MLDE \_ RLS

POR: JHONATAN CASTAÑEDA

***Sin Centrar:***

> library(readxl)

> excel\_sheets(path = file.path("data\_rls\_uti.xlsx"))

[1] "Hoja1" "Hoja2" "Hoja3"

> data <- read\_excel("data\_rls\_uti.xlsx",sheet = "Hoja1",col\_names = TRUE, na="")

> str(data)

Classes ‘tbl\_df’, ‘tbl’ and 'data.frame': 40 obs. of 2 variables:

$ Utilidad: num 6017 8049 8551 6720 7391 ...

$ Ventas : num 13270 17127 17814 16000 18026 ...

> summary (data)

Utilidad Ventas

Min. :4437 Min. :10020

1st Qu.:5291 1st Qu.:11482

Median :6732 Median :15314

Mean :6684 Mean :14881

3rd Qu.:8060 3rd Qu.:17910

Max. :9088 Max. :19841

> #Regresion lineal

> reg <- lm(Utilidad~Ventas , data)

> str(reg)

List of 12

$ coefficients : Named num [1:2] 137.08 0.44

..- attr(\*, "names")= chr [1:2] "(Intercept)" "Ventas"

$ residuals : Named num [1:40] 42 377 577 -456 -676 ...

..- attr(\*, "names")= chr [1:40] "1" "2" "3" "4" ...

$ effects : Named num [1:40] -42272 -8694 518 -484 -738 ...

..- attr(\*, "names")= chr [1:40] "(Intercept)" "Ventas" "" "" ...

$ rank : int 2

$ fitted.values: Named num [1:40] 5975 7672 7974 7176 8067 ...

..- attr(\*, "names")= chr [1:40] "1" "2" "3" "4" ...

$ assign : int [1:2] 0 1

$ qr :List of 5

..$ qr : num [1:40, 1:2] -6.325 0.158 0.158 0.158 0.158 ...

.. ..- attr(\*, "dimnames")=List of 2

.. .. ..$ : chr [1:40] "1" "2" "3" "4" ...

.. .. ..$ : chr [1:2] "(Intercept)" "Ventas"

.. ..- attr(\*, "assign")= int [1:2] 0 1

..$ qraux: num [1:2] 1.16 1.12

..$ pivot: int [1:2] 1 2

..$ tol : num 1e-07

..$ rank : int 2

..- attr(\*, "class")= chr "qr"

$ df.residual : int 38

$ xlevels : Named list()

$ call : language lm(formula = Utilidad ~ Ventas, data = data)

$ terms :Classes 'terms', 'formula' length 3 Utilidad ~ Ventas

.. ..- attr(\*, "variables")= language list(Utilidad, Ventas)

.. ..- attr(\*, "factors")= int [1:2, 1] 0 1

.. .. ..- attr(\*, "dimnames")=List of 2

.. .. .. ..$ : chr [1:2] "Utilidad" "Ventas"

.. .. .. ..$ : chr "Ventas"

.. ..- attr(\*, "term.labels")= chr "Ventas"

.. ..- attr(\*, "order")= int 1

.. ..- attr(\*, "intercept")= int 1

.. ..- attr(\*, "response")= int 1

.. ..- attr(\*, ".Environment")=<environment: R\_GlobalEnv>

.. ..- attr(\*, "predvars")= language list(Utilidad, Ventas)

.. ..- attr(\*, "dataClasses")= Named chr [1:2] "numeric" "numeric"

.. .. ..- attr(\*, "names")= chr [1:2] "Utilidad" "Ventas"

$ model :'data.frame': 40 obs. of 2 variables:

..$ Utilidad: num [1:40] 6017 8049 8551 6720 7391 ...

..$ Ventas : num [1:40] 13270 17127 17814 16000 18026 ...

..- attr(\*, "terms")=Classes 'terms', 'formula' length 3 Utilidad ~ Ventas

.. .. ..- attr(\*, "variables")= language list(Utilidad, Ventas)

.. .. ..- attr(\*, "factors")= int [1:2, 1] 0 1

.. .. .. ..- attr(\*, "dimnames")=List of 2

.. .. .. .. ..$ : chr [1:2] "Utilidad" "Ventas"

.. .. .. .. ..$ : chr "Ventas"

.. .. ..- attr(\*, "term.labels")= chr "Ventas"

.. .. ..- attr(\*, "order")= int 1

.. .. ..- attr(\*, "intercept")= int 1

.. .. ..- attr(\*, "response")= int 1

.. .. ..- attr(\*, ".Environment")=<environment: R\_GlobalEnv>

.. .. ..- attr(\*, "predvars")= language list(Utilidad, Ventas)

.. .. ..- attr(\*, "dataClasses")= Named chr [1:2] "numeric" "numeric"

.. .. .. ..- attr(\*, "names")= chr [1:2] "Utilidad" "Ventas"

- attr(\*, "class")= chr "lm"

> anova<-aov(reg)

> summary(reg)

Call:

lm(formula = Utilidad ~ Ventas, data = data)

Residuals:

Min 1Q Median 3Q Max

-676.35 -302.04 42.59 303.67 612.49

Coefficients:

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

(Intercept) 137.08270 282.69543 0.485 0.631

Ventas 0.43994 0.01859 23.663 <2e-16 \*\*\*

---

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

Residual standard error: 367.4 on 38 degrees of freedom

Multiple R-squared: 0.9364, Adjusted R-squared: 0.9348

F-statistic: 559.9 on 1 and 38 DF, p-value: < 2.2e-16

> summary(anova)

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

Ventas 1 75578286 75578286 559.9 <2e-16 \*\*\*

Residuals 38 5129142 134977

---

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

> #Valor cuartil t de student, fisher

> qt(0.975 , df =38)

[1] 2.024394

> qf(0.95 , df1=1,df2=38)

[1] 4.098172

> #B1 no es significativo, por lo que se debe centrar los datos

> #------------------------------------------------------------

> #intervalos de confianza

> confint(reg , level=0.95)

2.5 % 97.5 %

(Intercept) -435.2042893 709.3696869

Ventas 0.4022981 0.4775722

> names(reg)

[1] "coefficients" "residuals" "effects" "rank"

[5] "fitted.values" "assign" "qr" "df.residual"

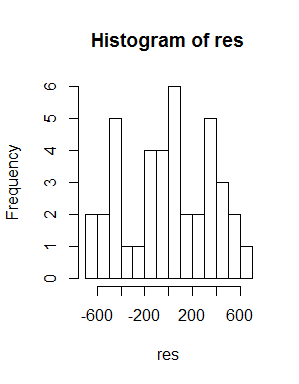
[9] "xlevels" "call" "terms" "model"

> res<-reg[["residuals"]]

> pred<-reg[["fitted.values"]]

> data2<-data.frame(data,Predicciones=pred,Residuos=res)

> hist(res,15)



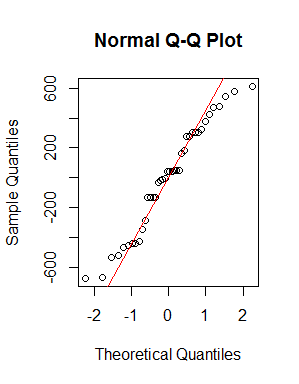
> mean(res)

[1] -8.14071e-15

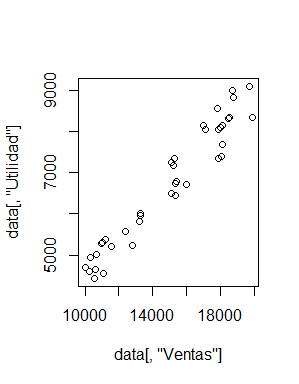
> #prueba de normalidad

> qqnorm(res)

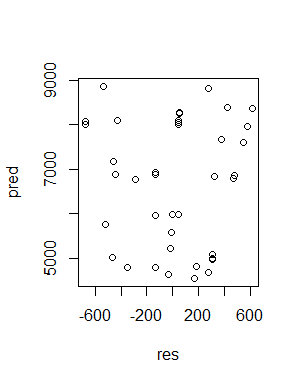
> qqline(res,col="red")



