Netflix

April 21, 2025

1 EDA Netflix data

2 Problem Statements tackled in the project

Q1: What is the most frequent genre in the dataset?

Q2: What genres has highest votes ?

Q3: What movie got the highest popularity ? what's its genre ?

Q4: What movie got the lowest popularity ? what's its genre ?

Q5: Which year has the most filmmed movies?

3 Data cleaning and preparation

[64]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

[65]:
```
df= pd.read_csv('netflix_raw.csv' , lineterminator= '\n')
```

[66]:
```
df.head()
```

[66]:

	Release_Date	Title	Overview	Popularity	Vote_Count	\
0	2021-12-15	Spider-Man: No Way Home	Peter Parker is unmasked and no longer able to…	5083.954	8940	
1	2022-03-01	The Batman	In his second year of fighting crime, Batman u…	3827.658	1151	
2	2022-02-25	No Exit	Stranded at a rest stop in the mountains durin…	2618.087	122	
3	2021-11-24	Encanto	The tale of an extraordinary family, the Madri…	2402.201	5076	
4	2021-12-22	The King's Man	As a collection of history's worst tyrants and…	1895.511	1793	

```
   Vote_Average Original_Language                                Genre  \
0           8.3                en  Action, Adventure, Science Fiction
1           8.1                en           Crime, Mystery, Thriller
2           6.3                en                            Thriller
3           7.7                en  Animation, Comedy, Family, Fantasy
4           7.0                en    Action, Adventure, Thriller, War

                                        Poster_Url
0  https://image.tmdb.org/t/p/original/1g0dhYtq4i…
1  https://image.tmdb.org/t/p/original/74xTEgt7R3…
2  https://image.tmdb.org/t/p/original/vDHsLnOWKl…
3  https://image.tmdb.org/t/p/original/4j0PNHkMr5…
4  https://image.tmdb.org/t/p/original/aq4Pwv5Xeu…
```

[ ]:

Checking for Null Values

[ ]:

[67]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9827 entries, 0 to 9826
Data columns (total 9 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   Release_Date       9827 non-null   object
 1   Title              9827 non-null   object
 2   Overview           9827 non-null   object
 3   Popularity         9827 non-null   float64
 4   Vote_Count         9827 non-null   int64
 5   Vote_Average       9827 non-null   float64
 6   Original_Language  9827 non-null   object
 7   Genre              9827 non-null   object
 8   Poster_Url         9827 non-null   object
dtypes: float64(2), int64(1), object(6)
memory usage: 691.1+ KB
```

[ ]:

Checking for duplicate rows

[ ]:

[68]:
```
df.duplicated().sum()
```

[68]: 0

[ ]:

A quick descriptive analysis to understand the spread and the data better

[ ]:

[69]:
```
df.describe()
```

[69]:
```
         Popularity    Vote_Count  Vote_Average
count  9827.000000   9827.000000   9827.000000
mean     40.326088   1392.805536      6.439534
std     108.873998   2611.206907      1.129759
min      13.354000      0.000000      0.000000
25%      16.128500    146.000000      5.900000
50%      21.199000    444.000000      6.500000
75%      35.191500   1376.000000      7.100000
max    5083.954000  31077.000000     10.000000
```

[ ]:

Parsing the Release_date column and extracting the year of release

[ ]:

[70]:
```
df['Release_Date'] = pd.to_datetime(df['Release_Date'])
print(df['Release_Date'].dtypes)
```

```
datetime64[ns]
```

[71]:
```
df['Release_Date'] = df['Release_Date'].dt.year
df['Release_Date'].dtypes
```

[71]:
```
dtype('int32')
```

[72]:
```
df.head()
```

[72]:
```
   Release_Date                  Title  \
0          2021  Spider-Man: No Way Home
1          2022               The Batman
2          2022                  No Exit
3          2021                  Encanto
4          2021          The King's Man

                                            Overview  Popularity  Vote_Count  \
0  Peter Parker is unmasked and no longer able to…    5083.954        8940
1  In his second year of fighting crime, Batman u…    3827.658        1151
2  Stranded at a rest stop in the mountains durin…    2618.087         122
3  The tale of an extraordinary family, the Madri…    2402.201        5076
```

```
4  As a collection of history's worst tyrants and…    1895.511        1793

   Vote_Average Original_Language                               Genre  \
0           8.3                en  Action, Adventure, Science Fiction
1           8.1                en            Crime, Mystery, Thriller
2           6.3                en                            Thriller
3           7.7                en  Animation, Comedy, Family, Fantasy
4           7.0                en    Action, Adventure, Thriller, War

