IMDB Score Prediction

Step 1: Data cleaning and pre-processing

Data Cleaning:

<u>1.Replace the missing values:</u>

Missing values should be replaced in the data set in order to perform further calculations. Here we make use of python's "fillna()" method which is used to fill the null values. The python code used is given below:

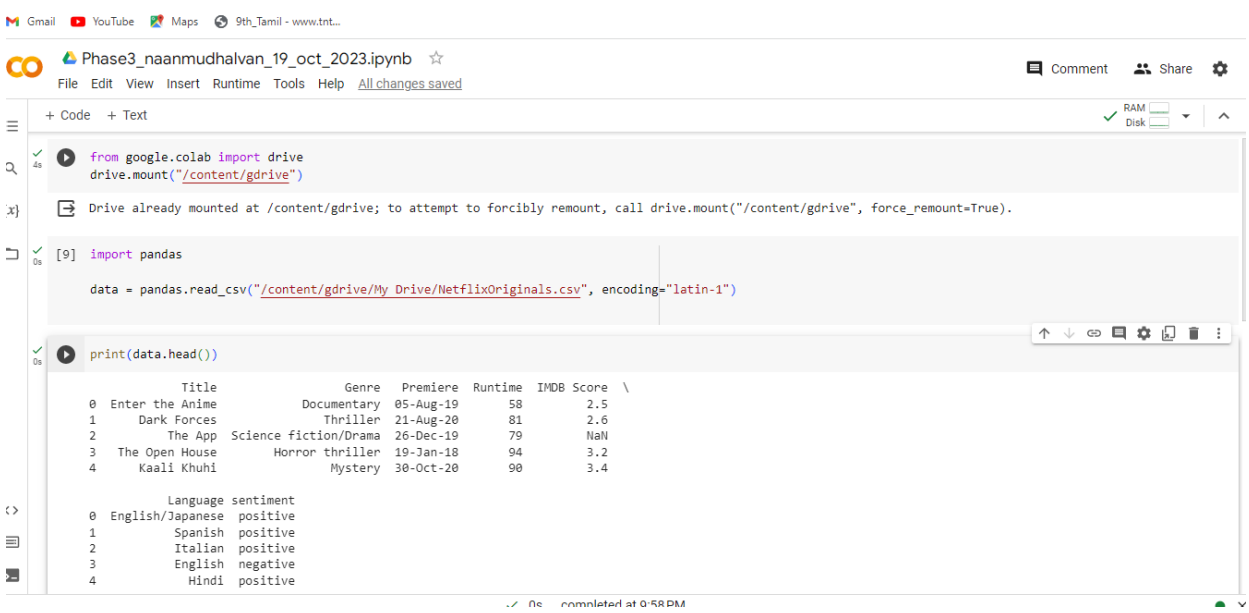
```
import pandas

data=pandas.read_csv("NetflixOriginals.csv")

data["IMDB Score"].fillna(5,inplace=True)
```

The execution of the code is shown below:

i) Before fillna()



ii) After fillna()

```python
data["IMDB Score"].fillna(5,inplace=True)
print(data.head())
```

```
              Title                  Genre   Premiere  Runtime  IMDB Score  \
0  Enter the Anime            Documentary  05-Aug-19       58         2.5
1      Dark Forces               Thriller  21-Aug-20       81         2.6
2          The App  Science fiction/Drama  26-Dec-19       79         5.0
3   The Open House        Horror thriller  19-Jan-18       94         3.2
4      Kaali Khuhi                Mystery  30-Oct-20       90         3.4

           Language sentiment
0  English/Japanese  positive
1           Spanish  positive
2           Italian  positive
3           English  negative
4             Hindi  positive
```

2.Converting categorical data to numerical data:

The categorical data such as male/female, positive/negative should be converted into numerical values. For example, 1-for male, 2-for female.

We use label encoder to do this conversion. The python code for this is:

```python
from sklearn.preprocessing import LabelEncoder

label_encoder=LabelEncoder()
data =label_encoder.fit_transform(data["sentiment"])
print(data)
```

```python
from sklearn.preprocessing import LabelEncoder
label_encoder=LabelEncoder()
data =label_encoder.fit_transform(data["sentiment"])
print(data)
```

```
[1 1 1 0 1 1 1 0 0 1 0 0 0 0 1 0 1 0 1 0 1 0 1 0 0 1 1 0 0 1 1 1 0 1 0 0 0
 0 1 0 0 1 0 0 1 1 0 0 1 0 1 1 1 1 0 0 0 0 1 1 0 0 1 0 0 1 0 0 0 0 0 0 1 1
 0 1 1 0 0 1 1 0 0 0 0 0 0 0 0 0 1 0 1 1 0 1 0 0 0 1 1 0 1 1 0 1 1 0 1 1 0
 0 0 1 1 1 1 0 0 0 1 1 0 0 1 0 0 0 1 1 1 1 0 0 0 0 1 0 1 0 0 0 0 1 0 1 1 1
 0 0 1 0 0 1 0 0 0 1 0 1 1 0 0 0 1 0 1 0 0 0 0 0 1 1 0 0 1 0 1 0 1 0 0 0 0
 0 1 0 1 0 1 1 1 1 0 0 0 0 1 0 0 1 1 1 0 1 0 0 1 1 0 0 0 1 0 1 1 0 1 0 0 1
 1 0 1 0 1 1 1 1 0 1 1 1 1 1 1 0 1 0 0 1 0 1 0 1 0 0 1 0 0 0 0 0 1 0 0 0 1
 1 1 0 0 0 0 1 0 0 0 1 0 1 1 1 0 1 0 1 0 1 1 0 0 0 1 0 1 1 1 1 1 1 1 1 1 0
 0 1 1 1 0 0 1 1 1 1 1 0 0 0 1 0 0 0 0 0 1 1 1 0 1 1 0 1 0 1 1 0 1 1 0 1 1
 1 1 0 1 1 0 0 0 1 0 1 0 0 0 1 0 0 1 0 0 1 0 0 1 0 0 0 1 0 1 1 1 0 0 0 1 1
 1 1 0 1 1 0 1 0 1 0 0 1 1 0 0 0 0 1 1 1 1 0 1 1 1 0 0 1 0 1 1 0 0 1 1 0 1
 1 0 1 0 1 0 0 0 0 0 1 0 0 0 0 0 1 0 1 1 1 0 1 0 0 0 1 0 1 1 1 1 1 0 1 0 0
 0 0 0 1 0 0 1 1 0 1 1 1 0 1 0 0 1 1 0 0 0 1 1 1 0 1 0 0 0 0 1 1 1 0 0 1 0
 1 0 1 1 0 0 1 1 0 0 1 1 1 1 0 0 0 0 1 0 1 1 0 1 1 0 1 0 1 0 0 0 0 0 1 0 1
 1 0 0 1 1 0 0 0 1 0 1 0 0 1 1 1 1 0 0 0 0 1 1 0 0 0 0 0 0 1 0 1 0 0 1 0 0
 0 1 1 1 0 0 0 0 0 1 0 1 1 1 0 1 1 1 0 0 0 0 0 0 1 1 0 1 0]
```

```python
file_data=pandas.get_dummies(data,columns=["sentiment"])
print(file_data)
```

```
file_data=pandas.get_dummies(data,columns=["sentiment"])
print(file_data)
```

```
     0  1
0    0  1
1    0  1
2    0  1
3    1  0
4    0  1
..  .. ..
579  0  1
580  0  1
581  1  0
582  0  1
583  1  0

[584 rows x 2 columns]
```

Otherwise, we can simply make use of replace() method:

```
data.replace({"positive":1,"negative":0},inplace=True)
print(data.head())
```

```
data.replace({"positive":1,"negative":0},inplace=True)
print(data.head())
```

```
                Title                  Genre   Premiere  Runtime  IMDB Score  \
0  Enter the Anime            Documentary  05-Aug-19       58         2.5
1      Dark Forces               Thriller  21-Aug-20       81         2.6
2          The App  Science fiction/Drama  26-Dec-19       79         5.0
3   The Open House        Horror thriller  19-Jan-18       94         3.2
4      Kaali Khuhi                Mystery  30-Oct-20       90         3.4

           Language  sentiment
0  English/Japanese          1
1           Spanish          1
2           Italian          1
3           English          0
4             Hindi          1
```

3. Removal of outliers:

Outliers are the values that does not match the value range in the dataset. For example, if A=[1,3,4,2,6,8,7,100], then 100 is the outlier since it is a out of range value in 'A'.

The removal of outliers from IMDB data st is very important because it may affect our prediction value.

The following python code removes the outliers from our dataset.

```
import numpy as np
Q1=data['IMDB Score'].quantile(0.25)
Q3=data['IMDB Score'].quantile(0.75)
IQR=Q3-Q1
lower_bound=Q1-1.5*IQR
upper_bound=Q3+1.5*IQR
outliers=data[(data['IMDB Score']<lower_bound)|(data['IMDB Score']>upper_bound)]
print("Outliers in IMDB Scores:",outliers)
```

+ Code + Text

```
[42] Q1=data['IMDB Score'].quantile(0.25)
     Q3=data['IMDB Score'].quantile(0.75)
     IQR=Q3-Q1
```

```
lower_bound=Q1-1.5*IQR
upper_bound=Q3+1.5*IQR
outliers=data[(data['IMDB Score']<lower_bound)|(data['IMDB Score']>upper_bound)]
print("Outliers in IMDB Scores:",outliers)
```

```
Outliers in IMDB Scores:                                        Title                Genre   Premiere  \
0                              Enter the Anime          Documentary  05-Aug-19
1                                  Dark Forces             Thriller  21-Aug-20
3                               The Open House      Horror thriller  19-Jan-18
4                                  Kaali Khuhi              Mystery  30-Oct-20
5                                        Drive               Action  01-Nov-19
6                            Leyla Everlasting               Comedy  04-Dec-20
7              The Last Days of American Crime  Heist film/Thriller  05-Jun-20
583  David Attenborough: A Life on Our Planet          Documentary  04-Oct-20

     Runtime  IMDB Score          Language  sentiment
0         58         2.5  English/Japanese          1
1         81         2.6           Spanish          1
3         94         3.2           English          0
4         90         3.4             Hindi          1
5        147         3.5             Hindi          1
6        112         3.7           Turkish          1
7        149         3.7           English          0
```

```
data_cleaned=data[(data['IMDB Score']>=lower_bound)&(data['IMDB Score']<=upper_bound)]
print(data_cleaned)
```

```
+ Code  + Text
data_cleaned=data[(data['IMDB Score']>=lower_bound)&(data['IMDB Score']<=upper_bound)]
print(data_cleaned)
```

```
                                        Title                    Genre  \
2                                     The App    Science fiction/Drama
8                                     Paradox  Musical/Western/Fantasy
9                           Sardar Ka Grandson                   Comedy
10                        Searching for Sheela              Documentary
11                                   The Call                    Drama
..                                        ...                      ...
578   Ben Platt: Live from Radio City Music Hall             Concert Film
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

      Premiere  Runtime  IMDB Score                 Language  sentiment
2    26-Dec-19       79         5.0                  Italian          1
8    23-Mar-18       73         3.9                  English          0
9    18-May-21      139         4.1                    Hindi          1
10   22-Apr-21       58         4.1                  English          0
11   27-Nov-20      112         4.1                   Korean          0
..         ...      ...         ...                      ...        ...
578  20-May-20       85         8.4                  English          0
579  31-Dec-18      125         8.4                  English          1
580  09-Oct-15       91         8.4  English/Ukranian/Russian          1
581  16-Dec-18      153         5.0                  English          0
582  08-Dec-20       89         8.6               Portuguese          1
```

✓ 0s completed at 11:10 PM

Thus, the cleaning of dataset is done successfully.

Data preprocessing:

In order to perform IMDB score prediction, we need to split the data into training and testing. The following code is used to split the data:

```
from sklearn.model_selection import train_test_split
x = data_cleaned[['Runtime', 'sentiment']]
y = data_cleaned['IMDB Score']
X_train, X_test, y_train, y_test = train_test_split(x, y, test_size=0.2,
random_state=42)
```

```
[576 rows x 7 columns]
```

```
[11] from sklearn.model_selection import train_test_split
     x = data_cleaned[['Runtime', 'sentiment']]
     y = data_cleaned['IMDB Score']
     X_train, X_test, y_train, y_test = train_test_split(x, y, test_size=0.2, random_state=42)
```

Thus, data cleaning and data pre-processing is done successfully.

Done by,

Janani A, Shree Varshenee K