“Tarea 3”

Mirtha Ayala

Plantear el problema de regresión como un problema de mínimos cuadrados, encontrar el vector $\hat{\beta}$ que resuelva $$\beta=argmin_{\beta \in R^p}||Y-X\beta||^2$$ y encontrar la solución teórica.

¿Por qué este planteamiento nos da un ajuste lineal a nuestros datos?

El método de Mínimos Cuadrados es un problema de ajuste lineal dado que la función que aproxima es lineal en los parámetros. Por tanto estos explicarán únicamente la relación lineal entre las distintas X_{i} con Y

¿Podríamos usarlo para ajustar polinomios (ej $y=x^2$)?

Sí es posible

Argumentar la relación entre la solución encontrada y un problema de proyección en subespacios vectoriales de álgebra lineal. ¿Cuál es la relación particular con el teorema de Pitágoras?

La variabilidad de la variable dependiente puede dividirse en dos componentes: 1. la referente a su relación con las variables indepenientes 2.la de los residuos. Esta descomposición de la varianza de la variable dependiente en dos varianzas es el “Teorema de Pitágoras” del Análisis de Regresión Lineal que, para efectos del modelo anterior, la varianza de las puntuaciones observadas es igual a la varianza de las puntuaciones estimadas más la varianza de los residuos.

¿Qué logramos al agregar una columna de unos en la matriz ?

Permite incluir la información que aporta β_{0}. Se trata de información adicional que las variables independientes o explicativas no aportan.

Plantear el problema de regresión ahora como un problema de estadística donde los errores son no correlacionados con distribución

\[\varepsilon_{i}\sim N(0, \sigma)\]

¿Cuál es la función de verosimilitud del problema anterior? Hint: empiecen por escribir el problema como Mostrar que la solución de máxima verosimilitud es la misma que la del problema de mínimos cuadrados.

Investiga el contenido del Teorema de Gauss-Markov sobre mínimos cuadrados.

El Teorema de Gauss-Márkov establece que en un modelo lineal general (MLG) en el que se cumplan los siguientes supuestos: - Correcta especificación: el MLG ha de ser una combinación lineal de los parámetros\(\beta\) y no necesariamente de las variables: \(Y=X\beta+u\) - Muestreo aleatorio simple: la muestra de observaciones del vector \((y_{i},\,x_{2i},\,x_{3i},\,\dots ,\,x_{ki})\) es una muestra aleatoria simple y, por lo tanto, el vector \((y_{i},X'_{i})\) es independiente del vector \((y_{i},X'_{j})\) - Esperanza condicionada de los errores nula: \(E(u_{i}|X'_{i})=0\) - Correcta identificación: la matriz de regresoras (X) ha de tener rango completo: rg(X)=K<=N - Homocedasticidad: la varianza del error condicional a las variables explicativas es constante a lo largo de las observaciones:\(Var(U|X)=\alpha^{2}I\)

El estimador mínimo cuadrático ordinario (MCO) de \(\beta\) es el estimador lineal e insesgado óptimo, es decir, el estimador MCO es el estimador eficiente dentro de la clase de estimadores lineales e insesgados.

```
##Parte Aplicada

#install.packages("ggplot2")

# Cargar la base diamonds que se encuentra en el paquete ggplot2 .
library("ggplot2")
```

```
## Warning: package 'ggplot2' was built under R version 3.3.3
```

```
require(ggplot2)
data(diamonds)
names(diamonds)
```

```
##  [1] "carat"   "cut"     "color"   "clarity" "depth"   "table"   "price"
##  [8] "x"       "y"       "z"
```

```
head(diamonds)
```

```
## # A tibble: 6 × 10
##   carat       cut color clarity depth table price     x     y     z
##   <dbl>     <ord> <ord>   <ord> <dbl> <dbl> <int> <dbl> <dbl> <dbl>
## 1  0.23     Ideal     E     SI2  61.5    55   326  3.95  3.98  2.43
## 2  0.21   Premium     E     SI1  59.8    61   326  3.89  3.84  2.31
## 3  0.23      Good     E     VS1  56.9    65   327  4.05  4.07  2.31
## 4  0.29   Premium     I     VS2  62.4    58   334  4.20  4.23  2.63
## 5  0.31      Good     J     SI2  63.3    58   335  4.34  4.35  2.75
## 6  0.24 Very Good     J    VVS2  62.8    57   336  3.94  3.96  2.48
```

```
str(diamonds)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    53940 obs. of  10 variables:
##  $ carat  : num  0.23 0.21 0.23 0.29 0.31 0.24 0.24 0.26 0.22 0.23 ...
##  $ cut    : Ord.factor w/ 5 levels "Fair"<"Good"<..: 5 4 2 4 2 3 3 3 1 3 ...
##  $ color  : Ord.factor w/ 7 levels "D"<"E"<"F"<"G"<..: 2 2 2 6 7 7 6 5 2
5 ...
##  $ clarity: Ord.factor w/ 8 levels "I1"<"SI2"<"SI1"<..: 2 3 5 4 2 6 7 3 4
5 ...
##  $ depth  : num  61.5 59.8 56.9 62.4 63.3 62.8 62.3 61.9 65.1 59.4 ...
##  $ table  : num  55 61 65 58 58 57 57 55 61 61 ...
##  $ price  : int  326 326 327 334 335 336 336 337 337 338 ...
##  $ x      : num  3.95 3.89 4.05 4.2 4.34 3.94 3.95 4.07 3.87 4 ...
##  $ y      : num  3.98 3.84 4.07 4.23 4.35 3.96 3.98 4.11 3.78 4.05 ...
##  $ z      : num  2.43 2.31 2.31 2.63 2.75 2.48 2.47 2.53 2.49 2.39 ...
```

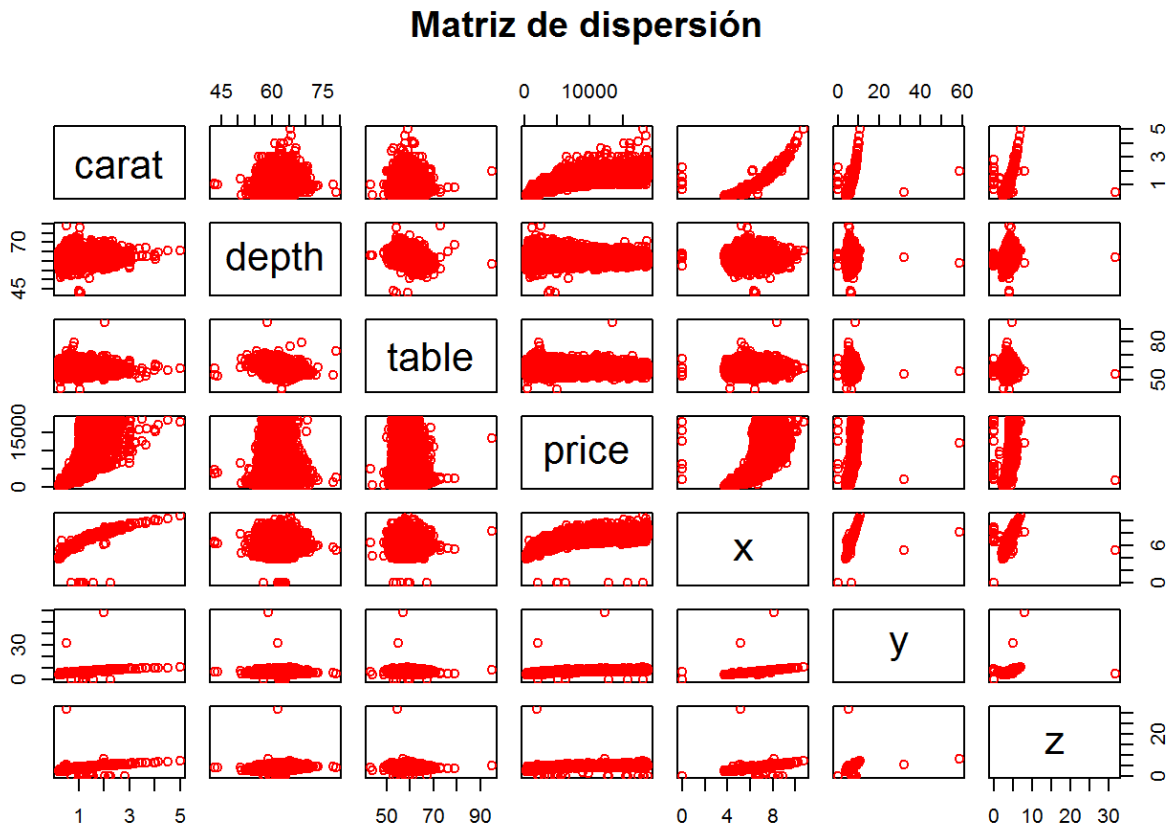
```
#Regresion Lineal para estimar el Precio (Y= Price) y eliminar las variables ca
tegóricas
diamonds2 <-subset(diamonds,select = -c(2,3,4))
head(diamonds2)
```

```
## # A tibble: 6 × 7
##   carat depth table price     x     y     z
##   <dbl> <dbl> <dbl> <int> <dbl> <dbl> <dbl>
## 1  0.23  61.5    55   326  3.95  3.98  2.43
## 2  0.21  59.8    61   326  3.89  3.84  2.31
## 3  0.23  56.9    65   327  4.05  4.07  2.31
## 4  0.29  62.4    58   334  4.20  4.23  2.63
## 5  0.31  63.3    58   335  4.34  4.35  2.75
## 6  0.24  62.8    57   336  3.94  3.96  2.48
```

```
diamonds2<- as.data.frame(diamonds2)

#Generamos la matriz de dispersión y correlación para identificar las variable
s importantes en términos de que explican la variabilidad del Precio

pairs(diamonds2, col= "red", main="Matriz de dispersión")
```




```
cor(diamonds2$price,diamonds2$carat)
```

```
## [1] 0.9215913
```

```
library(Hmisc)
```

```
## Warning: package 'Hmisc' was built under R version 3.3.3
```

```
## Loading required package: lattice
```

```
## Loading required package: survival
```

```
## Warning: package 'survival' was built under R version 3.3.3
```

```
## Loading required package: Formula
```

```
##
## Attaching package: 'Hmisc'
```

```
## The following objects are masked from 'package:base':
##
##     format.pval, round.POSIXt, trunc.POSIXt, units
```

```
rcorr(as.matrix(diamonds2))
```

```
##       carat depth table price     x     y    z
## carat  1.00  0.03  0.18  0.92  0.98  0.95 0.95
## depth  0.03  1.00 -0.30 -0.01 -0.03 -0.03 0.09
## table  0.18 -0.30  1.00  0.13  0.20  0.18 0.15
## price  0.92 -0.01  0.13  1.00  0.88  0.87 0.86
## x      0.98 -0.03  0.20  0.88  1.00  0.97 0.97
## y      0.95 -0.03  0.18  0.87  0.97  1.00 0.95
## z      0.95  0.09  0.15  0.86  0.97  0.95 1.00
##
## n= 53940
##
##
## P
##       carat  depth  table  price  x      y      z
## carat        0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
## depth 0.0000        0.0000 0.0134 0.0000 0.0000 0.0000
## table 0.0000 0.0000        0.0000 0.0000 0.0000 0.0000
## price 0.0000 0.0134 0.0000        0.0000 0.0000 0.0000
## x     0.0000 0.0000 0.0000 0.0000        0.0000 0.0000
## y     0.0000 0.0000 0.0000 0.0000 0.0000        0.0000
## z     0.0000 0.0000 0.0000 0.0000 0.0000 0.0000
```

