

# Machine Learning/Data Mining Concepts

May 2024

COMP8811 Data Analytics and Intelligence

Neda Sakhaee



# Data Mining

- **Data mining (knowledge discovery from data)**

Extraction of interesting (non-trivial, implicit, previously unknown and potentially useful) patterns or knowledge from huge amount of data

- **Alternative names**

Knowledge discovery (mining) in databases (KDD), knowledge extraction, data/pattern analysis, data archeology, data dredging, information harvesting, business intelligence, etc.



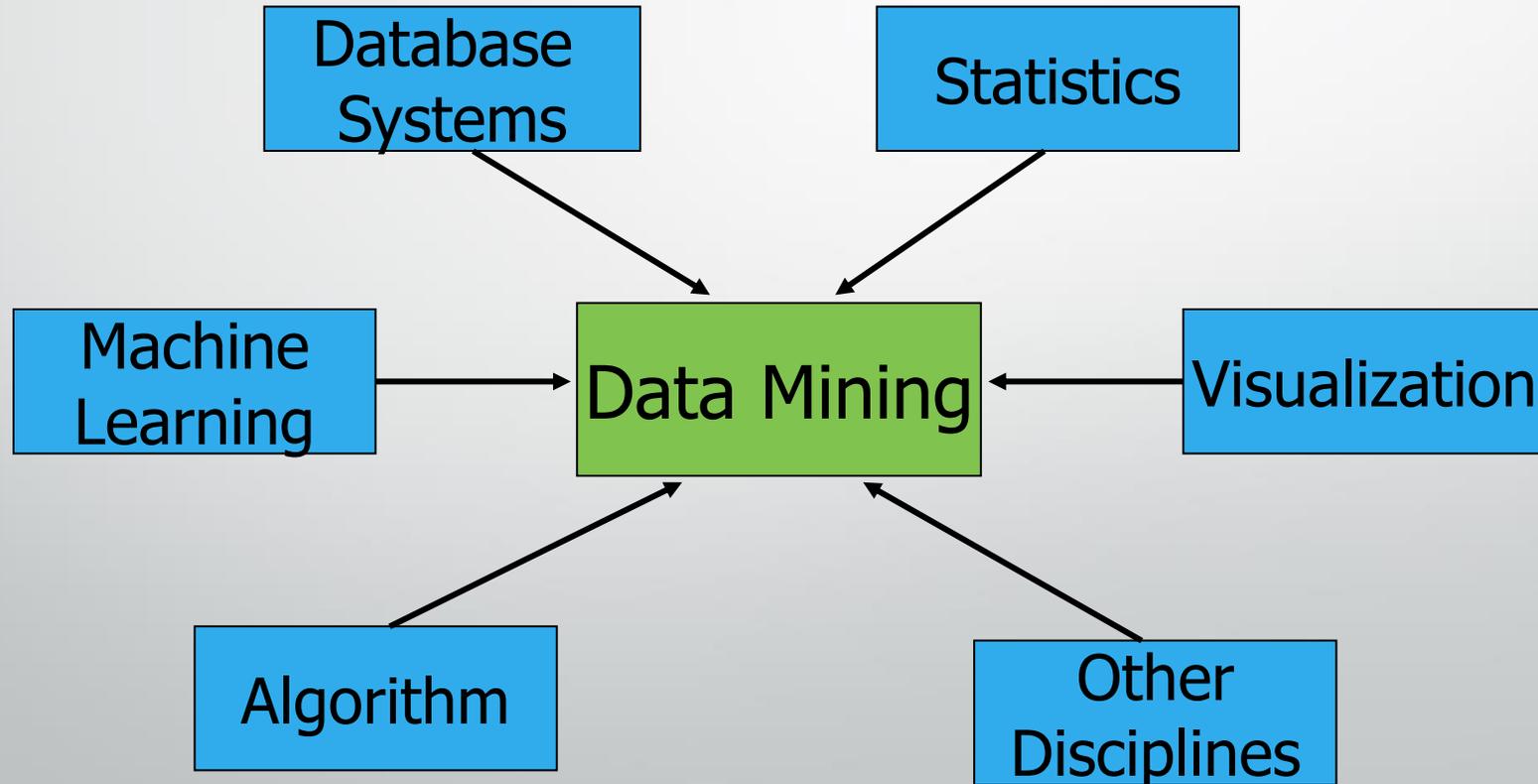
# Data Mining (History)

- *The term “data mining” has been around since at least 1983 – as a pejorative term in the statistics community*
- **Key founders / technology contributors:**
- Usama Fayyad, JPL (then Microsoft, then his own company, Digimine, now **Yahoo! Research labs**)
- Gregory Piatetsky-Shapiro (then GTE, now his own data mining consulting company, Knowledge Stream Partners)
- Rakesh Agrawal (**IBM Research**)



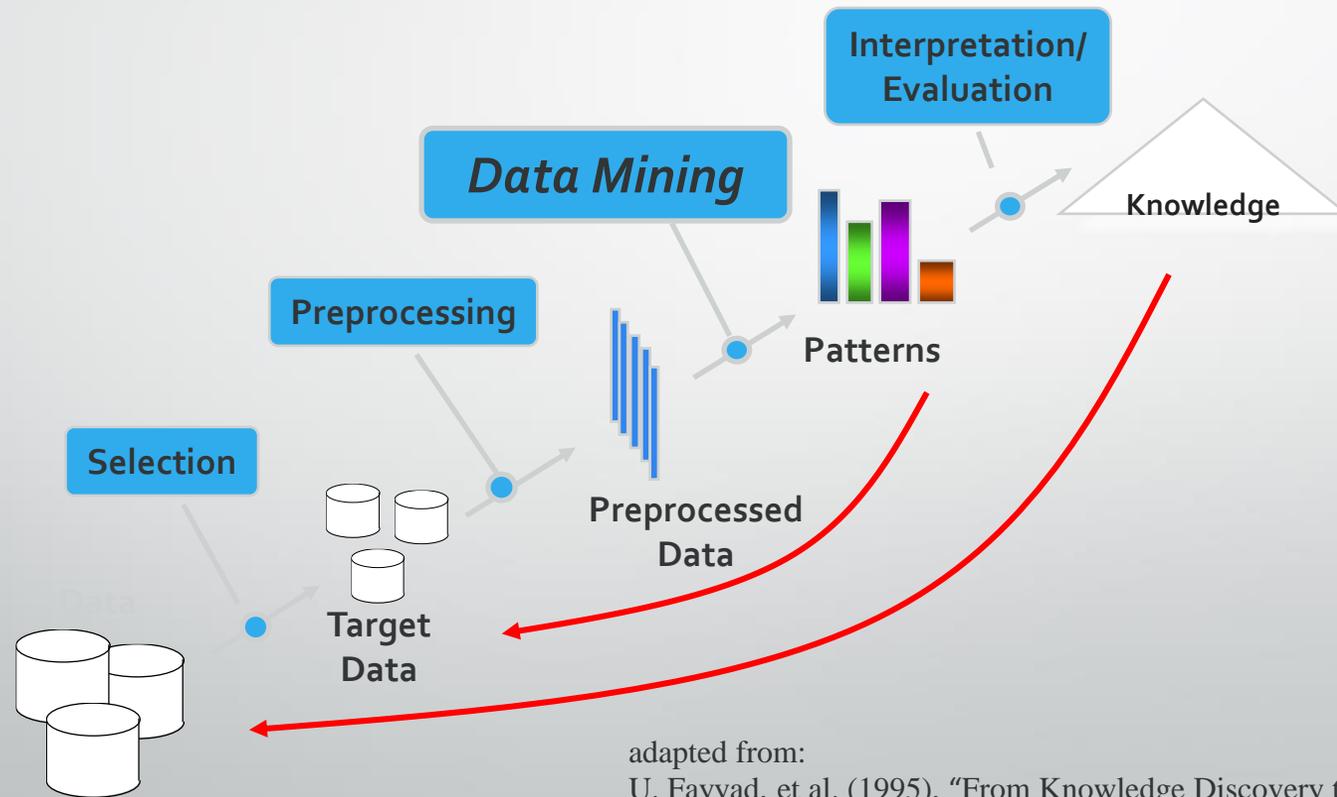
# Data Mining

Confluence of Multiple Disciplines



# Data Mining

## KDD Process



adapted from:  
U. Fayyad, et al. (1995), "From Knowledge Discovery to Data Mining: An Overview," Advances in Knowledge Discovery and Data Mining, U. Fayyad et al. (Eds.), AAAI/MIT Press



# Machine Learning Concepts

- **Classification**
- **Association rules**
- **Clustering**
- **Numeric prediction (regression)**



# Data Mining Concepts

- **Classification:**  
predicting a discrete class
- **Association:**  
detecting associations between features
- **Clustering:**  
grouping similar instances into clusters
- **Numeric prediction:**  
predicting a numeric quantity



# Possible Applications

## Data analysis and decision support

- **Market analysis and management**

Target marketing, customer relationship management (CRM), market basket analysis, cross selling, market segmentation

- **Risk analysis and management**

Forecasting, customer retention, improved underwriting, quality control, competitive analysis

- **Fraud detection and detection of unusual patterns (outliers)**

- **Other Applications**

- Text mining (news group, email, documents) and Web mining
- DNA and bio-data analysis



# Market Analysis and Management

- **Where does the data come from?**

Credit card transactions, loyalty cards, discount coupons, customer complaint calls, plus (public) lifestyle studies

- **Target marketing**

Find clusters of “model” customers who share the same characteristics: interest, income level, spending habits, etc.  
Determine customer purchasing patterns over time

- **Cross-market analysis**

Associations/co-relations between product sales, & prediction based on such association



# Corporate Analysis & Risk Management

- **Finance planning and asset evaluation**
  - ✓ Cash flow analysis and prediction
  - ✓ Claim analysis and evaluate assets
- **Competition**
  - ✓ Monitor competitors and market directions
  - ✓ Group customer into classes(or cluster) based pricing procedure
  - ✓ Set pricing strategy



# What is Machine Learning?

Arthur Samuel described it as: "the field of study that gives computers the ability to learn without being explicitly programmed." This is an older, informal definition.

Tom Mitchell provides a more modern definition: "A computer program is said to learn from experience  $E$  with respect to some class of tasks  $T$  and performance measure  $P$ , if its performance at tasks in  $T$ , as measured by  $P$ , improves with experience  $E$ ."



Example: playing checkers.

- E = the experience of playing many games of checkers
- T = the task of playing checkers.
- P = the probability that the program will win the next game.

In general, any machine learning problem can be assigned to one of two broad classifications:

- Supervised learning and
- Unsupervised learning.



# Supervised Learning

In supervised learning, we are given a data set and already know what our correct output should look like, having the idea that there is a relationship between the input and the output.

## Example 1:

- Given data about the size of houses on the real estate market, try to predict their price. Price as a function of size is a continuous output, so this is a regression problem.
- We could turn this example into a classification problem by instead making our output about whether the house "sells for more or less than the asking price." Here we are classifying the houses based on price into two discrete categories.

