QRM II Graded Assignment (2), Period 1 2025

Material by Sjoerd van Alten and Klervie Toczé

Group Number 63: YA.E. Gaziev (Aziz), J.C.L. Chen (Josie), S.E. Yakali (Selim), Y. Seo (Yonu)

09-10-2025

```
movies <- readr::read_tsv("Datasets/movies1.tsv")
```

```
## Rows: 505 Columns: 19
## -- Column specification --------------------------------------------------------
## Delimiter: "\t"
## chr   (8): keywords, original_language, title, genre, first_actor, first_act...
## dbl  (10): index, budget, popularity, revenue, runtime, vote_average, vote_c...
## date  (1): release_date
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

0 Introduction

This assignment is to be completed in groups of 3-4. Further, all students in your group need to be assigned to the same R tutorial group (Friday's tutorial). You can sign yourself up for a group on Canvas. Please do so **before the start of your first R tutorial on Friday September 5th.** You can use the Discussion Board in Canvas if you do not have a group yet or if your group is incomplete.

The assignment has 5 parts, and each part corresponds to the course material of that week (with the exclusion of week 6, for which there is no R programming material).

You are supposed to hand in these assignments on Canvas at the following dates:

- **Deadline 1** *Thursday September 25th, at 23:59pm*: you are supposed to hand in weeks 1, 2, and 3 of this assignment. This will determine 18% of your overall course grade
- **Deadline 2** *Thursday October 9th, at 23:59pm*: you are supposed to hand in weeks 4, and 5 of this assignment. This will determine 12% of your overall course grade

The R tutorials (each Friday) will consist of two halves. During the first half, you will discuss the tutorial exercises. These can be downloaded separately from Canvas. During the second half, you can work on this graded assignment within your own group. The purpose is that you find out how to work with R for doing statistical analyses by yourself. The tutorial exercises are meant to teach you basic commands to get you started, but to answer the problem sets in this assignment, you might need to research your own solutions, and use functions and commands not described in the tutorial exercises. Learning how to solve your own research problems is integral part of learning R. When you and your group get stuck on how to approach an exercise, the hierarchy in finding your way is as follows:

- use the concepts from the tutorial exercises;

- use the cheat sheets available on Canvas;
- use Google, YouTube, StackOverflow, or another website;
- ask the teacher.

The use of generative AI is **not** permitted and may result in a grade of 0. See the AI protocol in the course manual for details.

To answer the assignment, you can simply fill out this R markdown document. There are designated places which you can fill with R code. There are also designated spaces for you to answer each question. Often, the structure of an answer will be as follows. First, you type the R code in the designated box. This will show how you analyzed the data to get the answer to the question. Below the box for the R code, you will then summarize your answer to the question, i.e. what are the conclusions that you draw from the data analysis?

When handing in, you are supposed to submit this .Rmd file, and a knitted version of this document. You can knit this document to pdf, word, or html. Knitting to pdf requires you to have a .tex distribution installed on your computer. Knitting to Word requires you to have Word installed.

The exercises are designed such that you should be able to finish the majority of them during the tutorial each week. If you are not able to finish them fully during that time, you are expected to work on it in your own time using the computers on campus or your own device. It is best to meet as a group in-person when working together. If you want to work remotely, github is a good platform to guarantee smooth collaboration. Alternatively, you can email this .Rmd file back and forth to one another as a group, but this is not recommended as it is more cumbersome.

We encourage you to keep your code blocks, printing statements, and final answers, as short as possible. In any case, there is a page limit of 6 pages per week, which encompasses the total length of this document which consists of the questions, your coding lines, and your answers. When your answers to questions of the respective week exceed this page limit, they will not be graded, resulting in zero points.

Each week consists of 1, 2, or 3 subquestions. The total amount of points you can earn per week is 20 points.

1 Week 1

1. Find the dataset "movies1.tsv" on Canvas. Describe your data set: How many observations does it have. How many variables are there? How many subjects? What consists of a subject? [**4 points**]

```r
head(movies)
```

```
## # A tibble: 6 x 19
##   index  budget keywords original_language title popularity release_date revenue
##   <dbl>   <dbl> <chr>    <chr>             <chr>      <dbl> <date>         <dbl>
## 1  1773  2.70e7 fbi isl~ en                Mind~     17.2   2004-05-07    2.11e7
## 2  2540  1.5 e7 fire wi~ en                Kram~     31.6   2015-11-26    6.15e7
## 3  1174  4   e7 police ~ en                Ride~     25.1   2016-01-14    1.25e8
## 4  3262  0      masseus~ en                Enou~     15.0   2013-09-18    2.53e7
## 5  4324  0      christi~ en                Fait~      0.148 2006-10-27    0
## 6   214  1.20e8 u.s. ai~ en                The ~     25.8   2000-03-15    3.26e8
## # i 11 more variables: runtime <dbl>, vote_average <dbl>, vote_count <dbl>,
## #   genre <chr>, release_year <dbl>, release_month <dbl>, release_day <dbl>,
## #   first_actor <chr>, first_actor_gender <chr>, director_first_name <chr>,
## #   director_gender <chr>
```

```r
nrow(movies)
```

```
## [1] 505
```

```r
ncol(movies)
```

```
## [1] 19
```

Your Answer:

The Number of rows shows 505 observations The Number of columns shows 19 variables. The number of subjects in this dataset is equal to 505 same as observations. Since each subject is a movie in this dataset.

2. Which of the following types of variables are present in your data set? (i) nominal; (ii) ordinal; (iii); interval; (iv) ratio. If present, name one example of such a variable present in your data set. [**4 points**]

```r
nominal_var<- "title"
ordinal_var<- "popularity"
interval_var<-"release_date"
ratio_var<- "revenue"
```

Your Answer:

nominal example: “title” This dataset does not have an obvious ordinal example, but “popularity” can be considered as one. interval example:%22release_date” ratio example:%22revenue”

3. A movie studio wants to know which types of movies give maximal profit. Perform the following steps to provide the movie studio with an analysis which corresponds to their request:

a. Create the variable profits as the revenue of a movie minus its budget. Report its mean, median, maximum, and minimum. [**2 points**]
b. Which movie has the highest profits in your data set and how much are these profits. Which movie has the lowest and how much are its profits? If multiple movies have the exact same highest or lowest profits, give only one example. [**2 points**]
c. Create a boxplot of the variable profits. Make sure it has an appropriate title, and appropriate titles and labels for the x- and y-axis. Give Q1, Q2, Q3, and Q4. What does this tell you about the nature of making money in the movies industry? [**2 points**]
d. Add a new variable to your data set the log of profits. When creating this variable, what happens to movies for which profits is zero or negative? What then happens when you calculate the mean of log of profits? [**2 points**]
e. For movies that have a profit of zero or less, replace log of profits with “NA”. What is now the mean of log of profits? Create a boxplot for log of profits, again with an appropriate title, x- and y-axis labels. How does it compare to the boxplot you made under c.)? [**2 points**]
f. Create a scatterplot of with the runtime of movies on the x-axis and the average vote of movies on the y-axis. What do you conclude from the scatterplot? Are movies with a longer runtime considered worse or better by the audience, or does the audience not have a preference? Why do you think this is the case? [**2 points**]

For each step, you should provide first all the code you used to answer the question and then formulate an answer using full sentences.

Step a

```r
movies$profits<-movies$revenue-movies$budget
mean_profits<-mean(movies$profits)
median_profits<-median(movies$profits)
max_profit<-max(movies$profits)
min_profits<-min(movies$profits)
```

Your Answer:

Mean Profit = 63121475.22 Median Profit = 1900000 Max Profit = 2550965087 Min Profit = -90000000

Step b

```r
movies[which.max(movies$profits), "title"]
```

```
## # A tibble: 1 x 1
##   title
##   <chr>
## 1 Avatar
```

```r
movies[which.min(movies$profits), "title"]
```

```
## # A tibble: 1 x 1
##   title
##   <chr>
## 1 Mighty Joe Young
```

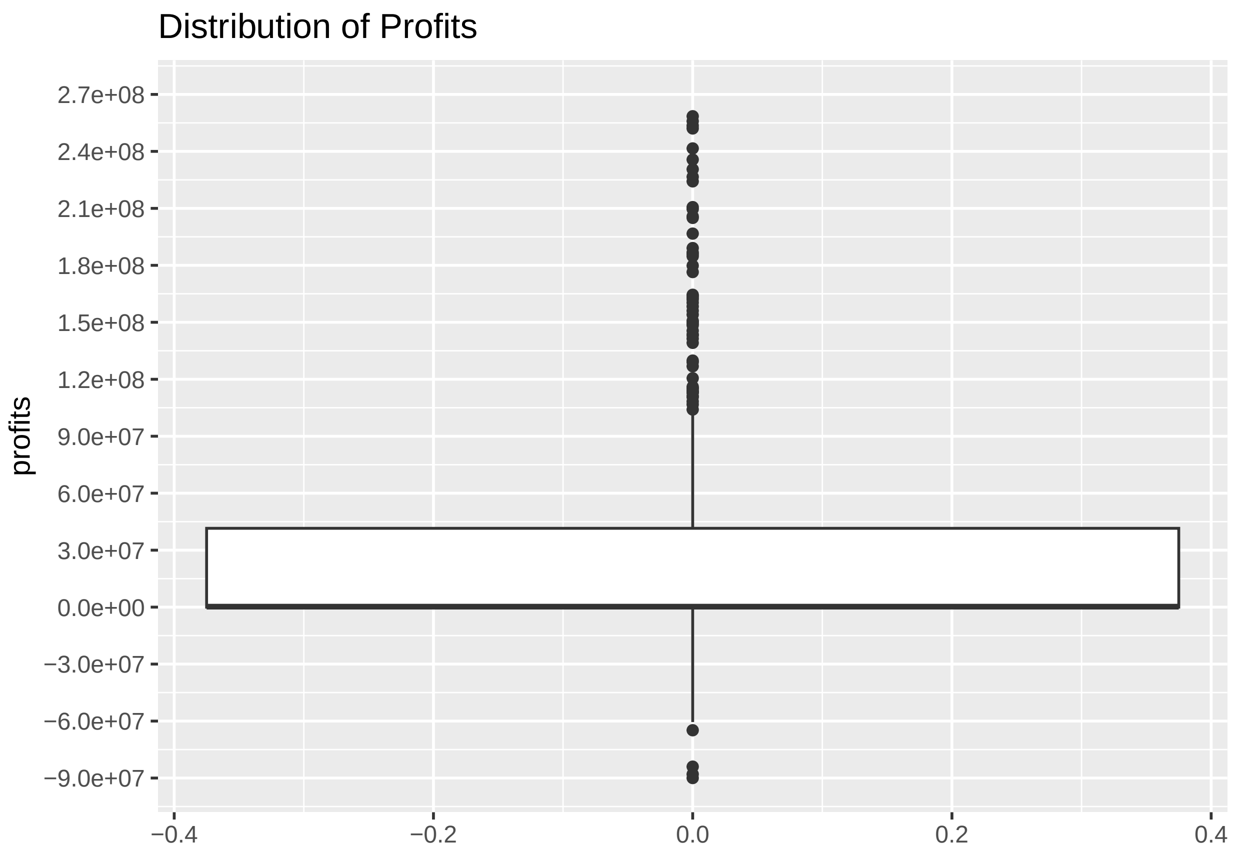
Your Answer:

The movie with the maximum profit is “Avatar” and the movie with the minimum profit is “Mighty Joe Young”.

Step c