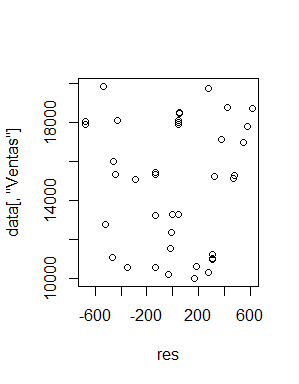
> plot(data[,"Ventas"],data[,"Utilidad"])



> plot(res,pred)



> plot(res, data[,"Ventas"])



Como se vio anteriormente, se debe centrar los datos puesto que B1 no es significativo, por tanto se hace:

#Simplemente para cargar la librería.

> library(readxl)

#para cargar la data y poder utilizarla, aquí están los datos observados de las dos variables que se quiere relacionar con una regresión lineal

> excel\_sheets(path = file.path("data\_rls\_uti.xlsx"))

[1] "Hoja1" "Hoja2" "Hoja3"

> data <- read\_excel("data\_rls\_uti.xlsx",sheet = "Hoja1",col\_names = TRUE, na="")

#un resumen de lo cargado se obtiene al hacer:

> str(data)

Classes ‘tbl\_df’, ‘tbl’ and 'data.frame': 40 obs. of 2 variables:

$ Utilidad: num 6017 8049 8551 6720 7391 ...

$ Ventas : num 13270 17127 17814 16000 18026 ...

#y/o

> summary (data)

Utilidad Ventas

Min. :4437 Min. :10020

1st Qu.:5291 1st Qu.:11482

Median :6732 Median :15314

Mean :6684 Mean :14881

3rd Qu.:8060 3rd Qu.:17910

Max. :9088 Max. :19841

#para poder centrar las 2 variables, utilidad y ventas, se resta componente a componente la media de las variables respectivamente asi:

> data[,"Utilidad"] <- data[,"Utilidad"]-mean(data[,"Utilidad"])

> data[,"Ventas"] <- data[,"Ventas"]-mean(data[,"Ventas"])

#un resumen de lo cargado ya centrado se obtiene al hacer:

> str(data)

Classes ‘tbl\_df’, ‘tbl’ and 'data.frame': 40 obs. of 2 variables:

$ Utilidad: num -666.7 1365.3 1867.3 36.3 707.3 ...

$ Ventas : num -1611 2246 2933 1119 3145 ...

#y/o

> summary (data)

Utilidad Ventas

Min. :-2246.72 Min. :-4860.9

1st Qu.:-1392.47 1st Qu.:-3398.7

Median : 48.27 Median : 432.6

Mean : 0.00 Mean : 0.0

3rd Qu.: 1376.28 3rd Qu.: 3028.8

Max. : 2404.28 Max. : 4960.1

> #Regresion lineal

Para que r almacene todos los datos obtenidos de hacer la regresión lineal de utilidad y ventas en una nueva variable llamada reg se tiene que hacer:

> reg <- lm(Utilidad~Ventas , data)

> anova<-aov(reg)

#un resumen de esa regresión lineal y de anova serían:

> str(reg)

List of 12

$ coefficients : Named num [1:2] -6.44e-13 4.40e-01

..- attr(\*, "names")= chr [1:2] "(Intercept)" "Ventas"

$ residuals : Named num [1:40] 42 377 577 -456 -676 ...

..- attr(\*, "names")= chr [1:40] "1" "2" "3" "4" ...

$ effects : Named num [1:40] 2.27e-12 -8.69e+03 5.18e+02 -4.84e+02 -7.38e+02 ...

..- attr(\*, "names")= chr [1:40] "(Intercept)" "Ventas" "" "" ...

$ rank : int 2

$ fitted.values: Named num [1:40] -709 988 1290 492 1384 ...

..- attr(\*, "names")= chr [1:40] "1" "2" "3" "4" ...

$ assign : int [1:2] 0 1

$ qr :List of 5

..$ qr : num [1:40, 1:2] -6.325 0.158 0.158 0.158 0.158 ...

.. ..- attr(\*, "dimnames")=List of 2

.. .. ..$ : chr [1:40] "1" "2" "3" "4" ...

.. .. ..$ : chr [1:2] "(Intercept)" "Ventas"

.. ..- attr(\*, "assign")= int [1:2] 0 1

..$ qraux: num [1:2] 1.16 1.12

..$ pivot: int [1:2] 1 2

..$ tol : num 1e-07

..$ rank : int 2

..- attr(\*, "class")= chr "qr"

$ df.residual : int 38

$ xlevels : Named list()

$ call : language lm(formula = Utilidad ~ Ventas, data = data)

$ terms :Classes 'terms', 'formula' length 3 Utilidad ~ Ventas

.. ..- attr(\*, "variables")= language list(Utilidad, Ventas)

.. ..- attr(\*, "factors")= int [1:2, 1] 0 1

.. .. ..- attr(\*, "dimnames")=List of 2

.. .. .. ..$ : chr [1:2] "Utilidad" "Ventas"

.. .. .. ..$ : chr "Ventas"

.. ..- attr(\*, "term.labels")= chr "Ventas"

.. ..- attr(\*, "order")= int 1

.. ..- attr(\*, "intercept")= int 1

.. ..- attr(\*, "response")= int 1

.. ..- attr(\*, ".Environment")=<environment: R\_GlobalEnv>

.. ..- attr(\*, "predvars")= language list(Utilidad, Ventas)

.. ..- attr(\*, "dataClasses")= Named chr [1:2] "numeric" "numeric"

.. .. ..- attr(\*, "names")= chr [1:2] "Utilidad" "Ventas"

$ model :'data.frame': 40 obs. of 2 variables:

..$ Utilidad: num [1:40] -666.7 1365.3 1867.3 36.3 707.3 ...

..$ Ventas : num [1:40] -1611 2246 2933 1119 3145 ...

..- attr(\*, "terms")=Classes 'terms', 'formula' length 3 Utilidad ~ Ventas

.. .. ..- attr(\*, "variables")= language list(Utilidad, Ventas)

.. .. ..- attr(\*, "factors")= int [1:2, 1] 0 1

.. .. .. ..- attr(\*, "dimnames")=List of 2

.. .. .. .. ..$ : chr [1:2] "Utilidad" "Ventas"

.. .. .. .. ..$ : chr "Ventas"

.. .. ..- attr(\*, "term.labels")= chr "Ventas"

.. .. ..- attr(\*, "order")= int 1

.. .. ..- attr(\*, "intercept")= int 1

.. .. ..- attr(\*, "response")= int 1

.. .. ..- attr(\*, ".Environment")=<environment: R\_GlobalEnv>

.. .. ..- attr(\*, "predvars")= language list(Utilidad, Ventas)

.. .. ..- attr(\*, "dataClasses")= Named chr [1:2] "numeric" "numeric"

.. .. .. ..- attr(\*, "names")= chr [1:2] "Utilidad" "Ventas"

- attr(\*, "class")= chr "lm"

#y/o

> summary(reg)

Call:

lm(formula = Utilidad ~ Ventas, data = data)

Residuals:

Min 1Q Median 3Q Max

-676.35 -302.04 42.59 303.67 612.49

Coefficients:

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

(Intercept) -6.442e-13 5.809e+01 0.00 1

Ventas 4.399e-01 1.859e-02 23.66 <2e-16 \*\*\*

---

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

Residual standard error: 367.4 on 38 degrees of freedom

Multiple R-squared: 0.9364, Adjusted R-squared: 0.9348

F-statistic: 559.9 on 1 and 38 DF, p-value: < 2.2e-16

#y

> summary(anova)

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

Ventas 1 75578286 75578286 559.9 <2e-16 \*\*\*

Residuals 38 5129142 134977

---

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

> #Valor cuartil t de student, fisher

> qt(0.975 , df =38)

[1] 2.024394

> qf(0.95 , df1=1,df2=38)

[1] 4.098172

#intervalos de confianza

> confint(reg , level=0.95)

2.5 % 97.5 %

(Intercept) -117.5968432 117.5968432

Ventas 0.4022981 0.4775722

#se tiene entonces que el término B1 es 0 y que el termino de B2 es el mismo que se obtuvo sin centrar los datos, que es lo que esperábamos, además que nuestra aproximación de B2 es 4.399e-01 > 0 que es también lo que se esperaba teóricamente, que si aumentan las ventas, aumente la utilidad, por tanto el modelo tiene sentido y además es significativo.

