netflix-business-case

November 10, 2023

1 Netflix - Data Exploration and Visualisation Business Case

About Netflix: Netflix, Inc. is an American technology and media services provider and production company headquartered in Los Gatos, California Netflix was founded by Marc Randolph and Reed Hastings on August 29, 1997, in Scotts Valley, California. The company's primary business is its subscription-based streaming service, which offers online streaming of a library of films and television series, including those produced in-house. 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.

---

2 1. Defining Problem Statement and Analysing basic metrics

Import Libraries

Importing the libraries we need

[236]:
```
import numpy as np
import pandas as pd
import matplotlib
import matplotlib.pyplot as plt
import seaborn as sns
```

3 Loading The Dataset

[237]:
```
netflix_df = pd.read_csv("Business Case Netflix.csv")
```

Let's check the first 5 data

[238]:
```
netflix_df.head()
```

[238]:
```
  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…
```

Let's check the full data

```
[239]: netflix_df
```

```
[239]:      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
       …        …      …                    …              …
       8802   s8803    Movie                 Zodiac    David Fincher
       8803   s8804  TV Show            Zombie Dumb              NaN
       8804   s8805    Movie             Zombieland  Ruben Fleischer
       8805   s8806    Movie                   Zoom     Peter Hewitt
       8806   s8807    Movie                 Zubaan      Mozez Singh
```

```
                                                   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
…                                                 …            …
8802  Mark Ruffalo, Jake Gyllenhaal, Robert Downey J…  United States
8803                                                NaN            NaN
8804  Jesse Eisenberg, Woody Harrelson, Emma Stone, …  United States
8805  Tim Allen, Courteney Cox, Chevy Chase, Kate Ma…  United States
8806  Vicky Kaushal, Sarah-Jane Dias, Raaghav Chanan…          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
…                  …           …    …        …
8802   November 20, 2019          2007      R    158 min
8803        July 1, 2019          2018  TV-Y7  2 Seasons
8804    November 1, 2019          2009      R     88 min
8805    January 11, 2020          2006     PG     88 min
8806       March 2, 2019          2015  TV-14    111 min

                                              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

                                            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…
…                                                 …
```

```
8802  A political cartoonist, a crime reporter and a…
8803  While living alone in a spooky town, a young g…
8804  Looking to survive in a world taken over by zo…
8805  Dragged from civilian life, a former superhero…
8806  A scrappy but poor boy worms his way into a ty…

[8807 rows x 12 columns]
```

The dataset contains over 8807 titles, 12 descriptions. After a quick view of the data frames, it looks like a typical movie/TV shows data frame without ratings. We can also see that there are NaN values in some columns.

4 2. Observations on the shape of data, data types of all the attributes, conversion of categorical attributes to 'category' (If required), missing value detection, statistical summary

To get All Columns of this data so we have to check attributes by netflix_df.columns .

[240]:
```
netflix_df.columns
```

[240]:
```
Index(['show_id', 'type', 'title', 'director', 'cast', 'country', 'date_added',
       'release_year', 'rating', 'duration', 'listed_in', 'description'],
      dtype='object')
```

The shape of data : The shape of data can be checked by netflix.ndim. it is a 2-Dimensional dataset.

[241]:
```
netflix_df.ndim
```

[241]:
```
2
```

Data types of all the Columns

[242]:
```
netflix_df.info()
```

```
<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
```

Statistical Summary Before Data Cleaning:

[243]:

```
netflix_df.describe()
```

[243]:

```
       release_year
count   8807.000000
mean    2014.180198
std        8.819312
min     1925.000000
25%     2013.000000
50%     2017.000000
75%     2019.000000
max     2021.000000
```

Missing Value Detection:

Data Profiling & Cleaning

Data Cleaning means the process of identifying incorrect, incomplete, inaccurate, irrelevant, or missing pieces of data and then modifying, replacing, or deleting them as needed. Data Cleansing is considered as the basic element of Data Science.

[244]:

```
print('\nColumns with missing value:')
print(netflix_df.isnull().any())
```

```

Columns with missing value:
show_id         False
type            False
title           False
director         True
cast             True
country          True
date_added       True
release_year    False
rating           True
duration         True
listed_in       False
description     False
dtype: bool
```

From the info, we know that there are 8807 entries and 12 columns to work with for this EDA. There are a few columns that contain null values, “director,” “cast,” “country,” “date_added,” “rating.”

```
[245]: netflix_df.T.apply(lambda x: x.isnull().sum(), axis = 1)
```

```
[245]: show_id            0
       type               0
       title              0
       director        2634
       cast             825
       country          831
       date_added        10
       release_year       0
       rating             4
       duration           3
       listed_in          0
       description        0
       dtype: int64
```

```
[246]: netflix_df.isnull().sum().sum()
```

```
[246]: 4307
```

There are a total of 4307 null values across the entire dataset with 2634 missing points under “director”, 825 under “cast”, 831 under “country”, 11 under “date_added”, 4 under “rating” and 3 under “duration”. We will have to handle all null data points before we can dive into EDA and modelling.

Imputation is a treatment method for missing value by filling it in using certain techniques

Can use mean, mode, or use predictive modelling. In this case study, we will discuss the use of the fillna function from Pandas for this imputation. Drop rows containing missing values. Can use the dropna function from Pandas.

```
[247]: netflix_df.director.fillna("No Director", inplace=True)
       netflix_df.cast.fillna("No Cast", inplace=True)
       netflix_df.country.fillna("Country Unavailable", inplace=True)
       netflix_df.dropna(subset=["date_added","duration", "rating"], inplace=True)
```

```
[248]: netflix_df.isnull().any()
```

```
[248]: show_id         False
       type            False
       title           False
       director        False
       cast            False
       country         False
```

```
date_added      False
release_year    False
rating          False
duration        False
listed_in       False
description     False
dtype: bool
```

For missing values, the easiest way to get rid of them would be to delete the rows with the missing data. However, this wouldn't be beneficial to our EDA since the is a loss of information. Since "director", "cast", and "country" contain the majority of null values, we chose to treat each missing value is unavailable. The other two label "date_added"," duration" and "rating" contain an insignificant portion of the data, so it drops from the dataset. Finally, we can see that there are no more missing values in the data frame.

Statistical Summary After Data Cleaning:

[249]:

```
netflix_df.describe()
```

[249]:

```
       release_year
count   8790.000000
mean    2014.183163
std        8.825466
min     1925.000000
25%     2013.000000
50%     2017.000000
75%     2019.000000
max     2021.000000
```

5 3. Non-Graphical Analysis:

Non-Graphical Analysis involves calculating the summary statistics, without using pictorial or graphical representations. There are 3 main functions that Pandas library provide us, and I will be discussing about them. Those functions are:

1. **info()**

2. **isna().sum() or isnull().sum()**

3. **describe()**

then we will dive into Value count and Unique attributes.

Checking the data using .head()

[250]:

```
netflix_df.head()
```

[250]:
```
  show_id     type                  title         director  \
0      s1    Movie   Dick Johnson Is Dead  Kirsten Johnson
1      s2  TV Show          Blood & Water      No Director
2      s3  TV Show              Ganglands  Julien Leclercq
3      s4  TV Show  Jailbirds New Orleans      No Director
4      s5  TV Show           Kota Factory      No Director

                                                cast              country  \
0                                            No Cast        United States
1  Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban…         South Africa
2  Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi…  Country Unavailable
3                                            No Cast  Country Unavailable
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…
```

1.info() - mainly indicates the number of features, non-null count, and data type of each features. Additionally, it also shows the number of features in present in each data type(s). This helps us to determine how many numerical and categorical features we have.

[251]:
```
netflix_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 8790 entries, 0 to 8806
Data columns (total 12 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   show_id       8790 non-null   object
 1   type          8790 non-null   object
 2   title         8790 non-null   object
 3   director      8790 non-null   object
 4   cast          8790 non-null   object
 5   country       8790 non-null   object
 6   date_added    8790 non-null   object
 7   release_year  8790 non-null   int64
 8   rating        8790 non-null   object
 9   duration      8790 non-null   object
 10  listed_in     8790 non-null   object
 11  description   8790 non-null   object
dtypes: int64(1), object(11)
memory usage: 892.7+ KB
```

2.Read The Description Of The Data

```
[252]: netflix_df.describe()
```

```
[252]:        release_year
       count   8790.000000
       mean    2014.183163
       std        8.825466
       min     1925.000000
       25%     2013.000000
       50%     2017.000000
       75%     2019.000000
       max     2021.000000
```

3. isna().sum() or isnull().sum()

```
[253]: netflix_df.T.apply(lambda x: x.isnull().sum(), axis = 1)
```

```
[253]: show_id         0
       type            0
       title           0
       director        0
       cast            0
       country         0
       date_added      0
       release_year    0
       rating          0
       duration        0
       listed_in       0
       description     0
       dtype: int64
```

Director having More Counts

```
[373]: df1 = netflix_df["director"].value_counts().to_frame().reset_index().
        ↪rename(columns = {"index" : "Director_Name" ,"director" : "Count"})
       df1
```

```
[373]:
                       Director_Name  Count
0                        No Director   2621
1                      Rajiv Chilaka     19
2             Raúl Campos, Jan Suter     18
3                        Suhas Kadav     16
4                       Marcus Raboy     16
…                              …    …
4522  Raymie Muzquiz, Stu Livingston      1
4523                    Joe Menendez      1
4524                      Eric Bross      1
4525                  Will Eisenberg      1
4526                     Mozez Singh      1

[4527 rows x 2 columns]
```

In this data , No Director shows us there is no director name present in the director column but having Max number od count so it's great to handle NaN data and Rajiv Chilaka has direct the maximum number of shows on netflix and He is the director of the Animation shows like cartoons "CHOTA BHEEM" is the one of the popular shows.

Countries where Netflix is Popular

```
[374]: df2 = netflix_df["country"].value_counts().to_frame().reset_index().
        ↪rename(columns = {"index" : "Country_Name" ,"country" : "Count"})
       df2
```

```
[374]:
                               Country_Name  Count
0                             United States   2809
1                                     India    972
2                       Country Unavailable    829
3                            United Kingdom    418
4                                     Japan    243
..                                      …    …
744              Romania, Bulgaria, Hungary      1
745                      Uruguay, Guatemala      1
746                France, Senegal, Belgium      1
747  Mexico, United States, Spain, Colombia      1
748            United Arab Emirates, Jordan      1

[749 rows x 2 columns]
```

After analysing this data Movies and TV Show on Netflix is most liked by the United States followed by India and then United Kingdom. And Netflix is not the first choice in United Arab Emirates, Jordan and Mexico.

Addition of Movies & TV Shows over time

```
[375]: df3 = netflix_df["release_year"].value_counts().to_frame().reset_index().
        ↪rename(columns = {"index" : "Year" ,"release_year" : "Count"})
       df3
```

```
[375]:     Year  Count
       0   2018   1146
       1   2017   1030
       2   2019   1030
       3   2020    953
       4   2016    901
       ..   …      …
       69  1959      1
       70  1925      1
       71  1961      1
       72  1947      1
       73  1966      1

       [74 rows x 2 columns]
```

In this analysis we find that Maximum number of the Movies and TV Show added on Netflix in 2018 means the busy year on the netflix is 2018 followed by 2017 and 2019.

Counts of Movies and TV Show on Netflix

```
[376]: df4 = netflix_df["type"].value_counts().to_frame().reset_index().rename(columns␣
        ↪= {"index" : "Type" ,"type" : "Count"})
       df4
```

```
[376]:       Type  Count
       0    Movie   6126
       1  TV Show   2664
```

In this Analysis we find that the movies and Tv Show counts in the Netlix data, here we have 6126 counts for Movies and 2664 count for TV Show.

Actors by Movie/TV Show Count

```
[378]: df5 = netflix_df["cast"].value_counts().to_frame().reset_index().rename(columns␣
        ↪= {"index" : "Cast" ,"cast" : "Count"})
       df5
```

```
[378]:                                                  Cast  Count
       0                                             No Cast    825
       1                                   David Attenborough     19
       2     Vatsal Dubey, Julie Tejwani, Rupa Bhimani, Jig…     14
       3                                         Samuel West     10
       4                                         Jeff Dunham      7
```

```
…                                                    …    …
7674  Sanjay Dutt, Arjun Kapoor, Kriti Sanon, Zeenat…      1
7675  Lika Berning, Bobby van Jaarsveld, Marlee van …      1
7676  Lisa Vicari, Dennis Mojen, Walid Al-Atiyat, Ch…      1
7677  Piotr Cyrwus, Mikołaj Kubacki, Anna Radwan, Ma…      1
7678  Vicky Kaushal, Sarah-Jane Dias, Raaghav Chanan…      1

[7679 rows x 2 columns]
```

In this Analysis we find that there is 825 null value in cast column and David Attenborough did 19 movies which is listed in Netflix followed by Vatsal Dubey , Rupa Bhimani , Julie Tejwani did 14 movies .

6 4. Visual Analysis - Univariate, Bivariate after pre-processing of the data

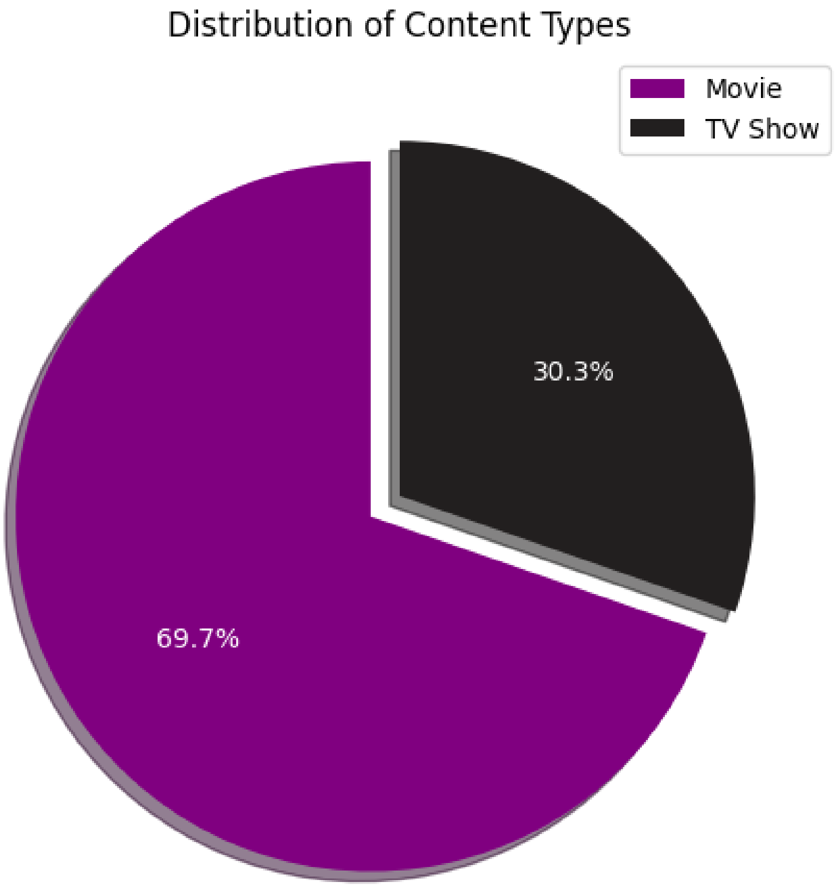
---

4.1. Netflix Content By Type -

Analysis entire Netflix dataset consisting of both movies and shows. Let's compare the total number of movies and shows in this dataset to know which one is the majority.

[371]:

```python
# Calculate the percentage distribution of content types
x = netflix_df.groupby(['type'])['type'].count()
y = len(netflix_df)
r = ((x/y) * 100).round(2)

# Create a DataFrame to store the percentage distribution
mf_ratio = pd.DataFrame(r)
mf_ratio.rename({'type': '%'}, axis=1, inplace=True)

# Plot the 3D-effect pie chart
plt.figure(figsize=(6, 6))
colors = ['#800080', '#221f1f']
explode = (0.1, 0)
plt.pie(mf_ratio['%'], labels=mf_ratio.index, autopct='%1.1f%%',
colors=colors, explode=explode, shadow=True, startangle=90,
textprops={'color': 'white'})

plt.legend(loc='upper right')
plt.title('Distribution of Content Types')
plt.show()
```



There are far more movie titles (69.7%) that TV shows titles (30.3%) in terms of title

4.2. Amount of Content as a Function of Time: Distplot

we will explore the amount of content Netflix has added throughout the previous years. Since we are interested in when Netflix added the title onto their platform, we will add a "year_added" column to show the date from the "date_added" columns.

```
[369]: netflix_df["year_added"] = pd.to_datetime(netflix_df.date_added).dt.year
       netflix_df
```

```
[369]:      show_id     type                   title          director  \
       0         s1    Movie    Dick Johnson Is Dead  Kirsten Johnson
       1         s2  TV Show           Blood & Water      No Director
       2         s3  TV Show               Ganglands  Julien Leclercq
       3         s4  TV Show   Jailbirds New Orleans      No Director
       4         s5  TV Show            Kota Factory      No Director
       …        …       …                       …                …
       8802   s8803    Movie                  Zodiac    David Fincher
```

```
8803   s8804  TV Show            Zombie Dumb      No Director
8804   s8805    Movie             Zombieland  Ruben Fleischer
8805   s8806    Movie                   Zoom     Peter Hewitt
8806   s8807    Movie                 Zubaan      Mozez Singh

                                                   cast              country  \
0                                               No Cast        United States
1     Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban…         South Africa
2     Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi…  Country Unavailable
3                                               No Cast  Country Unavailable
4     Mayur More, Jitendra Kumar, Ranjan Raj, Alam K…                India
…                                                   …                  …
8802  Mark Ruffalo, Jake Gyllenhaal, Robert Downey J…        United States
8803                                            No Cast  Country Unavailable
8804  Jesse Eisenberg, Woody Harrelson, Emma Stone, …        United States
8805  Tim Allen, Courteney Cox, Chevy Chase, Kate Ma…        United States
8806  Vicky Kaushal, Sarah-Jane Dias, Raaghav Chanan…                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
…                    …            …    …        …
8802   November 20, 2019          2007      R    158 min
8803        July 1, 2019          2018  TV-Y7  2 Seasons
8804    November 1, 2019          2009      R     88 min
8805    January 11, 2020          2006     PG     88 min
8806       March 2, 2019          2015  TV-14    111 min

                                              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

                                            description  year_added  \
0     As her father nears the end of his life, filmm…        2021
1     After crossing paths at a party, a Cape Town t…        2021
```

```
2     To protect his family from a powerful drug lor…        2021
3     Feuds, flirtations and toilet talk go down amo…        2021
4     In a city of coaching centers known to train I…        2021
…                                                  …         …
8802  A political cartoonist, a crime reporter and a…        2019
8803  While living alone in a spooky town, a young g…        2019
8804  Looking to survive in a world taken over by zo…        2019
8805  Dragged from civilian life, a former superhero…        2020
8806  A scrappy but poor boy worms his way into a ty…        2019

     month_added
0      September
1      September
2      September
3      September
4      September
…            …
8802    November
8803        July
8804    November
8805     January
8806       March

[8790 rows x 14 columns]
```

[370]:

```
netflix_year_df = netflix_df["year_added"].value_counts().to_frame().
  ↪reset_index().rename(columns={"index": "year",
"year_added":"count"})
```

[ ]:

```
netflix_year_df
```

[306]:

```
movies_data = netflix_df.loc[netflix_df["type"] == "Movie"]
movies_year_df = movies_data.year_added.value_counts().to_frame().reset_index().
  ↪rename(columns={"index":
"year", "year_added":"count"})
```

[ ]:

```
movies_data
```

[260]:

```
movies_year_df
```

[260]:

```
   year  count
0  2019   1424
1  2020   1284
2  2018   1237
3  2021    993
4  2017    836
5  2016    251
```

```
6   2015     56
7   2014     19
8   2011     13
9   2013      6
10  2012      3
11  2009      2
12  2008      1
13  2010      1
```

[307]:
```
tvShow_data = netflix_df.loc[netflix_df["type"] == "TV Show"]
shows_year_df = tvShow_data.year_added.value_counts().to_frame().reset_index().
  ↪rename(columns={"index":
"year", "year_added":"count"})
```

[262]:
```
tvShow_data
```

[262]:
```
     show_id     type                 title         director  \
1         s2  TV Show         Blood & Water      No Director
2         s3  TV Show             Ganglands  Julien Leclercq
3         s4  TV Show  Jailbirds New Orleans      No Director
4         s5  TV Show          Kota Factory      No Director
5         s6  TV Show        Midnight Mass    Mike Flanagan
…        …       …                    …               …
8795   s8796  TV Show         Yu-Gi-Oh! Arc-V      No Director
8796   s8797  TV Show           Yunus Emre      No Director
8797   s8798  TV Show             Zak Storm      No Director
8800   s8801  TV Show      Zindagi Gulzar Hai      No Director
8803   s8804  TV Show           Zombie Dumb      No Director

                                                   cast  \
1     Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban…
2     Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi…
3                                               No Cast
4     Mayur More, Jitendra Kumar, Ranjan Raj, Alam K…
5     Kate Siegel, Zach Gilford, Hamish Linklater, H…
…                                                  …
8795  Mike Liscio, Emily Bauer, Billy Bob Thompson, …
8796  Gökhan Atalay, Payidar Tüfekçioglu, Baran Akbu…
8797  Michael Johnston, Jessica Gee-George, Christin…
8800  Sanam Saeed, Fawad Khan, Ayesha Omer, Mehreen …
8803                                            No Cast

                                      country          date_added  \
1                                South Africa  September 24, 2021
2                         Country Unavailable  September 24, 2021
3                         Country Unavailable  September 24, 2021
4                                       India  September 24, 2021
```

```
5                           Country Unavailable  September 24, 2021
…                                             …                 …
8795                              Japan, Canada         May 1, 2018
8796                                     Turkey    January 17, 2017
8797  United States, France, South Korea, Indonesia  September 13, 2018
8800                                   Pakistan   December 15, 2016
8803                        Country Unavailable        July 1, 2019

      release_year rating   duration  \
1             2021  TV-MA  2 Seasons
2             2021  TV-MA   1 Season
3             2021  TV-MA   1 Season
4             2021  TV-MA  2 Seasons
5             2021  TV-MA   1 Season
…            …    …        …
8795          2015  TV-Y7  2 Seasons
8796          2016  TV-PG  2 Seasons
8797          2016  TV-Y7  3 Seasons
8800          2012  TV-PG   1 Season
8803          2018  TV-Y7  2 Seasons

                                              listed_in  \
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 …
5                     TV Dramas, TV Horror, TV Mysteries
…                                                  …
8795                              Anime Series, Kids' TV
8796                   International TV Shows, TV Dramas
8797                                            Kids' TV
8800  International TV Shows, Romantic TV Shows, TV …
8803              Kids' TV, Korean TV Shows, TV Comedies

                                            description  year_added
1     After crossing paths at a party, a Cape Town t…        2021
2     To protect his family from a powerful drug lor…        2021
3     Feuds, flirtations and toilet talk go down amo…        2021
4     In a city of coaching centers known to train I…        2021
5     The arrival of a charismatic young priest brin…        2021
…                                                 …         …
8795  Now that he's discovered the Pendulum Summonin…        2018
8796  During the Mongol invasions, Yunus Emre leaves…        2017
8797  Teen surfer Zak Storm is mysteriously transpor…        2018
8800  Strong-willed, middle-class Kashaf and carefre…        2016
8803  While living alone in a spooky town, a young g…        2019
```

```
[2664 rows x 13 columns]
```

[263]:

```
shows_year_df
```

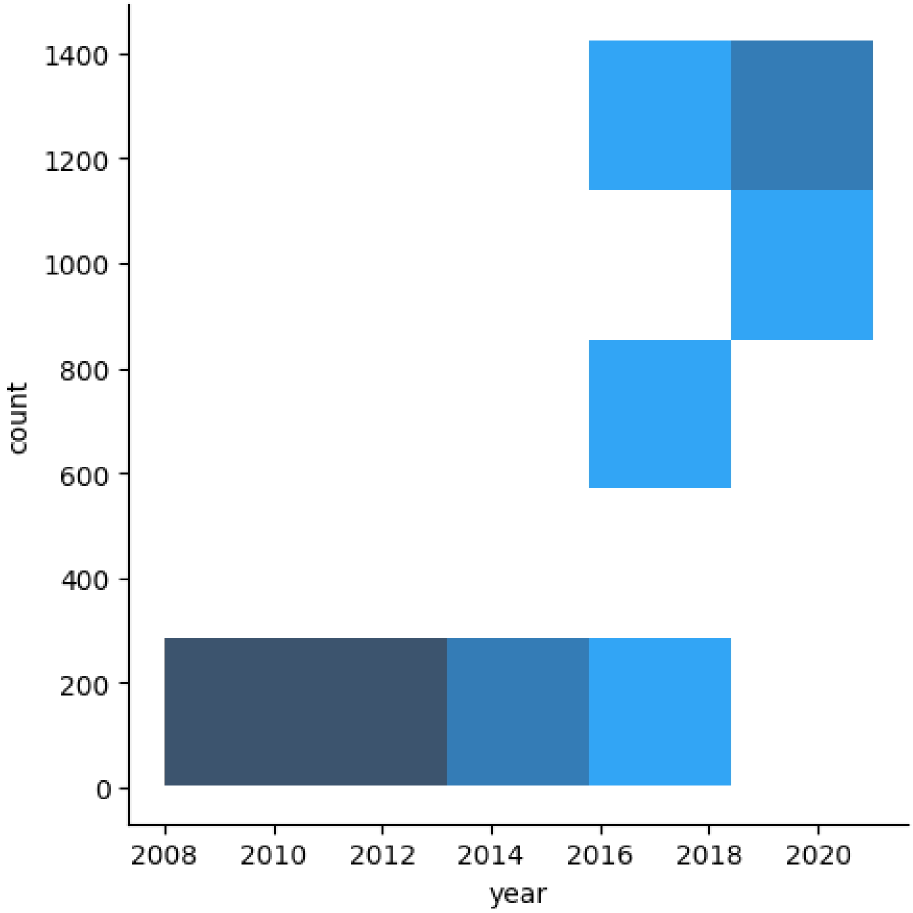
[263]:

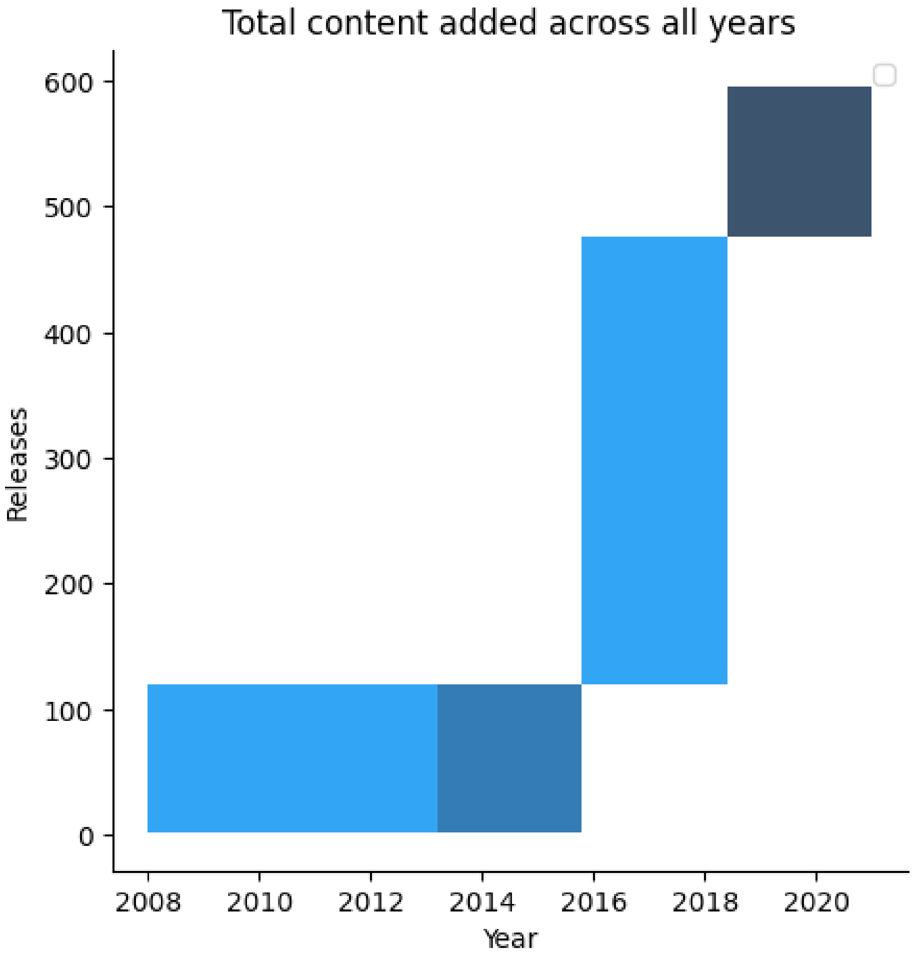
```
   year  count
0  2020    595
1  2019    592
2  2021    505
3  2018    411
4  2017    349
5  2016    175
6  2015     26
7  2014      5
8  2013      5
9  2008      1
```

[384]:

```
fig, ax = plt.subplots(figsize=(7, 5))
sns.displot(data=movies_year_df, x='year', y='count')
sns.displot (data=shows_year_df, x='year', y='count')
ax.set_xticks(np.arange(2008, 2022, 1))
plt.title("Total content added across all years", )
plt.legend(['Movie','TV Show'])
plt.ylabel("Releases")
plt.xlabel("Year")
plt.show()
```

1.0
0.8
0.6
0.4
0.2
0.0


1400
1200
1000
800
600
400
200
0
count
2008
2010
2012
2014
2016
2018
2020
year



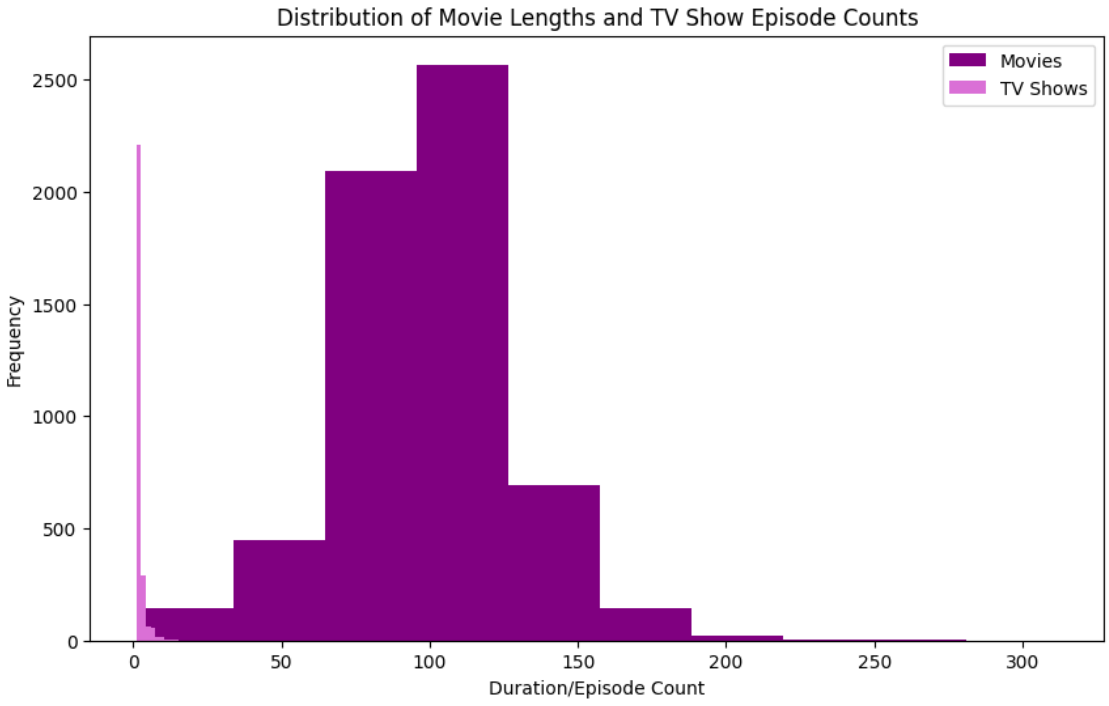
4.3. Distribution of Movie Lengths and TV Show Episode Counts

Understanding the Duration of movies and TV shows provides insights into the content's length and helps viewers plan their watching time. By examining the distribution of movie lengths and TV show durations, we can better understand the content available on Netflix.

To achieve this, we extract the movie lengths, and TV show episode counts from the 'duration' column. We then plot histograms and box plots to visualize the distribution of movie lengths and TV show durations.

[372]:

```
# Extract the movie lengths and TV show episode counts
movie_lengths = df_movies['duration'].str.extract('(\d+)', expand=False).
  ↪astype(int)
tv_show_episodes = df_tv_shows['duration'].str.extract('(\d+)', expand=False).
  ↪astype(int)

# Plot the histogram
plt.figure(figsize=(10, 6))
plt.hist(movie_lengths, bins=10, color='#800080', label='Movies')
plt.hist(tv_show_episodes, bins=10, color='#DA70D6', label='TV Shows')

# Customize the plot
plt.xlabel('Duration/Episode Count')
plt.ylabel('Frequency')
plt.title('Distribution of Movie Lengths and TV Show Episode Counts')
plt.legend()

# Show the plot
plt.show()
```



Analyzing the histograms, we can observe that most movies on Netflix have a duration of around 100 minutes. On the other hand, most TV shows on Netflix have only one season.

Additionally, by examining the box plots, we can see that movies longer than approximately 2.5 hours are considered outliers. For TV shows, finding those with more than four seasons is uncommon.

4.4. Exploring the countries contribution with the most content of Netflix.

Next is exploring the countries by the amount of the produces content of Netflix. We need to separate all countries within a film before analysing it, then removing titles with no countries available.

[291]:

```
import plotly.graph_objects as go
from plotly.offline import init_notebook_mode, iplot
```

We need to separate all countries within a film before analyzing it, then removing titles with no countries available.

[292]:

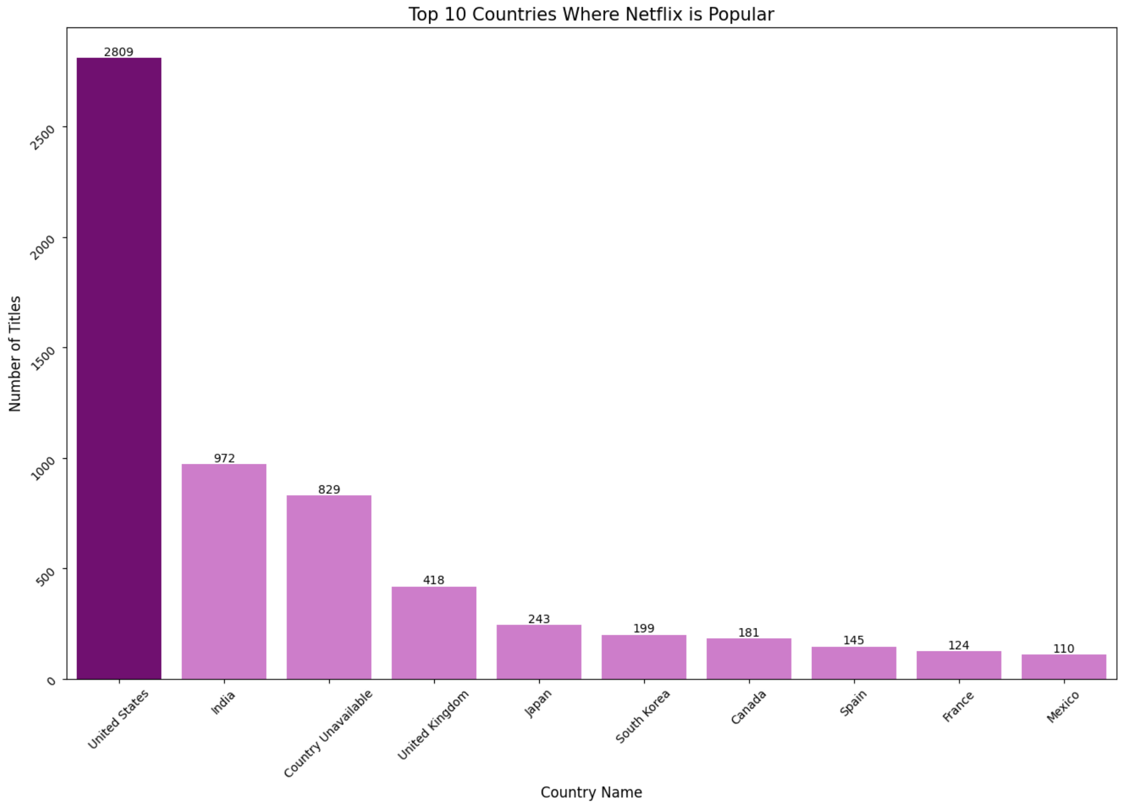
```
filtered_countries = netflix_df.set_index('title').country.str.split(', ',
expand=True).stack().reset_index(level=1, drop=True);
filtered_countries = filtered_countries[filtered_countries != 'Country␣
 ↪Unavailable']
iplot([go.Choropleth(
locationmode='country names',
locations=filtered_countries,
z=filtered_countries.value_counts()
)])
```

4.5. Top 10 Countries Where Netflix is Popular

to identify the top 10 countries where Netflix is popular, we can use the following code:

[352]:

```
# Remove white spaces from 'country' column
netflix_df['country'] = netflix_df['country'].str.rstrip()

# Find value counts
country_counts = netflix_df['country'].value_counts()

# Select the top 10 countries
top_10_countries = country_counts.head(10)

# Plot the top 10 countries
plt.figure(figsize=(16, 10))
colors = ['#800080'] + ['#DA70D6'] * (len(top_10_countries) - 1)
bar_plot = sns.barplot(x=top_10_countries.index, y=top_10_countries.values,␣
 ↪palette=colors)

plt.xlabel('Country Name', fontsize = 12)
plt.ylabel('Number of Titles', fontsize = 12)
plt.title('Top 10 Countries Where Netflix is Popular', fontsize = 15)
plt.xticks(rotation = 45, fontsize = 10)
plt.yticks(rotation = 45, fontsize = 10)

# Add count values on top of each bar
for index, value in enumerate(top_10_countries.values):
    bar_plot.text(index, value, str(value), ha='center', va='bottom')

plt.show()
```



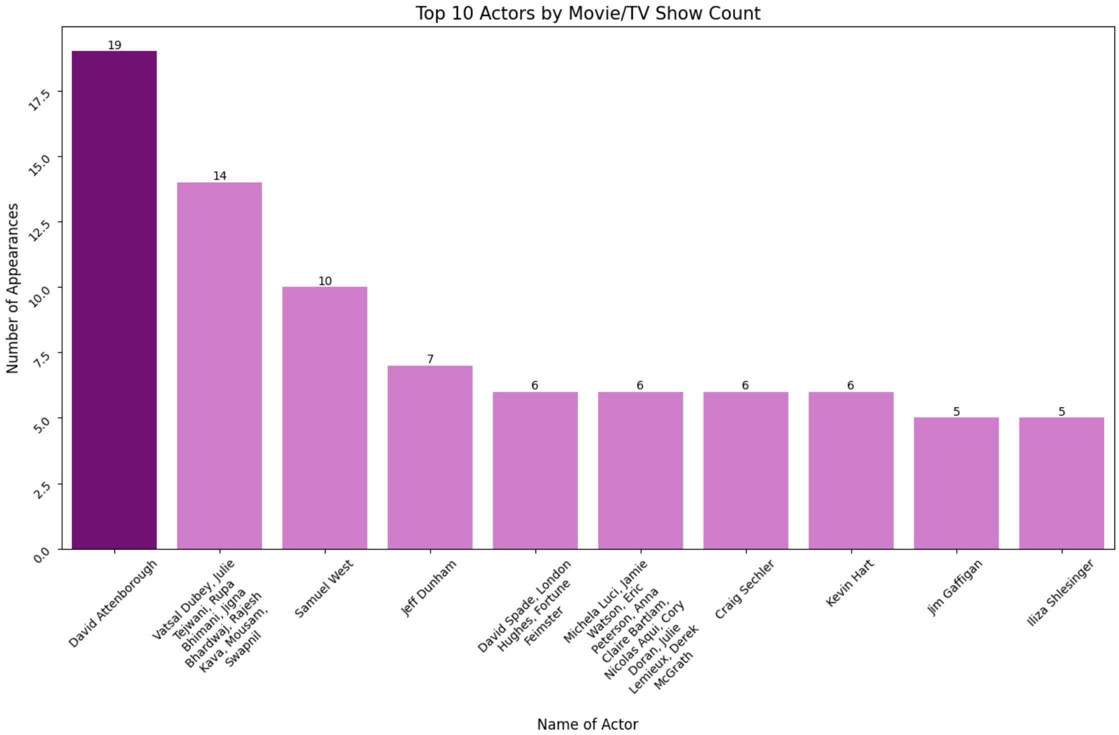
The bar chart visualization reveals that the United States is the top country where Netflix is popular.

