Mini-Project 4: Cleaning and Analyzing Movie Data

Brady Sherry

November 20, 2024

Abstract

With the mass amount of data available nowadays along with the various software that has been designed to work with it, data science/analysis has been a rapidly growing field. By cleaning, organizing, and visualizing data, we can uncover quantitative or categorical patterns that provide insights and answer key questions. In the programming language Python, for example, there are several libraries and functions tailored specifically for these purposes, with the "pandas" library being the most widespread. Given a set of data, pandas can be used to filter, sort, rename, aggregate, and merge this data to its desired form, and then additional libraries can be utilized to output statistics or visualizations of certain columns/rows.

1 Introduction

In this project, we will be working with movie data, particularly from "Rotten Tomatoes" and "iMDb", two of the most popular websites for movie and TV show ratings, reviews, and information. Thus, we have two different datasets, which are previewed in Figure 1 below.

	Title	Certificate	Director	Runtime	Studio	RT Critics Score	RT Critic Reviews	RT Audience Score	RT Audience Reviews
0	The Angry Birds Movie 2	PG (for rude humor and action)	Thurop Van Orman	100 minutes	Columbia Pictures	73%	107	84%	4,023
1	Legend Of The Demon Cat (Kûkai)	NR	Kaige Chen	129 minutes	Well Go USA	91%	11	37%	74
2	Dora and the Lost City of Gold	PG (for action and some impolite humor)	James Bobin	102 minutes	Paramount Pictures	84%	148	88%	6,715
3	Luce	R (for language throughout, sexual content, n...	Julius Onah	109 minutes	NEON	91%	151	77%	284
4	Good Boys	R (for strong crude sexual content, drug and ...	Gene Stupnitsky	95 minutes	Universal Pictures	80%	237	86%	13,007

	Poster	Title	Year	Certificate	Runtime	Genre	iMDb Critics Score	iMDb Metascore	Director	Cast	iMDb Audience Reviews	Description	iMDb Critic Reviews	Review Title	Review
0	https://m.media-amazon.com/images/M/MV5BYWRkZj...	The Idea of You	2023.0	R	115.0	Comedy, Drama, Romance	6.4	67.0	Michael Showalter	Anne Hathaway, Nicholas Galitzine, Ella Rubin,...	28,744	Solène, a 40-year-old single mom, begins an un...	166	Hypocrisy as an idea	This film, as well as the reaction to it, is a...
1	https://m.media-amazon.com/images/M/MV5BZGI4NT...	Kingdom of the Planet of the Apes	2023.0	PG-13	145.0	Action, Adventure, Sci-Fi	7.3	66.0	Wes Ball	Owen Teague, Freya Allan, Kevin Durand, Peter ...	22,248	Many years after the reign of Caesar, a young ...	183	A phenomenal start to another trilogy!	I'm a big fan of all the planet of the apes, a...
2	https://m.media-amazon.com/images/M/MV5BZjIyOT...	Unfrosted	2023.0	PG-13	97.0	Biography, Comedy, History	5.5	42.0	Jerry Seinfeld	Isaac Bae, Jerry Seinfeld, Chris Rickett, Rach...	18,401	In 1963 Michigan, business rivals Kellogg's an...	333	not funny	Pretty much the worst criticism you can lay on...

Figure 1: Previews of the Rotten Tomatoes and iMDb Datasets

Note that each of these datasets contain audience and critic review scores, which is something that we will be able to compare later. Also included are details of the movie such as its title, certificate, director, runtime, studio, poster, year, genre, and cast.

2 Problem Statement

The goal with this data is to find any notable relationships or differences between the two websites, and to see what kinds of movies that audiences and critics enjoy. To accomplish this, we must first "clean" the data to get it into a format that we can easily analyze. We will need to remove unnecessary columns, transform data entries into certain formats, and then merge the datasets together before we begin this analysis.

