Movielens_Project

January 30, 2022

MovieLens Project

[1]:

```
import numpy as np
import pandas as pd
from pandas import Series, DataFrame
import matplotlib.pyplot as plt
from matplotlib import style
import seaborn as sns
%matplotlib inline
```

Read the file data

[2]:

```
movies_df = pd.read_csv(
    'movies.dat',
    sep='::',
    names=['MovieID','Title','Genres'],
    engine='python',
    header=None
)
users_df = pd.read_csv(
    'users.dat',
    sep='::',
    names=['UserID','Gender','Age', 'Occupation', 'zip-code'],
    engine='python',
    header=None
)
ratings_df = pd.read_csv(
    'ratings.dat',
    sep='::',
    names=['UserID','MovieID','Rating', 'Timestamp'],
    parse_dates=['Timestamp'],
    engine='python',
    header=None
)
```

[3]:

```
movies_df.head() # first five info of movies.dat dataset
```

```
[3]:    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
```

```
[4]: users_df.head() # first five info of users.dat dataset
```

```
[4]:    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
```

```
[5]: ratings_df.head() # first five info of ratings.dat dataset
```

```
[5]:    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
```

Merge Create a new dataset [Master_Data] with MovieID Title UserID Age Gender Occupation Rating

```
[6]: movie_ratings_df = pd.merge(movies_df, ratings_df, on='MovieID')
     movie_ratings_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1000209 entries, 0 to 1000208
Data columns (total 6 columns):
 #   Column     Non-Null Count    Dtype
---  ------     --------------    -----
 0   MovieID    1000209 non-null  int64
 1   Title      1000209 non-null  object
 2   Genres     1000209 non-null  object
 3   UserID     1000209 non-null  int64
 4   Rating     1000209 non-null  int64
 5   Timestamp  1000209 non-null  int64
dtypes: int64(4), object(2)
memory usage: 53.4+ MB
```

```
[7]: movie_ratings_df.head()
```

[7]:

```
   MovieID             Title                       Genres  UserID  Rating  \
0        1  Toy Story (1995)  Animation|Children's|Comedy       1       5   
1        1  Toy Story (1995)  Animation|Children's|Comedy       6       4   
2        1  Toy Story (1995)  Animation|Children's|Comedy       8       4   
3        1  Toy Story (1995)  Animation|Children's|Comedy       9       5   
4        1  Toy Story (1995)  Animation|Children's|Comedy      10       5   

   Timestamp  
0  978824268  
1  978237008  
2  978233496  
3  978225952  
4  978226474  
```

[8]:

```python
movie_ratings_users_df = pd.merge(
    movie_ratings_df,
    users_df,
    on='UserID'
)
movie_ratings_users_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1000209 entries, 0 to 1000208
Data columns (total 10 columns):
 #   Column      Non-Null Count    Dtype 
---  ------      --------------    ----- 
 0   MovieID     1000209 non-null  int64 
 1   Title       1000209 non-null  object
 2   Genres      1000209 non-null  object
 3   UserID      1000209 non-null  int64 
 4   Rating      1000209 non-null  int64 
 5   Timestamp   1000209 non-null  int64 
 6   Gender      1000209 non-null  object
 7   Age         1000209 non-null  int64 
 8   Occupation  1000209 non-null  int64 
 9   zip-code    1000209 non-null  object
dtypes: int64(6), object(4)
memory usage: 83.9+ MB
```

[9]:

```python
movie_ratings_users_df.head()
```

[9]:

```
   MovieID                                      Title  \
0        1                           Toy Story (1995)   
1       48                          Pocahontas (1995)   
2      150                           Apollo 13 (1995)   
3      260  Star Wars: Episode IV - A New Hope (1977)   
4      527                    Schindler's List (1993)   
```

```
                                 Genres  UserID  Rating  Timestamp Gender  \
0           Animation|Children's|Comedy       1       5  978824268      F
1  Animation|Children's|Musical|Romance       1       5  978824351      F
2                                 Drama       1       5  978301777      F
3       Action|Adventure|Fantasy|Sci-Fi       1       4  978300760      F
4                             Drama|War       1       5  978824195      F

   Age  Occupation zip-code
0    1          10    48067
1    1          10    48067
2    1          10    48067
3    1          10    48067
4    1          10    48067
```

Master_Data

[10]:
```
Master_Data = movie_ratings_users_df.drop(
    ['zip-code', 'Timestamp'],
    axis=1
)
Master_Data.head()
```

[10]:
```
   MovieID                                      Title  \
0        1                           Toy Story (1995)
1       48                          Pocahontas (1995)
2      150                           Apollo 13 (1995)
3      260  Star Wars: Episode IV - A New Hope (1977)
4      527                    Schindler's List (1993)

                                 Genres  UserID  Rating Gender  Age  \
0           Animation|Children's|Comedy       1       5      F    1
1  Animation|Children's|Musical|Romance       1       5      F    1
2                                 Drama       1       5      F    1
3       Action|Adventure|Fantasy|Sci-Fi       1       4      F    1
4                             Drama|War       1       5      F    1

   Occupation
0          10
1          10
2          10
3          10
4          10
```

