and compared three different optimization methods, one gradient-descent based (ALG717) and two meta-heuristic approaches, differential evolution (DE) and the differential ant-stigmergy algorithm (DASA). We were especially interested in the performance of the recent and promising DASA approach.

We evaluated the three approaches on both simulated and measured time-course data and found that DASA outperforms ALG717, yielding much better solutions (parameter values). DASA produces slightly better solutions than DE, but not significantly better: However, it does so with a much faster convergence rate and is thus overall better than DE. This makes it our best candidate so far to include as a parameter optimization method within our equation discovery approaches to constructing systems biology models [3], which consider both different equation structures and different parameter values.

**Acknowledgements.** The real experimental data (total protein concentrations of Rab5 and Rab7) come from the group of Marino Zerial at the Max-Planck Institute for Cell Biology and Genetics in Dresden, Germany. Thanks to Perla Del Conte-Zerial and her coworkers for providing these data and the cut-out switch model used in this study. We would especially like to to thank Yannis Kalaidzidis for the invaluable discussions and feedback on the preliminary experimental results.

## References

1. D. S. Bunch, D. M. Gay and R. E. Welsch: Algorithm 717: Subroutines for maximum likelihood and quasi-likelihood estimation of parameters in nonlinear regression models. *ACM T. Math. Software*, 19(1):109–130 (1993)
2. P. Del Conte-Zerial, L. Brusch, J. C. Rink, C. Collinet, Y. Kalaidzidis, M. Zerial and A. Deutsch: Membrane identity and GTPase cascades regulated by toggle and cut-out switches. *Mol. Syst. Biol.*, 4–206 (2008)
3. S. Džeroski and L. Todorovski: Equation discovery for systems biology: finding the structure and dynamics of biological networks from time course data. *Curr Opin Biotechnol* 19(4):360-368, 2008.
4. P. Korošec and J. Šilc: High-dimensional real-parameter optimization using the differential ant-stigmergy algorithm. *Int. J. Intell. Comput. Cybernetics*, 2(1):34–51 (2009)
5. R. Storn and K. Price: Differential Evolution – A simple and efficient heuristic for global optimization over continuous spaces.*J. Global Optim.*, 11:341–359 (1997)
6. K. Tashkova, P. Korošec, J. Šilc, L. Todorovski and S. Džeroski: Parameter estimation in an endocytosis model using bio-inspired optimization algorithms. In: 4th Intl. Conf. Bioinspired Optimization Methods and Application, pp.67–82. (2010)

# Identifying Proteins Involved in Parasitism by Discovering Degenerated Motifs

Celine Vens[1,2], Etienne Danchin[2], and Marie-Noëlle Rosso[2]

[1]Department of Computer Science, Katholieke Universiteit Leuven
Celestijnenlaan 200A, 3001 Leuven, Belgium
celine.vens@cs.kuleuven.be
[2]Institut National de la Recherche Agronomique
400 route des Chappes, BP 167, 06903 Sophia-Antipolis Cedex, France
{etienne.danchin,rosso}@sophia.inra.fr

## 1 Introduction

Identifying motifs in biological sequences is an important challenge in biology. Proteins involved in the same biological system or physiological function (e.g., immune response, chemo-sensation, secretion, signal transduction,...) are subject to similar evolutionary and functional pressures that have an outcome at the protein sequence level. Finding motifs specific to proteins involved in the same process can help deciphering the determinants of their fate and thus be used in identifying new candidate proteins involved in important biological systems.

To our knowledge all currently available methods search motifs in protein sequences at the amino acid level, sometimes allowing degenerate motifs to comply with point variations [1, 2]. However, it is known that conservation of the three-dimensional structure is more important than conservation of the actual sequence for the biological function and proteins that have no detectable sequence similarity can fold in similar structures. At a given position in the sequence, the nature and physico-chemical properties of amino acids in protein families is more conserved than the amino acid itself.

We propose a method that allows to identify emerging motifs based both on conservation of amino acids and on the physico-chemical properties of these residues. Given a set of protein sequences known to be involved in a common biological system (positive set) and a set of protein sequences known not to be involved in that system (negative set) our method is able to identifiy motifs that are frequent in positive sequences while infrequent or absent in negative sequences. The identified motifs can then be used to mine the wealth of protein data now available, in order to identify new previously uncharacterized proteins involved in biological processes of importance.

In this work, the biological system of interest is the protein secretion of a plant parasitic nematode (roundworm). The nematode in question, *Meloidogyne incognita* [3], is a major crop devastator, and controlling it has become an important issue. In this context, it is important to identify the proteins secreted by the nematode into the plant (e.g. cell-wall degrading enzymes that allow the parasite to enter the plant).

## 2 Identifying Degenerated Amino Acid Patterns

### 2.1 Formal Task Description

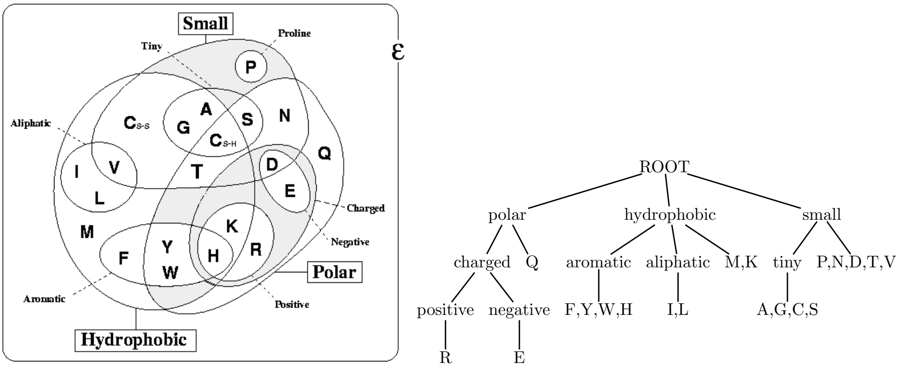
We define the task of identifying degenerated emerging protein motifs as follows:

**Given:** (1) a set of positive proteins $P$, and a set of negative proteins $N$, (2) two frequency thresholds $F_P$ and $F_N$, (3) a set of physico-chemical amino acid properties $C$ and a partial order $\preceq$ defined on the union of $C$ and the amino acid alphabet $A$. For all $ca_1, ca_2 \in C \cup A$: $ca_1 \preceq ca_2$ if and only if $ca_1$ is more general than $ca_2$.

**Find:** the set of all patterns $M$, using symbols in $C \cup A$, that have $freq(M, P) \geq F_P$ and $freq(M, N) \leq F_N$. The function $freq(X, Y)$ returns the number of proteins in set $Y$ that contain the pattern $X$.

### 2.2 Classification scheme

Several amino acid classifications exist in the literature. In this work, we use the classification by Russell et al [4]. It describes amino acids according to their hydrophobicity, size, and polarity, see Fig. 1(a).



(a) (b)

**Fig. 1.** (a) Venn diagram of amino acid properties [4]. (b) Spanning tree.

### 2.3 Algorithm

The algorithm we propose is based on the well-known generate-and-test principle, introduced in the Apriori algorithm [5]. At each iteration, a set of candidates is generated, whose frequency is tested. Given the partial order mentioned above, the search space of all possible patterns is structured as a lattice, with an artificial root element that denotes the empty pattern. The lattice represents an ordering relation: a pattern $(p_1, p_2, p_3, ..., p_n)$ is more general than another pattern $(q_1, q_2, q_3, ..., q_m)$, if and only if $n \leq m$ and for each pair $(p_i, q_i)$ it holds that $p_i \preceq q_i$.

In order to conduct the search efficiently, the candidate generation exploits the antimonotonicity properties of the frequency constraints. This results in the following rules:

- If for a pattern $M$ it holds that $freq(M, P) \leq F_P$, then the pattern does not need to be specialized, since for all its children $C$, it will hold that $freq(C, P) \leq F_P$.
- If for a pattern $M$ it holds that $freq(M, N) \leq F_N$, then for all its children $C$, it will hold that $freq(C, N) \leq F_N$, we do not need to test them.

We have implemented the algorithm using a depth-first search strategy. Essentially, the algorithm looks for those patterns that are frequent in the positive sequences, and meanwhile checks if they are infrequent in the negative sequences. We discuss its most important parts.

**Candidate generation.** In order to perform a complete search, it is important that each relevant pattern in the lattice is considered, and that no pattern is considered more than once. To achieve this, our candidate generation method traverses the lattice from general to specific, and at each step performs two basic operations to generate new candidates given a pattern:

- add a top-level element of the partial order
- minimally specialize the last element of the pattern

In order to ensure that no pattern is considered more than once, we first construct a spanning tree out of the partial order DAG (see Fig 1(b)), and specialize the pattern using this tree.

**Candidate pruning and testing.** When testing a candidate, it is not necessary to check the complete set of positive sequences, it suffices to check the sequences containing the parent candidate, and only in the case all parents have passed the minimal frequency threshold $F_P$. In order to exploit this property, we have to make sure that all parents have been tested before a pattern is considered, i.e. the spanning tree of the amino acids and their properties has to be constructed in a way that, in depth-first traversal, all parents of a node are visited before the node itself is visited. The tree shown in Fig. 1(b) fulfils this constraint.

## 3 Results

We have generated a set of 100 *M. incognita* proteins, that were experimentally proven to be secreted into plants. As negative set, we took 130 proteins that are conserved in non-parasitic nematodes, i.e. that are unlikely to be involved in parasitism.

As we are interested in identifying motifs that are specific to secreted proteins, the maximal frequency threshold for the negative set, $F_N$, was set to 0, i.e., we look for so-called jumping patterns. The minimal frequency threshold for the positives, $F_P$, was set to 20.

The algorithm has identified 3 motifs:

- (hydrophobic charged polar small hydrophobic small hydrophobic tiny small small hydrophobic)
- (hydrophobic hydrophobic small polar polar hydrophobic T hydrophobic polar small hydrophobic hydrophobic)
- (small small small small polar small tiny hydrophobic polar polar hydrophobic small)

Together, these motifs cover 40 of the secreted proteins. Six proteins, including 2 plant cell-wall degrading (PCWD) enzymes, contain all 3 motifs. If we search the complete proteome of *M. incognita*, consisting of 19212 proteins [3], for proteins that contain the 3 motifs (assuming that these would be the most probable of being putative secreted proteins), we obtain a set of 43 proteins that were not included in the training set. Among these, we observe 4 extra PCWD enzymes, which have not yet been experimentally shown to be secreted. We are currently looking into the rest of the set.

## 4 Conclusions

We have proposed an algorithm for the identification of protein motifs that are not restricted to a sequence of amino acids, but can involve physico-chemical amino acid properties. The algorithm uses a traditional generate-and-test approach, with a specific candidate generation operator and pruning step. The algorithm was applied to the task of identifying motifs specific to the secreted proteins of a plant-parasitic nematode, resulting in three degenerate motifs, and a list of 43 candidate proteins to be tested for their involvedness in parasitism.

## References

1. Bailey, T., Elkan, C.: Fitting a mixture model by expectation maximization to discover motifs in biopolymers. In: Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology, AAAI Press (1994) 28–36
2. Ji, X., Bailey, J.: An efficient technique for mining approximately frequent substring patterns. In: Proceedings of the Seventh IEEE International Conference on Data Mining Workshops, IEEE Computer Society (2007) 325–330
3. Abad, P.e.a.: Genome sequence of the metazoan plant-parasitic nematode meloidogyne incognita. Nat Biotechnol. **26**(8) (2008) 909–915
4. Betts, M., Russell, R.: Amino acid properties and consequences of subsitutions. In: Bioinformatics for Geneticists. Wiley (2003)
5. Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., Verkamo, A.I.: Fast discovery of association rules. In: Advances in Knowledge Discovery and Data Mining. AAAI/MIT Press (1996)

# Part III

# Poster Presentations: Abstracts

# Applications of nested sampling optimisation in systems biology

Stuart Aitken

Centre for Systems Biology, University of Edinburgh, Edinburgh

## Abstract

*Background:* Stochastic models are commonly used in systems biology to represent the interaction of small numbers of molecules, and the discrete states that a molecule might adopt. The optimisation of complex stochastic models is challenging as, typically, they cannot be solved analytically.
*Results:* Nested sampling is an effective method for sampling the posterior distributions of model parameters. The samples are obtained as a by-product of calculating the Bayesian evidence. Nested sampling requires a likelihood function, and, in the context of systems biology, the extent to which the data is explained by a given set of model parameters can be computing by an approximate log likelihood function derived from a number of Gillespie simulations. This optimisation strategy is therefore generic, and applicable to kinetic data and steady-state distributions. We have demonstrated that this approach performs well as an optimiser for a number of systems biology models, including models of circadian rhythms.
*Conclusions:* We show that for a range of models, parameters can be optimised, and their standard deviation estimated, by computing a small number of posterior samples by nested sampling.

## Introduction

A number of recent reviews have highlighted the importance of model optimisation to systems biology [1, 5], and the insights that can be gained by a Bayesian approach that considers the posterior distributions of parameters. MCMC is a standard strategy for optimisation and posterior analysis, but suffers from a number of practical problems. Adaptations of MCMC have been developed in order to analyse systems biology models [2], and alternatives to MCMC are examined in [3]. This paper explores the use of nested sampling [4] in systems modelling.

## The nested sampling algorithm

Nested sampling [4] explores the Bayesian evidence, transforming the multi-dimensional integral for the evidence into a one-dimensional integral over the prior mass. The sorted likelihood function *L(x)* is used as an evolving constraint

A

```
transc = 5.14;
transl = 0.5;
d1 = 1.0;
d2 = 0.25;
M=0;
P=0;
->M,transc;
M->,d1;
M-> M + P,transl;
P->,d2;
```

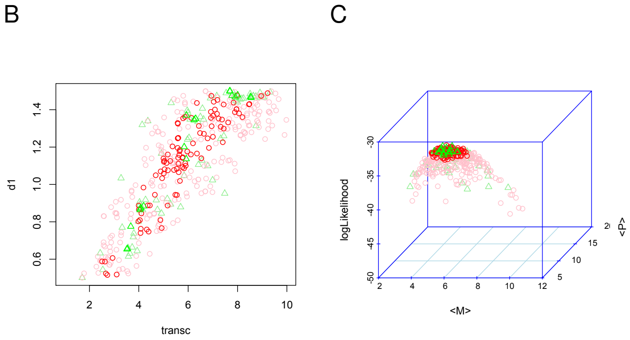


**Fig. 1.** A model of transcription and translation. (A) The model in Dizzy syntax. (B) Analysis of posterior and active samples: values for transc and d1 in the posterior samples (20 samples:light green; 100 samples: pink) and the active samples (20 samples: green; 100 samples: red). The trade-off between transc and d1 is apparent. (C) Mean M (transc/d1) and mean P (transc*transl/d1*d2), colour coding as in (B).

in the generation of a set of objects $x$, randomly sampled from the prior. (An object is an array of parameter values.) Given a set of $n$ active objects, the worst is replaced by a new object, subject to the constraint $L(x) > L^*$ (where $L^*$ is the log likelihood of the worst sample). The new object is discovered by an exploration method that takes one of the remaining *n-1* objects as a starting point. The constraint $L^*$ is then updated, and the process repeats. Nested sampling computes the mean and standard deviation for model parameters based on the set of posterior samples (i.e. those eliminated from the active set). This method promises to be more efficient than MCMC and to cope better with multi-modal posteriors. We report novel initial results on the application of nested sampling to model optimisation in systems biology, and to the optimisation of stochastic models that lack an analytical solution, focusing on the use of the posterior samples generated.

## Model optimisation

A simple model of transcription and translation is defined in Fig. 1A where mRNA ($M$) is transcribed and may be degraded, and protein ($P$) is translated from mRNA and may be degraded. Samples from the distribution of $M$ and $P$ form the data to be fitted, and these samples have been designed to have one known solution. This four parameter optimisation problem permits an informative analysis of the posterior samples to be made.

From the structure of the model, there is clearly a trade-off between the rates *transc* and *d1* in defining *M*. Fig. 1B shows that the posterior and active samples cluster along the diagonal *<M> = transc/d1 = 5.14*. Fig. 1C shows that the mean values of mRNA and protein that can be calculated from the samples have a peak log likelihood at *<M>=5.14* and *<P>=10.28*, and that the posterior and active samples cluster towards the optima. Such analyses have obvious uses should the posterior have multiple modes.

We are exploring the optimisation of more complex systems biology models using nested sampling, and in future work we shall consider model comparison through the Bayesian evidence calculation, and the definition of criteria to assist modellers with the practical application of the method.

**Acknowledgments.**

The Centre for Systems Biology at Edinburgh is a Centre for Integrative Systems Biology (CISB) funded by BBSRC and EPSRC, reference BB/D019621/1. SA is partially supported by a Wellcome Trust Value In People award.

## References

[1] Julio Banga. Optimization in computational systems biology. *BMC Systems Biology*, 2(1):47, 2008.

[2] R. J. Boys, D.J. Wilkinson, and T. B. L. Kirkwood. Bayesian inference for a discretely observed stochastic kinetic model. *Statistics and Computing*, 18(2):125–135, 2008.

[3] Paul Fearnhead. Computational methods for complex stochastic systems: a review of some alternatives to mcmc. *Statistics and Computing*, 18(2):151–171, 2008.

[4] John Skilling. Nested sampling for general Bayesian computation. *Bayesian Analysis*, 1(4):833–860, 2006.

[5] Eberhard O. Voit. Model identification: A key challenge is computational systems biology. In *Optimization and Systems Biology, Lecture Notes in Operations Research 9*, 2008.

# On the applicability of Bayesian univariate methods as filters in complex GWAS analysis

Péter Antal[1], András Gézsi[3], András Millinghoffer[1], Gergely Hajós[1], Csaba Szalai[2], and András Falus[3]

Dept. of Meas. and Inf. Sys., Budapest University of Technology and Economics
Inflammation Bio. and Immunogenomics Research Group, Hungarian Acad. of Sci.
Dept. of Genetics, Cell- and Immunobiology, Semmelweis University, Hungary
antal@mit.bme.hu

**Abstract.** Bayesian methods are more and more widespread in genetic association studies, both in univariate settings for averaging over effect strength (i.e. over parameters), and in complex multivariate models for averaging over relations as well. First we discuss a new impetus for the Bayesian approach, the analysis of probabilistic data, which will be a new major challenge in the analysis of rare variants from new generation sequencing data. Next we discuss a combination of univariate and multivariate Bayesian methods, namely Bayesian networks, which balances between performance and computational complexity. We present results about the performance of this two-step approach using a realistic genome-wide single-nucleotide polymorphism dataset, identify characteristic errors and practical thresholds.

## 1 Introduction

The relative scarcity of the results of genetic association studies (GAS) prompted many research directions, such as the use of the Bayesian framework [4, 14]; and the use of complex models, such as Bayesian networks, which can learn non-transitive(!), multivariate, non-linear relations between target and explanatory variables, treat multiple targets(!), and allow scalable multivariate analysis [2]. Because their applicability in genome-wide association studies is hindered by the high computational complexity, we implemented a two-phased combined method, in which a Bayesian univariate method filters the variables for the subsequent deep analysis for interactions (for the univariate method, see [14]).

## 2 The Bayesian framework for GAS

Due to its direct semantics, the Bayesian approach has an in-built automated correction for the multiple testing problem (i.e. the posterior is less peaked with increasing model complexity and decreasing sample size, see Section 4). From another point of view, the Bayesian statistical framework is ideal for trading sample complexity for computational complexity (i.e. applying computation intensive model-averaging to quantify the sufficiency of the data). Bayesian conditional methods e.g. using logistic regression or multilayer perceptrons, are widely used in biomedicine and in GASs (e.g., see [8, 1, 13, 3, 18]). Although the conditional approach is capable for multivariate analysis and also copes with conditional relevance and interactions, the model-based approach offers many advantages.

1. *Strong relevance.* Clear semantics for the explicit, faithful representation of strongly relevant (e.g. non-transitive) relations.
2. *Structure posterior.* In case of complete data the parameters can be analytically marginalized.
3. *Independence map and causal structure.* It offers a graphical representation for the interactions and conditional relevance, and optionally for the causal relations [17, 10].
4. *Multi-targets.* It is applicable for multiple targets [2].
5. *Incomplete data.* It offers integrated management of incomplete data within Bayesian inference.
6. *Haplotype level.* It can perform inherent haplotype analysis.
7. *Prior incorporation.* It allows better prior incorporation both at parameter and structural levels.
8. *Post fusion.* It offers better semantics for the construction of meta probabilistic knowledge bases [16].

Another recent motive for the Bayesian approach stems from the attempts to cope with rare variants by aggregating them w.r.t. a corresponding gene or pathway (i.e., from prior incorporation). Both nominal and quantitative variables $V_i'$ can be induced based on the original sets of variables $\underline{V}$, using deterministic transformations $V_i' = f_i(\underline{V})$. However, typically, there is a considerable uncertainty over these transformations, and it is more practical to expect a Bayesian transformation, in which each deterministic transformation has a prior distribution $p(F_i = f_i)$. This implicitly defines a conditional distribution for stochastic mapping (assuming discrete $F_i$ for simplicity) $p_i(v_i'|\underline{v}) = \sum_{f_i} p(f_i) f_i(\underline{v})$. The analysis of corresponding distribution over possible datasets $p(D_N'|D_N)$ fits to the Bayesian framework in many respects.

## 3 Combination of univariate and multivariate methods

Bayesian methods are more and more widespread in genetic association studies, particularly in genome-wide analysis of single-nucleotide polymorphisms (see e.g. [4, 14, 11]). The advantage of this approach is partly explained by the averaging over effect strength (i.e. over parameters), however an important open question is the selection of prior, particularly in the new era of rare variants.

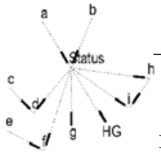
On the other hand the Bayesian framework is popular for complex multivariate models as well, such as for Bayesian networks. The Bayesian inference over structural properties of Bayesian networks was proposed in [5, 6], wich was continued by a series of important extensions [15, 7, 12]. In [2], we introduced the concepts of multitarget relevance, feature aggregation, and scalable multivariate relevance with polynomial cardinality to bridge the gap between the linearity of the MBM level and the exponentiality of the MBS level.

A straightforward combination consists of a filtering univariate and a subsequent multivariate method. We examined the relation of priors both analytically and experimentally.

