R, R Markdown, data frames, and categorical data

Overview

Quick review from last class

- Statistics concepts
- R as a calculator, objects, data types
- Functions

Continuation with R

- Vectors and packages

R Markdown

- Formatting
- Code Chunks

More R

- Data frames
- If there is time: Categorical data - statistics and plots

Any questions about anything?

Announcement: learning groups!

Huaze (Gary) is organizing learning groups where students can get together (independent of TAs) to work on the homework and other class projects.

If you are interested in being part of a learning group please sign up!

- A link to sign up will be put on Canvas and has been sent out as an announcement.
- Please sign up by 11:59pm on Thursday 9/5

Announcement: Short reading and homework 1

1. Please read the article The Big Lies People Tell in Online Dating and fill out a quick survey about the article

2. Also, it would be good to start on homework 1

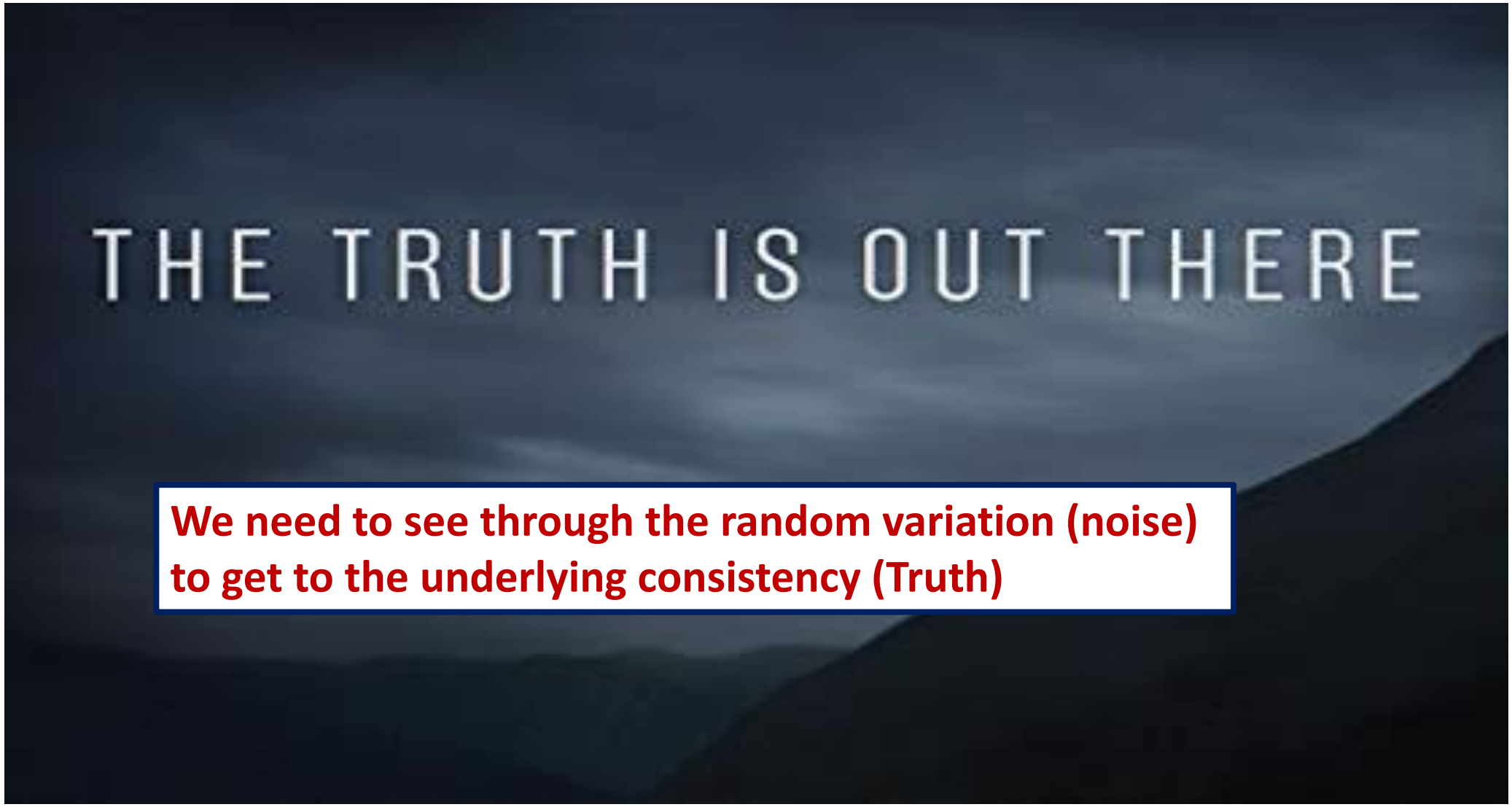
> SDS230::download_homework(1)

