Análisis de regresión

Capítulo 8: Multicolinealidad

Mg. Sc. J. Eduardo Gamboa U.

Introducción

Una situación ideal para un modelo de regresión lineal implicaría que las variables explicativas sean independientes entre ellas, lo que sería equivalente a decir que estas variables son ortogonales.

Sin embargo, esto no ocurre en la realidad, y más bien las regresoras suelen estar asociadas, lo que denominamos colinealidad. Si este fenómeno sucede en varias variables, hablamos de multicolinealidad.

Si existe dependencia lineal entre variables explicativas, esto se verá reflejado en la matriz $\mathbf{X}$ del modelo y no existirá $(\mathbf{X}'\mathbf{X})^{-1}$, en consecuencia no se podrá estimar el vector de coeficientes β.

Por ejemplo, intente invertir $(\mathbf{X}'\mathbf{X})^{-1}$, siendo $\mathbf{X}$ la siguiente matriz:

```
(data.frame(x1 = c(6,6,4,7,4,2),
x2 = c(1,4,5,9,10,5),
x3 = c(6,6,4,7,4,2) + c(1,4,5,9,10,5)) |> as.matrix() -> X1)
```

```
     x1 x2 x3
[1,]  6  1  7
[2,]  6  4 10
[3,]  4  5  9
[4,]  7  9 16
[5,]  4 10 14
[6,]  2  5  7
```

Causas

La multicolinealidad puede originarse si se usa un conjunto incorrecto de variables, si se usa un modelo polinomial no ortogonal (siendo mayor el efecto de colinealidad si el rango de x es corto) o si el modelo está sobredefinido (se tiene más variables que observaciones).

Por ejemplo, intente invertir las siguientes matrices:

```
options(scipen=999)
(data.frame(x1 = c(0.01,4,7,25, 600),
x2 = c(0.01,4,7,25, 600)**2,
x3 = c(0.01,4,7,25, 600)**4) |> as.matrix() -> X2)
```

```
         x1          x2                   x3
[1,]   0.01      0.0001            0.00000001
[2,]   4.00     16.0000          256.00000000
[3,]   7.00     49.0000         2401.00000000
[4,]  25.00    625.0000       390625.00000000
[5,] 600.00 360000.0000 129600000000.00000000
```

```
(data.frame(x1 = c(3,6,4),
x2 = c(3,5,7),
x3 = c(2,7,9),
x4 = c(1,1,9),
x5 = c(6,8,18))|> as.matrix() -> X3)
```

```
     x1 x2 x3 x4 x5
[1,]  3  3  2  1  6
[2,]  6  5  7  1  8
[3,]  4  7  9  9 18
```

Consecuencias

La multicolinealidad puede tener varios efectos negativos en un modelo de regresión lineal. En primer lugar, puede hacer que los coeficientes estimados sean inestables contradictorios (signos opuestos a lo esperado) y difíciles de interpretar. Esto se debe a que la multicolinealidad aumenta la varianza de los coeficientes, lo que resulta en grandes fluctuaciones en sus valores cuando se hacen pequeños cambios en los datos de entrada.

Asimismo, la multicolinealidad también puede afectar negativamente la capacidad predictiva del modelo. Cuando las variables regresoras están altamente correlacionadas, el modelo puede sobreajustarse a los datos de entrenamiento y tener dificultades para generalizar correctamente a nuevos datos.

Diagnóstico

1. Explorar la matriz de correlación en búsqueda de valores altos y/o gráficos de dispersión en pares (solo para colinealidad).

```
cor(X1)
```

```
             x1          x2        x3
x1  1.00000000 -0.04368781 0.4531628
x2 -0.04368781  1.00000000 0.8707790
x3  0.45316278  0.87077899 1.0000000
```

```
cor(X2)
```

```
          x1        x2        x3
x1 1.0000000 0.9994024 0.9993461
x2 0.9994024 1.0000000 0.9999987
x3 0.9993461 0.9999987 1.0000000
```

```
cor(X3)
```

```
           x1         x2         x3         x4          x5
x1  1.00000000 0.3273268 0.5447048 -0.1889822 -0.03394221
x2  0.32732684 1.0000000 0.9707253  0.8660254  0.93325653
x3  0.54470478 0.9707253 1.0000000  0.7205767  0.81965616
x4 -0.18898224 0.8660254 0.7205767  1.0000000  0.98782916
x5 -0.03394221 0.9332565 0.8196562  0.9878292  1.00000000
```

2. Número de condición: Viene dado por la raíz cuadrada del ratio del mayor y menor autovalor de la matriz de correlación de las variables predictoras estandarizadas $k = \sqrt{\frac{\lambda_{max}}{\lambda_{min}}}$. Su valor debe ser menor a 30, mejor si es menor a 10. Cuando se estima este valor para cada variable del modelo de regresión, se tienen los índices de condición: $k_j = \sqrt{\frac{\lambda_{max}}{\lambda_j}}$, donde λ_j es el j-ésimo autovalor de $(\mathbf{X}'\mathbf{X})$. Los números de condición mostrados para las matrices de ejemplo. ¿Por qué no se ha podido calcular el número de condición en el tercer caso?

```
eigen(cor(scale(X1)))$values |> max() -> lmax1
eigen(cor(scale(X1)))$values |> min() -> lmin1
sqrt(lmax1/lmin1)
```

```
[1] 31351346
```

```
eigen(cor(scale(X2)))$values |> max() -> lmax2
eigen(cor(scale(X2)))$values |> min() -> lmin2
sqrt(lmax2/lmin2)
```

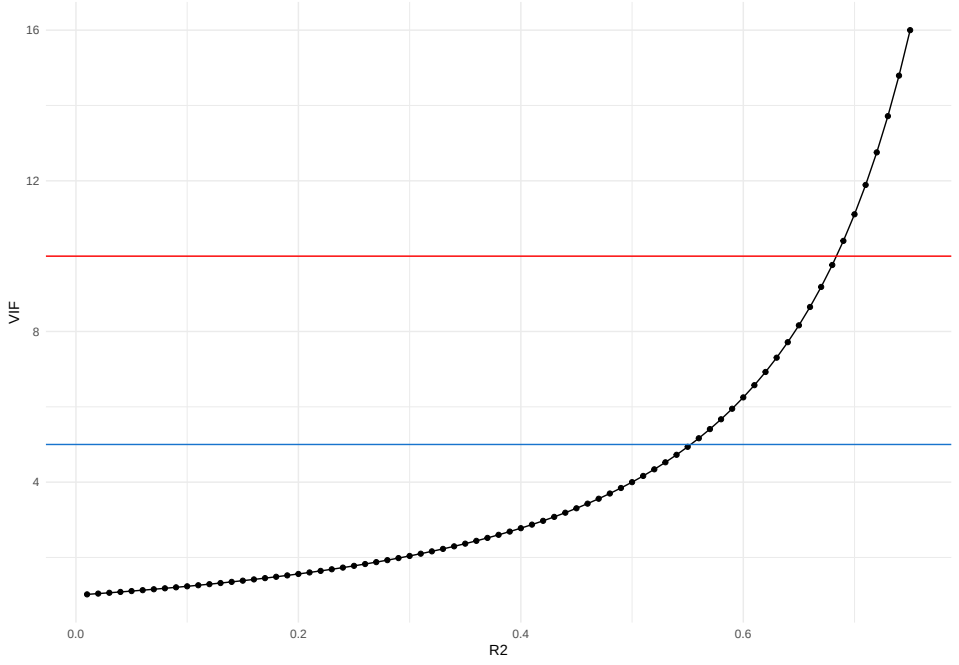
```
[1] 7497.039
```

```
eigen(cor(scale(X3)))$values |> max() -> lmax3
eigen(cor(scale(X3)))$values |> min() -> lmin3
sqrt(lmax3/lmin3)
```

```
[1] NaN
```

3. Factor de inflación de varianza (VIF): El VIF de la j-ésima variable viene dado por el j-ésimo término de la diagonal de $(\mathbf{X}'\mathbf{X})^{-1}$. También se puede hallar como $VIF_j = \frac{1}{1-R_j^2}$ donde R_j^2 es el coeficiente de determinación usando la j-ésima variable predictora como respuesta, y las demás predictoras como independientes. Su valor debe ser cercano a 1. Si es mayor a 5 es indicador de multicolinealidad. Si es superior a 10, esta multicolinealidad. es muy fuerte. Note la relación entre R^2 y VIF.

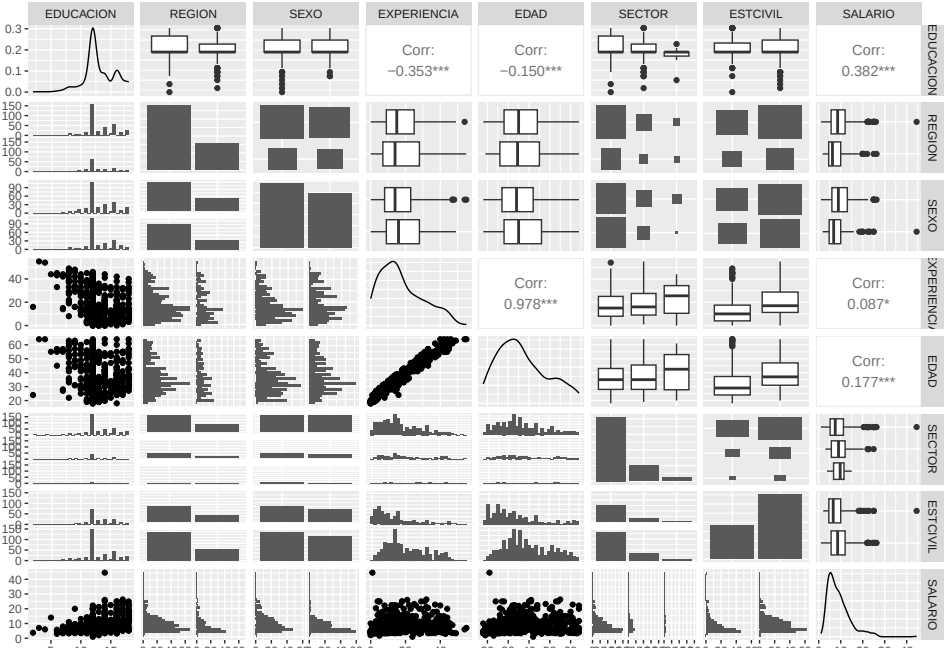
```r
R2 = seq(0.01,0.75,0.01)
VIF = 1/(1-R2)**2
library(ggplot2)
data.frame(R2,VIF) |>
ggplot(aes(x = R2, y = VIF))+
  geom_line()+
  geom_point()+
  geom_hline(yintercept = 5,colour="dodgerblue3")+
  geom_hline(yintercept = 10,colour="red")+
  theme_minimal()
```


16
12
8
4
VIF
0.0
0.2
0.4
0.6
R2

Ejemplo

Se busca estimar el salario medio de una persona en función a un conjunto de variables, compuesto por su educación (en años), región de residencia (0=No vive en el sur, 1=Sí vive en el sur), sexo (0=Masculino, 1=Femenino), experiencia (en años), edad (en años), sector laboral (0=Otro,1=Manufactura,2=Construcción) y estado civil (0=Soltero,1=Casado). Los datos se encuentran en el archivo **U8_datos_1.xlsx**

```
library(readxl)
library(dplyr)
library(GGally)
read_excel('U8_datos_1.xlsx') |>
  mutate(REGION = as.factor(REGION),
         SEXO = as.factor(SEXO),
         SECTOR = as.factor(SECTOR),
         ESTCIVIL = as.factor(ESTCIVIL)) -> datos
datos |> ggpairs()
```


EDUCACION
REGION
SEXO
EXPERIENCIA
EDAD
SECTOR
ESTCIVIL
SALARIO
Corr:
−0.353***
Corr:
−0.150***
Corr:
0.382***
Corr:
0.978***
Corr:
0.087*
Corr:
0.177***

Luego, vamos a realizar un proceso de selección de variables como vimos en la clase anterior

```
lm(SALARIO ~ ., datos) -> modelo
library(olsrr)
```

```
modelo |> ols_step_backward_p(prem = 0.10)
modelo |> ols_step_backward_aic()
modelo |> ols_step_forward_p(prem = 0.10)
modelo |> ols_step_forward_aic()
```

En base a lo obtenido, nos quedamos con el modelo que contiene las variables explicativas: Educacion, Experiencia, Sexo y Region.

Ahora vamos a analizar utilizando el concepto de multicolinealidad. Encontrando el número de condición así como los índices de condición:

```
modelo |> model.matrix() -> X
eigen(cor(scale(X[,-1])))$values |> max() -> lambda_max
eigen(cor(scale(X[,-1])))$values |> min() -> lambda_min
sqrt(lambda_max/lambda_min)
```

```
[1] 150.1125
```

```r
data.frame(variables = (modelo |> coef() |> names())[-1],
indices = sqrt(lambda_max/eigen(cor(scale(X[,-1])))$values))
```

```
    variables    indices
1   EDUCACION   1.000000
2     REGION1   1.341434
3       SEXO1   1.434928
4 EXPERIENCIA   1.462339
5        EDAD   1.573416
6     SECTOR1   1.685379
7     SECTOR2   1.826577
8   ESTCIVIL1 150.112465
```

```r
library(car)
modelo |> vif()
```

```
                    GVIF Df GVIF^(1/(2*Df))
EDUCACION     230.963848  1       15.197495
REGION          1.032908  1        1.016321
SEXO            1.046920  1        1.023191
EXPERIENCIA  5175.071174  1       71.937968
EDAD         4636.649246  1       68.092946
SECTOR          1.088093  2        1.021331
ESTCIVIL        1.088342  1        1.043236
```

```
# A tibble: 9 x 5
  term        estimate std.error statistic      p.value
  <chr>          <dbl>     <dbl>     <dbl>        <dbl>
1 (Intercept)   -1.84      6.80     -0.271 0.787
2 EDUCACION      1.33      1.12      1.19  0.233
3 REGION1       -0.835     0.429    -1.95  0.0519
4 SEXO1         -2.26      0.394    -5.73  0.0000000166
5 EXPERIENCIA    0.505     1.12      0.453 0.651
6 EDAD          -0.401     1.11     -0.360 0.719
7 SECTOR1        1.03      0.504     2.03  0.0424
8 SECTOR2        0.611     0.958     0.638 0.524
9 ESTCIVIL1      0.538     0.421     1.28  0.202
```

El modelo presenta un fuerte problema de multicolinealidad ya que el número de condición 150.1125 es mayor a 30. La experiencia, edad, sector y estado civil muestran multicolinealidad, siendo mayor en LA PRIMERA variable. Es probable que sus errores estándar se encuentren inflados. Por otro lado, al analizar el VIF, optamos por retirar la experiencia, pues es la variable que presenta mayor factor de inflación de varianza (71.937). Este indicador también muestra problemas para la edad y la educación, mas no para el estado civil

Pasamos a analizar el modelo sin la variable Experiencia

```
modelo2 = lm(SALARIO ~ .-EXPERIENCIA, datos)
modelo2 |> model.matrix() -> X
eigen(cor(scale(X[,-1])))$values |> max() -> lambda_max
eigen(cor(scale(X[,-1])))$values |> min() -> lambda_min
sqrt(lambda_max/lambda_min)
```

```
[1] 1.451017
```

```r
data.frame(variables = (modelo2 |> coef() |> names())[-1],
indices = sqrt(lambda_max/eigen(cor(scale(X[,-1])))$values))
```

```
  variables  indices
1 EDUCACION 1.000000
2   REGION1 1.062505
3     SEXO1 1.130698
4      EDAD 1.151083
5   SECTOR1 1.254635
6   SECTOR2 1.412441
7 ESTCIVIL1 1.451017
```

```
modelo2 |> vif()
```

```
              GVIF Df GVIF^(1/(2*Df))
EDUCACION 1.082372  1        1.040371
REGION    1.032283  1        1.016014
SEXO      1.044137  1        1.021830
EDAD      1.122769  1        1.059608
SECTOR    1.087861  2        1.021277
ESTCIVIL  1.086477  1        1.042342
```

Vemos que el número de condición disminuyó de 150.1125 a 1.45. Vemos además que ninguna variable presenta problemas de multicolinealidad (el número de condición así como los VIF están en el rango adecuado).

```
modelo2 |> tidy()
```

```
# A tibble: 8 x 5
  term        estimate std.error statistic  p.value
  <chr>          <dbl>     <dbl>     <dbl>    <dbl>
1 (Intercept)   -4.87     1.31      -3.72  2.22e- 4
2 EDUCACION      0.829    0.0763    10.9   6.19e-25
3 REGION1       -0.840    0.428     -1.96  5.04e- 2
4 SEXO1         -2.25     0.393     -5.72  1.77e- 8
5 EDAD           0.104    0.0173     5.98  4.23e- 9
6 SECTOR1        1.02     0.504      2.03  4.27e- 2
7 SECTOR2        0.616    0.957      0.643 5.20e- 1
8 ESTCIVIL1      0.530    0.420      1.26  2.08e- 1
```

Evaluando los cambios: Errores estándar

```
modelo |> tidy() |> select(term,std.error)
```

```
# A tibble: 9 x 2
  term        std.error
  <chr>           <dbl>
1 (Intercept)     6.80
2 EDUCACION       1.12
3 REGION1         0.429
4 SEXO1           0.394
5 EXPERIENCIA     1.12
6 EDAD            1.11
7 SECTOR1         0.504
8 SECTOR2         0.958
9 ESTCIVIL1       0.421
```

```
modelo2 |> tidy() |> select(term,std.error)
```

```
# A tibble: 8 x 2
  term        std.error
  <chr>           <dbl>
```

R^2 ajustados:

```
summary(modelo)$adj.r.squared
```

```
[1] 0.2566201
```

```
summary(modelo2)$adj.r.squared
```

```
[1] 0.2577438
```

AIC

```
modelo |> AIC()
```

```
[1] 3116.178
```

```
modelo2 |> AIC()
```

```
[1] 3114.386
```

Note que los procesos de selección de variables por pasos no retiraron la variable experiencia. Podemos volver a ejecutar dichos procesos retirando previamente esta variable (en algunas situaciones podrían coincidir y retirar las mismas variables):

```
lm(SALARIO ~ .-EXPERIENCIA, datos) -> modelo_nuevo
```

```
modelo_nuevo |> ols_step_backward_p(prem = 0.10)
modelo_nuevo |> ols_step_backward_aic()
modelo_nuevo |> ols_step_forward_p(prem = 0.10)
modelo_nuevo |> ols_step_forward_aic()
```