Midterm Project

Introduction:

The Freddiemac dataset serves as a reference for understanding the Single Family Loan-Level Dataset and covers approximately 26.3 million fixed-rate mortgages (including HARP loans) originated between January 1, 1999 and September 30, 2017.  Monthly loan performance data, including credit performance information up to and including property disposition, is being disclosed through March 31, 2018.  Specific credit performance information in the dataset includes voluntary prepayments and loans that were Foreclosure Alternatives and REOs. Specific actual loss data in the dataset includes net sales proceeds, MI recoveries, non-MI recoveries, expenses, current deferred UPB, and due date of last paid installment.

Dataset:

The Freddiemac dataset basically is divided into to types:

1. Sample Dataset

Sample dataset is basically a smaller portion of the entire Freddiemac dataset. The sample dataset is basically zip files of each of the year from 1999 to 2017 and each of the year zip files contain 2 files : Origination data and Monthly performance data

1. Historical Dataset

Historical dataset is the entire Freddiemac dataset. The historical dataset is very huge and hence each year’s data has been split into 4 quarters. Each of the year-quarter zip file of the historical data contain 2 files : Origination data and Monthly performance data

Data Download :

The ‘SAMPLE’ data from Freddiemac website is downloaded programmatically and data downloaded is extracted to get origination and monthly performance file for each of the year from 2005 to 2007.

The login to the website is done automatically using web bot which is **Web** automation library for **python**. It is based on the selenium framework for **web** browser automation.



Once the website has been auto-logged in, the data is downloaded and extracted. The data from two different sets of files (origination and monthly performance) is grouped by year and created into dataframes.



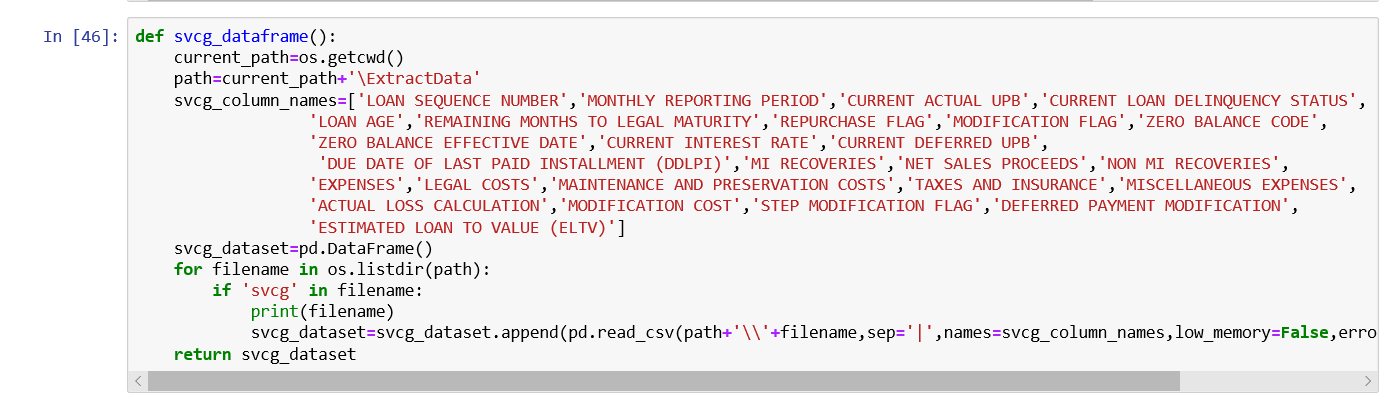
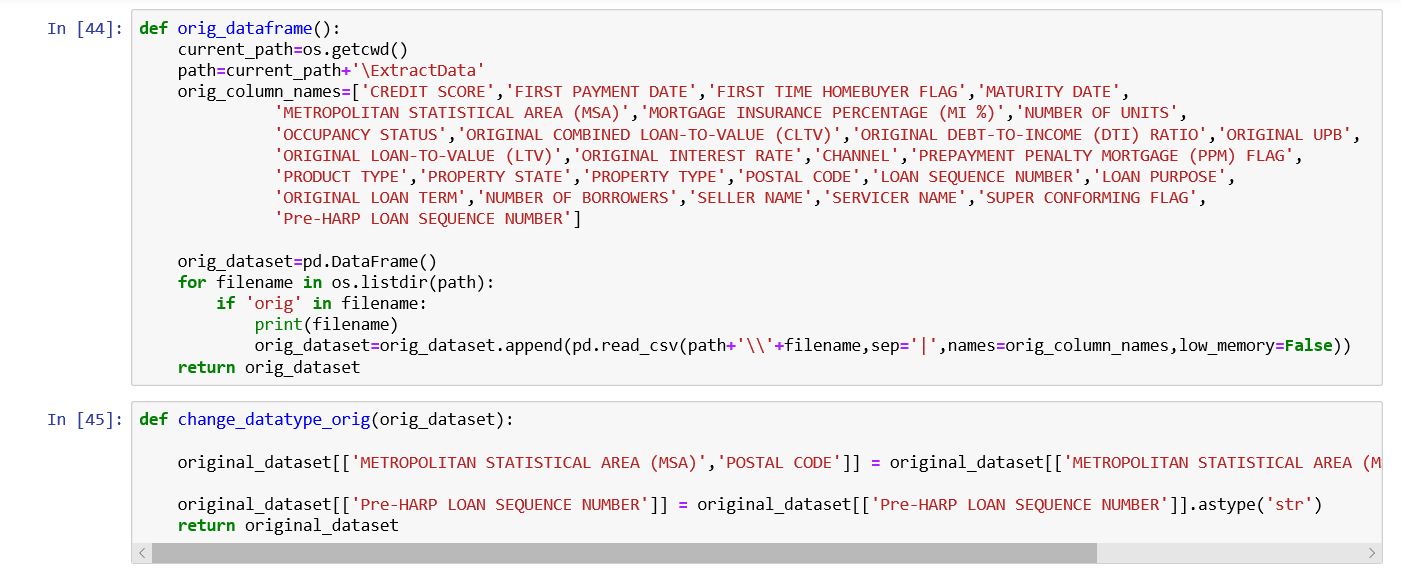
So now two data frames are created – One with the origination data from year 2005 to 2017 and one with monthly performance data from 2005 to 2017.

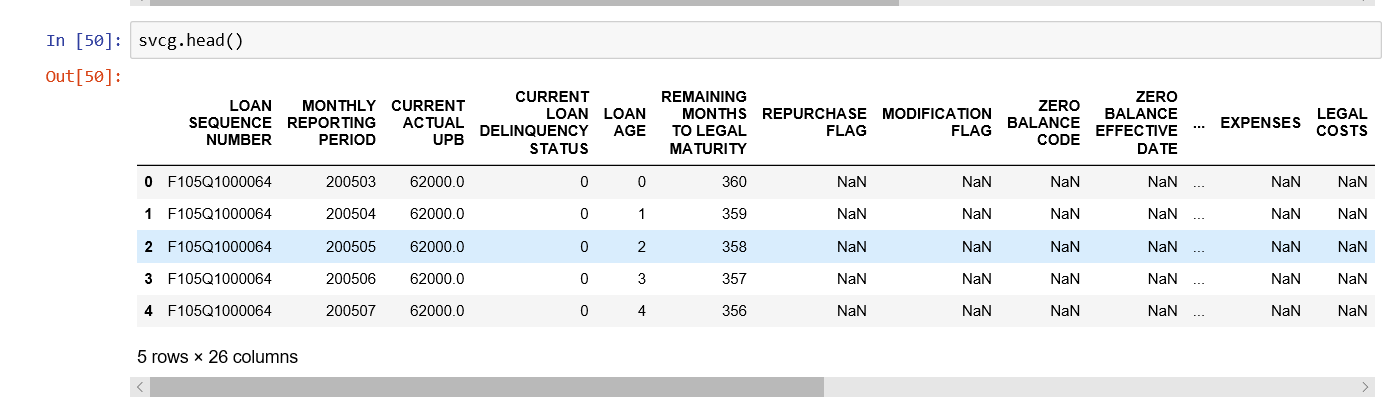
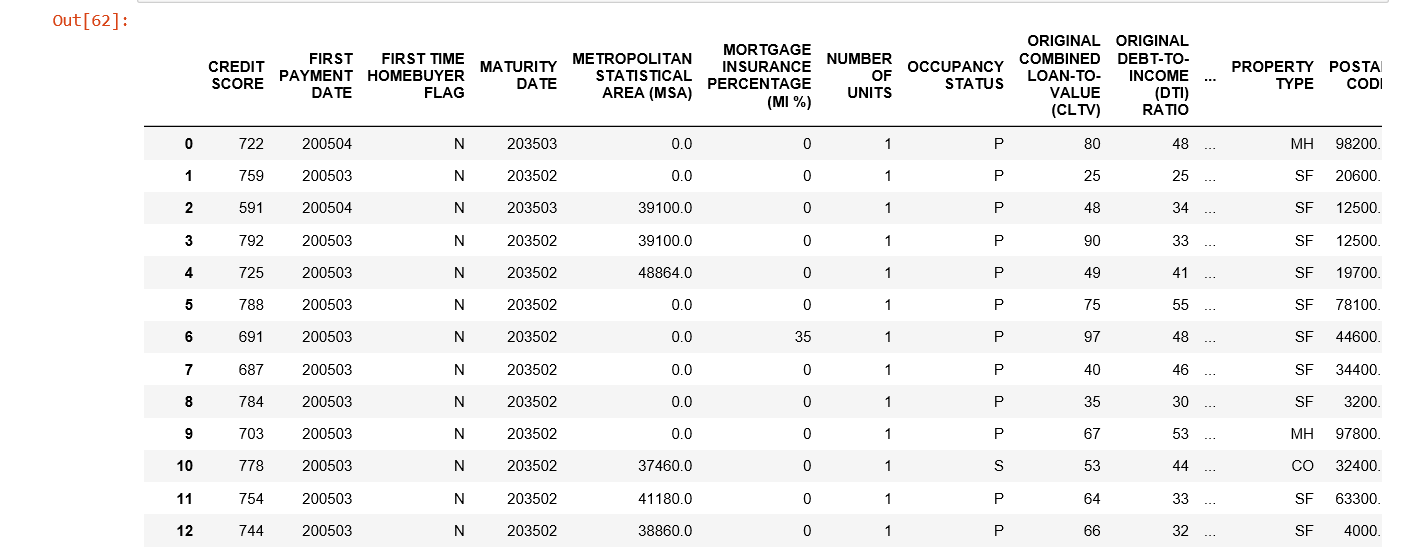
Data Preprocessing and Cleaning:

Once the two dataframes are created, they are cleaned by handling null values and changing the datatypes of some of the columns.

