

Movielens Case Study

Learning project for Python Data Science using pandas library and matplotlib to analyze and visualize data collected from 3 data sets; Movies, Ratings, and Users data sets

Movielens Case Study

DESCRIPTION

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 :

Here, we ask you to perform the analysis using the Exploratory Data Analysis technique. You need to find features affecting the ratings of any particular movie and build a model to predict the movie ratings.

Domain: Entertainment

Analysis Tasks to be performed:

Import the three datasets Create a new dataset [Master_Data] with the following columns MovieID Title UserID Age Gender Occupation Rating. (Hint: (i) Merge two tables at a time. (ii) Merge the tables using two primary keys MovieID & UserID) Explore the datasets using visual representations (graphs or tables), also include your comments on the following: User Age Distribution User rating of the movie "Toy Story" Top 25 movies by viewership rating Find the ratings for all the movies reviewed by for a particular user of user id = 2696 Feature Engineering: Use column genres:

Find out all the unique genres (Hint: split the data in column genre making a list and then process the data to find out only the unique categories of genres) Create a separate column for each genre category with a one-hot encoding (1 and 0) whether or not the movie belongs to that genre. Determine the features affecting the ratings of any particular movie. Develop an appropriate model to predict the movie ratings

Dataset Description :

These files contain 1,000,209 anonymous ratings of approximately 3,900 movies made by 6,040 MovieLens users who joined MovieLens in 2000.

Ratings.dat

Format - UserID::MovieID::Rating::Timestamp

Field Description

UserID Unique identification for each user MovieID Unique identification for each movie
Rating User rating for each movie Timestamp Timestamp generated while adding user review
UserIDs range between 1 and 6040 The MovieIDs range between 1 and 3952
Ratings are made on a 5-star scale (whole-star ratings only) A timestamp is represented in seconds since the epoch is returned by time(2) Each user has at least 20 ratings

Users.dat

Format - UserID::Gender::Age::Occupation::Zip-code

Field Description UserID Unique identification for each user Genre Category of each movie
Age User's age Occupation User's Occupation Zip-code Zip Code for the user's location All demographic information is provided voluntarily by the users and is not checked for accuracy. Only users who have provided demographic information are included in this data set.

Gender is denoted by an "M" for male and "F" for female Age is chosen from the following ranges:

Value Description

1 "Under 18" 18 "18-24" 25 "25-34" 35 "35-44" 45 "45-49" 50 "50-55" 56 "56+"

Occupation is chosen from the following choices: Value Description 0 "other" or not specified 1 "academic/educator" 2 "artist" 3 "clerical/admin" 4 "college/grad student" 5 "customer service" 6 "doctor/health care" 7 "executive/managerial" 8 "farmer" 9 "homemaker" 10 "K-12 student" 11 "lawyer" 12 "programmer" 13 "retired" 14

"sales/marketing" 15 "scientist" 16 "self-employed" 17 "technician/engineer" 18
"tradesman/craftsman" 19 "unemployed" 20 "writer"

Movies.dat Format - MovieID::Title::Genres

Field Description MovieID Unique identification for each movie Title A title for each movie
Genres Category of each movie

Titles are identical to titles provided by the IMDB (including year of release)

Genres are pipe-separated and are selected from the following genres: Action Adventure
Animation Children's Comedy Crime Documentary Drama Fantasy Film-Noir Horror
Musical Mystery Romance Sci-Fi Thriller War Western Some MovieIDs do not correspond
to a movie due to accidental duplicate entries and/or test entries Movies are mostly
entered by hand, so errors and inconsistencies may exist

Import Important Library

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn import metrics
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import GaussianNB

warnings.simplefilter('ignore')
pd.set_option('display.max_column',30)
```

Analysis Task 1

```
ratings=pd.read_csv('ratings.dat',sep="::",header=None,names=['UserID',
,'MovieID','Rating','Timestamp'])
```

```
ratings.head()
```

	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

```
users=pd.read_csv('users.dat',sep="::",header=None,names=['UserID','Gender',
,'Age','Occupation','Zip-c'])
```

```
users.head()
```

	UserID	Gender	Age	Occupation	Zip-c
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

```
movies=pd.read_csv('movies.dat',sep="::",header=None,names=['MovieID',
'Title','Genres'],encoding=('ISO-8859-1'))
```

```
movies.head()
```

	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

```
# print ahape of all three datasets
print("Rating dataset Shape: ",ratings.shape)
print("User dataset Shape: ",users.shape)
print("movies dataset Shape: ",movies.shape)
```

```
Rating dataset Shape: (1000209, 4)
User dataset Shape: (6040, 5)
movies dataset Shape: (3883, 3)
```

Task 2

Create a new dataset [Master_Data] with the following columns MovieID Title UserID Age Gender Occupation Rating. (Hint: (i) Merge two tables at a time. (ii) Merge the tables using two primary keys MovieID & UserId)

```
# Merging two dataset movies and rating on the key MovieID
movie_ratings=pd.merge(movies,ratings,how='inner',on='MovieID')
```

```
# Merging the third dataset users on the UserID
df_final=pd.merge(movie_ratings,users,how='inner',on='UserID')
```

```
# Creating master data with necessary columns
Master_data=df_final[['MovieID','Title','UserID','Age','Gender','Occupation','Rating']]
```

```
# Checking for duplicates
Master_data.duplicated().sum()

0

# Master data information
Master_data.info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 1000209 entries, 0 to 1000208
Data columns (total 7 columns):
 #   Column          Non-Null Count  Dtype
---  -
 0   MovieID        1000209 non-null  int64
 1   Title          1000209 non-null  object
 2   UserID         1000209 non-null  int64
 3   Age            1000209 non-null  int64
 4   Gender         1000209 non-null  object
 5   Occupation     1000209 non-null  int64
 6   Rating         1000209 non-null  int64
dtypes: int64(5), object(2)
memory usage: 61.0+ MB
```

Task 3

-Explore the datasets using visual representations (graphs or tables), also include your comments on the following:

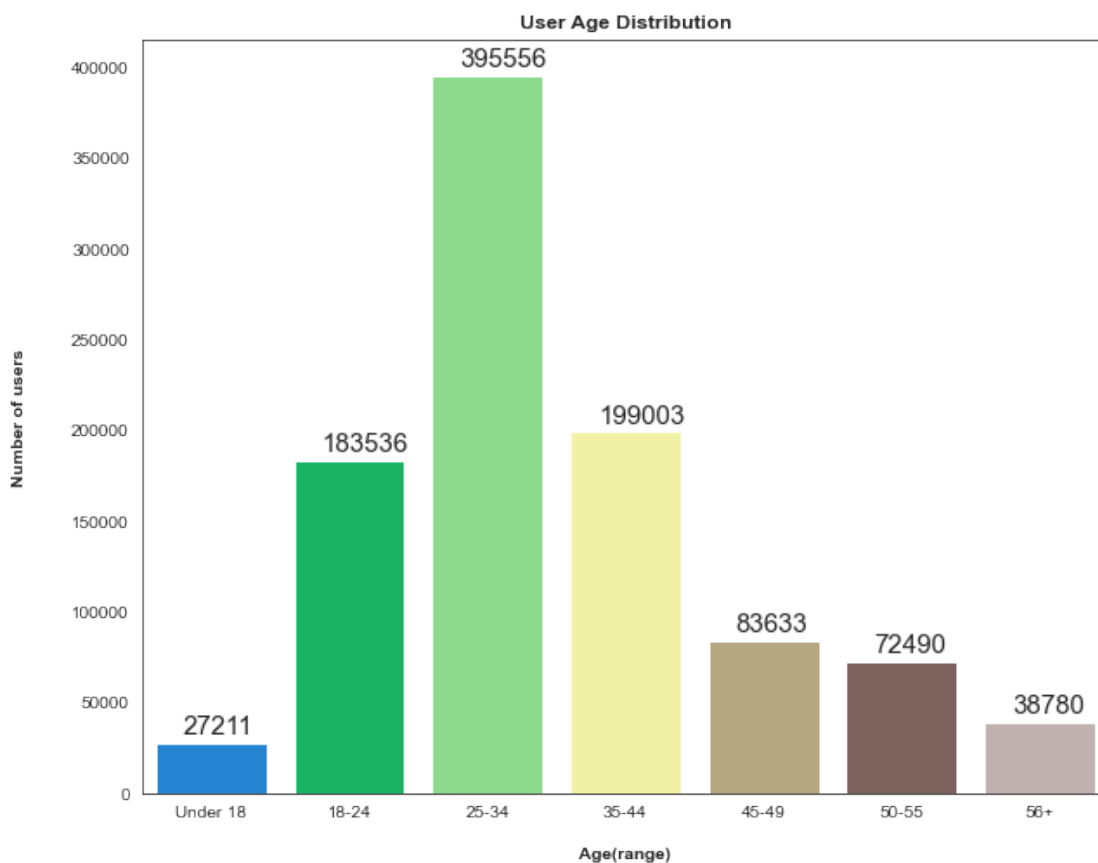
- 1.-User Age Distribution
- 2.-User rating of the movie "Toy Story"
- 3.-Top 25 movies by viewership rating
- 4.-Find the ratings for all the movies reviewed by for a particular user of user id = 2696

1.Age Distribution