Homework 1 is due on Gradescope by 11pm on Sunday September 8th

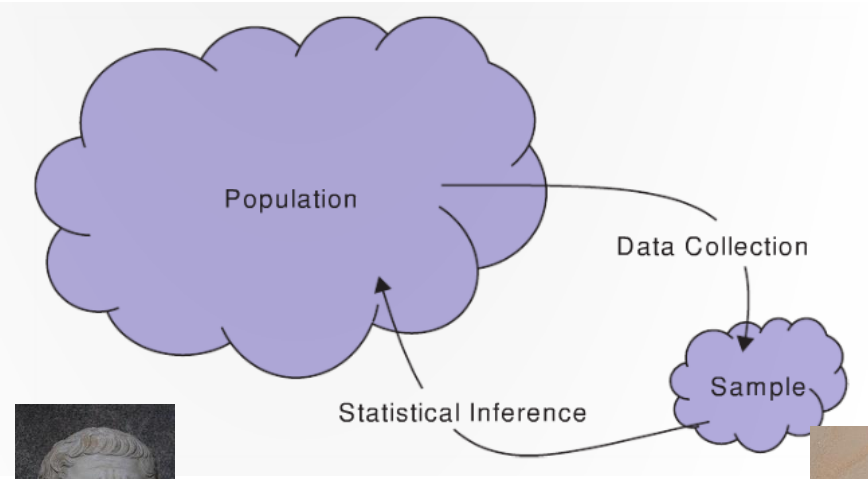
- Instructions for how to submit homework on Gradescope are on Canvas

QUICK
REVIEW

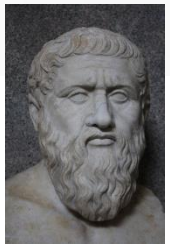
Quick Review of central concepts in Intro Statistics

