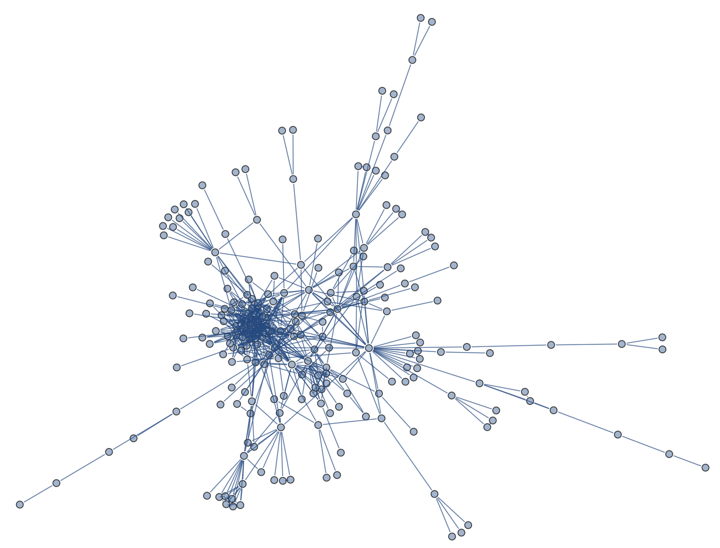
Random function priors for exchangeable arrays with applications to graphs and relational data

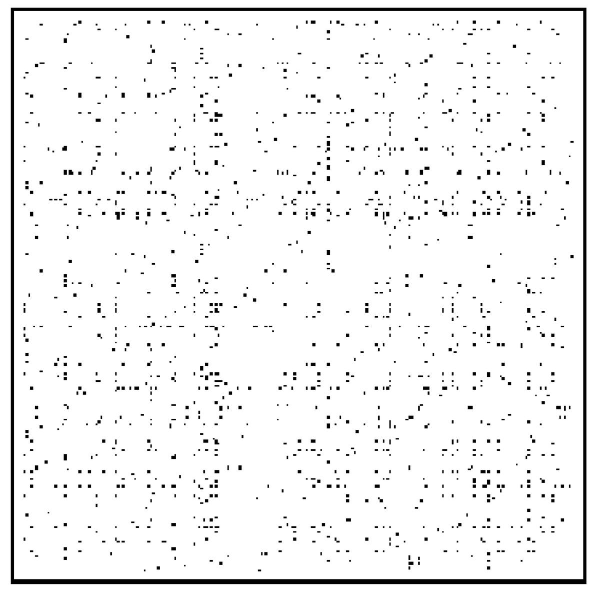
James Robert Lloyd[1], Peter Orbanz[2], Zoubin Ghahramani[1], Daniel M. Roy[1]

1: Department of Engineering, University of Cambridge, UK 2: Department of Statistics, Columbia University, USA

Structured relational data typically encoded in the form of arrays. . .

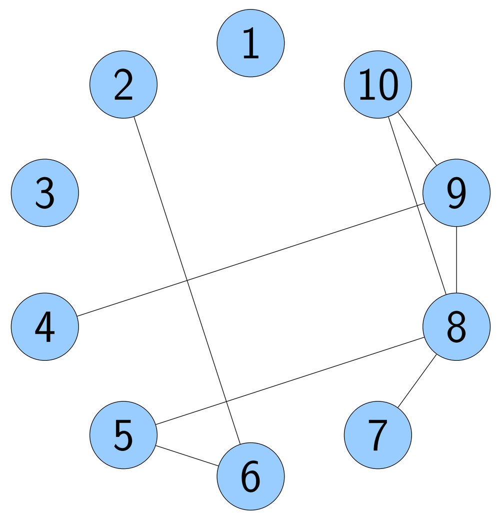


A protein interactome. . .

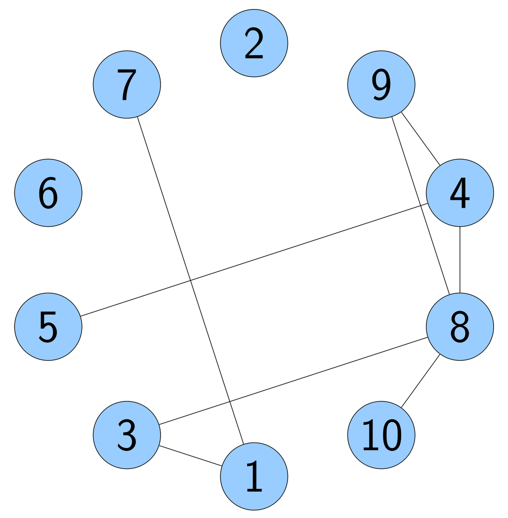


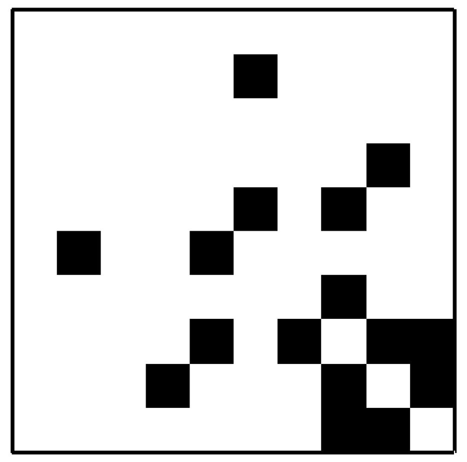
. . . encoded as an array

. . . which are often invariant to permutations of rows and columns i.e. they are exchangeable

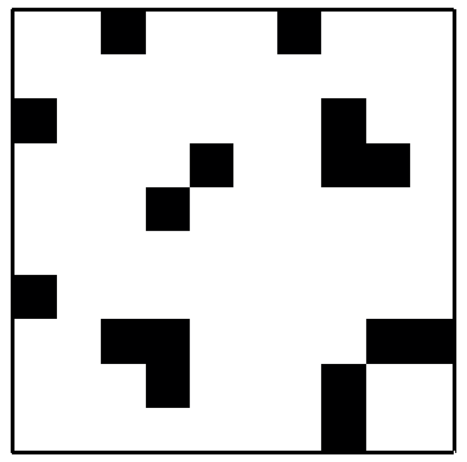


$\equiv$





$\equiv$



When the labelling of nodes is arbitrary, the two adjacency matrices should be treated equivalently.

Probability distributions for exchangeable arrays can be characterised. . .

Definition. An array $X = (X_{ij})_{i,j\in\mathbb{N}}$ is called an *exchangeable array* if

$$(X_{ij}) \overset{d}{=} (X_{\pi(i)\pi(j)}) \qquad \text{for every } \pi \in \mathbb{S}_\infty \,.$$

Theorem (Aldous, Hoover). *A random 2-array (X_{ij}) is exchangeable if and only if there is a random (measurable) function $F : [0,1]^3 \to \mathcal{X}$ such that*

$$(X_{ij}) \overset{d}{=} (F(U_i, U_j, U_{ij})).$$

for every collection $(U_i)_{i\in\mathbb{N}}$ and $(U_{ij})_{i\leq j\in\mathbb{N}}$ of i.i.d. $\mathrm{Uniform}[0,1]$ random variables, where $U_{ji} = U_{ij}$ for $j < i \in \mathbb{N}$.

. . . inspiring a simple Bayesian nonparametric model

We decompose the function F into two functions $\Theta : [0,1]^2 \to \mathcal{W}$ and $H : [0,1] \times \mathcal{W} \to \mathcal{X}$ for a suitable space $\mathcal{W}$, such that

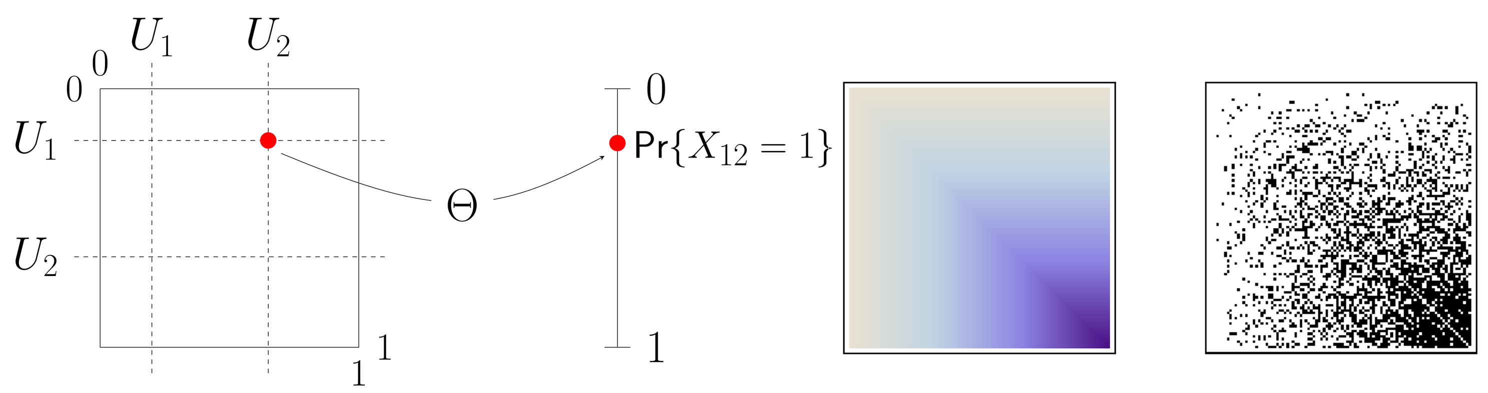
$$(X_{ij}) \overset{d}{=} (F(U_i, U_j, U_{ij})) = (H(U_{ij}, \Theta(U_i, U_j))) \,.$$

Inspiring the following generative model

$$\begin{aligned} \Theta &\sim \mathcal{GP}(0, \kappa) \\ U_1, U_2, \dots &\sim_{\text{iid}} \mathrm{Uniform}[0,1] \\ X_{ij} \,|\, W_{ij} &\sim P[\,.\,|W_{ij}] \end{aligned}$$

$$\text{where } W_{ij} = \Theta(U_i, U_j).$$

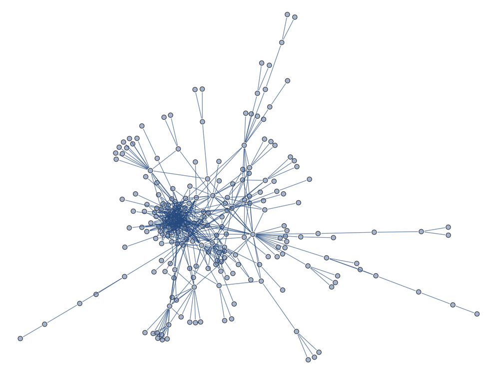
Example: Sampling a symmetric binary 2-array i.e. an undirected graph:



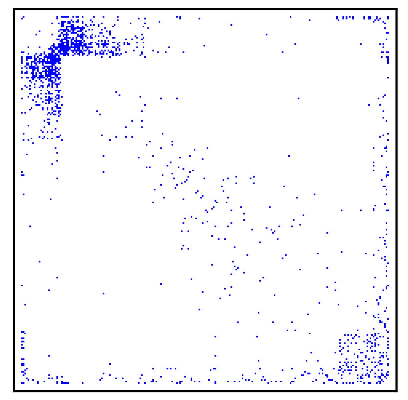
Inference: MCMC version of Gaussian process approximation

- Inducing points in subset of regressors appoximation treated as random variables (locations and values)
- Gaussian process sampled using elliptical slice sampling; all other parameters sampled using slice sampling

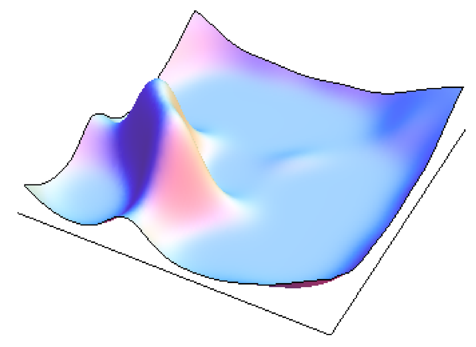
Latent variables are interpretable



A protein interactome



Adjacency matrix sorted by MAP embedding



MAP Θ

Sorting the adjacency matrix of the protein interactome using the (approximate) MAP values of the U_i reveals interpretable structure in the data. The higher density of edges along the diagonal reveals homophily. The block structure in the top left reveals stochastic equivalence. These features can also be seen in the MAP Θ.

Representation allows for common perspective on a variety of models

Graph data

Random function model	Θ	$\sim \mathcal{GP}(0, \kappa)$
Latent class	W_{ij}	$= \Lambda_{U_i U_j}$ where $U_i \in \{1, \dots, K\}$
IRM	W_{ij}	$= \Lambda_{U_i U_j}$ where $U_i \in \{1, \dots, \infty\}$
Latent distance	W_{ij}	$= -\lvert U_i - U_j \rvert$
Eigenmodel	W_{ij}	$= U_i' \Lambda U_j$
LFRM	W_{ij}	$= U_i' \Lambda U_j$ where $U_i \in \{0,1\}^\infty$
ILA	W_{ij}	$= \sum_d \mathbb{I}_{U_{id}} \mathbb{I}_{U_{jd}} \Lambda^{(d)}_{U_{id} U_{jd}}$ where $U_i \in \{0, \dots, \infty\}^\infty$
SMGB	Θ	$\sim \mathcal{GP}(0, \kappa_1 \otimes \kappa_2)$

Real-valued array data

Random function model	Θ	$\sim \mathcal{GP}(0, \kappa)$
Mondrian process based	Θ	$=$ piece-wise constant random function
PMF	W_{ij}	$= U_i' V_j$
GPLVM	Θ	$\sim \mathcal{GP}(0, \kappa \otimes \delta)$

Flexibility of nonparametric model warranted by empirical prediction study

AUC results

Data set	High school			NIPS			Protein		
Latent dim.	1	2	3	1	2	3	1	2	3
PMF	0.747	0.792	0.792	0.729	0.789	0.820	0.787	0.810	0.841
Eigenmodel	0.742	0.806	0.806	0.789	0.818	0.845	0.805	0.866	0.882
GPLVM	0.744	0.775	0.782	0.888	0.876	0.883	0.877	0.883	0.873
RFM	**0.815**	**0.827**	**0.820**	**0.907**	**0.914**	**0.919**	**0.903**	**0.910**	**0.912**