A Project Report

on

Prediction of Employees salary using Regression Technique

Submitted in partial fulfillment of the

requirement for the award of the degreeof

Bachelor of Technology in Computer Science and Engineering

GALGOTIAS UNIVERSITY

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

Under The Supervision of
Name of Supervisor: Ms. Anjum Mohd Aslam
Designation: Assistant Professor

Submitted By
HARSHIT SHUKLA(20SCSE1010256)

Project ID :- BT2303

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING / DEPARTMENT OF COMPUTERAPPLICATION
GALGOTIAS UNIVERSITY, GREATER NOIDA
INDIA

MAY,2022

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE'S DECLARATION

I/We hereby certify that the work which is being presented in the project entitled **"Prediction of Employees salary using Regression Technique"** in partial fulfillment of the requirements for the award of the **Bachelor of Technology in Computer Science and Engineering** submitted in the **School of Computing Science and Engineering** of Galgotias University, Greater Noida, is an original work carried out during the period of **JAN2022 TO MAY 2022**, under the supervision of ____________________________________

Department of Computer Science and Engineering, Galgotias University, Greater Noida

The matter presented in the thesis/project/dissertation has not been submitted by me/us for the award of any other degree of this or any other places.

HARSHIT SHUKLA(20SCSE1010256)

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Supervisor

CERTIFICATE

CERTIFICATE

The Final Project Viva-Voce examination of 20SCSE1010256: Harshit Shukla has been held on __________________and his/her work is recommended for the award of **Bachelor of Technology in Computer Science and Engineering.**

Signature of Examiner(s) **Signature of Supervisor(s)**

Signature of Project Coordinator **Signature of Dean**

Date:

Place: Greater Noida

Abstract:

Data of a company is very important for them. As analysis of data is very important for the growth of company. It is because company needs to follow the trend of market to be in market and grow. In a company or industry , managing their employees is one of the difficult and most important criteria ,in accordance to make more profit to the company with less use of resources. So, it became very necessary for any company to keep view of , how much they are paying and to whom and on the basis of current demand , it is required to analyze on the basis of previous salaries of employees ,that how much extra employee of different payee scales , are needed to meet the requirement of company . Company doesnot require these analysis only for hiring and firing the staff according to requirements but they also have to give promotions to their employees on the basis of their performance and abilities and also to manage all these such tasks . So there is a need of some techniques , methods , model etc to do analysis and may be that is the reason why companies spend huge amount in finding such type of techniques .

PROBLEM IDENTIFICATION:

Techniques , models for doing analysis on the dataset of a company are being improved from many years. Analysis is not new for this century, it is being done from thousand' s of years back. Athough, nowadays we are very advanced in doing this comparatively, But the hope of improvement is always there. Many companies are using different softwares,techniques, model , AI , etc to do different type of analysis . But we always try to improve the accuracy of models . So we need consistent improvement in accuracy of data models.

In accordance to grow in the market, companies do lots of analysis on the salary of their employees, for which they are always in search of such models which provide them better accuracy . So that they can easily manage their expenditure on their employees and the situation of growth of company as well. Forexample, if a company needs 100 software engineer for their new projects , then how they will decide that what range of payee scale ,they should offer the engineers so that company don't go in loss in worst condition and they can make profit . For this , they have to analyze their previous pattern of salary of employees and their growth and after analyzing they have to make decision which must be profitable for them. That is the reason why they always try to implement model which is more accurate .

PROPOSED SOLUTION:

So , here we are trying to make a model which can predict salary using machine learning and we are trying to improve the accuracy . Our model can work on different attributes for better prediction and we are using regression for the analyzation of data sets .

Regression analysis is a statistical method to model the relationship between a dependent (target) and independent (predictor) variables with one or more independent variables. More

specifically, Regression analysis helps us to understand how the value of the dependent variable is changing corresponding to an independent variable when other independent variables are held fixed.
Regression is a supervised learning technique which helps in finding the correlation between variables and enables us to predict the continuous output variable based on the one or more predictor variables. It is mainly used for **prediction, forecasting, time series modeling, and determining the causal-effect relationship between variables**.

Algorithm

Step 1: Import the dataset Step
2: Read the dataset Step
3: Calculate the total missing in each column of dataset Step
4: Perform Data cleaning Step
5: Imputing Missing Values Step
6: Data understanding through visualization (Compare every column with sales to observe which aspect is affecting sale of item) Step
7: Apply different regression technique and observe the result.

There are various types of regressions which are used in data science and machine learning. Each type has its own importance on different scenarios, but at the core, all the regression methods analyze the effect of the independent variable on dependent variables. Here we are discussing some important types of regression which are given below:

- **Linear Regression**
- **Logistic Regression**
- **Polynomial Regression**
- **Support Vector Regression**
- **Decision Tree Regression**
- **Random Forest Regression**
- **Ridge Regression**
- **Lasso Regression:**

As of now we are using Linear regression , Lasso regression ad Ridge Regressio

in our model . So Lets understand these three regression in detail.

Linear Regression:

- Linear regression is a statistical regression method which is used for predictive analysis.

- It is one of the very simple and easy algorithms which works on regression and shows the relationship between the continuous variables.
- Linear regression shows the linear relationship between the independent variable (X- axis) and the dependent variable (Y-axis), hence called linear regression.

Linear Regression is a machine learning algorithm based on supervised learning. It performs a regression task. Regression models a target prediction value based on independent variables. It is mostly used for finding out the relationship between variables and forecasting. Different regression models differ based on – the kind of relationship between dependent and independent variables, they are considering and the number of independent variables being used.

Linear regression performs the task to predict a dependent variable value (y) based on a given independent variable (x). So, this regression technique finds out a linear relationship between x (input) and y(output). Hence, the name is Linear Regression.

In the figure above, X (input) is the work experience and Y (output) is the salary of a person. The regression line is the best fit line for our model

Hypothesis function for Linear Regression :

$$y = \theta_1 + \theta_2.x$$

While training the model we are given :

x: input training data (univariate – one input

variable(parameter)) y: labels to data (supervised learning

When training the model – it fits the best line to predict the value of y for a given value of x. The model gets the best regression fit line by finding the best θ1 and θ2 values.

θ1: intercept

θ2: coefficient of x

Once we find the best θ1 and θ2 values, we get the best fit line. So when we are finally using our model for prediction, it will predict the value of y for the input value

of x. Cost Function (J):

By achieving the best-fit regression line, the model aims to predict y value such that the error difference between predicted value and true value is minimum. So, it is very important to update the θ1 and θ2 values, to reach the best value that minimize the error between predicted y value (pred) and true y value (y).

Cost function(J) of Linear Regression is the Root Mean Squared Error (RMSE) between predicted y value (pred) and true y value (y).

Gradient Descent:

To update θ1 and θ2 values in order to reduce Cost function (minimizing RMSE value) and achieving the best fit line the model uses Gradient Descent. The idea is to start with random θ1 and θ2 values and then iteratively updating the values, reaching minimum cost.

Lasso Regression:

- Lasso regression is another regularization technique to reduce the complexity of the model.
- It is similar to the Ridge Regression except that penalty term contains only the absolute weights instead of a square of weights.
- Since it takes absolute values, hence, it can shrink the slope to 0, whereas Ridge Regression can only shrink it near to 0.
- It is also called as **L1 regularization**. The equation for Lasso regression will be:

$$\mathrm{L}(x, y) = \mathrm{Min}\left(\sum_{i=1}^{n} (y_i - w_i\, x_i)^2 + \lambda \sum_{i=1}^{n} |\mathbf{w_i}\ | \right)$$

Ridge Regression:

A Ridge regressor is basically a regularized version of a Linear Regressor. i.e to the original cost function of linear regressor we add a regularized term that forces the learning algorithm to fit the data and helps to keep the weights lower as possible. The regularized term has the

parameter ‘alpha’ which controls the regularization of the model i.e helps in reducing the variance of the estimates.
Cost Function for Ridge Regressor.

Here we are using Jupyter(Python) Notebook for the implementation of our model because python has huge varieties of libraries and it is considered as one of the best language for the machine learning and AI (Artificial Intelligence).
Some of the library that we are using :-

Numpy

This is used for working in domain of linear algebra , transformation , matrices etc.

Pandas

This is used for the manipulation in data and analyzing that. For example , we have used this to import data frames in the model.

HARDWARE & SOFWARE REQUIREMENTS

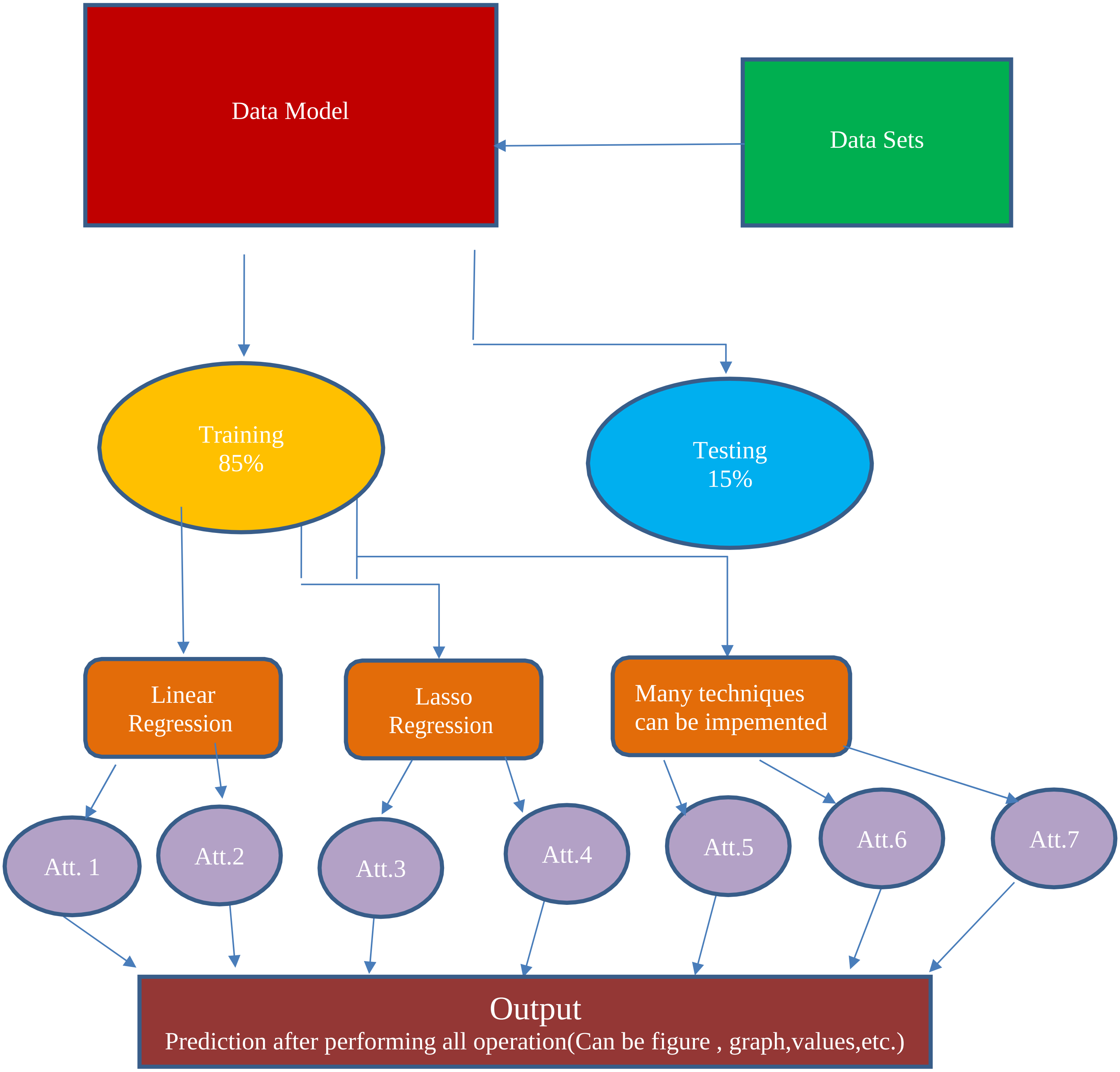
Hardware requirements:

1. *Laptop/PC/Any*

computer Software requirements:

1.pyhton should be installed and environment variable should be set.
*2.*all required libraries should be installed.

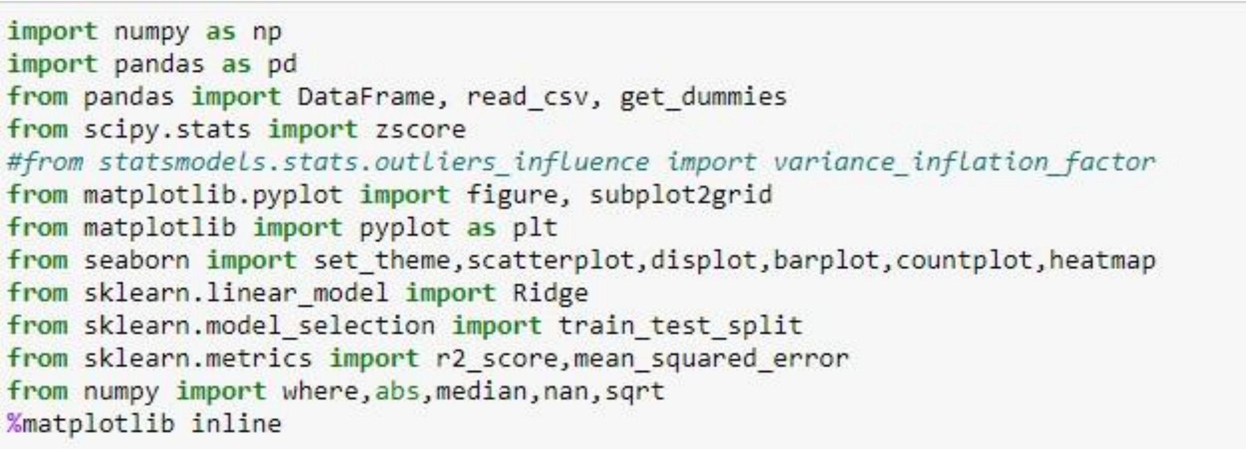
BLOCK DIAGRAM & DESCRIPTION:



This block diagram shows that data model takes dataset as input and can perform multiple n techniques like regression etc , and it can work on many attributes as much as user want and provide the result on the basis of previous data and predict the salary

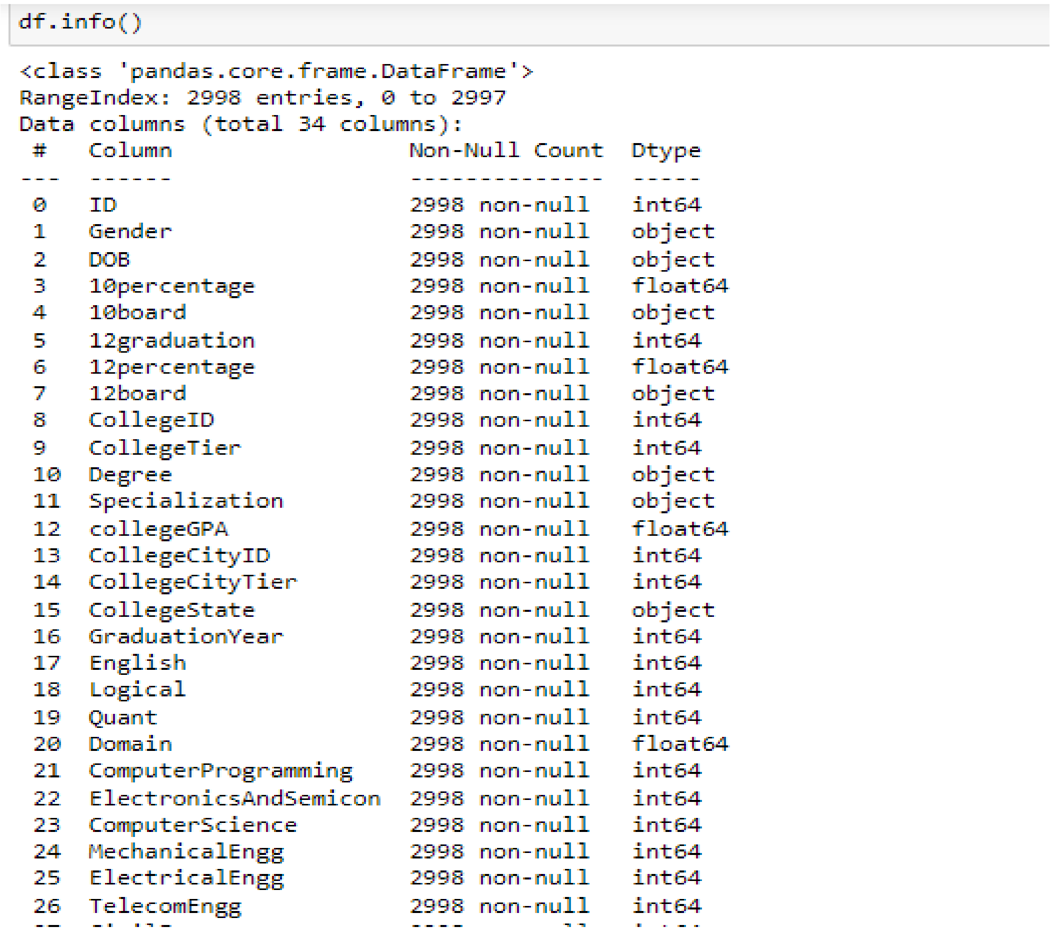
APPROACH FOLLOWED:

1.IMPORTING ALL THE REQUIRED LIBRARIES



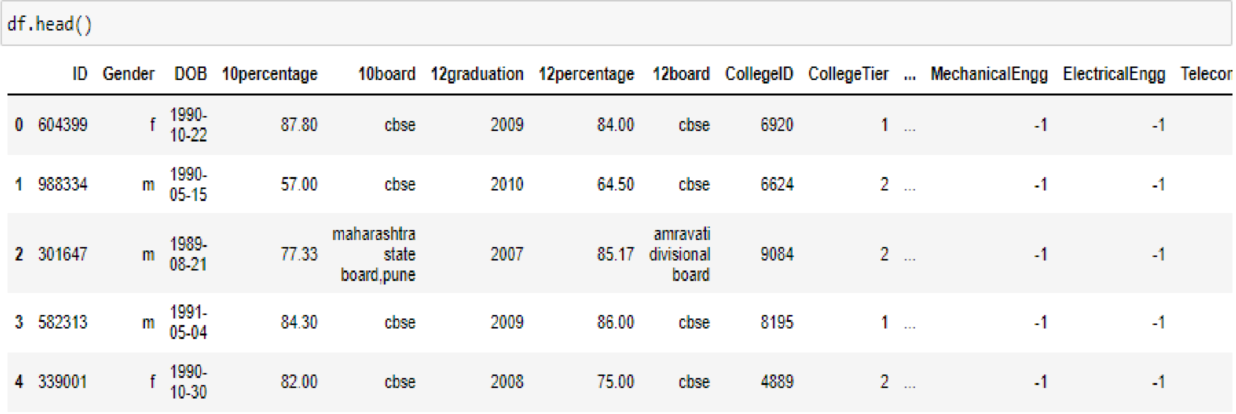
```
import numpy as np
import pandas as pd
from pandas import DataFrame, read_csv, get_dummies
from scipy.stats import zscore
#from statsmodels.stats.outliers_influence import variance_inflation_factor
from matplotlib.pyplot import figure, subplot2grid
from matplotlib import pyplot as plt
from seaborn import set_theme,scatterplot,displot,barplot,countplot,heatmap
from sklearn.linear_model import Ridge
from sklearn.model_selection import train_test_split
from sklearn.metrics import r2_score,mean_squared_error
from numpy import where,abs,median,nan,sqrt
%matplotlib inline
```

2.Dataset and its processing:



```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2998 entries, 0 to 2997
Data columns (total 34 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   ID                     2998 non-null   int64
 1   Gender                 2998 non-null   object
 2   DOB                    2998 non-null   object
 3   10percentage           2998 non-null   float64
 4   10board                2998 non-null   object
 5   12graduation           2998 non-null   int64
 6   12percentage           2998 non-null   float64
 7   12board                2998 non-null   object
 8   CollegeID              2998 non-null   int64
 9   CollegeTier            2998 non-null   int64
 10  Degree                 2998 non-null   object
 11  Specialization         2998 non-null   object
 12  collegeGPA             2998 non-null   float64
 13  CollegeCityID          2998 non-null   int64
 14  CollegeCityTier        2998 non-null   int64
 15  CollegeState           2998 non-null   object
 16  GraduationYear         2998 non-null   int64
 17  English                2998 non-null   int64
 18  Logical                2998 non-null   int64
 19  Quant                  2998 non-null   int64
 20  Domain                 2998 non-null   float64
 21  ComputerProgramming    2998 non-null   int64
 22  ElectronicsAndSemicon  2998 non-null   int64
 23  ComputerScience        2998 non-null   int64
 24  MechanicalEngg         2998 non-null   int64
 25  ElectricalEngg         2998 non-null   int64
 26  TelecomEngg            2998 non-null   int64
```



```
df.head()
```

	ID	Gender	DOB	10percentage	10board	12graduation	12percentage	12board	CollegeID	CollegeTier	...	MechanicalEngg	ElectricalEngg	Telecor
0	604399	f	1990-10-22	87.80	cbse	2009	84.00	cbse	6920	1	...	-1	-1	
1	988334	m	1990-05-15	57.00	cbse	2010	64.50	cbse	6624	2	...	-1	-1	
2	301647	m	1989-08-21	77.33	maharashtra state board,pune	2007	85.17	amravati divisional board	9084	2	...	-1	-1	
3	582313	m	1991-05-04	84.30	cbse	2009	86.00	cbse	8195	1	...	-1	-1	
4	339001	f	1990-10-30	82.00	cbse	2008	75.00	cbse	4889	2	...	-1	-1	

3. CLEANING OF DATASET:

After cleaning of dataset Preprocessing of this dataset includes doing analysis on the independent variables like checking for null values in each column and then replacing or filling them with supported appropriate data types, so that analysis and model fitting is not hindered from its way to accuracy. Shown above are some of the representations obtained by using Pandas tools which tells about variable count for numerical columns and modal values for categorical columns.

Matplot

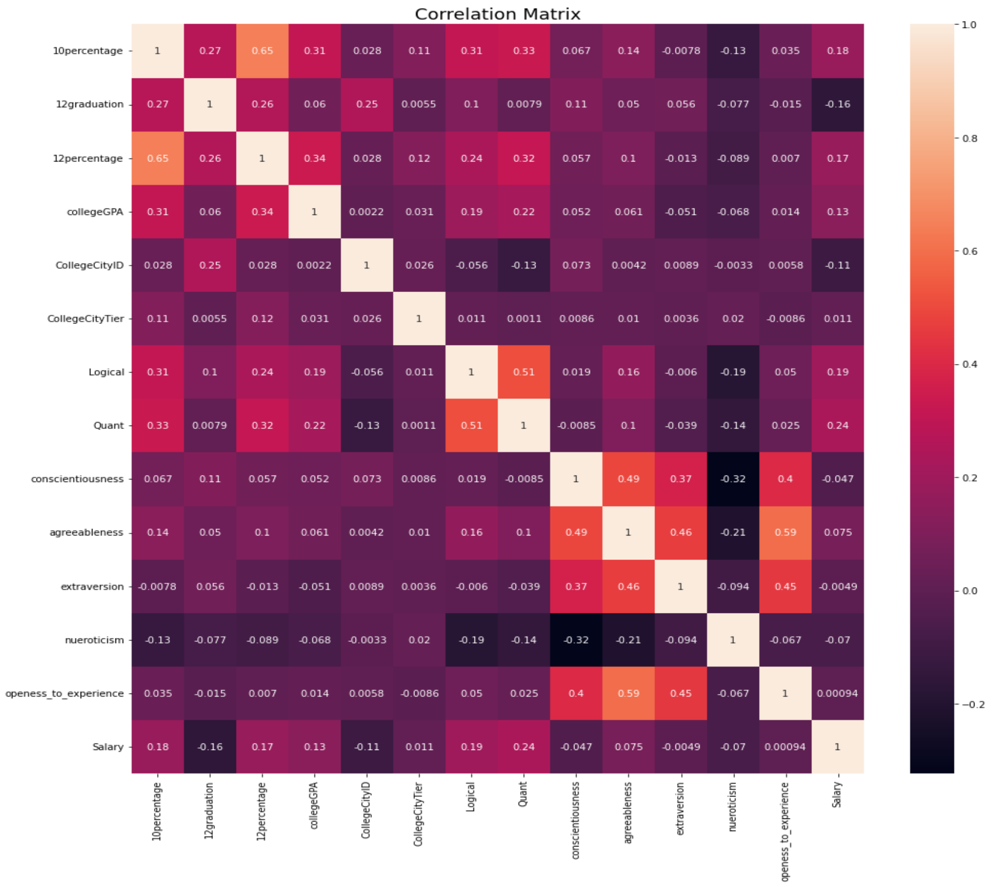
This is used for plotting figures and graph.

Seaborn

This is used for doing customization in themes colors etc.

sklearn.linear_model

We are using Correlation matrix for testing the accuracy of our model.
Using this correlation matrix we are trying to check our accuracy. On this basis we have compared the results of both the regression technique.
There are various no.of attribute available :-
'ID', 'Gender', 'DOB', '10percentage', '10board', '12graduation', '12percentage', '12board', 'CollegeID', 'CollegeTier ', 'Degree', 'Specialization', 'collegeGPA', 'CollegeCityID', 'CollegeCityTier', 'CollegeState ', 'GraduationYear', 'English', 'Logical', 'Quant', 'Domain', 'ComputerProgramming', 'ElectronicsAndSemicon', 'ComputerScience', 'MechanicalEngg', 'ElectricalEngg', 'TelecomEngg', 'CivilEngg', 'conscientiousness', 'agreeableness', 'extraversion', 'nueroticism', 'openess_to_experience', 'Salary'], dtype='object'

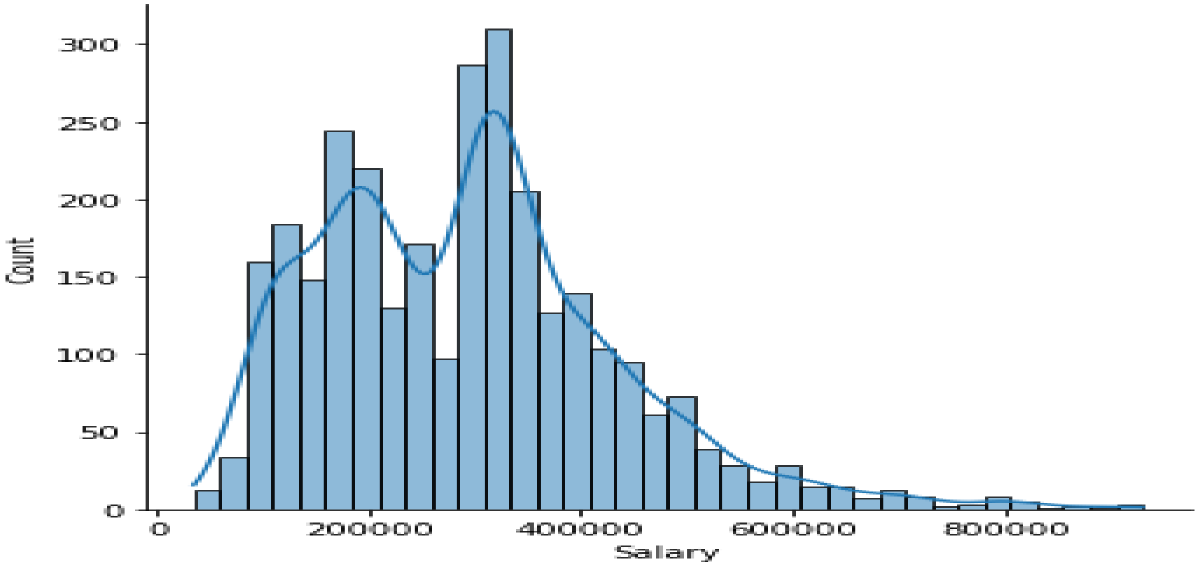


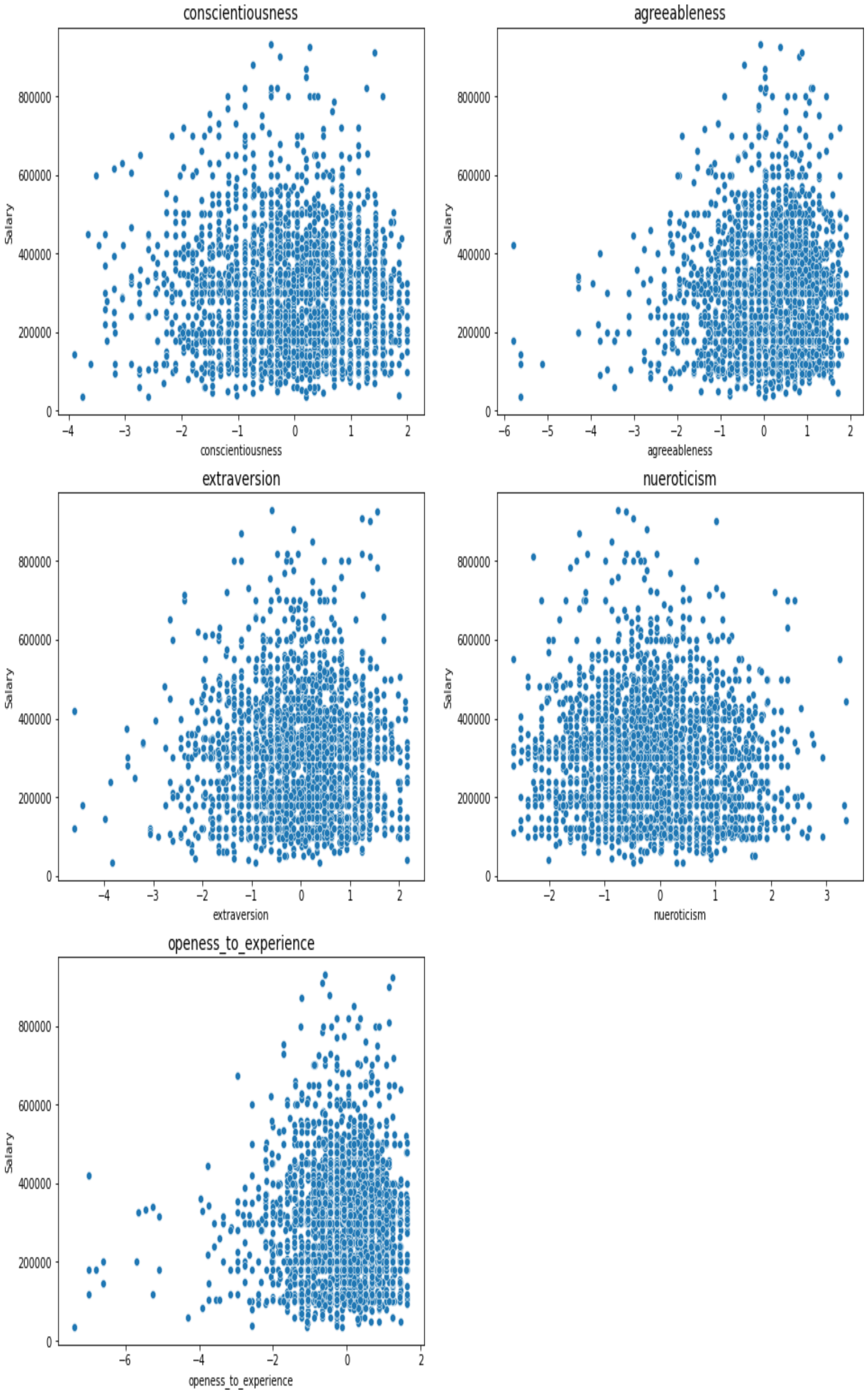
But we don’t need all of them in our model . SO we will only use relevant attributes and drop the rest of attributres. So the following attribute we are using :-
'10percentage', '10board', '12graduation', '12percentage',
'12board', 'Degree', 'Specialization', 'collegeGPA',
'CollegeCityID',

```
       'CollegeCityTier', 'Logical', 'Quant', 'conscientiousness',
       'agreeableness', 'extraversion', 'nueroticism',
       'openess_to_experience', 'Salary'],
      dtype='object'
```

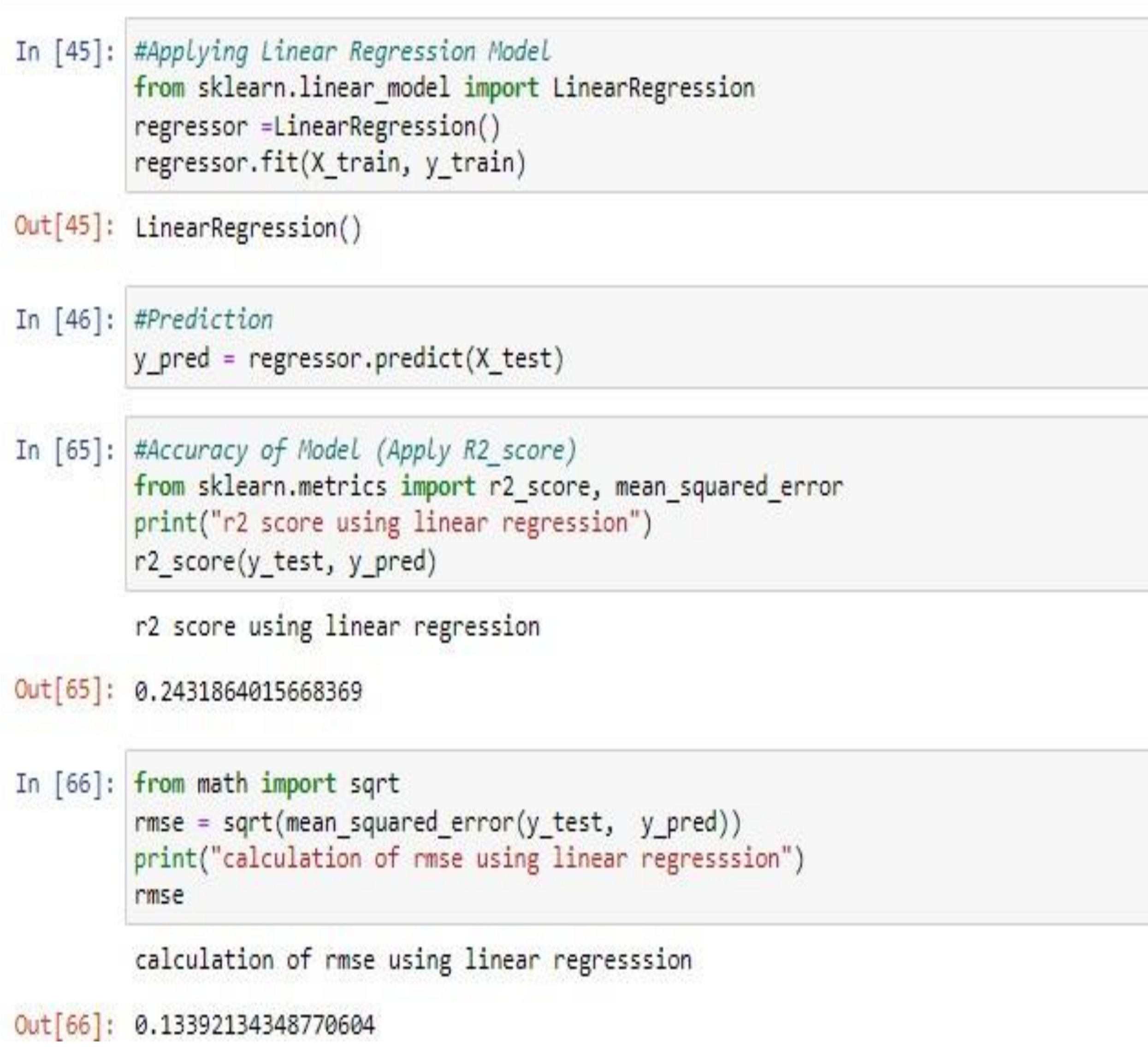
Further we have continuously normalized the model until it only shows relevant Attribute.

Visualizing the skewness of data:

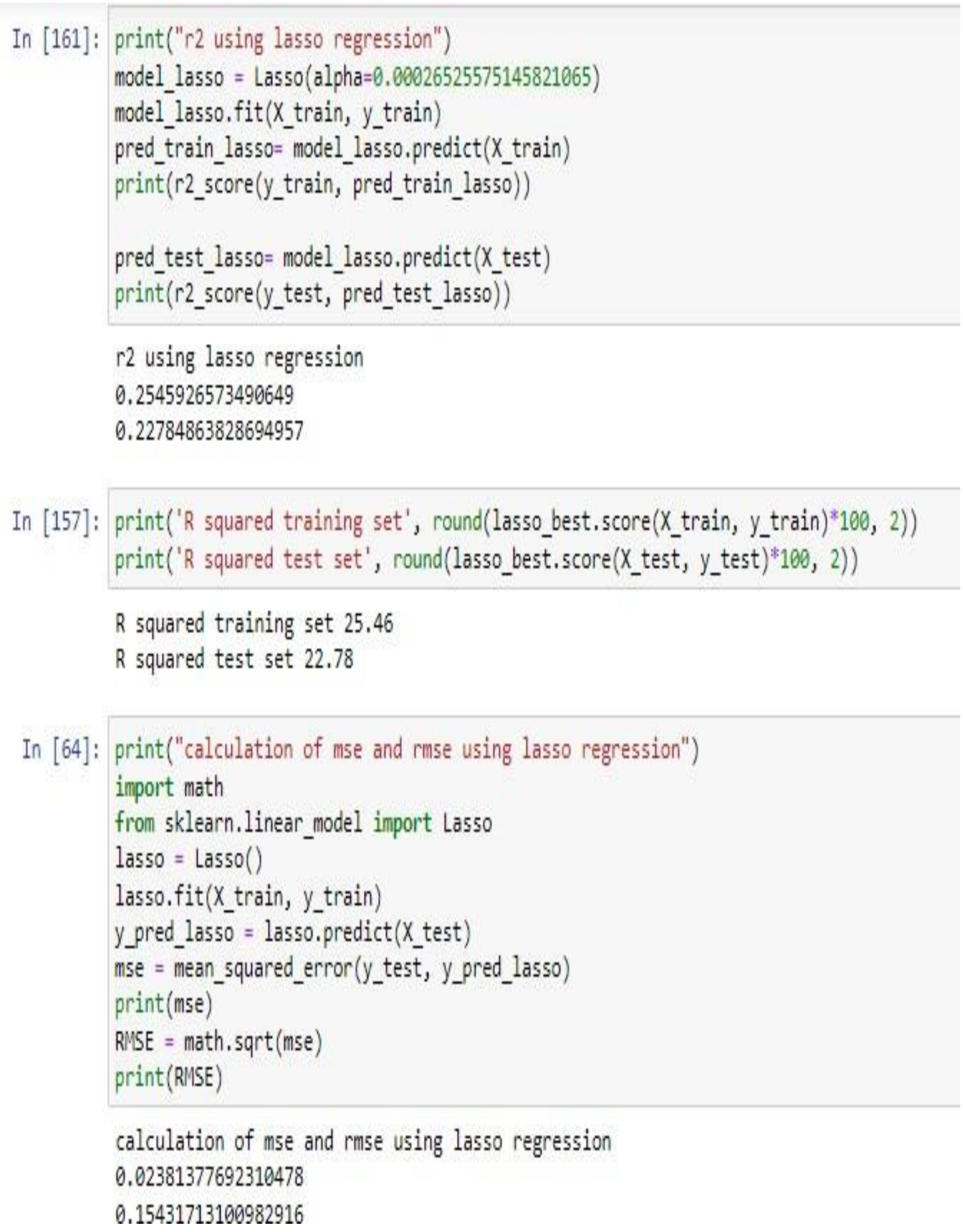



conscientiousness
Salary
800000
600000
400000
200000
0
−4
−3
−2
−1
0
1
2
conscientiousness
agreeableness
Salary
800000
600000
400000
200000
0
−6
−5
−4
−3
−2
−1
0
1
2
agreeableness
extraversion
Salary
800000
600000
400000
200000
0
−4
−3
−2
−1
0
1
2
extraversion
nueroticism
Salary
800000
600000
400000
200000
0
−2
−1
0
1
2
3
nueroticism
openess_to_experience
Salary
800000
600000
400000
200000
0
−6
−4
−2
0
2
openess_to_experience

ImplementationofMachineLearningModels:



```
In [45]: #Applying Linear Regression Model
         from sklearn.linear_model import LinearRegression
         regressor =LinearRegression()
         regressor.fit(X_train, y_train)

Out[45]: LinearRegression()

In [46]: #Prediction
         y_pred = regressor.predict(X_test)

In [65]: #Accuracy of Model (Apply R2_score)
         from sklearn.metrics import r2_score, mean_squared_error
         print("r2 score using linear regression")
         r2_score(y_test, y_pred)

         r2 score using linear regression

Out[65]: 0.2431864015668369

In [66]: from math import sqrt
         rmse = sqrt(mean_squared_error(y_test,  y_pred))
         print("calculation of rmse using linear regresssion")
         rmse

         calculation of rmse using linear regresssion

Out[66]: 0.13392134348770604
```



```
In [161]: print("r2 using lasso regression")
          model_lasso = Lasso(alpha=0.00026525575145821065)
          model_lasso.fit(X_train, y_train)
          pred_train_lasso= model_lasso.predict(X_train)
          print(r2_score(y_train, pred_train_lasso))

          pred_test_lasso= model_lasso.predict(X_test)
          print(r2_score(y_test, pred_test_lasso))
```

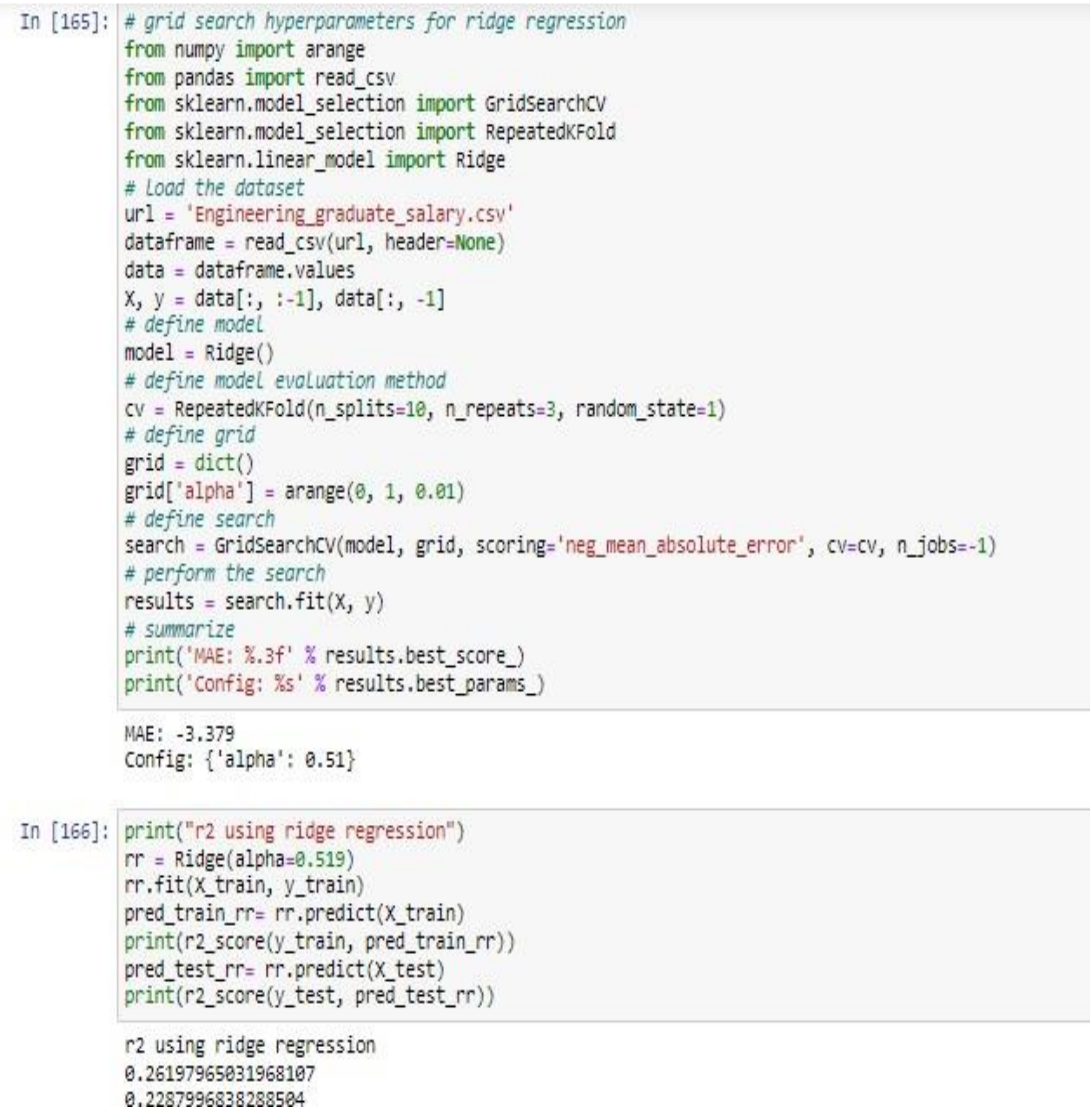
```
r2 using lasso regression
0.2545926573490649
0.22784863828694957
```

```
In [157]: print('R squared training set', round(lasso_best.score(X_train, y_train)*100, 2))
          print('R squared test set', round(lasso_best.score(X_test, y_test)*100, 2))
```

```
R squared training set 25.46
R squared test set 22.78
```

```
In [64]: print("calculation of mse and rmse using lasso regression")
         import math
         from sklearn.linear_model import Lasso
         lasso = Lasso()
         lasso.fit(X_train, y_train)
         y_pred_lasso = lasso.predict(X_test)
         mse = mean_squared_error(y_test, y_pred_lasso)
         print(mse)
         RMSE = math.sqrt(mse)
         print(RMSE)
```

```
calculation of mse and rmse using lasso regression
0.02381377692310478
0.15431713100982916
```



```
In [165]: # grid search hyperparameters for ridge regression
          from numpy import arange
          from pandas import read_csv
          from sklearn.model_selection import GridSearchCV
          from sklearn.model_selection import RepeatedKFold
          from sklearn.linear_model import Ridge
          # load the dataset
          url = 'Engineering_graduate_salary.csv'
          dataframe = read_csv(url, header=None)
          data = dataframe.values
          X, y = data[:, :-1], data[:, -1]
          # define model
          model = Ridge()
          # define model evaluation method
          cv = RepeatedKFold(n_splits=10, n_repeats=3, random_state=1)
          # define grid
          grid = dict()
          grid['alpha'] = arange(0, 1, 0.01)
          # define search
          search = GridSearchCV(model, grid, scoring='neg_mean_absolute_error', cv=cv, n_jobs=-1)
          # perform the search
          results = search.fit(X, y)
          # summarize
          print('MAE: %.3f' % results.best_score_)
          print('Config: %s' % results.best_params_)
```

```
MAE: -3.379
Config: {'alpha': 0.51}
```

```
In [166]: print("r2 using ridge regression")
          rr = Ridge(alpha=0.519)
          rr.fit(X_train, y_train)
          pred_train_rr= rr.predict(X_train)
          print(r2_score(y_train, pred_train_rr))
          pred_test_rr= rr.predict(X_test)
          print(r2_score(y_test, pred_test_rr))
```

```
r2 using ridge regression
0.26197965031968107
0.2287996838288504
```

RESULT AND CONCLUSION:

R2 using linear regression is 0.2431864015668369
R2 using lasso regression is 0.22784863828694957
R2 using ridge regression is 0.2287996838288504
Root mean square error using linear regression is 0.13392134348770604
Root mean square error using lasso regression is 0.15431713100982916
Root mean square error using ridge regression is 0.13517267240707206

We can say that
the linear regression is providing better result
with r2 score of 24.31 and with minimum root mean square error of 0.1339
The lasso and ridge Regression model is providing the nearly same value but less than linear.
so we will refer linear regression method to predict the salary of the employees.

FUTURE SCOPE:

As of now we have only implemented lasso and linear regression, but in future we can implement as many new techniques of regression or some other technique .

Also we have tested our model with some combinations of attribute, but we can use N no. of combinations of attribute which can produce different outputs and more accuracy y can be achieved.

GITHUB LINK:

https://github.com/riddhi1305/INTEL-SALARY-PREDICTION-PROJECT.git