BAN 620 Case Study 1

Boston Housing

Summitted by:
Ajay Kumar (Net ID: sb6667)

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. Create a boston_df data frame by uploading the original data set into Python. 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.

```
Modified column titles with no space and one word for titles:

Index(['CRIME', 'ZONE', 'INDUST', 'CHAR_RIV', 'NIT_OXIDE', 'ROOMS', 'AGE',
       'DISTANCE', 'RADIAL', 'TAX', 'ST_RATIO', 'LOW_STAT', 'MVALUE',
       'C_MVALUE'],
      dtype='object')
```

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.

```
Column with Object data type in the dataset are:
CHAR_RIV
C_MVALUE

Category levels and changed variable type of CHAR_RIV:
Index(['N', 'Y'], dtype='object')
category

Category levels and changed variable type of C_MVALUE:
Index(['No', 'Yes'], dtype='object')
category
```

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.

Descriptive Statistics	CRIME	ZONE	INDUST	NIT_OXIDE	ROOMS	AGE	DISTANCE	RADIAL	TAX	ST_RATIO	LOW_STAT	MVALUE	CHAR_RIV_Y	C_MVALUE_Yes
count	506.00	506.00	506.00	506.00	506.00	506.00	506.00	506.00	506.00	506.00	506.00	506.00	506.00	506.00
mean	3.61	11.36	11.14	0.55	6.28	68.57	3.80	9.55	408.24	18.46	12.65	22.53	0.07	0.17
std	8.60	23.32	6.86	0.12	0.70	28.15	2.11	8.71	168.54	2.16	7.14	9.20	0.25	0.37
min	0.01	0.00	0.46	0.39	3.56	2.90	1.13	1.00	187.00	12.60	1.73	5.00	0.00	0.00
25%	0.08	0.00	5.19	0.45	5.89	45.03	2.10	4.00	279.00	17.40	6.95	17.03	0.00	0.00
50%	0.26	0.00	9.69	0.54	6.21	77.50	3.21	5.00	330.00	19.05	11.36	21.20	0.00	0.00
75%	3.68	12.50	18.10	0.62	6.62	94.08	5.19	24.00	666.00	20.20	16.96	25.00	0.00	0.00
max	88.98	100.00	27.74	0.87	8.78	100.00	12.13	24.00	711.00	22.00	37.97	50.00	1.00	1.00

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.

```
Regression Model for BostonHousing Training Set:
Intercept:  43.65
       Predictor  Coefficient
0          CRIME        -0.14
1           ZONE         0.01
2         INDUST         0.12
3      NIT_OXIDE       -16.47
4          ROOMS         0.89
5            AGE        -0.01
6       DISTANCE        -0.72
7         RADIAL         0.20
8            TAX        -0.01
9       ST_RATIO        -0.58
10      LOW_STAT        -0.45
11    CHAR_RIV_Y         2.11
12  C_MVALUE_Yes        10.99
```

Equation of Regression line: $y = B_0 + B_1 + B_2 ...... + B_n$

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 R_2 and adjusted R_2 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 R_2 and adjusted R_2 for the training and validation dataset.

```
Prediction Performance Measures for Training Set
r2 :  0.839
Adjusted r2 :  0.832
AIC :  1663.52
BIC :  1719.22

Prediction Performance Measures for Validation Set
r2 :  0.834
adjusted r2 :  0.822
AIC :  1156.17
BIC :  1205.87
```

1. adjusted R_2 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 R_2 or adjusted R_2 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, R_2 or adjusted R_2 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:

```
Accuracy Measures for Training Set - All Variables

Regression statistics

                      Mean Error (ME) : 0.0000
       Root Mean Squared Error (RMSE) : 3.5845
            Mean Absolute Error (MAE) : 2.5961
          Mean Percentage Error (MPE) : -2.7127
Mean Absolute Percentage Error (MAPE) : 13.1715

Accuracy Measures for Validation Set - All Variables

Regression statistics

                      Mean Error (ME) : 0.4347
       Root Mean Squared Error (RMSE) : 3.8763
            Mean Absolute Error (MAE) : 2.7696
          Mean Percentage Error (MPE) : -2.2773
Mean Absolute Percentage Error (MAPE) : 13.3233
```

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

```
Regression Model for Training Set Using Exhaustive Search

Intercept  43.89
      Predictor  Coefficient
0         CRIME        -0.14
1        INDUST         0.11
2     NIT_OXIDE       -16.89
3         ROOMS         0.86
4      DISTANCE        -0.63
5        RADIAL         0.19
6           TAX        -0.01
7      ST_RATIO        -0.61
8      LOW_STAT        -0.46
9    CHAR_RIV_Y         2.13
10  C_MVALUE_Yes       11.11
```

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)

```
Accuracy Measures for Validation Set Using Exhaustive Search

Regression statistics

                      Mean Error (ME) : 0.4505
       Root Mean Squared Error (RMSE) : 3.8674
            Mean Absolute Error (MAE) : 2.7724
          Mean Percentage Error (MPE) : -2.1963
Mean Absolute Percentage Error (MAPE) : 13.3441
```

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

```
Regression Model for Training Set Using Forward Selection

Intercept  42.76
     Predictor  Coefficient
0        CRIME        -0.14
1    NIT_OXIDE       -15.95
2        ROOMS         0.87
3     DISTANCE        -0.71
4       RADIAL         0.11
5     ST_RATIO        -0.60
6     LOW_STAT        -0.45
7   CHAR_RIV_Y         2.36
8  C_MVALUE_Yes        10.97
```

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)

```
Accuracy Measures for Validation Set Using Forward Selection

Regression statistics

                      Mean Error (ME) : 0.4321
       Root Mean Squared Error (RMSE) : 3.9314
            Mean Absolute Error (MAE) : 2.8585
          Mean Percentage Error (MPE) : -2.3792
Mean Absolute Percentage Error (MAPE) : 13.8040
```

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.

All Predictors	11 Predictors from Exhaustive Search	9 Predictors from Foprward Selection
Mean Error (ME) : 0.4347 Root Mean Squared Error (RMSE) : 3.8763 Mean Absolute Error (MAE) : 2.7696 Mean Percentage Error (MPE) : -2.2773 Mean Absolute Percentage Error (MAPE) : 13.3233	Mean Error (ME) : 0.4505 Root Mean Squared Error (RMSE) : 3.8674 Mean Absolute Error (MAE) : 2.7724 Mean Percentage Error (MPE) : -2.1963 Mean Absolute Percentage Error (MAPE) : 13.3441	Mean Error (ME) : 0.4321 Root Mean Squared Error (RMSE) : 3.9314 Mean Absolute Error (MAE) : 2.8585 Mean Percentage Error (MPE) : -2.3792 Mean Absolute Percentage Error (MAPE) : 13.8040

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.