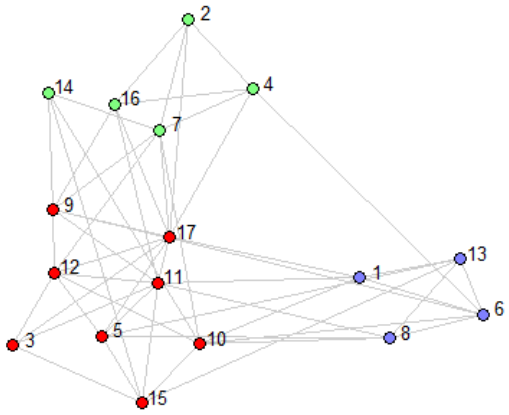


Generative models for social network data

Kevin S. Xu (University of Toledo)

James R. Foulds (University of California-San Diego)

SBP-BRiMS 2016 Tutorial

About Us

Kevin S. Xu

- Assistant professor at University of Toledo
- 3 years research experience in industry
- Research interests:
  - Machine learning
  - Network science
  - Physiological data analysis

James R. Foulds

- Postdoctoral scholar at UCSD
- Research interests:
  - Bayesian modeling
  - Social networks
  - Text
  - Latent variable models

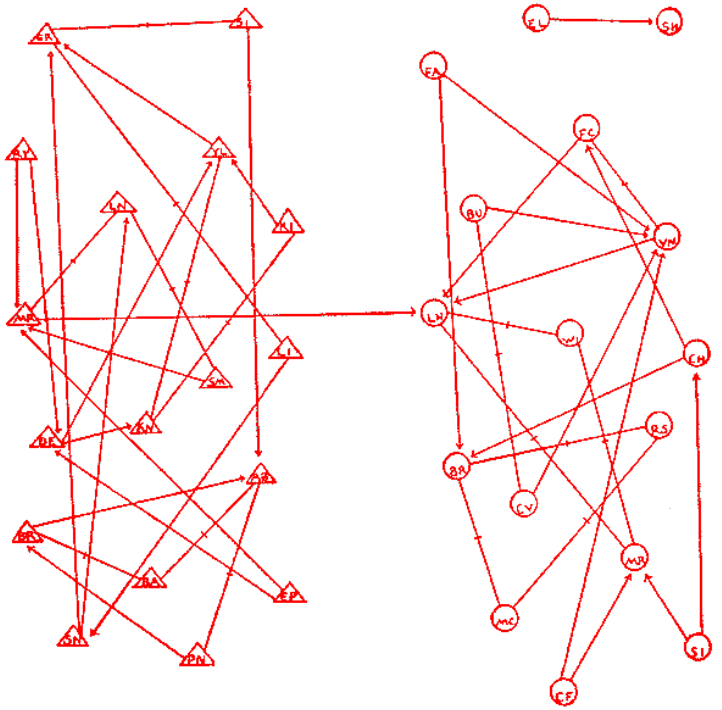
Outline

- Mathematical representations of social networks and generative models
  - Introduction to generative approach
  - Connections to sociological principles
- Fitting generative social network models to data
  - Example application scenarios
  - Model selection and evaluation

Social networks today

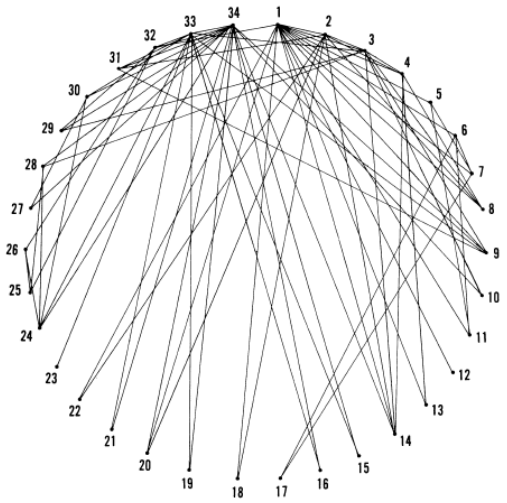
Social network analysis: an interdisciplinary endeavor

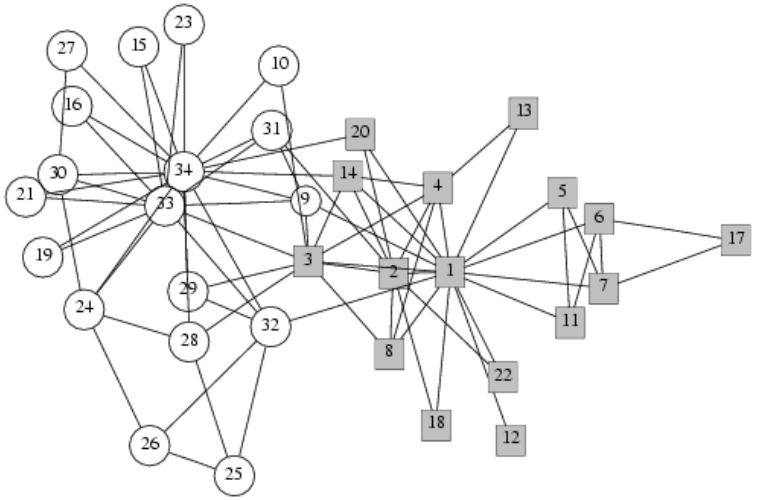
- Sociologists have been studying social networks for decades!
- First known empirical study of social networks: Jacob Moreno in 1930s
  - Moreno called them sociograms
- Recent interest in social network analysis (SNA) from physics, EECS, statistics, and many other disciplines



Social networks as graphs

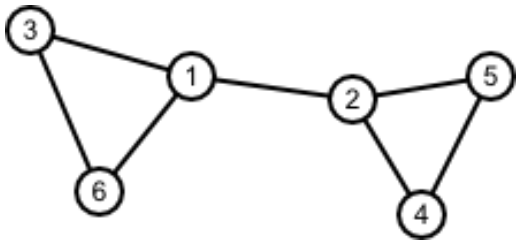
- A social network can be represented by a graph $G = (V, E)$
  - V: vertices, nodes, or actors typically representing people
  - E: edges, links, or ties denoting relationships between nodes
  - Directed graphs used to represent asymmetric relationships
- Graphs have no natural representation in a geometric space
  - Two identical graphs drawn differently
  - Moral: visualization provides very limited analysis ability
  - How do we model and analyze social network data?



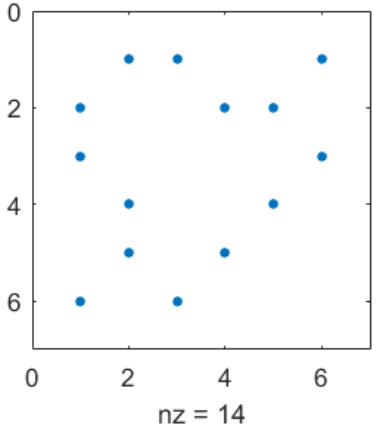


Matrix representation of social networks

- Represent graph by $n \times n$ adjacency matrix or sociomatrix $\mathbf{Y}$
  - $y_{ij} = 1$ if there is an edge between nodes i and j
  - $y_{ij} = 0$ otherwise



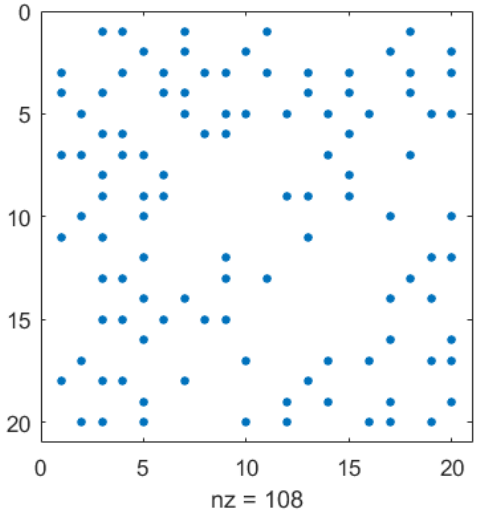
$$\mathbf{Y} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

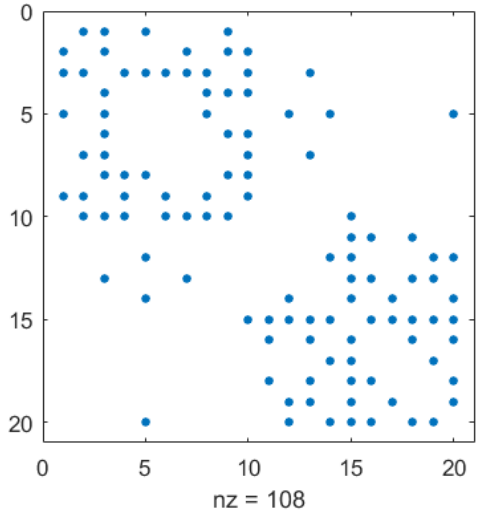


- Easily extended to directed and weighted graphs

Exchangeability of nodes

- Nodes are typically assumed to be (statistically) exchangeable by symmetry
- Row and column permutations to adjacency matrix do not change graph
  - Needs to be incorporated into social network models





Sociological principles related to edge formation

- Homophily or assortative mixing
  - Tendency for individuals to bond with similar others
  - Assortative mixing by age, gender, social class, organizational role, node degree, etc.
  - Results in transitivity (triangles) in social networks
    - "My friend of my friend is my friend"
- Equivalence of nodes
  - Two nodes are structurally equivalent if their relations to all other nodes are identical
    - Approximate equivalence recorded by similarity measure
  - Two nodes are regularly equivalent if their neighbors are similar (not necessarily common neighbors)

Brief history of social network models

- Early 1900s – sociology and social psychology precursors to SNA (Georg Simmel)
- 1930s – Graphical depictions of social networks: sociograms (Jacob Moreno)
- 1960s – Small world / 6-degrees of separation experiment (Stanley Milgram)
- 1970s – Mathematical models of social networks (Erdos-Renyi-Gilbert)
- 1980s – Statistical models (Holland and Leinhardt, Frank and Strauss)
- 1990s – Statistical physicists weigh in: preferential attachment, small world models, power-law degree distributions (Barabasi et al.)
- 2000s – Today – Machine learning approaches, latent variable models

Generative models for social networks

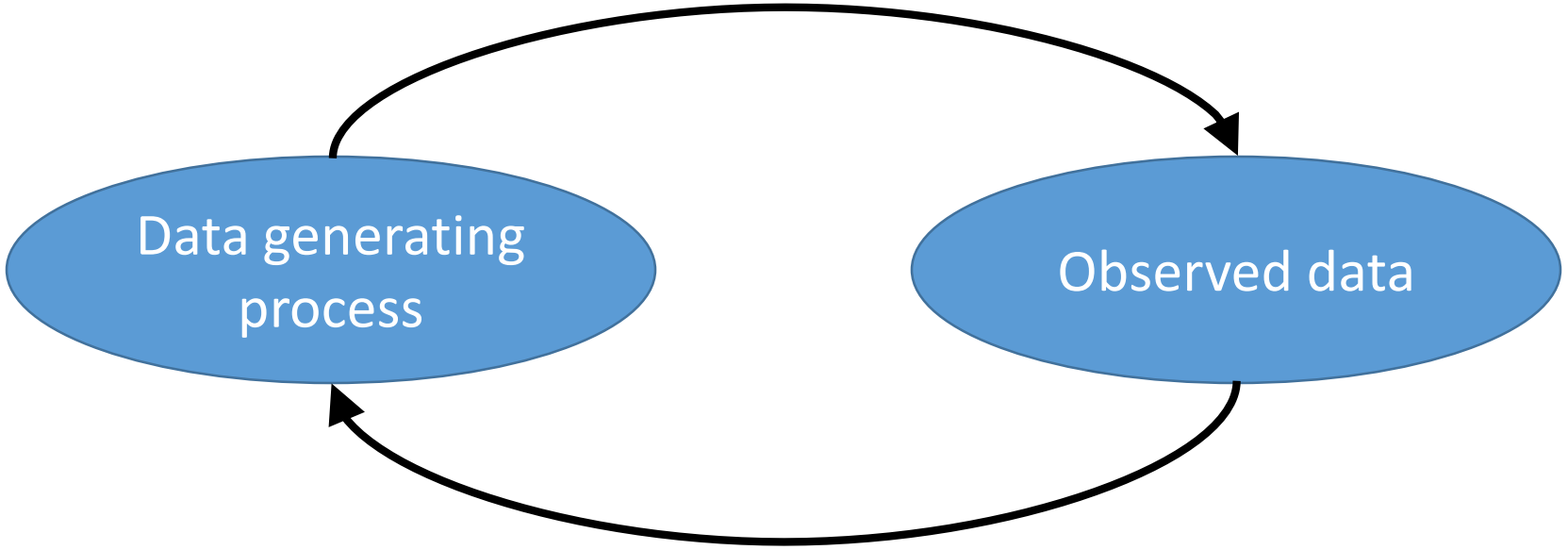
- A **generative model** is one that can simulate new networks

- Two distinct schools of thought:
  - Probability models (non-statistical)
    - Typically simple, 1-2 parameters, not learned from data
    - Can be studied analytically
  - Statistical models
    - More parameters, latent variables
    - Learned from data via statistical techniques

Probability and Inference

Mathematics/physics: Erdős-Rényi, preferential attachment,…

Probability



Inference

Statistics/machine learning: ERGMs, latent variable models…

Figure based on one by Larry Wasserman, "All of Statistics"

Probability models for networks (non-statistical)

Erdős-Rényi model

- There are two variants of this model
- The $G(N, E)$ model is a probability distribution over graphs with a **fixed number of edges**.
- It posits that all graphs on N nodes with E edges are equally likely

Erdős-Rényi model

- The $G(N, p)$ model posits that each edge is “on” with probability *p*

- Probability of adjacency matrix

$$Pr(\mathbf{Y}|p) = \prod_{i<j} p^{Y_{ij}} (1-p)^{1-Y_{ij}}$$

Erdős-Rényi model

- Adjacency matrix likelihood:

$$Pr(\mathbf{Y}|p) = \prod_{i<j} p^{Y_{ij}}(1-p)^{1-Y_{ij}}$$

- Number of edges is binomial.

$$Pr(E|p) = \binom{\binom{N}{2}}{E} p^E (1-p)^{\binom{N}{2}-E}$$

- For large N, this is well approximated by a Poisson

$$E \approx\sim Poisson(\binom{N}{2}p)$$

Preferential attachment models

- The Erdős-Rényi model assumes nodes typically have about the same degree (# edges)

- Many real networks have a **degree distribution** following a power law (possibly controversial?)

$$P(k) \propto k^{-\lambda}$$

- **Preferential attachment** is a variant on the G(N,p) model to address this (Barabasi and Albert, 1999)

Preferential attachment models

- Initially, no edges, and N_0 nodes.

- For each remaining node n
  - Add n to the network
  - For i =1:m
    - Connect n to a random existing node with probability proportional to its degree (+ smoothing counts),

$$Pr(m) = \frac{k_m + k_0}{\sum_i (k_i + k_0)}$$

- A Polya urn process! Rich get richer.

