Covariate effects in ERGMs

567 Statistical analysis of social networks

Peter Hoff

Statistics, University of Washington
Dutch college data

Friendship ties among 32 college students enrolled in a particular program.

- Relations on a 6 point scale, from "dislike" to "best friends";
- Relations measured at seven time points;
- Sex, smoking status and subprogram category also available.

Questions:
- What are the effects of sex, smoking status and subgroup on tie formation?
- Is there substantial in and outdegree heterogeneity, or reciprocity?
- How does the network evolve over time?
Dutch college data

Friendship ties among 32 college students enrolled in a particular program.

- Relations on a 6 point scale, from "dislike" to "best friends";
- Relations measured at seven time points;
- Sex, smoking status and subprogram category also available.

Questions:

- What are the effects of sex, smoking status and subgroup on tie formation?
- Is their substantial in and outdegree heterogeneity, or reciprocity?
- How does the network evolve over time?
Dutch college data

Friendship ties among 32 college students enrolled in a particular program.

- Relations on a 6 point scale, from "dislike" to "best friends";
- Relations measured at seven time points;
- Sex, smoking status and subprogram category also available.

Questions:

- What are the effects of sex, smoking status and subgroup on tie formation?
- Is there substantial in and outdegree heterogeneity, or reciprocity?
- How does the network evolve over time?
Dutch college data

Friendship ties among 32 college students enrolled in a particular program.

- Relations on a 6 point scale, from "dislike" to "best friends"
- Relations measured at seven time points
- Sex, smoking status and subprogram category also available

Questions:

- What are the effects of sex, smoking status and subgroup on tie formation?
- Is there substantial in and outdegree heterogeneity, or reciprocity?
- How does the network evolve over time?
Dutch college data

Friendship ties among 32 college students enrolled in a particular program.

• Relations on a 6 point scale, from "dislike" to "best friends";
• Relations measured at seven time points;
• Sex, smoking status and subprogram category also available.

Questions:

• What are the effects of sex, smoking status and subgroup on tie formation?
• Is their substantial in and outdegree heterogeneity, or reciprocity?
• How does the network evolve over time?
Dutch college data

Friendship ties among 32 college students enrolled in a particular program.

- Relations on a 6 point scale, from "dislike" to "best friends";
- Relations measured at seven time points;
- Sex, smoking status and subprogram category also available.

Questions:

- What are the effects of sex, smoking status and subgroup on tie formation?
- Is there substantial in and outdegree heterogeneity, or reciprocity?
- How does the network evolve over time?
Dutch college data

Friendship ties among 32 college students enrolled in a particular program.

- Relations on a 6 point scale, from "dislike" to "best friends";
- Relations measured at seven time points;
- Sex, smoking status and subprogram category also available.

Questions:

- What are the effects of sex, smoking status and subgroup on tie formation?
- Is their substantial in and outdegree heterogeneity, or reciprocity?
- How does the network evolve over time?
Dutch college data

Friendship ties among 32 college students enrolled in a particular program.

- Relations on a 6 point scale, from "dislike" to "best friends";
- Relations measured at seven time points;
- Sex, smoking status and subprogram category also available.

Questions:

- What are the effects of sex, smoking status and subgroup on tie formation?
- Is there substantial in and outdegree heterogeneity, or reciprocity?
- How does the network evolve over time?
Dutch college data

Friendship ties among 32 college students enrolled in a particular program.

- Relations on a 6 point scale, from "dislike" to "best friends";
- Relations measured at seven time points;
- Sex, smoking status and subprogram category also available.

Questions:

- What are the effects of sex, smoking status and subgroup on tie formation?
- Is there substantial in and outdegree heterogeneity, or reciprocity?
- How does the network evolve over time?
Dutch college data

Friendship ties among 32 college students enrolled in a particular program.

- Relations on a 6 point scale, from "dislike" to "best friends";
- Relations measured at seven time points;
- Sex, smoking status and subprogram category also available.

Questions:

- What are the effects of sex, smoking status and subgroup on tie formation?
- Is there substantial in and outdegree heterogeneity, or reciprocity?
- How does the network evolve over time?
For now, we’ll analyze

- the indicator of a positive relation;
- the network at the final timepoint.
For now, we’ll analyze

- the indicator of a positive relation;
- the network at the final timepoint.
For now, we’ll analyze

- the indicator of a positive relation;
- the network at the final timepoint.
For now, we’ll analyze

- the indicator of a positive relation;
- the network at the final timepoint.
Preliminary analysis

\[
\text{mean}(Y_7, \text{na.rm=TRUE})
\]
## [1] 0.1693548

\[
\text{mean}(Y_7[ \text{X$male==1, X$male==1} ], \text{na.rm=TRUE})
\]
## [1] 0.3571429

\[
\text{mean}(Y_7[ \text{X$male==0, X$male==0} ], \text{na.rm=TRUE})
\]
## [1] 0.1956522

\[
\text{mean}(Y_7[ \text{X$smoke==1, X$smoke==1} ], \text{na.rm=TRUE})
\]
## [1] 0.2692308

\[
\text{mean}(Y_7[ \text{X$smoke==0, X$smoke==0} ], \text{na.rm=TRUE})
\]
## [1] 0.2017544

\[
\text{SP<}-\text{outer(} \text{X$prog, X$prog, "==")}
\]
\[
\text{mean}(Y_7[\text{SP}], \text{na.rm=TRUE})
\]
## [1] 0.2861111
Preliminary analysis

mean(outdegree[X$smoker==1]) - mean(outdegree[X$smoker==0])
## [1] 0.3562753

mean(indegree[X$smoker==1]) - mean(indegree[X$smoker==0])
## [1] 0.2267206
Degree heterogeneity and Reciprocity

```r
### degree analysis
sd(outdegree)
## [1] 4.662825
sd(indegree)
## [1] 2.514474
cor(outdegree, indegree)
## [1] 0.1705822

### dyad census
M <- sum(Y7*t(Y7), na.rm=TRUE) / 2
A <- sum(Y7, na.rm=TRUE) - 2*M
N <- choose(nrow(Y7), 2) - M - A
p11 <- 2*M / (2*M + A)
p10 <- A / (A + 2*N)
log( p11 * (1-p10) / ((1-p11) * p10) )
## [1] 2.250258
```
Preliminary analysis

Some preliminary findings:

- **Covariate effects:**
  - homophily by sex, smoking behavior and program;
  - smokers seem more outgoing and popular.

- **Network patterns:**
  - positive reciprocity;
  - outdegree variance is larger than indegree variance, and little correlation between the two.

To summarize covariate effects, some researchers employ “network regression:”

- convert sociomatrix and covariate matrices to vectors;
- perform ordinary regression (OLS, logistic regression, Poisson regression).

Such a procedure should not be called network regression:

- it is just regression;
- it ignores the network structure to the data.
Preliminary analysis

Some preliminary findings:

- Covariate effects:
  - homophily by sex, smoking behavior and program;
  - smokers seem more outgoing and popular.

- Network patterns:
  - positive reciprocity;
  - outdegree variance is larger than indegree variance, and little correlation between the two.

To summarize covariate effects, some researchers employ “network regression:”

- convert sociomatrix and covariate matrices to vectors;
- perform ordinary regression (OLS, logistic regression, Poisson regression).

Such a procedure should not be called network regression:

- it is just regression;
- it ignores the network structure to the data.
Some preliminary findings:

- **Covariate effects:**
  - homophily by sex, smoking behavior and program;
  - smokers seem more outgoing and popular.

- **Network patterns:**
  - positive reciprocity;
  - outdegree variance is larger than indegree variance, and little correlation between the two.

To summarize covariate effects, some researchers employ “network regression:”

- convert sociomatrix and covariate matrices to vectors;
- perform ordinary regression (OLS, logistic regression, Poisson regression).

Such a procedure should **not** be called network regression:

- it is just regression;
- it ignores the network structure to the data.
Preliminary analysis

Some preliminary findings:

- **Covariate effects:**
  - homophily by sex, smoking behavior and program;
  - smokers seem more outgoing and popular.

- **Network patterns:**
  - positive reciprocity;
  - outdegree variance is larger than indegree variance, and little correlation between the two.

To summarize covariate effects, some researchers employ “network regression:”

- convert sociomatrix and covariate matrices to vectors;
- perform ordinary regression (OLS, logistic regression, Poisson regression).

Such a procedure should **not** be called network regression:

- it is just regression;
- it ignores the network structure to the data.
Preliminary analysis

Some preliminary findings:

- **Covariate effects:**
  - homophily by sex, smoking behavior and program;
  - smokers seem more outgoing and popular.

- **Network patterns:**
  - positive reciprocity;
  - outdegree variance is larger than indegree variance, and little correlation between the two.

To summarize covariate effects, some researchers employ “network regression:”

- convert sociomatrix and covariate matrices to vectors;
- perform ordinary regression (OLS, logistic regression, Poisson regression).

Such a procedure should **not** be called network regression:

- it is just regression;
- it ignores the network structure to the data.
Preliminary analysis

Some preliminary findings:

- **Covariate effects:**
  - homophily by sex, smoking behavior and program;
  - smokers seem more outgoing and popular.

- **Network patterns:**
  - positive reciprocity;
  - outdegree variance is larger than indegree variance, and little correlation between the two.

To summarize covariate effects, some researchers employ “network regression:”

- convert sociomatrix and covariate matrices to vectors;
- perform ordinary regression (OLS, logistic regression, Poisson regression).

Such a procedure should **not** be called network regression:

- it is just regression;
- it ignores the network structure to the data.
Some preliminary findings:
  • Covariate effects:
    • homophily by sex, smoking behavior and program;
    • smokers seem more outgoing and popular.
  • Network patterns:
    • positive reciprocity;
    • outdegree variance is larger than indegree variance, and little correlation between the two.

To summarize covariate effects, some researchers employ “network regression:”
  • convert sociomatrix and covariate matrices to vectors;
  • perform ordinary regression (OLS, logistic regression, Poisson regression).

Such a procedure should not be called network regression:
  • it is just regression;
  • it ignores the network structure to the data.
Some preliminary findings:

- **Covariate effects:**
  - homophily by sex, smoking behavior and program;
  - smokers seem more outgoing and popular.

