TAREA_3

JOSE RICARDO MERIDA LOPEZ 23006385

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.2.3
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.2.3
```

```
library(corrplot)
```

```
## Warning: package 'corrplot' was built under R version 4.2.3
```

```
## corrplot 0.92 loaded
```

```
library(PerformanceAnalytics)
```

```
## Warning: package 'PerformanceAnalytics' was built under R version 4.2.3
```

```
## Loading required package: xts
```

```
## Warning: package 'xts' was built under R version 4.2.3
```

```
## Loading required package: zoo
```

```
## Warning: package 'zoo' was built under R version 4.2.3
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
##
## ############################ Warning from 'xts' package ##########################
## #                                                                                #
## # The dplyr lag() function breaks how base R's lag() function is supposed to  #
## # work, which breaks lag(my_xts). Calls to lag(my_xts) that you type or       #
## # source() into this session won't work correctly.                               #
## #                                                                                #
## # Use stats::lag() to make sure you're not using dplyr::lag(), or you can add #
## # conflictRules('dplyr', exclude = 'lag') to your .Rprofile to stop           #
```

```
## # dplyr from breaking base R's lag() function.                                 #
## #                                                                             #
## # Code in packages is not affected. It's protected by R's namespace mechanism #
## # Set `options(xts.warn_dplyr_breaks_lag = FALSE)` to suppress this warning.  #
## #                                                                             #
## ###############################################################################
```

```
##
## Attaching package: 'xts'
```

```
## The following objects are masked from 'package:dplyr':
##
##     first, last
```

```
##
## Attaching package: 'PerformanceAnalytics'
```

```
## The following object is masked from 'package:graphics':
##
##     legend
```

```
dataset<-read.csv("Admisions.csv")

print(dataset)
```

```
##     Serial.No. GRE.Score TOEFL.Score University.Rating SOP LOR CGPA Research
## 1            1       337         118                 4 4.5 4.5 9.65        1
## 2            2       324         107                 4 4.0 4.5 8.87        1
## 3            3       316         104                 3 3.0 3.5 8.00        1
## 4            4       322         110                 3 3.5 2.5 8.67        1
## 5            5       314         103                 2 2.0 3.0 8.21        0
## 6            6       330         115                 5 4.5 3.0 9.34        1
## 7            7       321         109                 3 3.0 4.0 8.20        1
## 8            8       308         101                 2 3.0 4.0 7.90        0
## 9            9       302         102                 1 2.0 1.5 8.00        0
## 10          10       323         108                 3 3.5 3.0 8.60        0
##     Chance.of.Admit
## 1              0.92
## 2              0.76
## 3              0.72
## 4              0.80
## 5              0.65
## 6              0.90
## 7              0.75
## 8              0.68
## 9              0.50
## 10             0.45
```

```
dataset<-dataset %>%
  select(-Serial.No.)

dataset
```

```
##     GRE.Score TOEFL.Score University.Rating SOP LOR CGPA Research
## 1         337         118                 4 4.5 4.5 9.65        1
## 2         324         107                 4 4.0 4.5 8.87        1
## 3         316         104                 3 3.0 3.5 8.00        1
## 4         322         110                 3 3.5 2.5 8.67        1
```

```
## 5          314          103                  2 2.0 3.0 8.21        0
## 6          330          115                  5 4.5 3.0 9.34        1
## 7          321          109                  3 3.0 4.0 8.20        1
## 8          308          101                  2 3.0 4.0 7.90        0
## 9          302          102                  1 2.0 1.5 8.00        0
## 10         323          108                  3 3.5 3.0 8.60        0
```

