Introduction to R

Audrey Webb & Charles Li
University of California, Berkeley

July 24, 2018
Sichuan Center for Disease Control and Prevention
Institute of Public Health Informatics

Agenda

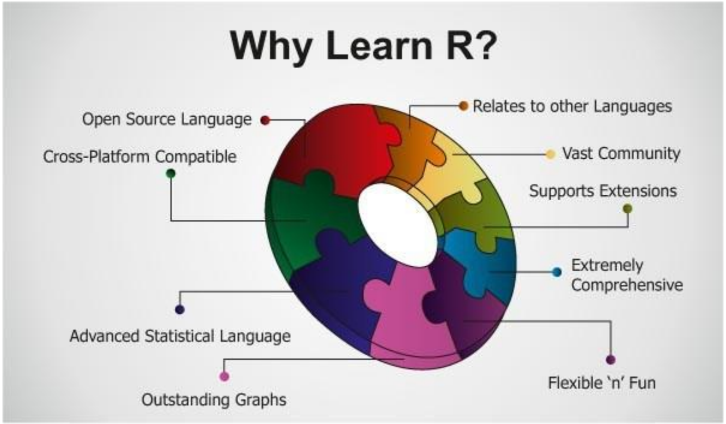
1. Overview & Setup
2. Data Types & Data Structures
3. Tools for Data Wrangling & Exploratory Data Analysis
4. Conducting Statistical Tests
5. Discussion, Feedback, & Further Resources

1. Overview & Setup
2. Data Types & Data Structures
3. Tools for Data Wrangling & Exploratory Data Analysis
4. Conducting Statistical Tests
5. Discussion, Feedback, & Further Resources

About R

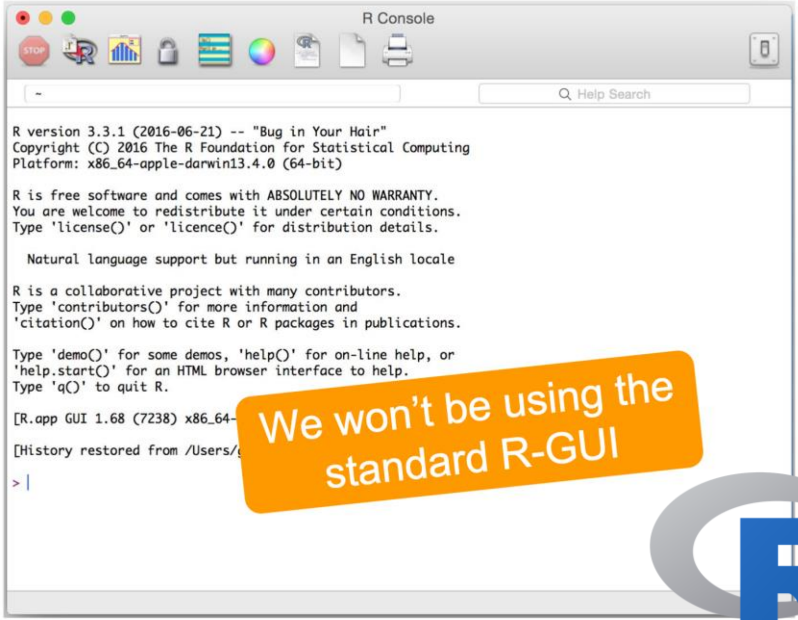
- a **programming language** for interactive data analysis developed in 1996
- enables highly **customizable** analyses that are easy to share and **replicate**
- employs a variety of **free, open-source** numerical and statistical tools for data analysis

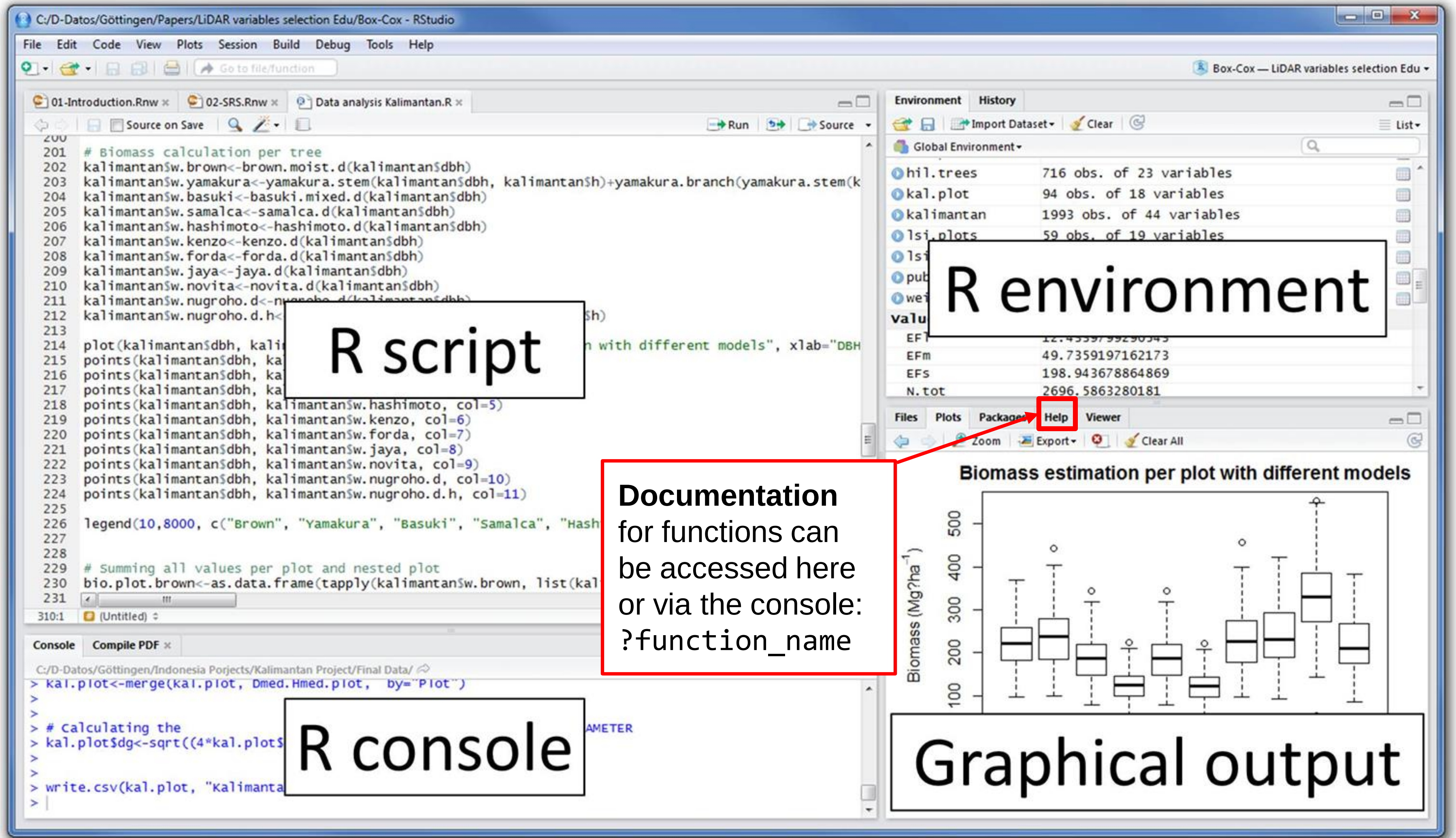
https://www.r-project.org



Using R

- while R comes with its own graphical user interface (GUI), we will be using a more advanced interface called **RStudio**
- in addition to providing a more sophisticated working environment, RStudio enables the creation of [R Markdown files (.rmd)](#) that combine narrative text and code into a single document




