Netflix_Data_Exploration_and_Visualization (2)

July 2, 2024

1 Netflix Data Exploration and Visualization

About NETFLIX

Netflix is one of the most popular media and video streaming platforms. They have over 10000 movies or tv shows available on their platform, as of mid-2021, they have over 222M Subscribers globally. This tabular dataset consists of listings of all the movies and tv shows available on Netflix, along with details such as - cast, directors, ratings, release year, duration, etc.

Business Problem

Analyze the data and generate insights that could help Netflix ijn deciding which type of shows/movies to produce and how they can grow the business in different countries.

DATASET INFO Show_id: Unique ID for every Movie / Tv Show Type: Identifier - A Movie or TV Show Title: Title of the Movie / Tv Show Director: Director of the Movie Cast: Actors involved in the movie/show Country: Country where the movie/show was produced Date_added: Date it was added on Netflix Release_year: Actual Release year of the movie/show Rating: TV Rating of the movie/show Duration: Total Duration - in minutes or number of seasons Listed_in: Genre Description: The summary description

[ ]:
```
#MOUNTING GOOGLE DRIVE
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call
drive.mount("/content/drive", force_remount=True).
```

[ ]:
```
from google.colab import files
uploaded = files.upload()
```

```
<IPython.core.display.HTML object>
```

```
Saving netflix.csv to netflix (3).csv
```

[ ]:
```
import numpy as np
import pandas as pd
import plotly.express as px
import seaborn as sns
from textblob import TextBlob
```

[ ]:
```
df = pd.read_csv("netflix.csv")
df.head(
)
```

[ ]:
```
  show_id     type                  title         director  \
0      s1    Movie   Dick Johnson Is Dead  Kirsten Johnson
1      s2  TV Show           Blood & Water              NaN
2      s3  TV Show               Ganglands  Julien Leclercq
3      s4  TV Show  Jailbirds New Orleans              NaN
4      s5  TV Show           Kota Factory              NaN

                                                cast        country  \
0                                                NaN  United States
1  Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban…   South Africa
2  Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi…            NaN
3                                                NaN            NaN
4  Mayur More, Jitendra Kumar, Ranjan Raj, Alam K…          India

          date_added  release_year rating   duration  \
0  September 25, 2021          2020  PG-13     90 min
1  September 24, 2021          2021  TV-MA  2 Seasons
2  September 24, 2021          2021  TV-MA   1 Season
3  September 24, 2021          2021  TV-MA   1 Season
4  September 24, 2021          2021  TV-MA  2 Seasons

                                      listed_in  \
0                                  Documentaries
1    International TV Shows, TV Dramas, TV Mysteries
2  Crime TV Shows, International TV Shows, TV Act…
3                            Docuseries, Reality TV
4  International TV Shows, Romantic TV Shows, TV …

                                         description
0  As her father nears the end of his life, filmm…
1  After crossing paths at a party, a Cape Town t…
2  To protect his family from a powerful drug lor…
3  Feuds, flirtations and toilet talk go down amo…
4  In a city of coaching centers known to train I…
```

[ ]:
```
df.shape
```

[ ]:
```
(8807, 12)
```

[ ]:
```
df.columns
```

[ ]:
```
Index(['show_id', 'type', 'title', 'director', 'cast', 'country', 'date_added',
       'release_year', 'rating', 'duration', 'listed_in', 'description'],
```

```
      dtype='object')
```

The groupby function is used to group the data by the 'rating' column and then the size() function counts the occurrences of each rating. The reset_index(name='counts') converts the resulting Series into a DataFrame with a column named 'counts'.

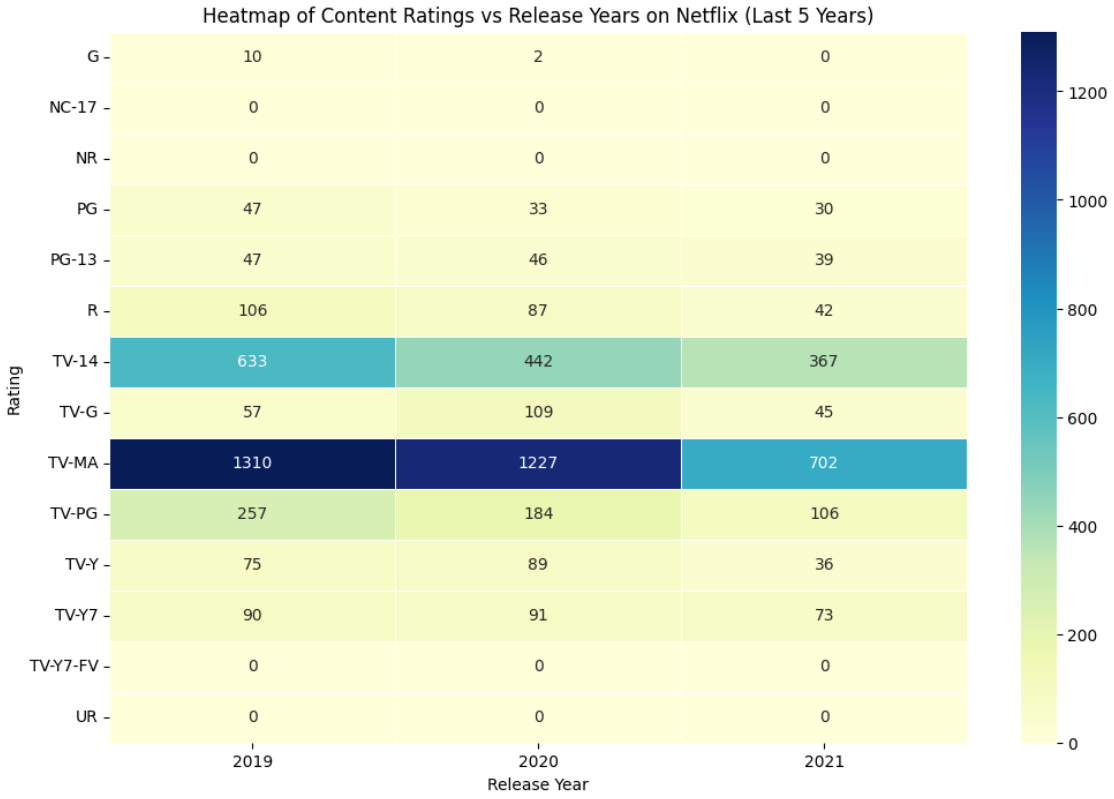
[ ]:
```
z = df.groupby(['rating']).size().reset_index(name='counts')
pieChart = px.pie(z, values='counts', names='rating',
                  title='Distribution of Content Ratings on Netflix',
                  color_discrete_sequence=px.colors.qualitative.Set3)
pieChart.show()
```

Overview: The heatmap illustrates the distribution of Netflix content ratings across the last 5 years (2019-2023). The ratings range from child-friendly categories (e.g., TV-Y) to content intended for mature audiences (e.g., TV-MA).

Detailed Insights: 1. Dominance of TV-MA and TV-14 Ratings: TV-MA (Mature Audiences): Highest Volume: TV-MA content saw the highest numbers, particularly in the years 2021 and 2022. This indicates Netflix's strong focus on content for mature audiences, such as dramas, thrillers, and adult-themed shows. Content Example: This category includes popular shows like "Stranger Things" and "The Witcher". TV-14: Second Most Popular: TV-14 content has also been prevalent, with significant numbers in all five years. This suggests Netflix is also targeting teenagers with a variety of shows suitable for ages 14 and up. Content Example: Shows like "Riverdale" and "13 Reasons Why" fall into this category. 2. Consistent Production of PG and PG-13 Content: PG (Parental Guidance): Steady Production: The amount of PG content has been stable over the years, indicating a continuous effort to produce family-friendly content. Content Example: Movies like "Enola Holmes" fit this category. PG-13: Teen-Friendly: PG-13 content shows a consistent presence, catering to teenagers and young adults. Content Example: Films like "To All the Boys I've Loved Before". 3. Increase in TV-G and TV-Y7 Content in 2021: TV-G (General Audiences): Noticeable Spike in 2021: There was a significant increase in TV-G content in 2021. This indicates a potential strategic shift towards producing more general audience content during this year. Content Example: Shows like "Fuller House". TV-Y7 (Children 7+): Increased Production in 2021: TV-Y7 content also saw an increase, suggesting more focus on content for older children. Content Example: Animated series like "The Dragon Prince". 4. Significant Drop in R and NC-17 Rated Content: R (Restricted): Decreasing Trend: The amount of R-rated content has been lower, suggesting Netflix might be producing fewer adult-only movies or shows. Content Example: Films like "The Irishman". NC-17: Almost Non-existent: NC-17 rated content has practically disappeared, reflecting a potential avoidance of extremely adult content. 5. Special Interest in TV-PG and TV-Y Content: TV-PG: Moderate Presence: TV-PG content has a moderate but stable presence, showing an effort to create content suitable for children with parental guidance. Content Example: Family shows like "A Series of Unfortunate Events". TV-Y (Young Children): Spike in 2020: There was a notable increase in TV-Y content in 2020, likely catering to younger children during the pandemic. Content Example: Shows like "Puffin Rock". 6. No Content for Specific Ratings in Recent Years: NC-17, UR, TV-Y7-FV: Absent Ratings: Ratings such as NC-17 (No Children 17 and Under Admitted), UR (Unrated), and TV-Y7-FV (Fantasy Violence) have no content listed in the past few years. This might be due to stricter content guidelines or changes in viewer preferences. Strategic Shift: This absence suggests a strategic decision to avoid controversial or less popular content ratings. 7. Increase in Total Content: Overall Growth:

The increasing numbers across various ratings indicate Netflix's strategy to expand its library significantly. The company is investing in a broad range of content to cater to diverse audience tastes. Wide Audience Reach: The consistent production across multiple ratings shows Netflix's effort to provide a wide array of content, ensuring there is something for everyone.

[ ]:
```
df[['director', 'cast', 'country']] = df[['director', 'cast', 'country']].
 ↪fillna('Unknown')
df['rating'].replace({"74 min": np.nan, "84 min": np.nan, "66 min": np.nan},␣
 ↪inplace=True)
df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')
df = df.astype({"type": "category", "rating": "category"})

last_5_years = df['release_year'] >= (pd.Timestamp.now().year - 5)
df_last_5_years = df[last_5_years]

df_heatmap = df_last_5_years[['rating', 'release_year']]

heatmap_data = df_heatmap.groupby(['rating', 'release_year']).size().
 ↪reset_index(name='count')

heatmap_matrix = heatmap_data.pivot(index='rating', columns='release_year',␣
 ↪values='count').fillna(0)

plt.figure(figsize=(12, 8))
sns.heatmap(heatmap_matrix, cmap='YlGnBu', annot=True, fmt='g', linewidths=0.5)
plt.title('Heatmap of Content Ratings vs Release Years on Netflix (Last 5␣
 ↪Years)')
plt.xlabel('Release Year')
plt.ylabel('Rating')
plt.show()
```



Conclusion: The heatmap reveals that Netflix has been focusing heavily on producing mature content (TV-MA) while maintaining a substantial amount of content for teenagers (TV-14) and family-friendly options (PG, PG-13). There is also a noticeable increase in content for younger audiences (TV-G, TV-Y7) in recent years. The reduction in R-rated and the absence of NC-17 and UR-rated content reflect a possible strategic shift towards more inclusive and widely acceptable content. Overall, Netflix's content strategy appears to be broad and inclusive, aiming to cater to a wide variety of audience preferences.

[ ]:
```
df['director']=df['director'].fillna('No Director Specified')
filtered_directors=pd.DataFrame()
filtered_directors=df['director'].str.split(',',expand=True).stack()
filtered_directors=filtered_directors.to_frame()
filtered_directors.columns=['Director']
directors=filtered_directors.groupby(['Director']).size().
 ↪reset_index(name='Total Content')
directors=directors[directors.Director !='No Director Specified']
directors=directors.sort_values(by=['Total Content'],ascending=False)
directorsTop5=directors.head()
directorsTop5=directorsTop5.sort_values(by=['Total Content'])
fig1=px.bar(directorsTop5,x='Total Content',y='Director',title='Top 5 Directors␣
 ↪on Netflix')
fig1.show()
```

Here we can see Rajiv chilaka is at the top and Jan suter and Raul Campos tied for the second place

[ ]:
```
df['cast']=df['cast'].fillna('No Cast Specified')
filtered_cast=pd.DataFrame()
filtered_cast=df['cast'].str.split(',',expand=True).stack()
filtered_cast=filtered_cast.to_frame()
filtered_cast.columns=['Actor']
actors=filtered_cast.groupby(['Actor']).size().reset_index(name='Total Content')
actors=actors[actors.Actor !='No Cast Specified']
actors=actors.sort_values(by=['Total Content'],ascending=False)
actorsTop5=actors.head()
actorsTop5=actorsTop5.sort_values(by=['Total Content'])
fig2=px.bar(actorsTop5,x='Total Content',y='Actor', title='Top 5 Actors on␣
 ↪Netflix')
fig2.show()
```

[ ]:
```
Here Anupam Kher is leading the bar charts followed by Rupa Bhimani and␣
 ↪Takahiro Sakuria
```

[ ]:
```
df1=df[['type','release_year']]
df1=df1.rename(columns={"release_year": "Release Year"})
df2=df1.groupby(['Release Year','type']).size().reset_index(name='Total␣
 ↪Content')
df2=df2[df2['Release Year']>=2010]
fig3 = px.line(df2, x="Release Year", y="Total Content",␣
 ↪color='type',title='Trend of content produced over the years on Netflix')
fig3.show()
```

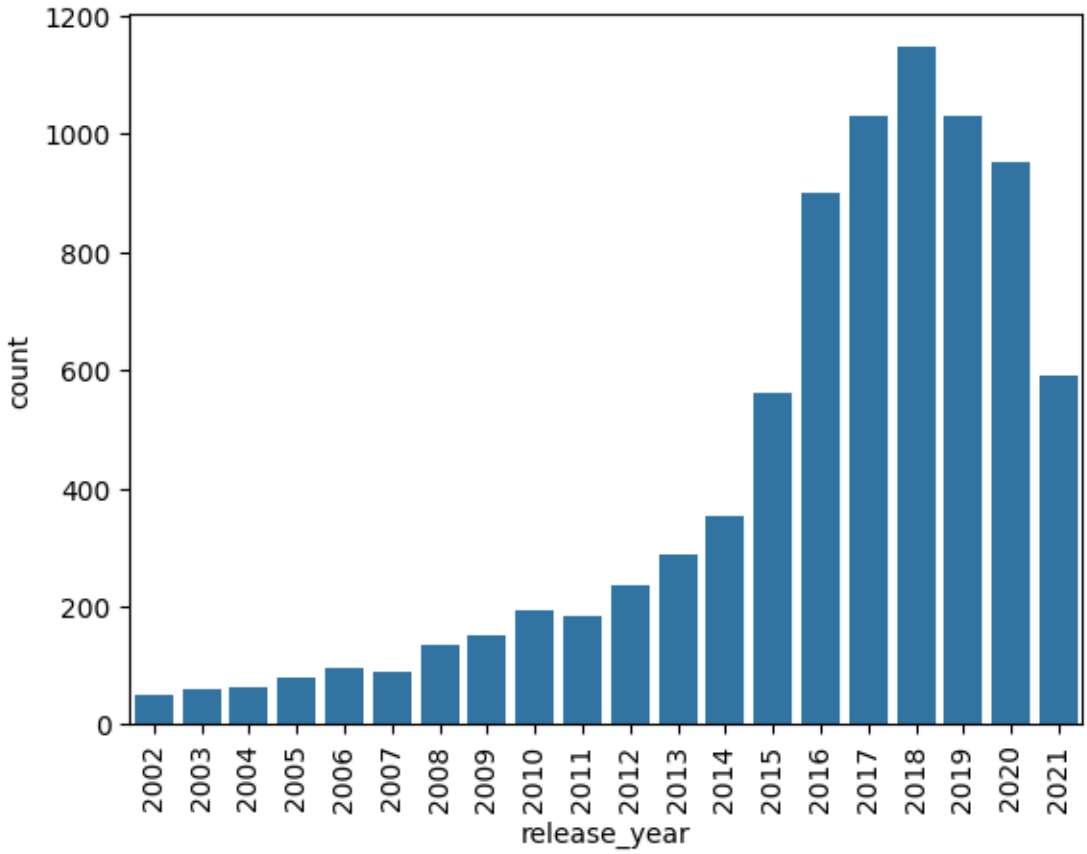
Movies are ar the top before 2020 TV Show crosses movie after 2020 in total content

[ ]:
```
df['count']=1
```

[ ]:
```

```

[ ]:
```
dff1=df.groupby('release_year').sum().reset_index()
dff2=dff1.tail(20)
```

[ ]:
```
import seaborn as sns
import matplotlib.pyplot as plt
sns.barplot(data=dff2,x='release_year',y='count')
plt.xticks(rotation=90)
plt.show()
```



[ ]:

[ ]:
```
df['type'].unique()
```

[ ]:
```
array(['Movie', 'TV Show'], dtype=object)
```

[ ]:
```
movies_df=df[df['type']=='Movie']
tv_show=df[df['type']!='Movie']
movies_df.shape
tv_show.shape
```

[ ]:
```
(2676, 13)
```

[ ]:
```
x=len(movies_df)
y=len(tv_show)
print('The no. of movies in netfix is : ',x,'\n\nThe no. of T V Shows in␣
 ↪netflix is : ',y)
```

```
The no. of movies in netfix is :  6131
```

```
The no. of T V Shows in netflix is :  2676
```

[ ]:
```
df['date_added']=pd.to_datetime(df['date_added'], errors='coerce')
```

[ ]:
```
df['Year']=df['date_added'].dt.year
```

[ ]:
```
dff2=df.groupby('Year')['count'].sum().reset_index()
```

[ ]:
```
dff2.sort_values('count',ascending=False,inplace=True)
dff2.head()
```

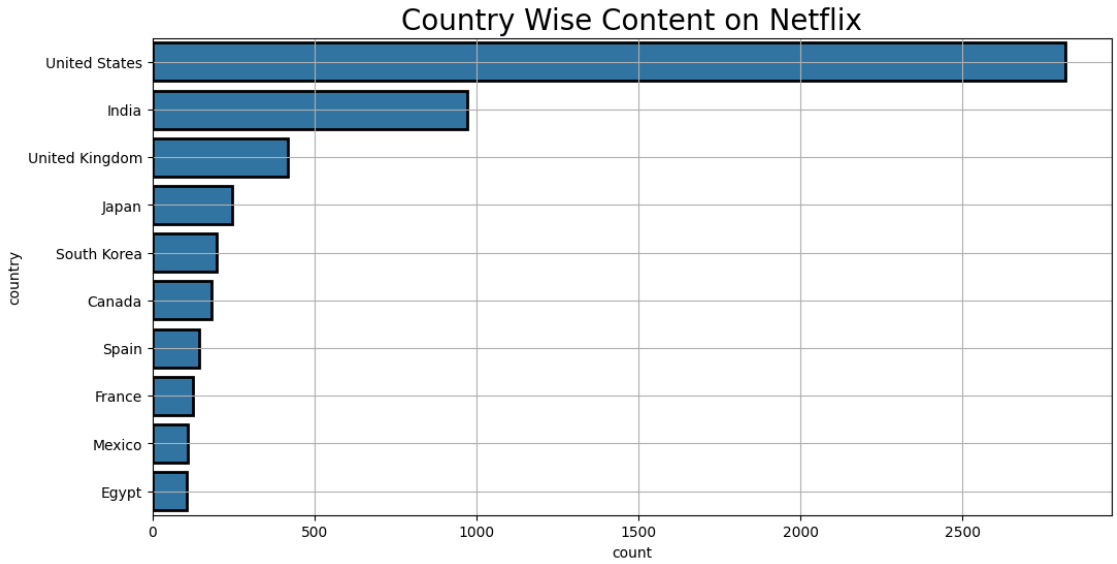
[ ]:
```
      Year  count
11  2019.0   1999
12  2020.0   1878
10  2018.0   1625
13  2021.0   1498
9   2017.0   1164
```

[ ]:
```
df['Month']=df['date_added'].dt.month
```

[ ]:
```
dff3=df.groupby('Month')['count'].sum().reset_index()
dff3.sort_values('count',ascending=False,inplace=True)
dff3.head(5)
```

[ ]:
```
    Month  count
6     7.0    819
11   12.0    797
8     9.0    765
3     4.0    759
9    10.0    755
```

[ ]:
```
plt.figure(figsize=(12,6))
sns.countplot(y='country',order=df['country'].value_counts().index[0:
 ↪10],data=df,ec='black',lw=2)
plt.title('Country Wise Content on Netflix',fontsize=20)
plt.grid(True)
plt.show()
```



[ ]:

[ ]:
```
df['listed_in']
```

[ ]:
```
0                                              Documentaries
1                International TV Shows, TV Dramas, TV Mysteries
2              Crime TV Shows, International TV Shows, TV Act…
3                                       Docuseries, Reality TV
4              International TV Shows, Romantic TV Shows, TV …
                                     …
8802                          Cult Movies, Dramas, Thrillers
8803                 Kids' TV, Korean TV Shows, TV Comedies
8804                                  Comedies, Horror Movies
8805                      Children & Family Movies, Comedies
8806          Dramas, International Movies, Music & Musicals
Name: listed_in, Length: 8807, dtype: object
```

[ ]:
```
df['listed_in'].value_counts().head()
```

[ ]:
```
listed_in
Dramas, International Movies                          362
Documentaries                                         359
Stand-Up Comedy                                       334
Comedies, Dramas, International Movies                274
Dramas, Independent Movies, International Movies      252
Name: count, dtype: int64
```

[ ]:
```
df['duration'].value_counts()
```

[ ]:
```
duration
1 Season     1793
2 Seasons     425
3 Seasons     199
90 min        152
94 min        146
              …
16 min          1
186 min         1
193 min         1
189 min         1
191 min         1
Name: count, Length: 220, dtype: int64
```

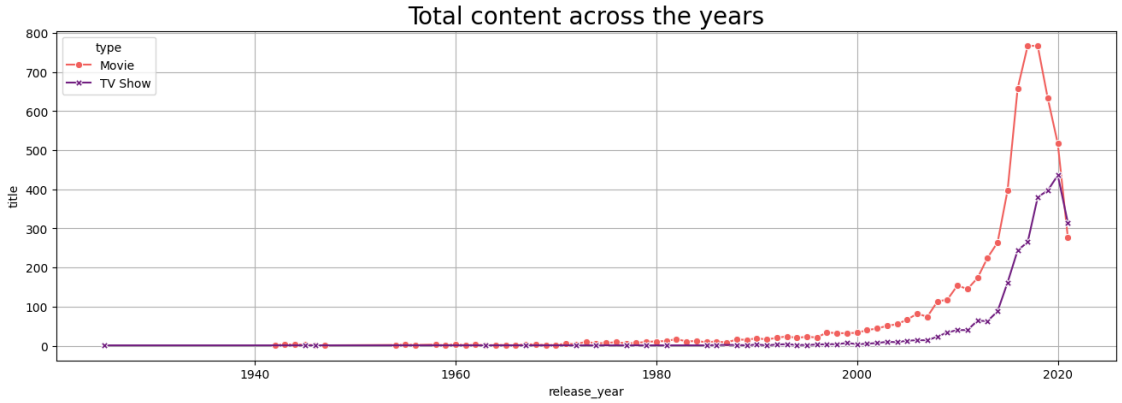
[ ]:
```
x=list(df['duration'].value_counts().head(10).keys())
x
```

[ ]:
```
['1 Season',
 '2 Seasons',
 '3 Seasons',
 '90 min',
 '94 min',
 '97 min',
 '93 min',
 '91 min',
 '95 min',
 '96 min']
```

[ ]:
```
type_year = (df.groupby(['type','release_year'])['title'].size()).reset_index()
type_year
```

[ ]:
```
         type  release_year  title
0       Movie          1942      2
1       Movie          1943      3
2       Movie          1944      3
3       Movie          1945      3
4       Movie          1946      1
..        …             …    …
114  TV Show           2017    265
115  TV Show           2018    380
116  TV Show           2019    397
117  TV Show           2020    436
118  TV Show           2021    315

[119 rows x 3 columns]
```

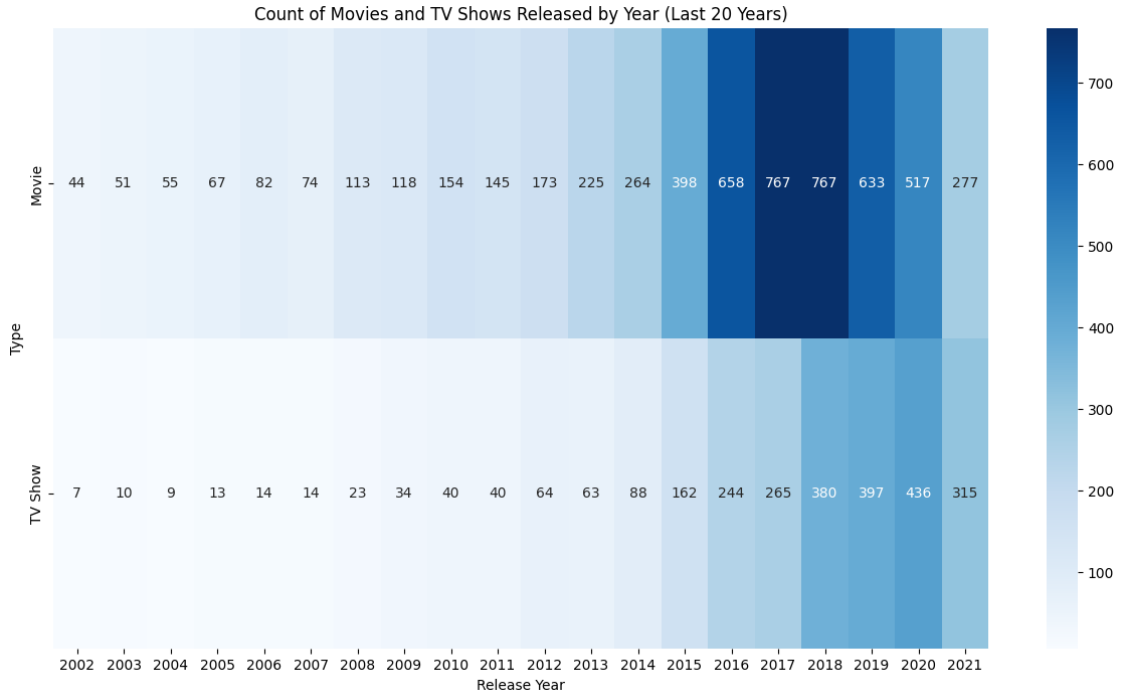
[ ]:
```
fig = plt.figure(figsize=(16,5))
sns.lineplot(data = type_year,x = 'release_year',y = 'title',hue = 'type',style␣
 ↪= 'type',palette='magma_r',markers=True, dashes=False)
plt.title("Total content across the years",fontsize=20)
plt.grid(True)
plt.show()
```



[ ]:
```
dfx=df[['release_year','description']]
dfx=dfx.rename(columns={'release_year':'Release Year'})
for index,row in dfx.iterrows():
    z=row['description']
    testimonial=TextBlob(z)
    p=testimonial.sentiment.polarity
    if p==0:
        sent='Neutral'
    elif p>0:
        sent='Positive'
    else:
        sent='Negative'
    dfx.loc[[index,2],'Sentiment']=sent


dfx=dfx.groupby(['Release Year','Sentiment']).size().reset_index(name='Total␣
 ↪Content')

dfx=dfx[dfx['Release Year']>=2010]
fig4 = px.bar(dfx, x="Release Year", y="Total Content", color="Sentiment",␣
 ↪title="Sentiment of content on Netflix")
fig4.show()
```

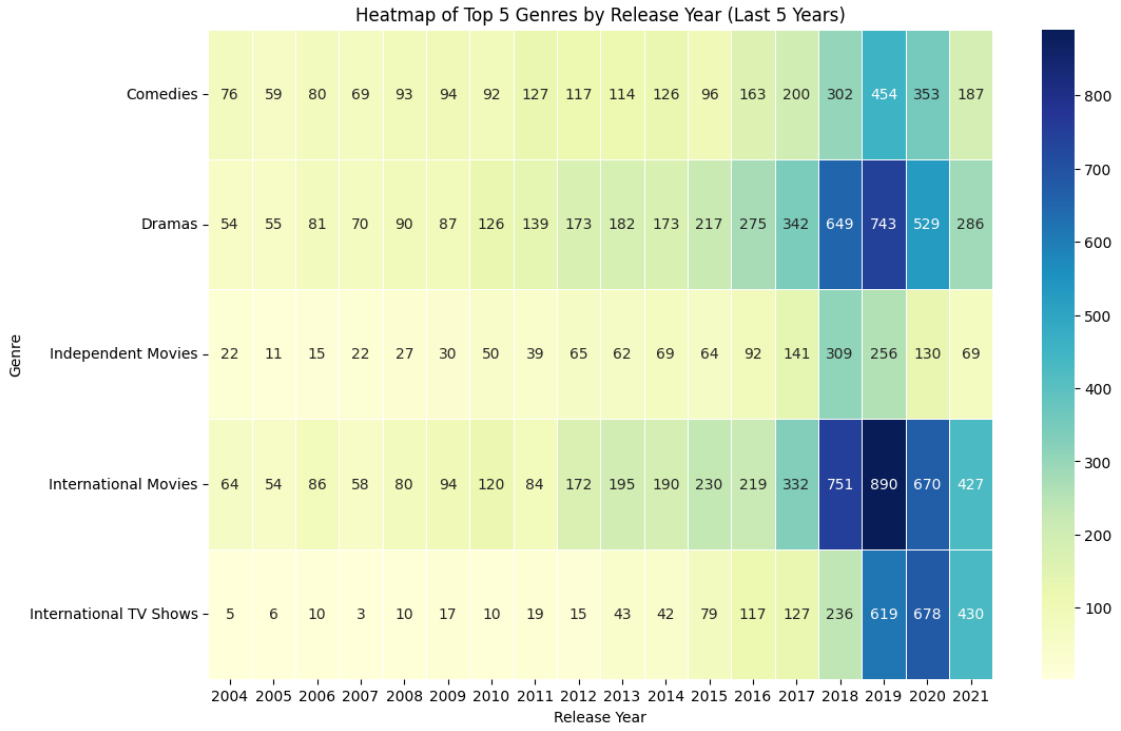
[ ]:
```
df_last_20_years = df[df['release_year'] >= df['release_year'].max() - 19]
```

```
df_agg = df_last_20_years.groupby(['release_year', 'type']).size().
 ↪reset_index(name='count')

heatmap_data = df_agg.pivot(index='type', columns='release_year',␣
 ↪values='count')

plt.figure(figsize=(15, 8))
sns.heatmap(heatmap_data, annot=True, cmap='Blues', fmt='d')
plt.title('Count of Movies and TV Shows Released by Year (Last 20 Years)')
plt.xlabel('Release Year')
plt.ylabel('Type')
plt.show()
```



[ ]:
```
# Data cleaning and preparation
df[['director', 'cast', 'country']] = df[['director', 'cast', 'country']].
 ↪fillna('Unknown')
df['rating'].replace({"74 min": np.nan, "84 min": np.nan, "66 min": np.nan},␣
 ↪inplace=True)
df['date_added'] = pd.to_datetime(df['date_added'], errors='coerce')
df = df.astype({"type": "category", "rating": "category"})

# Filter data for the last 5 years
last_5_years = df['date_added'].dt.year >= (pd.Timestamp.now().year - 5)
df_last_5_years = df[last_5_years]
```

```
# Explode the genres into multiple rows
df_last_5_years = df_last_5_years.assign(listed_in=df_last_5_years['listed_in'].
 ↪str.split(',')).explode('listed_in')

# Strip any leading or trailing whitespace characters
df_last_5_years['listed_in'] = df_last_5_years['listed_in'].str.strip()

# Group by genre and release year, then count the number of shows/movies in␣
 ↪each genre per year
genre_counts = df_last_5_years.groupby(['listed_in', 'release_year']).size().
 ↪reset_index(name='count')

# Get the top 5 genres by total count
top_5_genres = genre_counts.groupby('listed_in')['count'].sum().nlargest(5).
 ↪index

# Filter the data to include only the top 5 genres
top_5_genre_counts = genre_counts[genre_counts['listed_in'].isin(top_5_genres)]

# Pivot the data to create a matrix for the heatmap
heatmap_data = top_5_genre_counts.pivot(index='listed_in',␣
 ↪columns='release_year', values='count').fillna(0)

# Plotting the heatmap
plt.figure(figsize=(12, 8))
sns.heatmap(heatmap_data, cmap='YlGnBu', annot=True, fmt='g', linewidths=0.5)
plt.title('Heatmap of Top 5 Genres by Release Year (Last 5 Years)')
plt.xlabel('Release Year')
plt.ylabel('Genre')
plt.show()
```



Insights: 1. Comedies: Consistent Popularity: Comedies have consistently been among the top genres released on Netflix over the last 5 years. Steady Growth: There's a noticeable increase in the number of comedy releases, especially from 2019 onwards, indicating a growing audience preference for humorous content. Peak in 2020: The year 2020 saw the highest number of comedy releases, possibly reflecting a demand for lighthearted entertainment during challenging times. 2. Dramas: Strong Presence: Dramas maintain a strong presence across all years, often being the second most released genre after comedies. Stable Trend: The number of drama releases shows a stable trend, with slight fluctuations but overall consistent numbers each year. Consistent Audience Appeal: This genre appeals to a wide audience seeking compelling storytelling and emotional depth in their entertainment choices. 3. International Movies: Increasing Trend: There's a noticeable upward trend in the release of international movies, particularly from 2019 to 2021. Diverse Offerings: This genre reflects Netflix's strategy to cater to global audiences with a diverse range of international cinema. High Growth in 2021: 2021 stands out as a peak year for international movie releases, suggesting Netflix's increasing focus on expanding its global content library. 4. International TV Shows: Rising Trend: Similar to international movies, international TV shows have shown a steady increase in releases over the last 5 years. Consistent Growth: The numbers have been consistently growing, with significant increases from 2019 onwards. Diverse Cultural Content: This genre highlights Netflix's efforts to offer culturally diverse and region-specific TV shows to cater to international audiences. 5. Independent Movies: Varied Performance: Independent movies show a more variable trend compared to other genres. Fluctuations: The numbers fluctuate from year to year, indicating varying audience reception or Netflix's evolving strategy in promoting independent films. Steady Releases: Despite fluctuations, there's a consistent effort to include independent films in Netflix's catalog, appealing to viewers interested in unique storytelling and artistic expression. Conclusion: # The heatmap reveals a dynamic landscape of content genres on Netflix over the last

5 years. Comedies and dramas emerge as consistent favorites, reflecting widespread audience appeal for both light-hearted and emotionally engaging content. International movies and TV shows show significant growth, underscoring Netflix's commitment to global diversity in its content offerings. Independent movies, while showing variability, remain an integral part of Netflix's strategy to cater to niche audiences interested in alternative and indie filmmaking. Overall, the heatmap illustrates Netflix's diverse content strategy aimed at meeting varied viewer preferences and expanding its global footprint in the streaming industry.