Modelación lineal

Cristian Gañan, Valentina Ruiz, Maria Isabel Vasquez, Tatiana Marin, Daniel Marin, Marlon Tejada

Table of Contents

[Punto 1 2](#_Toc17555857)

[Datos 2](#_Toc17555858)

[1.1 3](#_Toc17555859)

[1.2 4](#_Toc17555860)

[1.3 6](#_Toc17555861)

[1.4 7](#_Toc17555862)

[Modelo lineal 8](#_Toc17555863)

[Modelo generalizado 11](#_Toc17555864)

[1.5 14](#_Toc17555865)

[Punto 2 15](#_Toc17555866)

[Datos 16](#_Toc17555867)

[2.1 17](#_Toc17555868)

[2.2 18](#_Toc17555869)

[2.3 21](#_Toc17555870)

[Punto 3 23](#_Toc17555871)

[Datos 23](#_Toc17555872)

[3.1 24](#_Toc17555873)

[3.2 27](#_Toc17555874)

[3.3 30](#_Toc17555875)

[3.4 31](#_Toc17555876)

[Punto 4 34](#_Toc17555877)

[Datos 34](#_Toc17555878)

[4.1 36](#_Toc17555879)

[4.2 41](#_Toc17555880)

[4.3 42](#_Toc17555881)

[Punto 5 45](#_Toc17555882)

# Punto 1

## Datos

Con los siguientes datos:

datos<- read.csv2("Libro1.csv", dec = ".")  
datos

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| x1 | x2 | x3 | y | y2 |
| 44.26 | 65.01 | 3.97 | 361.18 | 76 |
| 77.08 | 70.94 | 3.06 | 1120.07 | 87 |
| 25.99 | 50.93 | 5.50 | 114.67 | 52 |
| 34.98 | 56.18 | 4.56 | 225.14 | 84 |
| 63.93 | 68.16 | 3.31 | 767.82 | 93 |
| 24.93 | 49.02 | 5.66 | 110.05 | 48 |
| 10.06 | 0.81 | 11.58 | 18.34 | 14 |
| 83.51 | 78.31 | 2.96 | 1316.17 | 84 |
| 28.45 | 51.94 | 5.21 | 149.25 | 54 |
| 26.32 | 44.76 | 5.45 | 120.94 | 56 |
| 79.75 | 72.79 | 3.02 | 1198.42 | 86 |
| 41.96 | 63.59 | 4.09 | 323.28 | 84 |
| 75.55 | 70.51 | 3.08 | 1076.15 | 86 |
| 13.15 | 0.99 | 11.60 | 31.35 | 21 |
| 51.09 | 66.12 | 3.70 | 389.76 | 70 |
| 44.22 | 62.18 | 3.97 | 361.53 | 82 |
| 32.54 | 56.97 | 4.77 | 192.72 | 61 |
| 84.80 | 73.29 | 2.95 | 1354.97 | 88 |
| 80.05 | 72.14 | 3.01 | 1208.71 | 84 |
| 27.29 | 50.94 | 5.32 | 133.26 | 48 |
| 64.05 | 70.03 | 3.32 | 771.91 | 61 |
| 96.12 | 75.16 | 2.82 | 1754.01 | 50 |
| 11.20 | 18.03 | 9.30 | 21.23 | 16 |
| 73.47 | 70.98 | 3.12 | 1014.94 | 82 |
| 54.21 | 67.68 | 3.59 | 556.32 | 79 |
| 27.53 | 50.93 | 5.32 | 137.52 | 54 |
| 54.67 | 77.17 | 3.58 | 565.02 | 75 |
| 46.82 | 66.26 | 3.85 | 402.58 | 93 |
| 99.10 | 73.97 | 2.80 | 1866.11 | 23 |
| 11.31 | 6.18 | 10.48 | 22.26 | 18 |

## 1.1

Ajuste un modelo lineal entre , , , .

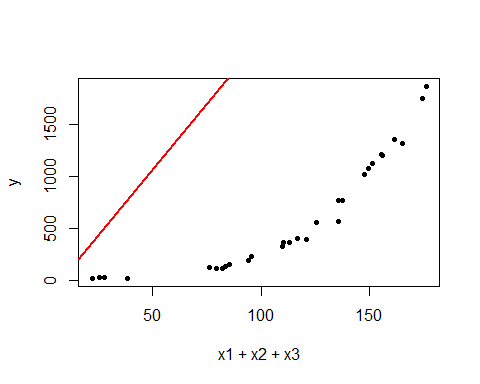
Se hace un modelo de la forma

library(tidyverse)  
library(modelr)  
  
dato<- datos %>% nest()  
modelo<- function(data){  
 lm(y ~ x1 + x2 + x3, data= data)  
}  
  
dato<- dato %>%   
 mutate(model= map(data, modelo))  
  
  
model1<- dato %>%   
 mutate(model1= map(model, broom::glance)) %>%   
 unnest(model1, .drop = TRUE)  
  
model1

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| r.squared | adj.r.squared | sigma | statistic | p.value | df | logLik | AIC | BIC | deviance | df.residual |
| 0.9829898 | 0.9810271 | 75.54716 | 500.8329 | 0 | 4 | -170.1644 | 350.3287 | 357.3347 | 148391.7 | 26 |

Según el r.squared, es un buen modelo de pendiente positiva y una relación de las variables casi perfecta se prodece a graficar para ver tal relación.

with(datos,plot(x= x1+x2+x3, y= y, pch= 20))  
abline(lm(y ~ x1 + x2 + x3, data= datos), col= "red", lwd= 2)



**Figura 1** modelo lineal con los datos

Sin embargo en **figura 1** es notable que el modelo no es bueno, perece ser más una función de tipo polinómica o en su definición mas concreta un modelo no lineal. Aquí se evidencia como los criterios del r.squared se quedan cortos mintiendo acerca de la linealidad del modelo.

## 1.2

Depure su modelo si lo amerita con ayuda de la library(leaps)

library(leaps)  
  
variables<- leaps(x= datos[,1:3], y= datos[,4], method = "r2")  
  
def<- cbind(variables$size,variables$which, variables$r2)  
colnames(def)<- c("variables","x1","x2","x3", "r2")  
as.data.frame(def)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | variables | x1 | x2 | x3 | r2 |
| X1 | 2 | 1 | 0 | 0 | 0.9450508 |
| X1.1 | 2 | 0 | 1 | 0 | 0.4621556 |
| X1.2 | 2 | 0 | 0 | 1 | 0.4380300 |
| X2 | 3 | 1 | 1 | 0 | 0.9829892 |
| X2.1 | 3 | 1 | 0 | 1 | 0.9815204 |
| X2.2 | 3 | 0 | 1 | 1 | 0.4733313 |
| X3 | 4 | 1 | 1 | 1 | 0.9829898 |

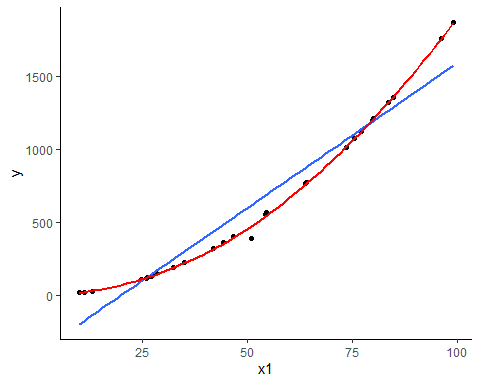
Se observa en la anterior tabla que el modelo con tiene un de pero este mismo con tambien tienen este porcetaje, sin embargo el modelo con solo explica la variablidad en un por lo cual es pertienente quitar las demás variables dejando solo ; el modelo quedaria de la forma: .

dato1<- datos %>% nest()  
modelo<- function(data){  
 lm(y ~ x1, data= data)  
}  
  
dato1<- dato1 %>%   
 mutate(model= map(data, modelo))  
  
  
model2<- dato1 %>%   
 mutate(model2= map(model, broom::tidy)) %>%   
 unnest(model2, .drop = TRUE)  
  
model2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| term | estimate | std.error | statistic | p.value |
| (Intercept) | -395.79470 | 50.8597285 | -7.782084 | 0 |
| x1 | 19.86006 | 0.9050125 | 21.944519 | 0 |

El modelo es altamente significativo, se observa entonces el grafíco para corroborar el ajuste de la linea.

ggplot(data = datos, mapping = aes(x = x1, y= y)) +  
 geom\_point() + theme\_classic() + geom\_smooth(method = "lm", se= FALSE) + geom\_smooth(se= FALSE, color= "red")



**figura 2** Modelo depurado

El modelo se ha ajustado mejor a los datos, sin embargo, la linea no describe del todo la tendencia (line azul) de los datos que al paracer se comportaran como una curva (linea roja) y no una linea recta

## 1.3

Depure con ayuda de la library(car) y analice ambos resultados

library(car)  
s<- as.data.frame(vif(lm(y ~ x1 + x2 + x3, data= datos)))  
  
colnames(s)<- c("VIF con todas las variables")  
s

|  |  |
| --- | --- |
|  | VIF con todas las variables |
| x1 | 3.047195 |
| x2 | 74.005135 |
| x3 | 68.598739 |

s2<- as.data.frame(vif(lm(y ~ x1 + x2, data= datos)))  
colnames(s2)<- c("VIF con x1 y x2")  
s2

|  |  |
| --- | --- |
|  | VIF con x1 y x2 |
| x1 | 2.982769 |
| x2 | 2.982769 |

s3<- as.data.frame(vif(lm(y ~ x1 + x3, data= datos)))  
colnames(s3)<- c("VIF con x1 y x3")  
s3

|  |  |
| --- | --- |
|  | VIF con x1 y x3 |
| x1 | 2.764865 |
| x3 | 2.764865 |

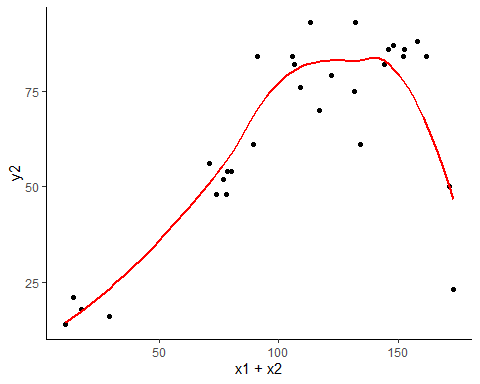
Al correr el de modelo con y ó y , el resultado el es mismo con leaps() pues hay que recordar que también los dos modelos planteados con estas variables eran viables, es decir, las dos formas de depuración dan lo mismo sin embargo, con leaps() al menos para este ejercicio fue mas practico pues los resultados se obtuvieron de formas mas inmediata relacionando todas las posibles interacciones.

