Regresión lineal - Análisis de los errores

Jacobo Hirsch Rodriguez

2024-09-04

Cargamos los datos

```
datos=read.csv("./estatura_peso.csv") #leer el dataset
```

```
str(datos)
```

```
## 'data.frame':    440 obs. of  3 variables:
##  $ Estatura: num  1.61 1.61 1.7 1.65 1.72 1.63 1.76 1.67 1.67 1.65 ...
##  $ Peso    : num  72.2 65.7 75.1 68.5 70.8 ...
##  $ Sexo    : chr  "H" "H" "H" "H" ...
```

como la variable sexo es un valor tipo char, es necesario hacer una variable dummy que represente dicha columna

```
# Convertir la variable 'sexo' a binaria: 1 para "H" (Hombre), 0 para "M" (Mujer)
datos$sexo_binario <- ifelse(datos$Sexo == "H", 1, 0)
```

y vamos a hacer un nuevo dataset que tenga solo las variables numericas

#Modelos

#1) Sin interacción

```
# Realizar la regresión lineal considerando 'estatura' y 'sexo_binario'
modelo_sin_interaccion <- lm(Peso ~ Estatura + sexo_binario, data = datos)

# Resumen del modelo para ver la ecuación y los coeficientes
summary(modelo_sin_interaccion)
```

```
##
## Call:
## lm(formula = Peso ~ Estatura + sexo_binario, data = datos)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -21.9505  -3.2491   0.0489   3.2880  17.1243
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)  -85.3191     7.1874  -11.87   <2e-16 ***
## Estatura      89.2604     4.5635   19.56   <2e-16 ***
## sexo_binario  10.5645     0.6317   16.72   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.381 on 437 degrees of freedom
## Multiple R-squared:  0.7837, Adjusted R-squared:  0.7827
## F-statistic: 791.5 on 2 and 437 DF,  p-value: < 2.2e-16
```

#2) Con interacción

```r
# Realizar la regresión lineal considerando 'estatura' y 'sexo_binario'
modelo_con_interaccion <- lm(Peso ~ Estatura * sexo_binario, data = datos)

# Resumen del modelo para ver la ecuación y los coeficientes
summary(modelo_con_interaccion)
```

```
##
## Call:
## lm(formula = Peso ~ Estatura * sexo_binario, data = datos)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -21.3256  -3.1107   0.0204   3.2691  17.9114
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)             -72.560     11.346  -6.395 4.13e-10 ***
## Estatura                 81.149      7.209  11.256  < 2e-16 ***
## sexo_binario            -11.124     14.950  -0.744    0.457
## Estatura:sexo_binario    13.511      9.305   1.452    0.147
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.374 on 436 degrees of freedom
## Multiple R-squared:  0.7847, Adjusted R-squared:  0.7832
## F-statistic: 529.7 on 3 and 436 DF,  p-value: < 2.2e-16
```

#El Validez del modelo

Analiza si el (los) modelo(s) obtenidos anteriormente son apropiados para el conjunto de datos. Realiza el análisis de los residuos:

obtenemos los residuos para el modelo 1 y para el modelo 2

##Residuos del modelo 1

```r
residuos_con_interaccion <- residuals(modelo_con_interaccion)
```

##Residuos del modelo 2

```r
residuos_sin_interaccion <- residuals(modelo_sin_interaccion)
```

##Normalidad de los residuos

hipotesis nula : los residuos siguen una distribución normal.

hipotesis alternativa : los residuos no siguen una distribución normal

regla de decisión : Si el valor p es menor a 0.05, se rechaza la hipótesis nula y se concluye que los residuos no siguen una distribución normal.

