dotplot (mtcars $mpg, labels = row.names(x), cex=,
groups = $cyl, gcolor = "black", color = "black",
Pch=19, main = "Dot plot example".
xlab = "Miles per Gallon")

UNIT-3

Probability distribution:

Probability distribution lie at heart of statistics, so naturally R provides numerous functions for making use of function like generating random number & calculating the distribution & quantile.

* To draw random numbers from natural distribution use rnorm (Pick) function.

Ex: rnorm (n=10)

or else, we can also specify values

Ex: rnorm (n=10, mean =100, sd = 20)

* dnorm is used to calculate density (Probability of a particular value) for the normal distribution.

Ex:
```
x = rnorm (10)
dnorm (x)
```

* dnorm returns the probability of a specific number occuring.

Like with rnorm, a mean and standard deviation can be specified for dnorm.

Eg: randNorm <- rnorm(10)
randNorm
dnorm(randNorm)
dnorm(c(-1,0,1))

Ex: r <- rnorm(30000)
rdensity <- dnorm(r)
require(ggplot2)
ggplot(data.frame(x=r, y=rdensity)) + aes(x=x, y=y) + geom_point() + labs(x="Random Normal variable", y="Density")

constructs aesthetic mapping

Normal distribution: is also known as the Gaussian distribution is the probability distribution that plots all of its values in a symmetrical fashion, and most of the results are situated around the probability's mean.

Many things follow a Normal Distribution:

(i) heights of a people.
(ii) size of things produced by machines
(iii) errors in measurement.
(iv) blood pressure.
(v) marks on a test.

The normal distribution has

Mean = median = mode.

Standard deviation.

The standard deviation is measure of how spread numbers are (or) standard deviation.

Variance: The average of squared differences from the mean.

Standard scores:

The no. of standard deviations from the mean is also called the "standard score", "sigma" or "z-score"

⇒ The opposite of Pnorm is qnorm. Give cumulative (property) Probability & Freq the quantile

Q. How many high must an individuals

Binomial distributions

$$P(x; n, p) = \binom{n}{x} p^x (1-p)^{n-x}$$

$$nCr = \frac{n!}{x!(n-x)!}$$

n is no. of trials P - probability of success of trial

rbinorm (n=1, size = 10, Prob=0.4)

→ rbinorm (n=1, size=10, Prob=0.4) output [1] 3

→ rbinorm (n=5, size=10, Prob=0.4) [1] 5 3 6 5 4

→ rbinom (n=10 size =10, Prob=0.4) [1] 5 3 4 4 4 5 3 5 3 3

→ binomdata <- data.frame(succes = rbinorm(n=1000, size =10, Prob=0.3)

ggplot (binomdata,)

1) find out the no.of possible outcomes when the coin is rolled (or) flipped for 10 times and also calculate the no.of success? (no.of occurences of heads)

$P(X=0) = {}^{10}C_0 = 1$

2) find the out the possible outcomes when a die is rolled for 5 times and calculate the no.of times the face 4 and no.of times face 6 occurs?

$\Rightarrow 6^5$

2^{10}.

1) $P(X=0) = {}^{10}C_0 = 1$

$P(X=1) = {}^{10}C_1 = \frac{10!}{1!\,9!} = 10$

$P(X=2) = {}^{10}C_2 = \frac{10!}{2!\,8!} = \frac{10\times 9\times 8!}{2\times 8!} = 45$

$P(X=3) = {}^{10}C_3 = \frac{10!}{3!\,7!} = \frac{10\times 9\times 8\times 7!}{3\times 2\times 7!} = 120$

$P(x=4) = {}^{10}C_4 = \frac{10 \times 9 \times 8 \times 7 \times 6!}{4 \times 3 \times 2 \times 1 \times 6!} = 210$

$P(x=5) = {}^{10}C_5 = \frac{10 \times 9 \times 8 \times 7 \times 6 \times 5!}{5 \times 4 \times 3 \times 2 \times 1 \times 5!} = 252$

$P(x=6) = {}^{10}C_6 = \frac{10 \times 9 \times 8 \times 7 \times 6 \times 5 \times 4!}{6 \times 5 \times 4 \times 3 \times 2 \times 1 \times 4!} = 210$

$P(x=7) = {}^{10}C_7 = \frac{10!}{7!(3!)} = \frac{10 \times 9 \times 8 \times 7 \times 6 \times 5 \times 4 \times 3!}{7! \times 3 \times 2} = 120$