## 1.4

Busque un buen modelo entre las variables , y . Justifique sus respuestas.

Se observa la tendencia de los datos en un grafíca para determinar como podría ir el modelo.

ggplot(data = datos, mapping = aes(x= x1+x2, y= y2)) +  
 geom\_point() + geom\_smooth(se= FALSE, color= "red") + theme\_classic()



**figura 1** Tendencía de datos

A primera vista parece buena idea hacer un modelo lineal entre las variables, sin embargo siguiendo la tendencia que muestran los datos también podría tratarse de un modelo no lineal; ya que los datos tienen forma de campana pero es muy difícil montarlo puesto que no se sabe de donde provienen los datos ni se conoce le fenómeno para darle respuesta de esta forma; se abandona esta posibilidad entonces, la otra alternativa que se puede evaluar es la de un modelo lineal generalizado pues si se detalla la variable de factible suponer que se trata de conteos pues son números enteros que podrían reflejar las repeticiones del experimento. Siguiendo lo analizado se procede a evaluar los dos modelos lineal y general para determinar el mas apropiado.

### Modelo lineal

dato2<- datos %>% nest()  
modelo3<- function(data){  
 lm(y2 ~ x1+x2, data= data)}  
  
dato2<- dato2 %>%   
 mutate(model= map(data, modelo3))  
  
  
model4<- dato2 %>%   
 mutate(model4= map(model, broom::tidy)) %>% unnest(model4, .drop = TRUE)  
  
model4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| term | estimate | std.error | statistic | p.value |
| (Intercept) | 11.5671695 | 7.9003172 | 1.464140 | 0.1547026 |
| x1 | -0.2901041 | 0.1833138 | -1.582554 | 0.1251679 |
| x2 | 1.1714501 | 0.2220131 | 5.276491 | 0.0000145 |

model5<- dato2 %>%   
 mutate(model5= map(model, broom::glance)) %>% unnest(model5, .drop = TRUE)  
  
model5

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| r.squared | adj.r.squared | sigma | statistic | p.value | df | logLik | AIC | BIC | deviance | df.residual |
| 0.6488958 | 0.6228881 | 15.34557 | 24.95012 | 7e-07 | 3 | -122.9125 | 253.8251 | 259.4299 | 6358.134 | 27 |

El modelo resulta significativo, su r.squared es aceptable, sin embargo, el termino no es siginificativo, es pertinente entonces hacer una depuración para observar si las dos variables son necesarias en el modelo.

variable<- leaps(x= datos[,1:2], y= datos[,5], method = "r2")  
  
def<- cbind(variable$size,variable$which, variable$r2)  
colnames(def)<- c("variables","x1","x2", "r2")  
as.data.frame(def)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | variables | x1 | x2 | r2 |
| X1 | 2 | 0 | 1 | 0.6163279 |
| X1.1 | 2 | 1 | 0 | 0.2868508 |
| X2 | 3 | 1 | 1 | 0.6488958 |

Según los resultados anteriores, es pertinente eliminar la variable pues solo explica el de la variabilidad, mientras explica el ; teniendo en cuenta que incluyendo las dos variables el es de entonces es buena idea eliminar

Se vuelve a correr el modelo sin

dato2<- datos %>% nest()  
modelo3<- function(data){  
 lm(y2 ~ x2, data= data)}  
  
dato2<- dato2 %>%   
 mutate(model= map(data, modelo3))  
  
  
model4<- dato2 %>%   
 mutate(model4= map(model, broom::tidy)) %>% unnest(model4, .drop = TRUE)  
  
model4

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| term | estimate | std.error | statistic | p.value |
| (Intercept) | 13.4257625 | 8.0196801 | 1.674102 | 0.1052469 |
| x2 | 0.8849904 | 0.1319574 | 6.706637 | 0.0000003 |

model5<- dato2 %>%   
 mutate(model5= map(model, broom::glance)) %>% unnest(model5, .drop = TRUE)  
  
model5

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| r.squared | adj.r.squared | sigma | statistic | p.value | df | logLik | AIC | BIC | deviance | df.residual |
| 0.6163279 | 0.6026253 | 15.75244 | 44.97899 | 3e-07 | 2 | -124.2431 | 254.4863 | 258.6898 | 6947.905 | 28 |

Es viable ahora evaluar los supuestos del modelo y comprobar su viabilidad.

anova(lm(y2 ~ x2, data= datos))

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Df | Sum Sq | Mean Sq | F value | Pr(>F) |
| x2 | 1 | 11161.062 | 11161.0615 | 44.97899 | 3e-07 |
| Residuals | 28 | 6947.905 | 248.1395 | NA | NA |

res<- resid(lm(y2 ~ x2, data= datos))  
  
shapiro.test(res)

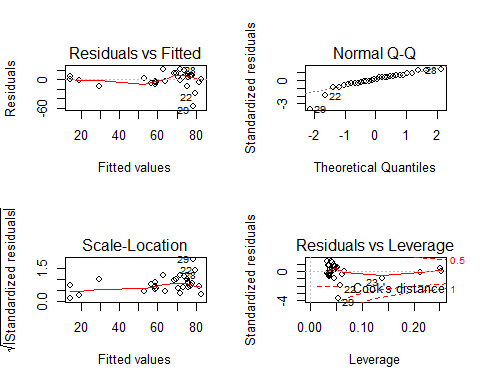
##   
## Shapiro-Wilk normality test  
##   
## data: res  
## W = 0.86939, p-value = 0.001624

library(lmtest)  
  
bptest(lm(y2 ~ x2, data= datos))

##   
## studentized Breusch-Pagan test  
##   
## data: lm(y2 ~ x2, data = datos)  
## BP = 1.4192, df = 1, p-value = 0.2335

Al hacer los test de normalidad y homocedasticidad, su puede concluir con un nivel , que hay homocesaticidad sin embargo no hay normalidad en los residuales, para corroborar se hacen los graficos pertinentes

par(mfrow= c(2,2))  
plot(lm(y2 ~ x2, data= datos))



**Figura 2** Gráfico de supuestos

En las gráficas anteriores es evidente observar, puntos palanca en el modelo, ¿Es posible esto? Dando una mirada hacia atrás en **figura 1** se puede evidenciar la tendencia de datos en forma de campana, es decir, los datos que tienen valores bajos son posibles, no es descabellado datos aquí, además, no se sabe proveniencia de los datos y así es difícil establecer el comportamiento del fenómeno, teniendo en cuenta estos factores, se decide pertinente, hacer el modelo generalizado, y analizarlo así poder concluir si este resultado es causa un conteo posible de los datos **(ver sección siguiente)**

### Modelo generalizado

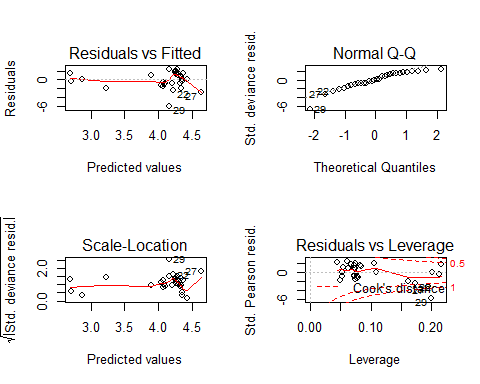
dato3<- datos %>% nest()  
modelo5<- function(data){  
 glm(y2 ~ x1 + x2, data= data, family = poisson)}  
  
dato3<- dato3 %>%   
 mutate(model= map(data, modelo5))  
  
  
model5<- dato3 %>%   
 mutate(model5= map(model, broom::tidy)) %>% unnest(model5, .drop = TRUE)  
  
model5

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| term | estimate | std.error | statistic | p.value |
| (Intercept) | 2.7746810 | 0.1133533 | 24.478171 | 0e+00 |
| x1 | -0.0084487 | 0.0015896 | -5.315079 | 1e-07 |
| x2 | 0.0301151 | 0.0026367 | 11.421414 | 0e+00 |

model6<- dato3 %>%   
 mutate(model6= map(model, broom::glance)) %>% unnest(model6, .drop = TRUE)  
  
model6

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| null.deviance | df.null | logLik | AIC | BIC | deviance | df.residual |
| 340.4805 | 29 | -136.7539 | 279.5078 | 283.7114 | 97.2317 | 27 |

par(mfrow= c(2,2))  
plot(glm(y2 ~ x1 + x2, data= datos, family = poisson))



**Figura 1** Gráfico de supuestos modelo generalizado

La descripción de este nuevo modelo explica mejor los datos **Figura 1**, además, las variables y son significativas, sin embargo, el son mayores en este caso al anterior modelo, pero si se comparan las de los dos casos, es evidente observar que el mejor modelo que ajusta mejor a los datos es el general, la para el modelo lineal es muy elevada lo que hace desconfiar de su viabilidad.

Según la **Figura 1** es que probable algunos datos sean atípicos en el modelo se explora esta posibilidad, dando un resultado positivo, los datos y resultaron ser puntos palanca, se corre un nuevo modelo sin estos datos:

m<- glm(y2 ~ x1 + x2, data= datos, family = poisson)  
influence.measures(m)

fil<- c(7,29,30)  
dato3<- datos %>% slice(-fil) %>% nest()  
modelo5<- function(data){  
 glm(y2 ~ x1 + x2, data= data, family = poisson)}  
  
dato3<- dato3 %>%   
 mutate(model= map(data, modelo5))  
  
  
model5<- dato3 %>%   
 mutate(model5= map(model, broom::tidy)) %>% unnest(model5, .drop = TRUE)  
  
model5

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| term | estimate | std.error | statistic | p.value |
| (Intercept) | 2.8882343 | 0.1431557 | 20.175472 | 0.0000000 |
| x1 | -0.0041642 | 0.0018156 | -2.293526 | 0.0218177 |
| x2 | 0.0250416 | 0.0032920 | 7.606906 | 0.0000000 |

model6<- dato3 %>%   
 mutate(model6= map(model, broom::glance)) %>% unnest(model6, .drop = TRUE)  
  
model6

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| null.deviance | df.null | logLik | AIC | BIC | deviance | df.residual |
| 192.8691 | 26 | -108.4132 | 222.8264 | 226.714 | 54.75729 | 24 |

Se nota entonces como los datos ( y ) si afectaban el modelo, tanto la rebajaron por lo cual , este modelo es el que mejor se ajusta a los datos; para concluir el modelo resultó siendo mejor ajustado por el modelo generalizado, el modelo lineal simple no lo ajusta bien, pues la de este es muy elevada comparada con la del .

## 1.5

Corra un modelo no lineal, similar al de la forma grafíquelo, analícelo profusamente y mire sus predicciones.

Para hacer el modelo, se necesitan unos valores de partida y como en este caso no se conocen se hizo uso de el algoritmo brute force para encontrar estos parametros.

library(nls2)  
fact<- function(x1,x3, a, b) {  
 (x1^a)\*(x3^b)  
}  
  
part1 <- expand.grid(a = seq(0, 10, length.out = 20),b= seq(0,0.9,length.out = 20))  
  
mnls<- nls2(y2 ~ fact(x1,x3,a,b), start= part1, data= datos, algorithm = "brute-force")

Hecho esto, se procede a correr con los parametros hallados anteriormente:

dato1<- datos %>% nest()  
modelo2<- function(data) {nls(y2 ~ fact(x1, x3, a , b), start = c(a= 1.05263, b= 0.00000), data = data)}  
  
dato1<- dato1 %>%   
 mutate(model= map(data, modelo2))  
  
  
model3<- dato1 %>%   
 mutate(model3= map(model, broom::tidy)) %>%   
 unnest(model3, .drop = TRUE)  
  
model3

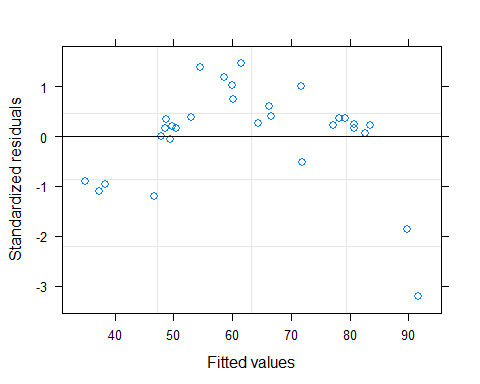
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| term | estimate | std.error | statistic | p.value |
| a | 0.8259603 | 0.0405688 | 20.359483 | 0.0e+00 |
| b | 0.7001545 | 0.1222894 | 5.725389 | 3.8e-06 |

model3<- dato1 %>%   
 mutate(model3= map(model, broom::glance)) %>%   
 unnest(model3, .drop = TRUE)  
  
model3

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| sigma | isConv | finTol | logLik | AIC | BIC | deviance | df.residual |
| 21.30135 | TRUE | 8.8e-06 | -133.2964 | 272.5928 | 276.7963 | 12704.93 | 28 |

Los parametros son significativos, sin embargo la devience esta un poco elevada, los datos además los residuales no son normales (ver prueba shapiro.test) y al paracer los residuales no son homocedasticos pues en la**Figura 1** se ve claramente un patron lo que evidenciaría este problema.

mon<- nls(y2 ~ x1^a\*x3^b, start = c(a= 1.05263, b= 0.00000), data = datos)  
  
plot(mon)



**Figura 1** Modelo no lineal, residuales

shapiro.test(residuals(mon))

##   
## Shapiro-Wilk normality test  
##   
## data: residuals(mon)  
## W = 0.87397, p-value = 0.002055

# Punto 2

Un muestreo del número de postes de Acacia mangiun y edad de plantación mostró la siguiente información en 3 localidades.

## Datos

punto2<- read.csv2("punto2.csv", dec = ".")  
  
punto2<- punto2 %>% rename(npostes= ï..nupostes)  
  
punto2

|  |  |  |
| --- | --- | --- |
| npostes | edad | localidad |
| 8 | 8.72 | cauca |
| 9 | 8.93 | cauca |
| 11 | 12.00 | cauca |
| 12 | 14.40 | cauca |
| 14 | 16.80 | cauca |
| 20 | 21.20 | cauca |
| 5 | 6.20 | cauca |
| 13 | 13.80 | cauca |
| 4 | 5.00 | Santa Elena |
| 6 | 4.45 | Santa Elena |
| 10 | 9.32 | Santa Elena |
| 14 | 14.90 | Santa Elena |
| 16 | 17.50 | Santa Elena |
| 21 | 23.00 | Santa Elena |
| 11 | 10.10 | Santa Elena |
| 10 | 7.50 | piedecuesta |
| 11 | 9.30 | piedecuesta |
| 12 | 12.20 | piedecuesta |
| 14 | 14.85 | piedecuesta |
| 17 | 15.30 | piedecuesta |
| 6 | 7.10 | piedecuesta |
| 12 | 13.50 | piedecuesta |
| 12 | 14.40 | Riosucio |
| 12 | 16.80 | Riosucio |
| 21 | 21.20 | Riosucio |
| 14 | 14.90 | Riosucio |
| 16 | 17.50 | Riosucio |
| 21 | 23.00 | Riosucio |
| 10 | 8.72 | Riosucio |
| 11 | 8.93 | Riosucio |

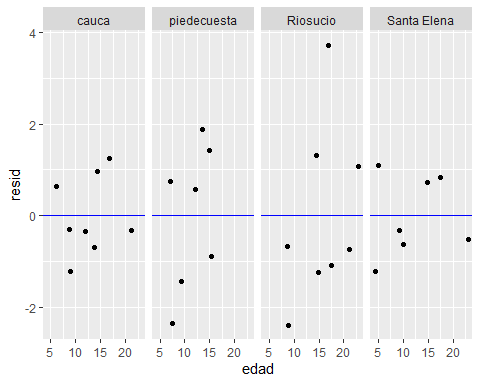
## 2.1

Evalúe un modelo polinomial máximo de grado 2 para cada región y depúrelo si se amerita

punto<- punto2 %>% group\_by(localidad) %>% nest()  
  
pol<- function(data) {  
 lm(edad~npostes+I(npostes^2), data = data)  
}  
  
punto<- punto %>%   
 mutate(model= map(data, pol))  
  
pun<- punto %>%   
 mutate(resids= map2(data, model, add\_residuals))  
  
resids<- unnest(pun, resids)  
   
mp<- punto %>%   
 mutate(pol= map(model, broom::tidy)) %>%   
 unnest(pol, .drop = TRUE)  
  
mp

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| localidad | term | estimate | std.error | statistic | p.value |
| cauca | (Intercept) | -0.5351812 | 2.4531987 | -0.2181565 | 0.8359314 |
| cauca | npostes | 1.2559741 | 0.4102526 | 3.0614653 | 0.0280551 |
| cauca | I(npostes^2) | -0.0076818 | 0.0159762 | -0.4808275 | 0.6509412 |
| Santa Elena | (Intercept) | 0.7918828 | 1.9390449 | 0.4083881 | 0.7039054 |
| Santa Elena | npostes | 0.7052541 | 0.3477876 | 2.0278297 | 0.1124902 |
| Santa Elena | I(npostes^2) | 0.0179211 | 0.0138038 | 1.2982691 | 0.2639900 |
| piedecuesta | (Intercept) | 1.2982528 | 7.3626403 | 0.1763298 | 0.8686024 |
| piedecuesta | npostes | 0.8224839 | 1.3041592 | 0.6306622 | 0.5624986 |
| piedecuesta | I(npostes^2) | 0.0031248 | 0.0559008 | 0.0558991 | 0.9581029 |
| Riosucio | (Intercept) | -18.3623723 | 17.2927951 | -1.0618510 | 0.3368768 |
| Riosucio | npostes | 3.5554053 | 2.3147932 | 1.5359494 | 0.1851484 |
| Riosucio | I(npostes^2) | -0.0779346 | 0.0724969 | -1.0750051 | 0.3314878 |

resids %>% ggplot(aes(x= edad, y= resid, group= localidad)) +  
 geom\_point() + facet\_grid(~localidad) + geom\_abline(slope = 0, color= "blue")



normal<- function(data){  
 r<- shapiro.test(data)  
 return(r$p.value)  
}  
   
resids %>% group\_by(localidad) %>% summarise(normalidad= normal(resid))

|  |  |
| --- | --- |
| localidad | normalidad |
| cauca | 0.4910310 |
| piedecuesta | 0.6374886 |
| Riosucio | 0.3777145 |
| Santa Elena | 0.4014940 |

Por tratarse de un muestreo del número de postes de Acacia mangium se utilizó un modelo lineal. En todos los modelos se observó una significancia intercepto como en el numpostes. Se modeló la variable independiente como (). Y también se observó que los residuales se distribuían en un rango que varía entre cero y 1.5

## 2.2

Busque si con todos los datos obtiene un mejor modelo

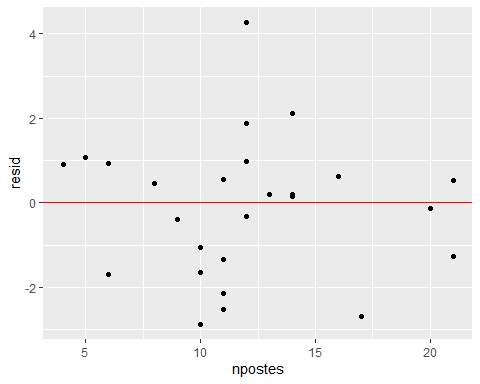
punto<- punto2 %>% nest()  
  
gmol<- function(data){  
 lm(edad~npostes + I(npostes^2), data = data) }  
  
  
punto<- punto %>%   
 mutate(model= map(data, gmol))  
  
pun<- punto %>%   
 mutate(resids= map2(data, model,add\_predictions), resids1= map2(data, model, add\_residuals))  
  
resids<- unnest(pun, resids, resids1)  
   
mp<- punto %>%   
 mutate(pol= map(model, broom::tidy)) %>%   
 unnest(pol, .drop = TRUE)  
mp

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| term | estimate | std.error | statistic | p.value |
| (Intercept) | -0.0009583 | 1.9887963 | -0.0004818 | 0.9996191 |
| npostes | 1.0095482 | 0.3162201 | 3.1925494 | 0.0035655 |
| I(npostes^2) | 0.0028607 | 0.0118019 | 0.2423940 | 0.8103067 |

mp2<- punto %>%   
 mutate(pol= map(model, broom::glance)) %>%   
 unnest(pol, .drop = TRUE)  
mp2

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| r.squared | adj.r.squared | sigma | statistic | p.value | df | logLik | AIC | BIC | deviance | df.residual |
| 0.9069554 | 0.9000632 | 1.639474 | 131.5917 | 0 | 3 | -55.81902 | 119.638 | 125.2428 | 72.57267 | 27 |

resids %>% ggplot(aes(x= npostes, y= resid)) +  
 geom\_point() + geom\_abline(slope = 0, color= "red")



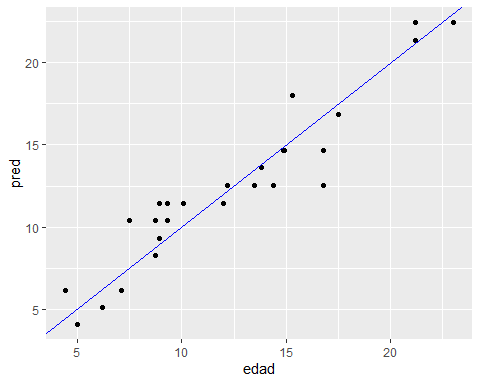
anova(lm(edad~npostes + I(npostes^2), data = punto2))

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Df | Sum Sq | Mean Sq | F value | Pr(>F) |
| npostes | 1 | 707.2463910 | 707.2463910 | 263.1245702 | 0.0000000 |
| I(npostes^2) | 1 | 0.1579258 | 0.1579258 | 0.0587549 | 0.8103067 |
| Residuals | 27 | 72.5726698 | 2.6878767 | NA | NA |

resids %>% summarise(normalidad= normal(resid))

|  |
| --- |
| normalidad |
| 0.2443377 |

resids %>% ggplot(aes(x= edad, y= pred)) +  
 geom\_point() + geom\_abline(slope = 1, color= "blue")



En base en los resultados obtenidos, se vio que era más conveniente realizar un modelo donde estén las tres localidades para obtener un mejor ajuste de este, ya que se logra una mayor significancia en nupostes. En este modelo se pudo comprobar la normalidad en los datos con una prueba de shapiro wilks.

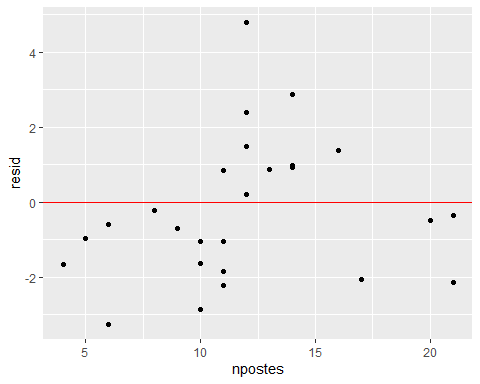
## 2.3

Explore una regresión no lineal

punto<- punto2 %>% nest()  
  
mln<- function(data){  
 nls(edad~a\*exp(npostes\*b), data=data, start = list(a=4.94, b=0.074))  
 }  
  
  
punto<- punto %>%   
 mutate(model= map(data, mln))  
  
pun<- punto %>%   
 mutate(resids= map2(data, model,add\_residuals))  
  
resids<- unnest(pun, resids)  
   
mp<- punto %>%   
 mutate(pol= map(model, broom::tidy)) %>%   
 unnest(pol, .drop = TRUE)  
mp

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| term | estimate | std.error | statistic | p.value |
| a | 4.9433126 | 0.4319762 | 11.44348 | 0 |
| b | 0.0739047 | 0.0053876 | 13.71767 | 0 |

resids %>% ggplot(aes(x= npostes, y= resid)) +  
 geom\_point() + geom\_abline(slope = 0, color= "red")



resids %>% summarise(normalidad= normal(resid))

|  |
| --- |
| normalidad |
| 0.6235991 |

En el modelo no lineal presenta una alta significancia en los parámetros, se puede observar que en ambos modelos hay un comportamiento similar cuando hay una estimación de datos. Por lo que se opta usar un modelo lineal simple por su simplicidad al momento de trabajar con este tipo de datos ya que es un modelo donde hay una mayor facilidad de moldear.

# Punto 3

A una especie de cuervos se les midió su edad, y longitud de ala, en varios zoológicos del mundo, con el fin de estudiar el crecimiento de las alas con los siguientes resultados de edad(ed) y longitud de alas(la):

## Datos

punto3<- read.csv2("punto3.csv", dec = ".")  
punto3<- punto3 %>% select(ed,la)  
punto3

|  |  |
| --- | --- |
| ed | la |
| 0.9 | 3.2 |
| 0.3 | 2.2 |
| 7.0 | 28.1 |
| 0.3 | 5.2 |
| 0.5 | 2.0 |
| 0.7 | 11.2 |
| 0.8 | 9.2 |
| 1.1 | 13.4 |
| 1.2 | 11.9 |
| 1.4 | 12.0 |
| 1.7 | 12.9 |
| 1.8 | 17.0 |
| 2.2 | 18.6 |
| 2.2 | 19.7 |
| 2.4 | 12.4 |
| 2.8 | 17.2 |
| 2.9 | 21.3 |
| 3.2 | 19.6 |
| 3.2 | 21.2 |
| 3.4 | 22.5 |
| 3.7 | 20.5 |
| 3.9 | 23.0 |
| 4.1 | 22.5 |
| 4.3 | 16.2 |
| 4.5 | 24.7 |
| 4.6 | 23.2 |
| 4.8 | 22.9 |
| 5.1 | 21.1 |
| 5.3 | 20.3 |
| 5.5 | 19.6 |
| 5.6 | 23.7 |
| 6.0 | 25.4 |
| 6.0 | 26.8 |
| 6.4 | 26.9 |
| 6.5 | 27.2 |
| 6.7 | 27.5 |
| 7.0 | 27.8 |
| 7.2 | 27.9 |
| 7.4 | 28.0 |
| 7.4 | 28.1 |
| 7.6 | 28.2 |
| 7.9 | 28.3 |
| 8.1 | 28.4 |
| 8.3 | 28.5 |
| 8.5 | 28.6 |
| 8.8 | 28.7 |
| 8.9 | 28.8 |
| 9.2 | 28.9 |
| 9.2 | 29.0 |
| 9.6 | 29.6 |

## 3.1

Haga propuestas de modelación lineal al respecto.

Se escogió la edad de los cuervos como la variable independiente para estimar los valores de la longitud de las alas de estos mismos a una edad determinada, porque los cuervos, una vez que están en los zoologicos se les registra en un ficha de ingreso el estado general del animal y a través de procesos en laboratorio se le estima la edad que tiene, además a estos se les asignan un identificador único numérico lo que permite que siempre se sepa cuanta edad tienen, además aquellos cuervos que nacen en cautiverio siempre se les lleva el registro de su edad desde el día cero hasta que mueren.

punto<- punto3 %>% nest()  
  
lm3<- function(data){  
 lm(la ~ ed, data = data)  
 }  
  
  
punto<- punto %>%   
 mutate(model= map(data, lm3))  
  
pun<- punto %>%   
 mutate(resids= map2(data, model,add\_residuals), resids2= map2(data, model, add\_predictions))  
  
resids<- unnest(pun, resids, resids2)  
   
mp<- punto %>%   
 mutate(pol= map(model, broom::tidy)) %>%   
 unnest(pol, .drop = TRUE)  
mp

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| term | estimate | std.error | statistic | p.value |
| (Intercept) | 9.071702 | 0.9183881 | 9.877852 | 0 |
| ed | 2.509512 | 0.1662574 | 15.094141 | 0 |

anova(lm(la ~ ed, data = punto3))

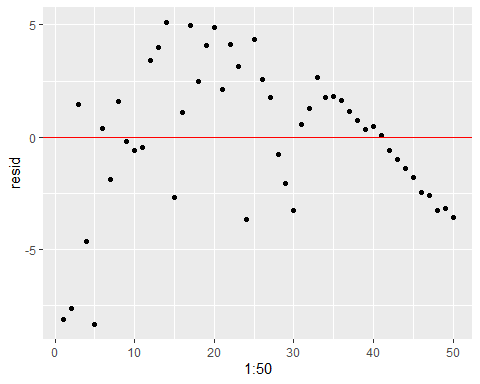
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Df | Sum Sq | Mean Sq | F value | Pr(>F) |
| ed | 1 | 2467.6582 | 2467.65821 | 227.8331 | 0 |
| Residuals | 48 | 519.8876 | 10.83099 | NA | NA |

Al crear un modelo lineal de la forma , se encontró que tanto el intercepto y la pendiente son significativos, además el es indicando que en aproximadamente un la variable independiente explica la variabilidad de la variable dependiente; al hacer la prueba de Shapiro Wilk, arroja un p-value dando a entender que no se presenta falta de normalidad, sin embargo al hacer la gráfica de los residuales estandarizados **Figura 1** se observa una ligero patrón de función trigonometrica despues de la observación lo que lleva a pensar que estos residuales se comportan de forma heterocedastica.

resids %>% summarise(normalidad= normal(resid))

|  |
| --- |
| normalidad |
| 0.0524704 |

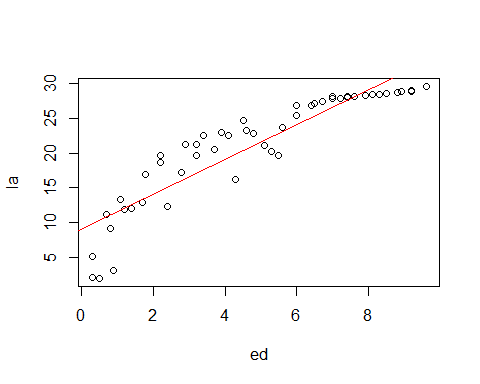
resids %>% ggplot(aes(x= 1:50, y= resid)) +  
 geom\_point() + geom\_abline(slope = 0, color= "red")



**Figura 1** residuales

Se puede ver claramente que el modelo lineal es un mal predictor de la longitud de las alas de los cuervos, ya que tiende a sobreestimar en 19.67% aproximadamente con una desviación estandar (o incertidumbre del 84.97%), lo cual significa que los predichos del modelo lineal oscilan individualmente entre una sobreestimación de 65.3% y 104.64%, además, el cuadrado medio del error (mean sq)es bastante elevado, se encuentra que hay errores que sobreestiman en un 416.32% aproximadamente y otros errores que subestiman en un -25.93% por lo cual se concluye que este es un mal modelo para explicar la longitud de las alas en función de la edad, además, al hacer la línea del modelo lineal, la cual predice la longitud de las alas en función de la edad de los cuervos, esta no pasa cerca de la gran mayoría de longitudes de alas verdaderos **figura 2**.

with(punto3 ,plot(x= ed, y= la))  
abline(lm(la ~ ed, data = punto3), col= "red")

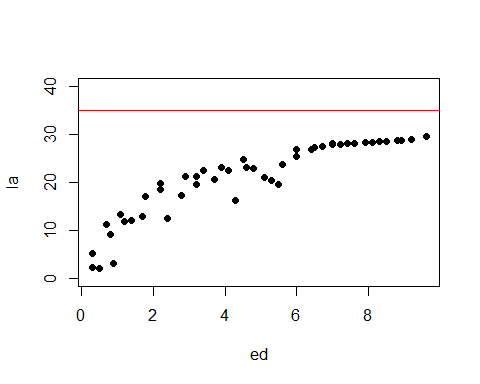


**Figura 2** Modelo lineal

## 3.2

Un estadístico cree que se puede modelar no linealmente de la siguiente forma: . Por la dificultad de encontrar los valores de partida, debe graficar primero edad vs longitud de ala, para que haga sus propuestas.

with(punto3, plot(ed, la, pch= 16, ylim= c(0,40)))  
abline(c(35,0), col= "red")



**Figura 3** Tendencia de datos

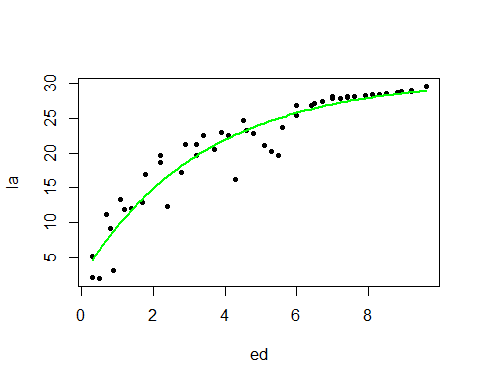
Dado que al parecer el modelo lineal no es el mejor predictor para las longitudes de las alas, se recurre a realizar un modelo No lineal el cual ha sido propuesto para este punto del taller, este modelo tiene tres parámetros (a,b y c), y para hallarlos es necesario graficar la longitud en función de la edad para observar la tendencia de los datos (**Figura3**) . Con la formula propuesta por el estadístico y analizando que cuando X tienda al infinito, la formula queda reducida a: , se encontró que la asíntota al parecer tiene valor aproximado a , después de tomar al parámetro : , se despejan los otros dos parámetros con ayuda de las fórmulas propuestas en este punto del taller, encontrando que , y .

punto<- punto3 %>% nest()  
  
nlm<- function(data){  
 nls(la~a-b\*exp(-c\*ed), data = data,  
 start = list(a=31, b=27.8, c=0.29))  
 }  
  
  
punto<- punto %>%   
 mutate(model= map(data, nlm))  
  
pun<- punto %>%   
 mutate(resids= map2(data, model,add\_residuals))  
  
resids<- unnest(pun, resids)  
   
mp<- punto %>%   
 mutate(pol= map(model, broom::tidy)) %>%   
 unnest(pol, .drop = TRUE)  
mp

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| term | estimate | std.error | statistic | p.value |
| a | 30.5687088 | 1.3969511 | 21.882448 | 0e+00 |
| b | 28.4486406 | 1.3209039 | 21.537253 | 0e+00 |
| c | 0.2973034 | 0.0460924 | 6.450163 | 1e-07 |

Luego de encontrar estos tres parámetros se sigue con la creación del modelo No lineal en donde se observa que todos sus parámetros resultan significativos después de cuatro iteraciones la formula converge mostrando que los tres parámetros son significativos. El modelo No Lineal es un mejor predictor de la longitud de las alas, porque este tan sólo tiende a sobrestimar en promedio un con una desviación estándar (o incertidumbre de ), lo cual significa que los predichos del modelo No Lineal oscilan individualmente entre una sobrestimación de 33.9% y 50.58 mientras que en el modelo lineal la sobrestimación oscilan entre el 65.3% y el ; al observar la **Figura 4** se puede notar claramente que con el modelo No Lineal se estiman mucho mejor la Longitud de las alas.

with(punto3 ,plot(ed,la,cex.main=1.85,cex.sub=2.5, pch= 20))  
x <- seq(min(punto3$ed), max(punto3$ed), 0.1)  
y\_lm <- predict(nls(la~a-b\*exp(-c\*ed), data = punto3,  
 start = list(a=31, b=27.8, c=0.29)), list(ed = x))  
lines(x, y\_lm,col="green",lwd=2)

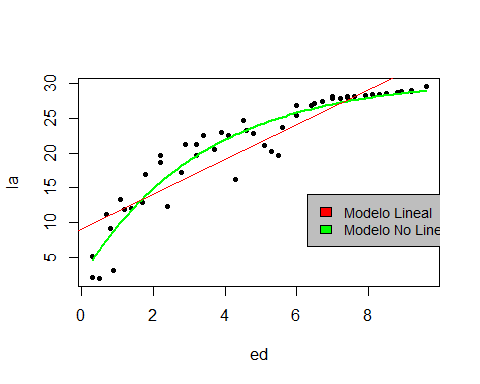


**Figura 4** Modelo no lineal

## 3.3

grafique y compare 3.1 y 3.2

with(punto3 ,plot(ed,la,cex.main=1.85,cex.sub=2.5, pch= 20))  
x <- seq(min(punto3$ed), max(punto3$ed), 0.1)  
y\_lm <- predict(nls(la~a-b\*exp(-c\*ed), data = punto3,  
 start = list(a=31, b=27.8, c=0.29)), list(ed = x))  
lines(x, y\_lm,col="green",lwd=2)  
abline(lm(la ~ ed, data = punto3), col= "red")  
legend(x=6.3,y=14,legend=c("Modelo Lineal","Modelo No Lineal"),  
 fill=c("red","green"),cex=0.9,text.font=0.2, bg='grey')



**Figura 5** Modelo lineal y no lineal

Al hacer la gráfica que muestra las dos lineal de tendencia que predicen el Modelo Lineal, y El No Lineal **Figura 5**, se puede observar claramente como los valores predichos por parte del Modelo No Lineal se ajustan mucho mejor a los verdaderos valores de las longitudes de las alas, a comparación del modelo lineal.

## 3.4

Use force brute, para ajustar su modelo.

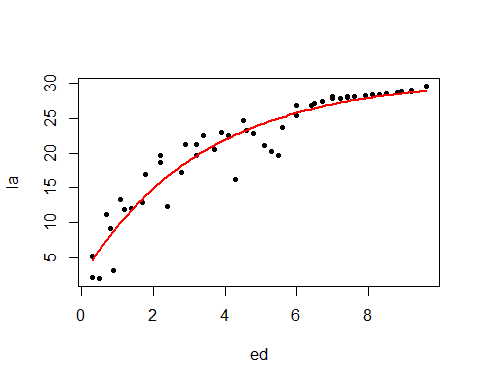
part<-expand.grid(a=seq(2,40,length.out = 20), b=seq(28,35,length.out = 20),  
 c=seq(0,1,length.out = 20))  
modc<-nls2(la~a-b\*exp(-c\*ed), data = punto3, start = part, algorithm = "brute-force")

punto<- punto3 %>% nest()  
  
nlm<- function(data){  
 nls(la~a-b\*exp(-c\*ed), data = data,  
 start = list(a= 30.0000, b= 28.3684, c= 0.3158))  
 }  
  
  
punto<- punto %>%   
 mutate(model= map(data, nlm))  
  
pun<- punto %>%   
 mutate(resids= map2(data, model,add\_residuals))  
  
resids<- unnest(pun, resids)  
   
mp<- punto %>%   
 mutate(pol= map(model, broom::tidy)) %>%   
 unnest(pol, .drop = TRUE)  
mp

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| term | estimate | std.error | statistic | p.value |
| a | 30.5685873 | 1.3969187 | 21.882868 | 0e+00 |
| b | 28.4486155 | 1.3208950 | 21.537378 | 0e+00 |
| c | 0.2973077 | 0.0460926 | 6.450225 | 1e-07 |

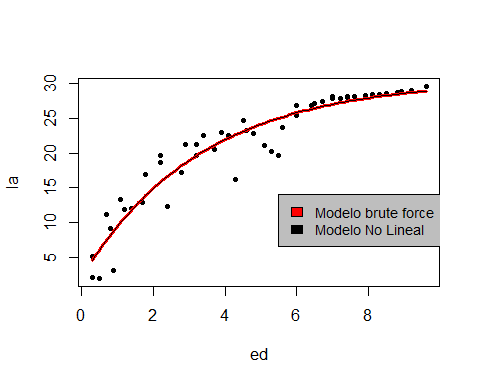
Con las nuevas estimaciones de los valores de partida ( y ) para el Modelo No Lineal con brute force, se encuentra que sus tres parámetros son significativos después de 8000 iteraciones, al hacer la tabla de cada uno de los porcentajes de errores entre la longitud del ala predicha y la verdadera longitud se encuentra que el error o sesgo disminuyo un poco y paso de sobrestimar el a y la desviación estándar pasó de al ; al observar la **Figura 6**se puede notar claramente que el Modelo con brute force se estiman mucho mejor la longitud de las alas a comparación de las estimaciones hechas por el modelo lineal, sin embargo no se encuentran diferencias significativas al graficar los dos modelos No lineales, prácticamente son el mismo modelo **Figura 7**.

with(punto3 ,plot(ed,la,cex.main=1.85,cex.sub=2.5, pch= 20))  
x <- seq(min(punto3$ed), max(punto3$ed), 0.1)  
y\_lm <- predict(nls(la~a-b\*exp(-c\*ed), data = punto3,  
 start = list(a=30.000, b=28.3684, c=0.3158)), list(ed = x))  
lines(x, y\_lm,col="red",lwd=2)



**Figura 6** Modelo con brute force

with(punto3 ,plot(ed,la,cex.main=1.85,cex.sub=2.5, pch= 20))  
x <- seq(min(punto3$ed), max(punto3$ed), 0.1)  
y\_lm <- predict(nls(la~a-b\*exp(-c\*ed), data = punto3,  
 start = list(a=30.000, b=28.3684, c=0.3158)), list(ed = x))  
y\_lm2<- predict(nls(la~a-b\*exp(-c\*ed), data = punto3,  
 start = list(a=31, b=27.8, c=0.29)), list(ed = x))  
lines(x, y\_lm,col="red",lwd=3.2)  
lines(x, y\_lm2, col= "black", lwd=1.3)  
  
legend(x=5.5,y=14,legend=c("Modelo brute force","Modelo No Lineal"),  
 fill=c("red","black"),cex=0.9,text.font=0.2, bg='grey')



**Figura 7** Modelos no lineales

# Punto 4

## Datos

Con los siguientes datos:

punto4<- read.csv2("punto4.csv", dec = ".")  
punto4<- punto4 %>% select(-ï..)  
punto4

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ID | sexo | clase | test | exam1 | exam2 | labo | final |
| est1 | f | 4 | 37 | 42 | 93 | 98 | 98 |
| est2 | f | 4 | 20 | 22 | 39 | 56 | 77 |
| est3 | f | 4 | 73 | 74 | 40 | 86 | 165 |
| est4 | f | 4 | 21 | 25 | 45 | 47 | 81 |
| est5 | f | 4 | 73 | 78 | 44 | 58 | 165 |
| est6 | f | 4 | 56 | 65 | 39 | 22 | 150 |
| est7 | f | 4 | 45 | 53 | 25 | 37 | 112 |
| est8 | f | 4 | 74 | 78 | 67 | 70 | 167 |
| est9 | f | 4 | 51 | 68 | 36 | 70 | 135 |
| est10 | f | 4 | 80 | 84 | 73 | 78 | 175 |
| est11 | f | 4 | 55 | 64 | 69 | 54 | 143 |
| est12 | f | 4 | 24 | 27 | 75 | 52 | 82 |
| est13 | f | 4 | 58 | 70 | 30 | 48 | 155 |
| est14 | f | 4 | 32 | 38 | 68 | 26 | 96 |
| est15 | f | 4 | 43 | 51 | 33 | 22 | 111 |
| est16 | f | 4 | 45 | 54 | 22 | 52 | 116 |
| est17 | f | 4 | 58 | 44 | 36 | 39 | 153 |
| est18 | f | 4 | 44 | 45 | 65 | 38 | 111 |
| est19 | f | 4 | 39 | 43 | 74 | 21 | 106 |
| est20 | f | 4 | 54 | 61 | 56 | 75 | 140 |
| est21 | f | 4 | 33 | 41 | 49 | 53 | 96 |
| est22 | f | 4 | 43 | 49 | 63 | 66 | 109 |
| est23 | f | 4 | 74 | 79 | 47 | 22 | 168 |
| est24 | m | 4 | 58 | 68 | 48 | 26 | 154 |
| est25 | m | 4 | 21 | 23 | 41 | 47 | 80 |
| est26 | m | 4 | 25 | 28 | 79 | 59 | 86 |
| est27 | m | 4 | 33 | 54 | 81 | 40 | 97 |
| est28 | m | 4 | 77 | 79 | 42 | 38 | 172 |
| est29 | m | 4 | 78 | 79 | 41 | 58 | 172 |
| est30 | m | 4 | 47 | 56 | 45 | 68 | 118 |
| est31 | m | 4 | 25 | 32 | 26 | 67 | 89 |
| est32 | m | 4 | 67 | 73 | 44 | 82 | 162 |
| est33 | m | 4 | 53 | 61 | 56 | 43 | 137 |
| est34 | m | 4 | 48 | 59 | 72 | 66 | 124 |
| est35 | m | 4 | 65 | 70 | 59 | 38 | 158 |
| est36 | m | 4 | 40 | 48 | 60 | 66 | 106 |
| est37 | m | 4 | 50 | 59 | 60 | 35 | 132 |
| est38 | m | 4 | 50 | 60 | 33 | 71 | 132 |
| est39 | m | 4 | 24 | 27 | 39 | 39 | 83 |
| est40 | m | 4 | 38 | 43 | 53 | 45 | 105 |
| est41 | m | 4 | 25 | 30 | 55 | 25 | 87 |
| est42 | m | 4 | 54 | 45 | 51 | 61 | 142 |
| est43 | m | 4 | 54 | 61 | 85 | 30 | 140 |
| est44 | m | 4 | 67 | 74 | 21 | 83 | 162 |
| est45 | m | 4 | 47 | 58 | 82 | 52 | 119 |
| est46 | m | 4 | 42 | 48 | 60 | 71 | 107 |
| est47 | m | 4 | 25 | 50 | 33 | 24 | 92 |
| est48 | m | 4 | 89 | 89 | 22 | 69 | 180 |
| est49 | m | 4 | 77 | 79 | 66 | 49 | 170 |
| est50 | m | 4 | 85 | 85 | 69 | 82 | 175 |

## 4.1

Modele el resultado del examen final con base en las pruebas previas, use criterios de depuración si se amerita y analice y juzgue sus resultados, incluidos los análisis y gráficos de los residuales del modelo y discuta sus resultados.

Se realiza un modelo lineal en el cual se toma final como variable dependiente y, como variables independientes test, labo, exam 1 y exam 2.

punto<- punto4 %>% nest()  
  
lm4<- function(data){  
 lm(final~ test + exam1 + exam2 + labo, data = data)  
 }  
  
  
punto<- punto %>%   
 mutate(model= map(data, lm4))  
  
#pun<- punto %>%   
 #mutate(resids= map2(data, model,add\_residuals))  
  
#resids<- unnest(pun, resids)  
   
mp<- punto %>%   
 mutate(pol= map(model, broom::tidy)) %>%   
 unnest(pol, .drop = TRUE)  
mp

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| term | estimate | std.error | statistic | p.value |
| (Intercept) | 50.5986422 | 4.1564893 | 12.1734086 | 0.0000000 |
| test | 1.6008694 | 0.1414655 | 11.3163260 | 0.0000000 |
| exam1 | 0.0914882 | 0.1443787 | 0.6336685 | 0.5295050 |
| exam2 | -0.0696396 | 0.0461714 | -1.5082846 | 0.1384711 |
| labo | -0.0790396 | 0.0445437 | -1.7744258 | 0.0827582 |

mp2<- punto %>%   
 mutate(pol= map(model, broom::glance)) %>%   
 unnest(pol, .drop = TRUE)  
mp2

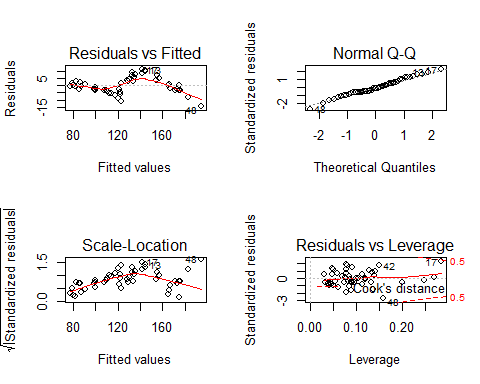
|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| r.squared | adj.r.squared | sigma | statistic | p.value | df | logLik | AIC | BIC | deviance | df.residual |
| 0.9690093 | 0.9662545 | 5.874672 | 351.7618 | 0 | 5 | -156.8454 | 325.6909 | 337.163 | 1553.03 | 45 |

**Tabla 1 resultados del modelo**

Lo anterior permite evidenciar que tanto el intercepto como la variable independiente “test” son significativas, el RSE es de y el , lo cual indica que los residuales no se alejan mucho de los valores predichos y que el el modelo explica aproximadamente el de la varianza. Además de eso, se realiza la prueba de Shapiro-Wilk con el fin de observar si hay normalidad en los residuales, la cual arroja un valor con un valor p de : se acepta la hipótesis nula, es decir, hay normalidad en los residuales. El AIC de este primer modelo es: , valor que sirve para comparar con los demás modelos planteados.

Una buena manera de analizar el modelo gráficamente es evaluando los residuales y los puntos que ejercen mayor palanca. En la **Figura 1**

par(mfrow= c(2,2))  
plot(lm(final~ test + exam1 + exam2 + labo, data = punto4))



**Figura 1**

De la **tabla 1** se puede observar que solo la variable test y el intercepto resultan significativos, mientras que las otras variables independientes no lo son, sin embargo, tal modelo resulta muy significativo (p-value ), lo cual podría indicar que tales variables pueden estar autocorrelacionadas. Dado lo anterior, se procede a depurar las variables por el método stepwise Backward, habiendo hecho eso, se escoge el modelo sin la variable exam 1 puesto que el AIC se minimiza, quedando como modelo 2: .

punto<- punto4 %>% nest()  
  
lm4<- function(data){  
 step(lm(final~ test + exam1 + exam2 + labo, data = data), direction = c("backward"))  
 }  
  
  
punto<- punto %>%   
 mutate(model= map(data, lm4))

## Start: AIC=181.8  
## final ~ test + exam1 + exam2 + labo  
##   
## Df Sum of Sq RSS AIC  
## - exam1 1 13.9 1566.9 180.24  
## <none> 1553.0 181.80  
## - exam2 1 78.5 1631.5 182.26  
## - labo 1 108.7 1661.7 183.18  
## - test 1 4419.6 5972.6 247.15  
##   
## Step: AIC=180.24  
## final ~ test + exam2 + labo  
##   
## Df Sum of Sq RSS AIC  
## <none> 1567 180.24  
## - exam2 1 79 1646 180.71  
## - labo 1 115 1682 181.78  
## - test 1 45133 46700 347.97

#pun<- punto %>%   
 #mutate(resids= map2(data, model,add\_residuals))  
  
#resids<- unnest(pun, resids)  
   
mp<- punto %>%   
 mutate(pol= map(model, broom::tidy)) %>%   
 unnest(pol, .drop = TRUE)  
mp

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| term | estimate | std.error | statistic | p.value |
| (Intercept) | 51.6082676 | 3.8139286 | 13.531524 | 0.0000000 |
| test | 1.6855065 | 0.0463044 | 36.400596 | 0.0000000 |
| exam2 | -0.0699341 | 0.0458677 | -1.524692 | 0.1341823 |
| labo | -0.0811126 | 0.0441335 | -1.837892 | 0.0725414 |

**Tabla 2 Resultados modelo con Backward**

Para analizar mejor el resultado obtenido, se realiza el resumen del segundo modelo propuesto, el cual resulta con un RSE, un R2similares al del primer modelo, pero con un AIC significativamente menor (). Sin embargo, se observa que tanto examen 2 como labo siguen sin ser significativas. **Tabla 2**

punto<- punto4 %>% nest()  
  
lm4<- function(data){  
 lm(final~ test, data = data)  
 }  
  
  
punto<- punto %>%   
 mutate(model= map(data, lm4))  
  
pun<- punto %>%   
 mutate(resids= map2(data, model,add\_residuals))  
  
resids<- unnest(pun, resids)  
   
mp<- punto %>%   
 mutate(pol= map(model, broom::tidy)) %>%   
 unnest(pol, .drop = TRUE)  
mp

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| term | estimate | std.error | statistic | p.value |
| (Intercept) | 44.522352 | 2.4612895 | 18.08904 | 0 |
| test | 1.669023 | 0.0461931 | 36.13146 | 0 |

mp2<- punto %>%   
 mutate(pol= map(model, broom::glance)) %>%   
 unnest(pol, .drop = TRUE)  
mp2

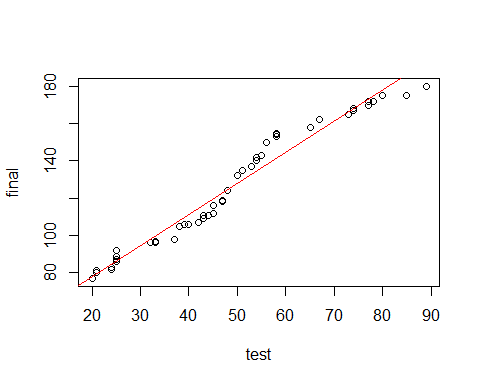
|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| r.squared | adj.r.squared | sigma | statistic | p.value | df | logLik | AIC | BIC | deviance | df.residual |
| 0.9645359 | 0.9637971 | 6.08482 | 1305.482 | 0 | 2 | -160.2162 | 326.4325 | 332.1685 | 1777.201 | 48 |

**tabla 3 Modelo 3**

Para continuar con la selección del modelo se realiza otras dos veces el proceso stepwise “Backward” obteniendo, un modelo 3 el cual utiliza como variable predictora únicamente a test, con un RSE de y un AIC de y cumpliendo tambien con el principio de parsimonia.

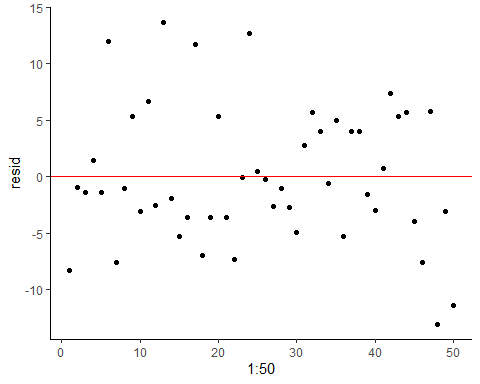
Finalmente, en la **Figura 2** se observa la tendencia aproximadamente lineal de los datos y la relación que tiene final con Test. En la **Figura 3** se pueden ver los residuales estandarizados siguiendo una tendencia de homogeneidad lo que indica que existe homocedasticidad.

with(punto4, plot(x= test, y= final))  
abline(lm(final~ test, data = punto4), col= "red")



**Figura 2** Tendencia de los datos

resids %>% ggplot(aes(x= 1:50, y= resid))+  
 geom\_point() + geom\_abline(slope = 0, color= "red") +  
 theme\_classic()



**Figura 3** Residuales

## 4.2

Muestre los coeficientes de correlación entre las variables independientes y saque algunas conclusiones de ello.

punt<- punto4 %>% select(test, exam1, exam2, labo)  
  
as.data.frame(cor(punt))

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | test | exam1 | exam2 | labo |
| test | 1.0000000 | 0.9476165 | -0.0942221 | 0.2699188 |
| exam1 | 0.9476165 | 1.0000000 | -0.0943055 | 0.2328852 |
| exam2 | -0.0942221 | -0.0943055 | 1.0000000 | 0.0489087 |
| labo | 0.2699188 | 0.2328852 | 0.0489087 | 1.0000000 |

**tabla 1 matriz de correlaciónes**

Los coeficientes de correlación muestran qué tanto se relaciona una variable predictora con otra. Un coeficiente positivo cercano a uno, muestra una correlación positiva casi perfecta; por el contrario, si el coeficiente de correlación es cercano a menos uno, la correlación será negativa.

La **tabla 1** muestra que test se encuentra altamente relacionada (cerca de un ) con exam1 y exam 2 así, cualquiera de las tres pudo ser escogida para el ajuste del modelo, sin embargo la segunda y tercera variable no fueron significativas, por tal motivo se prefirió trabajar solo con “test”.

Como resultado general se puede observar que todas las variables se encuentran altamente correlacionadas, exceptuando a “labo”, que presenta su máxima correlación con test, del y su mínima correlación con examen 2,cerca del , sin embargo, tal variable no resulta significativa y por ello no se incluye en el modelo.

## 4.3

Encuentre las distancias de COOK, significativas.

mp3<- punto %>%   
 mutate(pol= map(model, broom::augment)) %>%   
 unnest(pol, .drop = TRUE)  
mp3

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| final | test | .fitted | .se.fit | .resid | .hat | .sigma | .cooksd | .std.resid |
| 98 | 37 | 106.27622 | 1.0472289 | -8.2762176 | 0.0296202 | 6.025857 | 0.0290966 | -1.3807445 |
| 77 | 20 | 77.90282 | 1.6280948 | -0.9028198 | 0.0715919 | 6.147692 | 0.0009142 | -0.1539869 |
| 165 | 73 | 166.36106 | 1.3700902 | -1.3610601 | 0.0506994 | 6.145834 | 0.0014074 | -0.2295767 |
| 81 | 21 | 79.57184 | 1.5890689 | 1.4281568 | 0.0682009 | 6.145423 | 0.0021636 | 0.2431460 |
| 165 | 73 | 166.36106 | 1.3700902 | -1.3610601 | 0.0506994 | 6.145834 | 0.0014074 | -0.2295767 |
| 150 | 56 | 137.98766 | 0.9051958 | 12.0123377 | 0.0221304 | 5.888394 | 0.0450980 | 1.9963623 |
| 112 | 45 | 119.62840 | 0.8900293 | -7.6284049 | 0.0213950 | 6.045460 | 0.0175566 | -1.2673084 |
| 167 | 74 | 168.03008 | 1.4063347 | -1.0300835 | 0.0534173 | 6.147271 | 0.0008542 | -0.1739985 |
| 135 | 51 | 129.64255 | 0.8619684 | 5.3574547 | 0.0200672 | 6.098328 | 0.0081000 | 0.8894318 |
| 175 | 80 | 178.04422 | 1.6343737 | -3.0442240 | 0.0721452 | 6.131907 | 0.0104876 | -0.5193844 |
| 143 | 55 | 136.31864 | 0.8919452 | 6.6813611 | 0.0214873 | 6.069773 | 0.0135286 | 1.1100282 |
| 82 | 24 | 84.57891 | 1.4744785 | -2.5789134 | 0.0587194 | 6.136975 | 0.0059524 | -0.4368471 |
| 155 | 58 | 141.32571 | 0.9379812 | 13.6742909 | 0.0237625 | 5.808401 | 0.0629602 | 2.2744656 |
| 96 | 32 | 97.93110 | 1.1940353 | -1.9311006 | 0.0385069 | 6.142497 | 0.0020976 | -0.3236563 |
| 111 | 43 | 116.29036 | 0.9179764 | -5.2903581 | 0.0227598 | 6.099462 | 0.0090076 | -0.8795017 |
| 116 | 45 | 119.62840 | 0.8900293 | -3.6284049 | 0.0213950 | 6.125892 | 0.0039720 | -0.6027876 |
| 153 | 58 | 141.32571 | 0.9379812 | 11.6742909 | 0.0237625 | 5.902749 | 0.0458899 | 1.9418026 |
| 111 | 44 | 117.95938 | 0.9029300 | -6.9593815 | 0.0220198 | 6.062929 | 0.0150580 | -1.1565327 |
| 106 | 39 | 109.61426 | 0.9974711 | -3.6142644 | 0.0268723 | 6.125944 | 0.0050059 | -0.6021259 |
| 140 | 54 | 134.64962 | 0.8809204 | 5.3503845 | 0.0209594 | 6.098416 | 0.0084532 | 0.8886627 |
| 96 | 33 | 99.60012 | 1.1624882 | -3.6001240 | 0.0364991 | 6.125895 | 0.0068816 | -0.6027590 |
| 109 | 43 | 116.29036 | 0.9179764 | -7.2903581 | 0.0227598 | 6.054389 | 0.0171056 | -1.2119940 |
| 168 | 74 | 168.03008 | 1.4063347 | -0.0300835 | 0.0534173 | 6.149209 | 0.0000007 | -0.0050816 |
| 154 | 58 | 141.32571 | 0.9379812 | 12.6742909 | 0.0237625 | 5.857626 | 0.0540884 | 2.1081341 |
| 80 | 21 | 79.57184 | 1.5890689 | 0.4281568 | 0.0682009 | 6.148871 | 0.0001945 | 0.0728944 |
| 86 | 25 | 86.24794 | 1.4372211 | -0.2479368 | 0.0557894 | 6.149098 | 0.0000519 | -0.0419333 |
| 97 | 33 | 99.60012 | 1.1624882 | -2.6001240 | 0.0364991 | 6.137060 | 0.0035896 | -0.4353317 |
| 172 | 77 | 173.03715 | 1.5183126 | -1.0371538 | 0.0622626 | 6.147226 | 0.0010285 | -0.1760171 |
| 172 | 78 | 174.70618 | 1.5565904 | -2.7061772 | 0.0654415 | 6.135639 | 0.0074102 | -0.4600503 |
| 118 | 47 | 122.96645 | 0.8710306 | -4.9664517 | 0.0204914 | 6.105491 | 0.0071141 | -0.8246969 |
| 89 | 25 | 86.24794 | 1.4372211 | 2.7520632 | 0.0557894 | 6.135318 | 0.0064004 | 0.4654535 |
| 162 | 67 | 156.34692 | 1.1674702 | 5.6530803 | 0.0368126 | 6.091540 | 0.0171246 | 0.9466339 |
| 137 | 53 | 132.98059 | 0.8722056 | 4.0194079 | 0.0205467 | 6.120609 | 0.0046728 | 0.6674558 |
| 124 | 48 | 124.63548 | 0.8650819 | -0.6354751 | 0.0202125 | 6.148498 | 0.0001148 | -0.1055079 |
| 158 | 65 | 153.00887 | 1.1071317 | 4.9911271 | 0.0331057 | 6.104475 | 0.0119129 | 0.8341832 |
| 106 | 40 | 111.28329 | 0.9749258 | -5.2832878 | 0.0256713 | 6.099447 | 0.0101934 | -0.8796377 |
| 132 | 50 | 127.97352 | 0.8605314 | 4.0264781 | 0.0200004 | 6.120524 | 0.0045594 | 0.6684435 |
| 132 | 50 | 127.97352 | 0.8605314 | 4.0264781 | 0.0200004 | 6.120524 | 0.0045594 | 0.6684435 |
| 83 | 24 | 84.57891 | 1.4744785 | -1.5789134 | 0.0587194 | 6.144627 | 0.0022312 | -0.2674552 |
| 105 | 38 | 107.94524 | 1.0216089 | -2.9452410 | 0.0281886 | 6.133749 | 0.0034964 | -0.4910007 |
| 87 | 25 | 86.24794 | 1.4372211 | 0.7520632 | 0.0557894 | 6.148175 | 0.0004780 | 0.1271956 |
| 142 | 54 | 134.64962 | 0.8809204 | 7.3503845 | 0.0209594 | 6.052987 | 0.0159541 | 1.2208491 |
| 140 | 54 | 134.64962 | 0.8809204 | 5.3503845 | 0.0209594 | 6.098416 | 0.0084532 | 0.8886627 |
| 162 | 67 | 156.34692 | 1.1674702 | 5.6530803 | 0.0368126 | 6.091540 | 0.0171246 | 0.9466339 |
| 119 | 47 | 122.96645 | 0.8710306 | -3.9664517 | 0.0204914 | 6.121360 | 0.0045377 | -0.6586434 |
| 107 | 42 | 114.62133 | 0.9350648 | -7.6213346 | 0.0236150 | 6.045416 | 0.0194304 | -1.2675724 |
| 92 | 25 | 86.24794 | 1.4372211 | 5.7520632 | 0.0557894 | 6.088287 | 0.0279599 | 0.9728402 |
| 180 | 89 | 193.06543 | 1.9998350 | -13.0654346 | 0.1080172 | 5.808694 | 0.3129697 | -2.2735155 |
| 170 | 77 | 173.03715 | 1.5183126 | -3.0371538 | 0.0622626 | 6.132169 | 0.0088201 | -0.5154404 |
| 175 | 85 | 186.38934 | 1.8347669 | -11.3893410 | 0.0909215 | 5.897188 | 0.1927238 | -1.9631351 |

**tabla 1 resultados varios del modelo**

La distancia de Cook mide el efecto de eliminar una observación. Los puntos con residuales o influencia altas, suelen tener una distancia de Cook significativa;asimismo, dichos puntos pueden distorsionar el modelo, es por ello que se amerita esta examinación.

Como se ve en la **tabla 1**, los datos más influyentes son el 13, el 48 y el 50, los cuales resultan significativos. La distancia de cook es una medida de la influencia que ejerce cada dato sobre el modelo, pero como se puede ver en el gráfico de residuales estandarizados, no hay ningún dato que supere ni siquiera las tres desviaciones estándar, por lo que no se podría concluir que alguno de estos datos no sea posible (sea un outlier) en esta población de datos.

## Punto 5

**Se tomó la siguiente información de datos educativos, para intentar modelar la obtención de becas, de acuerdo con las características género, raza, índice de calidad (cain), sector educativo, tipo de educación a la que se aspiraría (clásica, vocacional, pregrado), y las notas en las asignaturas elegidas para programar las becas (, )**

La base de datos que se utiliza para modelar la variable dependiente “beca” en función de las demás variables, contiene variables de tipo binomial, es decir, variables de tipo éxito-fracaso. Para ajustar el modelo con éxito, inicialmente se transformaron las variables binomiales a variables dummy, pues permiten diferenciar mejor las clases de cada variable. Esto con la ayuda de las tres librerías propuestas

library(MASS)  
library(car)  
library(pscl)  
punto5<- read.csv2("gereno.csv", dec = ".")  
  
  
punto<- punto5 %>% nest()  
  
lm4<- function(data){  
 glm(bec~genero+raza+cain+sect+tipr+geo+filo+mate+cina+hist,data= data,family=binomial (link=logit))  
 }  
  
  
punto<- punto %>%   
 mutate(model= map(data, lm4))  
  
   
mp<- punto %>%   
 mutate(pol= map(model, broom::tidy)) %>%   
 unnest(pol, .drop = TRUE)  
mp

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| term | estimate | std.error | statistic | p.value |
| (Intercept) | 4.6428228 | 1.5888174 | 2.9221878 | 0.0034758 |
| generomuj | -0.0890654 | 0.3705103 | -0.2403858 | 0.8100312 |
| razaindi | 0.0426574 | 0.5407748 | 0.0788819 | 0.9371265 |
| razallan | 0.0254571 | 0.7863839 | 0.0323723 | 0.9741751 |
| razanegr | 0.8293106 | 0.7114727 | 1.1656253 | 0.2437660 |
| cainbaja | -0.7424359 | 0.5023832 | -1.4778278 | 0.1394539 |
| cainmedi | 0.0277526 | 0.3981102 | 0.0697109 | 0.9444238 |
| sectpubl | 0.0849576 | 0.4670292 | 0.1819107 | 0.8556528 |
| tiprpregr | -0.4020073 | 0.4799465 | -0.8376086 | 0.4022506 |
| tiprvoca | -1.4436912 | 0.5207853 | -2.7721429 | 0.0055689 |
| geo | -0.0985073 | 0.1940008 | -0.5077673 | 0.6116165 |
| filo | 0.1629257 | 0.2275662 | 0.7159486 | 0.4740231 |
| mate | 0.0050490 | 0.2163719 | 0.0233347 | 0.9813833 |
| cina | -0.4456067 | 0.2108980 | -2.1129010 | 0.0346092 |
| hist | -0.1052756 | 0.1738549 | -0.6055372 | 0.5448221 |

mp2<- punto %>%   
 mutate(pol= map(model, broom::glance)) %>%   
 unnest(pol, .drop = TRUE)  
mp2

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| null.deviance | df.null | logLik | AIC | BIC | deviance | df.residual |
| 247.6403 | 199 | -113.8861 | 257.7723 | 307.2471 | 227.7723 | 185 |

Este modelo inicial, incluyendo todas las variables, muestra que solo el intercepto, la clase “tiprvoca” y “cina” son significativas, es por ello que se procede a hacer una depuración de las variables vía backward, con el fin de minimizar y

El paso step.wise, arrojó varios posibles modelos de los cuales se probaron dos, el primero de ellos es bec~ cain+tipr+cina y el segundo es bec~tipr+cina , (este último contiene las dos variables que inicialmente dieron significativas). En ambos resultados se observa que ninguna otra variable comenzaba a ser significativa, tan solo “tiprvoca” y “cina”, sin embargo, ambos modelos reducen en , aunque la deviance continúa siento la misma.

punto<- punto5 %>% nest()  
  
lm5<- function(data){  
 glm(bec~cain+sect+tipr+cina,data= data,family=binomial (link=logit))  
 }  
  
  
punto<- punto %>%   
 mutate(model= map(data, lm5))  
  
   
mp<- punto %>%   
 mutate(pol= map(model, broom::tidy)) %>%   
 unnest(pol, .drop = TRUE)  
mp

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| term | estimate | std.error | statistic | p.value |
| (Intercept) | 4.8021561 | 1.2673606 | 3.7891001 | 0.0001512 |
| cainbaja | -0.5825532 | 0.4737051 | -1.2297803 | 0.2187794 |
| cainmedi | 0.0377895 | 0.3943320 | 0.0958316 | 0.9236543 |
| sectpubl | 0.0468627 | 0.4615561 | 0.1015319 | 0.9191282 |
| tiprpregr | -0.4428231 | 0.4425066 | -1.0007154 | 0.3169644 |
| tiprvoca | -1.4229435 | 0.5087600 | -2.7968858 | 0.0051598 |
| cina | -0.5004105 | 0.1505614 | -3.3236316 | 0.0008885 |

mp2<- punto %>%   
 mutate(pol= map(model, broom::glance)) %>%   
 unnest(pol, .drop = TRUE)  
mp2

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| null.deviance | df.null | logLik | AIC | BIC | deviance | df.residual |
| 247.6403 | 199 | -115.1045 | 244.2089 | 267.2971 | 230.2089 | 193 |

punto<- punto5 %>% nest()  
  
lm6<- function(data){  
 glm(bec~tipr+cina,data= data,family=binomial (link=logit))  
 }  
  
  
punto<- punto %>%   
 mutate(model= map(data, lm6))  
  
   
mp<- punto %>%   
 mutate(pol= map(model, broom::tidy)) %>%   
 unnest(pol, .drop = TRUE)  
mp

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| term | estimate | std.error | statistic | p.value |
| (Intercept) | 4.2710670 | 1.0593343 | 4.0318404 | 0.0000553 |
| tiprpregr | -0.3360126 | 0.4273803 | -0.7862144 | 0.4317419 |
| tiprvoca | -1.2828919 | 0.4904351 | -2.6158238 | 0.0089012 |
| cina | -0.4468083 | 0.1437427 | -3.1083892 | 0.0018811 |

mp2<- punto %>%   
 mutate(pol= map(model, broom::glance)) %>%   
 unnest(pol, .drop = TRUE)  
mp2

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| null.deviance | df.null | logLik | AIC | BIC | deviance | df.residual |
| 247.6403 | 199 | -116.2702 | 240.5405 | 253.7338 | 232.5405 | 196 |

Los resultados del último modelo, muestran que la obtención de beca tan solo dependen de las variables “cina” y “tipr”, para esta última, la clase que influye es la de tipo vocacional.