1:Defining Problem Statement and Analysing basic metrics

Ans:most of director assign null values so any one can not categorise films in director wise

```
df.isnull().sum()
```

```
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
```

2. Observations on the shape of data, data types of all the attributes, conversion of categorical attributes to 'category' (If required), missing value detection, statistical summary (10 Points)

2.1:shape of data :-`(8807, 12)`

2.2:data types :`df.dtypes`

```
df.dtypes
```

```
show_id          object
type             object
title            object
director         object
cast             object
country          object
date_added       object
release_year      int64
rating           object
duration         object
listed_in        object
description      object
dtype: object
```

2.3:conversion of categorical attributes to 'category'

```
df.groupby(['country'])['title'].count()
```

```
country
, France, Algeria                                      1
, South Korea                                          1
Argentina                                             56
Argentina, Brazil, France, Poland, Germany, Denmark    1
Argentina, Chile                                       2
                                                     ..
Venezuela                                              1
Venezuela, Colombia                                    1
Vietnam                                                7
West Germany                                           1
Zimbabwe                                               1
Name: title, Length: 748, dtype: int64
```

2.4:missing value detection, statistical summary

```
show_id          0.000000
type             0.000000
title            0.000000
director        29.908028
cast             9.367549
country          9.435676
date_added       0.113546
release_year     0.000000
rating           0.045418
duration         0.034064
listed_in        0.000000
description      0.000000
dtype: float64
```

2.4:statistical summary

```
df.describe(include="all")
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
count	8807	8807	8807	6173	7982	7976	8797	8807.000000	8803	8804	8807	8807
unique	8807	2	8807	4528	7692	748	1767	NaN	17	220	514	8775
top	s1	Movie	Dick Johnson Is Dead	Rajiv Chilaka	David Attenborough	United States	January 1, 2020	NaN	TV-MA	1 Season	Dramas, International Movies	Paranormal activity at a lush, abandoned prope...
freq	1	6131	1	19	19	2818	109	NaN	3207	1793	362	4
mean	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2014.180198	NaN	NaN	NaN	NaN
std	NaN	NaN	NaN	NaN	NaN	NaN	NaN	8.819312	NaN	NaN	NaN	NaN
min	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1925.000000	NaN	NaN	NaN	NaN
25%	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2013.000000	NaN	NaN	NaN	NaN
50%	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2017.000000	NaN	NaN	NaN	NaN
75%	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2019.000000	NaN	NaN	NaN	NaN
max	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2021.000000	NaN	NaN	NaN	NaN

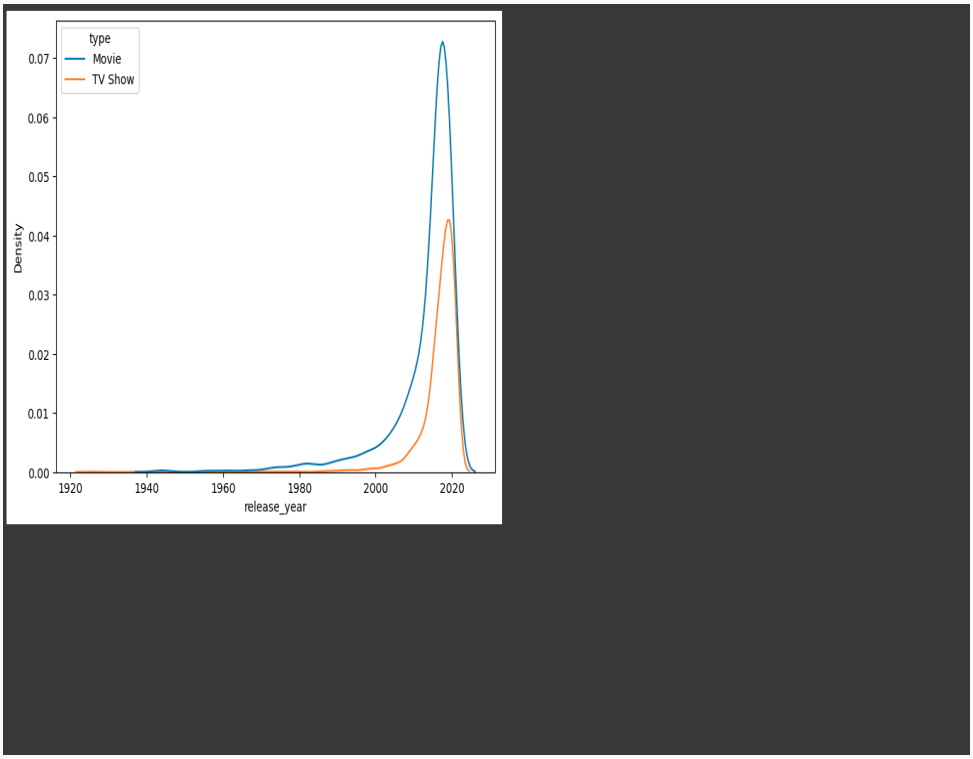
3. Non-Graphical Analysis: Value counts and unique attributes

