Statistical Computing with R: Masters in Data Science 503 (S14) First Batch, SMS, TU, 2021

Shital Bhandary

Associate Professor

Statistics/Bio-statistics, Demography and Public Health Informatics

Patan Academy of Health Sciences, Lalitpur, Nepal

Faculty, Data Analysis and Decision Modeling, MBA, Pokhara University, Nepal

Faculty, FAIMER Fellowship in Health Professions Education, India/USA.

Review Preview

- Basic graphics/plots:
  - Bar chart
  - Histogram
  - Density plot
  - Pie chart
  - Line chart
  - Scatterplot
  - Boxplot etc.

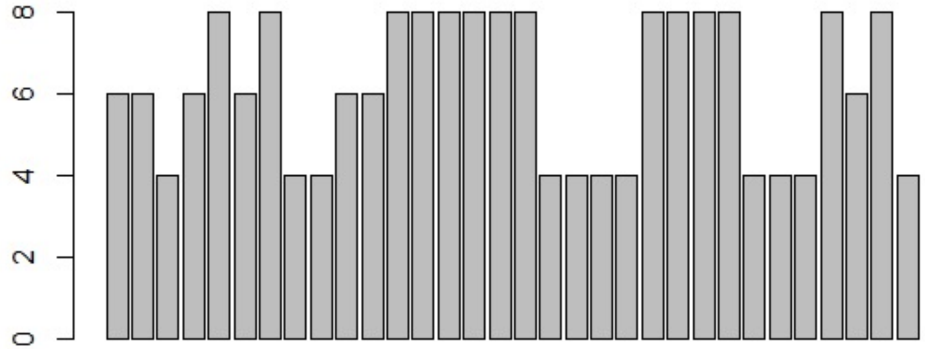
- Special graph:
  - Social Network Analysis

How to get bar diagram from data frame?

- df <- as.data.frame(mtcars)

#Bar plot of cylinder data

- barplot(df$cyl)



- This barplot shows the number of cylinders for 32 cars of the dataset

- **Do we want this?**

How to get bar diagram from data frame?

- df <- as.data.frame(mtcars)

# We need frequencies of cares with 4, 6 and 8 cylinders

- table(df$cyl)

#Bar plot of freq. of cylinder data

- barplot(table(df$cyl))

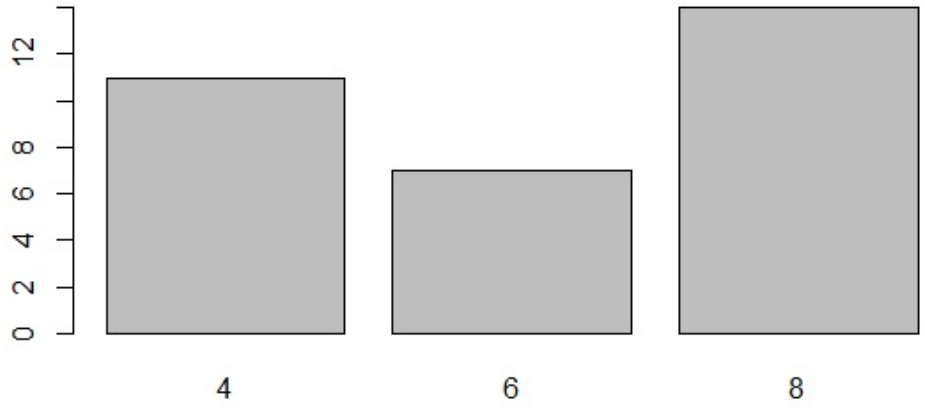
OR

#We can assign this as object

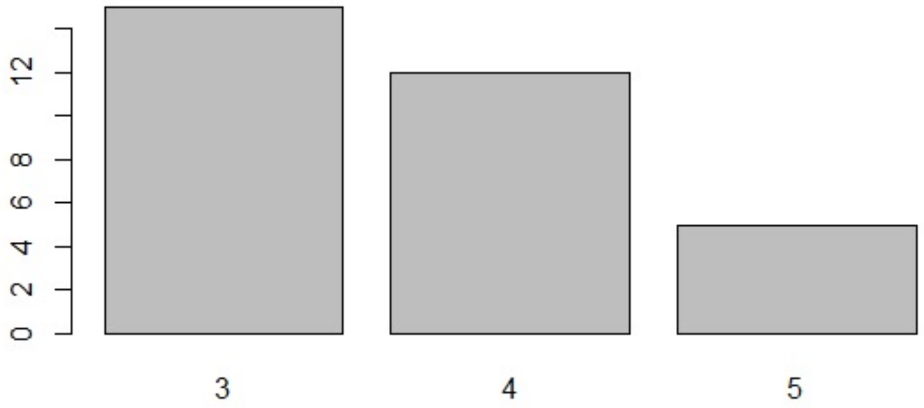
- bpd <- table(df$cyl)

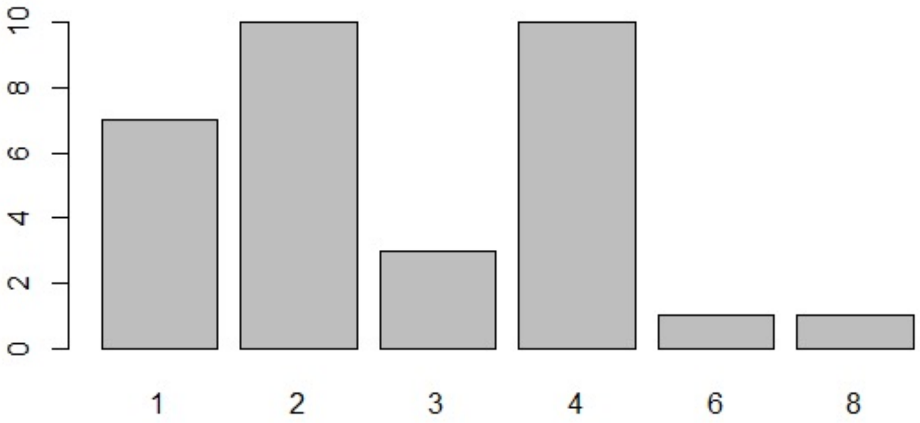
#Get the barplot

- barplot(bpd)



We can get the barplot of “gear” and “carb” too as they are factors (categorical variables)





Try to get barplot after declaring the “gear” and “carb” variables as factors too.

Check the structure of the data frame first!

Barplot of “mpg” variable: How to get it? mpg: miles per gallon (continuous variable)

#MPG – range for class interval

- range(df$mpg)
- R = 33.9 - 10.4 #23.5
- I = round(sqrt(R)) # 5

#We need to construct 5 classes with width of 5 (10, 15, 20, 25, 30)

