INSURANCE COST PREDICTION
BATCH_16_PROJECT GROUP_20

Submitted By:

SAURAV SINGH (REGNO:28)
MADIHA NOURIEN(REGNO:29)
PINNINTI BHARGAVI DAANA LAKSHMI(REGNO:26)
AISHWARYA K(REGNO:25)

Mentor :

MR. PRANAV JAIPURKAR
M.E(COMPUTER ENGINEERING)

MACHINE LEARNING AND ARTIFICIAL INTELLEGENCE PROJECT REPORT
ON
INSURANCE COST PREDICTION

BATCH_16_PROJECT GROUP_20

Submitted By:

SAURAV SINGH (REGNO:28)
MADIHA NAURIEN(REGNO:29)
PINNINTI BHARGAVI DAANA LAKSHMI(REGNO:26)
AISHWARYA K(REGNO:25)

Mentor :
MR. PRANAV JAIPURKAR
M.E(COMPUTER ENGINEERING)

ABSTRACT

The power of machine learning in understanding the patterns in data, analyzing and making decisions, has shown its importance in various sectors. Machine Learning requires reasonable amount of data to make accurate decisions. Data sharing and reliability of data is very crucial in machine learning in order to improve its accuracy.

Due to escalating healthcare costs, accurately predicting which patients will incur high costs is an important task for payers and providers of healthcare. High-cost claimants are patients who have annual costs above 250,000 and who represent just 0.16% of the insured population but currently account for 9% of all healthcare costs. In this study, we aimed to develop a high-performance algorithm to predict charges to inform a novel care management system

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of the people who made it possible, whose constant guidance and encouragement crowned our effort with success.

I express my sincere gratitude to our Mentor **MR. PRANAV JAIPURKAR M.E(COMPUTER ENGINEERING)**, Knowledge Solutions India for providing facilities.

TABLE OF CONTENT

LIST OF FIGURES OR GRAPH

CHAPTER 1

INTRODUCTION

Machine learning is a sub-domain of computer science which evolved from the study of pattern recognition in data, and also from the computational learning theory in artificial intelligence. It is the first-class ticket to most interesting careers in data analytics today. As data sources proliferate along with the computing power to process them, going straight to the data is one of the most straightforward ways to quickly gain insights and make predictions. Machine Learning can be thought of as the study of a list of sub-problems, viz: decision making, clustering, classification, forecasting, deep-learning, inductive logic programming, support vector machines, reinforcement learning, similarity and metric learning, genetic algorithms, sparse dictionary learning, etc. Supervised learning, or classification is the machine learning task of inferring a function from a labelled data. In Supervised learning, we have a training set, and a test set. The training and test set consists of a set of examples consisting of input and output vectors, and the goal of the supervised learning algorithm is to infer a function that maps the input vector to the output vector with minimal error. In an optimal scenario, a model trained on a set of examples will classify an unseen example in a correct fashion, which requires the model to generalize from the training set in a reasonable way. In layman's terms, supervised learning can be termed as the process of concept learning, where a brain is exposed to a set of inputs and result vectors and the brain learns the concept that relates said inputs to outputs. A wide array of supervised machine learning algorithms are available to the machine learning enthusiast, for example Neural Networks, Decision Trees, Support Vector Machines, Random Forest, Net, Majority Classifier 4 each have their own merits and demerits.

CHAPTER 2

SOFTWARE LIBRARIES

1.Numpy:

With NumPy, you can define arbitrary data types and easily integrate with most databases. NumPy can also serve as an efficient multi-dimensional container for any generic data that is in any datatype.

2.Pandas:

Pandas are turning up to be the most popular Python library that is used for data analysis with support for fast, flexible, and expressive data structures designed to work on both “relational” or “labelled” data.

Pandas today is an inevitable library for solving practical, real-world data analysis in Python. Pandas is highly stable, providing highly optimized performance.

