EXPLORATORY DATA ANALYSIS FOR MACHINE LEARNING

This project aims to bridge the gap between theoretical knowledge and practical application by leveraging the tools and techniques acquired throughout the course.

The initial phase describes the steps to prepare the chosen dataset. This will include data cleaning, preprocessing, and exploratory data analysis to ensure its suitability for subsequent analysis. Once the dataset is well-prepared, we will delve into hypothesis testing and later, maybe, machine learning techniques (either supervised or unsupervised) to extract valuable insights.

DATA SET

This section contains a brief description of the data set and a summary of its attributes. The data set chosen was the Rotten Tomatoes Movie Rating data set from Kaggle.

The dataset contains over 15,000 movies reviewed by Rotten Tomatoes. It includes detailed information about each movie, such as cast, plot summary, and critical reviews. Rotten Tomatoes is a review aggregation website that uses a Tomatometer score to determine whether a movie is "fresh" or "rotten." This dataset offers a valuable resource for analyzing movie trends, audience preferences, and the impact of critical reviews on box office success.

Attribute	Data Type	Description
`movie_title`	String	The title of the movie
`movie_info`	String	A summary of what the movie
`critics_consensus`	String	A summary of the critics thoughts about the movie
`rating`	String	The rating of the movie i.e. PG for Parental Guidance etc.
`genre`	String	A comma separated list of genres that apply to the movie
`directors`	String	The movie director or a comma separated list where more than one director is named
`writers`	String	The writer or a comma separated list where more than one writer is named
`cast`	String	A comma separated list of the cast members of the movie
`in_theaters_date`	Date	The date when the movie was first released in the theaters
`on_streaming_date`	Date	The date when the movie was first streamed
`runtime_in_minutes`	Integer	The runtime, in minutes, of the movie
`studio_name`	String	The name of the Studio that released the movie
`tomatometer_status`	String	The Tomato Meter Status, typically Certified Fresh or Fresh (good) or Rotten (bad)
`tomatometer_rating`	Integer	The Tomato Meter numeric rating for the movie
`tomatometer_count`	Integer	The count of the Tomato Meter rankings i.e. critics rankings
`audience_rating`	Integer	The audience rating for the movie
`audience_count`	Integer	The count of the Audience rankings

INITIAL DATA EXPLORATION AND CLEANING

Initial plan for data exploration and cleaning:

- Read in the data
- Validate data types
- Identify missing data
- Verify values / ranges

Read Data

I am using Polars for handling my data. With polars I could have chosen to describe a schema to use with the data but for this exercise I just loaded the data using `read_csv()`:

```
# Data Load
df = pl.read_csv("./data/rotten_tomatoes_movies.csv")
```

After reading the data using `df.shape` we can see that there are 17 attributes and 16638 rows of data.

Validate Data Types

Using `df.dtypes` we can validate the data type that were imported. Looking at the table below we can see that all data type are correctly imported expected for the two date fields. After fixing these data types and running `df_dtypes` again we can see that all data types are now correct.

```
df = df.with_columns(
    pl.col("in_theaters_date").str.to_date(),
    pl.col("on_streaming_date").str.to_date(),
)
```

Attribute	Data Type on Load	Data Type Corrected
movie_title	String	String
movie_info	String	String
critics_consensus	String	String
rating	String	String
genre	String	String
directors	String	String
writers	String	String
cast	String	String
in_theaters_date	String	Date
on_streaming_date	String	Date
runtime_in_minutes	Integer	Integer
studio_name	String	String
tomatometer_status	String	String
tomatometer_rating	Integer	Integer
tomatometer_count	Integer	Integer
audience_rating	Integer	Integer
audience_count	Integer	Integer

Identify Missing Data

The `df.null_count()` method gives us a report on the number of rows that contain nulls for each attribute. Data / rows that are identified as should be dropped will be removed during data cleaning.

Attribute	Missing Count	Comment
movie_title	0	

movie_info	24	Whilst the movie information would be useful it's probably not essential so missing values are not reason enough to drop rows with this attribute missing.
critics_consensus	8329	If text or sentiment analysis were to be use this text would be important but not reason enough to initially drop rows with this attribute missing.
rating	0	
genre	17	Genre is potentially important so drop rows with missing data
directors	114	Directors is also important so drop rows with missing data
writers	1349	Writer is less important so don't initially drop rows with missing data
cast	284	Cast is less important so don't initially drop rows with missing data. Further inspection would be required to determine if the cast are listed in importance.
in_theaters_date	815	If we want to make comparisons between theatre data and streaming date we need both so drop missing
on_streaming_date	2	If we want to make comparisons between theatre data and streaming date we need both so drop missing
runtime_in_minutes	155	Runtime is important so investigate of mean or median can be used
studio_name	416	Studio Name is less important so don't initially drop rows with missing data
tomatometer_status	0	
tomatometer_rating	0	
tomatometer_count	0	
audience_rating	252	Without audience rating the data in incomplete so drop
audience_count	252	Without audience count the data in incomplete

so drop

From table above it was decided to drop all rows where the data was missing in only the following columns:

- genre
- directors
- in_theaters_date
- on_streaming_date
- audience_rating
- audience_count

Removing rows where these columns contained a missing value reduced the number of rows to 15460.

Verify Data Values and Ranges

Looking at the ranges of the Date and Numeric data there are some causes for concern that should be investigated further:

statistic	in_theaters_date	on_streaming_date
count	15579	15579
null_count	0	0
mean	31/08/1999	21/04/2008
std		
min	01/06/1914	06/06/1935
25%	01/01/1993	29/10/2002
50%	16/06/2006	23/10/2007
75%	24/05/2013	03/12/2013
max	25/10/2019	01/11/2019

Two things jump out when we look at the in theatre and streaming data. A minimum streaming date of June 1935 appears unrealistic. Another point to investigate is the fact that the max streaming date is after the max in theatre date. Not in itself an issue but this should also be reviewed.

statistic	runtime_in_minutes	tomatometer_rating	tomatometer_count	audience_rating	audience_count
count	15460	15579	15579	15579	15579
null_count	119	0	0	0	0
mean	102.94	60.13	59.34	60.69	160063.31
std	25.23	28.56	67.47	20.40	1863878.94
min	1	0	5	0	5
25%	90	38	13	45	1011
50%	100	65	31	63	5425
75%	112	85	82	78	32264
max	2000	100	497	100	35797635

Looking at the numeric data there are a few things to look out for when examining the data further:

1. We need to come back and look at the 119 missing runtime data
2. The minimum runtime of 1 minute should be examined
3. The maximum runtime of 2000 minutes should also be examined
4. The maximum audience count of nearly 36 million should also be examined to determine if it is an outlier

FURTHER DATA EXPLORATION AND FEATURE ENGINEERING

Actions taken for data exploration and feature engineering.