Once the data is cleaned , the dataframes are written into csv

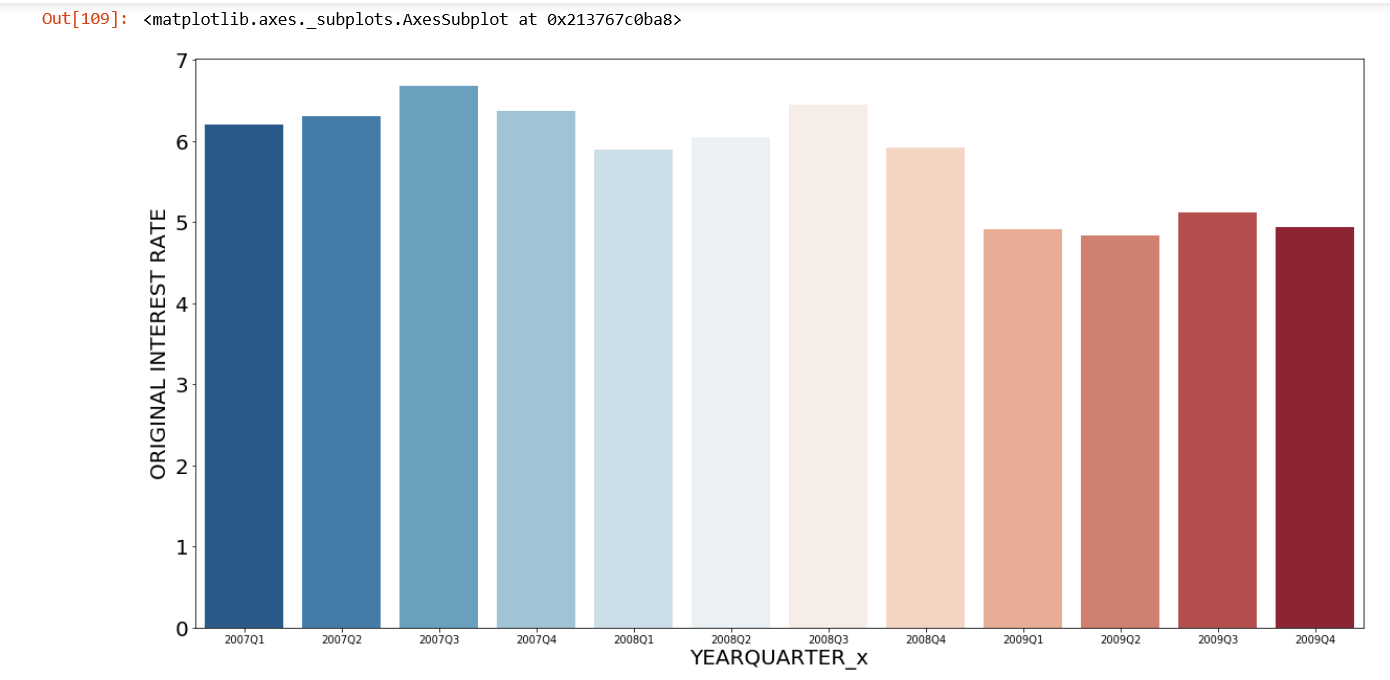
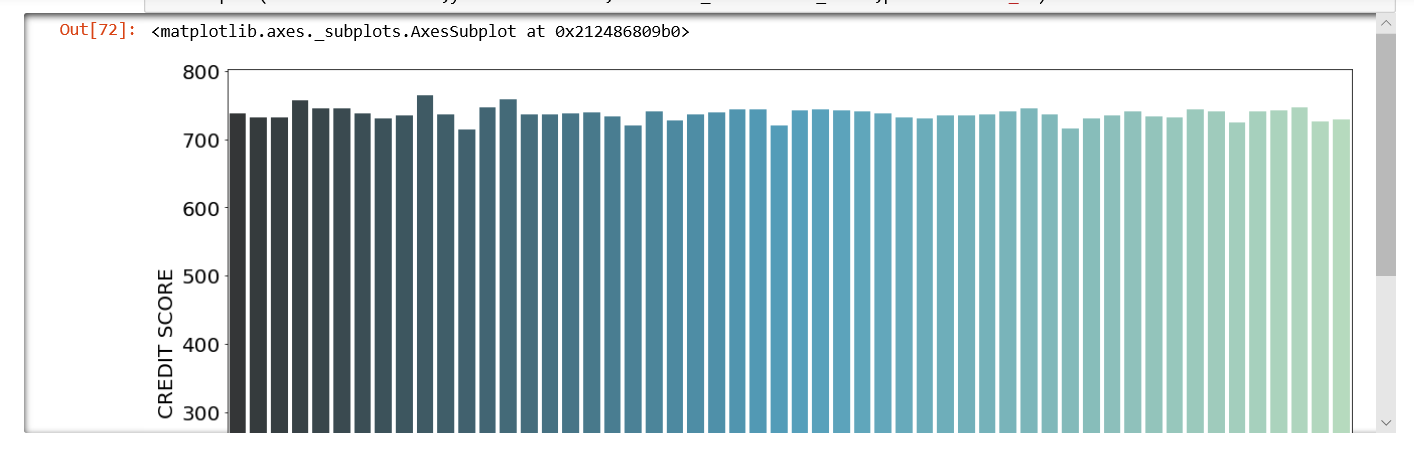


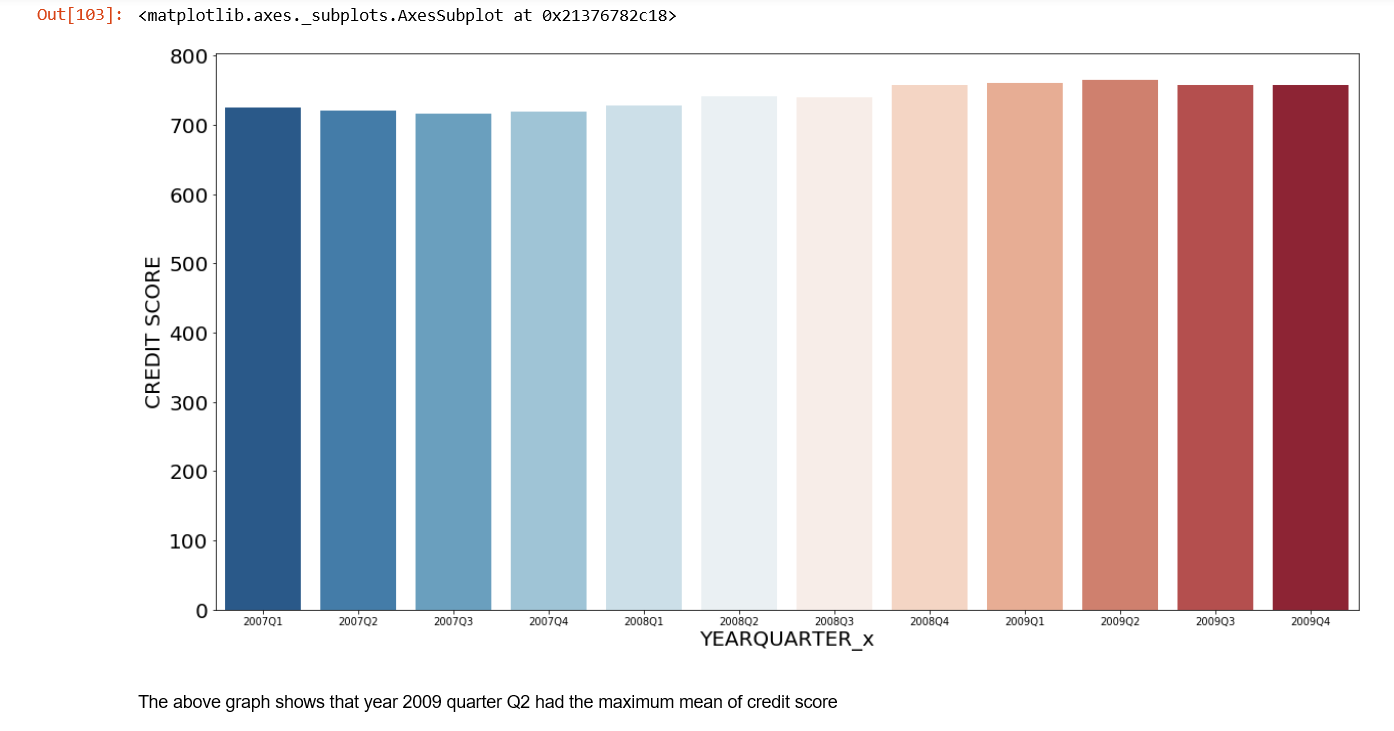
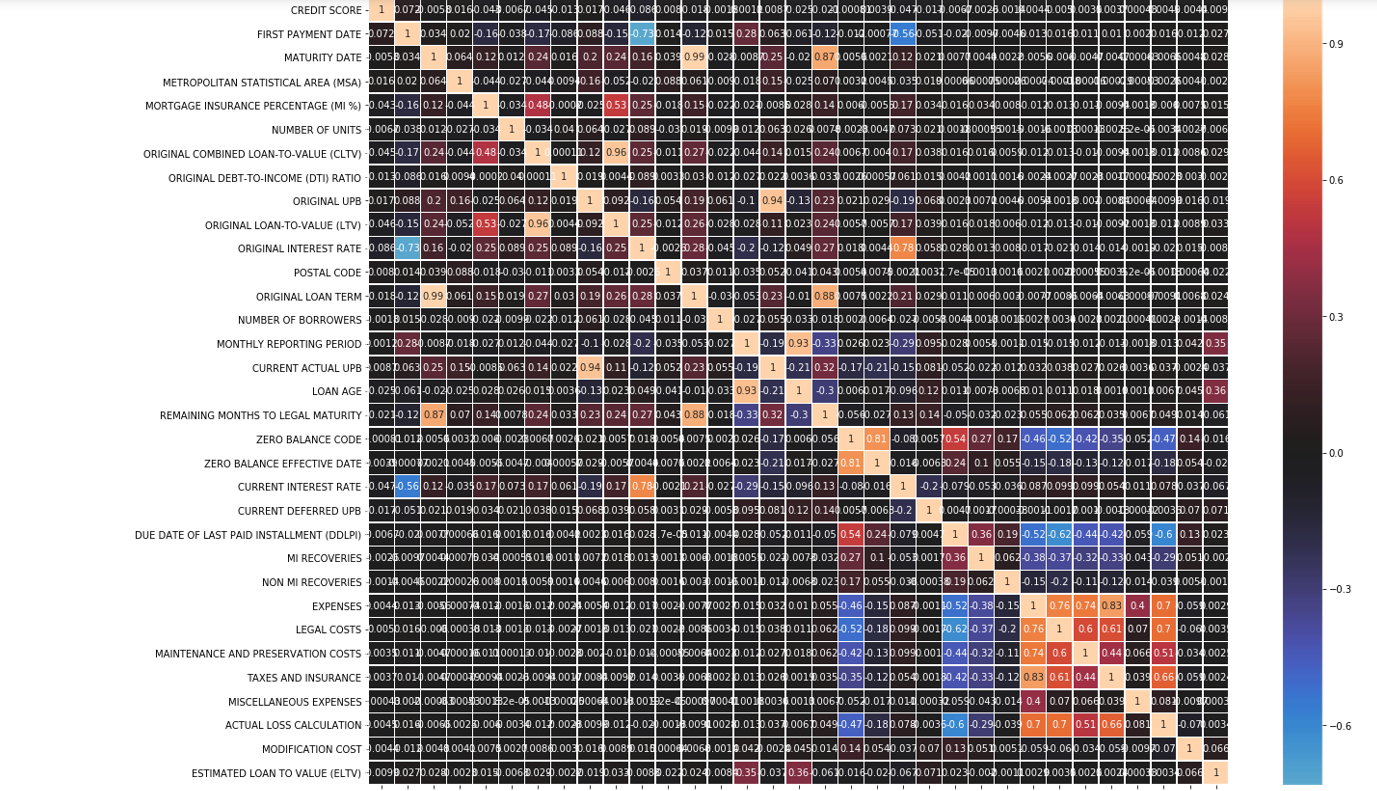
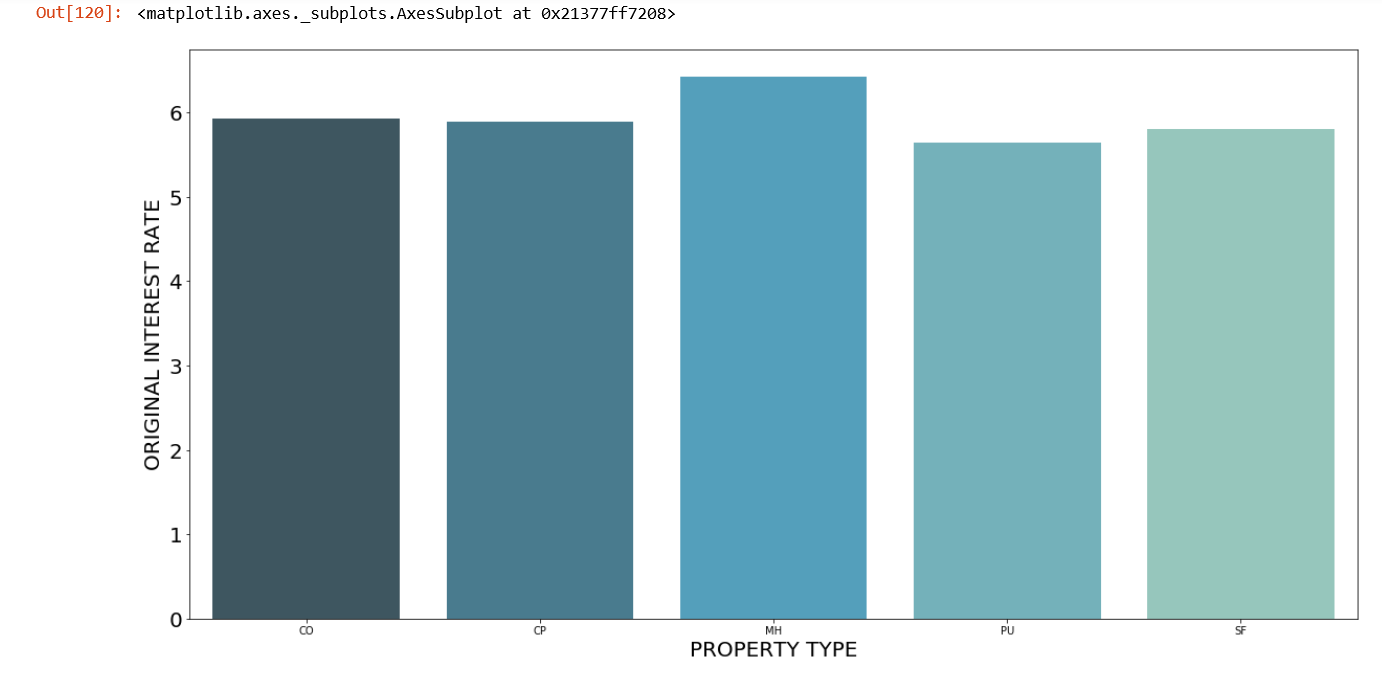




Exploratory Data Analysis :

The EDA is performed on various parameters of the dataset such as interest rate, upb, state, quarter etc.

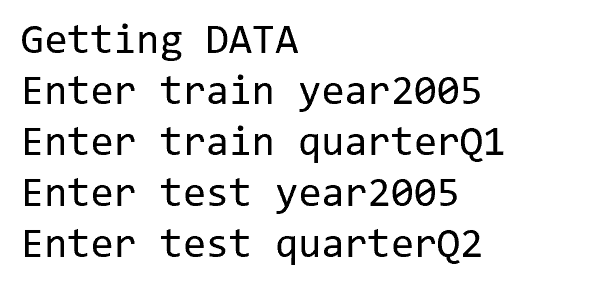




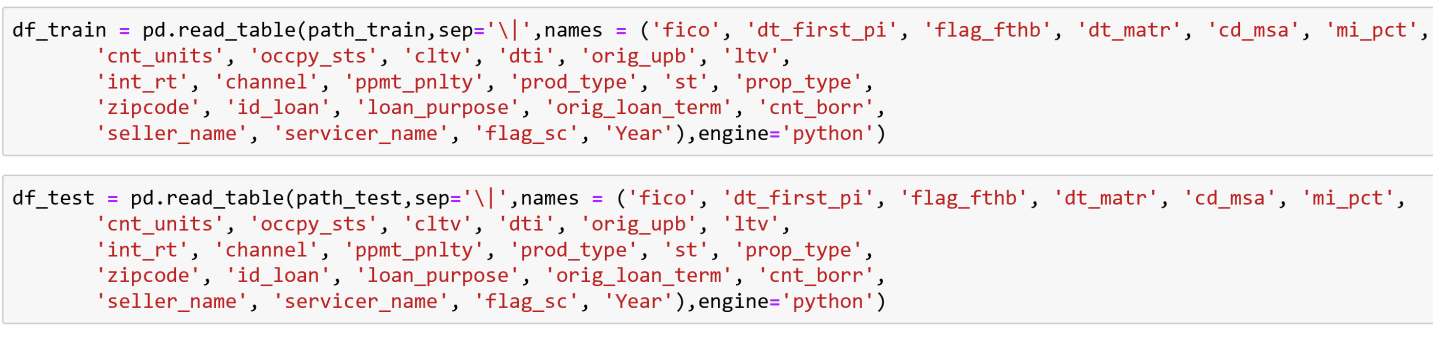
**Models to Predict Interest Rates**

Different prediction Models are used to predict the interest rates of a home loan. The data gets downloaded automatically and preprocessed before forming the train and test data. The user needs to specify the quarter (Q1-Q4) and year (2005-2018) of the train and test data. Web bot has been used internally to handle the login and download tasks.

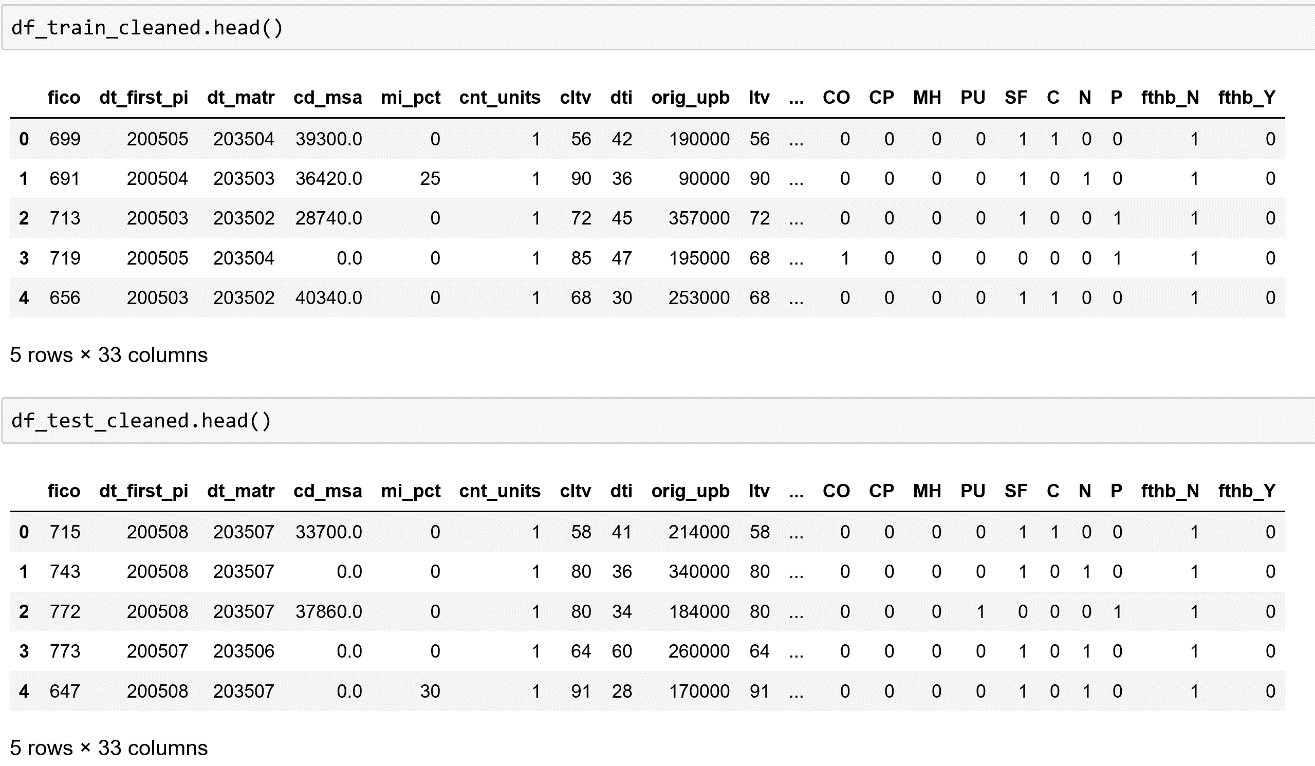
Specify the train quarter – Year and test quarter – Year



Extracting the Data



Data after cleaning and preprocessing

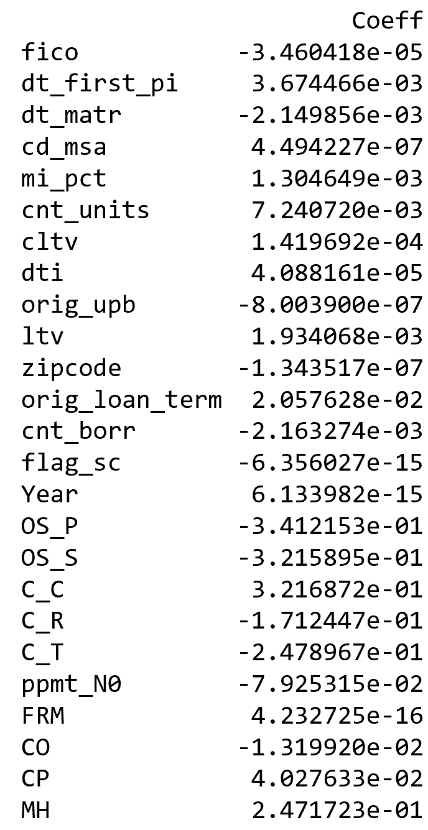


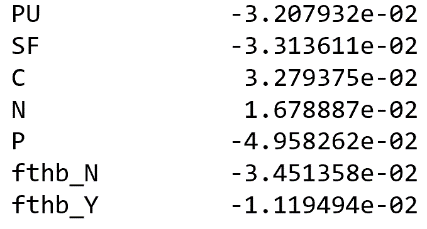
**Linear Regression Model**

**lm=LinearRegression()**

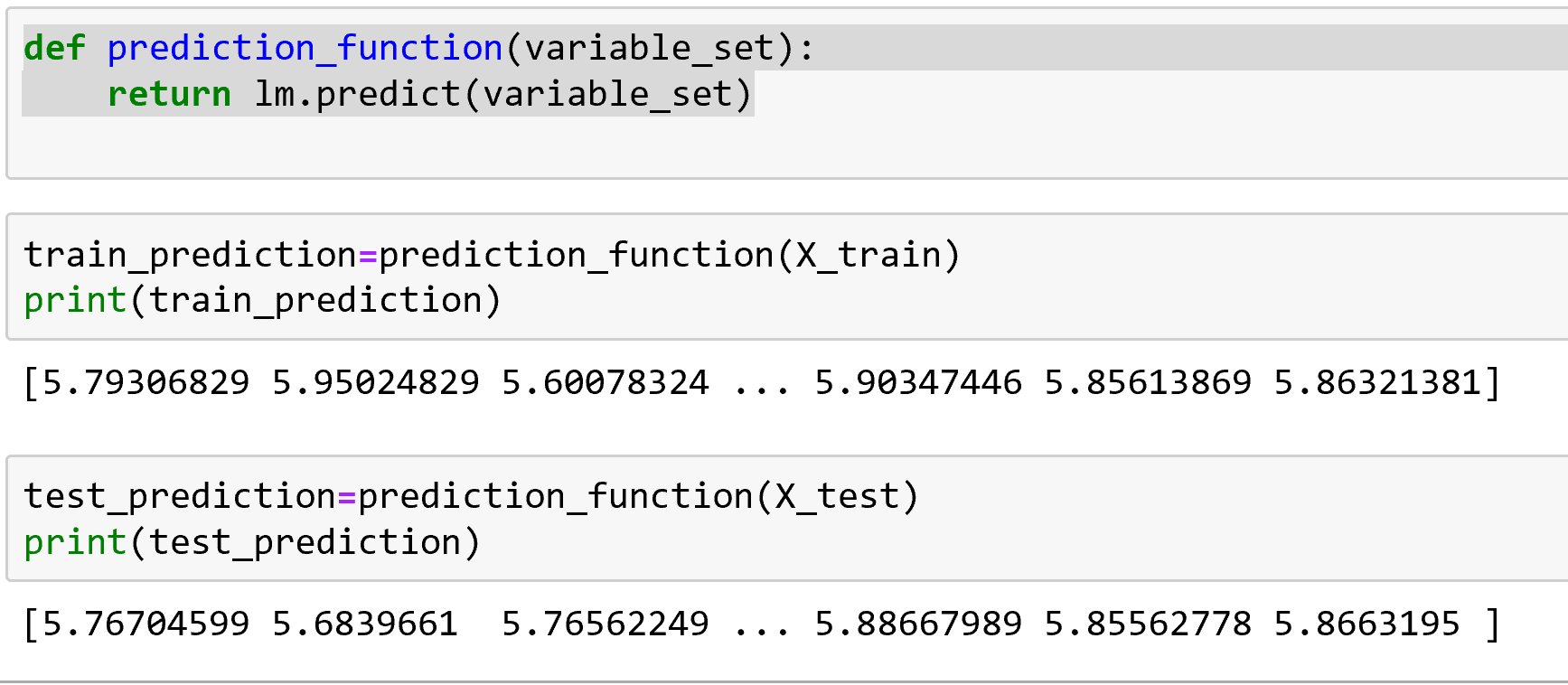
**lm.fit(X\_train,y\_train)**

Coefficients Table:

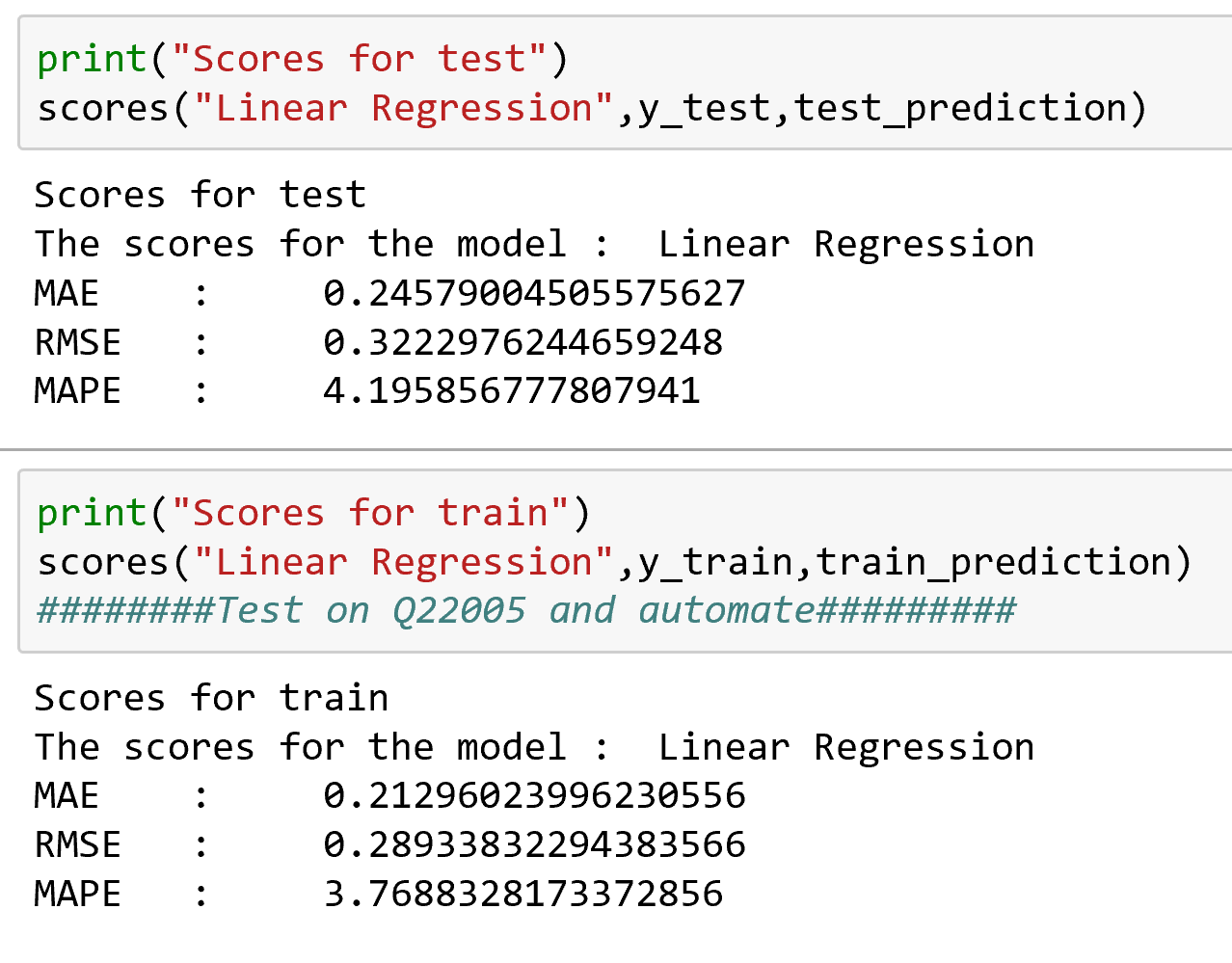


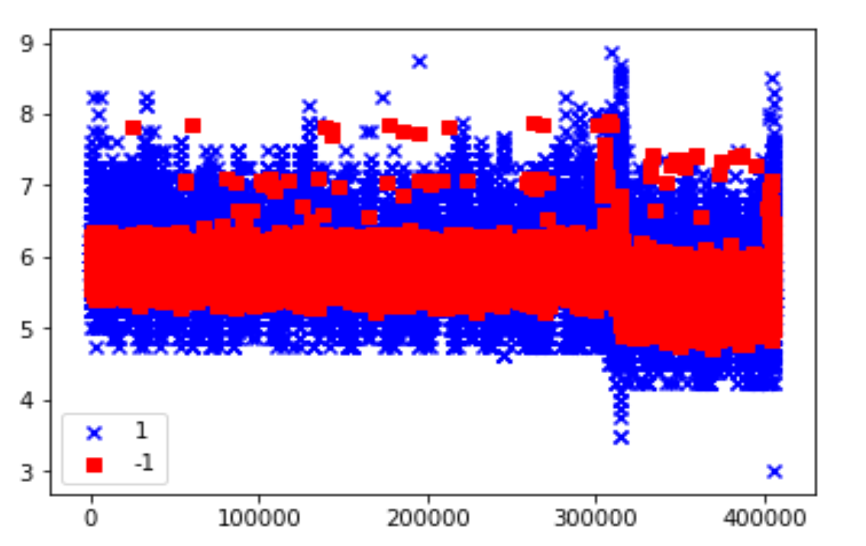


Predicting Interest Rates on both train and test quarters

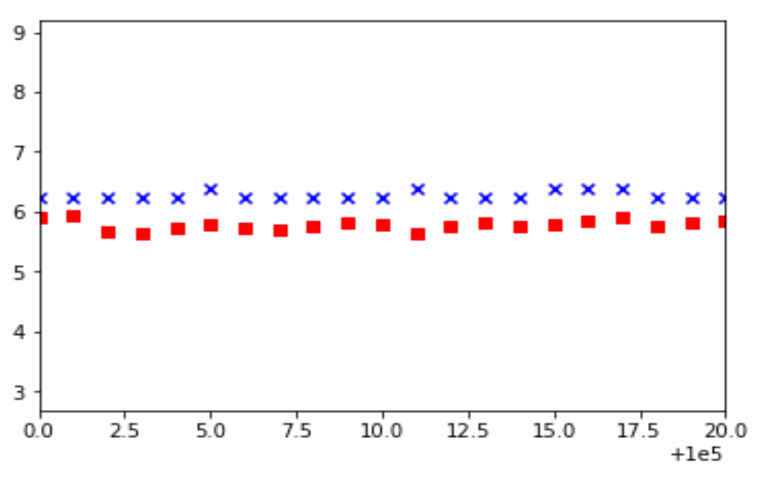


RESULTS:

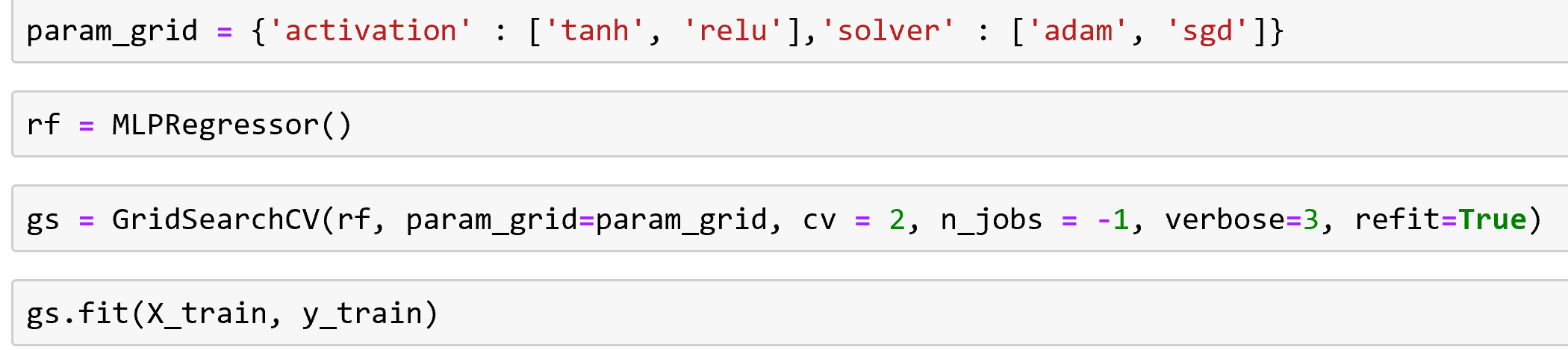
  
  
Graph of predicted and actual interest values



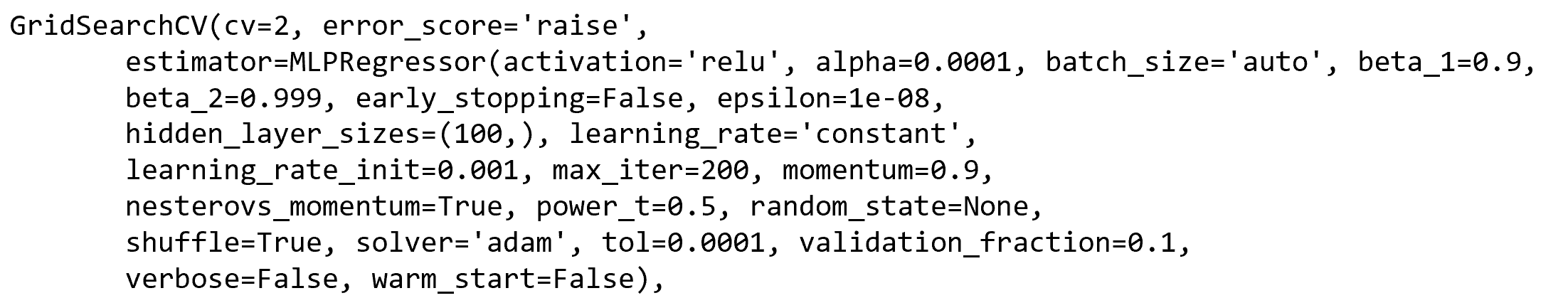
For a small interval of x



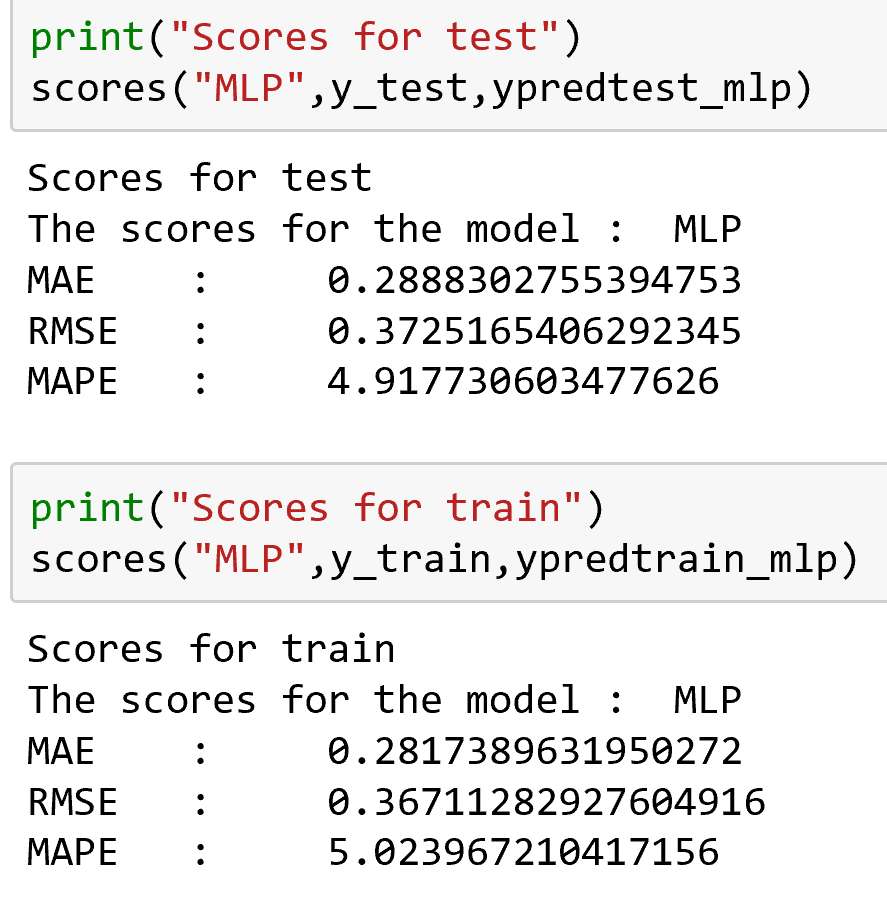
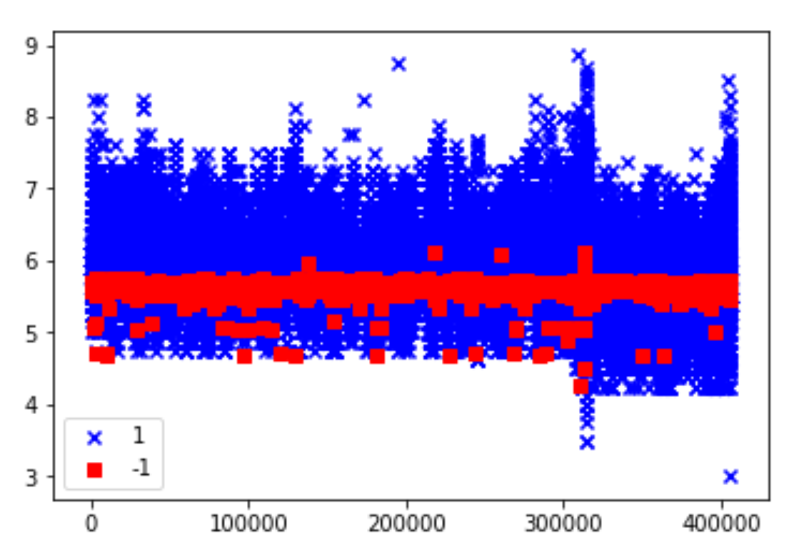
**Neural Network Model**

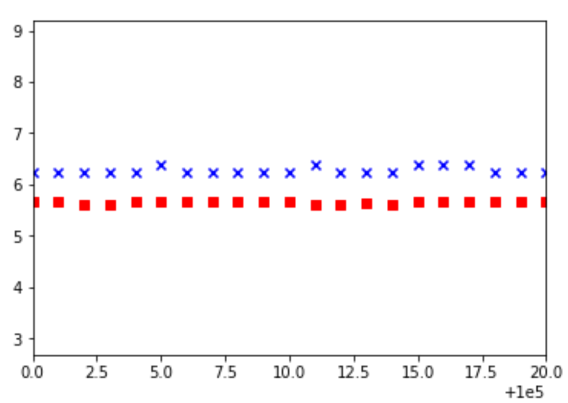


**Best Parameters and Hyper parameters**

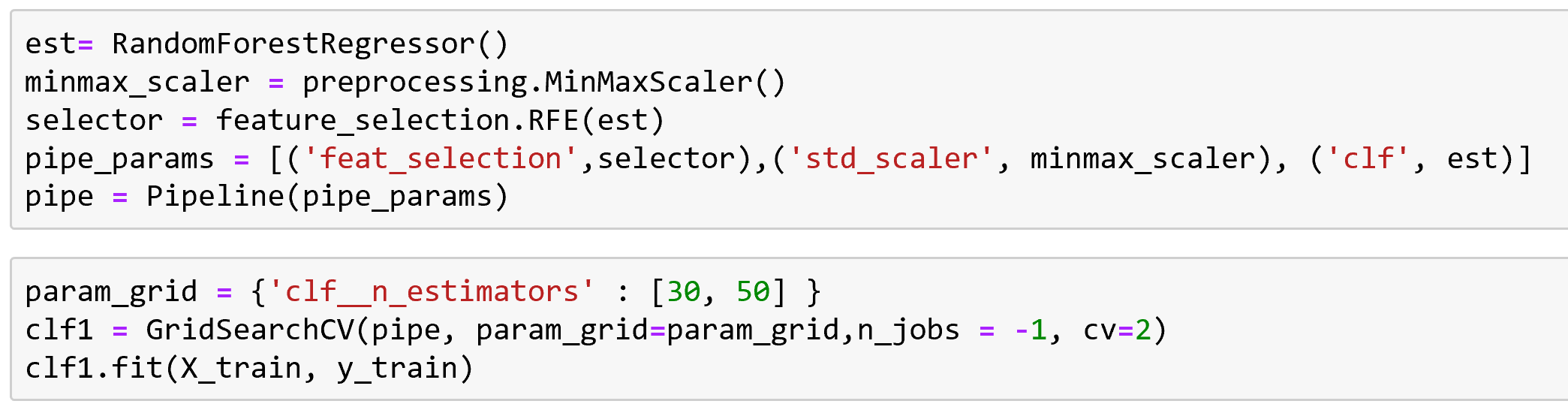
p

RESULTS:

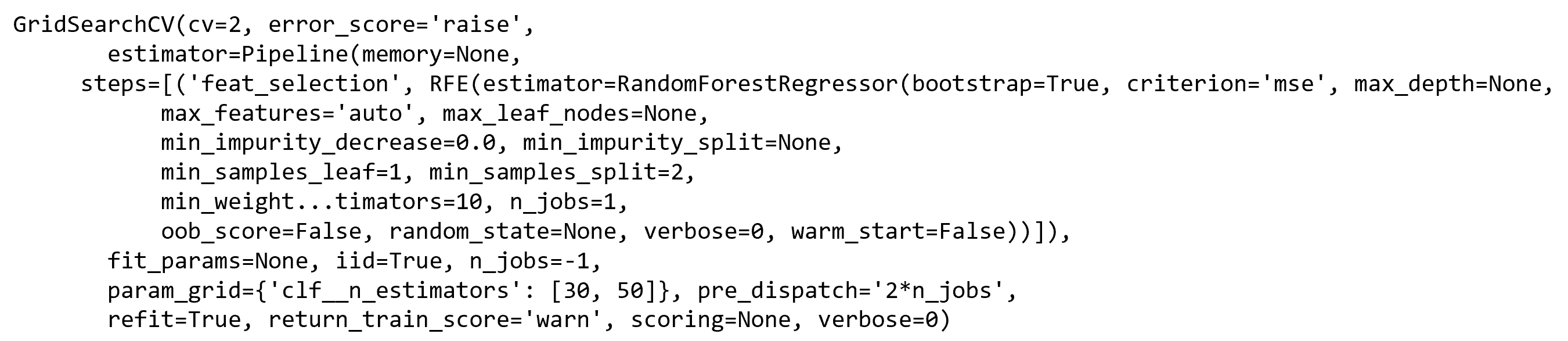




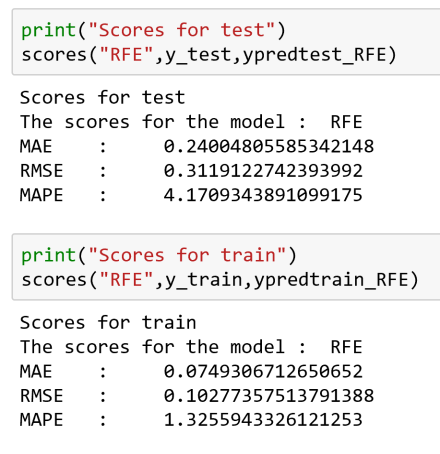
**Random Forest Model**

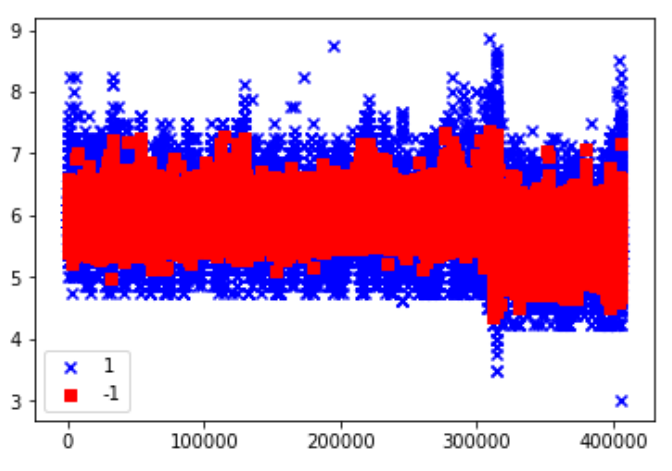
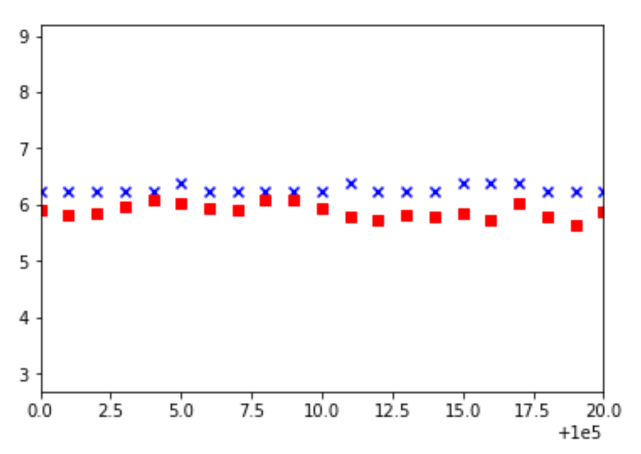


Best parameters and Hyperparameters



RESULTS:





**Feature Selection:**

**Forward Search**

With 20 best features

Description: A screenshot of a cell phone

Description generated with very high confidence

With 25 best features

Description: A screenshot of a cell phone

Description generated with very high confidence

With 30 best features

Description: A screenshot of a cell phone

Description generated with very high confidence

**Backward Search**

With 15 best features, With 25 best features, With 30 best features

Description: A screenshot of a cell phone

Description generated with very high confidence

**Exhaustive Search**

With best 3 features

Description: A screenshot of a cell phone

Description generated with very high confidence

With MinMax Scalar and best 30 features

Description: A screenshot of a cell phone

Description generated with very high confidence

**TPOT**

**3 Generations**

**Description: A picture containing screenshot

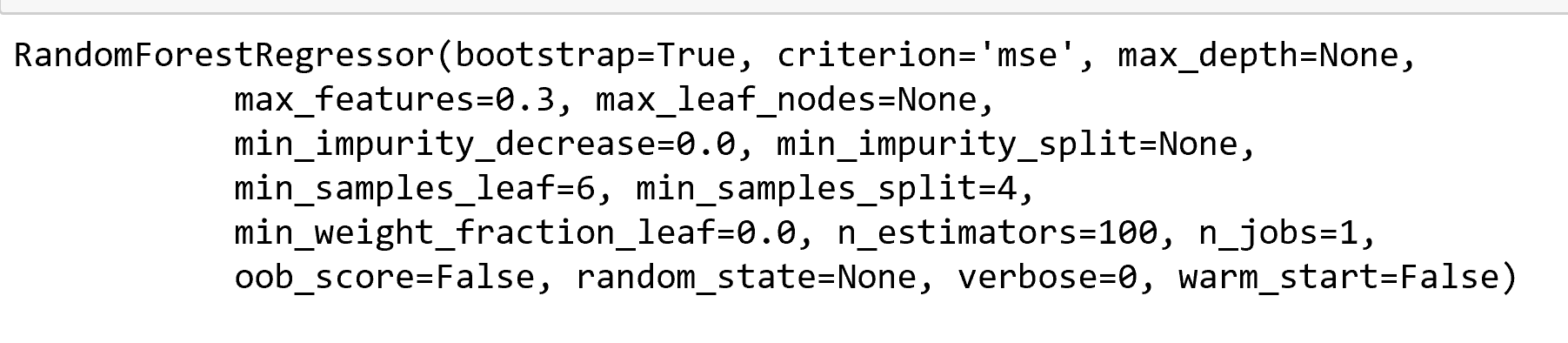
Description generated with high confidence**

**Best Feature**

**Description: A screenshot of a cell phone

Description generated with very high confidence**

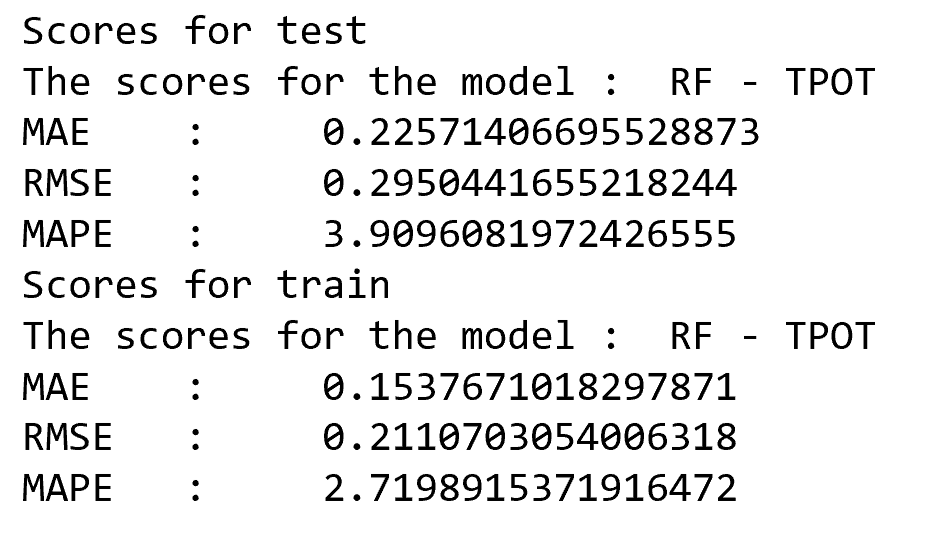
**RF Model From TPOT**



**RESULTS:**



**Therefore, best model is Random Forest Regressor**



**Analysis Using the Best Model – Random Forest**

**For rolling quarters – 2007**

**Train – Q1 - Test – Q2**

**Description: A screenshot of a cell phone

Description generated with very high confidence**

**Train – Q2 - Test – Q3**

**Description: A screenshot of a cell phone

Description generated with very high confidence**

**Train – Q3 - Test – Q4**

****

**Train – Q4 - Test – Q1(2008)**

**Description: A close up of a logo

Description generated with high confidence**

**Conclusion:**

* The algorithm performs well for the quarters of same year
* When the model was trained with Q4 2007 and used to predict interest rates for the Q1 of next year ie 2008, we got the least score

**After 2 Years**

* **Q1 2007 – Q1 2009**

**Description: A screenshot of a cell phone

Description generated with very high confidence**

* **Q Q1 2007 – Q1 2009**

**Description: A screenshot of a cell phone

Description generated with very high confidence**

* **Q1 2007 – Q1 2009**

**Description: A screenshot of a cell phone

Description generated with very high confidence**

* **Q1 2007 – Q1 2009**

**Description: A screenshot of a cell phone

Description generated with high confidence**

**Conclusion:**

* The doesn’t perform well when trained on the data from one year and used to predict the interest rates after 2 years
* This can be observed from the very high scores of test MAE/RMSE

**Economic Boom (2013)**

* **Trained on Q1 2005 – Predicting Q1 2013**

**Description: A screenshot of a cell phone

Description generated with very high confidence**

**Description: A picture containing screenshot

Description generated with very high confidence**

* From the scores and graph we can conclude that the algorithm can not used for predicting the interest rates for 2013.
* **Due to the Economic Boom the interest rates are really low and hence the predicted interest rates fall out of range to that of actual interest values**

**Change of Regime (2016)**

* **Trained on Q1 2007 – Tested on Q1 2016**

**Description: A screenshot of a cell phone

Description generated with very high confidence**

Description: A screenshot of a cell phone

Description generated with high confidence

* Model is recommended to calculated the interest rates during 2016 as well as the interest rates are decreased due to the regime change

**Conclusion:**

**The algorithm works good between the quarters of the same year and can be used to predict the interest rates of next quarter from the present quarter data within the same year**

**Whereas for predicting the rates after 2 years or during economic boom or change of regime, the algorithm doesn’t work as expected and hence can not be used**

**Part 2 – Classification**

Algorithms:

**Logistic Regression:**

Training and Testing accuracy:

Random Forest Regression - Score - Training: 0.9725

Random Forest Regression - Score - Testing: 0.9694

Confusion Matrix:

[[972138 230]

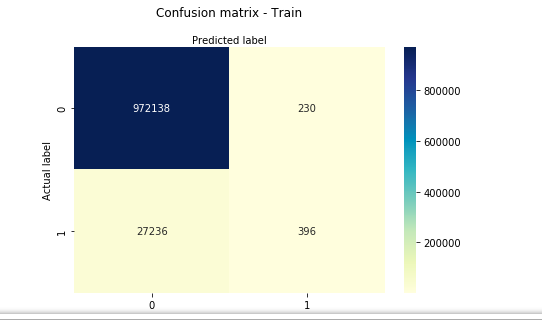
[ 27236 396]]

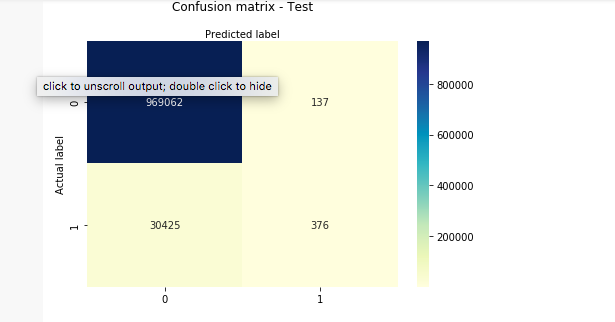
Confusion matrix accuracy (train): 0.972534

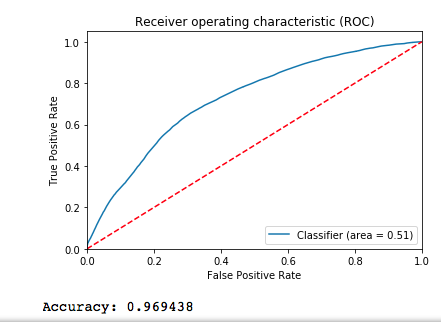
[[969062 137]

[ 30425 376]]

Confusion matrix accuracy (test): 0.969438







**Random Forest:**

Training and Testing Accuracy:

Random Forest Regression - Score - Training: 0.9747

Random Forest Regression - Score - Testing: 0.9246

Confusion Matrix:

[[972307 61]

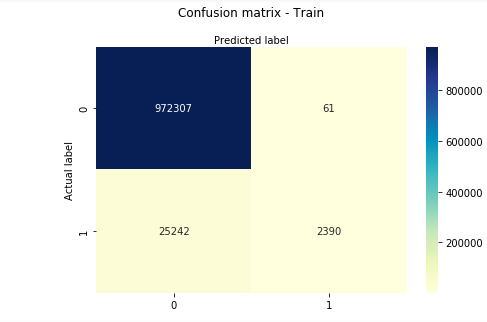
[ 25242 2390]]

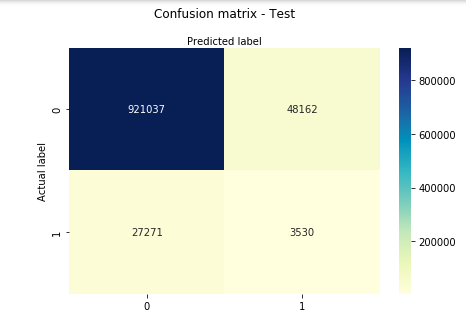
Confusion matrix accuracy (train): 0.974697

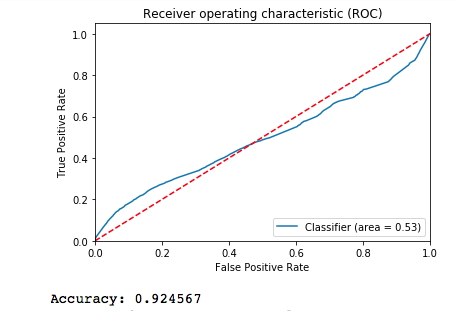
[[921037 48162]

[ 27271 3530]]

Confusion matrix accuracy (test): 0.924567







**Neural Network:**

Training and Testing accuracy:

Random Forest Regression - Score - Training: 0.9725

Random Forest Regression - Score - Testing: 0.9693

Confusion Matrix:

[[972021 347]

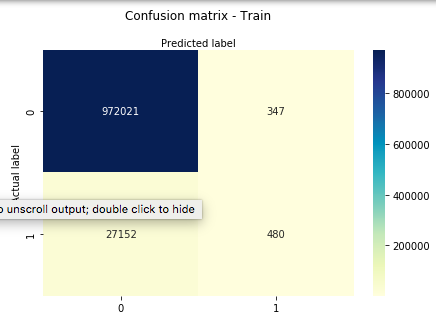
[ 27152 480]]

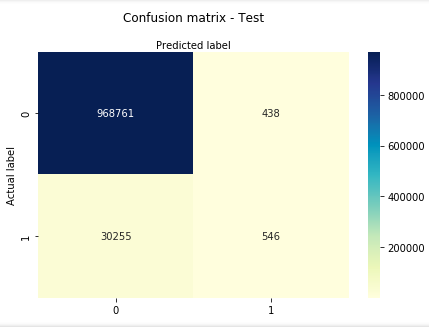
Confusion matrix accuracy (train): 0.972501

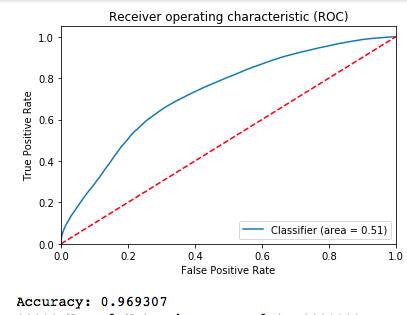
[[968761 438]

[ 30255 546]]

Confusion matrix accuracy (test): 0.969307







Main Script:

Output files:

Main script takes below parameters and downloads the quarter dataset for the provided inputs:

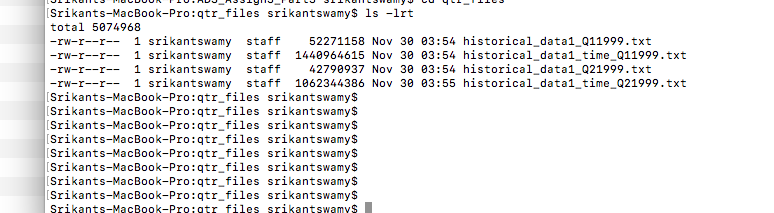
*Quarter\_year\_Train*

*Quarter\_year\_Test*

*User\_name*

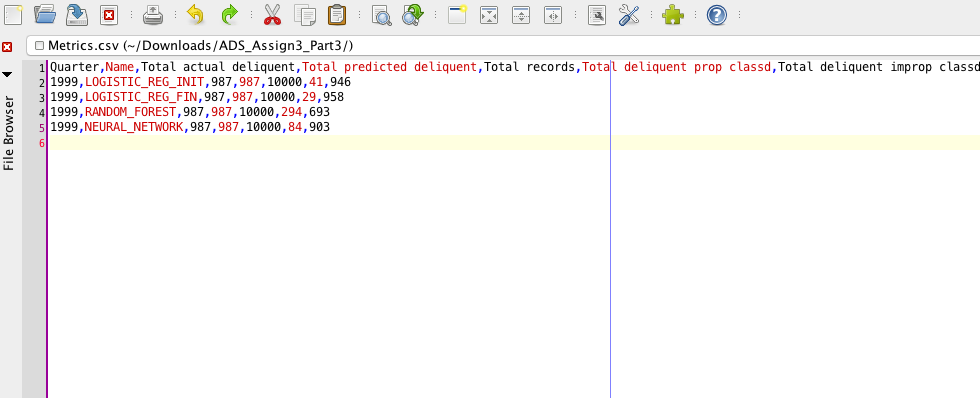
*Pass\_wd*

*Records*



Metric file is generated when the script when the wrapper script is running outside the main script.

(Note: To save time the wrapper script was ran for two years as shown in the snapshot))

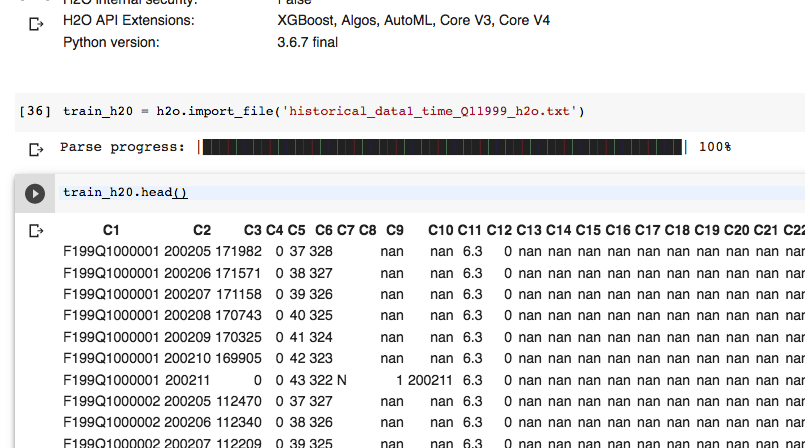


**Conclusion:**

Looking at the confusion matrix accuracy score and ROC graph, we can say that better performance and accuracy are achieved using neural network classifier model.

**Auto ML:**

H20 Auto ML



Team Contribution: