netflix-casestudy

February 17, 2024

[1]:

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

[93]:

```
df_net = pd.read_csv('netflix.csv')
```

[94]:

```
df_net.head()
```

[94]:

```
  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…
```

0.0.1 Defining Problem statement and Analysing basic metrics and observing the shape and missing value of the data

Problem statement is to analyse the given netflix data and provide insights for it

[95]:

```
df_net.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
```

We can see that only one column is of numerical type i.e, release_year and rest are of object type.

[96]:

```
df_net.describe()
```

[96]:

```
       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
```

[97]:

```
df_net.shape
```

[97]:

```
(8807, 12)
```

We have 8807 rows and 12 attributes in the dataset.

[98]:

```
# Detecting no of missing values
df_net.isnull().sum()
```

[98]:

```
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
```

The director column has highest missing values(2389) and duration column has 1 missing value.

[99]:

```
df_net.nunique()
```

[99]:

```
show_id         8807
type               2
title           8804
director        4528
cast            7692
country          748
date_added      1766
release_year      74
rating            17
duration         220
listed_in        514
description     8775
dtype: int64
```

[103]:

```
# Converting date-time column from object to date time
df_net['date_added'] = pd.to_datetime(df_net['date_added'], errors='coerce')

df_net.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    8712 non-null   datetime64[ns]
 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: datetime64[ns](1), int64(1), object(10)
memory usage: 825.8+ KB
```

[104]:

```
df_net.describe()
```

[104]:

```
                          date_added  release_year
count                           8712   8807.000000
mean   2019-05-22 20:19:40.165289216   2014.180198
min              2008-01-01 00:00:00   1925.000000
25%              2018-04-19 00:00:00   2013.000000
50%              2019-07-12 00:00:00   2017.000000
75%              2020-08-26 00:00:00   2019.000000
max              2021-09-25 00:00:00   2021.000000
std                              NaN      8.819312
```

0.1 Non Graphical Analysis

[105]:

```
df_net['type'].value_counts()
```

[105]:

```
type
Movie      6131
TV Show    2676
Name: count, dtype: int64
```

[106]:

```
print(df_net['title'].unique())
print(df_net['title'].nunique())
```

```
['Dick Johnson Is Dead' 'Blood & Water' 'Ganglands' … 'Zombieland'
 'Zoom' 'Zubaan']
8804
```

We have 8807 unique titles in the dataset.

[107]:

```
df_net['director'].nunique()
```

[107]:

```
4528
```

We have 4528 unique directors in the dataset.

[108]:

```
df_net['date_added'].value_counts()
```

[108]:

```
date_added
2020-01-01    109
2019-11-01     89
2018-03-01     75
2019-12-31     74
2018-10-01     71
             …
2017-04-04      1
2017-03-28      1
2017-03-07      1
2017-02-27      1
2020-01-11      1
Name: count, Length: 1699, dtype: int64
```

[109]:

```
df_net['release_year'].value_counts()
```

[109]:

```
release_year
2018    1147
2017    1032
2019    1030
2020     953
2016     902
        …
1959       1
1925       1
1961       1
1947       1
1966       1
Name: count, Length: 74, dtype: int64
```

Most title have been added in the year 2018.

[110]:

```
df_net['rating'].value_counts()
```

[110]:

```
rating
TV-MA       3207
TV-14       2160
TV-PG        863
R            799
PG-13        490
```

```
TV-Y7         334
TV-Y          307
PG            287
TV-G          220
NR             80
G              41
TV-Y7-FV        6
NC-17           3
UR              3
74 min          1
84 min          1
66 min          1
Name: count, dtype: int64
```

Here we can see that the data is not clean and we need to clean the data before we start with the analysis. for ex some durations are inserted in the rating which we have to move to the duration column while cleaning the data.

```
[111]: df_net['duration'].unique()
```

```
[111]: array(['90 min', '2 Seasons', '1 Season', '91 min', '125 min',
              '9 Seasons', '104 min', '127 min', '4 Seasons', '67 min', '94 min',
              '5 Seasons', '161 min', '61 min', '166 min', '147 min', '103 min',
              '97 min', '106 min', '111 min', '3 Seasons', '110 min', '105 min',
              '96 min', '124 min', '116 min', '98 min', '23 min', '115 min',
              '122 min', '99 min', '88 min', '100 min', '6 Seasons', '102 min',
              '93 min', '95 min', '85 min', '83 min', '113 min', '13 min',
              '182 min', '48 min', '145 min', '87 min', '92 min', '80 min',
              '117 min', '128 min', '119 min', '143 min', '114 min', '118 min',
              '108 min', '63 min', '121 min', '142 min', '154 min', '120 min',
              '82 min', '109 min', '101 min', '86 min', '229 min', '76 min',
              '89 min', '156 min', '112 min', '107 min', '129 min', '135 min',
              '136 min', '165 min', '150 min', '133 min', '70 min', '84 min',
              '140 min', '78 min', '7 Seasons', '64 min', '59 min', '139 min',
              '69 min', '148 min', '189 min', '141 min', '130 min', '138 min',
              '81 min', '132 min', '10 Seasons', '123 min', '65 min', '68 min',
              '66 min', '62 min', '74 min', '131 min', '39 min', '46 min',
              '38 min', '8 Seasons', '17 Seasons', '126 min', '155 min',
              '159 min', '137 min', '12 min', '273 min', '36 min', '34 min',
              '77 min', '60 min', '49 min', '58 min', '72 min', '204 min',
              '212 min', '25 min', '73 min', '29 min', '47 min', '32 min',
              '35 min', '71 min', '149 min', '33 min', '15 min', '54 min',
              '224 min', '162 min', '37 min', '75 min', '79 min', '55 min',
              '158 min', '164 min', '173 min', '181 min', '185 min', '21 min',
              '24 min', '51 min', '151 min', '42 min', '22 min', '134 min',
              '177 min', '13 Seasons', '52 min', '14 min', '53 min', '8 min',
              '57 min', '28 min', '50 min', '9 min', '26 min', '45 min',
              '171 min', '27 min', '44 min', '146 min', '20 min', '157 min',
              '17 min', '203 min', '41 min', '30 min', '194 min', '15 Seasons',
              '233 min', '237 min', '230 min', '195 min', '253 min', '152 min',
              '190 min', '160 min', '208 min', '180 min', '144 min', '5 min',
              '174 min', '170 min', '192 min', '209 min', '187 min', '172 min',
              '16 min', '186 min', '11 min', '193 min', '176 min', '56 min',
              '169 min', '40 min', '10 min', '3 min', '168 min', '312 min',
              '153 min', '214 min', '31 min', '163 min', '19 min', '12 Seasons',
              nan, '179 min', '11 Seasons', '43 min', '200 min', '196 min',
              '167 min', '178 min', '228 min', '18 min', '205 min', '201 min',
              '191 min'], dtype=object)
```

[112]:
```
df_net['director'].value_counts()
```

[112]:
```
director
Rajiv Chilaka                     19
Raúl Campos, Jan Suter            18
Marcus Raboy                      16
Suhas Kadav                       16
Jay Karas                         14
                                  ..
Raymie Muzquiz, Stu Livingston     1
Joe Menendez                       1
Eric Bross                         1
Will Eisenberg                     1
Mozez Singh                        1
Name: count, Length: 4528, dtype: int64
```

Rajiv Chilaka has made the most movies(19) in the dataset.

[113]:
```
print("No of Unique values: ",df_net['listed_in'].nunique())
print(df_net['listed_in'].unique())
```

```
No of Unique values:  514
['Documentaries' 'International TV Shows, TV Dramas, TV Mysteries'
 'Crime TV Shows, International TV Shows, TV Action & Adventure'
 'Docuseries, Reality TV'
 'International TV Shows, Romantic TV Shows, TV Comedies'
 'TV Dramas, TV Horror, TV Mysteries' 'Children & Family Movies'
 'Dramas, Independent Movies, International Movies'
 'British TV Shows, Reality TV' 'Comedies, Dramas'
 'Crime TV Shows, Docuseries, International TV Shows'
 'Dramas, International Movies' 'Children & Family Movies, Comedies'
 'British TV Shows, Crime TV Shows, Docuseries' 'TV Comedies, TV Dramas'
 'Documentaries, International Movies'
 'Crime TV Shows, Spanish-Language TV Shows, TV Dramas' 'Thrillers'
 'International TV Shows, Spanish-Language TV Shows, TV Action & Adventure'
 'International TV Shows, TV Action & Adventure, TV Dramas'
```

```
'Comedies, International Movies'
'Comedies, International Movies, Romantic Movies'
'Docuseries, International TV Shows, Reality TV'
'Comedies, International Movies, Music & Musicals' 'Comedies'
'Horror Movies, Sci-Fi & Fantasy' 'TV Comedies'
'British TV Shows, International TV Shows, TV Comedies'
'International TV Shows, TV Dramas, TV Thrillers' "Kids' TV"
'Dramas, International Movies, Thrillers'
'Action & Adventure, Dramas, International Movies'
"Kids' TV, TV Comedies" 'Action & Adventure, Dramas'
"Kids' TV, TV Sci-Fi & Fantasy"
'Action & Adventure, Classic Movies, Dramas'
'Dramas, Horror Movies, Thrillers'
'Action & Adventure, Horror Movies, Thrillers' 'Action & Adventure'
'Dramas, Thrillers' 'International TV Shows, TV Dramas'
'International TV Shows, TV Dramas, TV Sci-Fi & Fantasy'
'Action & Adventure, Anime Features, International Movies' 'Reality TV'
'Docuseries, International TV Shows'
'Documentaries, International Movies, Sports Movies'
'International TV Shows, Reality TV, Romantic TV Shows'
'British TV Shows, Docuseries, International TV Shows'
'Anime Series, International TV Shows'
'Comedies, Dramas, International Movies'
'Crime TV Shows, TV Comedies, TV Dramas'
'Action & Adventure, Comedies, Dramas' "Anime Series, Kids' TV"
'International Movies, Thrillers' "Kids' TV, Korean TV Shows"
'Documentaries, Sports Movies' 'Sci-Fi & Fantasy, Thrillers'
'Dramas, International Movies, Romantic Movies'
'Documentaries, Music & Musicals'
"Kids' TV, TV Comedies, TV Sci-Fi & Fantasy" "British TV Shows, Kids' TV"
'Docuseries, Science & Nature TV' 'Children & Family Movies, Dramas'
"Kids' TV, TV Dramas, Teen TV Shows"
'Crime TV Shows, International TV Shows, Spanish-Language TV Shows'
'Docuseries, International TV Shows, Spanish-Language TV Shows' 'Dramas'
'Comedies, Romantic Movies' 'Dramas, Romantic Movies'
'Comedies, Dramas, Independent Movies'
'Crime TV Shows, TV Action & Adventure, TV Comedies'
'Children & Family Movies, Music & Musicals'
'Action & Adventure, Classic Movies, Cult Movies'
'International TV Shows, TV Action & Adventure, TV Comedies'
'Action & Adventure, Sci-Fi & Fantasy' 'Action & Adventure, Comedies'
'Classic Movies, Comedies, Dramas' 'Comedies, Cult Movies'
'Comedies, Cult Movies, Music & Musicals' 'Comedies, Music & Musicals'
'TV Shows' 'Action & Adventure, International Movies'
'Anime Series, International TV Shows, Teen TV Shows'
'Action & Adventure, Children & Family Movies, Cult Movies'
'Comedies, Dramas, Romantic Movies'
'Comedies, Cult Movies, Sci-Fi & Fantasy' 'Classic Movies, Dramas'
```

```
'Action & Adventure, Children & Family Movies, Comedies'
'Dramas, Faith & Spirituality' 'Documentaries, LGBTQ Movies'
'Action & Adventure, Classic Movies' 'Docuseries'
'International TV Shows, TV Comedies' 'Dramas, Independent Movies'
'Action & Adventure, Comedies, International Movies'
'International TV Shows, Spanish-Language TV Shows, TV Dramas'
'Crime TV Shows, International TV Shows, TV Dramas'
'Action & Adventure, Horror Movies, International Movies'
'Comedies, International Movies, Sci-Fi & Fantasy'
'Action & Adventure, International Movies, Music & Musicals'
'Dramas, International Movies, Music & Musicals'
'Horror Movies, International Movies' 'Reality TV, Teen TV Shows'
'Crime TV Shows, TV Dramas, TV Mysteries'
'International TV Shows, Reality TV'
'International TV Shows, TV Comedies, TV Dramas'
'Dramas, Independent Movies, Romantic Movies' 'Horror Movies'
'Documentaries, LGBTQ Movies, Sports Movies'
'Horror Movies, International Movies, Thrillers'
'Action & Adventure, Anime Features'
'TV Dramas, TV Mysteries, TV Sci-Fi & Fantasy'
'International TV Shows, Spanish-Language TV Shows, TV Comedies'
'Children & Family Movies, Comedies, Music & Musicals'
'Comedies, Independent Movies'
'Anime Series, International TV Shows, Romantic TV Shows'
'Classic Movies, Dramas, Independent Movies'
'International TV Shows, Romantic TV Shows, Spanish-Language TV Shows'
'International TV Shows, TV Dramas, Teen TV Shows' 'Stand-Up Comedy'
'Action & Adventure, Anime Features, Children & Family Movies'
'International TV Shows, Romantic TV Shows, TV Dramas'
'International Movies, Music & Musicals'
'TV Action & Adventure, TV Dramas, TV Mysteries'
'Horror Movies, Independent Movies, International Movies'
'Comedies, Cult Movies, International Movies'
'Classic Movies, Dramas, International Movies' 'Movies'
'Crime TV Shows, Docuseries'
'Children & Family Movies, Comedies, Sci-Fi & Fantasy'
'Anime Series, International TV Shows, TV Thrillers'
'Action & Adventure, Horror Movies, Sci-Fi & Fantasy'
'Classic Movies, Comedies, Cult Movies' 'TV Dramas, Teen TV Shows'
'Action & Adventure, Sci-Fi & Fantasy, Thrillers'
'Children & Family Movies, Comedies, Dramas' 'Dramas, Sports Movies'
'Action & Adventure, Dramas, Sci-Fi & Fantasy'
'Action & Adventure, Comedies, Cult Movies'
'Dramas, Independent Movies, Thrillers' 'TV Dramas, TV Sci-Fi & Fantasy'
'Action & Adventure, International Movies, Thrillers'
'British TV Shows, International TV Shows, Reality TV'
'TV Action & Adventure, TV Dramas, Teen TV Shows' 'Anime Series'
'Crime TV Shows, TV Action & Adventure, TV Sci-Fi & Fantasy'
```

```
'Crime TV Shows, International TV Shows, TV Comedies'
'Stand-Up Comedy & Talk Shows, TV Comedies'
'Classic & Cult TV, TV Action & Adventure, TV Dramas'
'Children & Family Movies, Sports Movies'
'TV Action & Adventure, TV Sci-Fi & Fantasy'
'Anime Series, Stand-Up Comedy & Talk Shows' 'TV Dramas'
'Anime Features, Children & Family Movies, International Movies'
'Classic & Cult TV, Crime TV Shows, International TV Shows'
'Crime TV Shows, International TV Shows, Romantic TV Shows'
'Horror Movies, LGBTQ Movies'
'Action & Adventure, Dramas, Romantic Movies'
'Documentaries, International Movies, Music & Musicals'
'TV Comedies, TV Dramas, Teen TV Shows'
'Children & Family Movies, Comedies, Sports Movies'
'Children & Family Movies, Dramas, International Movies'
'Comedies, Documentaries, International Movies'
'Romantic TV Shows, TV Dramas' 'Anime Series, TV Horror, TV Thrillers'
'International Movies, Romantic Movies'
'TV Action & Adventure, TV Dramas, TV Sci-Fi & Fantasy'
"Kids' TV, Korean TV Shows, TV Comedies"
'British TV Shows, Crime TV Shows, International TV Shows'
'Crime TV Shows, TV Horror, TV Mysteries'
'Docuseries, International TV Shows, Science & Nature TV'
'British TV Shows, International TV Shows, TV Dramas'
"Kids' TV, TV Action & Adventure, TV Sci-Fi & Fantasy"
'International Movies, Romantic Movies, Thrillers'
'Action & Adventure, Cult Movies, International Movies'
'Action & Adventure, Comedies, Sci-Fi & Fantasy'
"International TV Shows, Kids' TV, TV Mysteries"
'Action & Adventure, Thrillers'
'Dramas, Faith & Spirituality, International Movies'
'Action & Adventure, Classic Movies, Comedies'
'Action & Adventure, Comedies, Sports Movies'
'Action & Adventure, Children & Family Movies, Classic Movies'
'Action & Adventure, Children & Family Movies, Dramas'
'Horror Movies, Thrillers' 'Action & Adventure, Romantic Movies'
'Dramas, Romantic Movies, Sci-Fi & Fantasy'
'Dramas, Music & Musicals, Romantic Movies'
'Anime Series, Crime TV Shows, International TV Shows'
'Reality TV, Romantic TV Shows'
'International Movies, Music & Musicals, Romantic Movies'
'Reality TV, TV Action & Adventure, TV Mysteries'
'Crime TV Shows, TV Dramas'
'International TV Shows, Reality TV, Spanish-Language TV Shows'
'Crime TV Shows, TV Dramas, TV Thrillers' 'British TV Shows, Docuseries'
'International TV Shows, Korean TV Shows, TV Comedies'
'Action & Adventure, Anime Features, Classic Movies'
'TV Action & Adventure, TV Dramas, TV Horror'
```

```
 'Crime TV Shows, International TV Shows, TV Thrillers'
 'Anime Series, Crime TV Shows, TV Horror' 'Anime Features, Documentaries'
 'Comedies, Horror Movies'
 'International TV Shows, Spanish-Language TV Shows, Stand-Up Comedy & Talk
Shows'
 'Children & Family Movies, Documentaries, International Movies'
 'Romantic TV Shows, TV Comedies, TV Dramas'
 'Dramas, Faith & Spirituality, Romantic Movies'
 'Dramas, Independent Movies, LGBTQ Movies'
 'Comedies, Independent Movies, LGBTQ Movies'
 'Action & Adventure, Cult Movies, Sci-Fi & Fantasy'
 'Cult Movies, Horror Movies' 'Action & Adventure, Dramas, Sports Movies'
 'Anime Series, Romantic TV Shows, Teen TV Shows'
 'Dramas, International Movies, LGBTQ Movies'
 'Dramas, Romantic Movies, Thrillers'
 'Children & Family Movies, Dramas, Faith & Spirituality'
 'Dramas, International Movies, Sports Movies'
 'Action & Adventure, Horror Movies'
 'Documentaries, International Movies, LGBTQ Movies'
 'Dramas, Independent Movies, Sci-Fi & Fantasy'
 'Comedies, Independent Movies, International Movies'
 'Reality TV, TV Horror, TV Thrillers'
 'TV Action & Adventure, TV Horror, TV Sci-Fi & Fantasy'
 'International TV Shows, TV Horror, TV Sci-Fi & Fantasy'
 'Independent Movies, International Movies, Thrillers'
 'Independent Movies, Thrillers' 'Documentaries, Dramas'
 'Action & Adventure, Sports Movies'
 'Dramas, International Movies, Sci-Fi & Fantasy'
 'Comedies, Independent Movies, Romantic Movies'
 'Horror Movies, Romantic Movies, Sci-Fi & Fantasy'
 'International TV Shows, Stand-Up Comedy & Talk Shows'
 'Action & Adventure, Anime Features, Horror Movies'
 'Cult Movies, Dramas, Music & Musicals' 'TV Dramas, TV Thrillers'
 'Crime TV Shows, International TV Shows, Korean TV Shows'
 'TV Horror, TV Mysteries, TV Thrillers'
 'Comedies, Horror Movies, International Movies'
 'Crime TV Shows, Docuseries, TV Mysteries'
 'Comedies, International Movies, Sports Movies'
 'Classic Movies, Music & Musicals' 'Reality TV, TV Comedies, TV Horror'
 'Children & Family Movies, Faith & Spirituality, Music & Musicals'
 'International TV Shows, Korean TV Shows, Stand-Up Comedy & Talk Shows'
 'Dramas, Music & Musicals'
 'Docuseries, Science & Nature TV, TV Action & Adventure'
 "British TV Shows, Kids' TV, TV Dramas"
 'International TV Shows, Korean TV Shows, Romantic TV Shows'
 'Horror Movies, Independent Movies'
 "Anime Series, Kids' TV, TV Action & Adventure"
 'Comedies, Dramas, Music & Musicals' 'TV Horror, Teen TV Shows'
```

```
'Comedies, LGBTQ Movies, Thrillers'
'Docuseries, Reality TV, Science & Nature TV'
'Crime TV Shows, Spanish-Language TV Shows, TV Action & Adventure'
'Romantic TV Shows, Teen TV Shows' 'TV Comedies, Teen TV Shows'
'Romantic TV Shows, TV Dramas, Teen TV Shows'
'Children & Family Movies, Sci-Fi & Fantasy'
'Romantic TV Shows, TV Action & Adventure, TV Dramas'
'Comedies, International Movies, LGBTQ Movies' 'Dramas, Sci-Fi & Fantasy'
"Kids' TV, TV Thrillers"
'TV Action & Adventure, TV Comedies, TV Sci-Fi & Fantasy'
'British TV Shows, Romantic TV Shows, TV Dramas'
'Anime Series, International TV Shows, Spanish-Language TV Shows'
'Docuseries, TV Comedies' 'Comedies, Romantic Movies, Sports Movies'
'TV Action & Adventure, TV Comedies, TV Dramas'
'Children & Family Movies, Dramas, Sports Movies'
'Action & Adventure, Dramas, Independent Movies'
'Spanish-Language TV Shows, TV Dramas' 'Dramas, LGBTQ Movies'
'TV Horror, TV Mysteries, TV Sci-Fi & Fantasy'
'Action & Adventure, Dramas, Faith & Spirituality'
'International TV Shows, TV Mysteries, TV Thrillers'
'British TV Shows, Classic & Cult TV, International TV Shows'
'Action & Adventure, Comedies, Independent Movies' 'Music & Musicals'
"British TV Shows, Kids' TV, TV Comedies"
'Docuseries, Spanish-Language TV Shows'
'Dramas, Independent Movies, Sports Movies'
'TV Dramas, TV Mysteries, TV Thrillers'
'Comedies, LGBTQ Movies, Music & Musicals'
'International TV Shows, TV Action & Adventure, TV Mysteries'
"Kids' TV, TV Comedies, Teen TV Shows"
'International TV Shows, TV Dramas, TV Horror'
'Comedies, International Movies, Thrillers'
'Classic & Cult TV, TV Action & Adventure, TV Sci-Fi & Fantasy'
'International TV Shows, TV Horror, TV Mysteries'
'Children & Family Movies, Documentaries'
'Music & Musicals, Romantic Movies' 'Romantic Movies'
'Children & Family Movies, Classic Movies, Comedies'
'TV Action & Adventure, TV Dramas'
'Dramas, LGBTQ Movies, Romantic Movies'
'Children & Family Movies, Comedies, Romantic Movies'
'Comedies, Sports Movies' 'International Movies'
'International TV Shows, Romantic TV Shows, TV Mysteries'
'Stand-Up Comedy & Talk Shows'
'Action & Adventure, International Movies, Romantic Movies'
'Reality TV, TV Comedies' 'Cult Movies, Dramas, International Movies'
"Kids' TV, TV Dramas"
'Crime TV Shows, International TV Shows, TV Mysteries'
'Action & Adventure, Sci-Fi & Fantasy, Sports Movies'
'TV Dramas, TV Sci-Fi & Fantasy, TV Thrillers'
```

```
'Romantic TV Shows, TV Dramas, TV Sci-Fi & Fantasy'
'Docuseries, TV Sci-Fi & Fantasy' 'Anime Features, International Movies'
"British TV Shows, Classic & Cult TV, Kids' TV"
'British TV Shows, Reality TV, Romantic TV Shows'
'Documentaries, Faith & Spirituality, International Movies'
"Kids' TV, Reality TV, TV Dramas" 'LGBTQ Movies, Thrillers'
'TV Action & Adventure, TV Mysteries, TV Sci-Fi & Fantasy'
'Reality TV, Science & Nature TV'
"Kids' TV, TV Action & Adventure, TV Comedies"
'International TV Shows, Romantic TV Shows, TV Action & Adventure'
'Children & Family Movies, Dramas, Independent Movies'
'Comedies, Music & Musicals, Romantic Movies'
'International TV Shows, Korean TV Shows, Reality TV'
'Classic & Cult TV, TV Dramas, TV Sci-Fi & Fantasy'
'Anime Features, Children & Family Movies'
'Action & Adventure, International Movies, Sci-Fi & Fantasy'
'Crime TV Shows, TV Action & Adventure, TV Dramas'
'Classic & Cult TV, TV Action & Adventure, TV Horror'
'International TV Shows, Korean TV Shows, TV Dramas'
'International TV Shows, TV Action & Adventure, TV Horror'
'Action & Adventure, Comedies, Romantic Movies'
'International TV Shows, Korean TV Shows, TV Action & Adventure'
"Classic & Cult TV, Kids' TV, TV Action & Adventure"
'Anime Series, International TV Shows, TV Horror'
'International TV Shows, Korean TV Shows, TV Horror'
'Children & Family Movies, Comedies, International Movies'
'International Movies, Sci-Fi & Fantasy'
'International Movies, Sci-Fi & Fantasy, Thrillers'
'Children & Family Movies, Dramas, Romantic Movies'
'Anime Series, Romantic TV Shows' 'Comedies, Dramas, LGBTQ Movies'
'British TV Shows, International TV Shows, TV Action & Adventure'
'Docuseries, Science & Nature TV, TV Comedies'
'International TV Shows, Stand-Up Comedy & Talk Shows, TV Comedies'
'Children & Family Movies, Dramas, Music & Musicals'
'Action & Adventure, Independent Movies, International Movies'
'Action & Adventure, Children & Family Movies, Sci-Fi & Fantasy'
'Horror Movies, Independent Movies, Sci-Fi & Fantasy'
'TV Dramas, TV Sci-Fi & Fantasy, Teen TV Shows'
'Anime Features, International Movies, Sci-Fi & Fantasy'
'Dramas, Independent Movies, Music & Musicals'
"Kids' TV, TV Comedies, TV Dramas"
'Children & Family Movies, Documentaries, Sports Movies'
'Independent Movies, Sci-Fi & Fantasy, Thrillers'
'Anime Features, Music & Musicals, Sci-Fi & Fantasy'
'TV Comedies, TV Dramas, TV Sci-Fi & Fantasy'
'Crime TV Shows, TV Action & Adventure'
'Comedies, Faith & Spirituality, Romantic Movies'
"Kids' TV, TV Action & Adventure"
```

```
'Action & Adventure, Independent Movies'
'International TV Shows, Reality TV, TV Comedies'
'Docuseries, Reality TV, Teen TV Shows'
'Crime TV Shows, International TV Shows, Reality TV'
'Anime Series, Teen TV Shows'
'Crime TV Shows, Romantic TV Shows, TV Dramas'
'Anime Features, Romantic Movies'
'Horror Movies, Sci-Fi & Fantasy, Thrillers'
'International TV Shows, TV Comedies, TV Sci-Fi & Fantasy'
'International TV Shows, Romantic TV Shows'
'Anime Features, Music & Musicals'
'Anime Features, International Movies, Romantic Movies'
'International TV Shows, Romantic TV Shows, Teen TV Shows'
'Docuseries, Stand-Up Comedy & Talk Shows'
'Horror Movies, Independent Movies, Thrillers'
'TV Action & Adventure, TV Comedies, TV Horror'
'Documentaries, Stand-Up Comedy' "Kids' TV, Spanish-Language TV Shows"
"British TV Shows, Kids' TV, TV Thrillers"
"Kids' TV, TV Action & Adventure, TV Dramas"
'Anime Series, Crime TV Shows' 'Dramas, Sci-Fi & Fantasy, Thrillers'
'TV Comedies, TV Dramas, TV Horror'
'Children & Family Movies, Comedies, LGBTQ Movies'
'International TV Shows, TV Action & Adventure, TV Sci-Fi & Fantasy'
'Docuseries, TV Dramas'
'Horror Movies, International Movies, Romantic Movies'
'Crime TV Shows, Docuseries, Science & Nature TV'
'International Movies, Music & Musicals, Thrillers'
"Kids' TV, Spanish-Language TV Shows, Teen TV Shows"
'Comedies, Horror Movies, Independent Movies'
'Action & Adventure, International Movies, Sports Movies'
'Action & Adventure, Independent Movies, Sci-Fi & Fantasy'
'Horror Movies, LGBTQ Movies, Music & Musicals'
'Comedies, Music & Musicals, Sports Movies'
'TV Horror, TV Mysteries, Teen TV Shows' 'Romantic TV Shows, TV Comedies'
"Kids' TV, Reality TV, Science & Nature TV"
'International Movies, Romantic Movies, Sci-Fi & Fantasy'
'TV Comedies, TV Horror, TV Thrillers' 'TV Action & Adventure'
'International TV Shows, Spanish-Language TV Shows, TV Horror'
'Crime TV Shows, TV Action & Adventure, TV Thrillers'
'Music & Musicals, Stand-Up Comedy' 'British TV Shows, TV Comedies'
'TV Comedies, TV Sci-Fi & Fantasy, Teen TV Shows'
'TV Comedies, TV Sci-Fi & Fantasy'
'Romantic TV Shows, Spanish-Language TV Shows, TV Comedies'
'Crime TV Shows, International TV Shows, TV Sci-Fi & Fantasy'
'British TV Shows, International TV Shows, Romantic TV Shows'
"Crime TV Shows, Kids' TV"
'Horror Movies, International Movies, Sci-Fi & Fantasy'
'TV Comedies, TV Mysteries'
```

```
'Cult Movies, Horror Movies, Independent Movies'
'British TV Shows, Docuseries, TV Comedies' 'Comedies, Documentaries'
'Reality TV, Science & Nature TV, TV Action & Adventure'
'TV Comedies, TV Dramas, TV Mysteries'
'Crime TV Shows, TV Comedies, Teen TV Shows'
"Docuseries, Kids' TV, Science & Nature TV"
'Reality TV, Spanish-Language TV Shows'
'Action & Adventure, Anime Features, Sci-Fi & Fantasy'
"Crime TV Shows, Kids' TV, TV Comedies"
'Dramas, Faith & Spirituality, Independent Movies'
'Documentaries, Faith & Spirituality'
'British TV Shows, International TV Shows, Stand-Up Comedy & Talk Shows'
'Comedies, Dramas, Faith & Spirituality' 'Classic & Cult TV, TV Comedies'
'Dramas, Romantic Movies, Sports Movies'
'Stand-Up Comedy & Talk Shows, TV Mysteries, TV Sci-Fi & Fantasy'
'TV Sci-Fi & Fantasy, TV Thrillers'
'Comedies, Independent Movies, Music & Musicals'
'Comedies, Cult Movies, Independent Movies'
'Documentaries, Dramas, International Movies'
'British TV Shows, TV Horror, TV Thrillers'
'British TV Shows, Docuseries, Science & Nature TV'
'Children & Family Movies, Comedies, Cult Movies' 'Sports Movies'
'Sci-Fi & Fantasy' 'Comedies, LGBTQ Movies'
'Comedies, Independent Movies, Thrillers'
'Classic Movies, Cult Movies, Dramas'
'British TV Shows, TV Comedies, TV Dramas'
'Action & Adventure, Children & Family Movies, Independent Movies'
'Action & Adventure, Documentaries, International Movies'
'Children & Family Movies, Independent Movies'
'Comedies, Cult Movies, Dramas'
'International TV Shows, TV Horror, TV Thrillers'
'Classic Movies, Thrillers' 'Crime TV Shows, TV Dramas, TV Horror'
'British TV Shows, Docuseries, Reality TV'
'Documentaries, LGBTQ Movies, Music & Musicals'
'Classic Movies, Dramas, Romantic Movies'
'Crime TV Shows, Romantic TV Shows, Spanish-Language TV Shows'
'Classic Movies, Cult Movies, Horror Movies'
'Anime Series, Crime TV Shows, TV Thrillers'
'Children & Family Movies, Classic Movies'
'Classic Movies, Comedies, International Movies'
'Comedies, Sci-Fi & Fantasy' 'Action & Adventure, Cult Movies, Dramas'
'Documentaries, Faith & Spirituality, Music & Musicals'
'British TV Shows, Classic & Cult TV, TV Comedies'
'International Movies, Sports Movies' 'International TV Shows'
"Classic & Cult TV, Kids' TV, Spanish-Language TV Shows"
'Romantic TV Shows, Spanish-Language TV Shows, TV Dramas'
'Children & Family Movies, Comedies, Faith & Spirituality'
'British TV Shows, Crime TV Shows, TV Dramas'
```

```
 'Classic Movies, Dramas, Music & Musicals'
 'Cult Movies, Horror Movies, Thrillers'
 'Action & Adventure, Classic Movies, Sci-Fi & Fantasy'
 'TV Action & Adventure, TV Comedies'
 'Classic Movies, Comedies, Music & Musicals' 'Independent Movies'
 'Documentaries, Horror Movies'
 'Classic & Cult TV, TV Horror, TV Mysteries'
 'Comedies, Faith & Spirituality, International Movies'
 'Dramas, Horror Movies, Sci-Fi & Fantasy'
 'British TV Shows, TV Dramas, TV Sci-Fi & Fantasy'
 'Comedies, Cult Movies, Horror Movies'
 'Comedies, Cult Movies, Sports Movies' 'Classic Movies, Documentaries'
 'Action & Adventure, Faith & Spirituality, Sci-Fi & Fantasy'
 'Action & Adventure, Children & Family Movies'
 'International TV Shows, Reality TV, TV Action & Adventure'
 'Docuseries, Science & Nature TV, TV Dramas' 'Anime Features'
 'Action & Adventure, Horror Movies, Independent Movies'
 'Action & Adventure, Classic Movies, International Movies'
 'Cult Movies, Independent Movies, Thrillers'
 'Crime TV Shows, TV Comedies'
 'Classic Movies, Cult Movies, Documentaries'
 "Classic & Cult TV, Kids' TV, TV Comedies"
 'Classic Movies, Dramas, LGBTQ Movies'
 'Classic Movies, Dramas, Sports Movies' 'Action & Adventure, Cult Movies'
 'Action & Adventure, Comedies, Music & Musicals'
 'Classic Movies, Horror Movies, Thrillers'
 'Classic Movies, Comedies, Independent Movies'
 'Children & Family Movies, Classic Movies, Dramas'
 'Dramas, Faith & Spirituality, Sports Movies'
 'Classic Movies, Comedies, Romantic Movies'
 'Dramas, Horror Movies, Music & Musicals'
 'Classic Movies, Independent Movies, Thrillers'
 'Children & Family Movies, Faith & Spirituality'
 'Classic Movies, Comedies, Sports Movies'
 'Comedies, Dramas, Sports Movies'
 'Action & Adventure, Romantic Movies, Sci-Fi & Fantasy'
 'Classic & Cult TV, TV Sci-Fi & Fantasy'
 'Comedies, Cult Movies, LGBTQ Movies'
 'Comedies, Horror Movies, Sci-Fi & Fantasy'
 'Action & Adventure, Comedies, Horror Movies'
 'Classic & Cult TV, Crime TV Shows, TV Dramas'
 'Action & Adventure, Documentaries, Sports Movies'
 'International Movies, LGBTQ Movies, Romantic Movies'
 'Cult Movies, Dramas, Thrillers']
```

[114]:

```python
print("No of Unique Countries: ",df_net['country'].nunique())
print(df_net['country'].unique())
```

```
No of Unique Countries:  748
['United States' 'South Africa' nan 'India'
 'United States, Ghana, Burkina Faso, United Kingdom, Germany, Ethiopia'
 'United Kingdom' 'Germany, Czech Republic' 'Mexico' 'Turkey' 'Australia'
 'United States, India, France' 'Finland' 'China, Canada, United States'
 'South Africa, United States, Japan' 'Nigeria' 'Japan'
 'Spain, United States' 'France' 'Belgium' 'United Kingdom, United States'
 'United States, United Kingdom' 'France, United States' 'South Korea'
 'Spain' 'United States, Singapore' 'United Kingdom, Australia, France'
 'United Kingdom, Australia, France, United States'
 'United States, Canada' 'Germany, United States'
 'South Africa, United States' 'United States, Mexico'
 'United States, Italy, France, Japan'
 'United States, Italy, Romania, United Kingdom'
 'Australia, United States' 'Argentina, Venezuela'
 'United States, United Kingdom, Canada' 'China, Hong Kong' 'Russia'
 'Canada' 'Hong Kong' 'United States, China, Hong Kong'
 'Italy, United States' 'United States, Germany'
 'United Kingdom, Canada, United States' ', South Korea' 'Ireland'
 'India, Nepal' 'New Zealand, Australia, France, United States' 'Italy'
 'Italy, Brazil, Greece' 'Argentina' 'Jordan' 'Colombia'
 'United States, Japan' 'Belgium, United Kingdom'
 'Switzerland, United Kingdom, Australia' 'Israel, United States'
 'Canada, United States' 'Brazil' 'Argentina, Spain' 'Taiwan'
 'United States, Nigeria' 'Bulgaria, United States'
 'Spain, United Kingdom, United States' 'United States, China'
 'United States, France' 'Spain, France, United Kingdom, United States'
 ', France, Algeria' 'Poland' 'Germany'
 'France, Israel, Germany, United States, United Kingdom' 'New Zealand'
 'Saudi Arabia' 'Thailand' 'Indonesia' 'Egypt, Denmark, Germany'
 'United States, Switzerland' 'Hong Kong, Canada, United States'
 'Kuwait, United States' 'France, Canada, United States, Spain'
 'France, Netherlands, Singapore' 'France, Belgium'
 'Ireland, United States, United Kingdom' 'Egypt' 'Malaysia' 'Israel'
 'Australia, New Zealand' 'United Kingdom, Germany' 'Belgium, Netherlands'
 'South Korea, Czech Republic' 'Australia, Germany' 'Vietnam'
 'United Kingdom, Belgium' 'United Kingdom, Australia, United States'
 'France, Japan, United States'
 'United Kingdom, Germany, Spain, United States'
 'United Kingdom, United States, France, Italy'
 'United States, Germany, Canada'
 'United States, France, Italy, United Kingdom'
 'United States, United Kingdom, Germany, Hungary'
 'United States, New Zealand' 'Sweden' 'China' 'Lebanon' 'Romania'
 'Finland, Germany' 'Lebanon, Syria' 'Philippines' 'Iceland' 'Denmark'
 'United States, India' 'Philippines, Singapore, Indonesia'
 'China, United States, Canada' 'Lebanon, United Arab Emirates'
 'Canada, United States, Denmark' 'United Arab Emirates'
 'Mexico, France, Colombia' 'Netherlands' 'Germany, United States, France'
 'United States, Bulgaria'
 'United Kingdom, France, Germany, United States' 'Norway, Denmark'
 'Syria, France, Lebanon, Qatar' 'United States, Czech Republic'
 'Mauritius' 'Canada, South Africa' 'Austria' 'Mexico, Brazil'
 'Germany, France' 'Mexico, United States'
 'United Kingdom, France, Spain, United States' 'United States, Australia'
 'United States, United Kingdom, France' 'United States, Russia'
 'United States, United Kingdom, New Zealand' 'Australia, United Kingdom'
 'Canada, Nigeria, United States'
 'France, United States, United Kingdom, Canada' 'France, United Kingdom'
 'India, United Kingdom' 'Canada, United States, Mexico'
 'United Kingdom, Germany, United States'
 'Czech Republic, United Kingdom, United States' 'China, United Kingdom'
 'Italy, United Kingdom' 'China, Taiwan'
 'United States, Brazil, Japan, Spain, India'
 'United States, China, United Kingdom' 'Cameroon'
 'Lebanon, Palestine, Denmark, Qatar' 'Japan, United States'
 'Uruguay, Germany' 'Egypt, Saudi Arabia'
 'United Kingdom, France, Poland, Germany, United States'
 'Ireland, Switzerland, United Kingdom, France, United States'
 'United Kingdom, South Africa, France'
 'Ireland, United Kingdom, France, Germany' 'Russia, United States'
 'United Kingdom, United States, France' 'United Kingdom,'
 'United States, India, United Kingdom' 'Kenya' 'Spain, Argentina'
 'India, United Kingdom, France, Qatar' 'Belgium, France'
 'Argentina, Chile' 'United States, Thailand' 'Chile, Brazil'
 'United States, Colombia' 'Canada, United States, United Kingdom'
 'Uruguay' 'Luxembourg' 'United States, Cambodia, Romania' 'Bangladesh'
 'Spain, Belgium, United States'
 'United Kingdom, United States, Australia'
 'Canada, United States, France' 'Portugal, United States'
 'Portugal, Spain' 'India, United States' 'United Kingdom, Ireland'
 'United Kingdom, Spain, United States' 'Hungary, United States'
 'United States, South Korea' 'Canada, United States, Cayman Islands'
 'India, France' 'France, Canada' 'Canada, Hungary, United States'
 'Norway' 'Canada, United Kingdom, United States'
 'United Kingdom, Germany, France, United States' 'Denmark, United States'
 'Senegal' 'France, Algeria'
 'United Kingdom, Finland, Germany, United States, Australia, Japan, France,
Ireland'
 'Philippines, Canada, United Kingdom, United States'
 'Ireland, France, Iceland, United States, Mexico, Belgium, United Kingdom, Hong
Kong'
 'Singapore' 'Kuwait' 'United States, France, Serbia'
 'United States, Italy' 'Spain, Italy'
 'United States, Ireland, United Kingdom, India'
 'United Kingdom, Singapore' 'Hong Kong, United States'
 'United States, Malta, France, United Kingdom'
 'United States, China, Canada' 'Canada, United States, Ireland'
 'Lebanon, Canada, France' 'Japan, Canada, United States'
 'Spain, France, Canada' 'Denmark, Singapore, Canada, United States'
 'United States, France, Denmark' 'United States, China, Colombia'
 'Spain, Thailand, United States' 'Mexico, Spain'
 'Ireland, Luxembourg, Belgium' 'China, United States' 'Canada, Belgium'
 'Canada, United Kingdom'
 'Lebanon, United Arab Emirates, France, Switzerland, Germany'
 'France, Belgium, Italy' 'Lebanon, United States, United Arab Emirates'
 'Lebanon, France' 'France, Lebanon' 'France, Lebanon, United Kingdom'
 'France, Norway, Lebanon, Belgium'
 'Sweden, Czech Republic, United Kingdom, Denmark, Netherlands'
 'United States, United Kingdom, India' 'Indonesia, Netherlands'
 'Turkey, South Korea' 'Serbia, United States' 'Namibia'
 'United Kingdom, Kenya' 'United Kingdom, France, Germany, Spain'
 'United Kingdom, France, United States, Belgium, Luxembourg, China, Germany'
 'Thailand, United States' 'United States, France, Canada, Belgium'
 'United Kingdom, China' 'Germany, China, United Kingdom'
 'Australia, New Zealand, United States'
 'Hong Kong, Iceland, United States' 'France, Australia, Germany'
 'United States, Belgium, Canada, France' 'South Africa, Angola'
 'United States, Philippines'
 'United States, United Kingdom, Canada, China'
 'United States, Canada, United Kingdom' 'Turkey, United States'
 'Peru, Germany, Norway' 'Mozambique' 'Brazil, France'
 'China, Spain, South Korea, United States' 'Spain, Germany'
 'Hong Kong, China' 'France, Belgium, Luxembourg, Cambodia,'
 'United Kingdom, Australia' 'Belarus' 'Indonesia, United Kingdom'
 'Switzerland, France, Belgium, United States' 'Ghana'
 'Spain, France, Canada, United States' 'Chile, Italy'
 'United Kingdom, Nigeria' 'Chile' 'France, Egypt' 'Egypt, France'
 'France, Brazil, Spain, Belgium' 'Egypt, Algeria'
 'Canada, South Korea, United States' 'Nigeria, United Kingdom'
 'United States, France, Canada' 'Poland, United States'
 'United Arab Emirates, Jordan, Lebanon, Saudi Arabia'
 'United States, Mexico, Spain, Malta'
 'Saudi Arabia, United Arab Emirates' 'Zimbabwe'
 'United Kingdom, Germany, United Arab Emirates, New Zealand'
 'Romania, United States' 'Canada, Nigeria'
 'Saudi Arabia, Netherlands, Germany, Jordan, United Arab Emirates, United
States'
 'United Kingdom, Spain' 'Finland, France'
 'United Kingdom, Germany, United States, France'
 'India, United Kingdom, China, Canada, Japan, South Korea, United States'
 'Italy, United Kingdom, France' 'United States, Mexico, Colombia'
 'Turkey, India' 'Italy, Turkey' 'United Kingdom, United States, Japan'
 'France, Belgium, United States' 'Puerto Rico, United States, Colombia'
```

```
'Uruguay, Argentina' 'United States, United Kingdom, Japan'
'United States, Argentina' 'United Kingdom, Italy'
'Ireland, United Kingdom'
'United Kingdom, France, Belgium, Canada, United States'
'Netherlands, Germany, Denmark, United Kingdom' 'Hungary'
'Austria, Germany' 'Taiwan, China'
'United Kingdom, United States, Ireland' 'South Korea, United States'
'Brazil, United Kingdom' 'Pakistan, United States'
'Romania, France, Switzerland, Germany' 'Romania, United Kingdom'
'France, Malta, United States' 'Cyprus'
'United Kingdom, France, Belgium, Ireland, United States'
'United States, Norway, Canada' 'Kenya, United States'
'France, South Korea, Japan, United States' 'Taiwan, Malaysia'
'Uruguay, Argentina, Germany, Spain'
'United States, United Kingdom, France, Germany, Japan'
'United States, France, Japan' 'United Kingdom, France, United States'
'Spain, France, United States' 'Indonesia, South Korea, Singapore'
'United States, Spain' 'Netherlands, Germany, Italy, Canada'
'Spain, Germany, Denmark, United States' 'Norway, Sweden'
'South Korea, Canada, United States, China' 'Argentina, Uruguay, Serbia'
'France, Japan' 'Mauritius, South Africa' 'United States, Poland'
'United Kingdom, United States, Germany, Denmark, Belgium, Japan'
'India, Germany' 'India, United Kingdom, Canada, United States'
'Philippines, United States' 'Romania, Bulgaria, Hungary'
'Uruguay, Guatemala' 'France, Senegal, Belgium' 'United Kingdom, Canada'
'Mexico, United States, Spain, Colombia' 'Canada, Norway'
'Singapore, United States' 'Finland, Germany, Belgium'
'United Kingdom, France' 'United States, Chile'
'United Kingdom, Japan, United States' 'Spain, United Kingdom'
'Argentina, United States, Mexico' 'United States, South Korea, Japan'
'Canada, Australia' 'United Kingdom, Hungary, Australia' 'Italy, Belgium'
'United States, United Kingdom, Germany' 'Switzerland'
'Singapore, Malaysia'
'France, Belgium, Luxembourg, Romania, Canada, United States'
'South Africa, Nigeria' 'Spain, France' 'United Kingdom, Hong Kong'
'Pakistan' 'Brazil, United States'
'Denmark, Brazil, France, Portugal, Sweden' 'India, Turkey'
'Malaysia, Singapore, Hong Kong' 'Philippines, Singapore'
'Australia, Canada' 'Taiwan, China, France, United States'
'Germany, Italy' 'Colombia, Peru, United Kingdom'
'Thailand, China, United States' 'Argentina, United States'
'Sweden, United States' 'Uruguay, Spain, Mexico'
'France, Luxembourg, Canada' 'Denmark, Spain' 'Chile, Argentina'
'United Kingdom, Belgium, Sweden' 'Canada, Brazil' 'Italy, France'
'Canada, Germany' 'Pakistan, United Arab Emirates' 'Ghana, United States'
'Mexico, Finland' 'United Arab Emirates, United Kingdom, India'
'Netherlands, Belgium' 'United States, Taiwan'
'Austria, Iraq, United States' 'United Kingdom, Malawi'
```

```
'Paraguay, Argentina' 'United Kingdom, Russia, United States'
'India, Pakistan' 'Indonesia, Singapore' 'Spain, Belgium'
'Iceland, Sweden, Belgium' 'Croatia' 'Uruguay, Argentina, Spain'
'United Kingdom, Ireland, United States'
'Canada, Germany, France, United States' 'United Kingdom, Japan'
'Norway, Denmark, Netherlands, Sweden' 'Hong Kong, China, United States'
'Ireland, Canada' 'Italy, Switzerland, France, Germany'
'Mexico, Netherlands' 'United States, Sweden' 'Germany, France, Russia'
'France, Iran, United States' 'United Kingdom, India'
'Russia, Poland, Serbia' 'Spain, Portugal' 'Peru' 'Mexico, Argentina'
'United Kingdom, Canada, United States, Cayman Islands'
'Indonesia, United States'
'United States, Israel, United Kingdom, Canada'
'Norway, Iceland, United States' 'Czech Republic, United States'
'United Kingdom, India, United States' 'United Kingdom, West Germany'
'India, Australia' 'United States,'
'Belgium, United Kingdom, United States' 'India, Germany, Austria'
'United States, Brazil, South Korea, Mexico, Japan, Germany'
'Spain, Mexico' 'China, Japan' 'Argentina, France'
'China, United States, United Kingdom'
'France, Luxembourg, United States' 'China, United States, Australia'
'Colombia, Mexico' 'United States, Canada, Ireland' 'Chile, Peru'
'Argentina, Italy' 'Canada, Japan, United States'
'United Kingdom, Canada, United States, Germany'
'Italy, Switzerland, Albania, Poland' 'United States, Japan, Canada'
'Cambodia' 'Italy, United States, Argentina'
'Saudi Arabia, Syria, Egypt, Lebanon, Kuwait'
'United States, Canada, Indonesia, United Kingdom, China, Singapore'
'Spain, Colombia'
'United Kingdom, South Africa, Australia, United States' 'Bulgaria'
'Argentina, Brazil, France, Poland, Germany, Denmark'
'United Kingdom, Spain, United States, Germany' 'Philippines, Qatar'
'Netherlands, Belgium, Germany, Jordan'
'United Arab Emirates, United States' 'Norway, Germany, Sweden'
'South Korea, China' 'Georgia' 'Soviet Union, India'
'Australia, United Arab Emirates' 'Canada, Germany, South Africa'
'South Korea, China, United States' 'India, Soviet Union' 'India, Mexico'
'Georgia, Germany, France' 'United Arab Emirates, Romania'
'India, Malaysia' 'Germany, Jordan, Netherlands'
'Turkey, France, Germany, Poland' 'Greece, United States'
'France, United Kingdom, United States' 'Norway, Germany'
'France, Morocco' 'Cambodia, United States' 'United States, Denmark'
'United States, Colombia, Mexico'
'United Kingdom, Italy, Israel, Peru, United States'
'Argentina, Uruguay, Spain, France'
'United Kingdom, France, United States, Belgium'
'France, Canada, China, Cambodia'
'United Kingdom, France, Belgium, United States' 'Chile, France'
```

```
 'Netherlands, United States' 'France, United Kingdom, India'
 'Czech Republic, Slovakia' 'Singapore, France' 'Spain, Switzerland'
 'United States, Australia, China' 'South Africa, United States, Germany'
 'United States, United Kingdom, Australia' 'Spain, Italy, Argentina'
 'Chile, Spain, Argentina, Germany' 'West Germany'
 'Austria, Czech Republic' 'Lebanon, Qatar'
 'United Kingdom, Jordan, Qatar, Iran' 'France, South Korea, Japan'
 'Israel, Germany, France' 'Canada, Japan, Netherlands'
 'United States, Hungary' 'France, Germany' 'France, Qatar'
 'United Kingdom, Germany, Canada' 'Ireland, South Africa'
 'Chile, United States, France' 'Belgium, France, Netherlands'
 'United Kingdom, Ukraine, United States'
 'Germany, Australia, France, China' 'Norway, United States'
 'United States, Bermuda, Ecuador'
 'United States, Hungary, Ireland, Canada'
 'United Kingdom, Egypt, United States'
 'United States, France, United Kingdom' 'Spain, Mexico, France'
 'United States, South Africa' 'Hong Kong, China, Singapore'
 'South Africa, China, United States' 'Denmark, France, Poland'
 'New Zealand, United Kingdom' 'Netherlands, Denmark, South Africa'
 'Iran, France' 'United Kingdom, United States, France, Germany'
 'Australia, France' 'Ireland, United Kingdom, United States'
 'United Kingdom, France, Germany' 'Canada, Luxembourg'
 'Brazil, Netherlands, United States, Colombia, Austria, Germany'
 'France, Canada, Belgium' 'Canada, France'
 'Bulgaria, United States, Spain, Canada' 'Sweden, Netherlands'
 'France, United States, Mexico'
 'Australia, United Kingdom, United Arab Emirates, Canada'
 'Australia, Armenia, Japan, Jordan, Mexico, Mongolia, New Zealand, Philippines,
South Africa, Sweden, United States, Uruguay'
 'India, Iran' 'France, Belgium, Spain'
 'Denmark, Sweden, Israel, United States' 'United States, Iceland'
 'United Kingdom, Russia' 'United States, Israel, Italy, South Africa'
 'Netherlands, Denmark, France, Germany' 'South Korea, Japan'
 'United Kingdom, Pakistan' 'France, New Zealand'
 'United Kingdom, Czech Republic, United States, Germany, Bahamas'
 'China, Germany, India, United States' 'Germany, Sri Lanka'
 'United States, India, Bangladesh' 'United States, Canada, France'
 'Brazil, France, Germany' 'Germany, United States, Hong Kong, Singapore'
 'France, Germany, Switzerland'
 'Germany, France, Luxembourg, United Kingdom, United States'
 'United Kingdom, Canada, Italy' 'Czech Republic, France'
 'Taiwan, Hong Kong, United States, China' 'Germany, Australia'
 'United Kingdom, Poland, United States' 'Denmark, Zimbabwe'
 'United Kingdom, South Africa' 'Finland, Sweden, Norway, Latvia, Germany'
 'South Africa, United States, New Zealand, Canada'
 'United States, Italy, United Kingdom, Liechtenstein'
 'Denmark, France, Belgium, Italy, Netherlands, United States, United Kingdom'
```

```
'United States, Australia, Mexico'
'United Kingdom, Czech Republic, Germany, United States'
'France, China, Japan, United States' 'United States, South Korea, China'
'Germany, Belgium' 'Pakistan, Norway, United States'
'United States, Canada, Belgium, United Kingdom' 'Venezuela'
'Canada, France, Italy, Morocco, United States' 'Canada, Spain, France'
'United States, Indonesia' 'Spain, France, Italy'
'United Arab Emirates, United States, United Kingdom'
'United Kingdom, Israel, Russia' 'Spain, Cuba' 'United States, Brazil'
'United States, France, Mexico' 'United States, Nicaragua'
'United Kingdom, United States, Spain, Germany, Greece, Canada'
'Italy, Canada, France' 'United Kingdom, Denmark, Canada, Croatia'
'Italy, Germany' 'United States, France, United Kingdom, Japan'
'United States, United Kingdom, Denmark, Sweden'
'United States, United Kingdom, Italy'
'United States, France, Canada, Spain' 'Russia, United States, China'
'United States, Canada, Germany' 'Ireland, United States'
'United States, United Arab Emirates' 'United States, Ireland'
'Ireland, United Kingdom, Italy, United States' 'Poland,'
'Slovenia, Croatia, Germany, Czech Republic, Qatar'
'Canada, United Kingdom, Netherlands' 'United States, Spain, Germany'
'India, Japan' 'China, South Korea, United States'
'United Kingdom, France, Belgium' 'Canada, Ireland, United States'
'United Kingdom, United States, Dominican Republic'
'United States, Senegal' 'Germany, United Kingdom, United States'
'South Africa, Germany, Netherlands, France'
'Canada, United States, United Kingdom, France, Luxembourg'
'Ireland, United States, France' 'Germany, United States, Canada'
'United Kingdom, Germany, Canada, United States'
'United States, France, Canada, Lebanon, Qatar'
'Netherlands, Belgium, United Kingdom, United States'
'France, Belgium, China, United States' 'United States, Chile, Israel'
'United Kingdom, Norway, Denmark, Germany, Sweden'
'Norway, Denmark, Sweden' 'China, India, Nepal'
'Colombia, Mexico, United States' 'United Kingdom, South Korea'
'Denmark, China' 'United States, Greece, Brazil' 'South Korea, France'
'United States, Australia, Samoa, United Kingdom'
'Germany, United Kingdom' 'Argentina, Chile, Peru' 'Turkey, Azerbaijan'
'Poland, West Germany' 'Germany, United States, Sweden' 'Canada, Spain'
'United States, Cambodia' 'United States, Greece'
'Norway, United Kingdom, France, Ireland' 'United Kingdom, Poland'
'Israel, Sweden, Germany, Netherlands' 'Switzerland, France'
'Italy, India' 'United States, Botswana'
'Chile, Argentina, France, Spain, United States'
'United States, India, South Korea, China'
'Denmark, Germany, Belgium, United Kingdom, France'
'Denmark, Germany, Belgium, United Kingdom, France, Sweden'
'France, Switzerland, Spain, United States, United Arab Emirates'
```

```
 'Brazil, India, China, United States'
 'Denmark, France, United States, Sweden' 'Australia, Iraq'
 'China, Morocco, Hong Kong' 'Canada, United States, Germany'
 'United Kingdom, Thailand' 'Venezuela, Colombia'
 'Colombia, United States' 'France, Germany, Czech Republic, Belgium'
 'Switzerland, Vatican City, Italy, Germany, France'
 'Portugal, France, Poland, United States'
 'United States, New Zealand, Japan'
 'United States, Netherlands, Japan, France' 'India, Switzerland'
 'Canada, India' 'United States, Morocco' 'Singapore, Japan, France'
 'Canada, Mexico, Germany, South Africa'
 'United Kingdom, United States, Canada'
 'Germany, France, United States, Canada, United Kingdom'
 'United States, Uruguay' 'India, Canada'
 'Ireland, Canada, United Kingdom, United States'
 'United States, Germany, Australia' 'Australia, France, Ireland'
 'Australia, India' 'United States, United Kingdom, Canada, Japan'
 'Sweden, United Kingdom, Finland' 'Hong Kong, Taiwan'
 'United States, United Kingdom, Spain, South Korea' 'Guatemala' 'Ukraine'
 'Italy, South Africa, West Germany, Australia, United States'
 'United States, Germany, United Kingdom, Australia'
 'Italy, France, Switzerland' 'Canada, France, United States'
 'Switzerland, United States' 'Thailand, Canada, United States'
 'China, Hong Kong, United States' 'United Kingdom, New Zealand'
 'Czech Republic, United Kingdom, France'
 'Australia, United Kingdom, Canada' 'Jamaica, United States'
 'Australia, United Kingdom, United States, New Zealand, Italy, France'
 'France, United States, Canada'
 'United Kingdom, France, Canada, Belgium, United States'
 'Denmark, United Kingdom, Sweden' 'United States, Hong Kong'
 'United States, Kazakhstan'
 'Argentina, France, United States, Germany, Qatar'
 'United States, Germany, United Kingdom'
 'United States, Germany, United Kingdom, Italy'
 'United States, New Zealand, United Kingdom' 'Finland, United States'
 'Spain, France, Uruguay' 'France, Canada, United States'
 'United States, Canada, China'
 'Ireland, Canada, Luxembourg, United States, United Kingdom, Philippines,
India'
 'United States, Czech Republic, United Kingdom' 'Israel, Germany'
 'Mexico, France'
 'Israel, Germany, Poland, Luxembourg, Belgium, France, United States'
 'Austria, United States' 'United Kingdom, Lithuania'
 'United States, Greece, United Kingdom'
 'United Kingdom, China, United States, India'
 'United States, Sweden, Norway' 'United Kingdom, United States, Morocco'
 'United States, United Kingdom, Morocco' 'Spain, Canada, United States'
 'United States, India, United Arab Emirates'
 'United Kingdom, Canada, France, United States' 'India, Germany, France'
 'Belgium, Ireland, Netherlands, Germany, Afghanistan'
 'France, Canada, Italy, United States, China'
 'Ireland, United Kingdom, Greece, France, Netherlands'
 'Denmark, Indonesia, Finland, Norway, United Kingdom, Israel, France, United
States, Germany, Netherlands'
 'New Zealand, United States'
 'United States, Australia, South Africa, United Kingdom'
 'United States, Germany, Mexico'
 'Somalia, Kenya, Sudan, South Africa, United States'
 'United States, Canada, Japan, Panama' 'United Kingdom, Spain, Belgium'
 'Serbia, South Korea, Slovenia'
 'Denmark, United Kingdom, South Africa, Sweden, Belgium'
 'Germany, Canada, United States'
 'Ireland, Canada, United States, United Kingdom'
 'New Zealand, United Kingdom, Australia'
 'United Kingdom, Australia, Canada, United States'
 'Germany, United States, Italy' 'United States, Venezuela'
 'United Kingdom, Canada, Japan'
 'United Kingdom, United States, Czech Republic'
 'United Kingdom, China, United States' 'United Kingdom, Brazil, Germany'
 'United Kingdom, Namibia, South Africa, Zimbabwe, United States'
 'Canada, United States, India, United Kingdom'
 'Switzerland, United Kingdom, United States'
 'United Kingdom, India, Sweden'
 'United States, Brazil, India, Uganda, China'
 'Peru, United States, United Kingdom'
 'Germany, United States, United Kingdom, Canada'
 'Canada, India, Thailand, United States, United Arab Emirates'
 'United States, East Germany, West Germany'
 'France, Netherlands, South Africa, Finland'
 'Egypt, Austria, United States' 'Russia, Spain'
 'Croatia, Slovenia, Serbia, Montenegro' 'Japan, Canada'
 'United States, France, South Korea, Indonesia'
 'United Arab Emirates, Jordan']
```

