Seminar 3 and HW 2

Lesnykh Kirill

February 1, 2019

```r
library(network)
```

```
## network: Classes for Relational Data
## Version 1.13.0.1 created on 2015-08-31.
## 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.
```

```r
library(sna)
```

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

```
## Warning: package 'statnet.common' was built under R version 3.5.2
```

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

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

```
## sna: Tools for Social Network Analysis
## Version 2.4 created on 2016-07-23.
## copyright (c) 2005, Carter T. Butts, University of California-Irvine
##  For citation information, type citation("sna").
##  Type help(package="sna") to get started.
```

```r
load('madmen.Rdata')

dim(mad.att)
```

```
## [1] 45  2
```

```r
head(mad.att)
```

```
##                Names Female
## 32       Abe Drexler      0
## 12           Allison      1
```

```
## 44 Bellhop in Baltimore      0
## 13      Bethany Van Nuys      1
## 1           Betty Draper      1
## 14        Bobbie Barrett      1
```

```
mad.matrix[1:6,1:2]
```

```
##                      Abe Drexler Allison
## Abe Drexler                    0       0
## Allison                        0       0
## Bellhop in Baltimore           0       0
## Bethany Van Nuys               0       0
## Betty Draper                   0       0
## Bobbie Barrett                 0       0
```

```
sum(as.character(mad.att[,1]) == colnames(mad.matrix))
```

```
## [1] 45
```

```
mad.net <- as.network(mad.matrix, directed=FALSE)

mad.att
```

```
##                       Names Female
## 32              Abe Drexler      0
## 12                  Allison      1
## 44     Bellhop in Baltimore      0
## 13         Bethany Van Nuys      1
## 1              Betty Draper      1
## 14           Bobbie Barrett      1
## 33 Brooklyn College Student      0
## 15                  Candace      1
## 2                Don Draper      0
## 16                    Doris      1
## 34            Duck Phillips      0
## 17              Faye Miller      1
## 27                 Franklin      0
## 28              Greg Harris      0
## 36                  Gudrun      1
## 3               Harry Crane      0
## 10           Henry Francis      0
## 25                    Hildy      1
## 39         Ida Blankenship      1
## 40              Jane Siegel      1
## 29                   Janine      1
## 26          Jennifer Crane      1
## 4             Joan Holloway      1
```

```
## 18                        Joy      1
## 45               Kitty Romano      1
## 5                  Lane Pryce      0
## 35                       Mark      0
## 19               Megan Calvet      1
## 20              Midge Daniels      1
## 41             Mirabelle Ames      1
## 42              Mona Sterling      1
## 6                 Peggy Olson      1
## 7               Pete Campbell      0
## 37          Playtex bra model      1
## 21              Rachel Menken      1
## 11                 Random guy      0
## 30              Rebecca Pryce      1
## 8              Roger Sterling      0
## 9                  Sal Romano      0
## 22                     Shelly      1
## 23            Suzanne Farrell      1
## 31                       Toni      1
## 38             Trudy Campbell      1
## 43                      Vicky      1
## 24 Woman at the Clios party      1
```

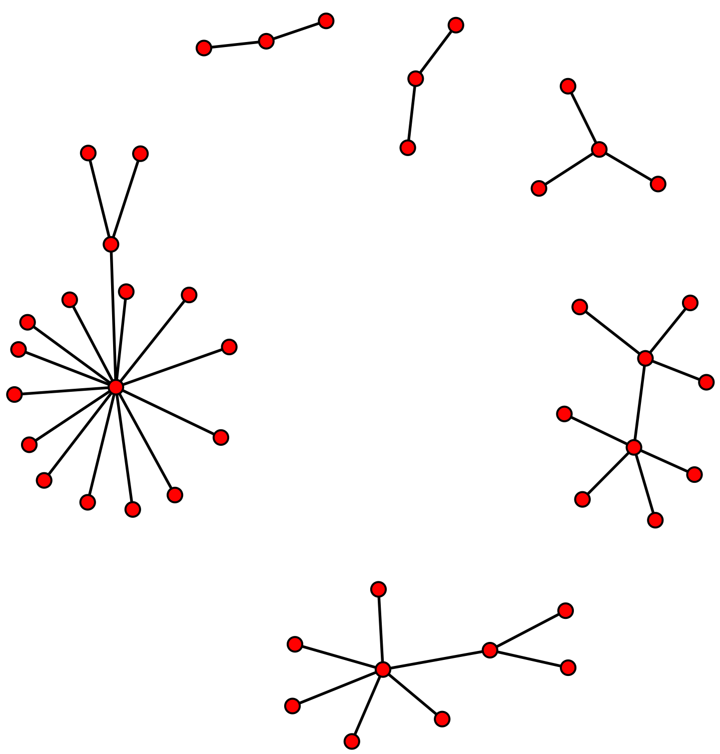
```r
set.vertex.attribute(mad.net, attrname='female', value=mad.att[,2])

mad.net
```

```
##  Network attributes:
##   vertices = 45
##   directed = FALSE
##   hyper = FALSE
##   loops = FALSE
##   multiple = FALSE
##   bipartite = FALSE
##   total edges= 39
##     missing edges= 0
##     non-missing edges= 39
##
##  Vertex attribute names:
##     female vertex.names
##
## No edge attributes
```

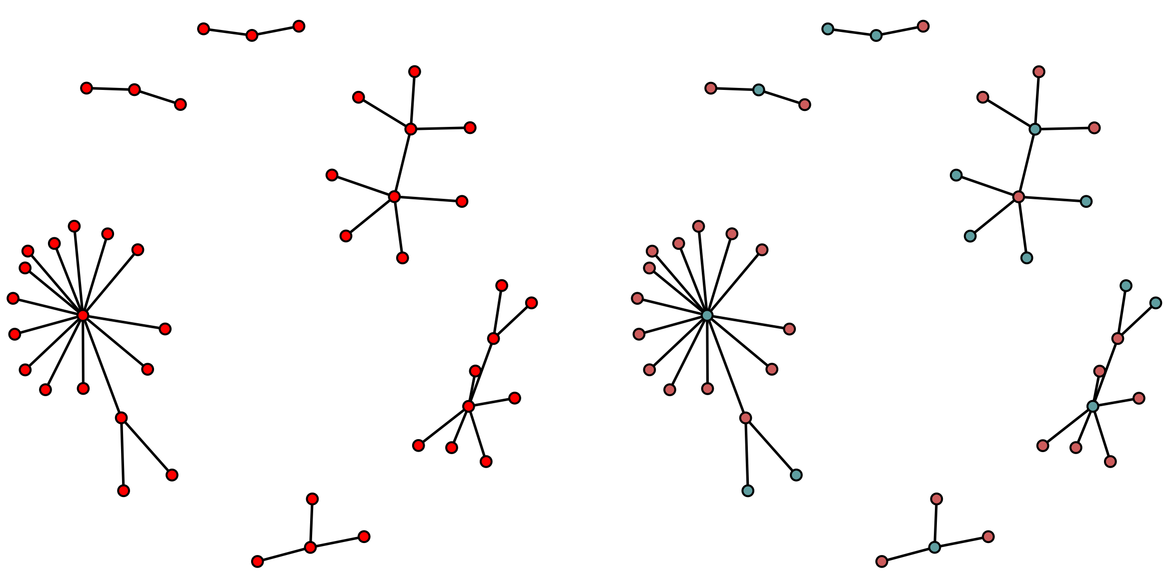
Assignment question: Why did we use option FALSE for command ???directed??? above, when creating a network?

We used option FALSE because the matrix don't have the directed ties, it contains undirected ones - the edges. Sexual contact is the type of the connection that is not directed.

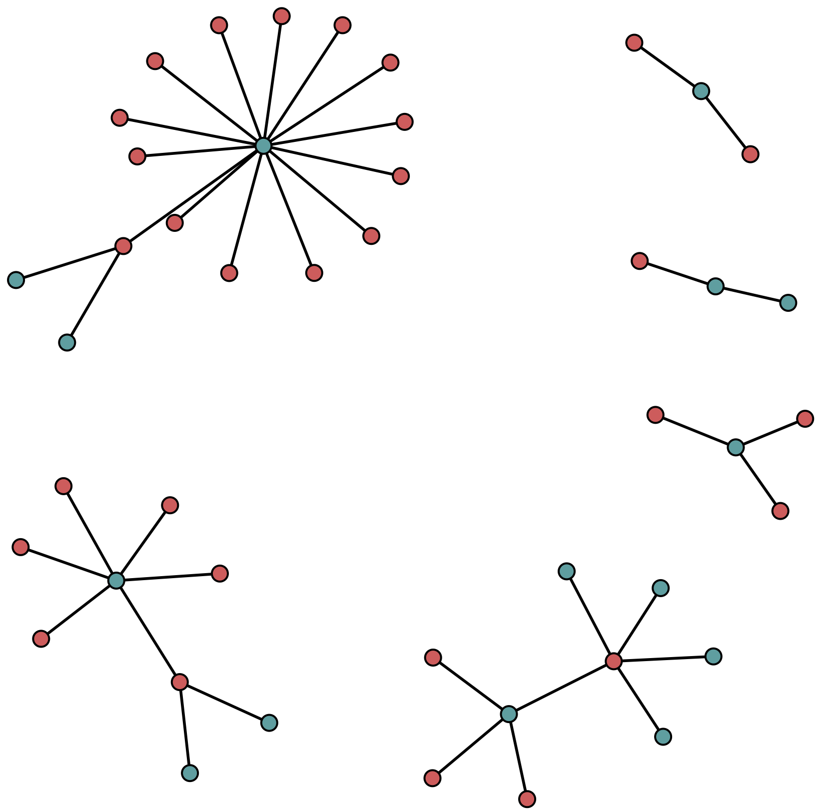
```
par(mar=c(1,1,1,1))
plot(mad.net)
```



```
colors <- ifelse(mad.att$Female == 1, 'indianred', 'cadetblue')
par(mar=c(0,0.5,0,0.5),mfrow=c(1,2))
plot(mad.net, vertex.col = colors,coord=plot(mad.net))
```



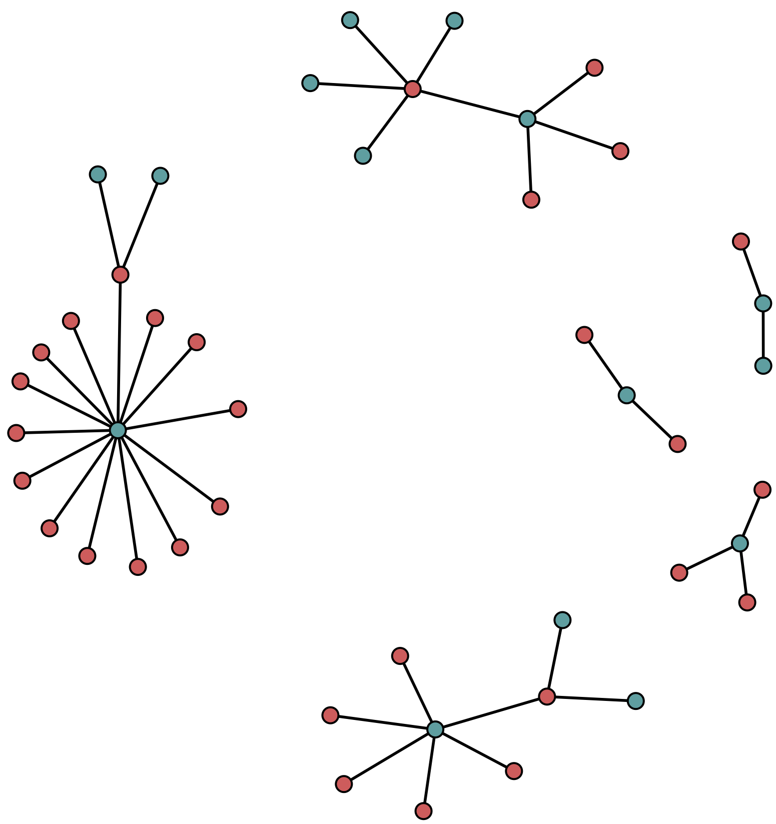
```
#pdf( 'myplot.pdf', width=4, height=4) # width and height are in inches
par(mar=c(0,0,0,0))
plot(mad.net, vertex.col = colors, mode = 'fruchtermanreingold' )
```



```
dev.off()
```

```
## null device
##           1
```

