20th Dec 2019

Exp. #6A: Building a Simple Linear Regression Model

The process of performing linear regression involves complex calculations owing to the number of variables. With the help of R, one can implement inbuilt functions that allow performing linear regression easily. In this experiment a linear regression model using R is implemented.

In order to build our linear regression model, we will make use of the 'cars' dataset and analyse the relationship between the variables – speed and distance.

1. Importing the Dataset

The data give the speed of cars and the distances taken to stop.

```
require(stats); require(graphics)
> head(cars)          #Displaying the first 6 observations
  speed dist
1     4    2
2     4   10
3     7    4
4     7   22
5     8   16
6     9   10
```

2. Visualising Linearity using Scatterplots

In order to visualise the linear relationship between independent and dependent variables, that is, distance and speed respectively, we use a Scatterplot. This can be done using the scatter.smooth() function:

```
scatter.smooth(x=cars$speed, y = cars$dist, main="Dist ~ Speed")
```

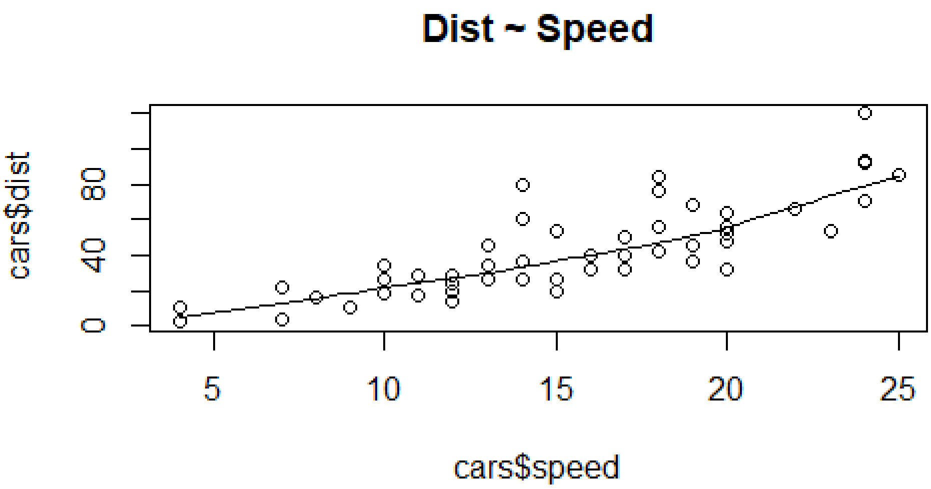


Fig. 1: Scatterplot of Speed and Stopping distance in ft

3. Measuring Correlation Coefficient

Correlation is a statistical measure of finding out linear dependence between the two variables. The values of the correlation coefficient range between -1 to +1. There can be two types of correlation – Positive Correlation and Negative Correlation.

In positive correlation, the value of one variable will increase with an increment in the other value or decrease with reduction in the other. This value will be closer to +1.

In the case of negative correlation, the value of the variable will decrease with an increase in another variable and increase with a decrease in other. Therefore, there is an inverse relationship between the two variables. The value of this correlation coefficient will be closer to -1.

We can evaluate the correlation coefficient through the following data of the cars:

```
> cor(cars$speed, cars$dist) #Finding Correlation between speed and distance
[1] 0.8068949
```

4. Building the Linear Model

In order to build our linear model, we will make use of the *lm()* function. We can write this function as follows:

```
> linear_model <- lm(dist~speed, data = cars)
> linear_model

Call:
lm(formula = dist ~ speed, data = cars)

Coefficients:
(Intercept)        speed
    -17.579        3.932
```

5. Diagnosing the Linear Model

After building our model, we can diagnose it by checking if it is statistically significant. In order to do so, we make use of the *summary()* function as follows:

```
> summary(linear_model)

Call:
lm(formula = dist ~ speed, data = cars)

Residuals:
    Min      1Q  Median      3Q     Max
-29.069  -9.525  -2.272   9.215  43.201

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -17.5791     6.7584  -2.601   0.0123 *
speed         3.9324     0.4155   9.464 1.49e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 15.38 on 48 degrees of freedom
Multiple R-squared:  0.6511,	Adjusted R-squared:  0.6438
F-statistic: 89.57 on 1 and 48 DF,  p-value: 1.49e-12
```

6. Calculating Standard Error and F – statistic

A Standard Error is the measurement of the standard deviation of the sample population from its mean.

$$\text{std\_error} = \sqrt{MSE}$$

Where MSE is the mean squared error. We can calculate the standard error in R as follows:

```
> Model_Summary <- summary(linear_model)
> Model_Coefficients <- Model_Summary$coefficients
> std_error <- Model_Coefficients["speed", "Std. Error"]
> std_error
[1] 0.4155128
```

Another measure of the goodness of fit is F-statistic:

$$F - Statistic \;=\; \frac{MSR}{MSE}$$

Where MSR is the mean squared regression:

```
> f_stat <- summary(linear_model)$fstatistic
> f_stat
   value    numdf    dendf
89.56711  1.00000 48.00000
```

The F-statistic that we obtained for this model is 89.56711. The other two values of *numdf* and *dendf* can be used as parameters for calculating the p-value.

Use Case of Linear Regression

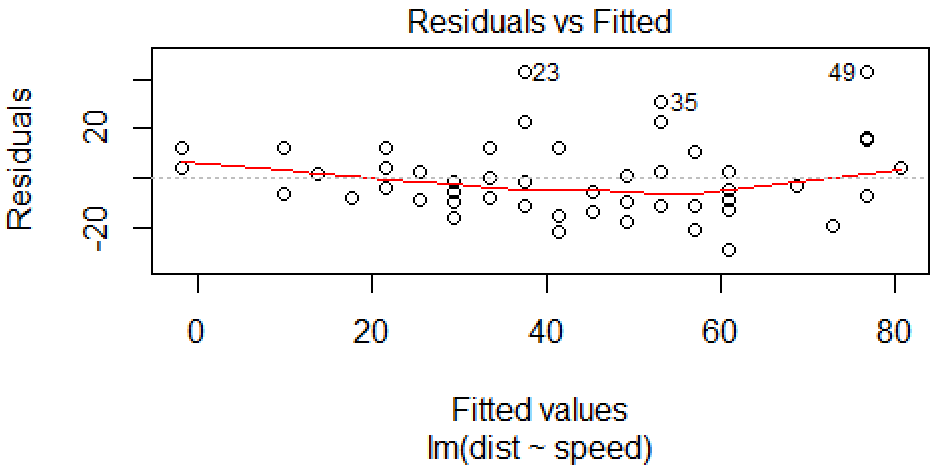
Linear Regression is used in various fields where a relationship between various instances (variables) is to be determined. Furthermore, with the determination of a relation, companies use linear regression to forecast future instances. Companies that have to increase their sales, use linear regression to identify the relationship between various factors and sale of their product.

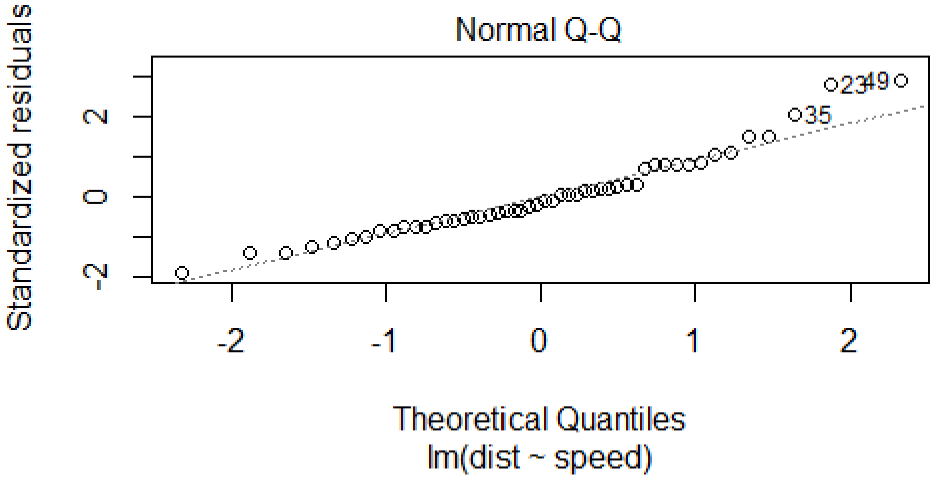
For example – A company might want to analyse any relation between the use of their product by certain age-groups. Therefore, after the identification of the relationship, they can forecast if the customer of a particular age will buy their product. Linear Regression is also used to predict the housing prices, price of the tickets and several other areas where one instance might affect another.

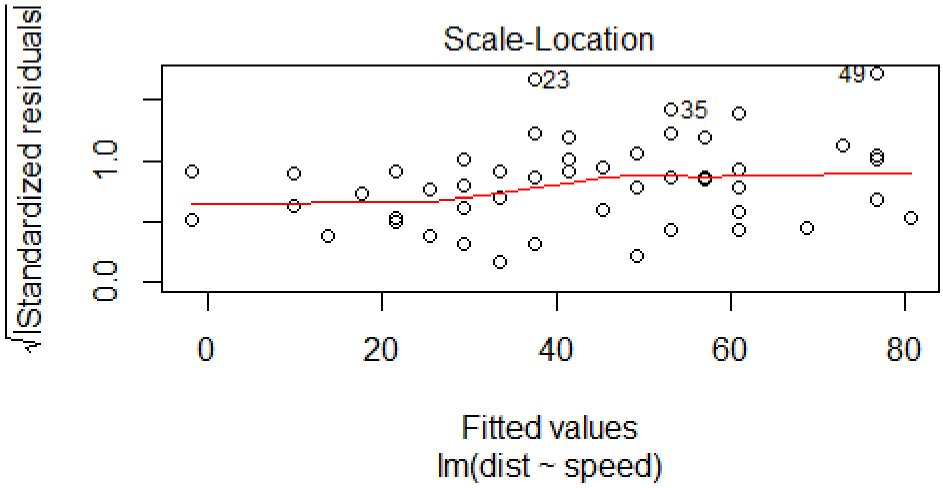
Summary

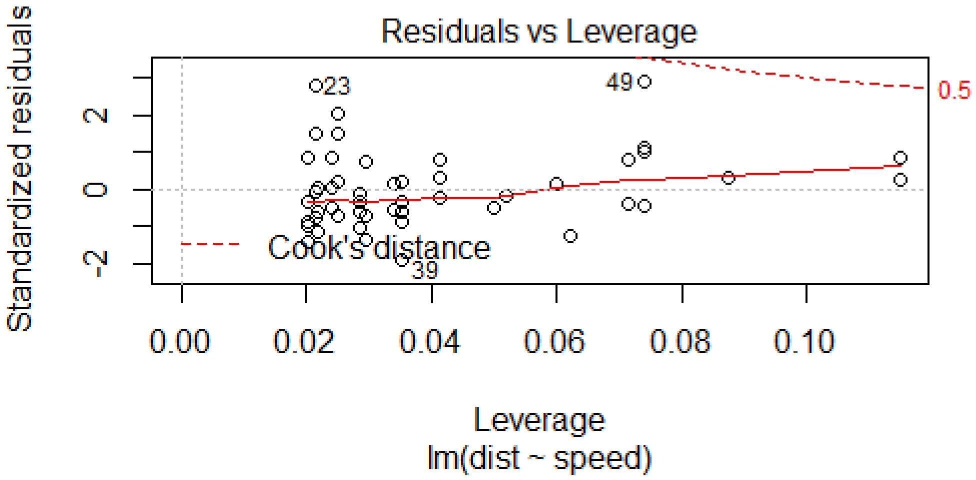
In this R experiment, we developed simple linear regression model using cars dataset. We saw the least square estimation and checking model accuracy along with model building and implementing simple linear regression in R.

```
plot(linear_model)
```


Residuals vs Fitted
Residuals
-20
20
0
20
40
60
80
23
35
49
Fitted values
lm(dist ~ speed)


Normal Q-Q
Standardized residuals
-2
0
2
-2
-1
0
1
2
23
49
35
Theoretical Quantiles
lm(dist ~ speed)


Scale-Location
√|Standardized residuals|
0.0
1.0
0
20
40
60
80
23
35
49
Fitted values
lm(dist ~ speed)


Residuals vs Leverage
Standardized residuals
-2
0
2
0.00
0.02
0.04
0.06
0.08
0.10
Leverage
lm(dist ~ speed)
23
49
39
0.5
Cook's distance