C:/D-Datos/Göttingen/Papers/LiDAR variables selection Edu/Box-Cox - RStudio
R script
R environment
Documentation for functions can be accessed here or via the console: ?function_name
R console
Graphical output
Biomass estimation per plot with different models

Packages in R

- the capabilities of R are extended through user-created **packages**, which allow for
  - specialized statistical analyses
  - graphical devices
  - import/export capabilities
  - reporting tools
  - …and many more functions
- in addition to a core set of packages included with R, more packages can be downloaded from **repositories** like
  - the Comprehensive R Archive Network (CRAN)
  - GitHub
  - Bioconductor
  - …and others

Setup Instructions

1. Download and install the current version of **R** from https://cran.rstudio.com/
2. Download and install **RStudio Desktop** from https://www.rstudio.com/products/rstudio/download
3. Explore additional packages of interest based on discipline/application using the CRAN Task Views page: https://cran.r-project.org/web/views/
4. Download and install packages of interest via the **console** in RStudio:
   a. `install.packages("package_name")`
   b. `library("package_name")`

1. Overview & Setup
2. Data Types & Data Structures
3. Tools for Data Wrangling & Exploratory Data Analysis
4. Conducting Statistical Tests
5. Discussion, Feedback, & Further Resources

Data for Programming Languages

Data types are classifications of data that define the operations that can be done on the data, its meaning, and how it can be stored.

Data structures are specialized formats for organizing and storing data that enable its efficient access and modification

Data Types	Data Structures
Basic kinds	*Containers*

Common Data Types in R

- **integer** (whole numbers)
- **double** (real, decimal numbers)
- **logical** (boolean)
- **character** (strings)

```
1L       # integer
2.5      # double (real)
TRUE     # logical
"hello"  # character
```

To check data type: `typeof()`

To convert between data types:

- `as.integer()`
- `as.numeric()`
- `as.logical()`
- `as.character()`

There are some special data values in R

`NULL` = null object

`NA` = Not Available (missing value)

`Inf` = positive infinite

`-Inf` = negative infinite

`NaN` = Not a Number (different from NA)

Data Structures in R

- a **vector**, the most basic data structure in R, is made up of contiguous cells containing values of the same data type

1	2	3	4	5	*numeric*

```
x <- c(1, 2, 3, 4, 5)
```

TRUE	FALSE	TRUE	FALSE	*logical*

```
y <- c(TRUE, FALSE, TRUE, FALSE)
```

"I"	"you"	"we"	"they"	*character*

```
z <- c(“I”, “you”, “we”, “they”)
```

- vectors are commonly subset using a numeric or logical index:

Bracket notation for vectors

`object[index]`

```
y[c(1, 3)]
```

```
TRUE TRUE
```

```
x[x > 1]
```

```
2 3 4 5
```

Data Structures in R

- a **factor** is another data structure designed to handle **categorical data**

small	medium	large

factor

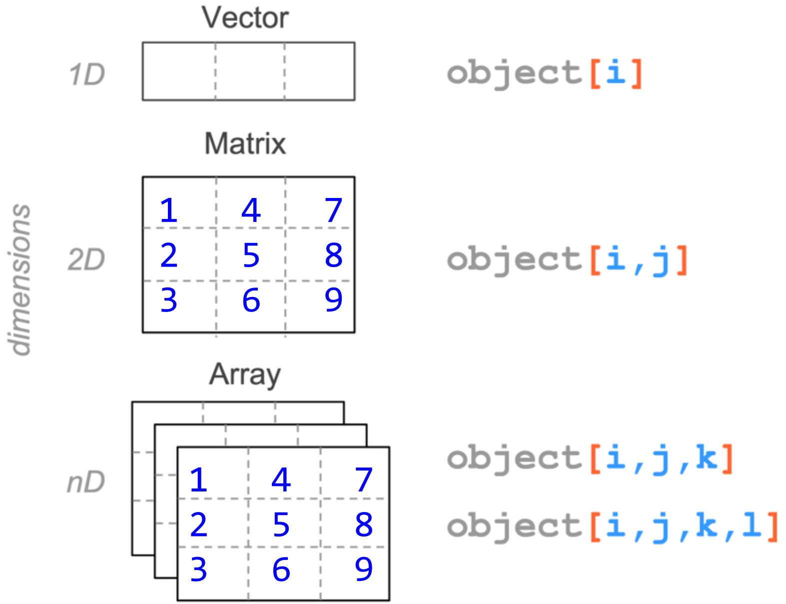
```
size <- c(“small”, “medium”, large”)

size <- factor(size)
```

- factors are internally stored as vectors of integers and behave a lot like vectors but have their own special properties

Example	R mode
`1, 2, 3`	numeric
`"small", "medium", "large"`	character
`small, medium, large`	factor

Data Structures in R

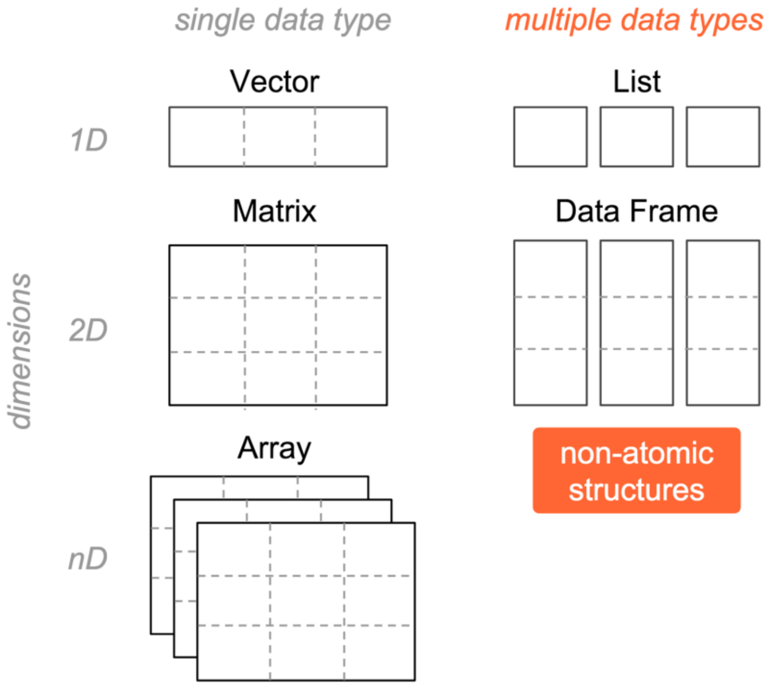
- **matrices** and **arrays** are other data structures that contain values of the same data type (**atomic structures**)




```r
a <- 1:9

A <- matrix(a, nrow = 3, ncol = 3)

b <- 1:27

dim(b) <- c(3, 3, 3)
```

Data Structures in R

- **lists** and **dataframes** are **non-atomic structures** that can store values of more than one data type



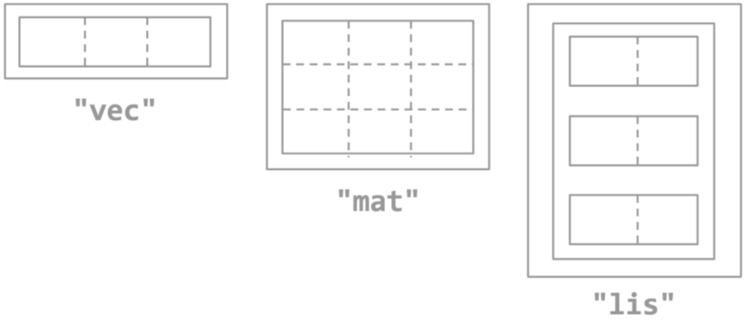
A **list**, R's most general data structure, can contain any other type of data structure (even other lists)

A **dataframe** is the primary data structure R uses for handling tabular datasets. It is treated by R as a special kind of list (stored in R as a list of vectors or a list of factors).

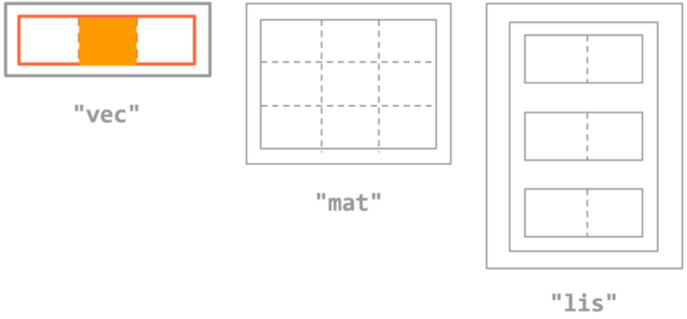
Data Structures in R

- a list can be subset using bracket [ ] or dollar $ notation (if its elements are named)

```
lst <- list(
  vec = c(1, 2, 3),
  mat = matrix(1:9, nrow = 3, ncol = 3),
  lis = list(1:2, c(TRUE, FALSE), c("a", "b"))
)
```



`lst$vec[2]` or `lst[[1]][2]`



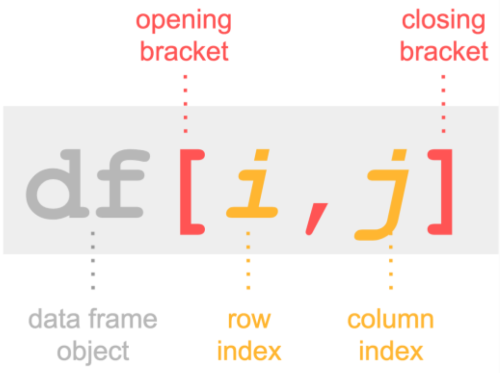
Data Structures in R

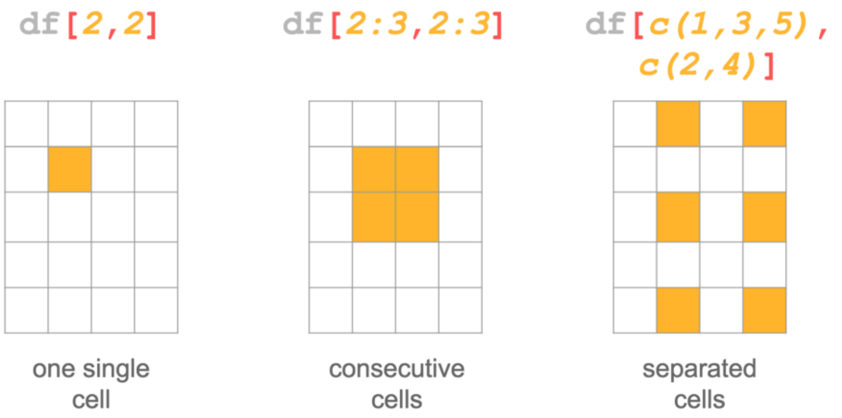
- creating a dataframe:

```r
df <- data.frame(
  name = c('Anakin', 'Padme', 'Luke', 'Leia'),
  gender = c('male', 'female', 'male', 'female'),
  height = c(1.88, 1.65, 1.72, 1.50),
  weight = c(84, 45, 77, 49)
)
```

	name	gender	height	weight
1	Anakin	male	1.88	84
2	Padme	female	1.65	45
3	Luke	male	1.72	77
4	Leia	female	1.50	49

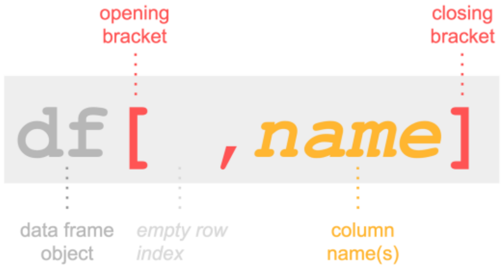
- subsetting a dataframe:





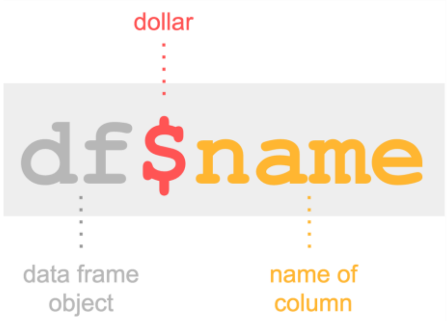
Data Structures in R

- selecting dataframe columns using names:




```r
# column Ozone
airquality[ ,"Ozone"]

# columns Wind and Temp
airquality[ ,c("Wind","Temp")]
```

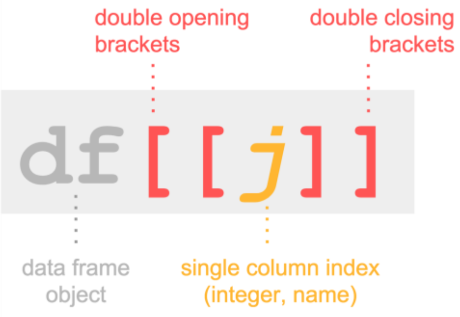
- selecting dataframe column using $ notation:




```r
# column Ozone
airquality$Ozone

# equivalently
airquality$"Ozone"

# equivalently
airquality$'Ozone'
```

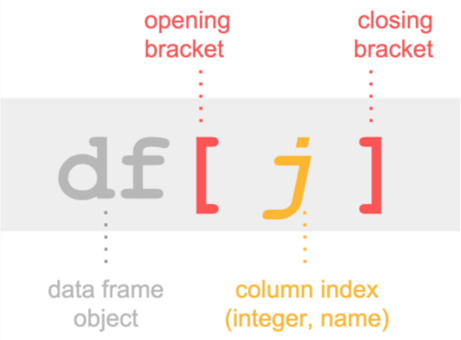
Data Structures in R

- selecting dataframe column using double brackets:




```r
# first column
airquality[[1]]

# column Wind
airquality[["Wind"]]
```

