pfinal

May 11, 2019

1 Carga de datos y preparación del espacio de trabajo.

1.1 Cargamos las bibliotecas / librerías que utilizamos

1.2 Definimos las funciones user-defined que hemos necesitado

In [43]:
```
# Preparamos el espacio de trabajo
# setwd('/Users/agus/Downloads/mdbd-master-3/proyecto_final')
setwd('C:/Users/ealcober/git/mdbd/proyecto_final')

r = getOption("repos")
r["CRAN"] = "http://cran.us.r-project.org"
options(repos = r)

# para trabajar con ficheros excel
if(!require("XLConnect"))install.packages("XLConnect")
library(XLConnect)

# dep 1
if(!require("dplyr"))install.packages("dplyr")
library(dplyr)

# dep 2
if(!require("ggplot2"))install.packages("ggplot2", repos=repos)
library(ggplot2)

if(!require("caTools"))install.packages("caTools", repos=repos)
library(caTools)

# dep 3
if(!require("RColorBrewer"))install.packages("RColorBrewer")
library(RColorBrewer)

# Función para eliminar fila por índice
removeRowByIndex <- function(x, row_index) {
  nr <- nrow(x)
  if (nr < row_index) {
```

```
    print('row_index exceeds number of rows')
  } else if (row_index == 1)
  {
    return(x[2:nr, ])
  } else if (row_index == nr) {
    return(x[1:(nr - 1), ])
  } else {
    return (x[c(1:(row_index - 1), (row_index + 1):nr), ])
  }
}

# Funcion que convierte a 0's los NAs
haz.cero.na=function(x){
  ifelse(is.na(x),0,x)}
```

2 Nuestra hipótesis inicial:

2.1 Queremos encontrar, si existe, una relación entre el nivel educativo de la población y el nivel de uso de plástico y su gestión en el rango de años 1960-2010.

2.2 De forma intuitiva, ante el incremento del efecto invernadero y la cantidad de plástico producida industrialmente, una población con mejor formación y educación habría de ser más eficiente en el tratamiento de residuos, o en sus elecciones a la hora de escoger en sus compras, otro tipo de residuos no plásticos.

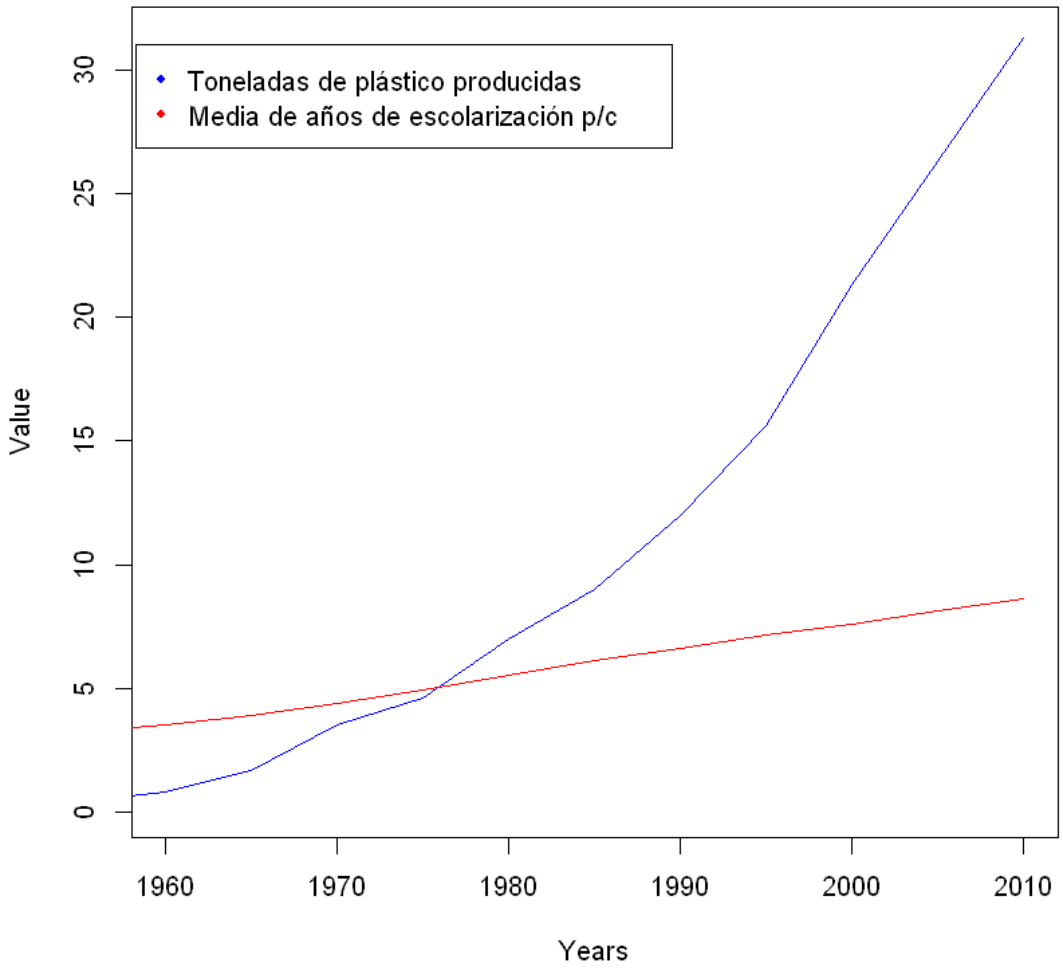
3 Comenzamos exponiendo brevemente unas cifras que ilustran la Producción de plástico global en Toneladas, en serie temporal, sobre la gráfica de "media de años que un adulto pasa escolarizado" a lo largo de los años estudiados.

In [44]:
```
gpp<-read.csv('sources/global-plastics-production.csv',header=TRUE,sep="," , quote="\'
sch<-read.csv('sources/mean-years-of-schooling-selected-countries.csv',header=TRUE,sep

gpp$Entity <- NULL
gpp$Code <- NULL

# Arreglamos los nombres de las variables
colnames(gpp) <- c("Year", "Prod de plastico global")
colnames(sch) <- c("Entity", "Code", "Year", "TotalYearsAtSchool")

# Sumamos por año, calculando la media de años que pasa escolarizada cada persona
schmean<-aggregate.data.frame(sch$TotalYearsAtSchool ,list(sch$Year), FUN=mean)
colnames(schmean) <- c("Year", "mean escolarizados")
prueba <- merge(gpp, schmean, by.y="Year", sort = TRUE)

# Dividimos las cantidades a x10 toneladas (1 unidad son 10 toneladas) por proporcion
prueba$`Prod de plastico global` <- prueba$`Prod de plastico global` / 10000000
```

```r
# Imprimimos la tabla y el gráfico con su leyenda correspondiente.
# Muestra de ambas tablas
tail(gpp)
tail(sch)
plot(prueba$Year,prueba$`Prod de plastico global`,type="l",col="blue",
     xlim=c(1960,2010), xlab="Years", ylab="Value")
lines(prueba$Year,prueba$`mean escolarizados`,col="red")

legend("bottomleft", legend=c("Toneladas de plástico producidas", "Media de años de es
       col=c("blue", "red"),
       pch=c(20,20),
        inset=c(0.005,0.83)
       )
```

	Year	Prod de plastico global
61	2010	313000000
62	2011	325000000
63	2012	338000000
64	2013	352000000
65	2014	367000000
66	2015	381000000

	Entity	Code	Year	TotalYearsAtSchool
3214	Zimbabwe	ZWE	1985	4.84
3215	Zimbabwe	ZWE	1990	5.97
3216	Zimbabwe	ZWE	1995	6.85
3217	Zimbabwe	ZWE	2000	7.26
3218	Zimbabwe	ZWE	2005	7.65
3219	Zimbabwe	ZWE	2010	7.86

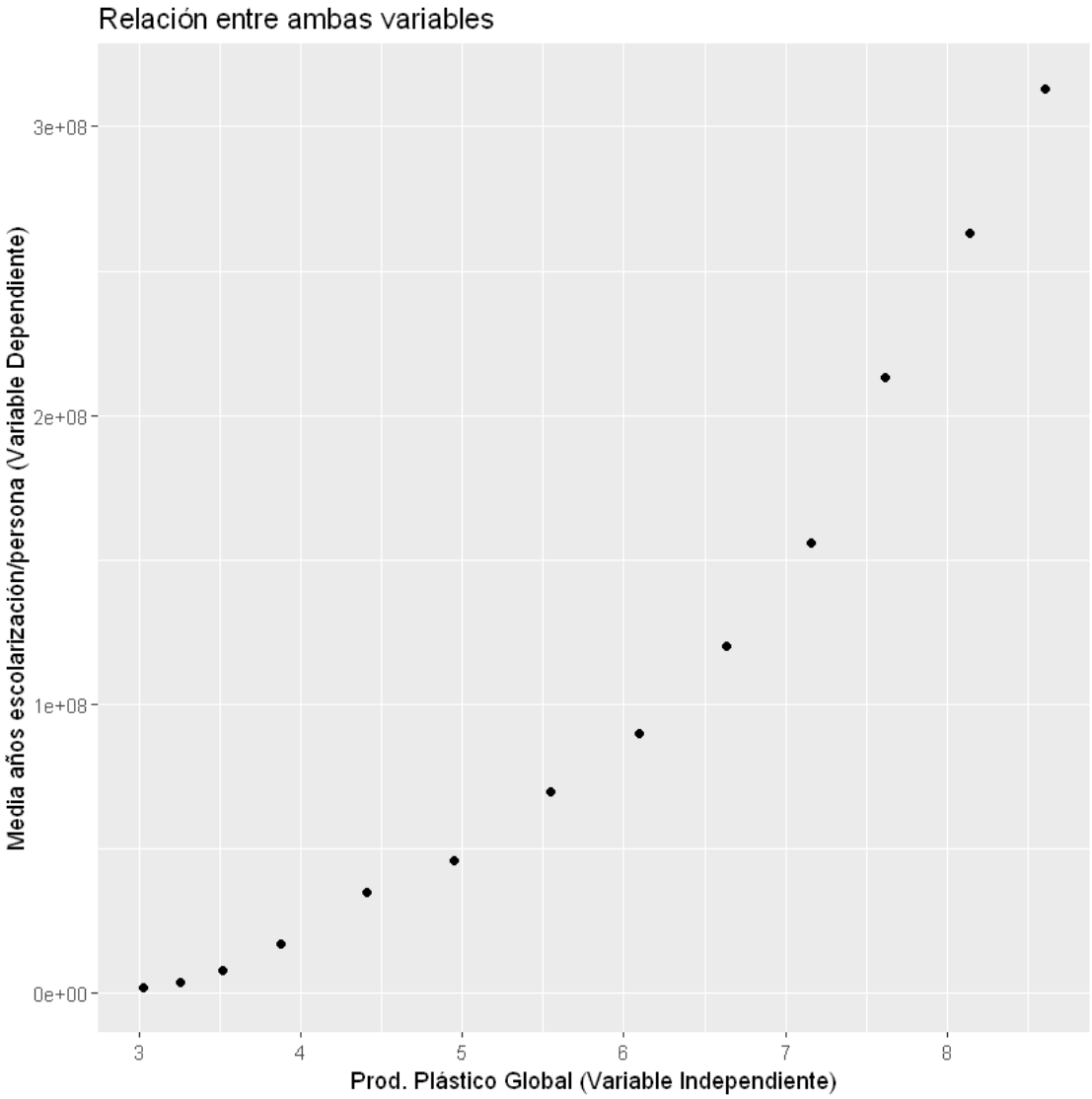


4 vamos a buscar una relación entre ambas variables, colocándolas en ambos ejes respectivos x e y en una gráfica de 2d.

In [45]:
```
# agrupación de ambas tablas por año
prueba <- merge(gpp, schmean, by.y="Year", sort = TRUE)



prueba<-prueba[,-1]
prueba
colnames(prueba) <- c('y', 'x')
ggplot() + geom_point(data = prueba, aes(x = prueba$x, y = prueba$y)) +
```

```
        xlab("Prod. Plástico Global (Variable Independiente)") +
        ylab("Media años escolarización/persona (Variable Dependiente)") +
        ggtitle("Relación entre ambas variables")
```

Prod de plastico global	mean escolarizados
2000000	3.026036
4000000	3.253243
8000000	3.516486
17000000	3.878198
35000000	4.411892
46000000	4.949279
70000000	5.547838
90000000	6.095315
120000000	6.636126
156000000	7.155495
213000000	7.612883
263000000	8.139279
313000000	8.607477


Relación entre ambas variables
Media años escolarización/persona (Variable Dependiente)
3e+08
2e+08
1e+08
0e+00
3
4
5
6
7
8
Prod. Plástico Global (Variable Independiente)

5 De forma contraria a como esperábamos la proporción aumenta:

6 A lo largo de los años ha ido aumentando el tiempo medio que una persona pasa escolarizada en su vida, y proporcionalmente a éste, la cantidad de plástico total producido en esos años.

7 Vamos a tratar de aproximarnos para predicciones posteriores mediante regresión: partimos el conjunto de nuestros datos en 70% prueba y 30% entrenamiento

```
In [46]: split = sample.split(prueba$y, SplitRatio = 0.7)
         nltrain = subset(prueba, split == TRUE)
         nltest = subset(prueba, split == FALSE)
         nltest
         nltrain
```

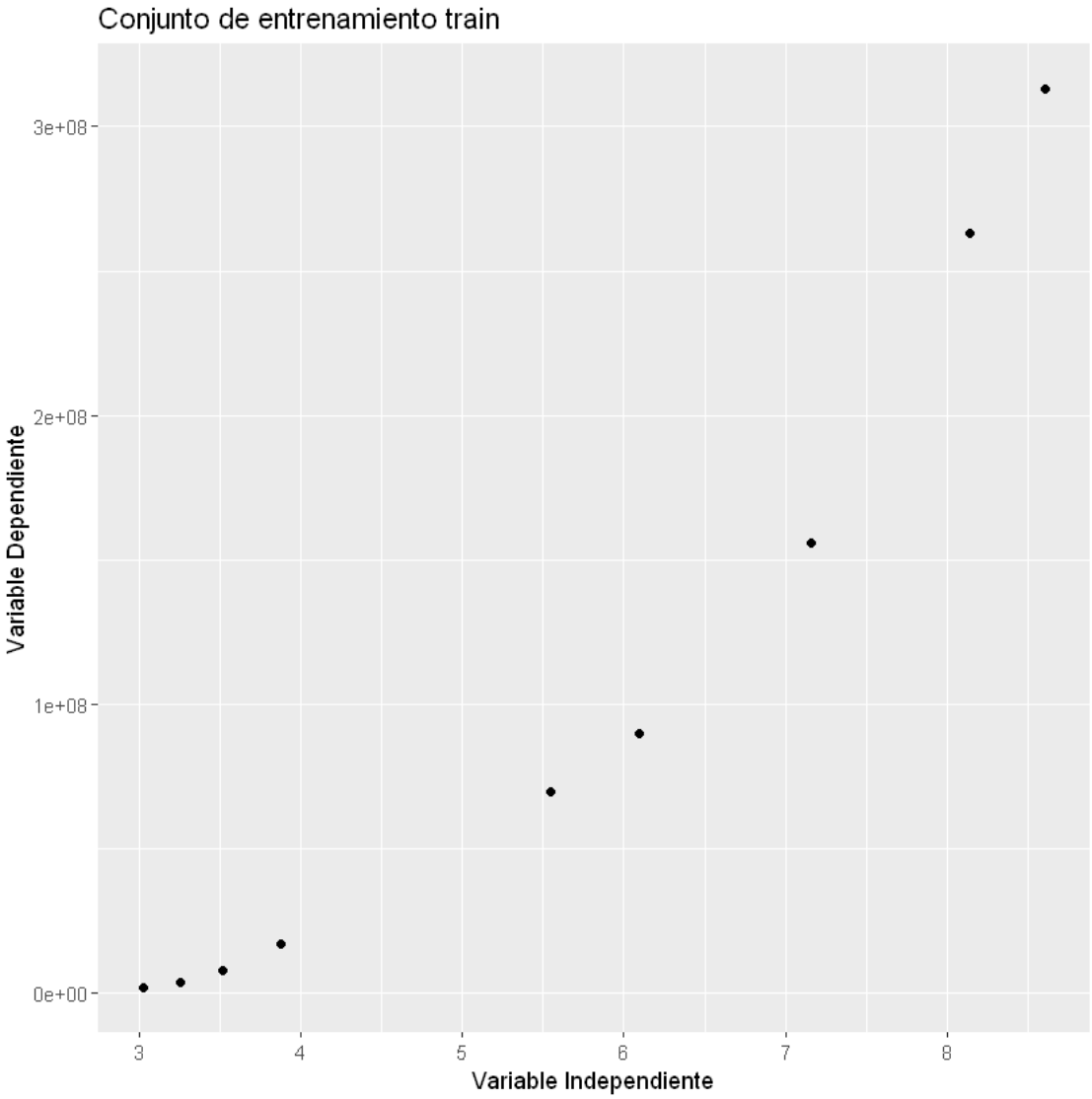
	y	x
5	35000000	4.411892
6	46000000	4.949279
9	120000000	6.636126
11	213000000	7.612883

	y	x
1	2000000	3.026036
2	4000000	3.253243
3	8000000	3.516486
4	17000000	3.878198
7	70000000	5.547838
8	90000000	6.095315
10	156000000	7.155495
12	263000000	8.139279
13	313000000	8.607477

```
In [47]: nltrain$x2 <- nltrain$x^2
         str(nltrain)
```

```
'data.frame':        9 obs. of  3 variables:
 $ y : int  2000000 4000000 8000000 17000000 70000000 90000000 156000000 263000000 313000000
 $ x : num  3.03 3.25 3.52 3.88 5.55 ...
 $ x2: num  9.16 10.58 12.37 15.04 30.78 ...
```

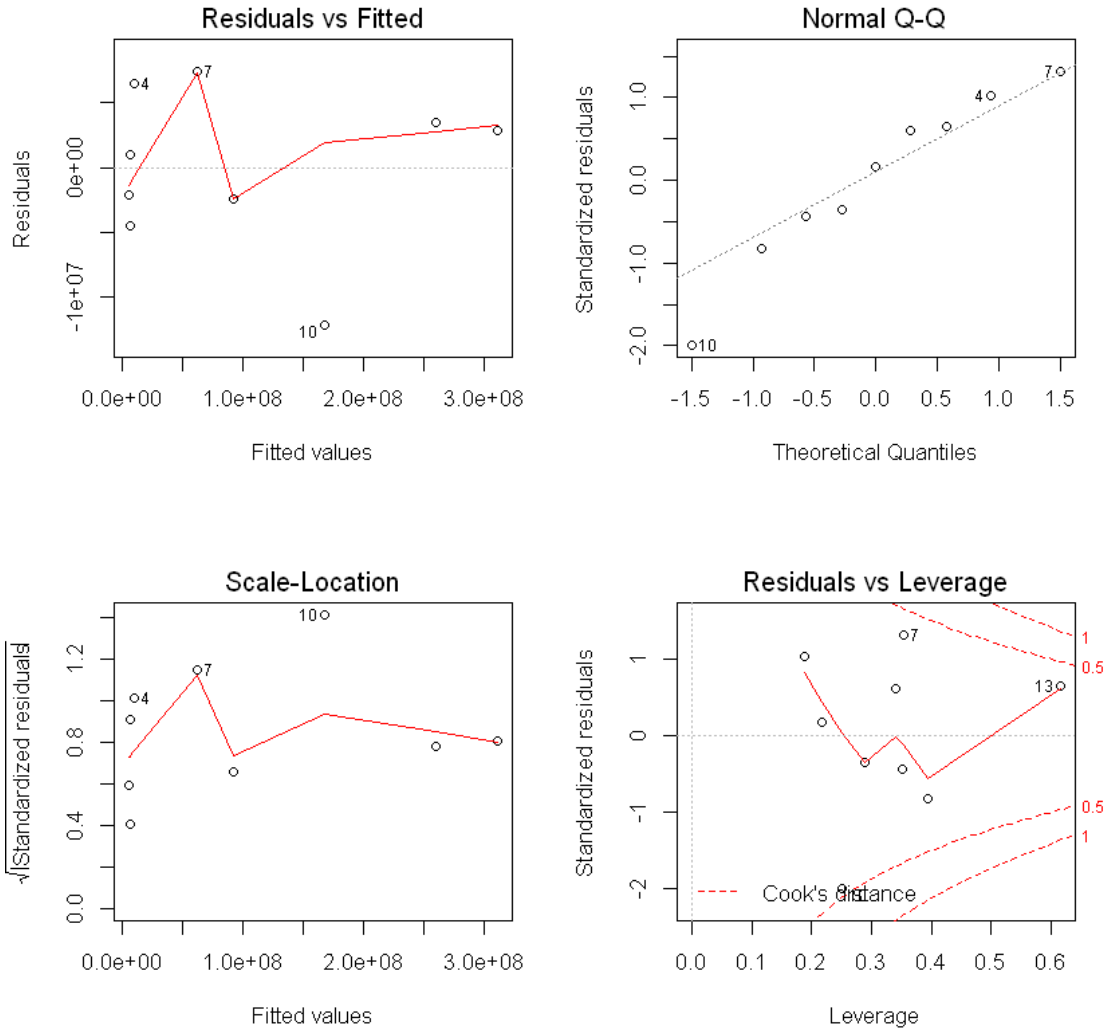
```
In [48]: ggplot() + geom_point(data = nltrain, aes(x = nltrain$x, y = nltrain$y)) +
           xlab("Variable Independiente") +
           ylab("Variable Dependiente") +
           ggtitle("Conjunto de entrenamiento train")
```




```
In [49]: set.seed(1234)
         regresion_lineal <- lm(y ~ x, data = nltrain)
         regresion_poly <- lm(y ~ x + x2, data = nltrain)

In [50]: summary(regresion_poly)
         summary(regresion_lineal)

         layout(matrix(c(1,2,3,4), 2, 2, byrow = TRUE))
         plot(regresion_poly)
```

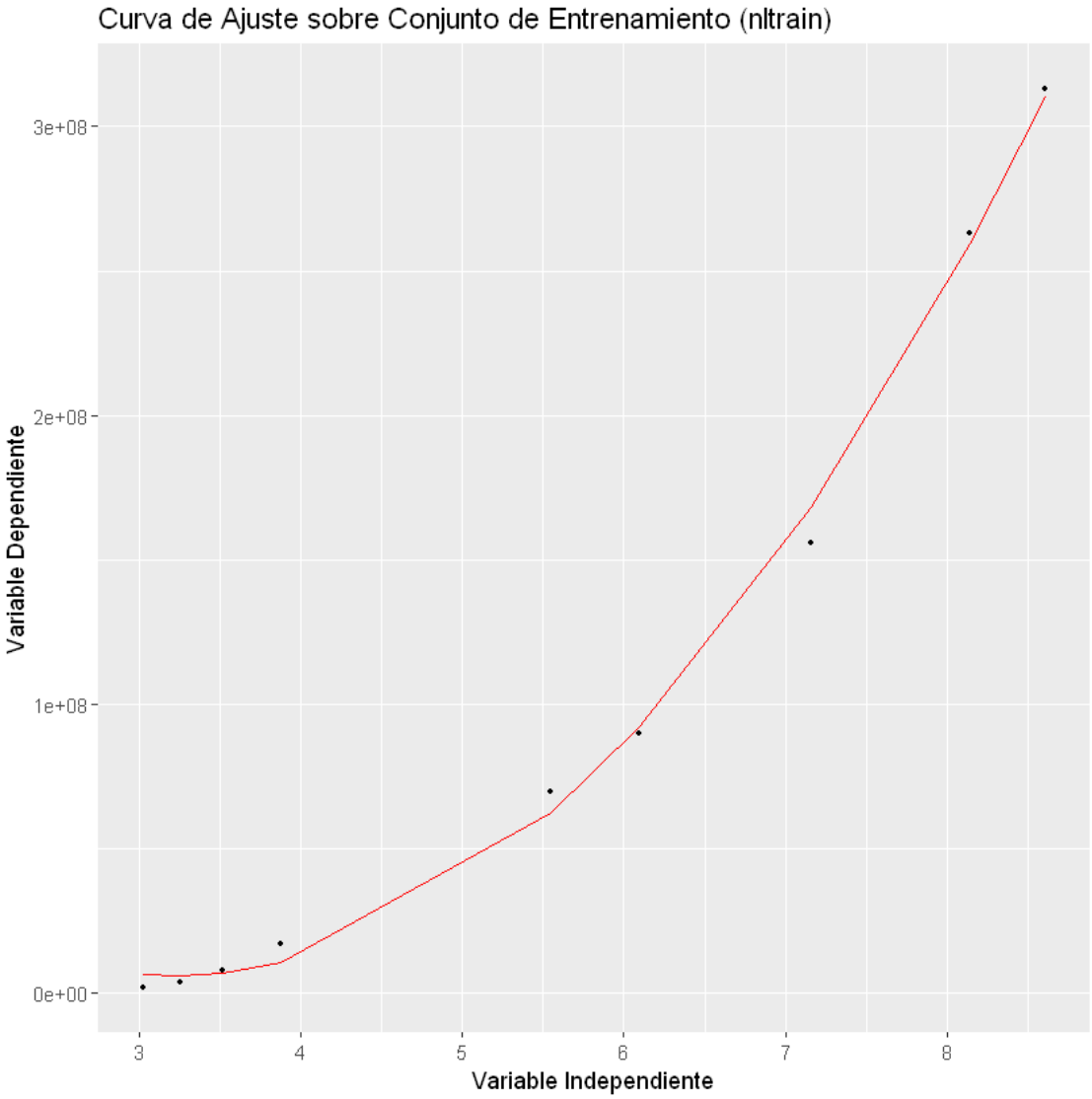
```
Call:
lm(formula = y ~ x + x2, data = nltrain)
```

```
Residuals:
      Min        1Q    Median        3Q       Max
-12144661  -2436559   1043313   3459191   7404378

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 115773853   24403860   4.744  0.00318 **
x           -67920362    9509424  -7.142  0.00038 ***
x2           10514901     826551  12.721 1.45e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7026000 on 6 degrees of freedom
Multiple R-squared:  0.9973,Adjusted R-squared:  0.9964
F-statistic:  1113 on 2 and 6 DF,  p-value: 1.943e-08




Call:
lm(formula = y ~ x, data = nltrain)

Residuals:
      Min        1Q    Median        3Q       Max
-45238522 -34545826   7293396  21132303  46707417

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -182740883   32816938  -5.568 0.000843 ***
x             52167835    5622406   9.279  3.5e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 34400000 on 7 degrees of freedom
Multiple R-squared:  0.9248,Adjusted R-squared:  0.9141
F-statistic: 86.09 on 1 and 7 DF,  p-value: 3.497e-05
```



7.1 Se muestran los resultados de aplicar una regresión lineal, y una polinomial de segundo grado. Como podemos observar en el caso lineal el error el bastante mayor que en el polinomial, luego vamos a intentar afinar la precisión de la predicción y ajuste de este modelo agregandole más características: x^2 y x^3

```
In [51]: y_poly_predict <- predict(regresion_poly, nltrain)
         ggplot() + geom_point(data = nltrain, aes(x = nltrain$x, y = nltrain$y), size = 0.7) +
           geom_line(aes( x = nltrain$x, y = y_poly_predict), color = "red") +
           xlab("Variable Independiente") +
           ylab("Variable Dependiente") +
           ggtitle("Curva de Ajuste sobre Conjunto de Entrenamiento (nltrain)")
```

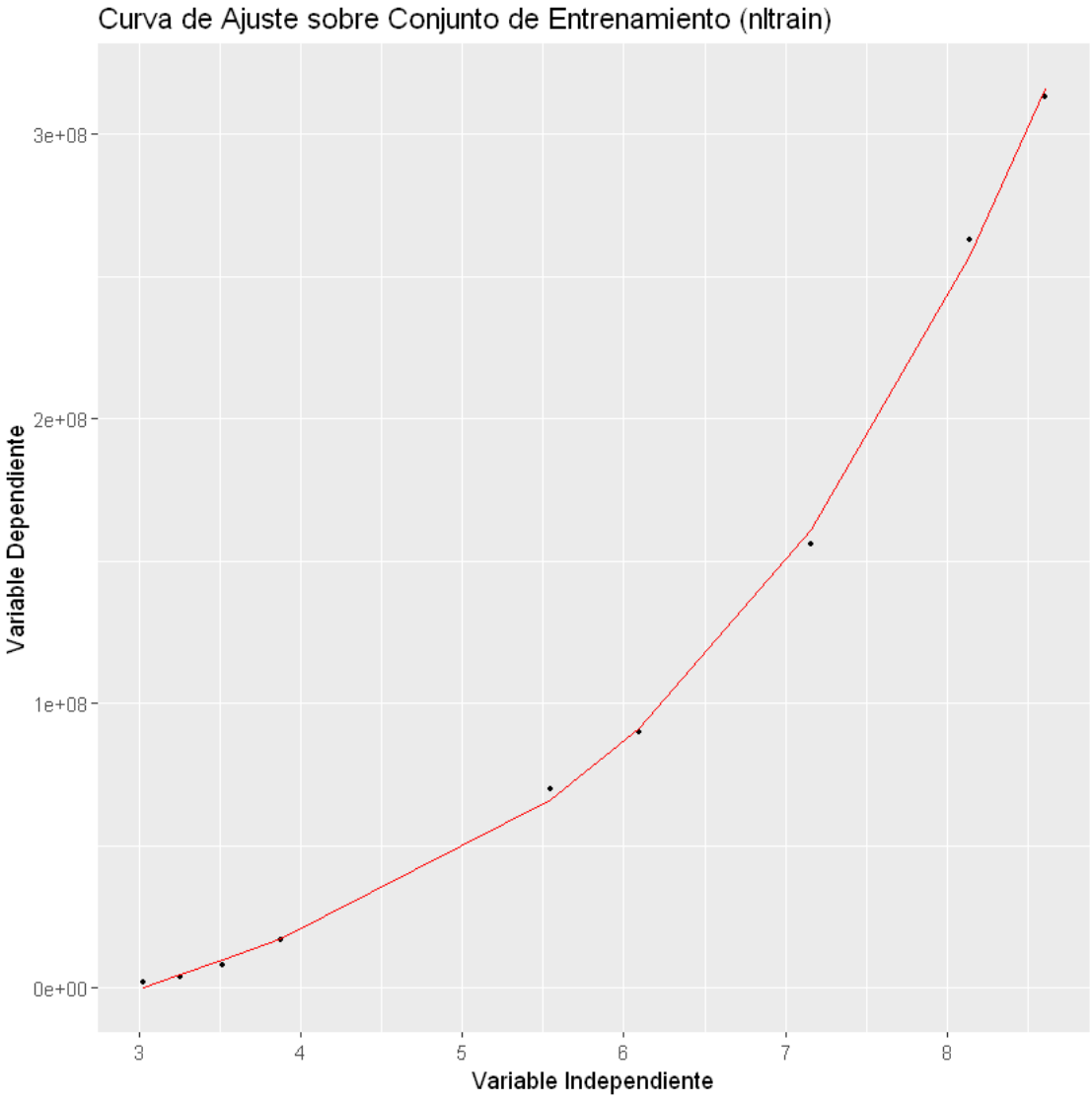



```
In [52]: nltrain$x3 <- nltrain$x^3
         regresion_poly <- lm(y ~ x + x2 + x3, data = nltrain)
         summary(regresion_poly)
```

```
Call:
lm(formula = y ~ x + x2 + x3, data = nltrain)

Residuals:
       1        2        3        4        7        8       10       12
 1860114  -691977 -1948857  -405158  4049699 -1200265 -4830283  5977657
      13
-2810931
```

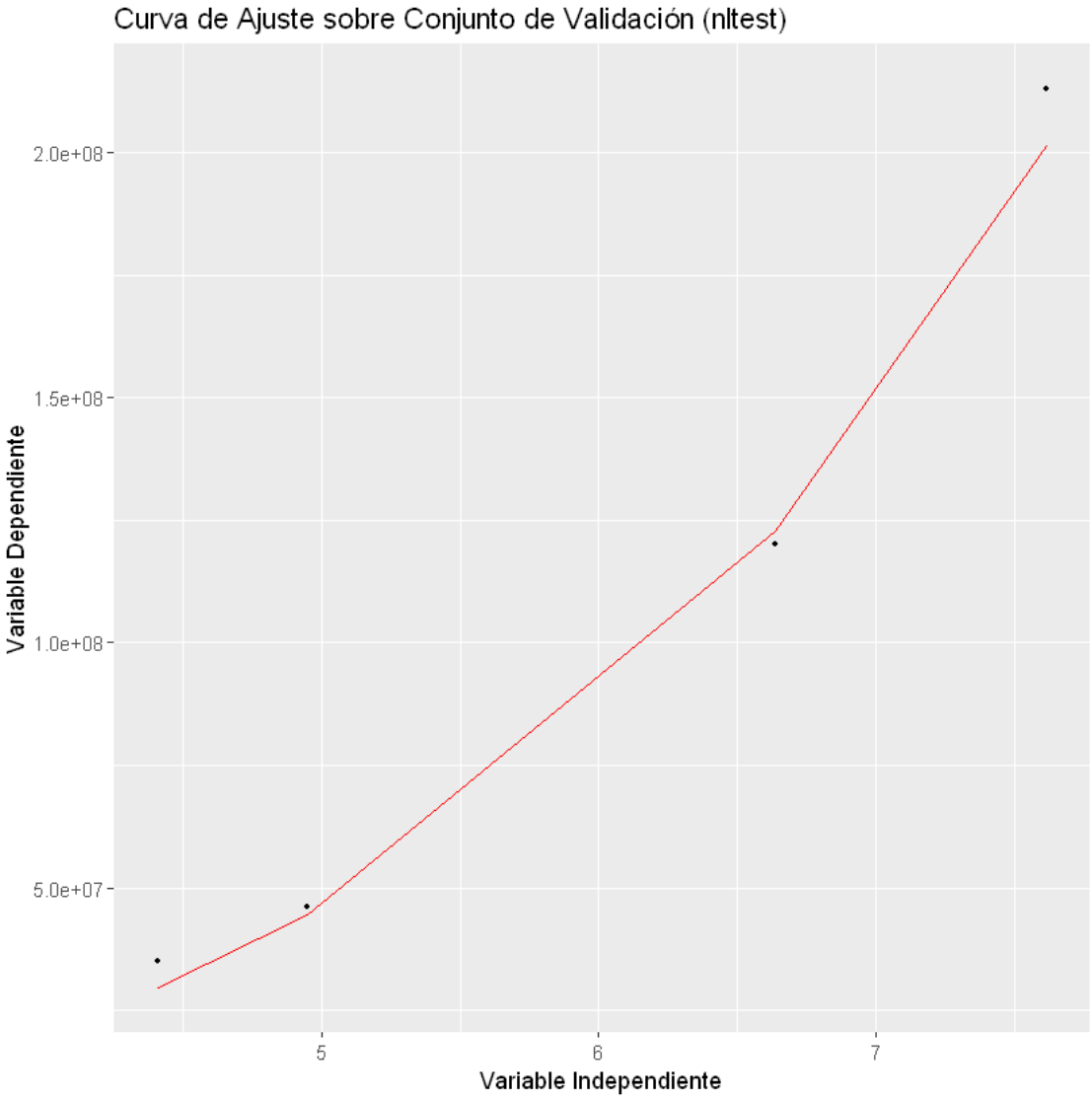
```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -109085931   69502154  -1.570   0.1773
x             64481673   40388244   1.597   0.1713
x2           -13514300    7270884  -1.859   0.1222
x3             1366013     412331   3.313   0.0212 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4306000 on 5 degrees of freedom
Multiple R-squared:  0.9992,Adjusted R-squared:  0.9987
F-statistic:  1979 on 3 and 5 DF,  p-value: 4.182e-08
```

```
In [53]: y_poly_predict <- predict(regresion_poly, nltrain)
         ggplot() +
         geom_point(data = nltrain, aes(x = nltrain$x, y = nltrain$y), size = 0.9) +
         geom_line(aes( x = nltrain$x, y = y_poly_predict), color = "red") +
           xlab("Variable Independiente") +
           ylab("Variable Dependiente") +
           ggtitle("Curva de Ajuste sobre Conjunto de Entrenamiento (nltrain)")
```




```
In [54]: nltest$x2 <- nltest$x^2
         nltest$x3 <- nltest$x^3
         y_poly_test_predict <- predict(regresion_poly, nltest)
         summary(y_poly_test_predict)

         ggplot() + geom_point(data = nltest, aes(x = x, y = y), size = 0.9) +
           geom_line(aes( x = nltest$x, y = y_poly_test_predict), color = "red") +
           xlab("Variable Independiente") +
           ylab("Variable Dependiente") +
           ggtitle("Curva de Ajuste sobre Conjunto de Validación (nltest)")
```

```
    Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
29655668  40880323  83753408  99608677 142481762 201272223
```




```
In [55]: predict_value_poly <- predict(regresion_poly, data.frame(x = 15,
                                                                  x2 = 15^2,
                                                                  x3 = 15^3))
         predict_value_poly
```

1: 2427714097.2266