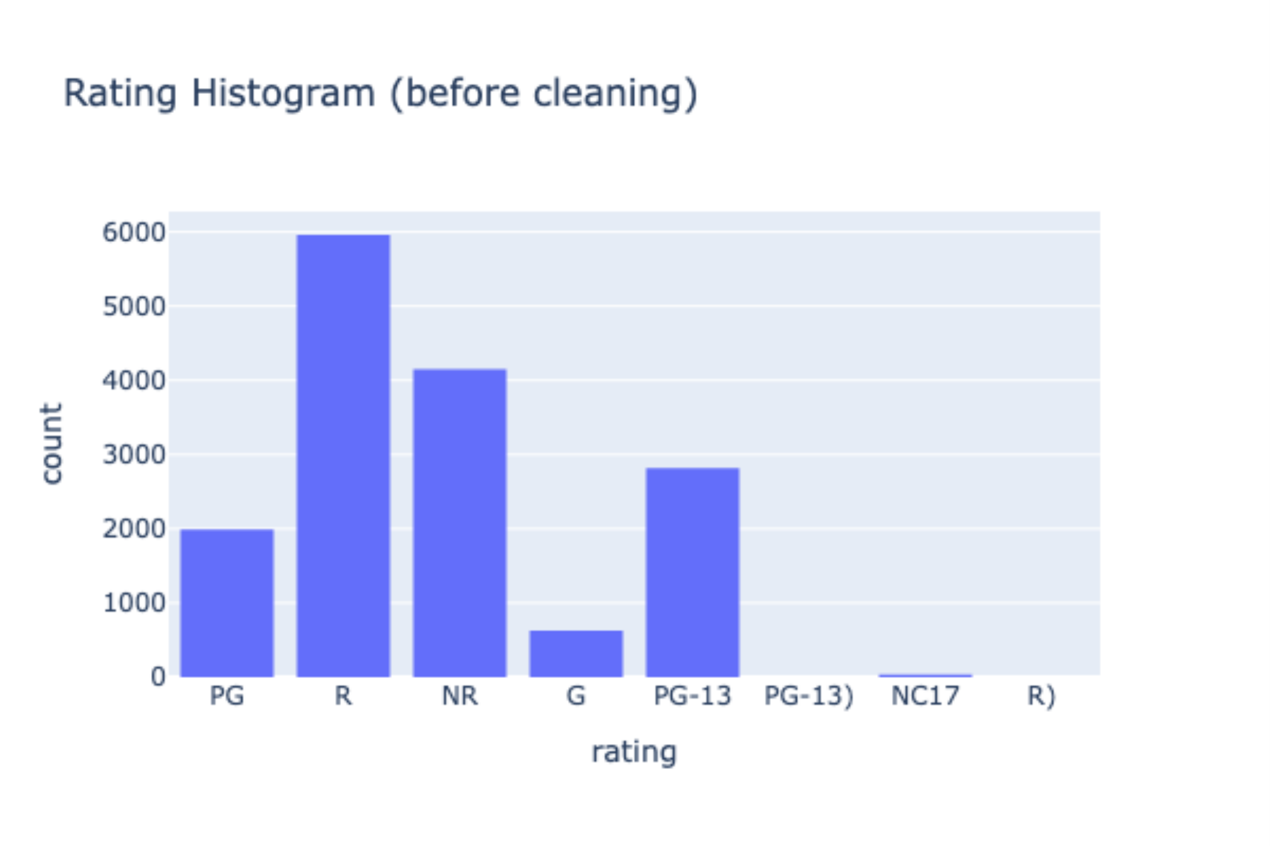
Categorical Data

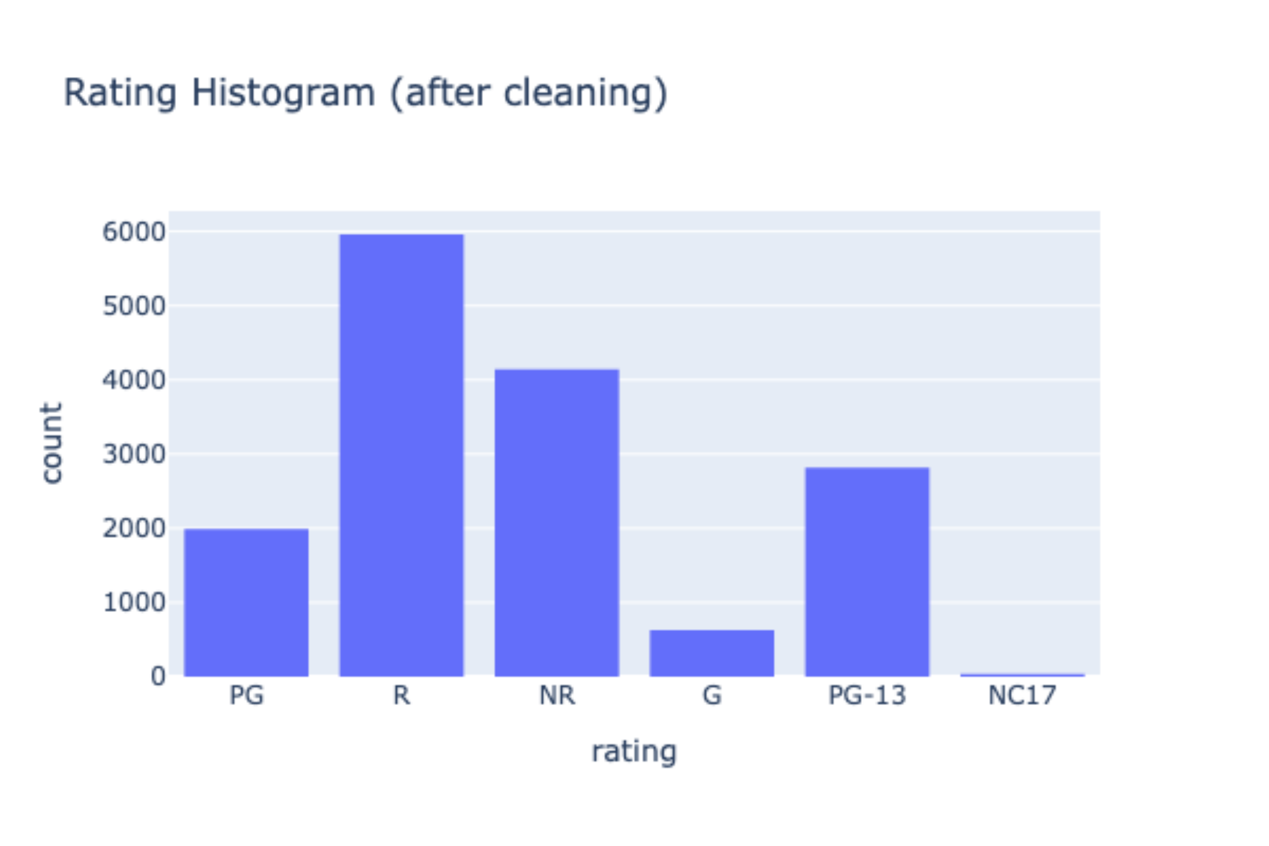
There are several categorical data type in the data. Genre could be considered categorical, however, as multiple genres are being combined to form compound genres e.g. Comedy and Romance, the categories become somewhat meaningless. At a later point, if genre were to be used for predictions, then one-hot encoding could be used to identify the unique genres associated with each movie. This is left to later as part of feature engineering. Other categorical data attributes to explore are:

- rating
- tomatometer_status

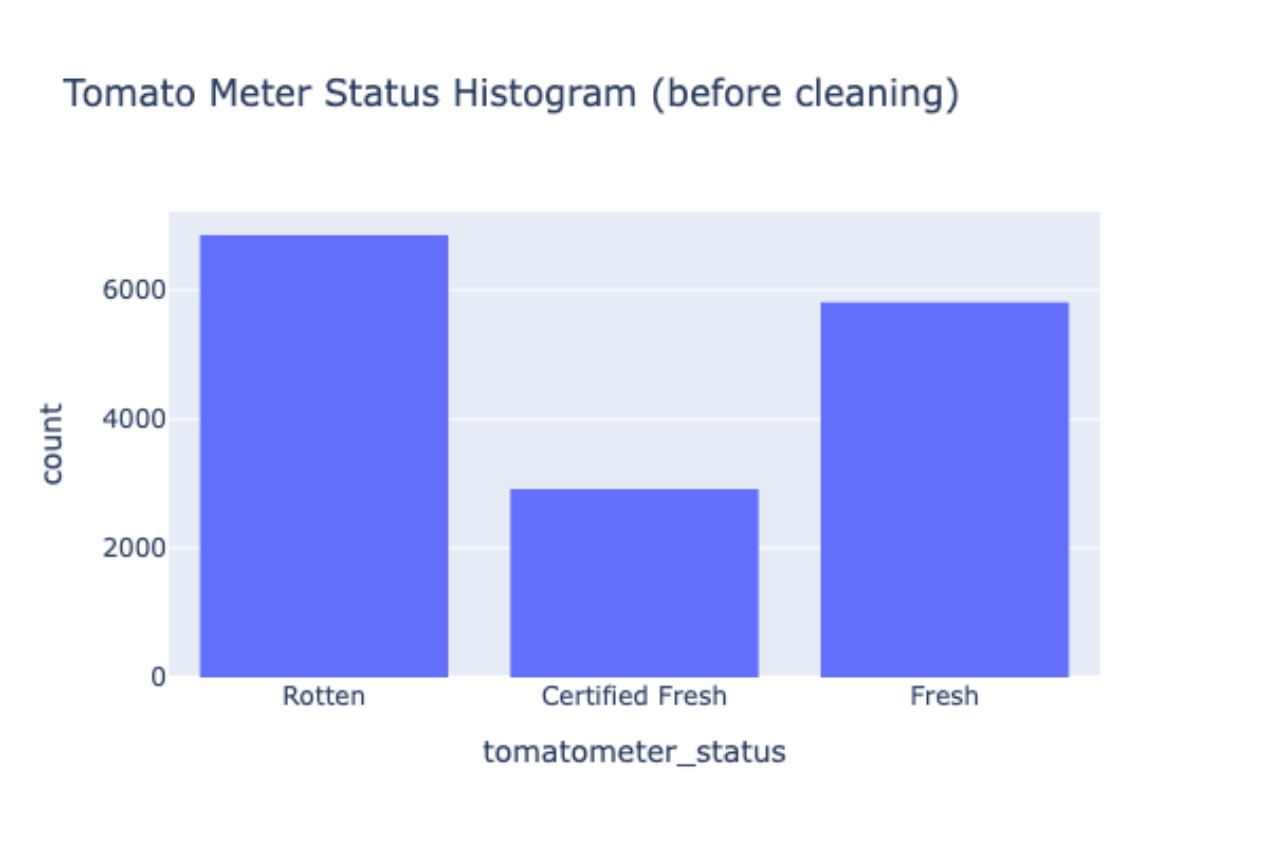
Rating

From the before histogram we can clearly see that there is probable data corruption i.e. `PG-13)` and `R)`. The after histogram shows the rating distribution after these issues are cleaned up.