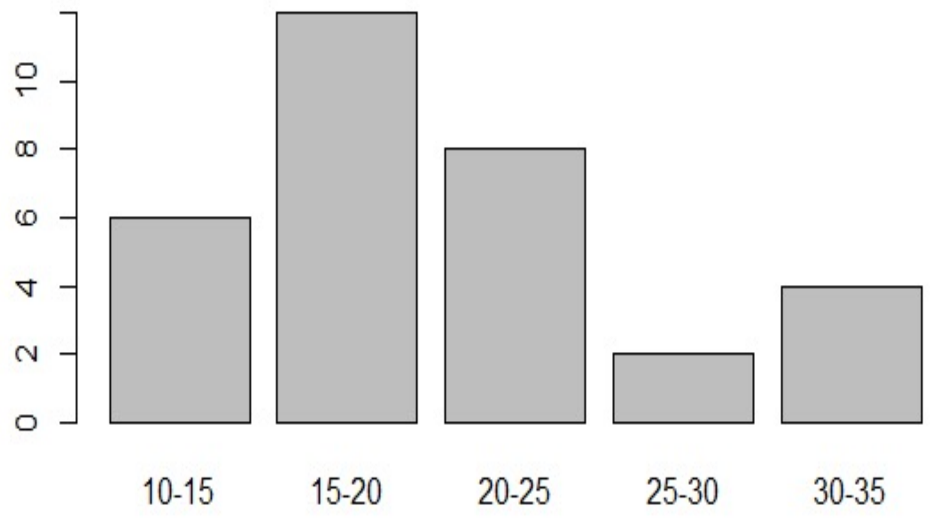
#We need to define the breaks

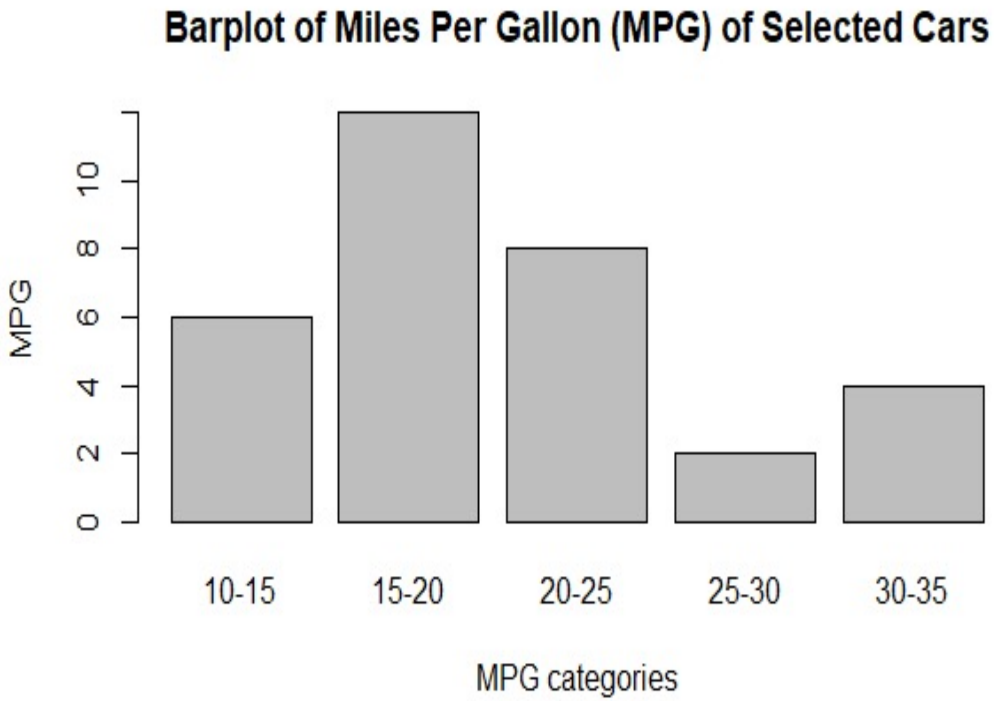
breaks = c(10, 15, 20, 25, 30, 35) or

breaks = seq(10, 35, by=5)

- mpg.bin <- cut(df$mpg, breaks, labels = c("10-15", "15-20", "20-25", "25-30", "30-35"))
- mpg.bin
- table(mpg.bin)
- mpg.bin.freq <- table(mpg.bin)

Outputs:



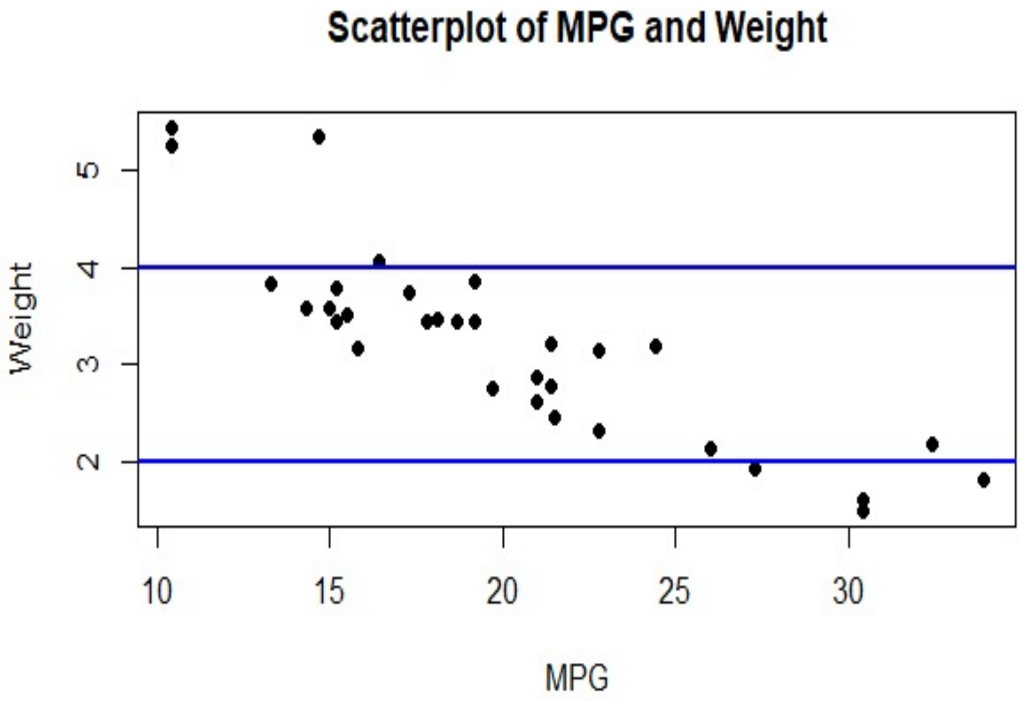


What to do if we want use and show the "inclusive" class intervals?

Scatterplot with horizontal “abline”:

```
#Scatterplot with abline
plot(df$mpg, df$wt, pch=16, main = "Scatterplot of MPG and Weight", xlab = "MPG", ylab = "Weight")
abline(h=2, col = "blue", lwd=2)
abline(h=4, col = "blue", lwd=2)
```

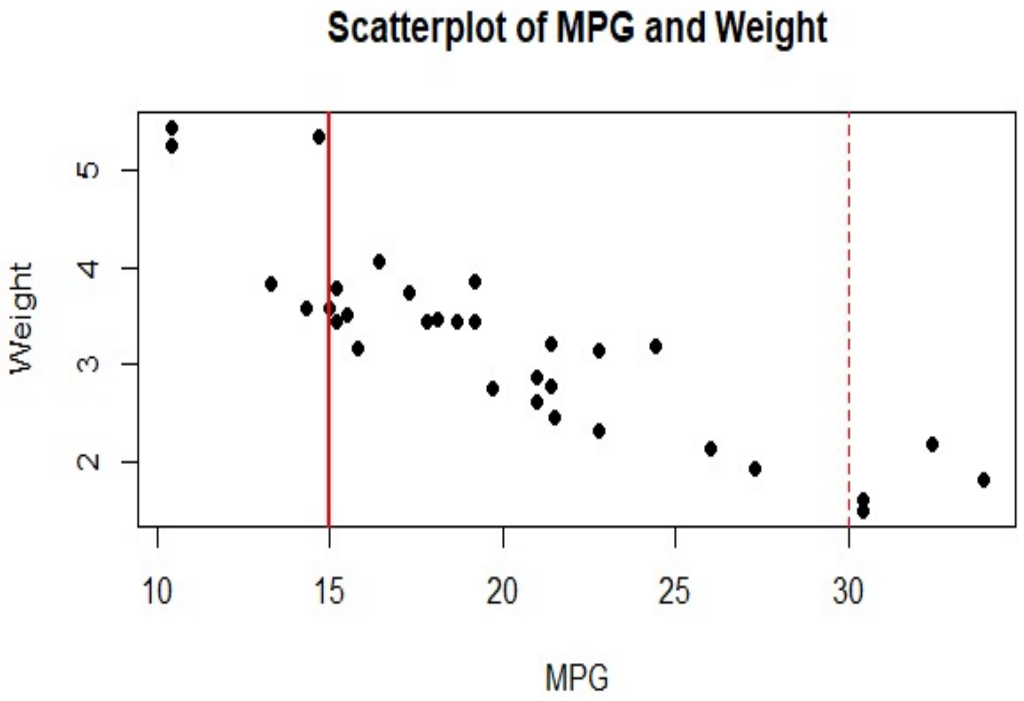
Here, h = horizontal line in y-axis and lwd = line width parameter



Scatterplot with vertical “abline”:

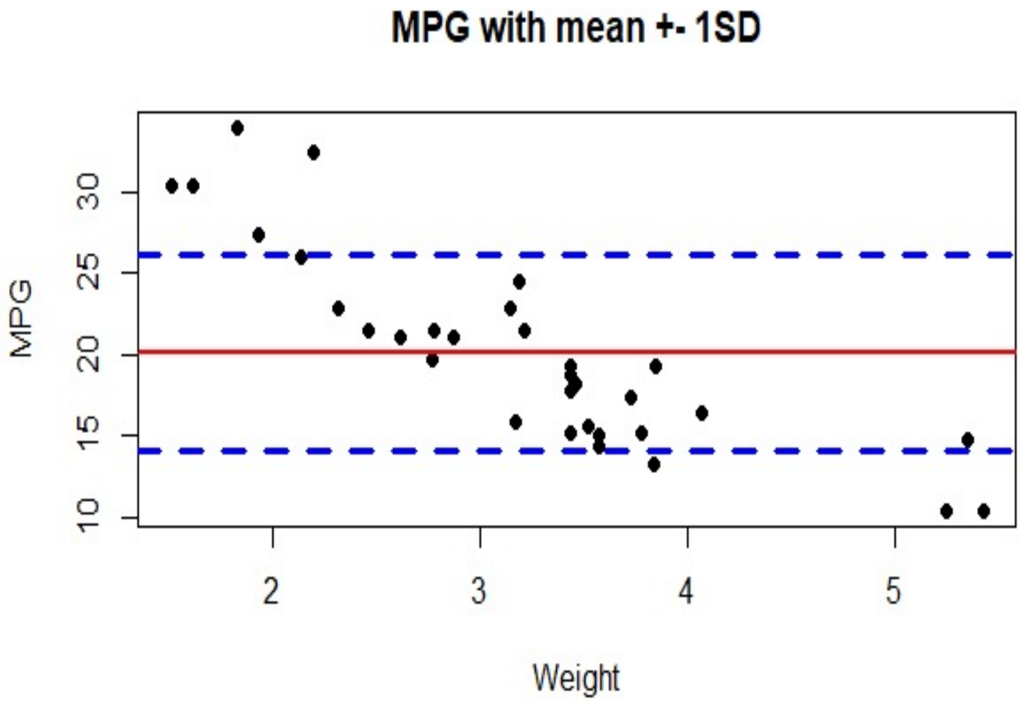
- plot(dfmpg, dfwt, pch=16, main = "Scatterplot of MPG and Weight", xlab = "MPG", ylab = "Weight")
- abline(v=15, col = "red", lwd=2)
- abline(v=30, col = "red", lty=2)

