Class 3: - Missing Values, Functions, Group Variables, and Intro to ML

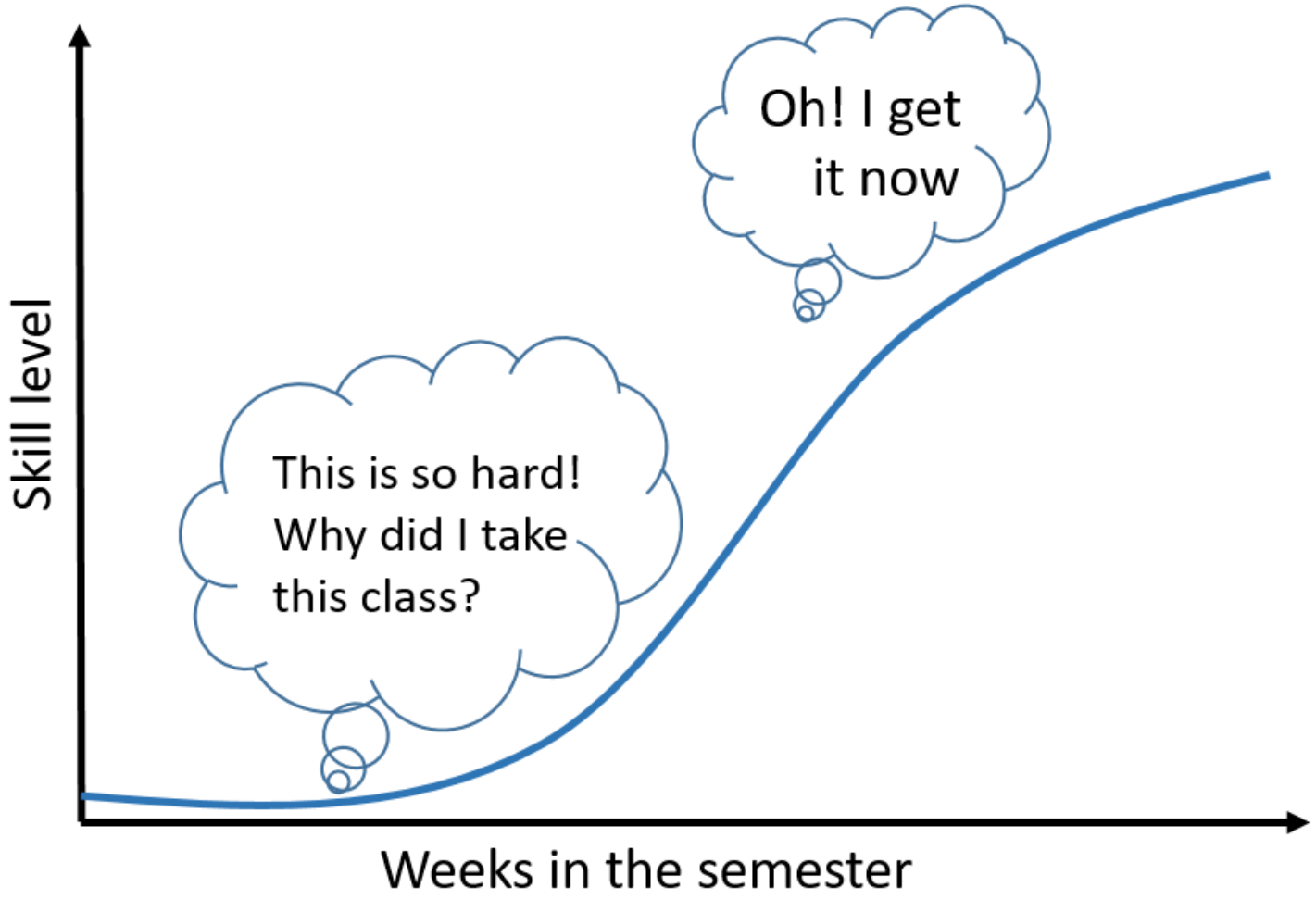
BUS 696

Prof. Jonathan Hersh

Class 3: Announcements

1. Data Analytics Week! October 5-9
2. Office Hours
   - TA: Wed 12-1; Thur 5-6
   - Mon 11-12noon; Wed 5-6;
3. Problem Set 1 posted – Due Sept 25
   - Must submit compiled HTML file using Rmarkdown
4. Posted Rmarkdown Homework Template
   - May use, don't have to

I apologize but we are at peak difficulty



Data Analytics Industry Week

Register on Handshake to get access to the following virtual events!

Data Analytics Accelerator Program Info Session
Monday, October 5 | 11 a.m. PST
Interested in pursuing a career in the growing field of data analytics? The Argyros School of Business is proud to present the new career skills-focused Analytics Accelerator Program. Learn more about what hard skills are needed to land a successful career in data analytics. Hear from Professor Toplansky and Dr. Hersh about how you can propel your success and prepare for 21st Century jobs that pay a premium.

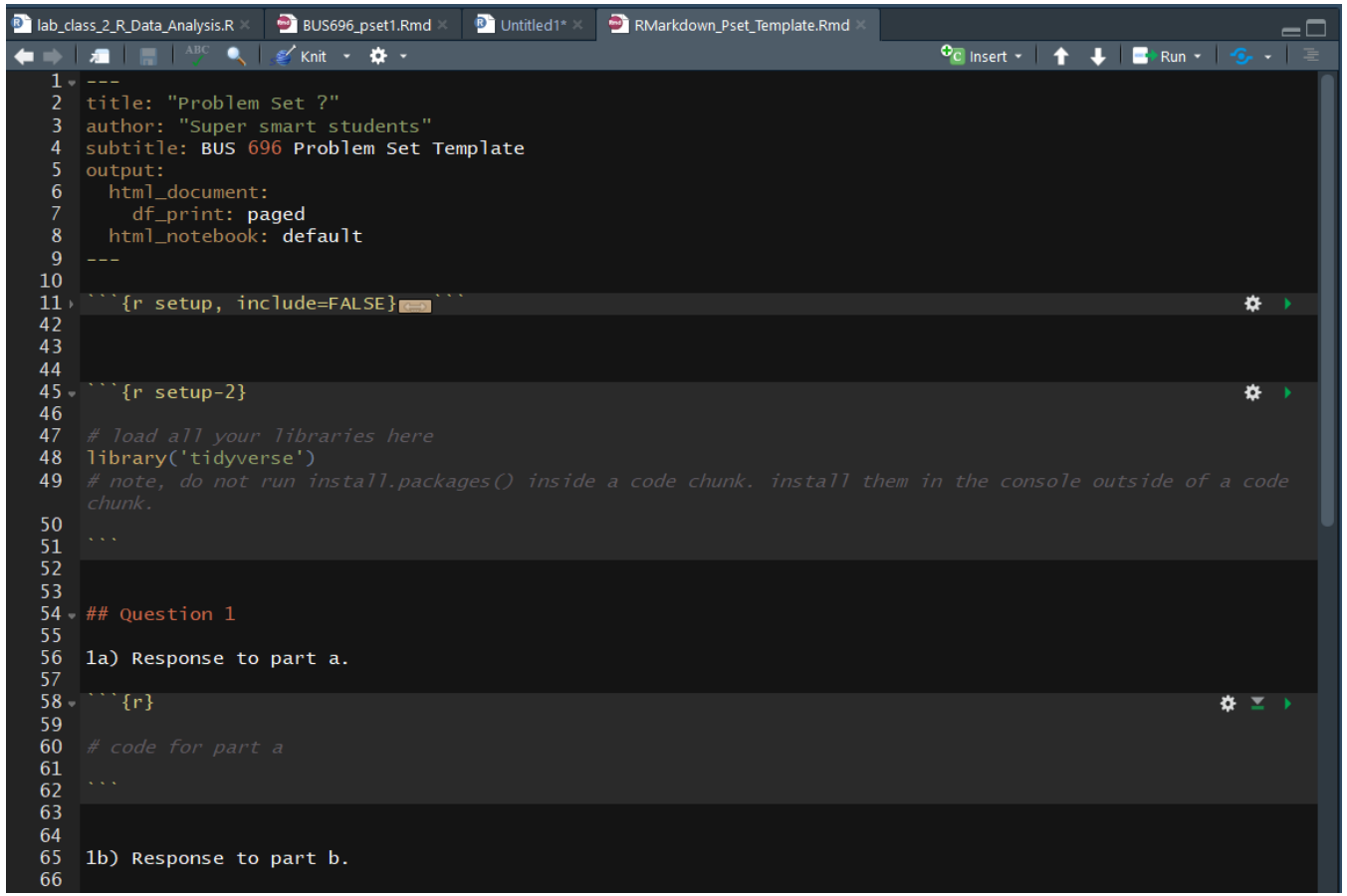
Careers in Data Analytics
Tuesday, October 6 | 12 p.m. PST
Hear from the renowned authors of Build a Career in Data Science, Jacqueline Nolis and Emily Robinson about careers in data analytics.

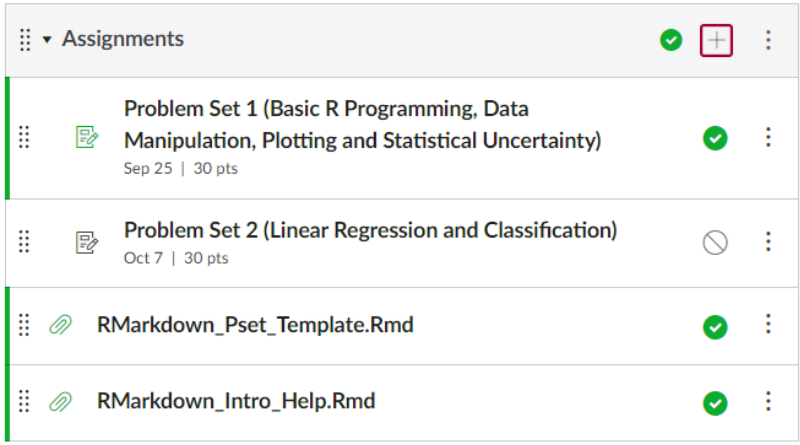
Data Analytics Industry Panel
Thursday, October 8 | 4:30 p.m. PST
This data analytics panel will feature industry experts in analytics from entertainment, healthcare, technology, and more.

Entertainment Analytics: Turning Data Into Insights
Friday, October 9| 12 p.m. PST
Come see a live demo and learn about turning data into actionable insights in Entertainment Analytics with Andre Vargas Head of the data department at leading entertainment and sports agency, Creative Artists Agency (CAA).

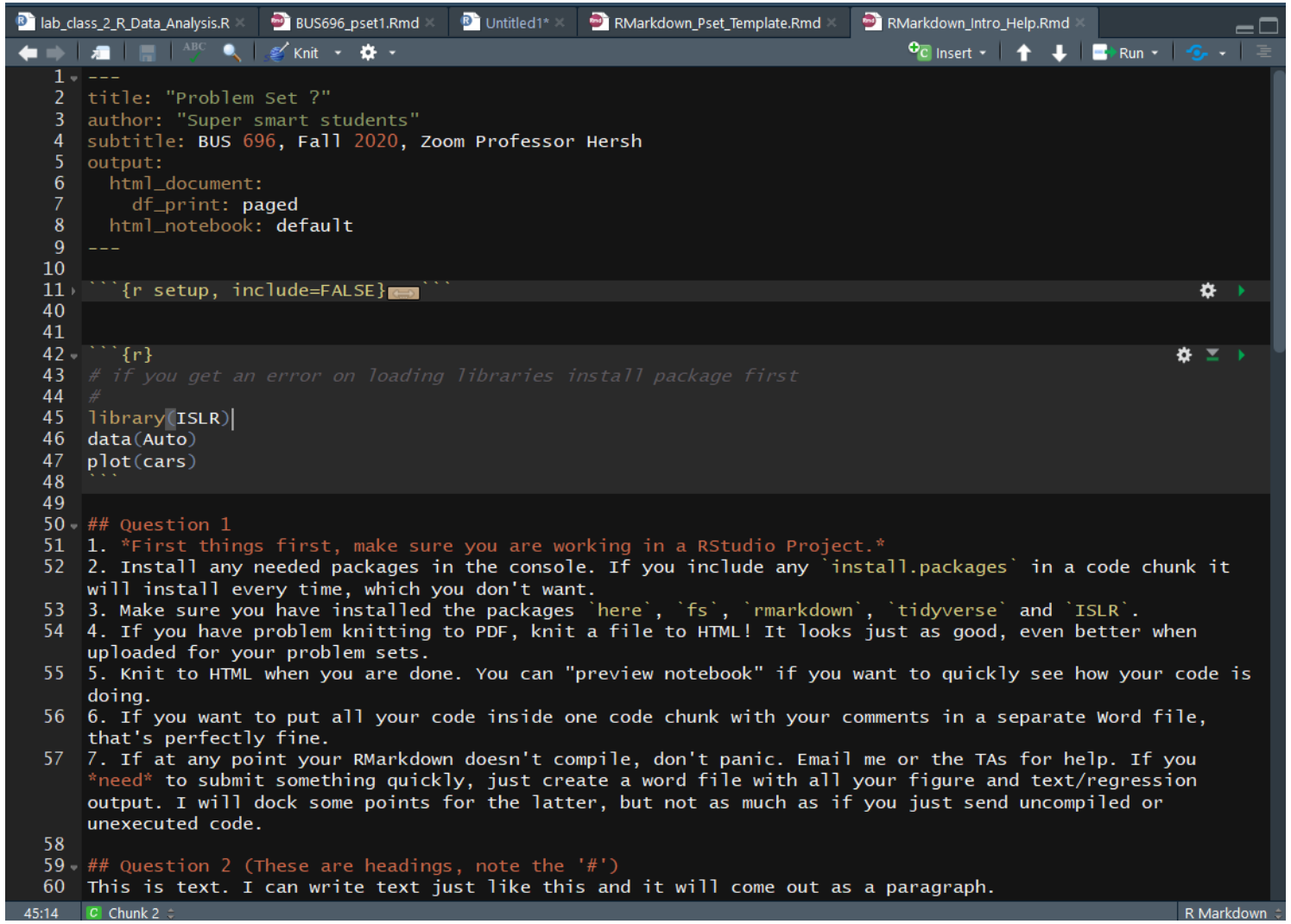
CHAPMAN UNIVERSITY | Argyros School of Business and Economics