Small world models (Watts and Strogatz)

- Start with nodes connected to K neighbors in a ring
- Randomly rewire each edge with probability β
- Has low average path length (**small world phenomenon**, "*6-degrees of separation*")

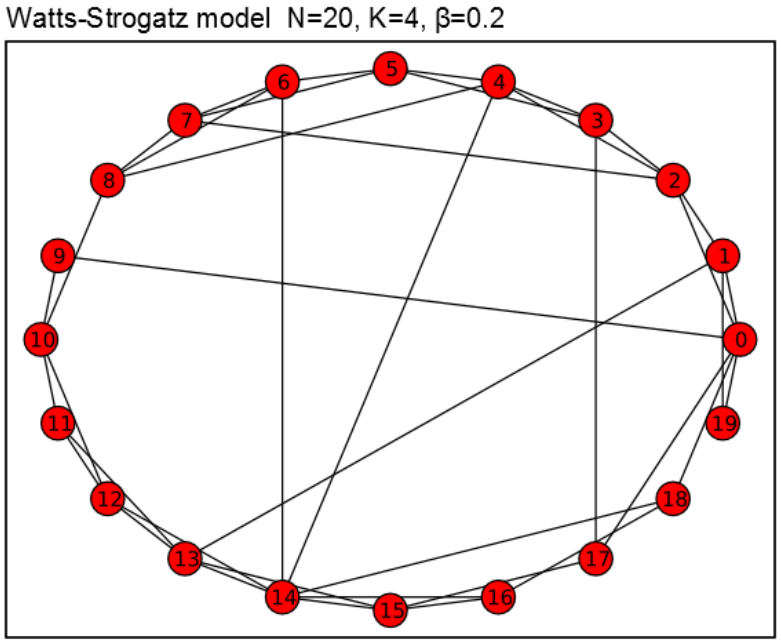


Figure due to Arpad Horvath, https://commons.wikimedia.org/wiki/File:Watts_strogatz.svg

Statistical network models

Exponential family random graphs (ERGMs)

$$Pr(Y = y|\theta) = \frac{1}{Z(\theta)} \exp\left(\theta^{\intercal} S(y, \mathbf{X})\right)$$

Arbitrary sufficient statistics

Covariates (gender, age, …)

E.g. “how many males are friends with females”

Exponential family random graphs (ERGMs)

- Pros:
  - Powerful, flexible representation
  - Can encode complex theories, and do substantive social science
  - Handles covariates
  - Mature software tools available,
    e.g. ergm package for statnet

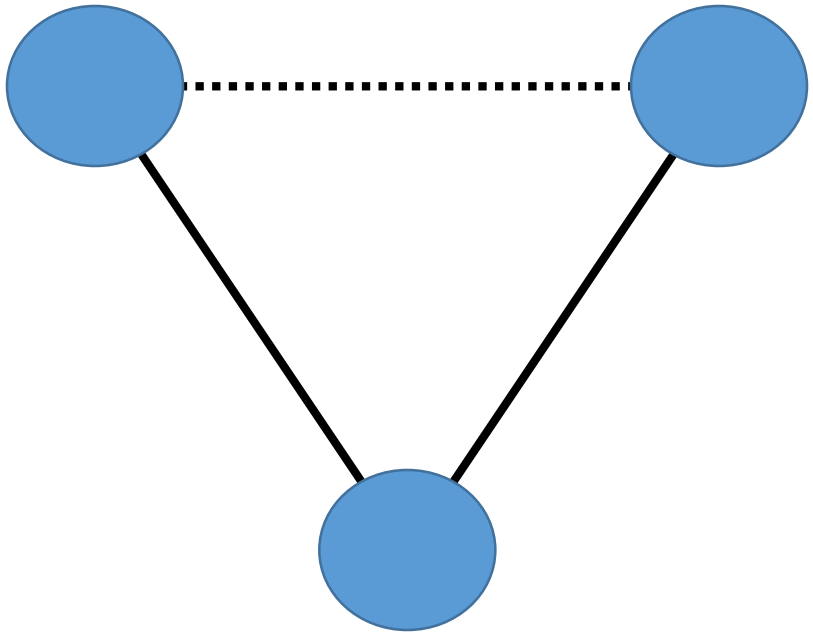
Exponential family random graphs (ERGMs)

- Cons:
  - Usual caveats of undirected models apply
    - Computationally intensive, especially learning
    - Inference may be intractable, due to partition function

Exponential family random graphs (ERGMs)

- Cons:
  - Usual caveats of undirected models apply
    - Computationally intensive, especially learning
    - Inference may be intractable, due to partition function
  - **Model degeneracy** can easily happen
    - "*a seemingly reasonable model can actually be such a bad mis-specification for an observed dataset as to render the observed data virtually impossible*"
      - **Goodreau (2007)**

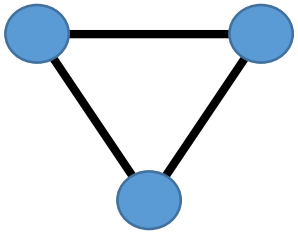
Triadic closure



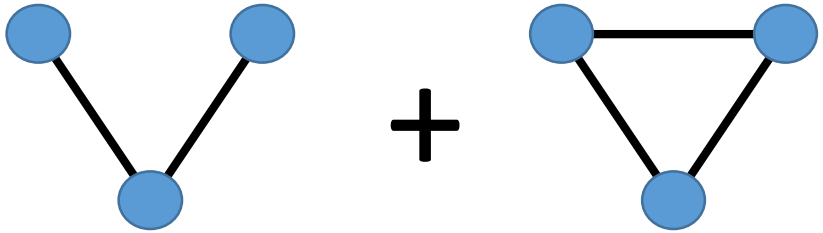
If two people have a friend in common, then there is an increased likelihood that they will become friends themselves at some point in the future.

Measuring triadic closure

- Mean clustering co-efficient:



$$C = \frac{\#\text{ triads with all three edges}}{\#\text{ triads with at least two edges}}$$



Simple ERGM for triadic closure leads to model degeneracy

$$Pr(Y = y|\theta) = \frac{1}{Z(\theta)} \exp\left(\theta^{\intercal} S(y, \mathbf{X})\right)$$

$$\theta = [\text{edge density}, \text{mean clustering coefficient}]^{\intercal}$$

Simple ERGM for triadic closure leads to model degeneracy

$$Pr(Y = y|\theta) = \frac{1}{Z(\theta)} \exp\left(\theta^{\intercal} S(y, \mathbf{X})\right)$$

$$\theta = [\text{edge density}, \text{mean clustering coefficient}]^{\intercal}$$

Depending on parameters, we could get:

- Graph is empty with probability close to 1
- Graph is full with probability close to 1
- Density, clustering distribution is bimodal, with little mass on desired density and triad closure

MLE may not exist!

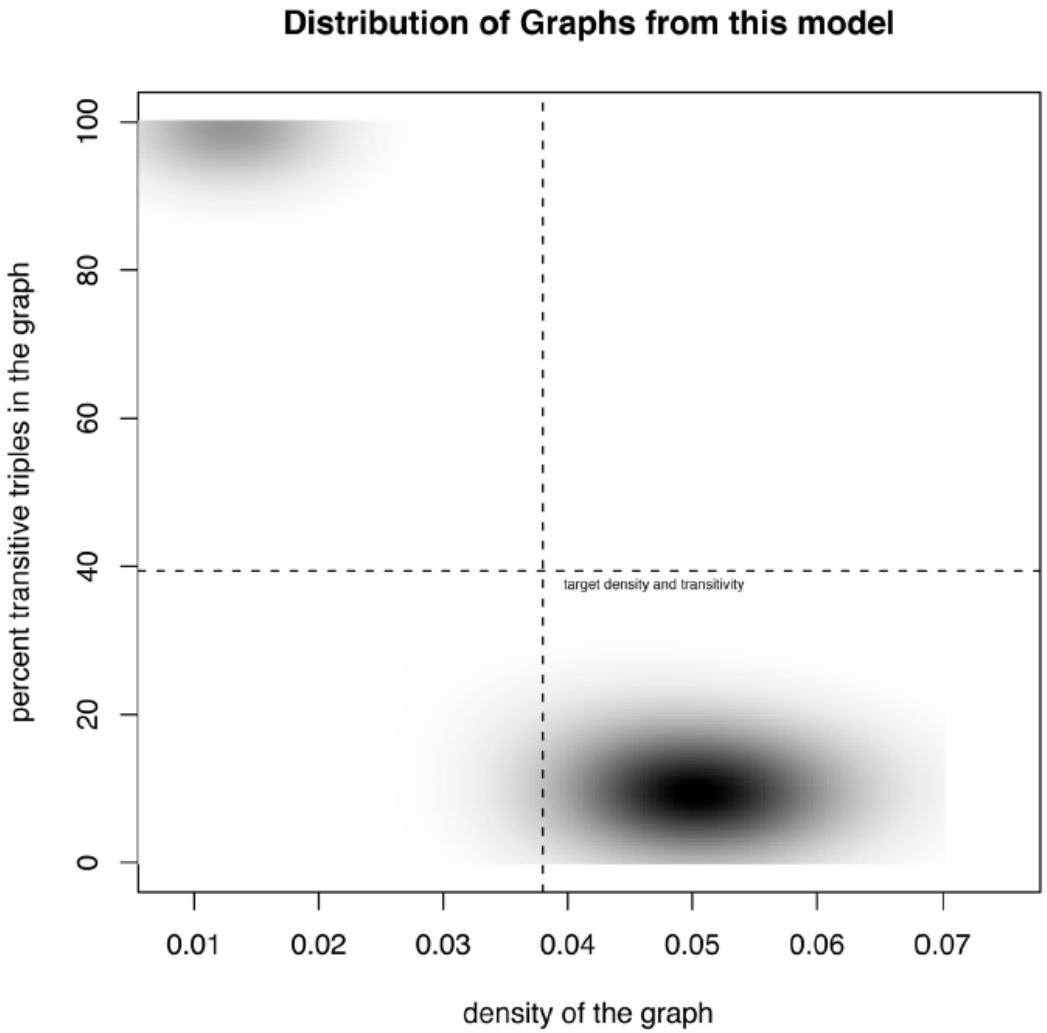
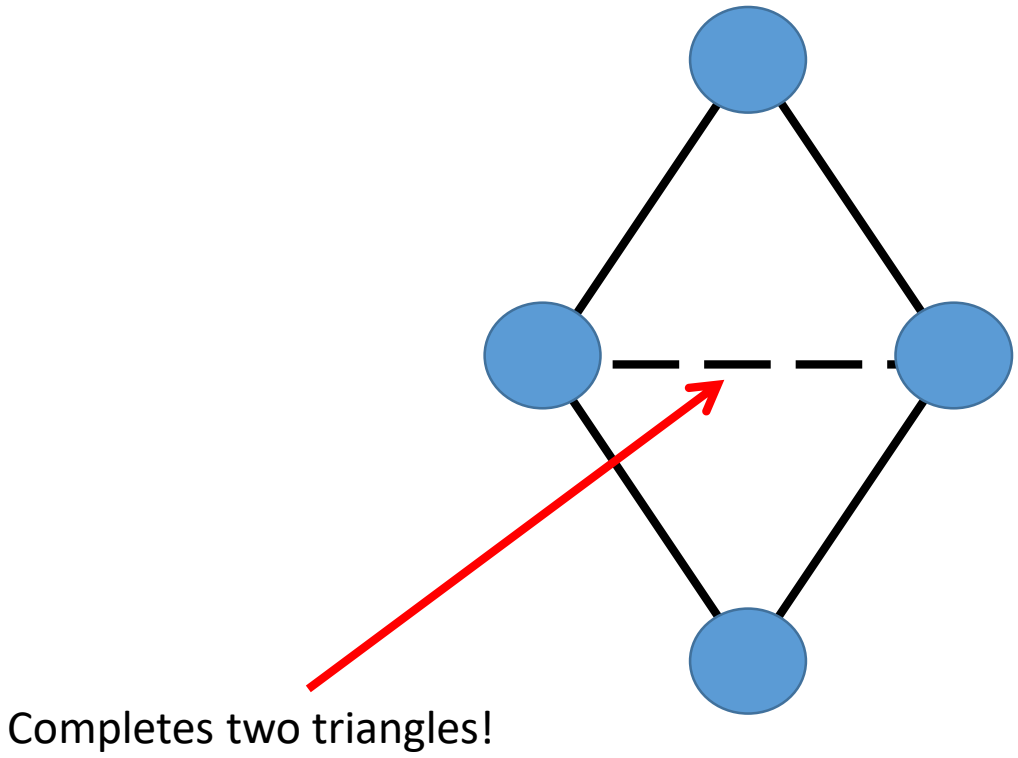


Figure 1.
Darker gray indicates higher probability density in this plot showing the true distribution of networks according to a particular two-statistic ERGM containing edge density and mean clustering coefficient. The population mean vector, specified by a particular choice of the model parameters, is shown at the intersection of the two dotted lines. The fact that there is very little probability mass near this mean is emblematic of degeneracy.

Handcock, M. S., Hunter, D. R., Butts, C. T., Goodreau, S. M., & Morris, M. (2008). statnet: Software tools for the representation, visualization, analysis and simulation of network data. Journal of statistical software, 24(1), 1548.

What is the problem?



Completes two triangles!

If an edge completes more triangles, it becomes overwhelming likely to exist.
This propagates to create more triangles …

Solution

- Change the model so that there are diminishing returns for completing more triangles
  - A different natural parameter for each possible number of triangles completed by one edge
  - Natural parameters $\eta(\theta)$ parameterized by a lower-dimensional θ, e.g. encoding geometrically decreasing weights (**curved exponential family**)

- **Moral of the story:** ERGMS are powerful, but require care and expertise to perform well

Latent variable models for social networks

- Model where observed variables are dependent on a set of unobserved or latent variables
  - Observed variables assumed to be conditionally independent given latent variables
- Why latent variable models?
  - Adjacency matrix $\mathbf{Y}$ is invariant to row and column permutations
  - Aldous-Hoover theorem implies existence of a latent variable model of form

$$y_{ij} = h(\theta, z_i, z_j, \epsilon_{ij})$$

for iid latent variables z_i and some function h

Latent variable models for social networks

- Latent variable models allow for heterogeneity of nodes in social networks
  - Each node (actor) has a latent variable $\mathbf{z}_i$
  - Probability of forming edge between two nodes is independent of all other node pairs given values of latent variables

$$p(\mathbf{Y}|\mathbf{Z},\theta) = \prod_{i \neq j} p\left(y_{ij}\middle|\mathbf{z}_i,\mathbf{z}_j,\theta\right)$$

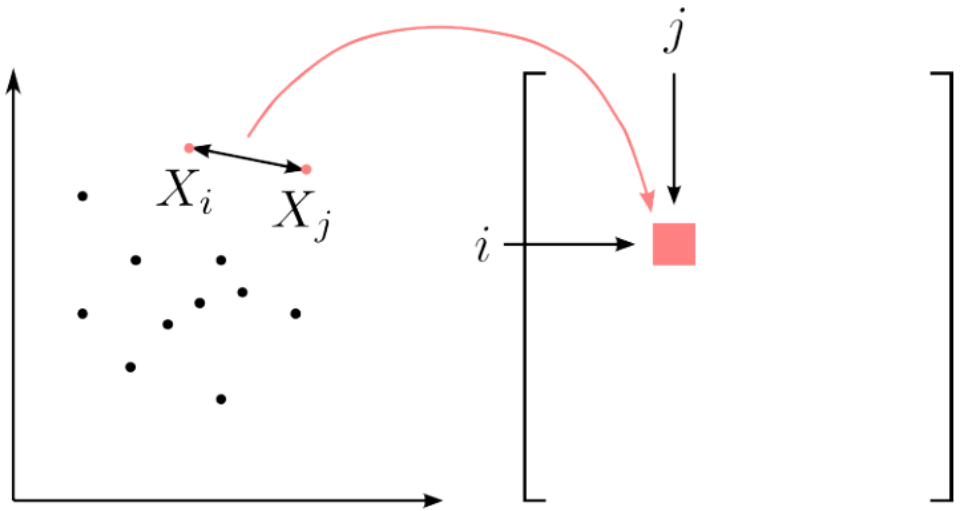
  - Ideally latent variables should provide an interpretable representation

(Continuous) latent space model

- Motivation: homophily or assortative mixing
  - Probability of edge between two nodes increases as characteristics of the nodes become more similar
- Represent nodes in an unobserved (latent) space of characteristics or "social space"
- Small distance between 2 nodes in latent space ➔ high probability of edge between nodes
  - Induces transitivity: observation of edges (i, j) and (j, k) suggests that i and k are not too far apart in latent space ➔ more likely to also have an edge

(Continuous) latent space model

- (Continuous) latent space model (LSM) proposed by Hoff et al. (2002)
  - Each node has a latent position $\mathbf{z}_i \in \mathbb{R}^d$
  - Probabilities of forming edges depend on distances between latent positions
  - Define pairwise affinities $\psi_{ij} = \theta - \|\mathbf{z}_i - \mathbf{z}_j\|_2$



Latent space model: generative process

1. Sample node positions in latent space

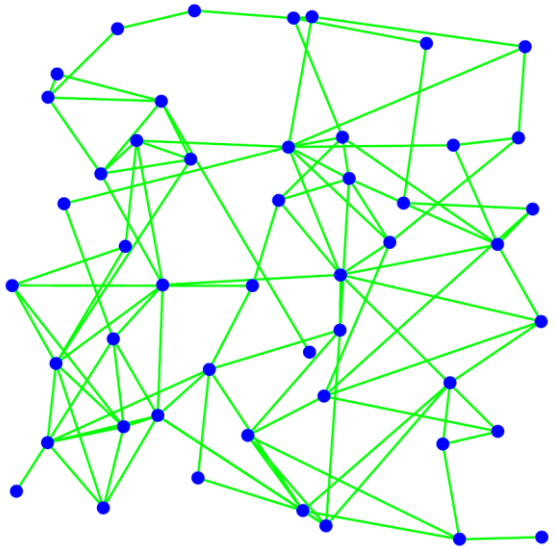
$$\mathbf{z}_i \sim \text{Gaussian}(\mathbf{0}, \kappa\mathbf{I})$$

