

PRACTICAL 9

Perform the Linear regression on the given data warehouse data.

Input Data

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

lm() Function :

This function creates the relationship model between the predictor and the response variable.

Syntax :

The basic syntax for lm() function in linear regression is –

lm(formula,data)

Following is the description of the parameters used :–

- formula is a symbol presenting the relation between x and y.
- data is the vector on which the formula will be applied.

A. Create Relationship Model & get the Coefficients

Values of height

```
x <- c(151, 174, 138, 186, 128, 136, 179, 163, 152, 131)
```

Values of width

```
y <- c(63, 81, 56, 91, 47, 57, 76, 72, 62, 48)
```

Apply the lm() function.

```
relation <- lm(y~x)
```

```
print(relation)
```

OUTPUT:

```
Call:
lm(formula = y ~ x)

Coefficients:
(Intercept)      x
   -38.4551      0.6746
```

B. Get the Summary of the Relationship

Values of height

```
x <- c(151, 174, 138, 186, 128, 136, 179, 163, 152, 131)
```

Values of width

```
y <- c(63, 81, 56, 91, 47, 57, 76, 72, 62, 48)
```

Apply the lm() function.

```
relation <- lm(y~x)
```

```
print(summary(relation))
```

OUTPUT:

```
Call:
lm(formula = y ~ x)

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 **
x             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
```

predict() Function

Syntax

The basic syntax for predict() in linear regression is –

```
predict(object, newdata)
```

Following is the description of the parameters used –

- **object** is the formula which is already created using the `lm()` function.
- **newdata** is the vector containing the new value for predictor variable.

C. Predict the weight of new persons

The predictor vector.

```
x <- c(151, 174, 138, 186, 128, 136, 179, 163, 152, 131)
```

The response vector.

```
y <- c(63, 81, 56, 91, 47, 57, 76, 72, 62, 48)
```

Apply the `lm()` function.

```
relation <- lm(y~x)
```

Find weight of a person with height 170.

```
a <- data.frame(x = 170)
```

```
result <- predict(relation,a)
```

```
print(result)
```

OUTPUT:

```
1  
76.22869
```

D. Visualize the Regression Graphically

Create the predictor and response variable.

```
x <- c(151, 174, 138, 186, 128, 136, 179, 163, 152, 131)
```

```
y <- c(63, 81, 56, 91, 47, 57, 76, 72, 62, 48)
```

```
relation <- lm(y~x)
```

```
# Give the chart file a name.
```

```
png(file = "linearregression.png")
```

```
# Plot the chart.
```

```
plot(y,x,col = "blue",main = "Height & Weight Regression",  
abline(lm(x~y)),cex = 1.3,pch = 16,xlab = "Weight in Kg",ylab = "Height in  
cm")
```

```
# Save the file.
```

```
dev.off()
```

```
null device  
1
```

```
# Plot the chart.
```

```
plot(y,x,col = "blue",main = "Height & Weight Regression",  
abline(lm(x~y)),cex = 1.3,pch = 16,xlab = "Weight in Kg",ylab = "Height in  
cm")
```

OUTPUT:

