rotten-tomatoes-project

July 30, 2025

1 Mid-Course Project Solutions

This project analyzes movie rating data sourced from Rotten Tomatoes. The dataset includes various features such as movie titles, genres, release dates, runtimes, critic scores (Tomatometer), and audience ratings.

The main objective of this analysis is to explore patterns in how movies are rated by critics versus general audiences, especially for films released from 2010 onwards. We clean and filter the data, visualize trends, and compare different genres and rating behaviors.

1.1 0. Read in the Data

[31]:
```python
# rotten tomatoes movie data set from Maven's data playground
import pandas as pd

movies = pd.read_csv('../Data/Rotten Tomatoes Movies.csv')
movies.head(3)
```

[31]:
```
                                         movie_title  \
0  Percy Jackson & the Olympians: The Lightning T…
1                                        Please Give
2                                                 10

                                          movie_info  \
0  A teenager discovers he's the descendant of a …
1  Kate has a lot on her mind. There's the ethics…
2  Blake Edwards' 10 stars Dudley Moore as George…

                                   critics_consensus rating  \
0  Though it may seem like just another Harry Pot…     PG
1  Nicole Holofcener's newest might seem slight i…      R
2                                                NaN      R

                                               genre          directors  \
0  Action & Adventure, Comedy, Drama, Science Fic…     Chris Columbus
1                                             Comedy  Nicole Holofcener
2                                    Comedy, Romance      Blake Edwards
```

```
             writers                                               cast  \
0       Craig Titley  Logan Lerman, Brandon T. Jackson, Alexandra Da…
1  Nicole Holofcener  Catherine Keener, Amanda Peet, Oliver Platt, R…
2      Blake Edwards  Dudley Moore, Bo Derek, Julie Andrews, Robert …

  in_theaters_date on_streaming_date  runtime_in_minutes  \
0       2010-02-12        2010-06-29                83.0
1       2010-04-30        2010-10-19                90.0
2       1979-10-05        1997-08-27               118.0

              studio_name tomatometer_status  tomatometer_rating  \
0        20th Century Fox             Rotten                  49
1  Sony Pictures Classics    Certified Fresh                  86
2             Waner Bros.              Fresh                  68

   tomatometer_count  audience_rating  audience_count
0                144             53.0        254287.0
1                140             64.0         11567.0
2                 22             53.0         14670.0
```

[32]:

```
# let's work with a subset of the data for this project
movies = movies[['movie_title', 'rating', 'genre',␣
 ↪'in_theaters_date','runtime_in_minutes',
                 'tomatometer_rating', 'tomatometer_count', 'audience_rating',␣
 ↪'audience_count']]
movies.head()
```

[32]:

```
                                       movie_title rating  \
0  Percy Jackson & the Olympians: The Lightning T…     PG
1                                      Please Give      R
2                                               10      R
3                  12 Angry Men (Twelve Angry Men)     NR
4                     20,000 Leagues Under The Sea      G

                                             genre in_theaters_date  \
0  Action & Adventure, Comedy, Drama, Science Fic…       2010-02-12
1                                           Comedy       2010-04-30
2                                  Comedy, Romance       1979-10-05
3                                  Classics, Drama       1957-04-13
4         Action & Adventure, Drama, Kids & Family       1954-01-01

   runtime_in_minutes  tomatometer_rating  tomatometer_count  audience_rating  \
0                83.0                  49                144             53.0
1                90.0                  86                140             64.0
2               118.0                  68                 22             53.0
3                95.0                 100                 51             97.0
4               127.0                  89                 27             74.0
```

```
   audience_count
0        254287.0
1         11567.0
2         14670.0
3        105000.0
4         68860.0
```

1.2 1. Explore the Data

How many movies are in this data set?

[33]:
```
# number of rows and columns
movies.shape
```

[33]:
```
(16638, 9)
```

Filter the data to only include movies that came out in 2010 or later. How many movies are in this new data set?

[34]:
```
# you get an error when trying to use a datetime method
# movies.in_theaters_date.dt.year
```

[35]:
```
# check the data types
movies.dtypes
```

[35]:
```
movie_title             object
rating                  object
genre                   object
in_theaters_date        object
runtime_in_minutes     float64
tomatometer_rating       int64
tomatometer_count        int64
audience_rating        float64
audience_count         float64
dtype: object
```

[36]:
```
# convert the in_theatres_date to a datetime field
movies['in_theaters_date'] = pd.to_datetime(movies.in_theaters_date)
movies.head(3)
```

[36]:
```
                                        movie_title rating  \
0  Percy Jackson & the Olympians: The Lightning T…     PG
1                                       Please Give      R
2                                                10      R

                                              genre in_theaters_date  \
0  Action & Adventure, Comedy, Drama, Science Fic…       2010-02-12
```

```
1                                             Comedy       2010-04-30
2                                    Comedy, Romance       1979-10-05

   runtime_in_minutes  tomatometer_rating  tomatometer_count  audience_rating  \
0                83.0                  49                144             53.0
1                90.0                  86                140             64.0
2               118.0                  68                 22             53.0

   audience_count
0        254287.0
1         11567.0
2         14670.0
```