- **Network patterns:**
  - positive reciprocity;
  - outdegree variance is larger than indegree variance, and little correlation between the two.

To summarize covariate effects, some researchers employ “network regression:”

- convert sociomatrix and covariate matrices to vectors;
- perform ordinary regression (OLS, logistic regression, Poisson regression).

Such a procedure should **not** be called network regression:

- it is just regression;
- it ignores the network structure to the data.
Preliminary analysis

Some preliminary findings:

- Covariate effects:
  - homophily by sex, smoking behavior and program;
  - smokers seem more outgoing and popular.

- Network patterns:
  - positive reciprocity;
  - outdegree variance is larger than indegree variance, and little correlation between the two.

To summarize covariate effects, some researchers employ “network regression:”

- convert sociomatrix and covariate matrices to vectors;
- perform ordinary regression (OLS, logistic regression, Poisson regression).

Such a procedure should not be called network regression:

- it is just regression;
- it ignores the network structure to the data.
Some preliminary findings:

- **Covariate effects:**
  - homophily by sex, smoking behavior and program;
  - smokers seem more outgoing and popular.

- **Network patterns:**
  - positive reciprocity;
  - outdegree variance is larger than indegree variance, and little correlation between the two.

To summarize covariate effects, some researchers employ “network regression:”

- convert sociomatrix and covariate matrices to vectors;
- perform ordinary regression (OLS, logistic regression, Poisson regression).

Such a procedure should **not** be called network regression:

- it is just regression;
- it ignores the network structure to the data.
Preliminary analysis

Some preliminary findings:

- Covariate effects:
  - homophily by sex, smoking behavior and program;
  - smokers seem more outgoing and popular.

- Network patterns:
  - positive reciprocity;
  - outdegree variance is larger than indegree variance, and little correlation between the two.

To summarize covariate effects, some researchers employ “network regression:”

- convert sociomatrix and covariate matrices to vectors;
- perform ordinary regression (OLS, logistic regression, Poisson regression).

Such a procedure should not be called network regression:

- it is just regression;
- it ignores the network structure to the data.
Preliminary analysis

Some preliminary findings:

- Covariate effects:
  - homophily by sex, smoking behavior and program;
  - smokers seem more outgoing and popular.

- Network patterns:
  - positive reciprocity;
  - outdegree variance is larger than indegree variance, and little correlation between the two.

To summarize covariate effects, some researchers employ “network regression:”

- convert sociomatrix and covariate matrices to vectors;
- perform ordinary regression (OLS, logistic regression, Poisson regression).

Such a procedure should not be called network regression:

- it is just regression;
- it ignores the network structure to the data.
Preliminary analysis

Some preliminary findings:

- **Covariate effects:**
  - homophily by sex, smoking behavior and program;
  - smokers seem more outgoing and popular.

- **Network patterns:**
  - positive reciprocity;
  - outdegree variance is larger than indegree variance, and little correlation between the two.

To summarize covariate effects, some researchers employ “network regression:”

- convert sociomatrix and covariate matrices to vectors;
- perform ordinary regression (OLS, logistic regression, Poisson regression).

Such a procedure should **not** be called network regression:

- **it is just regression;**
- it ignores the network structure to the data.
Preliminary analysis

Some preliminary findings:

- **Covariate effects:**
  - homophily by sex, smoking behavior and program;
  - smokers seem more outgoing and popular.

- **Network patterns:**
  - positive reciprocity;
  - outdegree variance is larger than indegree variance, and little correlation between the two.

To summarize covariate effects, some researchers employ “network regression:”

- convert sociomatrix and covariate matrices to vectors;
- perform ordinary regression (OLS, logistic regression, Poisson regression).

Such a procedure should not be called network regression:

- it is just regression;
- it ignores the network structure to the data.
Preliminary analysis

Some preliminary findings:

- **Covariate effects:**
  - homophily by sex, smoking behavior and program;
  - smokers seem more outgoing and popular.

- **Network patterns:**
  - positive reciprocity;
  - outdegree variance is larger than indegree variance, and little correlation between the two.

To summarize covariate effects, some researchers employ “network regression:”

- convert sociomatrix and covariate matrices to vectors;
- perform ordinary regression (OLS, logistic regression, Poisson regression).

Such a procedure should **not** be called network regression:

- it is just **regression**;
- it ignores the network structure to the data.
Some preliminary findings:

- **Covariate effects:**
  - homophily by sex, smoking behavior and program;
  - smokers seem more outgoing and popular.

- **Network patterns:**
  - positive reciprocity;
  - outdegree variance is larger than indegree variance, and little correlation between the two.

To summarize covariate effects, some researchers employ “network regression:”

- convert sociomatrix and covariate matrices to vectors;
- perform ordinary regression (OLS, logistic regression, Poisson regression).

Such a procedure should **not** be called network regression:

- it is just **regression**;
- it ignores the network structure to the data.
Logistic regression

Nevertheless, regression with appropriate covariates might be adequate.

In particular, network patterns could be explained by covariates:
- degree heterogeneity could be explained by one or more nodal covariates;
- reciprocity could be explained by a group comembership variable.

Let's do an ordinary logistic regression and evaluate the fit.

```R
XM <- array(dim=c(n,n,5))
XM[,1] <- matrix( X[,2], n,n)
XM[,2] <- t(XM[,1])
XM[,3] <- outer( X[,1], X[,1], "==")
XM[,4] <- outer( X[,2], X[,2], "==")
XM[,5] <- outer( X[,3], X[,3], "==")

y7 <- c(Y7)
x <- apply(XM, 3, "c")
colnames(x) <- c("rsmoke", "csmoke", "ssex", "ssmoke", "sprog")

fit.glm <- glm( y7 ~ x, family=binomial)
```
Logistic regression

Nevertheless, regression with appropriate covariates might be adequate.

In particular, network patterns could be explained by covariates:

- degree heterogeneity could be explained by one or more nodal covariates;
- reciprocity could be explained by a group comembership variable.

Let’s do an ordinary logistic regression and evaluate the fit.

```
XM<-array(dim=c(n,n,5) )
XM[,,1]<-matrix( X[,2] ,n,n)
XM[,,2]<-t(XM[,,1])
XM[,,3]<-outer( X[,1],X[,1],"==")
XM[,,4]<-outer( X[,2],X[,2],"==")
XM[,,5]<-outer( X[,3],X[,3],"==")

y7<-c(Y7)
x<-apply(XM,3,"c")
colnames(x)<-c("rsmoke","csmoke","ssex","ssmoke","sprog")

fit.glm<-glm( y7 ~ x ,family=binomial)
```
Logistic regression
Nevertheless, regression with appropriate covariates might be adequate.

In particular, network patterns could be explained by covariates:
  • degree heterogeneity could be explained by one or more nodal covariates;
  • reciprocity could be explained by a group comembership variable.

Let’s do an ordinary logistic regression and evaluate the fit.

\[
\begin{align*}
XM &\leftarrow \text{array(dim=c(n,n,5))} \\
XM(:,:,1) &\leftarrow \text{matrix(X[,2],n,n)} \\
XM(:,:,2) &\leftarrow \text{t(XM(:,:,1))} \\
XM(:,:,3) &\leftarrow \text{outer(X[,1],X[,1],"==")} \\
XM(:,:,4) &\leftarrow \text{outer(X[,2],X[,2],"==")} \\
XM(:,:,5) &\leftarrow \text{outer(X[,3],X[,3],"==")} \\
y7 &\leftarrow \text{c(Y7)} \\
x &\leftarrow \text{apply(XM,3,"c")} \\
colnames(x) &\leftarrow \text{c("rsmoke","csmoke","ssex","ssmoke","sprog")} \\
\text{fit.glm} &\leftarrow \text{glm(y7 ~ x,family=binomial)}
\end{align*}
\]
Logistic regression

Nevertheless, regression with appropriate covariates might be adequate.

In particular, network patterns could be explained by covariates:
- degree heterogeneity could be explained by one or more nodal covariates;
- reciprocity could be explained by a group comembership variable.

Let’s do an ordinary logistic regression and evaluate the fit.

```r
XM <- array(dim=c(n,n,5) )
XM[,1]<-matrix( X[,2] ,n,n)
XM[,2]<-t(XM[,1])
XM[,3]<-outer( X[,1],X[,1],"==")
XM[,4]<-outer( X[,2],X[,2],"==")
XM[,5]<-outer( X[,3],X[,3],"==")
y7<-c(Y7)
x<-apply(XM,3,"c")
colnames(x)<-c("rsmoke","csmoke","sssex","sssmoke","sprog")
fit.glm<-glm( y7 ~ x ,family=binomial)
```
Logistic regression

Nevertheless, regression with appropriate covariates might be adequate.

In particular, network patterns could be explained by covariates:

- degree heterogeneity could be explained by one or more nodal covariates;
- reciprocity could be explained by a group comembership variable.

Let’s do an ordinary logistic regression and evaluate the fit.

```r
XM <- array(dim=c(n, n, 5))
XM[, , 1] <- matrix(X[, 2], n, n)
XM[, , 2] <- t(XM[, , 1])
XM[, , 3] <- outer(X[, 1], X[, 1], "==")
XM[, , 4] <- outer(X[, 2], X[, 2], "==")
XM[, , 5] <- outer(X[, 3], X[, 3], "==")

y7 <- c(Y7)
x <- apply(XM, 3, "c")
colnames(x) <- c("rsmoke", "csmoke", "sssex", "sssmoke", "sprog")

fit.glm <- glm(y7 ~ x, family = binomial)
```
Logistic regression

Nevertheless, regression with appropriate covariates might be adequate.

In particular, network patterns could be explained by covariates:
- degree heterogeneity could be explained by one or more nodal covariates;
- reciprocity could be explained by a group comembership variable.