## 4 Results

We demonstrate the results on an artificial data set generated as follows: (1) We simulated a case-control dataset containing 10000 random samples with HAPGEN [19]. This program can handle markers in LD and can simulate datasets over large regions, such as whole chromosomes. We used the publicly available files that contain the haplotypes estimated as part of the HapMap project, and the estimated fine-scale recombination map derived from that data (HapMap rel#22 - NCBI Build 36 (dbSNP b126)). The generated dataset contained 33815 SNPs. By using HAPGEN we defined a partial disease model as a single disease causing variant, i.e. except this polymorphism, these SNPs defines the background. (2) As we reported earlier [2] the Bayesian network based method is capable to discover complex interactions, such as pure conditional (strong) relevance. In order to create a realistic disease model we manually created a network containing 9 explanatory variables (SNPs). We used this network for generating a complementary artificial dataset. (3) We integrated the two datasets and used the resulting dataset as the input in our analysis. This final dataset therefore contained 33824 SNPs. This dataset also defines the embedded datasets with sizes 500, 1000, 5000 and 10000. The

**Table 1.** (Left:) The Markov Blanket Graph of the outcome (Status) variable. (Middle:) Sensitivity as a function of sample (rows) and model size (columns) at acceptance threshold 0.5. (Right:) Positive predictive value as a function of sample (rows) and model size (columns) at acceptance threshold 0.5.



| | 10 | 20 | 50 | 100 | 200 | 300 |
|---|---|---|---|---|---|---|
| 500 | 0 | 0 | 0.1 | 0.1 | 0.1 | 0 |
| 1000 | 0.1 | 0.1 | 0.1 | 0 | 0.1 | 0.1 |
| 5000 | 0.3 | 0.4 | 0.3 | 0.3 | 0.3 | 0.2 |
| 10000 | 0.5 | 0.6 | 0.6 | 0.6 | 0.5 | 0.5 |

| | 10 | 20 | 50 | 100 | 200 | 300 |
|---|---|---|---|---|---|---|
| 500 | 0.00 | 0.00 | 0.04 | 0.03 | 0.05 | 0.00 |
| 1000 | 0.25 | 0.20 | 0.20 | 0.00 | 0.10 | 0.06 |
| 5000 | 0.60 | 0.50 | 0.19 | 0.50 | 0.60 | 0.33 |
| 10000 | 0.63 | 0.60 | 0.55 | 0.43 | 0.38 | 0.29 |

generated clinical variable (case-control status) served as the target variable, and the aim of our investigation was to identify all the relevant variables w.r.t. this target variable. There are 10 SNPs in total that are relevant, i.e. part of the MBG of the case-control status (see Table 1). To estimate the univariate and multivariate posteriors of relevance we investigated the following settings: (1) We used a Bayesian approach [14] to filter the dataset to 10, 20, 50, 100, 200 and 300 variables. (2) Next, we applied a DAG-based Markov Chain Monte Carlo method. The length of the burn-in and MC simulation were $10^6$ and $5 \cdot 10^6$, the probability of the DAG operators was uniform [9]. The $CH$ parameter prior and the uniform structure prior were used [6]. The maximum number of parents was 5. The rate of sensitivity and the positive predictive value at an acceptance threshold (i.e. the value of MBM probability above which we accept a factor to be relevant) of 0.5 are reported in the following tables.

The external performance measures such as the sensitivity and the positive predictive value in Table 1 indicate the expected trends w.r.t. sample size. There are 4 variables that our method are not able to find in either of the settings. Variables $c$ and $d$ are pure interaction terms that have no marginal effect on the target variable, so the univariate filtering method excludes them on the early stage of the analysis. The loss of $h$ and $i$ variable can be explained by the very small effect (Odds Ratio below 1.2) they have on the target variable.

## 5 Conclusion

Probabilistic graphical models are already widely applied tools in expression data analysis, in pedigree analysis, in linkage and association analysis. In the paper we presented the learning characteristics of a two-step Bayesian method in genetic association studies for systematically varying sample size and number of variables (for availability, see

## References

[1] P. Antal, G. Fannes, D. Timmerman, Y. Moreau, and B. De Moor. Bayesian applications of belief networks and multilayer perceptrons for ovarian tumor classification with rejection. *Artificial Intelligence in Medicine*, 29:39–60, 2003.

[2] P. Antal, A. Millinghoffer, G. Hullám, Cs. Szalai, and A. Falus. A bayesian view of challenges in feature selection: Feature aggregation, multiple targets, redundancy and interaction. *JMLR Proceeding*, 4:74–89, 2008.

[3] D. J. Balding. A tutorial on statistical methods for population association studies. *Nature*, 7:781–91, 2006.

[4] D. J. Balding. *Handbook of Statistical Genetics.* Wiley & Sons, 2007.

[5] W. L. Buntine. Theory refinement of Bayesian networks. In *Proc. of the 7th Conf. on Uncertainty in Artificial Intelligence (UAI-1991)*, pages 52–60. Morgan Kaufmann, 1991.

[6] G. F. Cooper and E. Herskovits. A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 9:309–347, 1992.

[7] N. Friedman and D. Koller. Being Bayesian about network structure. *Machine Learning*, 50:95–125, 2003.

[8] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian Data Analysis.* Chapman & Hall, London, 1995.

[9] P. Giudici and R. Castelo. Improving Markov Chain Monte Carlo model search for data mining. *Machine Learning*, 50:127–158, 2003.

[10] C. Glymour and G. F. Cooper. *Computation, Causation, and Discovery.* AAAI Press, 1999.

[11] Clive J. Hoggart, John C. Whittaker, Maria De Iorio, and David J. Balding. Simultaneous analysis of all snps in genome-wide and resequencing association studies. *PLoS Genetics*, 4(7):e1000130, 2008.

[12] M. Koivisto and K. Sood. Exact bayesian structure discovery in bayesian networks. *Journal of Machine Learning Research*, 5:549–573, 2004.

[13] C. Kooperberg and I. Ruczinski. Identifying interacting snps using monte carlo logic regression. *Genet Epidemiol*, 28(2):157–170, 2005.

[14] Stephens M. and Balding D.J. Bayesian statistical methods for genetic association studies. *Nature Review Genetics*, 10(10):681–690, 2009.

[15] D. Madigan, S. A. Andersson, M. Perlman, and C. T. Volinsky. Bayesian model averaging and model selection for Markov equivalence classes of acyclic digraphs. *Comm.Statist. Theory Methods*, 25:2493–2520, 1996.

[16] A. Millinghoffer, G. Hullám, and P. Antal. On inferring the most probable sentences in bayesian logic. In *Workshop notes on Intelligent Data Analysis in bioMedicine And Pharmacology (IDAMAP-2007)*, pages 13–18, 2007.

[17] J. Pearl. *Causality: Models, Reasoning, and Inference.* Cambridge University Press, 2000.

[18] M. A. Province and I. B. Borecki. Gathering the gold dust: Methods for assessing the aggregate impact of small effect genes in genomic scans. In *Proc. of the Pacific Symposium on Biocomputing (PSB08)*, volume 13, pages 190–200, 2008.

[19] Chris C. A. Spencer, Zhan Su, Peter Donnelly, and Jonathan Marchini. Designing genome-wide association studies: Sample size, power, imputation, and the choice of genotyping chip. *PLoS Genetics*, 5(5), 2009.

# Haplotype- and Pathway-based Aggregations for the Bayesian Analysis of Rare Variants

P. Antal[1], P. Sárközy[1], B. Zoltán[1], G. Hajós[1]
Cs. Szalai[2], and A. Falus[3]

Dept. of Meas. and Inf. Sys., Budapest University of Technology and Economics
Inflammation Bio. and Immunogenomics Research Group, Hungarian Acad. of Sci.
Dept. of Genetics, Cell- and Immunobiology, Semmelweis University, Hungary
antal@mit.bme.hu

**Abstract.** The statistical analysis of rare variants from new generation sequencing methods has become a central challenge. We discuss the hypothesis of "equivalent pathway degrading variants" with similar functional effects, both its single gene aspect arising from the transcription to post-translation chain, and its multivariate pathway aspect arising from cascades and modules. We propose a stochastic aggregation for incorporating uncertain knowledge, and describe and evaluate a method for the Bayesian analysis of uncertain data using Bayesian networks.

## 1 Introduction

New generation sequencing methods are rapidly changing the landscape of the research of common diseases, tumorgenetics, and immunogenetics, some already advocates the era of the "common disease-rare variants" (RV) [6]. However the discovery and use of rare genomic variants with strong effects became a central challenge.

At first in this paper, we summarize a unified approach to common and rare variants in common diseases. Second, we catalogue information sources for the univariate, gene-centered aggregation of variants, starting from transcription regulation to post-translational modifications. We also overview information sources for the multivariate, gene-gene/protein-protein associations and pathway based aggregation of variants. Next we demonstrate a method for the Bayesian analysis of uncertain data in the haplotype analysis of asthma.

## 2 Common and rare variants in common disease

After decades of research of rare variants of rare diseases, common variants (CVs) became central in the research of complex, common diseases.

The slower-than-expected progress and partial results of the corresponding GWAS line resulted in heavy criticism, e.g. the paper of McClellan and King concludes the failure of "the common disease (CD)-common variant (CV) hypothesis". Furthermore, it argues for a systems biology based evaluation of RVs in CDs [6].

To understand the implications of this debate w.r.t. new machine learning methods it is worthwhile to consider the following two aspects. The SNP distribution in the human population is as follows: in a $10^9$ human population, with $10^2$ new germline SNPs per person "every point mutation compatible with life is likely present", while for a pair of genomes CVs give most of the variability. The second such group of findings is that a CD is typically related to the combination of various degradations of more pathways, where the degradations of pathways are caused by both RVs and CVs. In fact, it is probably a tenable hypothesis that the distribution of the effect strength of CVs and RVs are comparable, because the current CVs are formed mainly e.g. by random drift, and proportionally there is no difference in coverage and functional role in pathway degradation. This implies that the majority of variants with strong effect are rare and we can talk about classes of "equivalent pathway degrading variants"
.

## 3 Aggregation of RVs

The weak or non-existing linkage of RVs limits the use of haplotypes or chromosomal regions for aggregations. However this property also limits the possibility of the discovery of non-functional associations.

In the univariate, gene-centered approach RVs can be aggregated along the transcriptional regulations to post-translational modifications chain as follows: transcription factor binding sites, miRNA binding sites, splice-regulatory element binding sites, and phosphorylation and glycosylation related variations and conserved regions.

In the multivariate, pathway degrading approach RVs can be aggregated w.r.t. pathway knowledge bases and gene-gene/protein-protein associations.

In each cases both nominal and quantitative variables $V_i'$ can be induced based on the original sets of variables $\underline{V}$, using deterministic transformations $V_i' = f_i(\underline{V})$. Typically, there is a considerable uncertainty over these transformations, and it is more practical to expect a Bayesian transformation, in which each deterministic transformation has a prior distribution $p(F_i = f_i)$. This implicitly defines a conditional distribution for stochastic mapping (assuming discrete $F_i$ for simplicity)

$$p_i(v'_i|\underline{v}) = \sum_{f_i} p(f_i) f_i(\underline{v}), \tag{1}$$

Note that by incorporating uncertainty over the aggregations, we can balance the increase of the number of transformed variables.

## 4 Pre-processing vs post-processing aggregation

Various numerically and statistically motivated transformation techniques in the data preprocessing phase are abundant, such as for normalization, standardization, and dimensionality reduction. These methods can be very valuable in RV aggregation, although the discrete nature of the data excludes many standard solutions. However the real challenge is to incorporate prior knowledge in RV transformation. The incorporation of such priors in data analysis is already common place, although not in the data preprocessing phase and not for detecting interactions, but in the postprocessing phase, such as in the Gene Ontology annotation analysis or in the Gene Set Enrichment Analysis (for Bayesian aggregation in the postprocessing phase, see [2]).

## 5 Analyzing uncertain data

Because of uncertainty in RV transformation and aggregation, the analysis of uncertain data is a central theme in the analysis of rare variants. We will concentrate on the Bayesian statistical framework for the analysis, particularly because of its ability to incorporate priors and aggregate posteriors [2]. Assuming a distribution over possible datasets $p(D'_N|D_N)$ defined by Eq. 1 various approaches are as follows

1. using only the most probable data set,
2. using multiple data sets with high probability,
3. Monte Carlo data-averaging in Bayesian model-averaging.

The Bayesian averaging over model properties $\alpha(M)$ is done using Metropolis-Hastings algorithms (M-H) [2]. To avoid multiple burn-in in case of multiple data sets, we can mix data-averaging and model-averaging in a joint Metropolis-Hastings scheme, in the M-H-within-Gibbs, which is a hybrid of M-H and Gibbs sampling, the Gibbs sampling steps and the M-H steps can follow each other successively [3]. The Gibbs sampling steps can be used to generate uncertain and missing values, then using the completed data set a structure can be sampled in the M-H step.

## 6 Results

We performed transformations in a partial genetic association study in asthma, both in the gene-centered and in the pathway-centered approach. Here we illustrate our Bayesian network based methods for the Bayesian analysis of uncertain data for haplotypes in the gene-centered approach.

Because of computational reasons such separation of blocking and phasing (haplotype inference [1]), and data analysis is a practical choice followed in many systems (see e.g. HapScope [5], HAPLOT [4], GEVALT [7]). We similarly follow this decomposition, in which the biomedical expert specifies the blocks a priori (corresponding to unrelated chromosomal regions), then the PHASE method is applied for each block [8] to generate a maximum a posteriori phasing and the distribution over possible phased genotyped data, and finally we apply our Bayesian model-based approach [2].

We applied the MC-DA-BMA method in a genetic association study in asthma research investigating 56 SNPs, 15 genes in chromosome 11 and 14 (settings: burn–in: 1000000, step: 5000000, 10 random datasets from phasing distribution).

**Table 1.** BNF maximum likelihood and the averaged results on chromosome 11 and 14. Where HT is the haplotype

| Region Name | average<br>MBM posterior | variance of<br>MBM posterior |
|---|---|---|
| FRMD6 Start HT 1 | 0.638754 | .0969 |
| FRMD6 Start HT 2 | 0.723333 | .1581 |
| AHNAK HT 1 | 0.160561 | .1368 |
| AHNAK HT 2 | 0.611516 | .0713 |

## 7 Conclusions

We focused on the Bayesian statistical framework for the analysis, particularly because of its ability to incorporate priors and to aggregate posteriors [2]. We summarized various types of aggregations of rare variants, proposed and implemented a stochastic aggregation scheme, which can be used both in the preprocessing and postprocessing phases. We implemented various sampling schemes to cope with uncertain data sets using parallel computing information resources (for availability, see

We demonstrated these methods for the Bayesian analysis of uncertain data in the haplotype analysis of asthma.

## References

[1] Clark A.G. Inference of haplotypes from pcr–amplified samples of diploid populations. *Molecular Biology and Evolution*, 7(2):111–122, 1990.

[2] P. Antal, A. Millinghoffer, G. Hullám, Cs. Szalai, and A. Falus. A bayesian view of challenges in feature selection: Feature aggregation, multiple targets, redundancy and interaction. *JMLR Proceeding*, 4:74–89, 2008.

[3] S. Chib and E. Greenberg. Understanding the metropolis-hastings algorithm. *The American Statistician*, 49(4):327–335, 1995.

[4] Kidd K.K. Gu S, Pakstis AJ. Haplot: a graphical comparison of haplotype blocks, tagsnp sets and snp variation for multiple populations. *Bioinformatics*, 21(20):3938–3939, 2005.

[5] William L. Rowe Jinghui Zhang. Hapscope: a software system for automated and visual analysis of functionally annotated haplotypes. *Nucleic Acids Research*, 30(23):5213–5221, 2002.

[6] J. McClellan and MC. King. Genetic heterogeneity in human disease. *Cell*, 141:210–217, 2010.

[7] Ron Shamir Ofir Davidovich, Gad Kimmel. Gevalt: An integrated software tool for genotype analysis. *BMC Bioinformatics*, 8(36):2105–2112, 2007.

[8] M. Stephens and P. Donnelly. A comparison of bayesian methods for haplotype reconstruction from population genotype data. *The American Society of Human Genetics*, 73(5):1162–1169, 2003.

# Large scale learning of combinatorial transcriptional dynamics from gene expression

H.M. Shahzad Asif and Guido Sanguinetti

School of Informatics, University of Edinburgh
{S.Asif,G.sanguinetti}@ed.ac.uk

We present a novel method to infer combinatorial regulation of gene expression by multiple transcription factors in large-scale transcriptional regulatory networks. The method implements a factorial hidden Markov model with a non-linear likelihood to represent the interactions between the hidden transcription factors. We propose a sampling-based inference mechanism as well as an efficient factorised variational approximation which allows application to genome-wide examples. We evaluate the method on a number of synthetic and real data sets, demonstrating the potential insights deriving from understanding combinatorial effects.

# Combining $\ell_1$-$\ell_2$ regularization with biological prior for multi-level hypoxia signature in Neuroblastoma

Annalisa Barla[1], Sofia Mosci[1], Lorenzo Rosasco[1,2], Alessandro Verri[1]
Paolo Fardin[3], Andrea Cornero[3], Massimo Acquaviva[3], and Luigi Varesio[3]

[1] DISI-Università di Genova, Via Dodecaneso 35, Genova, Italy
[2] CBCL, McGovern Institute, Artificial Intelligence Lab, BCS, MIT
[3] Molecular Biology Laboratory, Giannina Gaslini Institute, Genova

**Abstract.** We address the problem of building a signature for hypoxia from microarray data of heterogeneous neuroblastoma cell lines by means of $\ell_1$-$\ell_2$ regularization with double optimization, and integrating prior knowledge from the Gene Ontology (GO) repository. While unsupervised analysis highlights a strong signal that completely masks the more subtle response to hypoxia, a complex mechanisms that plays a crucial role in tumor progress, a Machine Learning based gene selection procedure applied to the entire transcriptome identified a high-level signature of 11 probesets discriminating the hypoxic state. Furthermore, we show that new signatures, with similar discriminatory power to the the high-level one, can be generated by a prior-knowledge based filtering in which a much smaller number of probesets, characterizing hypoxia-related biochemical pathways, are analyzed.

**Keywords:** $\ell_1$-$\ell_2$ regularization, prior knowledge, microarray, signature

## 1 Introduction and Experimental Setting

Machine learning based feature selection techniques have succeeded in analyzing heterogeneous microarray data from tissue samples. Conversely, analysis of cell lines in different conditions is conventionally approached with hypothesis testing. Here we consider a different in *in vitro* experimental design aimed at mimicking the situation occurring in *in vivo* samples by means of heterogeneous cell lines, and show how a machine learning based feature selection approach allows dealing with heterogeneity. In particular we study the application of $\ell_1$-$\ell_2$ regularization as a gene selection technique for detecting the signature characterizing the transcriptional response of neuroblastoma tumor cell lines to hypoxia, a condition of low oxygen tension that occurs in the tumor microenvironment and is negatively correlated with the progression of the disease. In order to mimic the situation occurring in the tumor mass, in which the hypoxia signal is perceived by cells differing in their genetic makeup, differentiation and progression, a set of 9 human neuroblastoma cell lines (GI-LI-N, ACN, GI-ME-N, IMR-32, LAN-1, SK-N- BE(2)C, SK-N-F1, and SK-N-SH) were cultured in a humidified incubator containing 20%, and 1% $O_2$ in normoxic and hypoxic condition, respectively. Microarray gene expressions were measured for 54613 probe sets with Affymetrix HG-U133 Plus 2.0 GeneChip (data are publicly available on the GEO repository under the record GSE17714), whereas more details of the experimental setting as well as the statistical analysis can be found in [6, 7].

## 2 First order analysis

Analysis by hierarchical, spectral, and k-means clustering or supervised approach based on t-test analysis divided the cell lines on the basis of genetic differences. Such result shows that the hypoxic and normoxic statuses, though biologically different, do not induce a modulation of gene expression comparable in magnitude to that induced, for example, by genetic alterations. Therefore, in order to discriminate the two statuses, one has to resort to more powerful techniques, i.e multivariate machine learning techniques, capable of identifying more subtle signals, even when masked by a strong transcriptional response.

## 3 Machine learning analysis

In order to identify the subtle hypoxia response we employ the $\ell_1$-$\ell_2$ regularization framework described in [5]. This technique fulfills all the desirable properties of a variable selection algorithm and is based on supervised learning techniques, enforcing sparsity by means of the $\ell_1$-norm penalty.

### 3.1 The $\ell_1$-$\ell_2$ regularization framework

The method is based on the optimization principle presented in [9] and further developed and studied in [4]. Assume we are given a $n \times p$ matrix built with the gene expression of $p$ genes for $n$ samples, with $p >> n$, and a vector $Y$ of $n$ binary labels. We consider a linear model $y \sim x\beta$, where $\beta = (\beta_1, \ldots, \beta_p)$ is a vector of gene weights. A classification rule can then be defined by taking $sign(x\beta)$. If $\beta$ is sparse then some genes will not contribute in building the estimator. The estimator defined by $\ell_1$-$\ell_2$ regularization solves the optimization problem:

$$\operatorname{argmin}\left\{\|X\beta - Y\|^2 + \tau\|\beta\|_1 + \mu\|\beta\|_2^2\right\}$$

where the least square error is penalized with the $\ell_1$ and $\ell_2$ norm of the weight vector, and the trade-off between the two terms is controlled by the parameter $\mu$. The role of the two penalties is different, the $\ell_1$ term (sum of absolute values) enforces the solution to be sparse, the $\ell_2$ term (sum of the squares) preserves correlation among genes. This approach guarantees consistency of the estimator [4] and enforces the sparsity of the solution by the $\ell_1$ term, while preserving correlation among input variables with the $\ell_2$ term. Differently to [9] we follow the approach proposed in [5], where the $\ell_1$-$\ell_2$ solution, computed through the simple iterative soft-thresholding, is followed by a second optimization, namely regularized least squares (RLS), to estimate the classifier on the selected genes. The parameter $\mu$, fixed a priori, governs the amount of correlation. By tuning $\mu$ we obtain a one-parameter family of solutions which are equivalent in terms of prediction accuracy, but differ on the degree of correlation among the selected features. The training for selection and classification requires the choice of the regularization parameters for both $\ell_1$-$\ell_2$ regularization and RLS, hence model selection and testing are performed within two nested loops of cross validation (see [2] for details). In order to assess a common list of probesets, we select the most stable ones, i.e. the most frequently selected probesets across the lists.

### 3.2 High-level analysis

When applied to the entire hypoxia data, the $\ell_1$-$\ell_2$ regularization distinguishes the normoxic and hypoxic statuses defining an overall signature composed by 11 stable probesets representing 8 genes modulated by hypoxia, and associated with a 17% leave-one-out error. The strong discriminative power of the high-level signature is shown in Figure 3.2 . In order to obtain a 3D representation, the data submatrix is projected on its 3 principal components, i.e. the components of maximum variance. It is evident that two classes of normoxic (blue circles) and hypoxic statuses (red squares) are clearly separated in the multidimensional space. We conclude that $\ell_1$-$\ell_2$ regularization algorithm is able to identify a set of genes that clearly separated the hypoxic from normoxic cell lines even in the case of the disturbance generated by the genetic alterations of the cell lines.

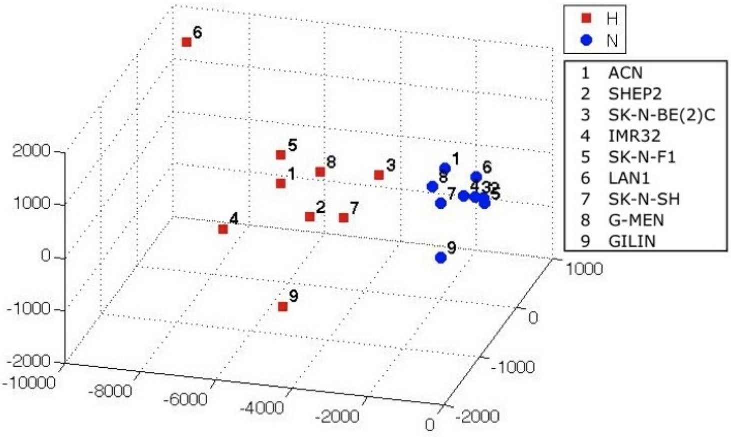


**Fig. 1.** 3-dimensional visualization of the data set restricted to the 11 selected probesets of the high-level signature. Red squares represent the cell lines in hypoxic status and the blue circles those in normoxia.

