Learning the structure of probabilistic graphical model with infinite directed acyclic graphs

Patrick Dallaire

Université Laval
Département d'informatique et de génie logiciel

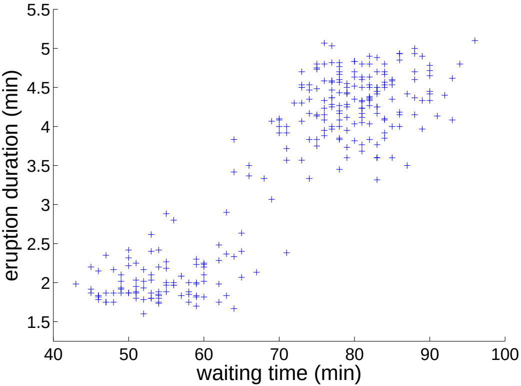
April 26, 2013

Introduction

Task: Discovering structure in data

Motivation: Predictions, reproduce data, knowledge discovery, etc.

Approach: Bayesian learning considering all possible structures

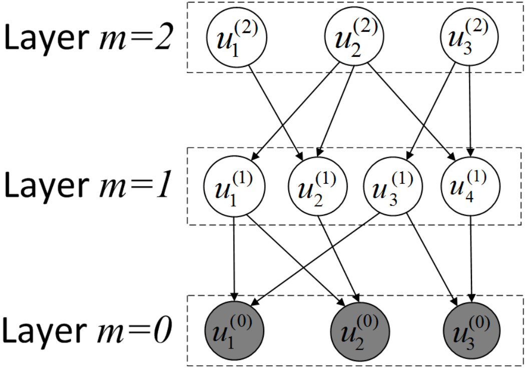


Outline of the talk

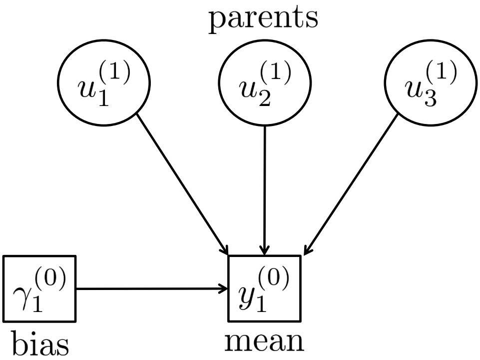
- The belief network
- Learning the structure
- Infinite dimensional layers (IBP)
- Infinitely deep networks (CIBP)
- Jumping-connections (eCIBP)
- Results on learning belief networks

Belief networks

Structure of the belief network

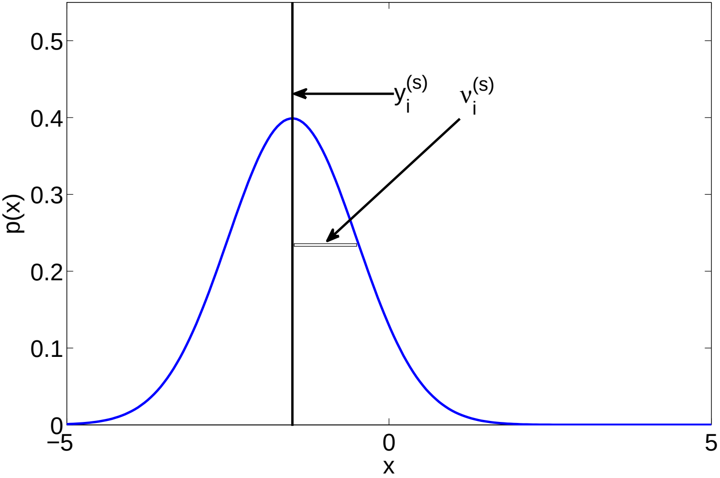


Unit activations - weighted sum

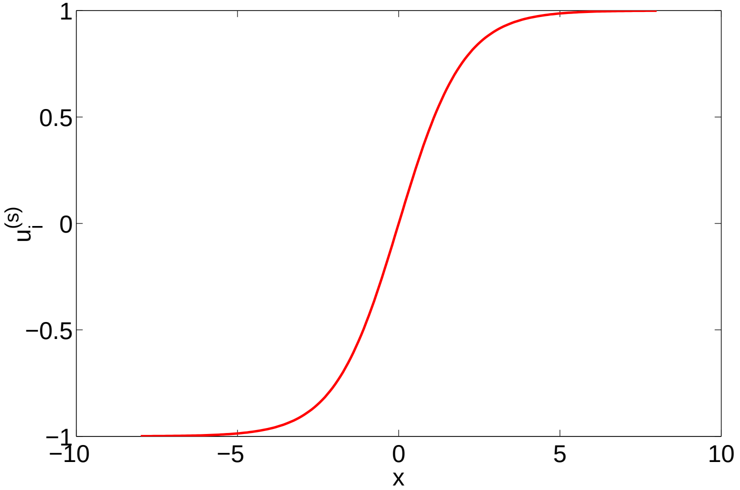


$$y_i^{(s)} = \gamma_i^{(s)} + \sum_k Z_{i,k}^{(s,m)} W_{i,k}^{(s,m)} u_k^{(m)}$$

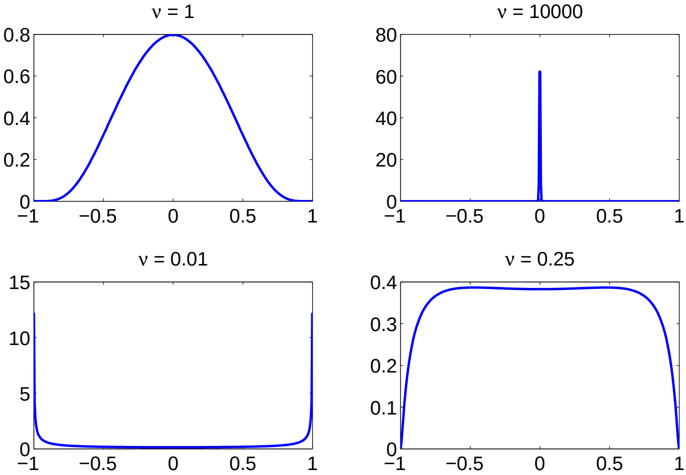
Unit activations - Gaussian random variable



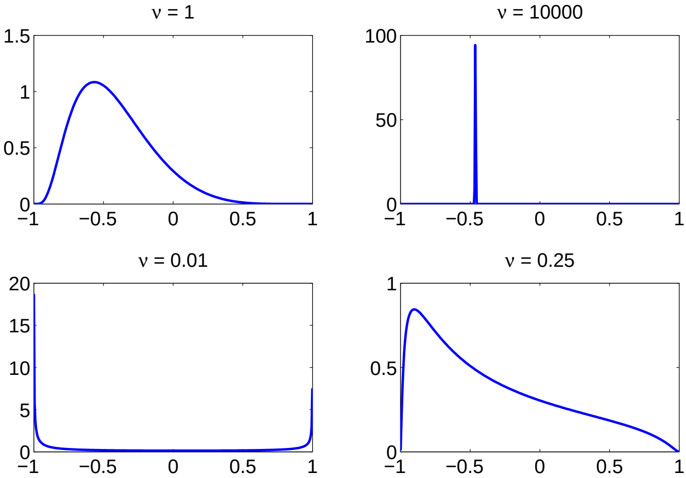
Unit activations - sigmoid transformation