4.6. Top 10 Actors by Movie/TV Show Count

To identify the top 10 actors with the highest number of appearances in movies and TV shows, we used the below code:

[349]:

```
# Count the occurrences of each actor
cast_counts = netflix_df['cast'].value_counts()[1:]

# Select the top 10 actors
top_10_cast = cast_counts.head(10)

plt.figure(figsize=(16, 8))
colors = ['#800080'] + ['#DA70D6'] * (len(top_10_cast) - 1)
bar_plot = sns.barplot(x=top_10_cast.index, y=top_10_cast.values,␣
 ↪palette=colors)

plt.xlabel('Name of Actor', fontsize = 12)
plt.ylabel('Number of Appearances', fontsize = 12)
plt.title('Top 10 Actors by Movie/TV Show Count', fontsize = 15)
plt.xticks(rotation = 45, fontsize = 10)
plt.yticks(rotation = 45, fontsize = 10)

import textwrap
max_width = 20
bar_plot.set_xticklabels(textwrap.fill(x.get_text(), max_width) for x in␣
 ↪bar_plot.get_xticklabels())
# Add count values on top of each bar
for index, value in enumerate(top_10_cast.values):
    bar_plot.text(index, value, str(value), ha='center', va='bottom')

plt.show()
```



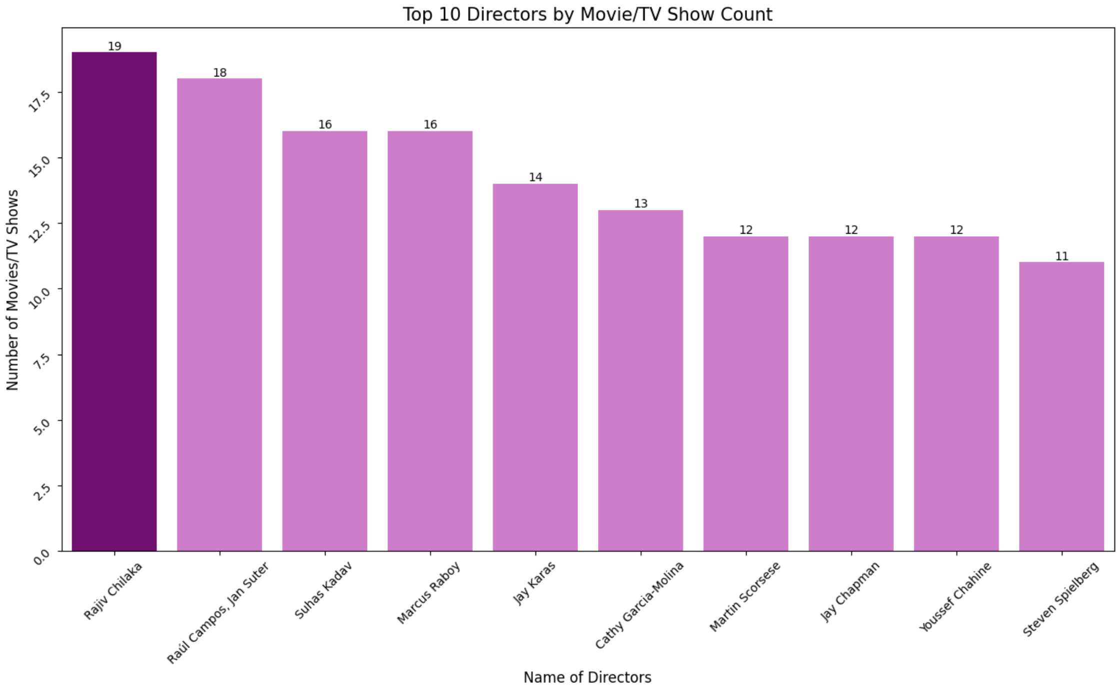
The bar chart shows that David Attenborough has the highest appearances in movies and TV shows

4.7. Top 10 Directors by Movie/TV Show Count

To identify the top 10 directors who have directed the highest number of movies or TV shows, we used the below code:

[348]:
```
# Count the occurrences of each actor
director_counts = netflix_df['director'].value_counts()[1:]

# Select the top 10 actors
```

```
top_10_directors = director_counts.head(10)

plt.figure(figsize=(16, 8))
colors = ['#800080'] + ['#DA70D6'] * (len(top_10_directors) - 1)
bar_plot = sns.barplot(x=top_10_directors.index, y=top_10_directors.values,␣
  ↪palette=colors)

plt.xlabel('Name of Directors', fontsize = 12)
plt.ylabel('Number of Movies/TV Shows', fontsize = 12)
plt.title('Top 10 Directors by Movie/TV Show Count', fontsize = 15)
plt.xticks(rotation = 45, fontsize = 10)
plt.yticks(rotation = 45, fontsize = 10)

# Add count values on top of each bar
for index, value in enumerate(top_10_directors.values):
    bar_plot.text(index, value, str(value), ha='center', va='bottom')

plt.show()
```



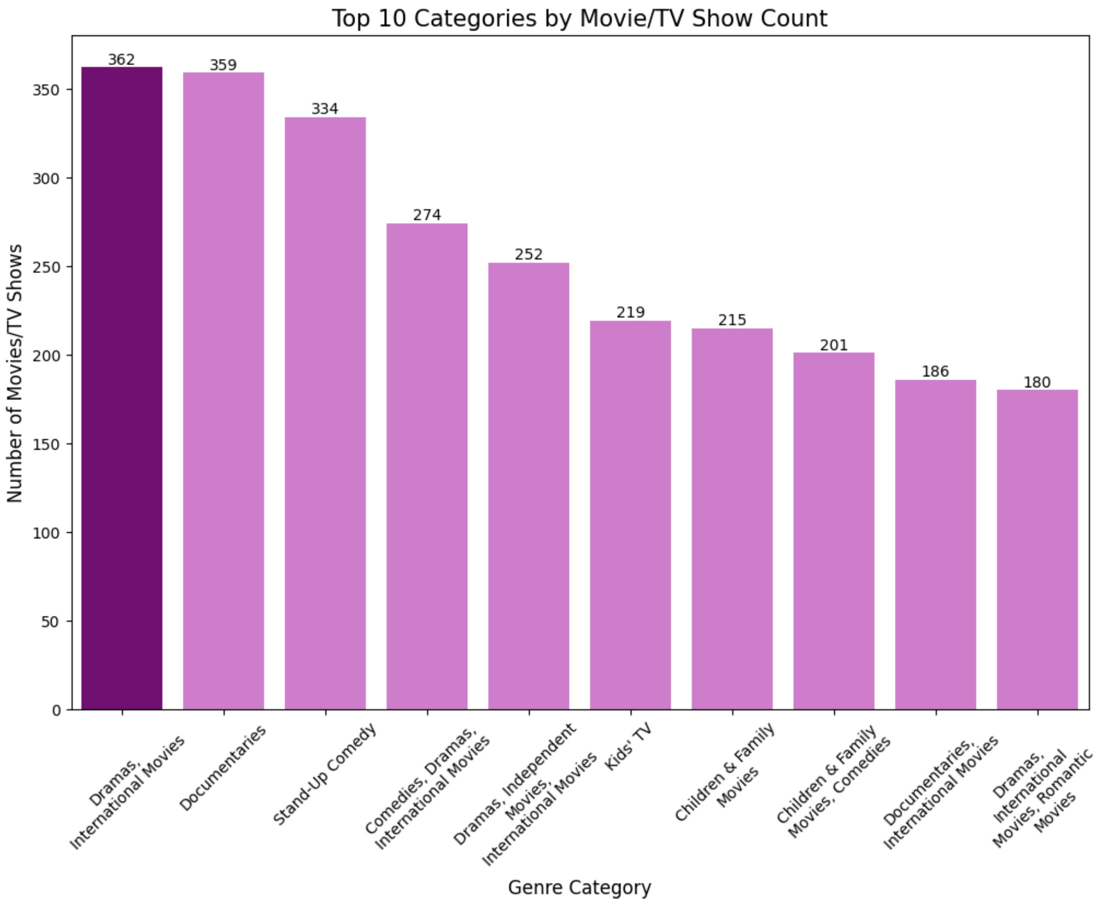
The bar chart displays the top 10 directors with the most movies or TV shows. Rajiv Chilaka seems to have directed the most content in the Netflix library.

4.8. Top 10 Categories by Movie/TV Show Count

To analyze the distribution of content in different categories, we can used the below code:

[358]:

```
netflix_df['listed_in'] = netflix_df['listed_in'].str.strip()

# Count the occurrences of each actor
listed_in_counts = netflix_df['listed_in'].value_counts()

# Select the top 10 actors
top_10_listed_in = listed_in_counts.head(10)
colors = ['#800080'] + ['#DA70D6'] * (len(top_10_directors) - 1)
plt.figure(figsize=(12, 8))
bar_plot = sns.barplot(x=top_10_listed_in.index, y=top_10_listed_in.values,␣
 ↪palette = colors)

# Customize the plot
plt.xlabel('Genre Category', fontsize = 12)
plt.ylabel('Number of Movies/TV Shows', fontsize = 12)
plt.title('Top 10 Categories by Movie/TV Show Count', fontsize = 15)
plt.xticks(rotation=45)

#spliting xticks
import textwrap
max_width = 20
bar_plot.set_xticklabels(textwrap.fill(x.get_text(), max_width) for x in␣
 ↪bar_plot.get_xticklabels())
# Add count values on top of each bar
for index, value in enumerate(top_10_listed_in.values):
    bar_plot.text(index, value, str(value), ha='center', va='bottom')

# Show the plot
plt.show()
```



The bar chart shows the top 10 categories of movies and TV shows based on their count. "International Movies" is the most dominant category, followed by "Dramas."

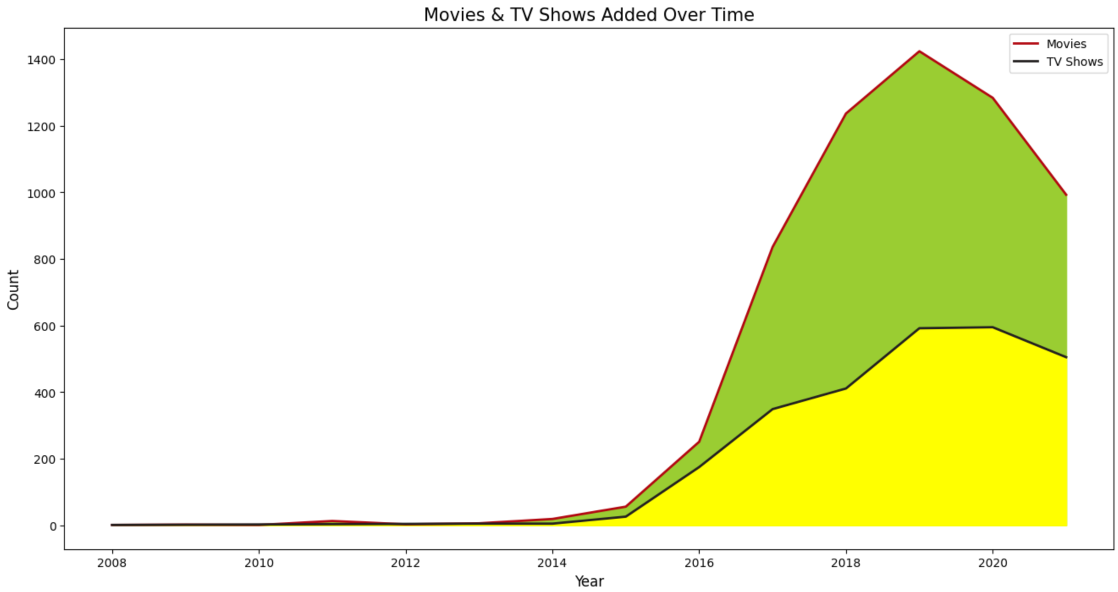
4.9. Movies & TV Shows Added Over Time

To analyze the addition of movies and TV shows over time, we can used the below code:

[361]:

```
# Filter the DataFrame to include only Movies and TV Shows
df_movies = netflix_df[netflix_df['type'] == 'Movie']
df_tv_shows = netflix_df[netflix_df['type'] == 'TV Show']

# Group the data by year and count the number of Movies and TV Shows
# added in each year
movies_count = df_movies['year_added'].value_counts().sort_index()
tv_shows_count = df_tv_shows['year_added'].value_counts().sort_index()

# Create a line chart to visualize the trends over time
plt.figure(figsize=(16, 8))
plt.plot(movies_count.index, movies_count.values, color='#b20710',
label='Movies', linewidth=2)
```

```
plt.plot(tv_shows_count.index, tv_shows_count.values, color='#221f1f',
label='TV Shows', linewidth=2)

# Fill the area under the line charts
plt.fill_between(movies_count.index, movies_count.values, color='#9ACD32')
plt.fill_between(tv_shows_count.index, tv_shows_count.values, color='#FFFF00')

# Customize the plot
plt.xlabel('Year', fontsize = 12)
plt.ylabel('Count', fontsize = 12)
plt.title('Movies & TV Shows Added Over Time', fontsize = 15)
plt.legend()

# Show the plot
plt.show()
```



The line chart illustrates the number of movies and TV shows added to Netflix over time. It visually represents the growth and trends in content additions, with separate lines for films and TV shows.

Netflix saw its real growth starting from the year 2015, & we can see it added more Movies than TV Shows over the years.

Also, it is interesting that the content addition dropped in 2020. This could be due to the pandemic situation.

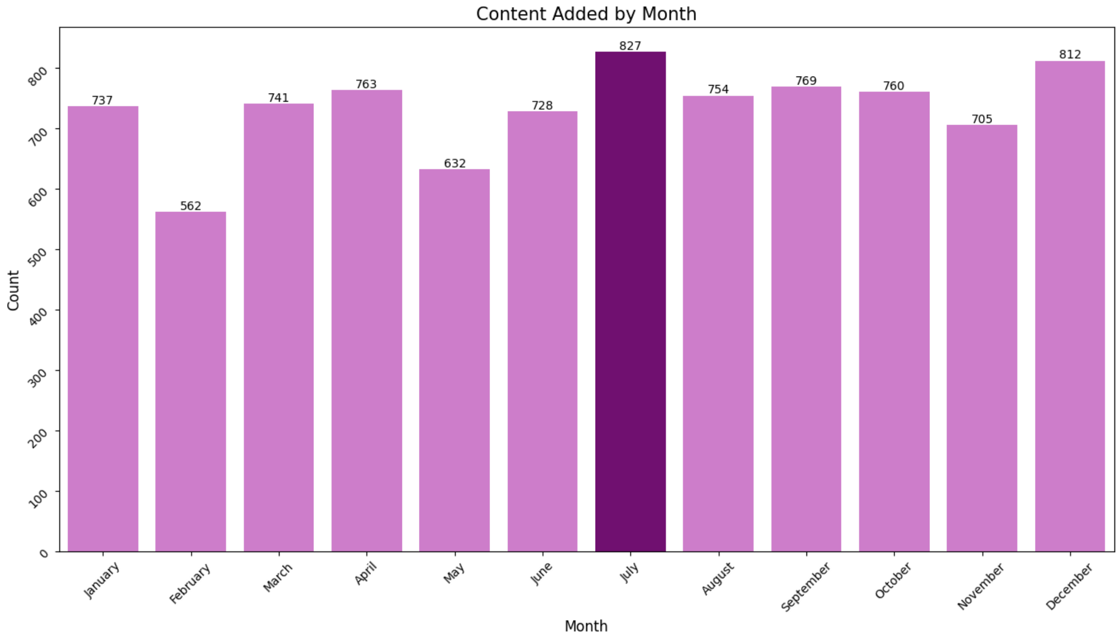
Next, we explore the distribution of content additions across different months. This analysis helps us identify patterns and understand when Netflix introduces new content.

4.10. Content Added by Month

To investigate this, we extract the month from the 'date_added' column and count the occurrences of each month. Visualizing this data as a bar chart allows us to quickly identify the months with the highest content additions.

[363]:

```
# Extract the month from the 'date_added' column
netflix_df['month_added'] = pd.to_datetime(netflix_df['date_added']).dt.
 ↪month_name()

# Define the order of the months
month_order = ['January', 'February', 'March', 'April', 'May', 'June', 'July',
               'August', 'September', 'October', 'November', 'December']

# Count the number of shows added in each month
monthly_counts = netflix_df['month_added'].value_counts().loc[month_order]

# Determine the maximum count
max_count = monthly_counts.max()

# Set the color for the highest bar and the rest of the bars
colors = ['#800080' if count == max_count else '#DA70D6' for count in␣
 ↪monthly_counts]

# Create the bar chart
plt.figure(figsize=(16, 8))
bar_plot = sns.barplot(x=monthly_counts.index, y=monthly_counts.values,␣
 ↪palette=colors)

# Customize the plot
plt.xlabel('Month', fontsize = 12)
plt.ylabel('Count', fontsize = 12)
plt.title('Content Added by Month', fontsize = 15)

# Add count values on top of each bar
for index, value in enumerate(monthly_counts.values):
    bar_plot.text(index, value, str(value), ha='center', va='bottom')

# Rotate x-axis labels for better readability
plt.xticks(rotation = 45, fontsize = 10)
plt.yticks(rotation = 45, fontsize = 10)

# Show the plot
plt.show()
```



The bar chart shows that July and December are the months when Netflix adds the most content to its library. This information can be valuable for viewers who want to anticipate new releases during these months.

Another crucial aspect of Netflix's content analysis is understanding the distribution of ratings. By examining the count of each rating category, we can determine the most prevalent types of content on the platform.

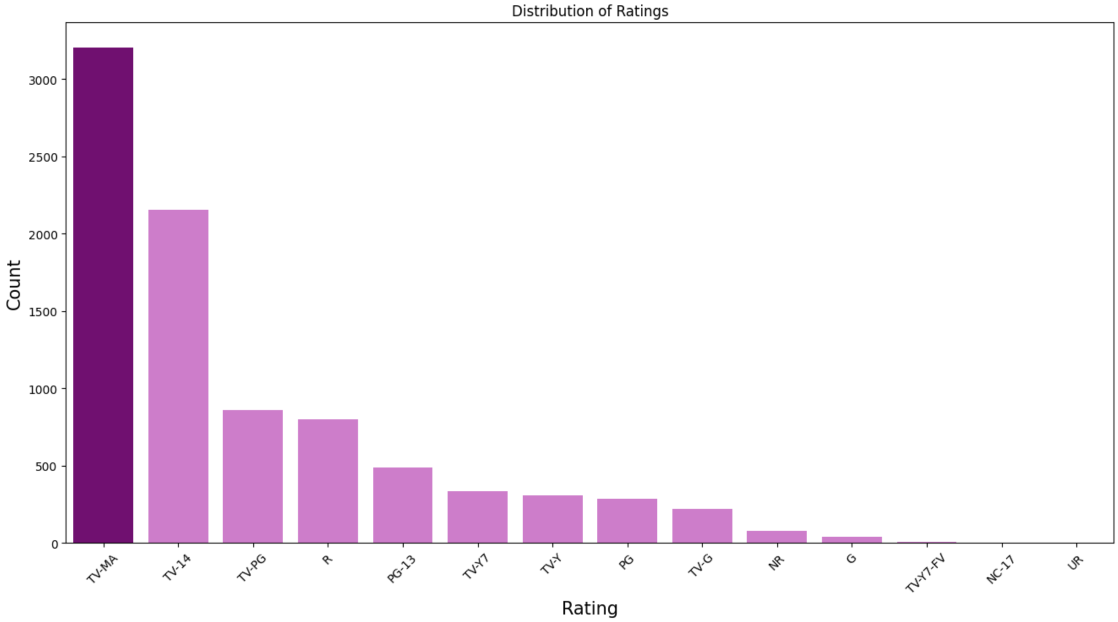
4.11. Distribution of Ratings

We start by calculating the occurrences of each rating category and visualize them using a bar chart. This visualization provides a clear overview of the distribution of ratings.

[364]:

```
# Count the occurrences of each rating
rating_counts = netflix_df['rating'].value_counts()

# Create a bar chart to visualize the ratings
plt.figure(figsize=(16, 8))
colors = ['#800080'] + ['#DA70D6'] * (len(rating_counts) - 1)
sns.barplot(x=rating_counts.index, y=rating_counts.values, palette=colors)

# Customize the plot
plt.xlabel('Rating', fontsize = 15)
plt.ylabel('Count', fontsize = 15)
plt.title('Distribution of Ratings')

# Rotate x-axis labels for better readability
plt.xticks(rotation=45)
```

```
# Show the plot
plt.show()
```



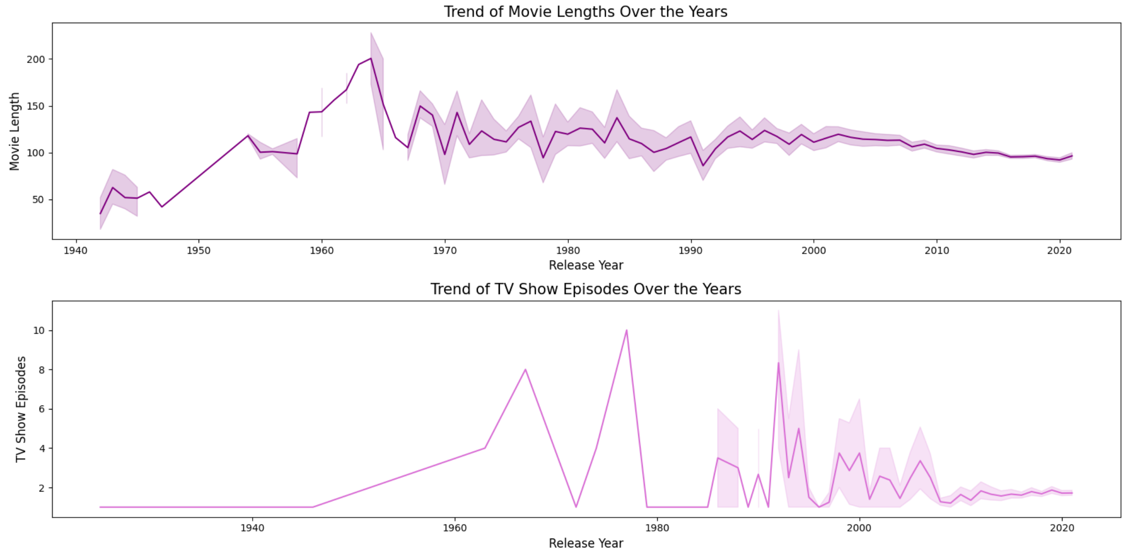
Upon analyzing the bar chart, we can observe the distribution of ratings on Netflix. It helps us identify the most common rating categories and their relative frequency.

4.12. The Trend of Movie/TV Show Lengths Over the Years

We can plot line charts to understand how movie lengths and TV show episode counts have evolved over the years. Identifying patterns or shifts in content duration by analyzing these trends.

We start by extracting the movie lengths and TV show episode counts from the ‘duration’ column. Then, we create line plots to visualize the changes in movie lengths and TV show episodes over the years.

```
[367]: import seaborn as sns
       import matplotlib.pyplot as plt

       # Extract the movie lengths and TV show episodes from the 'duration' column
       movie_lengths = df_movies['duration'].str.extract('(\d+)', expand=False).
        ↪astype(int)
       tv_show_episodes = df_tv_shows['duration'].str.extract('(\d+)', expand=False).
        ↪astype(int)

       # Create line plots for movie lengths and TV show episodes
       plt.figure(figsize=(16, 8))
       plt.subplot(2, 1, 1)
       sns.lineplot(data=df_movies, x='release_year', y=movie_lengths, color=colors[0])
       plt.xlabel('Release Year', fontsize = 12)
       plt.ylabel('Movie Length', fontsize = 12)
       plt.title('Trend of Movie Lengths Over the Years', fontsize = 15 )

       plt.subplot(2, 1, 2)
       sns.lineplot(data=df_tv_shows, x='release_year',␣
        ↪y=tv_show_episodes,color=colors[1])
       plt.xlabel('Release Year', fontsize = 12)
       plt.ylabel('TV Show Episodes', fontsize = 12)
       plt.title('Trend of TV Show Episodes Over the Years', fontsize = 15)

       # Adjust the layout and spacing
       plt.tight_layout()

       # Show the plots
       plt.show()
```



Analyzing the line charts, we observe exciting patterns. We can see that movie length initially increased until around 1963-1964 and then gradually dropped, stabilizing around an average of 100 minutes. This suggests a shift in audience preferences over time.

Regarding TV show episodes, we have noticed a consistent trend since the early 2000s, where most TV shows on Netflix have one to three seasons. This indicates a preference for shorter series or limited series formats among viewers.

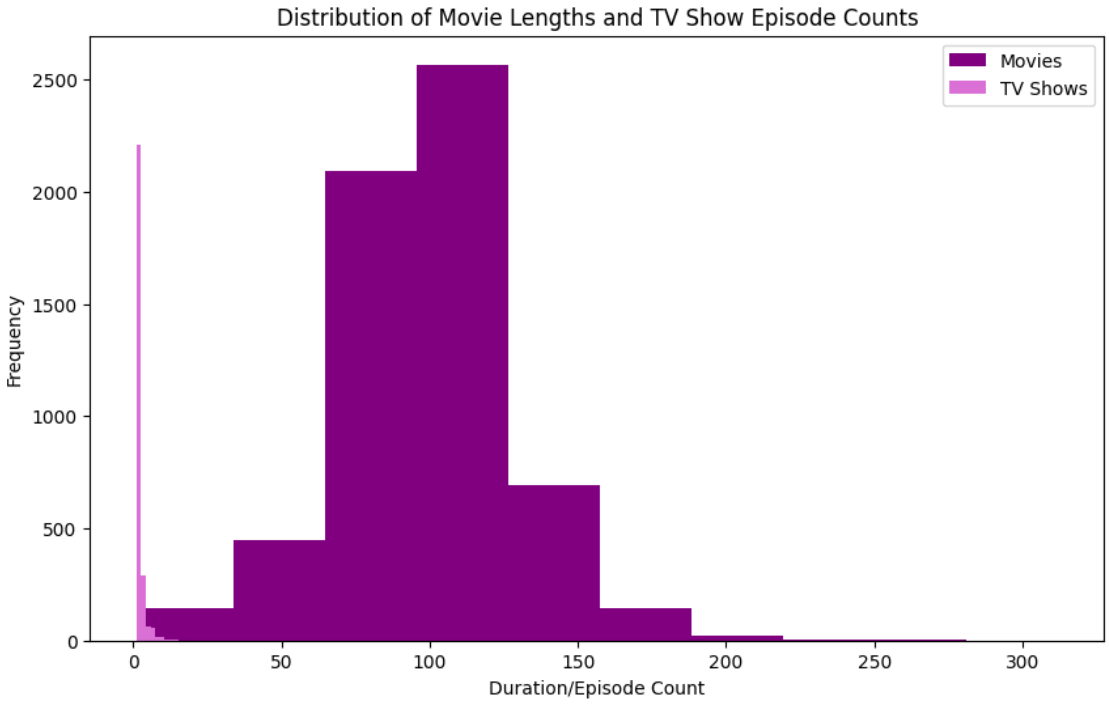
4.13. Distribution of Movie Lengths and TV Show Episode Counts

Understanding the Duration of movies and TV shows provides insights into the content's length and helps viewers plan their watching time. By examining the distribution of movie lengths and TV show durations, we can better understand the content available on Netflix.

To achieve this, we extract the movie lengths, and TV show episode counts from the ‘duration’ column. We then plot histograms and box plots to visualize the distribution of movie lengths and TV show durations.

[368]:

```
# Extract the movie lengths and TV show episode counts
movie_lengths = df_movies['duration'].str.extract('(\d+)', expand=False).
  ↪astype(int)
tv_show_episodes = df_tv_shows['duration'].str.extract('(\d+)', expand=False).
  ↪astype(int)

# Plot the histogram
plt.figure(figsize=(10, 6))
plt.hist(movie_lengths, bins=10, color='#800080', label='Movies')
plt.hist(tv_show_episodes, bins=10, color='#DA70D6', label='TV Shows')

# Customize the plot
plt.xlabel('Duration/Episode Count')
plt.ylabel('Frequency')
plt.title('Distribution of Movie Lengths and TV Show Episode Counts')
plt.legend()

# Show the plot
plt.show()
```



Analyzing the histograms, we can observe that most movies on Netflix have a duration of around 100 minutes. On the other hand, most TV shows on Netflix have only one season.

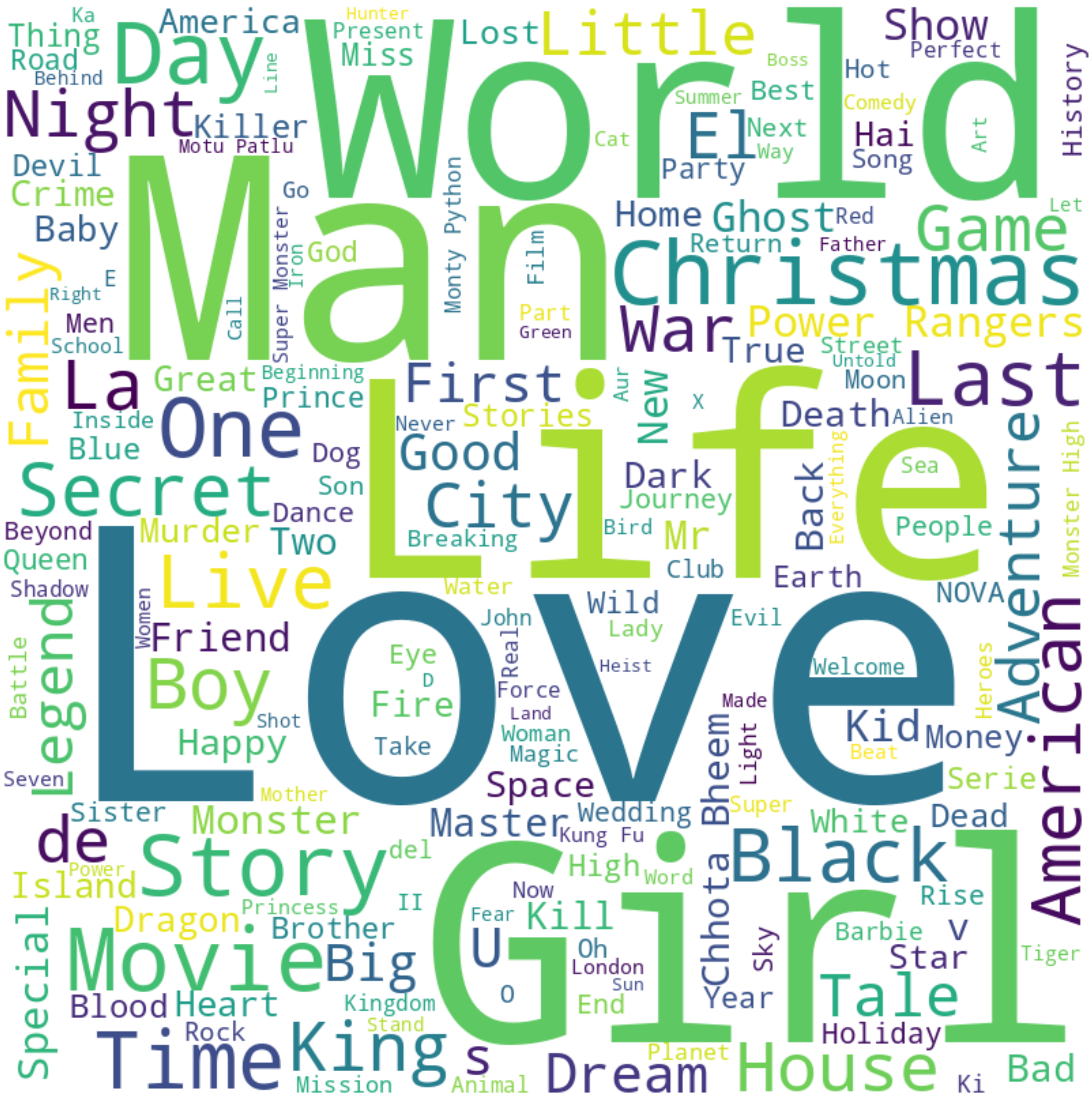
Additionally, by examining the box plots, we can see that movies longer than approximately 2.5 hours are considered outliers. For TV shows, finding those with more than four seasons is uncommon.

4.14. Most Common Words in Titles and Descriptions

Analyzing the most common words used in titles and descriptions can provide insights into the themes and content focus on Netflix. We can generate word clouds to uncover these patterns based on the titles and descriptions of Netflix's content.

[277]:

```
from wordcloud import WordCloud

# Concatenate all the titles into a single string
text = ' '.join(netflix_df['title'])

wordcloud = WordCloud(width = 800, height = 800,
                background_color ='white',
                min_font_size = 10).generate(text)

# plot the WordCloud image
plt.figure(figsize = (8, 8), facecolor = None)
plt.imshow(wordcloud)
plt.axis("off")
plt.tight_layout(pad = 0)

plt.show()
```

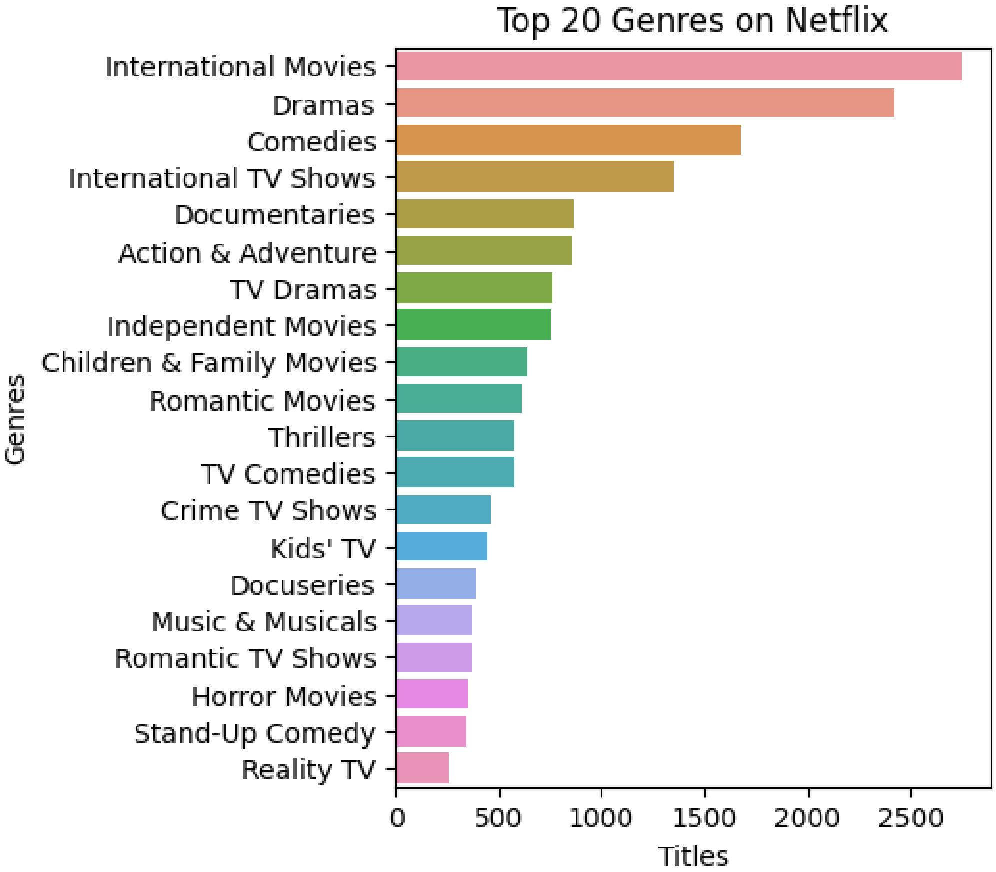


Examining the word cloud for titles, we observe that terms like “Love,” “Girl,” “Man,” “Life,” and “World” are frequently used, indicating the presence of romantic, coming-of-age, and drama genres in Netflix’s content library.

4.15. Top 20 Genres on Netflix: Count Plot

[278]:

```
filtered_genres = netflix_df.set_index('title').listed_in.str.split(', ',
expand=True).stack().reset_index(level=1, drop=True);
plt.figure(figsize=(4,5))
g = sns.countplot(y = filtered_genres,
order=filtered_genres.value_counts().index[:20])
plt.title('Top 20 Genres on Netflix')
plt.xlabel('Titles')
plt.ylabel('Genres')
plt.show()
```



From the graph, we know that International Movies take the first place, followed by dramas and comedies.

7 4.2 For categorical variable(s):

Boxplot

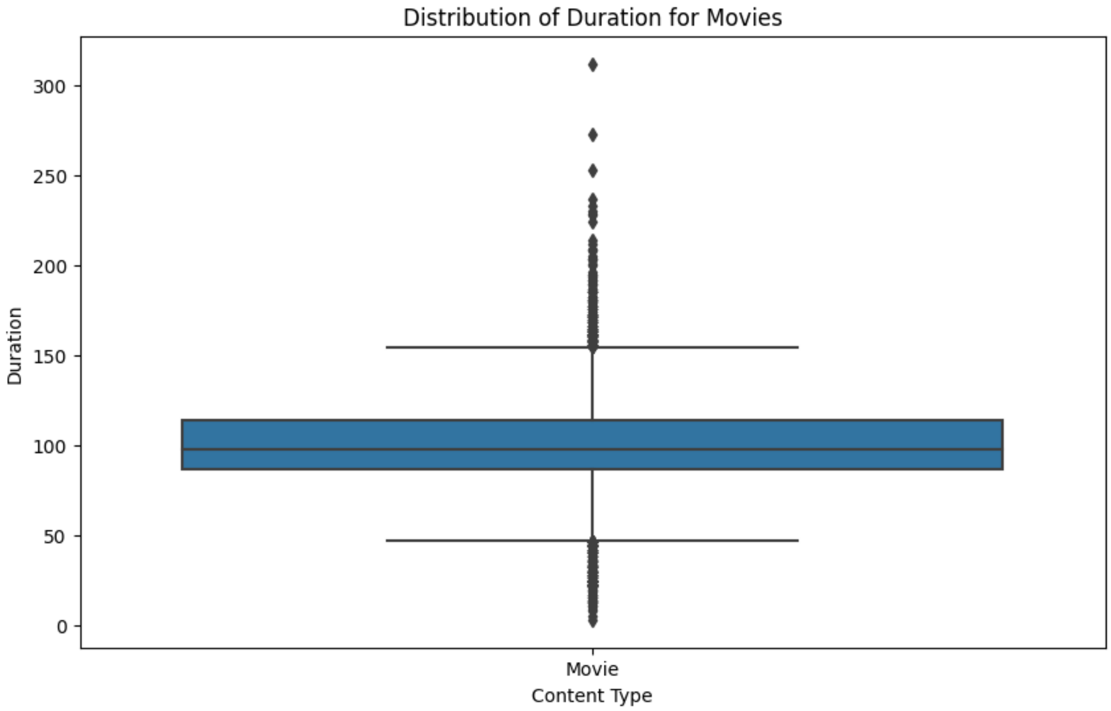
Duration Distribution for Movies and TV Shows

Analysing the duration distribution for movies and TV shows allows us to understand the typical length of content available on Netflix. We can create box plots to visualize these distributions and identify outliers or standard durations.

```
[279]: netflix_movies_df = netflix_df[netflix_df.type.str.contains("Movie")]
       netflix_movies_df['duration'] = netflix_movies_df['duration'].str.
        ↪extract('(\d+)',
       expand=False).astype(int)
       # Creating a boxplot for movie duration
```

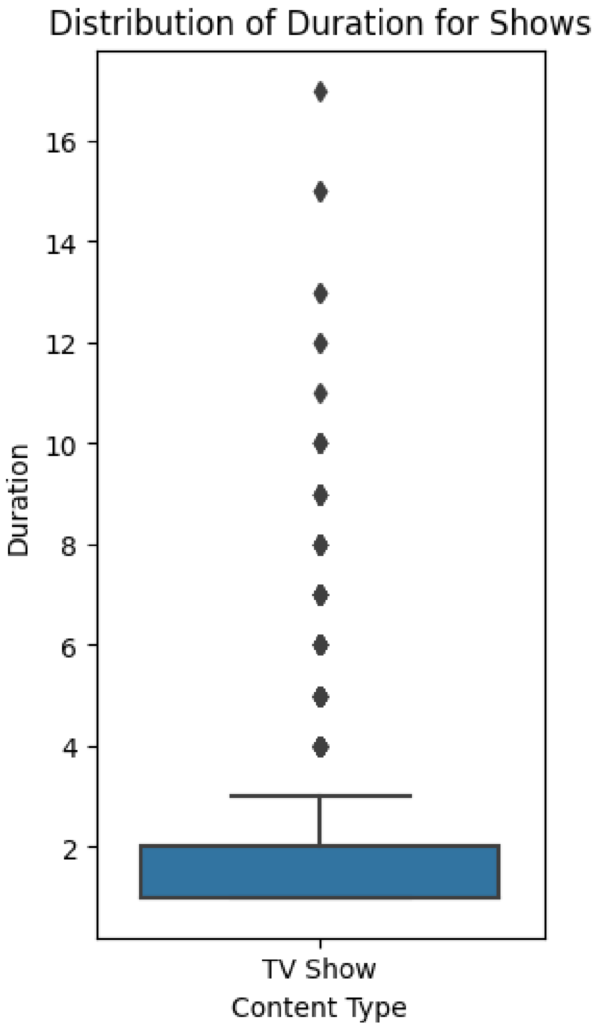
```
plt.figure(figsize=(10, 6))
sns.boxplot(data=netflix_movies_df, x='type', y='duration')
plt.xlabel('Content Type')
plt.ylabel('Duration')
plt.title('Distribution of Duration for Movies')
plt.show()
```

```
<ipython-input-279-03847279cf44>:2: 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
```



[280]:
```
netflix_shows_df = netflix_df[netflix_df.type.str.contains("TV Show")]
netflix_shows_df['duration'] = netflix_shows_df['duration'].str.extract('(\d+)',
expand=False).astype(int)
# Creating a boxplot for movie duration
plt.figure(figsize=(3, 6))
sns.boxplot(data=netflix_shows_df, x='type', y='duration')
plt.xlabel('Content Type')
plt.ylabel('Duration')
plt.title('Distribution of Duration for Shows')
plt.show()
```

```
<ipython-input-280-54aee4305ca4>:2: 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
```



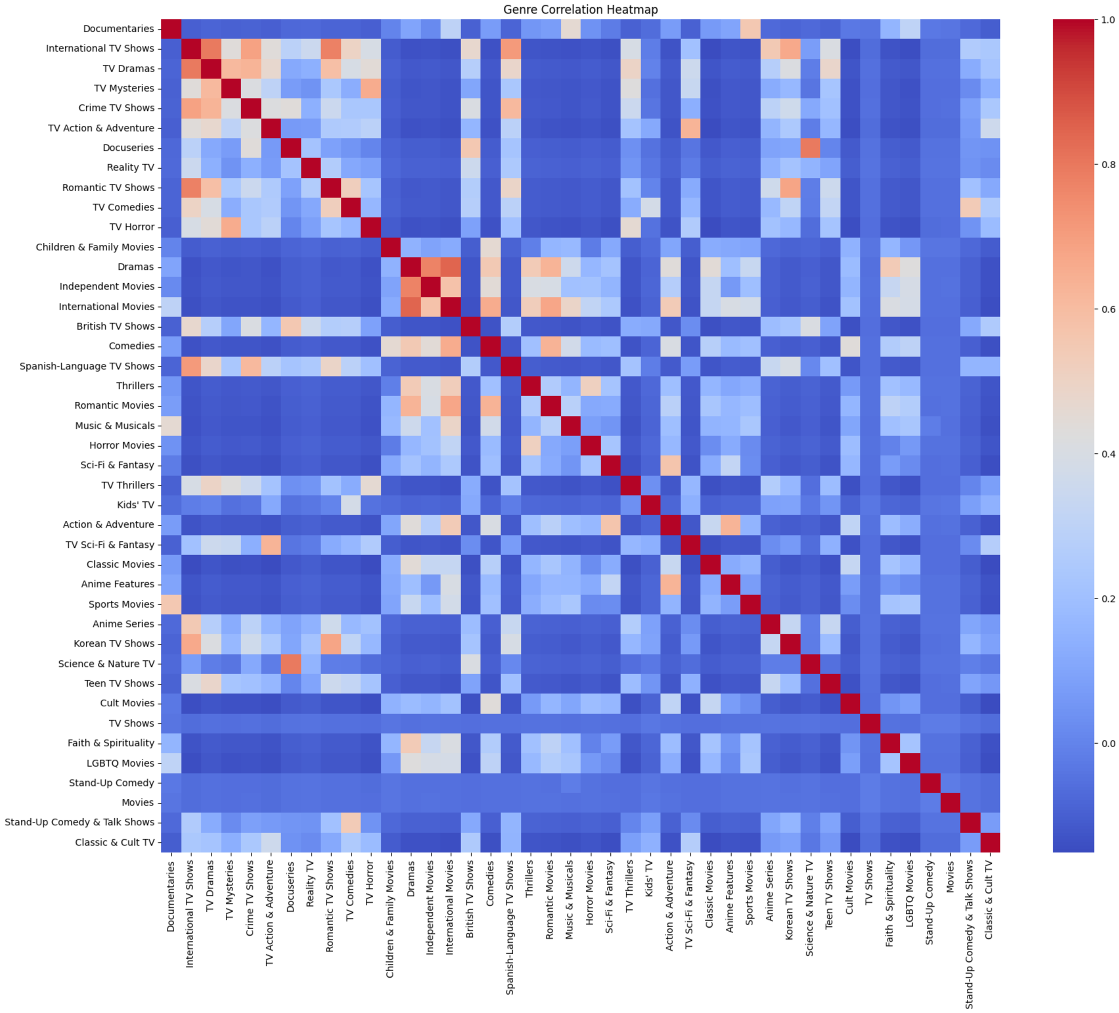
Analysing the movie box plot, we can see that most movies fall within a reasonable duration range, with few outliers exceedingly approximately 2.5 hours. This suggests that most movies on Netflix are designed to fit within a standard viewing time. For TV shows, the box plot reveals that most shows have one to four seasons, with very few outliers having longer durations. This aligns with the earlier trends, indicating that Netflix focuses on shorter series formats.

8 4.3 For correlation: Heatmaps, Pairplots

Genre Correlation Heatmap:

[281]:

```
# Extracting unique genres from the 'listed_in' column
genres = netflix_df['listed_in'].str.split(', ', expand=True).stack().unique()

# Create a new DataFrame to store the genre data
genre_data = pd.DataFrame(index=genres, columns=genres, dtype=float)

# Fill the genre data DataFrame with zeros
genre_data.fillna(0, inplace=True)

# Iterate over each row in the original DataFrame and update the genre data␣
 ↪DataFrame
for _, row in netflix_df.iterrows():
    listed_in = row['listed_in'].split(', ')
    for genre1 in listed_in:
        for genre2 in listed_in:
            genre_data.at[genre1, genre2] += 1

# Create a correlation matrix using the genre data
correlation_matrix = genre_data.corr()

# Create the heatmap
plt.figure(figsize=(20, 16))
sns.heatmap(correlation_matrix, annot=False, cmap='coolwarm')

# Customize the plot
plt.title('Genre Correlation Heatmap')
plt.xticks(rotation=90)
plt.yticks(rotation=0)

# Show the plot
plt.show()
```

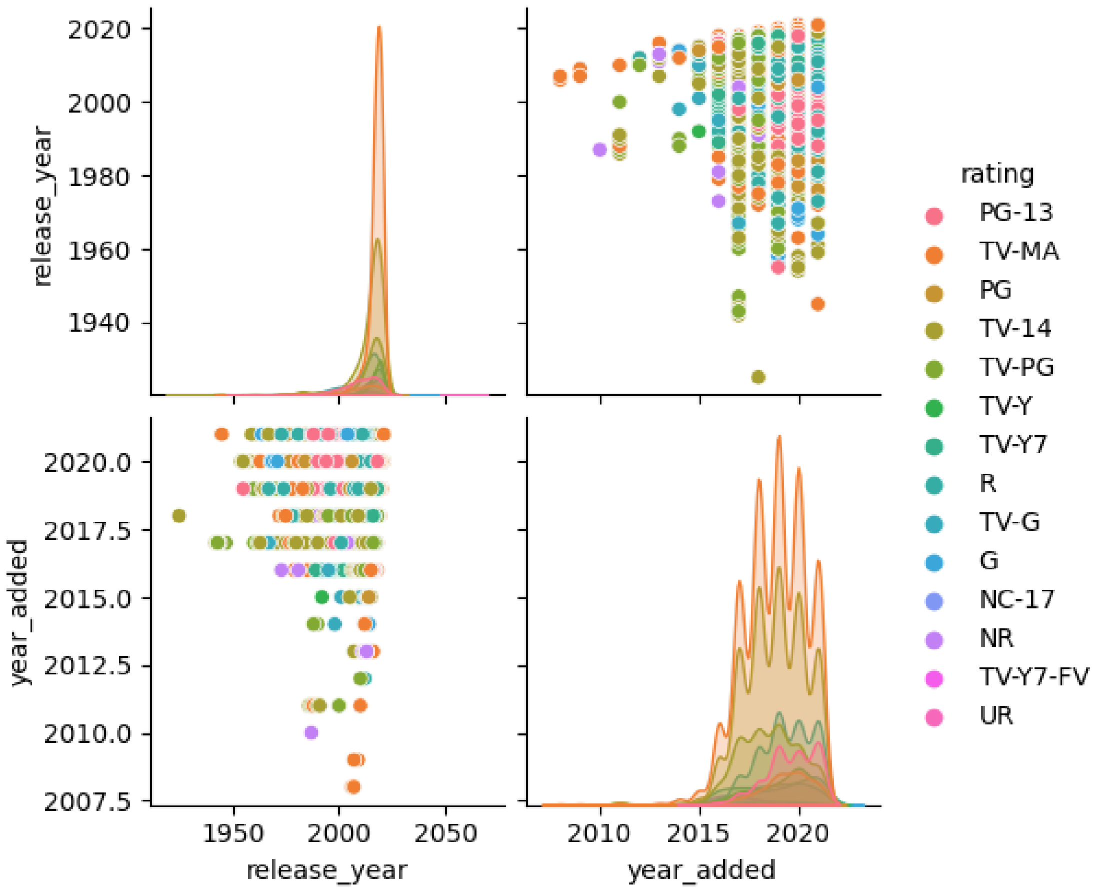


The heatmap demonstrates the correlation between different genres. By analysing the heatmap, we can identify strong positive correlations between specific genres, such as TV Dramas and International TV Shows, Romantic TV Shows, and International TV Shows.

Pairplots

A pairplot plot a pairwise relationships in a dataset. The pairplot function creates a grid of Axes such that each variable in data will by shared in the y-axis across a single row and in the x-axis across a single column.

```
[282]: sns.pairplot(netflix_df , hue = 'rating')
       plt.show()
```



9 5. Missing Value & Outlier check (Treatment optional)

What is an outlier?

In a random sampling from a population, an outlier is defined as an observation that deviates abnormally from the standard data. In simple words, an outlier is used to define those data values which are far away from the general values in a dataset. An outlier can be broken down into out-of-line data. For example, let us consider a row of data [10, 15, 22, 330, 30, 45, 60]. In this dataset, we can easily conclude that 330 is way off from the rest of the values in the dataset, thus 330 is an outlier. It was easy to figure out the outlier in such a small dataset, but when the dataset is huge, we need various methods to determine whether a certain value is an outlier or necessary information.

Why do we need to treat outliers?

Outliers can lead to vague or misleading predictions while using machine learning models. Specific models like linear regression, logistic regression, and support vector machines are susceptible to outliers. Outliers decrease the mathematical power of these models, and thus the output of the models becomes unreliable. However, outliers are highly subjective to the dataset. Some outliers may portray extreme changes in the data as well

Visual Detection

Box plots are a simple way to visualize data through quantiles and detect outliers. IQR(Interquartile Range) is the basic mathematics behind boxplots. The top and bottom whiskers can be understood as the boundaries of data, and any data lying outside it will be an outlier.

For categorical variable(s): Boxplot

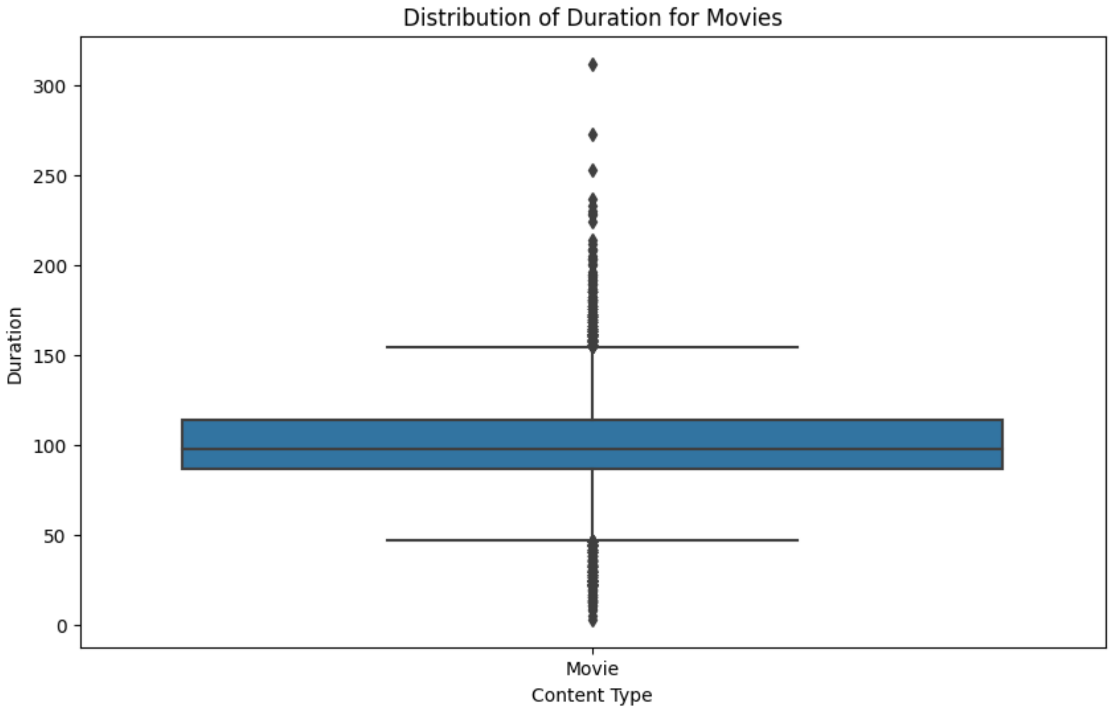
Duration Distribution for Movies and TV Shows

Analysing the duration distribution for movies and TV shows allows us to understand the typical length of content available on Netflix. We can create box plots to visualize these distributions and identify outliers or standard durations.

[283]:

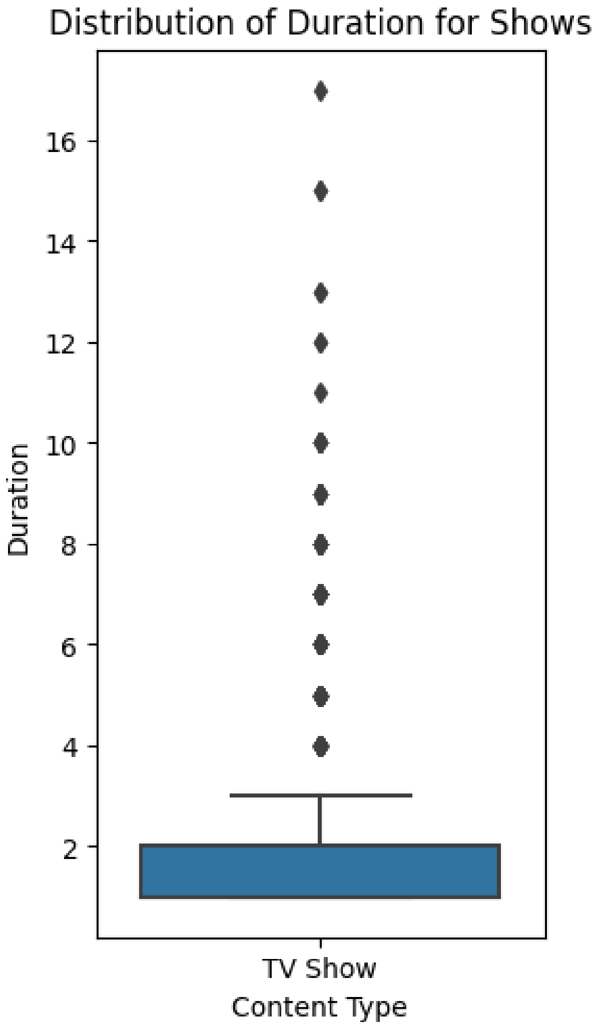
```python
netflix_movies_df = netflix_df[netflix_df.type.str.contains("Movie")]
netflix_movies_df['duration'] = netflix_movies_df['duration'].str.
  ↪extract('(\d+)',
expand=False).astype(int)
# Creating a boxplot for movie duration
plt.figure(figsize=(10, 6))
sns.boxplot(data=netflix_movies_df, x='type', y='duration')
plt.xlabel('Content Type')
plt.ylabel('Duration')
plt.title('Distribution of Duration for Movies')
plt.show()
```

```
<ipython-input-283-03847279cf44>:2: 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
```



[284]:
```
netflix_shows_df = netflix_df[netflix_df.type.str.contains("TV Show")]
netflix_shows_df['duration'] = netflix_shows_df['duration'].str.extract('(\d+)',
expand=False).astype(int)
# Creating a boxplot for movie duration
plt.figure(figsize=(3, 6))
sns.boxplot(data=netflix_shows_df, x='type', y='duration')
plt.xlabel('Content Type')
plt.ylabel('Duration')
plt.title('Distribution of Duration for Shows')
plt.show()
```

```
<ipython-input-284-54aee4305ca4>:2: 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
```



Analysing the movie box plot, we can see that most movies fall within a reasonable duration range, with few outliers exceedingly approximately 2.5 hours. This suggests that most movies on Netflix are designed to fit within a standard viewing time. For TV shows, the box plot reveals that most shows have one to four seasons, with very few outliers having longer durations. This aligns with the earlier trends, indicating that Netflix focuses on shorter series formats.

What are Missing values?

In a dataset, we often see the presence of empty cells, rows, and columns, also referred to as Missing values. They make the dataset inconsistent and unable to work on. Many machine learning algorithms return an error if parsed with a dataset containing null values. Detecting and treating missing values is essential while analyzing and formulating data for any purpose.

Detecting missing values

There are several ways to detect missing values in Python. isnull() function is widely used for the same purpose. **dataframe.isnull().values.any()** allows us to find whether we have any null values in the dataframe.

```
[285]: netflix = pd.read_csv("Business Case Netflix.csv")
```

```
[286]: print('\nColumns with missing value:')
       print(netflix.isnull().any())
```

```

Columns with missing value:
show_id         False
type            False
title           False
director         True
cast             True
country          True
date_added       True
release_year    False
rating           True
duration         True
listed_in       False
description     False
dtype: bool
```

From the info, we know that there are 8807 entries and 12 columns to work with for this EDA. There are a few columns that contain null values, "director," "cast," "country," "date_added," "ratng" and "duration".

dataframe.isnull().sum() this func on displays the total number of null values in each column.

```
[287]: netflix.T.apply(lambda x: x.isnull().sum(), axis = 1)
```

```
[287]: show_id            0
       type               0
       title              0
       director        2634
       cast             825
       country          831
       date_added        10
       release_year       0
       rating             4
       duration           3
       listed_in          0
       description        0
       dtype: int64
```

```
[288]: netflix.isnull().sum().sum()
```

```
[288]: 4307
```

There are a total of 4307 null values across the entire dataset with 2634 missing points under "director", 825 under "cast", 831 under "country", 11 under "date_added", 4 under "rating" and 3 under "duration". We will have to handle all null data points before we can dive into EDA and modelling.

Remedies to the outliers and missing values

Imputation is a treatment method for missing value by filling it in using certain techniques

Can use mean, mode, or use predictive modelling. In this case study, we will discuss the use of the fillna function from Pandas for this imputation. Drop rows containing missing values. Can use the dropna function from Pandas

[289]:

```
netflix_df.director.fillna("No Director", inplace=True)
netflix_df.cast.fillna("No Cast", inplace=True)
netflix_df.country.fillna("Country Unavailable", inplace=True)
netflix_df.dropna(subset=["date_added","duration", "rating"], inplace=True)
```

Check Missing Value

[290]:

```
netflix.isnull().any()
```

[290]:

```
show_id         False
type            False
title           False
director         True
cast             True
country          True
date_added       True
release_year    False
rating           True
duration         True
listed_in       False
description     False
dtype: bool
```

For missing values, the easiest way to get rid of them would be to delete the rows with the missing data. However, this wouldn't be beneficial to our EDA since the is a loss of information. Since "director", "cast", and "country" contain the majority of null values, we chose to treat each missing value is unavailable. The other two label "date_added"," duration" and "rating" contain an insignificant portion of the data so it drops from the dataset. Finally, we can see that there are no more missing values in the data frame.

10 6. Insight Based on Non-Graphical and Visual Analysis

Non-Graphical Analysis:

1. Directors Count: We have no director count as 4621 and the director which has the maximum count is Rajiv Chilaka with a count of 19 and the director which has least count are 5 of the

directors named Raymie Muzquiz, Stu Livingston, Joe Menendez, Eric Bross, Will Eisenberg, Mozez Singh which has only 1 count.

2. Top countries where netflix is popular: The country which has the maximum count is United States with a count of 2809 and the country with least count are Romania, Bulgaria, Hungary, Uruguay, Guatemala, France, Senegal, Belgium, Mexico, United States, Spain, Colombia, United Arab Emirates, Jordan with a single count of 1.
3. Movies and Tv shows added over time on netflix: The content which was maximum added on netflix in a year was 2018 with a count of 1146 and the minimum count of 1 was in the year 1959, 1925, 1961, 1947, 1966.
4. Movies and Tv shows count on netflix: The movie type content has the maximum count is 6126 and the tv show content has the count of 2664.
5. Cast who played maximum role in movies and tv shows on netflix: The no cast count has the maximum count which is 825 after that the David Attenborough has greater count of 19 and then least count with 1 are Sanjay Dutt, Arjun Kapoor, Kriti Sanon, Lika Berning, Bobby van Jaarsveld, Lisa Vicari, Dennis Mojen, Walid Al-Atiyat, Piotr Cyrwus, Mikołaj Kubacki, Anna Radwan, Vicky Kaushal, Sarah-Jane Dias and many more.

Visual Analysis:

1. Pie Plot: Range of attributes: In the Pie Plot we have shown the distribution of two content type, Tv show and Movie. Movies have 69.7 percentage and Tv shows have 30.3 percentage. So, clearly we can see through the analysis thrrough Pie Plot that movies are more in number than Tv shows on netflix. Distribution of the variables and relationship between them: Here in the Pie plot we have shown the relation between two of the content that is Tv show and Movie. The difference between two of them can be seen in the graphn. The red colour colour of the portion shows movie and the black colour of the portion shows Tv show. the percentage of movie is higher than tv show which says that netflix has more movie content than Tv show. Univariate analysis using a pie chart: The pie chart is a circular visual that displays the relative sizes here for two of the categories, Tv show & movie. Each slice of a pie chart represents a category and each category's size is proportional to its fraction of the total size of the data.
2. Bar plot: here we can see that top 10 categories by Movie/TV Show Count as the number of Movies/TV Shows which is allocated in y label and category which allocated in x label where Dramas, International Movies is maximum and Dramas, International Movies, Romantic Movies is minimum.
3. Box plots has provided a quick visual summary of the variability of values i.e, duration on y-axix in a dataset. They show the median, upper and lower quartiles, minimum and maximum values, and any outliers in the dataset. Outliers shows that the some of the movie content has occurrences passed to the maximum duration.
4. A Distplot or distribution plot, depicts the variation in the data distribution. Seaborn Distplot represents the overall distribution of continuous data variables. The Seaborn module along with the Matplotlib module is used to depict the distplot with different variations in it.
5. The histogram is a popular graphing tool. It is used to summarize discrete or continuous data that are measured on an interval scale. It is often used to illustrate the major features of the distribution of the data in a convenient form

11 7. BUSINESS INSIGHTS

With the help of Non-Graphic analysis and Visual Analysis, we have been able to learn about following factors:

1. **Quantity**: Our analysis revealed that Netflix had added more movies than TV shows, aligning with the expectation that movies dominate their content library.
2. **Content Addition**: July emerged as the month when Netflix adds the most content, closely followed by December, indicating a strategic approach to content release.
3. **Genre Correlation**: Strong positive associations were observed between various genres, such as TV dramas and international TV shows, romantic and international TV shows, and independent movies and dramas. These correlations provide insights into viewer preferences and content interconnections.
4. **Movie Lengths**: The analysis of movie durations indicated a peak around the 1960s, followed by a stabilization around 100 minutes, highlighting a trend in movie lengths over time.
5. **TV Show Episodes**: Most TV shows on Netflix have one season, suggesting a preference for shorter series among viewers.
6. **Common Themes**: Words like love, life, family, and adventure were frequently found in titles and descriptions, capturing recurring themes in Netflix content.
7. **Rating Distribution**: The distribution of ratings over the years offers insights into the evolving content landscape and audience reception.
8. **Data-Driven Insights**: Our data analysis journey showcased the power of data in unraveling the mysteries of Netflix's content landscape, providing valuable insights for viewers and content creators.
9. **Continued Relevance**: As the streaming industry evolves, understanding these patterns and trends becomes increasingly essential for navigating the dynamic landscape of Netflix and its vast library.
10. **Happy Streaming**: We hope this analysis has been an enlightening and entertaining journey into the world of Netflix, and we encourage you to explore the captivating stories within its ever-changing content offerings. Let the data guide your streaming adventures.

12 8. RECOMMENDATIONS

With the help of Non-Graphic analysis and Visual Analysis, we have been able to recommend below following points:

1. Netflix should focus some movies to be launched directly from netflix platform to reach more and more subscribers inspite of movies to be launched somewhere else which creates a less excitement to watch.
2. The netflix which releases a new content, there is nothing wrong in releasing such type, but sometimes we also want to watch something old movies beloved by multiple generations-one that we know won't disappoint. That's why some old times movies should also be listed of some most celebrated actors and actresses.

3. Some films are so great they belong to the ages. From war movies to biopics to iconic comedies, dramas and thrillers, these flicks should be dubbed in multiple languages, so that it can be watched in some small-small countries like Indonesia, Hong-Kong etc. to get subscribed with the users from there also by giving some offers to get people subscribed to the netflix platform.
4. Netflix has to focus on TV Shows also because there are people who will like to see tv shows rather than movies
5. We have seen most no of international movies genre so need to give priority to other geners like hooro,comedy..etc
6. Most of the movies released in Netflix is in a year 2019 so we need to go on increasing this value in order to attract people by showing that getting subscription is usefull as netflix is releasing more movies per year.
7. If Movies and TV show release in Netflix directly on the festival holidays, year end and week then it will Increase intrest of subscriber toward Netflix. ends which is to be mainly focussed.
8. Some movies can be released directly into Netflix which has some positive talk which may help in improving subscriptions.
9. Netflix Should focus on a Cast who has immense following and make use of it by doing a TV Shows or web series and Movies.
10. Advertisement in the country which has very less movies released should be increased and attract people of that country by making their native TV Shows.