What Factors Affect A Movie’s Revenue Performance?

## **Team members: Xuanchen Liu, Simeng Alexandra Cai**

## Motivation

Why movies data?

Our project is analyzing movies listed in the Full MovieLens Datasets to explore the factors that influence a movie’s box office performance in the worldwide scale.

Movies are one of the most popular forms of entertainment and make us learn something. However, factors that contribute to a movie’s success (we define a movie’s success mainly as its revenue in this scenario) are not that familiar with all audiences. Both of our team members are film enthusiasts and are interested in understanding what factors can impact a movie’s revenue performance and what factors normally have more impact than others.

Our goal

Specifically, we intend to see factors -such as director gender/popularity, primary cast gender/popularity, secondary cast gender/popularity, budget, release dates, production companies, genre, etc. -that influence a movie’s revenue. In addition, among these features, we would like to explore whether one or some specific factors can have more impact on a movie’s performance than others.

## Data Source

Primary Dataset

The primary dataset will be “movies\_metadata” and “credits” datasets from Kaggle. The “movies\_metadata” includes features such as language, budget, revenue, release dates, production countries from early 20th century to the end of 2020. The “credits” consists of film cast and crew information for all movies. The “movies\_metadata” dataset is 32.85 megabytes, and the “credits” dataset is 181.12 megabytes. The Movies Datasets are in a csv format and can be downloaded at

<https://www.kaggle.com/rounakbanik/the-movies-dataset>.

Secondary Dataset

The secondary dataset is retrieved from the TMDB website <https://developers.themoviedb.org/3/people/get-person-details>  to get the people popularity feature, including the popularity for movie directors and cast. This dataset contains cast/crew id, gender, and popularity. A total of 46360 records (rows) were retrieved.

For each cast or crew member, his/her popularity number will be retrieved in a JSON format through the TMDB API. The estimated size for a single JSON file is 3kb.

## Data Manipulation Methods

### Step 1: Extract “Director Id”, “Primary Cast Id” and “Secondary Cast Id” from Credits Dataset (Primary dataset)

### -Director id of each movie is extracted from the crew column in the credits dataframe by filtering the value of “job” and setting its crew value as “Director”.  We created a function get\_director\_id to achieve this.

- Considering large amounts of cast each movie has, only Primary Cast and Secondary Cast will be retrieved in this context, which are normally taken as the most important cast and leading actors/actresses for each movie. Specifically, Primary Cast id is extracted from the cast column by filtering the value of “order” of each cast as “0”, and Secondary Cast id is extracted by filtering the value of “order” of each cast as “1”.  According to the dataframe, the lower the number, the more important the cast is. We created a function called get\_cast\_ids to achieve this goal.

-New columns director id, primary cast id and secondary cast id are created for the “credit” dataframe.

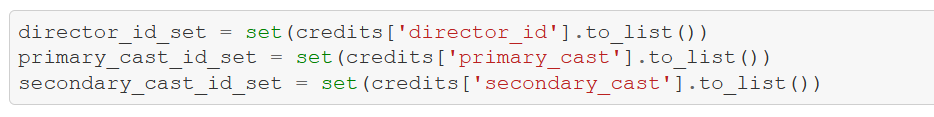
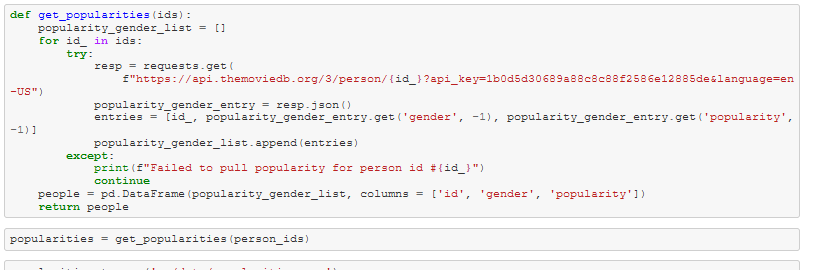




### Step2: Get popularities and Gender for Each Id from Secondary Dataset

-We created get\_popularities function to retrieve the popularities for all directors, primary cast and secondary cast based on their ids from the secondary dataset TMDB website.

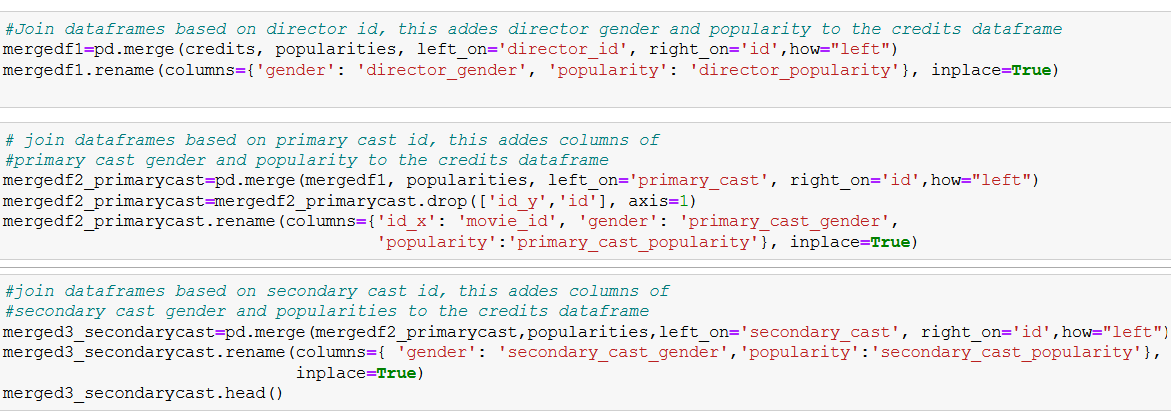
- A new dataframe “popularities” is created consisting columns of id, gender and popularity.



## Step3: Merging dataframes “credits” (Primary Dataset) and “popularities” (Secondary Dataset) Based on their common column “Id”

-We merged two dataframes—“credits” and “popularities” based on their common ids of directors, primary cast and secondary cast.

-We used left join to merge the “credits” (left) with the “popularities” (right) with director id  (for “credits” the column is called “director\_id” and for “popularities” the column is called “id”). We then got a dataframe with original data from “credits” as well as the popularity and gender information for directors from “popularities”. Similar process was applied to merge primary & secondary cast popularity and gender to the “credits”.

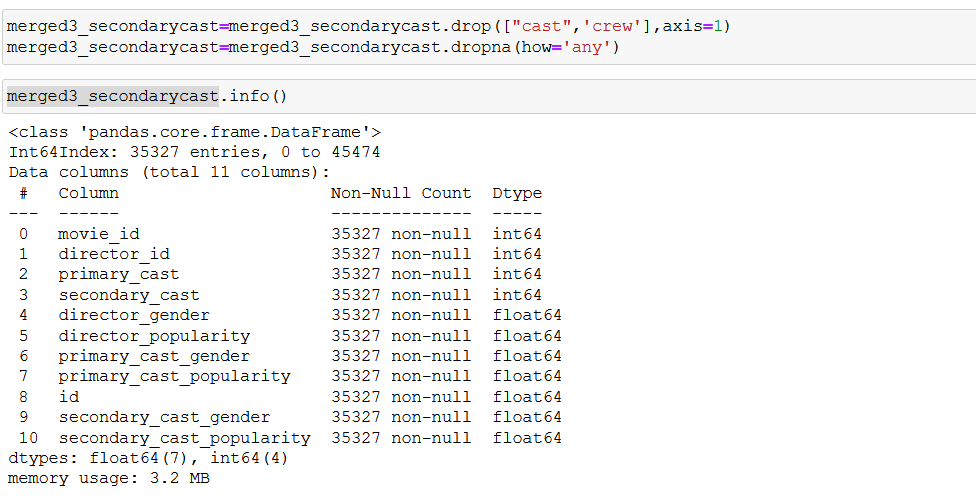


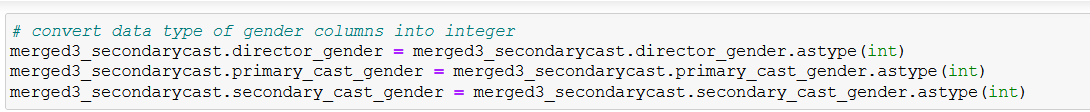
### Step4: Data Cleaning for “merged3\_secondarycast” dataframe

- Since useful information has been extracted, we will drop columns (‘cast’, and ‘crew’) that we don’t need for our analysis from “merged3\_secondarycast” we got from step 3 after merging the primary and secondary dataset.

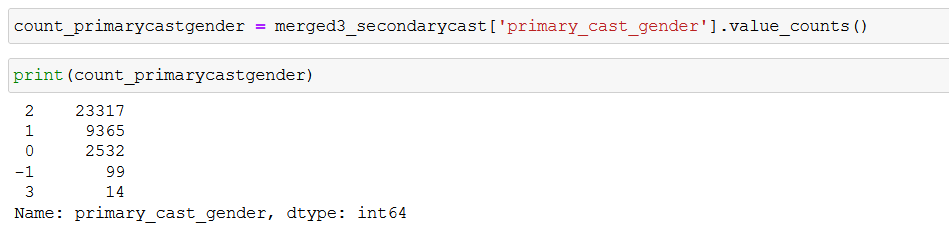
-Drop null value: we dropped rows with any null values in the dataframe using dropna().

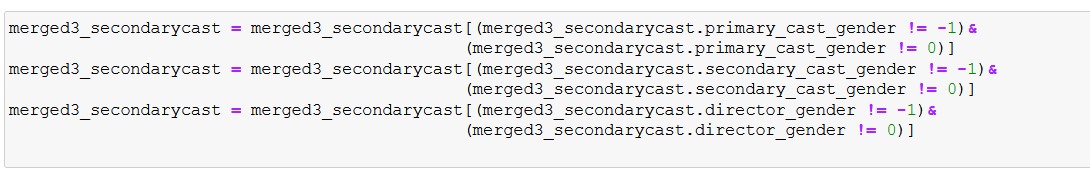
-Check data type for each column by using df.info() and use astype() to convert each column into appropriate data type.



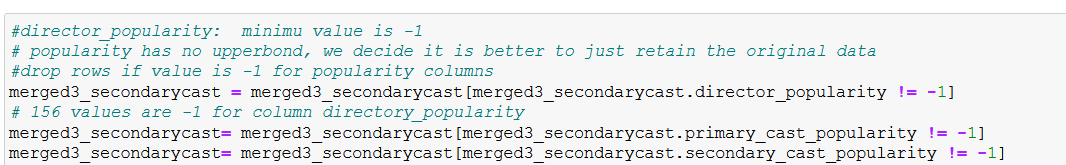


-We also used min() , max(), describe(), and value\_counts() to understand the features. We found that 1 represents female, 2 represents male, and 3 represents non-binary. We dropped rows with -1 and 0 used for null values of gender.



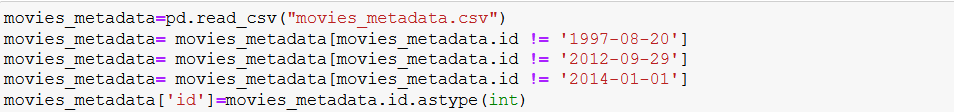


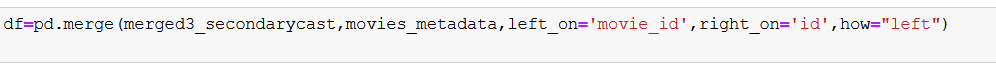
-We dropped rows for “popularity” columns that have null value of -1.



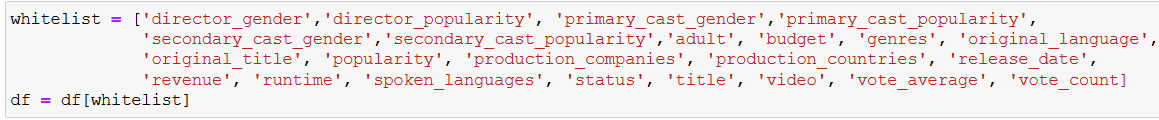
### Step 5: Merging Dateframes “mered3\_secondarycast” from step 4 and “movies\_metadata.csv” (one of the primary dataset) based on common column “movie id”.

-we cleaned the id column and converted the datatype into integer. We got error messages for values that can’t be converted because they are date-time values mistakenly input into the wrong column—therefore, astype() was used to identify invalid values for data cleaning.





### Step6: Selecting Columns(features) in “df” dataframe that we will be analyzing



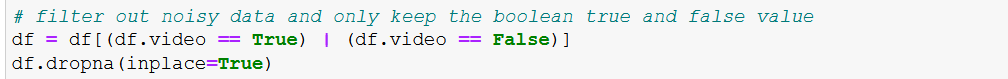
### Step7: Further Data Cleaning for “df” dataframe

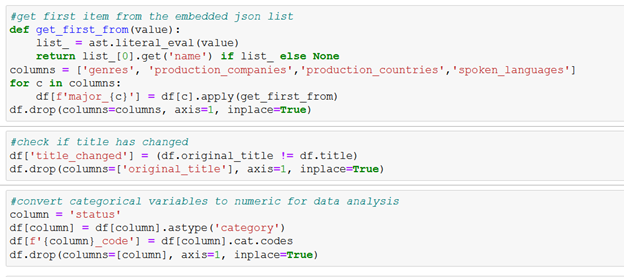
### -We filtered out noisy data for video column.

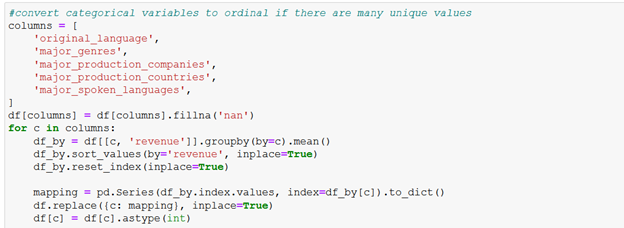
### -We created a function get\_first\_from() to get the first item from the embedded json list for columns, including “genres”, “production\_companies”, “production\_countries”, “spoken\_language” in the dataframe.

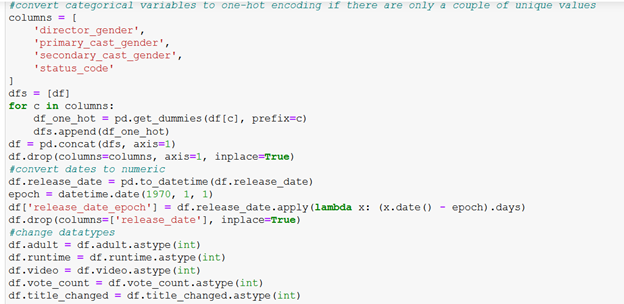
### -We converted the column values to appropriate datatype for analysis. For example, we used one-hot-encoding for categorical variables with only a couple of unique values, and converted categorical variables to ordinal forms for columns with many unique values.

### -we used min() and describe() to figure out that there is non-negative values for numerical columns and because there is no upper bound, we decided it is better to just retain the original data instead of removing outliers using z-score.









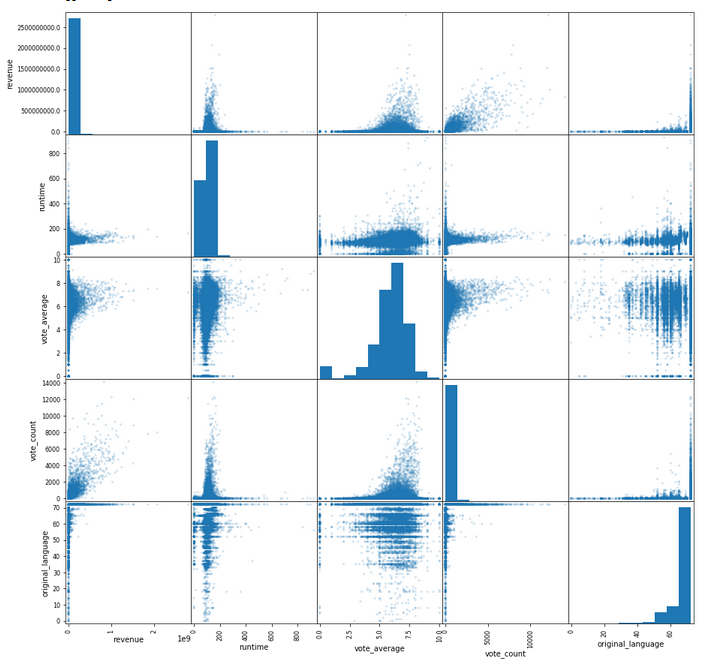
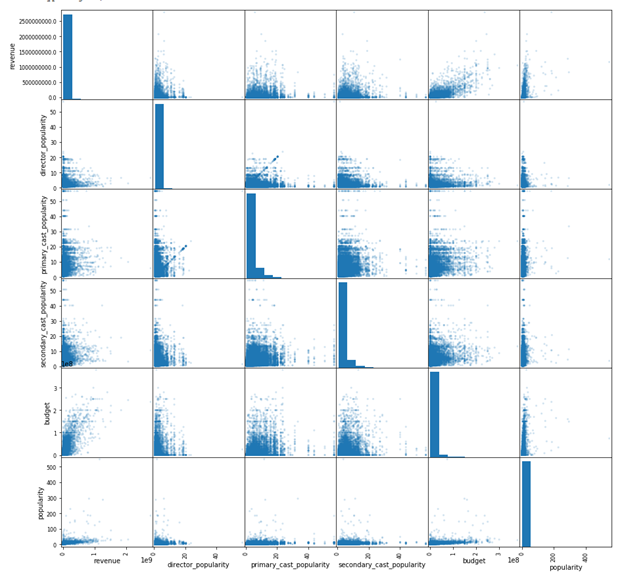
# Analysis and Visualization:

Step 1: From Figure 1 below we can see that there tends to show a linear trend between “revenue” and “budget”. At the same time, values of variables such as “director\_popularity”, “primary\_cast\_popularity”, “secondary\_cast\_poluarity” and “popularity”(representing the movie popularity) tend to get slanted to the left(these variables are right-skewed), which gives us some hints that we can do some log transformation for further exploration to see if there's any relationship between these variables and revenue.

https://lh4.googleusercontent.com/lUlW-YZzb5mxANYqLD6FjPdKlNfjr2vVigOUglVpNhG_S44qgnzxS1qclL8tLT1U-qH0jcseDM8c7uBjZRGiX2uvk9EirTc6EZMGdPaApdIiIlSs6Y414dM5BgFGAQ

#### Step2: From Figure 2 we can see a linear trend showing between “revenue” and “vote\_count”. Values of runtime tend to focus on the left side(right-skewed), which we can also do log transformation to further explore the relationship between revenue and runtime.

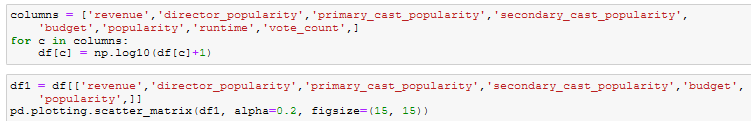
https://lh5.googleusercontent.com/hw91iAh3jeRB3nvGz43lI9qgkVMoNQqSbCUR7zTHtNr0aSDR2qrtKIf2Pusv6V8Z44myhnH2U4WdAXsg8t3_HrRPKUwz348vGUuhs7it2v2aSMibobpL6erhDQNLPA We'll take log of the following variables to explore further since we can not see any obvious regression trend at this moment.**https://lh5.googleusercontent.com/gq1Mo4a-lyatvuQGgiFU1jti2677wMag8Aw2vgfixihSPpkpIO-t9p7OOzBSDwdBhqXVGC02Gs-QUN8zA5BqmbpaaFrZkUGpTkIAsdYDu_R-v6jNztcef2Q23se9BQ**

 Figure 1 Figure 2

## Step3: Log transformation

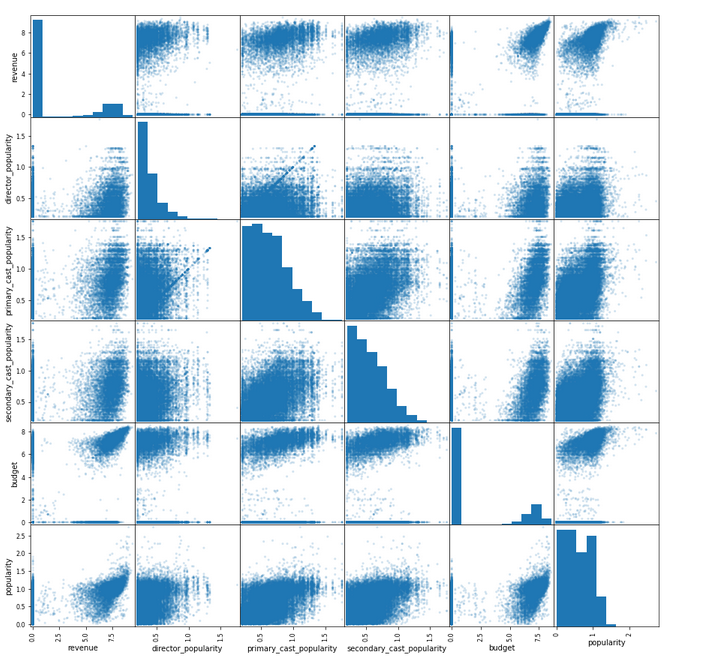
## -If the variable is right-skewed, take their log.