## Example 2:

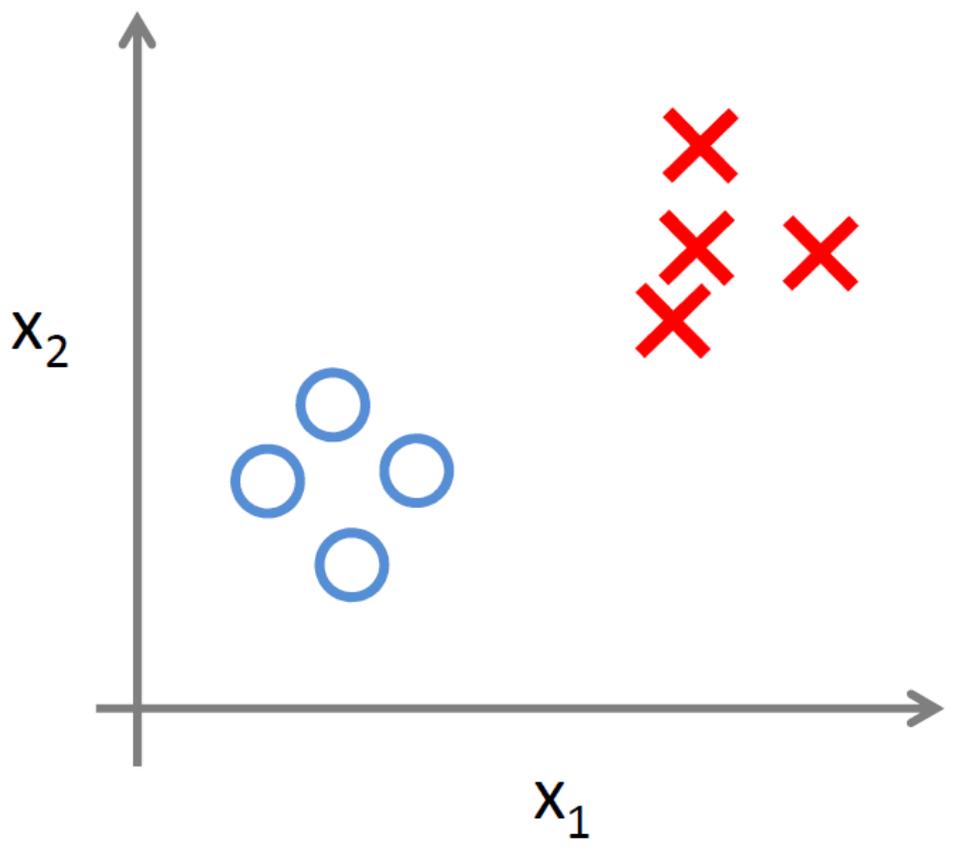
- (a) Regression - Given a picture of a person, we have to predict their age on the basis of the given picture
- (b) Classification - Given a patient with a tumor, we have to predict whether the tumor is malignant or benign.



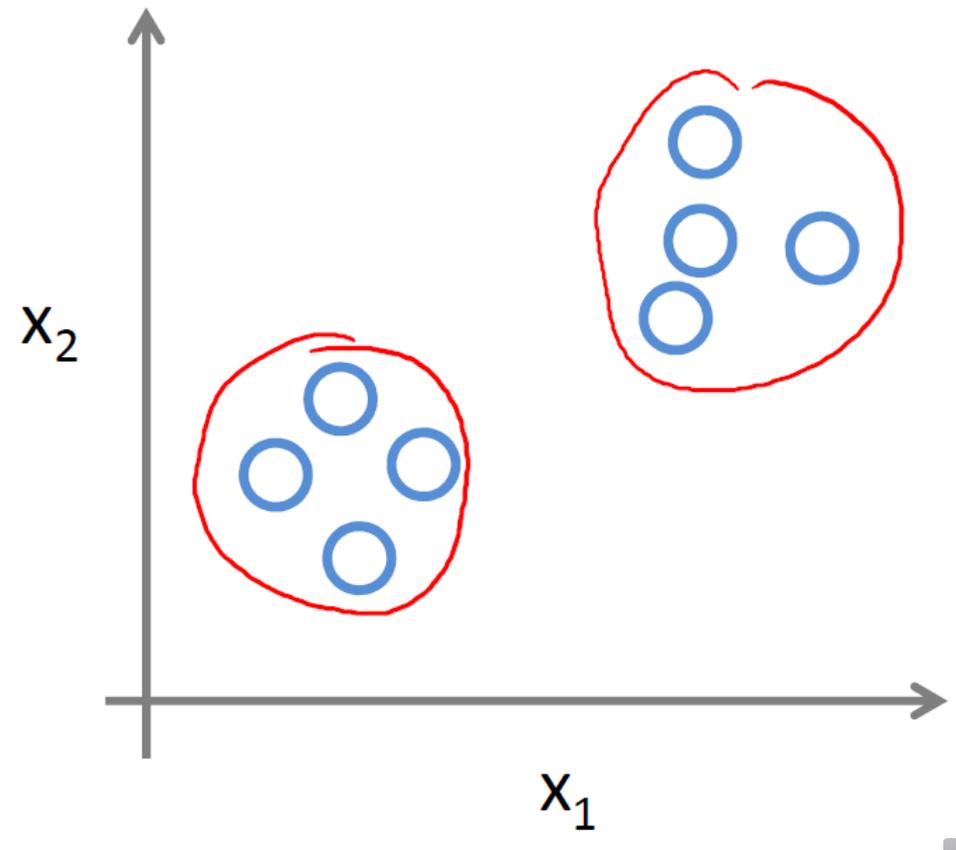
# Unsupervised Learning

- Unsupervised learning allows us to approach problems with little or no idea what our results should look like. We can derive structure from data where we don't necessarily know the effect of the variables.
- We can derive this structure by clustering the data based on relationships among the variables in the data.
- **Example:**
- Clustering: Take a collection of 1,000,000 different genes, and find a way to automatically group these genes into groups that are somehow similar or related by different variables, such as lifespan, location, roles, and so on.
- Non-clustering: The "Cocktail Party Algorithm", allows you to find structure in a chaotic environment. (i.e. identifying individual voices and music from a mesh of sounds at a [cocktail party](#)).

# Supervised Learning



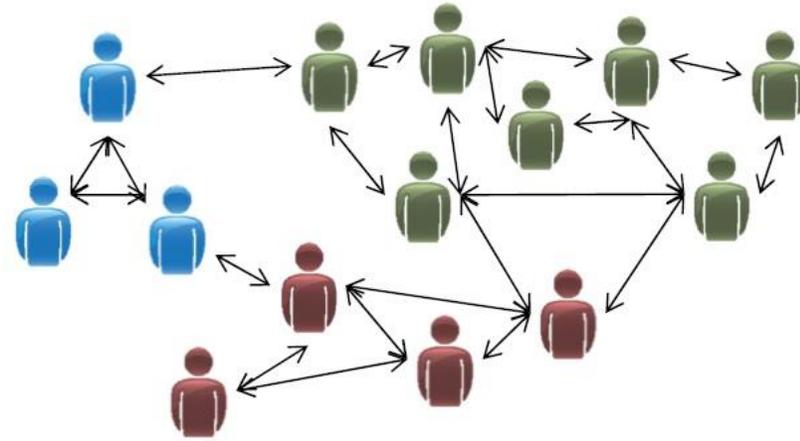
# Unsupervised Learning



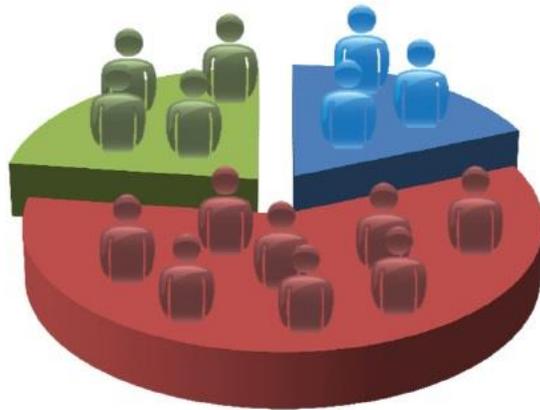
# Clustering



Organize computing clusters



Social network analysis



Market segmentation



Astronomical data analysis



# Classification (Naive Bayes method)

- What is Conditional Probability ?
- What is Bayes Theorem?
- What is NAIVE BAYES CLASSIFIER?
- Types of Naive Bayes Algorithm.
- How to use R for NB classification



# Conditional Probability

1. In probability theory, conditional probability is a measure of the probability of an event given that another event has already occurred.
2. If the event of interest is  $A$  and the event  $B$  is assumed to have occurred, "the conditional probability of  $A$  given  $B$ ", or "the probability of  $A$  under the condition  $B$ ", is usually written as  $P(A|B)$ , or sometimes  $P_B(A)$ .



# Example

## Chances of cough

The probability that any given person has a cough on any given day maybe only 5%. But if we know or assume that the person has a cold, then they are much more likely to be coughing. The conditional probability of coughing given that person have a cold might be a much higher i.e. 75%.



# Marbles in a Bag

2 blue and 3 red marbles are in a bag.

What are the chances of getting a blue marble?

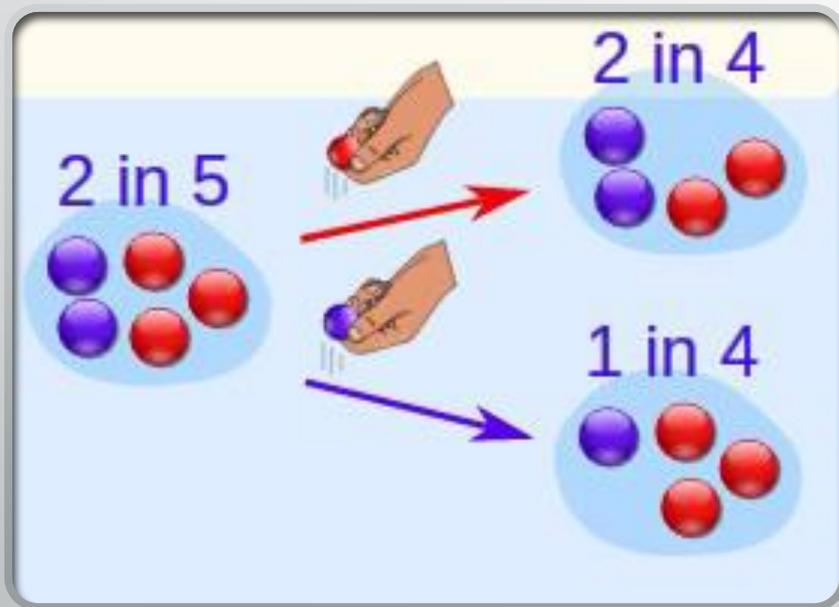
???

**Answer:**

**The chance is 2 in 5**



# situation may change!



After taking one out of these chances,

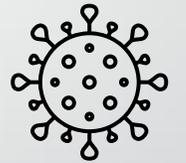
So the next time:

- if we got a red marble before, then the chance of a blue marble next is 2 in 4
- if we got a blue marble before, then the chance of a blue marble next is 1 in 4



# Likewise:

- Drawing a second ace from a deck given we got the first ace
- Finding the probability of having a disease given you were tested positive
- Finding the probability of liking Harry Potter given we know the person likes fiction.



# Bayes Theorem

- In probability theory and statistics, Bayes' theorem (alternatively Bayes' law or Bayes' rule) describes the probability of an event, based on prior knowledge of conditions that might be related to the event.
- For example, if cancer is related to age, then, using Bayes' theorem, a person's age can be used to more accurately to assess the probability that they have cancer, compared to the assessment of the probability of cancer made without knowledge of the person's age.



# The Formula for Bayes' theorem

probability a hypothesis is true  
given the evidence

probability a hypothesis is true  
(before any evidence is present)

$$P(H/E) = \frac{P(H) P(E/H)}{P(E)}$$

probability of seeing the evidence  
if the hypothesis is true

probability of observing the evidence



# Example

Suppose there are three bowls B1, B2, B3 and bowl B1 has 2 red and 4 blue coins; Bowl B2 has 1 red and 2 blue coins; bowl B3 contains 5 red and 4 blue coins. Suppose the probabilities for selecting the bowls is not the same but are:

- $P(B1) = 1/3$
- $P(B2) = 1/6$
- $P(B3) = 1/2$

Now, let us compute, assuming that a red coin was drawn what will be the probability that it came from bowl B1.



In mathematics terms, we need to find out  $P(B_1|R) = ???$

And according to Bayes' theorem

$$P(B_1|R) = P(R|B_1) * P(B_1) / P(R)$$

For that first, we need to calculate some probabilities which are:-

- Probability to select a red coin i.e  $P(R)$
- Probability to select the bowl 1 ( $B_1$ ) i.e  $P(B_1)$  which is already given  $1/3$
- Probability to select a red coin from  $B_1$  i.e  $P(R|B_1)$



$P(R)$

= (total number of red coins) / (total number of coins)

=  $(2+5+1)/(6+9+3) = 8/18$

=  $4/9$

So  $P(R) = 4/9$



## **P(R | B1)**

The probability of selecting a red coin given that it will be drawn from B1 is **2/6**

**P(B1)** was given i.e 1/3.

By putting all the values in formula:

$$P(B1 | R) = (2/6 * 1/3) / 4/9 = 2/8 = 0.25$$

**So we can say that if a red coin was drawn that it will be 25% chances that it was drawn from bowl 1 i.e B1.**



# NAIVE BAYES CLASSIFIER

- Naive Bayes is a kind of classifier which uses the Bayes Theorem.
- It predicts membership probabilities for each class such as the probability that given record or data point belongs to a particular class.
- The class with the highest probability is considered as the most likely class. This is also known as **Maximum A Posteriori (MAP)**.



**MAP(H)**

$$= \max( P(H|E) )$$

$$= \max( (P(E|H)*P(H))/P(E) )$$

$$= \max(P(E|H)*P(H))$$

P(E) is evidence probability, and it is used to normalize the result. It remains same so, removing it won't affect.



# Assumption

**Naive Bayes classifier assumes that all the features are unrelated to each other. Presence or absence of a feature does not influence the presence or absence of any other feature.**

*"A fruit may be considered to be an apple if it is red, round, and about 4" in diameter. Even if these features depend on each other or upon the existence of the other features, a naive Bayes classifier considers all of these properties to independently contribute to the probability that this fruit is an apple."*



In real datasets, we test a hypothesis given multiple evidence(feature). So, calculations become complicated. To simplify the work, the feature independence approach is used to 'uncouple' multiple evidence and treat each as an independent one.

**$P(H | \text{Multiple Evidences}) =$**

**$P(E_1 | H) * P(E_2 | H) \dots * P(E_n | H) * P(H) / P(\text{Multiple Evidences})$**



# Example

For understanding a theoretical concept, the best procedure is to try it on an example.

Let's consider a training dataset with 1500 records and 3 classes. We presume that there are no missing values in our data. We have 3 classes associated with Animal Types:

- Parrot,
- Dog,
- Fish.



The Predictor features set consists of 4 features as

- Swim
- Wings
- Green Colour
- Dangerous Teeth.

Swim, Wings, Green Colour, Dangerous Teeth. All the features are categorical variables with either of the 2 values: **T(True)** or **F( False)**.



The table shows a frequency table of our data. In our training data:

Swim	Wings	Green Colour	Dangerous Teeth	Animal Type
50/500	500/500	400/500	0	Parrot
450/500	0	0	500/500	Dog
500/500	0	100/500	50/500	Fish



- Parrots have 50(10%) value for Swim, i.e., 10% parrot can swim according to our data, 500 out of 500(100%) parrots have wings, 400 out of 500(80%) parrots are Green and 0(0%) parrots have Dangerous Teeth.
- Classes with Animal type Dogs shows that 450 out of 500(90%) can swim, 0(0%) dogs have wings, 0(0%) dogs are of Green colour and 500 out of 500(100%) dogs have Dangerous Teeth.
- Classes with Animal type Fishes shows that 500 out of 500(100%) can swim, 0(0%) fishes have wings, 100(20%) fishes are of Green colour and 50 out of 500(10%) Fishes have Dangerous Teeth.



Now, it's time to work on predict classes using the Naive Bayes model. We have taken 2 records that have values in their feature set, but the target variable needs to be predicted.

	Swim	Wings	Green	Teeth
1	True	False	True	False
2	True	False	True	True

We have to predict animal type using the feature values. We have to predict whether the animal is a Dog, a Parrot or a Fish

$$P(H | \text{Multiple Evidences}) = P(E1 | H) * P(E2 | H) \dots * P(En | H) * P(H) /$$

**P(Multiple Evidences)**

Let's consider the first record. The Evidence here is Swim & Green. The Hypothesis can be an animal type to be Dog, Parrot, Fish.



**For Hypothesis testing for the animal to be a Dog:**

$$P(\text{Dog} \mid \text{Swim, Green}) = P(\text{Swim} \mid \text{Dog}) * P(\text{Green} \mid \text{Dog}) * P(\text{Dog}) / P(\text{Swim, Green})$$

$$= 0.9 * 0 * 0.333 / P(\text{Swim, Green})$$

$$= 0$$

**For Hypothesis testing for the animal to be a Parrot:**

$$P(\text{Parrot} \mid \text{Swim, Green}) = P(\text{Swim} \mid \text{Parrot}) * P(\text{Green} \mid \text{Parrot}) * P(\text{Parrot}) / P(\text{Swim, Green})$$

$$= 0.1 * 0.80 * 0.333 / P(\text{Swim, Green})$$

$$= 0.0264 / P(\text{Swim, Green})$$

- **For Hypothesis testing for the animal to be a Fish:**

- $P(\text{Fish} \mid \text{Swim, Green}) = P(\text{Swim} \mid \text{Fish}) * P(\text{Green} \mid \text{Fish}) * P(\text{Fish}) / P(\text{Swim, Green})$
- Green)
- $= 1 * 0.2 * 0.333 / P(\text{Swim, Green})$
- $= 0.0666 / P(\text{Swim, Green})$

Swim	Wings	Green Colour	Dangerous Teeth	Animal Type
50/500	500/500	400/500	0	Parrot
450/500	0	0	500/500	Dog
500/500	0	100/500	50/500	Fish

- The denominator of all the above calculations is same i.e,  $P(\text{Swim, Green})$ .
- The value of  $P(\text{Fish} | \text{Swim, Green})$  is greater than  $P(\text{Parrot} | \text{Swim, Green})$ .
- **Using Naive Bayes, we can predict that the class of this record is Fish.**



**Let's consider the second record.**

The Evidence here is Swim, Green & Teeth. The Hypothesis can be an animal type to be Dog, Parrot, Fish.

**For Hypothesis testing for the animal to be a Dog:**

$$\begin{aligned}
 P(\text{Dog} \mid \text{Swim, Green, Teeth}) &= P(\text{Swim} \mid \text{Dog}) * P(\text{Green} \mid \text{Dog}) * P(\text{Teeth} \mid \text{Dog}) * \\
 &P(\text{Dog}) / P(\text{Swim, Green, Teeth}) \\
 &= 0.9 * 0 * 1 * 0.333 / P(\text{Swim, Green, Teeth}) = 0
 \end{aligned}$$



**For Hypothesis testing for the animal to be a Parrot:**

$$P(\text{Parrot} | \text{Swim, Green, Teeth}) = \frac{P(\text{Swim} | \text{Parrot}) * P(\text{Green} | \text{Parrot}) * P(\text{Teeth} | \text{Parrot}) * P(\text{Parrot})}{P(\text{Swim, Green, Teeth})}$$

$$= \frac{0.1 * 0.80 * 0 * 0.333}{P(\text{Swim, Green, Teeth})}$$

$$= 0$$

**For Hypothesis testing for the animal to be a Fish:**

$$P(\text{Fish} | \text{Swim, Green, Teeth}) = \frac{P(\text{Swim} | \text{Fish}) * P(\text{Green} | \text{Fish}) * P(\text{Teeth} | \text{Fish}) * P(\text{Fish})}{P(\text{Swim, Green, Teeth})}$$

$$= \frac{1 * 0.2 * 0.1 * 0.333}{P(\text{Swim, Green, Teeth})}$$

$$= \frac{0.00666}{P(\text{Swim, Green, Teeth})}$$

The value of  $P(\text{Fish} | \text{Swim, Green, Teeth})$  is the only positive value greater than 0. Using Naive Bayes, we can predict that the class of this record is **Fish**.



# Class Activity in R



```
# Libraries
library(naivebayes)
library(dplyr)
library(ggplot2)
library(psych)
```

Set Working Directory

```
#Read data file
data <- read.csv("binary.csv", header = T)
```

```
#contingency table
xtabs(~admit + rank, data = data)
```

admit	gre	gpa	rank
0	380	3.61	3
1	660	3.67	3
1	800	4.00	1
1	640	3.19	4
0	520	2.93	4
1	760	3.00	2

	rank			
admit	1	2	3	4
0	28	97	93	55
1	33	54	28	12

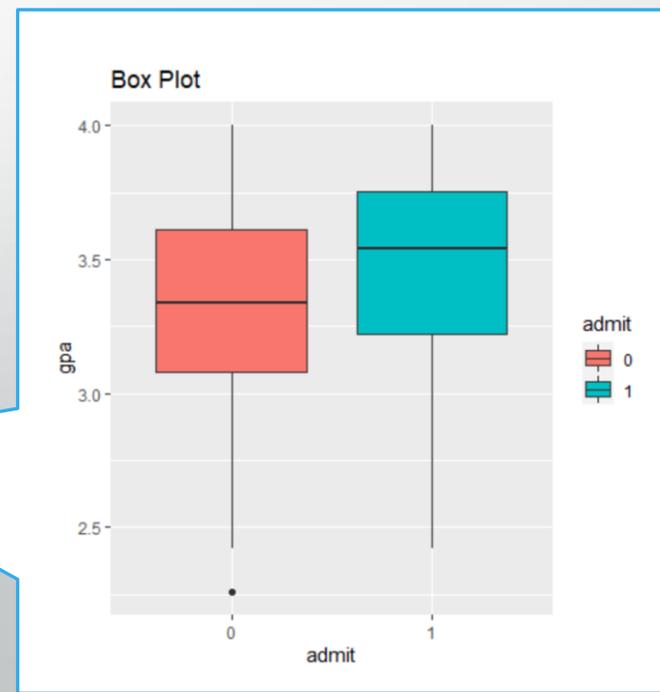
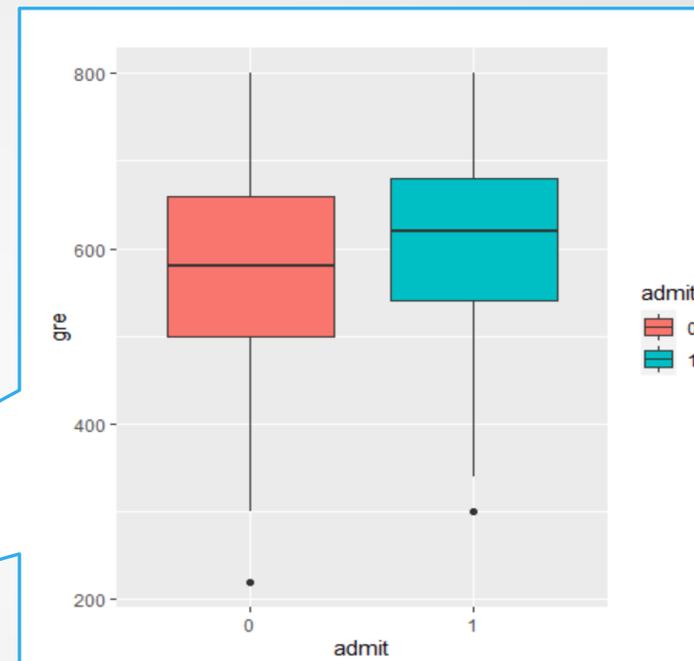


# Class Activity in R

```
#Rank & admit are categorical variables  
data$rank <- as.factor(data$rank)  
data$admit <- as.factor(data$admit)
```

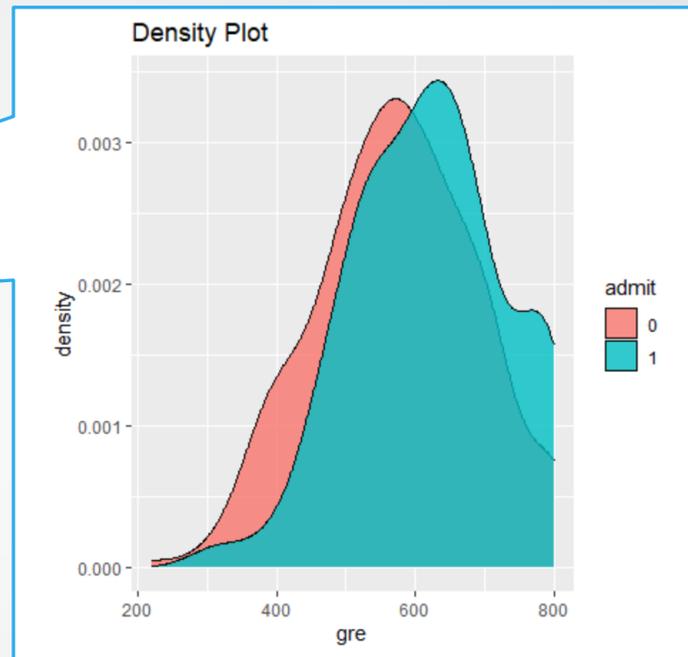
```
# Visualization  
pairs.panels(data[-1])  
data %>%  
  group_by(admit) %>%  
  ggplot(aes(x=admit, y=gre, fill=admit)) +  
  geom_boxplot()
```

```
data %>%  
  ggplot(aes(x=admit, y=gpa, fill=admit)) +  
  geom_boxplot() +  
  ggtitle('Box Plot')
```