De acuerdo a la matriz de correlación podemos pensar que Carat, X, Y y Z son las varibles que nos ayudarán a explicar el Precio aunque incluir todas puede generarnos problemas de colinealidad

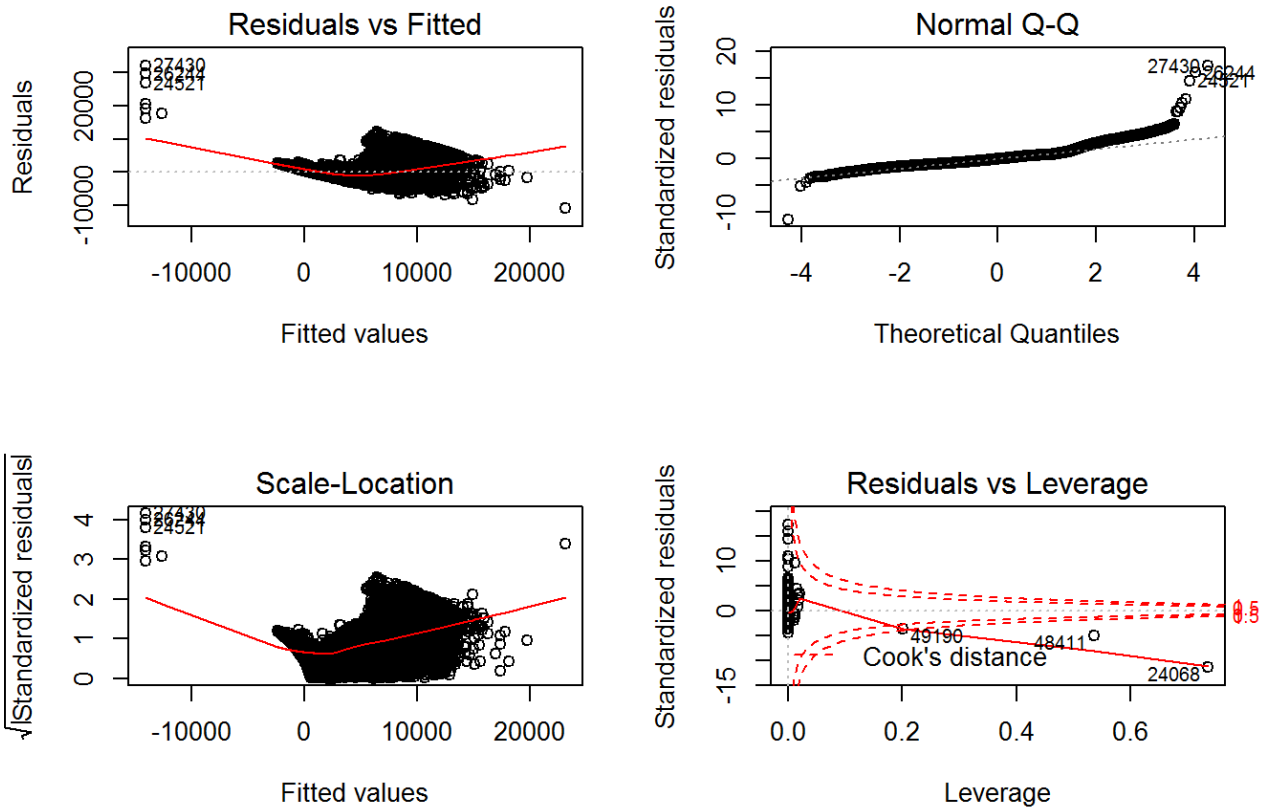
```
#Generar la regresión con todas las vars numéricas
PriceModel = lm(price ~ carat + depth + table + x + y + z, data = diamonds2)
summary(PriceModel)
```

```
##
## Call:
## lm(formula = price ~ carat + depth + table + x + y + z, data = diamonds2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -23878.2   -615.0    -50.7    347.9  12759.2
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 20849.316    447.562  46.584  < 2e-16 ***
## carat       10686.309     63.201 169.085  < 2e-16 ***
## depth        -203.154      5.504 -36.910  < 2e-16 ***
## table        -102.446      3.084 -33.216  < 2e-16 ***
## x           -1315.668     43.070 -30.547  < 2e-16 ***
## y              66.322     25.523   2.599  0.00937 **
## z              41.628     44.305   0.940  0.34744
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1497 on 53933 degrees of freedom
## Multiple R-squared:  0.8592, Adjusted R-squared:  0.8592
## F-statistic: 5.486e+04 on 6 and 53933 DF,  p-value: < 2.2e-16
```

```
PriceModel2 = lm(price ~ carat + y + z, data = diamonds2)
summary(PriceModel2)
```

```
##
## Call:
## lm(formula = price ~ carat + y + z, data = diamonds2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -21756.4   -693.7    -11.8    410.4  27761.0
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   644.55      86.87   7.420 1.19e-13 ***