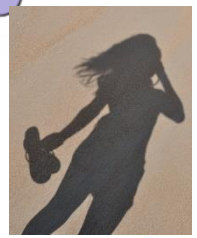


Parameters and statistics commonly used symbols



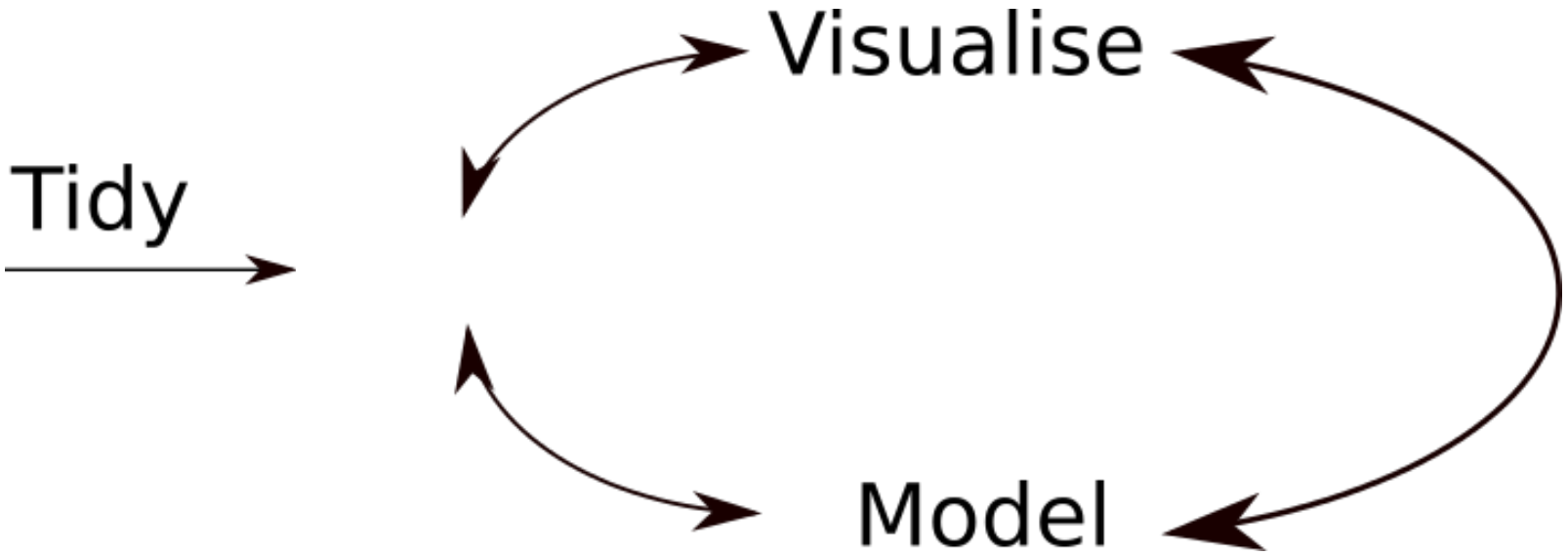
$$\bar{x} = \frac{\sum_i^n x_i}{n}$$

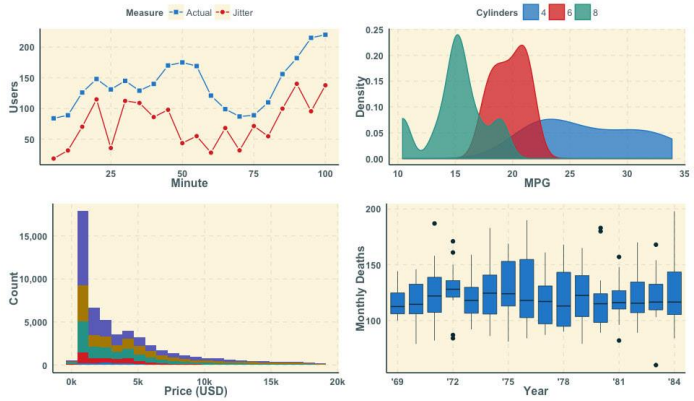




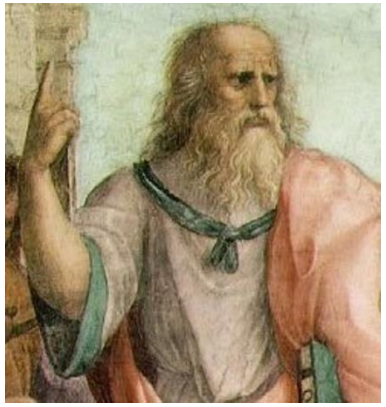
	Population parameter (Plato)	**Sample statistic (shadow)**
Mean		
Standard deviation		
Proportion		
Correlation		
Regression slope		

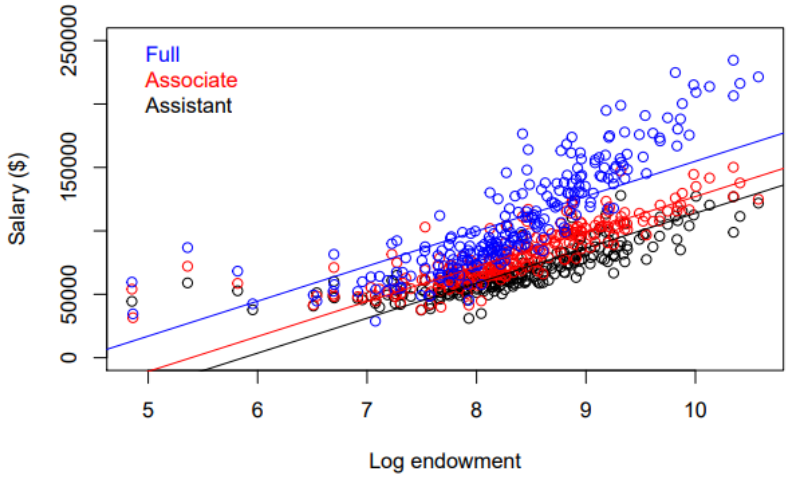
Sometimes the Truth is more complicated…





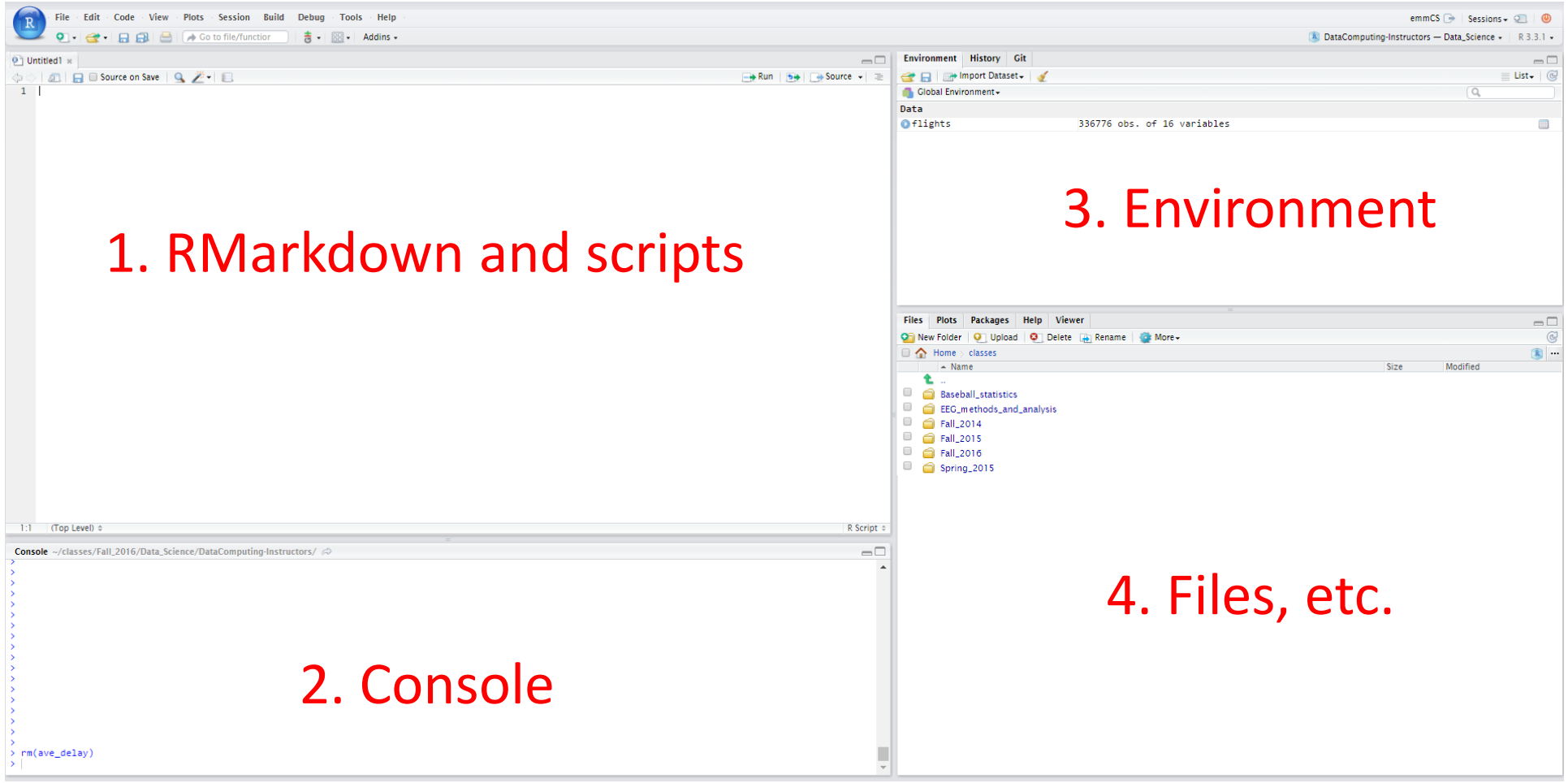
Curve information - Curve quality data												
Name	Formula	Slope at	Intercept	ED-20	ED-50	ED-80	Correlatic	Forced through origo				
Standard	Calc 1: C	standarc	standarc	3792394	27752	0.2	0.5	0.8	1	No		
Plate information												
Plate	Repeat	Barcode	Measurec	Chamber	Chamber	Humidity	Humidity	Ambient	Ambient	Formula	Measurement date	
1	1		N/A	N/A	N/A	N/A	N/A	N/A	N/A	Calc 1: C	standarc	standarc 10.12.2013 10:23:33
Background information												
Plate	Label	Result	Signal	Flashes/1	Measime	MeasInfo						
1	PicoGree	0	110307	10	0	De=1st Ex=Top Em=Top Wdw=N/A						
Calculate	standarc	standards on each plate) where Label: PicoGreenFilterTop(1) channel 1										
	1	2	3	4	5	6	7	8	9	10	11	12
A	-0.0011	-0.0011	-0.001	-0.001	-0.0011	-0.0012	-0.0011	-0.0011	-0.0012	-0.0012	0.9973	1.0026
B	0.0012	0.0014	0.0013	0.0012	0.0013	0.0012	0.0014	0.0003	-0.0011	-0.0011	0.0981	0.103
C	0.0016	0.0013	0.0013	0.0011	0.0012	0.0015	0.0016	-0.0004	-0.0011	-0.0011	0.0104	0.0095
D	0.0019	0.0024	0.0018	0.0015	-0.001	-0.001	-0.001	-0.001	-0.0011	-0.0011	0.0008	0.0009
E	-0.001	-0.0011	-0.0011	-0.0011	-0.001	-0.0012	-0.0011	-0.001	-0.0009	-0.0011	-0.0001	-0.0002
F	-0.001	-0.0011	-0.001	-0.001	-0.0012	-0.0011	-0.0011	-0.0009	-0.001	-0.001	-0.0003	-0.0002
G	-0.0011	-0.0011	-0.0011	-0.001	-0.001	-0.0012	-0.0011	-0.001	-0.001	-0.0011	-0.0002	0.0012
H	-0.0011	-0.0012	-0.0011	-0.001	-0.0011	-0.0011	-0.0012	-0.0011	-0.0011	-0.001	-0.0003	-0.0003





Questions?

