Sampling and proportions

Overview

Questions about the DataCamp exercises

Sampling

- Parameters and statistics
- Proportions, frequency tables
- Bar and pie charts
- Sampling variation

Announcement: Mass mutual data science development program

There will be an information session about the Mass Mutual Data Science development program

- Work for Mass Mutual and get a free Master in Data Science from UMass

When: Monday, September 17th at 4pm

Where: Ash conference room (ASH 137)

Questions about the R DataCamp exercises?

How did they go?

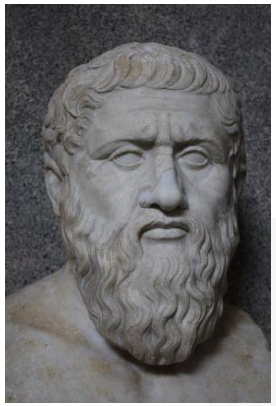
Remember: everything is cumulative, so you will need to use what you learned on the worksheet that is due on Sunday

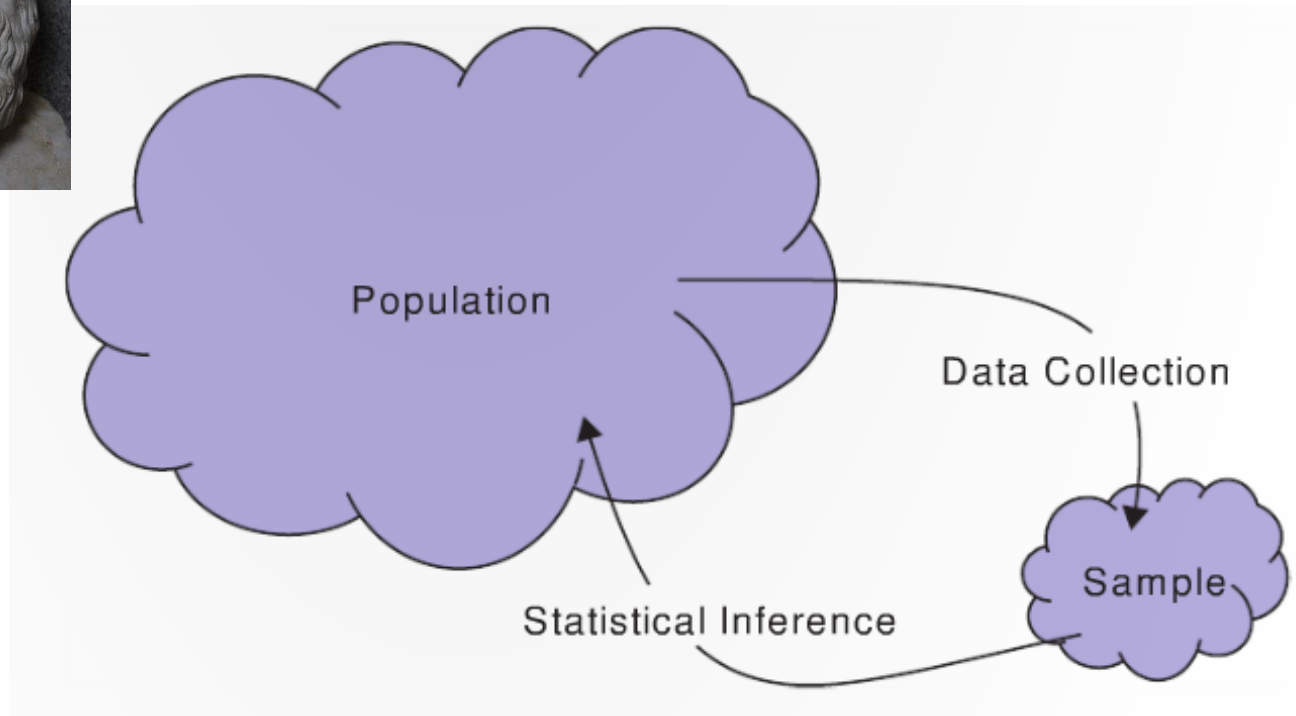
David can help anyone after class if they still have questions