```r
ggplot(movies, aes(y = profits))+
        geom_boxplot()+
        labs(title = "Distribution of Profits",
        ylab = "profits")+
        scale_y_continuous(limits = c(-90000000,270000000), breaks =
         ↪  seq(-90000000,270000000, by = 30000000))
```

```
## Warning: Removed 31 rows containing non-finite outside the scale range
## (`stat_boxplot()`).
```



```
quantile(movies$profits, probs = c(0.25, 0.5, 0.75, 1))
```

```
##        25%        50%        75%       100%
##          0    1900000   60514050 2550965087
```

Your Answer:

It shows that it is highly competitive industry where very few movies become profitable. It may be concluded that it is high risk industry but with high rewards.

Step d

```
movies$log_profits<-log(movies$profits)
```

```
## Warning in log(movies$profits): NaNs produced
```

```
mean(movies$log_profits, na.rm = TRUE)
```

```
## [1] -Inf
```

Your Answer:

When profits are equal to 0, the $log\_profits$ of the observation is equal to $-\infty$, and the log_profit of the observation with profits below 0 produced NaN, since the log of a negative number is undefined. When calculating the mean of log profits, those infinite and NaN values will be excluded if using `na.rm = TRUE`.

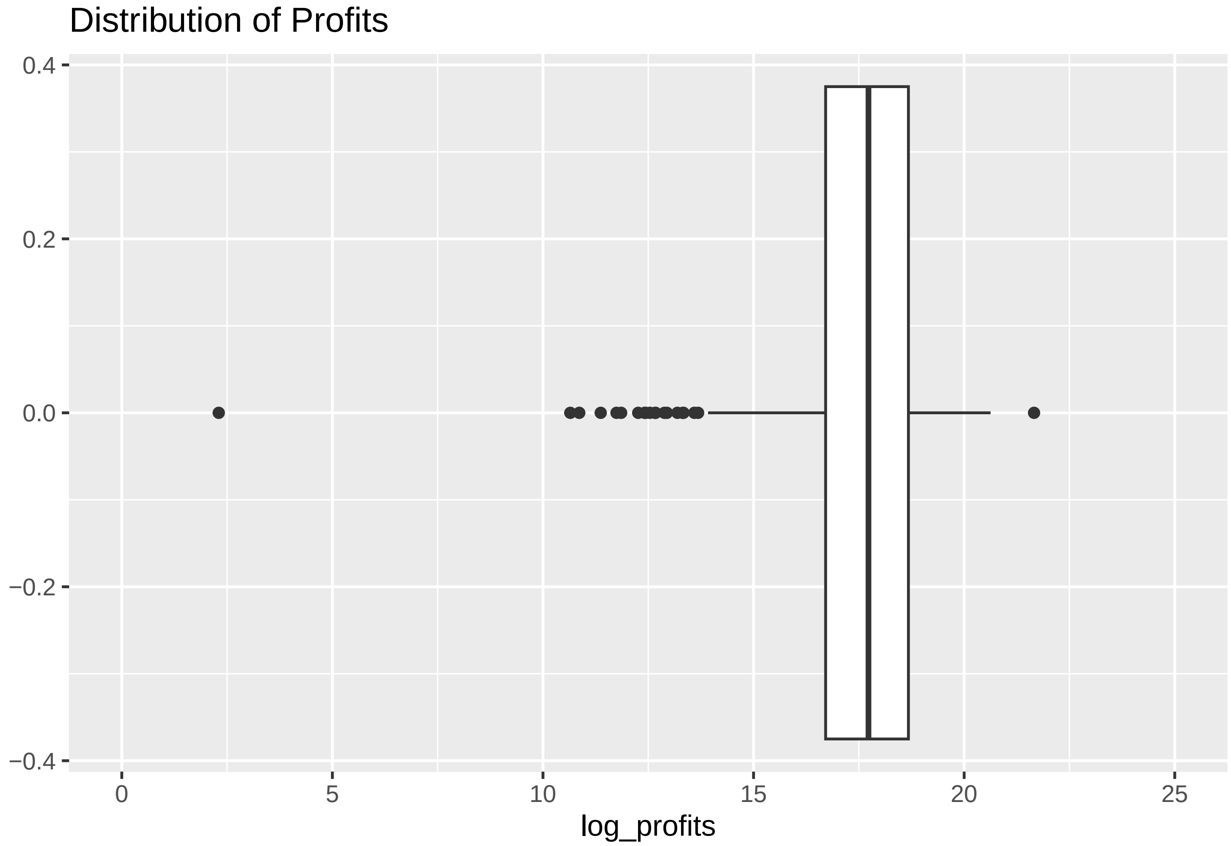
Step e

```
mean(movies$log_profits, na.rm = FALSE)
```

```
## [1] NaN
```

```
ggplot(movies, aes(x=log_profits))+
  geom_boxplot()+
  labs(title = "Distribution of Profits",
        xlab = "movies",
        ylab = "log_ptrofits")+
  scale_x_continuous(breaks = seq(0, 25, by = 5), limits = c(0,25))
```

```
## Warning: Removed 229 rows containing non-finite outside the scale range
## (`stat_boxplot()`).
```



Your Answer:

With values of profits equal or less than 0 being replaced with "NA", the mean of log_profit is 17.38. As for the differences with the boxplot under subquestion c. The profits boxplot included all values including negative, while log_profits boxplot only included positive values. The log_profits boxplot has less outliers and values are more symmetrically distributed strongly suggesting log_profits to have normal distribution. On the other hand, it ignores the observations that include unsuccessful movies with negative or small profitability, ignoring the competitive nature of the movie industry.

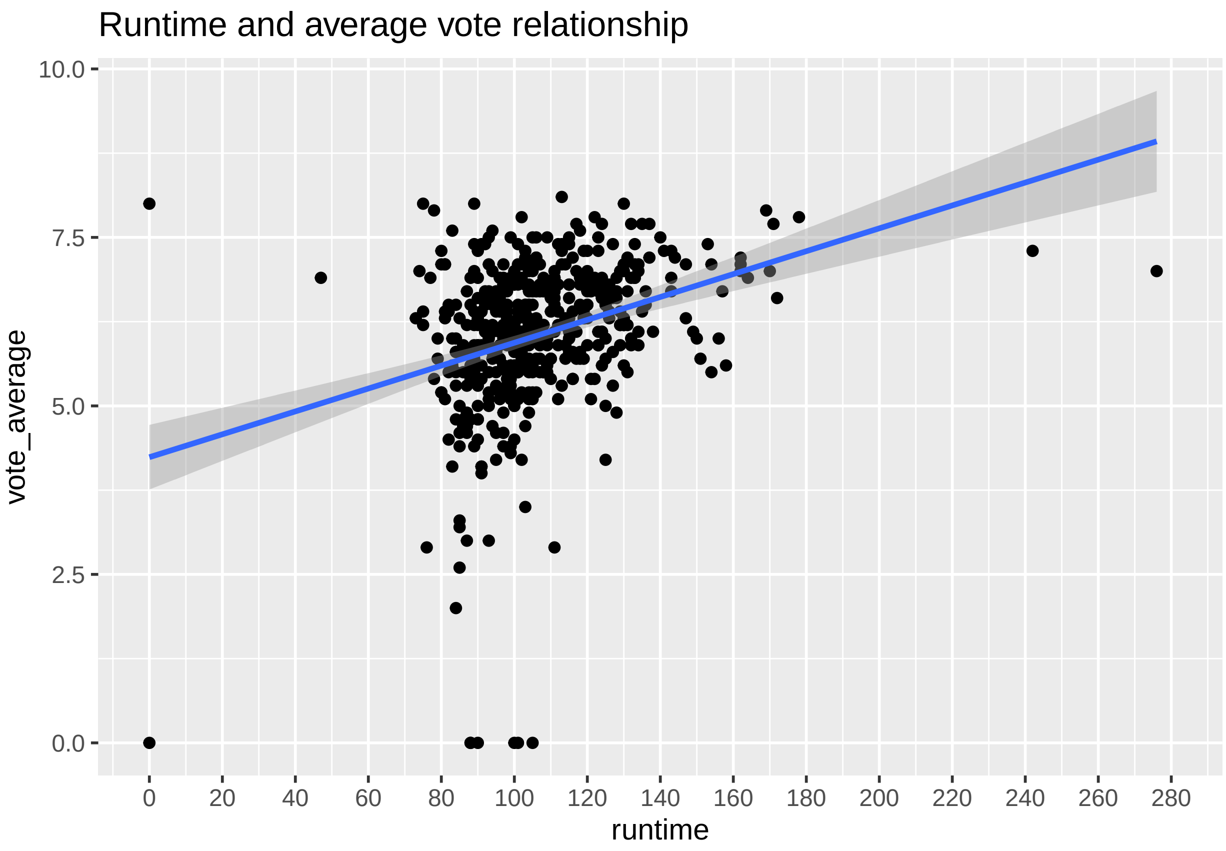
Step f

```r
ggplot(movies, aes(x = runtime, y = vote_average))+
  geom_point()+
  geom_smooth(method = lm, se = TRUE)+
  scale_x_continuous(limits = c(0,280), breaks = seq(0,280, by = 20))+
  labs(title = "Runtime and average vote relationship",
       x = "runtime",
       y = "vote_average")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 1 row containing non-finite outside the scale range
## (`stat_smooth()`).
```

```
## Warning: Removed 1 row containing missing values or values outside the scale range
## (`geom_point()`).
```



Your Answer:

The slope of the regression line is relatively flat with an upward position. This means that runtime has a relatively weak positive relationship with *vote_average*. It is evident that most of the observations are clustered around the middle, so there is no strong preference for longer or shorter movies. The majority of movies that fall in this runtime range are with vote_average of 5 and above, while very few movies with runtime of 80 minutes have vote_average lower than 5. Overall, it may be concluded that runtime and vote_average has weak relationships and viewers do not have a strong preference.

2 Week 2

1 Is your dataset movies1.tsv the full population, or is it a sample of a larger population? If the latter, how would you describe the full population? [**4 points**]

Your Answer:

The dataset movies1.tsv is a sample of a larger population. The full population would be all movies that have ever been released worldwide.

2

a. For which actor in your data set do you observe the most movies? [**2 points**]
b. What is the average revenue of the movie in which this actor plays and does the revenue lie above or below the revenue of an average movie according to your data set? [**2 points**]

c. How trustworthy do you consider your conclusion to answer 2b? Use the term "law of large numbers" in your explanation. [**2 points**]

step a

```
actor_counts <- table(movies$first_actor)
sorted_actors <- sort(actor_counts, decreasing = TRUE)
top_actor <- names(sort(table(movies$first_actor), decreasing = TRUE))[1]
cat(top_actor)
```

```
## Bruce Willis
```

Your Answer:

Bruce Willis

step b

```
top_actor_movies <- movies[movies$first_actor == top_actor, ]
actor_avg_revenue <- mean(top_actor_movies$revenue, na.rm = TRUE)
overall_avg_revenue <- mean(movies$revenue, na.rm = TRUE)

cat("The average revenue of the movie in which Bruce Willis plays:", actor_avg_revenue,
↪  "\n")
```

```
## The average revenue of the movie in which Bruce Willis plays: 116280090
```

```
cat("The average revenue of the movies in the dataset:", overall_avg_revenue, "\n")
```

```
## The average revenue of the movies in the dataset: 94815482
```

```
if(actor_avg_revenue > overall_avg_revenue) {
  cat("The revenue of his movies lies above the revenue of an average movie.\n")
} else {
  cat("The revenue of his movies is below the revenue of an average movie.\n")
}
```

```
## The revenue of his movies lies above the revenue of an average movie.
```

Your Answer:

The average revenue of the movie in which Bruce Willis plays: 116280090 The average revenue of the movies in the dataset: 94815482 The revenue of his movies lies above the revenue of an average movie.

step c

```r
sum(movies$first_actor == "Bruce Willis", na.rm = TRUE)
```

```
## [1] 7
```

Your Answer:

Bruce Willis appears in only 7 movies in the dataset, the sample size is relatively small. According to the law of large numbers, as the number of observations increases, the sample mean will converge to the true population mean. This is because the number of movies for him is limited, the observed average revenue may not be a reliable estimate of his true average movie revenue.

3 For this question, you will assume that your data set is the full population.

a. Recode profits such that it is expressed in millions. What is the variance of the variable profits (in millions) in your data set? [**2 points**]
b. Create a new data set, called movies_sample. Make sure that it is a random sample of your data set of 25 movies. What is the variance of profits in this random sample? How does it compare to the variance of profits in 2a? [**2 points**]
c. In a for loop, create 100 different samples of 25 movies, as in b, and estimate the variance within each sample. Save the variance of each sample in a vector called sample_vars. So the first position of the vector would have the variance of the first sample, the second position the variance of the second sample, etc. Print the start of this vector. [**2 points**]
d. Summarize and make a histogram of sample_vars. What is the mean, standard deviation and shape of its distribution? [**2 points**]
e. In your opinion, is a sample of 25 movies sufficient to get a reliable estimate of the population variance of profits, using the sample variance? Explain? [**2 points**]

step a

```r
movies$profits <- movies$revenue - movies$budget
movies$profits_millions <- movies$profits/1000000
sample_variance <- var(movies$profits_millions, na.rm = TRUE)
n <- sum(!is.na(movies$profits_millions))
population_variance <- sample_variance * (n-1)/n

cat("The variance of the variable profits:", population_variance, "\n")
```

```
## The variance of the variable profits: 30402.8
```

Your Answer:

The variance of the variable profits: 30402.8

step b

```r
set.seed(321)
movies_sample <- movies %>% sample_n(25)
variance_sample <- var(movies_sample$profits_millions, na.rm = TRUE)

cat("Sample profits variance:", variance_sample, "\n")
```

```
## Sample profits variance: 31872.76
```

```r
cat("The population profits variance:", population_variance, "\n")
```

```
## The population profits variance: 30402.8
```

```r
if(variance_sample > population_variance) {
  cat("The sample profits variance is larger than the population profits variance.\n")
} else {
  cat("The sample profits variance is smaller than the population profits variance.\n")
}
```

```
## The sample profits variance is larger than the population profits variance.
```

Your Answer:

Sample profits variance: 31872.76 The population profits variance: 30402.8 The sample profits variance is larger than the population profits variance.

step c

```r
sample_vars <- numeric(100)
set.seed(321)

for(i in 1:100) {
  sample_i <- movies %>% sample_n(25)
  sample_vars[i] <- var(sample_i$profits_millions, na.rm = TRUE)
}
cat(head(sample_vars))
```

```
## 31872.76 8966.419 6408.811 16055.5 15201.83 48963.43
```

Your Answer:

31872.76 8966.419 6408.811 16055.5 15201.83 48963.43

step d

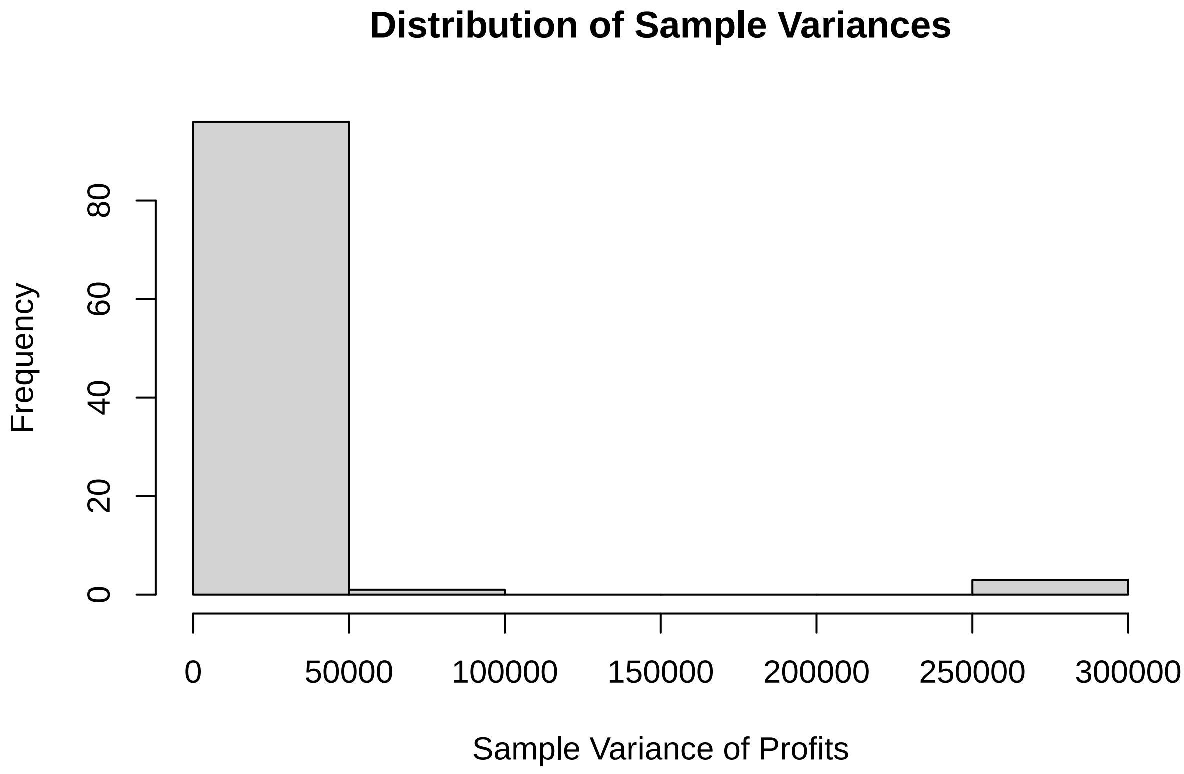
```r
mean_sample_vars <- mean(sample_vars, na.rm = TRUE)
sd_sample_vars <- sd(sample_vars, na.rm = TRUE)

cat("Mean of sample variances:", mean_sample_vars, "\n")
```

```
## Mean of sample variances: 24227.96
```

```r
cat("Standard deviation of sample variances:", sd_sample_vars, "\n")
```

```
## Standard deviation of sample variances: 45274.43
```

```r
hist(sample_vars,
     main = "Distribution of Sample Variances",
     xlab = "Sample Variance of Profits")
```



Your Answer:

Mean of sample variances: 24227.96 Standard deviation of sample variances: 45274.43

step e

Your Answer:

In our opinion, the sample of 25 movies is not sufficient to get a reliable estimate of the population variance of profits. As we saw, the sample variances vary widely depending on which movies are included, and the distribution of sample variances is highly skewed. This means that with such a small sample, the estimate is very unstable. A larger sample size would provide a more reliable and consistent estimate of the population variance.

3 Week 3

For the next part of the assignment, assume that the movies in your data frame are a random sample of a larger population of movies.

1

a. Create a new data set that only includes movies that are of the genre “Thriller”. For these thriller movies, give a 99 percent confidence interval for the variable *runtime*. Interpret the result. [**2 points**]
b. Now, assume that the variance of *runtime* amongst thriller movies in your data is exactly the same as the variance of *runtime* in the population. Under this assumption, give a 99 percent confidence interval for the variable *runtime* among thriller movies. Interpret the result. Is you confidence interval wider or less wide than the one you found under question 1a? Why is that the case? [**2 points**]

step a

```r
thriller <- movies %>%
  filter(genre == "Thriller", !is.na(runtime))
n <- nrow(thriller)
variable <- var(thriller$runtime)

alpha <- 0.01
CI <- (n -1)*variable / qchisq(c(0.995, 0.005), df = n -1)
CI
```

```
## [1] 186.5459 468.0735
```

Your Answer:

186.5459 and 468.0735

step b

```r
mean_thriller_runtime <- mean(thriller$runtime)
error <- sqrt(variable / n)
z <- qnorm(0.995)

CI_mean <- mean_thriller_runtime + c(-1,1)*z*error
CI_mean
```

```
## [1] 100.1081 110.8457
```

Your Answer:

100.1081 and 110.8457

2

a. Using an appropriate five-step procedure, set up a test for the null hypothesis that the variance of runtime equals 500. Clearly state your null hypothesis, alternative hypothesis your test statistic, your critical value, and your conclusion. [**2 points**]
b. For the validity of your test in 2a, what assumption about the distribution of revenue needs to hold? Make an appropriate plot to test this assumption. What do you conclude? [**2 points**]

step a

```
sigma_hypo <- 500
test_hypo <- (n - 1)*variable / sigma_hypo

alpha <- 0.05
critical_val <- qchisq(c(alpha/2 , 1 - alpha/2), df = n-1)
test_hypo
```

```
## [1] 36.14443
```

```
critical_val
```

```
## [1] 43.77595 88.00405
```

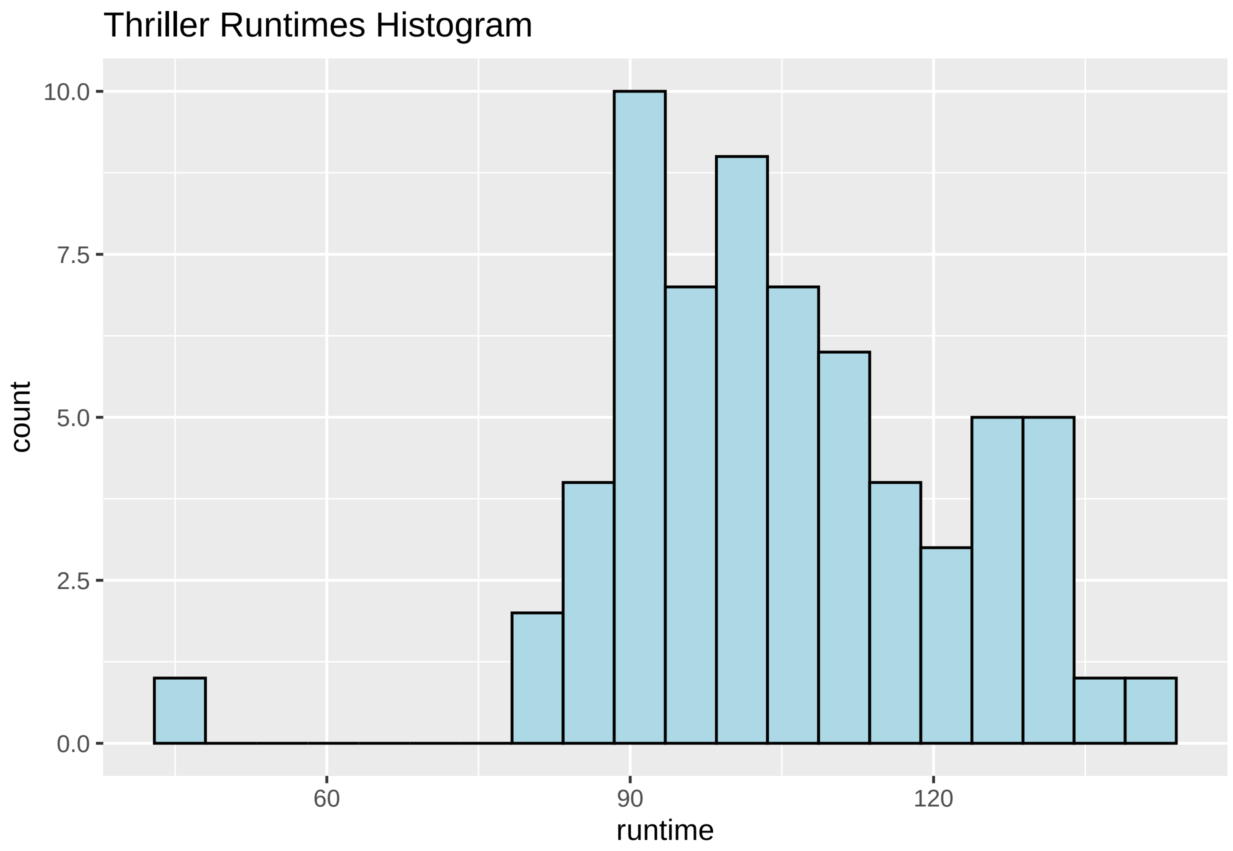
Your Answer:

Test statistic: 36.14
Critical values: 43.78 and 88.00

$H_0 : \sigma^2 = 500$
$H_1 : \sigma^2 \neq 500$
Since 36.14 is less than the lower critical value, we reject the null hypothesis at $\alpha = 0.05$.

step b

```
ggplot(thriller, aes(x = runtime)) +
  geom_histogram(bins = 20, fill = "lightblue", color = "black") +
  labs(title = "Thriller Runtimes Histogram")
```



Your Answer:

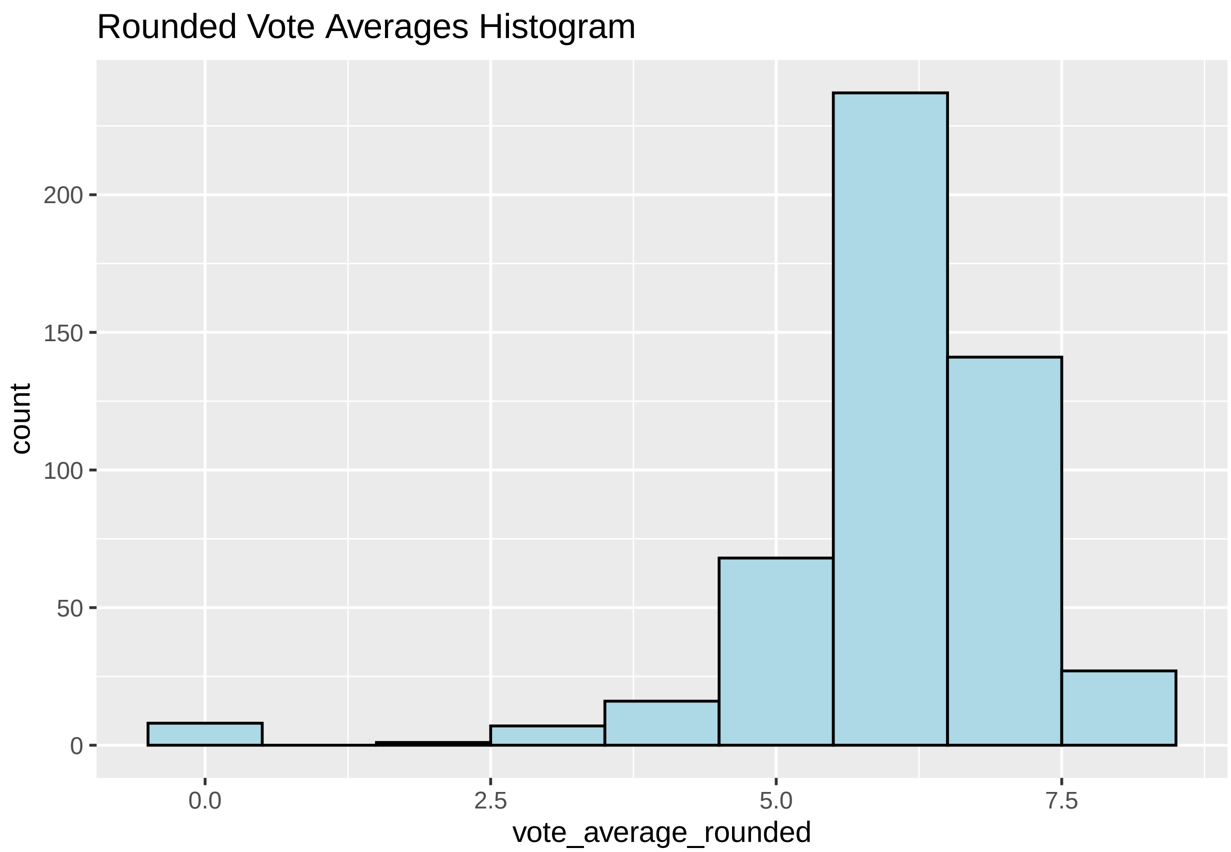
To validate the test for the variance of runtime, we assume the the runtime variable is normally distributed among thriller movies. To check this assumption, we plotted a histogram to see the runtime values of thriller movies. If the histogram appears to be bell-shaped and symmetric (which it does, with some minor peaks on the edges), it would suggest the normality assumption holds. If the histogram is skewed in any direction or shows large peaks, the normality assumption may not hold, which could affect the validity of the test.

3. There is an argument going on in the movie studio. *Bob* claims that they should make higher-quality movies, as this will bring in more profits. *Chantal* disagrees. She tells Bob that mediocre movies bring in the most profits. You are asked to advise on who is right.

a. Create a new variable called vote_average_rounded. Make sure this variable is the same as vote_average, but without any decimals (i.e., a 6.3 becomes a 6, a 8.7 an 8, etc.). Display a histogram of vote_average_rounded. [**2 points**]
b. Create a scatter plot with vote_average_rounded on the x axis and the mean of profits within each category of vote_average_rounded on the y-axis. Make sure it has an appropriate title, and appropriate titles and labels for the x- and y-axis. At which rating of movies are profits the highest? [**3 points**]
c. Recreate the scatter plot with year on the x axis and mean_profits on the y-axis, but now add bars around each point, indicating the 95% confidence interval. [**3 points**]
d. Write an advice to settle the argument between Bob and Chantal. [**4 points**]

step a

```
movies <- movies %>%
  mutate(vote_average_rounded = round(vote_average))

ggplot(movies, aes(x = vote_average_rounded)) +
  geom_histogram(binwidth = 1, fill = "lightblue", color = "black") +
  labs(title = "Rounded Vote Averages Histogram")
```



Your Answer:

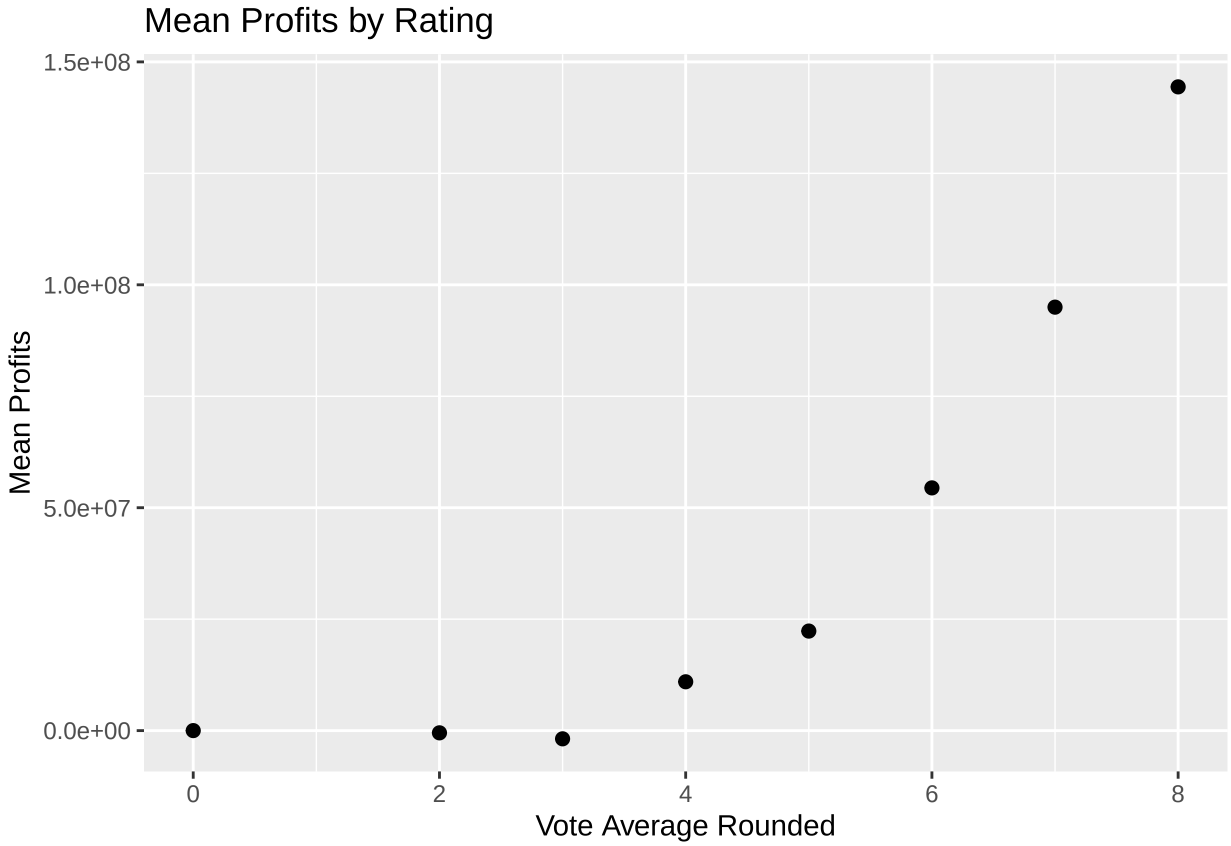
We observe here that a majority of the votes fall between values 5-7.

step b

```
rating_profits <- movies %>%
  group_by(vote_average_rounded) %>%
  summarize(mean_profits = mean(profits, na.rm = TRUE))
rating_profits
```

```
## # A tibble: 8 x 2
##   vote_average_rounded mean_profits
##                  <dbl>        <dbl>
## 1                    0        -62.8
## 2                    2    -500000
## 3                    3   -1841531.
## 4                    4   10952534.
## 5                    5   22329019.
## 6                    6   54446352.
## 7                    7   94988853.
## 8                    8  144404032.
```

```
ggplot(rating_profits, aes(x = vote_average_rounded, y = mean_profits)) +
  geom_point(size = 2, color = "black") +
  labs(title = "Mean Profits by Rating",
       x = "Vote Average Rounded",
       y = "Mean Profits")
```

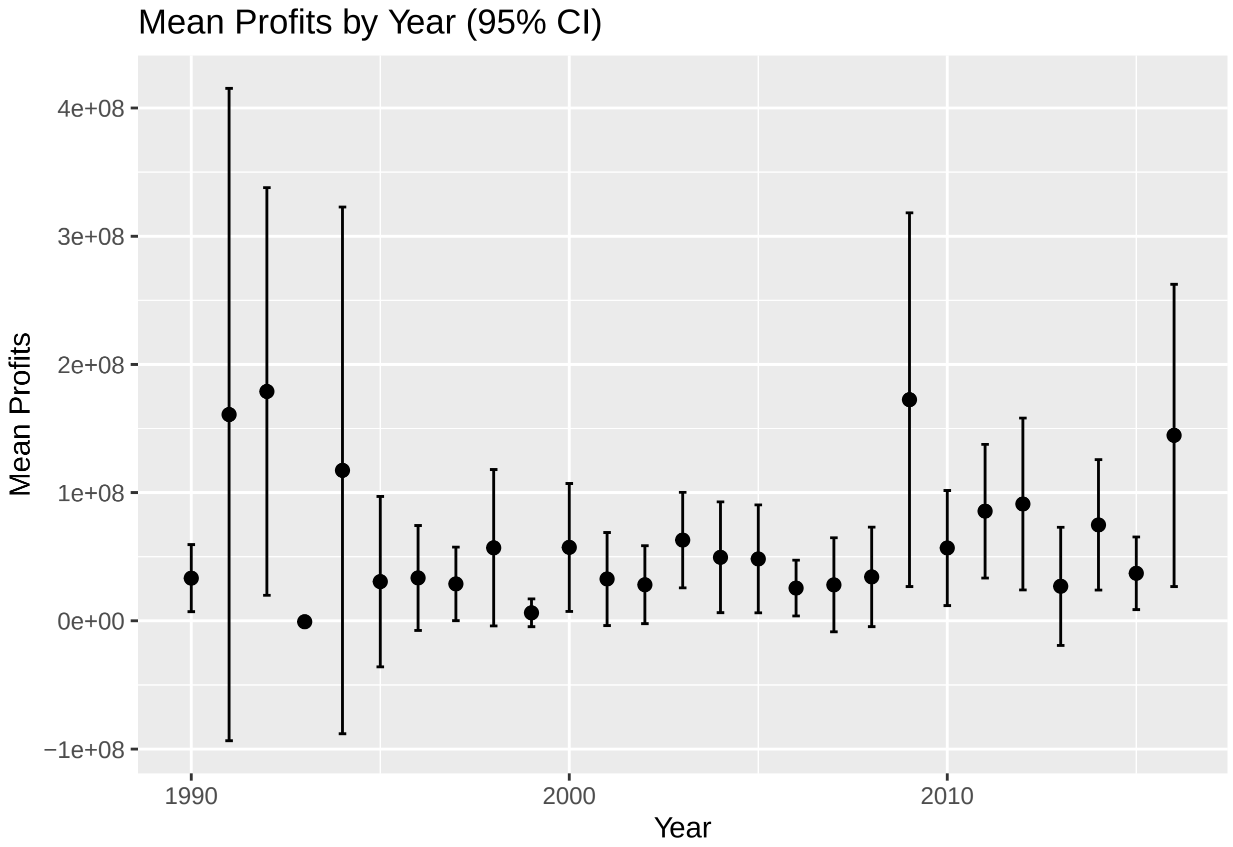


Your Answer:

We observe an exponential positive relationship between mean profits and average ratings

step c

```
z2 <- qnorm(0.975)
year_profits <- movies %>%
  group_by(release_year) %>%
  summarize(mean_profits = mean(profits, na.rm = TRUE),
            error = sd(profits, na.rm = TRUE)/sqrt(n()),
            .groups = "drop") %>%
  mutate(lower = mean_profits - z2*error,
         higher = mean_profits + z2*error)

ggplot(year_profits, aes(x = release_year , y = mean_profits)) +
  geom_point(size = 2) +
  geom_errorbar(aes(ymin = lower, ymax = higher), width = 0.2) +
  labs(title = "Mean Profits by Year (95% CI)", x = "Year", y = "Mean Profits")
```



Your Answer:

In this scatterplot we observe that the periods 1990-1995 and 2009 were quite profitable years for movies.

step d

Your Answer:

As the original argument between Bob and Chantal was about quality vs quantity, lets look at what the data proposes. For this argument, the second diagram proves to be most relevant, as it plots mean profits against average ratings for movies. From this, we observe that there is an exponential positive relationship between the two, thus Bob's claim that higher quality movies bring in the most profits holds true, but only to an extent, as it is likely possible to pump out more mediocre-quality movies than it is high-quality movies.

4 Week 4

1. There is another argument going on in the movie studio. *Bob* claims that production budgets are getting out of hand, and that the studio should focus on making cheaper movies. *Chantal* disagrees. She tells Bob that "Every dollar we spend on movie production is more than offset by the increase in movie profits' '.

a. Set up a regression model to test Chantal's claim, and estimate it. That is, estimate:

$$\text{Profits}_i = \beta_0 + \beta_1 \text{Budget}_i + \varepsilon_i.$$

Print a summary of your estimated model. [**2 points**]

b. What is the estimated value of β_1 and how do you interpet it? [**2 points**]

c. Test for the null hypothesis that $\beta_1 \geq 0$. Report the p-value and state your conclusion. [**2 points**]

d. Next, estimate the model

$$\text{Log Profits}_i = \beta_0 + \beta_1 \text{Log Budget}_i + \varepsilon_i.$$

When creating the variables Log Profits and Log Budget, make sure that movies with a Revenue or Budget of zero are assigned the value “NA”. Print a summary of your estimated model [**2 points**]

e. What is the estimated value of β_1 and how do you interpet it? [**2 points**]

f. Which model has better fit? The level-level model or the log-log model? Explain. [**2 points**]

g. Who do you think is correct? Bob or Chantal? What would you advise the movie studio to do? [**2 points**]

step a

```
movies$Profits <- movies$revenue - movies$budget
lm_a <- lm(Profits ~ budget, data = movies)
summary(lm_a)
```

```
## 
## Call:
## lm(formula = Profits ~ budget, data = movies)
## 
## Residuals:
##        Min         1Q     Median         3Q        Max 
## -420678243  -37658344   14833690   17600279 1964947903 
## 
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -1.760e+07  7.165e+06  -2.456   0.0144 *  
## budget       2.547e+00  1.302e-01  19.566   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 131700000 on 503 degrees of freedom
## Multiple R-squared:  0.4322, Adjusted R-squared:  0.431 
## F-statistic: 382.8 on 1 and 503 DF,  p-value: < 2.2e-16
```

Your Answer:

The estimated coefficient on budget is positive and highly significant.

step b

```
coef(lm_a)["budget"]
```

```
##   budget 
## 2.546909
```

Your Answer:

The estimated value of β_1 (the coefficient on budget) is 2.547.

step c

```
co <- summary(lm_a)$coefficients["budget", ]
tval <- co["t value"]
df   <- lm_a$df.residual

p_one_sided <- pt(tval, df)
p_one_sided
```

```
## t value
##       1
```

Your Answer:

One-sided p-value = 1. There is no evidence that $\beta_1 < 0$, and the estimate is strongly positive. Therefore, Chantal's claim seems more valid.

step d

```
movies$budget[movies$budget == 0]   <- NA
movies$revenue[movies$revenue == 0] <- NA
movies$Profits[movies$Profits <= 0] <- NA
movies$LogBudget  <- log(movies$budget)
movies$LogProfits <- log(movies$Profits)
lm_d <- lm(LogProfits ~ LogBudget, data = movies)
summary(lm_d)
```

```
##
## Call:
## lm(formula = LogProfits ~ LogBudget, data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.3205 -0.6734  0.2365  0.8935  3.6926
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.12020    1.06215   5.762 2.41e-08 ***
## LogBudget    0.67348    0.06171  10.914  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.429 on 253 degrees of freedom
##   (250 observations deleted due to missingness)
## Multiple R-squared:  0.3201, Adjusted R-squared:  0.3174
## F-statistic: 119.1 on 1 and 253 DF,  p-value: < 2.2e-16
```

Your Answer:

The estimated model is $\log(\text{Profits})_i = 6.12 + 0.673 \log(\text{Budget})_i$. The coefficient on Log Budget is positive and highly significant.

step e