```
# Finding Age Distribution
age_dist = Master_data['Age'].value_counts().to_frame()
age_dist.sort_index(inplace=True)
age_dist
```

	Age
1	27211
18	183536
25	395556
35	199003
45	83633
50	72490
56	38780

```
# Visual Representation of the age distribution
plt.figure(figsize=(10,8))
txt={'Weight':'bold'}
sns.set_style('white')
sns.countplot(x='Age',data=Master_data,palette='terrain')
plt.title('User Age Distribution',fontdict=txt)
plt.xlabel('Age(range)',fontdict=txt,labelpad=15)
plt.ylabel('Number of users',fontdict=txt,labelpad=25)
plt.xticks([0,1,2,3,4,5,6],['Under 18','18-24','25-34','35-44','45-49','50-55','56+'])
for i in range(7):
    plt.annotate(age_dist['Age'].iloc[i],xy=(i-0.2,age_dist['Age'].iloc[i]+5000),size=15,)
plt.show()
```



Most of the users are between the age is 25 to 34

2. User Rating for movies Toy Story

```
# Finding User rating for the movie Toy Story
Toy_story_df=Master_data[Master_data['Title']=='Toy Story (1995)']
```

```
TS_rating=Toy_story_df['Rating'].value_counts().to_frame()
TS_rating
```

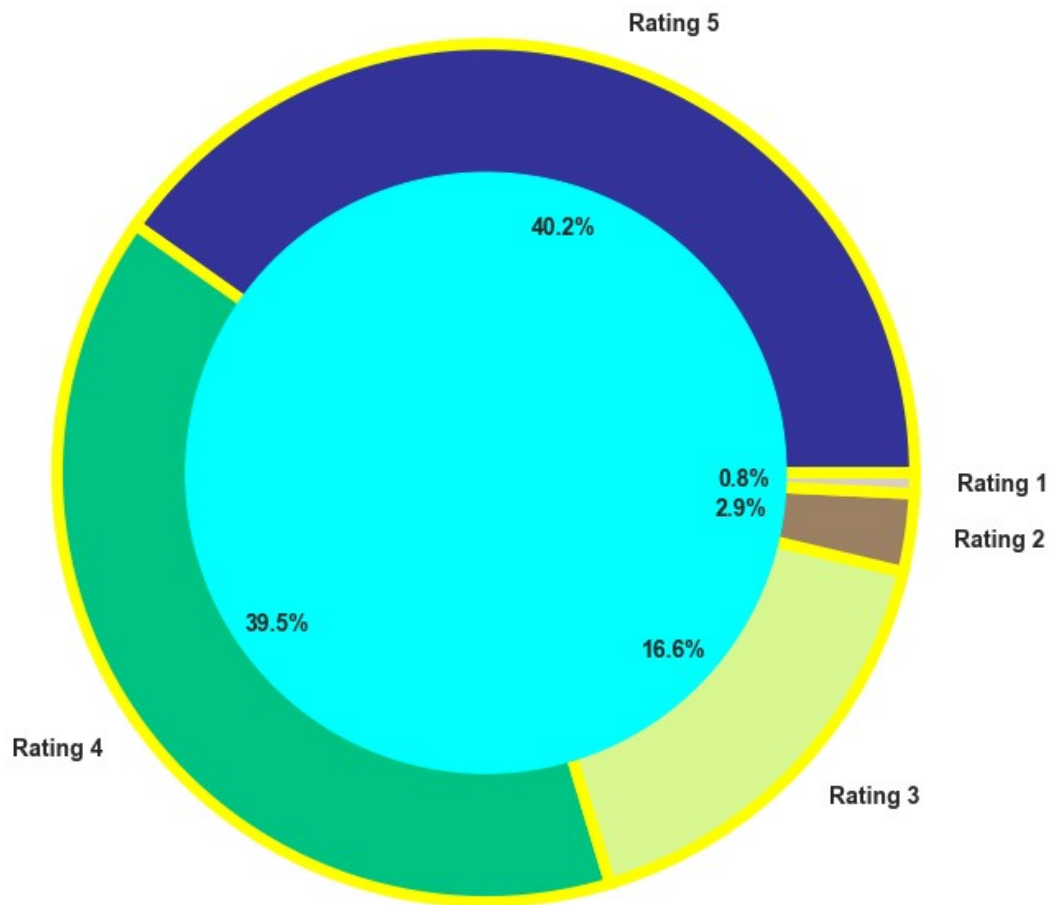
```
   Rating
4      835
5      820
3      345
2       61
1       16
```

```
TS_rating=list(Toy_story_df['Rating'].value_counts())
TS_rating
```

```
[835, 820, 345, 61, 16]
```

```
# Visual Representation showing the percentage of each rating for the
movie Toy Story
```

```
plt.figure(figsize=(12,12))
txt={'weight':'bold','size':14}
cmap=plt.get_cmap('terrain')
colors =cmap(np.arange(5)*59)
labels =['Rating 5','Rating 4','Rating 3','Rating 2','Rating 1']
plt.pie(TS_rating,labels=labels,wedgeprops={'linewidth':7,'edgecolor':
'yellow'},autopct='%1.1f%%',textprops=txt,colors=colors)
my_circle=plt.Circle((0,0),0.7,color='cyan')
p=plt.gcf()
p.gca().add_artist(my_circle)
plt.show()
```



```
# Finding the average overall rating
print("Agetage Rating for the movie Toy Story is
{:.2f}".format(Toy_story_df['Rating'].mean()))
```

Agetage Rating for the movie Toy Story is 4.15

The movie Toy Story was 79.7% rated good (4 and 5) and its average rating is 4.15.

3. Top 25 Movies by Viewership Rating

```
Top_25_movies=pd.DataFrame(Master_data.groupby('Title')
['Rating'].agg('mean')).sort_values(by='Rating',ascending=False).head(
25)
Top_25_movies['Title']= Top_25_movies.index
```



```
Top_25_movies.index=range(1,26)
Top_25_movies[['Title','Rating']]
```

	Title	Rating
1	Ulysses (Ulisse) (1954)	5.000000
2	Lured (1947)	5.000000
3	Follow the Bitch (1998)	5.000000
4	Bittersweet Motel (2000)	5.000000
5	Song of Freedom (1936)	5.000000
6	One Little Indian (1973)	5.000000
7	Smashing Time (1967)	5.000000
8	Schlafes Bruder (Brother of Sleep) (1995)	5.000000
9	Gate of Heavenly Peace, The (1995)	5.000000
10	Baby, The (1973)	5.000000
11	I Am Cuba (Soy Cuba/Ya Kuba) (1964)	4.800000
12	Lamerica (1994)	4.750000
13	Apple, The (Sib) (1998)	4.666667
14	Sanjuro (1962)	4.608696
15	Seven Samurai (The Magnificent Seven) (Shichin...	4.560510
16	Shawshank Redemption, The (1994)	4.554558
17	Godfather, The (1972)	4.524966
18	Close Shave, A (1995)	4.520548
19	Usual Suspects, The (1995)	4.517106
20	Schindler's List (1993)	4.510417
21	Wrong Trousers, The (1993)	4.507937
22	Dry Cleaning (Nettoyage à sec) (1997)	4.500000
23	Inheritors, The (Die Siebtelbauern) (1998)	4.500000
24	Mamma Roma (1962)	4.500000
25	Bells, The (1926)	4.500000

Out of top 25 movies first 10 movies has been rated 5

User ID = 2696 Rated movies

Extracting the deatails of User ID 2696