Conditional activation probabilities

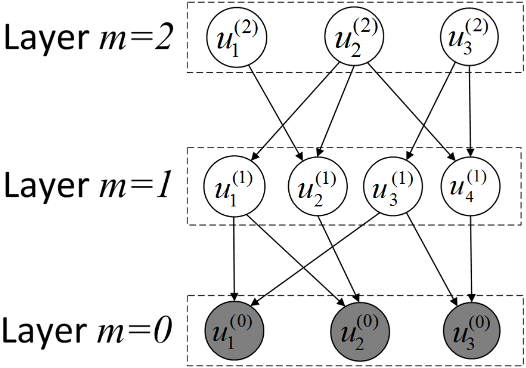


Conditional activation probabilities

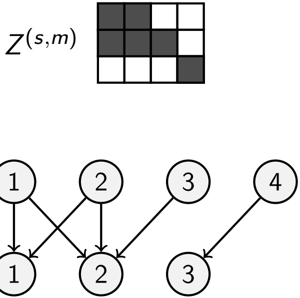


Learning the structure

Structure of the belief network

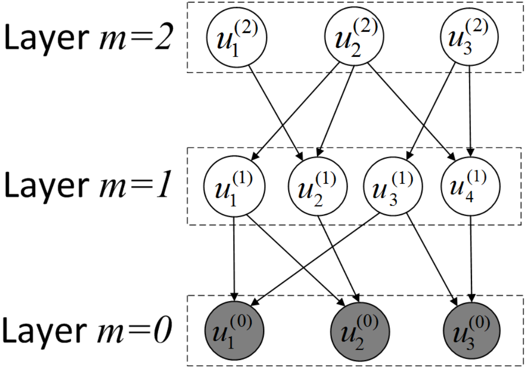


Structure as binary matrix



Infinite dimensional layer

Structure of the belief network

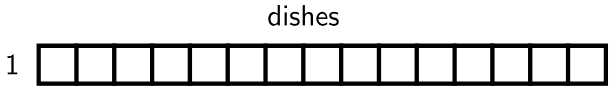


From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

From Indian buffet processes

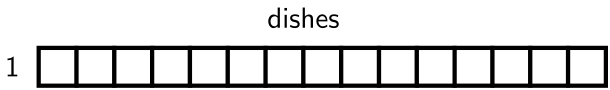
The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.



From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

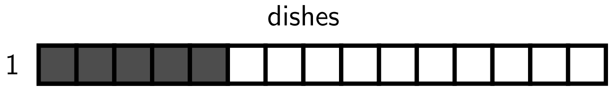
- first customer samples Poisson(α) dishes



From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

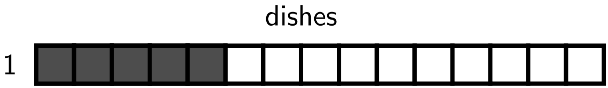
- first customer samples Poisson(α) dishes



From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

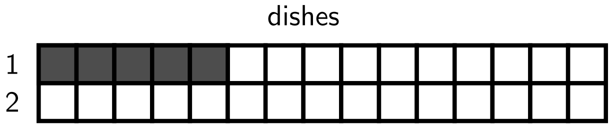
- first customer samples Poisson(α) dishes
- i^{th} customer tries :



From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

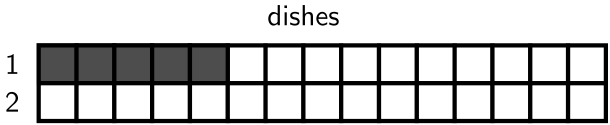
- first customer samples Poisson(α) dishes
- i^{th} customer tries :



From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

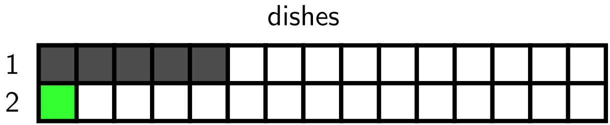
- first customer samples Poisson(α) dishes
- i^{th} customer tries :
  - previously sampled dish k with probability $n_k/(\beta + i - 1)$



From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

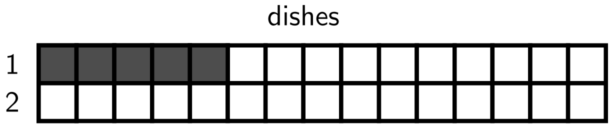
- first customer samples Poisson(α) dishes
- i^{th} customer tries :
  - previously sampled dish k with probability $n_k/(\beta + i - 1)$



From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

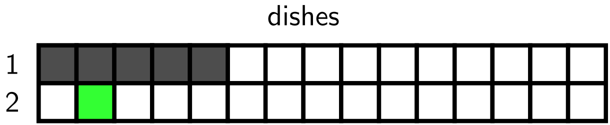
- first customer samples Poisson(α) dishes
- i^{th} customer tries :
  - previously sampled dish k with probability $n_k/(\beta + i - 1)$



From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

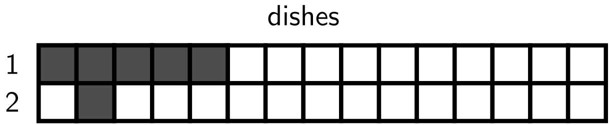
- first customer samples Poisson(α) dishes
- i^{th} customer tries :
  - previously sampled dish k with probability $n_k/(\beta + i - 1)$



From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

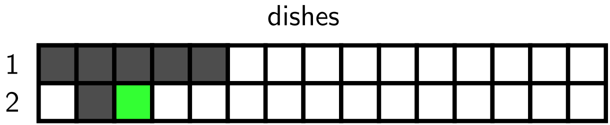
- first customer samples Poisson(α) dishes
- i^{th} customer tries :
  - previously sampled dish k with probability $n_k/(\beta + i - 1)$



From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

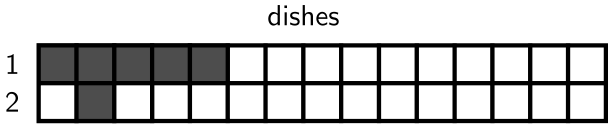
- first customer samples Poisson(α) dishes
- i^{th} customer tries :
  - previously sampled dish k with probability $n_k/(\beta + i - 1)$



From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

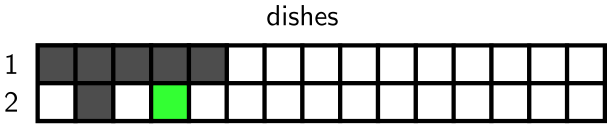
- first customer samples Poisson(α) dishes
- i^{th} customer tries :
  - previously sampled dish k with probability $n_k/(\beta + i - 1)$



From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

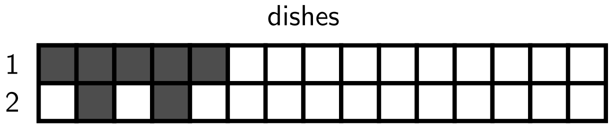
- first customer samples Poisson(α) dishes
- i^{th} customer tries :
  - previously sampled dish k with probability $n_k/(\beta + i - 1)$



From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

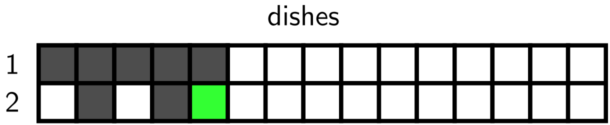
- first customer samples Poisson(α) dishes
- i^{th} customer tries :
    - previously sampled dish k with probability $n_k/(\beta + i - 1)$



From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

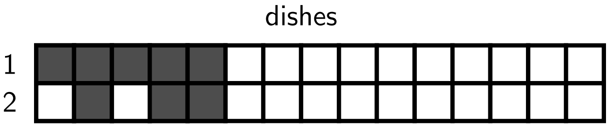
- first customer samples Poisson(α) dishes
- i^{th} customer tries :
  - previously sampled dish k with probability $n_k/(\beta + i - 1)$



From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

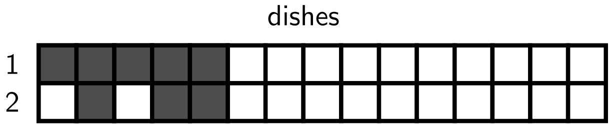
- first customer samples Poisson(α) dishes
- i^{th} customer tries :
    - previously sampled dish k with probability $n_k/(\beta + i - 1)$