[37]:
```
# filter on only movies from the 2010's and newer
movies = movies[movies.in_theaters_date.dt.year >= 2010]
movies.head(3)
```

[37]:
```
                                          movie_title rating  \
0   Percy Jackson & the Olympians: The Lightning T…       PG
1                                         Please Give      R
97                            Fireflies in the Garden      R

                                                genre in_theaters_date  \
0   Action & Adventure, Comedy, Drama, Science Fic…         2010-02-12
1                                              Comedy       2010-04-30
97                                              Drama       2011-10-14

    runtime_in_minutes  tomatometer_rating  tomatometer_count  \
0                 83.0                  49                144
1                 90.0                  86                140
97                98.0                  22                 54

    audience_rating  audience_count
0              53.0        254287.0
1              64.0         11567.0
97             45.0         45150.0
```

[38]:
```
# find the number of movies
movies.shape
```

[38]:
```
(6053, 9)
```

Find the highest rated movies according to both critics (*tomatometer_rating*) and the general audience (*audience_rating*).

[39]:
```
# highest rated movies by critics
movies.sort_values('tomatometer_rating', ascending=False).head()
```

[39]:

```
                    movie_title rating  \
7318                High Ground     NR
11941               Rodney King     NR
1468                      11:55     NR
13051  Stations of the Elevated     NR
2592               Among Wolves     NR

                                                   genre in_theaters_date  \
7318                       Documentary, Special Interest       2012-11-02
11941                                              Drama       2017-04-28
1468                                               Drama       2017-06-09
13051  Documentary, Musical & Performing Arts, Specia…       2014-10-17
2592                                         Documentary       2019-02-08

       runtime_in_minutes  tomatometer_rating  tomatometer_count  \
7318                 91.0                 100                  8
11941                52.0                 100                  9
1468                 80.0                 100                  5
13051                45.0                 100                  8
2592                 94.0                 100                 12

       audience_rating  audience_count
7318              74.0           295.0
11941              NaN             NaN
1468              81.0           378.0
13051             33.0           124.0
2592              91.0           106.0
```

[40]:

```
# highest rated movies by the audience
movies.sort_values('audience_rating', ascending=False).head()
```

[40]:

```
                   movie_title rating                                 genre  \
14580  The Most Dangerous Year     NR                           Documentary
4239                Charm City     NR                           Documentary
14566        The Miners' Hymns     NR  Documentary, Drama, Special Interest
4027    Calling All Earthlings     NR                           Documentary
7137                     Haunt      R            Horror, Mystery & Suspense

      in_theaters_date  runtime_in_minutes  tomatometer_rating  \
14580       2019-04-12                90.0                  91
4239        2018-10-19               108.0                 100
14566       2012-02-08                52.0                 100
4027        2018-06-29                74.0                  58
7137        2019-09-13                92.0                  68

       tomatometer_count  audience_rating  audience_count
14580                 11            100.0            40.0
```

```
4239                  16          100.0           24.0
14566                 10          100.0          148.0
4027                  12          100.0           34.0
7137                  38          100.0            7.0
```

These top movies seem to have very few critics and audience members writing the reviews. We want to look at only the most popular movies. Filter the movies data set to only include movies that have 100k+ audience ratings. How many movies are in this data set?

[41]:
```python
# there are about 300 movies for us to work with
movies_popular = movies[movies.audience_count > 100000]
movies_popular.shape
```

[41]:
```
(316, 9)
```

Find the highest rated **popular** movies according to both critics (*tomatometer_rating*) and the general audience (*audience_rating*).

[42]:
```python
# highest rated popular movies by critics
movies_popular.sort_values('tomatometer_rating', ascending=False).head()
```

[42]:
```
                    movie_title rating  \
7558   How to Train Your Dragon     PG
7925                 Inside Out     PG
15416               Toy Story 3      G
16634                  Zootopia     PG
9355         Mad Max: Fury Road      R

                                                   genre in_theaters_date  \
7558   Animation, Kids & Family, Science Fiction & Fa…       2010-03-26
7925                            Animation, Kids & Family       2015-06-19
15416                   Animation, Comedy, Kids & Family       2010-06-18
16634              Action & Adventure, Animation, Comedy       2016-03-04
9355       Action & Adventure, Science Fiction & Fantasy       2015-05-15

       runtime_in_minutes  tomatometer_rating  tomatometer_count  \
7558                 98.0                  99                208
7925                 94.0                  98                357
15416               103.0                  98                305
16634               108.0                  97                279
9355                120.0                  97                410

       audience_rating  audience_count
7558              91.0        312342.0
7925              89.0        136125.0
15416             89.0        606931.0
16634             92.0        100946.0
9355              85.0        127428.0
```

[43]:

```
# highest rated popular movies by the audience
movies_popular.sort_values('audience_rating', ascending=False).head()
```

[43]:

```
                               movie_title rating  \
16634                             Zootopia     PG
14397                    The King's Speech  PG-13
4077   Captain America: The Winter Soldier  PG-13
6950               Guardians of the Galaxy  PG-13
14549                          The Martian  PG-13

                                               genre in_theaters_date  \
16634          Action & Adventure, Animation, Comedy       2016-03-04
14397                                          Drama       2010-11-26
4077   Action & Adventure, Science Fiction & Fantasy       2014-04-04
6950   Action & Adventure, Science Fiction & Fantasy       2014-08-01
14549                      Science Fiction & Fantasy       2015-10-02

       runtime_in_minutes  tomatometer_rating  tomatometer_count  \
16634               108.0                  97                279
14397               118.0                  95                292
4077                136.0                  90                292
6950                121.0                  91                316
14549               164.0                  91                361

       audience_rating  audience_count
16634             92.0        100946.0
14397             92.0        144306.0
4077              92.0        281524.0
6950              92.0        254717.0
14549             91.0        131093.0
```

A lot of these popular movies seem to have a PG or PG-13 rating. How many movies fall under each type of rating?