2. Compute affinities between all pairs of nodes

$$\psi_{ij} = \theta - \|\mathbf{z}_i - \mathbf{z}_j\|_2$$

3. Sample edges between all pairs of nodes

$$P(Y_{ij} = 1|\psi_{ij}) = \sigma(\psi_{ij})$$



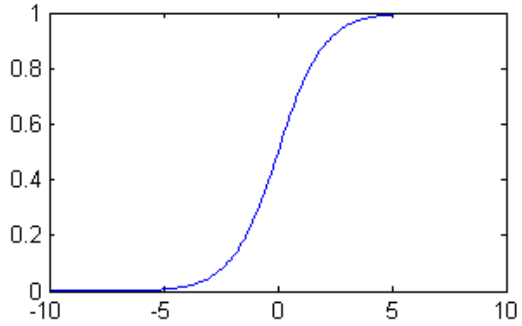


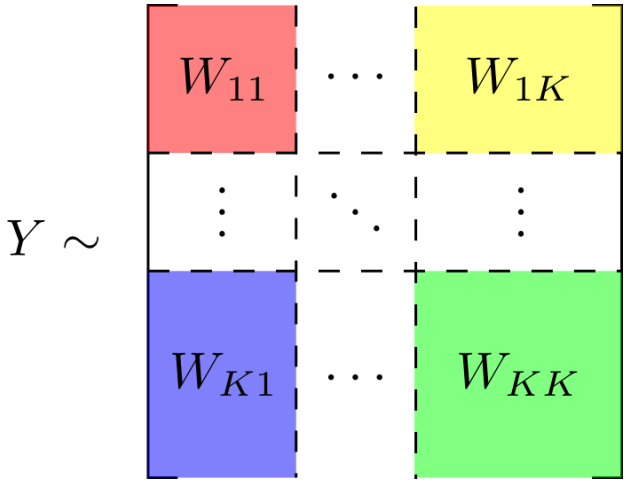
Figure due to P. D. Hoff, Modeling homophily and stochastic equivalence in symmetric relational data, NIPS 2008

Advantages and disadvantages of latent space model

- Advantages of latent space model
  - Visual and interpretable spatial representation of network
  - Models homophily (assortative mixing) well via transitivity
- Disadvantages of latent space model
  - 2-D latent space representation often may not offer enough degrees of freedom
  - Cannot model disassortative mixing (people preferring to associate with people with different characteristics)

Stochastic block model (SBM)

- First formalized by Holland et al. (1983)
- Also known as multi-class Erdős-Rényi model
- Each node has categorical latent variable $z_i \in \{1, \ldots, K\}$ denoting its class or group
- Probabilities of forming edges depend on class memberships of nodes ($K \times K$ matrix W)
  - Groups often interpreted as functional roles in social networks



Stochastic equivalence and block models

- Stochastic equivalence: generalization of structural equivalence
- Group members have identical probabilities of forming edges to members other groups
  - Can model both assortative and disassortative mixing

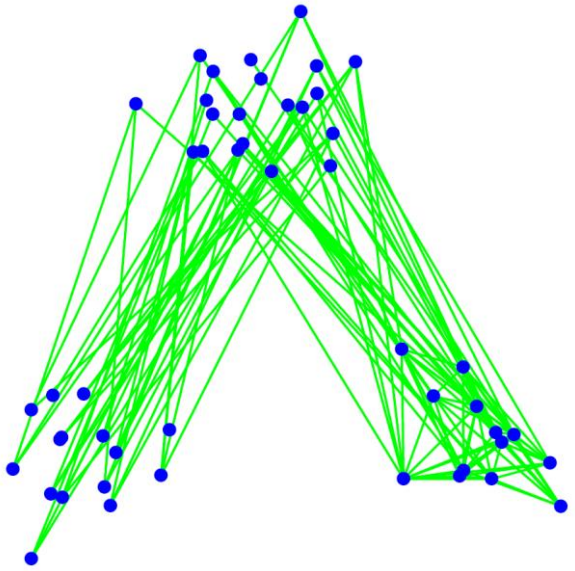
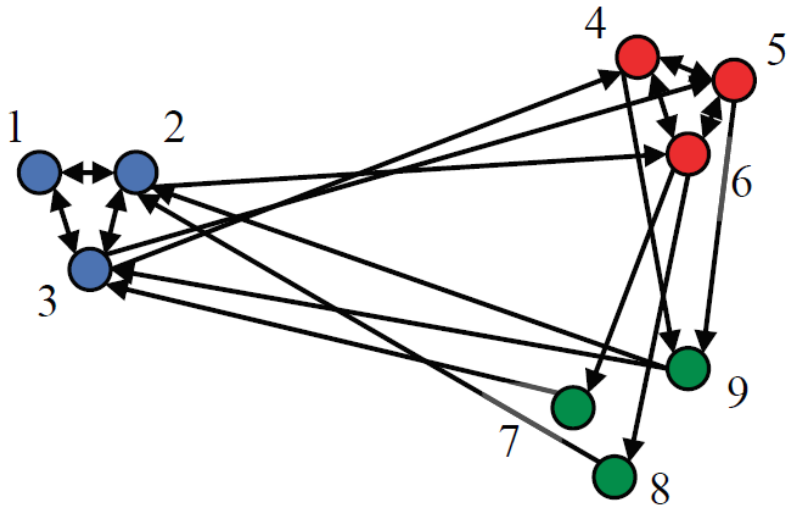


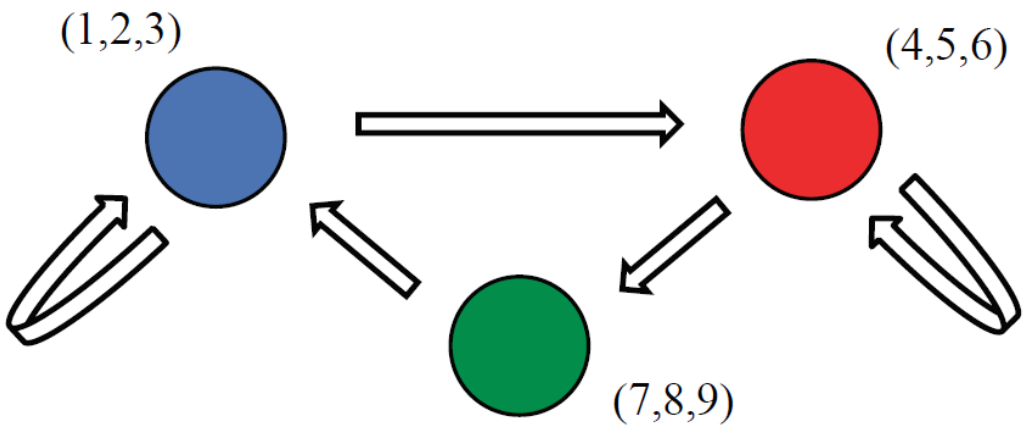
Figure due to P. D. Hoff, Modeling homophily and stochastic equivalence in symmetric relational data, NIPS 2008

Stochastic equivalence vs community detection

Original graph



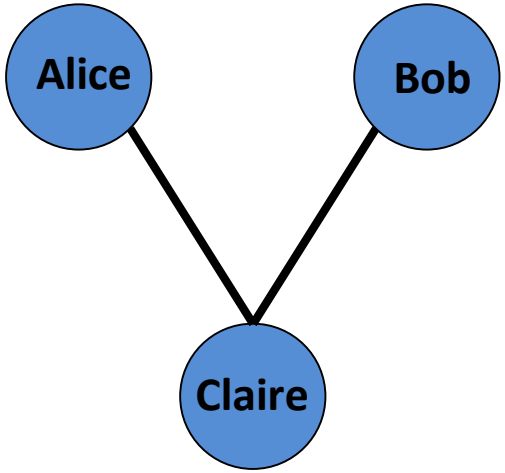
Blockmodel



Stochastically equivalent, but are not densely connected

Figure due to Goldenberg et al. (2009) - Survey of Statistical Network Models, Foundations and Trends

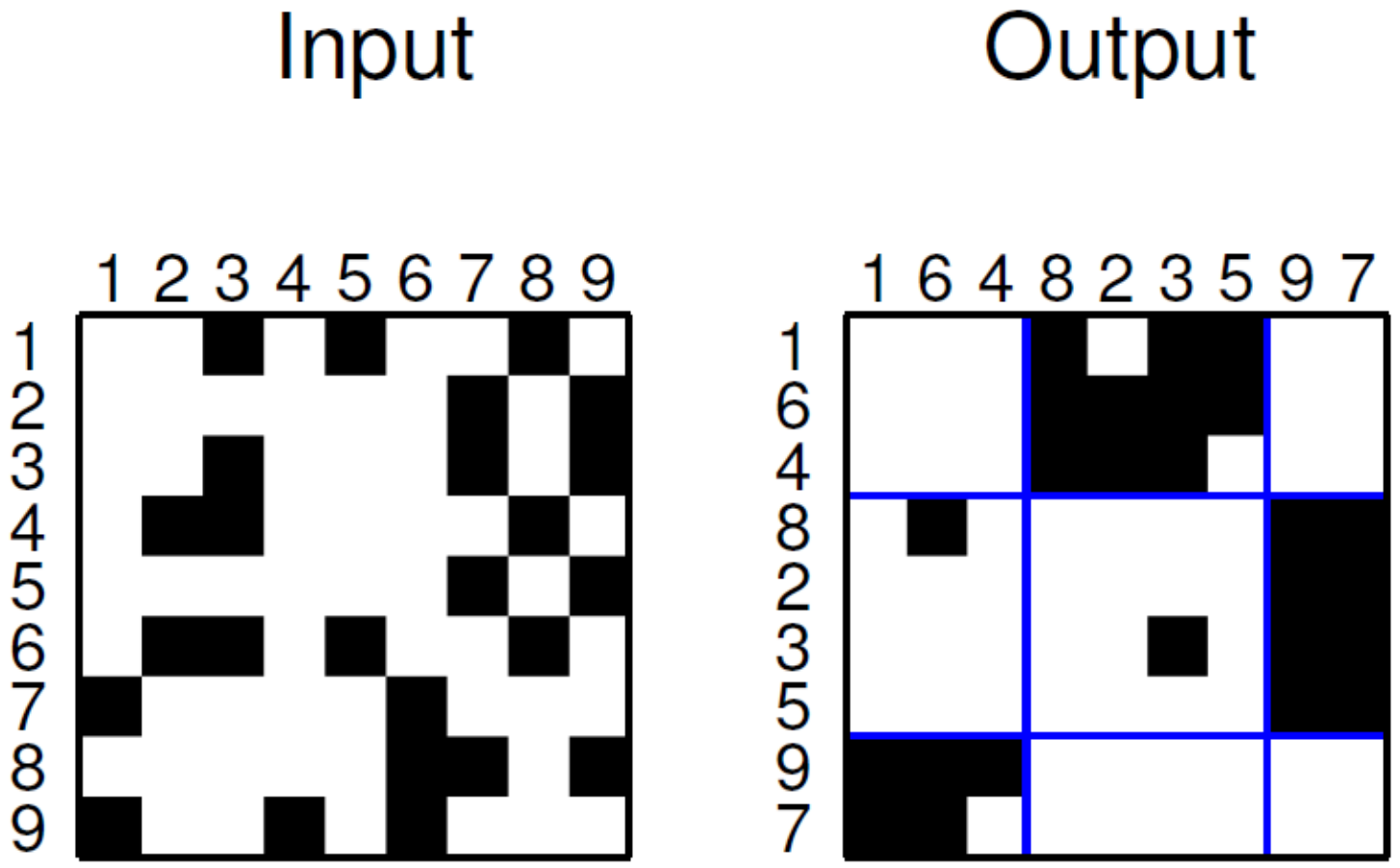
Stochastic blockmodel Latent representation



$$\mathbf{Z} =$$

	UCSD	UCI	UCLA
Alice	1		
Bob			1
Claire		1	

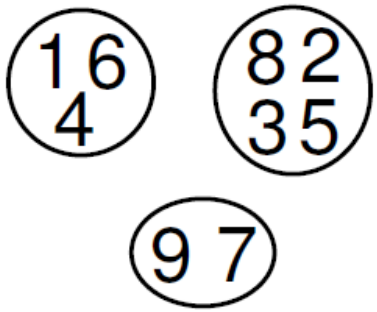
Reordering the matrix to show the inferred block structure



Kemp, Charles, et al. "Learning systems of concepts with an infinite relational model." AAAI. Vol. 3. 2006.

Model structure

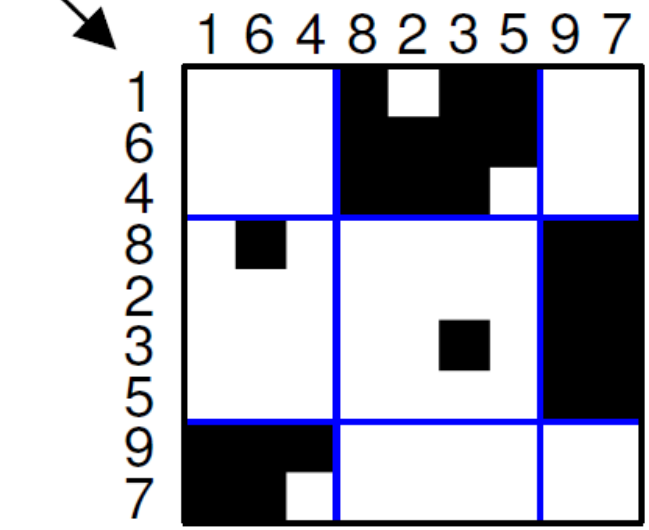
Latent groups Z



0.1	0.9	0.1
0.1	0.1	0.9
0.9	0.1	0.1

Interaction matrix W

(probability of an edge from block k to block k')



Kemp, Charles, et al. "Learning systems of concepts with an infinite relational model." AAAI. Vol. 3. 2006.

Stochastic block model generative process

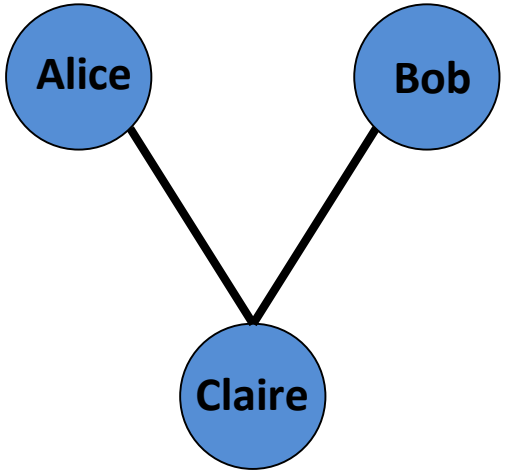
$W_{kk'}$: Probability that a node in group k connects to a node in k'

z_i: Latent group assignment for node i

For each pair of nodes (i, j)

$$Y_{ij} \sim \text{Bernoulli}(W_{z_i, z_j})$$

Stochastic block model
Latent representation



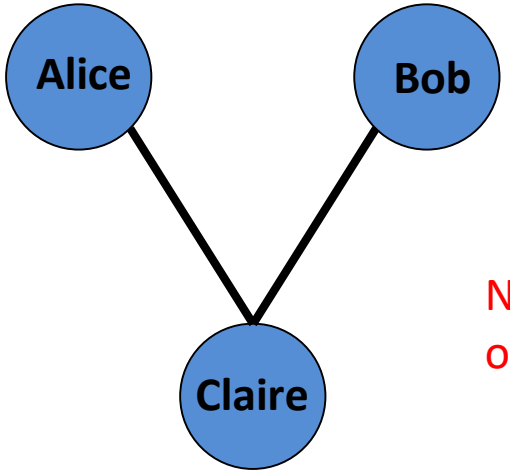
Nodes assigned to only one latent group.

