Daniel Douglass

423276

**Final Project**

***Exploring Movie Ratings and Users***

**Project Goal**

I have always been a huge fan of watching movies and I am an avid user of IMDb, always checking ratings for a movie before watching one. I am very interested in what characteristics influence whether or not someone enjoys a movie and how to recommend movies based on previous movies enjoyed or dislike by a person. The goal of this project is to examine what kinds of people like certain movies and does this affect how harshly they are to critique a movie. I want to examine occupation, gender, age, location, salary and other metrics to see how they correlate with rating behavior. I want to know if certain States like certain genres, a certain genres correlated with higher ratings or if income influences how critical a person is on rating a movie. Those and other questions regarding movie preference will be investigated in this project.

**Data**

The data I will be using will come from a variety of sources. Basic movie data, user data and user ratings will come from a Movie Lens data set. This data set is comprised of a movie table with 3,883 entries, a user table consisting of approximately 6,500 users and a rating table consisting of approximately 600,000 individual ratings. In order to get more information about the movies such as budget, runtime etc, I will be using the tmdb api that is able to pull such information. I will loop through the movies given from the movie lens dataset and pull corresponding attributes from the tmdb database. In addition, I will be matching up users zip codes to states and cities. To get the zip code dataset I will be using a zip code dataset from the federal government. In addition, I will be using an income by occupation dataset from the Bureau of Labor Statistics. All the data needs to have a specific format to be inserted into a postgres database. I will be using python to reformat and clean the data in order to have consistency across data from different sources.

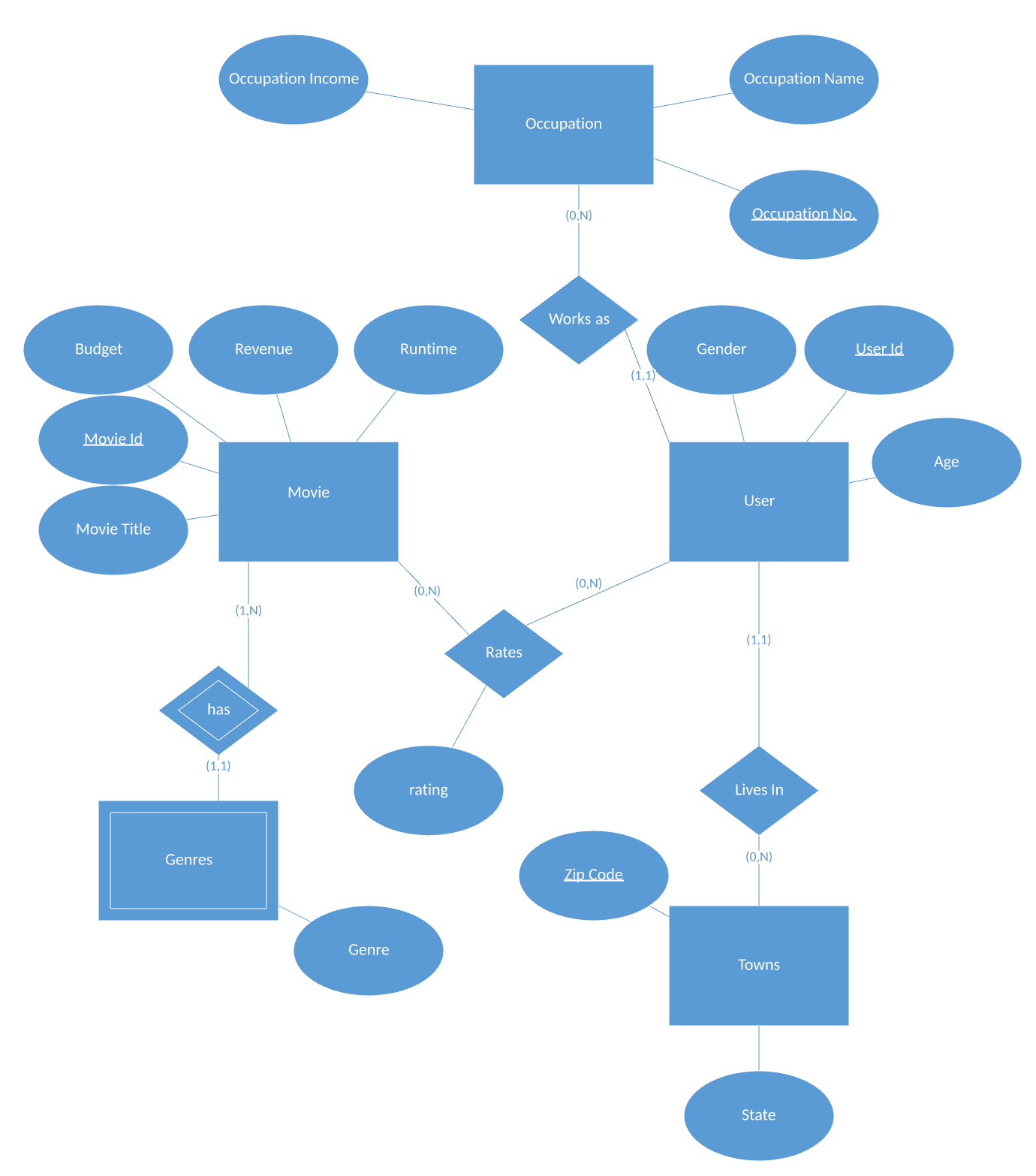
**Attributes**

The attributes that I will be using are MovieId (~3,800 distinct ids) , Movie Title, Genre, movie budget, movie runtime, movie revenue, UserId(~6,000), Gender, Age, Occupation, Zip-code, Rating, State, Occupation-title, mean income. The only attribute that may change of time is the revenue attribute. This attribute is very easy to update as if the revenue changes for a movie it only corresponds to one entry in the movie table. These attributes will be significant in helping to create a recommendation system and find patterns between users and movie preferences.

