Business Case: Netflix - Data Exploration and Visualisation

Netflix started as DVD rentals back in 1997, but now they provide access to best-in-class TV series, documentaries, feature films and games with streaming in more than 30 languages and 190 countries. The company's primary business is its subscription-based streaming service.

1. Problem Statement and basic metrics

Aim of this case study is to understand and find valuable insights which helps Netflix to identify profitable sources that can be produced in future with available data.

To analyse the data statistically as well as visually, we need to import popular python libraries like
pandas, numpy, matplotlib and seaborn

```python
# lets import libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```python
# lets import data
df = pd.read_csv('netflix analysis /netflix.csv')
```

Let's import data and put it in a new data frame in the name as (df) and understand its characters by using **head()** function, by default it retrieves the first 5 records.

```python
df.head()
```

show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description

```python
# lets understand the characteristics of the data by using some basic functions
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	Documentaries	As her father nears the end of his life, filmm...
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	International TV Shows, TV Dramas, TV Mysteries	After crossing paths at a party, a Cape Town t...
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA	1 Season	Crime TV Shows, International TV Shows, TV Act...	To protect his family from a powerful drug lor...

tail() function, retrieves last 5 records

```python
df.tail()
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in	description
8802	s8803	Movie	Zodiac	David Fincher	Mark Ruffalo, Jake Gyllenhaal, Robert Downey J...	United States	November 20, 2019	2007	R	158 min	Cult Movies, Dramas, Thrillers	A political cartoonist, a crime reporter and a...
8803	s8804	TV Show	Zombie Dumb	NaN	NaN	NaN	July 1, 2019	2018	TV-Y7	2 Seasons	Kids' TV, Korean TV Shows, TV Comedies	While living alone in a spooky town, a young g...
8804	s8805	Movie	Zombieland	Ruben Fleischer	Jesse Eisenberg, Woody Harrelson, Emma Stone, ...	United States	November 1, 2019	2009	R	88 min	Comedies, Horror Movies	Looking to survive in a world taken over by zo...

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

shape() → retrieves count of rows and columns,
ndim() → retrieves dimension of data set,
column → retrieves list of columns
dtypes → retrieves column data types

```
# lets understand more about data and use some specific techniques to overcome missing values.

df.shape

(8807, 12)

df.ndim

2

df.columns

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

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
```

Missing value detection

Missing data are major problem in data exploration but it can be rectifiable.There are many ways we can overcome missing values, some of them are like deleting rows/columns, replaced with mean/median/mode for predicting modelling, and imputation. In order to check null values in Pandas DataFrame, we use **isnull()** function. This function returns a data frame of Boolean values which are True for NaN values.

```
print('\n Columns with missing values: ')
print(df.isnull().any())
```

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

```
# missing data ratio
for i in df.columns:
    null_rate = df[i].isna().sum() / len(df) * 100
    if null_rate > 0 :
        print("{} null rate: {}%".format(i,round(null_rate,2)))
```

```
director null rate: 29.91%
cast null rate: 9.37%
country null rate: 9.44%
date_added null rate: 0.11%
rating null rate: 0.05%
duration null rate: 0.03%
```

As per the above results, missing values are in columns like director, cast, country and few in rating and duration. We can also calculate how many values are missing using **isnull().sum()** functions

```
# lets also calculate count of missing values in all columns
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
```

Treating Missing values

Missing values can be treated using
fillna() → to fill missing values
dropna() → to drop missing values

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

Statistical Summary

Statistical approach is helpful to understand count of values, mean(average), median, minimum ,maximum and also quartile range. Let’s understand its statistical result before cleaning the data i.e. overcoming missing and duplicate values using **describe()** function.

```
df.describe() # statistical summary before cleaning the data
```

	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

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

value_counts() → retrieves all the values
unique() → retrieves unique values and **nunique()** retrieves count of unique values

```
df.value_counts()
```

```
show_id  type   title                director            cast
country                                                date_added          release_year  rating  duration  listed_in
description
s10      Movie  The Starling         Theodore Melfi      Melissa McCarthy, Chris O'Dowd, Kevin Kline, Timothy Olyphant, Daveed Diggs, Skyle
```

```
df['listed_in'].unique()
```

```
array(['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',
```

```
df['listed_in'].nunique()
```

```
514
```

```
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
```

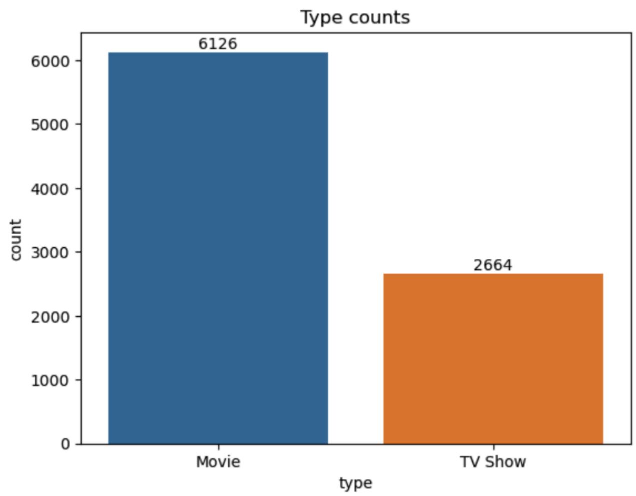
```
df['rating'].unique()
```

```
array(['PG-13', 'TV-MA', 'PG', 'TV-14', 'TV-PG', 'TV-Y', 'TV-Y7', 'R',
       'TV-G', 'G', 'NC-17', '74 min', '84 min', '66 min', 'NR', nan,
       'TV-Y7-FV', 'UR'], dtype=object)
```

4. Visual Analysis - Univariate, Bivariate

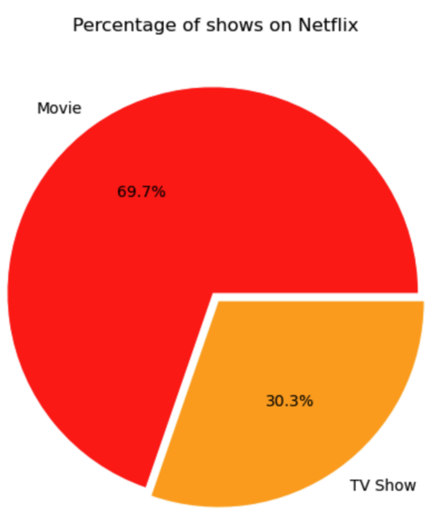
Visual analysis helps us to understand the data more easily with various visual supports, we can use bar plots, pie charts, histograms, scatterplots and many more. It is also easy to compare relations between variables. Here we can do,
Univariate → 1 variable analysis | Bivariate → 2 variables analysis | Multivariate → 3+ variables analysis

```
ax = sns.countplot(x ="type" ,data=df)
plt.title("Type counts")
ax.bar_label(ax.containers[0])
plt.show()
```



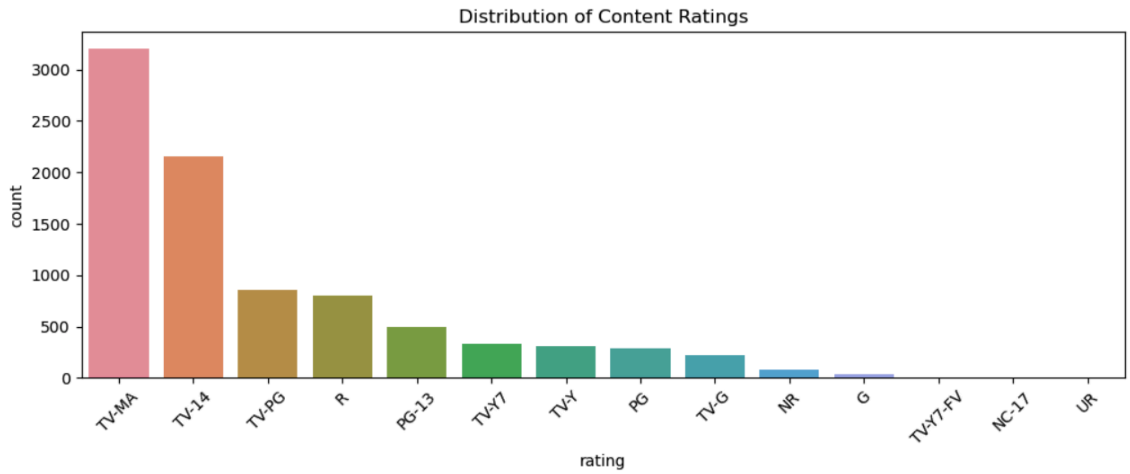
Using **countplot()** we can count results of both Movies and TV Shows. There are 6126 Movies and 2664 TV Shows were uploaded on Netflix, which clearly represents most of the content in the Movie category.

```
# also we can find out its ratio using (pie chart)
plt.figure(figsize=(10,6))
plt.title("Percentage of shows on Netflix")
pie = plt.pie(df.type.value_counts(), explode=(0.025,0.025),
              labels= df.type.value_counts().index,
              colors =['red','orange'], autopct = '%1.1f%%')
plt.show()
```



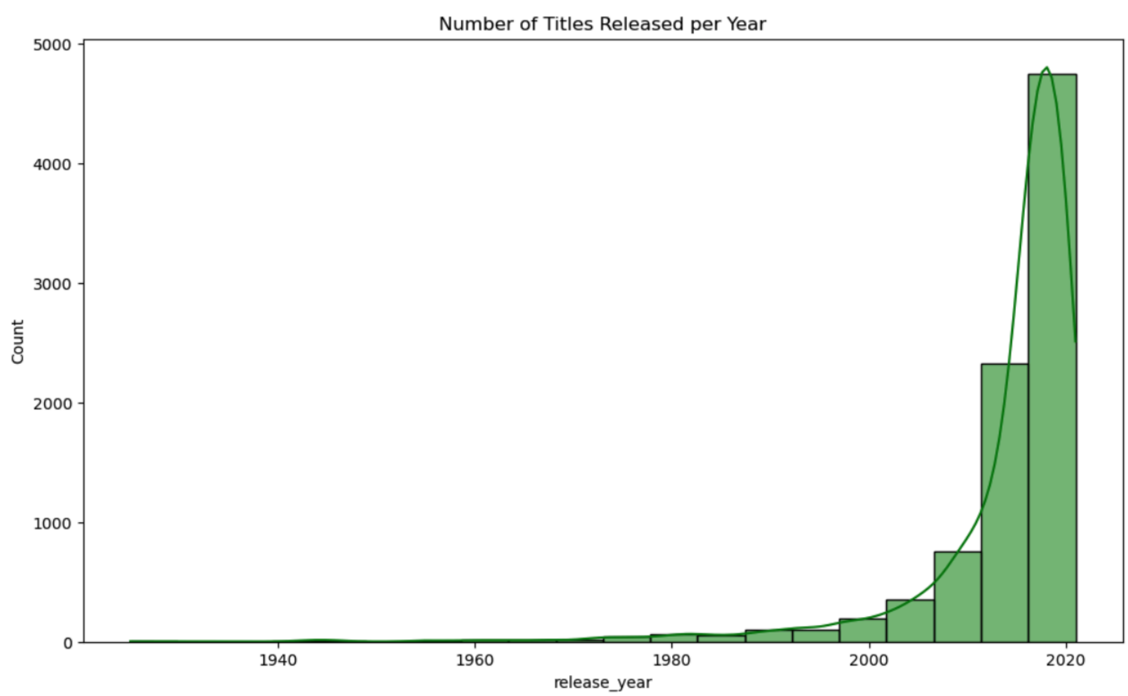
Using **pie-chart** we have found the ratio between Movies and TV shows , which clearly represents Movies are almost 70 % and TV shows are 30%

```
country_10 = df["country"].value_counts().head(10)
sns.barplot(x = country_10.index,y=country_10.values)
plt.title('Top 10 Countries Producing Netflix Content')
plt.xticks(rotation=90)
plt.show()
```

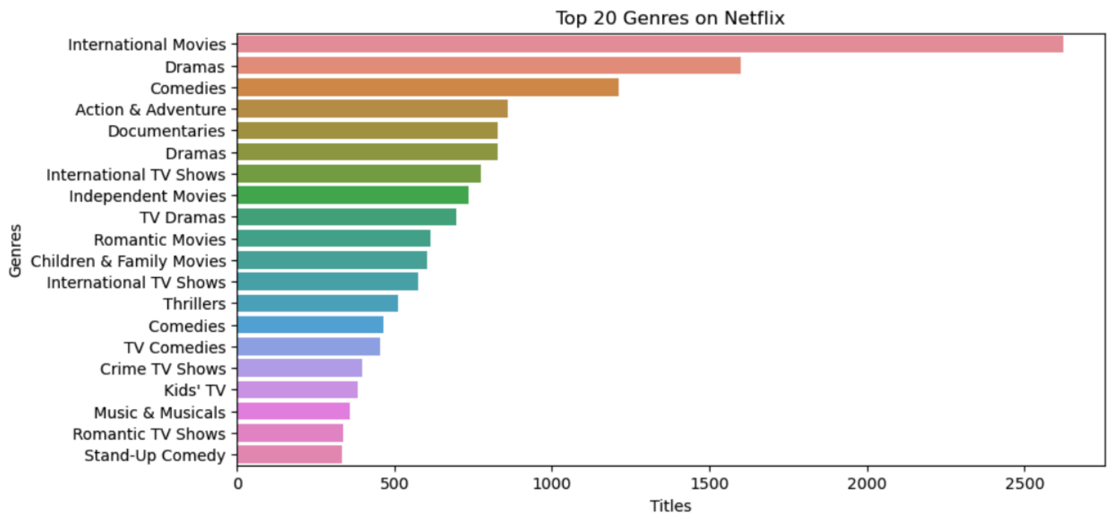


Using **barplot()** , we have found distribution of ratings of TV shows and movies, there are 17 ratings categories available on Netflix. Among them most rated category is TV-MA - (Mature AudienceOnly):Adults (17+) which also represents probably most of the Netflix users may belong above 17 years.

```
# lets use (histplot) to understand number of titles relased per year
plt.figure(figsize=(12,7))
sns.histplot(data=df,x="release_year",kde=True,bins = 20,color = "green")
plt.title('Number of Titles Released per Year')
plt.show()
```



```
# lets find out top 20 genres on Netflix using countplot
genre = df.set_index('title').listed_in.str.split(',' , expand = True).stack().reset_index(level=1, drop = True)
plt.figure(figsize=(10,5))
g = sns.countplot(y= genre,order = genre.value_counts().index[: 20])
plt.title('Top 20 Genres on Netflix')
plt.xlabel('Titles')
plt.ylabel('Genres')
plt.show()
```



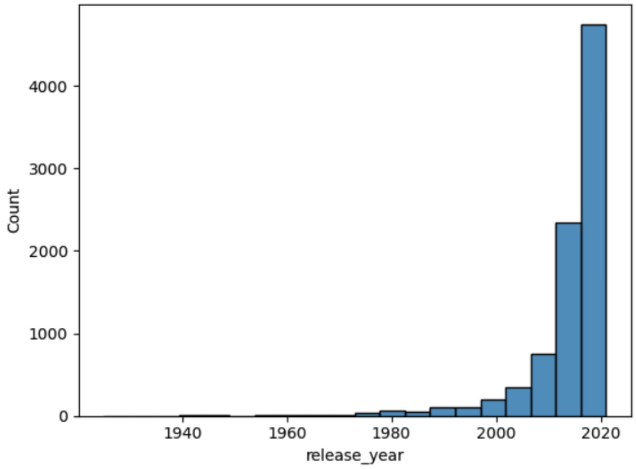
Top 20 genres were deducted using **countplot()**, International Movies are on top of all, which represents people preferring International genre movies from all around the world. Now lets understand most watching viewers from all over the world.

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

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


```
sns.histplot(df['release_year'], bins = 20)
```

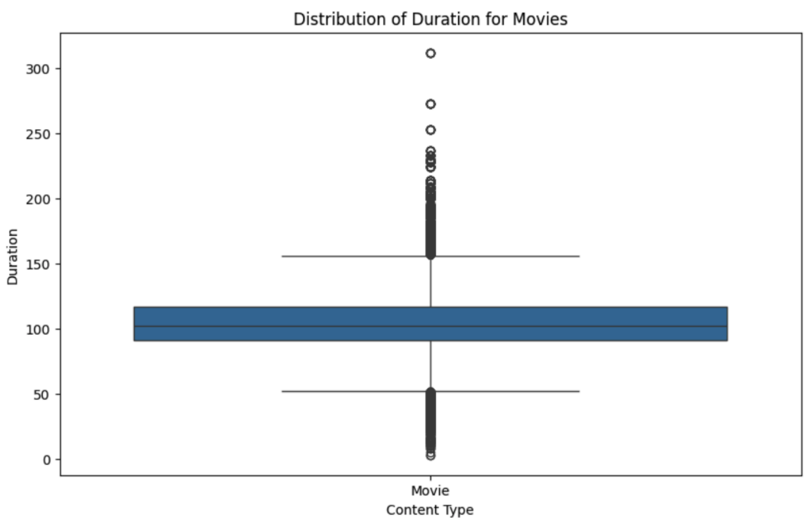
<Axes: xlabel='release_year', ylabel='Count'>




```
netflix_movies_df = df[df.type.str.contains("Movie")]
netflix_movies_df['duration'] = netflix_movies_df['duration'].str.extract('(\d+)',expand=False).astype(int)
```

Show hidden output

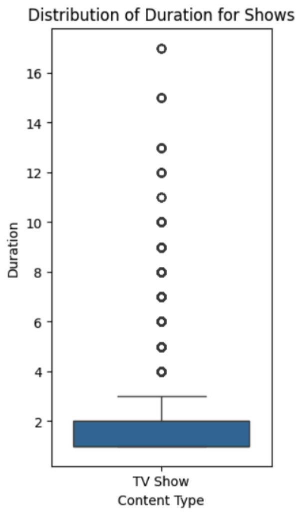
```
# Creating a boxplot for movie duration
plt.figure(figsize=(10, 6))
sns.boxplot(data=netflix_movies_df, x='type', y='duration')
plt.xlabel('Content Type')
plt.ylabel('Duration')
plt.title('Distribution of Duration for Movies')
plt.show()
```




```
[98] netflix_shows_df = df[df.type.str.contains("TV Show")]
     netflix_shows_df['duration'] = netflix_shows_df['duration'].str.extract('(\d+)',expand=False).astype(int)
```

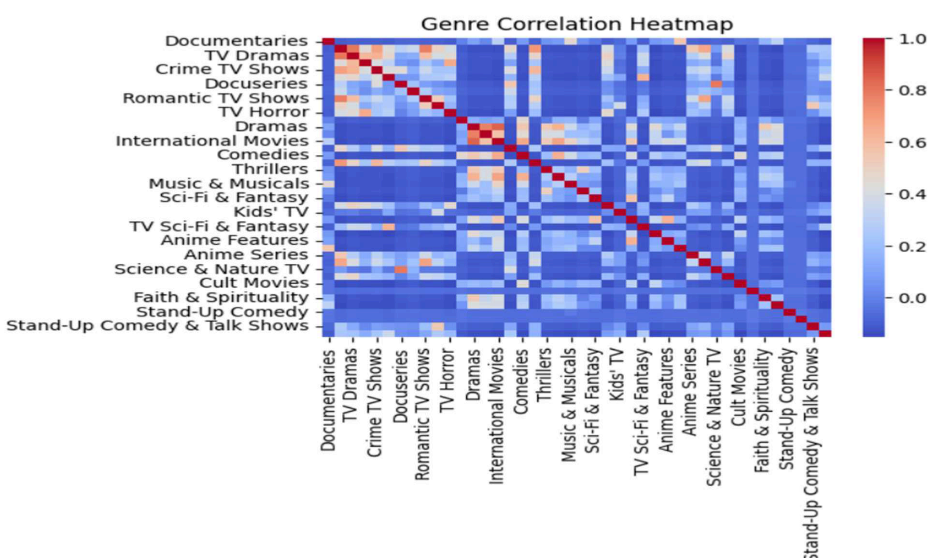
Show hidden output

```
# Creating a boxplot for movie duration
plt.figure(figsize=(3, 6))
sns.boxplot(data=netflix_shows_df, x='type', y='duration')
plt.xlabel('Content Type')
plt.ylabel('Duration')
plt.title('Distribution of Duration for Shows')
plt.show()
```



Heatmaps, pairplots

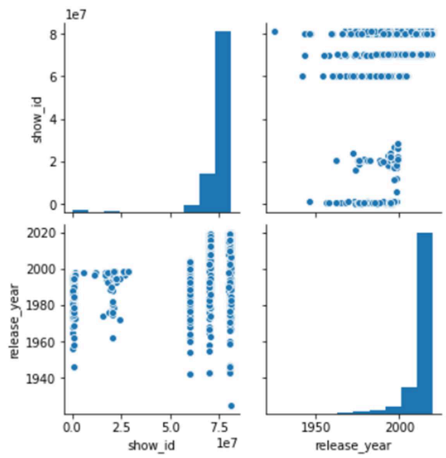
Heat Maps is a type of plot which is necessary when we need to find the dependent variables. One of the best ways to find the relationship between the features can be done using heat maps. Genres play a significant role in categorising and organising content onNetflix. Analysing the correlation between genres can reveal interesting relationships between different types of content. We create a genre data DataFrame to investigate genre correlation and fill it with zeros. By iterating over each row in the original DataFrame, we update the genre data DataFrame based on the listed genres. We then create a correlation matrix using this genre data and visualise it as a heatmap.



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

Pairplots

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



5. **Missing value & Outlier check**

Outlier

An outlier is a point or set of points that are different from other points. Sometimes they can be very high or very low. It's often a good idea to detect and remove the outliers. Because outliers are one of the primary reasons for resulting in a less accurate model. Hence it's a good idea to remove them. The outlier detection and removing that I am going to perform is called IQR score technique. Often outliers can be seen with visualisations using a box plot. Shown below are the box plots of movies and tv shows distribution.

For example, let us consider a row of data [10,15,22,330,30,45,60]. In this dataset, we can easily conclude that 330 is way off from the rest of the values in the dataset, thus 330 is an outlier. It was easy to figure out the outlier in such a small dataset, but when the dataset is huge, we need various methods to determine whether a certain value is an outlier or necessary information.

Why do we need to treat outliers?

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

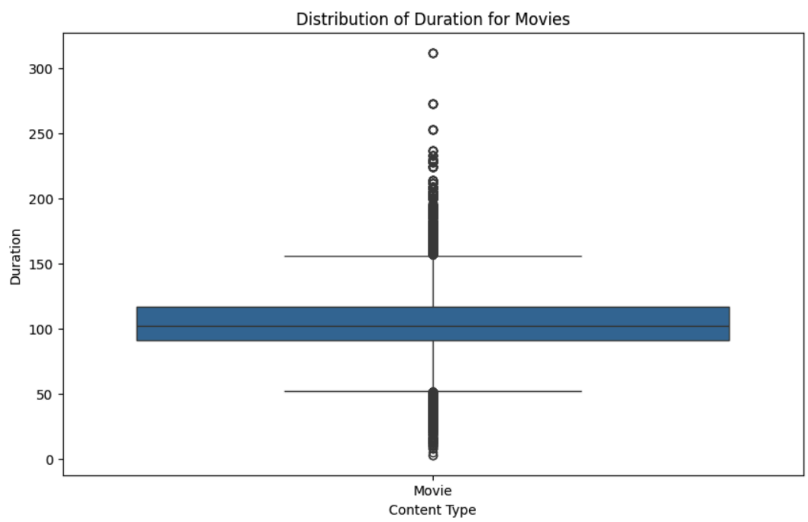
Visual Detection

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

```
netflix_movies_df = df[df.type.str.contains("Movie")]
netflix_movies_df['duration'] = netflix_movies_df['duration'].str.extract('(\d+)',expand=False).astype(int)
```

Show hidden output

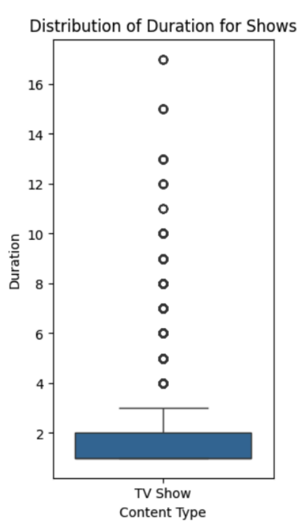
```
# Creating a boxplot for movie duration
plt.figure(figsize=(10, 6))
sns.boxplot(data=netflix_movies_df, x='type', y='duration')
plt.xlabel('Content Type')
plt.ylabel('Duration')
plt.title('Distribution of Duration for Movies')
plt.show()
```




```
[98] netflix_shows_df = df[df.type.str.contains("TV Show")]
     netflix_shows_df['duration'] = netflix_shows_df['duration'].str.extract('(\d+)',expand=False).astype(int)
```

Show hidden output

```
# Creating a boxplot for movie duration
plt.figure(figsize=(3, 6))
sns.boxplot(data=netflix_shows_df, x='type', y='duration')
plt.xlabel('Content Type')
plt.ylabel('Duration')
plt.title('Distribution of Duration for Shows')
plt.show()
```



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

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

Detecting missing values
There are several ways to detect missing values in Python. isnull() function is widely used for the same purpose.

```
print('\n Columns with missing values: ')
print(df.isnull().any())
```

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

```
# missing data ratio
for i in df.columns:
    null_rate = df[i].isna().sum() / len(df) * 100
    if null_rate > 0 :
        print("{} null rate: {}%".format(i,round(null_rate,2)))
```

```
director null rate: 29.91%
cast null rate: 9.37%
country null rate: 9.44%
date_added null rate: 0.11%
rating null rate: 0.05%
duration null rate: 0.03%
```

To check total number of missing values in each column , we can use isnull().sum() function

```
# lets also calculate count of missing values in all columns
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
```

There are a total of 4307 null values across the entire dataset with 2634 missing points under "director", 825 under "cast", 831 under "country", 11 under "date_added", 4 under "rating" and 3 under “duration ”. These null values were rectified using fillna() function.

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

After cleaning the null values

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

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

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

6. **Insights based on Non-Graphical and Visual Analysis**

- Range between recently released movies and old ones are vast in number. When using statistical approaches i.e. describe() function to understand the range , it results 1925 as minimum and 2021 as the maximum value. Which resembles Netflix has 96 years of visual treat in their platform.
- Most interesting fact is Genre, netflix has more than 50 genres, where top 10 genres are "International Movies", "Dramas", "Comedies", "International TV shows", "Documentaries", "Action and Adventure", "TV Dramas", "Independent Movies", "Children and Family Movies", "Romantic Movies".
- Ratio between TV shows and Movies were deducted using pie chart and it results 30% and 70% which shows most of the content are Movies
- Using Barplot, distribution of content ratings were found. In Netflix data there are 17 different types of rating content available among them here are top 5 ratings:
  TV-MA (Mature Audience Only):Adults (17+)
  TV-14 (Parents Strongly Cautioned):Children under 14 may require parental guidance.
  TV-PG (Parental Guidance Suggested):Suitable for general audiences but not for
children
  R (Restricted):Restricted to viewers aged 17
  PG-13 (Parents Strongly Cautioned): Suitable for viewers aged 13 and older, suitable with parents guidance.
- Most of the distributed movie content were analysed using box plot duration lies between 120 mins, which is probably acceptable duration to engage an audience

7. **Business Insights**

With the help of this article, we have been able to learn about-

1. Quantity: Our analysis revealed that Netflix had added more movies than TV shows, aligning with the expectation that movies dominate their content library.

2. Content Addition: July emerged as the month when Netflix adds the most content, closely followed by December, indicating a strategic approach to content release.

3. Genre Correlation: Strong positive associations were observed between various genres, such as TV dramas and international TV shows, romantic and international TV shows, and independent movies and dramas. These correlations provide insights into viewer preferences and content interconnections.

4. Movie Lengths: The analysis of movie durations indicated a peak around the 1960s, followed by a stabilisation around 100 minutes, highlighting a trend in movie lengths over time.

5. TV Show Episodes: Most TV shows on Netflix have one season, suggesting a preference for shorter series among viewers.

6. Common Themes: Words like love, life, family, and adventure were frequently found in titles and descriptions, capturing recurring themes in Netflix content.

7. Rating Distribution: The distribution of ratings over the years offers insights into the evolving content landscape and audience reception.

8. Data-Driven Insights: Our data analysis journey showcased the power of data in unravelling the mysteries of Netflix's content landscape, providing valuable insights for viewers and content creators.

9. Continued Relevance: As the streaming industry evolves, understanding these patterns and trends becomes increasingly essential for navigating the dynamic landscape of Netflix and its vast library.

10. Happy Streaming: We hope this blog has been an enlightening and entertaining journey into the world of Netflix, and we encourage you to explore the captivating stories within its ever-changing content offerings. Let the data guide your streaming adventures!

8. **Recommendations**

- Netflix has to focus on TV Shows also because there are people who will like to see tv shows rather than movies
- By approaching the top director we can plan some more movies/tv shows in order to increase the popularity
- Not only reaching top directors, we can also see the director with fewer movies and having high ratings as there may be some financial issues or anything so in order to get

good content netflix can reach them and netflix can produce the movie and give the director a chance.

- We have seen most no of international movies genre so need to give priority to other genres like horror,comedy..etc

- In TV Shows we may focus on thriller genre which will be helpful for having more number of seasons

- Most of the movies released in ott is in a year 2019 so we need to go on increasing this value in order to attract people by showing that getting subscription is useful as netflix is releasing more movies per year

- Mainly the release in ott should focus on the festival holidays, year end and weekends which is to be mainly focussed

- Some movies can be released directly into ott which has some positive talk which may help in improving subscriptions

- Should focus on a actor who has immense following and make use of it by doing a TV Shows or web series

- Advertisement in the country which has very less movies released should be increased and attract people of that country by making their native TV Shows