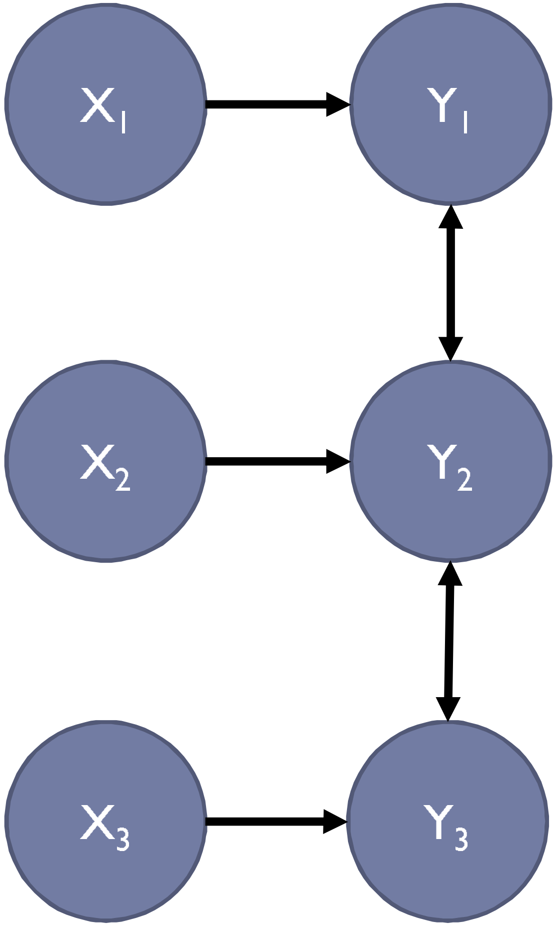
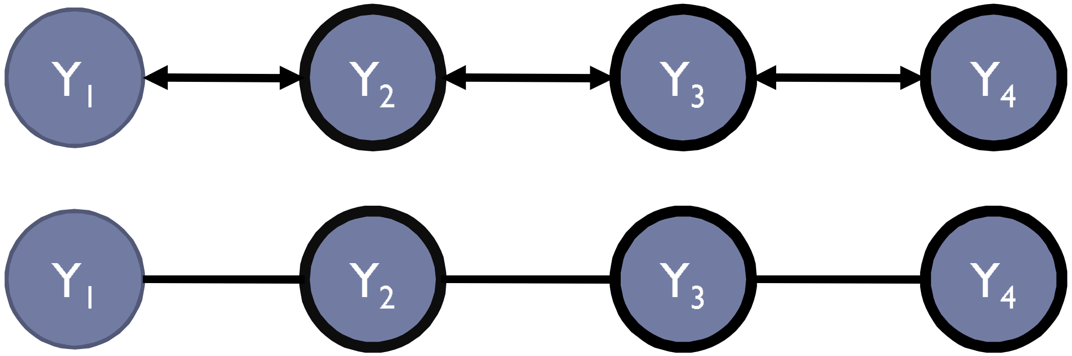
Hidden Common Cause Relations in Relational Learning

Poster ID: T62



- Ricardo Silva (Gatsby Unit/UCL)
- Wei Chu (Columbia)
- Zoubin Ghahramani (Cambridge)

