Structure Estimation in Graphical Models

Steffen Lauritzen, University of Oxford

Wald Lecture, World Meeting on Probability and Statistics

Istanbul 2012

Advances in computing has set focus on *estimation of structure*:

- Model selection (e.g. subset selection in regression)

Advances in computing has set focus on *estimation of structure*:

- Model selection (e.g. subset selection in regression)
- System identification (engineering)

Advances in computing has set focus on *estimation of structure*:

- Model selection (e.g. subset selection in regression)
- System identification (engineering)
- Structural learning (AI or machine learning)

Advances in computing has set focus on *estimation of structure*:

- Model selection (e.g. subset selection in regression)
- System identification (engineering)
- Structural learning (AI or machine learning)

Graphical models describe conditional independence structures, so good case for formal analysis.

Methods must scale well with data size, as *many* structures and *huge* collections of data are to be considered.

Why estimation of structure?

- Parallel to e.g. density estimation

Why estimation of structure?

- Parallel to e.g. density estimation
- Obtain quick overview of relations between variables in complex systems

Why estimation of structure?

- Parallel to e.g. density estimation
- Obtain quick overview of relations between variables in complex systems
- Data mining

Why estimation of structure?

- Parallel to e.g. density estimation
- Obtain quick overview of relations between variables in complex systems
- Data mining
- Gene regulatory networks

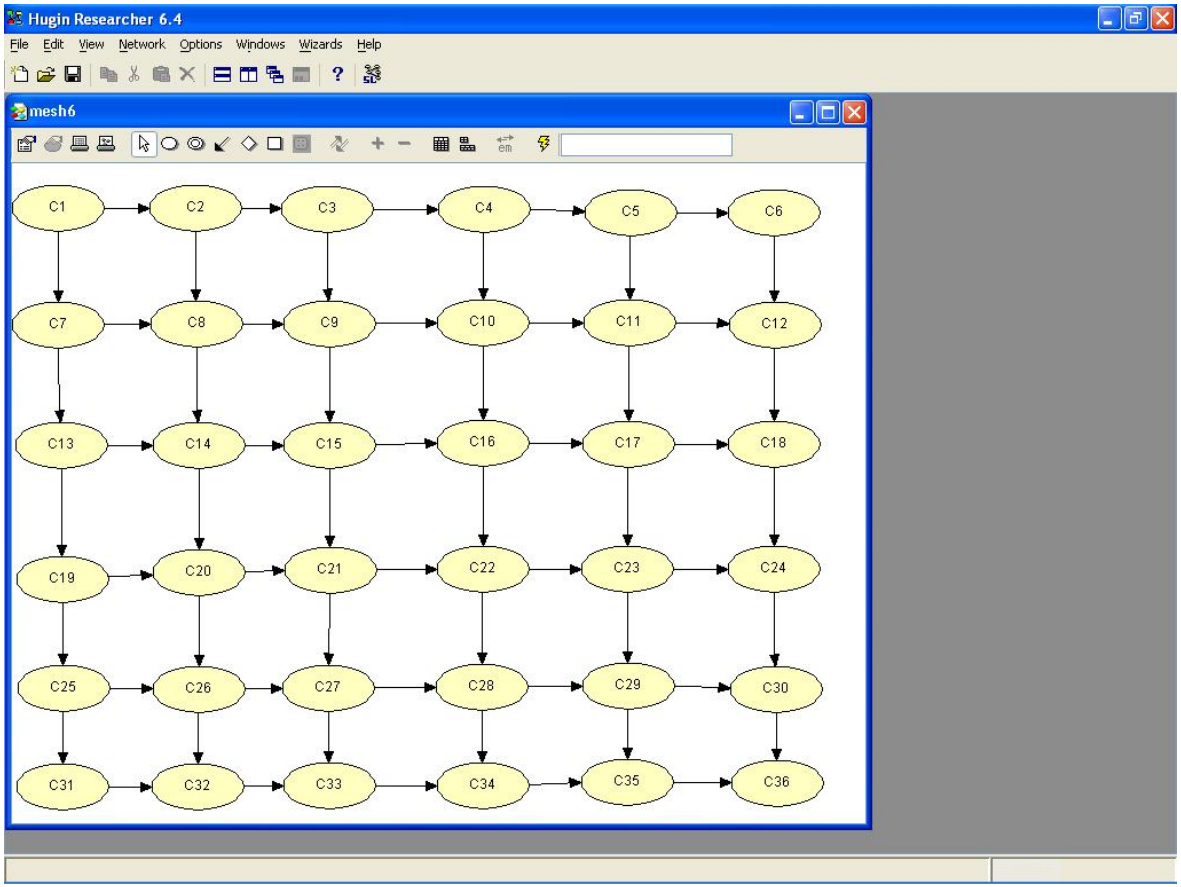
Why estimation of structure?

- Parallel to e.g. density estimation
- Obtain quick overview of relations between variables in complex systems
- Data mining
- Gene regulatory networks
- Reconstructing family trees from DNA information

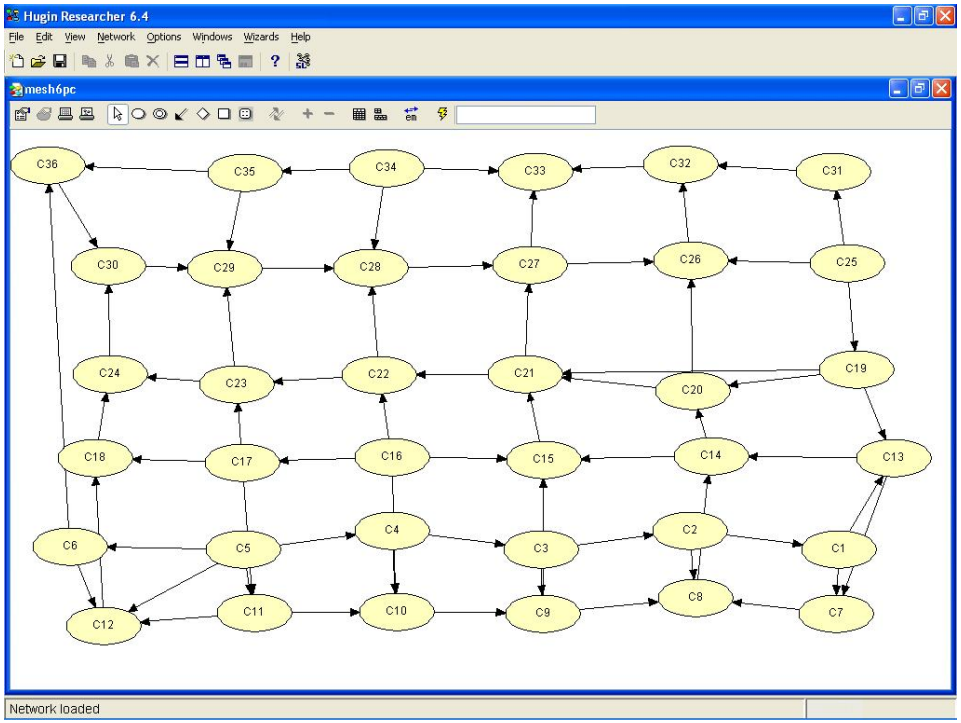
Why estimation of structure?