```r
coef(summary(lm_d))["LogBudget", "Estimate"]
```

```
## [1] 0.673483
```

Your Answer:

The estimated value of β_1 is 0.673.
This implies that, on average, a 1% increase in a movie's budget increases profits by about 0.67%. The coefficient can be interpreted as the elasticity of profits with respect to budget, and it is positive and significant.

step f

```r
summary(lm_a)$r.squared
```

```
## [1] 0.4321711
```

```r
summary(lm_d)$r.squared
```

```
## [1] 0.3201098
```

Your Answer:

lm_a = 0.432 and lm_d = 0.320. The model with the higher R^2 value fits the data better. Therefore, the level-level model explains more of the variation in profits and provides a better fit than the log-log model.

step g

```r
summary(lm_a)
```

```
##
## Call:
## lm(formula = Profits ~ budget, data = movies)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -420678243  -37658344   14833690   17600279 1964947903
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.760e+07  7.165e+06  -2.456   0.0144 *
## budget       2.547e+00  1.302e-01  19.566   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 131700000 on 503 degrees of freedom
## Multiple R-squared:  0.4322, Adjusted R-squared:  0.431
## F-statistic: 382.8 on 1 and 503 DF,  p-value: < 2.2e-16
```

```r
summary(lm_d)
```

```
##
## Call:
## lm(formula = LogProfits ~ LogBudget, data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.3205 -0.6734  0.2365  0.8935  3.6926
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  6.12020    1.06215   5.762 2.41e-08 ***
## LogBudget    0.67348    0.06171  10.914  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.429 on 253 degrees of freedom
##   (250 observations deleted due to missingness)
## Multiple R-squared:  0.3201, Adjusted R-squared:  0.3174
## F-statistic: 119.1 on 1 and 253 DF,  p-value: < 2.2e-16
```

```r
coef(lm_a)["budget"]
```

```
##   budget
## 2.546909
```

```r
coef(lm_d)["LogBudget"]
```

```
## LogBudget
##  0.673483
```

Your Answer:

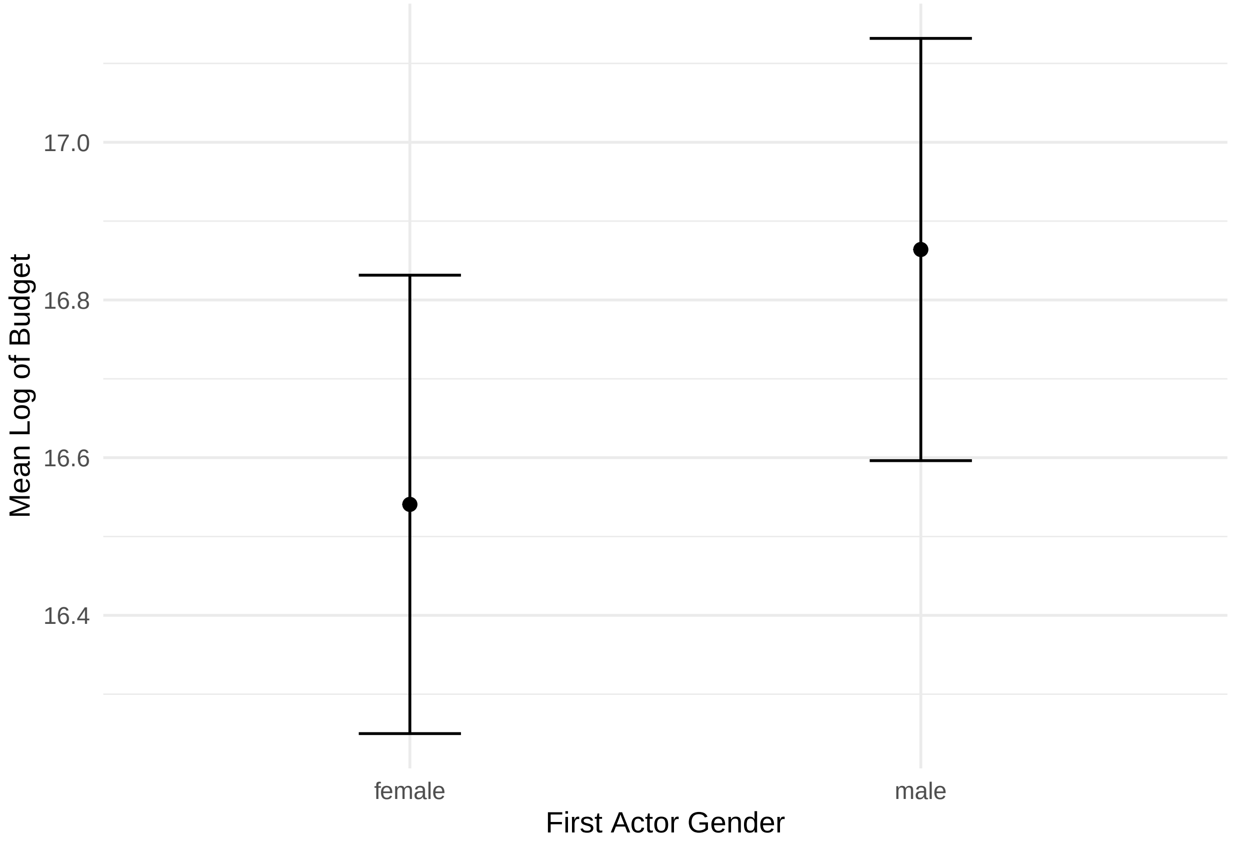
In the level–level model, the estimated coefficient on Budget is around 2.55, meaning that each additional dollar spent on production increases profits by about $2.55.
In the log–log model, the coefficient on Log Budget is 0.67, which implies that a 1% increase in a movie's budget is associated with about a 0.67% increase in profits.
Therefore, Chantal appears to be correct, since higher budgets generally lead to higher profits. However, the effect is not one-to-one, so the studio should invest more in promising movies, rather than uniformly increasing budgets.

2

a. Make a plot with a 95% confidence interval with the mean log of budget on the y-axis, and whether the first actor of the movie is male or female on the x-axis. What do you conclude? [**2 points**]
b. Estimate the following simple OLS model: $log(budget)_i = \beta_0 + \beta_1 (FirstActorMale)_i + \varepsilon_i$. Is the estimated coefficient for β_1 significantly different from zero? How do you interpret its estimate, and how does this relate to your conclusion in 2a? [**2 points**]
c. Now, have a close look at your data frame. Can you find any instances of male first actors who are wrongly labeled as being female, or vice versa? What would such mislabelling mean for the coefficient you estimated under 2b? [**2 points**]

step a

```
sum_budget <- movies %>%
  filter(!is.na(budget), budget > 0, !is.na(first_actor_gender)) %>%
  mutate(log_budget = log(budget)) %>%
  group_by(first_actor_gender) %>%
  summarise(mean_log_budget = mean(log_budget, na.rm = TRUE),
    se = sd(log_budget, na.rm = TRUE) / sqrt(n()))


 ggplot(sum_budget, aes(x = first_actor_gender, y = mean_log_budget)) +
  geom_point(size = 2, color = "black") +
  geom_errorbar(aes(ymin = mean_log_budget - 1.96*se,
                    ymax = mean_log_budget + 1.96*se),
                width = 0.2, color = "black") +
  labs(x = "First Actor Gender",
       y = "Mean Log of Budget",) +
  theme_minimal()
```



Your Answer:

It appears that movies with male lead actors have a higher average log budget than those with female lead actors. However, the overlap in the confidence intervals suggests that the difference is not large.

step b

```r
movies_0 <- movies %>%
  filter(!is.na(budget), budget > 0, !is.na(first_actor_gender)) %>%
  mutate(log_budget = log(budget))


lm3 <- lm(log(budget) ~ first_actor_gender, data = movies_0)
summary(lm3)
```

```
##
## Call:
## lm(formula = log(budget) ~ first_actor_gender, data = movies_0)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.1708  -0.4835   0.3527   1.0459   2.6171
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)             16.5407     0.2144   77.16   <2e-16 ***
## first_actor_gendermale   0.3232     0.2486    1.30    0.194
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.045 on 353 degrees of freedom
## Multiple R-squared:  0.004767,   Adjusted R-squared:  0.001948
## F-statistic: 1.691 on 1 and 353 DF,  p-value: 0.1943
```

Your Answer:

The coefficient for male first actors is positive but not statistically significant ($p = 0.19$). This suggests that movies with male lead actors have slightly higher budgets on average, but the difference is not statistically different from zero, consistent with the plot in (a).

step c

```r
duplicate <- movies[movies$first_actor %in%
               movies$first_actor[duplicated(movies$first_actor)], ]

head(duplicate)
```

```
## # A tibble: 6 x 26
##   index  budget keywords original_language title popularity release_date revenue
##   <dbl>   <dbl> <chr>    <chr>             <chr>      <dbl> <date>         <dbl>
## 1  1174 4   e7 police ~ en                Ride~      25.1  2016-01-14    1.25e8
## 2   214 1.20e8 u.s. ai~ en                The ~      25.8  2000-03-15    3.26e8
## 3  2738 1.25e7 duringc~ en                Abou~      11.8  2014-02-14    4.90e7
## 4  1566 3   e7 wife hu~ en                Abou~      19.9  2002-12-13    1.06e8
## 5  1349 NA     assassi~ en                Ghos~       8.10 1996-12-20   NA
## 6  1201 4   e7 hunter ~ en                Pred~      47.6  2010-07-03    1.26e8
## # i 18 more variables: runtime <dbl>, vote_average <dbl>, vote_count <dbl>,
## #   genre <chr>, release_year <dbl>, release_month <dbl>, release_day <dbl>,
## #   first_actor <chr>, first_actor_gender <chr>, director_first_name <chr>,
## #   director_gender <chr>, profits <dbl>, log_profits <dbl>,
## #   profits_millions <dbl>, vote_average_rounded <dbl>, Profits <dbl>,
## #   LogBudget <dbl>, LogProfits <dbl>
```

Your Answer:

Some actors show up more than once in the data. If any of them have the wrong gender, it will mean that there is a mistake in how the data was entered. This could tamper with the results obtained from 2b.

5 Week 5

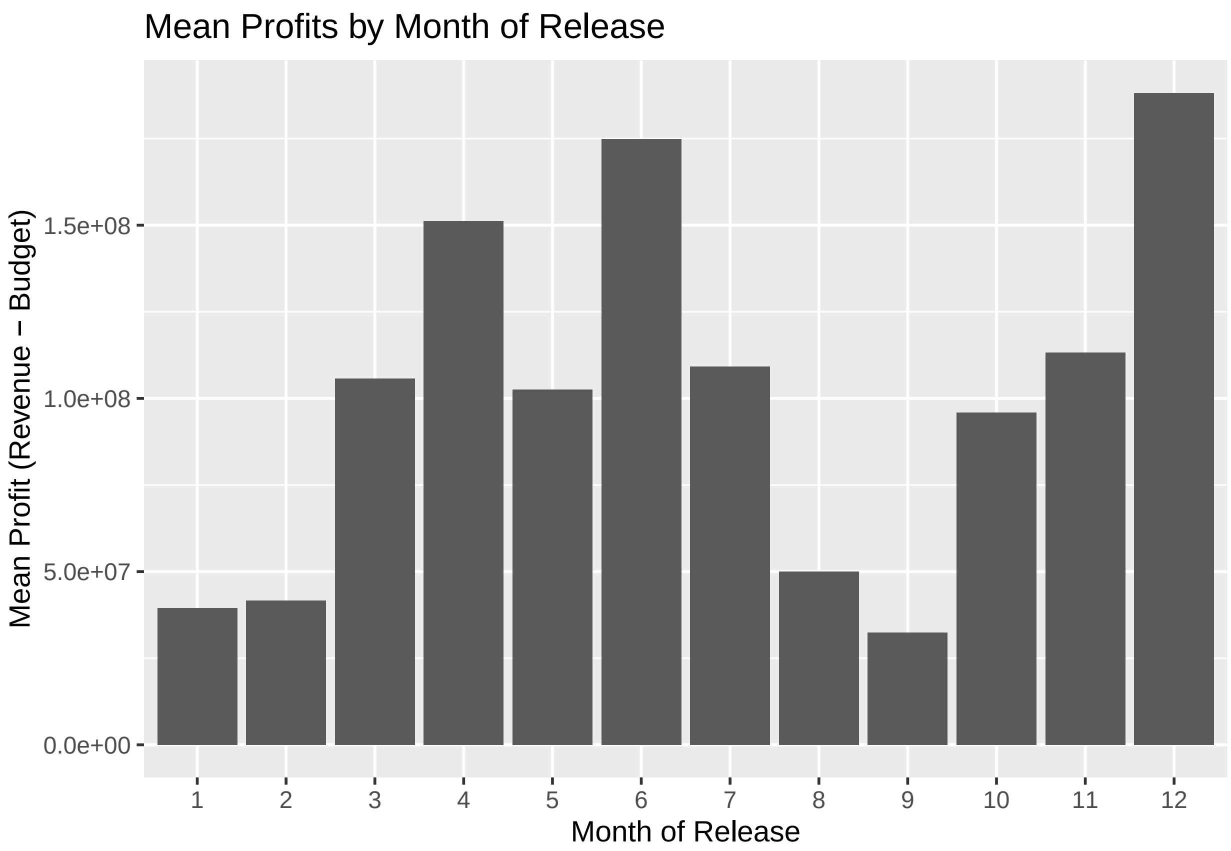
a. Create a plot of the mean profits by month of release. Do you see any indication that month of release matters to the profits of the movie? [**2 points**]
b. Estimate an OLS model which has as dependent variable the log of profits of a movie, and as independent variable the log of budget, a dummy for whether the movie was released in english or not, and a linear term for the month of release. Show a summary of the resulting model and interpret each coefficient. [**4 points**]
c. Test for the hypothesis that the coefficient that belongs to month of release is zero. [**2 points**]
d. Based on your plot in a.) do you consider the choice that month of release enters the model linearly under b.) reasonable? Estimate a specification that allows for a more flexible curve. In this new specification, test for the null hypothesis that month of release does not impact profits. This might require testing multiple terms at once. [**4 points**]
e. One executive at the studio wants to time the release of the movie to a specific month of the year such that they can maximize revenue. Based on your model under d.), What would you advise the movie studio regarding the timing of the release of the movie? [**2 points**]

The movie studio that you work at is releasing a new movie in 2026. It will be an English-spoken Thriller movie with a budget of 40,000,0000.

f. Estimate a model that is able to predict the revenue of this movie. Give its predicted revenue and include a 99% prediction interval. [**6 points**]

step a

```r
#WRITE YOUR CODE HERE

movies$profit <- movies$revenue - movies$budget
mean_profit_by_month <- aggregate(profit ~ release_month, data = movies, FUN = mean,
↪  na.rm = TRUE)

ggplot(mean_profit_by_month, aes(x = factor(release_month), y = profit)) +
  geom_bar(stat = "identity") +
  labs(
    title = "Mean Profits by Month of Release",
    x = "Month of Release",
    y = "Mean Profit (Revenue - Budget)"
  )
```



Your Answer:

The highest mean_profit_by_month can be seen during June and December, meaning that the release month matters. The reason behind such results can be the release of holiday-themed movies in these months.

step b

```
movies_positive <- subset(movies, profit > 0 & budget > 0)

movies_positive$english_dummy_var <- ifelse(movies_positive$original_language == "en", 1,
↪  0)

ols_model<-lm(log(profit)~log(budget)+english_dummy_var+release_month, data =
↪  movies_positive)

summary(ols_model)
```

```
##
## Call:
## lm(formula = log(profit) ~ log(budget) + english_dummy_var +
##     release_month, data = movies_positive)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.2345 -0.6628  0.2446  0.9191  3.7516
##
## Coefficients:
```

```
##                     Estimate Std. Error t value Pr(>|t|)    
## (Intercept)         6.53081    1.17907   5.539 7.67e-08 ***
## log(budget)         0.68112    0.06245  10.906  < 2e-16 ***
## english_dummy_var -0.66377    0.65315  -1.016    0.310    
## release_month      0.01517    0.02652   0.572    0.568    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.431 on 251 degrees of freedom
## Multiple R-squared:  0.3237, Adjusted R-squared:  0.3156 
## F-statistic: 40.05 on 3 and 251 DF,  p-value: < 2.2e-16
```

Your Answer:

Residuals: Min 1Q Median 3Q Max -7.2345 -0.6628 0.2446 0.9191 3.7516

Coefficients: Estimate Std. Error t value Pr(>|t|)
(Intercept) 6.53081 1.17907 5.539 7.67e-08 ***log(budget) 0.68112 0.06245 10.906 < 2e-16*** english_dummy_var -0.66377 0.65315 -1.016 0.310
release_month 0.01517 0.02652 0.572 0.568
— Signif. codes: 0 ‘’ ***0.001*** ‘’ *0.01* ‘’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.431 on 251 degrees of freedom Multiple R-squared: 0.3237, Adjusted R-squared: 0.3156 F-statistic: 40.05 on 3 and 251 DF, p-value: < 2.2e-16

The log(budget) coefficient shows that 1% increase in budget is associated with 0.68% increase in profits with p-value being less than 0.001 making it statistically significant. English_dummy_var shows to be statistically insignificant with p-value 0.310. The coefficient for release_month is very small meaning little affect on the profits once every varibale is controlled with p-value big enough making the variable statistically insignificant.

step c

```
#WRITE YOUR CODE HERE

ols_model<-lm(log(profit)~log(budget)+english_dummy_var+release_month, data =
↪  movies_positive)

summary(ols_model)
```

```
##
## Call:
## lm(formula = log(profit) ~ log(budget) + english_dummy_var + 
##     release_month, data = movies_positive)
##
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -7.2345 -0.6628  0.2446  0.9191  3.7516 
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)    
## (Intercept)         6.53081    1.17907   5.539 7.67e-08 ***
## log(budget)         0.68112    0.06245  10.906  < 2e-16 ***
## english_dummy_var -0.66377    0.65315  -1.016    0.310    
## release_month      0.01517    0.02652   0.572    0.568    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.431 on 251 degrees of freedom
## Multiple R-squared:  0.3237, Adjusted R-squared:  0.3156
## F-statistic: 40.05 on 3 and 251 DF,  p-value: < 2.2e-16
```

Your Answer:

Release month coefficient: 0.01517 (Std. Error 0.02652, t-statistic 0.572, p-value 0.568)

$H_0 : \beta_{\text{month}} = 0$
$H_1 : \beta_{\text{month}} \neq 0$ with $\alpha = 0.05$. t-statistic $= 0.572$, p-value $= 0.568$, $df = 251$. Two-tailed critical value $= \pm 1.97$. Therefore, since the absolute t-value is less than the critical value, we fail to reject the null hypothesis and conclude that release month does not have a significant linear effect on profits.

step d

```
movies$english <- as.integer(movies$original_language == "en")
movies$log_budget <- log10(movies$budget)


#Clean data, was receiving NA/NAN/InF error
movies <- movies %>%
  mutate(
    profit = revenue - budget,
    log_profits = log(profit),
    log_budget = log10(budget),
    english = as.integer(original_language == "en")
  ) %>%
  filter(
    is.finite(log_profits),
    is.finite(log_budget),
    !is.na(release_month),
    !is.na(english)
  )
```

```
## Warning: There was 1 warning in `mutate()`.
## i In argument: `log_profits = log(profit)`.
## Caused by warning in `log()`:
## ! NaNs produced
```

```
model2 <- lm(log_profits ~ log_budget + english + factor(release_month), data = movies)
summary(model2)
```

```
##
## Call:
## lm(formula = log_profits ~ log_budget + english + factor(release_month),
##     data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.8707 -0.6342  0.1559  0.8405  3.2313
##
## Coefficients:
```

```
##                          Estimate Std. Error t value Pr(>|t|)    
## (Intercept)               7.69494    1.22839   6.264 1.71e-09 ***
## log_budget                1.46158    0.14929   9.790  < 2e-16 ***
## english                  -0.63697    0.66377  -0.960   0.3382    
## factor(release_month)2   -0.92365    0.51914  -1.779   0.0765 .  
## factor(release_month)3   -0.43850    0.51369  -0.854   0.3942    
## factor(release_month)4   -0.06054    0.53999  -0.112   0.9108    
## factor(release_month)5   -0.34568    0.53535  -0.646   0.5191    
## factor(release_month)6    0.24604    0.49770   0.494   0.6215    
## factor(release_month)7   -0.03057    0.48445  -0.063   0.9497    
## factor(release_month)8   -0.60193    0.51576  -1.167   0.2443    
## factor(release_month)9   -0.79270    0.47069  -1.684   0.0935 .  
## factor(release_month)10 -0.50857    0.49577  -1.026   0.3060    
## factor(release_month)11 -0.01487    0.49705  -0.030   0.9762    
## factor(release_month)12  0.01734    0.47637   0.036   0.9710    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.415 on 241 degrees of freedom
## Multiple R-squared:  0.365,  Adjusted R-squared:  0.3307
## F-statistic: 10.65 on 13 and 241 DF,  p-value: < 2.2e-16
```

```
# Joint test: are all month terms zero?
anova(ols_model, model2)
```

```
## Warning in anova.lmlist(object, ...): models with response '"log_profits"'
## removed because response differs from model 1
```

```
## Analysis of Variance Table
##
## Response: log(profit)
##                    Df Sum Sq Mean Sq  F value Pr(>F)    
## log(budget)         1 243.39 243.394 118.8071 <2e-16 ***
## english_dummy_var   1   2.07   2.070   1.0104 0.3158    
## release_month       1   0.67   0.671   0.3274 0.5677    
## Residuals         251 514.21   2.049                    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

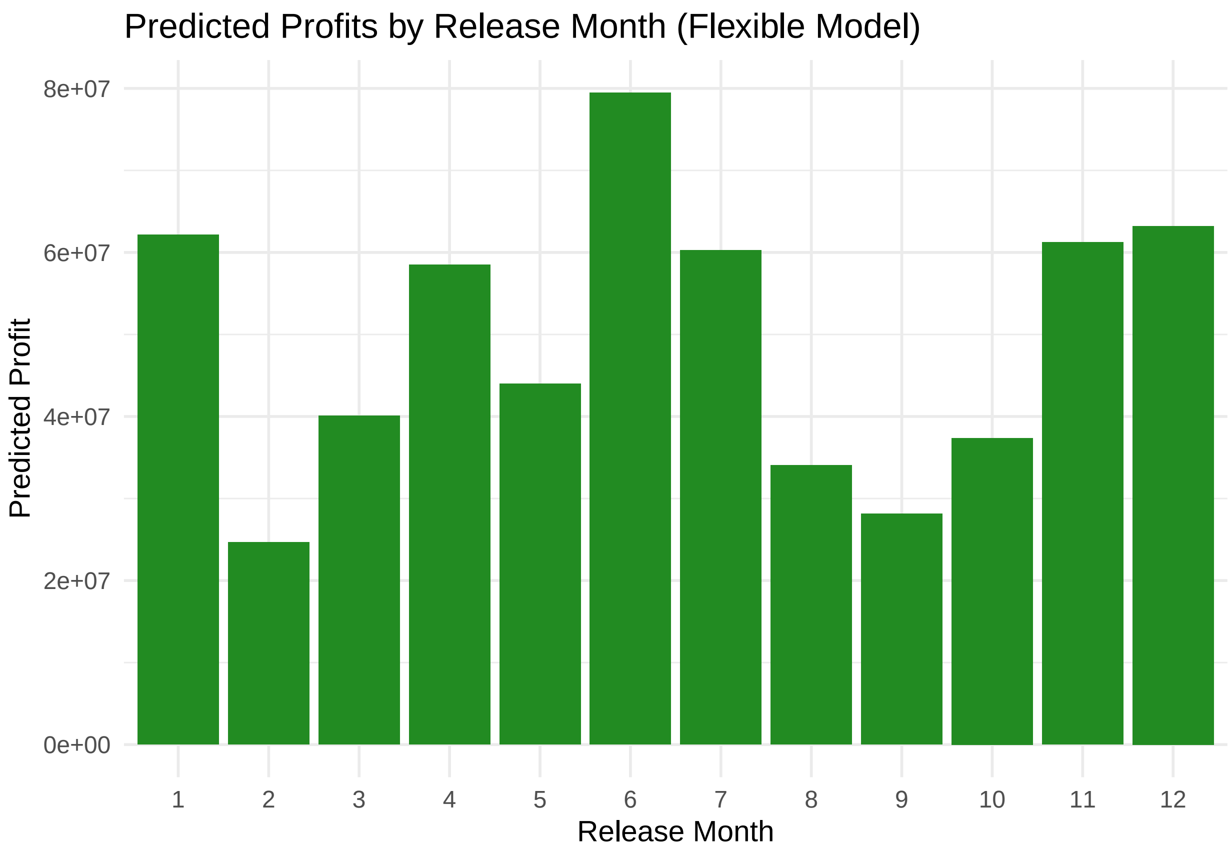
Your Answer:

Based on the mean profits by release month (see part a), the pattern does not appear strictly linear, suggesting that a more flexible modeling approach may be warranted. We estimated a new regression with month of release as a factor variable, allowing each month its own effect.

However, the joint hypothesis test (ANOVA) for the significance of the month dummies shows that they are not collectively statistically significant ($p > 0.05$). This means we fail to reject the null hypothesis that month of release does not impact profits, even when allowing for a flexible functional form. Thus, neither the linear nor the flexible specification provides strong evidence that month of release matters for profits in this data.

step e

```
testdata <- movies %>%
  summarize(
    log_budget = mean(log_budget, na.rm = TRUE),
    english = 1
  ) %>%
  crossing(release_month = 1:12)

testdata$pred_log_profit <- predict(model2, newdata = testdata)
testdata$pred_profit <- exp(testdata$pred_log_profit)

ggplot(testdata, aes(x = factor(release_month), y = pred_profit)) +
  geom_col(fill = "forestgreen") +
  labs(
    x = "Release Month",
    y = "Predicted Profit",
    title = "Predicted Profits by Release Month (Flexible Model)"
  ) +
  theme_minimal()
```

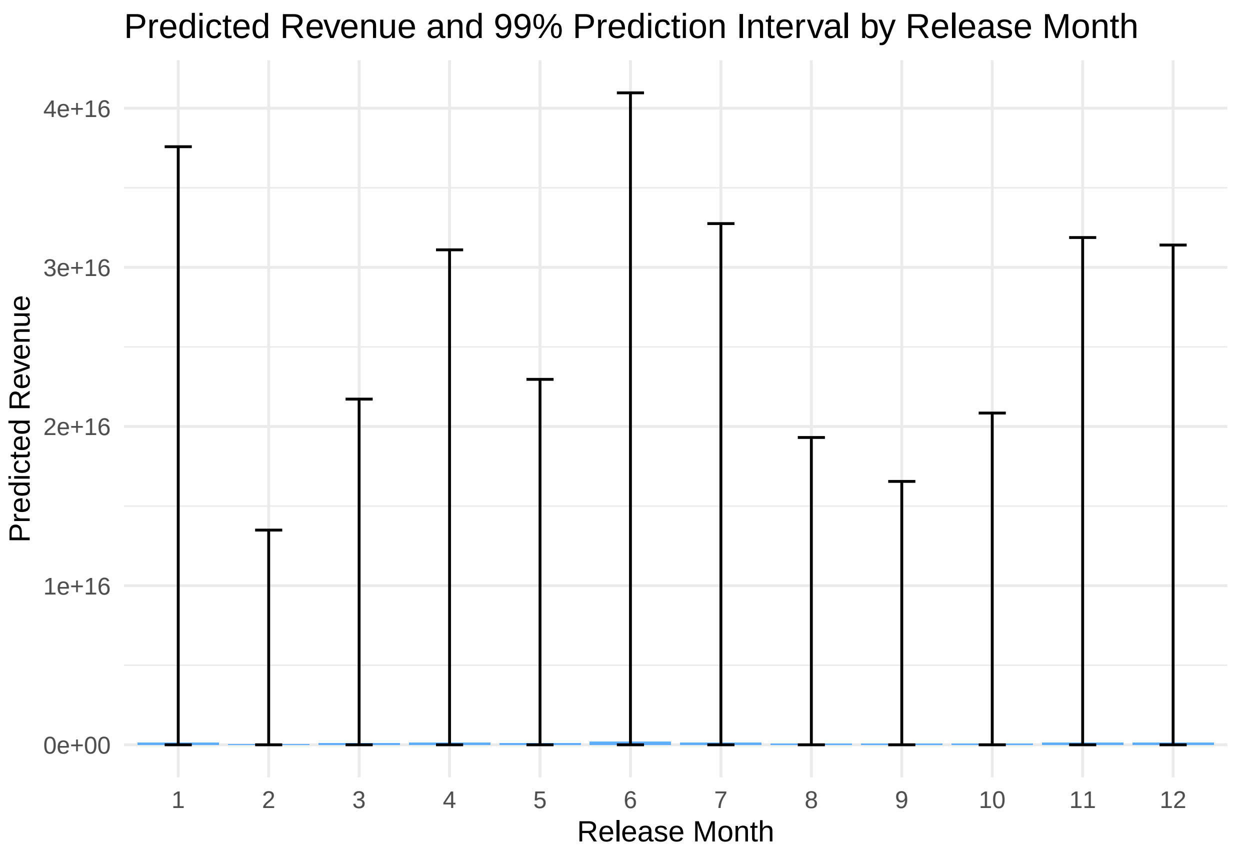


Your Answer:

Using the data from (D), we can create a predicted profitability graph by release month. In this graph, we can observe that the highest expected profits are in the month of June by a wide margin, followed by November, December and January which are all quite close to each other.

step f

```r
new_movie <- tibble(
  log_budget = log(4e7),
  english = 1,
  release_month = 1:12
)

# Use flexible model to get profit prediction for each month
pred <- predict(model2, newdata = new_movie, interval = "prediction", level = 0.99)
pred_df <- cbind(new_movie, as.data.frame(pred))
pred_df$pred_profit <- exp(pred_df$fit)
pred_df$pred_lwr <- exp(pred_df$lwr)
pred_df$pred_upr <- exp(pred_df$upr)

# Add budget back to get revenue prediction
pred_df <- pred_df %>%
  mutate(
    predicted_revenue = pred_profit + 4e7,
    lower_99 = pred_lwr + 4e7,
    upper_99 = pred_upr + 4e7
  )

# Plot predicted revenue with 99% prediction interval for each month
ggplot(pred_df, aes(x = factor(release_month), y = predicted_revenue)) +
  geom_col(fill = "dodgerblue", alpha = 0.7) +
  geom_errorbar(aes(ymin = lower_99, ymax = upper_99), width = 0.3, color = "black") +
  labs(
    x = "Release Month",
    y = "Predicted Revenue",
    title = "Predicted Revenue and 99% Prediction Interval by Release Month"
  ) +
  theme_minimal()
```



```r
# Show results for best month in a table as before
best_row <- pred_df[which.max(pred_df$pred_profit), ]
knitr::kable(best_row, digits = 0)
```

	log_budget	english	release_month	fit	lwr	upr	pred_profit	pred_lwr	pred_upr	predicted_revenue	lower_99	upper_99
6	18	1	6	33	28	38	1.919052e+14	8998210670	4.096515e+16	1.919052e+14	899038210670	4.096515e+16

Your Answer:

Using the flexible model, we predicted revenue for the new English Thriller with a $40,000,000 budget across all release months. The plot above shows the predicted revenue and 99% prediction interval for each month.

The model suggests June (month 6) would yield the highest expected revenue, with predicted revenues of 192 trillion dollars and a 99% prediction interval of between 899 billion and 40.97 quadrillion.

There is a considerably wide interval which reflects high uncertainty in movie revenue predictions, however June is the best month according to the model.

««« < Updated upstream

======= »»»> Stashed changes