**Project 1**

As a fan of both movies and data, I’m fascinated by what makes a movie well-loved. While there's no perfect formula for a “good” movie, taking the average rating from a large audience provides a general sense of its quality. By analyzing those ratings, along with various factors about each movie, we can uncover trends that point to what makes a movie more likely to be well-received. These trends can then be used to assist movie watchers in choosing their next film, this time one that is more statistically probable to be a good one.

The data for this project was found on Kaggle, and included 45,466 rows and 24 columns. The data was sourced from The Movie DataBase, or TMDB. The columns were adult (whether the film was an adult film), belongs\_to\_collection (what franchise, if any, the movie is a part of), budget, genres, homepage (the website of the film), id, imdb\_id, original\_language, original\_title, overview (a brief summary of the film), popularity, poster\_path (a link to the poster of the movie), production\_companies, production\_countries, Release Date, revenue, runtime, spoken\_languages, status (whether the movie was yet released), tagline, title, video, vote\_average, and vote\_count. The target was vote\_average, the average user ratings for each movie.

While robust, the dataset was in need of a fair amount of data preprocessing and cleaning. The first step was to remove any rows with less than 500 user votes. This was done to prevent movies with only a handful of popular or unpopular ratings from biasing the model. After this step was taken 2056 rows remained. At this point columns without significant predicting power, columns primarily composed of nulls, and columns with practically no variation were removed. Specifically adult, original\_title, belongs\_to\_collection, homepage, id, imdb\_id, overview, poster\_path, status, tagline, video, original\_language, and title were removed. The next problem was that multiple columns’ output was difficult to read, or use. An example being this “genres” output “[{'id': 53, 'name': 'Thriller'}, {'id': 18, 'name': 'Drama'}]”. Therefore, a bit of code that looked for ‘name’: ‘ and outputted the following word, in a comma separated list if needed, was implimented. Therefore, the previous output would be rewritten to appear like “Thriller, Drama”, a much more usable output. This same process was used on genres, production\_companies, production\_countries, and spoken\_languages. The next step in preprocessing was to split “Release Date” into multiple columns, ones for release year, release month, release day, and release day of the week. Then, all non-numeric columns were one-hot-encoded so that each possible result was its own boolean value. For genres, production\_companies, production\_countries, and spoken\_languages each column was looped through and split at any commas, so that movies with multiple genres, for example, would be represented in both of their genre’s columns. Finally, any columns with a sum less than 50 were removed from the dataset, so that a production company or genre with only a few movies would not bias the data.

The first visualization that I created was actually before “Release Date” was split into multiple columns. My original plan was to create a line graph showing the average score for every year that was included in the dataset. However, I quickly realized that, because this dataset encompassed over a hundred years of data, that was very impractical. So, I instead binned the data by decade, creating a line graph that represented each decades’ average score. At this point I discovered an interesting phenomenon, as the graph showed that, with only two exceptions, each decade's films were markedly worse than the previous ones. This made me realize the sampling bias inherent in my data, which was that it is much more likely for a bad contemporary movie to have been seen more than five-hundred times by modern day reviewers than one from the early 1920s, for instance. So, while this graph did not give very useful results itself, it did help guide my future ones. The primary visualizations for this project all followed the same pattern as each other, which was to take a hot-one-encoded column and graph each instance’s average rating and count. This easily shows which factors tend to correlate with both positive and negative reception to movies, as well as note which columns may be affected by sampling bias.

The results of these visualizations were very interesting. According to the visualizations, the best genre of movies is war, with an average score of 7.14, closely followed by historical movies, which had an average score of 7.10. However, a narrow third was dramas, with a score of 6.97. This was especially interesting as drama was the most common genre, with over eight-hundred entries. Rather unsurprisingly, the lowest scoring genre was horror, with a score of 6.31. Fantasy, Comedy, and Action made up the rest of the bottom four. Personally, I found this graph rather unsurprising, as the genres that are more often considered serious and win awards tended to take the top spots. When looking at the production company graph Warner Bros. took the top spot with a score of 6.76, even while having the most production credits in this dataset. Relativity Media was the least popular production studio, with a score of only 6.22, meaning that the variance between production studios was considerably less than between genres. Yet even less variance was shown in the following graph, which was for the production country. It showed that British films were the best, with a score of 6.74, and Canada films the worst with a score of 6.32. American films landed in the middle of the graph, with a score of 6.58, though that was likely influenced by the vast difference in representation within the dataset, similar to what happened with the graphic decades earlier. The best language for a film was shown to be German, with a score of 6.87, while Chinese and Russian films in particular should be avoided, as those were the only two languages who scored worse than the bloated English column. Finally, Release Year, Release Month, Release Day, and Release Day of the Week were also graphed, though their results were not expected to be very conclusive. Nevertheless one interesting thing could be pulled from the Release Month graph. In it the final four months of the year held the top four scores, likely because that is when movies wishing to make an award season push are most often released.

With this information the question needed to be answered, what does the ideal movie look like? According to this data, the movie should be a historical drama within a war setting, produced by Warner Brothers, made by the British but with German spoken throughout the film, ideally released on Saturday, December 2nd, 2002. When I asked ChatGPT which movie most closely resembled this description it recommended the film *Enemy at the Gate*. This movie was a historical drama within a war setting, made by the British with German spoken throughout the film. Unfortunately it was not produced by Warner Brothers, and its release date was a bit off, being released on Friday the 16th, March of 2001. To test this recommendation I checked its TMDB user score, which was 7.4. When comparing this rating to the dataset Enemy at the Gate was found to be in the top 16% of rated movies, which I feel is a fairly successful recommendation.

The impact of this project is admittedly limited. What it can do is help assist in recommending movies by steering its users toward factors that are statistically more likely to be beneficial. That being said, it should be acknowledged that a movie is of course more than the sum of its parts, so no perfect movie recommendation system can ever be created. This model absolutely cannot definitively tell someone which movies they will or will not like, only which films are more or less likely to be liked by the average user. Additionally, other important factors such as the cast and crew of a film are important to a film’s success. Finally, the dataset which was used has a very pronounced bias toward American movies released in the 2000s. This means that any further modeling would likely struggle with films that do not fit into that mold.

**References**

https://www.kaggle.com/datasets/rounakbanik/the-movies-dataset

https://chatgpt.com/

Note: ChatGPT was used in this assignment for specific code segments. Specifically, to assist in creating the graphs, with hot-one-encoding, and with splitting the Release Date column.

**Link to Code**

https://github.com/bigbadraj/Project-1-Movie-Correlation