Multiple Linear Regression on Boston Housing Dataset

Shreyansh Misra, Luke Pasterczyk, Akhila Garre, Maxwell Owens, and Maithili Revankar

2025-03-10

1 Introduction

Multiple Linear Regression estimates the relationship between a quantitative dependent variable Y and multiple independent variables $X_1, X_2, \ldots$. The relationship between the dependent variable Y and independent variables X_n is assumed to be linear. The equation takes the form $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \varepsilon$ where β_0 and β_1 are calculated such that the sum of the squares of the vertical differences between observed and predicted points is minimized. The equation is then used to predict values of Y for a given X_n values.

This investigation focused on predicting the median value of homes in a Boston neighborhood based on 12 predictors. We used the Boston Housing dataset, sourced from the StatLib archive (http://lib.stat.cmu.edu/datasets/boston), which was compiled based on the 1970 U.S Census. It contains records from 506 unlabelled neighborhoods in Boston.

R packages used and

result / formula

2 Data description

```
housing.df <- read.csv("boston_house_prices.csv")
num_rows <- nrow(housing.df)
sum_missingdata <- sum(is.na(housing.df))

cat("Number of Rows: ", num_rows, "   Rows with Missing Data: ", sum_missingdata)
```

```
## Number of Rows:  506    Rows with Missing Data:  0
```

2.1 Dataset

This dataset contains information about 506 neighborhoods in Boston, collected by the U.S Census Service in 1970 census. There are 506 records and 13 variables in the dataset. From an initial analysis, there were no missing data points.

```
head(housing.df, 3)
```

```
##      crim zn indus chas   nox    rm  age    dis rad tax ptratio lstat medv
## 1 0.00632 18  2.31    0 0.538 6.575 65.2 4.0900   1 296    15.3  4.98 24.0
## 2 0.02731  0  7.07    0 0.469 6.421 78.9 4.9671   2 242    17.8  9.14 21.6
## 3 0.02729  0  7.07    0 0.469 7.185 61.1 4.9671   2 242    17.8  4.03 34.7
```

2.2 Variables

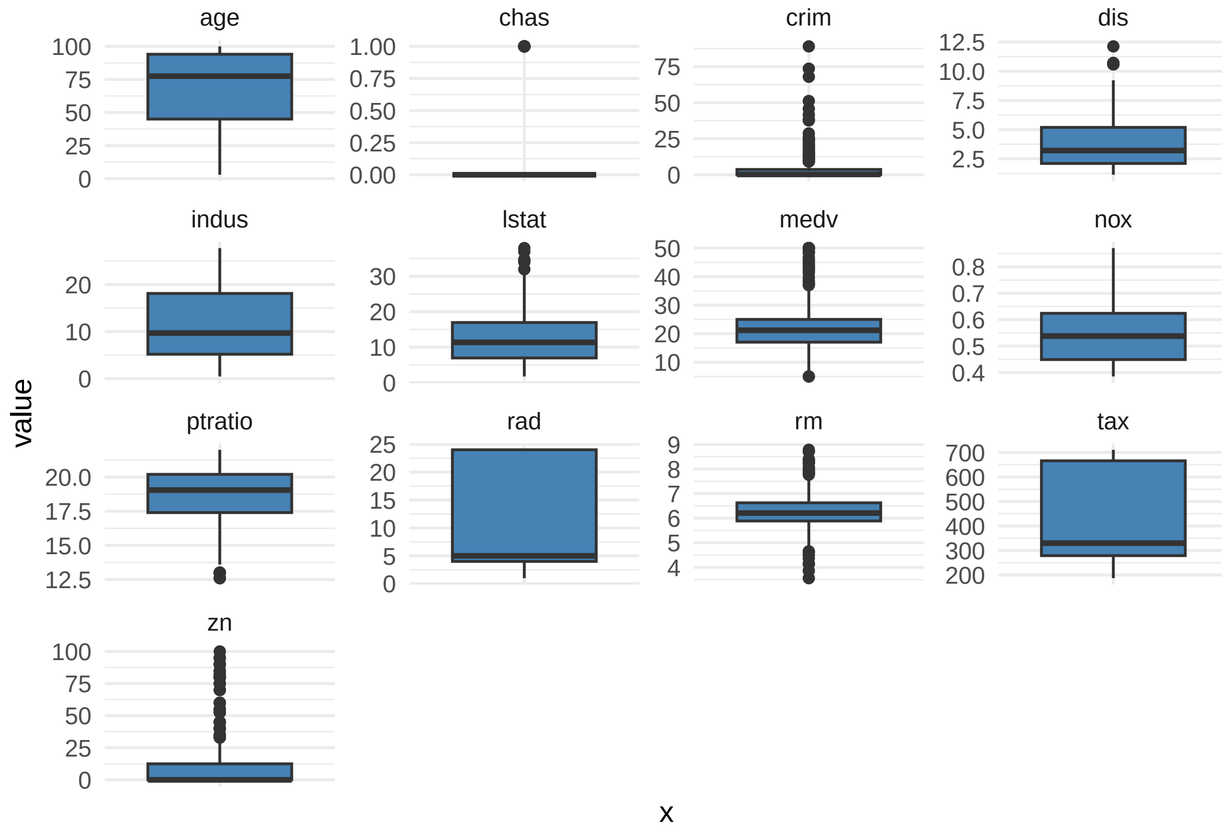
- `crim`: per capita crime rate by town
- `zn`: proportion of residential land zoned for lots over 25,000 sq.ft
- `indus`: proportion of non-retail business acres per town
- `chas`: Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)
- `nox`: nitric oxides concentration (parts per 10 million)
- `rm`: average number of rooms per dwelling
- `age`: proportion of owner-occupied units built prior to 1940
- `dis`: weighted distances to five Boston employment centres
- `rad`: index of accessibility to radial highways
- `tax`: full-value property-tax rate per USD 10,000
- `ptratio`: pupil-teacher ratio by town
- `lstat`: percentage of lower status of the population
- `medv`: median value of owner-occupied homes in USD 1000's

We can identify `medv` as our dependent variable as the median value of homes in the neighborhood is what we are predicting. The remaining 12 variables are our independent variables.

2.3 Outlier Detection

```r
housing_box <- housing.df %>%
  pivot_longer(cols = everything(), names_to = "variable", values_to = "value")

ggplot(housing_box, aes(x = "", y = value)) +
  geom_boxplot(fill = "steelblue") +
  theme_minimal() +
  facet_wrap(~ variable, scales = "free_y") +
  theme(axis.text.x = element_blank(), axis.ticks.x = element_blank())
```



3 Analysis

3.1 Train and Test Split

Our dataset was split into a training and testing set, where approximately 70% of the dataset is used for training and the remaining 30% is used for testing.

```
set.seed(123)

split <- 0.75

trainIndex <- createDataPartition(housing.df$medv, p = split)
trainIndex <- unlist(trainIndex)

train <- housing.df[trainIndex, ]
test <- housing.df[-trainIndex, ]

num_row_train <- nrow(train)
num_row_test <- nrow(test)

cat("Number of Rows in Train Set: ", num_row_train, "   Number of Rows in Test Set: ", num_row_test)
```

```
## Number of Rows in Train Set:  381    Number of Rows in Test Set:  125
```

Specifically, 381 records were used for training and 125 were reserved for testing.

```r
train_control <- trainControl(method="LOOCV")
model <- train(medv ~ . , method="lm", data = train, trControl=train_control)
print(model$results)
```

```
##   intercept    RMSE  Rsquared      MAE
## 1      TRUE 4.36563 0.7396655 3.126959
```

```r
summary(model)
```

```
## 
## Call:
## lm(formula = .outcome ~ ., data = dat)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -12.2140  -2.6532  -0.5244   1.6582  21.4295 
## 
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  37.304292   4.870431   7.659 1.67e-13 ***
## crim         -0.089699   0.034513  -2.599 0.009726 ** 
## zn            0.027534   0.013855   1.987 0.047636 *  
## indus        -0.015256   0.063180  -0.241 0.809323    
## chas          1.590398   0.938478   1.695 0.090986 .  
## nox         -17.237834   3.953790  -4.360 1.69e-05 ***
## rm            3.915358   0.427428   9.160  < 2e-16 ***
## age           0.003283   0.013761   0.239 0.811548    
## dis          -1.130373   0.196597  -5.750 1.88e-08 ***
## rad           0.246654   0.068341   3.609 0.000350 ***
## tax          -0.013630   0.003833  -3.556 0.000426 ***
## ptratio      -0.929882   0.132271  -7.030 1.01e-11 ***
## lstat        -0.448149   0.056305  -7.959 2.17e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 4.212 on 368 degrees of freedom
## Multiple R-squared:  0.7656, Adjusted R-squared:  0.758 
## F-statistic: 100.2 on 12 and 368 DF,  p-value: < 2.2e-16
```

The variables `indus`, `age`, and `chas` have p-values of 0.809323, 0.811548, and 0.090986 respectively, all of which are > 0.05 making them insignificant predictors of `medv`. As they do not influence the median value of homes in Boston to an extent that can be deemed signficiant, we can remove them from the linear model.

```r
model_significant <- train(medv ~ crim + zn + nox + rm + dis + rad + tax + ptratio + lstat,
      method="lm",
      data = train,
      trControl=train_control)

print(model_significant$results)
```

```
##   intercept     RMSE  Rsquared     MAE
## 1      TRUE 4.327847 0.7440531 3.11785
```

```r
summary(model_significant)
```

```
##
## Call:
## lm(formula = .outcome ~ ., data = dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.9239  -2.5768  -0.4792   1.7263  22.9325
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  37.402698   4.829497   7.745 9.24e-14 ***
## crim         -0.092689   0.034412  -2.694  0.00739 **
## zn            0.028351   0.013673   2.073  0.03882 *
## nox         -16.656155   3.626163  -4.593 5.98e-06 ***
## rm            3.973394   0.409668   9.699  < 2e-16 ***
## dis          -1.154383   0.184428  -6.259 1.07e-09 ***
## rad           0.255001   0.066573   3.830  0.00015 ***
## tax          -0.014503   0.003494  -4.150 4.12e-05 ***
## ptratio      -0.944586   0.130482  -7.239 2.62e-12 ***
## lstat        -0.447242   0.051881  -8.621  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.212 on 371 degrees of freedom
## Multiple R-squared:  0.7638, Adjusted R-squared:  0.7581
## F-statistic: 133.3 on 9 and 371 DF,  p-value: < 2.2e-16
```

3.2 Assumptions of Multiple Linear Regression

1. Linearity - the relationship between the independent variables and dependent variable should be linear
2. Independence Of Errors - each data point's error should be independent of other points' errors (no observation should influence another)
3. Homoscedasticity - variance of the errors remains consistent across all values of the independent variable
4. Normality Of Errors - errors are normally distributed

4 Model Evaluation and Prediction

You should base on the training set you should hav gotten a model in (3), then you must use subset selection and all necessary test to verify the final model that you get is the best model (Model assessment and model accuracy) and use it to make a prediction

```r
predictions <- predict(model_significant, newdata = test)
actual <- test$medv

mae <- mean(abs(predictions - actual)) # MAE
mse <- mean((predictions - actual)^2) # MSE
```

```r
rmse <- sqrt(mse) # RMSE

mae
```

```
## [1] 4.134073
```

```r
mse
```

```
## [1] 43.42458
```

```r
rmse
```

```
## [1] 6.589733
```

5 Conclusion

The summary your all your work and results in this part and point out positive side of your model, negative side of your model and possible future work or any factors that affect your model accuracy.

6 References

- https://www.cs.toronto.edu/~delve/data/boston/bostonDetail.html
- https://www.sthda.com/english/articles/40-regression-analysis/168-multiple-linear-regression-in-r/