Case Study 1 - Boston Housing

Predict median house price in new areas

Adriel Naranjo
ed2149

Data Mining - BAN 620
Instructor: Dr. Zinovy Radovilsky

1. *Upload, explore, clean, and preprocess data for multiple linear regression.*
   a. In the Boston Housing dataset there are 506 rows and 14 columns. Each of the rows represents a record of a certain home's attributes or features which is expressed as a column. The variable names of each column or attribute in this in the dataset are as follows:
      i. CRIME
      ii. ZONE
      iii. INDUST
      iv. CHAR_RIV
      v. NIT_OXIDE
      vi. ROOMS
      vii. AGE
      viii. DISTANCE
      ix. RADIAL
      x. TAX
      xi. ST_RATIO
      xii. LOW_STAT
      xiii. MVALUE
      xiv. C_MVALUE
   b. The variable name of each columns and their corresponding data types are as follows:
      i. CRIME float64
      ii. ZONE float64
      iii. INDUST float64
      iv. CHAR_RIV object ← need to convert and create dummy variable
      v. NIT_OXIDE float64
      vi. ROOMS float64
      vii. AGE float64
      viii. DISTANCE float64
      ix. RADIAL int64
      x. TAX int64
      xi. ST_RATIO float64
      xii. LOW_STAT float64
      xiii. MVALUE float64
      xiv. C_MVALUE object ← need to convert and create dummy variable

It can be seen that most of the variables in this dataset are decimal (floating point) numerical values but there are two variables that possess the 'object' type. **The two fields that have 'object' as their type are CHAR_RIV and C_MVALUE.** After creating dummy variables for these two variables, the dataset is transformed and has column names as follows:

CRIME ZONE INDUST NIT_OXIDE ROOMS AGE DISTANCE RADIAL TAX ST_RATIO LOW_STAT MVALUE **CHAR_RIV_Y** **C_MVALUE_Yes**

c. Below is a table displaying the descriptive statistics for the Boston Housing dataset.

	CRIME	ZONE	INDUST	NIT_OXIDE	ROOMS	AGE	DISTANCE	RADIAL	TAX	ST_RATIO	LOW_STAT	MVALUE	CHAR_RIV_Y	C_MVALUE_Yes
count	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000
mean	3.613524	11.363636	11.136779	0.554695	6.284634	68.574901	3.795043	9.549407	408.237154	18.455534	12.653063	22.532806	0.069170	0.166008
std	8.601545	23.322453	6.860353	0.115878	0.702617	28.148861	2.105710	8.707259	168.537116	2.164946	7.141062	9.197104	0.253994	0.372456
min	0.006320	0.000000	0.460000	0.385000	3.561000	2.900000	1.129600	1.000000	187.000000	12.600000	1.730000	5.000000	0.000000	0.000000
25%	0.082045	0.000000	5.190000	0.449000	5.885500	45.025000	2.100175	4.000000	279.000000	17.400000	6.950000	17.025000	0.000000	0.000000
50%	0.256510	0.000000	9.690000	0.538000	6.208500	77.500000	3.207450	5.000000	330.000000	19.050000	11.360000	21.200000	0.000000	0.000000
75%	3.677083	12.500000	18.100000	0.624000	6.623500	94.075000	5.188425	24.000000	666.000000	20.200000	16.955000	25.000000	0.000000	0.000000
max	88.976200	100.000000	27.740000	0.871000	8.780000	100.000000	12.126500	24.000000	711.000000	22.000000	37.970000	50.000000	1.000000	1.000000

A check for missing values was performed and we can see that there are **no missing values**.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 506 entries, 0 to 505
Data columns (total 14 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   CRIME         506 non-null    float64
 1   ZONE          506 non-null    float64
 2   INDUST        506 non-null    float64
 3   NIT_OXIDE     506 non-null    float64
 4   ROOMS         506 non-null    float64
 5   AGE           506 non-null    float64
 6   DISTANCE      506 non-null    float64
 7   RADIAL        506 non-null    int64
 8   TAX           506 non-null    int64
 9   ST_RATIO      506 non-null    float64
 10  LOW_STAT      506 non-null    float64
 11  MVALUE        506 non-null    float64
 12  CHAR_RIV_Y    506 non-null    uint8
 13  C_MVALUE_Yes  506 non-null    uint8
dtypes: float64(10), int64(2), uint8(2)
memory usage: 48.6 KB
```

```
CRIME           0
ZONE            0
INDUST          0
NIT_OXIDE       0
ROOMS           0
AGE             0
DISTANCE        0
RADIAL          0
TAX             0
ST_RATIO        0
LOW_STAT        0
MVALUE          0
CHAR_RIV_Y      0
C_MVALUE_Yes    0
dtype: int64
```

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

   a. To the right we can see the coefficients that were calculated when fitting the Boston Housing training data to a Linear Regression model.

Mathematical equation of this linear regression model is as follows:
MVALUE = 48.62 + (-0.15)CRIME +

(-0.01)ZONE + (0.13)INDUST +

(-17.86)NIT_OXIDE + (0.33)ROOMS +

(-0.01)AGE + (-0.66)DISTANCE +

(0.22)RADIAL + (-0.01)TAX +

(-0.63)ST_RATIO + (-0.47)LOW_STAT +

(2.33)CHAR_RIV_Y + (12.13)C_MVALUE_Yes

```
Regression Model for Boston Housing Training Set

Intercept:  48.62
       Predictor  Coefficient
0          CRIME        -0.15
1           ZONE        -0.01
2         INDUST         0.13
3      NIT_OXIDE       -17.86
4          ROOMS         0.33
5            AGE        -0.01
6       DISTANCE        -0.66
7         RADIAL         0.22
8            TAX        -0.01
9       ST_RATIO        -0.63
10      LOW_STAT        -0.47
11    CHAR_RIV_Y         2.33
12  C_MVALUE_Yes        12.13
```

   b. Based on the models predictions the R2 and adjusted R2 performance measures for training and validation partitions can be found below.

```
Prediction Performance Measures for Training Set
r2 :  0.83
Adjusted r2 :  0.824
Prediction Performance Measures for Validation Set
r2 :  0.852
adjusted r2 :  0.838
```

Conclusion: While the R2 for the validation set is higher than that of the training set, the adjusted R2 values are relatively close. This suggests that **the model is not showing strong signs of overfitting** based solely on these metrics.

   c. The common accuracy measures for training and validation data set (predictions) can be found below.
      i. Mean Error (ME) close to zero indicates that, on average, the model's predictions are unbiased.

ii. RMSE measures the average magnitude of the errors, with lower values indicating better model performance. The RMSE values for both sets are close, suggesting consistent performance.
iii. MAE represents the average absolute difference between predicted and actual values, with lower values indicating better accuracy. Again, the MAE values for both sets are similar.
iv. MPE and MAPE provide insights into the percentage errors. The negative values of MPE indicate that, on average, the model tends to slightly underestimate the target variable. MAPE values are also comparable between the sets.

Conclusion: Based on these accuracy statistics, **there doesn't appear to be a significant indication of overfitting**. The model's performance on the validation set is consistent with its performance on the training set, as evidenced by the similar RMSE, MAE, MPE, and MAPE values.

```
Accuracy Measures for Training Set - All Variables

Regression statistics

                      Mean Error (ME) : 0.0000
       Root Mean Squared Error (RMSE) : 3.7145
            Mean Absolute Error (MAE) : 2.6931
          Mean Percentage Error (MPE) : -2.7567
Mean Absolute Percentage Error (MAPE) : 13.2197

Accuracy Measures for Validation Set - All Variables

Regression statistics

                      Mean Error (ME) : 0.3667
       Root Mean Squared Error (RMSE) : 3.6868
            Mean Absolute Error (MAE) : 2.7428
          Mean Percentage Error (MPE) : -2.9628
Mean Absolute Percentage Error (MAPE) : 13.9356
```

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

   a. After performing the Exhaustive Search algorithm on the all of the predictors of the linear regression model, we can see below that the 10th iteration of the search performs the best as it maximizes the adjusted R-squared score and minimizes the Akaike Information Criterion (AIC).

	n	r2adj	AIC	AGE	CHAR_RIV_Y	CRIME	C_MVALUE_Yes	DISTANCE	INDUST	LOW_STAT	NIT_OXIDE	RADIAL	ROOMS	ST_RATIO	TAX	ZONE
0	1	0.615757	2227.470343	False	False	False	True	False	False	False	False	False	False	False	False	False
1	2	0.784502	2023.736517	False	False	False	True	False	False	True	False	False	False	False	False	False
2	3	0.793737	2009.222342	False	False	True	True	False	False	True	False	False	False	False	False	False
3	4	0.800829	1997.822810	False	True	True	True	False	False	True	False	False	False	False	False	False
4	5	0.804618	1992.008003	False	False	False	True	True	False	True	True	False	False	True	False	False
5	6	0.811403	1980.477479	False	True	False	True	True	False	True	True	False	False	True	False	False
6	7	0.816868	1971.047129	False	True	True	True	True	False	True	True	False	False	True	False	False
7	8	0.822139	1961.682655	False	True	True	True	True	False	True	True	True	False	True	False	False
8	9	0.822845	1961.248007	False	True	True	True	True	True	True	True	True	False	True	False	False
9	10	0.824545	1958.803262	False	True	True	True	True	True	True	True	True	False	True	True	False
10	11	0.824282	1960.300013	False	True	True	True	True	True	True	True	True	True	True	True	False
11	12	0.823903	1962.027384	False	True	True	True	True	True	True	True	True	True	True	True	True
12	13	0.823556	1963.683649	True	True	True	True	True	True	True	True	True	True	True	True	True

```
# Identify predictors and outcome of the regression model. n = 10
predictors_ex = ['CHAR_RIV_Y', 'CRIME', 'C_MVALUE_Yes', 'DISTANCE', 'INDUST', 'LOW_STAT', 'NIT_OXIDE', 'RADIAL',
                 'ROOMS', 'ST_RATIO', 'TAX']
outcome = 'MVALUE'
```

Above we can see the predictors that were chosen based on the Exhaustive Search and below we can see the intercept and coefficients along with the mathematical equation of this linear regression model. Furthermore, the common accuracy measures for the validation partition are displayed at the start of the next page.

MVALUE = 48.69 +
(-0.15)CRIME + (0.13)INDUST +
(-18.30)NIT_OXIDE + (0.29)ROOMS +

(-0.69)DISTANCE + (0.22)RADIAL + (-0.01)TAX +
(-0.62)ST_RATIO + (-0.48)LOW_STAT +

(2.31)CHAR_RIV_Y + (11.98)C_MVALUE_Yes

```
Regression Model for Training Set Using Exhaustive Search

Intercept  48.69
```

	Predictor	Coefficient
0	CHAR_RIV_Y	2.31
1	CRIME	-0.15
2	C_MVALUE_Yes	11.98
3	DISTANCE	-0.69
4	INDUST	0.13
5	LOW_STAT	-0.48
6	NIT_OXIDE	-18.30
7	RADIAL	0.22
8	ROOMS	0.29
9	ST_RATIO	-0.62
10	TAX	-0.01

```
Accuracy Measures for Validation Set - Exhaustive Search feature selection

Regression statistics

                      Mean Error (ME) : 0.3628
       Root Mean Squared Error (RMSE) : 3.6801
            Mean Absolute Error (MAE) : 2.7244
          Mean Percentage Error (MPE) : -2.9382
Mean Absolute Percentage Error (MAPE) : 13.8210
```

b. After performing Backwards Elimination on all the predictors of the linear regression model, we can see which predictors were considered the best.

```
Variables: CRIME, ZONE, INDUST, NIT_OXIDE, ROOMS, AGE, DISTANCE, RADIAL, TAX, ST_RATIO, LOW_STAT, CHAR_RIV_Y, C_MVALUE_Yes
Start: score=1963.68
Step: score=1962.03, remove AGE
Step: score=1960.30, remove ZONE
Step: score=1958.80, remove ROOMS
Step: score=1958.80, remove None

Best Variables from Backward Elimination Algorithm
['CRIME', 'INDUST', 'NIT_OXIDE', 'DISTANCE', 'RADIAL', 'TAX', 'ST_RATIO', 'LOW_STAT', 'CHAR_RIV_Y', 'C_MVALUE_Yes']
```

Below we can see the intercept and coefficients along with the mathematical equation of this linear regression model. Furthermore, the common accuracy measures for the validation partition are displayed.

MVALUE = 50.82 + (-0.15)CRIME + (0.13)INDUST + (-18.39)NIT_OXIDE + (-0.69)DISTANCE + (0.23)RADIAL + (-0.01)TAX + (-0.63)ST_RATIO + (-0.49)LOW_STAT + (2.34)CHAR_RIV_Y + (12.19)C_MVALUE_Yes

Regression Model for Training Set Using Backward Elimination

Intercept 50.82

	Predictor	Coefficient
0	CRIME	-0.15
1	INDUST	0.13
2	NIT_OXIDE	-18.39
3	DISTANCE	-0.69
4	RADIAL	0.23
5	TAX	-0.01
6	ST_RATIO	-0.63
7	LOW_STAT	-0.49
8	CHAR_RIV_Y	2.34
9	C_MVALUE_Yes	12.19

```
Accuracy Measures for Validation Set - Backward Elimination

Regression statistics

                      Mean Error (ME) : 0.3854
       Root Mean Squared Error (RMSE) : 3.7318
            Mean Absolute Error (MAE) : 2.7591
          Mean Percentage Error (MPE) : -2.8698
Mean Absolute Percentage Error (MAPE) : 13.9371
```

Analysis: The differences between the Exhaustive Search and the Backwards Elimination models are as follows:

i. Feature Space: Exhaustive Search keeps the 'Rooms' predictor that Backwards Elimination removes.
ii. Number of predictors: Exhaustive Search has 11 predictors and since Backwards Elimination removes the 'Rooms' predictor, it has one less i.e. 10 predictors.
iii. Accuracy Measures: Backwards Elimination does not perform as well as the Exhaustive Search model as all of the measures except MPE produce a higher magnitude of error.

c. Here are the common accuracy of all of the models produced in this work.

All predictors

```
Accuracy Measures for Validation Set - All Predictors

Regression statistics

                      Mean Error (ME) : 0.3667
       Root Mean Squared Error (RMSE) : 3.6868
            Mean Absolute Error (MAE) : 2.7428
          Mean Percentage Error (MPE) : -2.9628
Mean Absolute Percentage Error (MAPE) : 13.9356
```

Exhaustive Search

```
Accuracy Measures for Validation Set - Exhaustive Search feature selection

Regression statistics

                      Mean Error (ME) : 0.3628
       Root Mean Squared Error (RMSE) : 3.6801
            Mean Absolute Error (MAE) : 2.7244
          Mean Percentage Error (MPE) : -2.9382
Mean Absolute Percentage Error (MAPE) : 13.8210
```

Backwards Elimination

```
Accuracy Measures for Validation Set - Backward Elimination

Regression statistics

                      Mean Error (ME) : 0.3854
       Root Mean Squared Error (RMSE) : 3.7318
            Mean Absolute Error (MAE) : 2.7591
          Mean Percentage Error (MPE) : -2.8698
Mean Absolute Percentage Error (MAPE) : 13.9371
```

Conclusion: Upon analysis, the Exhaustive Search model exhibited the smallest RMSE compared to the 'All Predictors' and Backward Elimination models. This indicates that the

Exhaustive Search model is more accurate in predicting the target variable based on the validation data set. Although the Exhaustive Search model may be more complex due to the inclusion of almost all predictors, it strikes a balance between complexity and accuracy. On the other hand, the Backward Elimination model, while simpler due to its iterative elimination of predictors, may overlook important variables or relationships in the data, resulting in higher prediction errors.

Therefore, based on our analysis, we recommend using the Exhaustive Search model for making predictions in this case. Its superior performance in reducing prediction errors makes it a more reliable choice for accurate predictions, despite its slightly higher complexity compared to the Backward Elimination model but still not being as complex as the ‘All Predictors’ model.

Model Choice: Linear Regression using Exhaustive Search Algorithm