$P(x=8) = {}^{10}C_8 = \frac{10!}{8!\,2!} = \frac{10 \times 9 \times 8!}{8! \times 2} = 45$

$P(x=9) = {}^{10}C_9 = \frac{10!}{9!\,1!} = \frac{10 \times 9!}{9! \times 1} = 10$

$P(x=10) = {}^{10}C_{10} = \frac{10!}{10!\,0!} = \frac{1}{1} = 1$

3. Write a function for calculating density binomial distribution for probability of the occurence of 5 succes out of 10 trials. The probability of succes is 50%

Basic Statistics:

Statistics: "a branch of mathematics used to summarize, analyze, and interpret a group of number of observations".

- population: any group of interest or any group that researchers

Two types of statistics

- Descriptive statistics
- Inferential statistics.

1. Descriptive: are procedures used to summarize, organize set of measurements.

2. Inferential: procedures used that allow researchers to infer or generalize observations made with samples to the larger population from which they were selected.

Descriptive.

- use descriptive statistics to communicate with other researchers and the public.
- Descriptive statistics: central tendency and dispersion.

Measures of central tendency: We use statistical measures to locate a single score that is most representative of all scores in a distribution.

- Mean, Median, Mode.

• The notations used

Population : N

Sample : n

Mean for sample - $\bar{x}$

for population mean - μ

• Median :

- Data: 2 3 4 5 7 10 80 mean of these scores is 15.86

- 80 is an outlier (out of boundary)

- Mean fails to reflect most of the data. We use median instead of mean to remove the influence of an outlier.

- Median is the middle value in a distribution of data listed in a numeric order.

Mode:

- The value in a data set that occurs most often or most frequently.

Ex: 2, 3, 3, 3, 4, 4, 4, 4, 7, 8, 8, 8

Mode = 4

• Dispersion (variability) : A measure of the spread of scores in a distribution.

- Range, variance and standard deviation.

Range: difference b/w largest and smallest value

Variance: measures the averaged squared distance that scores deviate from the mean.

$$s^2 = \frac{\Sigma(x-\bar{x})^2}{n-1} \quad \text{or} \quad \frac{SS}{n-1}$$ (sum of squares)

↓

degree of freedom: the no. of scores in a sample that are free to vary.

Standard deviation: Square root of variance.

Correlation:

→ The correlation is one of the most common and most useful statistics.

→ A correlation is a single number that describes the degree of relationship between two variables

→ It is a statistical method which enable the researcher to find whether two variables are related and to what extent they are related.

→ correlation is considered as the sympathetic movement of two or more variables.

→ We can observe this when a change in one particular variable is accompanied by changes in other variables as well, as this happens either in the same or opposite direction, then the resultant variables are said to be correlated.

→ The word correlation is made of co-(meaning "together") and Relation.

- correlation is positive when the values increase together, and
- correlation is negative when one value decreases as the other increases.

Correlation is of three types:

- positive
- Negative
- No correlation.

↓

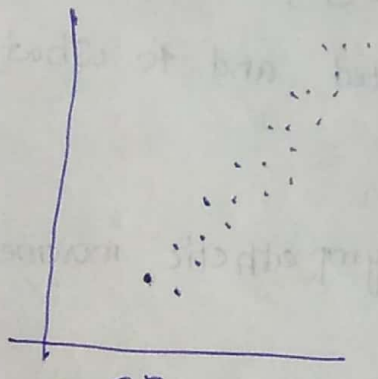
There is no change in a variable with any change in other variable.

Correlation can have a value:

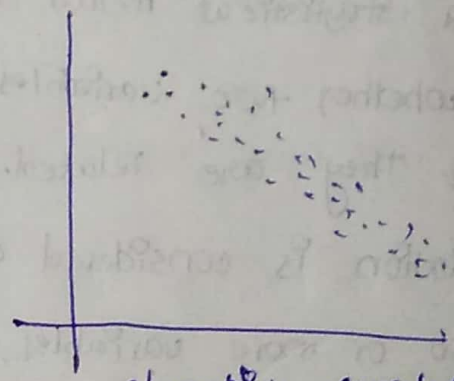
1 is a perfect positive correlation

0 is no correlation

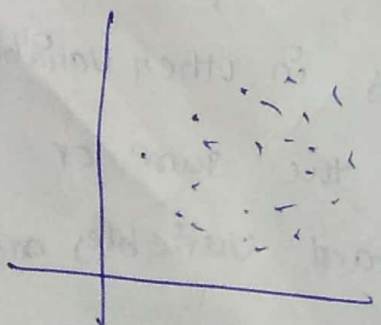
−1 is a perfect negative correlation.



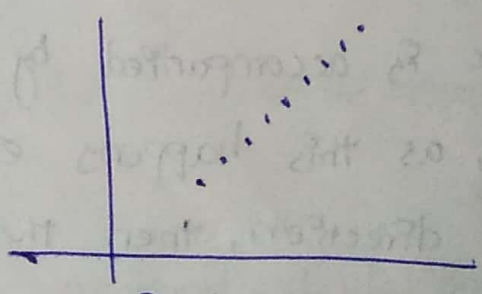
positive correlation



Negative correlation.



No correlation



Perfect positive correlation.

- If the points are close to each other than it is said to be strong correlation if the points are far from each other then it is negative correlation.

$$\text{correlation } r = \frac{n\Sigma xy - \Sigma x \Sigma y}{\sqrt{n\Sigma x^2 - (\Sigma x)^2}\sqrt{n\Sigma y^2 - (\Sigma y)^2}}$$

Ex:

Let's assume that we want to look the relationship exist b/w Height and self-esteem.

Hypothesis Statement: correlation coefficient (r) is equal to zero

Alternate Hypothesis: correlation coefficient (r) is not equal to zero.

Person	Height (x)	Self Esteem (y)	$n=10$
1	68	4.1	
2	71	4.6	
3	62	3.8	
4	75	4.4	
5	58	3.2	
6	60	3.1	
7	67	3.8	
8	68	4.1	
9	71	4.3	
10	69	3.7	

$\Sigma xy = 2980.6$ 2657 $(\Sigma x)^2 = 44756$

There are mainly three coefficients of correlation.

1. Karl Pearson's coefficient of correlation.
2. Pearson's rank
3. concurrent correlation

1. $$r = \frac{n\Sigma xy - \Sigma x \Sigma y}{\sqrt{n\Sigma x^2 - (\Sigma x)^2}\sqrt{n\Sigma y^2 - (\Sigma y)^2}}$$

$r = -1$ We say there is perfect negative correlation.

r between -1 & 0 We say that negative correlation.

$r = 0$ We say that it is no correlation

$r = 1$ We say perfect positive correlation

$r =$ b/w 0 & 1 positive correlation.

finding correlation coefficient can be done in 3 methods.

1. pearson's method
2. Spearman's method.
3. Kendall method

If we wanted to find in other method, the command is

cor(x, y, method = "method_name")

What is a t-test?

A t-test is a statistic

$$t = \frac{\text{variance between groups}}{\text{variance within groups}}$$

T test is often called Student's T test in the name of its founder "student".

→ T test is used to compare two different set of values

→ Generally applied to normal distribution which has a small set of values.

$$t = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}}$$

$\bar{x}_1$ = mean of first set of values

$\bar{x}_2$ = mean of second set of values

s_1 = standard deviation of first set of values.

s_2 = standard deviation of second set of values.

n_1 = Total no. of values in first set

n_2 = Total no. of values in second set

The t-test, and any statistical test of this sort, consists of three steps:

(i) Define the null and alternate hypothesis.

(ii) Calculate the t-statistics for the data.

(iii) compare t_{calc} to the tabulated t-value, for the appropriate significance level and degree of freedom

If $t_{calc} > t_{tab}$, we reject the null hypothesis and accept the alternate hypothesis. Otherwise, we accept the null hypothesis. and vice versa.

Regular day	exam	Field 1	Field 2	
16	32			
3	22			
17	23			
3	13	15.2	15.9	15.5
19	20	15.3	15.9	14.9
15	29	16.0	15.2	15.5
24	11	15.8	16.6	
23	25	15.6	15.2	
3.	13	14.9	15.8	
12	20	15.0	15.8	
32	80	15.4	16.2	
		15.6	15.6	
		15.7	15.6	
		15.5	15.8	
		15.2	15.5	
		15.5	15.5	
		15.1		
		15.3		
		15.0		

15.181
1642
4

15.17 - 21.63

16-15.1=
16.00
15.1

3
226.1
6

$\bar{x}_1 = 15.38$

$\bar{x}_2 = 15.68$

$s_1 = 0.31$

$s_2 = 0.40$

$$t = \frac{15.38 - 15.68}{\sqrt{\frac{(0.31)^2}{n_1} + \frac{(0.4)^2}{n_1}}}$$

0.31 × 0.31
0.31
093
000
0.0961
16

ANOVA (Analysis of variance)

After comparing two groups, the natural next step is comparing multiple groups.

$$f = \frac{\sum_i n_i (\bar{y}_i - \bar{y})^2 / (K-1)}{\sum_{ij} (y_{ij} - \bar{y}_i)^2 (N-K)}$$

$\bar{y}$ is overall mean $\bar{y}_i$ is mean of group.

y_{ij} observation j in group i.

n_i is no. of observations in group i.

N – total no. of observations. K – no. of groups.

Gro								
1	4.1	A	10	6.	B	19	6.	C
2	5	A	11	8.	B	20	7.	C
3	4.	A	12	4	B	21	6.	C
4	3.	A	13	5	B	22	6	C
5	2.	A	14	4.	B	23	7	C
6	4	A	15	6.	B	24	5	C
7	3	A	16	5.	B	25	6	C
8	4.	A	17	8.	B	26	5	C
9	4	A	18	6	B	27	5	C

3.7

$$f = \frac{9(3.7 - 5.22)^2 (3-1)}{\sum_{ij} (4 - 3.7)^2 (27-3)} = \frac{}{(0.09)(24)}$$

If the between group variation is significantly greater than the within group variation then it is likely that there is a stastitically significant difference between the groups.

→ If the b/w group variation is significantly greater than much larger than the within group variation, the means of different samples will not be equal (i;e is a stastically significant difference between the groups). If the b/w and within the group. variations are approximately the same size.

⇒ aov() is used to perform ANOVA model.

Manipulating Strings:
(Static Strings)

Paste: The function

Ex: Paste("Hello", "Jared", "and others")

[1] "Hello Jared and others".

→ sep, that determines what to put in b/w entries

Ex: Paste("Hello", "Jared", sep="/")

[1] "Hello/Jared".

→ We can also use vectors.

Paste(c("Hello", "Hey", "Howdy"), c("Jared", "Bob", "David"))

[1] "Hello Jared" "Hey Bob" "Howdy David"

→ paste("Hello", c("Jared", "Bob", "David"))

[1] "Hello Jared" "Hello Bob" "Hello David".

→ paste("Hello", c("Jared", "Bob", "David"), c("Goodbye", "Seeya"))

[1] "Hello Jared Goodbye", "Hello Bob Seeya", "Hello David Goodbye".

→ vectorofText <- c("Hello", "Everyone")

paste(vectorofText, collapse = " ")

(1)

sprintf:

While paste is convient for static string. for dynamic strings we use sprintf() function.

→ sprintf("Hello %s, Your party of %s will be seated in %s minutes", person, partysize, waitTime)

Note: %s Way replaced with its corresponding variable.

Extracting Text:

→ Often text needs to be ripped apart to be made useful, and while R has a no.of

→ readHTMLTable() to parse the table.

Ex: load("data/presidents.rdata")

theURL <- "http://

→ require (stringr)

> yearlist <- str_split (string = presidents$YEAR, pattern = "-")

> head (yearlist)

combin them into one matrix

> yearmatrix <- data.frame (Reduce (rbind, yearlist))

> head (yearMatrix)

give the columns good names

names (yearmatrix) <- c ("start", "stop")

bind the new columns onto the data.frame

Presidents <- cbind (presidents, yearmatrix)

convert the start and stop columns into numeric.

Presidents$start <- as.numeric (as.character (presidents$start))

Presidents$stop <- as.numeric (as.character (presidents$stop))

Syntax:

str_split (string, Pattern = n = Inf, simplify = True

Str-extract Extract matching pattern from a string.

Str-count count the no. of matches in a string.

Str-ends Detects the presence or absence of pattern at the begining or end of a string.

→ str_sub = (presidents$president, start = 1, end = 3)
get the first 3 characters.

str_sub(string, start = 12, end = -12)

→ presidents[str_sub(string = presidents$start, start = 4,
end = 4) == 1, c("year", "president", "start", "stop")]

Regular Expression

# returns TRUE/FALSE if john was found in the name.
. <- str_detect(presidents$president, pattern = "john")

→ warTimes[str_detect(string = warTimes, Pattern = "-")]

→ which(str_detect(string = warTimes, Pattern = "-"))

← TheStart <- sapply(theTimes, FUN = function(x) x[1])
> head(theStart)

just return elements where January was detected.
theStart[str_detect(string = theStat, Pattern = "January")]

# \d is a shortest for "[0-9]"
head(str_extract(string = theStart, "\d{4}"), 20)

→ # extract 4 digits at the beginning of the text.
head(str_extract(theStart, Pattern = "^\d{4}"), 20)
extract 4 digits at the beginning & ending of the text.
" (theStart, Pattern = "^\d{4}$"), 20)

↔ replace the ~~first~~ all digits seen with X.
head(str_replace_all(string = theStart, Pattern = "\d{4}"),

20)

Write an Aspx to implement Grid View?