```
Userid_2696=Master_data[Master_data['UserID']==2696]
```

```
Userid_2696=Userid_2696.sort_values('Rating',ascending=False,ignore_in  
dex=True)
```

```
Userid_2696[['MovieID','Title','Rating']]
```

	MovieID	Title	Rating
0	800	Lone Star (1996)	5
1	1645	Devil's Advocate, The (1997)	4
2	1783	Palmetto (1998)	4
3	1092	Basic Instinct (1992)	4
4	3176	Talented Mr. Ripley, The (1999)	4
5	1258	Shining, The (1980)	4
6	2389	Psycho (1998)	4
7	1892	Perfect Murder, A (1998)	4

8	1617	L.A. Confidential (1997)	4
9	1625	Game, The (1997)	4
10	1805	Wild Things (1998)	4
11	1711	Midnight in the Garden of Good and Evil (1997)	4
12	350	Client, The (1994)	3
13	1589	Cop Land (1997)	3
14	1097	E.T. the Extra-Terrestrial (1982)	3
15	1644	I Know What You Did Last Summer (1997)	2
16	2338	I Still Know What You Did Last Summer (1998)	2
17	1270	Back to the Future (1985)	2
18	2713	Lake Placid (1999)	1
19	3386	JFK (1991)	1

Result: This user has rated 5 to the movie title 'Lone Star (1996)' and Most of his ratings are 4

Feature Engineering:

Use column genres:

1.Find out all the unique genres (Hint: split the data in column genre making a list and then process the data to find out only the unique categories of genres)

2.Create a separate column for each genre category with a one-hot encoding (1 and 0) whether or not the movie belongs to that genre.

3.Determine the features affecting the ratings of any particular movie.

4.Develop an appropriate model to predict the movie ratings

storing Genres column to a variable names genre

```
genre=df_final['Genres']
genre.head()
```

```
0      Animation|Children's|Comedy
1      Animation|Children's|Musical|Romance
2      Drama
3      Action|Adventure|Fantasy|Sci-Fi
4      Drama|War
```

Name: Genres, dtype: object

For loop to store the splitted genre date and append to a list

```
list1=[]
for row in genre:
    row_list=row.split("|")
```

```
for i in range (len(row_list)):
    list1.append(row_list)
```

```
list1
```

```
[['Animation', 'Children's', 'Comedy'],
 ['Animation', 'Children's', 'Comedy'],
 ['Animation', 'Children's', 'Comedy'],
 ['Animation', 'Children's', 'Musical', 'Romance'],
 ['Animation', 'Children's', 'Musical', 'Romance'],
 ['Animation', 'Children's', 'Musical', 'Romance'],
 ['Animation', 'Children's', 'Musical', 'Romance'],
 ['Drama'],
 ['Action', 'Adventure', 'Fantasy', 'Sci-Fi'],
 ['Action', 'Adventure', 'Fantasy', 'Sci-Fi'],
 ['Action', 'Adventure', 'Fantasy', 'Sci-Fi'],
 ['Action', 'Adventure', 'Fantasy', 'Sci-Fi'],
 ['Drama', 'War'],
 ['Drama', 'War'],
 ["Children's", 'Drama'],
 ["Children's", 'Drama'],
 ['Animation', 'Children's', 'Comedy', 'Musical'],
 ['Animation', 'Children's', 'Comedy', 'Musical'],
 ['Animation', 'Children's', 'Comedy', 'Musical'],
 ['Animation', 'Children's', 'Comedy', 'Musical'],
 ['Animation', 'Children's', 'Musical'],
 ['Animation', 'Children's', 'Musical'],
 ['Animation', 'Children's', 'Musical'],
 ['Animation', 'Children's', 'Musical'],
 ['Animation', 'Children's', 'Musical'],
 ['Animation', 'Children's', 'Musical'],
 ['Crime', 'Drama', 'Thriller'],
 ['Crime', 'Drama', 'Thriller'],
 ['Crime', 'Drama', 'Thriller'],
 ['Animation', 'Children's', 'Musical'],
 ['Animation', 'Children's', 'Musical'],
 ['Animation', 'Children's', 'Musical'],
 ['Animation'],
 ['Animation', 'Comedy', 'Thriller'],
 ['Animation', 'Comedy', 'Thriller'],
 ['Animation', 'Comedy', 'Thriller'],
 ['Animation', 'Children's', 'Musical'],
 ['Animation', 'Children's', 'Musical'],
 ['Animation', 'Children's', 'Musical'],
 ['Musical', 'Romance'],
 ['Musical', 'Romance'],
 ['Adventure', 'Children's', 'Drama', 'Musical'],
 ['Adventure', 'Children's', 'Drama', 'Musical'],
 ['Adventure', 'Children's', 'Drama', 'Musical'],
 ['Adventure', 'Children's', 'Drama', 'Musical'],
 ['Musical'],
```

```
['Animation', 'Children's', 'Musical'],
['Animation', 'Children's', 'Musical'],
['Animation', 'Children's', 'Musical'],
['Children's', 'Comedy', 'Musical'],
['Children's', 'Comedy', 'Musical'],
['Children's', 'Comedy', 'Musical'],
['Animation', 'Children's', 'Musical'],
['Animation', 'Children's', 'Musical'],
['Animation', 'Children's', 'Musical'],
['Musical'],
['Children's', 'Drama', 'Fantasy', 'Sci-Fi'],
['Children's', 'Drama', 'Fantasy', 'Sci-Fi'],
['Children's', 'Drama', 'Fantasy', 'Sci-Fi'],
['Children's', 'Drama', 'Fantasy', 'Sci-Fi'],
['Drama'],
['Action', 'Adventure', 'Comedy', 'Romance'],
['Action', 'Adventure', 'Comedy', 'Romance'],
['Action', 'Adventure', 'Comedy', 'Romance'],
['Action', 'Adventure', 'Comedy', 'Romance'],
['Drama'],
['Drama'],
['Comedy', 'Sci-Fi'],
['Comedy', 'Sci-Fi'],
['Action', 'Adventure', 'Drama'],
['Action', 'Adventure', 'Drama'],
['Action', 'Adventure', 'Drama'],
['Drama'],
['Adventure', 'Animation', 'Children's', 'Comedy', 'Musical'],
['Adventure', 'Animation', 'Children's', 'Comedy', 'Musical'],
['Adventure', 'Animation', 'Children's', 'Comedy', 'Musical'],
['Adventure', 'Animation', 'Children's', 'Comedy', 'Musical'],
['Adventure', 'Animation', 'Children's', 'Comedy', 'Musical'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Drama'],
['Animation', 'Children's'],
['Animation', 'Children's'],
['Drama'],
['Drama'],
['Animation', 'Children's'],
['Animation', 'Children's'],
['Action', 'Drama', 'War'],
['Action', 'Drama', 'War'],
['Action', 'Drama', 'War'],
['Animation', 'Children's'],
['Animation', 'Children's'],
['Comedy'],
['Romance'],
['Animation', 'Children's', 'Comedy'],
['Animation', 'Children's', 'Comedy'],
```