Please open up RStudio



Review: R Basics

Arithmetic:

```
> 2 + 2
> 7 * 5
```

Assignment of values to ***objects***:

```
> a <- 4
> b <- 7
> z <- a + b
> z
[1] 11
```

Number journey…

Review: Character strings and Booleans

```
> a <- 7
> s <- "s is a terrible name for an object"
> b <- TRUE

> class(a)
[1] numeric

> class(s)
[1] character
```

Review: Functions

Functions use parenthesis: functionName(x)

```
> sqrt(49)
> tolower("DATA is AWESOME!")
```

To get help

```
> ? sqrt
```

One can add comments to your code

```
> sqrt(49)    # this takes the square root of 49
```

R continued…

Vectors

Vectors are ordered sequences of numbers or letters

The c() function is used to create vectors

```
> v <- c(5, 232, 5, 543)
> s <- c("statistics", "data", "science", "fun")
```

One can access elements of a vector using square brackets []

```
> s[4]        # what will the answer be?
```

We can get multiple elements from a vector too

```
> s[c(1, 2)]
```

Let’s continue in RStudio!

Vectors continued

One can assign a sequence of numbers to a vector

```
> z <- 2:10
> z[3]
```

One can test which elements are greater than a value

```
> z > 3
```

Can add names to vector elements

```
> names(v) <- c("first", "second", "third", "fourth")
```

Vectors continued

One can also apply functions to vectors

```
> z <- 2:10
> sqrt(z)
> mean(z)
```

Questions?

R packages

Packages add additional functionality to R

We will use many additional packages in this class

- gplyr, ggplot2, tidyr, etc.

There is a class specific package (SDS230) I wrote that you can use to download homework and other files

- All class materials are also on GitHub: https://github.com/emeyers/SDS230

Was everyone able to install the SDS230 package?

Downloading class 2 code

If you have the class SDS230 package, you can get code for today's class by typing the following commands at the console:

```
> library(SDS230)

> download_class_code(2)
```

R Markdown

R Markdown (.Rmd files) allow you to embed written descriptions, R code and the output of that code into a nice looking document

Creates a way to do reproducible research!

R Markdown

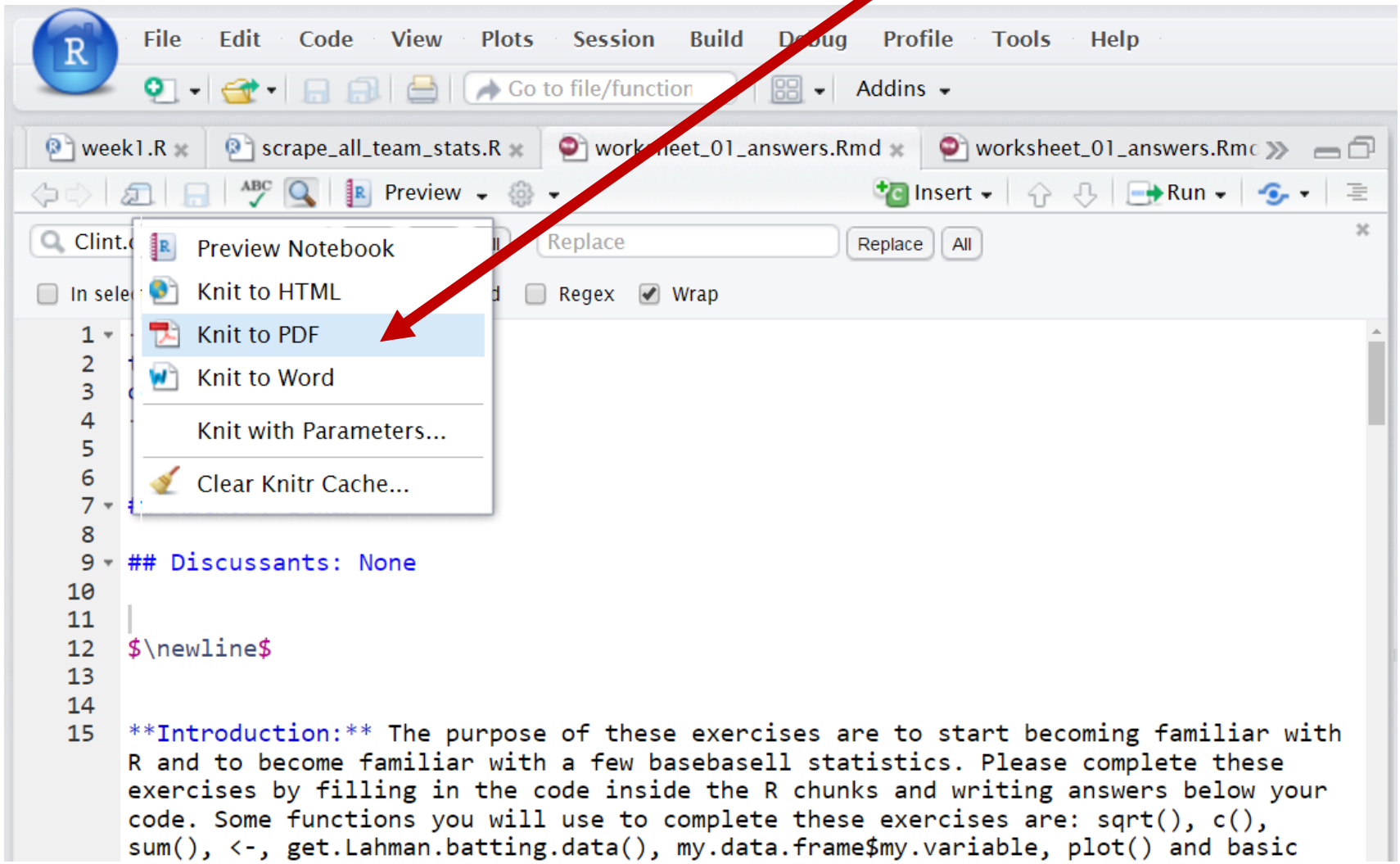
Everything in R chunks is executed as code:

```{r}
    # this is a comment
    # the following code will be executed
    2 + 3
```