May Use Problem Set Rmarkdown Template In Assignments Module





IF You Want to Learn More About Rmarkdown Go Through Rmarkdown Help

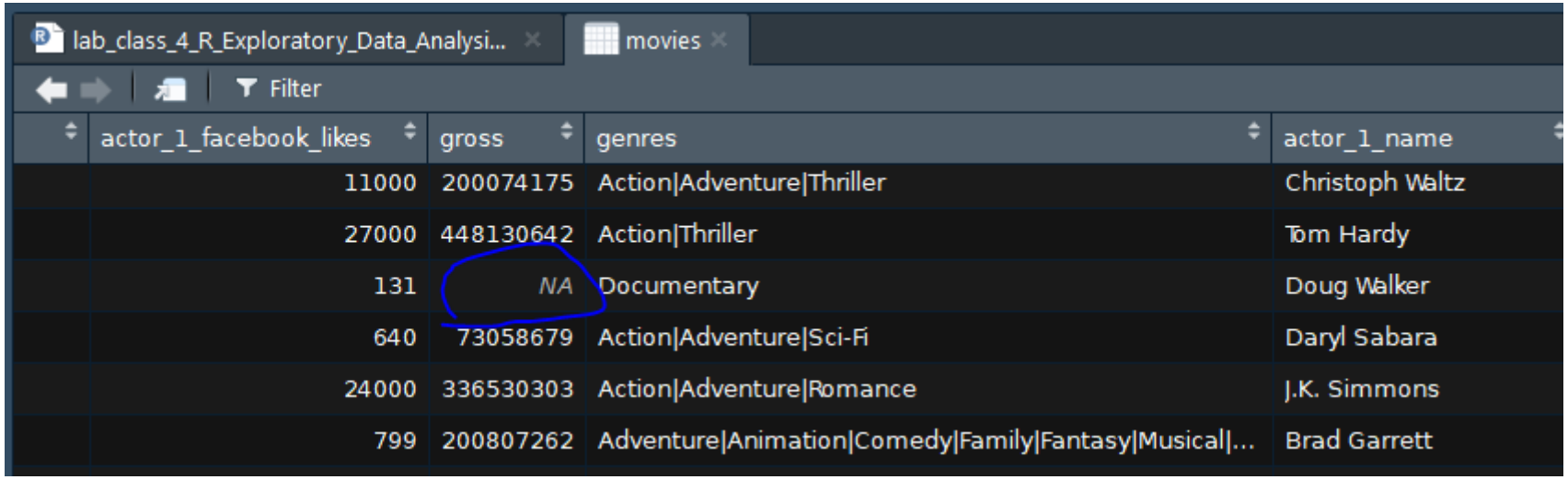


Class 3: Outline

1. Qs from last week?
2. **Basic Data Analysis**
   - Missing values
   - Loops
   - mutate to transform variables
   - Remove duplicates with distinct
   - Outputting “clean” data file”
3. **Data Analysis by Groups**
   - group_by() function
   - summarize() to create group variables
4. Data Analysis Lab Class 3
5. P-values
6. Introductory Machine Learning Concepts

Missing Values

```r
# ------------------------------------------------
# MISSING VALUES are values that are unknown in your dataset
# ------------------------------------------------
# R stores missing values as as NAs
is.na(NA)
1 > NA
1 + 1 == NA
NA == NA
y <- NA
y
x <- 1
y == x
```



actor_1_facebook_likes	gross	genres	actor_1_name
11000	200074175	Action|Adventure|Thriller	Christoph Waltz
27000	448130642	Action|Thriller	Tom Hardy
131	*NA*	Documentary	Doug Walker
640	73058679	Action|Adventure|Sci-Fi	Daryl Sabara
24000	336530303	Action|Adventure|Romance	J.K. Simmons
799	200807262	Adventure|Animation|Comedy|Family|Fantasy|Musical|...	Brad Garrett

Loops in R

```
# ---------------------------------------------
# LOOP through numbers using the FOR loop
# ---------------------------------------------

# for loops are created using the synthax
# for(i in start:end){
# do something with i
# }
```

```
> for(i in 1:10){
+   print(i)
+ }
[1] 1
[1] 2
[1] 3
[1] 4
[1] 5
[1] 6
[1] 7
[1] 8
[1] 9
[1] 10
```

LOOP through numbers using the FOR loop

```r
# how to see how many missings you have in each column?
# well, we want to sum through every column using a for loop
# then print the variable name using names(movies[i])
# then print the sum of is.na() for just that variable

# for each column in the movies
for(i in 1:ncol(movies)){

  # print the following
  print(

    # first print "Variable: "
    paste0("Variable: ",

           # then print the variable name, then "NAs: "
           names(movies)[i], " NAs: ",

           # then print the sum of the number of missing values
           # for that variable
           sum(is.na(movies %>% select(i)))
          )

        )
}
```

Functions in R

```r
# ------------------------------------------------
# Creating functions
# ------------------------------------------------
# we create a function in R by writing
# function_name <- function(input1, input2,...){
  # function arguments
# }

print_names <- function(data_frame){
  print(names(data_frame))
}
```

```
> print_names(movies)
 [1] "color"                 "director_name"              "num_critic_for_reviews"
 [4] "duration"              "director_facebook_likes"    "actor_3_facebook_likes"
 [7] "actor_2_name"          "actor_1_facebook_likes"     "gross"
[10] "genres"                "actor_1_name"               "movie_title"
[13] "num_voted_users"       "cast_total_facebook_likes"  "actor_3_name"
[16] "facenumber_in_poster"  "plot_keywords"              "movie_imdb_link"
[19] "num_user_for_reviews"  "language"                   "country"
[22] "content_rating"        "budget"                     "title_year"
[25] "actor_2_facebook_likes" "imdb_score"                "aspect_ratio"
[28] "movie_facebook_likes"
```

Build a function that prints number of missing values for each variable

```r
# Let's take the code we wrote above and translate
# it to a function called "num_missing".
# We can then call the function and pass our movies dataframe
# to it to export
num_missing <- function(data_frame){
  for(i in 1:ncol(movies)){
    print(
      paste0("Variable: ",
             names(movies)[i], " NAs: ",
             sum(is.na(movies %>% select(i)))
      )

    )
  }
}
```

```
> num_missing(movies)
[1] "Variable: color NAs: 0"
[1] "Variable: director_name NAs: 0"
[1] "Variable: num_critic_for_reviews NAs: 50"
[1] "Variable: duration NAs: 15"
[1] "Variable: director_facebook_likes NAs: 104"
[1] "Variable: actor_3_facebook_likes NAs: 23"
[1] "Variable: actor_2_name NAs: 0"
[1] "Variable: actor_1_facebook_likes NAs: 7"
[1] "Variable: gross NAs: 884"
[1] "Variable: genres NAs: 0"
[1] "Variable: actor_1_name NAs: 0"
[1] "Variable: movie_title NAs: 0"
[1] "Variable: num_voted_users NAs: 0"
[1] "Variable: cast_total_facebook_likes NAs: 0"
[1] "Variable: actor_3_name NAs: 0"
[1] "Variable: facenumber_in_poster NAs: 13"
[1] "Variable: plot_keywords NAs: 0"
[1] "Variable: movie_imdb_link NAs: 0"
[1] "Variable: num_user_for_reviews NAs: 21"
[1] "Variable: language NAs: 0"
[1] "Variable: country NAs: 0"
[1] "Variable: content_rating NAs: 0"
[1] "Variable: budget NAs: 492"
[1] "Variable: title_year NAs: 108"
[1] "Variable: actor_2_facebook_likes NAs: 13"
[1] "Variable: imdb_score NAs: 0"
[1] "Variable: aspect_ratio NAs: 329"
[1] "Variable: movie_facebook_likes NAs: 0"
```

MUTATE to Transform variables in your dataset

```
# ------------------------------------------------
# MUTATE to Transform variables in your dataset
# ------------------------------------------------

# adding new variables using mutate()
# note %<>% == DF <- DF %>%
# let's create new varibles budgetM and grossM that
# are budget and gross in units of millions
movies %<>% mutate(budgetM = budget/1000000,
                   grossM = gross/1000000,
                   profitM = grossM - budgetM)

movies %>% glimpse()

# so it looks like there's some outliers
# The most expensive movie ever made was Pirates of
# the Caribbean: On Stranger Tides
# which cost $387.8m. Any movies with a budget higher
# than this must be a data anomaly

# Let's use the filter command to remove these

movies_clean <- movies %>% filter(budgetM < 400)
```

Find Duplicate Rows with duplicated()

```
# ------------------------------------------------
# Find Duplicate Rows with duplicated()
# and find_duplicates() (must install hablar package)
# ------------------------------------------------
# number of duplicated rows
movies %>% duplicated() %>% sum()

# view duplicated rows
# install.packages(hablar)
movies %>% hablar::find_duplicates()
```

Output final clean version of dataset

```r
# ------------------------------------------------
# Output final clean version of dataset
# ------------------------------------------------
# remove duplicate rows, create new budget and gross variables,
# rename director and title
# remove budgets greater than 400M,
# order title, year, budget, director and gross first, then store in new file
movies_clean <-
  movies %>%
  distinct() %>%
  mutate(budgetM = budget/1000000,
         grossM = gross/1000000,
         profitM = grossM - budgetM) %>%
  rename(director = director_name,
         title = movie_title,
         year = title_year) %>%
  relocate(title, year, country, director, budgetM, grossM, imdb_score) %>%
  filter(budgetM < 400)

movies_clean %>% glimpse()
```

- Generally we do pre-processing on our dataset starting from a raw file.
- After these transformations we save a “clean” version of the dataset that is used for analysis

Create summary statistics by GROUP using group_by()

```r
# ---------------------------------------------
# Create summary statistics by GROUP using group_by()
# ---------------------------------------------
# group summaries using summarise and group_by
director_avg <-
  movies_clean %>%
    # group_by() is used to indicate the grouping variable
  group_by(director) %>%

    # summarize creates a new variable based on this group
    # here we create averages by director using the 'mean'
    # function
  summarize(gross_avg_director = mean(grossM, na.rm = TRUE))


# view results
director_avg %>% arrange(-gross_avg_director) %>% print()
```

Create averages, count and standard deviation by groups

```r
# ------------------------------------------------
# Create grouped variables using the Summarize function
# n() creates counts by
# sd() creates standard deviations
# ------------------------------------------------
# let's create budget by director, gross by director, profit by director,
# number films by director
director_df <-
  movies_clean %>%
  group_by(director) %>%
  summarize(

      # create average budget by director
      budget_avg_director = mean(budgetM, na.rm = TRUE),
      # create average gross by directory
      gross_avg_director = mean(grossM, na.rm = TRUE),
      # create average movie profit by director
      profit_avg_director = mean(profitM, na.rm = TRUE),
      # create variable that lists number of films
      # by director
      num_films = n(),
      # create a standard deviation of profit
      # by director
      profit_sd_director = sd(profitM, na.rm = TRUE)

      )
```

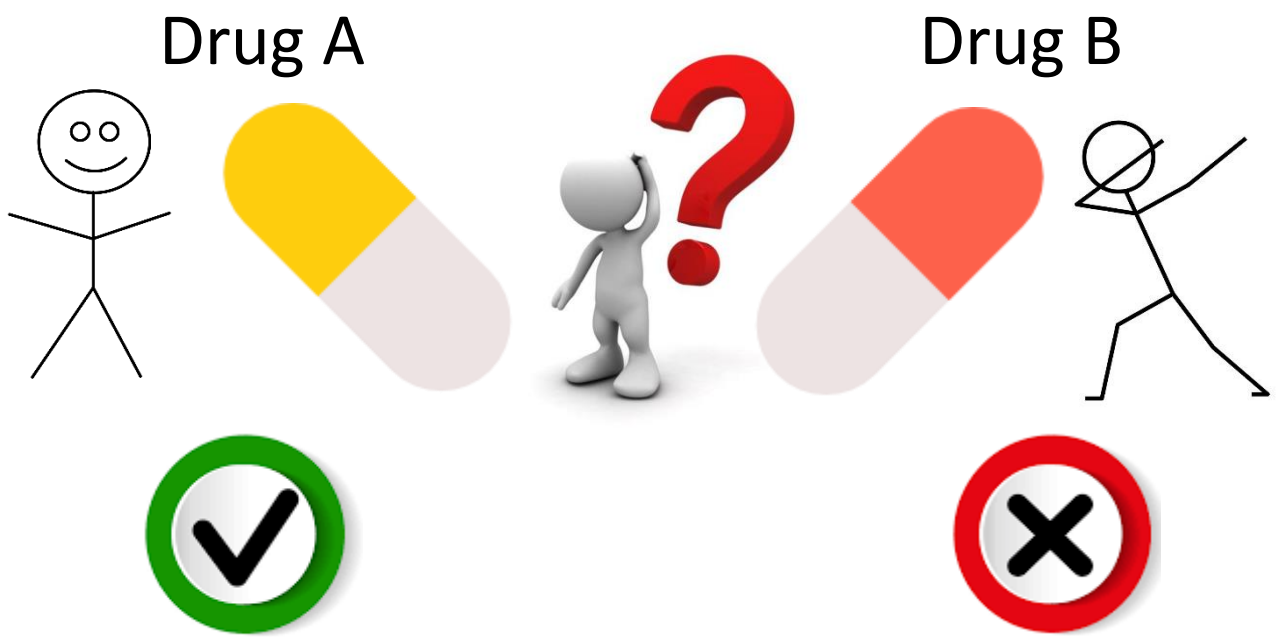
Lab Exercises

1. Print a dataframe with the film director name, and number of films for the 10 directors with the most films in the dataset
2. What movie genres have the highest average profit? (hint, must use a new group_by() command)
3. Which countries have the most films in the top 5000 IMDB database? (hint, must use a new group_by() command)
4. How many missing values are there for the profit_avg_director?
5. Why do some directors have “NA” for profit_avg_director?

Class 3: Outline

1. Qs from last week?
2. **Basic Data Analysis**
   - Missing values
   - Loops
   - mutate to transform variables
   - Remove duplicates with distinct
   - Outputting “clean” data file”
3. **Data Analysis by Groups**
   - group_by() function
   - summarize() to create group variables
4. Data Analysis Lab Class 3
5. P-values
6. Introductory Machine Learning Concepts

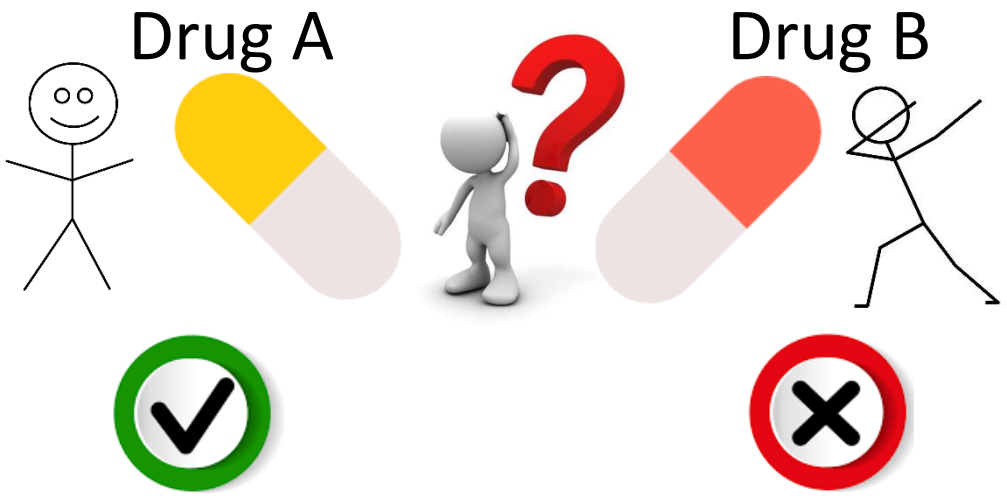
What Are P-Values?



- Suppose we want to know the effectiveness of Drug A vs Drug B
- We can give Drug A to 1 person, and give drug B to 1 person.
- Suppose person A gets better, and person B does not.
- Can we conclude drug A is better than drug B?

Adapted from the excellent StatQuest: https://youtu.be/vemZtEM63GY

What Are P-Values?



- **No! Perhaps…**
  - Person B didn't follow the instructions
  - Person B has pre-existing conditions
  - Person A is healthier
- Only by repeating the experiment many times can we learn whether Drug A > Drug B
- Ideally we express our confidence as a quantitative number, how likely it is that we find Drug A > Drug B only due to chance?

Repeating Experiment With More Observations

Drug A

Cured!	Not Cured!
73	125

37% Cured!

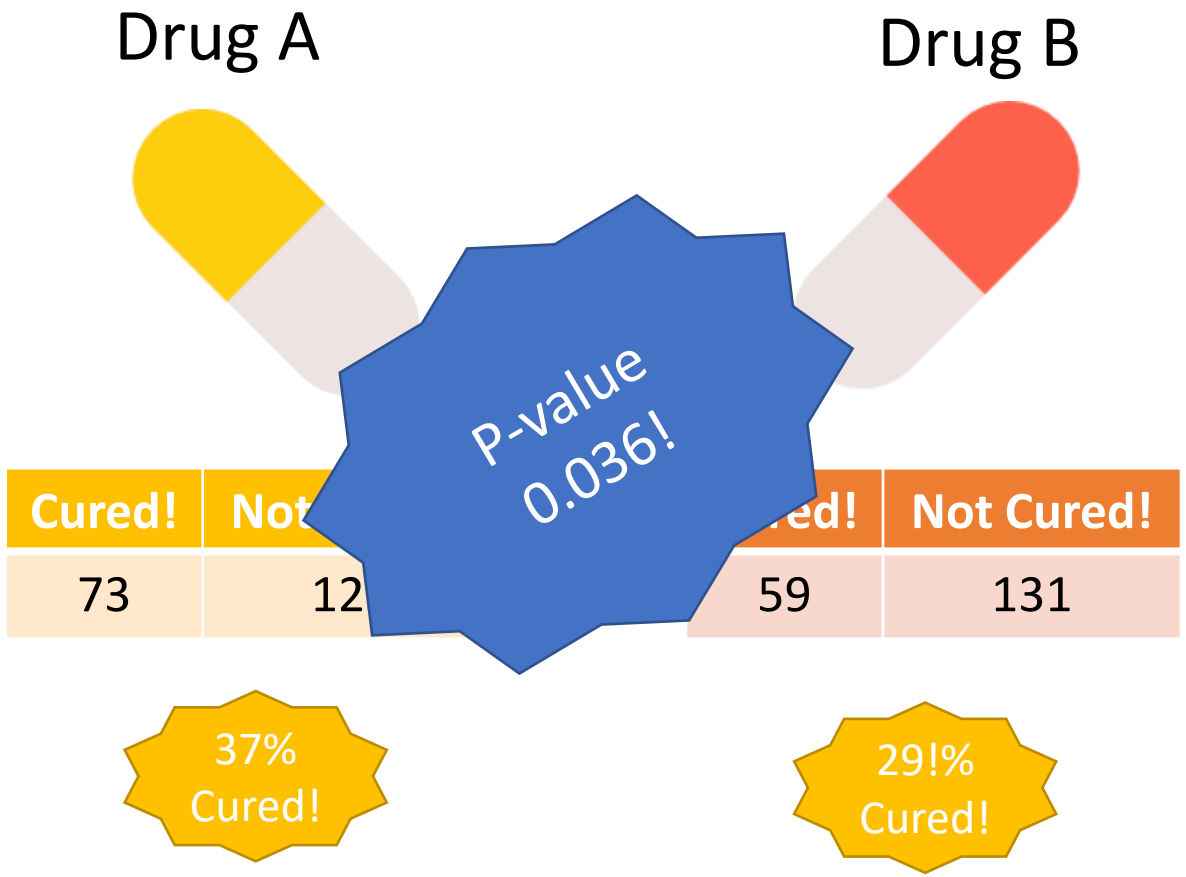
Drug B

Cured!	Not Cured!
59	131

29!% Cured!

- Suppose we repeat the experiment with ~400 patients
- We find that drug A has a cure rate of 37%, drug B a cure rate of 29%.
- But: No study is perfect. There are always random things that could influence drug A vs B
- **The p-value is a number between 0 and 1 that tells us how confident we should be between one hypothesis (H_0 or null) and another (H_1, alternative)**

Repeating Experiment With More Observations



- P value close to 0:
  - Result not likely due to chance
- P value close to 1:
  - More likely result is due to chance

- What is “enough evidence”?
  - Alpha = critical value
  - Commonly use ~ 0.05, 0.01, or 0.001
  - i.e. 5%, 1% or 0.1% chance difference due to randomness and there’s no difference

What do p-values measure?

P-value tells us the likelihood that, if the null hypothesis were true, we would receive a result as extreme as the one seen

Small p-values: unlikely that we would receive a result as extreme as the one seen if the null is true

P-value does not measure

- Size of effect
- Importance of a result
- Probability the alternative hypothesis is true

If You Find This Confusing You Are In Good Company

THE AMERICAN STATISTICIAN
2016, VOL. 70, NO. 2, 129–133
http://dx.doi.org/10.1080/00031305.2016.1154108

Taylor & Francis
Taylor & Francis Group

EDITORIAL

The ASA's Statement on *p*-Values: Context, Process, and Purpose

In February 2014, George Cobb, Professor Emeritus of Mathematics and Statistics at Mount Holyoke College, posed these questions to an ASA discussion forum:

> Q: Why do so many colleges and grad schools teach $p = 0.05$?
> A: Because that's still what the scientific community and journal editors use.
> Q: Why do so many people still use $p = 0.05$?
> A: Because that's what they were taught in college or grad school.

Cobb's concern was a long-worrisome circularity in the sociology of science based on the use of bright lines such as $p < 0.05$: "We teach it because it's what we do; we do it because it's what we teach." This concern was brought to the attention of the ASA Board.

The ASA Board was also stimulated by highly visible discussions over the last few years. For example, ScienceNews (Siegfried 2010) wrote: "It's science's dirtiest secret: The 'scientific method' of testing hypotheses by statistical analysis stands on a flimsy foundation." A November 2013, article in Phys.org Science News Wire (2013) cited "numerous deep flaws" in null hypothesis significance testing. A ScienceNews article (Siegfried 2014) on February 7, 2014, said "statistical techniques for testing hypotheses …have more flaws than Facebook's privacy policies." A week later, statistician and "Simply Statistics" blogger Jeff Leek responded. "The problem is not that people use P-values poorly," Leek wrote, "it is that the vast majority of data analysis is not performed by people properly trained to perform data analysis" (Leek 2014). That same week, statistician and science writer Regina Nuzzo published an article in *Nature* entitled "Scientific Method: Statistical Errors" (Nuzzo 2014). That article is now one of the most highly viewed *Nature* articles, as reported by altmetric.com (*http://www.altmetric.com/details/2115792#score*).

Of course, it was not simply a matter of responding to some 2014) and a statement on risk-limiting post-election audits (American Statistical Association 2010). However, these were truly policy-related statements. The VAM statement addressed a key educational policy issue, acknowledging the complexity of the issues involved, citing limitations of VAMs as effective performance models, and urging that they be developed and interpreted with the involvement of statisticians. The statement on election auditing was also in response to a major but specific policy issue (close elections in 2008), and said that statistically based election audits should become a routine part of election processes.

By contrast, the Board envisioned that the ASA statement on *p*-values and statistical significance would shed light on an aspect of our field that is too often misunderstood and misused in the broader research community, and, in the process, provides the community a service. The intended audience would be researchers, practitioners, and science writers who are not primarily statisticians. Thus, this statement would be quite different from anything previously attempted.

The Board tasked Wasserstein with assembling a group of experts representing a wide variety of points of view. On behalf of the Board, he reached out to more than two dozen such people, all of whom said they would be happy to be involved. Several expressed doubt about whether agreement could be reached, but those who did said, in effect, that if there was going to be a discussion, they wanted to be involved.

Over the course of many months, group members discussed what format the statement should take, tried to more concretely visualize the audience for the statement, and began to find points of agreement. That turned out to be relatively easy to do, but it was just as easy to find points of intense disagreement.

Q: Why do so many colleges and grad schools teach $p = 0.05$?
A: Because that's still what the scientific community and journal editors use.
Q: Why do so many people still use $p = 0.05$?
A: Because that's what they were taught in college or grad school.

Cobb's concern was a long-worrisome circularity in the sociology of science based on the use of bright lines such as $p < 0.05$: "We teach it because it's what we do; we do it because it's what we teach." This concern was brought to the attention of the ASA Board.

P-Values Commonly Misused or Misunderstood

Statistical tests, *P* values, confidence intervals, and power: a guide to misinterpretations

Sander Greenland[1] · Stephen J. Senn[2] · Kenneth J. Rothman[3] · John B. Carlin[4] · Charles Poole[5] · Steven N. Goodman[6] · Douglas G. Altman[7]

Received: 9 April 2016 / Accepted: 9 April 2016 / Published online: 21 May 2016

Abstract Misinterpretation and abuse of statistical tests, confidence intervals, and statistical power have been decried for decades, yet remain rampant. A key problem is that there are no interpretations of these concepts that are at once simple, intuitive, correct, and foolproof. Instead, correct use and interpretation of these statistics requires an attention to detail which seems to tax the patience of working scientists. This high cognitive demand has led to an epidemic of shortcut definitions and interpretations that are simply wrong, sometimes disastrously so—and yet these misinterpretations dominate much of the scientific literature. In light of this problem, we prov and a discussion of basic statistics that are and critical than typically found in tradition expositions. Our goal is to provide a resour tors, researchers, and consumers of sta knowledge of statistical theory and techr limited but who wish to avoid and spot misi We emphasize how violation of often uns protocols (such as selecting analyses for pres on the *P* values they produce) can lead to even if the declared test hypothesis is correc to large *P* values even if that hypothesis is then provide an explanatory list of 25 misint *P* values, confidence intervals, and power. with guidelines for improving statistical inte reporting.

