Graphical Models

and Protein Signalling Networks

Marco Scutari

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

November 5, 2012

Graphical Models

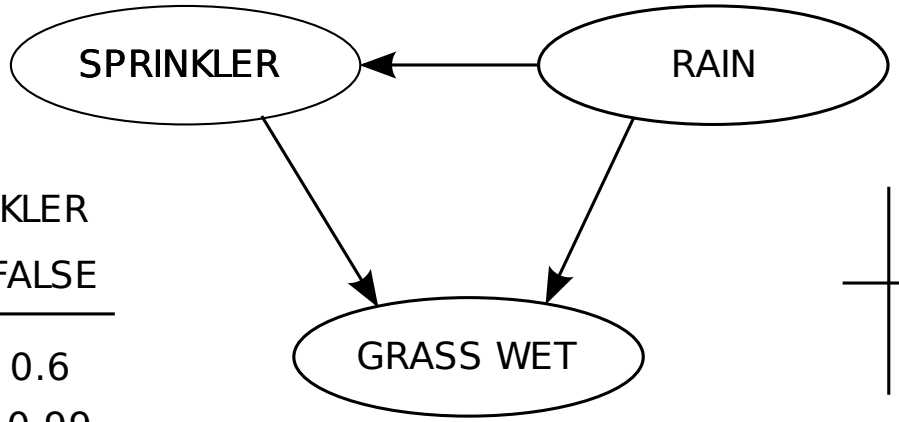
Graphical Models

Graphical models are defined by:

- a network structure, $\mathcal{G} = (\mathbf{V}, E)$, either an undirected graph (Markov networks, gene association networks, correlation networks, etc.) or a directed graph (Bayesian networks). Each node $v_i \in \mathbf{V}$ corresponds to a random variable X_i;
- a global probability distribution, $\mathbf{X}$, which can be factorised into a small set of local probability distributions according to the edges $e_{ij} \in E$ present in the graph.

This combination allows a compact representation of the joint distribution of large numbers of random variables and simplifies inference on the resulting parameter space.

A Simple Bayesian Network: Watson's Lawn



RAIN	SPRINKLER	
	TRUE	FALSE
FALSE	0.4	0.6
TRUE	0.01	0.99

RAIN	
TRUE	FALSE
0.2	0.8

SPRINKLER	RAIN	GRASS WET	
		TRUE	FALSE
FALSE	FALSE	0.0	1.0
FALSE	TRUE	0.8	0.2
TRUE	FALSE	0.9	0.1
TRUE	TRUE	0.99	0.01

Graphical Separation and Independence

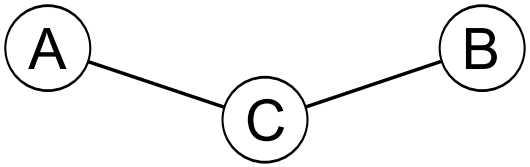
The main role of the graph structure is to express the conditional independence relationships among the variables in the model, thus specifying the factorisation of the global distribution. Different classes of graphs express these relationships with different semantics, which have in common the principle that graphical separation of two (sets of) nodes implies the conditional independence of the corresponding (sets of) random variables.

For networks considered here, separation is defined as:

- (u-)separation in Markov networks;
- d-separation in Bayesian networks.

Graphical Separation

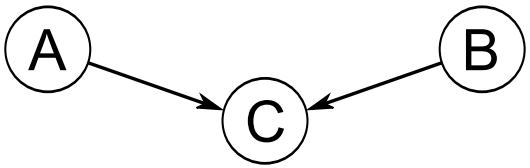
separation (undirected graphs)



$$\mathbf{A} \perp\!\!\!\perp \mathbf{B} \,|\, \mathbf{C}$$

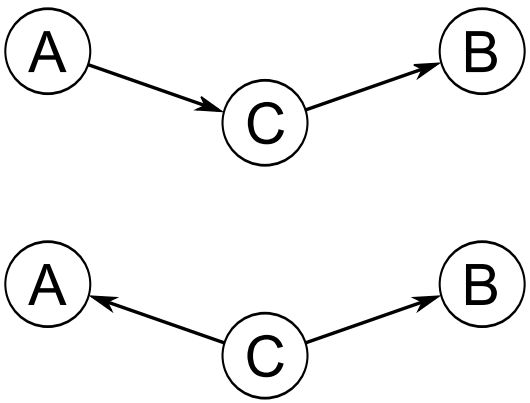
$$\mathrm{P}(\mathbf{A}, \mathbf{B}, \mathbf{C}) = \mathrm{P}(\mathbf{A} \,|\, \mathbf{C})\, \mathrm{P}(\mathbf{B} \,|\, \mathbf{C})\, \mathrm{P}(\mathbf{C})$$

d-separation (directed acyclic graphs)



$$\mathbf{A} \not\perp\!\!\!\perp \mathbf{B} \,|\, \mathbf{C}$$

$$\mathrm{P}(\mathbf{A}, \mathbf{B}, \mathbf{C}) = \mathrm{P}(\mathbf{C} \,|\, \mathbf{A}, \mathbf{B})\, \mathrm{P}(\mathbf{A})\, \mathrm{P}(\mathbf{B})$$



$$\mathbf{A} \perp\!\!\!\perp \mathbf{B} \,|\, \mathbf{C}$$

$$\begin{aligned} \mathrm{P}(\mathbf{A}, \mathbf{B}, \mathbf{C}) &= \\ &= \mathrm{P}(\mathbf{B} \,|\, \mathbf{C})\, \mathrm{P}(\mathbf{C} \,|\, \mathbf{A})\, \mathrm{P}(\mathbf{A}) \\ &= \mathrm{P}(\mathbf{A} \,|\, \mathbf{C})\, \mathrm{P}(\mathbf{B} \,|\, \mathbf{C})\, \mathrm{P}(\mathbf{C}) \end{aligned}$$

Maps and Independence

A graph $\mathcal{G}$ is a dependency map (or D-map) of the probabilistic dependence structure P of $\mathbf{X}$ if there is a one-to-one correspondence between the random variables in $\mathbf{X}$ and the nodes $\mathbf{V}$ of $\mathcal{G}$, such that for all disjoint subsets $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ of $\mathbf{X}$ we have

$$\mathbf{A} \perp\!\!\!\perp_P \mathbf{B} \,|\, \mathbf{C} \Longrightarrow \mathbf{A} \perp\!\!\!\perp_G \mathbf{B} \,|\, \mathbf{C}.$$

Similarly, $\mathcal{G}$ is an independency map (or I-map) of P if

$$\mathbf{A} \perp\!\!\!\perp_P \mathbf{B} \,|\, \mathbf{C} \Longleftarrow \mathbf{A} \perp\!\!\!\perp_G \mathbf{B} \,|\, \mathbf{C}.$$

$\mathcal{G}$ is said to be a perfect map of P if it is both a D-map and an I-map, that is

$$\mathbf{A} \perp\!\!\!\perp_P \mathbf{B} \,|\, \mathbf{C} \Longleftrightarrow \mathbf{A} \perp\!\!\!\perp_G \mathbf{B} \,|\, \mathbf{C},$$

and in this case P is said to be isomorphic to $\mathcal{G}$.

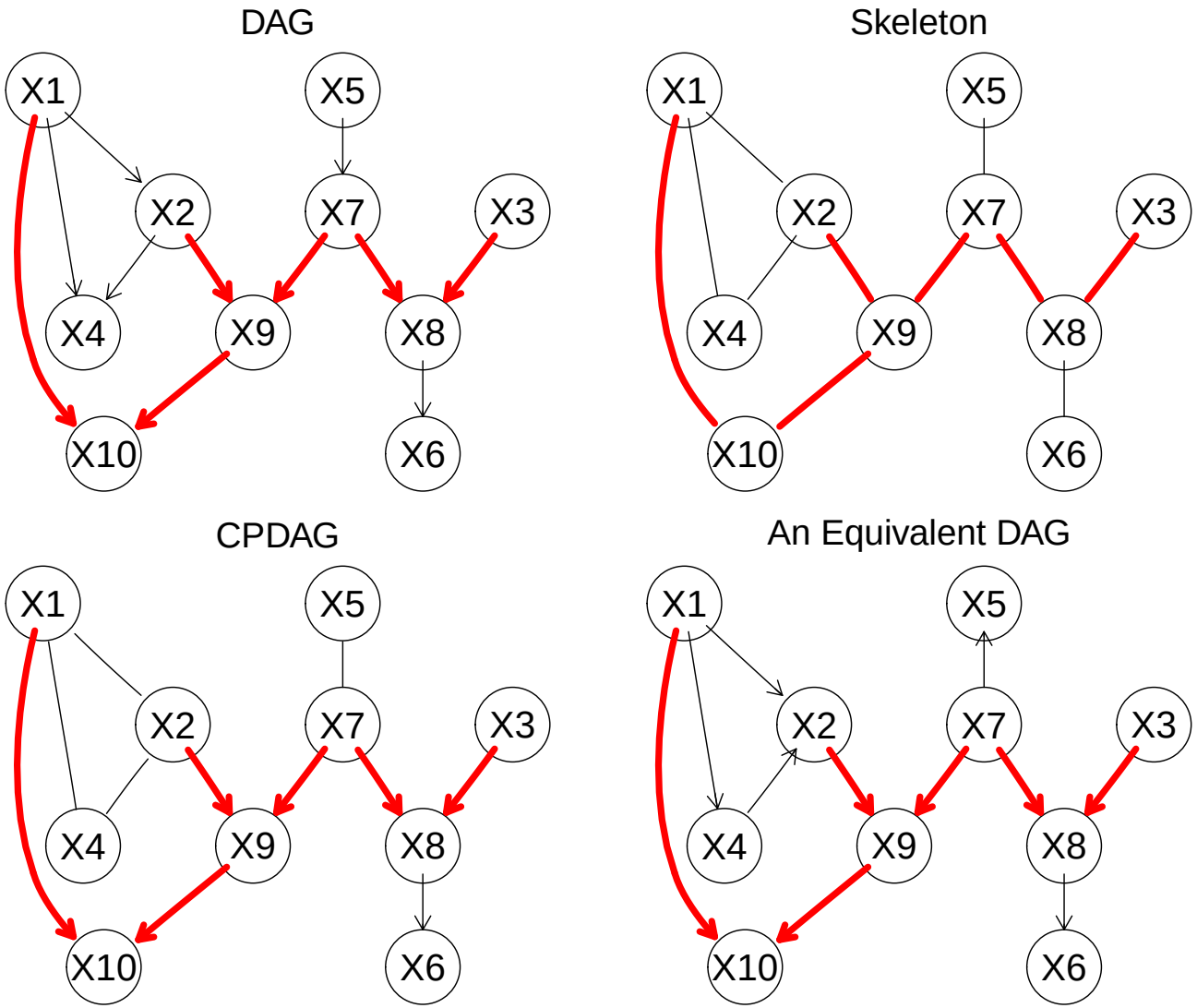
Graphical models are formally defined as I-maps under the respective definitions of graphical separation.

Bayesian Networks, Equivalence Classes and Moral Graphs

Following the definitions given in the previous couple of slides, the graph associated with a Bayesian network has three useful transforms:

- the skeleton: the undirected graph underlying a Bayesian network, i.e. the graph we get if we disregard edges' direction.
- the equivalence class: the graph (CPDAG) in which only edges which are part of a v-structure (i.e. $A \rightarrow C \leftarrow B$) and/or might result in one are directed. All valid combinations of the other edges' directions result in networks representing the same dependence structure P.
- the moral graph: the graph obtained by disregarding edges' direction and joining the two parents in each v-structure with an edge. This is essentially a way to transform a Bayesian network into a Markov network.

Skeletons and Equivalence Classes



Factorisation into Local Distributions

The most important consequence of defining graphical models as I-maps is the factorisation of the global distribution into local distributions:

- in Markov networks, local distributions are associated with the cliques C_i (maximal subsets of nodes in which each element is adjacent to all the others) in the graph,

$$\mathrm{P}(\mathbf{X}) = \prod_{i=1}^{k} \psi_i(\mathbf{C}_i),$$

  and the ψ_k functions are called potentials.

- in Bayesian networks, each local distribution is associated with a single node X_i and depends only on the joint distribution of its parents Π_{X_i}:

$$\mathrm{P}(\mathbf{X}) = \prod_{i=1}^{p} \mathrm{P}(X_i \,|\, \Pi_{X_i})$$

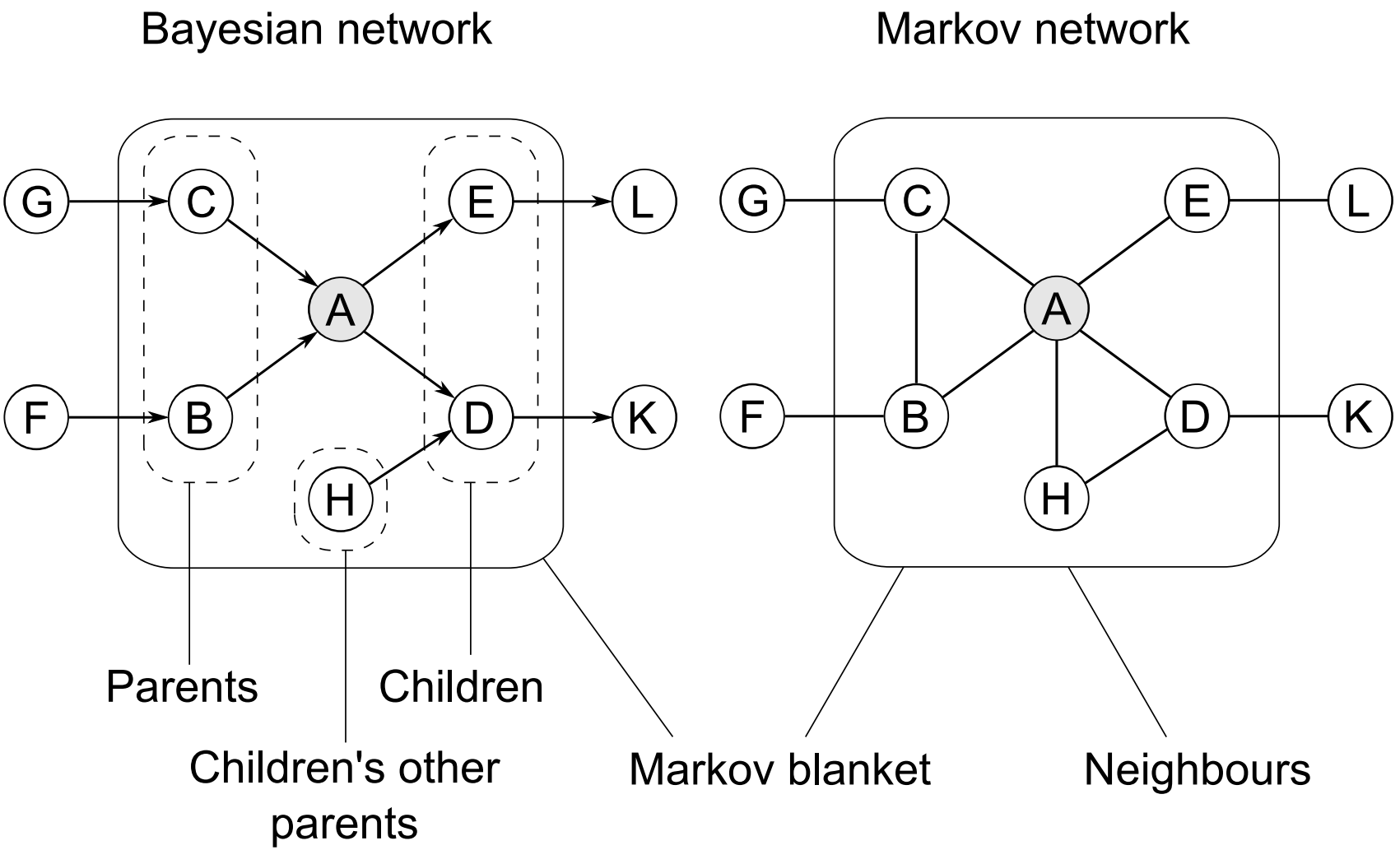
Neighbourhoods and Markov Blankets

Furthermore, for each node X_i two sets are defined:

- the neighbourhood, the set of nodes that are adjacent to X_i. These nodes cannot be made independent from X_i.
- the Markov blanket, the set of nodes that completely separates X_i from the rest of the graph. Generally speaking, it is the set of nodes that includes all the knowledge needed to do inference on X_i, from estimation to hypothesis testing to prediction, because all the other nodes are conditionally independent from X_i given its Markov blanket.

These sets are related in Markov and Bayesian networks; in particular, Markov blankets can be shown to be the same using a moral graph.

Neighbourhoods and Markov Blankets



Probability Distributions: Discrete and Continuous

Data used in graphical modelling should respect the following assumptions:

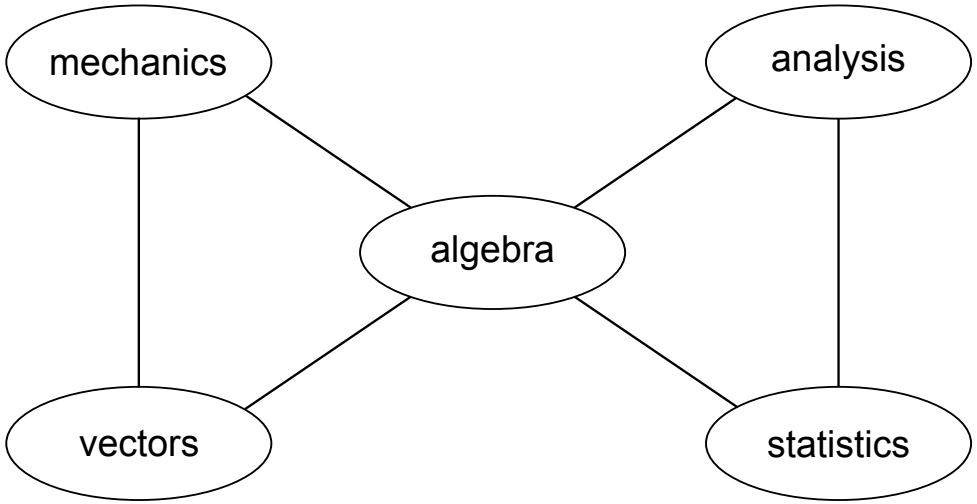
- if all the variables X_i are discrete, both the global and the local distributions are assumed to be multinomial. Local distributions are described using conditional probability tables;
- if all the variables X_i are continuous, the global distribution is assumed to be a multivariate Gaussian distribution, and the local distributions are univariate or multivariate Gaussian distributions. Local distributions are described using partial correlation coefficients;
- if both continuous and discrete variables are present, we can assume a mixture or conditional Gaussian distribution, discretise continuous attributes or use a nonparametric approach.

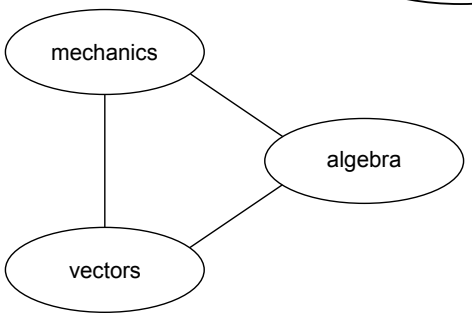
Other Distributional Assumptions

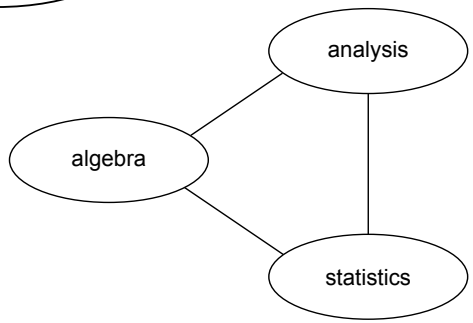
Other fundamental distributional assumptions are:

- observations must be independent. If some form of temporal or spatial dependence is present, it must be specifically accounted for in the definition of the network (as in *dynamic Bayesian networks*);
- if the model will be used as a causal graphical model, that is, to infer cause-effect relationship from experimental or (more frequently) observational data, there must be no latent or hidden variables that influence the dependence structure of the model;
- all the relationships between the variables in the network must be conditional independencies, because they are by definition the only ones that can be expressed by graphical models.

A Gaussian Markov Network (MARKS)



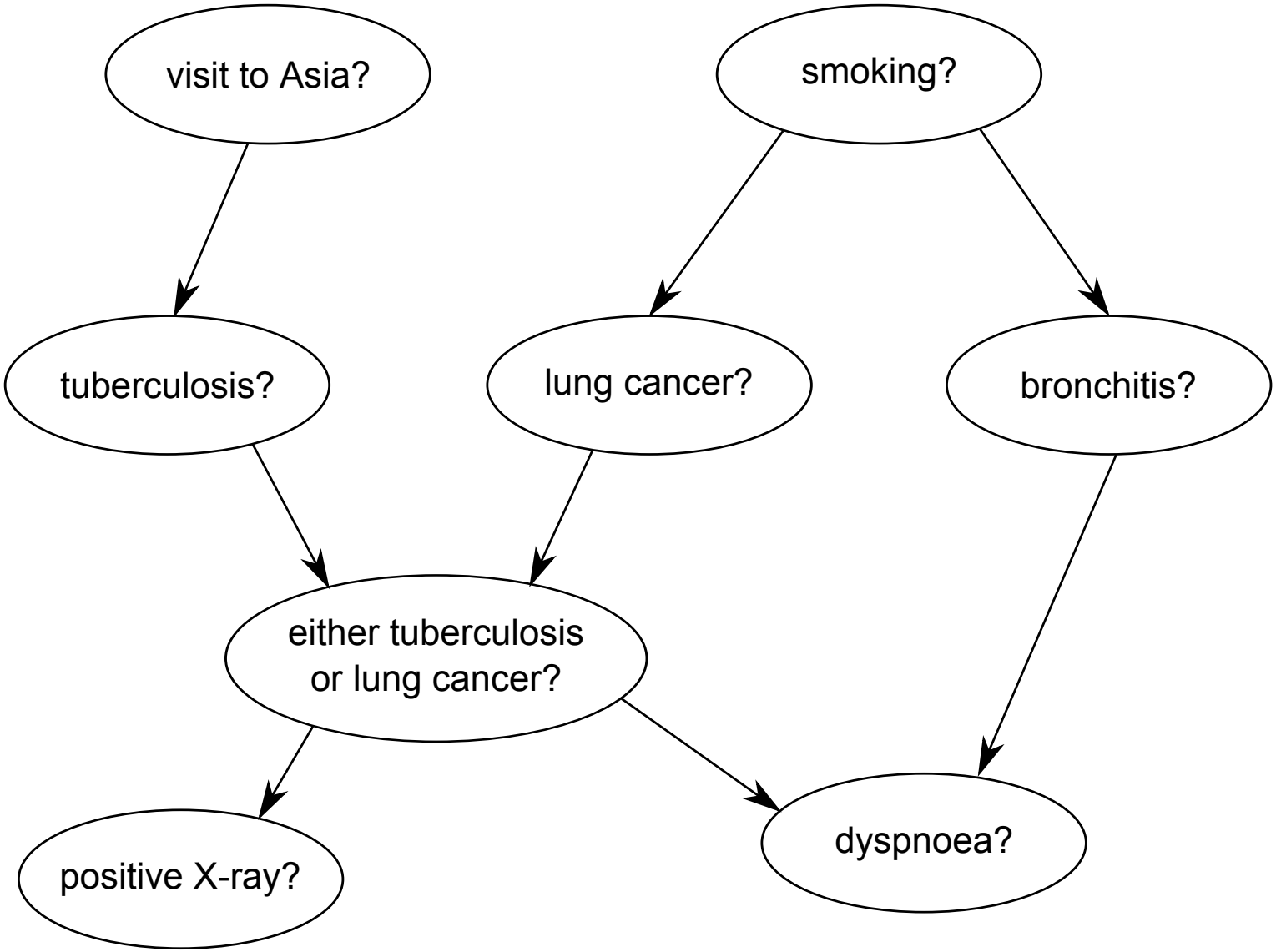




$$\begin{bmatrix} 1.000 & 0.332 & 0.235 \\ 0.332 & 1.000 & 0.327 \\ 0.235 & 0.327 & 1.000 \end{bmatrix}$$

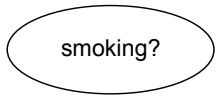
$$\begin{bmatrix} 1.000 & 0.451 & 0.364 \\ 0.451 & 1.000 & 0.256 \\ 0.364 & 0.256 & 1.000 \end{bmatrix}$$

A Discrete Bayesian Network (ASIA)

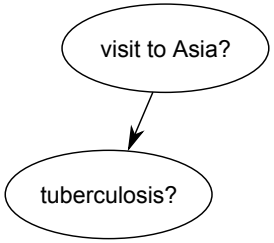


A Discrete Bayesian Network (ASIA)

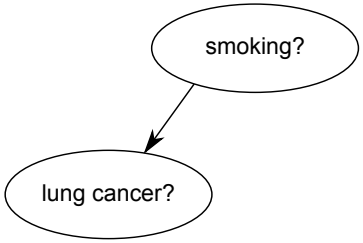
yes	no
0.01	0.99



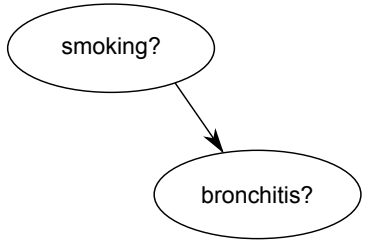
yes	no
0.50	0.50



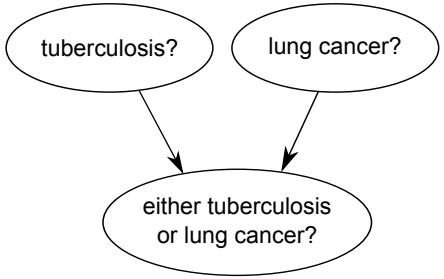
	yes	no
yes	0.05	0.95
no	0.01	0.99



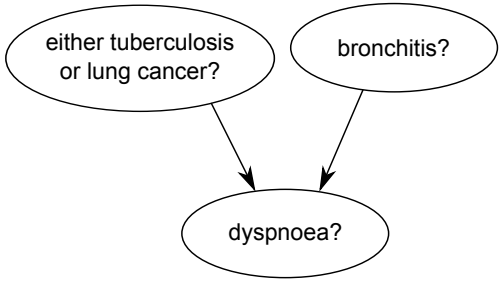
	yes	no
yes	0.10	0.90
no	0.01	0.99



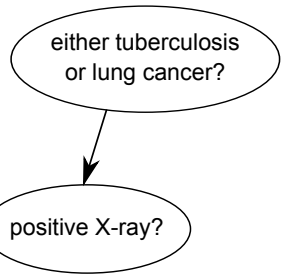
	yes	no
yes	0.60	0.40
no	0.30	0.70



	yes	no
yes:yes	1	0
yes:no	1	0
no:yes	1	0
no:no	0	1



	yes	no
yes:yes	0.90	0.10
yes:no	0.70	0.30
no:yes	0.80	0.20
no:no	0.10	0.90



	yes	no
yes	0.98	0.02
no	0.05	0.95

Limitations of These Probability Distribution

- no real-world, multivariate data set follows a multivariate Gaussian distribution; even if the marginal distributions are normal, not all dependence relationships are linear.
- computing partial correlations is problematic in most large data sets (and in a lot of small ones, too).
- parametric assumptions for mixed data have strong limitations, as they impose constraints on which edges may be present in the graph (e.g. a continuous node cannot be the parent of a discrete node).
- discretisation is a common solution to the above problems, but it discards useful information and it is tricky to get right (i.e. choosing a set of intervals such that the dependence relationships involving the original variable are preserved).
- ordered categorical variables are treated as unordered, again losing information.

Graphical Model Learning

Learning a Graphical Model

Model selection and estimation are collectively known as learning, and are usually performed as a two-step process:

1. structure learning, learning the graph structure from the data.
2. parameter learning, learning the local distributions implied by the graph structure learned in the previous step.

This workflow is implicitly Bayesian; given a data set $\mathcal{D}$ and if we denote the parameters of the global distribution as $\mathbf{X}$ with Θ, we have

$$\underbrace{\mathrm{P}(\mathcal{M} \mid \mathcal{D}) = \mathrm{P}(\mathcal{G}, \Theta \mid \mathcal{D})}_{learning} = \underbrace{\mathrm{P}(\mathcal{G} \mid \mathcal{D})}_{structure\,learning} \cdot \underbrace{\mathrm{P}(\Theta \mid \mathcal{G}, \mathcal{D})}_{parameter\,learning}$$

and structure learning is done in practice as

$$\mathrm{P}(\mathcal{G} \mid \mathcal{D}) \propto \mathrm{P}(\mathcal{G})\,\mathrm{P}(\mathcal{D} \mid \mathcal{G}) = \mathrm{P}(\mathcal{G}) \int \mathrm{P}(\mathcal{D} \mid \mathcal{G}, \Theta)\,\mathrm{P}(\Theta \mid \mathcal{G}) d\Theta.$$

Local Distributions: Divide and Conquer

Most tasks related to both learning and inference are NP-hard (they cannot be solved in polynomial time in the number of variables). They are still feasible thanks to the decomposition of $\mathbf{X}$ into the local distributions; under some assumptions (parameter independence) there is never the need to manipulate more than one of them at a time.

In Bayesian networks, for example, structure learning boils down to

$$\begin{aligned} \mathrm{P}(\mathcal{D} \,|\, \mathcal{G}) &= \int \prod \left[\mathrm{P}(X_i \,|\, \Pi_{X_i}, \Theta_{X_i}) \, \mathrm{P}(\Theta_{X_i} \,|\, \Pi_{X_i}) \right] d\Theta \\ &= \prod \left[\int \mathrm{P}(X_i \,|\, \Pi_{X_i}, \Theta_{X_i}) \, \mathrm{P}(\Theta_{X_i} \,|\, \Pi_{X_i}) d\Theta_{X_i} \right] \end{aligned}$$

and parameter learning boils down to

$$\mathrm{P}(\Theta \,|\, \mathcal{G}, \mathcal{D}) = \prod \mathrm{P}(\Theta_{X_i} \,|\, \Pi_{X_i}, \mathcal{D}).$$

Structure Learning

The Big Three: Constraint-based, Score-based and Hybrid

Despite the (sometimes confusing) variety of theoretical backgrounds and terminology they can all be traced to only three approaches:

- constraint-based algorithms: they use statistical tests to learn conditional independence relationships (called *constraints* in this setting) from the data and assume that the graph underlying the probability distribution is a perfect map to determine the correct network structure.
- score-based algorithms: each candidate network is assigned a score reflecting its goodness of fit, which is then taken as an objective function to maximise.
- hybrid algorithms: conditional independence tests are used to learn at least part of the conditional independence relationships from the data, thus restricting the search space for a subsequent score-based search. The latter determines which edges are actually present in the graph and, in the case of Bayesian networks, their direction.

Constraint-based Structure Learning Algorithms

The mapping between edges and conditional independence relationships lies at the core of graphical modelling; therefore, one way to learn the structure of a graphical model is to check which ones of such relationships hold according to a suitable conditional independence test.

Such an approach results in a set of conditional independence constraints that identify a single graph (for a Markov network) or a single equivalence class (for a Bayesian network). In the latter case, the relevant edge directions are determined using more conditional independence tests to identify which v-structures are present in the graph.

The first constraint-based algorithm was pioneered by Verma & Pearl, and is names Inductive Causation. It's not usable in practice, but it provided a theoretical framework for later algorithms.

Conditional Independence Tests

Classic tests are used because they are fast but are not particularly good.

- asymptotic discrete tests: mutual information/log-likelihood ratio and Pearson's X^2 with a χ^2 distribution.
- asymptotic continuous tests: Fisher's Z, with a $N(0, 1)$ distribution, and mutual information/log-likelihood ratio, with a χ^2 distribution.
- exact continuous tests: t test with a Student's t distribution.

Better alternatives are:

- permutation tests: all of the above, evaluated using the permutation distribution as the null distribution. The resulting structure is better for goodness-of-fit and prediction.
- shrinkage tests: log-likelihood ratio tests can be reworked as shrinkage tests whose behaviour is determined by a regularisation parameter λ. The resulting structure is closer to the "real" one and is therefore better for causal reasoning.

Other Constraint-based algorithms

- Peter & Clark (PC): a true-to-form implementation of the Inductive Causation algorithm, specifying only the order of the conditional independence tests. Starts from a saturated network and performs tests gradually increasing the number of conditioning nodes.
- Grow-Shrink (GS) and Incremental Association (IAMB) variants: these algorithms learn the Markov blanket of each node to reduce the number of tests required by the Inductive Causation algorithm. Markov blankets are learned using different forward and step-wise approaches; the initial network is assumed to be empty (i.e. not to have any edge).
- Max-Min Parents & Children (MMPC): uses a minimax approach to avoid conditional independence tests known *a priori* to accept the null hypothesis of independence.

Pros & Cons of Constraint-based Algorithms

- They depend heavily on the quality of the conditional independence tests they use; all proofs of correctness assume tests are always right. That's why asymptotic tests are bad, and non-regularised parametric tests are not ideal.
- They are consistent, but converge is slower than score-based and hybrid algorithms.
- At any single time they evaluate a small subset of variables, which makes them very memory efficient.
- They do not require multiple testing adjustment in most cases.
- They are embarrassingly parallel, so they scale extremely well.

Score-based Structure Learning Algorithms

The dimensionality of the space of graph structures makes an exhaustive search unfeasible in practice, regardless of the goodness-of-fit measure (called network score) used in the process. However, heuristics can still be used in conjunction with decomposable scores, i.e.

$$\text{Score}(\mathcal{G}) = \sum \text{Score}(X_i \,|\, \Pi_{X_i})$$

such as

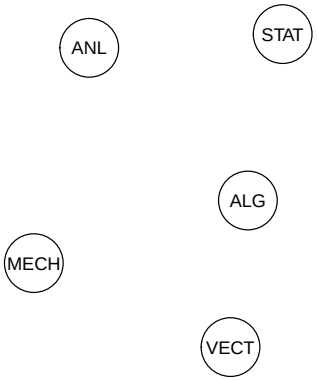
$$\text{BIC}(\mathcal{G}) = \sum \log \text{P}(X_i \,|\, \Pi_{X_i}) - \frac{|\Theta_{X_i}|}{2} \log n$$

$$\text{BDe}(\mathcal{G}), \text{BGe}(\mathcal{G}) = \sum \log \left[\int \text{P}(X_i \,|\, \Pi_{X_i}, \Theta_{X_i}) \, \text{P}(\Theta_{X_i} \,|\, \Pi_{X_i}) d\Theta_{X_i} \right]$$

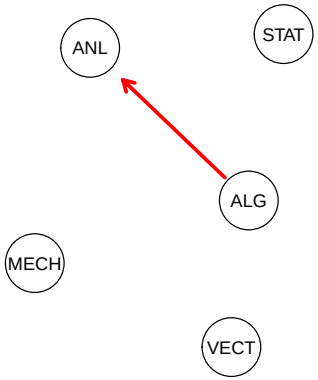
if each comparison involves structures differing in only one local distribution at a time.

The Hill-Climbing Algorithm

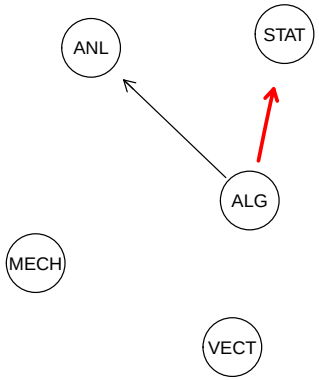
Initial BIC score: −1807.528



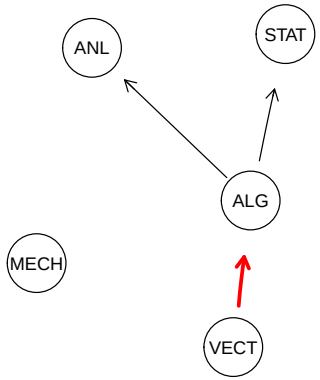
Current BIC score: −1778.804



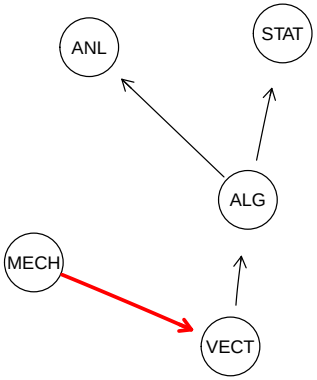
Current BIC score: −1755.383



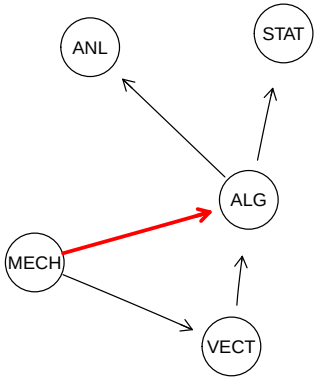
Current BIC score: −1737.176



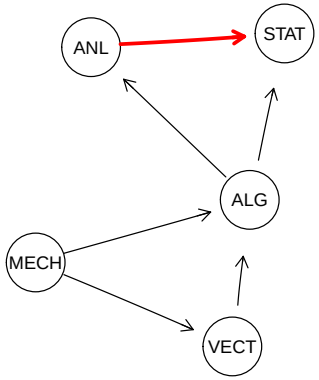
Current BIC score: −1723.325



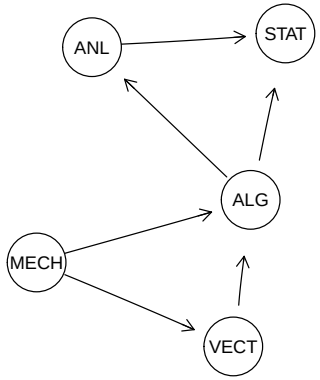
Current BIC score: −1720.901



Current BIC score: −1720.150



Final BIC score: −1720.150



Other Score-based Algorithms

- Hill-Climbing + Random Restart: performs several hill-climbing runs, perturbing the result of each one as the initial network for the next. It does get stuck in local maxima as often as plain hill-climbing.
- Greedy Equivalent Search: hill-climbing over equivalence classes rather than graph structures; the search space is much smaller.
- Tabu Search: a modified hill-climbing that keeps a list of the last k structures visited, and returns only if they are all worse than the current one.
- Genetic Algorithms: they perturb (*mutation*) and combine *crossover* features through several generations of structures, and keep the ones leading to better scores. Inspired by Darwinian evolution.
- Simulated Annealing: again similar to hill-climbing, but not looking at the maximum score improvement at each step. Very difficult to use in practice because of its tuning parameters.

Pros & Cons of Score-based Algorithms

- Convergence to the global maximum (i.e. the best structure) is not guaranteed for finite samples, the search may get stuck in a local maximum.
- They are consistent, and they converge faster than constraint-based algorithms, but this is more due to the properties of the BDe and BGe scores than the algorithms themselves.
- They require a definition of both the global and the local densities, and a matching decomposable network score.
- Most scores have tuning parameters, whereas conditional independence tests do not.

Hybrid Structure Learning Algorithms

Hybrid algorithms combine constraint-based and score-based algorithms to complement the respective strengths and weaknesses; they are considered the state of the art in current literature.

They work by alternating the following two steps:

- restrict: learn some conditional independence constraints to reduce the number of candidate networks;
- maximise: find the best network that satisfies those constraints and define a new set of constraints to improve on.

These steps can be repeated several times (until convergence), but one or two times is usually enough.

The algorithm that pioneered this approach is the Sparse Candidate by Friedman et al., and more recently Max-Min Hill-Climbing (MMHC).

Pros & Cons of Structure Learning Algorithms

- Since only the general framework is defined, it is easy to modify them to use newer constraint-based and score-based algorithms.
- You can mix and match conditional independence tests and network scores to create a learning algorithm ranging from frequentist to Bayesian to information-theoretic and anything in between (within reason).
- They are usually faster than the alternatives, and more stable.
- Tuning parameters can be difficult to tune for some configurations of algorithms, tests and scores.

Parameter Learning

The Big Three: Likelihood, Bayesian and Shrinkage

Once the structure of the model is known, the problem of estimating the parameters of the global distribution can be solved by estimating the parameters of the local distributions, one at a time.

Three common choices are:

- maximum likelihood estimators: just the usual empirical estimators. Often described as either maximum entropy or minimum divergence estimators in information-theoretic literature.
- Bayesian posterior estimators: posterior estimators, based on conjugate priors to keep computations fast, simple and in closed form.
- shrinkage estimators: regularised estimators based either on James-Stein or Bayesian shrinkage results.

Maximum Likelihood and Maximum Entropy Estimation

The classic estimators for (conditional) probabilities and (partial) correlations are a bad choice for almost all real-world problems.
They are still around because:

- they are used in benchmark simulations;
- computer scientists do not care much about parameter estimation.

However:

- maximum likelihood estimates are unstable in most multivariate problems, both discrete and continuous;
- for the multivariate Gaussian distribution, James & Stein proved in the 1950s that the maximum likelihood estimator for the mean is not admissible in 3+ dimensions;
- partial correlations are often ill-behaved because of that, even with Moore-Penrose pseudo-inverses;
- maximum likelihood estimates are non-smooth and create problems when using the graphical model for inference.

Maximum a Posteriori Bayesian Estimation

Bayesian posterior estimates are the sensible choice for parameter estimation according to Koller's & Friedman's tome on graphical models. Choices for the priors are limited (for computational reasons) to conjugate distributions, namely:

- the Dirichlet for discrete models, i.e.

$$Dir(\alpha_{k \mid \Pi_{X_i}=\pi}) \quad \xrightarrow{data} \quad Dir(\alpha_{k \mid \Pi_{X_i}=\pi} + n_{k \mid \Pi_{X_i}=\pi})$$

  meaning that $\hat{p}_{k \mid \Pi_{X_i}=\pi} = \alpha_{k \mid \Pi_{X_i}=\pi} / \sum_{\pi} \alpha_{k \mid \Pi_{X_i}=\pi}$.

- the Inverse Wishart for Gaussian models, i.e.

$$IW(\Psi, m) \quad \xrightarrow{data} \quad IW(\Psi + n\Sigma, m + n).$$

In both cases (when a non-informative prior is used) the only free parameter is the equivalent or imaginary sample size, which gives the relative weight of the prior compared to the observed sample.

Model Averaging

The Big Three: Frequentist, Bayesian and Hybrid

The results of both structure and parameter learning are noisy in most real-world settings, due to limitations in the data and in our knowledge of the processes that control them. Since parameters are learned conditional on the results of structure learning, using model averaging to obtain a stable network structure from the data is an essential step in graphical modelling.

- frequentist: generating network structures using bootstrap and model averaging (aka bagging).
- Bayesian: generating network structures from the posterior $\mathrm{P}(\mathcal{G} \mid \mathcal{D})$ using exhaustive enumeration or Markov Chain Mote Carlo approximations.
- hybrid: generating network structures again using bootstrap, but weighting them with their posterior probabilities when performing model averaging.

A Frequentist Approach: Friedman's Confidence

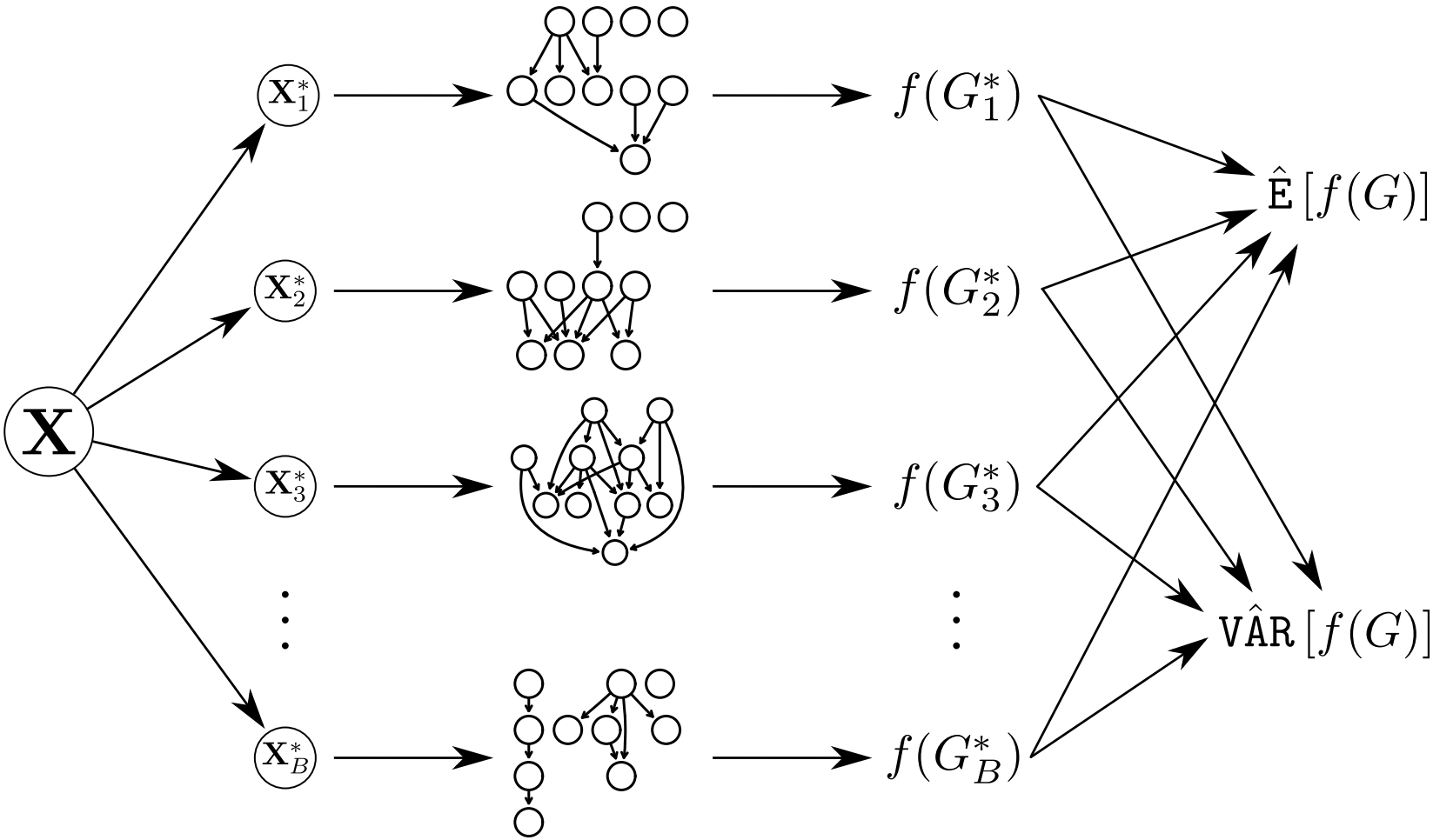
Friedman et al. proposed an approach to model validation based on bootstrap resampling and model averaging:

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 $\mathcal{G}_b = (\mathbf{V}, E_b)$ 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.

A Frequentist Approach: Friedman's Confidence



Identifying Significant Edges

- 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 principled way of identifying significant edges and the confidence threshold.

The Confidence Threshold

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

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

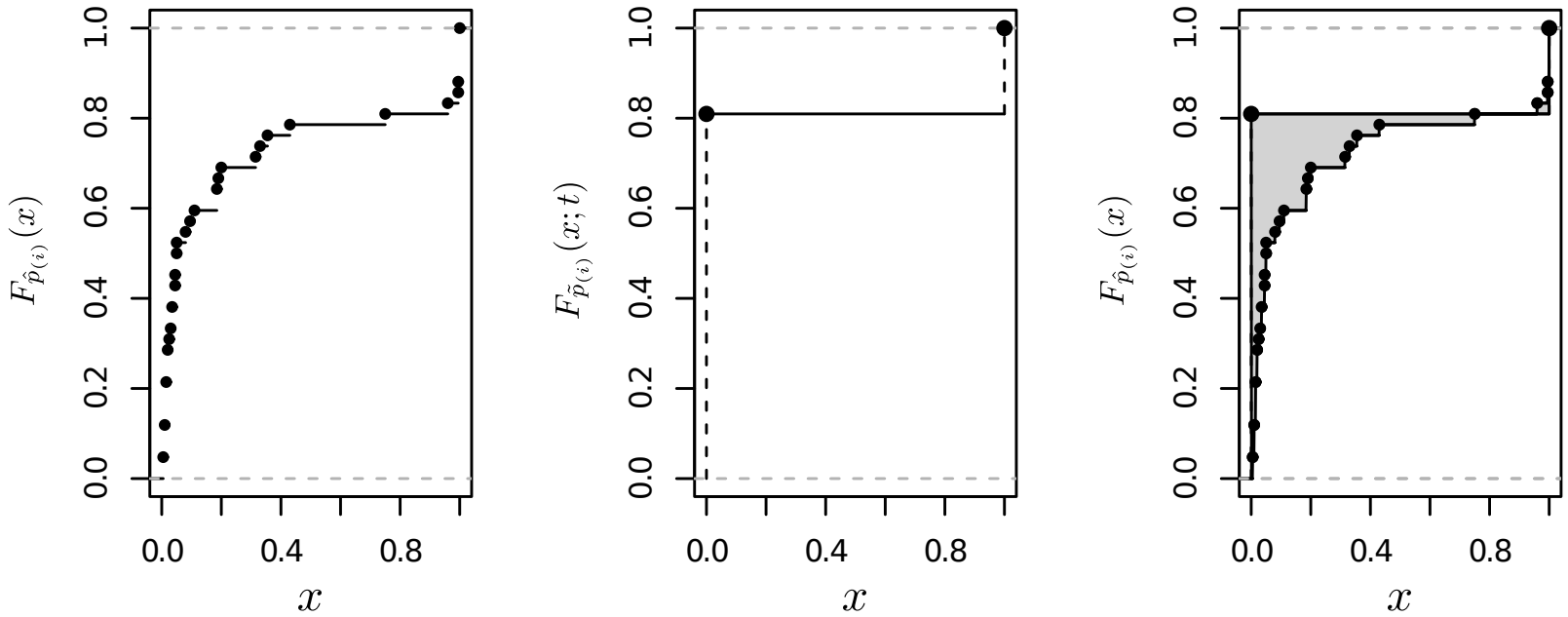
and

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

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

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

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).

Causal Protein-Signalling Networks

Source

What follows reproduces (to the best of my ability, and Karen Sachs' recollections about the implementation details that did not end up in the Methods section) the statistical analysis in the following paper:

Causal Protein-Signaling Networks Derived from Multiparameter Single-Cell Data
Karen Sachs, *et al.*
Science **308**, 523 (2005);
DOI: 10.1126/science.1105809

That's a landmark paper in applying Bayesian Networks because:

- it highlights the use of observational vs interventional data;
- results are validated using existing literature.

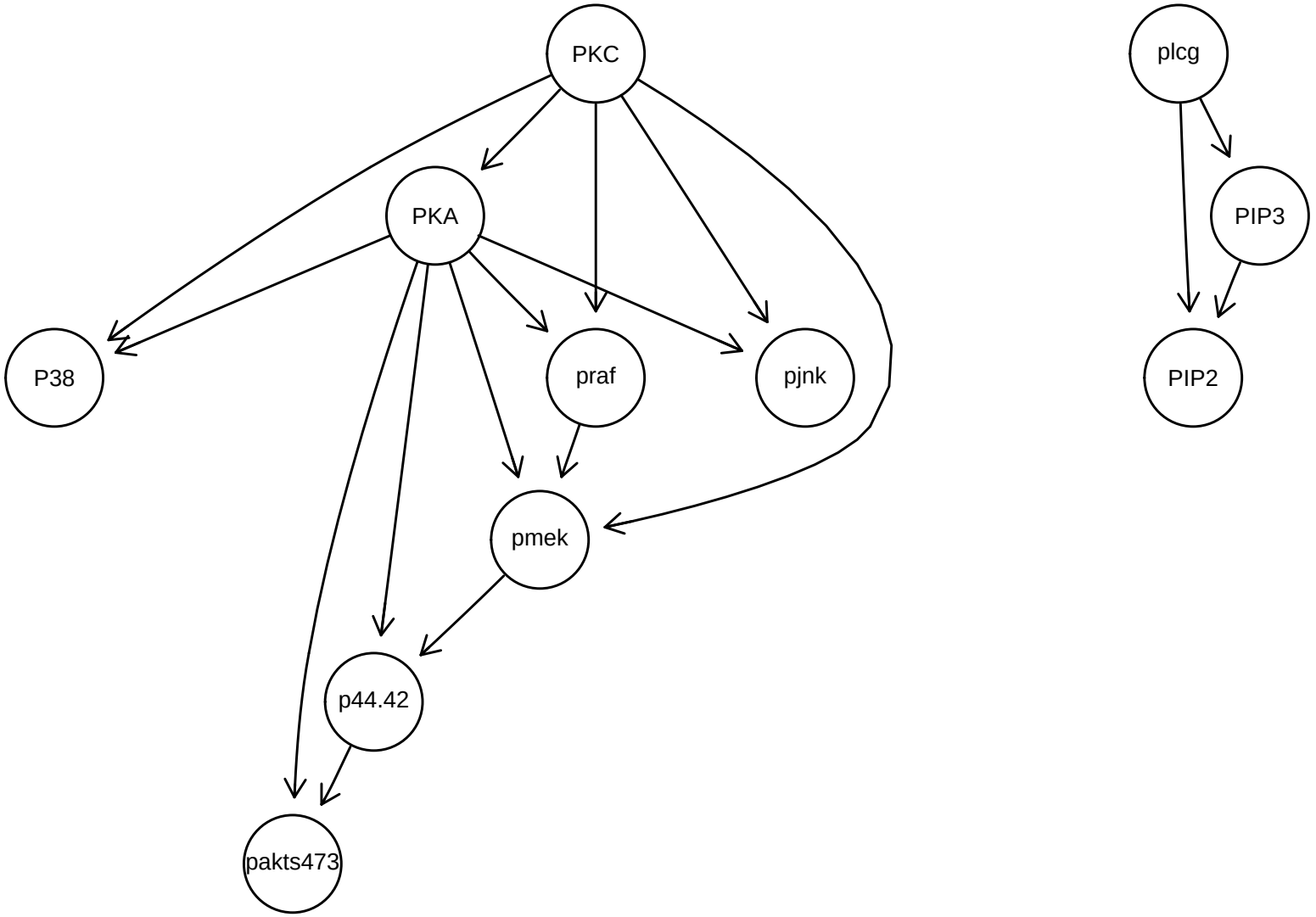
An Overview of the Data

The data consist in the simultaneous measurements of 11 phosphorylated proteins and phospholypids derived from thousands of individual primary immune system cells:

- 1800 data subject only to general stimolatory cues, so that the protein signalling paths are active;
- 600 data with with specific stimolatory/inhibitory cues for each of the following 4 proteins: pmek, PIP2, pakts473, PKA;
- 1200 data with specific cues for PKA.

Overall, the data set contains 5400 observations with no missing value.

Network Reconstructed from Literature



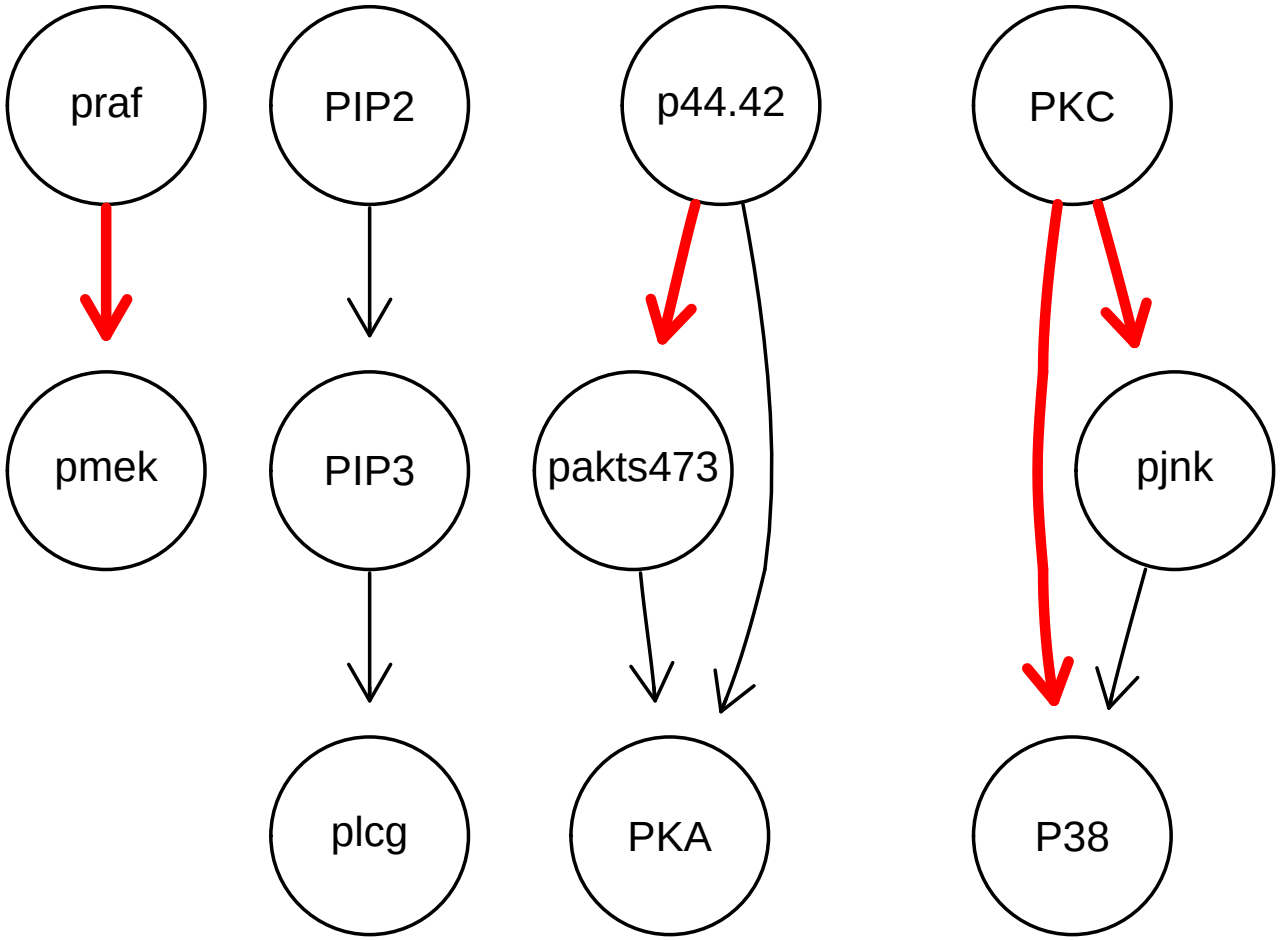
Using Only Observational Data

As a first, exploratory analysis, we can try to learn a network from the data that were subject only to general stimolatory cues. Since these cues only ensure the pathways are active, but do not tamper with them in any way, such data are observational (as opposed to interventional).

```
> library(bnlearn)
> hc(sachs, score = "bge", iss = 5)
```

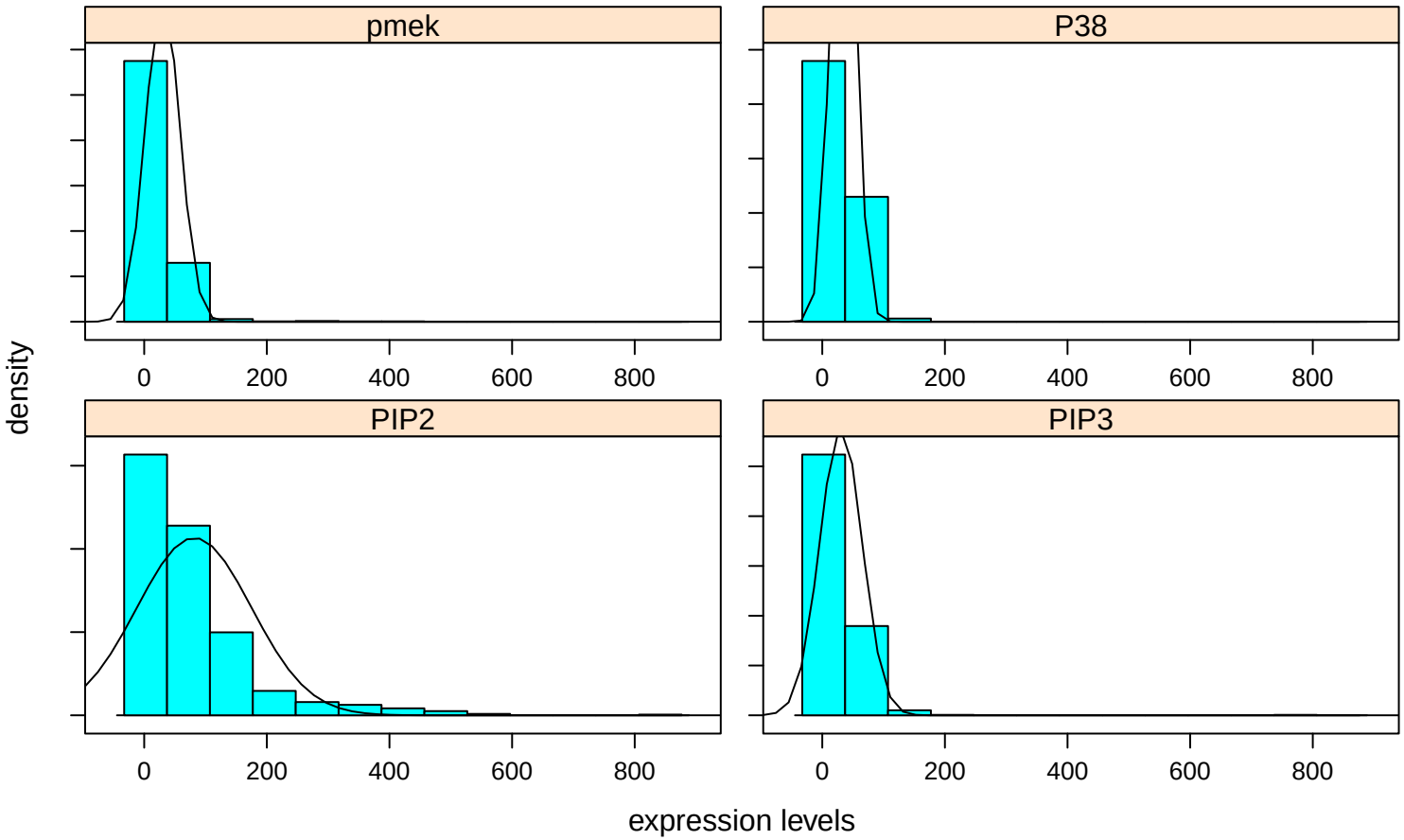
Classic algorithms in literature are not designed to handle interventional data, but work out-of-the box with observational ones.

Network Reconstructed from the Observational Data



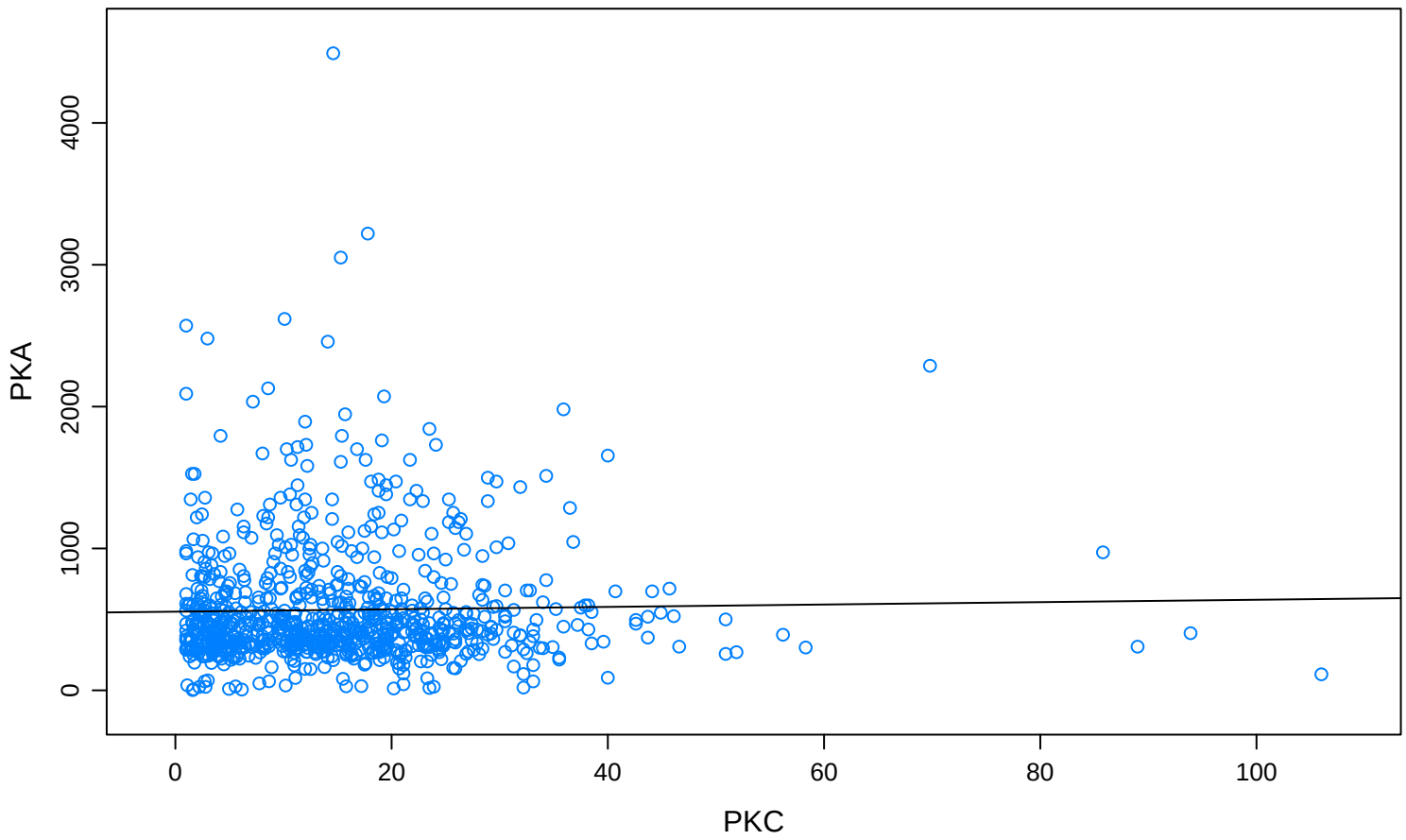
Arc highlighted in red are also present in the network reconstructed from literature.

Expression Data are not Symmetric



Therefore, assuming a Gaussian distribution is problematic.

Expression Data are not Linked by Linear Relationships



Therefore, tests for correlation are biased and have extremely low power.

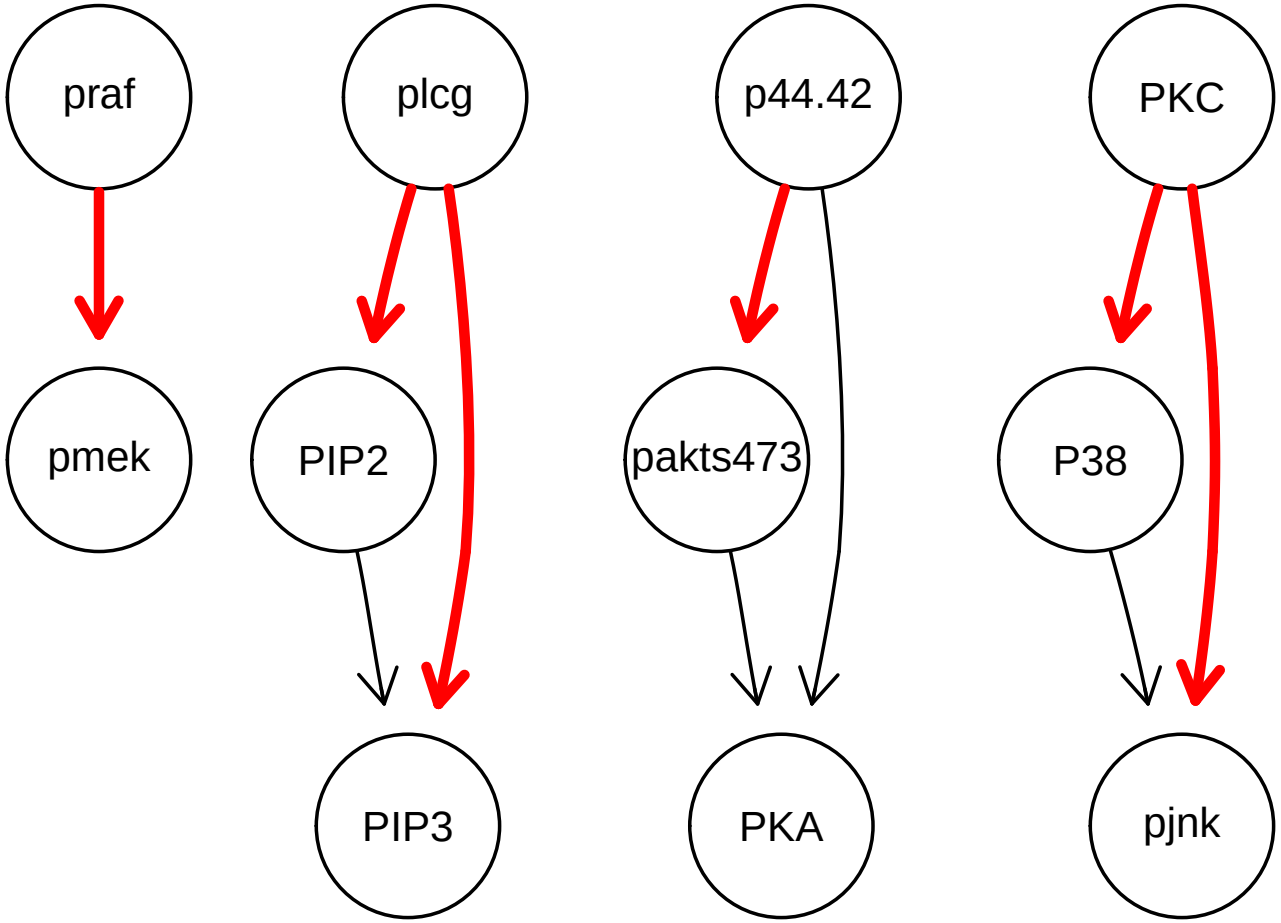
Discretize!

Since we cannot use Gaussian Bayesian networks, we can discretize them instead. Hartemink's method is designed to preserve as much as possible all pairwise dependencies, as opposed to marginal discretization methods.

```
> dsachs = discretize(sachs, method = "hartemink",
+                     breaks = 3, ibreaks = 60,
+                     idisc = "quantile")
```

Data are first marginalised in 60 intervals, which are subsequently collapsed while reducing the mutual information between the variables as little as possible. The process stops when each variable has 3 levels (i.e. low, average and high expression).

Network Reconstructed from the Discretized Data



Two more arcs are correctly identified, but most are still missing.

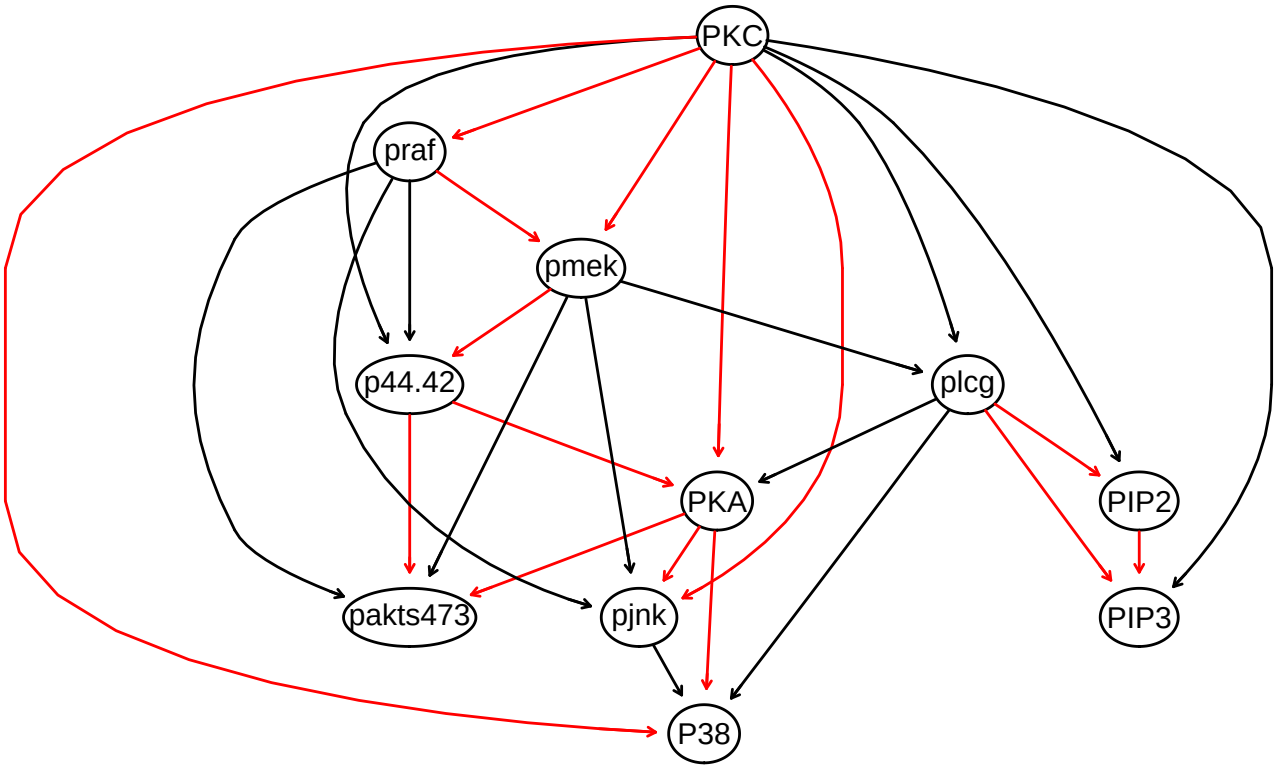
Considering Interventional Data

It is apparent from the previous networks that most signalling paths are not statistically recognisable unless we inhibit or stimulate the expression of at least some of the proteins in the network. Therefore, we include the interventional data in the analysis.

```
> INT = sapply(1:11, function(x)
+          { which(isachs$INT == x) })
> names(INT) = names(isachs)[1:11]
> hc(isachs[, 1:11], score = "mbde",
+     exp = INT, iss = 5)
```

Since the standard BDe score does not take interventions into account, we use a modified BDe score that disregards any causal influence for the proteins that have been inhibited or stimulated.

Network Reconstructed from the Interventional Data



More arcs are included, but there are many false positives.

Removing Noisy Arcs with Model Averaging

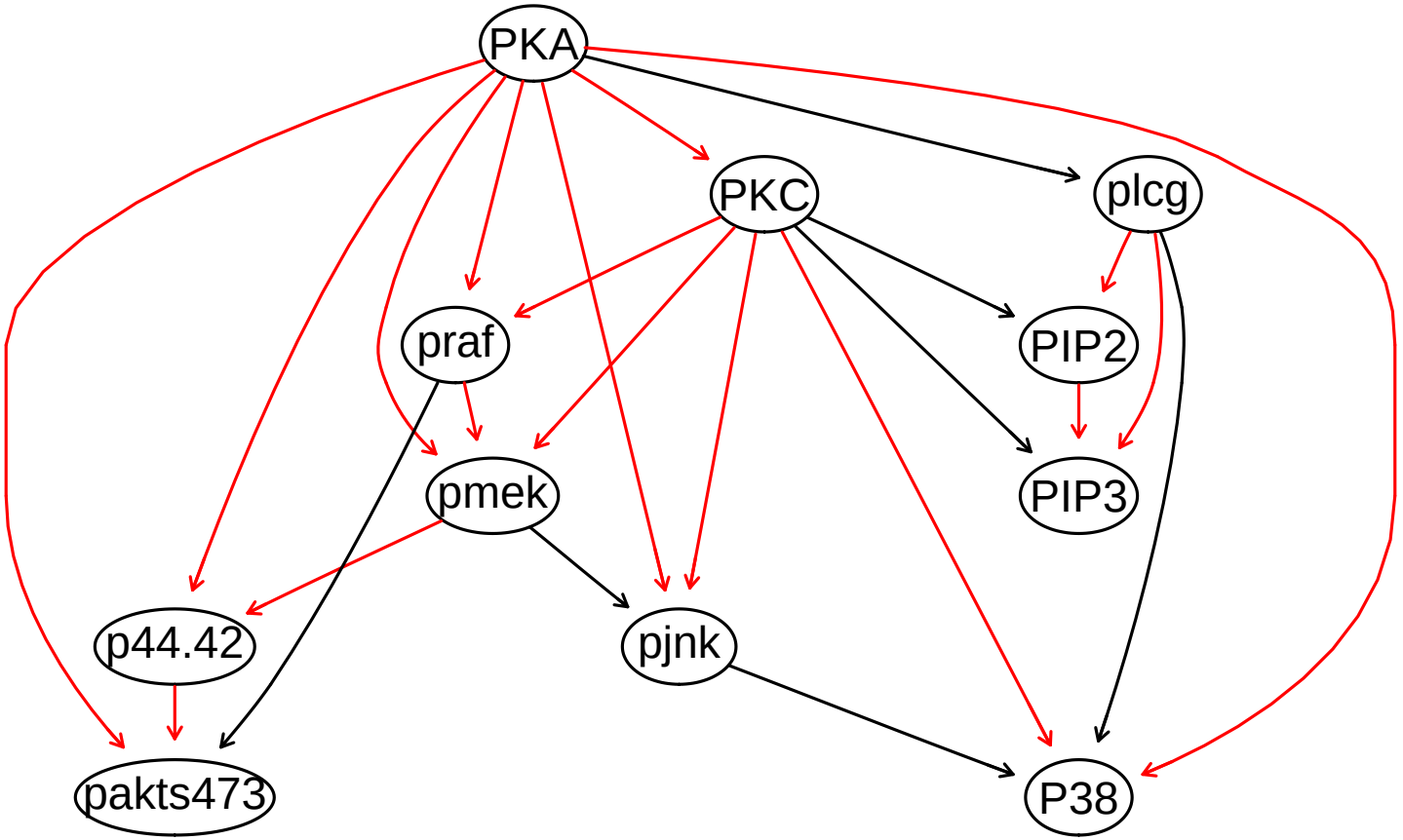
Two simple steps can be taken to remove noisy arcs:

- average multiple networks learned using different starting points for the structure learning algorithm;
- use TABU search instead of Hill-Climbing.

```
> start = random.graph(nodes = nodes,
+   method = "melancon", num = 500, burn.in = 10^5,
+   every = 100)
> netlist = lapply(start, function(net) {
+   tabu(isachs[, 1:11], score = "mbde", exp = INT,
+     iss = 10, start = net, tabu = 50) })
> arcs = custom.strength(netlist, nodes = nodes)
```

A similar approach was chosen as the best performing in Sachs *et al.* [27], with minor differences in results.

Interventional Data with Model Averaging



All the arcs supported by literature are present in the network.

Conclusions

Conclusions

- Graphical models combine many ideas from different fields to allow an intuitive manipulation of high-dimensional problems and the corresponding multivariate probability distributions.
- A sensible use of Bayesian and shrinkage techniques in structure and parameter learning allows a great deal of flexibility and results in good models.
- Properly validated graphical models can capture the dependence structure of the data even with very small sample sizes.
- The use of interventions and model averaging improves the quality of the learned networks dramatically.

Thanks!

References

References I

R. R. Bouckaert.
Bayesian Belief Networks: from Construction to Inference.
PhD thesis, Utrecht University, The Netherlands, 1995.

D. M. Chickering.
Optimal Structure Identification with Greedy Search.
Journal of Machine Learning Resesearch, 3:507–554, 2002.

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

J. Friedman, T. Hastie, and R. Tibshirani.
Sparse Inverse Covariance Estimation With the Graphical Lasso.
Biostatistics, 9:432–441, 2007.

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.

References II


N. Friedman and D. Koller.
Being Bayesian about Bayesian Network Structure: A Bayesian Approach to Structure Discovery in Bayesian Networks.
Machine Learning, 50(1–2):95–126, 2003.


N. Friedman, D. Pe'er, and I. Nachman.
Learning Bayesian Network Structure from Massive Datasets: The "Sparse Candidate" Algorithm.
In *Proceedings of 15th Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 206–221. Morgan Kaufmann, 1999.


D. Geiger and D. Heckerman.
Learning Gaussian Networks.
Technical report, Microsoft Research, Redmond, Washington, 1994.
Available as Technical Report MSR-TR-94-10.


F. Harary and E. M. Palmer.
Graphical Enumeration.
Academic Press, 1973.

References III


T. Hastie, R. Tibshirani, and J. Friedman.
The Elements of Statistical Learning: Data Mining, Inference, and Prediction.
Springer, 2nd edition, 2009.


J. Hausser and K. Strimmer.
Entropy Inference and the James-Stein Estimator, with Application to Nonlinear Gene Association Networks.
Journal of Machine Learning Resesearch, 10:1469–1484, 2009.


D. Heckerman, D. Geiger, and D. M. Chickering.
Learning Bayesian Networks: The Combination of Knowledge and Statistical Data.
Machine Learning, 20(3):197–243, September 1995.
Available as Technical Report MSR-TR-94-09.


C. J. Hoggart, J. C. Whittaker, M. De Iorio, and D. J. Balding.
Simultaneous Analysis of All SNPs in Genome-Wide and Re-Sequencing Association Studies.
PLoS Genetics, 4(7), 2008.

References IV


J. S. Ide and F. G. Cozman.
Random Generation of Bayesian Networks.
In *Proceedings of the 16th Brazilian Symposium on Artificial Intelligence*, pages 366–375. Springer-Verlag, 2002.


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.


W. James and C. Stein.
Estimation with Quadratic Loss.
In J. Neyman, editor, *Proceedings of the 4th Berkeley Symposium on Mathematical Statistics and Probability*, pages 361–379, 1961.


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

References V

K. Korb and A. Nicholson.
Bayesian Artificial Intelligence.
Chapman and Hall, 2nd edition, 2009.

S. Kullback.
Information Theory and Statistics.
Dover Publications, 1968.

O. Ledoit and M. Wolf.
Improved Estimation of the Covariance Matrix of Stock Returns with an Application to Portfolio Selection.
Journal of Empirical Finance, 10:603–621, 2003.

P. Legendre.
Comparison of Permutation Methods for the Partial Correlation and Partial Mantel Tests.
Journal of Statistical Computation and Simulation, 67:37–73, 2000.

References VI

D. Margaritis.
Learning Bayesian Network Model Structure from Data.
PhD thesis, School of Computer Science, Carnegie-Mellon University, Pittsburgh, PA, May 2003.
Available as Technical Report CMU-CS-03-153.

G. Melançon, I. Dutour, and M. Bousquet-Mélou.
Random Generation of DAGs for Graph Drawing.
Technical Report INS-R0005, Centre for Mathematics and Computer Sciences, Amsterdam, 2000.

J. Pearl.
Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.
Morgan Kaufmann, 1988.

F. Pesarin and L. Salmaso.
Permutation Tests for Complex Data: Theory, Applications and Software.
Wiley, 2010.

S. J. Russell and P. Norvig.
Artificial Intelligence: A Modern Approach.
Prentice Hall, 3rd edition, 2009.

References VII

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.
A Shrinkage Approach to Large-Scale Covariance Matrix Estimation and Implications for Functional Genomics.
Statistical Applications in Genetics and Molecular Biology, 4:32, 2005.

G. E. Schwarz.
Estimating the Dimension of a Model.
Annals of Statistics, 6(2):461 – 464, 1978.

M. Scutari and K. Strimmer.
Introduction to Graphical Modelling.
In D. J. Balding, M. Stumpf, and M. Girolami, editors, *Handbook of Statistical Systems Biology*. Wiley, 2011.
In print.

References VIII

P. Spirtes, C. Glymour, and R. Scheines.
Causation, Prediction, and Search.
MIT Press, 2000.

C. Stein.
Inadmissibility of the Usual Estimator for the Mean of a Multivariate Distribution.
In J. Neyman, editor, *Proceedings of the 3rd Berkeley Symposium on Mathematical Statistics and Probability*, pages 197–206, 1956.

I. Tsamardinos, C. F. Aliferis, and A. Statnikov.
Algorithms for Large Scale Markov Blanket Discovery.
In *Proceedings of the 16th International Florida Artificial Intelligence Research Society Conference*, pages 376–381. AAAI Press, 2003.

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.

T. S. Verma and J. Pearl.
Equivalence and Synthesis of Causal Models.
Uncertainty in Artificial Intelligence, 6:255–268, 1991.

References IX

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

S. Yaramakala and D. Margaritis.
Speculative Markov Blanket Discovery for Optimal Feature Selection.
In *Proceedings of the 5th IEEE International Conference on Data Mining*, pages 809–812. IEEE Computer Society, 2005.