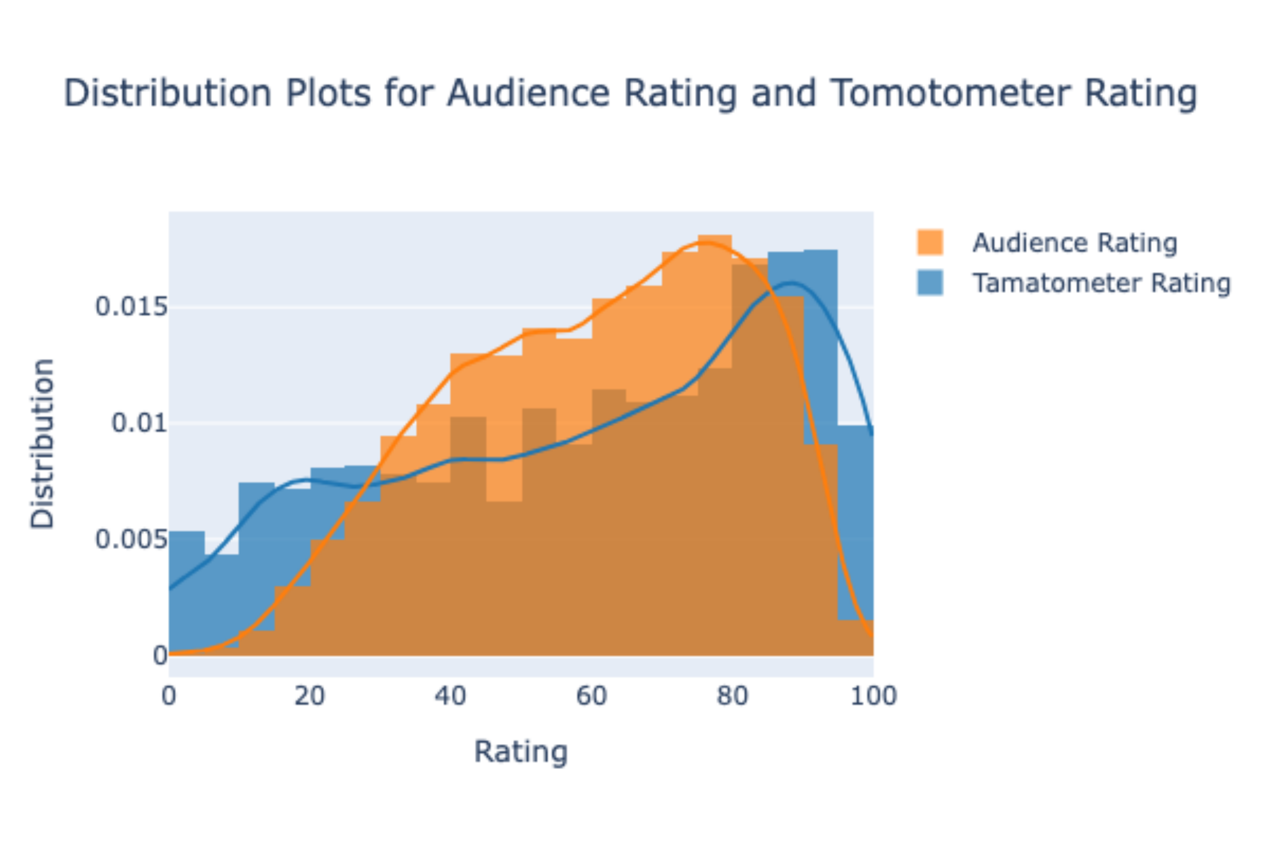


Tomato Meter Status



There are no data issues with the tomato meter status categorical attribute.

Distribution of Tomato Meter and Audience Ratings



As the plot shows, both our rating attributes deviates from the normal distribution. They both have a longer tail to the left, so we call it a negative skew. In statistics **skewness** is a measure of asymmetry of the distribution. Here, we can simply use the `skew()` function to calculate our skewness level of the Audience Rating and the Tomatometer Rating.

```
Tomato Meter Rating Skew: -0.4092542463397632
Audience Rating Skew: -0.31667314010118797
```

The range of skewness for a fairly symmetrical bell curve distribution is between -0.5 and 0.5; moderate skewness is -0.5 to -1.0 and 0.5 to 1.0; and highly skewed distribution is < -1.0 and > 1.0. In our case, we have -0.41 and -0.32, so our data could be considered only slightly skewed data.

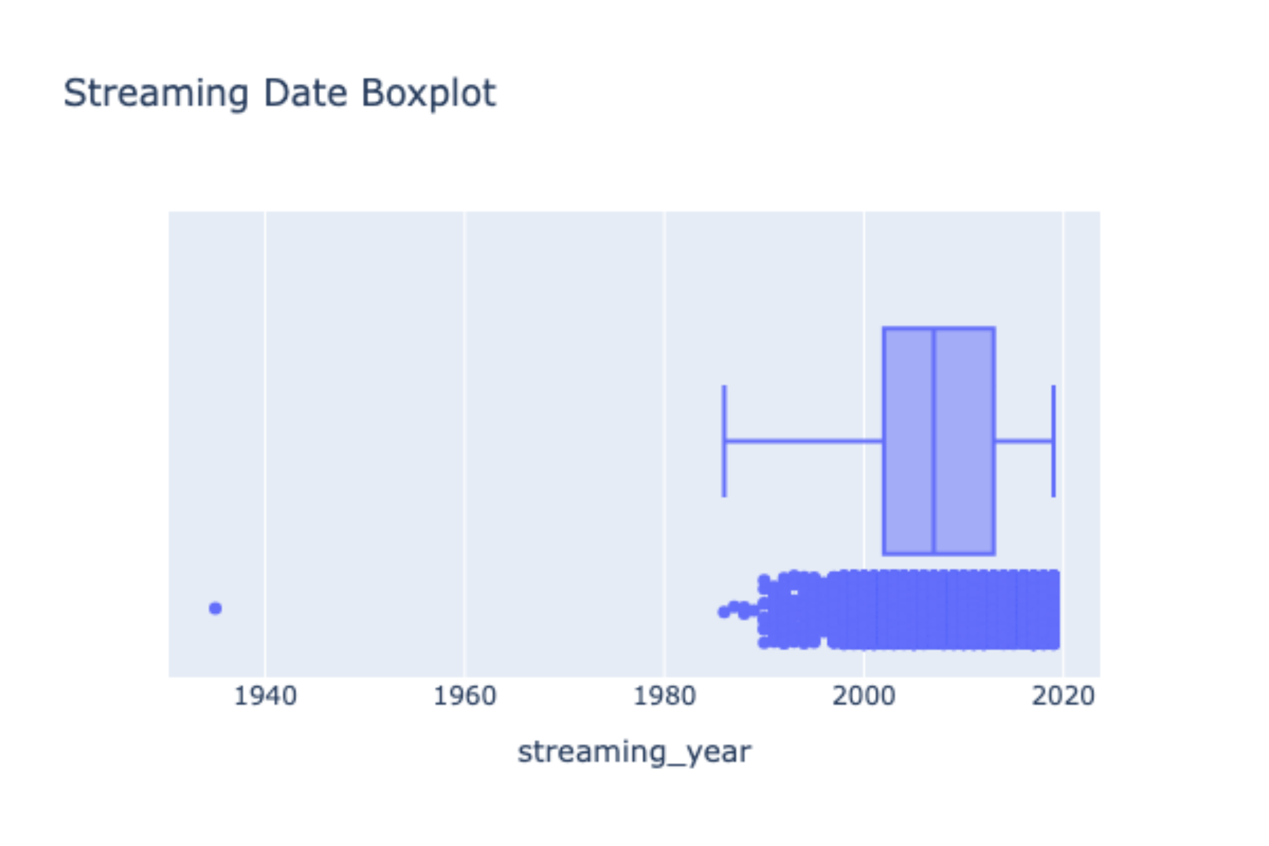
Now, we could try to transform our data using the `log()` function, so it looks more normally distributed however, in this scenario, this is not warranted especially as there are zero value rating which we cannot take the log of.

Further Analysis of Theatre and Streaming Date

Earlier we identified that there was a streaming date from 1935 and also at least one instance where the streaming date was before the theatre date. We'll now explore this further. To facilitate this analysis the following 4 new attributes / features were added:

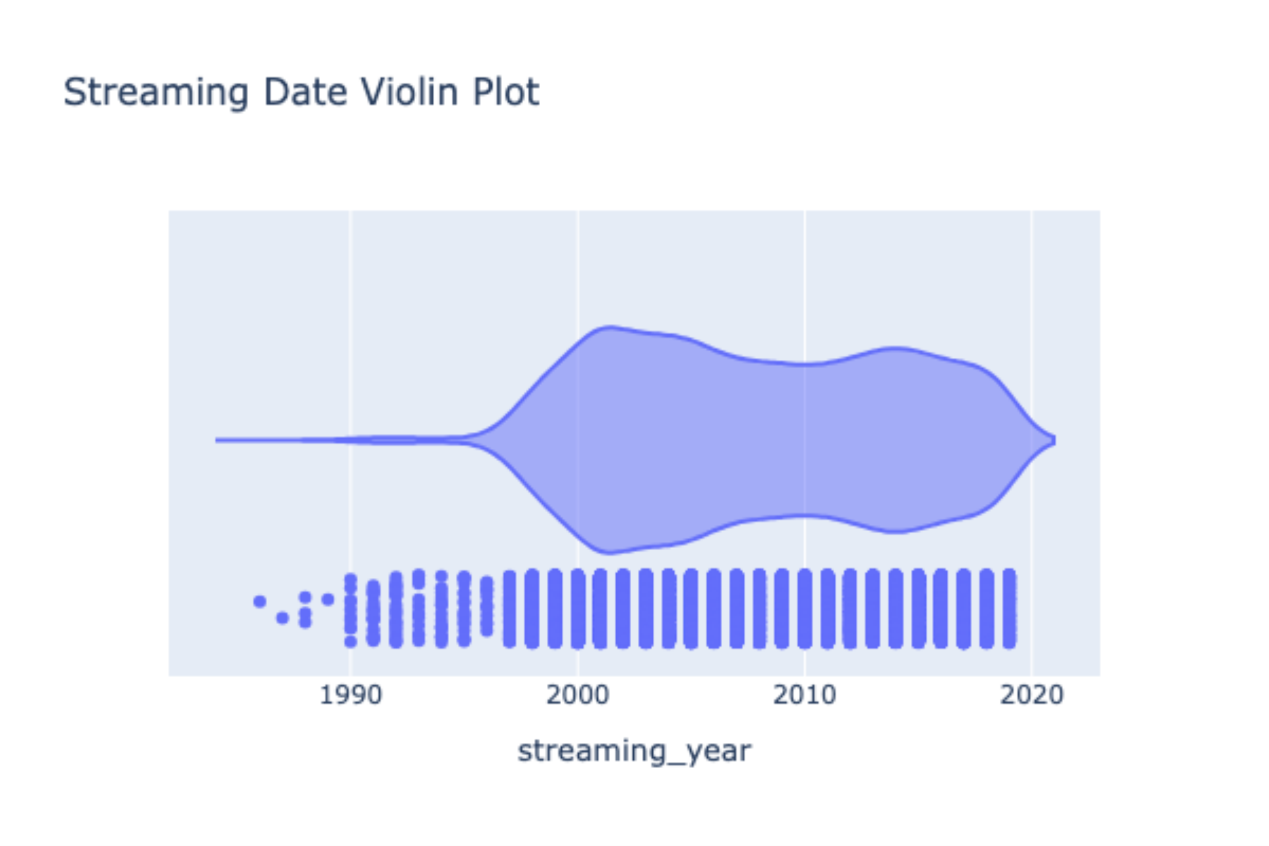
- theatre_month
- theatre_year
- streaming_month
- streaming_year

We'll start with a Box-plot of the streaming data:



From this Box-plot we can clearly see that the 1935 streaming date is an outlier and obviously wrong. We can also see that 1990 is the first year where there are a significant number of movies streamed. Let's examine all movies with a streaming date before 1990.

In total there were only 7 movies streamed before 1990 but, apart from "The 39 Steps" which, according to the data, was in theatres August 1935 and stream in June 1935. This is obviously incorrect. From the Rotten Tomatoes own website we can see that the actual movie release date to theatres was June 6th, 1935 and the first stream date was January12th, 2017. As we are correcting the streaming date we will also correct the first in theatres date.



Once we have cleaned up the data for "The 39 Steps" movie we get a much cleaner box-plot. If we change the bon-plot to a violin plot as shown above we can see a surge in the early 2000's as older movies are released online. Then as we move towards 2020 we see a decline in the number of movies streamed. The cause of this is unknown but would potentially be an interesting area of investigation.

Next we will look at the films that were streamed before they were in theatres. This is not an obvious issue but it is good to understand the potential size of the problem if it is an issue.

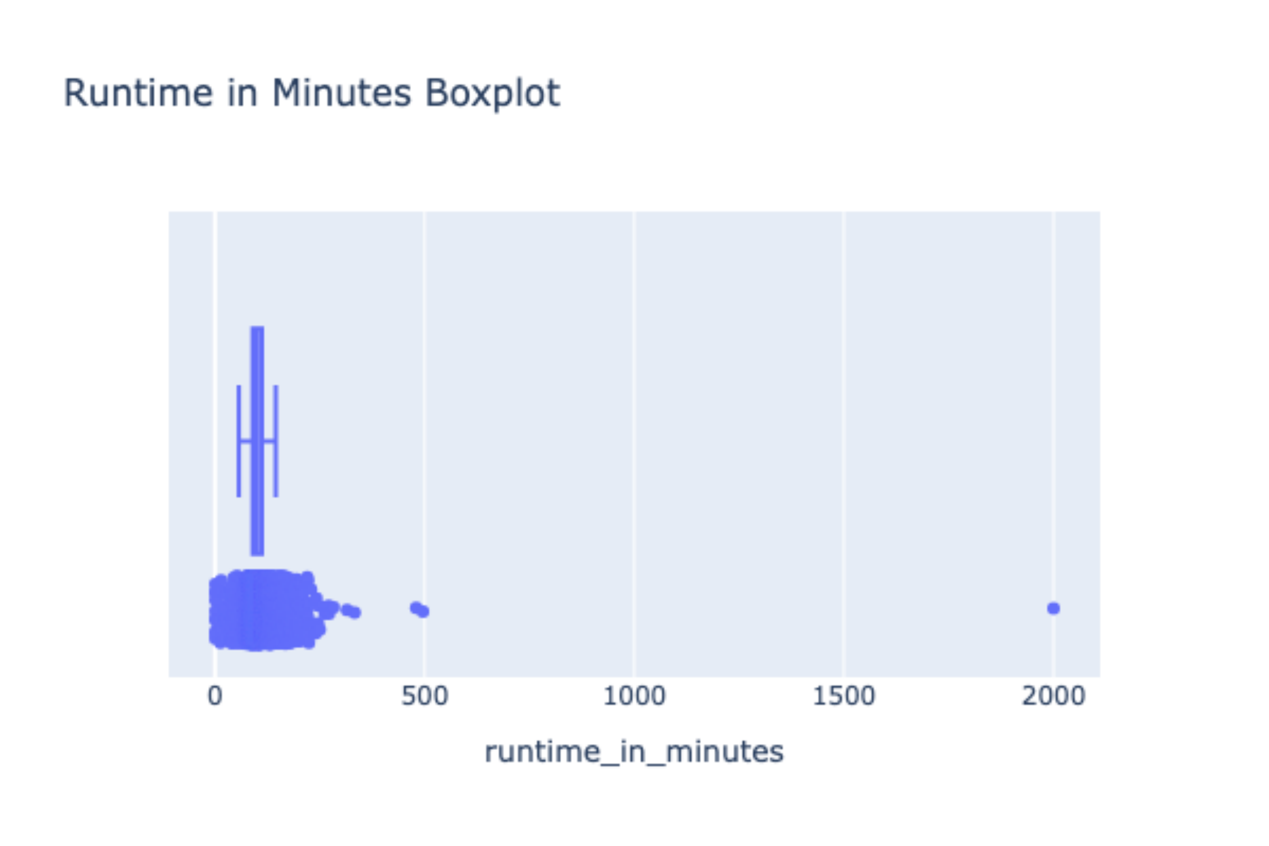
In total there are only 165 movies in the data set where the streaming date is before the theatre date. Closer examination of these movies by comparing them to the actual entries in the Rotten Tomatoes website suggests that the data set appears to be somewhat unreliable and does not accurately reflect the website data. Given this discovery there are two courses of action:

1. "Accept" the data in good faith and continue to use it
2. "Reject" the entire data set and start again.

Given that this is a learning exercise we will continue to use the data even though, at this point, we know that it is somewhat tainted.

Movie Runtime

Now we come back to the movie runtime. From the box-plot below we can see that there are 3 obvious potential outliers:

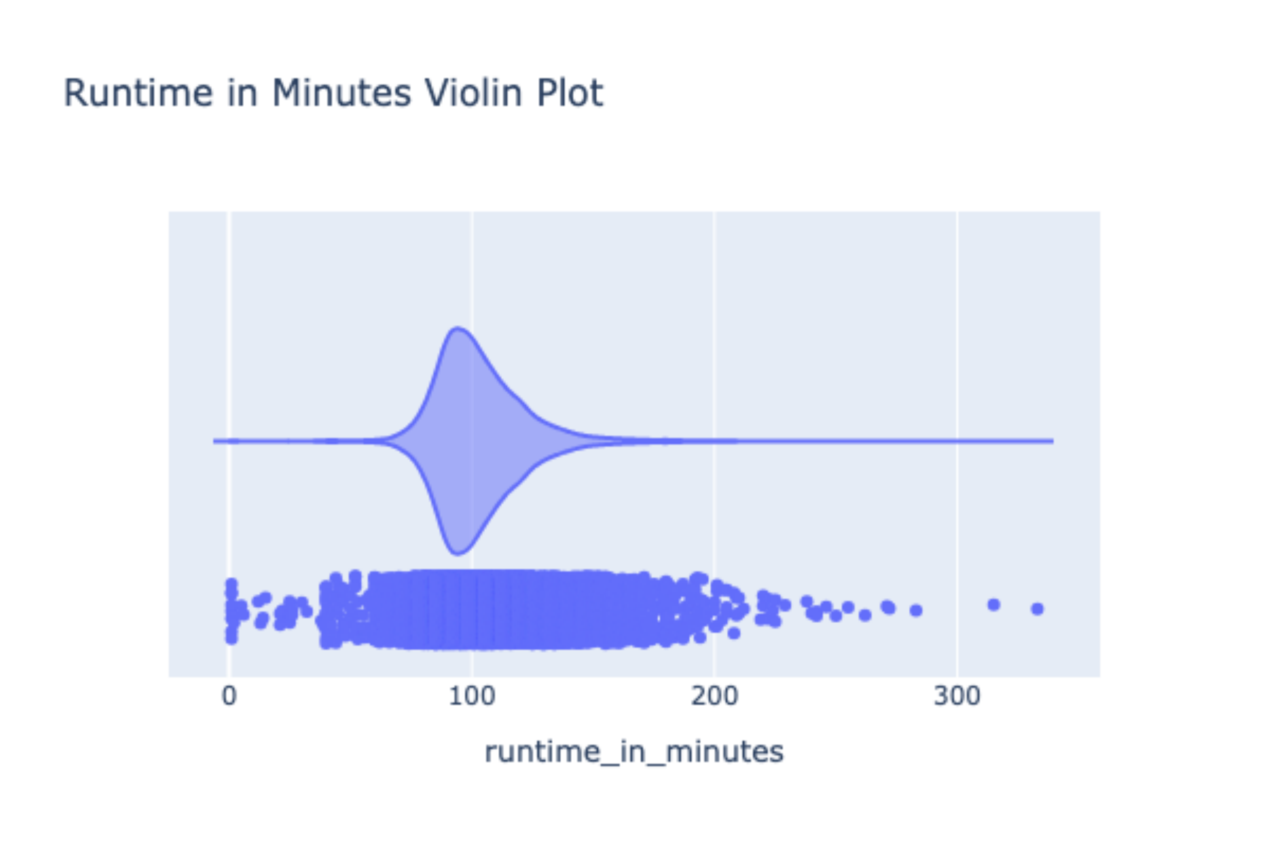


There are 3 movies with a runtime of greater than 420 minutes (7 hours). They are:

- Love on the Run

    - Suggested runtime: 496 minutes
    - Actual runtime: 80 minutes
- Never Sleep Again: The Elm Street Legacy
    - Suggested runtime: 480 minutes
    - Actual runtime: not available
- Terror Tract
    - Suggested runtime: 2000 minutes
    - Actual runtime: 93 minutes

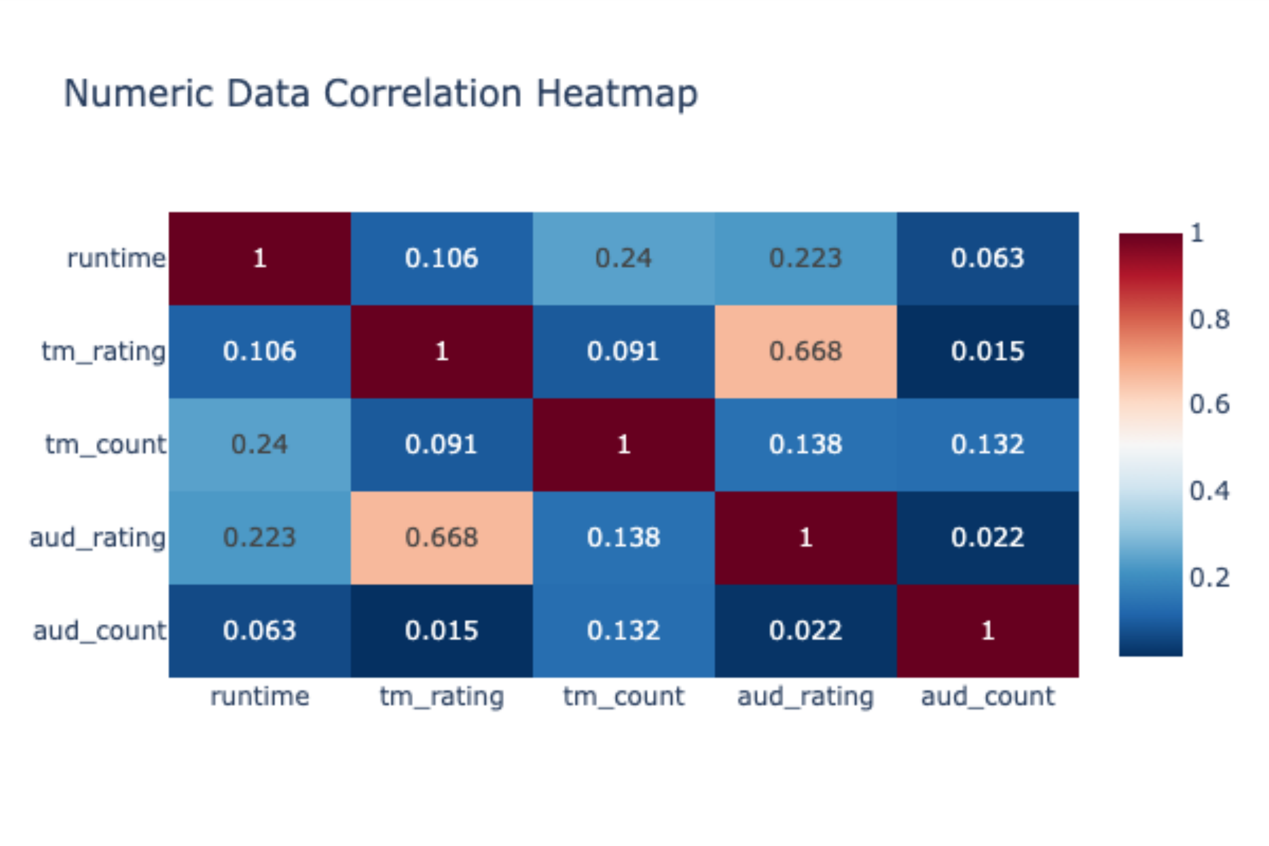
From the box-plot it is also clear that there are a number of movies with zero minutes runtime. These are probably incorrect but we will leave them for now. If we "fix" the obvious 3 outliers and then also fill missing values with the median value the following more reasonable violin plot is given.



As we can see most of the runtimes cluster around the 90 minutes mark, which makes sense, but is it not unreasonable for a movie to run to more than 3-4 hours. There is probably still a degree of inaccuracy in the data set when it comes to the actual runtime versus the value in the data set.

Correlation

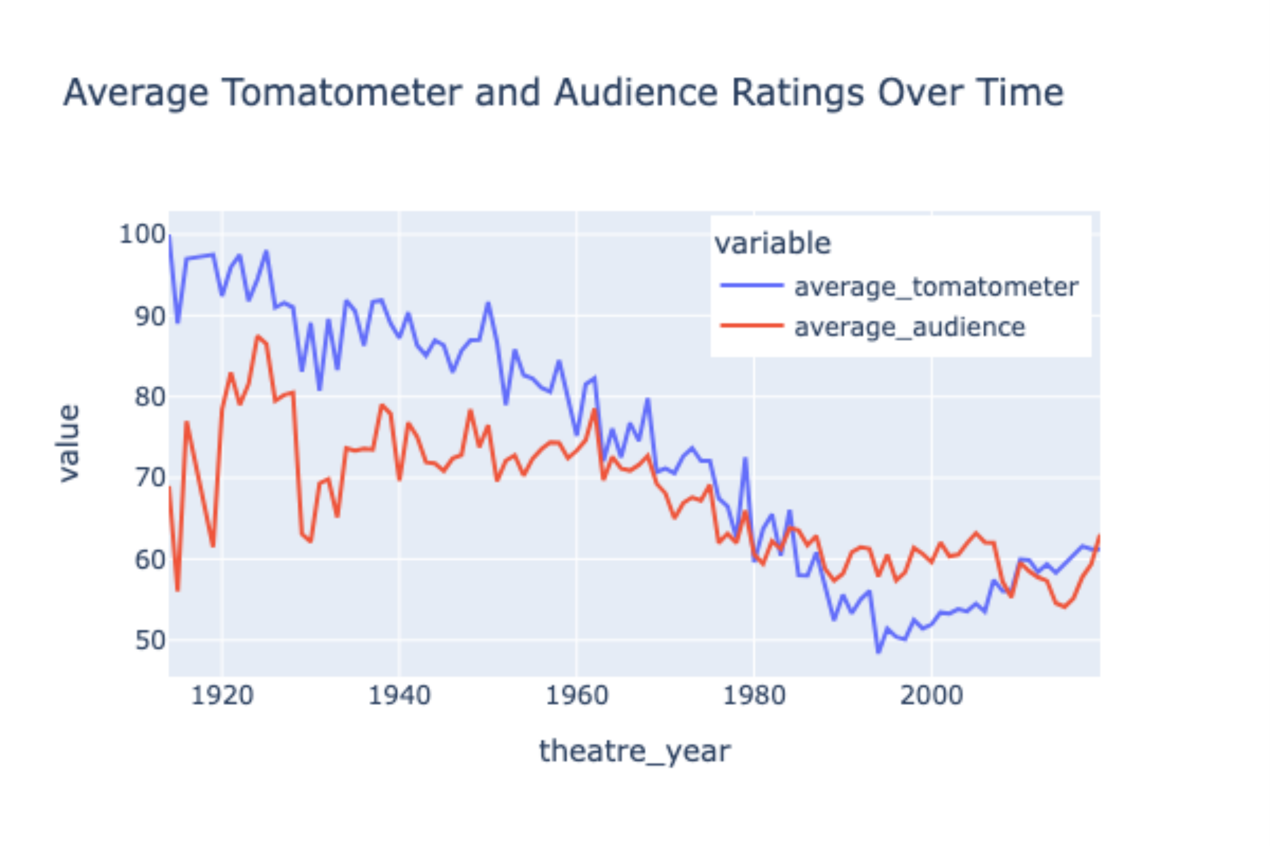
We will complete this in-depth EDA with a look at correlation across the attributes and finished with some pair plots to visualise. First we will have a general look at the correlation between all the numeric data.



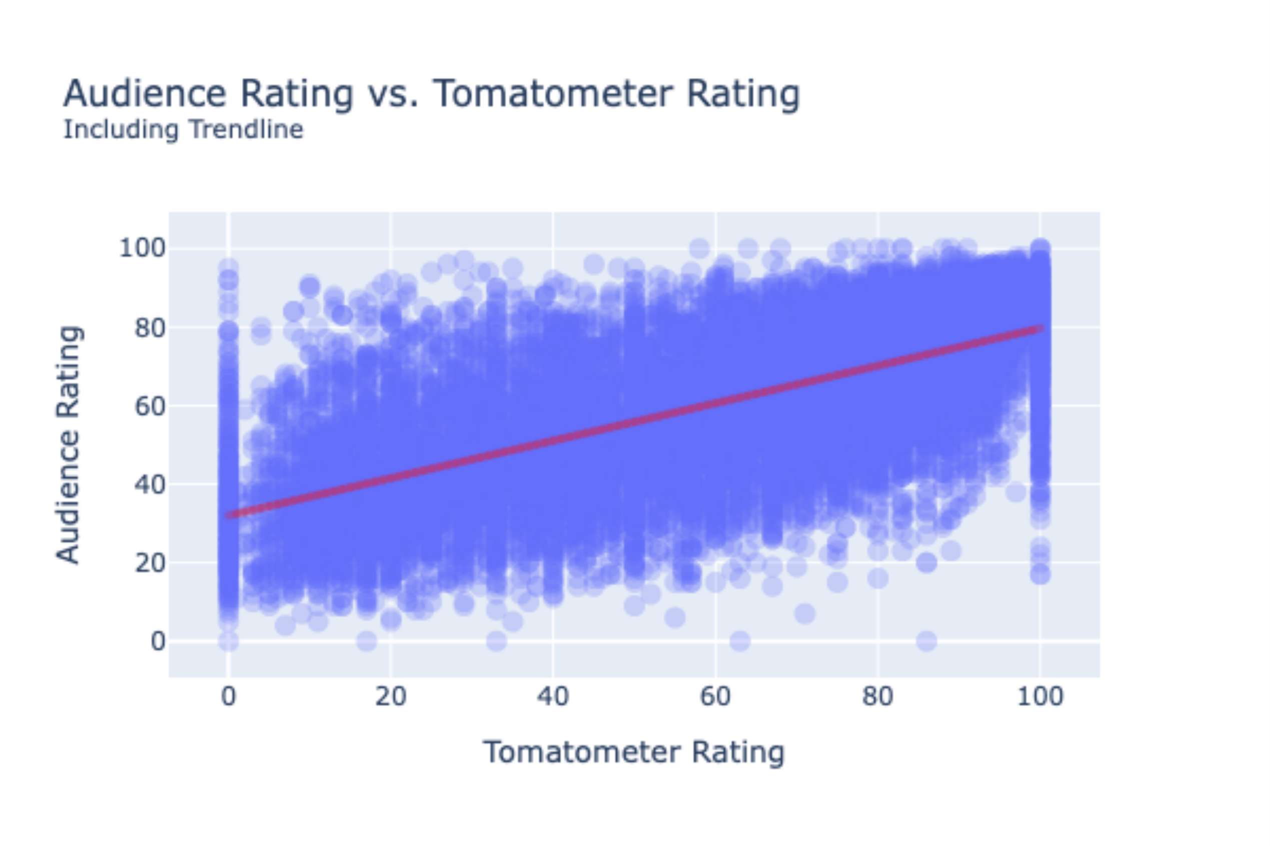
Numeric Data Correlation Heatmap

	runtime	tm_rating	tm_count	aud_rating	aud_count
runtime	1	0.106	0.24	0.223	0.063
tm_rating	0.106	1	0.091	0.668	0.015
tm_count	0.24	0.091	1	0.138	0.132
aud_rating	0.223	0.668	0.138	1	0.022
aud_count	0.063	0.015	0.132	0.022	1

We can see that there is a correlation between the Tomato Meter Rating and the Audience rating. By plotting a trend line of the average ratings each year we can see this.



We can investigate further by viewing a scatter plot of the Tomato Meter rating versus the Audience rating:



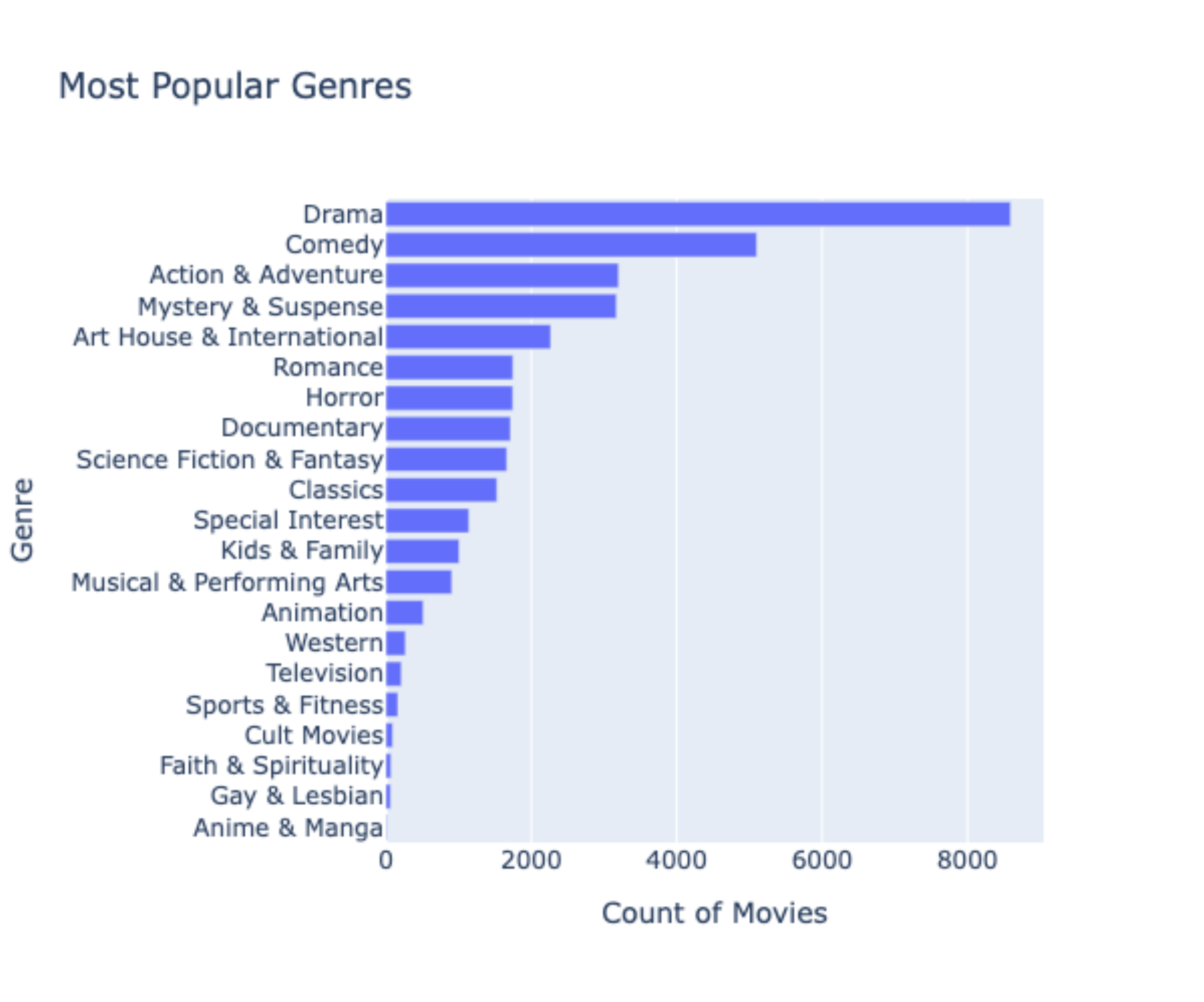
By including an "ordinary least squares" trend line we can see that there is a degree of correlation. Using the `corr()` function we see that the Pearson correlation coefficients between the two columns is `0.67`.

Feature Engineering

There are four columns of comma separated data:

- genre
- directors
- writer
- cast

Each of these attributes can be one shot encoded for later use in machine learning for both classification and regression machine learning models. Let's take a look at the genre attribute as an example. As mentioned previously the genre data for each movie is a comma separated list of genres. When we process this data we end up with a total of 21 genres. From the bar chart below we can see that Drama is by far the most frequently occurring genre followed by Comedy.



Performing the same analysis on the other similar attributes we see:

- The are a total of 8458 unique directors and the maximum number of movies any one director directed was 37 but with a median value of 1 and an average of just over 2 most directors have only directed in one or two movies.
- The are more writers than directors with 13948 unique writers but one writer is credited on 1099 movies. Is this realistic but with a median value of 1 and an average of just less than 2 most writers have only credited in one or two movies.
- Finally there were 194686 unique cast members with one actor appearing in 199 movies but with a median value of 1 and an average of 2 most actors have only appeared in one or two movies

KEY FINDINGS AND INSIGHTS

Key Findings and Insights, which synthesises the results of Exploratory Data Analysis in an insightful and actionable manner.

From the analysis above it can be seen that there are some issues with the quality of the data. During close inspection of outliers and suspicious data values we saw that the data was not actually representative of the true Rotten Tomato data on the website. That said the following key findings and insights can be shared:

1. That there does appear to be some correlation between the Tomato Meter Score and the Audience Score. These may be good candidates for a regression prediction machine learning exercise
2. The Genre attribute is potentially a good candidate for a classification exercise. It may also be a good contributor / predictor of the rating for the movie
3. With median / mean values of 1 and 2 for the directors, writers and cast members of the movies it is not immediately obvious that these attributes would be useful in either a regression of classification but by applying Principle Component Analysis (PCA) or by using SVM's on this sparse data may prove helpful.

HYPOTHESIS SETTING

Formulating at least 3 hypothesis about this data.

- First hypothesis
  - H_0 = There is no difference between the Tomato Meter Score and the Audience Score.
  - H_1 = There is a difference between the Tomato Meter Score and the Audience Score.
- Second hypothesis

    - H_0 = The Year the movie was first screened makes no difference between the Tomato Meter Score .
    - H_1 = The Year the movie was first screened makes a difference to the Tomato Meter Score.
- Third hypothesis
    - H_0 = The Genre of the movie makes no difference between the Tomato Meter Score .
    - H_1 = The Genre of the movie makes a difference to the Tomato Meter Score.