con interacción

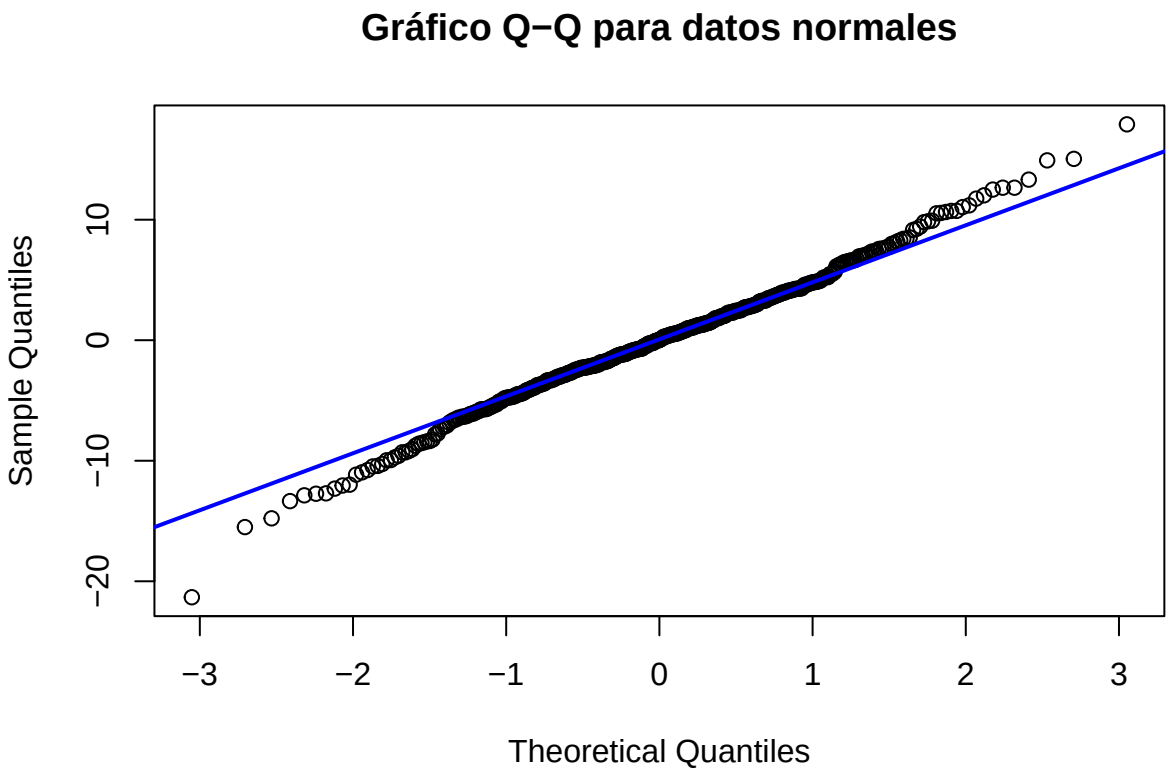
sacamos la prueba de shapiro-wilk

```r
# Prueba de Shapiro-Wilk
shapiro.test(residuos_con_interaccion)
```

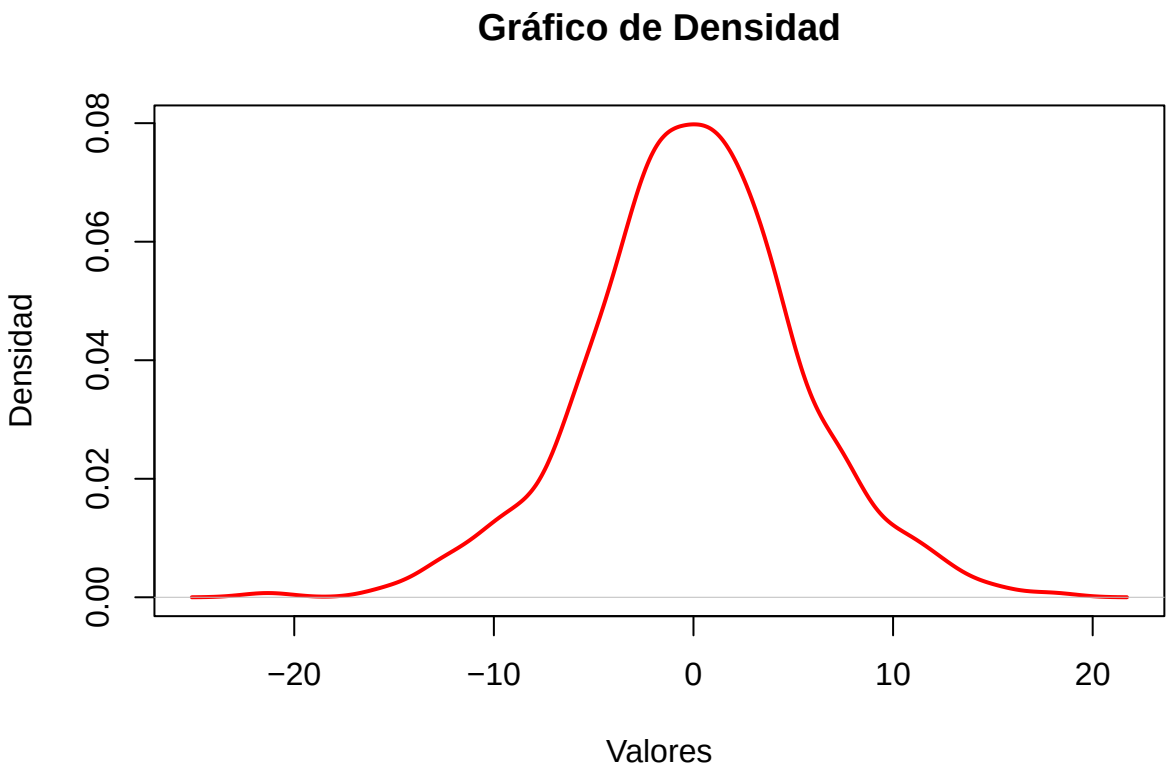
```
##
##  Shapiro-Wilk normality test
##
## data:  residuos_con_interaccion
## W = 0.99356, p-value = 0.05772
```

vamos a graficar los residuos sin interaccion para visualizar su distribucion normal

```r
#Gráfico Q-Q (Quantile-Quantile plot)
qqnorm(residuos_con_interaccion,
       main = "Gráfico Q-Q para datos normales")
qqline(residuos_con_interaccion, col = "blue", lwd = 2)
```




```r
#Gráfico de densidad
plot(density(residuos_con_interaccion),
     col = "red",
     lwd = 2,
     main = "Gráfico de Densidad",
     xlab = "Valores",
     ylab = "Densidad")
```