> names(reg)

[1] "coefficients" "residuals" "effects" "rank"

[5] "fitted.values" "assign" "qr" "df.residual"

[9] "xlevels" "call" "terms" "model"

#para tener en una nueva variable aleatoria llamada res los residuos obtenidos de la regresión, y en una llamada pred las predicciones se hace:

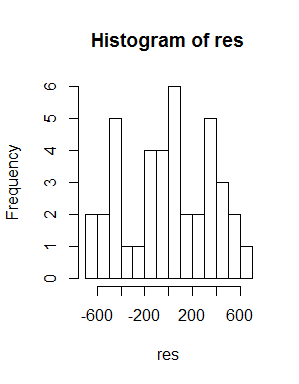
> res<-reg[["residuals"]]

> pred<-reg[["fitted.values"]]

> data2<-data.frame(data,Predicciones=pred,Residuos=res)

#para crear un histograma con 14 grupos se hace:

> hist(res,15)



#podemos observar que el histograma de los residuos es idéntico al anterior.

> mean(res)

[1] -7.771561e-16

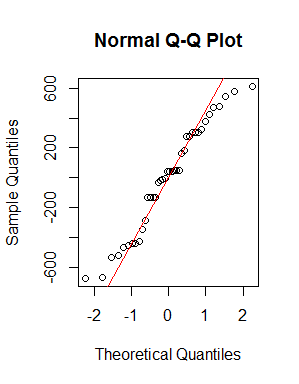
# se espera que los residuos sigan una ley normal con media 0 y varianza constante, en este caso, mean(res)=-7.771561e-16 que es bastante cercano a 0, se debe observar que se iso lo mismo en los datos centrados y que se obtuvo un mean(res)= -8.14071e-15, la diferencia radica en que estamos asumiendo que los residuos tienen media 0, pero en nuestro caso son aproximadamente de media 0 mas no igual, pero aun así ambos valores son muy cercanos a 0.

> #prueba de normalidad

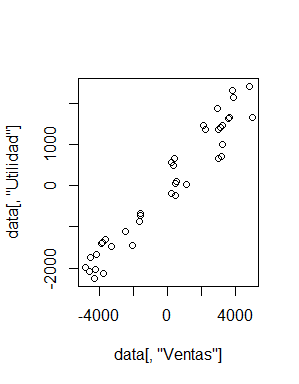
#Aplicamos las siguientes funciones para ver una recta teorica que esperamos siguen los residuos:

> qqnorm(res)

> qqline(res,col="red")



#El grafico de los datos que estamos estudiando seria:



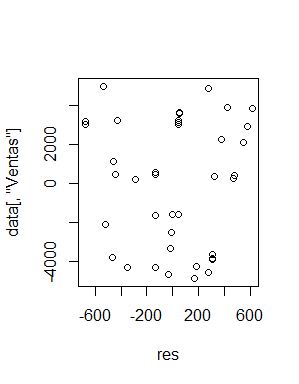
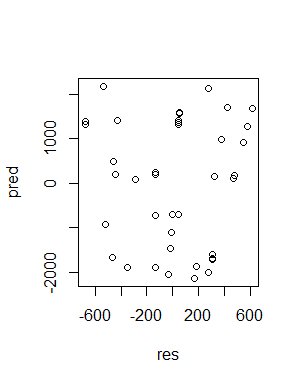
#que no es mas que el mismo grafico que se obtuvo con los datos sin centrar pero cambiando el origen por el punto de las medias de las variables.

#Se usa las siguientes funciones para graficar X e Y (las variables estudiadas) vs los residuos, y así analizar posibles problemas en el modelo, debido a supuestos que no se cumplen.

plot(res,pred)

plot(res, data[,"Ventas"])

#obteniendose:



#lo cual nos dice que no existe evidencia de violación de hipótesis, y que no es necesario el transformar ninguna variable.