                                        Poster_Url
0  https://image.tmdb.org/t/p/original/1g0dhYtq4i…
1  https://image.tmdb.org/t/p/original/74xTEgt7R3…
2  https://image.tmdb.org/t/p/original/vDHsLnOWKl…
3  https://image.tmdb.org/t/p/original/4j0PNHkMr5…
4  https://image.tmdb.org/t/p/original/aq4Pwv5Xeu…
```

[ ]:

Dropping unwanted columns

[ ]:

[73]:
```
cols = ['Overview','Original_Language','Poster_Url']
```

[74]:
```
df.drop(cols, axis=1 , inplace = True)
df.columns
```

[74]:
```
Index(['Release_Date', 'Title', 'Popularity', 'Vote_Count', 'Vote_Average',
       'Genre'],
      dtype='object')
```

[75]:
```
df.head()
```

[75]:
```
   Release_Date                    Title  Popularity  Vote_Count  \
0          2021  Spider-Man: No Way Home    5083.954        8940
1          2022               The Batman    3827.658        1151
2          2022                  No Exit    2618.087         122
3          2021                  Encanto    2402.201        5076
4          2021          The King's Man    1895.511        1793

   Vote_Average                               Genre
0           8.3  Action, Adventure, Science Fiction
1           8.1            Crime, Mystery, Thriller
2           6.3                            Thriller
3           7.7  Animation, Comedy, Family, Fantasy
4           7.0    Action, Adventure, Thriller, War
```

[ ]:

categorising the vote column into 4 groups (not popular, average, good, popular)

[ ]:

[76]:
```
def categorize_col(df , col, labels):
    edges= [df[col].describe()['min'],
            df[col].describe()['25%'],
            df[col].describe()['50%'],
            df[col].describe()['75%'],
            df[col].describe()['max']]
    df[col]= pd.cut(df[col], edges, labels=labels, duplicates ='drop' )
    return df
```

[77]:
```
labels = ['Not_popular','Average','Good','Popular']
categorize_col(df , 'Vote_Average' ,labels)
df['Vote_Average'].unique()
```

[77]:
```
['Popular', 'Average', 'Good', 'Not_popular', NaN]
Categories (4, object): ['Not_popular' < 'Average' < 'Good' < 'Popular']
```

[78]:
```
df.head()
```

[78]:

	Release_Date	Title	Popularity	Vote_Count	Vote_Average	Genre
0	2021	Spider-Man: No Way Home	5083.954	8940	Popular	Action, Adventure, Science Fiction
1	2022	The Batman	3827.658	1151	Popular	Crime, Mystery, Thriller
2	2022	No Exit	2618.087	122	Average	Thriller
3	2021	Encanto	2402.201	5076	Popular	Animation, Comedy, Family, Fantasy
4	2021	The King's Man	1895.511	1793	Good	Action, Adventure, Thriller, War

[79]:
```
df['Vote_Average'].value_counts()
```

[79]:
```
Vote_Average
Not_popular    2467
Popular        2450
Good           2412
Average        2398
Name: count, dtype: int64
```

[80]:
```
df.dropna(inplace= True)
df.isna().sum()
```

[80]:
```
Release_Date    0
Title           0
Popularity      0
Vote_Count      0
Vote_Average    0
Genre           0
dtype: int64
```

[81]:
```
df.head()
```

[81]:
```
   Release_Date                  Title  Popularity  Vote_Count Vote_Average  \
0          2021  Spider-Man: No Way Home    5083.954        8940      Popular
1          2022               The Batman    3827.658        1151      Popular
2          2022                  No Exit    2618.087         122      Average
3          2021                  Encanto    2402.201        5076      Popular
4          2021          The King's Man    1895.511        1793         Good

                              Genre
0  Action, Adventure, Science Fiction
1           Crime, Mystery, Thriller
2                           Thriller
3  Animation, Comedy, Family, Fantasy
4   Action, Adventure, Thriller, War
```

[ ]:

Split genre into a list and then explode df to have 1 genre per row

[ ]:

[84]:
```
df['Genre'] = df['Genre'].fillna('').astype(str)

df['Genre'] = df['Genre'].apply(lambda x: x.split(', ') if ',' in x else [x.
 ↪strip()])

df = df.explode('Genre').reset_index(drop=True)

df = df[df['Genre'].str.strip() != '']

print(df.head())
```

```
   Release_Date                  Title  Popularity  Vote_Count Vote_Average  \
0          2021  Spider-Man: No Way Home    5083.954        8940      Popular
1          2021  Spider-Man: No Way Home    5083.954        8940      Popular
2          2021  Spider-Man: No Way Home    5083.954        8940      Popular
```

```
3          2022              The Batman    3827.658        1151      Popular
4          2022              The Batman    3827.658        1151      Popular

             Genre
0           Action
1        Adventure
2  Science Fiction
3            Crime
4          Mystery
```

[85]:
```
df
```

[85]:
```
       Release_Date                                Title  Popularity  \
0              2021          Spider-Man: No Way Home    5083.954
1              2021          Spider-Man: No Way Home    5083.954
2              2021          Spider-Man: No Way Home    5083.954
3              2022                       The Batman    3827.658
4              2022                       The Batman    3827.658
…               …                                …           …
25547          2021  The United States vs. Billie Holiday      13.354
25548          2021  The United States vs. Billie Holiday      13.354
25549          1984                          Threads      13.354
25550          1984                          Threads      13.354
25551          1984                          Threads      13.354

       Vote_Count Vote_Average            Genre
0            8940      Popular           Action
1            8940      Popular        Adventure
2            8940      Popular  Science Fiction
3            1151      Popular            Crime
4            1151      Popular          Mystery
…             …           …                …
25547         152         Good            Drama
25548         152         Good          History
25549         186      Popular              War
25550         186      Popular            Drama
25551         186      Popular  Science Fiction

[25552 rows x 6 columns]
```

[ ]:

casting Genre column into category

[ ]:

[86]:
```
df['Genre'] = df['Genre'].astype('category')
df['Genre'].dtypes
```

[86]:
```
CategoricalDtype(categories=['Action', 'Adventure', 'Animation', 'Comedy',
'Crime',
                  'Documentary', 'Drama', 'Family', 'Fantasy', 'History',
                  'Horror', 'Music', 'Mystery', 'Romance', 'Science Fiction',
                  'TV Movie', 'Thriller', 'War', 'Western'],
, ordered=False, categories_dtype=object)
```

[87]:
```
df.nunique()
```

[87]:
```
Release_Date     100
Title           9415
Popularity      8088
Vote_Count      3265
Vote_Average       4
Genre             19
dtype: int64
```

[88]:
```
df.head()
```

[88]:

	Release_Date	Title	Popularity	Vote_Count	Vote_Average	Genre
0	2021	Spider-Man: No Way Home	5083.954	8940	Popular	Action
1	2021	Spider-Man: No Way Home	5083.954	8940	Popular	Adventure
2	2021	Spider-Man: No Way Home	5083.954	8940	Popular	Science Fiction
3	2022	The Batman	3827.658	1151	Popular	Crime
4	2022	The Batman	3827.658	1151	Popular	Mystery

4 Data Visualization

[ ]:

[89]:
```
sns.set_style('whitegrid')
```

[ ]:

Q1 : What is the most frequent genre of movies released in netflix

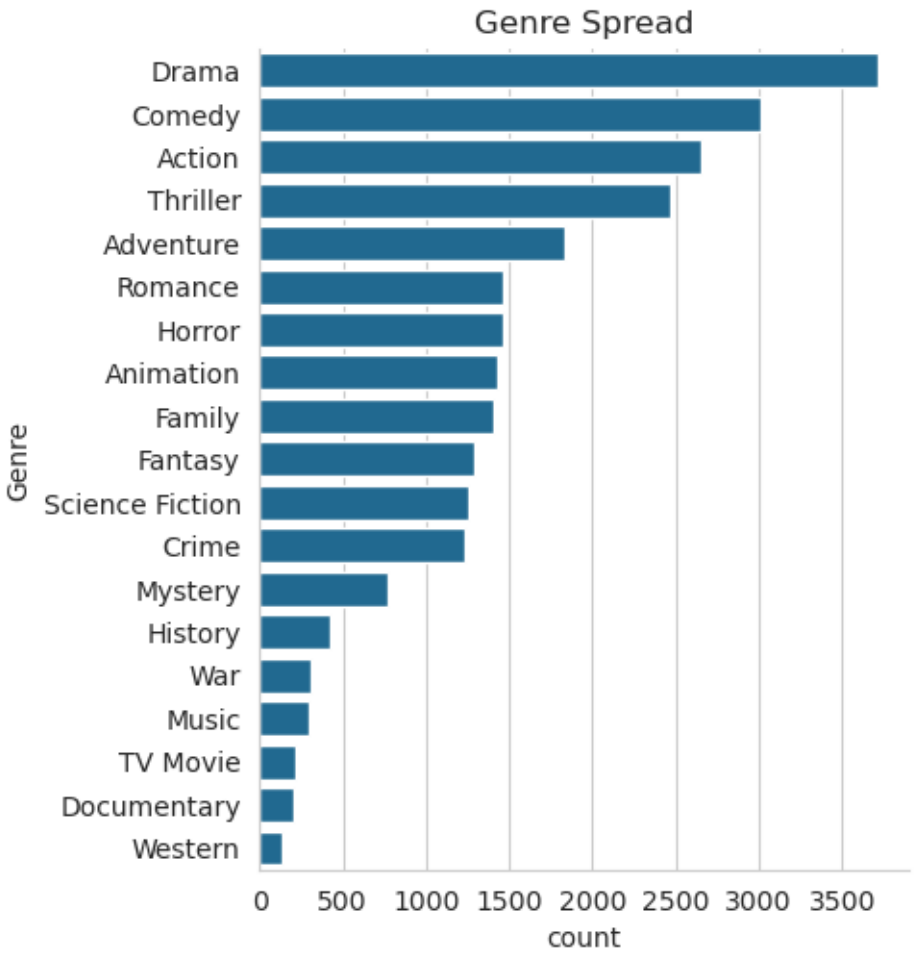
[ ]:

[90]:
```
df['Genre'].describe()
```

[90]:
```
count     25552
unique       19
top       Drama
freq       3715
Name: Genre, dtype: object
```

[94]:
```
sns.catplot(y= 'Genre' , data = df , kind = 'count' ,
            order= df ['Genre'].value_counts().index , color= '#0f6fa2' )
plt.title("Genre Spread")
plt.show()
```