```
cor(dataset)
```

```
##                    GRE.Score TOEFL.Score University.Rating       SOP       LOR
## GRE.Score          1.0000000   0.8272004         0.6353762 0.6134977 0.5246794
## TOEFL.Score        0.8272004   1.0000000         0.6497992 0.6444104 0.5415633
## University.Rating 0.6353762   0.6497992         1.0000000 0.7280236 0.6086507
## SOP                0.6134977   0.6444104         0.7280236 1.0000000 0.6637069
## LOR                0.5246794   0.5415633         0.6086507 0.6637069 1.0000000
## CGPA               0.8258780   0.8105735         0.7052543 0.7121543 0.6374692
## Research           0.5633981   0.4670121         0.4270475 0.4081158 0.3725256
## Chance.of.Admit    0.8103506   0.7922276         0.6901324 0.6841365 0.6453645
##                         CGPA  Research Chance.of.Admit
## GRE.Score          0.8258780 0.5633981       0.8103506
## TOEFL.Score        0.8105735 0.4670121       0.7922276
## University.Rating 0.7052543 0.4270475       0.6901324
## SOP                0.7121543 0.4081158       0.6841365
## LOR                0.6374692 0.3725256       0.6453645
## CGPA               1.0000000 0.5013110       0.8824126
## Research           0.5013110 1.0000000       0.5458710
## Chance.of.Admit    0.8824126 0.5458710       1.0000000
```

```
corrplot(cor(dataset),
         method = "number",
         type = "upper")
```

```
chart.Correlation(dataset, histogram = TRUE, cex=1)
```

```
dataset<-dataset %>%
  select(GRE.Score,Chance.of.Admit)
```

———————— EJERICICIO 1 ————————-

———————— PREGUNTA 1 ————————-

Un arreglo con los valores de los estimadores para Bo y B1

———— RESPUESTA 1 ————————

```
n_1<-nrow(dataset)

dataset <- dataset %>% mutate(xy = GRE.Score * Chance.of.Admit)
dataset <- dataset %>% mutate(xx = GRE.Score * GRE.Score)

resum_1<-dataset %>%
  summarise(
    sum_x = sum(GRE.Score),
    sum_y = sum(Chance.of.Admit),
    sum_xy = sum(xy),
    sum_xx = sum(xx)
```

```
  )

x_sum<-as.double(resum_1$sum_x)
y_sum<-as.double(resum_1$sum_y)
xy_sum<-as.double(resum_1$sum_xy)
xx_sum<-as.double(resum_1$sum_xx)

beta_1<-((x_sum*y_sum-n_1*xy_sum)/(x_sum*x_sum-n_1*xx_sum))


print("BETA 1")
```

```
## [1] "BETA 1"
```

```
print(beta_1)
```

```
## [1] 0.01012587
```

```
beta_0<-((y_sum-beta_1*x_sum)/(n_1))

print("BETA 0")
```

```
## [1] "BETA 0"
```

```
print(beta_0)
```

```
## [1] -2.482815
```

——————— EJERICICIO 1 ———————-

——————— PREGUNTA 2 ———————-

El valor del coeficiente de determinación R^2 del modelo.

———— RESPUESTA 2 ————————

```
y_mean <- mean(dataset$Chance.of.Admit)

dataset <- dataset %>% mutate(y_test = beta_0+GRE.Score *beta_1)

dataset <- dataset %>% mutate(error_dif_y_ymean = (Chance.of.Admit-y_mean)^2)

dataset <- dataset %>% mutate(error_dif_ytest_ymean = (y_test-y_mean)^2)

dataset <- dataset %>% mutate(error_dif = abs(y_test-Chance.of.Admit))


resum_2<-dataset %>%
  summarise(
    sum_y_ymean = sum(error_dif_y_ymean),
    sum_ytest_ymean = sum(error_dif_ytest_ymean)
  )

y_ymean_sum<-as.double(resum_2$sum_y_ymean)
ytest_ymean_sum<-as.double(resum_2$sum_ytest_ymean)
```

```r
val_r2<-ytest_ymean_sum/y_ymean_sum

print(val_r2)
```

```
## [1] 0.6566682
```

———————— EJERICICIO 1 ————————-

———————— PREGUNTA 3 ————————-

El coeficiente de correlación r (raíz cuadrada de r#).

———————— RESPUESTA 3 ————————

```r
val_r<-sqrt(val_r2)
print(val_r)
```

```
## [1] 0.8103506
```

———————— EJERICICIO 1 ————————-

———————— PREGUNTA 4 ————————-

Un arreglo con los valores de los residuos.

———————— RESPUESTA 4 ————————

```r
print(dataset$error_dif)
```

```
##   [1] 9.603881e-03 3.796756e-02 3.039411e-03 2.228418e-02 4.670885e-02
##   [6] 4.127722e-02 1.758994e-02 4.404638e-02 7.519839e-02 3.378417e-01
```

```r
hist(dataset$error_dif)
```

———————— EJERICICIO 1 ————————- ## ———————— PREGUNTA 5 ————————- ## Una gráfica con la nube de puntos y la recta de regresión del modelo.

———————— RESPUESTA 5 ————————

```r
# Graficar la nube de puntos desde el dataframe
plot(dataset$GRE.Score, dataset$Chance.of.Admit, pch = 16, col = "blue", xlab = "x", ylab = "y", main =


abline(a = beta_0, b = beta_1, col = "red", lwd = 2)
```

———————— EJERICICIO 2 ————————-

———————— PREGUNTA 1 ————————-

Realice un análisis estadístico sobre todas las variables del dataset, recuerde que pude usar la función summary().

———————— RESPUESTA 1 ————————

```r
dataset<-read.csv("Admisions.csv")

dataset<-dataset %>%
  select(-Serial.No.)


lm1<-dataset %>%
  lm(formula = Chance.of.Admit ~ .)

summary(lm1)
```

```
##
## Call:
## lm(formula = Chance.of.Admit ~ ., data = .)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.266657 -0.023327  0.009191  0.033714  0.156818
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)       -1.2757251  0.1042962 -12.232  < 2e-16 ***
## GRE.Score          0.0018585  0.0005023   3.700 0.000240 ***
## TOEFL.Score        0.0027780  0.0008724   3.184 0.001544 **
## University.Rating  0.0059414  0.0038019   1.563 0.118753
## SOP                0.0015861  0.0045627   0.348 0.728263
## LOR                0.0168587  0.0041379   4.074 5.38e-05 ***
## CGPA               0.1183851  0.0097051  12.198  < 2e-16 ***
## Research           0.0243075  0.0066057   3.680 0.000259 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.05999 on 492 degrees of freedom
## Multiple R-squared:  0.8219, Adjusted R-squared:  0.8194
## F-statistic: 324.4 on 7 and 492 DF,  p-value: < 2.2e-16
```

——————— EJERICICIO 2 ———————-

——————— PREGUNTA 2 ———————-

Realice una gráfica de densidad para cada una de las variables numéricas en el dataset: GRE.Score, TOEFEL.Score, CGPA y Chance of Admit.

———— RESPUESTA 2 ———————-

```r
hist(dataset$GRE.Score, freq = FALSE, main = "Histograma y densidad",
     ylab = "Densidad")
```

```r
hist(dataset$TOEFL.Score, freq = FALSE, main = "Histograma y densidad",
     ylab = "Densidad")
```

```r
hist(dataset$CGPA, freq = FALSE, main = "Histograma y densidad",
     ylab = "Densidad")
```

```r
hist(dataset$Chance.of.Admit, freq = FALSE, main = "Histograma y densidad",
     ylab = "Densidad")
```

———————— EJERICICIO 2 ————————-

———————— PREGUNTA 3 ————————-

Realice una gráfica de correlación entre las variables del inciso anterior.

———————— RESPUESTA 3 ————————-

