Project

Julian Gullett

5/26/2021

```
library(sna)
```

```
## Loading required package: statnet.common
```

```
##
## Attaching package: 'statnet.common'
```

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

```
## Loading required package: network
```

```
## network: Classes for Relational Data
## Version 1.16.1 created on 2020-10-06.
## copyright (c) 2005, Carter T. Butts, University of California-Irvine
##                     Mark S. Handcock, University of California -- Los Angeles
##                     David R. Hunter, Penn State University
##                     Martina Morris, University of Washington
##                     Skye Bender-deMoll, University of Washington
##  For citation information, type citation("network").
##  Type help("network-package") to get started.
```

```
## sna: Tools for Social Network Analysis
## Version 2.6 created on 2020-10-5.
## copyright (c) 2005, Carter T. Butts, University of California-Irvine
##  For citation information, type citation("sna").
##  Type help(package="sna") to get started.
```

```
library(igraph)
```

```
##
## Attaching package: 'igraph'
```

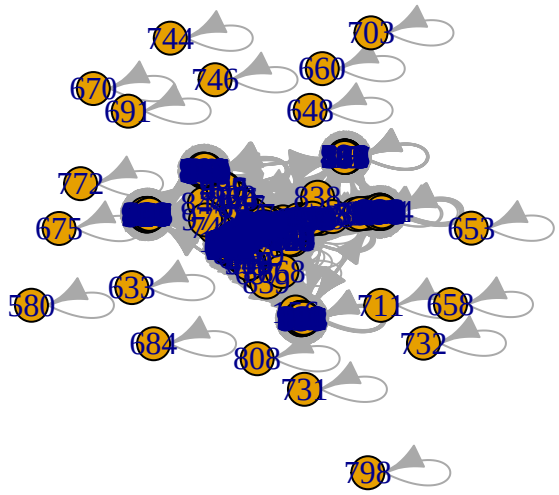
```
## The following objects are masked from 'package:sna':
##
##     betweenness, bonpow, closeness, components, degree, dyad.census,
##     evcent, hierarchy, is.connected, neighborhood, triad.census
```

```
## 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
```

```
edgelist <- scan('C:/Users/Julian/Documents/Dartmouth/Terms/Spring Term/QBS 122/Project/email-Eu-core.t
edgelist <- matrix(edgelist, ncol = 2, byrow = TRUE)

nodes <- unique(c(edgelist[,1],edgelist[,2]))
gnet <- graph_from_data_frame(d=edgelist, vertices=nodes, directed=TRUE)

adjmat <- matrix(0, nrow=length(nodes), ncol=length(nodes))

for (row in 1:nrow(edgelist)) {

  i = edgelist[row,1]
  j = edgelist[row,2]

  adjmat[i,j] <- 1

}

sum(adjmat)
```

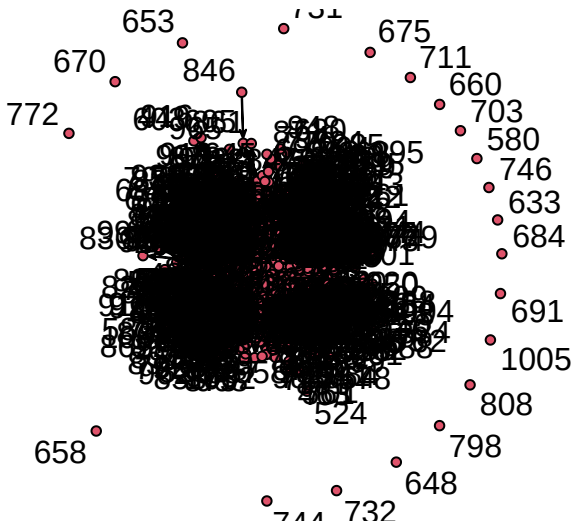
```
## [1] 25499
```

```
plot(gnet,mode="fruchtermanreingold",displaylabels=T)
```



