Summary:

We trained the Model using 80% of our movies data and test it using 20%

Pre-processing:

Problem 1:

Directors Data was less than revenues Data (Directors = 47 rows) while revenue (465) so when we merged these two files, we found a lot of nulls in the directors' column → We Used an online API to fill the missing values in the directors' column, after filling the data there were some directors with unknown values so we dropped them.

Problem 2:

Genre column has some missing values, so we used the API to sill the missing data

Problem 3:

MPAA-ratings column has some missing values, so we used the API to sill the missing data

Problem 4:

Revenue column has a string data type while it was representing money values (int/float for more accuracy) so we had to split this string to remove the Dollar sign and commas between the digits and then we converted the output into float

Problem 5:

Release date column was an Object data type and the year value was only two digits so we had to convert it into datetime and split it into three new columns (year/month/day) after that we found that some years were in the future (ex: 2071,2055,…) so we looped over these year and adjust it into (1971, 1955, …) then we saved the year, day, month into the data frame and dropped the release-date column

Problem 6:

We detected some outliers in the revenue column (ex: some movies have 0 revenue and other movies where have 2048 revenue), so we dropped the outliers values from the dataframe

Problem 7:

Dealing with categorical values

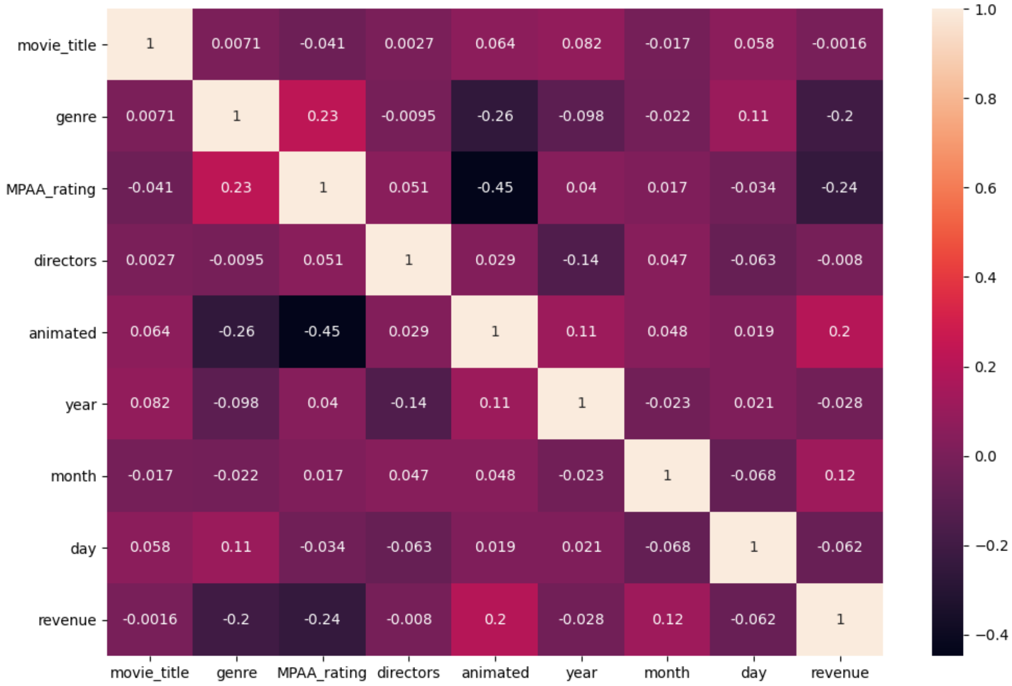
In Model 1: we used label encoder -> gives every categorical value a number

In Model 2: we used panadas.get_dummies to create column for every unique value and the value of the column is binary (0 or 1), but it results an overfitted model, so we have used label encoder for directory column.

Feature Selection:

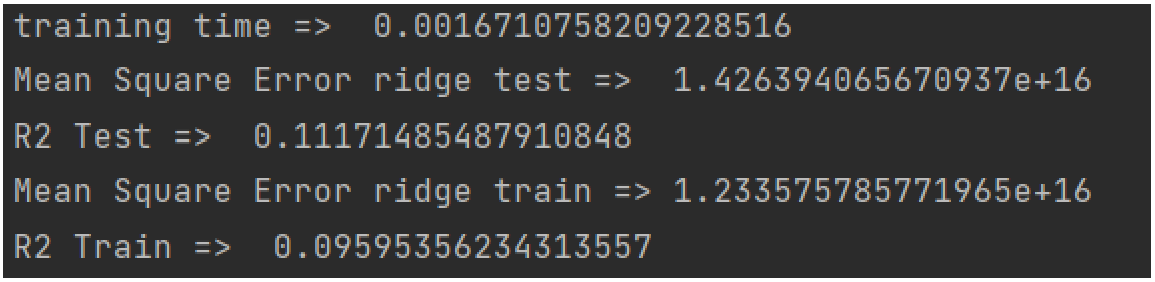
After calculating correlation, we have found out that <movie-title> had a very small correlation with the <revenue> (about -0.0016), so we had to drop it. Also, we have dropped <characters> since it has caused a lot of duplicates and a huge MSE.

The correlation threshold we agreed upon on 0.09 for optimal MSE.

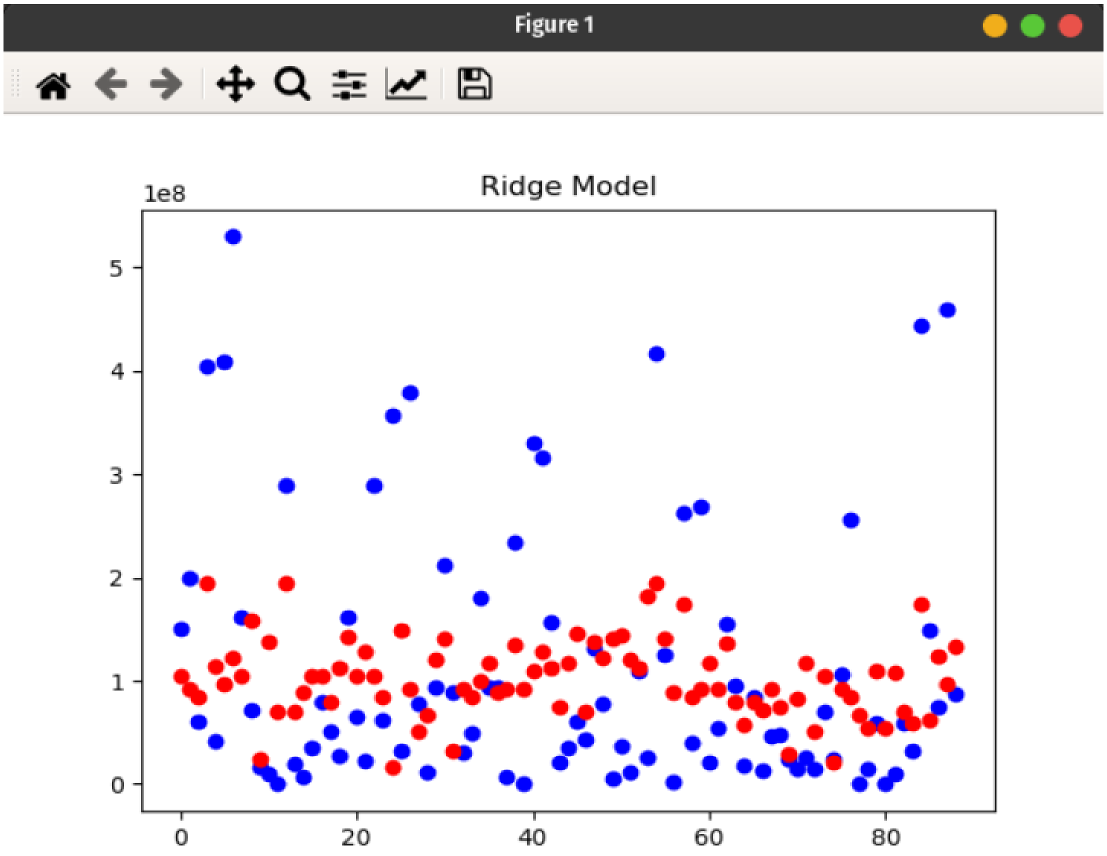


Model 1:

By using ridge algorithm



```
training time =>  0.0016710758209228516
Mean Square Error ridge test =>  1.426394065670937e+16
R2 Test =>  0.11171485487910848
Mean Square Error ridge train => 1.233575785771965e+16
R2 Train =>  0.09595356234313557
```



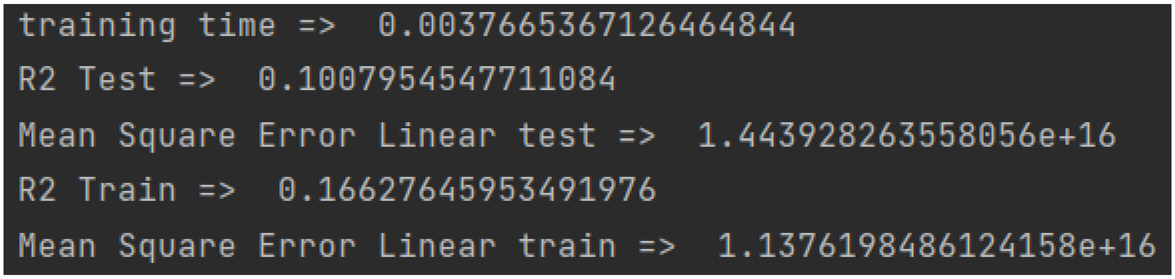
Blue -> Test

Red -> Prediction

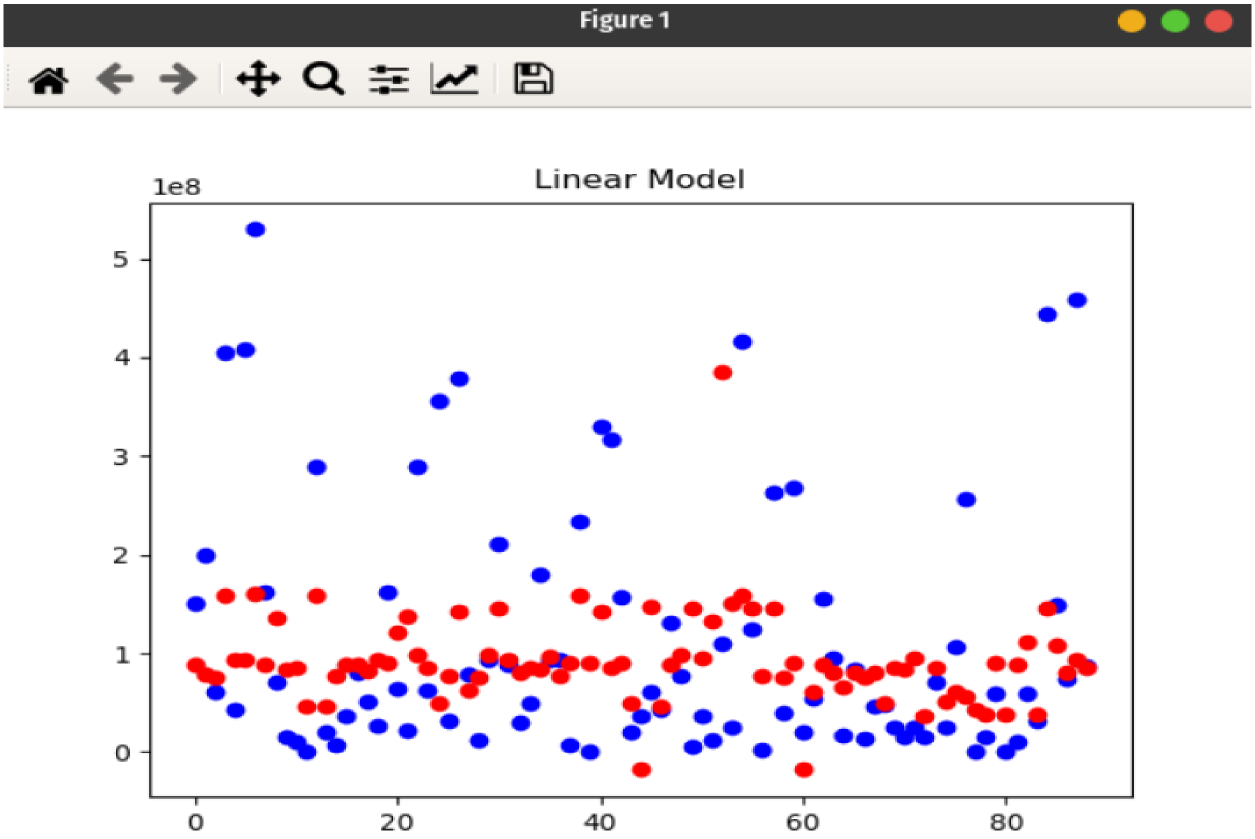
Model 2:

By using polynomial regression algorithm -> the model has overfitted the data and we got high MSE error

By using Multiple Linear regression algorithm ->



```
training time =>  0.0037665367126464844
R2 Test =>  0.1007954547711084
Mean Square Error Linear test =>  1.443928263558056e+16
R2 Train =>  0.16627645953491976
Mean Square Error Linear train =>  1.1376198486124158e+16
```



Blue -> Test

Red -> Prediction

Conclusion:

- Dealing with categorical data using label encoder gives better results than dealing with it using Dummies.
- MSE in the models are nearly the same
  MSE Model 1 test = 1.426394065670937e+16
  MSE Model 2 test = 1.443928263558056e+16