```r
dataset_1<- dataset %>%
  select(GRE.Score, TOEFL.Score, CGPA, Chance.of.Admit)

chart.Correlation(dataset_1, histogram = TRUE, cex=1)
```

```
## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter

## Warning in par(usr): argument 1 does not name a graphical parameter
```

———————— EJERICICIO 2 ————————- ## ———————— PREGUNTA 4 ————————- ## Realice comentarios sobre el análisis estadístico de las variables ## numéricas y la gráfica de correlación.

——————— RESPUESTA 4 ———————————-

En los graficos anteriores se puede observar que hay una fuerte correlacion entre las variables GRE.Score, TOEFL.Score, CGPA y la variable Chance.of.Admit, por lo que esta ultima si puede tener una alta dependencia de las variables anteriores.

——————— EJERICICIO 2 ———————————-

——————— PREGUNTA 5 ———————————-

Realice un scatter plot (nube de puntos) de todas las variables numéricas contra la variable Chance of Admit.

——————— RESPUESTA 5 ———————————-

```
# Crear un dataframe de ejemplo

# Establecer el diseño de la imagen
par(mfrow = c(3, 2))  # 2 filas y 2 columnas de paneles

# Graficar cada scatterplot en un panel separado
plot(dataset$Chance.of.Admit, dataset$GRE.Score, pch = 16, col = "blue", xlab = "x1", ylab = "y", main =
plot(dataset$TOEFL.Score, dataset$Chance.of.Admit, pch = 16, col = "red", xlab = "x2", ylab = "y", main
plot(dataset$Chance.of.Admit, dataset$University.Rating, pch = 16, col = "green", xlab = "x3", ylab = "
plot(dataset$Chance.of.Admit, dataset$SOP, pch = 16, col = "black", xlab = "x3", ylab = "y", main = "SO
plot(dataset$Chance.of.Admit, dataset$LOR, pch = 16, col = "cyan", xlab = "x3", ylab = "y", main = "LOR
plot(dataset$Chance.of.Admit, dataset$CGPA, pch = 16, col = "gray", xlab = "x3", ylab = "y", main = "CG
```

——————— EJERICICIO 2 ———————————-

——————— PREGUNTA 6 ———————————-

Utilizando la función train y trainControl para crear un crossvalidation y le permita evaluar los ## siguientes modelos:

- **Chance of Admit ~ TOEFEL.Score.**
- **Chance of Admit ~ CGPA.**
- **Chance of Admit ~ GRE.Score.**
- **Chance of Admit ~ TOEFEL.Score + CGPA.**
- **Chance of Admit ~ TOEFEL.Score + GRE.Score.**
- **Chance of Admit ~ GRE.Score + CGPA.**
- **Chance of Admit ~ TOEFEL.Score + CGPA + GRE.Score.**

——————— RESPUESTA 6 ———————————

- **Chance of Admit ~ TOEFEL.Score.**

```
lm1<-dataset %>%
  lm(formula = Chance.of.Admit ~ TOEFL.Score)
```

```
summ_lm1<-summary(lm1)
summ_lm1
```

```
##
## Call:
## lm(formula = Chance.of.Admit ~ TOEFL.Score, data = .)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.31337 -0.04990  0.01310  0.05633  0.20725
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.2489882  0.0681317  -18.33   <2e-16 ***
## TOEFL.Score  0.0183850  0.0006346   28.97   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08621 on 498 degrees of freedom
## Multiple R-squared:  0.6276, Adjusted R-squared:  0.6269
## F-statistic: 839.4 on 1 and 498 DF,  p-value: < 2.2e-16
```

- **Chance of Admit ~ CGPA.**

```
lm2<-dataset %>%
  lm(formula = Chance.of.Admit ~ CGPA)

summ_lm2<-summary(lm2)
summ_lm2
```

```
##
## Call:
## lm(formula = Chance.of.Admit ~ CGPA, data = .)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.276592 -0.028169  0.006619  0.038483  0.176961
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.04434    0.04230  -24.69   <2e-16 ***
## CGPA         0.20592    0.00492   41.85   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06647 on 498 degrees of freedom
## Multiple R-squared:  0.7787, Adjusted R-squared:  0.7782
## F-statistic:  1752 on 1 and 498 DF,  p-value: < 2.2e-16
```

- **Chance of Admit ~ GRE.Score.**

```
lm3<-dataset %>%
  lm(formula = Chance.of.Admit ~ GRE.Score)
```

```
summ_lm3<-summary(lm3)
summ_lm3
```

```
##
## Call:
## lm(formula = Chance.of.Admit ~ GRE.Score, data = .)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.33784 -0.04479  0.00417  0.05449  0.18568
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.4828147  0.1038994  -23.90   <2e-16 ***
## GRE.Score    0.0101259  0.0003281   30.86   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08278 on 498 degrees of freedom
## Multiple R-squared:  0.6567, Adjusted R-squared:  0.656
## F-statistic: 952.5 on 1 and 498 DF,  p-value: < 2.2e-16
```

- **Chance of Admit ~ TOEFEL.Score + CGPA**

```
lm4<-dataset %>%
  lm(formula = Chance.of.Admit ~ TOEFL.Score + CGPA)

summ_lm4<-summary(lm4)
summ_lm4
```

```
##
## Call:
## lm(formula = Chance.of.Admit ~ TOEFL.Score + CGPA, data = .)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.279799 -0.025222  0.007857  0.038331  0.142758
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.238516   0.050491 -24.529  < 2e-16 ***
## TOEFL.Score  0.005208   0.000803   6.486 2.13e-10 ***
## CGPA         0.163472   0.008075  20.245  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06389 on 497 degrees of freedom
## Multiple R-squared:  0.7959, Adjusted R-squared:  0.7951
## F-statistic: 969.2 on 2 and 497 DF,  p-value: < 2.2e-16
```

- **Chance of Admit ~ TOEFEL.Score + GRE.Score.**

```
lm5<-dataset %>%
  lm(formula = Chance.of.Admit ~ TOEFL.Score + GRE.Score)

summ_lm5<-summary(lm5)
summ_lm5
```

```
##
## Call:
## lm(formula = Chance.of.Admit ~ TOEFL.Score + GRE.Score, data = .)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.31903 -0.03969  0.01197  0.04904  0.15750
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.1802676  0.1024365 -21.284   <2e-16 ***
## TOEFL.Score  0.0089600  0.0010084   8.886   <2e-16 ***
## GRE.Score    0.0061350  0.0005429  11.300   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07698 on 497 degrees of freedom
## Multiple R-squared:  0.7037, Adjusted R-squared:  0.7025
## F-statistic: 590.3 on 2 and 497 DF,  p-value: < 2.2e-16
```

- **Chance of Admit ~ GRE.Score + CGPA.**

```
lm6<-dataset %>%
  lm(formula = Chance.of.Admit ~ GRE.Score + CGPA)

summ_lm6<-summary(lm6)
summ_lm6
```

```
##
## Call:
## lm(formula = Chance.of.Admit ~ GRE.Score + CGPA, data = .)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.296359 -0.024807  0.006763  0.038731  0.149210
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.634969   0.091336 -17.901  < 2e-16 ***
## GRE.Score    0.003207   0.000445   7.206 2.16e-12 ***
## CGPA         0.156464   0.008311  18.826  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06331 on 497 degrees of freedom
## Multiple R-squared:  0.7996, Adjusted R-squared:  0.7988
```

```
## F-statistic: 991.4 on 2 and 497 DF,  p-value: < 2.2e-16
```

- **Chance of Admit ~ TOEFEL.Score + CGPA + GRE.Score.**

```
lm7<-dataset %>%
  lm(formula = Chance.of.Admit ~ TOEFL.Score + GRE.Score + CGPA)

summ_lm7<-summary(lm7)
summ_lm7
```

```
##
## Call:
## lm(formula = Chance.of.Admit ~ TOEFL.Score + GRE.Score + CGPA,
##     data = .)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.293061 -0.020722  0.008274  0.036718  0.141429
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.5968093  0.0909032 -17.566  < 2e-16 ***
## TOEFL.Score  0.0031986  0.0008953   3.573 0.000388 ***
## GRE.Score    0.0023519  0.0005007   4.697 3.42e-06 ***
## CGPA         0.1435741  0.0089717  16.003  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06258 on 496 degrees of freedom
## Multiple R-squared:  0.8046, Adjusted R-squared:  0.8034
## F-statistic: 680.9 on 3 and 496 DF,  p-value: < 2.2e-16
```

——————— EJERICICIO 3 ———————-

——————— PREGUNTA 1 ———————-

Modelo #1:

——————— RESPUESTA 1 ———————

```
Call:
lm(formula = ROLL ~ UNEM, data = datavar)

Residuals:
     Min       1Q   Median       3Q      Max
-7640.0 -1046.5    602.8   1934.3   4187.2

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   3957.0     4000.1   0.989   0.3313
UNEM          1133.8      513.1   2.210   0.0358 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3049 on 27 degrees of freedom
Multiple R-squared: 0.1531,  Adjusted R-squared: 0.1218
F-statistic: 4.883 on 1 and 27 DF,  p-value: 0.03579
```

value 4 value 1 value 3 value 2

El valor 1 indica la probabilidad que tiene el modelo de que sea mayor a 2.5 el valor de t. El valor 2 debe ser mayor a 5. El valor 3 indica el R2, este debe ser mayor a 0.55 El valor 4 indica el error encontrado en el modelo, (la distancia que hay del modelo al punto lejano) Por lo que para el modelo 1 la variable UNEM, no es significativa para nuestro modelo.

El valor 1 indica la probabilidad que tiene el modelo de que sea mayor a 2.5 el valor de t. El valor 2 debe ser mayor a 5. El valor 3 indica el R2, este debe ser mayor a 0.55 El valor 4 indica el error encontrado en el modelo, (la distancia que hay del modelo al punto lejano) Por lo que para el modelo 2, las variables HGRAD, INC, aportan al modelo, en el caso de std. ERROR nos indica que el error es muy grande para este modelo, por lo que la variable UNEM, nos esta generando mucho ruido.

El valor 1 indica la probabilidad que tiene el modelo de que sea mayor a 2.5 el valor de t. El valor 2 debe ser mayor a 5. El valor 3 indica el R2, este debe ser mayor a 0.55 El valor 4 indica el error encontrado en el modelo, (la distancia que hay del modelo al punto lejano) Por lo que para el modelo 2, las variables HGRAD, INC, aportan al modelo, en el caso de std. ERROR nos indica que el error es muy grande para este modelo, por lo que la variable UNEM, nos esta generando mucho ruido.

```
Call:
lm(formula = ROLL ~ UNEM + HGRAD + INC, data = datavar)

Residuals:
      Min        1Q    Median        3Q       Max
-1148.840  -489.712    -1.876   387.400  1425.753

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -9.153e+03  1.053e+03  -8.691 5.02e-09 ***
UNEM         4.501e+02  1.182e+02   3.809 0.000807 ***
HGRAD        4.065e-01  7.602e-02   5.347 1.52e-05 ***
INC          4.275e+00  4.947e-01   8.642 5.59e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 670.4 on 25 degrees of freedom
Multiple R-squared: 0.9621,  Adjusted R-squared: 0.9576
F-statistic: 211.5 on 3 and 25 DF,  p-value: < 2.2e-16
```

value 4 (Min)
value 1 (Pr(>|t|))
value 3 (Residual standard error)
value 2 (F-statistic)

Figure 1: modelo_1

```
Call:
lm(formula = Cab.Price ~ Months, data = training_set)

Residuals:
    Min      1Q  Median      3Q     Max
-11.034  -2.305  -1.034   2.764   9.241

