2gaoomtiq

January 3, 2023

1 DATA SCIENCE WITH PYTHON : Movielens Case Study

Background of Problem Statement :

The GroupLens Research Project is a research group in the Department of Computer Science and Engineering at the University of Minnesota. Members of the GroupLens Research Project are involved in many research projects related to the fields of information filtering, collaborative filtering, and recommender systems. The project is led by professors John Riedl and Joseph Konstan. The project began to explore automated collaborative filtering in 1992 but is most well known for its worldwide trial of an automated collaborative filtering system for Usenet news in 1996. Since then the project has expanded its scope to research overall information by filtering solutions, integrating into content-based methods, as well as, improving current collaborative filtering technology.

Problem Objective :

Using the Exploratory Data Analysis technique to find out features affecting the ratings of any particular movie and to build a model to predict the movie ratings.

[1]:

```
#importing pandas dataframe
import pandas as pd

import warnings
warnings.filterwarnings('ignore')

#importing seaborn
import seaborn as sns

#importing pandas profiling
import pandas_profiling as pf

#importing matplolib
import matplotlib
import matplotlib.pyplot as plt
%matplotlib inline

#importing reg-ex
import re

#Hold out method for splitting data
```

```
from sklearn.model_selection import train_test_split

#importing accuracy_score
from sklearn.metrics import accuracy_score

#importing LGBMClassifier
from lightgbm import LGBMClassifier

#importing xgboost
import xgboost
```

1.0.1 Importing the three datasets

[2]:

```
rating = ['UserID','MovieID','Rating','Timestamp']
user = ['UserID','Gender','Age','Occupation','Zip-code']
movie = ['MovieID','Title','Genres']
```

[3]:

```
rating_df = pd.read_csv('ratings.dat',header=None,delimiter='::',names=rating)
print(rating_df.head())
print()
print(rating_df.shape)
```

```
   UserID  MovieID  Rating  Timestamp
0       1     1193       5  978300760
1       1      661       3  978302109
2       1      914       3  978301968
3       1     3408       4  978300275
4       1     2355       5  978824291

(1000209, 4)
```

[4]:

```
user_df = pd.read_csv('users.dat',header=None,delimiter='::',names=user)
print(user_df.head())
print()
print(user_df.shape)
```

```
   UserID Gender  Age  Occupation Zip-code
0       1      F    1          10    48067
1       2      M   56          16    70072
2       3      M   25          15    55117
3       4      M   45           7    02460
4       5      M   25          20    55455

(6040, 5)
```

[5]:

```
movie_df = pd.read_csv('movies.dat',header=None,delimiter='::',names=movie)
print(movie_df.head())
print()
print(movie_df.shape)
```

```
   MovieID                               Title                        Genres
0        1                    Toy Story (1995)   Animation|Children's|Comedy
1        2                      Jumanji (1995)  Adventure|Children's|Fantasy
2        3             Grumpier Old Men (1995)                Comedy|Romance
3        4            Waiting to Exhale (1995)                  Comedy|Drama
4        5  Father of the Bride Part II (1995)                        Comedy

(3883, 3)
```

[6]:
```
movie_df = pd.read_csv('movies.dat',header=None,delimiter='::',names=movie)
print(movie_df.head())
print()
print(movie_df.shape)
```

```
   MovieID                               Title                        Genres
0        1                    Toy Story (1995)   Animation|Children's|Comedy
1        2                      Jumanji (1995)  Adventure|Children's|Fantasy
2        3             Grumpier Old Men (1995)                Comedy|Romance
3        4            Waiting to Exhale (1995)                  Comedy|Drama
4        5  Father of the Bride Part II (1995)                        Comedy

(3883, 3)
```

1.0.2 Merging the three datasets

[7]:
```
df = rating_df.merge(user_df,how='outer',on='UserID')
df = df.merge(movie_df,how='outer',on='MovieID')
df.head()
```

[7]:
```
   UserID  MovieID  Rating    Timestamp Gender   Age  Occupation Zip-code  \
0     1.0     1193     5.0  978300760.0      F   1.0        10.0    48067
1     2.0     1193     5.0  978298413.0      M  56.0        16.0    70072
2    12.0     1193     4.0  978220179.0      M  25.0        12.0    32793
3    15.0     1193     4.0  978199279.0      M  25.0         7.0    22903
4    17.0     1193     5.0  978158471.0      M  50.0         1.0    95350

                                    Title Genres
0  One Flew Over the Cuckoo's Nest (1975)  Drama
1  One Flew Over the Cuckoo's Nest (1975)  Drama
2  One Flew Over the Cuckoo's Nest (1975)  Drama
3  One Flew Over the Cuckoo's Nest (1975)  Drama
4  One Flew Over the Cuckoo's Nest (1975)  Drama
```

[8]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1000386 entries, 0 to 1000385
Data columns (total 10 columns):
UserID        1000209 non-null float64
MovieID       1000386 non-null int64
Rating        1000209 non-null float64
Timestamp     1000209 non-null float64
Gender        1000209 non-null object
Age           1000209 non-null float64
Occupation    1000209 non-null float64
Zip-code      1000209 non-null object
Title         1000386 non-null object
Genres        1000386 non-null object
dtypes: float64(5), int64(1), object(4)
memory usage: 84.0+ MB
```

[9]:

```
df.shape
```

[9]:

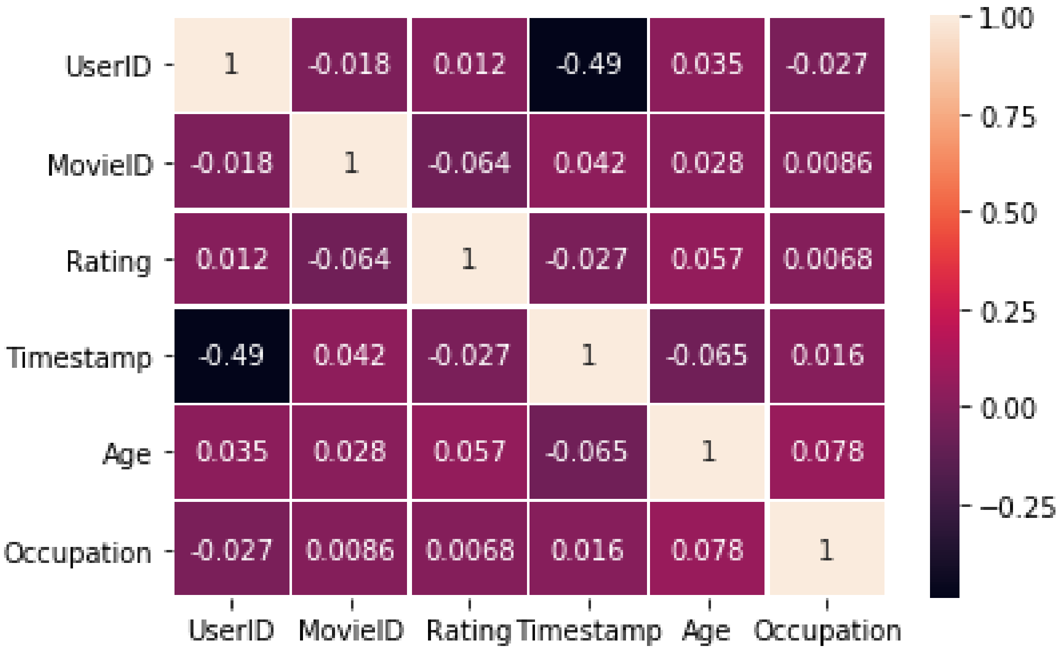
```
(1000386, 10)
```

[10]:

```
corr = df.corr()
sns.heatmap(corr,annot= True,linewidths=0.5)
```

[10]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x2917c570470>
```



1.0.3 Extracting the pandas profiling report

[11]:

```
pf.describe(df)
pfr = pf.ProfileReport(df)
pfr.to_file('Movielens_pfr.html')
```

[12]:

```
print('Na values in the data frame is :')
def is_na(x):
    for i in x.columns:
        print(i,'column',' :',x[i].isna().sum(),'\n')
is_na(df)
```

```
Na values in the data frame is :
UserID column  : 177

MovieID column  : 0

Rating column  : 177

Timestamp column  : 177

Gender column  : 177

Age column  : 177

Occupation column  : 177

Zip-code column  : 177

Title column  : 0

Genres column  : 0
```

