**Case Study**

**Boston Housing**

**Summitted by:**

**Ajay**

**Case Study Summary:**

The following case study of Boston Housing reads the historical data set from BostonHousing.csv, clean it, and then predict the median value of the House in Boston for new records using Multiple Regression Model. First it uses 13 predictors from the datasets for the target output. And then, it optimizes the accuracy of prediction of price by iterating number of predictors in the model using Exhaustive search method and forward selection of predictors in the dataset.

**Main Chapters:**

1. Upload, explore, clean, and preprocess data for multiple linear regression.
2. Develop multiple linear regression with all 13 predictors.
3. Develop multiple linear regression with reduced number of predictors.

**1A. Lets Determine and present in this report the data frame dimensions, i.e., number of rows and columns.**

Solution:

|  |  |
| --- | --- |
| Number of Rows | Number of Columns |
| 506 | 14 |

.

**1B. Display in Python the column titles. If some of them contain two (or more) words, convert them into one-word titles, and present the modified titles in your report.**

Solution:

Below is the screenshot of modified column names from python code file.

Text, letter

Description automatically generated

**1C. Display in Python column data types. If some of them are listed as “object’, convert them into dummy variables, and provide in your report the modified list of column titles with dummy variables.**

Solution:

Below is the screenshot of modified column’s data type (category) from python code file.

A screenshot of a computer

Description automatically generated with medium confidence

**1D. Display in Python the descriptive statistics for all columns in the modified boston\_df data frame (after converting to one-word titles and dummy variables). Check if there are missing records (values) in the columns. Present the table with descriptive statistics in your report, and comment about the missing values. You don’t need to comment on the values of outliers (min/max) or their extreme values.**

Solution:

Below is the screenshot of Descriptive statistics of the boston\_df.

**Table

Description automatically generated**

We know that in case of N missing values in a column, count for the same column will be reduced by N. In boston\_df data frame, count for each column is equal to the total number of rows of the data frame. This helps us to conclude that there is no missing value in any of the column in the entire data frame.

**2A. Develop in Python outcome and predictor variables, partition the data set (60% for training and 40% for validation partitions), and train the multiple linear regression model using Linear Regression with the training data set. Identify and display in Python intercept and regression coefficients of this model. Provide these coefficients in your**

**report and present the mathematical equation of this linear regression model.**

Solution:

Below is the screenshot of intercept and coefficient of all 13 predictors.

**Table

Description automatically generated**

Equation of Regression line: y = B0 + B1 + B2……+ Bn

MVALUE = 43.65 - 0.14(CRIME) + 0.01(ZONE) + 0.12(INDUST) - 16.47(NIT\_OXIDE) + 0.89(ROOMS) - 0.01(AGE) - 0.72(DISTANCE) + 0.20(RADIAL) - 0.01(TAX) - 0.58(ST\_RATIO) - 0.45(LOW\_STAT) + 2.11(CHAR\_RIV\_Y) + 10.99(C\_MVALUE\_Yes)

**2B. Using the multiple regression model, identify in Python predictions for validation and training predictors (valid\_X and train\_X). Based on these predictions, identify and display in Python R2 and adjusted R2 performance measures for training and validation partitions. Present and compare these performance measures in your report and explain if there is a possibility of overfitting.**

Solution:

Below is the screenshot of R2 and adjusted R2 for the training and validation dataset.

**A screenshot of a computer

Description automatically generated with medium confidence**

1. adjusted R2 for training Model is 0.832, and for validation set is 0.822. Hence the measure of fit is better for training dataset for our Model.
2. We know that if R2 or adjusted R2 Varies between training and validation datasets by significant margin, then it must be considered for the possibility of overfitting of Model. But in our training and valid datasets, R2 or adjusted R2 are close (almost similar). Hence, we can conclude that there is no overfitting

**2C. Identify and display in Python the common accuracy measures for training and validation data set (predictions). Provide and compare these accuracy measures in your report and assess again a possibility of overfitting.**

Solution:

**Table

Description automatically generated**

1. It can be observed that from above output, both RMSE and MAPE are slightly lower for training set, Therefore, based on common accuracy measure Training set output are better.
2. RMSE for training and valid sets are 3.58 and 3.87 resp. and MAPE are 13.17 and 13.32 resp. This shows that there is no significant variation between the common accuracy measures of training and validation data frame. Hence there is no overfitting.

**3A. Use the Exhaustive Search algorithm in Python to identify the best predictors for the multiple linear regression model. Based on these predictors, train a new multiple linear regression model using the respective training data set predictors. Identify and display in Python the intercept and regression coefficients of this model and the common accuracy measures for validation partition. Provide these coefficients in your report and present the mathematical equation of the respective multiple linear regression model.**

Solution**:** Below is the screenshot of intercept and coefficients of 11 best predictors resulted from Exhaustive search Method

**Table

Description automatically generated**

Equation of Regression line:

MVALUE = 43.89 - 0.14(CRIME) + 0.11(INDUST) - 16.89(NIT\_OXIDE) + 0.86(ROOMS) - 0.63(DISTANCE) + 0.19(RADIAL) - 0.01(TAX) - 0.61(ST\_RATIO) - 0.46(LOW\_STAT) + 2.13(CHAR\_RIV\_Y) + 11.11(C\_MVALUE\_Yes)

Text

Description automatically generated with medium confidence

**3B. Use the Forward Selection algorithm in Python exactly as discussed in 3a. Provide the same results in your report as discussed in 3a. Also, explain the differences between the best predictors (number and specific predictors used) in the models in 3a and 3b.**

Solution**:** Below is the screenshot of intercept and coefficients of 9 predictors resulted from Forward selection method

**Table

Description automatically generated**

Equation of Regression line:

MVALUE = 42.76 - 0.14(CRIME) – 15.95(NIT\_OXIDE) + 0.87(ROOMS) - 0.71 (DISTANCE) + 0.11 (RADIAL) - 0.60(ST\_RATIO) - 0.45(LOW\_STAT) + 2.36(CHAR\_RIV\_Y) + 10.97(C\_MVALUE\_Yes)

**Text

Description automatically generated with medium confidence**

**3C. Present and compare in your report the common accuracy measures for validation data set of the three linear regression models: with all predictors, based on the Exhaustive Search algorithm, and based on Forward Selection algorithm. Using the value of RMSE and the number of variables in each model, which model would you recommend using for making predictions in this case? Briefly explain your answer.**

Solution: Below is the screenshot of Accuracy Measures of 3 Models.

Graphical user interface, text, application

Description automatically generated

1. We can see from above table, Common Accuracy Measure RMSE is lowest for 11 Predictors resulted from Exhaustive search Method. Hence, I would pick Model made of 11 predictors selected from Exhaustive search method.
2. Also, we can see that Second best model is using all 13 Predictors which would lead to more computational cost. Hence it is another reason to keep pick second model (Exhaustive search) over the first Model.