PREDICTING INSURANCE COST

USING MACHINE LEARNING MODELS

LOYOLA ANDHRA COLLEGE FAITH DUTY	GEETA SATYA SURYA TEJA ADINA
MANIPAL INSTITUTE OF TECHNOLOGY MANIPAL *A Constituent Institution of Manipal University*	KARTHIK GOGISETTY
NITTE EDUCATION TRUST NMAM INSTITUTE OF TECHNOLOGY	SIDDHI SANJAY SHENOY
INSTITUTE OF SCIENCE POONA'S (I.D.N6. PU/PN/SC.1531200) COLLEGE OF COMPUTER SCIENCES (Affiliated to Savitribai Phule Pune University, and Recognized by Gov. of Maharashtra)	VAIBHAVI MOHAN UBHE

Knowledge Solutions India

Skill development | Certification | Placement prep

Contents

1. ABSTRACT:

Insurance is a policy that eliminates or decreases loss costs occurred by various risks. Various factors influence the cost of insurance. These considerations contribute to the insurance policy formulation. Machine Learning (ML) for the insurance industry sector can make the wording of insurance policies more efficient.

2. INTRODUCTION:

We are on a planet full of threats and uncertainty people, households, companies, properties, and property are exposed to different risk forms and the risk levels can vary. These dangers contain the risk of death, health, and property loss or assets. Life and wellbeing are the greatest parts of people's lives. But risks cannot usually be avoided, so the world of finance has developed numerous products to shield individuals and organizations from these risks by using financial capital to reimburse them. Insurance is, therefore, a policy that decreases or eliminates the loss incurred by various risks. Concerning the value of insurance in the lives of individuals, it becomes important for the insurance companies to be sufficiently precise to measure or quantify the amount covered by this policy and the insurance charges which must be paid for it. Various important factors estimate these charges. Different tools are used to calculate insurance premium. ML is beneficial here. ML may generalize the effort or method to formulate the policy. These ML models can be learned by themselves. Past records are fed into the model. The requisite factors to measure the payments can then be defined as the model inputs, then the model can correctly anticipate insurance policy costs. This decreases human effort and resources and improves the company's profitability. Thus, the accuracies can be improved with ML. Our objective is to forecast insurance charges in this article. The value of insurance fees is based on different variables. As a result, insurance fees are continuous values. The regression is the best choice available to fulfil our needs. We use various regression models in this analysis since there are many independent variables used to calculate the dependent (target) variable. For this study, the dataset for cost of health insurance is used. Pre-processing of the dataset is done first. Then we trained regression models with training data and finally evaluated these models based on testing data.

3. DATASET:

The data set includes seven attributes, the data set is separated into two-part the first part called training data, and the second called test data; training data makes up about 80 percent of the total data used, and the rest for test data. The training data set is applied to build a model as a predictor of medical insurance cost year and the test set will use to evaluate the regression model. The following table shows the Description of the Dataset.

NAME	DESCRIPTION
age	Age of primary beneficiary
sex	Insurance contractor gender, female, male
BMI	Body mass index, providing an understanding of body, weights that are relatively high or low relative to height, objective index of body weight (kg / m ^ 2) using the ratio of height to weight, ideally 18.5 to 24.9
children	Number of children covered by health insurance / Number of dependents
smoker	Smoking
region	The beneficiary's residential area in the US, northeast, southeast, southwest, northwest
charges	Individual medical costs billed by health insurance

4. SOFTWARE LIBRARIES USED:

1. Numpy
2. Pandas
3. Seaborn
4. Matplotlib
5. Sklearn

5. DATA PREPROCESSING:

Each one of the attributes, contribute in estimating the cost of insurance (dependent variable). In this stage, the data was scrutinized and updated properly to efficiently source the data to the ML algorithms. First, null values and duplicates were checked for. There were no null values, but one duplicate entry was found and cleared. The target variable (charges) was then examined.

Changing the categorical variables into numeric values

gender	Male/Female 1 = Male, 0= Female
smoker	whether a client is a smoker or not 1=yes 0=no
region	where the client lives 1= southwest 2= southeast 3= northwest 4= northeast

Studying skewness of the target (charges) variable:

The measures of central tendency for the charges columns is as follows:

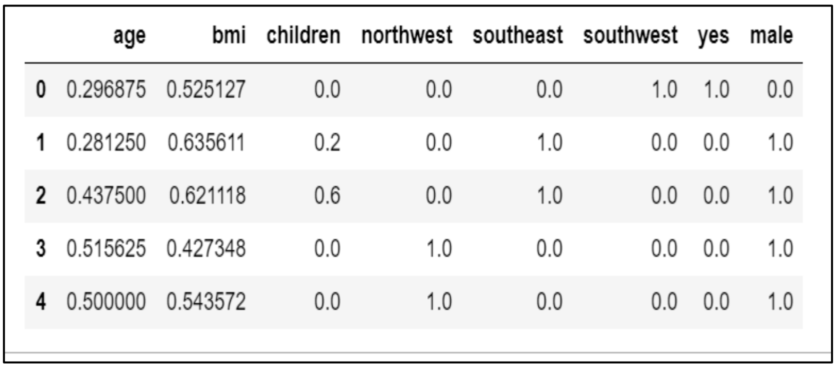
mean	median	std	min	max	1st quartile	3rd quartile
13279.12	9386.16	12110.36	1121.87	63770.43	4746.34	16657.72

Because the mean value is greater than the median, as shown in the above Table, this implies that the distribution of health insurance charges is positively skewed (right-skewed).

Normalizing:

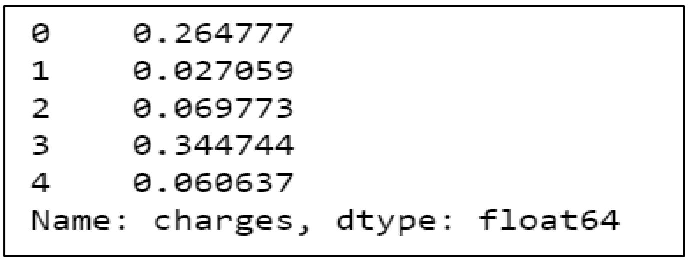
To get the values in a fixed range, we used MaxAbs scalar to scale the data. Normalization refers to rescaling real-valued numeric attributes into a 00 to 11 range. Data normalization is used in random forest to make model training less sensitive to the scale of features. It leads to a more accurate model allows our model to converge to the better weights.

Normalization for x column



	age	bmi	children	northwest	southeast	southwest	yes	male
0	0.296875	0.525127	0.0	0.0	0.0	1.0	1.0	0.0
1	0.281250	0.635611	0.2	0.0	1.0	0.0	0.0	1.0
2	0.437500	0.621118	0.6	0.0	1.0	0.0	0.0	1.0
3	0.515625	0.427348	0.0	1.0	0.0	0.0	0.0	1.0
4	0.500000	0.543572	0.0	1.0	0.0	0.0	0.0	1.0

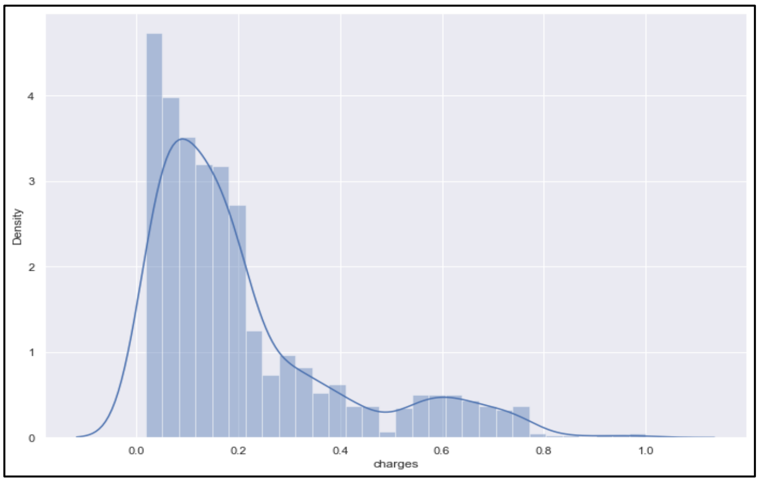
Normalization for y column



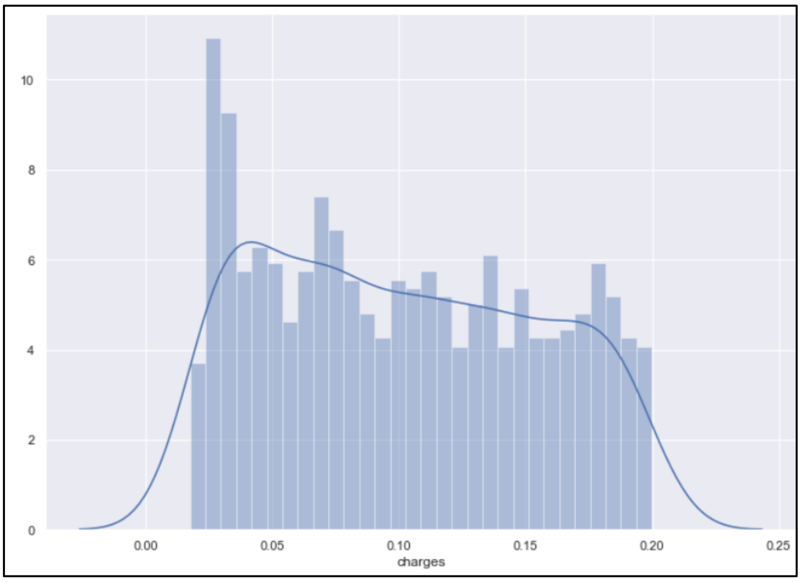
```
0    0.264777
1    0.027059
2    0.069773
3    0.344744
4    0.060637
Name: charges, dtype: float64
```

Removing outliers:

We can see there are outliers in the target variable(charges) in the below figure.

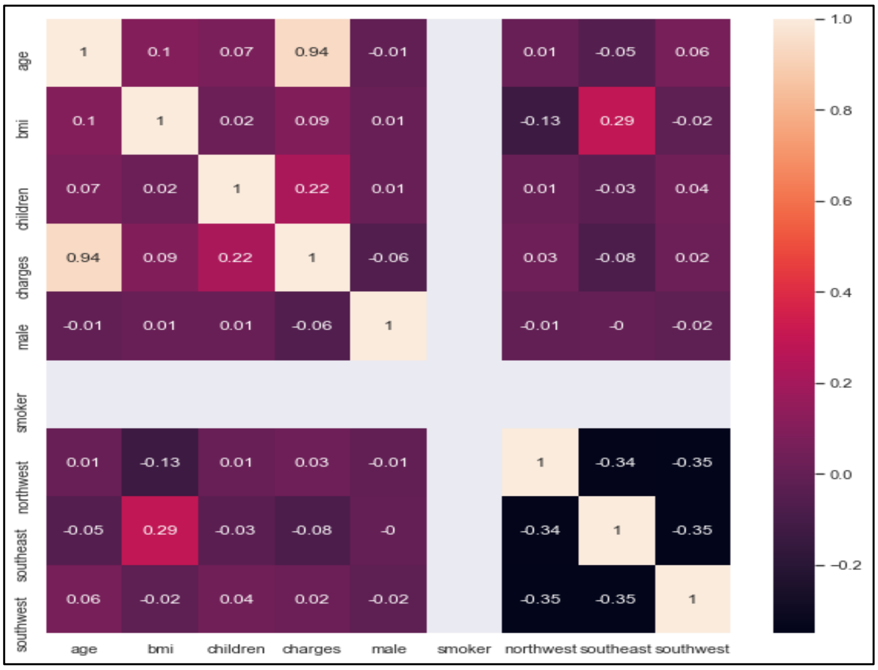


We removed the outliers by using drop function which are greater than 0.2



Data Visualization:

Then we check the correlation between the variables.



	age	bmi	children	charges	male	smoker	northwest	southeast	southwest
age	1	0.1	0.07	0.94	-0.01		0.01	-0.05	0.06
bmi	0.1	1	0.02	0.09	0.01		-0.13	0.29	-0.02
children	0.07	0.02	1	0.22	0.01		0.01	-0.03	0.04
charges	0.94	0.09	0.22	1	-0.06		0.03	-0.08	0.02
male	-0.01	0.01	0.01	-0.06	1		-0.01	-0	-0.02
smoker									
northwest	0.01	-0.13	0.01	0.03	-0.01		1	-0.34	-0.35
southeast	-0.05	0.29	-0.03	-0.08	-0		-0.34	1	-0.35
southwest	0.06	-0.02	0.04	0.02	-0.02		-0.35	-0.35	1

6. APPLICATION OF VARIOUS MACHINE LEARNING MODELS:

6.1. Multiple Linear Regression

Multiple linear regression (MLR) is a statistical technique that uses several explanatory variables to predict the outcome of a response variable. The goal of MLR is to model the linear relationship between the explanatory (independent) variables and response (dependent) variable.

Formula: $y = b_1x_1 + b_2x_2 + \dots + b_nx_n + c$.

Where, c is constant

$x_1, x_2, \dots\dots\dots\dots, x_n$ are the independent variables

$b_1, b_2, \dots\dots\dots\dots, b_n$ are the weights of the independent variables

Defining x and y:

As we can see in the above correlation matrix, the Variables that most influence charges are age, bmi, children, smoker. And we take them as x. The charges variable is taken as y.

Splitting of data into training dataset and testing dataset:

The whole dataset is divided into two parts. 70% of the dataset for training dataset and 30% for testing dataset.

Fitting the data into the multiple linear regression model:

We have fitted the data into the multiple linear regression model. Now we can predict the y values.

Performance of the model:

For calculating the performance of the model, we used certain metrics which are Mean Squared Error, Root Mean Squared Error and R2 score.

For training set:

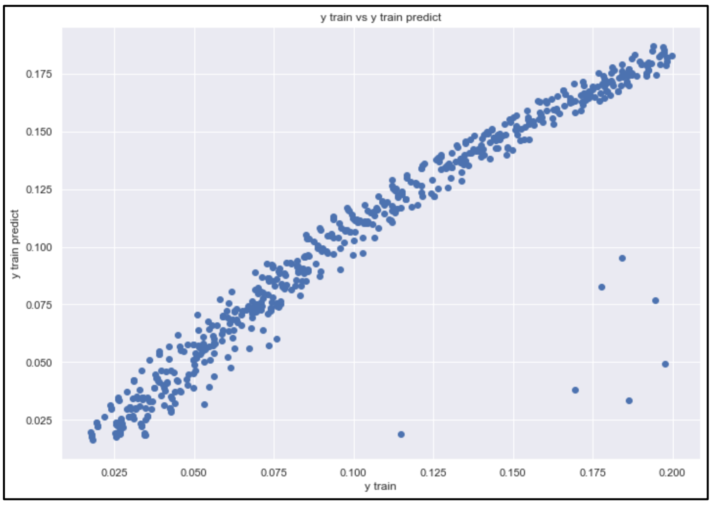
MSE	0.00023910845804612973
RMSE	0.015463132219771313
R2	0.9157266616349637

For testing set:

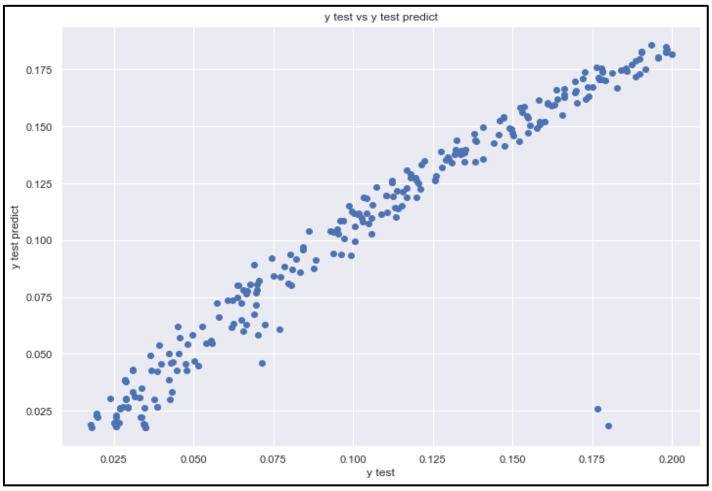
MSE	0.00026006526351051 08
RMSE	0.016126539105167 94
R2	0.9087636286747932

Plotting the actual values and predicted values:

For training set:



For testing set:



INFERENCE:

From the above graphs, we can state that the model is best fitted as there is a straight line in the scatter plot when we plotted against y_train and y_train_predict values and also against y_test and y_test_predict.

The R2score for the test dataset is 91.57% and for the training set is around 90.87% which seems to have no over-fitting problem after removing outliers, feature selection and normalizing the data accordingly.

6.2. Random Forest Regression

Random Forest Regression is a supervised learning algorithm that uses the technique of ensemble learning method which combines predictions from multiple machine learning algorithms to make more accurate predictions than a single model for regression.

Decision trees has variance, but when we combine the multiple decision trees the resultant variance is much lower as each decision tree is trained for a given data set and there by our output depends on multiple decision trees. Where the final output depends on every decision tree used for training. In classification-based problems we can take the majority count to give a result. In regression we can take the mean or any other mathematically equivalent result for final output. A random forest is a technique that can perform both regression and classification tasks with the use of the multiple decision trees explained. The basic concept remains the same that is to reply on multiple decision trees to decide an outcome than depending on a single decision tree.

Performance of the model:

For calculating the performance of the model, we used certain metrics which are Mean Squared Error, Root Mean Squared Error and R2 score.

For training set:

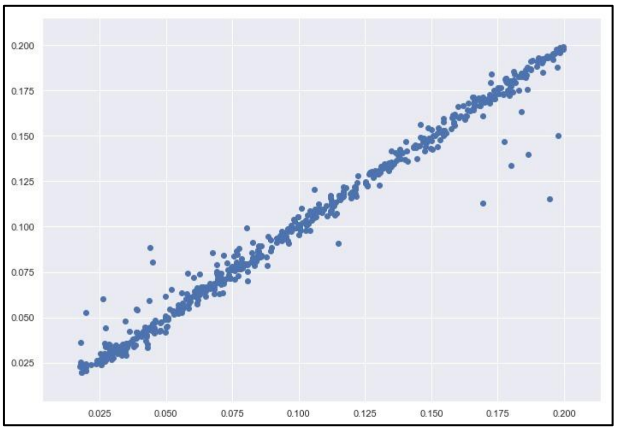
RMSE	0.007008615494561676
R2	0.9829762180344884

For testing set:

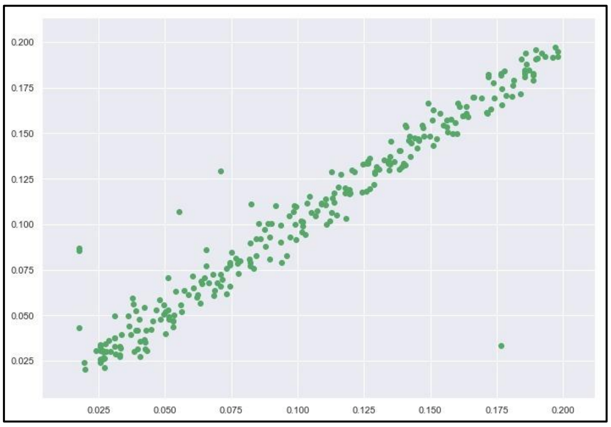
RMSE	0.01241493957816707
R2	0.9438373117599578

Plotting the actual values and predicted values:

For training set:



For testing set:



INFERENCE:

We can observe that the value of R2 score for the test dataset is 94% and for the training set is 98% in the given data and also calculated the RMSE value for the training set and the testing dataset have 0.0070 and 0.012 respectively.

6.3. Multiple Linear Regression with Principal Component Analysis

It is an unsupervised, non – parametric statistical technique primarily used for dimensional reduction of a given dataset in machine learning models.

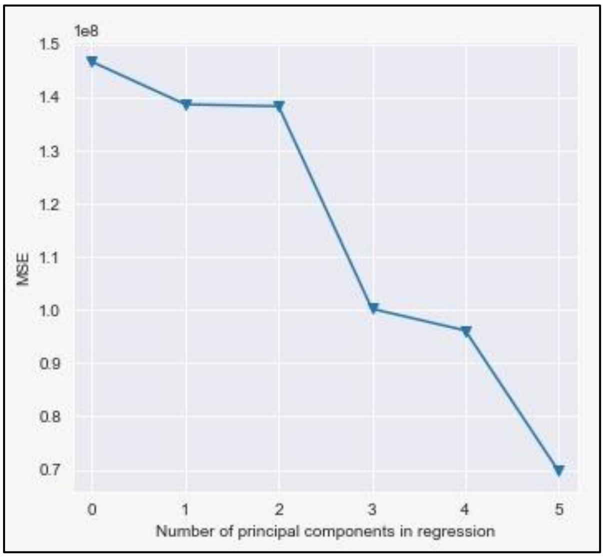
The curse of dimensionality is one of the most commonly occurring problems in ML. All the features that exist in each data set might make the training slow and inefficient. Hence, we go for the reduction of dimensions of the dataset. This kind of problems is technically also referred to as the curse of dimensionality.

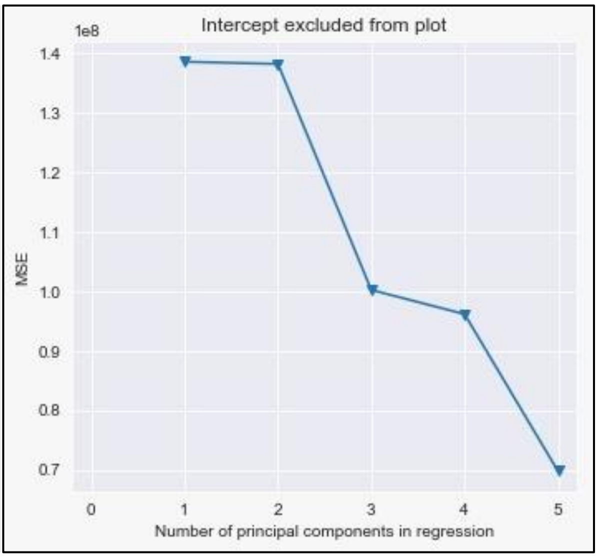
In PCA, the algorithm finds a low – dimensional representation of the data while retaining as much variance as possible. The main concept behind PCA is that, if few features in each data set are highly correlated to each other it combines them into one single feature thereby reducing dimension and computational time for training. These newly derived components are referred to as principal components

It can be used for:

- Noise filtering
- Visualisation
- Feature Extraction
- Stock market predictions
- Gene data analysis

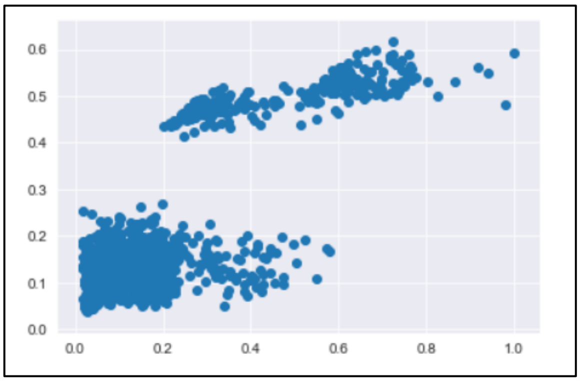
The primary problem associated with high dimensionality in the machine learning field in model overfitting reduces the ability to generalize beyond the example in the training set. It is essential to perform feature scaling before performing PCA if there is a significant difference between the features of the dataset



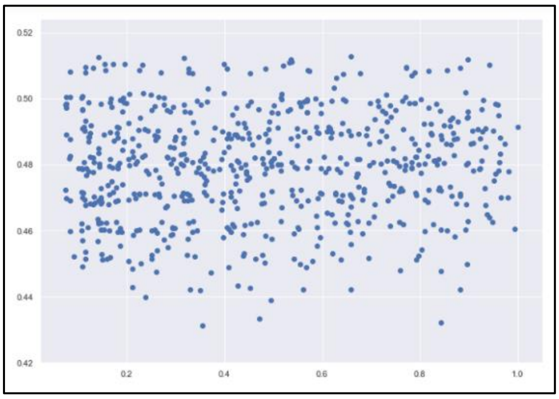


We can also observe that after fitting the data for a multiple regression the R2 score for the test set come out to be 66% which can be improved by adding data to the existing dataset as we observed removing outliners harmed this data set as shown below:

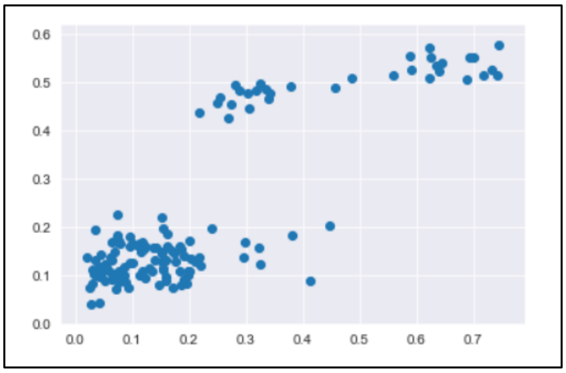
The model performance for testing set:



The model performance after removing the outlier:



RMSE is 0.104 and the R2 score is 70% which is decent as it is like the training set.



INFERENCE:

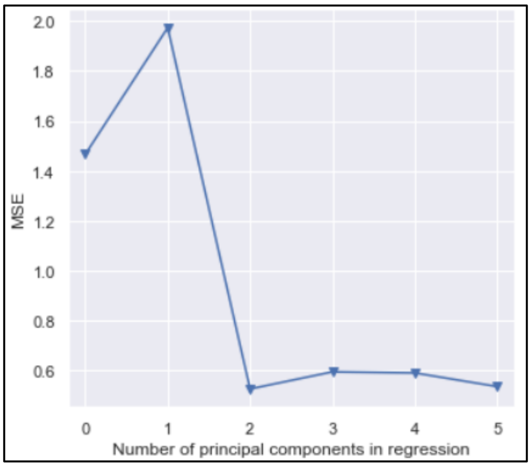
We can observe that the value of R2 score for the training set is 65% and test dataset is 70% for the in the given data and also calculated the RMSE value for the training set and the testing dataset have 0.65 and 0.104 respectively.

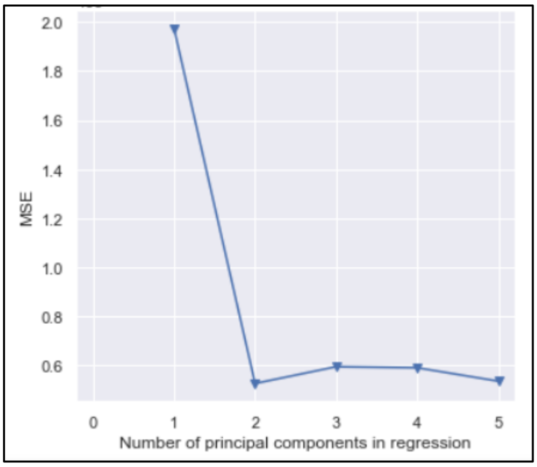
6.4. Random Forest Regression with Principal Component Analysis

PCA can make interpreting each “feature” a little harder when we analyse the “feature importance” of our Random Forest model. However, PCA performs dimensionality reduction, which can reduce the number of features for the Random Forest to process, so PCA might help speed up the training of your Random Forest model. Note that computational cost is one of the biggest drawbacks of Random Forests (it can take a long time to run the model). PCA can become really important especially when you are working with hundreds or even thousands of predicting features. So, if the most important thing is to simply have the best performing model, and interpreting feature importance can be sacrificed, then PCA may be useful to try.

A scree plot is a diagnostic tool to check whether PCA works well on your data or not. Principal components are created in order of the amount of variation they cover.

Using Eigen values, the following scree plot has been plotted:



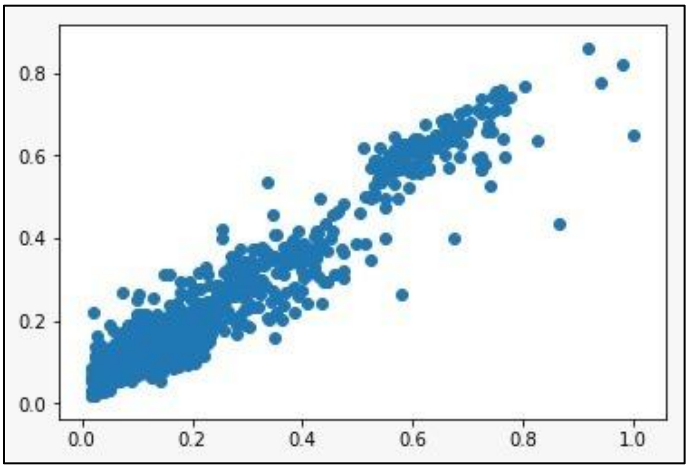


INFERENCE:

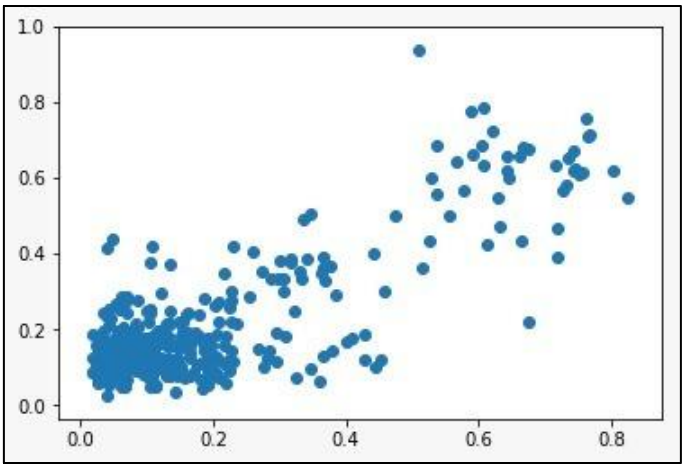
We can observe that the value of R2 score for the test dataset is 64% and for

the training set is 92% in the given data and also calculated the RMSE value for the training set and the testing dataset have 0.050 and 0.118 respectively.

For training set:



For testing set:



CONCLUSION:

For this dataset the computations that were taken place without PCA seemed promising as they had straight fit model outputs when a scatter plot was drawn between y train and y train predict.

- We could observe that the performance of the model was optimum when the data set was performed by following the method without PCA for both RFR and MLR.
- From the graphs plotted we can also conclude that RFR model has optimum fit for the data set when compared to MLR model.
- The RFR with PCA model seemed too overfit for the given data set hence the problem can be solved by either adding more data or by choosing a best model for the existing data