- Here, v=Vertical line at x-axis and lty = line type parameter



Scatterplot with mean ± 1*sd of y-variable:

- plot(dfwt, dfmpg, pch=16)
- abline(h=mean(df$mpg), lwd = 2, col = "red")
- abline(h=mean(df$mpg) + 1*sd(df$mpg), col = "blue", lwd=3, lty = 2)
- abline(h=mean(df$mpg) - 1*sd(df$mpg), col = "blue", lwd=3, lty = 2)



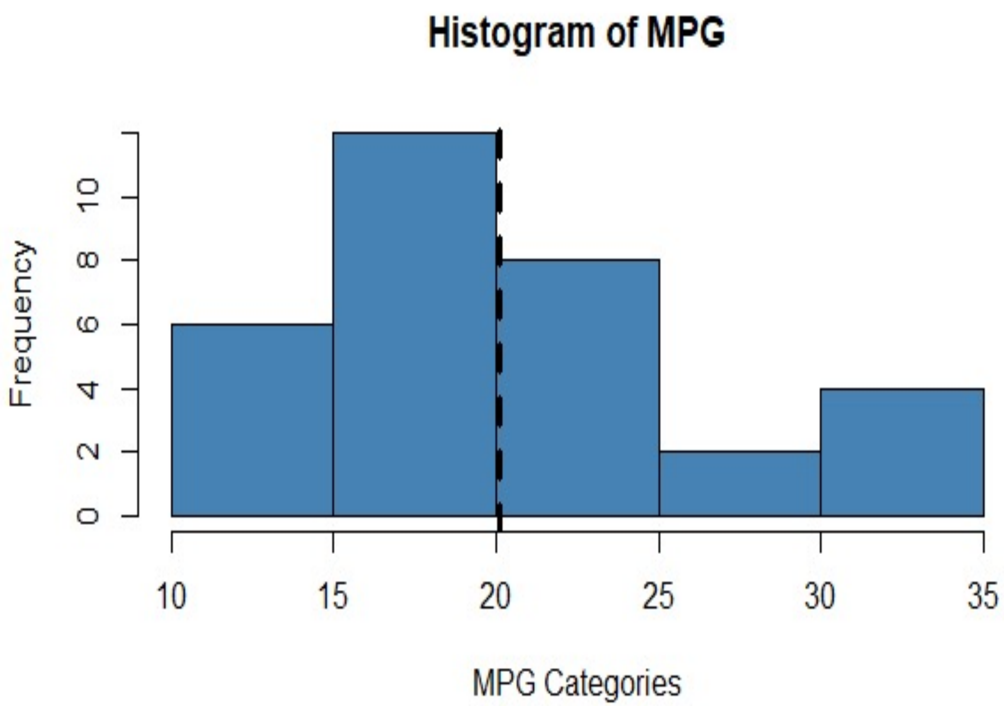
Try to add mean ± 2*sd of mpg in this scatterplot!

Can you see both the bands? If not, why?

Histogram and abline:

- hist(df$mpg, col = "steelblue", main = "Histogram of MPG", xlab = "MPG Categories")
- abline(v=mean(df$mpg), lwd=3, lty=2)

- Note: Histogram can be used to located “mode” of the numerical variable!



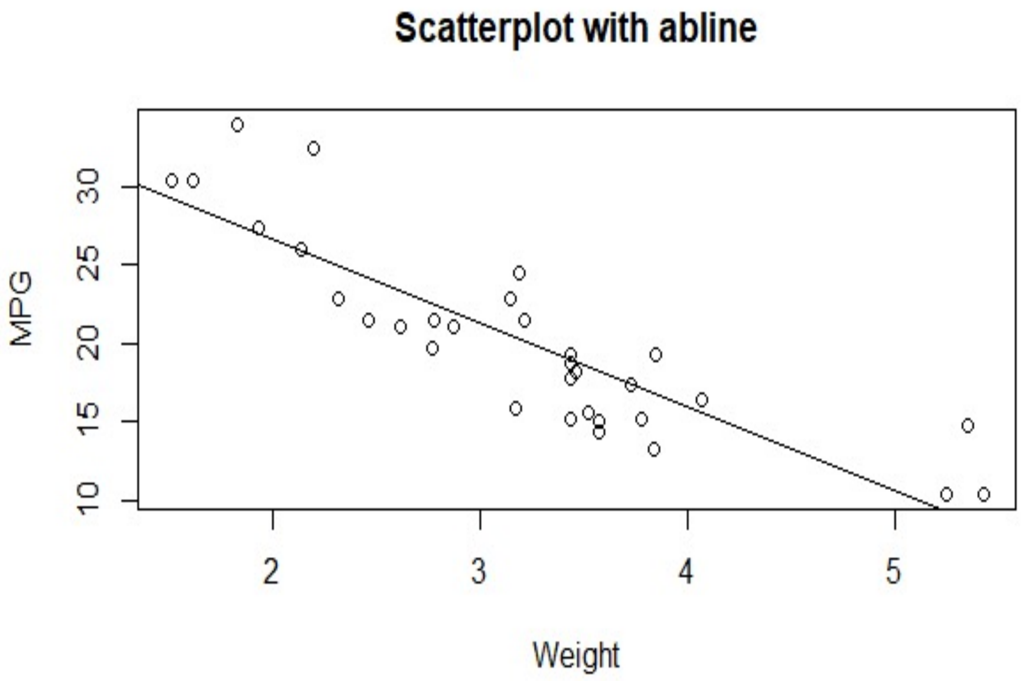
Which graph/s must be used to locate the “median” of the numerical variable graphically?

Scatterplot with “abline” from a model:

- plot(dfwt, dfmpg, main = "Scatterplot with abline", xlab = "Weight", ylab = "MPG")
- reg_mod <- lm(df$mpg ~ df$wt)
- abline(reg_mod)

OR

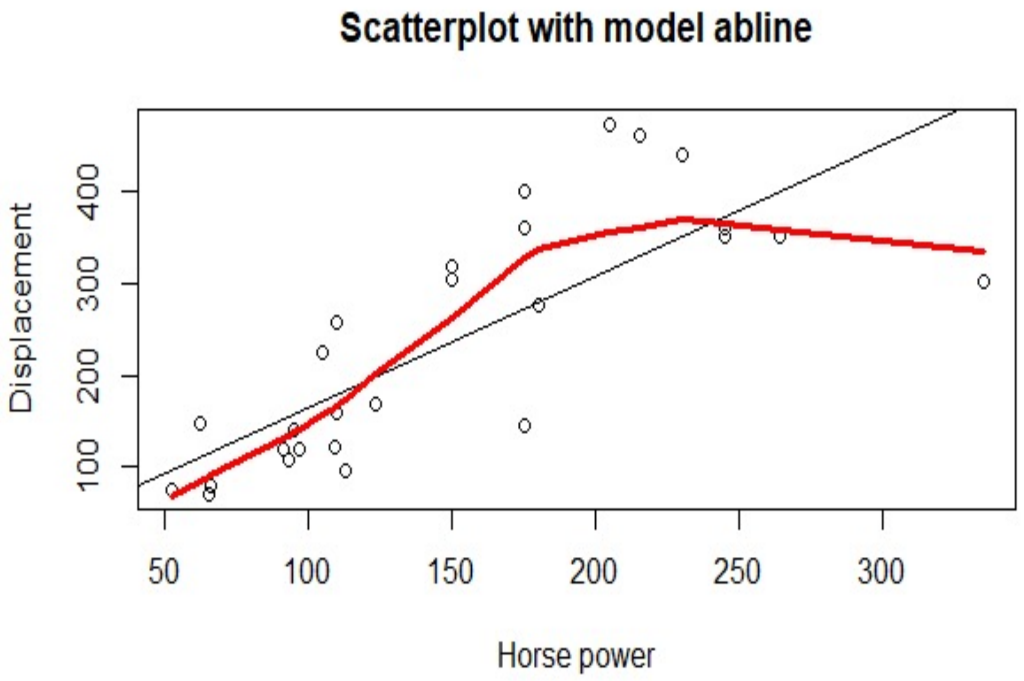
- plot(dfwt, dfmpg, main = "Scatterplot with abline", xlab = "Weight", ylab = "MPG")
- abline(lm(df$mpg ~ df$wt))



Scatterplot with “abline” and “lines” for a non-linear data:

- plot(dfhp, dfdisp, main = "Scatterplot with model abline", xlab = "Horse power", ylab = "Displacement")
- abline(lm(df$disp ~ df$hp))
- lines(lowess(dfhp, dfdisp), col = "red", lwd = 3)

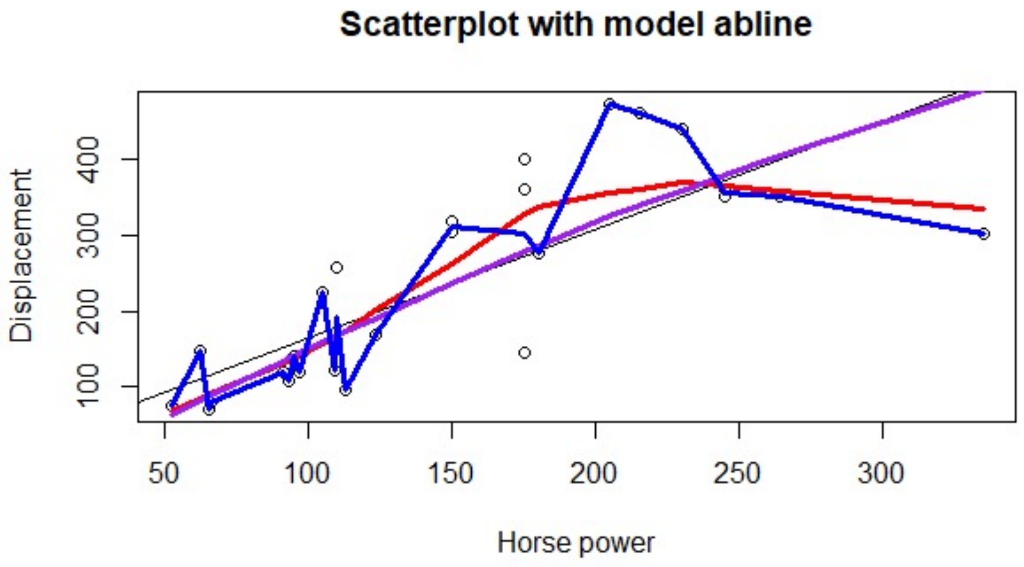
- Lowess = Locally weighted Scatterplot Smoothing



Show general additive model, quadratic and cubic model lines for this scatterplot and decide which one is the “better” fit!

Scatterplot with “abline” and “lines” for a non-linear data with different LOWESS function values:

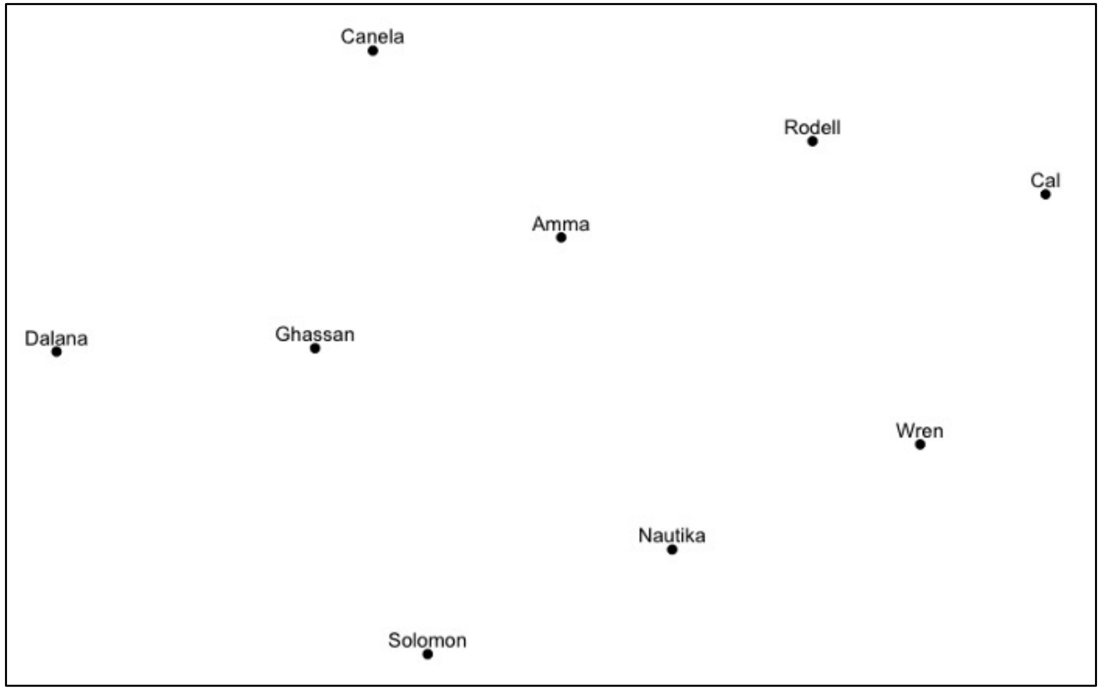
- plot(dfhp, dfdisp, main = "Scatterplot with model abline", xlab = "Horse power", ylab = "Displacement")
- abline(lm(df$disp ~ df$hp))
- lines(lowess(dfhp, dfdisp), col = "red", lwd = 3)
- lines(lowess(dfhp, dfdisp, **f=1**), col = "purple", lwd = 3)
- lines(lowess(dfhp, dfdisp, **f=0.1**), col = "blue", lwd = 3)

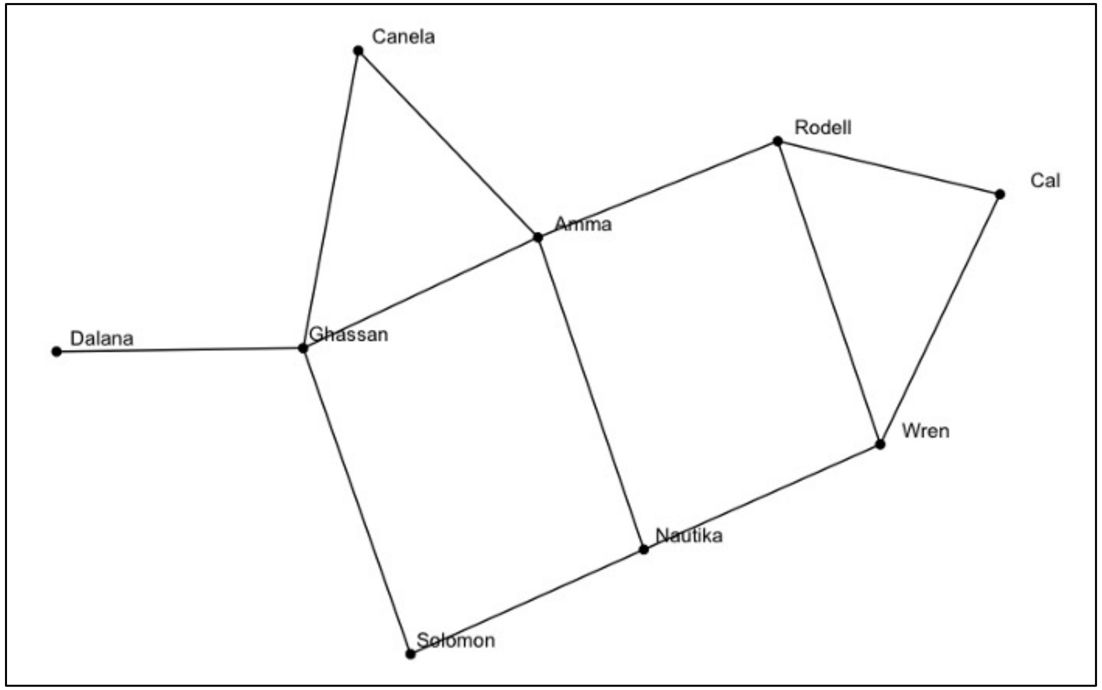


Question/Queries so far?

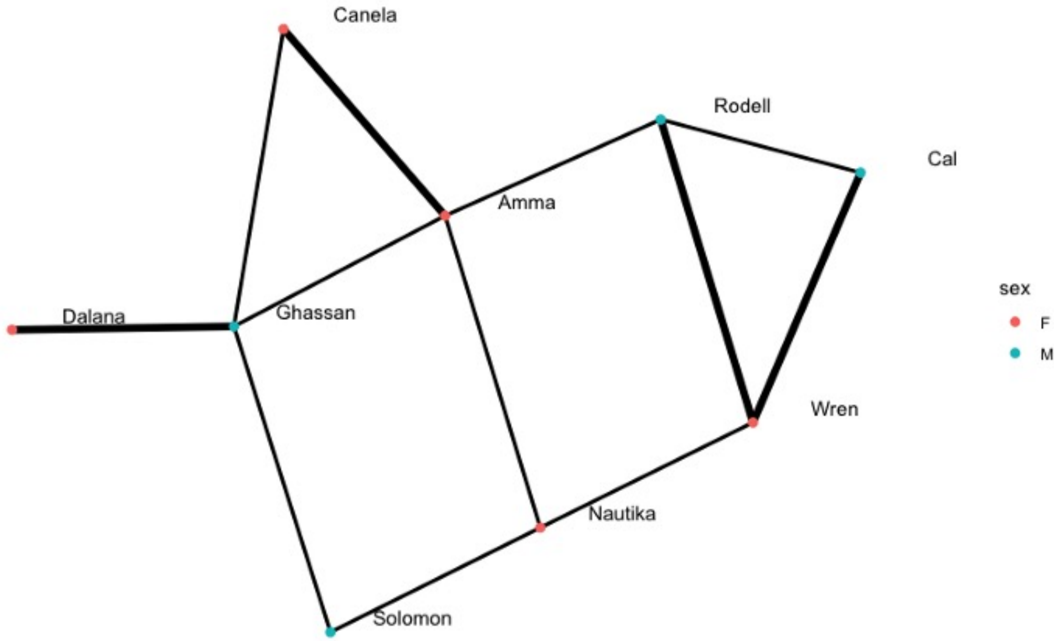
Social Network Analysis: Nodes and Links

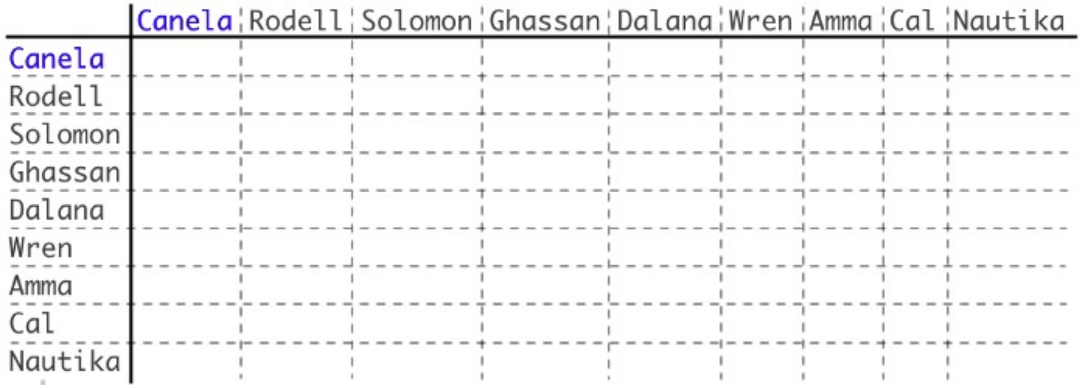
https://towardsdatascience.com/how-to-model-a-social-network-with-r-878b3a76c5a1





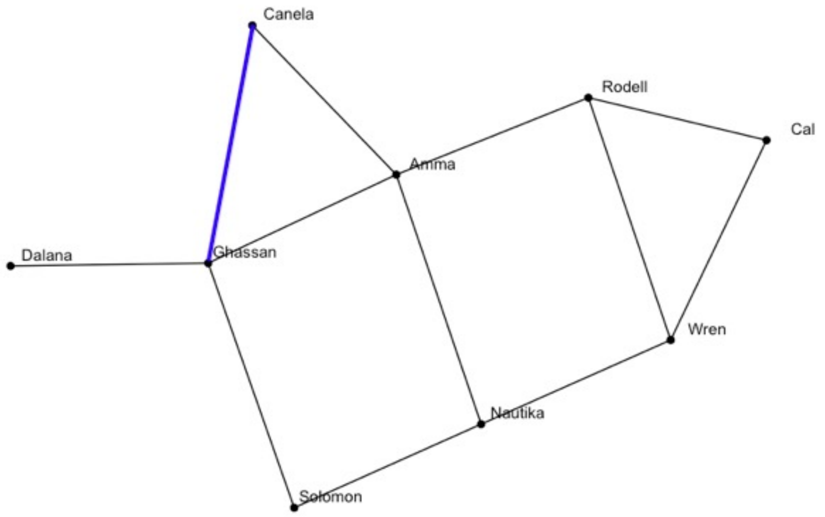
SNA: Attributes

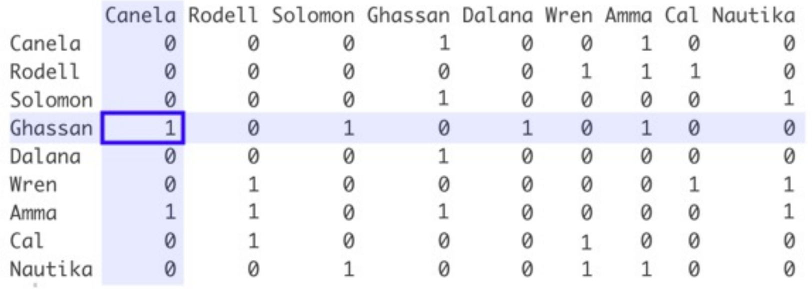




	Canela	Rodell	Solomon	Ghassan	Dalana	Wren	Amma	Cal	Nautika
Canela									
Rodell									
Solomon									
Ghassan									
Dalana									
Wren									
Amma									
Cal									
Nautika									

SNA Basics:

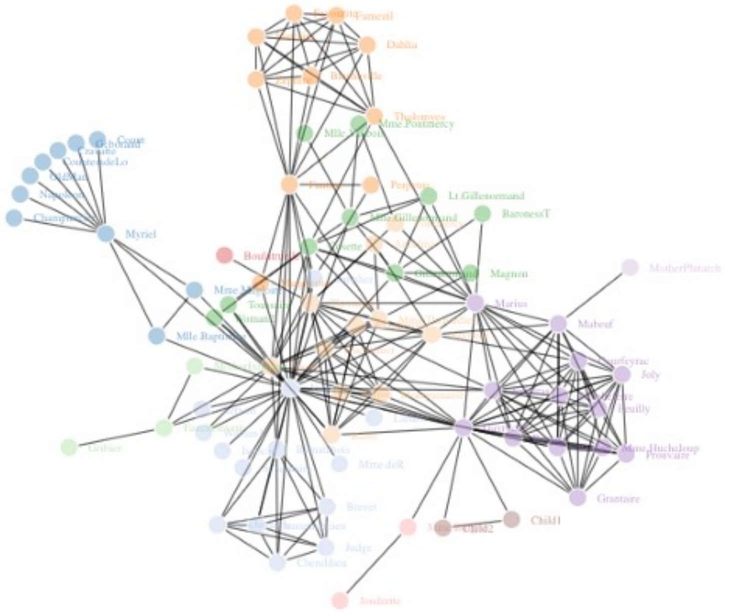


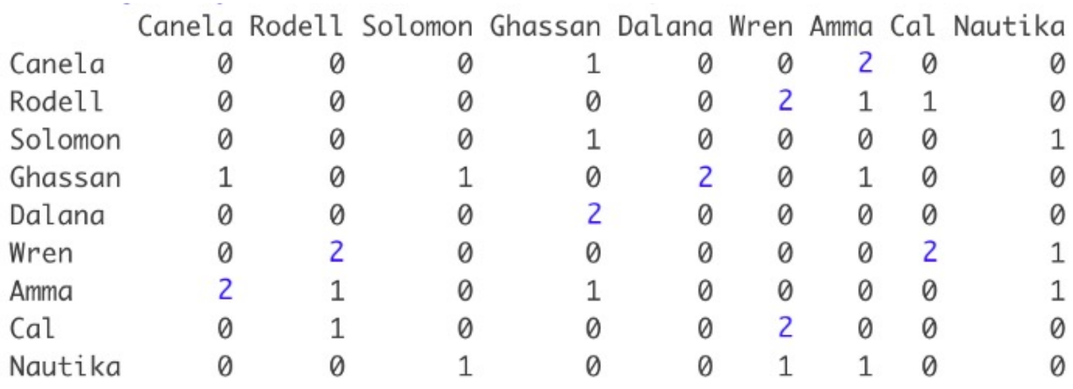


	Canela	Rodell	Solomon	Ghassan	Dalana	Wren	Amma	Cal	Nautika
Canela	0	0	0	1	0	0	1	0	0
Rodell	0	0	0	0	0	1	1	1	0
Solomon	0	0	0	1	0	0	0	0	1
Ghassan	1	0	1	0	1	0	1	0	0
Dalana	0	0	0	1	0	0	0	0	0
Wren	0	1	0	0	0	0	0	1	1
Amma	1	1	0	1	0	0	0	0	1
Cal	0	1	0	0	0	1	0	0	0
Nautika	0	0	1	0	0	1	1	0	0

- The entries in the table, show whether a link exists between two nodes:
- **1** in row “*Ghassan*”, column “*Canela*” shows that there is a link between these two.
- **0** indicates that there is no link between the two nodes.

SNA Examples:





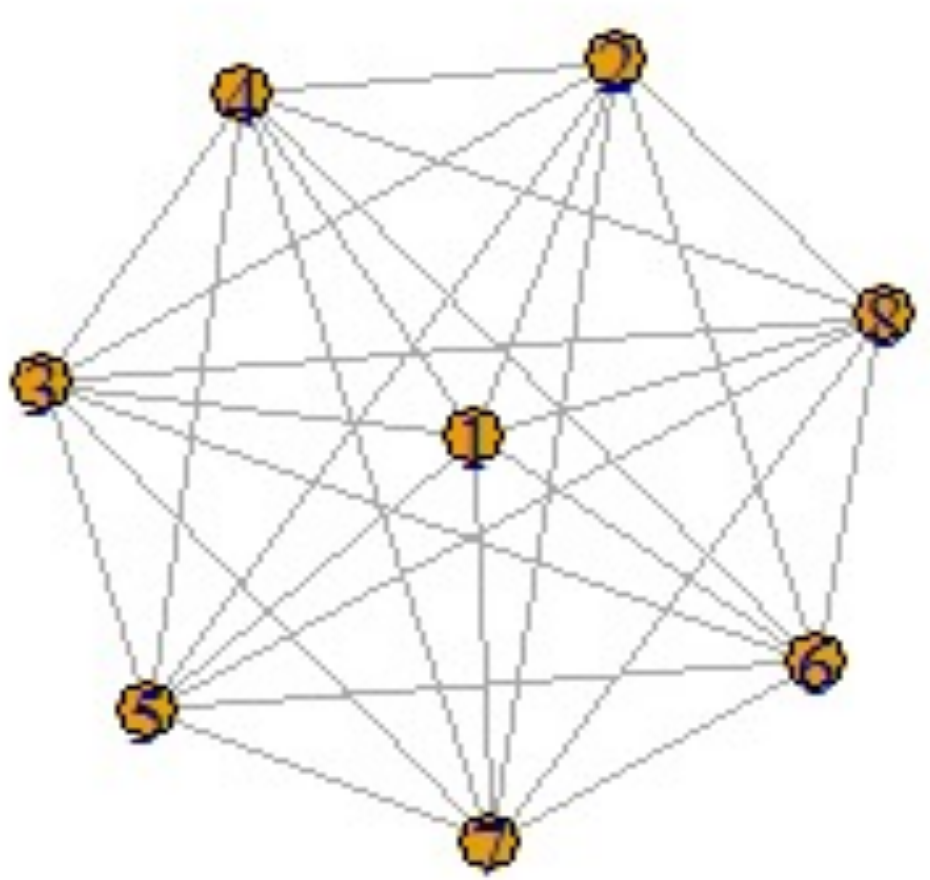
	Canela	Rodell	Solomon	Ghassan	Dalana	Wren	Amma	Cal	Nautika
Canela	0	0	0	1	0	0	2	0	0
Rodell	0	0	0	0	0	2	1	1	0
Solomon	0	0	0	1	0	0	0	0	1
Ghassan	1	0	1	0	2	0	1	0	0
Dalana	0	0	0	2	0	0	0	0	0
Wren	0	2	0	0	0	0	0	2	1
Amma	2	1	0	1	0	0	0	0	1
Cal	0	1	0	0	0	2	0	0	0
Nautika	0	0	1	0	0	1	1	0	0

Social Network Analysis in R: igraph & sna

https://www.geeksforgeeks.org/social-network-analysis-using-r-programming/

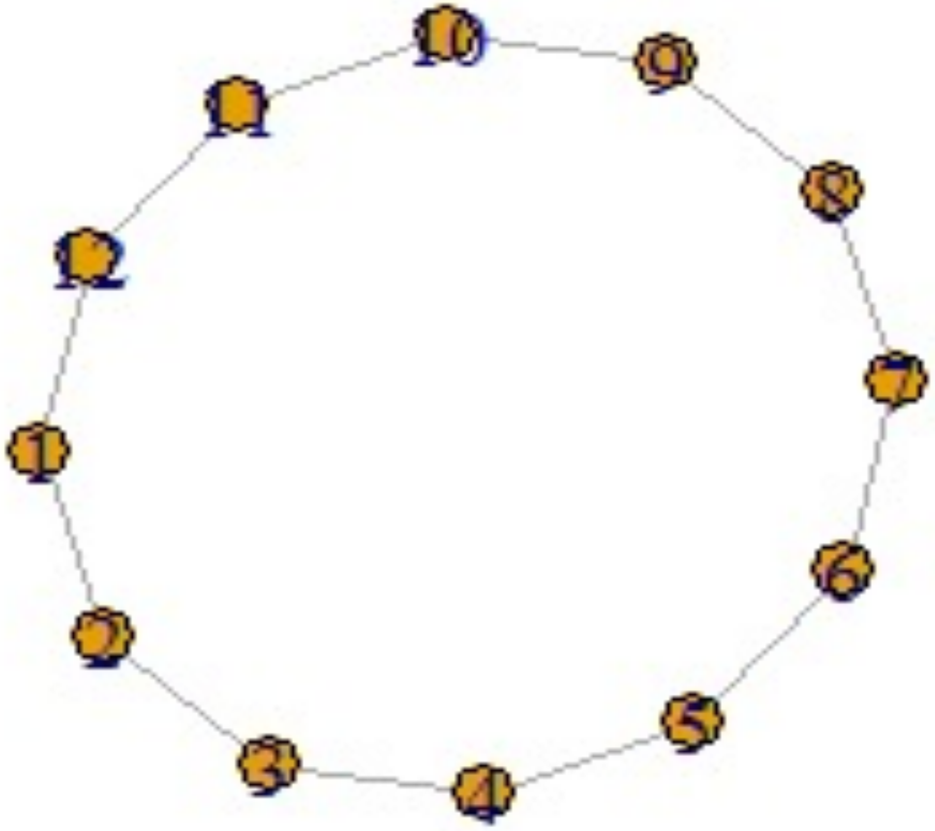
- library(igraph)
- Full_Graph <- make_full_graph(8, directed = FALSE)
- plot(Full_Graph)

- **SNA**
  - **Nodes**
  - **Links**
  - **Attributes**



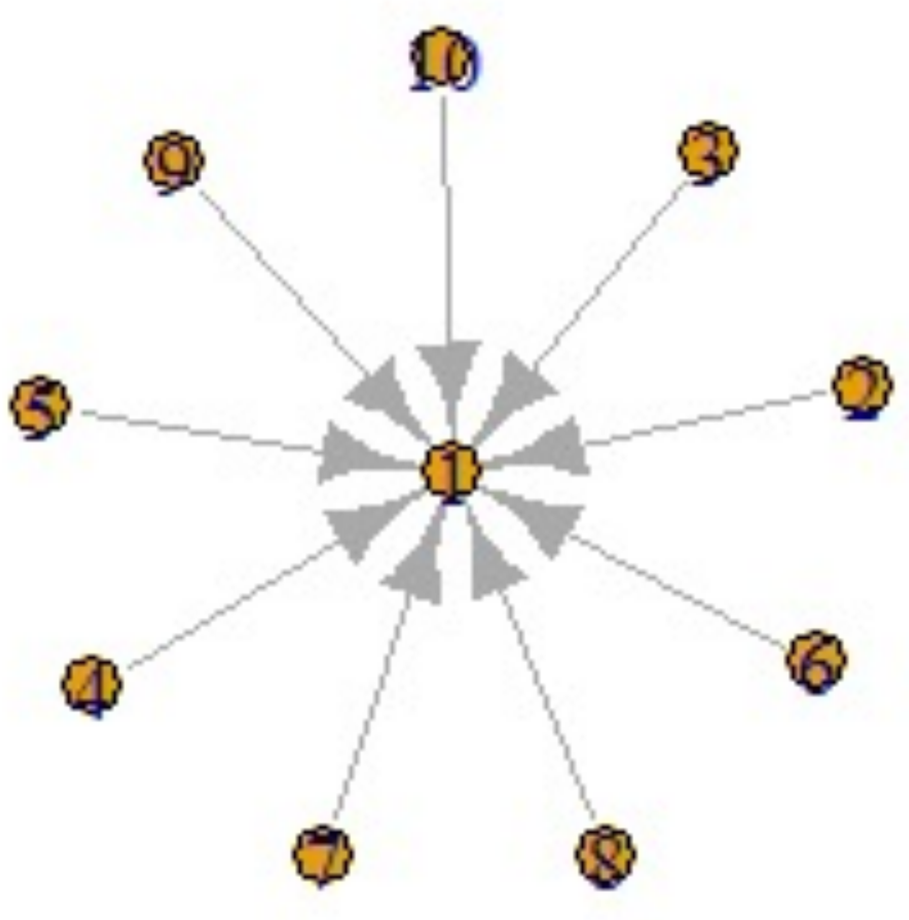
SNA: Ring Graph

- library(igraph)
- Ring_Graph <- make_ring(12, directed = FALSE, mutual = FALSE, circular = TRUE)
- plot(Ring_Graph)



SNA: Star Graph

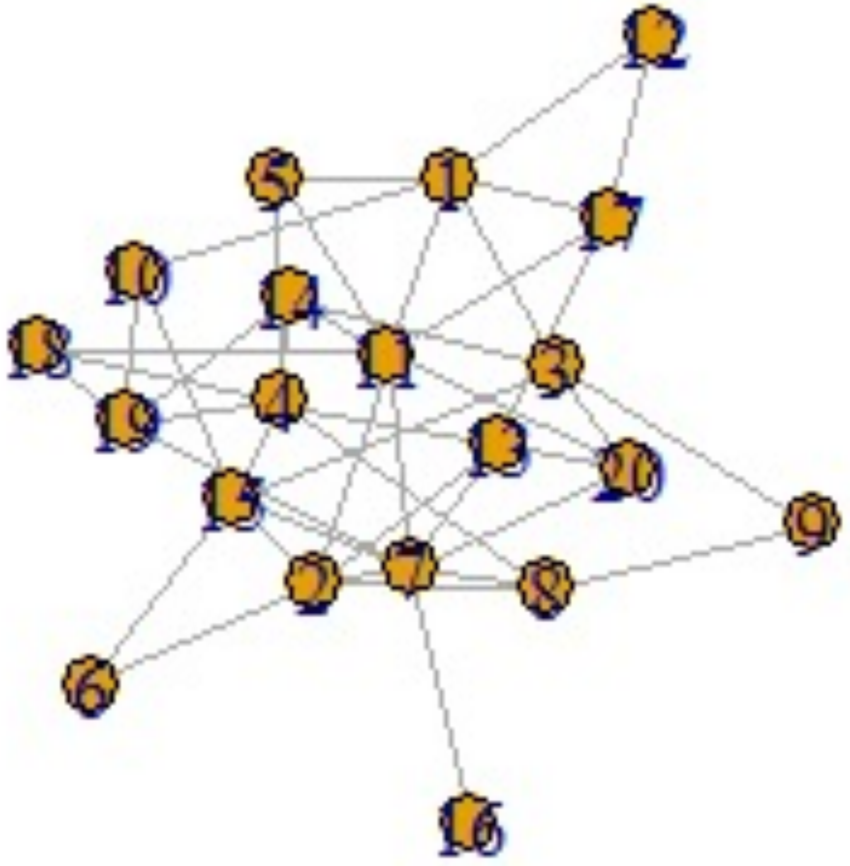
- library(igraph)
- Star_Graph <- make_star(10, center = 1)
- plot(Star_Graph)



SNA: Random Graph

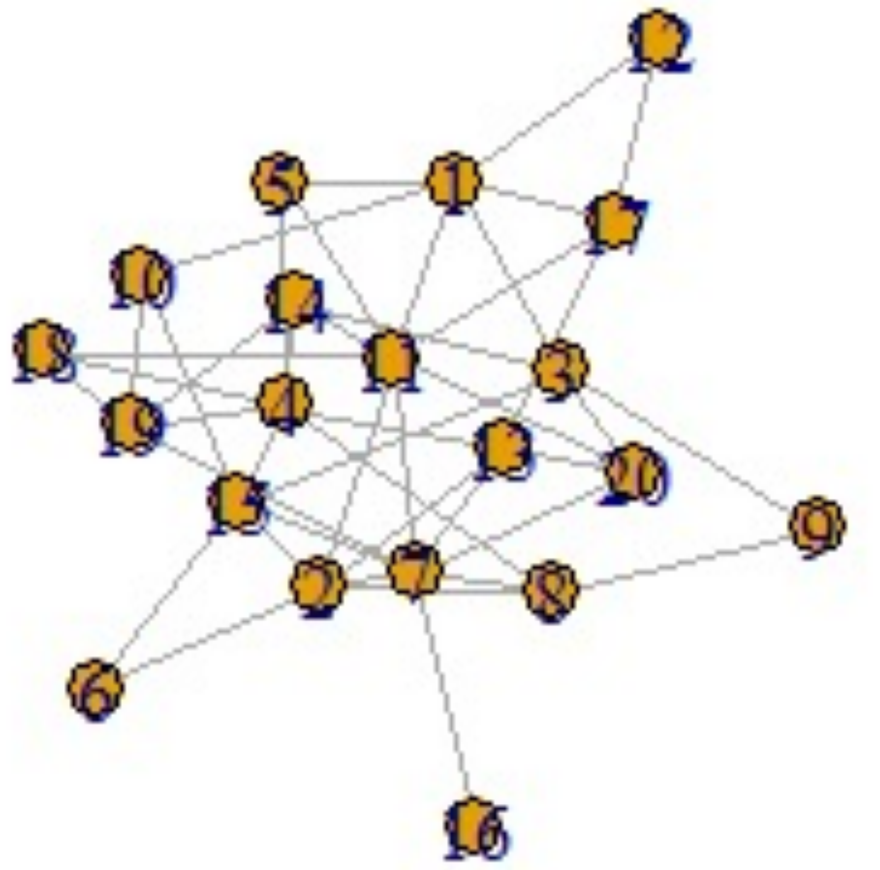
- library(igraph)
- gnp_Graph <- sample_gnp(20, 0.3, directed = FALSE, loops = FALSE)
- plot(gnp_Graph)

- **SNA graph: 20 nodes with constant probability of 0.3!**



SNA: Random Graph and its degree of each node/vertex

- library(igraph)
- gnp_Graph <- sample_gnp(20, 0.3, directed = FALSE, loops = FALSE)
- plot(gnp_Graph)
- degree(gnp_Graph)
- [1] 6 6 5 7 3 2 8 4 2 3 8 2 5 4 6 1 4 3 5 4
- The degree function is used to find out the number of vertices does each vertex is connected to.

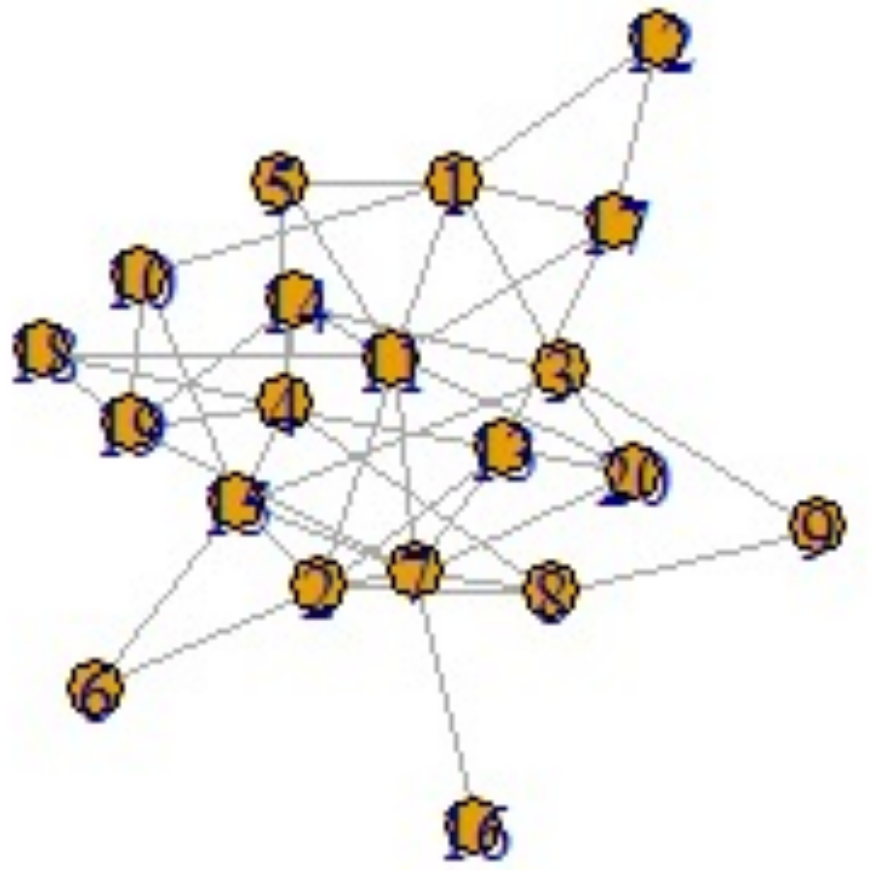


SNA: Random Graph and its betweenness

betweenness() function is defined by the number of shortest paths going through a vertex or an

edge.   the higher the betweenness score associated with a vertex, the more control over the network

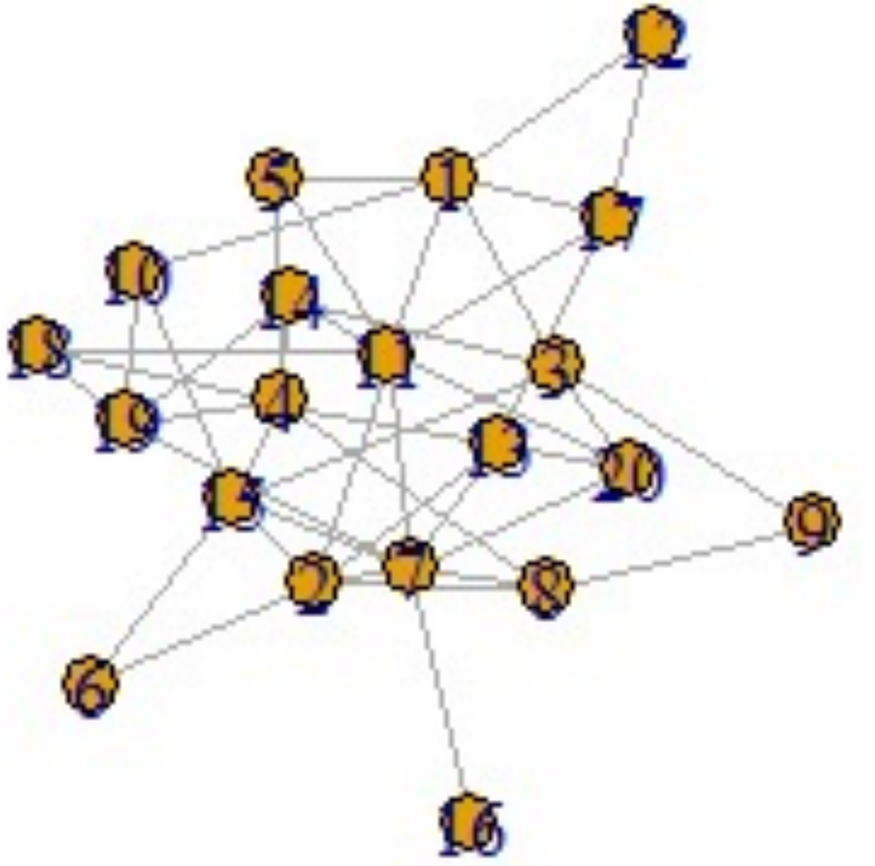
- library(igraph)
- gnp_Graph <- sample_gnp(20, 0.3, directed = FALSE, loops = FALSE)
- plot(gnp_Graph)
- betweeness(gnp_Graph)
- [1] 20.4301587 12.9523810 15.2373016 18.8817460 2.1944444 0.0000000 30.1690476
- [8] 8.6500000 1.3611111 4.4261905 30.4301587 0.0000000 9.5119048 3.7000000
- [15] 16.8833333 0.0000000 7.7055556 0.8333333 7.4333333 3.2000000



SNA: Random Graph and its density

It is the ratio of the number of edges to the total number of possible edges.

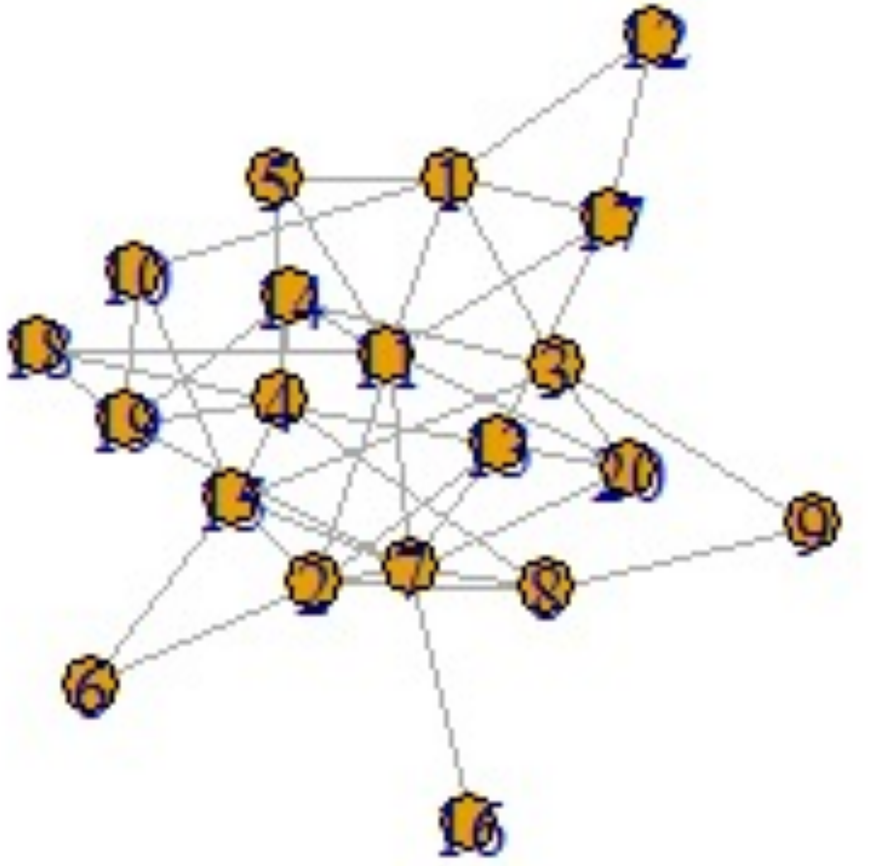
- library(igraph)
- gnp_Graph <- sample_gnp(20, 0.3, directed = FALSE, loops = FALSE)
- plot(gnp_Graph)
- samp_density <- edge_density(gnp_Graph, loops = F)
- samp_density
- 0.2315789 (Full model = 1)



SNA: Random Graph and Cliques

A clique can be defined as a group of vertices where all possible links are present.

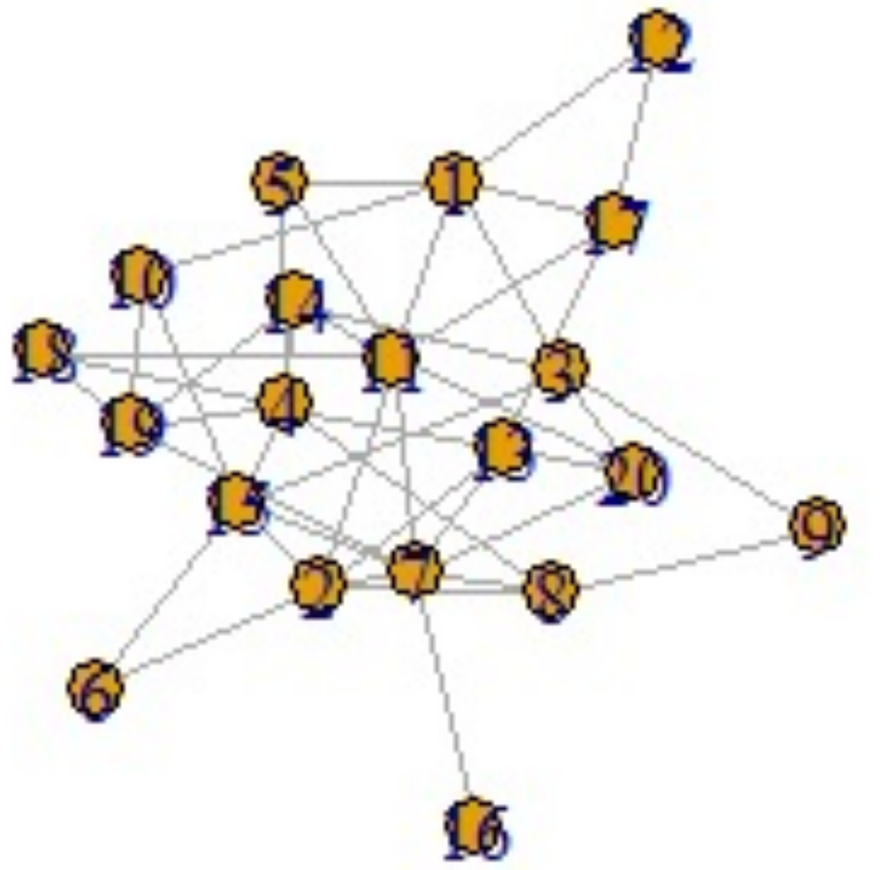
- library(igraph)
- gnp_Graph <- sample_gnp(20, 0.3, directed = FALSE, loops = FALSE)
- plot(gnp_Graph)
- samp_density <- edge_density(gnp_Graph, loops = F)
- clique_num(gnp_Graph)
- 3



SNA: Random Graph and Components

A group of connected network vertices is called a component. So it’s possible that a can have multiple components that aren’t interconnected.

- library(igraph)
- gnp_Graph <- sample_gnp(20, 0.3, directed = FALSE, loops = FALSE)
- plot(gnp_Graph)
- samp_density <- edge_density(gnp_Graph, loops = F)
- components(gnp_Graph)
- $membership
- [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
- $csize
- [1] 20
- $no
- [1] 1



Question/Queries?

- Basics of Social Network Analysis (SNA) in R: Examples https://www.youtube.com/watch?v=0xsM0MbRPGE

- SNA for Text Mining

https://www.rdatamining.com/examples/social-network-analysis

- Multidimensional Scaling with touch of SNA

https://www.rdatamining.com/examples/multidimensional-scaling-mds

Thank you!

@shitalbhandary