## carat        9466.12      52.44 180.508  < 2e-16 ***
## y            -152.31      21.46  -7.099 1.28e-12 ***
## z            -958.46      35.33 -27.132  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1531 on 53936 degrees of freedom
## Multiple R-squared:  0.8528, Adjusted R-squared:  0.8528
## F-statistic: 1.041e+05 on 3 and 53936 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(2,2))
plot(lm(price ~ x + y + z, data = diamonds2))
```



¿Qué tan bueno fue el ajuste? Una buena respuesta incluye argumentaciones teóricas y visualizaciones. Puntos adicionales si investigan como usar alguna de las librerias ggplot2 o plotly para sus gráficas.

¿Qué tan bueno fue el ajuste? ¿Qué medida puede ayudarnos a saber la calidad del ajuste? \(R^{2}\) ¿Cuál fue el valor de que ajustó su modelo y que relación tiene con la calidad del ajuste?

R^{2}=.85 Se interpreta como la varianza explicada por el modelo o el porcentaje de variabilidad del Precio que es explicada por las variables incluida en el modelo. Dado que toma valores entre 0 y 1 evalor resultante (.85) puede considerarse como muy alto.

Dada la alta correlación entre la variable X y Carat eliminé X y la R^{2} ajustada disminuyó mínimamente. Como se observa en el Summary y de acuerdo al p-value, las variables Y, Z y Carat son significativas (***). Adicional, con el análsis gráfico de los residuales considero que se cumplen los supuestos de aleatoriedad, normalidad y homecedaticidad lo que contribuye a validar los resultados observados en nuestra regresión.

¿Cuál es el ángulo entre \(Y y Y\widehat\)?. Hint: usen la y el arcocoseno.

```
angulo <- acos(sqrt(.8528))
angulo * 180/pi
```

```
## [1] 22.56098
```

Defininan una funcion que calcule la logverosimilitud de unos parámetros