On Identifying Significant Edges in Graphical Models

Marco Scutari[1] and Radhakrishnan Nagarajan[2]

[1]Genetics Institute
University College London
m.scutari@ucl.ac.uk

[2]Division of Biomedical Informatics
University of Arkansas for Medical Sciences
rnagarajan@uams.edu

July 2, 2011

Graphical Models: Definitions & Learning

Graphical Models

Graphical models are defined by two components:

- a network structure, either an undirected graph (Markov networks [2, 19], gene association networks [14], correlation networks [17], etc.) or a directed graph (Bayesian networks [7, 8]). Each node corresponds to a random variable;
- a global probability distribution, which can be factorised into a small set of local probability distributions according to the topology of the graph.

This combination allows a compact representation of the joint distribution of large numbers of random variables and simplifies inference on the parameters of the model.

Structure and Parameter Learning

Likewise, learning a graphical model is a two-stage process:

1. structure learning: learning the structure of the network underlying the graphical model, i.e. estimating the dependencies present in the data and adding the associated edges to the model;
2. parameter learning: using the decomposition into local probabilities given by the network structure learned in the previous step to estimate the parameters of the local distributions.

Several approaches have been proposed for both steps [1, 7], covering all aspects of graphical model estimation.

Network Structure Validation

Model validation techniques have not been developed at a similar pace, particularly in the case of network structures:

- the few available measures of structural difference are completely descriptive in nature (i.e. Hamming distance [6] or SHD [18]), and are difficult to interpret;
- unless the true global probability distribution is known it is difficult to assess the quality of graphical models without ad-hoc solutions; this limits the study of the properties of network structures to few reference data sets [3, 9].

A more systematic approach to model validation, and in particular to the problem of identifying statistically significant edges in a network, is required for graphical models learned from real data.

Identifying Significant Edges

Friedman's Confidence

Friedman et al. [4] proposed an approach to model validation based on bootstrap resampling and model averaging:

1. For $b = 1, 2, \ldots, m$:
   1.1 sample a new data set $\mathbf{X}_b^*$ from the original data $\mathbf{X}$ using either parametric or nonparametric bootstrap;
   1.2 learn the structure of the graphical model $G_b = (\mathbf{V}, E_b)$ from $\mathbf{X}_b^*$.
2. Estimate the confidence that each possible edge e_i is present in the true network structure $\mathcal{G}_0 = (\mathbf{V}, E_0)$ as

$$\hat{p}_i = \hat{\mathrm{P}}(e_i) = \frac{1}{m} \sum_{b=1}^{m} \mathbb{1}_{\{e_i \in E_b\}},$$

where $\mathbb{1}_{\{e_i \in E_b\}}$ is equal to 1 if $e_i \in E_b$ and 0 otherwise.

Evaluating Confidence Values

- The confidence values $\hat{\mathbf{p}} = \{\hat{p}_i\}$ do not sum to one and are dependent on one another in a nontrivial way; the value of the confidence threshold (i.e. the minimum confidence for an edge to be accepted as an edge of $\mathcal{G}_0$) is an unknown function of both the data and the structure learning algorithm.
- The ideal/asymptotic configuration $\tilde{\mathbf{p}}$ of confidence values would be

$$\tilde{p}_i = \begin{cases} 1 & \text{if } e_i \in E_0 \\ 0 & \text{otherwise} \end{cases},$$

  i.e. all the networks $\mathcal{G}_b$ have exactly the same structure.
- Therefore, identifying the configuration $\tilde{\mathbf{p}}$ "closest" to $\hat{\mathbf{p}}$ provides a statistically-motivated way of identifying significant edges and the confidence threshold.

The Confidence Threshold

Consider the order statistics $\tilde{\mathbf{p}}_{(.)}$ and $\hat{\mathbf{p}}_{(.)}$ and the cumulative distribution functions (CDFs) of their elements:

$$F_{\hat{\mathbf{p}}_{(.)}}(x) = \frac{1}{k} \sum_{i=1}^{k} \mathbb{1}_{\{\hat{p}_{(i)} < x\}}$$

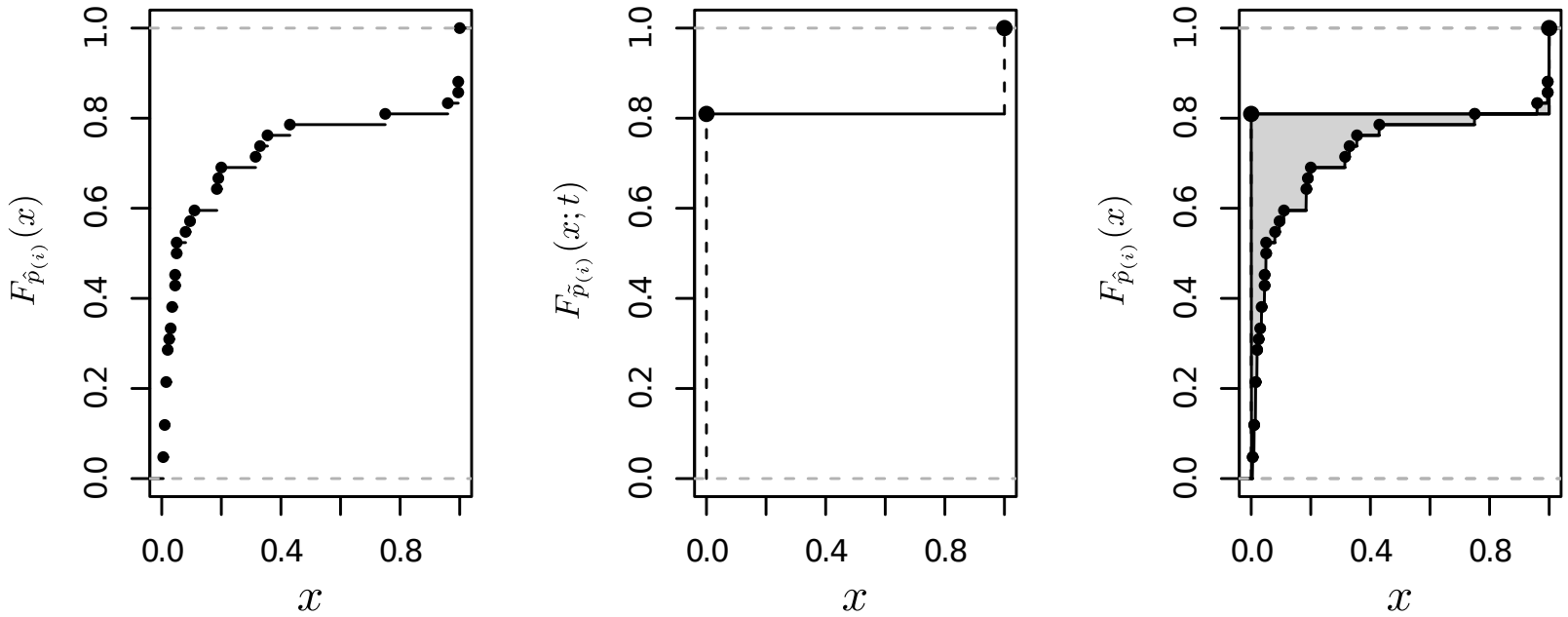
and

$$F_{\tilde{\mathbf{p}}_{(.)}}(x;t) = \begin{cases} 0 & \text{if } x \in (-\infty, 0) \\ t & \text{if } x \in [0, 1) \\ 1 & \text{if } x \in [1, +\infty) \end{cases} .$$

t corresponds to the fraction of elements of $\tilde{\mathbf{p}}_{(.)}$ equal to zero and is a measure of the fraction of non-significant edges, and provides a threshold for separating the elements of $\tilde{\mathbf{p}}_{(.)}$:

$$e_{(i)} \in E_0 \iff \hat{p}_{(i)} > F^{-1}_{\tilde{\mathbf{p}}_{(.)}}(t).$$

The CDFs $F_{\hat{\mathbf{p}}_{(\cdot)}}(x)$ and $F_{\tilde{\mathbf{p}}_{(\cdot)}}(x;t)$



One possible estimate of t is the value $\hat{t}$ that minimises some distance between $F_{\hat{\mathbf{p}}_{(\cdot)}}(x)$ and $F_{\tilde{\mathbf{p}}_{(\cdot)}}(x;t)$; an intuitive choice is using the L_1 norm of their difference (i.e. the shaded area in the picture on the right).

An L_1 Estimator for the Confidence Threshold

Since $F_{\hat{\mathbf{p}}_{(\cdot)}}$ is piecewise constant and $F_{\tilde{\mathbf{p}}_{(\cdot)}}$ is constant in $[0, 1]$, the L_1 norm of their difference simplifies to

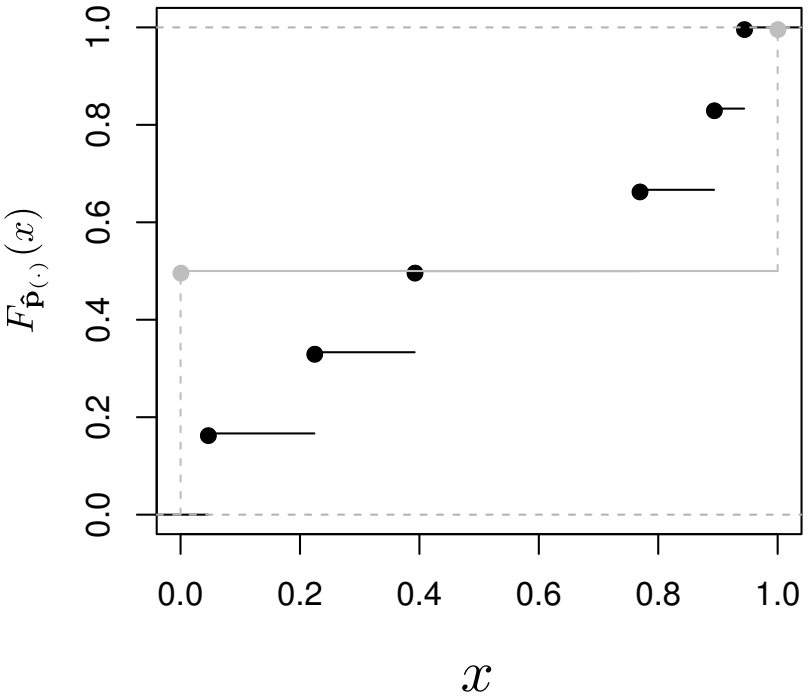
$$\begin{aligned} L_1\left(t; \hat{\mathbf{p}}_{(\cdot)}\right) &= \int \left| F_{\hat{\mathbf{p}}_{(\cdot)}}(x) - F_{\tilde{\mathbf{p}}_{(\cdot)}}(x; t) \right| dx \\ &= \sum_{x_i \in \left\{\{0\} \cup \hat{\mathbf{p}}_{(\cdot)} \cup \{1\}\right\}} \left| F_{\hat{\mathbf{p}}_{(\cdot)}}(x_i) - t \right| (x_{i+1} - x_i). \end{aligned}$$

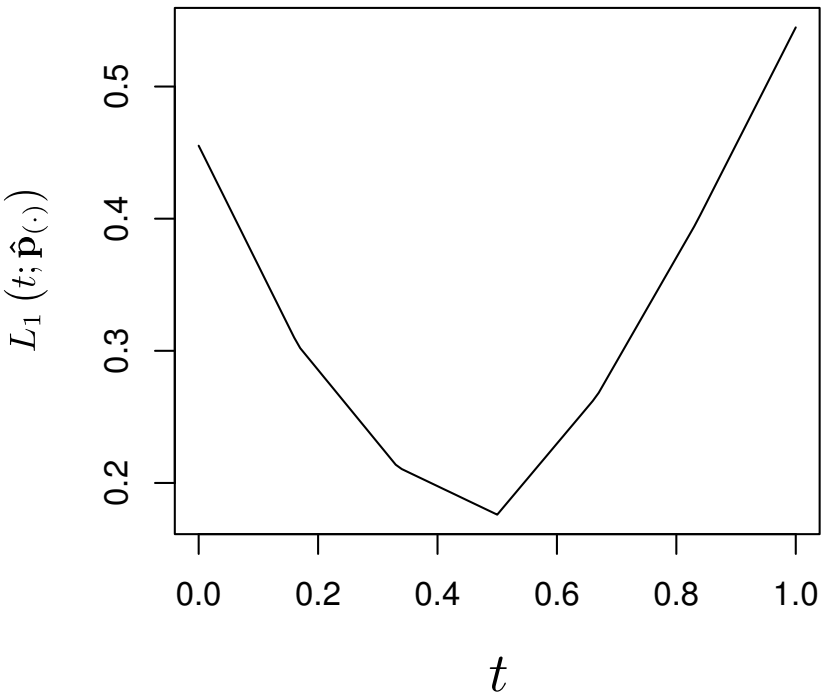
This form has two important properties:

- can be computed in linear time from $\hat{\mathbf{p}}_{(\cdot)}$;
- its minimisation is straightforward using linear programming [11].

Furthermore, the L_1 norm does not place as much weight on large deviations as other norms (L_2, L_∞), making it robust against a wide variety of configurations of $\hat{\mathbf{p}}_{(\cdot)}$.

A Simple Example





Consider a graph with 4 nodes and confidence values

$$\hat{\mathbf{p}}_{(\cdot)} = \{0.0460, 0.2242, 0.3921, 0.7689, 0.8935, 0.9439\}$$

Then $\hat{t} = \min_t L_1\left(t; \hat{\mathbf{p}}_{(\cdot)}\right) = 0.4999816$ and $F^{-1}_{\tilde{\mathbf{p}}_{(\cdot)}}(0.4999816) = 0.3921$; only three edges are considered significant.

Applications to Gene Networks

Analysis of Functional Relationships

We measured the effectiveness of the proposed method on two gene networks from Nagarajan et al. [10] and Sachs et al. [13] using the bnlearn package [16, 15] for R [12].

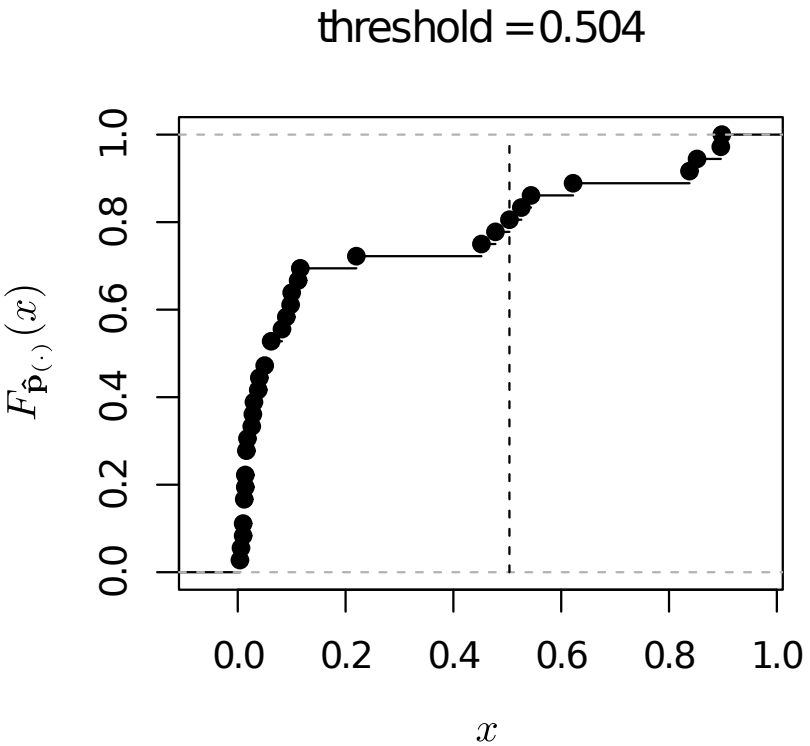
- Functional relationships have been investigated using Bayesian networks, as in the original papers;
- 500 bootstrapped network structures G_b have been learned from each data set, with the same learning algorithms, scores and parameters as in the original papers;
- Following Imoto et al. [5], we will consider the edges of the Bayesian networks disregarding their direction. Edges identified as significant will be oriented according to the direction observed with the highest frequency in the bootstrapped networks $\mathcal{G}_b$.

