Problem Set 2

Johnny Chapman Hersh MGSC 310

## Load Packages

library(readr)  
library('tidyverse')

## ── Attaching packages ─────────────────────────────────────────────── tidyverse 1.2.1 ──

## ✔ ggplot2 3.2.1 ✔ purrr 0.3.2  
## ✔ tibble 2.1.3 ✔ dplyr 0.8.3  
## ✔ tidyr 0.8.3 ✔ stringr 1.4.0  
## ✔ ggplot2 3.2.1 ✔ forcats 0.4.0

## ── Conflicts ────────────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

library('ggthemes')

### Question 1, ISLR Chapter 2, question 2:

1. Regression and interference because you are looking at what factors contribute to CEO Salary. n = 500 p = 4
2. Classification and prediction because we want to predict if the company is going to fail or be a success. This means we group our output into a class. n = 20 p = 13
3. Regression and prediction because we are trying to predict the % change in US/Euro Exchange rate based on weekly data from US, British, and German stock markets. n = 52 (% change weekly) p = 4

### Question 2, ISLR Chapter 2, question 4:

1. Classification
2. Email Spam. Whether an email should be considered spam and put in junk folder. Interference
3. Speech Recognition on the phone. Where your name is a class and is compared to the name on file. Interference
4. Credit Card Applications categorized into good, bad or okay credit.
5. Regression
6. Finding how much money you are going to need for gas on your upcoming road trip based on how many miles you went and how much it cost previous times. This is a prediction problem.
7. Forecasting Revenues and Expenses. This may be used to see the main drivers of the business. This can help in predicting future revenue
8. Predict housing prices based on location, number of bedrooms, number of bathrooms, and floors.
9. Cluster
10. Fantasy Football - Players on your team, points by player weekly, winning players. Little data at the start of season to predict which players are going to perform.
11. Identifying fake news based on content. It examines the words of the article and then clusters them. For this problem, certain words are more common in fake news articles. And when the percentage of those words is high, there is greater chance it is fake news.
12. Identifying criminal activity. This mean finding out what is criminal activity and what is not.

### Question A:

Download the Top 5000 movies on IMDB from the following link:

movies <- read.csv("Datasets/movie\_metadata.csv")

### Question B:

Run the code below to filter out films with unreasonably large budgets. Also use the code to create new variables (mutate) that are simplified versions of the genres and budget variables.

movies <- movies %>% filter(budget < 4e+08)  
movies <- movies %>% mutate(genre\_main = unlist(map(strsplit(as.character(movies$genres),  
 "\\|"), 1)), grossM = gross/1e+06, budgetM = budget/1e+06)  
movies <- movies %>% mutate(genre\_main = factor(genre\_main) %>%  
 fct\_drop())

### Question C:

Use the mutate function to generate profitM which is the difference between a movie’s gross and its budget, and the variable ROI which is the return on investment, specifically profit as a ratio of budget.

movies <- movies %>% mutate(profitM = grossM - budgetM, ROI = profitM / budgetM)

### Question D:

What is the average ROI for firms in the dataset? Use the function geom\_histogram() to create a histogram of ROI for movies in the database.

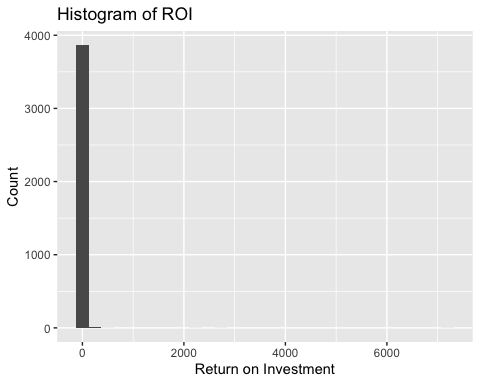
movies <- movies %>% drop\_na(ROI)  
mean(movies$ROI, na.rm = TRUE)

## [1] 5.273088

The average return on investments is $5.273 million.

ggplot(movies, aes(movies$ROI)) +  
 geom\_histogram() + labs(x = "Return on Investment", y = "Count", title = "Histogram of ROI")

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



### Question E:

From the plot above, it should be clear several outliers throw off the plot. Use the filter command to filter out films that have an ROI greater than 10. Just to be careful, count the number of films which match this criteria using the count() function.

movies <- movies %>% filter(ROI < 10)  
count(movies, vars = "ROI < 10")

## # A tibble: 1 x 2  
## vars n  
## <chr> <int>  
## 1 ROI < 10 3734

### Question F:

Use the group\_by() function to aggregate films by genre\_main and create mean ROI by genre using the summarize() command. Which film genres have the highest return on investment (ROI)? (Note we can also create standard errors by including in the summarize() command se\_ROI\_genre = sd(ROI, na.rm = TRUE)/sqrt(n()))

movies\_genre\_main <- movies %>% group\_by(genre\_main) %>%   
 summarize(avg\_ROI\_genre = mean(ROI, na.rm = TRUE),   
 se\_ROI\_genre = sd(ROI, na.rm = TRUE)/sqrt(n()), n = n())  
  