```
/opt/conda/envs/anaconda-2024.02-py310/lib/python3.10/site-
packages/seaborn/categorical.py:641: FutureWarning: The default of
observed=False is deprecated and will be changed to True in a future version of
pandas. Pass observed=False to retain current behavior or observed=True to adopt
the future default and silence this warning.
  grouped_vals = vals.groupby(grouper)
/opt/conda/envs/anaconda-2024.02-py310/lib/python3.10/site-
packages/seaborn/categorical.py:641: FutureWarning: The default of
observed=False is deprecated and will be changed to True in a future version of
pandas. Pass observed=False to retain current behavior or observed=True to adopt
the future default and silence this warning.
  grouped_vals = vals.groupby(grouper)
```



[ ]:

Q2: Which has highest votes in vote avg column

[ ]:

[95]:
```
df.head()
```

[95]:

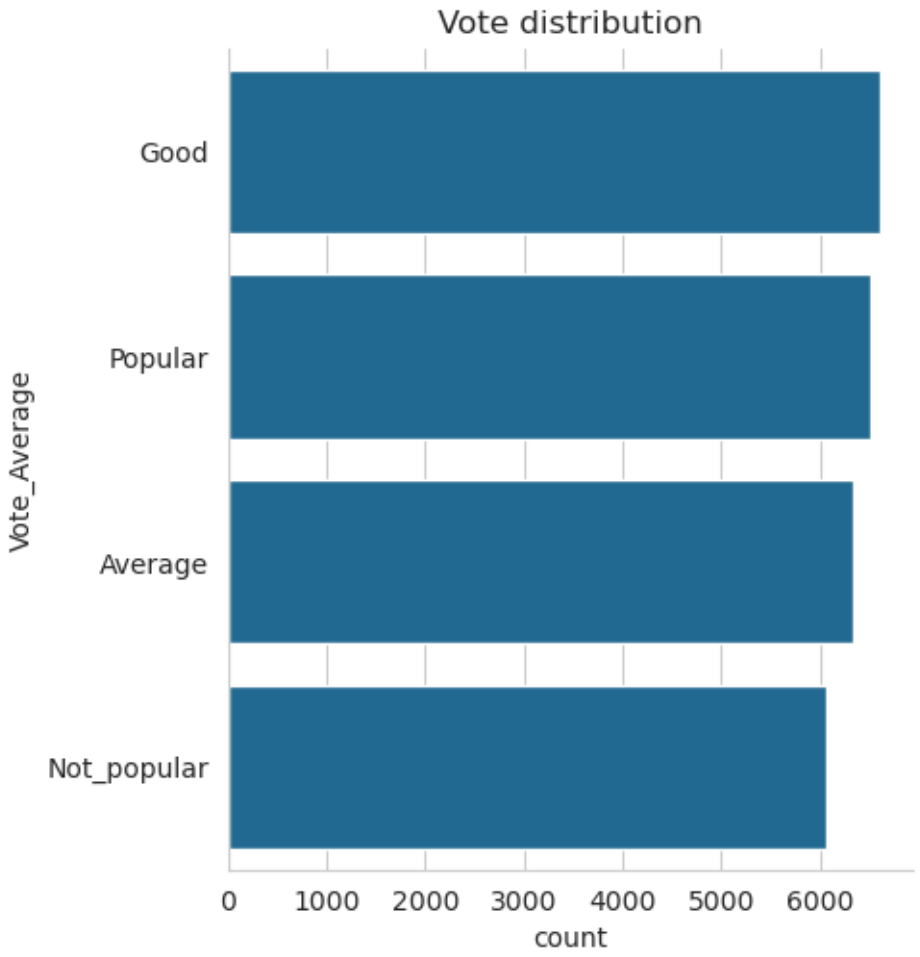
	Release_Date	Title	Popularity	Vote_Count	Vote_Average	\
0	2021	Spider-Man: No Way Home	5083.954	8940	Popular	
1	2021	Spider-Man: No Way Home	5083.954	8940	Popular	
2	2021	Spider-Man: No Way Home	5083.954	8940	Popular	
3	2022	The Batman	3827.658	1151	Popular	
4	2022	The Batman	3827.658	1151	Popular	

	Genre
0	Action

