Statistical Computing with R: Masters in Data Sciences 503 (S18) Third Batch, SMS, TU, 2024

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.

Unit 4: Review Preview 1

- Average
  - Geometric mean
  - Harmonic mean
- Moments
  - First order
  - Second order
  - Third order
  - Fourth order
- Normal distribution
  - Skewness
  - Kurtosis
- Relative measures
  - Quintiles
  - Percentiles
  - Standard or z-score
- Breakdown analysis: “apply”

Measures of central tendencies

- Computing measure of central tendency, dispersion, moments and relations position in R using packages and functions/scripts

- Measures of Central Tendency – mean, median, mode, geometric mean, harmonic mean
- Measure of Dispersion – standard deviation, inter-quartile range, range
- Moments – mean, standard deviation, skewness, kurtosis
- Relative position – percentile, quartiles and z-score

Geometric mean in R:

https://www.r-bloggers.com/2021/08/calculate-geometric-mean-in-r/

- **GM=(x_1, x_2, x_3, ……, x_n)$^{1/n}$**
- **exp(mean(log(x)))**
- data <- c(1, 15, 12, 5, 18, 11, 12, 15, 18, 25)
- exp(mean(log(data)))
- 10.37383 (Interpretation?)

- `data <- c(1, 15, 12, 5, 0, 18, 11, 12, 15, 18, 25, 0, -11)`
- exp(mean(log(data[data>0])))
- 10.37383
- **Interpretation?**
- **Better for summarizing simple rates, ratios and proportions!**

Harmonic mean in R: https://www.geeksforgeeks.org/harmonic-mean-in-r/

- Harmonic mean (H) is defined as:

$$H = \frac{n}{\frac{1}{x_1}+\frac{1}{x_2}+\frac{1}{x_3}+\ldots+\frac{1}{x_n}}$$

- It is available in “psych” package
- Syntax: *harmonic.mean(x)*

Harmonic mean in R: https://www.geeksforgeeks.org/harmonic-mean-in-r/

load the library
- library("psych")

create dataframe
- data=data.frame(col1=c(12,2,3,4),
- col2=c(34,32,1,0),
- col3=c(2,45,3,2))

display
- print(data)

harmonic mean of column1
- print(data$col1)

harmonic mean of column2
- print(data$col2)

harmonic mean of column3
- print(data$col3)

- **Interpretation?**

- **Better for summarizing instantaneous rates!**

Moments in R: ChatGPT plug-in for Google Chrome

- To calculate moments in R, you can use the moments package, which provides functions for various statistical moments.

- Here's an example of calculating the first four moments (mean, variance, skewness, and kurtosis) for a numeric vector x:

# Load the 'moments' package
library(moments)

Calculate the moments of x <- c(1, 2, 3, 4, 5)

- mean_x <- mean(x) #First moment
- var_x <- var(x) #Second moment
- skewness_x <- skewness(x) #Third
- kurtosis_x <- kurtosis(x) #Fourth
- **Interpretations?**

Third and fourth moments

https://www.itl.nist.gov/div898/handbook/eda/section3/eda35b.htm

- Skewness – symmetricity
- Pearson’s coefficient of skewness
- Bowley’s coefficient of skewness
- Coefficient = 0 = Symmetrical

- Kurtosis – peakedness
- Pearson’s coefficient of kurtosis
- Coefficient of excess kurtosis
- 0 = Mesokurtic for excess kurtosis
- <0 = Platykurtic for excess kurtosis
- >0 = Letptokurtic for excess kurtosis

Percentile & quintiles in R: ChatGPT plug-in for Google Chrome

- In R, you can calculate percentiles or quartiles or quntiles using the quantile() function.
- The function takes two arguments: the data vector and the desired percentile.
- Here's an example:

Create a sample vector of data

- data <- c(12, 5, 9, 17, 3, 8, 10)

Calculate the 75th percentile

- percentile_75 <- quantile(data, 0.75)

Print the result

- print(percentile_75)

- The output will be the 75th percentile of the data vector, in this case, 11.5.

Standard score or z-score in R: ChatGPT plug-in for Google Chrome

- In R, you can calculate the z-score using the scale() function. The scale() function standardizes a numeric vector by subtracting the mean and dividing by the standard deviation.

- # Create a vector of numeric values
- values <- c(10, 15, 12, 8, 20)

- # Calculate the z-score
- z_scores <- scale(values)

- # Print the z-scores
- print(z_scores)

- This will give you the standardized z-scores for each value in the vector.

Breakdown analysis with “apply” function

https://www.geeksforgeeks.org/apply-lapply-sapply-and-tapply-in-r/

- **tapply() function**

- The tapply() helps us to compute **statistical measures** (mean, median, min, max, etc..) or a self-written function operation **for each factor variable in a vector**.

- It helps us to create a subset of a vector and then apply some functions to each of the subsets.