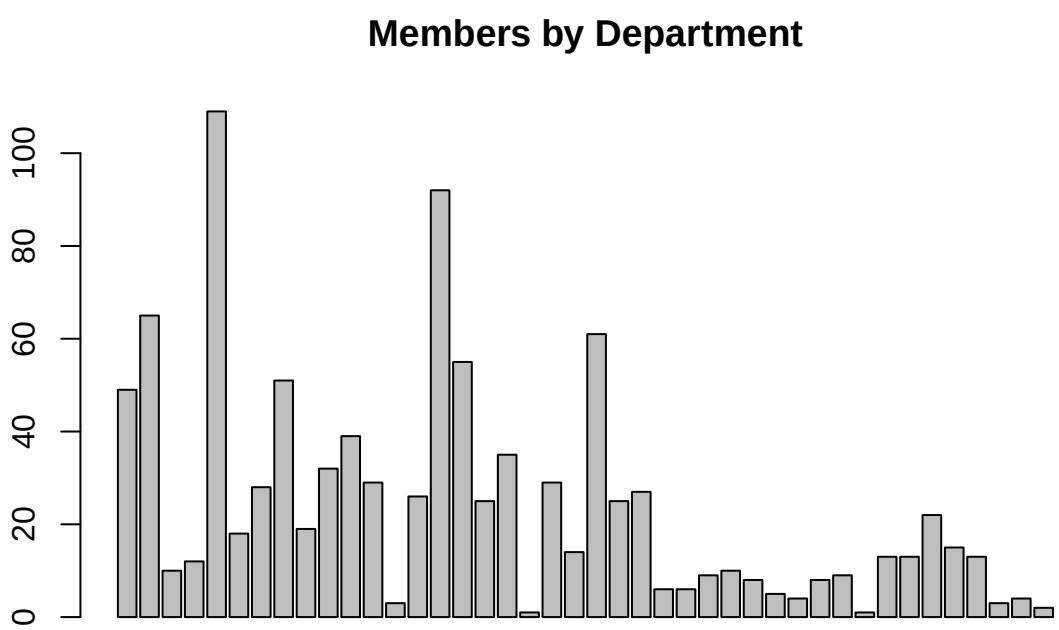

```
nodedata <- scan('C:/Users/Julian/Documents/Dartmouth/Terms/Spring Term/QBS 122/Project/email-Eu-core-d

nodedata <- matrix(nodedata,nrow=length(nodes),byrow=T)

colnames(nodedata) <- c('Node', 'DepartmentID')

nodedata <- data.frame(nodedata)

detach("package:igraph", unload = TRUE)
```

```
nodecov <- list(department=nodedata[,2])

pnet <- network(adjmat,directed=TRUE,matrixtype="adjacency",
 vertex.attr=nodecov,
 vertex.attrnames=c("DepartmentID"))
plot(pnet,mode="fruchtermanreingold",displaylabels=T)
```



This is not a very pleasant plot of the network.

```
depsize <- data.frame(table(nodedata$DepartmentID))
colnames(depsize) <- c("DepartmentID", 'Members')

barplot(depsize$Members, main="Members by Department")
```




```
nodedata$DepSize <- 0

for (row in 1:nrow(nodedata)) {

  nodedata[row, "DepSize"] <- depsize[nodedata[row,"DepartmentID"]+1, "Members"]

}
```

I thought this plot would be helpful to visualize the sizes of each department. Each department has their own bar, and we can see a few outliers. A couple department have >80 members and ~15 have <10 members.

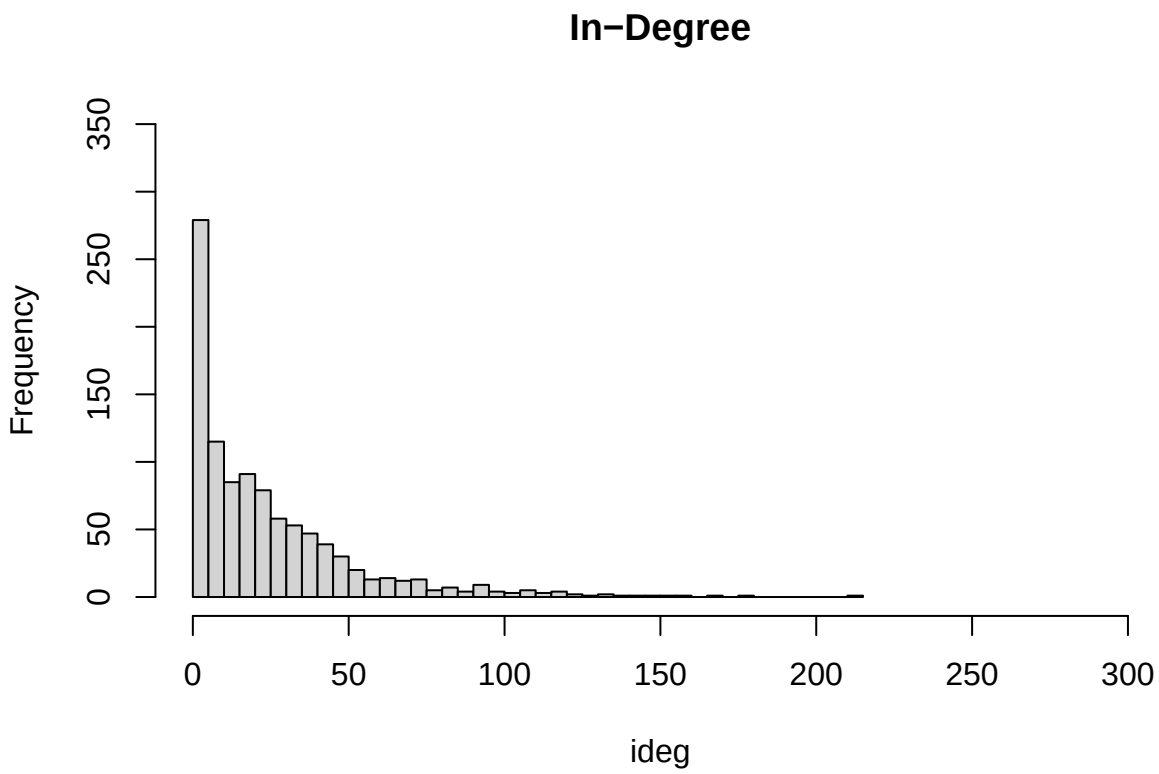
```
ideg <- degree(adjmat, cmode='indegree')
odeg <- degree(adjmat, cmode='outdegree')

nodedata$IDeg <- ideg
nodedata$ODeg <- odeg

summary(ideg)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    5.00   17.00   24.73   35.00  211.00
```