Not always an appropriate assumption

$$\mathbf{Z} =$$

	Running	Dancing	Fishing
Alice	1		
Bob			1
Claire		1	

Mixed membership stochastic blockmodel (MMSB)



Nodes represented by **distributions** over latent groups (roles)

$$\mathbf{Z} =$$

	Running	Dancing	Fishing
Alice	0.4	0.4	0.2
Bob	0.5	0.5	
Claire	0.1		0.9

Airoldi et al., (2008)

Mixed membership stochastic blockmodel (MMSB)

$\pi^{(i)}$: Mixed membership vector for node i

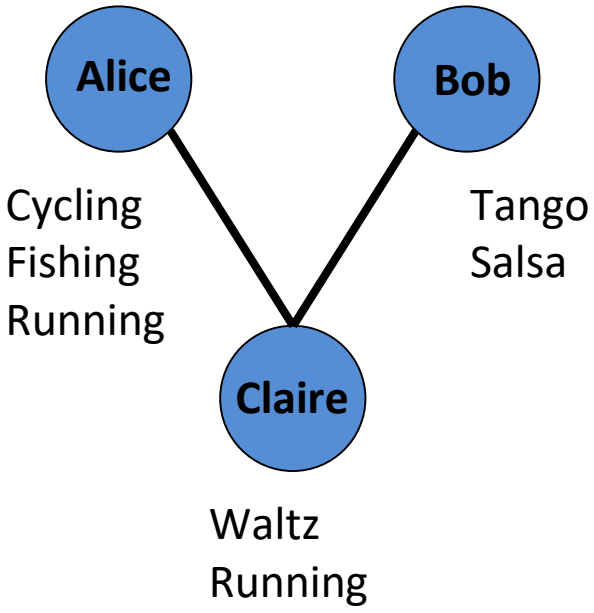
$W_{kk'}$: Probability that group k connects to group k'

For each pair of nodes (i, j)

$$z_i^{(ij)} \sim \text{discrete}(\pi^{(i)})$$
$$z_j^{(ij)} \sim \text{discrete}(\pi^{(j)})$$
$$Y_{ij} \sim \text{Bernoulli}(W_{z_i^{(ij)}, z_j^{(ij)}})$$

Airoldi et al., (2008)

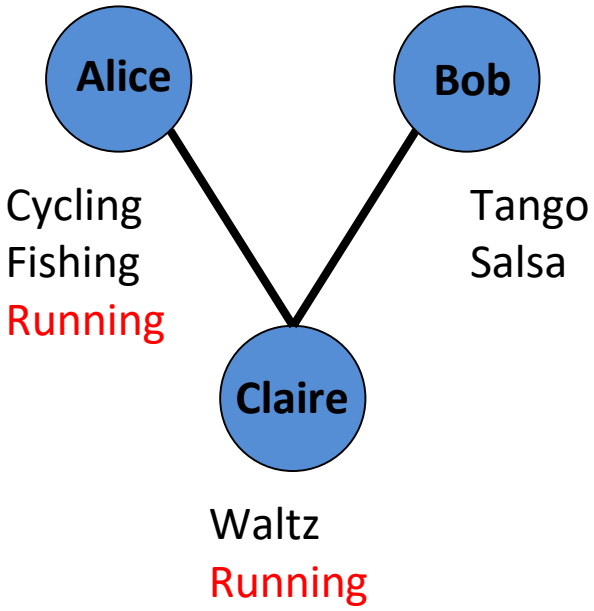
Latent feature models



Mixed membership implies a kind of "conservation of (probability) mass" constraint:
If you like cycling more, you must like running less, to sum to one

Miller, Griffiths, Jordan (2009)

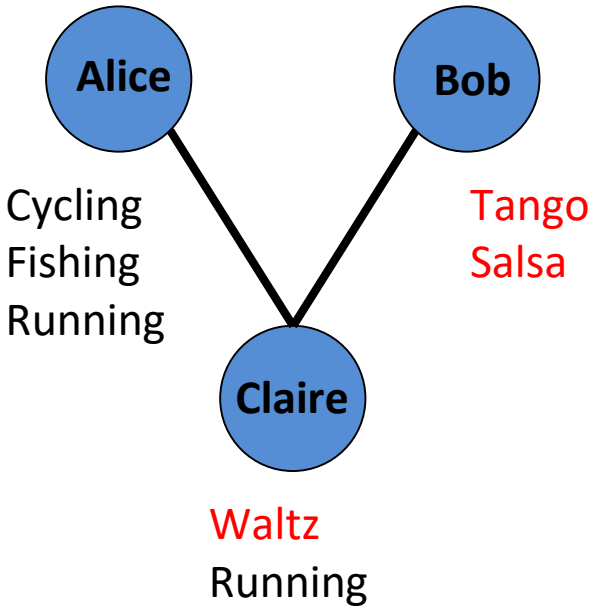
Latent feature models



Mixed membership implies a kind of "conservation of (probability) mass" constraint:
If you like cycling more, you must like running less, to sum to one

Miller, Griffiths, Jordan (2009)

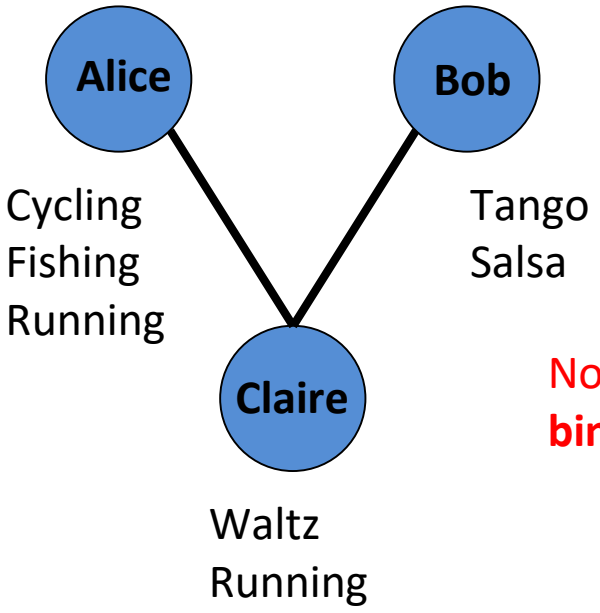
Latent feature models



Mixed membership implies a kind of "conservation of (probability) mass" constraint:
If you like cycling more, you must like running less, to sum to one

Miller, Griffiths, Jordan (2009)

Latent feature models



Nodes represented by
binary vector of latent features

Z =

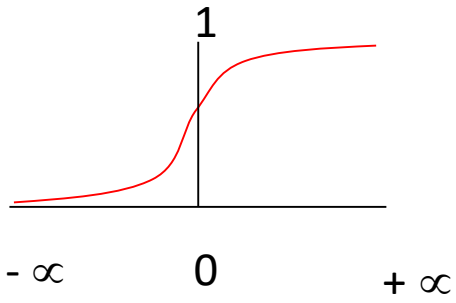
	Cycling	Fishing	Running	Tango	Salsa	Waltz
Alice	■	■	■			
Bob				■	■	
Claire			■			■

Miller, Griffiths, Jordan (2009)

Latent feature models

- Latent Feature Relational Model LFRM (Miller, Griffiths, Jordan, 2009) likelihood model:

$$P(Y_{ij} = 1 | \ldots) = \sigma(\mathbf{z}_i \mathbf{W} \mathbf{z}_j^\intercal)$$



- "If I have feature *k*, and you have feature *l*, add W_{kl} to the log-odds of the probability we interact"

- Can include terms for network density, covariates, popularity,…, as in the p2 model

Outline

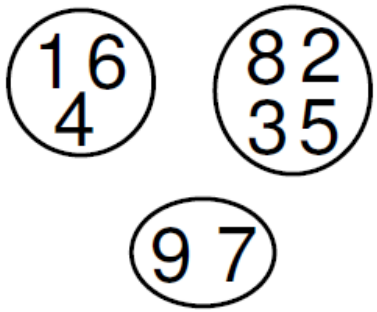
- Mathematical representations of social networks and generative models
  - Introduction to generative approach
  - Connections to sociological principles
- Fitting generative social network models to data
  - Example application scenarios
  - Model selection and evaluation

Application 1: Facebook wall posts

- Network of wall posts on Facebook collected by Viswanath et al. (2009)
  - Nodes: Facebook users
  - Edges: directed edge from i to j if i posts on j's Facebook wall
- What model should we use?
  - (Continuous) latent space and latent feature models do not handle directed graphs in a straightforward manner
  - Wall posts might not be transitive, unlike friendships
- Stochastic block model might not be a bad choice as a starting point

Model structure

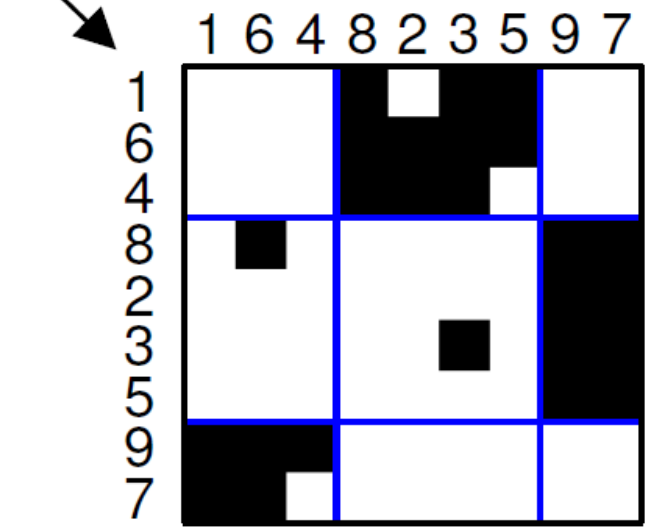
Latent groups Z



0.1	0.9	0.1
0.1	0.1	0.9
0.9	0.1	0.1

Interaction matrix W

(probability of an edge from block k to block k')



Kemp, Charles, et al. "Learning systems of concepts with an infinite relational model." AAAI. Vol. 3. 2006.

Fitting stochastic block model

- A priori block model: assume that class (role) of each node is given by some other variable
  - Only need to estimate $W_{kk'}$: probability that node in class k connects to node in class k' for all k, k'
- Likelihood given by

Number of actual edges in block (k, k')

Number of possible edges in block (k, k')

$$Pr(\mathbf{Y}|\mathbf{W}, \mathbf{Z}) = \exp\left\{\sum_{k=1}^{K}\sum_{k'=1}^{K}\left[m_{kk'}\log W_{kk'} + (n_{kk'} - m_{kk'})\log(1 - W_{kk'}\right]\right\}$$

- Maximum-likelihood estimate (MLE) given by

$$\hat{W}_{kk'} = \frac{m_{kk'}}{n_{kk'}}$$

Estimating latent classes

- Latent classes (roles) are unknown in this data set
  - First estimate latent classes $\mathbf{Z}$ then use MLE for $\mathbf{W}$
- MLE over latent classes is intractable!
  - $\sim K^N$ possible latent class vectors
- Spectral clustering techniques have been shown to accurately estimate latent classes
  - Use singular vectors of (possibly transformed) adjacency matrix to estimate classes
  - Many variants with differing theoretical guarantees

Spectral clustering for directed SBMs

1. Compute singular value decomposition $Y = U\Sigma V^T$
2. Retain only first K columns of U, V and first K rows and columns of Σ
3. Define coordinate-scaled singular vector matrix $\tilde{Z} = \left[U\Sigma^{1/2}\ V\Sigma^{1/2}\right]$
4. Run k-means clustering on rows of $\tilde{Z}$ to return estimate $\hat{Z}$ of latent classes

Scales to networks with thousands of nodes!

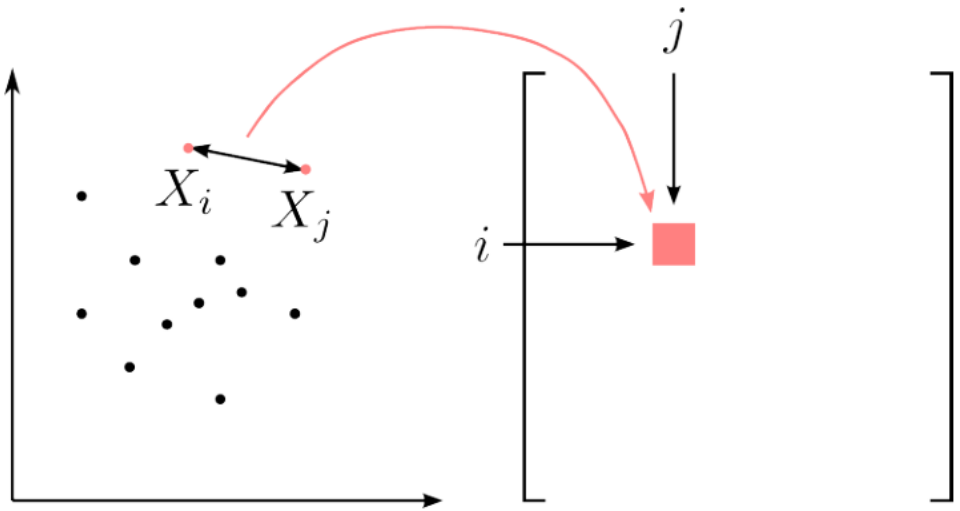
Demo of SBM on Facebook wall post network

Application 2: social network of bottlenose dolphin interactions

- Data collected by marine biologists observing interactions between 62 bottlenose dolphins
  - Introduced to network science community by Lusseau and Newman (2004)
  - Nodes: dolphins
  - Edges: undirected relations denoting frequent interactions between dolphins
- What model should we use?
  - Social interactions here are in a group setting so lots of transitivity may be expected
  - Interactions associated by physical proximity
  - Use latent space model to estimate latent positions

(Continuous) latent space model

- (Continuous) latent space model (LSM) proposed by Hoff et al. (2002)
  - Each node has a latent position $\mathbf{z}_i \in \mathbb{R}^d$
  - Probabilities of forming edges depend on distances between latent positions
  - Define pairwise affinities $\psi_{ij} = \theta - \|\mathbf{z}_i - \mathbf{z}_j\|_2$



$$p(Y|Z,\theta) = \prod_{i \neq j} \frac{e^{y_{ij}\psi_{ij}}}{1 + e^{\psi_{ij}}}$$

Estimation for latent space model

- Maximum-likelihood estimation
  - Log-likelihood is concave in terms of pairwise distance matrix D but not in latent positions Z
  - First find MLE in terms of D then use multi-dimensional scaling (MDS) to get initialization for Z
  - Faster approach: replace D with shortest path distances in graph then use MDS
  - Use non-linear optimization to find MLE for Z
- Latent space dimension often set to 2 to allow visualization using scatter plot

Scales to ~1000 nodes

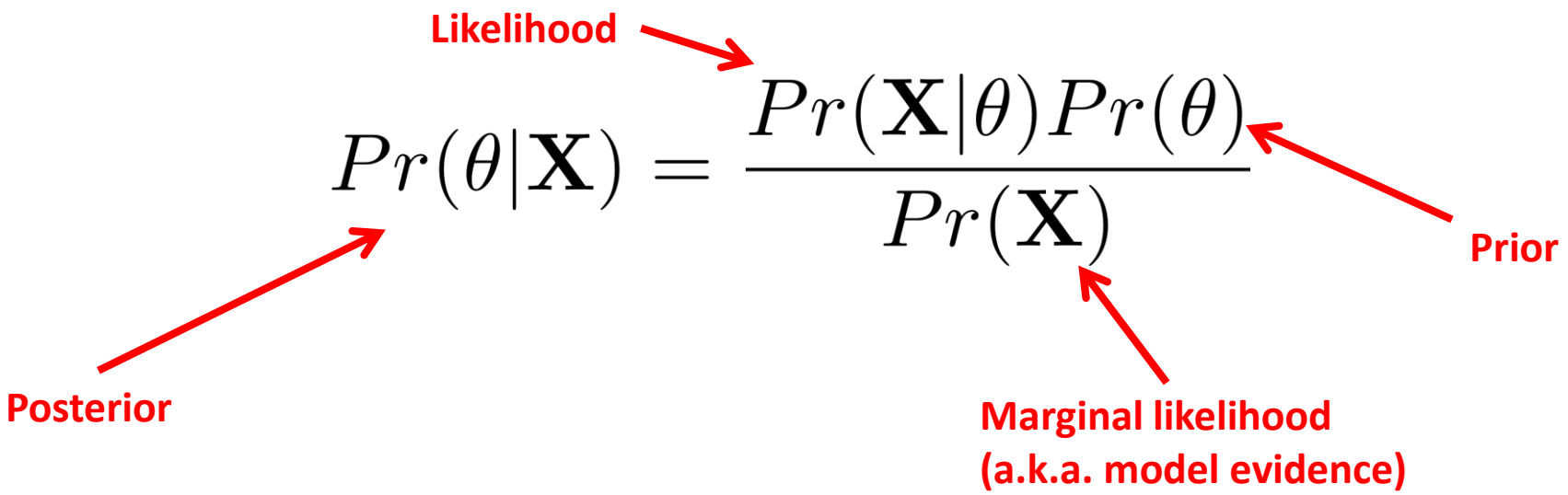
Demo of latent space model on dolphin network

Bayesian inference

- As a Bayesian, all you have to do is write down your prior beliefs, write down your likelihood, and apply Bayes ‘ rule,

$$Pr(\theta|\mathbf{X}) = \frac{Pr(\mathbf{X}|\theta)Pr(\theta)}{Pr(\mathbf{X})}$$

