USING R FOR EPIDEMIOLOGICAL RESEARCH

Kiffer G. Card, PhD
Kirk J. Hepburn, MPP

INTRODUCTION TO R

Simon Fraser University, Faculty of Health Sciences, Health Sciences (HSCI) 432

OUTLINE

- R Basics
- Exploring and Summarizing Data
- Combining Data Sets
- Recoding Variables
- Descriptive Statistics

1.1. R-BASICS

Types of statistical software
About *R*
Installing *R*
Installing *RStudio*
A quick tour of *RStudio*
Getting help
R packages
R syntax
Annotating your *R* code

Types of Statistical Software

- Command-line software (e.g., SAS, R, SPSS scripts)
  - requires knowledge of syntax of commands
  - reproducible results through scripts
  - detailed analyses possible
- GUI-based software (e.g., SPSS point-and-click functions, Excel)
  - does not require knowledge of commands
  - not reproducible actions
- Hybrid types (both command-line and GUI)

Types of Statistical Software

SPSS, STATA, and SAS users are like muggles. They are limited in their ability to change their environment. They have to rely on algorithms that have been developed for them. The way they approach a problem is constrained by how SAS/IBM-employed programmers thought to approach them. And they have to pay money to use these constraining algorithms.

Types of Statistical Software

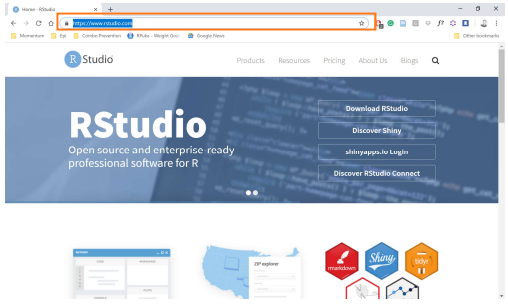
R users are like wizards. They can rely on functions (spells) that have been developed for them by statistical researchers, but they can also create their own. They don't have to pay for the use of them, and once experienced enough (like Dumbledore), they are almost unlimited in their ability to change their environment.

About R

- R is a **statistical language** that was developed in response to Bell Labs (1976) S/S-plus language.
- R is commonly accessed through the **integrated development environment** called ***RStudio***.
- **Base R** performs a vast number of useful statistical operations.
- Base R can be enhanced with **packages**.
- These packages are **open source,** created by the community of R users, and typically documented in the *Journal of Statistical Software*. These resources are therefore **open for public use free of charge**.

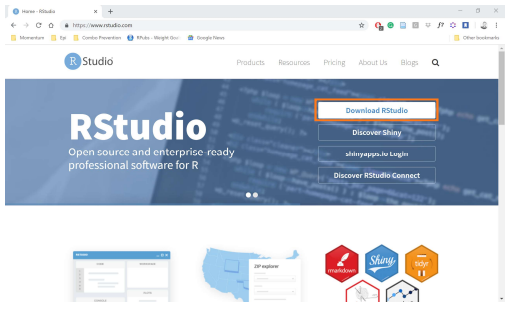
R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Installing *R*



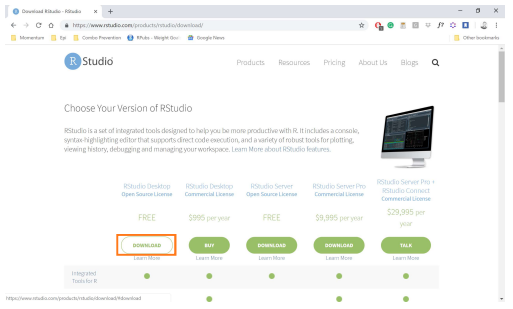
R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Installing *R*



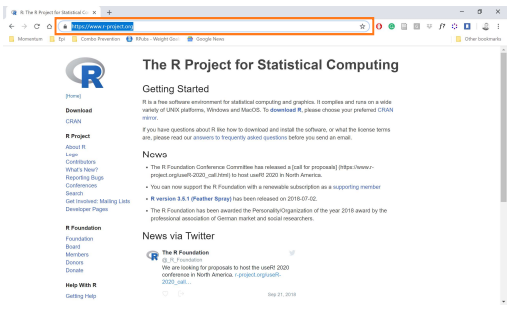
R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Installing *R*



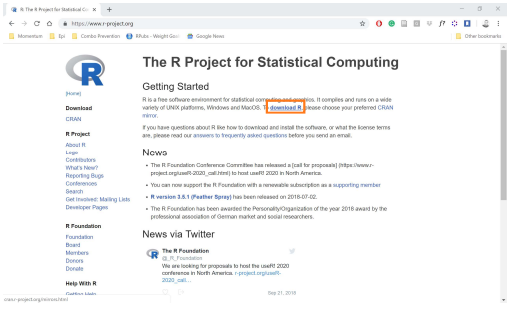
R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Installing *RStudio*



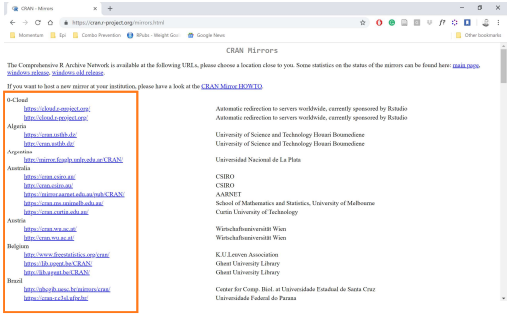
R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Installing *RStudio*