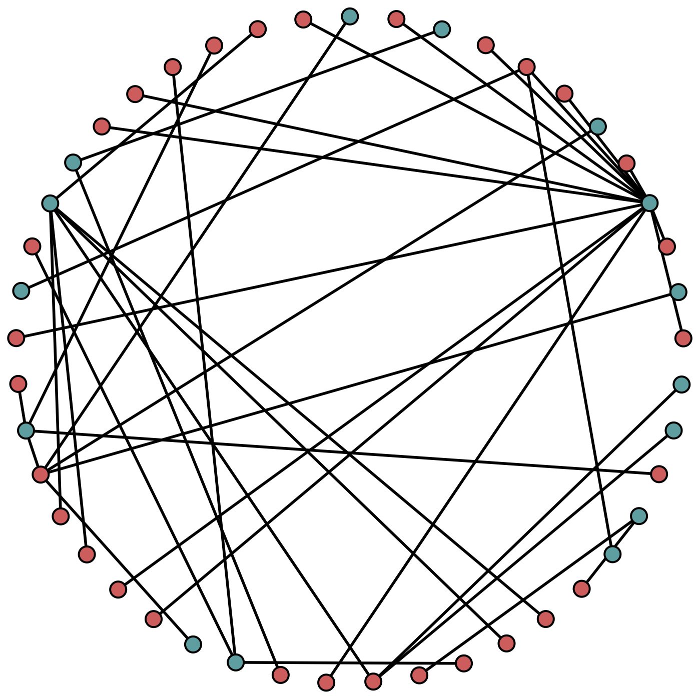
```
#png( 'myplot.png')
par(mar=c(0,0,0,0))
plot(mad.net, vertex.col = colors)
```



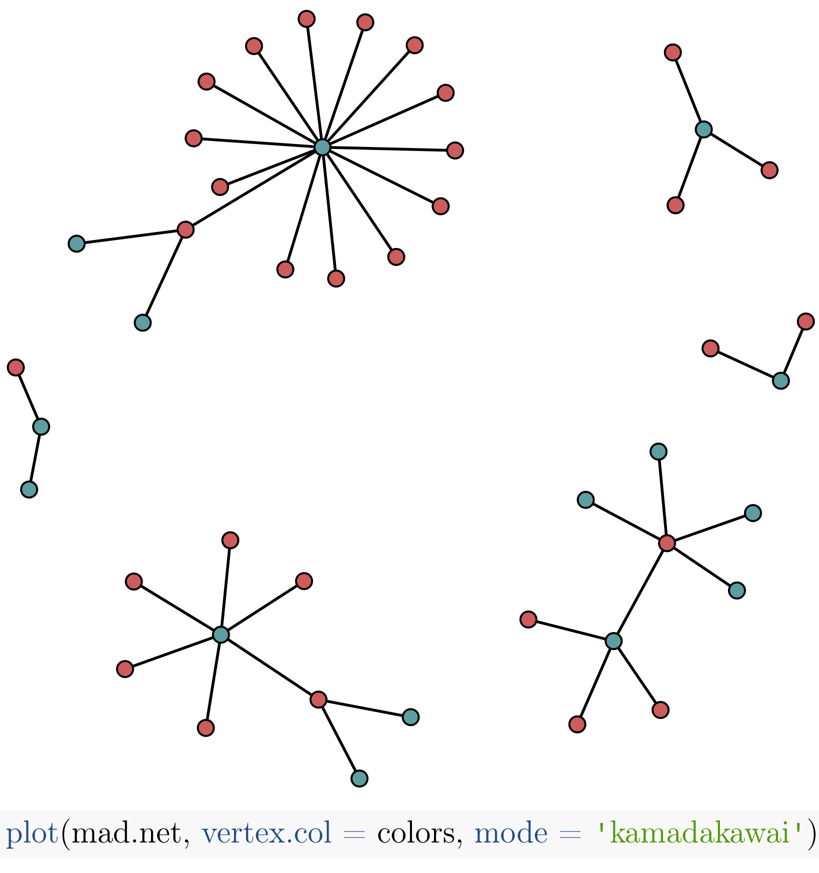
```
dev.off()
```

```
## null device
##           1
```

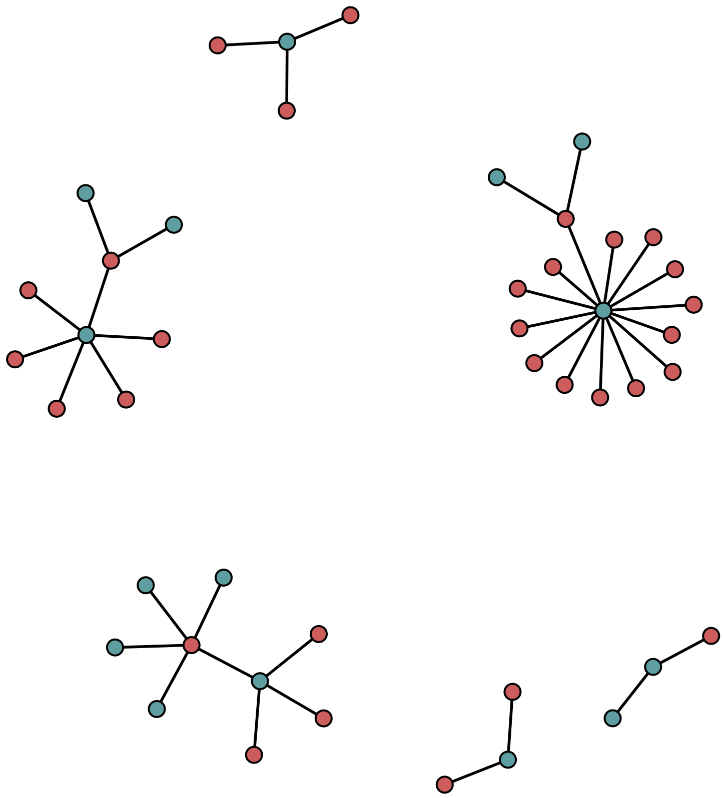
```
par(mar=c(0,0,0,0))
plot(mad.net, vertex.col = colors, mode = 'circle')
```



```
plot(mad.net, vertex.col = colors, mode = 'fruchtermanreingold')
```



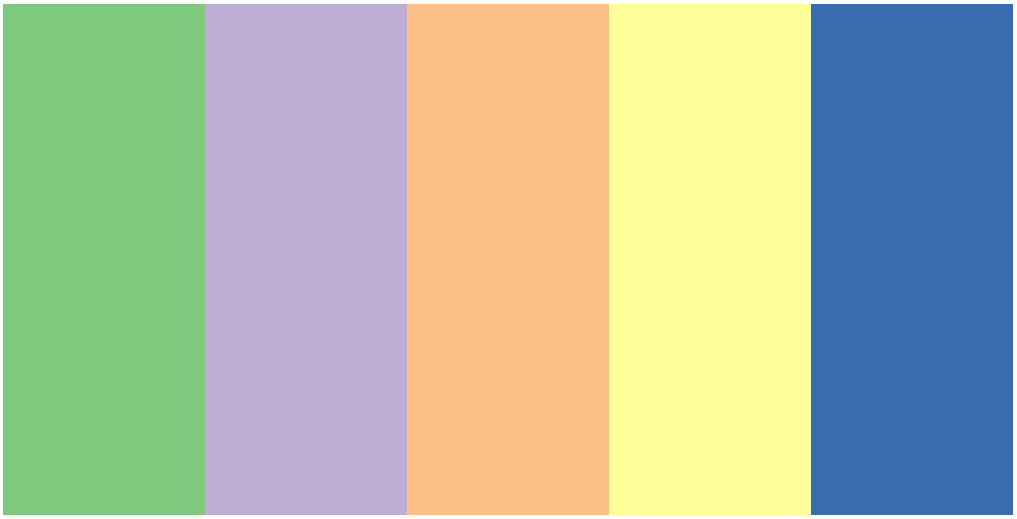
```
plot(mad.net, vertex.col = colors, mode = 'kamadakawai')
```



```r
library(RColorBrewer)
par(mar=c(2,2,2,2))
display.brewer.pal(12, 'Paired')
```

```
par(mar=c(1,1,1,1),mfrow=c(2,3))
```

```
display.brewer.pal(5, 'Accent')
```



Accent (qualitative)

```
display.brewer.pal(5, 'Dark2')
```

Dark2 (qualitative)

```
display.brewer.pal(5, 'Pastel1')
```

Pastel1 (qualitative)

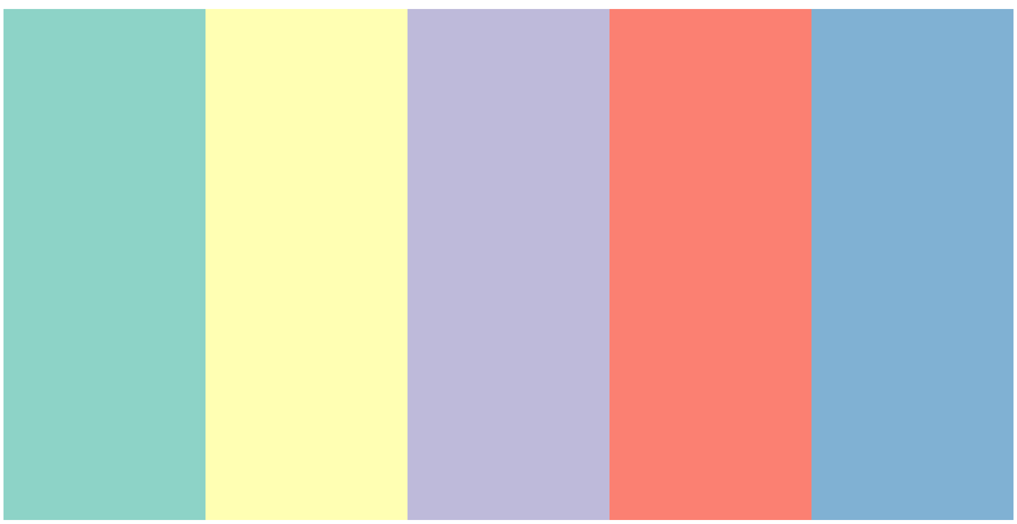
```
display.brewer.pal(5, 'Set1')
```

Set1 (qualitative)

```
display.brewer.pal(5, 'Set2')
```

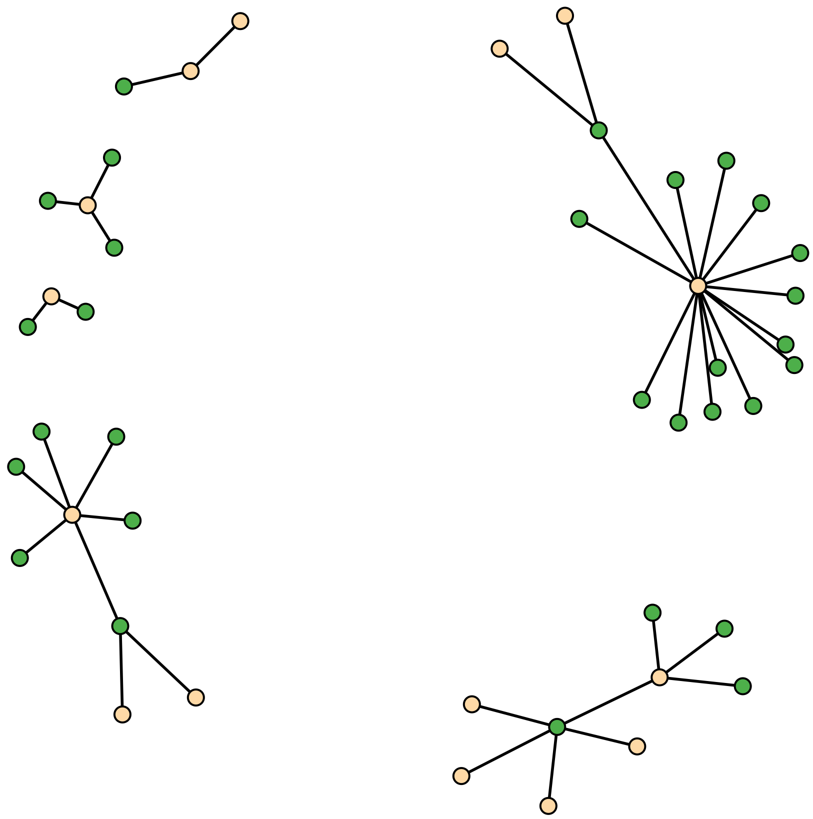
Set2 (qualitative)

```
display.brewer.pal(5, 'Set3')
```



Set3 (qualitative)

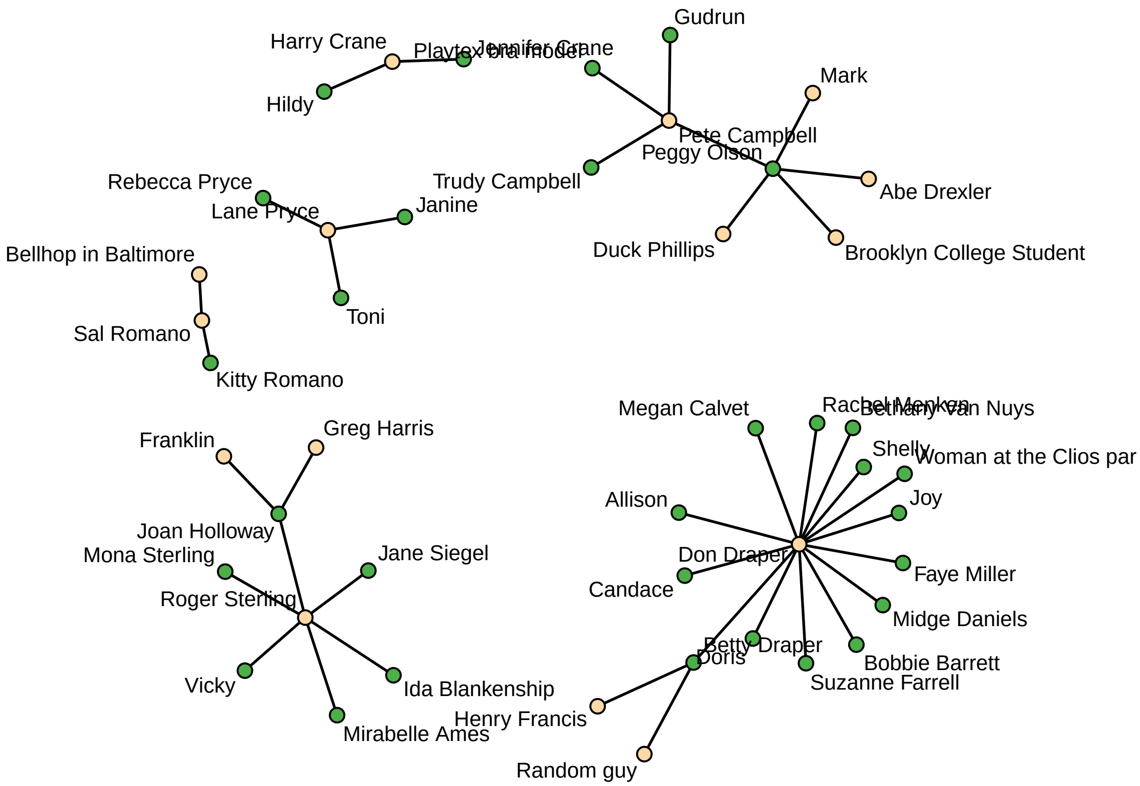
```
col1 <- brewer.pal(5, 'Set1')
colPastel<-brewer.pal(5, 'Pastel1')

colors <- ifelse(mad.att$Female == 1, col1[3], colPastel[5])
par(mar=c(0,0,0,0))
plot(mad.net, vertex.col = colors )
```



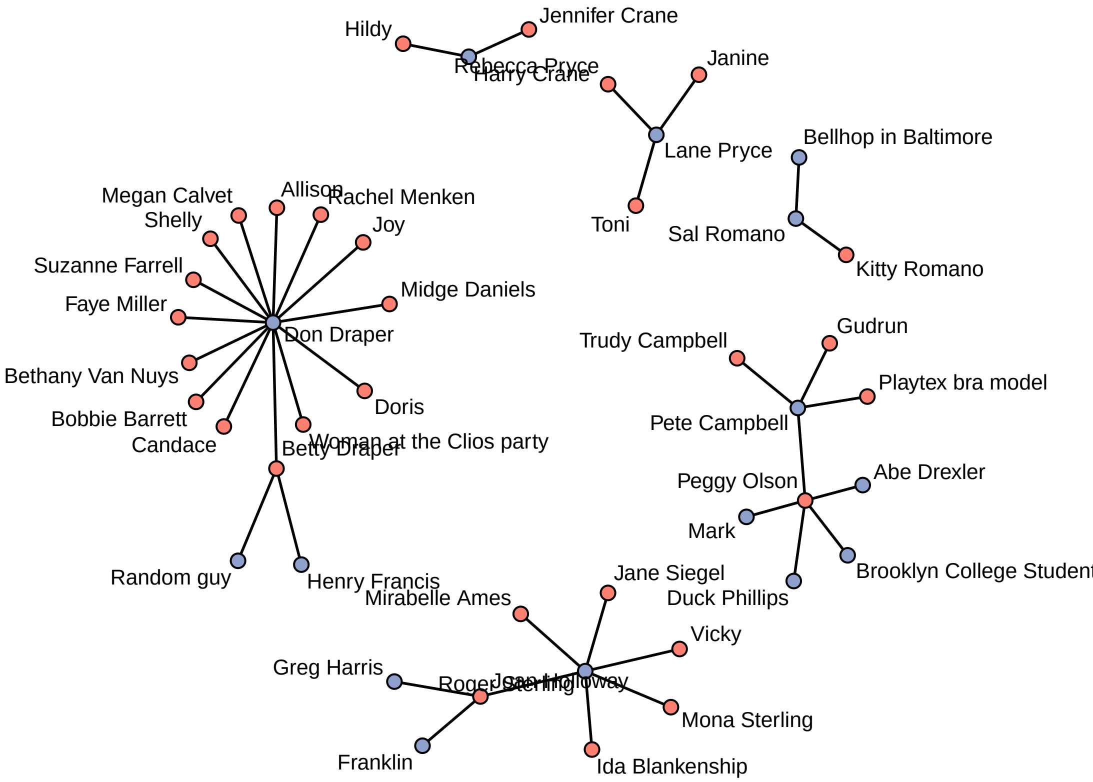
Assignment task: Please examine the options in the ???network.layout???"command and perform the following:

1. Create the madmen.net with labels.
2. Experiment with options by adding attributes, changing vertex or edge colors, finding the best position for labels. While this task may take a while, it will count as complete if you generate at least one graph that is different from the graphs I???ve shown you in this assignment. The more different graphs with options you generate, the better - extra-credit never hurts anyone.

```
par(mar=c(1,1,1,1))
plot(mad.net,
     vertex.col = colors,
     displaylabels=TRUE,
     label.cex=.7,
     label.pos=0,
     mode ='fruchtermanreingold')
```




```r
par(mar=c(1,1,1,1))
plot(mad.net,
     vertex.col = (ifelse(mad.att$Female == 1, (brewer.pal(5, 'Set3'))[4], (brewer.pal(5, 'Set2')[3]))),
     displaylabels=TRUE,
     label.cex=.7,
     label.pos=0,
     mode ='fruchtermanreingold')
```



I have generated two different graphs of mad.net with labels. There is not a lot variations to build a new graphs, because there is only one possible attribute to add to the network.

```
load('flo.Rdata')
flo.marriage<-as.network(flo.marriage)
library('sna')
network.dyadcount(flo.marriage)
```

```
## [1] 240
```

```
dyad.census(flo.marriage)
```

```
##      Mut Asym Null
## [1,]  20    0  100
```

```
network.edgecount(flo.marriage)
```

```
## [1] 40
```

```
network.density(flo.marriage)
```

```
## [1] 0.1666667
```

```
triad.census(flo.marriage)
```

```
##      003 012 102 021D 021U 021C 111D 111U 030T 030C 201 120D 120U 120C 210
## [1,] 324   0 195    0    0    0    0    0    0    0  38    0    0    0    0
```

```
##       300
## [1,]   3
```

```r
gtrans(flo.marriage, measure='weak')
```

```
## [1] 0.1914894
```

```r
kpath.census(flo.marriage, mode = "digraph",
  tabulate.by.vertex = FALSE, path.comembership = "none",
  dyadic.tabulation = "none")
```

```
## $path.count
##   1   2   3
##  40  94 174
```

```r
kcycle.census(flo.marriage, maxlen = 4, mode = "digraph",
  tabulate.by.vertex = FALSE, cycle.comembership = "none")
```

```
## $cycle.count
##  2  3  4
## 20  6  4
```

```r
geo.dist<-geodist(flo.marriage)
class(geo.dist)
```

```
## [1] "list"
```

```r
summary(geo.dist)
```

```
##        Length Class  Mode
## counts 256    -none- numeric
## gdist  256    -none- numeric
```

```r
summary(geo.dist$counts)
```

```
##        V1              V2               V3               V4
##  Min.   :0.00   Min.   :0.000   Min.   :0.000   Min.   :0.000
##  1st Qu.:1.00   1st Qu.:1.000   1st Qu.:1.000   1st Qu.:1.000
##  Median :1.00   Median :1.000   Median :1.000   Median :1.000
##  Mean   :1.25   Mean   :1.062   Mean   :1.125   Mean   :1.562
##  3rd Qu.:1.25   3rd Qu.:1.000   3rd Qu.:1.000   3rd Qu.:2.250
##  Max.   :3.00   Max.   :2.000   Max.   :2.000   Max.   :3.000
##        V5               V6               V7               V8
##  Min.   :0.000   Min.   :0.000   Min.   :0.000   Min.   :0.000
##  1st Qu.:1.000   1st Qu.:1.000   1st Qu.:1.000   1st Qu.:1.000
##  Median :1.000   Median :1.000   Median :1.000   Median :1.000
##  Mean   :1.188   Mean   :1.062   Mean   :1.312   Mean   :1.312
##  3rd Qu.:1.250   3rd Qu.:1.000   3rd Qu.:2.000   3rd Qu.:2.000
##  Max.   :2.000   Max.   :2.000   Max.   :2.000   Max.   :2.000
```

```
##        V9              V10             V11             V12        
##  Min.   :0.00   Min.   :0.00   Min.   :0.00   Min.   :0.0000  
##  1st Qu.:1.00   1st Qu.:1.00   1st Qu.:1.00   1st Qu.:0.0000  
##  Median :1.00   Median :1.00   Median :1.00   Median :0.0000  
##  Mean   :1.25   Mean   :1.25   Mean   :1.25   Mean   :0.0625  
##  3rd Qu.:1.25   3rd Qu.:1.25   3rd Qu.:2.00   3rd Qu.:0.0000  
##  Max.   :3.00   Max.   :3.00   Max.   :2.00   Max.   :1.0000  
##       V13              V14             V15              V16       
##  Min.   :0.0000   Min.   :0.00   Min.   :0.000   Min.   :0.000  
##  1st Qu.:1.0000   1st Qu.:1.00   1st Qu.:1.000   1st Qu.:1.000  
##  Median :1.0000   Median :1.00   Median :1.000   Median :1.000  
##  Mean   :0.9375   Mean   :1.25   Mean   :1.062   Mean   :1.188  
##  3rd Qu.:1.0000   3rd Qu.:1.25   3rd Qu.:1.000   3rd Qu.:1.250  
##  Max.   :1.0000   Max.   :3.00   Max.   :2.000   Max.   :2.000
```

```
summary(geo.dist$gdist)
```

```
##        V1              V2              V3             V4              V5       
##  Min.   :0.00   Min.   :0.00   Min.   : 0   Min.   :0.00   Min.   :0.00  
##  1st Qu.:2.00   1st Qu.:1.75   1st Qu.: 2   1st Qu.:1.75   1st Qu.:1.75  
##  Median :3.00   Median :2.00   Median : 2   Median :2.00   Median :3.00  
##  Mean   : Inf   Mean   : Inf   Mean   :Inf   Mean   : Inf   Mean   : Inf  
##  3rd Qu.:3.25   3rd Qu.:3.00   3rd Qu.: 3   3rd Qu.:3.25   3rd Qu.:3.25  
##  Max.   : Inf   Max.   : Inf   Max.   :Inf   Max.   : Inf   Max.   : Inf  
##        V6              V7            V8            V9              V10       
##  Min.   :0.00   Min.   : 0   Min.   : 0   Min.   :0.00   Min.   :0.00  
##  1st Qu.:2.75   1st Qu.: 1   1st Qu.: 2   1st Qu.:1.00   1st Qu.:3.00  
##  Median :3.00   Median : 2   Median : 3   Median :2.00   Median :3.50  
##  Mean   : Inf   Mean   :Inf   Mean   :Inf   Mean   : Inf   Mean   : Inf  
##  3rd Qu.:4.00   3rd Qu.: 3   3rd Qu.: 4   3rd Qu.:2.25   3rd Qu.:4.25  
##  Max.   : Inf   Max.   :Inf   Max.   :Inf   Max.   : Inf   Max.   : Inf  
##       V11              V12            V13             V14             V15    
##  Min.   :0.00   Min.   : 0   Min.   :0.00   Min.   :0.00   Min.   : 0  
##  1st Qu.:1.75   1st Qu.:Inf   1st Qu.:1.75   1st Qu.:2.00   1st Qu.: 1  
##  Median :3.00   Median :Inf   Median :2.00   Median :2.50   Median : 2  
##  Mean   : Inf   Mean   :Inf   Mean   : Inf   Mean   : Inf   Mean   :Inf  
##  3rd Qu.:4.00   3rd Qu.:Inf   3rd Qu.:2.25   3rd Qu.:3.25   3rd Qu.: 3  
##  Max.   : Inf   Max.   :Inf   Max.   : Inf   Max.   : Inf   Max.   :Inf  
##       V16       
##  Min.   :0.00  
##  1st Qu.:1.75  
##  Median :2.00  
##  Mean   : Inf  
##  3rd Qu.:3.00  
##  Max.   : Inf
```

```r
summary(geo.dist$counts[,3])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   1.000   1.000   1.125   1.000   2.000
```

```r
summary(geo.dist$counts[3,])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   1.000   1.000   1.125   1.000   2.000
```

Assignment question: What can you say about the last two commands? Why is the result what it is?

The result of those two lines contains the information about the Vertex 3 of the network Flo.marriage.. The information contains some indicators about the amount of the ties that concreate vertice has. So, that means, that some vertex 3 has minimum 0 ties with any other vertex, has maximum 3 ties with any other vertex, the median amount of the amount of ties equals 1, and the mean amount of all the ties equals 1.125. I suppose, that those indicators might contain a more usefeul and interesting information if the network model would have much more nodes and ties and in general would be more complex.

Howework 2

```r
load('trade.Rdata')
dim(crude)
```

```
## [1] 24 24
```

```r
dim(diplomacy)
```

```
## [1] 24 24
```

```r
dim(food)
```

```
## [1] 24 24
```

```r
dim(manufacture)
```

```
## [1] 24 24
```

```r
dim(minerals)
```

```
## [1] 24 24
```

```r
dim(trade.all)
```

```
## [1] 24 24
```

```r
dim(trade.att)
```

```
## [1] 24  4
```

```r
sum(colnames(trade.all) == rownames(trade.att))
```

[1] 24

1. Are the matrices symmetric?
2. What does that mean for resulting networks? Would they be directed or undirected?

The matrices have been checked with the commands "dim" and summing the amont of the rows and columns. There is an equal amount of rows and columns in all the matrices, so we can be sure that they are symmetric

We have to be sure about it, because otherwisw we won't be able to build the network models properly. Also, that means that the ties between the nodes might be undirected as well as undirected. For some models there might be built only undirected ties. Except for the "trade.all". If we are going to use atributes, it is possible that the ties would be durected.

```r
class(trade.all)
```

[1] "data.frame"

```r
trade.all<-as.matrix(trade.all)

trade.any <- ifelse(trade.all > 0, 1, 0)

trade.2 <- ifelse(trade.all > 1, 1, 0)

trade.max <- ifelse(trade.all == 5, 1, 0)
```

1. How would you justify any of these choices? Please refer to specific social theories to make your answer more legitimate.

2. What are the impirical implication of these choices?

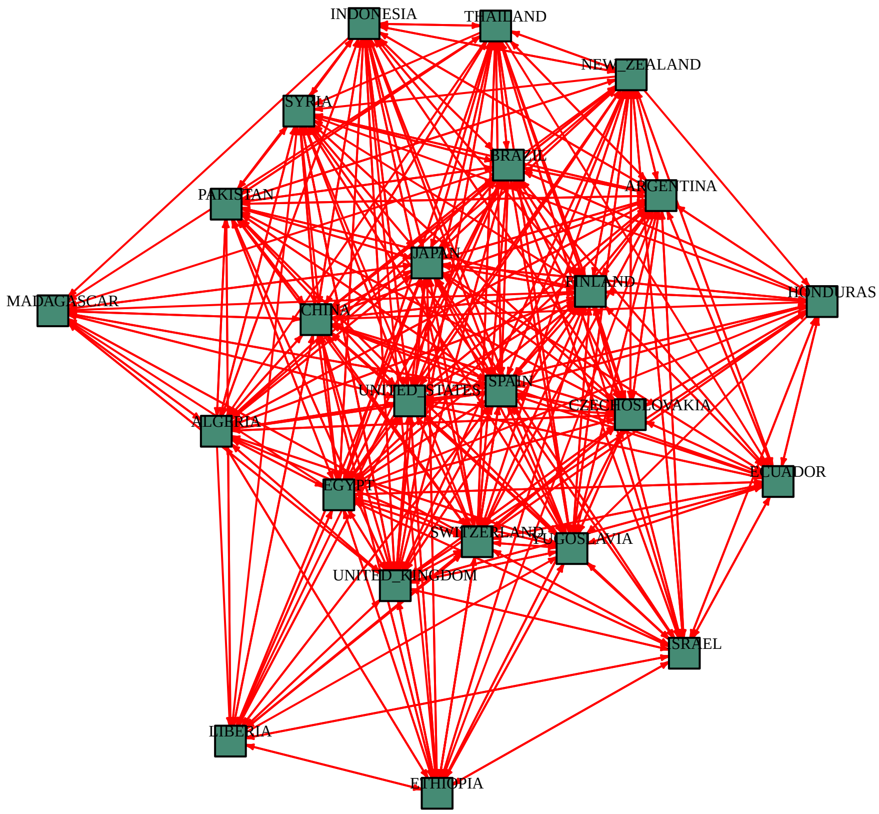
3. In the code above we recoded the matrix, to the 3 other ones. There are 3 matrices now: containing the any relations between countries as ties, containing more than 1 types of relations as ties, and containing all 5 types of relations as ties. That means each other network contains the ties of a rising power. I mean, the more types of political relations countries have, the stronger theier political connection is. In general it is possible to characterise the given relations between countries with the classic theory of realism in international relations. Some of the following characteristcis are followed: countries are independent players, there is no any type of rule above them, so they co-exist in the terms of anarchy. Also, all countries tend to collect resources and exchange them in the way they see as more profitable. I suppose, relating to the theory of realsim of international relations is appropriate in that case.

4. The impirical implication of choices to sum different types of economic and polytical relations is the way to measure the power of different ties between countries. In that way, it is possible to see, which countries collaborate with each other in a more closer way.

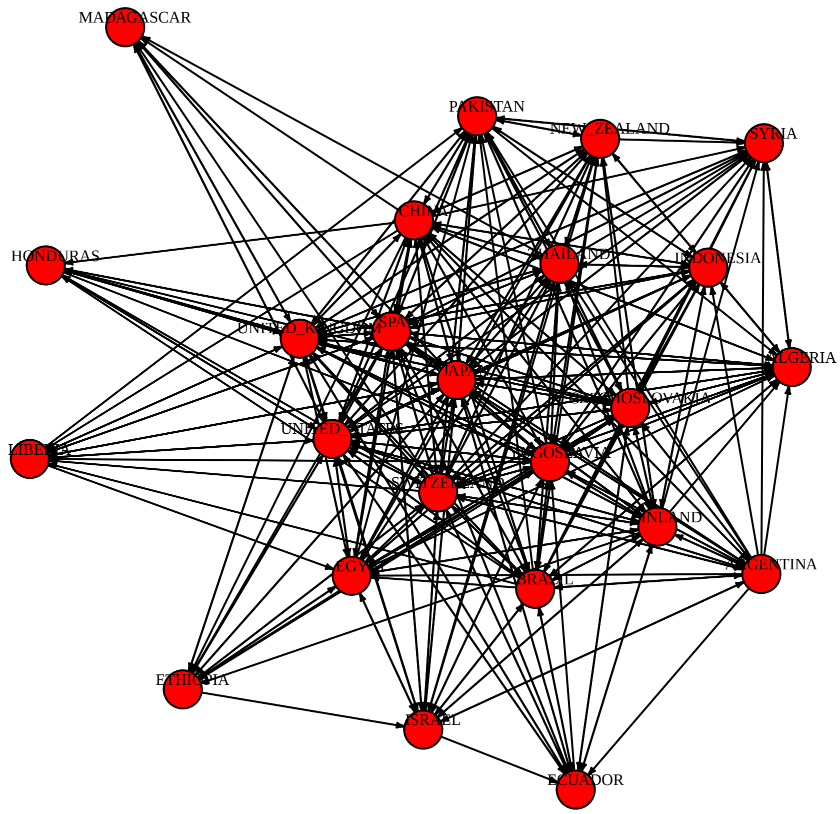
```r
detach(package:sna)
detach(package:network)
library(igraph)
```

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

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

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

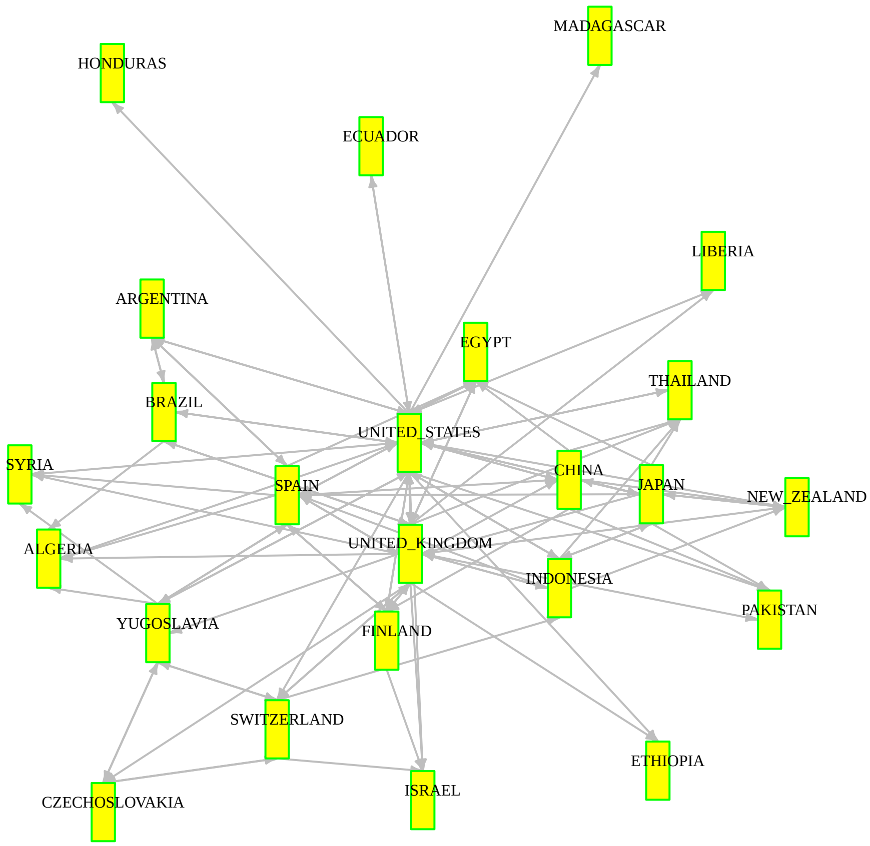
```r
tradegraph.any <-graph.adjacency(trade.any,
mode=c("directed"),
weighted=NULL,
diag=FALSE)

tradegraph.2 <-graph.adjacency(trade.2,
mode=c("directed"),
weighted=NULL,
diag=FALSE)

tradegraph.5 <-graph.adjacency(trade.max,
mode=c("directed"),
weighted=NULL,
diag=FALSE)

par(mar=c(0,0,0,0))

plot(tradegraph.any,
  vertex.size = 8,
  edge.arrow.size = .2,
  vertex.label.cex = .5,
  vertex.color = 'aquamarine4',
  edge.color='red',
  vertex.shape = 'square',
  vertex.label.dist = .5,
  vertex.label.color = 'black')
```




```
par(mar=c(0,0,0,0))
plot(tradegraph.2,
  vertex.size = 10,
  edge.arrow.size = .2,
  vertex.label.cex = .5,
  vertex.color = 'red',
  edge.color='black',
  vertex.shape = 'circle',
  vertex.label.dist = .5,
  vertex.label.color = 'black')
```




```
par(mar=c(0,0,0,0))
plot(tradegraph.5,
  vertex.size = 6,
  edge.arrow.size = .3,
  edge.color='gray',
  vertex.label.cex = .5,
  vertex.color = 'yellow',
  vertex.shape = 'crectangle',
  vertex.frame.color = 'green',
  vertex.label.dist = .5,
  vertex.label.color = 'black')
```



1. What differences do you observe between the graphs where the cutpoint is any tie, at least two ties, and all ties present?

2. What information can you gather from these observed differences to help you expand on your earlier theoretical justification of ties? Alternatively, does your theoretical justification seem reasonable in light of new information obtained from these graphs?

3. It is possible to observe, that with the rise of the meaning of cutpoint, the amount of edges drop. It is reasonable, the types of international relations have different grades. Moreover, it is interesting, that with the rise of meaning of cutpoint, the amount of directed links rise. It is possible to see, that developed countries have more one-way pointed ties. We can prove, that countiries with more different resources try to help a less developed ones.

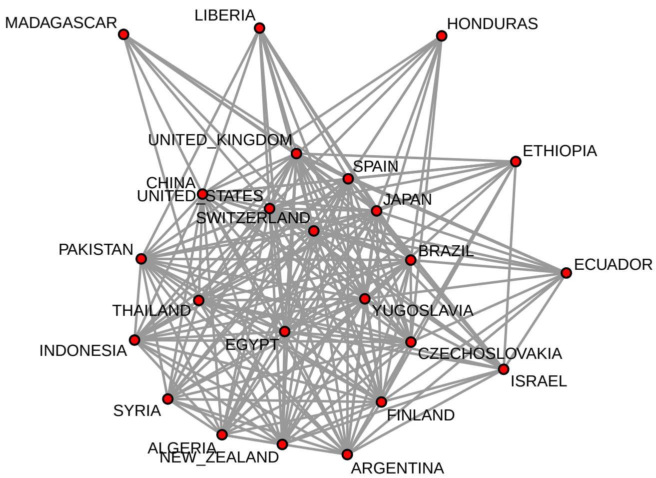
4. From the analysis of that graphs I can tell, that the theoretical justification is not too perfect for the ties. The theory can't completely explain why stronger countries help less developed ones. We can suppose that in that way they somehow collect any type of symbolic resource or gain power over other countris. In that way, I think, it is possible to expand the earlier theoretical frames.

```
detach(package:igraph)
library(network)
```

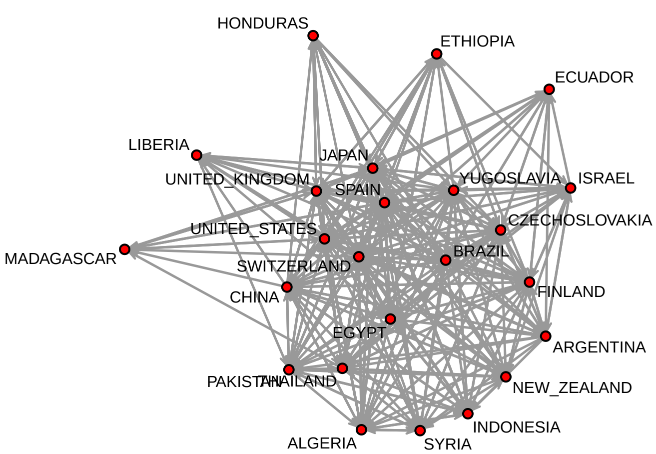
```
## network: Classes for Relational Data
```

```
## Version 1.13.0.1 created on 2015-08-31.
## 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.
```

```r
tradenet.sym.2<- network(trade.2, directed=FALSE)
plot(tradenet.sym.2,
displaylabels=TRUE,
label.cex =.5,
edge.col = 'gray60')
```




```r
tradenet2 <- network(trade.2)
plot(tradenet2,
displaylabels=TRUE,
label.cex =.5,
edge.col = 'gray60')
```



```r
tradenet.any<-as.network(trade.any)

network.density(tradenet.any)
```

```
## [1] 0.8097826
```

```r
network.density(tradenet.sym.2)
```

```
## [1] 0.7536232
```

```r
network.density(tradenet2)
```

```
## [1] 0.6666667
```

```r
library(igraph)
```

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

## 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
```

```r
graph.density(tradegraph.5)
```

```
## [1] 0.1811594
```

```r
diameter(tradegraph.any)
```

```
## [1] 2
```

```r
diameter(tradegraph.2)
```

```
## [1] 2
```

```r
diameter(tradegraph.5)
```

```
## [1] 3
```

Assignment question. Of course, there are differences between directed and undirected networks on the graph and with stats. Please answer the following questions: 1. What are the differences in graphs and how would you interpret them? 2. What are the differences in centrality? 3. What is the diameter and how do you expect it to wary?

1. The same results have been made earlier. With the rise of meaning of cutpoint, the amount of ties drops. That is why the measure of density also drops. To the amount of possible ties, the amount of all real ties decreases and the amount of nodes stays the same. It tells us, that not all the relations between the countries are strong, some connections are weaker, and if we take them of the network, the network will have only the strongest ones.

2. The difference in centrality means, that with the reductions of ties in the network, the amount of all connections slowly falls, so the network gets less complex. The more ties the network has, the more complex and more "compleate" it is.

3. The diameter is the largest geodesic within the graph between any pair of nodes. It can indicate how large and tight the network is. So, it is natural, that the value of diameter rises: the amount of ties shortens, so the lenghts of geodesics rises.

```
detach(package:igraph)
library(sna)
```

```
## sna: Tools for Social Network Analysis
## Version 2.4 created on 2016-07-23.
## copyright (c) 2005, Carter T. Butts, University of California-Irvine
##  For citation information, type citation("sna").
##  Type help(package="sna") to get started.
```

```
tradenet5 <- network(trade.max)

components(tradenet.any)
```

```
## Node 1, Reach 24, Total 24
## Node 2, Reach 24, Total 48
## Node 3, Reach 24, Total 72
## Node 4, Reach 24, Total 96
## Node 5, Reach 24, Total 120
## Node 6, Reach 24, Total 144
## Node 7, Reach 24, Total 168
## Node 8, Reach 24, Total 192
## Node 9, Reach 24, Total 216
## Node 10, Reach 24, Total 240
## Node 11, Reach 24, Total 264
## Node 12, Reach 24, Total 288
## Node 13, Reach 24, Total 312
## Node 14, Reach 24, Total 336
```

```
## Node 15, Reach 24, Total 360
## Node 16, Reach 24, Total 384
## Node 17, Reach 24, Total 408
## Node 18, Reach 24, Total 432
## Node 19, Reach 24, Total 456
## Node 20, Reach 24, Total 480
## Node 21, Reach 24, Total 504
## Node 22, Reach 24, Total 528
## Node 23, Reach 24, Total 552
## Node 24, Reach 24, Total 576
```

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

```r
components(tradenet2)
```

```
## Node 1, Reach 24, Total 24
## Node 2, Reach 24, Total 48
## Node 3, Reach 24, Total 72
## Node 4, Reach 24, Total 96
## Node 5, Reach 24, Total 120
## Node 6, Reach 24, Total 144
## Node 7, Reach 24, Total 168
## Node 8, Reach 24, Total 192
## Node 9, Reach 24, Total 216
## Node 10, Reach 24, Total 240
## Node 11, Reach 24, Total 264
## Node 12, Reach 24, Total 288
## Node 13, Reach 24, Total 312
## Node 14, Reach 24, Total 336
## Node 15, Reach 24, Total 360
## Node 16, Reach 24, Total 384
## Node 17, Reach 24, Total 408
## Node 18, Reach 24, Total 432
## Node 19, Reach 24, Total 456
## Node 20, Reach 24, Total 480
## Node 21, Reach 24, Total 504
## Node 22, Reach 24, Total 528
## Node 23, Reach 24, Total 552
## Node 24, Reach 24, Total 576
```

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

