

# Birds of a Feather or Friend of a Friend? Social Network Analysis Using ERGMs\*

\* Exponential-family Random Graph Models

David Hunter

Department of Statistics  
Penn State University

BERD Seminar, March 30, 2015

# What is a network?

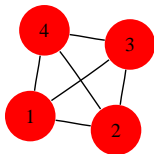
A network is a (mathematical) graph:

A set  $V$  and a set  $E \subset V \times V$

# What is a network?

A network is a (mathematical) graph:

A set  $V$  and a set  $E \subset V \times V$  (Yuck!)



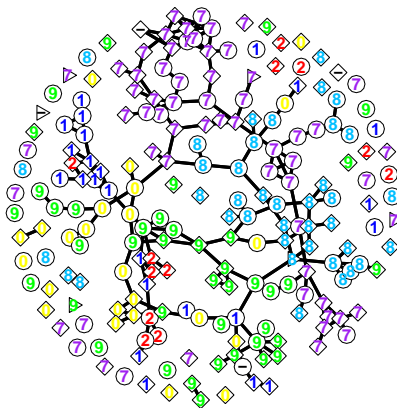
## “Nicer” definition

A representation of "relational data" in the form of a set of nodes along with a set of edges connecting some pairs of nodes.

- Example: Edges might represent contacts between individuals that could result in disease transmission.
- Usually, we also have some information on nodes (e.g., sex, race, age) or edges.

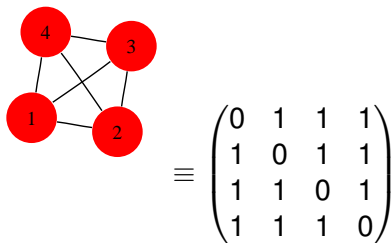
# Example Network: High School Friendship Data

School 10: 205 Students



- An edge indicates a mutual friendship.
- Colored labels give grade level, 7 through 12.
- Circles = female, squares = male, triangles = unknown.

# Notation: (Possibly symmetric) matrix $y$ of 0's and 1's



So that for all  $i$  and  $j$  from 1 to (in this case) 4,

$$y_{ij} = \begin{cases} 0 & \text{if there is no edge between } i \text{ and } j \\ 1 & \text{if there is an edge between } i \text{ and } j. \end{cases}$$

# Parameters vs. Statistics

N.B.: “Random variable”  $\equiv$  “statistic”

- In probability. . .  
The *parameters* are fixed and known, and they control the behavior of *random variables* via a *model*.

# Parameters vs. Statistics

N.B.: “Random variable”  $\equiv$  “statistic”

- In probability. . .  
The *parameters* are fixed and known, and they control the behavior of *random variables* via a *model*.
- In statistics (“probability in reverse”). . .  
The *random variables* (data) are fixed and known, and they are used to estimate the *parameters* that gave rise to them via a *model*.



# Parameters vs. Statistics

## Example

Model: Suppose we play 100 independent win-or-lose games, each with win probability  $\theta$ .

Then  $\theta$  is the parameter, and the number of wins is a statistic.



# Parameters vs. Statistics

## Example

Model: Suppose we play 100 independent win-or-lose games, each with win probability  $\theta$ .

Then  $\theta$  is the parameter, and the number of wins is a statistic.

- In probability...  
Fix  $\theta = 244/495$ ; go to the casino and see how you do!

# Parameters vs. Statistics

## Example

Model: Suppose we play 100 independent win-or-lose games, each with win probability  $\theta$ .

Then  $\theta$  is the parameter, and the number of wins is a statistic.

- In probability. . .  
Fix  $\theta = 244/495$ ; go to the casino and see how you do!
- In statistics (“probability in reverse”). . .  
I just got back from the casino, where I won 51 out of 100 games. What can I say about  $\theta$ ?



# What is a random network?

Ans: Any probability distribution on the set of all possible networks (say, on a fixed set of  $n$  nodes).

# What is a random network?

Ans: Any probability distribution on the set of all possible networks (say, on a fixed set of  $n$  nodes).

What does that mean?

Ans: We assign each possible network a probability:

# What is a random network?

Ans: Any probability distribution on the set of all possible networks (say, on a fixed set of  $n$  nodes).

What does that mean?

Ans: We assign each possible network a probability:

$$P \left[ Y = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix} \right] = \frac{1}{64},$$

# What is a random network?

Ans: Any probability distribution on the set of all possible networks (say, on a fixed set of  $n$  nodes).

What does that mean?

Ans: We assign each possible network a probability:

$$P \left[ Y = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix} \right] = \frac{1}{64}, P \left[ Y = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix} \right] = \frac{1}{64},$$

and so on.

# The Gilbert-Erdős-Rényi model for random networks

Fortunately, there are better ways than explicit enumeration to associate probabilities with networks!

Instead, let the probability of network  $Y$  be a known function (model) of some fixed parameters  $\theta$ .

# The Gilbert-Erdős-Rényi model for random networks

Fortunately, there are better ways than explicit enumeration to associate probabilities with networks!

Instead, let the probability of network  $Y$  be a known function (model) of some fixed parameters  $\theta$ .

