Stats218 HW2

Peter Racioppo

11/17/2020

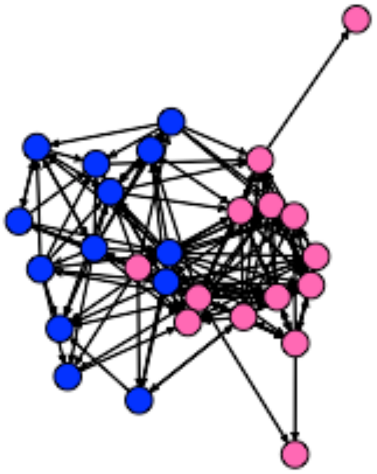
1) Mixing within a Classroom: Here we consider a network of strong friendship ties among 13 boys and 14 girls in a sixth-grade classroom, as collected by Hansell (1984). Each student was asked if they liked each other student “a lot”, “some”, or “not much”. Here we consider a strong friendship tie to exist if a student likes another student “a lot.” Also recorded is the sex of each student. The data for the following questions is in the networkdata package.

```
library(networkdata)
data(hansell)
help(hansell)
```

---

a) Plot the network. Based on the plot alone, does there appear to be a general preference for same-sex friendship ties?

```
# Record sexes in array s
v = hansell$val
s = 1:length(v)*0
for(i in 1:length(v)){
        x=v[[i]]
        if(x$sex == "male"){
                s[i] = 4
        }
        else if(x$sex == "female"){
                s[i] = "hotpink"
        }
        else {
                s[i] = -1}
}

plot.network(hansell, vertex.cex = 3,vertex.col = s)
```



Yes, there does appear to be a preference for same-sex friendship ties, as there are clearly-discernible clusters.

---

b) Fit an Erdos-Rényi model, that is, the homogeneous Bernoulli model, to the network using the ergm command:

```
library(ergm)

## Warning: package 'ergm' was built under R version 3.6.2

##
## ergm: version 3.11.0, created on 2020-10-14
## Copyright (c) 2020, Mark S. Handcock, University of California -- Los
Angeles
##                       David R. Hunter, Penn State University
##                       Carter T. Butts, University of California -- Irvine
##                       Steven M. Goodreau, University of Washington
##                       Pavel N. Krivitsky, UNSW Sydney
##                       Martina Morris, University of Washington
##                       with contributions from
##                       Li Wang
##                       Kirk Li, University of Washington
##                      Skye Bender-deMoll, University of Washington
##                      Chad Klumb
##                      Michal Bojanowski, Kozminski University
##                      Ben Bolker
## Based on "statnet" project software (statnet.org).
## For license and citation information see statnet.org/attribution
## or type citation("ergm").

## NOTE: Versions before 3.6.1 had a bug in the implementation of the bd()
## constraint which distorted the sampled distribution somewhat. In
## addition, Sampson's Monks datasets had mislabeled vertices. See the
## NEWS and the documentation for more details.

## NOTE: Some common term arguments pertaining to vertex attribute and
## level selection have changed in 3.10.0. See terms help for more
## details. Use 'options(ergm.term=list(version="3.9.4"))' to use old
## behavior.

fit.er <- ergm(hansell ~ edges)

## Starting maximum pseudolikelihood estimation (MPLE):

## Evaluating the predictor and response matrix.

## Maximizing the pseudolikelihood.

## Finished MPLE.

## Stopping at the initial estimate.

## Evaluating log-likelihood at the estimate.

# summary(fit.er)
```

Compute, based on the output, the MLE of the log-odds of a tie between two randomly chosen students.

```
beta = fit.er$coef
print(beta)

##    edges
## -1.24454
```

Compute, based on the output, the MLE of the probability of a tie between two randomly chosen students.

```
prob = exp(beta)/(1+exp(beta))
print(prob)

##     edges
## 0.2236467
```

How does it compare to the density of ties in the network?

```
density = 157/27^2
print(density)

## [1] 0.2153635

print(density/prob)

##    edges
## 0.962963
```

The MLE of the probability is almost equal to the network density. (The density is 96% of the MLE probability.)

Does the model fit better than the model that says that all networks among these students are equally likely? Quote a statistical test to support your claim.

Yes, the residual deviance for this model is 746.2 on 701 degrees of freedom, which is far better than the null deviance of 973.2 on 702 degrees of freedom.

There are 2$^{(n}$2) = 2^729 ~= 3.5 x 10^-220 possible graphs between these nodes. If each is drawn with equal probability. There are M x (n^2 choose M) graphs with M edges or less, so the chance of observing a density as low as the observed density is about 1.75 × 10^-54. In other words, the model that says that all networks among these students are equally likely is a poor fit.

If the observed graph is drawn from a graph distribution given by the Erdos-Renyi model with MLE probability p, the probability of observing this graph is p^M x (1-p)^((n choose 2)-M), where n = 27 and M = 157, and p = 0.2236467. This comes to 0.2236^157 x 0.7764^194 ~= 3.5 x 10^-124. Thus, the observed graph is about 10^96 times more likely under our Erdos-Renyi model than a model in which every possible graph is equally likely.

c) Fit a model for homophily of ties by sex using the ergm command:

```
fit.homo <- ergm(hansell~edges + nodematch("sex"))

## Starting maximum pseudolikelihood estimation (MPLE):

## Evaluating the predictor and response matrix.

## Maximizing the pseudolikelihood.

## Finished MPLE.

## Stopping at the initial estimate.
## Evaluating log-likelihood at the estimate.

summary(fit.homo)

## Call:
## ergm(formula = hansell ~ edges + nodematch("sex"))
##
## Iterations:  5 out of 20
##
## Monte Carlo MLE Results:
##               Estimate Std. Error MCMC % z value Pr(>|z|)
## edges          -1.9841     0.1608      0 -12.341   <1e-04 ***
## nodematch.sex   1.2954     0.1979      0   6.547   <1e-04 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##      Null Deviance: 973.2  on 702  degrees of freedom
##  Residual Deviance: 699.1  on 700  degrees of freedom
##
## AIC: 703.1    BIC: 712.2    (Smaller is better.)
```

Based on the output, is there homophily based on sex? Quote a statistical test to support your claim.