sin interacción

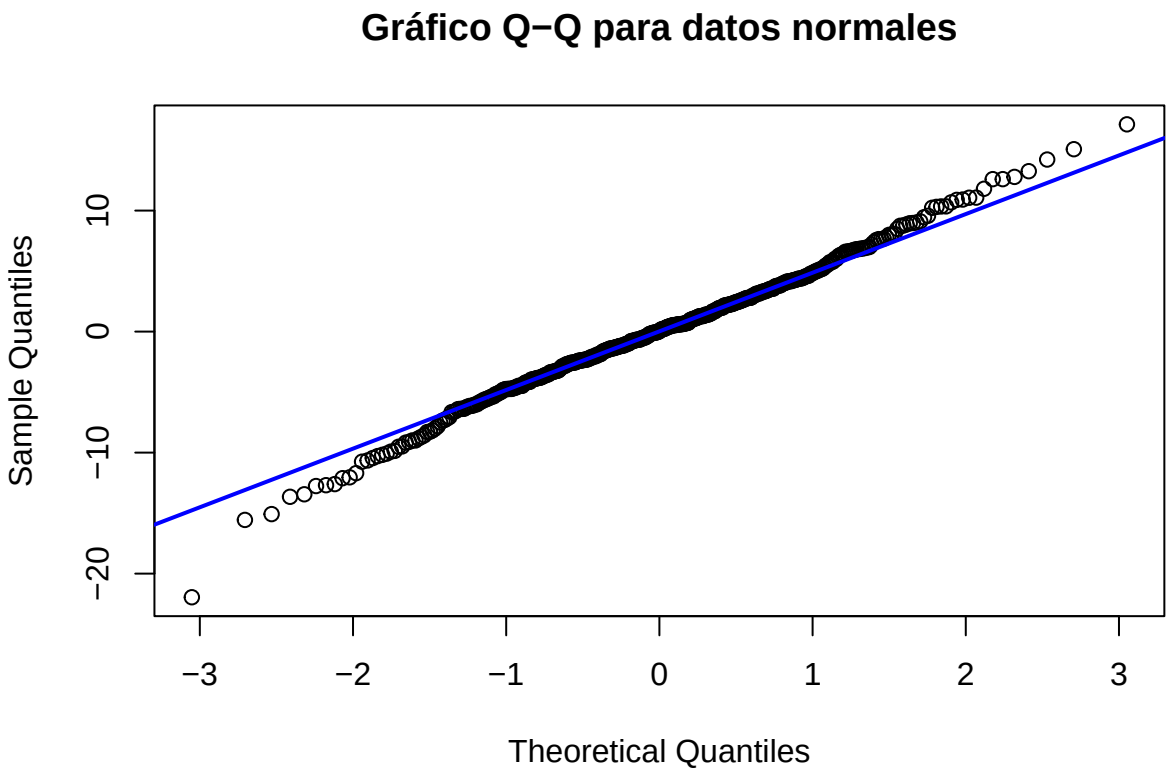
sacamos la prueba de shapiro-wilk

```
# Prueba de Shapiro-Wilk
shapiro.test(residuos_sin_interaccion)
```

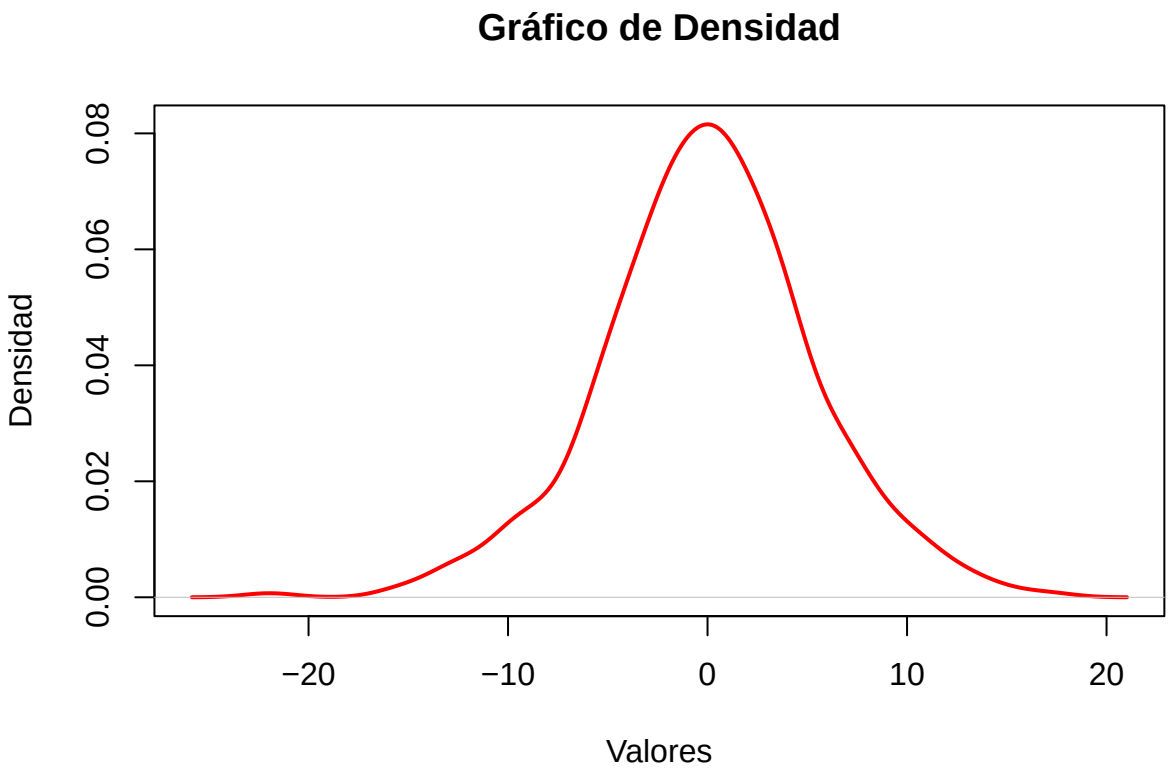
```
##
##  Shapiro-Wilk normality test
##
## data:  residuos_sin_interaccion
## W = 0.99337, p-value = 0.0501
```