## Example: The Gilbert-Erdős-Rényi model

Let  $\theta$  some fixed constant between 0 and 1. Let  $P(Y = y)$  be equal to  $\theta^{E(y)}(1 - \theta)^{\bar{E}(y)}$ , where  $E(y)$  is the number of edges in  $y$  and  $\bar{E}(y)$  is the number of non-edges in  $y$ .



# Uniformly random networks

Gilbert-Erdős-Rényi model (reminder):

Let  $\theta$  some fixed constant between 0 and 1. Let

$$P(Y = y) = \theta^{E(y)}(1 - \theta)^{\bar{E}(y)},$$

where  $E(y)$  is the number of edges in  $y$  and  $\bar{E}(y)$  is the number of non-edges in  $y$ .



# Uniformly random networks

## Gilbert-Erdős-Rényi model (reminder):

Let  $\theta$  some fixed constant between 0 and 1. Let

$$P(Y = y) = \theta^{E(y)}(1 - \theta)^{\bar{E}(y)},$$

where  $E(y)$  is the number of edges in  $y$  and  $\bar{E}(y)$  is the number of non-edges in  $y$ .

For  $\theta = 1/2$ , the probability is the same for every  $y$ :

$$P(Y = y) = \left(\frac{1}{2}\right)^{E(y) + \bar{E}(y)}$$

This is the *uniform* distribution. People often say “random networks” when they mean “uniform random networks”.

# Exponential family, or p-star, models

## Exponential-Family Random Graph Model (ERGM)

$$P_{\theta}(Y = y) \propto \exp\{\theta^t g(y)\}$$

or

$$P_{\theta}(Y = y) = \frac{\exp\{\theta^t g(y)\}}{\kappa(\theta)},$$

where

- $Y$  is a random network on  $n$  nodes (a matrix of 0's and 1's)
- $\theta$  is a vector of parameters
- $g(y)$  is a known vector of network statistics on  $y$
- $\kappa(\theta)$  makes all the probabilities sum to 1

# Exponential family, or p-star, models

## Exponential-Family Random Graph Model (ERGM)

$$P_{\theta}(Y = y) \propto \exp\{\theta^t g(y)\}$$

or

$$P_{\theta}(Y = y) = \frac{\exp\{\theta^t g(y)\}}{\kappa(\theta)},$$

where

- $Y$  is a random network on  $n$  nodes (a matrix of 0's and 1's)
- $\theta$  is a vector of parameters
- $g(y)$  is a known vector of network statistics on  $y$
- $\kappa(\theta)$  makes all the probabilities sum to 1

*Example: If  $g(y)$  is univariate and equal to  $E(y)$ , we get G-E-R!*

PENNSTATE



# Outline



# Interplay of network parameters and statistics

model class

$$\theta \rightarrow P(y)$$

$$\exp\{\theta^t g(y)\}$$

# Interplay of network parameters and statistics

model class

$$\theta \rightarrow P(y)$$

$$\exp\{\theta^t g(y)\}$$

$$\begin{array}{c} \longrightarrow \\ \uparrow \\ \text{data: } y^{\text{obs}} \end{array}$$



# Interplay of network parameters and statistics

model class

$$\theta \rightarrow P(y)$$

$$\exp\{\theta^t g(y)\}$$

parameter

estimate

$$\hat{\theta}$$

$\rightarrow$   
 $\uparrow$   
data:  $y^{\text{obs}}$





# Interplay of network parameters and statistics

model class  
 $\theta \rightarrow P(y)$   
 $\exp\{\theta^t g(y)\}$

$\rightarrow$   
 $\uparrow$   
data:  $y^{\text{obs}}$

parameter  
estimate  
 $\hat{\theta}$

$\rightarrow$

Fitted  
model  
 $\exp\{\hat{\theta}^t g(y)\}$



# Interplay of network parameters and statistics

model class

$$\theta \rightarrow P(y)$$

$$\exp\{\theta^t g(y)\}$$

→  
↑  
data:  $y^{\text{obs}}$



parameter  
estimate

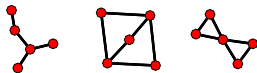
$$\hat{\theta}$$

→

Fitted  
model

$$\exp\{\hat{\theta}^t g(y)\}$$

↓  
Randomly generated  
networks  $\tilde{Y}_1, \tilde{Y}_2, \dots$



# Interplay of network parameters and statistics

model class  
 $\theta \rightarrow P(y)$   
 $\exp\{\theta^t g(y)\}$

$\rightarrow$   
 $\uparrow$   
data:  $y^{\text{obs}}$



parameter  
estimate  
 $\hat{\theta}$

$\rightarrow$

Fitted  
model  
 $\exp\{\hat{\theta}^t g(y)\}$

$\downarrow$   
Randomly generated  
networks  $\tilde{Y}_1, \tilde{Y}_2, \dots$



- Question: How does  $y^{\text{obs}}$  “look” as a representative of the sample  $\tilde{Y}_1, \tilde{Y}_2, \dots$ ? (Gives us evidence of *model fit*.)
- But first: How do we estimate  $\hat{\theta}$ ?

# The goal of estimation

## Exponential-family Random Graph Model (ERGM)

$$P_{\theta}(Y = y) = \frac{\exp\{\theta^t g(y)\}}{\kappa(\theta)}$$

- When  $\theta$  is known, this is a probability model describing the random behavior of  $Y$ .