**Queries**

The queries I would use would be to join my user table with the zip code table and occupation table to get a city, state and income for each user. The zip code table would be joined on a users zip code and the occupation table will be joined on the key for a users occupation. From there I would be able to join on the ratings table and movie table to get ratings per person along with the attributes from the movie table. This would allow me to use many variations of group by statements to get desired results. I would want to group by state and get average ratings per state. Also I would make another query that would group by state, genre to get average rating per genre for each state. Then I would get the max of the average ratings per genre to see which genres are most popular/get the best ratings. These are a few queries I will run along with other queries that will give insight to the movie data I have collected.

**Database ER Diagram**



**Exploratory Data Analysis of Movie, Ratings and User Data**

The goal of this project is to examine what factors influence rating behavior and the ratings of movies across a variety of variables. The following analysis will show trends in the data in how location, income, age, and gender of a user may contribute to how critical users are when rating a movie. In addition, certain characteristics of movie will be examined such as runtime, revenue and genre to see if there are correlations between these variables and how highly rated that movie is.

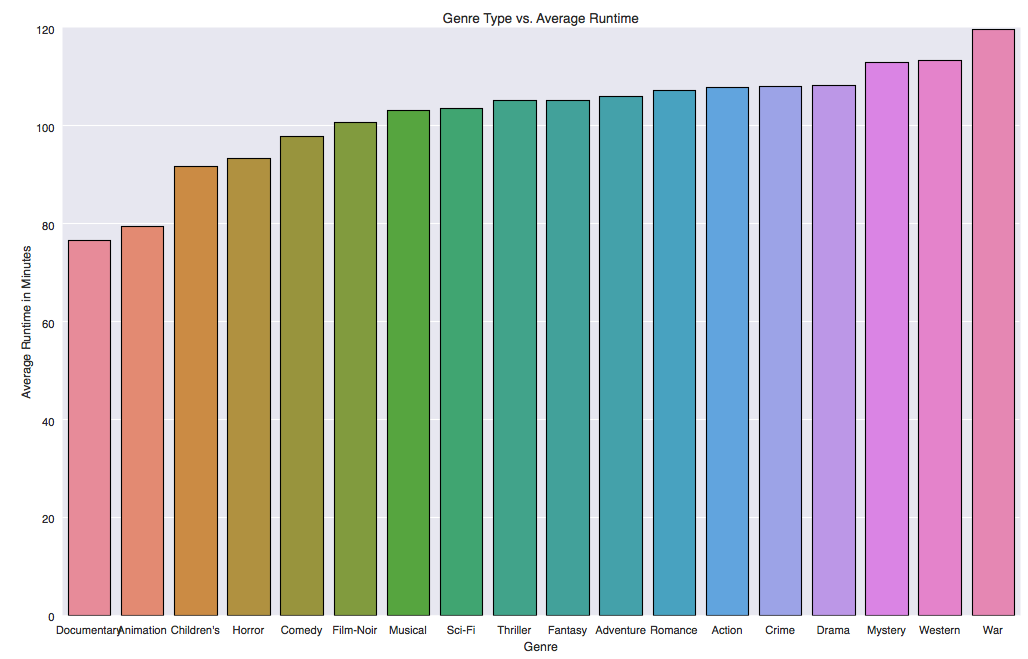
**Pair Plot of Movie Characteristics - Runtime, Revenue, Budget, number of Rating of the movie, Average Rating (All movies included had at least 1,000 ratings)**



The dots in each plot represent individual movies with their respective variables. The Average Rating vs. numberOfRatings scatterplot shows an upward trend that as the more ratings a movie has the more likely it will be scored higher. This is expected, as popular movies should be good movies in general. Also in the top right corner we can see Average Rating vs. Runtime. This plot shows that longer movies in general seem to have a better rating than shorter movies. This may be due to longer movies being able to captivate the audience with a detailed in depth plot. This may contribute to high ratings in comparison to a comedy movie, which is generally short, and has a less involved plot line. Users may be more inclined to rate a movie higher if the story line is more complex. A third plot of interest is the revenue vs. average rating. This plot show no apparent trend between revenue and Average rating, this is very interesting as it may show box office numbers are not necessary correlated with how good a movie actually is.

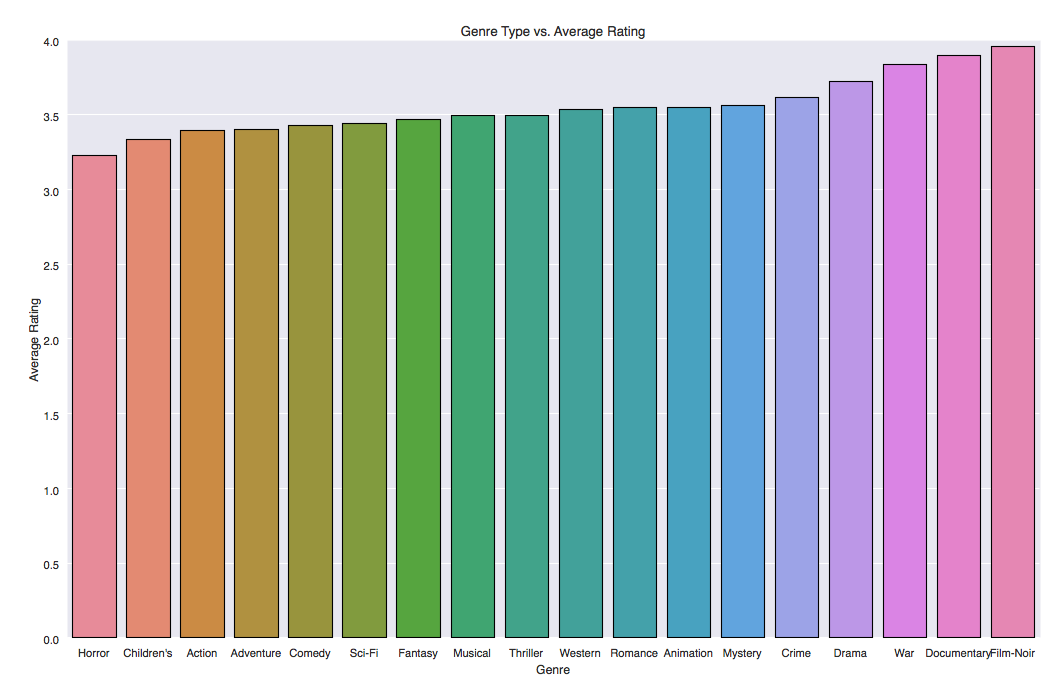
The next two graphs will explore the relationship between runtime and average rating.

**Average Runtime of Genres**



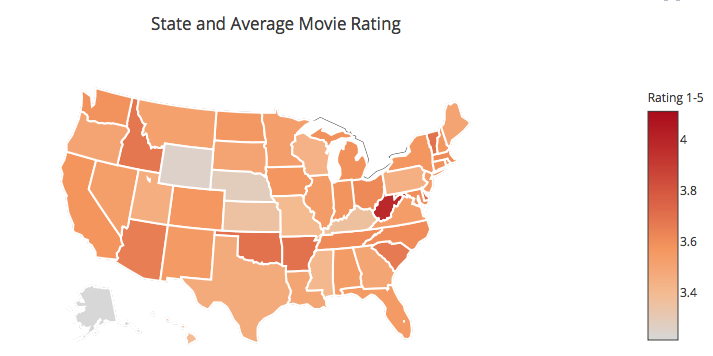
The above shows the average runtime of movies dependent on genre. Movies that are more focused on a plot line such as War, Drama, and Mystery have the longest average runtimes, while lighter movies such as comedies and children’s film are significantly shorter (~30 minutes). The only exceptions to this general trend are documentary movies, which are the shortest along with animation movies.

**Average Rating of Movies By Genre**

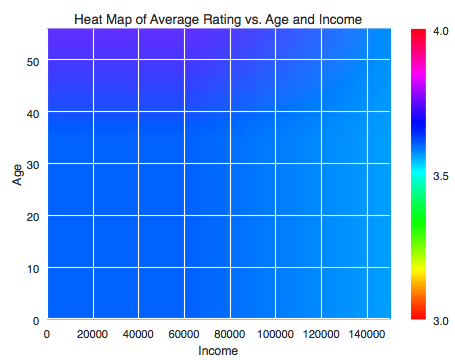


Comparing the two graphs above we see that the longer movies generally have much higher ratings than the short movies. War, drama, Crime and Mystery all score high on both average rating and runtime while shorter movies such as Horror, children’s, and comedy score lower. The most interesting genre is documentaries. As a genre documentaries are by far the shortest but achieve a very high average score. The graphs above demonstrate genres that tell a story with a purpose or meaning generate the highest ratings. Documentaries, War and Dramas all have this trait in common. They focus on content rather than laughs, screams or action such as the genres with lower average ratings. Runtime may indicate certain non-superficial aspects of a movie affecting the ratings.

