# import pandas library

import pandas as pd

# Get the data

column\_names = ['user\_id', 'item\_id', 'rating', 'timestamp']

path = '/content/file.tsv'

df = pd.read\_csv(path, sep='\t', names=column\_names)

# Check the head of the data

df.head()

# Check out all the movies and their respective IDs

movie\_titles = pd.read\_csv('/content/Movie\_Id\_Titles.csv')

movie\_titles.head()

data = pd.merge(df, movie\_titles, on='item\_id')

data.head()

# Calculate mean rating of all movies

data.groupby('title')['rating'].mean().sort\_values(ascending=False).head()

# Calculate count rating of all movies

data.groupby('title')['rating'].count().sort\_values(ascending=False).head()

# creating dataframe with 'rating' count values

ratings = pd.DataFrame(data.groupby('title')['rating'].mean())

ratings['num of ratings'] = pd.DataFrame(data.groupby('title')['rating'].count())

ratings.head()

import matplotlib.pyplot as plt

import seaborn as sns

sns.set\_style('white')

%matplotlib inline

# plot graph of 'num of ratings column'

plt.figure(figsize =(10, 4))

ratings['num of ratings'].hist(bins = 70)

# Sorting values according to

# the 'num of rating column'

moviemat = data.pivot\_table(index ='user\_id',

      columns ='title', values ='rating')

moviemat.head()

ratings.sort\_values('num of ratings', ascending = False).head(10)

# analysing correlation with similar movies

starwars\_user\_ratings = moviemat['Star Wars (1977)']

liarliar\_user\_ratings = moviemat['Liar Liar (1997)']

starwars\_user\_ratings.head()

# analysing correlation with similar movies

similar\_to\_starwars = moviemat.corrwith(starwars\_user\_ratings)

similar\_to\_liarliar = moviemat.corrwith(liarliar\_user\_ratings)

corr\_starwars = pd.DataFrame(similar\_to\_starwars, columns =['Correlation'])

corr\_starwars.dropna(inplace = True)

corr\_starwars.head()

# Similar movies like starwars

corr\_starwars.sort\_values('Correlation', ascending = False).head(10)

corr\_starwars = corr\_starwars.join(ratings['num of ratings'])

corr\_starwars.head()

corr\_starwars[corr\_starwars['num of ratings']>100].sort\_values('Correlation', ascending = False).head()