vamos a graficar los residuos sin interaccion para visualizar su distribucion normal

```
#Gráfico Q-Q (Quantile-Quantile plot)
qqnorm(residuos_sin_interaccion,
       main = "Gráfico Q-Q para datos normales")
qqline(residuos_sin_interaccion, col = "blue", lwd = 2)
```




```r
#Gráfico de densidad
plot(density(residuos_sin_interaccion),
     col = "red",
     lwd = 2,
     main = "Gráfico de Densidad",
     xlab = "Valores",
     ylab = "Densidad")
```



conclusión para la normalidad de los residuos: los resultados del valor p para ambas pruebas son superiores a 0.05 por lo que no se puede rechazar la hipotesis nula para ninguno de los modelos. Además los datos parece que se distribuyen normalmente en los graficos.

##Verificación de media cero

hipotesis nula : la media de los errores es igual a 0

hiporesis alrernativa : la media de los errores es diferente de 0

regla de decisión con prueba t : Si el valor p es menor que 0.05, se rechaza la hipótesis nula y se concluye que la media de los residuos es significativamente diferente de 0.

con interacción

podemos obtener la media de los errores (residuos)

```
media_con_interaccion = mean(residuos_con_interaccion)
print(media_con_interaccion)
```

```
## [1] 2.422305e-16
```

realizamos la prueba t para el modelo con interacción

```
# Realizar la prueba t para verificar si la media es 0
prueba_t <- t.test(residuos_con_interaccion, mu = 0)
```

```r
# Mostrar el resultado de la prueba
print(prueba_t)
```

```
##
##  One Sample t-test
##
## data:  residuos_con_interaccion
## t = 9.4878e-16, df = 439, p-value = 1
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -0.5017741  0.5017741
## sample estimates:
##    mean of x
## 2.422305e-16
```

sin interacción

odemos obtener la media de los errores (residuos)

```r
media_sin_interaccion = mean(residuos_sin_interaccion)
print(media_sin_interaccion)
```

```
## [1] -1.61487e-17
```

realizamos la prueba t para el modelo sin interaccion

```r
# Realizar la prueba t para verificar si la media es 0
prueba_t <- t.test(residuos_sin_interaccion, mu = 0)

# Mostrar el resultado de la prueba
print(prueba_t)
```

```
##
##  One Sample t-test
##
## data:  residuos_sin_interaccion
## t = -6.31e-17, df = 439, p-value = 1
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -0.5029859  0.5029859
## sample estimates:
##    mean of x
## -1.61487e-17
```

conclusión para la verificación de la media 0:

para ambos casos el valor p fue de 1, el valor más alto posible que indica que no hay evidencia para descartar la hipotesis nula

##Homocedasticidad

hipotesis nula: la varianza de los errores es consrante (hay homocedasticidad)

hipotesis alternativa : la varianza de los errores no es constante (hay heterocedasticidad)

regla de decisión: si el valor p es menor o igual que 0.05 entonces rechazamos la hipotesis nula

para realizar las pruebas vamos a necesitar descargar un paquete que contiene la prueba que vamos a utilizar

```
# Cargar el paquete lmtest
library(lmtest)
```