```
['Animation', 'Children's', 'Comedy'],
['Drama'],
['Animation', 'Children's'],
['Animation', 'Children's'],
['Action', 'Crime', 'Romance'],
['Action', 'Crime', 'Romance'],
['Action', 'Crime', 'Romance'],
['Thriller'],
['Comedy'],
['Comedy', 'Fantasy'],
['Comedy', 'Fantasy'],
['Comedy', 'Drama'],
['Comedy', 'Drama'],
['Comedy'],
['Drama'],
['Animation', 'Children's', 'Comedy'],
['Animation', 'Children's', 'Comedy'],
['Animation', 'Children's', 'Comedy'],
['Drama'],
['Drama'],
['Animation', 'Children's', 'Comedy'],
['Animation', 'Children's', 'Comedy'],
['Animation', 'Children's', 'Comedy'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Children's', 'Comedy', 'Drama'],
['Children's', 'Comedy', 'Drama'],
['Children's', 'Comedy', 'Drama'],
['Animation', 'Children's', 'Musical', 'Romance'],
['Animation', 'Children's', 'Musical', 'Romance'],
['Animation', 'Children's', 'Musical', 'Romance'],
['Animation', 'Children's', 'Musical', 'Romance'],
['Drama', 'Musical'],
['Drama', 'Musical'],
['Drama', 'Romance', 'War', 'Western'],
['Drama', 'Romance', 'War', 'Western'],
['Drama', 'Romance', 'War', 'Western'],
['Drama', 'Romance', 'War', 'Western'],
['Crime', 'Drama'],
['Crime', 'Drama'],
['Animation', 'Children's', 'Musical'],
['Animation', 'Children's', 'Musical'],
['Animation', 'Children's', 'Musical'],
['Action', 'Comedy', 'Western'],
['Action', 'Comedy', 'Western'],
['Action', 'Comedy', 'Western'],
['Action', 'Romance', 'Thriller'],
['Action', 'Romance', 'Thriller'],
['Action', 'Romance', 'Thriller'],
['Western'],
```

```
["Children's", 'Comedy'],
["Children's", 'Comedy'],
['Animation', "Children's", 'Comedy', 'Musical'],
['Animation', "Children's", 'Comedy', 'Musical'],
['Animation', "Children's", 'Comedy', 'Musical'],
['Animation', "Children's", 'Comedy', 'Musical'],
['Adventure', 'Drama', 'Western'],
['Adventure', 'Drama', 'Western'],
['Adventure', 'Drama', 'Western'],
['Animation', "Children's", 'Musical'],
['Animation', "Children's", 'Musical'],
['Animation', "Children's", 'Musical'],
['Comedy', 'Romance'],
['Comedy', 'Romance'],
['Comedy', 'Drama', 'Romance'],
['Comedy', 'Drama', 'Romance'],
['Comedy', 'Drama', 'Romance'],
['Drama', 'Romance', 'War'],
['Drama', 'Romance', 'War'],
['Drama', 'Romance', 'War'],
['Musical', 'Romance'],
['Musical', 'Romance'],
['Drama', 'Romance', 'War'],
['Drama', 'Romance', 'War'],
['Drama', 'Romance', 'War'],
["Children's", 'Comedy', 'Western'],
["Children's", 'Comedy', 'Western'],
["Children's", 'Comedy', 'Western'],
["Children's", 'Comedy', 'Musical'],
["Children's", 'Comedy', 'Musical'],
["Children's", 'Comedy', 'Musical'],
['Adventure', 'Animation', "Children's", 'Musical'],
['Adventure', 'Animation', "Children's", 'Musical'],
['Adventure', 'Animation', "Children's", 'Musical'],
['Adventure', 'Animation', "Children's", 'Musical'],
['Musical'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Musical', 'Romance'],
['Musical', 'Romance'],
['Action', 'Romance'],
['Action', 'Romance'],
['Comedy', 'Romance'],
['Comedy', 'Romance'],
['Action', 'Adventure', 'Romance', 'Sci-Fi', 'War'],
['Action', 'Adventure', 'Romance', 'Sci-Fi', 'War'],
['Action', 'Adventure', 'Romance', 'Sci-Fi', 'War'],
['Action', 'Adventure', 'Romance', 'Sci-Fi', 'War'],
['Action', 'Adventure', 'Romance', 'Sci-Fi', 'War'],
['Drama', 'Romance'],
```

```
['Drama', 'Romance'],
['Comedy', 'Musical', 'Romance'],
['Comedy', 'Musical', 'Romance'],
['Comedy', 'Musical', 'Romance'],
['Comedy', 'Romance'],
['Comedy', 'Romance'],
['Adventure', 'Animation', 'Children's', 'Comedy', 'Musical'],
['Adventure', 'Animation', 'Children's', 'Comedy', 'Musical'],
['Adventure', 'Animation', 'Children's', 'Comedy', 'Musical'],
['Adventure', 'Animation', 'Children's', 'Comedy', 'Musical'],
['Adventure', 'Animation', 'Children's', 'Comedy', 'Musical'],
['Comedy', 'Romance'],
['Comedy', 'Romance'],
['Drama', 'Romance', 'Thriller'],
['Drama', 'Romance', 'Thriller'],
['Drama', 'Romance', 'Thriller'],
['Animation', 'Children's', 'Musical'],
['Animation', 'Children's', 'Musical'],
['Animation', 'Children's', 'Musical'],
['Adventure', 'Children's', 'Comedy'],
['Adventure', 'Children's', 'Comedy'],
['Adventure', 'Children's', 'Comedy'],
['Musical', 'Romance'],
['Musical', 'Romance'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Action', 'Adventure', 'Romance'],
['Action', 'Adventure', 'Romance'],
['Action', 'Adventure', 'Romance'],
['Children's', 'Fantasy', 'Musical'],
['Children's', 'Fantasy', 'Musical'],
['Children's', 'Fantasy', 'Musical'],
['Animation', 'Children's', 'Comedy', 'Musical', 'Romance'],
['Animation', 'Children's', 'Comedy', 'Musical', 'Romance'],
['Animation', 'Children's', 'Comedy', 'Musical', 'Romance'],
['Animation', 'Children's', 'Comedy', 'Musical', 'Romance'],
['Animation', 'Children's', 'Comedy', 'Musical', 'Romance'],
['Children's', 'Comedy'],
['Children's', 'Comedy'],
['Comedy', 'Fantasy', 'Romance'],
['Comedy', 'Fantasy', 'Romance'],
['Comedy', 'Fantasy', 'Romance'],
['Comedy'],
['Comedy', 'Romance'],
['Comedy', 'Romance'],
['Action', 'Adventure', 'Comedy', 'Romance'],
['Action', 'Adventure', 'Comedy', 'Romance'],
['Action', 'Adventure', 'Comedy', 'Romance'],
['Action', 'Adventure', 'Comedy', 'Romance'],
['Comedy', 'Romance'],
```

```
['Comedy', 'Romance'],
['Comedy', 'Drama', 'Romance'],
['Comedy', 'Drama', 'Romance'],
['Comedy', 'Drama', 'Romance'],
['Action', 'Romance', 'Thriller'],
['Action', 'Romance', 'Thriller'],
['Action', 'Romance', 'Thriller'],
['Comedy', 'Drama'],
['Comedy', 'Drama'],
['Drama'],
['Comedy'],
['Western'],
['Drama'],
['Comedy', 'Romance'],
['Comedy', 'Romance'],
['Western'],
['Comedy', 'Romance'],
['Comedy', 'Romance'],
['Comedy'],
['Comedy', 'Romance'],
['Comedy', 'Romance'],
['Comedy', 'Drama'],
['Comedy', 'Drama'],
['Action', 'Drama'],
['Action', 'Drama'],
['Comedy', 'Musical'],
['Comedy', 'Musical'],
['Musical'],
['Musical'],
['Action'],
['Western'],
['Comedy', 'Drama', 'Romance'],
['Comedy', 'Drama', 'Romance'],
['Comedy', 'Drama', 'Romance'],
['Adventure', 'Drama', 'Romance', 'Sci-Fi'],
['Adventure', 'Drama', 'Romance', 'Sci-Fi'],
['Adventure', 'Drama', 'Romance', 'Sci-Fi'],
['Adventure', 'Drama', 'Romance', 'Sci-Fi'],
['Action', 'Crime'],
['Action', 'Crime'],
['Action', 'Drama', 'War'],
['Action', 'Drama', 'War'],
['Action', 'Drama', 'War'],
['Animation', "Children's", 'Comedy'],
['Animation', "Children's", 'Comedy'],
['Animation', "Children's", 'Comedy'],
['Comedy', 'Drama'],
['Comedy', 'Drama'],
['Drama'],
['Drama', 'Thriller'],
```


['Drama', 'Thriller'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Drama', 'Sci-Fi'],
['Drama', 'Sci-Fi'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Drama'],
['Comedy', 'Romance'],
['Comedy', 'Romance'],
['Action', 'Crime', 'Drama'],
['Action', 'Crime', 'Drama'],
['Action', 'Crime', 'Drama'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Drama', 'Musical'],
['Drama', 'Musical'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Action', 'Drama', 'War'],
['Action', 'Drama', 'War'],
['Action', 'Drama', 'War'],
['Drama', 'Thriller'],
['Drama', 'Thriller'],
['Drama'],
['Drama', 'Romance', 'War'],
['Drama', 'Romance', 'War'],
['Drama', 'Romance', 'War'],
['Drama', 'Thriller', 'War'],
['Drama', 'Thriller', 'War'],
['Drama', 'Thriller', 'War'],
['Action', 'Romance', 'Thriller'],
['Action', 'Romance', 'Thriller'],
['Action', 'Romance', 'Thriller'],
['Drama', 'Thriller'],
['Drama', 'Thriller'],
['Drama', 'Horror'],
['Drama', 'Horror'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Drama', 'Romance', 'War', 'Western'],
['Drama', 'Romance', 'War', 'Western'],
['Drama', 'Romance', 'War', 'Western'],
['Drama', 'Romance', 'War', 'Western'],
['Drama'],
['Drama'],
['Action', 'Thriller'],
['Action', 'Thriller'],
['Crime', 'Drama'],
['Crime', 'Drama'],

```
['Drama'],
['Comedy', 'Drama'],
['Comedy', 'Drama'],
['Action', 'Adventure', 'Thriller'],
['Action', 'Adventure', 'Thriller'],
['Action', 'Adventure', 'Thriller'],
['Action', 'Romance', 'Thriller'],
['Action', 'Romance', 'Thriller'],
['Action', 'Romance', 'Thriller'],
['Action'],
['Drama', 'Thriller'],
['Drama', 'Thriller'],
['Action', 'Drama', 'War'],
['Action', 'Drama', 'War'],
['Action', 'Drama', 'War'],
['Comedy', 'Drama'],
['Comedy', 'Drama'],
['Action', 'Adventure', 'Sci-Fi'],
['Action', 'Adventure', 'Sci-Fi'],
['Action', 'Adventure', 'Sci-Fi'],
['Drama'],
['Drama'],
['Drama'],
['Drama'],
['Drama', 'War'],
['Drama', 'War'],
['Drama'],
['Action', 'Crime', 'Romance'],
['Action', 'Crime', 'Romance'],
['Action', 'Crime', 'Romance'],
['Comedy', 'Drama'],
['Comedy', 'Drama'],
['Action', 'Sci-Fi', 'Thriller'],
['Action', 'Sci-Fi', 'Thriller'],
['Action', 'Sci-Fi', 'Thriller'],
['Crime', 'Drama', 'Thriller'],
['Crime', 'Drama', 'Thriller'],
['Crime', 'Drama', 'Thriller'],
['Drama'],
['Action', 'Adventure', 'Thriller'],
['Action', 'Adventure', 'Thriller'],
['Action', 'Adventure', 'Thriller'],
['Animation', 'Sci-Fi'],
['Animation', 'Sci-Fi'],
['Drama', 'Thriller'],
['Drama', 'Thriller'],
['Drama'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Drama'],
```

['Action', 'Adventure', 'Romance', 'Sci-Fi', 'War'],
['Action', 'Adventure', 'Romance', 'Sci-Fi', 'War'],
['Action', 'Adventure', 'Romance', 'Sci-Fi', 'War'],
['Action', 'Adventure', 'Romance', 'Sci-Fi', 'War'],
['Action', 'Adventure', 'Romance', 'Sci-Fi', 'War'],
['Crime', 'Drama'],
['Crime', 'Drama'],
['Comedy', 'Romance'],
['Comedy', 'Romance'],
['Adventure', 'Animation', 'Sci-Fi', 'Thriller'],
['Adventure', 'Animation', 'Sci-Fi', 'Thriller'],
['Adventure', 'Animation', 'Sci-Fi', 'Thriller'],
['Adventure', 'Animation', 'Sci-Fi', 'Thriller'],
['Action', 'Drama', 'Romance'],
['Action', 'Drama', 'Romance'],
['Action', 'Drama', 'Romance'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Drama'],
['Crime', 'Drama'],
['Crime', 'Drama'],
['Comedy', 'Drama'],
['Comedy', 'Drama'],
['Action', 'Drama', 'Thriller', 'War'],
['Action', 'Drama', 'Thriller', 'War'],
['Action', 'Drama', 'Thriller', 'War'],
['Action', 'Drama', 'Thriller', 'War'],
['Action', 'Sci-Fi', 'Thriller'],
['Action', 'Sci-Fi', 'Thriller'],
['Action', 'Sci-Fi', 'Thriller'],
['Action', 'Adventure', 'Comedy', 'Sci-Fi'],
['Action', 'Adventure', 'Comedy', 'Sci-Fi'],
['Action', 'Adventure', 'Comedy', 'Sci-Fi'],
['Action', 'Adventure', 'Comedy', 'Sci-Fi'],
['Crime', 'Drama', 'Mystery'],
['Crime', 'Drama', 'Mystery'],
['Crime', 'Drama', 'Mystery'],
['Drama'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Drama', 'Sci-Fi', 'Thriller'],
['Drama', 'Sci-Fi', 'Thriller'],
['Drama', 'Sci-Fi', 'Thriller'],
['Drama'],
['Drama'],
['Drama'],
['Drama'],
['Drama'],

```
['Comedy', 'Drama'],
['Comedy', 'Drama'],
['Drama'],
['Comedy', 'Crime', 'Drama', 'Mystery'],
['Comedy', 'Crime', 'Drama', 'Mystery'],
['Comedy', 'Crime', 'Drama', 'Mystery'],
['Comedy', 'Crime', 'Drama', 'Mystery'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Drama'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Action', 'Drama', 'Romance'],
['Action', 'Drama', 'Romance'],
['Action', 'Drama', 'Romance'],
['Drama'],
['Drama'],
['Drama'],
['Action', 'Comedy', 'Drama'],
['Action', 'Comedy', 'Drama'],
['Action', 'Comedy', 'Drama'],
['Action', 'Adventure', 'Romance'],
['Action', 'Adventure', 'Romance'],
['Action', 'Adventure', 'Romance'],
['Action', 'Crime', 'Drama'],
['Action', 'Crime', 'Drama'],
['Action', 'Crime', 'Drama'],
['Action', 'Drama', 'War'],
['Action', 'Drama', 'War'],
['Action', 'Drama', 'War'],
['Crime', 'Drama'],
['Crime', 'Drama'],
['Action', 'Crime', 'Thriller'],
['Action', 'Crime', 'Thriller'],
['Action', 'Crime', 'Thriller'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Drama'],
['Drama', 'Thriller'],
['Drama', 'Thriller'],
['Comedy', 'Drama'],
['Comedy', 'Drama'],
['Drama'],
['Drama'],
['Comedy', 'Romance'],
['Comedy', 'Romance'],
['Action', 'Drama', 'War'],
['Action', 'Drama', 'War'],
```

```
['Action', 'Drama', 'War'],
['Adventure', "Children's", 'Drama'],
['Adventure', "Children's", 'Drama'],
['Adventure', "Children's", 'Drama'],
['Drama'],
['Action', 'Thriller'],
['Action', 'Thriller'],
['Drama'],
['Action', 'Sci-Fi', 'Thriller'],
['Action', 'Sci-Fi', 'Thriller'],
['Action', 'Sci-Fi', 'Thriller'],
['Action', 'Sci-Fi', 'Thriller'],
['Action', 'Sci-Fi', 'Thriller'],
['Action', 'Sci-Fi', 'Thriller'],
['Drama', 'Mystery'],
['Drama', 'Mystery'],
['Drama', 'Thriller'],
['Drama', 'Thriller'],
['Action', 'Crime', 'Romance'],
['Action', 'Crime', 'Romance'],
['Action', 'Crime', 'Romance'],
['Action', 'Comedy', 'Sci-Fi', 'Thriller'],
['Action', 'Comedy', 'Sci-Fi', 'Thriller'],
['Action', 'Comedy', 'Sci-Fi', 'Thriller'],
['Action', 'Comedy', 'Sci-Fi', 'Thriller'],
['Drama'],
['Drama'],
['Comedy', 'Drama'],
['Comedy', 'Drama'],
['Drama'],
['Action', 'Adventure', 'Sci-Fi', 'Thriller'],
['Action', 'Adventure', 'Sci-Fi', 'Thriller'],
['Action', 'Adventure', 'Sci-Fi', 'Thriller'],
['Action', 'Adventure', 'Sci-Fi', 'Thriller'],
['Drama'],
['Drama'],
['Action', 'Drama'],
['Action', 'Drama'],
['Drama', 'Thriller'],
['Drama', 'Thriller'],
['Drama'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Drama'],
['Drama'],
['Animation', "Children's"],
['Animation', "Children's"],
['Drama'],
['Drama'],
['Drama'],
```

```
['Action', 'Thriller'],
['Action', 'Thriller'],
['Action', 'Drama', 'Romance', 'Thriller'],
['Action', 'Drama', 'Romance', 'Thriller'],
['Action', 'Drama', 'Romance', 'Thriller'],
['Action', 'Drama', 'Romance', 'Thriller'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Drama'],
['Action', 'Crime'],
['Action', 'Crime'],
['Action', 'Thriller'],
['Action', 'Thriller'],
['Drama', 'Mystery'],
['Drama', 'Mystery'],
['Action', 'Drama'],
['Action', 'Drama'],
['Drama'],
['Comedy'],
['Comedy', 'Drama'],
['Comedy', 'Drama'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Animation', "Children's", 'Comedy'],
['Animation', "Children's", 'Comedy'],
['Animation', "Children's", 'Comedy'],
['Drama', 'Thriller'],
['Drama', 'Thriller'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Crime', 'Thriller'],
['Crime', 'Thriller'],
['Crime', 'Thriller'],
['Crime', 'Thriller'],
['Drama'],
['Documentary'],
['Comedy'],
['Drama'],
['Drama'],
['Action', 'Adventure', 'Thriller'],
['Action', 'Adventure', 'Thriller'],
['Action', 'Adventure', 'Thriller'],
['Comedy', 'Crime', 'Fantasy'],
['Comedy', 'Crime', 'Fantasy'],
['Comedy', 'Crime', 'Fantasy'],
['Action', 'Romance', 'Thriller'],
['Action', 'Romance', 'Thriller'],
['Action', 'Romance', 'Thriller'],
['Drama'],
['Drama'],
```

```
['Action', 'Thriller'],
['Action', 'Thriller'],
['Action', 'Adventure', 'Sci-Fi'],
['Action', 'Adventure', 'Sci-Fi'],
['Action', 'Adventure', 'Sci-Fi'],
['Drama'],
['Drama'],
['Drama', 'War'],
['Drama', 'War'],
['Drama'],
['Adventure', 'Drama', 'Western'],
['Adventure', 'Drama', 'Western'],
['Adventure', 'Drama', 'Western'],
['Drama', 'Thriller'],
['Drama', 'Thriller'],
['Comedy', 'Romance'],
['Comedy', 'Romance'],
['Crime', 'Drama', 'Thriller'],
['Crime', 'Drama', 'Thriller'],
['Crime', 'Drama', 'Thriller'],
['Animation'],
['Animation', 'Comedy', 'Thriller'],
['Animation', 'Comedy', 'Thriller'],
['Animation', 'Comedy', 'Thriller'],
['Drama'],
['Drama'],
['Comedy', 'Drama', 'Romance'],
['Comedy', 'Drama', 'Romance'],
['Comedy', 'Drama', 'Romance'],
['Action', 'Thriller'],
['Action', 'Thriller'],
['Drama', 'Romance', 'War'],
['Drama', 'Romance', 'War'],
['Drama', 'Romance', 'War'],
['Drama', 'Romance', 'War'],
['Drama', 'Romance', 'War'],
['Drama', 'Romance', 'War'],
['Drama'],
['Comedy', 'Drama'],
['Comedy', 'Drama'],
['Crime', 'Thriller'],
['Crime', 'Thriller'],
['Animation', 'Comedy'],
['Animation', 'Comedy'],
['Action', 'Adventure', 'Romance', 'Sci-Fi', 'War'],
['Action', 'Adventure', 'Romance', 'Sci-Fi', 'War'],
['Action', 'Adventure', 'Romance', 'Sci-Fi', 'War'],
['Action', 'Adventure', 'Romance', 'Sci-Fi', 'War'],
['Action', 'Adventure', 'Romance', 'Sci-Fi', 'War'],
['Crime', 'Drama'],
```

['Crime', 'Drama'],
['Action', 'Crime', 'Drama'],
['Action', 'Crime', 'Drama'],
['Action', 'Crime', 'Drama'],
['Animation', 'Comedy'],
['Animation', 'Comedy'],
['Action', 'Drama', 'War'],
['Action', 'Drama', 'War'],
['Action', 'Drama', 'War'],
['Comedy', 'Romance'],
['Comedy', 'Romance'],
['Comedy', 'Romance'],
['Comedy', 'Romance'],
['Documentary'],
['Thriller'],
['Action', 'Adventure', 'Sci-Fi'],
['Action', 'Adventure', 'Sci-Fi'],
['Action', 'Adventure', 'Sci-Fi'],
['Drama', 'Thriller'],
['Drama', 'Thriller'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Comedy'],
['Crime', 'Drama'],
['Crime', 'Drama'],
['Comedy', 'Crime'],
['Comedy', 'Crime'],
['Action', 'Adventure', 'Thriller'],
['Action', 'Adventure', 'Thriller'],
['Action', 'Adventure', 'Thriller'],
['Drama', 'Sci-Fi'],
['Drama', 'Sci-Fi'],
['Crime', 'Film-Noir', 'Mystery', 'Thriller'],
['Crime', 'Film-Noir', 'Mystery', 'Thriller'],
['Crime', 'Film-Noir', 'Mystery', 'Thriller'],
['Crime', 'Film-Noir', 'Mystery', 'Thriller'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Drama', 'Sci-Fi', 'Thriller'],
['Drama', 'Sci-Fi', 'Thriller'],
['Drama', 'Sci-Fi', 'Thriller'],
['Romance'],
['Drama'],
['Drama'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Comedy', 'Romance'],
['Comedy', 'Romance'],
['Comedy', 'Drama'],
['Comedy', 'Drama'],


```
['Action', 'Crime', 'Romance'],
['Action', 'Crime', 'Romance'],
['Action', 'Crime', 'Romance'],
['Sci-Fi', 'Thriller'],
['Sci-Fi', 'Thriller'],
['Comedy'],
['Drama'],
['Action', 'Drama', 'War'],
['Action', 'Drama', 'War'],
['Action', 'Drama', 'War'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Crime', 'Drama'],
['Crime', 'Drama'],
['Action', 'Crime', 'Thriller'],
['Action', 'Crime', 'Thriller'],
['Action', 'Crime', 'Thriller'],
['Animation', "Children's"],
['Animation', "Children's"],
['Comedy'],
['Comedy', 'Drama'],
['Comedy', 'Drama'],
['Animation', "Children's", 'Comedy'],
['Animation', "Children's", 'Comedy'],
['Animation', "Children's", 'Comedy'],
['Action', 'Sci-Fi', 'Thriller'],
['Action', 'Sci-Fi', 'Thriller'],
['Action', 'Sci-Fi', 'Thriller'],
['Comedy'],
['Action', 'Crime', 'Romance'],
['Action', 'Crime', 'Romance'],
['Action', 'Crime', 'Romance'],
['Thriller'],
['Comedy', 'Drama'],
['Comedy', 'Drama'],
['Drama', 'War'],
['Drama', 'War'],
['Drama'],
['Drama'],
['Animation', "Children's", 'Comedy'],
['Animation', "Children's", 'Comedy'],
['Animation', "Children's", 'Comedy'],
['Drama', 'Thriller'],
['Drama', 'Thriller'],
['Drama'],
['Drama'],
['Drama'],
['Comedy'],
['Comedy', 'Drama'],
['Comedy', 'Drama'],
```

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

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

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

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

```
['Comedy', 'Romance'],
['Musical'],
['Adventure', "Children's", 'Drama', 'Musical'],
['Adventure', "Children's", 'Drama', 'Musical'],
['Adventure', "Children's", 'Drama', 'Musical'],
['Adventure', "Children's", 'Drama', 'Musical'],
['Drama', 'Romance', 'War'],
['Drama', 'Romance', 'War'],
['Drama', 'Romance', 'War'],
['Drama'],
['Drama', 'Mystery', 'Sci-Fi', 'Thriller'],
['Drama', 'Mystery', 'Sci-Fi', 'Thriller'],
['Drama', 'Mystery', 'Sci-Fi', 'Thriller'],
['Drama', 'Mystery', 'Sci-Fi', 'Thriller'],
['Drama'],
['Romance'],
['Musical'],
['Drama', 'Romance'],
['Drama', 'Romance'],
['Drama'],
['Drama'],
['Musical'],
['Action', 'Adventure', 'Romance', 'War'],
['Action', 'Adventure', 'Romance', 'War'],
['Action', 'Adventure', 'Romance', 'War'],
['Action', 'Adventure', 'Romance', 'War'],
['Drama'],
['Adventure', "Children's", 'Fantasy'],
['Adventure', "Children's", 'Fantasy'],
['Adventure', "Children's", 'Fantasy'],
["Children's", 'Drama'],
["Children's", 'Drama'],
['Adventure', "Children's"],
['Adventure', "Children's"],
["Children's", 'Comedy'],
["Children's", 'Comedy'],
['Adventure', "Children's", 'Fantasy', 'Sci-Fi'],
['Adventure', "Children's", 'Fantasy', 'Sci-Fi'],
['Adventure', "Children's", 'Fantasy', 'Sci-Fi'],
['Adventure', "Children's", 'Fantasy', 'Sci-Fi'],
['Comedy'],
['Animation', "Children's", 'Musical'],
['Animation', "Children's", 'Musical'],
['Animation', "Children's", 'Musical'],
['Animation', "Children's"],
['Animation', "Children's"],
['Animation', "Children's"],
['Animation', "Children's"],
["Children's", 'Comedy', 'Musical'],
["Children's", 'Comedy', 'Musical'],
```

```

["Children's", 'Comedy', 'Musical'],
['Adventure', 'Animation', "Children's", 'Musical'],
['Adventure', 'Animation', "Children's", 'Musical'],
['Adventure', 'Animation', "Children's", 'Musical'],
...]

# unique_genre = list(set(list1))
# print(len(list1))
# if i in list

# l=set(unique_genre)

# creating a new variable onehotcode to perform onehotencoding
onehotcode=df_final.copy()
onehotcode.head(3)

```

	MovieID	Title	Genres
UserID \			
0	1	Toy Story (1995)	Animation Children's Comedy
1			
1	48	Pocahontas (1995)	Animation Children's Musical Romance
1			
2	150	Apollo 13 (1995)	Drama
1			

	Rating	Timestamp	Gender	Age	Occupation	Zip-c
0	5	978824268	F	1	10	48067
1	5	978824351	F	1	10	48067
2	5	978301777	F	1	10	48067

```

# Converting categorical variable to numerical variable
dummy=onehotcode['Genres'].str.get_dummies()
dummy.head(3)

```

	Action	Adventure	Animation	Children's	Comedy	Crime
Documentary \						
0	0	0	1	1	1	0
0						
1	0	0	1	1	0	0
0						
2	0	0	0	0	0	0
0						

	Drama	Fantasy	Film-Noir	Horror	Musical	Mystery	Romance	Sci-
Fi \								
0	0	0	0	0	0	0	0	
0								
1	0	0	0	0	1	0	1	
0								
2	1	0	0	0	0	0	0	
0								

	Western
0	0
1	0
2	0

Feature Engineering Task 3

- Determine the features affecting the ratings of any particular movie.

```
# Converting categorical column 'Gender' into numerical column.
```

```
gender_dummy=final_dataset['Gender'].str.get_dummies()
```

```
# Merging the new numerical column to final dataset based on index
```

```
final_dataset=pd.merge(final_dataset,gender_dummy,how='inner',left_index=True,right_index=True)
```

```
final_dataset.head(3)
```

	MovieID	Title	Genres
UserID \ 0	1	Toy Story (1995)	Animation Children's Comedy
1	48	Pocahontas (1995)	Animation Children's Musical Romance
1	150	Apollo 13 (1995)	Drama
2			
1			

	Rating	Timestamp	Gender	Age	Occupation	Zip-c	Action	Adventure
0	5	978824268	F	1	10	48067	0	0
1	5	978824351	F	1	10	48067	0	0
2	5	978301777	F	1	10	48067	0	0

	Animation	Children's	Comedy	...	Drama	Fantasy	Film-Noir
Horror \							
0	1	1	1	...	0	0	0
0							
1	1	1	0	...	0	0	0
0							
2	0	0	0	...	1	0	0
0							

[illegible]

```

1      1      0      1      0      0      0      0      1      0
1
2      0      0      0      0      0      0      0      1      0
1

```

```

M_y
0      0
1      0
2      0

```

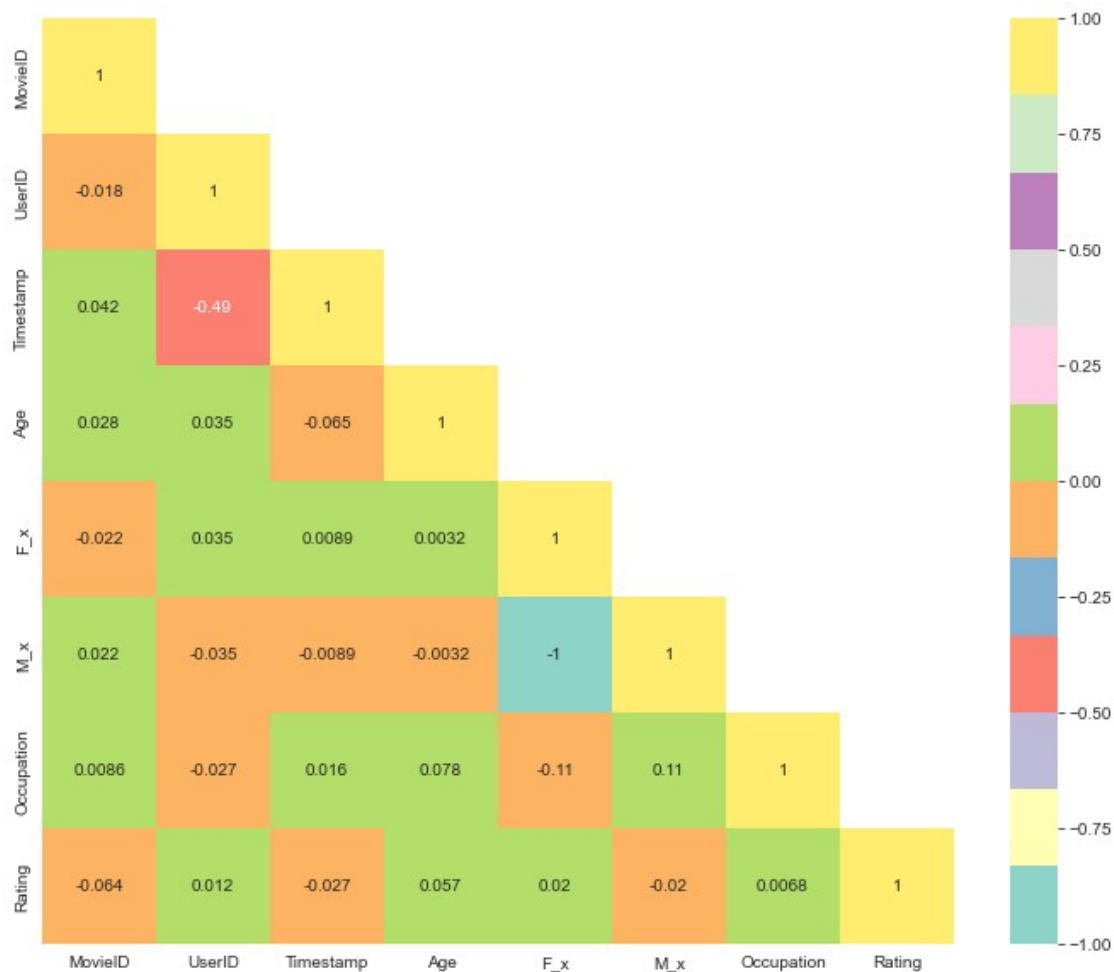
[3 rows x 32 columns]

Createing Heatmap

```

m=np.ones_like(final_dataset[['MovieID','UserID','Timestamp','Age','F_
x','M_x','Occupation','Rating']].corr())
m[np.tril_indices_from(m)]=0
sns.set_style('white')
plt.figure(figsize=(12,10))
sns.heatmap(final_dataset[['MovieID','UserID','Timestamp','Age','F_x',
'M_x','Occupation','Rating']].corr(), annot=True,cmap='Set3',mask=m)
plt.show()

```



```
# Createing a new dataframe with only continous data and dummies
features=final_dataset[['MovieID','UserID','F_x','M_x','Timestamp','Age',
'e','Occupation','Action','Adventure','Animation',
```

```
"Children's",'Comedy','Crime','Documentary','Drama','Fantasy','Film-
Noir',
```

```
'Horror','Musical','Mystery','Romance','Sci-
Fi','Thriller','War','Western','Rating']]
```

```
features
```

	MovieID	UserID	F_x	M_x	Timestamp	Age	Occupation	Action
\								
0	1	1	1	0	978824268	1	10	0
1	48	1	1	0	978824351	1	10	0
2	150	1	1	0	978301777	1	10	0
3	260	1	1	0	978300760	1	10	1
4	527	1	1	0	978824195	1	10	0
...
1000204	3513	5727	0	1	958489970	25	4	0
1000205	3535	5727	0	1	958489970	25	4	0
1000206	3536	5727	0	1	958489902	25	4	0
1000207	3555	5727	0	1	958490699	25	4	1
1000208	3578	5727	0	1	958490171	25	4	1

	Adventure	Animation	Children's	Comedy	Crime	Documentary
Drama \						
0	0	1	1	1	0	0
0						
1	0	1	1	0	0	0
0						
2	0	0	0	0	0	0
1						
3	1	0	0	0	0	0
0						
4	0	0	0	0	0	0
1						
...

...						
1000204	0	0	0	0	0	0
1						
1000205	0	0	0	1	0	0
0						
1000206	0	0	0	1	0	0
0						
1000207	0	0	0	0	0	0
0						
1000208	0	0	0	0	0	0
1						

	Fantasy	Film-Noir	Horror	Musical	Mystery	Romance	Sci-Fi
\							
0	0	0	0	0	0	0	0
1	0	0	0	1	0	1	0
2	0	0	0	0	0	0	0
3	1	0	0	0	0	0	1
4	0	0	0	0	0	0	0
...

1000204	0	0	0	0	0	0	0
1000205	0	0	1	0	0	0	0
1000206	0	0	0	0	0	1	0
1000207	0	0	0	0	0	0	0
1000208	0	0	0	0	0	0	0

	Thriller	War	Western	Rating
0	0	0	0	5
1	0	0	0	5
2	0	0	0	5
3	0	0	0	4
4	0	1	0	5
...
1000204	1	0	0	4
1000205	1	0	0	2
1000206	0	0	0	5
1000207	1	0	0	3
1000208	0	0	0	5

```
[1000209 rows x 26 columns]
```

```
# Storing features and target variable
```

```
x=features.drop(['Rating'],axis=1)
```

```
y=features['Rating']
```

```
# Feature Selection using SelectKBest
```

```
from sklearn.feature_selection import SelectKBest
```

```
from sklearn.feature_selection import chi2
```

```
# Selecting best 10 features
```

```
bestfeatures=SelectKBest(score_func=chi2,k=10)
```

```
fit=bestfeatures.fit(x,y)
```

```
dfscores=pd.DataFrame(fit.scores_)
```

```
dfcolumns=pd.DataFrame(x.columns)
```

```
# concat two dataframes for better visualization
```

```
featuresScores=pd.concat([dfcolumns,dfscores],axis=1)
```

```
featuresScores.columns=['Features','Score'] # naming the dataframe columns
```

```
print(featuresScores.nlargest(10,'Score'))
```

	Features	Score
4	Timestamp	1.638876e+08
0	MovieID	3.341344e+06
1	UserID	1.720993e+05
5	Age	1.865494e+04
14	Drama	9.705111e+03
17	Horror	9.192491e+03
23	War	6.405101e+03
16	Film-Noir	3.826022e+03
21	Sci-Fi	1.845628e+03
7	Action	1.734519e+03

```
# Features selected are stored in a dataframe
```

```
feature_selected_df
```

```
=features[['Timestamp','MovieID','UserID','Age','Drama','Horror','War',  
, 'Film-Noir','Sci-Fi','Action']]
```

```
feature_selected_df
```

	Timestamp	MovieID	UserID	Age	Drama	Horror	War	Film-
Noir \								
0	978824268	1	1	1	0	0	0	
0								
1	978824351	48	1	1	0	0	0	
0								
2	978301777	150	1	1	1	0	0	
0								
3	978300760	260	1	1	0	0	0	
0								
4	978824195	527	1	1	1	0	1	

```

0
...      ...      ...      ...      ...      ...      ...      ...
.
1000204  958489970      3513      5727      25      1      0      0
0
1000205  958489970      3535      5727      25      0      1      0
0
1000206  958489902      3536      5727      25      0      0      0
0
1000207  958490699      3555      5727      25      0      0      0
0
1000208  958490171      3578      5727      25      1      0      0
0

      Sci-Fi  Action
0          0      0
1          0      0
2          0      0
3          1      1
4          0      0
...      ...      ...
1000204      0      0
1000205      0      0
1000206      0      0
1000207      0      1
1000208      0      1

[1000209 rows x 10 columns]

```

Feature Engineering Task 4

-Develop an appropriate model to predict the movie ratings

Splitting dataset for training and testing

```
x_train,x_test,y_train,y_test=train_test_split(feature_selected_df,y,t
est_size=0.20,random_state=1)
```

Size of train and test data

```
x_train.shape,x_test.shape,y_train.shape,y_test.shape
```

```
((800167, 10), (200042, 10), (800167,), (200042,))
```

model 1- Decision Tree

Object creation

```
clf=DecisionTreeClassifier()
```

Model Fitting

```
clf=clf.fit(x_train,y_train)
```

Prediction of test data

```
y_pred=clf.predict(x_test)
```

```
print("Decision Tree accuracy :  
{:.2f}".format(metrics.accuracy_score(y_test,y_pred)))
```

Decision Tree accuracy : 0.32

```
a=pd.DataFrame()  
a['y_test']=y_test  
a['y_pred']=y_pred  
a.head(20)
```

	y_test	y_pred
630120	4	3
229398	5	5
758377	3	4
159240	5	4
254252	4	3
27168	3	3
196538	3	2
37123	3	3
982048	4	3
994502	2	3
873073	5	3
684013	1	3
852424	4	4
410026	5	5
285794	3	4
141761	4	5
787144	4	4
154218	3	2
162643	5	4
372999	5	5

Model 2-K-Nearest Neighbours

```
knn=KNeighborsClassifier(n_neighbors=7).fit(x_train,y_train)  
knnaccuracy =knn.score(x_test,y_test)  
knn_predictions = knn.predict(x_test)  
print("KNN test accuracy : {:.4f}".format(knnaccuracy))  
print("KNN train accuracy :  
{:.4f}".format(knn.score(x_train,y_train)))
```

KNN test accuracy : 0.3568
KNN train accuracy : 0.5238

```
gnb=GaussianNB().fit(x_train,y_train)  
gnbaccuracy=gnb.score(x_test,y_test)  
gnb_predictions = gnb.predict(x_test)  
print("GNB test accuracy : {:.4f}".format(gnbaccuracy))  
print("GNB train accuracy :  
{:.4f}".format(gnb.score(x_train,y_train)))
```

GNB test accuracy : 0.3485
GNB train accuracy : 0.3470

Comparing the accuracy score of all three models, k-Nearest Neighbours has higher accuracy than other models