**Advanced Statistical Methods using R**

1. **Descriptive Statistics: (lungdata)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **lungcap** | **age** | **height** | **smoke** | **gender** | **caesarean** |
| 6.475 | 6 | 62.1 | no | male | no |
| 10.125 | 18 | 74.7 | yes | female | no |
| 9.55 | 16 | 69.7 | no | female | yes |
| 11.125 | 14 | 71 | no | male | no |
| 4.8 | 5 | 56.9 | no | male | no |
| 6.225 | 11 | 58.7 | no | female | no |
| 4.95 | 8 | 63.3 | no | male | yes |
| 5.667 | 12 | 53.6 | no | male | no |
| 14.145 | 14 | 67.4 | yes | female | no |
| 13.234 | 10 | 56.5 | yes | female | no |
| 13.25 | 6 | 72.5 | no | female | yes |
| 12.95 | 8 | 71 | no | female | yes |
| 13.56 | 15 | 65.4 | yes | male | yes |
| 23.22 | 19 | 54.9 | no | male | no |
| 14.234 | 15 | 56.7 | no | male | no |
| 12.7 | 24 | 56.8 | yes | male | no |
| 13.921 | 23 | 78.8 | no | male | yes |
| 13.342 | 12 | 67.7 | no | female | yes |
| 9.467 | 34 | 45.9 | yes | feamle | no |
| 8.567 | 5 | 50.9 | yes | female | yes |

> attach(lungdata)

> names(lungdata)

[1] "lungcap" "age" "height" "smoke" "gender" "caesarean"

> #ask for summeries for the lungdata

> summary(lungdata)

> > summary(lungcap)

> help(mean)

> ?mean

> table(smoke)

> table(smoke)/20(no of observations)

> table(smoke,gender)

> mean(lungcap)

> median(lungcap)

> var(lungcap)

> sd(lungcap)

> sqrt(var(lungcap))

> sd(lungcap)^2

> min(lungcap)

> max(lungcap)

> range(lungcap)

> quantile(lungcap,probs=0.90)

> quantile(lungcap,probs=c(0.20,0.40,0.90,1))

> sum(lungcap)

> sum(lungcap)/20

> cor(lungcap,age)

> cor(lungcap,age,method="pearson")

> cor(lungcap,age,method="spearman")

> var(lungcap,age)

**Skewness and Kurtosis**

**# packages to be installed (moments,normtest and goftest)**

> library(moments)

>Skweness(lungcap)

>kurtosis(lungcap)

> library(normtest)

>**Shapiro.test(lungcap)**

**>ad.test(lungcap)**

**>Lillie.test(lungcap)**

> install.packages("goftest")

**Normality Test**

> qqnorm(lungcap)

> qqline(lungcap,col=2,lwd=3)

**Uni-varite Analyis**

1. **One sample T-test**

|  |  |
| --- | --- |
| **Scale** | **One Variable** |
| **Metric** | **Mean, Median Mode, variance, SD**  **One sample T-test** |
| **Non-Metric** | **Count, Percentage, Mode** |

**Example: 1**

**x<-c(34,35,45,34,56,23,34)**

**Test Value = 40**

**Null hypothesis: H0:µ=40**

**Alternative Hypothesis: H1:µ≠40**

**?t.test**

**One sample t-test**

> t.test(x,mu=40)

> t.test(x,mu=40,alternative = c("two.sided"),conf.level=0.95)

> t.test(x,mu=40,alternative = c("greater"),conf.level=0.95)

> t.test(x,mu=40,alternative = c("less"),conf.level=0.95)

One Sample t-test

data: x

t = -0.6892, df = 6, p-value = 0.5164

alternative hypothesis: true mean is not equal to 40

95 percent confidence interval:

27.64904 46.92239

sample estimates:

mean of x

37.28571

**Null Hypothesis accepted as 0.5164>0.05**

**Example: 2:one\_t**

|  |
| --- |
| diameter |
| 24 |
| 35 |
| 34 |
| 45 |
| 34 |
| 23 |
| 45 |
| 34 |
| 34 |
| 45 |
| 45 |
| 67 |
| 45 |
| 34 |

**#import data**

> attach(one\_t)

> t.test(one\_t,mu=35)

> t.test(diameter,mu=35)

1. **Independent Sample t-test(Two sample t-test): Example 1**

> x1<-c(34,54,45,67,45,45,37)

>y1<-c(23,45,34,23,45,67,34)

> t.test(x1,y1,mu=0)

> t.test(x1,y1,mu=0,alternative=c("two.sided"),paired=F,var.equal = T,conf.level = 0.05)

Welch Two Sample t-test

data: x1 and y1

t = 1.119, df = 10.882, p-value = 0.2872

alternative hypothesis: true difference in means is not equal to 0

95 percent confidence interval:

-7.756765 23.756765

sample estimates:

mean of x mean of y

46.71429 38.71429

**Null hypothesis accepted as 0.2872>0.05**

**Example 2**

|  |  |
| --- | --- |
| Mumbai | Delhi |
| 2 | 3 |
| 3 | 4 |
| 3 | 5 |
| 4 | 6 |
| 5 | 5 |
| 4 | 5 |
| 4 | 5 |
| 5 | 4 |
| 3 | 3 |
| 4 | 3 |
| 5 | 5 |
| 4 | 6 |
| 3 | 6 |
| 3 | 6 |
| 4 | 5 |
| 3 | 3 |
| 4 | 4 |
| 5 | 5 |
| 4 | 6 |
| 4 | 5 |
| 5 | 5 |
| 3 | 5 |
| 4 | 4 |
| 5 | 3 |
| 4 | 3 |
| 3 | 5 |
| 3 | 6 |
| 4 | 6 |
| 5 | 6 |
| 3 | 5 |
| 4 | 3 |

**#Import Data**

> attach(independent\_t)

t.test(Mumbai,Delhi,mu=0)

> t.test(Mumbai,Delhi,mu=0,alternative=c("two.sided"),paired=F,var.equal = T,conf.level = 0.05)

1. **Paired t-test: Example -1**

**After1<-c(5,6,7,4,5,6,7,6,5,5)**

**Before1<-c(3,4,2,1,3,4,5,6,3,5)**

> t.test(Before1,After1,mu=0,paired=T)

> t.test(Before1,After1,mu=0,alternative=c("two.sided"),paired=T,conf.level = 0.95)

Paired t-test

data: Before1 and After1

t = -4.4721, df = 9, p-value = 0.00155

alternative hypothesis: true difference in means is not equal to 0

95 percent confidence interval:

-3.0116674 -0.9883326

sample estimates:

mean of the differences

-2

**Example : 2**

|  |  |
| --- | --- |
| Before | After |
| 2 | 3 |
| 3 | 4 |
| 3 | 5 |
| 4 | 6 |
| 5 | 5 |
| 4 | 5 |
| 4 | 5 |
| 5 | 4 |
| 3 | 3 |
| 4 | 3 |
| 5 | 5 |
| 4 | 6 |
| 3 | 6 |
| 3 | 6 |
| 4 | 5 |
| 3 | 3 |
| 4 | 4 |
| 5 | 5 |
| 4 | 6 |
| 4 | 5 |
| 5 | 5 |
| 3 | 5 |
| 4 | 4 |
| 5 | 3 |
| 4 | 3 |
| 3 | 5 |
| 3 | 6 |
| 4 | 6 |
| 5 | 6 |
| 3 | 5 |
| 4 | 3 |

**#Import data**

> attach(pair\_t)

> t.test(Bef,Aft,mu=0,alternative=c("two.sided"),paired=T,conf.level = 0.95)

Paired t-test

data: Bef and Aft

t = -3.5403, df = 30, p-value = 0.001327

alternative hypothesis: true difference in means is not equal to 0

95 percent confidence interval:

-1.3225261 -0.3548932

sample estimates:

mean of the differences

-0.8387097

1. **One way anova (One way Anova : Example 1**

> x1<-c(3,4,5,4,3,4,5)

> x2<-c(4,5,7,8,5,6,7)

> x3<-c(9,4,5,6,7,8,9)

> combined\_group<-data.frame(cbind(x1,x2,x3))

> stacked\_group<-stack(combined\_group)

|  |  |
| --- | --- |
| Values | ind |
| 3 | x1 |
| 4 | x1 |
| 4 | x1 |
| 4 | x1 |
| 4 | x2 |
| 4 | x3 |
| 5 | x1 |
| 5 | x1 |
| 5 | x2 |
| 5 | x2 |
| 5 | x3 |
| 6 | x2 |
| 6 | x3 |
| 7 | x2 |
| 7 | x2 |
| 7 | x3 |
| 8 | x2 |
| 8 | x3 |
| 9 | x3 |
| 9 | x3 |

> anova\_result<-aov(values~ind,data=stacked\_group)

> summary(anova\_result)

Df Sum Sq Mean Sq F value Pr(>F)

ind 2 30.10 15.048 6.971 0.00573 \*\*

Residuals 18 38.86 2.159

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

> View(stacked\_group)

> TukeyHSD(anova\_result)

Tukey multiple comparisons of means

95% family-wise confidence level

Fit: aov(formula = values ~ ind, data = stacked\_group)

$ind

diff lwr upr p adj

x2-x1 2.0000000 -0.00435014 4.004350 0.0505521

x3-x1 2.8571429 0.85279272 4.861493 0.0050831

x3-x2 0.8571429 -1.14720728 2.861493 0.5313653

**Example 2**

|  |  |
| --- | --- |
| **Place** | **Sales** |
| mumbai | 35 |
| mumbai | 30 |
| mumbai | 55 |
| mumbai | 65 |
| mumbai | 40 |
| mumbai | 20 |
| mumbai | 35 |
| mumbai | 33 |
| mumbai | 50 |
| delhi | 50 |
| delhi | 45 |
| delhi | 20 |
| delhi | 15 |
| delhi | 30 |
| delhi | 20 |
| delhi | 29 |
| delhi | 60 |
| delhi | 50 |
| kolkata | 10 |
| kolkata | 15 |
| kolkata | 30 |
| kolkata | 11 |
| kolkata | 15 |
| kolkata | 20 |
| kolkata | 60 |
| kolkata | 55 |
| kolkata | 25 |

**# import data**

> head(X1anova)

> summary(X1anova)

> str(X1anova)

> #as.factor()

> X1anova$Place<-as.factor(X1anova$Place)

> str(X1anova)

> anova1<-aov(Sales~Place,data=X1anova)

> summary(anova1)

Df Sum Sq Mean Sq F value Pr(>F)

Place 2 848 424.1 1.591 0.225

Residuals 24 6400 266.7

> TukeyHSD(anova1)

Tukey multiple comparisons of means

95% family-wise confidence level

Fit: aov(formula = Sales ~ Place, data = X1anova)

$Place

diff lwr upr p adj

kolkata-delhi -8.666667 -27.890458 10.55712 0.5079043

mumbai-delhi 4.888889 -14.334903 24.11268 0.8024026

mumbai-kolkata 13.555556 -5.668236 32.77935 0.2040539

> model.tables(anova1,"mean")

Tables of means

Grand mean

34.18519

Place

delhi kolkata mumbai

35.44 26.78 40.33

1. **Two Way Anova(2-Way Anova)**

|  |  |  |
| --- | --- | --- |
| Place | Education | Sales |
| 1 | 1 | 35 |
| 1 | 2 | 30 |
| 1 | 3 | 55 |
| 1 | 3 | 65 |
| 1 | 2 | 40 |
| 1 | 1 | 20 |
| 1 | 2 | 35 |
| 1 | 2 | 33 |
| 1 | 2 | 50 |
| 2 | 3 | 50 |
| 2 | 3 | 45 |
| 2 | 1 | 20 |
| 2 | 1 | 15 |
| 2 | 2 | 30 |
| 2 | 1 | 20 |
| 2 | 2 | 29 |
| 2 | 3 | 60 |
| 2 | 2 | 50 |
| 3 | 1 | 10 |
| 3 | 2 | 15 |
| 3 | 3 | 30 |
| 3 | 2 | 11 |
| 3 | 1 | 15 |
| 3 | 2 | 20 |
| 3 | 3 | 60 |
| 3 | 2 | 55 |
| 3 | 1 | 25 |

**# Import data**

> head(X2anova)

> summary(X2anova)

> str(X2anova)

**#as.factor()**

> X2anova$Place<-as.factor(X2anova$Place)

> X2anova$Education<-as.factor(X2anova$Education)

> anova1<-aov(Sales~Place+Education,data=X2anova)

> summary(anova1)

Df Sum Sq Mean Sq F value Pr(>F)

Place 2 848 424.1 3.454 0.0496 \*

Education 2 3698 1849.0 15.055 7.59e-05 \*\*\*

Residuals 22 2702 122.8

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

> TukeyHSD(anova1)

Tukey multiple comparisons of means

95% family-wise confidence level

Fit: aov(formula = Sales ~ Place + Education, data = X2anova)

$Place

diff lwr upr p adj

2-1 -4.888889 -18.01225 8.2344715 0.6239667

3-1 -13.555556 -26.67892 -0.4321952 0.0420625

3-2 -8.666667 -21.79003 4.4566937 0.2431079

$Education

diff lwr upr p adj

2-1 11.99074 -0.7158983 24.69738 0.0667549

3-1 31.19444 16.7864700 45.60242 0.0000527

3-2 19.20370 5.9636889 32.44372 0.0039312

**Interaction Effect**

> anova2<-aov(Sales~Place+Education+Place:Education,data=X2anova)

> summary(anova2)

Df Sum Sq Mean Sq F value Pr(>F)

Place 2 848 424.1 2.926 0.079374 .

Education 2 3698 1849.0 12.756 0.000355 \*\*\*

Place:Education 4 93 23.2 0.160 0.955850

Residuals 18 2609 145.0

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

> TukeyHSD(anova2)

Tukey multiple comparisons of means

95% family-wise confidence level

Fit: aov(formula = Sales ~ Place + Education + Place:Education, data = X2anova)

$Place

diff lwr upr p adj

2-1 -4.888889 -19.37369 9.5959168 0.6707068

3-1 -13.555556 -28.04036 0.9292501 0.0688642

3-2 -8.666667 -23.15147 5.8181390 0.3022629

$Education

diff lwr upr p adj

2-1 11.99074 -2.034112 26.01559 0.1015978

3-1 31.19444 15.291756 47.09713 0.0002577

3-2 19.20370 4.590142 33.81727 0.0094147

> model.tables(anova2,"mean")

Tables of means

Grand mean

34.18519

Place

1 2 3

40.33 35.44 26.78

rep 9.00 9.00 9.00

Education

1 2 3

20.77 32.76 51.96

rep 8.00 12.00 7.00

Place:Education

Education

Place 1 2 3

1 27.50 37.60 60.00

rep 2.00 5.00 2.00

2 18.33 36.33 51.67

rep 3.00 3.00 3.00

3 16.67 25.25 45.00

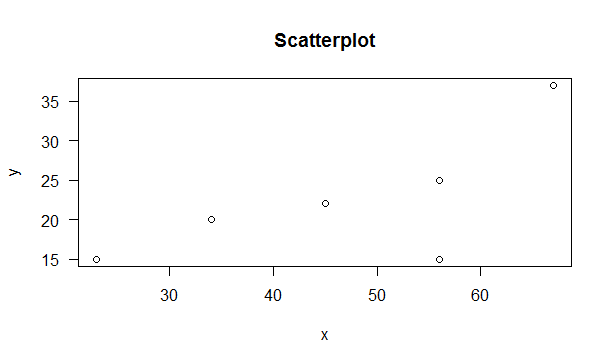
rep 3.00 4.00 2.00

1. **Correlation Analysis: Example-1**

> x<-c(23,34,45,56,56,67)

> y<-c(15,20,22,15,25,37)

plot(x,y,main="Scatterplot",las=1)



> cor(x,y,method="pearson")

[1] 0.6913044

> cor(x,y,method="spearman")

[1] 0.6617647

> cor(x,y,method="kendall")

[1] 0.6428571

> cor.test(x,y,method="pearson")

Pearson's product-moment correlation

data: x and y

t = 1.9135, df = 4, p-value = 0.1282

alternative hypothesis: true correlation is not equal to 0

95 percent confidence interval:

-0.2739560 0.9627362

sample estimates:

cor

0.6913044

> cor.test(x,y,method="pearson",alt="greater",conf.level = 0.99)

Pearson's product-moment correlation

data: x and y

t = 1.9135, df = 4, p-value = 0.06412

alternative hypothesis: true correlation is greater than 0

99 percent confidence interval:

-0.4563312 1.0000000

sample estimates:

cor

0.6913044

> cov(x,y)

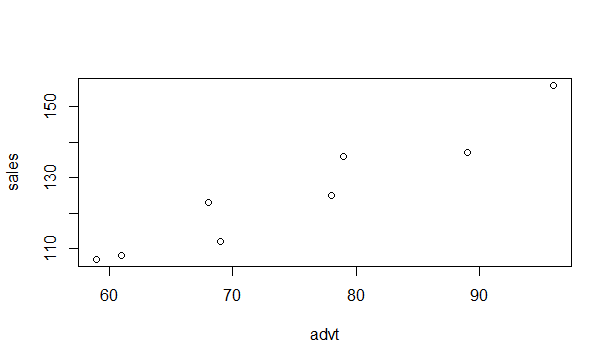
[1] 91.66667

**Example 2: File name (CA)**

|  |  |
| --- | --- |
| advt | sales |
| 78 | 125 |
| 89 | 137 |
| 96 | 156 |
| 69 | 112 |
| 59 | 107 |
| 79 | 136 |
| 68 | 123 |
| 61 | 108 |
| 89 | 137 |
| 96 | 156 |
| 69 | 112 |
| 59 | 107 |
| 79 | 136 |
| 68 | 123 |
| 61 | 108 |

> attach(CA)

> plot(advt,sales,main="Scatter",las=1)



> cor(advt,sales,method="pearson")

[1] 0.9570193

> cor.test(advt,sales,method="pearson",conf.level = 0.99)

Pearson's product-moment correlation

data: advt and sales

t = 11.898, df = 13, p-value = 2.316e-08

alternative hypothesis: true correlation is not equal to 0

99 percent confidence interval:

0.8228681 0.9901214

sample estimates:

cor

0.9570193

> cov(advt,sales)

[1] 211.7619

Example 3

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **lungcap** | **age** | **height** | **smoke** | **gender** | **caesarean** |
| 6.475 | 6 | 62.1 | no | male | no |
| 10.125 | 18 | 74.7 | yes | female | no |
| 9.55 | 16 | 69.7 | no | female | yes |
| 11.125 | 14 | 71 | no | male | no |
| 4.8 | 5 | 56.9 | no | male | no |
| 6.225 | 11 | 58.7 | no | female | no |
| 4.95 | 8 | 63.3 | no | male | yes |
| 5.667 | 12 | 53.6 | no | male | no |
| 14.145 | 14 | 67.4 | yes | female | no |
| 13.234 | 10 | 56.5 | yes | female | no |
| 13.25 | 6 | 72.5 | no | female | yes |
| 12.95 | 8 | 71 | no | female | yes |
| 13.56 | 15 | 65.4 | yes | male | yes |
| 23.22 | 19 | 54.9 | no | male | no |
| 14.234 | 15 | 56.7 | no | male | no |
| 12.7 | 24 | 56.8 | yes | male | no |
| 13.921 | 23 | 78.8 | no | male | yes |
| 13.342 | 12 | 67.7 | no | female | yes |
| 9.467 | 34 | 45.9 | yes | feamle | no |
| 8.567 | 5 | 50.9 | yes | female | yes |