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

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

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

con interacción

vamos a realizar la prueba de white

```
# Realizar la prueba de White
prueba_white_con_interaccion <- bptest(modelo_con_interaccion, ~ fitted(modelo_con_interaccion) + I(fit

# Mostrar los resultados de la prueba
print(prueba_white_con_interaccion)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  modelo_con_interaccion
## BP = 27.183, df = 2, p-value = 1.251e-06
```

vamos a verificar la hipotesis nula

```
# Extraer el valor p de la prueba
p_value_white_con_interaccion <- prueba_white_con_interaccion$p.value

# Comparar el valor p con el umbral de significancia de 0.05
if (p_value_white_con_interaccion > 0.05) {
  print("No se rechaza la hipótesis nula: no hay evidencia de heterocedasticidad.")
} else {
  print("Se rechaza la hipótesis nula: hay evidencia de heterocedasticidad.")
}
```

```
## [1] "Se rechaza la hipótesis nula: hay evidencia de heterocedasticidad."
```

sin interacción

vamos a realizar la prueba de white

```r
# Realizar la prueba de White
prueba_white_sin_interaccion <- bptest(modelo_sin_interaccion, ~ fitted(modelo_sin_interaccion) + I(fit

# Mostrar los resultados de la prueba
print(prueba_white_sin_interaccion)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  modelo_sin_interaccion
## BP = 27.271, df = 2, p-value = 1.197e-06
```

vamos a verificar la hipotesis nula

```r
# Extraer el valor p de la prueba
p_value_white_sin_interaccion <- prueba_white_sin_interaccion$p.value

# Comparar el valor p con el umbral de significancia de 0.05
if (p_value_white_sin_interaccion > 0.05) {
  print("No se rechaza la hipótesis nula: no hay evidencia de heterocedasticidad.")
} else {
  print("Se rechaza la hipótesis nula: hay evidencia de heterocedasticidad.")
}
```

```
## [1] "Se rechaza la hipótesis nula: hay evidencia de heterocedasticidad."
```

conclusiones para deterinar la homocedasticidad del modelo:

en ambos modelos se puede observar que hay heterocedasticidad por lo que se rechaza la hipotesis nula

##independencia

hipotesis nula: los errores no estan correlacionados

hipotesis alternativa : los errores estan correlacionados

regla de decisión: si el valor p de la prueba es menor a 0.05 se rechaza la hipotesis nula

con interacción

hacemos el modelo durbin watson para verificar independencia entre los residuos

```r
# Realizar la prueba de Durbin-Watson
prueba_dw_con_interaccion <- dwtest(modelo_con_interaccion)

# Mostrar los resultados de la prueba
print(prueba_dw_con_interaccion)
```

```
##
##  Durbin-Watson test
##
## data:  modelo_con_interaccion
## DW = 1.8646, p-value = 0.07113
## alternative hypothesis: true autocorrelation is greater than 0
```

verificamos que el valor p no sea menor a 0

```
# Extraer el valor p
p_value_dw_con_interaccion <- prueba_dw_con_interaccion$p.value

# Interpretar los resultados
if (p_value_dw_con_interaccion > 0.05) {
  print("No hay evidencia de correlación en los residuos (hipótesis nula no rechazada).")
} else {
  print("Hay evidencia de correlación en los residuos (hipótesis nula rechazada).")
}
```

```
## [1] "No hay evidencia de correlación en los residuos (hipótesis nula no rechazada)."
```

sin interacción

hacemos el modelo durbin watson para verificar independencia entre los residuos

```
# Realizar la prueba de Durbin-Watson
prueba_dw_sin_interaccion <- dwtest(modelo_sin_interaccion)

# Mostrar los resultados de la prueba
print(prueba_dw_sin_interaccion)
```

```
##
##  Durbin-Watson test
##
## data:  modelo_sin_interaccion
## DW = 1.8663, p-value = 0.07325
## alternative hypothesis: true autocorrelation is greater than 0
```

verificamos que el valor p no sea menor a 0

```
# Extraer el valor p
p_value_dw_sin_interaccion <- prueba_dw_sin_interaccion$p.value

# Interpretar los resultados

if (p_value_dw_sin_interaccion > 0.05) {
  print("No hay evidencia de correlación en los residuos (hipótesis nula no rechazada).")
} else {
  print("Hay evidencia de correlación en los residuos (hipótesis nula rechazada).")
}
```

```
## [1] "No hay evidencia de correlación en los residuos (hipótesis nula no rechazada)."
```

conclusión para la independencia de los residuos:

la hipotesis nula no fue rechazada para ninguno de los residuos por lo que no se muestra en ninguno de los modelos que los errores estén correlacionados.

##linealidad

para verificar la linealidad vamos a utilizar una prueba RESET

La prueba RESET de Ramsey (Regression Equation Specification Error Test) es utilizada para detectar posibles errores de especificación en un modelo de regresión lineal. La prueba examina si hay variables omitidas o si la forma funcional del modelo es incorrecta.

hipotesis nula: no hay términos omitidos que indican linealidad

hipotesis alternativa: hay una especificación errónea en el modelo que indica no linealidad

con interacción

vamos a hacer la prueba reset

```
# Realizar la prueba RESET de Ramsey
prueba_reset_con_interaccion <- resettest(modelo_con_interaccion)

# Mostrar los resultados de la prueba
print(prueba_reset_con_interaccion)
```

```
##
##  RESET test
##
## data:  modelo_con_interaccion
## RESET = 2.9635, df1 = 2, df2 = 434, p-value = 0.05268
```

extraemos el valor p para evalualr la regla de decisión

```
# Extraer el valor p
p_value_reset_con_interaccion <- prueba_reset_con_interaccion$p.value

# Interpretar los resultados
if (p_value_reset_con_interaccion > 0.05) {
  print("No se rechaza la hipótesis nula: el modelo no parece tener errores de especificación.")
} else {
  print("Se rechaza la hipótesis nula: es probable que haya errores de especificación en el modelo.")
}
```

```
## [1] "No se rechaza la hipótesis nula: el modelo no parece tener errores de especificación."
```

sin interacción

vamos a hacer la prueba reset

```
# Realizar la prueba RESET de Ramsey
prueba_reset_sin_interaccion <- resettest(modelo_sin_interaccion)

# Mostrar los resultados de la prueba
print(prueba_reset_sin_interaccion)
```

```
##
##  RESET test
```

```
##
## data:  modelo_sin_interaccion
## RESET = 3.1306, df1 = 2, df2 = 435, p-value = 0.04468
```

extraemos el valor p para evalualr la regla de decisión

```r
# Extraer el valor p
p_value_reset_sin_interaccion <- prueba_reset_sin_interaccion$p.value

# Interpretar los resultados
if (p_value_reset_sin_interaccion > 0.05) {
  print("No se rechaza la hipótesis nula: el modelo no parece tener errores de especificación.")
} else {
  print("Se rechaza la hipótesis nula: es probable que haya errores de especificación en el modelo.")
}
```

```
## [1] "Se rechaza la hipótesis nula: es probable que haya errores de especificación en el modelo."
```

conclusiones para la linealidad del modelo:

parece ser que para el modelo que no tiene interacción la prueba arroja que hace falta especificar variables en el modelo, y en la prueba con interaccion no por lo que parece ser que ese factor faltante puede ser la interaccion entre la estatura y el sexo

Utiliza el comando: plot(modelo). Observa las gráficas obtenidas y contesta: ¿Cuáles son las diferencias y similitudes de estos gráficos con respecto a los que ya habías analizado? Estos gráficos, ¿cambian en algo las conclusiones que ya habías obtenido? Emite una conclusión final sobre el mejor modelo de regresión lineal que conjunte lo que hiciste en las tres partes de esta actividad.

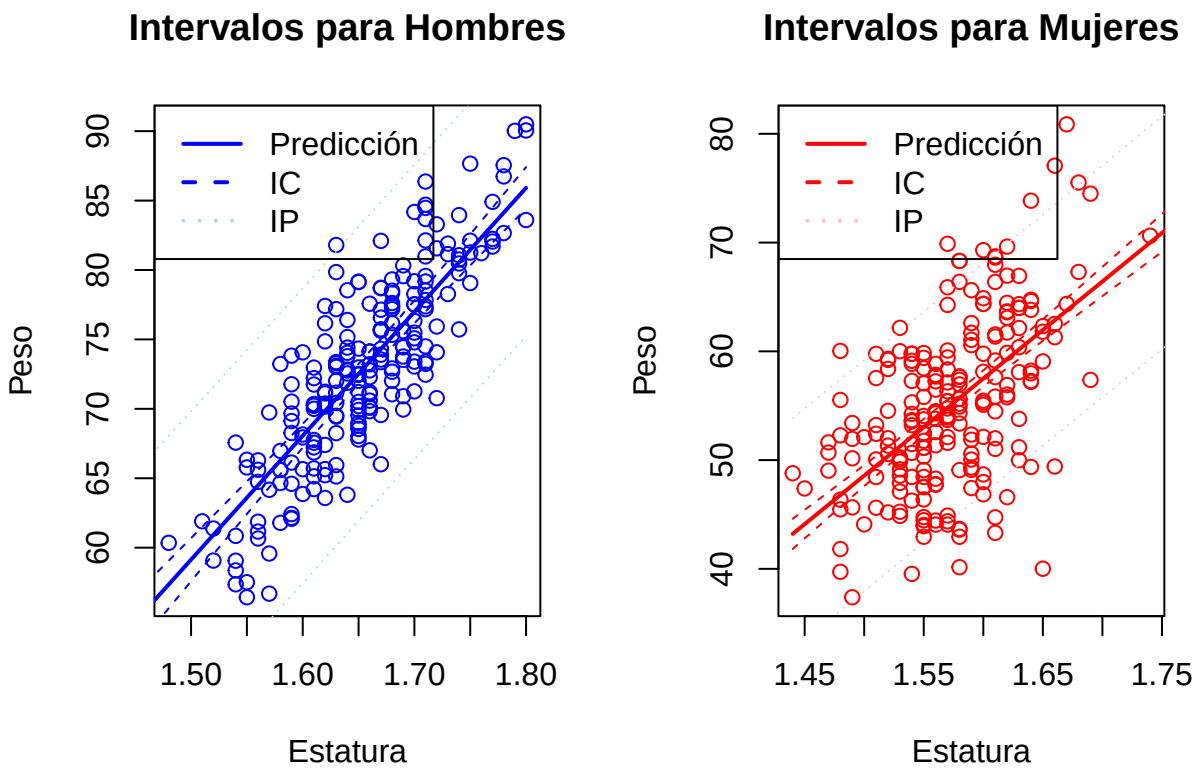
#Intervalos de confianza Con los datos de las estaturas y pesos de los hombres y las mujeres construye la gráfica de los intervalos de confianza y predicción para la estimación y predicción de Y para el mejor modelo seleccionado. Interpreta y comenta los resultados obtenidos

##Graficas para el modelo sin interacción

```r
# Crear un nuevo conjunto de datos con las combinaciones de Estatura y sexo_binario
# Aquí usamos el rango de estaturas en los datos y mantenemos sexo_binario fijo
estaturas_seq <- seq(min(datos$Estatura), max(datos$Estatura), length.out = 100)

# Crear dos conjuntos de datos: uno para hombres y otro para mujeres
nuevos_datos_hombres <- data.frame(Estatura = estaturas_seq, sexo_binario = 1)
nuevos_datos_mujeres <- data.frame(Estatura = estaturas_seq, sexo_binario = 0)
```

```r
# Obtener las predicciones y los intervalos para hombres
predicciones_hombres <- predict(modelo_sin_interaccion, nuevos_datos_hombres,
                                interval = "confidence", level = 0.95)

prediccion_hombres <- predict(modelo_sin_interaccion, nuevos_datos_hombres,
                              interval = "prediction", level = 0.95)

# Obtener las predicciones y los intervalos para mujeres
predicciones_mujeres <- predict(modelo_sin_interaccion, nuevos_datos_mujeres,
                                interval = "confidence", level = 0.95)

prediccion_mujeres <- predict(modelo_sin_interaccion, nuevos_datos_mujeres,
```

```r
                              interval = "prediction", level = 0.95)

# Configurar la ventana gráfica para dos gráficos en la misma fila
par(mfrow = c(1, 2))  # 1 fila, 2 columnas

# Gráfico para hombres
plot(datos$Estatura[datos$sexo_binario == 1], datos$Peso[datos$sexo_binario == 1],
     col = "blue", xlab = "Estatura", ylab = "Peso",
     main = "Intervalos para Hombres")

lines(estaturas_seq, predicciones_hombres[, "fit"], col = "blue", lwd = 2)
lines(estaturas_seq, predicciones_hombres[, "lwr"], col = "blue", lty = 2)
lines(estaturas_seq, predicciones_hombres[, "upr"], col = "blue", lty = 2)

lines(estaturas_seq, prediccion_hombres[, "lwr"], col = "lightblue", lty = 3)
lines(estaturas_seq, prediccion_hombres[, "upr"], col = "lightblue", lty = 3)

legend("topleft", legend = c("Predicción", "IC", "IP"),
       col = c("blue", "blue", "lightblue"),
       lty = c(1, 2, 3), lwd = 2)

# Gráfico para mujeres
plot(datos$Estatura[datos$sexo_binario == 0], datos$Peso[datos$sexo_binario == 0],
     col = "red", xlab = "Estatura", ylab = "Peso",
     main = "Intervalos para Mujeres")

lines(estaturas_seq, predicciones_mujeres[, "fit"], col = "red", lwd = 2)
lines(estaturas_seq, predicciones_mujeres[, "lwr"], col = "red", lty = 2)
lines(estaturas_seq, predicciones_mujeres[, "upr"], col = "red", lty = 2)

lines(estaturas_seq, prediccion_mujeres[, "lwr"], col = "pink", lty = 3)
lines(estaturas_seq, prediccion_mujeres[, "upr"], col = "pink", lty = 3)

legend("topleft", legend = c("Predicción", "IC", "IP"),
       col = c("red", "red", "pink"),
       lty = c(1, 2, 3), lwd = 2)
```

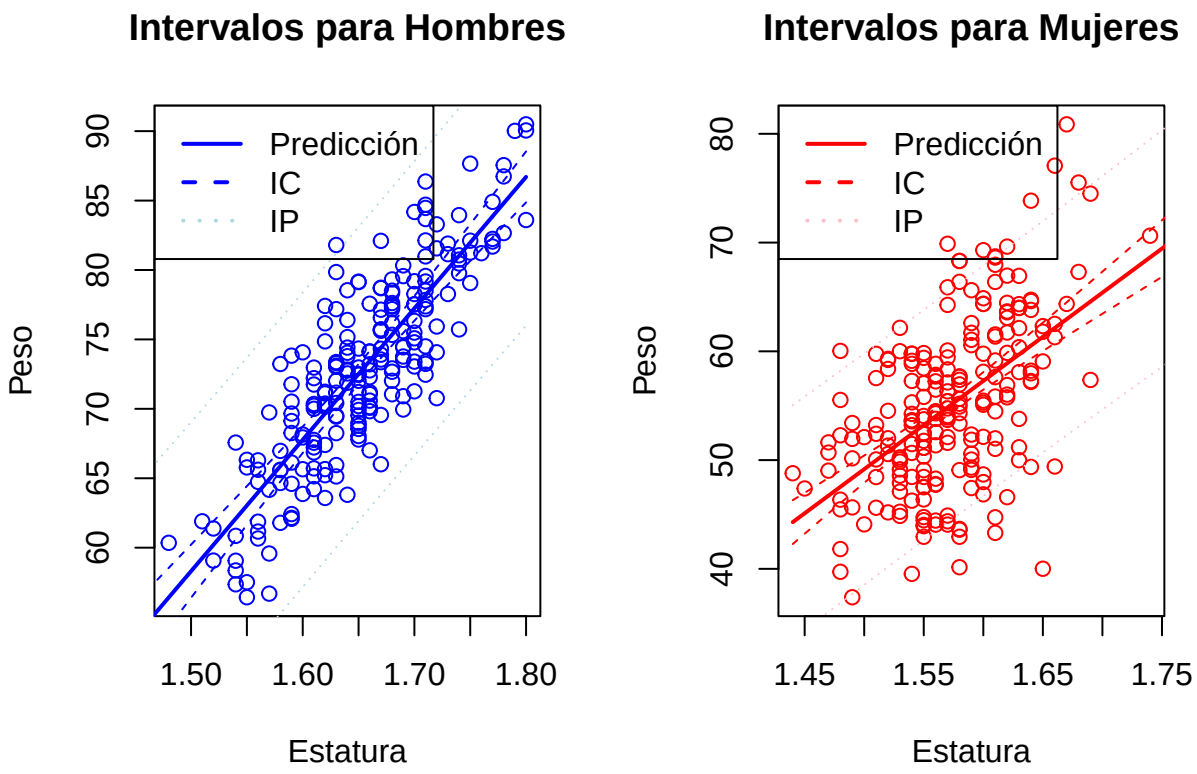



```
# Resetear la ventana gráfica para futuros gráficos
par(mfrow = c(1, 1))
```

##Graficas para el modelo sin interacción

```
# Obtener las predicciones y los intervalos para hombres
predicciones_hombres <- predict(modelo_con_interaccion, nuevos_datos_hombres,
                                interval = "confidence", level = 0.95)

prediccion_hombres <- predict(modelo_con_interaccion, nuevos_datos_hombres,
                              interval = "prediction", level = 0.95)

# Obtener las predicciones y los intervalos para mujeres
predicciones_mujeres <- predict(modelo_con_interaccion, nuevos_datos_mujeres,
                                interval = "confidence", level = 0.95)

prediccion_mujeres <- predict(modelo_con_interaccion, nuevos_datos_mujeres,
                              interval = "prediction", level = 0.95)

# Configurar la ventana gráfica para dos gráficos en la misma fila
par(mfrow = c(1, 2))  # 1 fila, 2 columnas

# Gráfico para hombres
plot(datos$Estatura[datos$sexo_binario == 1], datos$Peso[datos$sexo_binario == 1],
     col = "blue", xlab = "Estatura", ylab = "Peso",
```

```r
     main = "Intervalos para Hombres")

lines(estaturas_seq, predicciones_hombres[, "fit"], col = "blue", lwd = 2)
lines(estaturas_seq, predicciones_hombres[, "lwr"], col = "blue", lty = 2)
lines(estaturas_seq, predicciones_hombres[, "upr"], col = "blue", lty = 2)

lines(estaturas_seq, prediccion_hombres[, "lwr"], col = "lightblue", lty = 3)
lines(estaturas_seq, prediccion_hombres[, "upr"], col = "lightblue", lty = 3)

legend("topleft", legend = c("Predicción", "IC", "IP"),
       col = c("blue", "blue", "lightblue"),
       lty = c(1, 2, 3), lwd = 2)

# Gráfico para mujeres
plot(datos$Estatura[datos$sexo_binario == 0], datos$Peso[datos$sexo_binario == 0],
     col = "red", xlab = "Estatura", ylab = "Peso",
     main = "Intervalos para Mujeres")

lines(estaturas_seq, predicciones_mujeres[, "fit"], col = "red", lwd = 2)
lines(estaturas_seq, predicciones_mujeres[, "lwr"], col = "red", lty = 2)
lines(estaturas_seq, predicciones_mujeres[, "upr"], col = "red", lty = 2)

lines(estaturas_seq, prediccion_mujeres[, "lwr"], col = "pink", lty = 3)
lines(estaturas_seq, prediccion_mujeres[, "upr"], col = "pink", lty = 3)

legend("topleft", legend = c("Predicción", "IC", "IP"),
       col = c("red", "red", "pink"),
       lty = c(1, 2, 3), lwd = 2)
```




```
# Resetear la ventana gráfica para futuros gráficos
par(mfrow = c(1, 1))
```

Conclusiones:

en ambos casos parece ser que los datos de las mujeres se encuentran mas dispersos que los de los hombbres, y únicamente para el caso de las mujeres parece ser que los intervalos de predicción no abarcan todos los datos a comparación de las graficas en los hombres que todos los datos estan acotados en los limites de los intervalos de predicción. Esto parece indicar que para ningúno de los modelos se esta ajustando correctamente para el conjunto de las mujeres o simplemente que los pesos de las mujeres se encuentran mas dispersos que los pesos de los hombres