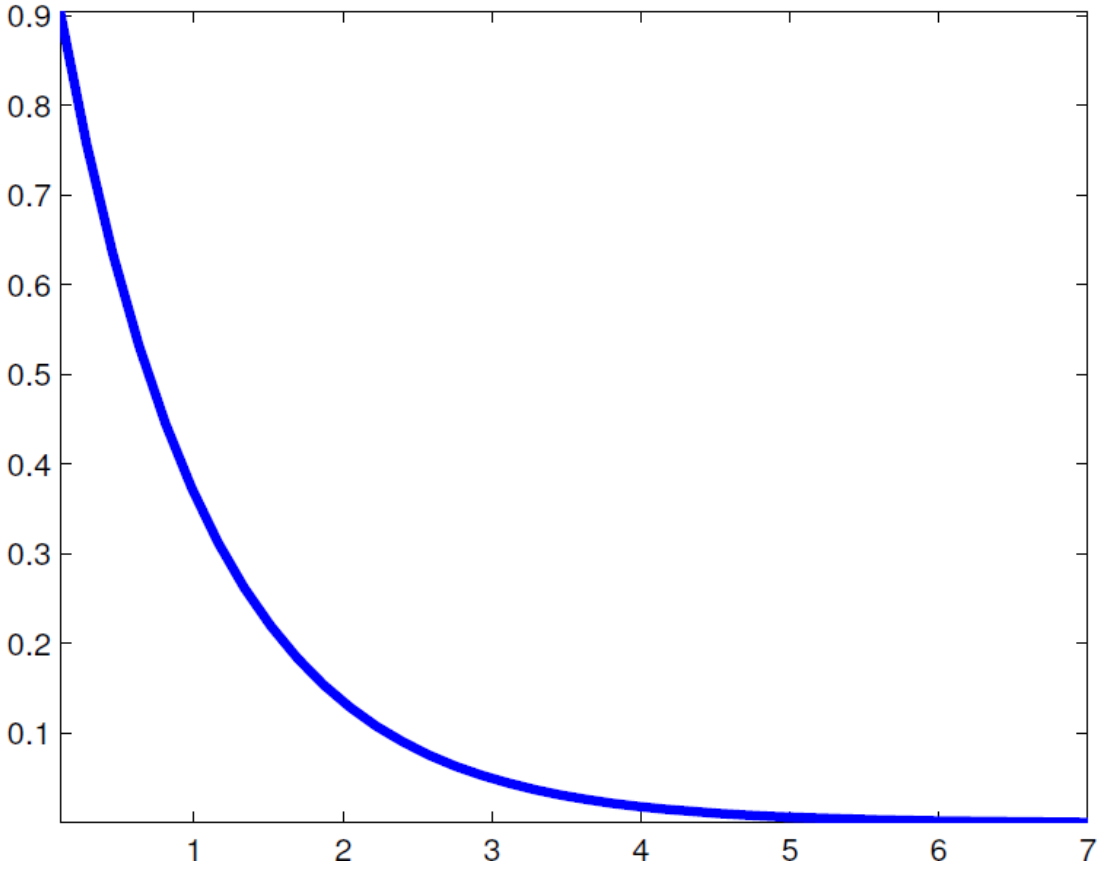
Elements of Bayesian Inference



$$Pr(\mathbf{X}) = \int Pr(\mathbf{X}|\theta)Pr(\theta)d\theta$$

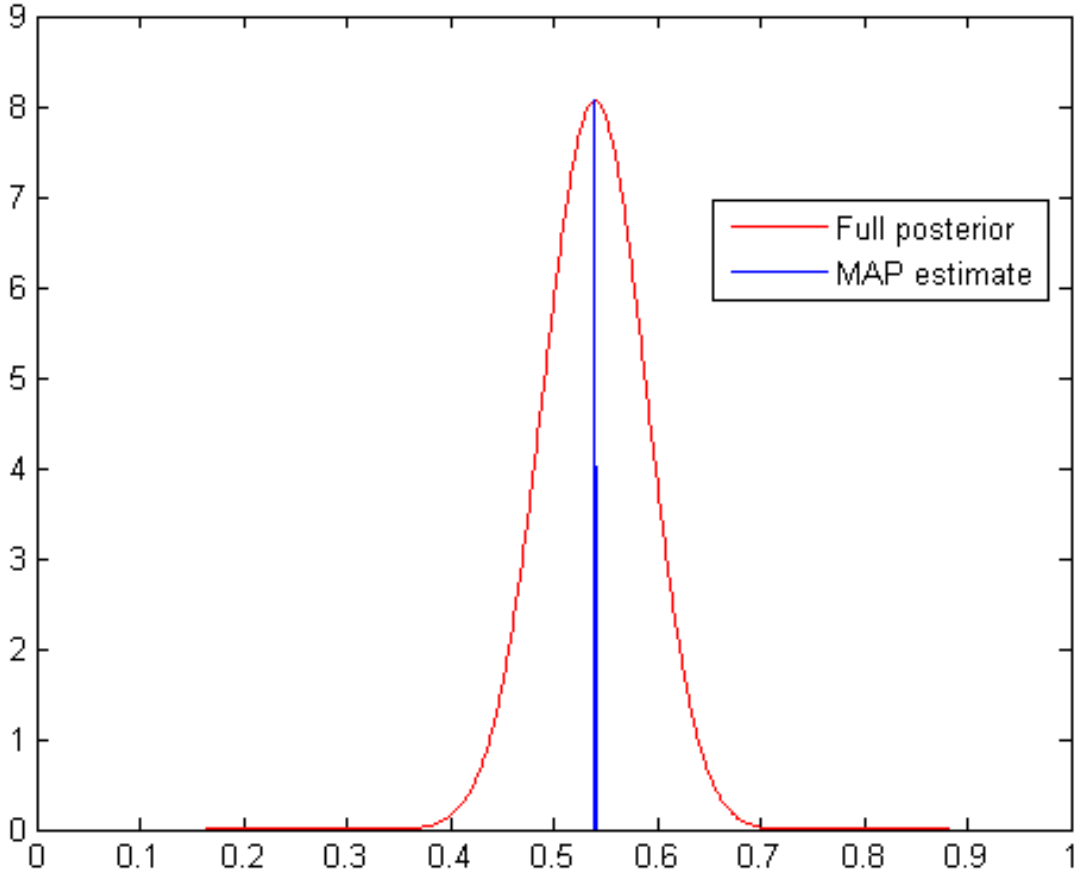
is a normalization constant that does not depend on the value of θ. It is the probability of the data under the model, marginalizing over all possible θ's.

The full posterior distribution can be very useful



The mode (MAP estimate) is unrepresentative of the distribution

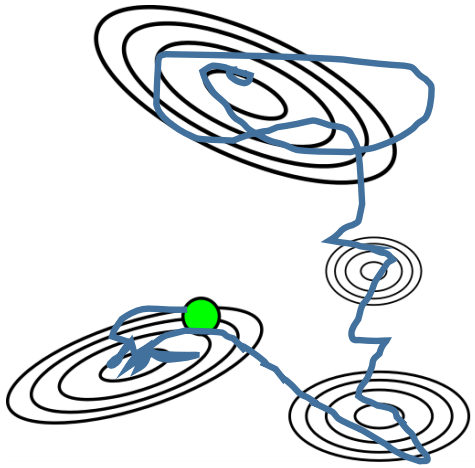
MAP estimate can result in overfitting



Markov chain Monte Carlo

- **Goal**: approximate/summarize a distribution, e.g. the posterior, with a set of samples

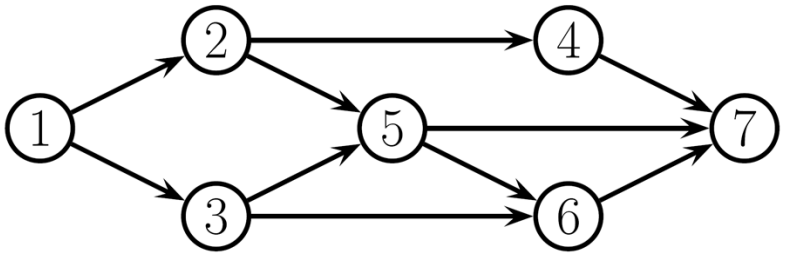
- **Idea**: use a Markov chain to simulate the distribution and draw samples



Gibbs sampling

- Sampling from a complicated distribution, such as a Bayesian posterior, can be hard.

- Often, sampling one variable at a time, given all the others, is much easier.



- Graphical models:
  Graph structure gives us Markov blanket

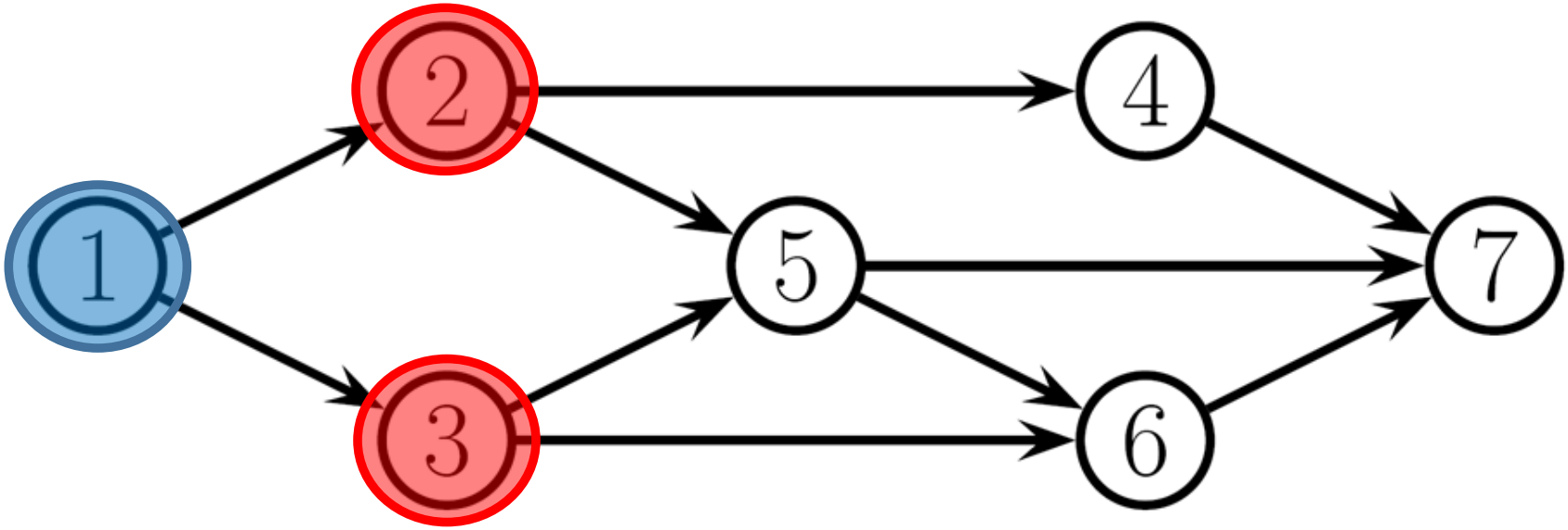
Gibbs sampling

- Update variables one at a time by drawing from their conditional distributions

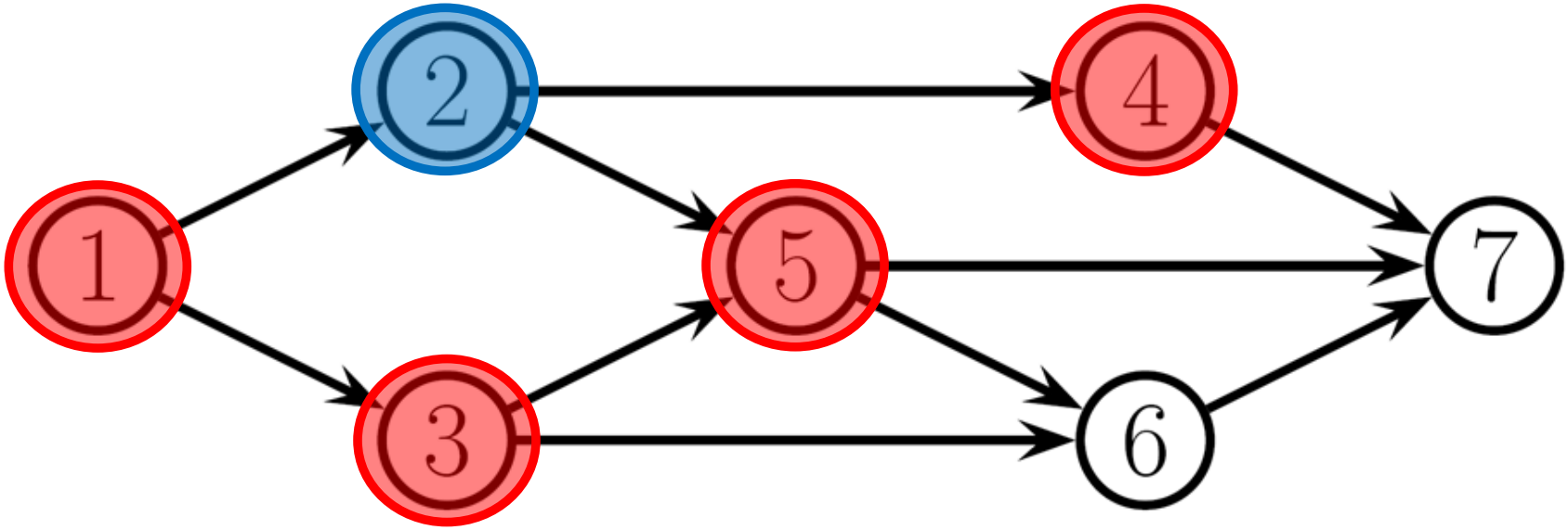
$$\mathbf{z}_i := \mathbf{z}_i^{(new)}, \mathbf{z}_i^{(new)} \sim Pr(\mathbf{z}_i | \mathbf{z}_{\neg i})$$

- In each iteration, sweep through and update all of the variables, in any order.