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  72.6826     3.2377   22.45 6.92e-10 ***
Months        4.8626     0.3495   13.91 7.18e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.657 on 10 degrees of freedom
Multiple R-squared:  0.9509,    Adjusted R-squared:  0.946
F-statistic: 193.6 on 1 and 10 DF,  p-value: 7.181e-08
```

value 4 (Min)
value 1 (Pr(>|t|))
value 3 (Residual standard error)
value 2 (F-statistic)

El valor 1 indica la probabilidad que tiene el modelo de que sea mayor a 2.5 el valor de t. El valor 2 debe ser

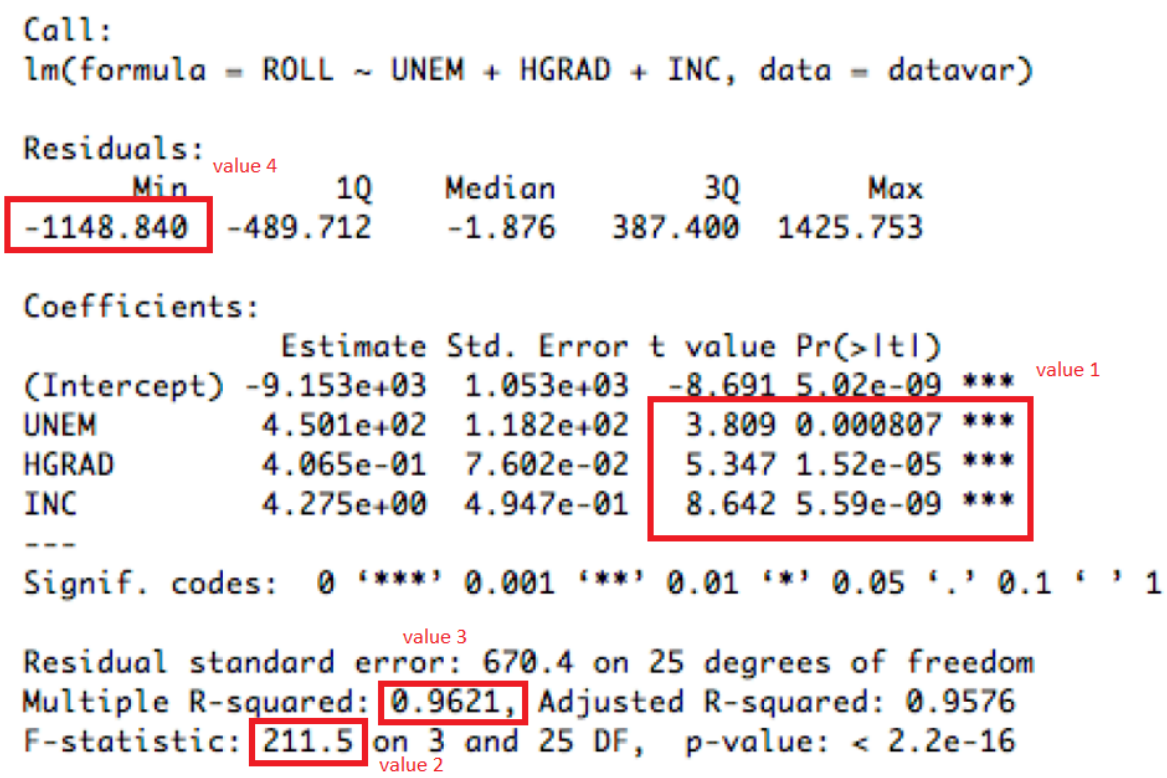

Figure 2: modelo_2

mayor a 5. El valor 3 indica el R2, este debe ser mayor a 0.55 El valor 4 indica el error encontrado en el modelo, (la distancia que hay del modelo al punto lejano) Por lo que para el modelo 2, la variable Months si es significante para el modelo, ya que cumple con estos parametros.