From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

- first customer samples Poisson(α) dishes
- i^{th} customer tries :
  - previously sampled dish k with probability $n_k/(\beta + i - 1)$
  - adds Poisson($\alpha\beta/(\beta + i - 1)$) new dishes

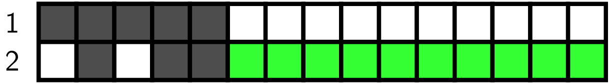


From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

- first customer samples Poisson(α) dishes
- i^{th} customer tries :
  - previously sampled dish k with probability $n_k/(\beta + i - 1)$
  - adds Poisson$(\alpha\beta/(\beta + i - 1))$ new dishes

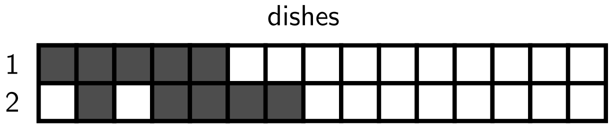
dishes



From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

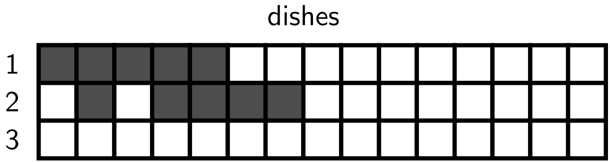
- first customer samples Poisson(α) dishes
- i^{th} customer tries :
  - previously sampled dish k with probability $n_k/(\beta + i - 1)$
  - adds Poisson($\alpha\beta/(\beta + i - 1)$) new dishes



From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

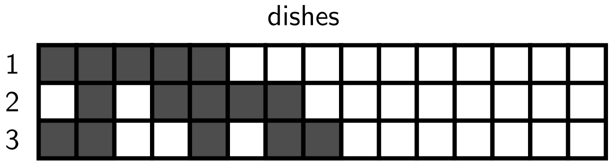
- first customer samples Poisson(α) dishes
- i^{th} customer tries :
  - previously sampled dish k with probability $n_k/(\beta + i - 1)$
  - adds Poisson($\alpha\beta/(\beta + i - 1)$) new dishes



From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

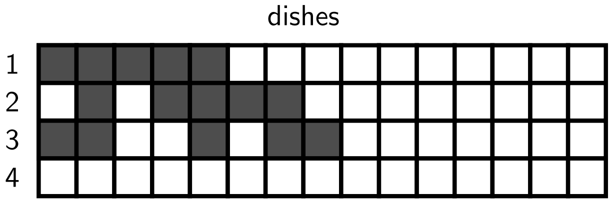
- first customer samples Poisson(α) dishes
- i^{th} customer tries :
  - previously sampled dish k with probability $n_k/(\beta + i - 1)$
  - adds Poisson($\alpha\beta/(\beta + i - 1)$) new dishes



From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

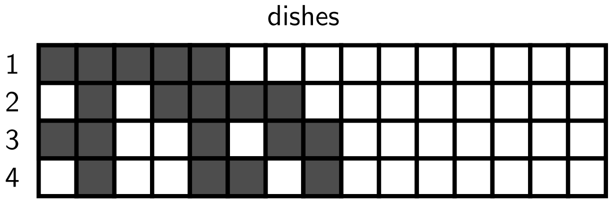
- first customer samples Poisson(α) dishes
- i^{th} customer tries :
  - previously sampled dish k with probability $n_k/(\beta + i - 1)$
  - adds Poisson($\alpha\beta/(\beta + i - 1)$) new dishes



From Indian buffet processes

The IBP(α, β) metaphor consists in customers choosing various dishes from an infinitely long buffet.

- first customer samples Poisson(α) dishes
- i^{th} customer tries :
  - previously sampled dish k with probability $n_k/(\beta + i - 1)$
  - adds Poisson($\alpha\beta/(\beta + i - 1)$) new dishes



From Beta processes

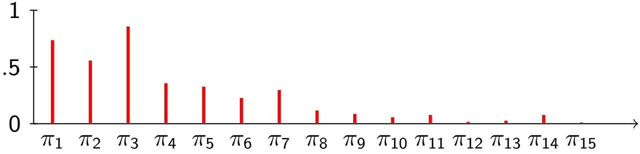
We can generate a $K^{(m-1)} \times \infty$ binary matrix $Z^{(m-1,m)}$ by

- sampling an infinite discrete measure $B^{(m)} \sim \mathrm{BP}(\beta, B_0^{(m)})$
- draw samples from a Bernoulli process $Z_{i,\cdot}^{(m-1,m)} \sim \mathrm{BeP}(B^{(m)})$

From Beta processes

We can generate a $K^{(m-1)} \times \infty$ binary matrix $Z^{(m-1,m)}$ by

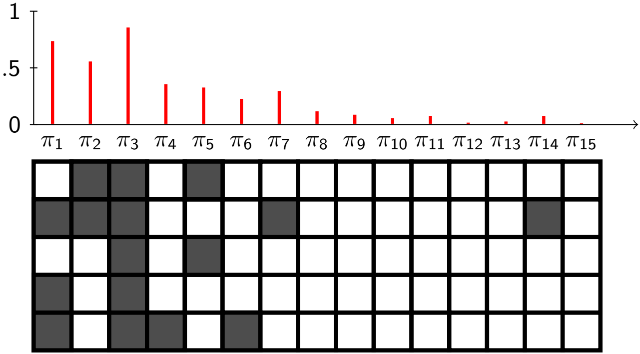
- sampling an infinite discrete measure $B^{(m)} \sim \mathrm{BP}(\beta, B_0^{(m)})$
- draw samples from a Bernoulli process $Z_{i,\cdot}^{(m-1,m)} \sim \mathrm{BeP}(B^{(m)})$



From Beta processes

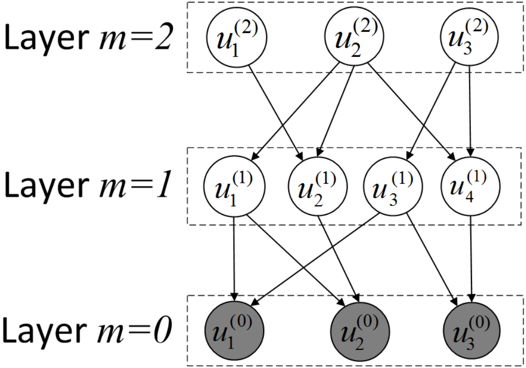
We can generate a $K^{(m-1)} \times \infty$ binary matrix $Z^{(m-1,m)}$ by

- sampling an infinite discrete measure $B^{(m)} \sim \mathrm{BP}(\beta, B_0^{(m)})$
- draw samples from a Bernoulli process $Z_{i,\cdot}^{(m-1,m)} \sim \mathrm{BeP}(B^{(m)})$



Infinitely deep network

Structure of the belief network



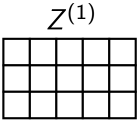
The cascading indian buffet process

Assume infinitely many layer $M \to \infty$ and apply IBP recursively

The cascading indian buffet process

Assume infinitely many layer $M \to \infty$ and apply IBP recursively

- N customers enter the first restaurant and apply the IBP

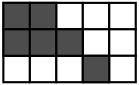


The cascading indian buffet process

Assume infinitely many layer $M \to \infty$ and apply IBP recursively

- N customers enter the first restaurant and apply the IBP
- The $K^{(m)}$ unique dishes sampled in a restaurant corresponds to the number of customers entering the next

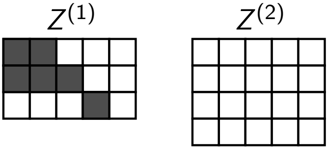
$Z^{(1)}$



The cascading indian buffet process

Assume infinitely many layer $M \to \infty$ and apply IBP recursively

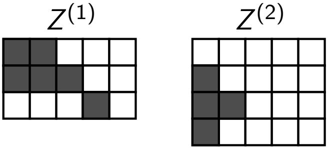
- N customers enter the first restaurant and apply the IBP
- The $K^{(m)}$ unique dishes sampled in a restaurant corresponds to the number of customers entering the next



The cascading indian buffet process

Assume infinitely many layer $M \to \infty$ and apply IBP recursively

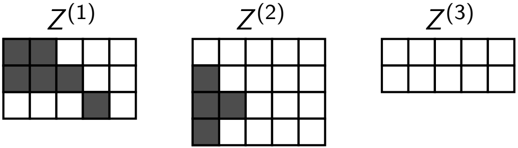
- N customers enter the first restaurant and apply the IBP
- The $K^{(m)}$ unique dishes sampled in a restaurant corresponds to the number of customers entering the next



The cascading indian buffet process

Assume infinitely many layer $M \to \infty$ and apply IBP recursively

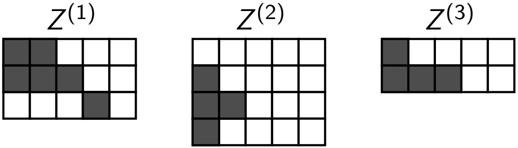
- N customers enter the first restaurant and apply the IBP
- The $K^{(m)}$ unique dishes sampled in a restaurant corresponds to the number of customers entering the next



The cascading indian buffet process

Assume infinitely many layer $M \rightarrow \infty$ and apply IBP recursively

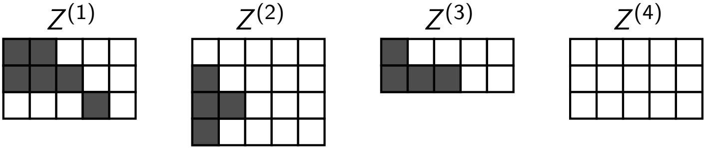
- N customers enter the first restaurant and apply the IBP
- The $K^{(m)}$ unique dishes sampled in a restaurant corresponds to the number of customers entering the next



The cascading indian buffet process

Assume infinitely many layer $M \to \infty$ and apply IBP recursively

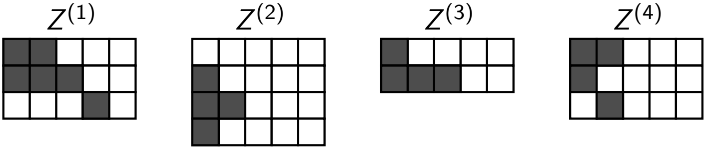
- N customers enter the first restaurant and apply the IBP
- The $K^{(m)}$ unique dishes sampled in a restaurant corresponds to the number of customers entering the next



The cascading indian buffet process

Assume infinitely many layer $M \to \infty$ and apply IBP recursively

- N customers enter the first restaurant and apply the IBP
- The $K^{(m)}$ unique dishes sampled in a restaurant corresponds to the number of customers entering the next



Directed acyclic graph with CIBP

The binary value $Z_{i,k}^{(s,m)}$ indicates whether an edge is leaving the k^{th} nodes in layer m to connect to the i^{th} nodes in layer s.

Directed acyclic graph with CIBP

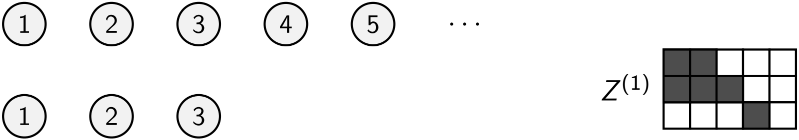
The binary value $Z_{i,k}^{(s,m)}$ indicates whether an edge is leaving the k^{th} nodes in layer m to connect to the i^{th} nodes in layer s.

Directed acyclic graph with CIBP

The binary value $Z_{i,k}^{(s,m)}$ indicates whether an edge is leaving the k^{th} nodes in layer m to connect to the i^{th} nodes in layer s.

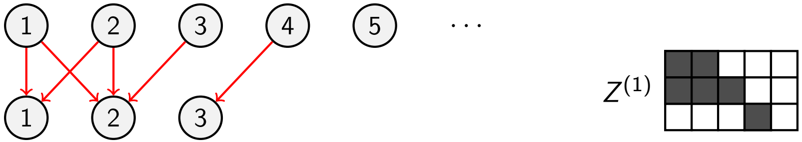
Directed acyclic graph with CIBP

The binary value $Z_{i,k}^{(s,m)}$ indicates whether an edge is leaving the k^{th} nodes in layer m to connect to the i^{th} nodes in layer s.



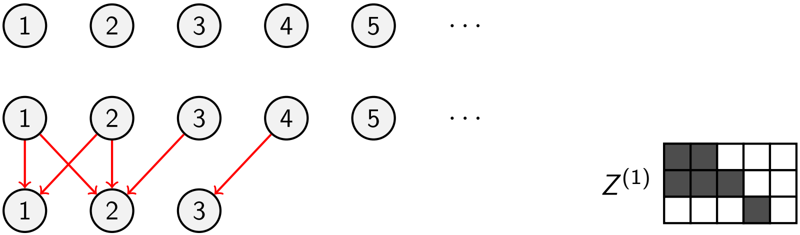
Directed acyclic graph with CIBP

The binary value $Z_{i,k}^{(s,m)}$ indicates whether an edge is leaving the k^{th} nodes in layer m to connect to the i^{th} nodes in layer s.



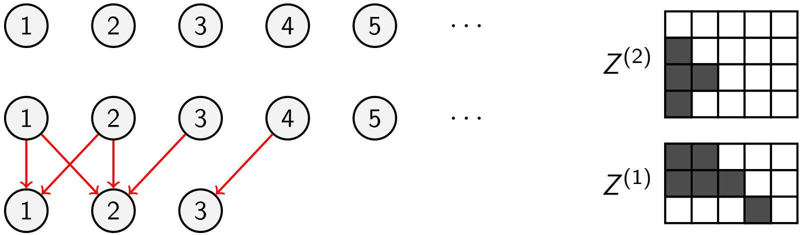
Directed acyclic graph with CIBP

The binary value $Z_{i,k}^{(s,m)}$ indicates whether an edge is leaving the k^{th} nodes in layer m to connect to the i^{th} nodes in layer s.



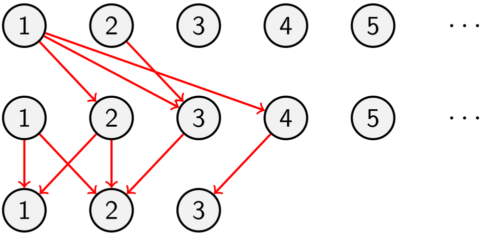
Directed acyclic graph with CIBP

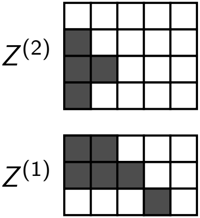
The binary value $Z_{i,k}^{(s,m)}$ indicates whether an edge is leaving the k^{th} nodes in layer m to connect to the i^{th} nodes in layer s.



Directed acyclic graph with CIBP

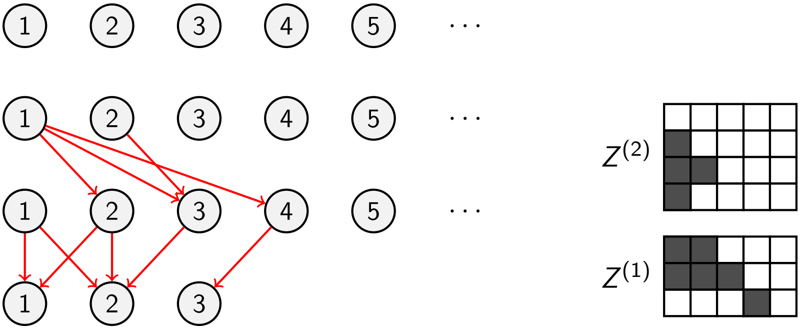
The binary value $Z_{i,k}^{(s,m)}$ indicates whether an edge is leaving the k^{th} nodes in layer m to connect to the i^{th} nodes in layer s.





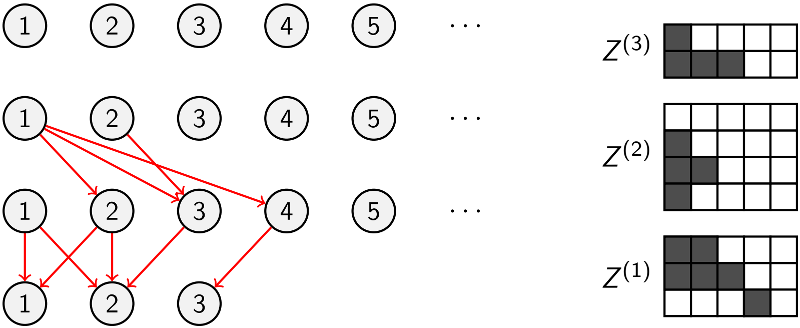
Directed acyclic graph with CIBP

The binary value $Z_{i,k}^{(s,m)}$ indicates whether an edge is leaving the k^{th} nodes in layer m to connect to the i^{th} nodes in layer s.



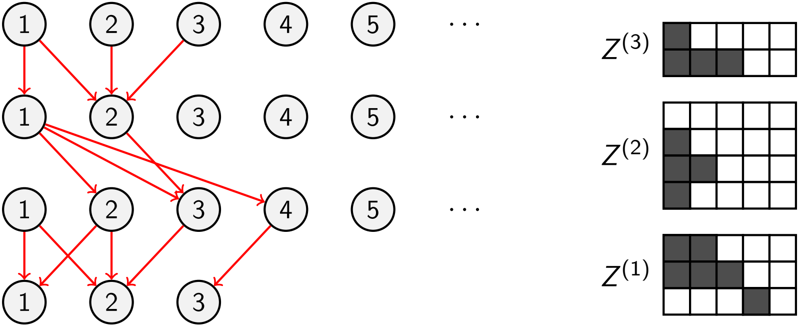
Directed acyclic graph with CIBP

The binary value $Z_{i,k}^{(s,m)}$ indicates whether an edge is leaving the k^{th} nodes in layer m to connect to the i^{th} nodes in layer s.



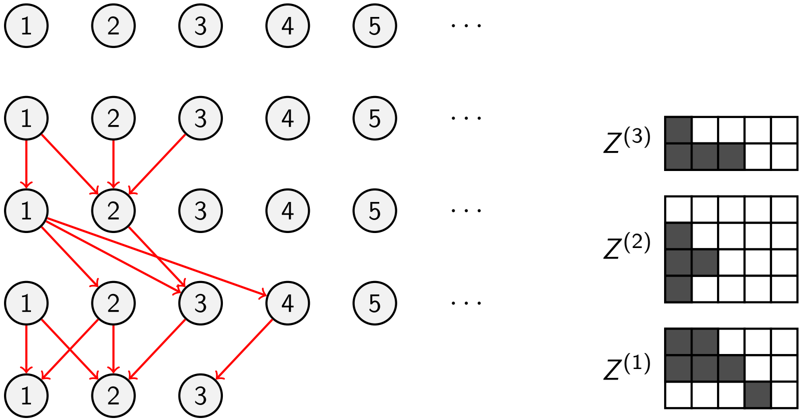
Directed acyclic graph with CIBP

The binary value $Z_{i,k}^{(s,m)}$ indicates whether an edge is leaving the k^{th} nodes in layer m to connect to the i^{th} nodes in layer s.



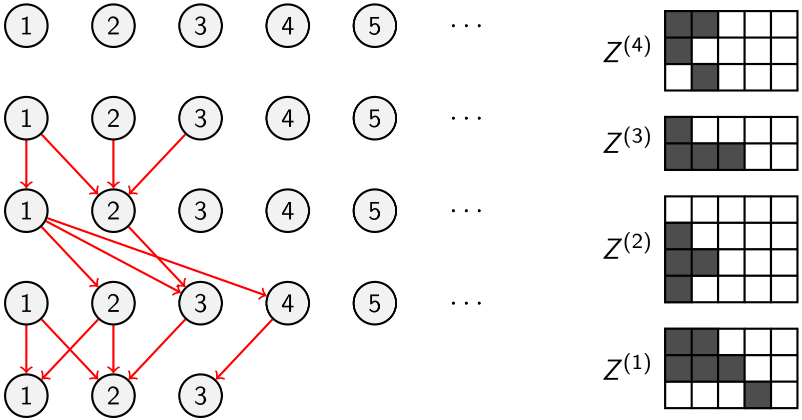
Directed acyclic graph with CIBP

The binary value $Z_{i,k}^{(s,m)}$ indicates whether an edge is leaving the k^{th} nodes in layer m to connect to the i^{th} nodes in layer s.



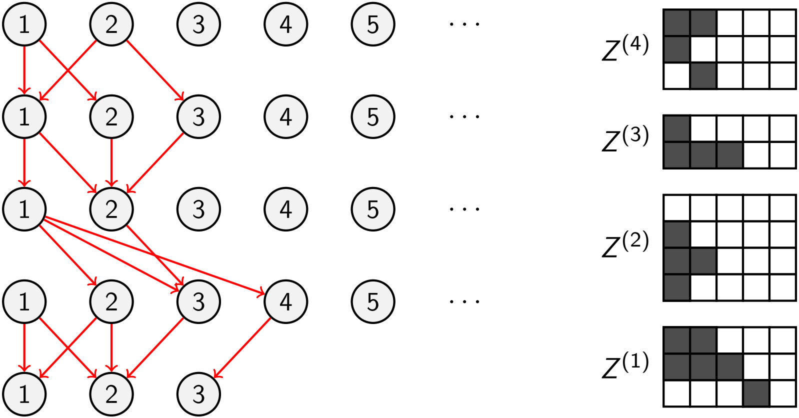
Directed acyclic graph with CIBP

The binary value $Z_{i,k}^{(s,m)}$ indicates whether an edge is leaving the k^{th} nodes in layer m to connect to the i^{th} nodes in layer s.



Directed acyclic graph with CIBP

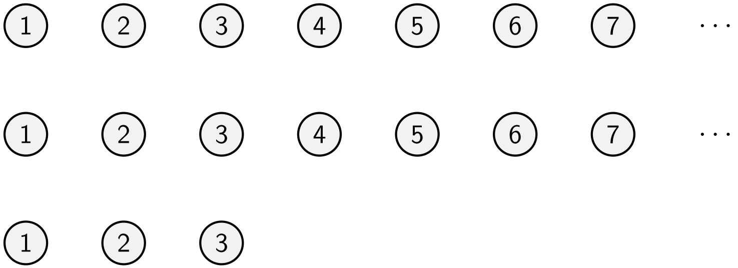
The binary value $Z_{i,k}^{(s,m)}$ indicates whether an edge is leaving the k^{th} nodes in layer m to connect to the i^{th} nodes in layer s.



Beta process representation for CIBP

To generate sequence of binary matrices, we can recursively:

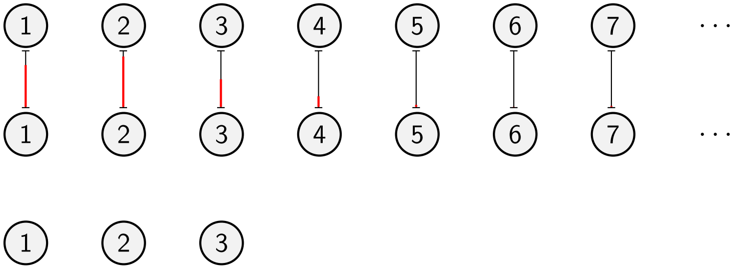
- draw a measure $B^{(m)}$ where $\pi_k^{(m)}$ is the popularity of dish $\theta_k^{(m)}$.
- construct $Z^{(m-1,m)}$ by sampling $Z_{i,\cdot}^{(m-1,m)} \sim \text{BeP}(B^{(m)})$ for the $K^{(m-1)}$ customers entering this restaurant.