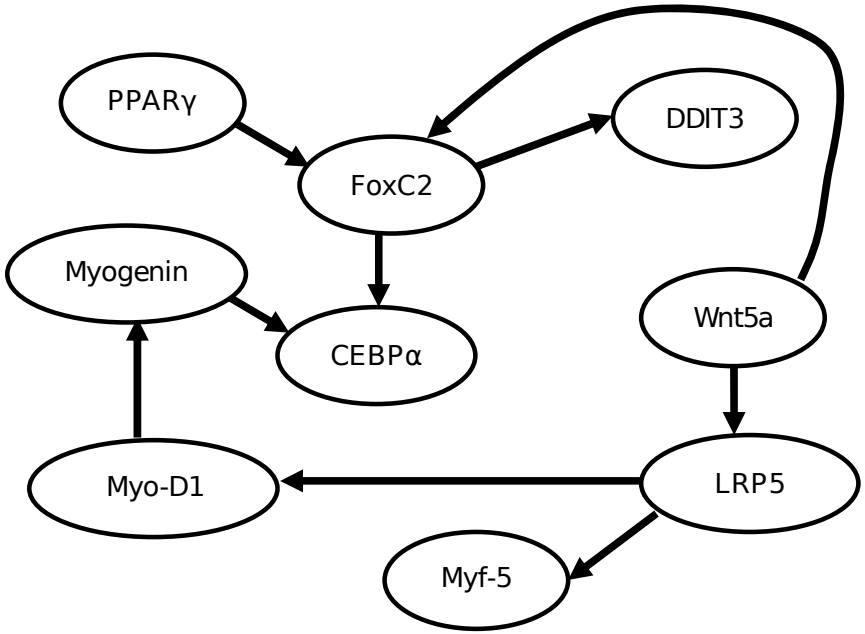
Differentiation Potential of Aged Myogenic Progenitors

The clonal gene expression data in Nagarajan et al. [10] was generated (for 12 genes) from RNA isolated from 34 clones of myogenic progenitors obtained from 24-months old mice. The objective was to study the interplay between crucial myogenic, adipogenic, and Wnt-related genes orchestrating aged myogenic progenitor differentiation.

In the same study, the authors estimated the significance threshold by randomly permuting the expression of each gene and learning Bayesian network structures from the resulting data sets. Model averaging of these networks provided the noise floor distribution for the edges; confidence values falling outside its range were deemed significant. This approach, however, is slower than just computing an L_1 norm and may result in a large number of false positives on large data sets.

Differentiation Potential of Aged Myogenic Progenitors





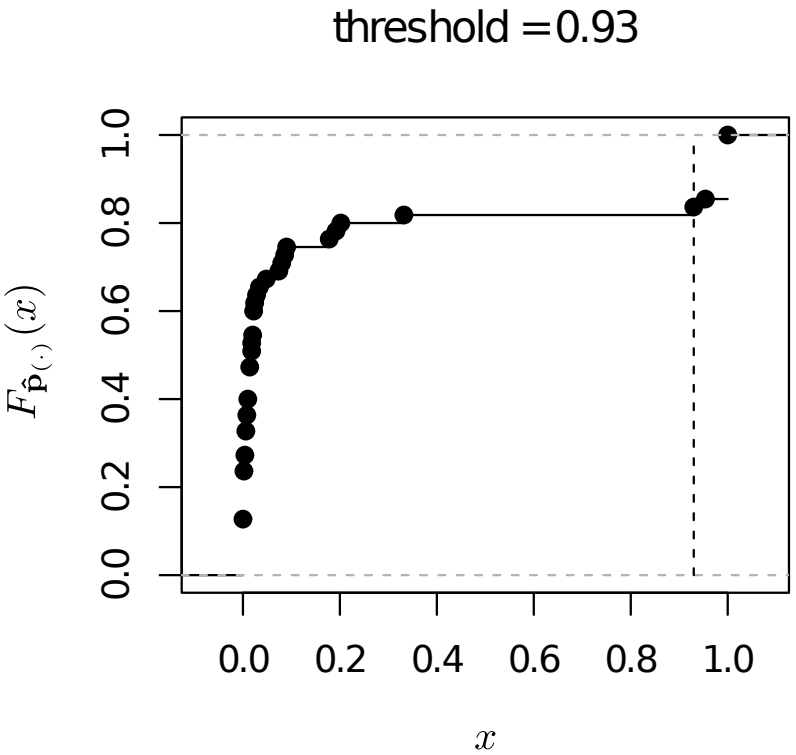
All edges identified as significant in the earlier study are also identified by the proposed approach; directionality of the edges is also revealed, unlike the original network in Nagarajan et al. [10].

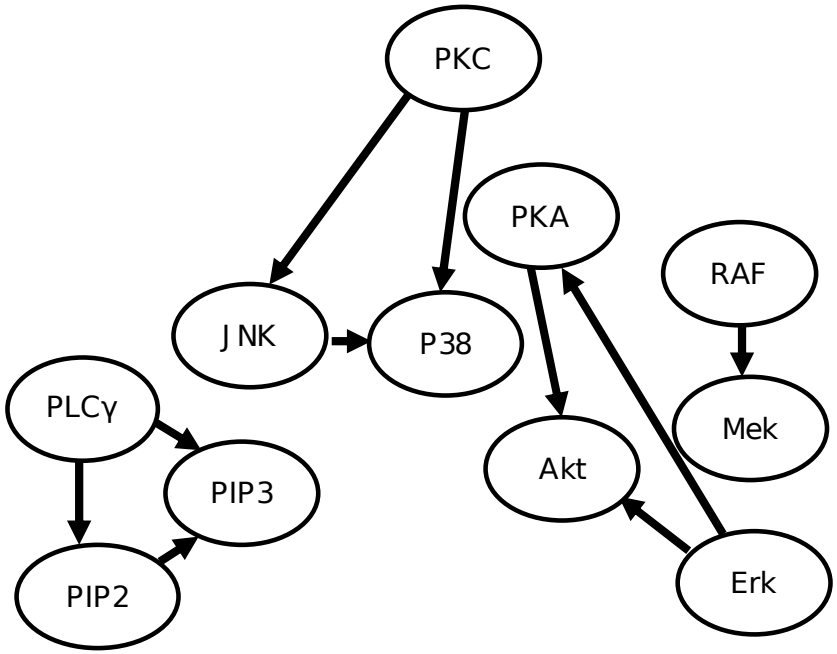
Protein Signalling in Flow Cytometry Data

Sachs et al. [13] used Bayesian networks as a tool for identifying causal influences in cellular signalling networks from simultaneous measurement of 11 phosphorylated proteins and phospholipids across single cells.

Significant edges were selected using model averaging but with an ad-hoc significance threshold of 0.85, first on 854 non-perturbed observations and then on several sets of perturbed data. This combination cannot be analysed with our approach, because each subset of the data follows a different probability distribution and therefore there is no single "true" network $\mathcal{G}_0$; therefore we limit ourselves to the unperturbed data.

Protein Signalling in Flow Cytometry Data





Again all edges identified as significant in the observational data are also identified by the proposed approach; directionality of the edges is also revealed, unlike the original network, and agrees with with the network learned with the help of perturbed data in Sachs et al. [13].

Conclusions

Conclusions

- Model validation is often performed using an ad-hoc thresholds for the identification of significant edges. Such ad-hoc approaches can have a pronounced effect on the resulting networks and biological conclusions.
- The minimisation of the L_1 norm of the difference between the CDF of the observed confidence levels and the CDF their ideal/asymptotic configuration provides straightforward and statistically-motivated approach for identifying significant edges.
- The proposed approach is defined in a very general setting and can be applied to many classes of graphical models learned from any kind of data.
- The effectiveness of the proposed approach is demonstrated on two different gene networks different studies.

Thanks!

References

References I


E. Castillo, J. M. Gutiérrez, and A. S. Hadi.
Expert Systems and Probabilistic Network Models.
Springer, 1997.


D. I. Edwards.
Introduction to Graphical Modelling.
Springer, 2nd edition, 2000.


G. Elidan.
Bayesian Network Repository, 2001.


N. Friedman, M. Goldszmidt, and A. Wyner.
Data Analysis with Bayesian Networks: A Bootstrap Approach.
In *Proceedings of the 15th Annual Conference on Uncertainty in Artificial Intelligence (UAI-99)*, pages 206 – 215. Morgan Kaufmann, 1999.


S. Imoto, S. Y. Kim, H. Shimodaira, S. Aburatani, K. Tashiro, S. Kuhara, and S. Miyano.
Bootstrap Analysis of Gene Networks Based on Bayesian Networks and Nonparametric Regression.
Genome Informatics, 13:369–370, 2002.

References II


D. Jungnickel.
Graphs, Networks and Algorithms.
Springer-Verlag, 3rd edition, 2008.


D. Koller and N. Friedman.
Probabilistic Graphical Models: Principles and Techniques.
MIT Press, 2009.


K. Korb and A. Nicholson.
Bayesian Artificial Intelligence.
Chapman & Hall, 2nd edition, 2010.


P. Murphy and D. Aha.
UCI Machine Learning Repository, 1995.


R. Nagarajan, S. Datta, M. Scutari, M. L. Beggs, G. T. Nolen, and C. A. Peterson.
Functional Relationships Between Genes Associated with Differentiation Potential of Aged Myogenic Progenitors.
Frontiers in Physiology, 1(21):1–8, 2010.

References III

J. Nocedal and S. J. Wright.
Numerical Optimization.
Springer-Verlag, 1999.

R Development Core Team.
R: A Language and Environment for Statistical Computing.
R Foundation for Statistical Computing, Vienna, Austria, 2010.

K. Sachs, O. Perez, D. Pe'er, D. A. Lauffenburger, and G. P. Nolan.
Causal Protein-Signaling Networks Derived from Multiparameter Single-Cell Data.
Science, 308(5721):523–529, 2005.

J. Schäfer and K. Strimmer.
An Empirical Bayes Approach to Inferring Large-Scale Gene Association Networks.
Bioinformatics, 21:754–764, 2004.

M. Scutari.
Learning Bayesian Networks with the bnlearn R Package.
Journal of Statistical Software, 35(3):1–22, 2010.

References IV

M. Scutari.
bnlearn: Bayesian Network Structure Learning, 2011.
R package version 2.4.

R. Steuer.
On the Analysis and Interpretation of Correlations in Metabolomic Data.
Briefings in Bioinformatics, 7(2):151–158, 2006.

I. Tsamardinos, L. E. Brown, and C. F. Aliferis.
The Max-Min Hill-Climbing Bayesian Network Structure Learning Algorithm.
Machine Learning, 65(1):31–78, 2006.

J. Whittaker.
Graphical Models in Applied Multivariate Statistics.
Wiley, 1990.