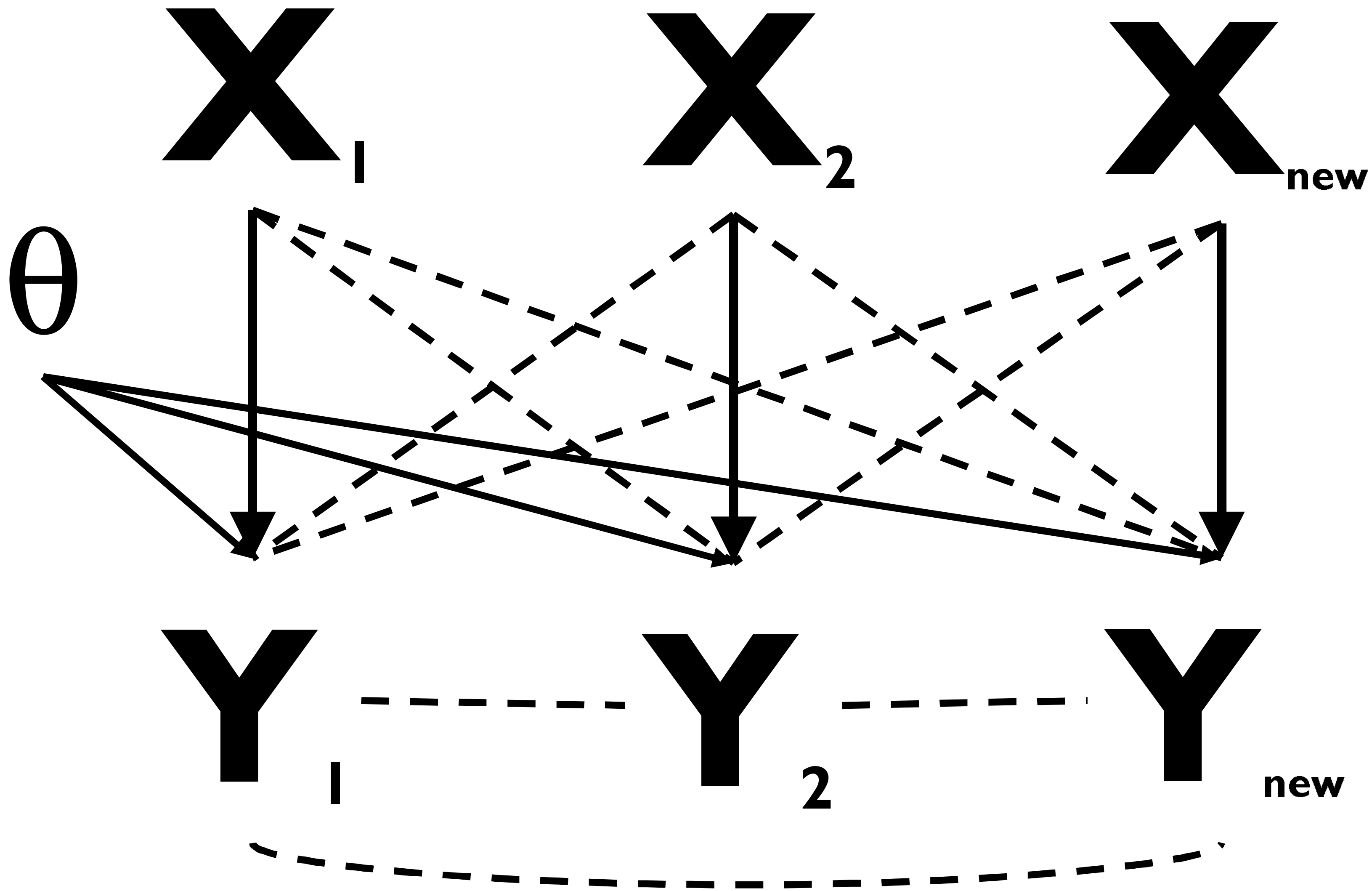
silva@statslab.cam.ac.uk



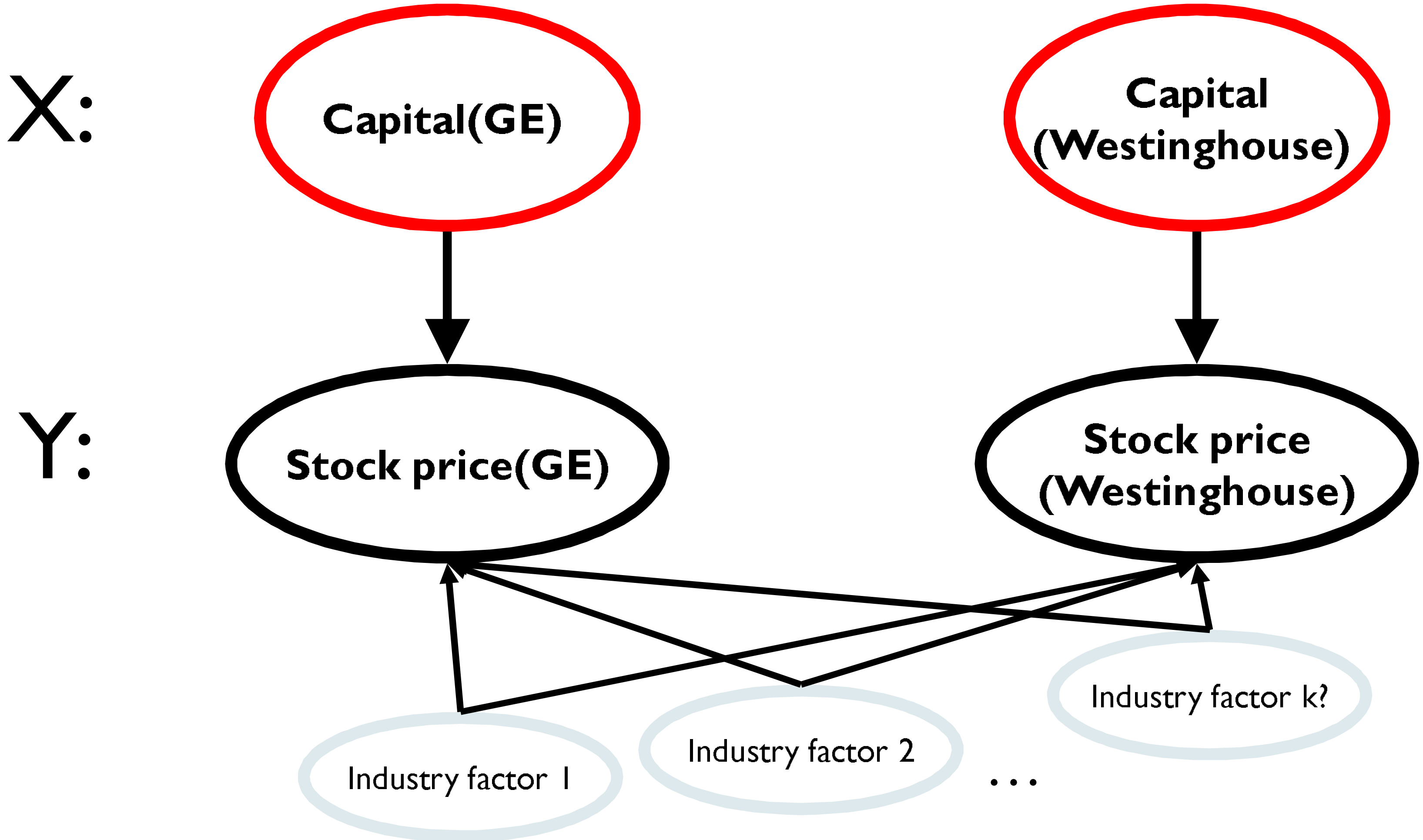
In a Nutshell

- The problem: classification with non-iid data
  - The source of non-iidness: relational information
- A new family of models:
  - Where conditioning creates dependence
  - This means chains of training points generate "long distance" dependencies
  - Distinct from and complements Markov networks
- Experiments with classification of text documents