```r
components(tradenet5)
```

```
## Node 1, Reach 1, Total 1
## Node 2, Reach 24, Total 25
## Node 3, Reach 24, Total 49
```

```
## Node 4, Reach 24, Total 73
## Node 5, Reach 24, Total 97
## Node 6, Reach 24, Total 121
## Node 7, Reach 24, Total 145
## Node 8, Reach 1, Total 146
## Node 9, Reach 24, Total 170
## Node 10, Reach 1, Total 171
## Node 11, Reach 24, Total 195
## Node 12, Reach 1, Total 196
## Node 13, Reach 24, Total 220
## Node 14, Reach 1, Total 221
## Node 15, Reach 1, Total 222
## Node 16, Reach 24, Total 246
## Node 17, Reach 1, Total 247
## Node 18, Reach 24, Total 271
## Node 19, Reach 24, Total 295
## Node 20, Reach 1, Total 296
## Node 21, Reach 24, Total 320
## Node 22, Reach 24, Total 344
## Node 23, Reach 24, Total 368
## Node 24, Reach 24, Total 392
```

```
## [1] 9
```

Assignment question. What are the differences between the three networks? How would you explain them from the theoretical level?

The differences between the networks are, as told above, that the networks with the stronger ties with the same amount of nodes have less edges at all. Also, the third network lets us see, that there are a lot of directed ties from developed countries to the all other ones. That can be interpreted as strong countries bring help to the less developed ones. As interpreted earlier, it in some way responds to the theory: stronger countries try to establish power over the weaker countries.

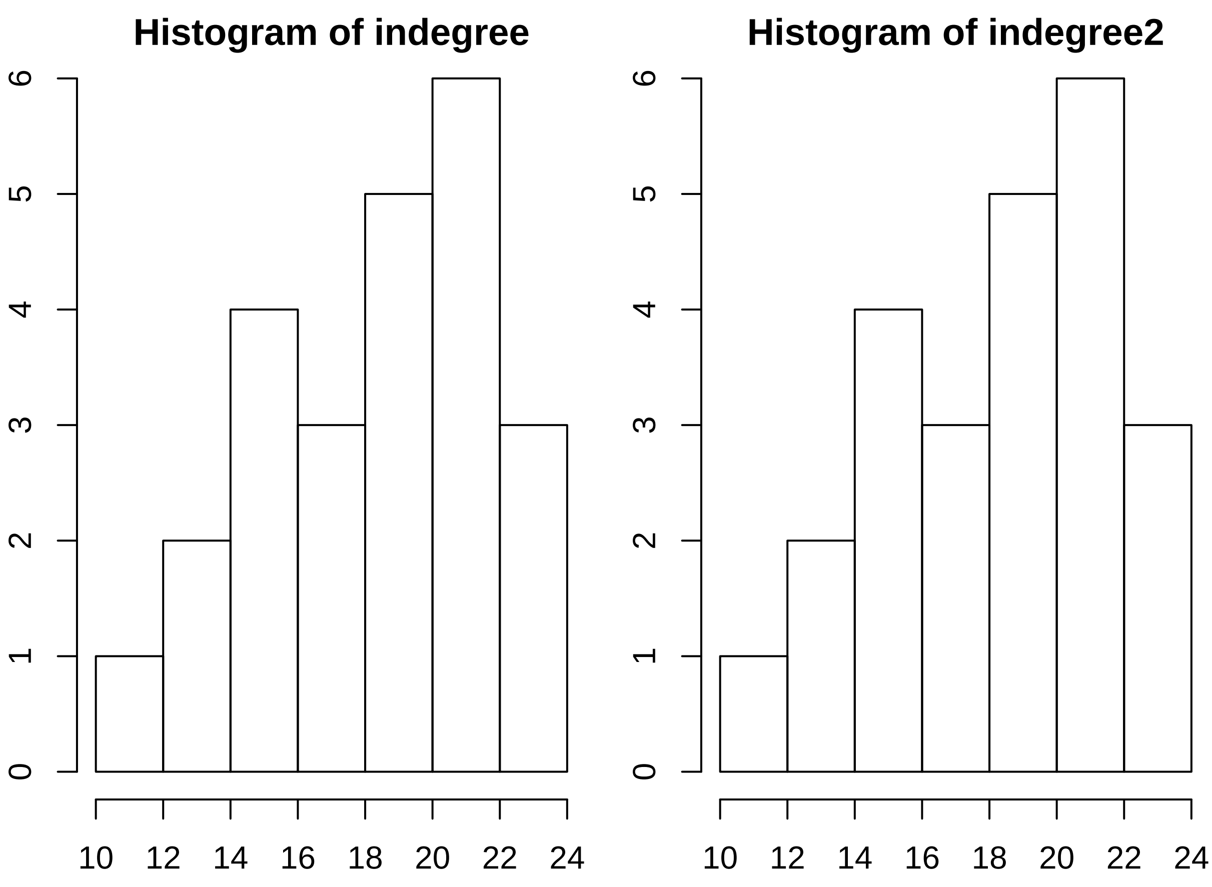
```
geo.dist <-geodist(tradenet.sym.2)
geo.dist.dir <-geodist(tradenet2)
summary(geo.dist.dir)
```

```
##        Length Class  Mode
## counts 576    -none- numeric
## gdist  576    -none- numeric
```

```
summary(geo.dist)
```

```
##        Length Class  Mode
## counts 576    -none- numeric
## gdist  576    -none- numeric
```

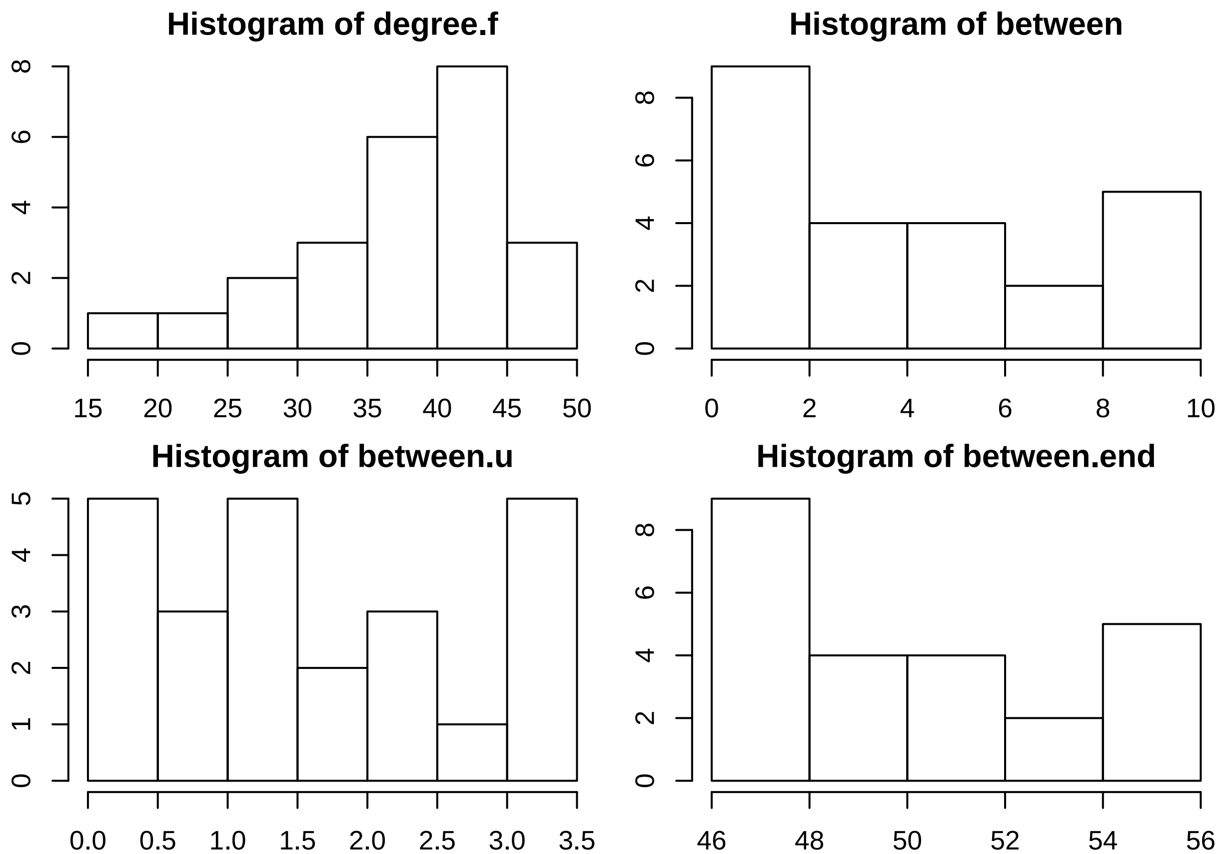
```
tradenet.valued <- as.network(trade.all,directed=TRUE)
trade.att.valued <- trade.att

indegree <- degree(tradenet.valued,
gmode = 'digraph',

diag = FALSE,
cmode = 'indegree',
rescale = FALSE,
ignore.eval = FALSE)

indegree2 <- degree(tradenet.valued,
gmode = 'digraph',
diag = FALSE,
cmode = 'indegree',
rescale = FALSE,
ignore.eval = TRUE)

par(mar=c(2,2,1,1),mfrow=c(1,2))
hist(indegree)
hist(indegree2)
```



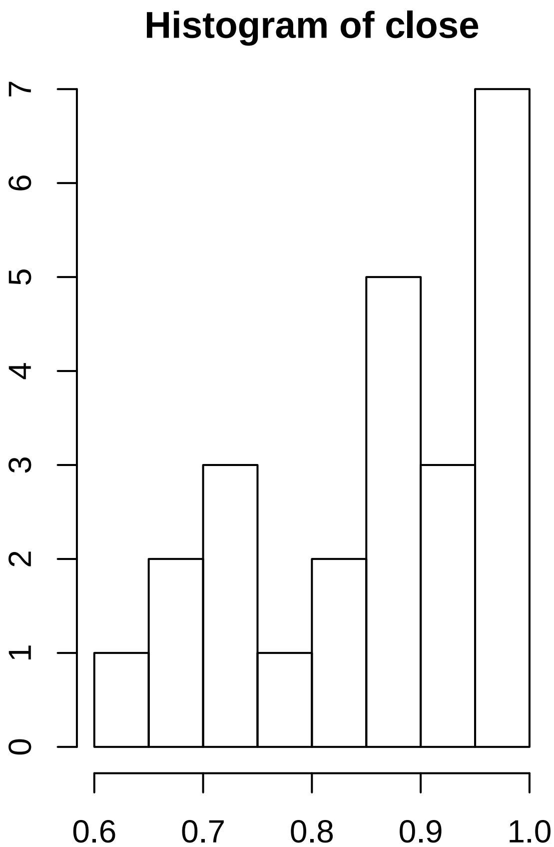
Assignment question. Is there a difference between valued data and non-valued data for degree centrality? Why?

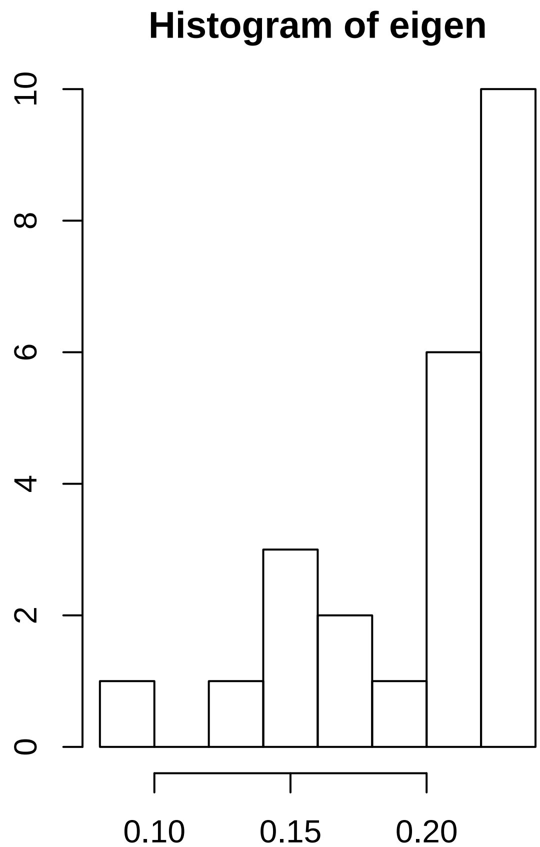
There is no difference for degree centrality because the degree is just the amount of ties the nodes have. They are not related in any way to the types of the edges or nodes: degree only shows the amount of the ties.

```
outdegree <- degree(tradenet.valued,
gmode = 'digraph',
diag = FALSE,
cmode = 'outdegree',
rescale = FALSE)

# Freeman's degree (in + out):
degree.f <- degree(tradenet.valued,
gmode = 'digraph',
diag = FALSE,
cmode = 'freeman',
rescale = FALSE)

between <- betweenness(tradenet.valued,
gmode = 'digraph',
diag = FALSE,
cmode = 'directed')

between.u <- betweenness(tradenet.valued,
gmode = 'digraph',
diag = FALSE,
cmode = 'undirected')

between.end <- betweenness(tradenet.valued,
gmode = 'digraph',
diag = FALSE,
cmode = 'endpoints')

between.proxi <- betweenness(tradenet.valued,
gmode = 'digraph',
diag = FALSE,
cmode = 'proximalsrc')
par(mar=c(2,2,2,1),mfrow=c(2,2)) # Create a 2x2 matrix of plots

hist(degree.f)
hist(between)
hist(between.u)
hist(between.end)
```




```
# Closeness
close <- closeness(tradenet.valued,
gmode = 'digraph',
# gmode ='graph',
diag = FALSE,
cmode = 'directed',
rescale = FALSE)

#Eigenvalue
eigen <- evcent(tradenet.valued,
gmode = 'digraph',
diag = FALSE,
rescale = FALSE)
par(mar=c(2,2,2,2),mfrow=c(1,2))
hist(close)
hist(eigen)
```





Assignment question. Why do some of centrality histograms look the same while others look so different? What do they each show us?

The centralities histograms are different because all of them are based on the different ways of calculating measure. The histograms show the distribution of values of centralities for each node. Some centralities, like degree and closeness, are dependant on the measure of how tight the nodes are connected. While the centralities of betweenes and Eigenvalue are dependent to the how far from each other the nodes are. That is why those measures are lower.

Based on that measures we can say that the network is a tight one, that is why degree and closeness are higher in general than betweenes and Eigenvalue.

```
trade.att.valued <- cbind(trade.att.valued, indegree, outdegree, degree.f, between, close, eigen)
class(trade.att.valued)
```

```
## [1] "data.frame"
```

```
names(trade.att.valued)
```

```
##  [1] "POP_GROWTH" "GNP"        "SCHOOLS"    "ENERGY"     "indegree"
##  [6] "outdegree"  "degree.f"   "between"    "close"      "eigen"
```

```
centralization(tradenet.valued, FUN ='degree',
normalize = TRUE)
```

[1] 0.2075099

```
centralization(tradenet.valued, FUN ='betweenness',
normalize = TRUE)
```

[1] 0.0113248

```
centralization(tradenet.valued, FUN ='closeness',
normalize = TRUE)
```

[1] 0.1560584

```
centralization(tradenet.valued, FUN ='evcent',
normalize = TRUE)
```

[1] 0.03954187

Assignment question. What do indexes above mean? What do they tell us about our network?

The indexes above are the different measures for calculating a centralization for the network "tradenet.valued". The each measure might tell us about the variuos charachteristics of the network

The degree centralization measure is based on the amount of ties each node has. It is the highest measure of centralization of all above, so we can tell, that all the nodes have a lot of connections with each other.

The closeness centralization measure is based on the lenhts of geodesics each node has. Based on this measure, the network is a really tight one: all the nodes are relatively close to each other.

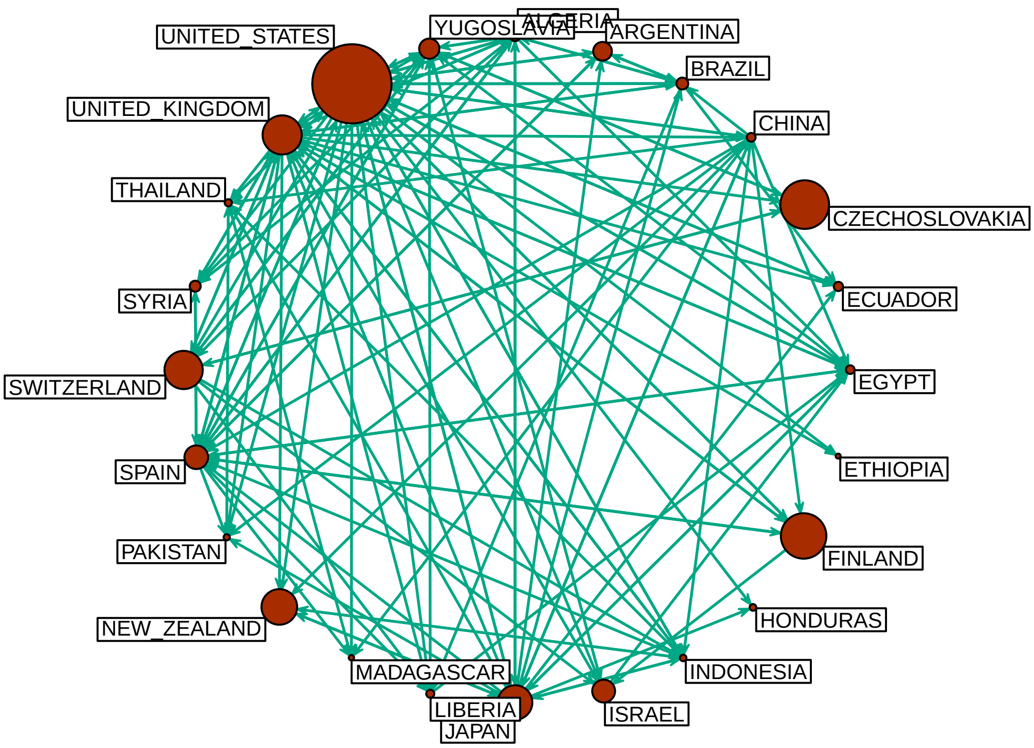
The betweenness centralization measure is based on the amount of nodes of geodesics each node has, less amount - more measure. Based on this measure, the network is a actually a close one: all the nodes are close to each other.

The evcent centralization measure is based on the proportion of own centrality of the node to the centralities of other nodes. Thsi measure proves the result of the other measures: all the nodes are close to each other.

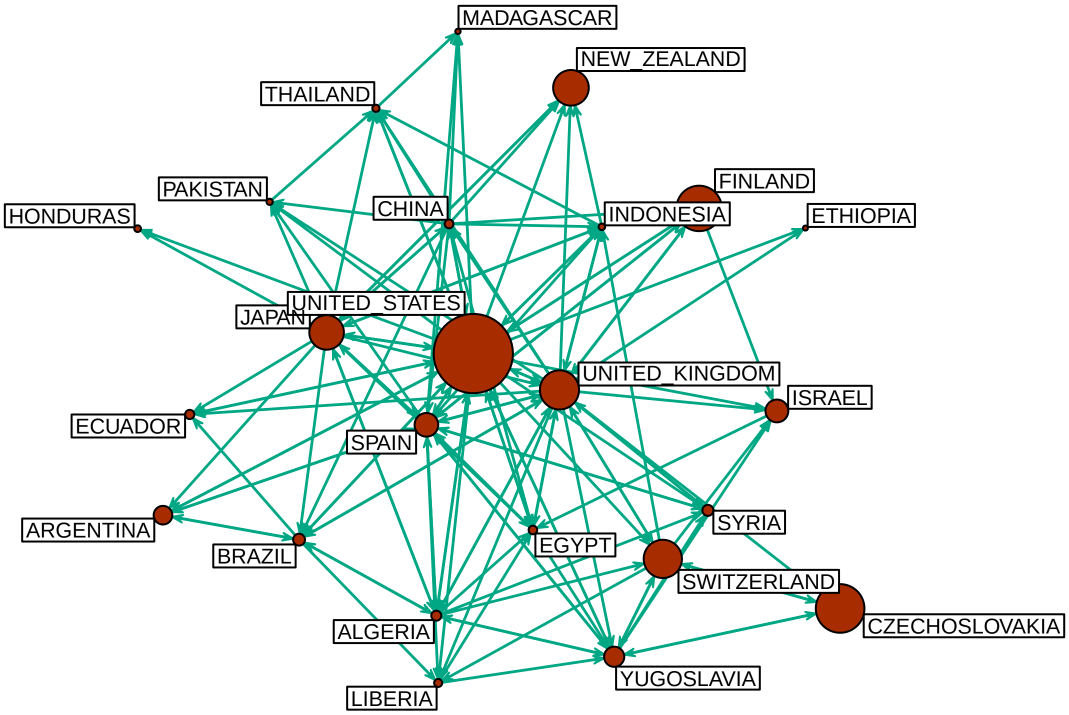
Final assignment task. There are several networks in the ???trade.Rdata??? file, described above. We have fully explored the ???trade.all??? network. Now, select one of the individual trade networks (manufacture, food, crude, etc.) and show me everything you???ve learned in this class so far. At the very minimum, please do the following: 1. Create an appropriate graph with all possible options. 2. Generate all possible network measures. 3. Tell me what inferences you can make about your selected network based on the information you???ve obtained. Supplement your arguments with logic and theory.

