# Introducción a la Regresión Lineal Múltiple

Joaquín Amat Rodrigo j.amatrodrigo@gmail.com

Julio, 2016

## Tabla de contenidos



Versión PDF: https://github.com/JoaquinAmatRodrigo/Estadistica-con-R

# Introducción

*La información aquí presente se ha obtenido principalmente de OpenIntro Statistics, Tutorials by William B. King Coastal Carolina Universit, Introduction to Statistical Learning y Linear Models with R. En los dos últimos se puede encontrar información mucho más detallada sobre la regresión lineal múltiple.*

La regresión lineal múltiple permite generar un modelo lineal en el que el valor de la variable dependiente o respuesta ($Y$) se determina a partir de un conjunto de variables independientes llamadas predictores ($X1$, $X2$, $X3$...). Es una extensión de la regresión lineal simple, por lo que es fundamental comprender esta última. Los modelos de regresión múltiple pueden emplearse para predecir el valor de la variable dependiente o para evaluar la influencia que tienen los predictores sobre ella (esto último se debe que analizar con cautela para no malinterpretar causa-efecto).

Los modelos lineales múltiples siguen la siguiente ecuación:

$$Y_i = (\beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \cdots + \beta_n X_{ni}) + e_i$$

- $\beta_0$: es la ordenada en el origen, el valor de la variable dependiente $Y$ cuando todos los predictores son cero.
- $\beta_i$: es el efecto promedio que tiene el incremento en una unidad de la variable predictora $X_i$ sobre la variable dependiente $Y$, manteniéndose constantes el resto de variables. Se conocen como coeficientes parciales de regresión.
- $e_i$: es el residuo o error, la diferencia entre el valor observado y el estimado por el modelo.

Es importante tener en cuenta que la magnitud de cada coeficiente parcial de correlación depende de las unidades en las que se mida la variable predictora a la que corresponde, por lo que su magnitud no está asociada con la importancia de cada predictor. Para poder determinar qué impacto tienen en el modelo cada una de las variables, se emplean los *coeficientes parciales estandarizados*, que se obtienen al estandarizar (sustraer la media y dividir entre la desviación estándar) las variables predictoras previo ajuste del modelo.

# Condiciones para la regresión lineal múltiple

Los modelos de correlación lineal múltiple requieren de las mismas condiciones que los modelos lineales simples más otras adicionales.

**No colinealidad o multicolinealidad:**

En los modelos lineales múltiples los predictores deben ser independientes, no debe de haber colinealidad entre ellos. La colinealidad ocurre cuando un predictor está linealmente relacionado con uno o varios de los otros predictores del modelo o cuando es la combinación lineal de otros predictores. Como consecuencia de la colinealidad no se puede identificar de forma precisa el efecto individual que tiene cada una de las variables colineales sobre la variable respuesta, lo que se traduce en un incremento de la varianza de los coeficientes de regresión estimados hasta el punto que resulta prácticamente imposible establecer su significancia estadística. Además, pequeños cambios en los datos provocan grandes cambios en las estimaciones de los coeficientes. Si bien la colinealidad propiamente dicha existe solo si el coeficiente de correlación simple o múltiple entre algunas de las variables independientes es 1, esto raramente ocurre en la realidad. Sin embargo, es frecuente encontrar la llamada *casi-colinealidad* o *multicolinealidad no perfecta*.

No existe un método estadístico concreto para determinar la existencia de colinealidad o multicolinealidad entre los predictores de un modelo de regresión, sin embargo, se han desarrollado numerosas reglas prácticas que tratan de determinar en qué medida afecta a la estimación y contraste de un modelo. Los pasos recomendados a seguir son:

- Si el coeficiente de determinación $R^2$ es alto pero ninguno de los predictores resulta significativo, hay indicios de colinealidad.
- Calcular una matriz de correlación en la que se estudia la relación lineal entre cada par de predictores. Es importante tener en cuenta que, a pesar de no obtenerse ningún coeficiente de correlación alto, no está asegurado que no exista multicolinealidad. Se puede dar el caso de tener una relación lineal casi perfecta entre tres o más variables y que las correlaciones simples entre pares de estas mismas variables no sean mayores que 0.5.
- Generar un modelo de regresión lineal simple entre cada uno de los predictores frente al resto. Si en alguno de los modelos el *coeficiente de determinación* $R^2$ es alto, estaría señalando a una posible colinealidad.
- Tolerancia (*TOL*) y Factor de Inflación de la Varianza (*VIF*). Se trata de dos parámetros que vienen a cuantificar lo mismo (uno es el inverso del otro). El *VIF* de cada predictor se calcula según la siguiente fórmula:

$$VIF_{\hat{\beta}_j} = \frac{1}{1 - R^2}$$

$$Tolerancia_{\hat{\beta}_j} = \frac{1}{VIF_{\hat{\beta}_j}}$$

donde $R^2$ se obtiene de la regresión del predictor $X_j$ sobre los otros predictores. Esta es la opción más recomendada, los límites de referencia que se suelen emplear son:

- VIF = 1: Ausencia total de colinealidad
- 1 < VIF < 5: La regresión puede verse afectada por cierta colinealidad.
- 5 < VIF < 10: Causa de preocupación
- El termino tolerancia es $1/VIF$ por lo que los límites recomendables están entre 1 y 0.1.

En caso de encontrar colinealidad entre predictores, hay dos posibles soluciones. La primera es excluir uno de los predictores problemáticos intentando conservar el que, a juicio del investigador, está influyendo realmente en la variable respuesta. Esta medida no suele tener mucho impacto en el modelo en cuanto a su capacidad predictiva ya que, al existir colinealidad, la información que aporta uno de los predictores es redundante en presencia del otro. La segunda opción consiste en combinar las variables colineales en un único predictor, aunque con el riesgo de perder su interpretación.

Cuando se intenta establecer relaciones causa-efecto, la colinealidad puede llevar a conclusiones muy erróneas, haciendo creer que una variable es la causa cuando en realidad es otra la que está influenciando sobre ese predictor.

**Parsimonia:**

Este término hace referencia a que el mejor modelo es aquel capaz de explicar con mayor precisión la variabilidad observada en la variable respuesta empleando el menor número de predictores, por lo tanto, con menos asunciones.

**Relación lineal entre los predictores numéricos y la variable respuesta:**

Cada predictor numérico tiene que estar linealmente relacionado con la variable respuesta $Y$ mientras los demás predictores se mantienen constantes, de lo contrario no se puede introducir en el modelo. La forma más recomendable de comprobarlo es representando los residuos del modelo frente a cada uno de los predictores. Si la relación es lineal, los residuos se distribuyen de forma aleatoria entorno a cero. Estos análisis son solo aproximados, ya que no hay forma de saber si realmente la relación es lineal cuando el resto de predictores se mantienen constantes.

**Distribución normal de los residuos:**

Los residuos se deben distribuir de forma normal con media cero. Para comprobarlo se recurre a histogramas, a los cuantiles normales o a test de hipótesis de normalidad.

**Variabilidad constante de los residuos (homocedasticidad):**

La varianza de los residuos debe de ser constante en todo el rango de observaciones. Para comprobarlo se representan los residuos. Si la varianza es constante, se distribuyen de forma aleatoria manteniendo una misma dispersión y sin ningún patrón específico. Una distribución cónica es un claro identificador de falta de homocedasticidad. También se puede recurrir a contrastes de homocedasticidad como el test de *Breusch-Pagan*.

**No autocorrelación (Independencia):**

Los valores de cada observación son independientes de los otros, esto es especialmente importante de comprobar cuando se trabaja con mediciones temporales. Se recomienda representar los residuos ordenados acorde al tiempo de registro de las observaciones, si existe un cierto patrón hay indicios de autocorrelación. También se puede emplear el test de hipótesis de Durbin-Watson.

**Valores atípicos, con alto leverage o influyentes:**

Es importante identificar observaciones que sean atípicas o que puedan estar influenciando al modelo. La forma más fácil de detectarlas es a través de los residuos, tal como se explica en el capítulo de Regresión Lineal Simple.

**Tamaño de la muestra:**

No se trata de una condición de por sí pero, si no se dispone de suficientes observaciones, predictores que no son realmente influyentes podrían parecerlo. En el libro *Hanbook of biological statistics* recomiendan que el número de observaciones sea como mínimo entre 10 y 20 veces el número de predictores del modelo.

La gran mayoría de condiciones se verifican utilizando los residuos, por lo tanto, se suele generar primero el modelo y posteriormente validar las condiciones. De hecho, el ajuste de un modelo debe verse como un proceso iterativo en el que se ajusta el modelo, se evalúan sus residuos y se mejora. Así hasta llegar a un modelo óptimo.

# Selección de predictores para generar el mejor modelo

La evaluación de un modelo de regresión múltiple así como la elección de qué predictores se deben de incluir en el modelo es uno de los pasos más importantes en la modelización estadística. En los siguientes apartados se introduce este tema de forma muy simplificada. Para un desarrollo más detallado ver capítulo dedicado a la elección de predictores: Selección de predictores y mejor modelo lineal múltiple: subset selection, ridge regression, lasso regression y dimension reduction.

## Evaluación del modelo en conjunto

Al igual que ocurre en los modelos lineales simples, $R^2$ (coeficiente de determinación) es un cuantificador de la bondad de ajuste del modelo. Se define como el porcentaje de varianza de la variable $Y$ que se explica mediante el modelo respecto al total de variabilidad. Por lo tanto, permite cuantificar como de bueno es el modelo para predecir el valor de las observaciones.

En los modelos lineales múltiples, cuantos más predictores se incluyan en el modelo mayor es el valor de $R^2$, ya que, por poco que sea, cada predictor va a explicar una parte de la variabilidad observada en $Y$. Es por esto que $R^2$ no puede utilizarse para comparar modelos con distinto número de predictores.

$R^2_{ajustado}$ introduce una penalización al valor de $R^2$ por cada predictor que se introduce en el modelo. El valor de la penalización depende del número de predictores utilizados y del tamaño de la muestra, es decir, del número de grados de libertad. Cuanto mayor es el tamaño de la muestra, más predictores se pueden incorporar en el modelo. $R^2_{ajustado}$ permite encontrar el mejor modelo, aquel que consigue explicar mejor la variabilidad de $Y$ con el menor número de predictores. Si bien es un método para evaluar la bondad de ajuste muy utilizado, hay otros.

$$R^2_{ajustado} = 1 - \frac{SSE}{SST} x \frac{n-1}{n-k-1} = R^2 - (1-R^2)\frac{n-1}{n-k-1} = 1 - \frac{SSE/df_e}{SST/df_t}$$

siendo $SSE$ la variabilidad explicada por el modelo (*Sum of Squares Explained*), $SST$ la variabilidad total de $Y$ (*Sum of Squares Total*), $n$ el tamaño de la muestra y $k$ el número de predictores introducidos en el modelo.

Para conocer la variabilidad que explica cada uno de los predictores incorporadas en el modelo se recurre a un ANOVA, ya que es el método que se encarga de analizar la varianza.

Tal y como ocurre en los modelos lineales simples o en los estudios de correlación, por muy alta que sea la bondad de ajuste, si el test F no resulta significativo no se puede aceptar el modelo como válido puesto que no es capaz de explicar la varianza observada mejor de lo esperado por azar.

## Selección de los predictores

A la hora de seleccionar los predictores que deben formar parte del modelo se pueden seguir varios métodos:

**Método jerárquico:** basándose en el criterio del analista, se introducen unos predictores determinados en un orden determinado.

**Método de entrada forzada:** se introducen todos los predictores simultáneamente.

**Método paso a paso (*stepwise*):** emplea criterios matemáticos para decidir qué predictores contribuyen significativamente al modelo y en qué orden se introducen. Dentro de este método se diferencias tres estrategias:

- Dirección *forward*: El modelo inicial no contiene ningún predictor, solo el parámetro $\beta_0$. A partir de este se generan todos los posibles modelos introduciendo una sola variable de entre las disponibles. Aquella variable que mejore en mayor medida el modelo se selecciona. A continuación se intenta incrementar el modelo probando a introducir una a una las variables restantes. Si introduciendo alguna de ellas mejora, también se selecciona. En el caso de que varias lo hagan, se selecciona la que incremente en mayor medida la capacidad del modelo. Este proceso se repite hasta llegar al punto en el que ninguna de las variables que quedan por incorporar mejore el modelo.

- Dirección *backward*: El modelo se inicia con todas las variables disponibles incluidas como predictores. Se prueba a eliminar una a una cada variable, si se mejora el modelo, queda excluida. Este método permite evaluar cada variable en presencia de las otras.

- Doble o mixto: Se trata de una combinación de la selección *forward* y *backward*. Se inicia igual que el *forward* pero tras cada nueva incorporación se realiza un test de extracción de predictores no útiles como en el *backward*. Presenta la ventaja de que si a medida que se añaden predictores, alguno de los ya presentes deja de contribuir al modelo, se elimina.

El método paso a paso requiere de algún criterio matemático para determinar si el modelo mejora o empeora con cada incorporación o extracción. Existen varios parámetros empelados, de entre los que destacan el $C_p$ , AIC, BIC y *$R^2$-ajustado*, cada uno de ellos con ventajas e inconvenientes. El método *Akaike(AIC)* tiende a ser más restrictivo e introducir menos predictores que el *$R^2$-ajustado*. Para un mismo set de datos, no todos los métodos tienen porque concluir en un mismo modelo.

Es frecuente encontrar ejemplos en los que la selección de los predictores se basa en el *p-value* asociado a cada uno. Si bien este método es sencillo e intuitivo, presenta múltiples inconvenientes: la inflación del error tipo I debida a las múltiples comparaciones, la eliminación de los predictores menos significativos tiende a incrementar la significancia de los otros predictores … Por esta razón, a excepción de casos muy sencillos con pocos predictores, es preferible no emplear los *p-values* como criterio de selección.

En el caso de variables categóricas, si al menos uno de sus niveles es significativo, se considera que la variable es significativa. Cabe mencionar que, si una variable se excluye del modelo como predictor, significa que no aporta información adicional al modelo, pero sí puede estar relacionada con la variable respuesta.

En *R* la función `step()` permite encontrar el mejor modelo basado en AIC utilizando cualquiera de las 3 variantes del método paso a paso.

## Variables nominales/categóricas como predictores

Cuando se introduce una variable categórica como predictor, un nivel se considera el de referencia (normalmente codificado como 0) y el resto de niveles se comparan con él. En el caso de que el predictor categórico tenga más de dos niveles, se generan lo que se conoce como variables *dummy*, que son variables creadas para cada uno de los niveles del predictor categórico y que pueden tomar el valor de 0 o 1. Cada vez que se emplee el modelo para predecir un valor, solo una variable *dummy* por predictor adquiere el valor 1 (la que coincida con el valor que adquiere el predictor en ese caso) mientras que el resto se consideran 0. El valor del coeficiente parcial de correlación $\beta_i$ de cada variable *dummy* indica el porcentaje promedio en el que influye dicho nivel sobre la variable dependiente $Y$ en comparación con el nivel de referencia de dicho predictor.

La idea de variables *dummy* se entiende mejor con un ejemplo. Supóngase un modelo en el que la variable respuesta *peso* se predice en función de la *altura* y *nacionalidad* del sujeto. La variable nacionalidad es cualitativa con 3 niveles (americana, europea y asiática). A pesar de que el predictor inicial es *nacionalidad*, se crea una variable nueva por cada nivel, cuyo valor puede ser 0 o 1. De tal forma que la ecuación del modelo completo es:

$$peso = \beta_0 + \beta_1 altura + \beta_2 americana + \beta_3 europea + \beta_4 asiatica$$

Si el sujeto es europeo, las variables dummy *americana* y *asiatica* se consideran 0, de forma que el modelo para este caso se queda en:

$$peso = \beta_0 + \beta_1 altura + \beta_3 europea$$

## Validación cruzada (Cross-Validation)

Una vez seleccionado el mejor modelo que se puede crear con los datos disponibles, se tiene que comprobar su validez prediciendo nuevas observaciones que no se hayan empleado para entrenarlo, de este modo se verifica si el modelo se puede generalizar. La validación cruzada consiste en estudiar la precisión de un modelo a través de diferentes muestras. Una estrategia comúnmente empleada es dividir aleatoriamente los datos en dos grupos (70%-30%), ajustar el modelo con el primer grupo y evaluar la precisión de las predicciones con el segundo. Para una descripción más detallada de la validación cruzada consultar: Validación de modelos de regresión: Cross-validation, OneLeaveOut, Bootstrap

## Identificación de valores atípicos (*outliers*), de alto leverage o influyentes

Independientemente de que el modelo se haya podido aceptar, siempre es conveniente identificar si hay algún posible *outlier*, observación con alto *leverage* o influyente, puesto que podría estar condicionando en gran medida el modelo. La eliminación de este tipo de observaciones debe de analizarse con detalle y dependiendo de la finalidad del modelo. Si el fin es predictivo, un modelo sin *outliers* ni observaciones altamente influyentes suele ser capaz de predecir mejor la mayoría de casos. Sin embargo, es muy importante prestar atención a estos valores ya que, de no tratarse de errores de medida, pueden ser los casos más interesantes. El modo adecuado a proceder cuando se sospecha de algún posible valor atípico o influyente es calcular el modelo de regresión incluyendo y excluyendo dicho valor. En el capítulo Regresión Lineal Simple se describe con detalle cómo realizar el análisis para detectarlos.

## Ejemplo1. Predictores numéricos

*Un estudio quiere generar un modelo que permita predecir la esperanza de vida media de los habitantes de una ciudad en función de diferentes variables. Se dispone de información sobre: habitantes, analfabetismo, ingresos, esperanza de vida, asesinatos, universitarios, heladas, área y densidad poblacional.*

```
# El data set empleado es el state.x77 Para facilitar su interpretación se
# renombra y se modifica
require(dplyr)
datos <- as.data.frame(state.x77)
datos <- rename(habitantes = Population, analfabetismo = Illiteracy,
                ingresos = Income, esp_vida = `Life Exp`, asesinatos = Murder,
                universitarios = `HS Grad`, heladas = Frost, area = Area,
                .data = datos)
datos <- mutate(.data = datos, densidad_pobl = habitantes * 1000/area)
```

### 1.Analizar la relación entre variables

El primer paso a la hora de establecer un modelo lineal múltiple es estudiar la relación que existe entre variables. Esta información es crítica a la hora de identificar cuáles pueden ser los mejores predictores para el modelo, qué variables presentan relaciones de tipo no lineal (por lo que no pueden ser incluidas) y para identificar colinealidad entre predictores. A modo complementario, es recomendable representar la distribución de cada variable mediante histogramas.

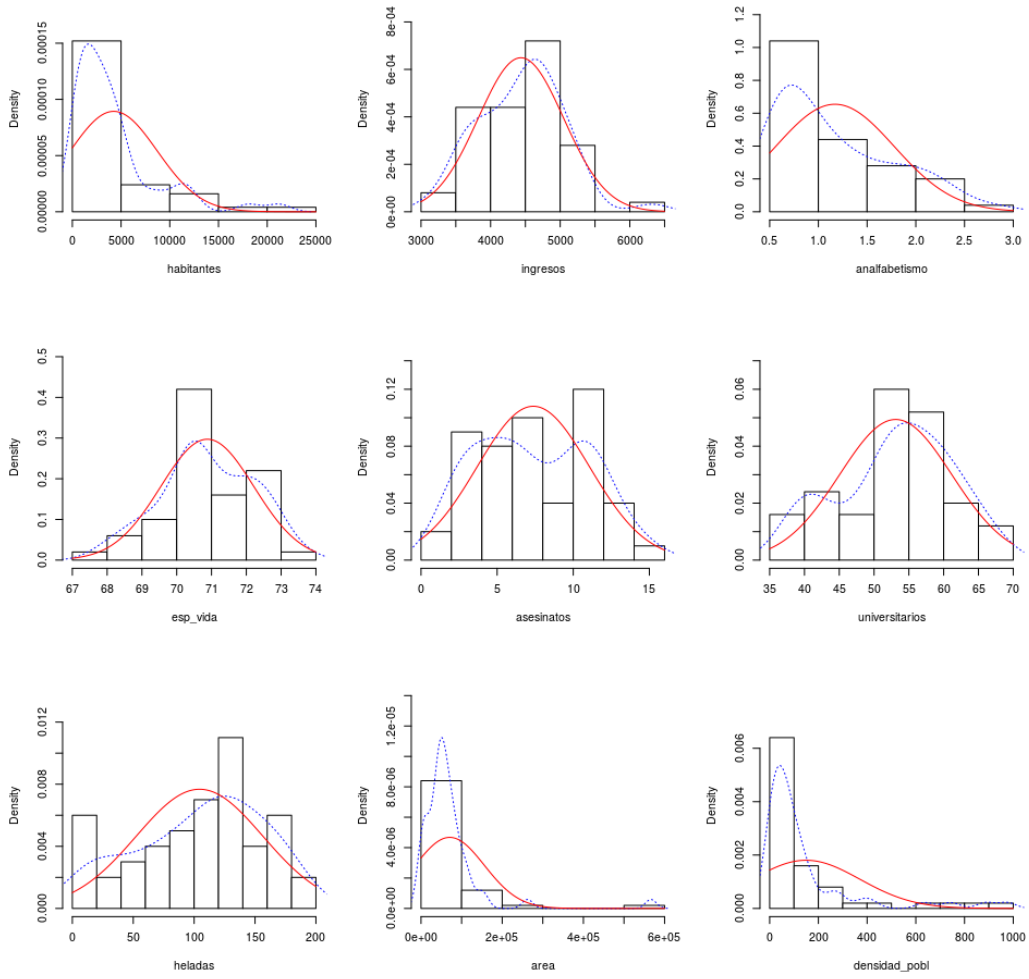
Las dos formas principales de hacerlo son mediante representaciones gráficas (gráficos de dispersión) y el cálculo del coeficiente de correlación de cada par de variables.

```
round(cor(x = datos, method = "pearson"), 3)
```

```
##                habitantes ingresos analfabetismo esp_vida asesinatos
## habitantes          1.000    0.208         0.108   -0.068      0.344
## ingresos            0.208    1.000        -0.437    0.340     -0.230
## analfabetismo       0.108   -0.437         1.000   -0.588      0.703
## esp_vida           -0.068    0.340        -0.588    1.000     -0.781
## asesinatos          0.344   -0.230         0.703   -0.781      1.000
## universitarios     -0.098    0.620        -0.657    0.582     -0.488
## heladas            -0.332    0.226        -0.672    0.262     -0.539
## area                0.023    0.363         0.077   -0.107      0.228
## densidad_pobl       0.246    0.330         0.009    0.091     -0.185
##                universitarios heladas   area densidad_pobl
## habitantes             -0.098  -0.332  0.023         0.246
## ingresos                0.620   0.226  0.363         0.330
## analfabetismo          -0.657  -0.672  0.077         0.009
## esp_vida                0.582   0.262 -0.107         0.091
## asesinatos             -0.488  -0.539  0.228        -0.185
## universitarios          1.000   0.367  0.334        -0.088
## heladas                 0.367   1.000  0.059         0.002
```

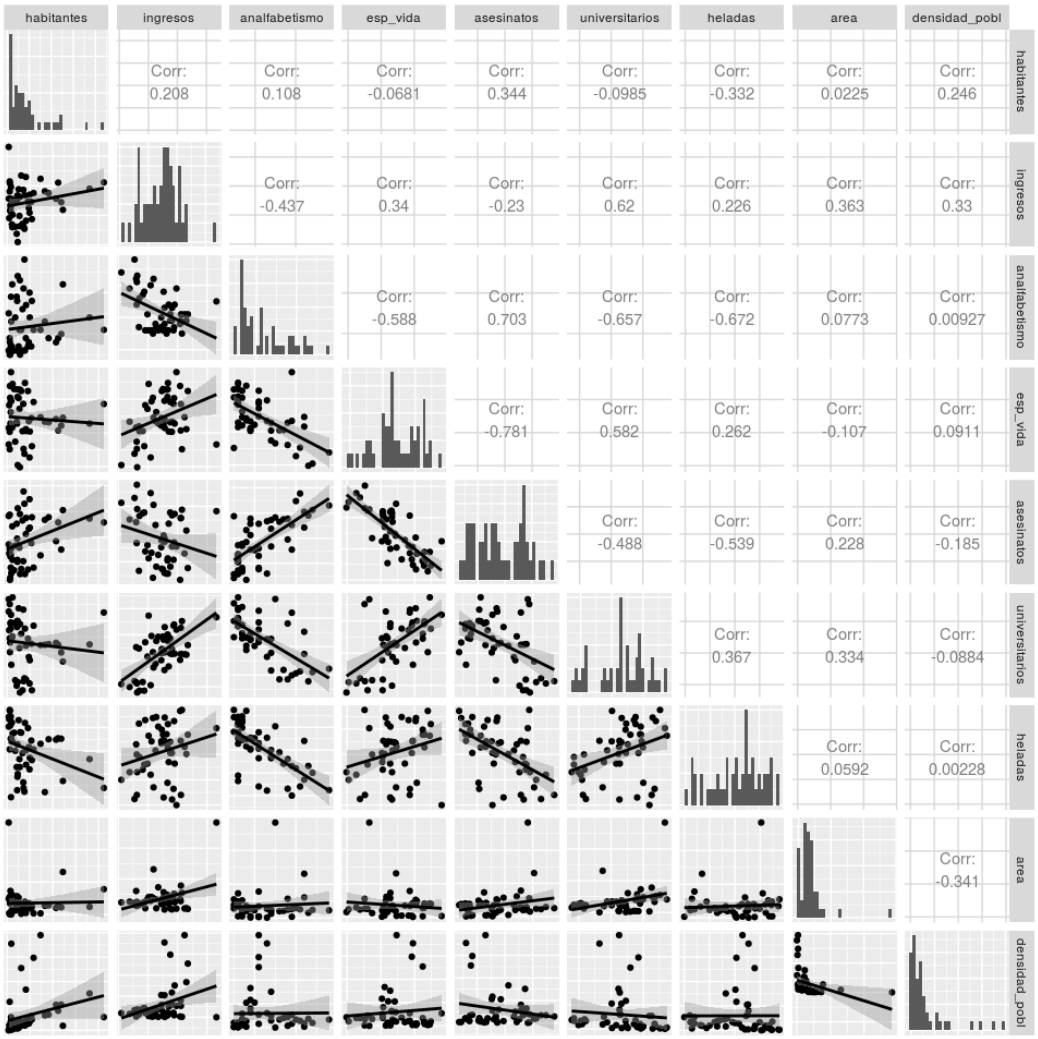
```
## area                    0.334   0.059  1.000        -0.341
## densidad_pobl          -0.088   0.002 -0.341         1.000
```

```
require(psych)
multi.hist(x = datos, dcol = c("blue", "red"), dlty = c("dotted", "solid"),
           main = "")
```




Otros paquetes permiten representar a la vez los diagramas de dispersión, los valores de correlación para cada par de variables y la distribución de cada una de las variables.

```
require(GGally)
ggpairs(datos, lower = list(continuous = "smooth"),
        diag = list(continuous = "bar"), axisLabels = "none")
```




Del análisis preliminar se pueden extraer las siguientes conclusiones:

- Las variables que tienen una mayor relación lineal con la esperanza de vida son: asesinatos (r= -0.78), analfabetismo (r= -0.59) y universitarios (r= 0.58).
- Asesinatos y analfabetismo están medianamente correlacionados (r = 0.7) por lo que posiblemente no sea útil introducir ambos predictores en el modelo.
- Las variables habitantes, área y densidad poblacional muestran una distribución exponencial, una transformación logarítmica posiblemente haría más normal su distribución.

## 2.Generar el modelo

Como se ha explicado en la introducción, hay diferentes formas de llegar al modelo final más adecuado. En este caso se va a emplear el método *mixto* iniciando el modelo con todas las variables como predictores y realizando la selección de los mejores predictores con la medición *Akaike(AIC)*.

```
modelo <- lm(esp_vida ~ habitantes + ingresos + analfabetismo + asesinatos +
             universitarios + heladas + area + densidad_pobl, data = datos)
summary(modelo)
```

```
##
## Call:
## lm(formula = esp_vida ~ habitantes + ingresos + analfabetismo +
##     asesinatos + universitarios + heladas + area + densidad_pobl,
##     data = datos)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.47514 -0.45887 -0.06352  0.59362  1.21823
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     6.995e+01  1.843e+00  37.956  < 2e-16 ***
## habitantes      6.480e-05  3.001e-05   2.159   0.0367 *
## ingresos        2.701e-04  3.087e-04   0.875   0.3867
## analfabetismo   3.029e-01  4.024e-01   0.753   0.4559
## asesinatos     -3.286e-01  4.941e-02  -6.652 5.12e-08 ***
## universitarios  4.291e-02  2.332e-02   1.840   0.0730 .
## heladas        -4.580e-03  3.189e-03  -1.436   0.1585
## area           -1.558e-06  1.914e-06  -0.814   0.4205
## densidad_pobl  -1.105e-03  7.312e-04  -1.511   0.1385
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7337 on 41 degrees of freedom
## Multiple R-squared:  0.7501, Adjusted R-squared:  0.7013
## F-statistic: 15.38 on 8 and 41 DF,  p-value: 3.787e-10
```

El modelo con todas las variables introducidas como predictores tiene un $R^2$ alta (0.7501), es capaz de explicar el 75,01% de la variabilidad observada en la esperanza de vida. El *p-value* del modelo es significativo (3.787e-10) por lo que se puede aceptar que el modelo no es por azar, al menos uno de los coeficientes parciales es distinto de 0. Muchos de ellos no son significativos, lo que es un indicativo de que podrían no contribuir al modelo.

## 3.Selección de los mejores predictores

En este caso se van a emplear la estrategia de *stepwise mixto*. El valor matemático empleado para determinar la calidad del modelo va a ser *Akaike(AIC)*.

```
step(object = modelo, direction = "both", trace = 1)
```

```
## Start:  AIC=-22.89
## esp_vida ~ habitantes + ingresos + analfabetismo + asesinatos +
##     universitarios + heladas + area + densidad_pobl
##
##                   Df Sum of Sq    RSS     AIC
## - analfabetismo    1    0.3050 22.373 -24.208
## - area             1    0.3564 22.425 -24.093
## - ingresos         1    0.4120 22.480 -23.969
## <none>                         22.068 -22.894
## - heladas          1    1.1102 23.178 -22.440
## - densidad_pobl    1    1.2288 23.297 -22.185
## - universitarios   1    1.8225 23.891 -20.926
## - habitantes       1    2.5095 24.578 -19.509
## - asesinatos       1   23.8173 45.886  11.707
##
## Step:  AIC=-24.21
## esp_vida ~ habitantes + ingresos + asesinatos + universitarios +
##     heladas + area + densidad_pobl
##
##                   Df Sum of Sq    RSS     AIC
## - area             1    0.1427 22.516 -25.890
## - ingresos         1    0.2316 22.605 -25.693
## <none>                         22.373 -24.208
## - densidad_pobl    1    0.9286 23.302 -24.174
## - universitarios   1    1.5218 23.895 -22.918
## + analfabetismo    1    0.3050 22.068 -22.894
## - habitantes       1    2.2047 24.578 -21.509
## - heladas          1    3.1324 25.506 -19.656
## - asesinatos       1   26.7071 49.080  13.072
##
## Step:  AIC=-25.89
## esp_vida ~ habitantes + ingresos + asesinatos + universitarios +
##     heladas + densidad_pobl
##
##                   Df Sum of Sq    RSS     AIC
## - ingresos         1     0.132 22.648 -27.598
## - densidad_pobl    1     0.786 23.302 -26.174
## <none>                         22.516 -25.890
## - universitarios   1     1.424 23.940 -24.824
## + area             1     0.143 22.373 -24.208
## + analfabetismo    1     0.091 22.425 -24.093
## - habitantes       1     2.332 24.848 -22.962
## - heladas          1     3.304 25.820 -21.043
```

```
## - asesinatos      1    32.779 55.295  17.033
##
## Step:  AIC=-27.6
## esp_vida ~ habitantes + asesinatos + universitarios + heladas +
##     densidad_pobl
##
##                   Df Sum of Sq    RSS     AIC
## - densidad_pobl    1     0.660 23.308 -28.161
## <none>                         22.648 -27.598
## + ingresos         1     0.132 22.516 -25.890
## + analfabetismo    1     0.061 22.587 -25.732
## + area             1     0.043 22.605 -25.693
## - habitantes       1     2.659 25.307 -24.046
## - heladas          1     3.179 25.827 -23.030
## - universitarios   1     3.966 26.614 -21.529
## - asesinatos       1    33.626 56.274  15.910
##
## Step:  AIC=-28.16
## esp_vida ~ habitantes + asesinatos + universitarios + heladas
##
##                   Df Sum of Sq    RSS     AIC
## <none>                         23.308 -28.161
## + densidad_pobl    1     0.660 22.648 -27.598
## + ingresos         1     0.006 23.302 -26.174
## + analfabetismo    1     0.004 23.304 -26.170
## + area             1     0.001 23.307 -26.163
## - habitantes       1     2.064 25.372 -25.920
## - heladas          1     3.122 26.430 -23.877
## - universitarios   1     5.112 28.420 -20.246
## - asesinatos       1    34.816 58.124  15.528

##
## Call:
## lm(formula = esp_vida ~ habitantes + asesinatos + universitarios +
##     heladas, data = datos)
##
## Coefficients:
##    (Intercept)      habitantes      asesinatos  universitarios
##      7.103e+01       5.014e-05      -3.001e-01       4.658e-02
##        heladas
##     -5.943e-03
```

El mejor modelo resultante del proceso de selección ha sido:

$$esp\_vida \sim habitantes + asesinatos + universitarios + heladas$$

```
modelo <- (lm(formula = esp_vida ~ habitantes + asesinatos + universitarios +
              heladas, data = datos))
summary(modelo)
```

```
##
## Call:
## lm(formula = esp_vida ~ habitantes + asesinatos + universitarios +
##     heladas, data = datos)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.47095 -0.53464 -0.03701  0.57621  1.50683
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     7.103e+01  9.529e-01  74.542  < 2e-16 ***
## habitantes      5.014e-05  2.512e-05   1.996  0.05201 .
## asesinatos     -3.001e-01  3.661e-02  -8.199 1.77e-10 ***
## universitarios  4.658e-02  1.483e-02   3.142  0.00297 **
## heladas        -5.943e-03  2.421e-03  -2.455  0.01802 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7197 on 45 degrees of freedom
## Multiple R-squared:  0.736,  Adjusted R-squared:  0.7126
## F-statistic: 31.37 on 4 and 45 DF,  p-value: 1.696e-12
```

Es recomendable mostrar el intervalo de confianza para cada uno de los coeficientes parciales de correlación:

```
confint(lm(formula = esp_vida ~ habitantes + asesinatos + universitarios + heladas,
    data = datos))
```

```
##                       2.5 %        97.5 %
## (Intercept)     6.910798e+01 72.9462729104
## habitantes     -4.543308e-07  0.0001007343
## asesinatos     -3.738840e-01 -0.2264135705
## universitarios  1.671901e-02  0.0764454870
## heladas        -1.081918e-02 -0.0010673977
```

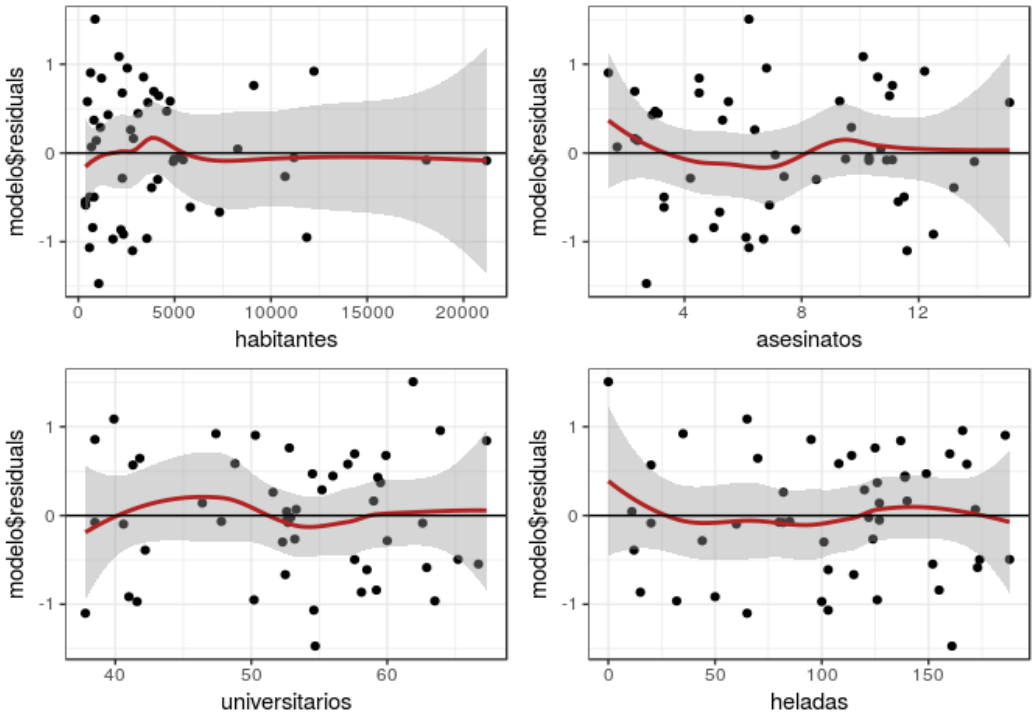
Cada una de las pendientes de un modelo de regresión lineal múltiple (coeficientes parciales de correlación de los predictores) se define del siguiente modo: Si el resto de variables se mantienen constantes, por cada unidad que aumenta el predictor en cuestión, la variable (*Y*) varía en promedio tantas unidades como indica la pendiente. Para este ejemplo, por cada unidad que aumenta el predictor *universitarios*, la esperanza de vida aumenta en promedio 0.04658 unidades, manteniéndose constantes el resto de predictores.

## 4.Validación de condiciones para la regresión múltiple lineal

Relación lineal entre los predictores numéricos y la variable respuesta:

Esta condición se puede validar bien mediante diagramas de dispersión entre la variable dependiente y cada uno de los predictores (como se ha hecho en el análisis preliminar) o con diagramas de dispersión entre cada uno de los predictores y los residuos del modelo. Si la relación es lineal, los residuos deben de distribuirse aleatoriamente en torno a 0 con una variabilidad constante a lo largo del eje X. Esta última opción suele ser más indicada ya que permite identificar posibles datos atípicos.

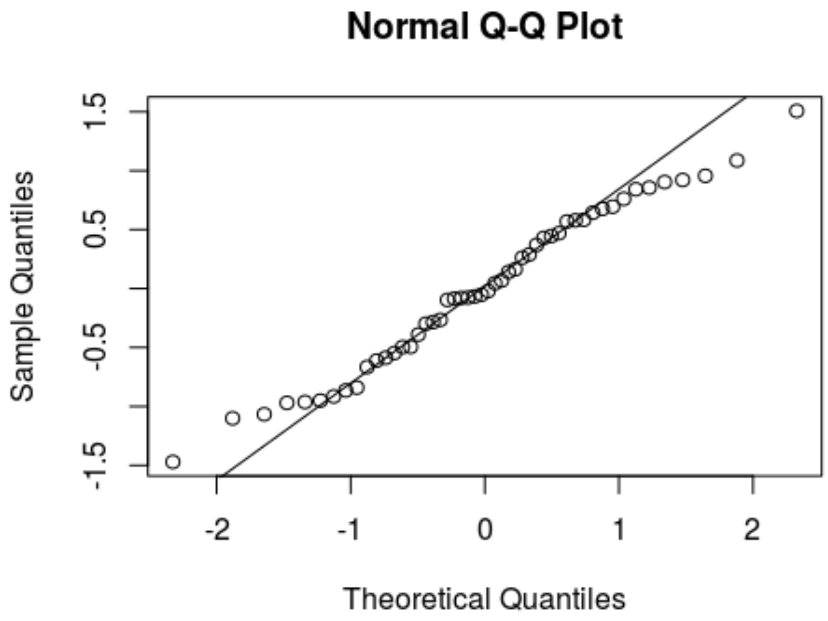
```
require(ggplot2)
require(gridExtra)
plot1 <- ggplot(data = datos, aes(habitantes, modelo$residuals)) + geom_point() +
         geom_smooth(color = "firebrick") + geom_hline(yintercept = 0) + theme_bw()
plot2 <- ggplot(data = datos, aes(asesinatos, modelo$residuals)) + geom_point() +
         geom_smooth(color = "firebrick") + geom_hline(yintercept = 0) + theme_bw()
plot3 <- ggplot(data = datos, aes(universitarios, modelo$residuals)) +
         geom_point() + geom_smooth(color = "firebrick") +
         geom_hline(yintercept = 0) + theme_bw()
plot4 <- ggplot(data = datos, aes(heladas, modelo$residuals)) + geom_point() +
         geom_smooth(color = "firebrick") + geom_hline(yintercept = 0) + theme_bw()
grid.arrange(plot1, plot2, plot3, plot4)
```




Se cumple la linealidad para todos los predictores

Distribución normal de los residuos:

```
qqnorm(modelo$residuals)
qqline(modelo$residuals)
```




```
shapiro.test(modelo$residuals)
```

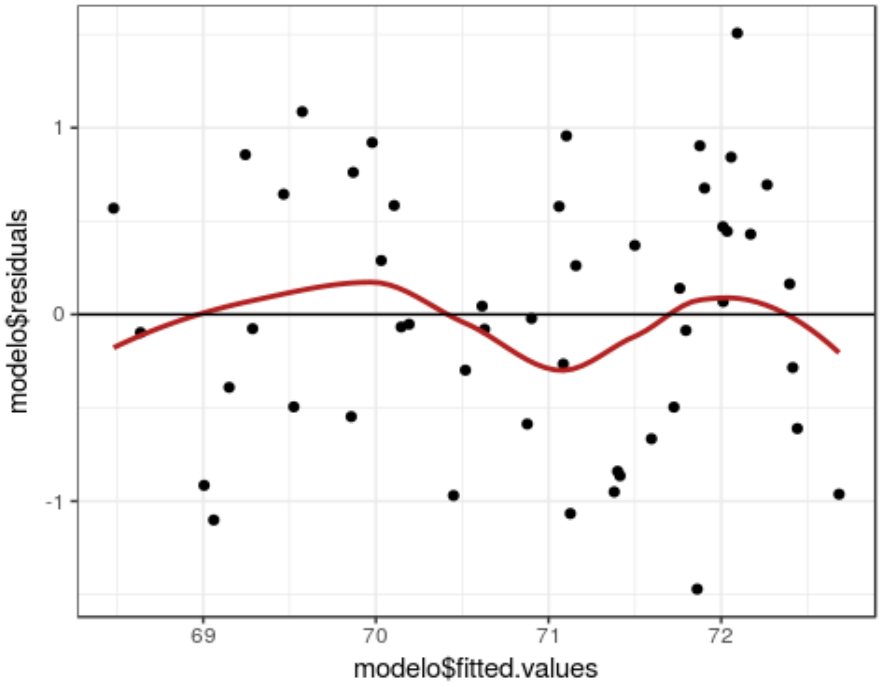
```
##
##  Shapiro-Wilk normality test
##
## data:  modelo$residuals
## W = 0.97935, p-value = 0.525
```

Tanto el análisis gráfico como es test de hipótesis confirman la normalidad.

Variabilidad constante de los residuos (homocedasticidad):

Al representar los residuos frente a los valores ajustados por el modelo, los primeros se tienen que distribuir de forma aleatoria en torno a cero, manteniendo aproximadamente la misma variabilidad a lo largo del eje X. Si se observa algún patrón específico, por ejemplo forma cónica o mayor dispersión en los extremos, significa que la variabilidad es dependiente del valor ajustado y por lo tanto no hay homocedasticidad.

```
ggplot(data = datos, aes(modelo$fitted.values, modelo$residuals)) + geom_point() +
geom_smooth(color = "firebrick", se = FALSE) + geom_hline(yintercept = 0) +
theme_bw()
```




```
library(lmtest)
bptest(modelo)
```

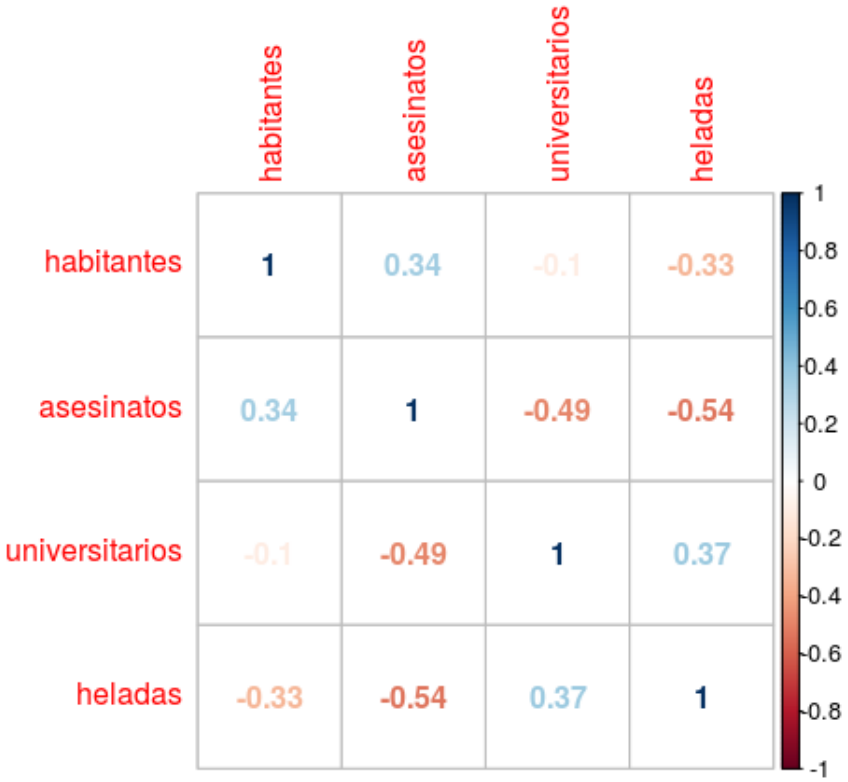
```
##
##  studentized Breusch-Pagan test
##
## data:  modelo
## BP = 6.2721, df = 4, p-value = 0.1797
```

No hay evidencias de falta de homocedasticidad.

No multicolinealidad:

Matriz de correlación entre predictores.

```
require(corrplot)
corrplot(cor(select(datos, habitantes, asesinatos, universitarios, heladas)),
         method = "number")
```




Análisis de Inflación de Varianza (VIF):

```
require(car)
vif(modelo)
```

```
##     habitantes     asesinatos universitarios       heladas
##       1.189835       1.727844       1.356791       1.498077
```

No hay predictores que muestren una correlación lineal muy alta ni inflación de varianza.

Autocorrelación:

```
require(car)
dwt(modelo, alternative = "two.sided")
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1      0.02867262      1.913997   0.758
##  Alternative hypothesis: rho != 0
```

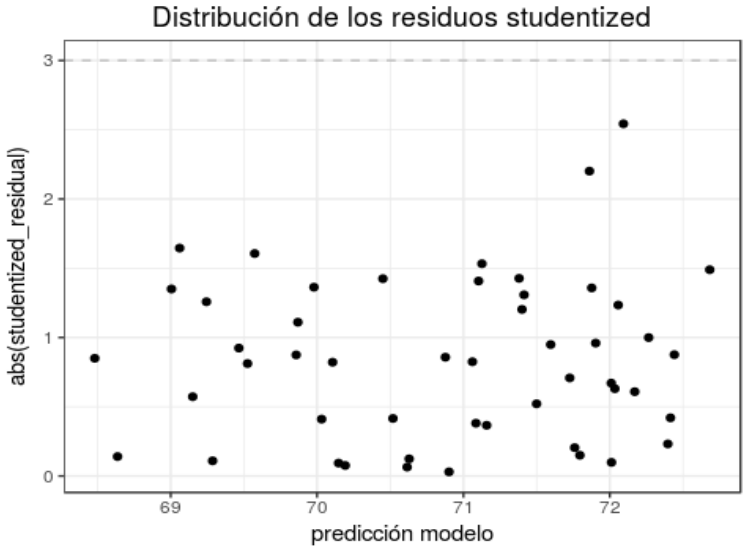
No hay evidencia de autocorrelación

Tamaño de la muestra:

No existe una condición establecida para el número mínimo de observaciones, pero para prevenir que una variable resulte muy influyente cuando realmente no lo es, se recomienda que la cantidad de observaciones sea entre 10 y 20 veces el número de predictores. En este caso debería haber como mínimo 40 observaciones y se dispone de 50 por lo que es apropiado.

## 5.Identificación de posibles valores atípicos o influyentes

```
library(dplyr)
datos$studentized_residual <- rstudent(modelo)
ggplot(data = datos, aes(x = predict(modelo), y = abs(studentized_residual))) +
geom_hline(yintercept = 3, color = "grey", linetype = "dashed") +
# se identifican en rojo observaciones con residuos estandarizados absolutos > 3
geom_point(aes(color = ifelse(abs(studentized_residual) > 3, 'red', 'black'))) +
scale_color_identity() + labs(title = "Distribución de los residuos studentized",
                              x = "predicción modelo") +
theme_bw() + theme(plot.title = element_text(hjust = 0.5))
```



```
which(abs(datos$studentized_residual) > 3)
```

```
## integer(0)
```

No se identifica ninguna observación atípica.

```
summary(influence.measures(modelo))
```

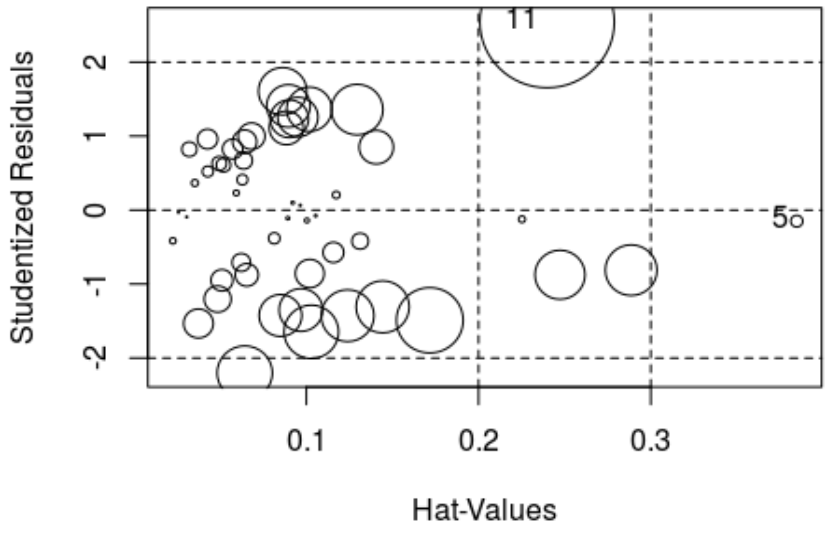
```
## Potentially influential observations of
##   lm(formula = esp_vida ~ habitantes + asesinatos + universitarios +
heladas, data = datos) :
##
##    dfb.1_ dfb.hbtn dfb.assn dfb.unvr dfb.hlds dffit   cov.r   cook.d
## 2   0.41   0.18    -0.40    -0.35    -0.16    -0.50    1.36_*  0.05
## 5   0.04  -0.09     0.00    -0.04     0.03    -0.12    1.81_*  0.00
## 11 -0.03  -0.57    -0.28     0.66    -1.24_*   1.43_*  0.74    0.36
## 28  0.40   0.14    -0.42    -0.29    -0.28    -0.52    1.46_*  0.05
## 32  0.01  -0.06     0.00     0.00    -0.01    -0.07    1.44_*  0.00
##    hat
## 2   0.25
## 5   0.38_*
## 11  0.24
## 28  0.29
## 32  0.23
```

En la tabla generada se recogen las observaciones que son significativamente influyentes en al menos uno de los predictores (una columna para cada predictor). Las tres últimas columnas son 3 medidas distintas para cuantificar la influencia. A modo de guía se pueden considerar excesivamente influyentes aquellas observaciones para las que:

- Leverages (hat): Se consideran observaciones influyentes aquellas cuyos valores *hat* superen $2.5((p+1)/n)$, siendo $p$ el número de predictores y $n$ el número de observaciones.
- Distancia Cook (cook.d): Se consideran influyentes valores superiores a 1.

La visualización gráfica de las influencias se obtiene del siguiente modo:

```
influencePlot(modelo)
```




```
##       StudRes       Hat       CookD
## 5  -0.1500614 0.3847592 0.002879053
## 11  2.5430162 0.2397924 0.363778638
```

Los análisis muestran varias observaciones influyentes (posición 5 y 11) que exceden los límites de preocupación para los valores de *Leverages* o *Distancia Cook*. Estudios más exhaustivos consistirían en rehacer el modelo sin las observaciones y ver el impacto.

## 6.Conclusión

El modelo lineal múltiple

$$Esperanza\ de\ vida = 5.014e^{-05} habitantes - 3.001e^{-01} asesinatos + 4.658e^{-02} universitarios - 5.943e^{-03} heladas$$

es capaz de explicar el 73.6% de la variabilidad observada en la esperanza de vida ($R^2$: 0.736, $R^2$-Adjusted: 0.7126). El test F muestra que es significativo (*p-value*: 1.696e-12). Se satisfacen todas las condiciones para este tipo de regresión múltiple. Dos observaciones (posición 5 y 11) podrían estar influyendo de forma notable en el modelo.

# Ejemplo2. Predictores numéricos y categóricos.

*Se dispone de un dataset que contiene información de 30 libros. Se conoce del peso total de cada libro, el volumen que tiene y el tipo de tapas (duras o blandas). Se quiere generar un modelo lineal múltiple que permita predecir el peso de un libro en función de su volumen y del tipo de tapas.*

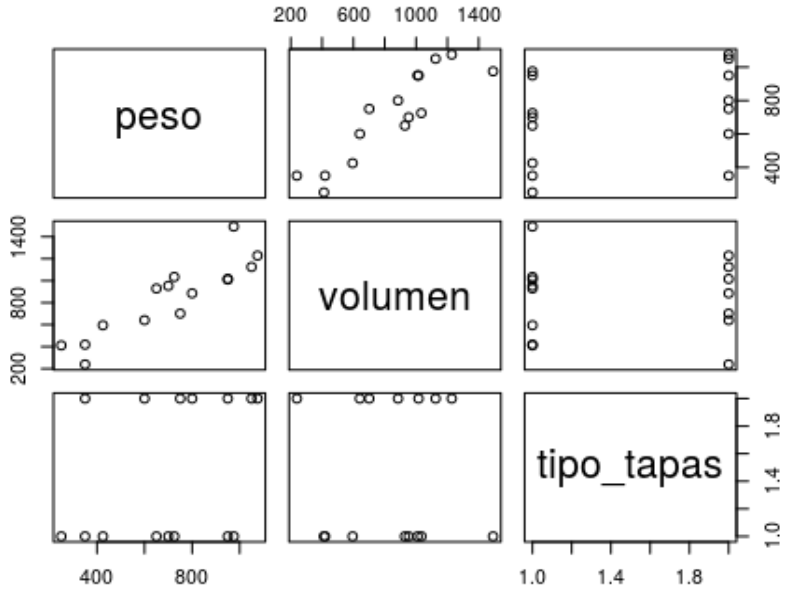
```
datos <- data.frame(peso = c(800, 950, 1050, 350, 750, 600, 1075, 250, 700,
    650, 975, 350, 950, 425, 725), volumen = c(885, 1016, 1125, 239, 701, 641,
    1228, 412, 953, 929, 1492, 419, 1010, 595, 1034), tipo_tapas = c("duras",
    "duras", "duras", "duras", "duras", "duras", "duras", "blandas", "blandas",
    "blandas", "blandas", "blandas", "blandas", "blandas", "blandas"))
head(datos, 4)
```

```
##   peso volumen tipo_tapas
## 1  800     885      duras
## 2  950    1016      duras
## 3 1050    1125      duras
## 4  350     239      duras
```

### 1.Analizar la correlación entre cada par de variables cuantitativas y diferencias del valor promedio entre las categóricas

Se enfrentan cada par de variables cuantitativas mediante un diagrama de dispersión múltiple (*pairwise scatterplot*) para intuir si existe relación lineal o monotónica con la variable respuesta. Si no la hay, no es adecuado emplear un modelo de regresión lineal. Además, se estudia la relación entre variables para detectar posible colinealidad. Para las variables de tipo categórico se genera un *boxplot* con sus niveles para intuir su influencia en la variable dependiente.

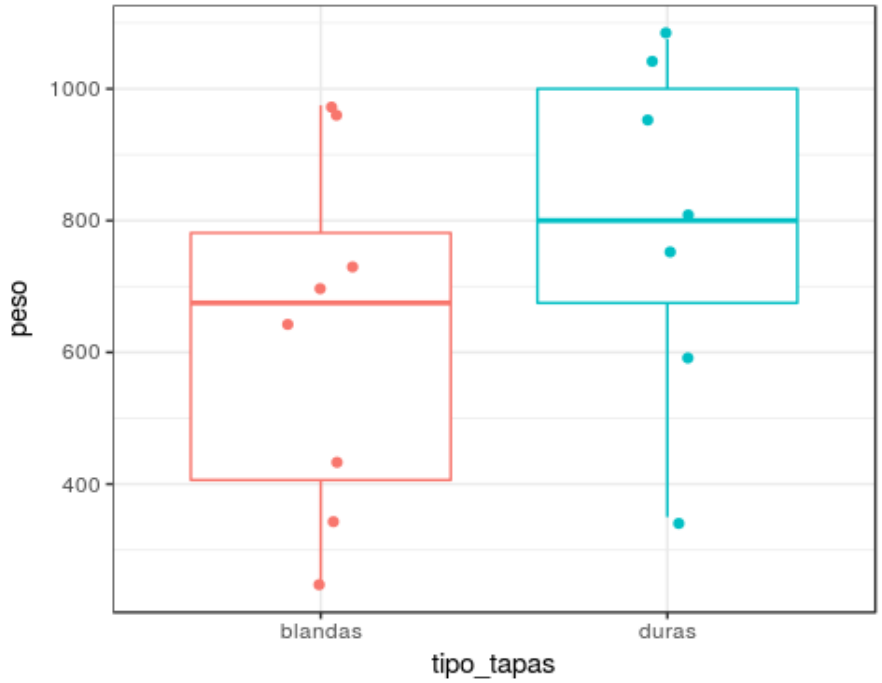
```
pairs(datos)
```




```
cor.test(datos$peso, datos$volumen, method = "pearson")
```

```
##
##  Pearson's product-moment correlation
##
## data:  datos$peso and datos$volumen
## t = 7.271, df = 13, p-value = 6.262e-06
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.7090393 0.9651979
## sample estimates:
##       cor
## 0.8958988
```

```
ggplot(data = datos, mapping = aes(x = tipo_tapas, y = peso, color = tipo_tapas)) +
geom_boxplot() + geom_jitter(width = 0.1) +
theme_bw() + theme(legend.position = "none")
```




El análisis gráfico y de correlación muestran una relación lineal significativa entre la variable *peso* y *volumen*. La variable *tipo_tapas* parece influir de forma significativa en el peso. Ambas variables pueden ser buenos predictores en un modelo lineal múltiple para la variable dependiente peso.

## 2.Generar el modelo lineal múltiple

```
modelo <- lm(peso ~ volumen + tipo_tapas, data = datos)
summary(modelo)
```

```
##
## Call:
## lm(formula = peso ~ volumen + tipo_tapas, data = datos)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -110.10  -32.32  -16.10   28.93  210.95
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      13.91557   59.45408   0.234 0.818887
## volumen           0.71795    0.06153  11.669  6.6e-08 ***
```

```
## tipo_tapasduras 184.04727   40.49420   4.545 0.000672 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 78.2 on 12 degrees of freedom
## Multiple R-squared:  0.9275, Adjusted R-squared:  0.9154
## F-statistic: 76.73 on 2 and 12 DF,  p-value: 1.455e-07
```

```
confint(modelo)
```

```
##                        2.5 %      97.5 %
## (Intercept)     -115.6237330 143.4548774
## volumen            0.5839023   0.8520052
## tipo_tapasduras   95.8179902 272.2765525
```

Cada una de las pendientes de un modelo de regresión lineal múltiple se define del siguiente modo: Si el resto de variables se mantienen constantes, por cada unidad que aumenta el predictor en cuestión, la variable $Y$ varía en promedio tantas unidades como indica la pendiente. En el caso del predictor *volume*, si el resto de variables no varían, por cada unidad de *volumen* que aumenta el libro el peso se incrementa en promedio 0.71795 unidades.

Cuando un predictor es cualitativo, uno de sus niveles se considera de referencia (el que no aparece en la tabla de resultados) y se le asigna el valor de 0. El valor de la pendiente de cada nivel de un predictor cualitativo se define como el promedio de unidades que dicho nivel está por encima o debajo del nivel de referencia. Para el predictor *tipo_tapas*, el nivel de referencia es *tapas blandas* por lo que si el libro tiene este tipo de tapas se le da a la variable el valor 0 y si es de *tapas duras* el valor 1. Acorde al modelo generado, los libros de tapa dura son en promedio 184.04727 unidades de peso superiores a los de tapa blanda.

$$Peso\ libro = 13.91557 + 0.71795\ volumen + 184.04727\ tipotapas$$

El modelo es capaz de explicar el 92.75% de la variabilidad observada en el peso de los libros *(R-squared: 0.9275)*. El valor de $R^2$-ajustado es muy alto y cercano al $R^2$ *(Adjusted R-squared: 0.9154)* lo que indica que el modelo contiene predictores útiles. El test F muestra un *p-value* de 1.455e-07 por lo que el modelo en conjunto es significativo. Esto se corrobora con el *p-value* de cada predictor, en ambos casos significativo.

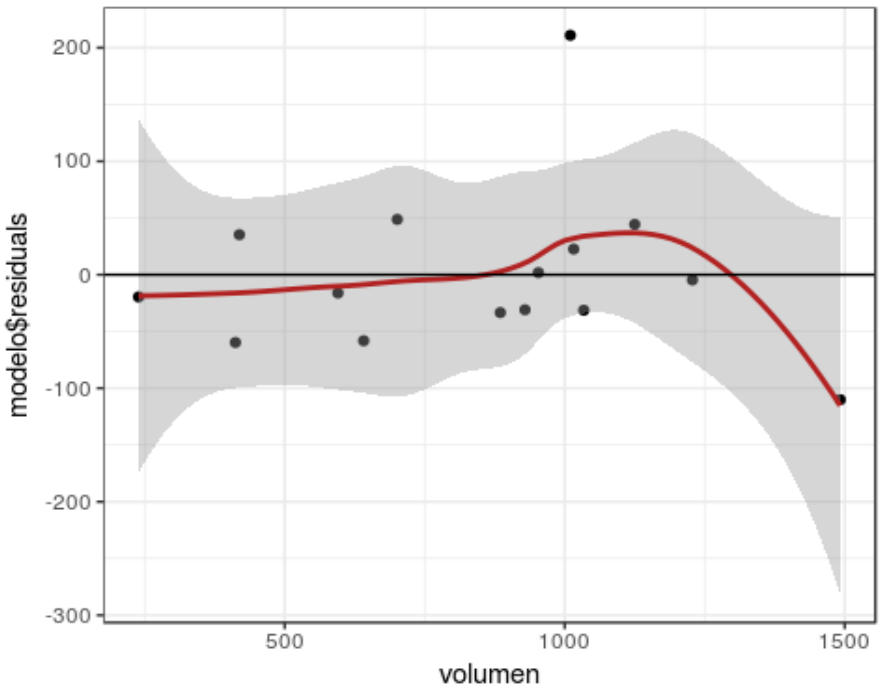
### 3.Elección de los predictores

En este caso, al solo haber dos predictores, a partir del *summary* del modelo se identifica que ambas variables incluidas son importantes.

### 4.Condiciones para la regresión múltiple lineal

1.Relación lineal entre los predictores numéricos y la variable dependiente:

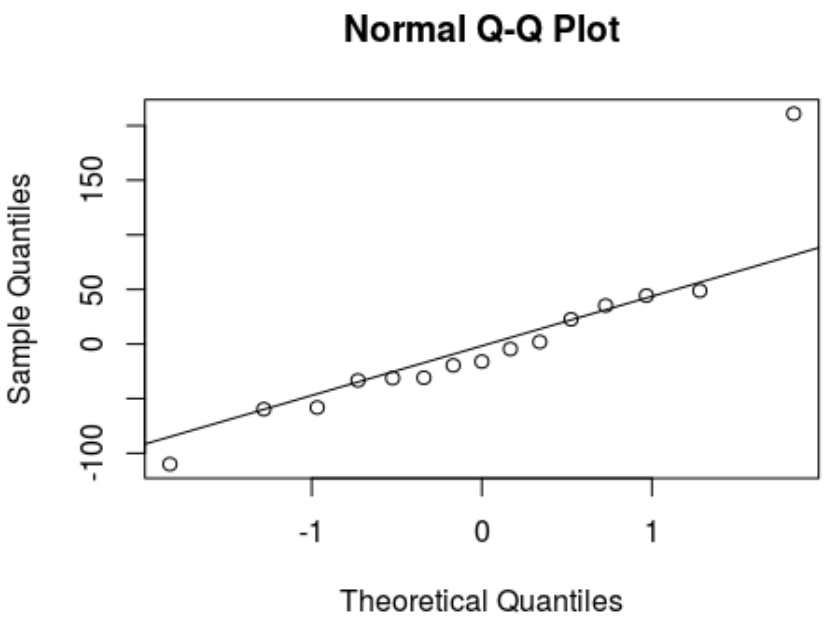
```
require(ggplot2)
ggplot(data = datos, aes(x = volumen, y = modelo$residuals)) + geom_point() +
geom_smooth(color = "firebrick") + geom_hline(yintercept = 0) + theme_bw()
```




Se satisface la condición de linealidad. Se aprecia un posible dato atípico.

2.Distribución normal de los residuos:

```
qqnorm(modelo$residuals)
qqline(modelo$residuals)
```




```
shapiro.test(modelo$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  modelo$residuals
## W = 0.85497, p-value = 0.02043
```

La condición de normalidad no se satisface, posiblemente debido a un dato atípico. Se repite el análisis excluyendo la observación a la que pertenece el residuo atípico.

```
which.max(modelo$residuals)
```

```
## 13
```

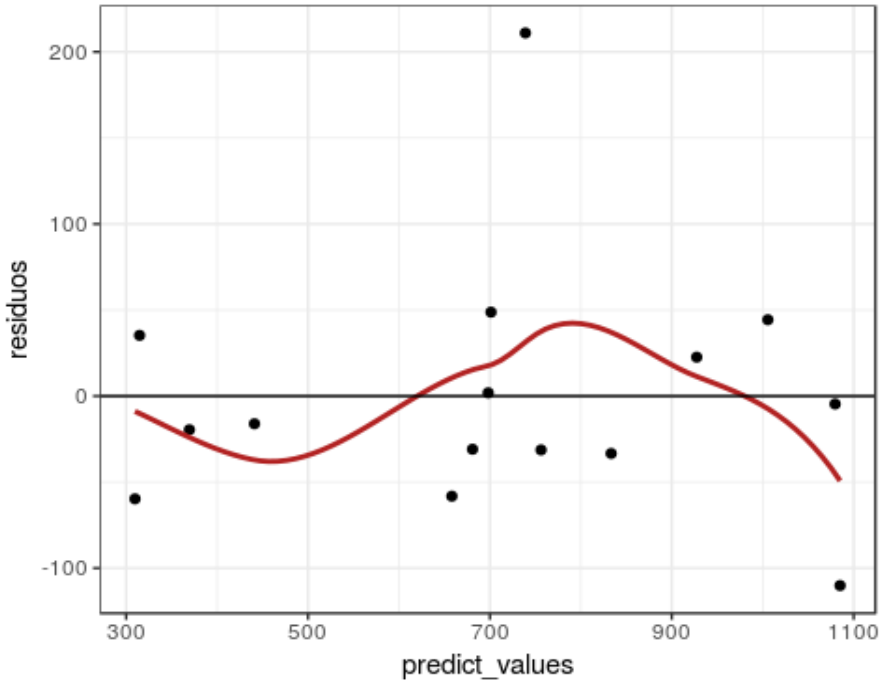
```
shapiro.test(modelo$residuals[-13])
```

```
##
##  Shapiro-Wilk normality test
##
## data:  modelo$residuals[-13]
## W = 0.9602, p-value = 0.7263
```

Se confirma que los residuos sí se distribuyen de forma normal a excepción de un dato extremo. Es necesario estudiar en detalle la influencia de esta observación para determinar si el modelo es más preciso sin ella.

3.Variabilidad constante de los residuos:

```
ggplot(data = data.frame(predict_values = predict(modelo),
                         residuos = residuals(modelo)),
       aes(x = predict_values, y = residuos)) + geom_point() +
geom_smooth(color = "firebrick", se = FALSE) + geom_hline(yintercept = 0) +
theme_bw()
```




```
library(lmtest)
bptest(modelo)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  modelo
## BP = 2.0962, df = 2, p-value = 0.3506
```

No hay evidencias que indiquen falta de homocedasticidad.

4.No multicolinealidad:

Dado que solo hay un predictor cuantitativo no se puede dar colinealidad.

5.Autocorrelación:

```
require(car)
dwt(modelo,alternative = "two.sided")
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1    0.0004221711      1.970663   0.786
##  Alternative hypothesis: rho != 0
```

No hay evidencia de autocorrelación

6.Tamaño de la muestra:

No existe una condición establecida para el número mínimo de observaciones pero, para prevenir que una variable resulte muy influyente cuando realmente no lo es, se recomienda que la cantidad de observaciones sea entre 10 y 20 veces el número de predictores. En este caso debería haber como mínimo 20 observaciones y se dispone de 15 por lo que se debería considerar incrementar la muestra.

## 5.Identificación de posibles valores atípicos o influyentes

```
require(car)
outlierTest(modelo)
```

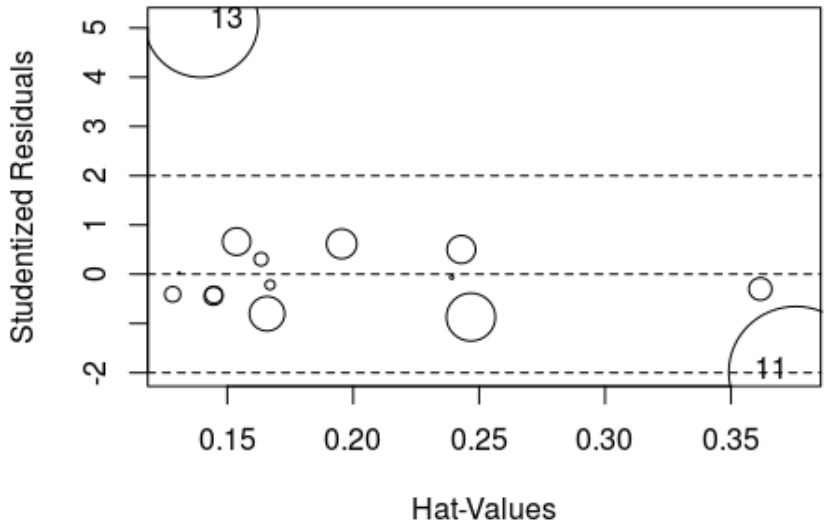
```
##    rstudent unadjusted p-value Bonferonni p
## 13 5.126833         0.00032993     0.004949
```

Tal como se apreció en el estudio de normalidad de los residuos, la observación 13 tiene un residuo estandarizado >3 (más de 3 veces la desviación estándar de los residuos) por lo que se considera un dato atípico. El siguiente paso es determinar si es influyente.

```
summary(influence.measures(modelo))
```

```
## Potentially influential observations of
##   lm(formula = peso ~ volumen + tipo_tapas, data = datos) :
##
##    dfb.1_ dfb.vlmn dfb.tp_t dffit   cov.r   cook.d hat
## 4  -0.16   0.18    -0.10    -0.23    1.98_*  0.02   0.36
## 11  0.70  -1.26_*   0.57    -1.54_*  0.83    0.64   0.38
## 13  0.31   0.67    -1.31_*   2.07_*  0.04_*  0.46   0.14
```

```
influencePlot(modelo)
```




```
##      StudRes       Hat     CookD
## 11 -1.989711 0.3757842 0.6372979
## 13  5.126833 0.1397761 0.4581972
```

El análisis muestran varias observaciones influyentes aunque ninguna excede los límites de preocupación para los valores de *Leverages hat* ($> 2.5(2+1)/15 = 0.5$) o *Distancia Cook* (>1). Estudios más exhaustivos consistirían en rehacer el modelo sin las observaciones y ver el impacto.

## 6.Conclusión

El modelo lineal múltiple

$$Peso\ libro = 13.91557 + 0.71795\ volumen + 184.04727\ tipotapas$$

es capaz de explicar el 92.75% de la variabilidad observada en el peso de los libros (R-squared: 0.9275, Adjusted R-squared: 0.9154). El test F muestra que es significativo (1.455e-07). Se satisfacen todas las condiciones para este tipo de regresión.

# Extensión del modelo lineal

Los modelos de regresión lineal presentan dos grandes ventajas, que son capaces de describir con suficiente precisión muchos escenarios que se dan en el mundo real y que los resultados son fácilmente interpretables. Sin embargo, para que sean totalmente válidos se tienen que satisfacer una serie de condiciones muy restrictivas que en la práctica no siempre se cumplen. Dos de las condiciones más importantes son que la relación entre los predictores y la variable respuesta debe ser *aditiva* y *lineal*. La aditividad implica que el efecto que tienen los cambios en el predictor $X_j$ sobre la variable respuesta $Y$ es independiente de los valores que tomen los otros predictores del modelo. La condición de linealidad implica que la variación en la variable respuesta $Y$ debida al cambio de una unidad en el predictor $X_j$ es constante, independientemente del valor de $X_j$. Existen dos aproximaciones clásicas que permiten relajar estas condiciones cuando se trabaja con modelos lineales.

## Interacción de predictores

*Supóngase que el departamento de ventas de una empresa quiere estudiar la influencia que tiene la publicidad a través de distintos canales sobre el número de ventas de un producto. Se dispone de un conjunto de datos que contiene los ingresos (en millones) conseguido por ventas en 200 regiones, así como la cantidad de presupuesto, también en millones, destinado a anuncios por radio, TV y periódicos en cada una de ellas.*

```
tv <- c(230.1, 44.5, 17.2, 151.5, 180.8, 8.7, 57.5, 120.2, 8.6, 199.8, 66.1,
    214.7, 23.8, 97.5, 204.1, 195.4, 67.8, 281.4, 69.2, 147.3, 218.4, 237.4,
    13.2, 228.3, 62.3, 262.9, 142.9, 240.1, 248.8, 70.6, 292.9, 112.9, 97.2,
    265.6, 95.7, 290.7, 266.9, 74.7, 43.1, 228, 202.5, 177, 293.6, 206.9, 25.1,
    175.1, 89.7, 239.9, 227.2, 66.9, 199.8, 100.4, 216.4, 182.6, 262.7, 198.9,
    7.3, 136.2, 210.8, 210.7, 53.5, 261.3, 239.3, 102.7, 131.1, 69, 31.5, 139.3,
    237.4, 216.8, 199.1, 109.8, 26.8, 129.4, 213.4, 16.9, 27.5, 120.5, 5.4,
    116, 76.4, 239.8, 75.3, 68.4, 213.5, 193.2, 76.3, 110.7, 88.3, 109.8, 134.3,
    28.6, 217.7, 250.9, 107.4, 163.3, 197.6, 184.9, 289.7, 135.2, 222.4, 296.4,
    280.2, 187.9, 238.2, 137.9, 25, 90.4, 13.1, 255.4, 225.8, 241.7, 175.7,
    209.6, 78.2, 75.1, 139.2, 76.4, 125.7, 19.4, 141.3, 18.8, 224, 123.1, 229.5,
    87.2, 7.8, 80.2, 220.3, 59.6, 0.7, 265.2, 8.4, 219.8, 36.9, 48.3, 25.6,
    273.7, 43, 184.9, 73.4, 193.7, 220.5, 104.6, 96.2, 140.3, 240.1, 243.2,
    38, 44.7, 280.7, 121, 197.6, 171.3, 187.8, 4.1, 93.9, 149.8, 11.7, 131.7,
    172.5, 85.7, 188.4, 163.5, 117.2, 234.5, 17.9, 206.8, 215.4, 284.3, 50,
    164.5, 19.6, 168.4, 222.4, 276.9, 248.4, 170.2, 276.7, 165.6, 156.6, 218.5,
    56.2, 287.6, 253.8, 205, 139.5, 191.1, 286, 18.7, 39.5, 75.5, 17.2, 166.8,
    149.7, 38.2, 94.2, 177, 283.6, 232.1)
radio <- c(37.8, 39.3, 45.9, 41.3, 10.8, 48.9, 32.8, 19.6, 2.1, 2.6, 5.8, 24,
    35.1, 7.6, 32.9, 47.7, 36.6, 39.6, 20.5, 23.9, 27.7, 5.1, 15.9, 16.9, 12.6,
```

```
    3.5, 29.3, 16.7, 27.1, 16, 28.3, 17.4, 1.5, 20, 1.4, 4.1, 43.8, 49.4, 26.7,
    37.7, 22.3, 33.4, 27.7, 8.4, 25.7, 22.5, 9.9, 41.5, 15.8, 11.7, 3.1, 9.6,
    41.7, 46.2, 28.8, 49.4, 28.1, 19.2, 49.6, 29.5, 2, 42.7, 15.5, 29.6, 42.8,
    9.3, 24.6, 14.5, 27.5, 43.9, 30.6, 14.3, 33, 5.7, 24.6, 43.7, 1.6, 28.5,
    29.9, 7.7, 26.7, 4.1, 20.3, 44.5, 43, 18.4, 27.5, 40.6, 25.5, 47.8, 4.9,
    1.5, 33.5, 36.5, 14, 31.6, 3.5, 21, 42.3, 41.7, 4.3, 36.3, 10.1, 17.2, 34.3,
    46.4, 11, 0.3, 0.4, 26.9, 8.2, 38, 15.4, 20.6, 46.8, 35, 14.3, 0.8, 36.9,
    16, 26.8, 21.7, 2.4, 34.6, 32.3, 11.8, 38.9, 0, 49, 12, 39.6, 2.9, 27.2,
    33.5, 38.6, 47, 39, 28.9, 25.9, 43.9, 17, 35.4, 33.2, 5.7, 14.8, 1.9, 7.3,
    49, 40.3, 25.8, 13.9, 8.4, 23.3, 39.7, 21.1, 11.6, 43.5, 1.3, 36.9, 18.4,
    18.1, 35.8, 18.1, 36.8, 14.7, 3.4, 37.6, 5.2, 23.6, 10.6, 11.6, 20.9, 20.1,
    7.1, 3.4, 48.9, 30.2, 7.8, 2.3, 10, 2.6, 5.4, 5.7, 43, 21.3, 45.1, 2.1,
    28.7, 13.9, 12.1, 41.1, 10.8, 4.1, 42, 35.6, 3.7, 4.9, 9.3, 42, 8.6)
periodico <- c(69.2, 45.1, 69.3, 58.5, 58.4, 75, 23.5, 11.6, 1, 21.2, 24.2,
    4, 65.9, 7.2, 46, 52.9, 114, 55.8, 18.3, 19.1, 53.4, 23.5, 49.6, 26.2, 18.3,
    19.5, 12.6, 22.9, 22.9, 40.8, 43.2, 38.6, 30, 0.3, 7.4, 8.5, 5, 45.7, 35.1,
    32, 31.6, 38.7, 1.8, 26.4, 43.3, 31.5, 35.7, 18.5, 49.9, 36.8, 34.6, 3.6,
    39.6, 58.7, 15.9, 60, 41.4, 16.6, 37.7, 9.3, 21.4, 54.7, 27.3, 8.4, 28.9,
    0.9, 2.2, 10.2, 11, 27.2, 38.7, 31.7, 19.3, 31.3, 13.1, 89.4, 20.7, 14.2,
    9.4, 23.1, 22.3, 36.9, 32.5, 35.6, 33.8, 65.7, 16, 63.2, 73.4, 51.4, 9.3,
    33, 59, 72.3, 10.9, 52.9, 5.9, 22, 51.2, 45.9, 49.8, 100.9, 21.4, 17.9,
    5.3, 59, 29.7, 23.2, 25.6, 5.5, 56.5, 23.2, 2.4, 10.7, 34.5, 52.7, 25.6,
    14.8, 79.2, 22.3, 46.2, 50.4, 15.6, 12.4, 74.2, 25.9, 50.6, 9.2, 3.2, 43.1,
    8.7, 43, 2.1, 45.1, 65.6, 8.5, 9.3, 59.7, 20.5, 1.7, 12.9, 75.6, 37.9, 34.4,
    38.9, 9, 8.7, 44.3, 11.9, 20.6, 37, 48.7, 14.2, 37.7, 9.5, 5.7, 50.5, 24.3,
    45.2, 34.6, 30.7, 49.3, 25.6, 7.4, 5.4, 84.8, 21.6, 19.4, 57.6, 6.4, 18.4,
    47.4, 17, 12.8, 13.1, 41.8, 20.3, 35.2, 23.7, 17.6, 8.3, 27.4, 29.7, 71.8,
    30, 19.6, 26.6, 18.2, 3.7, 23.4, 5.8, 6, 31.6, 3.6, 6, 13.8, 8.1, 6.4, 66.2,
    8.7)
ventas <- c(22.1, 10.4, 9.3, 18.5, 12.9, 7.2, 11.8, 13.2, 4.8, 10.6, 8.6, 17.4,
    9.2, 9.7, 19, 22.4, 12.5, 24.4, 11.3, 14.6, 18, 12.5, 5.6, 15.5, 9.7, 12,
    15, 15.9, 18.9, 10.5, 21.4, 11.9, 9.6, 17.4, 9.5, 12.8, 25.4, 14.7, 10.1,
    21.5, 16.6, 17.1, 20.7, 12.9, 8.5, 14.9, 10.6, 23.2, 14.8, 9.7, 11.4, 10.7,
    22.6, 21.2, 20.2, 23.7, 5.5, 13.2, 23.8, 18.4, 8.1, 24.2, 15.7, 14, 18,
    9.3, 9.5, 13.4, 18.9, 22.3, 18.3, 12.4, 8.8, 11, 17, 8.7, 6.9, 14.2, 5.3,
    11, 11.8, 12.3, 11.3, 13.6, 21.7, 15.2, 12, 16, 12.9, 16.7, 11.2, 7.3, 19.4,
    22.2, 11.5, 16.9, 11.7, 15.5, 25.4, 17.2, 11.7, 23.8, 14.8, 14.7, 20.7,
    19.2, 7.2, 8.7, 5.3, 19.8, 13.4, 21.8, 14.1, 15.9, 14.6, 12.6, 12.2, 9.4,
    15.9, 6.6, 15.5, 7, 11.6, 15.2, 19.7, 10.6, 6.6, 8.8, 24.7, 9.7, 1.6, 12.7,
    5.7, 19.6, 10.8, 11.6, 9.5, 20.8, 9.6, 20.7, 10.9, 19.2, 20.1, 10.4, 11.4,
    10.3, 13.2, 25.4, 10.9, 10.1, 16.1, 11.6, 16.6, 19, 15.6, 3.2, 15.3, 10.1,
    7.3, 12.9, 14.4, 13.3, 14.9, 18, 11.9, 11.9, 8, 12.2, 17.1, 15, 8.4, 14.5,
    7.6, 11.7, 11.5, 27, 20.2, 11.7, 11.8, 12.6, 10.5, 12.2, 8.7, 26.2, 17.6,
    22.6, 10.3, 17.3, 15.9, 6.7, 10.8, 9.9, 5.9, 19.6, 17.3, 7.6, 9.7, 12.8,
    25.5, 13.4)

datos <- data.frame(tv, radio, periodico, ventas)
```

El modelo lineal múltiple que se obtiene empleando las variables *tv*, *radio* y *periódico* como predictores de ventas es el siguiente:

```
modelo <- lm(formula = ventas ~ tv + radio + periodico, data = datos)
summary(modelo)
```

```
##
## Call:
## lm(formula = ventas ~ tv + radio + periodico, data = datos)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.8277 -0.8908  0.2418  1.1893  2.8292
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.938889   0.311908   9.422   <2e-16 ***
## tv           0.045765   0.001395  32.809   <2e-16 ***
## radio        0.188530   0.008611  21.893   <2e-16 ***
## periodico   -0.001037   0.005871  -0.177     0.86
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.686 on 196 degrees of freedom
## Multiple R-squared:  0.8972, Adjusted R-squared:  0.8956
## F-statistic: 570.3 on 3 and 196 DF,  p-value: < 2.2e-16
```

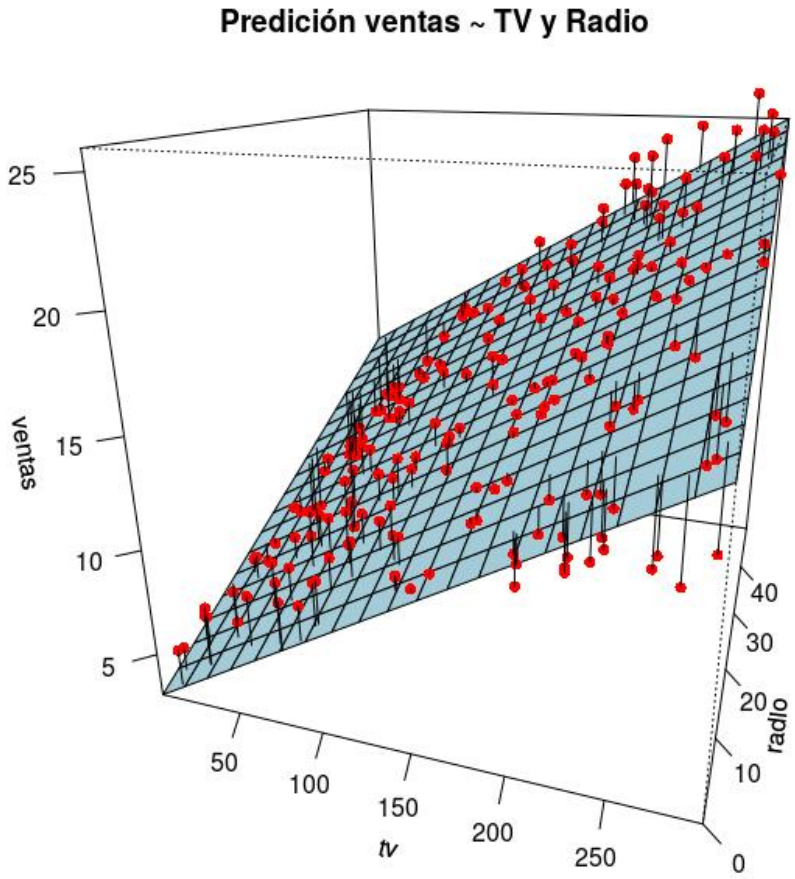
Acorde al *p-value* obtenido para el coeficiente parcial de correlación de *periodico*, esta variable no contribuye de forma significativa al modelo. Como resultado de este análisis se concluye que las variables *tv* y *radio* están asociadas con la cantidad de ventas.

```
modelo <- update(modelo, .~. -periodico)
summary(modelo)
```

```
##
## Call:
## lm(formula = ventas ~ tv + radio, data = datos)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.7977 -0.8752  0.2422  1.1708  2.8328
##
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.92110    0.29449   9.919   <2e-16 ***
## tv           0.04575    0.00139  32.909   <2e-16 ***
## radio        0.18799    0.00804  23.382   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.681 on 197 degrees of freedom
## Multiple R-squared:  0.8972, Adjusted R-squared:  0.8962
## F-statistic: 859.6 on 2 and 197 DF,  p-value: < 2.2e-16
```

```r
# Al ser un modelo con dos predictores continuos se puede representar en 3D
rango_tv <- range(datos$tv)
nuevos_valores_tv <- seq(from = rango_tv[1], to = rango_tv[2], length.out = 20)
rango_radio <- range(datos$radio)
nuevos_valores_radio <- seq(from = rango_radio[1], to = rango_radio[2],
                            length.out = 20)

predicciones <- outer(X = nuevos_valores_tv, Y = nuevos_valores_radio,
                      FUN = function(tv, radio) {predict(object = modelo,
                                                         newdata = data.frame(tv, radio))
                                                         })
superficie <- persp(x = nuevos_valores_tv, y = nuevos_valores_radio,
                    z = predicciones, theta = 18, phi = 20, col = "lightblue",
                    shade = 0.1, xlab = "tv", ylab = "radio", zlab = "ventas",
                    ticktype = "detailed", main = "Predición ventas ~ TV y Radio")

observaciones <- trans3d(datos$tv, datos$radio, datos$ventas, superficie)
error         <- trans3d(datos$tv, datos$radio, fitted(modelo), superficie)
points(observaciones, col = "red", pch = 16)
segments(observaciones$x, observaciones$y, error$x, error$y)
```




El modelo lineal a partir del cual se han obtenido las conclusiones asume que el efecto sobre las ventas debido a un incremento en el presupuesto de uno de los medios de comunicación es independiente del presupuesto gastado en los otros. Por ejemplo, el modelo lineal considera que el efecto promedio sobre las ventas debido a aumentar en una unidad el presupuesto de anuncios en TV es siempre de 0.04575, independientemente de la cantidad invertida en anuncios por radio. Sin embargo, la representación gráfica muestra que el modelo tiende a sobrevalorar las ventas cuando el presupuesto es muy alto en uno de los medios pero muy bajo en el otro. Por contra, los valores de ventas predichos por el modelo están por debajo de las ventas reales cuando el presupuesto está repartido de forma equitativa entre ambos medios. Este comportamiento sugiere que existe interacción entre los predictores, por lo que el efecto de cada uno de ellos sobre la variable respuesta depende en cierta medida del valor que tome el otro predictor.

Tal y como se ha definido previamente, un modelo lineal con dos predictores sigue la ecuación:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + e$$

Acorde a esta definición, el incremento de una unidad en el predictor $X_1$ produce un incremento promedio de la variable $Y$ de $\beta_1$. Modificaciones en el predictor $X_2$ no alteran este hecho, y lo mismo ocurre con $X_2$ respecto a $X_1$. Para que el modelo pueda contemplar la interacción se introduce un tercer predictor, llamado *interaction term*, que se construye con el producto de los predictores $X_1$ y $X_2$.

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 X_2 + e$$

La reorganización de los términos resulta en:

$$Y = \beta_0 + (\beta_1 + \beta_3 X_2) X_1 + \beta_2 X_2 + e$$

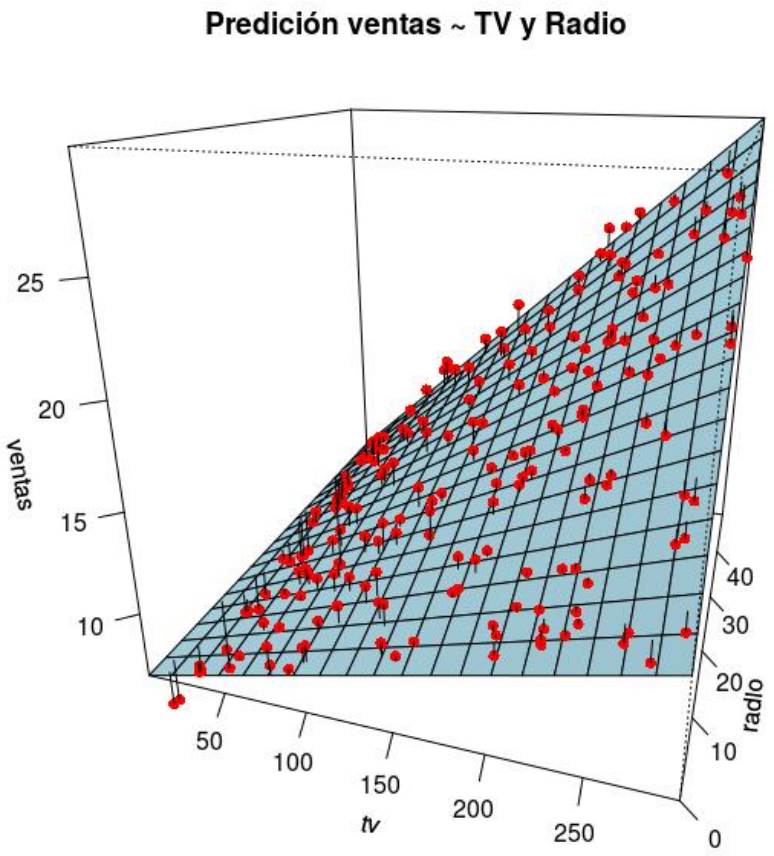
El efecto de $X_1$ sobre $Y$ ya no es constante, sino que depende del valor que tome $X_2$.

En R se puede introducir interacción entre predictores de dos formas, indicando los predictores individuales y entre cuales se quiere evaluar la interacción, o bien de forma directa.

```
modelo_interaccion <- lm(formula = ventas ~ tv + radio + tv:radio, data = datos)
summary(modelo_interaccion)

# lm(formula = ventas ~ tv * radio, data = datos) es equivalente.
```

```
##
## Call:
## lm(formula = ventas ~ tv + radio + tv:radio, data = datos)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.3366 -0.4028  0.1831  0.5948  1.5246
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 6.750e+00  2.479e-01  27.233   <2e-16 ***
## tv          1.910e-02  1.504e-03  12.699   <2e-16 ***
## radio       2.886e-02  8.905e-03   3.241   0.0014 **
## tv:radio    1.086e-03  5.242e-05  20.727   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9435 on 196 degrees of freedom
## Multiple R-squared:  0.9678, Adjusted R-squared:  0.9673
## F-statistic:  1963 on 3 and 196 DF,  p-value: < 2.2e-16
```

```
# Al ser un modelo con dos predictores continuos se puede representar en 3D
rango_tv          <- range(datos$tv)
nuevos_valores_tv <- seq(from = rango_tv[1], to = rango_tv[2], length.out = 20)
rango_radio       <- range(datos$radio)
nuevos_valores_radio <- seq(from = rango_radio[1], to = rango_radio[2],
                            length.out = 20)
# La función outer() permite aplicar una función a cada combinación de los
# parámetros x, y pasados como argumento, es una alternativa a expand.grid()
predicciones <- outer(X = nuevos_valores_tv, Y = nuevos_valores_radio,
                      FUN = function(tv, radio) {
                            predict(object = modelo_interaccion,
                                    newdata = data.frame(tv, radio))
                             })
superficie <- persp(x = nuevos_valores_tv, y = nuevos_valores_radio,
                    z = predicciones, theta = 18, phi = 20, col = "lightblue",
                    shade = 0.1, xlab = "tv", ylab = "radio", zlab = "ventas",
                    ticktype = "detailed", main = "Predición ventas ~ TV y Radio")
# Se pueden representar las observaciones a partir de las cuales se ha creado la
# superficie así como segmentos que midan la distancia respecto al modelo generado.
observaciones <- trans3d(datos$tv, datos$radio, datos$ventas, superficie)
error <- trans3d(datos$tv, datos$radio, fitted(modelo_interaccion), superficie)
points(observaciones, col = "red", pch = 16)
segments(observaciones$x, observaciones$y, error$x, error$y)
```



Los resultados muestran una evidencia clara de que la interacción *tv* x *radio* es significativa y de que el modelo que incorpora la interacción (*Adjusted R-squared* = 0.9673) es superior al modelo que solo contemplaba el efecto de los predictores por separado (*Adjusted R-squared* = 0.8956).

Se puede emplear un ANOVA para realizar un test de hipótesis y obtener un *p-value* que evalúe la hipótesis nula de que ambos modelos se ajustan a los datos igual de bien.

```
anova(modelo, modelo_interaccion)
```

```
## Analysis of Variance Table
##
## Model 1: ventas ~ tv + radio
## Model 2: ventas ~ tv + radio + tv:radio
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1    197 556.91
## 2    196 174.48  1    382.43 429.59 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

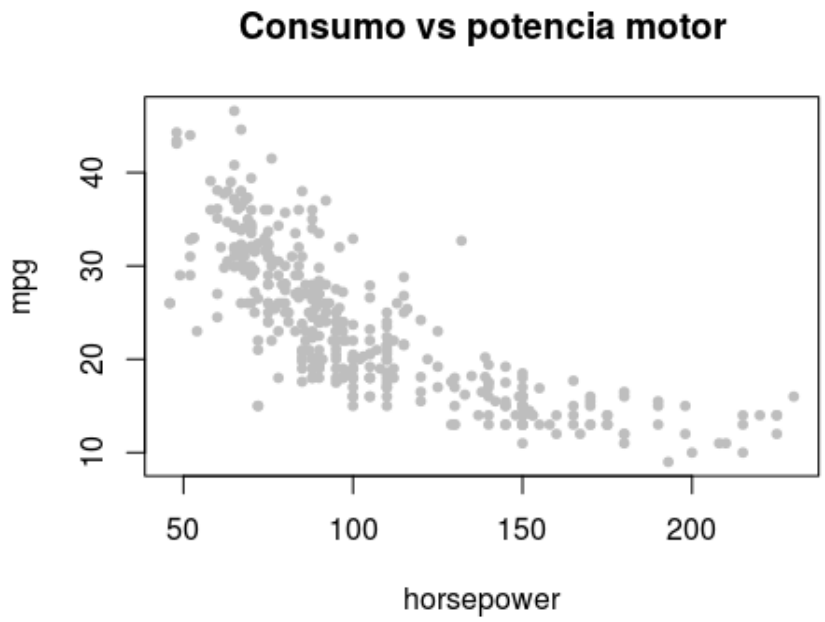
En los modelos de regresión lineal múltiple que incorporan interacciones entre predictores hay que tener en cuenta el *hierarchical principle*, según el cual, si se incorpora al modelo una interacción entre predictores, se deben incluir siempre los predictores individuales que participan en la interacción, independientemente de que su p-value sega significativo o no.

La interacción entre predictores no está limitada a predictores cuantitativos, también puede crearse interacción entre predictor cuantitativo y cualitativo.

## Regresión polinomial

*Una marca de coches quiere generar un modelo de regresión que permita predecir el consumo de combustible (mpg) en función de la potencia del motor (horsepower).*

```
library(ISLR)
attach(Auto)
plot(x = horsepower, y = mpg, main = "Consumo vs potencia motor", pch = 20,
     col = "grey")
```

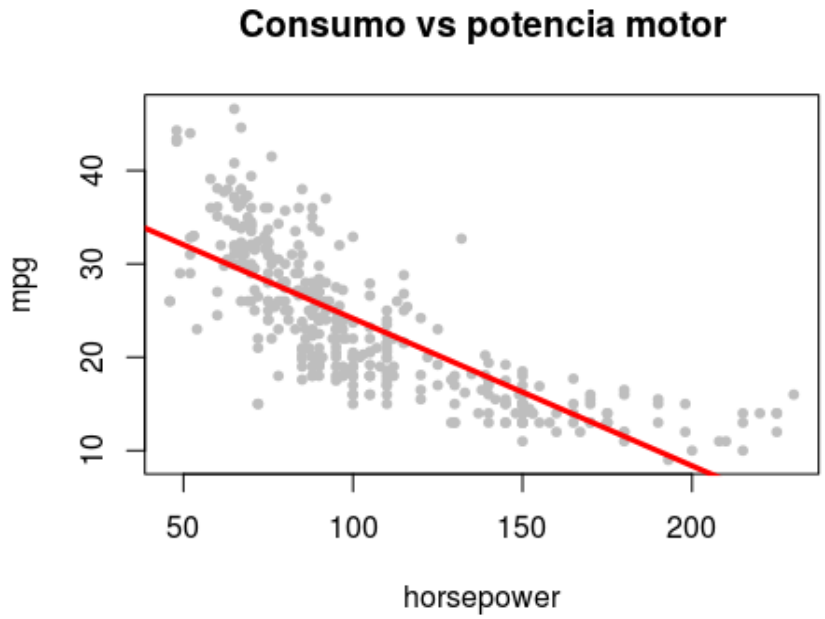



La representación gráfica de los datos muestra una fuerte asociación entre el consumo y la potencia del motor. La distribución de las observaciones apunta a que la relación entre ambas variables tiene cierta curvatura, por lo que un modelo lineal no puede captarla por completo.

```
attach(Auto)
modelo_lineal <- lm(formula = mpg ~ horsepower, data = Auto)
summary(modelo_lineal)
```

```
##
## Call:
## lm(formula = mpg ~ horsepower, data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.5710  -3.2592  -0.3435   2.7630  16.9240
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 39.935861   0.717499   55.66   <2e-16 ***
## horsepower  -0.157845   0.006446  -24.49   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.906 on 390 degrees of freedom
## Multiple R-squared:  0.6059, Adjusted R-squared:  0.6049
## F-statistic: 599.7 on 1 and 390 DF,  p-value: < 2.2e-16
```

```
plot(x = horsepower, y = mpg, main = "Consumo vs potencia motor", pch = 20,
     col = "grey")
abline(modelo_lineal, lwd = 3, col = "red")
```




Una forma de incorporar asociaciones no lineales a un modelo lineal es mediante transformaciones de los predictores incluidos en el modelo, por ejemplo, elevándolos a distintas potencias. En este caso, el tipo de curvatura es de tipo cuadrática, por lo que un polinomio de segundo grado podría mejorar el modelo.

En R se pueden generar modelos de regresión polinómica de diferentes formas:

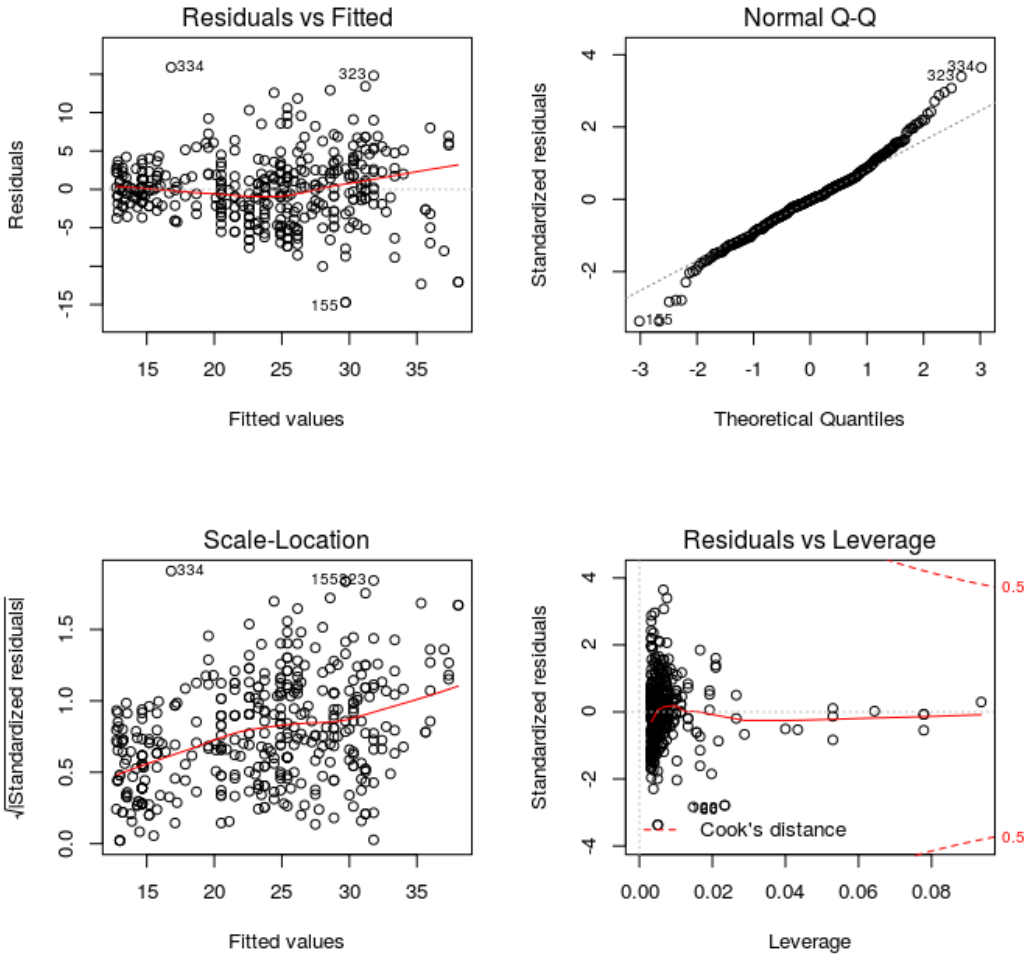
- Identificando cada elemento del polinomio: `modelo_pol2 <- lm(formula = mpg ~ horsepower + I(horsepower^2), data = Auto)` El uso de `I()` es necesario ya que el símbolo ^ tiene otra función dentro de las formula de R.
- Con la función `poly()`: `lm(formula = mpg ~ poly(horsepower, 2), data = Auto)`

```
modelo_cuadratico <- lm(formula = mpg ~ poly(horsepower, 2), data = Auto)
summary(modelo_cuadratico)
```

```
##
## Call:
## lm(formula = mpg ~ poly(horsepower, 2), data = Auto)
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.7135  -2.5943  -0.0859   2.2868  15.8961
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)              23.4459     0.2209  106.13   <2e-16 ***
## poly(horsepower, 2)1 -120.1377     4.3739  -27.47   <2e-16 ***
## poly(horsepower, 2)2   44.0895     4.3739   10.08   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.374 on 389 degrees of freedom
## Multiple R-squared:  0.6876, Adjusted R-squared:  0.686
## F-statistic:   428 on 2 and 389 DF,  p-value: < 2.2e-16
```

```
par(mfrow = c(2, 2))
plot(modelo_cuadratico)
```



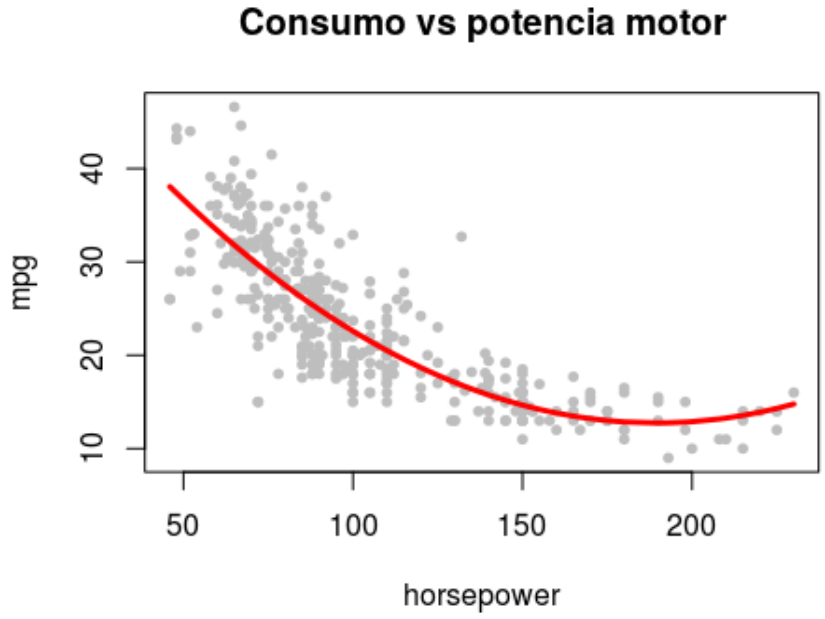
El valor $R^2$ del modelo cuadrático (0.6876) es mayor que el obtenido con el modelo lineal simple (0.6059) y el *p-value* del término cuadrático es altamente significativo. Se puede concluir que el modelo cuadrático recoge mejor la verdadera relación entre el consumo de los vehículos y la potencia de su motor.

Al tratarse de modelos anidados, es posible emplear un ANOVA para contrastar la hipótesis nula de que ambos modelos se ajustan a los datos igual de bien.

```
anova(modelo_lineal, modelo_cuadratico)
```

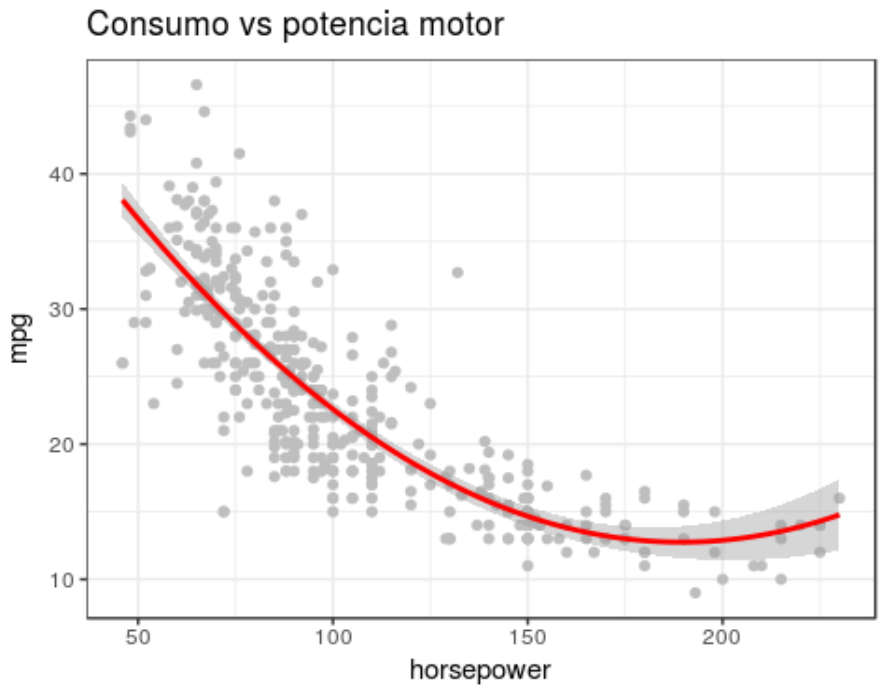
```
## Analysis of Variance Table
##
## Model 1: mpg ~ horsepower
## Model 2: mpg ~ poly(horsepower, 2)
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1    390 9385.9
## 2    389 7442.0  1    1943.9 101.61 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
plot(x = horsepower, y = mpg, main = "Consumo vs potencia motor", pch = 20,
     col = "grey")
puntos_interpolados <- seq(from = min(horsepower), to = max(horsepower), by = 1)
prediccion <- predict(object = modelo_cuadratico,
                      newdata = data.frame(horsepower = horsepower))
lines(sort(horsepower), prediccion[order(horsepower)], col = "red", lwd = 3)
```



Misma representación con `ggplot2`

```
library(ggplot2)
ggplot(Auto, aes(x = horsepower, y = mpg)) + geom_point(colour = "grey") +
stat_smooth(method = "lm", formula = y ~ poly(x, 2), colour = "red") +
labs(title = "Consumo vs potencia motor") +
theme_bw()
```

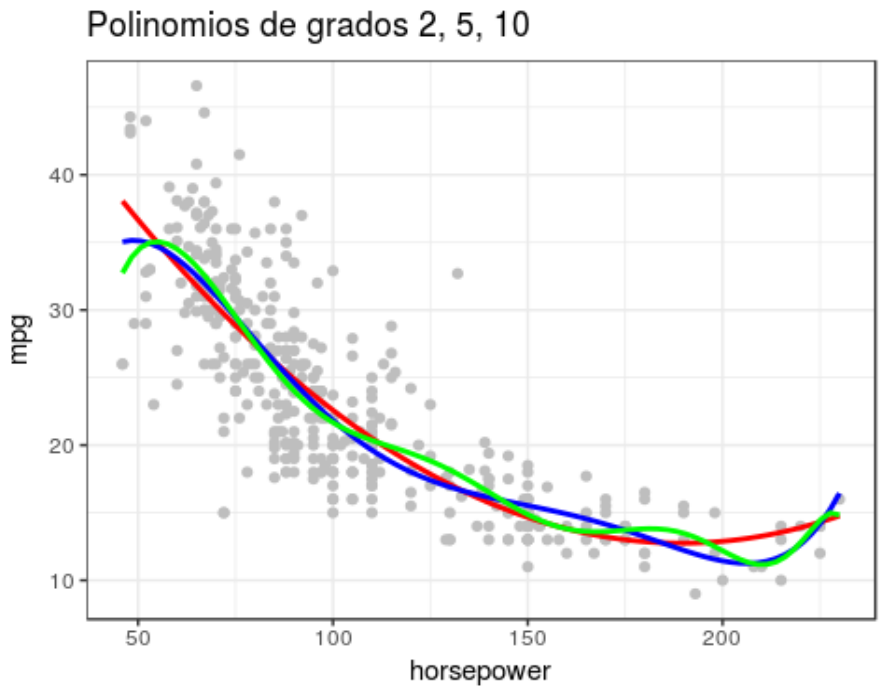



La elección del grado del polinomio influye directamente en la flexibilidad del modelo. Cuanto mayor es el grado del polinomio más se ajusta el modelo a las observaciones, un polinomio de grado $n^{o}observaciones - 1$ pasa por todos los puntos. Por lo tanto, es importante no excederse en el grado del polinomio para no causar problemas de *overfitting* (no suelen recomendarse grados superiores a 3-4). Existen varias estrategias para identificar el grado óptimo del polinomio:

- Incrementar secuencialmente el orden del polinomio hasta que la nueva incorporación no sea significativa.
- Iniciar el proceso con un polinomio de grado alto e ir eliminando secuencialmente, de mayor a menor, los términos no significativos.
- Emplear un ANOVA para comparar cada nuevo modelo con el de orden inferior y determinar si la mejora es significativa. Este proceso es equivalente al primero. Ver ejemplo en Regresión Polinomial: incorporar no-linealidad a los modelos lineales.
- Emplear *cross-validation*. En el capítulo Validación de modelos de regresión: Cross-validation, OneLeaveOut, Bootstrap se explica cómo emplear la validación-cruzada para identificar el grado adecuado.

Como norma general, si se incluye en el modelo un término polinómico, se tienen que incluir también como predictores los ordenes inferiores de esa variable aunque no sean significativos.

```
library(ggplot2)
ggplot(Auto, aes(x = horsepower, y = mpg)) +
geom_point(colour = "grey") + stat_smooth(method = "lm", formula = y ~ poly(x, 2),
colour = "red", se = FALSE) + stat_smooth(method = "lm", formula = y ~ poly(x, 5),
colour = "blue", se = FALSE) + stat_smooth(method = "lm",formula = y ~ poly(x, 10),
colour = "green", se= FALSE) + labs(title = "Polinomios de grados 2, 5, 10") +
theme_bw()
```



# Apuntes varios (miscellaneous)

En este apartado recojo comentarios, definiciones y puntualizaciones que he ido encontrando en diferentes fuentes y que, o bien no he tenido tiempo de introducir en el cuerpo principal del documento, o que he considerado que es mejor mantenerlos al margen como información complementaria.

## Identidad (identifiability) o colinealidad

*Linear Models with R, by Julian J. Faraway*

El método de mínimos cuadrados tiene una solución única solo si la matriz formada por los predictores es de rango máximo, es decir, que todas sus columnas (predictores) son linealmente independientes. En la práctica, esta condición de identidad suele violarse con frecuencia. Los siguientes son algunos escenarios en los que ocurre:

- Cuando uno de los predictores introducidos en el modelo es una transformación lineal o combinación de otros predictores presentes en el modelo. Por ejemplo, que la variable peso se introduzca en Kg y en libras o que se introduzcan como predictores el número de años de educación básica, el número de años de educación universitaria y el total de años de educación. Este problema se puede evitar estudiando la naturaleza de las variables disponibles y su relación.

- Sobresaturación del modelo, cuando hay más predictores que observaciones.

En R, cuando se intenta ajustar un modelo que sufre de falta de identidad, se excluyen automáticamente variables empezando por el final de la fórmula empleada para definir el modelo hasta alcanzar la identidad.

*El set de datos gala del paquete faraway contiene información sobre las islas que forman el archipielago de las Galápago. Entre las variables almacenadas están: número de especies que habitan la isla (*Species*), el área (*Area*) y altura (*Elevation*) de la isla, la distancia a la isla más próxima (*Nearest*), la distancia a la isla Santa Cruz (*Scruz*) y el área de la isla adyacente (*Adjacent*).*

```
library(faraway)
data(gala)
head(gala)
```

```
##              Species Endemics  Area Elevation Nearest Scruz Adjacent
## Baltra            58       23 25.09       346     0.6   0.6     1.84
## Bartolome         31       21  1.24       109     0.6  26.3   572.33
## Caldwell           3        3  0.21       114     2.8  58.7     0.78
## Champion          25        9  0.10        46     1.9  47.4     0.18
## Coamano            2        1  0.05        77     1.9   1.9   903.82
## Daphne.Major      18       11  0.34       119     8.0   8.0     1.84
```

Se añade una nueva variable que es combinación lineal de dos predictores, por ejemplo, la diferencia entre el área de la isla y el área de la isla adyacente, y se intenta ajustar el modelo. La función `lm()` detecta que el rango máximo es 6 y que se han introducido 7 predictores, por lo que excluye el último predictor.

```
gala$diferencia_area <- gala$Area - gala$Adjacent
modelo <- lm(Species ~ Area + Elevation + Nearest + Scruz + Adjacent +
             diferencia_area, data = gala)
summary(modelo)
```

```
##
## Call:
## lm(formula = Species ~ Area + Elevation + Nearest + Scruz + Adjacent +
##     diferencia_area, data = gala)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -111.679  -34.898   -7.862   33.460  182.584
##
## Coefficients: (1 not defined because of singularities)
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       7.068221  19.154198   0.369 0.715351
## Area             -0.023938   0.022422  -1.068 0.296318
## Elevation         0.319465   0.053663   5.953 3.82e-06 ***
## Nearest           0.009144   1.054136   0.009 0.993151
## Scruz            -0.240524   0.215402  -1.117 0.275208
## Adjacent         -0.074805   0.017700  -4.226 0.000297 ***
## diferencia_area         NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 60.98 on 24 degrees of freedom
## Multiple R-squared:  0.7658, Adjusted R-squared:  0.7171
## F-statistic:  15.7 on 5 and 24 DF,  p-value: 6.838e-07
```

## Estimación, intervalo de confianza y significancia de $\hat{\boldsymbol{\beta}}$

*Linear Models with R, by Julian J. Faraway*

Si únicamente se quiere obtener la estimación de los coeficientes de correlación ($\hat{\beta}$) de los predictores que forman un modelo, no es necesario asumir ninguna distribución de los residuos ($\epsilon$). Sin embargo, si además de la estimación se desea obtener intervalos de confianza o realizar test de hipótesis para contrastar su significancia, entonces sí es necesario. En el caso del método de mínimos cuadrados, se tiene que asumir que los residuos son independientes y que se distribuyen de forma normal con media 0 y varianza $\sigma^2$. Cuando la condición de normalidad no se satisface, existe la posibilidad de recurrir a los test de permutación para calcular significancia (*p-value*) y al *bootstrapping* para calcular intervalos de confianza.

## Comparación de modelos mediante test de hipótesis, F-test

*Linear Models with R, by Julian J. Faraway*

Cuando se dispone de múltiples predictores para crear un modelo, es recomendable estudiar si todos ellos son necesarios o si se puede conseguir un modelo más óptimo empleando solo algunos de ellos.

Supóngase un modelo $M$ y otro modelo $m$, de menor tamaño, formado por un subconjunto de los predictores contenidos en $M$. Si la diferencia en el ajuste es muy pequeña, acorde al principio de parsimonia, el modelo $m$ es más adecuado. Es posible contrastar si la diferencia en ajuste es significativa mediante la comparación de los residuos. En concreto el estadístico empleado es:

$$\frac{RSS_m - RSS_M}{RSS_M}$$

Para evitar que el tamaño del modelo influya en el contraste, se divide la suma de residuos cuadrados *RSS* de cada modelo entre sus grados de libertad. El estadístico resultante sigue una distribución $F$.

$$\frac{(RSS_m - RSS_M)/(df_m - df_M)}{RSS_M/(df_M)} \sim F_{df_m - df_M,\, df_M}$$

donde $df$ son los grados de libertad del modelo, que equivalen al número de observaciones menos el número de predictores.

**F-test para la significancia del modelo (todos sus predictores a la vez)**

Uno de los primeros resultados que hay que evaluar al ajustar un modelo es el resultado del test de significancia $F$. Este contraste responde a la pregunta de si el modelo en su conjunto es capaz de predecir la variable respuesta mejor de lo esperado por azar, o lo que es equivalente, si al menos uno de los predictores que forman el modelo contribuye de forma significativa. Para realizar este contraste se compara la suma de residuos cuadrados del modelo de interés con la del modelo sin predictores, formado únicamente por la media (también conocido como suma de cuadrados corregidos por la media, $TSS$).

$$F = \frac{(TSS - RSS)/(p-1)}{RSS/(n-p)}$$

Con frecuencia, la hipótesis nula y alternativa de este test se describen como:

$$H_0: \beta_1 = \ldots = \beta_{p-1} = 0$$
$$H_a: \text{al menos un } \beta_i \neq 0$$

Si el test $F$ resulta significativo, implica que el modelo es útil, pero no que sea el mejor. Podría ocurrir que alguno de sus predictores no fuese necesario.

*El set de datos gala del paquete faraway contiene información sobre las islas que forman el archipiélago de las Galápago. Entre las variables almacenadas están: número de especies que habitan la isla (*Species*), el área (*Area*) y altura (*Elevation*) de la isla, la distancia a la isla más próxima (*Nearest*), la distancia a la isla Santa Cruz (*Scruz*) y el área de la isla adyacente (*Adjacent*).*

```
library(faraway)
data(gala)
head(gala)
```

```
##              Species Endemics  Area Elevation Nearest Scruz Adjacent
## Baltra            58       23 25.09       346     0.6   0.6     1.84
## Bartolome         31       21  1.24       109     0.6  26.3   572.33
## Caldwell           3        3  0.21       114     2.8  58.7     0.78
## Champion          25        9  0.10        46     1.9  47.4     0.18
## Coamano            2        1  0.05        77     1.9   1.9   903.82
## Daphne.Major      18       11  0.34       119     8.0   8.0     1.84
```

Se ajusta un modelo con todos los predictores y el modelo nulo que contiene únicamente la medias.

```
modelo <- lm(Species ~ Area + Elevation + Nearest + Scruz + Adjacent, data = gala)
modelo_nulo <- lm(Species ~ 1, data = gala)
```

Se compara la suma de residuos cuadrados de los dos modelos. Esto puede hacerse mediante la función `anova()` indicando como argumento los dos modelos.

```
anova(modelo_nulo, modelo)
```

```
## Analysis of Variance Table
##
## Model 1: Species ~ 1
## Model 2: Species ~ Area + Elevation + Nearest + Scruz + Adjacent
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1     29 381081
## 2     24  89231  5    291850 15.699 6.838e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

El contraste muestra claras evidencias ($p - value = 6.838 * 10^-7$) en contra de la hipótesis nula de que $\beta_1 = \beta_2 = \beta_3 = \beta_4 = \beta_5 = 0$. El modelo en su conjunto es capaz de explicar la varianza en la variable *Species* mejor de lo esperado por azar. Al menos uno de sus predictores contribuye de forma significativa a la predicción.

El *p-value* obtenido es igual al obtenido al hacer el `summary` del modelo. Esto es así porque la función `summary()` hace este mismo contraste de forma automática.

```
summary(modelo)
```

```
## Call:
## lm(formula = Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
##     data = gala)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -111.679  -34.898   -7.862   33.460  182.584
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.068221  19.154198   0.369 0.715351
## Area        -0.023938   0.022422  -1.068 0.296318
## Elevation    0.319465   0.053663   5.953 3.82e-06 ***
```

```
## Nearest      0.009144   1.054136   0.009 0.993151
## Scruz       -0.240524   0.215402  -1.117 0.275208
## Adjacent    -0.074805   0.017700  -4.226 0.000297 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 60.98 on 24 degrees of freedom
## Multiple R-squared:  0.7658, Adjusted R-squared:  0.7171
## F-statistic:  15.7 on 5 and 24 DF,  p-value: 6.838e-07
```

A modo de comprobación, se reproduce el proceso paso por paso:

```
# Extracción de RSS de los modelos
rss_0 <- deviance(modelo_nulo)
rss_modelo <- deviance(modelo)

# Extracción de los grados de libertad
df_0 <- modelo_nulo$df.residual
df_modelo <- modelo$df.residual

# Cálculo estadístico F
f <- ((rss_0 - rss_modelo)/(df_0 - df_modelo)/(rss_modelo/df_modelo))

# Cálculo de p-value
1 - pf(q = f, df1 = df_0 - df_modelo, df2 = df_modelo)

## [1] 6.837893e-07
```

### F-test para la significancia de un predictor individual $\beta_i$

A parte de estudiar la significancia de un modelo en su conjunto, es interesante poder conocer si un predictor en particular contribuye a dicho modelo o si podría ser eliminado. En este caso, la hipótesis que se desea contrastar es:

$$H_0: \beta_i = 0$$

$$H_0: \beta_i \neq 0$$

Supóngase que se quiere determinar si el predictor *Area* contribuye realmente en el modelo `Species ~ Area + Elevation + Nearest + Scruz + Adjacent`. Para saberlo, se ajusta el modelo completo y un modelo más simple en el que se ha excluido el predictor de interés.

```
modelo_completo <- lm(Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
                      data = gala)
modelo_simple <- lm(Species ~ Elevation + Nearest + Scruz + Adjacent, data = gala)
```

Una vez ajustados, se recurre a un test $F$ que contraste la diferencia en sus RSS (*Residual Sum of Squares*). Si la diferencia es significativa hay evidencias de que el predictor sí contribuye al modelo.

```
anova(modelo_completo, modelo_simple)
```

```
## Analysis of Variance Table
##
## Model 1: Species ~ Area + Elevation + Nearest + Scruz + Adjacent
## Model 2: Species ~ Elevation + Nearest + Scruz + Adjacent
##   Res.Df   RSS Df Sum of Sq      F Pr(>F)
## 1     24 89231
## 2     25 93469 -1   -4237.7 1.1398 0.2963
```

En este caso (*p-value = 0.2963*), la hipótesis nula de que el coeficiente de regresión del predictor *Area* en igual a cero no puede ser rechazada. El predictor *Area* no contribuye al modelo.

El resultado del contraste realizado mediante el *F-test* es equivalente al que se obtiene mediante un *t-test* empleando como estadístico:

$$t = \frac{\hat{\beta}_i}{se(\hat{\beta}_i)}$$

y comprobando su significancia en una distribución *t-student* con $n - p$ grados de libertad.

Este último es el test estadístico que se muestra al hacer `summary()` de un objeto `lm()`. Cada uno de los *p-values* reportados debe interpretarse como la comparación entre el modelo completo y el modelo resultante si se excluye el predictor al que está asociado dicho *p-value*

```
summary(modelo_completo)
```

```
##
## Call:
## lm(formula = Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
##     data = gala)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -111.679  -34.898   -7.862   33.460  182.584
##
## Coefficients:
```

```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.068221  19.154198   0.369 0.715351
## Area        -0.023938   0.022422  -1.068 0.296318
## Elevation    0.319465   0.053663   5.953 3.82e-06 ***
## Nearest      0.009144   1.054136   0.009 0.993151
## Scruz       -0.240524   0.215402  -1.117 0.275208
## Adjacent    -0.074805   0.017700  -4.226 0.000297 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 60.98 on 24 degrees of freedom
## Multiple R-squared:  0.7658, Adjusted R-squared:  0.7171
## F-statistic:  15.7 on 5 and 24 DF,  p-value: 6.838e-07
```

IMPORTANTE: La comparación de modelos mediante el test $F$ solo es posible cuando los modelos comparados están anidados, es decir, cuando el modelo de menor tamaño está formado por un subconjunto de los predictores del modelo de mayor tamaño.

## Comparar si dos coeficientes de regresión de un mismo modelo son iguales

*Linear Models with R, by Julian J. Faraway*

Supóngase un modelo en el que se ajusta la variable respuesta *éxito escolar* sobre los predictores *horas de estudio con la madre* y *horas de estudio con el padre*. Si ambos predictores son significativos, podría ser de interés estudiar si ambos son iguales, es decir, si padres y madres influyen de la misma forma.

Las hipótesis que contrastar son:

$$H_0: \beta_1 = \beta_2$$

$$H_A: \beta_1 \neq \beta_2$$

O lo que es lo mismo:

$$H_0: \beta_1 - \beta_2 = 0$$

$$H_A: \beta_1 - \beta_2 \neq 0$$

Existen dos aproximaciones distintas para responder a esta pregunta:

Wald Test

Esta aproximación suele ser la más intuitiva, ya que sigue la estructura típica de un test de hipótesis en la que el estadístico se calcula como el valor observado menos el valor de la hipótesis nula dividido entre el error estándar.

$$\left(\frac{(\beta_1 - \beta_2) - 0}{SE(\beta_1 - \beta_2)}\right)^2 = F_{1,n-p-1}$$

Dado que por definición

$$SE(A \pm B)^2 = \sigma^2(A \pm B) = Var(A \pm B) = Var(A) - Var(B) \pm 2Cov(A, B)$$

Se obtiene que

$$\left(\frac{(\beta_1 - \beta_2)}{\sqrt{sd(\beta_1)^2 + sd(\beta_2)^2 - 2sd(\beta_1, \beta_2)}}\right)^2 = F_{1,n-p-1}$$

Comparando el estadístico obtenido con una distribución *F* se pude conocer su significancia.

Test F con modelos anidados

Esta aproximación consiste en comparar dos modelos mediante un test *F*. El modelo completo contiene los predictores de interés junto con otros (si los hay). En el segundo modelo se sustituyen los dos predictores individuales por uno nuevo calculado como la suma de ellos. Si al comparar los dos modelos la diferencia es significativa, entonces hay evidencias en contra de la hipótesis nula de que $\beta_1 = \beta_2$.

- modelo_completo: lm(y ~ A + B + ...)
- modelo_reducido: lm(y ~ I(A + B) + ...)

Descripción más detalla da en https://www3.nd.edu/~rwilliam/stats2/l42.pdf

## Test de permutación (simulación Monte Carlo) como alternativa al F-test para la significancia del modelo y sus predictores

*Linear Models with R, by Julian J. Faraway*

El contraste de hipótesis basado en el *F-test* se fundamenta en el supuesto de que los residuos se distribuyen de forma normal. Si bien el teorema del límite central demuestra que, aun cuando la distribución se aleja de la normalidad, los resultados de la inferencia son buenos si el tamaño muestral es grande, no existe un tamaño exacto a partir del cual se garantice su

validez. Los test de permutación ofrecen una alternativa que no requiere asumir la normalidad de los residuos.

Supóngase un modelo con dos predictores, `Species ~ Nearest + Scruz`, para el que se calcula el estadístico *F*. El *p-value* asociado indica cómo de probable es obtener un valor de *F* igual o más extremo si no existe relación entre la variable respuesta y los predictores. Cuando se cumple la condición de $\epsilon \sim N(0, \sigma^2)$, esta probabilidad se puede obtener de la *F-distribution*.

```
modelo <- lm(Species ~ Nearest + Scruz, data = gala)
# Extracción del estadístico F
summary(modelo)$fstatistic
```

```
##      value      numdf      dendf
##  0.6019558  2.0000000 27.0000000
```

```
# Calculo de p-value basado en la distribución F
1 - pf(q = 0.6019558, df1 = 2, df2 = 27)
```

```
## [1] 0.5549255
```

Cuando no se cumple, se puede simular la hipótesis nula de no asociación entre la variable respuesta y todos predictores intercambiando aleatoriamente (permutación) la variable respuesta entre las observaciones. Tras cada permutación se reajusta el modelo y se calcula el estadístico *F*. Una vez almacenadas suficientes permutaciones (1000 - 10000), se calcula el porcentaje de casos en los que *F* ha resultado ser igual o mayor que el valor obtenido en el modelo inicial.

```
set.seed(123)
f <- rep(NA, 4000)
for (i in 1:4000) {
    modelo_temp <- lm(sample(Species) ~ Nearest + Scruz, data = gala)
    f[i] <- summary(modelo_temp)$fstatistic["value"]
}
mean(f >= 0.6019558)
```

```
## [1] 0.55825
```

El *p-value* obtenido por permutaciones (*0.55825*) es muy similar al obtenido por el método teórico basado en normalidad (*0.5549255*).

El método de permutaciones presenta la ventaja de ser igual de bueno que el método teórico (si se realizan suficientes permutaciones) cuando se cumple la condición de normalidad y además se puede aplicar también cuando no se cumple. La desventaja es que requiere de mucha más computación.

La misma aproximación por permutaciones puede emplearse para determinar si un predictor en particular contribuye de forma significativa al modelo, o lo que es lo mismo, si

está asociado a la variable respuesta. En este caso, se permuta únicamente el valor del predictor de interés, simulando así la hipótesis nula de que no existe asociación con la variable respuesta. En cada permutación se ajusta el modelo y se almacena el estadístico *t* obtenido para el predictor. Tras realizar suficientes permutaciones, se calcula el porcentaje de casos en los que el valor absoluto de *t* es igual o mayor al valor absoluto observado en el modelo inicial.

Supóngase un modelo con dos predictores, `Species ~ Nearest + Scruz`, para el que se quiere calcular la significancia del predictor *Scruz*.

```
# Se ajusta el modelo de interés
modelo <- lm(Species ~ Nearest + Scruz, data = gala)

# Se extrae el estadístico t del predictor Scruz y su significancia acorde
# al método teórico
summary(modelo)$coefficients["Scruz", ]

##   Estimate Std. Error    t value   Pr(>|t|)
## -0.4406401  0.4025312 -1.0946731  0.2833295

# Test de permutaciones
set.seed(123)
t <- rep(NA, 4000)  # vector para almacenar los estadísticos
for (i in 1:4000) {
    modelo_temp <- lm(Species ~ Nearest + sample(Scruz), data = gala)
    t[i] <- summary(modelo_temp)$coefficients["sample(Scruz)", "t value"]
}

mean(abs(t) >= abs(-1.0946731))
```

```
## [1] 0.26775
```

El *p-value* obtenido por permutaciones (*0.26775*) es muy similar al obtenido por el método teórico basado en normalidad (*0.2833295*).

## Bootstrapping como alternativa para calcular intervalos de confianza de $\hat{\beta}$

*Linear Models with R, by Julian J. Faraway*

El método empleado por la función `confint()` para calcular los intervalos de confianza de los coeficientes de regresión de un modelo lineal se basa en la distribución *t-student* y siguen la estructura:

$$\hat{\beta}_i \pm t_{n-p}^{\alpha/2} se(\hat{\beta})$$

Para que los intervalos basados en esta distribución teórica sean válidos, es necesario que los residuos del modelo se distribuyan de forma normal. Sí esto no ocurre, el método de

*bootstrapping* ofrece una alternativa que no requiere de esta condición. A continuación se muestra un ejemplo de cómo aplicar esté método para estimar los *CI 95%* del modelo `Species ~ Area + Elevation + Nearest + Scruz + Adjacent`. Para una descripción más detallada del *bootstrapping* consultar Resampling: Test de permutación, Simulación de Monte Carlo y Bootstrapping.

*En el libro Linear Models with R by Julian J. Faraway se hace resampling sobre los residuos del modelo. A mi parecer es más intuitivo hacer el resampling sobre los pares (respuesta, predictores). Las estimaciones de los intervalos que se obtienen no son exactamente iguales, por lo que no puedo garantizar la validez de esta aproximación.*

```
library(faraway)
data("gala")
# Intervalos basados en distribución teórica
confint(lm(Species ~ Area + Elevation + Nearest + Scruz + Adjacent, data = gala),
           level = 0.95)
```

```
##                   2.5 %      97.5 %
## (Intercept) -32.4641006 46.60054205
## Area         -0.0702158  0.02233912
## Elevation     0.2087102  0.43021935
## Nearest      -2.1664857  2.18477363
## Scruz        -0.6850926  0.20404416
## Adjacent     -0.1113362 -0.03827344
```

```
# Intervalos estimados por bootstrapping
set.seed(123)
n <- 4000  # número de iteraciones
# matriz para almacenar los coef. obtenidos en cada iteración del proceso
matriz_coeficientes <- matrix(NA, nrow = n, ncol = 6)  # 6 = 5 predictores + 1
intercept
colnames(matriz_coeficientes) <- c("intercept", colnames(gala)[3:7])

for (i in 1:n) {
    sample <- gala[sample(1:nrow(gala), replace = TRUE), ]
    modelo_temp <- lm(Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
        data = sample)
    matriz_coeficientes[i, ] <- coef(modelo_temp)
}

# Cálculo de los del intervalo por quantiles 0.05 y 0.975
apply(X = matriz_coeficientes, MARGIN = 2, FUN = function(x) {
    quantile(x, c(0.025, 0.975))
})
```

```
##       intercept        Area  Elevation   Nearest      Scruz    Adjacent
## 2.5%  -29.97688 -0.08915871 0.04181927 -4.542415 -0.6127654 -0.14607942
## 97.5%  34.88773  0.42042898 0.50428406  1.
```

## Interpretación de los coeficientes de regresión de un modelo lineal

*Linear Models with R, by Julian J. Faraway*

Cuando se emplea un modelo de regresión lineal para extraer información sobre la relación existente entre los predictores y la variable respuesta, hay que ser cautelosos y no llegar a conclusiones que van más allá de lo que dice el modelo.

*El set de datos gala del paquete faraway contiene información sobre las islas que forman el archipiélago de las Galápago. Entre las variables almacenadas están: número de especies que habitan la isla (*Species*), el área (*Area*) y altura (*Elevation*) de la isla, la distancia a la isla más próxima (*Nearest*), la distancia a la isla Santa Cruz (*Scruz*) y el área de la isla adyacente (*Adjacent*). Se quiere emplear un modelo de regresión lineal para estudiar la relación que tiene el predictor* Elevation con el número de especies que hay en la *isla.*

```
library(faraway)
data("gala")
modelo <- lm(Species ~ Area + Elevation + Nearest + Scruz + Adjacent, data = gala)
coef(modelo)
```

```
##  (Intercept)         Area    Elevation      Nearest        Scruz
##  7.068220709 -0.023938338  0.319464761  0.009143961 -0.240524230
##     Adjacent
## -0.074804832
```

Si se interpreta el coeficiente de regresión como que, por cada unidad que aumenta *Elevantion*, el número de especies se incrementa en promedio *0.3194* unidades, se estaría estableciendo una conclusión errónea. De hecho, si utilizando los mismos datos se ajusta un modelo más sencillo, el valor del coeficiente cambia.

```
modelo <- lm(Species ~ Elevation, data = gala)
coef(modelo)
```

```
## (Intercept)   Elevation
##  11.3351132   0.2007922
```

Esto ocurre porque el coeficiente de regresión de un determinado predictor siempre tiene que interpretarse en el contexto de los demás predictores incluidos en el modelo, puesto que influyen sobre él. La forma correcta es: El incremento en una unidad del predictor $X_i$

provoca un cambio promedio de $\hat{\beta}_i$ unidades en la variable respuesta, manteniéndose constantes el resto de predictores.

Con frecuencia, cuando las escalas en las que se miden los predictores son muy distintas en orden de magnitud, suelen estandarizarse los predictores previo ajuste del modelo. Si esto ocurre, la interpretación de los coeficientes de regresión pasa a ser: El incremento en una unidad estándar del predictor $X_i$ provoca un cambio promedio de $\hat{\beta}_i$ unidades estándar en la variable respuesta, manteniéndose constantes el resto de predictores.

En los estudios observacionales, raramente se pueden mantener constantes los predictores a voluntad del investigador, de ahí que sea difícil establecer relaciones de causalidad.

## Precaución al evaluar la normalidad de los residuos por contraste de hipótesis

*Linear Models with R, by Julian J. Faraway*

Que los residuos de un modelo de regresión lineal se distribuyan de forma normal es una condición necesaria para que la significancia (*p-value*) y los intervalos de confianza asociados a los predictores (calculados a partir de modelos teóricos) sean precisos. Con frecuencia, esta condición se evalúa con contrastes de hipótesis tales como el *Shapiro-Wilk test*. Cuando esto ocurre, es importante entender la relación entre *p-value* y tamaño de muestra. A mayor número de residuos mayor potencia tiene el test y, por lo tanto, pequeñas desviaciones de la normalidad resultan significativas. A su vez, el teorema del límite central indica que, cuanto mayor el tamaño de la muestra, más robustos son los resultados frente a desviaciones de la normalidad. Dadas estas propiedades, suele ser más recomendable evaluar la normalidad de los residuos de forma gráfica mediante representación de los cuantiles teóricos `qqplot()`.

## Robust Regression, Generalized Least Squares y Weighted Least Squares: Alternativas a mínimos cuadrados cuando no se cumplen las condiciones de los residuos

*Linear Models with R, by Julian J. Faraway*

El método de regresión por mínimos cuadrados ordinarios (OLS) descrito en los apartados anteriores asume que los residuos/errores son independientes, con varianza constante y que se distribuyen de forma normal. En la práctica, estas condiciones no se cumplen con frecuencia, lo que hace necesario disponer de métodos de ajuste alternativos. Algunos de ellos son: *Robust Regression*, *Generalized Least Squares* y *Weighted Least Squares*.

## Robust Regression

Cuando los errores no siguen una distribución normal, los resultados obtenidos por mínimos cuadrados se ven afectados, siendo mayor el impacto cuanto más largas son las colas. Una solución simple pasa por eliminar los valores atípicos (*outliers*) que forman dichas colas, sin embargo, de confirmarse que no son errores de lectura, el modelo debería incluirlos puesto que son parte del fenómeno que se quiere estudiar. *Robust regression* consigue reducir la influencia de los valores atípicos en el ajuste del modelo. Los dos tipos de *robust regression* más empleados son: *M-Estimation* y *Least Trimmed Squares*.

M-Estimation

Tiene diferentes variantes dependiendo de la función que se emplee para atenuar el peso de las observaciones extremas. `rlm()` del paquete `MASS` emplea el método de *Huber*. Por lo general, este método consigue disminuir la influencia de valores atípicos siempre y cuando sean pocos y de una magnitud no excesiva.

*El set de datos gala del paquete* `faraway` *contiene información sobre las islas que forman el archipiélago de las Galápago. Entre las variables almacenadas están: número de especies que habitan la isla (*Species*), el área (*Area*) y altura (*Elevation*) de la isla, la distancia a la isla más próxima (*Nearest*), la distancia a la isla Santa Cruz (*Scruz*) y el área de la isla adyacente (*Adjacent*). Se quiere obtener un modelo lineal que prediga el número de especies en función del resto de variables.*

Se ajusta primero el modelo por mínimos cuadrados ordinarios y a continuación por *robust regression*.

```
library(faraway)
library(MASS)
data(gala)
gala$Isla <- rownames(gala)
modelo_ols <- lm(Species ~ Area + Elevation + Nearest + Scruz + Adjacent, data =
gala)
summary(modelo_ols)
```

```
##
## Call:
## lm(formula = Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
##     data = gala)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -111.679  -34.898   -7.862   33.460  182.584
##
## Coefficients:
```

```
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.068221  19.154198   0.369 0.715351
## Area        -0.023938   0.022422  -1.068 0.296318
## Elevation    0.319465   0.053663   5.953 3.82e-06 ***
## Nearest      0.009144   1.054136   0.009 0.993151
## Scruz       -0.240524   0.215402  -1.117 0.275208
## Adjacent    -0.074805   0.017700  -4.226 0.000297 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 60.98 on 24 degrees of freedom
## Multiple R-squared:  0.7658, Adjusted R-squared:  0.7171
## F-statistic:  15.7 on 5 and 24 DF,  p-value: 6.838e-07
```

```
modelo_rob <- rlm(Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
                  data = gala)
summary(modelo_rob)
```

```
##
## Call: rlm(formula = Species ~ Area + Elevation + Nearest + Scruz +
##     Adjacent, data = gala)
## Residuals:
##     Min      1Q  Median      3Q     Max
## -74.389 -18.353  -6.364  21.187 229.082
##
## Coefficients:
##             Value   Std. Error t value
## (Intercept)  6.3611 12.3897     0.5134
## Area        -0.0061  0.0145    -0.4214
## Elevation    0.2476  0.0347     7.1320
## Nearest      0.3592  0.6819     0.5267
## Scruz       -0.1952  0.1393    -1.4013
## Adjacent    -0.0546  0.0114    -4.7648
##
## Residual standard error: 29.73 on 24 degrees of freedom
```

Se observan varias diferencias al comparar los modelos. En primer lugar, el `summary()` del modelo `rlm` no devuelve el valor $R^2$ ya que, en el contexto de regresión robusta, no existe esté valor. Tampoco se muestra el *p-value* de cada coeficiente de regresión, aunque puede obtenerse por inferencia empleando la *t-distribution*.

```
coeficientes <- as.data.frame(summary(modelo_rob)$coefficients)
coeficientes$p_value <- 2 * (1 - pt(q = abs(coeficientes$`t value`),
                                    df = nrow(gala) - 6))
coeficientes
```

```
##                    Value  Std. Error    t value      p_value
## (Intercept)  6.361060552 12.38970334  0.5134151 6.123553e-01
## Area        -0.006111214  0.01450368 -0.4213562 6.772448e-01
```

```
## Elevation    0.247562039  0.03471125  7.1320396 2.263516e-07
## Nearest      0.359155561  0.68185741  0.5267312 6.032151e-01
## Scruz       -0.195241588  0.13933081 -1.4012808 1.739294e-01
## Adjacent    -0.054553495  0.01144919 -4.7648339 7.551261e-05
```

*Elevation* y *Adjacent* se identifican como significativos en ambos modelos, aunque con regresión robusta el *Std.Error* de las estimaciones es menor.

Otra de las ventajas de la *robust regresion* es que se puede estudiar el *weight* asignado a cada observación para identificar cuáles se han considerado como más atípicas.

```
wts <- modelo_rob$w
names(wts) <- rownames(gala)
sort(wts)
```

```
##    SantaCruz   SantaMaria SanCristobal        Pinta     Gardner1
##    0.1745816    0.3078288    0.4142330    0.5375752    0.6613011
##     Espanola     Gardner2       Baltra    Bartolome     Caldwell
##    0.6795050    0.8500380    1.0000000    1.0000000    1.0000000
##     Champion      Coamano Daphne.Major Daphne.Minor       Darwin
##    1.0000000    1.0000000    1.0000000    1.0000000    1.0000000
##         Eden      Enderby   Fernandina     Genovesa      Isabela
##    1.0000000    1.0000000    1.0000000    1.0000000    1.0000000
##     Marchena       Onslow       Pinzon   Las.Plazas       Rabida
##    1.0000000    1.0000000    1.0000000    1.0000000    1.0000000
##  SanSalvador      SantaFe      Seymour      Tortuga         Wolf
##    1.0000000    1.0000000    1.0000000    1.0000000    1.0000000
```

Least Trimmed Squares

Este tipo de *robust regression* ajusta el modelo mediante mínimos cuadrados pero empleando únicamente los *q* residuos de menor tamaño, ignorando por completo el resto de observaciones. Es por lo tanto más independiente de los valores atípicos que *M-estimation*, aunque depende del valor *q* que se especifique. La función `ltsre()` del paquete `MASS` elige por defecto un valor *q* teniendo en cuenta el número de observaciones y predictores incluidos en el modelo.

```
set.seed(123)
modelo_lts <- ltsreg(Species ~ Area + Elevation + Nearest + Scruz + Adjacent,
                     data = gala)
modelo_lts
```

```
## Call:
## lqs.formula(formula = Species ~ Area + Elevation + Nearest +
##     Scruz + Adjacent, data = gala, method = "lts")
##
## Coefficients:
```

```
## (Intercept)         Area    Elevation      Nearest        Scruz
##    12.50668      1.54536      0.01673      0.52349     -0.09407
##    Adjacent
##    -0.14259
##
## Scale estimates 12.96 10.99
```

No existe un método teórico que permita estimar el error estándar de los coeficientes de regresión generados mediante *Least Trimmed Squares*, pero se puede recurrir al *bootstrappimg* para calcularlos.

```
matriz_coef <- matrix(0, 5000, 6)
for (i in 1:5000) {
    newy <- predict(modelo_lts) + residuals(modelo_lts)[sample(30, replace = TRUE)]
    boot_reg <- ltsreg(newy ~ Area + Elevation + Nearest + Scruz + Adjacent,
        data = gala, nsamp = "best")
    matriz_coef[i, ] <- boot_reg$coef
}
colnames(matriz_coef) <- names(coef(modelo_lts))
apply(matriz_coef, MARGIN = 2, FUN = function(x) {quantile(x, c(0.025, 0.975))})
```

```
##       (Intercept)     Area   Elevation    Nearest      Scruz    Adjacent
## 2.5%   -0.5840274 1.413842 -0.03756839 -0.4710066 -0.3324358 -0.19842818
## 97.5%  33.9728458 1.666508  0.11339379  2.6105862  0.1891972 -0.03306539
```

En la gran mayoría de ocasiones, los resultados obtenidos empleando *robust regression* son similares a los que se llega si se utiliza regresión por mínimos cuadrados y se analizan las observaciones atípicas o influyentes. Cuando este último análisis no se puede realizar, bien porque el investigador no sabe o bien porque se trata de un proceso automatizado, la *robust regression* es más segura.

Dada la rapidez con la que se pueden aplicar estos métodos de ajuste, se puede emplear *robust regression* como método de confirmación. Primero se ajusta el modelo empleando mínimos cuadrados y, si al compararlo con *robust regression* no hay grandes diferencias, significa que el modelo no contiene observaciones influyentes.

### Generalized Least Squares

Es una alternativa a la regresión por mínimos cuadrados cuando existe correlación entre los residuos. Esto ocurre con frecuencia en series temporales. La función `gls()` del paquete `nlme` permite realizar ajustes por *generalized least squares*.

### Weighted Least Squares

Es una alternativa a la regresión por mínimos cuadrados que no se ve afectada por que los residuos no tengan varianza constante (falta de homocedasticidad). Algunos escenarios en los que suele ser recomendable su uso son:

- Cuando los residuos son proporcionales a alguno de los predictores, es decir, a medida que aumenta el valor del predictor también lo hacen los residuos.
- Cuando la variable $Y_i$ se ha obtenido a partir de $n_i$ observaciones (por ejemplo la media de $n$ repeticiones), dado que $Var(y_i) = \sigma^2/n_i$, puede ocurrir que la varianza sea proporcional al tamaño del grupo.
- Cuando la variable respuesta procede de diferentes fuentes, por ejemplo diferentes instrumentos, cada una con una precisión distinta, se asigna a cada fuente $i$ un peso tal que $weight_i = 1/sd(y_i)$.

La función `lm()` permite especificar el peso mediante el argumento `weight`.

## Transformación de variables

Linear Models with R, by Julian J. Faraway y wikipedia

Transformar la variable respuesta o los predictores puede ser una forma de mejorar el ajuste de un modelo o corregir la violación de alguna de las condiciones de regresión. Cuando la transformación se aplica a la variable respuesta, es importante recordar que las predicciones generadas por el modelo están en escala transformada. Una vez calculada la predicción, se puede transformar de vuelta a la escala original. Esto mismo aplica a los intervalos de confianza, primero se calculan los límites (superior e inferior) y luego se convierten de vuelta. En contraposición, los coeficientes de regresión obtenidos para los predictores se tienen que interpretar en el contexto de la transformación, no es posible aplicarles la inversa de la transformación e interpretarlos en la escala original.

### Transformación de Box-Cox

La transformación exponencial de la variable respuesta ($y^\lambda$) es con frecuencia una solución útil para mejorar los modelos, sin embargo, identificar el valor óptimo del exponente no es siempre inmediato. El método de *Box-Cox* permite encontrar si existe algún valor al que se pueda elevar la variable respuesta para mejorar el ajuste del modelo y, en tal caso, identificar cual es. Acorde a este método, se busca una transformación $y \rightarrow g_\lambda(y)$ tal que:

$$g_\lambda(y) = \begin{cases} \dfrac{y^\lambda - 1}{y} & \text{si } \lambda \neq 0 \\ log(y) & \text{si } \lambda = 0 \end{cases}$$

Cuando el objetivo del modelo es predictivo, se puede simplificar $\frac{y^\lambda - 1}{y}$ por $y^\lambda$.

El valor de $\lambda$ óptimo se identifica por máxima verosimilitud: se selecciona un intervalo de valores de $\lambda$, para cada valor se realiza la transformación y se calculan los cuadrados de los residuos. Aquel que tenga el menor valor de la suma de residuales será la mejor opción.

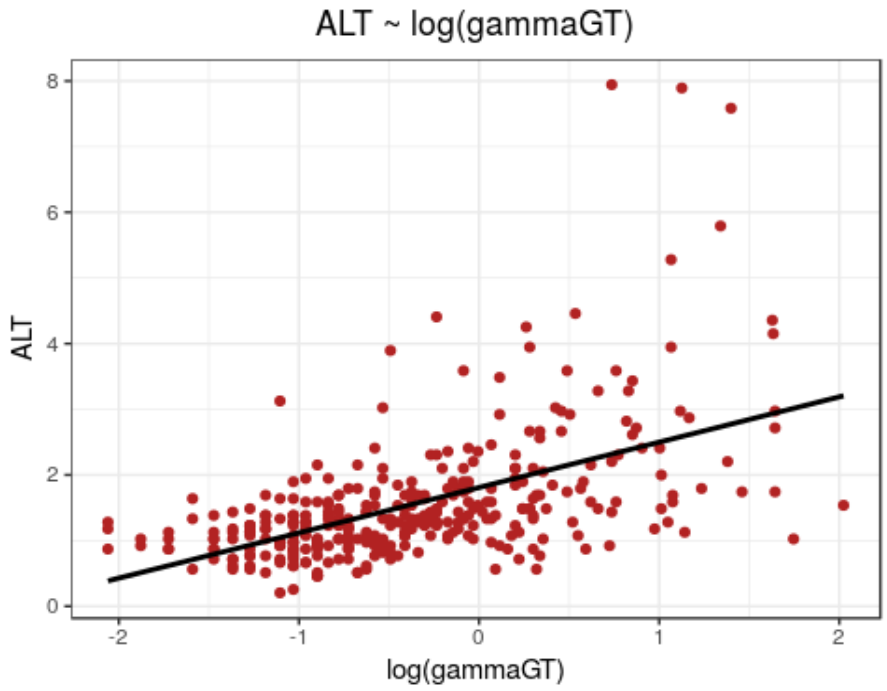
La transformación de la variable respuesta puede hacer que el modelo sea más difícil de interpretar, por lo que no es aconsejable a no ser que sea realmente necesaria. Para asegurarse de que lo es, además de calcular el valor óptimo de $\lambda$, se puede estimar su intervalo de confianza del 95%. Si este incluye el valor $\lambda = 1$ no hay evidencias suficientes a favor de la transformación. Cuando la transformación sí es necesaria, se aconseja elegir dentro del intervalo de confianza, aquel valor que facilite la interpretación del modelo. Por ejemplo, si el intervalo es [-0.7, 0.7], el valor 0.5 es el más adecuado ya que se corresponde con $\sqrt{(y)}$.

La trasformación *Box-Cox* solo es aplicable cuando la variable respuesta toma siempre valores positivos. Si algún $y_i < 0$, se puede sumar una constante a $y$ para que sea positiva, pero es una solución poco elegante. Otra característica de este tipo de transformación es que se ve altamente influenciada por *outliers*, haciendo que los valores absolutos de $\lambda$ sean sospechosamente altos.

La función `boxcox()` del paquete `MASS` realiza la transformación *Box-Cox*.

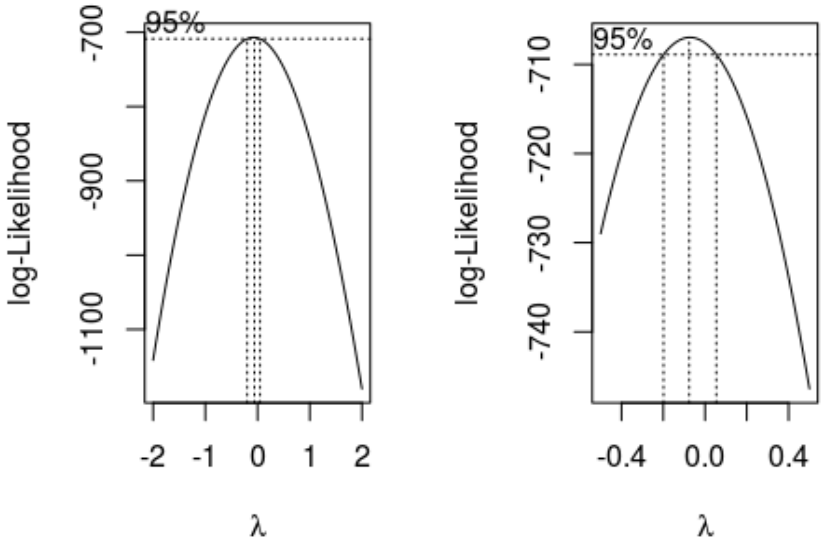
*El set de datos BUPA contiene información sobre los niveles en sangre de diferentes enzimas que se cree que están asociadas con enfermedades del hígado. Se quiere generar un modelo que permita predecir la concentración de ALT (V3) a partir del logaritmo de la concentración de γGT (V5).*

```
library(MASS)
library(aucm)
library(ggplot2)
data("bupa")
colnames(bupa)[colnames(bupa) == "V3"] <- "ALT"
colnames(bupa)[colnames(bupa) == "V5"] <- "gammaGT"

ggplot(data = bupa, aes(x = log(gammaGT), y = ALT)) +
 geom_point(color = "firebrick") +
 geom_smooth(method = "lm", se = FALSE, color = "black") +
 labs(title = "ALT ~ log(gammaGT)") +
 theme_bw() + theme(plot.title = element_text(hjust = 0.5))
```



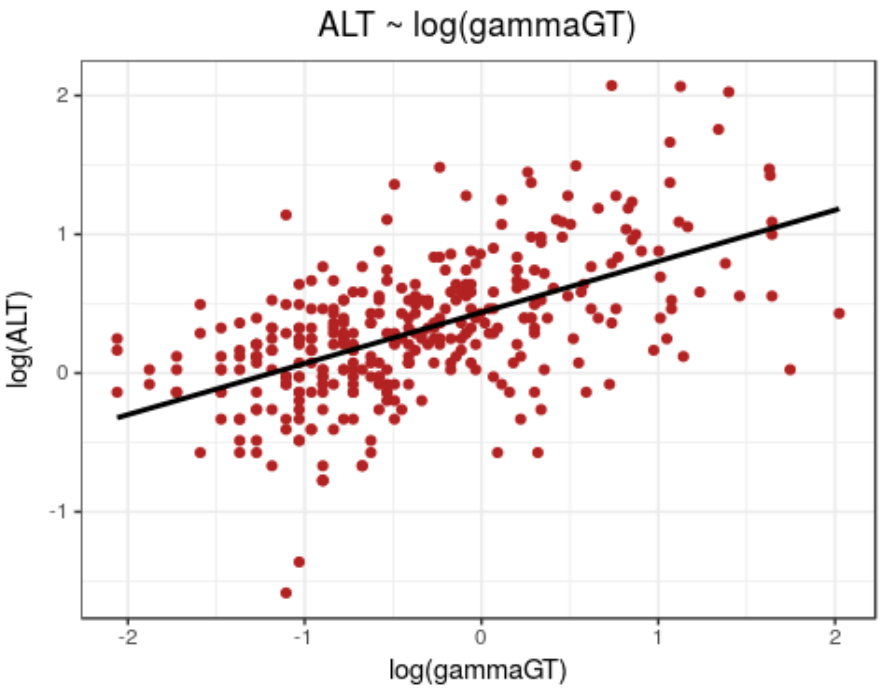

La representación gráfica muestra que, aunque la relación sí parece lineal, la varianza no es constante. Una transformación *Box-Cox* podría mejorar la condición de homocedasticidad.

```
par(mfrow = c(1, 2))
boxcox(ALT ~ log(gammaGT), lambda = -2:2, data = bupa)
# Se repite el proceso pero esta vez entrechando el rango de valores de lambda
boxcox(ALT ~ log(gammaGT), lambda = seq(-0.5, 0.5, 0.1), data = bupa)
```



El intervalo de confianza del 95% contiene el valor $\lambda = 0$, lo que sugiere que la transformación $y = log(y)$ podría mejorar el modelo.

```
ggplot(data = bupa, aes(x = log(gammaGT), y = log(ALT))) +
 geom_point(color = "firebrick") +
 geom_smooth(method = "lm", se = FALSE, color = "black") +
 labs(title = "ALT ~ log(gammaGT)") +
 theme_bw() + theme(plot.title = element_text(hjust = 0.5))
```




### Transformación logarítmica

Existen múltiples alternativas a la transformación exponencial *Box-Cox*. Una de ellas consiste en sumarle una constante a la variable respuesta y aplicarle logaritmos, $g_\alpha = log(y + \alpha)$. La función `logtrans()` del paquete `MASS` identifica el valor óptimo de $\alpha$ para esta transformación.

## Relación entre modelos lineales con un predictor cualitativo y el ANOVA

*Linear Models with R, by Julian J. Faraway*

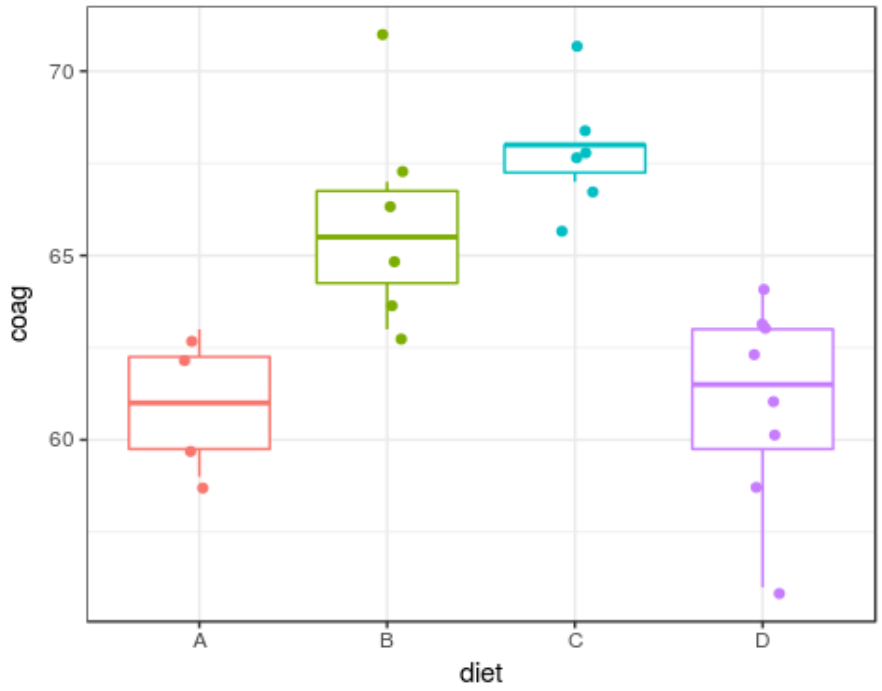
Tradicionalmente, el estudio de la influencia que tiene una variable cualitativa con más de dos niveles sobre una variable respuesta continua se hace mediante un análisis de varianza ANOVA: análisis de varianza para comparar múltiples medias. Existe la posibilidad de abordar este mismo estudio desde la perspectiva de un modelo de regresión lineal.

El set de datos `coagulation` del paquete `faraway` contiene los resultados de un experimento en el que 24 animales fueron asignados aleatoriamente a 4 grupos, cada uno de los cuales se alimentó con una dieta distinta. Se extrajeron muestras de sangre de cada uno de los animales y se midió el tiempo de coagulación. El objetivo del estudio era determinar si la dieta influye en la capacidad de coagulación.

```
library(faraway)
library(ggplot2)
library(dplyr)
data("coagulation")
head(coagulation)
```

```
##   coag diet
## 1   62    A
## 2   60    A
## 3   63    A
## 4   59    A
## 5   63    B
## 6   67    B
```

```
ggplot(data = coagulation, aes(x = diet, y = coag, color = diet)) +
geom_boxplot(outlier.shape = NA) +
geom_jitter(width = 0.1) +
theme_bw() +
theme(legend.position = "none")
```




```
# cálculo de la media y sd de cada grupo
coagulation %>% group_by(diet) %>% summarise(media = mean(coag), sd = sd(coag))
```

```
## # A tibble: 4 x 3
##     diet media       sd
##   <fctr> <dbl>    <dbl>
## 1      A    61 1.825742
## 2      B    66 2.828427
## 3      C    68 1.673320
## 4      D    61 2.618615
```

```
# test de normalidad para cada grupo
coagulation %>% group_by(diet) %>% summarise(shapiro_test =
                                               shapiro.test(coag)$p.value)
```

```
## # A tibble: 4 x 2
##     diet shapiro_test
##   <fctr>        <dbl>
## 1      A    0.7142802
## 2      B    0.5227052
## 3      C    0.2375366
## 4      D    0.5319098
```

La representación gráfica de los datos, junto con el cálculo de las medias, muestran evidencias de posibles diferencias significativas entre dietas. Las desviaciones típicas de los grupos son similares, por lo que se puede asumir homocedasticidad. No hay evidencia de falta de normalidad.

```
modelo <- lm(coag ~ diet, coagulation)
summary(modelo)
```

```
##
## Call:
## lm(formula = coag ~ diet, data = coagulation)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -5.00  -1.25   0.00   1.25   5.00
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 6.100e+01  1.183e+00  51.554  < 2e-16 ***
## dietB       5.000e+00  1.528e+00   3.273 0.003803 **
## dietC       7.000e+00  1.528e+00   4.583 0.000181 ***
## dietD       2.991e-15  1.449e+00   0.000 1.000000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.366 on 20 degrees of freedom
## Multiple R-squared:  0.6706, Adjusted R-squared:  0.6212
## F-statistic: 13.57 on 3 and 20 DF,  p-value: 4.658e-05
```

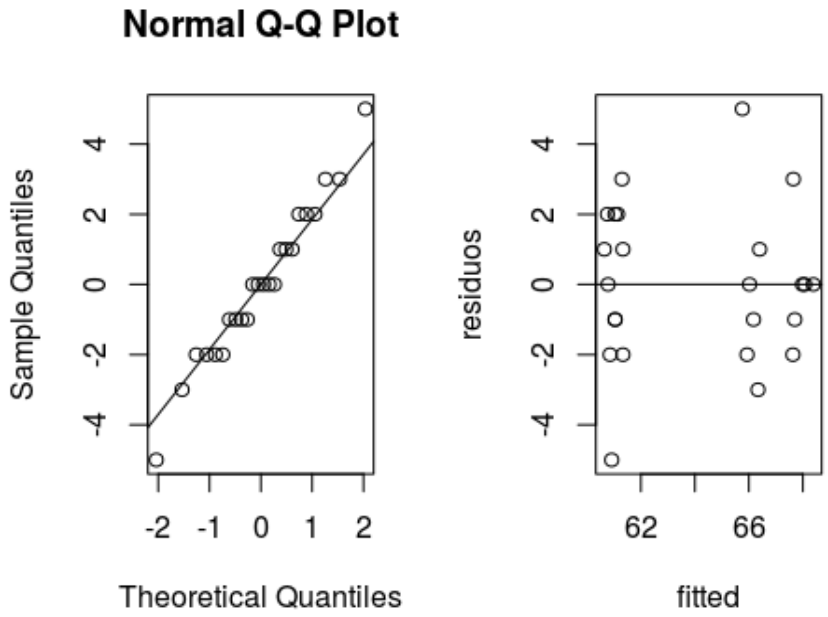
El valor estimado de los coeficientes se corresponde con la diferencia entre la media de cada nivel y la media del nivel de referencia (que viene dada por el intercept y que en este caso es *dietA*). Por ejemplo, la media del grupo *dietB* es de $61 + 5 = 66$. El *p-value* resultante del *F-test* (4.658e-05) indica que hay evidencias de que al menos un grupo difiere significativamente del resto. Este *p-value* es el resultado de comparar el modelo completo respecto al modelo sin predictores.

```
modelo <- lm(coag ~ diet, coagulation)
modelo_nulo <- lm(coag ~ 1, coagulation)
anova(modelo_nulo, modelo)
```

```
## Analysis of Variance Table
##
## Model 1: coag ~ 1
## Model 2: coag ~ diet
##   Res.Df RSS Df Sum of Sq      F    Pr(>F)
## 1     23 340
## 2     20 112  3       228 13.571 4.658e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Al igual que ocurre con la aproximación de ANOVA, la aproximación por regresión lineal también necesita que se cumplan las condiciones para ser válida.