3 Methodology

We will begin with the Rotten Tomatoes data, where we notice multiple issues: The Certificate column has unnecessary justifications in parentheses, the Runtime column should contain numerical values but has the word "minutes" in every entry, the Director column has multiple names included for some entries, the Studio column needs to be converted to the "string" data type, and the columns for Critic/Audience Score and Reviews need to be converted to the "float" data type. Figure 2 shows the code used to perform these changes.. Then, for the iMDb dataset, we find that there are columns that need to be dropped, the Genre column has multiple genres listed for some entries, the Classification column entries are inconsistent with the US system, the columns for Audience/Critic Reviews need to be converted to "float" data types, and the Critics Score column needs to be normalized to the same scale as the other scores. Figure 3 shows the code used to perform these changes.

```
#Part A: remove parantheses from the Certificate column
tomatoes['Certificate'] = tomatoes['Certificate'].str.replace(r'\(.*\)', '', regex=True)

#Part B: remove "minutes" from the Runtime column, and change type to float
tomatoes['Runtime'] = tomatoes['Runtime'].str.replace(r'minutes', '', regex=True)
tomatoes['Runtime'] = tomatoes['Runtime'].astype(float)

#Part C: remove all but one director from the Director column
tomatoes['Director'] = tomatoes['Director'].str.split(',').str[0]

#Part D: convert Studio column to string
tomatoes['Studio'] = tomatoes['Studio'].astype(str)

#Part E: convert the last 4 columns to float types

#RT Critics Score
tomatoes['RT Critics Score'] = tomatoes['RT Critics Score'].str.replace(r'\%', '', regex=True)
tomatoes['RT Critics Score'] = tomatoes['RT Critics Score'].astype(float)

#RT Critic Reviews
tomatoes['RT Critic Reviews'] = tomatoes['RT Critic Reviews'].astype(float)

#RT Audience Score
tomatoes['RT Audience Score'] = tomatoes['RT Audience Score'].str.replace(r'\%', '', regex=True)
tomatoes['RT Audience Score'] = tomatoes['RT Audience Score'].astype(float)

#RT Audience Reviews
tomatoes['RT Audience Reviews'] = tomatoes['RT Audience Reviews'].str.replace(r'\,', '', regex=True)
tomatoes['RT Audience Reviews'] = tomatoes['RT Audience Reviews'].astype(float)
```

Figure 2: Code Used to Clean the Rotten Tomatoes Dataset

```python
#Part A: delete the Poster, Cast, Description, Review Title, and Review columns
imdb.drop(['Poster', 'Cast', 'Description', 'Review Title', 'Review'], axis=1, inplace=True)

#Part B: only keep the first classification in the Genre column
imdb['Genre'] = imdb['Genre'].str.split(',').str[0]

#Part C: convert India ratings to US ratings in the Certificate column
conversion = {'A': 'R', 'U': 'G', 'All': 'G', 'Approved': 'G', 'UA': 'PG', 'U/A': 'PG',
    'UA 7+': 'PG', '7': 'PG', 'M/PG': 'PG', 'GP': 'PG', 'UA 13+': 'PG-13', '12': 'PG-13',
    '12+': 'PG-13', '13': 'PG-13', 'UA 16+': 'R', 'U/A 16+': 'R', '15+': 'R', '16': 'R',
    '16+': 'R', 'X': 'NC-17','18': 'NC-17', '18+': 'NC-17', 'Not Rated': 'NR', 'Unrated': 'NR', '(Banned)': 'NR'}

imdb['Certificate'] = imdb['Certificate'].replace(conversion)

#Part D: convert columns to float types

#iMDb Audience Reviews
imdb['iMDb Audience Reviews'] = imdb['iMDb Audience Reviews'].str.replace(r'\,', '', regex=True)
imdb['iMDb Audience Reviews'] = imdb['iMDb Audience Reviews'].astype(float)

#iMDb Critic Reviews
imdb['iMDb Critic Reviews'] = imdb['iMDb Critic Reviews'].str.replace(r'\,', '', regex=True)
imdb['iMDb Critic Reviews'] = imdb['iMDb Critic Reviews'].astype(float)

#multiply the Critics Score column by 10
imdb['iMDb Critics Score'] = imdb['iMDb Critics Score'] * 10
```

Figure 3: Code Used to Clean the iMDb Dataset

From here, we can easily combine these two datasets by their Title columns with the pd.merge() command, and we can then drop duplicate columns and reorder the remaining columns with similar methods as utilized previously. Figure 4 shows a preview of this dataset that we will be analyzing, which contains 685 unique movie titles.

	Title	Certificate	Director	Runtime	Studio	Year	Genre	RT Critics Score	RT Critic Reviews	RT Audience Score	RT Audience Reviews	iMDb Critics Score	iMDb Critic Reviews	iMDb Audience Score	iMDb Audience Reviews
0	The Angry Birds Movie 2	PG	Thurop Van Orman	100.0	Columbia Pictures	2014.0	Animation	73.0	107.0	84.0	4023.0	60.0	167.0	64.0	35376.0
1	Dora and the Lost City of Gold	PG	James Bobin	102.0	Paramount Pictures	1985.0	Action	84.0	148.0	88.0	6715.0	63.0	291.0	61.0	34806.0
2	Good Boys	R	Gene Stupnitsky	95.0	Universal Pictures	2003.0	Adventure	80.0	237.0	86.0	13007.0	NaN	604.0	67.0	81961.0
3	Brian Banks	PG-13	Tom Shadyac	99.0	Bleecker Street	2023.0	Biography	61.0	69.0	97.0	1831.0	58.0	60.0	72.0	9274.0
4	The Farewell	PG	Lulu Wang	98.0	A24	2019.0	Comedy	98.0	322.0	87.0	2490.0	89.0	311.0	75.0	71293.0

Figure 4: Full Cleaned and Merged Dataset

4 Visual Analysis

Now that our data is in the format that we want, we can create visualizations and statistical summaries for it. As mentioned before, we want to find relationships or differences between the websites, and we want to see if there are certain types of movies that audiences or critics typically enjoy. We will be analyzing the Genre and Certificate columns, so it will be beneficial for us to create bar graphs and pie charts for these columns. These visualizations, shown in Figure 5, will help us understand the proportion of data within each category.

