Movie Rating Analysis using Python

We all watch movies for entertainment, some of us never rate it, while some viewers always rate every movie they watch. This type of viewer helps in rating movies for people who go through the movie reviews before watching any movie to make sure they are about to watch a good movie.I will walk you through the task of Movie Rating Analysis using Python.

Analyzing the rating given by viewers of a movie helps many people decide whether or not to watch that movie. So, for the Movie Rating Analysis task, you first need to have a dataset that contains data about the ratings given by each viewer. For this task, I have collected a dataset from Kaggle that contains two files:

1. one file contains the data about the movie Id, title and the genre of the movie
2. and the other file contains the user id, movie id, ratings given by the user and the timestamp of the ratings

Now let’s get started with the task of movie rating analysis by importing the necessary Python libraries and the datasets:

In [1]:

```
import numpy as np
import pandas as pd
import warnings
warnings.filterwarnings("ignore")
movies = pd.read_csv("/Users/gulladhanush/Downloads/movies.dat",delimiter ="::")
print(movies.head())
```

```
   0000008      Edison Kinetoscopic Record of a Sneeze (1894)  \
0       10                La sortie des usines Lumière (1895)
1       12                      The Arrival of a Train (1896)
2       25  The Oxford and Cambridge University Boat Race ...
3       91                         Le manoir du diable (1896)
4      131                           Une nuit terrible (1896)

     Documentary|Short
0    Documentary|Short
1    Documentary|Short
2                  NaN
3         Short|Horror
4  Short|Comedy|Horror
```

In the above code, I have only imported the movies dataset that does not have any column names, so let’s define the column names:

In [2]:

```
movies.columns = ["ID", "Title", "Genre"]
print(movies.head())
```

```
    ID                                              Title                Genre
0   10                La sortie des usines Lumière (1895)    Documentary|Short
1   12                      The Arrival of a Train (1896)    Documentary|Short
2   25  The Oxford and Cambridge University Boat Race ...                  NaN
3   91                         Le manoir du diable (1896)         Short|Horror
4  131                           Une nuit terrible (1896)  Short|Comedy|Horror
```

Now let’s import the ratings dataset:

In [3]:

```
ratings = pd.read_csv("/Users/gulladhanush/Downloads/ratings.dat", delimiter='::')
print(ratings.head())
```

```
   1  0114508  8  1381006850
0  2   499549  9  1376753198
1  2  1305591  8  1376742507
2  2  1428538  1  1371307089
3  3    75314  1  1595468524
4  3   102926  9  1590148016
```

The rating dataset also doesn’t have any column names, so let’s define the column names of this data also:

In [4]:

```
ratings.columns = ["User", "ID", "Ratings", "Timestamp"]
print(ratings.head())
```

```
   User       ID  Ratings   Timestamp
0     2   499549        9  1376753198
1     2  1305591        8  1376742507
2     2  1428538        1  1371307089
3     3    75314        1  1595468524
4     3   102926        9  1590148016
```

Now I am going to merge these two datasets into one, these two datasets have a common column as ID, which contains movie ID, so we can use this column as the common column to merge the two datasets:

In [5]:

```
data = pd.merge(movies, ratings, on=["ID", "ID"])
print(data.head())
```

```
   ID                                              Title              Genre  \
0  10                La sortie des usines Lumière (1895)  Documentary|Short
1  12                      The Arrival of a Train (1896)  Documentary|Short
2  25  The Oxford and Cambridge University Boat Race ...                NaN
3  91                         Le manoir du diable (1896)       Short|Horror
4  91                         Le manoir du diable (1896)       Short|Horror

    User  Ratings   Timestamp
0  70577       10  1412878553
1  69535       10  1439248579
2  37628        8  1488189899
3   5814        6  1385233195
4  37239        5  1532347349
```

have a look at the distribution of the ratings of all the movies given by the viewers:

In [6]:

```
ratings = data["Ratings"].value_counts()
ratings
```

Out[6]:

```
8     219311
7     203476
9     128749
6     118323
10    107284
5      68458
4      27779
3      15258
1      10663
2       9053
0        278
Name: Ratings, dtype: int64
```

In [7]:

```
numbers = ratings.index
numbers
```

Out[7]:

```
Int64Index([8, 7, 9, 6, 10, 5, 4, 3, 1, 2, 0], dtype='int64')
```

In [8]:

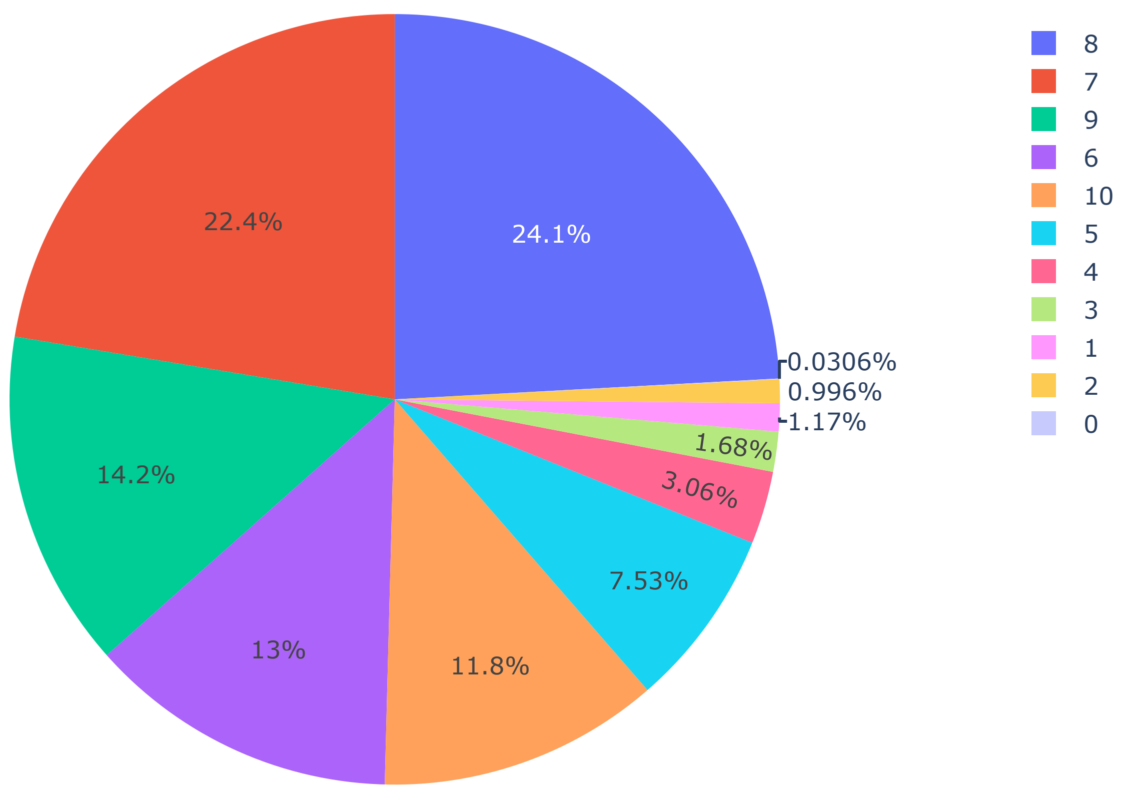
```
quantity = ratings.values
quantity
```

Out[8]:

```
array([219311, 203476, 128749, 118323, 107284,  68458,  27779,  15258,
        10663,   9053,    278])
```

In [9]:

```
ratings = data["Ratings"].value_counts()
numbers = ratings.index
quantity = ratings.values
import plotly.express as px
fig = px.pie(data, values=quantity, names=numbers)
fig.show()
```



So, according to the pie chart above, most movies are rated 8 by users. From the above figure, it can be said that most of the movies are rated positively.

As 10 is the highest rating a viewer can give, let’s take a look at the top 10 movies that got 10 ratings by viewers:

In [10]:

```
data2 = data.query("Ratings == 10")
print(data2["Title"].value_counts().head(10))
```

```
Joker (2019)                          1479
Interstellar (2014)                   1386
1917 (2019)                            820
Avengers: Endgame (2019)               812
The Shawshank Redemption (1994)        707
Gravity (2013)                         653
The Wolf of Wall Street (2013)         581
Hacksaw Ridge (2016)                   570
Avengers: Infinity War (2018)          535
La La Land (2016)                      510
Name: Title, dtype: int64
```

So, according to this dataset, Joker (2019) got the highest number of 10 ratings from viewers. This is how you can analyze movie ratings using Python as a data science beginner.