### 3.3 Pathway-level analysis

We then applied the $\ell_1$-$\ell_2$ regularization framework to subgroups of probesets, characterizing hypoxia-related biochemical pathways. These groups were obtained from the literature and were divided into three categories: hypoxia related [8], MYCN related [3], and neuroblastoma related groups [1]. The only classes associated with a leave one-out error lower than 20% were apoptosis (17%), glycolysis (11%), and oxidative phosphorylation (11%), all of them belonging to the hypoxia group. The new signatures highlight 41 probesets that were not previously included in the high-level signature. Furthermore, the 32 probesets of the oxidative phosphorylation-signature do not overlap with the high-level signature, demonstrating that the increased resolution generated by data filtering allows the identification of previously discarded relevant GO processes.

## 4 Discussion

Our study demonstrated that the $\ell_1$-$\ell_2$ regularization framework outperforms more conventional approaches allowing the definition of an unbiased and objective gene expression signature. The obtained model is able to discriminate between two cell statuses that, albeit biologically very different, do not elicit a modulation of gene expression comparable in magnitude to that induced by genetic alterations. Furthermore a GO based filtering overcomes the noisy nature of microarray data and allows generating robust signatures suitable for biomarker discovery and characterization of complex mechanisms such as hypoxia.

## References

1. S. Asgharzadeh et al. Prognostic significance of gene expression profiles of metastatic neuroblastomas lacking mycn gene amplification. *J Natl. Cancer Inst.*, 98(17), 2006.
2. A. Barla, S. Mosci, L. Rosasco, and A. Verri. A method for robust variable selection with significance assessment. In *Proc. of ESANN*, 2008.
3. E. Bell, J. Lunec, and D. Tweddle. Cell cycle regulation targets of mycn identified by gene expression microarrays. *Cell Cycle*, 6(10):1249–1256, May 2007.
4. C. De Mol, E. De Vito, and L. Rosasco. Elastic-net regularization in learning theory. *J. Complexity*, 25:201–230, 2009.
5. C. De Mol, S. Mosci, M. Traskine, and A. Verri. A regularized method for selecting nested groups of relevant genes from microarray data. *Journal of Computational Biology*, 16, 2009.
6. P. Fardin, A. Barla, S. Mosci, L. Rosasco, A. Verri, and L. Varesio. The l1-l2 regularization framework unmasks the hypoxia signature hidden in the transcriptome of a set of heterogeneous neuroblastoma cell lines. *BMC Genomics*, Jan 2009.
7. P. Fardin, A. Cornero, A. Barla, S. Mosci, M. Acquaviva, L. Rosasco, C. Gambini, A. Verri, and L. Varesio. Identification of multiple hypoxia signatures in neuroblastoma cell lines by l1-l2 regularization and data reduction. *Journal of Biomedicine and Biotechnology*, 2010.
8. A. Harris. Hypoxia–a key regulatory factor in tumour growth. *Nat Rev Cancer*, 2(1):38–47, 2002.
9. Z. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society, Series B*, 67:301–320, 2005.

# Maximum Likelihood and Polynomial System Solving

Kim Batselier and Bart De Moor

Afdeling ESAT, K.U. Leuven
{kim.batselier,bart.demoor}@esat.kuleuven.be

Discrete statistical models are probably one of the most important tools in bioinformatics. Learning the model parameters for these models from observations is commonly done via a maximum likelihood principle. In most cases however there are many solutions to this problem and only a local maximum can be found. The Expectation Maximization algorithm is the method of choice to tackle this problem. A new method is presented that allows to find the global maximum likelihood estimate. The focus is limited on a specific class of discrete statistical models. For these models it is shown that the maximum likelihood estimates correspond with the roots of a multivariate polynomial system. Then, a new algorithm is presented, set in a linear algebra framework, which allows to find all these roots by solving a generalized eigenvalue problem.

# Flow-based Bayesian estimation of differential equations for modeling biological networks

Nicolas Brunel$^{(1)}$ and Florence d'Alché-Buc$^{(1,2)}$

(1) Laboratoire IBISC, Université d'Evry, FRANCE (2) URA 2171, Institut Pasteur, FRANCE

## 1 Introduction

In recent years, there has been a growing interest in identifying complex dynamical systems in biochemistry and biology [15]. In this context, Ordinary Differential Equations (ODEs) have been widely studied for analyzing the dynamics of gene regulatory and signaling networks [11, 14]. In the present work, we consider the problem of estimating parameters and unobserved trajectories in differential equations from experimental data. Nowadays, parameter estimation in differential equations is still considered as a challenging problem when the dynamical system is only partially observed through noisy measurements and exhibit nonlinear dynamics. This is usually the case in reverse-modeling of regulatory and signaling networks [2, 16]. Some approaches address the estimation problem based on a Bayesian estimation of state-space models that integrate the ODE in the evolution equation. However, they suffer from two drawbacks: first they largely neglect the role of the initial condition and second, they assume the gaussianity of the posterior probability distribution of the parameters. In the present work, we are mainly interested in eliminating the first drawback by taking into account that the initial condition is a key parameter of the ODE solution. We propose to provide a proper solution to the ODE by estimating the parameters and the initial conditions. Another contribution is to improve on the Bayesian approach derived in [16] and in [21] by a better approximation of the posterior probability distribution.
We first define the estimation task by introducing the flow of an augmented ordinary differential equation. At this stage, we propose to address the problem with a Bayesian approach, and we approximate the posterior probability by a Population Monte Carlo scheme [7] and [3], consisting in an adaptive selection of the importance distribution.The non-recursive estimation is then applied on two typical systems biology models: the $\alpha$-pinene network [19] and the Repressilator network [8].

## 2 Flow of an ODE and learning of initial conditions

We consider a biological dynamical system, for instance a gene regulatory network, modeled by the following ordinary differential equation:

$$\dot{x}(t) = f(t, x(t), \theta) \qquad (1)$$

defined on the time interval $[0, T]$ $(T > 0)$. $x(t)$ is the state vector of dimension $d$: in the case of a regulatory network, it corresponds to the vector of the expression levels of $d$ genes. $f$ is a (time-dependent) vector field from $\mathbb{R}^d$ to $\mathbb{R}^d$, indexed by a parameter $\theta \in \Theta \subset \mathbb{R}^p$. The flow of a differential equation is defined as the function $\phi_\theta : (t, x_0) \mapsto \phi_\theta(t, x_0)$ which represents the influence of $x(0) = x_0$ on the solution. Now, let us introduce $N$ noisy measurements, $y_n \in \mathbb{R}^m, n = 0...N-1$, that are acquired from a smooth observation function $h : R^d \to R^m$ $(m \geq 1)$ at $N$ times $t_0 = 0 < t_1 < \ldots < t_{N-1} = T$:

$$y_n = h(\phi_\theta(t_n, x_0)) + \epsilon_n \tag{2}$$

where the noise $\epsilon_n$ is supposed to be Gaussian and homoscedastic. If we want to fully identify the ODE, we must estimate both the parameter $\theta$ and the initial condition $x(0)$ so that the solution $\phi_{\hat{\theta}}(\cdot, \hat{x}_0)$ of the system fits the observations $y_{0:N-1} = (y_0, \ldots, y_{N-1})$. When some states are hidden (typically $m < d$), the estimation is particularly difficult and the state-space model interpretation of the couple of equations (1-2) can give efficient algorithms, especially in the Bayesian setting. Our aim is to modify the iterative approach and to show that there is a benefit in jointly estimating $\theta$ and the initial condition $x_0$ in this framework. Despite the little interest of $x_0$ in general applications, it is in fact fundamental to estimate correctly $x_0$ in order to disentangle the influence of the parameter from the one of the initial value. Therefore, we suppose that the initial condition $x_0$ is unknown, so that we are also interested in its estimation. Finally, we want to estimate the augmented initial condition $z_0 = (x_0, \theta) \in R^{d+p}$ of the augmented state ODE model:

$$\begin{cases} \dot{x}(t) = f(t, x(t), \theta(t)) \\ \dot{\theta}(t) = 0 \end{cases} \tag{3}$$

with initial condition $z_0 = (x_0, \theta)$. The solution is the function $t \mapsto \phi(t, z_0)$ from $[0, T]$ to $R^{p+d}$.

## 3 Flow-based Bayesian Estimation and PMC

We consider the Bayesian inference framework for the estimation of the augmented initial condition. We call Flow-based Bayesian Estimation (FBE), the Bayesian approach that consists in estimating the augmented initial condition. If $\epsilon_n$ is Gaussian, the posterior distribution can be written as follows

$$\pi_{N-1}(z_0) = p(z_0 | y_{0:N-1}) \propto \exp\left(-e(y_{0:N-1}, z_0)\right) \pi_{-1}(z_0) \tag{4}$$

where $e(y_{0:N-1}, z_0) = \frac{1}{2\sigma^2} \sum_{n=0}^{N-1} \|y_n - h(\phi(t_n, z_0))\|^2$ and $\pi_{-1}$ is the prior distribution. Bayesian inference relies on the computation of a reliable approximation of $\pi_{N-1}(z_0)$, from which we can derive Bayesian estimators. In non-linear state-space models, the computation of this posterior probability can be done efficiently by recursive smoothing algorithms [5]. These classical algorithms are based on recursive computations of the filtering probabilities $p(z_n | y_{0:n})$ and several versions do exist for the computation of the smoothing probabilities

$p(z_n|y_{0:N-1})$. However, in these algorithms, the initial condition is estimated as an initial state and not as a parameter of the flow. As a consequence, the use of refined smoothing strategies remains problematic and calls for careful adaptations ([13, 10]). Moreover, the smoothed trajectories by smoothing are not solutions of the ODE; therefore, it might be preferable to turn to a non-recursive estimation of the (augmented) initial condition based. To test our hypothesis about the potential interest of a better estimation of the initial conditions in a Bayesian setting, we need to estimate the posterior distribution probability defined in (4). Several general simulation methods have been developed such as Markov Chain Monte Carlo (MCMC), Importance Sampling (IS) and variants [18] are commonly used and both are well-suited to the Bayesian setting. However, one difficulty of this Monte Carlo methods is that they can be very (computationally) intensive. A challenging difficulty of ODE learning is that the evaluation of the likelihood is costly due to the integration of the ODE. This point motivates us to focus on importance sampling algorithms. Population Monte Carlo (PMC) is a sequential Monte Carlo method, i.e. it is an iterated Importance Sampling Resampling algorithm (ISR) which sequentially moves and re-weights a population of weighted particles $(\xi_i, \tilde{\omega}_i), i = 1, \ldots, M$.
In all generality, a PMC scheme is defined for $t = 0, 1, \ldots, T$ and a sequence of proposal distributions $q_t$ defined on $(R^{d+p})$ The essential interest of PMC is to introduce a sequence of proposal distributions that are allowed to depend on all the past which enables to consider adaptive IS procedure based on the performance of the previous populations. A simple way to randomly perturb a population is to add an independent noise to each particle $\xi_{i,t-1}$, i.e. to modify independently each particle $\xi_{i,t} = \xi_{i,t-1} + \epsilon_{i,t}$ with $\epsilon_{i,t} \sim N(0, \Sigma_t)$ (usually $\Sigma_t = \sigma_t^2 I_{d+p}$). Then, at each iteration $t$, we have $\xi_{i,t} \sim N(\xi_{i,t-1}, \Sigma_t)$. General moves from $\xi_{i,t-1}$ to $\xi_{i,t}$ are described with a (Markov) transition kernel $K_{i,t}(\xi_{i,t-1}, \cdot)$. Through the resampling mechanism, particles moving in good regions are duplicated and particles moving to low credibility regions do vanish which permits a global amelioration of the population. This evolution rule described above is a simple random walk, and the mean size of the jumps is controlled by $\sigma_t$. It is interesting to propose at least several size of jumps by using a mixture of $D$ Gaussian transition kernels: $\epsilon_{i,t} \sim \sum_{d=1}^{D} \alpha_d N(0, \Sigma_{d,t})$. With such a D-kernel, the population is moved at each iteration $t$ at different speed $\Sigma_{d,t}$ selected with probability $\alpha_d$. We implement an adaptive kernel that changes the move according to the survival rate of a given move, that can be seen as determining the weights of the mixture of kernel proposals. This estimation problem of the weights $\alpha_d$ is solved by minimizing a Kullback-Leibler divergence with an EM-like algorithm [7].

## References

[1] Rodriguez-Fernandez, M., Egea, J.A., Banga, J.R.: Novel metaheuristic for parameter estimation in nonlinear dynamic biological systems.BMC Bioinformatics 7(483) (2006)

[2] Calderhead, B., Girolami, M., Lawrence, N.D.: Accelerating bayesian inference over nonlinear differential equations with gaussian processes. In: Koller, D., Schuurmans, D., Bengio, Y., Bottou, L. (eds.) Advances in Neural Information Processing Systems 21, pp. 217–224. MIT Press (2009)

[3] Cappé, O., Douc, R., Guillin, A., Marin, J.M., Robert, C.P.: Adaptive importance sampling in general mixture classes. Statistics and Computing 18(4), 447–459 (2008)

[4] Cappé, O., Guillin, A., Marin, J.M., Robert, C.P.: Population monte carlo. Journal of Computational and Graphical Statistics 13(4), 907–929 (2004)

[5] Cappé, O., Moulines, E., Rydén, T.: Inference in Hidden Markov Models. Springer-Verlag (2005)

[6] d'Alché-Buc, F., Brunel, N.J-B.: Learning and inference in computational systems biology. chap. Estimation of parametric nonlinear ODEs for biological networks identification. MIT Press (2010)

[7] Douc, R., Guillin, A., Marin, J.M., Robert, C.: Convergence of adaptive mixtures of importance sampling schemes. Annals of Statistics 35(1), 420–448 (2007)

[8] Elowitz, M., Leibler, S.: A synthetic oscillatory network of transcriptional regulators. Nature 403, 335–338 (2000)

[9] Gentle, J.E., Hardle, W., Mori, Y.: Handbook of computational statistics: concepts and methods. Springer (2004)

[10] Ionides, E., Breto, C., King, A.: Inference for nonlinear dynamical systems. Proceedings of the National Academy of Sciences 103, 18438–18443 (2006)

[11] de Jong, H.: Modeling and simulation of genetic regulatory systems: A literature review. Journal of Computational Biology 9(1), 67 – 103 (2002)

[12] Li, Z., Osborne, M.R., Prvan, T.: Parameter estimation of ordinary differential equations. IMA Journal of Numerical Analysis 25, pp. 264–285 (2005)

[13] Liu, J., West, M.: Combined parameter and state estimation in simulation-based filtering. In: Doucet, A., de Freitas, N., Gordon, N. (eds.) Sequential Monte Carlo Methods in Practice, pp. 197–217. Springer-Verlag (2001)

[14] Mendes, P.: Learning and inference in computational systems biology. chap. Comparative Assessment of Parameter Estimation and Inference Methods. MIT Press (2010)

[15] N.Lawrence, Girolami, M., Rattray, M., Sanguinetti, G.: Learning and Inference in Computational Systems Biology. MIT Press (2010)

[16] Quach, M., Brunel, N., d'Alché-Buc, F.: Estimating parameters and hidden variables in non-linear state-space models based on odes for biological networks inference. Bioinformatics 23(23), 3209–3216 (2007)

[17] Ramsay, J.O., Hooker, G., Campbell, D., Cao, J.: Parameter estimation for differential equations: A generalized smoothing approach. Journal of the Royal Statistical Society, Series B 69, 741–796 (2007)

[18] Robert, C.P., Casella, G.: Monte Carlo Statistical Methods. Springer (2004)

[19] Rodriguez-Fernandez, M., Egea, J.A., Banga, J.R.: Novel metaheuristic for parameter estimation in nonlinear dynamic biological systems. BMC Bioinformatics 7(483) (2006)

[20] Sitz, A., Schwarz, U., Kurths, J., Voss, H.: Estimation of parameters and unobserved components for nonlinear systems from noisy time series. Physical review E 66, 016210 (2002)

[21] Sun, X., Jin, L., Xiong, M.: Extended kalman filter for estimation of parameters in nonlinear state-space models of biochemical networks. PLoS ONE 3(11), e3758+ November (2008).

# Module Network-Based Classifiers for Redundant Domains

Borja Calvo[1,⋆] and Rubén Armañanzas[2,⋆]

[1] Intelligent Systems Group - borja.calvo@ehu.es
[2] Computational Intelligence Group - Cajal Blue Brain Project - r.armananzas@upm.es

## 1 Introduction

With the explosion in the amount of available data in domains such as biology and biomedicine, new problems arise in the induction of classification functions from data. As an example, the number of gene expression measured in each microarray experiment has grown from a few thousands to more than a million in less than ten years. In such situations, learning classifiers from the whole data is unfeasible, not only because of the computational overhead, but also because of the abundant presence of irrelevant and redundant variables that may have negative effects on the quality of the estimations.

Classically, if two (or more) variables provide the same information about the class, the dimensionality reduction algorithms only retain one of them and discard the rest. By doing so, we are actually discarding potentially valuable information, specially in scenarios with few instances from where statistical estimators are computed. In this work, we propose the use of the redundant information in the data to obtain more robust estimations and, therefore, more robust classification functions. In essence, the root idea is to adapt the paradigm of module networks (1) to create modules of redundant variables. Each module will map the same –or almost similar– conditional probability distributions given the class variable. Under this assumption, we can pool the data from all the variables in a module and then use the joint data to estimate the conditional probability distribution.

## 2 Module Network Classifiers

We propose the use of module networks (1) to create classification models that exploit the redundancy of the data. As in the case of classical Bayesian networks (BN) classifiers, having the class variable as parent of all the predictive variables simplifies the computation of the conditional probabilities used for the classification of new instances ($P(C = c|\mathbf{X} = \mathbf{x})$). Analogously, the class variable is set apart in a special module, the *class module*. The only variable of this module,

[⋆] Both authors have equally contributed to the paper.

the class variable, will be set as parent of all the modules in the model. Therefore, the class module will be the root node of the structure, without no possible parent in its case.

In order to check the potentialiaty of MNs as classifiers, we propose the simplest module network classifier (MNC), the equivalent to the naïve Bayes model in BN classifiers (2). Following the naïve Bayes paradigm, the parent list part of the structural learning will be fixed, having all the modules only the class variable as parent. However, in module networks the structural learning also involves the induction of the assignation function ($\mathcal{A}$). In a naïve Bayes module network classifier (nMNC), the $\mathcal{A}$ function has to group together variables with the same (or similar) conditional probability distributions given the class.

The simplicity of the naïve structure perfectly fits situations where few instances are available (in comparison with the number of variables). More complex models need to estimate more complex set of conditional parameters and, thus, more cases are needed to have robust estimations.

### 2.1 Learning naïve module network classifiers from data

In order to induce a nMNC form a set of examples, we need to define the assignation function that groups together variables that have similar conditional probability distributions. This is equivalent to assigning two variables $X_i$ and $X_j$ to the same module if $P_{(X_i,C)}(k,c) = P_{(X_j,C)}(k,c)$ and $r_i = r_j$ with $i = 1\, ldots, r_i$, $c = 1, \ldots, r_C$. The most straightforward approach to tackle this problem is by directly measuring the distance between these two joint probability distributions using the Kullback-Leibler divergence (3). The algorithm starts with an assignation function for which each variable is in a different module, i.e., $\mathcal{A}(X_i) = i$. Then, iteratively the algorithm merges the two modules[3] with the smallest symmetrical divergence. The merging process is repeated until the smallest distance between any two modules is greater than a given threshold.

## 3 Experimental Evaluation

We have tested the proposed algorithm using two real-life datasets. Both belongs to the bioinformatics domain where the special features of module-network classifiers can be exploited. The classifiers we have compared are the proposed nMNC and its classical counterpart, the naïve Bayes classifier (nB).

The validation of the classifiers were done using three cross-validation schemes: i) Ten times ten-fold (10x10 CV); ii) Leave-one-out (LOO CV); and, iii) Single ten-fold (10 CV). LOO CV tends to overfit specially in domains with a limited number of samples. In contrast, a repeated $k$-fold scheme has demonstrated good generalization power in the same dimensionality scenarios (4). The single run of the $k$-fold is included to illustrate the possible variance in the results if only a single validation is performed.

[3] The class module is not considered in this search.

The firs dataset is a proteomics database described in (5). Figure 1 shows the results obtained for this dataset. As we can see in the figure, the use of module networks does not improve the results when all the variables are used. However, when the irrelevant attributes are removed, grouping together similar variables increase the accuracy of the final classifier.

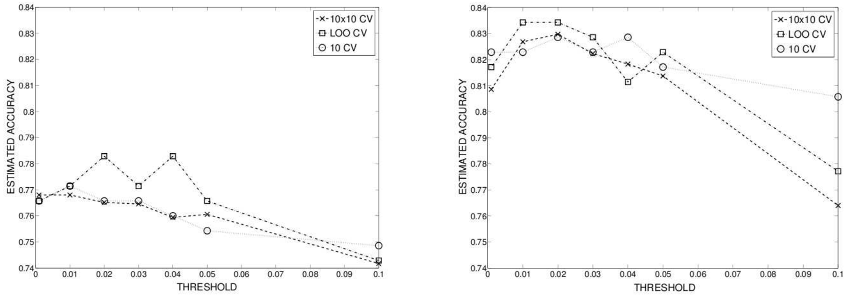


**Fig. 1.** Estimated mean accuracy estimations of the naïve module network classifier when the divergence threshold ranges from 0.001 to 0.1. Left plot shows the values using the original 120 features, whereas right plot gathers the correspondent values when the database is cleaned of irrelevant features. As indicated in the legend, three different validation schemes were performed.

In addition to the improvement in accuracy, the key feature of grouping together variables with similar probability distributions offers a step-forward from the naive structure of the classical paradigm (figures not shown).

The second dataset is the Leukemia database, a well known genomics benchmark (6). The results obatined in this dataset can be seen in Figure 2 As in the proteomics dataset, although the gain of using module networks is not clear when all the datasets are used, there is a moderate increasement of accuracy when the irrelevant features have been removed.

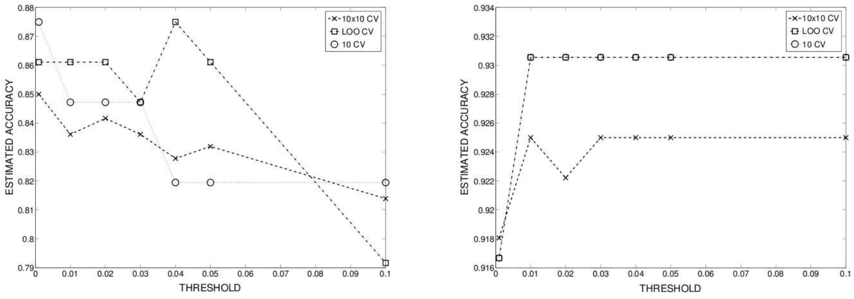


**Fig. 2.** Estimated mean accuracy of the naïve module network classifier when the divergence threshold ranges from 0.001 to 0.1. Left plot shows the values using the initial 1,161 features, whereas right plot gathers the correspondent values when the database is cleaned of irrelevant features. As indicated in the legend, three different validation schemes were performed.

## 4 Conclusions and Future Work

In this work we have explored the potential of using module networks to construct probabilistic classifiers in redundant datasets. Although this is a prelimiary study, we have shown that using module networks as classifiers can increase the accuracy of the learned models in redundant domains. However, there is a number of open questions that require further analysis:

1. Develop a computationally efficient algorithm to learn naïve module network classifiers from data.
2. Extend this idea to more complex models where dependecies between predictive variables are allowed.
3. Theoretical and empirical analysis of the proposed models and the use of module networks as classifiers.