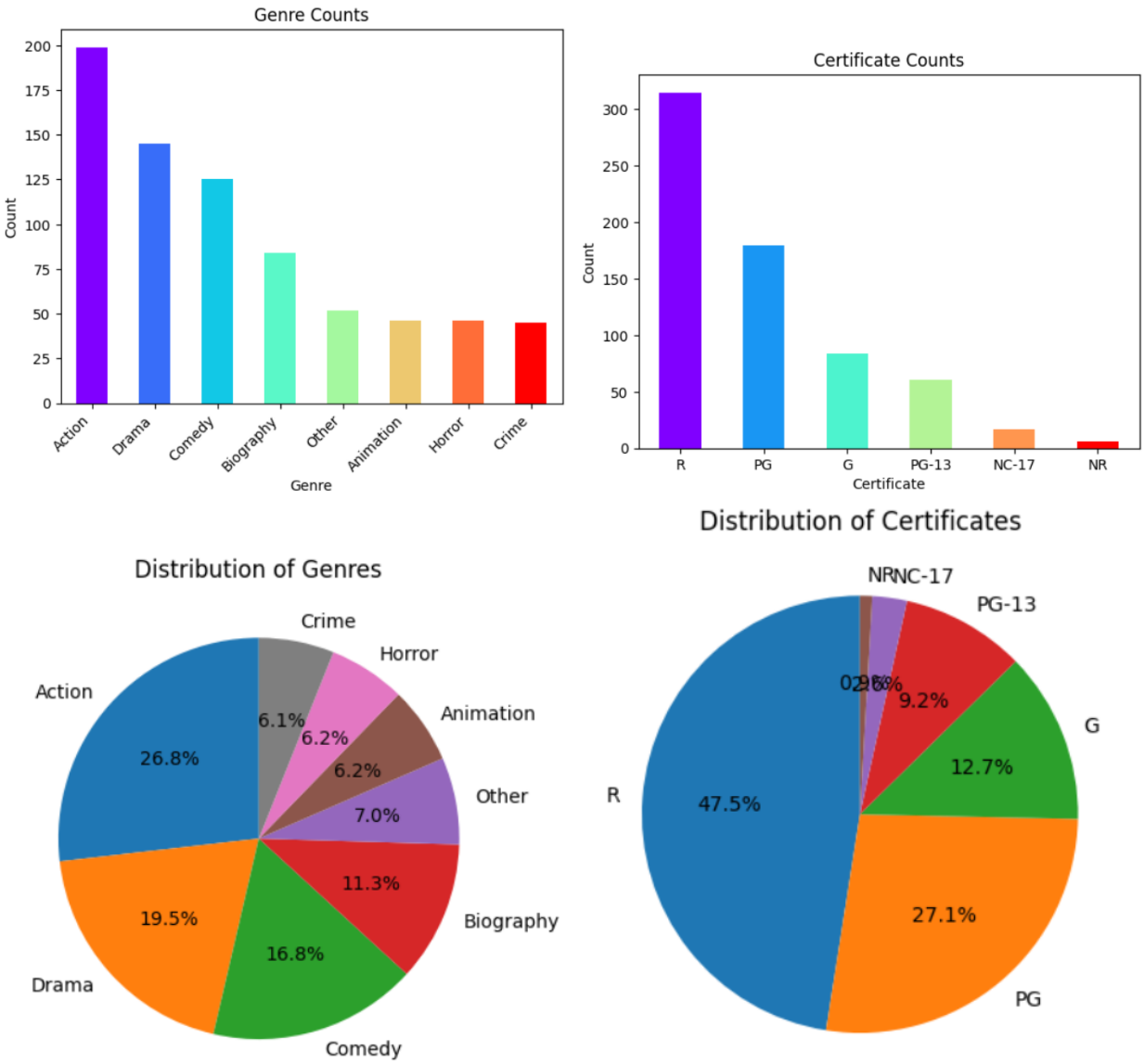


Figure 5: Proportions for the Genre and Certificate Columns

Next, we want to see which of these genres and certificates generally receive the best feedback from audiences and critics. To do this, we first average the Score columns and then create new conditional columns that classify the movie as "Fresh" if this average is greater than or equal to 60 or "Rotten" if otherwise. The code used to do this is shown in Figure 6. With these new columns made, we can use two-way tables to get a general idea of what percentage of the audiences/critics enjoyed or disliked the different categories of movies (See Figure 7).

```
#compute the average scores for Audience and Critics
merged['Audience Avg'] = merged[['RT Audience Score', 'iMDb Audience Score']].mean(axis=1)
merged['Critics Avg'] = merged[['RT Critics Score', 'iMDb Critics Score']].mean(axis=1)

#classify as 'Fresh' or 'Rotten' based on the average score
merged['Audience Classification'] = merged['Audience Avg'].apply(lambda x: 'Fresh' if x >= 60 else 'Rotten')
merged['Critics Classification'] = merged['Critics Avg'].apply(lambda x: 'Fresh' if x >= 60 else 'Rotten')

#drop the average columns
merged.drop(['Audience Avg', 'Critics Avg'], axis=1, inplace=True)
```

Figure 6: Code Used to Create New Classification Columns

Genre	Action	Animation	Biography	Comedy	Crime	Drama	Horror	Other
Audience Classification								
Fresh	0.51	0.59	0.8	0.53	0.62	0.58	0.48	0.62
Rotten	0.49	0.41	0.2	0.47	0.38	0.42	0.52	0.38

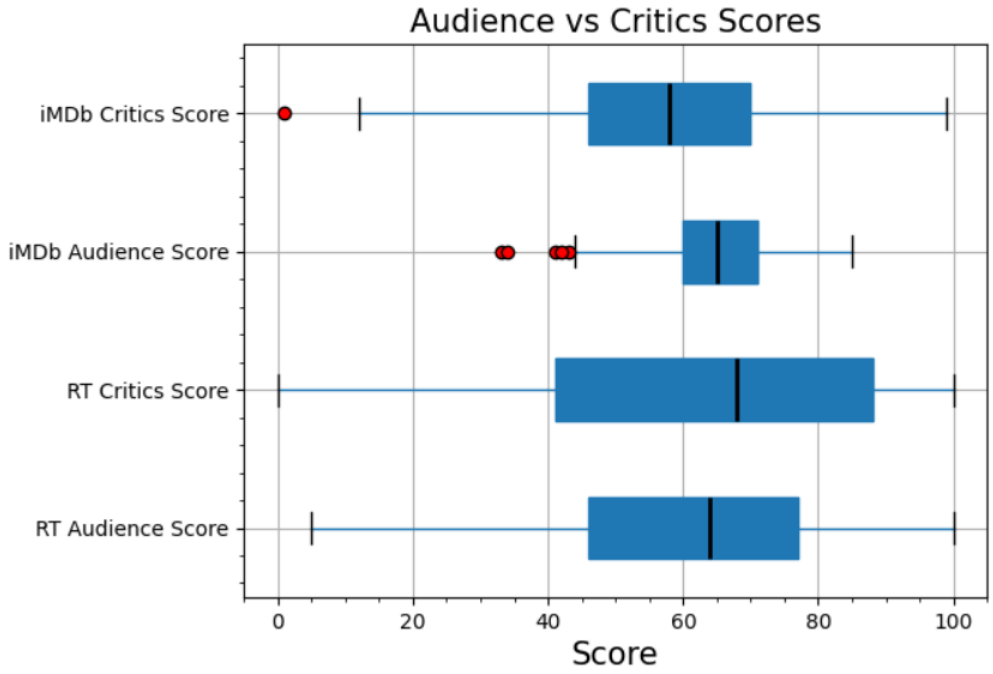
Genre	Action	Animation	Biography	Comedy	Crime	Drama	Horror	Other
Critics Classification								
Fresh	0.44	0.59	0.58	0.51	0.53	0.59	0.46	0.58
Rotten	0.56	0.41	0.42	0.49	0.47	0.41	0.54	0.42

