Learning the Structure of Graphical Models with Latent Variables:

Variational and non-parametric approaches

Zoubin Ghahramani

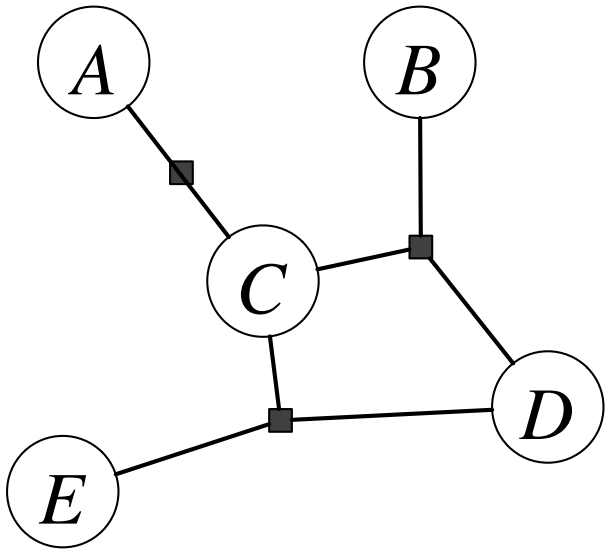
Department of Engineering
University of Cambridge, UK

Machine Learning Department
Carnegie Mellon University, USA

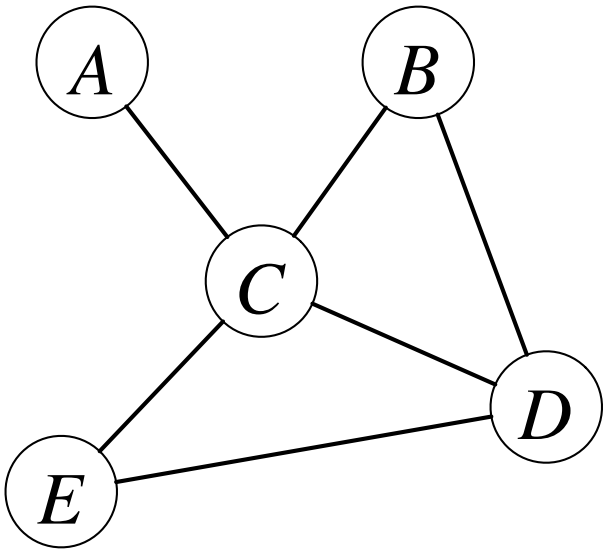
zoubin@eng.cam.ac.uk
http://learning.eng.cam.ac.uk/zoubin/

MLSS 2011

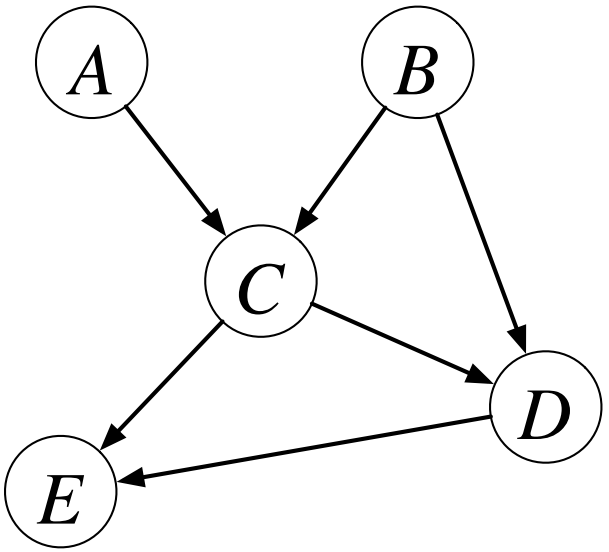
Three main kinds of graphical models



factor graph



undirected graph



directed graph

- Nodes correspond to random variables
- Edges represent statistical dependencies between the variables

Why do we need graphical models?

- Graphs are an **intuitive** way of representing and visualising the relationships between many variables. (Examples: family trees, electric circuit diagrams, regulatory networks)

- A graph allows us to abstract out the **conditional independence** relationships between the variables from the details of their parametric forms. Thus we can answer questions like: "Is A dependent on B given that we know the value of C ?" just by looking at the graph.

- Graphical models allow us to define general **message-passing algorithms** that implement probabilistic inference efficiently. Thus we can answer queries like "What is $p(A|C = c)$?" without enumerating all settings of all variables in the model.

Graphical models = statistics $\times$ graph theory $\times$ computer science.

Conditional Independence

Marginal Independence:

$$X \perp\!\!\!\perp Y \quad \Leftrightarrow \quad X \perp\!\!\!\perp Y | \emptyset \quad \Leftrightarrow \quad p(X, Y) = p(X)\, p(Y)$$

Conditional Independence:

$$X \perp\!\!\!\perp Y | V \quad \Leftrightarrow \quad p(X, Y | V) = p(X | V)\, p(Y | V)$$

Also, when $p(Y, V) > 0$:

$$X \perp\!\!\!\perp Y | V \quad \Leftrightarrow \quad p(X | Y, V) = p(X | V)$$

In general we can think of conditional independence between **sets of variables**:

$$\mathcal{X} \perp\!\!\!\perp \mathcal{Y} | \mathcal{V} \quad \Leftrightarrow \quad p(\mathcal{X}, \mathcal{Y} | \mathcal{V}) = p(\mathcal{X} | \mathcal{V})\, p(\mathcal{Y} | \mathcal{V})$$

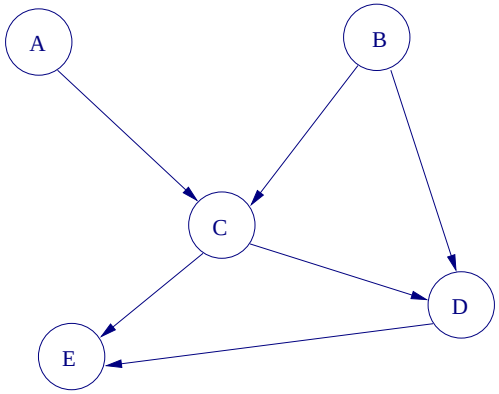
Conditional and Marginal Independence (Examples)

- Amount of Speeding Fine $\perp\!\!\!\perp$ Type of Car | Speed
- Lung Cancer $\perp\!\!\!\perp$ Yellow Teeth | Smoking
- (Position, Velocity)$_{t+1}$ $\perp\!\!\!\perp$ (Position, Velocity)$_{t-1}$ | (Position, Velocity)$_t$, Acceleration$_t$
- Child's Genes $\perp\!\!\!\perp$ Grandparents' Genes | Parents' Genes
- Ability of Team A $\perp\!\!\!\perp$ Ability of Team B
- not (Ability of Team A $\perp\!\!\!\perp$ Ability of Team B | Outcome of A vs B Game)

Directed Graphical Models

(aka Bayesian networks, belief networks, probabilistic directed acyclic graphs)

A directed acyclic graph (DAG) where each node is a random variable and the edges represent statistical dependencies between the variables.



The DAG represents a factorization of the joint probability.

$$p(A, B, C, D, E) = p(A)\ p(B)\ p(C|A, B)\ p(D|B, C)\ p(E|C, D)$$

Each variable is conditionally independent of its non-descendents given its parents.

We can distinguish between learning/inference/estimation of the model parameters and the model structure (i.e. conditional independence relationships).

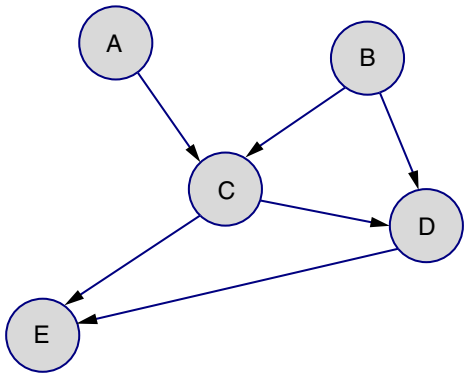
Some of the variables may be observed/measured, some may be hidden/latent/missing.

Model Selection for Discrete Graphs

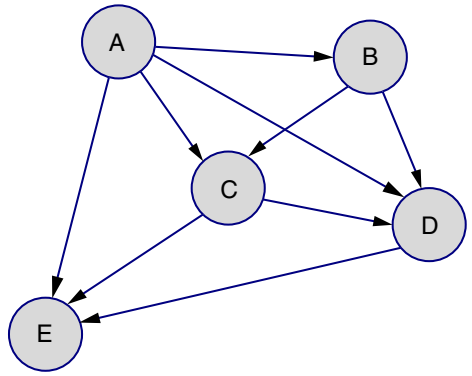
Which of the following graphical models is the data generating process?
Discrete directed acyclic graphical models: data $\mathbf{y} = (A, B, C, D, E)^n$

ALL OBSERVED

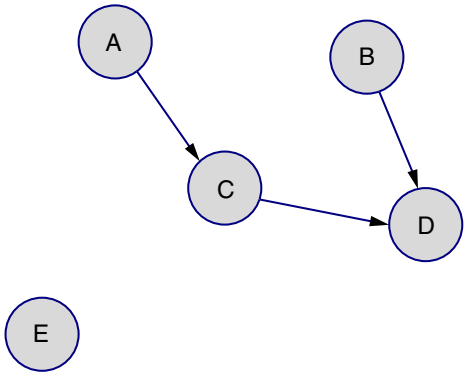
true



complex



simple

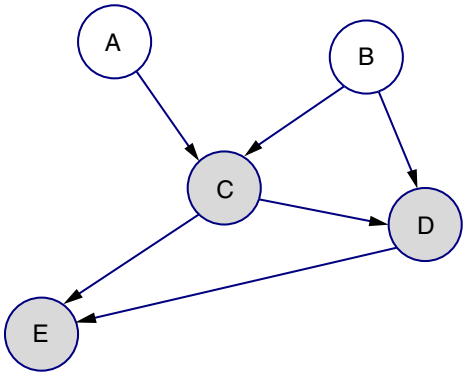


"easy"

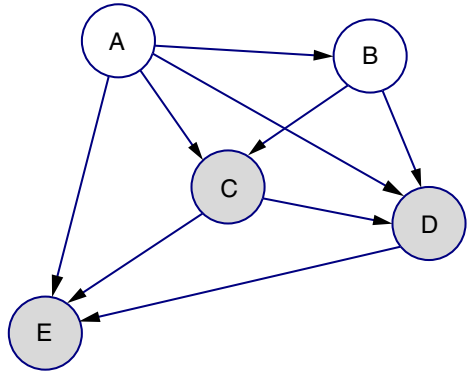
If the data are just $\mathbf{y} = (C, D, E)^n$, and $(A, B)^n$ are **hidden** variables... ?

OBS.+HIDDEN

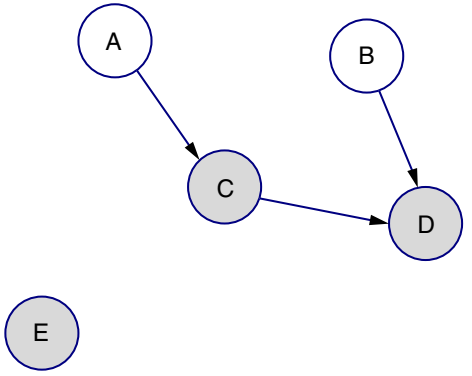
true



complex



simple



hard

Learning Model Structure using Marginal Likelihoods

Let m denote model structure, $\boldsymbol{\theta}$ denote model parameters, and $\mathbf{y}$ denote observed data.

We can compare model structures, m, based on their marginal likelihood given the observed data, $\mathbf{y}$:

$$p(\mathbf{y}|m) = \int p(\mathbf{y}|\boldsymbol{\theta}, m)\, p(\boldsymbol{\theta}|m)\, d\boldsymbol{\theta}, \qquad p(m|\mathbf{y}) = \frac{p(\mathbf{y}|m)p(m)}{p(\mathbf{y})}$$

Interpretation of the Marginal Likelihood: The probability of the data given the model structure averaging over all possible (unknown) parameter values. This *automatically* implements Occam's Razor within the Bayesian framework.

Computing Marginal Likelihoods can be Computationally Intractable

$$p(\mathbf{y}|m) = \int p(\mathbf{y}|\boldsymbol{\theta}, m)\, p(\boldsymbol{\theta}|m)\, d\boldsymbol{\theta}$$

- This can be a very complicated **high dimensional integral** over model parameters.

- The presence of **hidden/latent/missing variables**, $\mathbf{x}$, results in additional dimensions that need to be marginalized out.

$$p(\mathbf{y}|m) = \int \int p(\mathbf{y}, \mathbf{x}|\boldsymbol{\theta}, m)\, p(\boldsymbol{\theta}|m)\, d\mathbf{x}\, d\boldsymbol{\theta}$$

Practical Bayesian approaches

- Bayesian Information Criterion (e.g. BIC).

$$\log p(\mathbf{y}|m) \approx \log p(\mathbf{y}|\hat{\boldsymbol{\theta}}_{\mathrm{ML}}, m) - \frac{d}{2}\log N$$

- Laplace approximations:
  - Makes a Gaussian approximation about the posterior mode of the parameters.
- Markov chain Monte Carlo methods (MCMC):
  - converge to the desired distribution in the limit, but:
  - many samples are required to ensure accuracy.
  - sometimes hard to assess convergence and reliably compute marginal likelihood.
- Variational approximations...

Lower Bounding the Marginal Likelihood

Variational Bayesian Learning

Let the latent variables be $\mathbf{x}$, data $\mathbf{y}$ and the parameters $\boldsymbol{\theta}$.
We can lower bound the marginal likelihood (using Jensen's inequality):

$$\begin{aligned}
\ln p(\mathbf{y}|m) &= \ln \int p(\mathbf{y}, \mathbf{x}, \boldsymbol{\theta}|m)\, d\mathbf{x}\, d\boldsymbol{\theta} \\
&= \ln \int q(\mathbf{x}, \boldsymbol{\theta}) \frac{p(\mathbf{y}, \mathbf{x}, \boldsymbol{\theta}|m)}{q(\mathbf{x}, \boldsymbol{\theta})}\, d\mathbf{x}\, d\boldsymbol{\theta} \\
&\geq \int q(\mathbf{x}, \boldsymbol{\theta}) \ln \frac{p(\mathbf{y}, \mathbf{x}, \boldsymbol{\theta}|m)}{q(\mathbf{x}, \boldsymbol{\theta})}\, d\mathbf{x}\, d\boldsymbol{\theta}.
\end{aligned}$$

Use a simpler, factorised approximation to $q(\mathbf{x}, \boldsymbol{\theta}) \approx q_{\mathbf{x}}(\mathbf{x}) q_{\boldsymbol{\theta}}(\boldsymbol{\theta})$:

$$\begin{aligned}
\ln p(\mathbf{y}|m) &\geq \int q_{\mathbf{x}}(\mathbf{x}) q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) \ln \frac{p(\mathbf{y}, \mathbf{x}, \boldsymbol{\theta}|m)}{q_{\mathbf{x}}(\mathbf{x}) q_{\boldsymbol{\theta}}(\boldsymbol{\theta})}\, d\mathbf{x}\, d\boldsymbol{\theta} \\
&= \mathcal{F}_m(q_{\mathbf{x}}(\mathbf{x}), q_{\boldsymbol{\theta}}(\boldsymbol{\theta}), \mathbf{y}).
\end{aligned}$$

(some significant contributors to this framework from 1993-1998: C. M. Bishop, G. E. Hinton, T. S. Jaakkola, M. I. Jordan, D. J. C. MacKay, R. M. Neal, L. K. Saul)

Variational Bayesian Learning . . .

Maximization of this lower bound, $\mathcal{F}_m$, can be done via **EM-like** iterative updates:

$$q_{\mathbf{x}}^{(t+1)}(\mathbf{x}) \quad \propto \quad \exp\left[\int \ln p(\mathbf{x},\mathbf{y}|\boldsymbol{\theta},m)\, q_{\boldsymbol{\theta}}^{(t)}(\boldsymbol{\theta})\, d\boldsymbol{\theta}\right] \qquad \text{E-like step}$$

$$q_{\boldsymbol{\theta}}^{(t+1)}(\boldsymbol{\theta}) \quad \propto \quad p(\boldsymbol{\theta}|m)\ \exp\left[\int \ln p(\mathbf{x},\mathbf{y}|\boldsymbol{\theta},m)\, q_{\mathbf{x}}^{(t+1)}(\mathbf{x})\, d\mathbf{x}\right] \qquad \text{M-like step}$$

Maximizing $\mathcal{F}_m$ is equivalent to minimizing KL-divergence between the *approximate posterior*, $q_{\boldsymbol{\theta}}(\boldsymbol{\theta})\ q_{\mathbf{x}}(\mathbf{x})$ and the *true posterior*, $p(\boldsymbol{\theta},\mathbf{x}|\mathbf{y},m)$:

$$\ln p(\mathbf{y}|m) - \mathcal{F}_m(q_{\mathbf{x}}(\mathbf{x}), q_{\boldsymbol{\theta}}(\boldsymbol{\theta}), \mathbf{y}) = \int q_{\mathbf{x}}(\mathbf{x})\ q_{\boldsymbol{\theta}}(\boldsymbol{\theta}) \ln \frac{q_{\mathbf{x}}(\mathbf{x})\ q_{\boldsymbol{\theta}}(\boldsymbol{\theta})}{p(\boldsymbol{\theta},\mathbf{x}|\mathbf{y},m)}\, d\mathbf{x}\, d\boldsymbol{\theta} = \mathbf{KL}(q\|p)$$

The Variational Bayesian EM algorithm

EM for MAP estimation

Goal: maximize $p(\boldsymbol{\theta}|\mathbf{y}, m)$ w.r.t. $\boldsymbol{\theta}$

E Step: compute

$$q_{\mathbf{x}}^{(t+1)}(\mathbf{x}) = p(\mathbf{x}|\mathbf{y}, \boldsymbol{\theta}^{(t)})$$

M Step:

$$\boldsymbol{\theta}^{(t+1)} = \operatorname{argmax}_{\boldsymbol{\theta}} \int q_{\mathbf{x}}^{(t+1)}(\mathbf{x}) \ln p(\mathbf{x}, \mathbf{y}, \boldsymbol{\theta}) \, d\mathbf{x}$$

Variational Bayesian EM

Goal: approximate $p(\mathbf{y}|m)$, $p(\boldsymbol{\theta}|\mathbf{y}, m)$

VB-E Step: compute

$$q_{\mathbf{x}}^{(t+1)}(\mathbf{x}) = p(\mathbf{x}|\mathbf{y}, \bar{\boldsymbol{\phi}}^{(t)})$$

VB-M Step:

$$q_{\boldsymbol{\theta}}^{(t+1)}(\boldsymbol{\theta}) \propto \exp\left[\int q_{\mathbf{x}}^{(t+1)}(\mathbf{x}) \ln p(\mathbf{x}, \mathbf{y}, \boldsymbol{\theta}) \, d\mathbf{x}\right]$$

Properties:

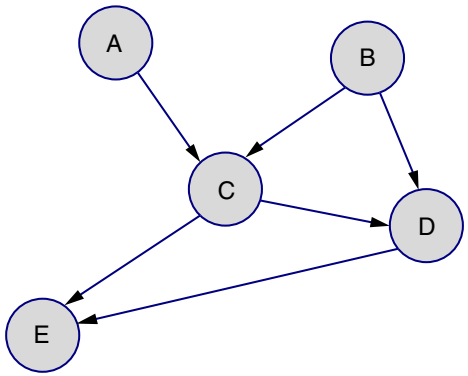
- Based on computing **expected natural parameters**, $\bar{\phi}$, under q.
- Reduces to the EM algorithm if we constrain $q_{\boldsymbol{\theta}}(\boldsymbol{\theta})$ to be a delta-function.
- $\mathcal{F}_m$ increases monotonically, and incorporates the model complexity penalty.
- Analytical parameter distributions (but not constrained to be Gaussian).
- VB-E step has same complexity as corresponding E step.

Model Selection for Discrete Graphs

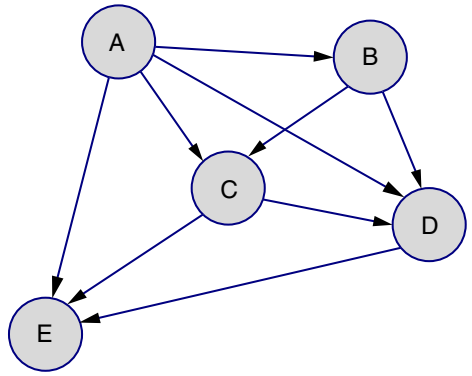
Which of the following graphical models is the data generating process?
Discrete directed acyclic graphical models: data $\mathbf{y} = (A, B, C, D, E)^n$

ALL OBSERVED

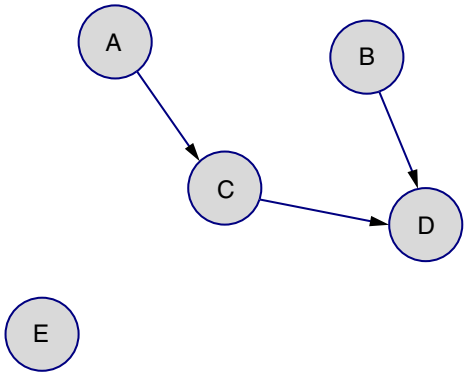
true



complex



simple

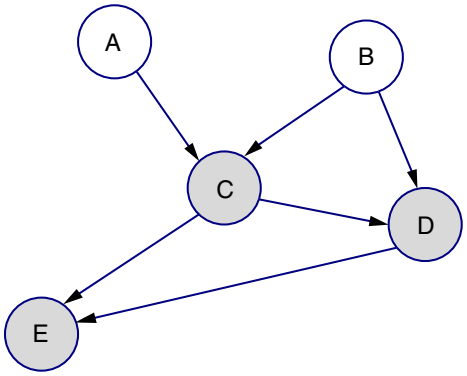


marginal likelihood tractable

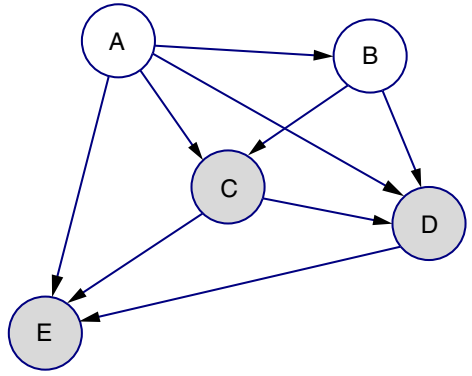
If the data are just $\mathbf{y} = (C, D, E)^n$, and $(A, B)^n$ are **hidden** variables... ?

OBS.+HIDDEN

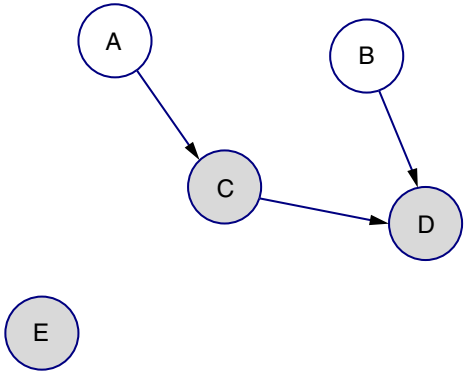
true



complex



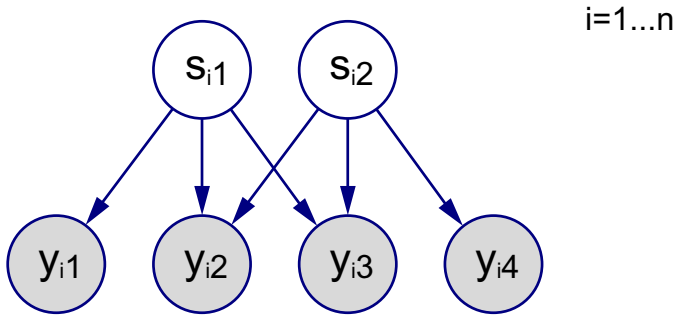
simple



marginal likelihood intractable

A case study for discrete directed graphs

- Bipartite structure: only hidden variables can be parents of observed variables.
- **Two** binary hidden variables, and **four** five-valued discrete observed variables.



- Conjugate prior is Dirichlet, Conjugate-Exponential model, so VB-EM algorithm is a straightforward modification of EM.

- **Experiment:** There are 136 distinct structures (out of 256) with 2 latent variables as potential parents of 4 conditionally independent observed vars.

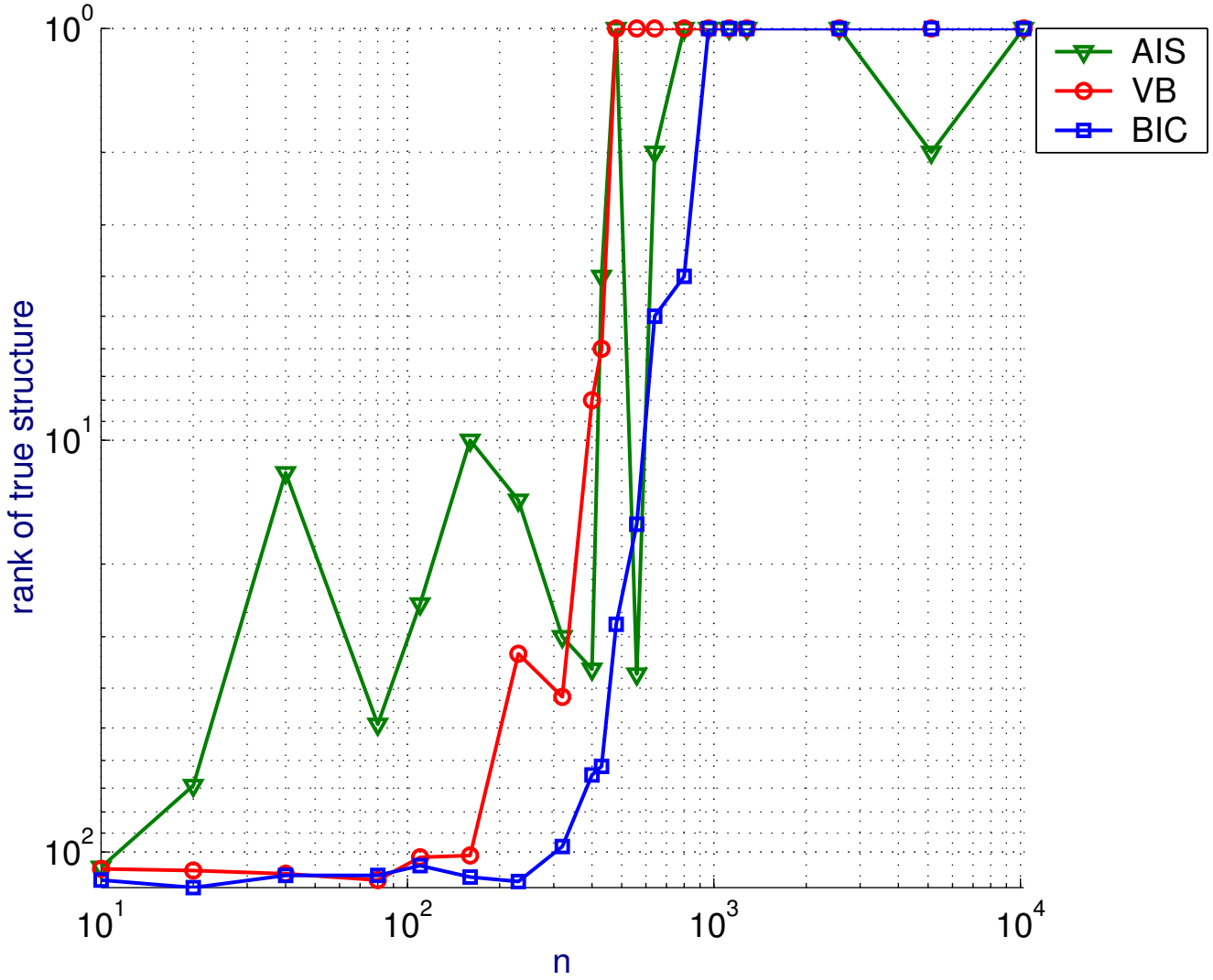
- **Score** each structure for twenty varying size data sets:

  $n \in \{10, 20, 40, 80, 110, 160, 230, 320, 400, 430, 480, 560, 640, 800, 960, 1120, 1280, 2560, 5120, 10240\}$

using 3 methods: BIC, VB, and a **gold standard** Annealed Importance Sampling AIS

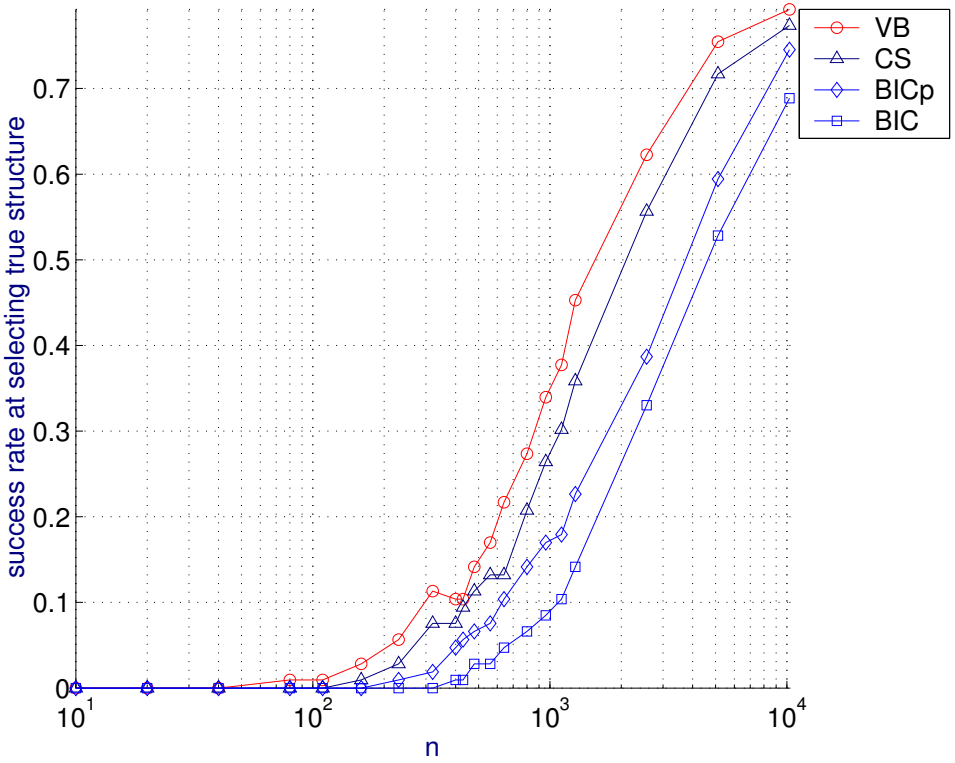
- 2720 graph scores computed, times for each: BIC (1.5s), VB (4s), AIS (400s).

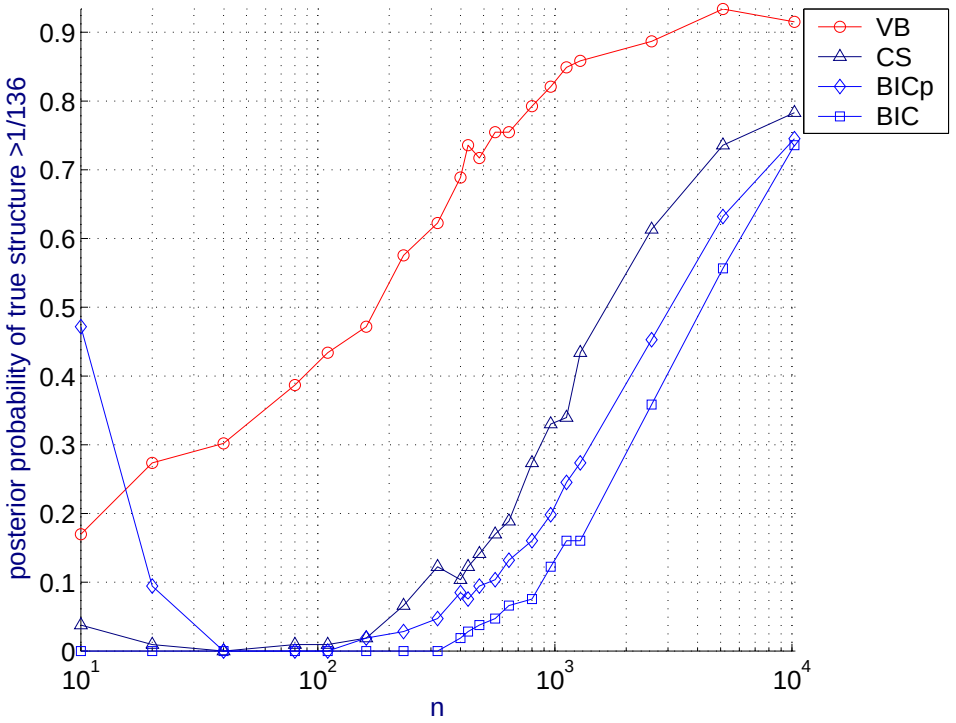
Results



VB score finds correct structure earlier, and more reliably than BIC.

Results, averaged over about 100 parameter draws



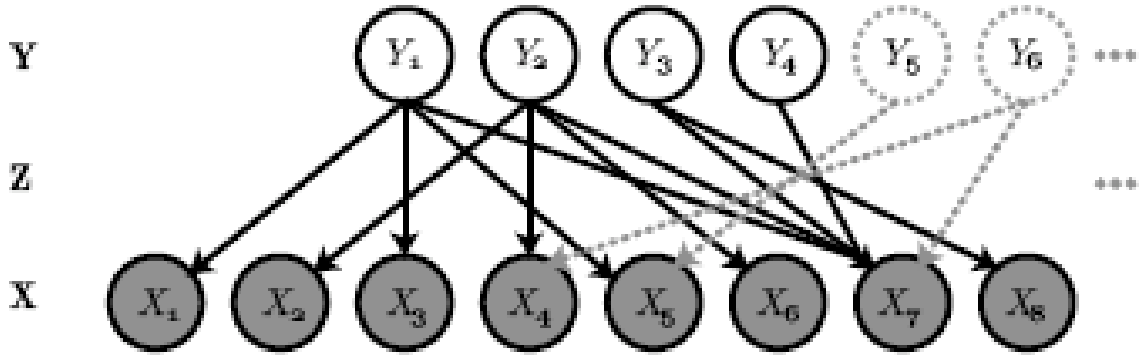


VB is also more accurate than Cheeseman-Stutz (CS) approximation to the marginal likelihood. In fact we can prove than VB $\geq$ CS (Beal and Ghahramani, *Bayesian Analysis*, 2006).

Summary of case study

- Learning structure of graphical models with latent variables is hard, but there are many ways of approximating marginal likelihoods.
- Variational Bayesian EM is a viable method for inference in conjugate exponential models.
- These methods have advantages over MCMC in that they can provide fast approximate Bayesian inference. Especially important in machine learning applications with large data sets.
- Results of case study:
  - VB is uniformly better than BIC and CS, at little computational cost
  - AIS is sometimes better than VB, but is sensitive to tuning parameters of MCMC, and about 100 times slower.

Part II: How many latent variables should there be?



$\mathbf{Y}$ - latent factors (e.g. diseases)
$\mathbf{Z}$ - graph structure (binary adjacency matrix)
$\mathbf{X}$ - observed binary features (e.g. symptoms)

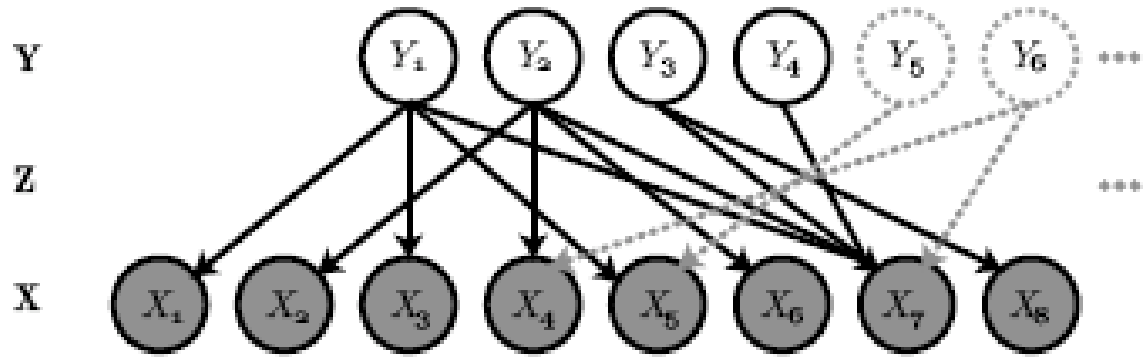
Solution 1: Do model comparison for $m = 1$, $m = 2$, ...

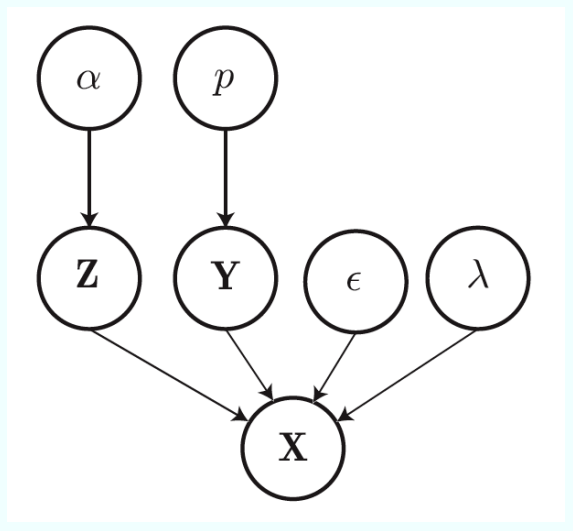
Solution 2: Assume potentially $m = \infty$ of which we only observe a finite number.

Note: this is analogous to the question of how many mixture components to use (model selection for finite mixture model vs infinite mixture model using Dirichlet processs mixtures).

Graphical models with infinitely many latent variables

"A Non-Parametric Bayesian Method for Inferring Hidden Causes" (Frank Wood, Tom Griffiths, & Ghahramani, *Uncertainty in Artificial Intelligence*, 2006)





$\mathbf{Y}$ - binary latent factors (diseases)
$\mathbf{Z}$ - graph structure
$\mathbf{X}$ - observed binary features (symptoms)

"Noisy-or" observations: $P(x_{it} = 1|\mathbf{Z}, \mathbf{Y}, \lambda, \epsilon) = 1 - (1 - \lambda)^{\sum_k z_{ik} y_{kt}}(1 - \epsilon)$

What should we use as $P(\mathbf{Z})$?

The matrix $\mathbf{Z}$ is a binary matrix of size (N = number of observed variables) $\times$ (K = number of latent variables).

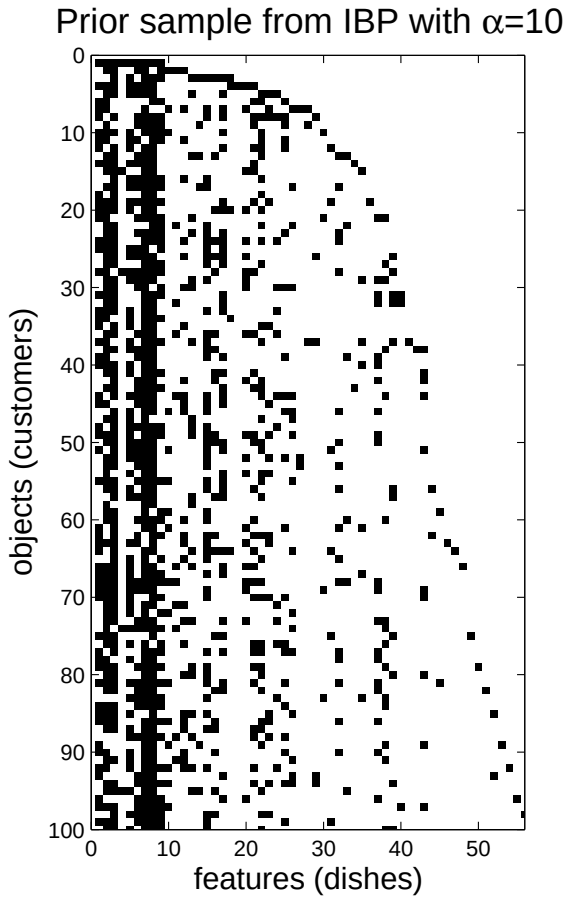
But $K \to \infty$.

We can define a consistent distribution over such infinite sparse binary matrices using the "Indian Buffet Process" (IBP) (cf Chinese restaurant process, Aldous 1985; Pitman 2002).

A sample from prior shown on right.

Note "rich get richer" property.

We can derive a Gibbs sampler for this model.



Graphical models with infinitely many latent variables

Gibbs sampling traces

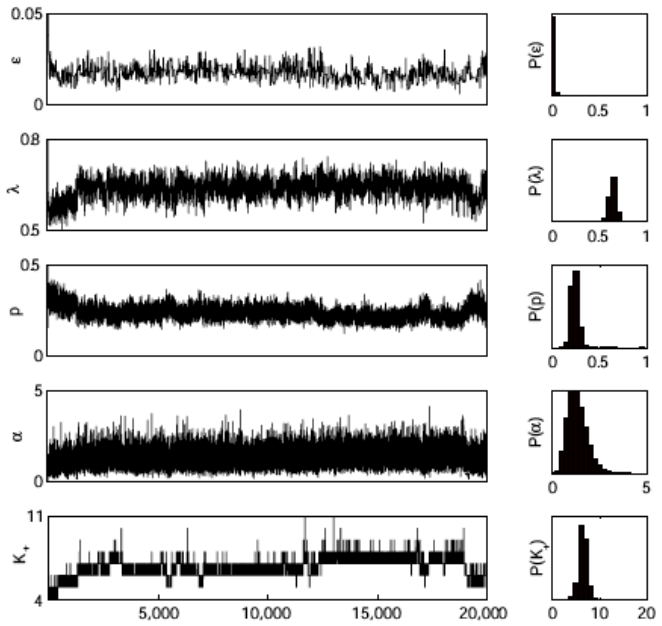


Figure 5: Trace plots and histograms for the Gibbs sampler applied to the signs exhibited by 50 stroke patients. The left column shows the current value of $\epsilon, \lambda, p, \alpha$, and K_+ as the sampler progressed, where K_+ is obtained by examining the current $\mathbf{Z}$ sample. The right column shows histograms of the same variables computed over the samples.

Comparison to RJMCMC

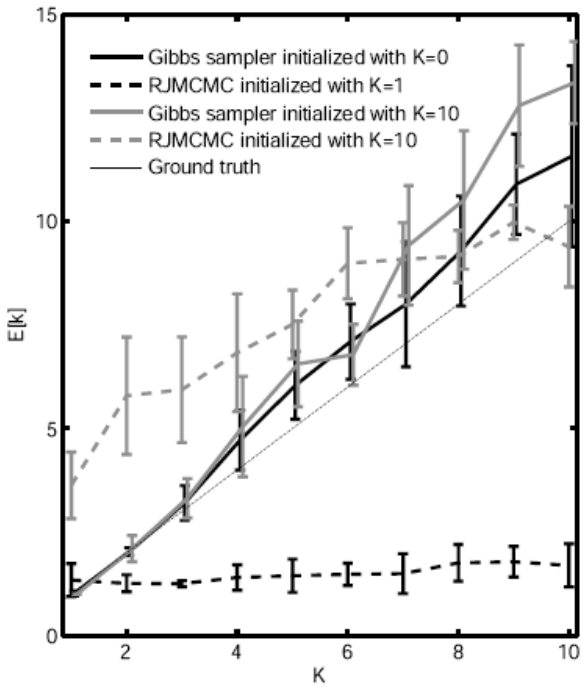


Figure 3: Learning the number of hidden causes using both RJMCMC and Gibbs sampling. Each line show the mean and standard deviation of the expected value of the dimensionality of the model (K for RJMCMC, and K_+ for Gibbs) taken over 500 iterations of sampling for each of 10 datasets.

Seems to work reliably, and mixed better than RJMCMC.

Graphical models with infinitely many latent variables

(with Frank Wood and Tom Griffiths)
Inferring stroke localization from patient symptoms:

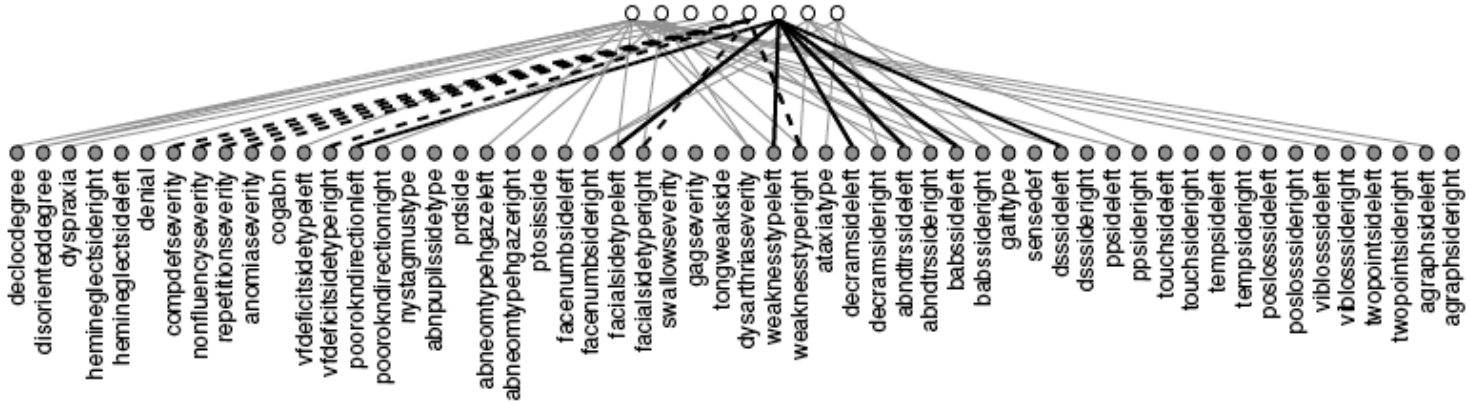


Figure 6: Causal structure with highest posterior probability. Two grouping of signs are highlighted. In solid black, we find a grouping of poor optokinetic nystagmus, lack of facial control, weakness, decreased rapid alternating movements, abnormal deep tendon reflexes, Babinski sign, and double simultaneous stimulation neglect, all on the left side, consistent with a right frontal/parietal infarct. In dashed black, we find a grouping of comprehension deficit, non-fluency, repetition, anomia, visual field deficit, facial weakness, and general weakness, with the latter three on the right side, generally consistent with a left temporal infarct.

(50 stroke patients, 56 symptoms/signs)

Summary of Part II

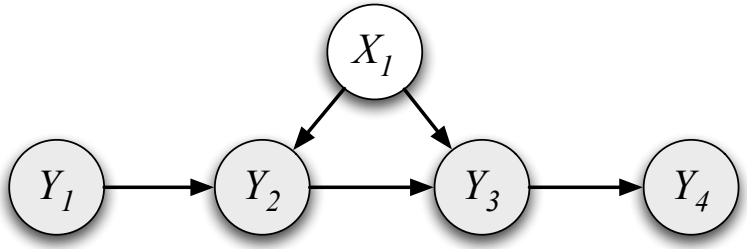
- It is possible to do inference and learning in graphical models with infinitely many latent variables.
- The graph structure can be parametrized using Indian Buffet Processes.
- Sampling from the distribution over structures in these models does not explicitly require approximating the marginal likelihood.

Part III

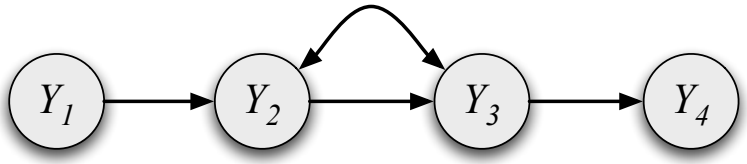
Just a pointer, since not much time available...

Goal: to learn a graph structure when we believe there are unobserved latents.

Approach 1: Explicitly represent these latents in a *directed graph* and learn the structure using one of the above methods.



Approach 2: Use a *different* graphical formalism which can represent all possible conditional independence relationships that arise from marginalizing out unobserved latents.[1]



Directed Mixed Graphs

[1] Note: DAGs cannot do this.

Part III b

- To do Bayesian inference in Directed Mixed Graphs we need priors over parameters that obey the implied conditional independence relationships.

- We have parameterized G-Inverse Wishart priors for the case of Gaussian DMGs such that
  - they obey the appropriate constraints
  - a simple and valid Gibbs sampling scheme can be devised.

- We have also defined a "Variational Monte Carlo" scheme which should be fast in high dimensions.

Silva, R. and Ghahramani, Z. (2006) Bayesian Inference for Gaussian Mixed Graph Models. *Uncertainty in Artificial Intelligence* (UAI-2006).

Summary: Graphs with Latent Variables

- Variational methods can be used to learn graphical model structure with latent variables.[2]

- It is possible to do inference in graphs with infinitely many latent variables.[3]

- Unobserved latents can lead to conditional independencies that are naturally represented as Directed Mixed Graphs; we have developed Bayesian inference procedures for the Gaussian and Discrete cases.[4]

http://learning.eng.cam.ac.uk/zoubin
zoubin@eng.cam.ac.uk

[2] Beal, M.J. and Ghahramani, Z. (2006) Variational Bayesian learning of directed graphical models with hidden variables. *Bayesian Analysis*. **1**:793–832.

[3] Wood, F., Griffiths, T.L. and Ghahramani, Z. (2006) A Non-Parametric Bayesian Method for Inferring Hidden Causes. In *Uncertainty in Artificial Intelligence (UAI-2006)*. 536–543

[4] Silva, R., and Ghahramani, Z. (2009) The Hidden Life of Latent Variables: Bayesian Learning with Mixed Graph Models. *Journal of Machine Learning Research* **10**(Jun):1187–1238.

Collaborators

Matthew Beal - University of Buffalo

Thomas Griffiths - UC Berkeley

Frank Wood - Brown University

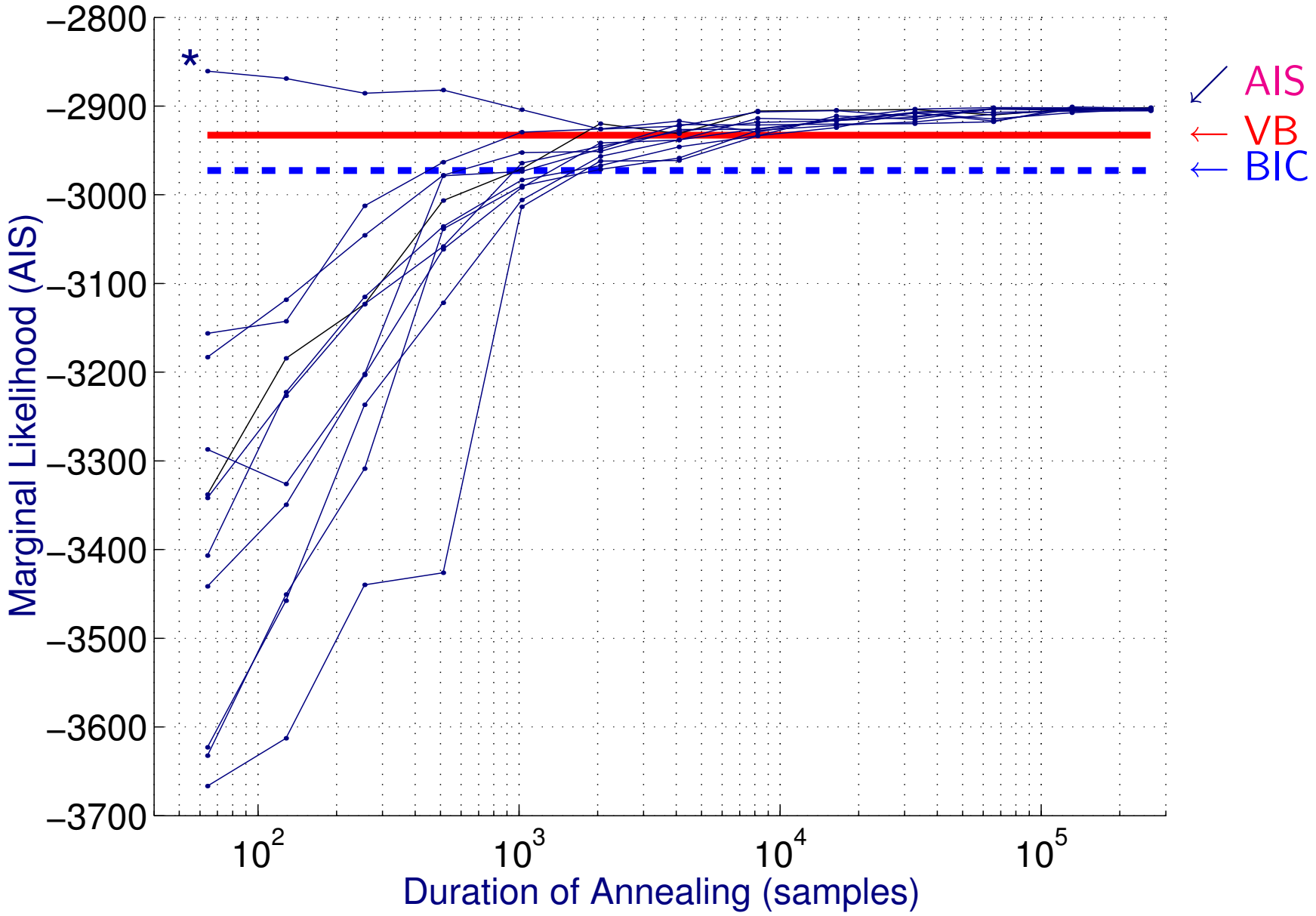
Ricardo Silva - University College London

Appendix

Conjugate Exponential Models

- If the joint probability of the hidden and observed data is in the exponential family
- ...and the prior over parameters is conjugate,
- ... then the variational Bayesian procedure becomes a **simple modification of EM**.
- This can actually include many interesting models (e.g. FA, SSM, HMM, MoG, DPM...)

How many samples of AIS are needed?



About 10^4 sweeps of sampling needed to achieve VB lower bound.