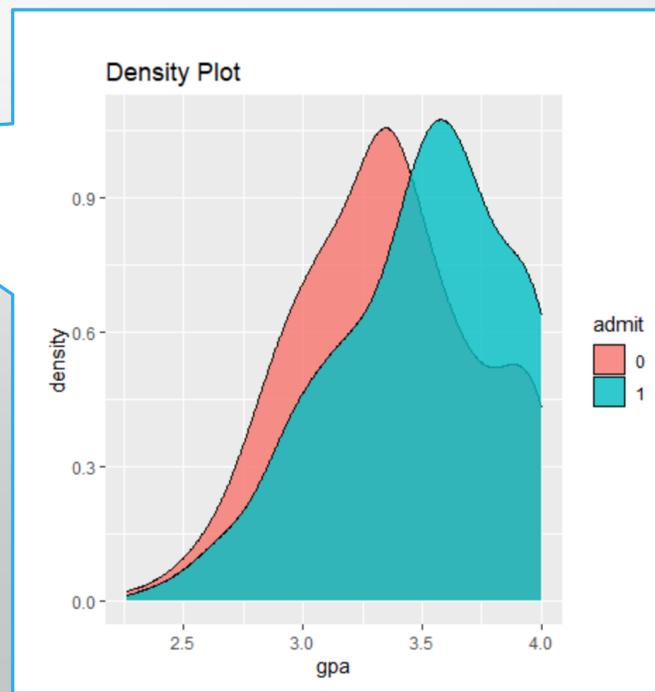


# Class Activity in R

```
data %>%  
  ggplot(aes(x=gre, fill=admit)) +  
  geom_density(alpha=0.8, color='black') +  
  ggtitle('Density Plot')
```



```
data %>%  
  ggplot(aes(x=gpa, fill=admit)) +  
  geom_density(alpha=0.8, color='black') +  
  ggtitle('Density Plot')
```



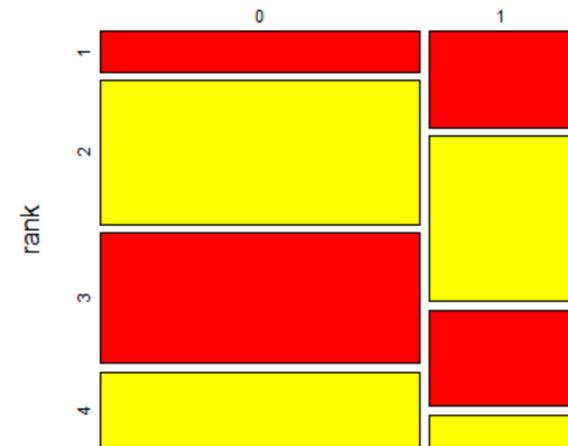
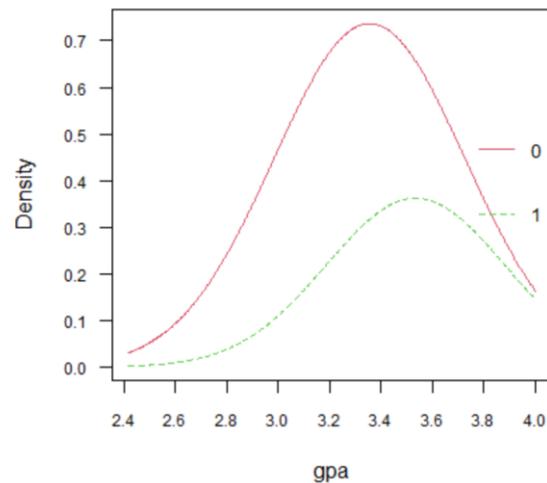
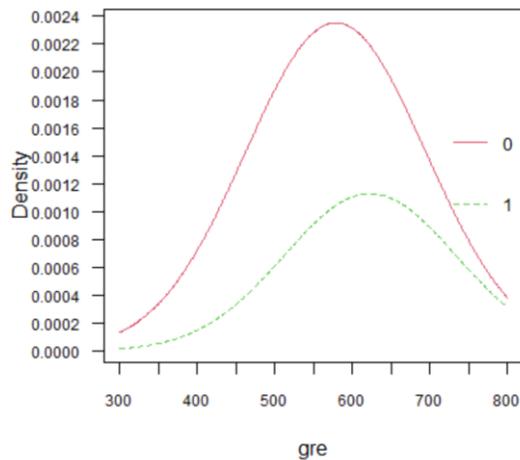
# Class Activity in R



```
# Split data into Training (80%) and Testing (20%) datasets
set.seed(1234)
ind <- sample(2, nrow(data), replace=TRUE, prob=c(0.8, .2))
train <- data[ind==1,]
test <- data[ind==2,]
```

▶ data	400 obs. of 4 variables
▶ test	75 obs. of 4 variables
▶ train	325 obs. of 4 variables
Values	
ind	int [1:400] 1 1 1 1 2 1 :

```
# Naive Bayes
model <- naive_bayes(admit ~ ., data = train)
model
plot(model)
```



# Class Activity in R



```
# Predict
p <- predict(model, train, type= 'prob')
head(cbind(p, train))
```

```
      0      1 admit gre  gpa rank
1 0.8449088 0.1550912    0 380 3.61    3
2 0.6214983 0.3785017    1 660 3.67    3
3 0.2082304 0.7917696    1 800 4.00    1
4 0.8501030 0.1498970    1 640 3.19    4
6 0.6917580 0.3082420    1 760 3.00    2
7 0.6720365 0.3279635    1 560 2.98    1
```

```
# Misclassification error - train data
p1 <- predict(model, train)
(tab1 <- table(p1, train$admit))
1 - sum(diag(tab1))/ sum(tab1)
```

```
p1    0    1
  0 196   69
  1   27   33
> 1 - sum(diag(tab1))/ sum(tab1)
[1] 0.2953846
> |
```

```
# Misclassification error - test data
p2 <- predict(model, test)
(tab2 <- table(p2, test$admit))
1 - sum(diag(tab2))/ sum(tab2)
```

```
p2    0    1
  0 47 21
  1  3  4
> 1 - sum(diag(tab2))/ sum(tab2)
[1] 0.32
```



# Break

- 15 min



# Association rules



# Association rules

- Association rule mining finds interesting association or correlation relationships among a large set of data items.
- With massive amounts of data continuously being collected and stored, many industries are becoming interested in mining association huge amounts of business transaction records can help in many business decision making processes, such as catalogue design, cross-marketing, and loss-leader analysis.



A typical example of association rule mining is market basket analysis.



# Association rules

**Analyses and predicts customer behaviour.**

If / then statements.

Examples:

If Bread => is likely to buy Butter.

If someone purchase bread then he/she likely to purchase butter.

Buys{onions, potatoes}=> buys{tomatoes}



# Parts of association rules

Bread=>butter[20%, 45%]

Bread: Antecedent

Butter: Consequent

20% is Support

(an indication of how frequently the itemset appears in the dataset)

And 45% is Confidence

(an indication of how often the rule has been found to be true)



$A \Rightarrow B$

Support denoted probability that contains A

Confidence denotes probability that a transaction containing A also contains B.



# Support and Confidence

Consider in a super market:

Total transactions: 100

Bread: 20

$20/100 * 100 = 20\%$  which is **support**.

In 20 transaction of bread, butter : 9 transactions

So,  $9/20 * 100 = 45\%$  which is **confidence**.



# Applications

- Web usages mining
- Banking
- Bio informatics
- Market based analysis
- Credit/ debit card analysis
- Product clustering
- Catalog design



# Apriori Algorithm Association Rules

- If you brought tooth brush, there will be suggestion of tooth paste or if you brought beer there will be suggestion of chips and potato cracker etc.
- Many ecommerce websites are using these trends of suggestion in market. This is called Apriori Algorithms. This is machine learning algorithms and a lot of ecommerce websites (like flipcart, amazon) are using this.



# Apriori Algorithm

## Association Rules

Let us assume:

Our minimum acceptable Support is 60%

Our minimum acceptable Confidence is 80%

Consider the below transaction table:

Transaction ID	Item set
T <sub>1</sub>	M, O, N, K, E, Y
T <sub>2</sub>	D, O, N, K, E, Y
T <sub>3</sub>	M, A, K, E
T <sub>4</sub>	M, U, C, K, Y
T <sub>5</sub>	C, O, O, K, E

Min Support means:  $(60/100)*5 = 3$

Min Confidence means:  $(80/100)*5 = 4$



# Apriori

One item set scenarios:

Item Set	M	O	N	K	E	Y	D	A	U	C
Support Count	3	4	2	5	4	3	1	1	1	2

L1: (The item set which are frequently repeating using minimum support):

Item Set	Support Count
M	3
O	4
N	2
K	5
E	4
Y	3



# Apriori

Two item set scenarios:

Item Set	M, O	M, K	M, E	M, Y	O, K	O, E	O, Y	K, E	K, Y	E, Y
Support Count	1	3	2	2	3	3	2	4	3	2

L2: (The item set which are frequently repeating using minimum support):

Item Set	Support Count
M, K	3
O, K	3
O, E	3
K, Y	3
K, E	4



# Apriori

Three item set scenarios:

Item Set	M, K, O	M, K, E	M, K, Y	O, K, E	O, K, Y
Support Count	1	2	2	3	2

L3: (The item set which are frequently repeating using minimum support):

Item Set	O, K, E
Support Count	3



# Now create association rules with support and confidence for O, K, E.

Association rules as like

O AND K GIVES E

Confidence= (support/no of time it occur i.e. O AND K OF  $O \wedge K \Rightarrow E$ )

For example confidence for o and k =  $(3/3)=1$

Association Rule	Support	Confidence	Confidence %
$O \wedge K \Rightarrow E$	3	$3/3=1$	100
$O \wedge E \Rightarrow K$	3	$3/3=1$	100
$K \wedge E \Rightarrow O$	3	$3/4=0.75$	75
$E \Rightarrow O \wedge K$	3	$3/4=0.75$	75
$K \Rightarrow O \wedge E$	3	$3/5=0.6$	60
$O \Rightarrow K \wedge E$	3	$3/4=0.75$	75

Compare them with Confidence level 80%:

Association Rule	Support	Confidence	Confidence %
$O \wedge K \Rightarrow E$	3	$3/3=1$	100
$O \wedge E \Rightarrow K$	3	$3/3=1$	100

Now this is called market basket analysis.



# Class Activity

Assume:

Minimum support:2, Minimum confidence:70%. Use Apriori algorithm to get frequent itemsets and strong association rules.

