IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Exploratory Data Analysis And Visualization: Netflix Data Using Python Libraries

1Deepti Sharma, 2Archana B. Saxena

1Professor, 2Professor

1JIMS Sec-5 Rohini,

2JIMS Sec-5 Rohini

Abstract

The phrase "Exploratory Data Analysis" combines two distinct methods that are found in the broad field of data science. In our case, that data is Netflix Data, an Open-Source Data Set that was acquired from Kaggle. After combining two additional sets, Geographic Latitudes & Longitudes and Netflix Title Critics/Reviews Data Set, the data was wrangled and exercised to derive maximum insights using EDA – Exploratory Data Analysis. The Python Library of Versatile Packages contains a variety of utility analytical tools that are used in the project. This work presents the methodical and perceptive application of techniques for Exploratory Data Analysis through the use of the relevant packages.

Keywords: —Exploratory Data Analysis; Data Analytics; Python;Matplotlib;Pandas; Seaborn; Numpy; Tensorflow – Keras

1. Introduction

We can define data visualisation as a graphical representation of information and data. Data cleaning, exploratory data analysis, and appropriate, effective communication with corporate stakeholders are its key uses. The need for data scientists is currently growing. We are moving closer and closer to a data-driven society every day. Making decisions based on data and using visualisation to communicate stories about what, when, and how data might help us reach a desirable result are both very advantageous.

Colours and patterns catch our attention. We can distinguish swiftly between blue and yellow, and a circle from a square. A sort of visual art known as data visualisation not only captures our attention but also holds it there. With the aid of data visualisation, we can literally tell the stakeholders our whole body of numerical information through eye-catching graphs.

We are currently in "an age of big data," in which trillions of rows of data are produced every day. Data visualisation aids in both showcasing a certain aspect of the data and curating it into an easily understandable format.

2. Data Visualization Tools

1. Bar Plot :

It is a graph with rectangular bars whose length and height are proportionate to the values they stand for and which reflects a certain category of data. It has various parameters like: Y coordinates for Y-Bar, Width of the bar and height of the bar.

2. Timeline :

It consists of different parameters like: Y coordinate for horizontal line, xmin: the range for this option should be 0 to 1. With 0 serving as the default value, the extremes of the plot are denoted by 0 and 1, respectively and xmax: the range for this option should be 0 to 1. With 1 being the default value, 0 denotes the plot's extreme left and 1 denotes its extreme right.

3. About the Data Set

One of the most well-known media and video streaming services is Netflix. They offer more than 8000 films and TV episodes on their platform, and as of the middle of 2021, they had more over 200 million subscribers worldwide. This tabular dataset includes listings for all of the Netflix films and TV episodes, together with information about the actors, directors, ratings, release year, duration, and other factors.

4. About Data Visualization Libraries in Python

1. **Matplotlib** : It is a plotting library for the Python programming language and includes the Numpy extension for numerical mathematics. When utilising general-purpose GUI toolkits like Tkinter, QT, WxPython, or GTX, it offers an object-oriented API for embedding plots.
2. **Seaborn**: It is a fantastic visualisation library for Python-based statistical graphics plotting. It offers lovely default styles and colour schemes to enhance the appeal of statistics charts. It is constructed on top of the Matplotlib toolkit and is tightly integrated with the Pandas data structures. Seaborn aims to make visualisation the core of data exploration and comprehension. In order to better comprehend the dataset, it also offers dataset-oriented APIs that allow us to move between various visual representations of the same variables.
3. **Numpy:** It is a Python module that supports multi-dimensional arrays and matrices and offers a wide range of complex mathematical operations that may be applied to them.
4. **Pandas:** It is a robust, adaptable, and simple tool for data manipulation in the Python programming language.

5. Data Visualization:

Data visualization is divided into following steps:

1. Importing Libraries
2. Loading the dataset
3. Data Cleaning
4. Data Visualization
5. Creating Word Cloud

1. Importing Libraries:

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

2. Loading the Data Set:

```
netflix=pd.read_csv('../input/netflix-shows/netflix_titles.csv')
```

A. Display first 10 values

```
netflix.head(10)
```

	show_id	type	title	director	cast	country	date_added	release_year
0	81145628	Movie	Norm of the North: King Sized Adventure	Richard Finn, Tim Maltby	Alan Marriott, Andrew Toth, Brian Dobson, Cole...	United States, India, South Korea, China	September 9, 2019	2019
1	80117401	Movie	Jandino: Whatever it Takes	NaN	Jandino Asporaat	United Kingdom	September 9, 2016	2016
2	70234439	TV Show	Transformers Prime	NaN	Peter Cullen, Sumalee Montano, Frank Welker, J...	United States	September 8, 2018	2013

B. Check for Null Values:

```
netflix.isnull().sum()
```

C. Check Unique Values:

```
netflix.nunique()
```

3. Data Cleaning

A. Calculating the Missing Data

```
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 missing percentage: 30.68%
cast missing percentage: 9.22%
country missing percentage: 6.51%
date_added missing percentage: 0.13%
rating missing percentage: 0.09%
```

B. Dealing with the Missing Data

```
df['country'] = df['country'].fillna(df['country'].mode()[0])
df['cast'].replace(np.nan,'No data',inplace=True)
df['director'].replace(np.nan,'No data',inplace=True)
df.dropna(inplace=True)
df.drop_duplicates(inplace=True)
```

4. Data Visualization

A. Netflix's Brand

```
sns.palplot(['#221f1f', '#b20710', '#e50914','#f5f5f1'])
plt.title("Netflix brand palette",loc='left',fontfamily='serif',fontsize=15,y=1.2)
plt.show()
```

Fig 1: Netflix Brand Palette

B. Visualising the Ratio between Netflix's TV Shows and Movies

```
x = df.groupby(['type'])['type'].count()
y = len(df)
r=((x/y)).round(2)
mf_ratio = pd.DataFrame(r).T
```

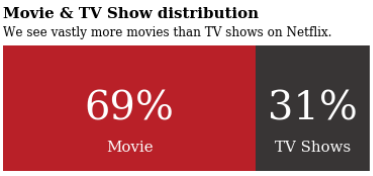


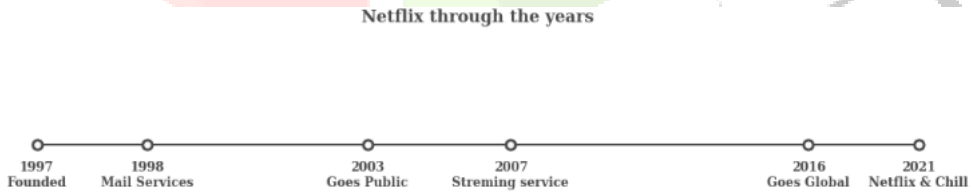
Fig 2: Movie and TV Show Distribution

C. Visualising Netflix's Timeline

a. Initializing the timeline list:

```
from datetime import datetime
tl_dates = [
    "1997nFounded",
    "1998nMail Services",
    "2003nGoes Public",
    "2007nStreming service",
    "2016nGoes Global",
    "2021nNetflix & Chill"
]
tl_x = [1,2,4,5.3,8,9]
```

D.Netflix timeline through the years:



E. Relation between Type and Rating

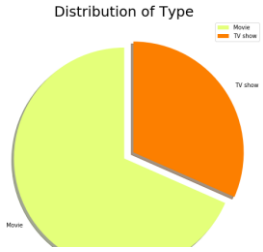


Fig 3: Distribution of Type and Rating

F. Histograms to Display Different Countries Using Netflix

```
df['country']
df['first_country'] = df['country'].apply(lambda x: x.split(",")[0])
df['first_country']
```

Calculate total occurrences of each Country:

```
first_country
USA          3379
India         956
UK            576
Canada        259
Japan         235
France        196
S. Korea      194
Spain         168
Mexico        123
Turkey        106
Name: count, dtype: int64
```

```
df['count']=1 #helper column
data = df.groupby('first_country')['count'].sum().sort_values(ascending=False)[:10]
```

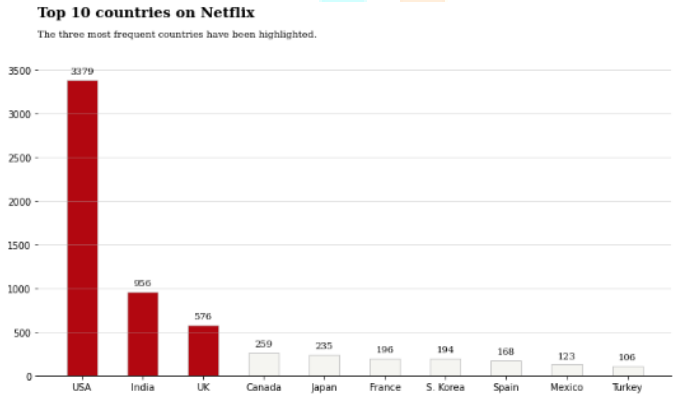


Fig 4: Top 10 Countries on Netflix

G. Creating Word Cloud

```
from wordcloud import WordCloud
```

```
plt.subplots(figsize=(25,15))
wordcloud = WordCloud(
                          background_color='white',
                          width=1920,
                          height=1080
                         ).generate(" ".join(df.country))
plt.imshow(wordcloud)
plt.axis('off')
plt.savefig('country.png')
plt.show()
```

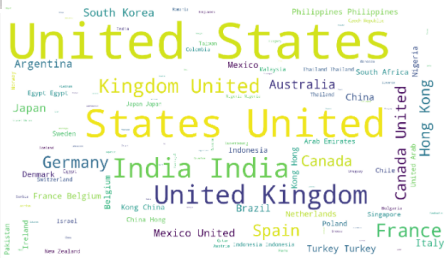


Fig 5: Word Cloud Representation

5. Results and Conclusion

In this work, authors explained the process of using Python for exploratory data analysis (EDA) using a Netflix data set. Many Python libraries were used including Pandas, Seaborn, Matplotlib, and Plotly, to visualise and analyse the data.

References:

[1] Kiranbala Nongthombam , Deepika Sharma, 2021, Data Analysis using Python, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 10, Issue 07 (July 2021)

[2] Jyoti Budhwar, Sukhdip Singh, 2021, Sentiment Analysis based Method for Amazon Product Reviews, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) ICACT – 2021 (Volume 09 – Issue 08)

[3] Soniya Grace, 2020, A Geospatial Analysis of Ground Water Quality Mapping using GIS in Sangareddy District, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 09, Issue 07 (July 2020)

[4] https://en.wikipedia.org/wiki/Data_analysis

[5] Sharma D., Saxena B. A., Aggarwal D. "Exploratory Sentiment Analysis of Sales Data", "European Economic Letters" (ABDC Journal, C Category), ISSN: 2323-5233, Vol 13 No. 4, October 2023, Page no: 982-986.

[6] Devesh Lowe, Bhavna Galhotra, Yukti Ahuja, "Unfurling the latest patterns of entertainment consumption by indian audience: a twitter sentiment analysis", International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-7 Issue-6C, April 2019.

[7] Devesh Lowe, Bhavna Galhotra, Yukti Ahuja, "Discovering Binge watching and Audience Engagement through Sentiment Analysis", International Journal of Advanced Science and Technology, ISSN 2005 4238, Vol. 29, No. 7, (2020), pp. 8030-8038.