- # load library tidyverse
- library(tidyverse)
-
- # print head of diamonds dataset
- head(diamonds)
-
- # apply tapply function to get average price by cut
- tapply(diamonds$price, diamonds$cut, mean)

Question/queries so far?

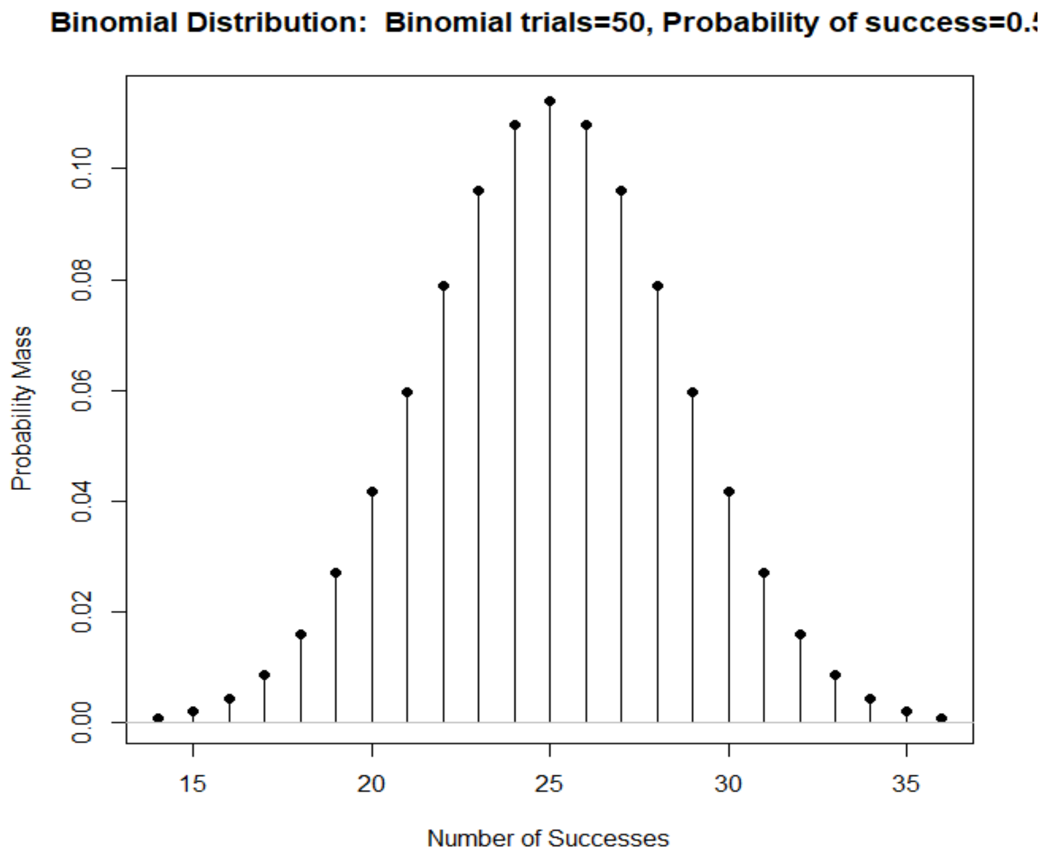
Unit 4: Review Preview 2

- Probability distribution functions
  - Discrete
  - Continuous
- Demo with selected distributions
- Normal approximations of binomial distribution
- Test of normality
  - Graphical
  - Test

Discrete probability distribution:

- **Binomial**
- Poisson
- Geometric
- Hypergeometric
- Negative binomial etc.

- Binomial distribution is used heavily in the classification models of supervised learning!



Discrete probability distribution: Binomial
(Check with p=0.3 and p=0.7 vs p=0.5 below!)