# The goal of estimation

## Exponential-family Random Graph Model (ERGM)

$$P_{\theta}(Y = y) = \frac{\exp\{\theta^t g(y)\}}{\kappa(\theta)}$$

- When  $\theta$  is known, this is a probability model describing the random behavior of  $Y$ .
- Statistical estimation is “probability in reverse”: We don’t know  $\theta$  but instead we observe  $Y = y^{\text{obs}}$ .

### Goal:

Use observed data (a network  $y^{\text{obs}}$ ) to determine a “best” model from the given ERGM class — i.e., a “best”  $\theta$  — in some sense.

We will focus on maximum likelihood estimation (MLE).

# The loglikelihood function

The model class:

$$P_{\theta}(Y = y) = \frac{\exp\{\theta^t g(y)\}}{\kappa(\theta)}, \text{ where } \kappa(\theta) = \sum_{\text{all possible graphs } z} \exp\{\theta^t g(z)\}$$

- The loglikelihood is just  $\log P_{\theta}(Y = y^{\text{obs}})$ , viewed as a function of  $\theta$ :

$$\ell(\theta) = \theta^t g(y^{\text{obs}}) - \log \kappa(\theta)$$

# The loglikelihood function

The model class:

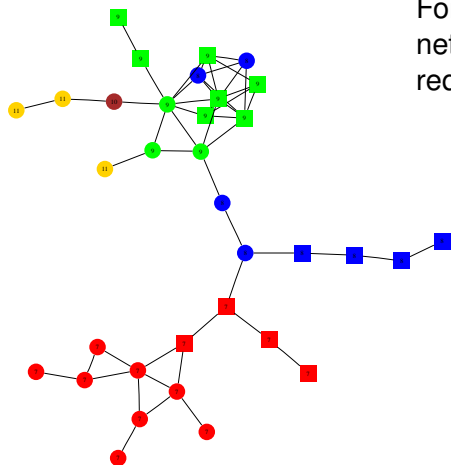
$$P_{\theta}(Y = y) = \frac{\exp\{\theta^t g(y)\}}{\kappa(\theta)}, \text{ where } \kappa(\theta) = \sum_{\text{all possible graphs } z} \exp\{\theta^t g(z)\}$$

- The loglikelihood is just  $\log P_{\theta}(Y = y^{\text{obs}})$ , viewed as a function of  $\theta$ :

$$\ell(\theta) = \theta^t g(y^{\text{obs}}) - \log \kappa(\theta)$$

- We want the  $\theta$  that maximizes  $\ell(\theta)$ , denoted here by  $\hat{\theta}$ . This is the **maximum likelihood estimator**.

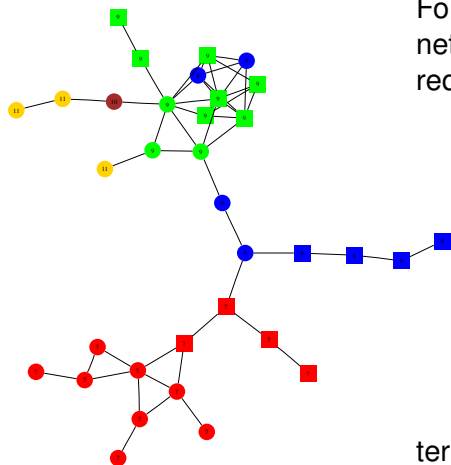
# Finding the MLE is sometimes very hard. . .



For this undirected, 34-node network, computing  $\ell(\theta)$  directly requires summation of



# Finding the MLE is sometimes very hard. . .



For this undirected, 34-node network, computing  $\ell(\theta)$  directly requires summation of

7,547,924,849,643,082,704,483,  
109,161,976,537,781,833,842,  
440,832,880,856,752,412,600,  
491,248,324,784,297,704,172,  
253,450,355,317,535,082,936,  
750,061,527,689,799,541,169,  
259,849,585,265,122,868,502,  
865,392,087,298,790,653,952

terms.

## Logistic regression:

We assume that

$$P(Y_{21} = 1) = f(\theta^t \mathbf{x}_{21})$$

$$P(Y_{31} = 1) = f(\theta^t \mathbf{x}_{31})$$

$$P(Y_{32} = 1) = f(\theta^t \mathbf{x}_{32})$$

⋮

where  $f(t) = \frac{e^t}{1+e^t}$ .

*We must also assume that the  $Y_{ij}$  are independent!*

If our choice of  $g(y)$  implies an  $\mathbf{x}$  making this independence assumption valid, then ERGM is merely logistic regression.

Then, finding the MLE  $\hat{\theta}$  is (relatively) simple.



*Far better an **approximate answer to the 'right' question**, which is often vague, than an **'exact' answer to the wrong question**, which can always be made precise.*

— John W. Tukey

- **MLE (maximum likelihood estimation)**: Well-established method but very hard because the normalizing constant  $\kappa(\theta)$  is difficult to evaluate, so we approximate it instead.
- **MPLE (maximum pseudo-likelihood estimation)**: Easy to do using logistic regression, but based on an independence assumption that is often not justified.

Several authors, notably van Duijn et al. (2007), argue forcefully against the use of MPLE (except when MLE=MPLE!).

# The R project and statnet (cf. J. Stat Soft, v. 24)

## R: An open-source statistical package



R version 2.7.0 (2008-04-22)  
Copyright (C) 2008 The R Foundation for Statistical Computing  
ISBN 3-900051-07-0

R est un logiciel libre livré sans AUCUNE GARANTIE.  
Vous pouvez le redistribuer sous certaines conditions.  
Tapez 'license()' ou 'licence()' pour plus de détails.

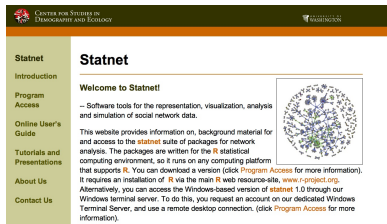
R est un projet collaboratif avec de nombreux contributeurs.  
Tapez 'contributors()' pour plus d'information et  
'citation()' pour la façon de le citer dans les publications.

Tapez 'demo()' pour des démonstrations, 'help()' pour l'aide  
en ligne ou 'help.start()' pour obtenir l'aide au format HTML.  
Tapez 'q()' pour quitter R.

```
> library(network) ; library(ergm)
```

## statnet: An R package for network analysis

See volume 24 of *Journal of Statistical Software*

A screenshot of the Statnet website homepage. The header includes the logo for the Center for Studies in Demography and Ecology and the University of Pennsylvania. The main content area is titled "Statnet" and includes a "Welcome to Statnet!" message. The message describes the software tools for network data analysis and provides information on how to access the software, including downloading a version or using a Windows terminal server. A circular network graph is displayed on the right side of the page. The left sidebar contains navigation links: Introduction, Program Access, Online User's Guide, Tutorials and Presentations, About Us, and Contact Us.

[www.r-project.org](http://www.r-project.org)

[www.statnet.org](http://www.statnet.org)



# Outline

# Goodness of fit intuition

ERGM  
class  
 $\exp\{\theta^t g(y)\}$

# Goodness of fit intuition

ERGM  
class

$$\exp\{\theta^t g(y)\}$$

→

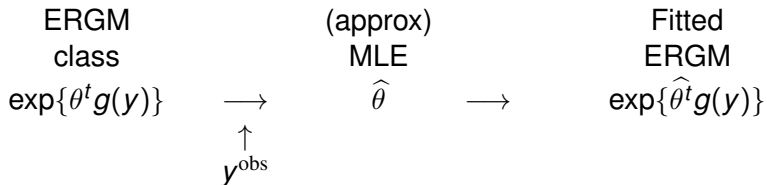
↑  
 $y^{\text{obs}}$







# Goodness of fit intuition



# Goodness of fit intuition

ERGM  
class  
 $\exp\{\theta^t g(y)\}$

$\longrightarrow$   
 $\uparrow$   
 $y^{\text{obs}}$



(approx)  
MLE  
 $\hat{\theta}$

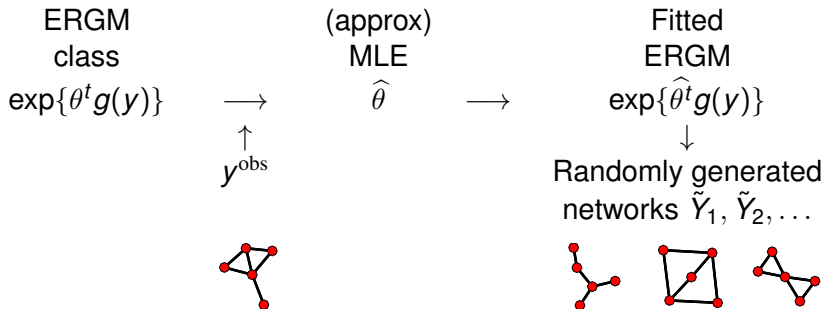
$\longrightarrow$

Fitted  
ERGM  
 $\exp\{\hat{\theta}^t g(y)\}$

$\downarrow$   
Randomly generated  
networks  $\tilde{Y}_1, \tilde{Y}_2, \dots$



# Goodness of fit intuition

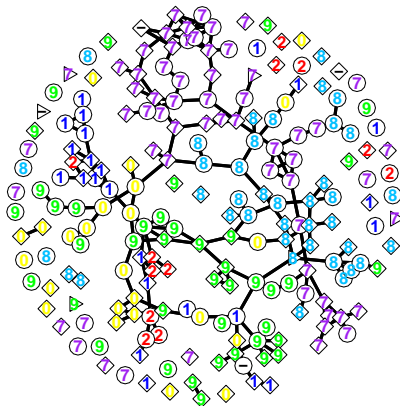


- Question: How does  $y^{\text{obs}}$  “look” as a representative of the sample  $\tilde{Y}_1, \tilde{Y}_2, \dots$ ?

# The eyeball test

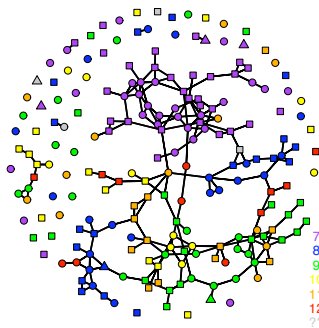
The data:

School 10: 205 Students



Simulated network,  
model A:

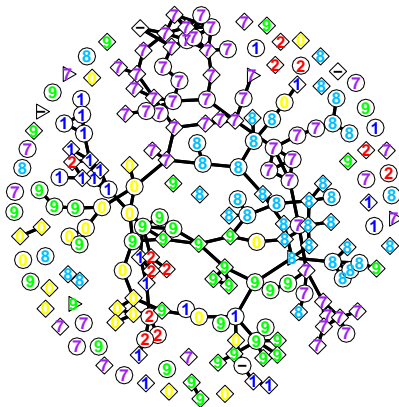
Simulated graph: By grade



# The eyeball test (cont'd)

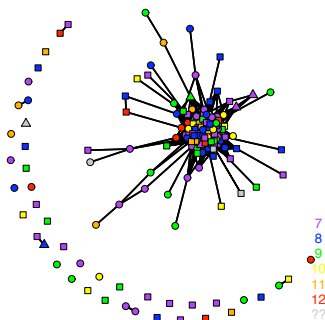
The data:

School 10: 205 Students



Simulated network,  
model B:

Simulated graph: By grade

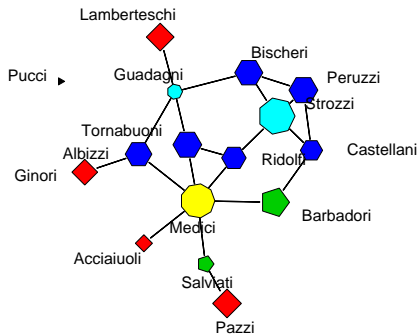


# The models

- Model A:  $g(y)$  contains terms for
  - # of edges
  - Homophily effects of grade, sex, and race factors
  - Main effects of grade, sex, and race factors
  - $\sum_i (.632)^i EP_i$ , where  $EP_i = \#$  edges with  $i$  shared partners
- Model B:  $g(y)$  contains terms for
  - # of edges
  - # of neighbors of the same sex (homophily effect)
  - # of 2-stars
  - # of triangles

(Note: It was necessary to use MPLE to fit Model B)

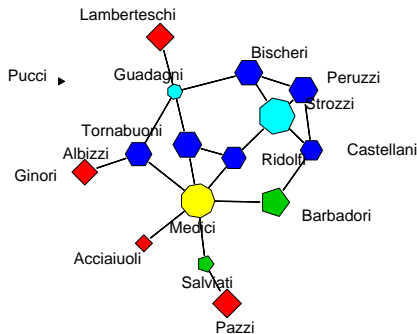
## A well-known example:



## Florentine marriage data

- Edge indicates marriage tie between families
- Sides=degree + 3
- Color=degree
- Size=log(wealth)

## A well-known example:



### Florentine marriage data

- Edge indicates marriage tie between families
- Sides=degree + 3
- Color=degree
- Size=log(wealth)

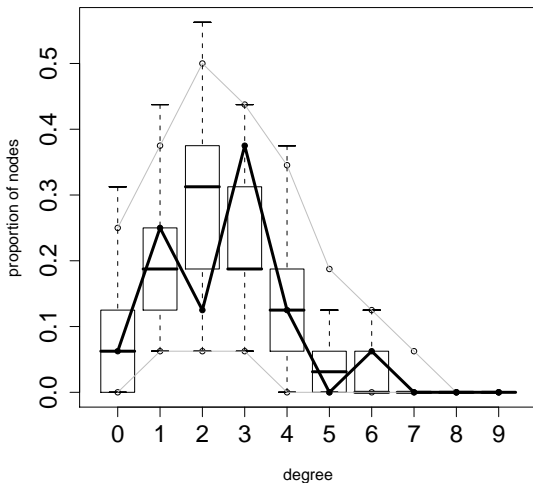
```
modell <- ergm(flomarriage ~ edges + kstar(2))
```



# Graphical GOF check: degree distribution

```
modell <- ergm(flomarriage ~ edges + kstar(2))  
gof(modell)
```

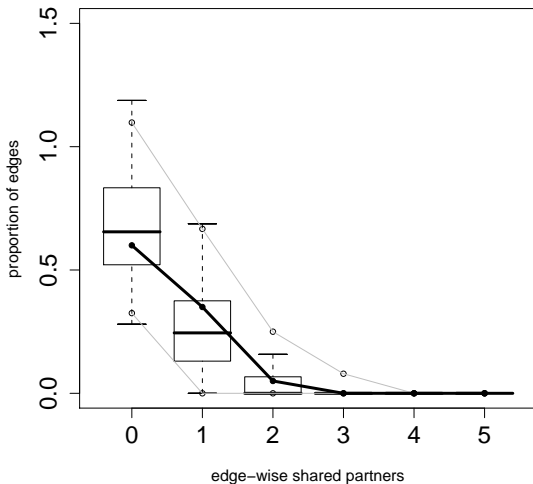
## Goodness-of-fit diagnostics



# Graphical GOF: edgewise shared partner distribution

```
modell1 <- ergm(flomarriage ~ edges + kstar(2))  
gof(modell1)
```

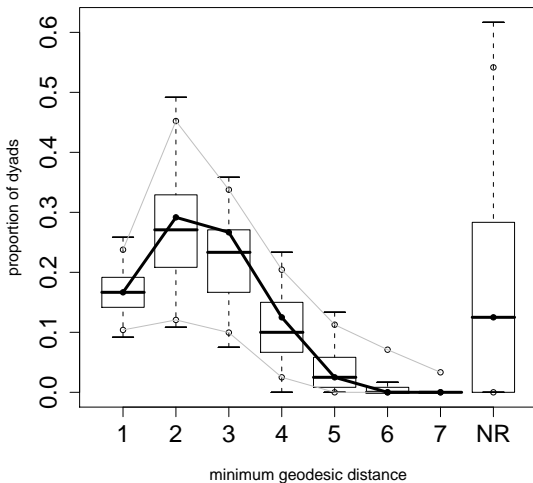
## Goodness-of-fit diagnostics



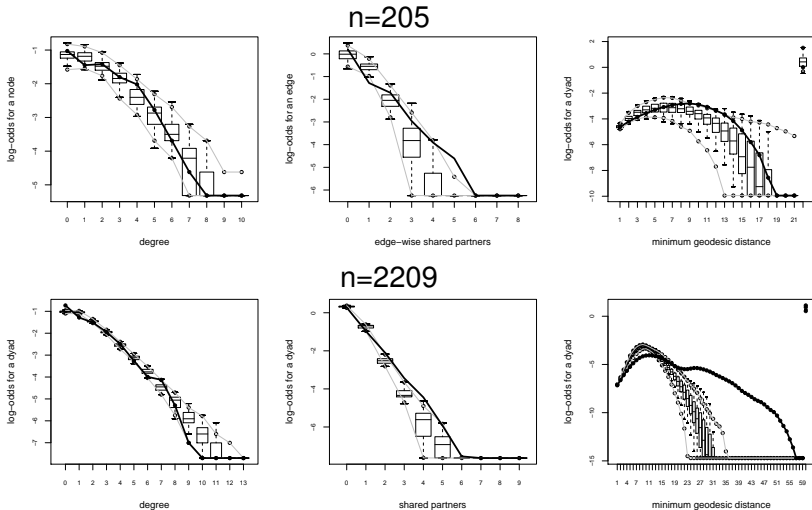
# Graphical GOF check: geodesic distance distribution

```
modell1 <- ergm(flomarriage ~ edges + kstar(2))  
gof(modell1)
```

## Goodness-of-fit diagnostics



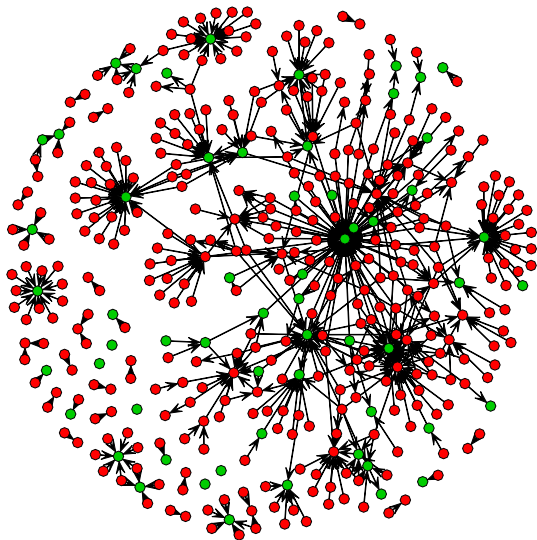
# GOF check: Less trivial examples



Taken from Hunter, Goodreau, Handcock (2008), *JASA*

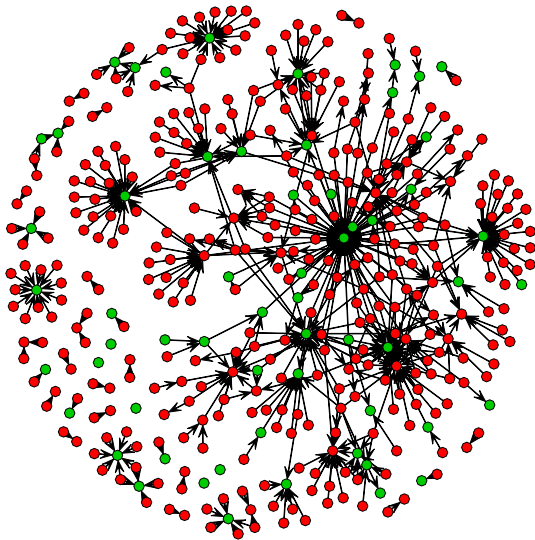
# Outline

# Example Bionet: E. Coli (Salgado et al 2001)



- A node is an operon
- Edge  $A \rightarrow B$  means  $A$  encodes the transcription factor that regulates  $B$ .
- Green indicates self-regulation

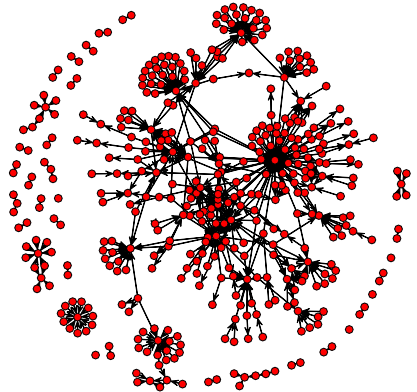
# Example Bionet: E. Coli (Salgado et al 2001)



- A node is an operon
- Edge  $A \rightarrow B$  means  $A$  encodes the transcription factor that regulates  $B$ .
- Green indicates self-regulation
- Goal: “Regression” with a network response.
- i.e., what does this network tell us about the processes that might have formed it?

# The E. Coli network as a network-class object in R

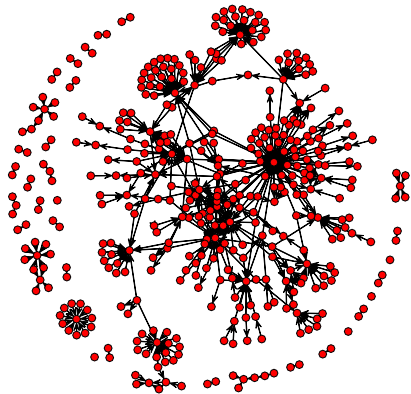
```
> plot(ecoli1)
```





# The E. Coli network as a network-class object in R

```
> plot(ecoli1)  
> class(ecoli1)  
[1] "network"
```

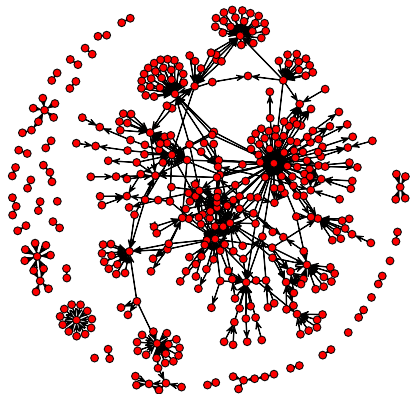


# The E. Coli network as a network-class object in R

```
> plot(ecoli1)
> class(ecoli1)
[1] "network"
> summary(ecoli1)
Network attributes:
vertices: 423
  directed : TRUE
  hyper    : FALSE
  loops    : FALSE
  multiple : FALSE
  bipartite : FALSE
total edges= 519
density = 0.002907465
```

Vertex attributes:

```
self:
FALSE TRUE
 364   59
```



# The E. Coli network as a network-class object in R

```
> plot(ecoli1)
```

```
> class(ecoli1)
```

```
[1] "network"
```

```
> summary(ecoli1)
```

```
Network attributes:
```

```
vertices: 423
```

```
directed : TRUE
```

```
hyper : FALSE
```

```
loops : FALSE
```

```
multiple : FALSE
```

```
bipartite : FALSE
```

```
total edges= 519
```

```
density = 0.002907465
```

```
Vertex attributes:
```

```
self:
```

```
FALSE TRUE
```

```
364 59
```

```
> summary(ecoli1 ~ edges + odegree(0:5))
```

```
edges odegree0 odegree1 odegree2 odegree3 odegree4 odegree5
```

```
519
```

```
81
```

```
229
```

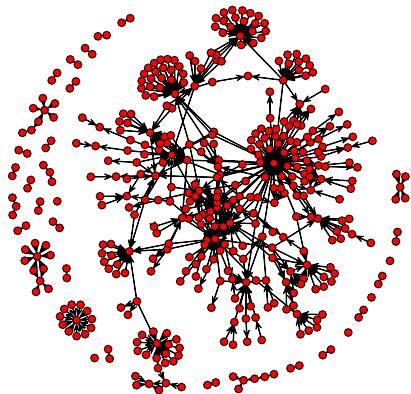
```
71
```

```
25
```

```
13
```

```
3
```

```
PENNSYLVANIA  
STATE  
UNIVERSITY
```



# A simple ERGM using the `ergm` package in R

The `ecoli1` network has 519 edges, 423 nodes (thus  $423 \times 422$  possible directed edges). Take  $g(y) = \#$  edges in  $y$ :

$$\begin{aligned} P(Y = y) &\propto \exp\{\theta \times \# \text{ edges in } y\} \\ &= \left( \frac{p}{1-p} \right)^{(\# \text{ edges in } y)} \end{aligned}$$

# A simple ERGM using the ergm package in R

The `ecoli1` network has 519 edges, 423 nodes (thus  $423 \times 422$  possible directed edges). Take  $g(y) = \#$  edges in  $y$ :

$$\begin{aligned} P(Y = y) &\propto \exp\{\theta \times \# \text{ edges in } y\} \\ &= \left(\frac{p}{1-p}\right)^{(\# \text{ edges in } y)} \end{aligned}$$

```
> modell <- ergm(ecoli1 ~ edges)
> modell$coef
  edges
-5.837562
```

This means the MLE is  $\hat{\theta} = -5.84$ . It is exact because  $g(y)$  in this example ensures independent  $Y_{ij}$ , so MPLE=MLE!

# A simple ERGM using the ergm package in R

The `ecoli1` network has 519 edges, 423 nodes (thus  $423 \times 422$  possible directed edges). Take  $g(y) = \#$  edges in  $y$ :

$$\begin{aligned} P(Y = y) &\propto \exp\{\theta \times \# \text{ edges in } y\} \\ &= \left(\frac{p}{1-p}\right)^{(\# \text{ edges in } y)} \end{aligned}$$

```
> modell <- ergm(ecoli1 ~ edges)
> modell$coef
  edges
-5.837562
```

This means the MLE is  $\hat{\theta} = -5.84$ . It is exact because  $g(y)$  in this example ensures independent  $Y_{ij}$ , so MPLE=MLE!

Observe:

```
> log(519 / (423 * 422 - 519))
[1] -5.837562
```

# Another simple ERGM

Let's consider an edge to have one of four categories, depending on whether or not each of its endpoints is self-regulating:

```
> model2 <- ergm(ecoli1 ~ nodemix("self"))  
> summary(model2)
```

Pseudolikelihood Results:

	Estimate	Std. Error	MCMC s.e.	p-value
<code>mix.self.FALSE.FALSE</code>	-6.61974	0.07543	NA	<1e-04
<code>mix.self.TRUE.FALSE</code>	-7.48923	0.28874	NA	<1e-04
<code>mix.self.FALSE.TRUE</code>	-4.22686	0.05730	NA	<1e-04
<code>mix.self.TRUE.TRUE</code>	-5.04049	0.21389	NA	<1e-04

This choice of  $g(y)$  also ensures independent  $Y_{ij}$ , so the MLE and the MPLE are the same. Thus, the MLE is found analytically and there is no MCMC standard error.

# Yet another ERGM for the E. Coli network

Saul and Filkov (2007) fit a model with these terms:

<b>Term(s)</b>	<b>Description:</b>
Edges	Number of Edges
2-Deg, ..., 5-Deg	Nodes with degree 2, ..., 5
2-Star	Pairs of edges with one end in common
GWDeg	Single statistic: Weighted sum of 1-Deg, ..., $(n - 1)$ -Deg with weights tending to 1 at a geometric rate



# Yet another ERGM for the E. Coli network

Saul and Filkov (2007) fit a model with these terms:

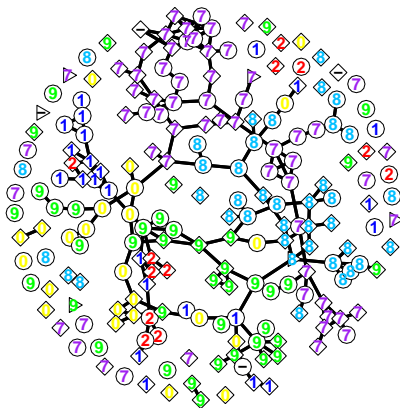
Term(s)	Description:
Edges	Number of Edges
2-Deg, ..., 5-Deg	Nodes with degree 2, ..., 5
2-Star	Pairs of edges with one end in common
GWDeg	Single statistic: Weighted sum of 1-Deg, ..., $(n - 1)$ -Deg with weights tending to 1 at a geometric rate

```
> model3 <- ergm(ecoli2 ~ edges + degree(2:5) + kstar(2) +  
+ gwdegree(3.75, fixed=TRUE), MPLEonly=TRUE)
```

NB: The `ecoli2` network is undirected, probably not a good idea in this situation.

# High School Friendship Data Revisited

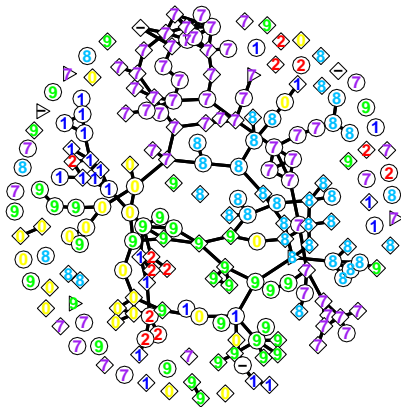
School 10: 205 Students



- In this network, we see a lot of clustering (essentially, triangles).
- What causes the clustering?

# Birds of a Feather or Friend of a Friend?

School 10: 205 Students



- BOAF: Selective mixing, a.k.a. assortative mixing
- FOAF: Triad closure
- NB: BOAF alone gives independent node pairs but FOAF leads to dependence.

Both forces are present; see Goodreau et al. (2008) for fuller analysis.

# Cited References

- Goodreau SM, Kitts JA, Morris M (2008, *Demography*) Birds of a Feather or Friend of a Friend? Using Exponential Random Graph Models to Investigate Adolescent Social Networks
- Hunter DR, Goodreau SM, and Handcock MS (2008, *J. Am. Stat. Assoc.*) Goodness of fit for social network models.
- Saul ZM and Filkov V (2007, *Bioinformatics*), Exploring biological network structure using exponential random graph models.
- Salgado H et al. (2001, *Nucleic Acids Res.*), Regulondb (version 3.2): Transcriptional regulation and operon organization in Escherichia Coli k-12.
- van Duijn MAJ, Gile K, and Handcock MS (2007, CSSS working paper<sup>1</sup> no. 74), Comparison of maximum pseudo likelihood and maximum likelihood estimation of exponential family random graph models.

---

<sup>1</sup><http://www.csss.washington.edu/Papers>