**Average Movie Rating vs. US States**

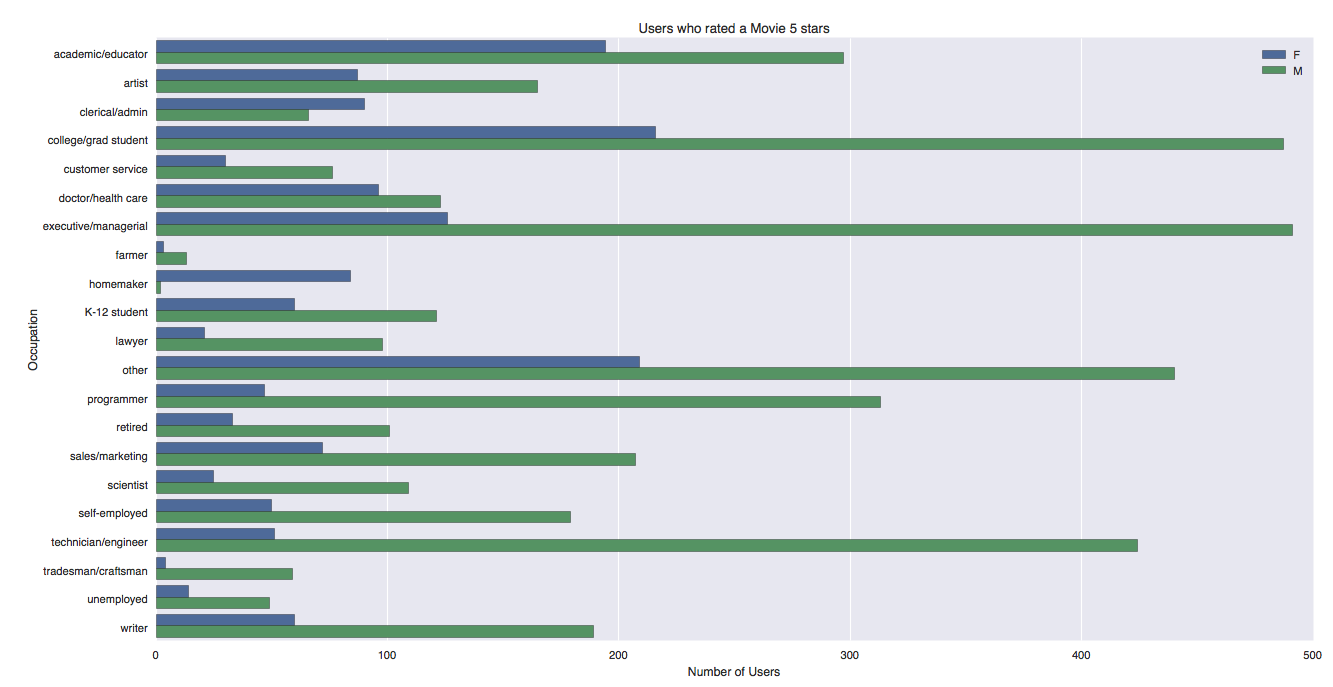


Above shows the average rating movies dependent on state. It is shown that people in West Virginia are most likely to give a higher average rating (3.92) to a movie than someone in Alaska or Nebraska, which has an average rating of 3.2. The place where someone lives could affect his or her penchant for scoring a movie higher. Their place of residency could be raising more generous types of people or less critical people. The reported happiness levels of states would be a good indicator if this hypothesis could be a factor in this result.

**Average Rating vs. Age vs. Income Level**

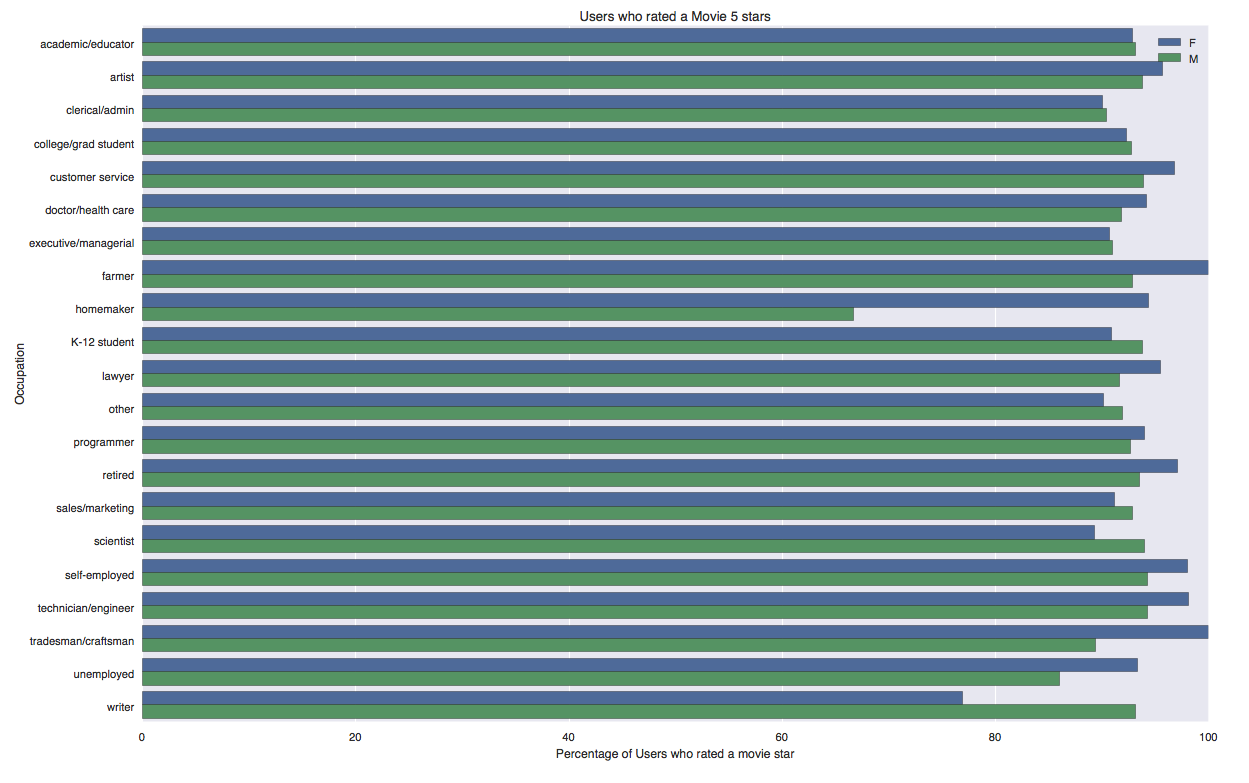
The above graph shows a heat map of age vs. income vs. average rating. The average rating is denoted by the color on the graph. The difference between the average rating and these variables is very subtle. As we move higher on the income scale the shade of blue gets slightly lighter indicating that higher incomes usually result in lower average ratings, but only by a small margin. This may suggest that wealthier people may be more critical of movies in comparison to students or unemployed people. Also as age increases the shade of blue gets darker. This may be that retired older people are not very critical of movies or are more generous thereby more likely to give out higher ratings. This may be that with more free time comes less care or being less critical of things as opposed to wealthier people who on average may work more and have less free time to watch movies.

**Number of Users Who Rated a Movie 5 Stars by Gender and Occupation**



The above graph shows the number of users who rated a movie 5 stars breaking up those users by occupation and gender. It is obvious that the number of users rating movies is of a majority male. In all occupations except administration and Homemaker the number of males exceed the number of females who rated a movie 5 stars. To further understand if occupation has any effect on a user giving a higher rating the percentages of occupation and genders rating a movie 5 stars can be investigated.

**Number of Users Who Rated a Movie 5 Stars by Gender and Occupation**



Shown above we see that the majority of people approximately 90% decide to rate at least one movie they see 5 stars. In addition due to all of the percentages of users rating 5 stars are consistent, we cannot really determine if occupation has an effect on users rating a movie high. We do see that the smallest percentage of people rating movies 5 stars is homemaking males. But because the number of homemaking males is so low, we cannot make any conclusions about this finding.

**Conclusion**

The findings from the exploratory data analysis of the movie, ratings and users data sets are not extremely apparent. Due to the large volume of data, the average ratings across a variety of variables do not have a large variance. Thus we do not seem significant differences when changing our independent variables such as genre, income, age etc. to identify trends in the average ratings of movies. This may affect of impression of certain influences on the ratings of movies. The most significant trend seen is plotting the average rating per state. It shows a large difference between the lowest rating states and highest rating states. The majority of the states were in the middle with no apparent trends between them. In conclusion there are certain factors that may influence the rating of a movie, but more in depth analysis is needed to make conclusive findings.

**Appendix**

**Database Setup Code:**

#SQL create database

CREATE DATABASE daniel\_douglass\_moviedb;

REVOKE ALL on DATABASE daniel\_douglass\_moviedb FROM PUBLIC;

CREATE TABLE location(

zipcode varchar(10) PRIMARY KEY,

state varchar(2) NOT NULL

);

CREATE TABLE occupation(

occupationId integer PRIMARY KEY,

occupationName varchar(40),

income integer

);

CREATE TABLE movies(

movieId integer PRIMARY KEY,

movieTitle varchar(100) NOT NULL,

release\_date date,

runtime float,

budget float,

revenue float

);

CREATE TABLE users(

userId integer PRIMARY KEY,

gender varchar(1),

age integer,

occupationId integer references occupation(occupationId),

zipcode varchar(10)

);

CREATE TABLE ratings(

userId integer references users(userId),

movieId integer references movies(movieId),

rating float NOT NULL

);

CREATE TABLE genres(

movieId integer references movies(movieId),

genre varchar(40) NOT NULL

);

#in Xquartz:

scp moviesTable.csv daniel.douglass@postgres.cec.wustl.edu:~/

scp genresTable.csv daniel.douglass@postgres.cec.wustl.edu:~/

scp occupations.csv daniel.douglass@postgres.cec.wustl.edu:~/

scp zipcodclean.csv daniel.douglass@postgres.cec.wustl.edu:~/

scp users.csv daniel.douglass@postgres.cec.wustl.edu:~/

scp ratings.csv daniel.douglass@postgres.cec.wustl.edu:~/

#in psql

\COPY movies (movieId, movieTitle, release\_date,runtime,budget,revenue) FROM '~/moviesTable.csv' DELIMITER ',';

\COPY genres (movieId, genre) FROM '~/genresTable.csv' DELIMITER ',';

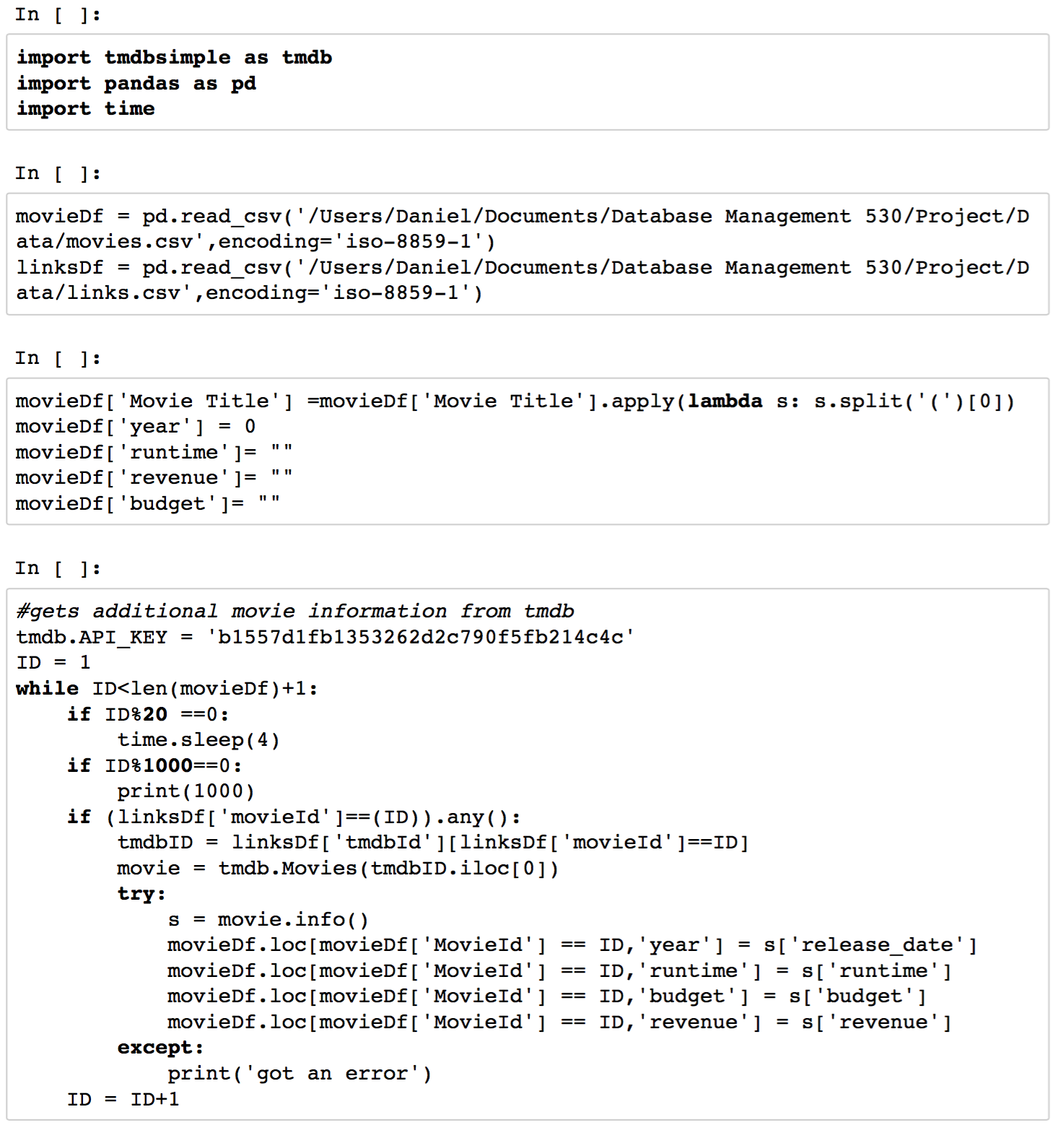
\COPY occupation (occupationId, occupationName,income) FROM '~/occupations.csv' DELIMITER ',';

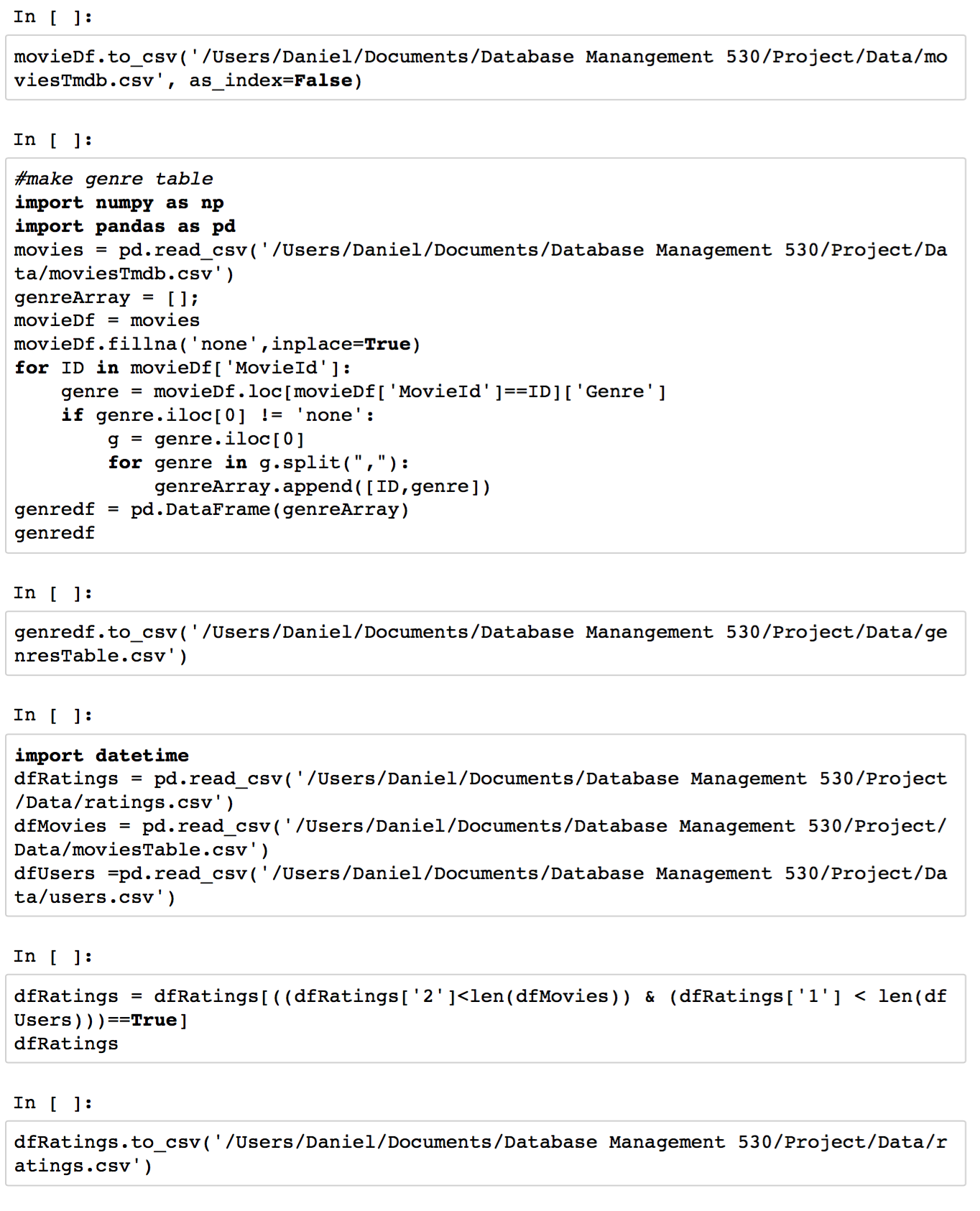
\COPY location (zipcode, state) FROM '~/zipcodeclean.csv' DELIMITER ',';

\COPY users (userId, gender,age,occupationId,zipcode) FROM '~/users.csv' DELIMITER ',';

\COPY ratings (userId,movieId,rating) FROM '~/ratings.csv' DELIMITER ',';

**Data Scraping and Cleaning Script:**

****



**Query and Graphing Code:**

**from** sqlalchemy **import** create\_engine  
**import** psycopg2  
**import** plotly.plotly **as** py  
**import** plotly  
**from** plotly.graph\_objs **import** \*  
**import** pandas **as** pd  
**import** numpy **as** np  
**import** matplotlib.pyplot **as** plt  
**from** matplotlib.image **import** NonUniformImage  
**from** matplotlib **import** cm  
**import** matplotlib  
**import** seaborn **as** sns  
  
plotly.tools.set\_credentials\_file(username=**'DanielDouglass'**, api\_key=**'4m3kqfrwh5'**)  
  
params = {  
 **'database'**: **'daniel\_douglass\_moviedb'**,  
 **'user'**: **"daniel.douglass"**,  
 **'password'**: **"R3d$ox2004"**,  
 **'host'**:**'127.0.0.1'**,  
 **'port'**:5432  
}  
conn = psycopg2.connect(\*\*params)  
curs = conn.cursor()  
  
**print "database connected"**curs.execute(**'''select avg(rating) as rate, state  
 from  
 ratings  
 left join  
 (select userId, state  
 from users, location  
 where users.zipcode = location.zipcode) as a  
 on  
 ratings.userId = a.userId  
 group by state order by rate desc;'''**)  
d = curs.fetchall()  
df = pd.DataFrame(d)  
df.columns = [**'Ratings'**,**'State'**]  
data = [dict(  
 type=**'choropleth'**,  
 autocolorscale=True,  
 locations=df[**'State'**],  
 z=df[**'Ratings'**],  
 locationmode=**'USA-states'**,  
 marker=dict(  
 line=dict(  
 color=**'rgb(255,255,255)'**,  
 width=2  
 )),  
 colorbar=dict(  
 title=**"Rating 1-5"**)  
)]  
  
layout = dict(  
 title=**'State and Average Movie Rating'**,  
 geo=dict(  
 scope=**'usa'**,  
 projection=dict(type=**'albers usa'**),  
 showlakes=True,  
 lakecolor=**'rgb(255, 255, 255)'**),  
)  
  
fig = dict( data=data, layout=layout )  
py.iplot( fig, filename='d3-cloropleth-map' )curs.execute(**'''select rate, genre, c.state  
 from (  
 select avg(rating) as rate, genre, a.state  
 from  
 ratings  
 left join  
 (select userId, state  
 from users, location  
 where users.zipcode = location.zipcode) as a  
 on  
 ratings.userId = a.userId  
 left join  
 genres  
 on  
 genres.movieId = ratings.movieId  
 WHERE  
 genres.genre IS NOT NULL  
 and state IS NOT NULL  
 group by state, genre) as f  
 inner join  
 (select max(rate) as mrate, b.state  
 from (  
 select avg(rating) as rate, genre, d.state  
 from  
 ratings  
 left join  
 (select userId, state  
 from users, location  
 where users.zipcode = location.zipcode) as d  
 on  
 ratings.userId = d.userId  
 left join  
 genres  
 on  
 genres.movieId = ratings.movieId  
 WHERE  
 genres.genre IS NOT NULL  
 and state IS NOT NULL  
 group by state, genre) as b  
 group by b.state) as c  
 on c.state = f.state  
 and c.mrate =f.rate ;'''**)  
  
  
d = curs.fetchall()  
df2 = pd.DataFrame(d)  
df2.columns = [**'Ratings'**,**'genre'**,**'State'**]  
  
curs.execute(**'''select age, income, avg(rating)  
 from  
 ratings  
 left join  
 users  
 on  
 ratings.userId = users.userId  
 join  
 occupation  
 on  
 occupation.occupationId = users.occupationId  
 group by age,income;  
'''**)  
  
  
d = curs.fetchall()  
df = pd.DataFrame(d)  
df.columns = [**'age'**,**'income'**,**'Rating'**]  
ratings = df.pivot\_table(**'Rating'**, **'age'**, **'income'**)  
ratings.fillna(3.5,inplace=True)  
  
  
  
fig = plt.figure()  
interp = **'bilinear'**ax = fig.add\_subplot(221)  
norm = matplotlib.colors.Normalize(vmin=3.0, vmax=4.0, clip=False)  
im = NonUniformImage(ax, norm=norm, interpolation=interp, extent=(0, 170000, 0, 60),  
 cmap=cm.hsv)  
  
im.set\_data(df[**'income'**].unique(), df[**'age'**].unique(), ratings)  
ax.images.append(im)  
ax.set(xlabel=**'Income'**,ylabel = **'Age'**, title=**'Heat Map of Average Rating vs. Age and Income'**)  
ax.set\_xlim(0, 150000)  
ax.set\_ylim(0, 56)  
  
cbar = fig.colorbar(im, ticks=[3, 3.5, 4,4.5])  
plt.show()  
  
  
curs.execute(**''' select avg(runtime) as runtime, genre  
 from genres  
 left join  
 movies  
 on genres.movieId = movies.movieId  
 group by genre order by runtime;  
'''**)  
  
  
d = curs.fetchall()  
df = pd.DataFrame(d)  
df.columns = [**'Average Runtime'**,**'Genre'**]  
ax = sns.barplot(x = df[**'Genre'**],y = df[**'Average Runtime'**], linewidth=1)  
ax.set(xlabel=**'Genre'**,ylabel = **'Average Runtime in Minutes'**, title=**'Genre Type vs. Average Runtime'**)  
sns.plt.show()  
  
curs.execute(**''' select avg(rating) as rating, genre  
 from ratings  
 left join  
 movies  
 on ratings.movieId = movies.movieId  
 left join  
 genres  
 on  
 genres.movieId = movies.movieId  
 group by genre order by rating;  
'''**)  
  
  
d = curs.fetchall()  
dfA = pd.DataFrame(d)  
dfA.columns = [**'Average Rating'**,**'Genre'**]  
ax = sns.barplot(x = dfA[**'Genre'**],y = dfA[**'Average Rating'**], linewidth=1)  
ax.set(xlabel=**'Genre'**,ylabel = **'Average Rating'**, title=**'Genre Type vs. Average Rating'**)  
sns.plt.show()  
  
  
  
curs.execute(**''' select avg(rating) as rating, gender, genre  
 from ratings  
 left join  
 users  
 on  
 users.userId = ratings.userId  
 left join  
 movies  
 on ratings.movieId = movies.movieId  
 left join  
 genres  
 on  
 genres.movieId = movies.movieId  
 group by genre, gender order by rating;  
'''**)  
d = curs.fetchall()  
dfMF = pd.DataFrame(d)  
dfMF.columns = [**'Average Rating'**, **'Gender'**,**'Genre'**]  
  
ax = sns.barplot(x = dfMF[**'Genre'**],y = dfMF[**'Average Rating'**],hue=dfMF[**'Gender'**], linewidth=1)  
ax.set(xlabel=**'Genre'**,ylabel = **'Average Rating'**, title=**'Genre vs. Average Rating by Gender'**)  
sns.plt.show()  
  
*#pairplot of attributes based on movieId*curs.execute(**''' select movies.movieId, runtime,revenue,budget, avg(rating)  
 from ratings  
 left join  
 movies  
 on ratings.movieId = movies.movieId  
 where  
 runtime>0  
 and budget >1000  
 group by movies.movieId;  
'''**)  
  
  
d = curs.fetchall()  
df = pd.DataFrame(d)  
df.columns = [**'MovieId'**,**'Runtime'**,**'revenue'**,**'budget'**,**'Average Rating'**]  
df.drop(**'MovieId'**,inplace=True,axis=1)  
ax = sns.pairplot(df)  
sns.plt.show()  
  
  
  
  
curs.execute(**''' select movies.movieId, runtime,revenue,budget, a.numberOfRatings, avg(rating)  
 from ratings  
 left join  
 (select movieId, count(\*) as numberOfRatings from ratings  
 group by movieId) as a  
 on  
 a.movieId = ratings.movieId  
 left join  
 movies  
 on  
 ratings.movieId = movies.movieId  
 where  
 runtime>0  
 and budget >1000  
 and numberOfRatings>1000  
 group by movies.movieId, a.numberOfRatings;  
'''**)  
  
d = curs.fetchall()  
df = pd.DataFrame(d)  
df.columns = [**'MovieId'**,**'Runtime'**,**'revenue'**,**'budget'**,**'numberOfRatings'**,**'Average Rating'**]  
df.drop(**'MovieId'**,inplace=True,axis=1)  
ax = sns.pairplot(df)  
sns.plt.show()  
  
  
conn = psycopg2.connect(\*\*params)  
curs = conn.cursor()  
  
  
curs.execute(**'''select rat.\*,100.0\*cast(rat.numUsers as float)/cast(totalUsers.total as float) from  
(select count(distinct ratings.userId) as numUsers, gender, occupationName  
 from ratings  
 left join  
 users  
 on  
 users.userId = ratings.userId  
 join  
 occupation  
 on  
 users.occupationId = occupation.occupationId  
 left join  
 movies  
 on ratings.movieId = movies.movieId  
 where rating = 5  
 group by gender, occupationName) as rat  
 join  
 (select count(distinct users.userId) as total, gender, occupationName  
 from users  
 left join  
 occupation  
 on  
 occupation.occupationId = users.OccupationId  
 group by gender, occupationName) as totalUsers  
 on  
 rat.gender = totalUsers.gender  
 and rat.occupationName = totalUsers.occupationName  
'''**)  
  
d = curs.fetchall()  
df = pd.DataFrame(d)  
df.columns = [**'numberOfPeople'**,**'gender'**,**'occupationName'**,**'ratioOfUsers'**]  
ax = sns.barplot(x=df[**'ratioOfUsers'**],y = df[**'occupationName'**],hue = df[**'gender'**])  
ax.set(xlabel=**'Percentage of Users who rated a movie star'**,ylabel=**'Occupation'**, title=**'Users who rated a Movie 5 stars'**)  
ax.set\_style(**"ticks"**)  
plt.legend()  
sns.plt.show()