# TidyTuesday_Pixar_Films

Yang Yuetong

2025-09-14

### Suggested questions:

Why are some values missing in the datasets? Which films have the highest score in each rating system? Are there distinct differences in ratings? Download the box_office dataset from the {pixarfilms} package. How does the box_office_us_canada value compare to the various ratings? Is the trend different for box_office_worldwide?

```r
# Import Pixar Films and Public Response Data from Github

pixar_films <- readr::read_csv('https://raw.githubusercontent.com/rfordatascience/tidytuesday/main/data/
```

```
## Rows: 27 Columns: 5
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (2): film, film_rating
## dbl  (2): number, run_time
## 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.
```

```r
public_response <- readr::read_csv('https://raw.githubusercontent.com/rfordatascience/tidytuesday/main/d
```

```
## Rows: 24 Columns: 5
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): film, cinema_score
## dbl (3): rotten_tomatoes, metacritic, critics_choice
##
## 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.
```

```r
box_office <- readr::read_csv('https://raw.githubusercontent.com/erictleung/pixarfilms/master/data-raw/b
```

```
## Rows: 28 Columns: 5
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): film
## dbl (4): budget, box_office_us_canada, box_office_other, box_office_worldwide
##
## 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.
```

## Cleaning

```r
#Clean missing values to be recorded as NA
pixar_films <- pixar_films %>%
  mutate(
    film_rating = na_if(film_rating, "N/A"),
    film_rating = na_if(film_rating, "Not Rated")
)
```

```r
#drop extra row index column
pixar_films <- pixar_films %>%
  select(-number)
```

```r
#Find which column has missing values
colSums(is.na(pixar_films))
```

```
##         film release_date     run_time  film_rating
##            1            0            2            4
```

Upon inspecting the pixar_films dataset, film name is missing for the film that was released in 2023-06-16. A quick search of the release date and the run time would suggest that the film is Elemental. However, the run time is 103 minutes which does not match the data. run_time and film_rating also have missing values which could be due to inconsistent formatting.

```r
colSums(is.na(public_response))
```

```
##            film rotten_tomatoes      metacritic    cinema_score  critics_choice
##               0               1               1               2               3
```

In the public_response dataset, the ratings for Luca is not available. We may drop the entire row. rating from cinema score is missing for Soul but other ratings from other critics could be useful.

```r
public_response <- public_response %>% filter(film != "Luca")
```

Cinema score does not provide numerical scores, making it difficult to compare against other ratings.

```r
public_response <- public_response %>% select(-cinema_score) #drop cinema score column
```

Convert the rating values to long format for data visualisation.

```r
public_response <- public_response %>% pivot_longer(
  cols = c("rotten_tomatoes","metacritic","critics_choice"),
  names_to = "ratings",
  values_to = "ratings_value"
)
```

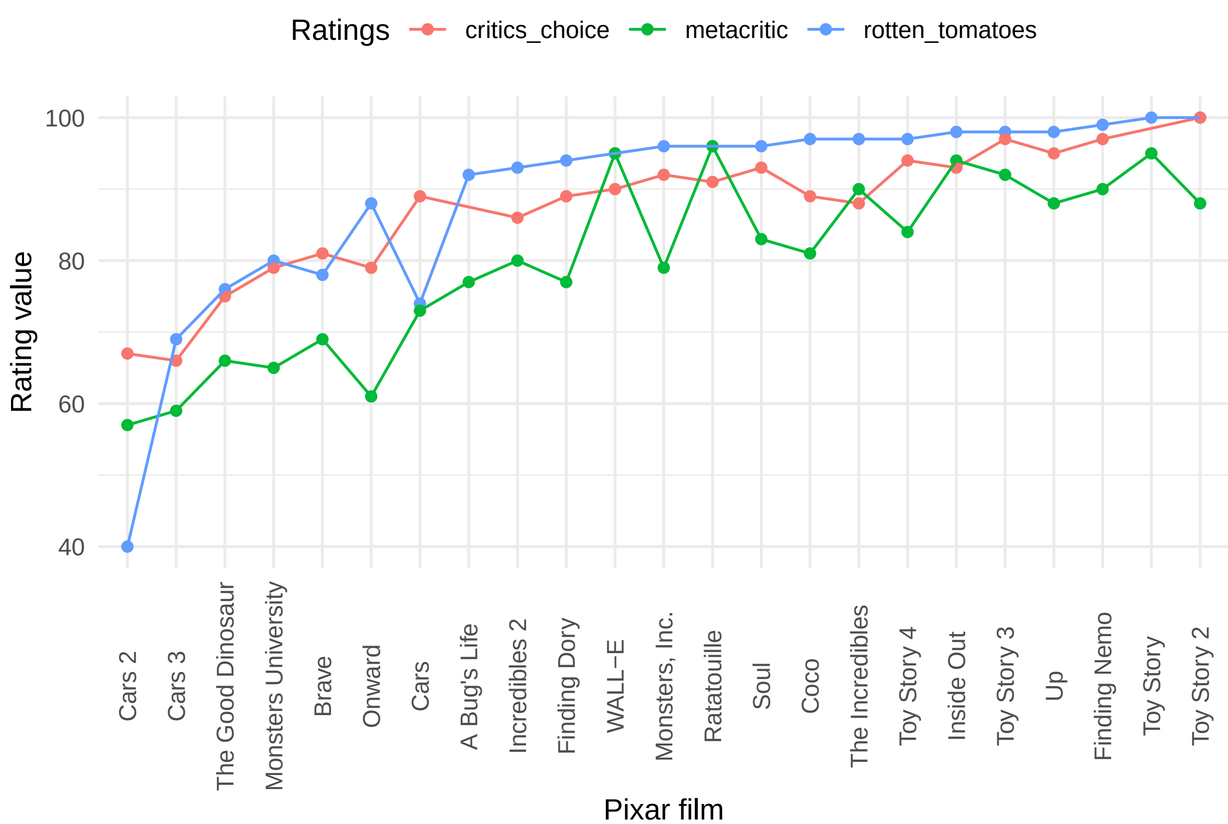
We want to drop rows where ratings is missing, then create a column containing max rating across different rating types, in order to sort the rows from the lowest to highest max rating.

```r
public_response <- public_response %>%
  drop_na(ratings_value) %>%
  group_by(film) %>%
  mutate(max_rating = max(ratings_value, na.rm = FALSE)) %>%
  ungroup() %>%
  mutate(film = fct_reorder(film, max_rating, .desc = FALSE)) %>%
  arrange(film)
```

### Trends for ratings of pixar films across the three critics

Plot a line graph to compare ratings for pixar films.

```r
public_response %>%
  ggplot(aes(x = film, y = ratings_value, col = ratings, group = ratings)) +
  geom_point() +
  geom_line(aes(group = ratings)) +
  scale_fill_brewer(palette = "Set1") +
  labs(x = "Pixar film", y = "Rating value") +
  guides(col = guide_legend(title = "Ratings")) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5),
        legend.position = "top")
```




Firstly, rotten tomatoes has the highest rating for Toy Story 2 and Toy Story. Metacritic has the highest rating for Ratatouille. Critic choices has the highest rating for Toy Story 2. It seems like most people / critics like the classic movies such as Finding Nemo, Toy Story 1 and 2.

## Further statistical analysis to decide if the mean ratings for classic movies and new movies is significantly different

We can categorise the Pixar movies into two categories: New and Classic. Then calculate the mean for each category, and observe the distribution.

After consulting different AI tools (ChatGPT, Gemini): Although the term "classic" is subjective, it generally refers to the early, groundbreaking films that established Pixar's reputation for innovative animation, compelling storytelling, and emotional depth. Film critics, audiences, and industry observers widely agree that the studio's "classic era" includes the films from its founding up to around 2010.

Hence, we will categorise the releases up till Toy Story 3 as "classic".

```
pixar_films <- pixar_films %>% mutate(category = ifelse(as.numeric(format(release_date, "%Y"))<=2010,"cl
```

We will keep the ratings from rotten_tomatoes as a litmus test. Another possible way is to calculate a mean from all three critics for each movie.

```
rotten_tomatoes <- public_response %>% filter (ratings == "rotten_tomatoes")
```

Do a left join for rotten_tomatoes as the left table, pixar_films as the right table.

```
rotten_tomatoes_merged <- left_join(rotten_tomatoes, pixar_films, by = "film",)
```

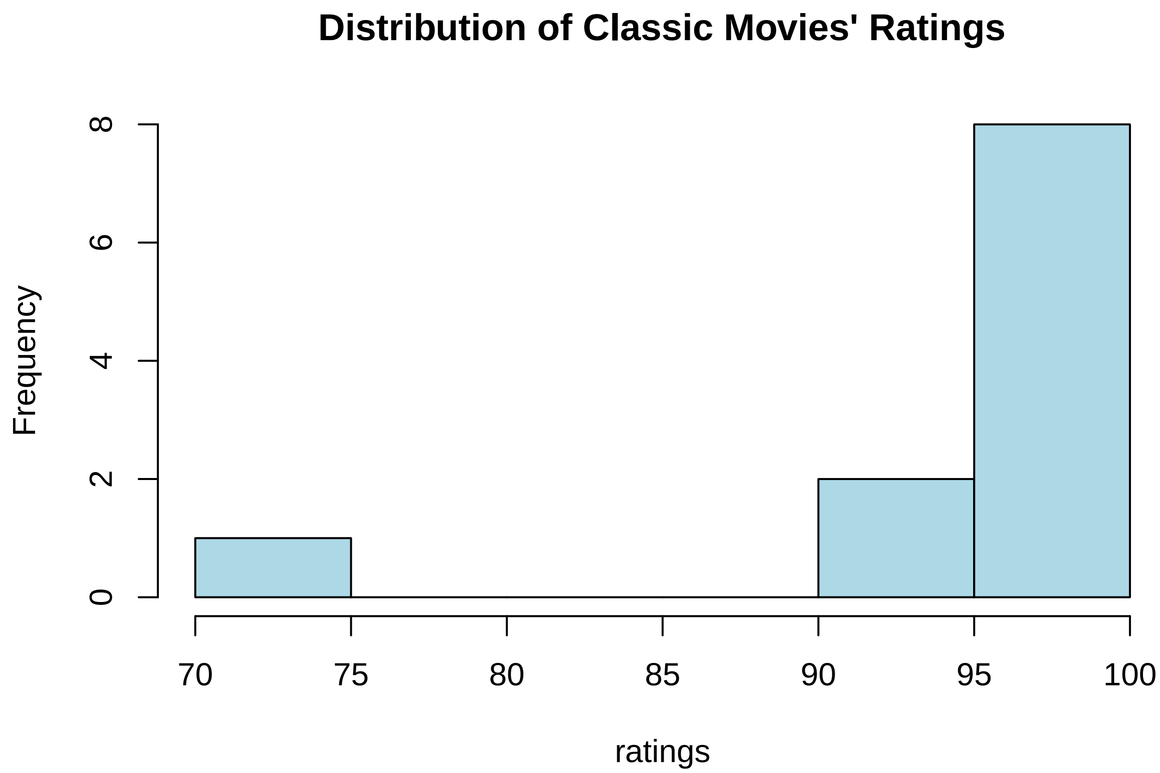
We will calculate the mean ratings using rotten_tomatoes for classic and new categories.

```
classic <- rotten_tomatoes_merged[rotten_tomatoes_merged$category == "classic","ratings_value"]
new <- rotten_tomatoes_merged[rotten_tomatoes_merged$category == "new","ratings_value"]
```

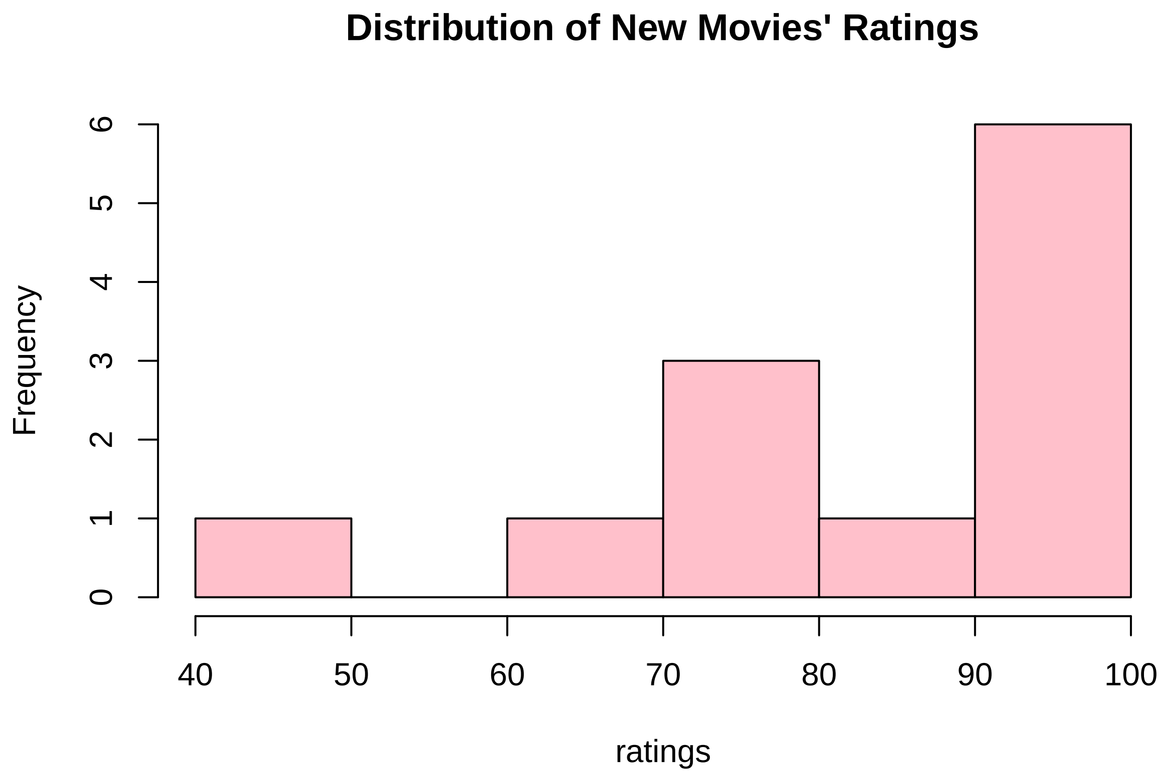
```
mean_classic <- classic$ratings_value %>% mean()
mean_new <- new$ratings_value %>% mean()
```

Observe distributions of classic and new ratings.

```
hist(classic$ratings_value,
     main = "Distribution of Classic Movies' Ratings",
     xlab = "ratings",
     col = "lightblue")
```




```r
hist(new$ratings_value,
     main = "Distribution of New Movies' Ratings",
     xlab = "ratings",
     col = "pink")
```




As n = 11 and 12 for classic and new pixar movies respectively, and the distributions are non-normal, we should do a permutation test to decide if the differences in mean ratings between classic and new movies are significant.

$H_0 : \mu_{\text{classic}} - \mu_{\text{new}} = 0$

$H_1 : \mu_{\text{classic}} - \mu_{\text{new}} \neq 0$

```
#Observed mean

obs <- mean_classic - mean_new
obs
```

```
## [1] 11.16667
```

```
ratings <- rotten_tomatoes_merged$ratings_value
```

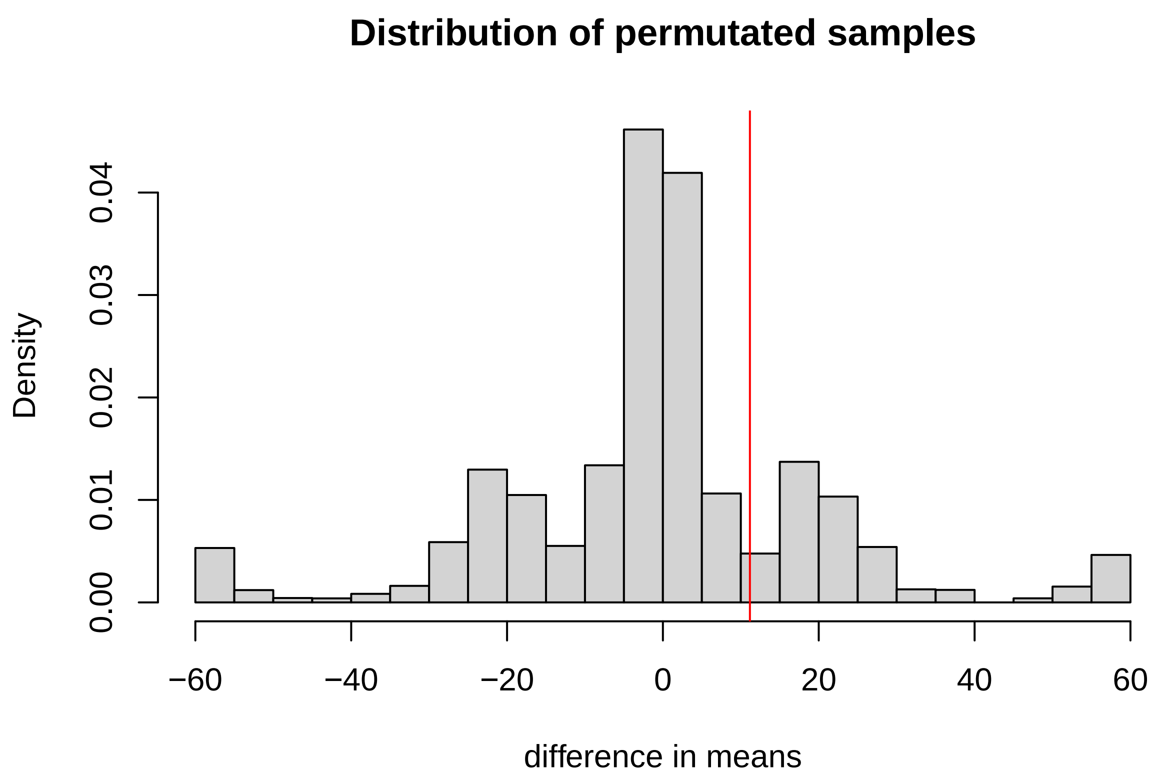
```
#Expected mean under the null hypothesis
N <-100000
set.seed(123)

result <- numeric(N)

for (i in 1:N) {
shuffled <- sample(ratings, replace = FALSE)
classic_perm <- shuffled[1:length(classic)]
new_perm <- shuffled[length(classic)+1:length(classic)+length(new)]
```

```
result[i] = mean(classic_perm) - mean(new_perm)
}
```

```
hist(result, probability = TRUE, main = "Distribution of permutated samples", xlab = "difference in mean
abline (v = obs, col = "red")
```




```
2*((sum(result >= obs) + 1) / (N + 1))
```

```
## [1] 0.4244158
```

Since the p-value is 0.42, which is greater than the 0.05 significance level, we fail to reject the null hypothesis. This indicates that there is insufficient evidence to conclude a significant difference in mean ratings between classic and new Pixar movies.

Although some newer films, such as Inside Out, may have lower ratings compared to earlier releases, they continue to make a meaningful impact on audiences.

### Trends for Box Office

```
colSums(is.na(box_office))
```

```
##                  film               budget box_office_us_canada
```

```
##                     0                    1                    0
##     box_office_other box_office_worldwide
##                     0                    0
```

The missing value is in Luca's budget. If we are not using that column, we can remove it. We can also remove box_office_other.

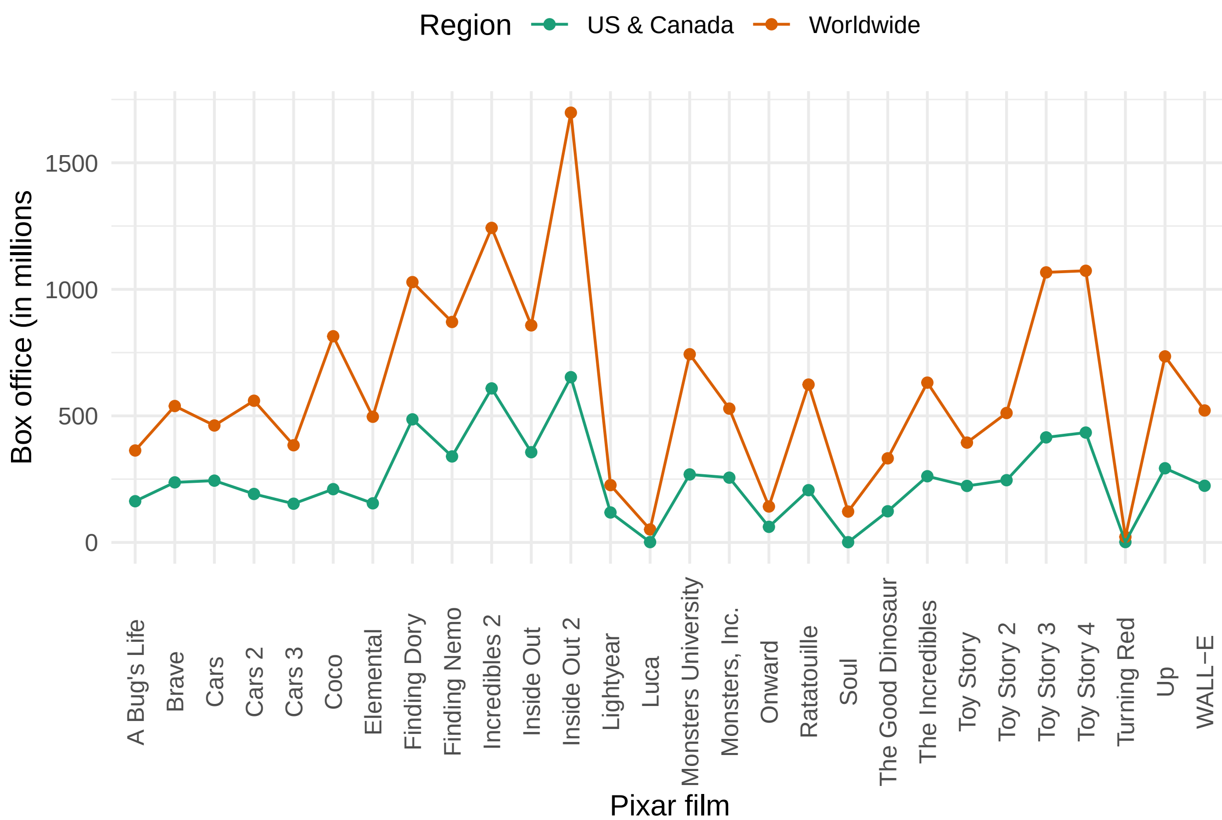
```
box_office <- box_office %>% select(-budget,-box_office_other)
```

We will now compare the box_office in US/Canada vs Worldwide for all of the pixar_films

Similarly, convert the box_office values to long format for data visualisation.

```
box_office <- box_office %>% pivot_longer(
  cols = c("box_office_us_canada","box_office_worldwide"),
  names_to = "region",
  values_to = "box_office"
  ) %>%
  mutate(region = case_when(
    region == "box_office_us_canada" ~ "US & Canada",
    region == "box_office_worldwide" ~ "Worldwide",
    TRUE ~ region
  ))
```

```
box_office %>%
  ggplot(aes(x = film, y = box_office / 1e6, col = region, group = region)) +
  geom_point() +
  geom_line(aes(group = region)) +
  scale_color_brewer(palette = "Dark2") +
  labs(x = "Pixar film", y = "Box office (in millions") +
  guides(col = guide_legend(title = "Region")) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5),
        legend.position = "top")
```




The trend seems to be similar for US/Cananda vs Worldwide. The largest difference is for Inside Out 2 suggesting that alot more people outside US/Canada watch Inside Out 2, compared to other pixar films. This could indicate potential for international marketing for future Inside Out releases, since there is a large proportion of international audience for Inside Out 2.

## AI Tool Declaration

GPT-5, 2.5 Flash were used to refine the sentences and check for code logic.