Trabajo de Evaluación Continua de Modelos de Regresión

Curso 2021/2022

Xiana Carrera Alonso, Pablo Díaz Viñambres

Introducción

En este documento se describe y documenta el ajuste y análisis de un modelo de regresión lineal simple en base a los datos proporcionados para una variable explicativa X y una variable respuesta Y.

El estudio se fundamentará en los conceptos teóricos relacionados con los modelos de regresión lineal simple y su validación que fueron estudiados a lo largo de los Temas 6 y 7 de la asignatura de Inferencia Estadística. Se hará referencia explícita a los mismos a medida que sean empleados.

Asimismo, se utilizará R para realizar las operaciones necesarias para el análisis. Los detalles relativos al empleo de sus funciones se detallarán o bien en el propio informe o bien a través de comentarios sobre el código.

Modelo de regresión lineal simple

Recordemos que un modelo de regresión sirve para representar la dependencia de una variable Y respecto de una o varias variables X. En particular, en el modelo de regresión lineal simple se consideran variables X e Y univariantes (esto es, reflejan el valor de una sola característica) y parte de las hipótesis de linealidad, homocedasticidad y normalidad e independencia de los errores (véase una explicación detallada de las mismas en el ejercicio 7).

Consideraremos una muestra extraída bajo diseño fijo, esto es, con datos $(x_1, Y_1), ..., (x_n, Y_n)$, donde $x_1, ..., x_n$ están fijados por el experimentador.

Así, tendremos

$$Y_i = \beta_0 + \beta_1 \cdot x_i + \epsilon_i \quad \text{para } i \in 1, ..., n$$

donde $\epsilon_1, ..., \epsilon_n \in N(0, \sigma^2)$ y son independientes.

Este modelo presenta 3 parámetros: β_0, el valor inicial de la media de la variable respuesta cuando X es 0 (es decir, la ordenada en el origen de la recta de regresión asociada al modelo); β_1, la cantidad en la que crece dicha media cada vez que X se incrementa en una unidad (la pendiente de la recta); y σ^2, la varianza del error (que, por hipótesis de homocedasticidad, toma un valor fijo para cualquier valor x de la variable explicativa).

Cuestiones preliminares

En cada una de las secciones del documento se trabajará con $\alpha = 0.01$ para los distintos contrastes, intervalos de confianza, etc. planteados. Equivalentemente, se empleará un nivel de significación $1 - \alpha = 0.99$.

```r
alfa <- 1 - 0.99
alfa
```

```
## [1] 0.01
```

```r
nivel <- 1 - alfa
nivel
```

```
## [1] 0.99
```

Esta elección se debe al enunciado del ejercicio 3, donde se pide emplear un nivel de significación del 99% para la construcción de intervalos de confianza de los parámetros del modelo. Para mantener entonces la consistencia en todo el informe, se decidió conservarlo en los demás apartados que lo requieren.

Librerías utilizadas

Cargamos a continuación todas las librerías que utilizaremos a lo largo de la ejecución. Si alguno de los paquetes no ha sido previamente instalado, debe ejecutarse la instrucción `install.packages("nombre_del_paquete")`.

```r
# Para instalar: install.packages("paquete_de_ejemplo")
library(ggplot2) # Para diagrama de dispersión con región de confianza
library(rpanel) # Controles adicionales en sm.regression
library(viridis) # Gradiente de colores en el histograma
library(nortest) # Necesario para lillie.test
library(car) # Necesario para QQPlot
library(sm) # Contraste no paramétrico de linealidad
library(lmtest) # Test de Harrison-McCabe (contraste de homocedasticidad)
```

Lectura de datos

En primer lugar, leemos los datos del archivo proporcionado, que cuenta con 76 variables respuesta, Y1, . . . , Y76, y una variable explicativa común, X. En nuestro caso, limitaremos el estudio a Y47, que denotaremos sencillamente como Y de aquí en adelante.

Nada más importar el archivo (para lo cual es necesario que el usuario cambie el directorio actual, empleando, por ejemplo, `setwd` o `Ctrl + May + H`), realizamos un pequeño análisis descriptivo de los datos empleando las funciones estándar `head`, `class`, `names`, `str` y `summary`. También comprobaremos la varianza y desviación típica de las variables y representaremos un histograma y un boxplot para ver su estructura general.

Por comodidad para cálculos posteriores, también guardamos el número de datos, n.

```r
# Configurar wd a la carpeta actual (solo en RStudio)
setwd(dirname(rstudioapi::getActiveDocumentContext()$path)) # nolint
# Ejemplo de uso de setwd para cambiar el directorio actual:
# setwd("C:\\Users\\Pablo\\Desktop\\IE_Regresion") # nolint

# Leemos los datos empleando read.table (por la extensión .txt)
# Indicamos que existe una cabecera, que las columnas están
# separadas por espacios y que el signo decimal es el punto.
datos <- read.table("datos_trabajo_temas6y7.txt",
        header = T, sep = " ", dec = ".")
# Vemos los nombres de las variables
names(datos)
```

```
##  [1] "Y1"  "Y2"  "Y3"  "Y4"  "Y5"  "Y6"  "Y7"  "Y8"  "Y9"  "Y10" "Y11" "Y12"
## [13] "Y13" "Y14" "Y15" "Y16" "Y17" "Y18" "Y19" "Y20" "Y21" "Y22" "Y23" "Y24"
## [25] "Y25" "Y26" "Y27" "Y28" "Y29" "Y30" "Y31" "Y32" "Y33" "Y34" "Y35" "Y36"
## [37] "Y37" "Y38" "Y39" "Y40" "Y41" "Y42" "Y43" "Y44" "Y45" "Y46" "Y47" "Y48"
## [49] "Y49" "Y50" "Y51" "Y52" "Y53" "Y54" "Y55" "Y56" "Y57" "Y58" "Y59" "Y60"
## [61] "Y61" "Y62" "Y63" "Y64" "Y65" "Y66" "Y67" "Y68" "Y69" "Y70" "Y71" "Y72"
## [73] "Y73" "Y74" "Y75" "Y76" "X"
```

```
# Y nos quedamos con las variables de interés
datos <- datos[, c("X", "Y47")]

# Comprobamos la estructura de las primeras filas
head(datos)
```

```
##        X     Y47
## 1 1.1370  1.4567
## 2 6.2230 -1.2109
## 3 6.0927 -2.1529
## 4 6.2338 -2.1343
## 5 8.6092 -4.9584
## 6 6.4031 -1.3663
```

```
# Comprobamos que el objeto resultante es un data.frame
class(datos)
```

```
## [1] "data.frame"
```

```
# Comprobamos la estructura de los datos
str(datos)
```

```
## 'data.frame':    120 obs. of  2 variables:
##  $ X  : num  1.14 6.22 6.09 6.23 8.61 ...
##  $ Y47: num  1.46 -1.21 -2.15 -2.13 -4.96 ...
```

```
# Seleccionamos las dos variables de interés
X <- datos[, "X"] # nolint
Y <- datos[, "Y47"] # nolint

# Guardamos el número de datos
n <- length(Y)

# Y realizamos un pequeño análisis estadístico exploratorio
summary(datos) # Obtenemos mínimo, máximo, media, mediana y cuantiles
```

```
##        X              Y47
##  Min.   :0.095   Min.   :-7.0839
##  1st Qu.:1.795   1st Qu.:-2.4687
##  Median :3.232   Median : 0.1661
##  Mean   :4.268   Mean   :-0.1815
##  3rd Qu.:6.667   3rd Qu.: 2.3278
##  Max.   :9.921   Max.   : 5.9654
```

```r
var(X) * (n - 1) / n # Varianza de X
```

```
## [1] 8.090512
```

```r
var(Y) * (n - 1) / n # Varianza de Y
```

```
## [1] 9.365897
```

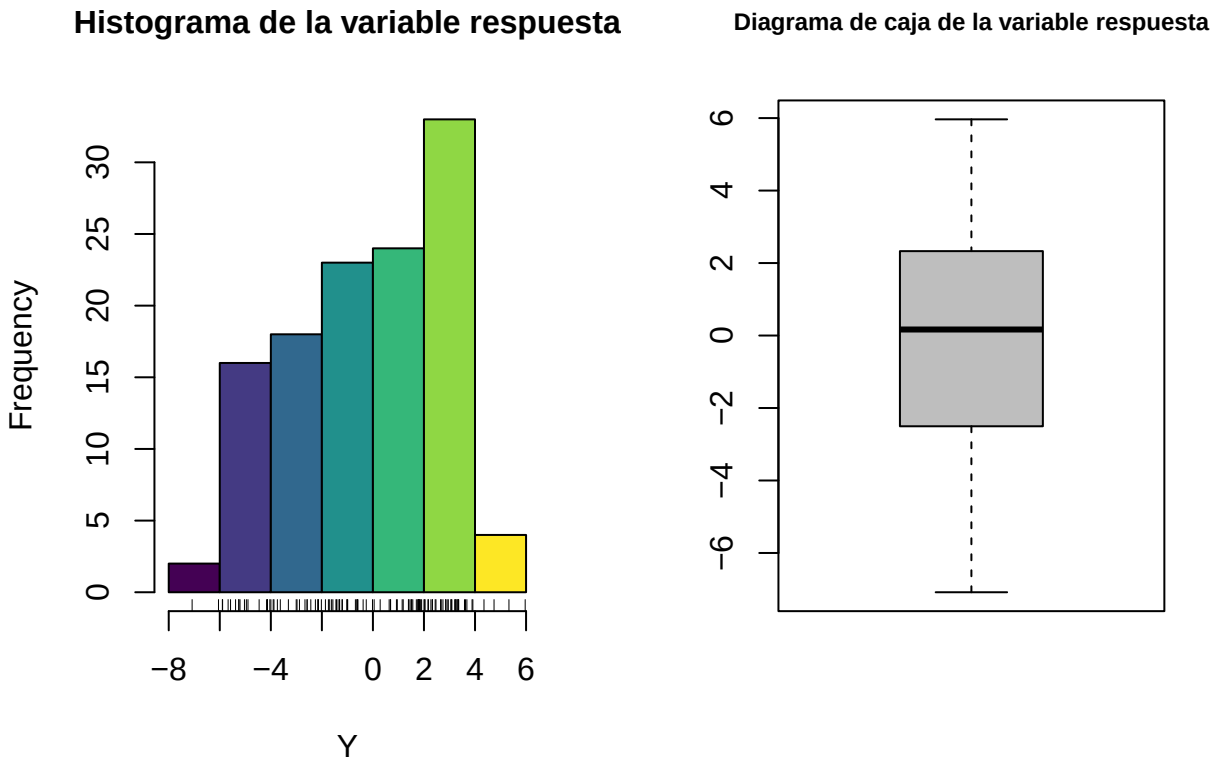
```r
sqrt(var(X) * (n - 1) / n) # Desviación típica de X
```

```
## [1] 2.844383
```

```r
sqrt(var(Y) * (n - 1) / n) # Desviación típica de Y
```

```
## [1] 3.060375
```

```r
par(mfrow = c(1, 2)) # Gráficos dispuestos en 1 fila con 2 columnas

# Utilizamos viridis para crear un gradiente de colores
hist(Y, col = viridis(7),
        main = "Histograma de la variable respuesta", cex.main = 1)
rug(Y)  # Concentración de los puntos de X
boxplot(Y, col = "gray",
        main = "Diagrama de caja de la variable respuesta", cex.main = 0.75)
```



```r
# Reiniciamos al valor predeterminado para las ventanas gráficas
par(mfrow = c(1, 1))
```

Podemos destacar los valores más relevantes del análisis exploratorio. Vemos que los valores de los datos de X abarcan el rango $[0.095, 9.921]$, y los de Y, $[-7.0839, 5.9654]$. Sus respectivas medianas son 3.432 para X, y 0.1661 para Y. Tenemos también que:

$$\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i = 4.268 \text{ unidades de X}, \quad \overline{Y} = \frac{1}{n}\sum_{i=1}^{n} Y_i = -0.1815 \text{ unidades de Y}$$

$$S_x^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x}) = 2.844383 \text{ (unidades de X)}^2, \quad S_Y^2 = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \overline{Y}) = 9.365897 \text{ (unidades de Y)}^2$$

$$S_x = \sqrt{S_x^2} = 2.844383 \text{ unidades de X}, \quad S_Y = \sqrt{S_Y^2} = 3.060375 \text{ unidades de Y}$$

Varios de estos valores, como la media o la varianza, nos serán de interés a lo largo del ajuste del modelo de regresión, y volveremos a ellos en puntos posteriores.

1) Relación entre variable explicativa y variable respuesta

En primer lugar, calculamos la covarianza y el coeficiente de correlación entre las variables:

$$S_{xY} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})(Y_i - \overline{Y}) \qquad r_{xY} = \frac{S_{xY}}{\sqrt{S_x^2}\sqrt{S_Y^2}}$$

Debemos tener en cuenta que R calcula la covarianza como una 'cuasi'covarianza, es decir, dividiendo entre $n-1$ en lugar de entre n. Para corregirlo, multiplicamos por $n-1$ y dividimos entre n, aunque también mostraremos el valor original. No afectará al coeficiente de correlación, pues el denominador se anula con los $\sqrt{n-1}$ de las cuasidesviaciones típicas.

```r
covar <- cov(X, Y) * (n - 1) / n
covar # Covarianza
```

```
## [1] -8.275695
```

```r
cov(X, Y) # Cuasicovarianza
```

```
## [1] -8.345238
```

```r
cor(X, Y) # Coeficiente de correlación
```

```
## [1] -0.9506962
```

Que la covarianza sea distinta de 0 nos indica que hay una relación lineal. Además, al ser negativa, deducimos que esta es indirecta/inversa, es decir, que al aumentar la variable X, la variable Y diminuye. Esto nos sirve de fundamento para que posteriormente planteemos una recta de regresión, pues por el contrario una covarianza

nula o muy próxima a 0 sería indicativa de que no existe una relación lineal significativa entre las variables (podría no haber relación, o ser esta de otro tipo, como cuadrática, cúbica, etc.). En tal caso, deberíamos descartar el ajute a un modelo de regresión lineal.

Adionalmente, ya podemos anticipar que la mencionada recta de regresión será decreciente, pues la pendiente estimada con la que la construiremos, $\beta_1 = \frac{S_{xY}}{S_x^2}$, tiene el mismo signo que S_{xY}, al ser la varianza siempre no negativa, y hemos obtenido que $S_{xY} < 0$.

Por un razonamiento análogo, r_{xY} también debe tener el mismo signo que S_{xY} y, en efecto, esto es lo que observamos en los resultados. La interpretación de su signo es, por tanto, la misma que la expuesta para la covarianza (relación lineal inversa/indirecta).

Ahora bien, no podemos sacar conclusiones acerca de la magnitud de la covarianza, pues esta tiene unidades (que ni siquiera conocemos). Por el contrario, el coeficiente de correlación es adimensional y, de hecho, sabemos que $r_{xY} \in [-1, 1]$. Como $|r_{xY}| > 0.75$, la relación entre las variables es fuerte. Esto es, tienen una correlación significativa. Cuando representemos el diagrama de dispersión de los datos y sobre el mismo, la recta de regresión, observaremos que los puntos son próximos a esta.

Representación gráfica

Para visualizar la relación entre la variable explicativa y la variable respuesta, emplearemos un diagrama de dispersión.

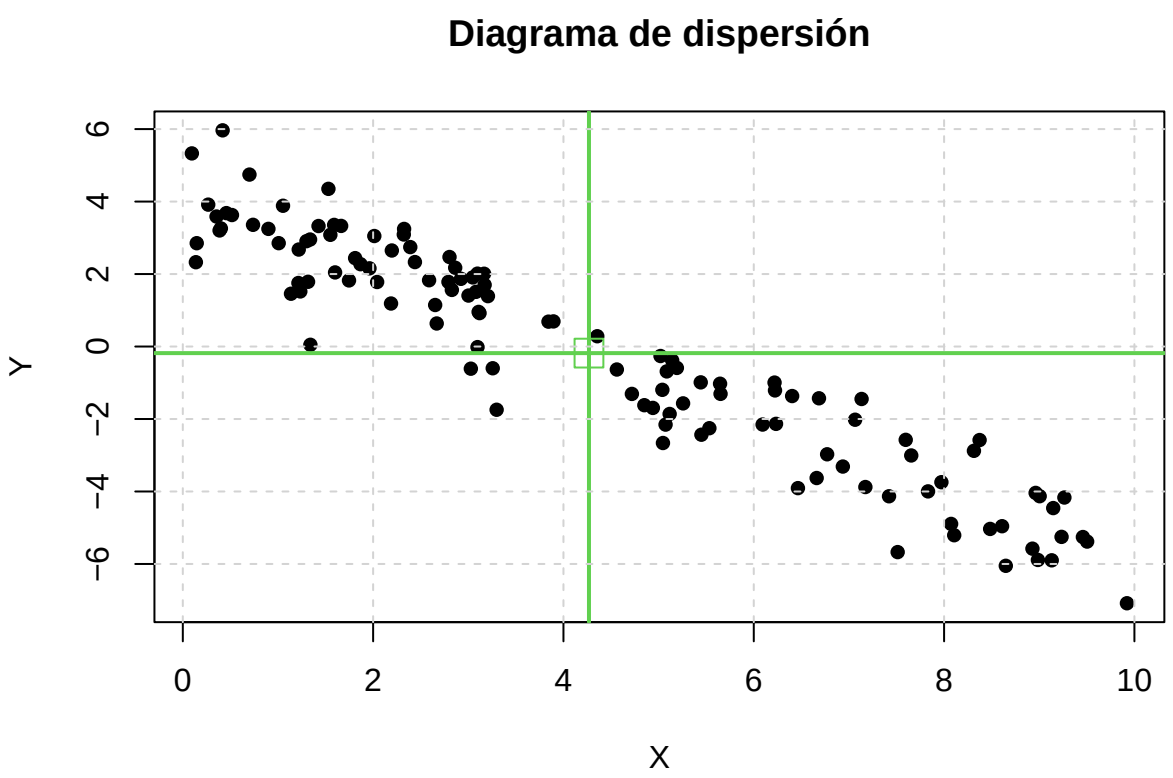
En primer lugar, hallamos el vector de medias o centro de gravedad aplicando `mean` en ambas variables:

```
mX <- mean(X) # Media de X
mY <- mean(Y) # Media de Y

c(mX, mY) # Mostramos el vector de medias
```

```
## [1]  4.2681600 -0.1815108
```

Como diagrama básico, emplearemos la función `plot`. Como recurriremos a este gráfico en particular en varias ocasiones a lo largo de este documento, vamos a definir una función que englobe la representación:

```
representar <- function() {
        plot(X, Y,
          main = "Diagrama de dispersión", pch = 16,
          sub = "Relación entre la variable explicativa y la variable respuesta",
                cex.sub = 0.60
        )

        # Añadimos un punto para el vector de medias
        points(mX, mY, pch = 12, col = 3, cex = 2)

        # Añadimos dos rectas para dividir en cuadrantes,
        # limitados por el vector de medias
        abline(v = mX, col = 3, lty = 1, lwd = 2) # Vertical
        abline(h = mY, col = 3, lty = 1, lwd = 2) # Horizontal

        # Añadimos una rejilla de fondo
        grid(lty = 2, col = "lightgray", lwd = 1)
}

representar() # Ejecutamos la función
```



Relación entre la variable explicativa y la variable respuesta

Se puede ver que la nube de puntos toma una forma descendente, lo cual es coherente con la correlación negativa de X e Y. También vemos que los datos están, de forma aproximada, uniformemente alineados en torno a una forma rectilínea. Todo esto motiva el establecimiento de un modelo lineal para la relación entre ambas variables que, recordemos, son de la forma:

$$Y = \beta_0 + \beta_1 X + \epsilon$$

Ajustamos entonces este modelo lineal a nuestros datos mediante la función `lm`, indicando Y como la variable dependiente, y X como la independiente:

```
modelo <- lm(Y ~ X)
modelo
```

```
##
## Call:
## lm(formula = Y ~ X)
##
## Coefficients:
## (Intercept)            X
##       4.184       -1.023
```

Obtenemos un $\beta_0 = 4.184$. Este es el intercepto, es decir, el valor estimado de Y cuando $X = 0$. La pendiente estimada de la recta será $\beta_1 = -1.023$. Esto será lo que disminuya la variable respuesta cuando la variable explicativa aumente en una unidad. Ambos resultados concuerdan con lo observado anteriormente en la nube de puntos.

El modelo obtenido almacena otros datos de interés, tales como los residuos (`modelo$residuals`), los valores ajustados de la Y (`modelo$fitted.values`), etc. Vemos más información haciendo un `summary`:

```
summary(modelo)
```

```
##
## Call:
## lm(formula = Y ~ X)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.76465 -0.72493  0.00685  0.71260  2.20924
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.18434    0.15755   26.56   <2e-16 ***
## X           -1.02289    0.03072  -33.30   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9571 on 118 degrees of freedom
## Multiple R-squared:  0.9038, Adjusted R-squared:  0.903
## F-statistic:  1109 on 1 and 118 DF,  p-value: < 2.2e-16
```

En la primera línea se nos muestra la fórmula empleada: Y~X.

A continuación aparece información descriptiva sobre los residuos. Podemos destacar que el valor del mínimo, -2.76465, es relativamente simétrico con respecto al del máximo, 2.20924, y que la mediana es próxima a 0: es 0.00685. Esto no nos da motivos para sospechar en contra del no cumplimiento de las hipótesis de homocedasticidad o normalidad.

Después se muestra información acerca de los coeficientes del modelo (β_0 y β_1), que obtendremos en el ejercicio 2. Por columnas, se indican:

1. `estimate`: los valores estimados del modelo.
2. `std. error`: el error típico estimado para el intercepto y para la pendiente, respectivamente.
3. `t value`: el cociente entre la columna `estimate` y la columna `std. error`.
4. `p value`: el p-valor de los contrastes de significación asociados.

En las últimas líneas se indica el `residual standard error`, que es la desviación típica del error, junto con los grados de la chi-cuadrado asociada a la varianza del error ($n - 2 = 118$). Después nos aparece el coeficiente de determinación (0.9038) y el coeficiente de determinación ajustado. Este último no nos será de especial interés, al ser este un modelo simple y no múltiple. Por último, se muestra el valor del estadístico de contraste del F-test (1109), que es equivalente al test de significación de la pendiente. La F de Snédecor empleada es de 1 grado de libertad en el numerador y 118 en el denominador, y el p-valor asociado es muy próximo a 0 ($< 2.2 \cdot 10^{-16}$). Veremos que esto demuestra que el parámetro de la pendiente es significativo.

En los siguientes ejercicios, analizaremos en profundidad otras características de este modelo y haremos inferencia a partir del mismo. Nótese que para que las conclusiones extraídas en estos ejercicios tengan validez, deberemos suponer que se cumplen las hipótesis de linealidad, homocedasticidad y normalidad e independencia de los errores. Las comprobaremos de forma precisa en el ejercicio 7, pero de no ser válida alguna de ellas, tendríamos que revisar y descartar multitud de resultados.

2) Estimaciones puntuales de los parámetros y representación del modelo

Para la estimación puntual de los parámetros intercepto β_0, pendiente β_1 y para la de la varianza del error σ^2 podemos aplicar directamente las fórmulas obtenidas en la parte teórica de la asignatura:

$$\hat{\beta}_0 = \overline{Y} - \frac{S_{xY}}{s_x^2}\overline{x}$$

$$\hat{\beta}_1 = \frac{S_{xY}}{s_x^2}$$

$$\hat{\sigma}^2 = \frac{1}{n-2} \cdot \sum_{i=1}^{n} (Y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i)^2$$

Veamos a través de R cuáles son sus valores:

```
# Obtenemos la varianza multiplicando la cuasivarianza por (n-1)/n
var.X <- var(X) * (n - 1) / n
beta0.gorro <- mY - covar * mX / var.X
beta0.gorro
```

```
## [1] 4.184343
```

```
beta1.gorro <- covar / var.X
beta1.gorro
```

```
## [1] -1.022889
```

```
var.error <- sum((Y - beta0.gorro - beta1.gorro * X)^2) / (n - 2)
var.error
```

```
## [1] 0.916048
```

```
sd.error <- sqrt(var.error)
# Vemos también el valor estimado de la desviación típica del error
sd.error
```

```
## [1] 0.957104
```

De manera alternativa, podemos obtenerlas a partir del propio modelo creado anteriomente por R:

```
modelo
```

```
##
## Call:
## lm(formula = Y ~ X)
##
## Coefficients:
## (Intercept)            X
##       4.184       -1.023
```

```r
# Intercepto beta0gorro y pendiente beta1gorro
modelo$coefficients
```

```
## (Intercept)           X
##    4.184343   -1.022889
```

```r
# Varianza del error
sum(modelo$residuals^2) / (n - 2)
```

```
## [1] 0.916048
```

En el código anterior hemos utilizado `modelo$residuals` para obtener los residuos del modelo (los errores de predicción):

$$\hat{\epsilon}_i = Y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i \quad \text{para } i \in 1, ..., n,$$

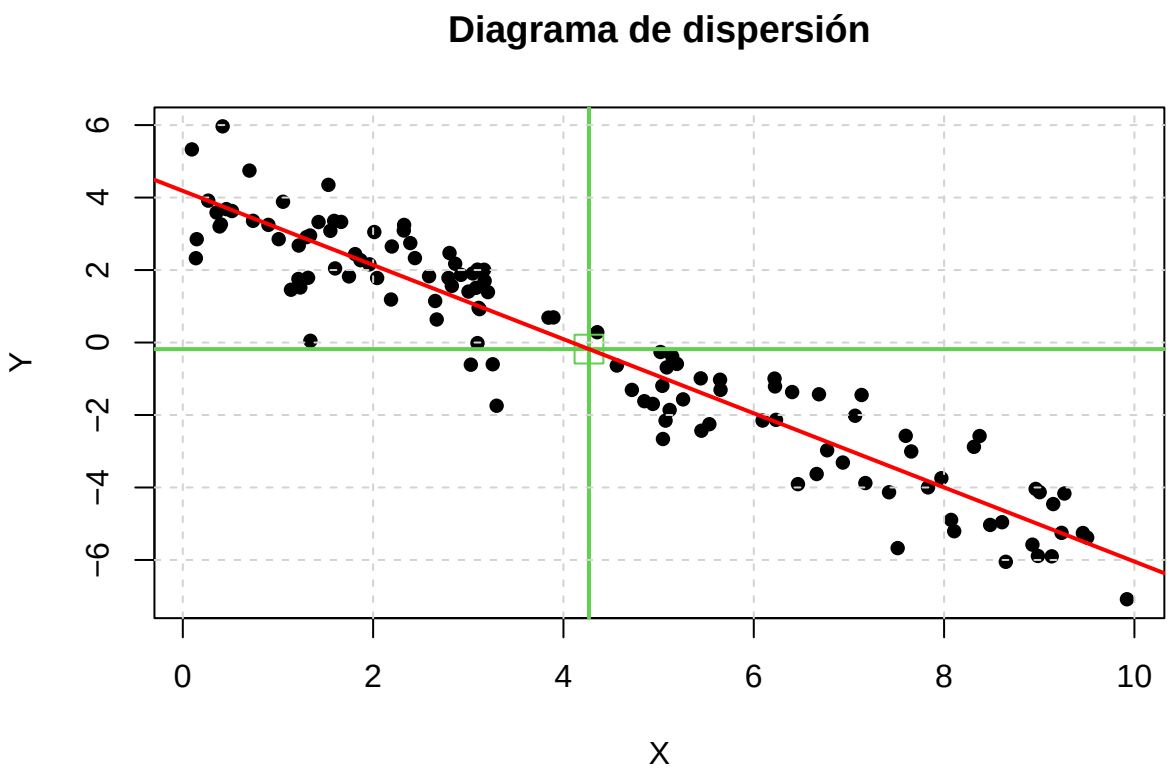
así como la expresión alternativa de la varianza del error:

$$\hat{\sigma}^2 = \frac{1}{n-2} \cdot \sum_{i=1}^{n} \hat{\epsilon}_i^2$$

Vemos que nuestros cálculos coinciden con los obtenidos por R, ya que las fórmulas empleadas son las mismas.

A continuación, en base a la representación definida anteriormente, añadimos la recta de regresión ajustada del modelo:

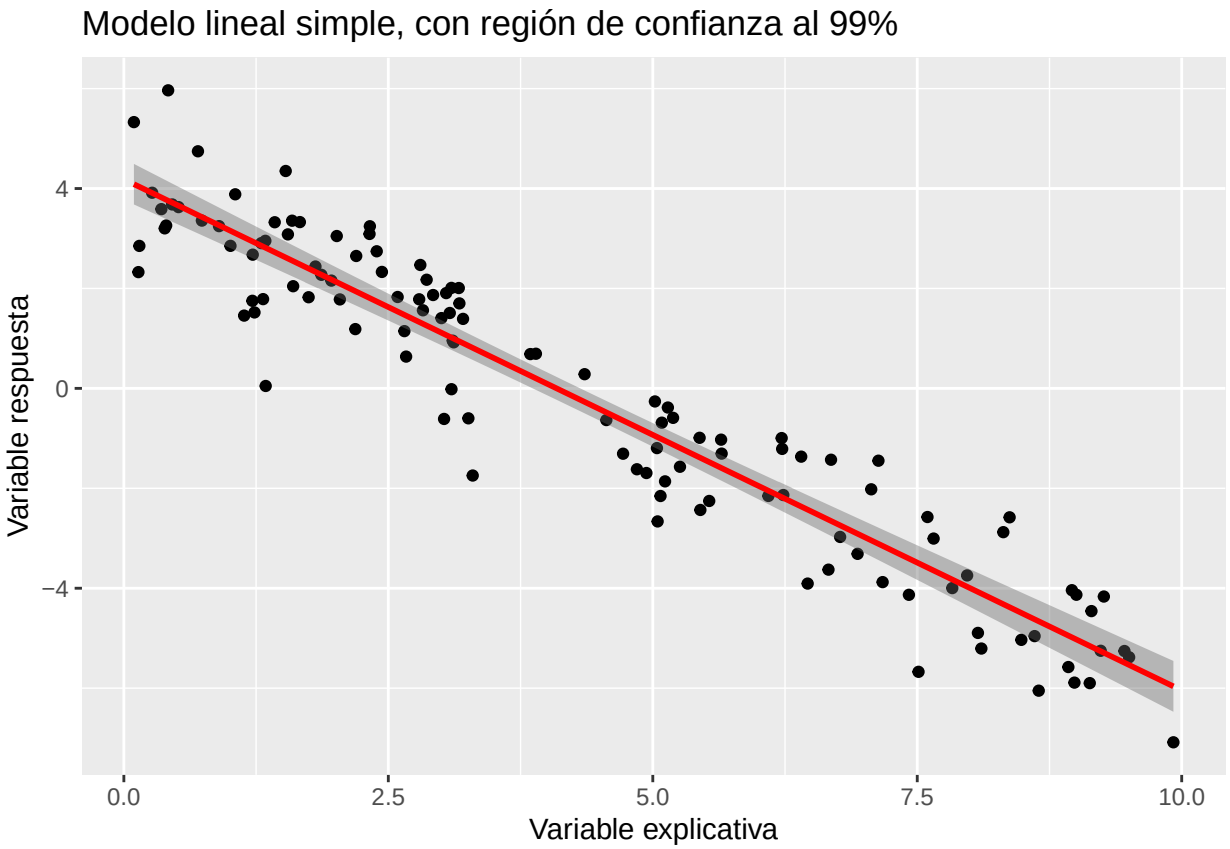
```r
# Llamamos a la función que crea el diagrama de dispersión de Y sobre X
representar()
# Añadimos la recta de regresión
abline(modelo, col = "red", lwd = 2)
```



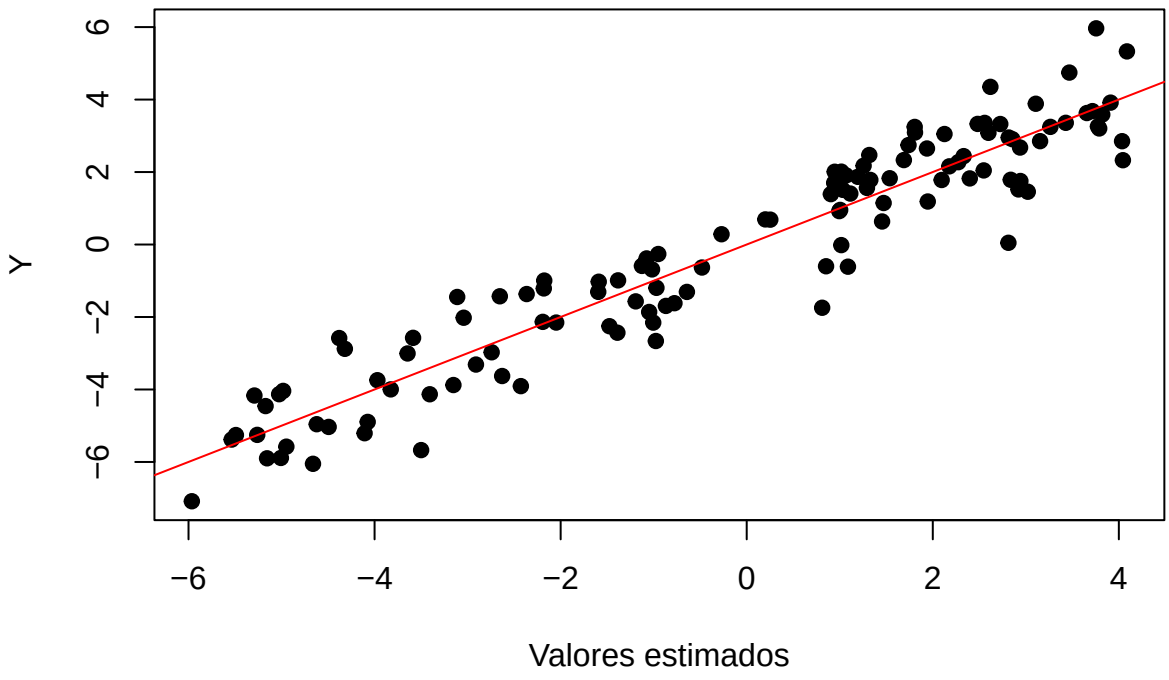
Relación entre la variable explicativa y la variable respuesta

Incluimos también una gráfica adicional usando la librería `ggplot2` e incluyendo la región o intervalo de confianza para los datos al nivel fijado del 99%:

```
ggplot(datos, aes(x = X, y = Y)) +
        geom_point() +
        geom_smooth(formula = y ~ x, level = 0.99, method = lm,
                color = "red", fill = "#666666", se = TRUE) +
        labs(
                y = "Variable respuesta",
                x = "Variable explicativa",
                title = "Modelo lineal simple, con región de confianza al 99%"
        )
```



Podemos representar también la relación entre los valores de Y y las estimaciones que da para cada uno de ellos el modelo (los $\tilde{Y}$). La aproximación será mejor cuanto menos se distancien los puntos de la diagonal, pues esta es la representación de $Y = \tilde{Y}$, que es lo que ocurriría en el hipótetico caso de que el modelo no tuviera error. Cabe destacar que este gráfico nos sirve como una medida de bondad de ajuste del modelo, ya que nos indica si este es adecuado o no.



3) Intervalos de confianza de los parámetros del modelo

A continuación, hallamos sendos intervalos de confianza para β_0, β_1 y σ^2. Todos estos se harán para el nivel de significación $\alpha = 0.01$ definido previamente.

Intervalos para β_0

El pivote que emplearemos para esta estimación será

$$\frac{\hat{\beta}_0 - \beta_0}{\hat{\sigma}\sqrt{\frac{1}{n} + \frac{\overline{x}^2}{nS_x^2}}} \in T_{n-2}$$

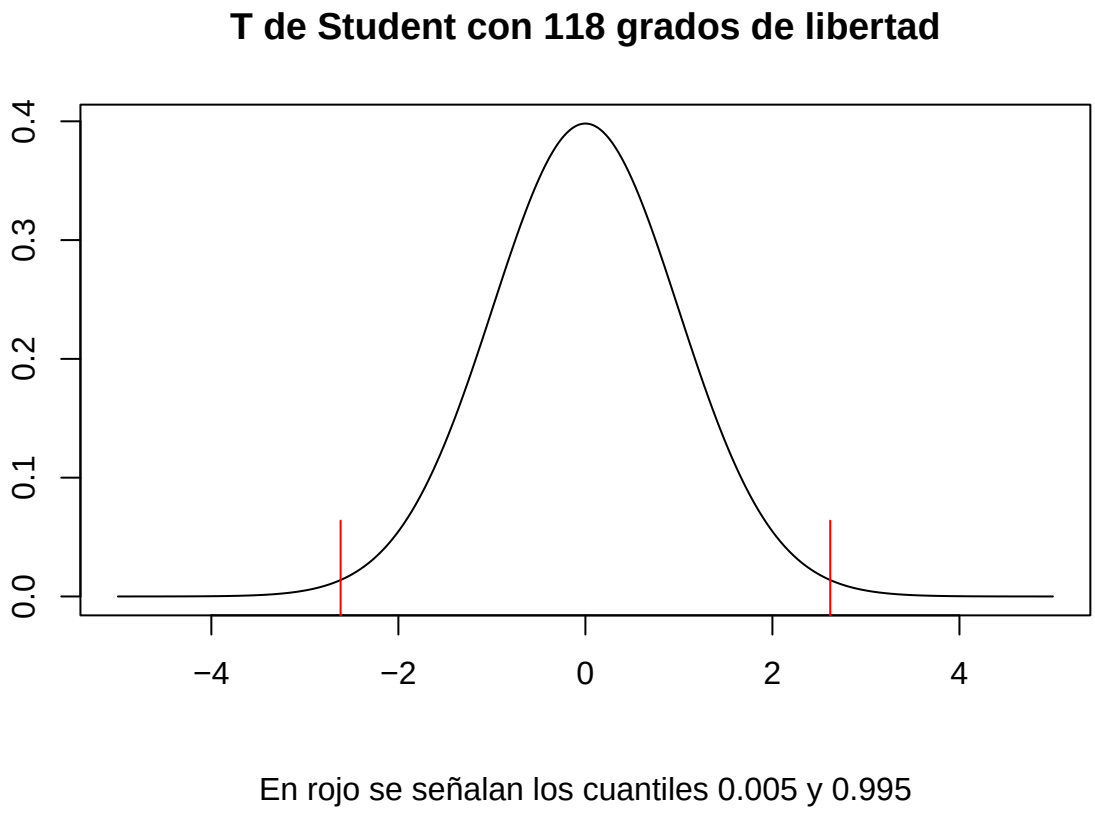
de modo que el intervalo de confianza será:

$$(\hat{\beta}_0 - t_{n-2,\alpha/2}\ \hat{\sigma}\ \sqrt{\frac{1}{n} + \frac{\overline{x}^2}{nS_x^2}},\quad \hat{\beta}_0 + t_{n-2,\alpha/2}\ \hat{\sigma}\ \sqrt{\frac{1}{n} + \frac{\overline{x}^2}{nS_x^2}})$$

dónde $t_{n-2,\frac{\alpha}{2}}$ es el cuantil $1 - \frac{\alpha}{2} = 0.995$ de una T de Student con $n - 2 = 118$ grados de libertad.

```r
# Creamos un vector de valores para el eje X
x_student <- seq(-5, 5, by = 0.01)

# Le aplicamos la función de densidad a cada punto
y_tstudent <- dt(x_student, df = n - 2)
```

```r
# Calculamos el cuantil 1-alfa/2
beta0.cuantil <- qt(1 - alfa / 2, df = n - 2)

# Y representamos
plot(x_student, y_tstudent,
        type = "l",
        main = "T de Student con 118 grados de libertad",
        xlab = "", ylab = "",
        sub = "En rojo se señalan los cuantiles 0.005 y 0.995"
)
segments(x0 = beta0.cuantil, y0 = -0.05, x1 = beta0.cuantil,
        y1 = dt(beta0.cuantil, df = n - 2) + 0.05, col = "red")
segments(x0 = -beta0.cuantil, y0 = -0.05, x1 = -beta0.cuantil,
        y1 = dt(-beta0.cuantil, df = n - 2) + 0.05, col = "red")
```



En primer lugar, obtengamos el intervalo de confianza a mano:

```r
# Cuantil 1-alfa/2 de T de Student con n-2 grados de libertad
# ya calculado en beta0.cuantil
beta0.extremoinferior <- beta0.gorro -
        beta0.cuantil * sqrt(var.error * (1 / n + mX^2 / (n * var.X)))
beta0.extremosuperior <- beta0.gorro +
        beta0.cuantil * sqrt(var.error * (1 / n + mX^2 / (n * var.X)))
beta0.IC <- c(beta0.extremoinferior, beta0.extremosuperior)
beta0.IC # Intervalo de confianza
```

```
## [1] 3.771852 4.596833
```

Interpretamos que, en base a los datos de esta muestra, el intervalo $(3.771852, 4.596833)$ contendrá al parámetro β_0 con una probabilidad del 99%.

Intervalos para β_1

Para el intervalo de confianza de la pendiente, usaremos ahora el pivote

$$\frac{\hat{\beta_1} - \beta_1}{\hat{\sigma}/S_x\sqrt{n}} \in T_{n-2}$$

a partir del cuál obtenemos el intervalo:

$$(\hat{\beta}_1 - t_{n-2,\alpha/2}\frac{\hat{\sigma}}{S_x\sqrt{n}}, \hat{\beta}_1 + t_{n-2,\alpha/2}\frac{\hat{\sigma}}{S_x\sqrt{n}})$$

donde de nuevo $t_{n-2,\frac{\alpha}{2}}$ es el cuantil $1-\frac{\alpha}{2} = 0.995$ de una T de Student con $n-2 = 118$ grados de libertad.

Obtenemos el intervalo a mano:

```
# Empleamos el mismo cuantil, dado que fijamos el mismo alfa
beta1.cuantil <- beta0.cuantil
beta1.extremoinferior <- beta1.gorro -
        beta1.cuantil * sqrt(var.error / (var.X * n))
beta1.extremosuperior <- beta1.gorro +
        beta1.cuantil * sqrt(var.error / (var.X * n))
beta1.IC <- c(beta1.extremoinferior, beta1.extremosuperior); beta1.IC
```

```
## [1] -1.1033105 -0.9424673
```

Interpretamos así que, en base a los datos de esta muestra, el intervalo $(-1.1033105, -0.9424673)$ contendrá al parámetro β_1 con una probabilidad del 99%.

Es interesante observar, que tanto como β_0 como para β_1 los intervalos están muy alejados del 0, esto nos da indicaciones acerca de que el resultado más probable del contraste de significación, que realizaremos en el ejercicio 4, será que estos parámetros son significativos y que no deberemos descartarlos.

Intervalos para σ^2

Por último, para el intervalo de confianza de la varianza del error, usaremos el pivote

$$\frac{(n-2)\hat{\sigma}^2}{\sigma^2} \in \chi^2_{n-2}$$

con lo que obtenemos el intervalo:

$$(\frac{(n-2)\hat{\sigma}^2}{\chi^2_{n-2,\frac{\alpha}{2}}}, \frac{(n-2)\hat{\sigma}^2}{\chi^2_{n-2,1-\frac{\alpha}{2}}})$$

donde $\chi^2_{n-2,\alpha/2}$ es el cuantil $1-\frac{\alpha}{2} = 0.995$ de una χ^2 con $n-2 = 118$ grados de libertad, y $\chi^2_{n-2,1-\alpha/2}$ es el cuantil $\frac{\alpha}{2} = 0.005$ de una χ^2 con $n-2 = 118$ grados de libertad.

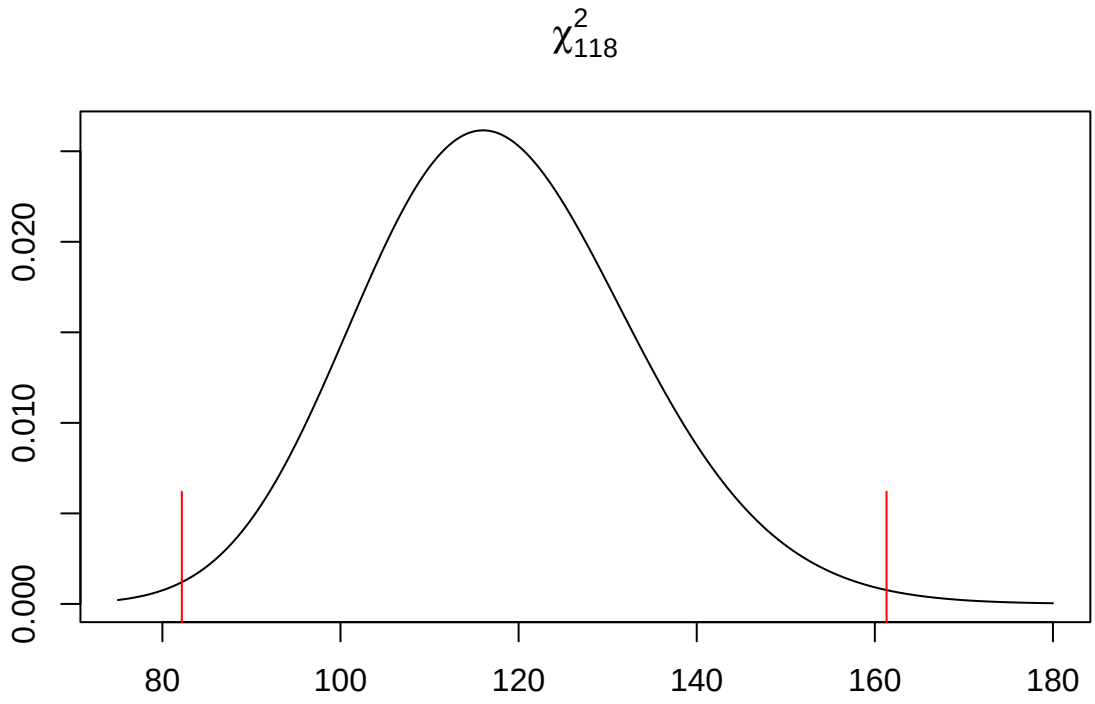
```r
# Creamos una sequencia de puntos para el eje X
x_chi <- seq(75, 180, by = 0.01)

# Le aplicamos la función de densidad a cada punto de la secuencia
y_chi <- dchisq(x_chi, df = n - 2)
# Cuantil alfa/2 de chi-cuadrado con n-2 grados de libertad
var.error.cuantilinferior <- qchisq(alfa / 2, df = n - 2)
var.error.cuantilinferior
```

```
## [1] 82.18544
```

```r
var.error.cuantilsuperior <- qchisq(1 - alfa / 2, df = n - 2) # Cuantil 1-alfa/2
var.error.cuantilsuperior
```

```
## [1] 161.3141
```

```r
# Y representamos
plot(x_chi, y_chi,
        type = "l",
        main = expression(chi[118]^2), xlab = "", ylab = "",
        sub = "En rojo se señalan los cuantiles 0.005 y 0.995"
)
segments(x0 = var.error.cuantilinferior, y0 = -0.05,
        x1 = var.error.cuantilinferior,
        y1 = dchisq(var.error.cuantilinferior, df = n - 2) + 0.005, col = "red")
segments(x0 = var.error.cuantilsuperior, y0 = -0.05,
        x1 = var.error.cuantilsuperior,
        y1 = dchisq(var.error.cuantilinferior, df = n - 2) + 0.005, col = "red")
```



En rojo se señalan los cuantiles 0.005 y 0.995

Calculamos entonces este intervalo a mano:

```
var.error.extremoinferior <- (n - 2) * var.error^2 / var.error.cuantilsuperior
var.error.extremosuperior <- (n - 2) * var.error^2 / var.error.cuantilinferior
var.error.IC <- c(var.error.extremoinferior, var.error.extremosuperior)
var.error.IC # Intervalo
```

```
## [1] 0.6138273 1.2048240
```

Interpretamos así que, en base a los datos de esta muestra, el intervalo $(0.6138273, 1.2048240)$ contendrá al parámetro σ^2 con una probabilidad del 99%.

Pero también podemos utilizar las funciones de R para calcular los intervalos de confianza para β_0 y β_1 de forma automática:

```
# Intervalo de confianza para la ordenada en el origen y la pendiente
# con nivel del 99% en base al modelo construido
confint(modelo, level = 0.99)
```

```
##                 0.5 %     99.5 %
## (Intercept)  3.771852  4.5968334
## X           -1.103311 -0.9424673
```

Vemos que los valores obtenidos coinciden con los extremos del intervalo que calculábamos anteriormente, pues R emplea la misma construcción que hemos visto de forma teórica.

Todos los intervalos de confianza aquí obtenidos tienen carácter aleatorio. De cada 100 intervalos así obtenidos, 99 contendrían al parámetro asociado.

No hay ninguna función estándar para la automatización del intervalo de confianza para la varianza del error.

4) Contrastes de significación asociados al intercepto y a la pendiente

A continuación, realizaremos los contrastes de significación sobre los parámetros $\hat{\beta}_0$ y $\hat{\beta}_1$ con el objetivo de determinar si el modelo se podría simplificar a uno con menos variables o no, y si el modelo de regresión tiene sentido, respectivamente. Las hipótesis de estos contrastes son pues:

$$\begin{cases} H_0 : \beta_0 = 0 \\ H_a : \beta_0 \neq 0 \end{cases}$$

y

$$\begin{cases} H_0 : \beta_1 = 0 \\ H_a : \beta_1 \neq 0 \end{cases}$$

Si no consiguiéramos demostrar la hipótesis alternativa para el primer contraste, tendríamos que considerar un modelo más sencillo, que no solo facilitaría el análisis estadístico, sino que sería más correcto y realista de cara a las interpretaciones que podamos hacer. En el caso del segundo contraste, no demostrar la hipótesis alternativa equivaldría a invalidar nuestro modelo, pues significaría que no hay regresión.

Emplearemos la significación definida para el análisis: $\alpha = 0.01$.

En primer lugar, calculamos los contrastes de forma manual.

Los estadísticos se construirán a partir de los pivotes empleados para obtener los intervalos de confianza del ejercicio 3, teniendo en cuenta que, bajo las respectivas hipótesis nulas, $\beta_0 = 0$ y $\beta_1 = 0$:

$$T_0 = \frac{\hat{\beta}_0}{\hat{\sigma}\sqrt{\frac{1}{n} + \frac{\overline{x}^2}{nS_x^2}}} \in T_{n-2} \qquad T_1 = \frac{\hat{\beta}_1}{\hat{\sigma}/S_x\sqrt{n}} \in T_{n-2}$$

Como estamos en un contraste bilateral, rechazaremos la hipótesis nula si los valores absolutos de los estadísticos son superiores al cuantil positivo de la T de Student, esto es,

$$\text{Rechazar } H_0 : \beta_0 = 0 \text{ si } |T_0| = \frac{|\hat{\beta}_0|}{\hat{\sigma}\sqrt{\frac{1}{n} + \frac{\overline{x}^2}{nS_x^2}}} > t_{n-2,\alpha/2}$$

$$\text{Rechazar } H_1 : \beta_1 = 0 \text{ si } |T_1| = \frac{|\hat{\beta}_1|}{\hat{\sigma}/S_x\sqrt{n}} > t_{n-2,\alpha/2}$$

Calculemos ambos estadísticos y comprobemos si están en la región de rechazo o en la de aceptación.

```
# Cálculo de los valores de los estadísticos de contraste
# para beta0 y beta1 bajo H0
beta0.t <- beta0.gorro / (sqrt(var.error * (1 / n + mX^2 / (n * var.X))))
beta0.t
```

```
## [1] 26.55861
```

```r
beta1.t <- beta1.gorro / (sd.error / sqrt(n * var.X))
beta1.t
```

```
## [1] -33.30029
```

```r
# Ambos contrastes caen en la región de rechazo
abs(beta0.t) > beta0.cuantil
```

```
## [1] TRUE
```

```r
abs(beta1.t) > beta1.cuantil
```

```
## [1] TRUE
```

Habiendo fijado previamente el nivel de significación $\alpha = 0.01$, llegamos en ambos casos a que no existen evidencias estadísticamente significativas a favor de H_a. Es decir, no tenemos pruebas a favor de H_0. Podemos asumir entonces que tanto el intercepto β_0 como la pendiente β_1 del modelo serán distintos de 0. En base a esto, lo sensato será no simplificar nuestro modelo y continuar realizando regresión lineal con estos 2 parámetros.

Podemos también ver los respectivos p-valores (el menor nivel de significación para el cual podemos aceptar la hipótesis nula). Puesto que trabajamos con un contraste bilateral, esta es la probabilidad que queda en las colas limitadas por $|T_{obs}|$ y $-|T_{obs}|$, donde T_obs es el valor del estadístico observado:

$$\mathbb{P}(|T| > |T_{obs}|), \text{ donde T es una distribución } T_{n-2}$$

Así,

```r
beta0.pvalor <- 2 * (1 - pt(abs(beta0.t), df = n - 2)); beta0.pvalor
```

```
## [1] 0
```

```r
beta1.pvalor <- 2 * (1 - pt(abs(beta1.t), df = n - 2)); beta1.pvalor
```

```
## [1] 0
```

Como vemos, en ambos contrastes el p-valor es nulo (prácticamente nulo), lo cual nos indica que las pruebas para rechazar la hipótesis nula son estadísticamente significativas no solo para nuestro α, sino también para cualquiera de los niveles de significación habituales o incluso con precisiones mucho mayores.

Alternativamente, podemos obtener los valores de estos dos contrastes de significación y su p-valor a partir de los datos presentes en el modelo de R. Para esto, usaremos la función summary:

```r
summary(modelo)
```

```
##
## Call:
## lm(formula = Y ~ X)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -2.76465 -0.72493  0.00685  0.71260  2.20924
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.18434    0.15755   26.56   <2e-16 ***
## X           -1.02289    0.03072  -33.30   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9571 on 118 degrees of freedom
## Multiple R-squared:  0.9038, Adjusted R-squared:  0.903
## F-statistic:  1109 on 1 and 118 DF,  p-value: < 2.2e-16
```

En concreto, los valores relevantes son el `t-value` y el `Pr(>|t|)` de las filas `(Intercept)` y `X` que se corresponden al cociente entre la columna `estimate` y la columna `std.error` (el valor observado del estadístico de contraste) y su p-valor, para el intercepto β_0 y la pendiente β_1 respectivamente. Como en ambos casos el p-valor es extremadamente pequeño (R nos indica que es menor que $2 \cdot 10^{-16}$), las interpretaciones coinciden con las ya expuestas: β_0 y β_1 son significativamente distintas de 0 y tienen ambas efecto sobre la variable respuesta. Rechazamos entonces la hipótesis nula y demostramos que $\beta_0 \neq 0$ y $\beta_1 \neq 0$.

5) Intervalos de confianza para la predicción sobre nuevos datos

En este apartado, consideremos los 3 nuevos valores para la variable explicativa: $2, 4, 6$. Obtendremos intervalos de confianza para la media condicionada de Y e intervalos de predicción. Para ello, será necesario comprobar primero que estos datos están dentro del rango de observación de X. Esto es necesario ya que no sabemos cómo se comporta el modelo fuera del rango observado, y no podemos hacer extrapolaciones de forma confiable.

```
nuevosValores <- c(2, 4, 6); nuevosValores
```

```
## [1] 2 4 6
```

```
# Los nuevos valores están contenidos dentro del rango observado
min(X) < min(nuevosValores) && max(X) > max(nuevosValores)
```

```
## [1] TRUE
```

```
# Construimos un data.frame con los nuevos datos
nuevosDatos <- data.frame("X" = nuevosValores)
```

Habiendo realizado esta verificación, ya podemos obtener los intervalos indicados, que se calcularán para el nivel de significación fijado anteriormente $\alpha = 0.01$. En primer lugar, hallaremos los intervalos de confianza para la media condicionada utilizando el estimador $\tilde{Y}_0 = \hat{\beta}_0 + \hat{\beta}_1 x_0 = \overline{Y} + \hat{\beta}_1(x_0 - \overline{x})$, que es insesgado para $\mathbb{E}[Y|X = x_0]$ y tiene varianza

$$\text{Var}(\tilde{Y}_0) = \frac{\sigma^2}{n_0}, \text{ dónde } n_0 = \frac{n}{1 + \frac{(x_0 - \overline{x})^2}{S_x^2}}$$

n_0 representa el número de observaciones disponibles para la estimación de $\mathbb{E}[Y|X = x_0]$, que depende de la proximidad de x_0 a $\overline{x}$ (n_0 es menor cuanto mayor es la distancia entre ambos, al alejarse x_0 de los valores muestrales). Por ese motivo, n_0 será mayor para 4 que para 2 y 6.

$\tilde{Y_0}$ es el valor estimado de la nueva media, condicionada a $X = x_0$.

En consecuencia, obtenemos el intervalo:

$$(\tilde{Y_0} - t_{n-2,\alpha/2}\,\frac{\hat{\sigma}}{\sqrt{n_0}}, \tilde{Y_0} + t_{n-2,\alpha/2}\,\frac{\hat{\sigma}}{\sqrt{n_0}})$$

que podemos calcular como sigue:

```
# Calculo del estimador y0tilde y del factor n0 del intervalo
nuevosValores.y0tilde <- beta0.gorro + beta1.gorro * nuevosValores
nuevosValores.y0tilde
```

```
## [1]  2.13856482  0.09278705 -1.95299072
```

```
nuevosValores.n0 <- n / (1 + (nuevosValores - mX)^2 / var.X)
nuevosValores.n0
```

```
## [1]  73.35526 118.94282  87.54559
```

```
# Extremos del intervalo de confianza para la media condicionada
# El cuantil usado es el mismo que en los I.C. de beta0
nuevosValores.extrInfMediaCond <- nuevosValores.y0tilde -
        beta0.cuantil * sd.error / sqrt(nuevosValores.n0)
nuevosValores.extrSupMediaCond <- nuevosValores.y0tilde +
        beta0.cuantil * sd.error / sqrt(nuevosValores.n0)
intervalos.media <- cbind(nuevosValores.extrInfMediaCond,
                          nuevosValores.extrSupMediaCond)
# Damos nombres a las columnas y filas de la tabla
colnames(intervalos.media) <- c("Extremo inferior", "Extremo superior")
rownames(intervalos.media) <- nuevosValores
# Intervalos para la media condicionada
intervalos.media
```

```
##   Extremo inferior Extremo superior
## 2        1.8459908        2.4311389
## 4       -0.1369772        0.3225513
## 6       -2.2208054       -1.6851761
```

o también podemos obtenerlos automáticamente utilizando la función `predict` sobre el modelo de R pasándole el argumento `interval = "confidence"`:

```
predict(modelo, newdata = nuevosDatos, interval = "confidence", level = nivel)
```

```
##           fit        lwr        upr
## 1  2.13856482  1.8459908  2.4311389
## 2  0.09278705 -0.1369772  0.3225513
## 3 -1.95299072 -2.2208054 -1.6851761
```

Los extremos inferiores se encuentran en la columna `lwr`, y los superiores, en `upr`, de forma que 2 tiene un intervalo asociado $(2.1371613, 2.1399684)$; 4, $(0.0916848, 0.0938893)$; y 6, $(-1.9542755, -1.9517059)$. Con los datos de la muestra, los valores de la media condicionada al añadir cada uno de los nuevos valores se encontrarán en el intervalo respectivo con un 99% de probabilidad.

Obtenemos ahora los intervalos de predicción, los cuáles serán más amplios que los anteriores dado que estamos considerando una variable aleatoria, Y_0 que es más difícil de estimar que un parámetro, $\mathbb{E}[Y|X = x_0]$. Además, estos intervalos de confianza no dependen de n, a diferencia de los primeros, de forma que la precisión de la estimación no aumentaría aunque incrementásemos el tamaño de la muestra. Aun así, ambos estarán centrados en el mismo punto, el valor correspondiente en la columna `fit`, que es el valor del estadístico asociado a la construcción del intervalo por el método pivotal.

En este caso, plantearemos en cómo se aproxima $\tilde{Y_0}$ a la nueva observación $Y = \beta_0 + \beta_1 x_0 + \epsilon$. Aquí tenemos que $\mathbb{E}[\tilde{Y_0}] = \mathbb{E}[Y_0]$ y $Var(\tilde{Y_0} - Y_0) = \sigma^2(1 + \frac{1}{n_0})$, de dónde tenemos el intervalo de confianza para la nueva observación Y_0:

$$(\tilde{Y_0} - t_{n-2,\alpha/2}\,\hat{\sigma}\sqrt{1 + \frac{1}{n_0}}, \tilde{Y_0} + t_{n-2,\alpha/2}\,\hat{\sigma}\sqrt{1 + \frac{1}{n_0}})$$

calculamos entonces estos intervalos para cada nuevo dato:

```
# Extremos del intervalo de confianza para la predicción
nuevosValores.extrInfPred <- nuevosValores.y0tilde -
        beta0.cuantil * sd.error * sqrt(1 + 1 / nuevosValores.n0)
nuevosValores.extrSupPred <- nuevosValores.y0tilde +
        beta0.cuantil * sd.error * sqrt(1 + 1 / nuevosValores.n0)
intervalos.prediccion <- cbind(nuevosValores.extrInfPred,
                        nuevosValores.extrSupPred)
# Damos nombres a las columnas y filas de la tabla
colnames(intervalos.prediccion) <- c("Extremo inferior", "Extremo superior")
rownames(intervalos.prediccion) <- nuevosValores
# Intervalos para la predicción de nuevos valores
intervalos.prediccion
```

```
##   Extremo inferior Extremo superior
## 2       -0.3842867        4.6614163
## 4       -2.4235539        2.6091280
## 6       -4.4730909        0.5671095
```

y para hallarlos automáticamente, pasamos esta vez el argumento `interval = "prediction"`:

```
predict(modelo, newdata = nuevosDatos, interval = "prediction", level = nivel)
```

```
##           fit        lwr       upr
## 1  2.13856482 -0.3842867 4.6614163
## 2  0.09278705 -2.4235539 2.6091280
## 3 -1.95299072 -4.4730909 0.5671095
```

Obtenemos los intervalos 2, $(2.12646188, 2.1506678)$; 4, $(0.08071534, 0.1048588)$; y 6, $(-1.96508046, -1.9409010)$. Con los datos de la muestra, los valores tomados por la variable respuesta al añadir cada uno de los nuevos valores de la variable explicativa se encontrarán en el intervalo respectivo con un 99% de probabilidad.

Los valores estimados de la media condicional y de los valores de la variable Y, que se pueden comprobar en la columna `fit`, coinciden, puesto que su cálculo es el mismo:

```
nuevosValores.y0tilde
```

```
## [1]  2.13856482  0.09278705 -1.95299072
```

6) Cálculo de una medida de bondad de ajuste del modelo lineal simple considerado

El objetivo del cálculo de una medida de bondad de ajuste es determinar como de "bueno" o "potente" es un modelo de regresión. En terminos estadísticos, esto se traduce a cuánta proporción de la variabilidad de Y puede ser explicada por el modelo de regresión. Una de las medidas principales para lograr este objetivo es el *coeficiente de determinación* R^2. En el caso del modelo lineal, podemos descomponer la varianza como

$$\sigma_Y^2 = \sum_{i=1}^{n}(Y_i - \overline{Y})^2 = \sum_{i=1}^{n}(Y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i)^2 + \sum_{i=1}^{n}(\hat{\beta}_0 + \hat{\beta}_1 x_i - \overline{Y})^2$$

Observamos que el segundo sumando representa la varianza explicada por la recta de regresión, mientras que el primero representa la no explicada. Es por tanto necesario que el cociente $\frac{RSS}{TSS}$, donde TSS (*Total Sum of Squares*) es la varianza total de Y y RSS (*Residual Sum of Squares*) es la no explicada por el modelo, sea lo menor posible.

Definimos entonces el coeficiente de determinación $R^2 = 1 - \frac{RSS}{TSS}$. En primer lugar, calcularemos este cociente a mano:

```
rss <- sum((Y - beta0.gorro - beta1.gorro * X)^2)
rss
```

```
## [1] 108.0937
```

```
tss <- var(Y)
r2 <- 1 - rss / tss
r2
```

```
## [1] -10.44502
```

Aunque también podemos obtenerla a partir del modelo de R, en el campo *Multiple R-Squared* del `summary`:

```
summary(modelo)
```

```
##
## Call:
## lm(formula = Y ~ X)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.76465 -0.72493  0.00685  0.71260  2.20924
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.18434    0.15755   26.56   <2e-16 ***
## X           -1.02289    0.03072  -33.30   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9571 on 118 degrees of freedom
## Multiple R-squared:  0.9038, Adjusted R-squared:  0.903
## F-statistic:  1109 on 1 and 118 DF,  p-value: < 2.2e-16
```

Obtenemos un valor de 0.9038, lo cual nos indica que nuestro modelo es bastante potente en términos habituales. Cabe destacar que el hecho de que el coeficiente de determinación sea alto, no nos indica aún que nuestro modelo sea correcto, lo cual será discutido en el apartado 7. El recíproco tampoco es cierto, podríamos tener un modelo correcto que no fuera bueno. Intuitivamente, este valor alto se contrasta con el hecho de que en el diagrama de dispersión los datos sean cercanos a la recta de regresión.

7) Validación del modelo de regresión lineal simple. Estudio de linealidad, homocedasticidad, normalidad e independencia.

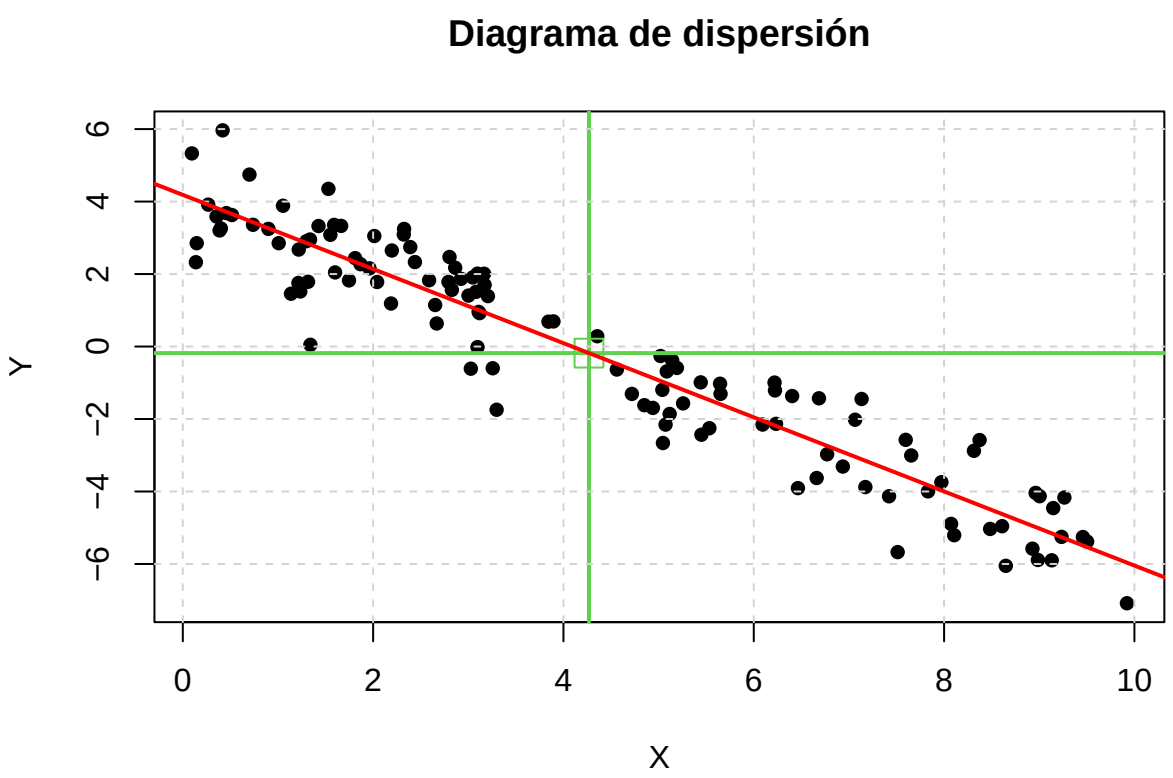
Las técnicas de inferencia empleadas hasta el momento son ciertas bajo el supuesto de que las 4 hipótesis del modelo de regresión lineal simple (linealidad, homocedasticidad, normalidad e independencia) se verifican. De lo contrario, no todas las interpretaciones obtenidas seguirían siendo válidas. Por ejemplo, si no se cumplieran las hipótesis de homocedasticidad, normalidad e independencia, los intervalos de confianza que hemos obtenido no serían válidos.

Bastaría que fallara una de las cuatro hipótesis para afirmar que el modelo de regresión lineal simple no es un modelo correcto para la muestra de datos.

Linealidad

En primer lugar, podemos tratar de aventurar si se los datos siguen una tendencia lineal. Emplearemos una aproximación exploratoria, a través de una interpretación gráfica. Para ello, revisitemos la representación previamente definida.

```
representar()
abline(modelo, col = "red", lwd = 2)
```



Relación entre la variable explicativa y la variable respuesta

Vemos que los puntos parecen distribuirse en torno a la recta de forma lineal. Si bien hay datos un tanto atípicos, especialmente en los extremos, esto no es lo suficientemente significativo como para rechazar la hipótesis. Tampoco se ve un patrón evidente en los datos (es esto lo que debemos tratar de detectar, y no solo corroborar que haya el mismo número de puntos por encima/debajo de la recta, que no es suficiente como para indicar linealidad).

Nótese que aunque se puede apreciar una menor concentración de puntos para valores de X comprendidos alrededor del valor 4, esto no es indicativo de una falta de linealidad. Dado que trabajamos bajo diseño fijo, se tiene que achacar a decisiones sobre las condiciones de medición o al propio diseño del experimento. Esta observación se puede comprobar a través del siguiente cuadro:

```
# Representamos el número de valores de X en cada intervalo de longitud 0.5,
# comenzando desde el mayor entero menor o igual que el dato mínimo,
# y finalizando en el menor entero mayor o igual que el dato máximo.
table(cut(X, breaks = seq(from = floor(min(X)),
       to = ceiling(max(X)), by = 0.5)))
```

```
##
##  (0,0.5]  (0.5,1]  (1,1.5]  (1.5,2]  (2,2.5]  (2.5,3]  (3,3.5]  (3.5,4]
##        9        4       11        9        8        8       13        2
##  (4,4.5]  (4.5,5]  (5,5.5]  (5.5,6]  (6,6.5]  (6.5,7]  (7,7.5]  (7.5,8]
##        1        4       11        3        6        4        4        5
##  (8,8.5]  (8.5,9]  (9,9.5] (9.5,10]
##        5        5        6        2
```

Con el objetivo de realizar una prueba más precisa, planteamos el siguiente contraste de hipótesis. Como hipótesis nula tenemos que la variable respuesta siga el modelo lineal simple que hemos estado considerando,

y como hipótesis nula, que siga un modelo parabólico, donde hay dependencia de la variable explicativa al cuadrado:

$$\begin{cases} H_0 : Y = \beta_0 + \beta_1 X + \epsilon \\ H_a : Y = \beta_0 + \beta_1 X + \beta_2 X^2 \epsilon \end{cases}$$

Ejecutamos la prueba:

```
# Empleamos power = 2 porque estamos considerando una alternativa cuadrática
resettest(modelo, power = 2)
```

```
##
##  RESET test
##
## data:  modelo
## RESET = 0.09269, df1 = 1, df2 = 117, p-value = 0.7613
```

Vemos que el p-valor es de 0.7613. Dado que es superior a nuestro α fijado, que era del 0.01, no existen evidencias estadísticamente significativas a favor de H_a (de hecho, el p-valor es superior a cualquiera de los niveles de significación habituales: 1%, 5% y 10%). Es decir, no tenemos pruebas en contra de H_0. Podemos asumir que la hipótesis nula es cierta: la variable respuesta se ajusta mejor a un modelo de regresión lineal simple que a uno cuadrático.

No obstante, este contraste solo nos ha aportado información sobre la equiparación con un modelo cuadrático. Podríamos plantearlo también para un modelo cúbico:

$$\begin{cases} H_0 : Y = \beta_0 + \beta_1 X + \epsilon \\ H_a : Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \epsilon \end{cases}$$

```
# Empleamos power = 3 porque estamos considerando una alternativa cúbica
resettest(modelo, power = 3)
```

```
##
##  RESET test
##
## data:  modelo
## RESET = 0.11319, df1 = 1, df2 = 117, p-value = 0.7371
```

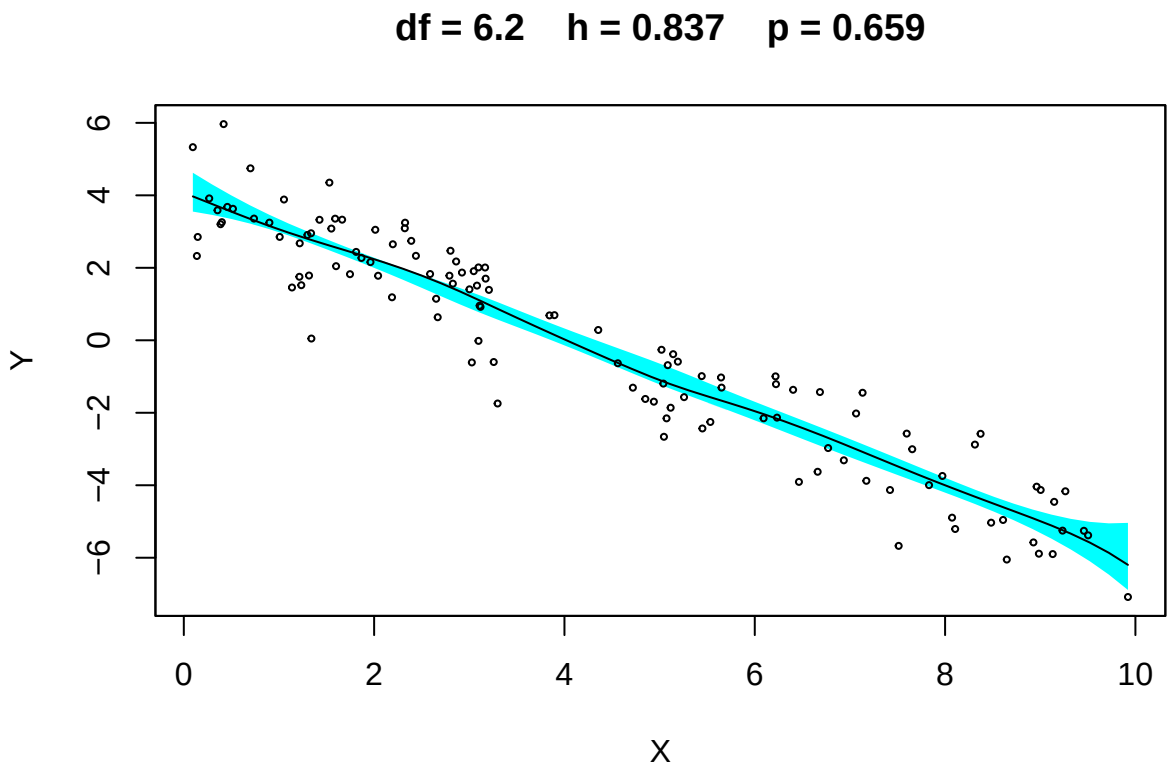
Como el p-valor es también superior a 0.01, de nuevo podemos aceptar la hipótesis nula de que el modelo lineal es válido.

Ahora bien, tendríamos que seguir experimentando para cualquier valores de power para contraponer estos modelos polinómicos, uno a uno, contra el lineal simple, pero ni siquiera en este caso estaríamos considerando todas las opciones de modelos. Dado que una exploración perfecta en este sentido es impracticable experimentalmente, podemos plantearnos en su lugar un contraste más general, con una alternativa no parámetrica:

$$\begin{cases} H_0 : Y = \beta_0 + \beta_1 X + \epsilon \\ H_a : Y = m(X) + \epsilon \end{cases}$$

Haciendo uso del paquete `sm`, realizamos la prueba de hipótesis:

```r
# Importamos rpanel para abrir un panel interactivo para la representación
# Los valores que sabemos interpretar son los que aparecen
# con las opciones por defecto
# Indicamos test=T para que se nos muestre un p-valor.
sm.regression(X, Y, model = "linear", panel = T, test = T)
```



La interpretación de la figura resultante es la siguiente. Con una línea negra nos aparece marcada una estimación no paramétrica de la regresión (sin asumir linealidad), y en azul, una región de confianza para el modelo lineal simple. Vemos que la línea negra se encuentra siempre dentro de la región azul. Por tanto, podemos asumir que la hipótesis nula es cierta, esto es, que los datos verifican la hipótesis de linealidad.

Al indicar test=T, hemos obtenido también un p-valor asociado. Como $0.659 > 0.01$, de nuevo no tenemos evidencias estadísticamente significativas a favor de H_a. Equivalentemente, no tenemos pruebas en contra de H_0. Podemos asumir que la hipótesis nula (que la variable respuesta se ajuste a un modelo de regresión lineal simple) es cierta.

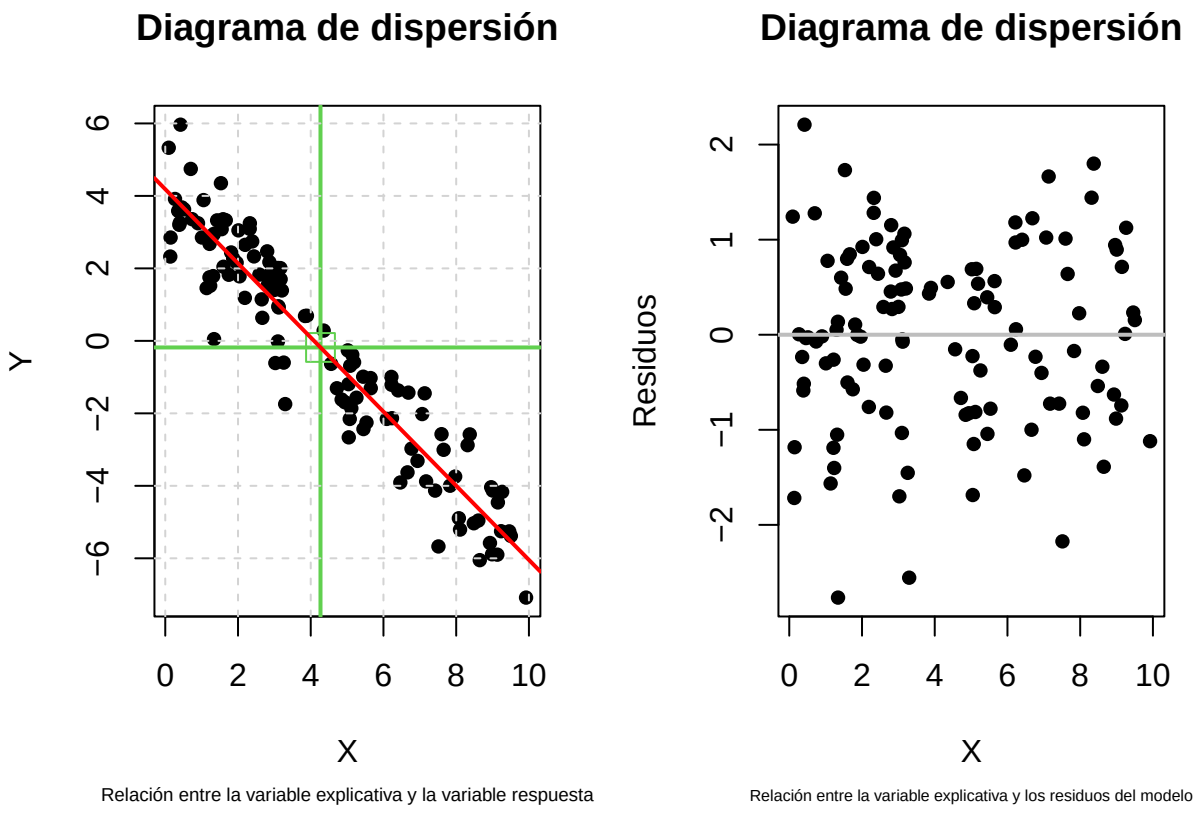
Homocedasticidad

Corroboraremos ahora que la varianza del error es fija e independiente del valor que toma la variable explicativa:

$$Var(\epsilon|X = x) = \sigma^2 \quad \forall x$$

Contrapongamos ahora los residuos del modelo a la variable explicativa. Se muestra también el diagrama de dispersión original:

```r
par(mfrow = c(1, 2))

representar()
abline(modelo, col = "red", lwd = 2)

residuos <- modelo$residuals

plot(X, residuos,
        main = "Diagrama de dispersión", pch = 16,
        sub = "Relación entre la variable explicativa y los residuos del modelo",
        ylab = "Residuos",
        cex.sub = 0.5
)
abline(h = 0, col = "gray", lwd = 2)
```




```r
par(mfrow = c(1, 1))
```

Vemos que la distribución de los residuos en el diagrama no sigue un patrón evidente, y que su desviación con respecto a la recta $x = 0$ parece ser la misma sin importar el intervalo de X considerado.

Tampoco sobre el diagrama de dispersión de la variable respuesta observamos una tendencia significativa acera de las desviaciones con la recta de regresión. En conjunción con lo anterior, podríamos aventurar, a primera vista, que los datos muestrales son verdaderamente homocedásticos.

Sí destacamos que la interpretación para la región central, en aproximadamente $(4, 4.5)$, puede no ser muy precisa, por falta de datos. Sin embargo, esto no basta para desmentir la hipótesis de homocedasticidad.

Para tener una confirmación precisa, nos planteamos el siguiente contraste de hipótesis:

$$\begin{cases} H_0 : \text{modelo homocedástico} \\ H_a : \text{modelo heterocedástico} \end{cases}$$

Ejecutamos un test de Harrison-McCabe con R, haciendo uso del previamente cargado paquete `lmtest`:

```
hmctest(Y ~ X)
```

```
##
##  Harrison-McCabe test
##
## data:  Y ~ X
## HMC = 0.55113, p-value = 0.77
```

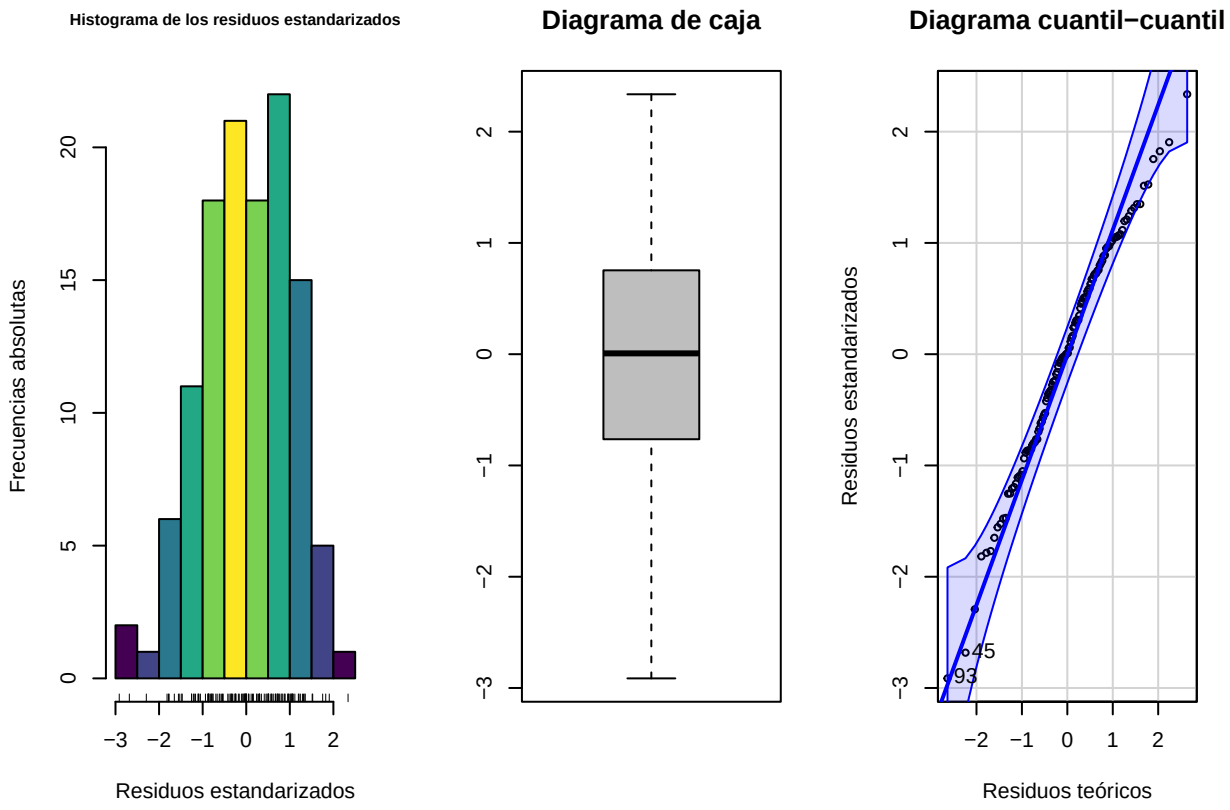
El p-valor es de $0.763 > 0.01 = \alpha$. Por un razonamiento análogo a los anteriores, no existen pruebas estadísticamente significativas para rechazar la hipótesis nula, y podemos asumirla como válida: aceptamos que el modelo es homocedástico.

Normalidad

Para corroborar que el error tiene distribución normal, haremos varias representaciones gráficas que nos permitan intuir si la hipótesis se ajusta a los datos. Trabajaremos con los residuos estandarizados, pues no tienen la misma varianza y la correlación entre cada 2 de ellos puede ser distinta (provienen de distribuciones diferentes).

Presentamos 3 gráficos: un histograma, un boxplot y un qqplot (para el cual necesitamos la librería `car`), aunque centraremos nuestra atención en el último de ellos, el más relevante en lo que concierne al estudio de la normalidad.

```
residuos.estan <- rstandard(modelo)

par(mfrow = c(1, 3))

hist(residuos.estan,
        col = c(viridis(n = 5, begin = 0, end = 0.8),
        viridis(n = 1, begin = 1),
        viridis(n = 5, begin = 0.8, end = 0)),
        main = "Histograma de los residuos estandarizados",
        xlab = "Residuos estandarizados",
        ylab = "Frecuencias absolutas",
        cex.main = 0.75
)
rug(residuos.estan)

boxplot(residuos.estan, col = "gray", main = "Diagrama de caja")

qqPlot(residuos.estan, main = "Diagrama cuantil-cuantil",
       ylab = "Residuos estandarizados", xlab = "Residuos teóricos")
```




```
## [1] 93 45
```

En el histograma podemos apreciar una cierta asimetría hacia la derecha (valores más altos). En el boxplot o diagrama de caja vemos que la media está centrada en el centro de la caja, un buen indicador. No obstante, la cola izquierda es de una longitud ligeramente mayor, lo cual es indicativo de la asimetría mencionada, al estar los datos más concentrados alrededor de valores más altos.

El QQPlot o diagrama cuantil-cuantil nos presenta una comparativa entre los cuantil muestrales de los residuos estandarizados y los cuantiles teóricos de una normal estándar. Si los residuos estandarizados presentaran una distribución normal de media 0 y varianza 1, se situarían alrededor de la recta diagonal resaltada. En nuestro caso, vemos que en la zona central el ajuste es bueno, pero hay una cierta desviación en las colas. Esto es especialmente notorio en la superior, donde los cuantiles muestrales son algo inferiores a los cuantiles teóricos de una normal, que es lógico y coherente con la asimetría indicada anteriormente.

Ahora bien, una representación visual es solamente un apoyo al estudio, y no podemos inferir de ella una conclusión estadísticamente definitoria. De hecho, pequeñas desviaciones con respecto a la hipótesis de linealidad, por ejemplo, podrían tener efecto sobre los gráficos obtenidos. Por ello , emplearemos directamente un test sobre los errores estandarizados con respecto a una distribución normal. Aunque hay varias opciones adecuadas, como el test de Kolmogorov-Smirnov y el test de Lilliefoids, el más ampliamente usado con este propósito es el test de Shapiro-Wilk, especialmente diseñado para contrastes de normalidad:

$$\begin{cases} H_0 : \epsilon \text{ sigue una distribución normal} \\ H_a : \epsilon \text{ no sigue una distribución normal} \end{cases}$$

Ejecutemos pues el contraste de especificación mencionado:

```
shapiro.test(residuos.estan)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuos.estan
## W = 0.98768, p-value = 0.3518
```

También podemos comprobar los resultados de otros tests:

```
# Semilla para la generación de números aleatorios en ks.test
set.seed(as.numeric(Sys.time()))
# Comprobamos si los residuos estandarizados coinciden en distribución
# con una muestra aleatoria N(0,1) de n datos
# El contraste es two-sided porque estamos interesados en comprobar ambas colas
ks.test(residuos.estan, rnorm(n), alternative = "two.sided")
```

```
##
##  Two-sample Kolmogorov-Smirnov test
##
## data:  residuos.estan and rnorm(n)
## D = 0.058333, p-value = 0.9868
## alternative hypothesis: two-sided
```

```
# Realizamos también un Lillie test
lillie.test(residuos.estan)
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  residuos.estan
## D = 0.057751, p-value = 0.4207
```

Todos los p-valores obtenidos mediante estos tests son superiores al α fijado, 0.01. Por consiguiente, no existen pruebas estadísticamente significativas a favor de H_a. No tenemos pruebas en contra de H_0 y, por tanto, podemos asumir que la hipótesis nula es cierta (que los errores tienen distribución normal).

Una observación adicional: en este caso, tenemos que el tamaño de la muestra, n, es mayor que 30, de modo que se pueden despreciar las impurezas debidas a utilizar los residuos en el estudio de la normalidad, en lugar de los errores (que no están sujetos a la aplicación del ajuste de mínimos cuadrados).

```
n
```

```
## [1] 120
```

Independencia

De entre las 4 hipótesis con las que trabaja el modelo, la independencia de los errores es la más difícil de corroborar. No tenemos información acerca del proceso de recogida de muestras, por lo que no podemos garantizarla en base a que los datos hayan sido medidos sobre objetos o individuos de forma independiente.

Debido a la complejidad inherente a este apartado, nos limitaremos a comprobar la independencia temporal. Realizamos un constraste de correlación temporal y, si los errores estén incorrelacionados, dado que ya hemos visto que también siguen una distribución normal, podremos deducir que son independientes.

Asumiremos que nuestros datos han sido medidos a lo largo del tiempo, de forma que tiene sentido realizar el contraste mencionado.

Nos preguntamos entonces si existe algún tipo de relación entre las observaciones, esto es:

$$\begin{cases} H_0 : \epsilon \text{ son incorrelacionados temporalmente} \\ H_a : \epsilon \text{ son correlacionados temporalmente de orden k} \end{cases}$$

En el contraste planteado, $k \in \mathbb{N}$, $k > 1$, es el retardo, esto es, la separación entre los instantes de tiempo que influyen sobre el instante actual. Realizaremos un test *Ljung-Box* con la función `Box.test`, para lo cual tendremos que fijar un k de antemano. Dado que probar todos los valores sería imposible y no tenemos información sobre la toma de datos, nos limitaremos a algunos valores representativos: $k = 1, 2$ y 3. Trabajaremos con el nivel de significación α previamente fijado: 0.01.

```
Box.test(residuos.estan, type = "Ljung-Box", lag = 1)
```

```
##
##  Box-Ljung test
##
## data:  residuos.estan
## X-squared = 0.12281, df = 1, p-value = 0.726
```

```
Box.test(residuos.estan, type = "Ljung-Box", lag = 2)
```

```
##
##  Box-Ljung test
##
## data:  residuos.estan
## X-squared = 0.69821, df = 2, p-value = 0.7053
```

```
Box.test(residuos.estan, type = "Ljung-Box", lag = 3)
```

```
##
##  Box-Ljung test
##
## data:  residuos.estan
## X-squared = 0.70779, df = 3, p-value = 0.8714
```

Vemos que los p-valores respectivos son 0.05949, 0.06122 y 0.08574. Todos ellos son superiores al nivel de significación fijado, de modo que, en base a nuestro criterio, no existen evidencias estadísticamente significativas a favor de H_a. Como no tenemos pruebas en contra de H_0, podemos asumir que es cierta, esto es, que los errores están incorrelacionados temporalmente. Como ya partíamos de que seguían una distribución normal, asumiremos entonces que son temporalmente independientes.

Resultados ejercicio 7

En base a los resultados de los contrastes realizados (bajo los distintos criterios y restricciones impuestas), nuestro modelo cumple las hipótesis de linealidad, homocedasticidad, normalidad e independencia. Es, por tanto, un modelo de regresión lineal correcto, con lo que los podemos aseverar que las interpretaciones que hicimos en anteriores ejercicios son válidas.

Conclusión

En base a todas las medidas y tests realizados en los ejercicios previos, llegamos a la conclusión de que nuestro modelo de regresión lineal simple se ajusta excelentemente a los datos medidos. Esto es debido a que se trata de un modelo tanto potente como válido, y a que fuera la opción indicada por los contrastes que lo equiparaban con modelos alternativos. De esta forma, hallamos una potente herramienta para relacionar las dos variables, explicar los datos muestrales tomados y predecir nuevos valores de Y.