Transaction ID	Items
1	I1, I3, I4
2	I2, I3, I5
3	I1, I2, I3, I5
4	I2, I5



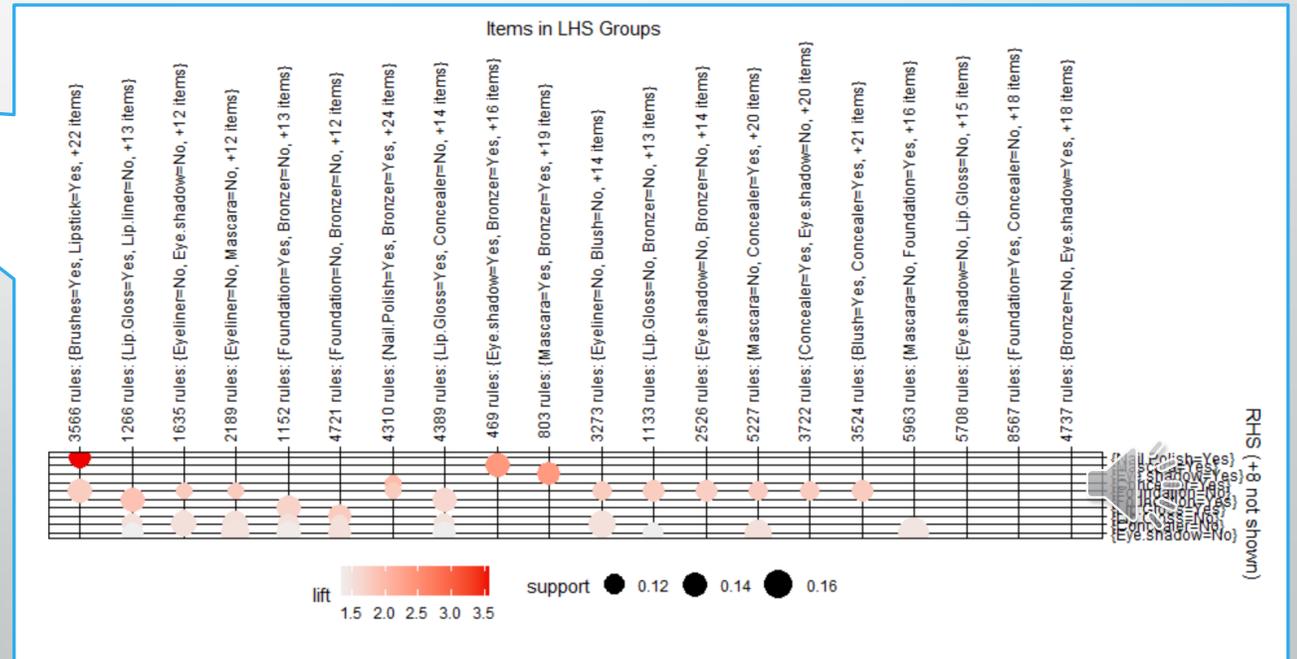
# Association Rules in R

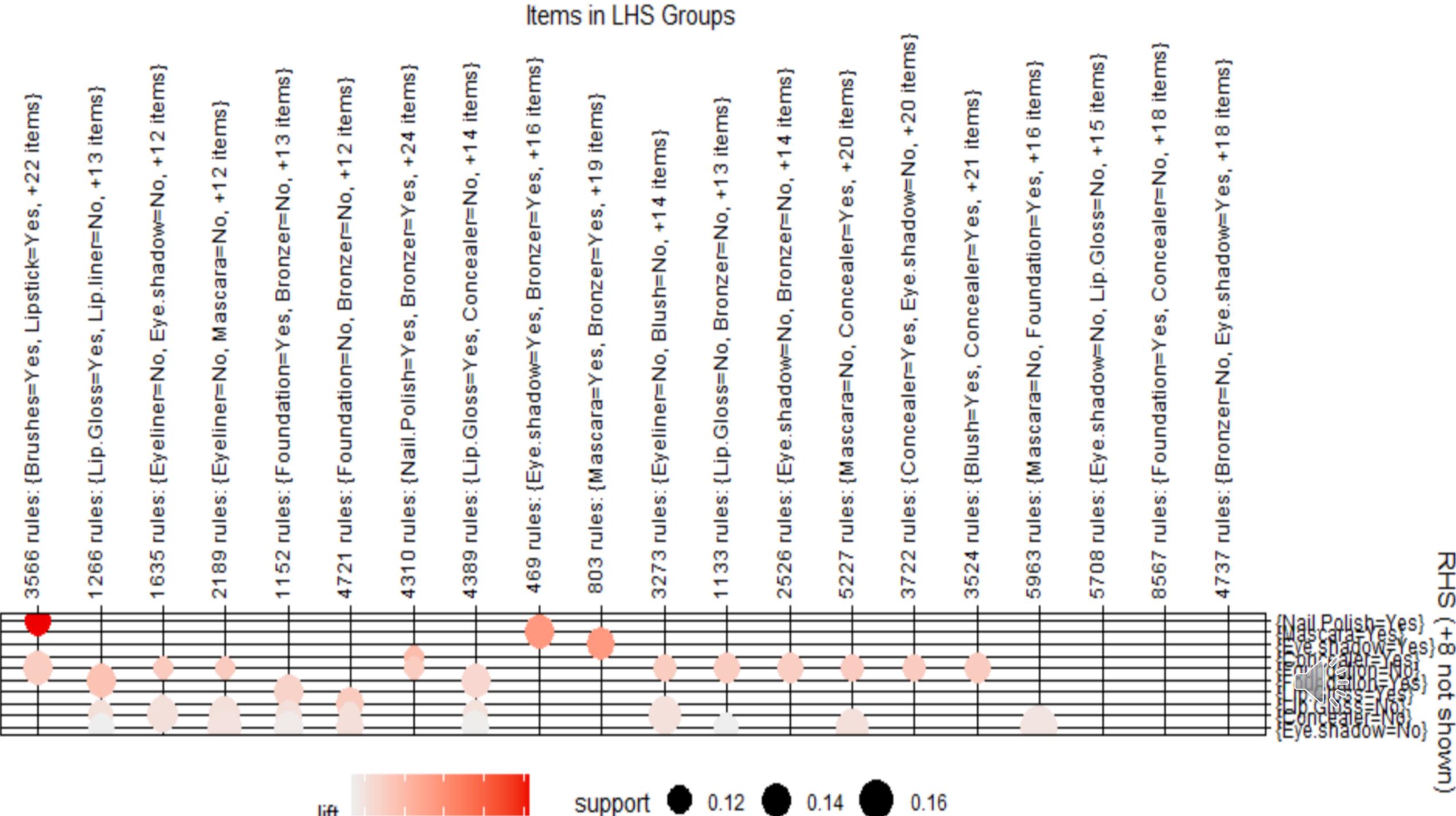
```
# Reading the data file
mydata<-read.csv("Cosmetics.csv",header=T)
```

```
# Finding association rules
library(arules)
rules <- apriori(mydata)
```

```
# Graphs and Charts
library(arulesViz)
plot(rules,method="grouped")
```

Bag	Blush	Nail.Polish	Brushes	Concealer	Eyebrow.Pencils	Bronzer	Lip
No	Yes	Yes	Yes	Yes	No	Yes	Ye
No	No	Yes	No	Yes	No	Yes	Ye
No	Yes	No	No	Yes	Yes	Yes	Ye
No	No	Yes	Yes	Yes	No	Yes	Nc
No	Yes	No	No	Yes	No	Yes	Ye
No	No	No	No	Yes	No	No	Nc
No	Yes	Yes	Yes	Yes	No	Yes	Ye
No	No	Yes	Yes	No	No	Yes	Nc
No	No	No	No	Yes	No	No	Nc
Yes	Yes	Yes	Yes	No	No	No	Nc
No	No	Yes	No	No	No	Yes	Nc
No	No	Yes	Yes	Yes	No	Yes	Nc
No	Yes	No	No	Yes	No	No	Ye



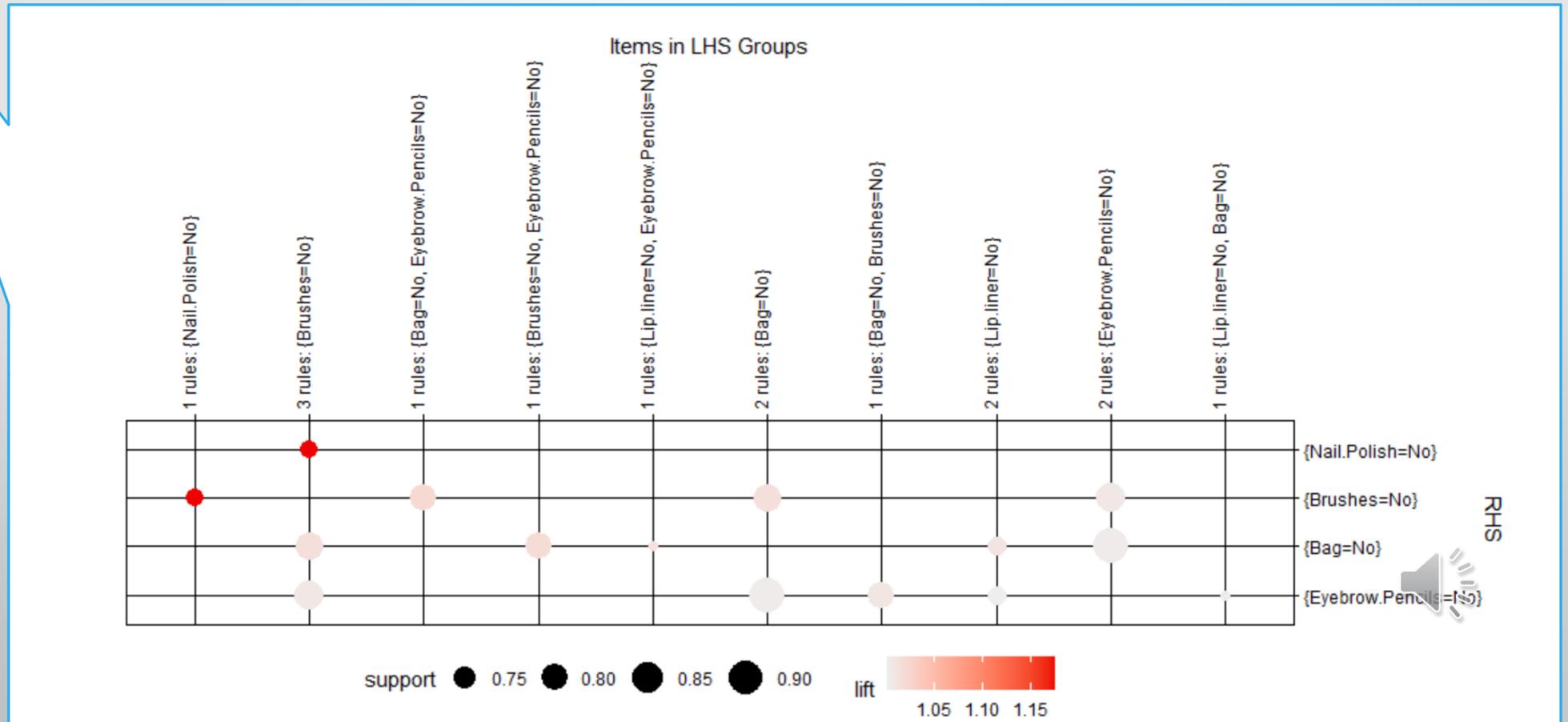


# Association Rules in R

# Rules with specified parameter values

```
rules <- apriori(mydata, parameter = list(minlen=2, maxlen=10, supp=.7, conf=.8))
```

```
plot(rules, method="grouped")
```

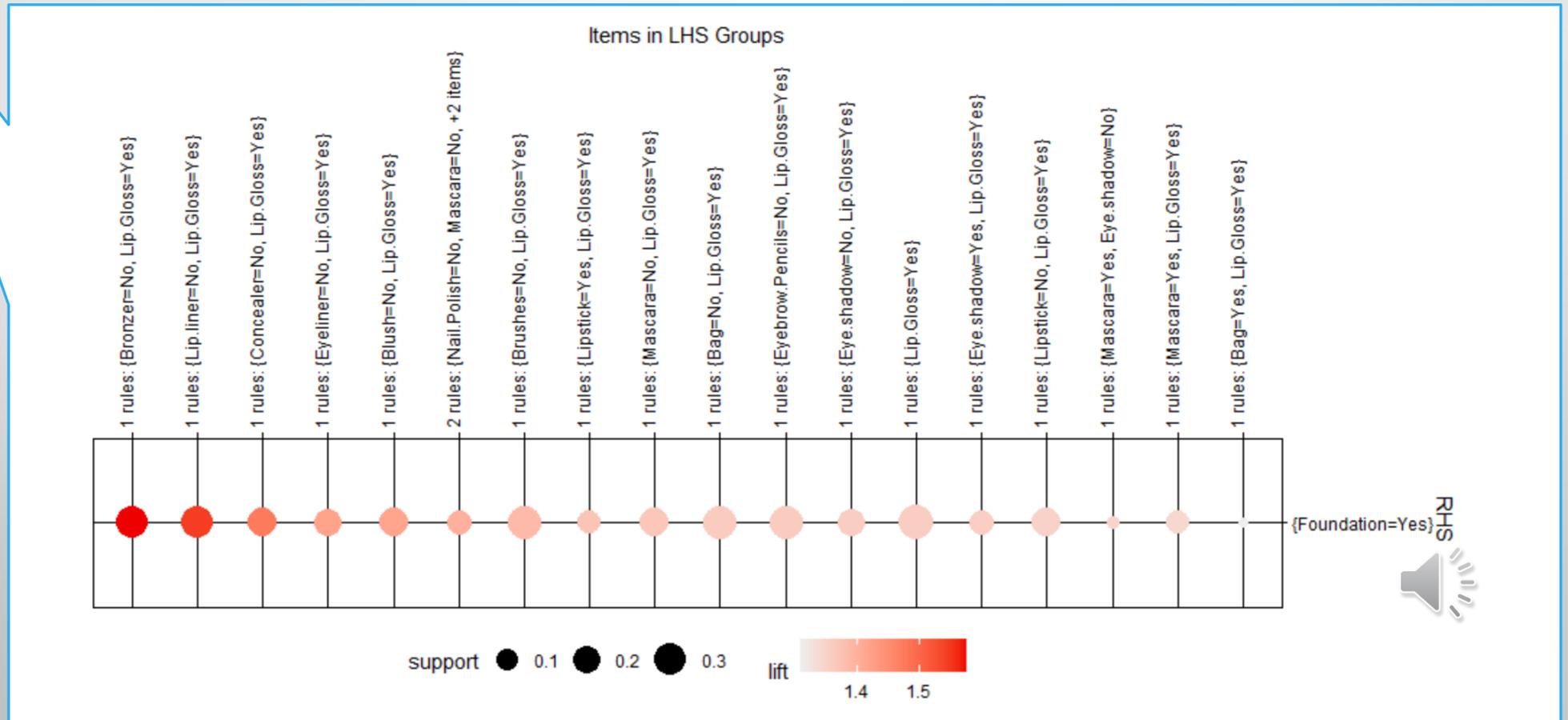


# Association Rules in R

# Finding interesting rules

```
rules <- apriori(mydata,parameter = list(minlen=2, maxlen=3,supp=.01,  
conf=.7),appearance=list(rhs=c("Foundation=Yes"),lhs=c("Bag=Yes", "Blush=Yes"),default="lhs"))
```

```
plot(rules,method="grouped")
```

