ESSENTIALS OF DATA SCIENCE

RRN : 200171601060

NAME: AMREEZ KHAN

TOPIC : NEWYORK CITY TRIP FARE PREDICTION

**AIM:**

NEW YORK CITY FARE PREDICTION

WE WILL TRAIN A MACHINE LEARNING MODEL USING 3 DIFFERENT ALGORITHMS TO PREDICT THE FARE

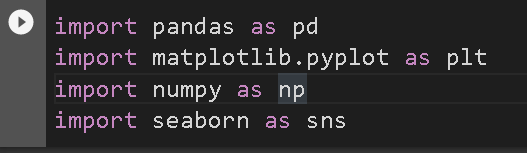
DATASET LINK :  [https://www.kaggle.com/c/new-york-city-taxi-fare-prediction/overview](https://jovian.ai/outlink?url=https%3A%2F%2Fwww.kaggle.com%2Fc%2Fnew-york-city-taxi-fare-prediction%2Foverview)

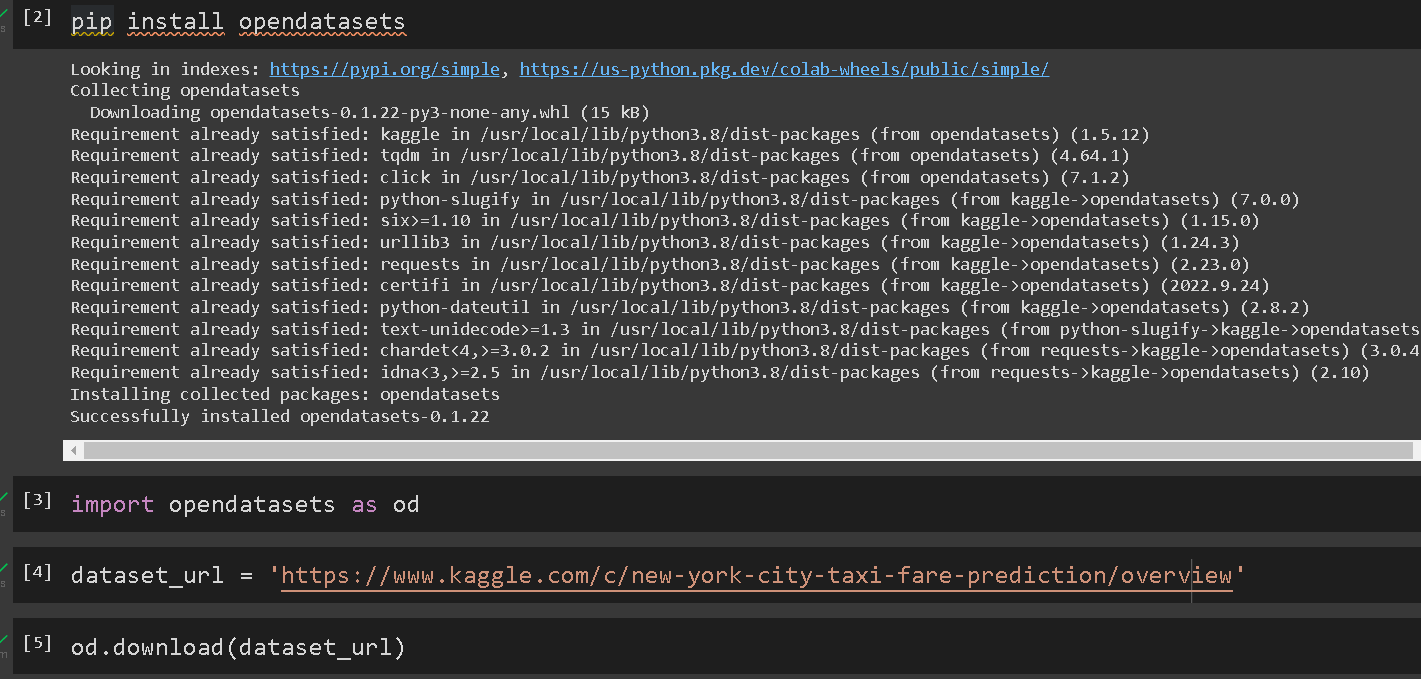
STEPS:

1. DOWNLOAD THE DATASET FROM KAGGLE
2. INSTALL THE REQUIRED LIBRARIES
3. Create baseline models and define the performance metrics
4. Build the machine learning models after performing necessary feature engineering and also after removing the outliers

SINCE WE USE GOOGLE COLAB THERE IS NO NEED OF INSTALLING THE BASIC LIBRARIES LIKE numpy,seaborn,pandas AS THEY ARE ALREADY PREINSTALLED

Now we will import the libraries



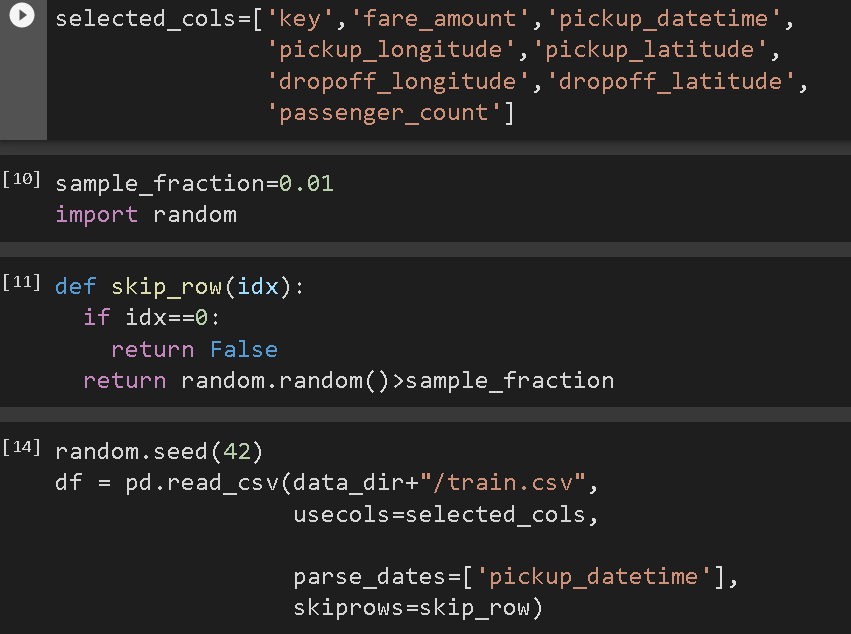


We are importing the necessary libraries and to download the dataset via google colab we are installing the opendatasets library to download

And importing it

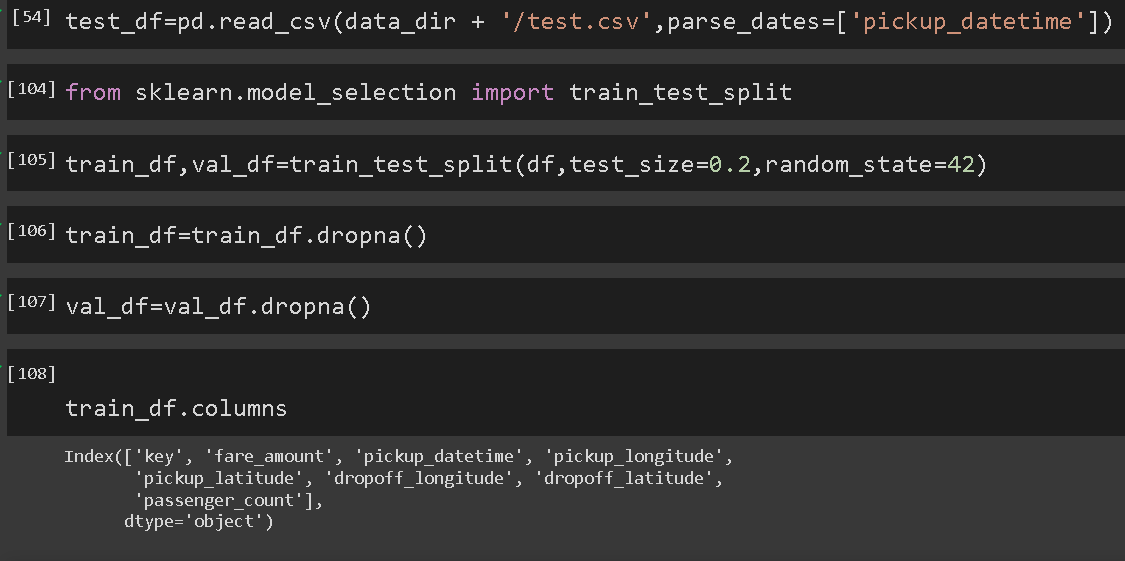


Since the dataset is massive we will use 1% of the dataset



Now we will split the dataset to train,validation,test data

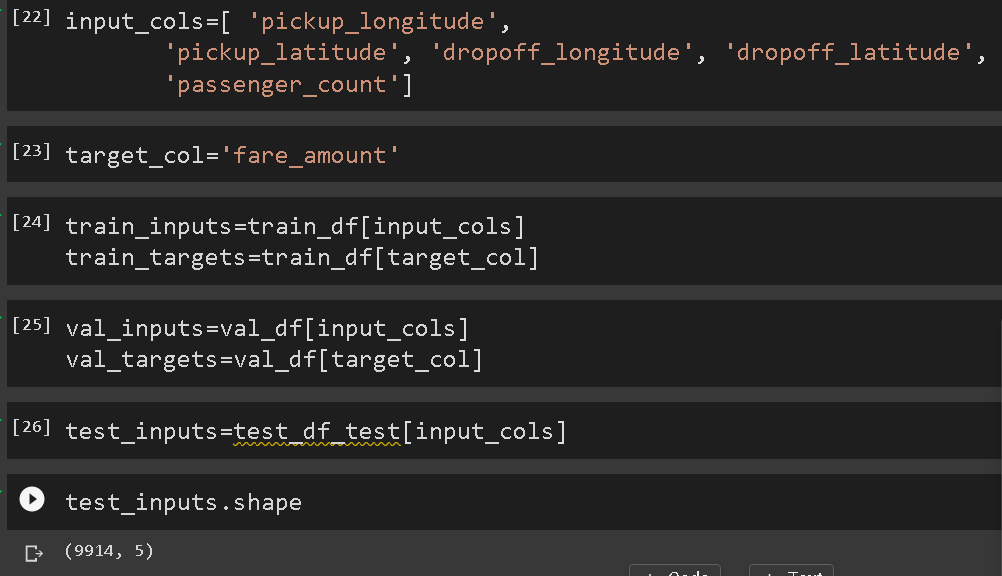
And we will define our input columns and target columns which we are going to use in the train/validation/test data

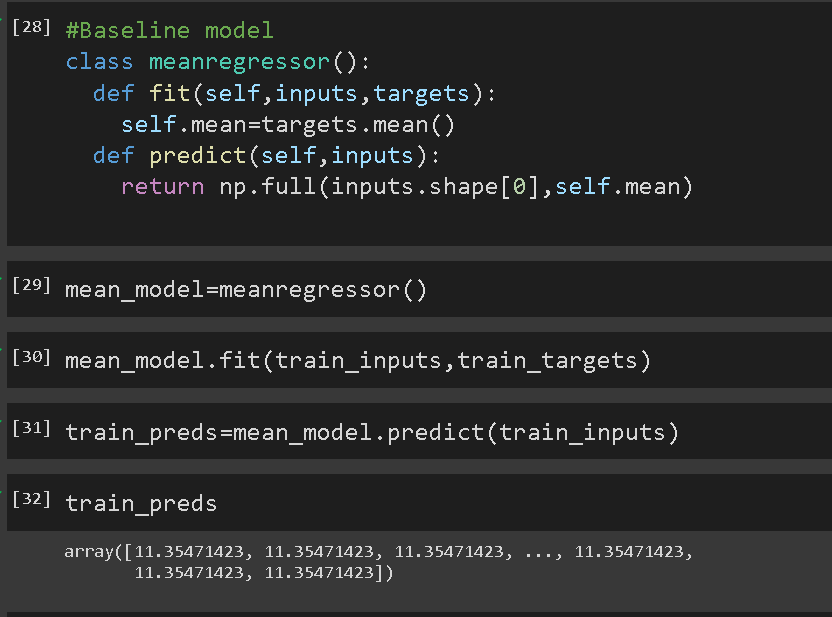


The performance metric we are using for this problem is root mean squared error

So firstly we would create an baseline model to check its root mean squared error so that we would get an idea of the error we should be focusing

NOTE: we are building this baseline model before doing the necessary feature engineering



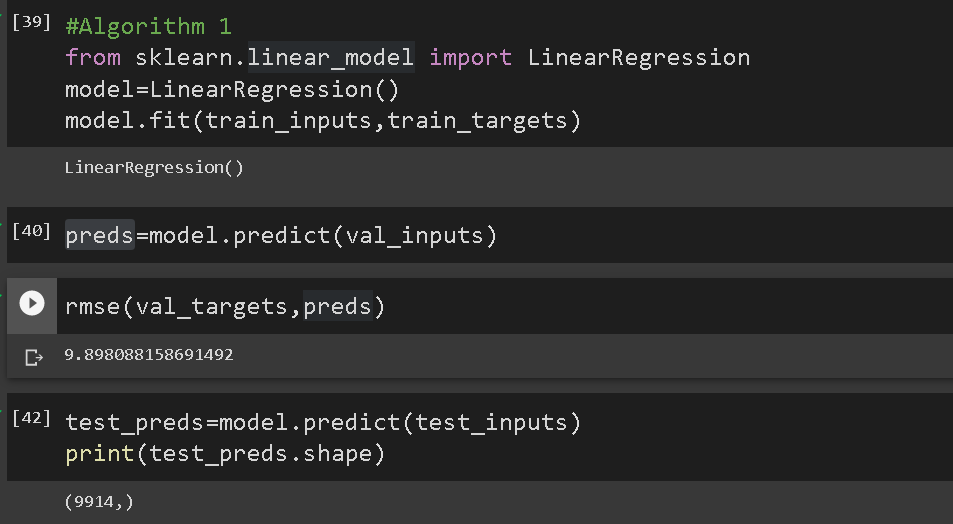


Here in the prediction for the training data we get an rmse of 11.35

So for the machine learning model we are building we should be aiming for an error less than 11.35

Our first Machine learning model/algorithm:

Linear Regression

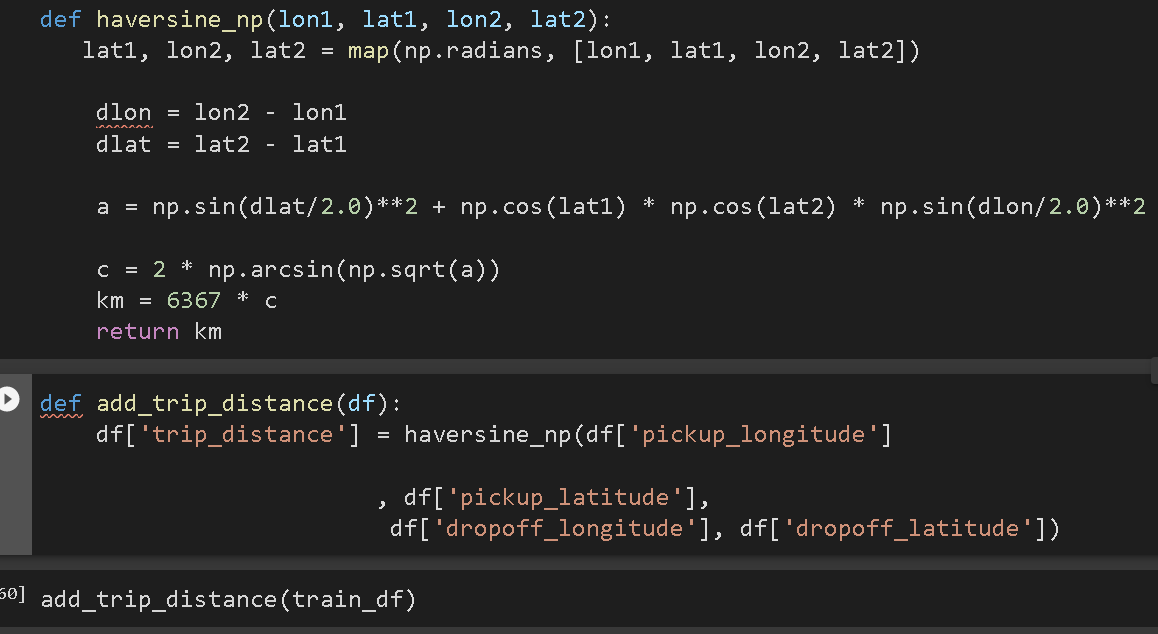


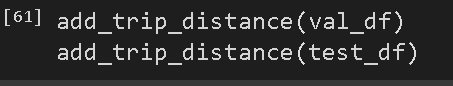
So as we can see we are getting a rmse of 9.98 which is much better than our baseline model we created

Before building our next machine learning models we would do some basic feature engineering to enhance the model

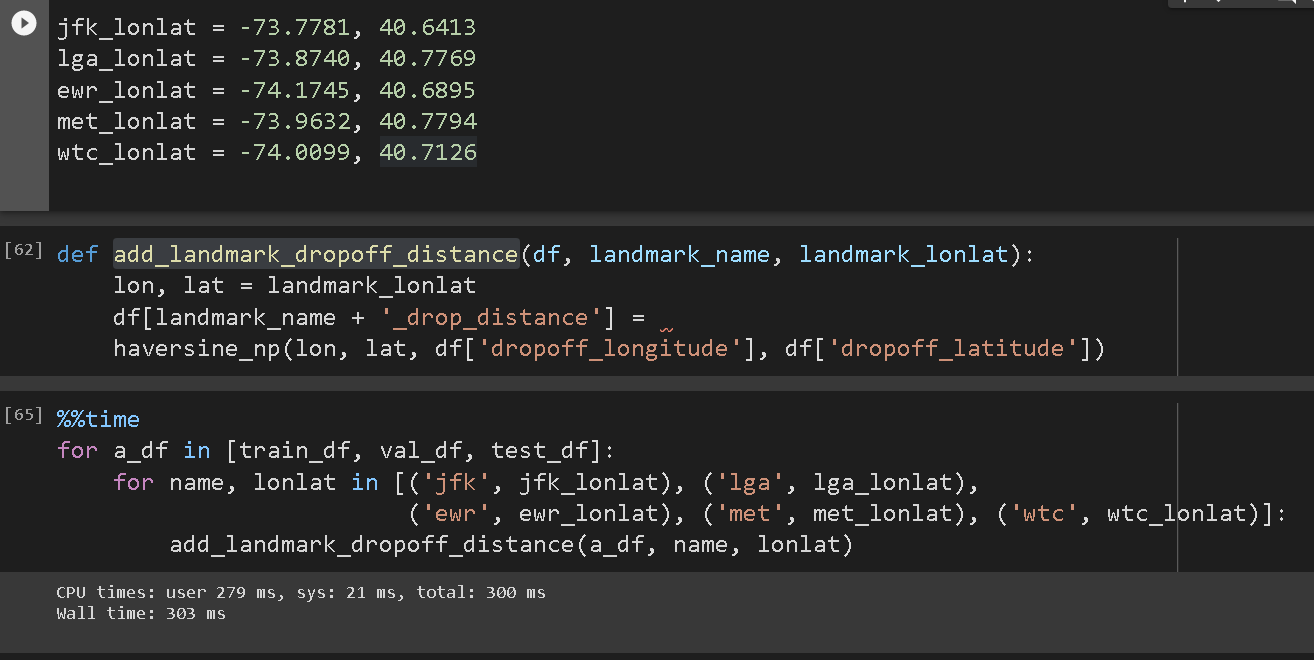
We would create a new feature called

“trip distance” to our training,validation,test dataset



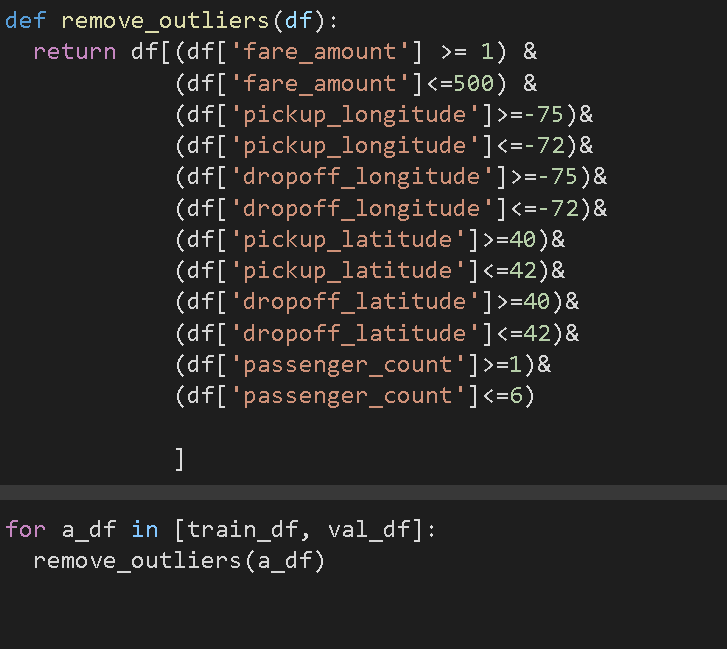


Now again we would create a feature called as add\_landmark\_dropoff\_distance



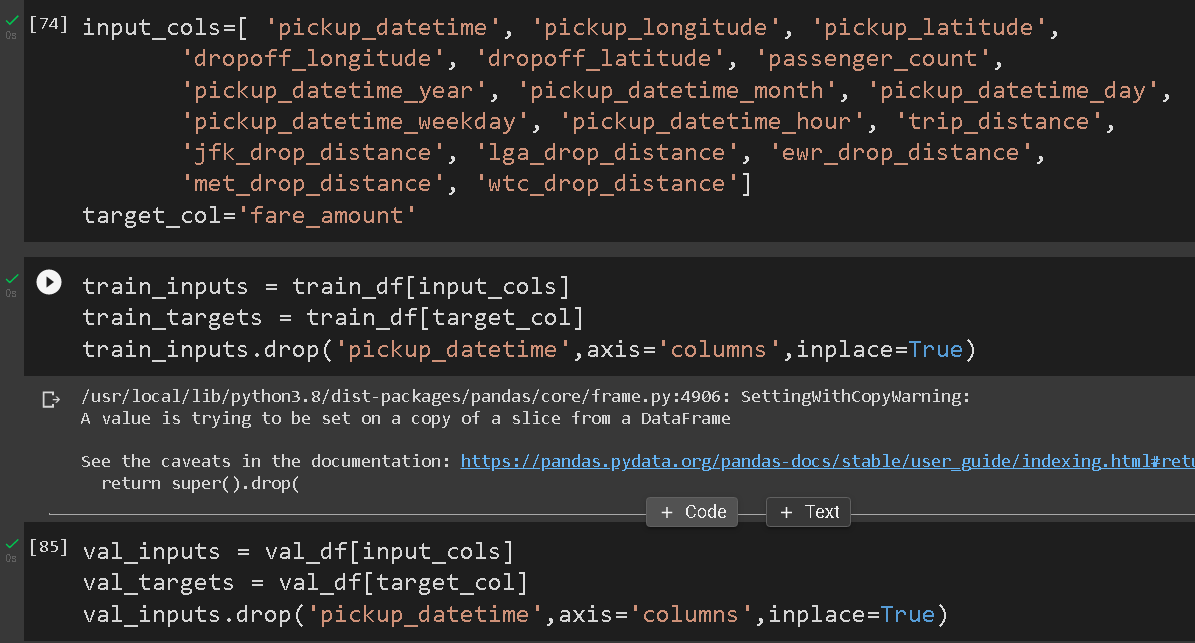
Now the next most important step is to remove the outlier because the outliers would massively impact our machine learning model and exponentially increases the root mean squared error which we do not want

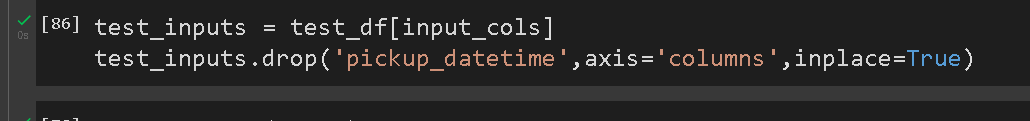
To happen with our model



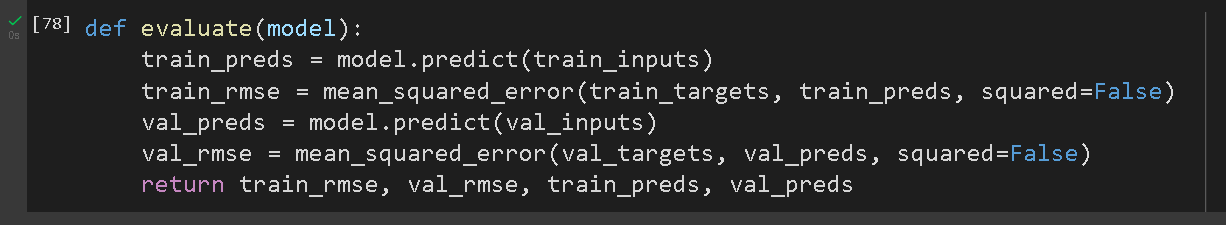
Now once again we would define our input columns for our training/validation/test data

And also the target column



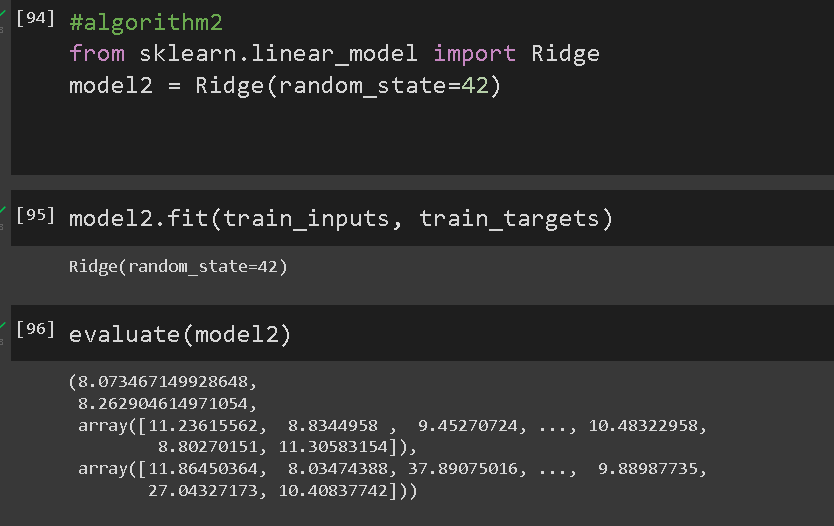


Now we would create a evaluate function for easily to reuse and find the rmse whenever we create a new machine learning model



Now we will create our second machine learning model

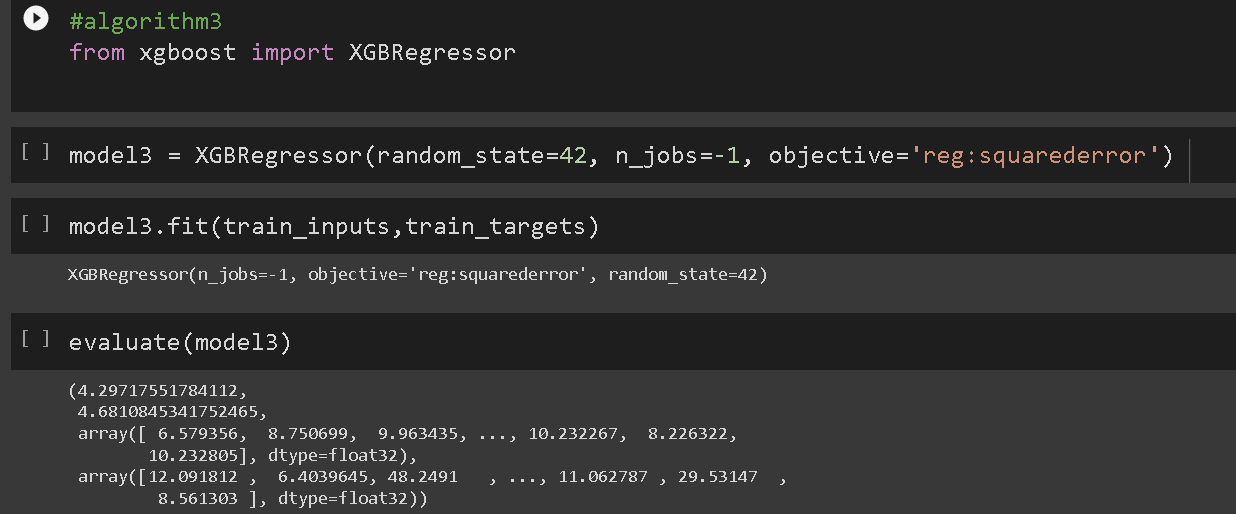
We will create a ridge regression model as our second algorithm



As now you can see the output we have improvised from 9.89 rmse to 8.07

So now again we would create our third machine learning model

And we would use the xbgboost algorithm to train our model



Result:

By building the xgboost regressor model we find a huge improvement as our root mean squared error reduced to 4.2 which is very good.So the xgboost algorithm worked very well than our Ridge regressor model and linear regression model