Nonparametric Bayesian Methods: Part IV

Tamara Broderick
ITT Career Development Assistant Professor
Electrical Engineering & Computer Science
MIT
Nonparametric Bayesian Methods: Part IV

[ slides, code: http://www.tamarabroderick.com/tutorials.html ]

Tamara Broderick
ITT Career Development Assistant Professor
Electrical Engineering & Computer Science
MIT
social: Facebook, Twitter, email
biological: ecological, protein, gene
transportation: roads, railways
Probabilistic models for graphs

\[ p(\cdot) \]

social: Facebook, Twitter, email
biological: ecological, protein, gene
transportation: roads, railways

[Holland et al 1983; Kemp et al 2006; Xu et al 2007; Airoldi et al 2008; Lloyd et al 2012]
Probabilistic models for graphs

\[ p(\cdot) \]

- Rich relationships, coherent uncertainties, prior info

social: Facebook, Twitter, email

biological: ecological, protein, gene

transportation: roads, railways

[Holland et al 1983; Kemp et al 2006; Xu et al 2007; Airoldi et al 2008; Lloyd et al 2012]
Probabilistic models for graphs

$p(\cdot)$

- Rich relationships, coherent uncertainties, prior info
- Example models: Stochastic block, mixed membership stochastic block, infinite relational, and many more

social: Facebook, Twitter, email
biological: ecological, protein, gene
transportation: roads, railways

[Holland et al 1983; Kemp et al 2006; Xu et al 2007; Airoldi et al 2008]
Probabilistic models for graphs

- Rich relationships, coherent uncertainties, prior info
- Example models: Stochastic block, mixed membership stochastic block, infinite relational, and many more

social: Facebook, Twitter, email
biological: ecological, protein, gene
transportation: roads, railways

[Holland et al 1983; Kemp et al 2006; Xu et al 2007; Airoldi et al 2008; Lloyd et al 2012]
Probabilistic models for graphs

\[ p(\cdot) \]

- Rich relationships, coherent uncertainties, prior info
- Example models: Stochastic block, mixed membership stochastic block, infinite relational, and many more
- Assume: Adding more data doesn’t change distribution of earlier data (projectivity)

[Holland et al 1983; Kemp et al 2006; Xu et al 2007; Airoldi et al 2008; Lloyd et al 2012]
Probabilistic models for graphs

- Rich relationships, coherent uncertainties, prior info
- Example models: Stochastic block, mixed membership stochastic block, infinite relational, and many more
- Assume: Adding more data doesn’t change distribution of earlier data (*projectivity*)
- **Problem**: model misspecification, dense graphs

social: Facebook, Twitter, email
biological: ecological, protein, gene
transportation: roads, railways

[Holland et al 1983; Kemp et al 2006; Xu et al 2007; Airoldi et al 2008; Lloyd et al 2012]
Probabilistic models for graphs

- Rich relationships, coherent uncertainties, prior info
- Example models: Stochastic block, mixed membership stochastic block, infinite relational, and many more
- Assume: Adding more data doesn’t change distribution of earlier data (*projectivity*)
- **Problem**: model misspecification, dense graphs
- **Solution**: a new framework for sparse graphs

- Social: Facebook, Twitter, email
- Biological: ecological, protein, gene
- Transportation: roads, railways

[Holland et al 1983; Kemp et al 2006; Xu et al 2007; Airoldi et al 2008; Lloyd et al 2012]
Probabilistic models for graphs

\[ p(\cdot) \]

- Rich relationships, coherent uncertainties, prior info
- Example models: Stochastic block, mixed membership stochastic block, infinite relational, and many more
- Assume: Adding more data doesn’t change distribution of earlier data (projectivity)
- **Problem**: model misspecification, dense graphs
- **Solution**: a new framework for sparse graphs

social: Facebook, Twitter, email
biological: ecological, protein, gene
transportation: roads, railways

[Holland et al 1983; Kemp et al 2006; Xu et al 2007; Airoldi et al 2008; Lloyd et al 2012]
Probabilistic models for graphs

- Rich relationships, coherent uncertainties, prior info
- Example models: Stochastic block, mixed membership stochastic block, infinite relational, and many more
- Assume: Adding more data doesn’t change distribution of earlier data (*projectivity*)
- **Problem**: model misspecification, dense graphs
- **Solution**: a *new framework* for sparse graphs

- **social**: Facebook, Twitter, email
- **biological**: ecological, protein, gene
- **transportation**: roads, railways

[Holland et al 1983; Kemp et al 2006; Xu et al 2007; Airoldi et al 2008; Lloyd et al 2012]
Probabilistic models for graphs

\[ p(\cdot) \]

- Rich relationships, coherent uncertainties, prior info
- Example models: Stochastic block, mixed membership
  stochastic block, infinite relational, and many more
- Assume: Adding more data doesn’t change distribution of
  earlier data (projectivity)
- **Problem**: model misspecification, dense graphs
- **Solution**: a new framework for **sparse graphs**

social: Facebook, Twitter, email
biological: ecological, protein, gene
transportation: roads, railways

[Holland et al 1983; Kemp et al 2006; Xu et al 2007; Airoldi et al 2008; Lloyd et al 2012]
Probabilistic models for graphs

- Rich relationships, coherent uncertainties, prior info
- Example models: Stochastic block, mixed membership stochastic block, infinite relational, and many more
- Assume: Adding more data doesn't change distribution of earlier data (projectivity)
- **Problem**: model misspecification, dense graphs
- **Solution**: a new framework for sparse graphs
  - Concurrent & independent graphs work by Crane & Dempsey

social: Facebook, Twitter, email
biological: ecological, protein, gene
transportation: roads, railways
Sequence of graphs

$G$
Sequence of graphs

\[ G_1 \]

\[ G \]
Sequence of graphs

$G_1$  

$G_2$  

$G$
Sequence of graphs

$G_1$  

$G_2$  

$G_3$  

$G$
Sequence of graphs

$G_1$  $G_2$  $G_3$  $G_4$

$G$
Sequence of graphs

$G_1$  $G_2$  $G_3$  $G_4$
Sequence of graphs

$G_1 \quad G_2 \quad G_3 \quad G_4$

If $\#\text{nodes}(G_n) \to \infty$, 
Sequence of graphs

G<sub>1</sub>  |  G<sub>2</sub>  |  G<sub>3</sub>  |  G<sub>4</sub>

If \( \#\text{nodes}(G_n) \to \infty \),

- *Dense* graph sequence  \( \#\text{edges}(G_n) \geq c \cdot [\#\text{nodes}(G_n)]^2 \)
Sequence of graphs

\[
\begin{align*}
G_1 & \quad \vdots \quad G_2 & \quad \vdots \quad G_3 & \quad \vdots \quad G_4 \\
& & & \\
\end{align*}
\]

If \( \#\text{nodes}(G_n) \to \infty \),

- **Dense graph sequence** \( \#\text{edges}(G_n) \geq c \cdot [\#\text{nodes}(G_n)]^2 \)
- **Sparse graph sequence** \( \#\text{edges}(G_n) \in o([\#\text{nodes}(G_n)]^2) \)
The Old Way: Nodes