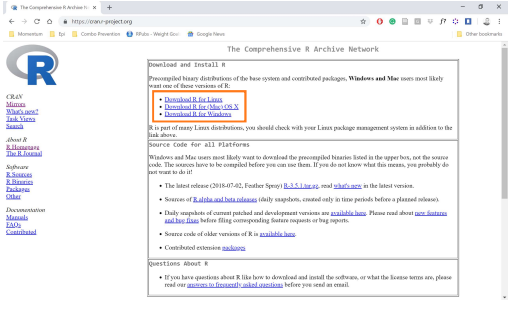
R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Installing *RStudio*



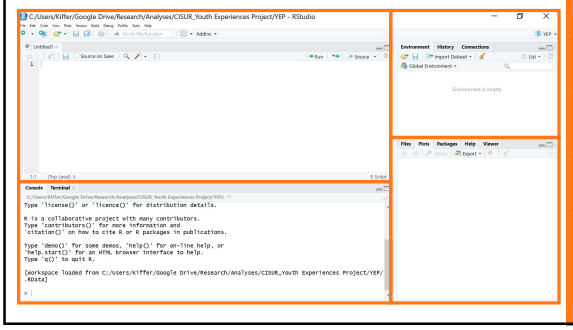
R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Installing *RStudio*



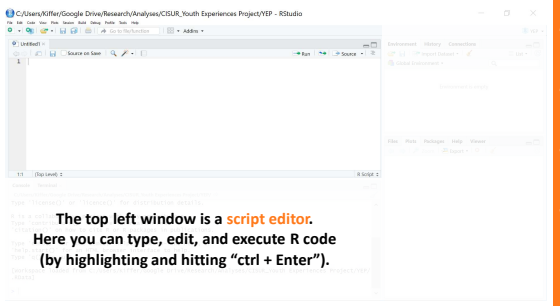
R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

A quick tour of *RStudio*



R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

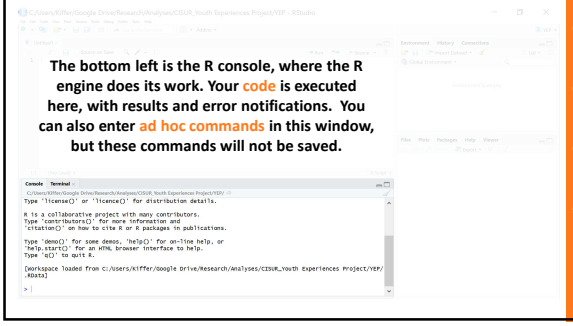
A quick tour of *RStudio*



The top left window is a script editor. Here you can type, edit, and execute R code (by highlighting and hitting "ctrl + Enter").

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

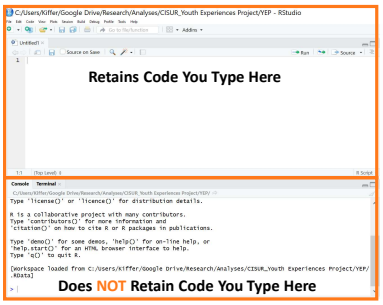
A quick tour of *RStudio*



The bottom left is the R console, where the R engine does its work. Your code is executed here, with results and error notifications. You can also enter ad hoc commands in this window, but these commands will not be saved.

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

A quick tour of *RStudio*



R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

A quick tour of *RStudio*

In the top right, R objects are displayed and summarized. **Objects** are named sets of data, whether a set of data from a survey, a list of names, a set of randomly generated numbers, or even functions you have **defined** (created) yourself.

Note: R is case sensitive. DATA, data, and Data would be three different objects.

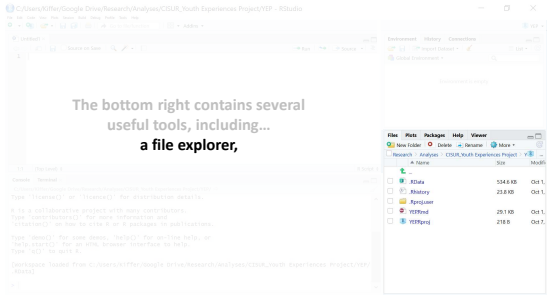
R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

A quick tour of *RStudio*

The bottom right contains several useful tools, including…

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

A quick tour of *RStudio*

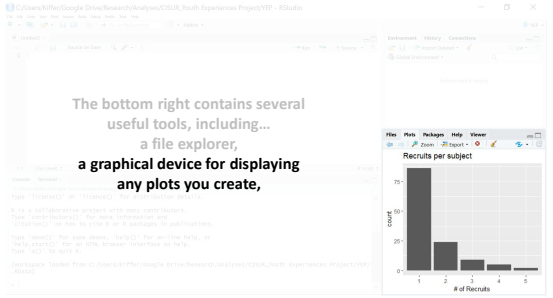


The bottom right contains several useful tools, including…

a file explorer,

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

A quick tour of *RStudio*



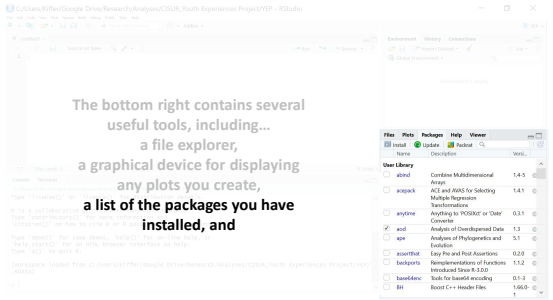
The bottom right contains several useful tools, including…