- Parallel to e.g. density estimation
- Obtain quick overview of relations between variables in complex systems
- Data mining
- Gene regulatory networks
- Reconstructing family trees from DNA information
- General interest in sparsity.

Markov mesh model

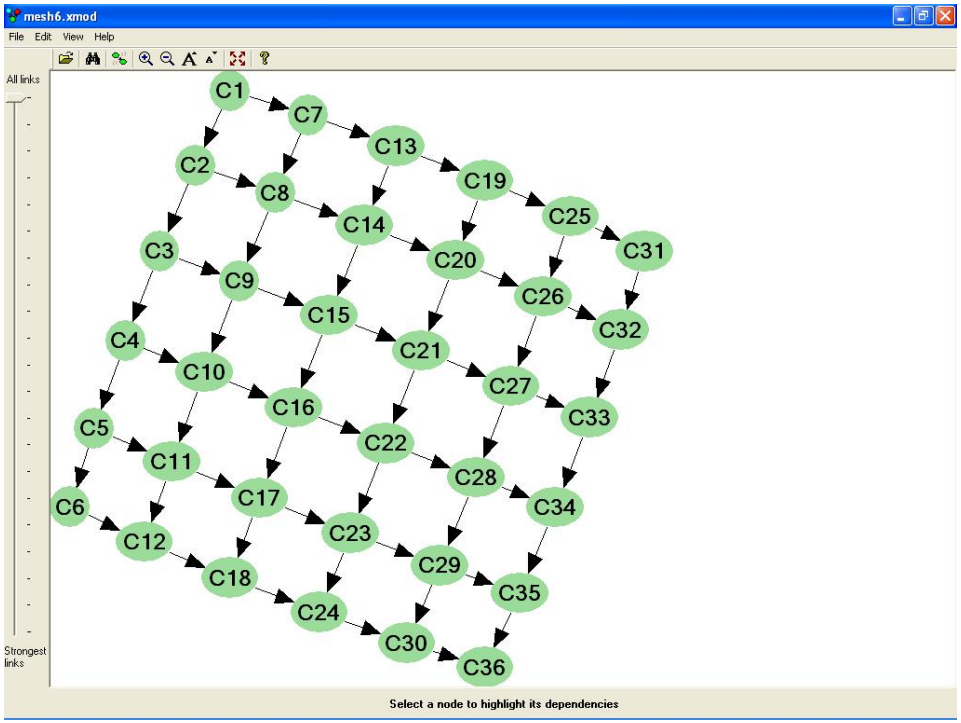


PC algorithm



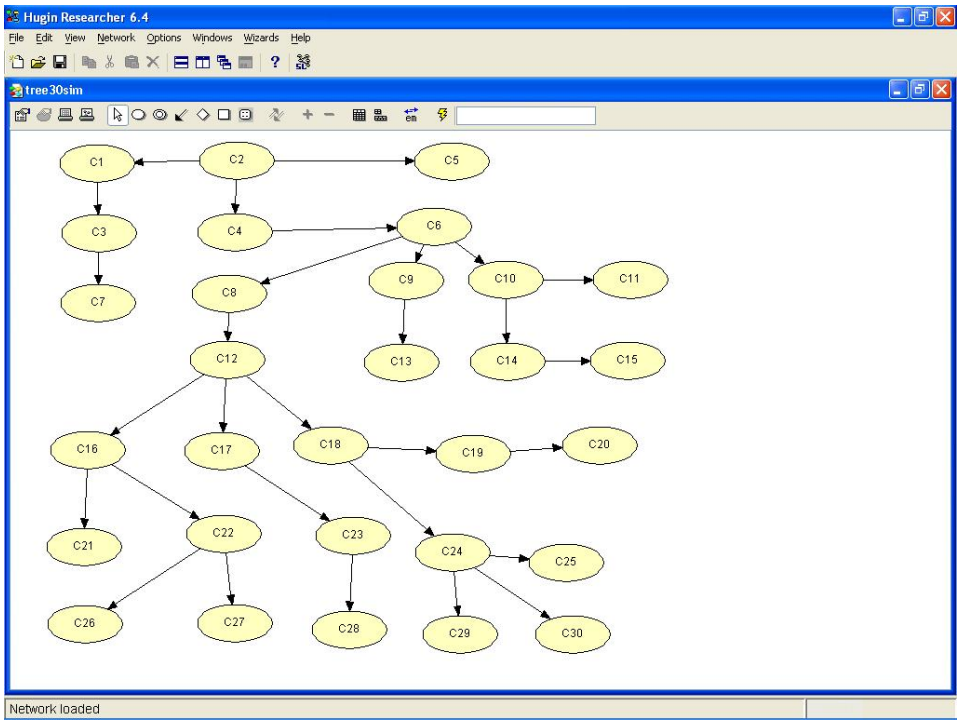
Crudest algorithm (HUGIN), 10000 simulated cases

Bayesian GES



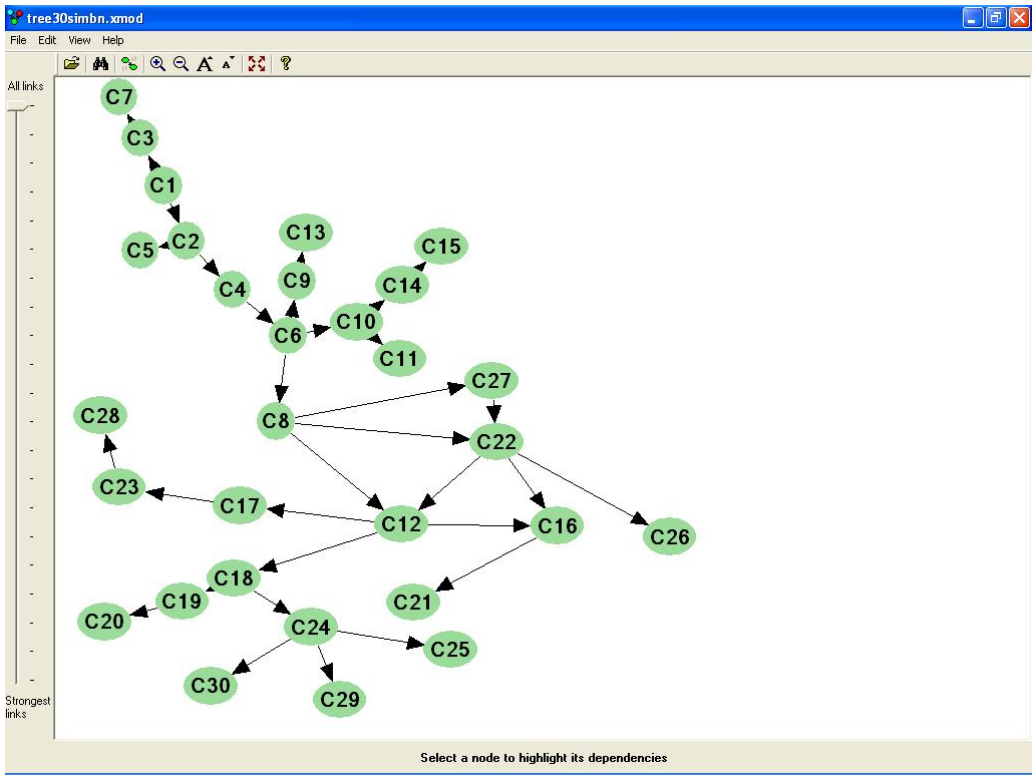
Crudest algorithm (WinMine), 10000 simulated cases

Tree model

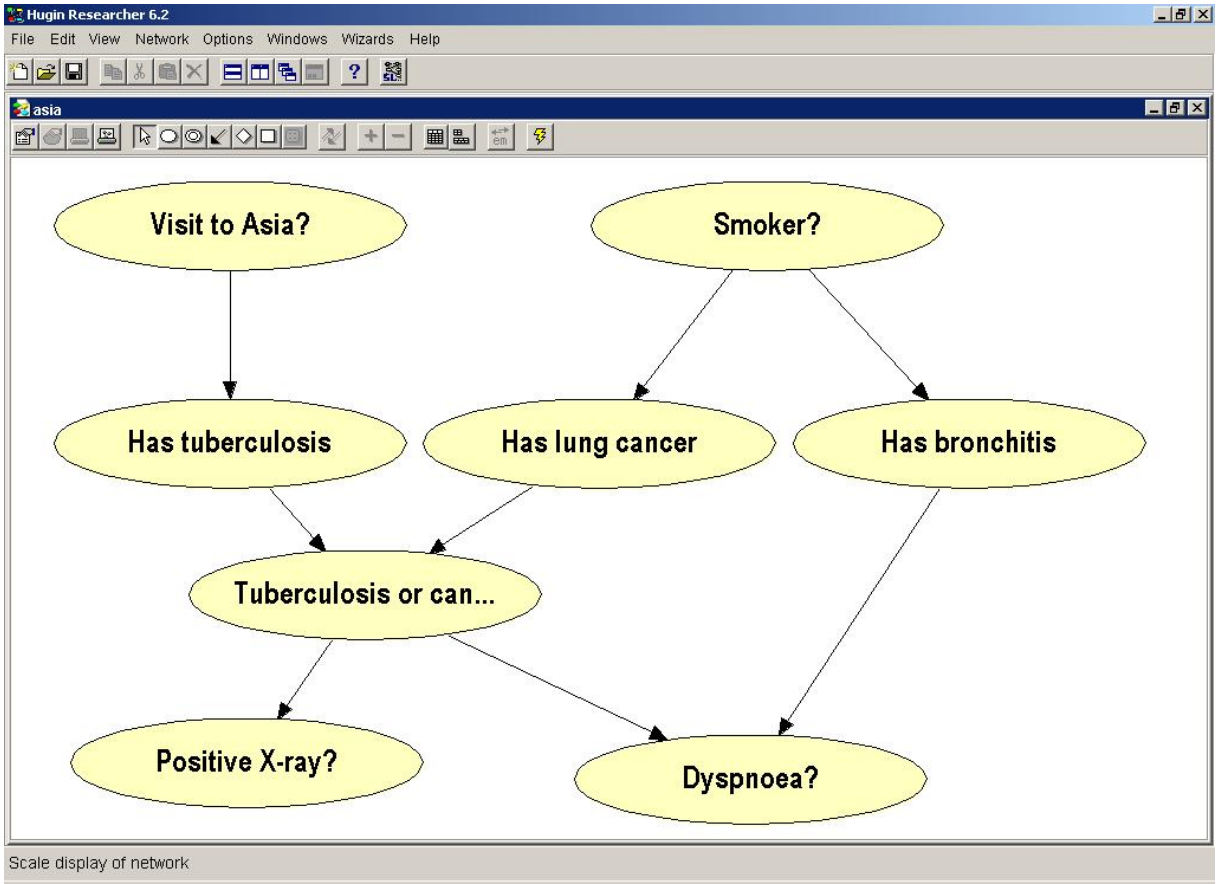


PC algorithm, 10000 cases, correct reconstruction

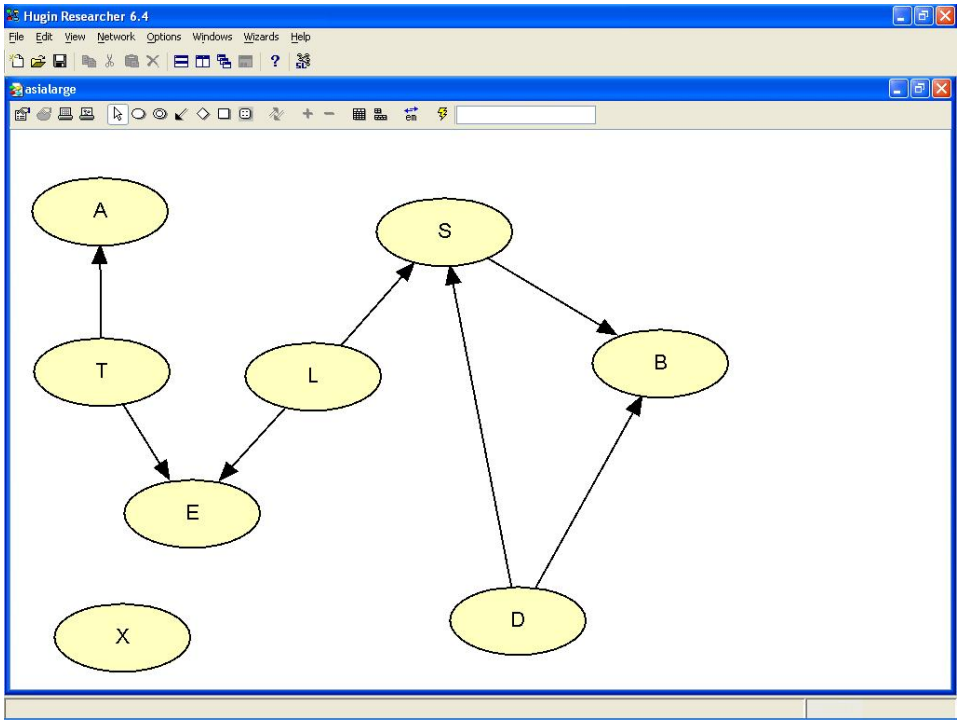
Bayesian GES on tree



Chest clinic

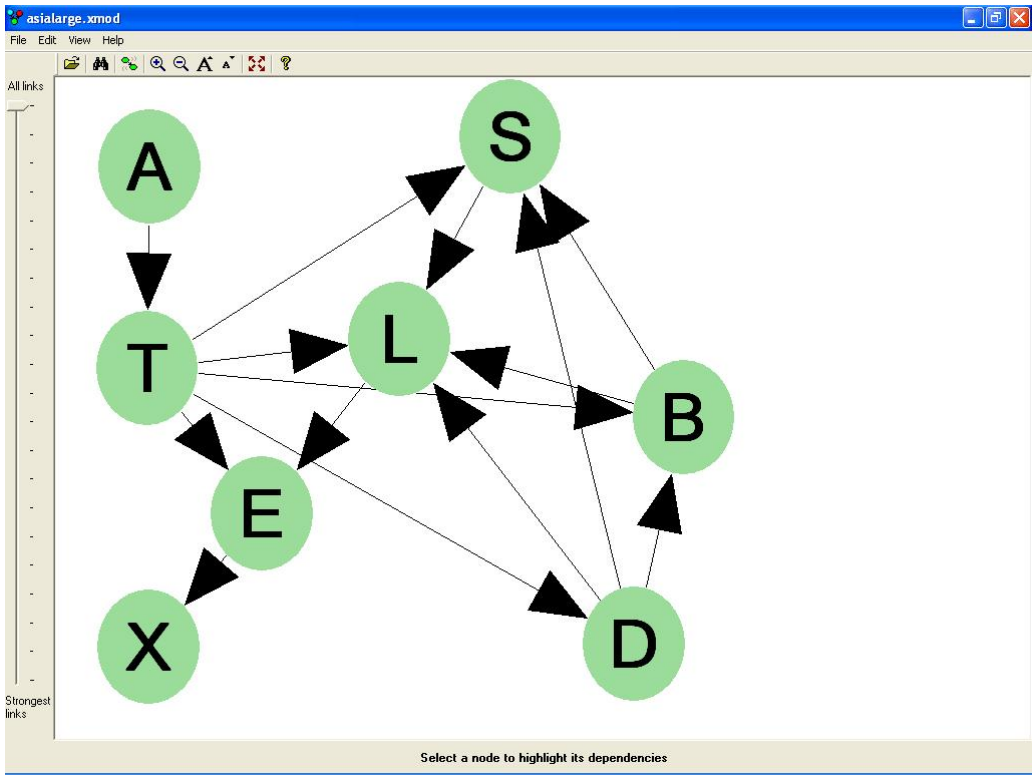


PC algorithm

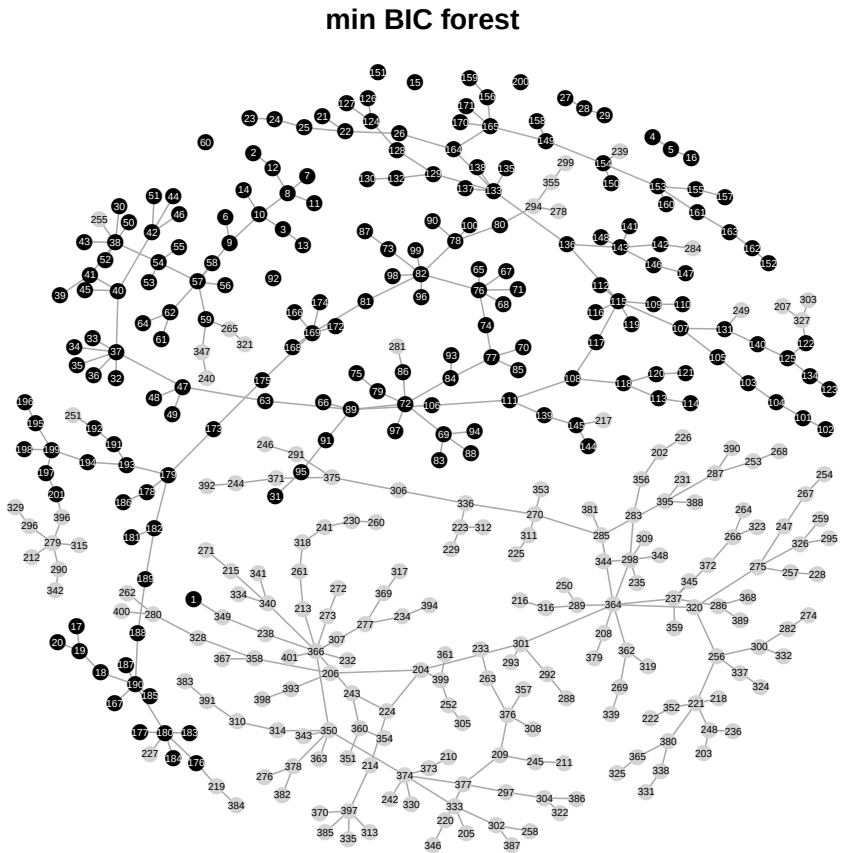


10000 simulated cases

Bayesian GES



SNPs and gene expressions



Methods for structure identification in graphical models can be classified into three types:

- *score-based methods:* For example optimizing a penalized likelihood by using convex programming e.g. `glasso`;

Methods for structure identification in graphical models can be classified into three types:

- *score-based methods:* For example optimizing a penalized likelihood by using convex programming e.g. `glasso`;
- *Bayesian methods:* Identifying posterior distributions over graphs; can also use posterior probability as score.

Methods for structure identification in graphical models can be classified into three types:

- *score-based methods:* For example optimizing a penalized likelihood by using convex programming e.g. `glasso`;
- *Bayesian methods:* Identifying posterior distributions over graphs; can also use posterior probability as score.
- constraint-based methods: Querying conditional independences and identifying compatible independence structures, for example PC, PC*, NPC, IC, CI, FCI, SIN, QP, ...

Assume P factorizes w.r.t. an unknown tree τ. Based on a sample $x^{(n)} = (x^1, \ldots, x^n)$ the likelihood function becomes

$$\begin{aligned} \ell(\tau, p) &= \log L(\tau) = \log p(x^{(n)} \mid \tau) \\ &= \sum_{e \in \mathbf{E}(\tau)} \log p(x_e^{(n)}) - \sum_{v \in V} \{\deg(v) - 1\} \log p(x_v^{(n)}). \end{aligned}$$

Maximizing this over p for a fixed tree τ yields the profile likelihood

$$\hat{l}(\tau) = l(\tau, \hat{p}) = \sum_{e \in E(\tau)} H_n(e) + \sum_{v \in V} H_n(v)$$

Here $H_n(e)$ is the empirical *cross-entropy* or *mutual information* between endpoint variables of the edge $e = \{u, v\}$:

$$H_n(e) = \sum \frac{n(x_u, x_v)}{n} \log \frac{n(x_u, x_v)/n}{n(x_u)n(x_v)/n^2}$$

and similarly $H_n(v)$ is the empirical entropy of X_v.

Based on this fact, (Chow and Liu, 1968) showed *MLE $\hat{\tau}$ of $\mathcal{T}$ has maximal weight*, where the *weight* of τ is

$$w(\tau) = \sum_{e \in E(\tau)} H_n(e) = \hat{l}(\tau) - \hat{l}(\phi_0).$$

Here ϕ_0 is the graph with all vertices isolated.

Fast algorithms (Kruskal Jr., 1956) compute maximal weight spanning tree from weights $W = (w_{uv}, u, v \in V)$.

Chow and Wagner (1978) show *a.s. consistency in total variation of* $\hat{P}$*:* If P factorises w.r.t. τ, then

$$\sup_x |p(x) - \hat{p}(x)| \to 0 \text{ for } n \to \infty,$$

so *if* τ *is unique for* P, $\hat{\tau} = \tau$ *for all* $n > N$ *for some* N.