Let’s do an ordinary logistic regression and evaluate the fit.

```r
XM <- array(dim=c(n,n,5))
XM[,,1] <- matrix(X[,2], n,n)
XM[,,2] <- t(XM[,,1])
XM[,,3] <- outer(X[,1],X[,1], "==")
XM[,,4] <- outer(X[,2],X[,2], "==")
XM[,,5] <- outer(X[,3],X[,3], "==")
y7 <- c(Y7)
x <- apply(XM, 3, "c")
colnames(x) <- c("rsmoke","csmoke","ssex","ssmoke","sprog")
fit.glm <- glm(y7 ~ x, family=binomial)
```
**Logistic regression fit**

```r
summary(fit.glm)
```

```r
##
## Call:
## glm(formula = y7 ~ x, family = binomial)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1144  -0.6909  -0.4446  -0.3119   2.4689
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -3.2122     0.2671  -12.027  < 2e-16 ***
## xrsmoke      0.2548     0.1882    1.354   0.175716
## xcsmoke      0.2130     0.1881    1.132   0.257500
## xssex        0.6930     0.2079    3.334   0.000857 ***
## xssmoke      0.7983     0.1863    4.285  1.83e-05 ***
## xsprog       1.1030     0.1800    6.127   8.94e-10 ***
##              ---
## Signif. codes:  0 '***'  0.001 '**'  0.01 '*'  0.05 '.'  0.1 ' ' 1
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 902.45  on 991  degrees of freedom
## Residual deviance: 819.39  on 986  degrees of freedom
##   (32 observations deleted due to missingness)
## AIC: 831.39
##
## Number of Fisher Scoring iterations: 5
```
Fitting logistic regression in \texttt{ergm}

Logistic regression is an ERGM with independent relations.

Suppose our model is

$$
\text{log odds}(y_{i,j} = 1) = \beta_0 + \beta_r x_{r,i} + \beta_c x_{c,j} + \beta_d x_{d,i,j}
$$

Then

$$
\Pr(Y = y | X, \beta) = \prod_{i \neq j} \left( \frac{e^{(\beta_0 + \beta_r x_{r,i} + \beta_c x_{c,j} + \beta_d x_{d,i,j}) y_{i,j}}}{1 + e^{\beta_0 + \beta_r x_{r,i} + \beta_c x_{c,j} + \beta_d x_{d,i,j}}} \right)
$$

$$
= c(X, \beta) \times \exp \left( \beta_0 \sum_{i \neq j} y_{i,j} + \beta_r \sum_{i \neq j} x_{r,i} y_{i,j} + \beta_c \sum_{i \neq j} x_{c,j} y_{i,j} + \beta_d \sum_{i \neq j} x_{d,i,j} y_{i,j} \right)
$$

The sufficient statistics simplify to the four-dimensional vector

$$
t(y) = \left( y_{.,.}, \sum_{i=1}^{n} x_{r,i} y_{i,.}, \sum_{j=1}^{n} x_{c,j} y_{.,j}, \sum_{i \neq j} x_{d,i,j} y_{i,j} \right).
$$
Fitting logistic regression in \texttt{ergm}

Logistic regression is an \texttt{ERGM} with independent relations.

Suppose our model is

$$\text{log odds}(y_{i,j} = 1) = \beta_0 + \beta_r x_{r,i} + \beta_c x_{c,j} + \beta_d x_{d,i,j}$$

Then

$$\Pr(Y = \mathbf{y} | \mathbf{X}, \mathbf{\beta}) = \prod_{i \neq j} \left( \frac{e^{(\beta_0 + \beta_r x_{r,i} + \beta_c x_{c,j} + \beta_d x_{d,i,j}) y_{i,j}}}{1 + e^{\beta_0 + \beta_r x_{r,i} + \beta_c x_{c,j} + \beta_d x_{d,i,j}}} \right)$$

$$= c(\mathbf{X}, \mathbf{\beta}) \times \exp \left( \beta_0 \sum_{i \neq j} y_{i,j} + \beta_r \sum_{i \neq j} x_{r,i} y_{i,j} + \beta_c \sum_{i \neq j} x_{c,j} y_{i,j} + \beta_d \sum_{i \neq j} x_{d,i,j} y_{i,j} \right)$$

The sufficient statistics simplify to the four-dimensional vector

$$\mathbf{t}(\mathbf{y}) = \left( y_{\cdot \cdot}, \sum_{i=1}^{n} x_{r,i} y_{i \cdot \cdot}, \sum_{j=1}^{n} x_{c,j} y_{j \cdot \cdot}, \sum_{i \neq j} x_{d,i,j} y_{i,j} \right)$$.
Fitting logistic regression in \texttt{ergm}

Logistic regression is an ERGM with independent relations.

Suppose our model is

\[
\log\text{odds}(y_{i,j} = 1) = \beta_0 + \beta_r x_{r,i} + \beta_c x_{c,j} + \beta_d x_{d,i,j}
\]

Then

\[
\Pr(Y = y|X, \beta) = \prod_{i \neq j} \left( \frac{e^{(\beta_0 + \beta_r x_{r,i} + \beta_c x_{c,j} + \beta_d x_{d,i,j}) y_{i,j}}}{1 + e^{\beta_0 + \beta_r x_{r,i} + \beta_c x_{c,j} + \beta_d x_{d,i,j}}} \right)
\]

\[
= c(X, \beta) \times \exp \left( \beta_0 \sum_{i \neq j} y_{i,j} + \beta_r \sum_{i \neq j} x_{r,i} y_{i,j} + \beta_c \sum_{i \neq j} x_{c,j} y_{i,j} + \beta_d \sum_{i \neq j} x_{d,i,j} y_{i,j} \right)
\]

The sufficient statistics simplify to the four-dimensional vector

\[
t(y) = \left( y_{i,j}, \sum_{i=1}^{n} x_{r,i} y_{i,j}, \sum_{j=1}^{n} x_{c,j} y_{j,i}, \sum_{i \neq j} x_{d,i,j} y_{i,j} \right).
\]
Fitting logistic regression in \texttt{ergm}

As logistic regression is an ERGM, we should be able to fit it with \texttt{ergm}.

We first need to convert the data to a network object:

\begin{verbatim}
library(ergm)
netdat<-network(Y7, vertex.attr=X)
\end{verbatim}

Sometimes you want to add vertex attributes one at a time:

\begin{verbatim}
netdat<-network(Y7)
set.vertex.attribute(netdat,"male",X[,1])
set.vertex.attribute(netdat,"smoker",X[,2])
set.vertex.attribute(netdat,"program",X[,3])
\end{verbatim}
As logistic regression is an ERGM, we should be able to fit it with \texttt{ergm}.

We first need to convert the data to a \texttt{network} object:

\begin{verbatim}
library(ergm)
netdat<-network(Y7,vertex.attr=X)
\end{verbatim}

Sometimes you want to add vertex attributes one at a time:

\begin{verbatim}
netdat<-network(Y7)
set.vertex.attribute(netdat,"male",X[,1])
set.vertex.attribute(netdat,"smoker",X[,2])
set.vertex.attribute(netdat,"program",X[,3])
\end{verbatim}
Fitting logistic regression in `ergm`

The model is then fit, as before, by specifying sufficient statistics:

```r
fit.ergm <- ergm( netdat ~ edges + nodeocov("smoker") + nodeicov("smoker") + nodematch("male") + nodematch("smoker") + nodematch("program") )
```

The terms `nodeicov`, `nodeocov` and `nodematch` create sufficient statistics out of nodal covariates:

- `nodeocov` creates a row regression effect;
- `nodeicov` creates a column regression effect;
- `nodematch` creates a dyadic binary indicator.

See the `ergm` manual for more details.
The model is then fit, as before, by specifying sufficient statistics:

```r
fit.ergm <- ergm( netdat ~ edges + nodeocov("smoker") + nodeicov("smoker") + nodematch("male") + nodematch("smoker") + nodematch("program") )
```

The terms `nodeicov`, `nodeocov` and `nodematch` create sufficient statistics out of nodal covariates:

- `nodeocov` creates a row regression effect;
- `nodeicov` creates a column regression effect;
- `nodematch` creates a dyadic binary indicator.

See the `ergm` manual for more details.
The model is then fit, as before, by specifying sufficient statistics:

```r
fit.ergm <- ergm( netdat ~ edges + nodeocov("smoker") + nodeicov("smoker") + nodematch("male") + nodematch("smoker") + nodematch("program") )
```

The terms `nodeicov`, `nodeocov` and `nodematch` create sufficient statistics out of nodal covariates:

- `nodeocov` creates a row regression effect;
- `nodeicov` creates a column regression effect;
- `nodematch` creates a dyadic binary indicator.

See the `ergm` manual for more details.
Fitting logistic regression in \texttt{ergm}

The model is then fit, as before, by specifying sufficient statistics:

\begin{verbatim}
fit.ergm <- \texttt{ergm( netdat \sim edges + nodeocov("smoker") + nodeicov("smoker") + nodematch("male") + nodematch("smoker") + nodematch("program") )}
\end{verbatim}

The terms \texttt{nodeicov}, \texttt{nodeocov} and \texttt{nodematch} create sufficient statistics out of nodal covariates:

\begin{itemize}
    \item \texttt{nodeocov} creates a row regression effect;
    \item \texttt{nodeicov} creates a column regression effect;
    \item \texttt{nodematch} creates a dyadic binary indicator .
\end{itemize}

See the \texttt{ergm} manual for more details.
The model is then fit, as before, by specifying sufficient statistics:

```r
fit.ergm <- ergm( netdat ~ edges + nodeocov("smoker") + nodeicov("smoker") + nodematch("male") + nodematch("smoker") + nodematch("program") )
```

The terms `nodeicov`, `nodeocov` and `nodematch` create sufficient statistics out of nodal covariates:

- `nodeocov` creates a row regression effect;
- `nodeicov` creates a column regression effect;
- `nodematch` creates a dyadic binary indicator.

See the `ergm` manual for more details.
Fitting logistic regression in ergm

The model is then fit, as before, by specifying sufficient statistics:

```r
fit.ergm <- ergm( netdat ~ edges + nodeocov("smoker") + nodeicov("smoker") + nodematch("male") + nodematch("smoker") + nodematch("program") )
```

The terms `nodeicov`, `nodeocov`, and `nodematch` create sufficient statistics out of nodal covariates:

- `nodeocov` creates a row regression effect;
- `nodeicov` creates a column regression effect;
- `nodematch` creates a dyadic binary indicator.

See the `ergm` manual for more details.
summary(fit.ergm)

##
## ================
## Summary of model fit
## ================
##
## Formula: netdat ~ edges + nodeocov("smoker") + nodeicov("smoker") + nodematch("male")
##   nodematch("smoker") + nodematch("program")
##
## Iterations: 5 out of 20
##
## Monte Carlo MLE Results:
##
## Estimate Std. Error MCMC % p-value
## edges -3.2122 0.2671 0 < 1e-04 ***
## nodeocov.smoker 0.2548 0.1882 0 0.176026
## nodeicov.smoker 0.2130 0.1881 0 0.257774
## nodematch.male 0.6930 0.2079 0 0.000889 ***
## nodematch.smoker 0.7983 0.1863 0 < 1e-04 ***
## nodematch.program 1.1030 0.1800 0 < 1e-04 ***
##
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Null Deviance: 1375 on 992 degrees of freedom
## Residual Deviance: 1292 on 986 degrees of freedom
##
## AIC: 1304    BIC: 1334    (Smaller is better.)
We fit this model without regard to its network structure:

- across sender heterogeneity/within sender correlation;
- across receiver heterogeneity/within receiver correlation;
- reciprocity/within dyad correlation.

It is possible that such patterns could be explained by covariates:

- heterogeneity in smoking leads to heterogeneity in degree;
- homophily for sex, smoking and group leads to reciprocity.

Let’s examine this with a goodness of fit evaluation.
We fit this model without regard to its network structure:

- across sender heterogeneity/within sender correlation;
- across receiver heterogeneity/within receiver correlation;
- reciprocity/within dyad correlation.

It is possible that such patterns could be explained by covariates:

- heterogeneity in smoking leads to heterogeneity in degree;
- homophily for sex, smoking and group leads to reciprocity.

Let’s examine this with a goodness of fit evaluation.
Goodness of fit

We fit this model without regard to its network structure:
- across sender heterogeneity/within sender correlation;
- across receiver heterogeneity/within receiver correlation;
- reciprocity/within dyad correlation.

It is possible that such patterns could be explained by covariates:
- heterogeneity in smoking leads to heterogeneity in degree;
- homophily for sex, smoking and group leads to reciprocity.

Let’s examine this with a goodness of fit evaluation.
Goodness of fit

We fit this model without regard to its network structure:

- across sender heterogeneity/within sender correlation;
- across receiver heterogeneity/within receiver correlation;
- reciprocity/within dyad correlation.

It is possible that such patterns could be explained by covariates:

- heterogeneity in smoking leads to heterogeneity in degree;
- homophily for sex, smoking and group leads to reciprocity.

Let’s examine this with a goodness of fit evaluation.
We fit this model without regard to its network structure:

- across sender heterogeneity/within sender correlation;
- across receiver heterogeneity/within receiver correlation;
- reciprocity/within dyad correlation.

It is possible that such patterns could be explained by covariates:

- heterogeneity in smoking leads to heterogeneity in degree;
- homophily for sex, smoking and group leads to reciprocity.

Let’s examine this with a goodness of fit evaluation.
We fit this model without regard to its network structure:

- across sender heterogeneity/within sender correlation;
- across receiver heterogeneity/within receiver correlation;
- reciprocity/within dyad correlation.

It is possible that such patterns could be explained by covariates:

- heterogeneity in smoking leads to heterogeneity in degree;
- homophily for sex, smoking and group leads to reciprocity.

Let’s examine this with a goodness of fit evaluation.
We fit this model without regard to its network structure:
  • across sender heterogeneity/within sender correlation;
  • across receiver heterogeneity/within receiver correlation;
  • reciprocity/within dyad correlation.

It is possible that such patterns could be explained by covariates:
  • heterogeneity in smoking leads to heterogeneity in degree;
  • homophily for sex, smoking and group leads to reciprocity.

Let’s examine this with a goodness of fit evaluation.
Goodness of fit

We fit this model without regard to its network structure:
- across sender heterogeneity/within sender correlation;
- across receiver heterogeneity/within receiver correlation;
- reciprocity/within dyad correlation.

It is possible that such patterns could be explained by covariates:
- heterogeneity in smoking leads to heterogeneity in degree;
- homophily for sex, smoking and group leads to reciprocity.

Let’s examine this with a goodness of fit evaluation.
Goodness of fit

We fit this model without regard to its network structure:

• across sender heterogeneity/within sender correlation;
• across receiver heterogeneity/within receiver correlation;
• reciprocity/within dyad correlation.

It is possible that such patterns could be explained by covariates:

• heterogeneity in smoking leads to heterogeneity in degree;
• homophily for sex, smoking and group leads to reciprocity.

Let’s examine this with a goodness of fit evaluation.
Goodness of fit

We fit this model without regard to its network structure:
- across sender heterogeneity/within sender correlation;
- across receiver heterogeneity/within receiver correlation;
- reciprocity/within dyad correlation.

It is possible that such patterns could be explained by covariates:
- heterogeneity in smoking leads to heterogeneity in degree;
- homophily for sex, smoking and group leads to reciprocity.

Let’s examine this with a goodness of fit evaluation.
Goodness of fit

```r
s.obs <- c(sd(rsum(Y7)), sd(csum(Y7)), mdyad(Y7))

py.hat <- fit.glm$fitted
s.SIM <- NULL
for(s in 1:S)
{
    Ysim <- matrix(NA, nrow(Y7), nrow(Y7))
    Ysim[!is.na(Y7)] <- rbinom(length(py.hat), 1, py.hat)
    s.SIM <- rbind(s.SIM, c(sd(rsum(Ysim)), sd(csum(Ysim)), mdyad(Ysim)))
}
```
**Goodness of fit**

```r
mean(s.SIM[,1]>=s.obs[1])
## [1] 0

mean(s.SIM[,2]>=s.obs[2])
## [1] 0.135

mean(s.SIM[,3]>=s.obs[3])
## [1] 0
```

**Evaluation:** These results indicate

- more outdegree heterogeneity than expected under the MLE;
- more reciprocity than expected;
- indegree heterogeneity is as expected.
Goodness of fit

\[ \text{mean}(s.SIM[,1] \geq s.obs[1]) \]

\[ \# [1] 0 \]

\[ \text{mean}(s.SIM[,2] \geq s.obs[2]) \]

\[ \# [1] 0.135 \]

\[ \text{mean}(s.SIM[,3] \geq s.obs[3]) \]

\[ \# [1] 0 \]

**Evaluation:** These results indicate

- more outdegree heterogeneity than expected under the MLE;
- more reciprocity than expected;
- indegree heterogeneity is as expected.
mean(s.SIM[,1]>=s.obs[1])
## [1] 0

mean(s.SIM[,2]>=s.obs[2])
## [1] 0.135

mean(s.SIM[,3]>=s.obs[3])
## [1] 0

**Evaluation:** These results indicate
- more outdegree heterogeneity than expected under the MLE;
- more reciprocity than expected;
- indegree heterogeneity is as expected.
Goodness of fit

```r
mean(s.SIM[,1]>=s.obs[1])
## [1] 0
mean(s.SIM[,2]>=s.obs[2])
## [1] 0.135
mean(s.SIM[,3]>=s.obs[3])
## [1] 0
```

**Evaluation:** These results indicate
- more outdegree heterogeneity than expected under the MLE;
- more reciprocity than expected;
- indegree heterogeneity is as expected.
Goodness of fit

```r
mean(s.SIM[,1]>=s.obs[1])
## [1] 0
mean(s.SIM[,2]>=s.obs[2])
## [1] 0.135
mean(s.SIM[,3]>=s.obs[3])
## [1] 0
```

**Evaluation:** These results indicate
- more outdegree heterogeneity than expected under the MLE;
- more reciprocity than expected;
- indegree heterogeneity is as expected.
This lack of fit can be addressed by adding statistics to the model:

```r
fit.p1cov.1 <- ergm( netdat ~ edges + sender + receiver + mutual +
                     nodeocov("smoker") + nodeicov("smoker") +
                     nodematch("male") + nodematch("smoker") + nodematch("program") )
```

```
## Error in which.package.InitFunction(fun): could not find function
"findFunction"
```

```r
summary(fit.p1cov.1)
```

```
## Error in summary(fit.p1cov.1): object 'fit.p1cov.1' not found
```
This lack of fit can be addressed by adding statistics to the model:

```r
fit.p1cov.1 <- ergm( netdat ~ edges + sender + receiver + mutual +
                   nodeocov("smoker") + nodeicov("smoker") +
                   nodematch("male") + nodematch("smoker") + nodematch("program") )
```

## Error in which.package.InitFunction(fun): could not find function
"findFunction"

```r
summary(fit.p1cov.1)
```

## Error in summary(fit.p1cov.1): object ’fit.p1cov.1’ not found
Regression terms

```r
fit.p1cov.1$coef[-(1:(2*n))]
```

```
## Error in eval(expr, envir, enclos): object 'fit.p1cov.1' not found
```
$p_1$ with alternative term order

```r
fit.p1cov.2 <- ergm(netdat ~
    nodeocov("smoker") + nodeicov("smoker") +
    nodematch("male") + nodematch("smoker") + nodematch("program") +
    edges + sender + receiver + mutual )
```

```
## Error in which.package.InitFunction(fun): could not find function "findFunction"

summary(fit.p1cov.2)

## Error in summary(fit.p1cov.2): object 'fit.p1cov.2' not found
```
fit.p1cov.2 <- ergm(netdat ~
    nodeocov("smoker") + nodeicov("smoker") +
    nodematch("male") + nodematch("smoker") + nodematch("program") +
    edges + sender + receiver + mutual )

## Error in which.package.InitFunction(fun): could not find function
"findFunction"

summary(fit.p1cov.2)

## Error in summary(fit.p1cov.2): object 'fit.p1cov.2' not found
The problem is **confounding** between these effects and the sender and receiver effects.

To illustrate this issue, consider a simple model with just

- sender effects;
- one sender-specific covariate.

\[
\text{Pr}(Y_{i,j} = y_{i,j}) = \frac{e^{(\mu + a_i + \beta x_i)y_{i,j}}}{1 + e^{\mu + a_i + \beta x_i}}
\]
The problem is **confounding** between these effects and the sender and receiver effects.

To illustrate this issue, consider a simple model with just

- sender effects;
- one sender-specific covariate.

\[
\Pr(Y_{i,j} = y_{i,j}) = \frac{e^{(\mu + a_i + \beta x_i) y_{i,j}}}{1 + e^{\mu + a_i + \beta x_i}}
\]
The problem is **confounding** between these effects and the sender and receiver effects.

To illustrate this issue, consider a simple model with just
- sender effects;
- one sender-specific covariate.

\[
\Pr(Y_{i,j} = y_{i,j}) = \frac{e^{(\mu + a_i + \beta x_i)y_{i,j}}}{1 + e^{\mu + a_i + \beta x_i}}
\]
The problem is **confounding** between these effects and the sender and receiver effects.

To illustrate this issue, consider a simple model with just

- sender effects;
- one sender-specific covariate.

\[
\Pr(Y_{i,j} = y_{i,j}) = \frac{e^{(\mu + a_i + \beta x_i)y_{i,j}}}{1 + e^{\mu + a_i + \beta x_i}}
\]
The problem is **confounding** between these effects and the sender and receiver effects.

To illustrate this issue, consider a simple model with just

- sender effects;
- one sender-specific covariate.

\[
Pr(Y_{i,j} = y_{i,j}) = \frac{e^{(\mu + a_i + \beta x_i)y_{i,j}}}{1 + e^{\mu + a_i + \beta x_i}}
\]
Confounding

The sufficient statistics can be found by summing the exponent over pairs:

\[
\sum_{i \neq j} \mu y_{i,j} + a_i y_{i,j} + \beta x_i y_{i,j} = \mu y. + \sum_i a_i y_i + \beta \sum_i x_i y_i.
\]

Naively, the parameters and sufficient statistics are

\[
\theta = (\mu, a_1, \ldots, a_n, \beta)
\]

\[
t(y) = (y., y_1., \ldots, y_n., \sum_i x_i y_i.)
\]

Note that

1. \( y. \) is a function of \( y_1., \ldots, y_n. \) (this leads to side conditions on the \( a_i \)'s);
2. \( \sum x_i y_i. \) is a function of \( y_1., \ldots, y_n. \) (the \( x_i \)'s are treated as “fixed” ).

This latter phenomenon means that \( \beta \) and the \( a_i \)'s are not jointly estimable.
Confounding

Let’s examine this more explicitly:

\[ t(y) \cdot \theta(\mu, a, \beta) = \mu y.. + \sum_i a_i y_i + \beta \sum_i x_i y_i. \]

\[ t(y) \cdot \theta(\mu, a - cx, \beta + c) = \mu y.. + \sum (a_i - cx_i)y_i + (\beta + c) \sum x_i y_i. \]

\[ = \mu y.. + \sum a_i y_i + \beta \sum x_i y_i. \]

\[ = t(y) \cdot \theta(\mu, a, \beta). \]

Nonidentifiability:
This result implies that for any two values of \( \beta \), say \( \beta_1 \) and \( \beta_2 \), there are vectors \( a_1 \) and \( a_2 \) such that

\[ l(\mu, a_1, \beta_1 : y) = l(\mu, a_2, \beta_2 : y). \]

The data information can’t distinguish between \( (\mu, a_1, \beta_1) \) and \( (\mu, a_2, \beta_2) \).
Confounding

Let’s examine this more explicitly:

\[ t(y) \cdot \theta(\mu, a, \beta) = \mu y \cdot + \sum_i a_i y_i + \beta \sum_i x_i y_i. \]

\[ t(y) \cdot \theta(\mu, a - cx, \beta + c) = \mu y \cdot + \sum_i (a_i - cx_i) y_i + (\beta + c) \sum_i x_i y_i. \]

\[ = \mu y \cdot + \sum_i a_i y_i + \beta \sum_i x_i y_i. \]

\[ = t(y) \cdot \theta(\mu, a, \beta). \]

Nonidentifiability:
This result implies that for any two values of \( \beta \), say \( \beta_1 \) and \( \beta_2 \), there are vectors \( a_1 \) and \( a_2 \) such that

\[ l(\mu, a_1, \beta_1 : y) = l(\mu, a_2, \beta_2 : y). \]

The data information can’t distinguish between \( (\mu, a_1, \beta_1) \) and \( (\mu, a_2, \beta_2) \).
Modeling options

There are three commonly used methods of addressing this issue:

1. fit the model without sender and receiver effects;
2. fit the model without sender and receiver regressors;
3. use a random effects model.

We don’t want to do 1 if the logistic regression model has been rejected.

We will fit the model in item 2, but use a two-stage procedure for estimating nodal covariate effects: For example,

- obtain $\hat{\theta} = (\hat{\mu}, \hat{a})$;
- fit the regression model $\hat{a}_i = \beta x_i + \epsilon_i$.

This is an ad-hoc approximation to the random effects approach:

- Model $y_{i,j}$ as a function of $a_i$;
- Model $a_i$ as a function of $x_i$

$$a_i = \beta x_i + \epsilon_i$$

$\{\epsilon_1, \ldots, \epsilon_n\} \sim \text{i.i.d. normal}(0, \sigma_a^2)$

We will cover such models shortly.
Modeling options

There are three commonly used methods of addressing this issue:

1. fit the model without sender and receiver effects;
2. fit the model without sender and receiver regressors;
3. use a random effects model.

We don’t want to do 1 if the logistic regression model has been rejected.

We will fit the model in item 2, but use a two-stage procedure for estimating nodal covariate effects: For example,

- obtain \( \hat{\theta} = (\hat{\mu}, \hat{\alpha}) \);
- fit the regression model \( \hat{\alpha}_i = \beta x_i + \epsilon_i \).

This is an ad-hoc approximation to the random effects approach:

- Model \( y_{i,j} \) as a function of \( a_i \);
- Model \( a_i \) as a function of \( x_i \)

\[
a_i = \beta x_i + \epsilon_i
\]

\( \{\epsilon_1, \ldots, \epsilon_n\} \sim \text{i.i.d.normal}(0, \sigma_a^2) \)

We will cover such models shortly.
Modeling options

There are three commonly used methods of addressing this issue:

1. fit the model without sender and receiver effects;
2. fit the model without sender and receiver regressors;
3. use a random effects model.

We don’t want to do 1 if the logistic regression model has been rejected.

We will fit the model in item 2, but use a two-stage procedure for estimating nodal covariate effects: For example,

- obtain $\hat{\theta} = (\hat{\mu}, \hat{a})$;
- fit the regression model $\hat{a}_i = \beta x_i + \epsilon_i$.

This is an ad-hoc approximation to the random effects approach:

- Model $y_{i,j}$ as a function of $a_i$;
- Model $a_i$ as a function of $x_i$.

$$a_i = \beta x_i + \epsilon_i$$

$\{\epsilon_1, \ldots, \epsilon_n\} \sim \text{i.i.d.normal}(0, \sigma^2_a)$

We will cover such models shortly.
Modeling options

There are three commonly used methods of addressing this issue:

1. fit the model without sender and receiver effects;
2. fit the model without sender and receiver regressors;
3. use a random effects model.

We don’t want to do 1 if the logistic regression model has been rejected.

We will fit the model in item 2, but use a two-stage procedure for estimating nodal covariate effects: For example,

- obtain $\hat{\theta} = (\hat{\mu}, \hat{a})$;
- fit the regression model $\hat{a}_i = \beta x_i + \epsilon_i$.

This is an ad-hoc approximation to the random effects approach:

- Model $y_{i,j}$ as a function of $a_i$;
- Model $a_i$ as a function of $x_i$.

$$a_i = \beta x_i + \epsilon_i$$

$\{\epsilon_1, \ldots, \epsilon_n\} \sim \text{i.i.d.normal}(0, \sigma^2_a)$

We will cover such models shortly.
Modeling options

There are three commonly used methods of addressing this issue:

1. fit the model without sender and receiver effects;
2. fit the model without sender and receiver regressors;
3. use a random effects model.

We don’t want to do 1 if the logistic regression model has been rejected.

We will fit the model in item 2, but use a two-stage procedure for estimating nodal covariate effects: For example,

- obtain $\hat{\theta} = (\hat{\mu}, \hat{a})$;
- fit the regression model $\hat{a}_i = \beta x_i + \epsilon_i$.

This is an ad-hoc approximation to the random effects approach:

- Model $y_{i,j}$ as a function of $a_i$;
- Model $a_i$ as a function of $x_i$

\[ a_i = \beta x_i + \epsilon_i \]
\[ \{\epsilon_1, \ldots, \epsilon_n\} \sim \text{i.i.d.normal}(0, \sigma_a^2) \]

We will cover such models shortly.
Modeling options

There are three commonly used methods of addressing this issue:

1. fit the model without sender and receiver effects;
2. fit the model without sender and receiver regressors;
3. use a random effects model.

We don’t want to do 1 if the logistic regression model has been rejected.

We will fit the model in item 2, but use a two-stage procedure for estimating nodal covariate effects: For example,

- obtain $\hat{\theta} = (\hat{\mu}, \hat{a})$;
- fit the regression model $\hat{a}_i = \beta x_i + \epsilon_i$.

This is an ad-hoc approximation to the random effects approach:

- Model $y_{i,j}$ as a function of $a_i$;
- Model $a_i$ as a function of $x_i$.

$$a_i = \beta x_i + \epsilon_i$$

$$\{\epsilon_1, \ldots, \epsilon_n\} \sim \text{i.i.d.normal}(0, \sigma_a^2)$$

We will cover such models shortly.
Modeling options

There are three commonly used methods of addressing this issue:

1. fit the model without sender and receiver effects;
2. fit the model without sender and receiver regressors;
3. use a random effects model.

We don’t want to do 1 if the logistic regression model has been rejected.

We will fit the model in item 2, but use a two-stage procedure for estimating nodal covariate effects: For example,

- obtain $\hat{\boldsymbol{\theta}} = (\hat{\mu}, \hat{\alpha})$;
- fit the regression model $\hat{a}_i = \beta x_i + \epsilon_i$.

This is an ad-hoc approximation to the random effects approach:

- Model $y_{i,j}$ as a function of $a_i$;
- Model $a_i$ as a function of $x_i$

\[ a_i = \beta x_i + \epsilon_i \]
\[ \{\epsilon_1, \ldots, \epsilon_n\} \sim \text{i.i.d. normal}(0, \sigma_a^2) \]

We will cover such models shortly.
Modeling options

There are three commonly used methods of addressing this issue:

1. fit the model without sender and receiver effects;
2. fit the model without sender and receiver regressors;
3. use a random effects model.

We don’t want to do 1 if the logistic regression model has been rejected.

We will fit the model in item 2, but use a two-stage procedure for estimating nodal covariate effects: For example,

- obtain \( \hat{\theta} = (\hat{\mu}, \hat{\alpha}) \);
- fit the regression model \( \hat{a}_i = \beta x_i + \epsilon_i \).

This is an ad-hoc approximation to the random effects approach:

- Model \( y_{i,j} \) as a function of \( a_i \);
- Model \( a_i \) as a function of \( x_i \)

\[
\begin{align*}
\hat{a}_i &= \beta x_i + \epsilon_i \\
\{\epsilon_1, \ldots, \epsilon_n\} &\sim \text{i.i.d.normal}(0, \sigma^2_a)
\end{align*}
\]

We will cover such models shortly.
Modeling options

There are three commonly used methods of addressing this issue:

1. fit the model without sender and receiver effects;
2. fit the model without sender and receiver regressors;
3. use a random effects model.

We don’t want to do 1 if the logistic regression model has been rejected.

We will fit the model in item 2, but use a two-stage procedure for estimating nodal covariate effects: For example,

- obtain $\hat{\theta} = (\hat{\mu}, \hat{\alpha})$;
- fit the regression model $\hat{a}_i = \beta x_i + \epsilon_i$.

This is an ad-hoc approximation to the random effects approach:

- Model $y_{i,j}$ as a function of $a_i$;
- Model $a_i$ as a function of $x_i$

$$a_i = \beta x_i + \epsilon_i$$

$\{\epsilon_1, \ldots, \epsilon_n\} \sim \text{i.i.d.normal}(0, \sigma_a^2)$

We will cover such models shortly.
Modeling options

There are three commonly used methods of addressing this issue:

1. fit the model without sender and receiver effects;
2. fit the model without sender and receiver regressors;
3. use a random effects model.

We don’t want to do 1 if the logistic regression model has been rejected.

We will fit the model in item 2, but use a two-stage procedure for estimating nodal covariate effects: For example,

- obtain $\hat{\theta} = (\hat{\mu}, \hat{\alpha})$;
- fit the regression model $\hat{a}_i = \beta x_i + \epsilon_i$.

This is an ad-hoc approximation to the random effects approach:

- Model $y_{i,j}$ as a function of $a_i$;
- Model $a_i$ as a function of $x_i$

$$a_i = \beta x_i + \epsilon_i$$

$$\{\epsilon_1, \ldots, \epsilon_n\} \sim \text{i.i.d. normal}(0, \sigma_a^2)$$

We will cover such models shortly.
Dyadic covariates for $p1$

```r
fit.p1cov.d <- ergm(netdat ~ nodematch("male") + nodematch("smoker") + nodematch("program") + edges + mutual + sender + receiver )
```

```r
## Error in which.package.InitFunction(fun): could not find function "findFunction"
```

```r
summary(fit.p1cov.d)
```

```r
## Error in summary(fit.p1cov.d): object 'fit.p1cov.d' not found
```
a.hat<-c(0,fit.p1cov.d$coef[4+(2:nrow(Y))] )

## Error in eval(expr, envir, enclos): object 'fit.p1cov.d' not found

b.hat<-c(0,fit.p1cov.d$coef[4+ nrow(Y)-1 + (2:nrow(Y))])

## Error in eval(expr, envir, enclos): object 'fit.p1cov.d' not found

## Error in xy.coords(x, y, xlabel, ylabel, log): object 'a.hat' not found
## Error in xy.coords(x, y, xlabel, ylabel, log): object 'b.hat' not found
## Error in plot(a.hat, b.hat): object 'a.hat' not found
How does a covariate $\mathbf{x} = \{x_1, \ldots, x_n\}$ relate to

- outgoingness ($a_1, \ldots, a_n$)?
- popularity ($b_1, \ldots, b_n$)?

```r
## Error in eval(expr, envir, enclos): object 'a.hat' not found
## Error in eval(expr, envir, enclos): object 'b.hat' not found
```
Nodal covariate effects

How does a covariate $\mathbf{x} = \{x_1, \ldots, x_n\}$ relate to

- outgoingness $(a_1, \ldots, a_n)$?
- popularity $(b_1, \ldots, b_n)$?

```r
## Error in eval(expr, envir, enclos): object 'a.hat' not found
## Error in eval(expr, envir, enclos): object 'b.hat' not found
```
Nodal covariate effects

How does a covariate $x = \{x_1, \ldots, x_n\}$ relate to

- outgoingness $(a_1, \ldots, a_n)$?
- popularity $(b_1, \ldots, b_n)$?

## Error in eval(expr, envir, enclos): object 'a.hat' not found
## Error in eval(expr, envir, enclos): object 'b.hat' not found
How does a covariate $\mathbf{x} = \{x_1, \ldots, x_n\}$ relate to

- outgoingness $(a_1, \ldots, a_n)$?
- popularity $(b_1, \ldots, b_n)$?

```r
## Error in eval(expr, envir, enclos): object 'a.hat' not found
## Error in eval(expr, envir, enclos): object 'b.hat' not found
```
lm(a.hat~xsmoke)

## Error in eval(expr, envir, enclos): object 'a.hat' not found

The problem here is that $\hat{a}_i$ is $-\infty$ for nodes with zero outdegree. What can we do?

0. give up;
1. remove the problematic observations;
2. replace the problematic observations with some large negative value;
3. fit a random effects model.

Item 1 removes information and biases the results:

- Zero degree nodes are highly informative about covariate effects.
- Their removal could bias the estimated effects towards zero.

Item 2 requires we can pick the “right” replacement value.
The problem here is that $\hat{a}_i$ is $-\infty$ for nodes with zero outdegree. What can we do?

0. give up;
1. remove the problematic observations;
2. replace the problematic observations with some large negative value;
3. fit a random effects model.

Item 1 removes information and biases the results:

- Zero degree nodes are highly informative about covariate effects.
- Their removal could bias the estimated effects towards zero.

Item 2 requires we can pick the “right” replacement value.
Statistical evaluation

lm(a.hat ~ xsmoke)

## Error in eval(expr, envir, enclos): object 'a.hat' not found

The problem here is that $\hat{a}_i$ is $-\infty$ for nodes with zero outdegree. What can we do?

0. give up;

1. remove the problematic observations;

2. replace the problematic observations with some large negative value;

3. fit a random effects model.

Item 1 removes information and biases the results:

- Zero degree nodes are highly informative about covariate effects.
- Their removal could bias the estimated effects towards zero.

Item 2 requires we can pick the “right” replacement value.
lm(a.hat~xsmoke)

```
## Error in eval(expr, envir, enclos): object 'a.hat' not found
```

The problem here is that $\hat{a}_i$ is $-\infty$ for nodes with zero outdegree. What can we do?

1. give up;
2. remove the problematic observations;
3. replace the problematic observations with some large negative value;
4. fit a random effects model.

Item 1 removes information and biases the results:

- Zero degree nodes are highly informative about covariate effects.
- Their removal could bias the estimated effects towards zero.

Item 2 requires we can pick the “right” replacement value.
The problem here is that $\hat{a}_i$ is $-\infty$ for nodes with zero outdegree. What can we do?

0. give up;
1. remove the problematic observations;
2. replace the problematic observations with some large negative value;
3. fit a random effects model.

Item 1 removes information and biases the results:

- Zero degree nodes are highly informative about covariate effects.
- Their removal could bias the estimated effects towards zero.

Item 2 requires we can pick the “right” replacement value.
The problem here is that $\hat{a}_i$ is $-\infty$ for nodes with zero outdegree. What can we do?

0. give up;
1. remove the problematic observations;
2. replace the problematic observations with some large negative value;
3. fit a random effects model.

Item 1 removes information and biases the results:

- Zero degree nodes are highly informative about covariate effects.
- Their removal could bias the estimated effects towards zero.

Item 2 requires we can pick the “right” replacement value.
Statistical evaluation

\texttt{lm(a.hat \sim xsmoke)}

\texttt{## Error in eval(expr, envir, enclos): object 'a.hat' not found}

The problem here is that $\hat{a}_i$ is $-\infty$ for nodes with zero outdegree. What can we do?

0. give up;
1. remove the problematic observations;
2. replace the problematic observations with some large negative value;
3. fit a random effects model.

Item 1 removes information and biases the results:

- Zero degree nodes are highly informative about covariate effects.
- Their removal could bias the estimated effects towards zero.

Item 2 requires we can pick the “right” replacement value.
lm(a.hat~xsmoke)

## Error in eval(expr, envir, enclos): object 'a.hat' not found

The problem here is that \( \hat{a}_i \) is \(-\infty\) for nodes with zero outdegree. What can we do?

0. give up;
1. remove the problematic observations;
2. replace the problematic observations with some large negative value;
3. fit a random effects model.

Item 1 removes information and biases the results:

- Zero degree nodes are highly informative about covariate effects.
- Their removal could bias the estimated effects towards zero.

Item 2 requires we can pick the “right” replacement value.
The problem here is that $\hat{a}_i$ is $-\infty$ for nodes with zero outdegree. What can we do?

0. give up;

1. remove the problematic observations;

2. replace the problematic observations with some large negative value;

3. fit a random effects model.

Item 1 removes information and biases the results:

- Zero degree nodes are highly informative about covariate effects.
- Their removal could bias the estimated effects towards zero.