```
minerals <- network(minerals, directed=TRUE)

set.vertex.attribute(minerals, attrname='Energy', value=trade.att[,4])
```

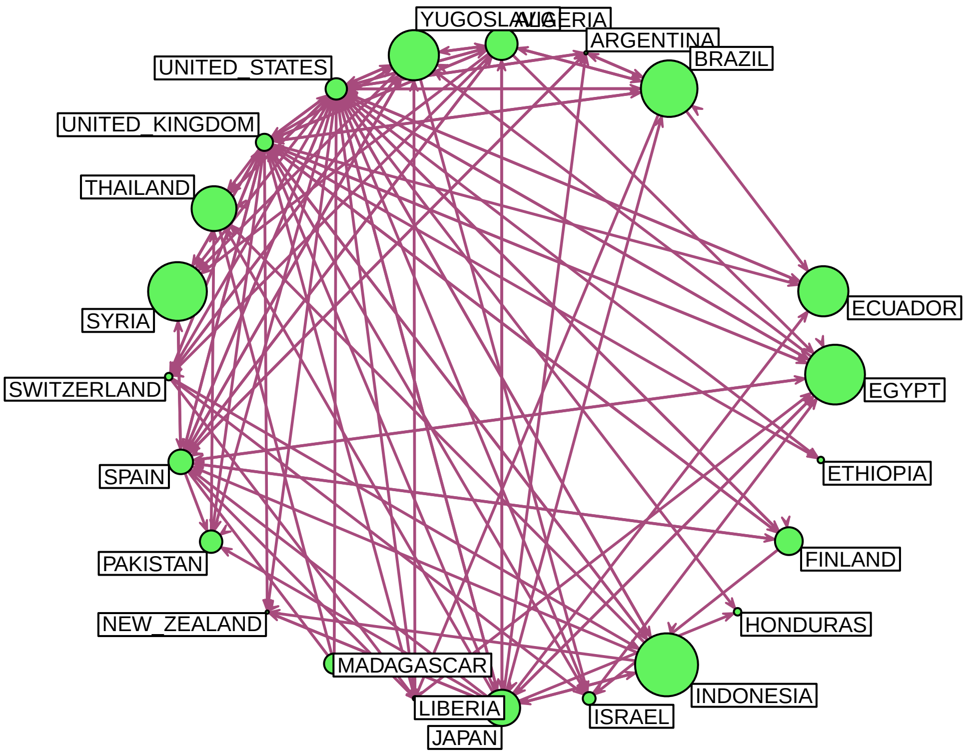
```r
set.vertex.attribute(minerals, attrname='GNP', value=trade.att[,2])


par(mar=c(0,0,0,0))
plot(minerals,
    displaylabels=TRUE,
    label.cex =.7,
    edge.col = '#00A784',
    vertex.col = "#A72C00",
    arrowhead.cex = 0.7,
    vertex.cex = (get.vertex.attribute(minerals, 'Energy')/2500+0.3),
    boxed.labels =TRUE,
    mode = 'circle')
```



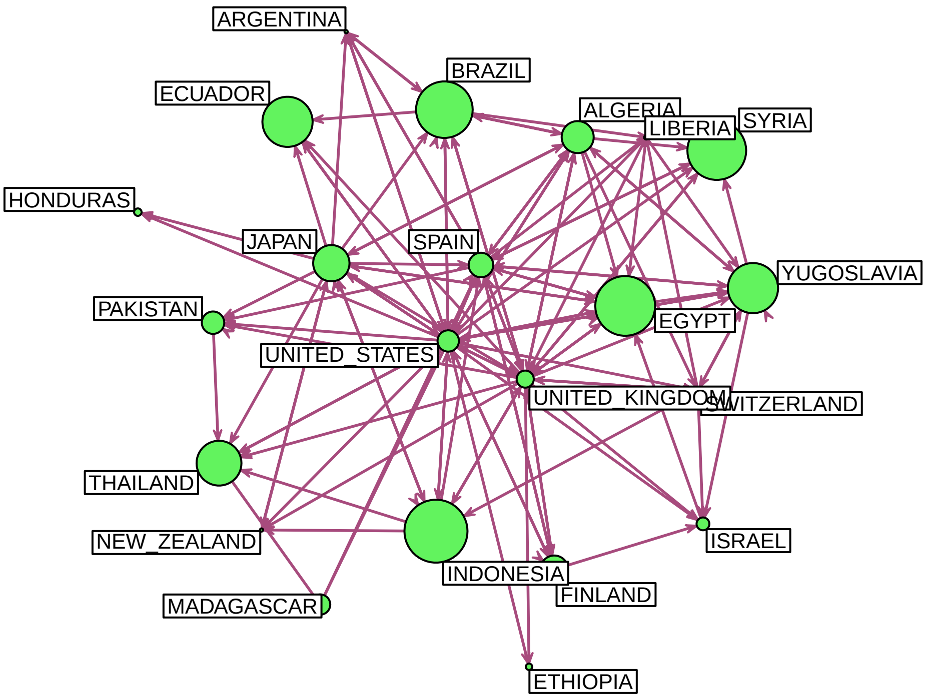

```r
par(mar=c(0,0,0,0))
plot(minerals,
    displaylabels=TRUE,
    label.cex =.7,
    edge.col = '#00A784',
    vertex.col = "#A72C00",
    arrowhead.cex = 0.7,
    vertex.cex = (get.vertex.attribute(minerals, 'Energy')/2500+0.3),
    boxed.labels =TRUE,
    mode = 'kamadakawai')
```




```r
par(mar=c(0,0,0,0))
plot(minerals,
    displaylabels=TRUE,
    label.cex =.7,
    edge.col = '#A74B7D',
    vertex.col = "#62F35E",
    arrowhead.cex = 0.7,
    vertex.cex = (get.vertex.attribute(minerals, 'GNP')/1.5),
    boxed.labels =TRUE,
    mode = 'circle')
```




```r
par(mar=c(0,0,0,0))
plot(minerals,
    displaylabels=TRUE,
    label.cex =.7,
    edge.col = '#A74B7D',
    vertex.col = "#62F35E",
    arrowhead.cex = 0.7,
    vertex.cex = (get.vertex.attribute(minerals, 'GNP')/1.5),
    boxed.labels =TRUE,
    mode = 'kamadakawai')
```




```r
network.dyadcount(minerals)
```

```
## [1] 552
```

```r
network.edgecount(minerals)
```

```
## [1] 135
```

```r
network.density(minerals)
```

```
## [1] 0.2445652
```

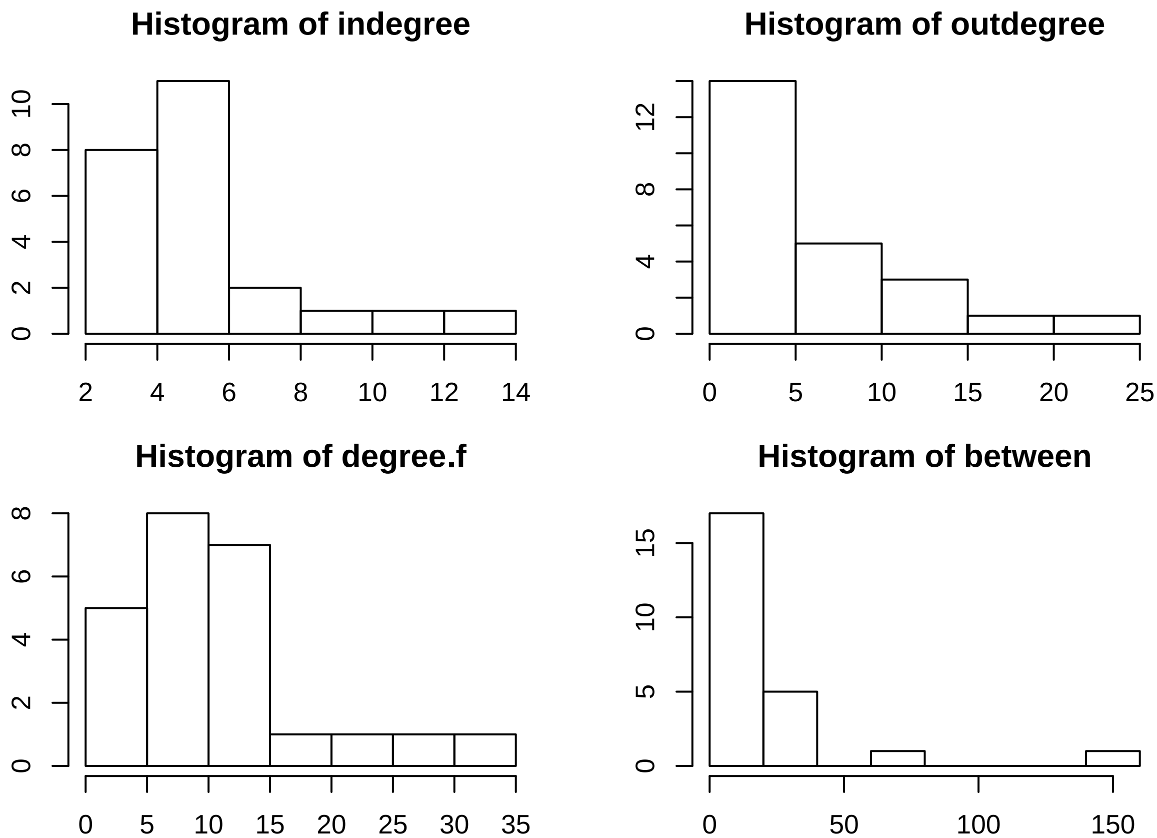
```r
components(minerals)
```

```
## Node 1, Reach 24, Total 24
## Node 2, Reach 24, Total 48
## Node 3, Reach 24, Total 72
## Node 4, Reach 24, Total 96
## Node 5, Reach 24, Total 120
## Node 6, Reach 24, Total 144
## Node 7, Reach 24, Total 168
## Node 8, Reach 1, Total 169
## Node 9, Reach 24, Total 193
## Node 10, Reach 1, Total 194
## Node 11, Reach 24, Total 218
## Node 12, Reach 24, Total 242
## Node 13, Reach 24, Total 266
## Node 14, Reach 24, Total 290
```

```
## Node 15, Reach 1, Total 291
## Node 16, Reach 24, Total 315
## Node 17, Reach 24, Total 339
## Node 18, Reach 24, Total 363
## Node 19, Reach 24, Total 387
## Node 20, Reach 24, Total 411
## Node 21, Reach 24, Total 435
## Node 22, Reach 24, Total 459
## Node 23, Reach 24, Total 483
## Node 24, Reach 24, Total 507
```

```
## [1] 4
```

```r
indegree <- degree(minerals,
  gmode = 'digraph',
  diag = FALSE,
  cmode = 'indegree',
  rescale = FALSE,
  ignore.eval = FALSE)

outdegree <- degree(minerals,
  gmode = 'digraph',
  diag = FALSE,
  cmode = 'outdegree',
  rescale = FALSE)

# Freeman's degree (in + out):
degree.f <- degree(minerals,
  gmode = 'digraph',
  diag = FALSE,
  cmode = 'freeman',
  rescale = FALSE)

between <- betweenness(minerals,
  gmode = 'digraph',
  diag = FALSE,
  cmode = 'directed')

par(mar=c(2,2,3,3),mfrow=c(2,2))
hist(indegree)
hist(outdegree)
hist(degree.f)
hist(between)
```




```r
centralization(minerals, FUN ='degree',
  normalize = TRUE)
```

```
## [1] 0.5632411
```

```r
library(network)
centralization(minerals, FUN ='betweenness',
  normalize = TRUE)
```

```
## [1] 0.2864698
```

```r
centralization(minerals, FUN ='closeness',
  normalize = TRUE)
```

```
## [1] 0.5070737
```

```r
centralization(minerals, FUN ='evcent',
  normalize = TRUE)
```

```
## [1] 0.3039906
```

For the Final assignment task I have built various graphs of the netwotk "minerals", that contains the information about the trading relations of energetic resources. I have built 4 graphs: two, including attribute of energy potential of each country and two including attribute of GNP of each country. Those graphs lets us make some interesting conclusions.

It is possible to see that the countries with the larger amount of energy sell it to the weaker countries. It can be concluded that cuontries try to exchange the resources they contain for some other resources and that is quite a rational practice. Moreover, it can be observed, and the most amount of ties have countries like USA and The UK, those are countries with the most energy and they sell to the other ones.

The one more thing that can be told is that the energy in those countries might be coming to the rising temps of national economics. In less developed countries the level of GNP is much higher that of the developed countries.

What for all the measures that have been built, that can tell us, that the network has quite high level ties, the measures of centralization of closeness and degree are about 0.5. That means that in general all nodes are conected to each other on a quite medium level.

In conclusion, that graphs of network, that have been built allow us to make some assumptions on the way the global world operates. The developed countries sell the energe to the less developed ones, and partly thanks to that, they are able to achive high levels of GNP and rise the levels of national economy.