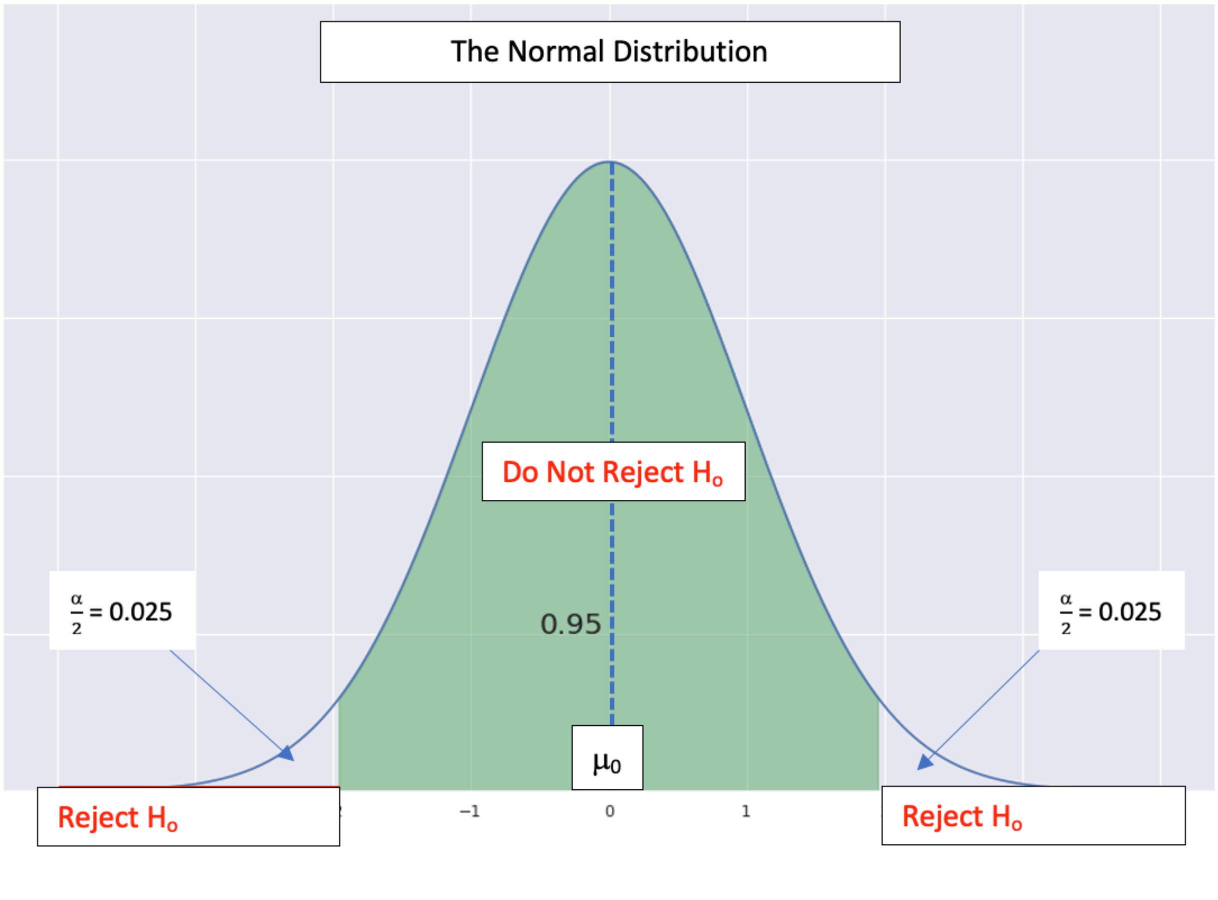
SIGNIFICANCE TEST

Conducting a formal significance test for one of the hypotheses and discuss the results.

In this first example, we will show how to prove (or disprove), with statistical evidence, that there is no difference between the Tomato Meter Score and the Audience Score.

1. Choose a sample statistic
    1. The first step in hypothesis testing is to choose a sample test statistic. Hypothesis testing allows us to check the sample statistic against a statistic of another sample or population. Let $\mu 1$ be the population mean for the Tomato Meter Score and $\mu 2$ be the the population mean for the Audience Score. We will compare these mean values, $\mu 1$ and $\mu 2$, statistically.
2. Define hypothesis (Null and Alternative)

    1. The next step is to define the hypothesis to be tested. Hypothesis is defined in two ways - null hypothesis and alternative hypothesis. Null hypothesis is a statistical hypothesis which assumes that the difference in observations is due to a random factor. It is denoted by H_0. Alternative hypothesis is the opposite of null hypothesis. It assumes that the difference in observations is the result of a real effect. The alternate hypothesis is denoted by H_1.
    2. H_0 : μ1 - μ2 = 0 i.e. there is no difference between the Tomato Meter Score and the Audience Score
    3. H_1 :μ1 - μ2 $\neq$ 0 i.e. there is a difference between the Tomato Meter Score and the Audience Score
3. The equal sign in the null hypothesis indicates that it is a 2-tailed test.
4. Set the decision criteria
    1. To set the criteria for a decision, we state the level of significance for a test. It could be 5%, 1% or 0.5%. Based on the level of significance, we can make a decision whether to accept the null hypothesis and reject the alternate, and vise versa.
    2. The diagram below describes the principles of hypothesis testing. We will choose 5% significance level. Therefore, our $\alpha = 0.05$. Since we have a 2-tailed test, we have to divide alpha by 2, which gives us 0.025. So, if the calculated p-value is less than alpha, we will reject the null hypothesis.



3. In this exercise, we will use one of the t-test, z-score, f-score or chi-squared statistics to evaluate our results.

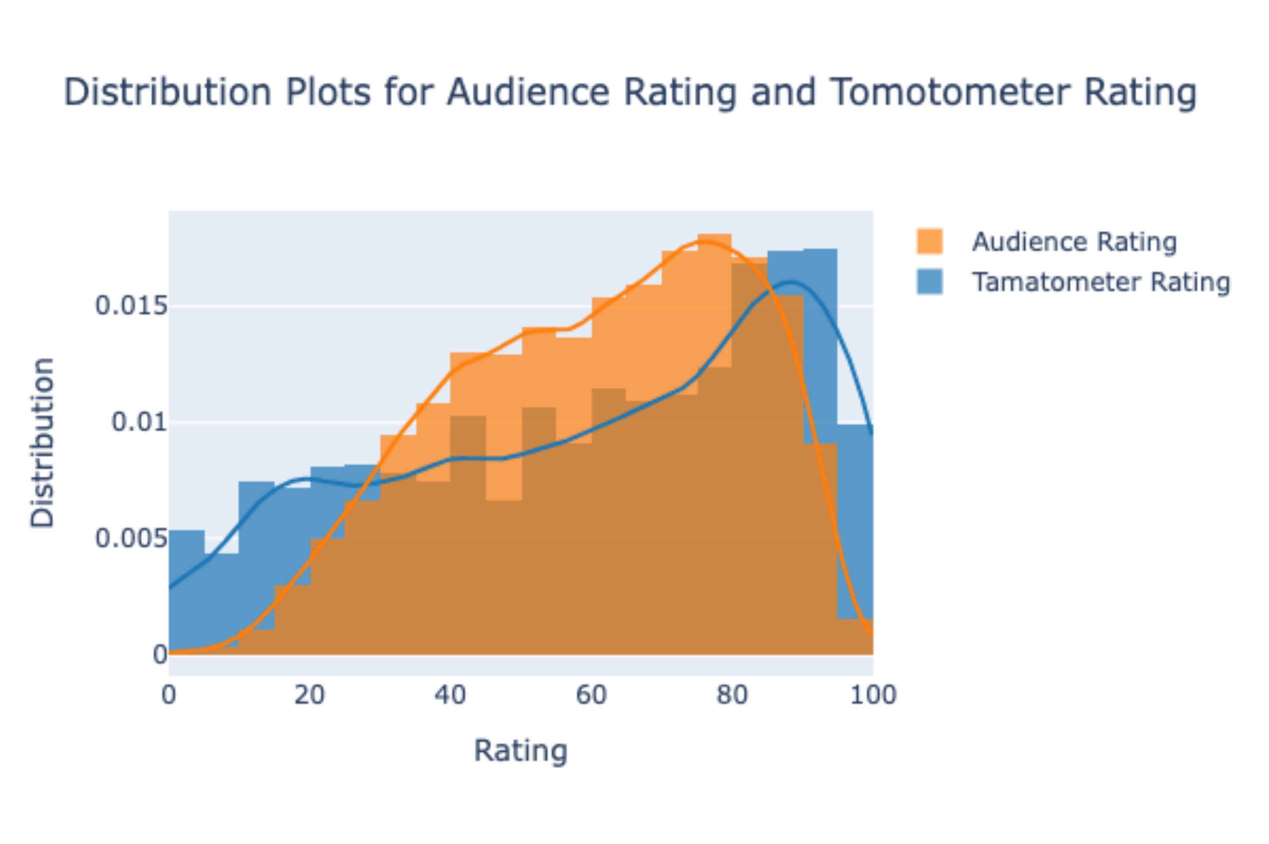
    1. A t-test is used for testing the mean of one population against a standard or comparing the means of two populations if you do not know standard deviation of the the population and when you have a limited sample ($n < 30$). If you know the standard deviation of the populations , you may use a z-test.

    2. A z-test is used for testing the mean of a population versus a standard, or comparing the means of two populations, with large ($n \geq 30$) samples, whether you know the population standard deviation or not. It is also used for testing the proportion of some characteristic versus a standard proportion, or comparing the proportions of two populations.

    3. An f-test is used to compare variances between 2 populations. The samples can be any size. It is the basis of ANOVA.

4. chi-squared test is used to determine whether there is a statistically significant difference between the expected and the observed frequencies in one or more categories of a contingency table. A contingency table is a tabular representation of categorical data. It shows the frequency distribution of the variables.

5. Plot the distribution of 'rating' values for Tomato Meter and Audience. We did this earlier:



   1. Although the two distributions are similar we can see that the Tomato Meter Ratings have a higher occurrence of lower / higher ratings

6. Now let's compare the mean values:

   1. Tomato Meter Rating mean: 60.13

   2. Audience Rating mean: 60.69

7. Next, we will obtain our statistics, t-value and p-value. We will use `scipy.stats` library and `ttest_ind()` function to calculate these parameters.

```python
# Set Alpha
alpha = 0.05

# Calculate and Print
t_value1, p_value1 = stats.ttest_ind(
  df["tomatometer_rating"],
  df["audience_rating"]
)
print(f"t_value1: {t_value1}\np_value1: {p_value1}")
```

2. t_value1: -1.9728816665214437

3. p_value1: 0.04851787178864086

Conclusion: since p_value 0.04851787178864086 is less than alpha 0.05 **Reject** the null hypothesis there is no difference between the Tomato Meter Score and the Audience Score.