3.Matplotlib:

Matplotlib is a data visualization library that is used for 2D plotting to produce publication-quality image plots and figures in a variety of formats.

The library helps to generate histograms, plots, error charts, scatter plots, bar charts with just a few lines of code.

4.scikit-learn:

Scikit-learn is a free machine learning library for Python. It features various algorithms like support vector machine, random forests, and k-neighbours, and it also supports Python numerical and scientific libraries like NumPy and SciPy .

5.Seaborn:

Seaborn is a data visualization library built on top of matplotlib and closely integrated with pandas data structures in Python.

Visualization is the central part of Seaborn which helps in exploration and understanding of data. One has to be familiar with Numpy and Matplotlib and Pandas to learn about Seaborn.

CHAPTER 3

Experimental Evaluation

Regression analysis is a quantitative research method which is used when the study involves modelling and analysing several variables, where the relationship includes a dependent variable and one or more independent variables.

Regression analysis is a quantitative method used to test the nature of relationships between a dependent variable and one or more independent variables. Regression methods form the backbone of much of the analyses in research. In general, these methods are used to estimate associations between variables, especially when one or more of these are variables are continuous.

1.1 Methodology

Step 1: Frame the Problem

In this project, we create machine learning models to predict the 'Insurance Cost' with minimum MSE, minimum RMSE and maximum R-Square Score. We build the following models:

1. Multiple linear regressor (MLR)
2. Random Forest Regressor (RFR)
3. MLR with PCA
4. RFR with PCA

Step 2: Collection of Raw Data

The Insurance dataset contains 7 features as shown below:

1. age: age of primary beneficiary
2. sex: insurance contractor gender (female, male)
3. bmi: Body Mass Index, providing an understanding of the 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
4. children: Number of children/ dependents covered by health insurance
5. smoker: Smoking status of the beneficiary
6. region: The beneficiary's residential area in the US (northeast, southeast, southwest, northwest)
7. charges: Individual medical costs billed by health insurance

Step 3: Process the Data for Analysis

- Now that you have all of the raw data, you'll need to process it before you can do any analysis.
- You need to process, explore, and condition data before modeling. The cleaner your data, the better are your predictions.
- Categorical variables are usually represented as 'strings' or 'categories' and are finite in number.
- Further, we can see there are two kinds of categorical data:

  i. Ordinal Data: The categories have an inherent order
  ii. Nominal Data: The categories do not have an inherent order

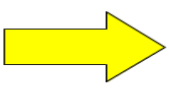
1. Label Encoding or Ordinal Encoding

   - We use this categorical data encoding technique when the categorical feature is ordinal.
   - In this case, retaining the order is important. Hence encoding should reflect the sequence.
   - In Label encoding, each label is converted into an integer value.

2. One Hot Encoding

   - We use this categorical data encoding technique when the features are nominal (do not have any order).
   - In one hot encoding, for each level of a categorical feature, we create a new variable.
   - Each category is mapped with a binary variable containing either 0 or 1. Here, 0 represents the absence, and 1 represents the presence of that category.