movies\_genre\_main

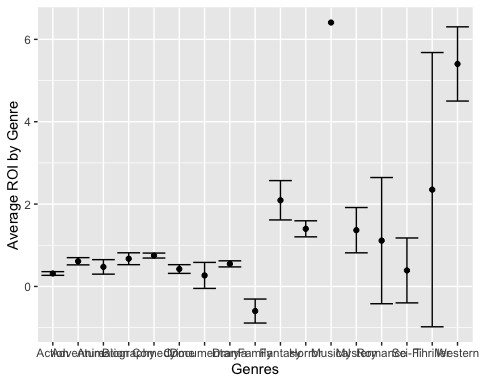
## # A tibble: 17 x 4  
## genre\_main avg\_ROI\_genre se\_ROI\_genre n  
## <fct> <dbl> <dbl> <int>  
## 1 Action 0.315 0.0447 951  
## 2 Adventure 0.612 0.0877 371  
## 3 Animation 0.475 0.176 43  
## 4 Biography 0.673 0.144 204  
## 5 Comedy 0.750 0.0603 992  
## 6 Crime 0.423 0.106 252  
## 7 Documentary 0.268 0.316 37  
## 8 Drama 0.548 0.0738 673  
## 9 Family -0.597 0.291 2  
## 10 Fantasy 2.09 0.478 31  
## 11 Horror 1.40 0.195 136  
## 12 Musical 6.41 NA 1  
## 13 Mystery 1.37 0.550 24  
## 14 Romance 1.11 1.53 3  
## 15 Sci-Fi 0.389 0.788 8  
## 16 Thriller 2.35 3.33 3  
## 17 Western 5.40 0.901 3

### Question G:

Use ggplot to create plots of the average ROI by genre using geom\_point(). Note you can also use geom\_errorbar() to create standard error bars using the code below. # geom\_errorbar(mapping = aes(ymin = avg\_ROI\_genre - # se\_ROI\_genre, ymax = avg\_ROI\_genre + se\_ROI\_genre),…

ggplot(movies\_genre\_main, aes(x = genre\_main, y = avg\_ROI\_genre)) +  
 geom\_point() + geom\_errorbar(mapping = aes(ymin = avg\_ROI\_genre -  
 se\_ROI\_genre, ymax = avg\_ROI\_genre + se\_ROI\_genre)) + labs(x = "Genres", y = "Average ROI by Genre")

## Warning: Removed 1 rows containing missing values (geom\_errorbar).



### Question H:

Use the summarize command to compute averages by actor\_1\_name for ROI and profit. Also within your summarize command calculate num\_films as the number of films by actor. Use the arrange() command to sort the dataframe in descending order by average actor ROI. Finally use slice() to print the first twenty rows. Which three actors have the highest ROIs?

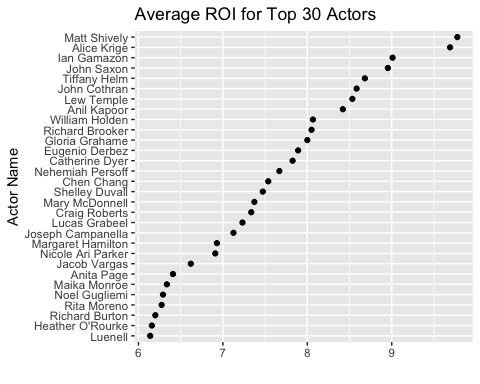
actor\_summary <- movies %>% group\_by(actor\_1\_name) %>%   
 summarize(avg\_ROI\_actor = mean(ROI, na.rm = TRUE),   
 avg\_profitM\_actor = mean(profitM, na.rm = TRUE),   
 num\_films = n()) %>%   
 arrange(desc(avg\_ROI\_actor))   
  
  
actor\_summary <- actor\_summary %>% drop\_na(avg\_profitM\_actor)  
actor\_summary

## # A tibble: 1,429 x 4  
## actor\_1\_name avg\_ROI\_actor avg\_profitM\_actor num\_films  
## <fct> <dbl> <dbl> <int>  
## 1 Matt Shively 9.78 48.9 1  
## 2 Alice Krige 9.69 53.3 1  
## 3 Ian Gamazon 9.01 0.0631 1  
## 4 John Saxon 8.95 40.3 1  
## 5 Tiffany Helm 8.68 19.1 1  
## 6 John Cothran 8.58 51.5 1  
## 7 Lew Temple 8.53 17.1 1  
## 8 Anil Kapoor 8.42 126. 1  
## 9 William Holden 8.07 24.2 1  
## 10 Richard Brooker 8.05 32.2 1  
## # … with 1,419 more rows

### Question I:

Use geom\_point() to plot the 30 actors with the highest ROI. Note we can use the top\_n() command to plot only the top 30 actors. We can also use fct\_reorder() to order the actors by highest ROI. library(forcats)

library(forcats)  
  
  
ggplot(data = actor\_summary %>% top\_n(30, wt = avg\_ROI\_actor),   
 mapping = aes(x = fct\_reorder(actor\_1\_name, avg\_ROI\_actor) %>%   
 fct\_drop(), y = avg\_ROI\_actor)) +   
 geom\_point() + coord\_flip() +   
 labs(x = "Actor Name",y = "", title = "Average ROI for Top 30 Actors")



### Question J:

Plot the 30 actors with the lowest return on investment

ggplot(data = actor\_summary %>% top\_n(-30, wt = avg\_ROI\_actor),   
 mapping = aes(x = fct\_reorder(actor\_1\_name, avg\_ROI\_actor) %>%   
 fct\_drop(), y = avg\_ROI\_actor)) +   
 geom\_point() + coord\_flip() +   
 labs(x = "Actor Name",y = "", title = "Average ROI for Bottom 30 Actors")

