Predicting Housing Prices in Boston

Jayant Babu

2023-02-20

Introduction

In this report, we will be predicting the median house prices in Boston by using the Boston housing dataset. The Boston Housing dataset contains information about housing values in the suburbs of Boston, including 506 observations and 14 variables, such as per capita crime rate, average number of rooms per dwelling, and the median value of owner-occupied homes. In this project, we utilized various techniques for data exploration in R, including summary statistics, histograms, correlation matrices, heatmaps, and scatterplots and we finally use a linear regression model to predict the pricing of houses.

```
1. CRIM       per capita crime rate by town
2. ZN         proportion of residential land zoned for lots over
              25,000 sq.ft.
3. INDUS      proportion of non-retail business acres per town
4. CHAS       Charles River dummy variable (= 1 if tract bounds
              river; 0 otherwise)
5. NOX        nitric oxides concentration (parts per 10 million)
6. RM         average number of rooms per dwelling
7. AGE        proportion of owner-occupied units built prior to 1940
8. DIS        weighted distances to five Boston employment centres
9. RAD        index of accessibility to radial highways
10. TAX       full-value property-tax rate per $10,000
11. PTRATIO   pupil-teacher ratio by town
12. B         1000(Bk - 0.63)^2 where Bk is the proportion of blacks
              by town
13. LSTAT     % lower status of the population
14. MEDV      Median value of owner-occupied homes in $1000's
```

Figure 1: Column Names and what they mean

Load the Dataset

We load the dataset from the MASS library and print the first 5 entries and the summary of the dataset.

```
library(MASS)
housing <- Boston
summary(Boston)
```

```
##       crim                zn             indus            chas        
##  Min.   : 0.00632   Min.   :  0.00   Min.   : 0.46   Min.   :0.00000  
##  1st Qu.: 0.08205   1st Qu.:  0.00   1st Qu.: 5.19   1st Qu.:0.00000  
##  Median : 0.25651   Median :  0.00   Median : 9.69   Median :0.00000  
##  Mean   : 3.61352   Mean   : 11.36   Mean   :11.14   Mean   :0.06917  
##  3rd Qu.: 3.67708   3rd Qu.: 12.50   3rd Qu.:18.10   3rd Qu.:0.00000  
##  Max.   :88.97620   Max.   :100.00   Max.   :27.74   Max.   :1.00000  
##       nox               rm             age              dis        
##  Min.   :0.3850   Min.   :3.561   Min.   :  2.90   Min.   : 1.130  
##  1st Qu.:0.4490   1st Qu.:5.886   1st Qu.: 45.02   1st Qu.: 2.100  
##  Median :0.5380   Median :6.208   Median : 77.50   Median : 3.207  
##  Mean   :0.5547   Mean   :6.285   Mean   : 68.57   Mean   : 3.795  
##  3rd Qu.:0.6240   3rd Qu.:6.623   3rd Qu.: 94.08   3rd Qu.: 5.188  
##  Max.   :0.8710   Max.   :8.780   Max.   :100.00   Max.   :12.127  
##       rad              tax           ptratio          black       
##  Min.   : 1.000   Min.   :187.0   Min.   :12.60   Min.   :  0.32  
##  1st Qu.: 4.000   1st Qu.:279.0   1st Qu.:17.40   1st Qu.:375.38  
##  Median : 5.000   Median :330.0   Median :19.05   Median :391.44  
##  Mean   : 9.549   Mean   :408.2   Mean   :18.46   Mean   :356.67  
##  3rd Qu.:24.000   3rd Qu.:666.0   3rd Qu.:20.20   3rd Qu.:396.23  
##  Max.   :24.000   Max.   :711.0   Max.   :22.00   Max.   :396.90  
##      lstat            medv      
##  Min.   : 1.73   Min.   : 5.00  
##  1st Qu.: 6.95   1st Qu.:17.02  
##  Median :11.36   Median :21.20  
##  Mean   :12.65   Mean   :22.53  
##  3rd Qu.:16.95   3rd Qu.:25.00  
##  Max.   :37.97   Max.   :50.00
```

```
head(housing)
```

```
##      crim zn indus chas   nox    rm  age    dis rad tax ptratio  black lstat
## 1 0.00632 18  2.31    0 0.538 6.575 65.2 4.0900   1 296    15.3 396.90  4.98
## 2 0.02731  0  7.07    0 0.469 6.421 78.9 4.9671   2 242    17.8 396.90  9.14
## 3 0.02729  0  7.07    0 0.469 7.185 61.1 4.9671   2 242    17.8 392.83  4.03
## 4 0.03237  0  2.18    0 0.458 6.998 45.8 6.0622   3 222    18.7 394.63  2.94
## 5 0.06905  0  2.18    0 0.458 7.147 54.2 6.0622   3 222    18.7 396.90  5.33
## 6 0.02985  0  2.18    0 0.458 6.430 58.7 6.0622   3 222    18.7 394.12  5.21
##   medv
## 1 24.0
## 2 21.6
## 3 34.7
## 4 33.4
## 5 36.2
## 6 28.7
```

Data Cleaning

So there are several missing values in the dataset so we will be removing all the rows with missing entries.

```
numberOfNA <- length(which(is.na(housing)==T))
if(numberOfNA>0) {
```

```
  housing <- housing[complete.cases(housing),]
}
```

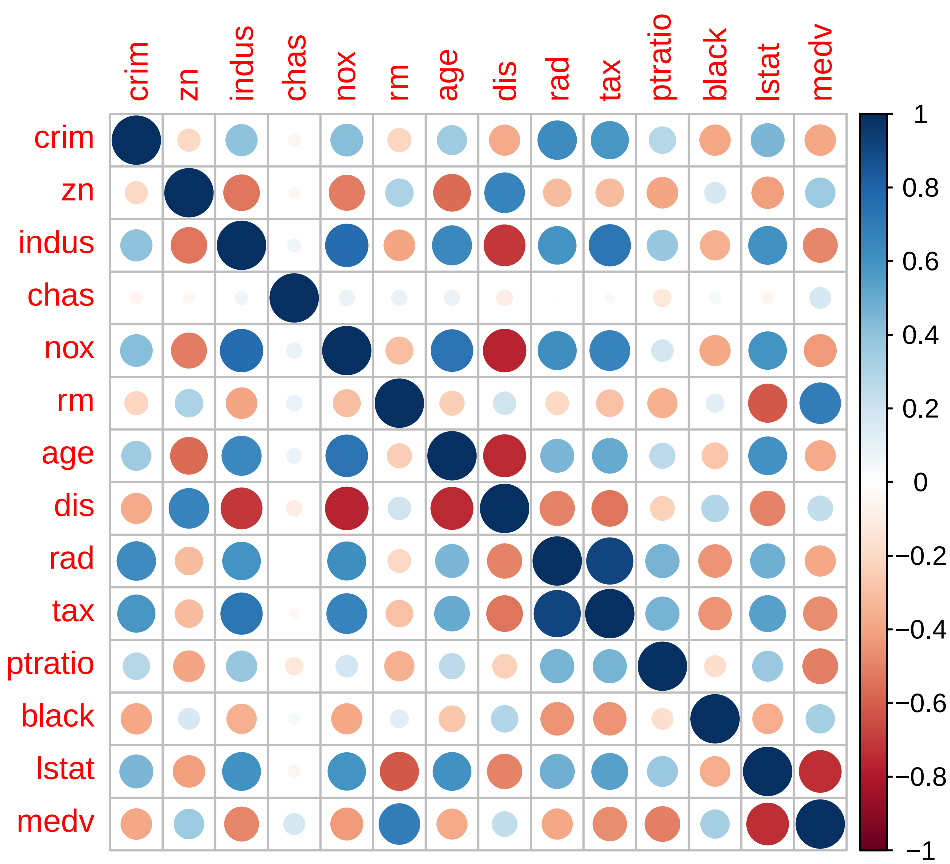
Plotting - Exploring the dataset

Plotting the dataset to get a better understanding of the data.

Correlation Plot and Heatmap

The correlation plot, is a visualization of the correlation matrix for the Boston Housing dataset. The correlation matrix shows the correlation coefficients between each pair of features in the dataset, with values ranging from -1 to 1. A value of 1 indicates a perfect positive correlation between two features, while a value of -1 indicates a perfect negative correlation. A value of 0 indicates no correlation between the features.

```
library(corrplot)
corrplot(cor(housing))
```

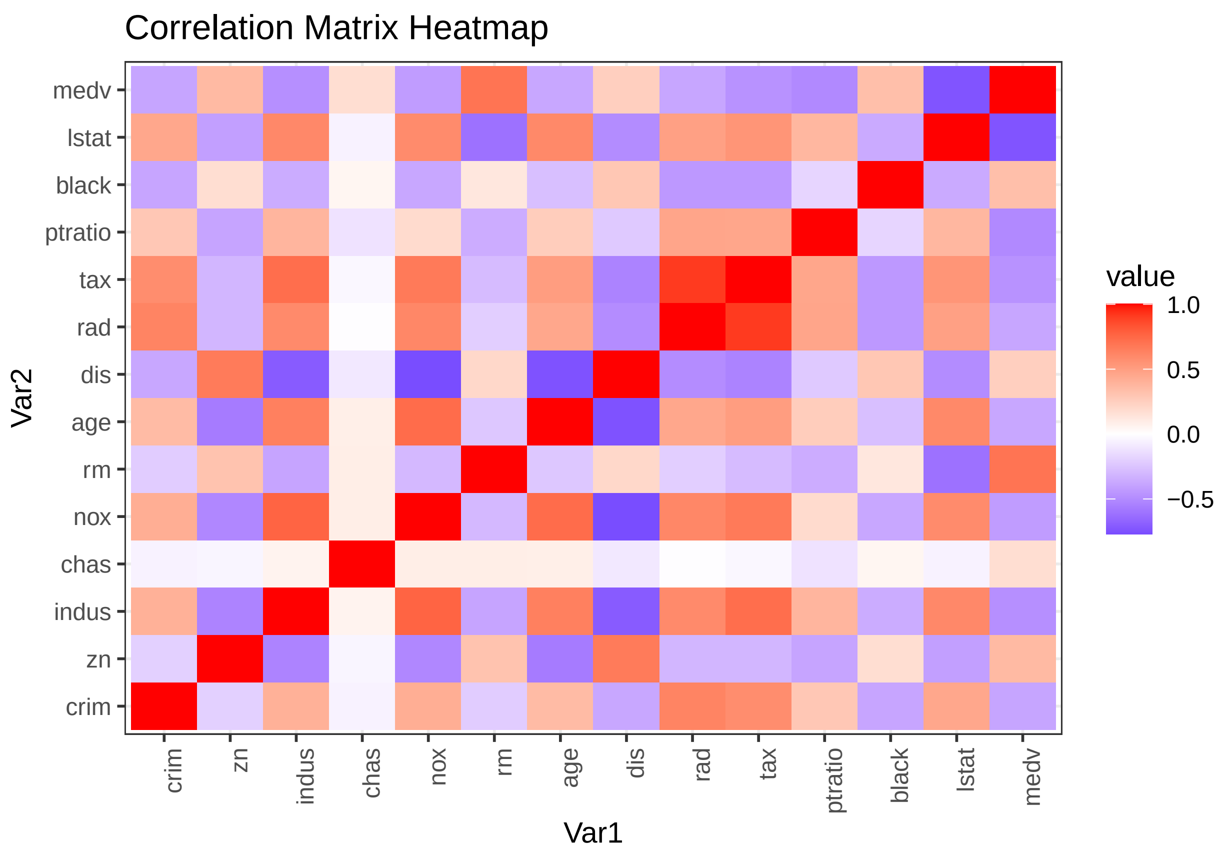



```
correlation_matrix <- cor(housing)
```

```
library(ggplot2)
library(reshape2)

melted_correlation_matrix <- melt(correlation_matrix)
```

```
ggplot(melted_correlation_matrix, aes(x = Var1, y = Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient2(low = "blue", high = "red", mid = "white", midpoint = 0) +
  theme_bw() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  labs(title = "Correlation Matrix Heatmap")
```

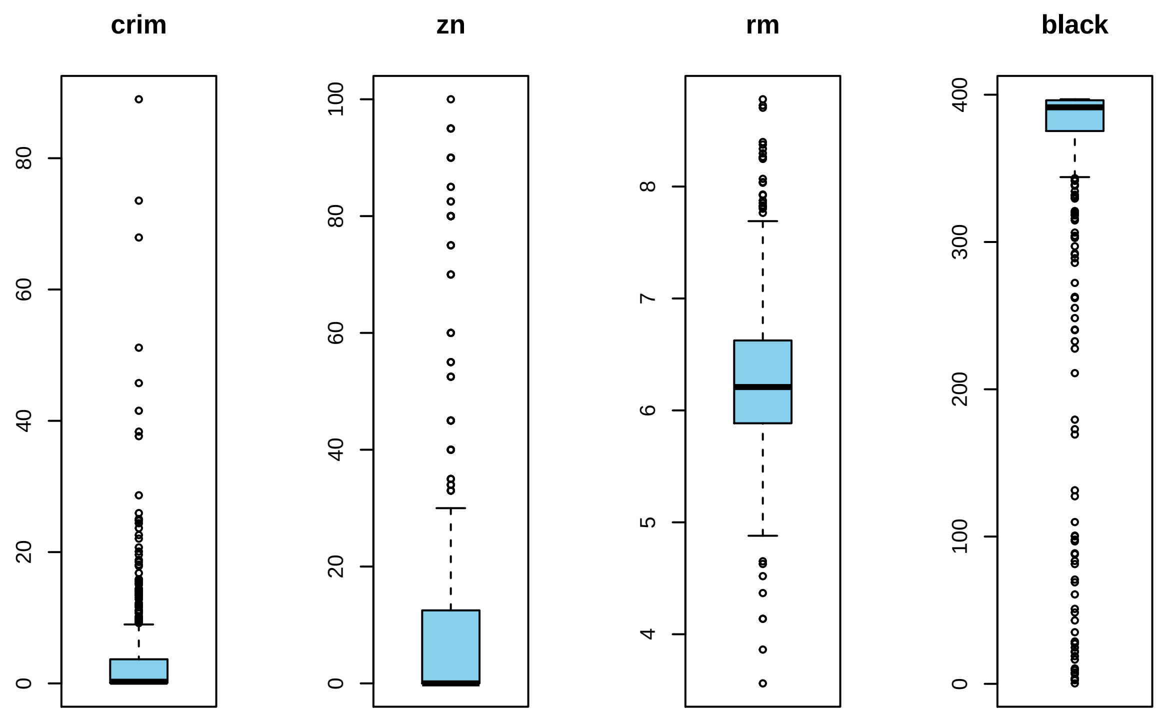


The correlation plot can help us identify patterns and relationships within the dataset. We can see that the rm variable (average number of rooms per dwelling) has a strong positive correlation with the medv variable (median value of owner-occupied homes), while the lstat variable (percent of lower status population) has a strong negative correlation with medv

Boxplots

Now when we look at the summary we can see that variables ‘crim’, ‘zn’, ‘rm’ and ‘black’ have a large difference between their median and mean which indicates lot of outliers in respective variables.

```
{
par(mfrow = c(1, 4))
boxplot(housing$crim, main='crim',col='Sky Blue')
boxplot(housing$zn, main='zn',col='Sky Blue')
boxplot(housing$rm, main='rm',col='Sky Blue')
boxplot(housing$black, main='black',col='Sky Blue')}
```



Splitting the Dataset

We will be splitting the dataset into train and test data where 75% is training data and the rest for testing.

```
set.seed(123)
train_indices <- sample(nrow(housing), round(0.75 * nrow(housing)))
train_data <- housing[train_indices, ]
test_data <- housing[-train_indices, ]
```

Linear Regression Model

In this project we will be using a linear regression model to predict the median value of owner-occupied homes (medv) based on a set of input variables from the Boston Housing dataset. The linear regression model is a type of statistical model that uses a linear function to model the relationship between a dependent variable (medv) and one or more independent variables (crim, zn, indus, etc.). The linear regression model provides a simple method for modeling the relationship between the medv variable and the input variables in the Boston Housing dataset.

```
lm.fit1 <- lm(medv~.,data=train_data)
summary(lm.fit1)
```

```
##
## Call:
```

```
## lm(formula = medv ~ ., data = train_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.6847  -2.6971  -0.5087   1.5846  24.6123
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  38.898461   5.856832   6.642 1.12e-10 ***
## crim         -0.103197   0.034685  -2.975 0.003122 **
## zn            0.054350   0.015781   3.444 0.000640 ***
## indus        -0.007058   0.074743  -0.094 0.924817
## chas          3.659667   0.969875   3.773 0.000188 ***
## nox         -16.442841   4.442579  -3.701 0.000248 ***
## rm            3.339814   0.479463   6.966 1.52e-11 ***
## age           0.002727   0.015395   0.177 0.859480
## dis          -1.552263   0.229313  -6.769 5.16e-11 ***
## rad           0.292258   0.075635   3.864 0.000132 ***
## tax          -0.010406   0.004382  -2.374 0.018089 *
## ptratio      -0.889898   0.152092  -5.851 1.08e-08 ***
## black         0.006765   0.003336   2.028 0.043304 *
## lstat        -0.599322   0.057264 -10.466  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.73 on 366 degrees of freedom
## Multiple R-squared:  0.7472, Adjusted R-squared:  0.7382
## F-statistic:  83.2 on 13 and 366 DF,  p-value: < 2.2e-16
```

Accuracy Score

```r
predicted <- predict(lm.fit1, newdata = test_data)

MSE <- mean((predicted - test_data$medv)^2)
MSE
```

```
## [1] 24.04534
```

So from the summary we can see that the age, indus and zn variables have a really high p-value therefore skewing our model so to get a more accurate model we need to remove these variables and retrain.

```r
lm.fit2 <- lm(medv~.-indus-age-zn+rm*lstat-black+rm*rad+lstat*rad,data=train_data)
summary(lm.fit2)
```

```
##
## Call:
## lm(formula = medv ~ . - indus - age - zn + rm * lstat - black +
##     rm * rad + lstat * rad, data = train_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -9.7006 -1.9798 -0.1774  1.5307 23.4414
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -28.616210   6.196881  -4.618 5.37e-06 ***
## crim         -0.102692   0.026813  -3.830 0.000151 ***
## chas          2.660638   0.734839   3.621 0.000335 ***
## nox         -11.134797   3.148502  -3.537 0.000457 ***
## rm           12.014801   0.634853  18.925  < 2e-16 ***
## dis          -0.743345   0.140209  -5.302 1.99e-07 ***
## rad           2.749042   0.309951   8.869  < 2e-16 ***
## tax          -0.007967   0.002812  -2.833 0.004863 **
## ptratio      -0.624575   0.112722  -5.541 5.76e-08 ***
## lstat         2.015714   0.244179   8.255 2.77e-15 ***
## rm:lstat     -0.364947   0.040650  -8.978  < 2e-16 ***
## rm:rad       -0.322988   0.047089  -6.859 2.95e-11 ***
## rad:lstat    -0.035532   0.004007  -8.868  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.584 on 367 degrees of freedom
## Multiple R-squared:  0.8544, Adjusted R-squared:  0.8496
## F-statistic: 179.5 on 12 and 367 DF,  p-value: < 2.2e-16
```

Accuracy Score

```
predicted <- predict(lm.fit2, newdata = test_data)

MSE <- mean((predicted - test_data$medv)^2)
MSE
```

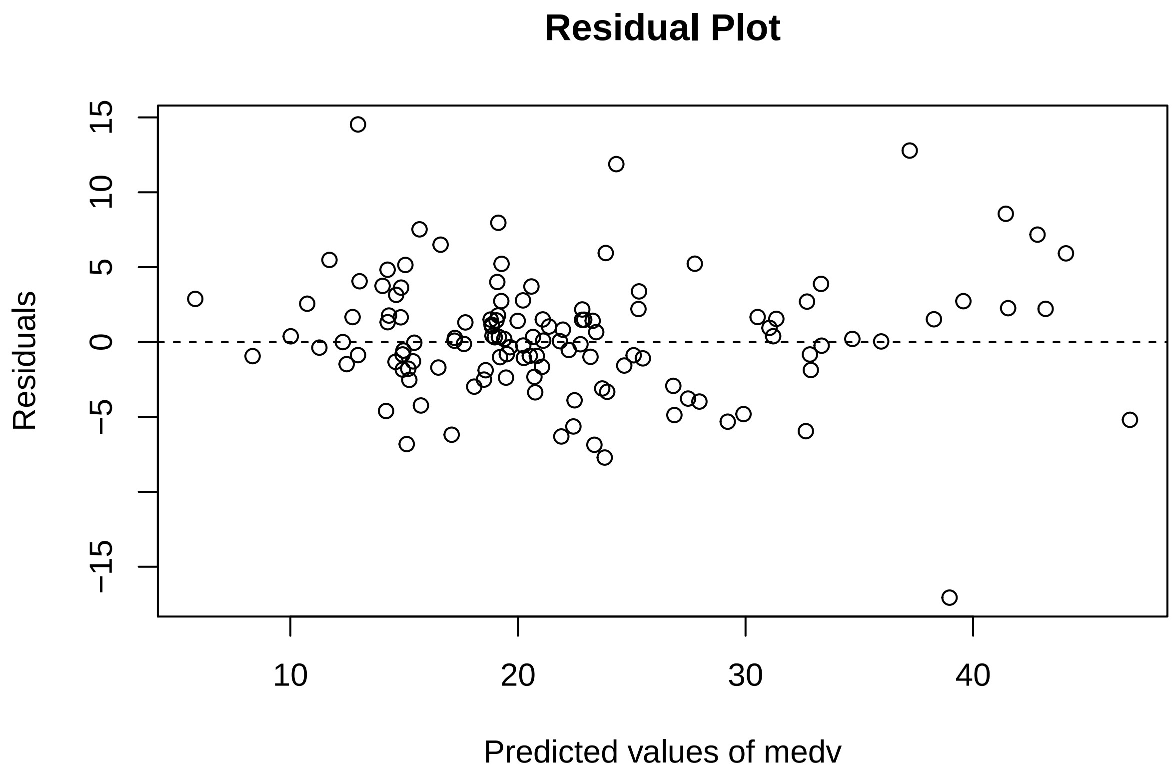
```
## [1] 16.77564
```

Residual Plot

In this project, residuals refer to the differences between the actual values of the dependent variable (medv) and the predicted values of medv based on the linear regression model.

```
pred <- predict(lm.fit2, newdata = test_data)
residuals <- test_data$medv - pred

{plot(pred, residuals, xlab = "Predicted values of medv", ylab = "Residuals", main = "Residual Plot")
abline(h = 0, lty = 2)}
```



So from this plot and with the MSE of 16.77 we can tell that our model is pretty accurate to predict the pricing of houses in Boston.

Prediction

So now we will be creating an example house with all the input variables and try predicting the median-value price of the house.

```
new_house <- data.frame(crim = 0.1, zn = 0, indus = 8, chas = 0, nox = 0.5,
                        rm = 5, age = 60, dis = 5, rad = 4, tax = 300,
                        ptratio = 15, black = 300, lstat = 15)

price <- predict(lm.fit2, newdata = new_house)
price
```

```
##        1
## 15.67389
```

Conclusion

In conclusion, we can use the linear regression model to predict the median value of owner-occupied homes based on the input variables from the Boston Housing dataset with reasonable accuracy. However, there may be other factors that influence the value of medv that are not captured by the model, and it is important to interpret the results with caution and validate the model using additional datasets or techniques.