$G_1$
The Old Way: Nodes

$G_1$
The Old Way: Nodes

$G_1$

$G_2$
The Old Way: Nodes

$G_1$

$G_2$

$G_3$

[Hoover 1979, Aldous 1981]
The Old Way: Nodes

\[ G_1 \]

\[ G_2 \]

\[ G_3 \]

\[ G_4 \]

[Hoover 1979, Aldous 1981]
The Old Way: Exchangeability

$G_1$  $G_2$  $G_3$  $G_4$

[Hoover 1979, Aldous 1981]
The Old Way: Exchangeability

$G_1$

$G_2$

$G_3$

$G_4$

[Hoover 1979, Aldous 1981]
The Old Way: Exchangeability

\[ G_1 \]

\[ G_2 \]

\[ G_3 \]

\[ G_4 \]

\[ p( ) \]

[Hoover 1979, Aldous 1981]
The Old Way: Exchangeability

$G_1$  

1

$G_2$  

1

1

$G_3$  

1

2

$G_4$  

1

2

4

3

$p(\begin{array}{ccc} 2 & 3 \\ 1 & 4 \end{array}) = p(\begin{array}{ccc} 4 & 1 \\ 2 & 3 \end{array})$

[Hoover 1979, Aldous 1981]
The Old Way: Node exchangeability

\[ p(\quad ) = p(\quad ) \]

[Hoover 1979, Aldous 1981]
Aldous-Hoover

[Hoover 1979, Aldous 1981]
Aldous-Hoover

$W(x, y)$

[Hoover 1979, Aldous 1981]
Aldous-Hoover

\[ W(x, y) \]

[Hoover 1979, Aldous 1981]
Aldous-Hoover

\[ W(x, y) \]

[Hoover 1979, Aldous 1981]
Aldous-Hoover

$W(x, y)$
Aldous-Hoover

\[ W(x, y) \]
Aldous-Hoover

\[ W(x, y) \]
Aldous-Hoover

\[ W(x, y) \]
Aldous-Hoover

$W(x, y)$

[Hoover 1979, Aldous 1981]
Aldous-Hoover

\[ W(x, y) \]

[Hoover 1979, Aldous 1981]
Aldous-Hoover

\[ W(x, y) \]

[Hoover 1979, Aldous 1981]
Aldous-Hoover
Aldous-Hoover

\[ W(x, y) \]

[Hoover 1979, Aldous 1981]
Aldous-Hoover

\[ W(x, y) \]

[Hoover 1979, Aldous 1981]
Aldous-Hoover

\[ W(x, y) \]

[Hoover 1979, Aldous 1981]
Thm (AH). Every node-exchangeable graph has a graphon rep
Thm (AH). Every node-exchangeable graph has a graphon rep

\[ \mathbb{E}[\#\text{edges}(G_n)] \]
Thm (AH). Every node-exchangeable graph has a graphon rep

\[ \mathbb{E}[\#\text{edges}(G_n)] = \mathbb{E} \left[ \binom{n}{2} \frac{1}{2} \int_0^1 \int_0^1 W(x, y) \, dx \, dy \right] \]
Thm (AH). Every node-exchangeable graph has a graphon rep

\[ \mathbb{E} [\# \text{edges}(G_n)] = \mathbb{E} \left[ \binom{n}{2} \frac{1}{2} \int_0^1 \int_0^1 W(x, y) \, dx \, dy \right] \]

\[ \sim cn^2 \]

[Hoover 1979, Aldous 1981, Orbanz, Roy 2015]
Aldous-Hoover

Thm (AH). Every node-exchangeable graph has a *graphon* rep

\[
E[\#\text{edges}(G_n)] = E \left[ \binom{n}{2} \frac{1}{2} \int_0^1 \int_0^1 W(x, y) \, dx \, dy \right]
\]

\[
\sim cn^2 = c \cdot [\#\text{nodes}(G_n)]^2
\]

[Hoover 1979, Aldous 1981, Orbanz, Roy 2015]
Aldous-Hoover

\[ W(x, y) \]

Thm (AH). Every node-exchangeable graph has a graphon rep

\[
\mathbb{E}[\#\text{edges}(G_n)] = \mathbb{E} \left[ \binom{n}{2} \frac{1}{2} \int_0^1 \int_0^1 W(x, y) \, dx \, dy \right] \\
\sim cn^2 = c \cdot [\#\text{nodes}(G_n)]^2
\]

Cor. Every node-exch graph sequence is dense (or empty) a.s.

[Hoover 1979, Aldous 1981, Orbanz, Roy 2015]
Thm (AH). Every node-exchangeable graph has a graphon rep

\[ \mathbb{E}[\#\text{edges}(G_n)] = \mathbb{E} \left[ \binom{n}{2} \frac{1}{2} \int_0^1 \int_0^1 W(x, y) \, dx \, dy \right] \]

\[ \sim cn^2 = c \cdot [\#\text{nodes}(G_n)]^2 \]

Cor. Every node-exch graph sequence is dense (or empty) a.s.

Intuition: To a given node, all other nodes look the same.

[Hoover 1979, Aldous 1981, Orbanz, Roy 2015]
Aldous-Hoover

\[ W(x, y) \]

Thm (AH). Every node-exchangeable graph has a graphon rep

\[ \mathbb{E}[\#\text{edges}(G_n)] = \mathbb{E} \left[ \binom{n}{2} \frac{1}{2} \int_0^1 \int_0^1 W(x, y) \, dx \, dy \right] \]

\[ \sim cn^2 = c \cdot [\#\text{nodes}(G_n)]^2 \]

Cor. Every node-exch graph sequence is dense (or empty) a.s.

Intuition: To a given node, all other nodes look the same.

[Caron, Fox 2014; Veitch, Roy 2015; Borgs, Chayes, Cohn, Holden 2016; Broderick, Cai 2015; Cai, Broderick 2015a,b; Crane, Dempsey 2015a,b, 2016a; Cai, Campbell, Broderick 2016]
A New Way: Edges

$G_1$
A New Way: Edges

$G_1$
A New Way: Edges

$G_1$  

$G_2$

[Broderick, Cai 2015; Cai, Broderick 2015a,b; Crane, Dempsey 2015a,b,16a; Cai, Campbell, Broderick 2016]
A New Way: Edges

$G_1$  

$G_2$  

$G_3$  

Broderick, Cai 2015; Cai, Broderick 2015a,b; Crane, Dempsey 2015a,b,16a; Cai, Campbell, Broderick 2016
A New Way: Edges

$G_1$  

$G_2$  

$G_3$  

$G_4$  

[Broderick, Cai 2015; Cai, Broderick 2015a,b; Crane, Dempsey 2015a,b,16a; Cai, Campbell, Broderick 2016]
Edge exchangeability

$G_1$ $G_2$ $G_3$ $G_4$

[Broderick, Cai 2015; Cai, Broderick 2015a,b; Crane, Dempsey 2015a,b,16a; Cai, Campbell, Broderick 2016]
Edge exchangeability