Use this popular movies data set going forward in this notebook.

[44]:

```
# number of movies that fall under each type of rating
movies_popular.rating.value_counts()
```

[44]:

```
rating
PG-13    160
R        100
PG        51
G          5
Name: count, dtype: int64
```

What is the average audience rating for each movie rating type? Which rating type is most highly rated?

[45]:

```
# PG-13 movies are most highly rated
movies_popular.groupby('rating')['audience_rating'].mean()
```

[45]:

```
rating
G        66.200000
PG       66.823529
PG-13    67.293750
R        63.010000
Name: audience_rating, dtype: float64
```

1.3 2. Create New Columns

Create a column in the DataFrame called 'Animation' and return a 1 if a movie is an 'Animation' movie and 0 otherwise. Do the same for *Action & Adventure* and *Comedy*.

Hint: use np.where and str.contains

[46]:

```
movies_popular.head()
```

[46]:

```
                                           movie_title rating  \
0    Percy Jackson & the Olympians: The Lightning T…     PG
248                                        Tron Legacy      PG
265                                      The Last Song      PG
274                                           Repo Men       R
284                                          Predators       R

                                                 genre in_theaters_date  \
0    Action & Adventure, Comedy, Drama, Science Fic…       2010-02-12
248      Action & Adventure, Science Fiction & Fantasy       2010-12-17
265                      Drama, Kids & Family, Romance       2010-03-31
274      Action & Adventure, Science Fiction & Fantasy       2010-03-19
284  Action & Adventure, Horror, Science Fiction & …       2010-07-09

     runtime_in_minutes  tomatometer_rating  tomatometer_count  \
0                  83.0                  49                144
248               125.0                  51                239
265               107.0                  20                118
274               119.0                  22                151
284               107.0                  65                198

     audience_rating  audience_count
0               53.0        254287.0
248             63.0        171385.0
265             66.0        160777.0
274             41.0        100453.0
284             52.0        159760.0
```

[47]:

```
import numpy as np

movies_popular['Animation'] = np.where(movies_popular.genre.str.
  ↪contains('Animation'), 1, 0)
```

```
C:\Users\Dell\AppData\Local\Temp\ipykernel_11528\1899839453.py:3:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  movies_popular['Animation'] =
np.where(movies_popular.genre.str.contains('Animation'), 1, 0)
```

[48]:

```
# copy the movie to avoid a warning
movies_popular = movies[movies.audience_count > 100000].copy()
```

[49]:

```
movies_popular['Animation'] = np.where(movies_popular.genre.str.
  ↪contains('Animation'), 1, 0)
```

[50]:

```
movies_popular['Action & Adventure'] = np.where(movies_popular.genre.str.
  ↪contains('Action & Adventure'), 1, 0)
```

[51]:

```
movies_popular['Comedy'] = np.where(movies_popular.genre.str.
  ↪contains('Comedy'), 1, 0)
```

Create a table where each row is a rating, each column is a genre and each value is the number of movies of that particular rating and genre. What insights do you gather?

[52]:

```
movies_popular.groupby('rating')[['Animation', 'Action & Adventure', 'Comedy']].
  ↪sum()
```

[52]:

```
        Animation  Action & Adventure  Comedy
rating
G               5                   3       5
PG             26                  27      29
PG-13           0                 102      35
R               0                  41      35
```

Find the average critic and audience rating for an Animation movie vs a non-Animation movie. Do the same for Action & Adventure and Comedy. What insights do you gather?

[53]:

```
# both critics and the general audience love animated movies
movies_popular.groupby('Animation')[['tomatometer_rating', 'audience_rating']].
  ↪mean()
```

[53]:
```
           tomatometer_rating  audience_rating
Animation
0                   58.340351        64.831579
1                   75.258065        75.161290
```

[54]:
```
# the general audience likes action movies more than critics
movies_popular.groupby('Action & Adventure')[['tomatometer_rating',␣
 ↪'audience_rating']].mean()
```

[54]:
```
                    tomatometer_rating  audience_rating
Action & Adventure
0                            59.111888        65.391608
1                            60.734104        66.219653
```

[55]:
```
# comedies have lower ratings than other genres
movies_popular.groupby('Comedy')[['tomatometer_rating', 'audience_rating']].
 ↪mean()
```

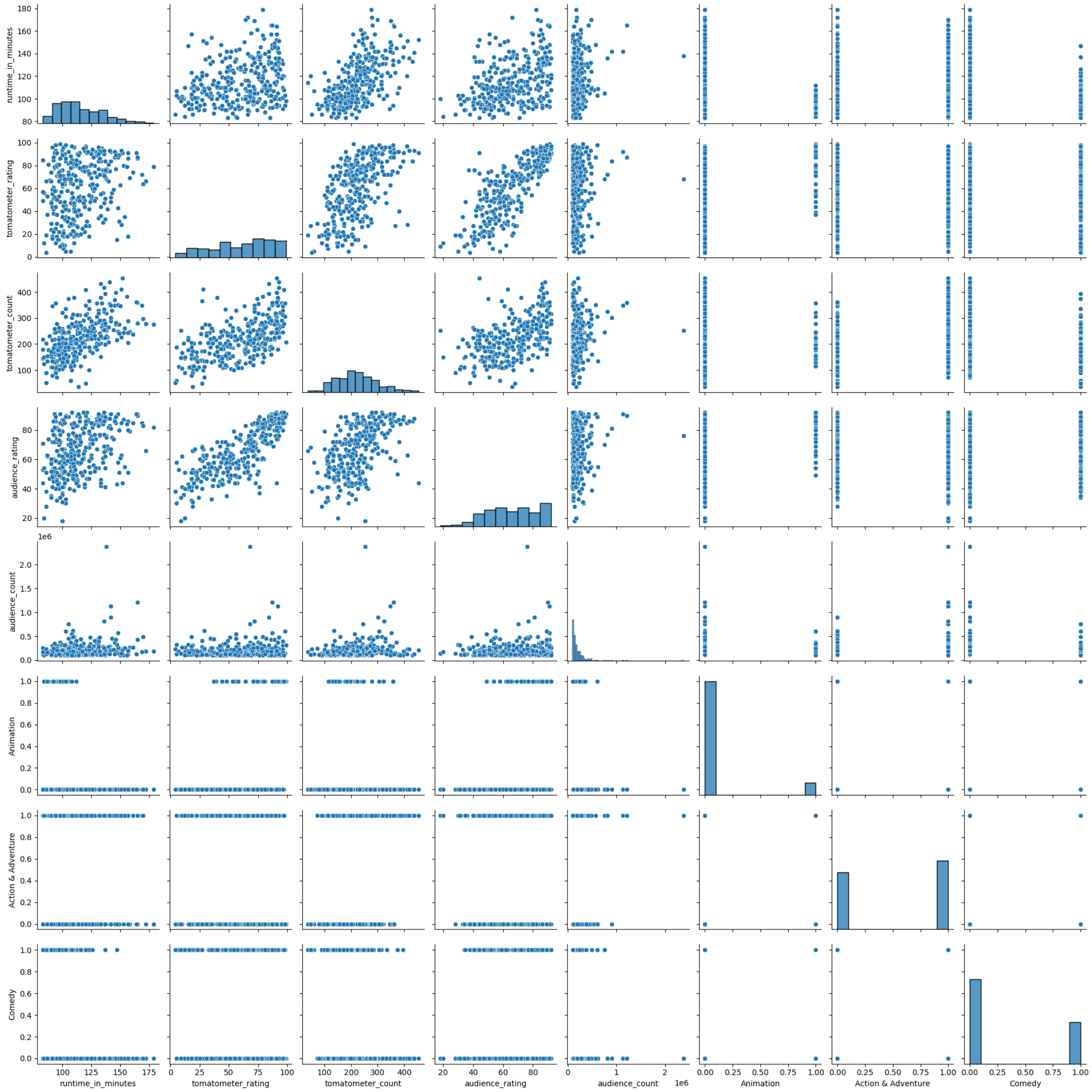
[55]:
```
        tomatometer_rating  audience_rating
Comedy
0                62.169811        67.353774
1                55.576923        62.769231
```

1.4 3. Visualize the Data

Create a pair plot from the popular movies DataFrame.

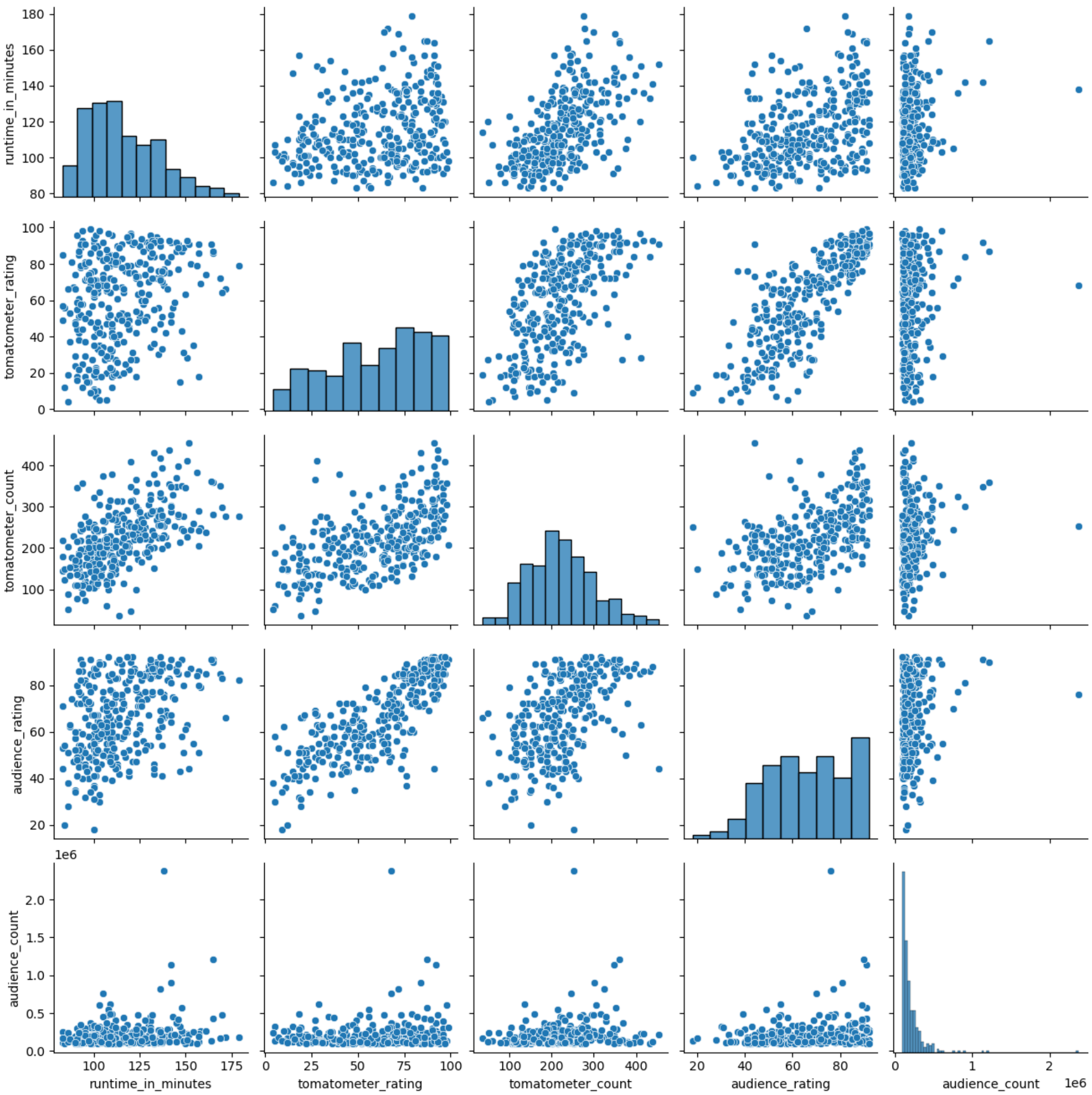
[56]:
```
import seaborn as sns
```

[57]:
```
# this chart has too many plots
sns.pairplot(movies_popular);
```



[58]:

```
# excluding the newly created columns
sns.pairplot(movies_popular.iloc[:, :-3]);
```



What insights can you gather from this pair plot? * How do the critic ratings (tomatometer_rating) compare with the audience ratings (compare the histograms)? * What are some surprising findings about the run times of movies compared with other fields (look at the scatter plots)? * What is the most popular movie by far in terms of the number of audience ratings?

[59]:
```
# critics give harsher reviews -- there are quite a few low ratings in the tomatometer histogram
# the run time of movies seems to be correlated with the number of critic ratings
# the most popular movie is Shutter Island with lots of audience ratings and not as many critic ratings --
## this is so extreme that it could potentially be an outlier / error
```

[60]:

```
movies_popular[movies_popular.audience_count > 1000000]
```

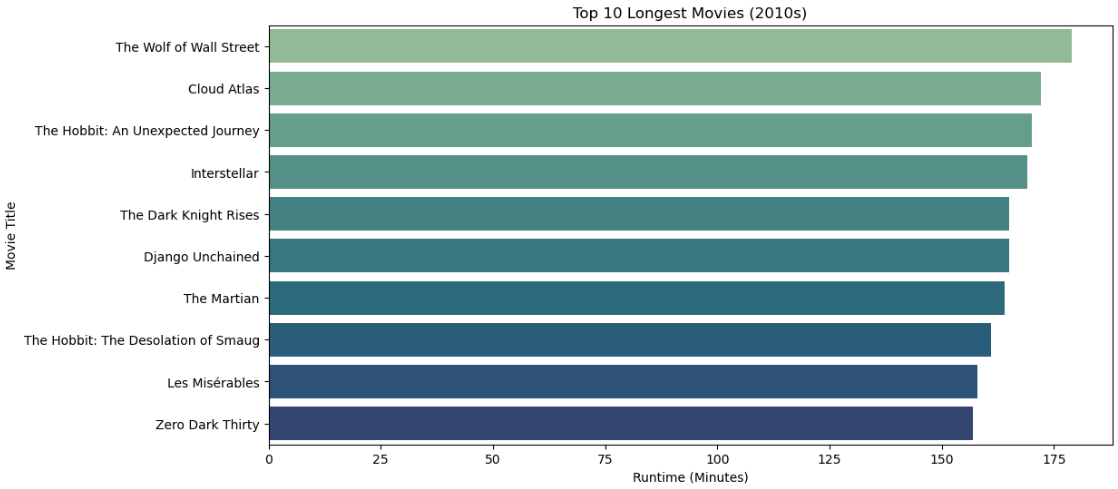
[60]:

```
                movie_title rating  \
1646          Shutter Island      R
9581   Marvel's The Avengers  PG-13
13936  The Dark Knight Rises  PG-13

                                                 genre in_theaters_date  \
1646   Action & Adventure, Drama, Mystery & Suspense       2010-02-19
9581   Action & Adventure, Science Fiction & Fantasy       2012-05-04
13936  Action & Adventure, Drama, Mystery & Suspense       2012-07-20

       runtime_in_minutes  tomatometer_rating  tomatometer_count  \
1646                138.0                  68                253
9581                142.0                  92                348
13936               165.0                  87                360

       audience_rating  audience_count  Animation  Action & Adventure  Comedy
1646              76.0       2373625.0          0                   1       0
9581              91.0       1134955.0          0                   1       0
13936             90.0       1210957.0          0                   1       0
```

[66]:

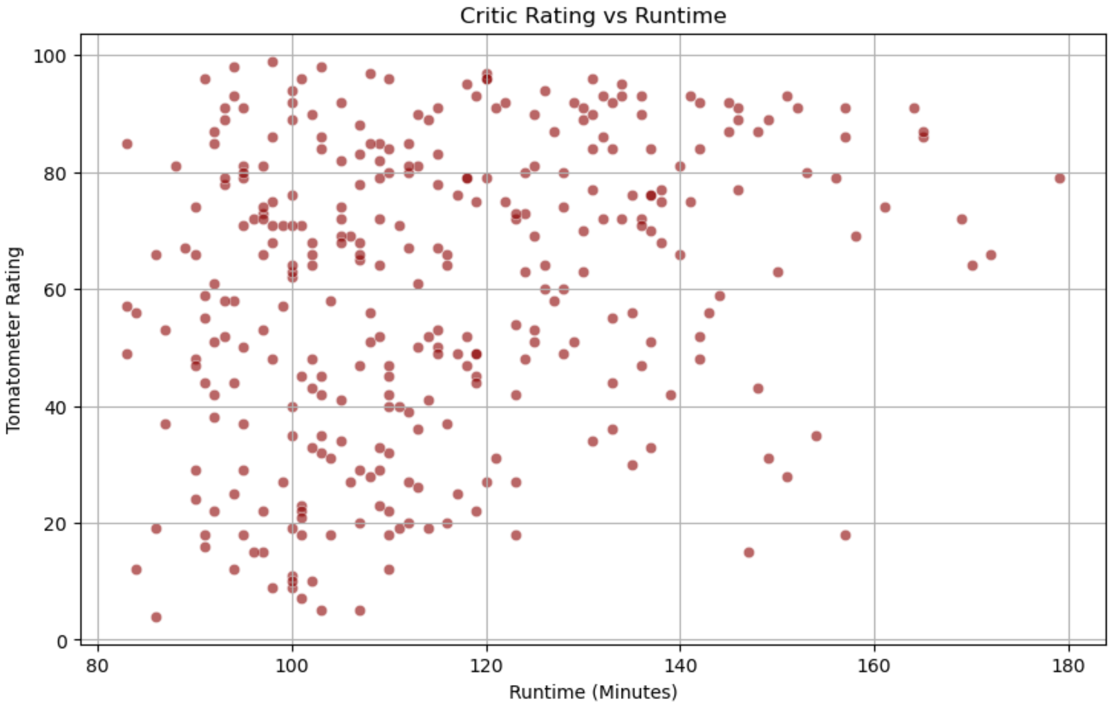
```
longest_movies = movies_popular.sort_values(by='runtime_in_minutes',␣
 ↪ascending=False).head(10)

plt.figure(figsize=(12, 6))
sns.barplot(data=longest_movies, x='runtime_in_minutes', y='movie_title',␣
 ↪palette='crest')
plt.title('Top 10 Longest Movies (2010s)')
plt.xlabel('Runtime (Minutes)')
plt.ylabel('Movie Title')
plt.show()
```

```
C:\Users\Dell\AppData\Local\Temp\ipykernel_11528\2994786234.py:4: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in
v0.14.0. Assign the `y` variable to `hue` and set `legend=False` for the same
effect.

  sns.barplot(data=longest_movies, x='runtime_in_minutes', y='movie_title',
palette='crest')
```



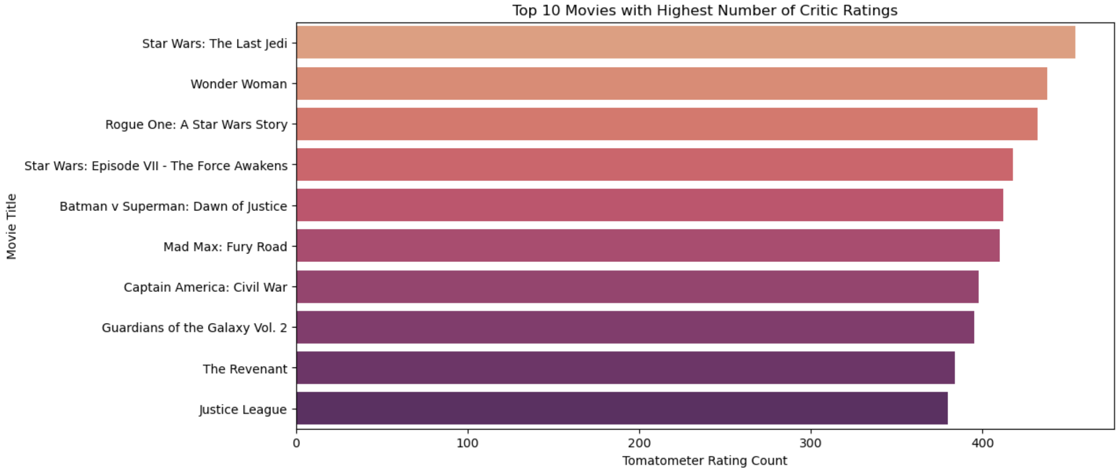
[68]:

```
plt.figure(figsize=(10, 6))
sns.scatterplot(data=movies_popular, x='runtime_in_minutes',␣
 ↪y='tomatometer_rating', alpha=0.6, color='darkred')
plt.title('Critic Rating vs Runtime')
plt.xlabel('Runtime (Minutes)')
plt.ylabel('Tomatometer Rating')
plt.grid(True)
plt.show()
```

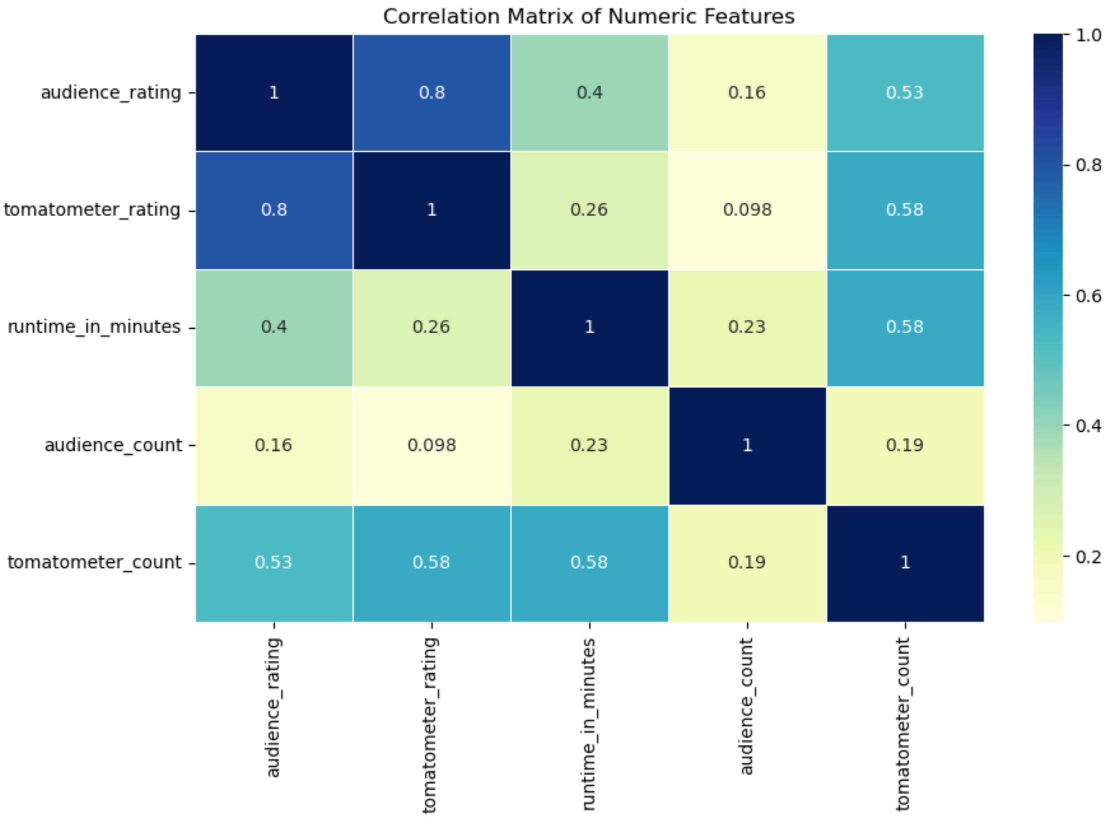


[70]:
```python
most_rated_by_critics = movies_popular.sort_values('tomatometer_count',␣
 ↪ascending=False).head(10)

plt.figure(figsize=(12, 6))
sns.barplot(data=most_rated_by_critics, x='tomatometer_count', y='movie_title',␣
 ↪palette='flare')
plt.title('Top 10 Movies with Highest Number of Critic Ratings')
plt.xlabel('Tomatometer Rating Count')
plt.ylabel('Movie Title')
plt.show()
```

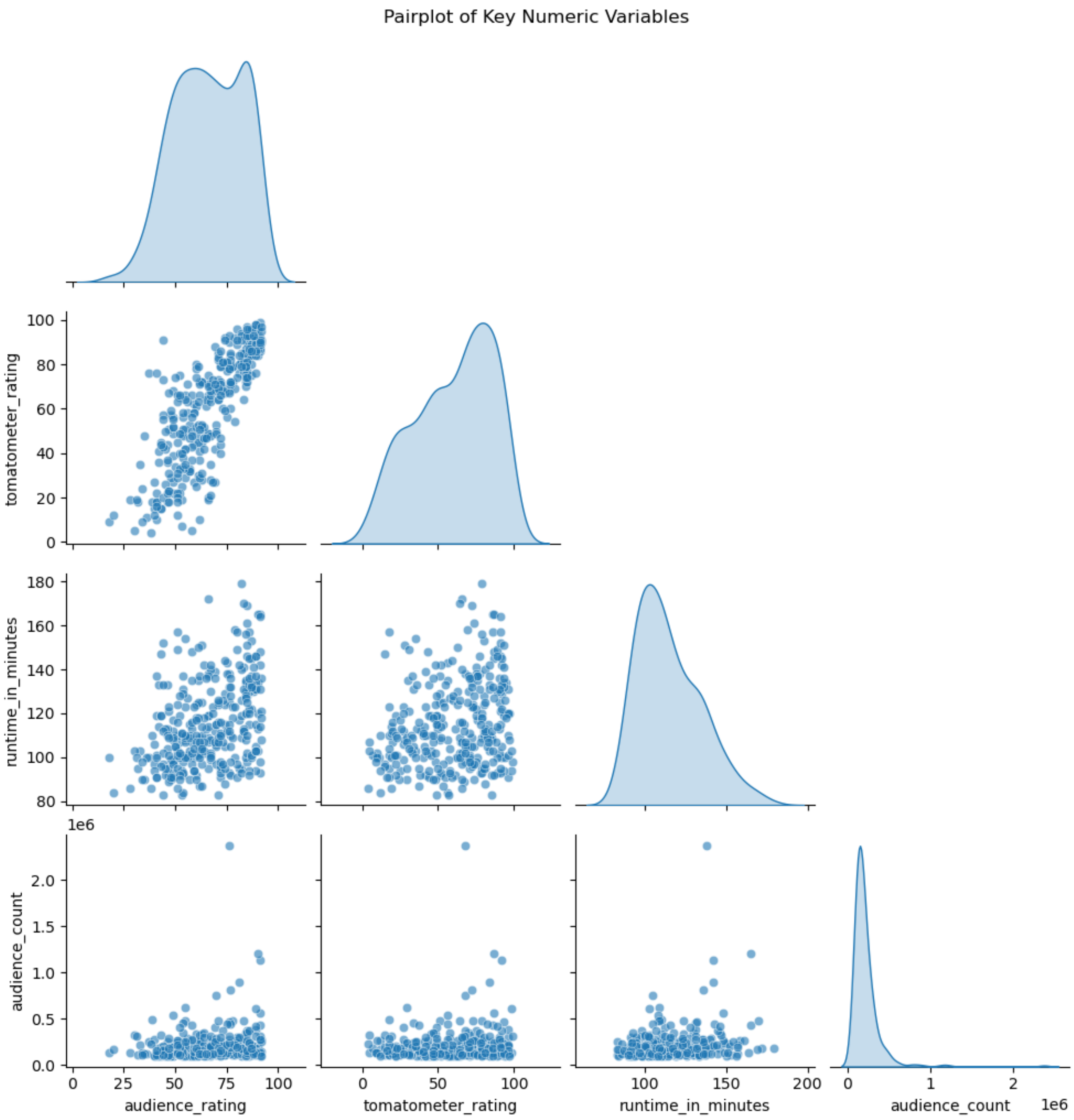
```
C:\Users\Dell\AppData\Local\Temp\ipykernel_11528\1177114352.py:4: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in
v0.14.0. Assign the `y` variable to `hue` and set `legend=False` for the same
effect.

  sns.barplot(data=most_rated_by_critics, x='tomatometer_count',
y='movie_title', palette='flare')
```



[72]:
```python
plt.figure(figsize=(10, 6))
numeric_cols = movies_popular[['audience_rating', 'tomatometer_rating',␣
 ↪'runtime_in_minutes', 'audience_count', 'tomatometer_count']]
corr_matrix = numeric_cols.corr()
sns.heatmap(corr_matrix, annot=True, cmap='YlGnBu', linewidths=0.5)
plt.title('Correlation Matrix of Numeric Features')
plt.show()
```



[75]:
```
sns.pairplot(movies_popular[['audience_rating', 'tomatometer_rating', 'runtime_in_minutes', 'audience_count']],
             corner=True, diag_kind='kde', plot_kws={'alpha':0.6})
plt.suptitle('Pairplot of Key Numeric Variables', y=1.02)
plt.show()
```


Pairplot of Key Numeric Variables
tomatometer_rating
runtime_in_minutes
audience_count
1e6
audience_rating
tomatometer_rating
runtime_in_minutes
audience_count
1e6
0
20
40
60
80
100
80
100
120
140
160
180
0.0
0.5
1.0
1.5
2.0
0
25
50
75
100
0
50
100
100
150
200
0
1
2

```
[ ]:
```