# Bibliography

[1] Segal, E. et al.: Learning module networks. Journal of Machine Learning Research **6** (2005) 557–588

[2] Minsky, M.: Steps toward artificial intelligence. Proceedings of the Institute of Radio Engineers **49** (1961) 8–30

[3] Cover, T.M., Thomas, J.A.: Elements of Information Theory. 2 edn. John Wiley & Sons, Inc. (2006)

[4] Statnikov, A. et al.: A comprehensive evaluation of multicategory classification methods for microarray gene expression cancer diagnosis. Bioinformatics **21**(5) (2005) 631–643

[5] Ray, S. et al.: Classification and prediction of clinical Alzheimer's diagnosis based on plasma signaling proteins. Nature Medicine **13**(11) (2007) 1359–1362

[6] Golub, T.R. et al.: Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. Science **286**(5439) (1999) 531–537

# Query-based Biclustering of Gene Expression Data using Probabilistic Relational Models

Lore Cloots[1*], Hui Zhao[1*], Tim Van den Bulcke[2], Yan Wu[1], Riet De Smet[1], Valerie Storms[1], Pieter Meysman[1], Kristof Engelen[1] and Kathleen Marchal[1]

[1] Department of Microbial and Molecular Systems, K.U.Leuven, Kasteelpark Arenberg 20, 3001 Leuven, Belgiun
[2] i-ICT, Universitair Ziekenhuis Antwerpen, Wilrijkstraat 10, 2650 Edegem, Belgium
{lore.cloots, hui.zhao, yan.wu, riet.desmet, valerie.storms, pieter.meysman, kristof.engelen, kathleen.marchal}@biw.kuleuven.be; tim.van.den.bulcke@uza.be

**Abstract.** With the availability of large scale expression compendia it becomes possible to retrieve genes with an expression profile similar to a set of genes of interest (*i.e.*, seed or query genes), in a subset of relevant experiments. To that end, a query-based strategy is needed that maximally exploits the coexpression behaviour of the seed genes, but that at the same time is robust against the presence of noise in the seed set. Therefore, we developed *Pro*Bic, a query-based biclustering method based on Probabilistic Relational Models (PRMs) that exploits the use of prior distributions to extract the information contained within the seed set. We applied *Pro*Bic on an *Escherichia coli* compendium and compared its performance with that of previously published query-based biclustering algorithms, QDB and ISA. *Pro*Bic is developed in a flexible framework, and detects biologically relevant, high quality biclusters that maintain relevant seed genes, even in the presence of noise.

**Keywords:** Query-based biclustering, gene expression compendia, probabilistic relational models (PRM).

## 1 Introduction

With the large body of publicly available gene expression data, compendia are being compiled that assess gene expression in a plethora of conditions and perturbations [1]. Comparing own experimental data with these large scale gene expression compendia allows viewing own findings in a more global cellular context. To this end query-based biclustering techniques [2-6] can be used that combine gene with condition selection to identify genes that are coexpressed with genes of interest (*i.e.*, seed set or seed genes). These algorithms do not only differ from each other in their search strategy but also in the way they exploit the expression signal embedded in the seed genes. Some algorithms only use the mean seed profile to initialize the search [3]

[*] These authors contributed equally to this work.

while others also impose similarity between the biclusters and seed mean profiles during the iterations of the algorithm [5].

For a user it is important that the obtained query-based biclusters recapitulate as much as possible the information from the seed genes, but not at the expense of the quality of the obtained bicluster results. However, when seed sets are compiled from the output of experimental assays, it can not be guaranteed that all genes within the seed will be tightly coexpressed. When in such case relying too heavily on the seed profile to steer the biclustering, the results will become deteriorated as the algorithm is not able to let the data compensate for a non perfect seed profile. On the other hand when not sufficiently exploiting the seed information, the biclustering might miss its purpose as the seed genes will be lost. To tune in a flexible way the level to which the user wants the seed knowledge to be weighted in the final result we developed a query-based biclustering method called *Pro*Bic, in the framework of probabilistic relational models [7-9]. Seed information is exploited via a Bayesian prior. We compared our algorithm with two of the best state-of-the art query-based biclustering algorithms on a real case study in *Escherichia coli*.

## 2 Model framework

An overview of the *Pro*Bic probabilistic relational model is shown in Figure 1: it contains the classes Gene, Array and Expression. For each class, a set of specific gene, array and expression objects exists (denoted by the lowercase letters *g*, *a* and *e* respectively). The complete set of genes, array and expression objects that belong to a certain class are indicated by uppercase letters *G*, *A* and *E*. For the Gene (Array) class, a Boolean attribute $B_b$ indicates whether a gene (array) belongs to a bicluster *b* or not. For each gene (array) object, the gene-bicluster labels $g.B_b$ (over all biclusters *b*) and the array-bicluster labels $a.B_b$ are the hidden variables of the model. Each object *e* of the Expression class has one single numeric attribute *e.level* that contains the expression level for each specific gene and array combination. The array class has an additional attribute ID that uniquely identifies each individual array object *a*. The conditional probability distribution $P(e.level|e.gene.B,e.array.B,e.array.ID)$ is modeled as a set of Normal distributions, one for each array-bicluster combination. A number of marginal distributions $P(a.B_b)$, $P(g.B_b)$ and $P(g.B)$ allow expert knowledge to be introduced in the model. To learn the model, we applied a hard-assignment EM approach [10]. As an initialization of the hidden variables (the gene to bicluster and array to bicluster assignments) a set of seed genes is used.

## 3 Results and discussion

We used *Pro*Bic to search for genes tightly coexpressed with known regulons, both simple and complex, in *E. coli* [11]. Known regulons were used as seed genes and additional genes retrieved in the resulting biclusters were considered potential undocumented targets for the regulon's associated transcription factor(s). We benchmarked our method with other query-based biclustering algorithms for which a high performance on real datasets was shown previously, *i.e.*, QDB [5] and ISA [6]. To assess the results, we evaluated the expression quality of the obtained biclusters, their biological relevance and examined the way the different algorithms coped with the seed genes.

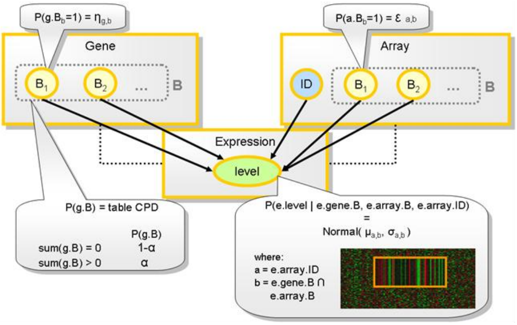


**Fig. 1:** overview of the *Pro*Bic model.

### 3.1 Performance of the algorithms as a function of the seed quality

In case a seed set is informative for the queried expression dataset (meaning that the dataset indeed contains additional genes being coexpressed with the seed), all algorithms are able to find biclusters that maintain the seed genes or at least part of it and include additional genes. For seed sets that are non-informative for the queried dataset, ISA has a large number of biclusters that loose their seed genes, whereas both *Pro*Bic and QDB tend to keep (a part of) the seed genes in their biclusters. Also was found that high quality seed sets give high quality bicluster results (*i.e.*, tightly coexpressed genes whose profile differs from the noise level). In addition, whereas the quality of QDB biclusters depends largely on the quality of the seed set, the expression quality of bicluster results of both *Pro*Bic and ISA are much less sensitive towards quality changes.

### 3.2 Difference between the algorithms in handling noisy seed genes

To systematically analyze the robustness of the different algorithms against the presence of noisy genes in a seed set we designed simulated experiments whereby a certain number of random genes was added to five seed sets and assessed to what extent the different algorithms were able to remove these noisy seed genes from the complete seed set in order to retrieve a bicluster that was centered around the true seed genes. From the experiments could be concluded that *Pro*Bic is most robust against the presence of noisy seed genes.

### 3.3 Relevance of the obtained biclusters

To assess the biological relevance of the obtained biclusters obtained using the seed sets and the extent to which they recapitulated the original regulon from which the seed genes were derived, we calculated functional enrichment and motif enrichment. We found that both ISA and *Pro*Bic largely outperform QDB at the level of motif enrichment and functional overrepresentation. Biclusters retrieved by ISA and *Pro*Bic show a comparable motif enrichment and a slightly better functional enrichment for those derived from ISA than for those obtained by *Pro*Bic.

## 4 Conclusions

*Pro*Bic is a query-based biclustering algorithm, designed to detect biologically relevant, high quality biclusters that retain their seed genes even in the presence of noise or when dealing with low quality seeds. In addition, the underlying PRM based framework is extendable towards integrating additional data sources such as motif information, ChIP-chip information that can further help refining the obtained biclusters.

## References

1. Fierro, A.C., Vandenbussche, F., Engelen, K., Van de Peer, Y., Marchal, K.: Meta Analysis of Gene Expression Data within and Across Species. Curr. Genomics. 9, 525-534 (2008)
2. Owen, A.B., Stuart, J., Mach, K., Villeneuve, A.M., Kim, S.: A gene recommender algorithm to identify coexpressed genes in C. elegans. Genome Res. 13, 1828-1837 (2003)
3. Bergmann, S., Ihmels, J., Barkai, N.: Iterative signature algorithm for the analysis of large-scale gene expression data. Physical review. E.67, 031902-1-031902-18 (2003)
4. Wu, C.J., Kasif, S.: GEMS: a web server for biclustering analysis of biclustering data. Nucleic Acids Res. 33, W596-W599 (2005)
5. Dhollander, T., Sheng, Q., Lemmens, K., De Moor, B., Marchal, K., Moreau, Y.: Query-driven module discovery in microarray data. Bioinformatics. 23, 2573-2580 (2007)
6. Hibbs, M.A., Hess, D.C., Myers, C.L., Huttenhower, C., Li, K., Troyanskaya, O.G.: Exploring the functional landscape of gene expression: directed search of large microarray compendia. Bioinformatics. 23, 2692-2699 (2007)
7. Koller, D., Pfeffer, A.: Probabilistic frame-based systems. In: Proceedings of the Fifteenth National Conference on Artificial Intelligence, pp. 580-587. Madison (1998)
8. Friedman N, Getoor L, Koller D, Pfeffer A: Learning probabilistic relational models. In: International Joint Conference on Artificial Intelligence, pp. 1300-1309. Stockholm (1999)
9. Getoor, L., Friedman, N., Koller, D., Taskar, B.: Learning probabilistic models of relational structure. In: Proceedings of the 18th International Conference on Machine Learning, pp. 170-177. San Francisco (2001)
10. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society series B. 39, 1-38 (1977)
11. Gama-Castro, S., Jimenez-Jacinto, V., Peralta-Gil, M., Santos-Zavaleta, A., Penaloza-Spinola, M.I., Contreras-Moreira, B., Segura-Salazar, J., Muniz-Rascado, L., Martinez-Flores, I., Salgado, H., Bonavides-Martinez, C., Abreu-Goodger, C., Rodriguez-Penagos, C., Miranda-Rios, J., Morett, E., Merino, E., Huerta, A.M., Trevino-Quintanilla, L., Collado-Vides, J.: RegulonDB: gene regulation model of Escherichia coli K-12 beyond transcription, active (experimental) annotated promoters and Textpresso navigation. Nucleic Acids Res. 36, D120-124 (2008)

# Multi-label prediction of enzyme classes using InterPro signatures

Luna De Ferrari[1], Stuart Aitken[2], Jano van Hemert[3], Igor Goryanin[1]

[1]Computational Systems Biology and Bioinformatics, [2]Artificial Intelligence Applications Institute, [3]UK National e-Science Centre - School of Informatics, University of Edinburgh

**Abstract.** In this work we use InterPro protein signatures to predict enzymatic function. We evaluate the method over more than 300,000 proteins (55% enzymes, 45% non-enzymes) for which Swiss-Prot and KEGG have agreeing Enzyme Commission annotations. We applied multi-label classification to account for proteins with multiple enzymatic functions (about 3% of UniProt) using Mulan, a library of algorithms based on the Weka framework. We achieved $> 97\%$ recall, accuracy and precision in predicting enzymatic classes. To understand the role played by the data set size, we compared smaller data sets, either random or specific to taxonomic domains such as archaea, bacteria, fungi, invertebrates, plants and vertebrates. We find that the success of prediction increases with data set size. Limiting the data to a particular taxonomic set, while saving computational time, only covers a reduced set of enzymatic classes and achieves better accuracy than a random set only if the proteins are grouped by high level taxonomic domains (archaea, bacteria and eukaria).

## 1 Background

Manual curation will never complete the functional annotation of all available proteomes, at the current rate of genome sequencing [2]. A contributing problem is the annotation of proteins with enzymatic reactions, that is, to assign to each protein the chemical reactions it is able to perform. We propose and evaluate a method to automatically assign one or more enzymatic functions to a protein. InterPro signatures are among the highest contributors to the performance of protein function prediction methods [7]. InterPro is a database of conserved sequence signatures and domains: any sequence can be scanned in silico for the presence of InterPro signatures, using the InterProScan algorithm. In this work we aim at predicting the Enzyme Commission (EC) numbers of any sequenced protein with high accuracy, recall and precision, using *exclusively InterPro signatures* as protein attributes. Our method differs from work such as [4], which predicts enzyme families with support vector machines, in allowing *multi-label classification*, that is, the association of multiple enzymatic functions to each protein. Our approach is widely applicable, since it exclusively uses information contained in the protein sequence, as opposed to methods such as [3] which also require existing or derived structural information. The work closest to ours is probably [1] where hierarchical classification was applied to about 6000 enzymes, obtaining about 85% accuracy in predicting EC numbers. The objective of that work, though, was to validate a particular algorithm. We used Mulan [8] instead: an open-source library of published multi-label algorithms built on the Weka framework [10], to make our method independent from a particular algorithm implementation. In addition, we obtained extremely good results (over 97% correct predictions) over a very extensive real-life data set consisting of about 300,000 manually curated protein entries.

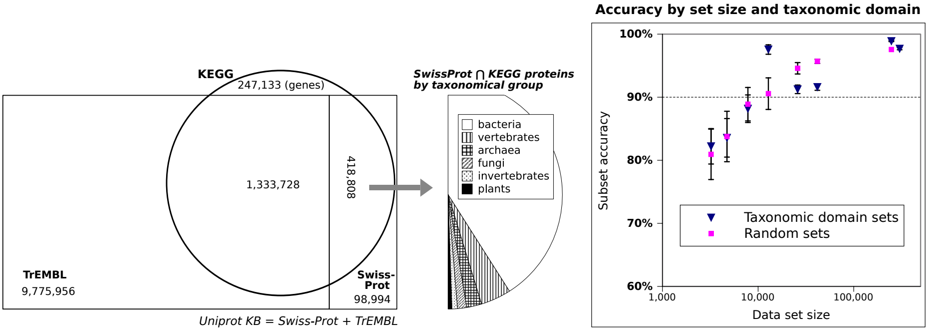


**Fig. 1.** ***Left Figure:*** *The shared protein content of UniProt and KEGG. The circle represents KEGG; the left rectangle (TrEMBL) plus the right (Swiss-Prot) compose the UniProt Knowledge Base. The intersection between Swiss-Prot and KEGG has been expanded to show the distribution of taxonomic groups. For legibility, the areas in the pseudo Venn diagram are not exactly proportional to the number of proteins.* ***Right Figure:*** *Cross-evaluation results: comparison between taxonomic and random sets. The five taxonomic sets (triangles) above the dashed 90% mark are, from left to right: archaea, vertebrates, eukaria, bacteria and Swiss-Prot⋈KEGG. More details in Table 1.*

## 2 Method and data sources

ML-kNN [11] had the best prediction results among the algorithms existing in Mulan version 1.2.0 (data not shown). It was also among the fastest on our data: about 12 hours per fold of a 10-fold cross-evaluation of 300,000 instances, on a dedicated machine with 2.00GHz CPU and 2GB RAM. ML-kNN is a multi-label, lazy-learning approach algorithm derived from the traditional K-Nearest Neighbour. The best choice for the number of neighbours was k=1 (data not shown). For baseline, we used the Zero Rule algorithm. Each instance in our data represents a UniProt protein, having as class label one or more Enzyme Commission (EC) numbers and as attributes the presence of one or more InterPro signatures (protein domains, catalytic sites, sequence repeats etc.). Only 51% of KEGG entries and 11% of UniProt entries are annotated with EC numbers, so, in order to include non-enzymes in our classification task, we interpreted the lack of EC annotation as *lack of enzymatic activity*. Hence we assigned a "0.0.0.0" pseudo EC number to all the KEGG and UniProt entries without EC annotation. In addition, we included in our data sets incomplete EC classes (such as 1.-.-.- or

1.2.-.-). The main data set, from now on indicated as ’*SwissProt* $\bowtie$ *KEGG*’ consists of all protein instances 1. having EC annotations *agreeing* in both Swiss-Prot and KEGG (an annotation being a couple in the form [UniProt Accession Number, EC number]) and 2. having at least one InterPro signature. This extensive data set has been submitted to *two independent manual curations,* in which none of the authors were involved. The set contains 302,068 distinct UniProt protein records (166,426 enzymes and 135,642 non enzymes). The protein instances in this data are sparse, having an average of 3.55 InterPro signatures (attribute values) and 3.97 EC numbers (class labels) per protein. The data was taken from UniProt Knowledge base release 2010_07 of 15-Jun-2010 (sum of Swiss-Prot release 2010_07 and TrEMBL release 2010_07), InterPro release 27.0, KEGG release 55.0 (1-Jul-2010) and ExPASy ENZYME database (release 15-Jun-2010). The data was further processed using Ondex [5,6] and MySQL.

| *Data set* | *Instances (proteins)* | *Attributes (InterPro signatures)* | *Class labels (EC numbers)* | *Average subset accuracy* | *Subset accuracy Std Dev* |
|---|---|---|---|---|---|
| Plants | 3,222 | 2,785 | 611 | 82.2% | 2.8% |
| Random | 3,222 | 3,650 | 678 | 80.9% | 4.0% |
| Invertebrates | 4,723 | 3,886 | 714 | 83.6% | 3.1% |
| Random | 4,723 | 4,320 | 804 | 83.8% | 4.0% |
| Fungi | 7,822 | 4,088 | 796 | 88.2% | 2.2% |
| Random | 7,822 | 5,236 | 935 | 88.9% | 2.7% |
| Archaea | 12,807 | 2,596 | 501 | 97.6% | 0.7% |
| Random | 12,807 | 6,139 | 1,069 | 90.6% | 2.5% |
| Vertebrates | 25,903 | 7,128 | 1,240 | 91.2% | 0.7% |
| Random | 25,903 | 7,729 | 1,316 | 94.6% | 0.9% |
| Eukaria | 41,670 | 8,576 | 1,594 | 91.6% | 0.5% |
| Random | 41,670 | 8,859 | 1,525 | 95.7% | 0.3% |
| Bacteria | 247,570 | 6,990 | 1,340 | 98.9% | 0.1% |
| Random | 247,570 | 12,425 | 2,134 | 97.5% | 0.2% |
| Swiss-Prot$\bowtie$KEGG | 302,068 | 12,696 | 2,184 | 97.7% | 0.2% |

**Table 1.** *Cross-evaluation results by data set size and type (taxonomic or random). The domain eukaria is composed by fungi, plants, vertebrates and invertebrates. See also Figure 1.*

## 3 Results and discussion

We submitted the data sets smaller than 40,000 proteins to two rounds of 10-fold cross evaluation (one round for bigger samples) and we present the average value of subset accuracy, the strictest measure of prediction success, as it requires the predicted set of class labels to be an *exact match* of the true set of labels [9]. The cross evaluation results are presented in Table 1. The total data set ’*SwissProt* $\bowtie$ *KEGG*’ achieves 97.7% ±0.2% subset accuracy (for comparison: 44% ±0.2% subset accuracy with the ZeroR algorithm). The table also presents sets containing only proteins from a particular taxonomic domain such as archaea, bacteria or eukaria, composed by fungi, invertebrates, plants and vertebrates. As visible in Figure 1, the accuracy of prediction generally increases when the data set size increases. Also, unexpectedly, taxonomic data sets do not seem to yield better cross-evaluation accuracy than random sets of the same size (with the exception of the bacteria and archaea sets), and they also cover a reduced set of enzymatic functions. In conclusion, the method described can be applied to *any sequenced protein*, without need for existing annotation, however, it works best for proteins having InterPro signatures. To give an indication, 87% of KEGG proteins and 77% of UniProt (76% of TrEMBL and 95% of Swiss-Prot) have at least one signature, rising to 85% of the 11 million proteins in UniProt, if the so-called "not integrated" InterPro signatures were to be included.

*LDF is funded by BBSRC Grant BBF5290381 of Centre for Systems Biology Edinburgh and the University of Newcastle.*

## References

1. Katja Astikainen, Liisa Holm, Esa Pitkänen, Sandor Szedmak, and Juho Rousu. Towards structured output prediction of enzyme function. *BMC Proc*, 2 Suppl 4:S2, 2008.
2. William A Baumgartner, K. Bretonnel Cohen, Lynne M Fox, George Acquaah-Mensah, and Lawrence Hunter. Manual curation is not sufficient for annotation of genomic databases. *Bioinformatics*, 23(13):i41–i48, Jul 2007.
3. Karsten M Borgwardt, Cheng Soon Ong, Stefan Schönauer, S. V N Vishwanathan, Alex J Smola, and Hans-Peter Kriegel. Protein function prediction via graph kernels. *Bioinformatics*, 21 Suppl 1:i47–i56, Jun 2005.
4. C.Z. Cai, L.Y. Han, Z.L. Ji, and Y.Z. Chen. Enzyme family classification by support vector machines. *Proteins: Structure, Function, and Bioinformatics*, 55:66–76, 2004.
5. Jacob Köhler, Jan Baumbach, Jan Taubert, Michael Specht, Andre Skusa, Alexander Rüegg, Chris Rawlings, Paul Verrier, and Stephan Philippi. Graph-based analysis and visualization of experimental results with ondex. *Bioinformatics*, 22(11):1383–1390, Jun 2006.
6. Atem Lysenko, Matthew Morritt Hindle, Jan Taubert, Mansoor Saqi, and Christopher John Rawlings. Data integration for plant genomics–exemplars from the integration of arabidopsis thaliana databases. *Brief Bioinform*, 10(6):676–693, Nov 2009.
7. Igor V Tetko, Igor V Rodchenkov, Mathias C Walter, Thomas Rattei, and Hans-Werner Mewes. Beyond the 'best' match: machine learning annotation of protein sequences by integration of different sources of information. *Bioinformatics*, 24(5):621–628, Mar 2008.
8. G. Tsoumakas, I. Katakis, and I. Vlahavas. *Mining Multi-label Data. In: Data Mining and Knowledge Discovery Handbook.* Springer, 2010.
9. Grigorios Tsoumakas and Ioannis Vlahavas. Random k -labelsets: An ensemble method for multilabel classification, 2007.
10. Ian H Witten and Eibe Frank. *Data Mining - Practical machine learning tools and techniques with Java implementations.* Morgan Kaufmann, San Francisco, 2005.
11. Min-Ling Zhang and Zhi-Hua Zhou. Ml-knn: A lazy learning approach to multi-label learning. *Pattern Recognition*, 40(7):2038–2048, July 2007.

# Reconstructing Developmental Gene Networks using Heterogeneous Dynamic Bayesian Networks with Information Sharing

Frank Dondelinger[1,3], Sophie Lebre[2] and Dirk Husmeier[1]

[1] Biomathematics and Statistics Scotland, Edinburgh, U.K.
{frankd,dirk}@bioss.ac.uk
[2] University of Strasbourg - LSIIT - UMR 7005, FRANCE slebre@unistra.fr
[3] School of Informatics, University of Edinburgh

Background: Dynamic Bayesian networks (DBNs) are frequently applied to the problem of inferring gene regulatory networks from transcriptomic profiles. However, classical DBNs cannot deal with the heterogeneity that arises if we investigate the changing gene interactions during morphogenesis and embryogenesis. In particular, we are interested in the problem of inferring the gene regulation networks for the developmental stages of Drosophila melanogaster from a muscle development gene expression time series.