#Number of trials

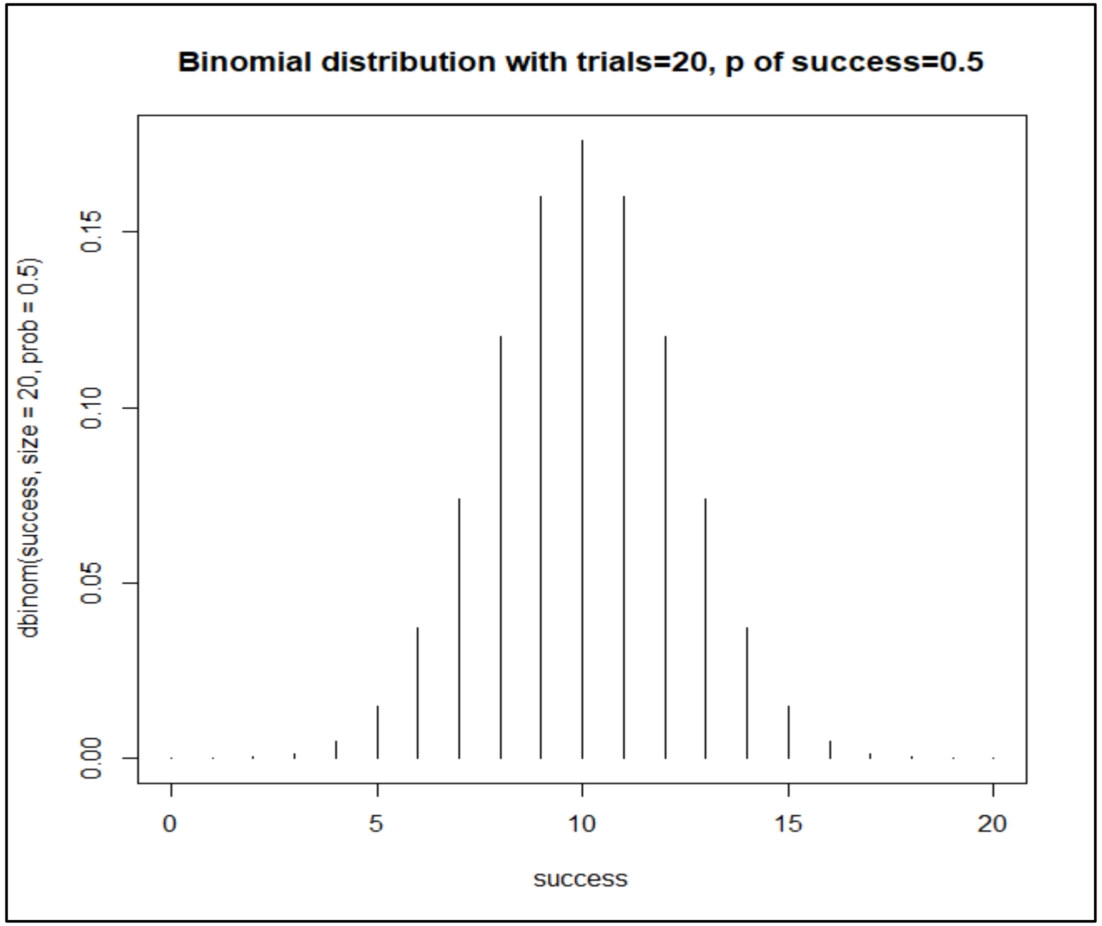
success <- 0:20

Binomial Probability distribution with success probability of 0.5

dbinom(success, size=20, prob=0.5)

#Plot

plot(success, dbinom(success, size=20, prob=0.5), type="h“, main = Binomial distribution with n=20 and p of success=0.5”)



Let's get/check the data of success and binomial probabilities (Do this in excel):

binomc <- cbind(success, binomd)

binomc (Results are on the right side >>>)

How was the "binomd" values created?

$$P(X=x)=\binom{n}{x}p^x(1-p)^{n-x} \quad ;x=0,1,2,\dots,n.$$

$$where\ \binom{n}{x}=\frac{n!}{x!\ *\ (n-x)!}$$

$$n!=n*(n-1)*\dots*3*2*1$$

Prove that: sum of "binomd" = 1 in R and Excel! Why is this important?

	success	binomd
[1,]	0	0.000000953674
[2,]	1	0.000019073486
[3,]	2	0.000181198120
[4,]	3	0.001087188720
[5,]	4	0.004620552063
[6,]	5	0.014785766602
[7,]	6	0.036964416504
[8,]	7	0.073928833008
[9,]	8	0.120134353638
[10,]	9	0.160179138184
[11,]	10	0.176197052002
[12,]	11	0.160179138184
[13,]	12	0.120134353638
[14,]	13	0.073928833008
[15,]	14	0.036964416504
[16,]	15	0.014785766602
[17,]	16	0.004620552063

What is normal approximation of binomial distribution? When to use it??

- Is it related to the sample size of the successes and failures?
- Earlier when n*p >5 and n*q>5, it was considered that it will approximate the normal distribution
- Now, it is set at n*p>10 and n*q>10!

- Which regression model is used when we need to use normal distribution for dichotomous or binary dummy dependent variable (Yes = 1 and No = 0)
- Logistic regression model because log transformation was used to convert the exponential equation form to make it linear model!

Q2: When and how to use?

- Poisson distribution?
- Zero-inflated Poisson distribution. When to use?
- Negative binomial distribution. When to use?
- Hypergeometric distribution?
  - Fisher's exact test??

Continuous probability distributions:

- Normal
- T
- Chi-square
- F
- Exponential
- Logistic etc.

Normal Distribution Formula

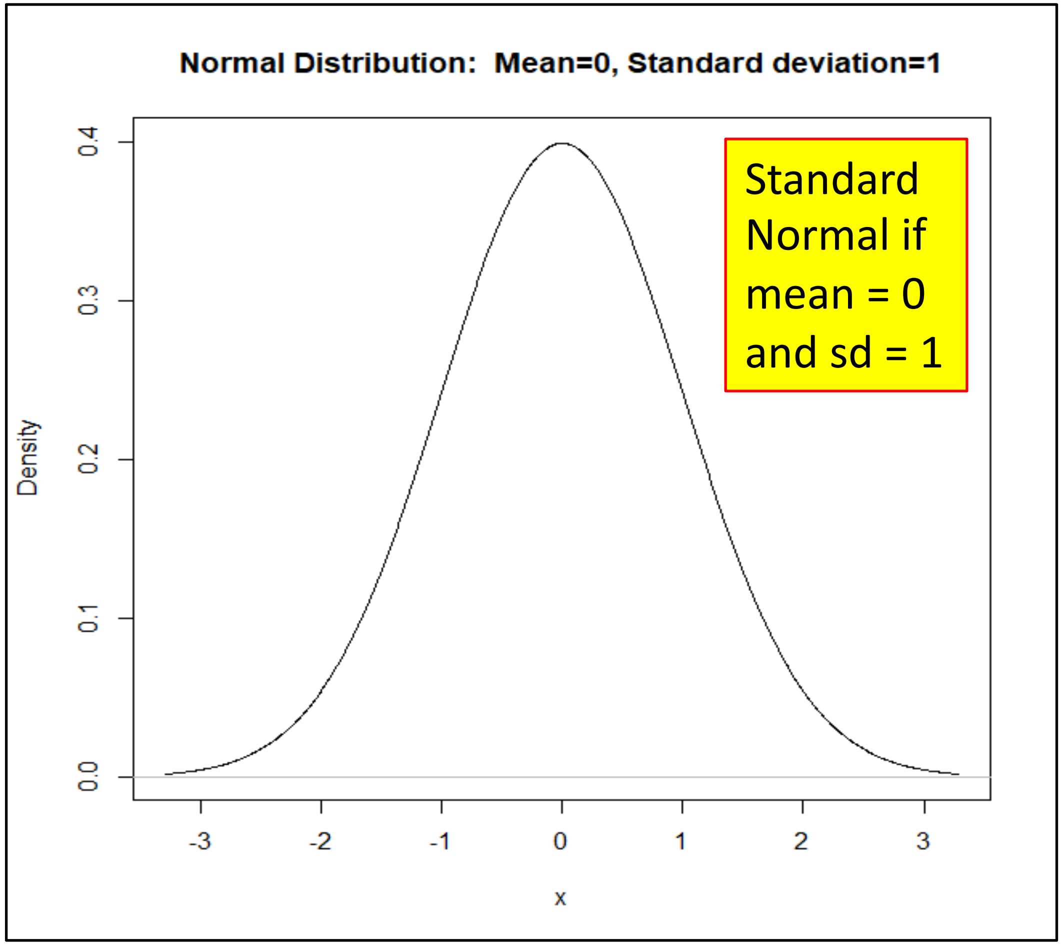
$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$$

$\mu =$ mean of x

$\sigma =$ standard deviation of x

$\pi \approx$ 3.14159 ...

$e \approx$ 2.71828 ...



- Normal/Standard Normal Distribution is used in the linear and general linear regression models of supervised learning!

Normal Distribution of values between -4 and +4 with pre-defined population mean and sd:

#Define mean and SD

pop_mean <- 50

pop_sd <- 5

#Define lower and upper limits

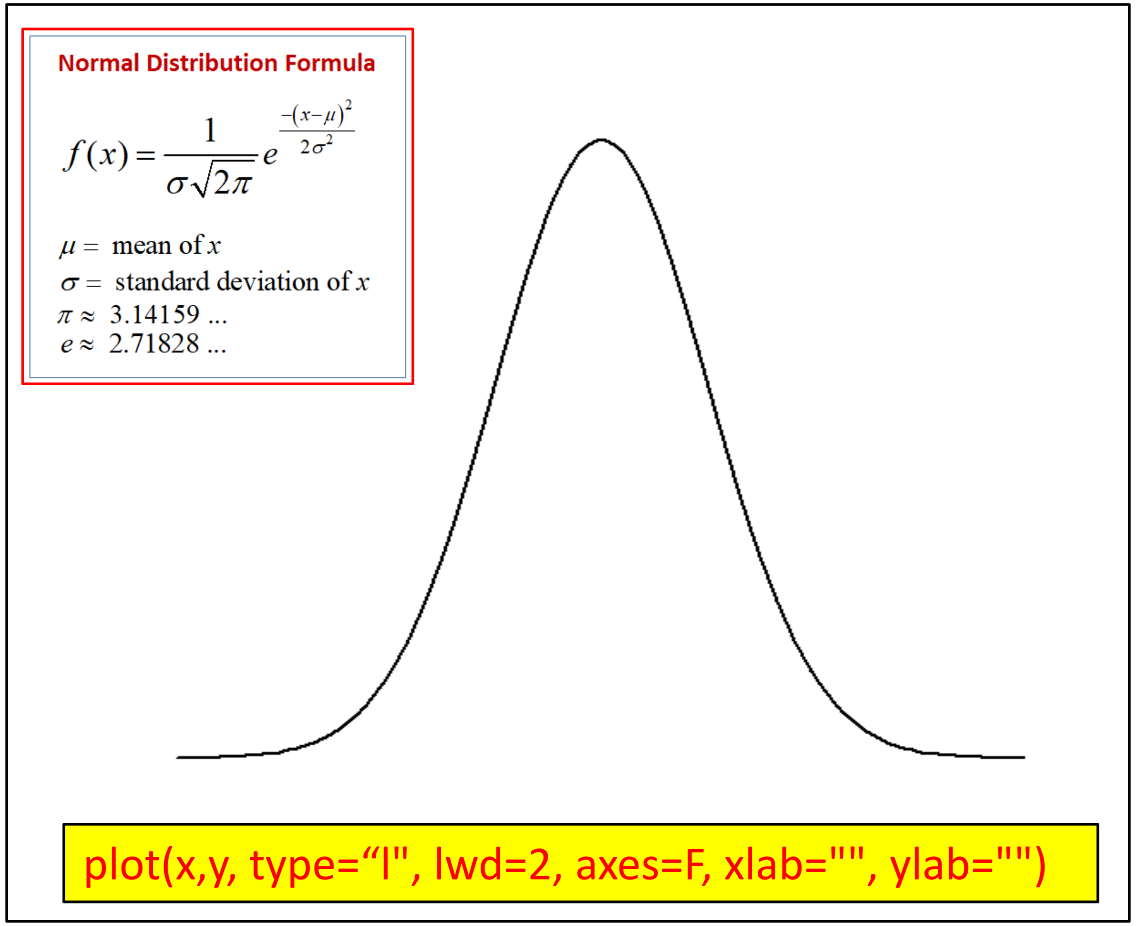
LL <- pop_mean – pop_sd

UL <- pop_mean + pop_sd

#Create a sequence of 100 x values based on pop mean and sd

x <- seq(**-4,4**, length=100)*pop_sd+pop_mean

y <- dnorm(x, pop_mean, pop_sd)



Adding x-axis values and mean in the curve:

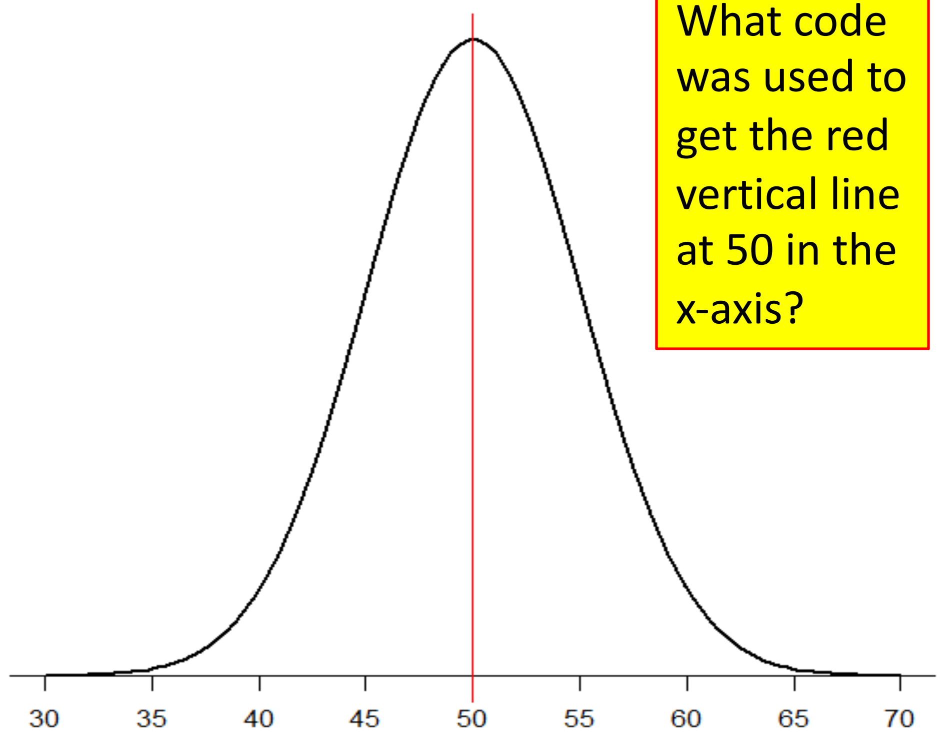
```
plot(x,y, type=“l", lwd=2, axes=F, xlab="", ylab="")
```

```
sd_axis_bounds = 5
```

```
axis_bounds <- seq(-sd_axis_bounds*pop_sd + pop_mean, sd_axis_bounds*pop_sd + pop_mean, by=pop_sd)
```

```
axis(side=1, at=axis_bounds, pos=0)
```

```
abline(??)
```



What code was used to get the red vertical line at 50 in the x-axis?

Class work/Assignment 1:

- Get this graph and **provide annotation in** it as follows:

- 45-55: mean ± 1SD = 67% data
- 40-60: mean ± 2SD = 95% data
- 35-65: mean ± 3SD = 99% data

Note: You can use ggplot2 package, if required!

Why normal distribution is important?

- When continuous variable follows the theoretical normal distribution then we **must** summarize that variable using mean and standard deviation

- We can also use t-test and 1-way ANOVA to compare means across two or more categories of categorical variables respectively

- When continuous variable do not follow the theoretical normal distribution then we **must** summarize that variable using median and inter-quartile range

- We can only use median test to compare median across two or more categories of the categorical variables

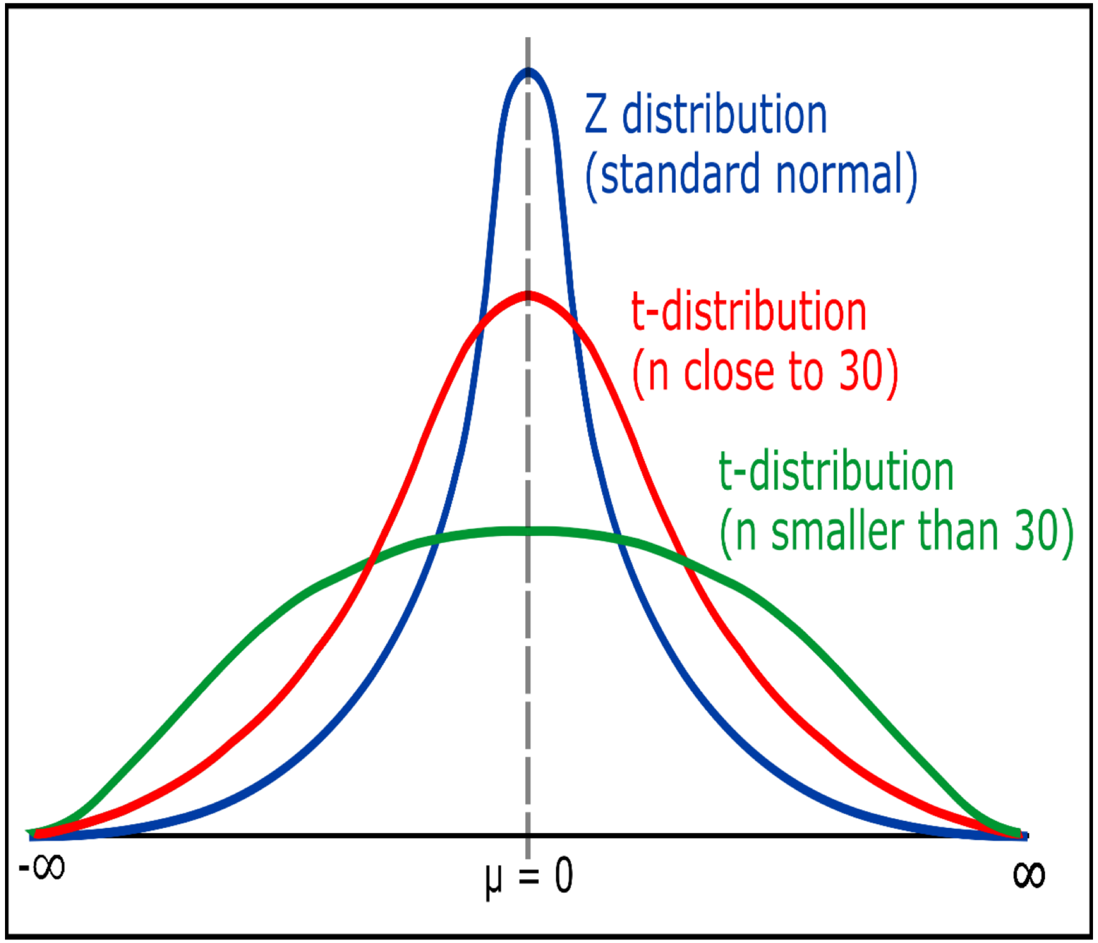
Q3: Why these test must not be used?