> cor(lungdata[,1:3])

lungcap age height

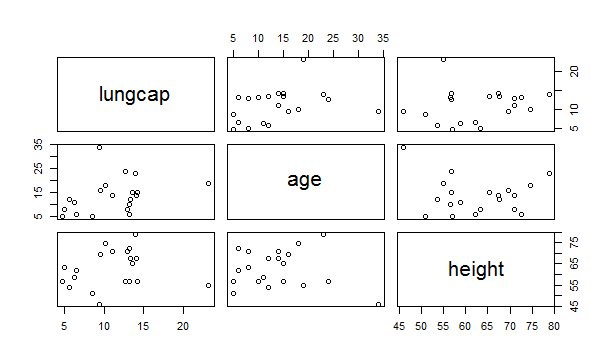
lungcap 1.0000000 0.3363845 0.1607966

age 0.3363845 1.0000000 -0.1257641

height 0.1607966 -0.1257641 1.0000000

> plot(lungdata)

> pairs(lungdata[,1:3])



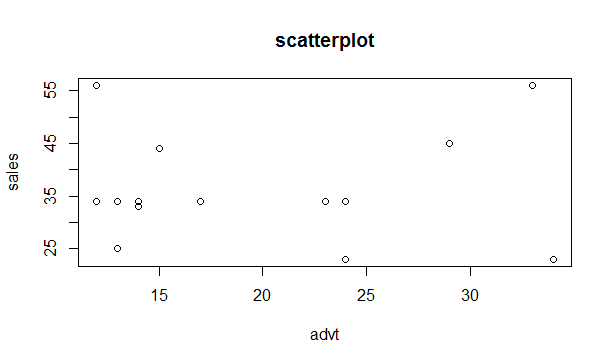
1. **Simple Regression**

|  |  |
| --- | --- |
| advt | sales |
| 15 | 44 |
| 13 | 34 |
| 13 | 25 |
| 12 | 34 |
| 12 | 56 |
| 17 | 34 |
| 24 | 23 |
| 29 | 45 |
| 33 | 56 |
| 23 | 34 |
| 23 | 34 |
| 24 | 34 |
| 14 | 33 |
| 23 | 34 |
| 14 | 34 |
| 34 | 23 |

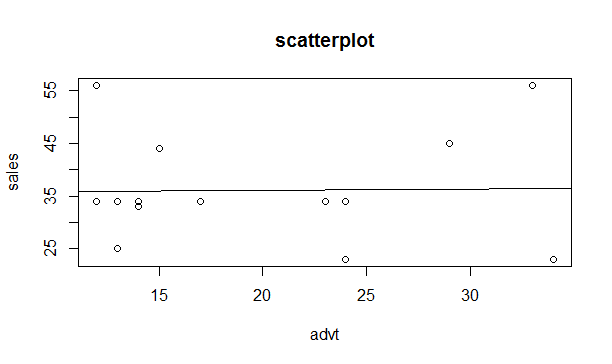
**# import data**

> attach(SR)

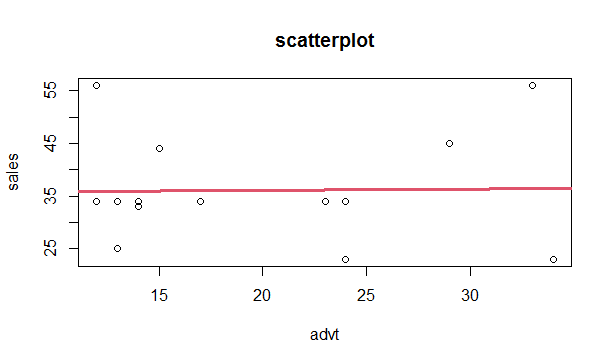
> plot(advt,sales,main="scatterplot")

****

> abline(mod)

****

> abline(mod,col=2,lwd=3)

****

> cor(advt,sales)

[1] 0.02166026

> help(lm)

> mod<-lm(sales~advt)

> summary(mod)

Call

lm(formula = sales ~ advt)

Residuals:

Min 1Q Median 3Q Max

-13.4557 -2.3499 -2.0572 0.6492 20.1706

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 35.48778 7.53065 4.712 0.000333 \*\*\*

advt 0.02847 0.35119 0.081 0.936538

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 10.16 on 14 degrees of freedom

Multiple R-squared: 0.0004692, Adjusted R-squared: -0.07093

F-statistic: 0.006571 on 1 and 14 DF, p-value: 0.9365

> attributes(mod)

$names

[1] "coefficients" "residuals" "effects" "rank" "fitted.values"

[6] "assign" "qr" "df.residual" "xlevels" "call"

[11] "terms" "model"

$class

[1] "lm"

> mod$coefficients

(Intercept) advt

35.48778301 0.02846895

> mod$coef

(Intercept) advt

35.48778301 0.02846895

> coef(mod)

(Intercept) advt

35.48778301 0.02846895

> anova(mod)

Analysis of Variance Table

Response: sales

Df Sum Sq Mean Sq F value Pr(>F)

advt 1 0.68 0.678 0.0066 0.9365

Residuals 14 1444.26 103.161

> abline(mod)

> plot(mod)

Hit <Return> to see next plot:

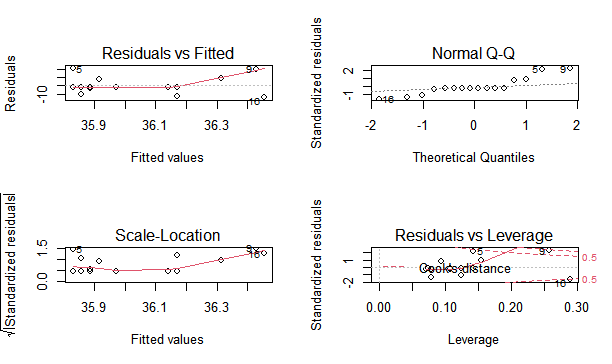
Hit <Return> to see next plot:

Hit <Return> to see next plot:

Hit <Return> to see next plot:

> par(mfrow=c(2,2))

> plot(mod)