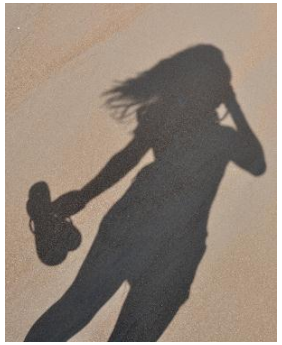
R Review

```
> a <- 7    # assigning a number to a variable
> s <- "hello everyone" # assigning a character string to a variable
> class(a)    # seeing what type of data is in a variable

> sqrt(49)  # functions
> ? sqrt  # getting help for a function (how else can we get help?)

> v  <-  c(5, 232, 5, 543)     # creating a vector with numbers
> v[3]     # extracting a value from a vector
```

Back to the central concepts in Statistics







Categorical variables

Where do samples/data come from?

In past classes I have had students try out sampling by counting 100 sprinkles…

1	orange
2	red
3	green
4	white
5	white
6	white
7	white
8	white
9	red

The ***sample size*** (n) is the number of items in the sample
What is ***n*** here?

Sampling example

Questions:

1) What are the observational units (cases)?

2) What is the variable?

3) Is the variable categorical or quantitative?

4) What is the population?

5) Do you think the samples we are getting are representative of the population?

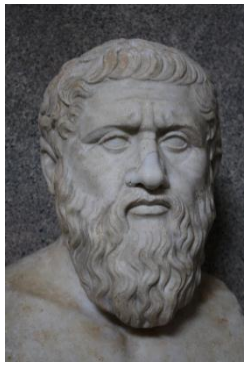
1	orange
2	red
3	green
4	white
5	white
6	white
7	white
8	white
9	red

Population parameters vs. sample statistics

A **statistic** is a number that is computed from ***data in a sample***

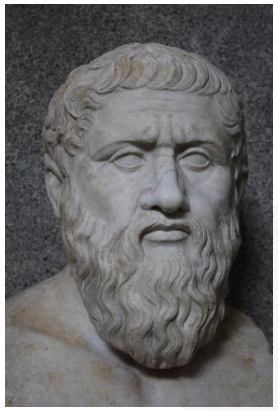
- Not to be confused with Statistics, which is a field of study

A **parameter** is a number that describes some aspect of a ***population***

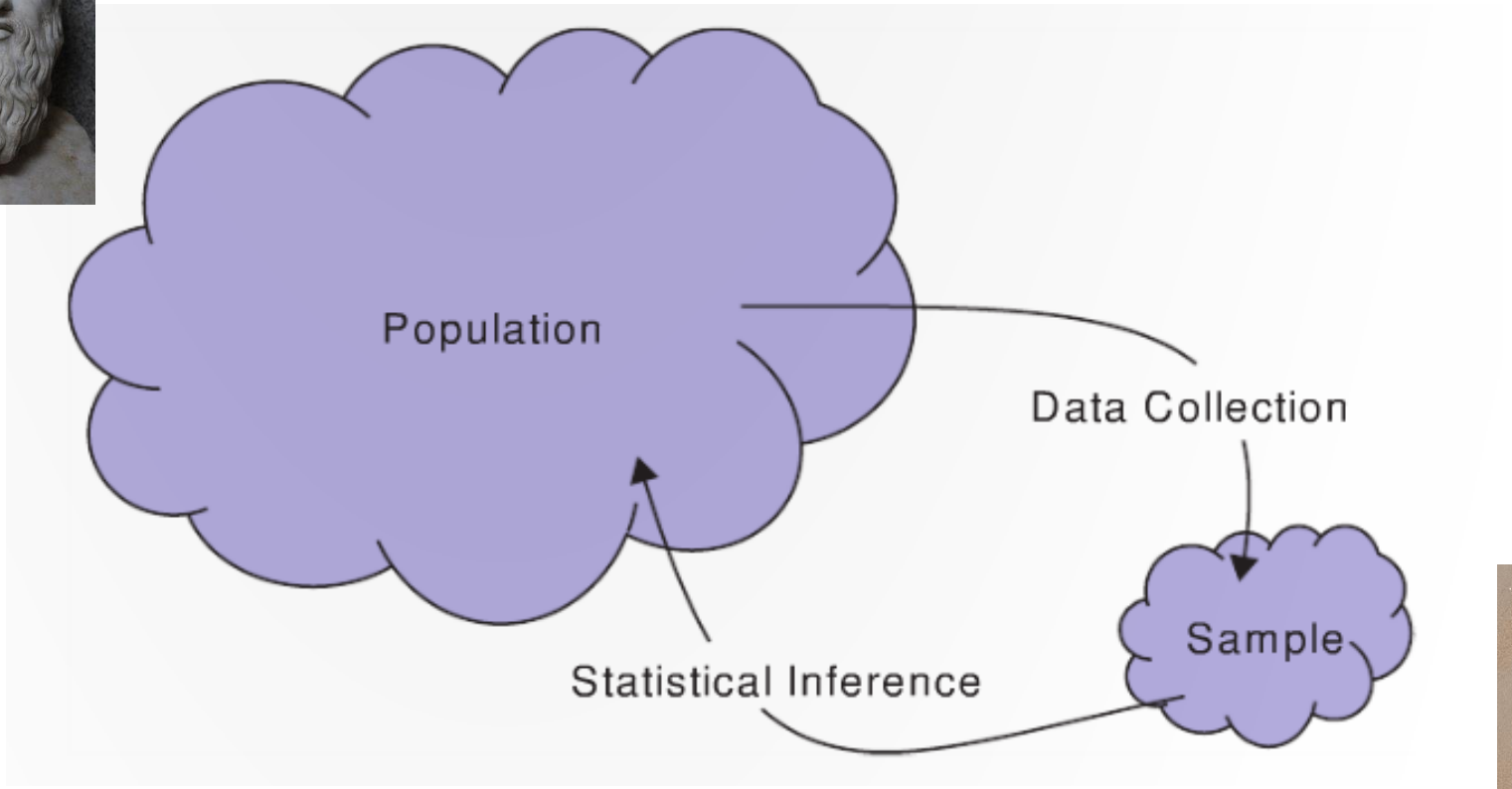


?

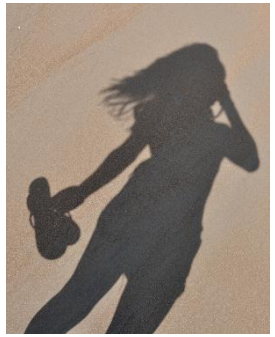
Parameters and statistics



parameters



statistics



Proportions

For a *single* ***categorical variable***, the main statistic of interest is the ***proportion*** in each category

- E.g., the proportion of red sprinkles

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

Example proportion of red sprinkles

The sample

- orange, red, green, white, white, white, …, pink

The proportion for a ***sample*** is denoted $\hat{p}$ (pronounced "p-hat")

- $\hat{p}_{red} = 13/100 = 0.13$

The proportion for a ***population*** is denoted π (the book uses p)

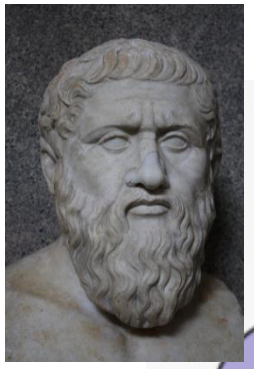
- π_{red} proportion if we had measured all sprinkles in the population