$G_1$  $G_2$  $G_3$  $G_4$

[Broderick, Cai 2015; Cai, Broderick 2015a,b; Crane, Dempsey 2015a,b,16a; Cai, Campbell, Broderick 2016]
Edge exchangeability

\[ G_1 \]
\[ G_2 \]
\[ G_3 \]
\[ G_4 \]

\[ p(1, 2, 3, 4) \]

[Broderick, Cai 2015; Cai, Broderick 2015a,b; Crane, Dempsey 2015a,b,16a; Cai, Campbell, Broderick 2016]
Edge exchangeability

\[ p(G_1) = p(G_2) \]

\[ p(G_3) = p(G_4) \]

[Broderick, Cai 2015; Cai, Broderick 2015a,b; Crane, Dempsey 2015a,b,16a; Cai, Campbell, Broderick 2016]
Thm (CCB). A wide class of edge-exchangeable graph models yields sparse graph sequences.

\[ p(1, 2, 3, 4) = p(2, 4, 1, 3) \]
What we know so far

- Goal 1: characterization theorem for edge-exchangeable graphs
- Goal 2: sparsity theorem for edge-exchangeable graphs
Clustering
Clustering

“Clusters”
Clustering

"Clusters"
# Clustering

<table>
<thead>
<tr>
<th></th>
<th>Cat</th>
<th>Dog</th>
<th>Mouse</th>
<th>Lizard</th>
<th>Sheep</th>
</tr>
</thead>
<tbody>
<tr>
<td>Picture 1</td>
<td><img src="image1" alt="Cat" /></td>
<td><img src="image2" alt="Dog" /></td>
<td><img src="image3" alt="Mouse" /></td>
<td><img src="image4" alt="Lizard" /></td>
<td><img src="image5" alt="Sheep" /></td>
</tr>
<tr>
<td>Picture 2</td>
<td><img src="image1" alt="Cat" /></td>
<td><img src="image2" alt="Dog" /></td>
<td><img src="image3" alt="Mouse" /></td>
<td><img src="image4" alt="Lizard" /></td>
<td><img src="image5" alt="Sheep" /></td>
</tr>
<tr>
<td>Picture 3</td>
<td><img src="image1" alt="Cat" /></td>
<td><img src="image2" alt="Dog" /></td>
<td><img src="image3" alt="Mouse" /></td>
<td><img src="image4" alt="Lizard" /></td>
<td><img src="image5" alt="Sheep" /></td>
</tr>
<tr>
<td>Picture 4</td>
<td><img src="image1" alt="Cat" /></td>
<td><img src="image2" alt="Dog" /></td>
<td><img src="image3" alt="Mouse" /></td>
<td><img src="image4" alt="Lizard" /></td>
<td><img src="image5" alt="Sheep" /></td>
</tr>
<tr>
<td>Picture 5</td>
<td><img src="image1" alt="Cat" /></td>
<td><img src="image2" alt="Dog" /></td>
<td><img src="image3" alt="Mouse" /></td>
<td><img src="image4" alt="Lizard" /></td>
<td><img src="image5" alt="Sheep" /></td>
</tr>
<tr>
<td>Picture 6</td>
<td><img src="image1" alt="Cat" /></td>
<td><img src="image2" alt="Dog" /></td>
<td><img src="image3" alt="Mouse" /></td>
<td><img src="image4" alt="Lizard" /></td>
<td><img src="image5" alt="Sheep" /></td>
</tr>
<tr>
<td>Picture 7</td>
<td><img src="image1" alt="Cat" /></td>
<td><img src="image2" alt="Dog" /></td>
<td><img src="image3" alt="Mouse" /></td>
<td><img src="image4" alt="Lizard" /></td>
<td><img src="image5" alt="Sheep" /></td>
</tr>
</tbody>
</table>

- Groups: clusters
# Clustering

<table>
<thead>
<tr>
<th></th>
<th>Cat</th>
<th>Dog</th>
<th>Mouse</th>
<th>Lizard</th>
<th>Sheep</th>
</tr>
</thead>
<tbody>
<tr>
<td>Picture 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Picture 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Picture 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Picture 4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Picture 5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Picture 6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Picture 7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Groups: clusters
- Exchangeable
Feature allocation

<table>
<thead>
<tr>
<th>Cat</th>
<th>Dog</th>
<th>Mouse</th>
<th>Lizard</th>
<th>Sheep</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Picture 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Picture 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Picture 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Picture 4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Picture 5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Picture 6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Picture 7</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# Feature allocation

<table>
<thead>
<tr>
<th></th>
<th>Cat</th>
<th>Dog</th>
<th>Mouse</th>
<th>Lizard</th>
<th>Sheep</th>
</tr>
</thead>
<tbody>
<tr>
<td>Picture 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Picture 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Picture 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Picture 4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Picture 5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Picture 6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Picture 7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

• Groups: features
Feature allocation

- Groups: features
- Exchangeable

[Broderick, Jordan, Pitman 2013; Broderick, Pitman, Jordan 2013]
Graph

Edge 1

Edge 2

Edge 3

Edge 4

Edge 5

Edge 6

Edge 7
Graph

- Groups: vertices
Graph

<table>
<thead>
<tr>
<th>Edge 1</th>
<th>Edge 2</th>
<th>Edge 3</th>
<th>Edge 4</th>
<th>Edge 5</th>
<th>Edge 6</th>
<th>Edge 7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cat</td>
<td>Dog</td>
<td>Mouse</td>
<td>Lizard</td>
<td>Sheep</td>
<td>Cat</td>
<td>Dog</td>
</tr>
</tbody>
</table>

- Groups: vertices
- Edge-exchangeable
Graph

<table>
<thead>
<tr>
<th></th>
<th>Cat</th>
<th>Dog</th>
<th>Mouse</th>
<th>Lizard</th>
<th>Sheep</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edge 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Groups: vertices
- Edge-exchangeable
Graph

- Groups: vertices
- Edge-exchangeable
### Graph

<table>
<thead>
<tr>
<th></th>
<th>Cat</th>
<th>Dog</th>
<th>Mouse</th>
<th>Lizard</th>
<th>Sheep</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edge 1</td>
<td></td>
<td><img src="#" alt="Red" /></td>
<td></td>
<td></td>
<td><img src="#" alt="Yellow" /></td>
</tr>
<tr>
<td>Edge 2</td>
<td><img src="#" alt="Red" /></td>
<td></td>
<td></td>
<td></td>
<td><img src="#" alt="Green" /></td>
</tr>
<tr>
<td>Edge 3</td>
<td><img src="#" alt="Blue" /></td>
<td><img src="#" alt="Blue" /></td>
<td></td>
<td></td>
<td><img src="#" alt="Yellow" /></td>
</tr>
<tr>
<td>Edge 4</td>
<td><img src="#" alt="Blue" /></td>
<td><img src="#" alt="Blue" /></td>
<td><img src="#" alt="Orange" /></td>
<td></td>
<td><img src="#" alt="Yellow" /></td>
</tr>
<tr>
<td>Edge 5</td>
<td><img src="#" alt="Blue" /></td>
<td><img src="#" alt="Blue" /></td>
<td></td>
<td></td>
<td><img src="#" alt="Yellow" /></td>
</tr>
<tr>
<td>Edge 6</td>
<td><img src="#" alt="Blue" /></td>
<td><img src="#" alt="Blue" /></td>
<td><img src="#" alt="Green" /></td>
<td></td>
<td><img src="#" alt="Yellow" /></td>
</tr>
<tr>
<td>Edge 7</td>
<td><img src="#" alt="Blue" /></td>
<td><img src="#" alt="Blue" /></td>
<td></td>
<td></td>
<td><img src="#" alt="Yellow" /></td>
</tr>
</tbody>
</table>

- Groups: vertices
- Edge-exchangeable
Graph

- Groups: vertices
- Edge-exchangeable
**Graph**

<table>
<thead>
<tr>
<th>Cat</th>
<th>Dog</th>
<th>Mouse</th>
<th>Lizard</th>
<th>Sheep</th>
</tr>
</thead>
<tbody>
<tr>
<td>🟥</td>
<td>🟥</td>
<td></td>
<td></td>
<td>🟢</td>
</tr>
<tr>
<td>🟥</td>
<td>🟥</td>
<td></td>
<td></td>
<td>🟢</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>🟢</td>
</tr>
<tr>
<td>🟦</td>
<td></td>
<td></td>
<td></td>
<td>🟢</td>
</tr>
<tr>
<td>🟦</td>
<td>🟦</td>
<td></td>
<td></td>
<td>🟢</td>
</tr>
<tr>
<td>🟦</td>
<td>🟦</td>
<td></td>
<td></td>
<td>🟢</td>
</tr>
<tr>
<td>🟦</td>
<td>🟦</td>
<td></td>
<td></td>
<td>🟢</td>
</tr>
</tbody>
</table>

- Groups: vertices
- Edge-exchangeable
Graph

- Groups: vertices
- Edge-exchangeable

<table>
<thead>
<tr>
<th></th>
<th>Cat</th>
<th>Dog</th>
<th>Mouse</th>
<th>Lizard</th>
<th>Sheep</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edge 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Graph

- Groups: vertices
- Edge-exchangeable
### Graph

<table>
<thead>
<tr>
<th></th>
<th>Cat</th>
<th>Dog</th>
<th>Mouse</th>
<th>Lizard</th>
<th>Sheep</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edge 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Groups: vertices
- Edge-exchangeable
### Graph

<table>
<thead>
<tr>
<th>Edge 1</th>
<th>Edge 2</th>
<th>Edge 3</th>
<th>Edge 4</th>
<th>Edge 5</th>
<th>Edge 6</th>
<th>Edge 7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cat</td>
<td>Dog</td>
<td>Mouse</td>
<td>Lizard</td>
<td>Sheep</td>
<td>Cat</td>
<td>Dog</td>
</tr>
</tbody>
</table>

- Groups: vertices
- Edge-exchangeable
**Graph**

- Groups: vertices
- Edge-exchangeable

<table>
<thead>
<tr>
<th></th>
<th>Cat</th>
<th>Dog</th>
<th>Mouse</th>
<th>Lizard</th>
<th>Sheep</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edge 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Graph

- Groups: vertices
- Edge-exchangeable
<table>
<thead>
<tr>
<th>Edge 1</th>
<th>Cat</th>
<th>Dog</th>
<th>Mouse</th>
<th>Lizard</th>
<th>Sheep</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edge 2</td>
<td>Cat</td>
<td>Dog</td>
<td>Mouse</td>
<td>Lizard</td>
<td>Sheep</td>
</tr>
<tr>
<td>Edge 3</td>
<td>Cat</td>
<td>Dog</td>
<td>Mouse</td>
<td>Lizard</td>
<td>Sheep</td>
</tr>
<tr>
<td>Edge 4</td>
<td>Cat</td>
<td>Dog</td>
<td>Mouse</td>
<td>Lizard</td>
<td>Sheep</td>
</tr>
<tr>
<td>Edge 5</td>
<td>Cat</td>
<td>Dog</td>
<td>Mouse</td>
<td>Lizard</td>
<td>Sheep</td>
</tr>
<tr>
<td>Edge 6</td>
<td>Cat</td>
<td>Dog</td>
<td>Mouse</td>
<td>Lizard</td>
<td>Sheep</td>
</tr>
<tr>
<td>Edge 7</td>
<td>Cat</td>
<td>Dog</td>
<td>Mouse</td>
<td>Lizard</td>
<td>Sheep</td>
</tr>
</tbody>
</table>

- Groups: vertices
- Edge-exchangeable
Graph

- Groups: vertices
- Edge-exchangeable
# Graph

<table>
<thead>
<tr>
<th></th>
<th>Cat</th>
<th>Dog</th>
<th>Mouse</th>
<th>Lizard</th>
<th>Sheep</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edge 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge 7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Groups: vertices
- Edge-exchangeable
Graph

- Groups: vertices
- Edge-exchangeable
Exchangeable clustering distributions are characterized.

What about:
- Exchangeable feature allocations?
- Edge-exchangeable graphs?
Exchangeable probability functions

\[ P(\begin{array}{ccc}
1 & 2 & \cdots & K \\
N & & & \\
& & & \\
\end{array}) = p(S_N, 1, \ldots, S_{N,K}) \]

[Pitman 1995]
Exchangeable probability functions

\[ P(1, 2, \ldots, K) = p(S_{N,1}, \ldots, S_{N,K}) \]
Exchangeable probability functions

\[ \mathbb{P}( \cdots ) = p(S_{N,1}, \ldots, S_{N,K}) \]

Size of Kth cluster

[Ref: Pitman 1995]
Exchangeable probability functions

exchangeable \textbf{partition} probability function (EPPF)

\[ \mathbb{P}(\ldots) = p(S_{N,1}, \ldots, S_{N,K}) \]

Size of Kth cluster
Exchangeable probability functions

Thm (Pitman). Every exchangeable clustering has an exchangeable \textit{partition} probability function (EPPF)

\[ P(S_{N,1}, \ldots, S_{N,K}) = p \]
Exchangeable probability functions

Thm (Pitman). Every exchangeable clustering has an exchangeable \textbf{partition} probability function \((\text{EPPF})\)

\[
\mathbb{P}(1, 2, \ldots, K) = p(N; S_{N,1}, \ldots, S_{N,K})
\]

Size of \(K\)th cluster

[ Pitman 1995 ]
Exchangeable probability functions

Thm (Pitman). Every exchangeable clustering has an exchangeable partition probability function (EP PF)

\[ P(1 \ 2 \ \ldots \ K) = p(N; S_{N,1}, \ldots, S_{N,K}) \]
Exchangeable probability functions

Thm (Pitman). Every exchangeable clustering has an exchangeable partition probability function (EPPF)

\[ P(1, 2, \ldots, K) = p(N; S_{N,1}, \ldots, S_{N,K}) \]

Size of Kth feature
**Exchangeable probability functions**

Thm (Pitman). Every exchangeable clustering has an exchangeable **partition** probability function (EPF).

Exchangeable **feature** probability function (EFPF)

\[
P(1 \ 2 \ \cdots \ K) = p(N; S_{N,1}, \ldots, S_{N,K})
\]

[13]

[Pitman 1995; Broderick, Pitman, Jordan 2013]
Exchangeable probability functions

Thm (Pitman). Every exchangeable clustering has an exchangeable \textbf{partition} probability function (EPPF)

Prop (BPJ). Many, but not all, exchangeable feature allocations have an exchangeable \textbf{feature} probability function (EFPF)

\[
P(1, 2, \ldots, K) = p(N; S_{N,1}, \ldots, S_{N,K})
\]

Size of \(K\)th feature

[Broderick, Pitman, Jordan 2013]
Exchangeable probability functions

Thm (Pitman). Every exchangeable clustering has an exchangeable \textbf{partition} probability function (EPPF)

Prop (BPJ). Many, but not all, exchangeable feature allocations have an exchangeable \textbf{feature} probability function (EFFP)

\[
P(\vdots) = p(N; S_{N,1}, \ldots, S_{N,K})
\]

[13] [Pitman 1995; Broderick, Pitman, Jordan 2013; Campbell, Cai, Broderick 2016]
Exchangeable probability functions

Thm (Pitman). Every exchangeable clustering has an exchangeable **partition** probability function ($\text{EP}PF$)

Prop (BPJ). Many, but not all, exchangeable feature allocations have an exchangeable **feature** probability function ($\text{EF}PF$)

$$\mathbb{P}(N; S_{N,1}, \ldots, S_{N,K}) = p(N; S_{N,1}, \ldots, S_{N,K})$$
Exchangeable probability functions

Thm (Pitman). Every exchangeable clustering has an exchangeable \textit{partition} probability function \((\text{EPPF})\)

Prop (BPJ). Many, but not all, exchangeable feature allocations have an exchangeable \textit{feature} probability function \((\text{EIFF})\)

\[
P(\text{vertex} = 1 \ 2 \ \cdots \ K) = p(N; S_{N,1}, \ldots, S_{N,K})
\]
Exchangeable probability functions

Thm (Pitman). Every exchangeable clustering has an exchangeable partition probability function (EPF).

Prop (BPJ). Many, but not all, exchangeable feature allocations have an exchangeable feature probability function (EF).

\[
\mathbb{P}(\text{vertex } = 1, 2, \ldots, K) = p(N; S_{N,1}, \ldots, S_{N,K})
\]

Degree of Kth vertex

[Petman 1995; Broderick, Pitman, Jordan 2013; Campbell, Cai, Broderick 2016]
Exchangeable probability functions

Thm (Pitman). Every exchangeable clustering has an exchangeable partition probability function (EPPF)

Prop (BPJ). Many, but not all, exchangeable feature allocations have an exchangeable feature probability function (EFPF)

\[
\mathbb{P}(\text{vertex } = 1 \ 2 \ \cdots \ K, \text{edge } = 1 \ 2 \ N) = p(S_{N,1}, \ldots, S_{N,K})
\]
Thm (Pitman). Every exchangeable clustering has an exchangeable **partition** probability function (EPFP).

Prop (BPJ). Many, but not all, exchangeable feature allocations have an exchangeable **feature** probability function (EFPF).

Definition (CCB). Exchangeable **vertex** probability function (EVPF).

\[
P( \vdots ) = p(S_{N,1}, \ldots , S_{N,K})
\]
Exchangeable probability functions

Q: Does every edge-exchangeable graph have an EVPF?
Exchangeable probability functions

Q: Does every edge-exchangeable graph have an EVPF?
A: No. Counterexample:

[Broderick, Jordan, Pitman 2013]
Q: Does every edge-exchangeable graph have an EVPF?

A: No. Counterexample:

```
edge = 1
  2
...```

[Broderick, Jordan, Pitman 2013]
Exchangeable probability functions

Q: Does every edge-exchangeable graph have an EVPF?
A: No. Counterexample:

edge = 1
2...
N

[Broderick, Jordan, Pitman 2013]
Q: Does every edge-exchangeable graph have an EVPF?

A: No. Counterexample:

[Diagram of a graph with labeled edges and nodes]

[Broderick, Jordan, Pitman 2013]
Q: Does every edge-exchangeable graph have an EVPF?

A: No. Counterexample:

[Diagram of a graph with labeled edges and nodes]

[Broderick, Jordan, Pitman 2013]
Q: Does every edge-exchangeable graph have an EVPF?

A: No. Counterexample:
Q: Does every edge-exchangeable graph have an EVPF?

A: No. Counterexample:

---

[Broderick, Jordan, Pitman 2013]
Q: Does every edge-exchangeable graph have an EVPF?
A: No. Counterexample:
Exchangeable probability functions

Q: Does every edge-exchangeable graph have an EVPF?
A: No. Counterexample:
Q: Does every edge-exchangeable graph have an EVPF?
A: No. Counterexample:

[Diagram of a graph with labeled edges and vertices]

[Broderick, Jordan, Pitman 2013]
Exchangeable probability functions

Q: Does every edge-exchangeable graph have an EVPF?

A: No. Counterexample:

[Broderick, Jordan, Pitman 2013]
Q: Does every edge-exchangeable graph have an EVPF?

A: No. Counterexample:

\[
\mathbb{P}(\text{row }= \begin{array}{ccc} \text{blue} & \text{green} & \text{white} \end{array}) = p_1 \\
\mathbb{P}(\text{row }= \begin{array}{ccc} \text{white} & \text{red} & \text{yellow} \end{array}) = p_2 \\
\mathbb{P}(\text{row }= \begin{array}{ccc} \text{green} & \text{red} & \text{white} \end{array}) = p_3 \\
\mathbb{P}(\text{row }= \begin{array}{ccc} \text{blue} & \text{white} & \text{yellow} \end{array}) = p_4
\]
Q: Does every edge-exchangeable graph have an EVPF?

A: No. Counterexample:

\[
\begin{align*}
P(\text{row } = 1) &= p_1 \\
P(\text{row } = 2) &= p_2 \\
P(\text{row } = 3) &= p_3 \\
P(\text{row } = 4) &= p_4
\end{align*}
\]
Exchangeable probability functions

Q: Does every edge-exchangeable graph have an EVPF?  
A: No. Counterexample:

\[
\begin{align*}
\Pr(\text{row } = \begin{array}{c}
\text{\blue{\ 	ext{\green{}}}}
\end{array}) &= p_1 \\
\Pr(\text{row } = \begin{array}{c}
\text{\green{\ 	ext{\red{}}}}
\end{array}) &= p_2 \\
\Pr(\text{row } = \begin{array}{c}
\text{\red{\ 	ext{\green{}}}}
\end{array}) &= p_3 \\
\Pr(\text{row } = \begin{array}{c}
\text{\blue{\ 	ext{\red{}}}}
\end{array}) &= p_4
\end{align*}
\]
Exchangeable probability functions

Q: Does every edge-exchangeable graph have an EVPF?

A: No. Counterexample:

\[
\begin{align*}
\mathbb{P}(\text{row } = \begin{array}{c}
\text{blue} \\
\text{green} \\
\text{white}
\end{array}) &= p_1 \\
\mathbb{P}(\text{row } = \begin{array}{c}
\text{white} \\
\text{red} \\
\text{yellow}
\end{array}) &= p_2 \\
\mathbb{P}(\text{row } = \begin{array}{c}
\text{white} \\
\text{green} \\
\text{red}
\end{array}) &= p_3 \\
\mathbb{P}(\text{row } = \begin{array}{c}
\text{blue} \\
\text{white} \\
\text{yellow}
\end{array}) &= p_4
\end{align*}
\]

\[
\begin{align*}
\mathbb{P}(\begin{array}{c}
\text{blue} \\
\text{green}
\end{array}) &= p_1 p_2 \\
\mathbb{P}(\begin{array}{c}
\text{yellow} \\
\text{red}
\end{array}) &= p_3 p_4
\end{align*}
\]

[Broderick, Jordan, Pitman 2013]
Exchangeable probability functions

Q: Does every edge-exchangeable graph have an EVPF?

A: No. Counterexample:

\[ P(\text{row } = \begin{array}{ccc} \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \end{array} ) = p_1 \]
\[ P(\text{row } = \begin{array}{ccc} \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \end{array} ) = p_2 \]
\[ P(\text{row } = \begin{array}{ccc} \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \end{array} ) = p_3 \]
\[ P(\text{row } = \begin{array}{ccc} \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \end{array} ) = p_4 \]

\[ P(\begin{array}{c} \text{blue} \\ \text{green} \end{array}) \neq P(\begin{array}{c} \text{yellow} \\ \text{red} \end{array}) \]

\[ p_1 p_2 \neq p_3 p_4 \]

[Broderick, Jordan, Pitman 2013]
Exchangeable probability functions

Q: Does every edge-exchangeable graph have an EVPF?
A: No. Counterexample:

\[ P(\text{row } = \begin{array}{c|c|c|c} & & \end{array}) = p_1 \]
\[ P(\text{row } = \begin{array}{c|c|c|c} \end{array}) = p_2 \]
\[ P(\text{row } = \begin{array}{c|c|c|c} & & \end{array}) = p_3 \]
\[ P(\text{row } = \begin{array}{c|c|c|c} \end{array}) = p_4 \]

\[ P(\begin{array}{c|c} \end{array}) \neq P(\begin{array}{c|c} \end{array}) \]

\[ p_1p_2 \neq p_3p_4 \]

Cor. Not every exchangeable feature allocation has an EFPF.

[Broderick, Jordan, Pitman 2013]
What we know so far

- Goal 1: characterization theorem for edge-exchangeable graphs
- Goal 2: sparsity theorem for edge-exchangeable graphs
What we know so far

- Goal 1: characterization theorem for edge-exchangeable graphs
- Goal 2: sparsity theorem for edge-exchangeable graphs
Clustering
Clustering
Clustering
Clustering

[Kingman 1978]
Clustering

[Kingman 1978]
Clustering

[Kingman 1978]
Clustering

[Image of a diagram with numbers 1 to 4 and a grid with colors blue, yellow, and orange]
Clustering

[Kingman 1978]
Clustering

[Kingman 1978]
Clustering

[Kingman 1978]
Clustering

Thm (Kingman). A clustering is exchangeable iff it has a Kingman paintbox representation

[Kingman 1978]
Thm (Kingman). A clustering is exchangeable iff it has a Kingman paintbox representation.
Clustering

Thm (Kingman). A clustering is exchangeable iff it has a Kingman paintbox representation.
Feature allocation

Cat feature
Dog feature
Mouse feature
Lizard feature
Sheep feature
Horse feature
Feature allocation

[Broderick, Pitman, Jordan 2013]
Feature allocation

[Broderick, Pitman, Jordan 2013]
Feature allocation

Cat feature
Dog feature
Mouse feature
Lizard feature
Sheep feature
Horse feature

[Broderick, Pitman, Jordan 2013]
Feature allocation

[Broderick, Pitman, Jordan 2013]
Feature allocation

[Broderick, Pitman, Jordan 2013]
Feature allocation

[Cat feature, Dog feature, Mouse feature, Lizard feature, Sheep feature, Horse feature]
Feature allocation

[Broderick, Pitman, Jordan 2013]
Feature allocation

Thm (BPJ). A feature allocation is exchangeable iff it has a feature paintbox representation.
Thm (BPJ). A feature allocation is exchangeable iff it has a feature paintbox representation.

Feature allocation

Cat feature
Dog feature
Mouse feature
Lizard feature
Sheep feature
Horse feature

[B, Pitman, Jordan 2013; Campbell, Cai, B 2016]
Cat feature
Dog feature
Mouse feature
Lizard feature
Sheep feature
Horse feature
Edge-exchangeable graph

Cat node
Dog node
Mouse node
Lizard node
Sheep node
Horse node

[Cai, B 2015a,b; Cai, Campbell, B, 2016; Campbell, Cai, B, 2016]
Edge-exchangeable graph

[Cai, B 2015a,b; Cai, Campbell, B, 2016; Campbell, Cai, B, 2016]
Edge-exchangeable graph

[Cai, B 2015a,b; Cai, Campbell, B, 2016; Campbell, Cai, B, 2016]
Edge-exchangeable graph

Cat node
Dog node
Mouse node
Lizard node
Sheep node
Horse node

[Cai, B 2015a,b; Cai, Campbell, B, 2016; Campbell, Cai, B, 2016]
Edge-exchangeable graph

1. Cat node
2. Dog node
3. Mouse node
4. Lizard node
5. Sheep node
6. Horse node

[Cai, B 2015a,b; Cai, Campbell, B, 2016; Campbell, Cai, B, 2016]
Edge-exchangeable graph

[Diagram showing a grid with various nodes labeled as Cat, Dog, Mouse, Lizard, Sheep, and Horse, along with a network graph.]
Edge-exchangeable graph

[Cai, B 2015a,b; Cai, Campbell, B, 2016; Campbell, Cai, B, 2016]
Edge-exchangeable graph

[Cat node, Dog node, Mouse node, Lizard node, Sheep node, Horse node]

[Cai, B 2015a,b; Cai, Campbell, B, 2016; Campbell, Cai, B, 2016]
Cor (CCB). A graph sequence is edge-exchangeable iff it has a graph paintbox.
Cor (CCB). A graph sequence is edge-exchangeable iff it has a graph paintbox Extends to hypergraphs

Any edge-exchangeable graph sequence has a graph paintbox.

Extends to hypergraphs

Cat node
Dog node
Mouse node
Lizard node
Sheep node
Horse node

[Cai, B 2015a,b; Cai, Campbell, B, 2016; Campbell, Cai, B, 2016; Crane, Dempsey 2016b]
Cor (CCB). A graph sequence is edge-exchangeable iff it has a graph paintbox. Any edge-exchangeable graph sequence has a graph paintbox Extends to hypergraphs + self-edges, skips.

Mouse node
Lizard node
Sheep node
Horse node

[Cai, B 2015a,b; Cai, Campbell, B, 2016; Campbell, Cai, B, 2016; Crane, Dempsey 2016b]
What we know so far

- Goal 1: characterization theorem for edge-exchangeable graphs
- Goal 2: sparsity theorem for edge-exchangeable graphs
What we know so far

- Goal 1: characterization theorem for edge-exchangeable graphs
- Goal 2: sparsity theorem for edge-exchangeable graphs
What we know so far

- Thm 1: characterization theorem for edge-exchangeable graphs
- Goal 2: sparsity theorem for edge-exchangeable graphs
How to prove sparsity?
How to prove sparsity?

- Need # nodes to go to infinity
How to prove sparsity?

• Need # nodes to go to infinity
• Need countable $\infty$ of latent nodes
How to prove sparsity?

- Need # nodes to go to infinity
- Need countable $\infty$ of latent nodes
How to prove sparsity?

- Need # nodes to go to infinity
- Need countable $\infty$ of latent nodes
- Graph frequency model/vertex popularity model
How to prove sparsity?

- Need # nodes to go to infinity
- Need countable $\infty$ of latent nodes
- Graph frequency model/vertex popularity model
- Draw a rate $w_i$ for each vertex $i$
How to prove sparsity?

- Need # nodes to go to infinity
- Need countable $\infty$ of latent nodes
- Graph frequency model/vertex popularity model
- Draw a rate $w_i$ for each vertex $i$
How to prove sparsity?

- Need # nodes to go to infinity
- Need countable $\infty$ of latent nodes
- Graph frequency model/vertex popularity model
- Draw a rate $w_i$ for each vertex $i$
How to prove sparsity?

• Need # nodes to go to infinity
  • Need countable $\infty$ of latent nodes
• Graph frequency model/vertex popularity model
  • Draw a rate $w_i$ for each vertex $i$
  • Draw edge $\{i,j\}$ with probability proportional to $w_iw_j$
How to prove sparsity?

- Need # nodes to go to infinity
- Need countable $\infty$ of latent nodes
- Graph frequency model/vertex popularity model
  - Draw a rate $w_i$ for each vertex $i$
  - Draw edge $\{i,j\}$ with probability proportional to $w_i w_j$
How to prove sparsity?

- Need # nodes to go to infinity
- Need countable $\infty$ of latent nodes
- Graph frequency model/vertex popularity model
  - Draw a rate $w_i$ for each vertex $i$
  - Draw edge $\{i,j\}$ with probability proportional to $w_iw_j$

[Caron, Fox 2014; Cai, Campbell, B 2016; Crane, Dempsey 2016a; Palla, Caron, Teh 2016; Herlau, Schmidt 2016]
How to prove sparsity?

- Need # nodes to go to infinity
- Need countable $\infty$ of latent nodes
- Graph frequency model/vertex popularity model
- Draw a rate $w_i$ for each vertex $i$
- Draw edge $\{i,j\}$ with probability proportional to $w_i w_j$

[Caron, Fox 2014; Cai, Campbell, B 2016; Crane, Dempsey 2016a; Palla, Caron, Teh 2016; Herlau, Schmidt 2016]
How to prove sparsity?

- Need # nodes to go to infinity
- Need countable $\infty$ of latent nodes
- Graph frequency model/vertex popularity model
  - Draw a rate $w_i$ for each vertex $i$
  - Draw edge $\{i,j\}$ with probability proportional to $w_i w_j$

[Caron, Fox 2014; Cai, Campbell, B 2016; Crane, Dempsey 2016a; Palla, Caron, Teh 2016; Herlau, Schmidt 2016]
How to prove sparsity?

- Need # nodes to go to infinity
- Need countable $\infty$ of latent nodes
- Graph frequency model/vertex popularity model
  - Draw a rate $w_i$ for each vertex $i$
  - Draw edge $\{i,j\}$ with probability proportional to $w_i w_j$

[Caron, Fox 2014; Cai, Campbell, B 2016; Crane, Dempsey 2016a; Palla, Caron, Teh 2016; Herlau, Schmidt 2016]
How to prove sparsity?

- Need # nodes to go to infinity
- Need countable $\infty$ of latent nodes
- Graph frequency model/vertex popularity model
- Draw a rate $w_i$ for each vertex $i$
- Draw edge $\{i,j\}$ with probability proportional to $w_iw_j$

[Caron, Fox 2014; Cai, Campbell, B 2016; Crane, Dempsey 2016a; Palla, Caron, Teh 2016; Herlau, Schmidt 2016]
How to prove sparsity?

• Need # nodes to go to infinity
  • Need countable $\infty$ of latent nodes
• Graph frequency model/vertex popularity model
  • Draw a rate $w_i$ for each vertex $i$
  • Draw edge $\{i,j\}$ with probability proportional to $w_i w_j$

[Cai, Campbell, Broderick 2016]
How to prove sparsity?

- Need # nodes to go to infinity
- Need countable $\infty$ of latent nodes
- Graph frequency model/vertex popularity model
  - Draw a rate $w_i$ for each vertex $i$
  - Draw edge $\{i,j\}$ with probability proportional to $w_i w_j$
How to prove sparsity?

• Need # nodes to go to infinity
  • Need countable $\infty$ of latent nodes
• Graph frequency model/vertex popularity model
  • Draw a rate $w_i$ for each vertex $i$
  • Draw edge $\{i,j\}$ with probability proportional to $w_i w_j$

Thm (CCB).
• Suppose $w_i \sim \text{PoissonPointProcess}(\nu)$, $\nu$ regularly varying
How to prove sparsity?

- Need \# nodes to go to infinity
- Need countable \( \infty \) of latent nodes
- Graph frequency model/vertex popularity model
- Draw a rate \( w_i \) for each vertex \( i \)
- Draw edge \( \{i,j\} \) with probability proportional to \( w_i w_j \)

**Thm (CCB).**

- Suppose \( w_i \sim \text{PoissonPointProcess}(\nu) \), \( \nu \) regularly varying:

\[
\int_x^1 \nu(dw) \sim x^{-\alpha} l(x^{-1}), \ x \to 0 \quad \forall c > 0, \ \lim_{x \to \infty} \frac{l(cx)}{l(x)} = 1
\]

[23]

[Cai, Campbell, Broderick 2016]
How to prove sparsity?

- Need # nodes to go to infinity
- Need countable $\infty$ of latent nodes
- Graph frequency model/vertex popularity model
- Draw a rate $w_i$ for each vertex $i$
- Draw edge \{i, j\} with probability proportional to $w_i w_j$

**Thm (CCB).**

- Suppose $w_i \sim \text{PoissonPointProcess}(\nu)$, $\nu$ regularly varying:
  \[ \int_{x}^{1} \nu(dw) \sim x^{-\alpha} l(x^{-1}), \ x \to 0 \quad \forall c > 0, \lim_{x \to \infty} \frac{l(cx)}{l(x)} = 1 \]
- Then $|V_n| \xrightarrow{a.s.} \Theta(n^\alpha l(n))$, $|E_n| \xrightarrow{a.s.} \Theta(n)$
How to prove sparsity?

- Need # nodes to go to infinity
- Need countable \( \infty \) of latent nodes
- Graph frequency model/vertex popularity model
- Draw a rate \( w_i \) for each vertex \( i \)
- Draw edge \( \{i,j\} \) with probability proportional to \( w_i w_j \)

Thm (CCB).

- Suppose \( w_i \sim \text{PoissonPointProcess}(\nu) \), \( \nu \) regularly varying:
  \[
  \int_x^1 \nu(dw) \sim x^{-\alpha} l(x^{-1}), \quad x \to 0 \quad \forall c > 0, \quad \lim_{x \to \infty} \frac{l(cx)}{l(x)} = 1
  \]
- Then \( |V_n| \overset{a.s.}{=} \Theta(n^\alpha l(n)), |E_n| \overset{a.s.}{=} \Theta(n) \)
- & for binary edges: \( |\overline{E}_n| \overset{a.s.}{=} O \left(l(n^{1/2}), \min \left(n^{\frac{1+\alpha}{2}}, l(n)n^{\frac{3\alpha}{2}}\right)\right) \)
How to prove sparsity?

- Need # nodes to go to infinity
- Need countable $\infty$ of latent nodes
- Graph frequency model/vertex popularity model
- Draw a rate $w_i$ for each vertex $i$
- Draw edge $\{i,j\}$ with probability proportional to $w_i w_j$

**Thm (CCB).**

- Suppose $w_i \sim \text{PoissonPointProcess}(\nu)$, $\nu$ regularly varying:
  \[
  \int_{x}^{1} \nu(dw) \sim x^{-\alpha} l(x^{-1}), \ x \to 0 \quad \forall c > 0, \ \lim_{x \to \infty} \frac{l(cx)}{l(x)} = 1
  \]
- Then $|V_n| \overset{a.s.}{=} \Theta(n^{\alpha} l(n)), |E_n| \overset{a.s.}{=} \Theta(n)$
- & for binary edges: $|\overline{E}_n| \overset{a.s.}{=} O\left(l(n^{1/2}), \min\left(n^{\frac{1+\alpha}{2}}, l(n)n^{\frac{3\alpha}{2}}\right)\right)$

**Cor (CCB).** A wide class of edge-exchangeable graph models yields sparse graph sequences.

[23] [Cai, Campbell, Broderick 2016]
Example graph frequency model
Empirical: can achieve range of (sparse & dense) power laws

[alpha = 0.6]

[Image of a log-log plot showing the relationship between the number of edges and the number of vertices with a power law distribution indicated by a straight line.

[Cai, Campbell, Broderick 2016]
Example graph frequency model
Empirical: can achieve range of (sparse & dense) power laws

[Cai, Campbell, Broderick 2016]
Example graph frequency model
Empirical: can achieve range of (sparse & dense) power laws
  • Multigraph

[Cai, Campbell, Broderick 2016]
Example graph frequency model
Empirical: can achieve range of (sparse & dense) power laws

- Multigraph
- Binary graph

[Cai, Campbell, Broderick 2016]
What we know so far

- Thm 1: characterization theorem for edge-exchangeable graphs
- Goal 2: sparsity theorem for edge-exchangeable graphs
What we know so far

- Thm 1: characterization theorem for edge-exchangeable graphs
- Thm 2: sparsity exists in edge-exchangeable graphs
What we know so far

- Thm 1: characterization theorem for edge-exchangeable graphs
- Thm 2: sparsity exists in edge-exchangeable graphs

- edge-exchangeable ↔ graph paintbox

- EVertexPF

-_sparse?
What we know so far

- Thm 1: characterization theorem for edge-exchangeable graphs
- Thm 2: sparsity exists in edge-exchangeable graphs
What we know so far

- Thm 1: characterization theorem for edge-exchangeable graphs
- Thm 2: sparsity exists in edge-exchangeable graphs
What we know so far

- Thm 1: characterization theorem for edge-exchangeable graphs
- Thm 2: sparsity exists in edge-exchangeable graphs
What we know so far

- Thm 1: characterization theorem for edge-exchangeable graphs
- Thm 2: sparsity exists in edge-exchangeable graphs

edge-exchangeable ⇔ graph paintbox

EVertexPF

sparse
What we know so far

- Thm 1: characterization theorem for edge-exchangeable graphs
- Thm 2: sparsity exists in edge-exchangeable graphs
What we know so far

- Thm 1: characterization theorem for edge-exchangeable graphs
- Thm 2: sparsity exists in edge-exchangeable graphs

[Campbell, Cai, Broderick 2016]
What we know so far

- Thm 1: characterization theorem for edge-exchangeable graphs
- Thm 2: sparsity exists in edge-exchangeable graphs

Proof: Use our result: EFPF $\leftrightarrow$ feature frequency model

[Campbell, Cai, Broderick 2016; Broderick, Jordan, Pitman 2013]
What we know so far

- Thm 1: characterization theorem for edge-exchangeable graphs
- Thm 2: sparsity exists in edge-exchangeable graphs
What we know so far

- Thm 1: characterization theorem for edge-exchangeable graphs
- Thm 2: sparsity exists in edge-exchangeable graphs
What we know so far

- Thm 1: characterization theorem for edge-exchangeable graphs
- Thm 2: sparsity exists in edge-exchangeable graphs
What we know so far

- Thm 1: characterization theorem for edge-exchangeable graphs
- Thm 2: sparsity exists in edge-exchangeable graphs
- Also: Characterization of “dust” (new for features, traits, and graphs)
What we know so far

- Thm 1: characterization theorem for edge-exchangeable graphs
- Thm 2: sparsity exists in edge-exchangeable graphs
- Also: Characterization of “dust” (new for features, traits, and graphs)

What’s next

[Campbell, Cai, Broderick 2016]
What we know so far

- Thm 1: characterization theorem for edge-exchangeable graphs
- Thm 2: sparsity exists in edge-exchangeable graphs
- Also: Characterization of “dust” (new for features, traits, and graphs)

What’s next

- Characterize all sparse, edge-exchangeable graphs

[Campbell, Cai, Broderick 2016]
What we know so far

- Thm 1: characterization theorem for edge-exchangeable graphs
- Thm 2: sparsity exists in edge-exchangeable graphs
- Also: Characterization of “dust” (new for features, traits, and graphs)

What’s next

- Characterize all sparse, edge-exchangeable graphs
- Characterize the different types of power laws (edges, triangles, degree distributions, etc.)
What we know so far

- Thm 1: characterization theorem for edge-exchangeable graphs
- Thm 2: sparsity exists in edge-exchangeable graphs
- Also: Characterization of “dust” (new for features, traits, and graphs)

What’s next

- Characterize all sparse, edge-exchangeable graphs
- Characterize the different types of power laws (edges, triangles, degree distributions, etc.)
- Models and inference; truncation approximations

[Campbell, Cai, Broderick 2016; Broderick, Jordan, Pitman 2012; Campbell*, Huggins*, How, Broderick 2016]
Nonparametric Bayes

- Bayesian
  \[ P(\text{parameters}|\text{data}) \propto P(\text{data}|\text{parameters})P(\text{parameters}) \]
- Not parametric (i.e. not finite parameter, unbounded/growing/infinite number of parameters)
References (page 1 of 2)


  - *NIPS 2015 Workshop on Networks in the Social & Information Sciences.*
  - *NIPS 2015 Workshop, Bayesian Nonparametrics: The Next Generation.*

  - *NIPS 2016 Workshop on Adaptive & Scalable Nonparametric Methods in ML.*
  - *NIPS 2016 Workshop on Practical Bayesian Nonparametrics.*


* Shared first authorship
References (page 2 of 2)