Color
Red
Red
Yellow
Green
Yellow



Red	Yellow	Green
1	0	0
1	0	0
0	1	0
0	0	1

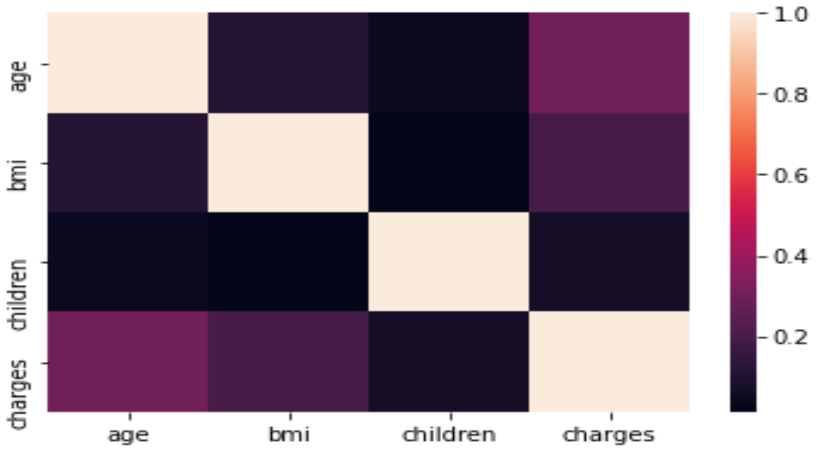
(3.1) OneHotEncoding

Step 4: Explore the Data

- Once we have a first version of our data, we need to perform exploratory data analysis (EDA)
- Exploratory Data Analysis refers to the critical process of performing initial investigations on data so as to discover patterns, to spot anomalies, to test hypothesis and to check assumptions with the help of summary statistics and graphical representations.
- At a high-level, the goal of EDA is to probe our data in as many ways as possible to gain an understanding for its characteristics.
- A few questions that can guide our exploration:

    1. What is the label distribution? Do we have a balanced/imbalanced dataset?
    2. What are the distributions of feature values?
    3. Are there any features that are malformed?

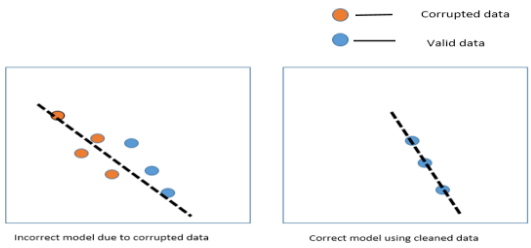
- Among other things, the goals of these questions are to derive insights about our data that can ultimately inform our modelling choices such as features or algorithms to use.
- In this stage, you need to determine the method and technique to draw the relation between input variables.
- Planning for a model is performed by using different statistical formulas and visualization tools.



(3.2) Correlation heat map for data preprocessing

Step 5: Data Cleaning

- Data cleaning means the process of identifying the incorrect, incomplete, inaccurate, irrelevant or missing part of the data and then modifying, replacing or deleting them according to the necessity.



(3.3) Data cleaning visualization

- At the gist of it all, Machine Learning is a **data**-driven AI. In machine learning, if the data is irrelevant or error-prone then it leads to an incorrect model building.
- Now let's take a closer look in the different ways of cleaning data:

1. Inconsistent column: If your Data Frame contains columns that are irrelevant or you are never going to use them then you can drop them to give more focus on the columns you will work on.
2. Missing data: When you start to work with real world data, you will find that most of the dataset contains missing values. Handling missing values is very important because if you leave the missing values as it is, it may affect your analysis and machine learning models. If you find any missing values in the dataset you can perform any of these three tasks on it:
    i. Leave as it is
    ii. Filling the missing values
    iii. Drop them
3. Duplicate rows: Datasets may contain duplicate entries. It is one of the easiest tasks to delete duplicate rows.
4. Outliers: In statistics, an outlier is a data point that differs significantly from other observations. For detecting the outliers, we can use:
    i. Box Plot
    ii. Scatter plot
    iii. Z-score etc.

In conclusion, data cleaning is very import for making your analytics and machine learning models error-free. A small error in the dataset can cause you a lot of problem. So, always try to make your data clean.

Step 6: Perform in-depth Analysis

- This step of the process is where you're going to have to apply your statistical, mathematical and technological knowledge and leverage all of the data science tools at your disposal to crunch the data and find every insight you can.
- In this step, the actual model building process starts.
- Here, Data scientist distributes datasets for training and testing.
- The model once prepared is tested against the "testing" dataset.

- **Training and Testing**

1. Splitting the data set into training and testing subsets helps to assess the performance of the model over an independent data set. Typically, we train the model using training data subset and then evaluate the model's performance using the testing data subset, which is independent of the training data subset.
2. Splitting the data set also helps in having a check on model's overfitting.

- **Assessing the Fit of Regression Models**

1. R-Square Score

- The coefficient of determination (R-squared) is a statistical metric that is used to measure how much of the variation in outcome can be explained by the variation in the independent variables.
- R2 always increases as more predictors are added to the MLR model, even though the predictors may not be related to the outcome variable.
- It indicates the goodness of fit of the model.
- R-squared has the useful property that its scale is intuitive: it ranges from zero to one, with zero indicating that the proposed model does not improve prediction over the mean model, and one indicating perfect prediction.
- Improvement in the regression model results in proportional increases in R-squared.

2. RMSE

- The RMSE is the square root of the variance of the residuals.

- It indicates the absolute fit of the model to the data, i.e., how close the observed data points are to the model's predicted values.
- RMSE is an absolute measure of fit.
- Lower values of RMSE indicate better fit.
- RMSE is a good measure of how accurately the model predicts the response, and it is the most important criterion for fit if the main purpose of the model is prediction.

3. MSE

- The mean squared error (MSE) tells you how close a regression line is to a set of points.
- It does this by taking the distances from the points to the regression line (these distances are the "errors") and squaring them.
- The squaring is necessary to remove any negative signs. It also gives more weight to larger differences.
- It's called the mean squared error as you're finding the average of a set of errors.
- The lower the MSE, the better the forecast.

Step 7: Exploring the results

- In statistics, the actual value is the value that is obtained by observation or by measuring the available data. It is also called the observed value. The predicted value is the value of the variable predicted based on the regression analysis.
- The difference between the actual value or observed value and the predicted value is called the residual in regression analysis.
- The difference between the actual and the predicted value is the residual which is defined as:
- $$e = y - \hat{y}$$

  Here, e is the residual, y is the observed or actual value and $\hat{y}$ is the predicted value.
  Each actual value has a predicted value and hence each data point has one residual.
- If the difference between the actual value and the predicted value is positive, then the data points are above the regression line.
- If the difference between the actual value and the predicted value is negative, then the data points are below the regression line.

- If the difference is zero, then that data points lie on the regression line. If the line of best fit is the best fit then the sum of the difference between the actual value and the predicted values is always zero.
- The residuals play a vital role to validate the obtained regression model.
- In this stage, you deliver the final baselined model with reports, code, and technical documents.
- Model is deployed into a real-time production environment after thorough testing.

CHAPTER :4

ALGORITHM

4.1. Multiple linear regressor (MLR)

Multiple linear regression is used to estimate the relationship between two or more independent variables and one dependent variable. You can use multiple linear regression when you want to know:
How strong the relationship is between two or more independent variables and one dependent variable (e.g. how rainfall, temperature, and amount of fertilizer added affect crop growth).
The value of the dependent variable at a certain value of the independent variables (e.g. the expected yield of a crop at certain levels of rainfall, temperature, and fertilizer addition).

- **Assumptions of multiple linear regression**

Multiple linear regression makes all of the same assumptions as simple linear regression:
Homogeneity of variance (homoscedasticity): the size of the error in our prediction doesn't change significantly across the values of the independent variable.
Independence of observations: the observations in the dataset were collected using statistically valid methods, and there are no hidden relationships among variables.
In multiple linear regression, it is possible that some of the independent variables are actually correlated with one another, so it is important to check these before developing the regression model. If two independent variables are too highly correlated (r2 > ~0.6), then only one of them should be used in the regression model.
Normality: The data follows a normal distribution.
Linearity: the line of best fit through the data points is a straight line, rather than a curve or some sort of grouping factor.
How to perform a multiple linear regression
Multiple linear regression formula

The formula for a multiple linear regression is:
Multiple linear regression formula

$$y = \beta_0 + \beta_1 X_1 + ... + \beta_n X_n + \varepsilon$$

(4.1) MLR Formula

y = the predicted value of the dependent variable
B0 = the y-intercept (value of y when all other parameters are set to 0)