[13]:

```
df.dropna(inplace=True)
```

[14]:

```
df.Rating.isna().value_counts()
```

[14]:

```
False    1000209
Name: Rating, dtype: int64
```

[15]:

```
def df_unique(X):
    for i in X.columns:
        print('Column : ',i,'\n',X[i].unique(), '\n Total unique values is: ',␣
 ↪X[i].nunique())
       ␣
 ↪print('----------------------------------------------------------------------')
```

```
df_unique(df)
```

```
Column :  UserID
 [1.000e+00 2.000e+00 1.200e+01 … 2.982e+03 3.893e+03 4.211e+03]
 Total unique values is:  6040
----------------------------------------------------------------------
Column :  MovieID
 [1193  661  914 … 2845 3607 2909]
 Total unique values is:  3706
----------------------------------------------------------------------
Column :  Rating
 [5. 4. 3. 2. 1.]
 Total unique values is:  5
----------------------------------------------------------------------
Column :  Timestamp
 [9.78300760e+08 9.78298413e+08 9.78220179e+08 … 9.58846401e+08
 9.76029116e+08 9.57273353e+08]
 Total unique values is:  458455
----------------------------------------------------------------------
Column :  Gender
 ['F' 'M']
 Total unique values is:  2
----------------------------------------------------------------------
Column :  Age
 [ 1. 56. 25. 50. 18. 45. 35.]
 Total unique values is:  7
----------------------------------------------------------------------
Column :  Occupation
 [10. 16. 12.  7.  1.  3.  4.  8. 17.  0.  2.  9. 19. 18. 15. 11. 20. 13.
  5. 14.  6.]
 Total unique values is:  21
----------------------------------------------------------------------
Column :  Zip-code
 ['48067' '70072' '32793' … '74403' '79401' '77662']
 Total unique values is:  3439
----------------------------------------------------------------------
Column :  Title
 ["One Flew Over the Cuckoo's Nest (1975)"
 'James and the Giant Peach (1996)' 'My Fair Lady (1964)' …
 'White Boys (1999)' 'One Little Indian (1973)'
 'Five Wives, Three Secretaries and Me (1998)']
 Total unique values is:  3706
----------------------------------------------------------------------
Column :  Genres
 ['Drama' "Animation|Children's|Musical" 'Musical|Romance'
 "Animation|Children's|Comedy" 'Action|Adventure|Comedy|Romance'
 'Action|Adventure|Drama' 'Comedy|Drama'
```

```
"Adventure|Children's|Drama|Musical" 'Musical' 'Comedy'
"Animation|Children's" 'Comedy|Fantasy' 'Animation' 'Comedy|Sci-Fi'
'Drama|War' 'Romance' "Animation|Children's|Musical|Romance"
"Children's|Drama|Fantasy|Sci-Fi" 'Drama|Romance'
'Animation|Comedy|Thriller'
"Adventure|Animation|Children's|Comedy|Musical"
"Animation|Children's|Comedy|Musical" 'Thriller' 'Action|Crime|Romance'
'Action|Adventure|Fantasy|Sci-Fi' "Children's|Comedy|Musical"
'Action|Drama|War' "Children's|Drama" 'Crime|Drama|Thriller'
'Action|Crime|Drama' 'Action|Adventure|Mystery' 'Crime|Drama'
'Action|Adventure|Sci-Fi|Thriller' 'Action|Adventure|Romance|Sci-Fi|War'
'Action|Thriller' 'Action|Drama' 'Comedy|Drama|Western'
'Action|Adventure|Crime' 'Action|Crime|Mystery|Thriller'
'Comedy|Drama|Romance' 'Comedy|Drama|War' 'Drama|Sci-Fi'
'Action|Drama|Thriller' 'Action|Comedy|Western' 'Adventure|Comedy|Drama'
'Drama|Thriller' 'Comedy|Romance' 'Action|Drama|Romance|Thriller'
'Action|Crime|Thriller' 'Action|Sci-Fi|Thriller' 'Action|Horror|Sci-Fi'
'Action|Sci-Fi' 'Action|Romance|War' 'Adventure|Drama|Romance|Sci-Fi'
'Action|Adventure|Sci-Fi' 'Drama|Romance|War' 'Action|Drama|Romance'
'Crime|Drama|Film-Noir|Thriller' 'Adventure|Drama|Western'
'Action|Adventure|Drama|Sci-Fi|War' 'Action|Adventure|Thriller'
'Action|Adventure|Romance|Thriller' 'Action|Adventure' 'Comedy|Horror'
'Action|Crime|Drama|Thriller' 'Action|Mystery|Romance|Thriller'
'Action|Romance|Thriller' 'Action|Comedy|Drama' 'Action'
'Action|Sci-Fi|War' 'Action|Comedy|Crime|Drama'
'Action|Adventure|Romance' 'Comedy|Romance|War' 'Comedy|Thriller'
'Action|Adventure|Comedy' 'Action|Comedy' 'Adventure|Thriller'
'Action|Adventure|Fantasy' 'Action|Adventure|Horror'
'Action|Adventure|Comedy|Sci-Fi' 'Action|Adventure|Comedy|Horror'
'Western' 'Adventure|Comedy' 'Adventure|Drama'
'Action|Adventure|Horror|Thriller' 'Comedy|Western'
"Animation|Children's|Comedy|Musical|Romance" 'Action|Western'
'Action|Horror|Sci-Fi|Thriller' 'Action|Horror'
'Adventure|Animation|Film-Noir' 'Drama|Romance|Thriller'
'Crime|Drama|Romance|Thriller' 'Crime|Thriller' 'Animation|Comedy'
'Documentary' 'Crime|Film-Noir|Mystery|Thriller' 'Drama|Horror'
'Mystery|Sci-Fi|Thriller' 'Drama|Mystery' 'Horror|Romance'
'Horror|Sci-Fi' 'Horror' 'Sci-Fi|Thriller' 'Crime' 'Action|Crime'
'Crime|Horror' 'Drama|Mystery|Thriller' 'Comedy|Crime'
'Drama|Sci-Fi|Thriller' "Children's|Comedy" 'Horror|Mystery|Thriller'
'Film-Noir|Mystery' 'Comedy|Crime|Mystery|Thriller' 'Drama|Musical'
'Adventure|Sci-Fi' "Children's|Comedy|Drama" 'Action|Romance'
"Adventure|Animation|Children's|Musical" 'Comedy|Musical'
"Children's|Fantasy|Musical" "Children's|Comedy|Western"
'Drama|Romance|War|Western' "Adventure|Children's|Comedy"
'Comedy|Fantasy|Romance' 'Comedy|Musical|Romance'
"Adventure|Children's|Drama" 'Action|Drama|Thriller|War'
'Drama|Thriller|War' 'Adventure|Animation|Sci-Fi|Thriller'
```

```
'Animation|Sci-Fi' 'Comedy|Crime|Drama|Mystery' 'Crime|Drama|Mystery'
'Action|Comedy|Sci-Fi|Thriller' 'Comedy|Crime|Fantasy'
'Horror|Sci-Fi|Thriller' "Adventure|Children's|Comedy|Fantasy|Sci-Fi"
'Film-Noir|Mystery|Thriller' 'Adventure' 'Comedy|War'
'Comedy|Romance|Thriller' "Action|Children's|Fantasy"
"Adventure|Children's|Fantasy" 'Action|Adventure|Comedy|Crime'
'Adventure|Musical' "Animation|Children's|Drama|Fantasy"
'Comedy|Mystery|Thriller' 'Action|Adventure|Crime|Drama'
"Children's|Fantasy|Sci-Fi" "Adventure|Children's" 'War'
'Comedy|Horror|Musical|Sci-Fi' "Children's|Comedy|Fantasy" 'Sci-Fi|War'
"Animation|Children's|Fantasy|Musical" "Children's|Sci-Fi"
"Adventure|Children's|Fantasy|Sci-Fi" 'Mystery|Thriller'
'Comedy|Horror|Musical' 'Action|Horror|Thriller' 'Adventure|Fantasy'
'Drama|Mystery|Sci-Fi|Thriller' 'Crime|Drama|Sci-Fi'
"Adventure|Children's|Musical" 'Action|Sci-Fi|Thriller|War'
'Adventure|War' 'Action|Adventure|Romance|War'
'Action|Drama|Fantasy|Romance' 'Adventure|Comedy|Sci-Fi'
'Comedy|Sci-Fi|Western' 'Action|Adventure|Comedy|Horror|Sci-Fi'
"Adventure|Children's|Comedy|Fantasy" 'Film-Noir|Sci-Fi' 'Drama|Fantasy'
"Children's|Drama|Fantasy" "Children's|Fantasy" 'Fantasy|Sci-Fi'
'Action|Comedy|Musical' 'Adventure|Fantasy|Sci-Fi'
'Action|Adventure|Sci-Fi|War' "Action|Adventure|Children's|Comedy"
"Adventure|Children's|Drama|Romance" "Adventure|Children's|Sci-Fi"
"Children's" 'Comedy|Drama|Musical' 'Comedy|Fantasy|Romance|Sci-Fi'
'Comedy|Crime|Drama' 'Sci-Fi' 'Adventure|Fantasy|Romance'
'Adventure|Romance' 'Adventure|Western' 'Action|Drama|Mystery'
'Adventure|Animation|Sci-Fi' 'Adventure|Romance|Sci-Fi' 'Horror|Thriller'
'Action|Adventure|Mystery|Sci-Fi' 'Adventure|Drama|Thriller'
'Comedy|Horror|Thriller' 'Action|Comedy|Crime|Horror|Thriller'
'Crime|Horror|Mystery|Thriller' 'Crime|Horror|Thriller'
'Crime|Drama|Mystery|Thriller' 'Animation|Musical'
'Action|Sci-Fi|Western' 'Crime|Drama|Film-Noir'
'Adventure|Sci-Fi|Thriller' 'Drama|Fantasy|Romance|Thriller'
'Mystery|Sci-Fi' 'Action|Crime|Sci-Fi' 'Comedy|Mystery'
'Action|Romance|Sci-Fi' 'Crime|Film-Noir|Mystery' 'Comedy|Drama|Sci-Fi'
'Sci-Fi|Thriller|War' 'Film-Noir|Thriller'
'Action|Adventure|Animation|Horror|Sci-Fi'
'Action|Sci-Fi|Thriller|Western' 'Comedy|Horror|Sci-Fi'
'Crime|Film-Noir|Thriller' 'Comedy|Crime|Thriller'
'Film-Noir|Sci-Fi|Thriller' "Adventure|Animation|Children's|Sci-Fi"
'Action|Adventure|Drama|Romance' "Children's|Musical"
'Action|Comedy|Musical|Sci-Fi' 'Action|Drama|Sci-Fi|Thriller'
'Action|Comedy|Fantasy' 'Action|War' 'Action|Comedy|Sci-Fi|War'
'Comedy|Crime|Horror' 'Action|Comedy|War'
"Action|Adventure|Children's|Sci-Fi" "Action|Children's"
'Comedy|Documentary' 'Action|Adventure|Animation'
'Action|Mystery|Thriller'
"Action|Animation|Children's|Sci-Fi|Thriller|War" 'Crime|Drama|Romance'
```