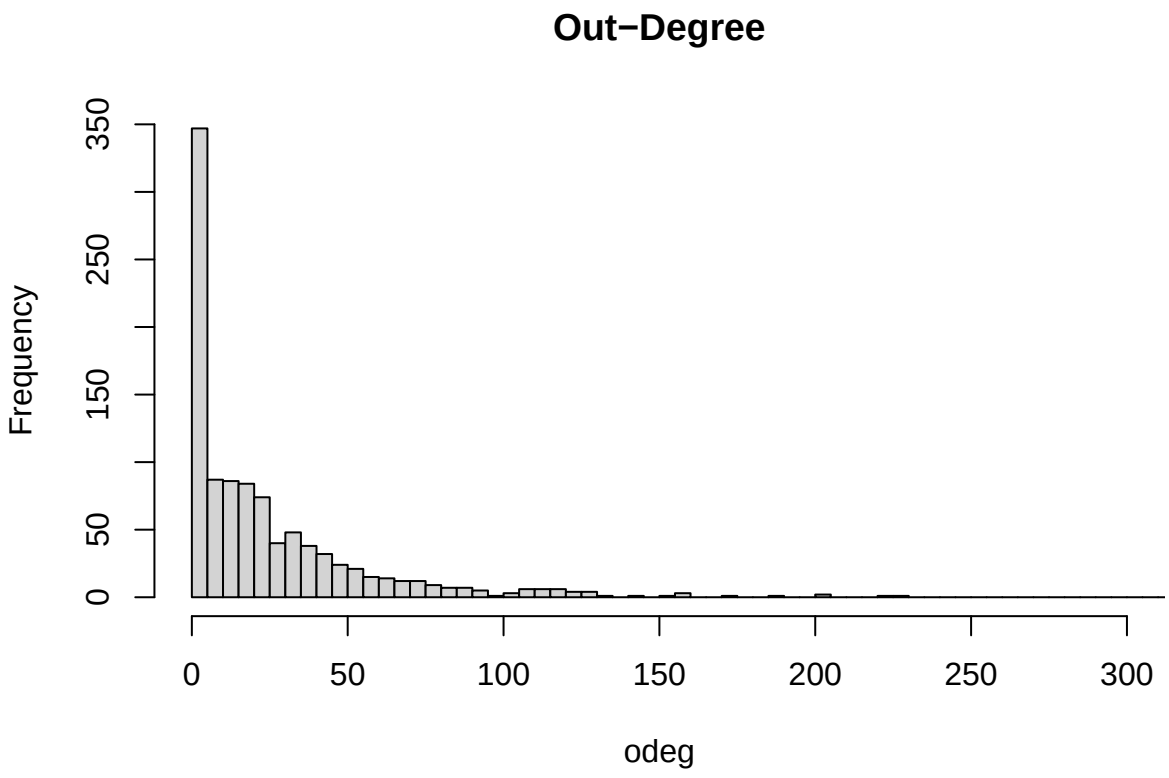
```
summary(odeg)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    1.00   14.00   24.73   34.00  333.00
```

```
hist(ideg, breaks=50, xlim=c(0,300), ylim=c(0,350), main='In-Degree')
```




```
hist(odeg, breaks=50, xlim=c(0,300), ylim=c(0,350), main='Out-Degree')
```




```
length(which(ideg==0))
```

```
## [1] 41
```

```
length(which(odeg==0))
```

```
## [1] 182
```

The degree distributions of this network are shown above. There are 41 nodes with an in-degree of 0, and 182 nodes with an out-degree of 0.

```
nodedata$WithinDepartmentIDeg <- 0
nodedata$WithinDepartmentODeg <- 0

for (i in unique(nodedata$DepartmentID)) {

  idx <- which(nodedata$DepartmentID==i)
  subnet <- adjmat[idx, idx]

  nodedata[idx, 'WithinDepartmentIDeg'] <- degree(subnet, cmode='indegree')
  nodedata[idx, 'WithinDepartmentODeg'] <- degree(subnet, cmode='outdegree')

}
```

```
## Error in as.edgelist.sna(dat): as.edgelist.sna input must be an adjacency matrix/array, edgelist mati
```

```
nodedata$IDegRatio <- nodedata$WithinDepartmentIDeg/nodedata$IDeg
nodedata$ODegRatio <- nodedata$WithinDepartmentODeg/nodedata$ODeg

nodedata[is.nan(nodedata$IDegRatio), "IDegRatio"] <- 0
nodedata[is.nan(nodedata$ODegRatio), "ODegRatio"] <- 0
```

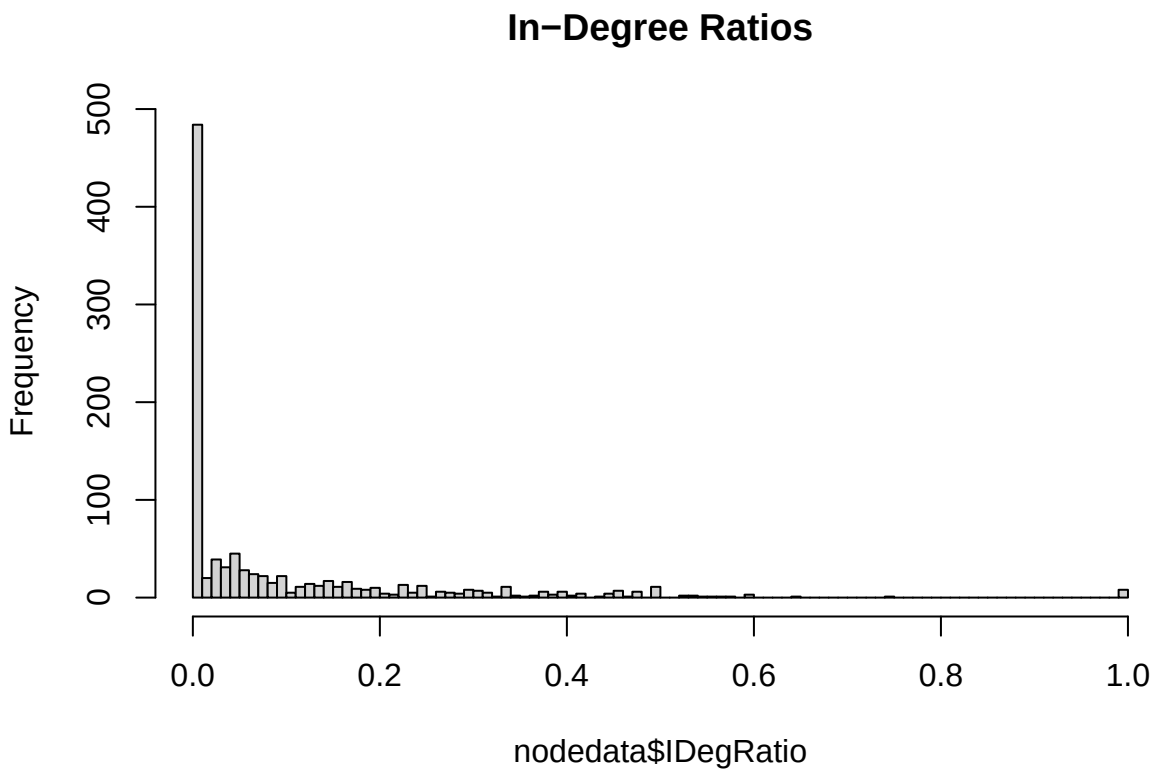
```
length(which(nodedata$IDegRatio==0))
```

```
## [1] 482
```

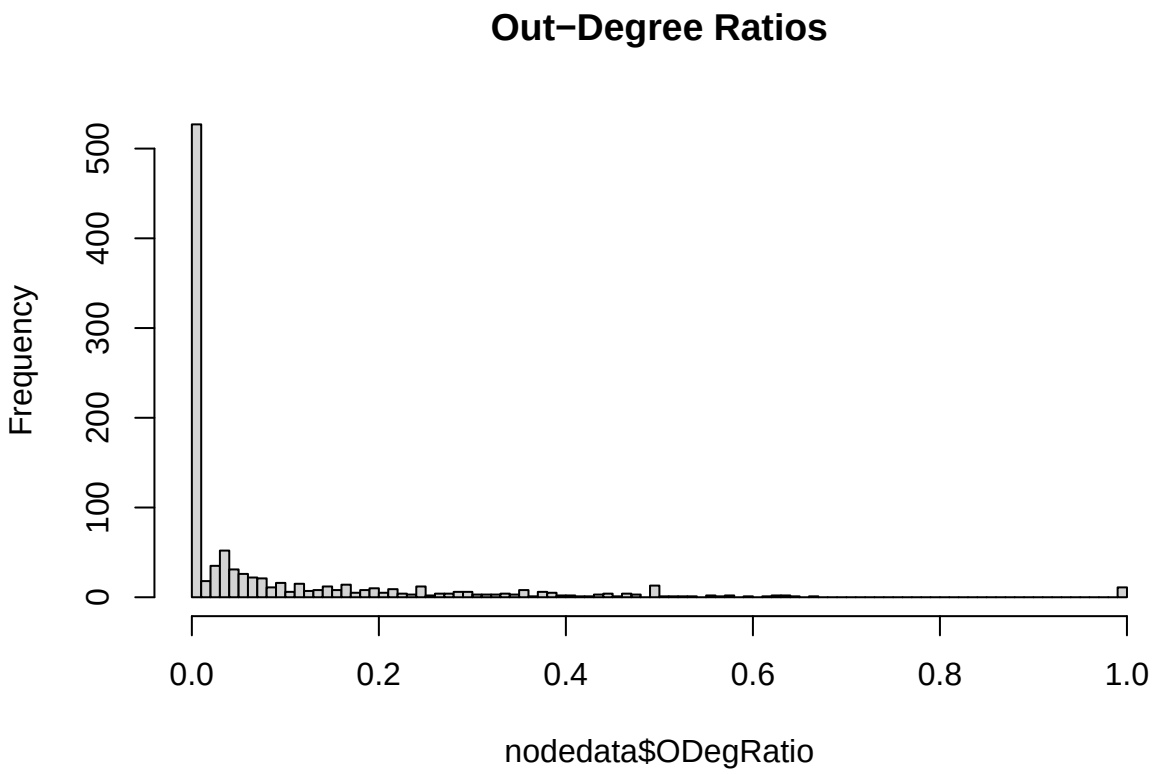
```
length(which(nodedata$ODegRatio==0))
```

```
## [1] 524
```

```
hist(nodedata$IDegRatio, breaks=100, main='In-Degree Ratios')
```




```
hist(nodedata$ODegRatio, breaks=100, main='Out-Degree Ratios')
```



The last few chunks have been focused on degree statistics within departments. Overall, these histograms show that most nodes in this network have their edges with other nodes that are not within their department. 482 nodes have a within-department in-degree of 0, and 524 nodes have a within-department out-degree of 0.

```
dyads <- dyad.census(adjmat)
print(dyads)
```

```
##       Mut Asym   Null
## [1,] 8836 7186 488488
```

```
#triads <- triad.census(adjmat,g="digraph")
#print(triads)

density <- (dyads[1]*2 + dyads[2]) / sum(dyads)
print(density)
```

```
## [1] 0.04927157
```

```
library(statnet)
```

```
## Warning: package 'statnet' was built under R version 4.0.5
```

```
## Loading required package: tergm
```

```
## Loading required package: ergm

##
## 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.

## Loading required package: networkDynamic

##
## networkDynamic: version 0.10.1, created on 2020-01-16
## Copyright (c) 2020, Carter T. Butts, University of California -- Irvine
##                     Ayn Leslie-Cook, University of Washington
##                     Pavel N. Krivitsky, University of Wollongong
##                     Skye Bender-deMoll, University of Washington
##                     with contributions from
##                     Zack Almquist, University of California -- Irvine
##                     David R. Hunter, Penn State University
##                     Li Wang
##                     Kirk Li, University of Washington
##                     Steven M. Goodreau, University of Washington
##                     Jeffrey Horner
##                     Martina Morris, University of Washington
## Based on "statnet" project software (statnet.org).
## For license and citation information see statnet.org/attribution
## or type citation("networkDynamic").

##
```

```
## tergm: version 3.7.0, created on 2020-10-15
## Copyright (c) 2020, Pavel N. Krivitsky, UNSW Sydney
##                     Mark S. Handcock, University of California -- Los Angeles
##                     with contributions from
##                     David R. Hunter, Penn State University
##                     Steven M. Goodreau, University of Washington
##                     Martina Morris, University of Washington
##                     Nicole Bohme Carnegie, New York University
##                     Carter T. Butts, University of California -- Irvine
##                     Ayn Leslie-Cook, University of Washington
##                     Skye Bender-deMoll
##                     Li Wang
##                     Kirk Li, University of Washington
##                     Chad Klumb
## Based on "statnet" project software (statnet.org).
## For license and citation information see statnet.org/attribution
## or type citation("tergm").

## Loading required package: ergm.count

## Warning: package 'ergm.count' was built under R version 4.0.5

##
## ergm.count: version 3.4.0, created on 2019-05-15
## Copyright (c) 2019, Pavel N. Krivitsky, University of Wollongong
##                     with contributions from
##                     Mark S. Handcock, University of California -- Los Angeles
##                     David R. Hunter, Penn State University
## Based on "statnet" project software (statnet.org).
## For license and citation information see statnet.org/attribution
## or type citation("ergm.count").

## NOTE: The form of the term 'CMP' has been changed in version 3.2 of
## 'ergm.count'. See the news or help('CMP') for more information.

## Loading required package: tsna

## Warning: package 'tsna' was built under R version 4.0.5

##
## statnet: version 2019.6, created on 2019-06-13
## Copyright (c) 2019, 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, University of Wollongong
##                     Skye Bender-deMoll
##                     Martina Morris, University of Washington
## Based on "statnet" project software (statnet.org).
## For license and citation information see statnet.org/attribution
## or type citation("statnet").

## unable to reach CRAN
```

```r
library(coda)
```

```r
model5 <- ergmMPLE(pnet~edges + mutual + ttriple,output="fit")
```

```
## 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.
```

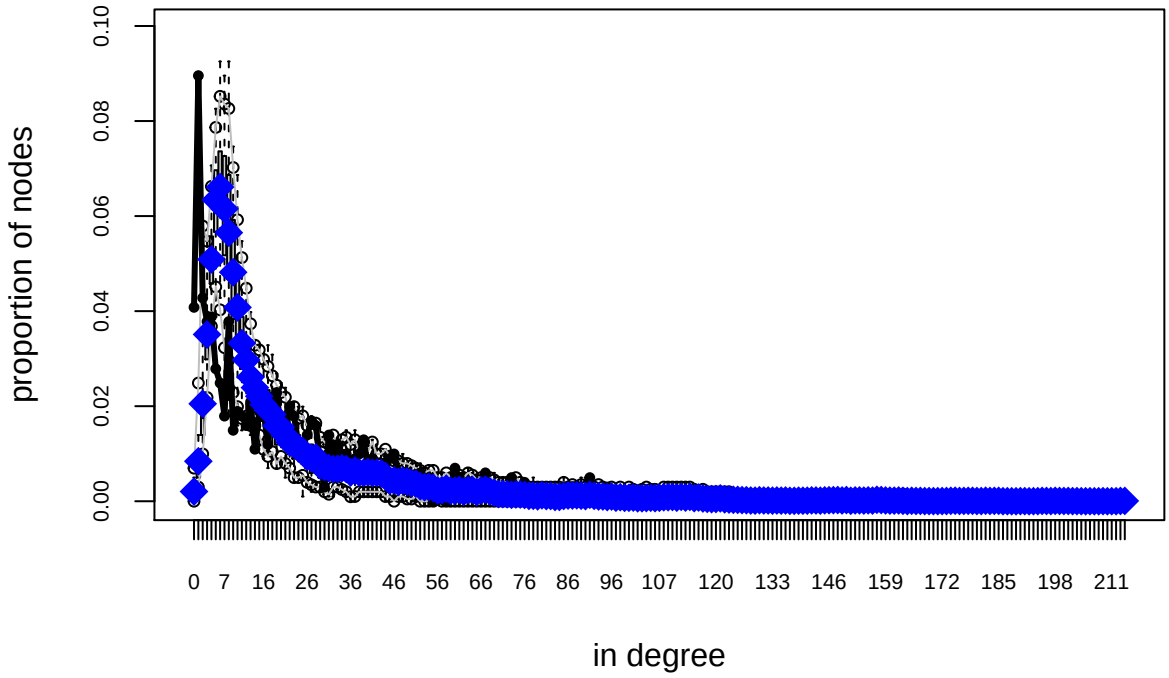
```r
summary(model5)
```

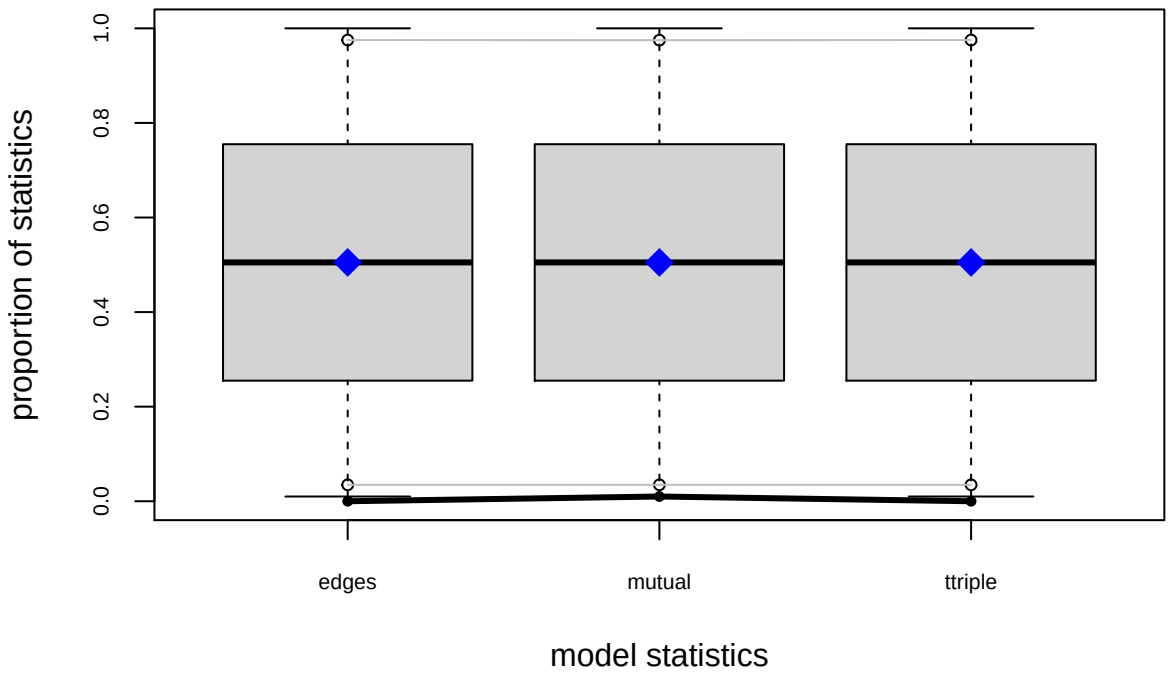
```
## Call:
## ergm(formula = formula, constraints = constraints, estimate = "MPLE",
##     control = control, verbose = verbose)
##
## Iterations:  NA
##
## Maximum Pseudolikelihood Results:
##           Estimate Std. Error MCMC % z value Pr(>|z|)
## edges   -5.3926776  0.0140853      0  -382.9   <1e-04 ***
## mutual   4.4942955  0.0206475      0   217.7   <1e-04 ***
## ttriple  0.0557305  0.0004257      0   130.9   <1e-04 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Warning:  The standard errors are based on naive pseudolikelihood and are suspect.
##
##      Null Pseudo-deviance: 1398799  on 1009020  degrees of freedom
##  Residual Pseudo-deviance:   97838  on 1009017  degrees of freedom
##
## AIC: 97844    BIC: 97880    (Smaller is better.)
```

```r
model5.gof <- gof(model5~idegree,control=control.gof.formula(nsim=100),verbose=F)
```

```
## Warning in check.control.class(c("gof.ergm", "gof.formula"), "gof.ergm"):
## Using control.gof.formula(...) as the control parameter of gof.ergm(...)
## is suboptimal and may overwrite some settings that should be preserved. Use
## control.gof.ergm(...) instead.
```

```r
plot(model5.gof)
```


proportion of nodes
0.00
0.02
0.04
0.06
0.08
0.10
0 7 16 26 36 46 56 66 76 86 96 107 120 133 146 159 172 185 198 211
in degree



I do not fully understand these ERGMs. I tried to slot the email dataset into the example code given during lectures.