Everything outside R chunks appears as text

Knitting to a pdf

Turn in a pdf or html document with your solutions to Canvas



R Markdown

Note: When you knit, RMarkdown files **do not have access to variables in the global environment**, but instead have their own environment.

Why is this a good thing???

Formatting in R Markdown

We can add formatting to text outside the code chunks

Examples:

```
## Level 2 header
**bold**
![](https://statistics.yale.edu/sites/default/files/logo2.png)
```

LaTeX in R Markdown

We can also add LaTeX symbols to documents using $\symbol$ syntax

For example, try these:

θ

$\hat{p}$

Knit early and knit often to avoid errors!!!

LaTeX in R Markdown

I have added a link on Canvas in the resources section to help find LaTeX symbols

How else could you get help to learn more about LaTeX symbols?

HELPDESK CAT SAYS
GOOGLE IS YOUR FRIEND!
memegenerator.net

To repeat: avoid hard to debug code!

Only change a few lines at a time and then knit your document to make sure everything is working!

If you document isn't knitting:

- **For code chunks:** use the # symbol to comment out code until you can find the line of code that is giving the error message
- **Outside of code chunk:** cut out part of the document until it knits and then paste it back

Announcement: Homework 1

Due Sunday September 8th at 11pm

- I recommend getting started early on this!

To download the homework please do the following:

> library(SDS230)

> download_homework(1)

From the file panel, open the homework and try knitting it

Announcement: Homework 1

Instructions for how to submit homework on Gradescope are on Canvas

- Please mark all pages that answers correspond to on Gradescope!

Be sure to also "show your work" by printing out any values you report

- Although don't print out hundreds of access pages of numbers

Ask/answer questions on Ed Discussions, but don't give away the solutions!

Questions?

Data frames

Data frames contain structured data

	age	body_type	diet	drinks	drugs	education
1	22	a little extra	strictly anything	socially	never	working on college/university
2	35	average	mostly other	often	sometimes	working on space camp
3	38	thin	anything	socially	*NA*	graduated from masters program
4	23	thin	vegetarian	socially	*NA*	working on college/university
5	29	athletic	*NA*	socially	never	graduated from college/university
6	29	average	mostly anything	socially	*NA*	graduated from college/university

OK Cupid data

Back to R: Data frames

Data frames contain structured data

```
> library(SDS230)
> download_data("profiles_revised.csv")     #  only needs to be run once
> profiles <- read.csv("profiles_revised.csv")
> View(profiles)       # the View() function only works in R Studio!
```

	age	body_type	diet	drinks	drugs	education
1	22	a little extra	strictly anything	socially	never	working on college/university
2	35	average	mostly other	often	sometimes	working on space camp
3	38	thin	anything	socially	*NA*	graduated from masters program
4	23	thin	vegetarian	socially	*NA*	working on college/university
5	29	athletic	*NA*	socially	never	graduated from college/university
6	29	average	mostly anything	socially	*NA*	graduated from college/university

Data Frames

Variables

Cases

	age	body_type	diet	drinks	drugs	education
1	22	a little extra	strictly anything	socially	never	working on college/university
2	35	average	mostly other	often	sometimes	working on space camp
3	38	thin	anything	socially	*NA*	graduated from masters program
4	23	thin	vegetarian	socially	*NA*	working on college/university
5	29	athletic	*NA*	socially	never	graduated from college/university
6	29	average	mostly anything	socially	*NA*	graduated from college/university

An Example Dataset

Quantitative Variable

Categorical Variable

Cases (observational units)

	age	body_type	diet	drinks	drugs	education
1	22	a little extra	strictly anything	socially	never	working on college/university
2	35	average	mostly other	often	sometimes	working on space camp
3	38	thin	anything	socially	*NA*	graduated from masters program
4	23	thin	vegetarian	socially	*NA*	working on college/university
5	29	athletic	*NA*	socially	never	graduated from college/university
6	29	average	mostly anything	socially	*NA*	graduated from college/university

Data frames

We can extract the columns of a data frame as vector objects using the $ symbol

```
> the_ages <- profiles$age
```

Can you get the mean() age of users in this data set?

```
> mean(the_ages)
```

Let's continue in RStudio!

Extracting rows from a data frame

We can extract rows from a data frame in a similar way as extracting values from a vector by using the square brackets

```
> profiles[1, ]  # returns the first row of the data frame
> profiles[, 1]  # returns the first column of the data
```

Note, the first column of the profiles data frame is the variable *age*, so we can also get the first column using:

```
> profiles$age  # this is the same as profiles[, 1]
```

Extracting rows from a data frame

We can also create vectors of numbers or Booleans specifying which rows we want to extract from a data frame

```
# create a vector with the numbers 1, 10, 20
> my_vec <- c(1, 10, 20)

# use my_vec to get the 1st, 10th, and 20th row in profiles
> small_profiles <- profiles[my_vec, ]
> dim(small_profiles)  # number of rows and columns in the data frame
```

Extracting rows from a data frame

Finally, we can also extract rows by creating a Boolean vector that is of the same length as the number of rows in the data frame

TRUE values will be extracted from the data frame while FALSE values will not

```
# create a vector of booleans
> my_bools <- c(TRUE, FALSE, TRUE)

# use the Boolean vector to get the 1st and 3rd row
> small_profiles[my_bools, ]
```

Questions?

Categorical data

Categorical variables

What is a categorical variable?

- A: A categorical variable assigns each observation to one of *k* groups

Which variables in the profiles data frame are categorical?

- Is heights a categorical variable?

For categorical variables, we usually want to view:

- How many items are each category OR
- The proportion (or percentage) of items in each category

$$\text{Proportion in a category} = \frac{\text{number in that category}}{\text{total number}}$$

Categorical data

```
# Get information about drinking behavior
> drinking_vec <- profiles$drinks

# Create a table showing how often people drink
> drinks_table <- table(drinking_vec)
> drinks_table
```

Relative frequency table

We can create a relative frequency table using the function:

```
> prop.table(my_table)
```

Can you create a relative frequency table for the drinking behavior of the people in the okcupid data set?

```
> drinks_table <- table(profiles$drinks)
> prop.table(drinks_table)
```

What is the proper statistical notation for these values: $\hat{p}$ or π ?

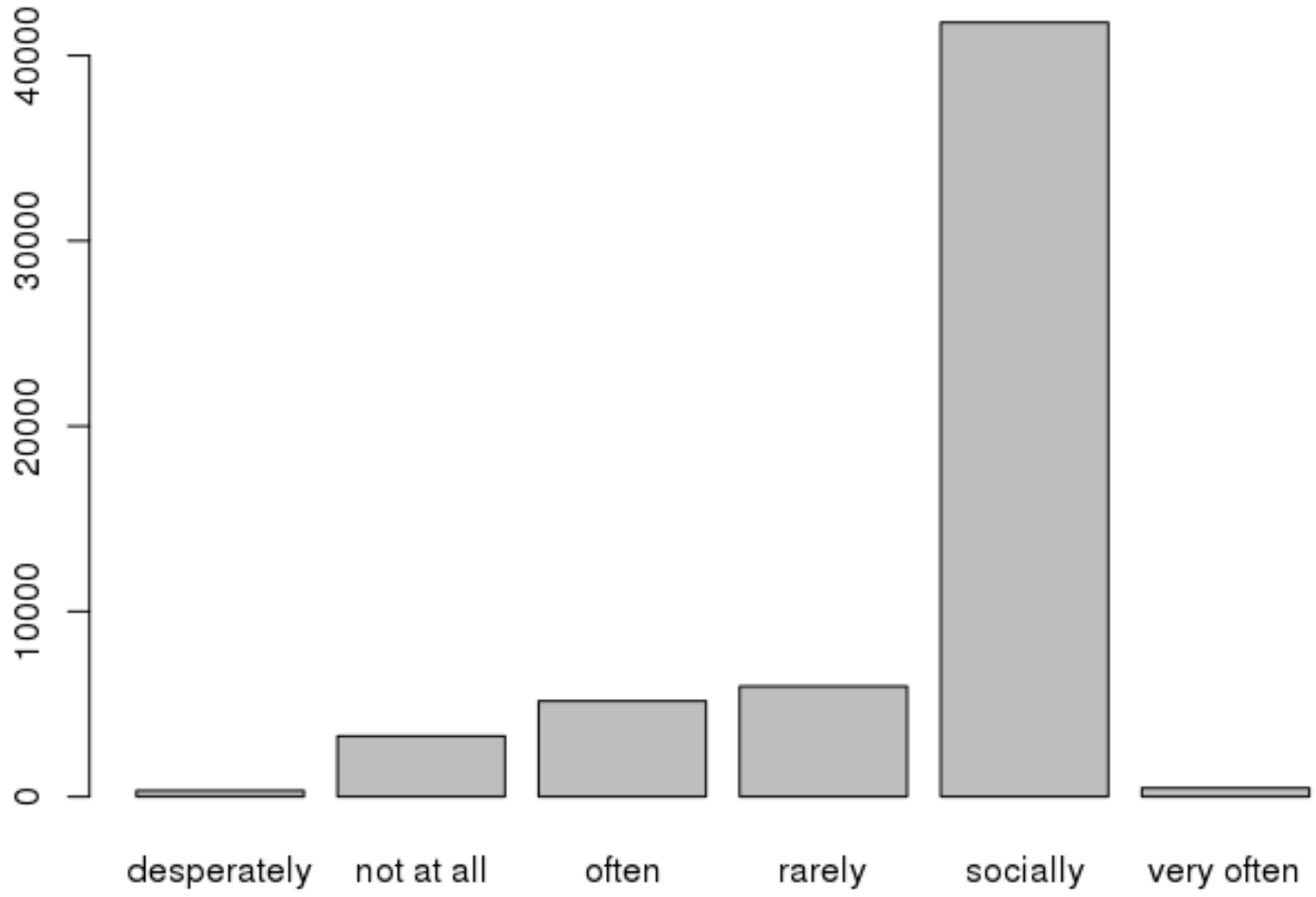
Bar plots (pun intended?)

We can plot the number of items in each category using a bar plot

```
> barplot(my_table)
```

Can you create a bar plot for the drinking behavior of the people in the okcupid data set?

```
> drinks_table <- table(profiles$drinks)
> barplot(drinks_table)
```



What is wrong with this plot?

Details matter!

Can you figure out how to label the axes?

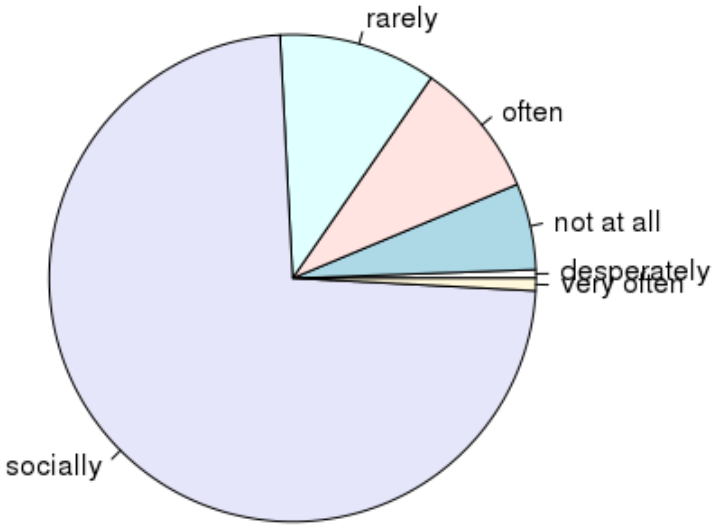
- A: ? barplot
- A: xlab and ylab!

```
> barplot(drinks_table,
        ylab = "Count",
        xlab = "Type of drinker",
        main = "Counts of different types of drinkers")
```

Pie charts

We can also use the pie() function to create pie charts

> pie(drinks_table)



Which is best: bar plots or pie charts?

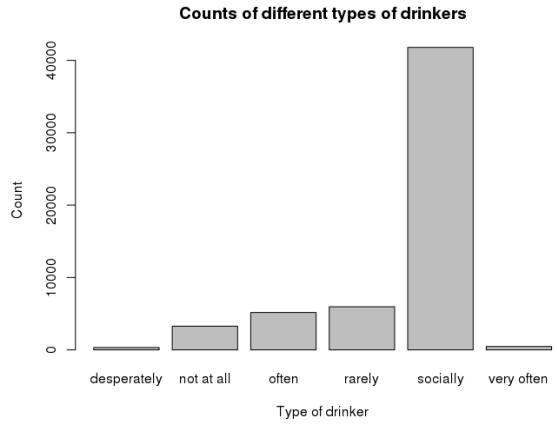
```
> barplot(table(profiles$sex, useNA = "always"))
```

```
> pie(table(profiles$sex, useNA = "always"))
```

Q1: Is one better than the other?

Q2: Can you figure out how to add colors to these plots?

Removing social drinkers



Social drinkers are dominating our plot ☹

We can get rid of social drinkers by only plotting counts less than 10,000

```
> nonsocial_inds <- drinks_table < 10000
> nonsocial_drinks_table <- drinks_table[nonsocial_inds]
> barplot(nonsocial_drinks_table)
```

Questions?

For next class…

1. Please read the article The Big Lies People Tell in Online Dating and fill out a quick survey about the article

2. Also, it would be good to start on homework 1

> SDS230::download_homework(1)

Homework 1 is due on Gradescope by 11pm on Sunday September 8th

- Instructions for how to submit homework on Gradescope are on Canvas