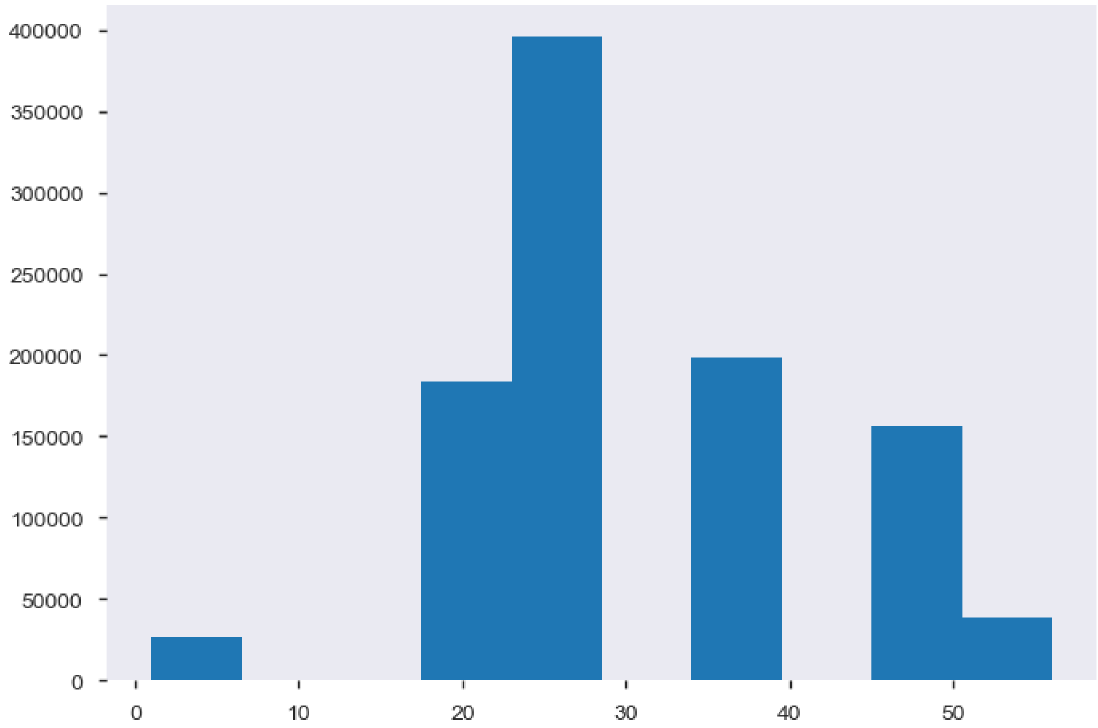
```
 'Crime|Film-Noir' 'Mystery|Romance|Thriller'
 'Comedy|Mystery|Romance|Thriller' 'Action|Adventure|Sci-Fi|Thriller|War'
 'Adventure|Crime|Sci-Fi|Thriller' 'Action|Adventure|Western'
 "Animation|Children's|Fantasy|War" 'Action|Adventure|Comedy|War'
 "Children's|Comedy|Sci-Fi"
 "Adventure|Animation|Children's|Comedy|Fantasy" 'Drama|Musical|War'
 'Drama|Mystery|Romance' 'Adventure|Drama|Romance' 'Film-Noir'
 'Film-Noir|Romance|Thriller' 'Drama|Film-Noir' 'Romance|Thriller'
 'Action|Adventure|War' 'Mystery' 'Action|Adventure|Drama|Thriller'
 'Musical|Romance|War' 'Drama|Western'
 'Action|Drama|Mystery|Romance|Thriller' 'Adventure|Comedy|Musical'
 'Documentary|Musical' 'Action|Thriller|War' 'Adventure|Comedy|Romance'
 "Adventure|Children's|Comedy|Fantasy|Romance" 'Romance|War'
 'Comedy|Romance|Sci-Fi' 'Action|Mystery|Sci-Fi|Thriller'
 "Children's|Horror" 'Adventure|Musical|Romance'
 "Adventure|Children's|Comedy|Musical" "Children's|Comedy|Mystery"
 'Action|Comedy|Romance|Thriller' 'Action|Drama|Western'
 "Animation|Children's|Comedy|Romance" 'Comedy|Mystery|Romance'
 'Action|Crime|Mystery' 'Comedy|Drama|Thriller' 'Musical|War'
 'Documentary|Drama' 'Action|Adventure|Crime|Thriller'
 "Action|Adventure|Children's" "Adventure|Children's|Romance"
 "Adventure|Animation|Children's"
 "Action|Adventure|Animation|Children's|Fantasy"
 "Adventure|Animation|Children's|Fantasy" 'Drama|Film-Noir|Thriller'
 'Crime|Mystery' 'Documentary|War' 'Action|Comedy|Crime'
 'Drama|Romance|Sci-Fi' 'Horror|Mystery' 'Drama|Horror|Thriller'
 "Action|Adventure|Children's|Fantasy" 'Animation|Mystery'
 'Drama|Romance|Western' 'Romance|Western' 'Comedy|Film-Noir|Thriller'
 'Fantasy' 'Film-Noir|Horror']
 Total unique values is:  301
----------------------------------------------------------------------
```

1.0.4 Exploring the datasets using visual representations

1.0.5 Visualizing the User Age Distribution

```
[16]: df.Age.hist(grid=False)
```

```
[16]: <matplotlib.axes._subplots.AxesSubplot at 0x291783d8898>
```



1.0.6 Visualizing User rating of the movie “Toy Story”

[17]:
```python
def fn(x):
    return re.search("Toy Story".lower(), x.lower())!=None
title = df.iloc[0].Title
title
```

[17]:
```
"One Flew Over the Cuckoo's Nest (1975)"
```

[18]:
```python
re_tit = df["Title"].apply(fn)
re_tit.head()
```

[18]:
```
0    False
1    False
2    False
3    False
4    False
Name: Title, dtype: bool
```