$\hat{p}$ is a ***point estimate*** of π

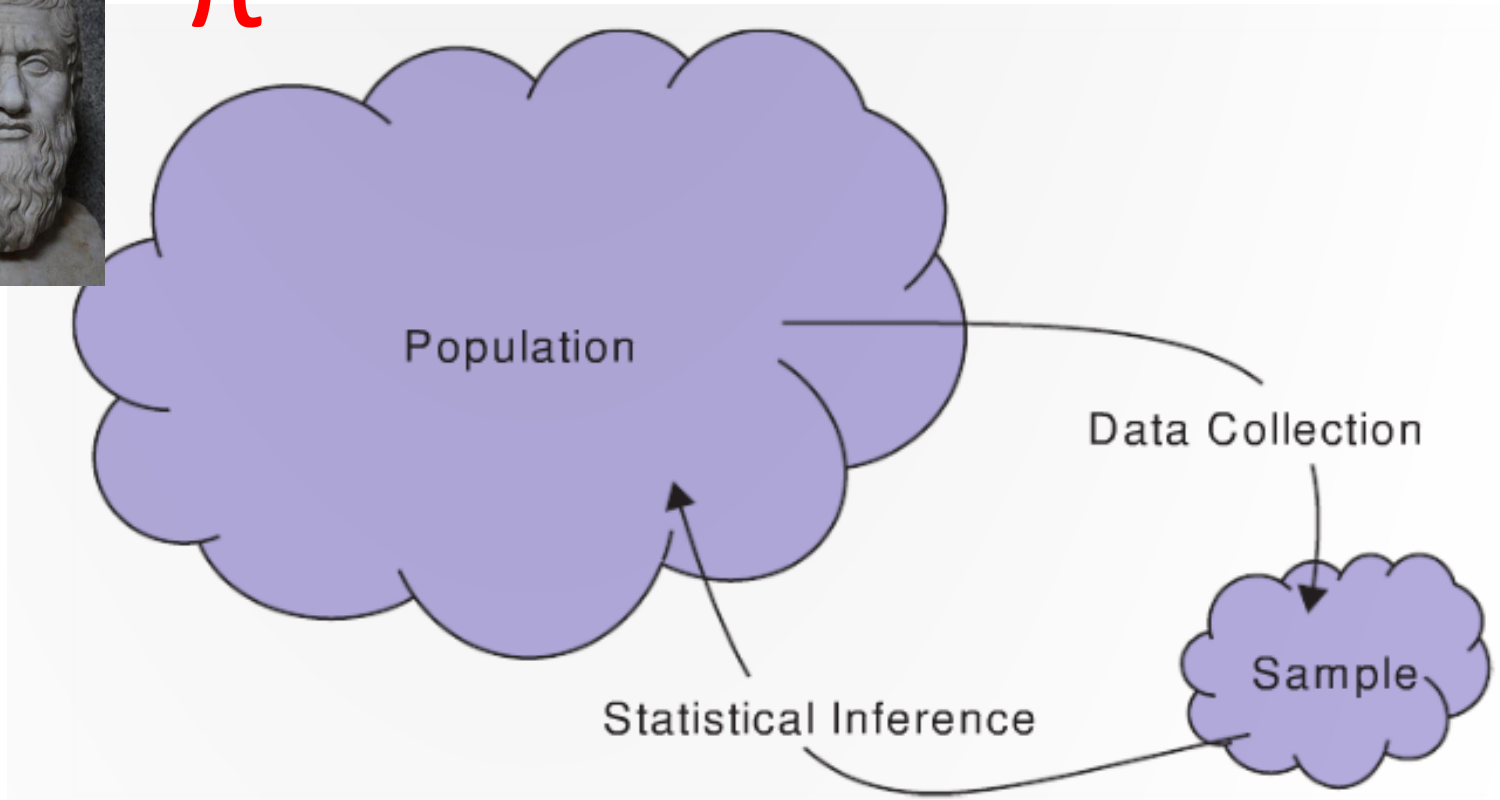
- i.e., $\hat{p}$ our best guess of what π is

Sample vs. Population proportion

parameter

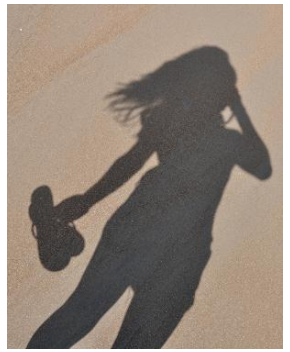


π



$\hat{p}$

statistic



Calculating counts on a categorical variable

The count of how many items are in each category can be summarized in a ***frequency table***

Color	green	orange	pink	red	white	yellow		Total
Count	20	11	9	13	36	11		100

In R:

```
> my_sample <- c("orange", "red", "green", "white", " white", ... )
> my_table <- table(my_sample)
```

```
R: > table(categorical_vector)
```

Calculating counts on a categorical variable

The count of how many items are in each category can be summarized in a ***frequency table***:

Color	green	orange	pink	red	white	yellow		Total
Count	20	11	9	13	36	11		100

In R:

```
> source("/home/shared/intro_stats/get_sprinkle_sample.R")
> my_sample <- get_sprinkle_sample(100)   # get a sample of 100 sprinkles
> my_table <- table(my_sample)
```

Calculating proportions (relative frequencies) on a categorical variable

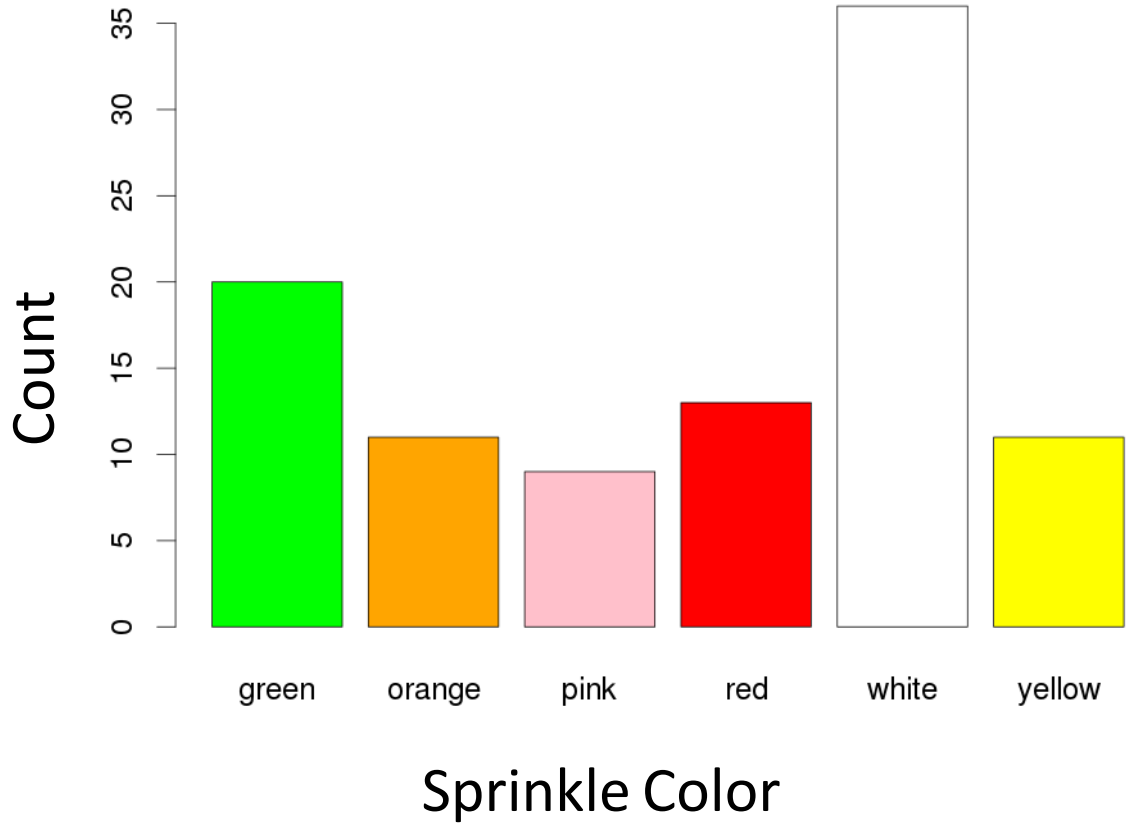
We can convert a frequency table into a relative frequency table by dividing each cell by the total number of items

Color	green	orange	pink	red	white	yellow		Total
Count	.20	.11	.09	.13	.36	.11		1

In R:

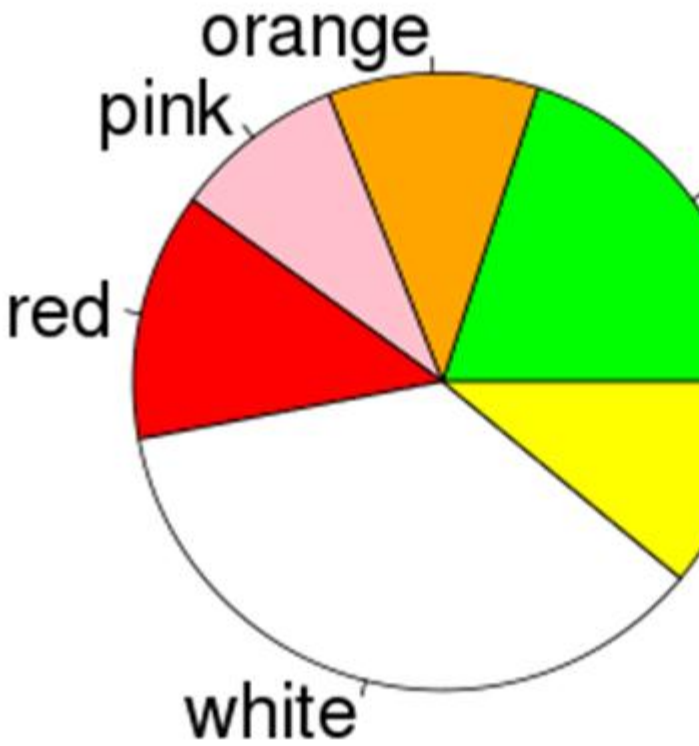
```
> my_table <- table(my_sample)
> prop.table(my_table)
```

Bar Chart



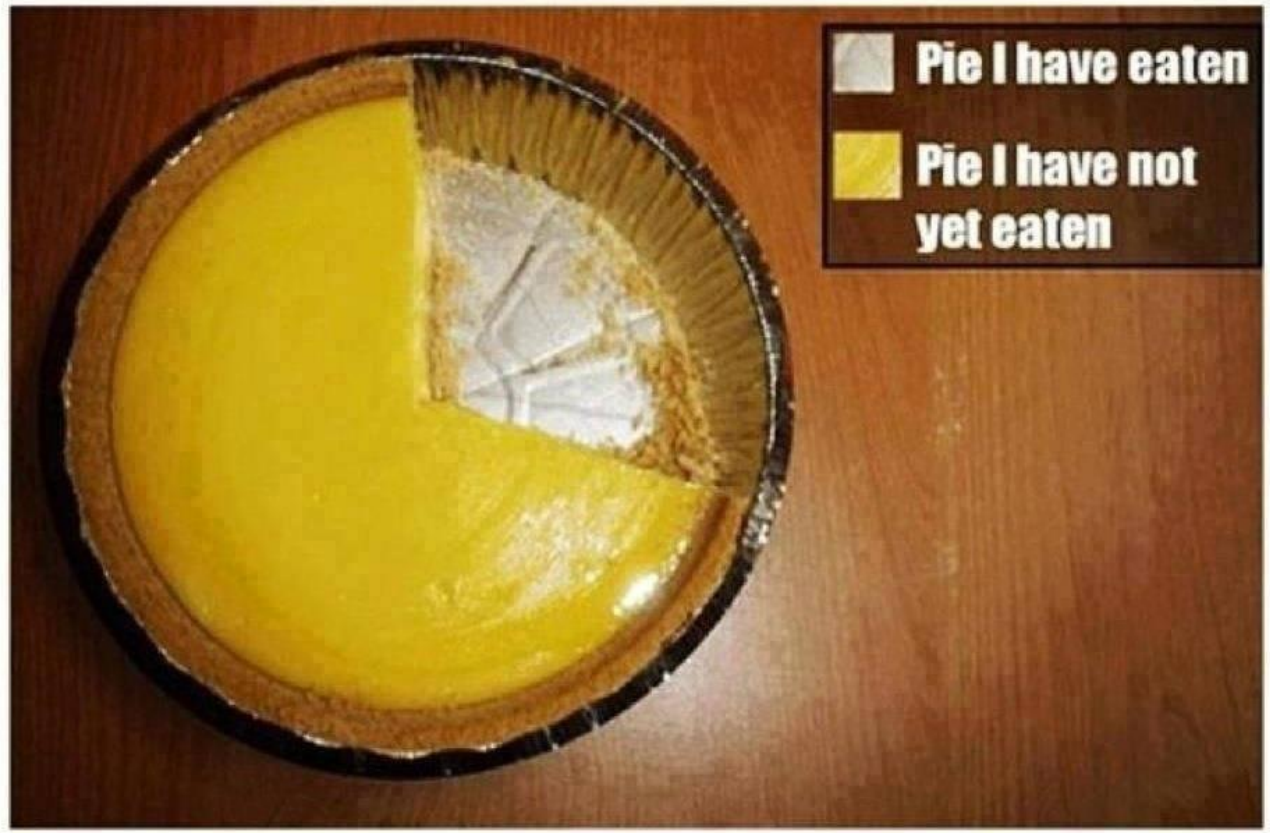

```
R: > barplot(my_table)
```

Pie Chart

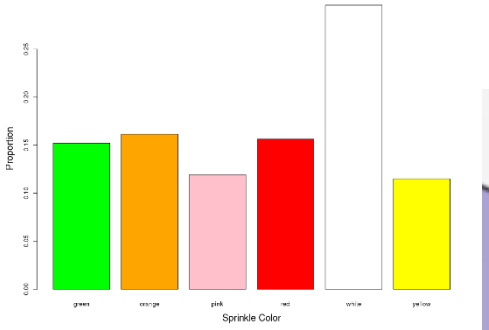



```
R: > pie(my_table)
```

World's Most Accurate Pie Chart



Summary: Sample and Population proportion

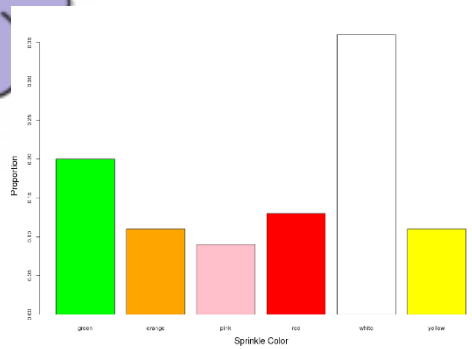


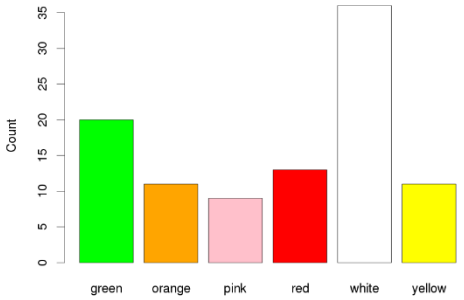
π_{red}

Categorical distribution

Bar chart

$\hat{p}_{red}$





π_{red}

$\hat{p}_{red} = 0.20$

$\hat{p}_{red} = 0.15$

$\hat{p}_{red} = 0.24$

Sprinkle samples in R

```
> source("/home/shared/intro_stats/get_sprinkle_sample.R")

# get a sample of 100 sprinkles
    > my_sample <- get_sprinkle_sample(100)

# Try calculating the proportion of red sprinkles several times…
    > sprinkle_table <- table(my_sample)
    > sprinkle_proportions <- prop.table(sprinkle_table)
    > sprinkle_proportions["red"]
```

Do you get the same $\hat{p}_{red}$ each time?
How much to the statistics (point estimates) vary?

Summary of concepts

1. A **statistic** is a number that is computed from ***data in a sample***

- The number of items in a sample is called the ***sample size*** and is usually denoted with the symbol n

2. A **parameter** is a number that describes some aspect of a ***population***

3. A point estimate is using a value of a statistic as a guess for the value of a parameter

4. When calculating proportions:

- The proportion statistic is denoted $\hat{\mathbf{p}}$
- The population proportion is denoted $\boldsymbol{\pi}$
- Thus $\hat{p}$ is a ***point estimate*** of π

5. Proportions can be summarized in a **relative frequency table** and can be visualized using **bar plots** and **pie charts**

Summary of R

```
# a vector of character strings (or factors)
my_sample <- c("orange", "red", "green", "white", " white", ... )

# creating a table using the table() function
my_table <- table(my_sample)

# creating a frequency table using the prop.table() function
prop.table(my_table)

# creating bar and pie charts
bar(my_table)
pie(my_table)
```

Worksheet 1!

1. Load some class specific functions using the code

```
> source("/home/shared/intro_stats/cs206_functions.R")
```

2. Go to the console and copy the worksheet using the following commands:

```
> get_worksheet(1)
```

RMarkdown

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

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

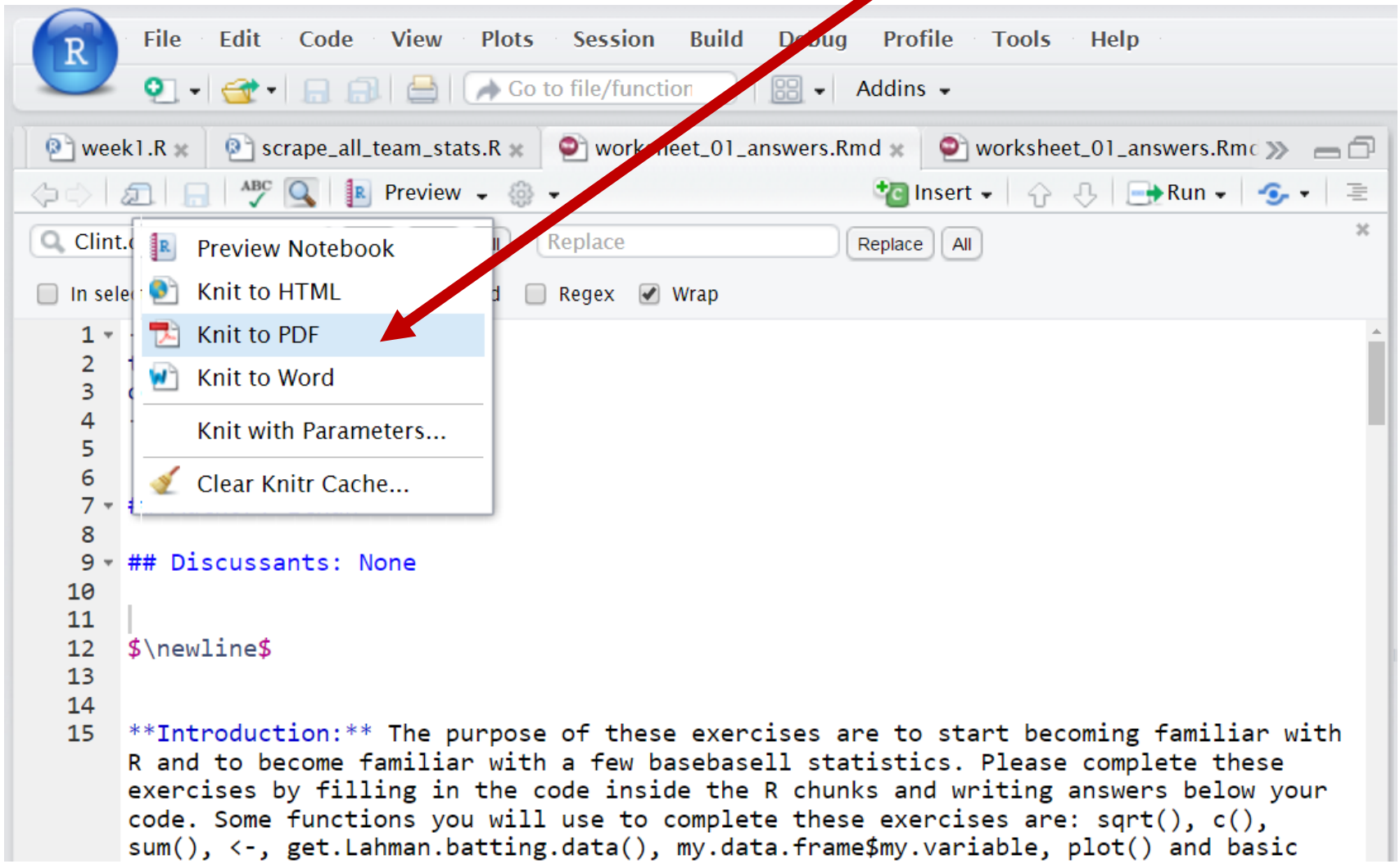
RMarkdown

Note: Rmarkdown files do not have access to variables in the global environment, but instead have their own environment.

Why is this a good thing???

Knitting to a pdf

Turn in a pdf of your solutions to Moodle



Avoid hard to debug code!

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

Comment out parts of the code that isn’t working (using the # symbol) until you can find the line of code that is giving the error message

Worksheet 1

Worksheet 1 is due at 11:55pm on Sunday September 16th

Use the #worksheet_01 channel on Slack to ask any questions that come up

- David or James, could you add everyone to this channel?