If P does not factorize w.r.t. a tree, $\hat{P}$ *converges to closest tree-approximation* $\tilde{P}$ *to* P (Kullback-Leibler distance).

Note that if P is Markov w.r.t. and undirected graph $\mathcal{G}$ and we define weights on edges as $w(e) = H(e)$ by cross-entropy, $\tilde{P}$ is Markov w.r.t any maximal weight spanning tree of $\mathcal{G}$.

Forests

Note that if we consider forests instead of trees, i.e. allow missing branches, it is still true that

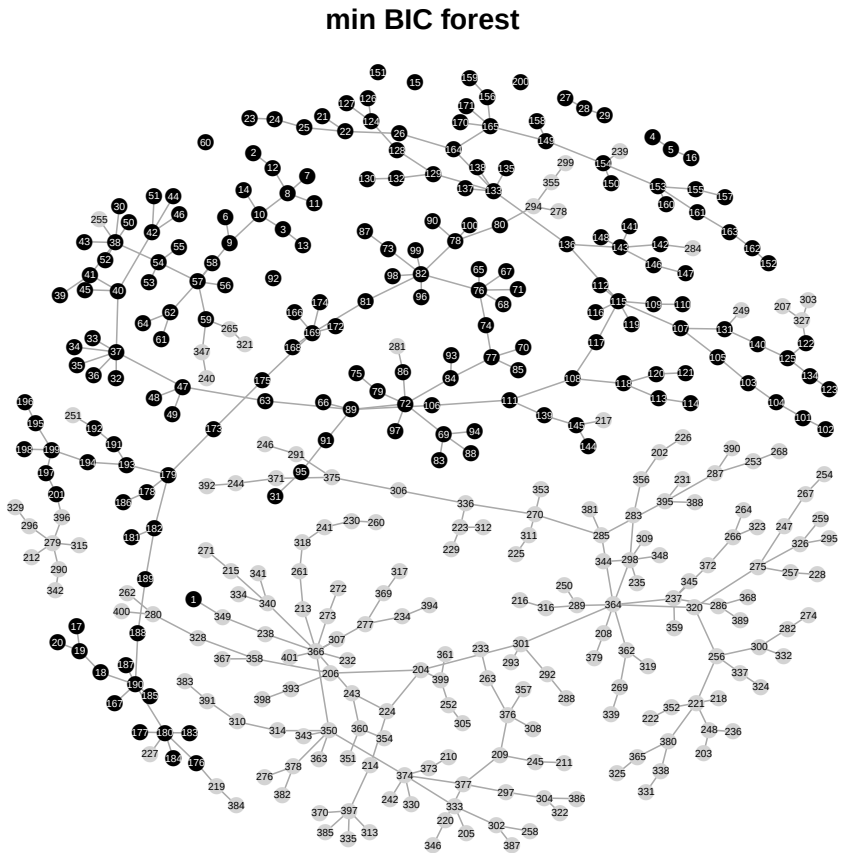
$$\hat{l}(\phi) = l(\phi, \hat{p}) = \sum_{e \in E(\phi)} H_n(e) + \sum_{v \in V} H_n(v).$$

Thus if we add a penalty for each edge, e.g. proportional to the number of additional parameters q_e of introducing the edge e, we can find the maximum penalised forest by Kruskal's algorithm using weights

$$\tilde{w}(\phi) = \sum_{e \in E(\phi)} \{H_n(e) - \lambda q_e\}.$$

This has been exploited in the package `gRapHD` (Edwards et al., 2010), see also Panayidou (2011) and Højsgaard et al. (2012).

SNPs and gene expressions



Gaussian Trees

If $X = (X_v, v \in V)$ is regular multivariate Gaussian, it factorizes w.r.t. an undirected graph if and only if its *concentration matrix* $K = \Sigma^{-1}$ satisfies

$$k_{uv} = 0 \iff u \not\sim v.$$

Results of Chow et al. are easily extended to Gaussian trees, with the weight of a tree determined as

$$w(\tau) = \sum_{e \in E(\tau)} -\log(1 - r_e^2),$$

with r_e^2 being the empirical correlation coeffient between X_u and X_v.

- *Direct likelihood methods* (ignoring penalty terms) lead to sensible results.

- *Direct likelihood methods* (ignoring penalty terms) lead to sensible results.
- (Boootstrap) sampling distribution of tree MLE can be *simulated*

- *Direct likelihood methods* (ignoring penalty terms) lead to sensible results.
- (Boootstrap) sampling distribution of tree MLE can be *simulated*
- *Penalty terms additive along branches*, so highest AIC or BIC scoring tree (forest) also available using a MWST algorithm.

- *Direct likelihood methods* (ignoring penalty terms) lead to sensible results.
- (Boootstrap) sampling distribution of tree MLE can be *simulated*
- *Penalty terms additive along branches*, so highest AIC or BIC scoring tree (forest) also available using a MWST algorithm.
- Tree methods scale extremely well with both sample size and number of variables;

- *Direct likelihood methods* (ignoring penalty terms) lead to sensible results.
- (Boootstrap) sampling distribution of tree MLE can be *simulated*
- *Penalty terms additive along branches*, so highest AIC or BIC scoring tree (forest) also available using a MWST algorithm.
- Tree methods scale extremely well with both sample size and number of variables;
- Pairwise marginal counts are *sufficient statistics* for the tree problem (empirical covariance matrix in the Gaussian case).

- *Direct likelihood methods* (ignoring penalty terms) lead to sensible results.
- (Boootstrap) sampling distribution of tree MLE can be *simulated*
- *Penalty terms additive along branches*, so highest AIC or BIC scoring tree (forest) also available using a MWST algorithm.
- Tree methods scale extremely well with both sample size and number of variables;
- Pairwise marginal counts are *sufficient statistics* for the tree problem (empirical covariance matrix in the Gaussian case).

Note sufficiency holds despite parameter space very different from open subset of $\mathcal{R}^k$.

Consider an undirected Gaussian graphical model and the l_1-penalized log-likelihood function