- selecting dataframe columns using vector notation:




```r
# first column
airquality[1]

# columns from 1 to 3
airquality[1:3]

# columns 2, 4, 6
airquality[c(2,4,6)]
```

Data Structures in R

- inspecting a dataframe:

	name	gender	height	weight
1	Anakin	male	1.88	84
2	Padme	female	1.65	45
3	Luke	male	1.72	77
4	Leia	female	1.50	49

```
> summary(df)
     name       gender       height           weight     
 Anakin:1   female:2   Min.   :1.500   Min.   :45.00  
 Leia  :1   male  :2   1st Qu.:1.613   1st Qu.:48.00  
 Luke  :1              Median :1.685   Median :63.00  
 Padme :1              Mean   :1.688   Mean   :63.75  
                       3rd Qu.:1.760   3rd Qu.:78.75  
                       Max.   :1.880   Max.   :84.00  
```

Function	Description
`str()`	structure
`head()`	First rows
`tail()`	Last rows
`summary()`	Descriptive statistics
`dim()`	Dimensions (# rows, # columns)
`nrow()`	Number of rows
`ncol()`	Number of columns
`names()`	Column names
`colnames()`	Column names
`rownames()`	Row names
`dimnames()`	List with row and column names

1. Overview & Setup
2. Data Types & Data Structures
3. Tools for Data Wrangling & Exploratory Data Analysis
4. Conducting Statistical Tests
5. Discussion, Feedback, & Further Resources

1. Importing Tabular Data

- most plain text files (e.g. `.csv`, `.txt`, `.dat`) can be read using the R base functions `read.table()`, `read.csv()`, and `read.delim()`
  - e.g. `read.table('C:/Users/Sichuan CDC/Desktop/file.txt')`
  - for these functions, it is also good practice to set `stringsAsFactors = FALSE` to avoid converting all character strings to **factors** (categorical variables) by default
- additional packages may be installed to read other types of files
- consider installing **readr**, a package that makes it easier and faster (10x faster than base functions) to read many types of tabular data

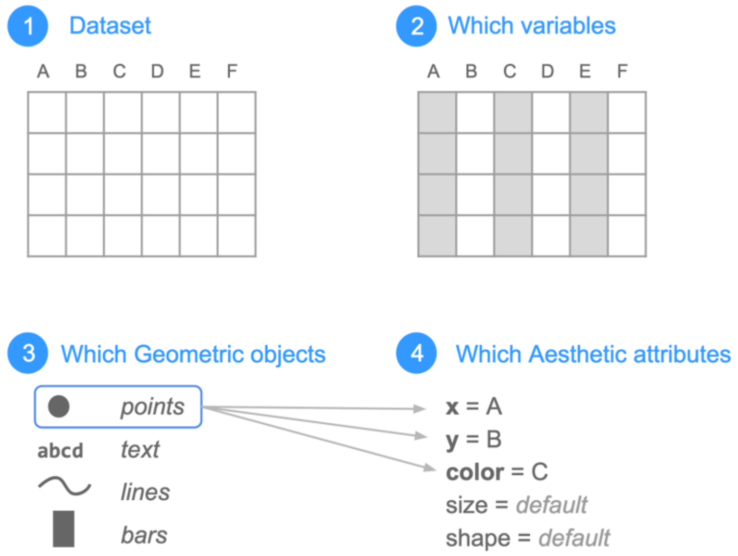
Type	Package	Function
Excel	gdata	**read.xls()**
Excel	xlsx	**read.xlsx()**
Excel	readxl	**read_excel()**
Excel	XLConnect	**readWorksheet()**
SPSS	foreign	**read.spss()**
SAS	foreign	**read.ssd()**
SAS	foreign	**read.xport()**
Matlab	R.matlab	**readMat()**
Stata	foreign	**read.dta()**
Octave	foreign	**read.octave()**
Minitab	foreign	**read.mtp()**
Systat	foreign	**read.systat()**

2. Cleaning and Manipulating Data

- the package `dplyr` provides many user-friendly functions for subsetting and manipulating data, including
  - `mutate()` to create new variables with functions of existing variables
  - `select()` to pick variables (columns) based on their names
  - `filter()` to pick cases (rows) based on their values
  - `arrange()` to change the ordering of rows
  - `summarise()` to collapse many values down to a single summary
- the input and output of all functions is a dataframe, with the 1st argument being the old dataframe, and subsequent arguments describing what to do with that dataframe (using its variable names, without quotes)

3. Visualizing Data

- the package `ggplot2` provides a way to produce visually pleasing graphics with automated inclusion of plot elements like legends, axes, and colors
- `ggplot2` aims to produce a wide range of statistical graphics with a compact syntax (mapping data to geometric objects and aesthetic attributes) and independent components

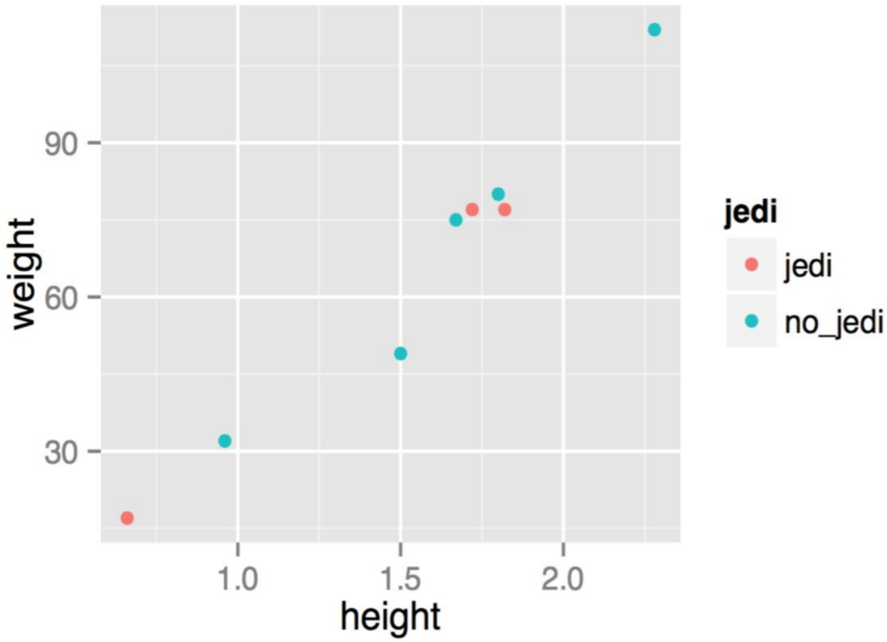


3. Visualizing Data

- example: building a **scatterplot** with the data from `starwars`:

name	gender	height	weight	jedi	species
Luke Skywalker	male	1.72	77	jedi	human
Leia Skywalker	female	1.5	49	no_jedi	human
Obi-Wan Kenobi	male	1.82	77	jedi	human
Han Solo	male	1.8	80	no_jedi	human
R2-D2	male	0.96	32	no_jedi	droid
C-3PO	male	1.67	75	no_jedi	droid
Yoda	male	0.66	17	jedi	yoda
Chewbacca	male	2.28	112	no_jedi	wookiee

Let's use these variables to make a scatterplot



3. Visualizing Data

- `ggplot2` syntax: mapping data to geometric objects and aesthetic attributes

```
ggplot(data = starwars) +
geom_point(aes(x = height, y = weight, color = jedi))
```

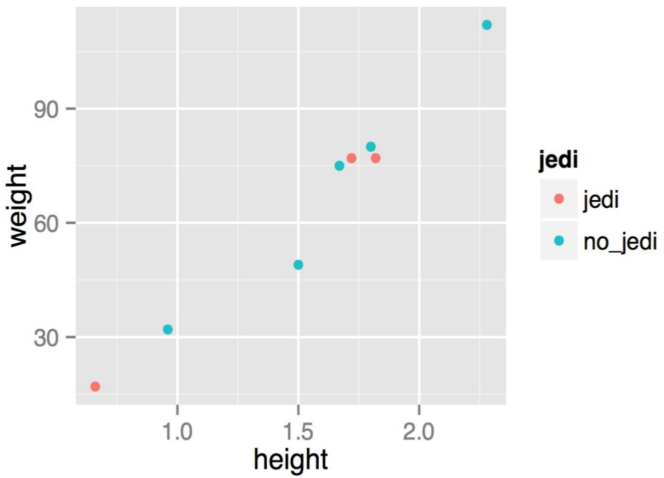
Dataset: starwars

Variables: height, weight, jedi

Geoms: points

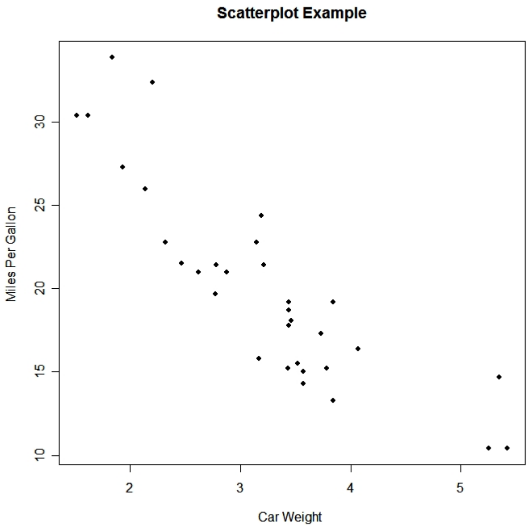
Aesthetic (perceptive attributes):

- X-axis: height
- Y-axis: weight
- Color: jedi

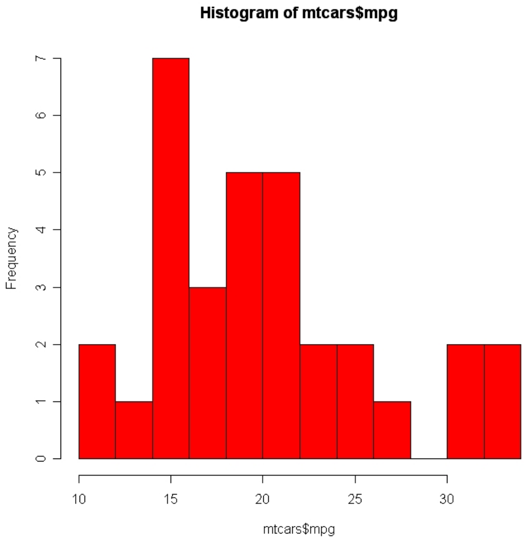


3. Visualizing Data

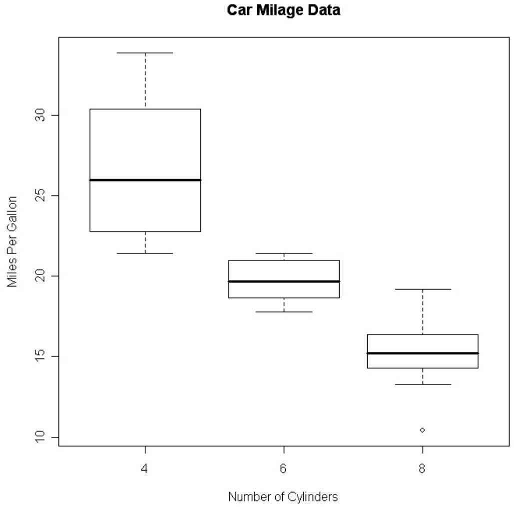
- nonetheless, there are many base R functions that are well-suited for quick exploratory analyses of data:




```
plot(wt, mpg, main="Scatterplot Example",
             xlab="Car Weight ",
ylab="Miles Per Gallon ", pch=19)
```




```
hist(mtcars$mpg, breaks = 12, col = ‘red’)
```

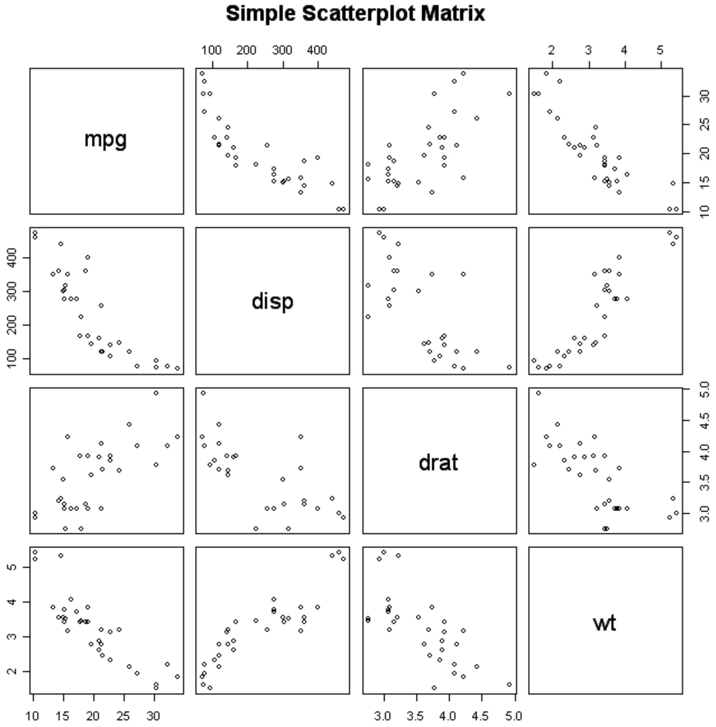



```
boxplot(mpg~cyl,data=mtcars,
main="Car Mileage Data",
             xlab="Number of
Cylinders", ylab="Miles Per
Gallon")
```

3. Visualizing Data

- nonetheless, there are many base R functions that are well-suited for quick exploratory analyses of data:

```
pairs(~mpg+disp+drat+wt,data=mtcars,
   main="Simple Scatterplot Matrix")
```



1. Overview & Setup
2. Data Types & Data Structures
3. Tools for Data Wrangling & Exploratory Data Analysis
4. Conducting Statistical Tests
5. Discussion, Feedback, & Further Resources

Conducting Statistical Tests

A **statistical test,** or **hypothesis test,** provides a mechanism for making quantitative decisions about a process. The intent is to determine whether there is enough evidence to "reject" a hypothesis, or theory, about the process. We call this the null hypothesis.

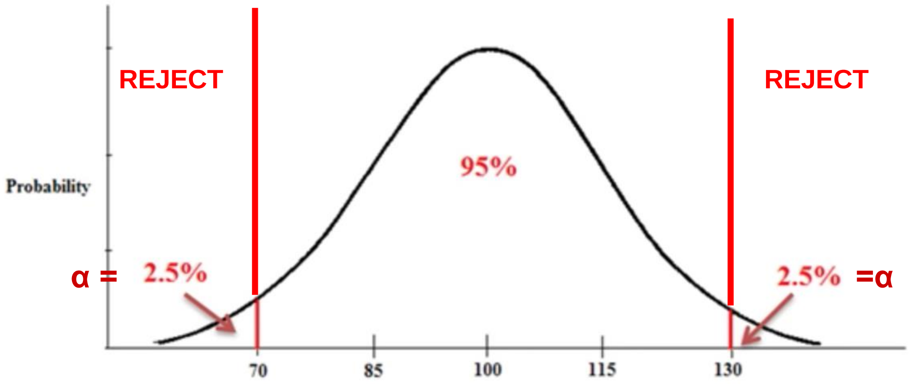
Steps:

1. Understand the study / experiment, and determine which hypothesis test to use
2. State the null hypothesis and alternative hypothesis
3. Determine / calculate parameters appropriate for the test
4. Calculate the test statistic
5. Find the critical value from the appropriate distribution table
6. Compare the test statistic and the critical value ; this will result in our conclusion on whether we reject or fail to reject the null hypothesis

Conducting Statistical Tests

There are four common statistical tests that can be easily implemented in R :

1. Z-test
2. T-test
3. ANOVA
4. Chi-Square



Z-test

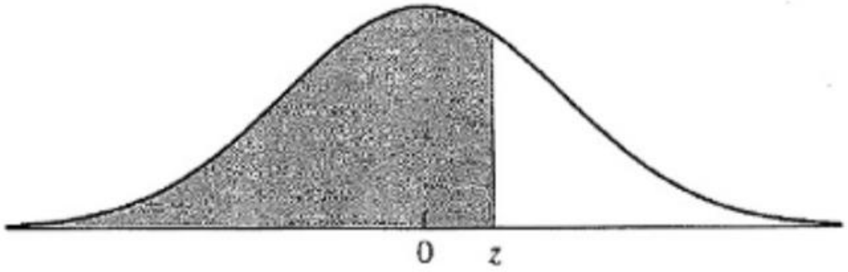
- **Parametric test** (assumes a normal distribution)
- Sample size is large ($n > 30$), population standard deviation may or may not be known
- Asks what is the probability of getting a certain sample mean
- The z-test uses z-scores and a normal distribution to determine the probability that the sample mean is drawn randomly from a known population

Hypotheses -

- **Null:** There is no statistically significant difference between the sample mean and the population mean.
- **Alternative:** There is a statistically significant difference between the sample mean and the population mean.

Z-test

$$Z = \frac{\bar{x} - \mu}{\frac{\sigma}{\sqrt{n}}}$$



Z	0.00	0.01	0.02
0.0	0.5000	0.5040	0.5080
0.1	0.5398	0.5438	0.5478
0.2	0.5793	0.5832	0.5871
0.3	0.6179	0.6217	**0.6255**

Our z test statistic is 0.32

According to the table,
$P(z<0.32) = 0.6255$

Therefore, $P(z>=0.32)$
$= 1 - 0.6255 = 0.3725$

P>0.05, therefore our decision is to NOT reject the null hypothesis

Z-test Example

Problem Statement:
Data represents the heights of 100 children from Hong Kong. We sample 50 from the population of their height (in inches) and weight (in pounds). We want to know if the sample of the children's heights are significantly different from the height of the population.

Index	Height (inches)	Weight (pounds)
1	65.78000000000000	112.98999999999999
2	71.52000000000000	136.49000000000001
3	69.40000000000001	153.03000000000000
4	68.22000000000000	142.34000000000000
5	67.79000000000001	144.30000000000001

Null Hypothesis: There is no statistically significant difference between the sample mean and the population mean.

Alternative Hypothesis: There is a statistically significant difference between the sample mean and the population mean (sample mean is greater than the population mean).

```
#z-stat calculation
sample_mean
z <- (sample_mean - pop_mean)/(pop_sd/sqrt(n))

#calculating the p-value/critical value
pnorm(z)
p_value <- 1 -pnorm(z)
p_value
```

T-test

- **Parametric test** (assumes a normal distribution)
- Sample size is small (n < 30), population standard deviation is not known
- Asks what is the probability of getting a certain sample mean?
- Compares the means of a variable from two groups

- **one sample t-test:** compares the means of two different groups (e.g., reaction times on a task for women vs. men)
  - **Null:** There is no statistically significant difference between the sample mean and the population mean.
  - **Alternate:** There is a significant difference between the sample mean and the population mean.
- **two sample t-test:** compares the means of the same group at two different times (e.g., reaction times for the same people on a task before or after a training period)
  - **Null:** There is no statistically significant difference between the means of the two variables/samples.
  - **Alternate:** There is a significant difference between the means of the two variables/samples.
- **paired t-test:** similar to the two sample t-test, except that the independence assumption is not valid.

T-test

$$t = \frac{(\bar{x}_1 - \bar{x}_2) - (\mu_1 - \mu_2)}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}}$$

DF = n1 + n2 - 2

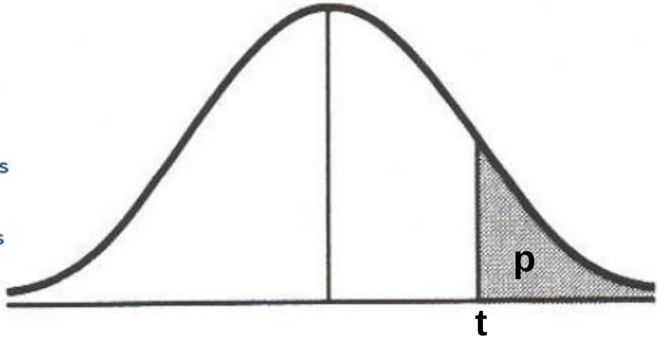
$$t = \frac{\bar{x} - \mu}{\frac{s}{\sqrt{n}}}$$

DF = n-1

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

A big t-value = different groups

A small t-value = similar groups



DF \ p	0.05	0.025	.02
1	6.314	12.71	15.89
2	2.920	4.303	4.849
3	2.353	3.182	3.482
4	2.132	2.776	2.999

One-Sample T-test Example

Problem Statement:
An outbreak of Salmonella-related illness was attributed to ice cream produced at a certain factory. Scientists measured the level of Salmonella in 9 randomly sampled batches of ice cream. The levels (in MPN/g) were:
0.593 0.142 0.329 0.691 0.231 0.793 0.519 0.392 0.418
Is there evidence that the mean level of Salmonella in the ice cream is greater than 0.3 MPN/g?

Let μ be the mean level of Salmonella in all batches of ice cream.

Null Hypothesis: There is no statistically significant difference between the sample mean and the population mean ($\mu = 0.3$)

Alternative Hypothesis: There is a statistically significant difference between the sample mean and the population mean ($\mu > 0.3$)

One-Sample

```
t.test(x, alternative="greater", mu=0.3)
```

```
        One Sample t-test

data:  x
t = 2.2050588385132, df = 8, p-value = 0.02926516484245
alternative hypothesis: true mean is greater than 0.3
95 percent confidence interval:
 0.324513289335024               Inf
sample estimates:
         mean of x
0.456444444444444
```

Two-Sample T-test Example

Problem Statement:
6 subjects were given a drug (treatment group) and an additional 6 subjects a placebo (control group). Their reaction time to a stimulus was measured (in ms). We want to perform a two-sample t-test for comparing the means of the treatment and control groups.

Let mu1 be the mean of the population taking medicine and mu2 the mean of the untreated population.

Null Hypothesis: there is no statistically significant difference between the group means (mu1-mu2=0)

Alternative Hypothesis: there is a statistically significant difference between the group means (mu1-mu2<0)

Two-Sample

```
#assuming equal standard deviation
t.test(Control,Treat,alternative="less", var.equal=TRUE)
```

```
        Two Sample t-test

data:  Control and Treat
t = -3.4456126735365, df = 10, p-value = 0.003136062175405
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval:
                -Inf -6.082743540872289
sample estimates:
         mean of x          mean of y
 88.83333333333333 101.66666666666667
```

```
#not assuming equal standard deviation
t.test(Control,Treat,alternative="less")
```

```
        Welch Two Sample t-test

data:  Control and Treat
t = -3.4456126735365, df = 9.4796824926709, p-value = 0.003391230079207
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval:
                -Inf -6.044949278000698
sample estimates:
         mean of x          mean of y
 88.83333333333333 101.66666666666667
```

Paired T-test Example

Problem Statement:
A study was performed to test whether cars get better mileage on premium gas than on regular gas. Each of 10 cars was first filled with either regular or premium gas, decided by a coin toss, and the mileage for that tank was recorded. The mileage was recorded again for the same cars using the other kind of gasoline. We use a paired t-test to determine whether cars get significantly better mileage with premium gas.

Let mu1 be the mean of the population of cars using premium gas and mu2 the mean of the population of cars using regular gas.

Null Hypothesis: there is no statistically significant difference between the group means (mu1-mu2=0)

Alternative Hypothesis: there is a statistically significant difference between the group means (mu1-

Paired t-test

```
t.test(prem,reg,alternative="greater", paired=TRUE)
```

```
	Paired t-test

data:  prem and reg
t = 4.4721359549996, df = 9, p-value = 0.0007749429558509
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 1.180206974429332                Inf
sample estimates:
mean of the differences
                      2
```

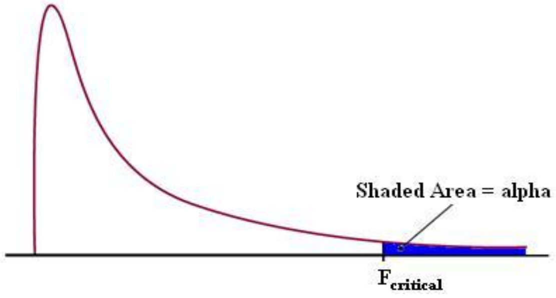
ANOVA

- ANalysis Of VAriance
- **Parametric test** (normal distribution)
- Asks whether the population means of several groups are equal (generalization of t-test for more than two groups)
- <u>**One-Way:**</u>
  - Assumptions:
    - The dependent variable is normally distributed (as well as response variable residuals).
    - The two groups have approximately equal variance on the dependent variable.
    - Variables are independent and identically distributed (iid).
  - Hypotheses:
    - **Null:** There are no significant differences between the groups' / samples' means (and thus all come from a larger overall population)
    - **Alternate:** There is a significant difference between the groups' / samples' means.

ANOVA

Source of Variation	Sums of Squares (SS)	Degrees of Freedom (df)	Mean Squares (MS)	F
Between Treatments	$SSB = \Sigma n_j (\bar{X}_j - \bar{X})^2$	k-1	$MSB = \frac{SSB}{k-1}$	$F = \frac{MSB}{MSE}$
Error (or Residual)	$SSE = \Sigma\Sigma (X - \bar{X}_j)^2$	N-k	$MSE = \frac{MSE}{N-k}$	
Total	$SST = \Sigma\Sigma (X - \bar{X})^2$	N-1		

- X = individual observation,
- $\bar{X}_j$ = sample mean of the j^{th} treatment (or group),
- $\bar{X}$ = overall sample mean,
- k = the number of treatments or independent comparison groups, and
- N = total number of observations or total sample size.



Variance Between + Variance Within = Total Variance

$$\frac{\text{Variance Between}}{\text{Variance Within}}$$

Total Variance Components

Degree of Freedom: NUMERATOR

Degree of Freedom: DENOMINATOR

$\alpha = 0.05$	1	2	3	4
1	161.45	199.50	215.71	224.58
2	18.51	19.00	19.16	19.25
3	10.13	9.55	9.28	9.12
4	7.71	6.94	6.59	6.39

ANOVA Example

Problem Statement:
A drug company tested three formulations of a pain relief medicine for migraine headache sufferers. For the experiment 27 volunteers were selected and 9 were randomly assigned to one of three drug formulations. The subjects were instructed to take the drug during their next migraine headache episode and to report their pain on a scale of 1 to 10 (10 being most pain).

Null Hypothesis: there are no significant differences between the drug groups' means

Alternative Hypothesis: there is a significant difference between the drug group's means

DrugA 4 5 4 3 2 4 3 4 4 | **DrugB** 6 8 4 5 4 6 5 8 6 | **DrugC** 6 7 6 6 7 5 6 5 5

```
results = aov(pain ~ drug, data=migraine)  # pain is the response (represents the response variable)
                                           # and drug is the factor (the variable that separates the data into groups)
summary(results)  #significance codes are the categorizations of the p-value
```

```
summary(results)
```

```
            Df        Sum Sq        Mean Sq  F value     Pr(>F)
drug         2 28.2222222222 14.111111111 11.90625 0.00025588 ***
Residuals   24 28.4444444444  1.185185185185
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
pairwise.t.test(pain, drug, p.adjust="bonferroni")
```

```
        Pairwise comparisons using t tests with pooled SD

data:  pain and drug

  A               B
B 0.0011857576436 -
C 0.0006844387757 1.0000000000000

P value adjustment method: bonferroni
```

```
TukeyHSD(results, conf.level = 0.95)
```

```
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = pain ~ drug, data = migraine)

$drug
                 diff                lwr              upr            p adj
B-A 2.1111111111111103  0.8295027637144792 3.392719458507742 0.001107263887066
C-A 2.2222222222222201  0.9406138748255890 3.503830569618851 0.0006452912758791
C-B 0.1111111111111098 -1.1704972362855213 1.392719458507741 0.9745173136672108
```

Chi-Square Test

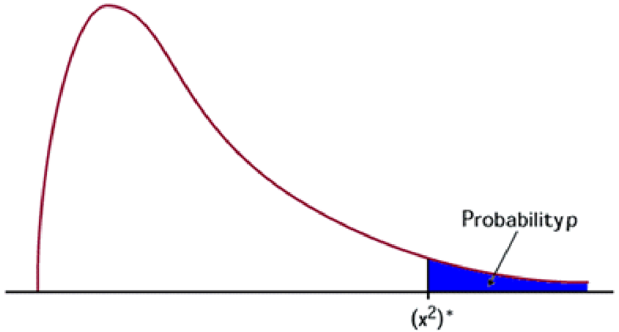
- **Nonparametric test** (does not assume known distribution)
- Used for nominal data (such as wanting to see if there is a difference based on gender to your project question)
- Used to find out how the observed value of a given phenomena is significantly different from the expected value

<u>**Goodness-of-Fit:**</u> Determines if the observed frequencies are different from what we would expect to find (often times using population data or theoretical data). Used to compare the observed sample distribution with the expected probability distribution.

- Assumptions:
  - The population is at least 10 times as large as the sample.
  - The expected value for each level of the variable is at least 5.
  - Sampling method is simple random sampling (SRS).
- Hypotheses:
  - **Null:** there is no significant difference between the observed and the expected value (expected distribution is correct).
  - **Alternate:** there is a significant difference between the observed and the expected value (expected distribution is not correct).

Chi-Square Test

$$X^2 = \frac{(O_i - E_i)^2}{E_i}$$



df	0.5	0.10	0.05	0.02	0.01	0.001
1	0.455	2.706	3.841	5.412	6.635	10.827
2	1.386	4.605	5.991	7.824	9.210	13.815
3	2.366	6.251	7.815	9.837	11.345	16.268
4	3.357	7.779	9.488	11.668	13.277	18.465
5	4.351	9.236	11.070	13.388	15.086	20.517

Chi-Square Test Example

Problem Statement:
We collected wild tulips and found that 81 were red, 50 were yellow, and 27 were white. Are these colors equally common? We want to know if there is any significant difference between the observed proportions and the expected proportions.

Null hypothesis: There is no significant difference between the observed and the expected value.

Alternative hypothesis: There is a significant difference between the observed and the expected value.

```
tulip <- c(81, 50, 27)
res1 <- chisq.test(tulip, p = c(1/3, 1/3, 1/3))    #p is the observed proportions
res1$expected  #all larger than 5, which is a requirement of using chi-sq goodness of fit test

res2 <- chisq.test(tulip, p = c(1/2, 1/3, 1/6))   #p is the expected proportions
```

res1

```
	Chi-squared test for given probabilities

data:  tulip
X-squared = 27.886075949367, df = 2, p-value = 8.802692814076e-07
```

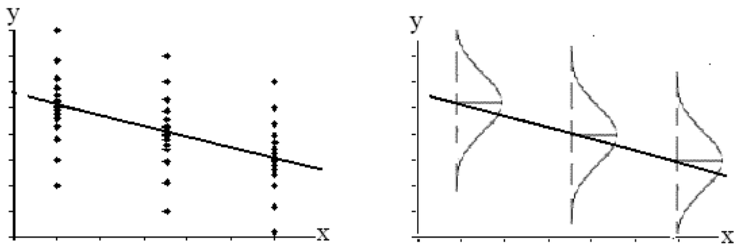
res2

```
	Chi-squared test for given probabilities

data:  tulip
X-squared = 0.20253164556962, df = 2, p-value = 0.9036927788237
```

Simple Linear Regression

$$Y = \beta_0 + \beta_1 x + \epsilon$$

- Explores and models the relationship between two or more variables
- We can carry out the previously mentioned hypotheses test on the coefficients of a simple linear regression



Assumptions:

- Random errors are
  - normally distributed (**Parametric**)
  - Independently and identically distributed (iid)

Hypotheses:

- T-test
  - **Null:** the true slope equals some constant value($\beta_1 = \beta_{1,0}$)
  - **Alternative:** the true slope does not equal some constant value ($\beta_1 \neq \beta_{1,0}$)
- ANOVA
  - **Null:** regression model is not statistically significant (a regression model cannot be applied to the observed data) ($\beta_1 = 0$)
  - **Alternative: regression model is statistically significant (**a regression model can be applied to the observed data) ($\beta_1 \neq 0$)

Simple Linear Regression

Problem Statement:
Decide whether there is a significant relationship between the variables, distance and speed, in the linear regression model of the data set "cars" at 0.05 significance level.

Null Hypothesis: There is no significant relationship between the variables (β = 0)

Alternative Hypothesis: There is a significant relationship between the variables (β != 0)

```
data(cars)
lm_model = lm(dist~., data = cars)
summary(lm_model)
```

```
Residuals:
          Min            1Q        Median            3Q           Max
-29.069080291971 -9.525321167883 -2.271854014599  9.214715328467 43.201284671533

Coefficients:
                  Estimate        Std. Error  t value   Pr(>|t|)
(Intercept) -17.5790948905109  6.7584401693792 -2.60106  0.012319 *
speed         3.9324087591241  0.4155127766571  9.46399 1.4898e-12 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 15.37958674882 on 48 degrees of freedom
Multiple R-squared:  0.6510793807583,   Adjusted R-squared:  0.6438102011907
F-statistic: 89.56710653647 on 1 and 48 DF,  p-value: 1.489836496295e-12
```

1. Overview & Setup
2. Data Types & Data Structures
3. Data Wrangling & Exploratory Data Analysis
4. Conducting Statistical Tests
5. Discussion, Feedback, & Further Resources

Questions, Comments, & Feedback

Further Resources

- R学习路线 - https://blog.csdn.net/BETTINA26/article/details/52956391
- 时间序列分析及应用: R语言 http://bbs.pinggu.org/a-1423839.html
- R technical manuals: https://cran.r-project.org/manuals.html
- R contributed documentation (**some written in Chinese**): https://cran.r-project.org/other-docs.html
    - R导论 (An introduction to R): https://cran.r-project.org/doc/contrib/Ding-R-intro_cn.pdf
    - 153分钟学会R: https://cran.r-project.org/doc/contrib/Liu-FAQ.pdf
- CRAN task views: https://cran.r-project.org/web/views
- The R Journal: https://journal.r-project.org
- R-related posts in **StackOverflow** (forum for asking programming questions): http://stackoverflow.com/questions/tagged/r
- R Bloggers: https://www.r-bloggers.com

Acknowledgements

Some of our presentation content was adopted from instructional material developed for UC Berkeley's Statistics 133 course ("Concepts in Computing with Data") by Prof. Gaston Sanchez.

For direct access to course materials and more tutorials, please visit https://github.com/ucb-stat133/stat133-fall-2017/.

Thanks for listening, and welcome to R!