[11]:
```
Master_Data.describe(include='all')
```

[11]:

```
              MovieID                Title   Genres        UserID  \
count    1.000209e+06              1000209  1000209  1.000209e+06
unique            NaN                 3706      301           NaN
top               NaN  American Beauty (1999)   Comedy           NaN
freq              NaN                 3428   116883           NaN
mean     1.865540e+03                  NaN      NaN  3.024512e+03
std      1.096041e+03                  NaN      NaN  1.728413e+03
min      1.000000e+00                  NaN      NaN  1.000000e+00
25%      1.030000e+03                  NaN      NaN  1.506000e+03
50%      1.835000e+03                  NaN      NaN  3.070000e+03
75%      2.770000e+03                  NaN      NaN  4.476000e+03
max      3.952000e+03                  NaN      NaN  6.040000e+03

               Rating   Gender           Age    Occupation
count    1.000209e+06  1000209  1.000209e+06  1.000209e+06
unique            NaN        2           NaN           NaN
top               NaN        M           NaN           NaN
freq              NaN   753769           NaN           NaN
mean     3.581564e+00      NaN  2.973831e+01  8.036138e+00
std      1.117102e+00      NaN  1.175198e+01  6.531336e+00
min      1.000000e+00      NaN  1.000000e+00  0.000000e+00
25%      3.000000e+00      NaN  2.500000e+01  2.000000e+00
50%      4.000000e+00      NaN  2.500000e+01  7.000000e+00
75%      4.000000e+00      NaN  3.500000e+01  1.400000e+01
max      5.000000e+00      NaN  5.600000e+01  2.000000e+01
```

[12]:

```
Master_Data.info()
```

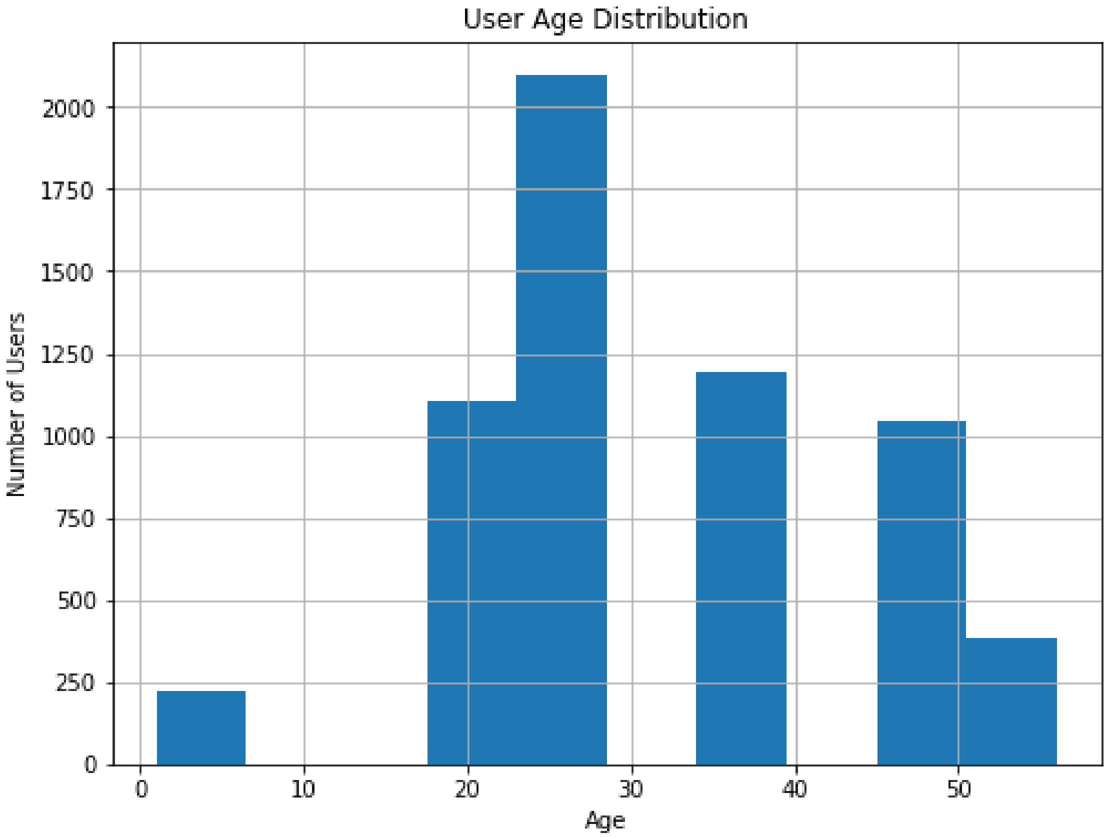
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1000209 entries, 0 to 1000208
Data columns (total 8 columns):
 #   Column      Non-Null Count    Dtype
---  ------      --------------    -----
 0   MovieID     1000209 non-null  int64
 1   Title       1000209 non-null  object
 2   Genres      1000209 non-null  object
 3   UserID      1000209 non-null  int64
 4   Rating      1000209 non-null  int64
 5   Gender      1000209 non-null  object
 6   Age         1000209 non-null  int64
 7   Occupation  1000209 non-null  int64
dtypes: int64(5), object(3)
memory usage: 68.7+ MB
```

Visual Representations of Data User Age Distribution

[13]:

```
# user age distribution
plt.figure(figsize=(8,6))
```

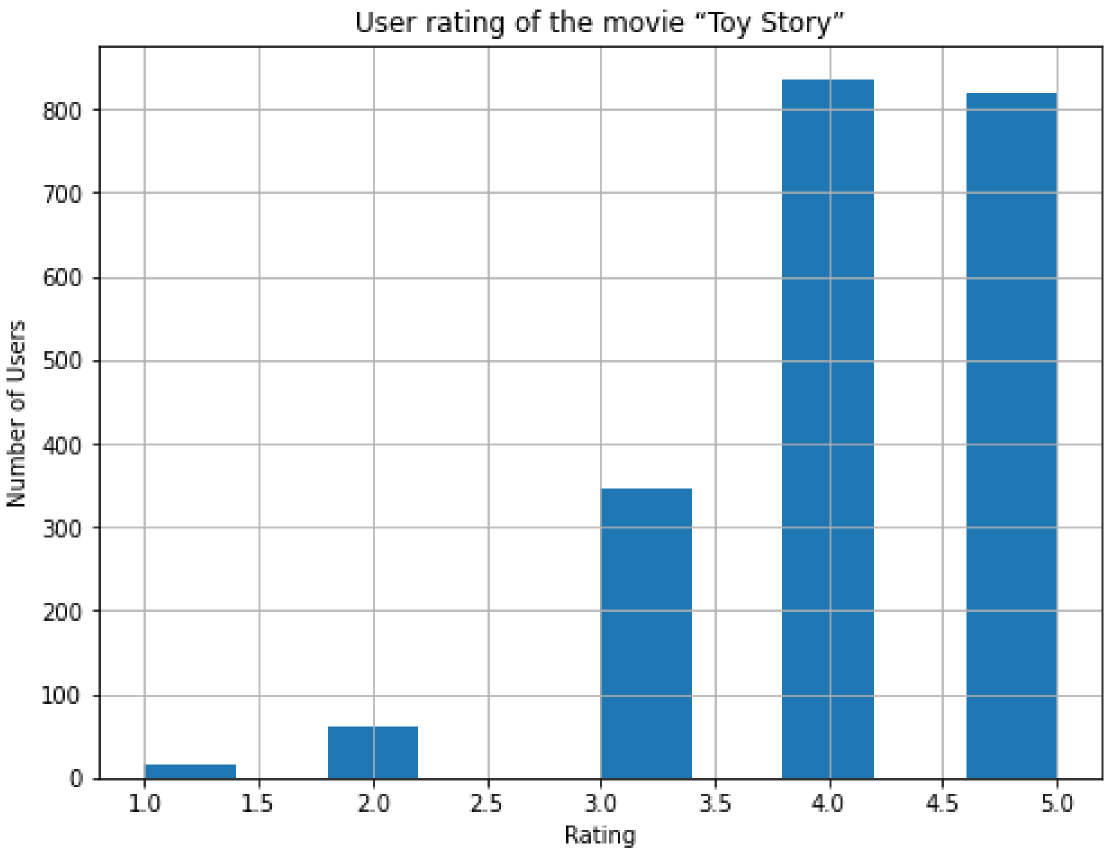
```
users_df.Age.hist()
plt.title('User Age Distribution')
plt.xlabel('Age')
plt.ylabel('Number of Users')
plt.show()
```



User rating of the movie “Toy Story”

[14]:
```
plt.figure(figsize=(8,6))
movies_grouped = movie_ratings_df.groupby('Title')
toy_story = movies_grouped.get_group('Toy Story (1995)')
toy_story['Rating'].hist()
plt.title('User rating of the movie "Toy Story"')
plt.xlabel('Rating')
plt.ylabel('Number of Users')

plt.show()
```



Top 25 movies by viewership rating Avrage rating of the movies

```
[15]: rating_avg = movie_ratings_df.groupby('Title')['Rating'].mean()
      rating_avg.head()
```

```
[15]: Title
      $1,000,000 Duck (1971)           3.027027
      'Night Mother (1986)             3.371429
      'Til There Was You (1997)        2.692308
      'burbs, The (1989)               2.910891
      …And Justice for All (1979)    3.713568
      Name: Rating, dtype: float64
```

```
[16]: rating_avg = rating_avg.sort_values(ascending=False)
      rating_avg.head()
```

```
[16]: Title
      Gate of Heavenly Peace, The (1995)    5.0
      Lured (1947)                          5.0
      Ulysses (Ulisse) (1954)               5.0
```

```
Smashing Time (1967)                  5.0
Follow the Bitch (1998)               5.0
Name: Rating, dtype: float64
```

Number of ratings for the movies

[17]:

```
rating_count = movie_ratings_df.groupby('Title')['Rating']
rating_count = rating_count.count().sort_values(ascending=False)
rating_count[:25]
```

[17]:

```
Title
American Beauty (1999)                                   3428
Star Wars: Episode IV - A New Hope (1977)                2991
Star Wars: Episode V - The Empire Strikes Back (1980)    2990
Star Wars: Episode VI - Return of the Jedi (1983)        2883
Jurassic Park (1993)                                     2672
Saving Private Ryan (1998)                               2653
Terminator 2: Judgment Day (1991)                        2649
Matrix, The (1999)                                       2590
Back to the Future (1985)                                2583
Silence of the Lambs, The (1991)                         2578
Men in Black (1997)                                      2538
Raiders of the Lost Ark (1981)                           2514
Fargo (1996)                                             2513
Sixth Sense, The (1999)                                  2459
Braveheart (1995)                                        2443
Shakespeare in Love (1998)                               2369
Princess Bride, The (1987)                               2318
Schindler's List (1993)                                  2304
L.A. Confidential (1997)                                 2288
Groundhog Day (1993)                                     2278
E.T. the Extra-Terrestrial (1982)                        2269
Star Wars: Episode I - The Phantom Menace (1999)         2250
Being John Malkovich (1999)                              2241
Shawshank Redemption, The (1994)                         2227
Godfather, The (1972)                                    2223
Name: Rating, dtype: int64
```

[18]:

```
rating_avg_count = pd.DataFrame(data=rating_avg)
rating_avg_count['number_of_ratings'] = pd.DataFrame(rating_count)
rating_avg_count.head()
```

[18]:

```
                                    Rating  number_of_ratings
Title
Gate of Heavenly Peace, The (1995)     5.0                  3
Lured (1947)                           5.0                  1
Ulysses (Ulisse) (1954)                5.0                  1
```

```
Smashing Time (1967)                      5.0                    2
Follow the Bitch (1998)                   5.0                    1
```

[19]:

```
rating_avg_count.describe()
```

[19]:

```
             Rating  number_of_ratings
count   3706.000000        3706.000000
mean       3.238892         269.889099
std        0.672925         384.047838
min        1.000000           1.000000
25%        2.822705          33.000000
50%        3.331546         123.500000
75%        3.740741         350.000000
max        5.000000        3428.000000
```

Top 25 movies by viewership rating excluding movies with less than 10 ratings

[20]:

```
filter_data = rating_avg_count[rating_avg_count['number_of_ratings'] > 10]
filter_data[:25]
```

[20]:

```
                                                      Rating  \
Title
Sanjuro (1962)                                      4.608696
Seven Samurai (The Magnificent Seven) (Shichini…  4.560510
Shawshank Redemption, The (1994)                    4.554558
Godfather, The (1972)                               4.524966
Close Shave, A (1995)                               4.520548
Usual Suspects, The (1995)                          4.517106
Schindler's List (1993)                             4.510417
Wrong Trousers, The (1993)                          4.507937
Sunset Blvd. (a.k.a. Sunset Boulevard) (1950)       4.491489
Raiders of the Lost Ark (1981)                      4.477725
Rear Window (1954)                                  4.476190
Paths of Glory (1957)                               4.473913
Star Wars: Episode IV - A New Hope (1977)           4.453694
Third Man, The (1949)                               4.452083
Dr. Strangelove or: How I Learned to Stop Worry…  4.449890
For All Mankind (1989)                              4.444444
Wallace & Gromit: The Best of Aardman Animation…  4.426941
To Kill a Mockingbird (1962)                        4.425647
Double Indemnity (1944)                             4.415608
Casablanca (1942)                                   4.412822
World of Apu, The (Apur Sansar) (1959)              4.410714
Sixth Sense, The (1999)                             4.406263
Yojimbo (1961)                                      4.404651
Pather Panchali (1955)                              4.404255
Lawrence of Arabia (1962)                           4.401925
```

```
                                                    number_of_ratings
Title
Sanjuro (1962)                                                     69
Seven Samurai (The Magnificent Seven) (Shichini…                628
Shawshank Redemption, The (1994)                                 2227
Godfather, The (1972)                                            2223
Close Shave, A (1995)                                             657
Usual Suspects, The (1995)                                       1783
Schindler's List (1993)                                          2304
Wrong Trousers, The (1993)                                        882
Sunset Blvd. (a.k.a. Sunset Boulevard) (1950)                     470
Raiders of the Lost Ark (1981)                                   2514
Rear Window (1954)                                               1050
Paths of Glory (1957)                                             230
Star Wars: Episode IV - A New Hope (1977)                        2991
Third Man, The (1949)                                             480
Dr. Strangelove or: How I Learned to Stop Worry…               1367
For All Mankind (1989)                                             27
Wallace & Gromit: The Best of Aardman Animation…                438
To Kill a Mockingbird (1962)                                      928
Double Indemnity (1944)                                           551
Casablanca (1942)                                                1669
World of Apu, The (Apur Sansar) (1959)                             56
Sixth Sense, The (1999)                                          2459
Yojimbo (1961)                                                    215
Pather Panchali (1955)                                             47
Lawrence of Arabia (1962)                                         831
```

The ratings for all the movies reviewed by user ID 2696

```
[21]: user_2696 = movie_ratings_users_df[movie_ratings_users_df['UserID'] == 2696]
      user_2696
```

```
[21]:
        MovieID                                           Title  \
991035      350                              Client, The (1994)
991036      800                                Lone Star (1996)
991037     1092                           Basic Instinct (1992)
991038     1097               E.T. the Extra-Terrestrial (1982)
991039     1258                             Shining, The (1980)
991040     1270                       Back to the Future (1985)
991041     1589                                 Cop Land (1997)
991042     1617                        L.A. Confidential (1997)
991043     1625                                Game, The (1997)
991044     1644          I Know What You Did Last Summer (1997)
991045     1645                    Devil's Advocate, The (1997)
991046     1711  Midnight in the Garden of Good and Evil (1997)
```

```
991047     1783                                 Palmetto (1998)
991048     1805                              Wild Things (1998)
991049     1892                        Perfect Murder, A (1998)
991050     2338    I Still Know What You Did Last Summer (1998)
991051     2389                                   Psycho (1998)
991052     2713                              Lake Placid (1999)
991053     3176                 Talented Mr. Ripley, The (1999)
991054     3386                                      JFK (1991)

                                  Genres  UserID  Rating  Timestamp Gender  \
991035            Drama|Mystery|Thriller    2696       3  973308886      M
991036                     Drama|Mystery    2696       5  973308842      M
991037                  Mystery|Thriller    2696       4  973308886      M
991038   Children's|Drama|Fantasy|Sci-Fi    2696       3  973308690      M
991039                            Horror    2696       4  973308710      M
991040                     Comedy|Sci-Fi    2696       2  973308676      M
991041               Crime|Drama|Mystery    2696       3  973308865      M
991042  Crime|Film-Noir|Mystery|Thriller    2696       4  973308842      M
991043                  Mystery|Thriller    2696       4  973308842      M
991044           Horror|Mystery|Thriller    2696       2  973308920      M
991045     Crime|Horror|Mystery|Thriller    2696       4  973308904      M
991046        Comedy|Crime|Drama|Mystery    2696       4  973308904      M
991047        Film-Noir|Mystery|Thriller    2696       4  973308865      M
991048      Crime|Drama|Mystery|Thriller    2696       4  973308886      M
991049                  Mystery|Thriller    2696       4  973308904      M
991050           Horror|Mystery|Thriller    2696       2  973308920      M
991051             Crime|Horror|Thriller    2696       4  973308710      M
991052                   Horror|Thriller    2696       1  973308710      M
991053            Drama|Mystery|Thriller    2696       4  973308865      M
991054                     Drama|Mystery    2696       1  973308842      M

        Age  Occupation zip-code
991035   25           7    24210
991036   25           7    24210
991037   25           7    24210
991038   25           7    24210
991039   25           7    24210
991040   25           7    24210
991041   25           7    24210
991042   25           7    24210
991043   25           7    24210
991044   25           7    24210
991045   25           7    24210
991046   25           7    24210
991047   25           7    24210
991048   25           7    24210
991049   25           7    24210
```

```
991050   25          7    24210
991051   25          7    24210
991052   25          7    24210
991053   25          7    24210
991054   25          7    24210
```

Feature Engineering The unique genres

[22]:
```
movie_ratings_df['Genres'].value_counts().head()
```

[22]:
```
Comedy            116883
Drama             111423
Comedy|Romance     42712
Comedy|Drama       42245
Drama|Romance      29170
Name: Genres, dtype: int64
```

[23]:
```
movie_ratings_df['Genres'].unique()
```

[23]:
```
array(["Animation|Children's|Comedy", "Adventure|Children's|Fantasy",
       'Comedy|Romance', 'Comedy|Drama', 'Comedy',
       'Action|Crime|Thriller', "Adventure|Children's", 'Action',
       'Action|Adventure|Thriller', 'Comedy|Drama|Romance',
       'Comedy|Horror', "Animation|Children's", 'Drama',
       'Action|Adventure|Romance', 'Drama|Thriller', 'Drama|Romance',
       'Thriller', 'Action|Comedy|Drama', 'Crime|Drama|Thriller',
       'Drama|Sci-Fi', 'Romance', 'Adventure|Sci-Fi', 'Adventure|Romance',
       "Children's|Comedy|Drama", 'Documentary', 'Drama|War',
       'Action|Crime|Drama', 'Action|Adventure', 'Crime|Thriller',
       "Animation|Children's|Musical|Romance", "Children's|Comedy",
       'Drama|Mystery', 'Sci-Fi|Thriller',
       'Action|Comedy|Crime|Horror|Thriller', 'Drama|Musical',
       'Crime|Drama|Romance', 'Adventure|Drama', 'Action|Thriller',
       "Adventure|Children's|Comedy|Musical", 'Action|Drama|War',
       'Action|Adventure|Crime', 'Crime', 'Drama|Mystery|Romance',
       'Action|Drama', 'Drama|Romance|War', 'Horror',
       'Action|Adventure|Comedy|Crime', 'Comedy|War',
       'Action|Adventure|Mystery|Sci-Fi', 'Drama|Thriller|War',
       'Action|Romance|Thriller', 'Crime|Film-Noir|Mystery|Thriller',
       'Action|Adventure|Drama|Romance', "Adventure|Children's|Drama",
       'Action|Sci-Fi|Thriller', 'Action|Adventure|Sci-Fi',
       "Action|Children's", 'Horror|Sci-Fi', 'Action|Crime|Sci-Fi',
       'Western', "Animation|Children's|Comedy|Romance",
       "Children's|Drama", 'Crime|Drama',
       'Drama|Fantasy|Romance|Thriller', 'Drama|Horror', 'Comedy|Sci-Fi',
       'Mystery|Thriller', "Adventure|Children's|Comedy|Fantasy|Romance",
       'Action|Adventure|Fantasy|Sci-Fi', 'Drama|Romance|War|Western',
'Action|Drama|Thriller', 'Crime|Drama|Romance|Thriller',
'Action|Adventure|Western', 'Horror|Thriller',
"Children's|Comedy|Fantasy", 'Film-Noir|Thriller',
'Action|Comedy|Musical|Sci-Fi', "Children's",
'Drama|Mystery|Thriller', 'Comedy|Romance|War', 'Action|Comedy',
"Adventure|Children's|Romance", "Animation|Children's|Musical",
'Comedy|Crime|Fantasy', 'Action|Comedy|Western', 'Action|Sci-Fi',
'Action|Adventure|Comedy|Romance', 'Comedy|Thriller',
'Horror|Sci-Fi|Thriller', 'Mystery|Romance|Thriller',
'Comedy|Western', 'Drama|Western',
'Action|Adventure|Crime|Thriller', 'Action|Comedy|War',
'Comedy|Mystery', 'Comedy|Mystery|Romance', 'Comedy|Drama|War',
'Action|Drama|Mystery', 'Comedy|Crime|Horror', 'Film-Noir|Sci-Fi',
'Comedy|Romance|Thriller', "Action|Adventure|Children's|Sci-Fi",
"Children's|Comedy|Musical", 'Action|Adventure|Comedy',
'Action|Crime|Romance',
"Action|Adventure|Animation|Children's|Fantasy",
"Animation|Children's|Comedy|Musical", 'Adventure|Drama|Western',
'Action|Adventure|Crime|Drama',
'Action|Adventure|Animation|Horror|Sci-Fi', 'Action|Horror|Sci-Fi',
'War', 'Action|Adventure|Mystery', 'Mystery',
'Action|Adventure|Fantasy',
"Adventure|Animation|Children's|Comedy|Fantasy", 'Sci-Fi',
'Documentary|Drama', 'Action|Adventure|Comedy|War',
'Crime|Film-Noir|Thriller', 'Animation',
'Action|Adventure|Romance|Thriller', 'Animation|Sci-Fi',
'Animation|Comedy|Thriller', 'Film-Noir', 'Sci-Fi|War',
'Adventure', 'Comedy|Crime', 'Action|Sci-Fi|War',
'Comedy|Fantasy|Romance|Sci-Fi', 'Fantasy',
'Action|Mystery|Thriller', 'Comedy|Musical',
'Action|Adventure|Sci-Fi|Thriller', "Children's|Drama|Fantasy",
'Adventure|War', 'Musical|Romance', 'Comedy|Musical|Romance',
'Comedy|Mystery|Romance|Thriller', 'Film-Noir|Mystery', 'Musical',
"Adventure|Children's|Drama|Musical",
'Drama|Mystery|Sci-Fi|Thriller', 'Romance|Thriller',
'Film-Noir|Romance|Thriller', 'Crime|Film-Noir|Mystery',
'Adventure|Comedy', 'Action|Adventure|Romance|War', 'Romance|War',
'Action|Drama|Western', 'Action|Crime',
"Children's|Comedy|Western", "Adventure|Children's|Comedy",
"Children's|Comedy|Mystery", "Adventure|Children's|Fantasy|Sci-Fi",
"Adventure|Animation|Children's|Musical",
"Adventure|Children's|Musical", 'Crime|Film-Noir',
"Adventure|Children's|Comedy|Fantasy",
"Children's|Drama|Fantasy|Sci-Fi", 'Action|Romance',
'Adventure|Western', 'Comedy|Fantasy', 'Animation|Comedy',
'Crime|Drama|Film-Noir', 'Action|Adventure|Drama|Sci-Fi|War',
'Action|Sci-Fi|Thriller|War', 'Action|Western',
"Action|Animation|Children's|Sci-Fi|Thriller|War",
'Action|Adventure|Romance|Sci-Fi|War',
'Action|Horror|Sci-Fi|Thriller',
'Action|Adventure|Comedy|Horror|Sci-Fi', 'Action|Comedy|Musical',
'Mystery|Sci-Fi', 'Film-Noir|Mystery|Thriller',
'Adventure|Comedy|Drama', 'Action|Adventure|Comedy|Horror',
'Action|Drama|Mystery|Romance|Thriller', 'Comedy|Mystery|Thriller',
'Adventure|Animation|Sci-Fi|Thriller', 'Action|Drama|Romance',
'Action|Adventure|Drama', 'Comedy|Drama|Musical',
'Documentary|War', 'Drama|Musical|War', 'Action|Horror',
'Horror|Romance', 'Action|Comedy|Sci-Fi|War', 'Crime|Drama|Sci-Fi',
'Action|Romance|War', 'Action|Comedy|Crime|Drama',
'Action|Drama|Thriller|War', "Action|Adventure|Children's",
"Action|Adventure|Children's|Fantasy",
"Adventure|Animation|Children's|Comedy|Musical",
'Action|Adventure|Comedy|Sci-Fi', "Children's|Fantasy",
'Crime|Drama|Mystery', 'Action|Mystery|Sci-Fi|Thriller',
'Action|Mystery|Romance|Thriller', 'Adventure|Thriller',
'Action|Thriller|War', 'Action|Crime|Mystery',
'Horror|Mystery|Thriller', 'Crime|Horror|Mystery|Thriller',
'Comedy|Drama|Thriller', 'Drama|Sci-Fi|Thriller',
'Drama|Romance|Thriller', 'Action|Adventure|Sci-Fi|War',
'Comedy|Crime|Drama|Mystery', 'Comedy|Crime|Mystery|Thriller',
'Film-Noir|Sci-Fi|Thriller', 'Adventure|Sci-Fi|Thriller',
'Crime|Drama|Mystery|Thriller', 'Comedy|Crime|Drama',
'Comedy|Documentary', 'Documentary|Musical',
'Action|Drama|Sci-Fi|Thriller',
"Adventure|Animation|Children's|Fantasy",
'Adventure|Comedy|Romance', 'Mystery|Sci-Fi|Thriller',
'Action|Comedy|Crime', "Animation|Children's|Fantasy|War",
'Action|Crime|Drama|Thriller', 'Comedy|Sci-Fi|Western',
"Children's|Fantasy|Musical", 'Fantasy|Sci-Fi',
"Children's|Comedy|Sci-Fi", "Action|Adventure|Children's|Comedy",
"Adventure|Children's|Drama|Romance",
"Adventure|Children's|Sci-Fi",
"Adventure|Children's|Comedy|Fantasy|Sci-Fi",
"Animation|Children's|Comedy|Musical|Romance",
"Children's|Musical", 'Drama|Fantasy',
"Animation|Children's|Fantasy|Musical", 'Adventure|Comedy|Musical',
"Children's|Sci-Fi", "Children's|Horror", 'Comedy|Fantasy|Romance',
'Comedy|Crime|Thriller', "Adventure|Animation|Children's|Sci-Fi",
'Action|Crime|Mystery|Thriller', 'Adventure|Musical',
"Animation|Children's|Drama|Fantasy", "Children's|Fantasy|Sci-Fi",
'Adventure|Fantasy|Romance', 'Crime|Horror',
'Action|Adventure|Horror', 'Adventure|Fantasy|Sci-Fi',
'Drama|Film-Noir|Thriller', 'Action|Comedy|Fantasy',
'Sci-Fi|Thriller|War', 'Action|Adventure|Sci-Fi|Thriller|War',
'Action|Adventure|Drama|Thriller', 'Crime|Horror|Thriller',
'Animation|Musical', 'Action|War',
'Action|Comedy|Romance|Thriller', 'Comedy|Horror|Thriller',
'Drama|Horror|Thriller', 'Action|Sci-Fi|Thriller|Western',
'Drama|Romance|Sci-Fi', 'Action|Adventure|Horror|Thriller',
'Comedy|Film-Noir|Thriller', 'Comedy|Horror|Musical|Sci-Fi',
'Comedy|Romance|Sci-Fi', 'Action|Comedy|Sci-Fi|Thriller',
'Action|Sci-Fi|Western', 'Comedy|Horror|Musical', 'Crime|Mystery',
'Animation|Mystery', 'Action|Horror|Thriller',
'Action|Drama|Fantasy|Romance', 'Horror|Mystery',
"Adventure|Animation|Children's", 'Musical|Romance|War',
'Adventure|Drama|Romance', 'Adventure|Animation|Film-Noir',
'Action|Adventure|Animation', 'Comedy|Drama|Western',
'Adventure|Comedy|Sci-Fi', 'Drama|Romance|Western',
'Comedy|Drama|Sci-Fi', 'Action|Drama|Romance|Thriller',
'Adventure|Romance|Sci-Fi', 'Film-Noir|Horror',
'Crime|Drama|Film-Noir|Thriller', 'Action|Adventure|War',
'Romance|Western', "Action|Children's|Fantasy",
'Adventure|Drama|Thriller', 'Adventure|Fantasy', 'Musical|War',
'Adventure|Musical|Romance', 'Action|Romance|Sci-Fi',
'Drama|Film-Noir', 'Comedy|Horror|Sci-Fi',
'Adventure|Drama|Romance|Sci-Fi', 'Adventure|Animation|Sci-Fi',
'Adventure|Crime|Sci-Fi|Thriller'], dtype=object)
```

Genre category with a one-hot encoding (1 and 0)

[26]:

```
movie_ratings_selected_df = movie_ratings_users_df[[
    'Gender',
    'Age',
    'Occupation',
    'Rating',
    'Genres'
]]
```

[31]:

```
Genre = movie_ratings_selected_df['Genres']
Genre = Genre.str.get_dummies().add_prefix('Genres_')
movie_ratings_genres_df = pd.concat(
    [movie_ratings_selected_df.drop(
        ['Genres'],
        axis=1
    ),
     Genre],
    axis=1
)
movie_ratings_genres_df.head()
```

[31]:

```
  Gender  Age  Occupation  Rating  Genres_Action  Genres_Adventure  \
0      F    1          10       5              0                 0
1      F    1          10       5              0                 0
2      F    1          10       5              0                 0
3      F    1          10       4              1                 1
4      F    1          10       5              0                 0

   Genres_Animation  Genres_Children's  Genres_Comedy  Genres_Crime  …  \
0                 1                  1              1             0  …
1                 1                  1              0             0  …
2                 0                  0              0             0  …
3                 0                  0              0             0  …
4                 0                  0              0             0  …

   Genres_Fantasy  Genres_Film-Noir  Genres_Horror  Genres_Musical  \
0               0                 0              0               0
1               0                 0              0               1
2               0                 0              0               0
3               1                 0              0               0
4               0                 0              0               0

   Genres_Mystery  Genres_Romance  Genres_Sci-Fi  Genres_Thriller  Genres_War  \
0               0               0              0                0           0
1               0               1              0                0           0
2               0               0              0                0           0
3               0               0              1                0           0
4               0               0              0                0           1

   Genres_Western
0               0
1               0
2               0
3               0
4               0

[5 rows x 22 columns]
```

[32]:

```
movie_ratings_genres_df = pd.get_dummies(
    movie_ratings_genres_df,
    columns=['Gender']
)
```

[33]:

```
movie_ratings_genres_df.head()
```

[33]:

```
   Age  Occupation  Rating  Genres_Action  Genres_Adventure  Genres_Animation  \
0    1          10       5              0                 0                 1
1    1          10       5              0                 0                 1
```

```
2    1          10       5              0                 0                 0
3    1          10       4              1                 1                 0
4    1          10       5              0                 0                 0

   Genres_Children's  Genres_Comedy  Genres_Crime  Genres_Documentary  ...  \
0                  1              1             0                   0  ...
1                  1              0             0                   0  ...
2                  0              0             0                   0  ...
3                  0              0             0                   0  ...
4                  0              0             0                   0  ...

   Genres_Horror  Genres_Musical  Genres_Mystery  Genres_Romance  \
0              0               0               0               0
1              0               1               0               1
2              0               0               0               0
3              0               0               0               0
4              0               0               0               0

   Genres_Sci-Fi  Genres_Thriller  Genres_War  Genres_Western  Gender_F  \
0              0                0           0               0         1
1              0                0           0               0         1
2              0                0           0               0         1
3              1                0           0               0         1
4              0                0           1               0         1

   Gender_M
0         0
1         0
2         0
3         0
4         0

[5 rows x 23 columns]
```

```
[34]: movie_ratings_genres_df.columns
```

```
[34]: Index(['Age', 'Occupation', 'Rating', 'Genres_Action', 'Genres_Adventure',
             'Genres_Animation', 'Genres_Children's', 'Genres_Comedy',
             'Genres_Crime', 'Genres_Documentary', 'Genres_Drama', 'Genres_Fantasy',
             'Genres_Film-Noir', 'Genres_Horror', 'Genres_Musical', 'Genres_Mystery',
             'Genres_Romance', 'Genres_Sci-Fi', 'Genres_Thriller', 'Genres_War',
             'Genres_Western', 'Gender_F', 'Gender_M'],
            dtype='object')
```

Features affecting the ratings of any particular movie.

```
[35]: movie_ratings_genres_df.dtypes
```

[35]:

```
Age                    int64
Occupation             int64
Rating                 int64
Genres_Action          int64
Genres_Adventure       int64
Genres_Animation       int64
Genres_Children's      int64
Genres_Comedy          int64
Genres_Crime           int64
Genres_Documentary     int64
Genres_Drama           int64
Genres_Fantasy         int64
Genres_Film-Noir       int64
Genres_Horror          int64
Genres_Musical         int64
Genres_Mystery         int64
Genres_Romance         int64
Genres_Sci-Fi          int64
Genres_Thriller        int64
Genres_War             int64
Genres_Western         int64
Gender_F               uint8
Gender_M               uint8
dtype: object
```

Linear Regression

[36]:

```
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split

from sklearn import metrics

lineReg = LinearRegression(
    copy_X=True,
    fit_intercept=True,
    n_jobs=1,
    normalize=False
)
```

[37]:

```
movie_ratings_users_sample_df = movie_ratings_genres_df.sample(
    n=50000,
    random_state=0
)
movie_ratings_users_sample_df.head()
```

[37]:

```
        Age  Occupation  Rating  Genres_Action  Genres_Adventure  \
324271   18           4       4              0                 0
```

```
818637   18            4       3              0                 0
148677   18           14       5              0                 0
778790   50            7       4              0                 0
525489   25            2       5              0                 0

        Genres_Animation  Genres_Children's  Genres_Comedy  Genres_Crime  \
324271                 0                  0              1             0
818637                 1                  1              0             0
148677                 0                  0              0             0
778790                 0                  0              1             1
525489                 0                  0              0             0

        Genres_Documentary  …  Genres_Horror  Genres_Musical  \
324271                   0  …              0               0
818637                   0  …              0               1
148677                   0  …              0               0
778790                   0  …              0               0
525489                   0  …              0               0

        Genres_Mystery  Genres_Romance  Genres_Sci-Fi  Genres_Thriller  \
324271               0               0              0                0
818637               0               0              0                0
148677               0               0              0                1
778790               0               0              0                0
525489               0               1              0                0

        Genres_War  Genres_Western  Gender_F  Gender_M
324271           0               0         0         1
818637           0               0         1         0
148677           0               0         0         1
778790           0               0         0         1
525489           0               0         0         1

[5 rows x 23 columns]
```

[38]:

```python
x = movie_ratings_users_sample_df.drop('Rating', axis=1)
y = movie_ratings_users_sample_df['Rating']
```

[39]:

```python
x.shape
```

[39]:

```
(50000, 22)
```

[40]:

```python
x_train, x_test, y_train, y_test = train_test_split(
    x,
    y,
    test_size=0.20,
    random_state=0
)
```

[41]:

```
linear_reg = LinearRegression()
```

[44]:

```
linear_reg.fit(x_train, y_train)
```

[44]:

```
LinearRegression()
```

[45]:

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

Evaluation

[46]:

```
print(
    'y-intercept: ',
    linear_reg.intercept_
)
print(
    'Beta coefficients: ',
    linear_reg.coef_
)
print(
    'Mean Abs Error  MAE: ',
    metrics.mean_absolute_error(y_test, y_pred)
)
print(
    'Mean Sq Error  MSE: ',
    metrics.mean_squared_error(y_test, y_pred)
)
print(
    'Root Mean Sq Error RMSE:',
    np.sqrt(metrics.mean_squared_error(y_test, y_pred))
)
print(
    'r2 value: ',
    metrics.r2_score(y_test, y_pred)
)
```

```
y-intercept:  3.371413755515969
Beta coefficients:  [ 0.00406322  0.00098825 -0.0933231   0.00822898  0.41190314
-0.32536968
 -0.00937548  0.07845926  0.43311855  0.22781148  0.07368389  0.3951835
 -0.29085584  0.12523149  0.02288591  0.00234758 -0.01347635  0.06128953
  0.30880281  0.14777492  0.01440465 -0.01440465]
Mean Abs Error  MAE:  0.8978299534841195
Mean Sq Error  MSE:  1.1977731707567232
Root Mean Sq Error RMSE: 1.0944282391992282
r2 value:  0.03795269985311833
```

Age, and Occupation are the main features affecting the ratings for the movies

```
[47]: x_train.dtypes
```

```
[47]: Age                      int64
      Occupation               int64
      Genres_Action            int64
      Genres_Adventure         int64
      Genres_Animation         int64
      Genres_Children's        int64
      Genres_Comedy            int64
      Genres_Crime             int64
      Genres_Documentary       int64
      Genres_Drama             int64
      Genres_Fantasy           int64
      Genres_Film-Noir         int64
      Genres_Horror            int64
      Genres_Musical           int64
      Genres_Mystery           int64
      Genres_Romance           int64
      Genres_Sci-Fi            int64
      Genres_Thriller          int64
      Genres_War               int64
      Genres_Western           int64
      Gender_F                 uint8
      Gender_M                 uint8
      dtype: object
```

```
[48]: prediction_df = pd.DataFrame({'Test': y_test, 'Prediction': y_pred})
      prediction_df.head()
```

```
[48]:         Test  Prediction
      187446     4    4.322363
      69421      4    3.439548
      941725     3    3.408593
      841836     4    3.652663
      869012     4    3.559433
```

```
[ ]:
```