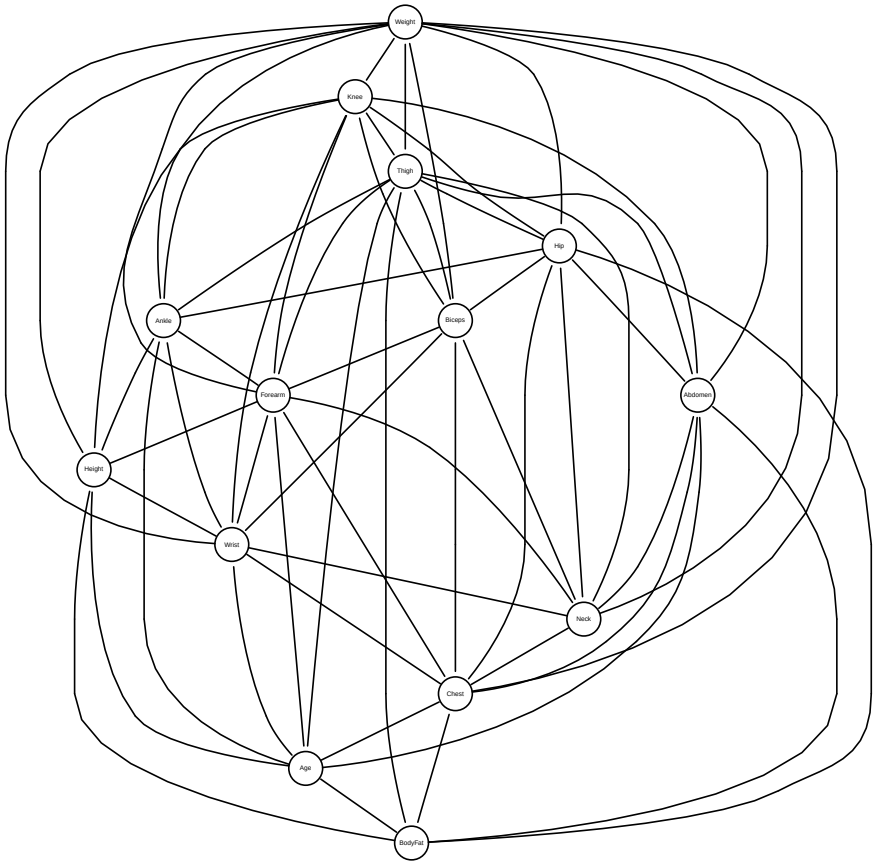
$$2\ell_{pen}(K) = \log\det K - \mathrm{tr}(KW) - \lambda||K||_1^*.$$

The penalty $||K||_1^* = \sum_{i \neq j} |k_{ij}|$ is essentially a convex relaxation of the number of edges in the graph and optimization of the penalized likelihood will typically lead to several $k_{ij} = 0$ and thus in effect estimate a particular graph.

This penalized likelihood can be maximized efficiently (Banerjee et al., 2008) as implemented in the *graphical lasso* (Friedman et al., 2008).

Beware: not scale-invariant!

glasso for bodyfat

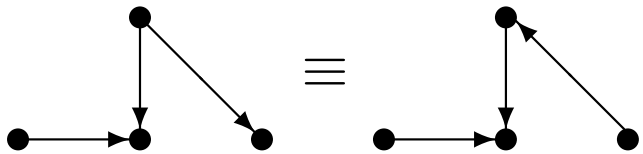


Markov equivalence

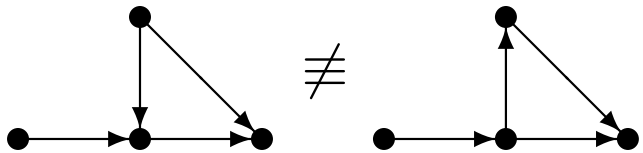
$\mathcal{D}$ and $\mathcal{D}'$ are equivalent if and only if:

1. $\mathcal{D}$ and $\mathcal{D}'$ have same *skeleton* (ignoring directions)
2. $\mathcal{D}$ and $\mathcal{D}'$ have same unmarried parents

so



but



Equivalence class searches

Searches directly in equivalence classes of DAGs.

Define *score function* $\sigma(\mathcal{D})$, with the property that

$$\mathcal{D} \equiv \mathcal{D}' \Rightarrow \sigma(\mathcal{D}) = \sigma(\mathcal{D}').$$

This holds e.g. if score function is *AIC or BIC or full Bayesian posterior with strong hyper Markov prior* (based upon Dirichlet or inverse Wishart distributions).

Equivalence class with maximal score is sought.

dlasso? problems with invariance over equivalence classes!

Greedy equivalence class search

1. Initialize with empty DAG
2. Repeatedly search among equivalence classes with a *single additional edge* and go to class with highest score - until no improvement.
3. Repeatedly search among equivalence classes with a *single edge less* and move to one with highest score - until no improvement.

For BIC, for example, this algorithm yields consistent estimate of equivalence class for P. (Chickering, 2002)

For g in specified set of graphs, Θ_g is associated parameter space so that P factorizes w.r.t. g if and only if $P = P_\theta$ for some $\theta \in \Theta_g$.

π_g is prior on Θ_g. Prior $p(g)$ is uniform for simplicity.

Based on $x^{(n)}$, posterior distribution of G is

$$p^*(g) = p(g \mid x^{(n)}) \propto p(x^{(n)} \mid g) = \int_{\Theta_g} p(x^{(n)} \mid g, \theta)\, \pi_g(d\theta).$$

Bayesian analysis looks for *MAP estimate* g^* maximizing $p^*(g)$ or attempts to *sample from posterior*, using e.g. MCMC.

For connected decomposable graphs and *strong hyper Markov priors* Dawid and Lauritzen (1993) show

$$p(x^{(n)} \mid g) = \frac{\prod_{C \in \mathcal{C}} p(x_C^{(n)})}{\prod_{S \in \mathcal{S}} p(x_S^{(n)})},$$

where each factor has explicit form. $\mathcal{C}$ are the *cliques* of g and $\mathcal{S}$ the *separators* (mininal cutsets).

Hence, if the prior distributions over a class of graphs is uniform, the posterior distribution has the form

$$p(g \mid x^{(n)}) \propto \frac{\prod_{C \in \mathcal{C}} p(x_C^{(n)})}{\prod_{S \in \mathcal{S}} p(x_S^{(n)})} = \frac{\prod_{C \in \mathcal{C}} w(C)}{\prod_{S \in \mathcal{S}} w(S)}.$$

Byrne (2011) shows that a distribution over decomposable graphs having the form

$$p(g) \propto \frac{\prod_{C \in \mathcal{C}} w(C)}{\prod_{S \in \mathcal{S}} w(S)},$$

satisfies a *structural Markov property* so that for subsets A and B with $V = A \cup B$, it holds that

$$\mathcal{G}_A \perp\!\!\!\perp \mathcal{G}_B \,|\, (A, B) \text{ is a decomposition of } \mathcal{G}.$$

In words, the finer structure of the graph bits in two components of any decomposition are independent.

Trees are decomposable, so for trees we get

$$p(\tau \mid x^{(n)}) \propto \prod_{e \in E(\tau)} p(x_e^{(n)})$$

which more illuminating can be expressed as

$$p(\tau \mid x^{(n)}) \propto \prod_{e \in E(\tau)} B_e$$

where

$$B_{uv} = \frac{p(x_{uv}^{(n)})}{p(x_u^{(n)})p(x_v^{(n)})}$$

is the *Bayes factor* for independence along the edge uv.

MAP estimates of trees can be computed (also in Gaussian case)

Good direct algorithms exist for generating random spanning trees (Guénoche, 1983; Broder, 1989; Aldous, 1990; Propp and Wilson, 1998) so *full posterior analysis is possible for trees.*

MCMC methods for exploring posteriors of undirected graphs have been developed.

Even for forests, it seems complicated to sample from the posterior distribution (Dai, 2008).

$$p(\phi \mid x^{(n)}) \propto \prod_{e \in E(\phi)} B_e$$

Some challenges for undirected graphs

- Find feasible algorithm for (perfect) simulation from a distribution over decomposable graphs as

$$p(g) \propto \frac{\prod_{C \in \mathcal{C}} w(C)}{\prod_{S \in \mathcal{S}} w(S)},$$

  where $w(A), A \subseteq V$ are a prescribed set of positive weights.
- Find feasible algorithm for obtaining MAP in decomposable case. This may not be universally possible as problem is NP-complete, even for bounded maximum clique size.

Posterior distribution for DAG

For strong directed hyper Markov priors it holds that

$$p(x^{(n)} \mid d) = \prod_{v \in V} p(x_v^{(n)} \mid x_{\mathrm{pa}(v)}^{(n)})$$

so

$$p(d \mid x^{(n)}) \propto \prod_{v \in V} p(x_v^{(n)} \mid x_{\mathrm{pa}(v)}^{(n)}),$$

see e.g. Spiegelhalter and Lauritzen (1990); Cooper and Herskovits (1992); Heckerman et al. (1995)

Challenge: Find good algorithm for sampling from this full posterior.

First step of constraint-based methods (eg PC-algorithm) is to identify *skeleton* of $\mathcal{G}$, which is the undirected graph with $\alpha \not\sim \beta$ if and only if there exists $S \subseteq V \setminus \{\alpha, \beta\}$ with $\alpha \perp_g \beta \,|\, S$.

The skeleton stage of any such algorithm is thus correct for any type of graph which represents the independence structure!

In particular, the *the PC algorithm can easily be adapted to UGs and CHGs* (Meek, 1996).

Challenge: the properties of these algorithms when translated to sampling situations are not sufficiently well understood.

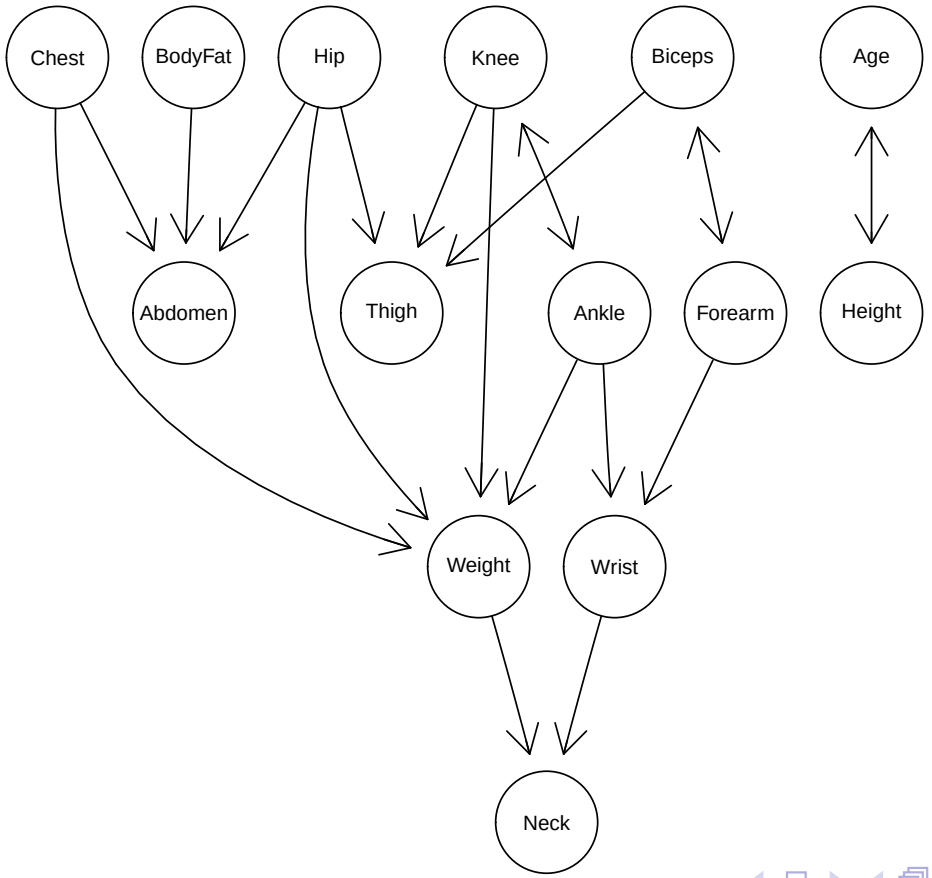
Step 1: Identify skeleton using that, for P faithful,

$$u \not\sim v \iff \exists S \subseteq V \setminus \{u, v\} : X_u \perp\!\!\!\perp X_v \mid X_S.$$

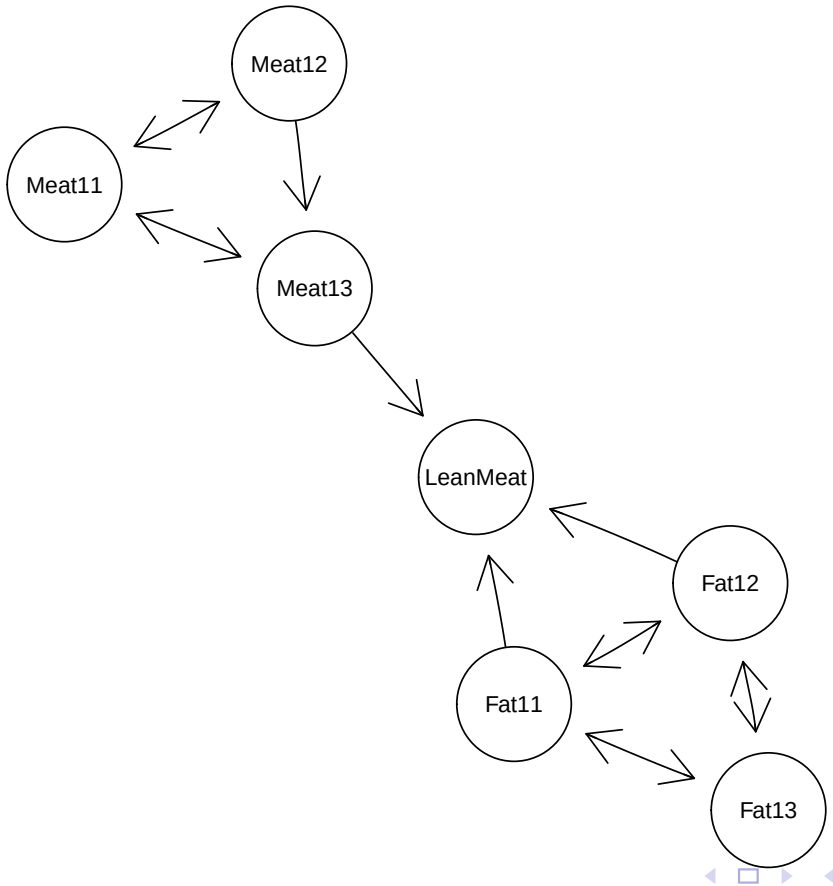
Begin with complete graph, check for $S = \emptyset$ and remove edges when independence holds. Then continue for increasing $|S|$.
PC-algorithm (Spirtes et al., 1993) exploits that only S with $S \subseteq \mathrm{ne}(u)$ or $S \subseteq \mathrm{ne}(v)$ needs checking, ne refers to current skeleton.

Step 2: *Identify directions to be consistent with independence relations found in Step 1.*

pcalg for bodyfat



pcalg for carcass



Faithfulness

A given distribution P is in general compatible with a variety of structures, i.e. if P corresponds to complete independence. To identify a structure $\mathcal{G}$ something like the following must hold

P is said to be *faithful* to $\mathcal{G}$ if

$$A \perp\!\!\!\perp B \mid X_S \iff A \perp_g B \mid S.$$

Most distributions are faithful. More precisely, for DAGs it holds that the non-faithful distributions form a Lebesgue null-set in parameter space associated with a DAG.

Exact properties of PC-algorithm

If P is faithful to DAG $\mathcal{D}$, PC-algorithm finds $\mathcal{D}'$ equivalent to $\mathcal{D}$.
It uses N independence checks where N is at most

$$N \leq 2\binom{|V|}{2}\sum_{i=0}^{d}\binom{|V|-1}{i} \leq \frac{|V|^{d+1}}{(d-1)!},$$

where d is the maximal degree of any vertex in $\mathcal{D}$. f So worst case complexity is exponential, but *algorithm fast for sparse graphs.* Sampling properties are less well understood although consistency results exist.

Constraint based methods establish fundamentally two lists:

1. An *independence list* $\mathcal{I}$ of triplets (α, β, S) with $\alpha \perp\!\!\!\perp \beta \,|\, S$; identifies *skeleton*;
2. A *dependence list* $\mathcal{D}$ of triplets (α, β, S) with $\neg(\alpha \perp\!\!\!\perp \beta \,|\, S)$.

They get established recursively, for increasing size of S, and the list $\mathcal{I}$ is most likely to have errors. The lists may or may not be consistent with a DAG model, but methods exist for checking this. *Question:* Assume a current pair of input lists is consistent with compositional graphoid axioms and a new triplet (α, β, S) is considered. Can it be verified whether it can be consistently added to any of the two lists? Graph representable? Compositional?

- Seems out of hand to extend Bayesian and score based methods to more general graphical models;

- Seems out of hand to extend Bayesian and score based methods to more general graphical models;
- Fully Bayesian methods do typically not scale well with number of variables;

- Seems out of hand to extend Bayesian and score based methods to more general graphical models;
- Fully Bayesian methods do typically not scale well with number of variables;
- Constraint based methods have less clear formal inference basis; *challenge to improve this*.

- Seems out of hand to extend Bayesian and score based methods to more general graphical models;
- Fully Bayesian methods do typically not scale well with number of variables;
- Constraint based methods have less clear formal inference basis; *challenge to improve this*.
- Constraint based methods have been developed to work in cases where P is *faithful* and conditional independence queries can be resolved without error.

- Factorisation properties for Markov distributions;

- Factorisation properties for Markov distributions;
- Local computation algorithms for speeding up any relevant computation;

- Factorisation properties for Markov distributions;
- Local computation algorithms for speeding up any relevant computation;
- Causality and causal interpretations;

- Factorisation properties for Markov distributions;
- Local computation algorithms for speeding up any relevant computation;
- Causality and causal interpretations;
- Non-parametric graphical models for large scale data analysis;

- Factorisation properties for Markov distributions;
- Local computation algorithms for speeding up any relevant computation;
- Causality and causal interpretations;
- Non-parametric graphical models for large scale data analysis;
- Probabilistic expert systems, for example in forensic identification problems;

- Factorisation properties for Markov distributions;
- Local computation algorithms for speeding up any relevant computation;
- Causality and causal interpretations;
- Non-parametric graphical models for large scale data analysis;
- Probabilistic expert systems, for example in forensic identification problems;
- Markov theory for infinite graphs (huge graphs).

- Factorisation properties for Markov distributions;
- Local computation algorithms for speeding up any relevant computation;
- Causality and causal interpretations;
- Non-parametric graphical models for large scale data analysis;
- Probabilistic expert systems, for example in forensic identification problems;
- Markov theory for infinite graphs (huge graphs).
- *THANK YOU FOR LISTENING!*

Aldous, D. (1990). A random walk construction of uniform spanning trees and uniform labelled trees. *SIAM Journal on Discrete Mathematics*, 3(4):450–465.

Banerjee, O., Ghaoui, L. E., and dAspremont, A. (2008). Model selection through sparse maximum likelihood estimation for multivariate Gaussian or binary data. 9:485–216.

Broder, A. (1989). Generating random spanning trees. In *Thirtieth Annual Symposium of Computer Science*, pages 442–447.

Byrne, S. (2011). *Hyper and Structural Markov Laws for Graphical Models*. PhD thesis, Statistical Laboratory, University of Cambridge.

Chickering, D. M. (2002). Optimal structure identification with greedy search. *Journal of Machine Learning Research*, 3:507–554.

Chow, C. K. and Liu, C. N. (1968). Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, 14:462–467.

Chow, C. K. and Wagner, T. J. (1978). Consistency of an estimate of tree-dependent probability distributions. *IEEE Transactions on Information Theory*, 19:369–371.

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

Dai, H. (2008). Perfect sampling methods for random forests. *Advances in Applied Probability*, 40:897–917.

Dawid, A. P. and Lauritzen, S. L. (1993). Hyper Markov laws in the statistical analysis of decomposable graphical models. *The Annals of Statistics*, 21:1272–1317.

Edwards, D., de Abreu, G., and Labouriau, R. (2010). Selecting high-dimensional mixed graphical models using minimal AIC or BIC forests. *BMC Bioinformatics*, 11(1):18.

Friedman, J., Hastie, T., and Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441.

Guénoche, A. (1983). Random spanning tree. *Journal of Algorithms*, 4(3):214 –220.

Heckerman, D., Geiger, D., and Chickering, D. M. (1995). Learning Bayesian networks: The combination of knowledge and statistical data. *Machine Learning*, 20:197–243.

Højsgaard, S., Edwards, D., and Lauritzen, S. (2012). *Graphical Models with R*. Springer-Verlag, New York.

Kruskal Jr., J. B. (1956). On the shortest spanning subtree of a

graph and the travelling salesman problem. *Proceedings of the American Mathematical Society*, 7:48–50.

Meek, C. (1996). *Selecting Graphical Models: Causal and Statistical Reasoning*. PhD thesis, Carnegie–Mellon University.

Panayidou, K. (2011). *Estimation of Tree Structure for Variable Selection*. PhD thesis, Department of Statistics, University of Oxford.

Propp, J. G. and Wilson, D. B. (1998). How to get a perfectly random sample from a generic Markov chain and generate a random spanning tree of a directed graph. *Journal of Algorithms*, 27(2):170 –217.

Spiegelhalter, D. J. and Lauritzen, S. L. (1990). Sequential updating of conditional probabilities on directed graphical structures. *Networks*, 20:579–605.

Spirtes, P., Glymour, C., and Scheines, R. (1993). *Causation, Prediction and Search*. Springer-Verlag, New York. Reprinted by MIT Press.