[19]:
```python
toystory = df[df["Title"].apply(fn)]
toystory
```

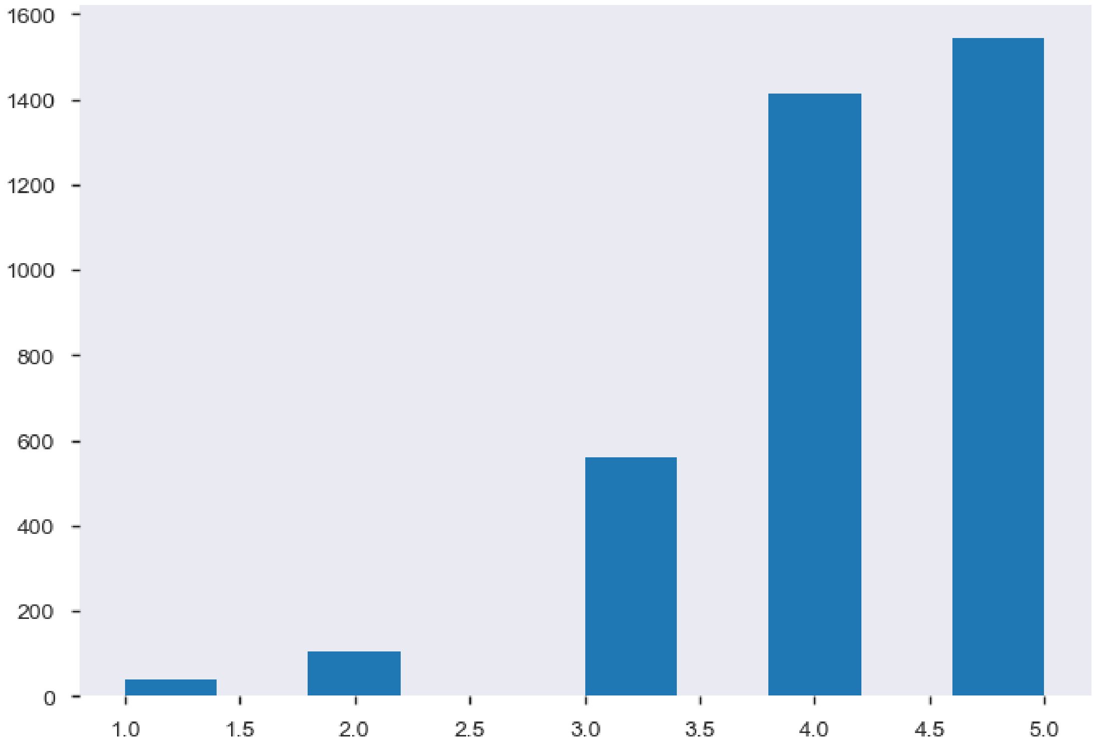
```
[19]:        UserID  MovieID  Rating     Timestamp Gender   Age  Occupation  \
      41626     1.0        1     5.0  9.788243e+08      F   1.0        10.0
      41627     6.0        1     4.0  9.782370e+08      F  50.0         9.0
      41628     8.0        1     4.0  9.782335e+08      M  25.0        12.0
      41629     9.0        1     5.0  9.782260e+08      M  25.0        17.0
      41630    10.0        1     5.0  9.782265e+08      F  35.0         1.0
      41631    18.0        1     4.0  9.781548e+08      F  18.0         3.0
      41632    19.0        1     5.0  9.785560e+08      M   1.0        10.0
      41633    21.0        1     3.0  9.781393e+08      M  18.0        16.0
      41634    23.0        1     4.0  9.784636e+08      M  35.0         0.0
      41635    26.0        1     3.0  9.781307e+08      M  25.0         7.0
      41636    28.0        1     3.0  9.789853e+08      F  25.0         1.0
      41637    34.0        1     5.0  9.781030e+08      F  18.0         0.0
      41638    36.0        1     5.0  9.780613e+08      M  25.0         3.0
      41639    38.0        1     5.0  9.780462e+08      F  18.0         4.0
      41640    44.0        1     5.0  9.780194e+08      M  45.0        17.0
      41641    45.0        1     4.0  9.779900e+08      F  45.0        16.0
      41642    48.0        1     4.0  9.779759e+08      M  25.0         4.0
      41643    49.0        1     5.0  9.779725e+08      M  18.0        12.0
      41644    51.0        1     5.0  9.779478e+08      F   1.0        10.0
      41645    56.0        1     5.0  9.779389e+08      M  35.0        20.0
      41646    60.0        1     4.0  9.779320e+08      M  50.0         1.0
      41647    65.0        1     5.0  9.913688e+08      M  35.0        12.0
      41648    68.0        1     3.0  9.913760e+08      M  18.0         4.0
      41649    73.0        1     3.0  9.778678e+08      M  18.0         4.0
      41650    75.0        1     5.0  9.778511e+08      F   1.0        10.0
      41651    76.0        1     5.0  9.778471e+08      M  35.0         7.0
      41652    78.0        1     4.0  9.785706e+08      F  45.0         1.0
      41653    80.0        1     3.0  9.777869e+08      M  56.0         1.0
      41654    90.0        1     3.0  9.938729e+08      M  56.0        13.0
      41655    92.0        1     4.0  9.776468e+08      F  18.0         4.0
      …       …      …     …           …    …   …         …
      56801  5905.0     3114     5.0  9.573757e+08      F  35.0        20.0
      56802  5908.0     3114     4.0  9.573736e+08      M  25.0         4.0
      56803  5917.0     3114     5.0  9.576779e+08      F  50.0         1.0
      56804  5922.0     3114     5.0  9.574700e+08      M  56.0         3.0
      56805  5930.0     3114     4.0  9.572325e+08      F  35.0        17.0
      56806  5933.0     3114     5.0  9.572206e+08      M  25.0         2.0
      56807  5943.0     3114     5.0  9.572014e+08      F  45.0         1.0
      56808  5948.0     3114     5.0  1.013430e+09      M  56.0        13.0
      56809  5953.0     3114     5.0  9.571428e+08      M   1.0        10.0
      56810  5964.0     3114     5.0  9.569939e+08      M  18.0         5.0
      56811  5971.0     3114     5.0  9.569546e+08      M  35.0         7.0
      56812  5972.0     3114     5.0  9.762056e+08      F  25.0        20.0
      56813  5975.0     3114     5.0  9.569466e+08      M  25.0        14.0
      56814  5980.0     3114     3.0  9.569379e+08      M  56.0         1.0
      56815  5981.0     3114     5.0  9.569316e+08      M  35.0         7.0
```

```
56816  5982.0     3114     3.0  9.569358e+08      M  35.0           1.0
56817  5985.0     3114     4.0  9.611180e+08      F  18.0           4.0
56818  5989.0     3114     5.0  9.568736e+08      F   1.0          10.0
56819  5991.0     3114     5.0  1.000093e+09      F  35.0          20.0
56820  5992.0     3114     5.0  9.568655e+08      F  18.0           4.0
56821  5995.0     3114     5.0  9.568559e+08      F  35.0           1.0
56822  5996.0     3114     5.0  9.597986e+08      F  25.0           0.0
56823  6000.0     3114     3.0  9.568789e+08      M  45.0          17.0
56824  6015.0     3114     5.0  9.567787e+08      F  25.0           9.0
56825  6016.0     3114     5.0  9.567788e+08      M  45.0           1.0
56826  6022.0     3114     5.0  9.567557e+08      M  25.0          17.0
56827  6024.0     3114     4.0  9.567494e+08      M  25.0          12.0
56828  6027.0     3114     4.0  9.567268e+08      M  18.0           4.0
56829  6036.0     3114     4.0  9.567102e+08      F  25.0          15.0
56830  6037.0     3114     4.0  9.567192e+08      F  45.0           1.0

         Zip-code               Title                       Genres
41626       48067    Toy Story (1995)  Animation|Children's|Comedy
41627       55117    Toy Story (1995)  Animation|Children's|Comedy
41628       11413    Toy Story (1995)  Animation|Children's|Comedy
41629       61614    Toy Story (1995)  Animation|Children's|Comedy
41630       95370    Toy Story (1995)  Animation|Children's|Comedy
41631       95825    Toy Story (1995)  Animation|Children's|Comedy
41632       48073    Toy Story (1995)  Animation|Children's|Comedy
41633       99353    Toy Story (1995)  Animation|Children's|Comedy
41634       90049    Toy Story (1995)  Animation|Children's|Comedy
41635       23112    Toy Story (1995)  Animation|Children's|Comedy
41636       14607    Toy Story (1995)  Animation|Children's|Comedy
41637       02135    Toy Story (1995)  Animation|Children's|Comedy
41638       94123    Toy Story (1995)  Animation|Children's|Comedy
41639       02215    Toy Story (1995)  Animation|Children's|Comedy
41640       98052    Toy Story (1995)  Animation|Children's|Comedy
41641       94110    Toy Story (1995)  Animation|Children's|Comedy
41642       92107    Toy Story (1995)  Animation|Children's|Comedy
41643       77084    Toy Story (1995)  Animation|Children's|Comedy
41644       10562    Toy Story (1995)  Animation|Children's|Comedy
41645       60440    Toy Story (1995)  Animation|Children's|Comedy
41646       72118    Toy Story (1995)  Animation|Children's|Comedy
41647       55803    Toy Story (1995)  Animation|Children's|Comedy
41648       53706    Toy Story (1995)  Animation|Children's|Comedy
41649       53706    Toy Story (1995)  Animation|Children's|Comedy
41650       01748    Toy Story (1995)  Animation|Children's|Comedy
41651       55413    Toy Story (1995)  Animation|Children's|Comedy
41652       98029    Toy Story (1995)  Animation|Children's|Comedy
41653       49327    Toy Story (1995)  Animation|Children's|Comedy
41654       85749    Toy Story (1995)  Animation|Children's|Comedy
41655       44243    Toy Story (1995)  Animation|Children's|Comedy
```

```
…           …                  …                          …
56801       78006  Toy Story 2 (1999)  Animation|Children's|Comedy
56802       19711  Toy Story 2 (1999)  Animation|Children's|Comedy
56803       94550  Toy Story 2 (1999)  Animation|Children's|Comedy
56804       94561  Toy Story 2 (1999)  Animation|Children's|Comedy
56805       78681  Toy Story 2 (1999)  Animation|Children's|Comedy
56806       98227  Toy Story 2 (1999)  Animation|Children's|Comedy
56807       19806  Toy Story 2 (1999)  Animation|Children's|Comedy
56808       12124  Toy Story 2 (1999)  Animation|Children's|Comedy
56809       21030  Toy Story 2 (1999)  Animation|Children's|Comedy
56810       97202  Toy Story 2 (1999)  Animation|Children's|Comedy
56811       49504  Toy Story 2 (1999)  Animation|Children's|Comedy
56812       55428  Toy Story 2 (1999)  Animation|Children's|Comedy
56813       55104  Toy Story 2 (1999)  Animation|Children's|Comedy
56814       42503  Toy Story 2 (1999)  Animation|Children's|Comedy
56815       01776  Toy Story 2 (1999)  Animation|Children's|Comedy
56816       56082  Toy Story 2 (1999)  Animation|Children's|Comedy
56817  78705-5221  Toy Story 2 (1999)  Animation|Children's|Comedy
56818       74114  Toy Story 2 (1999)  Animation|Children's|Comedy
56819       94025  Toy Story 2 (1999)  Animation|Children's|Comedy
56820       21046  Toy Story 2 (1999)  Animation|Children's|Comedy
56821       14618  Toy Story 2 (1999)  Animation|Children's|Comedy
56822       87114  Toy Story 2 (1999)  Animation|Children's|Comedy
56823       30075  Toy Story 2 (1999)  Animation|Children's|Comedy
56824       80013  Toy Story 2 (1999)  Animation|Children's|Comedy
56825       37209  Toy Story 2 (1999)  Animation|Children's|Comedy
56826       57006  Toy Story 2 (1999)  Animation|Children's|Comedy
56827       53705  Toy Story 2 (1999)  Animation|Children's|Comedy
56828       20742  Toy Story 2 (1999)  Animation|Children's|Comedy
56829       32603  Toy Story 2 (1999)  Animation|Children's|Comedy
56830       76006  Toy Story 2 (1999)  Animation|Children's|Comedy

[3662 rows x 10 columns]
```

```
[20]: toystory.Rating.hist(grid=False)
```

```
[20]: <matplotlib.axes._subplots.AxesSubplot at 0x291020f6400>
```



1.0.7 Top 25 movies by viewership rating

```
[21]: top_25 = df.groupby(["MovieID", "Title"]).Timestamp.count().
        ↪sort_values(ascending=False)
      top_25
```

```
[21]: MovieID  Title
      2858     American Beauty (1999)                                3428
      260      Star Wars: Episode IV - A New Hope (1977)             2991
      1196     Star Wars: Episode V - The Empire Strikes Back (1980)      2990
      1210     Star Wars: Episode VI - Return of the Jedi (1983)     2883
      480      Jurassic Park (1993)                                  2672
      2028     Saving Private Ryan (1998)                            2653
      589      Terminator 2: Judgment Day (1991)                     2649
      2571     Matrix, The (1999)                                    2590
      1270     Back to the Future (1985)                             2583
      593      Silence of the Lambs, The (1991)                      2578
      1580     Men in Black (1997)                                   2538
      1198     Raiders of the Lost Ark (1981)                        2514
      608      Fargo (1996)                                          2513
      2762     Sixth Sense, The (1999)                               2459
      110      Braveheart (1995)                                     2443
```

```
2396     Shakespeare in Love (1998)                                  2369
1197     Princess Bride, The (1987)                                  2318
527      Schindler's List (1993)                                     2304
1617     L.A. Confidential (1997)                                    2288
1265     Groundhog Day (1993)                                        2278
1097     E.T. the Extra-Terrestrial (1982)                           2269
2628     Star Wars: Episode I - The Phantom Menace (1999)            2250
2997     Being John Malkovich (1999)                                 2241
318      Shawshank Redemption, The (1994)                            2227
858      Godfather, The (1972)                                       2223
356      Forrest Gump (1994)                                         2194
2716     Ghostbusters (1984)                                         2181
296      Pulp Fiction (1994)                                         2171
1240     Terminator, The (1984)                                      2098
1        Toy Story (1995)                                            2077
                                                                     …
624      Condition Red (1995)                                           1
2213     Waltzes from Vienna (1933)                                     1
2619     Mascara (1999)                                                 1
396      Fall Time (1995)                                               1
2039     Cheetah (1989)                                                 1
2277     Somewhere in the City (1997)                                   1
1843     Slappy and the Stinkers (1998)                                 1
3904     Uninvited Guest, An (2000)                                     1
2254     Choices (1981)                                                 1
3607     One Little Indian (1973)                                       1
226      Dream Man (1995)                                               1
1709     Legal Deceit (1997)                                            1
3881     Bittersweet Motel (2000)                                       1
3647     Running Free (2000)                                            1
658      Billy's Holiday (1995)                                         1
3172     Ulysses (Ulisse) (1954)                                        1
655      Mutters Courage (1995)                                         1
2235     One Man's Hero (1999)                                          1
651      Superweib, Das (1996)                                          1
644      Happy Weekend (1996)                                           1
3220     Night Tide (1961)                                              1
2226     Ring, The (1927)                                               1
3656     Lured (1947)                                                   1
642      Roula (1995)                                                   1
641      Little Indian, Big City (Un indien dans la ville) (1994)       1
2218     Juno and Paycock (1930)                                        1
2217     Elstree Calling (1930)                                         1
3382     Song of Freedom (1936)                                         1
2214     Number Seventeen (1932)                                        1
402      Open Season (1996)                                             1
Name: Timestamp, Length: 3706, dtype: int64
```

```
[22]: print('Top 25 movies by viewership rating')
      print(top_25[:25])
```

```
Top 25 movies by viewership rating
MovieID  Title
2858     American Beauty (1999)                                   3428
260      Star Wars: Episode IV - A New Hope (1977)                2991
1196     Star Wars: Episode V - The Empire Strikes Back (1980)    2990
1210     Star Wars: Episode VI - Return of the Jedi (1983)        2883
480      Jurassic Park (1993)                                     2672
2028     Saving Private Ryan (1998)                               2653
589      Terminator 2: Judgment Day (1991)                        2649
2571     Matrix, The (1999)                                       2590
1270     Back to the Future (1985)                                2583
593      Silence of the Lambs, The (1991)                         2578
1580     Men in Black (1997)                                      2538
1198     Raiders of the Lost Ark (1981)                           2514
608      Fargo (1996)                                             2513
2762     Sixth Sense, The (1999)                                  2459
110      Braveheart (1995)                                        2443
2396     Shakespeare in Love (1998)                               2369
1197     Princess Bride, The (1987)                               2318
527      Schindler's List (1993)                                  2304
1617     L.A. Confidential (1997)                                 2288
1265     Groundhog Day (1993)                                     2278
1097     E.T. the Extra-Terrestrial (1982)                        2269
2628     Star Wars: Episode I - The Phantom Menace (1999)         2250
2997     Being John Malkovich (1999)                              2241
318      Shawshank Redemption, The (1994)                         2227
858      Godfather, The (1972)                                    2223
Name: Timestamp, dtype: int64
```

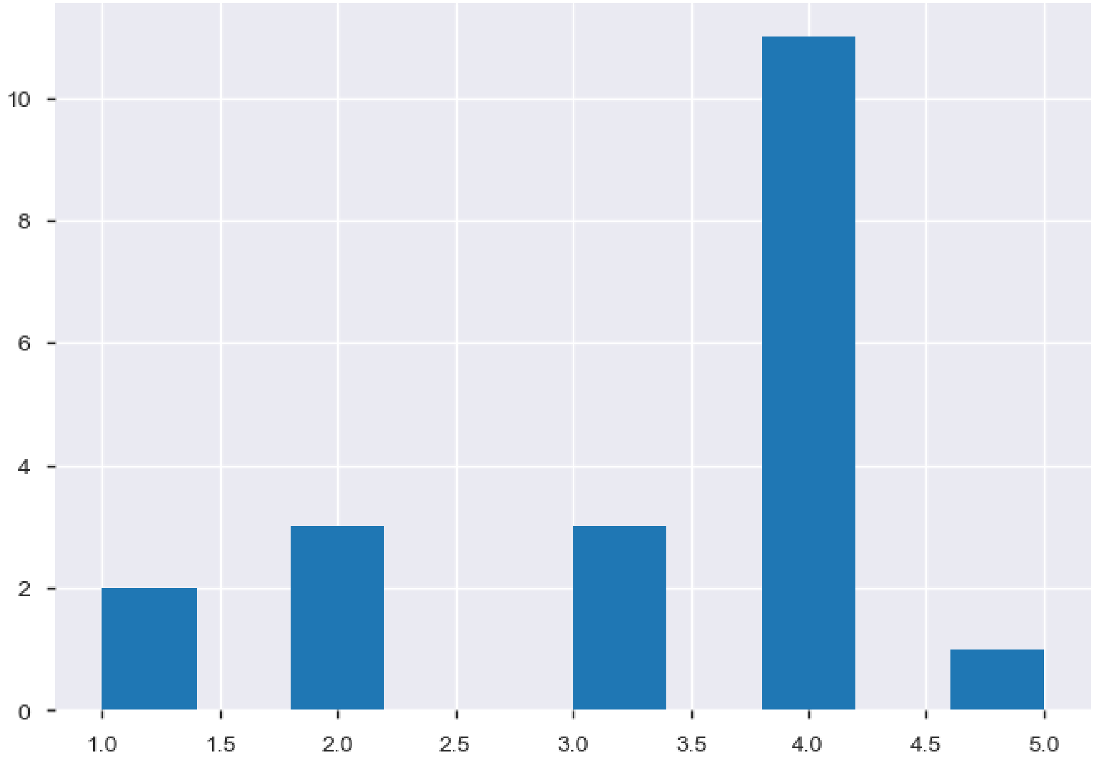
1.0.8 The ratings for all the movies reviewed by for a particular user of user id = 2696

```
[23]: usr_2696 = df.loc[df.UserID==2696, "Rating"].sort_values(ascending=False)
      usr_2696.head(),usr_2696.shape
```

```
[23]: (250014    5.0
       603189    4.0
       371178    4.0
       689379    4.0
       618708    4.0
       Name: Rating, dtype: float64, (20,))
```

```
[24]: usr_2696.hist()
```

```
[24]: <matplotlib.axes._subplots.AxesSubplot at 0x29102103cf8>
```



1.0.9 Finding all the unique genres

```
[25]: df.Genres.unique()
```

```
[25]: array(['Drama', "Animation|Children's|Musical", 'Musical|Romance',
             "Animation|Children's|Comedy", 'Action|Adventure|Comedy|Romance',
             'Action|Adventure|Drama', 'Comedy|Drama',
             "Adventure|Children's|Drama|Musical", 'Musical', 'Comedy',
             "Animation|Children's", 'Comedy|Fantasy', 'Animation',
             'Comedy|Sci-Fi', 'Drama|War', 'Romance',
             "Animation|Children's|Musical|Romance",
             "Children's|Drama|Fantasy|Sci-Fi", 'Drama|Romance',
             'Animation|Comedy|Thriller',
             "Adventure|Animation|Children's|Comedy|Musical",
             "Animation|Children's|Comedy|Musical", 'Thriller',
             'Action|Crime|Romance', 'Action|Adventure|Fantasy|Sci-Fi',
             "Children's|Comedy|Musical", 'Action|Drama|War',
             "Children's|Drama", 'Crime|Drama|Thriller', 'Action|Crime|Drama',
             'Action|Adventure|Mystery', 'Crime|Drama',
             'Action|Adventure|Sci-Fi|Thriller',
             'Action|Adventure|Romance|Sci-Fi|War', 'Action|Thriller',
             'Action|Drama', 'Comedy|Drama|Western', 'Action|Adventure|Crime',
             'Action|Crime|Mystery|Thriller', 'Comedy|Drama|Romance',
             'Comedy|Drama|War', 'Drama|Sci-Fi', 'Action|Drama|Thriller',
             'Action|Comedy|Western', 'Adventure|Comedy|Drama',
             'Drama|Thriller', 'Comedy|Romance',
             'Action|Drama|Romance|Thriller', 'Action|Crime|Thriller',
             'Action|Sci-Fi|Thriller', 'Action|Horror|Sci-Fi', 'Action|Sci-Fi',
             'Action|Romance|War', 'Adventure|Drama|Romance|Sci-Fi',
             'Action|Adventure|Sci-Fi', 'Drama|Romance|War',
             'Action|Drama|Romance', 'Crime|Drama|Film-Noir|Thriller',
             'Adventure|Drama|Western', 'Action|Adventure|Drama|Sci-Fi|War',
             'Action|Adventure|Thriller', 'Action|Adventure|Romance|Thriller',
             'Action|Adventure', 'Comedy|Horror', 'Action|Crime|Drama|Thriller',
             'Action|Mystery|Romance|Thriller', 'Action|Romance|Thriller',
             'Action|Comedy|Drama', 'Action', 'Action|Sci-Fi|War',
             'Action|Comedy|Crime|Drama', 'Action|Adventure|Romance',
             'Comedy|Romance|War', 'Comedy|Thriller', 'Action|Adventure|Comedy',
             'Action|Comedy', 'Adventure|Thriller', 'Action|Adventure|Fantasy',
             'Action|Adventure|Horror', 'Action|Adventure|Comedy|Sci-Fi',
             'Action|Adventure|Comedy|Horror', 'Western', 'Adventure|Comedy',
             'Adventure|Drama', 'Action|Adventure|Horror|Thriller',
             'Comedy|Western', "Animation|Children's|Comedy|Musical|Romance",
             'Action|Western', 'Action|Horror|Sci-Fi|Thriller', 'Action|Horror',
             'Adventure|Animation|Film-Noir', 'Drama|Romance|Thriller',
             'Crime|Drama|Romance|Thriller', 'Crime|Thriller',
             'Animation|Comedy', 'Documentary',
             'Crime|Film-Noir|Mystery|Thriller', 'Drama|Horror',
             'Mystery|Sci-Fi|Thriller', 'Drama|Mystery', 'Horror|Romance',
             'Horror|Sci-Fi', 'Horror', 'Sci-Fi|Thriller', 'Crime',
             'Action|Crime', 'Crime|Horror', 'Drama|Mystery|Thriller',
             'Comedy|Crime', 'Drama|Sci-Fi|Thriller', "Children's|Comedy",
             'Horror|Mystery|Thriller', 'Film-Noir|Mystery',
             'Comedy|Crime|Mystery|Thriller', 'Drama|Musical',
             'Adventure|Sci-Fi', "Children's|Comedy|Drama", 'Action|Romance',
             "Adventure|Animation|Children's|Musical", 'Comedy|Musical',
             "Children's|Fantasy|Musical", "Children's|Comedy|Western",
             'Drama|Romance|War|Western', "Adventure|Children's|Comedy",
             'Comedy|Fantasy|Romance', 'Comedy|Musical|Romance',
             "Adventure|Children's|Drama", 'Action|Drama|Thriller|War',
             'Drama|Thriller|War', 'Adventure|Animation|Sci-Fi|Thriller',
             'Animation|Sci-Fi', 'Comedy|Crime|Drama|Mystery',
             'Crime|Drama|Mystery', 'Action|Comedy|Sci-Fi|Thriller',
             'Comedy|Crime|Fantasy', 'Horror|Sci-Fi|Thriller',
             "Adventure|Children's|Comedy|Fantasy|Sci-Fi",
             'Film-Noir|Mystery|Thriller', 'Adventure', 'Comedy|War',
             'Comedy|Romance|Thriller', "Action|Children's|Fantasy",
             "Adventure|Children's|Fantasy", 'Action|Adventure|Comedy|Crime',
             'Adventure|Musical', "Animation|Children's|Drama|Fantasy",
             'Comedy|Mystery|Thriller', 'Action|Adventure|Crime|Drama',
             "Children's|Fantasy|Sci-Fi", "Adventure|Children's", 'War',
             'Comedy|Horror|Musical|Sci-Fi', "Children's|Comedy|Fantasy",
             'Sci-Fi|War', "Animation|Children's|Fantasy|Musical",
             "Children's|Sci-Fi", "Adventure|Children's|Fantasy|Sci-Fi",
             'Mystery|Thriller', 'Comedy|Horror|Musical',
             'Action|Horror|Thriller', 'Adventure|Fantasy',
             'Drama|Mystery|Sci-Fi|Thriller', 'Crime|Drama|Sci-Fi',
             "Adventure|Children's|Musical", 'Action|Sci-Fi|Thriller|War',
             'Adventure|War', 'Action|Adventure|Romance|War',
             'Action|Drama|Fantasy|Romance', 'Adventure|Comedy|Sci-Fi',
             'Comedy|Sci-Fi|Western', 'Action|Adventure|Comedy|Horror|Sci-Fi',
             "Adventure|Children's|Comedy|Fantasy", 'Film-Noir|Sci-Fi',
             'Drama|Fantasy', "Children's|Drama|Fantasy", "Children's|Fantasy",
             'Fantasy|Sci-Fi', 'Action|Comedy|Musical',
             'Adventure|Fantasy|Sci-Fi', 'Action|Adventure|Sci-Fi|War',
             "Action|Adventure|Children's|Comedy",
             "Adventure|Children's|Drama|Romance",
             "Adventure|Children's|Sci-Fi", "Children's",
             'Comedy|Drama|Musical', 'Comedy|Fantasy|Romance|Sci-Fi',
             'Comedy|Crime|Drama', 'Sci-Fi', 'Adventure|Fantasy|Romance',
             'Adventure|Romance', 'Adventure|Western', 'Action|Drama|Mystery',
             'Adventure|Animation|Sci-Fi', 'Adventure|Romance|Sci-Fi',
             'Horror|Thriller', 'Action|Adventure|Mystery|Sci-Fi',
             'Adventure|Drama|Thriller', 'Comedy|Horror|Thriller',
             'Action|Comedy|Crime|Horror|Thriller',
             'Crime|Horror|Mystery|Thriller', 'Crime|Horror|Thriller',
             'Crime|Drama|Mystery|Thriller', 'Animation|Musical',
             'Action|Sci-Fi|Western', 'Crime|Drama|Film-Noir',
             'Adventure|Sci-Fi|Thriller', 'Drama|Fantasy|Romance|Thriller',
             'Mystery|Sci-Fi', 'Action|Crime|Sci-Fi', 'Comedy|Mystery',
             'Action|Romance|Sci-Fi', 'Crime|Film-Noir|Mystery',
             'Comedy|Drama|Sci-Fi', 'Sci-Fi|Thriller|War', 'Film-Noir|Thriller',
             'Action|Adventure|Animation|Horror|Sci-Fi',
             'Action|Sci-Fi|Thriller|Western', 'Comedy|Horror|Sci-Fi',
             'Crime|Film-Noir|Thriller', 'Comedy|Crime|Thriller',
             'Film-Noir|Sci-Fi|Thriller',
             "Adventure|Animation|Children's|Sci-Fi",
             'Action|Adventure|Drama|Romance', "Children's|Musical",
             'Action|Comedy|Musical|Sci-Fi', 'Action|Drama|Sci-Fi|Thriller',
             'Action|Comedy|Fantasy', 'Action|War', 'Action|Comedy|Sci-Fi|War',
             'Comedy|Crime|Horror', 'Action|Comedy|War',
             "Action|Adventure|Children's|Sci-Fi", "Action|Children's",
             'Comedy|Documentary', 'Action|Adventure|Animation',
             'Action|Mystery|Thriller',
             "Action|Animation|Children's|Sci-Fi|Thriller|War",
             'Crime|Drama|Romance', 'Crime|Film-Noir',
             'Mystery|Romance|Thriller', 'Comedy|Mystery|Romance|Thriller',
             'Action|Adventure|Sci-Fi|Thriller|War',
             'Adventure|Crime|Sci-Fi|Thriller', 'Action|Adventure|Western',
             "Animation|Children's|Fantasy|War", 'Action|Adventure|Comedy|War',
             "Children's|Comedy|Sci-Fi",
             "Adventure|Animation|Children's|Comedy|Fantasy",
             'Drama|Musical|War', 'Drama|Mystery|Romance',
             'Adventure|Drama|Romance', 'Film-Noir',
             'Film-Noir|Romance|Thriller', 'Drama|Film-Noir',
             'Romance|Thriller', 'Action|Adventure|War', 'Mystery',
             'Action|Adventure|Drama|Thriller', 'Musical|Romance|War',
             'Drama|Western', 'Action|Drama|Mystery|Romance|Thriller',
             'Adventure|Comedy|Musical', 'Documentary|Musical',
             'Action|Thriller|War', 'Adventure|Comedy|Romance',
             "Adventure|Children's|Comedy|Fantasy|Romance", 'Romance|War',
             'Comedy|Romance|Sci-Fi', 'Action|Mystery|Sci-Fi|Thriller',
             "Children's|Horror", 'Adventure|Musical|Romance',
             "Adventure|Children's|Comedy|Musical", "Children's|Comedy|Mystery",
             'Action|Comedy|Romance|Thriller', 'Action|Drama|Western',
             "Animation|Children's|Comedy|Romance", 'Comedy|Mystery|Romance',
             'Action|Crime|Mystery', 'Comedy|Drama|Thriller', 'Musical|War',
             'Documentary|Drama', 'Action|Adventure|Crime|Thriller',
             "Action|Adventure|Children's", "Adventure|Children's|Romance",
             "Adventure|Animation|Children's",
             "Action|Adventure|Animation|Children's|Fantasy",
             "Adventure|Animation|Children's|Fantasy",
             'Drama|Film-Noir|Thriller', 'Crime|Mystery', 'Documentary|War',
             'Action|Comedy|Crime', 'Drama|Romance|Sci-Fi', 'Horror|Mystery',
             'Drama|Horror|Thriller', "Action|Adventure|Children's|Fantasy",
             'Animation|Mystery', 'Drama|Romance|Western', 'Romance|Western',
             'Comedy|Film-Noir|Thriller', 'Fantasy', 'Film-Noir|Horror'],
            dtype=object)
```

[26]:

```
Genres_list = df.Genres.tolist()
genre_list = []
i = 0
while(i<len(Genres_list)):
    genre_list+= Genres_list[i].split('|')
    i+=1
```

[27]:

```
unique_gen = list(set(genre_list))
print(unique_gen)
print()
print("Length of the unique Genre : ",len(unique_gen))
```

```
["Children's", 'Animation', 'Fantasy', 'War', 'Musical', 'Mystery', 'Western',
'Thriller', 'Drama', 'Sci-Fi', 'Film-Noir', 'Crime', 'Horror', 'Action',
'Comedy', 'Adventure', 'Romance', 'Documentary']

Length of the unique Genre :  18
```

1.0.10 Creating a separate column for each genre category with a one-hot encoding (1 and 0)

[28]:

```
new_data = pd.concat([df,df.Genres.str.get_dummies()], axis=1)
print(new_data.columns)
```

```
Index(['UserID', 'MovieID', 'Rating', 'Timestamp', 'Gender', 'Age',
       'Occupation', 'Zip-code', 'Title', 'Genres', 'Action', 'Adventure',
       'Animation', "Children's", 'Comedy', 'Crime', 'Documentary', 'Drama',
       'Fantasy', 'Film-Noir', 'Horror', 'Musical', 'Mystery', 'Romance',
       'Sci-Fi', 'Thriller', 'War', 'Western'],
      dtype='object')
```

[29]:

```
new_data.head()
```

[29]:

```
   UserID  MovieID  Rating    Timestamp Gender   Age  Occupation Zip-code  \
0     1.0     1193     5.0  978300760.0      F   1.0        10.0    48067
1     2.0     1193     5.0  978298413.0      M  56.0        16.0    70072
2    12.0     1193     4.0  978220179.0      M  25.0        12.0    32793
3    15.0     1193     4.0  978199279.0      M  25.0         7.0    22903
4    17.0     1193     5.0  978158471.0      M  50.0         1.0    95350

                                    Title Genres  …  Fantasy  Film-Noir  \
0  One Flew Over the Cuckoo's Nest (1975)  Drama  …        0          0
1  One Flew Over the Cuckoo's Nest (1975)  Drama  …        0          0
2  One Flew Over the Cuckoo's Nest (1975)  Drama  …        0          0
3  One Flew Over the Cuckoo's Nest (1975)  Drama  …        0          0
4  One Flew Over the Cuckoo's Nest (1975)  Drama  …        0          0

   Horror  Musical  Mystery  Romance  Sci-Fi  Thriller  War  Western
0       0        0        0        0       0         0    0        0
1       0        0        0        0       0         0    0        0
2       0        0        0        0       0         0    0        0
3       0        0        0        0       0         0    0        0
4       0        0        0        0       0         0    0        0

[5 rows x 28 columns]
```