A single title can have different countries

[115]:

```python
print("No of Unique Countries: ",df_net['cast'].nunique())
print(df_net['cast'].unique())
```

```
No of Unique Countries:  7692
[nan
 'Ama Qamata, Khosi Ngema, Gail Mabalane, Thabang Molaba, Dillon Windvogel,
Natasha Thahane, Arno Greeff, Xolile Tshabalala, Getmore Sithole, Cindy
Mahlangu, Ryle De Morny, Greteli Fincham, Sello Maake Ka-Ncube, Odwa Gwanya,
Mekaila Mathys, Sandi Schultz, Duane Williams, Shamilla Miller, Patrick
Mofokeng'
```

```
 'Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabiha Akkari, Sofia Lesaffre, Salim
Kechiouche, Noureddine Farihi, Geert Van Rampelberg, Bakary Diombera'
 …
 'Jesse Eisenberg, Woody Harrelson, Emma Stone, Abigail Breslin, Amber Heard,
Bill Murray, Derek Graf'
 'Tim Allen, Courteney Cox, Chevy Chase, Kate Mara, Ryan Newman, Michael
Cassidy, Spencer Breslin, Rip Torn, Kevin Zegers'
 'Vicky Kaushal, Sarah-Jane Dias, Raaghav Chanana, Manish Chaudhary, Meghna
Malik, Malkeet Rauni, Anita Shabdish, Chittaranjan Tripathy']
```

Similarly, a single title can also have multiple casts

Preprocessing the data

```
[116]: #Unnesting cast column in a separate dataframe first
df1 = df_net[['title', 'cast']]
df1 = df1.dropna(axis = 0)
df1['cast'] = df1['cast'].str.split(',')
df1
```

```
[116]:                                  title  \
1                             Blood & Water
2                                 Ganglands
4                              Kota Factory
5                             Midnight Mass
6      My Little Pony: A New Generation
…                                         …
8801                                Zinzana
8802                                 Zodiac
8804                             Zombieland
8805                                   Zoom
8806                                 Zubaan

                                                    cast
1      [Ama Qamata,  Khosi Ngema,  Gail Mabalane,  Th…
2      [Sami Bouajila,  Tracy Gotoas,  Samuel Jouy,  …
4      [Mayur More,  Jitendra Kumar,  Ranjan Raj,  Al…
5      [Kate Siegel,  Zach Gilford,  Hamish Linklater…
6      [Vanessa Hudgens,  Kimiko Glenn,  James Marsde…
…                                                  …
8801   [Ali Suliman,  Saleh Bakri,  Yasa,  Ali Al-Jab…
8802   [Mark Ruffalo,  Jake Gyllenhaal,  Robert Downe…
8804   [Jesse Eisenberg,  Woody Harrelson,  Emma Ston…
8805   [Tim Allen,  Courteney Cox,  Chevy Chase,  Kat…
8806   [Vicky Kaushal,  Sarah-Jane Dias,  Raaghav Cha…

[7982 rows x 2 columns]
```

[128]:
```
df1 = df1.explode('cast')
df1
```

[128]:
```
                 title                     cast
1     Blood & Water               Ama Qamata
1     Blood & Water              Khosi Ngema
1     Blood & Water            Gail Mabalane
1     Blood & Water           Thabang Molaba
1     Blood & Water         Dillon Windvogel
…                 …                        …
8806         Zubaan         Manish Chaudhary
8806         Zubaan             Meghna Malik
8806         Zubaan            Malkeet Rauni
8806         Zubaan           Anita Shabdish
8806         Zubaan   Chittaranjan Tripathy

[64126 rows x 2 columns]
```

[129]:
```
# Unnesting cast column in a separate dataframe first
df2 = df_net[['title', 'country']]
df2 = df2.dropna(axis = 0)
df2['country'] = df2['country'].str.split(', ')
```

[130]:
```
df2 = df2.explode('country')
df2
```

[130]:
```
                        title        country
0     Dick Johnson Is Dead  United States
1            Blood & Water   South Africa
4             Kota Factory          India
7                  Sankofa  United States
7                  Sankofa          Ghana
…                       …            …
8801               Zinzana         Jordan
8802                Zodiac  United States
8804            Zombieland  United States
8805                  Zoom  United States
8806                Zubaan          India

[10014 rows x 2 columns]
```

Unnesting listed in column for the titles

[131]:
```
# Now, all columns have been unnested separately, combining all tables in one
dftemp = df1.merge(df2, on="title", how="outer")
dftemp
```

[131]:
```
                                 title               cast        country
0                        Blood & Water         Ama Qamata   South Africa
1                        Blood & Water        Khosi Ngema   South Africa
2                        Blood & Water      Gail Mabalane   South Africa
3                        Blood & Water     Thabang Molaba   South Africa
4                        Blood & Water   Dillon Windvogel   South Africa
…                                …                …             …
81607                           Winnie                NaN   South Africa
81608                           Winnie                NaN        Finland
81609                Women Behind Bars                NaN  United States
81610                        Woodstock                NaN  United States
81611  WWII: Report from the Aleutians                NaN  United States

[81612 rows x 3 columns]
```

Now, merging this unnested data with the main data so we can also get the other unnested columns

[132]:
```
df_final = df_net.merge(dftemp, on='title', how='outer')
df_final = df_final.drop(['cast_x', 'country_x'], axis=1)
df_final = df_final.rename({'cast_y':'cast', 'country_y':'country'},
axis = 1)
```

Removing the unnecessary columns and renaming the required columns

[133]:
```
df_final.head()
```

[133]:
```
  show_id     type                 title         director date_added  \
0      s1    Movie  Dick Johnson Is Dead  Kirsten Johnson 2021-09-25
1      s2  TV Show         Blood & Water              NaN 2021-09-24
2      s2  TV Show         Blood & Water              NaN 2021-09-24
3      s2  TV Show         Blood & Water              NaN 2021-09-24
4      s2  TV Show         Blood & Water              NaN 2021-09-24

   release_year rating   duration  \
0          2020  PG-13     90 min
1          2021  TV-MA  2 Seasons
2          2021  TV-MA  2 Seasons
3          2021  TV-MA  2 Seasons
4          2021  TV-MA  2 Seasons

                                         listed_in  \
0                                    Documentaries
1  International TV Shows, TV Dramas, TV Mysteries
2  International TV Shows, TV Dramas, TV Mysteries
3  International TV Shows, TV Dramas, TV Mysteries
4  International TV Shows, TV Dramas, TV Mysteries

                                         description             cast  \
0  As her father nears the end of his life, filmm…              NaN
1  After crossing paths at a party, a Cape Town t…       Ama Qamata
2  After crossing paths at a party, a Cape Town t…      Khosi Ngema
3  After crossing paths at a party, a Cape Town t…    Gail Mabalane
4  After crossing paths at a party, a Cape Town t…   Thabang Molaba

         country
0  United States
1   South Africa
2   South Africa
3   South Africa
4   South Africa
```

[134]:

```
df_final['duration_new'] = df_final['duration'].str.split(' ').str[0].
 ↪astype(float)
df_final.head()
```

[134]:

```
  show_id     type                 title         director date_added  \
0      s1    Movie  Dick Johnson Is Dead  Kirsten Johnson 2021-09-25
1      s2  TV Show         Blood & Water              NaN 2021-09-24
2      s2  TV Show         Blood & Water              NaN 2021-09-24
3      s2  TV Show         Blood & Water              NaN 2021-09-24
4      s2  TV Show         Blood & Water              NaN 2021-09-24

   release_year rating   duration  \
0          2020  PG-13     90 min
1          2021  TV-MA  2 Seasons
2          2021  TV-MA  2 Seasons
3          2021  TV-MA  2 Seasons
4          2021  TV-MA  2 Seasons

                                         listed_in  \
0                                    Documentaries
1  International TV Shows, TV Dramas, TV Mysteries
2  International TV Shows, TV Dramas, TV Mysteries
3  International TV Shows, TV Dramas, TV Mysteries
4  International TV Shows, TV Dramas, TV Mysteries

                                         description             cast  \
0  As her father nears the end of his life, filmm…              NaN
1  After crossing paths at a party, a Cape Town t…       Ama Qamata
2  After crossing paths at a party, a Cape Town t…      Khosi Ngema
3  After crossing paths at a party, a Cape Town t…    Gail Mabalane
4  After crossing paths at a party, a Cape Town t…   Thabang Molaba

         country  duration_new
0  United States          90.0
```

```
1    South Africa           2.0
2    South Africa           2.0
3    South Africa           2.0
4    South Africa           2.0
```

df_final is the final data after pre-processing

[135]:
```
df_final.shape
```

[135]:
```
(81902, 13)
```

0.2 Visual Analysis

Univariate graphs

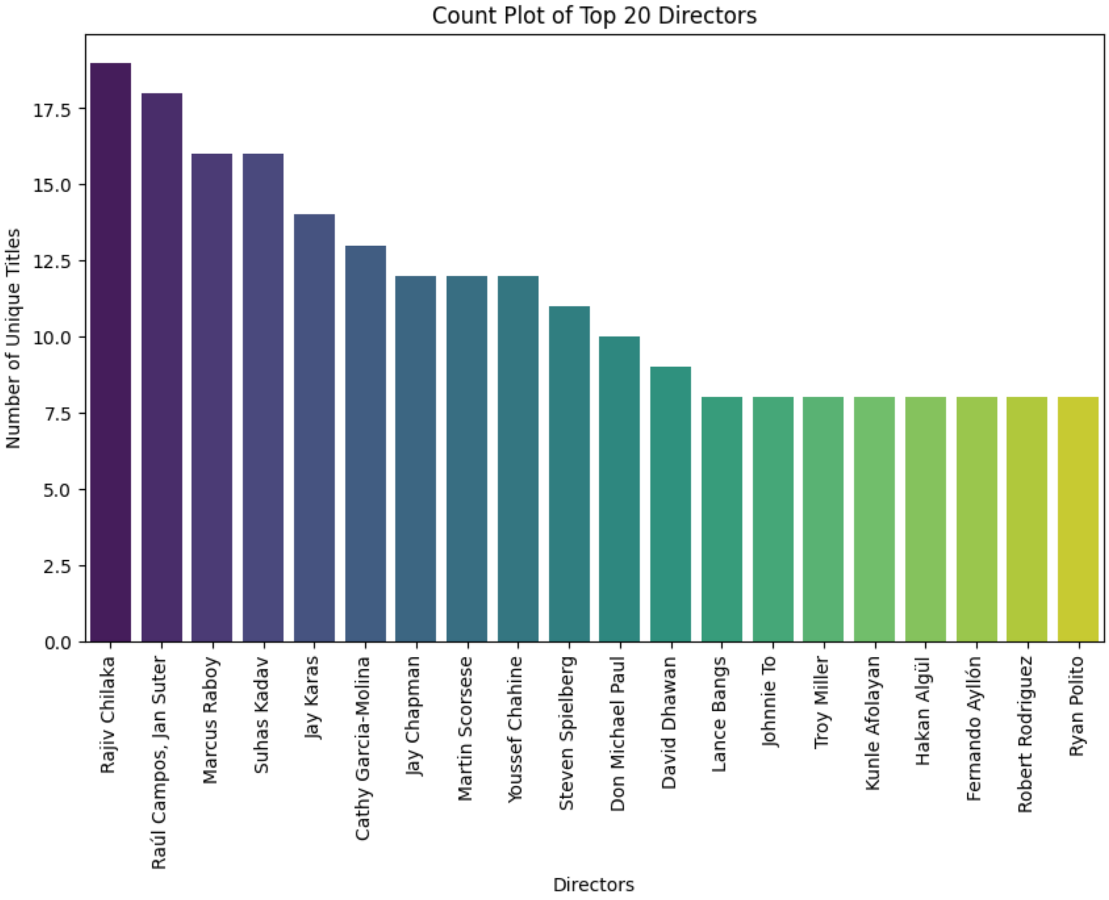
[137]:
```
df_final.groupby('director')['title'].nunique().sort_values(ascending=False)[:
 ↪20]
```

[137]:
```
director
Rajiv Chilaka              19
Raúl Campos, Jan Suter     18
Marcus Raboy               16
Suhas Kadav                16
Jay Karas                  14
Cathy Garcia-Molina        13
Jay Chapman                12
Martin Scorsese            12
Youssef Chahine            12
Steven Spielberg           11
Don Michael Paul           10
David Dhawan                9
Lance Bangs                 8
Johnnie To                  8
Troy Miller                 8
Kunle Afolayan              8
Hakan Algül                 8
Fernando Ayllón             8
Robert Rodriguez            8
Ryan Polito                 8
Name: title, dtype: int64
```

Showing Bar graph for only top 10 directors, this graph is univariate only, value_counts is done so as to not get duplicate title values for one director

[139]:
```
director_counts = df_final.groupby('director')['title'].nunique().
 ↪sort_values(ascending=False)[:20]
plt.figure(figsize=(10, 6))
sns.barplot(x=director_counts.index, y=director_counts,
palette='viridis')
plt.xticks(rotation=90)
plt.xlabel('Directors')
plt.ylabel('Number of Unique Titles')
plt.title('Count Plot of Top 20 Directors')
plt.show()
```

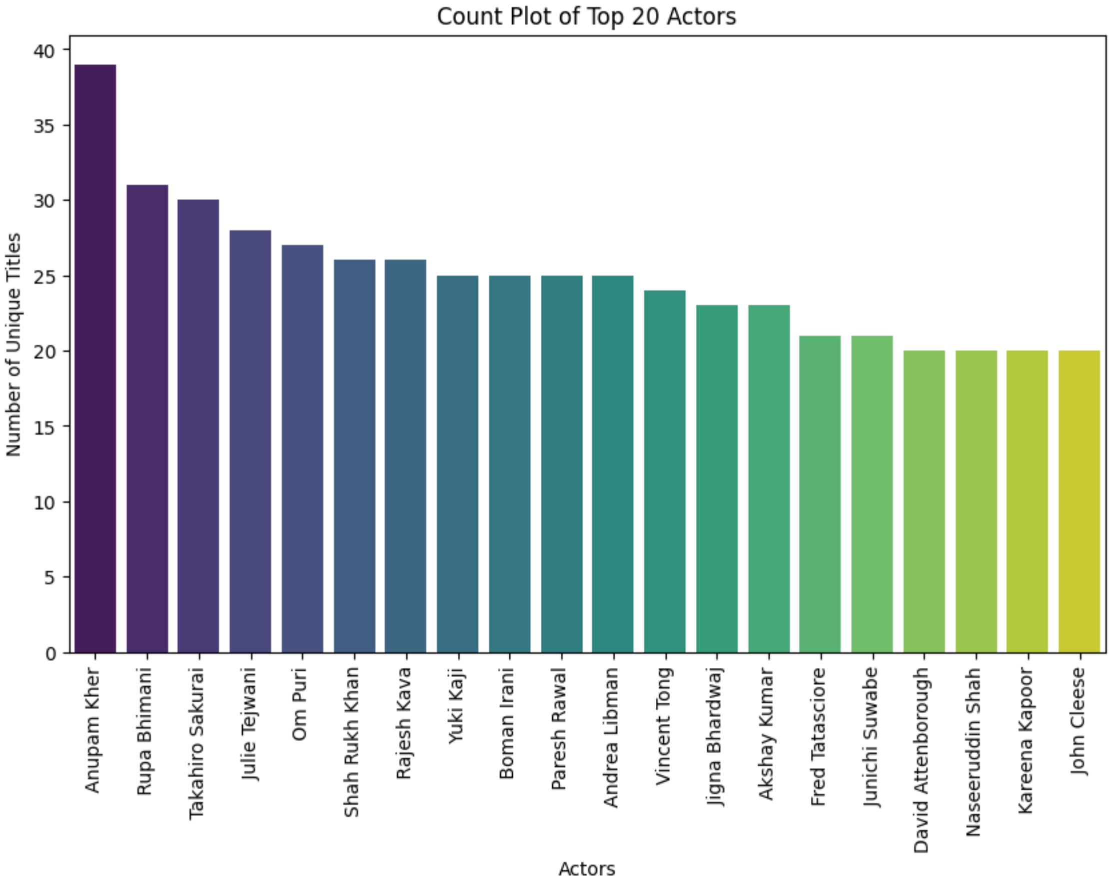
```
C:\Users\kulde\AppData\Local\Temp\ipykernel_10608\2918323794.py:3:
FutureWarning:

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

  sns.barplot(x=director_counts.index, y=director_counts,
```



This graph shows top 20 directors who have the most number of titles.

[141]:
```
actor_counts = df_final.groupby('cast')['title'].nunique().
  ↪sort_values(ascending=False)[:20]
plt.figure(figsize=(10, 6))
sns.barplot(x=actor_counts.index, y=actor_counts, palette='viridis')
plt.xticks(rotation=90)
plt.xlabel('Actors')
plt.ylabel('Number of Unique Titles')
plt.title('Count Plot of Top 20 Actors')
plt.show()
```

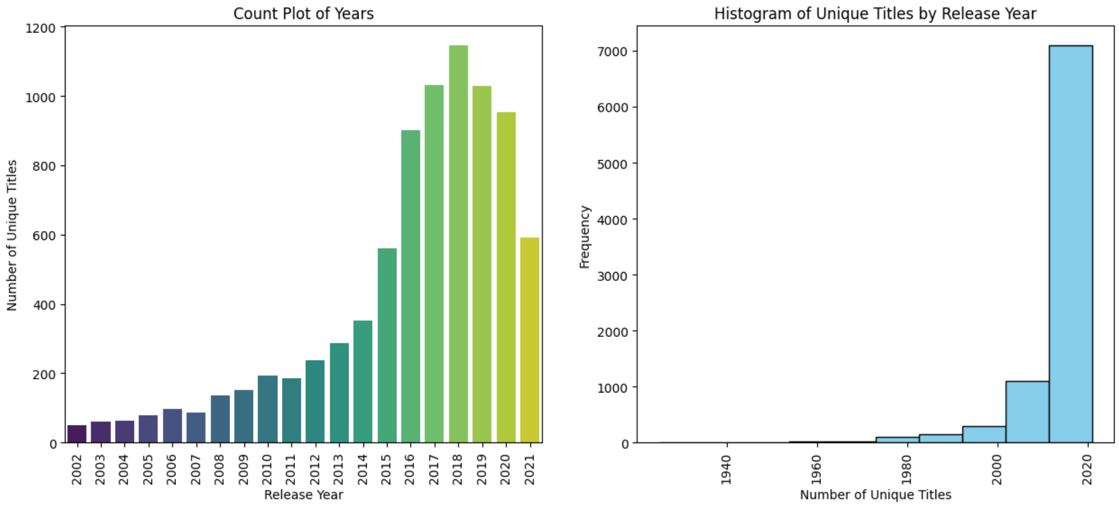
```
C:\Users\kulde\AppData\Local\Temp\ipykernel_10608\725069213.py:3: FutureWarning:

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

  sns.barplot(x=actor_counts.index, y=actor_counts, palette='viridis')
```



This graph shows top 20 actors who have the most number of titles and the Anupam Kher has the most number of titles.

```
[142]: releaseyear_counts = df_final.groupby('release_year')['title'].nunique().
        ↪sort_values(ascending=False)[:20]
       result_list = np.repeat(df_final.groupby('release_year')
       ['title'].nunique().index, df_final.groupby('release_year')
       ['title'].nunique())
       plt.figure(figsize=(15, 6))
       plt.subplot(1, 2, 1)
       sns.barplot(x=releaseyear_counts.index, y=releaseyear_counts,
       palette='viridis')
       plt.xticks(rotation=90)
       plt.xlabel('Release Year')
       plt.ylabel('Number of Unique Titles')
       plt.title('Count Plot of Years')
       plt.subplot(1, 2, 2)
       plt.hist(result_list, bins=10, color='skyblue', edgecolor='black')
       plt.xticks(rotation=90)
       plt.xlabel('Number of Unique Titles')
       plt.ylabel('Frequency')
       plt.title('Histogram of Unique Titles by Release Year')
       plt.show()
```

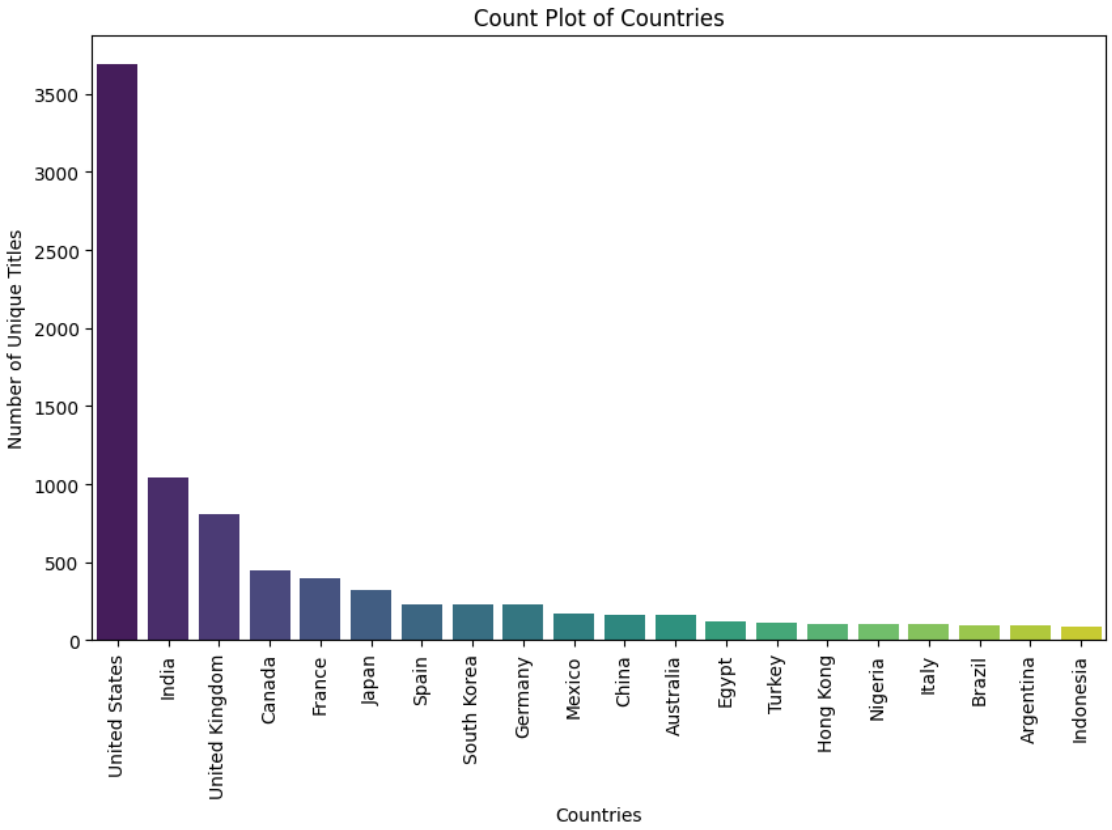
```
C:\Users\kulde\AppData\Local\Temp\ipykernel_10608\3153731356.py:7:
FutureWarning:

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

  sns.barplot(x=releaseyear_counts.index, y=releaseyear_counts,
```



These graphs show the years in which majority of titles were made, and the second graph shows the histogram that represents spread of movies and tv shows over the years.

[143]:
```python
country_counts = df_final.groupby('country')['title'].nunique().sort_values(ascending=False)[:20]
plt.figure(figsize=(10, 6))
sns.barplot(x=country_counts.index, y=country_counts,
palette='viridis')
plt.xticks(rotation=90)
plt.xlabel('Countries')
plt.ylabel('Number of Unique Titles')
plt.title('Count Plot of Countries')
plt.show()
```

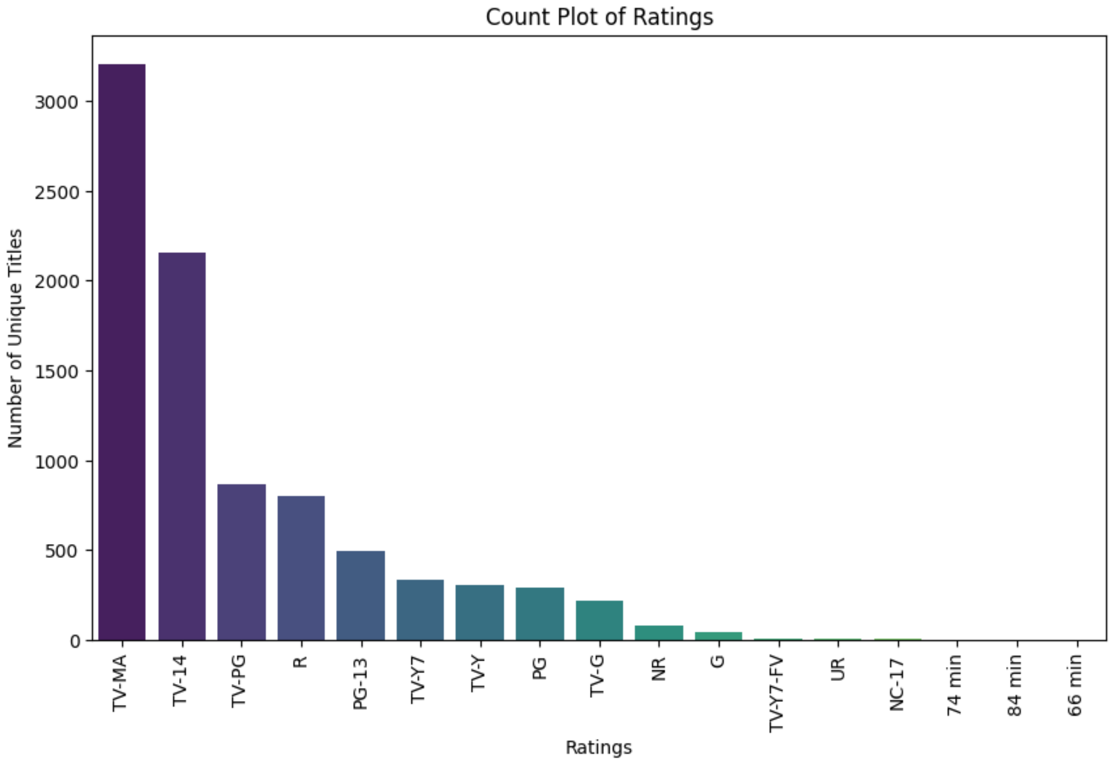
```
C:\Users\kulde\AppData\Local\Temp\ipykernel_10608\4221877869.py:3:
FutureWarning:

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

  sns.barplot(x=country_counts.index, y=country_counts,
```



It can be seen from above graph that majority of the titles are made for United States.

[145]:
```
rating_counts = df_final.groupby('rating')['title'].nunique().
 ↪sort_values(ascending=False)
plt.figure(figsize=(10, 6))
sns.barplot(x=rating_counts.index, y=rating_counts, palette='viridis')
plt.xticks(rotation=90)
plt.xlabel('Ratings')
plt.ylabel('Number of Unique Titles')
plt.title('Count Plot of Ratings')
plt.show()
```

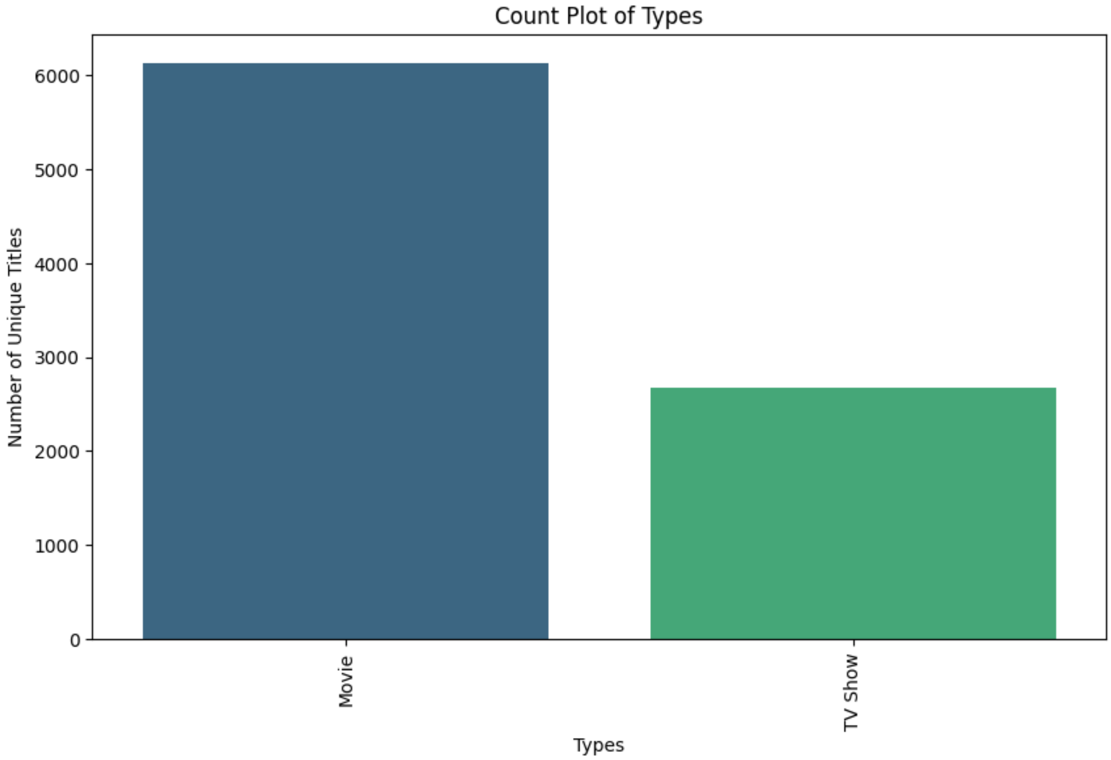
```
C:\Users\kulde\AppData\Local\Temp\ipykernel_10608\1126638516.py:3:
FutureWarning:

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

  sns.barplot(x=rating_counts.index, y=rating_counts, palette='viridis')
```



Most of the titles were made for TV-MA rating, followed by TV-14 and the rest followed as in graph.

[146]:
```
type_counts = df_final.groupby('type')['title'].nunique().
  ↪sort_values(ascending=False)
plt.figure(figsize=(10, 6))
sns.barplot(x=type_counts.index, y=type_counts, palette='viridis')
plt.xticks(rotation=90)
plt.xlabel('Types')
plt.ylabel('Number of Unique Titles')
plt.title('Count Plot of Types')
plt.show()
```

```
C:\Users\kulde\AppData\Local\Temp\ipykernel_10608\2412895408.py:3:
FutureWarning:

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

  sns.barplot(x=type_counts.index, y=type_counts, palette='viridis')
```



It can be seen from the graph that most titles were movies and tv shows are lesser as compared to the movies.

[148]:

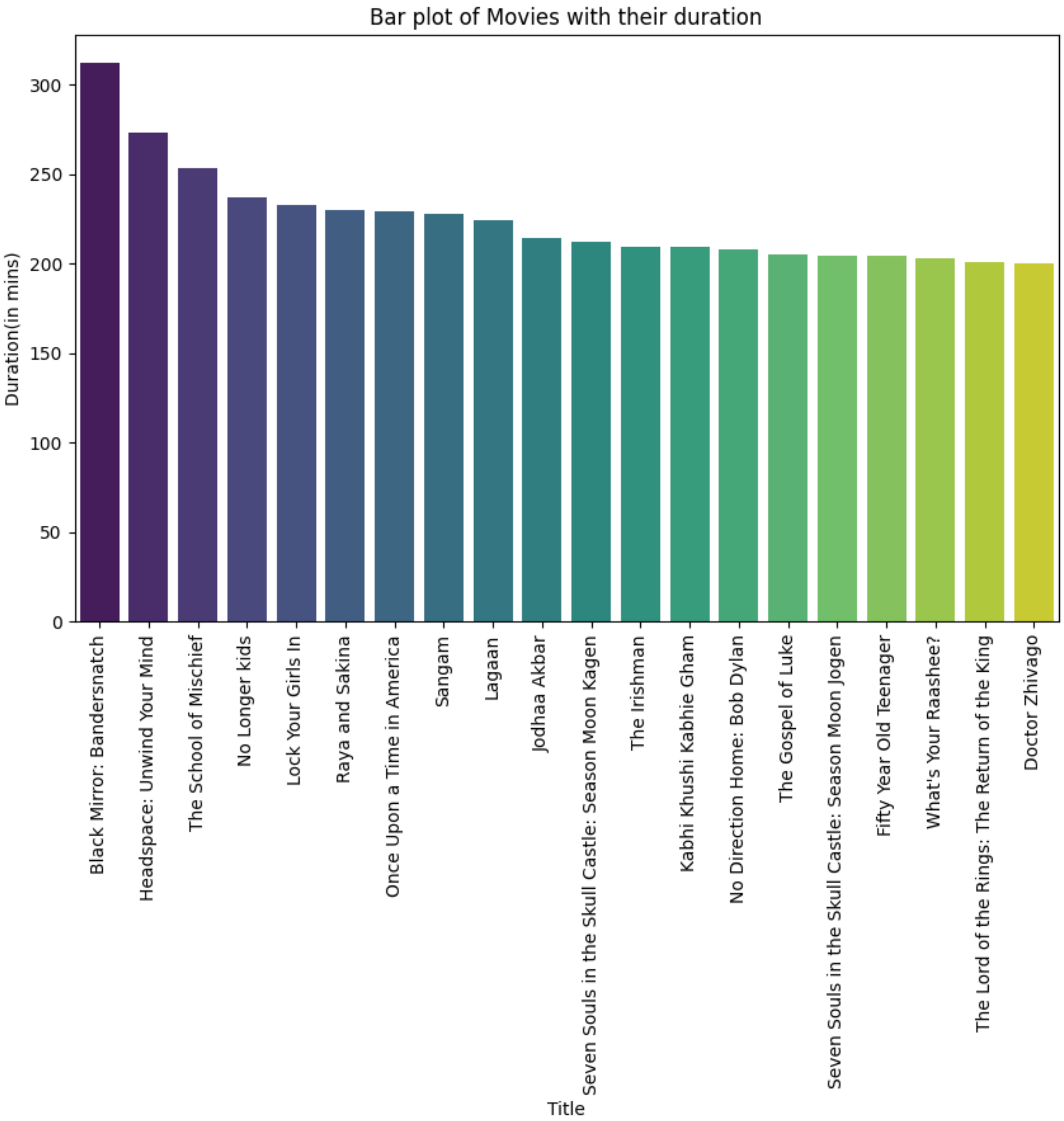
```
movies_counts = df_final.loc[(df_final['type'] ==
'Movie')].sort_values(by='duration_new',
ascending=False).drop_duplicates(subset='title')[:20]
tv_shows_counts = df_final.loc[(df_final['type'] == 'TVShow')].
 ↪sort_values(by='duration_new',
ascending=False).drop_duplicates(subset='title')[:20]
```

Bivariate analysis

[149]:

```
plt.figure(figsize=(10, 6))
sns.barplot(x='title', y='duration_new', data=movies_counts,
palette='viridis')
plt.xticks(rotation=90)
plt.xlabel('Title')
plt.ylabel('Duration(in mins)')
plt.title('Bar plot of Movies with their duration')
plt.show()
```

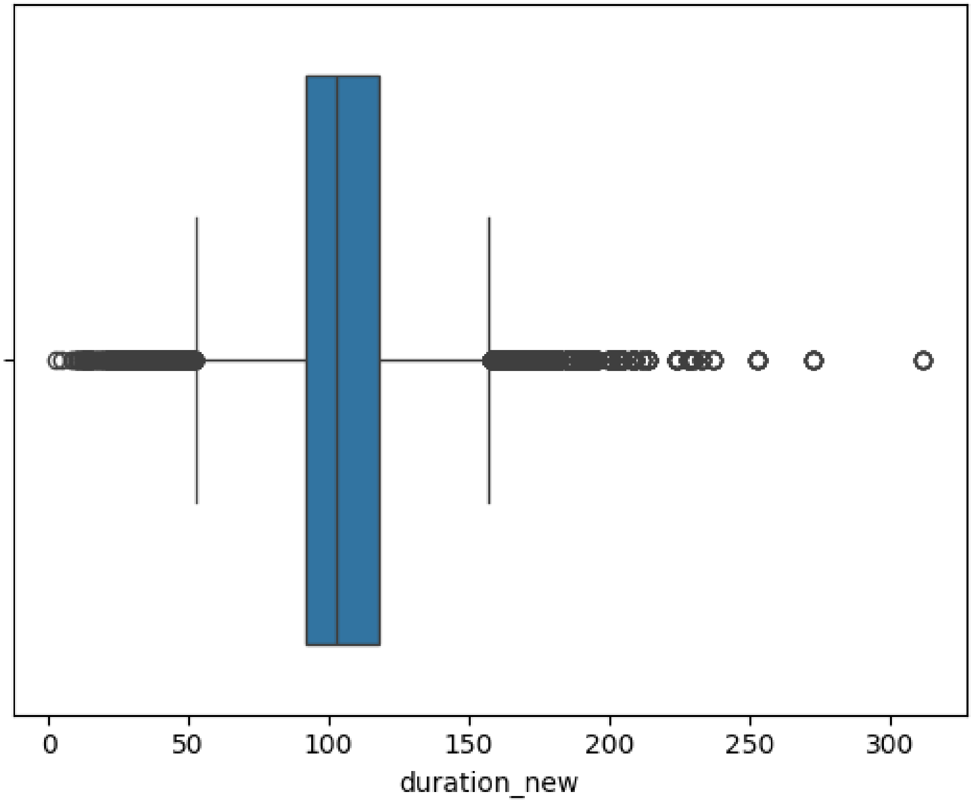
```
C:\Users\kulde\AppData\Local\Temp\ipykernel_10608\2667032672.py:2:
FutureWarning:

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

  sns.barplot(x='title', y='duration_new', data=movies_counts,
```



The above graph shows top 20 movies according to their durations in mins.

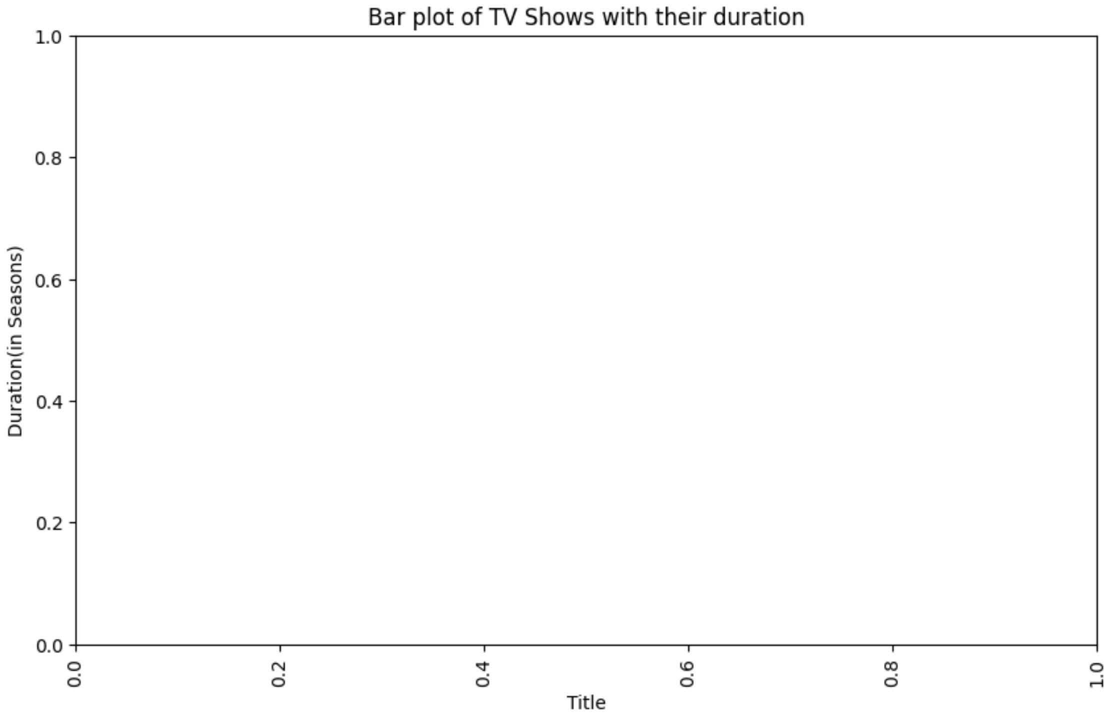
```
[150]: sns.boxplot(x='duration_new',
       data=df_final[df_final['type']=='Movie'])
```

```
[150]: <Axes: xlabel='duration_new'>
```



The above graph shows median duration and spread of movies durations.

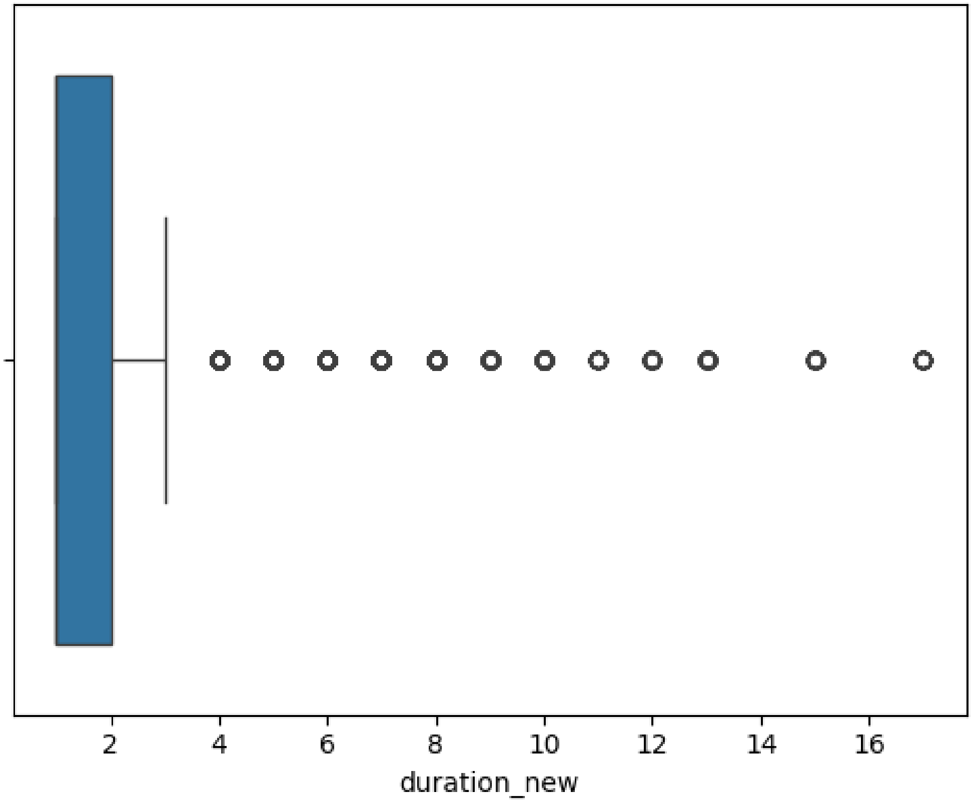
[151]:
```
plt.figure(figsize=(10, 6))
sns.barplot(x='title', y='duration_new', data=tv_shows_counts,
palette='viridis')
plt.xticks(rotation=90)
plt.xlabel('Title')
plt.ylabel('Duration(in Seasons)')
plt.title('Bar plot of TV Shows with their duration')
plt.show()
```



The above graph shows top 20 tv shows according to their durations in number of seasons they have in total.

```
[157]: sns.boxplot(x='duration_new', data=df_final[df_final['type']=='TV Show'])
```

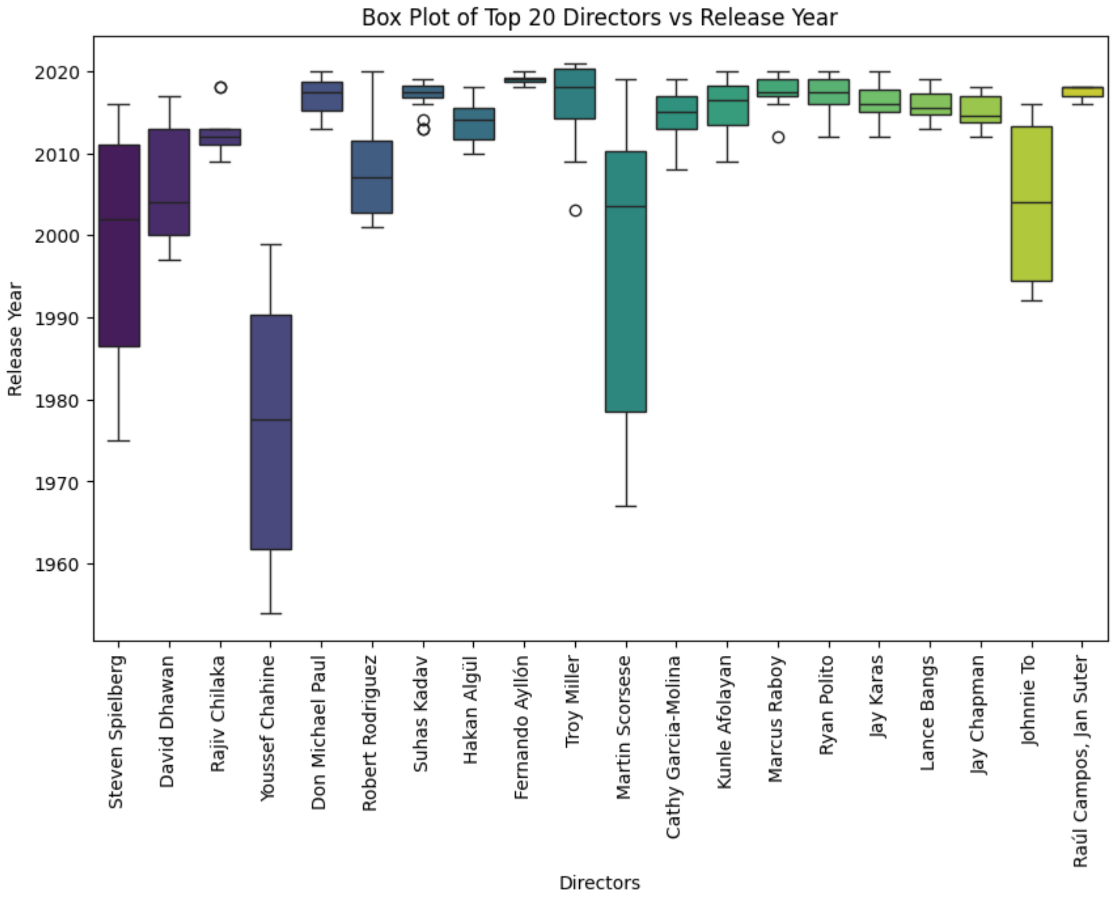
```
[157]: <Axes: xlabel='duration_new'>
```



The above graph shows the spread of duration(in seasons) for tv shows.

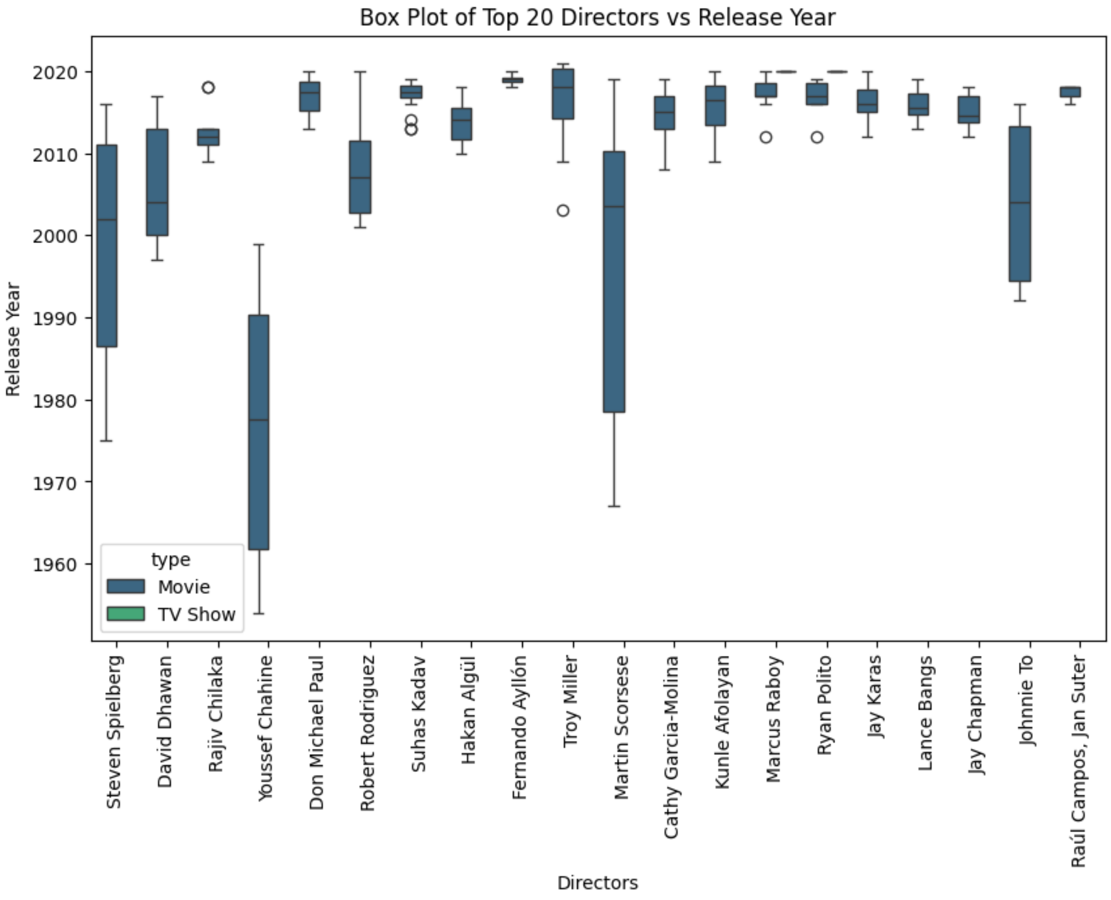
[158]:
```
plt.figure(figsize=(10, 6))
sns.boxplot(x='director', y='release_year',
data=df_final[df_final['director'].isin(director_counts.index[:20])].
  ↪drop_duplicates(subset='title'), palette='viridis')
plt.xticks(rotation=90)
plt.xlabel('Directors')
plt.ylabel('Release Year')
plt.title('Box Plot of Top 20 Directors vs Release Year')
plt.show()
```

```
C:\Users\kulde\AppData\Local\Temp\ipykernel_10608\2645077151.py:2:
FutureWarning:

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

  sns.boxplot(x='director', y='release_year',
```



This graph shows the spread of titles over years for the top 20 directors who made the most titles.

```
[159]: plt.figure(figsize=(10, 6))
       sns.boxplot(x='director', y='release_year', hue='type',
       data=df_final[df_final['director'].isin(director_counts.index[:20])].
         ↪drop_duplicates(subset='title'), palette='viridis')
       plt.xticks(rotation=90)
       plt.xlabel('Directors')
       plt.ylabel('Release Year')
       plt.title('Box Plot of Top 20 Directors vs Release Year')
       plt.show()
```

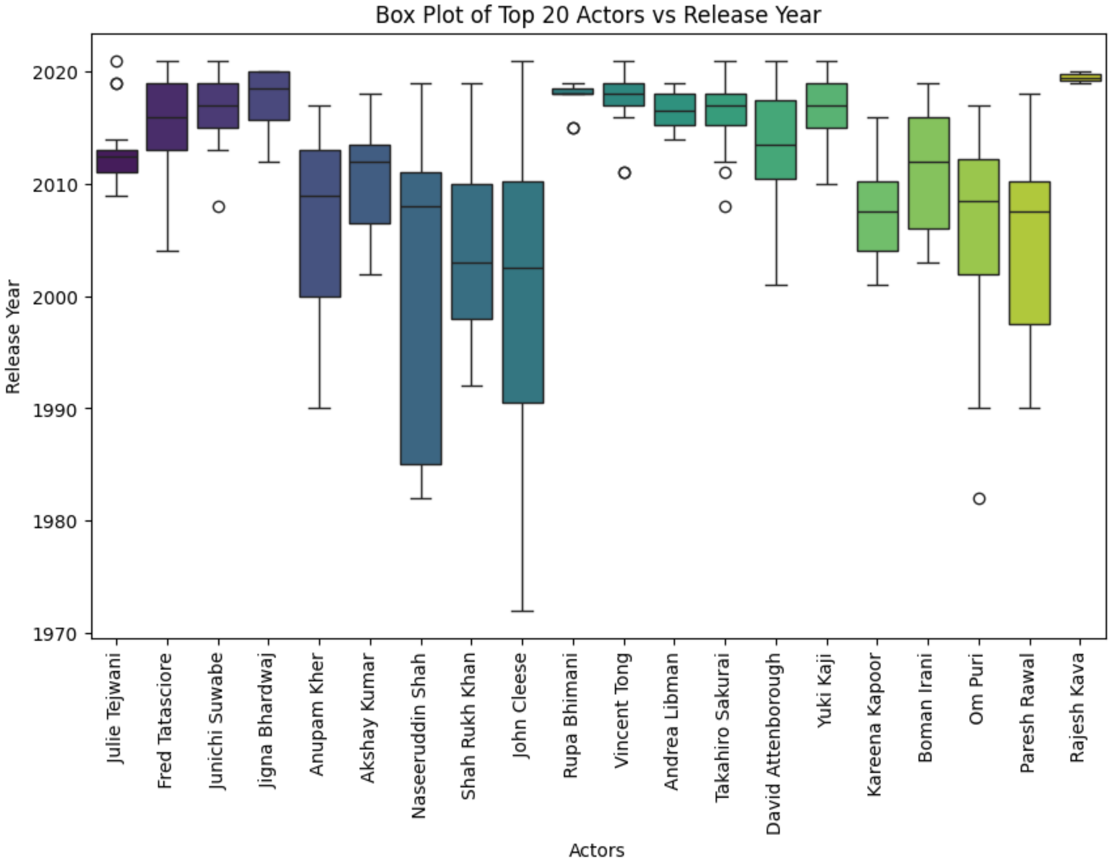


This graph shows the spread of titles over years for the top 20 directors who made the most titles divided by type of the title.

```
[161]: plt.figure(figsize=(10, 6))
sns.boxplot(x='cast', y='release_year',
data=df_final[df_final['cast'].isin(actor_counts.index[:20])].
  ↪drop_duplicates(subset='title'), palette='viridis')
plt.xticks(rotation=90)
plt.xlabel('Actors')
plt.ylabel('Release Year')
plt.title('Box Plot of Top 20 Actors vs Release Year')
plt.show()
```

```
C:\Users\kulde\AppData\Local\Temp\ipykernel_10608\121562394.py:2: FutureWarning:

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

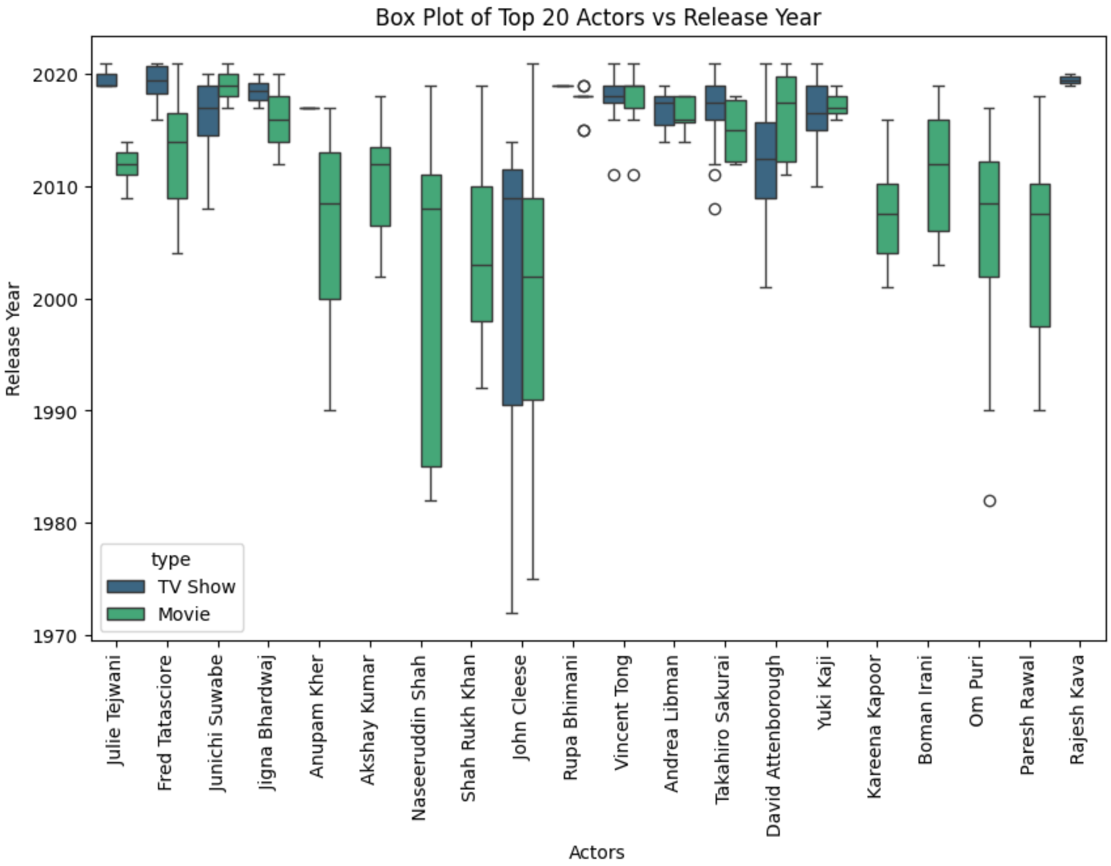
```
sns.boxplot(x='cast', y='release_year',
```



This graph shows the spread of titles over years for the top 20 actors who worked in most titles.

[162]:

```
plt.figure(figsize=(10, 6))
sns.boxplot(x='cast', y='release_year', hue='type',
data=df_final[df_final['cast'].isin(actor_counts.index[:20])].
 ↪drop_duplicates(subset='title'), palette='viridis')
plt.xticks(rotation=90)
plt.xlabel('Actors')
plt.ylabel('Release Year')
plt.title('Box Plot of Top 20 Actors vs Release Year')
plt.show()
```



The above graph shows top actors titles divided by type.

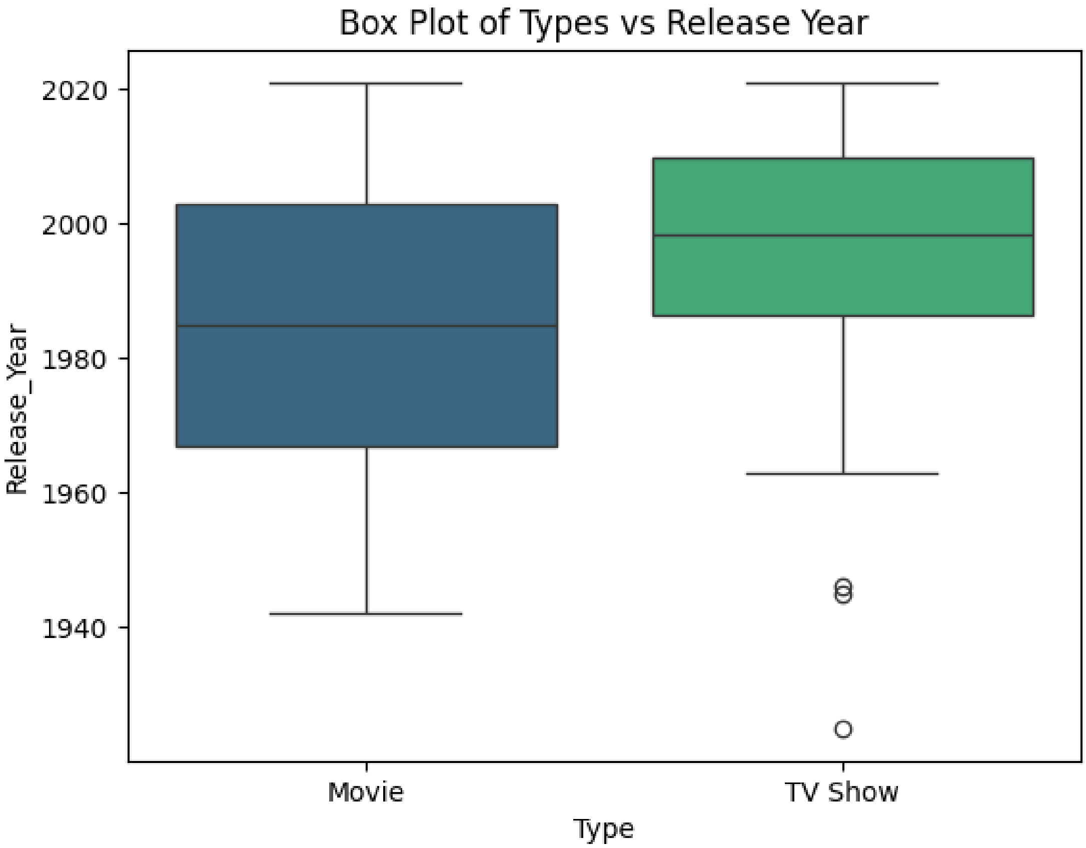
[163]:

```
sns.boxplot(x="type", y='release_year', data =
df_final.groupby(['type', 'release_year'])
['title'].nunique().reset_index(), palette='viridis')
plt.xlabel('Type')
plt.ylabel('Release_Year')
plt.title('Box Plot of Types vs Release Year')
```

```
C:\Users\kulde\AppData\Local\Temp\ipykernel_10608\4082153236.py:1:
FutureWarning:

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

  sns.boxplot(x="type", y='release_year', data =
```

[163]:

```
Text(0.5, 1.0, 'Box Plot of Types vs Release Year')
```



It can be seen from above data that median of movies were made around 1980, while tv shows median is after that in around 2000s.

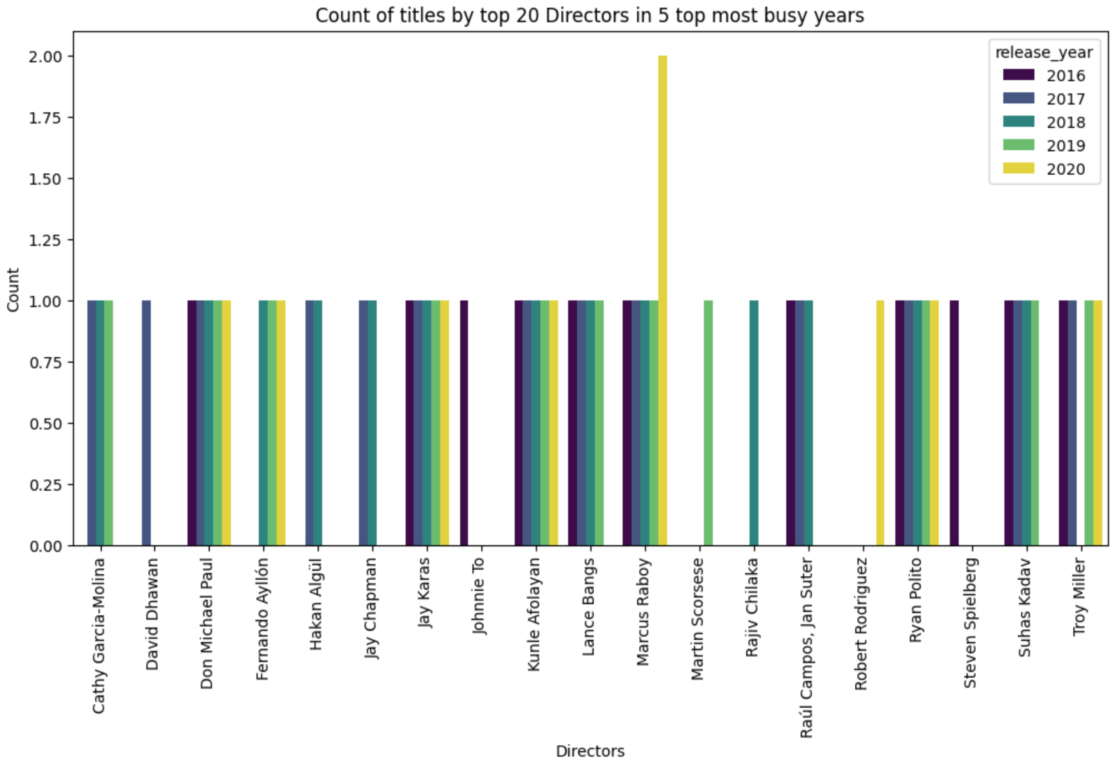
```
[164]: dd = df_final.groupby(['director', 'release_year'])['type'].nunique().
        ↪reset_index()
       dd = dd.loc[(dd['director'].isin(director_counts.index)) &
       (dd['release_year'].isin(releaseyear_counts.index[:5]))]
       dd
```

```
[164]:                     director  release_year  type
       858      Cathy Garcia-Molina          2017     1
       859      Cathy Garcia-Molina          2018     1
       860      Cathy Garcia-Molina          2019     1
       1231            David Dhawan          2017     1
       1406         Don Michael Paul          2016     1
       1407         Don Michael Paul          2017     1
       1408         Don Michael Paul          2018     1
       1409         Don Michael Paul          2019     1
       1410         Don Michael Paul          2020     1
       1616          Fernando Ayllón          2018     1
```

```
1617        Fernando Ayllón          2019     1
1618        Fernando Ayllón          2020     1
1890            Hakan Algül          2017     1
1891            Hakan Algül          2018     1
2248            Jay Chapman          2017     1
2249            Jay Chapman          2018     1
2255              Jay Karas          2016     1
2256              Jay Karas          2017     1
2257              Jay Karas          2018     1
2258              Jay Karas          2019     1
2259              Jay Karas          2020     1
2526             Johnnie To          2016     1
2921         Kunle Afolayan          2016     1
2922         Kunle Afolayan          2017     1
2923         Kunle Afolayan          2018     1
2924         Kunle Afolayan          2019     1
2925         Kunle Afolayan          2020     1
2955            Lance Bangs          2016     1
2956            Lance Bangs          2017     1
2957            Lance Bangs          2018     1
2958            Lance Bangs          2019     1
3287           Marcus Raboy          2016     1
3288           Marcus Raboy          2017     1
3289           Marcus Raboy          2018     1
3290           Marcus Raboy          2019     1
3291           Marcus Raboy          2020     2
3390        Martin Scorsese          2019     1
4356          Rajiv Chilaka          2018     1
4433 Raúl Campos, Jan Suter          2016     1
4434 Raúl Campos, Jan Suter          2017     1
4435 Raúl Campos, Jan Suter          2018     1
4599       Robert Rodriguez          2020     1
4727            Ryan Polito          2016     1
4728            Ryan Polito          2017     1
4729            Ryan Polito          2018     1
4730            Ryan Polito          2019     1
4731            Ryan Polito          2020     1
5233       Steven Spielberg          2016     1
5267            Suhas Kadav          2016     1
5268            Suhas Kadav          2017     1
5269            Suhas Kadav          2018     1
5270            Suhas Kadav          2019     1
5560            Troy Miller          2016     1
5561            Troy Miller          2017     1
5562            Troy Miller          2019     1
5563            Troy Miller          2020     1
```

[165]:

```
plt.figure(figsize=(12, 6))
sns.barplot(x='director', y='type', hue='release_year', data=dd,
palette='viridis')
plt.xticks(rotation=90)
plt.ylabel('Count')
plt.xlabel('Directors')
plt.title('Count of titles by top 20 Directors in 5 top most busy years')
plt.show()
```



The above graph shows how much titles the top 20 directors made in the most busy years and in which specific year.

[166]:

```
aa = df_final.groupby(['cast', 'release_year'])['type'].nunique().reset_index()
aa = aa.loc[(aa['cast'].isin(actor_counts.index)) &
(aa['release_year'].isin(releaseyear_counts.index[:5]))]
aa
```

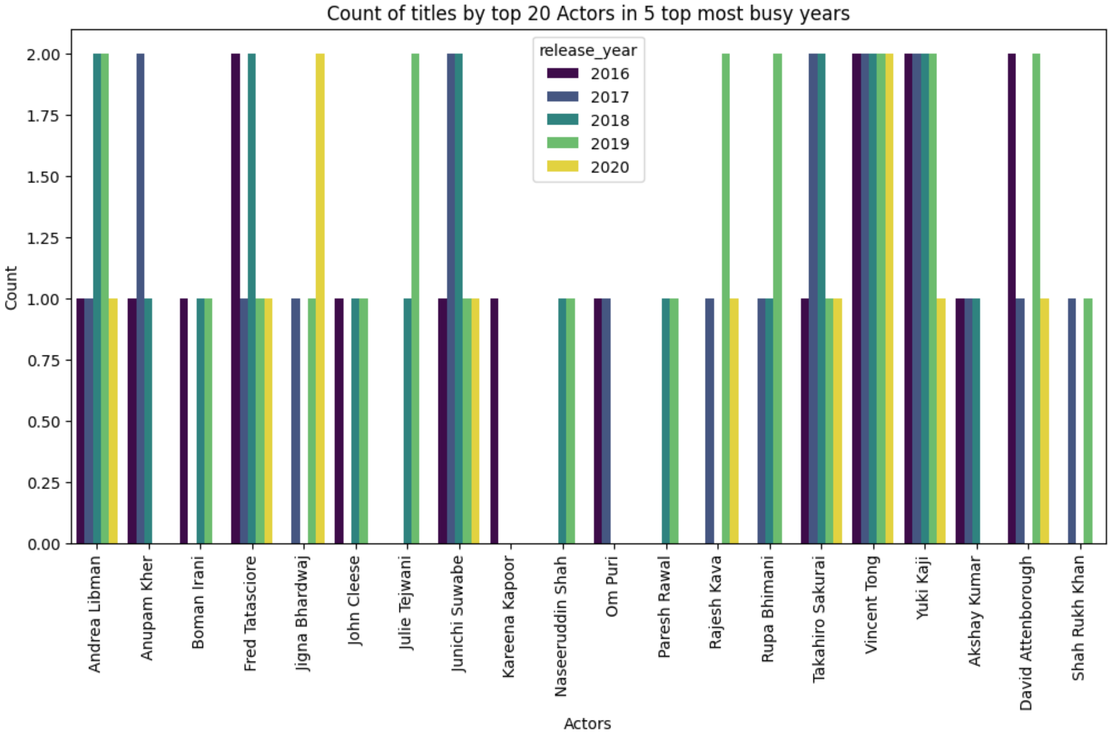
[166]:

```
                       cast  release_year  type
2876         Andrea Libman          2016     1
2877         Andrea Libman          2017     1
2878         Andrea Libman          2018     2
2879         Andrea Libman          2019     2
2880         Andrea Libman          2020     1
```

```
…                        …             …    …
53190  David Attenborough            2017      1
53191  David Attenborough            2019      2
53192  David Attenborough            2020      1
57898        Shah Rukh Khan          2017      1
57899        Shah Rukh Khan          2019      1

[66 rows x 3 columns]
```

[167]:

```
plt.figure(figsize=(12, 6))
sns.barplot(x='cast', y='type', hue='release_year', data=aa,
palette='viridis')
plt.xticks(rotation=90)
plt.ylabel('Count')
plt.xlabel('Actors')
plt.title('Count of titles by top 20 Actors in 5 top most busy years')
plt.show()
```

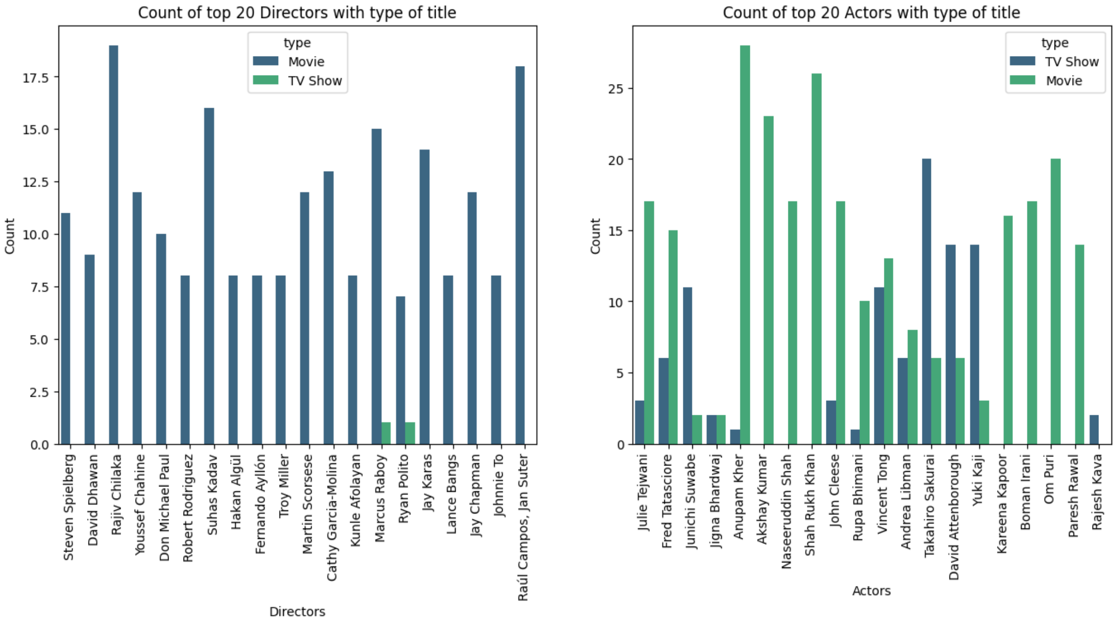


The above graph shows how much titles the top 20 actors worked in the most busy years and in which specific year.

[172]:

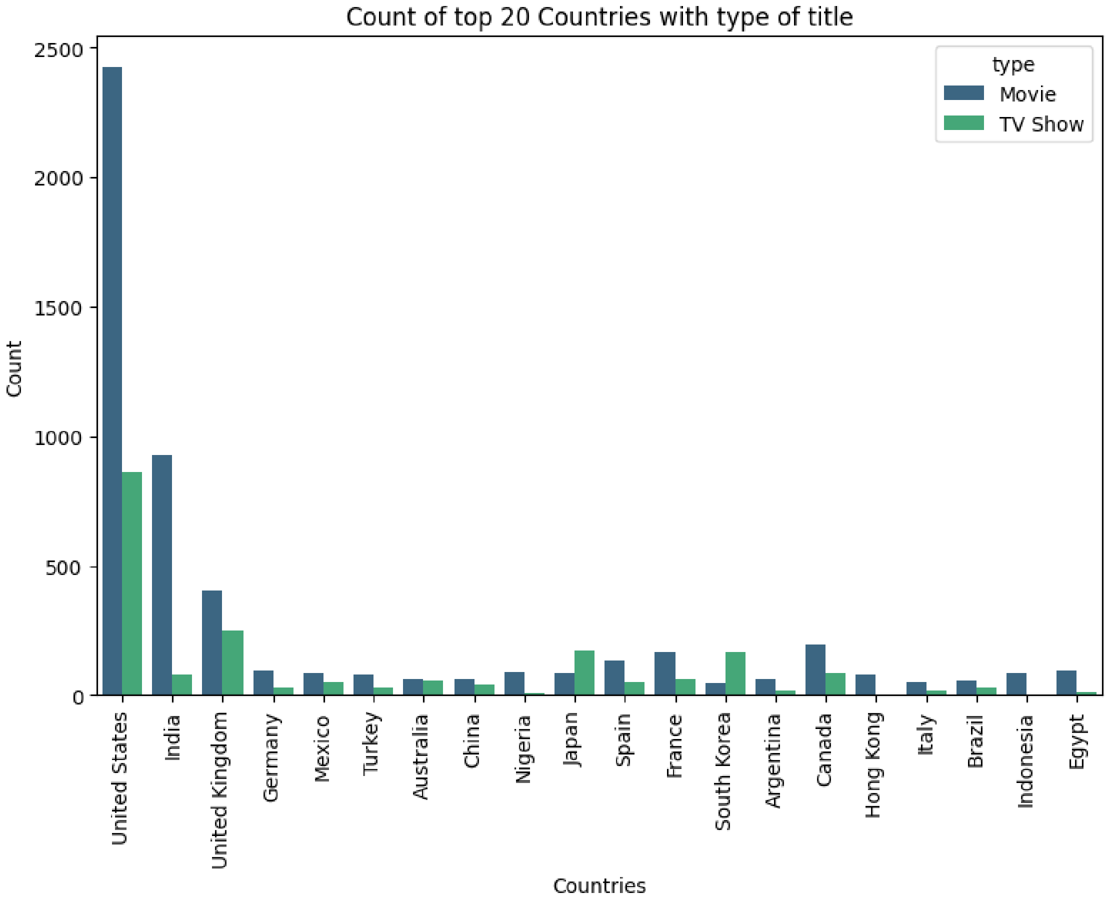
```
plt.figure(figsize=(15, 6))
plt.subplot(1, 2, 1)
```

```
sns.countplot(x='director', hue='type',
data=df_final[df_final['director'].isin(director_counts.index)].
  ↪drop_duplicates(subset='title'), palette='viridis')
plt.xticks(rotation=90)
plt.xlabel('Directors')
plt.ylabel('Count')
plt.title('Count of top 20 Directors with type of title')
plt.subplot(1, 2, 2)
sns.countplot(x='cast', hue='type',data=df_final[df_final['cast'].
  ↪isin(actor_counts.index)].drop_duplicates(subset='title'),palette='viridis')
plt.xticks(rotation=90)
plt.xlabel('Actors')
plt.ylabel('Count')
plt.title('Count of top 20 Actors with type of title')
plt.show()
```



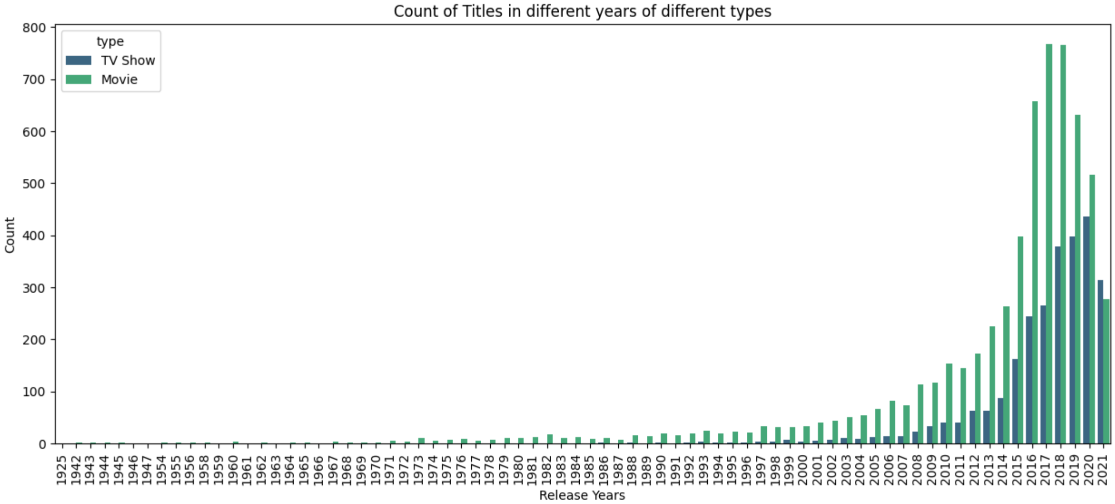
The above graphs show that only one top director made a tv show, but actors on the other hand work in tv shows as well though not as frequent as in movies

[174]:
```
plt.figure(figsize=(20, 6))
plt.subplot(1, 2, 1)
sns.countplot(x='country', hue='type',
data=df_final[df_final['country'].isin(country_counts.index)].
  ↪drop_duplicates(subset='title'), palette='viridis')
plt.xticks(rotation=90)
plt.xlabel('Countries')
plt.ylabel('Count')
plt.title('Count of top 20 Countries with type of title')
plt.show()
```



The above graph shows, moveis and tv shows made for countries, and it can be seen that most movies as well as tv shows are made for United States.

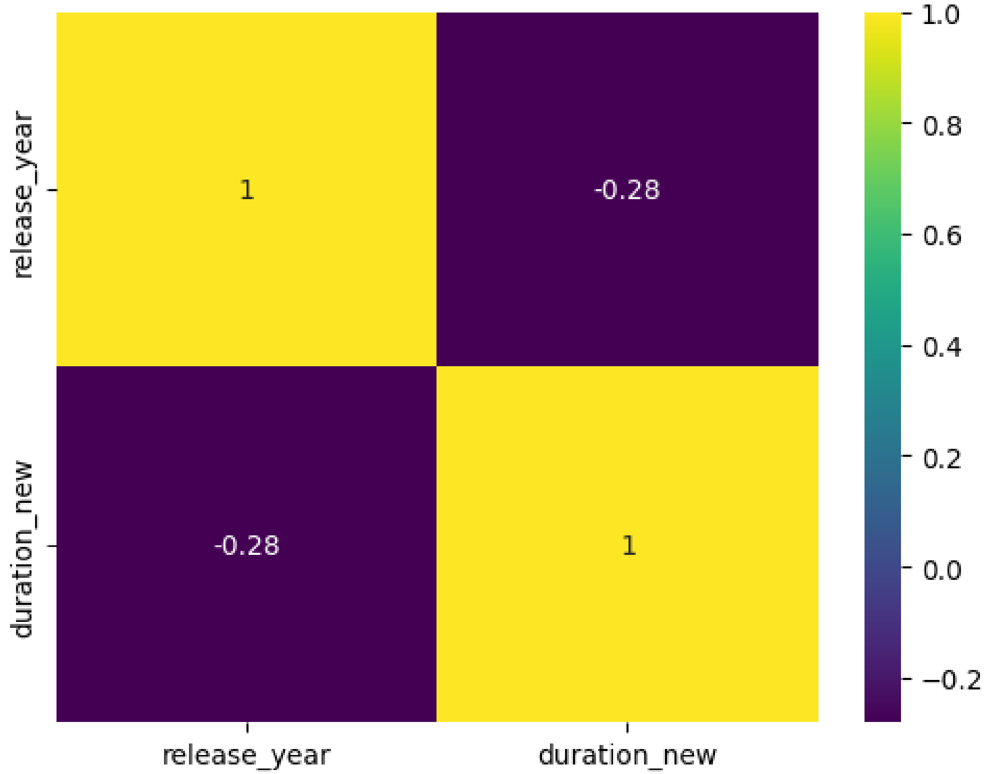
[175]:
```
plt.figure(figsize=(15, 6))
sns.countplot(x='release_year', hue='type',
data=df_final.drop_duplicates(subset='title'), palette='viridis')
plt.xticks(rotation=90)
plt.xlabel('Release Years')
plt.ylabel('Count')
plt.title('Count of Titles in different years of different types')
plt.show()
```



The above graph shows the spread of the movies and tv shows over the range of years from 1925 to 2021.

```
[184]: # Create the heatmap
       df_num = df_final.select_dtypes(include=['float64', 'int64'])
       sns.heatmap(df_num.corr(), cmap="viridis", annot=True)
```

```
[184]: <Axes: >
```



0.2.1 Missing values and outliers check

[179]:
```
df_net.info()
```

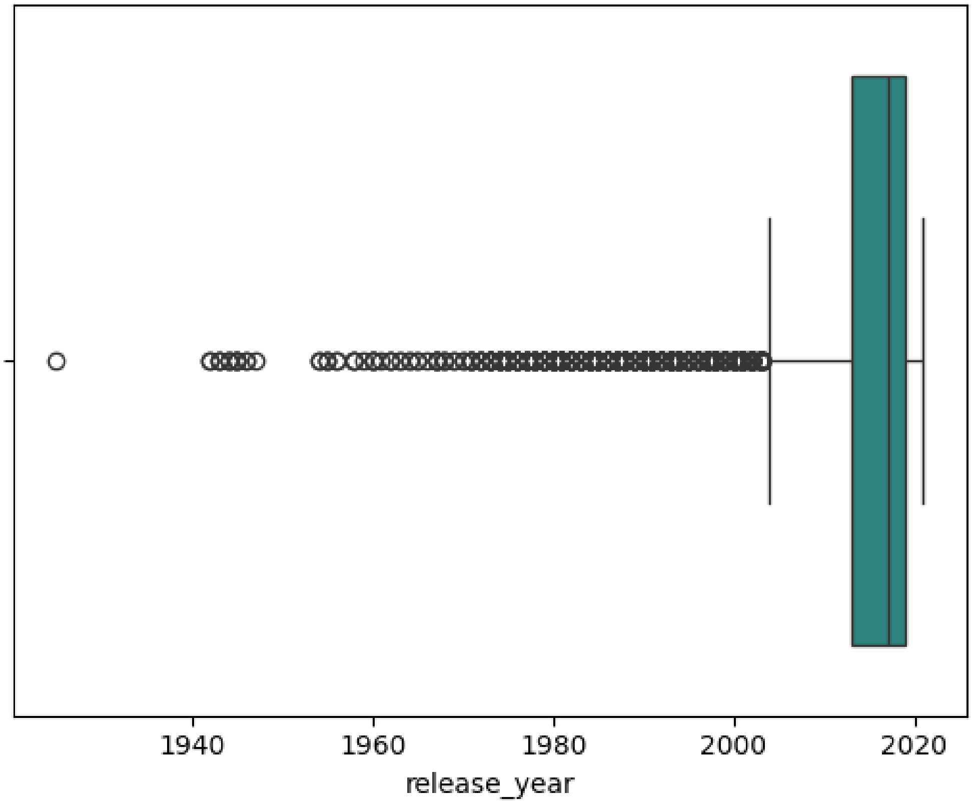
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8807 entries, 0 to 8806
Data columns (total 12 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   show_id       8807 non-null   object
 1   type          8807 non-null   object
 2   title         8807 non-null   object
 3   director      6173 non-null   object
 4   cast          7982 non-null   object
 5   country       7976 non-null   object
 6   date_added    8712 non-null   datetime64[ns]
 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: datetime64[ns](1), int64(1), object(10)
```

```
memory usage: 825.8+ KB
```

It is visible from above table, there are missing values in some of the columns

[180]:
```
sns.boxplot(x='release_year',
data=df_final.drop_duplicates(subset='title'), palette='viridis')
plt.show()
```

```
C:\Users\kulde\AppData\Local\Temp\ipykernel_10608\2183479551.py:1:
FutureWarning:

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

  sns.boxplot(x='release_year',
```



It can be seen from the above graph that there are outliers in the release_year.

0.2.2 Business Insights

1) The most popular Genres across the countries and in both TV Shows and Movies are Drama, Comedy andInternational TV Shows/Movies, so content aligning to that is recommended.

2) Add TV Shows in July/August and Movies in last week of the year/first month of the next year.

3) For USA audience 80-120 mins is the recommended length for movies and Kids TV Shows are also popularalong with the genres in first point, hence recommended.

4) For UK audience, recommended length for movies is same as that of USA (80-120 mins)

5) The target audience in USA and India is recommended to be 14+ and above ratings while for UK, itsrecommended to be completely Mature/R content .

6) Add movies for Indian Audience, it has been declining since 2018.

7) Anime Genre for Japan and Romantic Genre in TV Shows for South Korean audiences is recommended.

8) While creating content, take into consideration the popular actors/directors for that country. Also take intoaccount the director-actor combination which is highly recommended.

9) Rajiv Chilaka made the most number of movies, i.e. 19.

10) Anupan Kher acted in the most movies, i.e. 39.

11) Titles starting being made in 1950s, the production saw a boom in the early 2000s and in 2018 most titles were made countring almost to 1200.

12) TV-MA and TV-14 are the ratings most of the titles are made for.

13) More number of movies are made in total than tv shows.

14) Black Mirror is the longest movie in duration having 300+ mins of content while Grey's Anatomy is the longest running TV Show having 19 seasons in total.

15) Majority Top Directors have made their most movies in years 2000-2021 and majority of them have made only movies

0.2.3 Recommendations

1. The company should start involving titles which target more number of countries, as opposed to targeting only United States.
2. TV Shows should also be included, as right now the count of tv shows is drastically less than count of movies.
3. Old Tv Shows and Movies should be added more, as right now the data is mostly considered of recent titles.
4. Tv Shows could be made with more seasons so customers could delve into their stories more.
5. More titles could be made for non-popular ratings to target the specific viewer category.
6. Tv Shows could be made with top directors and top actors, and their lengths could be made more than average which could pull more audience for longer periods of time.