1. **Multiple Regression : Example : File name: MR1**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attitude | Awarness | Perception | Cost | Rating | Buying |
| 4 | 4.3 | 3.9 | 3.17 | 4 | 4.62 |
| 5 | 5 | 4.79 | 4.67 | 4.7 | 5.15 |
| 5.14 | 4 | 5 | 5.15 | 5 | 6.08 |
| 3.9 | 4 | 3.9 | 3.83 | 3.2 | 4.31 |
| 4.12 | 4.34 | 3.5 | 3.83 | 3.45 | 4.46 |
| 5 | 4.5 | 5.1 | 2.33 | 4.54 | 5.38 |
| 4.11 | 4.2 | 3.5 | 3.76 | 3.54 | 4.77 |
| 5.11 | 5.67 | 5.15 | 5.15 | 5.15 | 6 |
| 5.23 | 4.78 | 5 | 3.33 | 4.6 | 5.08 |
| 3.95 | 3.67 | 3.67 | 3.33 | 3.2 | 4.62 |
| 3.98 | 4.17 | 4 | 3.17 | 3.67 | 4.23 |
| 4.2 | 4.15 | 4.5 | 4.7 | 4.3 | 5 |
| 5.11 | 5.2 | 4.5 | 4.33 | 4.8 | 5.69 |
| 4.23 | 4.1 | 3.23 | 3.67 | 3.9 | 4.54 |
| 5.56 | 6 | 5.15 | 3.17 | 5.15 | 6.31 |
| 4.15 | 3.07 | 3.57 | 3 | 3.24 | 4.08 |
| 2.67 | 3.67 | 4.12 | 3.6 | 4.2 | 4.69 |
| 3.67 | 3.5 | 3.8 | 4 | 4 | 4 |
| 5.1 | 4.9 | 4.68 | 4.67 | 4.79 | 5.54 |
| 3.86 | 3.83 | 3.56 | 3.83 | 3.34 | 4.7 |
| 3.5 | 3.9 | 4.2 | 4.33 | 3.9 | 4.87 |
| 4.65 | 5.33 | 5.1 | 4.67 | 4.84 | 5.69 |
| 4 | 3.9 | 4.15 | 3.83 | 4.1 | 4.69 |
| 4 | 4.1 | 4.5 | 3.97 | 3.56 | 4.92 |
| 4.15 | 4.9 | 4.67 | 3.83 | 4.67 | 5.38 |
| 3.65 | 3.54 | 4.1 | 3.56 | 4.2 | 4.38 |
| 4.9 | 4.89 | 5.2 | 4.25 | 4.78 | 5.69 |
| 4.11 | 3.64 | 4.2 | 4.45 | 4.46 | 4.62 |
| 3.98 | 3.9 | 3.9 | 3.25 | 3.8 | 4.23 |
| 4.8 | 5 | 4.67 | 3.83 | 4.98 | 5.85 |
| 4.3 | 4.8 | 4.8 | 4.33 | 4.57 | 5.15 |
| 5 | 5 | 4.79 | 4.15 | 4.89 | 5.38 |
| 4.89 | 4.67 | 4.95 | 4.5 | 4.56 | 5.15 |
| 4.23 | 4.22 | 4.3 | 3.33 | 3.56 | 4.92 |
| 4.24 | 4.25 | 4.5 | 3.67 | 4.78 | 5 |
| 3.15 | 4 | 4.32 | 3.5 | 3.78 | 4.69 |
| 5.22 | 3.95 | 4.5 | 4.33 | 4.2 | 4.54 |
| 4.2 | 4.67 | 5.1 | 4.78 | 4.86 | 5.31 |
| 4.15 | 4.78 | 4.95 | 4.1 | 4.8 | 5.54 |
| 3.78 | 4.05 | 3.98 | 3.84 | 3.8 | 4.77 |
| 3.67 | 4.25 | 4.24 | 3.5 | 3.9 | 4.92 |
| 3.45 | 3.58 | 4.4 | 4.33 | 4 | 4.92 |
| 1.2 | 1.5 | 2 | 2 | 2 | 2.23 |
| 4.34 | 3.78 | 4.7 | 3.45 | 4.68 | 5.08 |
| 5.11 | 5.6 | 5.56 | 5.1 | 5.2 | 6 |
| 4.65 | 4.5 | 4.57 | 4 | 4.78 | 5.08 |
| 4.35 | 4.98 | 4.7 | 4.17 | 4.98 | 5.46 |
| 2.85 | 2.79 | 2.8 | 2.95 | 3 | 3.85 |
| 3.45 | 3.67 | 3.68 | 3.67 | 3.56 | 4.15 |
| 4.68 | 5 | 4.5 | 4.68 | 4.5 | 5 |
| 2.31 | 3.23 | 3 | 2.7 | 3 | 3.23 |
| 4.22 | 4 | 5.23 | 5.45 | 5.4 | 6.08 |
| 3.26 | 4.14 | 4.34 | 3.67 | 4.6 | 4.69 |
| 4.9 | 5 | 4.9 | 4.5 | 4.5 | 5.31 |
| 1 | 1.83 | 1 | 1 | 1 | 1.85 |
| 4.24 | 5 | 4.1 | 3.67 | 4.34 | 5.08 |
| 3.6 | 4.33 | 4.2 | 3.67 | 3.9 | 4.77 |
| 3.9 | 3.67 | 4 | 3.24 | 4 | 4.38 |
| 3.98 | 4.18 | 4 | 3.8 | 4.35 | 4.77 |
| 3.78 | 4.48 | 4.3 | 4.17 | 4.68 | 4.92 |
| 4.11 | 4.69 | 4.1 | 3.45 | 4.4 | 4.69 |
| 4.15 | 2 | 2 | 1.7 | 1.8 | 2.15 |
| 5.22 | 4 | 3.9 | 3.27 | 3.9 | 4.77 |
| 3.45 | 3.5 | 4 | 3.5 | 4.4 | 4.85 |
| 1.5 | 2 | 1.8 | 2 | 2 | 2 |
| 4 | 4.4 | 3.78 | 4.33 | 4.9 | 5.46 |
| 1.89 | 2 | 1.5 | 1.98 | 1.8 | 2.31 |
| 1.36 | 2 | 2 | 2 | 2 | 2.77 |
| 4.45 | 4.83 | 4.83 | 4.25 | 4.35 | 5.69 |
| 4.78 | 4.83 | 4.7 | 4.17 | 4.78 | 5.38 |
| 4.46 | 4.45 | 4.9 | 4.83 | 4.16 | 5.23 |
| 2.19 | 2.67 | 2.3 | 2.5 | 2.8 | 3 |
| 4.9 | 5.17 | 5.15 | 4.67 | 5.3 | 5.69 |
| 4.33 | 3.83 | 3.98 | 3.17 | 3.6 | 4.38 |
| 4.98 | 4.67 | 4.78 | 4.76 | 4.95 | 5.92 |
| 2.34 | 3.23 | 3 | 2.58 | 3.2 | 3.62 |
| 4.6 | 5 | 4.8 | 4.5 | 4.5 | 5.46 |
| 2.9 | 3.85 | 3 | 3.76 | 3 | 3.85 |
| 4.32 | 4 | 4.5 | 4.67 | 4.59 | 5.15 |
| 2 | 2.12 | 2 | 2.33 | 2 | 2.54 |
| 2.98 | 2.3 | 3 | 2.76 | 3.2 | 3.77 |
| 1.9 | 2 | 2 | 1.98 | 2.4 | 2.92 |
| 2.78 | 2.33 | 3 | 3 | 3 | 3.23 |
| 4.23 | 5 | 4.5 | 4.57 | 4.8 | 5.23 |
| 4.5 | 5 | 4.83 | 4.89 | 4.9 | 5.31 |
| 3.9 | 4.83 | 4 | 3.67 | 3.67 | 4.85 |
| 5 | 5.17 | 5 | 5.35 | 5.3 | 6.15 |
| 4.89 | 5.17 | 4.9 | 4.67 | 4.8 | 5.85 |
| 1 | 1 | 1 | 1.95 | 1 | 1.92 |
| 1.8 | 2 | 2 | 2.15 | 2 | 2.85 |
| 4.44 | 3.5 | 4 | 3.35 | 4 | 4.85 |
| 2.9 | 3 | 3 | 3.34 | 3.4 | 3.92 |
| 4.4 | 4.45 | 4.65 | 4.55 | 4.5 | 5.23 |
| 4.34 | 4 | 3.9 | 3.83 | 4 | 4.31 |
| 4.1 | 4.12 | 4 | 3.59 | 4.2 | 4.54 |
| 4.58 | 4.67 | 4.3 | 4.67 | 4.75 | 5.08 |
| 4.9 | 4.5 | 4.67 | 4.54 | 4.4 | 5 |
| 1.9 | 2.2 | 2 | 2.17 | 2 | 2.38 |
| 3.56 | 4 | 3.78 | 4.15 | 4 | 4.69 |
| 3.97 | 4.3 | 4 | 1.83 | 4.23 | 4.92 |
| 4 | 4 | 3.45 | 2.17 | 4.2 | 4.69 |
| 3.98 | 4.15 | 4.15 | 3.17 | 3.78 | 4.92 |
| 1.9 | 2 | 2 | 1.33 | 2 | 2.62 |
| 3.9 | 4.33 | 4.8 | 2.5 | 4.6 | 5.38 |
| 4.12 | 4.5 | 4.79 | 2.33 | 4.45 | 5.31 |
| 3.98 | 4.2 | 4 | 3.17 | 3.56 | 4.69 |
| 4.11 | 4.45 | 4.9 | 4.83 | 4.89 | 5.38 |
| 4.11 | 4.33 | 4.35 | 3.83 | 3.8 | 4.85 |
| 3.9 | 3.67 | 4.44 | 3.83 | 4.1 | 4.92 |
| 1.9 | 2 | 2 | 4.5 | 2 | 2.23 |
| 4.1 | 4.15 | 4.5 | 3 | 4.78 | 5 |
| 4.2 | 4.5 | 4 | 2.33 | 4.1 | 5 |
| 3.98 | 4.1 | 4 | 3.87 | 4.2 | 4.77 |
| 1.9 | 2 | 1.5 | 4.5 | 3.4 | 2.08 |
| 4.5 | 4.78 | 5 | 3.5 | 3.4 | 5.38 |
| 4 | 3.67 | 3.9 | 3.5 | 4.56 | 4.54 |
| 2.67 | 3.5 | 3.23 | 2.17 | 4.2 | 4 |
| 3.67 | 4 | 4.3 | 3.67 | 4.6 | 4.85 |
| 5.1 | 5.5 | 5.54 | 2 | 5.2 | 6.08 |
| 2.5 | 3 | 3 | 3.5 | 3 | 3.5 |
| 4 | 4 | 4.15 | 4.5 | 3.78 | 4.62 |
| 5 | 4.8 | 4.68 | 4.33 | 4.9 | 5.15 |
| 5.14 | 5.35 | 5.3 | 3.83 | 5.8 | 6.08 |
| 3.9 | 4 | 3.9 | 4.17 | 3.45 | 4.31 |
| 4.12 | 4 | 4.1 | 4.17 | 4.2 | 4.46 |
| 5 | 4.56 | 4.34 | 1.67 | 4.2 | 5.38 |
| 4.11 | 4 | 4.35 | 2 | 2.2 | 4.77 |
| 5.11 | 5.5 | 5.3 | 3.17 | 5.15 | 6 |
| 5.23 | 4.45 | 4.45 | 3.67 | 4.2 | 5.08 |
| 3.95 | 4 | 5.2 | 2.17 | 4 | 4.62 |
| 3.98 | 3.5 | 4 | 2.17 | 3.8 | 4.23 |
| 4.2 | 4.67 | 4.7 | 3.85 | 4.5 | 5 |
| 5.11 | 4.83 | 5 | 4 | 5.2 | 5.69 |
| 4.23 | 3.9 | 4 | 2.17 | 3.4 | 4.54 |
| 5.56 | 5.3 | 5.45 | 3.67 | 2.4 | 6.31 |
| 4.15 | 4 | 4 | 3.9 | 3.4 | 4.08 |
| 2.67 | 4 | 4 | 3.83 | 4.6 | 4.69 |
| 3.67 | 3.15 | 3.5 | 3.23 | 4 | 4 |
| 5.1 | 4.8 | 5 | 4.56 | 3.4 | 5.54 |
| 3.86 | 4 | 3.5 | 3.33 | 6 | 4.54 |
| 3.5 | 3.17 | 3.4 | 3.17 | 3.7 | 4.08 |
| 4.65 | 4.35 | 4.68 | 4.17 | 2.4 | 5.69 |
| 4 | 4 | 4 | 3.4 | 5.4 | 4.69 |
| 4 | 4 | 4.1 | 3.5 | 4.4 | 4.92 |
| 4.15 | 5 | 4.83 | 4.83 | 3.6 | 5.38 |
| 3.65 | 4.2 | 3.83 | 4 | 3.8 | 4.38 |
| 4.9 | 4.5 | 4.98 | 3.76 | 4 | 5.69 |
| 4.11 | 4 | 4.83 | 2.5 | 3.8 | 4.62 |
| 3.98 | 3.9 | 3.7 | 2.17 | 2.2 | 4.23 |
| 4.8 | 4.4 | 5.18 | 4.9 | 4.98 | 5.85 |
| 4.3 | 4.8 | 4.6 | 3.45 | 4 | 5.15 |
| 5 | 5.1 | 4.8 | 5.1 | 2.2 | 5.38 |
| 4.89 | 4.9 | 4.5 | 4.67 | 3.8 | 5.15 |
| 4.23 | 3.9 | 4.5 | 4 | 4.2 | 4.92 |
| 4.24 | 4.5 | 3.5 | 3.83 | 4.9 | 5 |
| 3.15 | 4.33 | 4.33 | 4.2 | 4.2 | 4.69 |
| 5.22 | 3.67 | 3.5 | 4.33 | 4 | 4.54 |
| 4.2 | 4.5 | 4.8 | 1.67 | 4.2 | 5.31 |
| 4.15 | 4.48 | 5.15 | 2.5 | 4 | 5.54 |
| 3.78 | 4.5 | 4.15 | 3.83 | 4 | 4.77 |
| 3.67 | 3.8 | 4.2 | 3.9 | 4.6 | 4.92 |
| 3.45 | 4 | 4.58 | 3.34 | 3.8 | 4.92 |
| 1.2 | 2 | 2 | 2 | 2 | 2.23 |
| 4.34 | 4.5 | 4.4 | 4.83 | 2 | 5.08 |
| 5.11 | 5.4 | 5.18 | 5.2 | 2.4 | 6 |
| 4.65 | 4.78 | 4.33 | 4.8 | 4.8 | 5.08 |
| 4.35 | 4.8 | 4.67 | 4.5 | 5.2 | 5.46 |
| 2.85 | 3.6 | 2.9 | 3.3 | 2.2 | 3.85 |
| 3.45 | 3.9 | 3 | 3.17 | 3.2 | 4.15 |
| 4.68 | 4.45 | 4.35 | 4.78 | 4.79 | 5 |
| 2.31 | 2.67 | 3 | 2.17 | 3 | 3.23 |
| 4.22 | 5.17 | 3.5 | 5.3 | 4.6 | 6.08 |
| 3.26 | 3.83 | 4.3 | 3.5 | 2.4 | 4.69 |
| 4.9 | 4.67 | 4.83 | 3.5 | 4.2 | 5.31 |
| 1 | 1 | 1 | 1.5 | 4 | 1.85 |
| 4.24 | 4.6 | 4.5 | 4.5 | 4.8 | 5.08 |
| 3.6 | 3.85 | 3.5 | 3.33 | 4.6 | 4.77 |
| 3.9 | 4 | 3.98 | 3.67 | 2.4 | 4.38 |
| 3.98 | 3.8 | 4.18 | 3.5 | 3.8 | 4.77 |
| 3.78 | 3.9 | 4.3 | 3.83 | 3.8 | 4.92 |
| 4.11 | 3.5 | 4.25 | 4.67 | 4 | 4.69 |
| 4.15 | 1.8 | 2 | 2 | 3.4 | 2.15 |
| 5.22 | 3.5 | 3.56 | 4.33 | 3 | 4.77 |
| 3.45 | 3.7 | 4 | 3.67 | 3.8 | 4.85 |
| 1.5 | 1.8 | 1.5 | 1.83 | 3.6 | 2 |
| 4 | 5.17 | 4.9 | 2.17 | 3.4 | 5.46 |
| 1.89 | 2 | 2 | 4.33 | 1.8 | 2.31 |
| 1.36 | 1.5 | 2 | 3.83 | 1.6 | 2.77 |
| 4.45 | 4 | 5.15 | 3.17 | 4.67 | 5.69 |
| 4.78 | 3.9 | 5 | 5.1 | 3.8 | 5.38 |
| 4.46 | 4.5 | 4.33 | 4 | 2.8 | 5.23 |
| 2.19 | 2.5 | 2.83 | 4.67 | 3.2 | 3 |
| 4.9 | 4 | 5 | 4.33 | 1.8 | 5.69 |
| 4.33 | 4.12 | 3.67 | 4.33 | 3.6 | 4.38 |
| 4.98 | 4.67 | 5.1 | 2.5 | 3.4 | 5.92 |
| 2.34 | 2.5 | 2.98 | 2.33 | 3 | 3.62 |
| 4.6 | 4.67 | 4.8 | 3.67 | 2.2 | 5.46 |
| 2.9 | 3 | 3 | 2.33 | 2.4 | 3.85 |
| 4.32 | 4.3 | 4.33 | 4.33 | 4.59 | 5.15 |
| 2 | 2 | 2.2 | 3.83 | 4 | 2.54 |
| 2.98 | 2.55 | 3 | 2.17 | 4.2 | 3.77 |
| 1.9 | 2 | 2.45 | 1.83 | 1.8 | 2.92 |
| 2.78 | 3.15 | 3 | 4.5 | 4 | 3.23 |
| 4.23 | 4.5 | 4.33 | 3.83 | 3 | 5.23 |
| 4.5 | 4.2 | 4.8 | 3.5 | 2.2 | 5.31 |
| 3.9 | 3.67 | 4.17 | 3.5 | 2.8 | 4.85 |
| 5 | 5.18 | 5.3 | 1.17 | 2.6 | 6.15 |
| 4.89 | 4.67 | 4.98 | 3.67 | 2.4 | 5.85 |
| 1 | 1.5 | 1 | 2.33 | 3.4 | 1.92 |
| 1.8 | 2 | 2 | 1.83 | 2 | 2.85 |
| 4.44 | 3.87 | 4.1 | 4.83 | 2.2 | 4.85 |
| 2.9 | 3 | 3 | 4.17 | 3.2 | 3.92 |
| 4.4 | 4.5 | 4.5 | 3.33 | 3.4 | 5.23 |
| 4.34 | 4 | 3.8 | 3.9 | 2 | 4.31 |
| 4.1 | 3.67 | 3.34 | 1.83 | 2.6 | 4.54 |
| 4.58 | 4.56 | 4.5 | 2.17 | 4.5 | 5.08 |
| 4.9 | 4.9 | 4.79 | 4.5 | 2.4 | 5 |
| 1.9 | 4 | 4.67 | 2 | 2.8 | 2.38 |
| 3.56 | 3.9 | 4 | 2.45 | 3.8 | 4.69 |
| 3.97 | 4.33 | 3.83 | 2.17 | 4.2 | 4.92 |
| 4 | 4 | 4 | 4.33 | 3 | 4.69 |
| 3.98 | 4.1 | 2.89 | 3.83 | 3.6 | 4.92 |
| 1.9 | 2 | 2 | 4.33 | 2 | 2.62 |
| 3.9 | 4 | 4.75 | 4.5 | 3.4 | 5.38 |
| 4.12 | 4 | 4.67 | 4.5 | 3 | 5.31 |
| 3.98 | 3.67 | 4.33 | 2.17 | 2.8 | 4.69 |
| 4.11 | 4.67 | 4.32 | 4.8 | 4.8 | 5.38 |
| 4.11 | 4.45 | 3.17 | 3.67 | 3.6 | 4.85 |
| 3.9 | 3.56 | 4.83 | 3.5 | 2.4 | 4.92 |
| 1.9 | 2 | 2 | 2 | 3.2 | 2.23 |
| 4.1 | 4.67 | 4.5 | 6.5 | 4.5 | 5 |
| 4.2 | 4.83 | 4.6 | 0.76 | 4.5 | 5 |
| 3.98 | 4.5 | 4 | 5.5 | 2.4 | 4.77 |
| 1.9 | 2 | 4.17 | 1.67 | 3.2 | 2.08 |
| 4.5 | 4 | 4.8 | 4.8 | 4 | 5.38 |
| 4 | 5 | 4.83 | 4.5 | 3.2 | 4.54 |
| 2.67 | 3.15 | 3.17 | 3.5 | 3.65 | 4 |
| 3.67 | 4.17 | 4.83 | 2.67 | 2.6 | 4.85 |
| 5.1 | 5.18 | 5.5 | 5.39 | 3.6 | 6.08 |
| 2.5 | 3.17 | 4.83 | 5 | 3 | 3.5 |
| 4 | 3.2 | 4 | 3.33 | 2.2 | 4.62 |
| 5 | 5 | 4.8 | 3.17 | 3.6 | 5.15 |
| 5.14 | 5.83 | 3.83 | 5.17 | 2.6 | 6.08 |
| 3.9 | 4.5 | 4.33 | 2.67 | 3.4 | 4.31 |
| 4.12 | 4.33 | 3.56 | 2.83 | 2.6 | 4.46 |
| 5 | 4.7 | 3.5 | 5.17 | 2.6 | 5.38 |
| 4.11 | 4 | 4.83 | 3.83 | 2.6 | 4.77 |
| 5.11 | 5 | 4.67 | 2.33 | 3 | 6 |
| 5.23 | 4.23 | 4.5 | 4.83 | 3.2 | 5.08 |
| 3.95 | 4 | 3.67 | 3.67 | 3.2 | 4.62 |
| 3.98 | 4.14 | 4.83 | 4.17 | 2.4 | 4.23 |
| 4.2 | 4.5 | 4.76 | 3.83 | 2.8 | 5 |
| 5.11 | 4.2 | 6 | 4.5 | 3 | 5.69 |
| 4.23 | 5 | 3.67 | 2.17 | 2.2 | 4.54 |
| 5.56 | 5.33 | 4.33 | 5.5 | 2.4 | 6.31 |
| 4.15 | 3.67 | 4.83 | 4.17 | 2.4 | 4.08 |
| 2.67 | 4.18 | 3.65 | 2.5 | 4.4 | 4.69 |
| 3.67 | 4 | 3.17 | 3.83 | 4.4 | 4 |
| 5.1 | 4.69 | 4.9 | 4.33 | 3.6 | 5.54 |
| 3.86 | 3.8 | 3.83 | 3.6 | 3 | 4.54 |
| 3.5 | 3.7 | 4.83 | 4 | 3.8 | 4.08 |
| 4.65 | 4.5 | 3.5 | 3.83 | 2.2 | 5.69 |
| 4 | 3.5 | 4 | 4.33 | 2.8 | 4.69 |
| 4 | 4.4 | 3.83 | 3.83 | 4.6 | 4.92 |
| 4.15 | 4.3 | 4.68 | 4 | 2.8 | 5.38 |
| 3.65 | 3.6 | 3.33 | 4.5 | 3.2 | 4.38 |
| 4.9 | 4.83 | 4.33 | 3.83 | 3.4 | 5.69 |
| 4.11 | 4.45 | 4 | 2.33 | 2.6 | 4.62 |
| 3.98 | 3.8 | 4.67 | 2.17 | 3 | 4.23 |
| 4.8 | 4.67 | 4.67 | 2.17 | 3 | 5.85 |
| 4.3 | 5 | 5 | 1.17 | 3.2 | 5.15 |
| 5 | 4.87 | 4.67 | 4.78 | 2.6 | 5.38 |
| 4.89 | 4.67 | 4.45 | 2.5 | 3.4 | 5.15 |
| 4.23 | 4.5 | 4 | 3 | 3.6 | 4.92 |
| 4.24 | 4.5 | 4.78 | 4.82 | 4.2 | 5 |
| 3.15 | 4.5 | 3.33 | 2.67 | 2.8 | 4.69 |
| 5.22 | 4 | 4.2 | 4.5 | 3 | 4.54 |
| 4.2 | 4.5 | 4.4 | 2.5 | 3.8 | 5.31 |
| 4.15 | 4.3 | 4.7 | 3.83 | 2.8 | 5.54 |
| 3.78 | 3.8 | 3.83 | 4.17 | 3.2 | 4.77 |
| 3.67 | 3.68 | 4 | 1.33 | 1.8 | 4.92 |
| 3.45 | 3.8 | 4 | 4.5 | 3.6 | 4.92 |
| 1.2 | 2 | 2 | 2.5 | 3.4 | 2.23 |
| 4.34 | 4.83 | 4.8 | 4 | 3 | 5.08 |
| 5.11 | 5.17 | 4.58 | 5.5 | 2.2 | 6 |
| 4.65 | 4.67 | 4.78 | 4 | 2.4 | 5.08 |
| 4.35 | 4.3 | 5.5 | 3.17 | 5 | 5.46 |
| 2.85 | 2.5 | 3 | 3 | 4 | 3.85 |
| 3.45 | 3.5 | 3.5 | 3.83 | 4.2 | 4.15 |
| 4.68 | 4.3 | 5.33 | 3.33 | 4.5 | 5 |
| 2.31 | 3 | 3 | 3.33 | 4 | 3.23 |
| 4.22 | 5.1 | 5 | 1.83 | 3 | 6.08 |
| 3.26 | 4.12 | 3 | 1.5 | 2.2 | 4.69 |
| 4.9 | 4.67 | 3.67 | 1.5 | 2.8 | 5.31 |
| 1 | 2 | 5.17 | 3.67 | 2.6 | 1.85 |
| 4.24 | 4.5 | 5.5 | 3.33 | 2.4 | 5.08 |
| 3.6 | 3.45 | 5.33 | 4.17 | 3.4 | 4.77 |
| 3.9 | 4.3 | 4 | 1.83 | 2 | 4.38 |
| 3.98 | 4.1 | 4.5 | 2.67 | 2.2 | 4.77 |
| 3.78 | 4.65 | 3.9 | 3.17 | 3.2 | 4.92 |
| 4.11 | 4 | 4.79 | 2.67 | 3.4 | 4.69 |
| 4.15 | 3 | 2 | 2.33 | 2 | 2.15 |
| 5.22 | 3.9 | 3.9 | 2.67 | 2.6 | 4.77 |
| 3.45 | 4.2 | 3.5 | 1.33 | 3.2 | 4.85 |
| 1.5 | 1.8 | 1.8 | 3.33 | 2 | 2 |
| 4 | 4.33 | 3.5 | 4.83 | 2.6 | 5.46 |
| 1.89 | 3.5 | 5.15 | 1.5 | 3.6 | 2.31 |
| 1.36 | 2.5 | 2 | 4.17 | 3 | 2.77 |
| 4.45 | 4.15 | 4.67 | 2.67 | 2.2 | 5.69 |
| 4.78 | 5.33 | 4 | 2.5 | 3.6 | 5.38 |
| 4.46 | 4.1 | 4.5 | 2.17 | 2.6 | 5.23 |
| 2.19 | 2.9 | 4.5 | 2.33 | 2.34 | 3 |
| 4.9 | 3.78 | 4.23 | 1.67 | 2.6 | 5.69 |
| 4.33 | 3.67 | 5.15 | 1.67 | 2.6 | 4.38 |
| 4.98 | 3.5 | 3.57 | 3.5 | 2.6 | 5.92 |
| 2.34 | 2.98 | 3.12 | 1.5 | 3 | 3.62 |
| 4.6 | 4.3 | 3.8 | 4.17 | 3.2 | 5.46 |
| 2.9 | 3.7 | 3 | 3.17 | 3.2 | 3.85 |
| 4.32 | 3 | 3.56 | 3.5 | 2.4 | 5.15 |
| 2 | 3 | 2 | 3.17 | 2.8 | 2.54 |
| 2.98 | 3 | 3 | 3.17 | 3 | 3.77 |
| 1.9 | 3 | 2 | 4.33 | 2.2 | 2.92 |

**#import MR1 data**

> attach(MR1)

> names(MR1)

> help(lm)

> model1<-lm(Buying~Attitude+Awarness+Perception+Cost+Rating)

> summary(model1)

Call:

lm(formula = Buying ~ Attitude + Awarness + Perception + Cost + Rating)

Residuals:

Min 1Q Median 3Q Max

-1.68177 -0.21698 0.01556 0.23072 1.20153

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 0.41711 0.10857 3.842 0.000148 \*\*\*

Attitude 0.37526 0.04015 9.346 < 2e-16 \*\*\*

Awarness 0.45181 0.04868 9.281 < 2e-16 \*\*\*

Perception 0.18950 0.03840 4.935 1.3e-06 \*\*\*

Cost 0.03171 0.02267 1.399 0.162826

Rating 0.02451 0.02536 0.966 0.334584

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3817 on 316 degrees of freedom

Multiple R-squared: 0.8639, Adjusted R-squared: 0.8618

F-statistic: 401.3 on 5 and 316 DF, p-value: < 2.2e-16

> confint(model1,conf.level=0.95)

2.5 % 97.5 %

(Intercept) 0.20349056 0.63072642

Attitude 0.29626143 0.45425336

Awarness 0.35603210 0.54759682

Perception 0.11394486 0.26505864

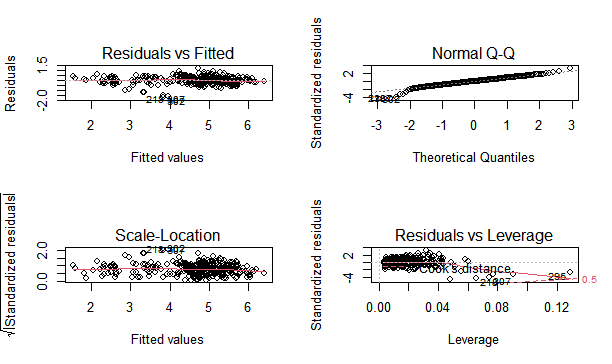
Cost -0.01288777 0.07630075

Rating -0.02539220 0.07441612

> plot(model1)

>par(mfrow=c(2,2))

> plot(model1)



1. **Chi-square test:**

**#Test of godness of fit**

**jobs<-c(11091,11282,15378,12696)**

**names(jobs)<-c("Project Management","Supply Chain","Service","Quality")**

**jobs**

Project Management Supply Chain Service Quality

11091 11282 15378 12696

**jobs/sum(jobs)**

Project Management Supply Chain Service Quality

0.2198545 0.2236407 0.3048348 0.2516701

**probability<-c(0.25,0.25,0.25,0.25)**

**#H0: Proportion of jobs in each category is 0.25**

**#Ha: Proportion of jobs in each category is not same.**

**chisq.test(jobs,p=probability)**

Chi-squared test for given probabilities

data: jobs

X-squared = 930.89, df = 3, p-value < 2.2e-16

**Example-1**

#C1,C2,C3,C4 and C5 number of students registered for 5 classes

> data<-c(23,45,34,34,45)

**#H0:p1=p2=p3=p4=p5**

**#H1:p1=!p2=!p3=!p4=!p5**

> chisq.test(data)

Chi-squared test for given probabilities

data: data

X-squared = 9.3591, df = 4, p-value = 0.05272

**Example 2:File name :chi1: Educational Background and Grade**

|  |  |
| --- | --- |
| **code** | **grade** |
| BCOM | B |
| BCOM | C |
| BCOM | A |
| BCOM | C |
| BCOM | B |
| BA | A |
| BA | A |
| BA | A |
| BA | B |
| BA | A |
| BCA | B |
| BCA | A |
| BCA | B |
| BCA | B |
| BCA | C |
| BE | C |
| BE | C |
| BE | A |
| BE | B |
| BE | C |
| BBA | C |
| BBA | B |
| BBA | C |
| BBA | C |
| BBA | C |

**#import data**

> attach(chi1)

> table(code)

code

BA BBA BCA BCOM BE

5 5 5 5 5

> table(grade)

grade

A B C

7 8 10

> table(code,grade)

grade

code A B C

BA 4 1 0

BBA 0 1 4

BCA 1 3 1

BCOM 1 2 2

BE 1 1 3

> TAB=table(code,grade)

> barplot(TAB,beside=T,legend=T)

> chisq.test(TAB,correct=T)

Pearson's Chi-squared test

data: TAB

X-squared = 13.571, df = 8, p-value = 0.09364

> CHI= chisq.test(TAB,correct=T)

> CHI

Pearson's Chi-squared test

data: TAB

X-squared = 13.571, df = 8, p-value = 0.09364

> attributes(CHI)

$names

[1] "statistic" "parameter" "p.value" "method" "data.name" "observed"

[7] "expected" "residuals" "stdres"

$class

[1] "htest"

> CHI$expected

grade

code A B C

BA 1.4 1.6 2

BBA 1.4 1.6 2

BCA 1.4 1.6 2

BCOM 1.4 1.6 2

BE 1.4 1.6 2

> fisher.test(TAB,conf.int = T,conf.level = 0.95)

Fisher's Exact Test for Count Data

data: TAB

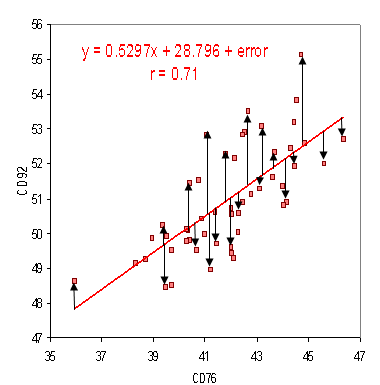
p-value = 0.1502

alternative hypothesis: two.sided

1. **Logistic Regression**

**What is Regression?**

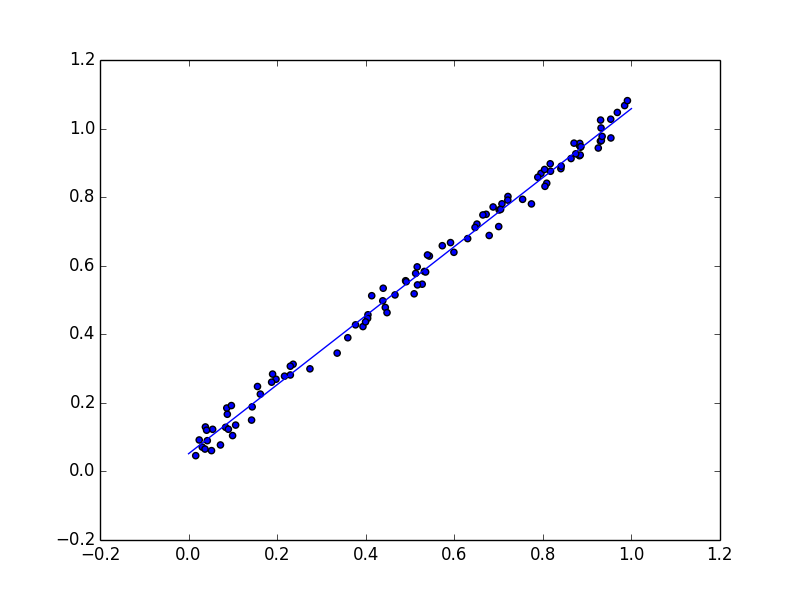
* **Regression analysis is a predictive modeling technique.**
* **It estimates the relationship between a dependent (target) and an independent variable(Predictor)**
* **Scatter plot with regression line.**

****

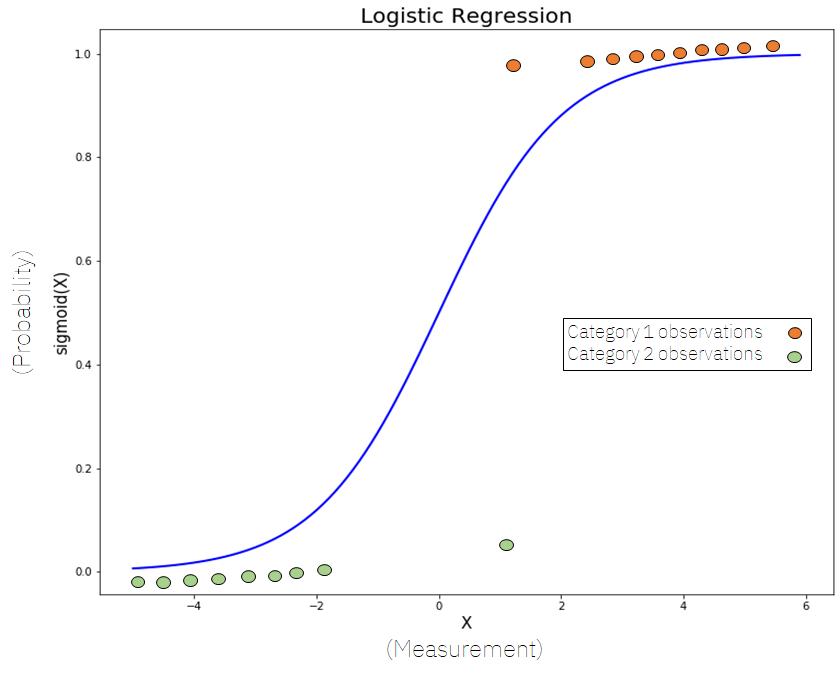
**Regression Equation: Y = 0.5297X+28.796, for any value of X , we can predict the value of Y.**

**Types of Regression**

1. **Linear Regression: When there is a linear relationship between independent and dependent variables.**

****

1. **Logistic Regression: When the dependent variable is categorical (0/1, True/False, Successful/Unsuccessful, A/B/C) in nature.**

****

**Sigmoide Curve(S-Curve)**

1. **Polynomial Regression: When the relationship between the independent and dependent variables is not linear.**

**Why Logistic Regression?**

**Whenever the outcome of the dependent variable (Y) is discrete like 0 or 1, Yes or No, A, B, C, we use logistic regression.**

**Why can’t we use linear regression?**

**Since our value of Y will be between 0 and 1 in logistic regression but in linear regression it may cross 0 or 1, so, the linear line has to be clipped at 0 and 1. With this our resulting curve cannot be formulated into a single formula. So we needed a new way to solve this kind of problem.. Hence logistic regression is required.**

**Equation for a straight line:**

**Y= β0+β1X1+β2X2+………..………. , Range of Y is from -**∞ to + ∞

Lts try to find the logistic regression from the above equation.

Y = **β0+β1X1+β2X2+………..**…………. In logistic equation Y can be only between 0 and 1.

Now, to get the range of Y between 0 to + ∞, lets transform Y

Y Y=0] 0

1-Y Y =1] ∞, Now, we have range between 0 to ∞

Let us transform it further, to get the range between - ∞ to ∞

Y

**Log = β0+β1X1+β2X2+………..……..**

**1-Y**

**What is logistic Regression?**

Logistic Regression or logit regression or logit model is a regression model where the dependent variable is categorical.

Categorical: Variables that can be only fixed values such as A,B or C , Yes or No.

Y= F(X), Y is dependent on X.

**How does logistic regression work?**

|  |
| --- |
| **IQ of Candidates**  Selected  147,120,121,128,110,119,133 |
| **110** |
| **147** |
| **120** |
| **107**    MODEL |
| **89** |
| **92** |
| **106** |
| **121** |
| **127** |
| **104**  Not Selected  107, 89, 92,106,104,114 |
| **137** |
| **133** |
| **114** |
| **126** |
| **121** |
| **119** |

Before creating the model, we divide our dataset into training data (estimation) and testing data (validation).

**Logistic Regression Equation:**

Y

**Log = β0+β1X1+β2X2+………..**

**1-Y**

**Logistic Regression Equation:**

Y e β0+β1X1+β2X2

**Logit(Y)=Log i.e. P(Y) =**

**1-Y 1+** e β0+β1X1+β2X2

**Example: Logistic Regression in R**

**Objective: To predict the patient is diabetic or not based on the following data.**

**Npreg= number of pregnancies**

**Glu= plasma glucose concentration**

**Bp=Blood Pressure**

**Skin: Triceps skin fold thickness**

**Bmi=body mass index**

**Ped =diabetes pedigree function**

**Age = Age in Years**

**Type: 1 for Yes and 0 for No diabetic**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Sr No.** | **npreg** | **glu** | **bp** | **skin** | **bmi** | **ped** | **age** | **type** |
| 1 | 6 | 148 | 72 | 35 | 33.6 | 0.627 | 50 | 1 |
| 2 | 1 | 85 | 66 | 29 | 26.6 | 0.351 | 31 | 0 |
| 3 | 1 | 89 | 66 | 23 | 28.1 | 0.167 | 37 | 0 |
| 4 | 3 | 78 | 50 | 32 | 31.1 | 0.248 | 26 | 1 |
| 5 | 2 | 197 | 70 | 45 | 30.5 | 0.158 | 53 | 1 |
| 6 | 5 | 166 | 72 | 19 | 25.8 | 0.587 | 51 | 1 |
| 7 | 0 | 118 | 84 | 47 | 45.8 | 0.551 | 31 | 0 |
| 8 | 1 | 103 | 30 | 38 | 43.3 | 0.183 | 33 | 1 |
| 9 | 3 | 126 | 88 | 41 | 39.3 | 0.704 | 27 | 0 |
| 10 | 9 | 119 | 80 | 35 | 29 | 0.263 | 29 | 1 |
| 10 | 6 | 148 | 72 | 35 | 33.6 | 0.345 | 39 | 1 |
| 10 | 1 | 47 | 66 | 29 | 26.6 | 0.351 | 31 | 1 |
| 10 | 1 | 89 | 72 | 23 | 28.1 | 0.167 | 21 | 0 |
| 10 | 3 | 78 | 50 | 32 | 31.1 | 0.248 | 26 | 1 |
| 10 | 2 | 197 | 70 | 45 | 30.5 | 0.158 | 53 | 0 |
| 10 | 5 | 166 | 67 | 19 | 25.8 | 0.587 | 51 | 1 |
| 10 | 0 | 148 | 69 | 49 | 45.8 | 0.341 | 31 | 1 |
| 10 | 1 | 103 | 30 | 38 | 43.3 | 0.245 | 33 | 0 |
| 10 | 3 | 126 | 88 | 41 | 39.3 | 0.704 | 27 | 0 |
| 10 | 9 | 119 | 80 | 35 | 29 | 0.263 | 29 | 1 |
| 10 | 6 | 148 | 72 | 35 | 33.6 | 0.627 | 50 | 0 |
| 10 | 1 | 85 | 66 | 29 | 26.6 | 0.456 | 31 | 1 |
| 10 | 1 | 89 | 66 | 23 | 28.1 | 0.167 | 21 | 0 |
| 10 | 3 | 78 | 50 | 32 | 31.1 | 0.248 | 26 | 1 |
| 10 | 2 | 197 | 82 | 45 | 30.5 | 0.158 | 53 | 1 |
| 10 | 5 | 160 | 72 | 19 | 25.8 | 0.587 | 54 | 0 |
| 10 | 0 | 139 | 67 | 47 | 45.8 | 0.551 | 31 | 1 |
| 10 | 1 | 103 | 30 | 34 | 43.3 | 0.183 | 39 | 0 |
| 10 | 3 | 126 | 88 | 41 | 39.3 | 0.704 | 27 | 1 |
| 10 | 9 | 125 | 80 | 35 | 29 | 0.263 | 27 | 1 |

**# Import data in R: File name logit**

> attach(logit)

> model<-glm(type~npreg+glu+bp+skin+bmi+ped+age,data=logit,family = "binomial")

> summary(model)

Call

glm(formula = type ~ npreg + glu + bp + skin + bmi + ped + age,

family = "binomial", data = logit)

Deviance Residuals:

Min 1Q Median 3Q Max

-1.9717 -0.8216 0.3997 0.9365 1.4174

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) 9.83340 7.24855 1.357 0.1749

npreg 0.34506 0.27146 1.271 0.2037

glu 0.02447 0.03064 0.798 0.4246

bp -0.10916 0.07010 -1.557 0.1194

skin 0.22007 0.11280 1.951 0.0511 .

bmi -0.32657 0.19175 -1.703 0.0886 .

ped 4.53606 4.38375 1.035 0.3008

age -0.12400 0.10953 -1.132 0.2576

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 40.381 on 29 degrees of freedom

Residual deviance: 31.049 on 22 degrees of freedom

AIC: 47.049

Number of Fisher Scoring iterations: 5

> res<-predict(model,logit,type="response")

> res

1 2 3 4 5 6 7 8

0.73862507 0.62170014 0.05763389 0.89145512 0.85685510 0.46377012 0.08260618

0.59482206

9 10 11 12 13 14 15 16

0.59365531 0.95008888 0.75467219 0.39342214 0.18766890 0.89145512 0.85685510 0.59884187

17 18 19 20 21 22 23 24

0.36622466 0.66042119 0.59365531 0.95008888 0.73862507 0.72572843 0.30783728 0.89145512

25 26 27 28 29 30

0.61761878 0.33984653 0.49052975 0.22437386 0.59365531 0.96581264

> logit

# A tibble: 30 x 9

`Sr No.` npreg glu bp skin bmi ped age type

*<dbl>* *<dbl>* *<dbl>* *<dbl>* *<dbl>* *<dbl>* *<dbl>* *<dbl>* *<dbl>*

1 1 6 148 72 35 33.6 0.627 50 1

2 2 1 85 66 29 26.6 0.351 31 0

3 3 1 89 66 23 28.1 0.167 37 0

4 4 3 78 50 32 31.1 0.248 26 1

5 5 2 197 70 45 30.5 0.158 53 1

6 6 5 166 72 19 25.8 0.587 51 1

7 7 0 118 84 47 45.8 0.551 31 0

8 8 1 103 30 38 43.3 0.183 33 1

9 9 3 126 88 41 39.3 0.704 27 0

10 10 9 119 80 35 29 0.263 29 1

# ... with 20 more rows

And so on…. Upto 30.

> table(Actualvalue=logit$type,Predictedvalue=res>0.5)

Predictedvalue

Actualvalue FALSE TRUE

0 6 6

1 4 14

> (6+14)/(6+6+4+14)

[1] 0.6666667

> table(Actualvalue=logit$type,Predictedvalue=res>0.3)

Predictedvalue

Actualvalue FALSE TRUE

0 4 8

1 0 18

> (4+18)/(4+8+0+18)

[1] 0.7333333

**Or**

**#Import data**

**#Attach data**

> install.packages("caTools")

> library("caTools")

> split<-sample.split((logit,splitRatio=0.8))

> split

>training<-subset(logit,split==”True”)

>testing<-subset(logit,split==”FALSE”)

# The data will split into training and testing with the ratio:80:20

>model<-glm(type~.,training,family=”binomial”)

>model<-glm(type~.-skin,training,family=”binomial”)

**# Null Deviance shows how well the response variable is predicted by a model that includes only the intercept.**

**# Residual deviance shows how well the response variable is predicted with the inclusion of independent variables.**

res<-predict(model,testing,type="response")

> res

>testing

> table(Actualvalue=testing$type,Predictedvalue=res>0.5)

> table(Actualvalue=testing$type,Predictedvalue=res>0.3)

**# How to find the threshold value?**

**# ROC**

**# Store the predicted values for training dataset in ‘res’ variable.**

>res<-predict (model,training,type="response")

> install.packages("ROCR")

> library(ROCR)

#import the library for the ROCR package

# Define the ‘ROCRPred’ and ‘ROCRPref’ variables

>ROCRPred=prediction(res,training$type)

>ROCRPref<-performance(ROCRPred,”tpr”,”fpr”)

>plot(ROCRPref,colorize=TRUE,print.cutoff.at=seq(0.1,by=0.1))

**# Use to calculate pseudo R2**

> install.packages("rcompanion")

> library(rcompanion)

> nagelkerke(model)

Y=

**>exp(y)/(1+exp(y)**

1. **Linear Discriminant Analysis : File name:dis1**

**Linear discriminant analysis** (**LDA**), **normal discriminant analysis** (**NDA**), or **discriminant function analysis** is a generalization of **Fisher's linear discriminant**, a method used in [statistics](https://en.wikipedia.org/wiki/Statistics), [pattern recognition](https://en.wikipedia.org/wiki/Pattern_recognition), and [machine learning](https://en.wikipedia.org/wiki/Machine_learning) to find a [linear combination](https://en.wikipedia.org/wiki/Linear_combination) of [features](https://en.wikipedia.org/wiki/Features_(pattern_recognition)) that characterizes or separates two or more classes of objects or events. The resulting combination may be used as a [linear classifier](https://en.wikipedia.org/wiki/Linear_classifier), or, more commonly, for [dimensionality reduction](https://en.wikipedia.org/wiki/Dimensionality_reduction) before later [classification](https://en.wikipedia.org/wiki/Statistical_classification).

**# Import data**

> attach(dis1)

> library(MASS)

> dis1

> head(dis1)

> ldaout<-lda(Buyer~Durability+Mileage+`Interior Design`+Look,dis1)

> ldaout

Call:

lda(Buyer ~ Durability + Mileage + `Interior Design` + Look,

data = dis1)

Prior probabilities of groups:

Buyer Non Buyer

0.5 0.5

Group means:

Durability Mileage `Interior Design` Look

Buyer 48.5 52.40 52.300 52.425

Non Buyer 28.6 33.45 31.925 36.000

Coefficients of linear discriminants:

LD1

Durability -0.04758718

Mileage -0.04911384

`Interior Design` -0.03528799

Look -0.04482465

> ldapred<-predict(ldaout,dis1)

> ldapred

> ldaclass<-ldapred$class

> ldaclass

> ldatable<-table(ldaclass,dis1$Buyer)

> ldatable

ldaclass Buyer Non Buyer

Buyer 38 1

Non Buyer 2 39

> accur<-sum(diag(ldatable))/sum(ldatable)\*100

> accur

[1] 96.25

1. **Exploratory Factor Analysis (EFA): File name –EFA\_Delta**

> r=cor(EFA\_Delta)

> install.packages("psych") for KMO and Bartlett.

> library(psych)

> KMO(EFA\_Delta)

> cortest.bartlett(EFA\_Delta or r)

or

> install.packages("REdaS") for KMO and Bartlett.

> library(REdaS)

> KMOS(EFA\_Delta)

> bart\_spher(EFA\_Delta)

>r=cor(EFA\_delta)

>r

>pca(r,nfactor=10,rotate=F)

>z=pca(r,nfactors =3,method =regression,rotate ="varimax",scores = T)

>z

>z$values

> print(Z$values,digits=3)

> print(Z$loadings,digits=3,cutoff = 0.7)