[30]:

```
df_new = new_data.drop(['Title','Zip-code','Timestamp','Genres'],axis=1)
df_new.head()
```

```
[30]:    UserID  MovieID  Rating Gender   Age  Occupation  Action  Adventure  \
      0     1.0     1193     5.0      F   1.0        10.0       0          0
      1     2.0     1193     5.0      M  56.0        16.0       0          0
      2    12.0     1193     4.0      M  25.0        12.0       0          0
      3    15.0     1193     4.0      M  25.0         7.0       0          0
      4    17.0     1193     5.0      M  50.0         1.0       0          0

         Animation  Children's  ...  Fantasy  Film-Noir  Horror  Musical  Mystery  \
      0          0           0  ...        0          0       0        0        0
      1          0           0  ...        0          0       0        0        0
      2          0           0  ...        0          0       0        0        0
      3          0           0  ...        0          0       0        0        0
      4          0           0  ...        0          0       0        0        0

         Romance  Sci-Fi  Thriller  War  Western
      0        0       0         0    0        0
      1        0       0         0    0        0
      2        0       0         0    0        0
      3        0       0         0    0        0
      4        0       0         0    0        0

      [5 rows x 24 columns]
```

```
[31]: print(df_new.columns)
```

```
Index(['UserID', 'MovieID', 'Rating', 'Gender', 'Age', 'Occupation', 'Action',
       'Adventure', 'Animation', 'Children's', 'Comedy', 'Crime',
       'Documentary', 'Drama', 'Fantasy', 'Film-Noir', 'Horror', 'Musical',
       'Mystery', 'Romance', 'Sci-Fi', 'Thriller', 'War', 'Western'],
      dtype='object')
```

1.0.11 Encoding the gender column

```
[32]: df_new.Gender = pd.get_dummies(df_new.Gender)
```

```
[33]: x = df_new.drop(['UserID','MovieID','Rating'],axis=1)
      x.shape
```

```
[33]: (1000209, 21)
```

1.0.12 The features affecting the ratings of any particular movie.

```
[34]: print('The features affecting the ratings of any particular movie:')
      print()
      print(x.columns)
```

```
The features affecting the ratings of any particular movie:
```

```
Index(['Gender', 'Age', 'Occupation', 'Action', 'Adventure', 'Animation',
       'Children's', 'Comedy', 'Crime', 'Documentary', 'Drama', 'Fantasy',
       'Film-Noir', 'Horror', 'Musical', 'Mystery', 'Romance', 'Sci-Fi',
       'Thriller', 'War', 'Western'],
      dtype='object')
```

```
[35]: y = df_new.Rating
      y.shape
```

```
[35]: (1000209,)
```

```
[36]: x.Occupation.value_counts()
```

```
[36]: 4.0     131032
      0.0     130499
      7.0     105425
      1.0      85351
      17.0     72816
      20.0     60397
      12.0     57214
      2.0      50068
      14.0     49109
      16.0     46021
      6.0      37205
      3.0      31623
      10.0     23290
      15.0     22951
      5.0      21850
      11.0     20563
      19.0     14904
      13.0     13754
      18.0     12086
      9.0      11345
      8.0       2706
      Name: Occupation, dtype: int64
```

```
[37]: x = x.join(pd.get_dummies(x.Occupation,prefix='Occupation'))
      x.head(),x.columns
```

```
[37]: (   Gender   Age  Occupation  Action  Adventure  Animation  Children's  Comedy  \
       0       1   1.0        10.0       0          0          0           0       0
       1       0  56.0        16.0       0          0          0           0       0
       2       0  25.0        12.0       0          0          0           0       0
       3       0  25.0         7.0       0          0          0           0       0
       4       0  50.0         1.0       0          0          0           0       0
```

```
    Crime  Documentary  …  Occupation_11.0  Occupation_12.0  Occupation_13.0
\
 0      0            0  …                0                0                0
 1      0            0  …                0                0                0
 2      0            0  …                0                1                0
 3      0            0  …                0                0                0
 4      0            0  …                0                0                0

    Occupation_14.0  Occupation_15.0  Occupation_16.0  Occupation_17.0  \
 0                0                0                0                0
 1                0                0                1                0
 2                0                0                0                0
 3                0                0                0                0
 4                0                0                0                0

    Occupation_18.0  Occupation_19.0  Occupation_20.0
 0                0                0                0
 1                0                0                0
 2                0                0                0
 3                0                0                0
 4                0                0                0

 [5 rows x 42 columns],
 Index(['Gender', 'Age', 'Occupation', 'Action', 'Adventure', 'Animation',
        'Children's', 'Comedy', 'Crime', 'Documentary', 'Drama', 'Fantasy',
        'Film-Noir', 'Horror', 'Musical', 'Mystery', 'Romance', 'Sci-Fi',
        'Thriller', 'War', 'Western', 'Occupation_0.0', 'Occupation_1.0',
        'Occupation_2.0', 'Occupation_3.0', 'Occupation_4.0', 'Occupation_5.0',
        'Occupation_6.0', 'Occupation_7.0', 'Occupation_8.0', 'Occupation_9.0',
        'Occupation_10.0', 'Occupation_11.0', 'Occupation_12.0',
        'Occupation_13.0', 'Occupation_14.0', 'Occupation_15.0',
        'Occupation_16.0', 'Occupation_17.0', 'Occupation_18.0',
        'Occupation_19.0', 'Occupation_20.0'],
       dtype='object'))
```

```
[38]: x = x.drop(['Occupation','Occupation_0.0'],axis=1)
      x.head(3),x.shape
```

```
[38]: (   Gender   Age  Action  Adventure  Animation  Children's  Comedy  Crime  \
       0       1   1.0       0          0          0           0       0      0
       1       0  56.0       0          0          0           0       0      0
       2       0  25.0       0          0          0           0       0      0

          Documentary  Drama  …  Occupation_11.0  Occupation_12.0  Occupation_13.0
      \
       0            0      1  …                0                0                0
       1            0      1  …               0                0                0
       2            0      1  …               0                1                0

          Occupation_14.0  Occupation_15.0  Occupation_16.0  Occupation_17.0  \
       0                0                0                0                0
       1                0                0                1                0
       2                0                0                0                0

          Occupation_18.0  Occupation_19.0  Occupation_20.0
       0                0                0                0
       1                0                0                0
       2                0                0                0

       [3 rows x 40 columns], (1000209, 40))
```

1.0.13 Deploying the hold out method

```
[39]: x_train, x_test, y_train, y_test = train_test_split(x,y,test_size=0.
       ↪2,random_state = 10,stratify=y)
```

1.0.14 Deploying the model

```
[40]: lgb = LGBMClassifier(boosting_type = 'gbdt',n_jobs= -1,objective='multiclass')
```

```
[41]: lgb.fit(x_train,y_train)
```

```
[41]: LGBMClassifier(boosting_type='gbdt', class_weight=None, colsample_bytree=1.0,
                     importance_type='split', learning_rate=0.1, max_depth=-1,
                     min_child_samples=20, min_child_weight=0.001, min_split_gain=0.0,
                     n_estimators=100, n_jobs=-1, num_leaves=31, objective='multiclass',
                     random_state=None, reg_alpha=0.0, reg_lambda=0.0, silent=True,
                     subsample=1.0, subsample_for_bin=200000, subsample_freq=0)
```

```
[42]: y_pred = lgb.predict(x_test)
```

```
[43]: print('LGBM accuracy score is : ', accuracy_score(y_test,y_pred)*100)
```

```
LGBM accuracy score is :  36.19589886123914
```

```
[44]: xgb = xgboost.XGBClassifier(n_jobs=-1)
```

```
[45]: xgb.fit(x_train,y_train)
```

```
[45]: XGBClassifier(base_score=0.5, booster='gbtree', colsample_bylevel=1,
                    colsample_bynode=1, colsample_bytree=1, gamma=0, learning_rate=0.1,
                    max_delta_step=0, max_depth=3, min_child_weight=1, missing=None,
                    n_estimators=100, n_jobs=-1, nthread=None,
                    objective='multi:softprob', random_state=0, reg_alpha=0,
                    reg_lambda=1, scale_pos_weight=1, seed=None, silent=None,
                    subsample=1, verbosity=1)
```

```
[46]: y_pred_xgb = xgb.predict(x_test)
```

```
[47]: print('XGB accuracy score is : ', accuracy_score(y_test,y_pred_xgb )*100)
```

```
XGB accuracy score is :  35.39156777076814
```

1.0.15 Accuracy score of both the model

LGBM accuracy score is : 36.19%

XGB accuracy score is : 35.39%