```
1        Adventure
2  Science Fiction
3            Crime
4          Mystery
```

[96]:
```
sns.catplot(y= 'Vote_Average' , data = df , kind = 'count' ,
            order= df ['Vote_Average'].value_counts().index , color= '#0f6fa2' )
plt.title("Vote distribution")
plt.show()
```

```
/opt/conda/envs/anaconda-2024.02-py310/lib/python3.10/site-
packages/seaborn/categorical.py:641: FutureWarning: The default of
observed=False is deprecated and will be changed to True in a future version of
pandas. Pass observed=False to retain current behavior or observed=True to adopt
the future default and silence this warning.
  grouped_vals = vals.groupby(grouper)
/opt/conda/envs/anaconda-2024.02-py310/lib/python3.10/site-
packages/seaborn/categorical.py:641: FutureWarning: The default of
observed=False is deprecated and will be changed to True in a future version of
pandas. Pass observed=False to retain current behavior or observed=True to adopt
the future default and silence this warning.
  grouped_vals = vals.groupby(grouper)
```



[ ]:

Q3: which movie got the highest popularity? what is its genre?

[ ]:

[97]:
```
df.head()
```

[97]:

	Release_Date	Title	Popularity	Vote_Count	Vote_Average	Genre
0	2021	Spider-Man: No Way Home	5083.954	8940	Popular	Action
1	2021	Spider-Man: No Way Home	5083.954	8940	Popular	
2	2021	Spider-Man: No Way Home	5083.954	8940	Popular	
3	2022	The Batman	3827.658	1151	Popular	
4	2022	The Batman	3827.658	1151	Popular	

```
1        Adventure
2  Science Fiction
3            Crime
4          Mystery
```

[98]:
```
df[df['Popularity'] == df['Popularity'].max()]
```

[98]:
```
   Release_Date                    Title  Popularity  Vote_Count Vote_Average  \
0          2021  Spider-Man: No Way Home    5083.954        8940      Popular
1          2021  Spider-Man: No Way Home    5083.954        8940      Popular
2          2021  Spider-Man: No Way Home    5083.954        8940      Popular

             Genre
0           Action
1        Adventure
2  Science Fiction
```

[ ]:

Q4: which movie got the lowest popularity? what is its genre?

[ ]:

[99]:
```
df[df['Popularity'] == df['Popularity'].min()]
```

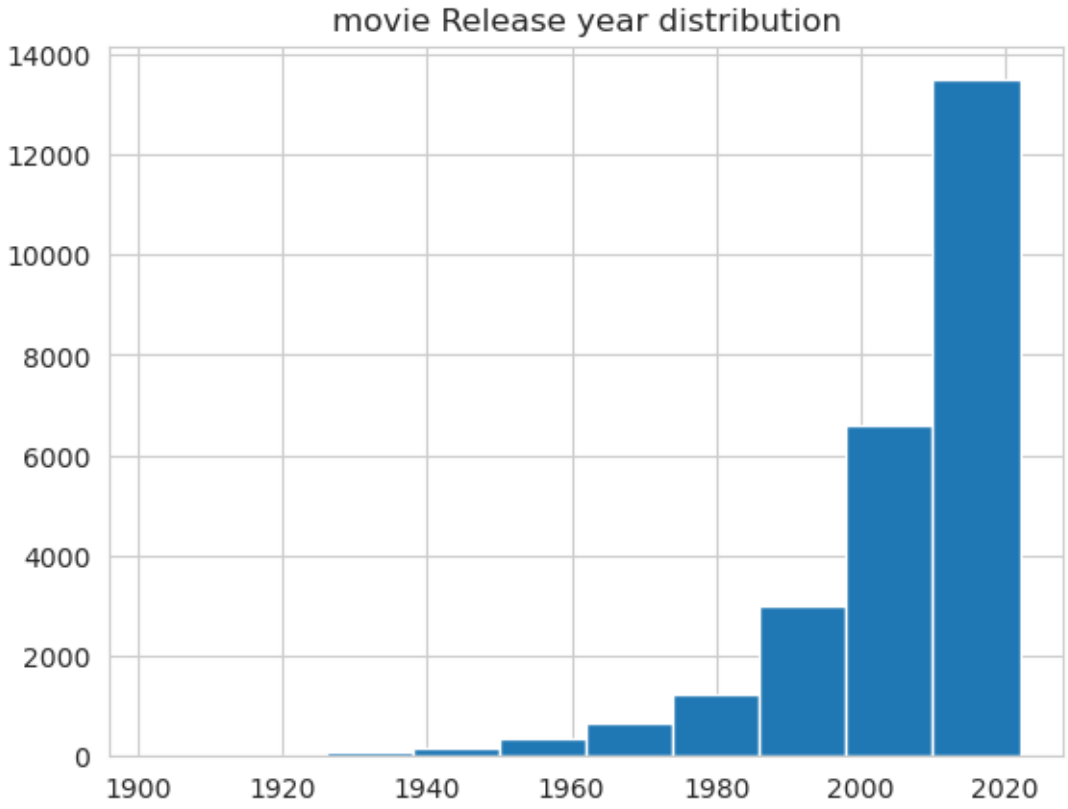
[99]:
```
       Release_Date                                Title  Popularity  \
25546          2021  The United States vs. Billie Holiday      13.354
25547          2021  The United States vs. Billie Holiday      13.354
25548          2021  The United States vs. Billie Holiday      13.354
25549          1984                               Threads      13.354
25550          1984                               Threads      13.354
25551          1984                               Threads      13.354

       Vote_Count Vote_Average            Genre
25546         152         Good            Music
25547         152         Good            Drama
25548         152         Good          History
25549         186      Popular              War
25550         186      Popular            Drama
25551         186      Popular  Science Fiction
```

[ ]:

Q5: which year has the most films

[ ]:

[100]:
```
df['Release_Date'].hist()
plt.title("movie Release year distribution")
plt.show()
```



[ ]:
```

```

5 Conclusion

Q1: What is the most frequent genre in the dataset?

Drama genre is the most frequent genre in our dataset and has appeared more than 14% of the times among 19 other genres.

Q2: What genres has highest votes ?

we have 25.5% of our dataset with popular vote (6520 rows). Drama again gets the

highest popularity among fans by being having more than 18.5% of movies popularities.

Q3: What movie got the highest popularity ? what's its genre ?

Spider-Man: No Way Home has the highest popularity rate in our dataset and it has genres of Action , Adventure and Sience Fiction .

Q3: What movie got the lowest popularity ? what's its genre ?

The united states, thread' has the highest lowest rate in our dataset and it has genres of music , drama , 'war', 'sci-fi' and history'.

Q4: Which year has the most filmmed movies?

year 2020 has the highest filmming rate in our dataset.

[ ]:

[ ]:

[ ]:

[ ]:

[ ]:

[ ]: