NETFLIX DATA ANALYSIS

PRESENTED BY

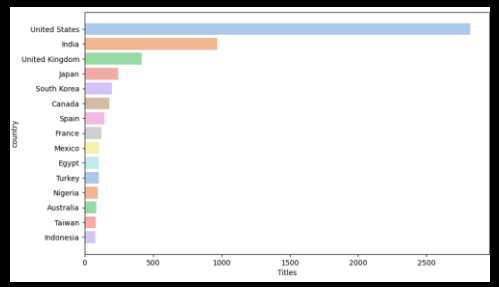
Aayush Choudhary

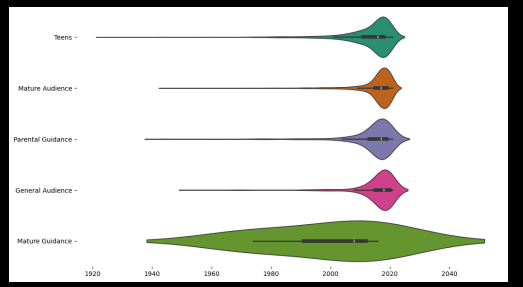
The best Group

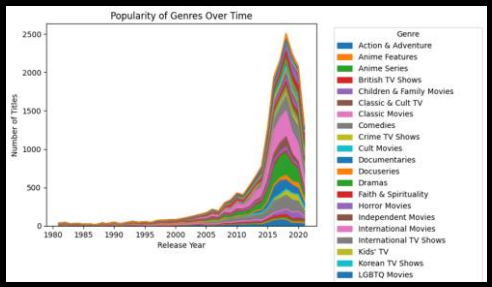
INTRODUCTION

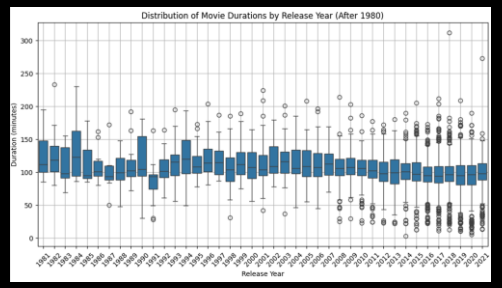
97% for you | 20+ | 2024

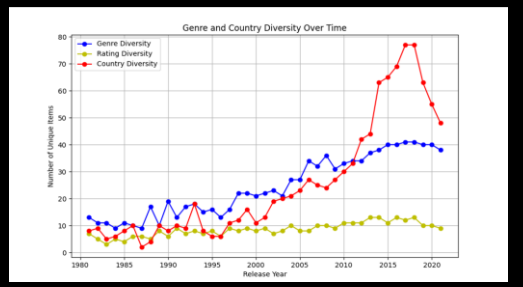
This project aims to analyze various aspects of Netflix's extensive library, including the growth of movies and TV shows, genre distribution, country of origin, content duration, rating categories, release patterns, director and cast involvement, genre popularity over time, and much more.

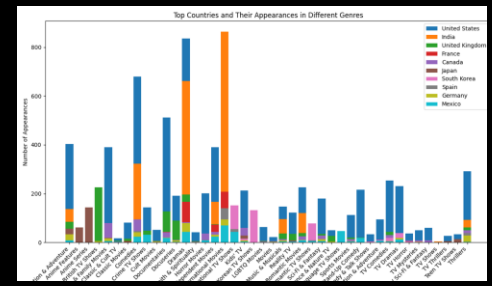


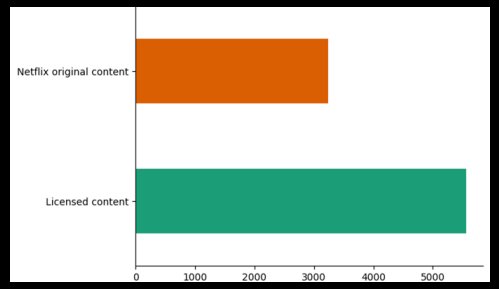


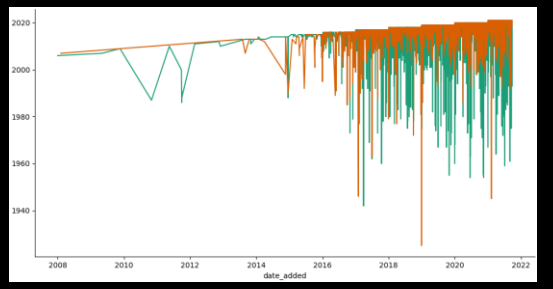


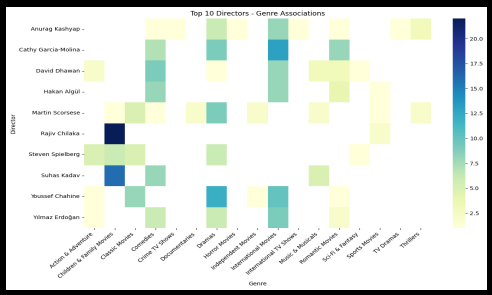












Skip

LOADING DATA .....

Libraries and Dataset Used for this project

97% for you | 20+ | 2024

List of Libraries

1. Pandas
2. NumPy
3. Matplotlib
4. Seaborn
5. Statistics
6. Itertools
7. Worldcloud

```
import numpy as np
import pandas as pd
import statistics as st
import matplotlib.pyplot as plt
import seaborn as sns
```

```
[4] df = pd.read_csv('/content/netflix_titles_2021 - netflix_titles_2021.csv')
```

```
[5] df.head()
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	Documentaries	As her father nears the end of his life, filmm...
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	International TV Shows, TV Dramas, TV Mysteries	After crossing paths at a party, a Cape Town t...
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA	1 Season	Crime TV Shows, International TV Shows, TV Act...	To protect his family from a powerful drug lor...
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA	1 Season	Docuseries, Reality TV	Feuds, flirtations and toilet talk go down amo...
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	2 Seasons	International TV Shows, Romantic TV Shows, TV ...	In a city of coaching centers known to train I...

Skip

DATA UNDERSTANDING

- Number of rows and columns: [8807,9]

Data Dictionary:

Column name	Type	Description
show_id	object	A unique identifier for each title.
type	object	The category of the title, which is either 'Movie' or 'TV Show'.
title	object	The name of the movie or TV show.
director	object	The director(s) of the movie or TV show.
cast	object	The list of main actors/actresses in the title.
country	object	The country or countries where the movie or TV show was produced.
date_added	object	The date the title was added to Netflix.
release_year	datetime64[ns]	The year the movie or TV show was originally released.
rating	object	The age rating of the title.
duration	int	The duration of the title, in minutes for movies and seasons for TV shows.
listed_in	object	The genres the title falls under.
description	object	A brief summary of the title.

Main Slide

DATA CLEANING

```
[29] def convert_to_list(string):
       lst = []
       lst = string.split(', ')
       return lst
```

```
[31] df.fillna({'rating':'unknown','cast':'unknown', 'country': 'unknown', 'director':'unknown'}, inplace=True)
     df.isna().sum()
```

```
[32] df['cast'] = df['cast'].apply(convert_to_list)
     df['director'] = df['director'].apply(convert_to_list)
     df['country'] = df['country'].apply(convert_to_list)
     df['listed_in'] = df['listed_in'].apply(convert_to_list)
     df['rating'] = df['rating'].replace('unkown', 'unknown')
     df['title'] = df['title'].str.lower()
     df['description'] = df['description'].str.lower()
     df
```

```
[18] df['date_added'] = pd.to_datetime(df['date_added'])
```

```
df_error = df[df['date_added'].dt.year < df['release_year']]
df.drop(df_error.index, inplace=True)
```

```
[21] df[df.director == 'Louis C.K.'].head()
```

```
[22] df.loc[df['director'] == 'Louis C.K.' , 'duration'] = df['rating']
     df[df.director == 'Louis C.K.'].head()
```

```
[23] df.loc[df['director'] == 'Louis C.K.' , 'rating'] = 'unknown'
     df[df.director == 'Louis C.K.'].head()
```

```
[24] new_df = df[df['type'] == 'Movie']
```

```
rating_to_audience_mapping = {
    'PG-13': 'Teens',
    'TV-MA': 'Mature Audience',
    'PG': 'Teens',
    'TV-14': 'Teens',
    'TV-PG': 'Parental Guidance',
    'TV-Y': 'General Audience',
    'TV-Y7': 'Teens',
    'R': 'Mature Audience',
    'TV-G': 'General Audience',
    'G': 'General Audience',
    'NC-17': 'Mature Audience',
    'unknown': 'Parental Guidance',
    'NR': 'Mature Audience',
    'TV-Y7-FV': 'Teens',
    'UR': 'Mature Guidance'
}

# Add a new column 'audience_category' based on the mapping
df['audience_category'] = df['rating'].map(rating_to_audience_mapping)
```

Skip

DATAFRAME FOR MOVIE AND TV SHOW

```
[24] new_df = df[df['type'] == 'Movie']
```

```
[25] new_df['duration'] = new_df['duration'].str.replace(' min', '').astype(int)
     total_minutes = new_df['duration'].sum()
     print(total_minutes)
```

	date_added	release_year	duration
count	6129	6129.000000	6129.000000
mean	2019-05-07 06:56:47.929515520	2013.119759	99.568935
min	2008-01-01 00:00:00	1942.000000	3.000000
25%	2018-04-01 00:00:00	2012.000000	87.000000
50%	2019-06-20 00:00:00	2016.000000	98.000000
75%	2020-07-24 00:00:00	2018.000000	114.000000
max	2021-09-25 00:00:00	2021.000000	312.000000
std	NaN	9.679256	28.293268

```
[26] new_df_TV = df[df['type'] == 'TV Show']
```

```
[27] new_df_TV['duration'] = new_df_TV['duration'].str.replace(' Seasons?$', '', regex=True).astype(int)
     total_minutes = new_df_TV['duration'].sum()
     print(total_minutes)
```

	date_added	release_year	duration
count	2654	2664.000000	2664.000000
mean	2019-06-10 13:43:05.380557568	2016.593468	1.760886
min	2008-02-04 00:00:00	1925.000000	1.000000
25%	2018-04-21 18:00:00	2016.000000	1.000000
50%	2019-08-15 12:00:00	2018.000000	1.000000
75%	2020-10-01 00:00:00	2020.000000	2.000000
max	2021-09-24 00:00:00	2021.000000	17.000000
std	NaN	5.749193	1.580804

BEFORE

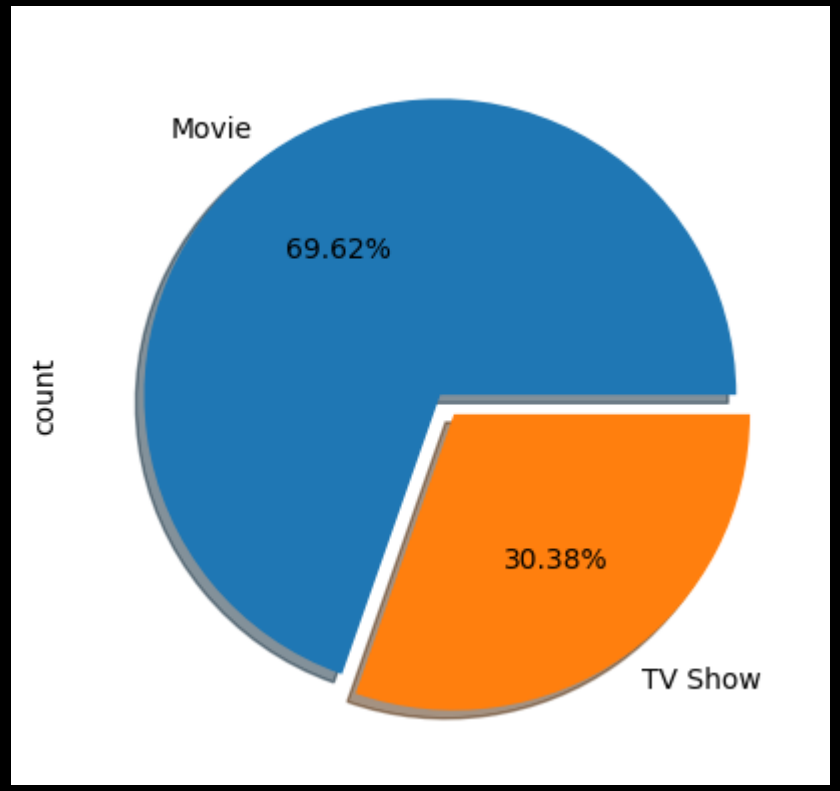
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8807 entries, 0 to 8806
Data columns (total 12 columns):
 #   Column        Non-Null Count  Dtype 
---  ------        --------------  ----- 
 0   show_id       8807 non-null   object
 1   type          8807 non-null   object
 2   title         8807 non-null   object
 3   director      6173 non-null   object
 4   cast          7982 non-null   object
 5   country       7976 non-null   object
 6   date_added    8797 non-null   object
 7   release_year  8807 non-null   int64 
 8   rating        8803 non-null   object
 9   duration      8804 non-null   object
 10  listed_in     8807 non-null   object
 11  description   8807 non-null   object
dtypes: int64(1), object(11)
memory usage: 825.8+ KB
```

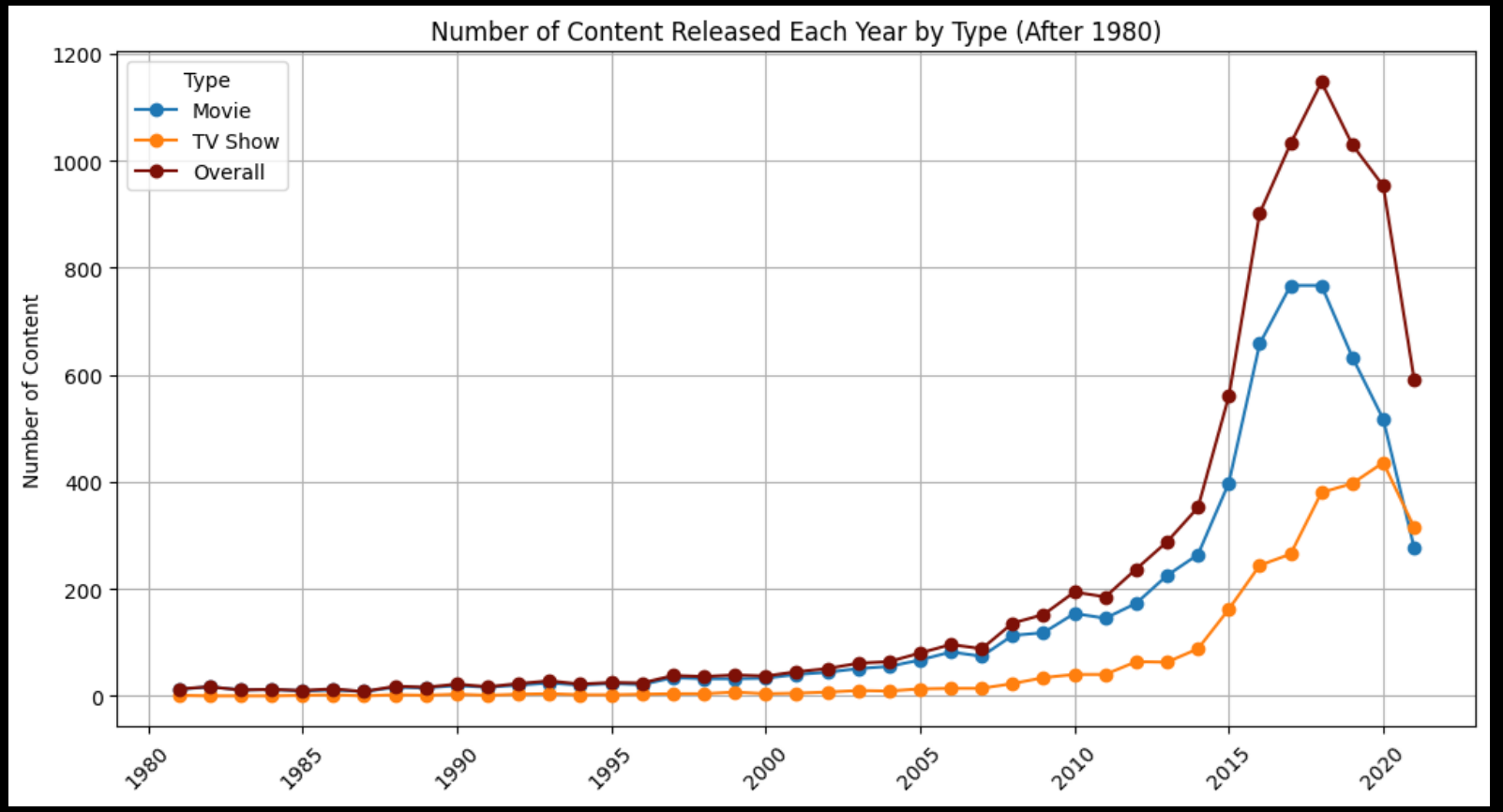
AFTER

```
<class 'pandas.core.frame.DataFrame'>
Index: 8793 entries, 0 to 8806
Data columns (total 13 columns):
 #   Column             Non-Null Count  Dtype         
---  ------             --------------  -----         
 0   show_id            8793 non-null   object        
 1   type               8793 non-null   object        
 2   title              8793 non-null   object        
 3   director           8793 non-null   object        
 4   cast               8793 non-null   object        
 5   country            8793 non-null   object        
 6   date_added         8783 non-null   datetime64[ns]
 7   release_year       8793 non-null   int64         
 8   rating             8793 non-null   object        
 9   duration           8793 non-null   object        
 10  listed_in          8793 non-null   object        
 11  description        8793 non-null   object        
 12  audience_category  8786 non-null   object        
dtypes: datetime64[ns](1), int64(1), object(11)
memory usage: 1.2+ MB
```

MOVIES AND TV SHOWS DISTRIBUTION

	Movie	TV Show	Total
Count	6131	2676	8807

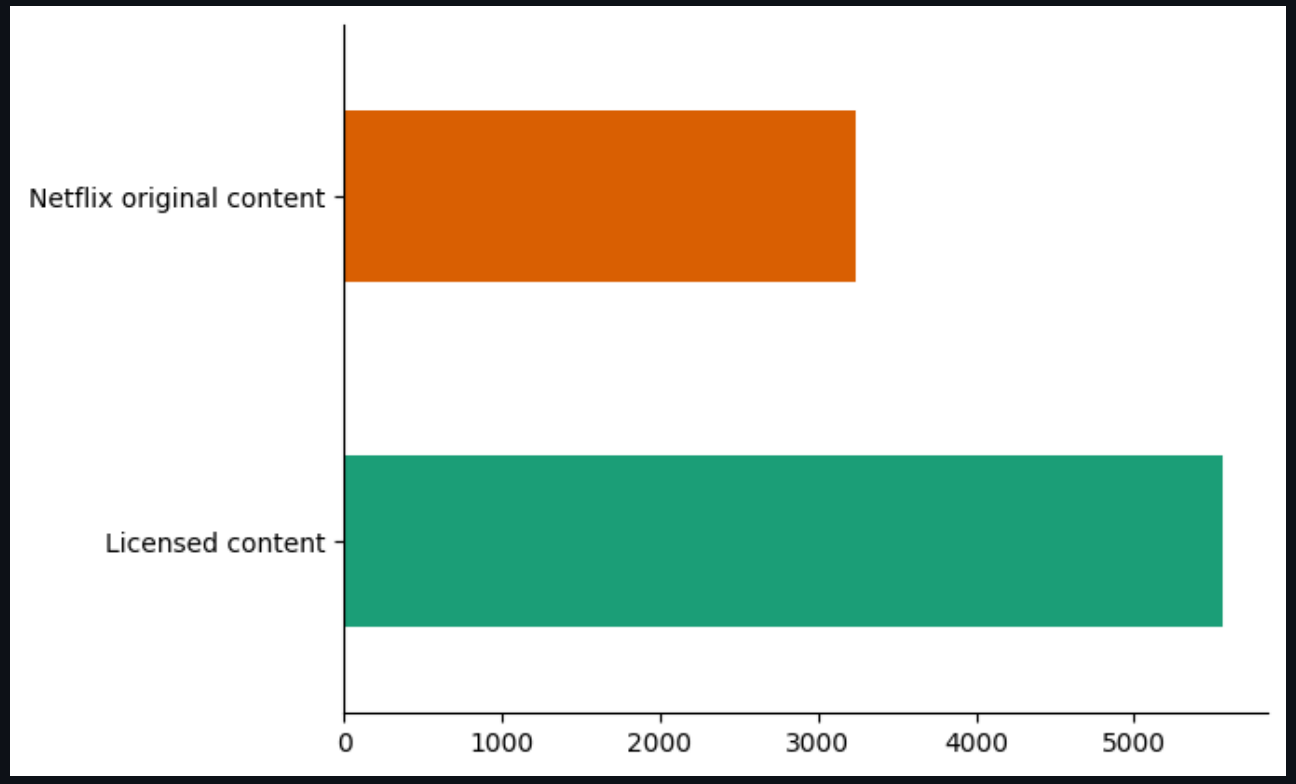




Skip

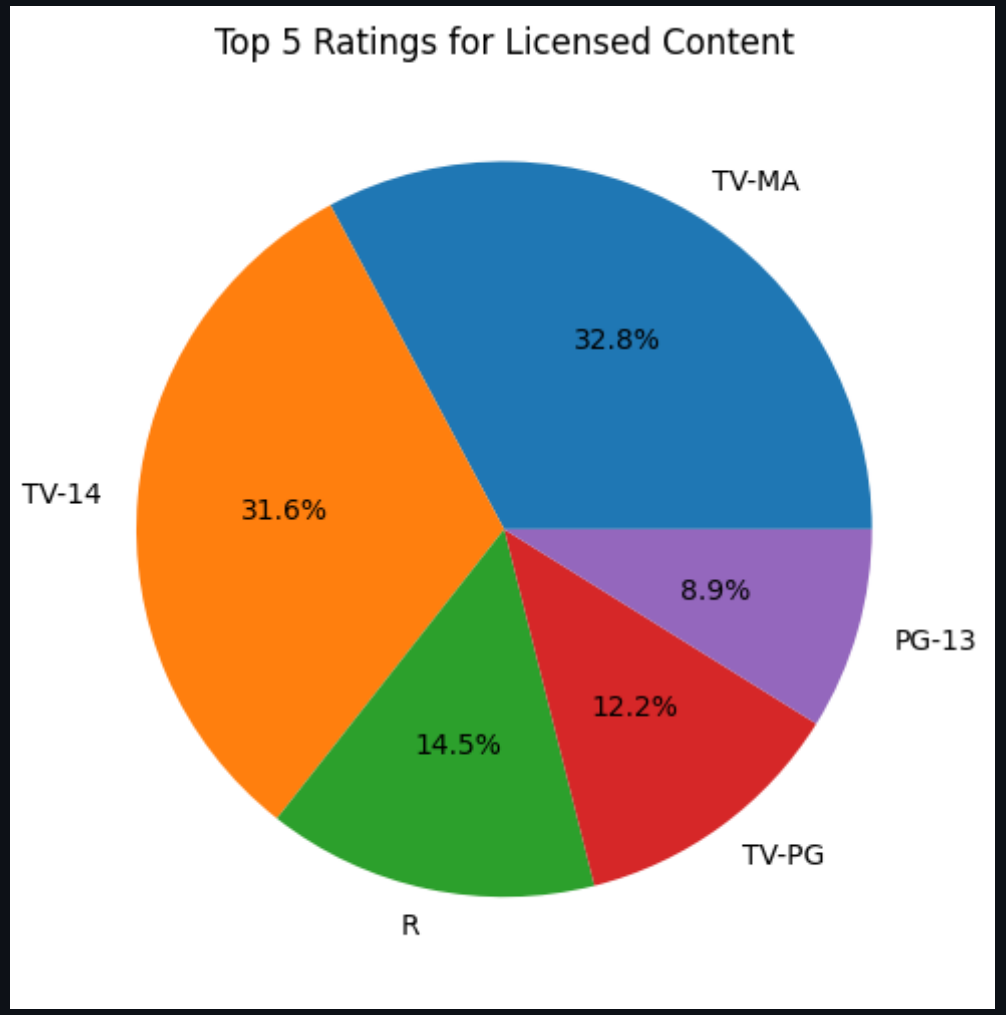
NETFLIX ORIGINAL CONTENT

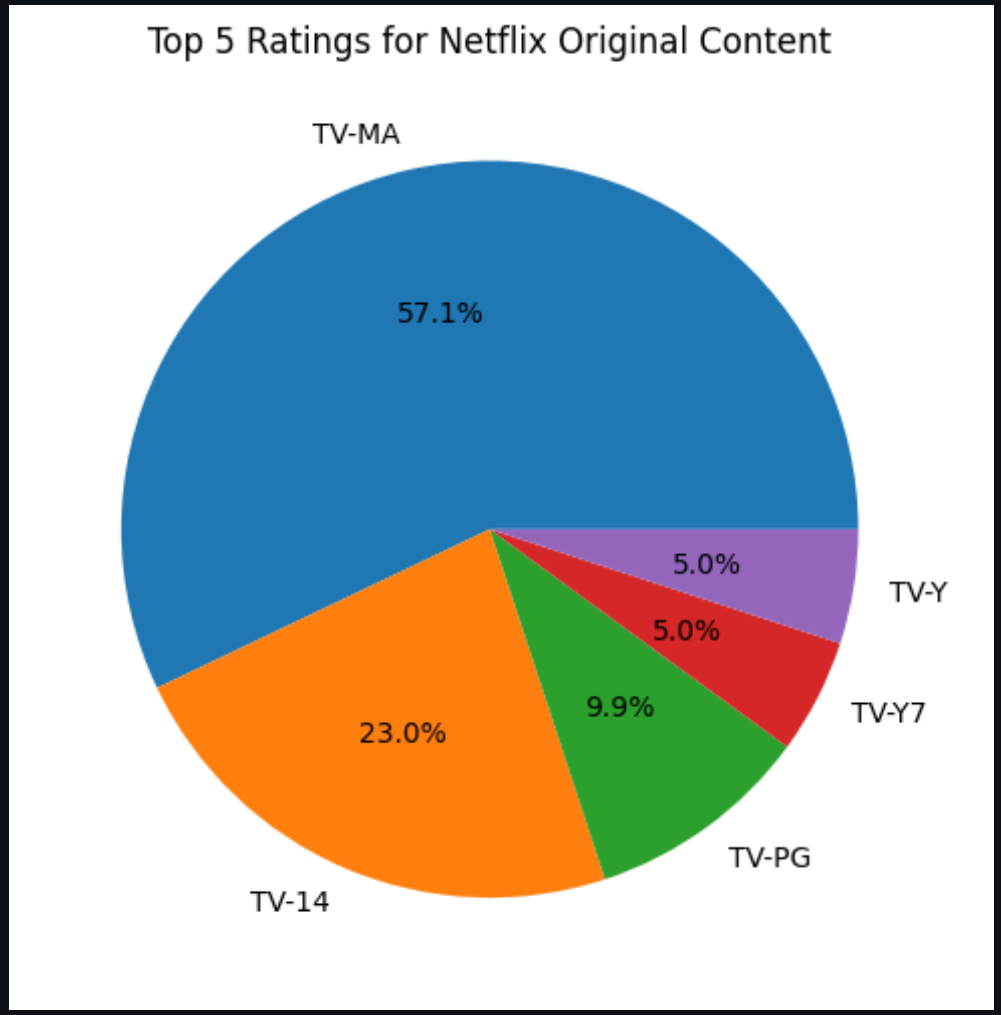
```
[899] df_content = df.copy()
      content_mapping = {
          True : 'Netflix original content',
          False : 'Licensed content'
      }
      df_content['content_type'] = df_content.apply(lambda x: content_mapping[x['date_added'].year == x['release_year']], axis=1)
```



Skip

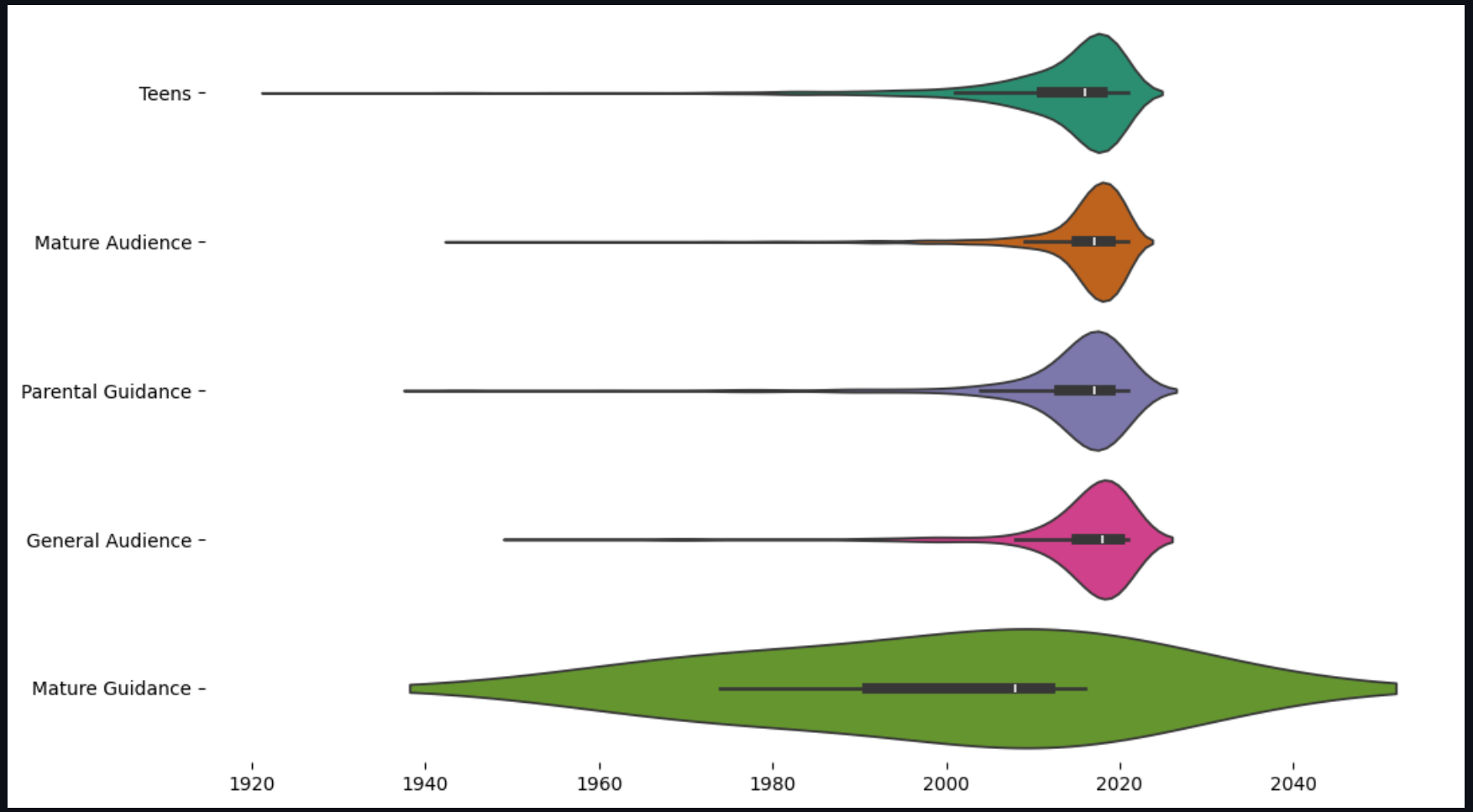
AUDIENCE TYPE





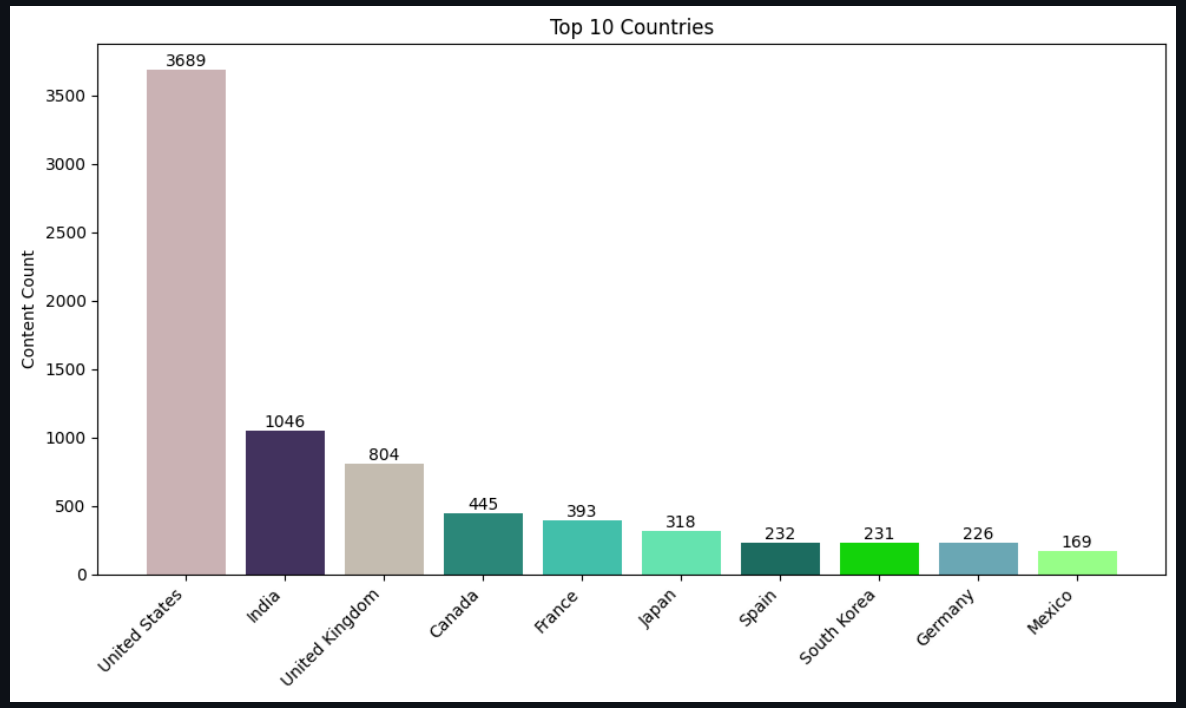
Skip

AUDIENCE TYPE DISTRIBUTION

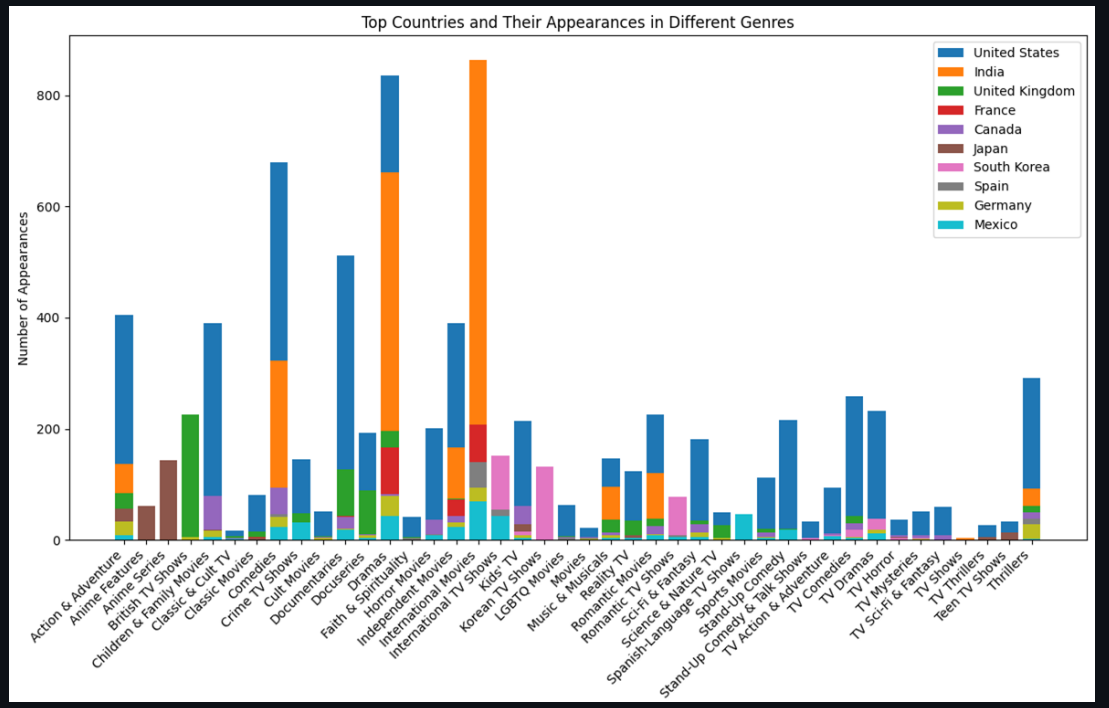


Skip

COUNTRY DISTRIBUTION

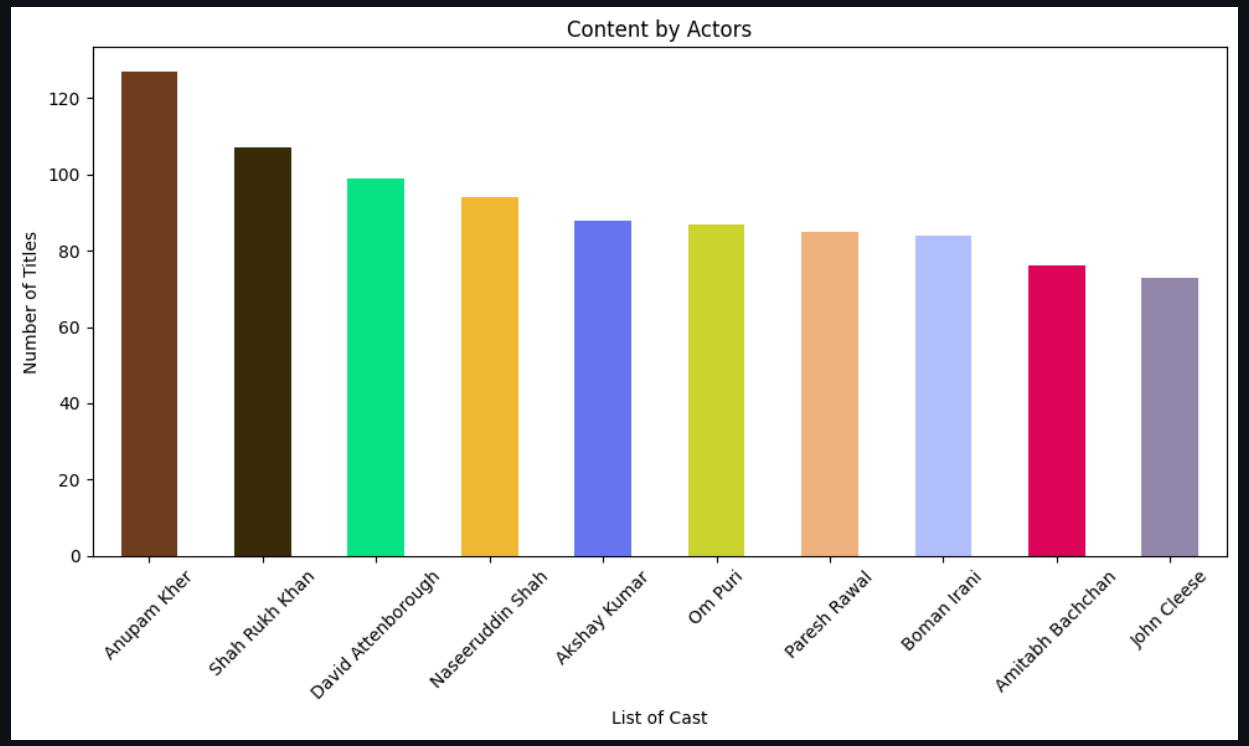


COUNTRY VS GENRE

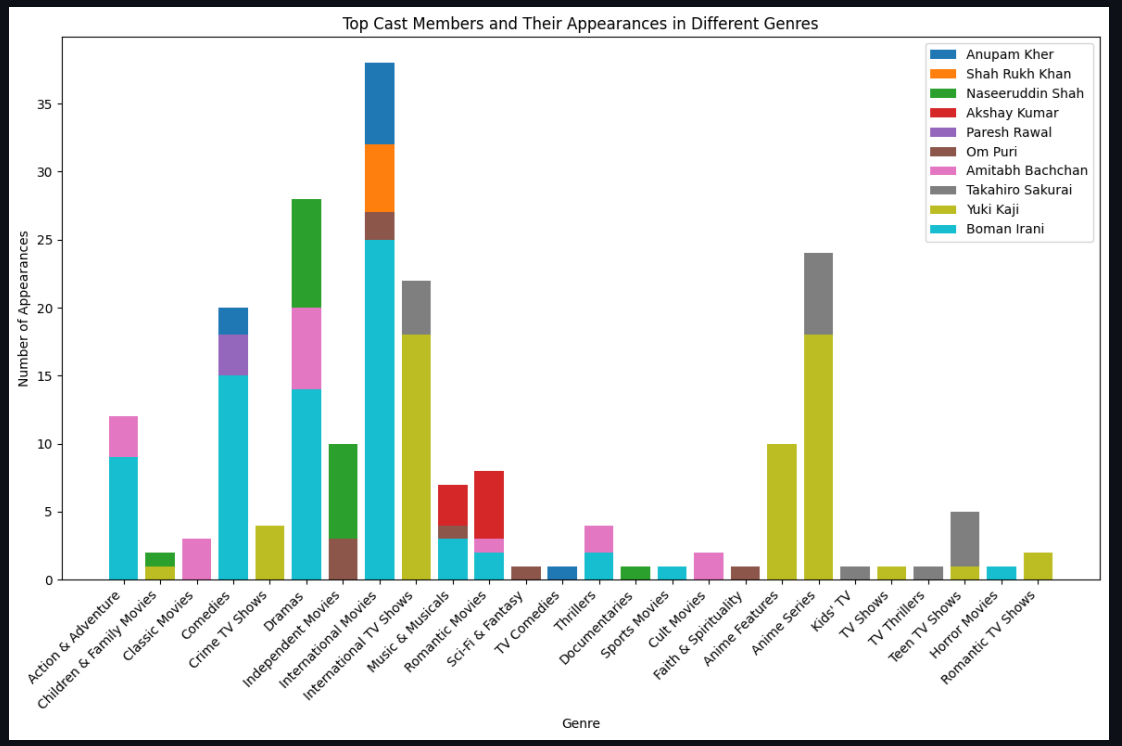


Skip

MOST POPULAR ACTORS

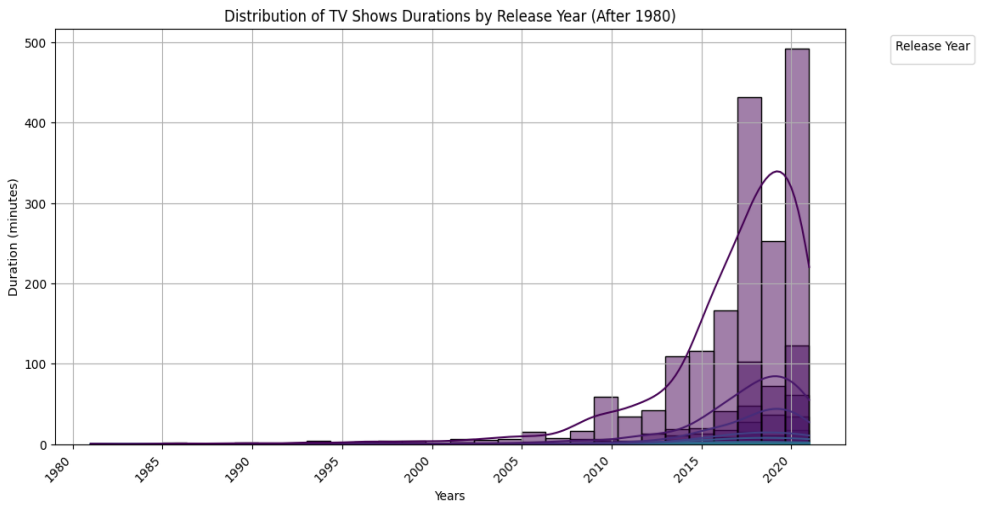


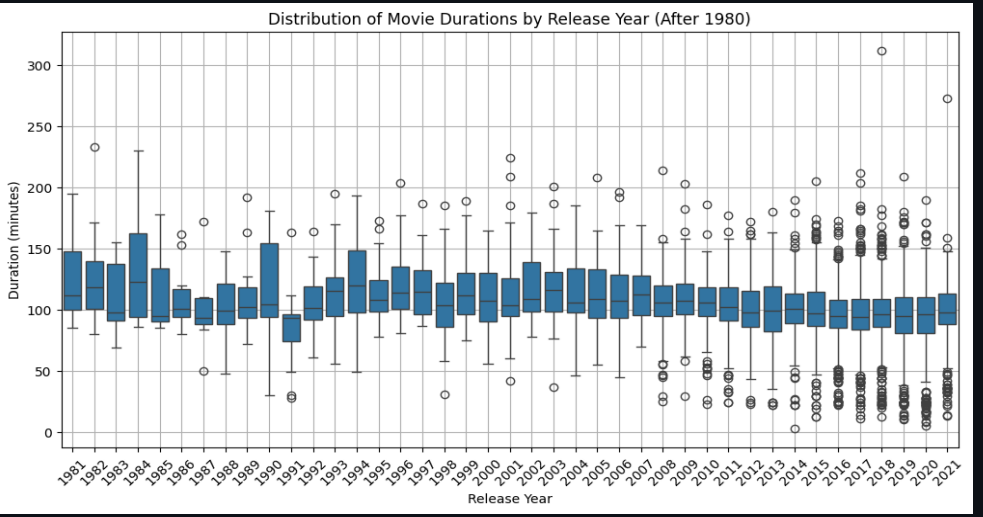
CAST VS GENRE

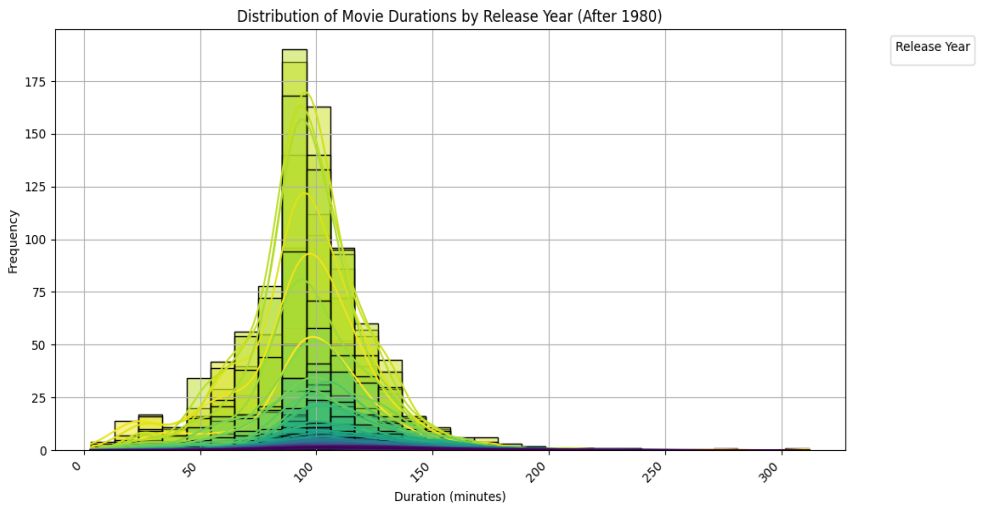


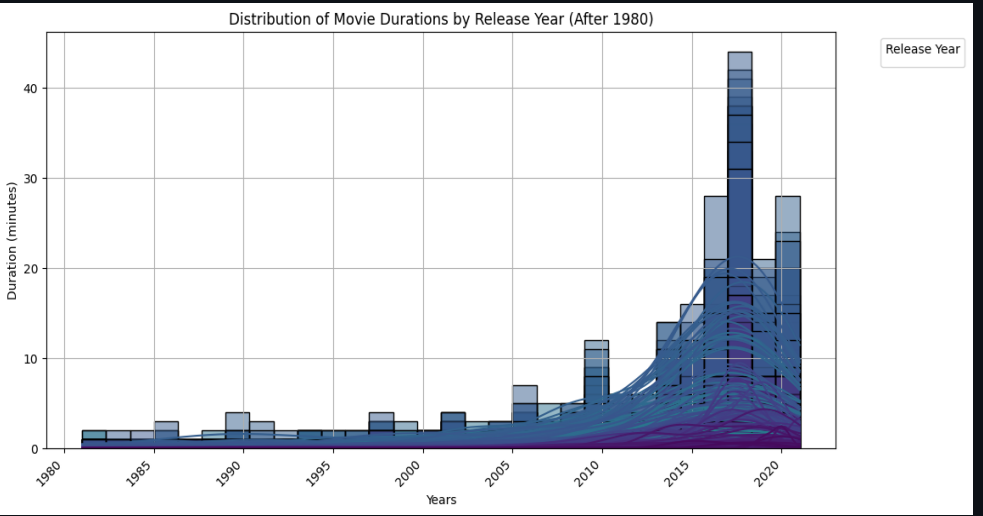
Skip

FINDING OUTLIERS



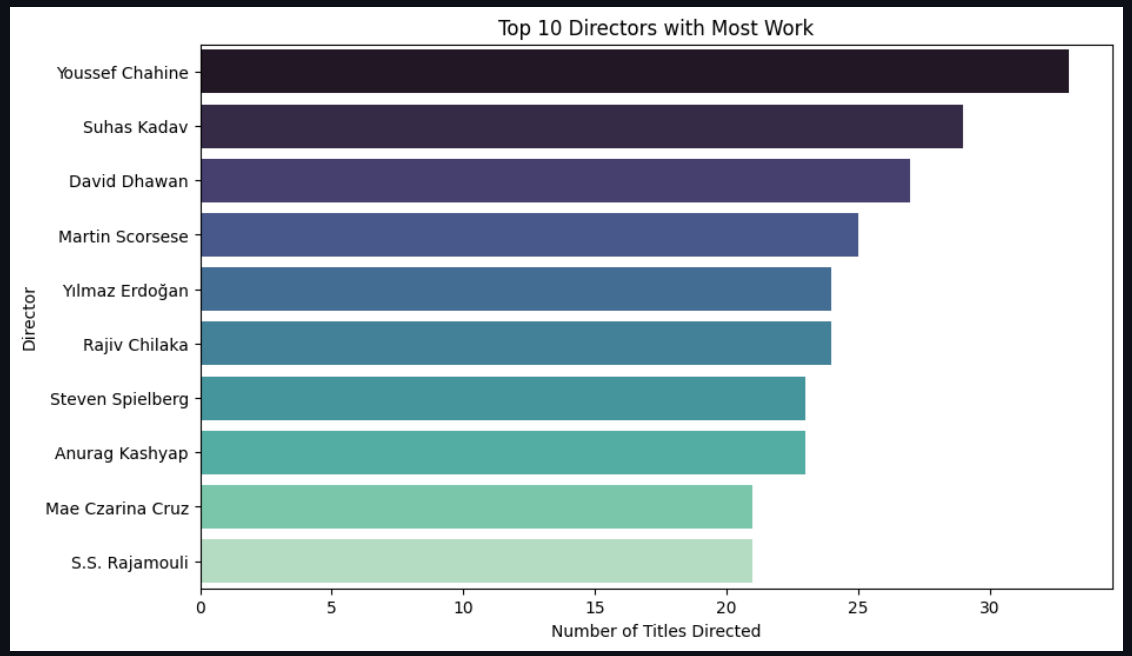


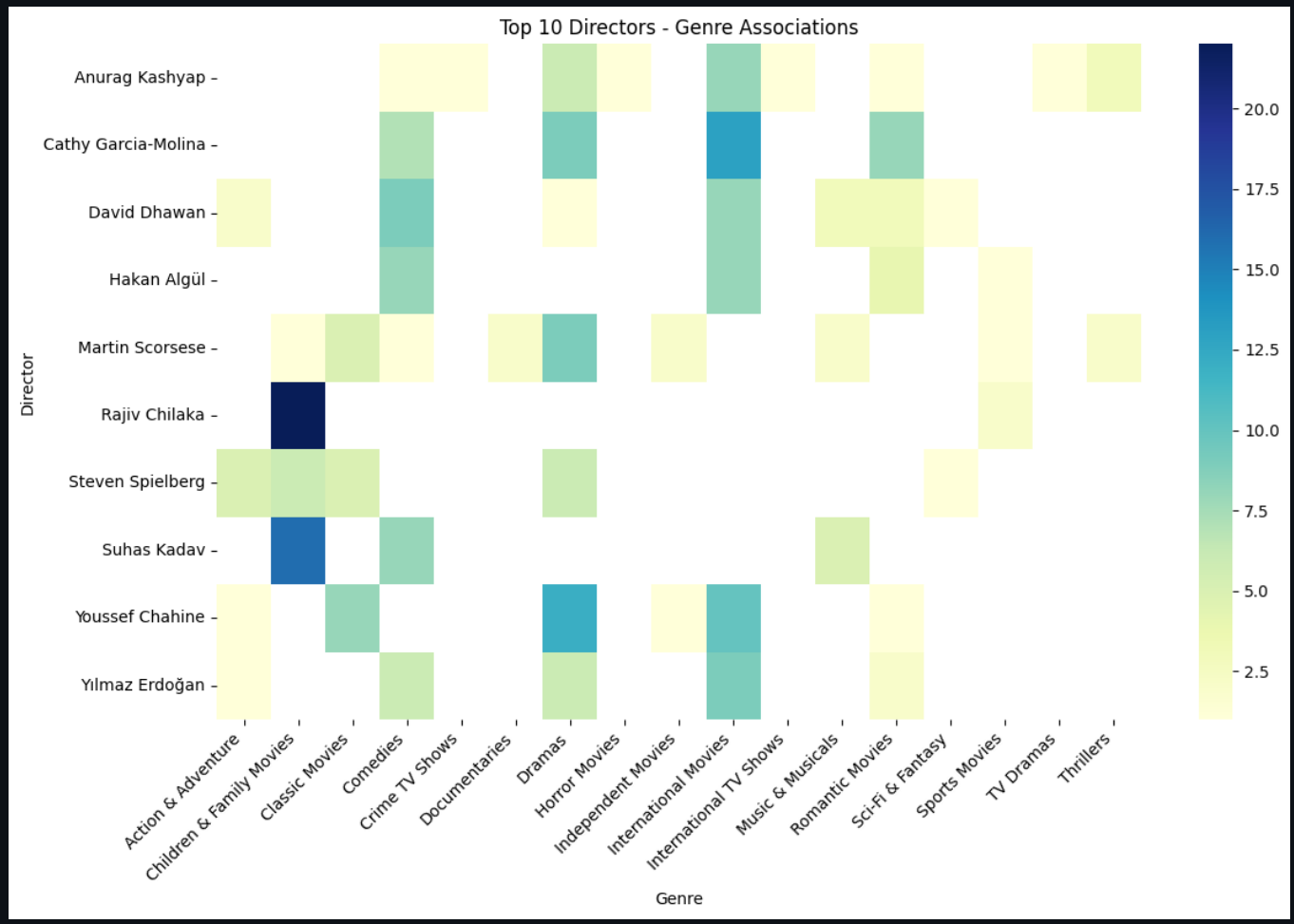




Skip

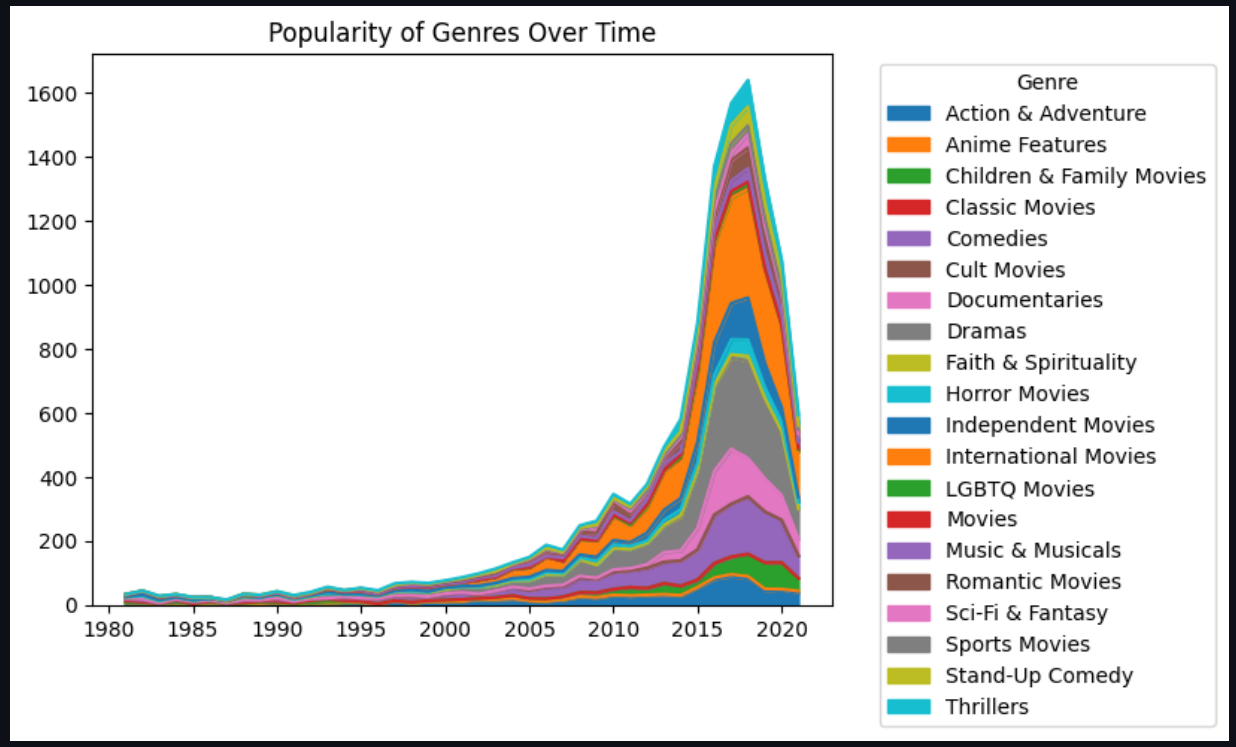
FAMOUS DIRECTORS

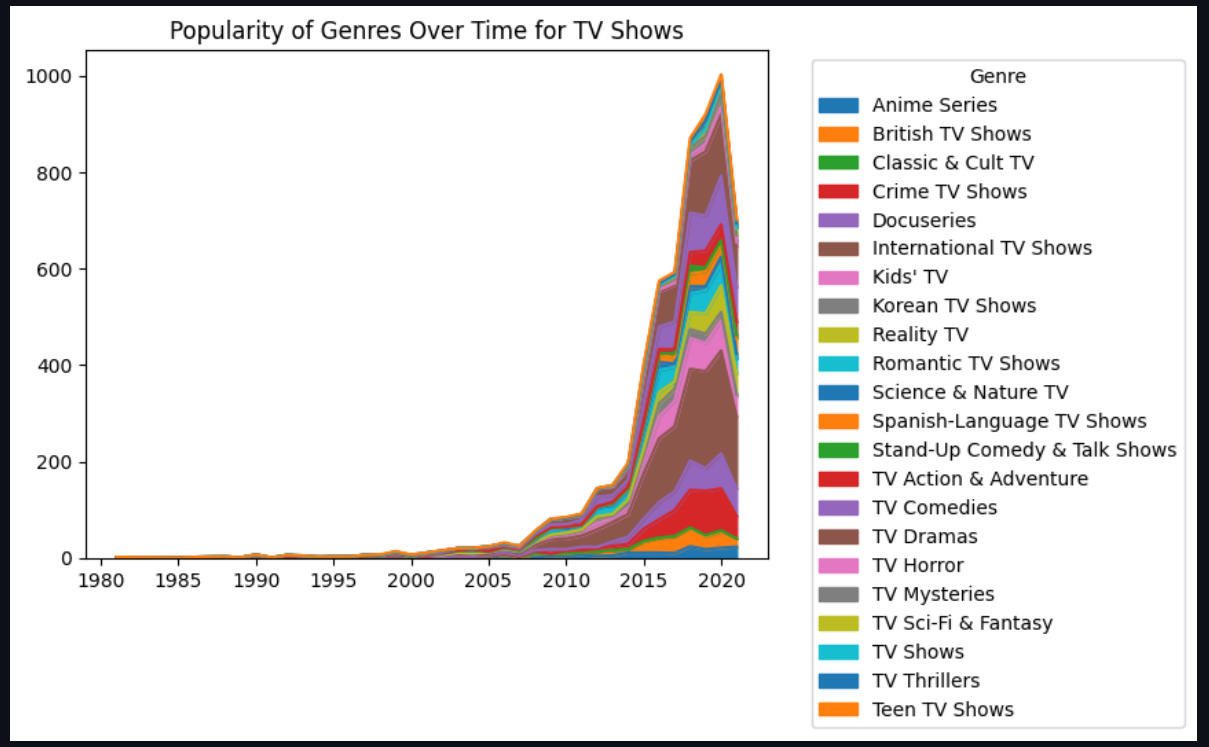




Skip

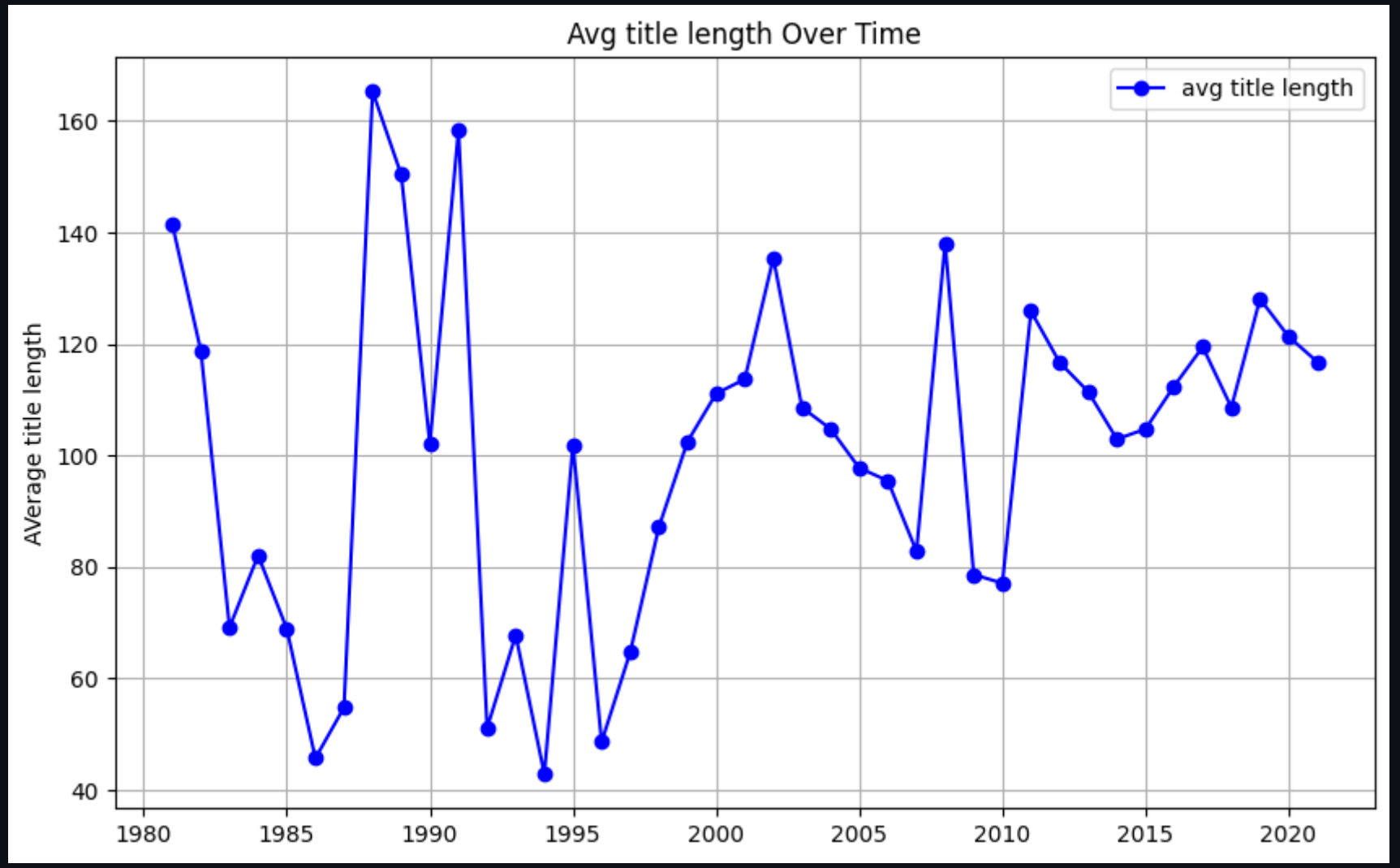
DIFFERENT GENRE CONTENT OVER TIME



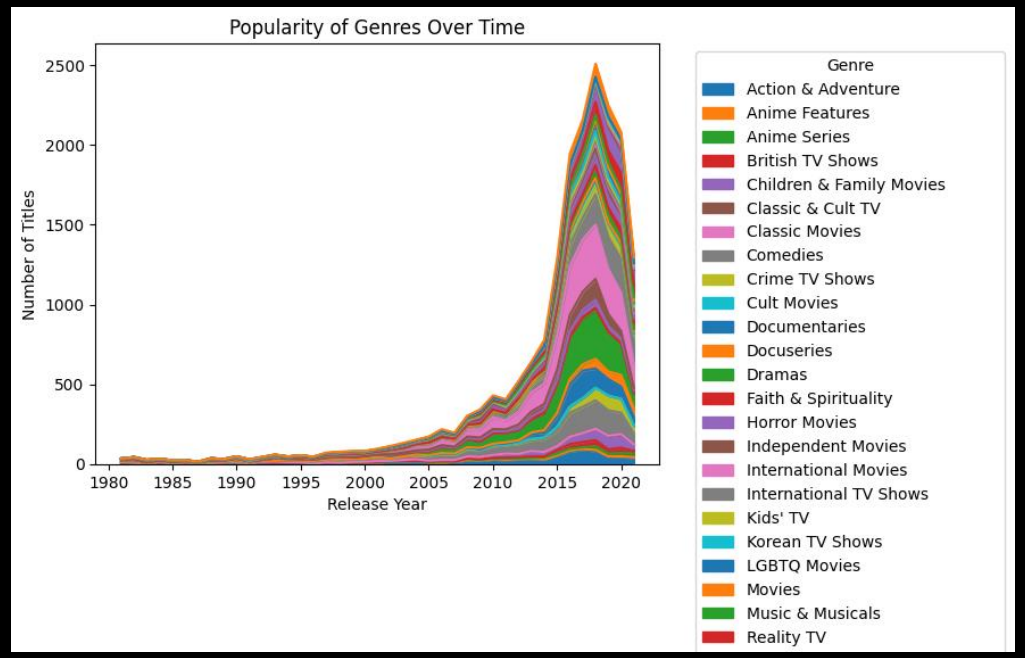


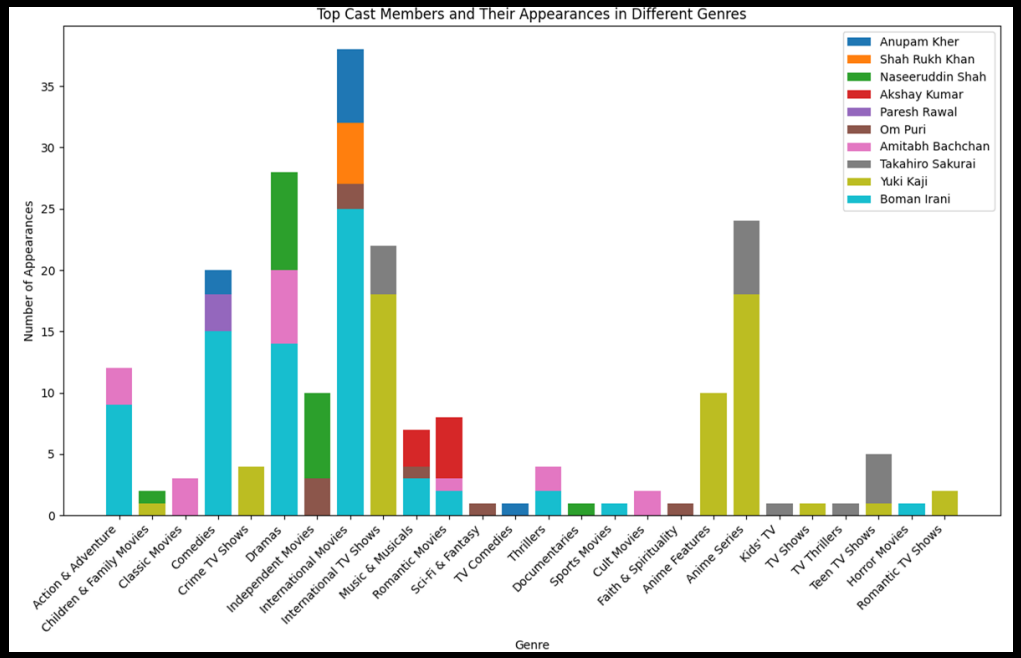
Skip

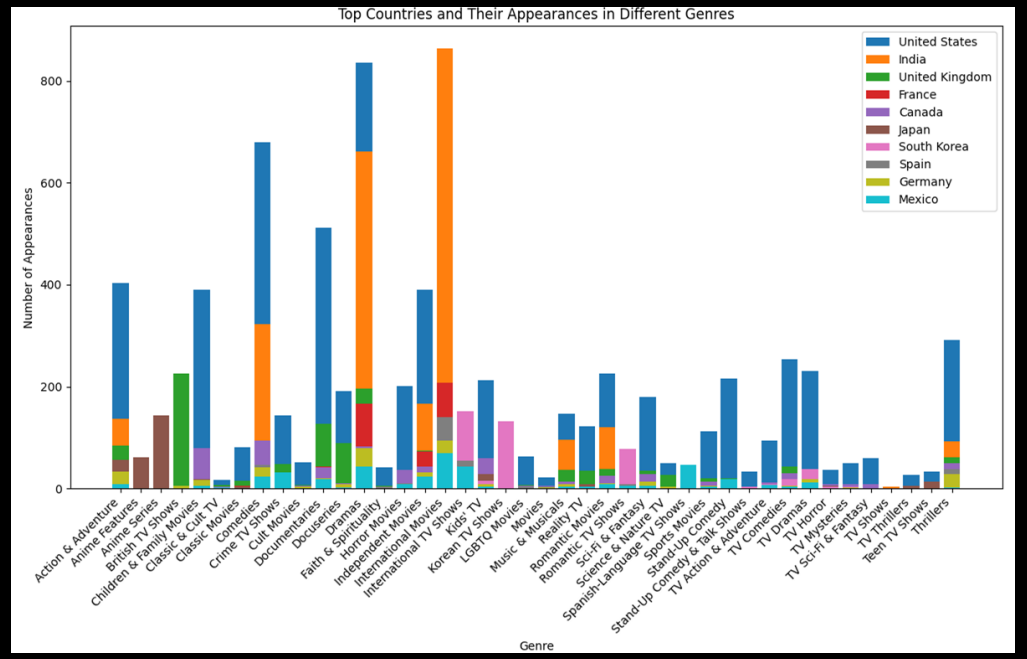
TITLE DIVERSITY OVER TIME

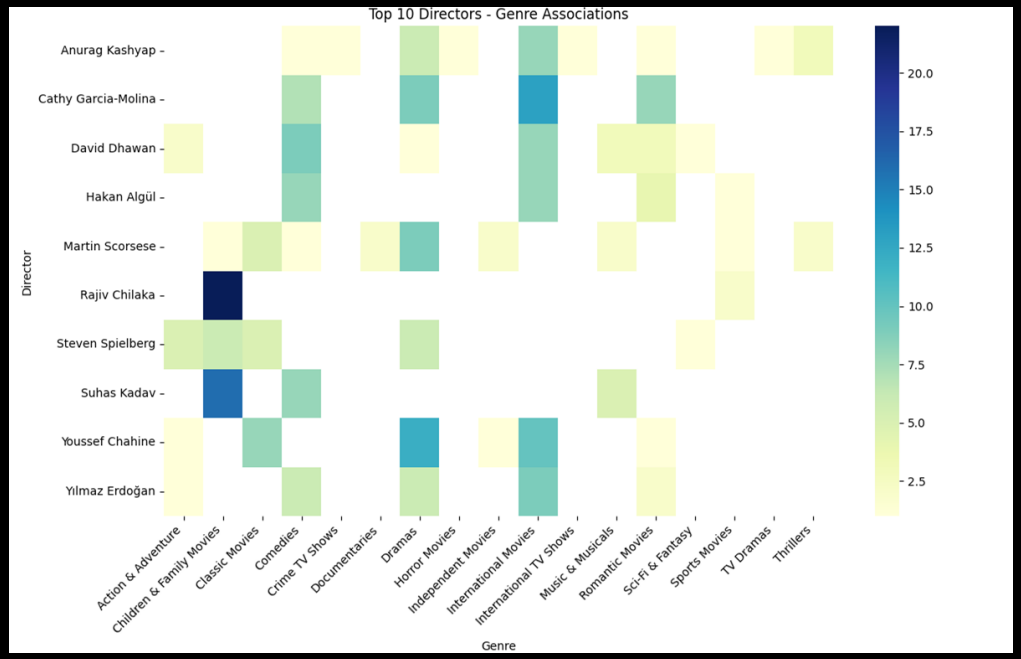


Skip


Popularity of Genres Over Time
Number of Titles
Release Year
Genre
Action & Adventure
Anime Features
Anime Series
British TV Shows
Children & Family Movies
Classic & Cult TV
Classic Movies
Comedies
Crime TV Shows
Cult Movies
Documentaries
Docuseries
Dramas
Faith & Spirituality
Horror Movies
Independent Movies
International Movies
International TV Shows
Kids' TV
Korean TV Shows
LGBTQ Movies
Movies
Music & Musicals
Reality TV


Top Cast Members and Their Appearances in Different Genres
Number of Appearances
Genre
Anupam Kher
Shah Rukh Khan
Naseeruddin Shah
Akshay Kumar
Paresh Rawal
Om Puri
Amitabh Bachchan
Takahiro Sakurai
Yuki Kaji
Boman Irani


Top Countries and Their Appearances in Different Genres
Number of Appearances
Genre
United States
India
United Kingdom
France
Canada
Japan
South Korea
Spain
Germany
Mexico


Top 10 Directors - Genre Associations
Director
Genre
Anurag Kashyap
Cathy Garcia-Molina
David Dhawan
Hakan Algül
Martin Scorsese
Rajiv Chilaka
Steven Spielberg
Suhas Kadav
Youssef Chahine
Yılmaz Erdoğan

Main Slide

INSIGHTS

Inferences and Conclusion

We have drawn many interesting inferences from the dataset Netflix titles; here's a summary of the few of them:

1. The most content type on Netflix is `Movies`.

2. The country by the amount of the produces content is the `United States`,

3. The most popular director on Netflix , with the most titles, is `Rajiv Chilaka`.

4. International Movies is a genre that is mostly in Netflix.

5. largest count of Netflix content is made with a `TV-MA` rating.

6. The most popular actor on Netflix movie, based on the number of titles, is `Anupam Kher`.

It's clear that Netflix has grown over the years. We can see it from the data that the company took certain approaches in their marketing strategy to break into new markets around the world.

Main Slide

THANKS