Aims: We have developed a Bayesian method for inferring heterogeneous DBNs that improves on the shortcomings of previous approaches. This method allows us to infer morphogenic transitions and changes in gene regulation, e.g. during muscle development in Drosophila melanogaster, using time series gene expression data.

Methods: We extend the method of Lebre (2007) by introducing information sharing among time series segments. Our method (1) avoids the need for data discretisation, (2) increases the flexibility over a time-invariant network structure, (3) avoids over-flexibility and overfitting by introducing a regularisation scheme and (4) allows all hyperparameters to be inferred from the data via a consistent Bayesian inference scheme.

Results: We evaluate our method on synthetic transcriptional profiles simulated in silico from a known gold-standard regulatory network, and show that a significant improvement over the unconstrained method without information sharing can be achieved. We apply our method to the problem of inferring the gene regulation networks for the embryo, larva, pupa and adult stages of Drosophila melanogaster from a muscle development gene expression time series, inferring both the temporal change points and the network structure. We note that we get better agreement with the known morphogenic transitions than alternative published methods. Furthermore, the changes we have detected in the gene regulatory interactions are consistent with independent findings reported in the literature.

Conclusions: We have shown that information sharing improves the reconstruction of regulatory networks from nonstationary gene expression time series, and that we can retrieve meaningful transitions and regulatory gene interactions. Consequently, our method represents a useful tool for detecting change points

and providing improved insight into time-dependent gene regulatory processes, e.g. during morphogenesis.

# On the assessment of variability factors in computational network inference

Marco Grimaldi

Fondazione Bruno Kessler
grimaldi@fbk.eu

**Abstract.** We analyze the impact that methodological and experimental parameters have on the performance of the Aracne and Keller algorithm for network inference. Focusing on *scale free* networks, we vary the size of the network, the amount of data at hand and the data normalization schema applied. In order to provide objective evaluation of the methods, we focus on synthetic data and we employ the MCC measure for unbiased scoring of the network reconstruction performance. Our evaluation indicates the data normalization method applied greatly influences the performance of the network inference algorithms tested.

## 1 Introduction

This work focuses on the analysis of two different network reconstruction algorithms quantitatively evaluating the impact that methodological and experimental parameters have on the inferred network. To provide objective evaluation [4] of the algorithms here tested, we focus on synthetic networks: taking into account *scale-free* graphs [1], we adopt a gene network simulator that has been recently proposed for the assessment of reverse engineering algorithms [3].

The network inference methods tested are: the Aracne algorithm [4] and the Keller algorithm [6]. Aracne is a general method based on an information theoretic approach able to address a wide range of network deconvolution problems (from transcriptional to metabolic networks, e.g.: [2, 4]). The Keller algorithm is a kernel-reweighted logistic regression method that extends the $l_1$-regularized logistic regression and has been applied to reverse engineer genome-wide interactions taking place during the life cycle of *Drosophila melanogaster*. It is our plan to apply these algorithms to time series of genomics and environmental data.

To ease the explanation of the algorithms, without any loss of generality, we will refer to the nodes of the network also as genes and to their output levels as expression data.

### 1.1 Synthetic Networks and Data Generation

We benchmark the two algorithms using synthetic data obtained starting from an adjacency matrix describing the network topology. The adjacency matrix is generated by a simple stochastic algorithm modeling a scale-free graph according to the Barabasi-Albert model [1]. We are interested in estimating the structures of interaction between nodes/genes, rather than the detailed strength or the direction of these interactions. Thus, we consider only discrete and symmetric adjacency matrices, representing with a value of 1 the presence of a link between two nodes. A value equal to 0 indicates no interaction.

Given an input adjacency matrix, the network simulator uses fuzzy logic to represent interactions among the regulators of each gene and adopts differential equations to generate continuous data. As in [4], we obtain synthetic expression values of gene $n$ ($n = 1, .., N$) at time $m$ ($m = 1, .., M$) by simulating its dynamics until the expression value reaches its steady state. Each simulation is randomly initialized by the simulator, thus each of the $M$ runs stabilizes around a different value.

### 1.2 Data Normalization

Although many of the real-world issues [7] in data preprocessing and normalization do not apply here as we deal with synthetic data, we are interested in verifying how some of the most common (and possibly simple) steps taken to normalize the data impact the accuracy of the network inference algorithms considered.

*Discretization:* taking into account microarray measurements and the various sources of noise that can be introduced during data acquisition, it is often preferred to consider only the qualitative level of gene expression rather than its actual value [6]: gene expression is modeled as either being up-regulated (+1) or down-regulated (−1) by comparing the given value to a threshold. In this work we calculate the discrete value of the expression for gene $n$ at each of the $M$ steps as the sign of the difference of its expression values at step $m$ and step $m - 1$.

*Rescaling:* when a scaling method is applied to the data, it is often assumed that different sets of intensities differ by a constant global factor [7]. In this work we test two different recaling methods:

- linear rescaling: each gene expression vector is rescaled linearly between $[-1, 1]$;
- statistical normalization: each gene expression vector is rescaled such that its mean value is equal to 0 and the standard deviation equal to 1.

### 1.3 Network Inference Algorithms

*Aracne:* this algorithm is a general method able to address a wide range of network deconvolution problems (from transcriptional to metabolic networks). The adjacency matrix returned by the algorithm is made symmetric and discretized with values in $\{0, 1\}$ (discretization is obtained by rounding values bigger than $10^{-3}$ to 1).

*Keller:* the Keller algorithm is a kernel-reweighted logistic regression method introduced for the reverse engineering of the dynamic interactions between genes based on their time series of expression values. Although the algorithm has been developed to uncover dynamic rewiring of gene transcription networks (e.g.: dynamic changes in their topology), we consider fixed network topology. The algorithm returns a symmetric and discrete (with values in $\{0, 1\}$) adjacency matrix – discretization is obtained by rounding values bigger than $10^{-3}$ to 1.

## 2 Experimental Evaluation

*Performance Metric:* we adopt the Matthews correlation coefficients [5] as metric: this is a balanced measure that takes into account both true/false positives and true/false negatives. The MCC is in essence a correlation coefficient between the observed and predicted binary classifications: it returns a value between $-1$ and $+1$. A coefficient of $+1$ represents a perfect prediction, 0 an average random prediction and $-1$ an inverse prediction [5].

*Performance Evaluation:* we are interested in verifying the MCC scores of the tested algorithms varying the size of the network, the amount of data available and the method adopted to normalize the data prior to network inference. We vary one parameter at a time and then measure the performance of the systems as the mean of ten randomly initialized runs. For each run, the network topology is randomly generated with the desired $N$ number of genes, the expression levels – the data – are (randomly) generated the required number of times ($M$), the selected normalization method is applied and the MCC values for the applied reverse engineering method recorded. The variability of the measurement is expressed as the standard deviation of the 10 independent runs (Figure 1).

Overall, Figure 1 indicates that the normalization procedure applied to the data has a big impact on the performance of the two network inference algoriothms. When we perform data discretization, the two curves stabilize around a MCC value of 0.2. Considering both linearly rescaled expression values and statically normalized values, both the algorithms show MCC curves that stabilize around MCC $= 0.4$. On the other hand, increasing the amount of data available has limited impact to the performance of the two algorithms

## 3 Conclusions

The evaluation indicates that the performance of the different inference algorithms greatly depend on the data normalization method applied. Both Aracne and Keller are negatively influenced by the discretization procedures: the relevant accuracy curves in Figure 1 stabilize around a MCC value of 0.2. In comparison, the accuracy curves relative to statistic normalization and linear rescaling saturate at a MCC value of 0.4. On the other hand, the amount of data ad hand has little impact on the MCC scores: the curves stabilize for ratio of time-steps ($M$) to network size values above 20%

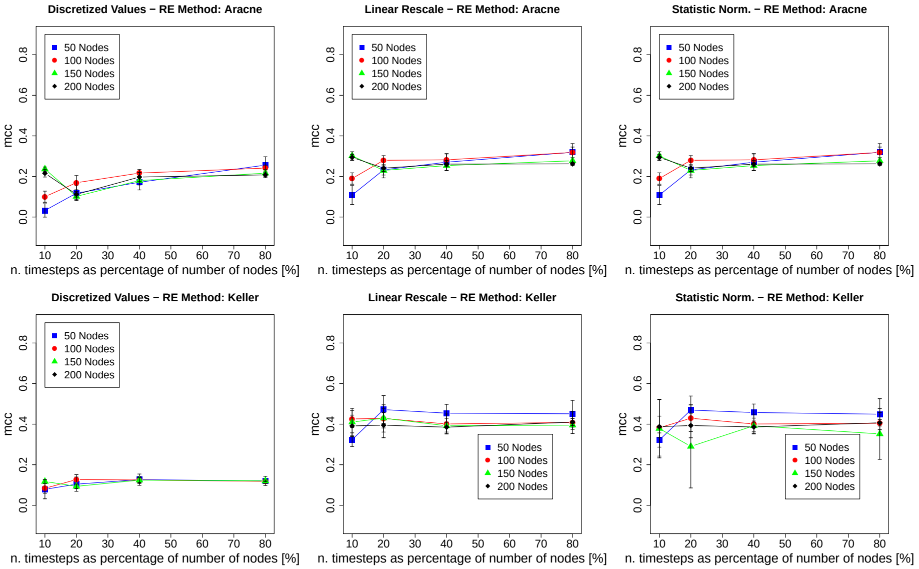


**Fig. 1.** MCC scores of the Aracne and Keller algorithms varying the size of the network, the amount of data and the normalization method.

*Acknowledgments* : this work has is funded by the Autonomous Province of Trento, project ENVIROCHANGE. The author is thankful to Cesare Furlanello and Giuseppe Jurman from FBK for the helpful comments and advices.

## References

1. A.L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999.
2. T. Cakr, M.M.W.B. Hendriks, J.A. Westerhuis, and A.K. Smilde. Metabolic network discovery through reverse engineering of metabolome data. *Metabolomics*, 5(3):318 – 329, 2009.
3. B. Di Camillo, G. Toffolo, and C. Cobelli. A gene network simulator to assess reverse engineering algorithms. *Annals of the New York Academy of Sciences*, 1158, 2009.
4. A.A. Margolin, I. Nemenman, K. Basso, C. Wiggins, G. Stolovitzky, D.R. Favera, and A. Califano. ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics*, 7(Suppl. 1)(7), 2006.
5. B. W. Matthews. Comparison of the predicted and observed secondary structure of t4 phage lysozyme. *Biochimica et biophysica acta*, 405(2):442–451, October 1975.
6. L. Song, M. Kolar, and E. P. Xing. Keller: estimating time-varying interactions between genes. *Bioinformatics*, 25(12), 2009.
7. C. Steinhoff and M. Vingron. Normalization and quantification of differential expression in gene expression microarrays. *Brief Bioinform*, 7(2):166–177, 2006.

# Explaining Kernel Based Predictions in Drug Design

Katja Hansen, David Baehrens and Klaus-Robert Müller

T.U. Berlin, Germany
khansen@cs.tu-berlin.de,david@baehrens.net,
klaus-robert.mueller@tu-berlin.de

Many modern QSAR/QSPR approaches result in black box models. While delivering excellent prediction performance, most methods will provide no answer as to why the model predicted a particular label for a certain molecule. However, sometimes human expert's intuition and model's predictions do not match, because the models training data includes relevant information unknown to the expert or vice versa.

We introduce a new method that explains predictions of kernel based models (such as Support Vector Machines or Gaussian Processes) by the means of calculating and visualizing the most relevant molecules from the training set of the model. This allows practitioners to understand how each prediction comes about. If a prediction conflicts with an expert's intuition, he can examine easily whether the grounds for the model's prediction are solid.

The new method was evaluated by a group of 40 people, including experts in pharmaceutics and chemistry. The participants were asked to evaluate toxicity predictions made by different models both with and without the explanations provided by our new method. Considering the explanations led to a statistically significant improvement of the expert as well as layman users' ability to identify reliable predictions.

# From gene expression to predicted gene regulation in the defence response of Arabidopsis to infection by *Botrytis cinerea*

Steven Kiddle[1,2], Richard Hickman[1,2], Katherine Denby[1,2], and Sach Mukherjee[3,4]

[1] Warwick Systems Biology Centre, Warwick University, CV4 7AL, UK
[2] Warwick HRI, Warwick University, Wellesbourne, CV35 9EF, UK
[3] Department of Statistics, Warwick University, CV4 7AL, UK
[4] Centre for Complexity Science, Warwick University, CV4 7AL, UK

Here we study the response of the *Arabidopsis thaliana* transcriptome to infection by *Botrytis cinerea* in an attempt to elucidate gene regulatory networks implicated in the plant defence response. Many of the genes whose mutants show altered susceptibility to *B. cinerea* in the literature are transcription factors (TFs), suggesting the importance of gene regulation in the defence response. Examples of TF mutants known to cause altered susceptibility include; *bos1, zfar1, wrky70, wrky33, ora59, camta3* and *ataf1* [1, 2, 4–7]. In the majority of cases it is not known how these TFs affect susceptibility.

To understand the resistance mechanisms that these TFs regulate, and to find novel regulators, we use a combination of machine learning and experimental techniques to predict and validate gene regulation. This Systems Biology approach to understanding the plant defence response has already revealed novel regulators and predicted networks of regulation that are currently being tested experimentally.

Temporal clustering and network inference are used to interpret time course data of Arabidopsis gene expression during infection by B. cinerea (Denby et al., manuscript in preparation). We employ a temporal clustering approach, TCAP, that accounts for key temporal features to find novel regulators of the plant defence response [3]. For some of these regulators we show experimental validation of their role in the defence response. Network inference, rooted in a graphical models formulation, is used to elucidate regulatory network topology of differentially expressed genes during *B. cinerea* infection. However, robust inference is challenging in the present setting. We therefore aid network inference by exploiting sequence information. Specifically, DNA sequences of the promoters of co-expressed genes are analysed for known TF binding sites, and specific TF families are found to be at least partly responsible for the observed co-expression. Network inference is then used to predict the single TFs from these families that bind to these sequences. Two regulators predicted to regulate groups of co-expressed genes turn out to have be previously found experimentally to be important to the defence response.

## References

1. S Abuqamar, X Chen, R Dhawan, B Bluhm, J Salmeron, S Lam, R A Dietrich, and T Mengiste. Expression profiling and mutant analysis reveals complex regulatory networks involved in Arabidopsis response to Botrytis infection. *Plant J*, 48(1):28–44, 2006.
2. Y Galon, R Nave, J M Boyce, D Nachmias, M R Knight, and H Fromm. Calmodulin-binding transcription activator (CAMTA) 3 mediates biotic defense responses in Arabidopsis. *FEBS Lett*, 582(6):943–948, 2008.
3. S J Kiddle, O P F Windram, S McHattie, A Mead, J Beynon, V Buchanan-Wollaston, K J Denby, and S Mukherjee. Temporal clustering by affinity propagation reveals transcriptional modules in Arabidopsis thaliana. *Bioinformatics*, 26(3):355–362, 2010.
4. T Mengiste, X Chen, J Salmeron, and R Dietrich. The BOTRYTIS SUSCEPTIBLE1 Gene Encodes an R2R3MYB Transcription Factor Protein That Is Required for Biotic and Abiotic Stress Responses in Arabidopsis. *Plant Cell*, 15:2551–2565, 2003.
5. M Pre, M Atallah, A Champion, and M De Vos. The AP2/ERF domain transcription factor ORA59 integrates jasmonic acid and ethylene signals in plant defense. *Plant Physiol*, 147:1347–1357, 2008.
6. X Wang, B M V S Basnayake, H Zhang, G Li, W Li, N Virk, T, and F Song. The Arabidopsis ATAF1, a NAC Transcription Factor, Is a Negative Regulator of Defense Responses Against Necrotrophic Fungal and Bacterial Pathogens. *Mol Plant Microbe In*, 22(10):1227–1236, 2009.
7. Z Zheng, S A Qamar, Z Chen, and T Mengiste. Arabidopsis WRKY33 transcription factor is required for resistance to necrotrophic fungal pathogens. *Plant J*, 48(4):592–605, 2006.

# Predictive clustering relates gene annotations to phenotype properties extracted from images

Dragi Kocev[1], Bernard Ženko[1], Petra Paul[2], Coenraad Kuijl[2],
Jacques Neefjes[2], and Sašo Džeroski[1]

[1] Department of Knowledge Technologies, Jozef Stefan Institute,
Jamova 39, 1000 Ljubljana, Slovenia
{Dragi.Kocev,Bernard.Zenko,Saso.Dzeroski}@ijs.si
[2] Division of Cell Biology and Centre for Biomedical Genetics,
the Netherlands Cancer Institute
Plesmanlaan 121, 1066 CX Amsterdam, Netherlands
{p.paul,c.kuijl,j.neefjes}@nki.nl

We address the task of grouping genes resulting in highly similar phenotypes upon siRNA mediated downregulation. The phenotypes are described by features extracted from images of the corresponding cellular assays. Both freely available general-purpose software, such as CellProfiler [4], and custom-made proprietary software can be used for this purpose. The features capture properties (such as intensity or texture) of the cells or their parts (nuclei, citoplasm, Golgi apparatus ...) in the images.

Clustering [6] produces partitions of the objects of interest (genes) into groups that are similar in a given feature space. In the context of the application of interest, this is a set of features extracted from the images of cellular assays. Besides finding clusters, e.g., groups of genes, we also aim to find descriptions/explanations for the clusters. The groups are explained in terms of a set of descriptors from a separate space, i.e., annotations of genes in terms of, e.g., the Gene Ontology [2] or the KEGG Pathway Database [5].

The typical approach to the problem at hand is to first cluster the phenotypes and elucidate the characteristics of the obtained clusters later on. Instead, we perform so-called constrained clustering, which yields both the clusters and their symbolic descriptions all in one step. The constrained clustering can be performed by using predictive clustering trees (PCTs) [3, 8, 9, 7], predictive clustering rules [10, 11] or ensembles of predicitve clustering rules [1]: These exemplify the paradigm of predictive clustering, which combines clustering and prediction.

In the presentation, we will describe the methods of building predictive clustering trees and ensembles of predictive clustering rules. We will also describe its application to the analysis of image data resulting from siRNA screens. These approaches have been used to analyze image data from a siRNA screen designed to study MHC Class II antigen presentation.

An example predictive clustering tree obtained in this domain is given in Figure 1. The tree has been produced by clustering phenotypes as described by 13 image features (such as intensity, texture, etc.). The cosine distance/similarity metric has been used for the clustering.

The internal nodes of the tree contain GO terms with which the genes are annotated. The leaves of the tree correspond to the clusters/groups of genes. For example, one such group (C1) includes the genes involved in the biological processes of 'defense response' (GO0006952) and 'regulation of metabolic processes' (GO001922).

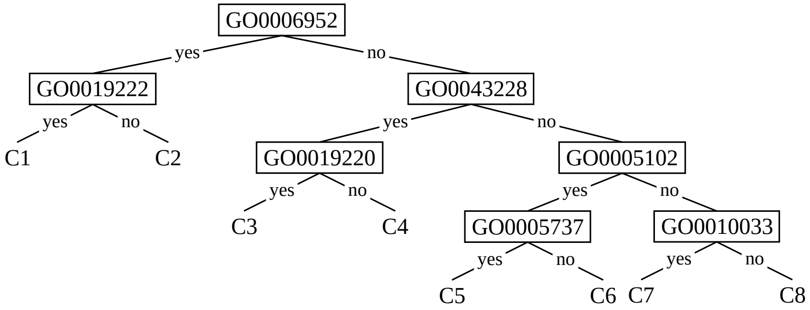


**Fig. 1.** A predictive clustering tree obtained from a siRNA screen for studying the MHC Class II antigen presentation. The internal nodes of the tree contain GO terms with which the genes are annotated. Leaves of the tree correspond to clusters of genes.

An example predictive clustering rule obtained in this domain is given in Table 2. The tree has been produced by clustering phenotypes as described by 6 image features. Feature selection was performed on the GO terms and only the selected subset of features was used to explain the clusters. The Euclidean distance measure was used. The cluster contains genes which are involved in 'regulation' (GO0065007) and in particular 'cellular nucleobase, nucleoside, nucleotide and nucleic acid metabolic process' (GO0006139).

**Table 1.** A predictive clustering rule obtained from a siRNA screen for studying the MHC Class II antigen presentation. The conditions in the antecedent describe the genes in the group in terms of their GO annotations.

```
IF GO0006139 = 1 AND
   GO0065007 = 1
THEN ClusterD1
```

In sum, we have applied predictive clustering to data from a siRNA screen designed to study MHC Class II antigen presentation. As a result of the predictive clustering process, we obtain clearly defined/described groups of genes, which yield similar phenotypes upon siRNA mediated downregulation. Groups of this kind can be used to identify pathways regulating the processes of interest (such as MHC Class II antigen presentation).

## References

1. Timo Aho, Bernard Ženko, and Sašo Džeroski. Rule ensembles for multi-target regression. In Wei Wang, Hillol Kargupta, et al., editors, *Proceedings of the Ninth IEEE International Conference on Data Mining (ICDM 2009)*, pages 21–30, Los Alamitos, CA, 2009. IEEE Computer Society.
2. Michael Ashburner et al. Gene ontology: tool for the unification of biology. the gene ontology consortium. *Nature Genetics*, 25:25–29, 2000.
3. Hendrik Blockeel, Luc De Raedt, and Jan Ramon. Top-down induction of clustering trees. In J. Shavlik, editor, *Proceedings of the 15th International Conference on Machine Learning*, pages 55–63. Morgan Kaufmann, 1998.
4. Anne Carpenter, Thouis Jones, Michael Lamprecht, Colin Clarke, In Kang, Ola Friman, David Guertin, Joo Chang, Robert Lindquist, Jason Moffat, Polina Golland, and David Sabatini. Cellprofiler: image analysis software for identifying and quantifying cell phenotypes. *Genome Biology*, 7(10):R100+, 2006.
5. Minoru Kanehisa, Susumu Goto, Miho Furumichi, Mao Tanabe, and Mika Hirakawa. Kegg for representation and analysis of molecular networks involving diseases and drugs. *Nucleic Acids Research*, 38(DB):D355–360, 2010.
6. Leonard Kaufman and Peter J. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis.* Wiley-Interscience, 1990.
7. Ivica Slavkov, Valentin Gjorgjioski, Jan Struyf, and Sašo Džeroski. Finding explained groups of time-course gene expression profiles with predictive clustering trees. *Molecular BioSystems*, 6(4):729–740, 2010.
8. Jan Struyf and Sašo Džeroski. Constraint based induction of multi-objective regression trees. In *Proc. of the 4th International Workshop on Knowledge Discovery in Inductive Databases KDID - LNCS 3933*, pages 222–233. Springer, 2006.
9. Celine Vens, Jan Struyf, Leander Schietgat, Sašo Džeroski, and Hendrik Blockeel. Decision trees for hierarchical multi-label classification. *Machine Learning*, 73(2):185–214, 2008.
10. Bernard Ženko and Sašo Džeroski. Learning classification rules for multiple target attributes. In Takashi Washio, Einoshin Suzuki, et al., editors, *Proceedings of the 12th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2008)*, LNCS, pages 454–465, Berlin, Germany, 2008. Springer.
11. Bernard Ženko, Jan Struyf and Sašo Džeroski. Analyzing time series gene expression data with predictive clustering rules. In Sašo Džeroski, Pierre Geurts and Juho Rousu, editors. *Machine learning in systems biology: proceedings of the Third International Workshop,* September 5-6, 2009, Ljubljana, Slovenia, pages 177-178. (Julkaisusarja - Helsingin yliopisto. Tietojenksittelytieteen laitos, report B-2009-1), 2009. University of Helsinki, Department of Computer Science.

# Automated analysis of biological oscillator models

Tomasz Konopka

Universite Libre de Bruxelles
tkonopka@ulb.ac.be

The purpose of the model selection analysis, also called reverse-engineering, is to identify equation structures and parameter values that best describe some given measured signals. Using the techniques based on mode decomposition, this task can be accomplished by solving systems of simultaneous polynomial equations (as opposed to minimizing nonlinear functions like in other approaches, e.g. least-squares optimization) and therefore produces reliable results in predictable time. Apart from speed, matching of data to model structures in an automated fashion also has a number of advantages over studying individual models manually. An important one is that, given a signal, it is not only possible to determine whether a proposed model structure is consistent with the data, but also to scan hundreds of alternative models according to similar criteria.

When applied to synthetic data sets, both those generated using ordinary differential equations with added noise and those obtained using stochastic methods, the presented workflow is capable of selecting correct model structures out of a space of hundreds of possibilities. It also outputs some model structures that fit the data quite well but do not have a correspondence in the synthetic system, providing a reminder of the limits of direct model fitting as a means of network inference.

The automated analysis is also applied to a micro-array dataset on gene expression oscillations in mouse liver cells under the circadian cycle [Hughes et al 2009]. There, because the underlying mechanisms are not known, the model selection procedure can be used to test theoretical models against a number of alternatives or to determine genes that may be active in a particular role (as defined by a term in a differential equation) in the expression regulation network.

# Shrinking Covariance Matrices using Biological Background Knowledge

Ondřej Kuželka and Filip Železný

Czech Technical University, Prague, Czech Republic

**Abstract.** We propose a novel method for covariance matrix estimation based on shrinkage with a target inferred from biological background knowledge using methods of inductive logic programming. We show that our novel method improves on the state of the art when sample sets are small and some background knowledge expressed in a subset of first-order logic is available. As we show in the experiments with genetic data, this background knowledge can be even only indirectly relevant to the modeling problem at hand.

## 1 Introduction

An important problem in modelling of gene expression data is estimation of large covariance matrices. Only quite recently it has been realized that the vast amount of structured knowledge available in databases like KEGG [5] could be used to improve the estimation of covariance matrices. So far, all the approaches following this idea used biological knowledge only in a restricted way. For example in [3], shrinkage targets for covariance matrices have been constructed with non-diagonal entries being non-zero for genes from the same gene groups. We introduce a novel method that exploits structured knowledge in a non-trivial way and improves on a state-of-the-art covariance estimation method.

## 2 SGLNs: Simple Gaussian Logic Networks

We will be working with existentially quantified conjunctions of first-order logical atoms (*conjunctions*), which we will also treat as sets. We say that a conjunction $C$ *$\theta$-subsumes* a conjunction $D$ (denoted by $C \preceq_\theta D$) iff there is a substitution $\theta$ such that $C\theta \subseteq D$. For example, if $C = a(B,C)$ and $D = a(x,y), b(y,z)$ then $C \preceq_\theta D$ because $C\theta \subseteq D$ for $\theta = \{A/x, B/y\}$. Next, we describe a framework termed *simple gaussian logic networks* (SGLNs) which borrows ideas from Markov logic networks and Bayesian logic programs [4].

**Definition 1 (Simple Gaussian Logic Networks).** *A Simple Gaussian Logic Network (SGLN) is a triple $(G, R, N)$ where $G = (g_i)$ (gaussian atoms) is a list of ground first-order atoms, $R$ (rules) is a set of conjunctions and $N$ (network) is a ground conjunction. A normal distribution $N(\mu, \Sigma)$ is said to comply with a SGLN $S = (G, R, N)$ if for $P = (p_{ij}) = \Sigma^{-1}$ it holds $p_{ij} = 0$ whenever there is no rule $r \in R$ and substitution $\theta$ such that $\{g_i, g_j\} \subseteq r\theta \subseteq N$.*

It is well-known [7] that a multivariate normal distribution with covariance matrix $\Sigma$ and precision matrix $P = \Sigma^{-1} = (p_{ij})$ can be viewed as a Markov network in which there are edges between any two variables (nodes) $v_i$ and $v_j$ for which $p_{ij} \neq 0$. Thus, SGLNs define an independence structure over the variables corresponding to the gaussian atoms $g_i$.

*Example 1.* Let us have a SGLN $S = (G, R, N)$ where $G = (g(a), g(b), g(c))$, $R = \{g(X), edge(X, Y), edge(Y, Z), g(Z)\}$ and $N = g(a), g(b), g(c), edge(a, b)$, $edge(b, c)$. Then any normal distribution with covariance matrix $\Sigma$ (below) complies with $S$.

$$\Sigma^{-1} = P = \begin{bmatrix} x & 0 & y \\ 0 & z & 0 \\ y & 0 & w \end{bmatrix}$$

We have not explained yet how to obtain a covariance matrix complying to a given SGLN $S$ and maximizing likelihood on a set of training examples $E$. The problem of estimating a covariance matrix with a given pattern of zeros in its inverse is known as *covariance selection* [1]. Given an ordinary covariance matrix estimated from data, one can find the maximum-likelihood estimate with a prescribed zero-pattern by means of convex optimization. Very often one has too few data samples compared to the number of variables. In such a case, it is impossible to estimate the covariance matrix reliably as $\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})(x_i - \overline{x})^T$. Instead, we have to apply a more advanced method, for example *shrinkage-based estimation* [6]. The basic idea of shrinkage is to obtain convex combinations of high-dimensional and lower dimensional models. The covariance selection method combined with the shrinkage based estimation of unconstrained covariance matrices gives us an effective tool for learning with SGLNs. First, we obtain an estimate of covariance matrix $\hat{\Sigma}$ using the shrinkage-based approach. Next, we use $\hat{\Sigma}$ as input together with the zero-pattern given by a given SGLN to the covariance selection procedure which gives us the estimate of the covariance matrix complying with the SGLN.

## 3 Estimating Covariance Matrices using SGLNs

In this section we briefly describe a simple method that uses SGLNs to improve covariance matrix estimation (Algorithm 1). It turns out that the covariance matrix obtained from an appropriate SGLN can be a very good shrinkage target. The rationale behind the SGLN-rule-learning part of the method is as follows. We assume that there are some rules which capture the dependency structure of the estimated distribution. First, we create a set of positive examples from unions of $d$-neighbourhoods[1] of *most correlated*[2] pairs of gaussian literals and a set of negative examples from the *least correlated* ones. We can expect that the rules which $\theta$-subsume many positive examples and few negative examples would be good rules of the SGLN.

[1] the neighbourhoods of depth $d$ in the (hyper)-graph theoretical sense

[2] Here, the word *correlation* refers to partial correlations, i.e. $p_{ij}/\sqrt{p_{ii}p_{jj}}$ where $p_{ij}$ are entries of the inverse of the covariance matrix.

**Algorithm 1** CovEstimate:

```
1: Input: Samples $S$, Set of gaussian atoms $G$, Conjunction $N$;
2: $\Sigma_0 \leftarrow$ Estimate covariance matrix from $S$ using [6]
3: $P = (p_{ij}) \leftarrow \Sigma_0^{-1}$ /* $P$ is the so-called precision matrix */
4: $PosExs \leftarrow \max\{k, \lfloor \frac{|V|}{2} \rfloor\}$ pairs $(i,j), i < j$ with highest $|p_{ij}/\sqrt{p_{ii}p_{jj}}|$
5: $NegExs \leftarrow \max\{k, \lfloor \frac{|V|}{2} \rfloor\}$ pairs $(i,j), i < j$ with lowest $|p_{ij}/\sqrt{p_{ii}p_{jj}}|$
6: $PosExs^*, NegExs^* \leftarrow$ Convert $PosExs$ and $NegExs$ to first-order clauses /*see the main
   text*/
7: $R \leftarrow$Construct a set of good rules using an ILP algorithm /*see main text*/
8: $\Sigma_1 \leftarrow$ Obtain an estimate of covariance matrix from $\Sigma_0$ complying with $(G, R, N)$
9: return $t \cdot \Sigma_0 + (1-t) \cdot \Sigma_1$ /* with $t$ selected using internal cross-validation */
```

*Example 2.* Let us have the network $N$ from Example 1 and a set of samples $M$. Let us suppose that we obtained the following covariance matrix by the shrinkage-based estimation method applied on samples from $M$.

$$\hat{\Sigma}^{-1} = \begin{bmatrix} 1 & 0 & -0.5 \\ 0 & 0.75 & 0 \\ -0.5 & 0 & 1 \end{bmatrix}$$

Now, let us construct the sets of positive and negative examples according to the recipe described in the preceding paragraphs. We set $d = 1$. Then $E^+ = \{e_1\}$ where $e_1$ corresponds to pair of gaussian literals $g(a), g(c)$ and $e_1 = (\{g(a), edge(a,b)\} \setminus \{g(b), g(c)\}) \cup (\{edge(b,c), g(c)\} \setminus \{g(a), g(b)\}) = g(a), edge(a,b), edge(b,c), g(c)$. Analogically, $E^- = \{e_2\}$ where $e_2 = g(a), g(b)$. It depends on the chosen language bias which rules would be induced. For example, if rules were restricted to connected clauses, the correct rule $g(X), edge(X,Y), edge(Y,Z), g(Z)$ from Example 1 would be one of them.

## 4 Experimental Results and Conclusions

In this section we show how SLGNs can be applied to covariance estimation of gene expression data. We used datasets from GEO database [2], namely GDS1209 and GDS1220. For each dataset, we generated 65 smaller datasets each corresponding to one pathway from KEGG database [5]. For each of these, we created a network $N$ consisting of relations contained in KEGG, e.g. relation of *activation* or *phosphorylation* among proteins etc. Then we compared the baseline shrinkage-based method (Shr.) [6] with our novel method (SGLN). SGLN clearly outperformed the existing method (cf. Table 1). Nevertheless, one could still argue that we could obtain the same or even better improvement if we replaced the matrix on line 8 of Algorithm 1 by a matrix obtained with a different zero pattern which would have the same number $K$ of non-zero elements but these non-zero elements would correspond to $K$ most correlated pairs of variables. In other words, one could ask whether the use of background knowledge brings us any benefits. Therefore we performed experiments also with this suggested method (Top-K), but again SLGN outperformed it. Using cross-validation, we measured both likelihood on unseen data and RMSE of estimates of values with

**Table 1.** Results on the gene expression datasets. **Top:** Average ratios of likelihoods on test data obtained by the different methods (smaller is better). **Bottom:** Average ratios of RMSEs on test data obtained by the different methods (smaller is better). Numbers in parentheses correspond to number of wins/losses/ties.

| Dataset | **L - SGLN x Shr.** | **L - SGLN x Top-K** | **L - Top-K x Shr.** |
|---|---|---|---|
| **GDS1209_15** | 0.907 (62/3/0) | 0.979 (39/25/1) | 0.927 (64/1/0) |
| **GDS1209_39** | 0.939 (63/2/0) | 0.980 (53/12/0) | 0.958 (64/1/0) |
| **GDS1220_10** | 0.937 (64/1/0) | 0.963 (55/10/0) | 0.974 (54/5/6) |
| **GDS1220_44** | 0.942 (63/2/0) | 0.975 (59/6/0) | 0.966 (54/2/9) |
| Dataset | **RMSE - SGLN x Shr.** | **RMSE - SGLN x Top-K** | **RMSE Top-K x Shr.** |
| **GDS1209_15** | 0.899 (57/8/0) | 0.968 (44/20/1) | 0.928 (58/7/0) |
| **GDS1209_39** | 0.983 (50/15/0) | 0.996 (37/28/0) | 0.987 (53/12/0) |
| **GDS1220_10** | 0.983 (52/13/0) | 0.991 (49/16/0) | 0.991 (51/8/6) |
| **GDS1220_44** | 0.981 (58/7/0) | 0.994 (43/22/0) | 0.987 (51/5/9) |

a randomly selected half of the variables (genes) set to known values. A one-sided binomial test ($\alpha = 0.05$) on the *number of wins* has shown that SGLN was always significantly better than Shr. and that in all but two cases SGLN was also significantly better than Top-K.

In this paper, we have introduced a novel method that is able to exploit structured background knowledge for estimation of covariance matrices and outperforms an existing state-of-the-art method. An interesting fact is that even though most of the background knowledge was not directly related to gene co-expression as the pathways contain more relations regarding products of the respective genes, it increased accuracy. It would be therefore interesting to interpret some of the learned rules from the biological point of view.

**Acknowledgements:** OK has been supported by the Grant Agency of the Czech Technical University in Prague, grant SGS10/073/OHK3/1T/13. FŽ has been supported by the Czech Grant Agency through project 201/09/1665.

## References

1. J. Dahl, L. Vandenberghe, V. Roychowdhury, Covariance selection for non-chordal graphs via chordal embedding. *Optimization Methods and Software*, 23 (4), 501-520, 2008.
2. R. Edgar, M. Domrachev, and A. Lash. Gene expression omnibus: NCBI gene expression and hybridization array data repository. *Nucleic Acids Res.*, 30 (1), 2002.
3. Feng Tai, Wei, Pan, Incorporating prior knowledge of gene functional groups into regularized discriminant analysis of microarray data, *Bioinformatics*, 2007
4. L. Getoor, B. Taskar, Introduction to Statistical Relational Learning, The MIT Press, 2007.
5. M. Kanehisa, S. Goto, S. Kawashima, Y. Okuno, and M. Hattori. The KEGG resource for deciphering the genome. Nucleic Acids Res, 32, 2004.
6. J. Schäffer, K.Strimmer, A Shrinkage Approach to Large-Scale Covariance Matrix Estimation and Implications for Functional Genomics, *Statistical Applications in Genetics and Molecular Biology*, 4 (1), 2005.
7. T. P. Speed, H. T. Kiiveri, Gaussian Markov Distribution over Finite Graphs, *The Annals of Statistics*, 14 (1), 1986.

# Filling in the gaps of biological network

Viet Anh Nguyen and Pietro Lio'

University of Cambridge, U.K.
vietanh0@gmail.com,pl219@cam.ac.uk

There is currently a tremendous growth in the amount of life science high through-put data which has been paralleled by similar growths of electronic communication, economics and social science data. The large amount of data will represent the start of a golden age for artificial intelligence and in particular bioinformatics machine learning techniques. Examples of high through-put life science data are the large number of completely sequenced genomes, 3D protein structures, DNA chips, and mass spectroscopy data. Large amounts of data are distributed across many sources over the web, with high degree of semantic heterogeneity and different levels of quality. These data must be combined with other data and processed by statistical tools for patterns, similarities, and unusual occurrences to be observed. The results of many experiments can be summarised in a large matrix, in which rows represent repetition of the experiment in different context, and the columns are the output of a single measurement. In particular genetic network data provide means for adjusting cellular metabolism, growth, and development to environmental alterations. The molecular communications generated by genetic networks can be triggered by different nutrients, ions, drugs and other compounds, but also by physical parameters such as temperature, pressure and pH. Biologists represent biochemical and gene networks as state transition diagrams with rates on transition. This approach provides an intuitive and qualitative understanding of the gene/protein interaction networks underlying basic cell functions through a graphical and database-oriented representation of the models. More mathematical approaches focus on modeling the relationships between biological parameters using a connectivity network, where nodes are molecules and edges represent weighted ontological/evolutionary connections. Therefore a genetic network can be represented by an adjacency matrix showing the value for each gene-gene interaction. In a very recent paper, Koukolikova et al. [1] has demonstrated the power of the iterative spectral algorithm proposed by Maslov and Zhang [2] in inferring missing values of protein contact maps and cytokine networks. While the estimation ability of the method depends considerably on the hidden feature dimension M, there is not yet a solid theoretical way to determine its value. We propose the use of Bayesian statistics to automatically determine the most appropriate M value at each iteration of the learning process [3,4]. As the result, the value of M is changed accordingly to the amount of information available at each step, and approaches a fixed value when the predictions start to converge. Conclusions We have developed a fully-automated Bayesian spectral algorithm which proves useful in estimating missing data as well as predicting the nature of noise containing within the investigated system. The evaluation on three different datasets of cytokine networks, enzyme-ligands, and microarray gene expressions have shown significant improvement compared to other general and data-specific methods. We show that the approach is very robust in handling large percentage of missing data both in terms of accurate prediction and quick convergence rate. We also investigate the role of nonlinearities and noise in the matching phase (i.e. for example protein and ligand). In particular, it is shown that nonlinearities appear as noise when linear investigation tools are used. We introduce the use of the distribution of data correlations in combination with our Bayesian spectral approach to make prediction on the noise nature within the investigated systems. A set of different types of noise is investigated.

## Reference

1. Z. Nicola-Koulikova, P. Lio', F. Bagnoli (2007) Inference on Missing Values in Genetic Networks Using High-Throughput Data EVOBIO LNCS Springer Verlag
2. S. Maslov and Y-C. Zhang (2001). Extracting Hidden Information from Knowledge Networks. Physical Review Letters 87, 248701 1-4.
3. V Nguyen, Z Koukolikova, F Bagnoli, P Lio' (2009) Noise and nonlinearities in high-throughput data. J. Statistical Mechanics P01014
4. V.A. Nguyen, Z. Nicola-Koulikova, F. Bagnoli, P. Lio' Bayesian inference on hidden knowledge in high-throughput molecular biology data. Lectures Notes in Artificial Intelligence (PRICAI-08) LNAI volume 5351, "PRICAI 2008: Trends in Artificial Intelligence", pages 829-838.

# Comparing diffusion and weak noise approximations for inference in reaction models

Andreas Ruttor, Florian Stimberg, and Manfred Opper

Computer Science, TU Berlin, Germany

## 1 Inference for reaction systems

The problem of probabilistic inference for stochastic reaction models in systems biology has attracted considerable interest, see e.g. [1]. A well studied problem is the limiting case, where the number of molecules in the system is sufficiently large to allow for a deterministic description of the dynamics by a set of (usually nonlinear) ordinary differential equations.

Inference becomes far more complicated when fluctuations are relevant. The dynamics is modelled by a continuous time Markov jump process, which describes stochastic changes of the number of molecules of a given type due to the reactions in the system. A simplified modelling of the stochastic dynamics is possible, when molecule numbers are large enough to be approximated by continuous random variables. A common computational technique for this limit is the replacement of the discrete jump process by a Markov process with continuous sample paths, i.e. by a diffusion process. Despite this simplification, statistical inference using Markov chain Monte Carlo (MCMC) methods is still computationally demanding [2].

A different approach to probabilistic inference designed for the same range of problems, where molecule numbers are not too small, has been termed weak noise approximation in [3] and was motivated by van Kampen's system size expansion [4]. It is based on the idea that relative fluctuations of molecule numbers may not be large in such cases and could—to lowest order—be well approximated by Gaussian random variables. This method leads to the solution of ordinary differential equations for the moments of the Gaussians, which can be solved in times that are usually much smaller than the ones required for Markov chain Monte Carlo approaches.

Hence, one might ask the question, whether the use of the simpler approach may lead to dramatically different results. To address this question we compare the results of the two methods on the well-known Lotka-Volterra model (for a definition of the reactions and rate constants see [5]). Other models of reaction systems are currently under investigation.

## 2 Diffusion approximation and MCMC

The state of reaction models is described by a vector $\mathbf{x} = (x_1, \ldots, x_M)^\top$, where $x_i$ is the number of molecules of species $i$. The stochastic dynamics is a Markov jump process (MJP) defined by a rate function $f\,(\mathbf{x}'|\mathbf{x})$ which determines the temporal change of transition probabilities via $P\,(\mathbf{x}', t+\Delta t|\mathbf{x}, t) \simeq \delta_{\mathbf{x}',\mathbf{x}} + \Delta t\; f\,(\mathbf{x}'|\mathbf{x})$ for $\Delta t \to 0$.

The diffusion approximation to this process is defined by the stochastic differential equation $d\mathbf{x}(t) = \mathbf{f}(\mathbf{x})dt + \mathbf{D}^{1/2}(\mathbf{x})d\mathbf{w}(t)$, for which the drift vector $\mathbf{f}$ and the diffusion matrix $\mathbf{D}$ agree with the first and second jump moments of the jump process, i.e.

$$\mathbf{f}(\mathbf{x}) = \sum_{\mathbf{x}'\neq\mathbf{x}} f(\mathbf{x}'|\mathbf{x})(\mathbf{x}'-\mathbf{x}) \qquad \mathbf{D}(\mathbf{x}) = \sum_{\mathbf{x}'\neq\mathbf{x}} (\mathbf{x}'-\mathbf{x})f(\mathbf{x}'|\mathbf{x})(\mathbf{x}'-\mathbf{x})^\top. \tag{1}$$

The probability density of time discretised sample paths of the diffusion process conditioned on noisy observations is given by

$$p(\mathbf{x}_{0:T}|D) \approx \frac{p(\mathbf{x}_0)}{Z}\left[\prod_{t=0,\Delta t,\ldots}^{T-\Delta t} \mathcal{N}(\mathbf{x}_{t+\Delta t}; \mathbf{x}_t + \mathbf{f}(\mathbf{x}_t)\Delta t, \mathbf{D}(\mathbf{x}_t))\right]\left[\prod_{i=1}^{n} \mathcal{N}(y_i; \mathbf{x}_{t_i}, \sigma^2)\right],$$

which is obtained from an Euler approximation to the SDE and where $\mathcal{N}(x; m, v)$ is the density at $x$ of a Gaussian with mean $m$ and variance $v$, $D = (y_1, \ldots, y_n)$ are the observations, and $\sigma^2$ is the variance of the observation noise. Samples from this density can be obtained by Metropolis-Hastings steps (see [2] for details).

## 3 Weak noise

The starting point of the weak noise approximation is an exact expression of the conditional marginal density of the state vector $p_t(\mathbf{x}|D) \propto p_t(\mathbf{x}|D_{<t})\, r_t(\mathbf{x})$, which is well known from the theory of hidden Markov models. The first factor $p_t(\mathbf{x}|D_{<t})$ is the conditional distribution of the state based only on the observations $D_{<t} \equiv \{\mathbf{y}_i\}_{t_i<t}$ *before* time $t$. For times between observations this probability fulfils the *forward* Fokker-Planck equation with jump conditions at the observations. And $r_t(\mathbf{x}) \equiv p(D_{\geq t}|\theta, \mathbf{x}_t = \mathbf{x})$ is the likelihood of *future observations* $D_{\geq t} = \{\mathbf{y}_i\}_{t_i\geq t}$ conditioned on the present state $\mathbf{x}(t) = \mathbf{x}$. The likelihood of all data is then $p(D|\theta) = \sum_{\mathbf{x}} p_0(\mathbf{x})r_0(\mathbf{x})$, where $p_0(\mathbf{x})$ is the distribution of the initial state. $r_t$ fulfils the Kolmogorov backward equation

$$\left[\frac{\partial}{\partial t} + \mathbf{f}(\mathbf{x}, t)^\top \nabla + \frac{1}{2}\mathrm{Tr}(\mathbf{D}(\mathbf{x}, t)\nabla^\top \nabla)\right] r_t(\mathbf{x}) = 0\,. \tag{2}$$

The weak noise expansion is based on the assumption that typical state vectors are close to a nonrandom time dependent state $\mathbf{b}(t)$. Therefore one sets $\mathbf{x} = \mathbf{b}(t) + \epsilon\mathbf{u}$ with an expansion parameter $\epsilon$ (which is set later to 1) and also rescales the *noise* $\mathbf{D} \to \epsilon^2\mathbf{D}$. An expansion of the backward equation up to order $\epsilon^2$ yields

$$r_t(\mathbf{x}) \propto \exp\left[-\frac{1}{2}(\mathbf{x}-\mathbf{b})^\top \mathbf{S}^{-1}(\mathbf{x}-\mathbf{b})\right], \tag{3}$$

where the macroscopic state $\mathbf{b}$ and the matrix $\mathbf{S}$ satisfy the differential equations

$$\dot{\mathbf{b}} = \mathbf{f}(\mathbf{b}) \qquad \dot{\mathbf{S}} = \mathbf{AS} + \mathbf{SA}^\top - \mathbf{D}(\mathbf{b}) \tag{4}$$

with $A_{ij}(t) = \partial_{x_j} f_i(\mathbf{x})\big|_{\mathbf{x}=\mathbf{b}(t)}$.

Using a similar expansion for the forward Fokker Planck equation a Gaussian approximation for $p_t(\mathbf{x}|D)$ is obtained, where the mean state vector $\mathbf{m}$ and the covariance matrix $\mathbf{C}$ evolve according to

$$\dot{\mathbf{m}} = \mathbf{g}(\mathbf{m}) \qquad \dot{\mathbf{C}} = \mathbf{HC} + \mathbf{CH}^\top + \mathbf{D}(\mathbf{m}) \tag{5}$$

with $H_{ij}(t) = \partial_{x_j} g_i(\mathbf{x})\big|_{\mathbf{x}=\mathbf{m}(t)}$ and

$$\mathbf{g}(\mathbf{x}, t) \approx \mathbf{f}(\mathbf{x}) - \mathbf{D}(\mathbf{b}(t))\mathbf{S}^{-1}(t)(\mathbf{x} - \mathbf{b}(t))\,. \tag{6}$$

Parameter estimation is based on the total likelihood $p(D|\theta)$ of all observations, which is the result of the backward integration. The expected order of magnitude of the parameters and other prior knowledge can be described in form of a prior distribution $p(\theta)$. Then approximate marginal posteriors are calculated from a Laplace approximation of the posterior density $p(\theta|D) \propto p(D|\theta)\ p(\theta)$. Setting $F(\theta) \equiv -\log\left(p(D|\theta)\ p(\theta)\right)$, Laplace's approximation is given by

$$-\log p(\theta_i|D) \approx F(\theta_i, \theta^*_{\setminus i}) + C + \frac{1}{2}\log\left|\frac{\partial^2 F(\theta_i, \theta_{\setminus i})}{\partial\theta^2}\right|_{\theta=\theta^*}, \tag{7}$$

where $\theta^*$ denotes the most likely parameters, i.e. $\theta^* = \arg\min_\theta F(\theta)$, and $\theta_{\setminus i}$ all parameters *without* $\theta_i$.

## 4 Comparison of Weak Noise and MCMC

Figure 1 compares the results of parameter estimation for MCMC sampling based on the approach of [2] and our weak noise approximation [3, 5]. Both methods have been implemented in Matlab. Obtaining 500,000 samples (50,000 discarded as burn-in, thinning factor 100) from MCMC took roughly 80.5 hours on a Intel Core 2 processor, while the approximate inference algorithm ran only for one hour. It is clearly visible, that it produces results comparable to those obtained by MCMC sampling, although it is vastly faster.

Results of state inference using both algorithms are shown in figure 2. Here the parameters of the model have been fixed to their true values $\alpha = 0.05$, $\beta = 0.01$, $\gamma = 0.05$, and $\delta = 0.01$. Obtaining 500,000 samples took 44.5 hours, while it was possible to calculate the marginal posterior using approximate inference in less than one minute. However, both results are nearly identical. Consequently, using the weak noise approximation enables very fast parameter estimation and state inference without loosing much accuracy.

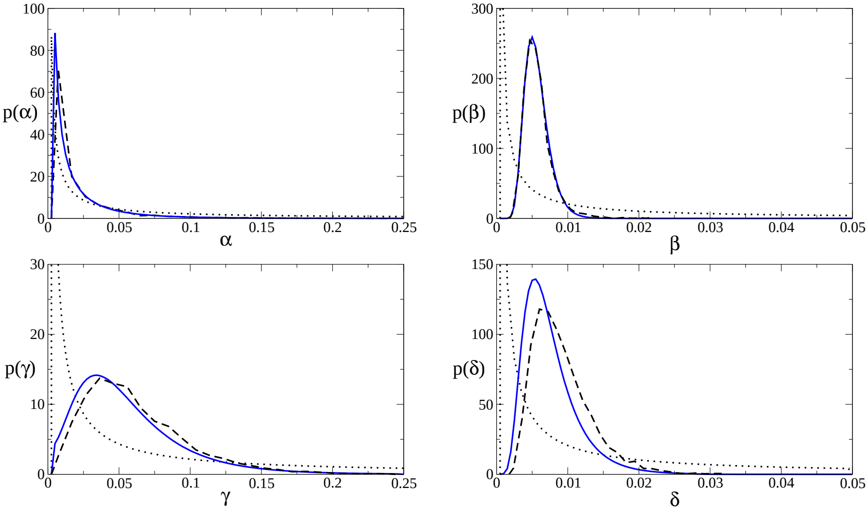


**Fig. 1.** Posterior distributions of the rate constants. Solid blue lines show the results of the approximation, while the histograms obtained from MCMC are plotted as dashed black lines. The prior used in both algorithms is denoted by dotted lines.

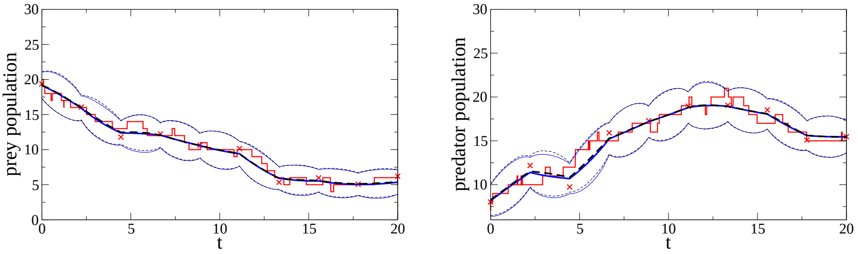


**Fig. 2.** Posterior distributions of the path. Solid blue lines show the results of the approximation, those obtained from MCMC are plotted as dashed black lines. The mean is denoted by thick lines, while thin lines surround the 95%-confidence interval. The true process and the observations taken from it are drawn as a red line and red crosses, respectively.

## References

1. Neil D. Lawrence, Mark Girolami, Magnus Rattray, and Guido Sanguinetti, editors. *Learning and Inference in Computational Systems Biology*. MIT Press, 2010.
2. A. Golightly and D. J. Wilkinson. Markov chain monte carlo algorithms for sde parameter estimation. In Lawrence et al. [1], chapter 9, pages 253–275.
3. Andreas Ruttor, Guido Sanguinetti, and Manfred Opper. Approximate inference for stochastic reaction processes. In Lawrence et al. [1], chapter 10, pages 189–205.
4. N. G. van Kampen. *Stochastic Processes in Physics and Chemistry*. North-Holland, Amsterdam, 1981.
5. Andreas Ruttor and Manfred Opper. Efficient statistical inference for stochastic reaction processes. *Phys. Rev. Lett.*, 103(23):230601, 2009.

# Prediction of catalytic efficiency to discover new enzymatic activities

Chloé Sarnowski, Pablo Carbonell, Mohamed Elati, Jean-Loup Faulon

ISSB, Genopole, Genopole Campus 1, Genavenir 6, 5 rue Henri Desbruères, 91030
EVRY, France

**Abstract.** Characterizing the catalytic efficiency of an enzyme for specific reactions might constitute a useful tool in the prediction of novel enzymatic activities. Here, an approach based on the random forest method is used to predict catalytic efficiency of an enzyme sequence for a particular reaction. Our efficiency predictor achieves a precision of 88% with a recall of 84% and an accuracy of 91%. For any given pair formed by an enzyme sequence and its putative reaction, our tool estimates the catalytic efficiency by applying the random forest method to a selection of sequence and chemical compounds descriptors. Moreover, we show that adding additional molecular signatures-based predictions as descriptors increases the performance of the predictor.

**Keywords:** random forests, classifiers combination, catalytic efficiency, enzyme annotation

## 1 Introduction

Predicting new enzymatic functions for a given enzyme sequence implies the development of both a reaction predictor and a catalytic efficiency predictor. Reaction catalytic parameters used in order to estimate efficiency are usually the Michaelis constant or $K_m$ which reflects the affinity of a substrate for an enzyme and the catalytic constant $k_{cat}$. The specificity constant or performance constant $(k_{cat}/K_m)$ is often used as a measure of catalytic efficiency to compare several substrates or reactions catalysed by an enzyme [5] . Currently it has been not reported a global predictor of catalytic efficiency for a protein sequence concerning a particular reaction. In this study, we present a method for predicting catalytic efficiency using sequence and chemical compounds descriptors. This method is based on the random forest algorithm [1]. It has been demonstrated that the classification method based on random forests achieves good results with unbalanced data [4]. We employed a classifier combination approach which improved predictions made from unbalanced data. Performance of our method leads us to think that it is possible to build a reliable predictor of catalytic efficiency of a sequence for a particular reaction.

# 2 Methods

## 2.1 Dataset from the ECE database

An Enzymatic Catalytic Efficiency (ECE) database was created from the databases KEGG (http://www.genome.jp/kegg) and BRENDA (http://www.brenda-enzymes.org/) in order to associate sequences with catalytic parameters such as $K_m$ and $k_{cat}$. The performance constant ($k_{cat}/K_m$) was discretized and used as a class of catalytic efficiency. To these sequences, characterized by reactions and parameters, some molecular descriptors of sequence and chemical compounds were associated. For sequences, the descriptors were physicochemical properties such as hydrophobicity, length, molecular weight and residue-level properties such as enrichment in aliphatic, aromatic, polar, etc amino acids. For the chemical compounds, the descriptors were a quantification of properties of the molecular structure and physicochemical properties such as solubility, molecular weight, or chirality.

## 2.2 Clustering of the dataset to define groups of similar reactions

Reactions in KEGG were clustered into groups of chemical similarity based on molecular signatures, a two-dimensional molecular descriptor based on the molecular graph of a molecule [3]. This clustering determines groups of similar reactions, which were used in order to set the domain of applicability of each predictor.

## 2.3 Using random forests for the prediction of catalytic efficiency

The random forests were proposed by Breiman in 2001 [1]. A random forest is composed of decision trees which can produce a decision with a sample of descriptors.The random forests are often used with a large unbalanced dataset with a large number of descriptors. A random tree is used with a random sample of data and at each division it is a random sample of descriptors that is used.

## 2.4 Combining classifiers to increase classifier performance

A simple method for combining classifiers was used. Besides the molecular descriptors, the method takes outputs from our previously proposed molecular signatures-kernel-based predictors of promiscuity [2] and EC number [3] as additional input values to the classifier. The rationale is to exploit the classifiers complementarity and to get a tradeoff between the performance of each other. For instance, the promiscuity of an enzyme might be a factor tlinked to the catalytic efficiency.

## 3 Results

The random forest classifier parameters were the number of trees and the number of variables tested at each division. In order to minimize the out of bag (oob) error[1], we chose a number of trees of 20 and we kept the default parameter for the mtry attribute. The performance of each cluster classifier was evaluated by 10-fold cross-validation. The ROC curve (Fig. 1) shows good performances of most of the cluster classifiers. We reach an average of precision of 88% with a recall of 84% and an accuracy of 91%.

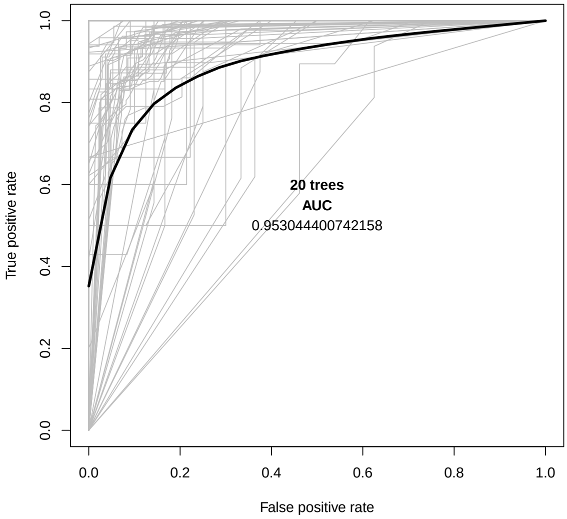


**Fig. 1.** ROC curve of the catalytic efficiency predictor trained with nbtree=20 for all the clusters by 10-fold cross-validation

We compared these performances with the performance of a classifier using balanced datasets in each cluster of reactions. We found that the accuracy was the same in the two cases (around 0.91). Moreover, we observed an increase in the performance of the catalytic efficiency predictor by adding molecular signatures-based predictions as input descriptors (Fig. 2). Finally we compared, by repeating 10 times a 10-fold cross-validation, our predictor with a classical decision tree classifier (j48) or a SVM classifier. The results (Fig. 3) showed that the error rate was lower for the random forests and suggested that this classifier was more efficient than the others.

[1] the prediction error on the data moved away from the learning dataset

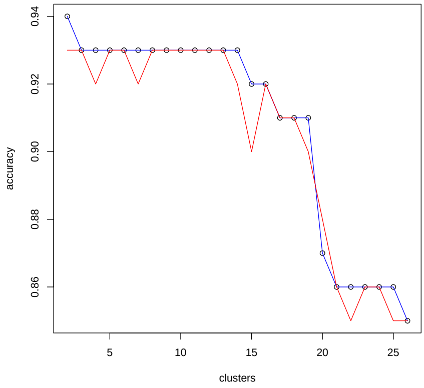


**Fig. 2.** Comparison of the accuracy of the predictor with (blue) or without (red) classifiers combination

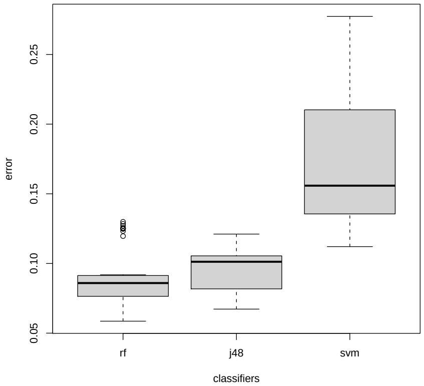


**Fig. 3.** Comparison of the accuracy of the random forest predictor with j48 and SVM classifiers

## 4 Conclusion

We introduced in this study an algorithm based on random forests to predict catalytic efficiency using sequence and chemical compounds descriptors. The classifier performance was better than those obtained in other classical classifiers. This better performance shows the reliability of our proposed catalytic efficiency predictor in order to discover novel catalytic activities for an enzyme sequence and a putative reaction.

## References

1. Breiman, J. : Random forests. Machine learning. 45, 5–32 (2001)
2. Carbonell, P., Faulon, J-L. : Molecular signatures-based prediction of enzyme promiscuity. BioInformatics. 26, 2012–2019 (2010)
3. Faulon, J-L., Misra, M., Martin, S., Sale, K., Sapra, R. : Genome scale enzyme-metabolite and drug-target interaction predictions using the signature molecular descriptor. BioInformatics. 2, 225–233 (2007)
4. Koshland, D.E. : The application and usefulness of the ratio kcat/Km. Bioorganic chemistry. 30, 211–213 (2002)
5. Liaw, A., Wiener, M. : Classification and Regression by Random forests. R News. 2/3, 18–22 (2002)

# Prediction of genetic interactions in yeast using machine learning

Marie Schrynemackers[1], Pierre Geurts[1], Louis Wehenkel[1] and M Madan Babu[2]

[1] Department of EE and CS & GIGA-R, University of Liège, Belgium
{marie.schrynemackers,P.Geurts,L.Wehenkel}@ulg.ac.be
[2] MRC Laboratory of Molecular Biology, Cambridge, UK
madanm@mrc-lmb.cam.ac.uk

We propose to use machine learning techniques to infer genetic interactions in Yeast by integrating various feature sets defined on genes. The approach is validated using four available genetic interactions maps (E-MAPs) as training networks.

The inference of the genetic interaction network of an organism is an important challenge in systems biology. The knowledge of these interactions is very useful to understand the functions of the genes and their products. In yeast S.cerevisiae, interactions subnetworks (E-MAPs) on four subsets of genes have been measured. For the time being, it remains however impossible to test experimentally the 18 millions potential interactions between the 6000 genes. In this work, we propose to use computational techniques based on machine learning to complete the experimentally confirmed interactions.

We proposed several strategies to transform this problem into one or several standard classification problems and we exploited two families of supervised learning algorithms: tree-based ensemble methods and support vector machines. We considered as inputs various feature sets, including chemo-genomic profiles, expression data, and morphological data. We validated the approach by using cross-validation on four available E-MAPs. We experimented with several protocols, including the completion of missing values in a given E-MAP and the prediction of interactions in one E-MAP from the others.

Globally, the best results are obtained with support vector machines. Cross-validation shows that we are able to predict new interactions with a reasonable accuracy. As expected, predictions of interactions between genes from the training E-MAPs are more accurate than predictions of interactions between genes not present in the training set. Some E-MAPs are also much easier to predict than others. Among input feature sets, the chemo-genomic profiles are the most predictive followed by the morphological data, while we found that expression profiles are not informative.

We have mostly focused on the prediction of negative interactions. Positive interactions are less frequent, which renders their prediction by machine learning techniques more challenging. We will now focus on these interactions. Future work will also consider the addition of other input features (e.g., interaction networks) or further methodological developments. Our ultimate goal is to make genome-wide predictions with our algorithms and to prioritise these predictions for an experimental validation.

# All Pairs Similarity Search for Short Reads

Kana Shimizu[1] and Koji Tsuda[1,2]

[1] Computational Biology Research Center(CBRC), National Institute of Advanced Science and Industrial Technology(AIST), Japan
[2] ERATO Minato Project, Japan Science and Technology Agency, Japan
shimizu-kana@aist.go.jp, koji.tsuda@aist.go.jp

**Abstract.** We developed a novel method SLIDESORT that enumerates all similar pairs from a string pool in terms of edit distance. The proposed method is based on a pattern growth algorithm that can effectively narrow down the search by finding chains of common $k$-mers. Evaluation on large-scale short read dataset shows that SLIDESORT was about 10-3000 times faster than other state-of-the-art methods.

## 1 Introduction

Recent progress in DNA sequencing technologies calls for fast and accurate algorithms that can evaluate sequence similarity for a huge amount of short reads. Searching similar pairs from a string pool is a fundamental process of de novo genome assembly, read clustering and other important analyses [4]. In this study, we designed and implemented an exact algorithm SLIDESORT that solves all pairs similarity search in terms of edit distance. Namely, given a set of $n$ sequences of equal length $\ell$, $s_1, \ldots, s_n$, SLIDESORT finds all pairs whose edit distance is at most $d$,

$$E = \{(i,j) \mid \mathit{EditDist}(s_i, s_j) \leq d, i < j\}. \quad (1)$$

Basically, similarity search problems are solved by finding a common $k$-mer and verifying the match or backtracking in an index structure of suffix array. Either approaches or a combination of the two approaches do not work well for short strings with large radius, because $k$-mer match of short length generates too many candidate pairs to be verified and the backtracking cost of suffix array is exponential to $d$. Using an efficient pattern growth algorithm, SLIDESORT discovers chains of common $k$-mers to narrow down the search without using large memory, and effectively reduces the number of edit distance calculations. As a result, it scales easily to 10 million sequences and is much faster than seed matching methods and suffix arrays for short sequences and large radius.

## 2 Method

Two similar strings share common substrings *in series*. Therefore, we can detect similar strings by detecting chains of common strings systematically. More precisely, there exists following property for $s_i$ and $s_j$ of $\mathit{EditDist}(s_i, s_j) < d$.

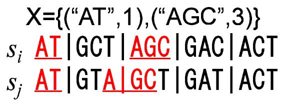


**Fig. 1.** An example pattern for block size 5 and edit-distance threshold 3. $s_i$ matches to $X$ with no offset in the first block and the third block. $s_j$ matches to $X$ with no offset in the first block but with -1 offset in the third block.

*If $s_i$ are divided into b blocks ($b > d$), there exists at least b-d blocks that exactly match to $s_j$ with bounded slide width $-\lfloor d/2 \rfloor \leq \boldsymbol{p} \leq \lfloor d/2 \rfloor$.*

SLIDESORT utilizes this property and systematically finds groups of sequences which share a chain of common $b - d$ blocks in the similar way to multiple sorting method [3], and then calculates edit distance of all sequence pairs in each group. In many case of the short read analyses, the size of the group sharing a long common substring is much smaller. Thus the proposed method can largely reduce costly edit distance calculations.

To find common $b-d$ blocks effectively, pattern growth approach is employed. Let us define a *pattern* of length $k$ be a sequence of strings and block indices,

$$X = [(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_k, y_k)],$$

where $x_i$ is a string and $y_i$ is a block index. Pattern $X$ matches to string $s$ with offset $\boldsymbol{p}$, if $x_i$ matches to $y_i$-th block with slide width $p_i$, for $i = 0, \ldots, k$. All occurrences of $X$ in the database are denoted as

$$C(X) = \{(i, \boldsymbol{p}) \mid X \text{ matches } s_i \text{ with offset } \boldsymbol{p}\}.$$

For convenience, the occurrence set $O(X)$ is defined as the set of sequences appearing in $C(X)$. The occurrence frequency (*support*) of $X$ is defined as $|O(X)|$. Figure 1 illustrates an example of patterns with $b = 5, d = 3$.

