Homework for Network Analysis Workshop

Min Hee Seo

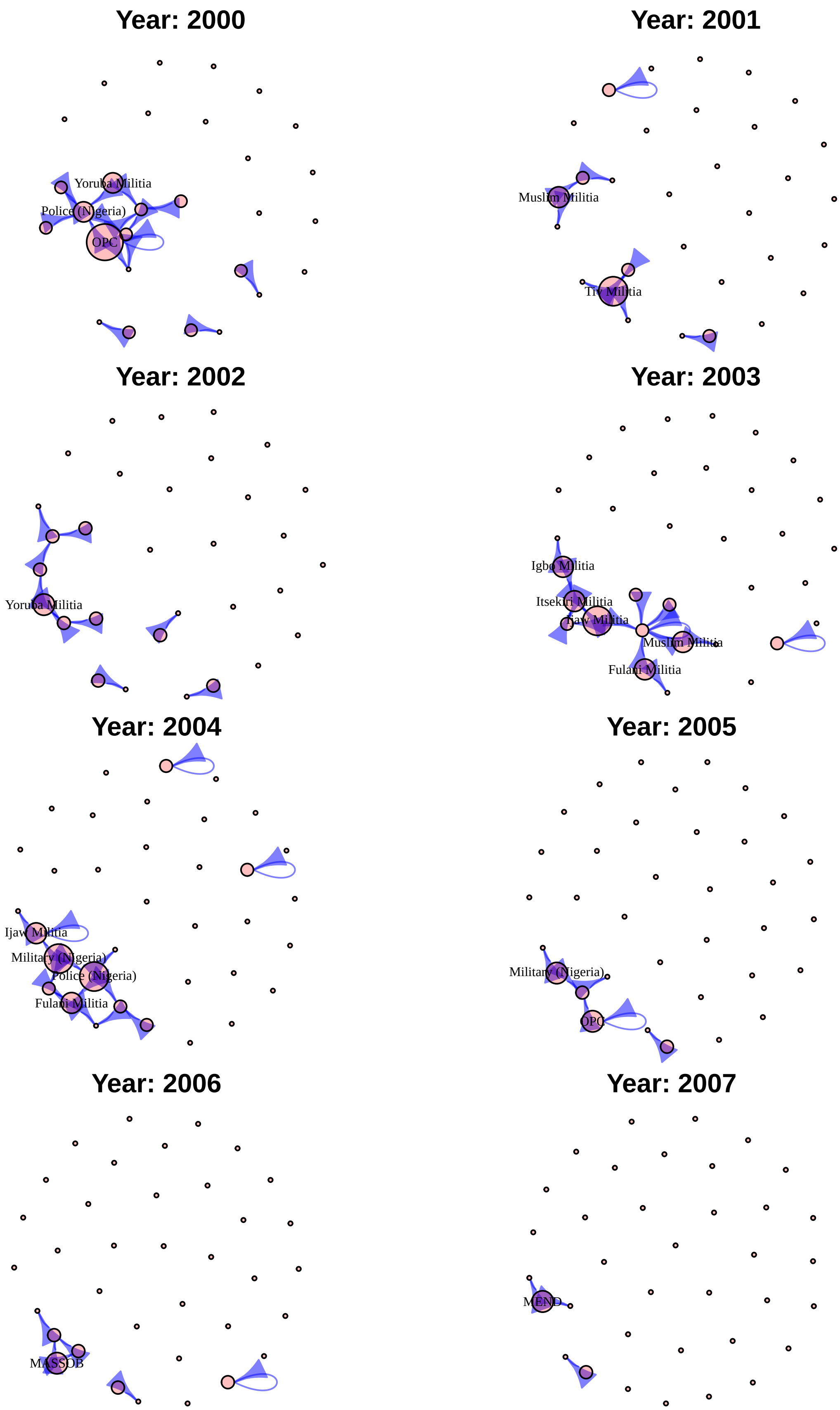
August 21st, 2018

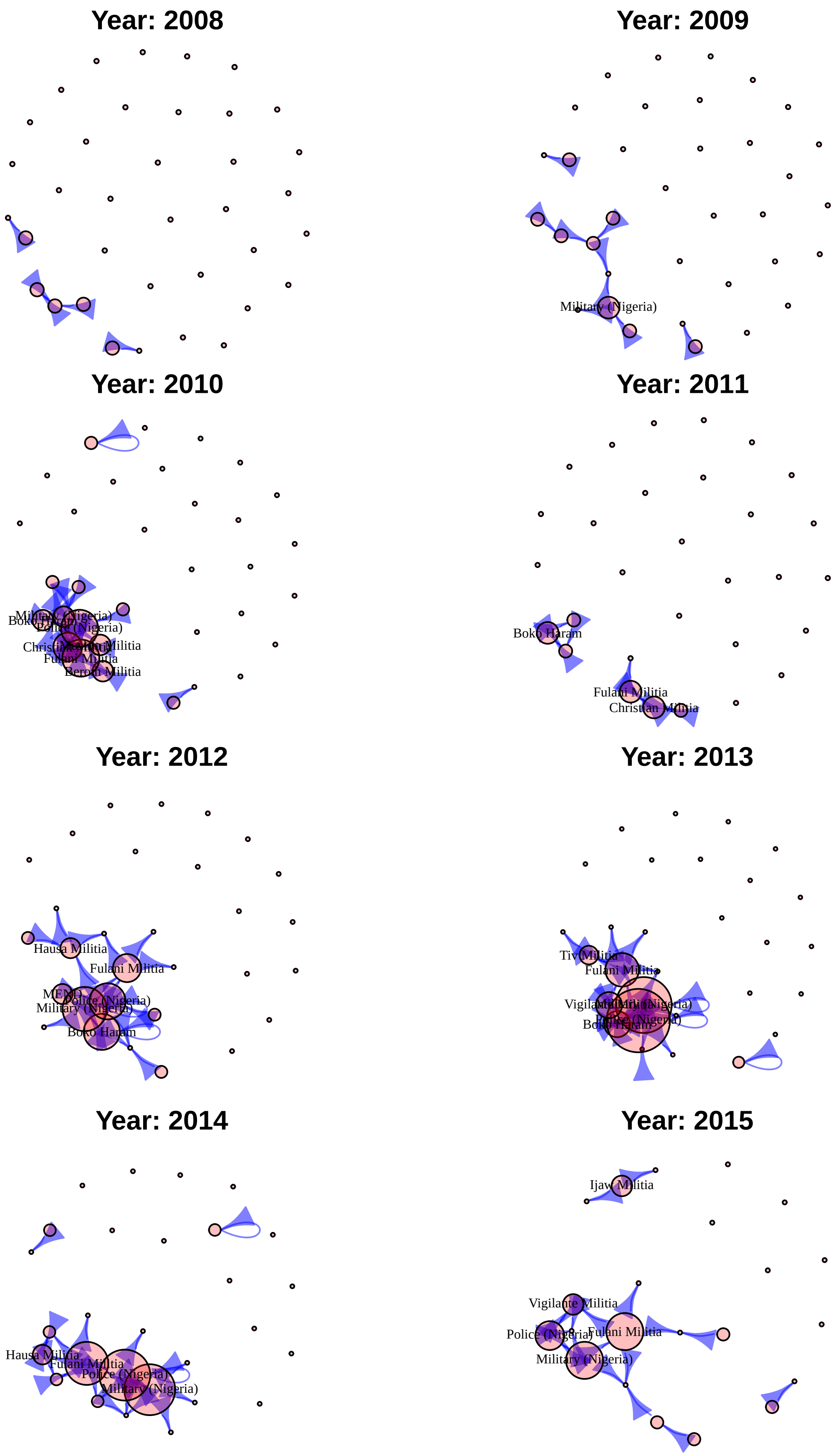
Exercise 1. Nigeria Data

```
# create network matrix by year and all years

# setup the working directory
setwd("/Users/minheeseo/Dropbox/Classes/2018_Classes/Network/network2018_hw1/")
# clean the workspace
rm(list = ls())
# loading data and R packages
library(igraph)
library(network)
load("nigeria.rda")
nigeria$sender <- gsub("\n", " ", nigeria$sender)
nigeria$receiver <- gsub("\n", " ", nigeria$receiver)

# create list where the length of list is time span
network.mat <- vector("list", length(unique(nigeria$year)) +
    1)
names(network.mat) <- unique(nigeria$year)

time <- unique(nigeria$year)
for (t in 1:length(time)) {
    slice <- NULL
    empty.mat <- NULL
    country.senter <- country.receiver <- c()
    slice <- nigeria[nigeria$year == time[t], ]
    country.sender <- unique(slice$sender)
    empty.mat <- matrix(0, length(country.sender), length(unique(slice$receiver)))
    empty.mat <- as.data.frame(empty.mat)
    rownames(empty.mat) <- country.sender
    colnames(empty.mat) <- unique(slice$receiver)
    for (i in 1:length(country.sender)) {
        country.receiver <- unique(slice$receiver[slice$sender ==
            country.sender[i]])
        for (j in 1:length(country.receiver)) {
            empty.mat[rownames(empty.mat) == country.sender[i],
                colnames(empty.mat) == country.receiver[j]] <- slice$conflict[slice$sender ==
                country.sender[i] & slice$receiver == country.receiver[j]]
        }
    }
    network.mat[[t]] <- empty.mat
}
# network.mat list contains 17 matrix each one for each year
country.sender <- unique(nigeria$sender)
empty.mat <- matrix(0, length(country.sender), length(unique(nigeria$receiver)))
empty.mat <- as.data.frame(empty.mat)
```