a file explorer,

a graphical device for displaying any plots you create,

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

A quick tour of *RStudio*



The bottom right contains several useful tools, including…

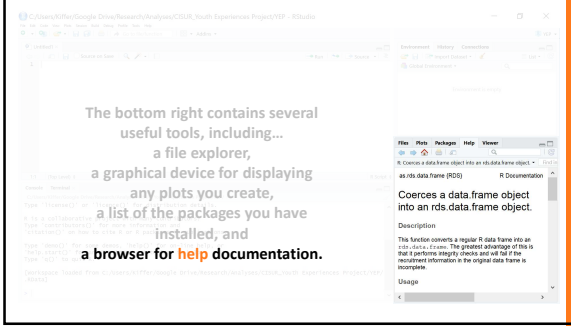
a file explorer,

a graphical device for displaying any plots you create,

a list of the packages you have installed, and

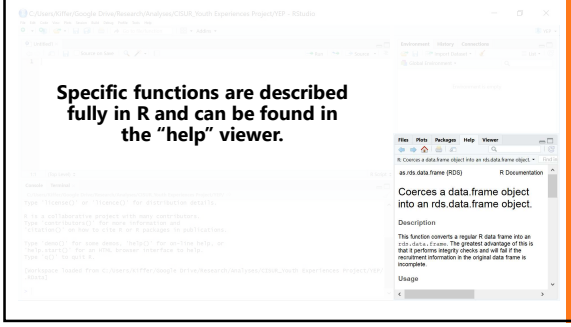
R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

A quick tour of *RStudio*



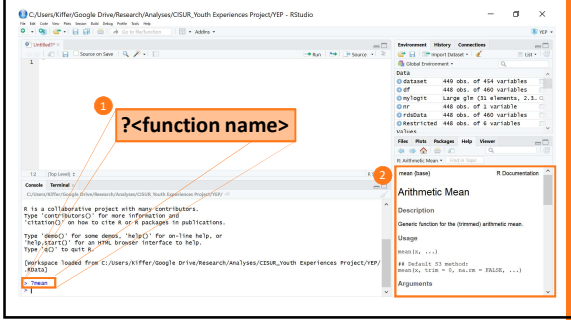
R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Getting Help



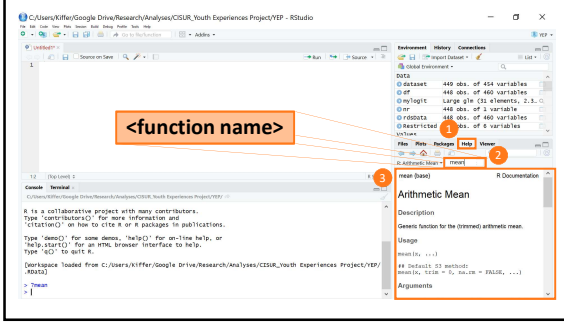
R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Getting Help



R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Getting Help



R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Getting Help

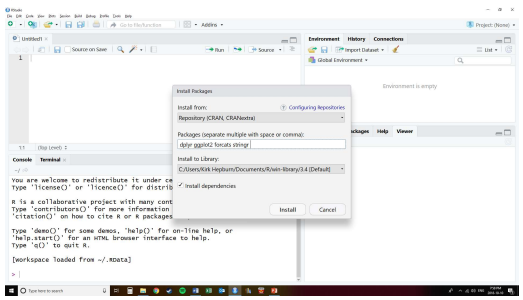
- Furthermore, if you have questions about a specific function or error, the R community is extremely active.
  - Your question has likely already been asked on *StackExchange* or similar websites:
    - https://stackexchange.com/
    - https://www.r-bloggers.com/
    - https://www.statmethods.net/

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

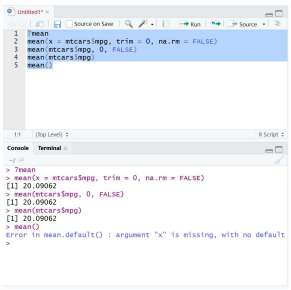
R Packages

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

R Packages



R Syntax



Arithmetic Mean

Description

Generic function for the (trimmed) arithmetic mean.

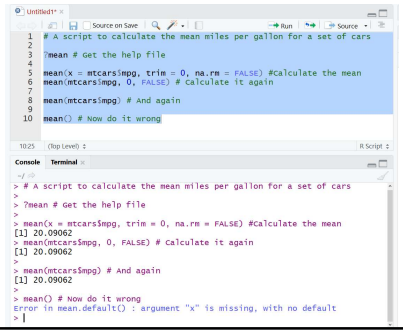
Usage

```
mean(x, ...)

## Default S3 method:
mean(x, trim = 0, na.rm = FALSE, ...)
```

Arguments

x	An R object. Currently there are methods for numeric/logical vectors and date, date-time and time interval objects. Complex vectors are allowed for `trim = 0`, only.
trim	the fraction (0 to 0.5) of observations to be trimmed from each end of `x` before the mean is computed. Values of trim outside that range are taken as the nearest endpoint.
na.rm	a logical value indicating whether NA values should be stripped before the computation proceeds.
...	further arguments passed to or from other methods.

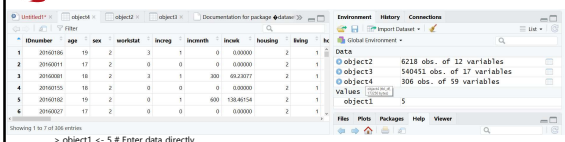
Annotating Your R Code



1.2. EXPLORING DATA

- Entering your data
- Using *R* to look at your data
- Types of data in *R*
- Data dictionaries
- Skip Patterns

Entering Data into R



```
> object1 <- 5 # Enter data directly

> object2 <- read.csv(file = "https://catalogue.data.gov.bc.ca/dataset/e43be180-7511-4e6f-84d3-adfc9f5c3e2b/resource/79cfd06f-883e-4106-a411-4b843ee39236/download/bc_liquor_store_product_price_list_july_2018.csv") # Enter a csv from the Internet

> library(readxl) # Load a package to help read Excel data

> object3 <- read_excel(path = "C:/Users/Kirk Hepburn/Downloads/fsa.xlsx", sheet = 1) # Read in a downloaded Excel dataset>

> library(haven) # Load another package to help read SPSS data

> object4 <- read_sav(file = "https://io.unisa.edu.au/pluginfile.php/1020313/mod_book/chapter/106604/HLTH1025_2016.sav") # Read in an SPSS dataset

> View(object4)

> object1
[1] 5
```

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Using R to Look at Your Data

Hide

```
rownames(ToothGrowth)
```

```
 [1] "1"  "2"  "3"  "4"  "5"  "6"  "7"  "8"  "9"  "10"
[11] "11" "12" "13" "14" "15" "16" "17" "18" "19" "20"
[21] "21" "22" "23" "24" "25" "26" "27" "28" "29" "30"
[31] "31" "32" "33" "34" "35" "36" "37" "38" "39" "40"
[41] "41" "42" "43" "44" "45" "46" "47" "48" "49" "50"
[51] "51" "52" "53" "54" "55" "56" "57" "58" "59" "60"
```

Hide

```
colnames(ToothGrowth)
```

```
[1] "len"  "supp" "dose"
```

Hide

```
ToothGrowth["51", "supp"]
```

```
[1] OJ
Levels: OJ VC
```

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

R Data Types

- R has a wide variety of classes of objects.
- The classes that you should be concerned with:
  - Vectors
    - Numeric - 1, 2, 5.3, 6, -2 , 4, NA
    - Factor - 1, 2, 3, 2, 3, 3, 2, 1, NA
    - Character/String - "one", "two", "three", NA
    - Logical - TRUE, TRUE, FALSE, NA
  - Data frames – objects that can contain multiple objects (as columns) with different classes
  - You can check an object's class using the "class()" function.

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Data Dictionaries

- A data dictionary, or metadata repository, as defined in the IBM Dictionary of Computing, is a "centralized repository of information about data such as meaning, relationships to other data, origin, usage, and format."
  - Provides question & response text
  - Provides variable name & response codes
  - Provides descriptive frequency counts & proportions
  - Provides information about skip/display logic
  - Provides codes for missing-ness
  - Provides information about the data type for each variable

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

1.3. SUMMARIZING DATA

Frequencies
Proportions
Cross-Tabulated Frequencies
Cross-Tabulated Proportions
Measures of Central Location
Measures of Spread
Measures of Kurtosis
Measures of Skew
Calculating Rates

Frequencies

- Count of the occurrences of a value
- "Cross-tabulate": to count the occurrences of a value across different sets of data

Frequency

Hide

```
table(Education = infert$education) # Frequency of education levels in infertility data
```

```
Education
  0-5yrs 6-11yrs 12+ yrs
      12     120     116
```

Cross-tabulation

Hide

```
table(Education = infert$education, NumberOfBirths = infert$parity) # Education level by Number of births
```

```
           NumberOfBirths
Education   1  2  3  4  5  6
  0-5yrs    3  0  0  3  0  6
  6-11yrs  42 62 21 12  3  0
  12+ yrs  54 39 15  3  3  2
```

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Proportions

- Proportion of total occurrences matching each value

Proportion Table

Hide

```
prop.table(table(Education = infert$education)) # Frequency of education levels in infertility data
```

```
Education
    0-5yrs   6-11yrs    12+ yrs
0.0483871 0.4838710 0.4677419
```

Cross-tabulation of Proportions

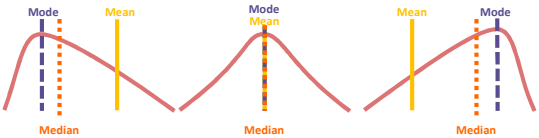
Hide

```
prop.table(table(Education = infert$education, NumberOfBirths = infert$parity), margin = 1) # Education level b
y Number of births
```

```
           NumberOfBirths
Education           1          2          3          4          5          6
  0-5yrs    0.2500000 0.00000000 0.00000000 0.25000000 0.00000000 0.50000000
  6-11yrs   0.3500000 0.35000000 0.17500000 0.10000000 0.02500000 0.00000000
  12+ yrs   0.4655172 0.33620690 0.12931034 0.02586207 0.02586207 0.01724138
```

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Measures of Central Location



The **mean**, **μ**, is the average value of the data set.
The **median** is the middle value of the data set.
The **mode** is the most common value of the data set.

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Measures of Central Location | Mean

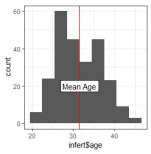
- "Average", with equal weight on either side
- Best for normal data
- Highly susceptible to outliers
- Good for continuous and discrete data
- Not good for ordinal or nominal data

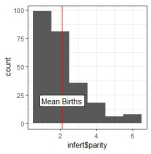
Mean

```
mean(infert$age) # Calculate mean age of sample
[1] 31.50603

mean(infert$parity) # Calculate mean number of births in sample
[1] 2.092742
```

Measures of Central Location | Mean

Which mean is a better representation of its data?





Measures of Central Location | Median

- The "middle point", or 50th percentile of the data
- Not sensitive to outliers, good for non-normal data
- Good for continuous, discrete, and ordinal data
- Not good for nominal data

Median

```
median(infert$age) # Calculate median age of sample
[1] 31

median(infert$parity) # Calculate median number of births in sample
[1] 2
```

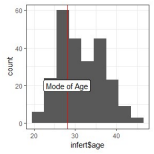
Measures of Central Location | Mode

- The most frequent value in the data
- Useful for all kinds of data
- No "easy" R function, so just look at a frequency table

Mode

```
table(infert$age) # Frequency of ages in sample to find mode
```

```
21 23 24 25 26 27 28 29 30 31 32 34 35 36 37 38 39 40 41 42 44
 6  6  3 15 15 15 30 12 12 21 15 18 15 12  8  9  6  3  6  3
```



Measures of Spread

- Range
- Interquartile Range
- Variance
- Standard Deviation
- Coefficient of Variation

Measures of Spread | Range

- Minimum and maximum values
- Not terribly useful when there are outliers, similar to the mean

Range

```
range(infert$age) # Find the range of ages in the data
```

```
[1] 21 44
```

```
range(infert$parity) # Find the range of births in the data
```

```
[1] 1 6
```

Measures of Spread | Interquartile Range

- How far apart are the 25th and 75th percentiles?
- Not as easily influenced by outliers

Interquartile Range

```
IQR(x = infert$age) # Find the interquartile range of ages in the data
```

```
[1] 7.25
```

```
IQR(x = infert$parity) # Find the interquartile range of births in the data
```

```
[1] 2
```


Measures of Spread | Summary

- Multiple measures of spread provide a good description of the data

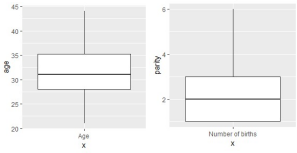
Summary

```
summary(infert$age) # Summarize the age data
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  21.00   28.00   31.00   31.50   35.25   44.00 
```

```
summary(infert$parity) # Summarize the number of births
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
  1.000   1.000   2.000   2.093   3.000   6.000 
```



Measures of Spread | Variance

- The amount of spread around the mean
- Useful for normal data

$$s^2 = \frac{\sum_{i=1}^{n}(X_i - X_{avg})^2}{n-1}$$

Variance

```
var(x = infert$age) # Calculate the variance of age in the data
```

```
[1] 27.57893
```

Measures of Spread | Standard Deviation

- The square root of the variance
- Useful for normal data
- Interpreted in units of the data, so more descriptive than the variance

Standard Deviation

```
sd(x = infert$age) # Calculate the standard deviation of age in the data
```

```
[1] 5.251565
```


Measures of Spread | Coefficient of Variation

- A way to compare the variation of two or more sets of data representing different quantities with different units
- Not frequently used

$$CV\ (\%) = \left(\frac{Standard\ deviation}{Mean}\right) \times 100$$

Coefficient of Variation

```
(sd(x = infert$age) / mean(x = infert$age)) * 100 # Calculate the coefficient of variation for age
```

```
[1] 16.6695
```


1.4. MERGING DATASETS

Adding Variables
Adding Observations
Wide vs. Long Data
Dropping Variables
Dropping Observations

Data Structure: Wide & Long Data

- Wide data

Person	Age	Weight
Bob	32	128
Alice	24	86
Steve	64	95

- Long data

Person	Variable	Value
Bob	Age	32
Bob	Weight	128
Alice	Age	24
Alice	Weight	86
Steve	Age	64
Steve	Weight	95

Wide data is preferable to long data for most analyses. Each row should be one case, each column should measure one variable.

If you happen to have long data, R has functions available to convert it to wide data (and back). For instance, the gather() and spread() functions from the tidyverse package.

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Data Structure: Summary Data

- Summary data is sometimes all that is available. It is not ideal as multiple cases are represented by each row.

CITY	IMAGE	LANGUAGE	MESSAGE	IMPRESSIONS	PEOPLE_SHOWN	CLICKS	REACTIONS	COMMENTS	SHARES	TOTAL_ENGAGEMENTS
Montreal	Image 1	English	Altruism	29,932	15,454	176	6	0	0	6
Montreal	Image 1	French	Altruism	20,087	13,266	180	3	0	1	4
Toronto	Image 1	English	Altruism	12,447	10,378	150	8	1	0	9
Vancouver	Image 1	English	Altruism	14,056	10,354	140	14	0	3	17
Montreal	Image 2	English	Altruism	18,384	12,404	144	8	5	0	13
Montreal	Image 2	French	Altruism	9,235	7,630	175	7	0	1	8
Toronto	Image 2	English	Altruism	11,880	10,023	171	11	5	2	18
Vancouver	Image 2	English	Altruism	12,725	10,064	148	25	3	3	31
Montreal	Image 1	English	Curiosity/Personal Health	18,828	12,458	246	4	2	0	6
Montreal	Image 1	French	Curiosity/Personal Health	28,680	16,450	218	4	0	1	5
Toronto	Image 1	English	Curiosity/Personal Health	22,171	15,856	155	17	0	4	21
Vancouver	Image 1	English	Curiosity/Personal Health	19,211	13,551	140	24	1	3	28
Montreal	Image 2	English	Curiosity/Personal Health	10,713	8,470	150	14	1	1	16
Montreal	Image 2	French	Curiosity/Personal Health	11,378	8,872	194	24	5	4	33
Toronto	Image 2	English	Curiosity/Personal Health	9,571	8,564	147	16	2	0	18
Vancouver	Image 2	English	Curiosity/Personal Health	14,487	10,751	161	7	0	0	7
Montreal	Image 1	English	Financial Incentive	16,055	10,705	172	5	2	2	9
Montreal	Image 1	French	Financial Incentive	12,201	8,958	211	13	2	2	17
Toronto	Image 1	English	Financial Incentive	9,907	8,561	195	11	0	3	14
Vancouver	Image 1	English	Financial Incentive	10,788	8,422	220	21	9	5	35
Montreal	Image 2	English	Financial Incentive	14,121	10,871	155	5	3	3	11
Montreal	Image 2	French	Financial Incentive	11,895	10,068	223	15	5	3	23
Toronto	Image 2	English	Financial Incentive	9,152	7,795	158	17	7	3	27
Vancouver	Image 2	English	Financial Incentive	9,630	8,043	240	16	4	3	23

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Adding Variables

- Sometimes necessary when additional data is collected on cases which already exist

```
band_members # Which band are they in?
```

name <chr>	band <chr>
Mick	Stones
John	Beatles
Paul	Beatles

3 rows

```
merge(x = band_members, y = band_instruments, by = "name") # Add in their instruments, only keep matches
```

name <chr>	band <chr>	plays <chr>
John	Beatles	guitar
Paul	Beatles	bass

2 rows

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Adding Observations

- Continuation of a survey, or multiple administrations of a survey

```
trial1 # First dataset
```

	Year <dbl>	Drug <chr>	Sex <chr>
	2010	A	M
	2010	B	F
	2010	A	NA

3 rows

```
rbind(trial1, trial2) # Combined datasets
```

	Year <dbl>	Drug <chr>	Sex <chr>
	2010	A	M
	2010	B	F
	2010	A	NA
	2011	B	F
	2011	B	F
	2011	A	M

6 rows

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Dropping Variables or Observations

- When you might need to do it
- How to do it

```
trial1[, -3] # Remove `Sex`, the third variable
```

	Year <dbl>	Drug <chr>
	2010	A
	2010	B
	2010	A

3 rows

```
trial1[-3, ] # Remove the third observation
```

	Year <dbl>	Drug <chr>	Sex <chr>
	2010	A	M
	2010	B	F

2 rows

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

1.5. RECODING VARIABLES

Recoding Categorical Variables
Categorizing Continuous Data
Combining Variables
Missing Data

Recoding Categorical Variables

- In the code below we collapse a three-level categorical variable measuring years of education into a dichotomous variable:

```
infert$educationR <- NA                                          ##Create New Variable
infert$educationR[infert$education == "0-5yrs"] <- "11 or less years"    ## 0-5 as 11 or less
infert$educationR[infert$education == "6-11yrs"] <- "11 or less years"   ## 6-11 as 11 or less
infert$educationR[infert$education == "12+ yrs"] <- "12 or more years"   ## Recode 12+ as 12 or more
```

- You could check your results using the CrossTable() function in the gmodels package:

```
library(gmodels)
CrossTable(infert$educationR, infert$education, prop.r = FALSE, prop.c = FALSE, prop.chisq = FALSE, prop.t = FALSE)
```

```
Total Observations in Table:  248

                    | infert$education
infert$educationR   |     0-5yrs |    6-11yrs |     12+ yrs | Row Total |
--------------------|------------|------------|------------|-----------|
 11 or less years   |         12 |        120 |          0 |       132 |
--------------------|------------|------------|------------|-----------|
 12 or more years   |          0 |          0 |        116 |       116 |
--------------------|------------|------------|------------|-----------|
      Column Total  |         12 |        120 |        116 |       248 |
--------------------|------------|------------|------------|-----------|
```

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Categorizing Continuous Data

