LinearRegression

Johnson Adebayo\_SCV1007 (Cohort 1)

4/3/2020

Source: <https://data-flair.training/blogs/r-linear-regression-tutorial/>

1. Importing the Dataset

data("cars")  
head(cars)

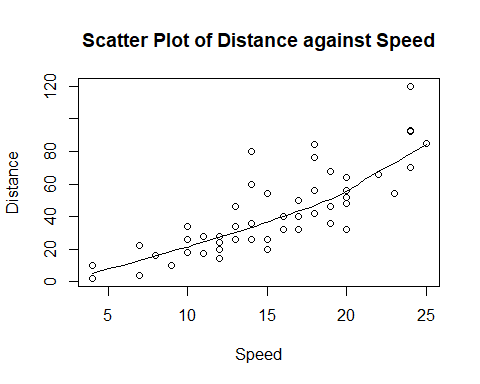
## speed dist  
## 1 4 2  
## 2 4 10  
## 3 7 4  
## 4 7 22  
## 5 8 16  
## 6 9 10

str(cars)

## 'data.frame': 50 obs. of 2 variables:  
## $ speed: num 4 4 7 7 8 9 10 10 10 11 ...  
## $ dist : num 2 10 4 22 16 10 18 26 34 17 ...

1. Visualising Linearity using Scatterplots

scatter.smooth(x=cars$speed, y = cars$dist, xlab ="Speed",ylab ="Distance", main="Scatter Plot of Distance against Speed")



1. Measuring Correlation Coefficient # The result below shows that there is a strong positive correlation btw speed and distance

cor(cars$speed, cars$dist) #Finding Correlation between speed and distance

## [1] 0.8068949

1. Building the Linear Model

# fitted linear model, dist-dependent variable, speed-independent variable  
linear\_model <- lm(dist~speed, data = cars)  
#summary(linear\_model)  
print(linear\_model)

##   
## Call:  
## lm(formula = dist ~ speed, data = cars)  
##   
## Coefficients:  
## (Intercept) speed   
## -17.579 3.932

1. 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: # The model is statistically significant (there is significant relationship btw dist and speed) because the null hypothesis is rejected (bcos p-value is 1.49e-12 $ less than 0.5)

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

1. Calculating Standard Error and F – statistic

Model\_Summary <- summary(linear\_model)  
Model\_Coefficients <- Model\_Summary$coefficients  
std\_error <- Model\_Coefficients["speed", "Std. Error"]  
print(std\_error)

## [1] 0.4155128

f\_stat <- summary(linear\_model)$fstatistic  
f\_stat

## value numdf dendf   
## 89.56711 1.00000 48.00000

linear\_model$coefficients

## (Intercept) speed   
## -17.579095 3.932409

(Model\_Summary$coefficients)

## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) -17.579095 6.7584402 -2.601058 1.231882e-02  
## speed 3.932409 0.4155128 9.463990 1.489836e-12

Model\_Coefficients["speed","t value"]

## [1] 9.46399

t1 <- summary(linear\_model)$coefficients  
t1

## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) -17.579095 6.7584402 -2.601058 1.231882e-02  
## speed 3.932409 0.4155128 9.463990 1.489836e-12