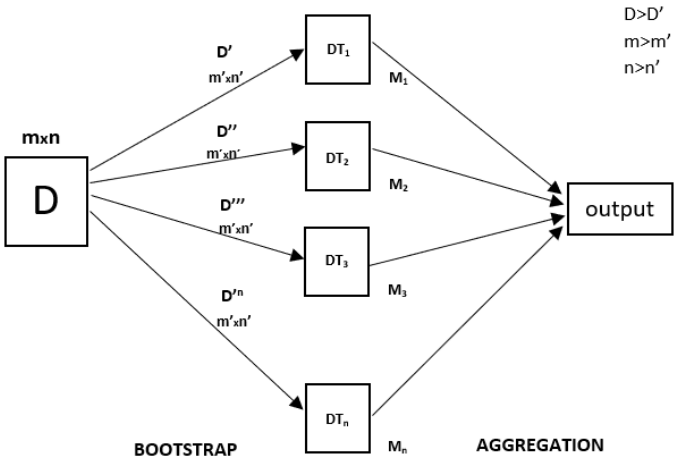
B1X1= the regression coefficient (B1) of the first independent variable (X1) (a.k.a. the effect that increasing the value of the independent variable has on the predicted y value)
… = do the same for however many independent variables you are testing
BnXn = the regression coefficient of the last independent variable
e = model error (a.k.a. how much variation there is in our estimate of y)
To find the best-fit line for each independent variable, multiple linear regression calculates three things:
The regression coefficients that lead to the smallest overall model error.
The t-statistic of the overall model. The associated p-value (how likely it is that the t-statistic would have occurred by chance if the null hypothesis of no relationship between the independent and dependent variables was true).
It then calculates the t-statistic and p-value for each regression coefficient in the model.

```
reg=linear_model.LinearRegression()
```

(4.2) Model selection

4.2. Random Forest Regressor (RFR):

Every decision tree has high variance, but when we combine all of them together in parallel then the resultant variance is low as each decision tree gets perfectly trained on that particular sample data and hence the output doesn't depend on one decision tree but multiple decision trees. In the case of a classification problem, the final output is taken by using the majority voting classifier. In the case of a regression problem, the final output is the mean of all the outputs. This part is Aggregation.



(4.3) Visualizing random forest algorithm

A Random Forest is an ensemble technique capable of performing both regression and classification tasks with the use of multiple decision trees and a technique called Bootstrap and Aggregation, commonly known as bagging. The basic idea behind this is to combine multiple decision trees in determining the final output rather than relying on individual decision trees.
Random Forest has multiple decision trees as base learning models. We randomly perform row sampling and feature sampling from the dataset forming sample datasets for every model. This part is called Bootstrap.

We need to approach the Random Forest regression technique like any other machine learning technique

- Design a specific question or data and get the source to determine the required data.
- Make sure the data is in an accessible format else convert it to the required format.
- Specify all noticeable anomalies and missing data points that may be required to achieve the required data.
- Create a machine learning model
- Set the baseline model that you want to achieve
- Train the data machine learning model.
- Provide an insight into the model with test data
- Now compare the performance metrics of both the test data and the predicted data from the model.

If it doesn't satisfy your expectations, you can try improving your model accordingly or dating your data or use another data modelling technique.
At this stage you interpret the data you have gained and report accordingly.

```
rfr = RandomForestRegressor(n_estimators = 49 , random_state = 0)
```

(4.4) Model selection

4.3.PCA

Principal component analysis helps make data easier to explore and visualize. It is a simple non-parametric technique for extracting information from complex and confusing data sets. Principal component analysis is focused on the maximum variance amount with the fewest number of principal components. One of the distinct advantages associated with the principal component analysis is that once patterns are found in the concerned data, compression of data is also supported. One makes use of principal component analysis to eliminate the number of variables or when there are too many predictors compared to number of observations or to avoid multicollinearity. It is closely related to canonical correlational analysis and makes use of orthogonal transformation in order to convert the set of observations containing correlated variables into a set of values known as principal components. The number of principal components used in principal component analysis is less than or equal to the lesser number of observations. Principal component analysis is sensitive to the relative scaling of the originally used variables.

Principal component analysis is widely used in many areas such as market research, social sciences and in industries where large data sets are used. The technique can also help in providing a lower-dimensional picture of the original data. Only minimal effort is needed in the case of principal component analysis for reducing a complex and confusing data set into a simplified useful information set.

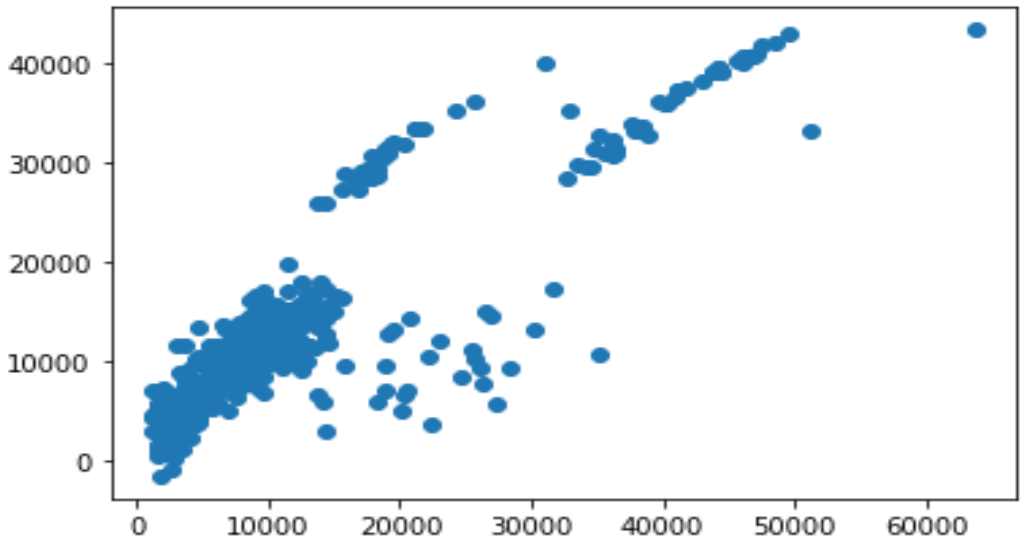
```
from sklearn.decomposition import PCA
import matplotlib.pyplot as plt
from sklearn.metrics import mean_squared_error, r2_score
import math

scaled_data = sc.fit_transform(df1)
pca = PCA(n_components=5)
pca.fit(scaled_data)
x_pca = pca.transform(scaled_data)
ev = pca.explained_variance_ratio_
```

(4.5) Model selection of PCA

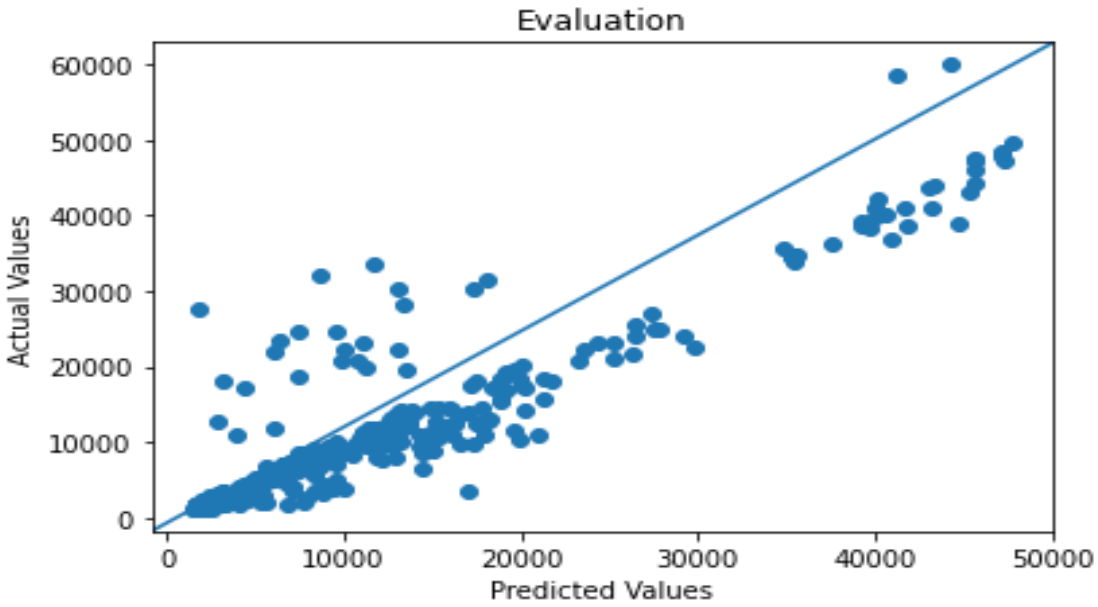
CHAPTER :5

GRAPHS AND RESULT



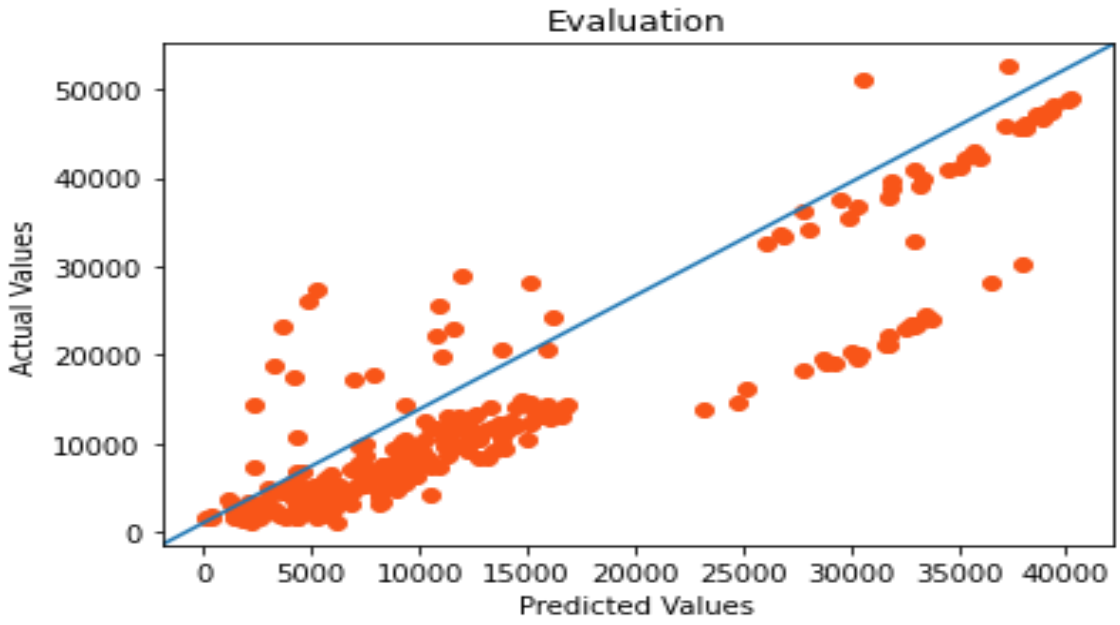
(5.1) Plotting Cross-Validated Predictions on MLR

1. MAE: 4069.0115869461947
2. MSE: 33764390.33773838
3. RMSE: 5810.713410394491
4. TRAINING SET PREDICTION SCORE : 0.7451222029094489
5. TESTING SET PREDICTION SCORE : 0.7596450053533924
6. ACTUAL CHARGES : 5425.02
7. MULTILINEAR PREDICTION : 8238.79



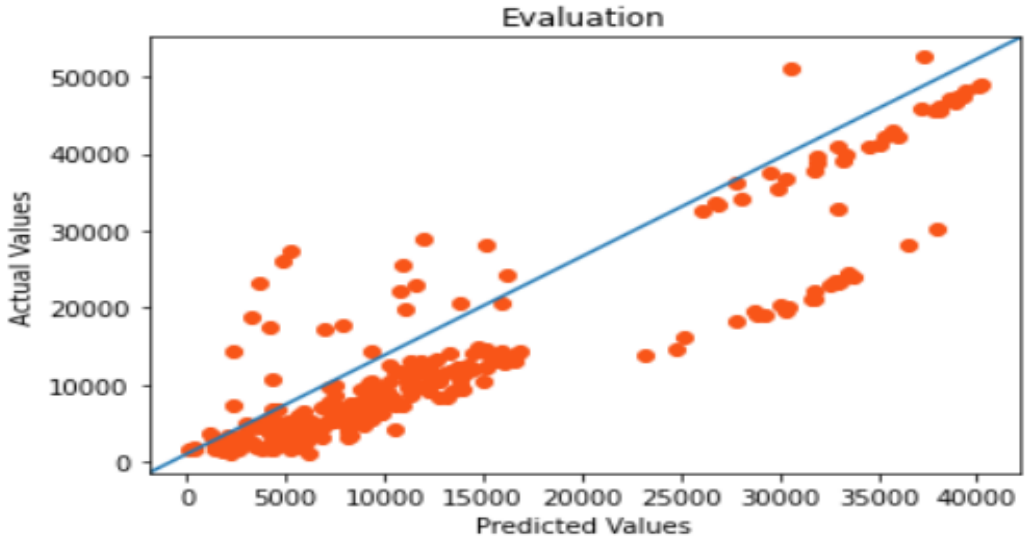
(5.2) Plotting Cross-Validated Predictions on RFR

1. RMSE-Random Forest : 4777.75 (square-rooted)
2. R2-Random Forest : 0.83
3. MSE-Random Forest : 22826857.22
4. TRAINING SET PREDICTION SCORE: 0.9757867485085552
5. TESTING SET PREDICTION SCORE : 0.8303030611618449
6. ACTUAL CHARGES : 11538.42
7. RANDOM FOREST PREDICTION : 12731.50



(5.3) Plotting Cross-Validated Predictions on RFR with PCA

1. RMSE-RANDOM FOREST : 5009.92 (square-rooted)
2. R2-RANDOM FOREST : 0.84
3. MSE-RANDOM FOREST : 25099313.09
4. TRAINING PREDICTION SCORE : 0.9662946235047881
5. TESTING PREDICTION SCORE : 0.8405729935224402



(5.4) Plotting Cross-Validated Predictions on MLR with PCA

1. RMSE-MULTILINEAR REGRESSION : 5650.46 (square-rooted)
2. R2-MULTILINEAR REGRESSION : 0.80
3. MSE-MULTILINEAR REGRESSION : 31927677.11
4. TRAINING PREDICTION SCORE : 0.7366969327839583
5. TESTING PREDICTION SCORE : 0.7993609962127467

CHAPTER 6

CONCLUSION

In this study, we preformed two models multilinear the results of conventional multiple linear regression (MLR) were compared with those of random forest regression (RFR), in the prediction
In general, RFR seemed to be superior to the MLR in terms of predictive value and error.

In the case of this data set, RFR appeared to be superior to MLR in terms of its explanatory value and error. This result suggests that RFR with PCA may have advantages over MLR with PCA for prediction of charges with this kind of data set, but that MLR can still have good predictive value in some cases.

s

CHAPTER 7

BIBLIOGRAPHY

1. https://www.mentorrbuddy.com/student/LearningDashBoards?sid=1188
2. https://youtu.be/FLkOX4Eez6o
3. https://www.w3schools.com/python/python_ml_getting_started.asp
4. https://www.geeksforgeeks.org/machine-learning/