Lista de ejercicios 2 - Análisis de regresión

Ciclo nivelación 2025-2

Mg. Sc. J. Eduardo Gamboa U.

En un curso universitario de Estadística General, se busca analizar qué factores influyen en el tiempo que un estudiante tarda en resolver un examen.

Antes del examen, los estudiantes rindieron una prueba diagnóstica para medir conocimientos previos. Además, se registraron las horas de estudio realizadas la semana anterior al examen, y se aplicó un breve cuestionario de ansiedad ante evaluaciones.

Durante el examen final se registró el tiempo de resolución en minutos. Los datos se encuentran en el archivo **Lista2_datos.csv**

Las variables son:

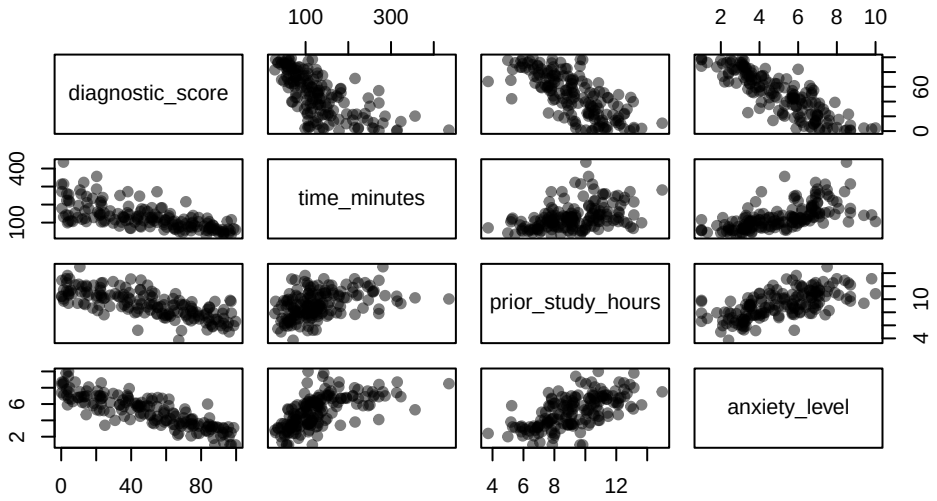
- time_minutes: tiempo de resolución del examen final (minutos)
- diagnostic_score: puntaje en la prueba diagnóstica
- prior_study_hours: tiempo de estudio (horas)
- anxiety_level: nivel de ansiedad (escala ordinal de 1 a 10)

1. Presentar y comentar un gráfico que permite ver las asociaciones entre las variables en estudio.

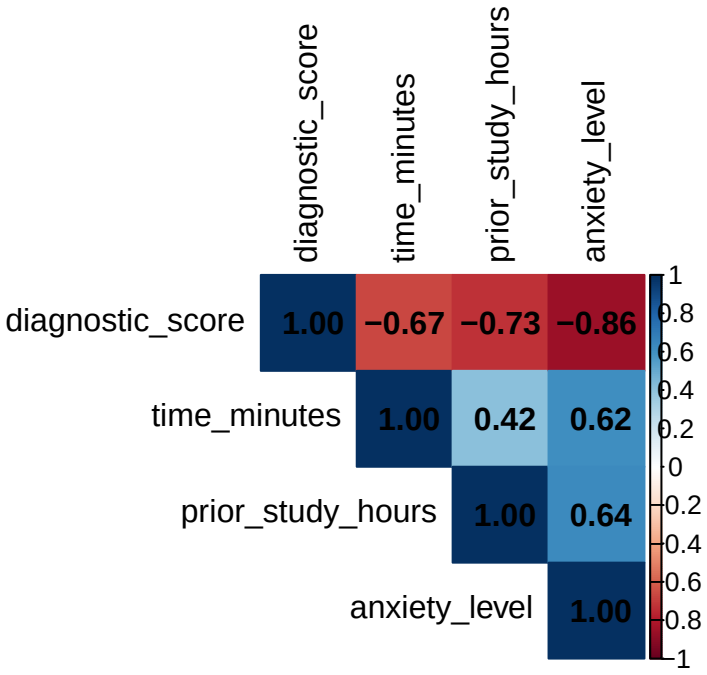
```
datos = read.csv('Lista2_datos.csv')
datos |> head()
```

```
  diagnostic_score time_minutes prior_study_hours anxiety_level
1             65.0       108.44              6.76           2.8
2             29.7        97.58             12.70           7.3
3             33.3        80.79             10.36           5.9
4             69.6        43.97              9.06           3.6
5             14.5       245.14             11.93           7.6
6             28.0       165.07             10.62           6.2
```

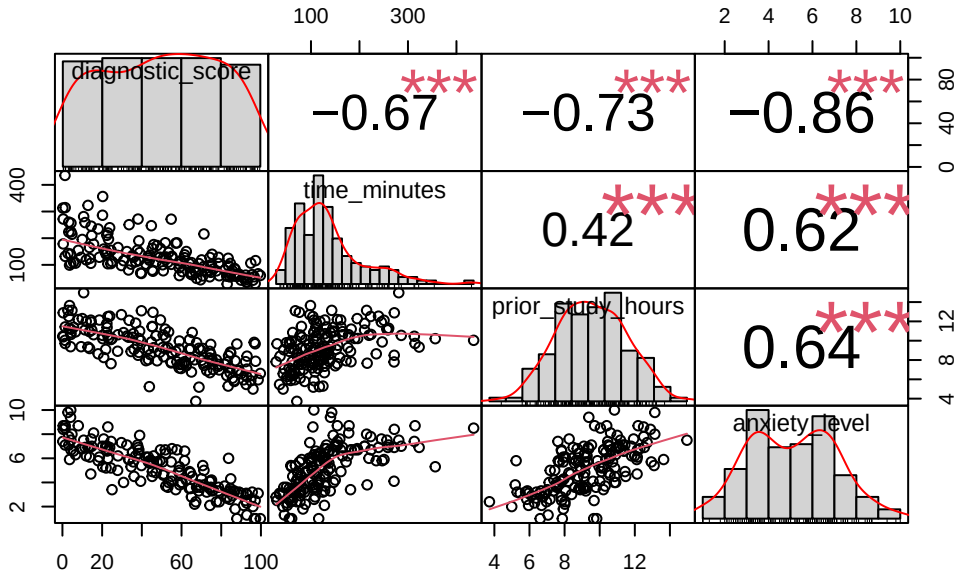
```r
pairs(datos, main = "Matriz de dispersión entre variables", pch = 19,
      col = rgb(0, 0, 0, 0.5))
```




```r
library(corrplot)
corrplot(cor(datos), method = "color",
         type = "upper", addCoef.col = "black", tl.col = "black")
```




```
library(PerformanceAnalytics)
chart.Correlation(datos)
```



La asociación entre el tiempo de resolución del examen y el puntaje de la prueba diagnóstica es directa o positiva, pero no está tan marcada. Por otro lado, el tiempo de resolución del examen se asocia de manera negativa o inversa y fuerte con el puntaje de la prueba diagnóstica. Finalmente, la asociación entre el nivel de ansiedad y el tiempo de resolución de la prueba, es positiva pero parece no ser lineal.

2. Formular el modelo de regresión lineal.

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 x_3 + \epsilon, \quad \epsilon \sim N(0, \sigma^2)$$

donde:

- Y: Tiempo de resolución del examen (variable respuesta)
- X_1: puntaje en la prueba diagnóstica (variable independiente)
- X_2: Horas de estudio (variable independiente)
- X_3: Nivel de ansiedad (variable independiente)
- β_0: Intercepto del modelo de regresión lineal
- $\beta_1, \beta_2, \beta_3$: coeficientes del modelo de regresión lineal
- ϵ: error aleatorio del modelo de regresión lineal

3. Presentar el modelo de regresión lineal estimado.

```
modelo1 <- lm(time_minutes ~ diagnostic_score + prior_study_hours + anxiety_level,
              data = datos)

coef(modelo1)
```

```
      (Intercept)  diagnostic_score prior_study_hours     anxiety_level
       220.661967         -1.521025         -5.196894          6.478634
```

$$\hat{Y} = 220.66 - 1.52X_1 - 5.2X_2 + 6.48X_3$$

4. Analizar el cumplimiento del supuesto de normalidad de errores.

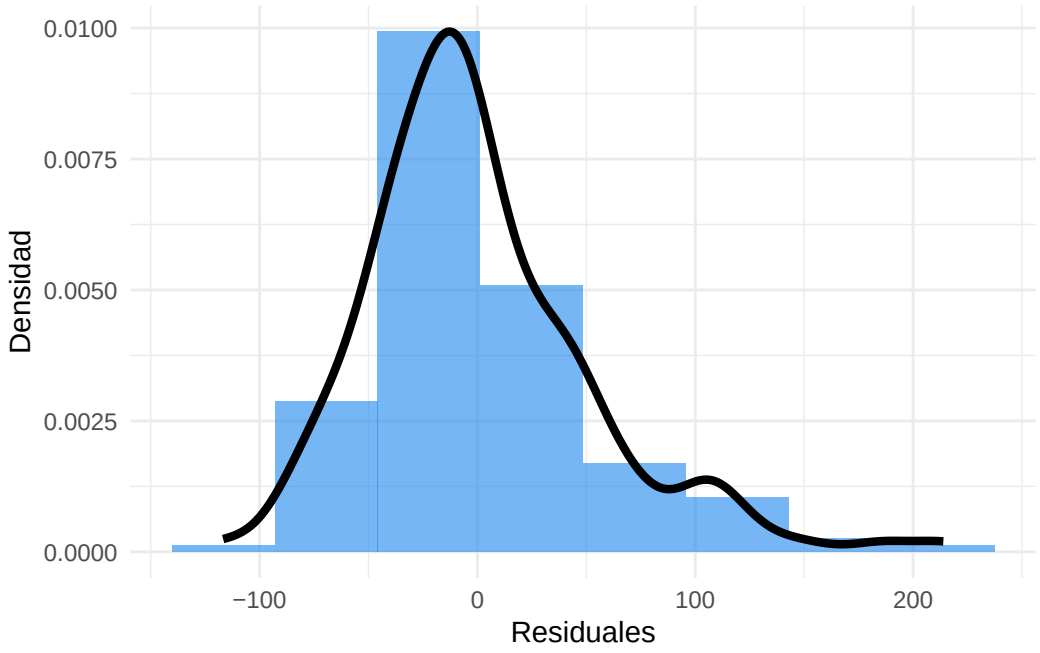
```
modelo1 |> residuals() -> res1

library(ggplot2)
```

```
Warning: package 'ggplot2' was built under R version 4.4.3
```

```r
data.frame(res1) |>
  ggplot(aes(x=res1,))+
  geom_histogram(aes(y =..density..),
                 bins = round(1+3.3*log10(nrow(datos))),
                 fill = "dodgerblue2",
                 alpha = 0.6)+
  geom_density(size = 1.5)+
  labs(x="Residuales",y="Densidad")+
  theme_minimal()
```

