Linear Regression

18BCE1104 - Ankita Duraphe

04/02/2021

### Regression Analysis on sample data

Consider the following 5 training examples: X = [2 3 4 5 6], Y = [12.8978, 17.7586, 23.3192, 28.3129, 32.1351] We want to learn a function f(x) of the form f(x) = a(x) + b which is parametrized by (a, b). Using squared error as the loss function, which of the following parameters would you like to use to model this function.

Create a dataframe for the given data

data <- data.frame(x = c(2, 3, 4, 5, 6), y = c(12.8978, 17.7586, 23.3192, 28.3129, 32.1351))  
data

## x y  
## 1 2 12.8978  
## 2 3 17.7586  
## 3 4 23.3192  
## 4 5 28.3129  
## 5 6 32.1351

View the dataset

head(data)

## x y  
## 1 2 12.8978  
## 2 3 17.7586  
## 3 4 23.3192  
## 4 5 28.3129  
## 5 6 32.1351

Number of rows and columns in the dataset

nrow(data)

## [1] 5

ncol(data)

## [1] 2

Checking summary(Mean, Mode, Median)

summary(data)

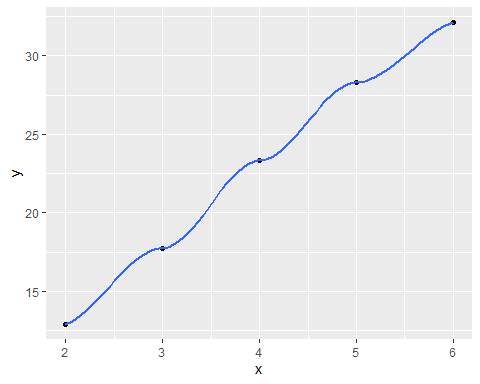
## x y   
## Min. :2 Min. :12.90   
## 1st Qu.:3 1st Qu.:17.76   
## Median :4 Median :23.32   
## Mean :4 Mean :22.88   
## 3rd Qu.:5 3rd Qu.:28.31   
## Max. :6 Max. :32.14

Check missing values

# is.na(mtcars)

Plot

library(ggplot2)  
ggplot(data, aes(x = x, y = y)) +  
 geom\_point() +  
 stat\_smooth()



Autocorrelation

cor(data$y, data$x)

## [1] 0.9982246

Applying Linear Regression Model

fit <- lm(y ~ x, data)

Checking Summary after applying Linear Regression to the Model

model <- lm(y ~ x, data = data)  
model

##   
## Call:  
## lm(formula = y ~ x, data = data)  
##   
## Coefficients:  
## (Intercept) x   
## 3.273 4.903

summary(model)

##   
## Call:  
## lm(formula = y ~ x, data = data)  
##   
## Residuals:  
## 1 2 3 4 5   
## -0.1811 -0.2232 0.4345 0.5253 -0.5554   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 3.2732 0.7166 4.568 0.0197 \*   
## x 4.9029 0.1689 29.028 8.98e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.5341 on 3 degrees of freedom  
## Multiple R-squared: 0.9965, Adjusted R-squared: 0.9953   
## F-statistic: 842.6 on 1 and 3 DF, p-value: 8.977e-05

**SE**

library(plotrix)  
std.error(data$y,na.rm)

## [1] 3.473033

**RSS**

library(qpcR)  
RSS(model)

## [1] 0.8558149

**95% Confidence Interval**

new.dat <- data.frame(x=10)  
predict(model, newdata = new.dat, interval = 'confidence')

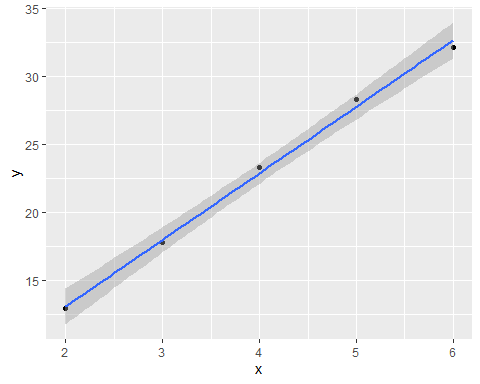
## fit lwr upr  
## 1 52.30206 48.9886 55.61552

From the output, the fitted stopping y at x = 10 is just above 52. The confidence interval of (48.9886, 55.61552) signifies the range in which the true population parameter lies at a 95% level of confidence.

**RSE Statistic**  
From the summary of the fitted model, Residual Standard Error: 0.5341 (on 3 degrees of freedom).

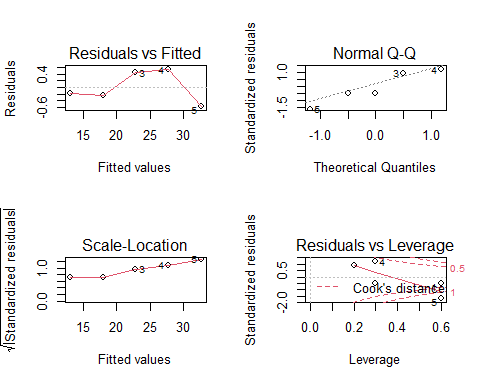
Visualization

ggplot(data, aes(x, y)) +  
 geom\_point() +  
 stat\_smooth(method = lm)



Plotting multiple graphs

par(mfrow=c(2,2))  
plot((fit))



### Linear Regression on Age and Height dataset

Load the required libraries

library(ggpubr)  
library(ggplot2)  
library(readxl)  
library(plotrix)

Upload the data

ageandheight = read\_excel("ageandheight.xls")

View the dataset

View(ageandheight)

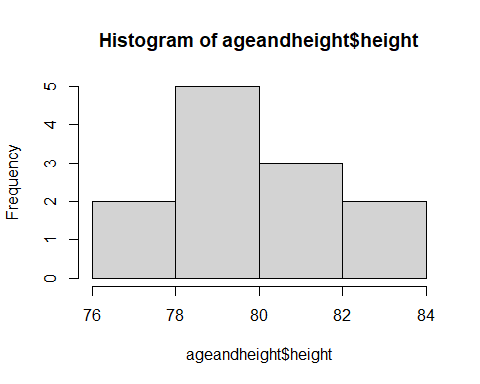
Regression - components: 1. independence of observation / autocorrelation

cor(ageandheight$age, ageandheight$height)

## [1] 0.9943661

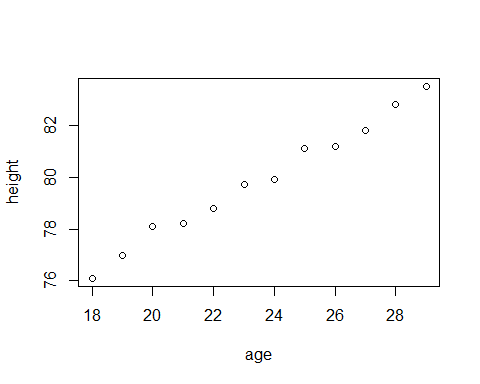
Regression - components: 2. normality / histogram

hist(ageandheight$height)



Regression - components: 3.linearity

plot(height~age, data = ageandheight)

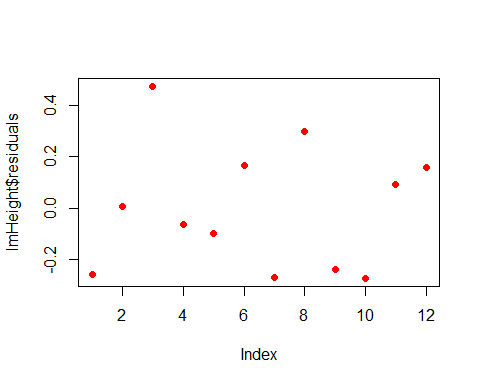


Regression - components: 4.Homoscedasticity / homogeneity of variance

lmHeight = lm(height~age, data = ageandheight)

for residual plots

plot(lmHeight$residuals, pch = 16, col = "red")



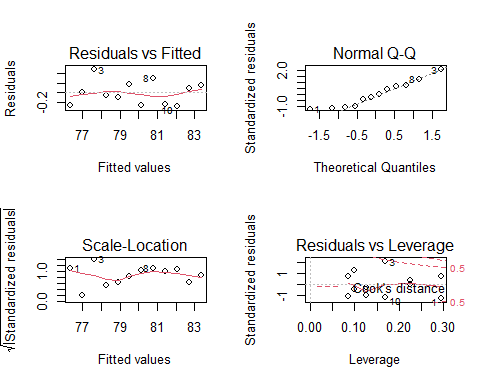
Review the results

summary(lmHeight)

##   
## Call:  
## lm(formula = height ~ age, data = ageandheight)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.27238 -0.24248 -0.02762 0.16014 0.47238   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 64.9283 0.5084 127.71 < 2e-16 \*\*\*  
## age 0.6350 0.0214 29.66 4.43e-11 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.256 on 10 degrees of freedom  
## Multiple R-squared: 0.9888, Adjusted R-squared: 0.9876   
## F-statistic: 880 on 1 and 10 DF, p-value: 4.428e-11

Regression - components: 4.Homoscedasticity / homogeneity of variance

par(mfrow=c(2,2))  
plot(lmHeight)

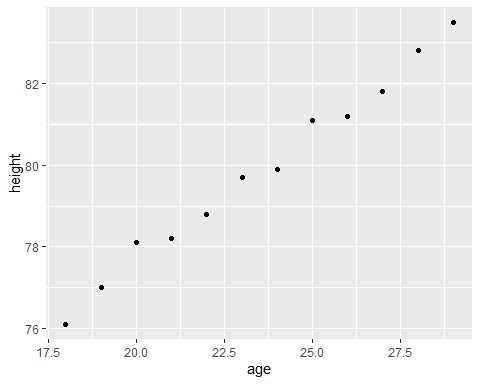


To go back to plotting one graph in the entire window

par(mfrow=c(1,1))

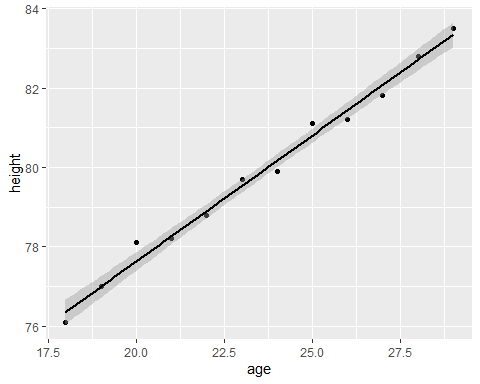
plot the data points on graph

lmHeight.graph = ggplot(ageandheight, aes(x=age,y=height))+ geom\_point()  
lmHeight.graph



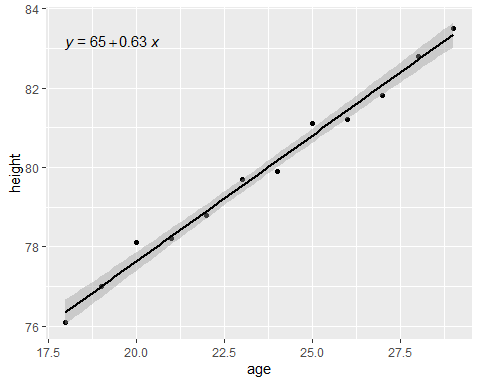
Add the linear regression line to the plotted data

lmHeight.graph = lmHeight.graph + geom\_smooth(method ="lm", col="black")  
lmHeight.graph



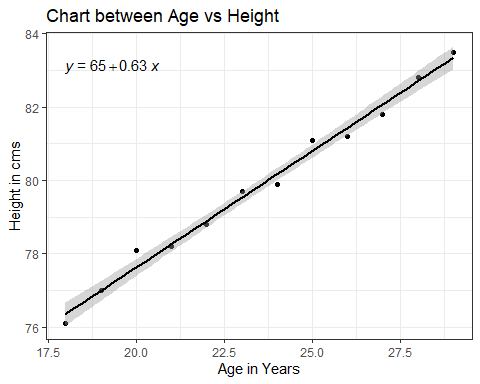
Add the equation for the regression line

lmHeight.graph = lmHeight.graph + stat\_regline\_equation()  
lmHeight.graph



Make the graph ready for publication

lmHeight.graph + theme\_bw() + labs(title = "Chart between Age vs Height", x = "Age in Years", y = "Height in cms")

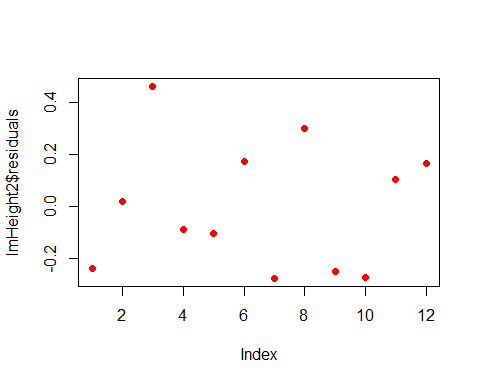


When a regression takes into account two or more predictors to create the linear regression, it’s called multiple linear regression. Height = a + Age × b1 + (Number of Siblings} × b2 Create a linear regression with two variables

lmHeight2 = lm(height~age + no\_siblings, data = ageandheight)

for residual plots

plot(lmHeight2$residuals, pch = 16, col = "red")



Review the results

summary(lmHeight2)

##   
## Call:  
## lm(formula = height ~ age + no\_siblings, data = ageandheight)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.2745 -0.2379 -0.0348 0.1676 0.4597   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 64.87033 0.65120 99.617 5.25e-15 \*\*\*  
## age 0.63631 0.02411 26.392 7.78e-10 \*\*\*  
## no\_siblings 0.01096 0.07010 0.156 0.879   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2694 on 9 degrees of freedom  
## Multiple R-squared: 0.9888, Adjusted R-squared: 0.9863   
## F-statistic: 397.1 on 2 and 9 DF, p-value: 1.669e-09

Detect Influential Points.

ageandheight[2, 2] = 7.7  
head(ageandheight)

## # A tibble: 6 x 3  
## age height no\_siblings  
## <dbl> <dbl> <dbl>  
## 1 18 76.1 1  
## 2 19 7.7 2  
## 3 20 78.1 4  
## 4 21 78.2 5  
## 5 22 78.8 3  
## 6 23 79.7 2

regression with outliers

lmHeight3 = lm(height~age, data = ageandheight)  
summary(lmHeight3)

##   
## Call:  
## lm(formula = height ~ age, data = ageandheight)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -53.704 -2.584 3.609 9.503 17.512   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 7.905 38.319 0.206 0.841  
## age 2.816 1.613 1.745 0.112  
##   
## Residual standard error: 19.29 on 10 degrees of freedom  
## Multiple R-squared: 0.2335, Adjusted R-squared: 0.1568   
## F-statistic: 3.046 on 1 and 10 DF, p-value: 0.1115

**SE**

library(plotrix)  
std.error(ageandheight$height,na.rm)

## [1] 6.065053

**RSS**

library(qpcR)  
RSS(lmHeight3)

## [1] 3721.847

**95% Confidence Interval**

new.dat <- data.frame(age=50)  
predict(lmHeight3, newdata = new.dat, interval = 'confidence')

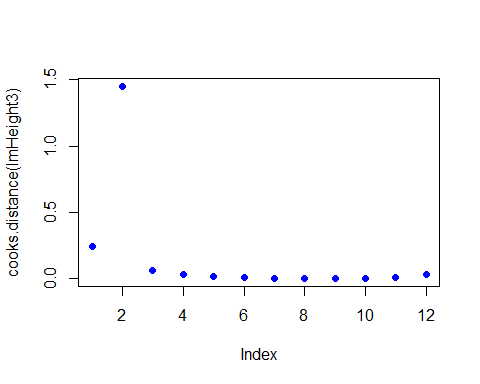
## fit lwr upr  
## 1 148.692 52.62959 244.7543

From the output, the fitted stopping height at an age of 50 is just above 148 cm. The confidence interval of (52.62959, 244.7543) signifies the range in which the true population parameter lies at a 95% level of confidence.

**RSE Statistic**  
From the summary of the fitted model, Residual Standard Error: 19.29 (on 10 degrees of freedom).

Plot the Cooks Distances

plot(cooks.distance(lmHeight3), pch = 16, col = "blue")



### Linear Regression on mtcars dataset

Load the required libraries

library(ggpubr)  
library(ggplot2)

sorting examples using mtcars dataset

attach(mtcars)  
View(mtcars)

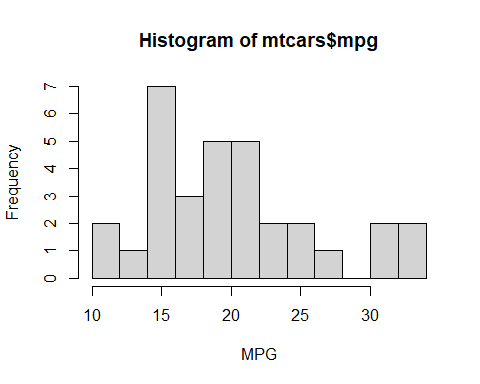
Pre-processing: Converting “am” into a categorical variable by assigning AT(Automatic Transmission) = 0, MT(Manual Transmission) = 1

mtData<-mtcars  
mtData$am <- as.factor(mtData$am)  
levels(mtData$am) <-c("AT", "MT")

Exploratory Analysis

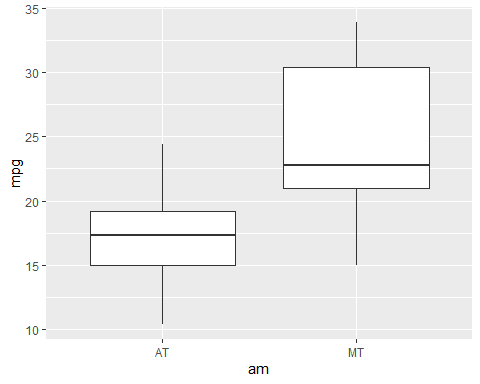
1. Histogram of Mpg

hist(mtcars$mpg,breaks = 10,xlab="MPG")



1. Boxplot of mpg and am

library(ggplot2)  
library(caret)  
ggplot(mtData, aes(x=am, y=mpg)) + geom\_boxplot()



Statistical Analysis

set.seed(12345)  
t.test(mtData$mpg~mtData$am)

##   
## Welch Two Sample t-test  
##   
## data: mtData$mpg by mtData$am  
## t = -3.7671, df = 18.332, p-value = 0.001374  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -11.280194 -3.209684  
## sample estimates:  
## mean in group AT mean in group MT   
## 17.14737 24.39231

Model Building

fit<-lm(mpg~as.numeric(am),data=mtData)  
summary(fit)

##   
## Call:  
## lm(formula = mpg ~ as.numeric(am), data = mtData)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -9.3923 -3.0923 -0.2974 3.2439 9.5077   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 9.902 2.628 3.768 0.000720 \*\*\*  
## as.numeric(am) 7.245 1.764 4.106 0.000285 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 4.902 on 30 degrees of freedom  
## Multiple R-squared: 0.3598, Adjusted R-squared: 0.3385   
## F-statistic: 16.86 on 1 and 30 DF, p-value: 0.000285

**SE**

library(plotrix)  
std.error(mpg,na.rm)

## [1] 1.065424

**RSS**

library(qpcR)  
RSS(fit)

## [1] 720.8966

**95% Confidence Interval**

new.dat <- data.frame(am=1)  
predict(fit, newdata = new.dat, interval = 'confidence')

## fit lwr upr  
## 1 17.14737 14.85062 19.44411

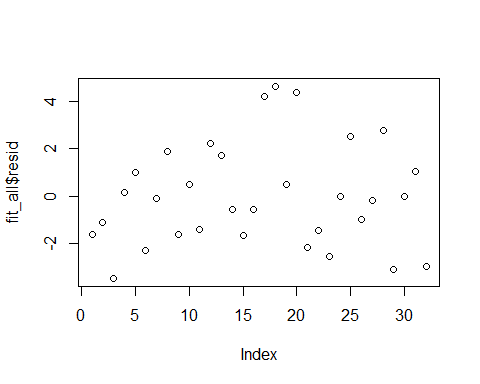
From the output, the fitted stopping mpg at am = 1 is just above 17. The confidence interval of (14.85062, 19.44411) signifies the range in which the true population parameter lies at a 95% level of confidence.

**RSE Statistic**  
From the summary of the fitted model, Residual Standard Error: 4.902 (on 30 degrees of freedom).

fit\_all<-lm(mpg~.,data=mtData)

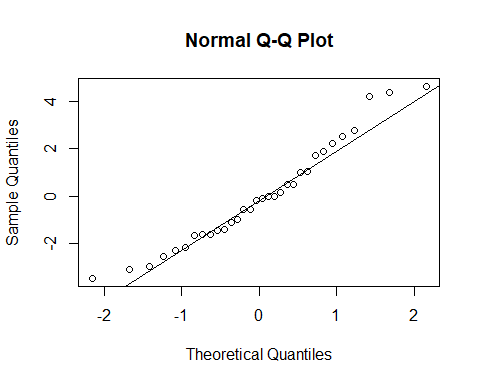
Residual Error Plot

plot(fit\_all$resid)



Residual plot

#hist(fit\_all$resid, main="Histogram of Residuals", ylab="Residuals")  
qqnorm(fit\_all$resid)   
qqline(fit\_all$resid)



**Conclusion:**  
Simple Linear & Multiple Linear Regression Analysis have been successfully performed on a sample data, ageandheight and mtcars dataset including SE, RSS, 95% Confidence Interval and RSE Statistic.