Regressin

De Farmer

4/26/2021

Linear Regression

#Linear regressin is used topredict the value of an outcome variable y on the basis of one or more imput vaiables x.ie it is used toestablish linear relationship betweenresponse and predictor variable.

#y=mx+b

#Syntax

#lm(formula,data)

#**Formula**-is the symbol that represents relation between x and y.

#**Data**-is a vector on which we will apply the formula.

Creating Relationship model and Getting Coefficients

```
x <- c(141,134,178,156,108,116,119,143,162,130)
y <- c(62,85,56,21,47,17,76,92,62,58)
#applying lm() function
model <- lm(y~x)
model
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Coefficients:
## (Intercept)            x
##    47.50833      0.07276
```

```
#summary of the model
summary(model)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -38.948  -7.390   1.869  15.933  34.087
##
## Coefficients:
```

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 47.50833   55.18118   0.861    0.414
## x            0.07276    0.39342   0.185    0.858
##
## Residual standard error: 25.96 on 8 degrees of freedom
## Multiple R-squared:  0.004257,   Adjusted R-squared:  -0.1202
## F-statistic: 0.0342 on 1 and 8 DF,  p-value: 0.8579
```

The Predict() Function

#predict(object,newdata)

#**Object**-is the formula that we have already predicted using the lm() function

#**NewData**-is the vector that contains the new value for predictor variable.

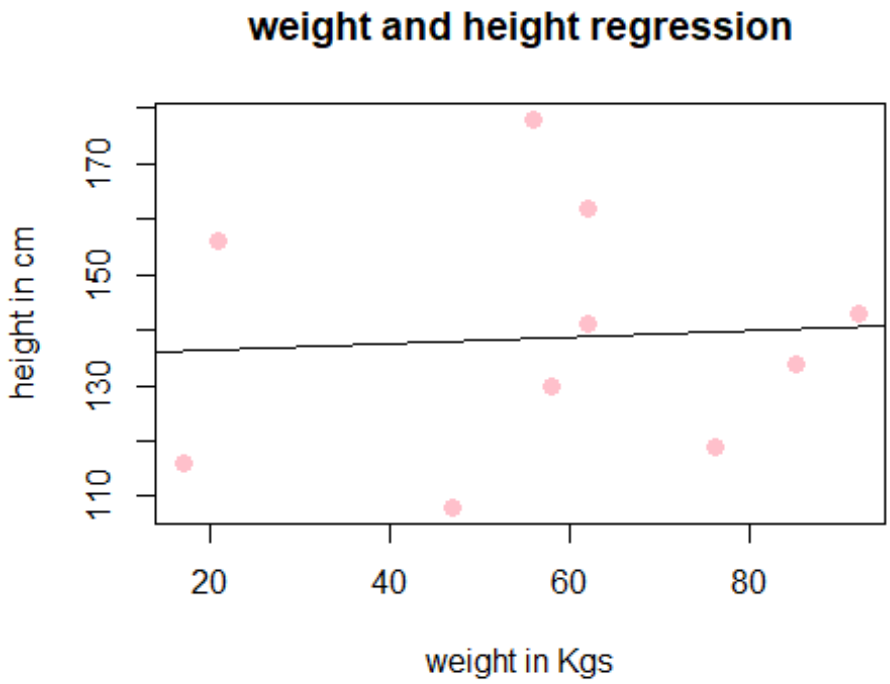
#Finding weight for person with height 170.

```
z <- data.frame(x=160)
predictr <-predict(model,z)
predictr
```

```
##        1
## 59.14977
```

#plotting Regression

```
plot(y,x,col="pink",main = "weight and height
regression",abline(lm(x~y)),cex=1.3,pch=16,xlab = "weight in Kgs",ylab =
"height in cm")
```



MULTIPLE LINEAR REGRESSION

#used to predict outcome variable (y) based on multiple distinct variables.

#y=b0+b1*x1+b2*x2+b3*x3.....bnxn

#The b value represents regression weight.They measure association between outcome and predictor variable.

#**Y**-response variable.

#**b0,b1,b2,b3...,bn**-coefficients

#**x1,x2,x3,....xn**-predictor vaiables

#Syntax

#lm(y~x1+x2+x3+.....xn,data)

```
data <-mtcars
data

##                    mpg cyl  disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4         21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag     21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710        22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive    21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8 360.0 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
```

```
## Duster 360          14.3   8 360.0 245 3.21 3.570 15.84  0  0    3    4
## Merc 240D           24.4   4 146.7  62 3.69 3.190 20.00  1  0    4    2
## Merc 230            22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2
## Merc 280            19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4
## Merc 280C           17.8   6 167.6 123 3.92 3.440 18.90  1  0    4    4
## Merc 450SE          16.4   8 275.8 180 3.07 4.070 17.40  0  0    3    3
## Merc 450SL          17.3   8 275.8 180 3.07 3.730 17.60  0  0    3    3
## Merc 450SLC         15.2   8 275.8 180 3.07 3.780 18.00  0  0    3    3
## Cadillac Fleetwood  10.4   8 472.0 205 2.93 5.250 17.98  0  0    3    4
## Lincoln Continental 10.4   8 460.0 215 3.00 5.424 17.82  0  0    3    4
## Chrysler Imperial   14.7   8 440.0 230 3.23 5.345 17.42  0  0    3    4
## Fiat 128            32.4   4  78.7  66 4.08 2.200 19.47  1  1    4    1
## Honda Civic         30.4   4  75.7  52 4.93 1.615 18.52  1  1    4    2
## Toyota Corolla      33.9   4  71.1  65 4.22 1.835 19.90  1  1    4    1
## Toyota Corona       21.5   4 120.1  97 3.70 2.465 20.01  1  0    3    1
## Dodge Challenger    15.5   8 318.0 150 2.76 3.520 16.87  0  0    3    2
## AMC Javelin         15.2   8 304.0 150 3.15 3.435 17.30  0  0    3    2
## Camaro Z28          13.3   8 350.0 245 3.73 3.840 15.41  0  0    3    4
## Pontiac Firebird    19.2   8 400.0 175 3.08 3.845 17.05  0  0    3    2
## Fiat X1-9           27.3   4  79.0  66 4.08 1.935 18.90  1  1    4    1
## Porsche 914-2       26.0   4 120.3  91 4.43 2.140 16.70  0  1    5    2
## Lotus Europa        30.4   4  95.1 113 3.77 1.513 16.90  1  1    5    2
## Ford Pantera L      15.8   8 351.0 264 4.22 3.170 14.50  0  1    5    4
## Ferrari Dino        19.7   6 145.0 175 3.62 2.770 15.50  0  1    5    6
## Maserati Bora       15.0   8 301.0 335 3.54 3.570 14.60  0  1    5    8
## Volvo 142E          21.4   4 121.0 109 4.11 2.780 18.60  1  1    4    2
```

```
head(data)
```

```
##                   mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

Creating relation model and finding coefficients

```
imput <- mtcars

model1 <- lm(mpg~wt+disp+hp,data=imput)
model1

##
## Call:
## lm(formula = mpg ~ wt + disp + hp, data = imput)
##
## Coefficients:
## (Intercept)           wt         disp           hp
##   37.105505    -3.800891    -0.000937    -0.031157
```

```r
b0 <- coef(model)[1]
b0
```

```
## (Intercept)
##    47.50833
```

```r
x_wt <- coef(model)[2]
x_wt
```

```
##          x
## 0.07275901
```

```r
x_disp <- coef(model)[3]
x_disp
```

```
## <NA>
##   NA
```

```r
x_hp <- coef(model)[4]
x_hp
```

```
## <NA>
##   NA
```