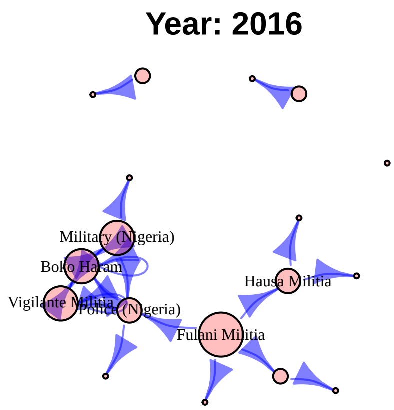
```r
rownames(empty.mat) <- country.sender
colnames(empty.mat) <- unique(nigeria$receiver)
for (i in 1:length(country.sender)) {
    country.receiver <- unique(nigeria$receiver[nigeria$sender ==
        country.sender[i]])
    for (j in 1:length(country.receiver)) {
        empty.mat[rownames(empty.mat) == country.sender[i], colnames(empty.mat) ==
            country.receiver[j]] <- sum(nigeria$conflict[nigeria$sender ==
            country.sender[i] & nigeria$receiver == country.receiver[j]])
    }
}
names(network.mat)[18] <- "All Conflict"
network.mat[[18]] <- empty.mat
```

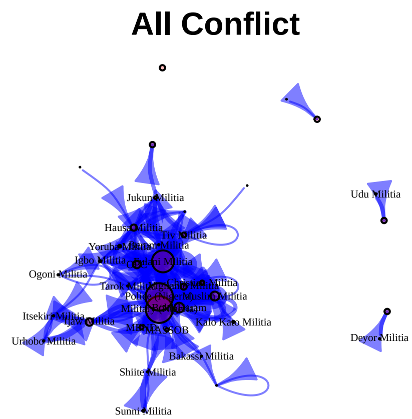
```r
# plot network by each year and all years

myblue <- rgb(red = 0, green = 0, blue = 1, alpha = .5)
mypink <- rgb(red = 1, green = 0, blue = 0, alpha = .25)

par(mfrow=c(2, 2), mar=c(0,0.2,1,0.2))
for(i in 1:4){
  g <- NULL
  g = graph_from_adjacency_matrix(as.matrix(network.mat[[i]]),
                                  mode='directed', weighted=TRUE)
  tiesSum = apply(g[], 1, sum)
# condition size based on # of ties
  V(g)$size <- (tiesSum+0.5)*6
# only label if # ties greater than 10
  V(g)$label <- ifelse( tiesSum>1, V(g)$name, NA )
  V(g)$label.cex <- 0.6
  plot(g,main=paste("Year:", names(network.mat)[i]),
     vertex.label=V(g)$label,
     vertex.size=V(g)$size,
     edge.width=E(g)$weight,
     vertex.color =mypink, # change color of nodes
     vertex.label.color = "black", # change color of labels
     edge.curved=.25, # add a 25% curve to the edges
     edge.color=myblue, # change edge color to grey
     layout=layout_with_fr)
}
```


Year: 2000
Yoruba Militia
Police (Nigeria)
OPC
Year: 2001
Muslim Militia
Tiv Militia
Year: 2002
Yoruba Militia
Year: 2003
Igbo Militia
Itsekiri Militia
Ijaw Militia
Muslim Militia
Fulani Militia
Year: 2004
Ijaw Militia
Military (Nigeria)
Police (Nigeria)
Fulani Militia
Year: 2005
Military (Nigeria)
OPC
Year: 2006
MASSOB
Year: 2007
MEND



Year: 2008
Year: 2009
Military (Nigeria)
Year: 2010
Boko Haram
Military (Nigeria)
Police (Nigeria)
Christian Militia
Fulani Militia
Berom Militia
Year: 2011
Boko Haram
Fulani Militia
Christian Militia
Year: 2012
Hausa Militia
Fulani Militia
MEND
Police (Nigeria)
Military (Nigeria)
Boko Haram
Year: 2013
Tiv Militia
Fulani Militia
Vigilante Militia
Military (Nigeria)
Police (Nigeria)
Boko Haram
Year: 2014
Hausa Militia
Fulani Militia
Police (Nigeria)
Military (Nigeria)
Year: 2015
Ijaw Militia
Vigilante Militia
Police (Nigeria)
Fulani Militia
Military (Nigeria)





Exercise 2. Measurements & Community detection

a

We can measure influence with the degree of nodes. For example, we can count total number of edges connected to a node to estimate the influence of an actor has on the network. By comparing network plot (above) and the actors who have the most degree (below), I am quite certain that I found an influential actor using degree.

```r
# to find an influential actor for each year
influence.actor <- vector("list", 18)
names(influence.actor) <- unique(nigeria$year)
for (t in 1:18) {
    temp <- graph.adjacency(as.matrix(network.mat[[t]]))
    degree <- degree(temp)
    names(degree) <- as.character(gsub("\n", " ", names(degree)))
    influence.actor[[t]] <- degree[which(degree == max(degree))]
}
# to find an influential actor overall
names(influence.actor)[18] <- "All Conflict"
temp <- graph.adjacency(as.matrix(network.mat[[18]]))
degree <- degree(temp)
names(degree) <- as.character(gsub("\n", " ", names(degree)))
influence.actor[[18]] <- degree[which(degree == max(degree))]

influence.actor  # for each year and all years
```

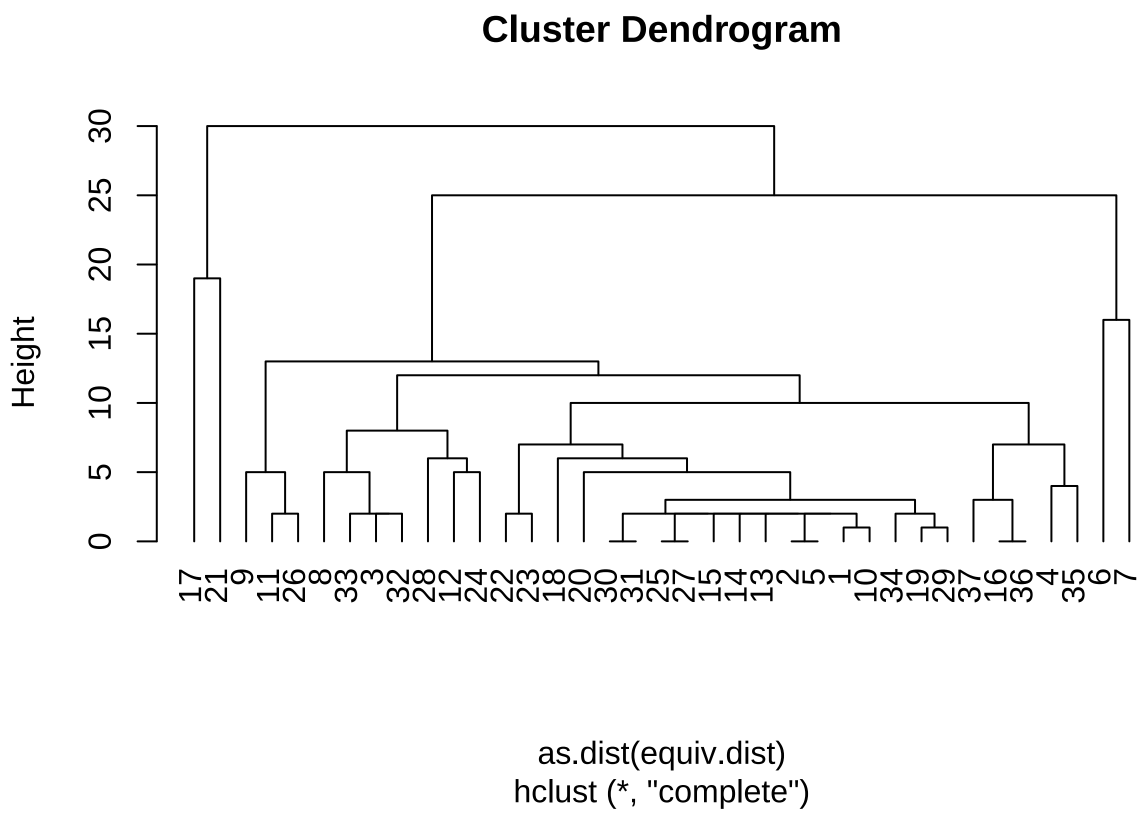
```
## $`2000`
## Police (Nigeria)
##                6
##
## $`2001`
## Tiv Militia
##           4
##
## $`2002`
##    Hausa Militia Police (Nigeria)   Yoruba Militia
##                3                3                3
##
```

```
## $`2003`
## Police (Nigeria)
##               7
##
## $`2004`
##   Fulani Militia     Ijaw Militia Police (Nigeria)
##               4                4                4
##
## $`2005`
##              OPC Police (Nigeria)
##               3                3
##
## $`2006`
## Military (Nigeria)
##                 3
##
## $`2007`
## MEND
##    2
##
## $`2008`
## Military (Nigeria)
##                 3
##
## $`2009`
## Military (Nigeria)   Police (Nigeria)
##                 3                  3
##
## $`2010`
## Police (Nigeria)
##              11
##
## $`2011`
## Christian Militia
##                4
##
## $`2012`
## Military (Nigeria)   Police (Nigeria)
##                 8                  8
##
## $`2013`
## Police (Nigeria)
##              12
##
## $`2014`
##   Fulani Militia Police (Nigeria)
##               8                8
##
## $`2015`
##     Fulani Militia Military (Nigeria)   Police (Nigeria)
##                 5                  5                  5
##
## $`2016`
## Boko Haram
```

```
##          7
##
## $`All Conflict`
## Police (Nigeria)
##               84
```

b

```r
library(sna)

# gauge data
g <- network(network.mat[[18]], directed = T)
eclusts <- equiv.clust(g)
plot(eclusts, hang = -1)  # seems like there are 8 groups
```



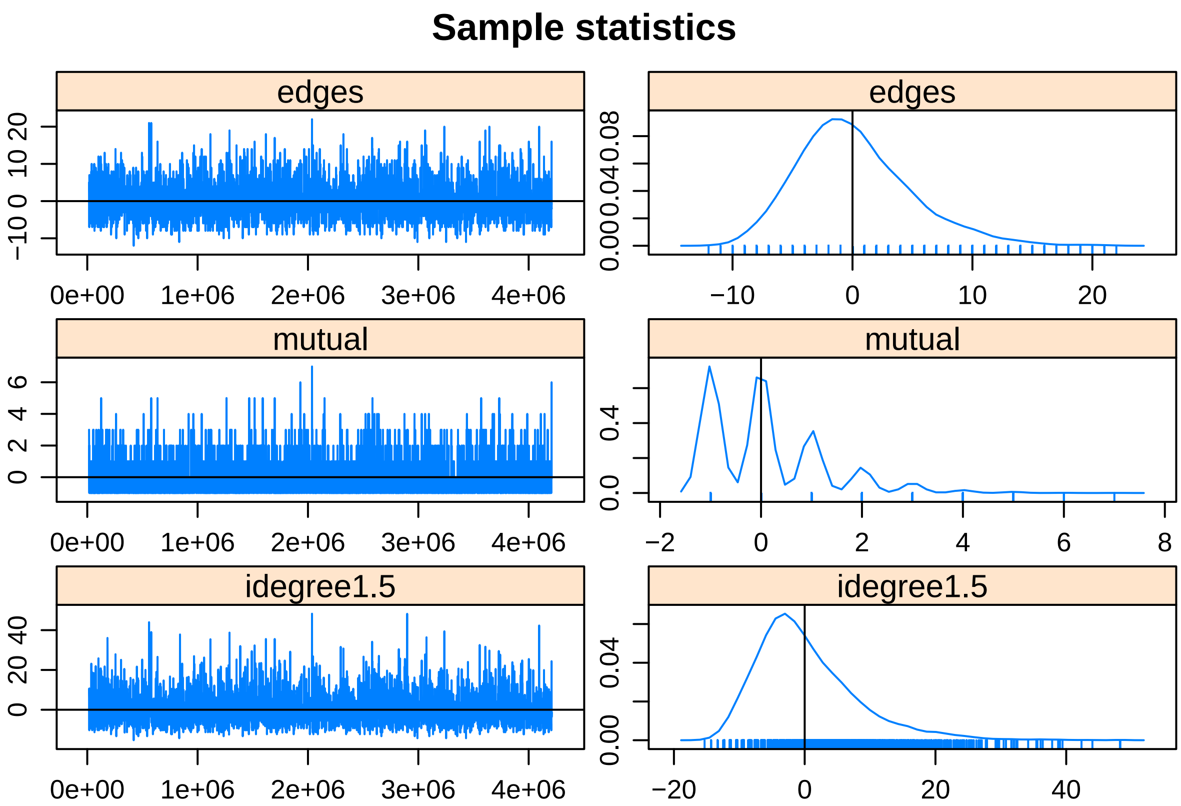

```r
# running the block model using cross-validation

# save the node classification from each run

# out of sample cross validation to validate 'k'

# report ROC statistics from each model
```

Exercise 3. ERGMs

```r
library(statnet)
# for 2004 data
set.seed(688346)
nigeria <- as.matrix(network.mat[[5]])
nigeria <- as.network.matrix(nigeria)
ergm.network <- ergm(nigeria ~ edges + mutual + idegree1.5)
mcmc.diagnostics(ergm.network)
```

```
## Sample statistics summary:
##
## Iterations = 16384:4209664
## Thinning interval = 1024
## Number of chains = 1
## Sample size per chain = 4096
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##              Mean    SD Naive SE Time-series SE
## edges      0.1370 4.753  0.07427        0.07609
## mutual     0.1008 1.140  0.01782        0.01882
## idegree1.5 0.3171 7.839  0.12249        0.12591
##
## 2. Quantiles for each variable:
##
##              2.5%    25%    50%   75% 97.5%
## edges       -8.00 -3.000  0.000 3.000  11.0
## mutual      -1.00 -1.000  0.000 1.000   3.0
## idegree1.5 -10.49 -5.118 -1.289 4.292  20.2
##
##
## Sample statistics cross-correlations:
##                edges    mutual idegree1.5
## edges      1.0000000 0.6453174  0.9630332
## mutual     0.6453174 1.0000000  0.6187392
## idegree1.5 0.9630332 0.6187392  1.0000000
##
## Sample statistics auto-correlation:
## Chain 1
##                  edges      mutual    idegree1.5
## Lag 0     1.0000000000 1.000000000  1.0000000000
## Lag 1024  0.0240893290 0.019275609  0.0274787803
## Lag 2048  0.0028486761 0.035254004  0.0005572102
## Lag 3072 -0.0007161102 0.004738489 -0.0009711013
## Lag 4096 -0.0040401289 0.025261493 -0.0002865421
## Lag 5120  0.0164390197 0.029916237  0.0100011180
##
## Sample statistics burn-in diagnostic (Geweke):
## Chain 1
##
## Fraction in 1st window = 0.1
```

```
## Fraction in 2nd window = 0.5
##
##      edges     mutual idegree1.5
##     0.8885     0.4216     1.1903
##
## Individual P-values (lower = worse):
##      edges     mutual idegree1.5
##  0.3742467  0.6733206  0.2339409
## Joint P-value (lower = worse):  0.4528554 .
```




```
##
## MCMC diagnostics shown here are from the last round of simulation, prior to computation of final para
```