Item 2 requires we can pick the “right” replacement value.
lm(a.hat~xsmoke)

## Error in eval(expr, envir, enclos): object 'a.hat' not found

The problem here is that $\hat{a}_i$ is $-\infty$ for nodes with zero outdegree. What can we do?

0. give up;
1. remove the problematic observations;
2. replace the problematic observations with some large negative value;
3. fit a random effects model.

Item 1 removes information and biases the results:

- Zero degree nodes are highly informative about covariate effects.
- Their removal could bias the estimated effects towards zero.

Item 2 requires we can pick the “right” replacement value.
lm(a.hat ~ xsmoke)

## Error in eval(expr, envir, enclos): object 'a.hat' not found

The problem here is that $\hat{a}_i$ is $-\infty$ for nodes with zero outdegree. What can we do?

0. give up;
1. remove the problematic observations;
2. replace the problematic observations with some large negative value;
3. fit a random effects model.

Item 1 removes information and biases the results:

- Zero degree nodes are highly informative about covariate effects.
- Their removal could bias the estimated effects towards zero.

Item 2 requires we can pick the “right” replacement value.
The problem here is that $\hat{a}_i$ is $-\infty$ for nodes with zero outdegree. What can we do?

0. give up;
1. remove the problematic observations;
2. replace the problematic observations with some large negative value;
3. fit a random effects model.

Item 1 removes information and biases the results:

- Zero degree nodes are highly informative about covariate effects.
- Their removal could bias the estimated effects towards zero.

Item 2 requires we can pick the “right” replacement value.
Ad-hoc statistical evaluation

```
a.hat[a.hat == -Inf] <- NA

## Error in a.hat[a.hat == -Inf] <- NA: object 'a.hat' not found

b.hat[b.hat == -Inf] <- NA

## Error in b.hat[b.hat == -Inf] <- NA: object 'b.hat' not found

summary(glm(a.hat~xsmoke))$coef

## Error in eval(expr, envir, enclos): object 'a.hat' not found

summary(glm(b.hat~xsmoke))$coef

## Error in eval(expr, envir, enclos): object 'b.hat' not found
```

The results suggest that smoking doesn’t have a large effect on sender or receiver effects, and hence on outgoingness or popularity.
Ad-hoc statistical evaluation

**However:** What if

- all $-\infty a_i$’s corresponded to smokers?
- all $-\infty b_j$’s corresponded to nonsmokers?

Either possibility would suggest estimating the parameter as further away from zero, making it “more significant.”

```r
xsmoke[is.na(a.hat)]
## Error in eval(expr, envir, enclos): object 'a.hat' not found
xsmoke[is.na(b.hat)]
## Error in eval(expr, envir, enclos): object 'b.hat' not found
mean(xsmoke)
## [1] 0.40625
mean(xsmoke[is.na(a.hat)])
## Error in mean(xsmoke[is.na(a.hat)]): object 'a.hat' not found
```

These results don’t give indications of strong relationships between smoking and the tendency to send or receive ties.
Ad-hoc statistical evaluation

However: What if

- all \(-\infty \ a_i\)'s corresponded to smokers?
- all \(-\infty \ b_j\)'s corresponded to nonsmokers?

Either possibility would suggest a estimating the parameter as further away from zero, making it “more significant.”

```r
xsmoke[is.na(a.hat)]
## Error in eval(expr, envir, enclos): object 'a.hat' not found
xsmoke[is.na(b.hat)]
## Error in eval(expr, envir, enclos): object 'b.hat' not found
mean(xsmoke)
## [1] 0.40625
mean(xsmoke[is.na(a.hat)])
## Error in mean(xsmoke[is.na(a.hat)]): object 'a.hat' not found
```

These results don’t give indications of strong relationships between smoking and the tendency to send or receive ties.
Ad-hoc statistical evaluation

However: What if

- all \(-\infty\) \(a_i\)’s corresponded to smokers?
- all \(-\infty\) \(b_j\)’s corresponded to nonsmokers?

Either possibility would suggest a estimating the parameter as further away from zero, making it “more significant.”

```r
xsmoke[is.na(a.hat)]
## Error in eval(expr, envir, enclos): object ’a.hat’ not found
xsmoke[is.na(b.hat)]
## Error in eval(expr, envir, enclos): object ’b.hat’ not found
mean(xsmoke)
## [1] 0.40625
mean(xsmoke[is.na(a.hat)])
## Error in mean(xsmoke[is.na(a.hat)]): object ’a.hat’ not found
```

These results don’t give indications of strong relationships between smoking and the tendency to send or receive ties.
Ad-hoc statistical evaluation

**However:** What if

- all \(-\infty\) \(a_i\)'s corresponded to smokers?
- all \(-\infty\) \(b_j\)'s corresponded to nonsmokers?

Either possibility would suggest a estimating the parameter as further away from zero, making it “more significant.”

```r
xsmoke[is.na(a.hat)]
## Error in eval(expr, envir, enclos): object ’a.hat’ not found
xsmoke[is.na(b.hat)]
## Error in eval(expr, envir, enclos): object ’b.hat’ not found
mean(xsmoke)
## [1] 0.40625
mean(xsmoke[is.na(a.hat)])
## Error in mean(xsmoke[is.na(a.hat)]): object ’a.hat’ not found
```

These results don’t give indications of strong relationships between smoking and the tendency to send or receive ties.
Ad-hoc statistical evaluation

**However:** What if

- all $-\infty a_i$’s corresponded to smokers?
- all $-\infty b_j$’s corresponded to nonsmokers?

Either possibility would suggest a estimating the parameter as further away from zero, making it “more significant.”

```r
xsmoke[is.na(a.hat)]
## Error in eval(expr, envir, enclos): object ’a.hat’ not found
xsmoke[is.na(b.hat)]
## Error in eval(expr, envir, enclos): object ’b.hat’ not found
mean(xsmoke)
## [1] 0.40625
mean(xsmoke[is.na(a.hat)])
## Error in mean(xsmoke[is.na(a.hat)]): object ’a.hat’ not found
```

These results don’t give indications of strong relationships between smoking and the tendency to send or receive ties.
Ad-hoc statistical evaluation

**However:** What if

- all \(-\infty \) \(a_i\)’s corresponded to smokers?
- all \(-\infty \) \(b_j\)’s corresponded to nonsmokers?

Either possibility would suggest a estimating the parameter as further away from zero, making it “more significant.”

```r
xsmoke[is.na(a.hat)] # Error in eval(expr, envir, enclos): object ’a.hat’ not found
xsmoke[is.na(b.hat)] # Error in eval(expr, envir, enclos): object ’b.hat’ not found
mean(xsmoke) # [1] 0.40625
mean(xsmoke[is.na(a.hat)]) # Error in mean(xsmoke[is.na(a.hat)]): object ’a.hat’ not found
```

These results don’t give indications of strong relationships between smoking and the tendency to send or receive ties.
```
mdyad(Y)
## [1] 43

## expected value conditional on outdegree  WF p. 517
((sum(Y,na.rm=TRUE)^2 - sum(rsum(Y)^2)))/(2*(nrow(Y)-1)^2)
## [1] 1.178715
```
fit.0 <- ergm(Y ~ edges)

## Error in which.package.InitFunction(fun): could not find function "findFunction"

s.SIM0 <- NULL
for(s in 1:S)
{
  Ysim <- as.matrix(simulate(fit.0))
  diag(Ysim) <- NA
  s.SIM0 <- rbind(s.SIM0, c(sd(rsum(Ysim)), sd(csum(Ysim)), mdyad(Ysim)))
}

## Error in simulate(fit.0): object 'fit.0' not found

## Error in hist.default(tH, xlim = xlim, main = "", prob = TRUE, col = ncol, : 'x' must be numeric
Additionally, we have the following covariates:

- **Nodal covariates:**
  - population
  - gdp
  - polity

- **Dyad covariates:**
  - exports
  - shared IGOs
  - geographic distance

Let’s see if these covariates account for any of the network patterns.
Additionally, we have the following covariates:

- **Nodal covariates:**
  - population
  - gdp
  - polity

- **Dyad covariates:**
  - exports
  - shared IGOs
  - geographic distance

Let’s see if these covariates account for any of the network patterns.
Additionally, we have the following covariates:

- **Nodal covariates:**
  - population
  - gdp
  - polity

- **Dyad covariates:**
  - exports
  - shared IGOs
  - geographic distance

Let’s see if these covariates account for any of the network patterns.
Covariate information

Additionally, we have the following covariates:

- **Nodal covariates:**
  - population
  - gdp
  - polity

- **Dyad covariates:**
  - exports
  - shared IGOs
  - geographic distance

Let’s see if these covariates account for any of the network patterns.
Additionally, we have the following covariates:

- **Nodal covariates:**
  - population
  - gdp
  - polity

- **Dyad covariates:**
  - exports
  - shared IGOs
  - geographic distance

Let’s see if these covariates account for any of the network patterns.
Additionally, we have the following covariates:

- **Nodal covariates:**
  - population
  - gdp
  - polity

- **Dyad covariates:**
  - exports
  - shared IGOs
  - geographic distance

Let's see if these covariates account for any of the network patterns.
Additionally, we have the following covariates:

- **Nodal covariates:**
  - population
  - gdp
  - polity

- **Dyad covariates:**
  - exports
  - shared IGOs
  - geographic distance

Let’s see if these covariates account for any of the network patterns.
Covariate information

Additionally, we have the following covariates:

- **Nodal covariates:**
  - population
  - gdp
  - polity

- **Dyad covariates:**
  - exports
  - shared IGOs
  - geographic distance

Let’s see if these covariates account for any of the network patterns.
Additionally, we have the following covariates:

- **Nodal covariates:**
  - population
  - gdp
  - polity

- **Dyad covariates:**
  - exports
  - shared IGOs
  - geographic distance

Let’s see if these covariates account for any of the network patterns.
Additionally, we have the following covariates:

- **Nodal covariates:**
  - population
  - gdp
  - polity

- **Dyad covariates:**
  - exports
  - shared IGOs
  - geographic distance

Let’s see if these covariates account for any of the network patterns.
It is common to log values of money, population and distance:

```
colnames(Xn)
## [1] "pop" "gdp" "polity"
Xn[,1:2]<-log(Xn[,1:2])
colnames(Xn)<-c("lpop","lgdp","polity")
netdat<-network(Y,vertex.attr=as.data.frame(Xn))
```

Dyad covariates enter into `ergm` via the `edgecov` function:

```
fit.cov.ergm<-ergm( netdat ~ edges +
    nodeocov("lpop") + nodeocov("lgdp") + nodeocov("polity") +
    nodeicov("lpop") + nodeicov("lgdp") + nodeicov("polity") +
    edgecov(Xpol) + edgecov(Xigo) + edgecov(Xldst) + edgecov(Xlexp) + edgecov(Xlimp))
```
Coding covariates

It is common to log values of money, population and distance:

```r
colnames(Xn)
## [1] "pop" "gdp" "polity"
Xn[,1:2]<-log(Xn[,1:2])
colnames(Xn)<-c("lpop","lgdp","polity")
netdat<-network(Y,vertex.attr=as.data.frame(Xn))
```

Dyad covariates enter into `ergm` via the `edgecov` function:

```r
fit.cov.ergm<-ergm( netdat ~ edges +
               nodeocov("lpop") + nodeocov("lgdp") + nodeocov("polity") +
               nodeicov("lpop") + nodeicov("lgdp") + nodeicov("polity") +
               edgecov(Xpol) + edgecov(Xigo) + edgecov(Xldst) + edgecov(Xlexp) + edgecov(Xlimp))
```
Logistic regression fit

```r
summary(fit.cov.ergm)
```

```r
##
## # Summary of model fit
## # ===============
##
## # Formula: netdat ~ edges + nodeocov("lpop") + nodeocov("lgdp") + nodeocov("polity") +
## #    nodeicov("lpop") + nodeicov("lgdp") + nodeicov("polity") +
## #    edgecov(Xpol) + edgecov(Xigo) + edgecov(Xldst) + edgecov(Xlexp) +
## #    edgecov(Xlimp)
##
## # Iterations: 9 out of 20
##
## # Monte Carlo MLE Results:
## #
## | Estimate | Std. Error | MCMC | % | p-value |
## |----------|------------|------|---|---------|
## | edges    | -2.548601  | 0.362832 | 0 | < 1e-04 *** |
## | nodeocov.lpop | 0.204650 | 0.083405 | 0 | 0.014150 * |
## | nodeocov.lgdp | 0.277993 | 0.080569 | 0 | 0.000561 *** |
## | nodeocov.polity | -0.081600 | 0.012262 | 0 | < 1e-04 *** |
## | nodeicov.lpop | 0.193615 | 0.083862 | 0 | 0.020970 * |
## | nodeicov.lgdp | 0.171160 | 0.079843 | 0 | 0.032071 * |
## | nodeicov.polity | -0.037818 | 0.012390 | 0 | 0.002274 ** |
## | edgecov.Xpol | -0.004510 | 0.001659 | 0 | 0.006551 ** |
## | edgecov.Xigo | -0.011437 | 0.005592 | 0 | 0.040844 * |
## | edgecov.Xldst | -2.663417 | 0.142696 | 0 | < 1e-04 *** |
## | edgecov.Xlexp | 0.058343 | 0.426599 | 0 | 0.891219 |
## | edgecov.Xlimp | -0.035318 | 0.428694 | 0 | 0.934341 |
##
## # Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```
s.SIM <- NULL

for(s in 1:S)
{
  Ysim <- as.matrix(simulate(fit.cov.ergm))
  diag(Ysim) <- NA
  s.SIM <- rbind(s.SIM, c(sd(rsum(Ysim)), sd(csum(Ysim)), mdyad(Ysim)))
}
Improvement via covariates

```r
## Error in hist.default(tH, xlim = xlim, main = "", prob = TRUE, col = ncol, :
  'x' must be numeric
```
fit.cov.ergm<-ergm( netdat ~ edges +
# nodeocov("lpop") + nodeocov("lgdp") + nodeocov("polity") +
# nodeicov("lpop") + nodeicov("lgdp") + nodeicov("polity") +
  sender + receiver + mutual +
edgecov(Xpol) + edgecov(Xigo) + edgecov(Xldst) + edgecov(Xlexp) + edgecov(Xlimp))

## Observed statistic(s) sender2, sender3, sender7, sender9, sender10, sender11, sender12, ...
## Iteration 1 of at most 20:
## The log-likelihood improved by 0.9757
## Iteration 2 of at most 20:
## The log-likelihood improved by 1.004
## Iteration 3 of at most 20:
## The log-likelihood improved by 1.049
## Iteration 4 of at most 20:
## The log-likelihood improved by 0.9582
## Iteration 5 of at most 20:
## The log-likelihood improved by 0.9893
## Iteration 6 of at most 20:
## The log-likelihood improved by 0.8685
## Iteration 7 of at most 20:
## The log-likelihood improved by 1.025
## Iteration 8 of at most 20:
## The log-likelihood improved by 0.8937
## Iteration 9 of at most 20:
## The log-likelihood improved by 0.9246
## Iteration 10 of at most 20:
## The log-likelihood improved by 0.9991
## Iteration 11 of at most 20:
## The log-likelihood improved by 0.9405
## Iteration 12 of at most 20:
## The log-likelihood improved by 0.985
## Iteration 13 of at most 20:
ahat <- c(0, fit.cov.ergm$coef[1:(nrow(Y)-1) ] )
bhat <- c(0, fit.cov.ergm$coef[nrow(Y)+(1:(nrow(Y)-1) ) ] )

summary(lm(ahat[ahat > -Inf] ~ Xn[ahat > -Inf, ]))

##
## Call:
## lm(formula = ahat[ahat > -Inf] ~ Xn[ahat > -Inf, ])
##
## Residuals:
##    Min 1Q Median 3Q Max
## -4.8238 -0.6376 -0.0533 0.5625 5.2171
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)            -0.72097   0.49420  -1.459  0.14957
## Xn[ahat > -Inf, ]lpop  0.16273   0.20737   0.785  0.43555
## Xn[ahat > -Inf, ]lgdp  0.51733   0.18447   2.804  0.00669 **
## Xn[ahat > -Inf, ]polity -0.04194   0.03270  -1.282  0.20441
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.611 on 63 degrees of freedom
## Multiple R-squared: 0.3432, Adjusted R-squared: 0.312
## F-statistic: 10.98 on 3 and 63 DF, p-value: 6.838e-06
Summary

- Covariates can be included in the ERGMs.
  - dyad level covariates: nodematch, edgecov and others;
  - node level covariates: nodeocov, nodeicov and others.

- Covariates can often partially explain degree heterogeneity and reciprocity.
- Node-level parameters are confounded with node-level covariate effects.
  - two stage approach: fit node-level parameters, and then relate to covariates;
  - random effects model: next lecture.
Summary

- Covariates can be included in the ERGMs.
  - dyad level covariates: `nodematch`, `edgecov` and others;
  - node level covariates: `nodeocov`, `nodeicov` and others.
- Covariates can often partially explain degree heterogeneity and reciprocity.
- Node-level parameters are confounded with node-level covariate effects.
  - two stage approach: fit node-level parameters, and then relate to covariates;
  - random effects model: next lecture.
Summary

• Covariates can be included in the ERGMs.
  • dyad level covariates: \texttt{nodematch}, \texttt{edgecov} and others;
  • node level covariates: \texttt{nodeocov}, \texttt{nodeicov} and others.

• Covariates can often partially explain degree heterogeneity and reciprocity.
• Node-level parameters are confounded with node-level covariate effects.
  • two stage approach: fit node-level parameters, and then relate to covariates;
  • random effects model: next lecture.
Summary

• Covariates can be included in the ERGMs.
  • dyad level covariates: nodematch, edgecov and others;
  • node level covariates: nodeocov, nodeicov and others.

• Covariates can often partially explain degree heterogeneity and reciprocity.
• Node-level parameters are confounded with node-level covariate effects.
  • two stage approach: fit node-level parameters, and then relate to covariates;
  • random effects model: next lecture.
Summary

• Covariates can be included in the ERGMs.
  • dyad level covariates: `nodematch`, `edgecov` and others;
  • node level covariates: `nodeocov`, `nodeicov` and others.

• Covariates can often partially explain degree heterogeneity and reciprocity.
• Node-level parameters are confounded with node-level covariate effects.
  • two stage approach: fit node-level parameters, and then relate to covariates;
  • random effects model: next lecture.
Summary

• Covariates can be included in the ERGMs.
  • dyad level covariates: `nodematch`, `edgecov` and others;
  • node level covariates: `nodeocov`, `nodeicov` and others.

• Covariates can often partially explain degree heterogeneity and reciprocity.

• Node-level parameters are confounded with node-level covariate effects.
  • two stage approach: fit node-level parameters, and then relate to covariates;
  • random effects model: next lecture.
Summary

- Covariates can be included in the ERGMs.
  - dyad level covariates: `nodematch`, `edgécov` and others;
  - node level covariates: `nodeocov`, `nodeicov` and others.

- Covariates can often partially explain degree heterogeneity and reciprocity.
- Node-level parameters are confounded with node-level covariate effects.
  - two stage approach: fit node-level parameters, and then relate to covariates;
  - random effects model: next lecture.
Summary

- Covariates can be included in the ERGMs.
  - dyad level covariates: `nodematch`, `edgecov` and others;
  - node level covariates: `nodeocov`, `nodeicov` and others.

- Covariates can often partially explain degree heterogeneity and reciprocity.
- Node-level parameters are confounded with node-level covariate effects.
  - two stage approach: fit node-level parameters, and then relate to covariates;
  - random effects model: next lecture.
Summary

• Covariates can be included in the ERGMs.
  • dyad level covariates: `nodematch`, `edgecov` and others;
  • node level covariates: `nodeocov`, `nodeicov` and others.

• Covariates can often partially explain degree heterogeneity and reciprocity.

• Node-level parameters are confounded with node-level covariate effects.
  • two stage approach: fit node-level parameters, and then relate to covariates;
  • random effects model: next lecture.