**CASE1**

**1.**

**a.** **Data frame Dimensions**

Here, the original dataset is loaded into a data frame named boston\_df and number of rows is obtained using the shape attribute, indicating the number of records in the dataset and number of columns is also obtained using the shape attribute, indicating the number of variables or features in the dataset.



A screenshot of a computer code

Description automatically generated

This indicates that there are 506 records and 14 variables in the dataset.

**b.** **Column Titles**

The column titles are displayed in Python using boston\_df.columns.

A close up of a text

Description automatically generated

Here, Converting the two-word titles into one-word titles.

A screenshot of a computer code

Description automatically generated

**c.** **Column Data Types and Dummy Variables**

Here, Column data types are identified and displayed using boston\_df.dtypes.

A screenshot of a computer

Description automatically generated

And columns which are of type 'object', they are converted into dummy variables.

Here we have two object data types i.e.., CHAR\_RIV and C\_MVALUE.

The modified data frame with dummy variables is displayed.

**A screenshot of a computer screen

Description automatically generated**

This indicates the conversion of categorical variables 'CHAR\_RIV' and 'C\_MVALUE' into dummy variables 'CHAR\_RIV\_N', 'CHAR\_RIV\_Y', 'C\_MVALUE\_No', and 'C\_MVALUE\_Yes'.

**d. Descriptive Statistics and Missing Values**

Descriptive statistics for all columns are displayed using boston\_df.describe().

Missing values in each column are checked using boston\_df.isnull().sum().

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Here in the output the count of rows is shown 506, which is equal to no of rows 506, we can conclude that they are no missing values in our data frame.

**2. Develop multiple linear regression with all 13 predictors.**

**a. Here, we created a Multiple Linear Regression model with Training Partition of 70% and Validation Partition being around 30%.**

A screenshot of a computer code

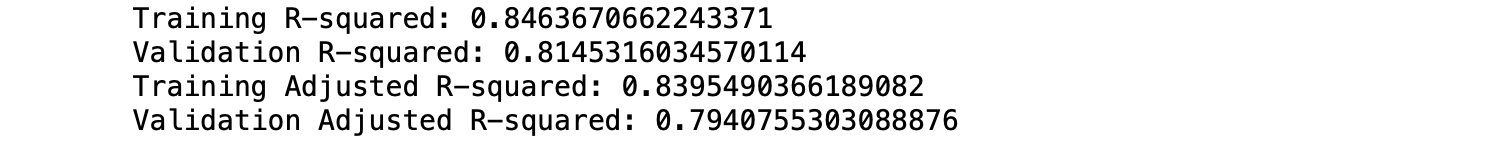
Description automatically generated

Linear Regression Equation is also displayed in the output.

A black text with numbers

Description automatically generated with medium confidence

**b. Below are the performance measures for training and validation data set.**



A white background with black text

Description automatically generated

The R-squared and adjusted R-squared values are high for both the training and validation datasets. In the training set, the R-squared value is 0.846 and the adjusted R-squared is 0.841, indicating strong performance. Similarly, in the validation set, the R-squared value is approximately 0.815 with an adjusted R-squared of 0.799, which is also high. The similarity between the validation and training set scores suggests there is no overfitting issue.

Additionally, the AIC and BIC values for the validation set, which are 423.17 and 459.45 respectively, are lower than those for the training set. This indicates that the model performs well in prediction tasks.

**c. Below is the Accuracy Measures for Training and validation Data Set.**

A screenshot of a computer program

Description automatically generated

So, here that the MAPE and RMSE values for the validation set are 13.3911 and 3.7175, respectively, which are very similar to the MAPE and RMSE values of the training set (13.3332 and 3.6748), it can be inferred that there is no risk of overfitting. The close resemblance between the validation and training set metrics suggests that there is not much difference and concluding that there is no overfitting.

**3. Develop multiple linear regression with reduced number of predictors.**

**a. Exhaustive search Algorithm.**

A table of numbers and letters

Description automatically generated with medium confidence

Here, Creating the Linear Regression Model with best predictors identified by Exhaustive search Method. The below is the output and model equation is also mentioned in the output.

A computer screen shot of a computer

Description automatically generated

Accuracy Measures and Predictions for the Predictors with Exhaustive search.

A screenshot of a computer error

Description automatically generated

**b. Backward Elimination algorithm**

Identifying of the best predictors.

A white background with black text

Description automatically generated

In above the Start score is 1962.03 and the variable of Room and ZONE are removed.

These are the best variables CRIME, INDUST, NIT\_OXIDE, DISTANCE, RADIAL, TAX, ST\_RATIO, LOW\_STAT, CHAR\_RIV\_N, C\_MVALUE\_No.

Here, Creating the Linear Regression Model with best predictors identified by Backward Elimination Method. The below is the output and model equation is also mentioned in the output.

A computer screen shot of a computer

Description automatically generated

Accuracy Measures and Predictions for the Predictors with Backward Elimination

A screenshot of a computer

Description automatically generated

**c. Comparison of all models**

A screenshot of a computer error

Description automatically generated

From the above, it is evident that the linear regression's RMSE using the best predictors derived from backward elimination and exhaustive search is 3.7318 and 03.6855, respectively. The RMSE of the linear regression, which considers every variable used in the prediction, is 03.7175.

The RMSE of exhaustive search is lower than that of RMSE value of Linear Regression which includes all variables and Backward Elimination.

Even though there isn't much of a difference between the values, an exhaustive search is still better than remaining two.

Whereas the Linear Regression based on Backward Elimination and Exhaustive search selected 11 predictor variables, the Linear Regression Model included all 13 predictor variables for the prediction.