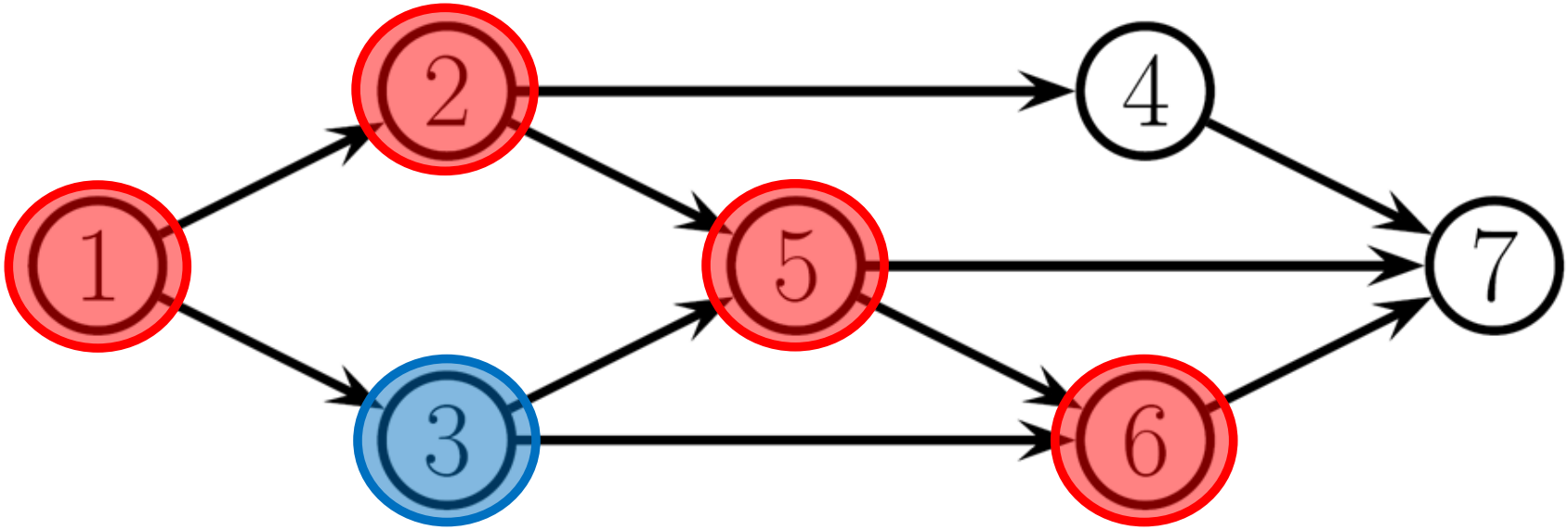
Gibbs sampling



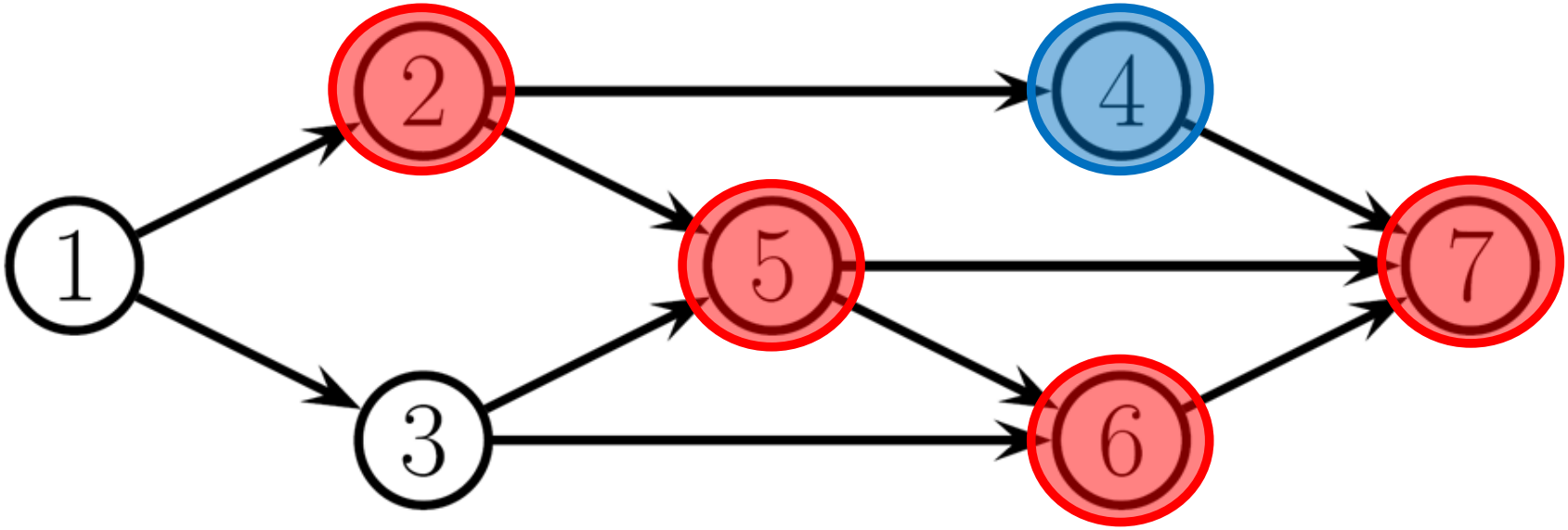
Gibbs sampling



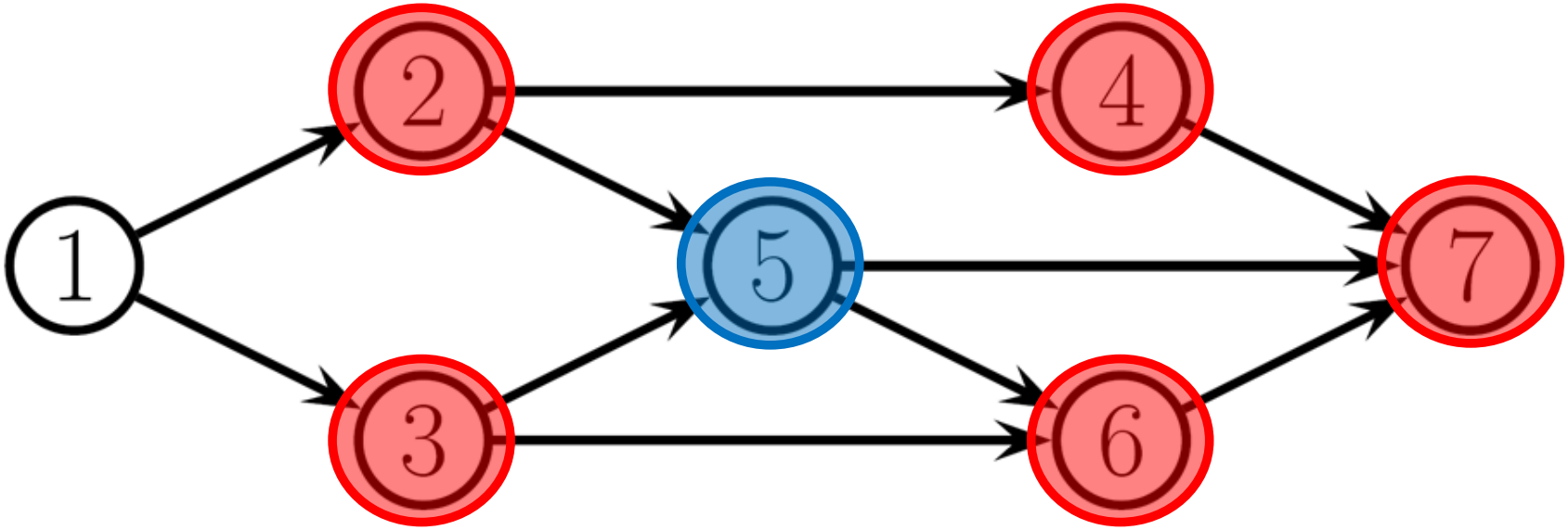
Gibbs sampling



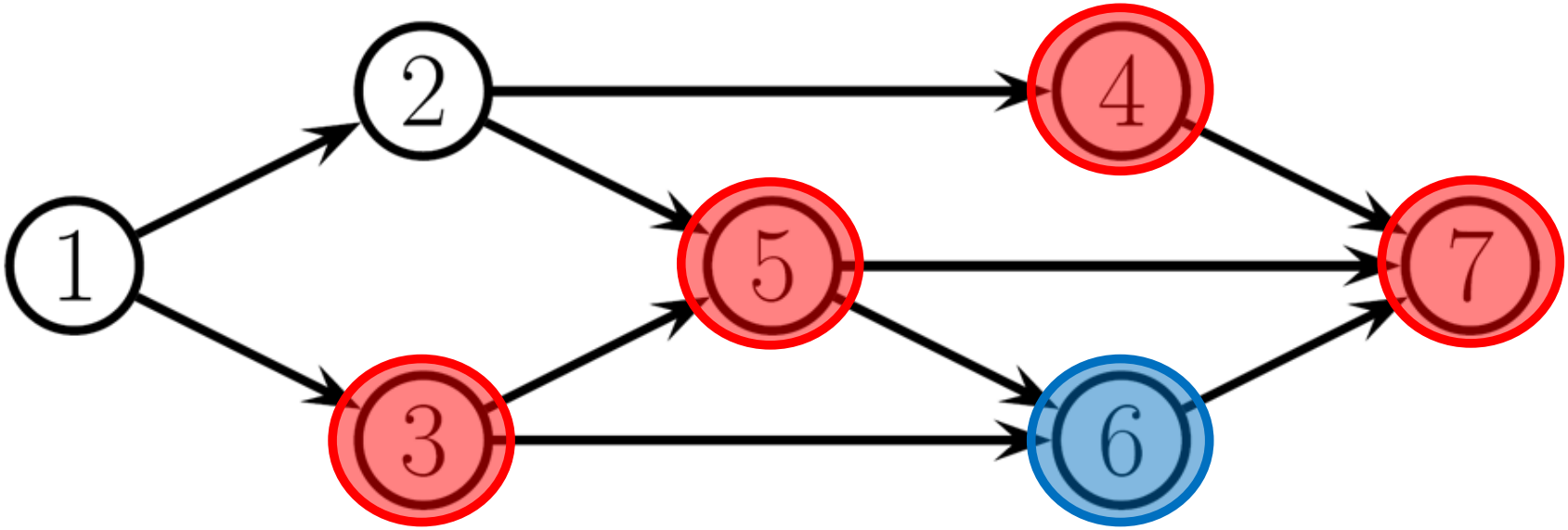
Gibbs sampling



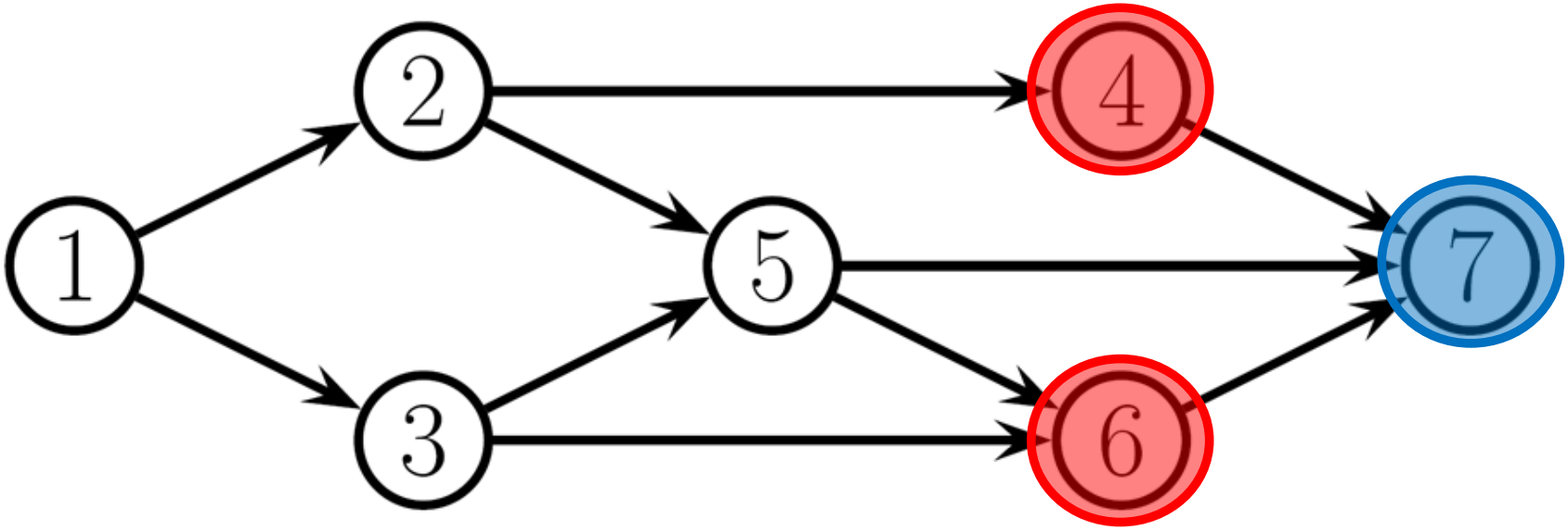
Gibbs sampling



Gibbs sampling



Gibbs sampling



Gibbs sampling for SBM

Initialize group assignments and parameters randomly

Until converged

For each pair of groups k, k'

$$W_{kk'} \sim \text{Beta}(n^{(1)}_{kk'} + \alpha_1, n^{(0)}_{kk'} + \alpha_0)$$

$$\pi \sim \text{Dirichlet}([n_1 + \alpha_1, \ldots, n_K + \alpha_K])$$

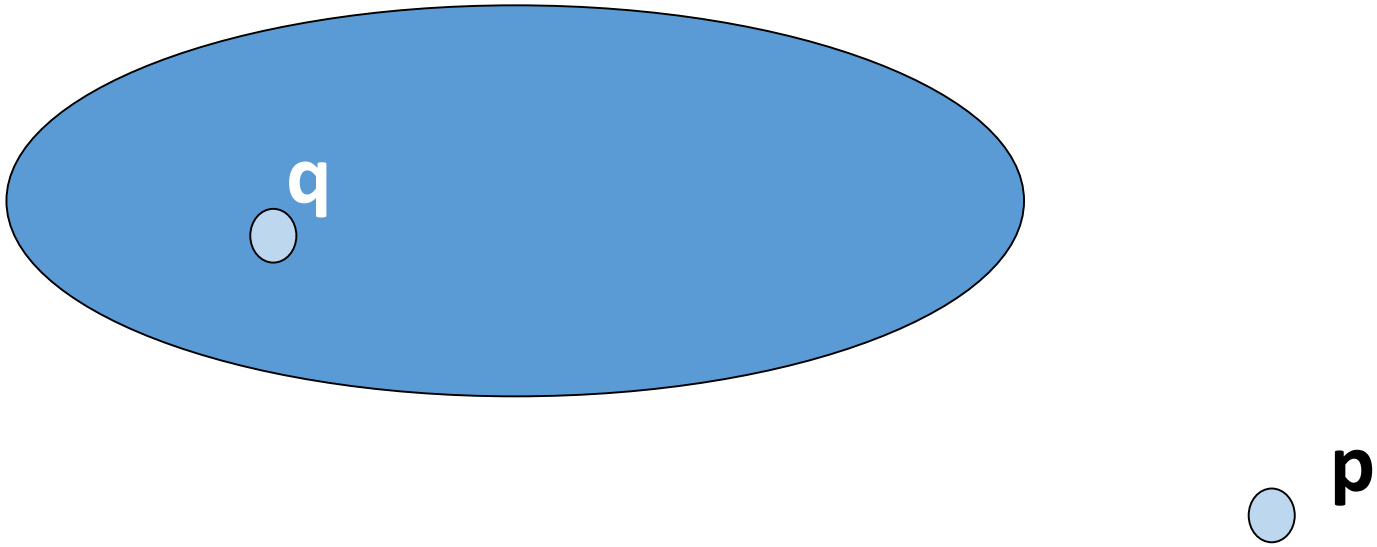
For node i

$$Pr(z_i = k) \propto \pi_k \prod_{k'=1}^{K} W_{kk'}^{n^{(1)}_{i,k'}} (1 - W_{kk'})^{n^{(0)}_{i,k'}}$$

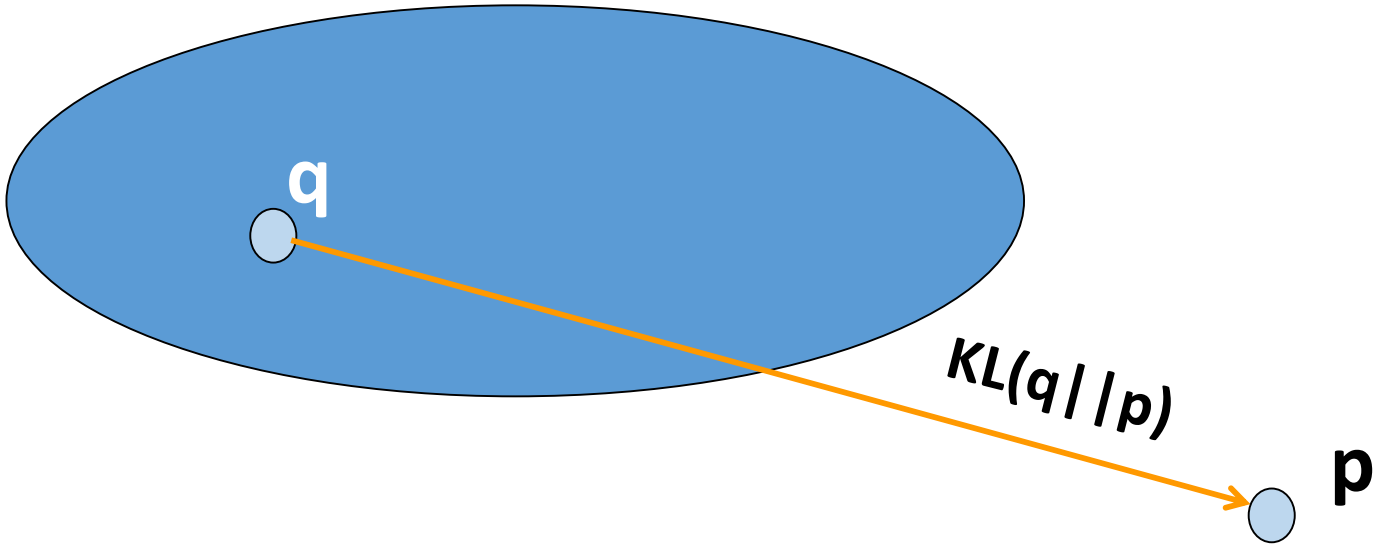
Variational inference

- Key idea:
  - Approximate distribution of interest p(z) with another distribution q(z)
  - Make q(z) tractable to work with
  - Solve an optimization problem to make q(z) as similar to p(z) as possible, e.g. in KL-divergence

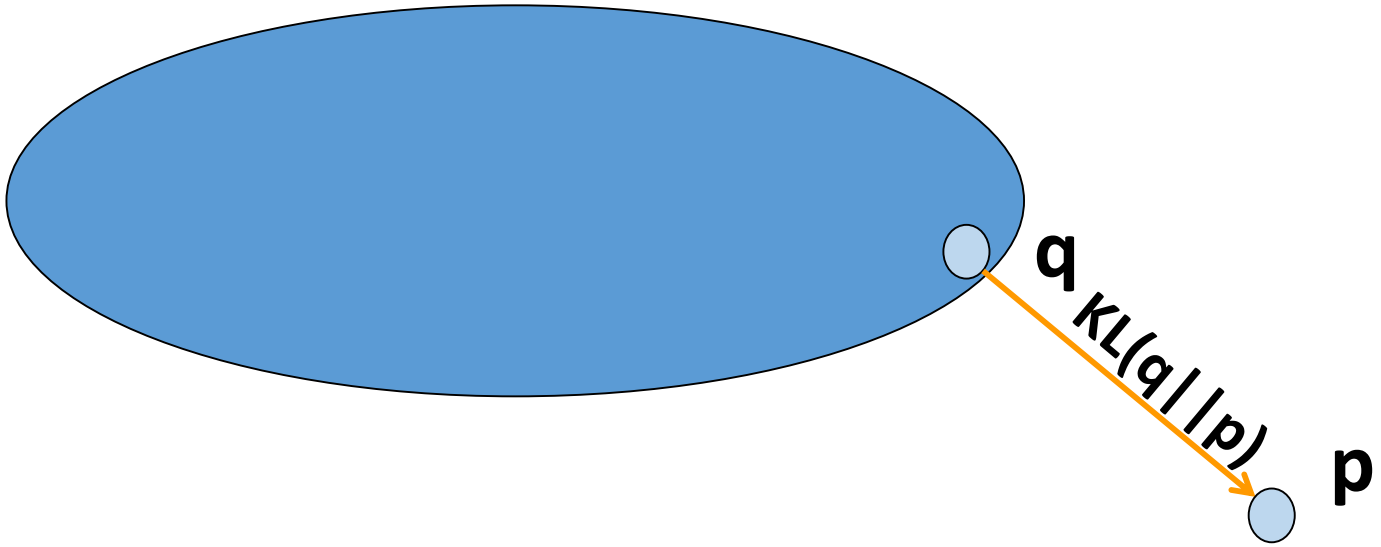
Variational inference



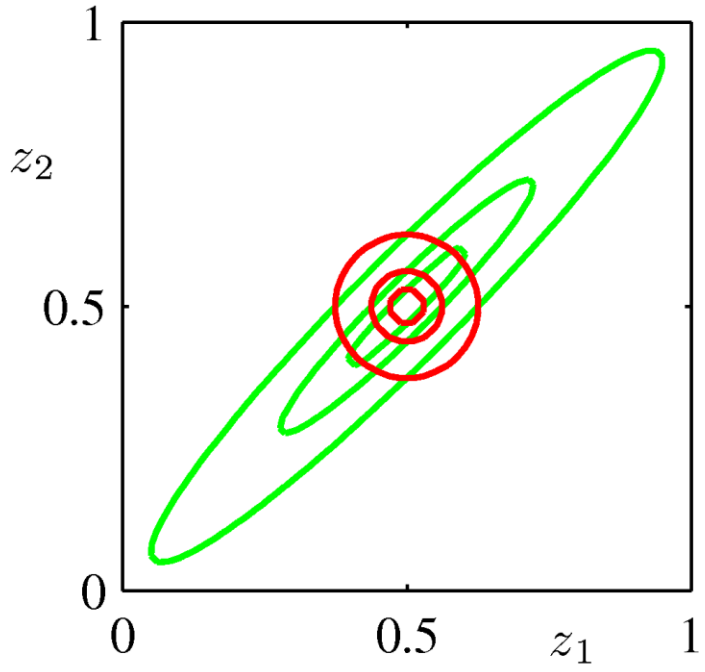
Variational inference



Variational inference

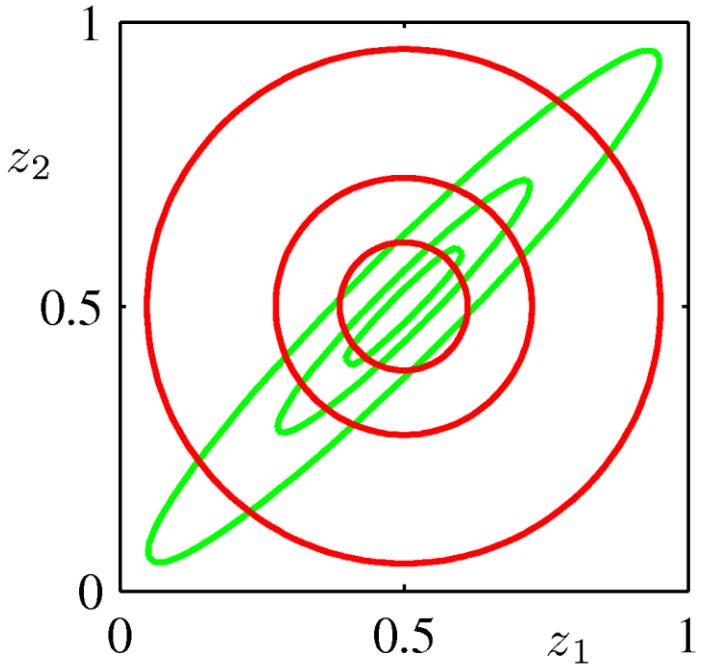


Reverse KL



(a)

Forwards KL



(b)

$$D_{KL}(q(\mathbf{z})\|p(\mathbf{z})) = \sum_{\mathbf{z}} q(\mathbf{z}) \log \frac{q(\mathbf{z})}{p(\mathbf{z})}$$

Blows up if *p* is small and *q* isn't.
Under-estimates the support

$$D_{KL}(p(\mathbf{z})\|q(\mathbf{z})) = \sum_{\mathbf{z}} p(\mathbf{z}) \log \frac{p(\mathbf{z})}{q(\mathbf{z})}$$

Blows up if *q* is small and *p* isn't.
Over-estimates the support

Figures due to Kevin Murphy (2012). Machine Learning: A Probabilistic Perspective

KL-divergence as an objective function for variational inference

$$
\begin{aligned}
D_{KL}(q(\mathbf{z})\|p(\mathbf{z}|\mathbf{x})) &= E_q\Big[\log \frac{q(\mathbf{z})}{p(\mathbf{z}|\mathbf{x})}\Big] \\
&= E_q[\log q(\mathbf{z})] - E_q[\log p(\mathbf{z}|\mathbf{x})] \\
&= E_q[\log q(\mathbf{z})] - E_q[\log p(\mathbf{z},\mathbf{x})] + \log p(\mathbf{x})
\end{aligned}
$$

KL-divergence as an objective function for variational inference

$$
\begin{aligned}
D_{KL}(q(\mathbf{z})\|p(\mathbf{z}|\mathbf{x})) &= E_q\Big[\log \frac{q(\mathbf{z})}{p(\mathbf{z}|\mathbf{x})}\Big] \\
&= E_q[\log q(\mathbf{z})] - E_q[\log p(\mathbf{z}|\mathbf{x})] \\
&= E_q[\log q(\mathbf{z})] - E_q[\log p(\mathbf{z},\mathbf{x})] + \log p(\mathbf{x})
\end{aligned}
$$

- Minimizing the KL is equivalent to maximizing

$$\mathcal{L}(q) = E_q[\log p(\mathbf{z},\mathbf{x})] - E_q[\log q(\mathbf{z})]$$

KL-divergence as an objective function for variational inference

$$
\begin{aligned}
D_{KL}(q(\mathbf{z})\|p(\mathbf{z}|\mathbf{x})) &= E_q\Big[\log \frac{q(\mathbf{z})}{p(\mathbf{z}|\mathbf{x})}\Big] \\
&= E_q[\log q(\mathbf{z})] - E_q[\log p(\mathbf{z}|\mathbf{x})] \\
&= E_q[\log q(\mathbf{z})] - E_q[\log p(\mathbf{z},\mathbf{x})] + \log p(\mathbf{x})
\end{aligned}
$$

- Minimizing the KL is equivalent to maximizing

$$
\begin{aligned}
\mathcal{L}(q) &= E_q[\log p(\mathbf{z},\mathbf{x})] - E_q[\log q(\mathbf{z})] \\
&= E_q[\log p(\mathbf{z},\mathbf{x})] + H[q]
\end{aligned}
$$

KL-divergence as an objective function for variational inference

$$
\begin{aligned}
D_{KL}(q(\mathbf{z})\|p(\mathbf{z}|\mathbf{x})) &= E_q\left[\log \frac{q(\mathbf{z})}{p(\mathbf{z}|\mathbf{x})}\right] \\
&= E_q[\log q(\mathbf{z})] - E_q[\log p(\mathbf{z}|\mathbf{x})] \\
&= E_q[\log q(\mathbf{z})] - E_q[\log p(\mathbf{z},\mathbf{x})] + \log p(\mathbf{x})
\end{aligned}
$$

- Minimizing the KL is equivalent to maximizing

$$
\begin{aligned}
\mathcal{L}(q) &= E_q[\log p(\mathbf{z},\mathbf{x})] - E_q[\log q(\mathbf{z})] \\
&= E_q[\log p(\mathbf{z},\mathbf{x})] + H[q]
\end{aligned}
$$

Fit the data well

Be flat

Mean field variational inference

- We still need to compute expectations over ***z***
- However, we have gained the option to restrict ***q(z)*** to make these expectations tractable.
- The mean field approach uses a fully factorized ***q(z)***

$$q(\mathbf{z}) = \prod_i q_i(z_i)$$

The entropy term decomposes nicely:

$$-E_q[\log q(\mathbf{z})] = -E_q[\log \prod_i q_i(z_i)] = \sum_i E_{q_i}[-\log q_i(z_i)] = \sum_i H(q_i)$$

Mean field algorithm

- Until converged
  - For each factor *i*
    - Select variational parameters γ_i such that

$$q_i(z_i|\gamma_i) :\propto \exp(E_{q_{\neg i}}[\log p(\mathbf{z}, \mathbf{x})])$$

- Each update monotonically improves the ELBO so the algorithm must converge

Deriving mean field updates for your model

- Write down the mean field equation explicitly,

$$\log q_i(z_i) := E_{q_{\neg i}}[\log p(\mathbf{z}, \mathbf{x})] + \text{ const}$$

- Simplify and apply the expectation.

- Manipulate it until you can recognize it as a log-pdf of a known distribution (hopefully).

- Reinstate the normalizing constant.

Mean field vs Gibbs sampling

- Both mean field and Gibbs sampling iteratively update one variable given the rest

- Mean field stores an entire distribution for each variable, while Gibbs sampling draws from one.

Pros and cons vs Gibbs sampling

- **Pros:**
  - Deterministic algorithm, typically converges faster
  - Stores an analytic representation of the distribution, not just samples
  - Non-approximate parallel algorithms
  - Stochastic algorithms can scale to very large data sets
  - No issues with checking convergence

- **Cons:**
  - Will never converge to the true distribution, unlike Gibbs sampling
  - Dense representation can mean more communication for parallel algorithms
  - Harder to derive update equations

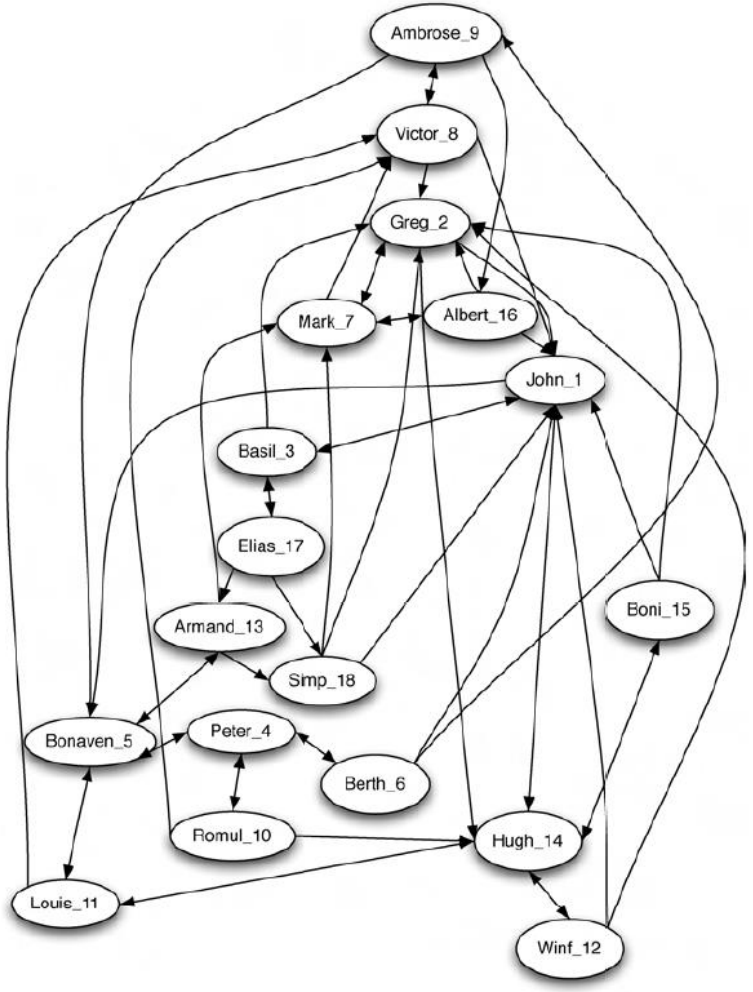
Variational inference algorithm for MMSB (Variational EM)

- Compute maximum likelihood estimates for interaction parameters $W_{kk'}$
- Assume fully factorized variational distribution for mixed membership vectors, cluster assignments

- Until converged
  - For each node
    - Compute variational discrete distribution over it's latent $z_{p->q}$ and $z_{q->p}$ assignments
    - Compute variational Dirichlet distribution over its mixed membership distribution
  - Maximum likelihood update for $\boldsymbol{W}$

Application of MMSB to Sampson's Monastery

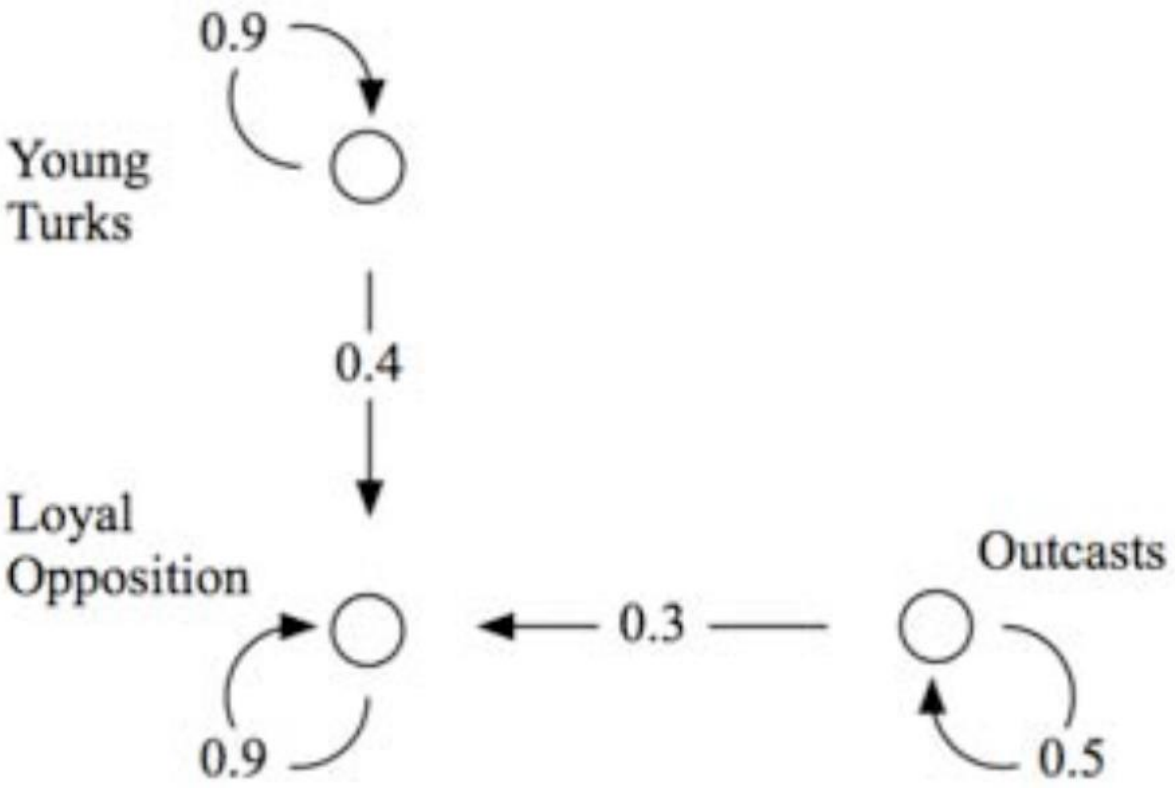
- Sampson (1968) studied friendship relationships between novice monks
- Identified several factions
  - Blockmodel appropriate?
- Conflicts occurred
  - Two monks expelled
  - Others left



Airoldi, E. M., Blei, D. M., Fienberg, S. E., & Xing, E. P. (2009). Mixed membership stochastic blockmodels. In Advances in Neural Information Processing Systems (pp. 33-40).

Application of MMSB to Sampson's Monastery

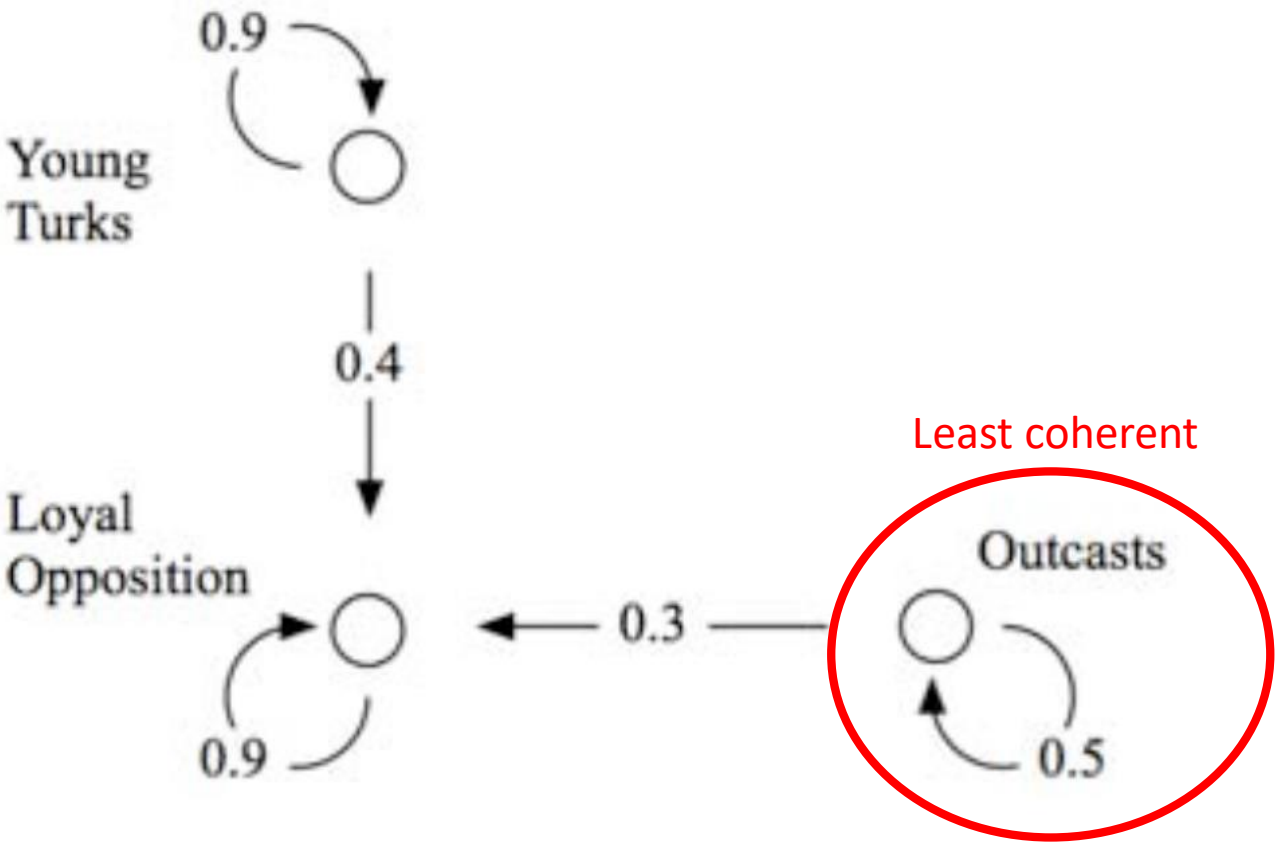
Estimated blockmodel



Airoldi, E. M., Blei, D. M., Fienberg, S. E., & Xing, E. P. (2009). Mixed membership stochastic blockmodels. In Advances in Neural Information Processing Systems (pp. 33-40).

Application of MMSB to Sampson's Monastery

Estimated blockmodel

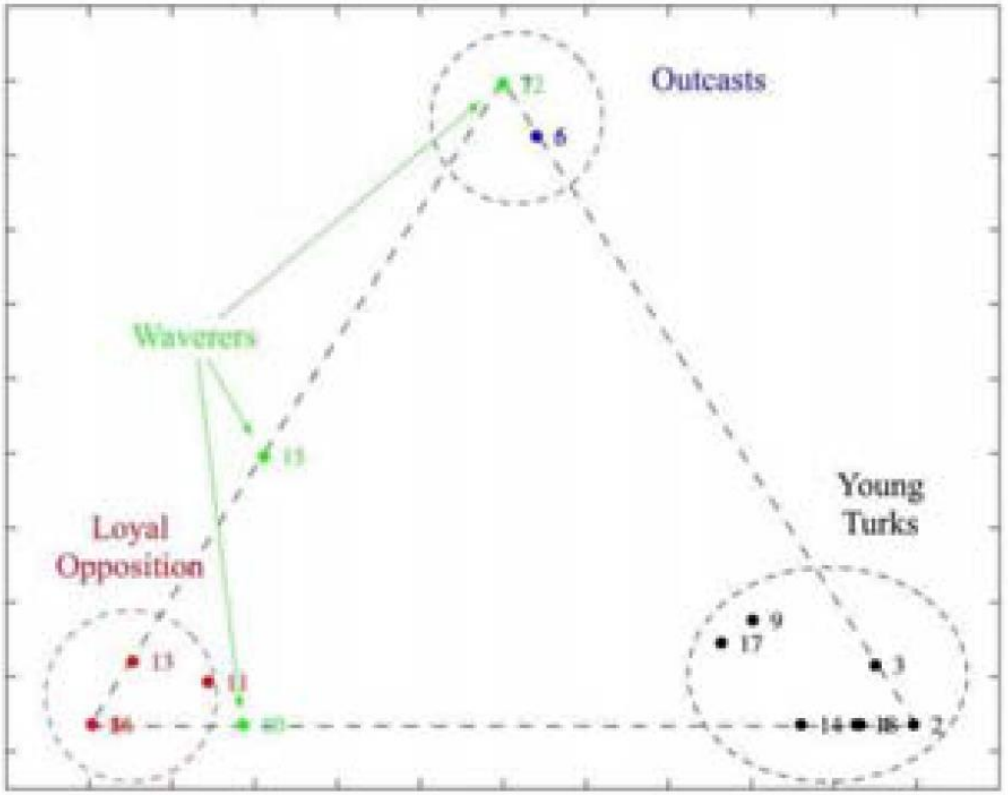


Airoldi, E. M., Blei, D. M., Fienberg, S. E., & Xing, E. P. (2009). Mixed membership stochastic blockmodels. In Advances in Neural Information Processing Systems (pp. 33-40).

Application of MMSB to Sampson's Monastery

Estimated Mixed membership vectors

(posterior mean)



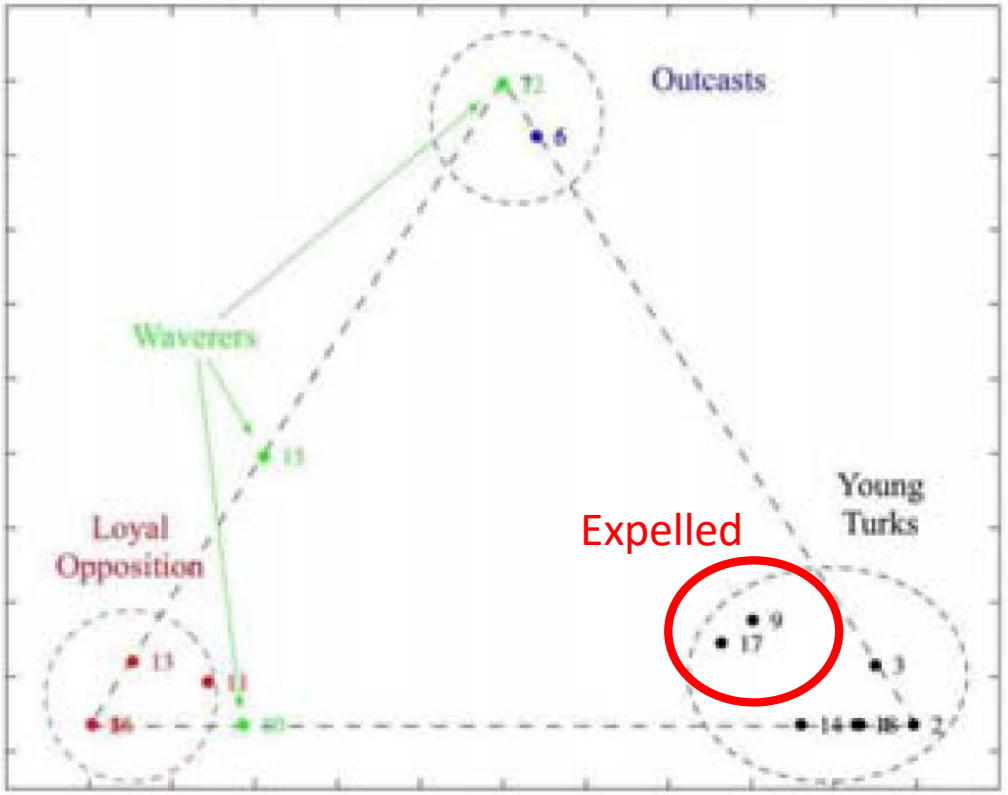
1	Ambrose
2	Boniface
3	Mark
4	Winfrid
5	Elias
6	Basil
7	Simplicius
8	Berthold
9	John Bosco
10	Victor
11	Bonaventure
12	Amand
13	Louis
14	Albert
15	Ramuald
16	Peter
17	Gregory
18	Hugh

Airoldi, E. M., Blei, D. M., Fienberg, S. E., & Xing, E. P. (2009). Mixed membership stochastic blockmodels. In Advances in Neural Information Processing Systems (pp. 33-40).

Application of MMSB to Sampson's Monastery

Estimated Mixed membership vectors

(posterior mean)



1 Ambrose
2 Boniface
3 Mark
4 Winfrid
5 Elias
6 Basil
7 Simplicius
8 Berthold
9 John Bosco
10 Victor
11 Bonaventure
12 Amand
13 Louis
14 Albert
15 Ramuald
16 Peter
17 Gregory
18 Hugh

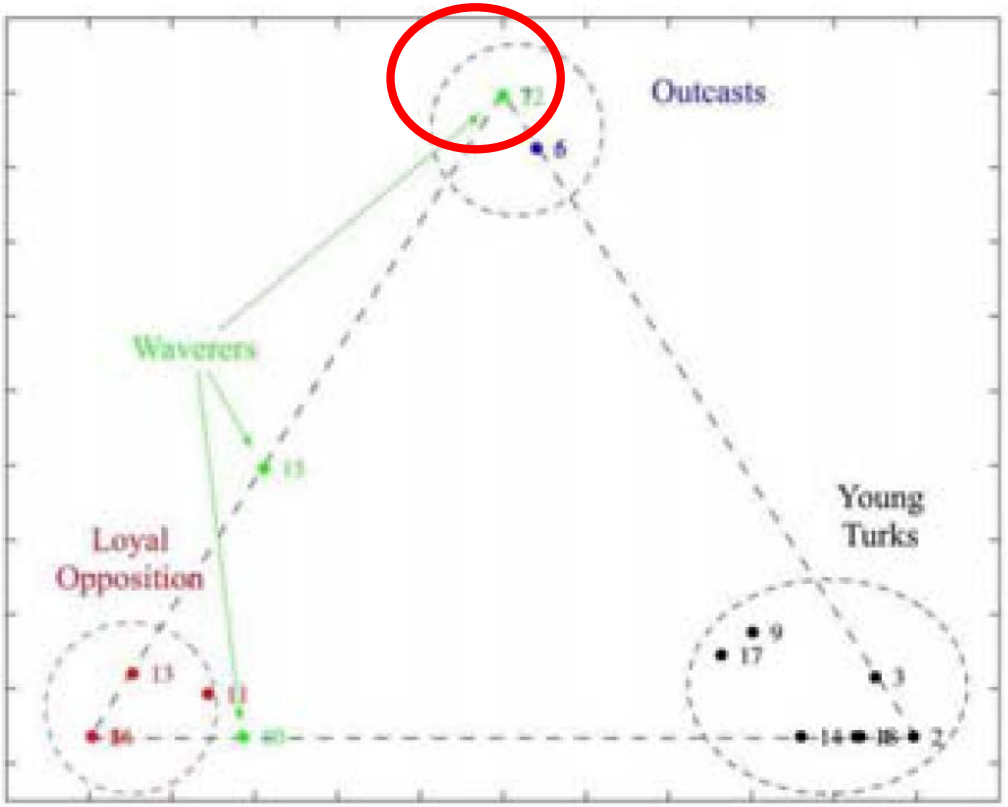
Airoldi, E. M., Blei, D. M., Fienberg, S. E., & Xing, E. P. (2009). Mixed membership stochastic blockmodels. In Advances in Neural Information Processing Systems (pp. 33-40).

Application of MMSB to Sampson's Monastery

Wavering not captured

Estimated Mixed membership vectors

(posterior mean)

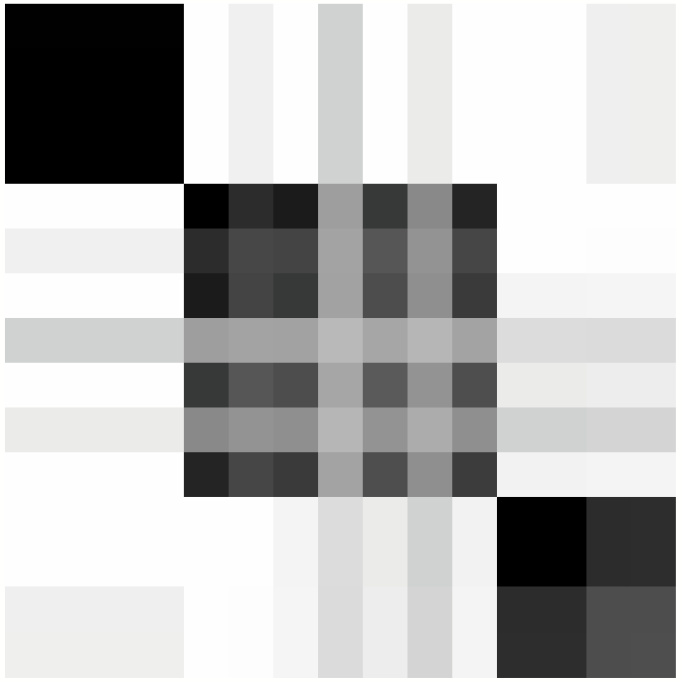


1 Ambrose
2 Boniface
3 Mark
4 Winfrid
5 Elias
6 Basil
7 Simplicius
8 Berthold
9 John Bosco
10 Victor
11 Bonaventure
12 Amand
13 Louis
14 Albert
15 Ramuald
16 Peter
17 Gregory
18 Hugh

Airoldi, E. M., Blei, D. M., Fienberg, S. E., & Xing, E. P. (2009). Mixed membership stochastic blockmodels. In Advances in Neural Information Processing Systems (pp. 33-40).

Application of MMSB to Sampson's Monastery

Original network
(whom do you like?)



Summary of network (use π's)

Airoldi, E. M., Blei, D. M., Fienberg, S. E., & Xing, E. P. (2009). Mixed membership stochastic blockmodels. In Advances in Neural Information Processing Systems (pp. 33-40).

Application of MMSB to Sampson's Monastery

Original network
(whom do you like?)

Denoise network (use z's)

Airoldi, E. M., Blei, D. M., Fienberg, S. E., & Xing, E. P. (2009). Mixed membership stochastic blockmodels. In Advances in Neural Information Processing Systems (pp. 33-40).

Scaling up Bayesian inference to large networks

- Two key strategies: **parallel/distributed**, and **stochastic algorithms**
- **Parallel/distributed algorithms**
  - Compute VB or MCMC updates in parallel
  - Communication overhead may be lower for MCMC
  - Not well understood for MCMC, but works in practice

- Stochastic algorithms
  - Stochastic variational inference
    - estimate updates based on subsamples. MMSB: *Gopalan et al. (2012)*
  - A related subsampling trick for MCMC in latent space models *(Raftery et al., 2012)*
  - Other general stochastic MCMC algorithms:
    - Stochastic gradient Langevin dynamics (Welling and Teh, 2011), Austerity MCMC (Korattika et al., 2014)

Evaluation of unsupervised models

- **Quantitative** evaluation
  - Measurable, quantifiable performance metrics

- **Qualitative** evaluation
  - Exploratory data analysis (EDA) using the model
  - Human evaluation, user studies,…

Evaluation of unsupervised models

- **Intrinsic** evaluation
  - Measure inherently good properties of the model
    - Fit to the data (e.g. link prediction), interpretability,…
- **Extrinsic** evaluation
  - Study usefulness of model for external tasks
    - Classification, retrieval, part of speech tagging,…

Extrinsic evaluation: What will you use your model for?

- If you have a **downstream task** in mind, you should probably evaluate based on it!

- Even if you don't, you could contrive one for evaluation purposes

- E.g. use latent representations for:
  - Classification, regression, retrieval, ranking…

Posterior predictive checks

- Sampling data from the posterior predictive distribution allows us to “look into the mind of the model” – G. Hinton

“This use of the word *mind* is not intended to be metaphorical. We believe that a mental state is the state of a hypothetical, external world in which a high-level internal representation would constitute veridical perception. That hypothetical world is what the figure shows.” **Geoff Hinton et al. (2006), A Fast Learning Algorithm for Deep Belief Nets.**

Posterior predictive checks

- Does data drawn from the model differ from the observed data, in ways that we care about?

- PPC:
  - Define a discrepancy function (a.k.a. test statistic) T(**X**).
    - Like a test statistic for a p-value. How extreme is my data set?
  - Simulate new data $\mathbf{X}^{(rep)}$ from the posterior predictive
    - Use MCMC to sample parameters from posterior, then simulate data
  - Compute T($\mathbf{X}^{(rep)}$) and T(**X**), compare. Repeat, to estimate:

$$\text{PPC} = P(T(\mathbf{X}^{(rep)}) > T(\mathbf{X})|\mathbf{X})$$