- Mann-Whitney U test **must not be used** to compare medians across two categories of a categorical variable?
- e.g. comparing age by sex as sex variable normally has two categories “male” and “female” if age is not normally distributed

- Kruskal-Wallis W test **must not be used** to compare medians across two categories of a categorical variable?
- e.g. comparing age by socio-economic status (SES) variable as SES has 3 categories (low, middle, high) or 5 categories (lowest, low, middle, high, highest) if age is normal!

T and Z distributions:

- T distribution is normally used when there is small sample size, say, random samples < 30
- As the sample size increases t-distribution behaves like normal distribution so we can use it for large samples too!
- **Linear regression is extension of t-test and 1-way ANOVA!**

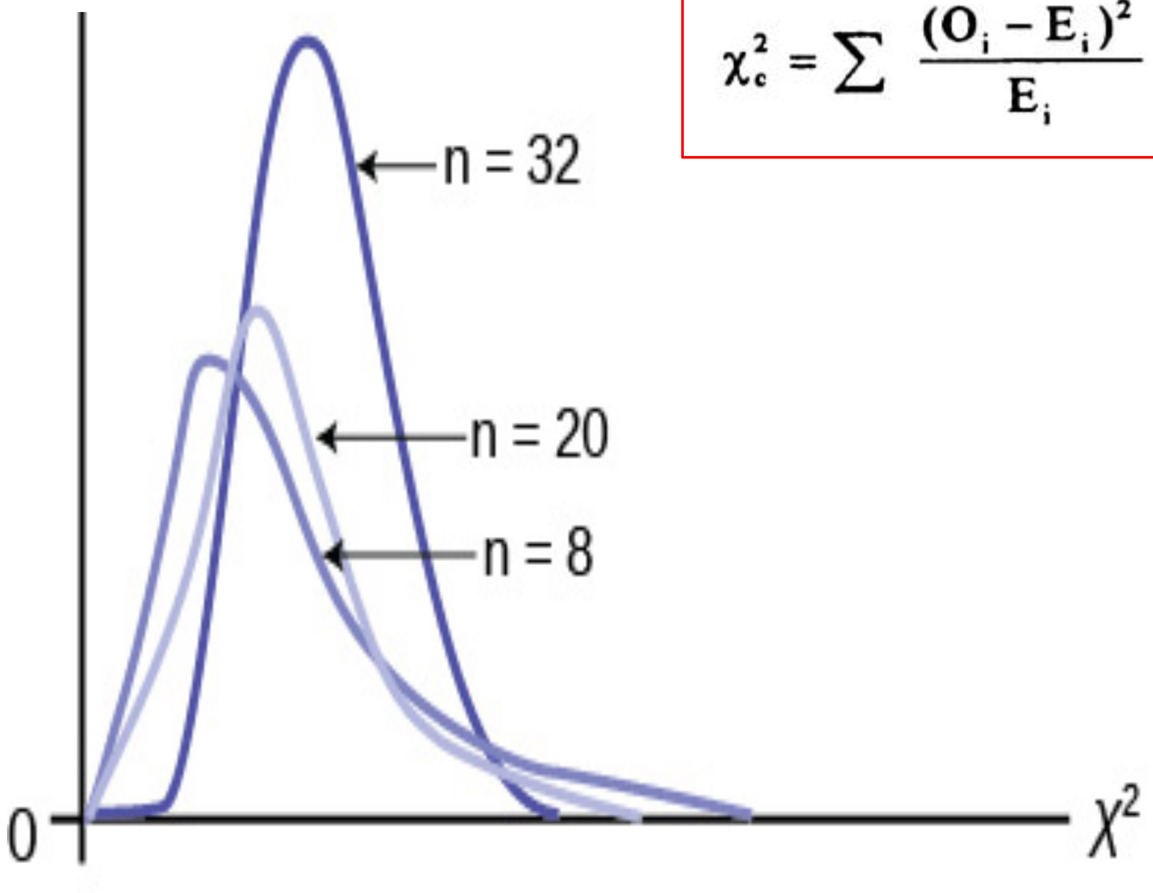


Chi-square and Z distributions:

- Chi-square distribution is normally used in contingency tables or cross-tabulations to find "association" between dependent and independent variable categories. It is also used for goodness-of-test and comparing proportions across categories!

- As the sample size increases chi-square distribution also behaves like normal distribution

- **Logistic regression is extension of <u>chi-square test</u>!**



$$\boxed{\chi_c^2 = \sum \frac{(O_i - E_i)^2}{E_i}}$$

Q4: Why?

- Logistic regression is described as the extension of the Pearson’s chi-square test?
- Both are used to get/test the association between two (or more variables)
- p-value<0.05 means association is statistically significant!

- **Prove it with an example!**
- **Hint**: Create a two-by-two table e.g. smoking vs lung cancer
- Get p-value from chi-square test
- Get p-value from bivariate logistic regression
- Are they same? If yes then good!

