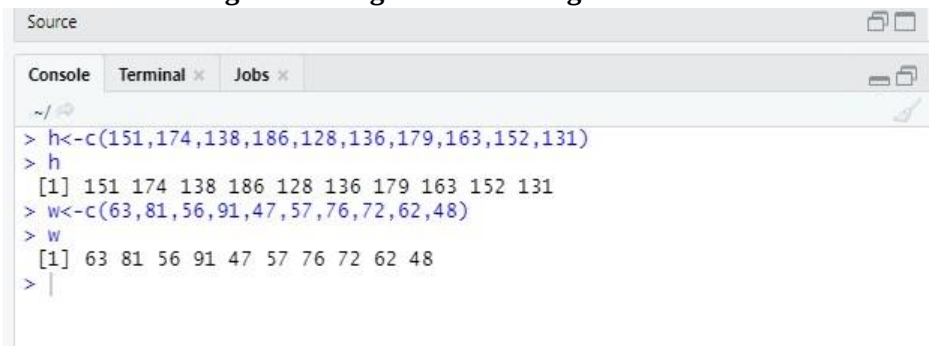
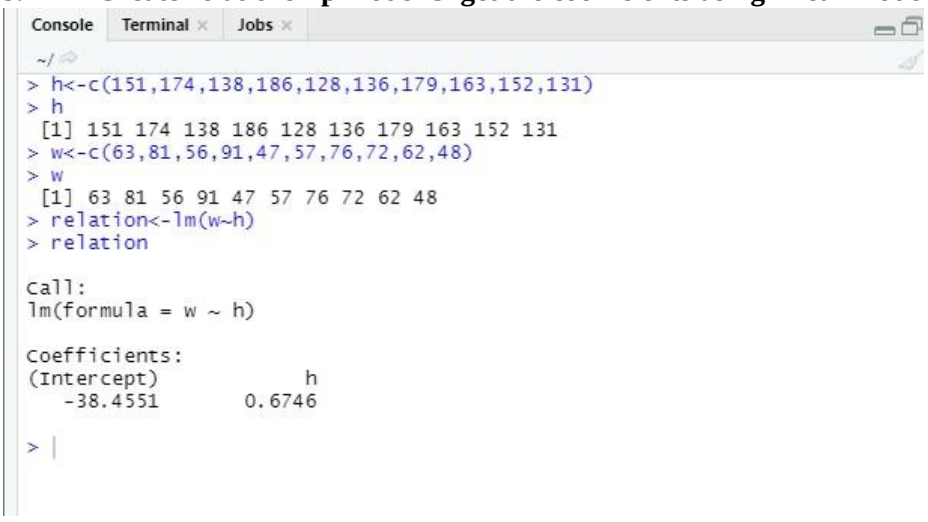
# Lab 7: Supervised Learning - Regression

1. **Below is the sample data representing the observations – #**
   **Values of height**
   **151, 174, 138, 186, 128, 136, 179, 163, 152, 131**
   **#Values of weight.**
   **63, 81, 56, 91, 47, 57, 76, 72, 62, 48**

**a. Create height and weight vectors using above values**




**b. Create relationship model & get the coefficients using linear model function of R (lm).**



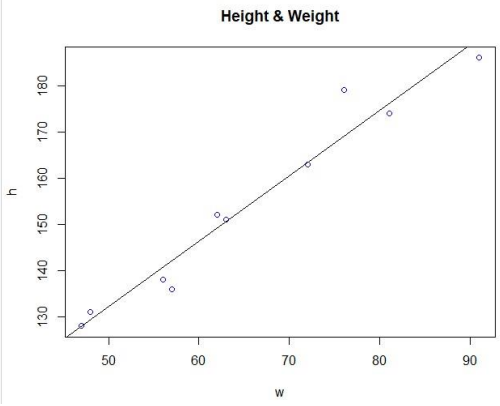
**c.Get the summary of the relationship and predict the weight of new persons whose height is 170.**

```
> h<-c(151,174,138,186,128,136,179,163,152,131)
> h
 [1] 151 174 138 186 128 136 179 163 152 131
> w<-c(63,81,56,91,47,57,76,72,62,48)
> w
 [1] 63 81 56 91 47 57 76 72 62 48
> relation<-lm(w~h)
> relation

Call:
lm(formula = w ~ h)

Coefficients:
(Intercept)            h
   -38.4551       0.6746

> summary(relation)

Call:
lm(formula = w ~ h)

Residuals:
    Min      1Q  Median      3Q     Max
-6.3002 -1.6629  0.0412  1.8944  3.9775

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -38.45509    8.04901  -4.778  0.00139 **
h             0.67461    0.05191  12.997 1.16e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.253 on 8 degrees of freedom
Multiple R-squared:  0.9548,    Adjusted R-squared:  0.9491
F-statistic: 168.9 on 1 and 8 DF,  p-value: 1.164e-06
```

```
> p<-data.frame(h=170)
> result<-predict(relation,p)
> result
       1
76.22869
>
```

**d.visualize the regression graphically.**

```
> plot(w,h,col="blue",main = "Height & Weight")
> abline(lm(h~w),xlab="weight in kg",ylab="height in cm")
>
```




## 2. Simple Linear regression

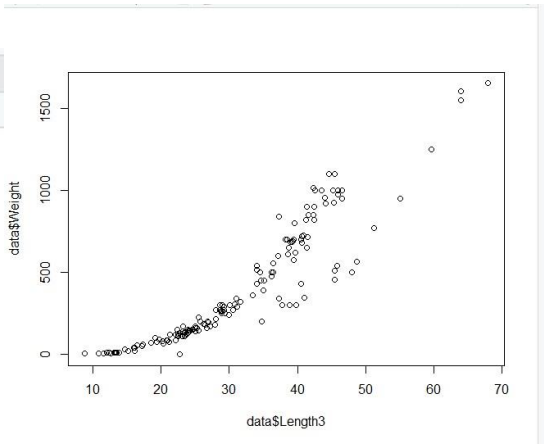
**Follow below step to implement Simple Linear regression on given database**

**a. Use the dataset Fish.csv for linear regression**

```
> data<-read.csv("D:/MCA_R/Fish.csv")
> data
      Species Weight Length1 Length2 Length3  Height  Width
1       Bream  242.0    23.2    25.4    30.0 11.5200 4.0200
2       Bream  290.0    24.0    26.3    31.2 12.4800 4.3056
3       Bream  340.0    23.9    26.5    31.1 12.3778 4.6961
4       Bream  363.0    26.3    29.0    33.5 12.7300 4.4555
5       Bream  430.0    26.5    29.0    34.0 12.4440 5.1340
6       Bream  450.0    26.8    29.7    34.7 13.6024 4.9274
7       Bream  500.0    26.8    29.7    34.5 14.1795 5.2785
8       Bream  390.0    27.6    30.0    35.0 12.6700 4.6900
9       Bream  450.0    27.6    30.0    35.1 14.0049 4.8438
10      Bream  500.0    28.5    30.7    36.2 14.2266 4.9594
11      Bream  475.0    28.4    31.0    36.2 14.2628 5.1042
12      Bream  500.0    28.7    31.0    36.2 14.3714 4.8146
13      Bream  500.0    29.1    31.5    36.4 13.7592 4.3680
14      Bream  340.0    29.5    32.0    37.3 13.9129 5.0728
15      Bream  600.0    29.4    32.0    37.2 14.9544 5.1708
```

**b. Plot the scatter graphs and check the relationship between Length3 and Weight columns of Fish dataset**

```
Console   Terminal ×   Jobs ×
~/
> plot(data$Length3, data$weight)
> lm(data$Length3~data$weight)

Call:
lm(formula = data$Length3 ~ data$weight)

Coefficients:
(Intercept)  data$weight
   19.30238      0.02994

>
```




**c. Randomize the dataset rows**

```
> data<-data[sample(nrow(data),),]
> head(data)
    Species Weight Length1 Length2 Length3  Height  Width
106   Perch  250.0    25.4    27.5    28.9  7.2828 4.5662
125   Perch 1000.0    39.8    43.0    45.2 11.9328 7.2772
54    Roach  272.0    25.0    27.0    30.6  8.5680 4.7736
159   Smelt   19.9    13.8    15.0    16.2  2.9322 1.8792
25    Bream  700.0    31.9    35.0    40.5 16.2405 5.5890
75    Perch   40.0    13.8    15.0    16.0  3.8240 2.4320
```

**d. Split the data set into Training Data set and Test Data set.**

```
> TrainData=data[1:111,]
> TestData=data[112:159,]
> TrainData
      Species Weight Length1 Length2 Length3  Height  Width
106     Perch  250.0    25.4    27.5    28.9  7.2828 4.5662
125     Perch 1000.0    39.8    43.0    45.2 11.9328 7.2772
54      Roach  272.0    25.0    27.0    30.6  8.5680 4.7736
159     Smelt   19.9    13.8    15.0    16.2  2.9322 1.8792
25      Bream  700.0    31.9    35.0    40.5 16.2405 5.5890
75      Perch   40.0    13.8    15.0    16.0  3.8240 2.4320
129      Pike  200.0    30.0    32.3    34.8  5.5680 3.3756
19      Bream  610.0    30.9    33.5    38.6 15.6330 5.1338
76      Perch   51.5    15.0    16.2    17.2  4.5924 2.6316
99      Perch  188.0    22.6    24.6    26.2  6.7334 4.1658
56  Whitefish  270.0    23.6    26.0    28.7  8.3804 4.2476
```

**e. Perform single linear regression analysis on training dataset columns Length3 as Y and Weight as X, using linear model function (lm).**

```
Console   Terminal   Jobs
~/
> rel<-lm(Length3~Weight,data = TrainData)
> summary(rel)

Call:
lm(formula = Length3 ~ Weight, data = TrainData)

Residuals:
     Min       1Q   Median       3Q      Max
-10.3466  -2.4865   0.0762   2.2089  12.9120

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 18.971630   0.621751   30.51   <2e-16 ***
Weight       0.029659   0.001138   26.07   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.407 on 109 degrees of freedom
Multiple R-squared:  0.8617,    Adjusted R-squared:  0.8605
F-statistic: 679.4 on 1 and 109 DF,  p-value: < 2.2e-16

>
```

**f. Predict the Length3 value using Testing dataset**

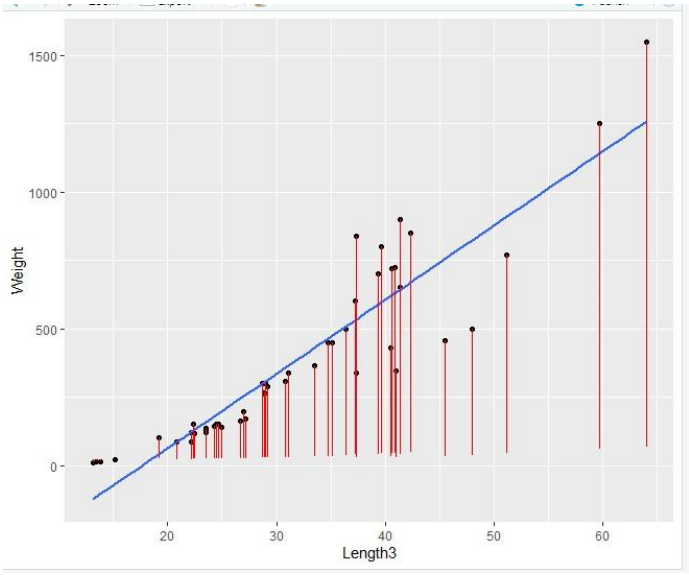
```
> pred<-predict(rel,newdata = TestData)
> pred
     120      133       60      138       78       27       53
44.18136 31.72479 42.69843 33.80088 21.93748 40.32575 27.57260
     103        4      135       88       44       95       89
27.86918 29.73767 32.49591 22.53065 23.42041 23.42041 22.82724
       6       58      144      142      104       40       47
32.31796 28.04713 64.94231 56.04476 26.68284 22.53065 23.12382
     140       13       90      117      152       16      155
41.80868 33.80088 22.97553 45.66428 19.26821 36.76673 19.33346
      66      101       26       84        3      115       39
23.42041 24.81436 40.47405 22.38236 29.05552 39.73258 21.55192
      50      157       81      134       72      118      112
23.74665 19.33346 21.49260 29.20381 27.86918 38.24966 43.88477
      14        9       49       45      105      158
29.05552 32.31796 23.98392 23.27211 26.83113 19.55590
>
```

**g. Analyze the Testing result using predicted and actual value of the Length3 column data and calculate correlation between them**

```
> data1<-data.frame(pred,TestData$Length3)
> data1
         pred TestData.Length3
120 44.18136             42.3
133 31.72479             40.5
60  42.69843             39.6
138 33.80088             48.0
78  21.93748             19.2
27  40.32575             40.6
53  27.57260             29.2
103 27.86918             28.7
4   29.73767             33.5
135 32.49591             45.5
88  22.53065             23.5
44  23.42041             24.7
95  23.42041             24.5
89  22.82724             23.5
6   32.31796             34.7
58  28.04713             30.8
144 64.94231             64.0
142 56.04476             59.7
104 26.68284             28.9
118 38.24966             41.4
112 43.88477             37.3
14  29.05552             37.3
9   32.31796             35.1
49  23.98392             27.2
45  23.27211             24.3
105 26.83113             28.9
158 19.55590             15.2
> cor(pred,TestData$Length3)
[1] 0.9162536
```

**h. Analyze the regression line with Residuals(line segment which represents the distance between y-value of the actual scatter plot points and the y values of the regression equation at those points) on a <u>scatter plot</u>**

```
Console  Terminal  Jobs
~/
> ggplot(rel, aes(Length3 , Weight))+ geom_point() +stat_smooth(method = lm, se = FALSE) + geom_segment(aes(xend = Length3, yend = .fitted),color = "red",size = 0.3)
`geom_smooth()` using formula 'y ~ x'
>
```



**3. Multiple Linear regression**

**Follow below step to implement Multiple Linear regression on given database**

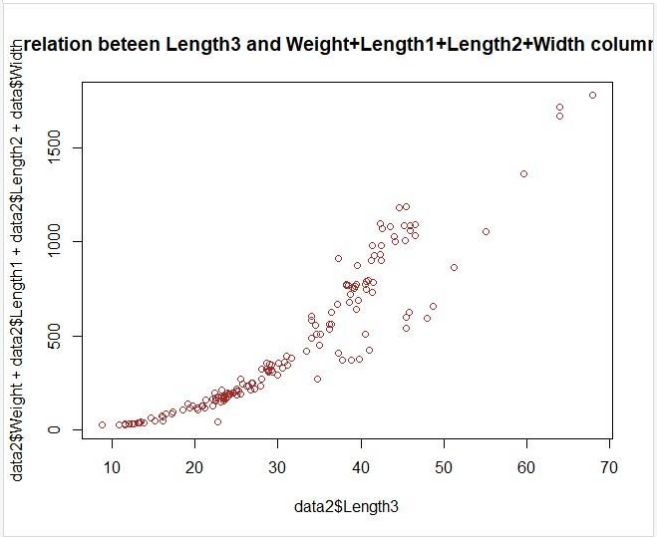
**a. Use the same training and testing dataset of Fish.csv created in exercise 2.**

```
Console   Terminal   Jobs
~/
> data2<-read.csv("D:\\MCA_R\\Fish.csv")
> data2
      Species Weight Length1 Length2 Length3  Height  Width
1       Bream  242.0    23.2    25.4    30.0 11.5200 4.0200
2       Bream  290.0    24.0    26.3    31.2 12.4800 4.3056
3       Bream  340.0    23.9    26.5    31.1 12.3778 4.6961
4       Bream  363.0    26.3    29.0    33.5 12.7300 4.4555
5       Bream  430.0    26.5    29.0    34.0 12.4440 5.1340
6       Bream  450.0    26.8    29.7    34.7 13.6024 4.9274
7       Bream  500.0    26.8    29.7    34.5 14.1795 5.2785
8       Bream  390.0    27.6    30.0    35.0 12.6700 4.6900
9       Bream  450.0    27.6    30.0    35.1 14.0049 4.8438
10      Bream  500.0    28.5    30.7    36.2 14.2266 4.9594
137      Pike  340.0    40.1    43.0    45.8 7.7860 5.1296
138      Pike  500.0    42.0    45.0    48.0  6.9600 4.8960
139      Pike  567.0    43.2    46.0    48.7  7.7920 4.8700
140      Pike  770.0    44.8    48.0    51.2  7.6800 5.3760
141      Pike  950.0    48.3    51.7    55.1  8.9262 6.1712
142      Pike 1250.0    52.0    56.0    59.7 10.6863 6.9849
 [ reached 'max' / getOption("max.print") -- omitted 17 rows ]
> TrainData<-data2[1:111,]
> TestData<-data2[112:159]
Error in `[.data.frame`(data2, 112:159) : undefined columns selected
> TestData<-data2[112:159,]
>
```

**b. Plot the scatter graphs and check the relationship between (Length3) and (Weight, Length1, Length2, Width) columns**

```
> relation<-lm(Length3~Weight+Length1+Length2+Width,data=data2)
> summary(relation)

Call:
lm(formula = Length3 ~ Weight + Length1 + Length2 + Width, data = data2)

Residuals:
     Min       1Q   Median       3Q      Max
-2.06066 -1.10806 -0.01334  0.73636  2.72262

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.908520   0.441878   2.056   0.0415 *
Weight       0.001476   0.000693   2.131   0.0347 *
Length1     -1.891323   0.310018  -6.101 8.19e-09 ***
Length2      2.809496   0.296775   9.467  < 2e-16 ***
Width       -0.104492   0.127804  -0.818   0.4149
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.108 on 154 degrees of freedom
Multiple R-squared:  0.9911,    Adjusted R-squared:  0.9909
F-statistic:  4300 on 4 and 154 DF,  p-value: < 2.2e-16

>
```

```
> TestData<-data2[112:159,]
>
> relation<-lm(Length3~Weight+Length1+Length2+Width,data = data2)
> summary(relation)

Call:
lm(formula = Length3 ~ Weight + Length1 + Length2 + Width, data = data2)

Residuals:
     Min       1Q   Median       3Q      Max
-2.06066 -1.10806 -0.01334  0.73636  2.72262

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.908520   0.441878   2.056   0.0415 *
Weight       0.001476   0.000693   2.131   0.0347 *
Length1     -1.891323   0.310018  -6.101 8.19e-09 ***
Length2      2.809496   0.296775   9.467  < 2e-16 ***
Width       -0.104492   0.127804  -0.818   0.4149
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.108 on 154 degrees of freedom
Multiple R-squared:  0.9911,    Adjusted R-squared:  0.9909
F-statistic:  4300 on 4 and 154 DF,  p-value: < 2.2e-16

> plot(data2$Length3,data2$Weight+data2$Length1+data2$Length2+data$Width,col="brown",main="relation beteen Length3 and Weight+Length1+Length2+Width colum
ns")
> |
```




**c. Perform multiple regression analysis on training dataset columns Length3 as Y and Weight, Length2, Length1, Width as X1, X2, X3, X4, using linear model function (lm).**

Console | Terminal | Jobs

~/

```
> multirel<-lm(Length3~Weight+Length1+Length2+Width,data = TrainData)
> summary(multirel)

Call:
lm(formula = Length3 ~ Weight + Length1 + Length2 + Width, data = TrainData)

Residuals:
    Min      1Q  Median      3Q     Max
-2.5563 -0.7389  0.2022  0.7034  1.9223

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.596584   0.800584   0.745 0.457809
Weight       0.004177   0.001094   3.817 0.000228 ***
Length1     -0.648346   0.390316  -1.661 0.099653 .
Length2      1.763292   0.381112   4.627 1.06e-05 ***
Width       -0.776338   0.330281  -2.351 0.020596 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.9701 on 106 degrees of freedom
Multiple R-squared:  0.9861,    Adjusted R-squared:  0.9856
F-statistic:  1886 on 4 and 106 DF,  p-value: < 2.2e-16

> |
```

**d. Predict the Length3 value using Testing dataset**

```
Console  Terminal  Jobs
~/
> pred1<-predict(multirel,newdata=TestData)
> pred1
     112      113      114      115      116      117      118      119
38.69708 40.44539 39.72205 41.53086 41.34507 43.64217 43.75499 43.35342
     120      121      122      123      124      125      126      127
45.23772 45.28966 45.58570 45.35259 48.63178 49.14133 49.25638 49.09226
     128      129      130      131      132      133      134      135
49.81443 36.31532 38.02102 38.96020 41.93730 42.82857 43.49642 48.15162
     136      137      138      139      140      141      142      143
48.27121 48.69271 51.00171 52.28703 55.23135 59.62082 65.42547 72.00000
     144      145      146      147      148      149      150      151
71.79115 75.22181 11.06192 11.75847 11.87688 12.21917 12.45755 12.55824
     152      153      154      155      156      157      158      159
13.12223 13.21334 13.51433 13.62182 13.94656 14.75043 15.73094 16.72302
>
```

**e. Analyze the Testing result using predicted and actual value of the Length3 column data and calculate correlation between them**

```
Console  Terminal  Jobs
~/
> df2<-data.frame(pred1,TestData$Length3)
> df2
       pred1 TestData.Length3
112 38.69708             37.3
113 40.44539             39.0
114 39.72205             38.3
115 41.53086             39.4
116 41.34507             39.3
117 43.64217             41.4
118 43.75499             41.4
119 43.35342             41.3
120 45.23772             42.3
121 45.28966             42.5
122 45.58570             42.4
123 45.35259             42.5
124 48.63178             44.6
125 49.14133             45.2
126 49.25638             45.5
```

**f. Analyze the regression line with Residuals(line segment which represents the distance between y-value of the actual scatter plot points and the y values of the regression equation at those points) on a <u>scatter plot</u>**

```
Console  Terminal  Jobs
~/
> ggplot(multirel, aes(Length3, Weight+Length1+Length2+Width))+ geom_point()+stat_smooth(m
ethod = lm, se = FALSE) +geom_segment(aes(xend=Length3,yend=.fitted),color="red",size=0.3)
`geom_smooth()` using formula 'y ~ x'
>
```