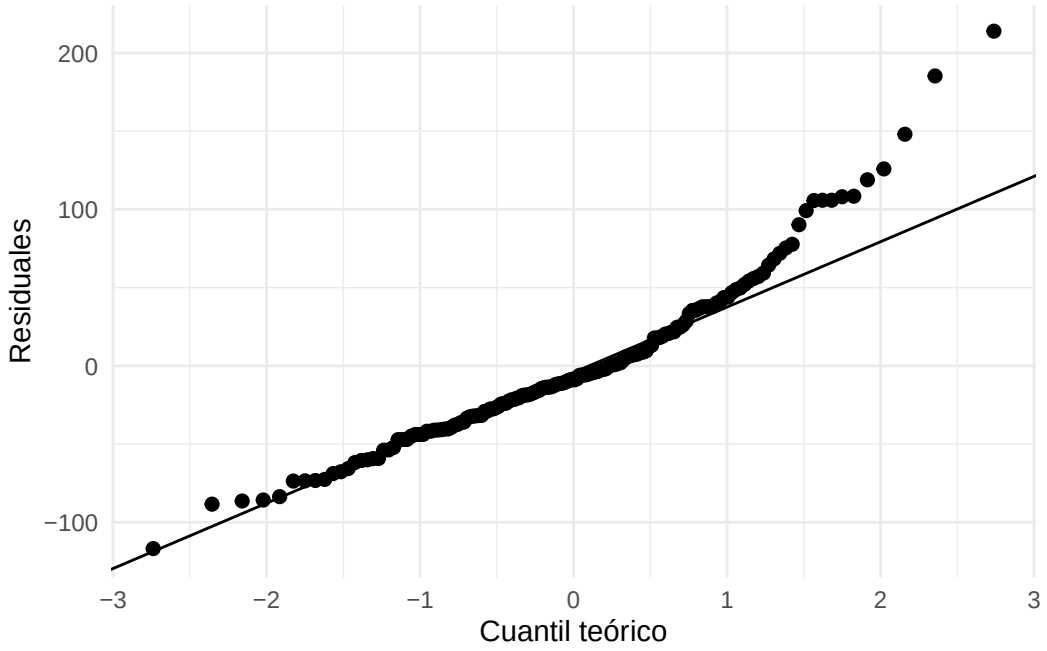
```
Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
i Please use `linewidth` instead.
```

```
Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2 3.4.0.
i Please use `after_stat(density)` instead.
```



Se aprecia asimetría en la distribución de los residuales, y una posible leptocurtosis, debido al apuntalamiento en el centro.

```r
data.frame(res1) |>
  ggplot(aes(sample=res1))+
  stat_qq(size = 2) +
  stat_qq_line(distribution = stats::qnorm)+
  labs(x = "Cuantil teórico", y = "Residuales")+
  theme_minimal()
```



Los puntos se alejan de la recta hacia el lado superior o derecho. Eso es señal de asimetría, lo que complicaría el cumplimiento del supuesto de normalidad de errores.

$H_0 : As = 0 \qquad H_1 : As \neq 0 \qquad \alpha = 0.05$

```r
library(moments)
res1 |> agostino.test()
```

```
    D'Agostino skewness test

data:  res1
skew = 1.0859, z = 4.9322, p-value = 8.129e-07
alternative hypothesis: data have a skewness
```

Se rechaza H_0, entonces existe evidencia de que el coeficiente de asimetría es distinto de cero (existe asimetría).

$H_0 : K = 3 \qquad H_1 : K \neq 3 \qquad \alpha = 0.05$

```
res1 |> anscombe.test()
```

```
    Anscombe-Glynn kurtosis test

data:  res1
kurt = 5.0661, z = 3.3685, p-value = 0.0007559
alternative hypothesis: kurtosis is not equal to 3
```

Se rechaza H_0, entonces existe evidencia de que el coeficiente de kurtosis es distinto de tres (los errores no son mesocúrticos).

Por lo tanto, al no ser simétricos ni mesocúrticos, seguimos acumulando evidencias que nos alejan de la normalidad.

H_0: los errores siguen una distribución normal

H_1: los errores no siguen una distribución normal

$\alpha = 0.05$

```
res1 |> shapiro.test()
```

```
    Shapiro-Wilk normality test

data:  res1
W = 0.93908, p-value = 2.029e-06
```

```
library(nortest)
res1 |> ad.test()
```

```
    Anderson-Darling normality test

data:  res1
A = 2.5582, p-value = 1.748e-06
```

```
res1 |> lillie.test()
```

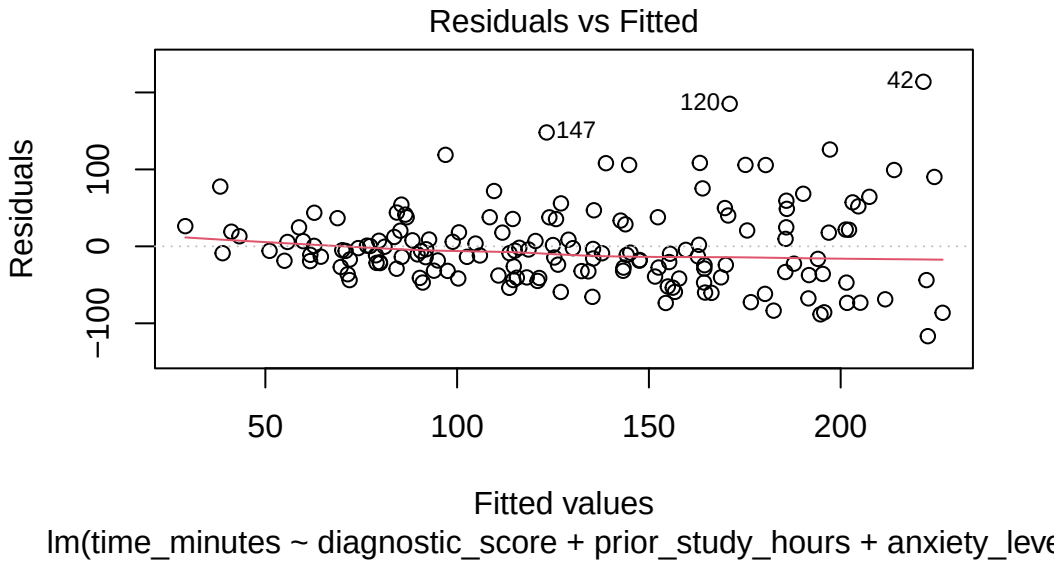
```
    Lilliefors (Kolmogorov-Smirnov) normality test

data:  res1
D = 0.11159, p-value = 4.048e-05
```

Las pruebas de normalidad dan evidencia de que los errores no siguen una distribución normal.

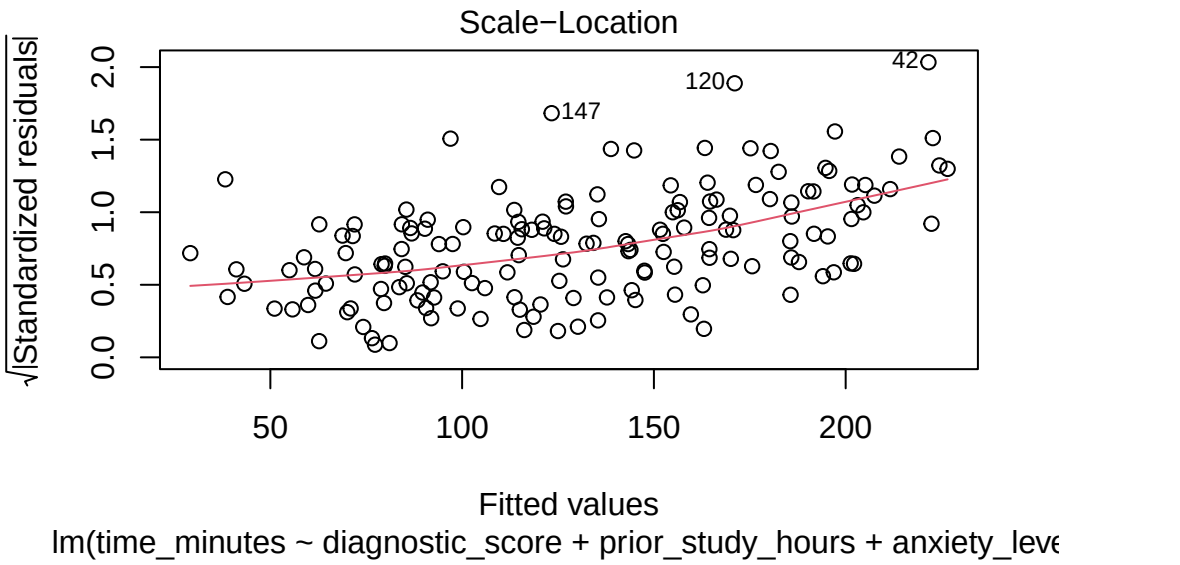
En conclusión, no se cumple el supuesto de normalidad de errores.

5. Analizar el cumplimiento del supuesto de homocedasticidad de errores.

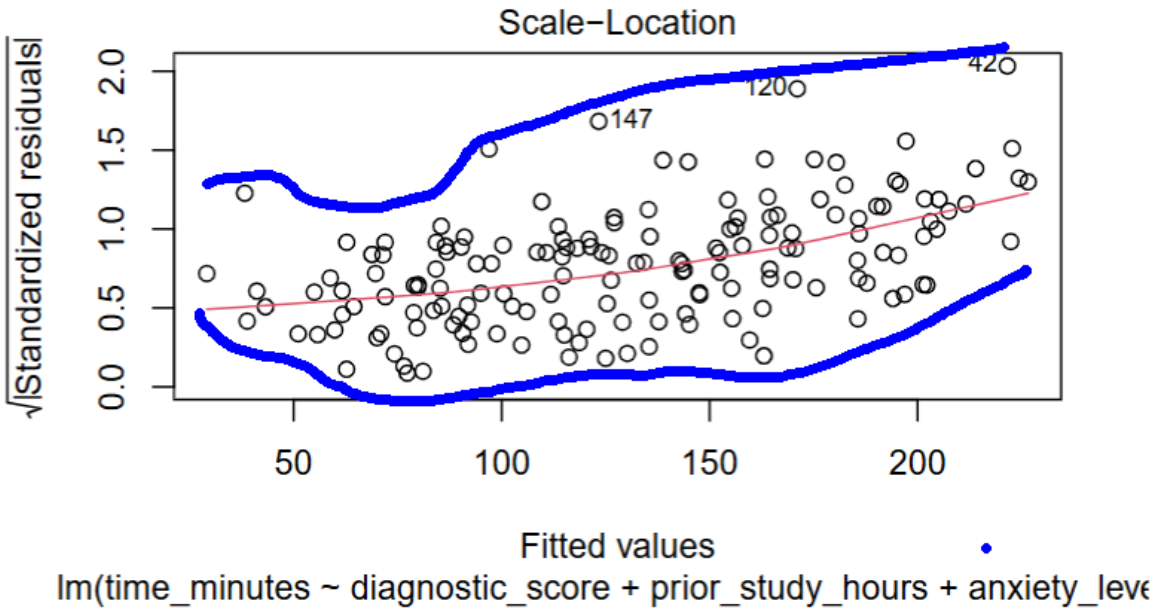
```
modelo1 |> plot(which=1)
```




```
modelo1 |> plot(which=3)
```