```
par(mfrow = c(1, 2))
qqnorm(modelo$residuals)
qqline(modelo$residuals)
plot(x = jitter(modelo$fitted.values), y = modelo$residuals, xlab = "fitted",
     ylab = "residuos")
abline(h = 0)
```




Dado que la variable respuesta son números enteros, la representación gráfica muestra cierta discontinuidad, aun así, la distribución se aproxima a la normal.

```
bartlett.test(coag ~ diet, coagulation)
```

```
##
##  Bartlett test of homogeneity of variances
##
## data:  coag by diet
## Bartlett's K-squared = 1.668, df = 3, p-value = 0.6441
```

Tampoco hay evidencias de falta de homocedasticidad.

Como resultado del análisis se puede concluir que existen evidencias suficientes para afirmar que al menos uno de los grupos tiene una media distinta, por lo que la dieta sí influye en la velocidad de coagulación. Para conocer cuál de ellos es, sería necesario recurrir a las comparaciones múltiples (no visto aquí).

## Tobit Regression: modelos lineales para datos censurados

Tal y como se ha visto en los apartados anteriores, el método de regresión por mínimos cuadrados ordinarios (OLS) es muy útil para estudiar una variable respuesta continua en función de uno o más predictores. Sin embargo, bajo determinadas circunstancias y a pesar de existir una relación lineal entre las variables, el método de mínimos cuadrados ordinarios no es adecuado.

**Datos censurados:** Se considera que los datos están censurados (*censored*) cuando existe un determinado límite en la variable respuesta (superior, inferior o ambos) a partir del cual a todas las observaciones se les asigna un mismo valor. Algunos ejemplos de datos censurados son:

- Un instrumento de medida, por ejemplo una balanza, tiene un límite de detección por debajo del cual todo valor se considera de 0 (censura inferior).
- En una encuesta se pregunta por el nivel de ingresos de las personas. Se divide la escala en múltiples intervalos, el último de los cuales, contempla como iguales a todas aquellas personas que cobren 5000 euros o más (censura superior).

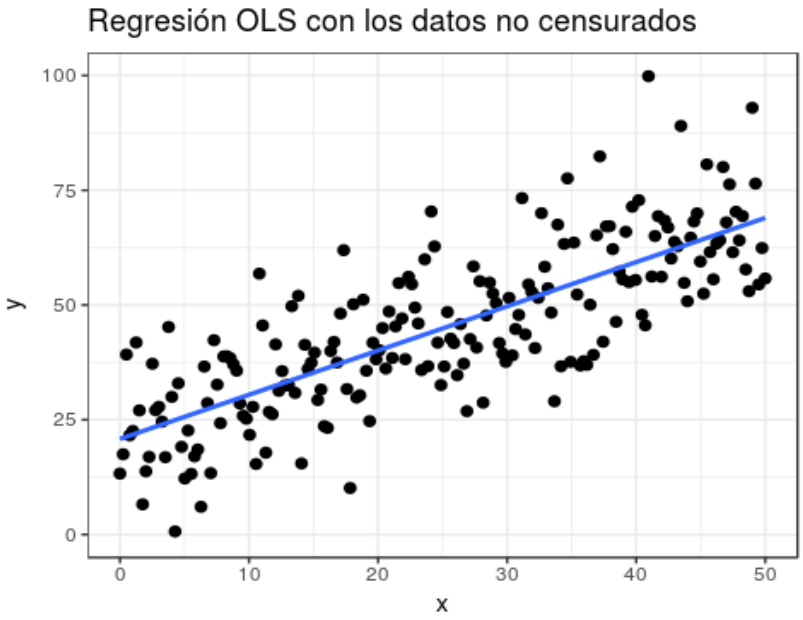
La característica fundamental de un escenario censurado es que hay una población subyacente en la que sí existen observaciones fuera de los límites de censura, sin embargo, debido a la incapacidad para detectarlas/seleccionarlas en el muestreo, la población observada parece no contenerlas.

**Datos truncados:** Las situaciones con datos truncados aparecen cuando en una población a partir de un determinado límite no existen observaciones. La diferencia fundamental respecto a los datos censurados es que, en estos últimos, las observaciones sí existen en la población latente pero no se pueden captar en el muestreo.

Aunque la diferencia puede parecer mínima, es muy importante tenerla en cuenta puesto que, si el objetivo último de la inferencia es obtener información sobre la población real, en el caso de escenarios censurados, hay que incluir de alguna forma esos eventos que existen pero no se observan. Este documento se centra en el tratamiento de casos censurados.

Las siguientes imágenes muestran una representación gráfica e intuitiva del impacto que tienen los datos censurados sobre la regresión por mínimos cuadrados. Supónganse dos variables $X$ e $Y$ que tienen una relación lineal. (Para poder simular los datos se considera que la relación sigue la ecuación $Y = X + 2 + error$).

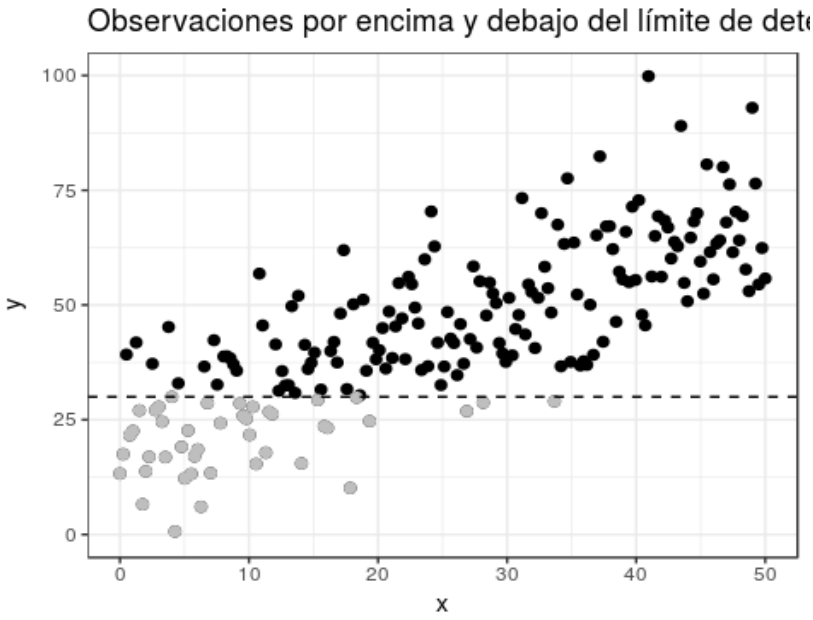
```
library(ggplot2)
library(dplyr)
library(gridExtra)
simulador <- function(x) {
    x + 20
}
observaciones <- simulador(seq(0, 50, length.out = 200))
# Para introducir la variancia propia de los datos observacionales se añade
# ruido aleatorio y normal a cada observación
set.seed(123)
observaciones <- observaciones + rnorm(n = 200, mean = 0, sd = 12)

datos <- data.frame(x = seq(0, 50, length.out = 200), y = observaciones)
ggplot(data = datos, aes(x = x, y = y)) +
geom_point(size = 2) +
geom_smooth(method = "lm", se = FALSE) +
theme_bw() +
lims(y = c(0, 100)) +
labs(title = "Regresión OLS con los datos no censurados")
```




Supóngase ahora que los investigadores no son capaces de cuantificar valores de la variable $Y$ inferiores a 30, a pesar de que sí que son capaces de detectarlos y por lo tanto saben que existen. La muestra observada pasaría a ser la siguiente:

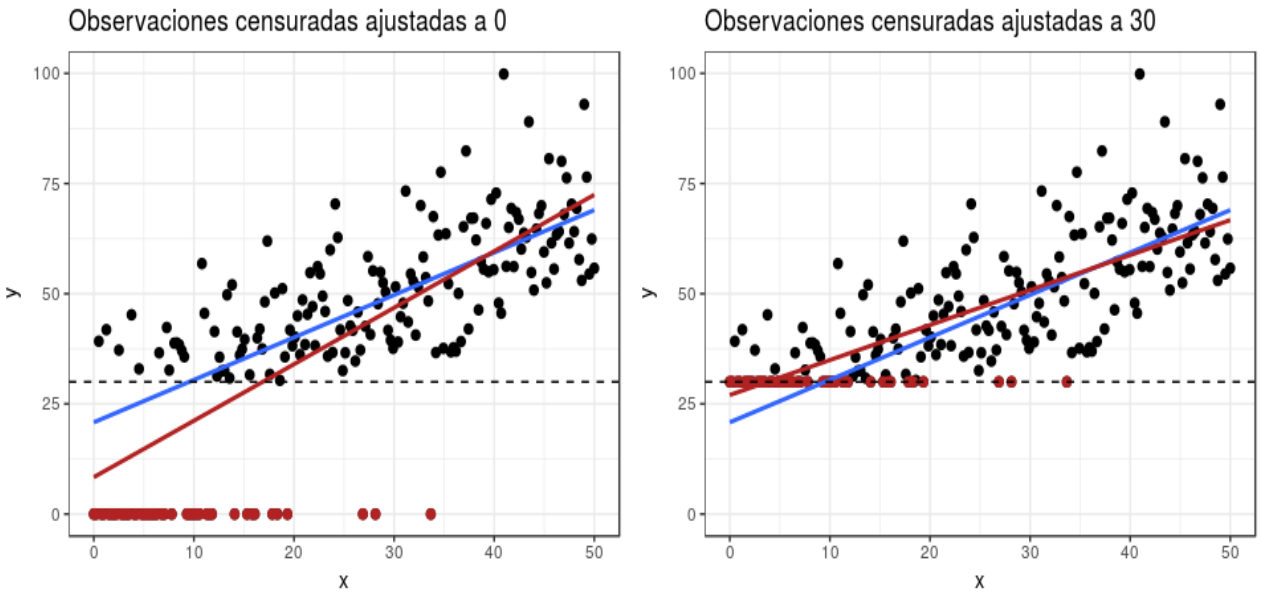
```
ggplot(data = datos, aes(x = x, y = y)) +
geom_point(size = 2) +
geom_point(data = filter(datos, y < 30), size = 2, color = "grey") +
geom_hline(yintercept = 30, linetype = "dashed") +
theme_bw() +
lims(y = c(0, 100)) +
```

```
labs(title = "Observaciones por encima y debajo del límite de detección")
```




Dada esta situación, los investigadores tienen 2 opciones: considerar todos los valores por debajo del límite de detección como 0 o como 30. Cuál de las dos opciones es más correcta depende del tipo de estudio que se esté haciendo. En las dos imágenes siguientes se puede ver cómo impacta cada tipo de censura al ajuste por mínimos cuadrados.

```
datos_0 <- datos
datos_0$y[datos_0$y < 30] <- 0
p1 <- ggplot(data = datos_0, aes(x = x, y = y)) +
      geom_point(size = 2) +
      geom_point(data = filter(datos_0, y == 0), size = 2, color = "firebrick") +
      geom_smooth(data = datos, method = "lm", se = FALSE) +
      geom_smooth(data = datos_0, method = "lm", se = FALSE, color = "firebrick") +
      geom_hline(yintercept = 30, linetype = "dashed") +
      lims(y = c(0, 100)) +
      theme_bw() +
      labs(title = "Observaciones censuradas ajustadas a 0")

datos_30 <- datos
datos_30$y[datos_30$y < 30] <- 30
p2 <- ggplot(data = datos_30, aes(x = x, y = y)) +
      geom_point(size = 2) +
      geom_point(data = filter(datos_30, y == 30), size = 2, color = "firebrick") +
      geom_smooth(data = datos, method = "lm", se = FALSE) +
      geom_smooth(data = datos_30, method = "lm", se = FALSE, color= "firebrick") +
      geom_hline(yintercept = 30, linetype = "dashed") +
      theme_bw() + lims(y = c(0, 100)) +
      labs(title = "Observaciones censuradas ajustadas a 30")
grid.arrange(p1, p2, ncol = 2)
```




En el primer caso (imagen izquierda), la recta de regresión aumenta su inclinación debido a la influencia que ejercen todas aquellas observaciones que antes tenían valores entre 0 y 30 y que ahora son exactamente 0. En el segundo caso (imagen derecha) el efecto es el contrario, ya que muchas observaciones han pasado a tener el valor de 30 cuando antes eran inferiores.

Además del evidente impacto sobre la pendiente de la recta de regresión, se unen otros dos factores que invalidan la utilización de mínimos cuadrado OLS con datos censurados. A medida que se aproxima el límite de censura y más observaciones son fijadas al valor establecido, la varianza se reduce, perdiéndose así la condición de homocedasticidad (varianza constante de los residuos) y de independencia. Ambas necesarias para considerar válido el método de regresión por mínimos cuadrados.

**Modelo de Tobit**

La regresión de Tobit (1958) considera que existe una variable latente $Y^*$ no observable y una variable $Y$ observable, formada por la parte no censurada de $Y^*$. El objetivo es ser capaz de estimar parámetros de $Y^*$ empleando solo una muestra de la parte observable. Acorde a esto, asumiendo censura por el límite inferior (misma idea para la superior), Tobit considera que el valor esperado de la variable censurada $Y$ puede definirse como:

$$E(y) = [P(No\ censurado)\ x\ E(y|y > \tau)] + [P(censurado)\ x\ E(y|y = \tau_y)]$$

$$y_i = \begin{cases} y_i^* & \text{if } y_i^* > \tau \\ \tau_y & \text{if } y_i^* \leq \tau \end{cases}$$

donde $P$ es probabilidad, $\tau$ límite de censura, $\tau_y$ el valor que se le asigna a la variable latente $Y*$ cuando se le aplica la censura, $y|y > \tau$ y $y|y > \tau_y$ son el condicional de que $Y$ sea mayor que el límite de censura y el valor asignado a la censura respectivamente.

Lo que hace útil al método de Tobit es que permite hacer estimaciones de $Y^*$ a pesar de tener observaciones únicamente de $Y$.

**Ejemplo**

*Considérese un estudio en el que se pretende crear un modelo que permita predecir el nivel académico (escala 200-800) de los estudiantes de una universidad. Para ello se emplea la nota obtenida en un examen de lectura, un examen de matemáticas y el tipo de programa al que están matriculados (académico, general o vocacional).*

Se trata de un modelo censurado ya que todos aquellos estudiantes que contesten correctamente todas las preguntas obtendrán un 800, a pesar de que no todos ellos tengan exactamente el mismo nivel académico. Lo mismo ocurre para los estudiantes que contesten mal todas las preguntas, todos obtendrán un 200.

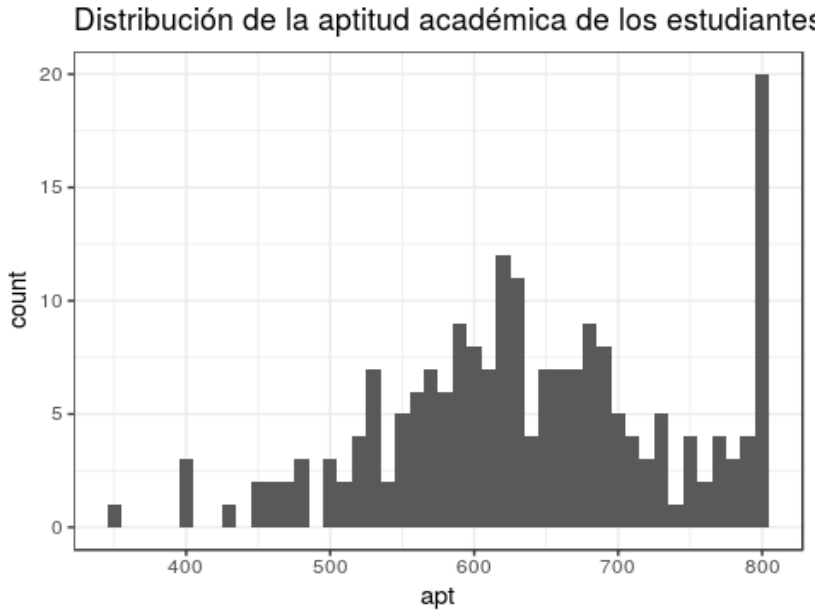
```
library(ggplot2)
datos <- read.csv("https://stats.idre.ucla.edu/stat/data/tobit.csv")
head(datos)
```

```
##   id read math       prog apt
## 1  1   34   40 vocational 352
## 2  2   39   33 vocational 449
## 3  3   63   48    general 648
## 4  4   44   41    general 501
## 5  5   47   43    general 762
## 6  6   47   46    general 658
```

```
summary(datos)
```

```
##        id              read            math              prog
##  Min.   :  1.00   Min.   :28.00   Min.   :33.00   academic  : 45
##  1st Qu.: 50.75   1st Qu.:44.00   1st Qu.:45.00   general   :105
##  Median :100.50   Median :50.00   Median :52.00   vocational: 50
##  Mean   :100.50   Mean   :52.23   Mean   :52.65
##  3rd Qu.:150.25   3rd Qu.:60.00   3rd Qu.:59.00
##  Max.   :200.00   Max.   :76.00   Max.   :75.00
##       apt
##  Min.   :352.0
##  1st Qu.:575.5
##  Median :633.0
##  Mean   :640.0
##  3rd Qu.:705.2
##  Max.   :800.0
```

```
ggplot(data = datos, aes(x = apt)) + geom_histogram(binwidth = 10) + theme_bw() +
labs(title = "Distribución de la aptitud académica de los estudiantes")
```




La exploración de los datos muestra que la puntuación mínima obtenida es de 352, por lo que, aun siendo posible la censura por el límite inferior, no ocurre en este set de datos. En el extremo superior de la distribución sí se observa incremento de eventos con una puntuación de 800 muy por encima de lo esperado acorde a una distribución normal (la que suele seguir este tipo de puntuaciones), lo que pone de manifiesto que sí hay censura.

Al conocerse la naturaleza de los datos, se sabe que se trata de datos censurados y no de datos truncados. A pesar de ello, una pista que ayuda a diferenciar ambos escenarios es que, cuando se trata de datos truncados en los que no existen observaciones más allá de un límite, no hay acumulación en el límite, simplemente se corta la distribución.

La función `vglm()` del paquete `VGAM` permite ajustar modelos mediante el método de Tobit.

```
library(VGAM)
modelo_tobit <- vglm(apt ~ read + math + prog, tobit(Upper = 800), data = datos)
summary(modelo_tobit)
```

```
##
## Call:
## vglm(formula = apt ~ read + math + prog, family = tobit(Upper = 800),
##     data = datos)
##
##
## Pearson residuals:
##               Min      1Q   Median     3Q   Max
## mu       -2.5684 -0.7311 -0.03976 0.7531 2.802
```

```
## loge(sd) -0.9689 -0.6359 -0.33365 0.2364 4.845
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept):1  209.55956   32.54590   6.439 1.20e-10 ***
## (Intercept):2    4.18476    0.05235  79.944  < 2e-16 ***
## read             2.69796    0.61928   4.357 1.32e-05 ***
## math             5.91460    0.70539   8.385  < 2e-16 ***
## proggeneral    -12.71458   12.40857  -1.025 0.305523
## progvocational -46.14327   13.70667  -3.366 0.000761 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Number of linear predictors:  2
##
## Names of linear predictors: mu, loge(sd)
##
## Log-likelihood: -1041.063 on 394 degrees of freedom
##
## Number of iterations: 5
```

Los coeficientes de regresión devueltos por un modelo Tobit se interpretan igual que los devueltos por un modelo de mínimos cuadrados OLS, pero referidos a la variable latente NO censurada.

La función `censReg()` del paquete `censreg` es otra de las múltiples implementaciones en R de modelos de regresión censurados.

```
library(censReg)
modelo_tobit <- censReg(apt ~ read + math + prog, right = 800, data = datos)
summary(modelo_tobit)
```

```
##
## Call:
## censReg(formula = apt ~ read + math + prog, right = 800, data = datos)
##
## Observations:
##          Total  Left-censored     Uncensored Right-censored
##            200              0            183              17
##
## Coefficients:
##                  Estimate Std. error t value  Pr(> t)
## (Intercept)    209.56597   32.77196   6.395 1.61e-10 ***
## read             2.69794    0.61881   4.360 1.30e-05 ***
## math             5.91448    0.70982   8.332  < 2e-16 ***
## proggeneral    -12.71476   12.40646  -1.025 0.305434
## progvocational -46.14390   13.72419  -3.362 0.000773 ***
## logSigma         4.18474    0.05301  78.946  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Newton-Raphson maximisation, 8 iterations
## Return code 1: gradient close to zero
## Log-likelihood: -1041.063 on 6 Df
```

Al comparar los resultados obtenidos por censReg() y vglm() parece haber pequeñas diferencias. Posiblemente se deban a que emplean diferentes métodos para maximizar la función de verosimilitud.

**Referencias**

https://menghublog.wordpress.com/2014/12/28/the-use-of-tobit-and-truncated-regressions-for-limited-dependent-variables/

https://stats.idre.ucla.edu/r/dae/tobit-models/

*Estimating Censored Regression Models in R using the censReg Package, Arne Henningsen University of Copenhagen*

# Bibliografía

*OpenIntro Statistics: Third Edition, David M Diez, Christopher D Barr, Mine Çetinkaya-Rundel*

*An Introduction to Statistical Learning: with Applications in R (Springer Texts in Statistics)*

*Linear Models with R, Julian J.Faraway*

*Points of Significance: Simple linear regression by Naomi Altman & Martin Krzywinski*

*Points of Significance: Multiple linear regression Martin Krzywinski & Naomi Altman*

*Métodos estadísticos en ingenieria Rafael Romero Villafranca, Luisa Rosa Zúnica Ramajo*

*Handbook of Biological Statistics*

*R Tutorials by William B. King, Ph.D* http://ww2.coastal.edu/kingw/statistics/R-tutorials/