Certificate	G	NC-17	NR	PG	PG-13	R
Audience Classification						
Fresh	0.6	0.47	0.33	0.54	0.75	0.57
Rotten	0.4	0.53	0.67	0.46	0.25	0.43

Certificate	G	NC-17	NR	PG	PG-13	R
Critics Classification						
Fresh	0.46	0.47	0.67	0.52	0.59	0.53
Rotten	0.54	0.53	0.33	0.48	0.41	0.47

Figure 7: Two-Way Tables: Genres and Certificates vs. Audience/Critics Classification

Moving on from this categorical data, let's now analyze the numerical data, particularly that of the Score columns. By first creating box plots and a statistical summaries for these columns, we can get a great overview of how the audiences and critics rate movies, on average, for each website (See Figure 8).



	RT Audience Score	RT Critics Score	iMDb Audience Score	iMDb Critics Score
count	742.00	742.00	738.00	683.00
mean	60.60	62.15	64.23	57.86
std	20.00	28.03	8.67	17.06
min	5.00	0.00	31.00	1.00
25%	45.25	39.25	59.00	46.00
50%	63.00	67.00	64.00	58.00
75%	77.00	88.00	70.75	70.00
max	100.00	100.00	85.00	99.00

Figure 8: Box Plots and Statistical Summaries for the Score Columns

Another helpful visualization to compare these scores is a histogram, which is similar to a box plot in its ability to show the spread a distribution of data. Figure 9 shows histograms for each of the Score columns. Included on these is a density curve, which shows the general pattern of the data.

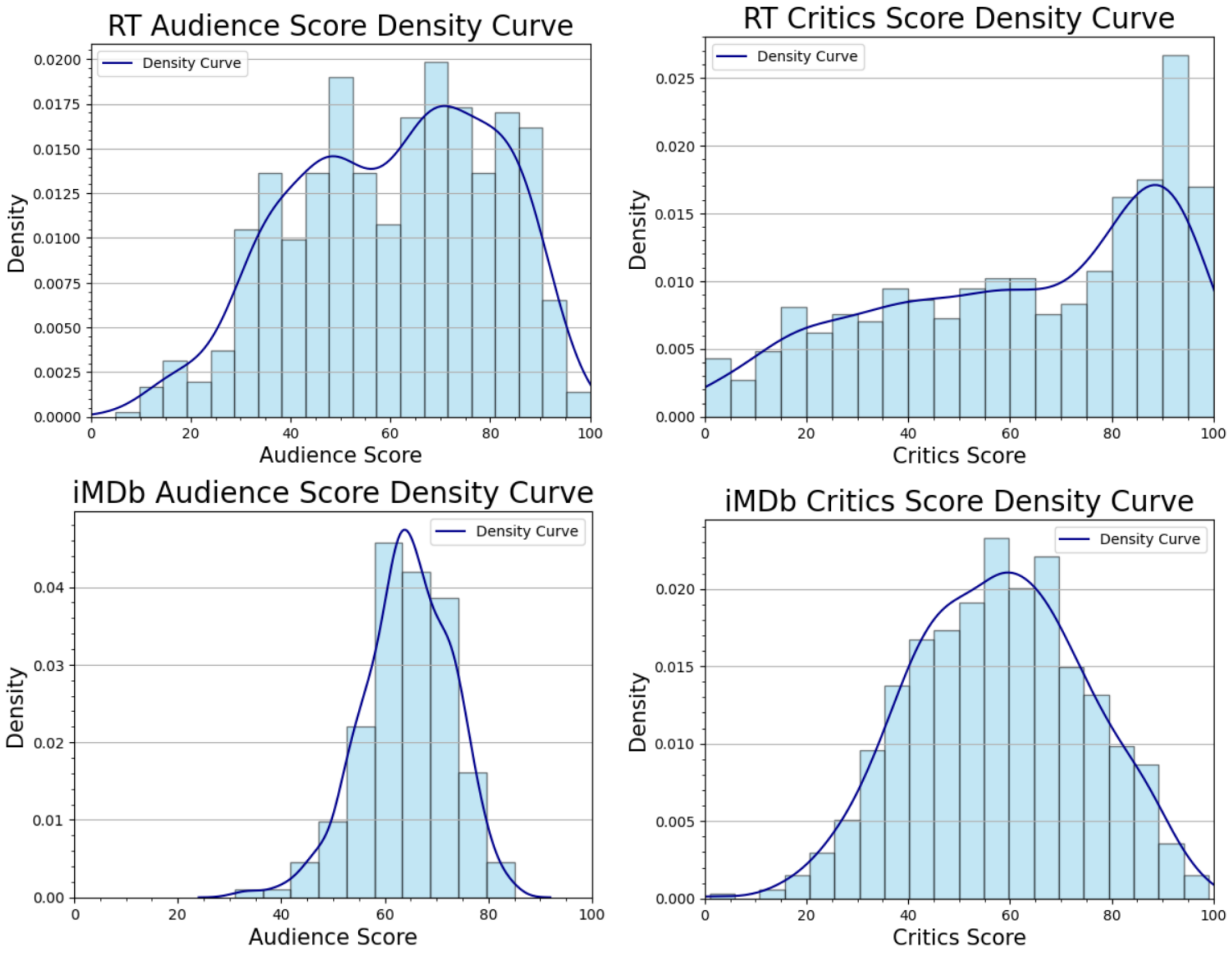


Figure 9: Histograms for the Score Columns

5 Discussion

We can now discuss what trends these visualizations reveal about the data. Beginning with the bar graphs and pie charts in Figure 5, we see that the genres are more evenly spread out than the certificates, with the highest proportion of movies being "Action" at about 27%, and all categories comprising at least 6% of the full data. The certificates, however, see R-rated and PG-rated movies alone fill 75% of that dataset, while categories such as NR and NC-17 barely have any of the entries. In general, larger sample sizes will yield more accurate data, while samples that are too small are often unreliable for our purposes. So, this information is very helpful to provide context for our further discussion of these categories, where we examine the two-way tables in Figure 7. Most of the categories seem to be pretty even in their classifications, with the percentages for "Fresh" and "Rotten" ranging from 0.4 to 0.6 in the majority of columns. However, we do see some significant results, with the audience classifying 80% of biographies and 75% of PG-13-rated movies as "Fresh", Apart from that, we find that the audience classifies 67% of NR-rated movies as "Rotten", while the critics

classify 67% of that same category as "Fresh", but as mentioned previously, this is likely misleading due to the very small sample size for NR.

Moving on to the quantitative data, we can start by discussing the box plots and statistical summaries in Figure 8. We first notice that there is little variation within the iMDb audience scores, with the a minimum value of 31 and a maximum of 85. For the iMDb critics, RT critics, and RT audience, though, the scores nearly cover the entire possible range from 0 to 100. In fact, the RT critics scores do encompass this whole range, and we see larger IQR and standard deviation values for the RT website in general as compared to iMDb. Another difference to note is that the iMDb data does not have audience entries for 4 of the movies, and it does not have critics scores for 59 of the movies, while RT has both scores present for every movie. As for a similarity between the two websites, one common measure we can see is center, as both the means and medians are fairly similar across the board. However, we do notice that the critics rate movies slightly lower, on average, than the audience does on iMDb, and the opposite is true for RT. Now, let's shift our focus to the histograms shown in Figure 9. These provide us with a better view of the variations, and they show us how much of the data is contained within different score ranges. Here, the iMDb scores for both audiences and critics follow a fairly "normal" density curve, while the data for RT seems to be left-skewed in both cases. In particular, the RT critics data is significantly skewed left, with a large portion of the data present in the highest two score ranges.

6 Conclusion

In summary, by cleaning and visualizing movie datasets for the Rotten Tomatoes and iMDb websites, we were able to answer our prompted questions fairly well. We were not able to find any significant movie preferences for the critics, but we did identify that the audiences tend to enjoy movies under the "Biography" genre as well as those that are rated PG-13. Additionally, we found that there is a lot more variation in scores for the Rotten Tomatoes website as compared to iMDb, and the highest scores seem to be given the most frequently by critics on RT.