En la primera gráfica (residuals vs fitted), las líneas que deberian "envolver" los puntos divergen cuando los valores ajustados crecen, de modo que los puntos no están homogéneamente distribuidos. Esto es una señal de heterocedasticidad.



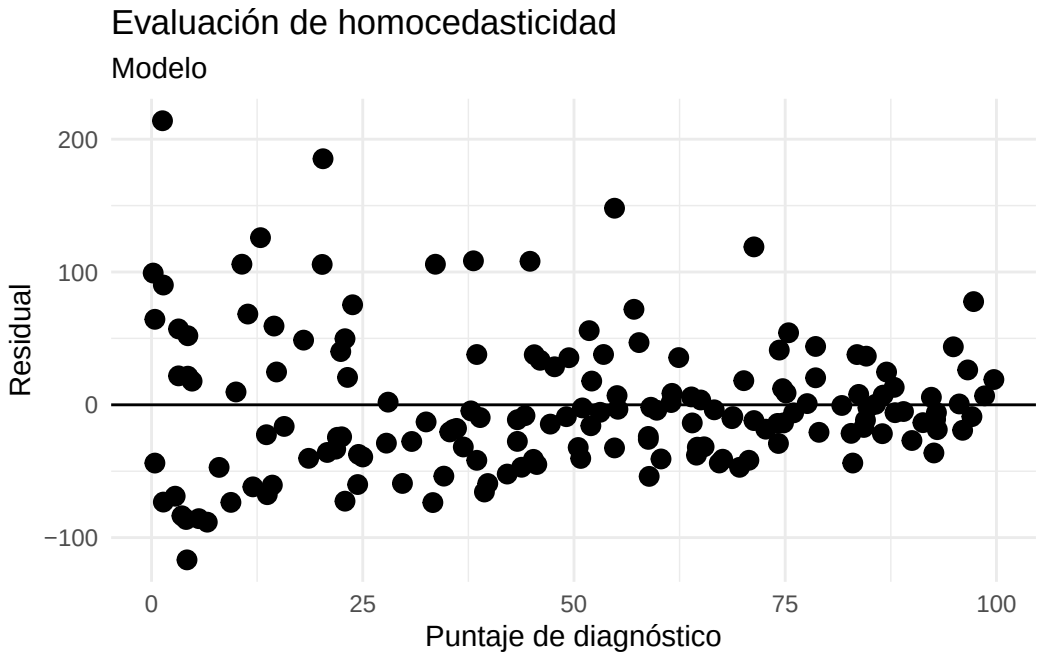
A diferencia de la primera gráfica, en Scale - Location no se aprecia una gran diferencia en la

variabilidad de los puntos.

```
library(broom)
```

```
Warning: package 'broom' was built under R version 4.4.3
```

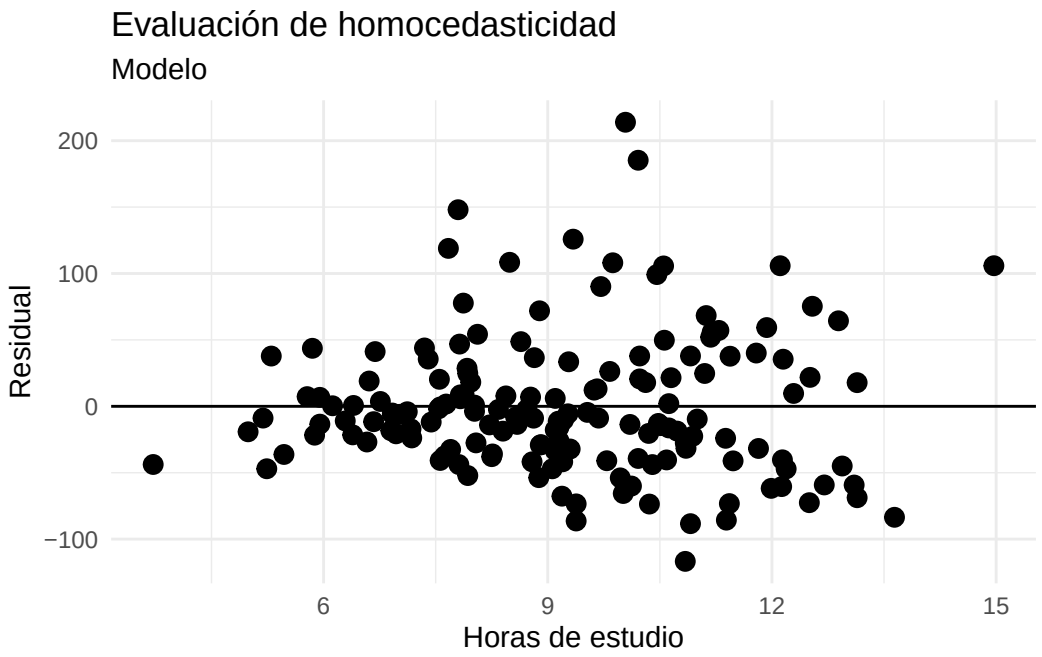
```
modelo1 |> augment() |>
  ggplot(aes(x=diagnostic_score,y=.resid))+
  geom_point(size = 3) +
  geom_hline(yintercept=0)+
  labs(x = "Puntaje de diagnóstico",
       y = "Residual",
       title = "Evaluación de homocedasticidad",
       subtitle = "Modelo")+
  theme_minimal()
```



Los residuales tienden a ser más dispersos cuando el puntaje de diagnóstico es bajo.

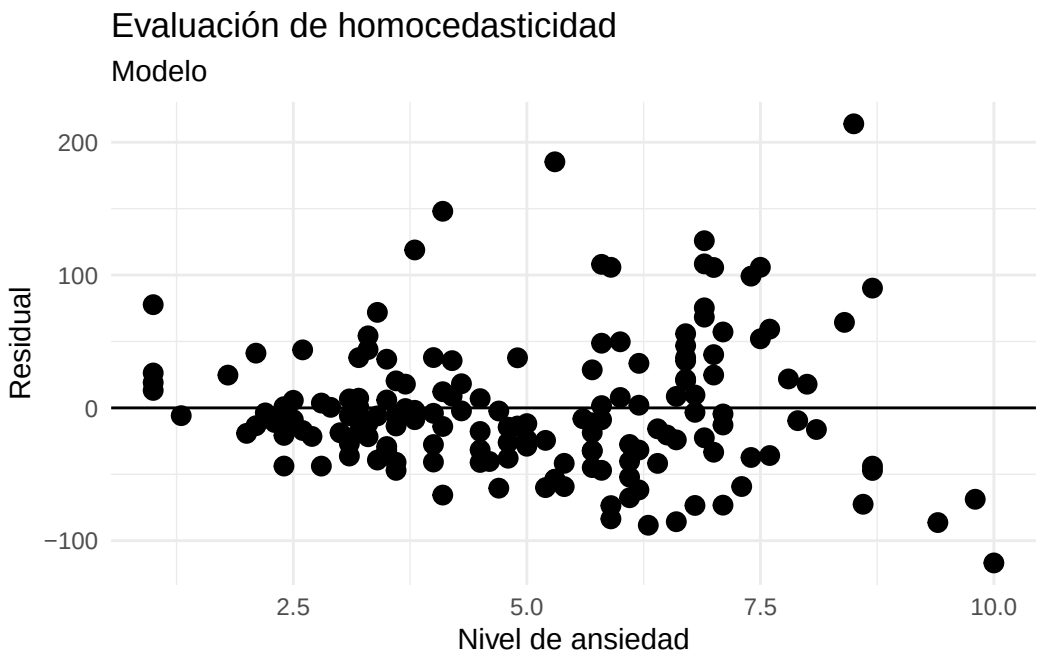
```
modelo1 |> augment() |>
  ggplot(aes(x=prior_study_hours,y=.resid))+
  geom_point(size = 3) +
  geom_hline(yintercept=0)+
```

```
  labs(x = "Horas de estudio",
       y = "Residual",
       title = "Evaluación de homocedasticidad",
       subtitle = "Modelo")+
  theme_minimal()
```



Los residuales presentan baja dispersión cuando las horas de estudio son inferiores.

```
modelo1 |> augment() |>
  ggplot(aes(x=anxiety_level,y=.resid))+
  geom_point(size = 3) +
  geom_hline(yintercept=0)+
  labs(x = "Nivel de ansiedad",
       y = "Residual",
       title = "Evaluación de homocedasticidad",
       subtitle = "Modelo")+
  theme_minimal()
```



Los residuales presentan mayor dispersión cuando los niveles de ansiedad son altos.

Entonces, al formarse patrones de residuales según los valores de la variable respuesta, es una evidencia más de que exista heterocedasticidad.

H_0: la varianza de los errores es constante

H_1: la varianza de los errores no es constante

$\alpha = 0.05$

```
library(car)
```

```
Cargando paquete requerido: carData
```

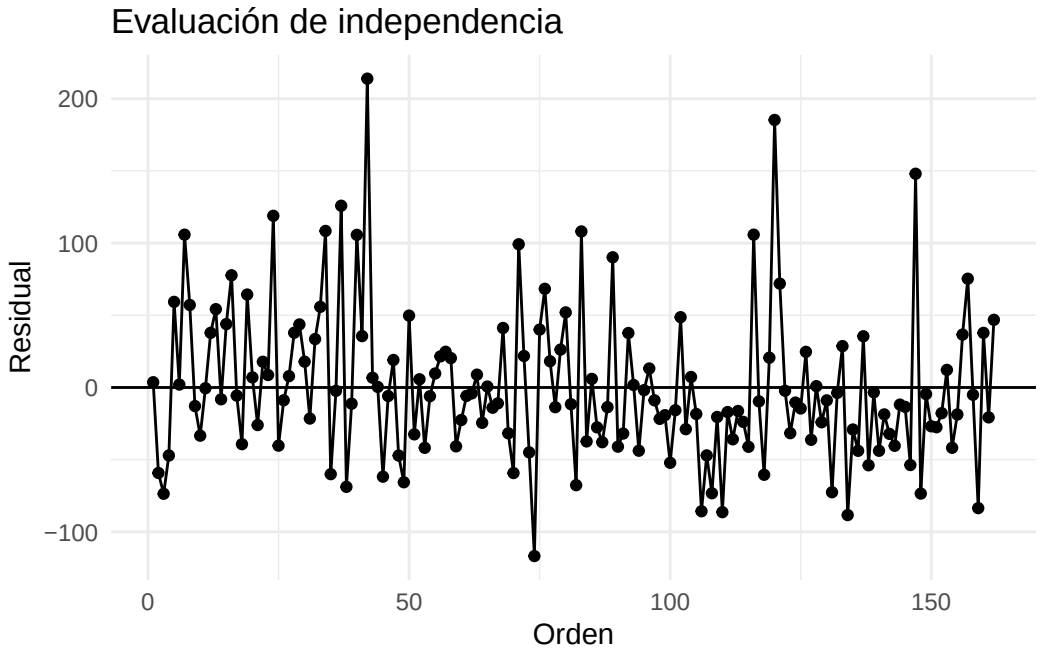
```
modelo1 |> ncvTest()
```

```
Non-constant Variance Score Test
Variance formula: ~ fitted.values
Chisquare = 43.15078, Df = 1, p = 5.068e-11
```

Se rechaza la hipótesis nula, por lo tanto no se verifica el supuesto de homogeneidad de varianzas de los errores.

6. Analizar el cumplimiento del supuesto de independencia de errores.

```
data.frame(res1) |>
  ggplot(aes(x=1:nrow(datos),y=res1))+
  geom_point(size = 1.5) +
  geom_line()+
  geom_hline(yintercept=0)+
  labs(x = "Orden", y = "Residual", title = "Evaluación de independencia") +
  theme_minimal()
```



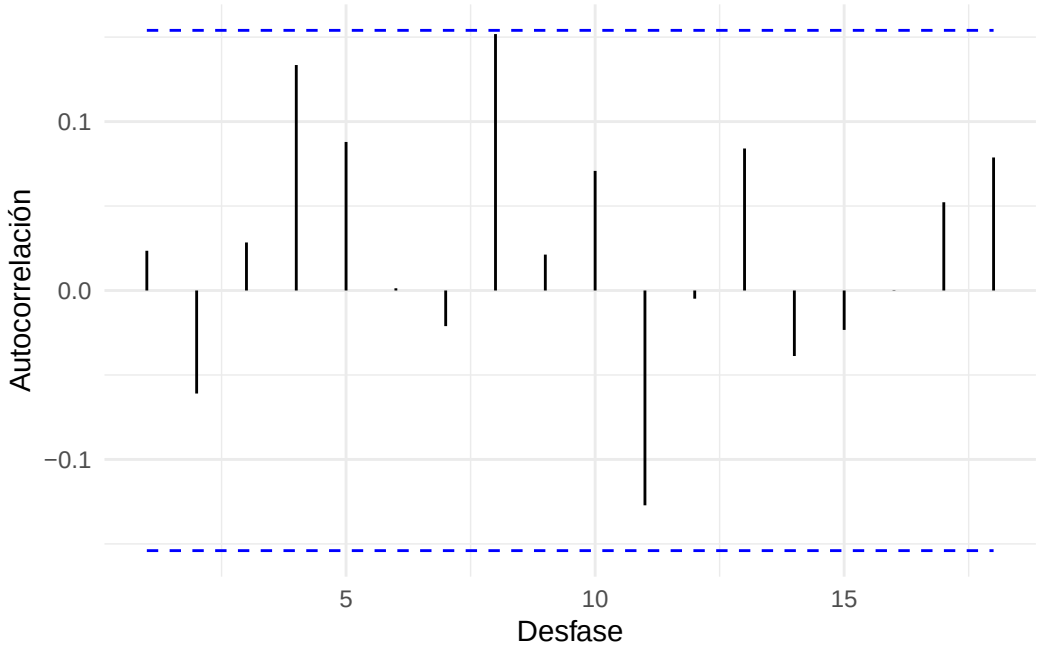
Visualizamos una primera evidencia de independencia de errores por la aleatoriedad de los puntos

```
library(ggfortify)
```

```
Warning: package 'ggfortify' was built under R version 4.4.3
```

```
res1 |>
  TSA::acf(lag = 18, plot=F) |>
  autoplot() +
  labs(x = "Desfase",y = "Autocorrelación") +
  theme_minimal()
```

```
Warning: `aes_string()` was deprecated in ggplot2 3.0.0.
i Please use tidy evaluation idioms with `aes()`.
i See also `vignette("ggplot2-in-packages")` for more information.
i The deprecated feature was likely used in the ggfortify package.
  Please report the issue at <https://github.com/sinhrks/ggfortify/issues>.
```



Todas las autocorrelaciones son estadísticamente iguales a cero (están dentro de las límites azules de confianza), lo que evidenciaría independencia de errores.

H_0: Los errores son independientes

H_1: Los errores no son independientes

$\alpha = 0.05$

```
library(car)
modelo1 |>
  durbinWatsonTest(alternative = "two.sided",
                   max.lag = 10,
                   reps = 1e5)
```

```
 lag Autocorrelation D-W Statistic p-value
   1      0.02352775      1.947898 0.72500
   2     -0.06098949      2.107944 0.45264
```

```
   3      0.02844076      1.913435 0.68288
   4      0.13346970      1.682359 0.07990
   5      0.08793277      1.765341 0.23526
   6      0.00130936      1.925615 0.92480
   7     -0.02107587      1.941718 0.91808
   8      0.15179522      1.587714 0.04270
   9      0.02125529      1.844428 0.72782
  10      0.07080258      1.742446 0.36150
 Alternative hypothesis: rho[lag] != 0
```

Se recomienda evaluar qué sucede a cada 8 filas, porque se ha encontrado una ligera autocorrelación en ese desfase. Por lo demás, no se evidencia falta de independencia.

7. En caso de incumplimiento de modelo, aplicar las siguientes transformaciones y comparar sus efectos: Raíz cuadrada, logaritmo, Box Cox, Mínimos cuadrados ponderados.

El principal problema detectado es la falta de normalidad y homocedasticidad de errores

a. Transformación raíz cuadrada

```
library(dplyr)
```

```
########################## 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.  #
#                                                                              #
################################################################################


Adjuntando el paquete: 'dplyr'
```

```
The following object is masked from 'package:car':

    recode

The following objects are masked from 'package:xts':

    first, last

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
```

```
datos |>
  mutate(raiz_tiempo = sqrt(time_minutes)) -> datos
modelo2 <- lm(raiz_tiempo ~ diagnostic_score + prior_study_hours + anxiety_level, datos)
```

b. Transformación logaritmo

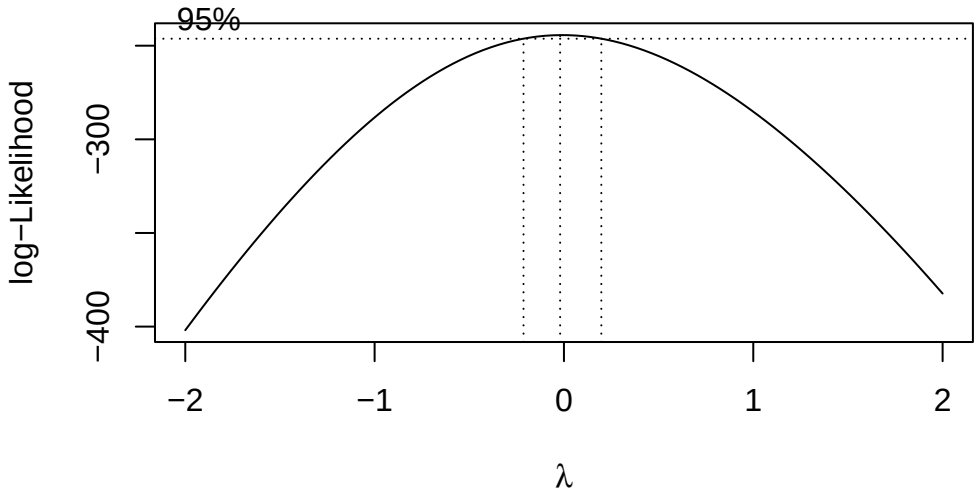
```
datos |>
  mutate(log_tiempo = log(time_minutes)) -> datos
modelo3 <- lm(log_tiempo ~ diagnostic_score + prior_study_hours + anxiety_level, datos)
```

c. Transformación Box Cox

```
library(MASS)
```

```
Warning: package 'MASS' was built under R version 4.4.3


Adjuntando el paquete: 'MASS'

The following object is masked from 'package:dplyr':

    select
```

```
modelo1 |> boxcox()
```



Box Cox sugiere usar $\lambda = 0$, lo que significa la transformación logaritmo, la cual ya se ejecutó en el caso b.

d. Mínimos cuadrados ponderados

```
pesos = datos$diagnostic_score
modelo4 <- lm(time_minutes ~ diagnostic_score + prior_study_hours + anxiety_level,
              datos,
              weights = pesos)
```

8. Presentar un cuadro comparativo y elegir el mejor modelo. Resolver las preguntas que siguen utilizando dicho mejor modelo.
9. Escribir la ecuación estimada del mejor modelo e interpretar sus coeficientes.
10. Ejecutar la prueba de hipótesis global del modelo, presentando el cuadro ANVA.
11. Ejecutar las pruebas de hipótesis individuales
12. Interpretar el coeficiente de determinación ajustado
13. Estimar el tiempo mediano del examen final si el puntaje diagnóstico fue de 65 puntos, estudió 10 horasy el nivel de ansiedad es 5.5, e indicar si coincide con el tiempo medio. Además, reportar el intervalo de confianza.

14. Predecir el tiempo que tardará en resolver el examen final un estudiante con puntaje diagnóstico de 50 puntos, que estudió 4.5 horas y con nivel de ansiedad 8. Compararlo con uno que tuvo 85 puntos diagnósticos, 12 horas de estudio y nivel de ansiedad 2.3