Yes. The homophily in this model is 1.2954, which is much larger than the standard error of 0.1979. This model has a residual deviance of 699.1 on 700 degrees of freedom, very close to the expected value of 700 under a Chi-squared distribution. The Erdos-Renyi model has a residual variance of 746.2 on 701 degrees of freedom, which is much worse, although still within a standard deviation of the expected value. The AIC and BIC for this model are both lower than for the Erdos-Renyi model. We can conclude that the model with homophily is a better fit than the Erdos-Renyi model.

Compute, based on the output, the MLE of the log-odds of a tie between two students of the same sex.

```
beta = fit.homo$coef[1] + fit.homo$coef[2]
print(beta)

##      edges
## -0.6887126
```

The MLE of the log-odds of a tie between two students of the same sex is -0.6887126 .

Compute, based on the output, the MLE of the probability of a tie between two students of the same sex.

```
prob = exp(beta)/(1+exp(beta))
print(c("MLE of the probability of a tie between two students of the same 
sex: ",prob))
```

```
## 
## "MLE of the probability of a tie between two students of the same sex: " 
##                                                                    edges 
##                                                      "0.334319526627219" 
```

The MLE of the probability of a tie between two students of the same sex is 0.3343195.

Compute, based on the output, the MLE of the probability of a tie between two students of the opposite sex. Does the model fit better than the Erdos-Rényi model? Quote a statistical test to support your claim.

```
beta = fit.homo$coef[1]
prob = exp(beta)/(1+exp(beta))
print(c("MLE of the probability of a tie between two students of the opposite 
sex: ",prob))
```

```
## 
## "MLE of the probability of a tie between two students of the opposite sex: 
" 
## 
edges 
## 
"0.120879121131478" 
```

```
anova(fit.er,fit.homo)
```

```
## Analysis of Variance Table
## 
## Model 1: hansell ~ edges
## Model 2: hansell ~ edges + nodematch("sex")
##      Df Deviance Resid. Df Resid. Dev Pr(>|Chisq|)    
## NULL                   702     973.18                 
## 1     1  226.974       701     746.20    < 2.2e-16 ***
## 2     1   47.067       700     699.14    6.861e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The MLE of the probability of a tie between two students of the opposite sex is 0.1208791.

Yes, the model fits better than the Erdos-Renyi model. This model has a residual deviance of 699.1 on 700 degrees of freedom, very close to the expected value of 700 under a Chi-squared distribution. The Erdos-Renyi model has a residual variance of 746.2 on 701 degrees of freedom, which is much worse, although still within a standard deviation of the expected value. The AIC and BIC for this model are both lower than for the Erdos-Renyi model. We can conclude that the model with homophily is a better fit than the Erdos-Renyi model.

The observed tie densities are 0.3736264 female-female, 0.1043956 female=male, 0.1373626 male-female, and 0.2884615 male-male.

Thus, the average tie density for students of the same sex is (0.3736264 x 14 x 13 + 0.2884615 x 13 x 12)/(14 x 13+13 x 12) = 0.3343195, which is identical to the MLE value. The average tie density for students of the opposite sex is (0.1043956+0.1373626)/2 = 0.1208791, which is again identical to the MLE value. The Erdos-Renyi model would underestimate the ties between students of the same sex and overestimate ties between students of opposite sex.

---

d) Fit a model for differential homophily of ties by sex using the ergm command:

```
fit.diff.homo <- ergm(hansell~edges + nodematch("sex", diff=TRUE))

## Starting maximum pseudolikelihood estimation (MPLE):

## Evaluating the predictor and response matrix.

## Maximizing the pseudolikelihood.

## Finished MPLE.

## Stopping at the initial estimate.

## Evaluating log-likelihood at the estimate.

summary(fit.diff.homo)

## Call:
## ergm(formula = hansell ~ edges + nodematch("sex", diff = TRUE))
##
## Iterations:  5 out of 20
##
## Monte Carlo MLE Results:
##                      Estimate Std. Error MCMC % z value Pr(>|z|)
## edges                 -1.9841     0.1608      0 -12.341   <1e-04 ***
## nodematch.sex.female   1.4674     0.2221      0   6.607   <1e-04 ***
## nodematch.sex.male     1.0813     0.2389      0   4.526   <1e-04 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##      Null Deviance: 973.2  on 702  degrees of freedom
##  Residual Deviance: 696.4  on 699  degrees of freedom
##
## AIC: 702.4    BIC: 716    (Smaller is better.)

anova(fit.homo,fit.diff.homo)
```

```
## Analysis of Variance Table
##
## Model 1: hansell ~ edges + nodematch("sex")
## Model 2: hansell ~ edges + nodematch("sex", diff = TRUE)
##      Df Deviance Resid. Df Resid. Dev Pr(>|Chisq|)
## NULL                   702     973.18
## 1     2  274.041       700     699.14      < 2e-16 ***
## 2     1    2.752       699     696.39      0.09711 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Based on the output, does the homophily differ between males and females? Quote a statistical test to support your claim.

According to this model, the homophily is 1.4674 for females and 1.0813 for males. The AIC and BIC for this model are both comparable to the model which fits on homophily but not differential homophily. The residual deviance is 699.1 on 700 degrees of freedom for the model with homophily and 696.4 on 699 degrees of freedom for the model with differential homophily, which are also comparable values. Thus, the differential homophily fits prediction of a higher homophily between females than males is not strongly supported by the evidence.

---

e) Fit a model for full mixing between the two sexes using the ergm command:

```
fit.mix <- ergm(hansell~nodemix("sex"))

## Starting maximum pseudolikelihood estimation (MPLE):

## Evaluating the predictor and response matrix.

## Maximizing the pseudolikelihood.

## Finished MPLE.

## Stopping at the initial estimate.

## Evaluating log-likelihood at the estimate.

summary(fit.mix)

## Call:
## ergm(formula = hansell ~ nodemix("sex"))
##
## Iterations:  5 out of 20
##
## Monte Carlo MLE Results:
##                       Estimate Std. Error MCMC % z value Pr(>|z|)
## mix.sex.female.female  -0.5167     0.1532      0  -3.372 0.000746 ***
## mix.sex.male.female    -1.8374     0.2153      0  -8.533  < 1e-04 ***
## mix.sex.female.male    -2.1493     0.2424      0  -8.867  < 1e-04 ***
## mix.sex.male.male      -0.9029     0.1767      0  -5.109  < 1e-04 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##      Null Deviance: 973.2  on 702  degrees of freedom
##  Residual Deviance: 695.5  on 698  degrees of freedom
##
## AIC: 703.5    BIC: 721.7    (Smaller is better.)
```

Compute, based on the output, the MLE of the log-odds of a tie from a male to a female student.

```
beta = fit.mix$coef
```

The MLE of the log-odds of a tie from a male to a female student is -1.8373700.