Beta process representation for CIBP

To generate sequence of binary matrices, we can recursively:

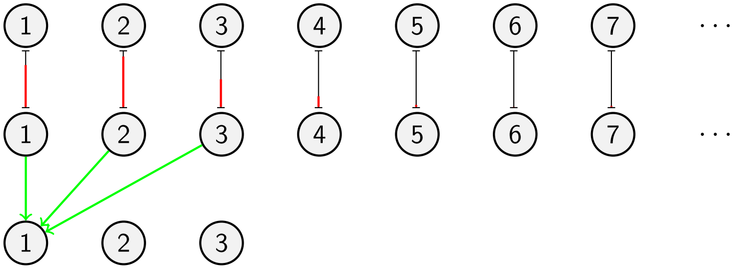
- draw a measure $B^{(m)}$ where $\pi_k^{(m)}$ is the popularity of dish $\theta_k^{(m)}$.
- construct $Z^{(m-1,m)}$ by sampling $Z_{i,\cdot}^{(m-1,m)} \sim \mathrm{BeP}(B^{(m)})$ for the $K^{(m-1)}$ customers entering this restaurant.



Beta process representation for CIBP

To generate sequence of binary matrices, we can recursively:

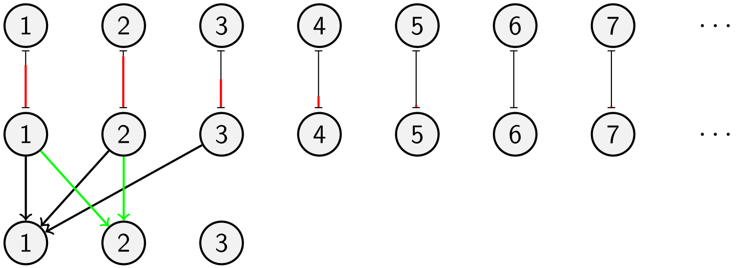
- draw a measure $B^{(m)}$ where $\pi_k^{(m)}$ is the popularity of dish $\theta_k^{(m)}$.
- construct $Z^{(m-1,m)}$ by sampling $Z_{i,\cdot}^{(m-1,m)} \sim \text{BeP}(B^{(m)})$ for the $K^{(m-1)}$ customers entering this restaurant.



Beta process representation for CIBP

To generate sequence of binary matrices, we can recursively:

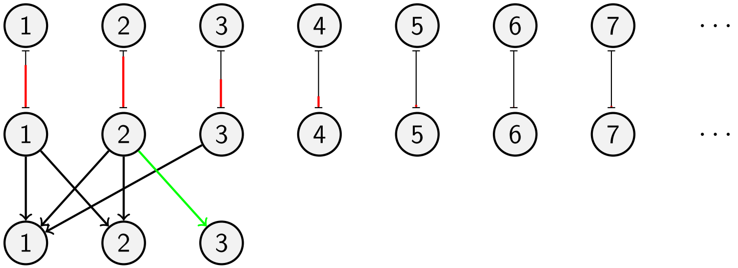
- draw a measure $B^{(m)}$ where $\pi_k^{(m)}$ is the popularity of dish $\theta_k^{(m)}$.
- construct $Z^{(m-1,m)}$ by sampling $Z_{i,\cdot}^{(m-1,m)} \sim \text{BeP}(B^{(m)})$ for the $K^{(m-1)}$ customers entering this restaurant.



Beta process representation for CIBP

To generate sequence of binary matrices, we can recursively:

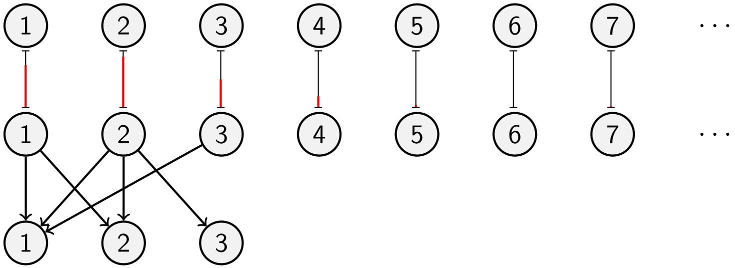
- draw a measure $B^{(m)}$ where $\pi_k^{(m)}$ is the popularity of dish $\theta_k^{(m)}$.
- construct $Z^{(m-1,m)}$ by sampling $Z_{i,\cdot}^{(m-1,m)} \sim \text{BeP}(B^{(m)})$ for the $K^{(m-1)}$ customers entering this restaurant.



Beta process representation for CIBP

To generate sequence of binary matrices, we can recursively:

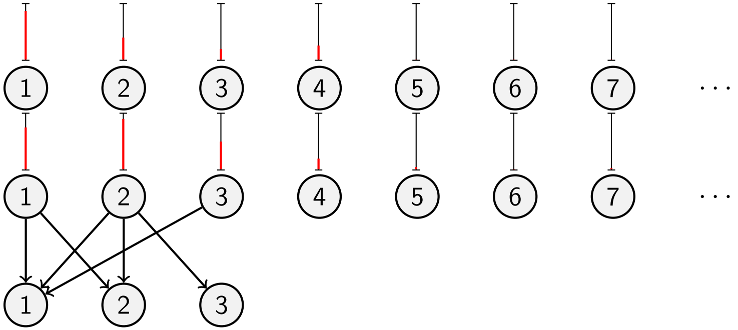
- draw a measure $B^{(m)}$ where $\pi_k^{(m)}$ is the popularity of dish $\theta_k^{(m)}$.
- construct $Z^{(m-1,m)}$ by sampling $Z_{i,\cdot}^{(m-1,m)} \sim \text{BeP}(B^{(m)})$ for the $K^{(m-1)}$ customers entering this restaurant.



Beta process representation for CIBP

To generate sequence of binary matrices, we can recursively:

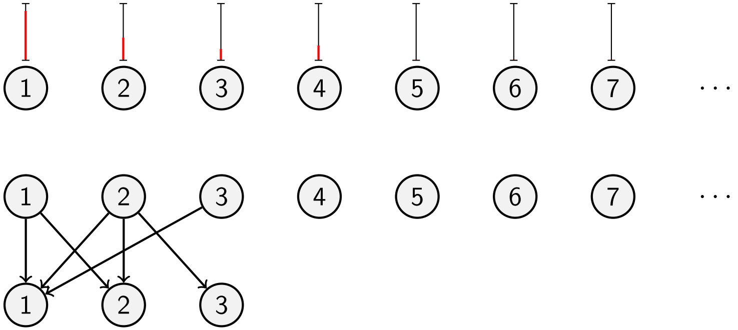
- draw a measure $B^{(m)}$ where $\pi_k^{(m)}$ is the popularity of dish $\theta_k^{(m)}$.
- construct $Z^{(m-1,m)}$ by sampling $Z_{i,\cdot}^{(m-1,m)} \sim \text{BeP}(B^{(m)})$ for the $K^{(m-1)}$ customers entering this restaurant.



Beta process representation for CIBP

To generate sequence of binary matrices, we can recursively:

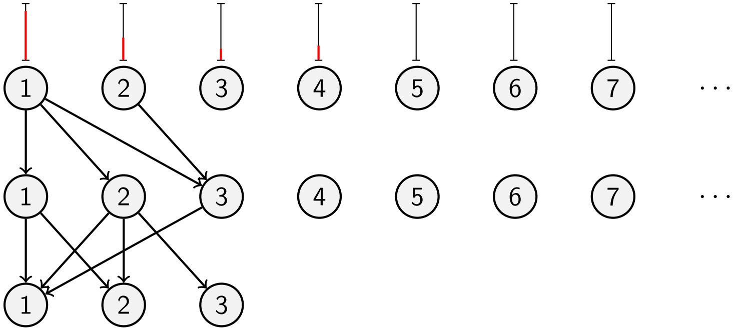
- draw a measure $B^{(m)}$ where $\pi_k^{(m)}$ is the popularity of dish $\theta_k^{(m)}$.
- construct $Z^{(m-1,m)}$ by sampling $Z_{i,\cdot}^{(m-1,m)} \sim \text{BeP}(B^{(m)})$ for the $K^{(m-1)}$ customers entering this restaurant.



Beta process representation for CIBP

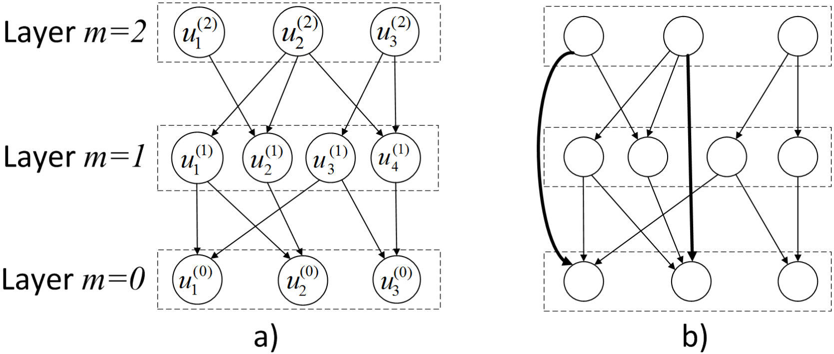
To generate sequence of binary matrices, we can recursively:

- draw a measure $B^{(m)}$ where $\pi_k^{(m)}$ is the popularity of dish $\theta_k^{(m)}$.
- construct $Z^{(m-1,m)}$ by sampling $Z_{i,\cdot}^{(m-1,m)} \sim \text{BeP}(B^{(m)})$ for the $K^{(m-1)}$ customers entering this restaurant.



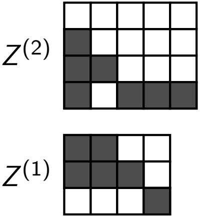
Extending the CIBP

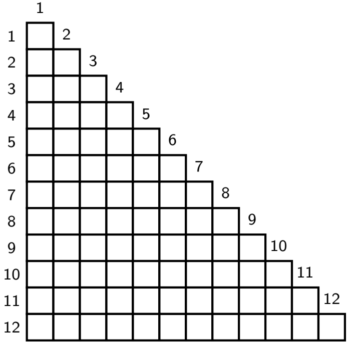
Structure of the belief network

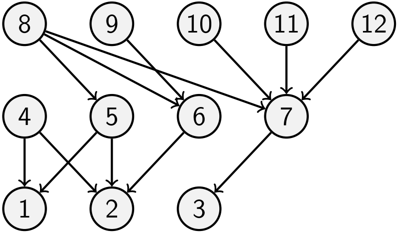


Triangular binary matrix representation

- Layer exchangeability is ensured by crosses.
- Adjacency matrix is IBP subdiagonal by blocks.
- Hierarchical BP allows jumping connections.

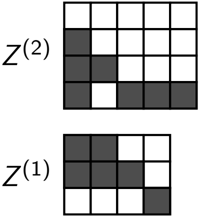


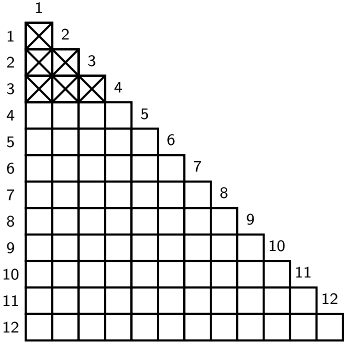


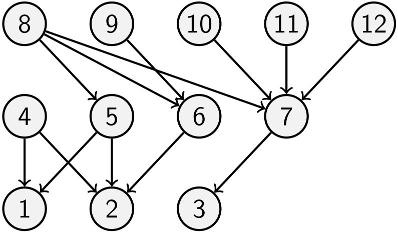


Triangular binary matrix representation

- Layer exchangeability is ensured by crosses.
- Adjacency matrix is IBP subdiagonal by blocks.
- Hierarchical BP allows jumping connections.

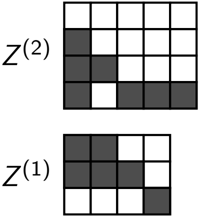


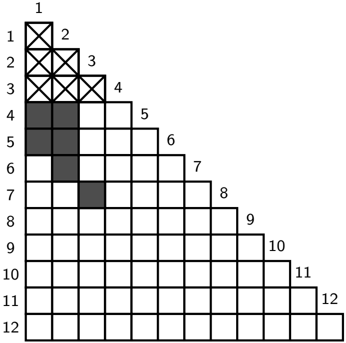


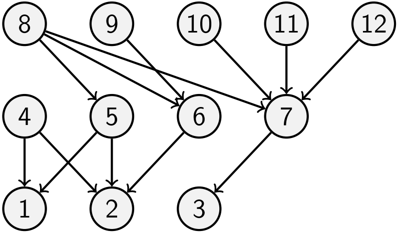


Triangular binary matrix representation

- Layer exchangeability is ensured by crosses.
- Adjacency matrix is IBP subdiagonal by blocks.
- Hierarchical BP allows jumping connections.

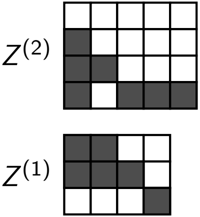


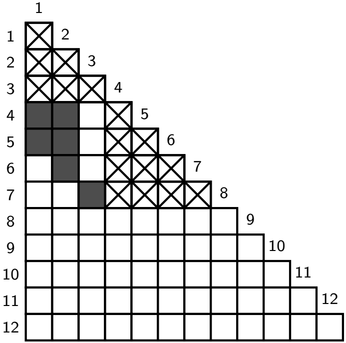


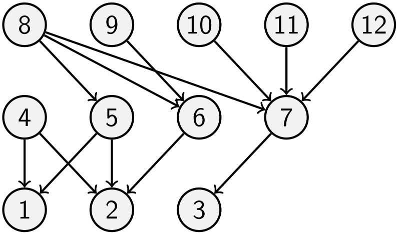


Triangular binary matrix representation

- Layer exchangeability is ensured by crosses.
- Adjacency matrix is IBP subdiagonal by blocks.
- Hierarchical BP allows jumping connections.

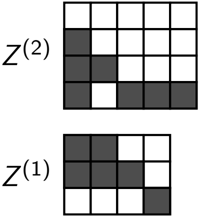


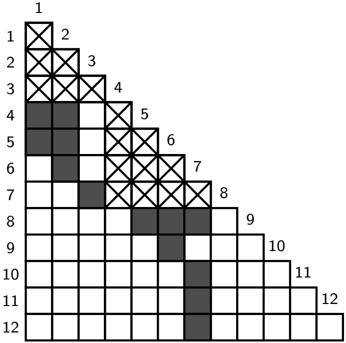


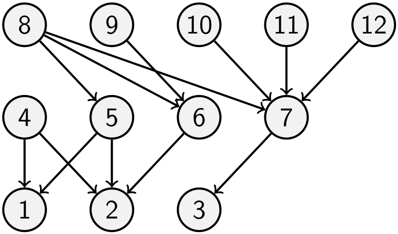


Triangular binary matrix representation

- Layer exchangeability is ensured by crosses.
- Adjacency matrix is IBP subdiagonal by blocks.
- Hierarchical BP allows jumping connections.

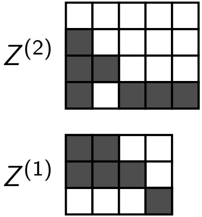


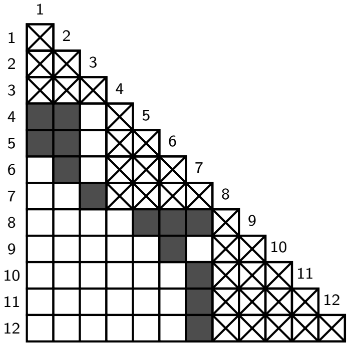


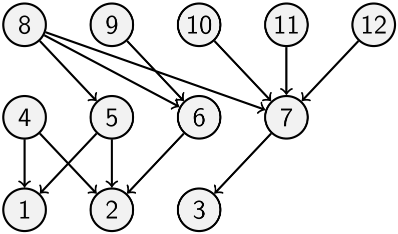


Triangular binary matrix representation

- Layer exchangeability is ensured by crosses.
- Adjacency matrix is IBP subdiagonal by blocks.
- Hierarchical BP allows jumping connections.

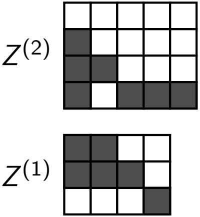


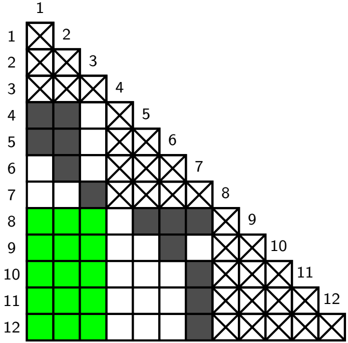


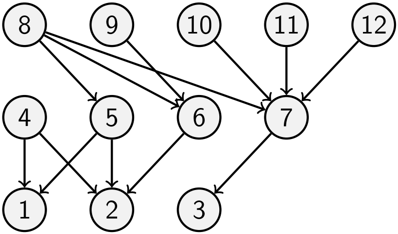


Triangular binary matrix representation

- Layer exchangeability is ensured by crosses.
- Adjacency matrix is IBP subdiagonal by blocks.
- Hierarchical BP allows jumping connections.

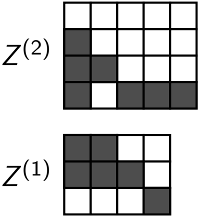


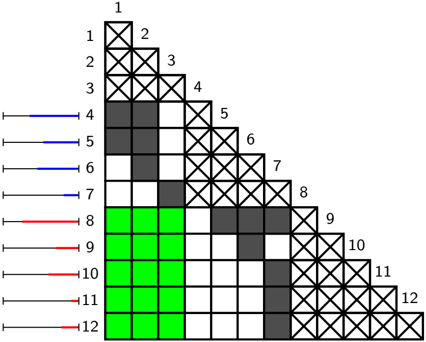


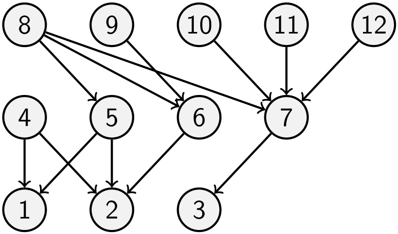


Triangular binary matrix representation

- Layer exchangeability is ensured by crosses.
- Adjacency matrix is IBP subdiagonal by blocks.
- Hierarchical BP allows jumping connections.







Extended cascading Indian buffet process metaphor

Every customers are associated to an Indian buffet restaurant

- The $K^{(m-1)}$ **customers** from restaurant $m-1$ enter restaurant m and select dishes according to the usual IBP.
- Next, the $K^{(s)}$ **visitors** from previous restaurants $s < m-1$ enters and select dishes according to probability:

$$\frac{\beta'}{\beta' + K(s)} \frac{n_k^{(m)}}{\beta + K(m)} + \frac{n_k^{(s,m)}}{\beta' + K(s)}$$

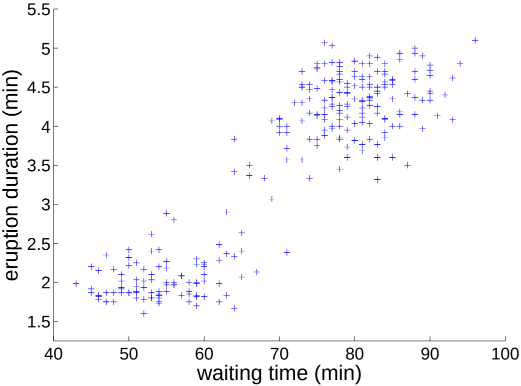
where $n_k^{(s,m)}$ denote the number of times dish k has been selected by preceding members of its visiting group s.

Results

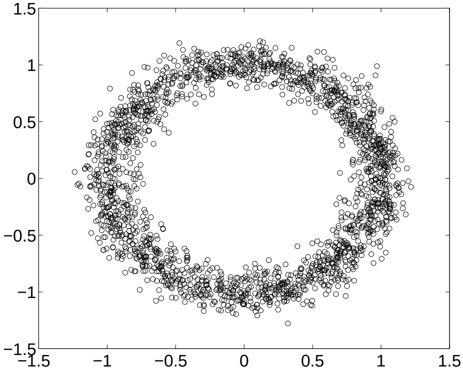
Experiments

- Learn the generative process of the data
- Produce fantasy data with posterior models
- Measure the testset/fantasy divergence
- Evaluate the structural complexity

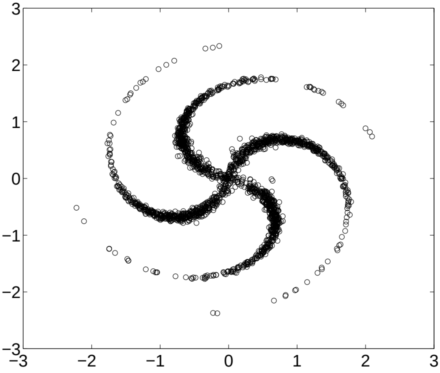
Dataset - Geyser



Dataset - Ring



Dataset - Pinwheel



More datasets

- Iris : 4 continuous dimensions
- Abalone : 7 continuous + 2 discretes dimensions

Results - Divergence

Table: Kullback-Leibler divergence estimations

Dataset	CIBP	eCIBP	DPMoG	KDE
Ring	0.049 ± 0.055	$\mathbf{0.030} \pm \mathbf{0.034}$	0.051 ± 0.029	0.085 ± 0.034
Pinwheel	0.162 ± 0.049	0.161 ± 0.041	$\mathbf{0.154} \pm \mathbf{0.080}$	0.216 ± 0.043
Geyser	0.078 ± 0.150	$\mathbf{0.075} \pm \mathbf{0.145}$	0.077 ± 0.120	0.143 ± 0.099
Iris	0.207 ± 0.333	$\mathbf{0.177} \pm \mathbf{0.280}$	0.231 ± 0.269	0.305 ± 0.227
Abalone	0.074 ± 0.080	$\mathbf{0.070} \pm \mathbf{0.108}$	4.760 ± 0.220	2.934 ± 0.134

Results - Structure complexity

Table: Total number of units

Dataset	CIBP	eCIBP
Ring	11.6 ± 4.1	$\mathbf{9.3} \pm \mathbf{3.2}$
Pinwheel	22.4 ± 3.0	$\mathbf{19.7} \pm \mathbf{1.6}$
Geyser	9.1 ± 4.3	$\mathbf{8.7} \pm \mathbf{3.9}$
Iris	14.2 ± 4.4	$\mathbf{10.3} \pm \mathbf{3.4}$
Abalone	54.2 ± 12.3	$\mathbf{45.2} \pm \mathbf{11.2}$

Results - Structure complexity

Table: Total number of edges

Dataset	CIBP	eCIBP
Ring	$\mathbf{20.3 \pm 8.4}$	20.6 ± 7.3
Pinwheel	105.4 ± 11.1	$\mathbf{104.1 \pm 9.3}$
Geyser	$\mathbf{13.1 \pm 9.9}$	$\mathbf{13.1 \pm 9.5}$
Iris	28.4 ± 11.4	$\mathbf{20.5 \pm 7.2}$
Abalone	233.8 ± 23.0	$\mathbf{210.4 \pm 23.8}$

End