All patterns of length $b - d$ are enumerated by a recursive pattern growth algorithm. In the algorithm, a pattern tree is constructed, where each node corresponds to a pattern (Figure 2). Nodes at depth $k$ contain patterns of length $k$. At first, patterns of length 1 are generated as follows. For each block $y_1 = 1, \ldots, d + 1$, all substrings corresponding to $y_1$ are collected from database with offsets $-\lfloor d/2 \rfloor \leq \boldsymbol{p} \leq \lfloor d/2 \rfloor$ and stored in a string pool. Applying radix sort to the string pool and scanning through the sorted result, repetition of equivalent strings can be detected. Each pattern of length 1, denoted as $X_1$, is constructed as a combination of the repeated string $\boldsymbol{x}_1$ and $y_1$, $X_1 \leftarrow \{(\boldsymbol{x}_1, y_1)\}$. At the same time, all occurrences $C(X_1)$ are recorded. If $s_i$ matches the same pattern $X_1$ by several different offsets, only the smallest offset is recorded. They form the nodes corresponding to depth 1 of the pattern tree. Given a pattern $X_t$ of length $t$, its children in the pattern tree are generated similarly as follows. For each $y_{t+1} = y_t + 1, \ldots, d + t + 1$, a string pool is made by collecting substrings of $O(X_t)$ corresponding to $y_{t+1}$ with offsets $-\lfloor d/2 \rfloor \leq \boldsymbol{p} \leq \lfloor d/2 \rfloor$. Because the string pool is made from the occurrence set only, the size of the pool decreases sharply as a pattern grows. By sorting and scanning, a next string $x_{t+1}$ is identified and the

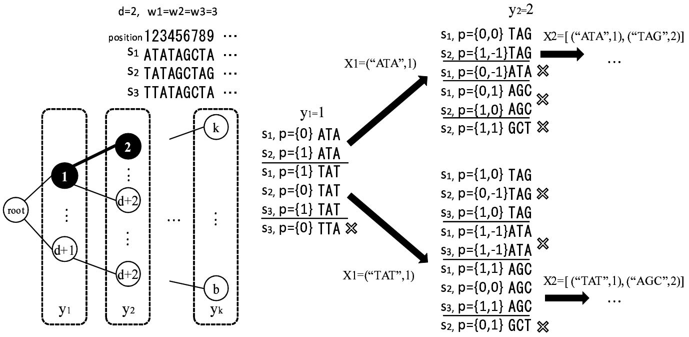


**Fig. 2.** Pattern growth and pruning process of the proposed method. Patterns are enumerated by traversing the tree in depth first manner. In each node, new elements are generated by sorting substrings in sequence pool (”ATA”, ”TAT”, ”TTA” for $y_1 = 1$). Useless patterns (”TTA” in this case) are removed. Remaining elements are added to yield new patterns. This process is executed by recursive call until the pattern size reaches $b - d$.

pattern is extended as $X_{t+1} \leftarrow X_t + \{(\boldsymbol{x}_{t+1}, y_{t+1})\}$, and the occurrences $C(X_t)$ are updated to $C(X_{t+1})$ as well. The pattern tree is effectively pruned to avoid generating useless patterns. As pattern growth proceeds in a depth-first manner, peak memory usage is kept small. As implied in the property, every neighbor pair (Figure 1) appears in the occurrence set $O(X)$ of at least one pattern. Since one of the pair must have zero offset, the set of eligible pairs is described as

$$P_X = \{(i,j) | i < j, i, j \in O(X), s_i \text{ matches } X \text{ with zero offset}\}.$$

We can ensure that no pair is reported twice by considering lexicographical order of a pattern and offsets. Since not all members of $P_X$ correspond to neighbors, we have to verify if they are neighbors by actual edit distance calculation.

## 3 Results and Discussion

The proposed method was compared to the state-of-the-art tools BWA [2] and SeqMap [1]. The former is based on suffix array and the latter is based on an ELAND-like methodology of using multiple indexes for all block combinations. BWA and SeqMap are applied to all pairs similarity search by creating an index from all short reads and querying it with the same set of reads. Also, our method was compared to the naive approach that calculates edit distances of all pairs. All the tools are evaluated on a public short read dataset generated by Illumina Genome Analyzer with sequence length 87 (SRR020262) which is downloaded from NCBI Sequence Read Archive. Figure 3 plots computation time and memory usage against the distance threshold $d$. SLIDESORT is consistently faster in all configurations. As the number of sequences grows and the

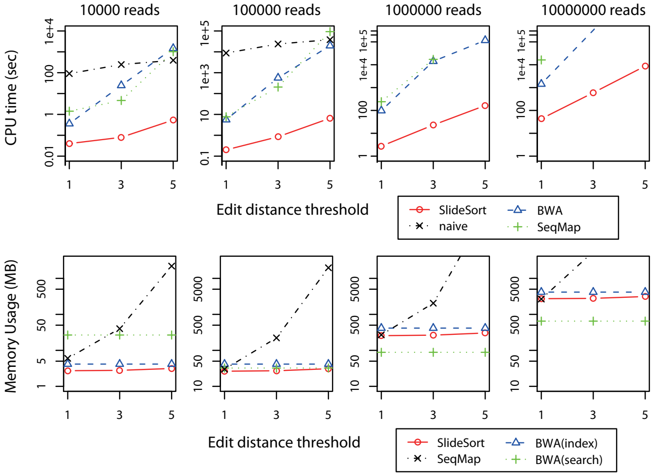


**Fig. 3.** Computation time and Memory usage.

distance threshold is increased, the difference from BWA and SeqMap becomes increasingly evident. Not all results are obtained, because of the 30GB memory limit and 300,000 seconds time limit. The peak memory of BWA for the search step is the smallest in most of the configurations, while that of SLIDESORT is comparable or slightly better than BWA's peak indexing memory. BWA is most efficient in space complexity, because its index size does not depend on the distance threshold. Instead, BWA's time complexity rapidly deteriorates as the edit distance threshold grows due to explosion of the number of traversed nodes in backtracking. In contrast, SeqMap indexes and hashes all the combination of key blocks, which leads to huge memory usage. SLIDESORT is similar to SeqMap in that it considers all block combinations, but is much more memory efficient. The difference is that SLIDESORT is an indexing free method which dynamically generates the pattern tree by depth first traversal. It allows us to maintain only necessary parts of tree in memory. All these results demonstrate practical merits of SLIDESORT.

## References

1. Jiang, H., Wong, W.H.: Seqmap: mapping massive amount of oligonucleotides to the genome. Bioinformatics **24**(20) (2008) 2395–6
2. Li, H., Durbin, R.: Fast and accurate short read alignment with burrows-wheeler transform. Bioinformatics **25**(14) (2009) 1754–60
3. Uno, T.: An efficient algorithm for finding similar short substrings from large scale string data. PAKDD (2008) 345–356
4. Zerbino, D.R., Birney, E.: Velvet: algorithms for de novo short read assembly using de bruijn graphs. Genome Res **18**(5) (2008) 821–9

# Discovering groups of genes with coordinated response to *M. leprae* infection

Ivica Slavkov[1], Darko Aleksovski[1], Nigel Savage[2], Kimberley V. Walburg[2], Tom H.M. Ottenhoff[2] and Sašo Džeroski[1]

[1] Department of Knowledge Technologies, Jožef Stefan Institute,
Jamova 39, 1000 Ljubljana, Slovenia
{ivica.slavkov, darko.aleksovski,saso.dzeroski}@ijs.si

[2] Department of Infectious Diseases, Leiden University Medical Center
Albinusdreef 2, 2333 ZA Leiden, Netherlands
{t.h.m.ottenhoff, k.v.walburg, n.d.l.savage} @lumc.nl

Gene expression is a temporal process that is highly regulated. Much work in bioinformatics studies this process in order to better understand the function of individual genes and to gain insight in complete biological systems. The task most commonly addressed in this context is the task of clustering time series of gene expression data, where the aim is to discover groups of genes with similar temporal profiles of expression and to find common characteristics of the genes in each group. Clustering genes by their time expression pattern is important, because genes that are co-regulated or have a similar function will have similar temporal profiles under certain conditions.

In our work, we develop and apply a clustering approach that is well suited for analysing short time series. Besides finding clusters, e.g., groups of genes, we also aim to find descriptions/explanations for the clusters. Instead of first clustering the expression time series and elucidating the characteristics of the obtained clusters later on, we perform so-called constrained clustering, which yields both the clusters and their symbolic descriptions all in one step.

The constrained clustering is performed by using predictive clustering trees (PCTs), which are a part of a more general framework, namely predictive clustering [1]. Predictive clustering partitions a given dataset into a set of clusters, such that the instances in a given cluster are similar to each other and dissimilar to the instances in other clusters. In this sense, predictive clustering is identical to regular clustering [3]. The difference is that predictive clustering associates a predictive model to each cluster.

In our specific analysis scenario, the prediction associated to each leaf of the PCT is a gene time-course expression profile. The descriptions associated to the cluster, i.e., the nodes of the PCT, are derived from the Gene Ontology (GO) [2]. A sample PCT is presented in Figure 1. On the left, there is a graphical representation of a PCT. Each internal node of the tree contains a GO term. The leaves of the tree contain the clusters ($C_1$ through $C_5$)of genes sharing common GO annotations. Each cluster also has a temporal response assigned to it the best, i.e., cluster prototype.

The rest of the figure represents the output of the PCT: the temporal profiles of each cluster, the number of genes in each cluster, and the error of the cluster prototype. The heatmap is another type of visualisation of the cluster prototype, described by its corresponding GO annotations, at the rightmost side on the figure. A detailed description of the use of PCTs for clustering gene expression time-course data can be found in [4].

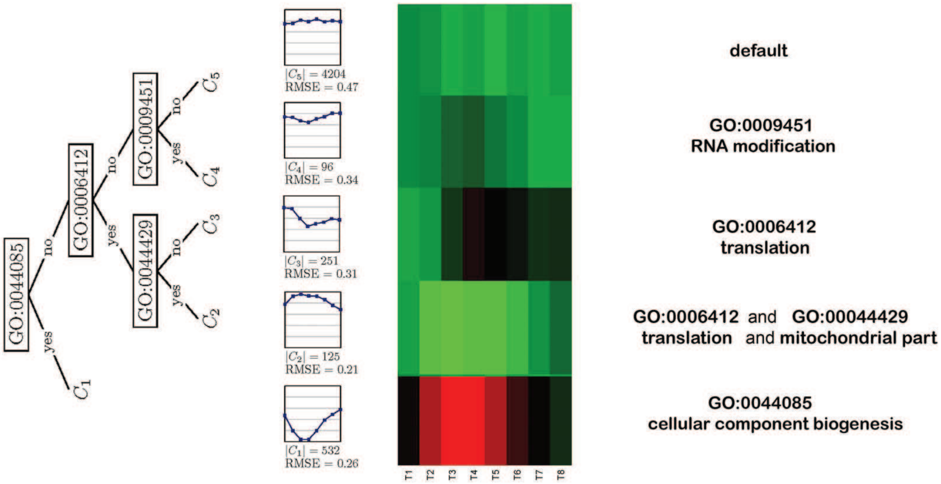


**Fig. 1.** A sample PCT used for clustering short time series of gene expression data

In this work, we use PCTs to analyse temporal expression profiles as observed in Schwann cells. Our primary interest was the temporal response of Schwann cells to infection with live *Mycobacterium leprae*. For comparison, the Schwann cells have also been exposed to four other different conditions, namely: infection with irradiated/sonicated *M. leprae*, infection with *M. smegmatis* expressing a *M.leprae* adhesion molecule, and control (growth in medium) conditions. For all conditions gene expression was measured at five distinct time points. The purpose of our analysis was to investigate which groups of genes, involved in certain cellular processes are responding in a coordinated manner to the different kind of stimuli.

**Table 1.** Identified groups of genes with a coordinated response to live *M. leprae* infection. Each group is described by combination of GO terms from the different GO hierarchies and they include different number of genes

| **Group description** | **Size** |
|---|---|
| cytoplasmic part; regulation of G-protein coupled receptor protein signaling pathway | 11 |
| cell part; microtubule-based process | 17 |
| cytoplasmic part; protein binding; guanyl-nucleotide exchange factor activity | 28 |
| protein binding; positive regulation of ligase activity | 27 |
| cell part; macromolecular complex; intermediate filament | 12 |
| cell part; regulation of ligase activity | 8 |
| protein binding; intracellular; GTPase activity | 12 |
| cytoplasmic part; protein binding; clathrin coated vesicle membrane | 26 |
| cytoplasmic part; translational elongation | 87 |
| protein binding; collagen | 21 |

The results show most distinctive difference in activation between pathways and cellular process during live infection as compared to the control case. As expected, if cells grow undisturbed in medium, the genes showing the most distinct activity profiles are those involved in general life-sustaining processes, such as mitochondrial respiration and ribosomal proteins synthesis. In contrast to this, when Schwann cells undergo infection with live *M. leprae*, many groups of genes with specific functions are up- or down- regulated, for example the regulation of G-protein coupled receptor protein signalling pathway. Gene ontology descriptions of some groups of genes that have a coordinated response to live *M. leprae* infection are presented in Table 1.

In sum, we have successfully applied predictive clustering to group human genes into clusters with similar temporal profiles of expression. This was studied in the context of infection of Schwann cells with *M. leprae*. We obtained clearly described groups of genes with distinct temporal profiles of expression. Groups of this kind can be used to identify pathways regulating the processes of interest (response to *M. leprae* infection).

## References

1. H. Blockeel, L. De Raedt, and J. Ramon. Top-down induction of clustering trees. In *15th Int'l Conf. on Machine Learning*, pages 55–63, 1998.
2. M. Ashburner et al. Gene Ontology: Tool for the unification of biology. The Gene Ontology Consortium. *Nature Genet.*, 25(1):25–29, 2000.
3. L. Kaufman and P.J. Rousseeuw, editors. *Finding groups in data: An introduction to cluster analysis.* John Wiley & Sons, 1990.
4. I. Slavkov, V. Gjorgjioski, J. Struyf, and S. Džeroski. Finding explained groups of time-course gene expression profiles with predictive clustering trees. *Molecular BioSystems*, 6(4):729–740, 2010.

# A Hierarchical Poisson Model for Next Generation cDNA Sequencing Libraries

Helene Thygesen[1], Peter-Bram 't Hoen[2] and A.H. Koos Zwinderman[3]

[1] Lancaster University
helene.h.thygesen@gmail.com
[2] LUMC P.A.C._t_Hoen@lumc.nl
[3] AMC a.h.zwinderman@amc.nl

Since the introduction of next generation sequencers, the sequencing of cDNA libraries is becoming an attractive alternative to microarrays for gene expression experiments. But the statistical methodology for cDNA sequencing data is not as mature as that for microarrays. The problem is that while the biological processes underlying the gene expression levels are best modeled using linear models (the distribution of the gene expression levels across genes(transcripts) is log-normal), the the measurement process is basically a count process. This calls for a lognormal-Poisson model, a model which is difficult to make inference in since the likelihood functions is not available on closed form.

We developed estimators for a multivariate log-normal Poisson model, in which the per-transcript variance is gamma distributed. The model provides good fit to real. An application of this model is shown on the poster, using simulated data: Model-based principal component analysis is much better than PCA based on raw or transformed data in terms of separating the main tissue sample effect (which is not interesting since it just reflects the dilution of the mRNA) from the expression profile. This allows users to visualize clustering of transcripts or (as illustrated on the poster) tissue samples.

Inference in models including tissue-sample related covariates, for example identification of differentiated transcripts in a two-group comparison study, is also possible, but depends on the modeling of the covariate effects as random effects across transcripts. This is difficult since the covariates in typical experiments only explain a small fraction of the overall variance in the data.

# Gene regulatory network reconstruction with a combination of genetics and genomics data

Jimmy Vandel, Simon de Givry, Brigitte Mangin, and Matthieu Vignes

BIA Unit, INRA Toulouse, France

**Abstract.** We develop here a discrete Bayesian network modeling with a structure learning approach for Gene Regulatory Network (GRN) reconstruction. It endeavours genetic variability (measured by markers on the genome) in a segregating population as a cause to genomics observations. Our results suggest that it improves the deciphering of GRN.

**Keywords:** discrete Bayesian network, graph inference, gene regulation, genetical genomics

## 1 Introduction

The gene is the functional unit carrying from one generation to the next information that allows organisms to achieve a proper survival. They are expressed in the cell so that in the end proteins, the active molecules of living organisms are produced. A better understanding of the regulation of genes is a gain towards dealing with genetical desease in animals or susceptibility to stresses in plants. Genes do not act independently from each other. They fulfil their role in a concerted manner. A convenient modelling of these interactions involve networks and inferring relationships in a GRN is a complex task. In particular, plenty of gene expression data governed by even more regulations are measured on a small sample of individuals (the segregating population) that represent a small subset of genetical variability in terms of background within the species but rich in terms of a cross between two strains that have a known difference in a phenotype of interest. Whilst first approaches have focused on thresholding local relationships between genes (*e.g.* correlations) to reconstruct the global network ([1]), recent approaches are, to our knowledge of two kinds: based either on Structural Equation Modeling (SEM, [4]) or on Bayesian Networks (BN, [3]). We chose to explore this latter probabilistic modeling. More precisely, we propose and assess a probabilistic model for both genetics and genomics data. It presuppose that we have a dense genetic map, that every gene has a measured stationary-state expression on the microarray, that the environment is fully controlled and that no epigenetic effect needs to be inclued, that is gene regulation signals exclusively stem from the sequence.

## 2 Bayesian network modeling and structure learning

A Bayesian network ([2]) is a Directed Acyclic Graph (DAG). Discrete random variables $(X_i)_{i=1\ldots n}$ modeling observations are on the edges of the graph and diredted edges state conditional dependencies in the joint distribution of the variables:

$$P(X) = \prod_{i=1}^{n} P(X_i \mid Pa(X_i)),$$

where $Pa(X_i)$ is the set of parents of $X_i$ in the network.

We first present a modeling that distinguish between two kinds of variables : $M_i$ for the 'marker' variable that can take two different values : 'normal' (0) or 'muted' (1) and $E_i$ for the gene expression data observation, the number of levels of which depends on the chosen discretization method (either a modified k-means or a Gaussian mixture). An additional simplification is that there is at most one funtional polymosrphism and that it is linked to a point mutation or Single Nucleotide Polymorphism (SNP) that can be identifed thanks to a close enough marker. This is described in Fig. 1.

We chose a simplified structure for the network by *fusing* the two nodes associated to a single gene in the network (Fig. 2). Its advantage is that the number of nodes is decreased two-fold making the structure inference much faster without a loss of information as far as the GRN reconstruction is concerned. The final output is the partially directed BN that represents the Markov equivalence class of the found network.

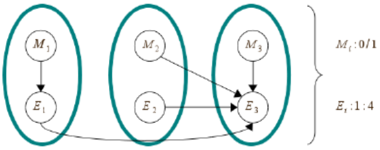


**Fig. 1.** Non fused model with the three different possible regulations: (i) *cis* - $M_3$ value impacts the expression $E_3$ so the mutation has to be in the promoter region, (ii) *cis-trans* - gene 3 is regulated by $E_1$: the mutation in the promoter region of $M_1$ can change its expression that in turns has an influence of the expression $E_3$ and (iii) *trans* - gene 2 regulates $E_3$ according to the status of $M_2$ (on the coding sequence).

Two main families of BN learning methods compete: maximum scoring of a fit of the network to available data and conditional independence testing. We focused on the former one. We specified the Bayesian Information Criterion in our setting:

$$BIC = \sum_{l} \sum_{i} \left[\log P(e_i \mid Pa(g_i^l), m_i^l) + \log P(m_i^l \ midPa(g_i^l))\right] - \frac{1}{2} \log m.Dim(B_G),$$

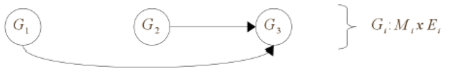


**Fig. 2.** Fused model that corresponds to the one in Fig. 1

where $m$ is the number of samples ($<< n$) and $g_i^l = (e_i^l, m_i^l)$ is the expression and genotype observation for gene $i$ and sample $l$. Note that the second term only depends upon the genetic linkage between markers and doesn't need to be learned so it is removed from the optimized function. We set $Dim(B_G) = \sum_i q_i$, $q_i$ being the number of possible configurations for $Pa(g_i)$. A Greedy Search (GS, from a MatLab library) was then applied as the number of possible networks would exceed computational limitations for more than $\sim 30$ nodes: simple local modification to an initial graph are considered (edge deletion, adding or reversal) and the one that maximises the score is kept. The initial graph stems from a statistical analysis to select for each (continuous) expression level $E_i$ the set of markers (limited to 9 for computational sake) which explains at best its variability. This selction takes place in a linear regresion model for Quantitative Trait Loci (QTL) cartography. We performed it with the MCQTL software (`http://carlit.toulouse.inra.fr/MCQTL/`). Since it should define a BN, cycles had to be removed with a heuristics.

## 3 Genetical genomics data simulation

Fifty 50-gene scale-free (hence having some biological feature) networks were retrieved from `http://www.comp-sysbio.org/AGN/`. Genotypes for $m = 500$ (so not really ) backcross individuals on a single 10-Morgan chromosome were generated (markers were randomly placed and the mutation can either be in the promoter or in the coding region but fixed among the population) with CarthaGène (`http://www.inra.fr/mia/T/CarthaGene/`). The gene expression simulation was based on an ODE and steady-state were obtained from this gene expression network dependencies (see [4] and `http://www.copasi.org/`). As mentioned earlier, gene expression data need to be discretized, a crucial step for downstream analysis. We chose a Gaussian mixture model approach and coupled it to a modified k-means when the expression level distribution is unimodal. More insight in this step could be of interest.

## 4 Experimental results and concluding remarks

Results are presented in terms of sensitivity ($= \frac{TP}{TP+FN}$) and precision ($= \frac{TP}{TP+FP}$) without taking into account the directions of edges. Those are means over the 50 network from Section 3. Table **??** the gain in using genetic data in addition to expression data only. We also show that the eQTL analysis provides a good initial graph to search the structure of the BN. BIC curves were also looked at (data not shown). An example of such a network can be found on Figure 3.

**Table 1.** (i) Exp: BN on $E_i$'s only, (ii) Exp+Gen, MWST init: fused BN (see Section 2) with a maximum spanning tree as an initial graph and (iii) Exp+Gen, eQTL init: same model with the eQTL analysis network as input of the algorithm

| | Exp only | Exp+Gen, MWST init | Exp+Gen, eQTL init |
|---|---|---|---|
| Precision | 0.26 | 0.52 | 0.61 |
| Sensibility | 0.23 | 0.39 | 0.48 |
| Edge number $(TP + FP)$ | 45 | 37 | 40 |

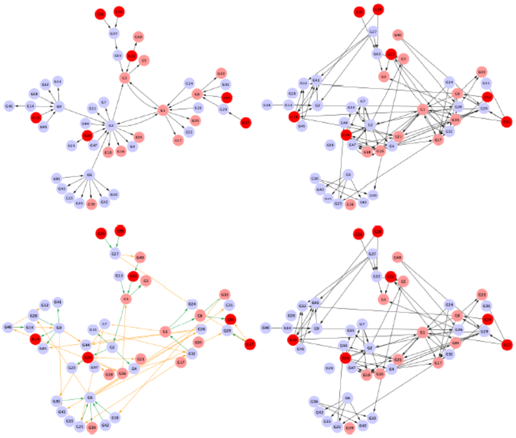


**Fig. 3.** Network reconstruction example: (top left) initial network to recover, (top right) network from the eQTL analysis, (bottom left) intial BN after edge removal and (bottom right) GS result. Gene with mutation in promoter region are in blue, those in coding region in pink or red (not regulated hence with constant gene expression level). A green edge is a TP, a wellow one is a FP.

We presented a method for the inference of structure of a BN that represents a GRN from both genetic and genomics data. Moreover, the eQTl statistical analysis seems a complementary method to the presented method. Still our absolute results seem a wee bit below those of [4] but our datasets cannot be compared. Future work include a thorough comparison with classical methods for GRN inference either from expression data only or from genetical genomics data: SEM, Gaussian graphical modeling, . . . . We also would like to use more efficient algorithms for structure inference and test our model in a situation where the number of sample is truly smaller than the number of genes. This is the topic of the next 'Systems Genetics' challenge of the DREAM5 competition.

## References

1. Ghazalpour A. *et al.*: Integrating genetic and network analysis to characterize genes related to mouse weight. Plos Genetics, 2, 1182–1192 (2006).
2. Naim P. *et al.*: Réseaux bayésiens. Eyrolles, 3rd edition (2008).
3. Zhu J. *et al.*: Increasing the power to detect causal associations by combining genotypic and expression data in segregating populations. Plos Comput Biol, 3(4), 692–703 (2007).
4. Liu B. *et al.*: Gene network inference via structural equation modeling in genetical genomics experiments. Genetics, 17, 1763–1776 (2008).

# Author Index