LINEAR MODELS

In its simplest form regression is used to determine the relationship b/w two variables. That is given one variable

→ That is given one variable, it tells us what we can expect from the outer variable.

→ This powerful tool, which is frequently taught and can accomplish a great deal of analysis with minimal effort, is called simple linear regression.

→ The outcome variable (what we are trying to predict) is called the "response" and the input variable (what we are using to predict) is the "predictor."

→ predictor to come up with some average value of the response.

$$y = a+bx+c$$

. y varies with x

→ The aim of linear regression is to find a mathematical equation for a continuous response variable y as a function of one or more.

```
ssy = |beta-[1] +\beta_[2]
```

→ linearmod <- lm(dist~speed, data = cars)

Print (linearmod)

Summary (linearmod)

Partitioning Algorithms

Partitioning method: construct a partition of a database D of n objects into a set of K clusters, s.t., min, min sum of squared distance.

$$\sum_{m=1}^{K} \sum_{t_{mi} \in K_m} (c_m - t_{mi})^2$$

→ Given a K_1 find a partition

→ K-Means clustering Method:

→ Given k, the k-Means algorithm is implemented in 4 Steps:

→ partition objects into k nonempty subsets.

→ compute seed points as the centroids of the clusters of the current partition (the center is the centeri.e, mean point, of the cluster)

→ Assign each object

A1 A2 A3 A4 A5 A6 A7 A8

0 √25 √56 √19 √ √ √ √

Partitioning Algorithms: Basic concept:

Partitioning a database D of n objects into a set of k clusters, such that the sum of squared distance is minimized (where c_i is the centroid or medoid of cluster C_i)

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} (p - c_i)^2$$

→ Given k, find a partition of k clusters that optimizes the chosen partitioning criterion.

K-means

→ Given k, the k-means Algorithm is implemented in four steps.

- partition objects into k nonempty subsets.
- Compute seed points as the centroids of the clusters of the current partitioning (the centroid is the center i;e mean point, of the cluster)
- Assign each point object to the cluster with the nearest seed point.
- Go back to step 2, stop when the assignment does not change.

dis

- The k-means algorithm is sensitive to outliers.

  Since an object with an extremely large value may substantially distort the distribution of the data.

K-Medoids: (PAM)

↓

Partition around ~~Mean~~ Medoids

→ Instead of taking the mean value of the object in a cluster as a reference point, medoids can be used, which is most centrally located object in a cluster.

Point	x-axis	y-axis
1	7	6
2	2	6
3	3	8
4	8	5
5	7	4
6	4	7
7	6	2
8	7	3
9	6	4
10	3	4

Let us choose (3,4) and (7,4) are the medoids.

$$d = \sqrt{(x_2-x_1)^2+(y_2-y_1)^2}$$

(3,4) (7,6)
x_1, y_1 x_2, y_2

$= \sqrt{(7-3)^2+(6-4)^2}$

$= \sqrt{(4)^2+(2)^2}$

$= \sqrt{16+4} = \sqrt{20}$

(7,4), (7,6)

$= \sqrt{(7-7)^2+(6-4)^2}$

$= \sqrt{0+4} = \sqrt{4} = 2$ ✓

(3,4) (2,6)

$= \sqrt{(2-3)^2+(6-4)^2}$

$= \sqrt{(1)^2+(2)^2}$

$= \sqrt{5}$ ✓

(7,4), (2,6)

$= \sqrt{(2-7)^2+(6-4)^2}$

$= \sqrt{(5)^2+(2)^2}$

$= \sqrt{25+4} = \sqrt{29}$

(3,4) (3,8)

$\sqrt{(3-3)^2+(8-4)^2}$

(7,4), (3,8)

$\sqrt{(3-7)^2+(8-4)^2}$

$\sqrt{0+16} = 4$ ✓

$(3,4), (8,5)$

$\sqrt{(8-3)^2 + (5-4)^2}$

$= \sqrt{(5)^2 + (1)^2} = \sqrt{26}$

$(3,4), (7,4)$

$= \sqrt{(7-3)^2 + (0)^2}$

$= \sqrt{(4)^2} = 4$

$(3,4), (4,7)$

$= \sqrt{(1)^2 + (3)^2}$

$= \sqrt{10}$ ✓

$(3,4), (6,2)$

$\sqrt{(3)^2 + (2)^2}$

$= \sqrt{9+4} = \sqrt{13}$

$(3,4), (7,3)$

$\sqrt{(4)^2 + (1)^2}$

$= \sqrt{16+1} = \sqrt{17}$

$(3,4), (6,4)$

$\sqrt{(3)^2 + 0} = \sqrt{9} = 3$

$(3,4), (3,4)$

$= 0$ ✓

$\sqrt{16+16} = \sqrt{32}$

$(7,4), (8,5)$

$\sqrt{(8-7)^2 + (5-4)^2}$

$= \sqrt{1+1} = \sqrt{2}$ ✓

$(7,4), (7,4)$

$\sqrt{(7-7)^2 + (4-4)^2}$

$= 0$ ✓

$(7,4), (4,7)$

$\sqrt{(3)^2 + (3)^2}$

$= \sqrt{9+9} = \sqrt{18}$

$(7,4), (6,2)$

$\sqrt{(1)^2 + (2)^2}$

$= \sqrt{5}$ ✓

$(7,4), (7,3)$

$\sqrt{(0)^2 + (1)^2}$

$= \sqrt{1} = 1$ ✓

$(7,4), (6,4)$

$\sqrt{(1)^2 + 0} = \sqrt{1} = 1$ ✓

$(7,4), (3,4)$

$\sqrt{(4)^2 + 0} = 4$

Cluster1 (3,4)

(2,6), (3,8), (4,7)

(3,4)

Cluster2 (7,4)

(7,6) (8,5), (7,4), (6,2)

(7,3), (6,4)

K-Mean

→ based on Mean

K- Medoids

→ based on Median.

K-Medoids

Algorithm.

1. Initially select k random points as the medoids from the given n data points of the dataset.
2. Associate each data point to the closest medoid by using any of the most common distance metrics.
3. for each pair of non-selected object h and selected object i, calculate the total swapping cost TC_{ih}

If $TC_{ih} < 0$ i is replaced by h.

→ Now calculating the cost which is nothing but the sum of distance of each non-selected point from the selected point.

Total cost = cost((3,4),(2,6)) + cost((3,4),(3,8)) + cost((3,4),(4,7)) + cost((7,3),(7,6)) + cost((7,3),(8,5)) + cost((7,3),(6,2)) + cost((7,3),(7,4)) + cost((7,3),(6,4))

= 3+4+4+3+3+2+1+2 = 22.

(7,3), (7,6)
$\sqrt{0+(3)^2} = 3$

(7,3), (2,6)
$\sqrt{(5)^2+(3)^2}$

(8,5) (7,6)
$\sqrt{(1)^2+(1)^2} = \sqrt{2}$

(8,5) (2,6)
$= \sqrt{(6)^2+(1)^2} = \sqrt{37}$

(8,5), (3,8)
$\sqrt{(5)^2+(3)^2} = \sqrt{34}$

(8,5), (8,1)
$= 0$

(8,5), (7,4)
$\sqrt{(1)^2+(1)^2} = \sqrt{2}$

(8,5), (4,7)
$= \sqrt{(4)^2+(2)^2} = \sqrt{20}$

(8,5), (6,3)
$\sqrt{(2)^2+(3)^2} = \sqrt{13}$

(8,5), (7,3)
$\sqrt{(1)^2+(2)^2} = \sqrt{5}$

(8,5) (6,4)
$= \sqrt{(2)^2+(1)^2} = \sqrt{5}$

(8,5), (3,4)
$\sqrt{(5)^2+(1)^2} = \sqrt{26}$

cluster1 (3,4)

(7,6), (2,6), (3,8)

(4,7), (6,2)

cluster2

(8,5), (7,4), (7,3) (7,6)

(6,4)

(3,4) (3,8) $|x_2-x_1|+|y_2-y_1|$

$|0|+|4| = 4$ ✓

(3,8) (8,5)

$|5|+|3| = 8$

(3,4) (7,6)

$|4|+|2| = 6$

(8,5) (7,6)

$|1|+|1| = 2$ ✓

(3,4) (2,6)

$|1|+|2| = 3$ ✓

(2,6) (8,5)

$|6|+|1| = 7$

(3,4), (8,5)

$|5|+|1| = 6$

(8,5), (8,5)

$= 0$ ✓

(3,4) (7,4)

$|4| = 4$

(8,5), (7,4)

$= 1+1 = 2$ ✓

(3,4), (4,7)

|1|+|3|=4.

(3,4), (6,2)

3+2=5

(3,4), (7,3)

3+1=4

(3,4), (6,4)

=3

(3,4) (3,4)

=0

(8,5) (4,7)

|4|+|2|=6

(8,5), (6,2)

2+3=5

(8,5), (7,3)

1+2=3.

(8,5), (6,4)

2+1=3

(8,5), (3,4)

= 5+1=6

(2,6), (4,7)

Total cost = 3+4+4+2+5+2+3+3 = 26

total cost (7,4) < total cost (8,5)

20 < 26

∴ (7,4) is the medoid.

(6,4) (3,4)

(6,4), (7,6)

= 1+2=3

(8,5), (6,4)

2+1=3

(6,2), (6,4)

=0+2=2

(2,6) (6,4)

4+2=6

(7,4), (6,4)

=1+0=1

(7,3) (6,4)

1+1=2

(3,8) (6,4)

3+4=7.

(4,7), (6,4)

2+3=5.

(3,4) (6,4)

=3

cluster1 (3,4)
(3,6), (3,8), (4,7)

(6,4)
(4,6), (6,5), (5,4)
(6,10) (9,3)

Total cost = 3+4+4+3+3+1+2+2 = 22

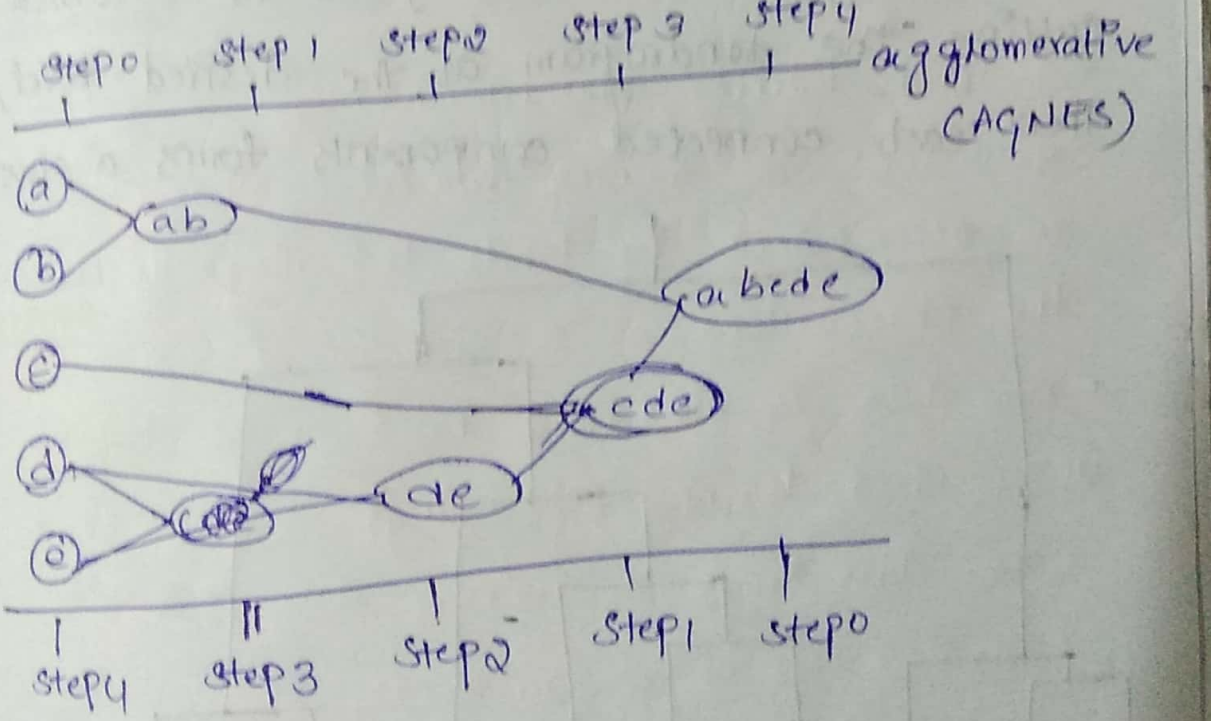
→ k-Medoids

xx <- cluster::pam(points, 2)

	ID	x	y
	5	7	4
	2	2	6

Hierarchical clustering:

- use distance matrix as clustering criteria. This method does not require the no. of clusters k as an input, but needs a termination condition.



AGNES

→ Introduced in Kaufmann and Rousseeuw

→ Implemented in Statistical packages e.g Splus

→ Use the single-link method and the dissimilarity matrix

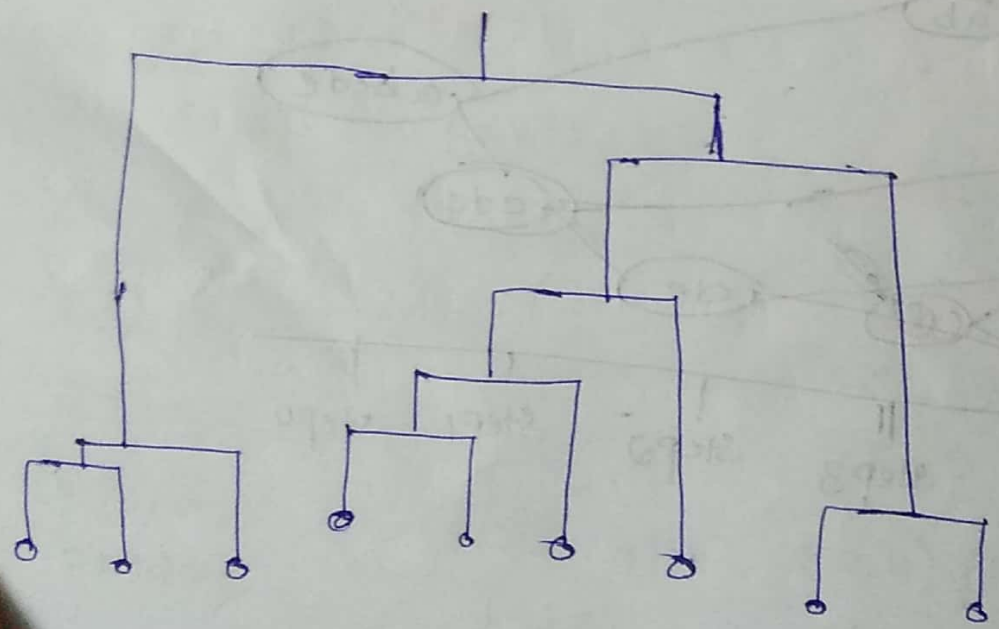
→ Merge nodes that have the least dissimilarity

→ Go in non-descending fashion

Dendrogram shows how clusters are Merge

Decompose data objects into a several levels of nested partitioning (tree of clusters), called a dendrogram.

→ A clustering of the data objects is obtained by cutting the dendrogram at the desired level, then each connected components forms a cluster



Algorithm

→ put each data point in its own cluster.

→ identify the ~~closest~~ closest

Rcode:

```
clusters <- hclust(dist(iris[, 3:4]))
plot(clusters)
```

↓ method for hierarchical clustering.

used to create distance matrix.

To cut the dendrogram

```
clusterCut <- cutree(clusters, 3)
> table(clusterCut, iris$Species)
```

out:

clusterCut	setosa	versicolor	virginica
1	50	0	0
2	0	21	50
3	0	29	0

DIANA

Classification:

Classification:

→ predicts categorical class labels (discrete or nominal)

→ classifies data (constructs a model) based on the training set and the values (class labels) in a classifying attribute and uses it in classifying new data.

Prediction:

- models continuous-valued functions, i.e. predicts unknown or missing values.

Typical applications

- credit approval
- Target marketing
- Medical diagnosis
- fraud detection.

Supervised learning (classification)

Supervision: The training data (observations, measurements) are accompanied by labels indicating the class of the observations.

- New data is classified based on the training set.

Unsupervised learning (clustering)

- The class labels of training data is unknown.
- Given a set of measurements, observations etc with the aim of establishing the existence of classes or clusters in the data.

Issues: (classification)

1. Data cleaning

   Preprocess data in order to reduce noise and handle missing values.

2. Relevance analysis (feature selection)

   Remove the irrelevant or redundant attributes.

3. Data transformation

Decision Tree

Algorithm:

Generate, decision tree.

Generate a decision tree from the training tuples of data partition D.

Input: