DATA-PT-EAST-APRIL-041524
Instructor: Alexander Booth
Contributors: Eugenio Elizondo, Andrew Garza, Sierra Sarkis, Marty Thompson, Robert Williard

Project 1 Technical Writeup

Intro to Dataset: Eugenio

The dataset chosen was received from kaggle/datasets for the purpose of project 1 data analysis. This dataset examines various titles on the Netflix streaming platform along with their IMDB Votes and Scores, release year, type, run time and age certification. Our group wished to uncover trends relating to the popularity of Netflix titles by examining how IMDB scores correlate with IMDB votes, the correlation between IMDB scores and release years or runtime, the average IMDB score of shows compares to the score of movies, and what trends can we see when grouping by decades.

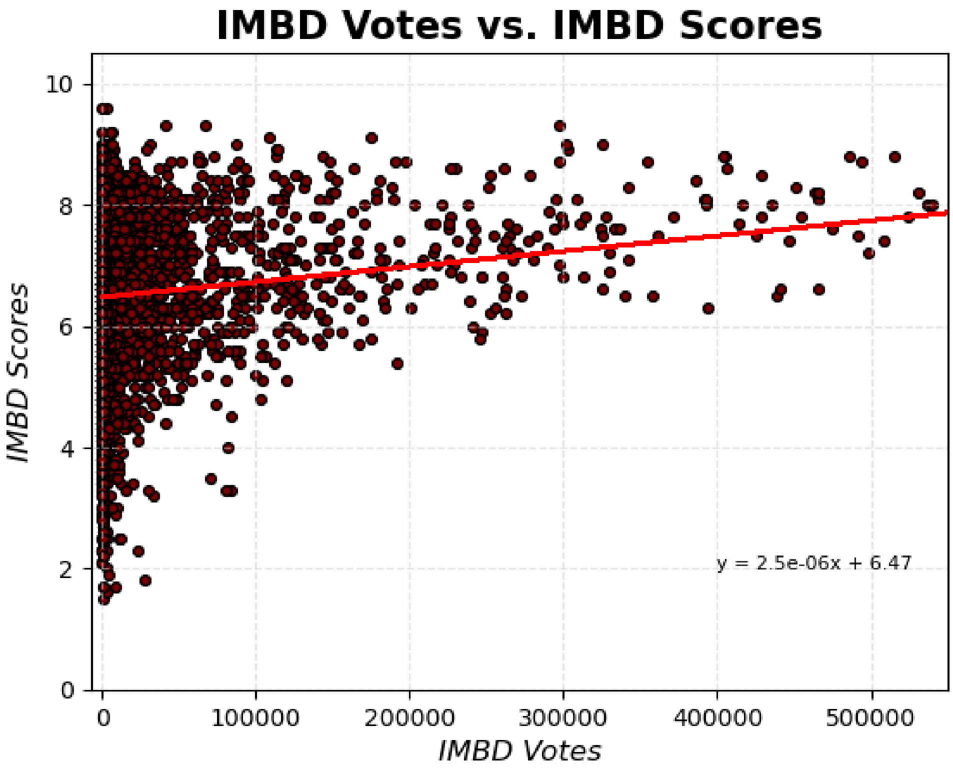
Data Cleaning Process: Marty

1. Initial look at the data to determine column and row content.
2. Replaced nulls in the age_certification column with “Unknown.
3. Dropped nulls from column imdb_votes.
4. Created a clean data frame using specific columns. Dropped ID, age_certification, title, type, description and runtime columns.

Question 1: Eugenio

How do the IMDB scores correlate with IMDB votes?

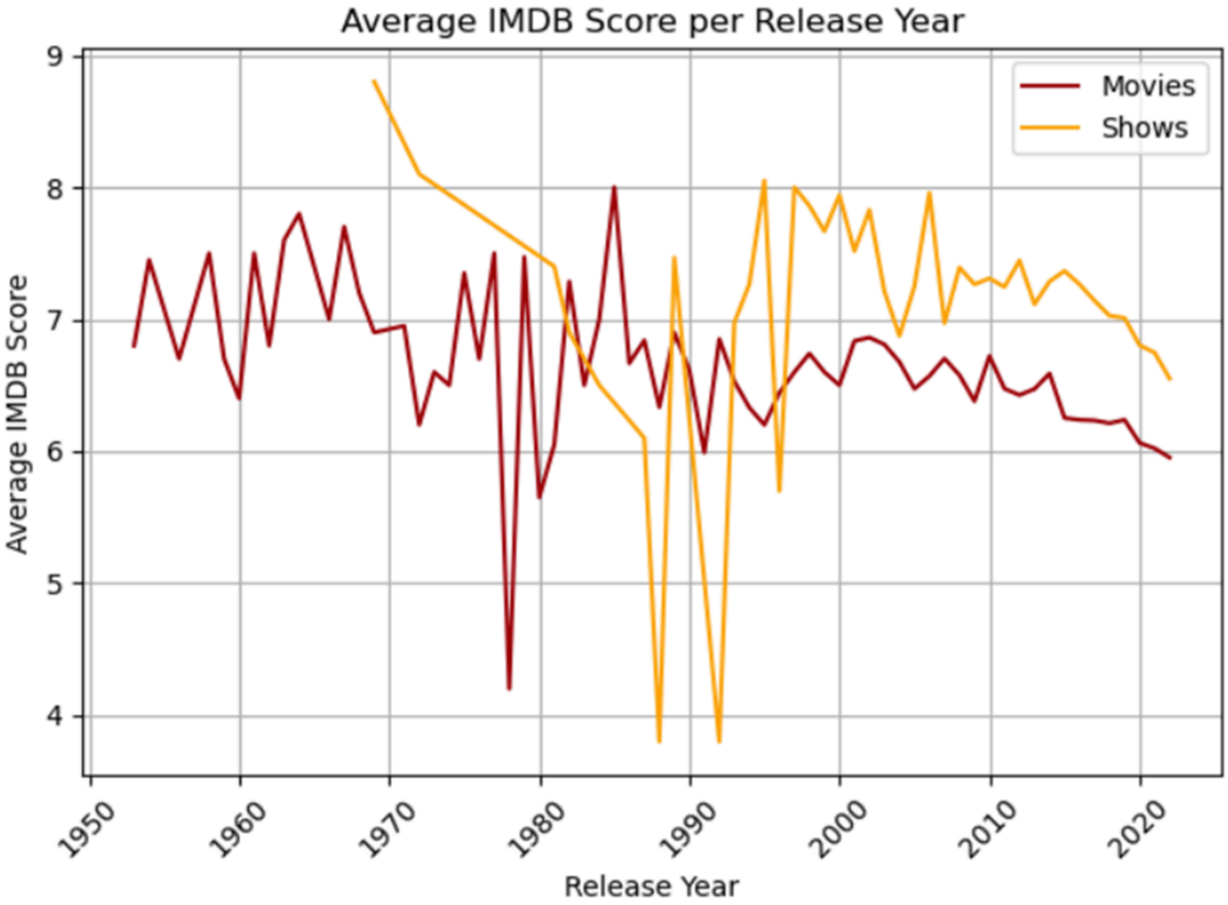
From the Netflix IMDB Scores, the correlation between the relationship between the IMDB scores and IMDB votes had a positive but weak value, r = 0.19, r^2 = 0.036. The graph only shows the value of the votes up to 500,000 due to the data having values that have votes up to 2 million so in order to show the cluster of the plots better, it was necessary to limit the size of the graph.



Notice how the largest cluster of movies is on the left of the graph, meaning that movies with a very small amount of votes greatly influence the line on the graph. Another noticeable observation is that the scatter plots seem to resemble the line of a logarithmic function meaning that movies with a higher vote count would likely have a higher score but would get close to 10 but never reach it. The best fit line was y= 2.5e-06x +6.47, so although a very weak positive correlation, when dealing with more values, the data is more supported on not being by probability but by correlation.

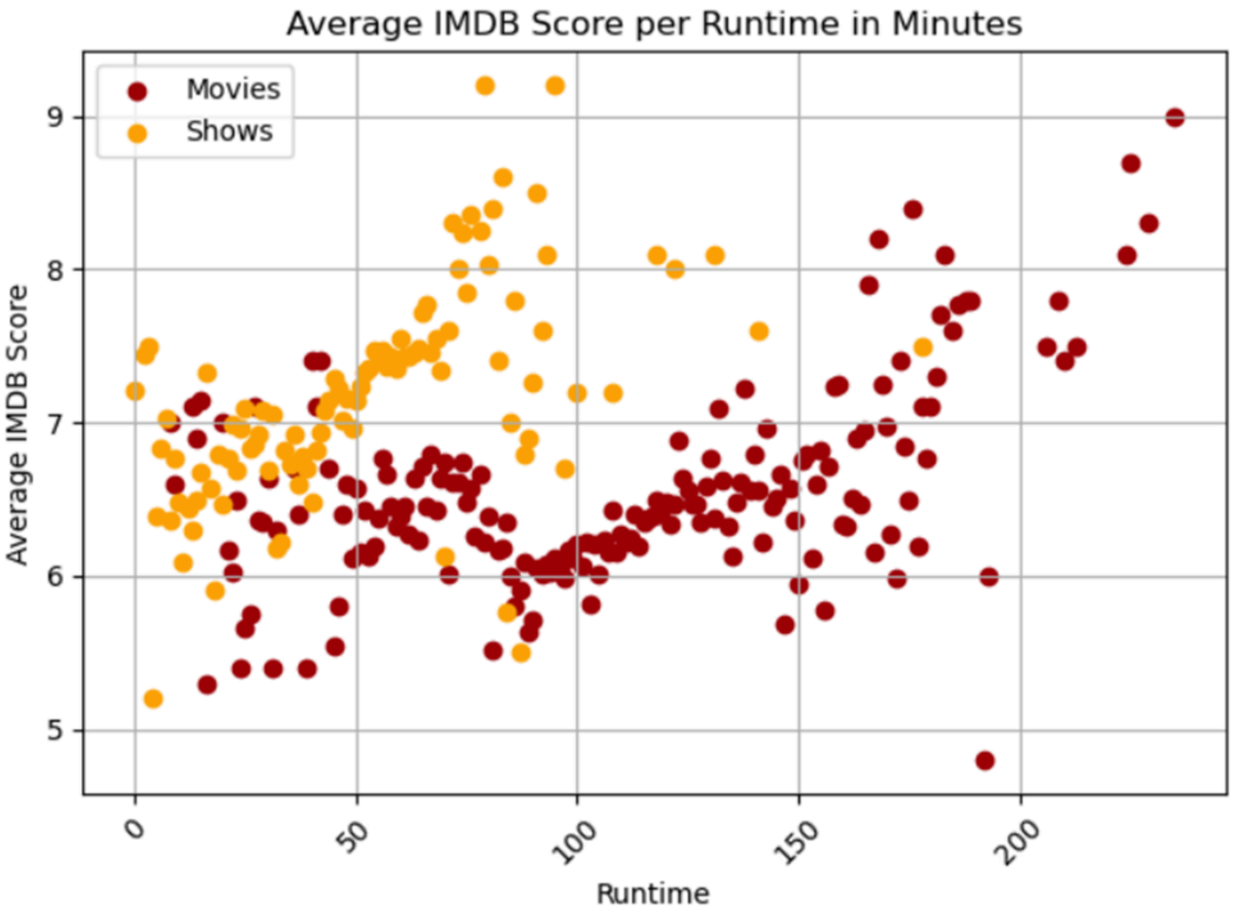
Question 2: Robert

We wanted to look for trends comparing scores to the years the shows were released as well as the runtimes.



In this visualization we can see the comparison between IMDB scores and the year both shows and movies were released. The first note is that prior to the late 1990s the average score fluctuated a lot per year. This is due to there being fewer movies and shows during those years which prevents average scores to find a middle ground.

Once we get into the 2000s the scores still fluctuate but they are steadier than before by not letting one bad or one good score dictate the overall average score for the year. We also see that scores for shows and movies in more recent release years are trending down, with shows performing better than movies.



Next we looked to see if IMDB scores were affected by different runtimes. Here we can see that the longer a show or movie is, the better the average score. We also see that there are outliers of different runtimes that have swayed the average score, because the amount of movies or shows with that runtime is limited.

Question 3: Sierra

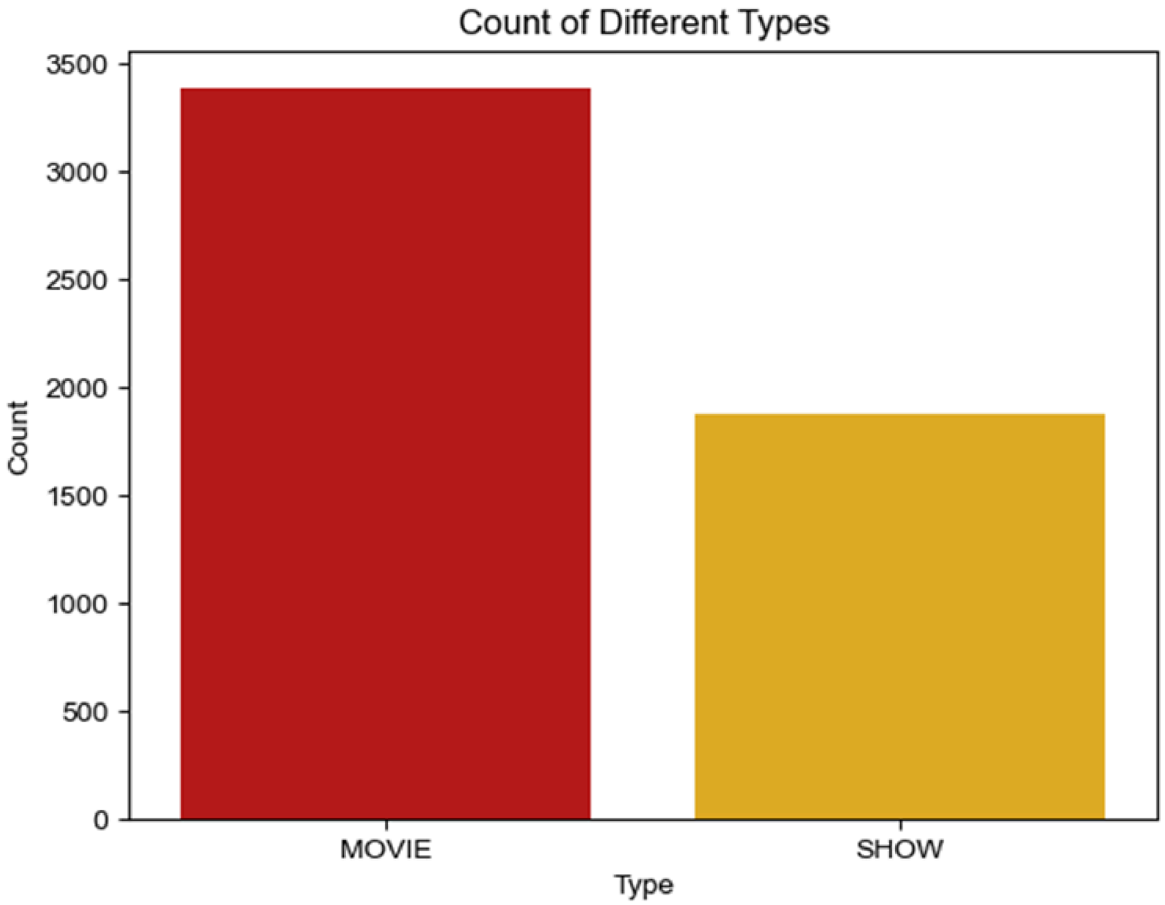
The third question we posed towards the dataset was: are there any significant differences when comparing IMDB Score and title type. In our dataset title type refers to the categorization of “MOVIE” or “SHOW”.

To begin understanding the differences between movies and shows a countplot was created based on type.

```
sns.countplot(x='type', data=clean_df, palette={"MOVIE": "#D00000", "SHOW":
"#FFBA08"})

sns.set_style("whitegrid", {'grid.linestyle': '--'})
```

```
plt.xlabel('Type')

plt.ylabel('Count')

plt.title('Count of Different Types')
```

This produced the following visual:

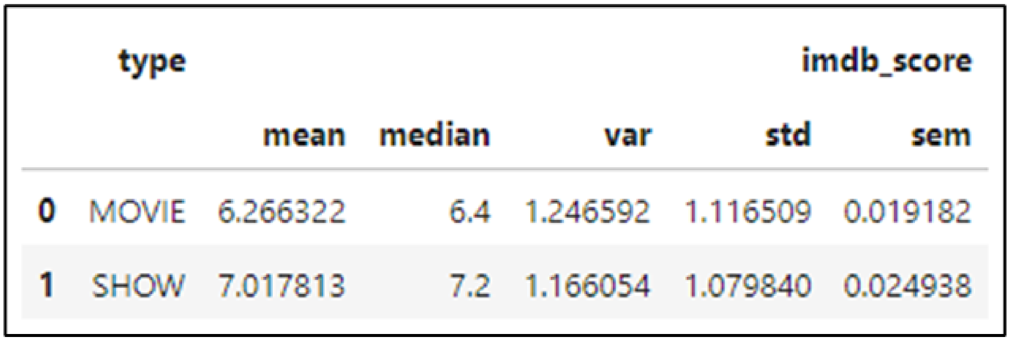


From this we can understand that we have a higher count of movies than shows in our dataset. This is important to keep in mind when determining the impact of the differences between types.

To further understand the differences between title type, we created a statistical summary. We accomplished this by aggregating the imdb_score column to find the mean, median, variance, standard deviation, and sem then grouped the values by type:

```
cols_agg = {
    "imdb_score": ["mean", "median", "var", "std", "sem"]
}
```

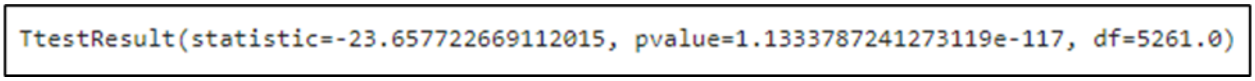
```
leaderboard = clean_df.groupby(["type"]).agg(cols_agg).reset_index()
leaderboard
```



	type	imdb_score				
		mean	median	var	std	sem
0	MOVIE	6.266322	6.4	1.246592	1.116509	0.019182
1	SHOW	7.017813	7.2	1.166054	1.079840	0.024938

We can see that the both title types have a pretty normal distribution because the mean and median for each type are very close. Additionally, the mean imdb_score for type SHOW is slightly higher than that of the MOVIE type. There does not appear to be a significant difference between types. To solidify this hypothesis that there is no significant difference, we ran a ttest.

```
from scipy.stats import ttest_ind


MOVIE = clean_df[clean_df['type'] == 'MOVIE']
SHOW = clean_df[clean_df['type'] == 'SHOW']

ttest_ind(MOVIE['imdb_score'], SHOW['imdb_score'])
```

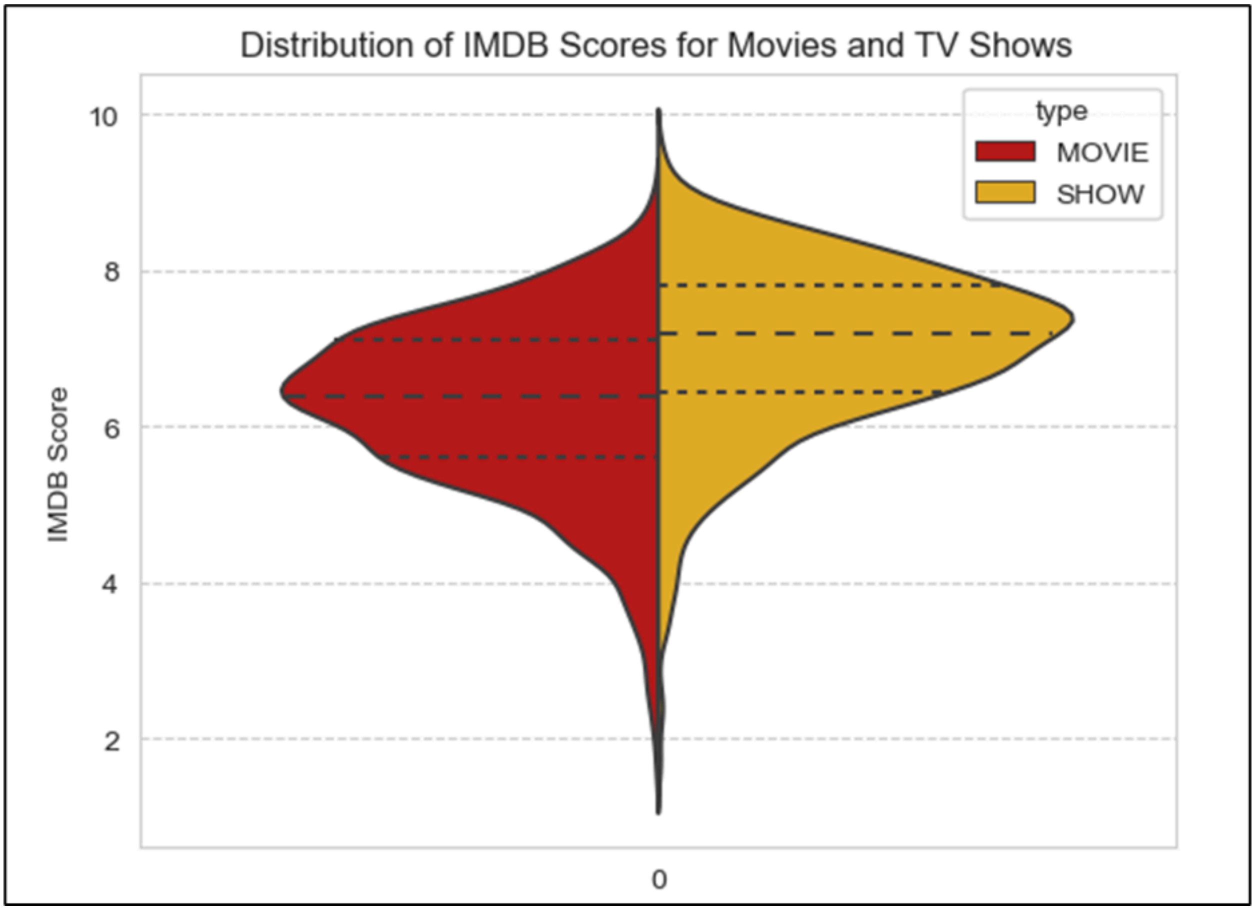


```
TtestResult(statistic=-23.657722669112015, pvalue=1.1333787241273119e-117, df=5261.0)
```

The pvalue result is greater than .05 so it confirms that there is no statistical significance.

Finally, to visualize the average imdb_score of both types we created a split violin plot.

```
clean_df["dummy"]=0
sns.violinplot(data=clean_df, x='dummy', y='imdb_score', inner="quart", split=True,
hue="type", palette={"MOVIE": "#D00000", "SHOW": "#FFBA08"})
sns.set_style("whitegrid", {'grid.linestyle': '--'})
plt.title('Distribution of IMDB Scores for Movies and TV Shows')
plt.ylabel('IMDB Score')plt.xlabel('')
plt.show()
```



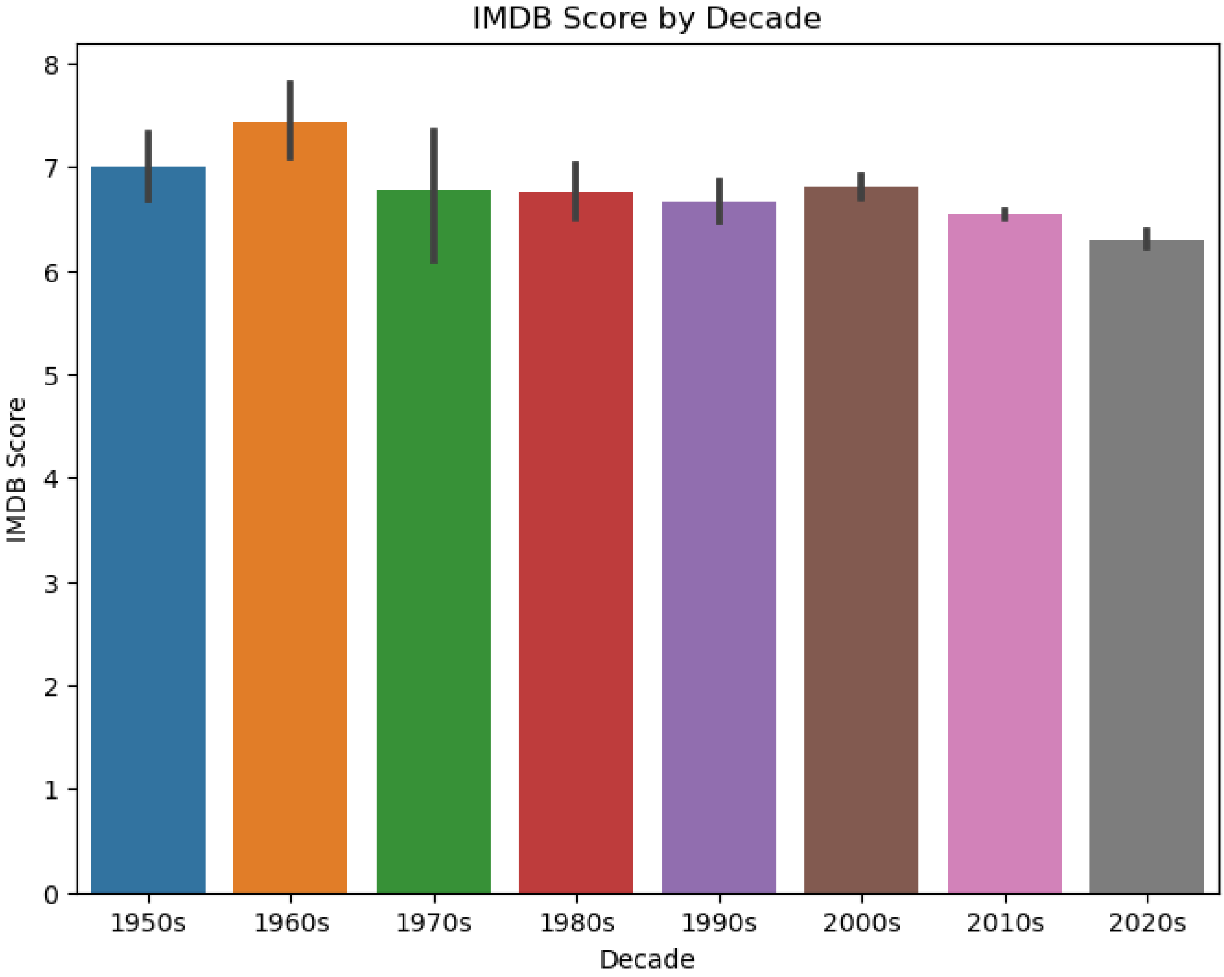
Question 4: Drew

By taking a broader view of the release dates provided in the dataset, we can examine trends from a decade perspective. To do this, we performed a pd.cut of the release date values provided to add a new column to the DataFrame which assigns a decade value to each entry (e.g. a movie released in 1952 gets a decade value of 1950s).

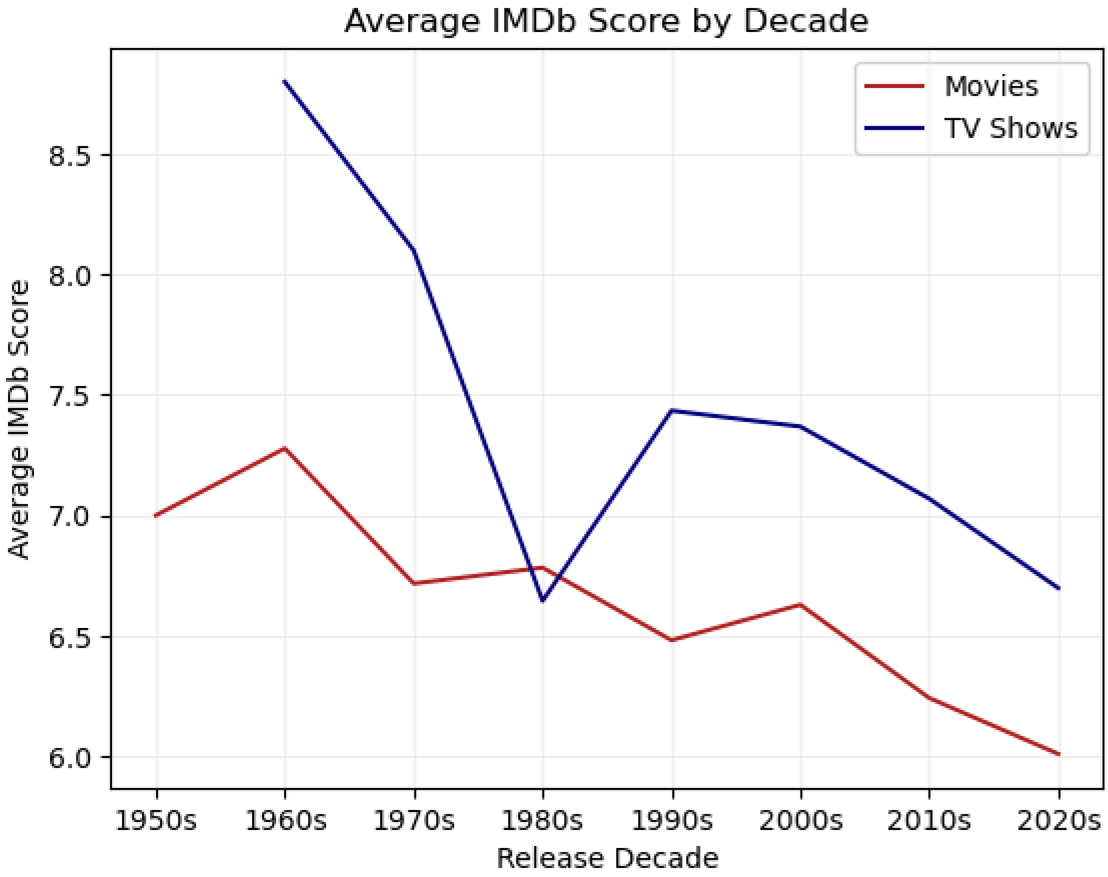
```
bins = [1950, 1960, 1970, 1980, 1990, 2000, 2010, 2020, 2030]
labels = ["1950s", "1960s", "1970s", "1980s", "1990s", "2000s", "2010s", "2020s"]
```

```
clean_df["decade"] = pd.cut (clean_df.release_year, bins, labels=labels, include_lowest=True)
```

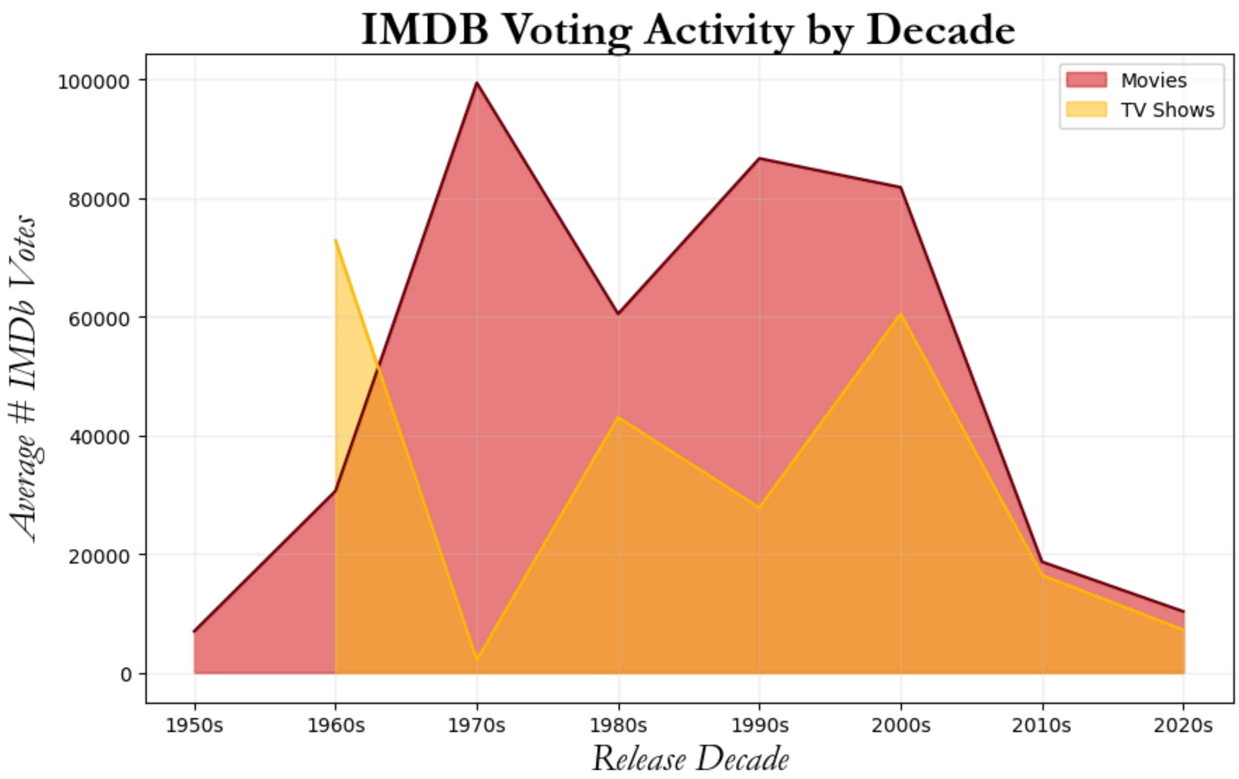
With the data divided, we can visualize some of the data in easier to digest columns. For example, a bar chart comparing IMDb Score by Decade which shows a slowly decreasing score as our dataset moves forward through the decades.



Translating this into a line chart and separating the data between Movies and TV Shows shows us a similar correlation with a marked dip in the TV show ratings from the 1980s. The largest variance here is within our data from the 1970s but it still follows a loose trend of scores slowly decreasing as we move through the decades.

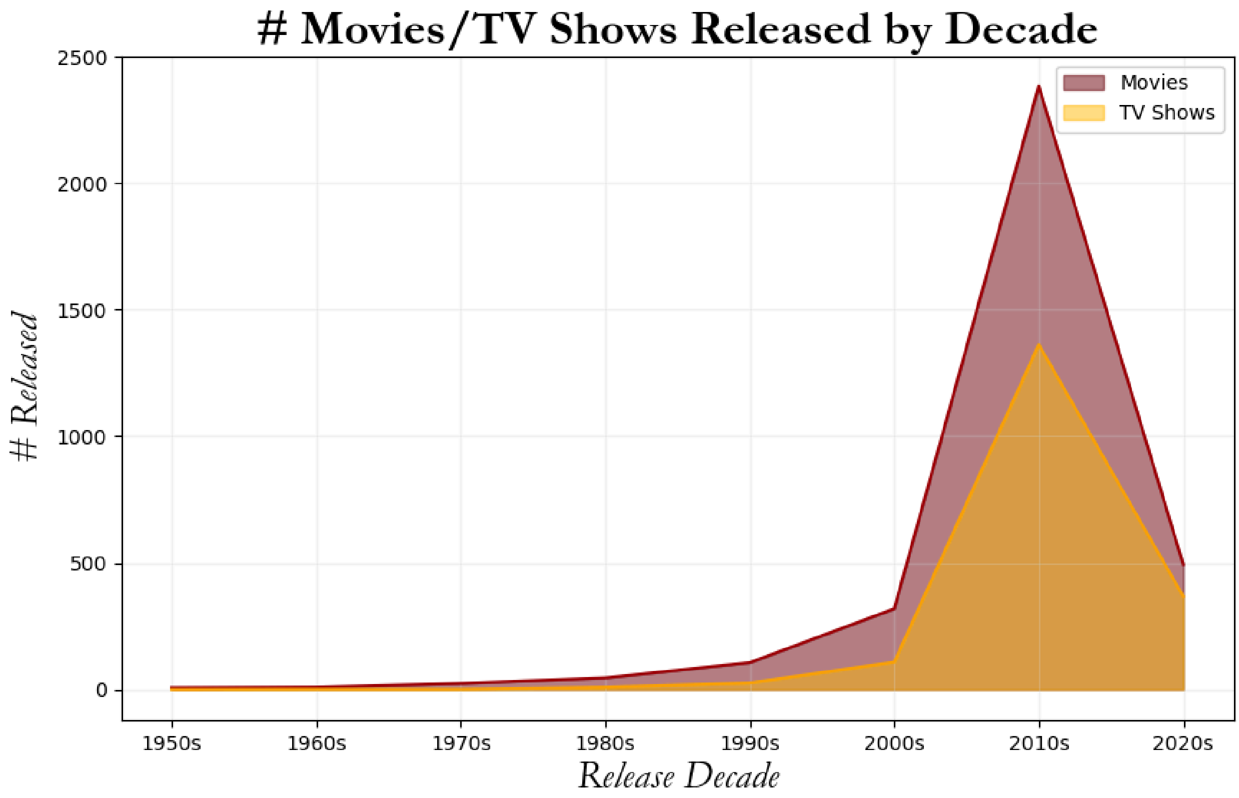


Moving on to the voting activity, we can see a noticeable difference in the average amount of votes on IMDb submitted when comparing Movies and TV Shows. There appears to be an inverse relationship between the two types of data up until the 2000s at which point the activity begins to line up between each type.



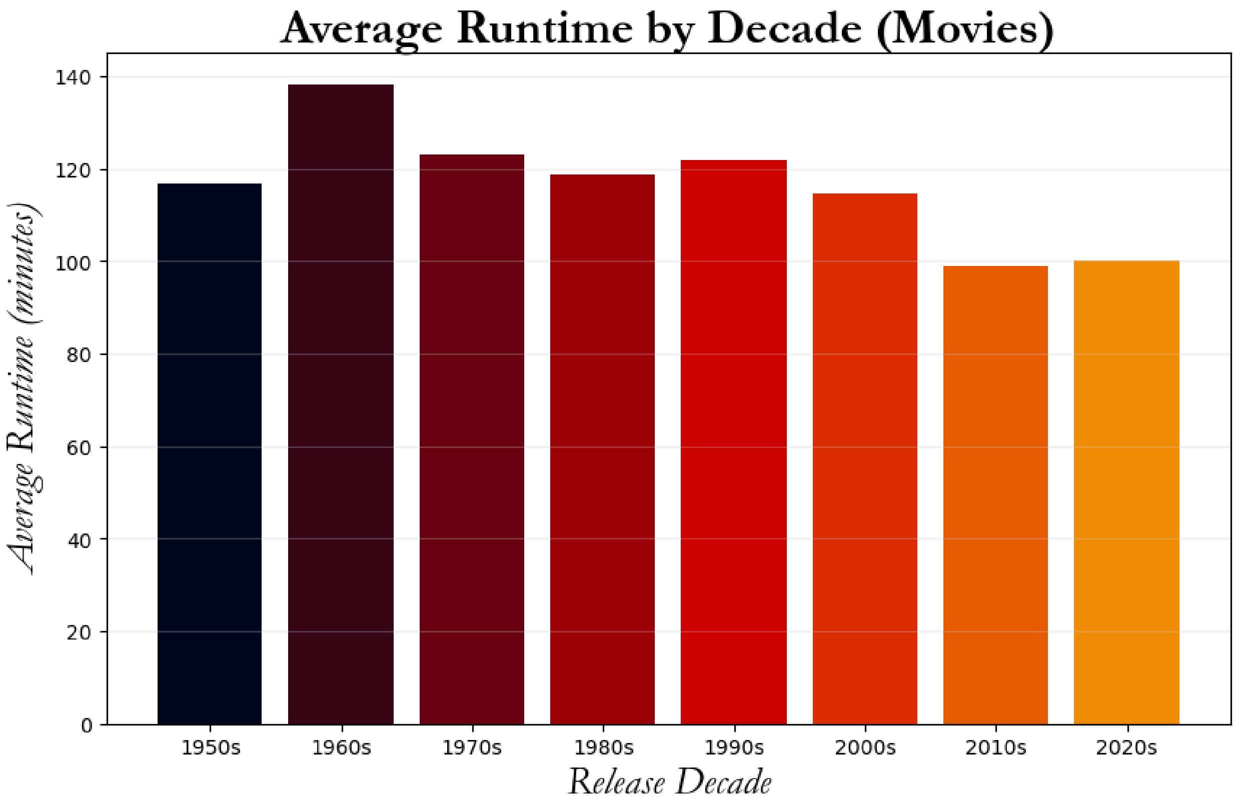
This could indicate an increased willingness for users on IMDb to provide scores for consumed media released in the last two decades.IMDb itself began in the 1980s as a private hobby database. It didn't reach the internet until the 1990s but even then it was in a forum at the beginning. The platform didn't reach widespread popularity until the late 1990s when it garnered the attention of Amazon. Looking through the dataset, it's clear to see that the voting discrepancy slowly began to close the gap as the platform grew.

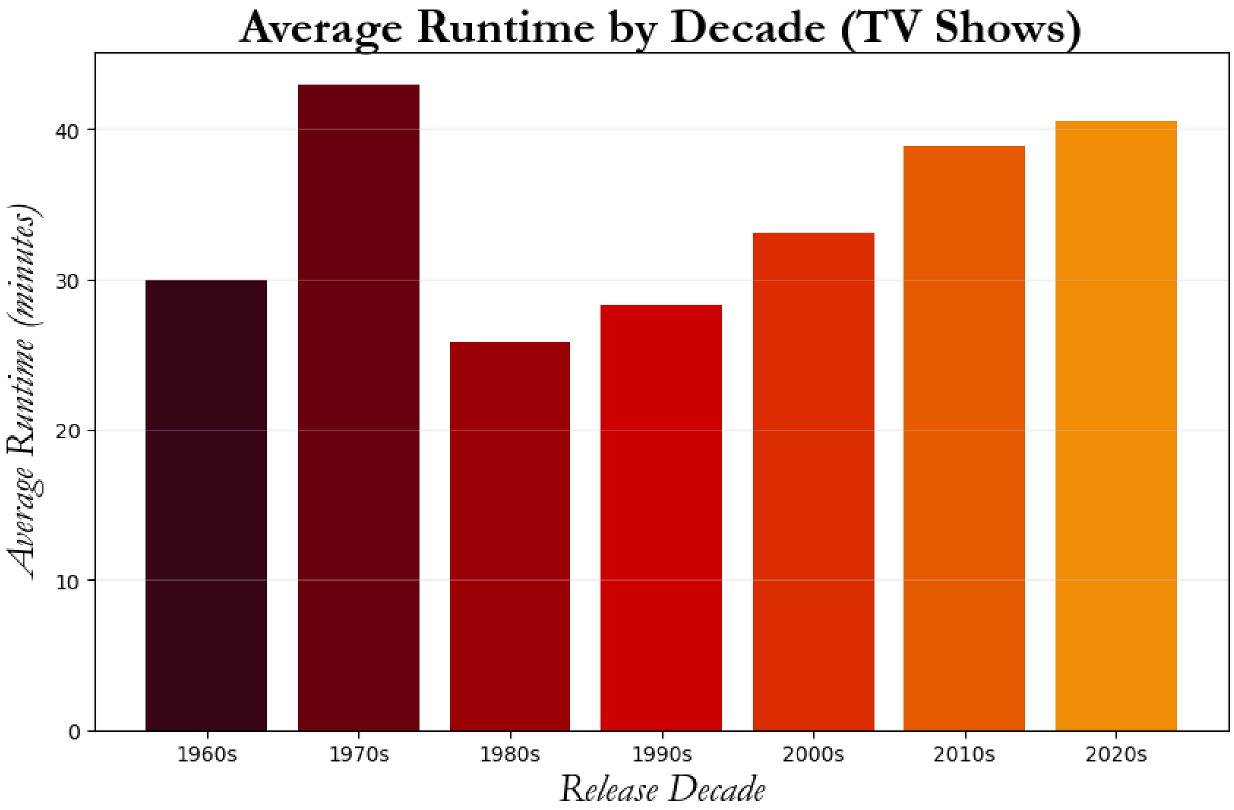
Taking a look at the numbers for how many titles released in each decade, we still notice a difference between movies and tv shows although a much smaller difference.



What may be more of a surprise at first glance is that the runtime between the two types of media seem to be converging as time goes on. Below we have separate bar charts for runtime trends with movies and tv shows.

We have a noticeable difference here with an increase in titles released beginning in the 1990s and a sharp increase after that in the 2000s. While it may be true that the release cadence from Hollywood may have increased as time goes on, this sharp difference may also be explained by a lack of data.


Average Runtime by Decade (Movies)
Average Runtime (minutes)
0
20
40
60
80
100
120
140
1950s
1960s
1970s
1980s
1990s
2000s
2010s
2020s
Release Decade


Average Runtime by Decade (TV Shows)
Average Runtime (minutes)
0
10
20
30
40
1960s
1970s
1980s
1990s
2000s
2010s
2020s
Release Decade

While the runtime for movies appears to be steadily decreasing through the decades, we notice the opposite trend for tv shows with the exception of the 1970s.

There could be many factors which explain the differences here. Firstly, the 1960s held many movies that reached a 3+ hour runtime which is rare these days according to our dataset. This could be explained by a cultural shift from seeing movies as an evening event to wanting a more easily consumed adventure.

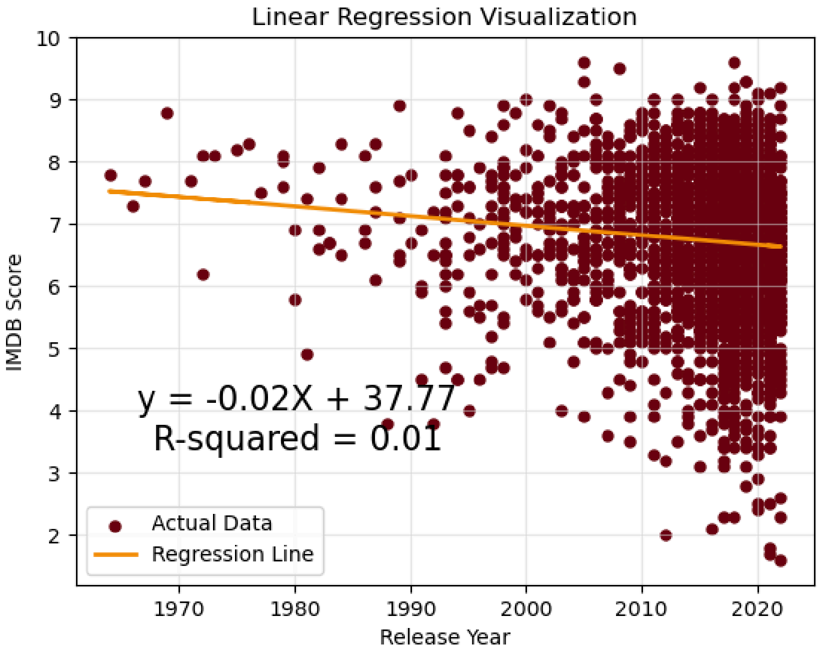
Comparatively, the runtime for tv shows appears to be increasing as the decades move forward. Again, we could see this as a cultural shift from shorter tv shows with a ~24 minute runtime which was aired directly to tv sets and supported by advertisements toward more curated stories made available by internet streaming that is not supplemented by advertisements.

We cannot discount the trends above as being a lack of data however, because we are only seeing data from media contained within Netflix at the time that our dataset was created. For us to identify long-term trends more accurately, we would need to obtain data that is not platform specific or at least gain data from more platforms to expand our dataset. Netflix rotates their media frequently and may only be keeping the movies and tv shows that are popular with their user base.

Statistical Regression: Marty

The Netflix Dataset will be analyzed to determine trends relating to the popularity of Netflix titles by examining how viewers would rate them.

The following linear relationship chart shows how the number of votes affects the imdb score.



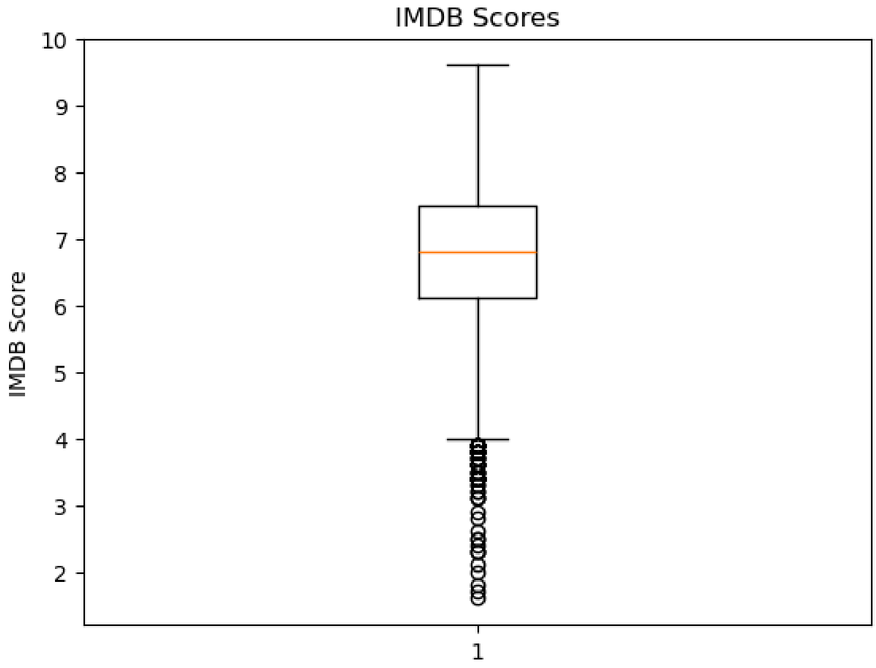
As the formula (y = -0.02X + 37.77) shows there is a negative linear relationship between changes in the dependent variable (y) and the independent variable (X). Y is the IMDb score and X is the number of votes.

The r^2 value is 0.01 which shows that for every unit increase in X (votes) Y (score) changes by 1%.

As decades increase the votes increase by a large margin but the scores change only slightly.

As a note, further analysis may change this conclusion. Such as why do the number of votes increase over time, analysis by country, societal issues (such as a war or economic boom or recession), etc.

The box plots below confirm outliers.



We also did a Z value analysis.

For the number of votes:

T-Statistic: 0.23199848692420444
P-Value: 0.8167568238544297

The T-statistic is close to 0 indicating that there is only a small difference between the mean of the two groups (the number of votes and IMDb scores).

The P-value of .8167 suggests that there is not enough information to reject the null hypothesis (the number of votes does not affect ratings).

For the release year.

```
T-Statistic: -0.3301184333795064
P-Value: 0.7416274901319813
```

The T-statistic is close to 0 indicating that there is only a small difference between the mean of the two groups (the release year and IMDb scores).

The P-value of .8167 suggests that there is not enough information to reject the null hypothesis (the release year does not affect ratings).