Factorial Mixture of Gaussians and the Marginal Independence Model

Ricardo Silva
Statistical Laboratory, University of Cambridge
silva@statslab.cam.ac.uk

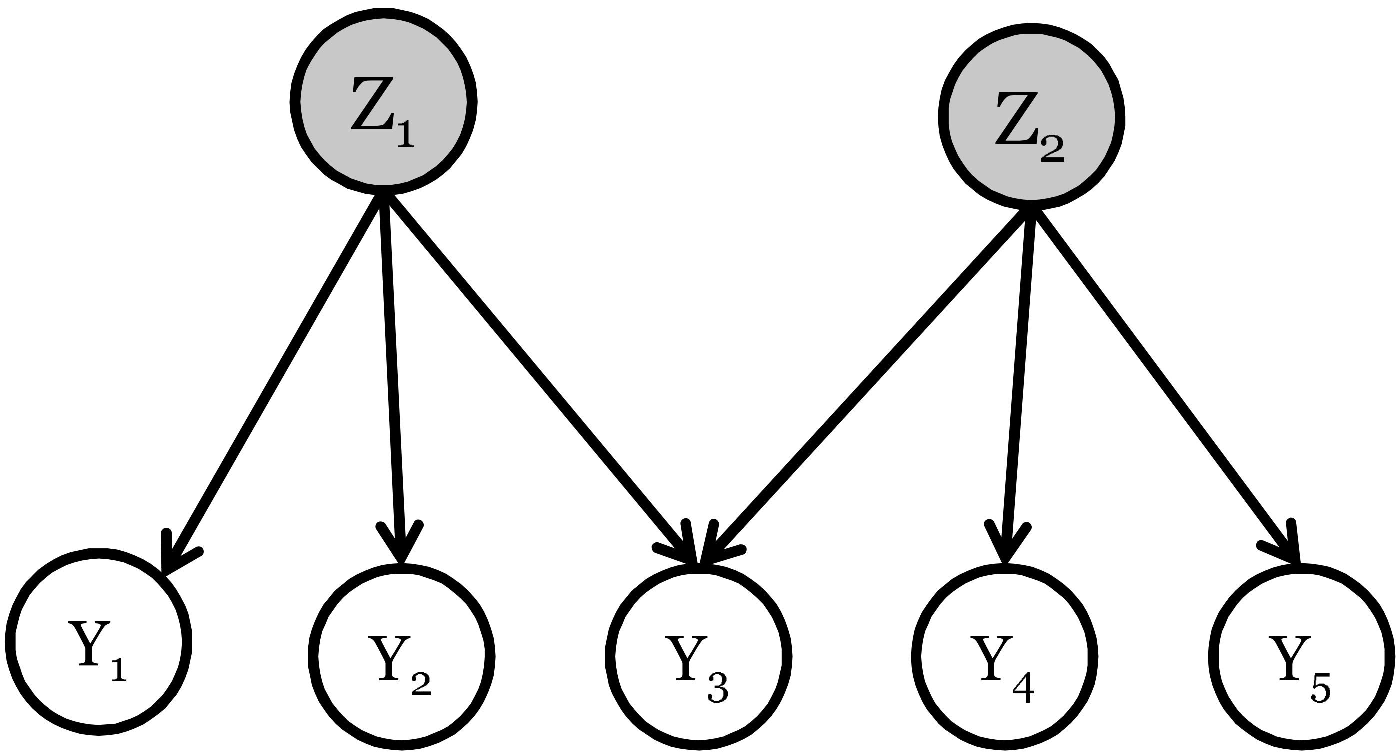
Joint work with Zoubin Ghahramani

Durham, July 2008

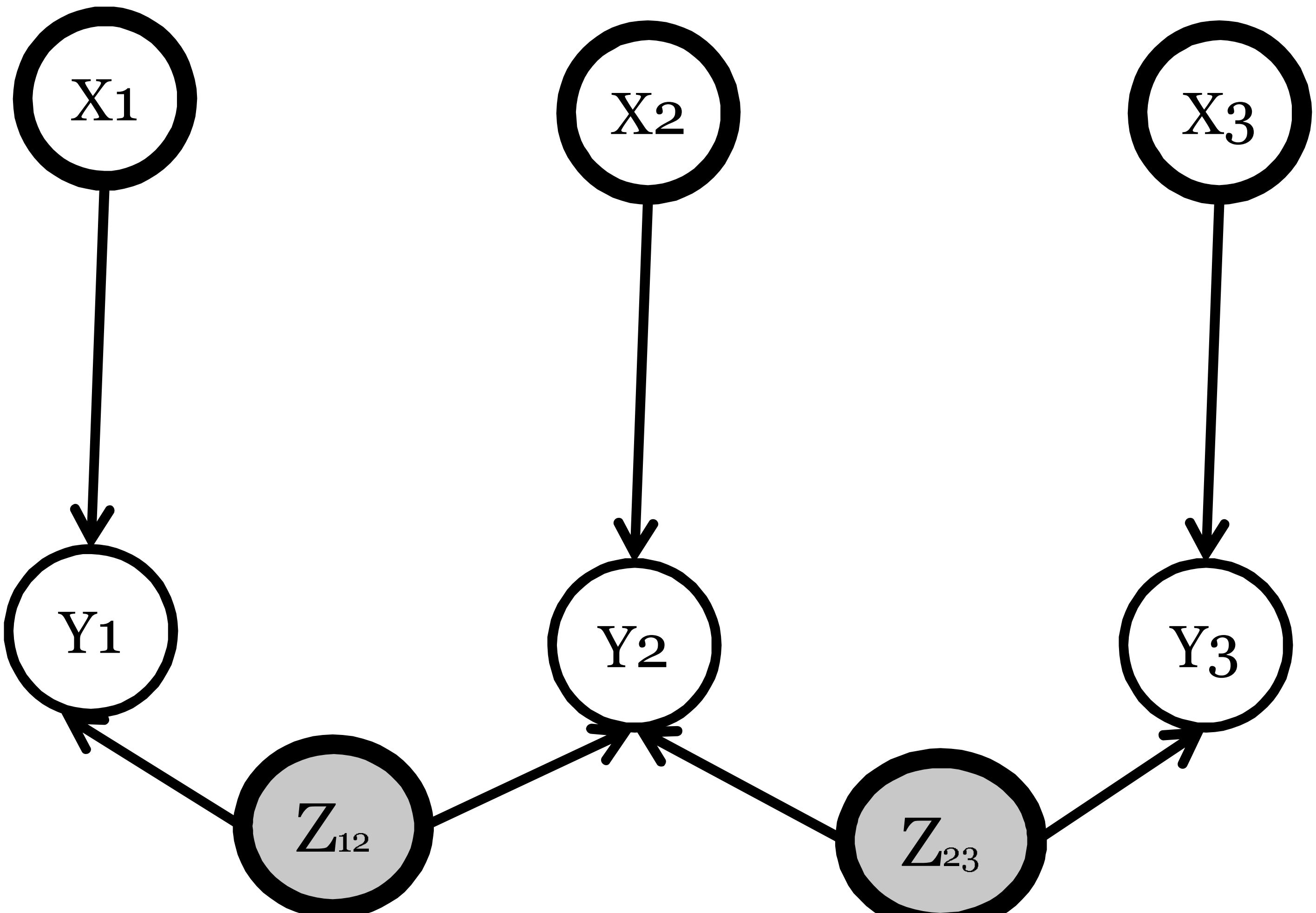
Goal

- To model sparse distributions subject to marginal independence constraints
- For continuous data

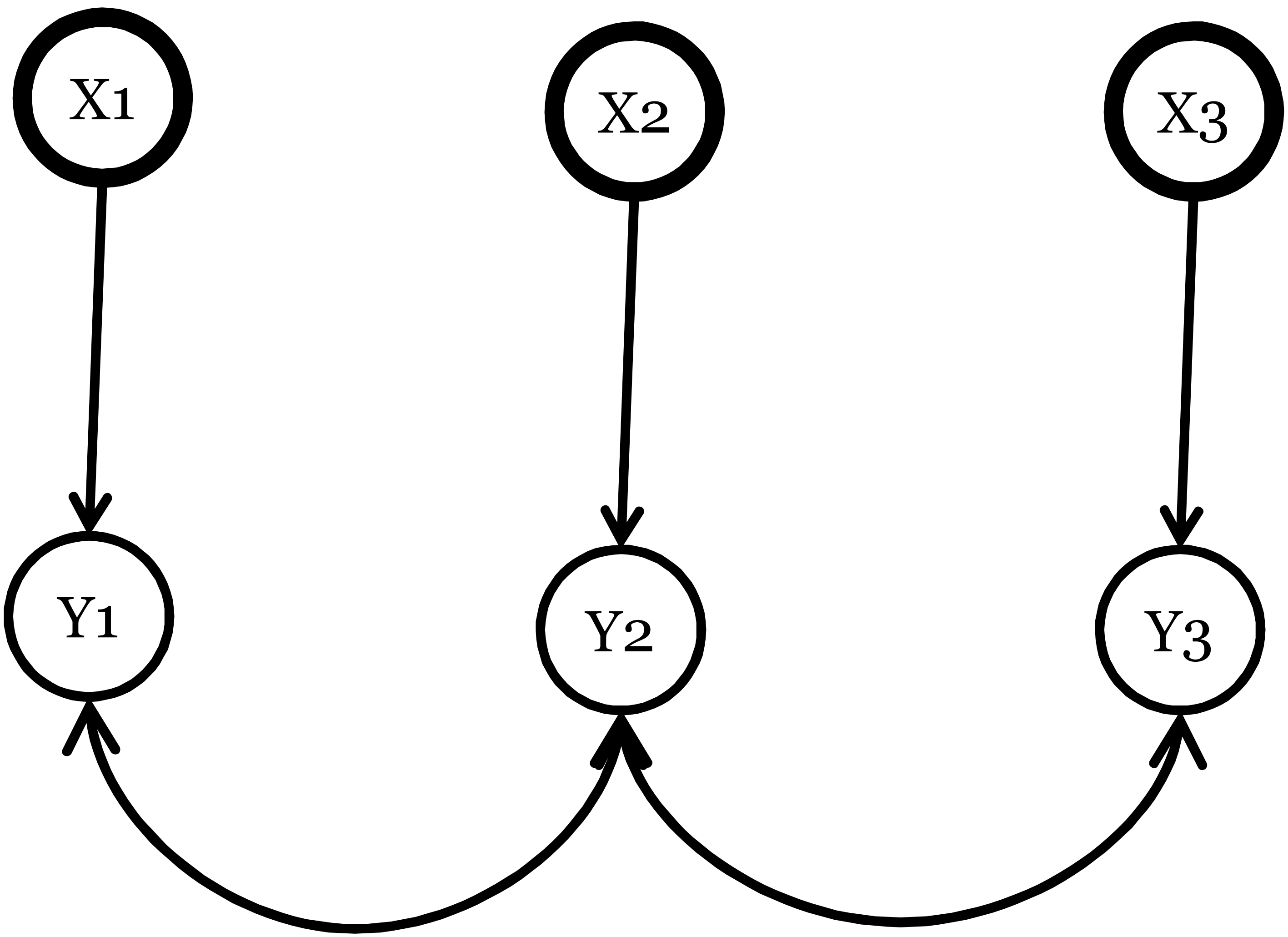
Why?



Why?



How?



General context

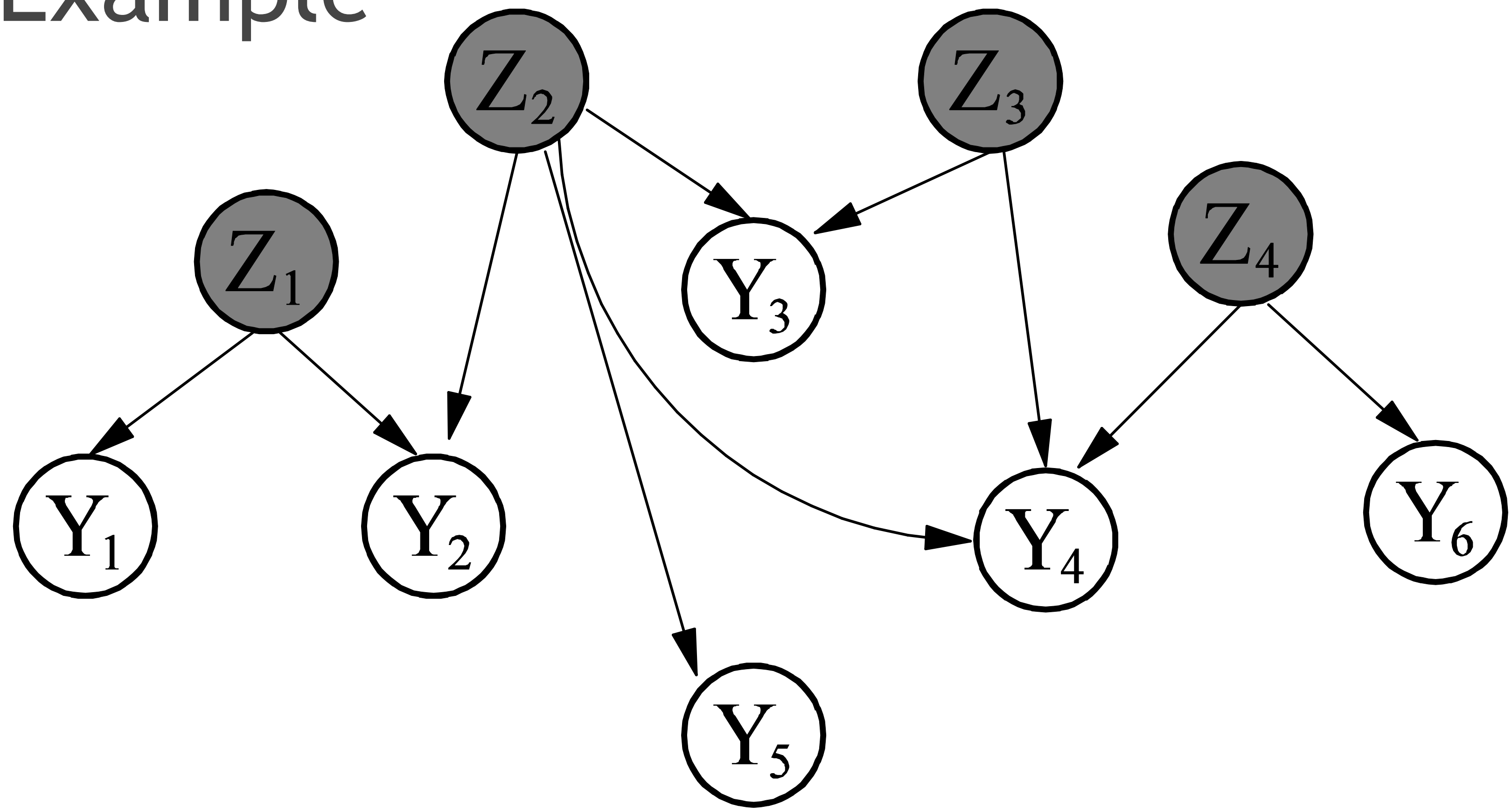
- $Y_i = f_i(X, Y) + E_i$, where E_i is an error term
- E is not a vector of independent variables
- Assumed: sparse structure of marginally dependent/independent variables
- Goal: estimating E-like distributions

Why not latent variable models?

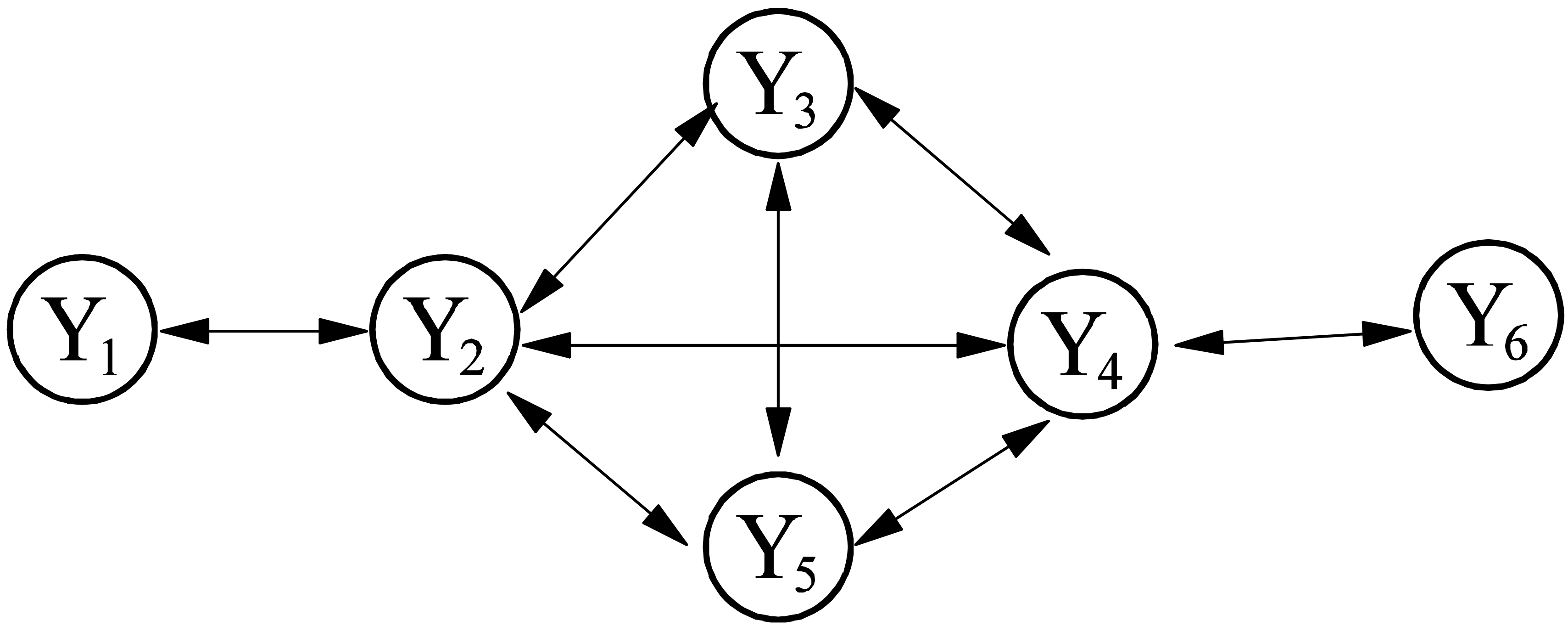
- Requires further decisions
  - How many latents? Which children?
  - Faces redundancy or overconstraining

- In the Bayesian case:
  - Punishes MCMC methods with (sometimes much) extra autocorrelation
  - Requires priors over parameters that you didn't even care in the first place

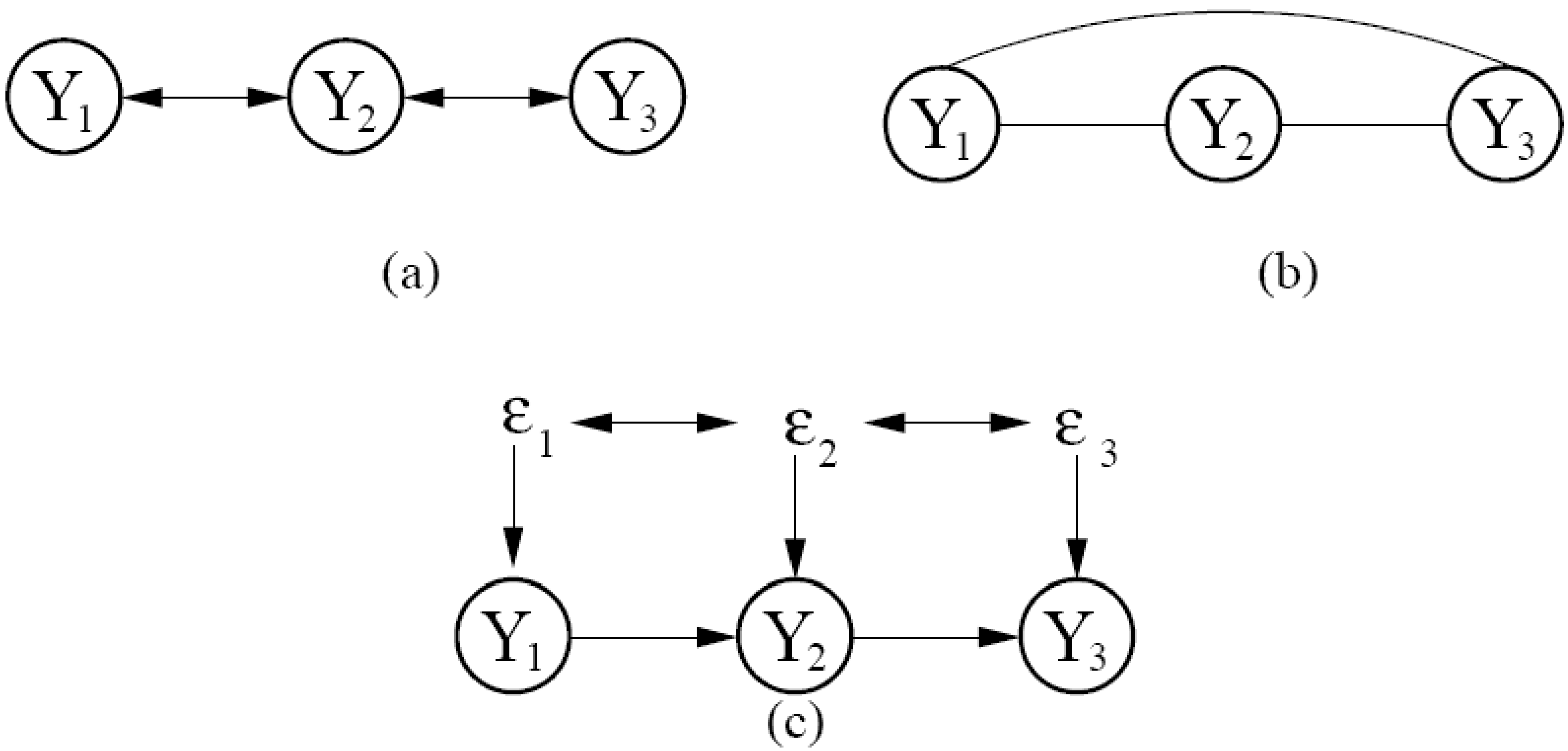
Example



Example



Example


Y_1
Y_2
Y_3
(a)
Y_1
Y_2
Y_3
(b)
ε_1
ε_2
ε_3
Y_1
Y_2
Y_3
(c)

Bi-directed models: The story so far

- Gaussian models
  - Maximum likelihood (Drton and Richardson, 2003)
  - Bayesian inference (Silva and Ghahramani, 2006, 2008)
- Binary models
  - Maximum likelihood (Drton and Richardson, 2008)

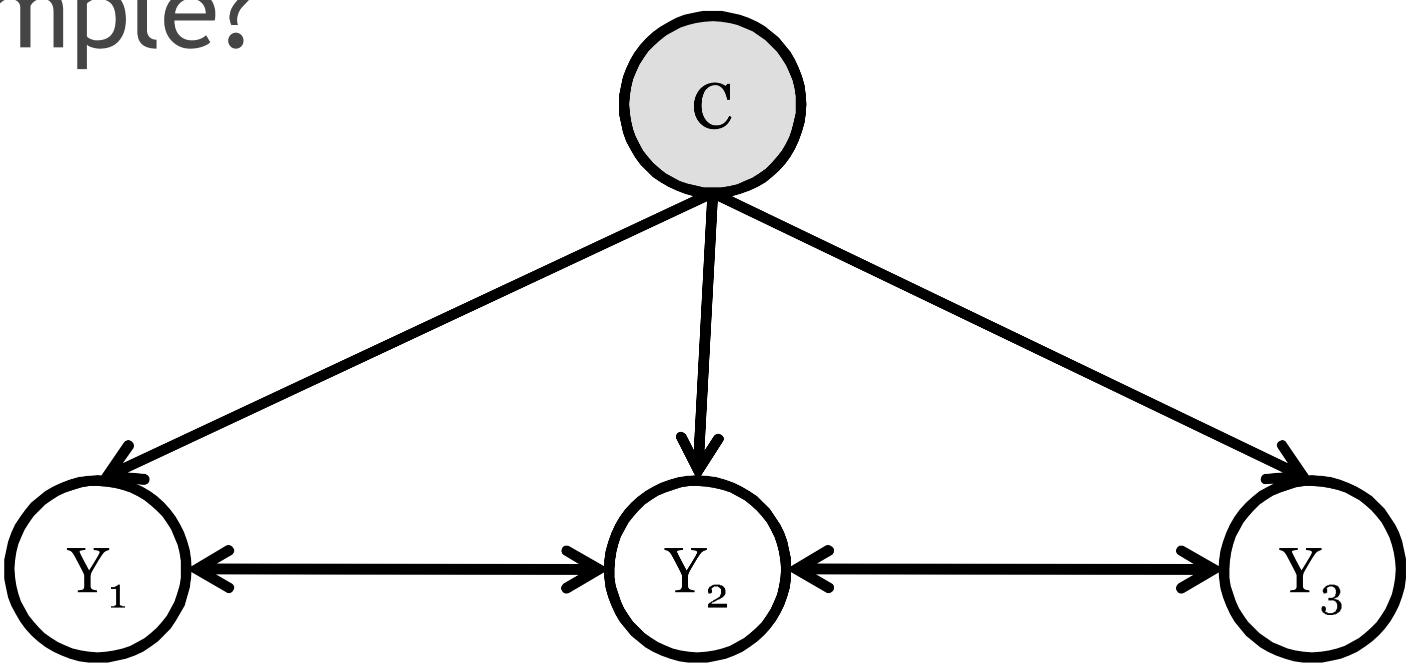
New model: mixture of Gaussians

- Latent variables: mixture indicators
  - Assumed #levels is decided somewhere else
- No “real” latent variables

Outline

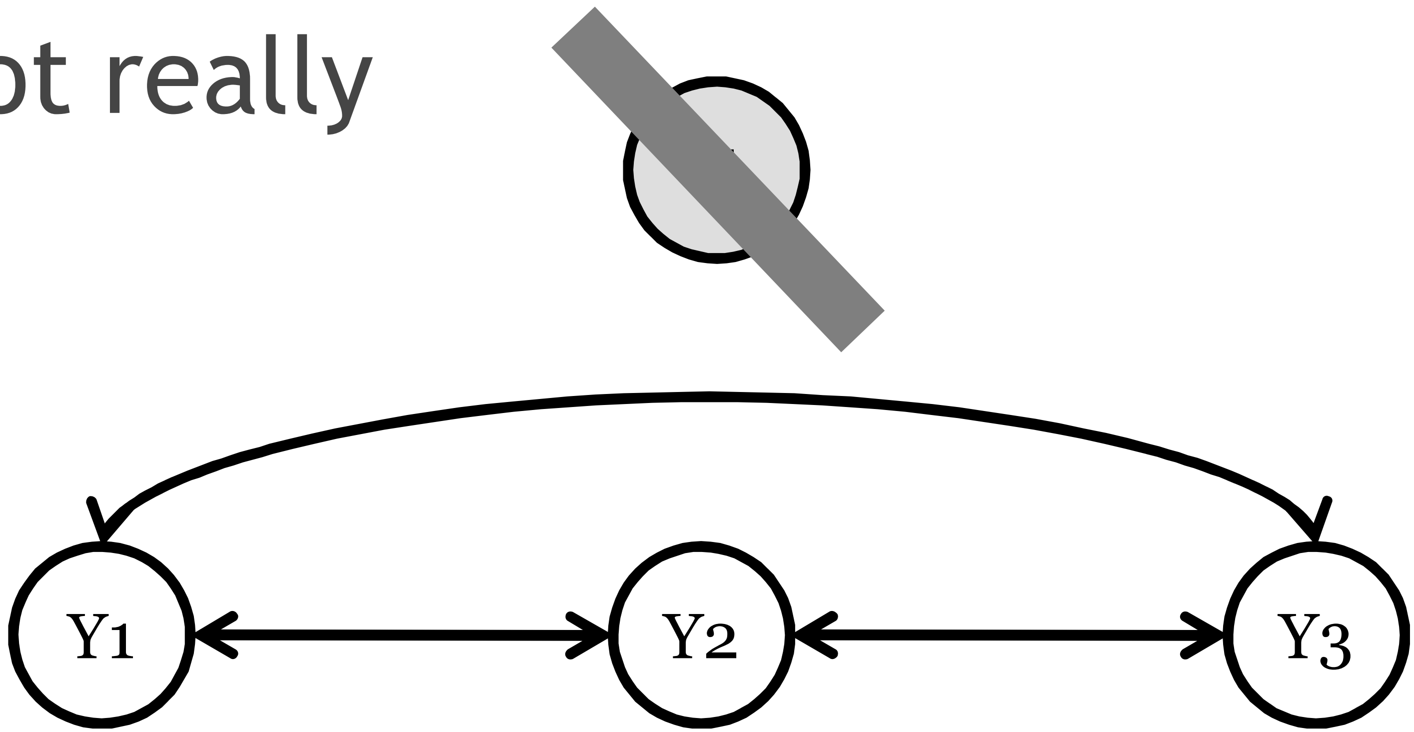
- We will focus on maximum likelihood and maximum a posteriori estimation
- Computationally hard even in sparse models
- Scalability will not be the focus here

Simple?

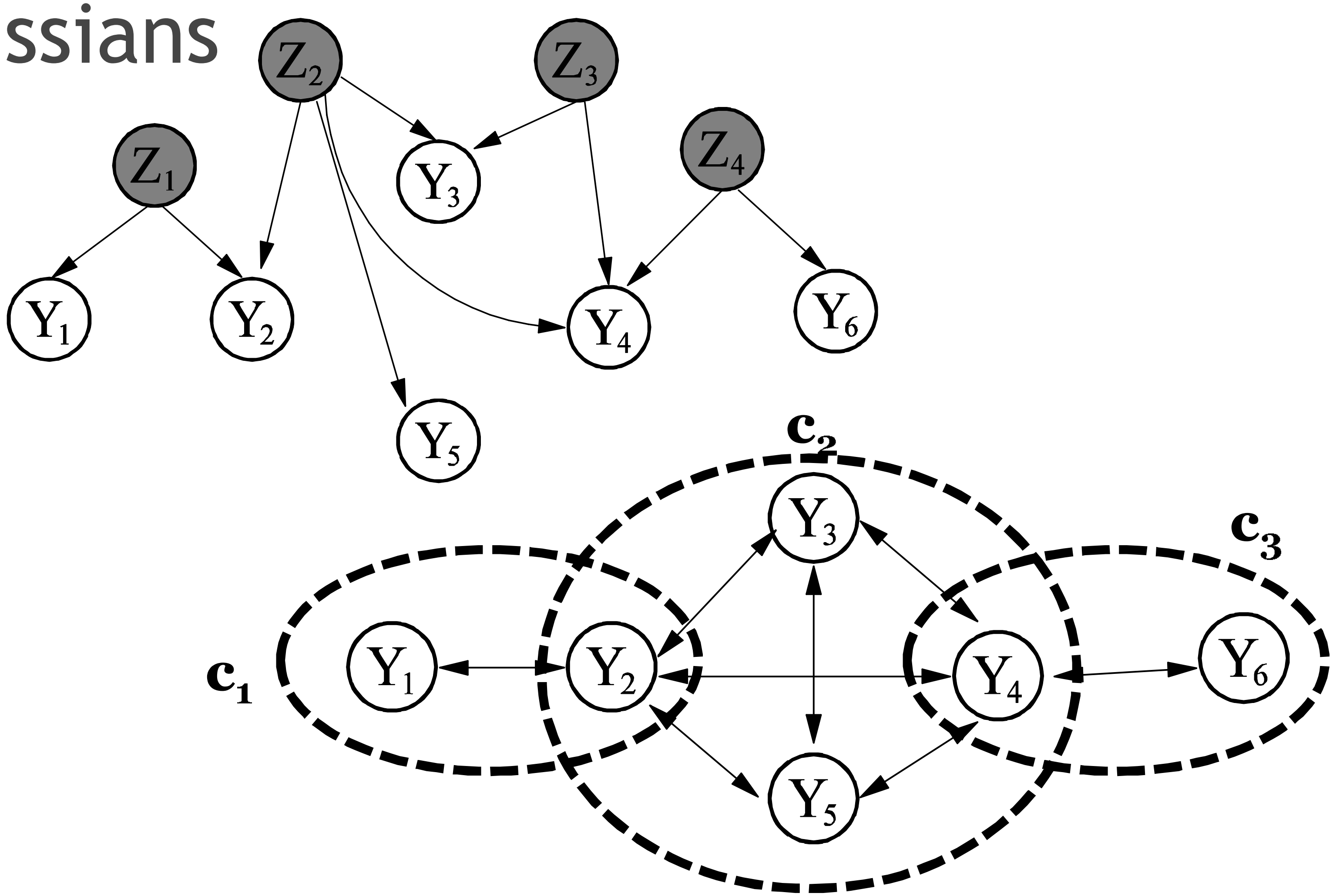


Y_1, Y_2, Y_3 jointly Gaussian with
sparse covariance matrix Σ_c indexed by C

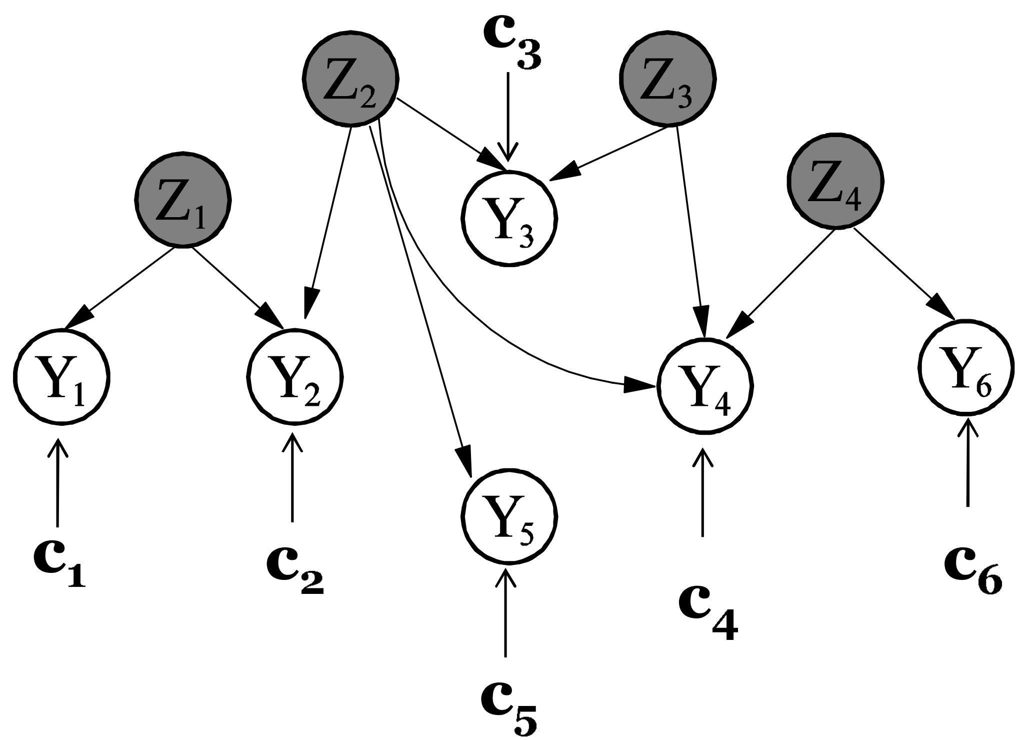
Not really



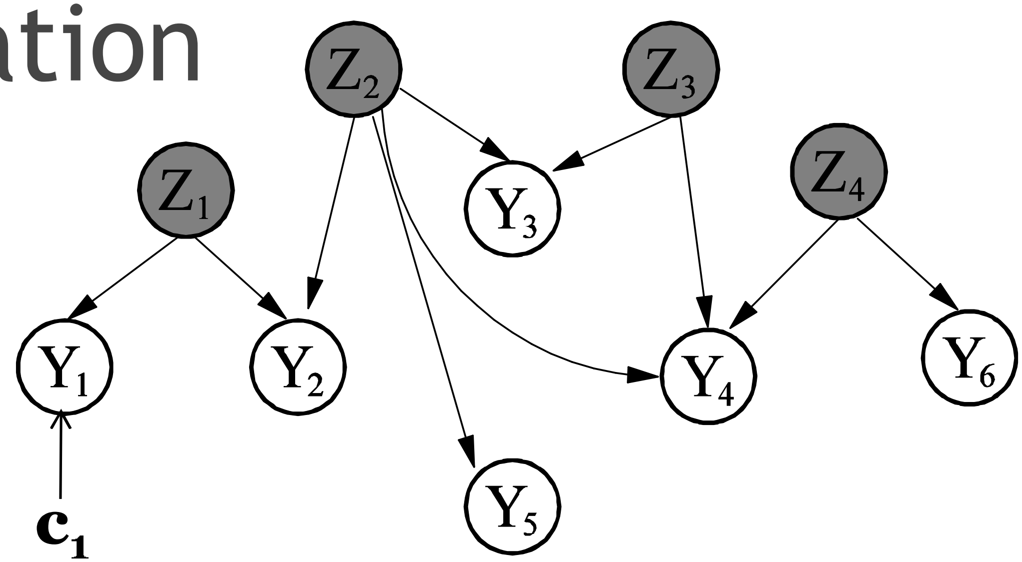
Required: a *factorial* mixture of Gaussians



Pairwise model



Parameterization

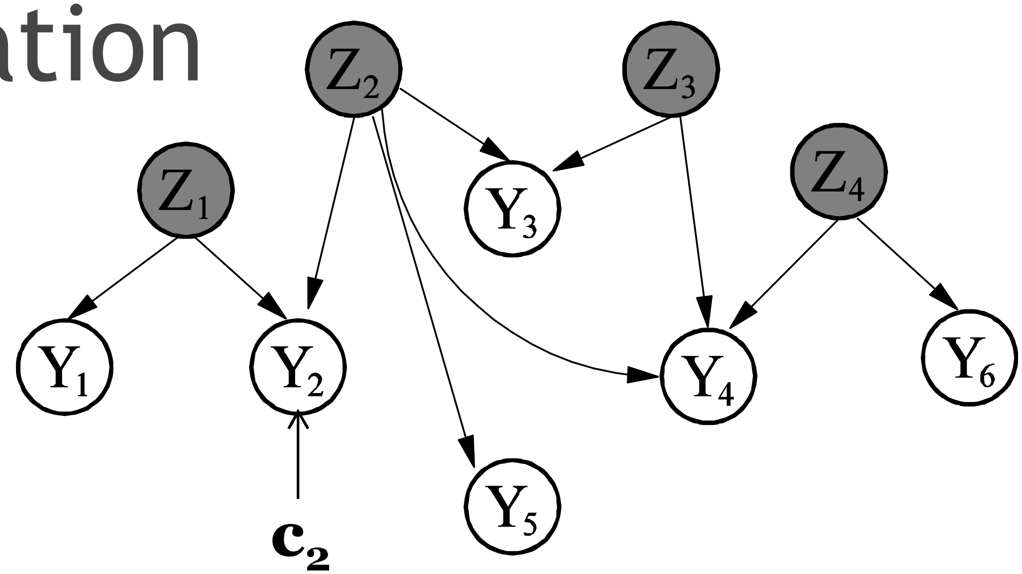


$$Y_1 = \lambda_{10}^{\mathbf{c}} + \lambda_{11}^{\mathbf{c}} Z_1 + \epsilon_1$$

$$\Lambda_1 \equiv \{\lambda_{10}^0, \lambda_{10}^1, \lambda_{11}^0, \lambda_{11}^1\} \text{ and variances } \{\upsilon_1^0, \upsilon_1^1\}.$$

Assume *Z* variables are zero-mean Gaussians, *c* variables are binary

Parameterization

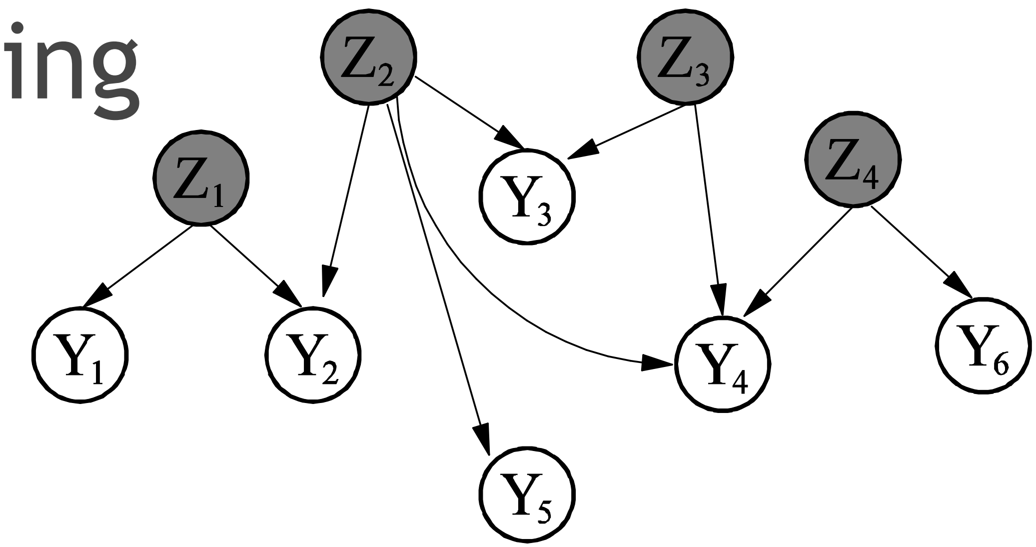


$$Y_2 = \lambda_{20}^{\mathbf{c}} + \lambda_{21}^{\mathbf{c}} Z_1 + \lambda_{22}^{\mathbf{c}} Z_2 + \epsilon_2$$

$$\Lambda_2 \equiv \{\lambda_{20}^0, \lambda_{20}^1, \lambda_{21}^0, \lambda_{21}^1, \lambda_{22}^0, \lambda_{22}^1\} \text{ and variances } \{\upsilon_2^0, \upsilon_2^1\}.$$

Assume *Z* variables are zero-mean Gaussians, *c* variables are binary

Implied indexing



$$\sigma_{ij}^{\mathbf{c}} = \sum_{v \in parents(Y_i, \mathcal{G}) \cap parents(Y_j, \mathcal{G})} \lambda_{iv}^{c_i} \lambda_{jv}^{c_j}$$

$$\mu_i^{\mathbf{c}} = \lambda_{i0}^{c_i}$$

Factorial mixture of Gaussians and the marginal independence model

- The general case for all latent structures

$$\mathbf{Y} \mid \mathbf{c} \sim \mathcal{N}(\mu^{\mathbf{c}}, \Sigma^{\mathbf{c}})$$

- Parameter pool:

$$\begin{aligned} \mu_i^{\mathbf{c}} &= \mu_i^{c_i} \\ \sigma_{ij}^{\mathbf{c}} &= \sigma_{ij}^{(c_i, c_j)} \end{aligned}$$

Size of the parameter space

- Let
  - m = number of edges
  - p = number of vertices
  - k = largest number of values among mixture indicators
- Total number of parameters:

$$O(mk^2 + pk)$$

- For $k = 2$, fewer parameters than a Gaussian if $m < O(p^2 / 8)$

Maximum likelihood estimation

- An EM framework

$$\mathcal{F}(\Theta;\mathcal{D}) = \sum_{i=1}^{n} -\frac{1}{2}\left\langle \log|\Sigma(\mathbf{c}^{(i)})| \right\rangle_{\pi'(\mathbf{c}^{(i)})} -\frac{1}{2}\left\langle (\mathbf{Y}^{(i)} - \mu^{\mathbf{c}^{(i)}})^T \Sigma(\mathbf{c}^{(i)})^{-1} (\mathbf{Y}^{(i)} - \mu^{\mathbf{c}^{(i)}}) \right\rangle_{\pi'(\mathbf{c}^{(i)})}$$

Maximum likelihood estimation

- An EM framework

$$\mathcal{F}(\Theta;\mathcal{D}) = \sum_{i=1}^{n} -\frac{1}{2}\left\langle \log|\Sigma(\mathbf{c}^{(i)})|\right\rangle_{\pi'(\mathbf{c}^{(i)})} -\frac{1}{2}\left\langle (\mathbf{Y}^{(i)} - \mu^{\mathbf{c}^{(i)}})^T \Sigma(\mathbf{c}^{(i)})^{-1} (\mathbf{Y}^{(i)} - \mu^{\mathbf{c}^{(i)}})\right\rangle_{\pi'(\mathbf{c}^{(i)})}$$

$$\forall \mathbf{c},\ \ \Sigma(\mathbf{c}) \text{ is positive definite}$$

Maximum a posteriori estimation

- A product of experts prior

$$f(\{\sigma_{ij}\}, \{\sigma_{ii}\}) \propto \prod_{ij;c} p_N(\sigma_{ij}^{(c_i,c_j)}; m, v) \prod_{i;c} p_G(\sigma_{ii}^{c_i}; \alpha, \beta) \times \mathcal{I}(\{\sigma_{ij}\}, \{\sigma_{ii}\})$$

- where
  - $p_N(\cdot)$ is a Gaussian density function
  - $p_G(\cdot)$ is an inverse gamma density function
  - $I(\cdot)$ is a indicator function (zero if some Σ^c not p.d.)

Algorithms

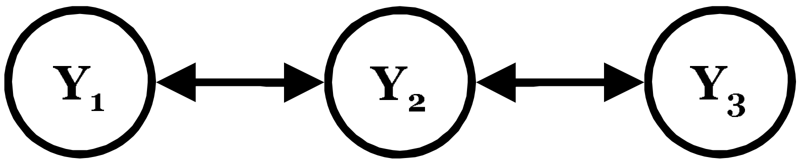
- Constraints
  - Positive definite constraints
  - Marginal independence constraints

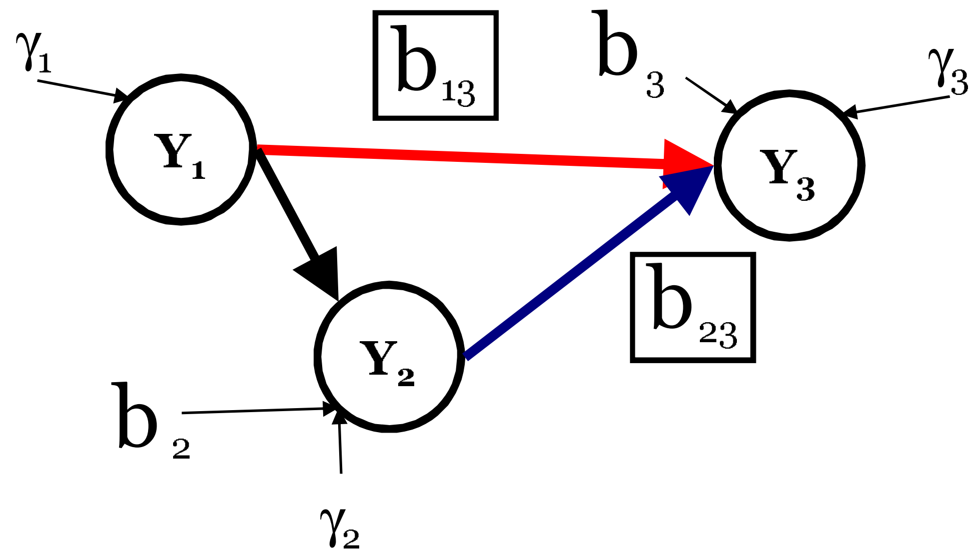
- Nonlinear optimization methods
  - Move over a subset of the parameter space while fixing the rest

Iterative conditional fitting: Gaussian case (Drton and Richardson, 2003)

- Choose some $Y_i \in \mathbf{Y}$
- Fix the covariance of $Y_{\setminus i} \equiv \mathbf{Y} \setminus Y_i$
- Fit the covariance of Y_i with $Y_{\setminus i}$, and its variance
- Marginal independence constraints introduced directly

Gaussian ICF

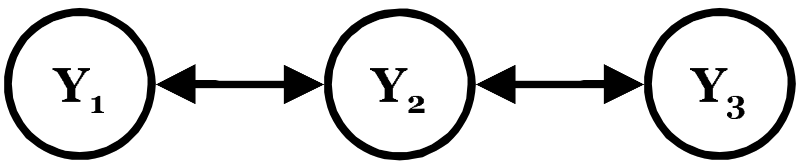


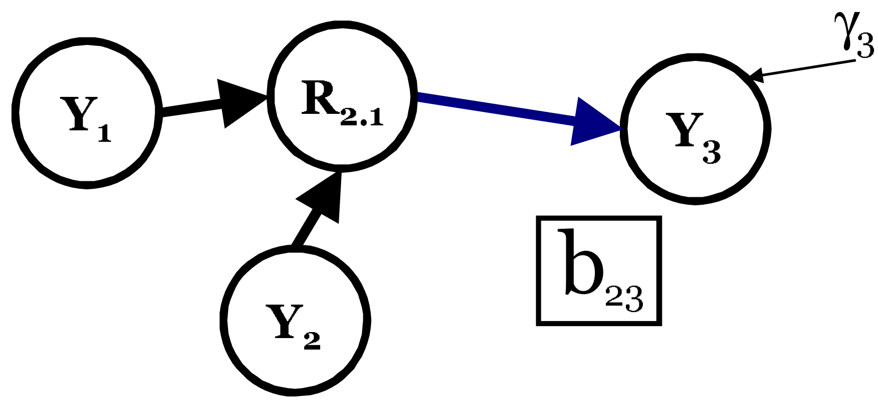


$$\Sigma_{12} b_3 = \Sigma_{3,12} \Rightarrow \begin{bmatrix}\sigma_{11} & \sigma_{12}\\ \sigma_{12} & \sigma_{22}\end{bmatrix}\begin{bmatrix} b_{13}\\ b_{23}\end{bmatrix} = \begin{bmatrix}\sigma_{31}\\ \sigma_{32}\end{bmatrix} = \begin{bmatrix}0\\ \sigma_{32}\end{bmatrix}$$

$$\sigma_{11} b_{13} + \sigma_{12} b_{23} = 0 \Rightarrow b_{13} = f(b_{23}, \Sigma_{12})$$

Gaussian ICF



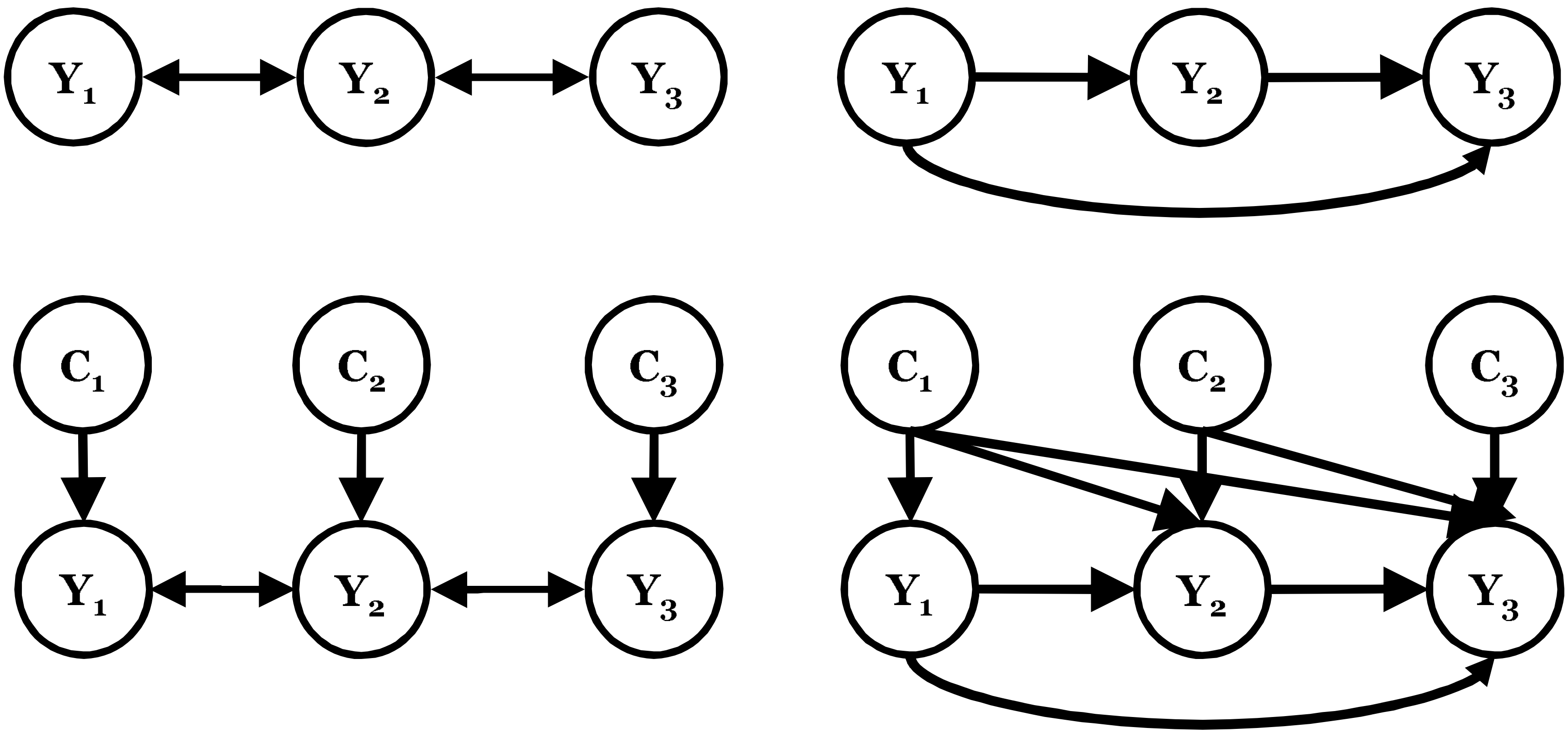


$$Y_3 = b_{23}R_{2.1} + \zeta_3,$$

where $R_{2.1}$ is the residual of the regression of Y_2 on Y_1

$$Y_i |\, \mathbf{Y}_{\setminus i} = \sum_{Y_j \text{ adjacent to } Y_i} b_{ij} R_j + \zeta_i$$

How does it change in the mixture of Gaussians case?



Parameter expansion

$$Y_i \mid \{\mathbf{c},\ \mathbf{Y}_{\setminus i}\} = \sum_{Y_j\ adjacent\ to\ Y_i} b_{ij}^{\mathbf{c}} R_j^{\mathbf{c}} + \zeta_i^{\mathbf{c}}$$

- Positive-definite constraints automatically satisfied
- No free lunch: an exponential number of parameters
  - Which means an exponential number of equality constraints

Parameter constraints

$$\sigma_{ij}^{\mathbf{c}} = \Sigma_{R,j}^{\mathbf{c}} b_i^{\mathbf{c}}, \quad \sigma_{ii}^{\mathbf{c}} = \gamma_i^{\mathbf{c}} + {b_i^{\mathbf{c}}}^T \Sigma_R^{\mathbf{c}} b_i^{\mathbf{c}}$$

$$\sigma_{ij}^{\mathbf{c}} = \sigma_{ij}^{\mathbf{c}'}, \text{if } c_i = c_i' \text{ and } c_j = c_j'$$

$$\sigma_{ii}^{\mathbf{c}} = \sigma_{ii}^{\mathbf{c}'}, \text{if } c_i = c_i'$$

Quadratic constraints

$$\gamma_i^{\mathbf{c}} + b_i^{\mathbf{c}T} \Sigma_R^{\mathbf{c}} b_i^{\mathbf{c}} = \gamma_i^{\mathbf{c}'} + b_i^{\mathbf{c}'T} \Sigma_R^{\mathbf{c}'} b_i^{\mathbf{c}'}, \text{if } c_i = c_i'$$

- Density function $f(Y_i \mid \mathbf{c}, \mathbf{Y}_{\backslash i})$ is not convex in γ and b

A relaxation

- Fix all γ
- Ignore variance (quadratic) constraints
- Optimize conditional (penalized) expected log-likelihood for *b*, given only linear constraints

$$\sigma_{ij}^{\mathbf{c}} = \sigma_{ij}^{\mathbf{c}'}, \text{if } c_i = c_i' \text{ and } c_j = c_j'$$

- "Doable" in closed formula
- Then optimize for γ with fixed *b*
  - Non-linear program, linear constraints only

Projecting back

- Before optimizing for γ, must guarantee feasible point
- For each value v of c_i, choose the instantiation $\mathbf{c}$ such that ${b_i^{\mathbf{c}}}^T \Sigma_R^{\mathbf{c}} b_i^{\mathbf{c}}$ is maximal, since

$$\gamma_i^{\mathbf{c}'} = \gamma_i^{\mathbf{c}} + {b_i^{\mathbf{c}}}^T \Sigma_R^{\mathbf{c}} b_i^{\mathbf{c}} - {b_i^{\mathbf{c}'}}^T \Sigma_R^{\mathbf{c}'} b_i^{\mathbf{c}'}, \text{ for } c_i = c_i' = v$$

will always be positive (necessary and sufficient condition)

Caveat emptor

- Overall method not guaranteed to always increase expected log-likelihood
  - I still found it to be very useful in practice
- In my implementation, I switch to a constrained non-linear optimizer when this happens
  - fmincon (MATLAB)

Recap

- Iterative conditional fitting: maximize expected conditional log-likelihood
- Transform to other parameter space
  - Exact algorithm: quadratic constraints, non-convex program
  - Relaxed algorithm:
    - only linear constraints
    - requires iterative method only for the (small) set of residual variances γ
      - size of independent γ = cardinality of c_i

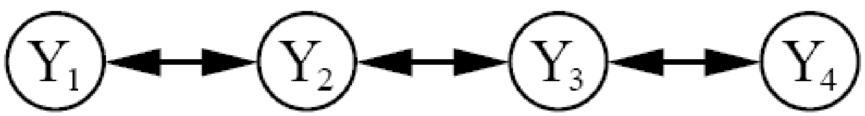
Approximations

- Taking expectations is expensive what to do?