- In the code below, I am recoding data which shows the effect of two sophoric drugs on changes in sleep duration among 10 patients.

```
sleep$dihrs <- NA                                          ##Create New Variable
sleep$dihrs[sleep$extra > 0.0] <- "Increased"              ## Recode level 1
sleep$dihrs[sleep$extra == 0.0] <- "No Change"             ## Recode level 2
sleep$dihrs[sleep$extra < 0.0] <- "Decreassed"             ## Recode level 3


sleep$dihrs2 <- NA                                                    ##Create New Variable
sleep$dihrs2[sleep$extra < 2.0 & sleep$extra > -2.0] <- "Small Change"      ## Recode level 1
sleep$dihrs2[sleep$extra == 2.0] <- "Large Change"                     ## Recode level 2
sleep$dihrs2[sleep$extra == -2.0] <- "Large Change"                    ## Recode level 2
sleep$dihrs2[sleep$extra > 2.0] <- "Large Change"                      ## Recode level 2
sleep$dihrs2[sleep$extra < -2.0] <- "Large Change"                     ## Recode level 2
```

- The first block of code categorizes numeric responses into a three level variable indicating whether the drug caused an increase, decrease, or no change in the number of hours the patient slept.
- The second block of code categorizes numeric responses into a two level category indicating either a large change or a small change.

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Categorizing Continuous Data

	extra	group	ID	dihrs	dihrs2
1	0.7	1	1	Increased	Small Change
2	-1.6	1	2	Decreassed	Small Change
3	-0.2	1	3	Decreassed	Small Change
4	-1.2	1	4	Decreassed	Small Change
5	-0.1	1	5	Decreassed	Small Change
6	3.4	1	6	Increased	Large Change
7	3.7	1	7	Increased	Large Change
8	0.8	1	8	Increased	Small Change
9	0.0	1	9	No Change	Small Change
10	2.0	1	10	Increased	Large Change
11	1.9	2	1	Increased	Small Change
12	0.8	2	2	Increased	Small Change
13	1.1	2	3	Increased	Small Change
14	0.1	2	4	Increased	Small Change
15	-0.1	2	5	Decreassed	Small Change
16	4.4	2	6	Increased	Large Change
17	5.5	2	7	Increased	Large Change
18	1.6	2	8	Increased	Small Change
19	4.6	2	9	Increased	Large Change
20	3.4	2	10	Increased	Large Change

- To double check your recode you could use the tapply() function to summarize the original variable by your new groups.

```
tapply(sleep$extra, sleep$dihrs2, summary)
```

```
$`Large Change`
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  2.000   3.400   3.700   3.857   4.500   5.500

$`Small Change`
    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
-1.6000 -0.1000  0.1000  0.2923  0.8000  1.9000
```

- As you can see the min and max of the "Small Change" category are within our expected values.

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Combining Variables

- In the code below we collapse the two variables we just created into a single variable:

```
sleep1dihrs3 <- NA                                                    ##Create New Variable
sleep1dihrs3[sleep1dihrs2 == "Large Change" & sleep1dihrs == "Increased"] <- "Large Increase"
sleep1dihrs3[sleep1dihrs2 == "Small Change" & sleep1dihrs == "Increased"] <- "Small Increase"
sleep1dihrs3[sleep1dihrs2 == "Large Change" & sleep1dihrs == "Decreased"] <- "Large Decrease"
sleep1dihrs3[sleep1dihrs2 == "Small Change" & sleep1dihrs == "Decreased"] <- "Small Decrease"
sleep1dihrs3[sleep1dihrs2 == "Large Change" &  sleep1dihrs == "No Change"] <- "No Change"
sleep1dihrs3[sleep1dihrs2 == "Small Change" &  sleep1dihrs == "No Change"] <- "No Change"
```

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Combining Variables

- You could check your results using the CrossTable() function in the gmodels package:

```
CrossTable(sleep1dihrs3, sleep1dihrs, prop.r = FALSE, prop.c = FALSE, prop.chisq = FALSE, prop.t = FALSE)
CrossTable(sleep1dihrs3, sleep1dihrs2, prop.r = FALSE, prop.c = FALSE, prop.chisq = FALSE, prop.t = FALSE)
```

sleep1dihrs3	sleep1dihrs2		
	Large Change	Small Change	Row Total
Large Increase	7	0	7
No Change	0	1	1
Small Decrease	0	5	5
Small Increase	0	7	7
Column Total	7	13	20

sleep1dihrs3	sleep1dihrs			
	Decreased	Increased	No Change	Row Total
Large Increase	0	7	0	7
No Change	0	0	1	1
Small Decrease	5	0	0	5
Small Increase	0	7	0	7
Column Total	5	14	1	20

	extra	group	ID	dihrs	dihrs2	dihrs3
1	0.7	1	1	Increased	Small Change	Small Increase
2	-1.6	1	2	Decreased	Small Change	Small Decrease
3	-0.2	1	3	Decreased	Small Change	Small Decrease
4	-1.2	1	4	Decreased	Small Change	Small Decrease
5	-0.1	1	5	Decreased	Small Change	Small Decrease
6	3.4	1	6	Increased	Large Change	Large Increase
7	3.7	1	7	Increased	Large Change	Large Increase
8	0.8	1	8	Increased	Small Change	Small Increase
9	0.0	1	9	No Change	Small Change	No Change
10	2.0	1	10	Increased	Large Change	Large Increase
11	1.9	2	1	Increased	Small Change	Small Increase
12	0.8	2	2	Increased	Small Change	Small Increase
13	1.1	2	3	Increased	Small Change	Small Increase
14	0.1	2	4	Increased	Small Change	Small Increase
15	-0.1	2	5	Decreased	Small Change	Small Decrease
16	4.4	2	6	Increased	Large Change	Large Increase
17	5.5	2	7	Increased	Large Change	Large Increase
18	1.6	2	8	Increased	Small Change	Small Increase
19	4.6	2	9	Increased	Large Change	Large Increase
20	3.4	2	10	Increased	Large Change	Large Increase