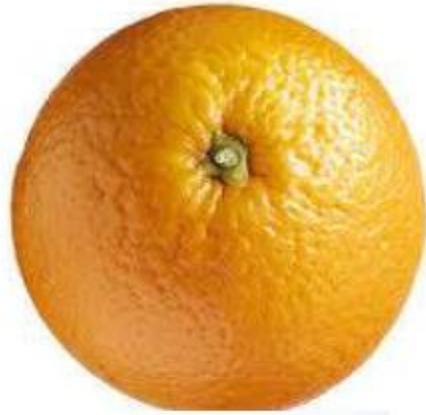


# Clustering

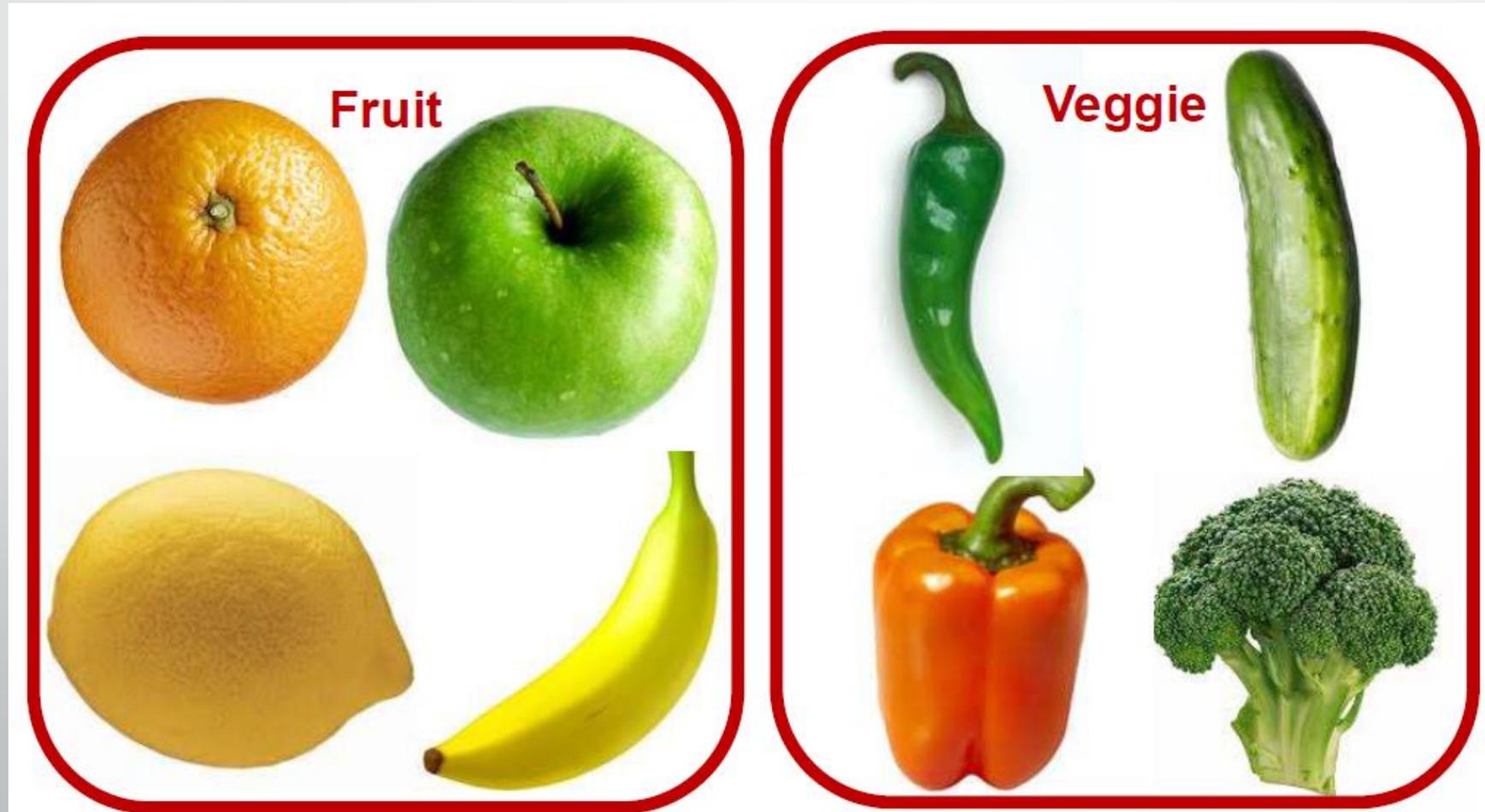


# What is Clustering?

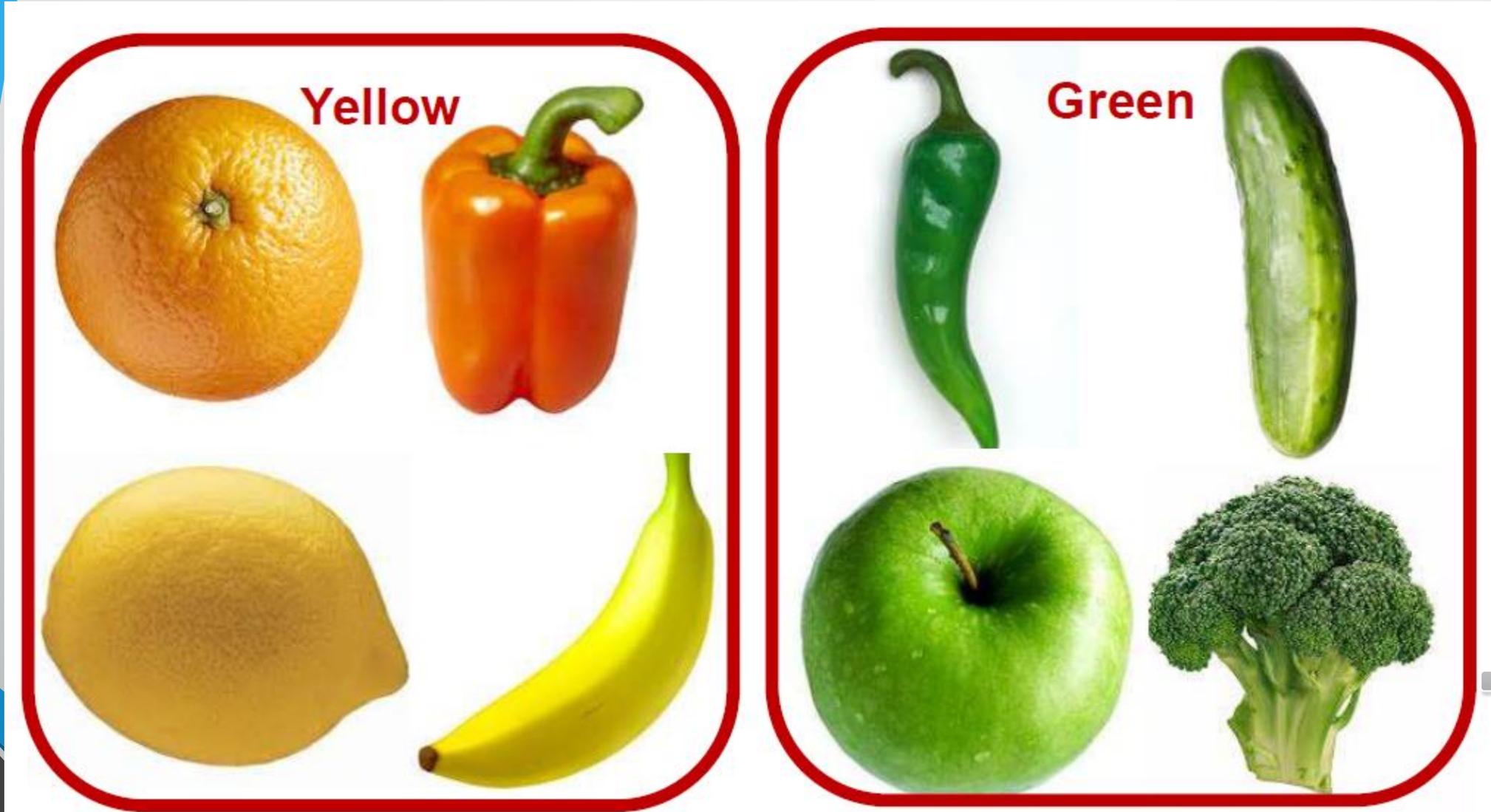
**Grouping of objects**



# Clustering I (By Type)



# Clustering II (By Colour)



# Another example

Original



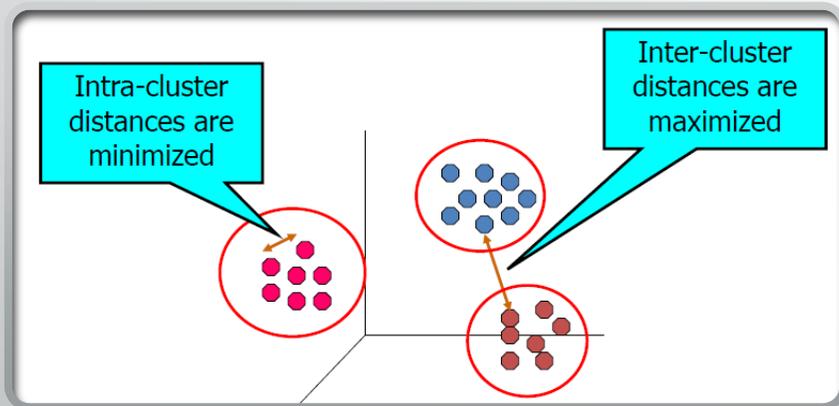
Clustering 1



Clustering 2



# What is Cluster Analysis?



**Grouping of objects** into classes such a way that

1. Objects in **same cluster** are **similar**
2. Objects in **different clusters** are **dissimilar**

## Segmentation vs. Clustering

1. Clustering is finding borders between groups,
2. Segmenting is using borders to form groups

Clustering is the method of creating segments



# General review, Clustering versus Classification

## **Classification – Supervised**

- Classes are predetermined
- we know in advance the stamping
- For example if we already diagnosed some disease

## **Clustering – Unsupervised**

- Classes are not known in advance
- we don't know in advance the stamping
- Market behaviour segmentation
- Or Gene analysis



# General Applications of Clustering

- Marketing: segmentation of the customer based on behavior
- Banking: ATM Fraud detection (outlier detection)
- ATM classification: segmentation based on time series
- Gene analysis: Identifying gene responsible for a disease
- Chemistry: Periodic table of the elements
- Image processing: identifying objects on an image (face detection)
- Insurance: identifying groups of car insurance policy holders with a high average claim cost
- Houses: identifying groups of houses according to their house type, value, and geographical location



# Similarities versus Dissimilarities

**There is no single definition of similarity or dissimilarity between data objects.**

**The definition of similarity or dissimilarity between objects depends on:**

- the **type** of the data considered
- what **kind of similarity** we are looking for



# Distance Measure

**Similarity/dissimilarity** between objects is often expressed in terms of a **distance measure**  $d(x,y)$

Ideally, every distance measure should be a **metric**, i.e., it should satisfy the following conditions:

1.  $d(x,y) \geq 0$
2.  $d(x,y) = 0$  if  $x = y$
3.  $d(x,y) = d(y,x)$
4.  $d(x,z) \leq d(x,y) + d(y,z)$



# Major Clustering Methods

**Partitioning algorithms:** Construct various partitions and then evaluate them by some criterion

**Hierarchy algorithms:** Create a hierarchical decomposition of the set of data (or objects) using some criterion

**Density-based:** based on connectivity and density functions

**Grid-based:** based on a multiple-level granularity structure

**Model-based:** A model is hypothesized for each of the clusters and the idea is to find the best fit of that model to each other



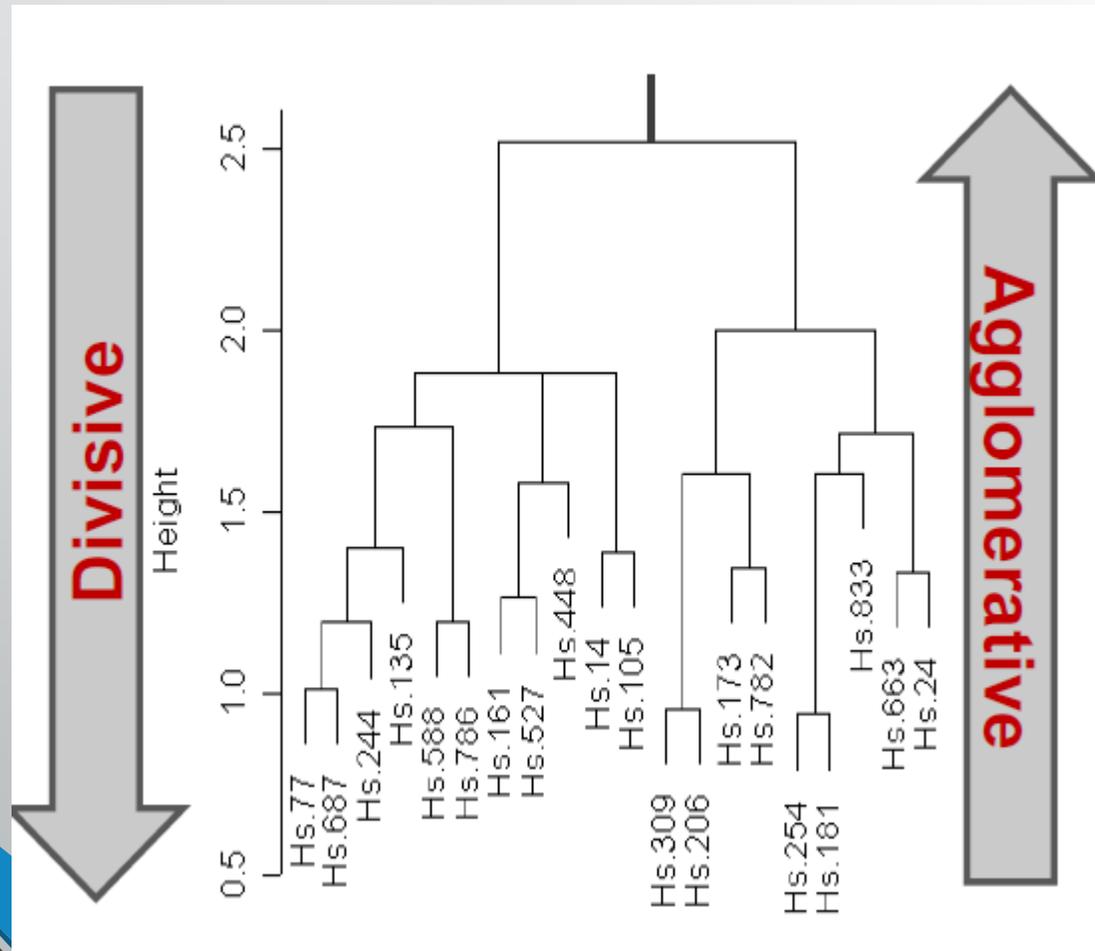
# Hierarchical Clustering

Construct a hierarchy of clusters not just a single partition of objects

- Use distance matrix as clustering criteria
- Does not require the number of clusters as an input, but needs a termination condition, e. g., number of clusters or a distance threshold for merging



# Hierarchical clustering



The hierarchy of clustering is given as a **clustering tree** or **dendrogram**

- leaves of the tree represent the individual objects
- internal nodes of the tree represent the clusters

Two main types of hierarchical clustering

- **agglomerative (bottom-up)**
  - place each object in its own cluster (a singleton)
  - merge in each step the two most similar clusters until there is only one cluster left or the termination condition is satisfied
- **divisive (top-down)**
  - start with one big cluster containing all the objects
  - divide the most distinctive cluster into smaller clusters and proceed until there are  $n$  clusters or the termination condition is satisfied

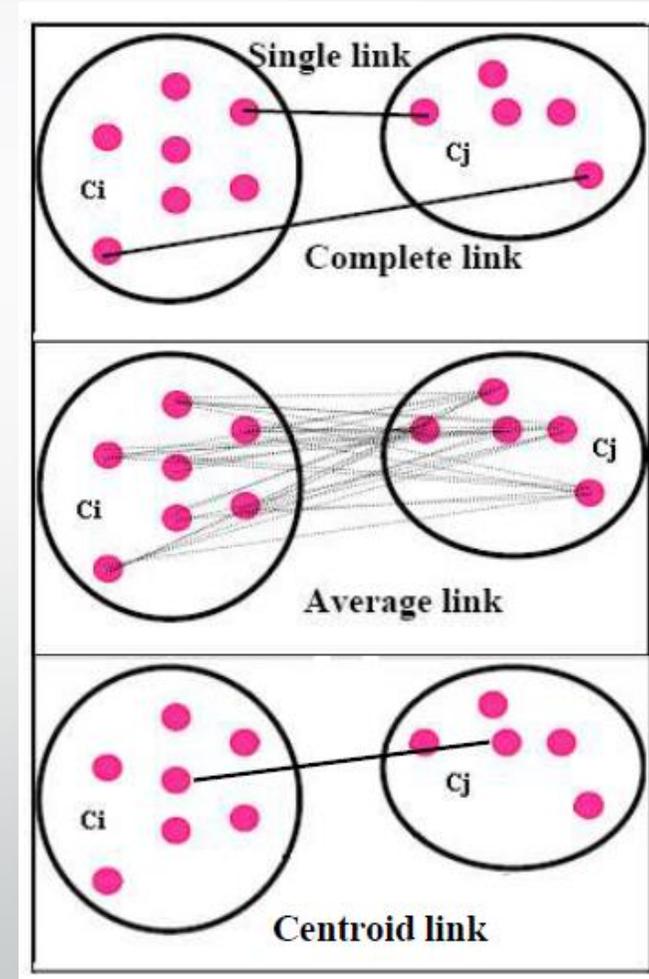
# Hierarchical Clustering Distance Measures

**Single link (nearest neighbor).** The distance between two clusters is determined by the distance of the two closest objects (nearest neighbors) in the different clusters.

**Complete link (furthest neighbor).** The distances between clusters are determined by the greatest distance between any two objects in the different clusters (i.e., by the "furthest neighbors").

**Pair-group average link.** The distance between two clusters is calculated as the average distance between all pairs of objects in the two different clusters.

**Pair-group centroid.** The distance between two clusters is determined as the distance between centroids.



# Hierarchical Clustering in R

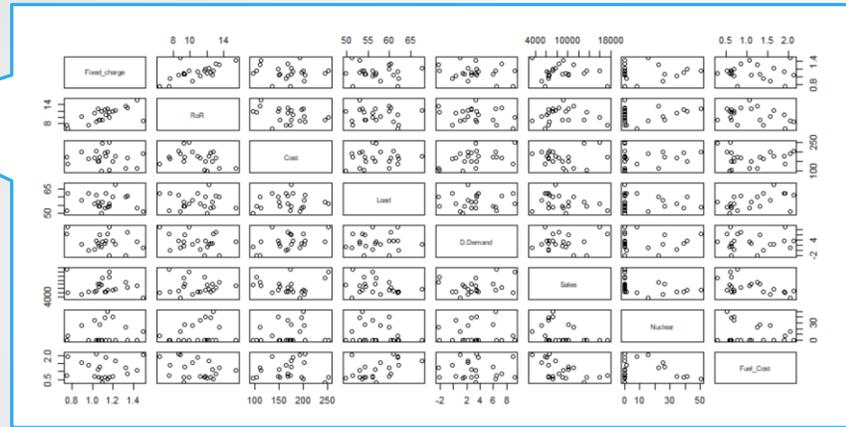
```
hcddata <- read.csv("utilities.csv", header=T)
```

```
#Basic statistics and visualisation
```

```
str(hcddata)
```

```
head(hcddata)
```

```
pairs(hcddata[2:9])
```

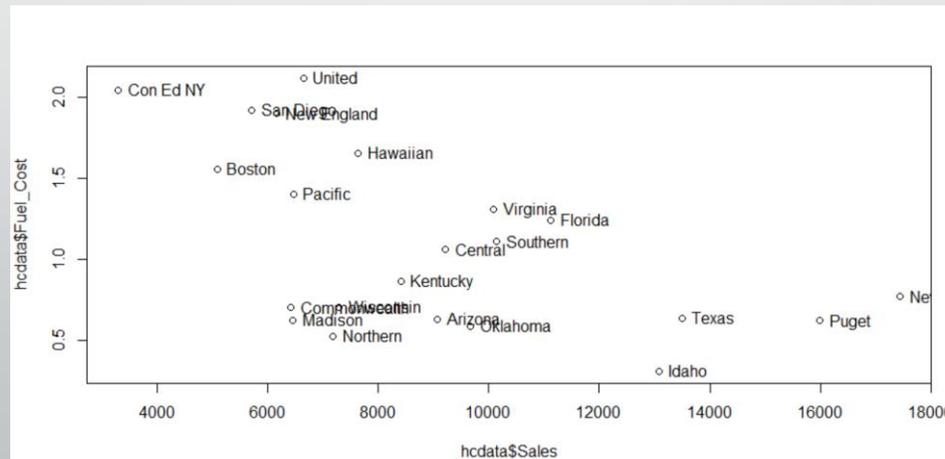


	Company	Fixed_charge	RoR	Cost	Load	D.Demand	Sales	Nuclear	Fuel_Cost
1	Arizona	1.06	9.2	151	54.4	1.6	9077	0.0	0.628
2	Boston	0.89	10.3	202	57.9	2.2	5088	25.3	1.555
3	Central	1.43	15.4	113	53.0	3.4	9212	0.0	1.058
4	Commonwealth	1.02	11.2	168	56.0	0.3	6423	34.3	0.700
5	Con Ed NY	1.49	8.8	192	51.2	1.0	3300	15.6	2.044
6	Florida	1.32	13.5	111	60.0	-2.2	11127	22.5	1.241
7	Hawaiian	1.22	12.2	175	67.6	2.2	7642	0.0	1.652
8	Idaho	1.10	9.2	245	57.0	3.3	13082	0.0	0.309
9	Kentucky	1.34	13.0	168	60.4	7.2	8406	0.0	0.862
10	Madison	1.12	12.4	197	53.0	2.7	6455	39.2	0.623
11	Nevada	0.75	7.5	173	51.5	6.5	17441	0.0	0.768
12	New England	1.13	10.9	178	62.0	3.7	6154	0.0	1.897
13	Northern	1.15	12.7	199	53.7	6.4	7179	50.2	0.527
14	Oklahoma	1.09	12.0	96	49.8	1.4	9673	0.0	0.588
15	Pacific	0.96	7.6	164	62.2	-0.1	6468	0.9	1.400

```
# Scatter plot
```

```
plot(hcddata$Fuel_Cost~ hcddata$Sales, data = hcddata)
```

```
with(hcddata,text(hcddata$Fuel_Cost ~ hcddata$Sales, labels=hcddata$Company,pos=4))
```



# Hierarchical Clustering in R

```
# Normalize
```

```
z <- hcdata[,-c(1,1)]
```

```
means <- apply(z,2,mean)
```

```
sds <- apply(z,2,sd)
```

```
nor <- scale(z,center=means,scale=sds)
```

```
# Calculate distance matrix
```

```
distance = dist(nor)
```

```
print(distance, digits=3)
```

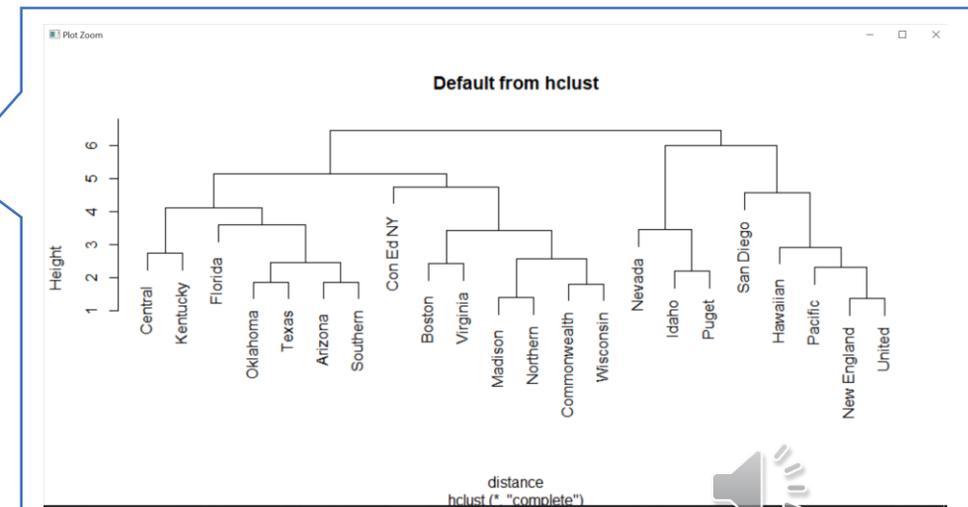
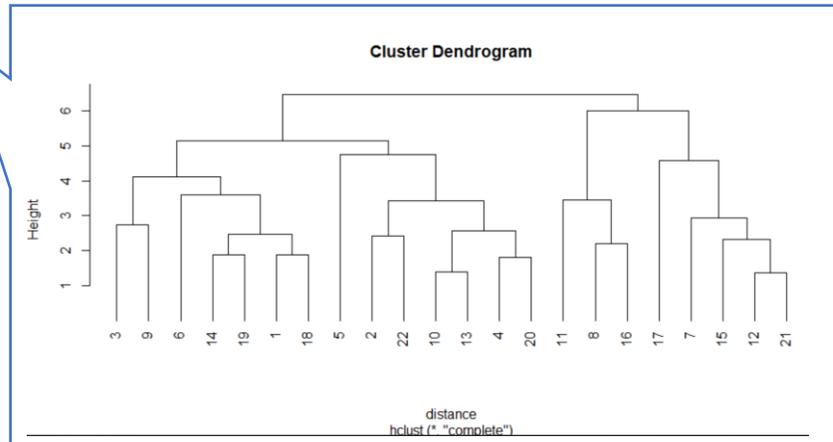
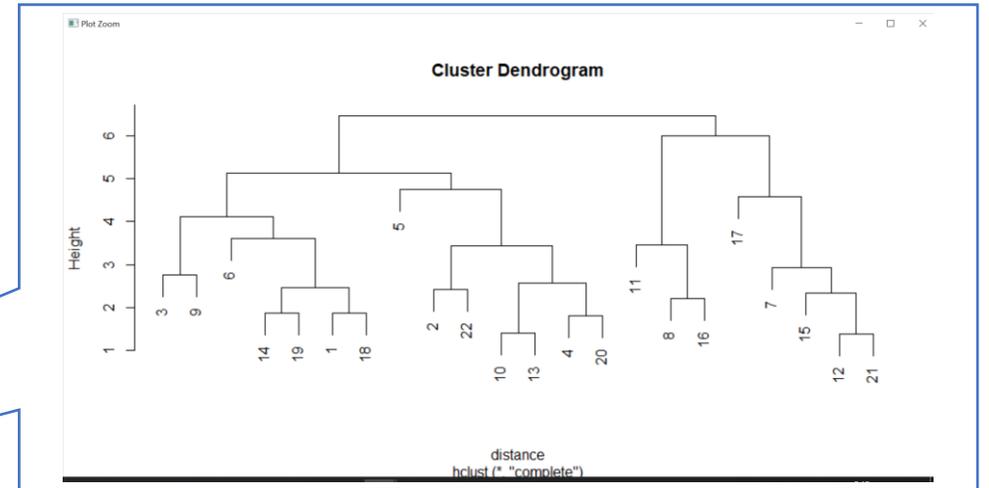
```
# Hierarchical agglomerative clustering
```

```
hcdata.hclust = hclust(distance)
```

```
plot(hcdata.hclust)
```

```
plot(hcdata.hclust,labels=hcdata$Company,main='Default from hclust')
```

```
plot(hcdata.hclust,hang=-1)
```



# Hierarchical Clustering in R

```
# Hierarchical agglomerative clustering using "average" linkage
hclust.hclust<-hclust(distance,method="average")
plot(hclust.hclust,hang=-1)
```

# Cluster membership

```
member = cutree(hclust.hclust,3)
table(member)
```

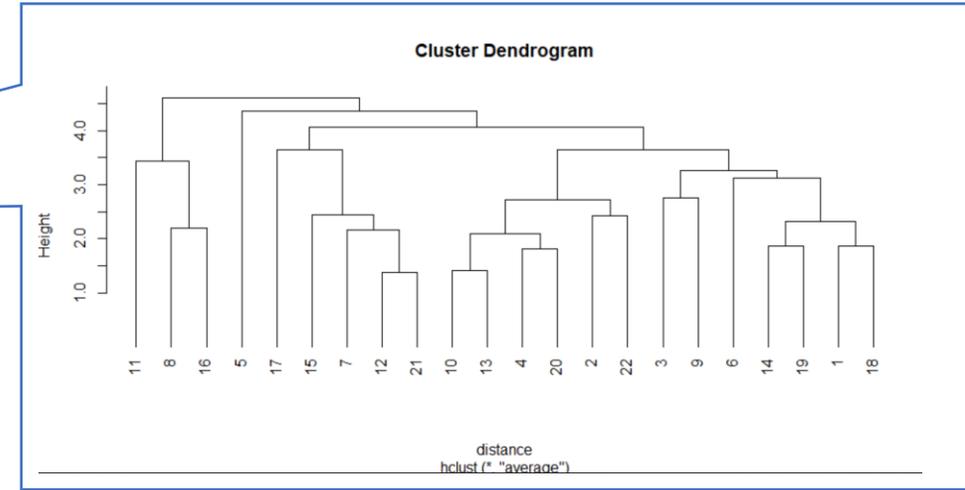
```
member
  1  2  3
18  1  3
```

# Characterizing clusters

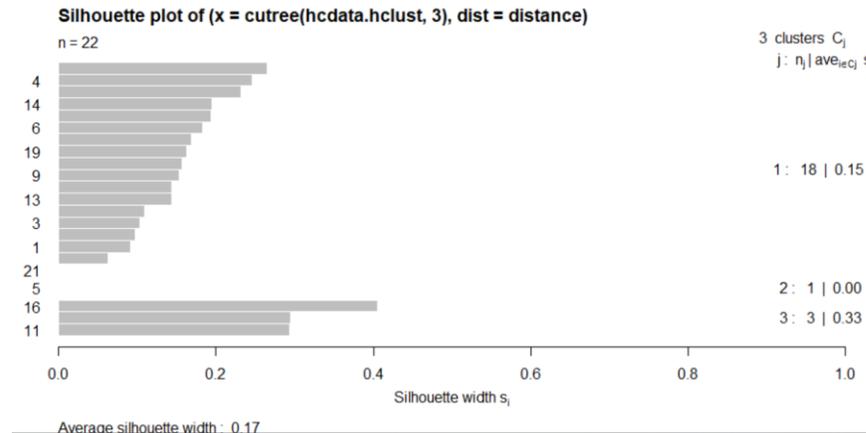
```
aggregate(nor,list(member),mean)
aggregate(hclust[,-c(1,1)],list(member),mean)
```

# Silhouette Plot

```
library(cluster)
plot(silhouette(cutree(hclust.hclust,3), distance))
```

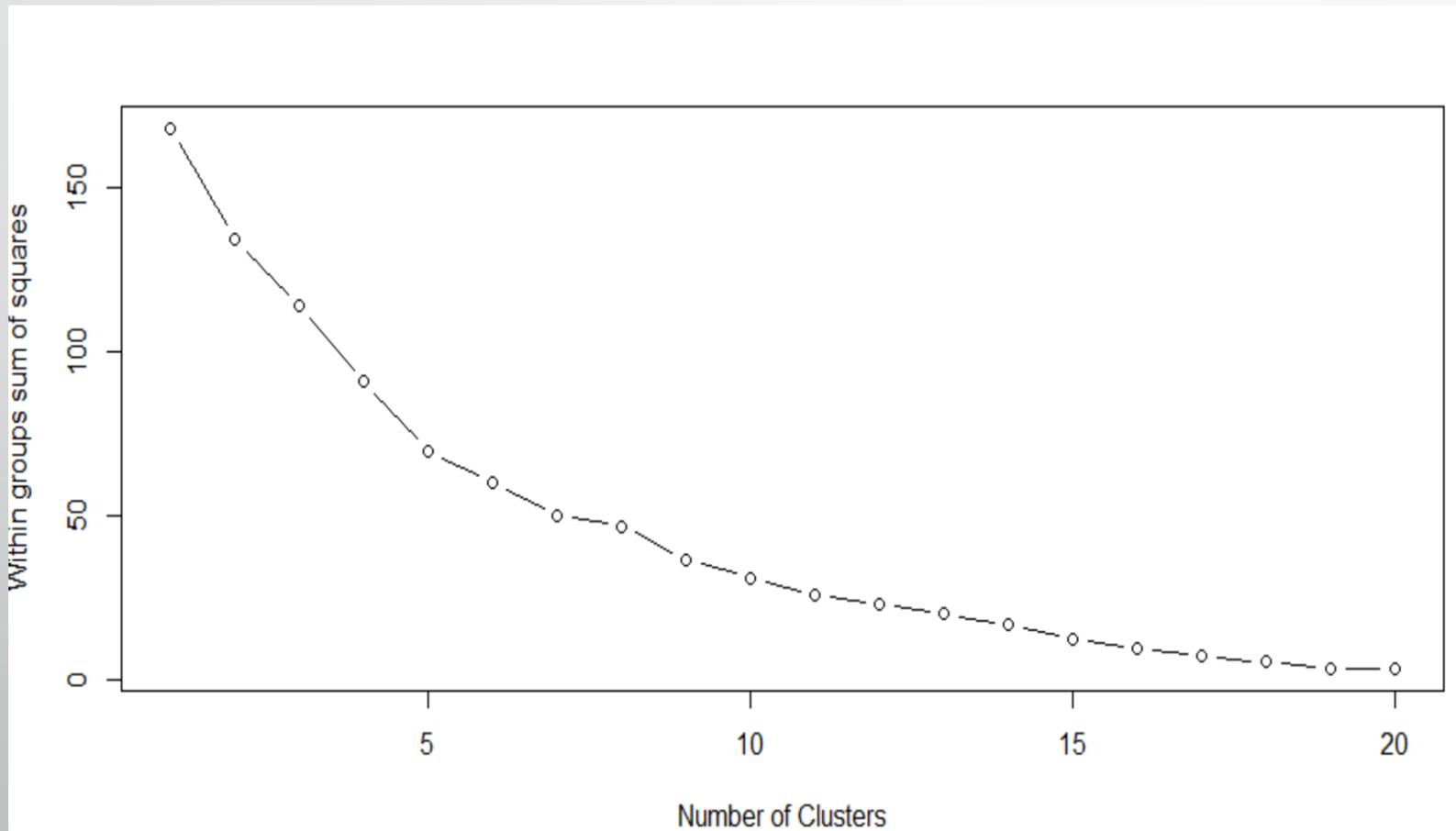


Group	1	Fixed_charge	RoR	Cost	
1	1	1.111667	11.155556	157.6667	
2	2	1.490000	8.800000	192.0000	
3	3	1.003333	8.866667	223.3333	
	Load	D.Demand	Sales	Nuclear	Fuel_Cost
1	57.65556	2.850000	8127.50	13.8	1.1399444
2	51.20000	1.000000	3300.00	15.6	2.0440000
3	54.83333	6.333333	15504.67	0.0	0.5656667



## # Scree Plot

```
wss <- (nrow(nor)-1)*sum(apply(nor,2,var))  
for (i in 2:20) wss[i] <- sum(kmeans(nor, centers=i)$withinss)  
plot(1:20, wss, type="b", xlab="Number of Clusters", ylab="Within groups sum of squares")
```





Questions:

- Today's course
- Assignment 2