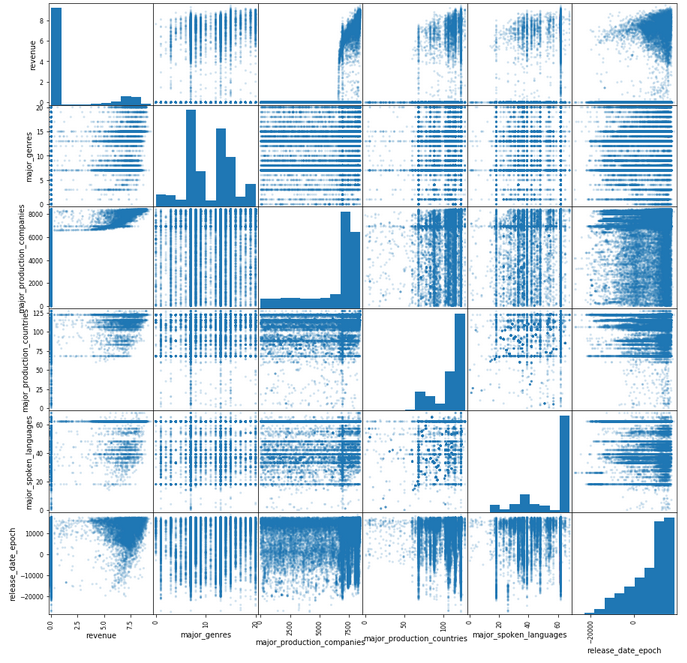
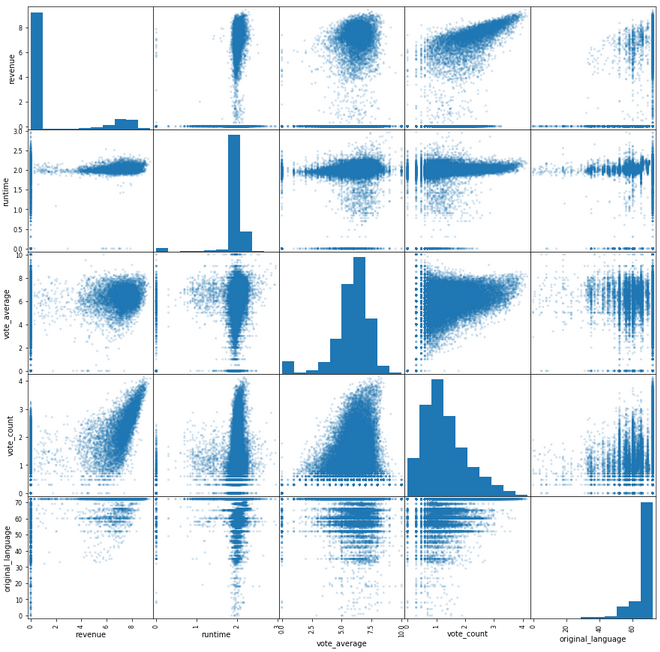
#### -After taking log, we can see some variables that seem unconnected with revenue before tend to show some linear trend, such as “director\_popularity”, “primary\_cast\_popularity”, “secondary\_cast\_popularity”, “budget”, and “popularity”. Revenue tends to increase when values of these independent variables increases. See below.



https://lh3.googleusercontent.com/eE_M2xdjnYi0BiOkYazjb1LI0rCixD2LAcJVzXL_UI7rMmt0jIU2iFtX0sJL4Ec0_R_Ai5vzhniiRaeayQT2lCe4bSdtEGK8t3bVZcdy8pPvps4Uq2Kb7HAcnQBZvQ

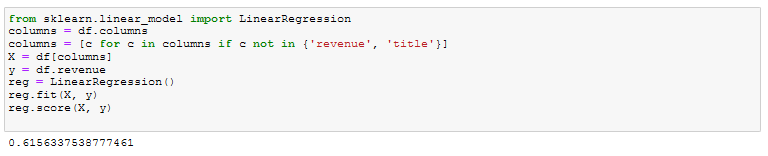
https://lh6.googleusercontent.com/3FprpSSwkDw9s0dIgmhToOWYKyiP0ErSMXU9uMHF1EDIkeRzJG8SYftRn87HvUq73SUpBIWhzUWo1olsUWrgtqwvqM3QQuxYABkB9baQjCbIdBlTV_QaFq7-bMm0IQ

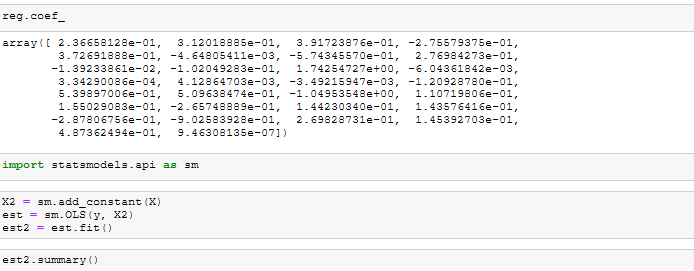




## Step4: Regression Analysis

-we used linear regression model to examine the linear relationship between independent variables such as “budget”, “popularity”, “gender” and the dependent variable “revenue”.





The summary of the OLS regression results shows the coefficient for each independent variable in the model and the corresponding p-value.

Step 5: We eliminated features with p-value being greater than 0.05 because we only want to keep variables that have a statistically significant effect on the dependent variable in our OLS regression model and we built the model again until we have a model with p-values of all features less than 0.05.

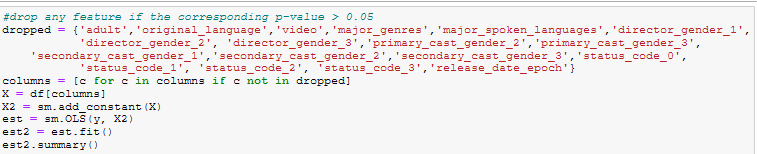
-The OLS regression results show that the model is:

Revenue=3.0063+0.2329\*director\_popularity+0.3256\*primary\_cast\_popularity+0.4087\*secondary\_cast\_popularity+0.3735\*budget-0.5639\*popularity+0.2766\*runtime-0.1023\*vote\_average+1.7337\*vote\_count+0.0003\*major\_production\_companies

-Features that have positive impact on revenue ordered from highest to lowest impact—higher coefficients mean higher impact:

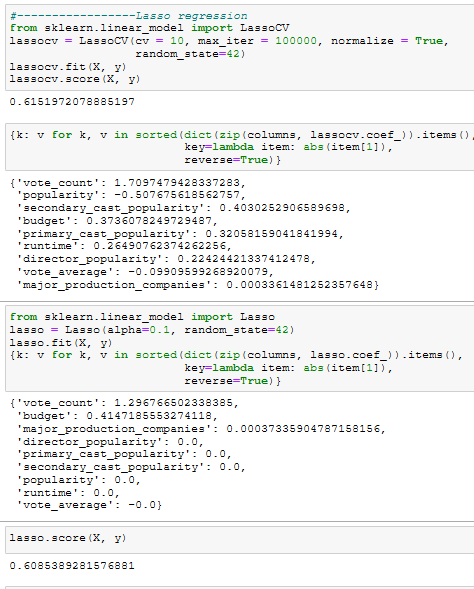
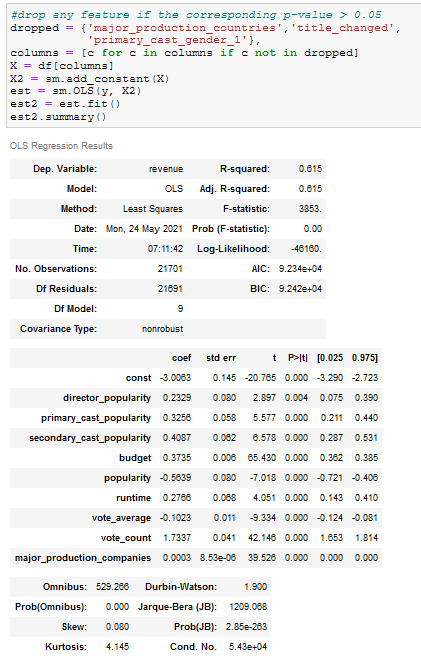
vote\_count> secondary\_cast\_popularity> budget> primary\_cast\_popularity> runtime> director\_popularity> major\_production\_companies

- Features that have negative impact on revenue: vote\_average & popularity



Step6: Lasso Regression

Using lasso model to further potentially simplify the model to get the most significant features. As we can see from the below result, **the most important features are “vote\_count”, “budget”, and “major\_production\_companies”.**



Step5:OLS regression Step6: Lasso regression

# Statement of Work:

Data Cleaning: Xuanchen Liu & Simeng Cai Data visualization and Analysis: Xuanchen Liu Report Writing: Simeng Cai