Test of normality: Key point of this lecture! (Goodness-Of-Fit with Chi-square variants):

- This is a goodness-of-fit test for comparing data against the normal distribution

- Most widely used tests are:
  - Jarque-Bera test
  - **Kolmogorov-Smirnov test (large samples i.e. n>100)**
  - **Shapiro-Wilk test (Small samples)**
  - Anderson-Darlington test etc.

- Test of normality is assessed:

- Graphically (suggestive):
  - Stem-leaf plot
  - Histogram
  - Q-Q plot

- Test (confirmative):
  - ?? (depends on sample size!)

Goodness-of-fit test for normal distribution

- H0: Data follows the normal distribution ($p>0.05$)
- H1: Data does not follow the normal distribution ($p<=0.05$)
- Here we want to accept the null hypothesis!
- Normally, we want to accept alternative hypothesis (H1) but while performing any goodness-of-fit test we need to accept the null hypothesis (H0)
- This applies to goodness-of-fit test for equality of variance as well (we will discuss it in the next class!)

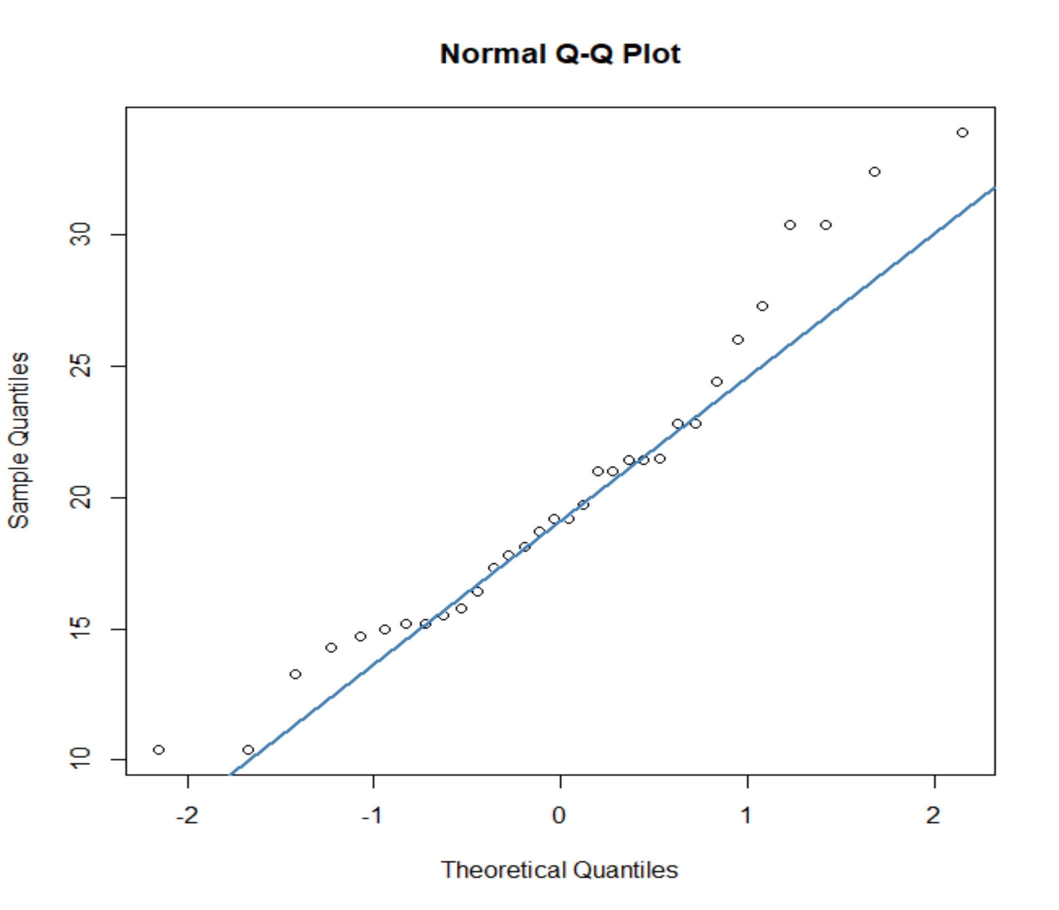
Assignment 2: Statistical tests are “robust”!

- Get stem-leaf plot, histogram and normal q-q plot of **mpg variable** of the “mtcars” data
- Test the normality of mpg variable of mtcars data using shapiro wilk test (**Why this test?**)
- shapiro.test(data)

Shapiro-Wilk normality test

data: mtcars$mpg

W = 0.94756, p-value = 0.1229



H_0: Data follows normal distribution ($p>0.05$)
H_1: Data do not follow normal distribution ($p<=0.05$)

H_0: No difference between data and normal distribution
H_1: Difference between data and normal distribution

Question/queries so far?

- Next class:

- Hypothesis testing with:

- Z-test
- T-test
- Proportion test …

Thank you!

@shitalbhandary