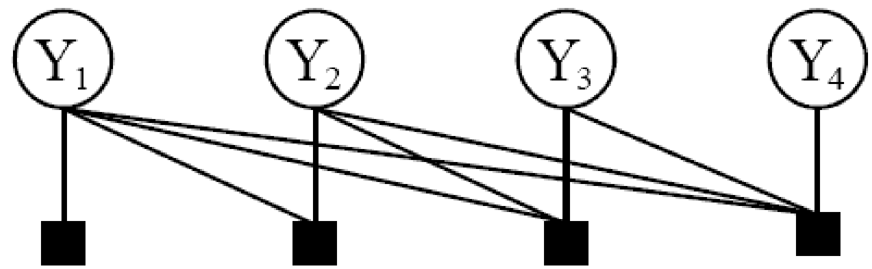
$$\mathcal{F}(\Theta;\mathcal{D}) = \sum_{i=1}^{n} -\frac{1}{2}\left\langle \log|\Sigma(\mathbf{c}^{(i)})| \right\rangle_{\pi'(\mathbf{c}^{(i)})} - \frac{1}{2}\left\langle (\mathbf{Y}^{(i)} - \mu^{\mathbf{c}^{(i)}})^T \Sigma(\mathbf{c}^{(i)})^{-1} (\mathbf{Y}^{(i)} - \mu^{\mathbf{c}^{(i)}}) \right\rangle_{\pi'(\mathbf{c}^{(i)})}$$

- Standard approximations use a “nice” π’(c)
  - E.g., mean-field methods (as in variational EM)
- Not enough!

Approximations: message-passing for free energy minimization?



(a)



(b)

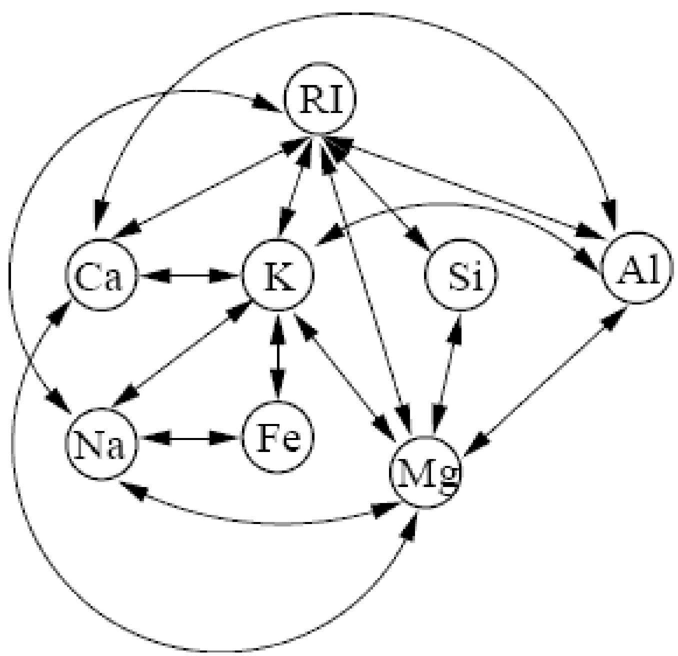
A simple approach?

- The Budgeted Variational Approximation
- As simple as it gets: maximize a variational bound forcing most combinations of c to give a zero value to $\pi'(c)$
  - Up to a pre-fixed budget
- How to choose which values?
- This guarantees positive-definitess only of those $\Sigma(c)$ with non-zero conditionals $\pi'(c)$
  - For predictions, project matrices first into PD cone

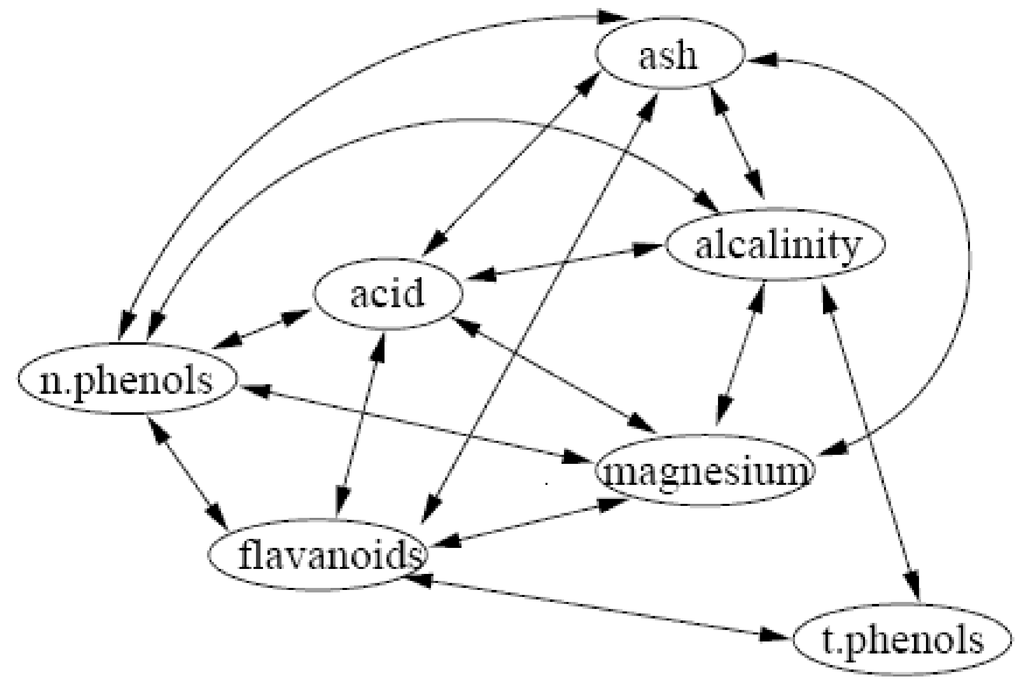
Experiments

- Some experiments evaluating predictive log-likelihood in test sets (UCI datasets)
- 5-fold cross-validation
- Learn structure by non-parametric tests of marginal independence (Gretton et al., 2007)
- Compare against latent variable models
  - For each clique in the bi-directed graph, introduce a latent, make it parent of the corresponding observed nodes

Experiment I (graph examples)



"Glass" dataset



"Wine" dataset

Experiment I

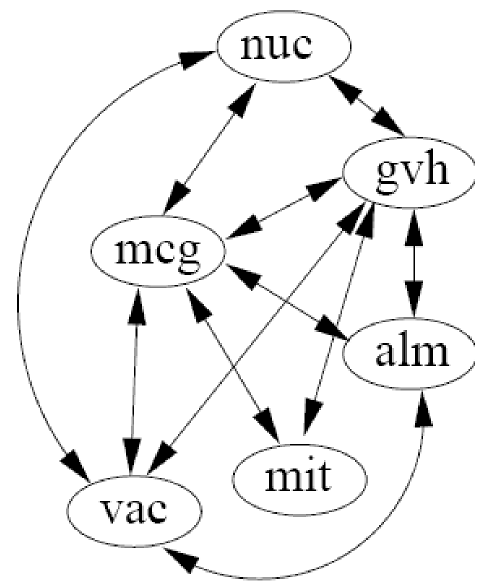
- Maximum likelihood, set k = 2
- Start bi-directed model from latent variable model solution

Fold	Glass	GlassLVM	Fire	FireLVM	Heart	HeartLVM	Wine	WineLVM
1	-8.31	-8.47	-8.62	-8.81	-6.05	-6.11	-8.69	-8.97
2	-8.74	-8.73	-9.29	-9.62	-7.71	-7.73	-8.73	-9.03
3	-5.11	-6.69	-6.89	-6.91	-6.58	-6.76	-8.34	-8.20
4	-7.12	-7.90	-7.94	-7.97	-5.48	-5.52	-9.91	-9.87
5	-3.69	-5.62	-7.49	-7.56	-6.18	-6.63	-8.33	-8.57

Relaxed algorithm: increases target function 50%-70% of the time

Experiment II

- Simple maximum a posteriori
  - standard Gaussian "experts" for the covariances, inverse gamma (2, 2) "experts" for the variances
- Data: YEAST (1484 points, 6 variables)
- Results: between -7.18 to -7.31
- Latent variable model (via maximum likelihood): -9.68 to -10.78



Conclusion

- Approximation methods are needed
- Development of full mixed graph solution
- Applications in sparse multiple regression, sparse heteroscedastic regression, causal inference, etc.

Thoughts on Bayesian methods

- MCMC method: a M-H proposal based on the relaxed fitting algorithm
  - Is that going to work well?
- Other priors?
- Problem is "doubly-intractable"
  - Not because of a partition function, but because of constraints
  - Are there any analogues to methods such as Murray/Ghahramani/McKay's?

Acknowledgements

- Thanks to Yee Whye Teh and Massimiliano Pontil for some useful discussions

Thank You