Editor's note This article has been published online as supplementary material with an article of Wasserstein RL, Lazar NA. The ASA's statement on p-values: context, process and purpose. The American Statistician 2016.

Albert Hofman, Editor-in-Chief EJE.

Common misinterpretations of single *P* values

1. **The *P* value is the probability that the test hypothesis is true; for example, if a test of the null hypothesis gave *P* = 0.01, the null hypothesis has only a 1 % chance of being true; if instead it gave *P* = 0.40, the null hypothesis has a 40 % chance of being true**. No! The *P* value *assumes* the test hypothesis is true—it is *not* a hypothesis probability and may be far from any reasonable probability for the test hypothesis. The *P* value simply indicates the degree to which the data conform to the pattern predicted by the test hypothesis and all the other assumptions used in the test (the underlying statistical model). Thus $P = 0.01$ would indicate that the data are not very close to what the statistical model (including the test hypothesis) predicted they should be, while $P = 0.40$ would indicate that the data are much closer to the model prediction, allowing for chance variation.

Class 3: Outline

1. Qs from last week?
2. **Basic Data Analysis**
   - Missing values
   - Loops
   - mutate to transform variables
   - Remove duplicates with distinct
   - Outputting “clean” data file”
3. **Data Analysis by Groups**
   - group_by() function
   - summarize() to create group variables
4. Data Analysis Lab Class 3
5. P-values
6. Introductory Machine Learning Concepts

Supervised vs Unsupervised Learning

Supervised Learning:

- For every x_i we observe some y_i
- Ex: random forests to predict loan default (y_i) based on applicant characteristics (x_i)

Unsupervised Learning:

- We only observe x_i
- Ex: clustering loan applicants based on characteristics (x_i)

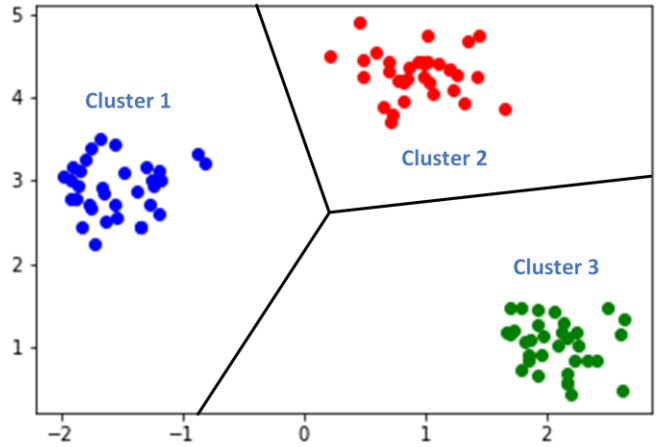
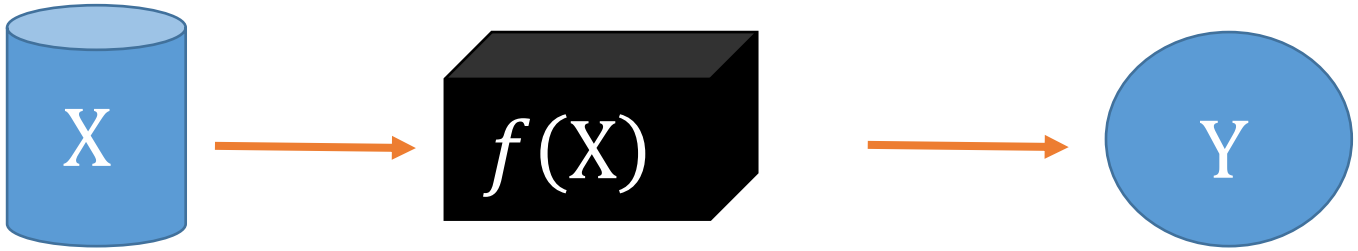


Fig.1. An Example Of Data Clustering

Supervised learning: learning $f(\mathrm{X})$ our predicted out given inputs

$$\mathrm{Y} = f(\mathrm{X}) + \epsilon$$



ϵ = “epsilon” (unexplained portion)

"Estimating" $\hat{f}(X)$

- $\mathrm{Y} = f(\mathrm{X}) + \epsilon$ is the true value
- We can only use data to "guess" at $f(\mathrm{X})$
- We call this guess $\hat{f}(X)$

How do we know when we've selected a "good" $\hat{f}(X)$?

- We reserve a portion of our data into a "test" set, estimate a model on the other part, and see how our model performs on this test set

Testing Training Data Subsets

Training set: (observation-wise) subset of data used to develop models

Testing/Training Split

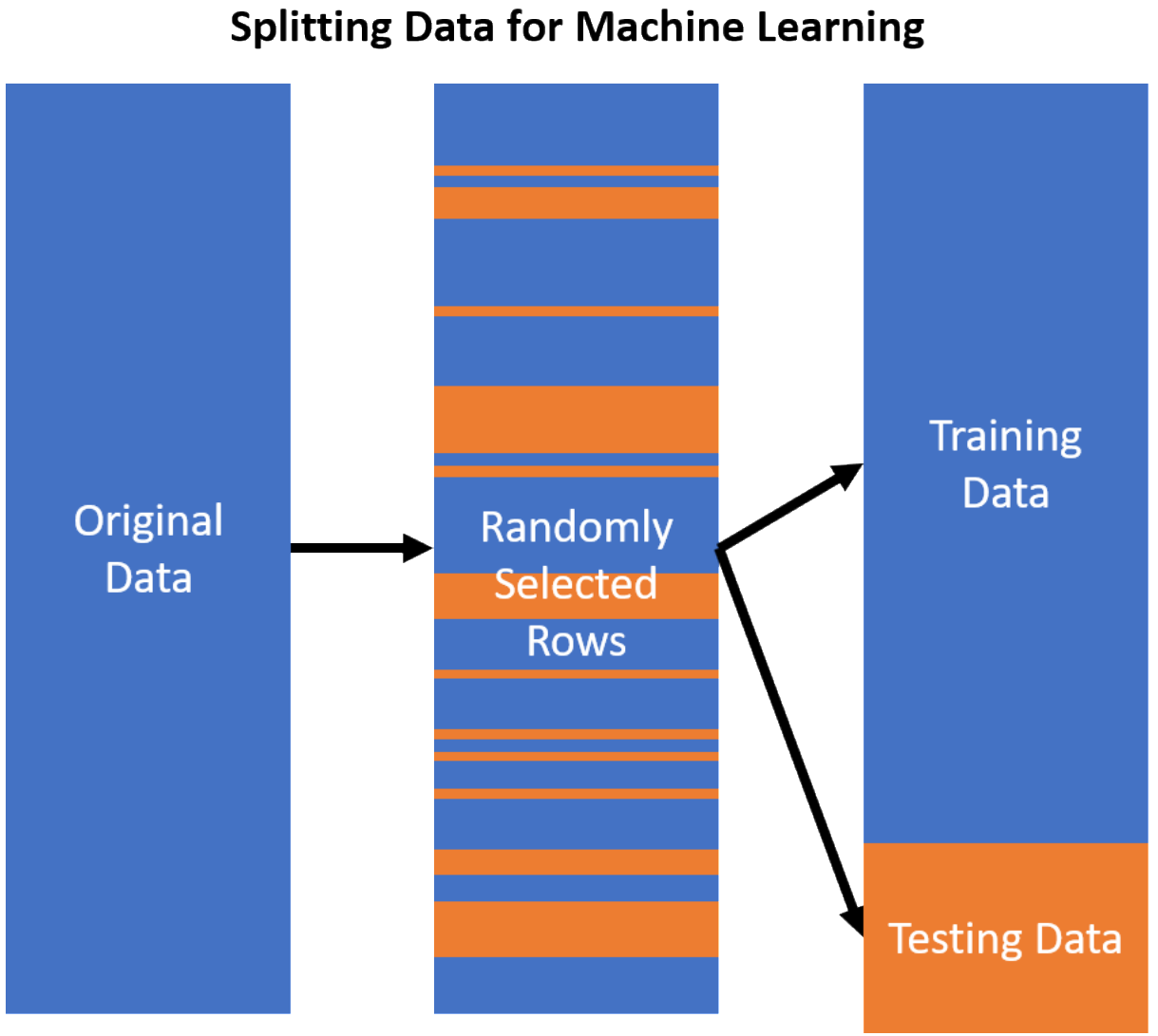
Training set: (observation-wise) subset of data used to develop models