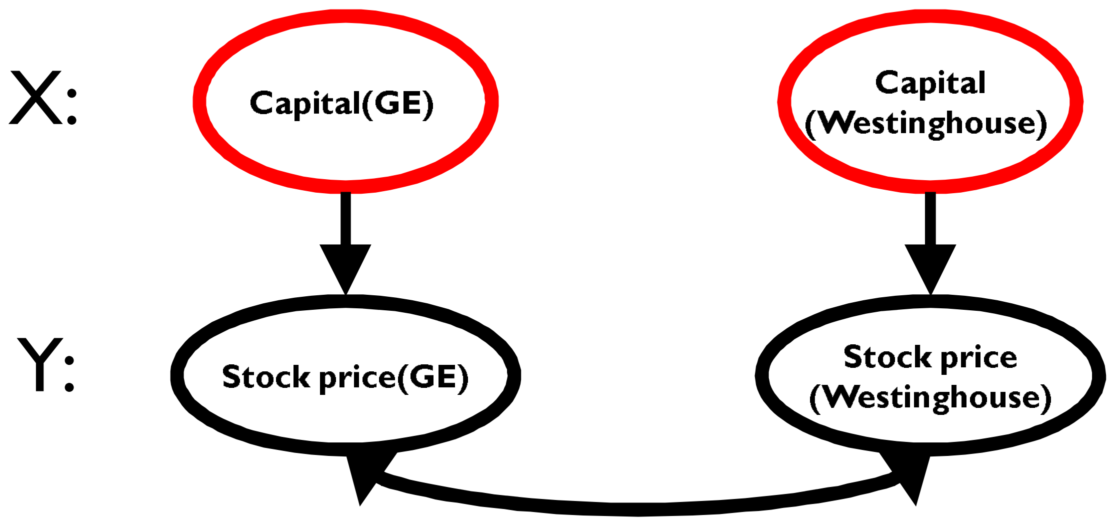
Learning with Non-IID Data

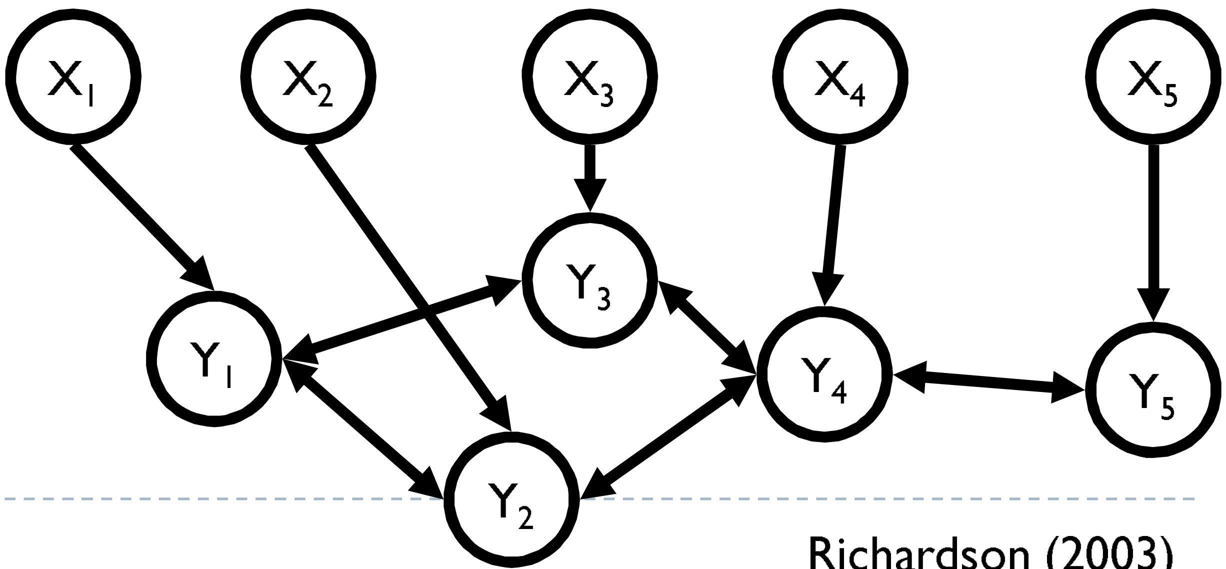


Hidden Common Cause Relations



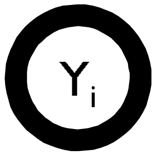
Notation: Directed Mixed Graphs



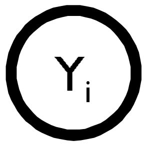


Richardson (2003)

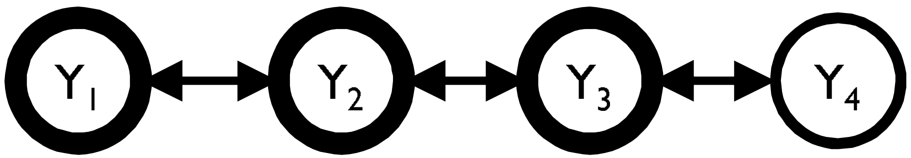
What are the implications? – a comparison with Markov networks/CRFs



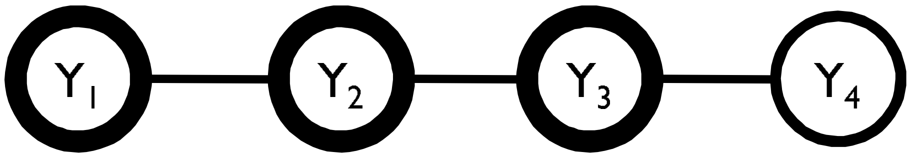
: observed node



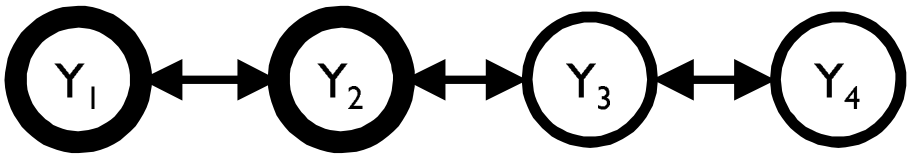
: unobserved node



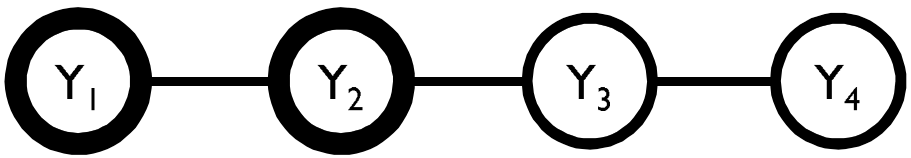
Information from Y_1 passes to Y_4



Information from Y_1 does not pass to Y_4



Information from Y_2 does not pass to Y_4



Information from Y_2 passes to Y_4

Model for Binary Classification

- Non-parametric probit regression

$$P(y_i = 1 | x_i) = P(y^*(x_i) > 0)$$
$$y^*(x_i) = f(x_i) + \varepsilon_i, \quad \varepsilon_i \sim N(0, 1)$$

- Zero-mean Gaussian process prior over $f(\cdot)$
- Relational dependency model:
  - Make $\{\varepsilon\}$ dependent multivariate Gaussian, unit variance
  - For convenience, decouple it into two error terms

$$\varepsilon = \varepsilon^* + \zeta$$

Dependency Model: the Decomposition

Independent from each other

$$\varepsilon = \varepsilon^* + \zeta$$

Marginally independent

Dependent according to relations

$$\Sigma_{\varepsilon} = \Sigma_{\varepsilon^*} + \Sigma_{\zeta}$$

Diagonal

Not diagonal, <u>with 0s only on unrelated pairs</u>

Dependency Model: the Decomposition

$$y^*(x_i) = f(x_i) + \varepsilon = f(x_i) + \zeta + \varepsilon^* = g(x_i) + \varepsilon^*$$

- If K was the original kernel matrix for $f(\cdot)$, the covariance of $g(\cdot)$ is simply

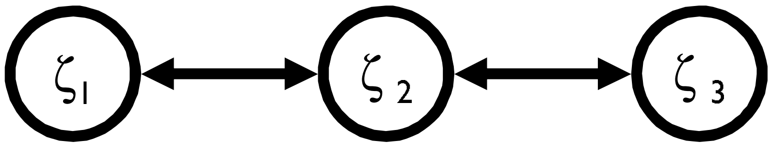
$$\Sigma_{g(.)} = K + \Sigma_{\varepsilon^*}$$

- Plugging-in Expectation-Propagation:
  - Likelihood does not factorize over $f(\cdot)$, but factorizes over $g(\cdot)$!

$$p(g \mid \mathbf{x}, y) \propto p(g \mid \mathbf{x}) \prod_i p(y_i \mid g(x_i))$$

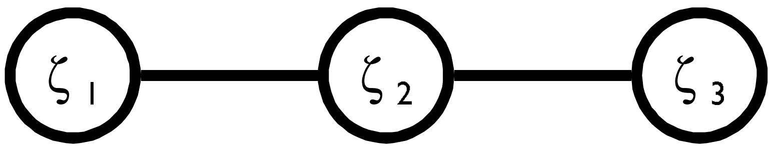
Parameterizing the Relational Covariance Σ_ζ

- "Poking" zeroes in a covariance matrix is tricky:



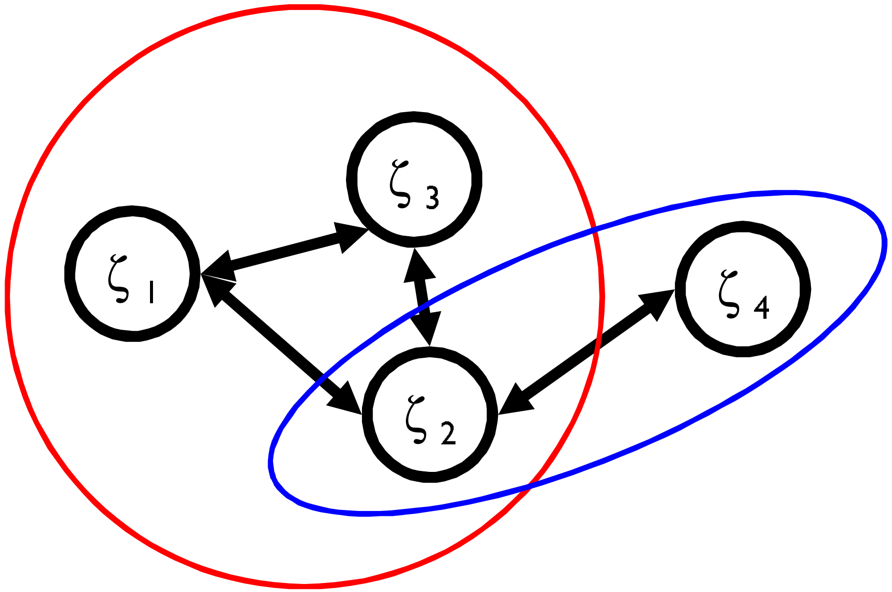
$$\begin{bmatrix} 1 & 0.8 & 0.8 \\ 0.8 & 1 & 0.8 \\ 0.8 & 0.8 & 1 \end{bmatrix} \xrightarrow{?} \begin{bmatrix} 1 & 0.8 & 0 \\ 0.8 & 1 & 0.8 \\ 0 & 0.8 & 1 \end{bmatrix}$$

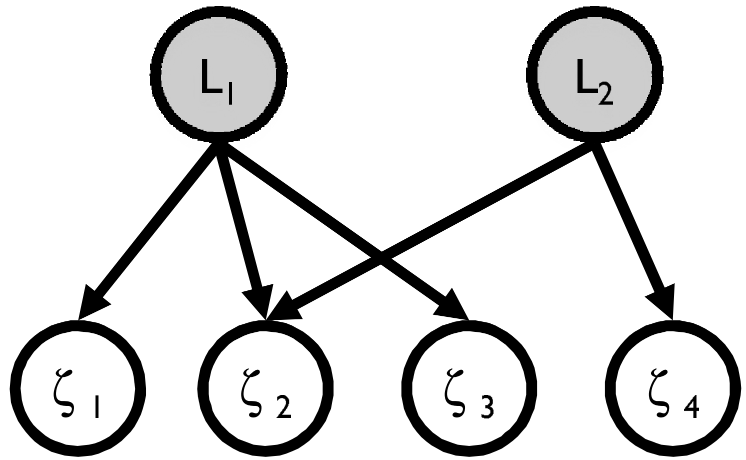
- (Note: Markov network forces zeros on the *inverse*)



$$\Sigma^{-1} = \begin{bmatrix} 1 & \rho_{12} & 0 \\ \rho_{12} & 1 & \rho_{23} \\ 0 & \rho_{23} & 1 \end{bmatrix}$$

Parameterizing the Relational Covariance Σ_ζ





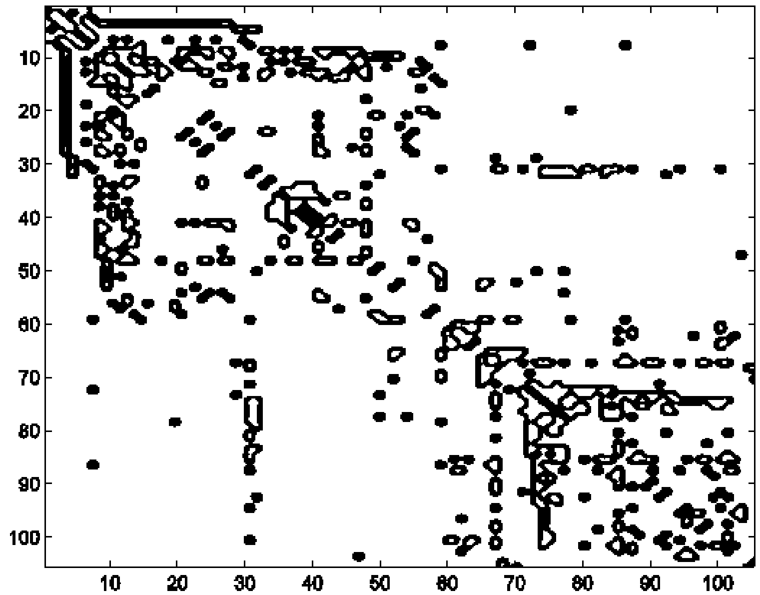
- $\zeta_1 = L_1 + \Delta_1$
- $\zeta_2 = L_1 + L_2 + \Delta_2$
- $\zeta_3 = L_1 + \Delta_3$
- $\zeta_4 = L_2 + \Delta_4$

- Find all cliques and create a latent variable for each.
- Rescale marginal correlation matrix **U** by a factor ρ
  - $\Sigma_\zeta = \rho U$
  - ρ becomes a hyperparameter in [0, 1]
- In practice, cannot extract all cliques
- Suggestion: triangulate and then extract
  - A relaxation of the problem (not always harmless)

Experimental Setup

- Three text classification tasks
- Comparisons:
  - Standard Gaussian Process classifiers
  - Standard GPs with link features
  - The relational GP (RGP) of Chu et al. (2006 – Last NIPS)
  - Our Mixed Graph Gaussian Process: XGP
  - Linear kernels
- Criterion:
  - Area under the curve (AUC)
- Transductive setting:
  - Test points given in advance

Experiment I: Political Books dataset



- 105 books: conservative or liberal?
  - Text extracted from Amazon.com front pages
  - Available at www.statslab.cam.ac.uk/~silva
- 50% training, 50% test
- AUC for standard GP: 0.92
- AUC for RGP and XGP about the same: 0.98

Experiment II: Subset of CORA

- Database of publications in Computer Science
- 1% for training, 99% for test (too easy)
  - Very "uniform" links – mostly between same class papers
- XGP cannot do better than RGP when there is so little training data to propagate information

Table 1: The averaged AUC scores of citation prediction on test cases of the Cora database are recorded along with standard deviation over 100 trials. "n" denotes the number of papers in one class. "Citations" denotes the citation count within the two paper classes.

Group	n	Citations	GPC	GPC with Citations	XGP
5vs1	346/488	2466	0.905 ± 0.031	0.891 ± 0.022	0.945 ± 0.053
5vs2	346/619	3417	0.900 ± 0.032	0.905 ± 0.044	0.933 ± 0.059
5vs3	346/1376	3905	0.863 ± 0.040	0.893 ± 0.017	0.883 ± 0.013
5vs4	346/646	2858	0.916 ± 0.030	0.887 ± 0.018	0.951 ± 0.042
5vs6	346/281	1968	0.887 ± 0.054	0.843 ± 0.076	0.955 ± 0.041
5vs7	346/529	2948	0.869 ± 0.045	0.867 ± 0.041	0.926 ± 0.076

Experiment III: WebKB

- Hardest task: "outlier" detection
  - Identify pages that are not student/faculty/department/project
- Notice that links between pages are of all sorts
  - Makes sense to propagate information only if class label is given
- 10% for training, 90% for test

Table 2: Comparison of the three algorithms on the task "other" vs. "not-other" in the WebKB domain. Results for GPC and RGP taken from [2]. The same partitions for training and test are used to generate the results for XGP. Mean and standard deviation of AUC results are reported.

University	Numbers			Other or Not		
	Other	All	Link	GPC	RGP	XGP
Cornell	617	865	13177	0.708 ± 0.021	0.884 ± 0.025	0.917 ± 0.022
Texas	571	827	16090	0.799 ± 0.021	0.906 ± 0.026	0.949 ± 0.015
Washington	939	1205	15388	0.782 ± 0.023	0.877 ± 0.024	0.923 ± 0.016
Wisconsin	942	1263	21594	0.839 ± 0.014	0.899 ± 0.015	0.941 ± 0.018

Conclusions

- Truly new relational model
  - Remember to think: graphical models are more than drawings
- Trivial to implement
  - One can reuse GP classifier code easily
- Requires one more hyperparameter only
- Many directions to explore:
  - So far, extremely simple covariance parameterizations
    - Several alternatives of parameterization as open directions
  - Combination of different relationships
    - Multiple kernel learning
  - Different models, heteroskedastic noise, full Bayesian learning, etc.
- Code available at http://www.statslab.cam.ac.uk/~silva