## Project 3 - Movielens Dataset Analysis

%matplotlib inline

##import quandl

import pandas as pd

from pandas import Series, DataFrame

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

from datetime import datetime

movies = pd.read\_csv('movies.dat',sep='::',header=None,names=['MovieID','Title','Genre'],engine='python')

movies.head()

|  | **MovieID** | **Title** | **Genre** |
| --- | --- | --- | --- |
| **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) | Comed |

ratings = pd.read\_csv('ratings.dat',sep='::',header=None,names=['UserID','MovieID','Rating','Timestamp'],engine='python')

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-code'],engine='python')

users.head()

| **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 |

Master\_data1 = pd.merge(movies,

ratings[['MovieID', 'UserID', 'Rating']],

left\_on='MovieID',right\_on='MovieID')

Master\_data1.head()

|  | **MovieID** | **Title** | **Genre** | **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 |

Master\_data2 = Master\_data1.drop(['Genre'], axis=1)

Master\_data2.head()

|  | **MovieID** | **Title** | **UserID** | **Rating** |
| --- | --- | --- | --- | --- |
| **0** | 1 | Toy Story (1995) | 1 | 5 |
| **1** | 1 | Toy Story (1995) | 6 | 4 |
| **2** | 1 | Toy Story (1995) | 8 | 4 |
| **3** | 1 | Toy Story (1995) | 9 | 5 |
| **4** | 1 | Toy Story (1995) | 10 | 5 |

Master\_data = pd.merge(Master\_data2,

users[['UserID','Gender','Age','Occupation']],

left\_on='UserID',right\_on='UserID')

Master\_data.head()

Master\_data.head()

|  | **MovieID** | **Title** | **UserID** | **Rating** | **Gender** | **Age** | **Occupation** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 1 | Toy Story (1995) | 1 | 5 | F | 1 | 10 |
| **1** | 48 | Pocahontas (1995) | 1 | 5 | F | 1 | 10 |
| **2** | 150 | Apollo 13 (1995) | 1 | 5 | F | 1 | 10 |
| **3** | 260 | Star Wars: Episode IV - A New Hope (1977) | 1 | 4 | F | 1 | 10 |
| **4** | 527 | Schindler's List (1993) | 1 | 5 | F | 1 | 10 |

## User rating of the movie “Toy Story”

M1 = Master\_data.Title.str.split('(').str[0].str.strip() == 'Toy Story'

M2 = Master\_data[M1]

print(M2['Rating'])

0 5

53 4

124 4

263 5

369 5

770 4

1075 5

1330 3

1352 4

1656 3

2056 3

2163 5

2327 5

2678 5

2778 5

2971 4

3268 4

Etc….

## Top 25 movies by viewership rating

Master\_data.nlargest(25,'Rating')

| **MovieID** | **Title** | **UserID** | **Rating** | **Gender** | **Age** | **Occupation** |
| --- | --- | --- | --- | --- | --- | --- |
| **0** | 1 | Toy Story (1995) | 1 | 5 | F | 1 | 10 |
| **1** | 48 | Pocahontas (1995) | 1 | 5 | F | 1 | 10 |
| **2** | 150 | Apollo 13 (1995) | 1 | 5 | F | 1 | 10 |
| **4** | 527 | Schindler's List (1993) | 1 | 5 | F | 1 | 10 |
| **8** | 595 | Beauty and the Beast (1991) | 1 | 5 | F | 1 | 10 |
| **17** | 1022 | Cinderella (1950) | 1 | 5 | F | 1 | 10 |
| **18** | 1028 | Mary Poppins (1964) | 1 | 5 | F | 1 | 10 |
| **19** | 1029 | Dumbo (1941) | 1 | 5 | F | 1 | 10 |
| **20** | 1035 | Sound of Music, The (1965) | 1 | 5 | F | 1 | 10 |
| **22** | 1193 | One Flew Over the Cuckoo's Nest (1975) | 1 | 5 | F | 1 | 10 |
| **26** | 1270 | Back to the Future (1985) | 1 | 5 | F | 1 | 10 |
| **27** | 1287 | Ben-Hur (1959) | 1 | 5 | F | 1 | 10 |
| **31** | 1836 | Last Days of Disco, The (1998) | 1 | 5 | F | 1 | 10 |
| **33** | 1961 | Rain Man (1988) | 1 | 5 | F | 1 | 10 |
| **36** | 2028 | Saving Private Ryan (1998) | 1 | 5 | F | 1 | 10 |
| **40** | 2355 | Bug's Life, A (1998) | 1 | 5 | F | 1 | 10 |
| **47** | 2804 | Christmas Story, A (1983) | 1 | 5 | F | 1 | 10 |
| **49** | 3105 | Awakenings (1990) | 1 | 5 | F | 1 | 10 |
| **56** | 48 | Pocahontas (1995) | 6 | 5 | F | 50 | 9 |
| **57** | 199 | Umbrellas of Cherbourg, The (Parapluies de Che... | 6 | 5 | F | 50 | 9 |
| **68** | 597 | Pretty Woman (1990) | 6 | 5 | F | 50 | 9 |
| **71** | 914 | My Fair Lady (1964) | 6 | 5 | F | 50 | 9 |
| **76** | 1035 | Sound of Music, The (1965) | 6 | 5 | F | 50 | 9 |
| **78** | 1088 | Dirty Dancing (1987) | 6 | 5 | F | 50 | 9 |
| **83** | 1380 | Grease (1978) | 6 | 5 | F | 50 | 9 |

## Find the ratings for all the movies reviewed by for a particular user of user id = 2696

Master\_data\_2696 = Master\_data.loc[Master\_data.UserID == 2696]

print(Master\_data\_2696.Rating)

991035 3

991036 5

991037 4

991038 3

991039 4

991040 2

991041 3

991042 4

991043 4

991044 2

991045 4

991046 4

991047 4

991048 4

991049 4

991050 2

991051 4

991052 1

991053 4

991054 1

Name: Rating, dtype: int64

## 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)

Master\_data\_Genre = pd.merge(Master\_data1,

users[['UserID','Gender','Age','Occupation']],

left\_on='UserID',right\_on='UserID')

Master\_data\_Genre.head(2)

|  | **MovieID** | **Title** | **Genre** | **UserID** | **Rating** | **Gender** | **Age** | **Occupation** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 1 | Toy Story (1995) | Animation|Children's|Comedy | 1 | 5 | F | 1 | 10 |
| **1** | 48 | Pocahontas (1995) | Animation|Children's|Musical|Romance | 1 | 5 | F | 1 | 10 |

Master\_data\_Genre.dropna(inplace = True)

MSGenre\_temp = Master\_data\_Genre["Genre"].str.split("|", n = -1, expand = True)

MSGenre\_temp.head()

|  | **0** | **1** | **2** | **3** | **4** | **5** |
| --- | --- | --- | --- | --- | --- | --- |
| **0** | Animation | Children's | Comedy | None | None | None |
| **1** | Animation | Children's | Musical | Romance | None | None |
| **2** | Drama | None | None | None | None | None |
| **3** | Action | Adventure | Fantasy | Sci-Fi | None | None |
| **4** | Drama | War | None | None | None | None |

Master\_data\_Genre["Genre1"]= MSGenre\_temp[0]

Master\_data\_Genre["Genre2"]= MSGenre\_temp[1]

Master\_data\_Genre["Genre3"]= MSGenre\_temp[2]

Master\_data\_Genre["Genre4"]= MSGenre\_temp[3]

Master\_data\_Genre["Genre5"]= MSGenre\_temp[4]

Master\_data\_Genre["Genre6"]= MSGenre\_temp[5]

Master\_data\_Genre.drop(columns =["Genre"], inplace = True)

Master\_data\_Genre.head()

|  | **MovieID** | **Title** | **UserID** | **Rating** | **Gender** | **Age** | **Occupation** | **Genre1** | **Genre2** | **Genre3** | **Genre4** | **Genre5** | **Genre6** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | 1 | Toy Story (1995) | 1 | 5 | F | 1 | 10 | Animation | Children's | Comedy | None | None | None |
| **1** | 48 | Pocahontas (1995) | 1 | 5 | F | 1 | 10 | Animation | Children's | Musical | Romance | None | None |
| **2** | 150 | Apollo 13 (1995) | 1 | 5 | F | 1 | 10 | Drama | None | None | None | None | None |
| **3** | 260 | Star Wars: Episode IV - A New Hope (1977) | 1 | 4 | F | 1 | 10 | Action | Adventure | Fantasy | Sci-Fi | None | None |
| **4** | 527 | Schindler's List (1993) | 1 | 5 | F | 1 | 10 | Drama | War | None | None | None | None |

U1 = Master\_data\_Genre.Genre1.unique()

U2 = Master\_data\_Genre.Genre2.unique()

U3 = Master\_data\_Genre.Genre3.unique()

U4 = Master\_data\_Genre.Genre4.unique()

U5 = Master\_data\_Genre.Genre5.unique()

U6 = Master\_data\_Genre.Genre6.unique()

UF = np.concatenate((U1,U2,U3,U4,U5,U6))

UF = list(filter(None, UF))

np.unique(UF)

array(['Action', 'Adventure', 'Animation', "Children's", 'Comedy',

'Crime', 'Documentary', 'Drama', 'Fantasy', 'Film-Noir', 'Horror',

'Musical', 'Mystery', 'Romance', 'Sci-Fi', 'Thriller', 'War',

'Western'], dtype='<U11')

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

## Average Rating per Movie

Master\_data\_Genre.head()

Ratings\_mean\_count = pd.DataFrame(Master\_data\_Genre.groupby('Title')['Rating'].mean())

Ratings\_mean\_count.head()

|  | **Rating** |
| --- | --- |
| **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 |

## Number of Ratings for a Movie

Ratings\_mean\_count['Rating\_counts'] = pd.DataFrame(Master\_data\_Genre.groupby('Title')['Rating'].count())

Ratings\_mean\_count.head()

|  | **Rating** | **Rating\_counts** |
| --- | --- | --- |
| **Title** |  |  |
| **$1,000,000 Duck (1971)** | 3.027027 | 37 |
| **'Night Mother (1986)** | 3.371429 | 70 |
| **'Til There Was You (1997)** | 2.692308 | 52 |
| **'burbs, The (1989)** | 2.910891 | 303 |
| **...And Justice for All (1979)** | 3.713568 | 199 |

## plotting a histogram for number of ratings

import matplotlib.pyplot as plt

import seaborn as sns

sns.set\_style('dark')

%matplotlib inline

plt.figure(figsize=(8,6))

plt.rcParams['patch.force\_edgecolor'] = True

Ratings\_mean\_count['Rating\_counts'].hist(bins=50)

## Plotting a histogram for average rating

plt.figure(figsize=(8,6))

plt.rcParams['patch.force\_edgecolor'] = True

Ratings\_mean\_count['Rating'].hist(bins=50)

## Average rating vs Number of Rating

plt.figure(figsize=(8,6))

plt.rcParams['patch.force\_edgecolor'] = True

sns.jointplot(x='Rating', y='Rating\_counts', data=Ratings\_mean\_count, alpha=0.4)

**## The graph above shows that, in general, movies with higher average ratings actually have more number of ratings, compared with movies that have lower average ratings.**