Test set: subset of data used during intermediate stages to “tune” model parameters

Rule of thumb 75% training 25% test -ish

Randomly Selecting Rows for Test or Training Sets

- Observations are randomly selected into either testing or training splits of the data



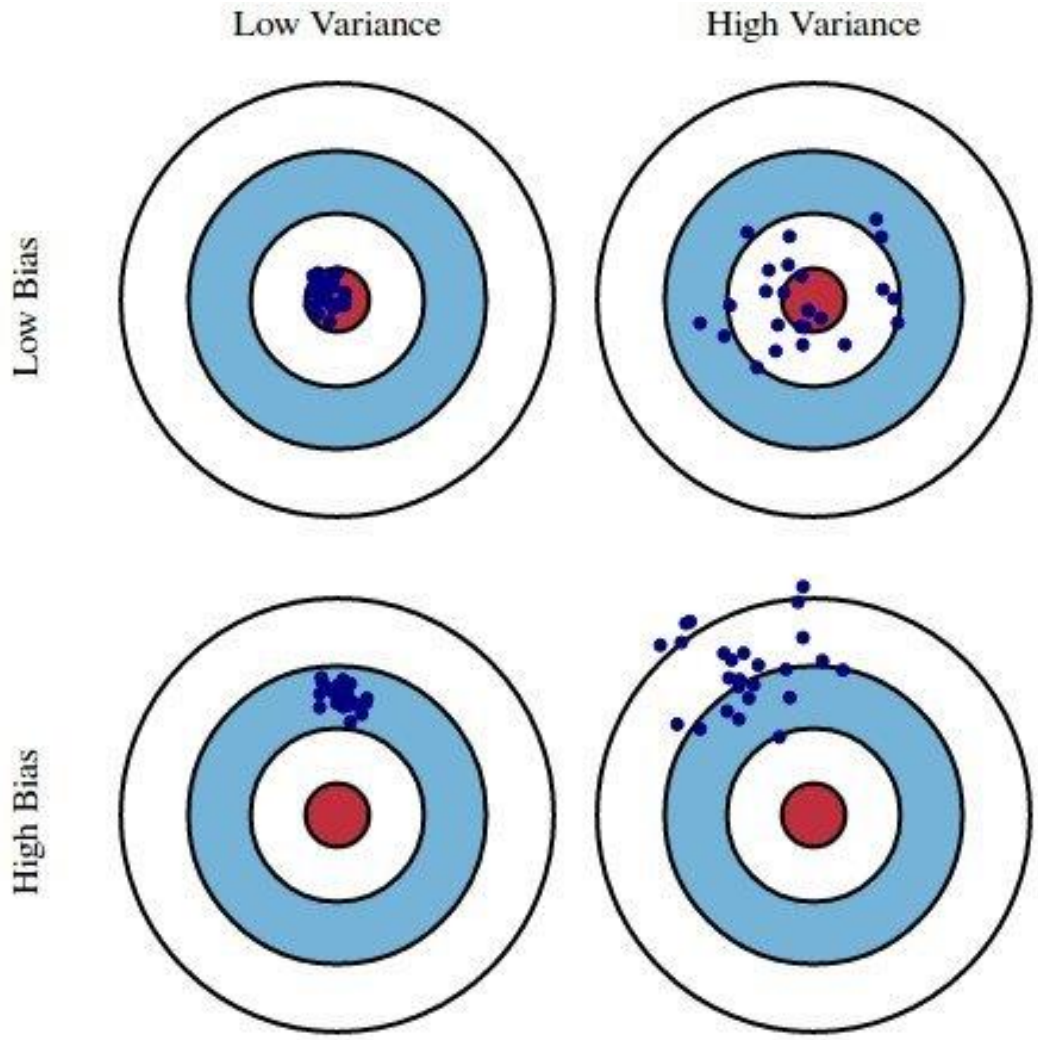
Bias and Variance

Bias: **Tendency of an in-sample statistic to over or under estimate the statistic in the *population***

Variance: **Tendency to noisily estimate a statistic**.

E.g., sensitivity to small fluctuations in the training dataset.

Bias-Variance Tradeoff



Class 3 Summary

- **Missing values** (NAs) indicate we don't know the value of a variable for that observation
  - Will need to make assumptions on how to treat these that can influence our results!
- **Functions** create "more readable" code.
- "**Clean**" version of datasets have been processed and are ready for analysis
- Use **group_by()** and **summarize()** to create statistics by groups (averages, standard deviations)
- P-values measure the discrepancy of the fit of a model or "null hypothesis" to data.
  - P-value tells us the likelihood that, if the null hypothesis were true, we would receive a result as extreme as the one seen
- **Supervised** models contain a y_i (target/outcome variable) for every x_i (descriptor variables)
- **Unsupervised** models contain only x_i
- **Training** data is the data we will use to estimate our model parameters
- **Testing** data is the data used to evaluate our model performance
- **Bias:** tendency of an in-sample statistic to over or underestimate the true value
- **Variance:** tendency to noisily estimate that statistic