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

1.6. DATA VISUALIZATIONS

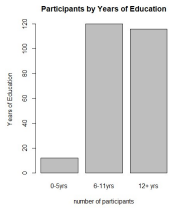
- Bar Charts
- Histograms
- Box Plots
- Scatter Plots
- Line Graphs

Data Visualizations

- Most plots can be generated using either
  - **Base R**
    or
  - **ggplot**
- Because the R community is quite active, you can usually figure out how to do plots by googling "How to make a _____ in r."
  - For example, I found the following two guides – both of which are quite useful:
    - Base R - https://www.statmethods.net/graphs/index.html
    - Ggplot - http://www.sthda.com/english/wiki/ggplot2-barplots-quick-start-guide-r-software-and-data-visualization

Bar Charts

- A bar chart or bar graph is a chart or graph that presents **categorical data** with rectangular bars with heights or lengths proportional to the values that they represent. The bars can be plotted vertically or horizontally.

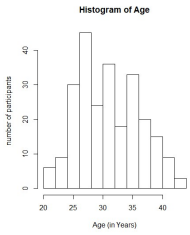



```
counts<-table(infert$education)
barplot(counts,
        main = "Participants by Years of Education",
        xlab = "number of participants", ylab = "Years of Education")
```

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Histograms

- A histogram is a plot that lets you discover, and show, the underlying frequency distribution (shape) of a set of **continuous data**.



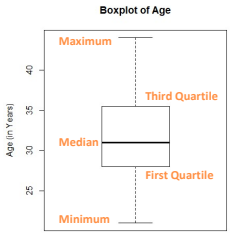

```
hist(infert$age,
     main = "Histogram of Age",
     xlab = "Age (in Years)", ylab = "number of participants")
```

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Box Plots

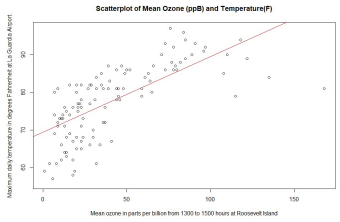
- The box plot (a.k.a. box and whisker diagram) is a standardized way of displaying the **distribution of continuous data** based on the five number summary: minimum, first quartile, median, third quartile, and maximum.

```
boxplot(infert$age,
      main = "Boxplot of Age",
      ylab = "Age (in Years)")
```



R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Scatter Plots



- A scatter plot is a type of plot or mathematical diagram using X & Y coordinates to display values from **two continuous variables** within a dataset – often as a means to highlight a **correlation**.

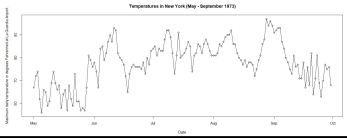
```
plot(airquality$Ozone, airquality$Temp,
    main = "Scatterplot of Mean Ozone (ppB) and Temperature(F)",
    xlab = "Mean ozone in parts per billion from 1300 to 1500 hours at Roosevelt Isla
    ylab = "Maximum daily temperature in degrees Fahrenheit at La Guardia Airport.")
abline(lm(airquality$Temp~airquality$Ozone), col="red") # linear regression line
```

R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Line Graphs

- A line graph, also known as a line chart, is a type of chart used to visualize the value of a **continuous variable over time**.

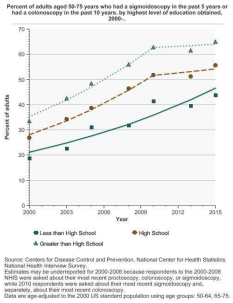
```
#Create A Year Variable Because the Air Quality Dataset Does Not Have this
airquality$Year <- 1973 ##All observations were in 1973

#Combine the Month Day and Year variables using the "paste" and "as.Date" functions.
airquality$date <- as.Date(paste(month.abb[airquality$Month],
                                 airquality$Day ,
                                 airquality$Year, sep="-"),
                           format = "%b-%d-%Y")
#Create a Plot
plot(airquality$date, airquality$Temp, type = "o",
     ylab = "Maximum daily temperature in degrees Fahrenheit at La Guardia Airport",
     xlab = "Date",
     main = "Temperatures in New York (May - September 1973)")
```



R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data

Essential Elements of a Good Graph

- Every graph should have a
  - A descriptive title
  - A key or legend
  - Units for each axis.
  - Titles for each axis.
  - A citation for the data source.
- Often times it is helpful to highlight key values in a graph.
- Error bars are often helpful to highlight uncertainty in a graph.
- Fit lines are often helpful when displaying data with individual observations.
- Generally graphs should have high visual contrast.
- While use of color can enhance a graph – it may also reduce accessibility.
- Footnotes or plot descriptions should be used to clarify any misleading visual factors or to describe limitations in the data.



R Basics
Exploring Data
Summarizing Data
Merging Data Sets
Recoding Variables
Plotting Data