Compute, based on the output, the MLE of the probability of a tie from a female to a male student.

```
beta = fit.mix$coef
ff_l = -0.5166907
mf_l = -1.8373700
fm_l = -2.1493112
mm_l = -0.9028677
ff_p = exp(ff_l)/(1+exp(ff_l))
fm_p = exp(fm_l)/(1+exp(fm_l))
mf_p = exp(mf_l)/(1+exp(mf_l))
mm_p = exp(mm_l)/(1+exp(mm_l))
print(fm_p)

## [1] 0.1043956
```

The MLE of the probability of a tie from a female to a male student is 0.1043956.

How does this compare to the empirical frequency of such ties?

```
# Record sexes in array s
v = hansell$val
s1 = 1:length(v)*0
for(i in 1:length(v)){
        x=v[[i]]
        if(x$sex == "male"){
                s1[i] = 1
        }
        else if(x$sex == "female"){
                s1[i] = -1
        }
}

adj = as.matrix.network.adjacency(hansell)
ff = 0
fm = 0
mf = 0
mm = 0
for(i in 1:length(v)){
        for(j in 1:length(v)){
                if(adj[i,j]==1){
                        if(s1[i]==-1 && s1[j]==-1){
                                ff = ff + 1}
                        if(s1[i]==-1 && s1[j]==1){
                                fm = fm + 1}
                        if(s1[i]==1 && s1[j]==-1){
                                mf = mf + 1}
                        if(s1[i]==1 && s1[j]==1){
                                mm = mm + 1}
                        }
        }
}
total = ff + fm + mf + mm # Equals total number of ties (157)

# Observed densities:
ff/(14*13)

## [1] 0.3736264

fm/(14*13)

## [1] 0.1043956

mf/(13*14)

## [1] 0.1373626

mm/(13*12)

## [1] 0.2884615
```

The MLE probabilities of ties are 0.3736264 female-female, 0.1043956 female-male, 0.1373626 male-female, and 0.2884615 male-male.

The observed tie densities are 0.3736264 female-female, 0.1043956 female=male, 0.1373626 male-female, and 0.2884615 male-male, exactly the same. The edge densities are predicted much better than in the other models.

Based on the output, does the full mixing model significantly improve over the other models fit? Quote statistical tests to support your claim.

```
anova(fit.er,fit.mix)

## Analysis of Variance Table
##
## Model 1: hansell ~ edges
## Model 2: hansell ~ nodemix("sex")
##      Df Deviance Resid. Df Resid. Dev Pr(>|Chisq|)
## NULL                   702     973.18
## 1     1  226.974       701     746.20    < 2.2e-16 ***
## 2     3   50.752       698     695.45    5.524e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

anova(fit.homo,fit.mix)

## Analysis of Variance Table
##
## Model 1: hansell ~ edges + nodematch("sex")
## Model 2: hansell ~ nodemix("sex")
##      Df Deviance Resid. Df Resid. Dev Pr(>|Chisq|)
## NULL                   702     973.18
## 1     2  274.041       700     699.14       <2e-16 ***
## 2     2    3.686       698     695.45       0.1584
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

anova(fit.diff.homo,fit.mix)

## Analysis of Variance Table
##
## Model 1: hansell ~ edges + nodematch("sex", diff = TRUE)
## Model 2: hansell ~ nodemix("sex")
##      Df Deviance Resid. Df Resid. Dev Pr(>|Chisq|)
## NULL                   702     973.18
## 1     3  276.793       699     696.39       <2e-16 ***
## 2     1    0.933       698     695.45        0.334
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The residual deviance for the full mixing model is 695.5 on 698 degrees of freedom, which is very close to the expected value of 698 under a Chi-squared distribution. However, this isn’t an improvement over the residual deviances for the homophily fits: 699.1 on 700 degrees of freedom for the model with homophily and 696.4 on 699 degrees of freedom for the model with differential homophily. The AIC and BIC for the full mixing model are 703.5 and 721.7, which are actually higher than for the models fit on homophily. Thus, there is not evidence that the full mixing model significantly improves on the other models.

2) Centrality and Prestige with the Holland-Leinhardt p1 model: Here we consider again Hansen's network of strong friendship ties among 13 boys and 14 girls in a sixth-grade classroom. Here we consider a network of strong friendship ties among 13 boys and 14 girls in a sixth- grade classroom, as collected by Hansell (1984). Each student was asked if they liked each other student "a lot", "some", or "not much". Here we consider a strong friendship tie to exist if a student likes another student "a lot." Also recorded is the sex of each student. The data are in the networkdata package

---

a) Fit the p1 model using the ergm command:

```
fit <- ergm(hansell ~ edges + nodematch("sex") + sender + receiver + mutual)

## Observed statistic(s) sender10, sender26, sender27, and receiver20 are at
their smallest attainable values. Their coefficients will be fixed at -Inf.

## Starting maximum pseudolikelihood estimation (MPLE):

## Evaluating the predictor and response matrix.

## Maximizing the pseudolikelihood.

## Finished MPLE.

## Starting Monte Carlo maximum likelihood estimation (MCMLE):

## Iteration 1 of at most 20:

## Optimizing with step length 1.

## The log-likelihood improved by 0.2952.

## Step length converged once. Increasing MCMC sample size.

## Iteration 2 of at most 20:

## Optimizing with step length 1.

## The log-likelihood improved by 0.09137.

## Step length converged twice. Stopping.

## Finished MCMLE.

## Evaluating log-likelihood at the estimate. Using 20 bridges: 1 2 3 4 5 6 7
8 9 10 11 12 13 14 15 16 17 18 19 20 .
## This model was fit using MCMC.  To examine model diagnostics and check
## for degeneracy, use the mcmc.diagnostics() function.

summary(fit)

## Call:
## ergm(formula = hansell ~ edges + nodematch("sex") + sender +
##     receiver + mutual)
```

```
##
## Iterations:  2 out of 20
##
## Monte Carlo MLE Results:
##               Estimate Std. Error MCMC % z value Pr(>|z|)
## edges         -2.13459    0.74860      0  -2.851  0.00435 **
## nodematch.sex  1.84083    0.27192      0   6.770  < 1e-04 ***
## sender2       -1.43267    0.89973      0  -1.592  0.11131
## sender3        0.28463    0.71368      0   0.399  0.69003
## sender4        1.37938    0.69115      0   1.996  0.04596 *
## sender5        1.51129    0.68850      0   2.195  0.02816 *
## sender6        0.92569    0.69933      0   1.324  0.18561
## sender7       -0.49950    0.75548      0  -0.661  0.50850
## sender8        0.34205    0.69722      0   0.491  0.62371
## sender9       -0.93296    0.81828      0  -1.140  0.25422
## sender10          -Inf    0.00000      0    -Inf  < 1e-04 ***
## sender11      -2.14888    1.13786      0  -1.889  0.05896 .
## sender12      -0.88676    0.81976      0  -1.082  0.27937
## sender13      -1.42722    0.90705      0  -1.573  0.11561
## sender14       0.02810    0.77958      0   0.036  0.97125
## sender15      -0.72514    0.81674      0  -0.888  0.37462
## sender16       3.38604    0.78786      0   4.298  < 1e-04 ***
## sender17      -2.98557    1.18388      0  -2.522  0.01167 *
## sender18      -2.11359    1.00859      0  -2.096  0.03612 *
## sender19       1.31316    0.75295      0   1.744  0.08115 .
## sender20       1.30148    0.74650      0   1.743  0.08126 .
## sender21      -0.69845    0.81089      0  -0.861  0.38905
## sender22       0.38436    0.76745      0   0.501  0.61649
## sender23       0.06174    0.77606      0   0.080  0.93660
## sender24      -1.53128    0.90426      0  -1.693  0.09038 .
## sender25       0.30928    0.77523      0   0.399  0.68993
## sender26          -Inf    0.00000      0    -Inf  < 1e-04 ***
## sender27          -Inf    0.00000      0    -Inf  < 1e-04 ***
## receiver2     -0.26452    0.79081      0  -0.334  0.73801
## receiver3     -1.52964    0.90908      0  -1.683  0.09245 .
## receiver4     -1.56486    0.92854      0  -1.685  0.09193 .
## receiver5     -0.14378    0.79837      0  -0.180  0.85708
## receiver6     -0.08506    0.77942      0  -0.109  0.91310
## receiver7     -1.45620    0.89083      0  -1.635  0.10212
## receiver8     -2.10397    1.00143      0  -2.101  0.03564 *
## receiver9     -0.29769    0.80303      0  -0.371  0.71085
## receiver10     0.05306    0.79248      0   0.067  0.94662
## receiver11    -0.96792    0.88544      0  -1.093  0.27433
## receiver12    -1.42233    0.91805      0  -1.549  0.12131
## receiver13    -0.27629    0.80847      0  -0.342  0.73254
## receiver14     0.59291    0.78468      0   0.756  0.44989
## receiver15     1.47091    0.77946      0   1.887  0.05915 .
## receiver16    -0.83792    0.82043      0  -1.021  0.30710
## receiver17     1.18785    0.79792      0   1.489  0.13657
## receiver18     0.91043    0.77347      0   1.177  0.23917
## receiver19    -1.61198    0.96144      0  -1.677  0.09361 .
## receiver20        -Inf    0.00000      0    -Inf  < 1e-04 ***
## receiver21     0.89429    0.78324      0   1.142  0.25355
## receiver22    -0.36676    0.82718      0  -0.443  0.65749
## receiver23    -0.72344    0.85905      0  -0.842  0.39971
## receiver24     0.63541    0.77877      0   0.816  0.41454
## receiver25     0.82872    0.79522      0   1.042  0.29735
## receiver26    -3.09293    1.40614      0  -2.200  0.02784 *
## receiver27    -2.10266    1.10237      0  -1.907  0.05647 .
## mutual         0.79424    0.45639      0   1.740  0.08181 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##      Null Deviance: 973.2  on 702  degrees of freedom
##  Residual Deviance:   NaN  on 647  degrees of freedom
##
## AIC: NaN    BIC: NaN    (Smaller is better.)
##
##  Warning: The following terms have infinite coefficient estimates:
##   sender10 sender26 sender27 receiver20
```

In this model, does there appear to be a general preference for same-sex friendship ties? Does there appear to be a general preference for mutual friendship ties? Quote statistical tests to support your claims.

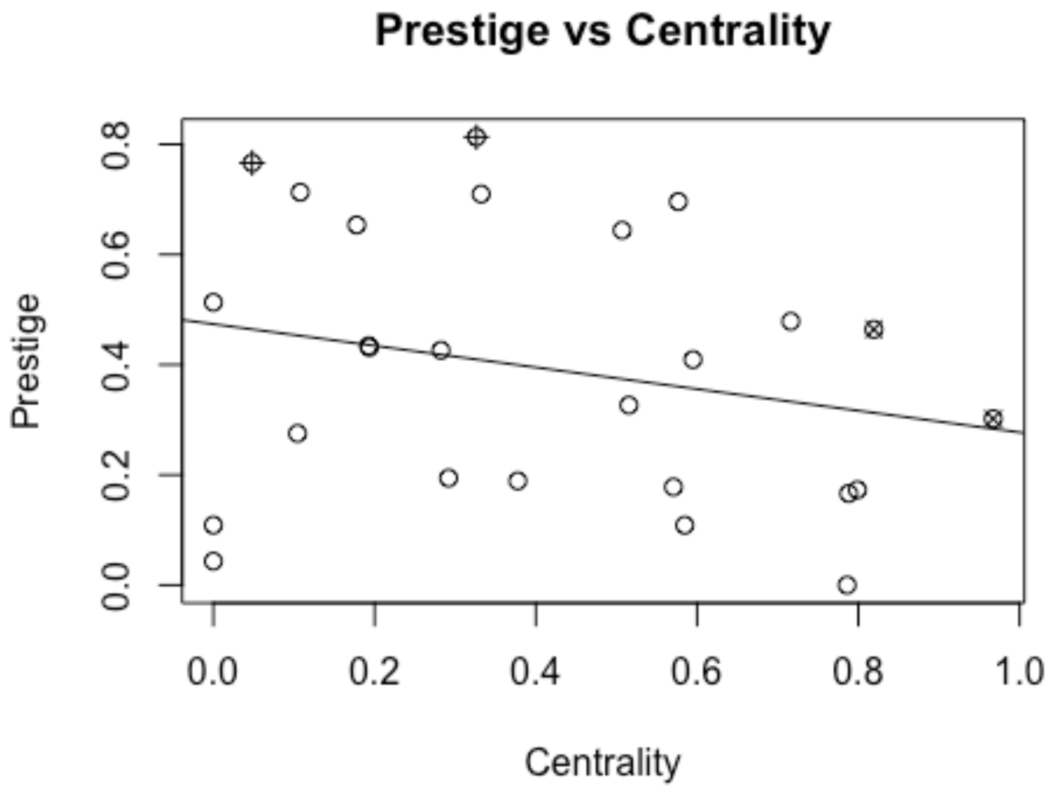
Yes, nodematch.sex = 1.83519 and mutual = 0.82465, both of which are greater than zero.

---

b) We can interpret the sender coefficients as measures of the “centrality” of the actor, and the receiver coefficients as measures of “prestige”. Plot the prestige versus the centrality scores for each node (see fit$coef). Find on the plot the top two students in terms of prestige and the top two in terms of centrality. Is their top rank apparent in the figure? How correlated are these measures to (Freeman’s) in-degree centrality and out-degree prestige, respectively?

```r
# detach(package:igraph, unload = TRUE)
library(sna)

sender = fit$coef[3:28]
receiver = fit$coef[29:54]

p_sender = exp(sender)/(1+exp(sender))
p_receiver = exp(receiver)/(1+exp(receiver))

indegree = degree(hansell,cmode="indegree")
outdegree = degree(hansell,cmode="outdegree")
```

```r
indegree_t = indegree[2:27]
outdegree_t = outdegree[2:27]

prestige1 = which.max(p_receiver)
p_receiver2 = p_receiver[-prestige1]
prestige2 = which.max(p_receiver2)
if(prestige2 >= prestige1){
                prestige2 = prestige2 + 1
        }
centrality1 = which.max(p_sender)
p_sender2 = p_sender[-centrality1]
centrality2 = which.max(p_sender2)
if(centrality2 >= centrality1){
                centrality2 = centrality2 + 1
}

plot(p_sender,p_receiver,main = "Prestige vs Centrality", xlab =
"Centrality", ylab = "Prestige")
lines(p_sender[centrality1],p_receiver[centrality1],pch=4,type="o")
lines(p_sender[centrality2],p_receiver[centrality2],pch=4,type="o")
lines(p_sender[prestige1],p_receiver[prestige1],pch=3,type="o")
lines(p_sender[prestige2],p_receiver[prestige2],pch=3,type="o")
abline(lm(p_receiver ~ p_sender))
```



```
cor1 = cor(outdegree_t,p_sender)
cor2 = cor(indegree_t,p_receiver)

print(c("Correlation between sender coefficients and out-degree = ", cor1))

## [1] "Correlation between sender coefficients and out-degree = "
## [2] "0.969285107611339"

print(c("Correlation between receiver coefficients and in-degree = ", cor2))

## [1] "Correlation between receiver coefficients and in-degree = "
## [2] "0.997104712968917"
```

The two nodes with the highest centrality are indicated with an “x” symbol and the nodes with the highest prestige are indicated with a “+” symbol. The nodes with the highest centrality and prestige are evident on the plot.

These measures are highly correlated with Freeman’s in-degree and out-degree centrality measures. The correlation between between sender coefficients and out-degree is 0.97 and the correlation between receiver coefficients and in-degree is 0.99.

c) Consider the typical magnitude of the sender and receiver coefficients. How do they compare in magnitude to the preferences for same-sex or mutual ties?

```
sex = fit$coef[2]
mutual = fit$coef[55]
sender = fit$coef[3:28]
receiver = fit$coef[29:54]

sender1 = sender
receiver1 = receiver
sender1[!is.finite(sender1)] <- NA
sender1 <- sender1[!is.na(sender1)]
receiver1[!is.finite(receiver1)] <- NA
receiver1 <- receiver1[!is.na(receiver1)]

print(c("The mean of the magnitudes of the non-infinite sender nodes is: ",
mean(abs(sender1))))
```

```
## [1] "The mean of the magnitudes of the non-infinite sender nodes is: "
## [2] "1.1569214789302"
```

```
print(c("The mean of the magnitudes of the non-infinite receiver nodes is: ",
mean(abs(receiver1))))
```

```
## [1] "The mean of the magnitudes of the non-infinite receiver nodes is: "
## [2] "1.01686211663833"
```

```
lower = sex - mutual
upper = sex + mutual
abs(sender + receiver)
```

```
##    sender2    sender3    sender4    sender5    sender6    sender7 
sender8 
## 1.69718636 1.24501032 0.18548734 1.36750603 0.84062615 1.95569919 
1.76192093 
##    sender9   sender10   sender11   sender12   sender13   sender14 
sender15 
## 1.23065205        Inf 3.11679521 2.30908899 1.70351947 0.62100382 
0.74577469 
##   sender16   sender17   sender18   sender19   sender20   sender21 
sender22 
## 2.54811359 1.79771399 1.20315303 0.29882149        Inf 0.19584014 
0.01760095 
##   sender23   sender24   sender25   sender26   sender27 
## 0.66170709 0.89586460 1.13800272        Inf        Inf
```

nodematch.sex = 1.83519 and mutual = 0.82465 The means of the magnitudes of the non-infinite sender and receiver nodes are on the same order of magnitude as the same-sex and mutual tie coefficients.

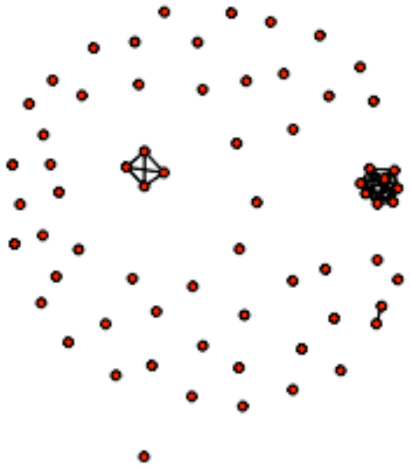
With a few exceptions, the magnitude of the sum of the sender and receiver coefficients appears to be within the range given by the same-sex tie coefficient plus or minus the mutual tie coefficient.

---

3) Cold war connections: The dataset coldwar contains information on military cooperation and conflict between countries from 1950 through 1985. For this analysis, we will only consider cooperative relations (so all negative relations in the sociomatrix should be set to zero).

```
library(networkdata)
data(coldwar)

# coldwar$cc[,,1] - t(coldwar$cc[,,1]) # Check that the matrix is symmetric

# Basic function to create network objects
f_Network <- function(data,dir,mat_type){
  net <- as.network(x = data, # the network object
                 directed = dir, # specify whether the network is directed
                 loops = FALSE, # do we allow self ties (should not allow
them)
                 matrix.type = mat_type # the type of input
                    )
}

library(igraph)

## Warning: package 'igraph' was built under R version 3.6.2

## 
## Attaching package: 'igraph'

## The following objects are masked from 'package:sna':
## 
##     betweenness, bonpow, closeness, components, degree, dyad.census,
##     evcent, hierarchy, is.connected, neighborhood, triad.census

## The following objects are masked from 'package:network':
## 
##     %c%, %s%, add.edges, add.vertices, delete.edges, delete.vertices,
##     get.edge.attribute, get.edges, get.vertex.attribute, is.bipartite,
##     is.directed, list.edge.attributes, list.vertex.attributes,
##     set.edge.attribute, set.vertex.attribute

## The following objects are masked from 'package:stats':
## 
##     decompose, spectrum

## The following object is masked from 'package:base':
## 
##     union
```

```
adj = coldwar$cc==1
cw_ig_1 = graph_from_adjacency_matrix(adj[,,1])
cw_net_1 = f_Network(adj[,,1],FALSE,"adjacency")
plot(cw_net_1, vertex.cex = 1)
```



---

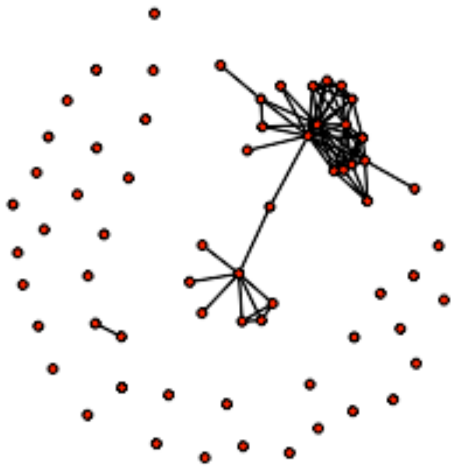
a) For each of the eight years in the dataset, identify and list the countries in the two largest connected components of that year's cooperation graph. Summarize the results with a few sentences.

```
f_CC <- function(g){
        components <- igraph::clusters(g, mode="weak")
        id1 <- which.max(components$csize)
        components2 = components$csize[-id1]
        id2 = which.max(components2)
        if(id2 >= id1){
                id2 = id2 + 1
        }
        sz1 = components$csize[id1]
        sz2 = components$csize[id2]
        vert_ids1 <- V(g)[components$membership == id1]
        vert_ids2 <- V(g)[components$membership == id2]
        ig1 = igraph::induced_subgraph(g, vert_ids1)
        ig2 = igraph::induced_subgraph(g, vert_ids2)
        return(list(id1,id2,vert_ids1,vert_ids2,ig1,ig2,sz1,sz2))
}

for(i in 1:8){
        g = graph_from_adjacency_matrix(adj[,,i])
        x = f_CC(g)
        print(c("Year: ",i))
        print(c("The size of the largest connected component is: ",x[[7]]))
        print(c("The size of the second largest connected component is:
",x[[8]]))
        print(c("The countries in the largest connected component are:
",x[[3]]))
        print(c("The countries in the second largest connected component are:
",x[[4]]))
}

## [1] "Year: " "1"
## [1] "The size of the largest connected component is: "
## [2] "8"
## [1] "The size of the second largest connected component is: "
## [2] "4"
##
## "The countries in the largest connected component are: "
##                                                      AUL
##                                                      "4"
##                                                      CAN
##                                                      "9"
##                                                      NEW
##                                                     "41"
##                                                      PHI
##                                                     "48"
##                                                      ROK
##                                                     "51"
##                                                      TUR
##                                                     "61"
##                                                      UKG
##                                                     "62"
##                                                      USA
##                                                     "63"
##
## "The countries in the second largest connected component are: "
##                                                             BUL
##                                                             "8"
##                                                             HUN
##                                                            "28"
##                                                             RUM
##                                                            "52"
##                                                             USR
##                                                            "64"
## [1] "Year: " "2"
## [1] "The size of the largest connected component is: "
## [2] "3"
## [1] "The size of the second largest connected component is: "
## [2] "1"
##
## "The countries in the largest connected component are: "
##                                                      ROK
##                                                     "51"
##                                                      TAW
##                                                     "59"
##                                                      USA
##                                                     "63"
##
## "The countries in the second largest connected component are: "
##                                                             AFG
##                                                             "1"
## [1] "Year: " "3"
## [1] "The size of the largest connected component is: "
## [2] "4"
## [1] "The size of the second largest connected component is: "
## [2] "2"
##
## "The countries in the largest connected component are: "
##                                                      NOR
##                                                     "43"
##                                                      ROK
##                                                     "51"
##                                                      TUR
##                                                     "61"
##                                                      USA
##                                                     "63"
##
## "The countries in the second largest connected component are: "
##                                                             CUB
##                                                            "14"
##                                                             USR
##                                                            "64"
## [1] "Year: " "4"
## [1] "The size of the largest connected component is: "
## [2] "5"
## [1] "The size of the second largest connected component is: "
## [2] "1"
##
## "The countries in the largest connected component are: "
##                                                      AUL
##                                                      "4"
##                                                      NEW
##                                                     "41"
##                                                      ROK
##                                                     "51" 
##                                                      UKG 
##                                                     "62" 
##                                                      USA 
##                                                     "63" 
## 
## "The countries in the second largest connected component are: " 
##                                                             AFG 
##                                                             "1" 
## [1] "Year: " "5"
## [1] "The size of the largest connected component is: "
## [2] "9"
## [1] "The size of the second largest connected component is: "
## [2] "3"
## 
## "The countries in the largest connected component are: " 
##                                                      AUL 
##                                                      "4" 
##                                                      ISR 
##                                                     "34" 
##                                                      JOR 
##                                                     "36" 
##                                                      NEW 
##                                                     "41" 
##                                                      PHI 
##                                                     "48" 
##                                                      ROK 
##                                                     "51" 
##                                                      THI 
##                                                     "60" 
##                                                      UKG 
##                                                     "62" 
##                                                      USA 
##                                                     "63" 
## 
## "The countries in the second largest connected component are: " 
##                                                             EGY 
##                                                            "19" 
##                                                             IRQ 
##                                                            "33" 
##                                                             USR 
##                                                            "64" 
## [1] "Year: " "6"
## [1] "The size of the largest connected component is: "
## [2] "2"
## [1] "The size of the second largest connected component is: "
## [2] "1"
## 
## "The countries in the largest connected component are: " 
##                                                      ROK 
##                                                     "51"
##                                                      USA
##                                                     "63"
## 
## "The countries in the second largest connected component are: "
##                                                             AFG
##                                                             "1"
## [1] "Year: " "7"
## [1] "The size of the largest connected component is: "
## [2] "8"
## [1] "The size of the second largest connected component is: "
## [2] "2"
## 
## "The countries in the largest connected component are: "
##                                                      CAN
##                                                      "9"
##                                                      GFR
##                                                     "23"
##                                                      ITA
##                                                     "35"
##                                                      NOR
##                                                     "43"
##                                                      NTH
##                                                     "44"
##                                                      ROK
##                                                     "51"
##                                                      UKG
##                                                     "62"
##                                                      USA
##                                                     "63"
## 
## "The countries in the second largest connected component are: "
##                                                             ISR
##                                                            "34"
##                                                             JOR
##                                                            "36"
## [1] "Year: " "8"
## [1] "The size of the largest connected component is: "
## [2] "7"
## [1] "The size of the second largest connected component is: "
## [2] "2"
## 
## "The countries in the largest connected component are: "
##                                                      EGY
##                                                     "19"
##                                                      GFR
##                                                     "23"
##                                                      HON
##                                                     "27"
##                                                      ROK
##                                                      "51" 
##                                                       SAU 
##                                                      "55" 
##                                                       UKG 
##                                                      "62" 
##                                                       USA 
##                                                      "63" 
## 
## "The countries in the second largest connected component are: " 
##                                                               GDR 
##                                                              "22" 
##                                                               USR 
##                                                              "64"
```

---

b) From the socioarray of cooperative events, create a relation that indicates whether or not two countries had any cooperative events during the time periods in the dataset. Make a plot of the graph for this relation, describe the connected components and compare and contrast the components with the those in (a).

```
coop = adj[,,1]
for(i in 2:8){
        coop = coop + adj[,,i]
}
coop = coop > 0

coop_ig = graph_from_adjacency_matrix(coop)
coop_net = f_Network(coop,FALSE,"adjacency")
plot(coop_net, vertex.cex = 1)
```



```
x = f_CC(coop_ig)

print(c("The size of the largest connected component is:",x[[7]]))

## [1] "The size of the largest connected component is:"
## [2] "27"

print(c("The size of the second largest connected component is:",x[[8]]))

## [1] "The size of the second largest connected component is:"
## [2] "2"

print(c("The countries in the largest connected component are:",x[[3]]))

## 
## "The countries in the largest connected component are:" 
##                                                     AUL 
##                                                     "4" 
##                                                     BUL 
##                                                     "8" 
##                                                     CAN 
##                                                     "9" 
##                                                     CUB 
##                                                    "14" 
##                                                     EGY
##                                                    "19"
##                                                     GDR
##                                                    "22"
##                                                     GFR
##                                                    "23"
##                                                     HON
##                                                    "27"
##                                                     HUN
##                                                    "28"
##                                                     IRQ
##                                                    "33"
##                                                     ISR
##                                                    "34"
##                                                     ITA
##                                                    "35"
##                                                     JOR
##                                                    "36"
##                                                     LEB
##                                                    "38"
##                                                     NEW
##                                                    "41"
##                                                     NOR
##                                                    "43"
##                                                     NTH
##                                                    "44"
##                                                     PHI
##                                                    "48"
##                                                     ROK
##                                                    "51"
##                                                     RUM
##                                                    "52"
##                                                     SAU
##                                                    "55"
##                                                     TAW
##                                                    "59"
##                                                     THI
##                                                    "60"
##                                                     TUR
##                                                    "61"
##                                                     UKG
##                                                    "62"
##                                                     USA
##                                                    "63"
##                                                     USR
##                                                    "64"
```

```
print(c("The countries in the second largest connected component
are:",x[[4]]))
```

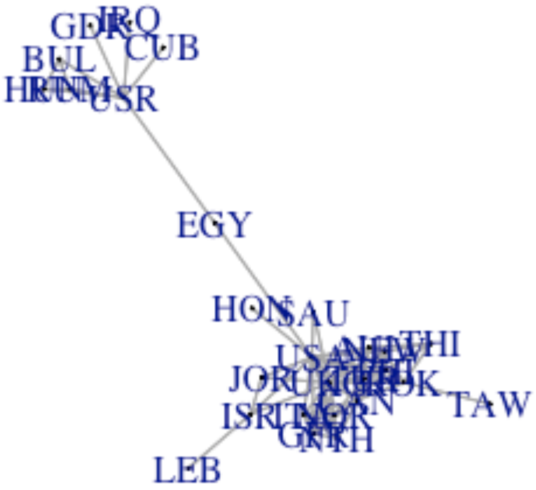
```
##
## "The countries in the second largest connected component are:"
##                                                                 CHN
##                                                                "11"
##                                                                 PRK
##                                                                "50"
```

The largest connected component in the combined graph is three times larger than the largest in any single year. This is because it contains the union of all of the connected components. The second largest connected component in the combined graph has only two vertices, which is smaller than in most years. This is because most of the connected vertices have been absorbed into the giant component.

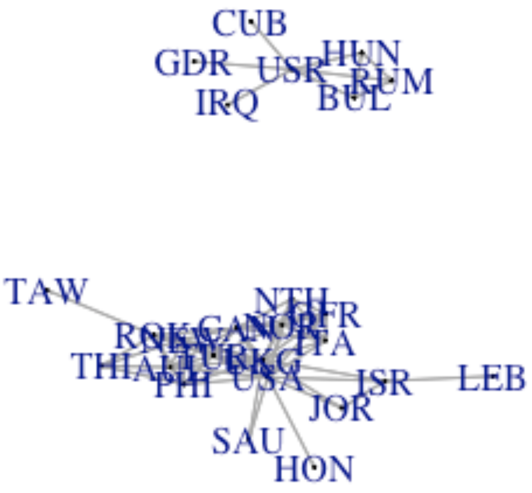
---

c) Identify a cutpoint of the large connected component in (b) so that its removal generates two components that more or less reflect the groupings you saw in (a). For each of these components, compute the closeness centralities of each node and identify the most central node (in each component separately).

```
lcc = x[[5]]
plot(lcc,vertex.size=1,edge.arrow.size=1/10,edge.arrow.width=1/10,edge.width=
1)
```



```
lcc2 = delete.vertices(lcc, "EGY")
plot(lcc2,vertex.size=1,edge.arrow.size=1/10,edge.arrow.width=1/10,edge.width
=1)
```



```
x = f_CC(lcc2)

print(c("The size of the largest connected component is:",x[[7]]))

## [1] "The size of the largest connected component is:"
## [2] "19"

print(c("The size of the second largest connected component is:",x[[8]]))

## [1] "The size of the second largest connected component is:"
## [2] "7"

cc1 = x[[5]]
cc2 = x[[6]]

print(c("The betweenness centralities for the largest connected component
are:",closeness(cc1)))

##
## "The betweenness centralities for the largest connected component are:"
##                                                                       AUL
##                                                      "0.0344827586206897"
##                                                                       CAN
##                                                      "0.0384615384615385"
##                                                                       GFR
##                                                      "0.0303030303030303"
##                                                                       HON
##                                                      "0.0263157894736842"
##                                                                       ISR
##                                                      "0.0294117647058824"
##                                                                       ITA
##                                                      "0.0303030303030303"
##                                                                       JOR
##                                                      "0.0285714285714286"
##                                                                       LEB
##                                                      "0.0196078431372549"
##                                                                       NEW
##                                                      "0.0344827586206897"
##                                                                       NOR
##                                                                 "0.03125"
##                                                                       NTH
##                                                      "0.0303030303030303"
##                                                                       PHI
##                                                      "0.0344827586206897"
##                                                                       ROK
##                                                      "0.0357142857142857"
##                                                                       SAU
##                                                       "0.027027027027027"
##                                                                       TAW
##                                                      "0.0222222222222222"
##                                                                       THI
##                                                      "0.0238095238095238"
##                                                                       TUR
##                                                      "0.0344827586206897"
##                                                                       UKG
##                                                      "0.0454545454545455"
##                                                                       USA
##                                                      "0.0476190476190476"
```

```
print(c("The most central country in the largest connected component
is:",which.max(closeness(cc1))))
```

```
## 
## "The most central country in the largest connected component is:" 
##                                                                 USA 
##                                                                "19"
```

```
print(c("The betweenness centralities for the second largest connected
component are:",closeness(cc2)))
```

```
##
## "The betweenness centralities for the second largest connected component
are:"
##
BUL
##
"0.111111111111111"
##
CUB
##
"0.0909090909090909"
##
GDR
##
"0.0909090909090909"
##
HUN
##
"0.111111111111111"
##
IRQ
##
"0.0909090909090909"
##
RUM
##
"0.111111111111111"
##
USR
##
"0.166666666666667"
```

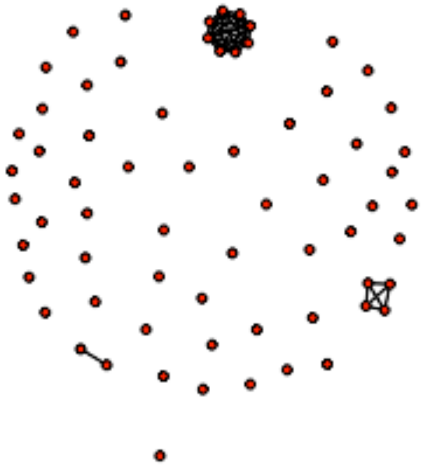
```
print(c("The most central country in the second largest connected component
is:",which.max(closeness(cc2))))
```

```
##
## "The most central country in the second largest connected component is:"
##                                                                       USR
##                                                                       "7"
```

---

4) Cold war testing: For the same dataset coldwar from the last question, test the hypothesis that the first military cooperation network (i.e., 1950-1954) follows a simple random graph model (SRG). Do this using the same degree centralization statistic as in the lectures. Note that this network is undirected, rather than directed as in class.

```
adj = coldwar$cc==1
cw_net_1 = f_Network(adj[,,1],FALSE,"adjacency")
plot(cw_net_1, vertex.cex = 1)
```



```
library(ergm)

fit <- ergm(cw_net_1 ~ edges)

## Starting maximum pseudolikelihood estimation (MPLE):

## Evaluating the predictor and response matrix.

## Maximizing the pseudolikelihood.

## Finished MPLE.

## Stopping at the initial estimate.

## Evaluating log-likelihood at the estimate.

summary(fit)

## Call:
## ergm(formula = cw_net_1 ~ edges)
##
## Iterations:  7 out of 20
##
## Monte Carlo MLE Results:
##       Estimate Std. Error MCMC % z value Pr(>|z|)
## edges  -4.0991     0.1704      0  -24.05   <1e-04 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##      Null Deviance: 2973.6  on 2145  degrees of freedom
##  Residual Deviance:  357.5  on 2144  degrees of freedom
##
## AIC: 359.5    BIC: 365.2    (Smaller is better.)
```

The residual deviance for the Erdos-Renyi model is 357.5 on 2144 degrees of freedom. In other words, it is very unlikely that the observed dataset was generated by this model. In fact, this model is much worse than the model in which every graph is equally likely. It is also unlikely that this dataset was generated by a model in which every graph is equally likely, since the null deviance is 2973.6 under 2145 degrees of freedom, which is far from the expected value under a chi-squared distribution of 2145.