```
df.nunique()
```

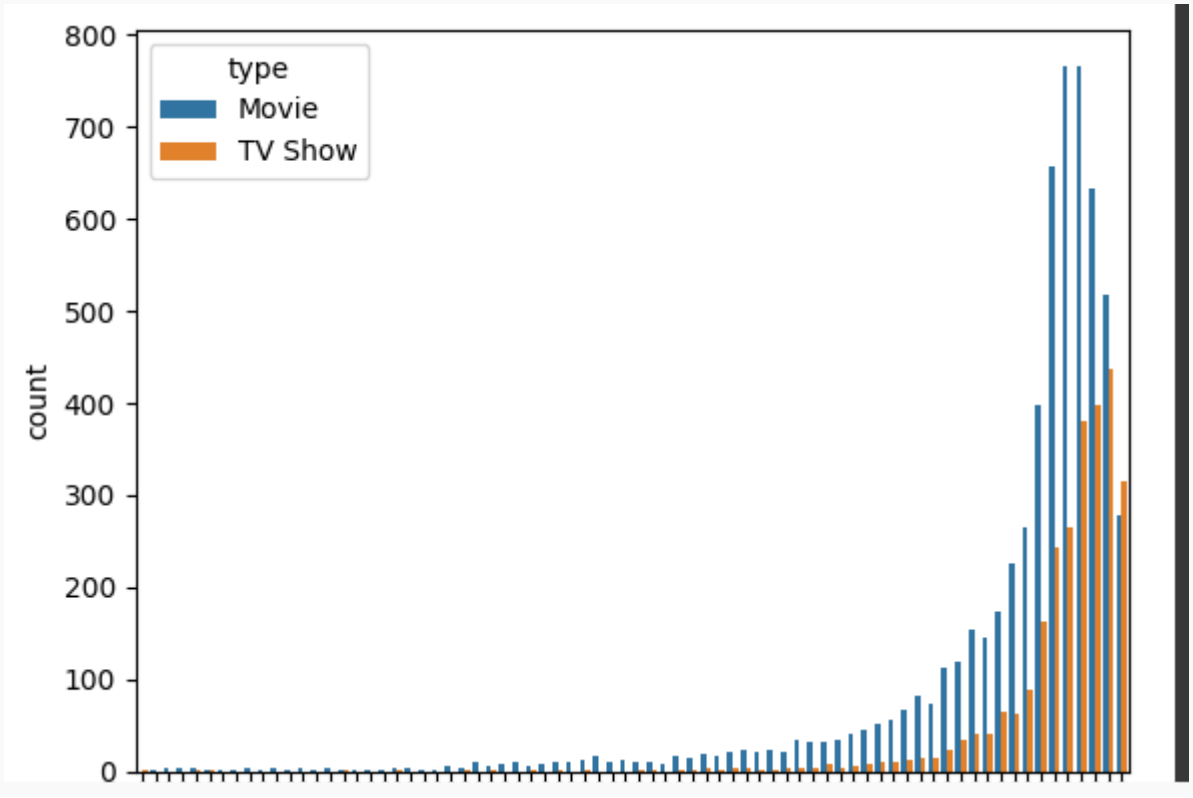
```
show_id         8807
type               2
title           8807
director        4528
cast            7692
country          748
date_added      1767
release_year      74
rating            17
duration         220
listed_in        514
description     8775
dtype: int64
```

4:1: Distplot plot for Movies and Tv Shows

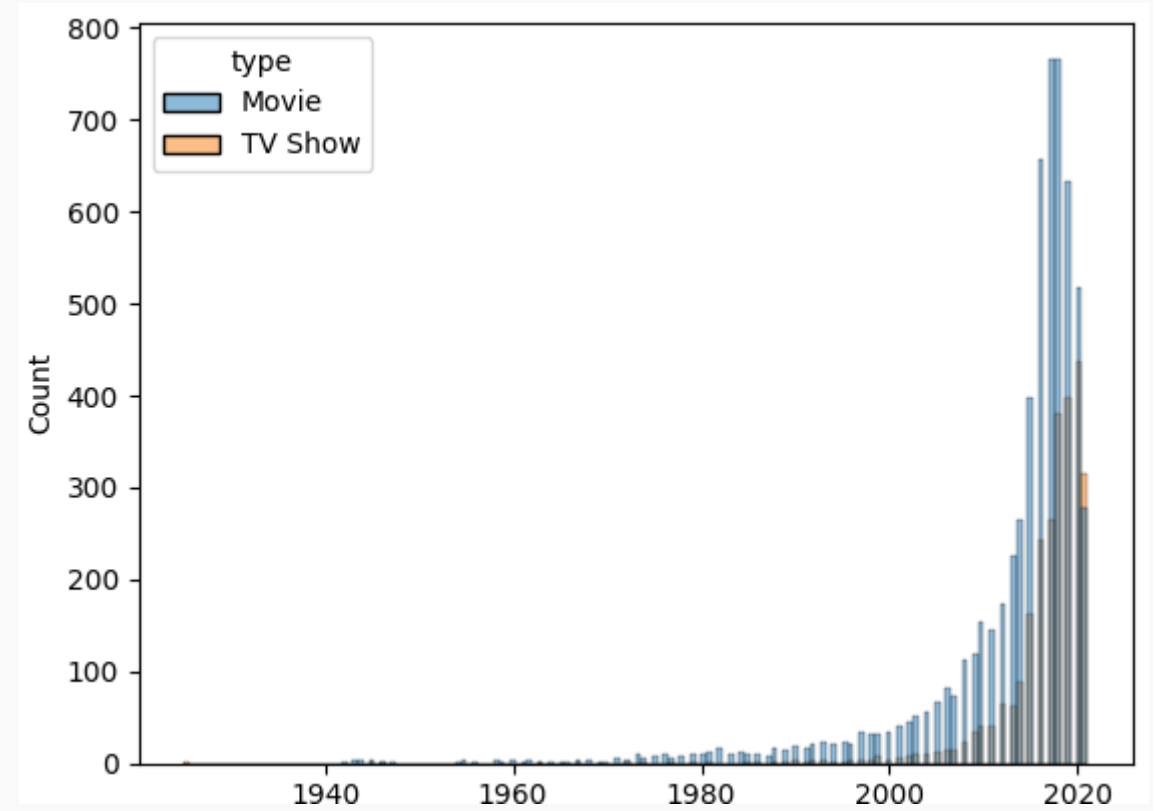
```
plt.figure(figsize=(10,8))
sns.displot(data=df, x="type")
plt.show()
```



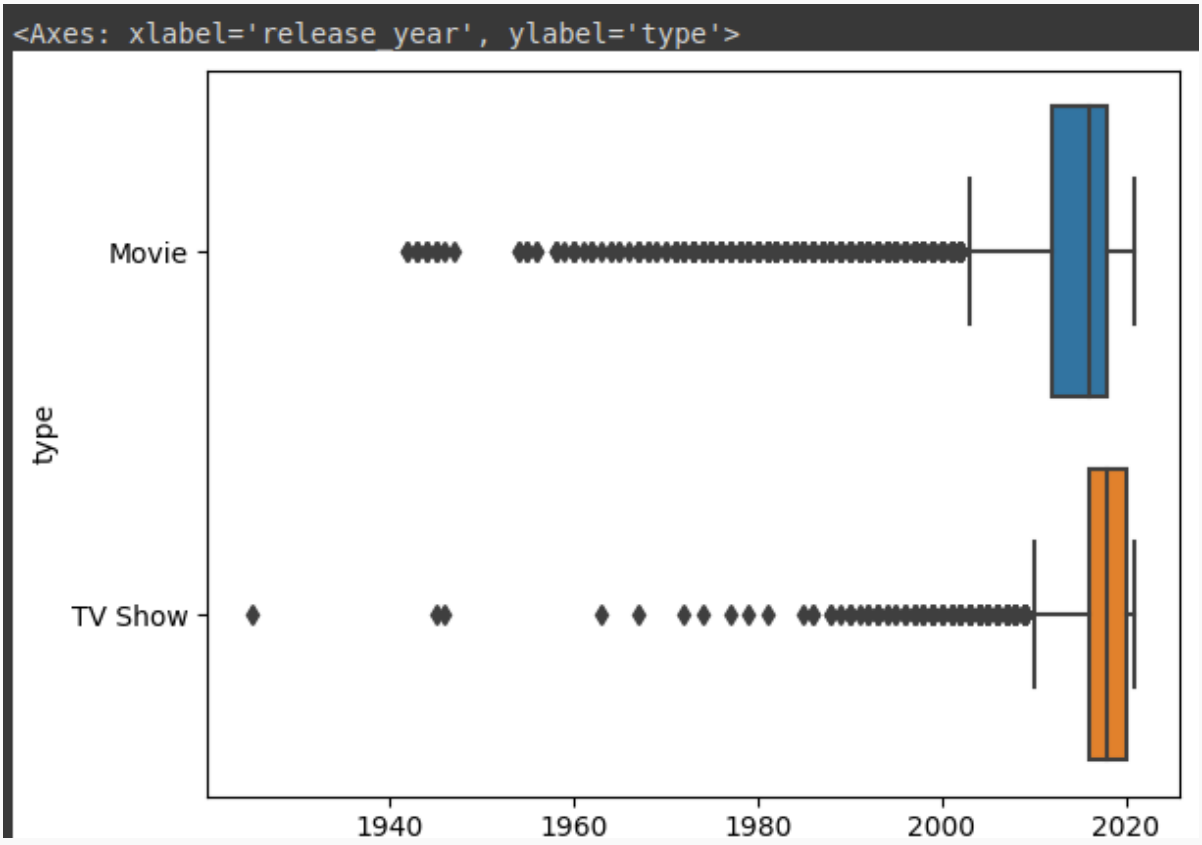

```
# count countplot
sns.countplot(data=df,x='release_year',hue='type')
plt.show()
```



#Histogram plot



4.2 For categorical variable(s): Boxplot



4.3 For correlation: Heatmaps, Pairplots

Heatmaps

```
plt.figure(figsize=(4,3))
sns.heatmap(data=movie.corr(),
            annot=True,
            cmap="coolwarm")
plt.title("heatmap")
plt.show()
```

Pairplots:-

```
movie[["new_dur","n"]]=movie["duration"].str.split(" ",expand=True)
movie["new_dur"]=movie["new_dur"].astype(int)
sns.pairplot(data=movie)
plt.show()
```

5. Missing Value & Outlier check (Treatment optional)

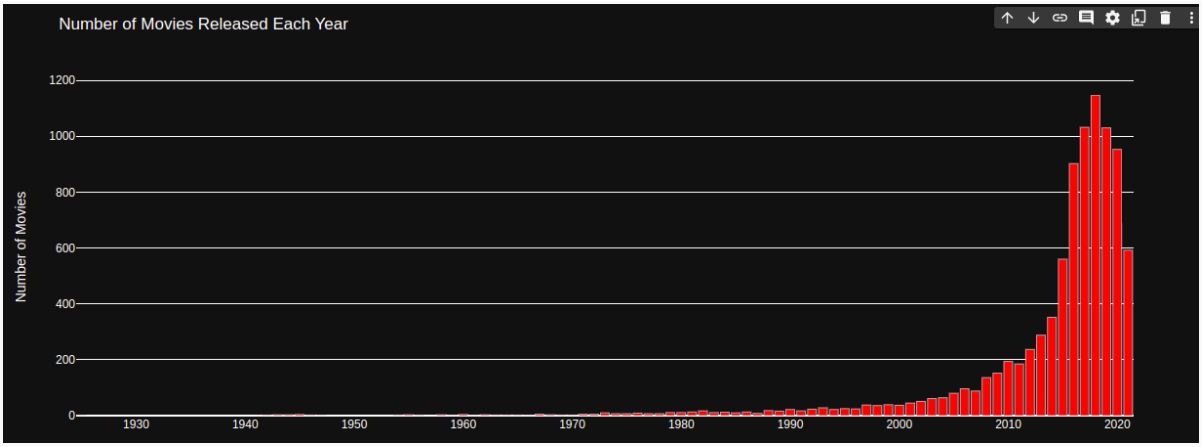
```
# fill null values with 'unknown colurms'
# df.isnull().sum()
df['country'].fillna('unknown country',inplace=True)
df['cast'].fillna('unknown cast',inplace=True)
df['date_added'].fillna(df['date_added'],inplace=True)
df['duration'].fillna(df['duration'].mean(),inplace=True)

df['duration'].fillna('unknown duration',inplace=True)
```

5.1 outlier check

```
import plotly.graph_objects as go
movies = df.loc[df['type']=='Movie']
movie_counts = movies['release_year'].value_counts().sort_index()
fig = go.Figure(data=go.Bar(x=movie_counts.index, y=movie_counts.values))
# Set the dark background and white font color
fig.update_layout(
        plot_bgcolor='rgb(17, 17, 17)',  # Dark background color
        paper_bgcolor='rgb(17, 17, 17)',  # Dark background color for the plot area
        font_color='white',  # White font color
        title='Number of Movies Released Each Year',  # Chart title
        xaxis=dict(title='Year'),  # X-axis label
```

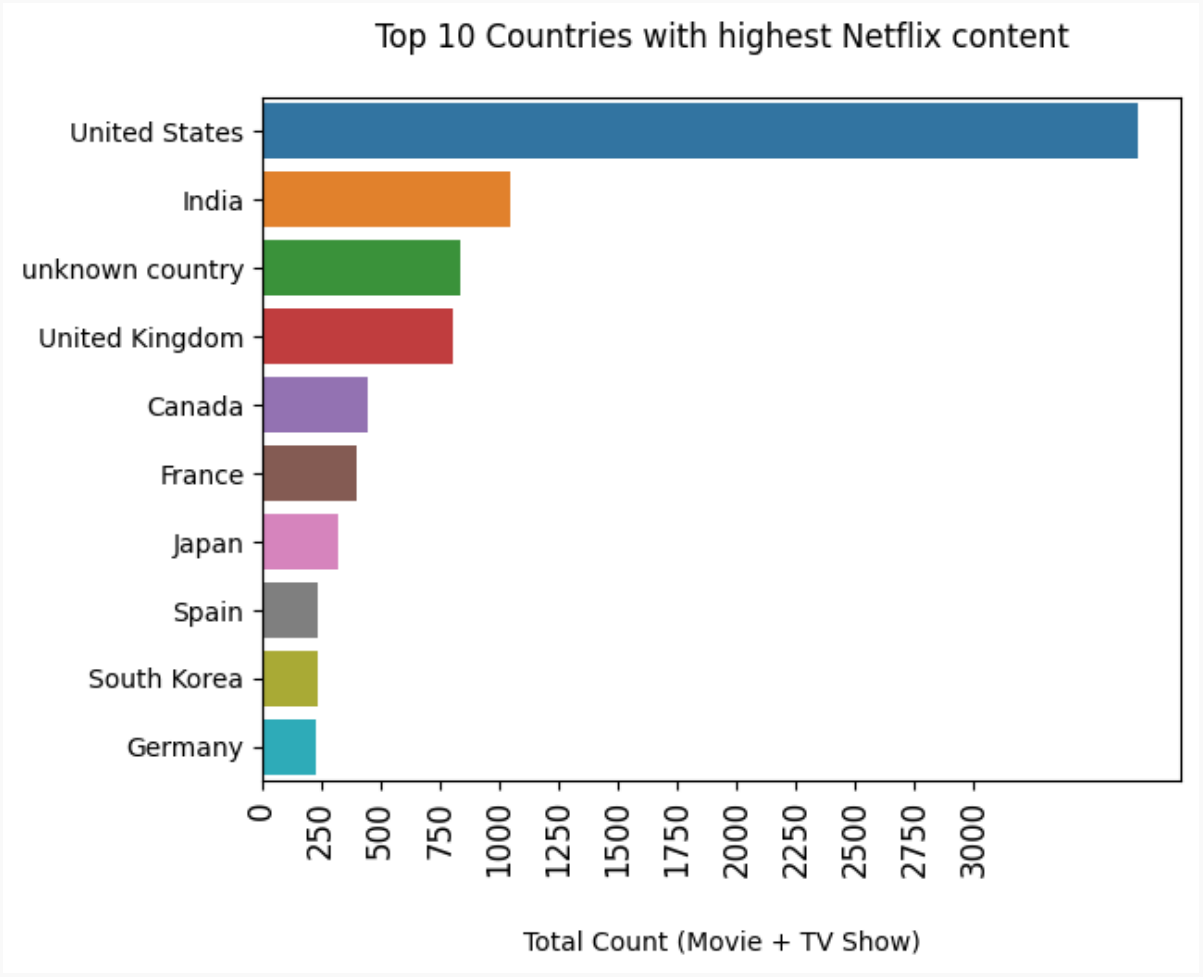
```
        yaxis=dict(title='Number of Movies')  # Y-axis label
)
fig.update_traces(marker_color='red')
fig.show()
```



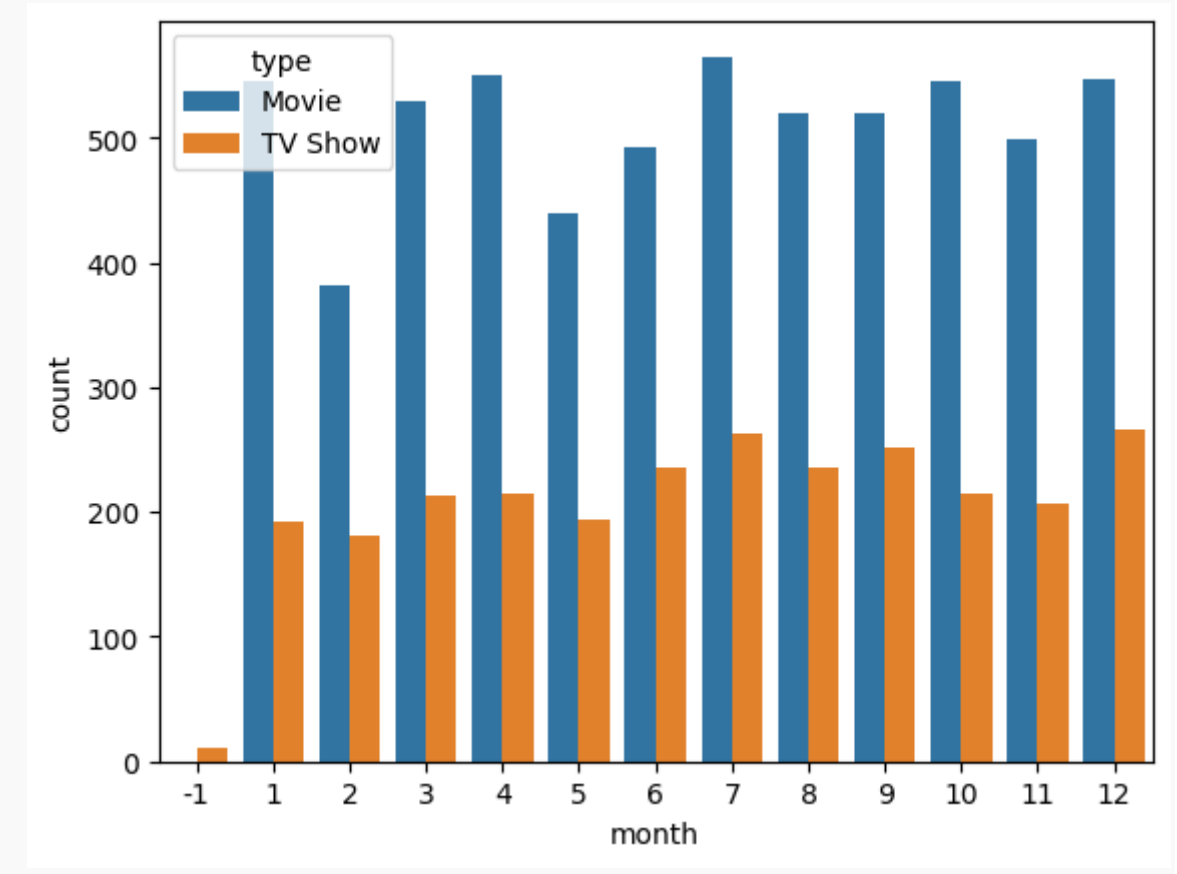
6:Insights based on Non-Graphical and Visual Analysis

6.1 Comments on the range of attributes

```
#Taking top 10
countries = country_true_count.head(10).index
Content_count = country_true_count.head(10).values
sns.barplot(y=countries,x=Content_count);
plt.title("Top 10 Countries with highest Netflix content",y=1.05);
plt.xticks(rotation=90 ,fontsize=12)
plt.xlabel('Total Count (Movie + TV Show)',labelpad=20);
plt.xticks(range(0,3250,250));
```



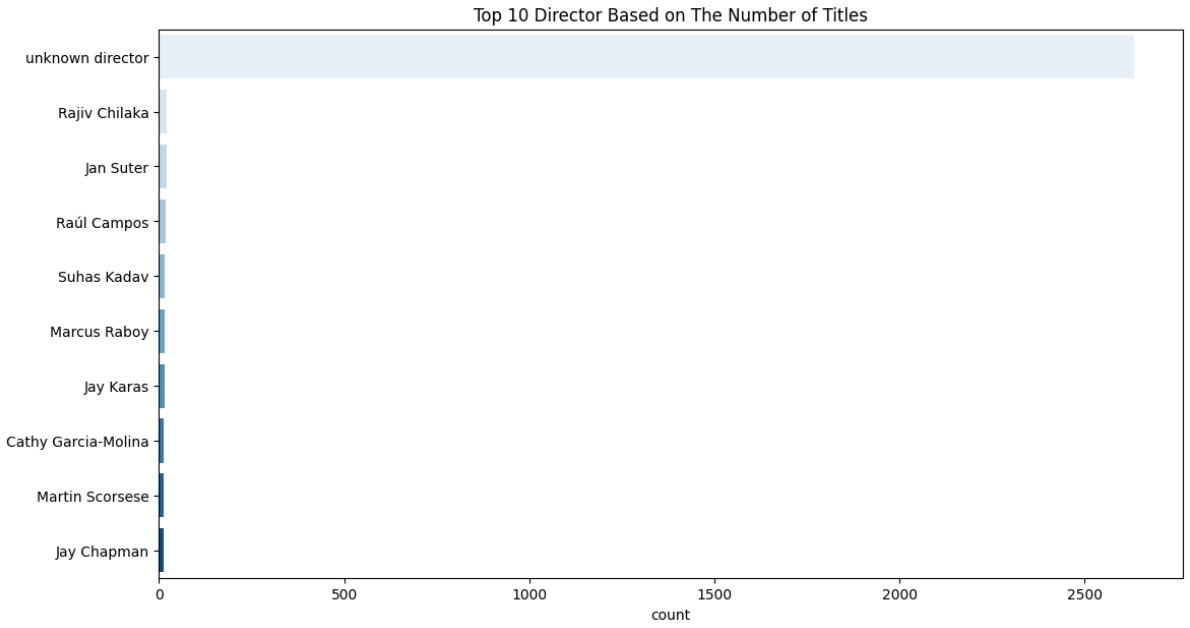
6.2 Comments on the distribution of the variables and relationship between them



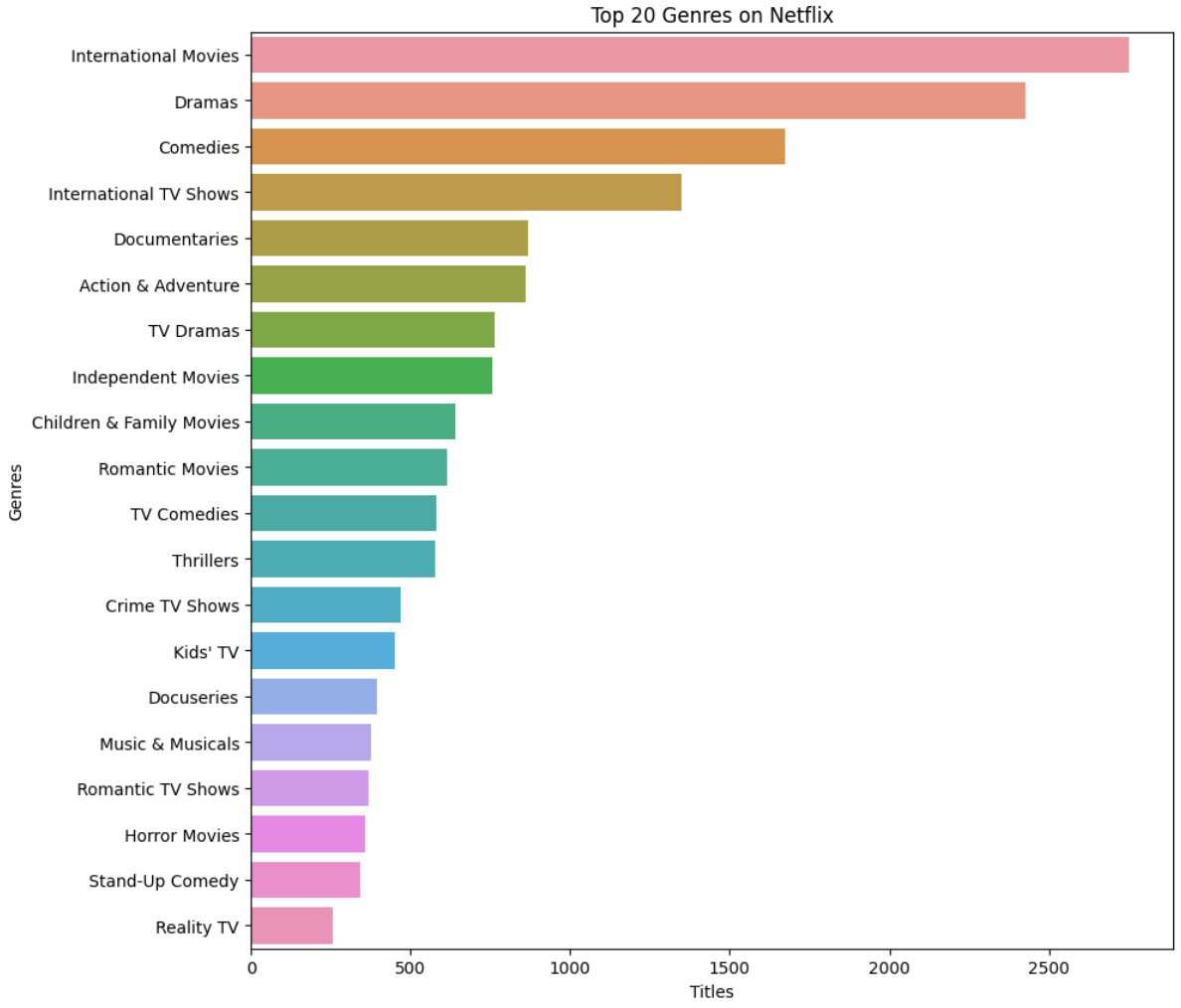
6.3 Comments for each univariate and bivariate plot

#To know the most popular director, we can visualize it.

```
filtered_directors = df[df.director != 'No
Director'].set_index('title').director.str.split(', ',
expand=True).stack().reset_index(level=1, drop=True)
plt.figure(figsize=(13,7))
plt.title('Top 10 Director Based on The Number of Titles')
sns.countplot(y = filtered_directors,
order=filtered_directors.value_counts().index[:10], palette='Blues')
plt.show()
```



Top Genres on Netflix



7. Business Insights (10 Points) - Should include patterns observed in the data along with what you can infer from it

- Netflix's main revenue is from **Movies** and main market is of **United States and India** . However, in United States, Netflix is more focussing on TV shows recently
- Netflix has different content strategies for different countries. For instance, longer movie duration content for India, Anime series in Japan, Romantic Shows in South Korea.
- More than 70% content is for **teens and mature audience**
- Majority of content is uploaded on **Friday** and in the Months from **October to January**

8:Recommendations - Actionable items for business. No technical jargon. No complications. Simple action items that everyone can understand

Amount of Content by Rating (Movies vs TV Shows)