PREDICTING IMDB SCORES

PHASE 3

GIVEN DATASET:

SOURCE:

https://www.kaggle.com/datasets/luiscorter/netflix-original-films-imdb-scores

DATASET EXPLANATION:

This dataset consists of all Netflix original films released as of June 1st, 2021. Additionally, it also includes all Netflix documentaries and specials. The data was webscraped off, which was then integrated with a dataset consisting of all of their corresponding IMDB scores. IMDB scores having majority of the films have 1,000+ reviews.

CONTENT INCLUDED IN THE DATASET IS,

- Title of the film
- Genre of the film
- Original premiere date

- Runtime in minutes
- IMDB scores (as of 06/01/21)
- Languages currently available (as of 06/01/21)

EXPLORATION AND PREPROCESSING:

Loading the Dataset:

Here, I'm going to load the dataset in google colab

Before that importing libraries is mandatory,

Importing Libraries:

```
import pandas as pd
import numpy as np

#Data Visulation
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.express as px
import plotly.graph_objects as go

from datetime import datetime

import statsmodels.api as sm

from warnings import filterwarnings
filterwarnings("ignore")
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
```

After importing libraries, load the dataset here how it looks like,

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

Choose Files NetflixOriginals.csv

- **NetflixOriginals.csv**(text/csv) - 38678 bytes, last modified: 10/11/2023 - 100% done

Saving NetflixOriginals.csv to NetflixOriginals (1).csv

Reading the Dataset:

```
movie=pd.read_excel("NetflixOriginals.xlsx",encoding = "ISO-8859-1")
print(movie)
dataDate=movie.copy()
```

After run the code, output will be shown as,

```
                                          Title                  Genre  \
0                                Enter the Anime            Documentary
1                                    Dark Forces               Thriller
2                                        The App  Science fiction/Drama
3                                 The Open House        Horror thriller
4                                    Kaali Khuhi                Mystery
..                                           ...                    ...
579        Taylor Swift: Reputation Stadium Tour           Concert Film
580  Winter on Fire: Ukraine's Fight for Freedom            Documentary
581                      Springsteen on Broadway           One-man show
582    Emicida: AmarElo - It's All For Yesterday            Documentary
583     David Attenborough: A Life on Our Planet            Documentary
```

Information About the Dataset:

The info() method provides a concise summary of the data in a pandas Data Frame. It typically includes the following information:

1. The number of non-null (non-missing) values in each column.
2. The data type of each column (e.g., int64, float64, object, datetime64, etc.).
3. The total memory usage of the Data Frame.

4. Additional information, such as the count, mean, standard deviation, minimum, and maximum values for numeric columns.

```
[ ]  movie.info()
```

Output:

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 584 entries, 0 to 583
Data columns (total 6 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   Title       584 non-null    object
 1   Genre       584 non-null    object
 2   Premiere    584 non-null    datetime64[ns]
 3   Runtime     584 non-null    int64
 4   IMDB Score  584 non-null    float64
 5   Language    584 non-null    object
dtypes: datetime64[ns](1), float64(1), int64(1), object(3)
memory usage: 27.5+ KB
```

Head() Function:

It serves the purpose of displaying the first few rows of the dataset.

```
[ ] movie.head()
```

Output:

	Title	Genre	Premiere	Runtime	IMDB Score	Language
0	Enter the Anime	Documentary	2019-08-05	58	2.5	English/Japanese
1	Dark Forces	Thriller	2020-08-21	81	2.6	Spanish
2	The App	Science fiction/Drama	2019-12-26	79	2.6	Italian
3	The Open House	Horror thriller	2018-01-19	94	3.2	English
4	Kaali Khuhi	Mystery	2020-10-30	90	3.4	Hindi

Describe() Function:

This is a Pandas Data Frame method that generates descriptive statistics of a dataset, typically for numeric columns.

```
movie.describe().T
```

Output:

	count	mean	std	min	25%	50%	75%	max
Runtime	584.0	93.577055	27.761683	4.0	86.0	97.00	108.0	209.0
IMDB Score	584.0	6.271747	0.979256	2.5	5.7	6.35	7.0	9.0

Checking For Missing Values:

The expression movie.isnull().values.any() is used to check whether there are any missing values in a Pandas DataFrame or a NumPy array. It returns a boolean value, indicating whether any missing values are present in the dataset.

```
[ ] movie.isnull().values.any()
```

Output:

```
False
```

VARIOUS ANALYSIS PERFORMED USING DATASET:

Descriptive Analysis:

It involves the process of summarizing and presenting data in a meaningful and informative way. The primary purpose of descriptive analysis is to provide a clear and concise overview of a dataset.

```python
# Display summary statistics of numerical columns
print(movie.describe())

# Calculate the mean IMDb score
mean_imdb_score = movie['IMDB Score'].mean()
print(f"Mean IMDb Score: {mean_imdb_score}")

# Count the number of movies in each content rating category
content_rating_counts = movie['Runtime'].value_counts()
print(content_rating_counts)
```

Output:

```
           Runtime  IMDB Score
count  584.000000  584.000000
mean    93.577055    6.271747
std     27.761683    0.979256
min      4.000000    2.500000
25%     86.000000    5.700000
50%     97.000000    6.350000
75%    108.000000    7.000000
max    209.000000    9.000000
Mean IMDb Score: 6.2717465753424655
97     24
98     19
94     19
95     18
100    17
       ..
148     1
147     1
7       1
57      1
153     1
Name: Runtime, Length: 124, dtype: int64
```

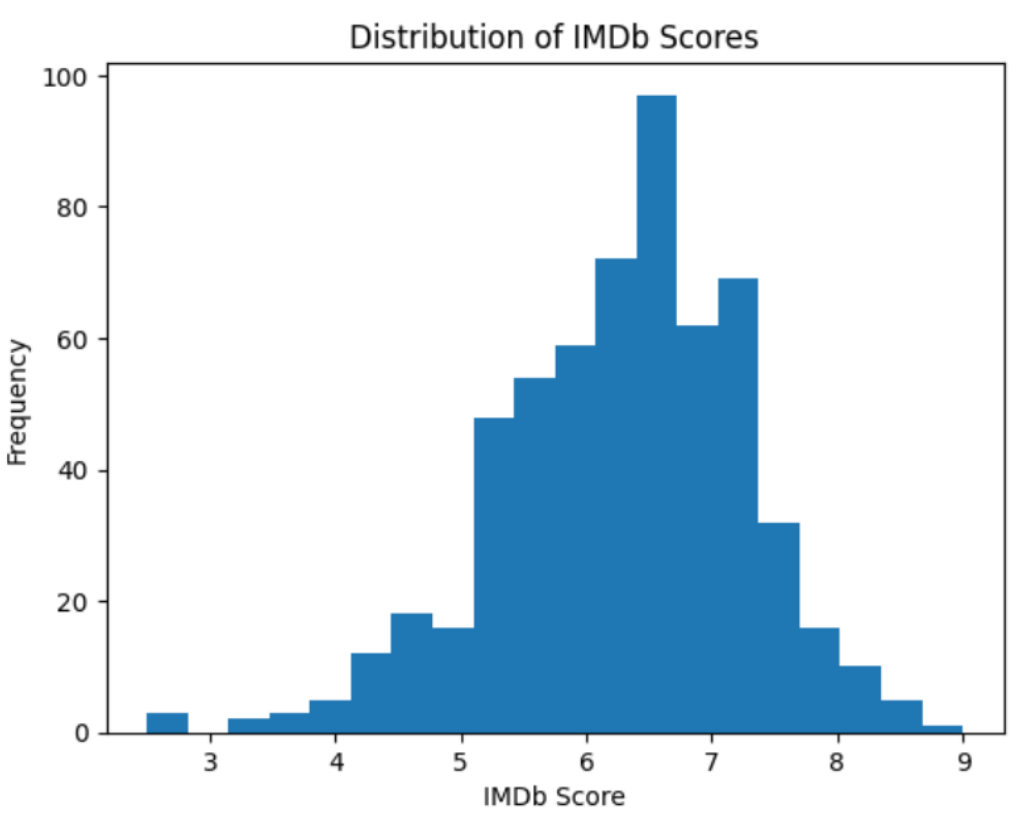
Data Visualisation:

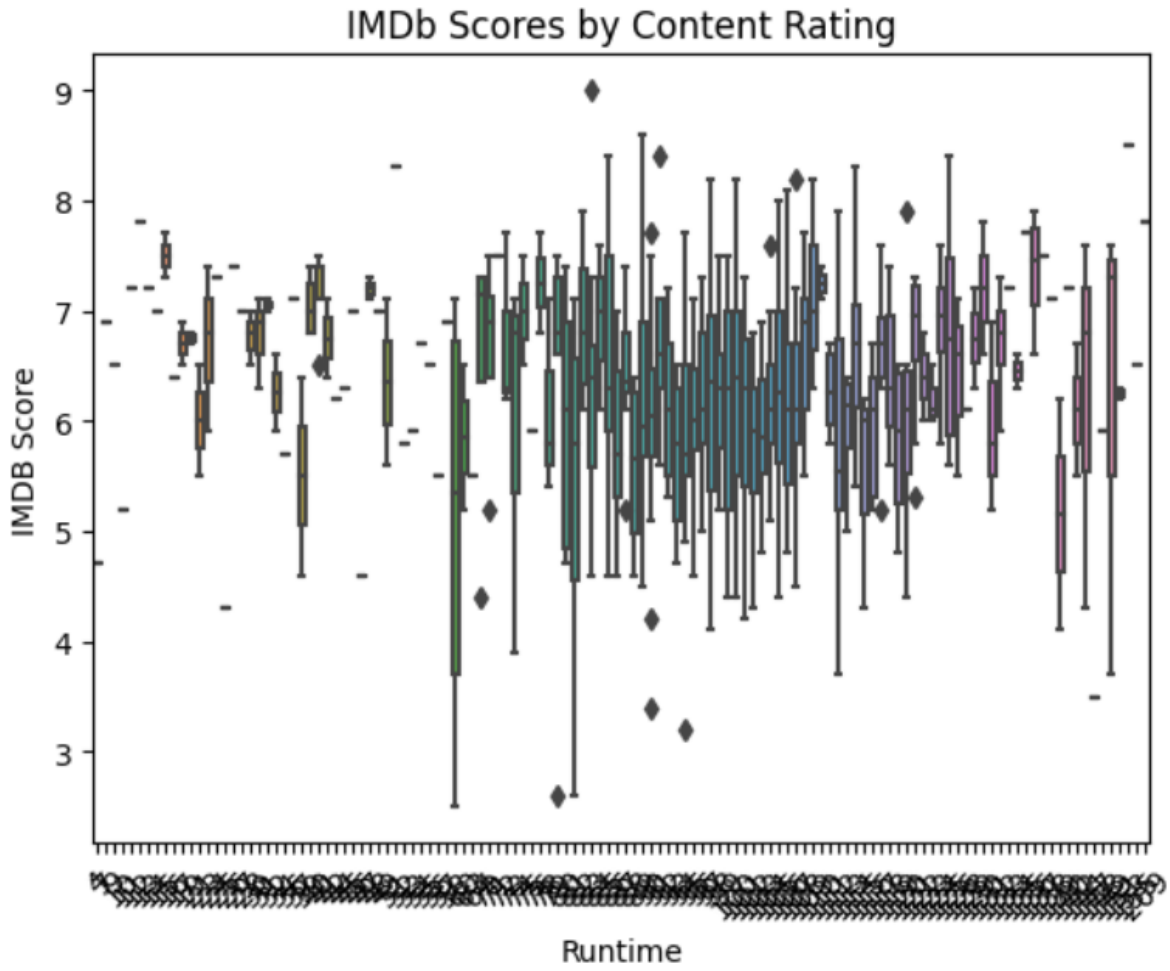
Data visualization with Matplotlib and Seaborn involves creating clear and appealing charts and plots to represent data, facilitating insights and communication. Matplotlib offers extensive customization, while Seaborn simplifies complex visualizations with high-level functions.

```
plt.hist(movie['IMDB Score'], bins=20)
plt.xlabel('IMDb Score')
plt.ylabel('Frequency')
plt.title('Distribution of IMDb Scores')
plt.show()

# Box plot to visualize IMDb scores by content rating
sns.boxplot(x='Runtime', y='IMDB Score', data=movie)
plt.xticks(rotation=45)
plt.title('IMDb Scores by Content Rating')
plt.show()
```

Output:





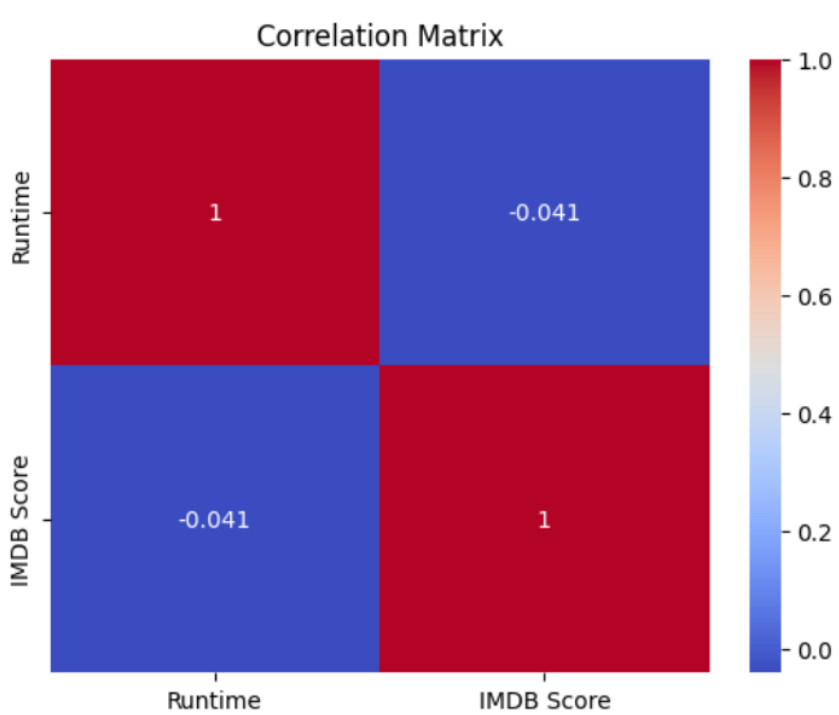
Correlation Analysis:

Correlation analysis assesses the strength and direction of a linear relationship between variables, helping to determine how closely they are related and whether they move together or in opposite directions in a dataset. It is often represented by a correlation coefficient, such as the Pearson correlation coefficient, which quantifies this relationship.

```
correlation_matrix = movie.corr()
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm')
plt.title('Correlation Matrix')
plt.show()

# Calculate the correlation between two specific columns
correlation = movie['Runtime'].corr(movie['IMDB Score'])
print(f"Correlation between 'Runtime' and 'IMDB Score': {correlation}")
```

<u>Output:</u>




```
Correlation between 'Runtime' and 'IMDB Score': -0.04089629142078858
```

Categorical Data Analysis:

Categorical data analysis involves the statistical examination of non-numeric data, such as categories, labels, or groupings, to understand patterns, relationships, and make inferences from qualitative information. Techniques include chi-squared tests, contingency tables, and logistic regression for modeling categorical outcomes.

```
[ ] # Count the number of movies in each genre
    genre_counts = movie['Genre'].str.split('|', expand=True).stack().value_counts()
    print(genre_counts)

    # Visualize the top 10 most common genres
    top_genres = genre_counts.head(10)
    top_genres.plot(kind='bar', rot=45)
    plt.title('Top 10 Movie Genres')
    plt.xlabel('Genre')
    plt.ylabel('Count')
    plt.show()
```

Output:

```
Documentary                            159
Drama                                   77
Comedy                                  49
Romantic comedy                         39
Thriller                                33
                                       ...
Romantic comedy-drama                    1
Heist film/Thriller                      1
Musical/Western/Fantasy                  1
Horror anthology                         1
Animation/Christmas/Comedy/Adventure     1
Length: 115, dtype: int64
```