Ejemplo práctico de regresión lineal simple, múltiple, polinomial e interacción entre predictores

Joaquín Amat Rodrigo *j.amatrodrigo@gmail.com*

Agosto, 2016

Índice

Introducción

Los siguientes ejemplos de regresión simple y múltiple se han obtenido del libro *Introduction to Statistical Learning*. El objetivo es mostrar los principales comandos en R para generar modelos lineales. Para obtener un modelo final robusto se tiene que analizar con más detalle cada una de las condiciones que se requieren para estos métodos. Para ver los conceptos teóricos de la regresión lineal múltiple consultar capítulo Introducción a la Regresión Lineal Múltiple. También se incluyen soluciones personales (validadas con otras en la web) de ejercicios propuestos en el libro.

Se van a emplear funciones contenidas en el paquete *MASS* y datos del paquete *ISL*

```
require(MASS)
require(ISLR)
data("Boston")
```

El *dataset* Boston del paquete MASS recoge la mediana del valor de la vivienda en 506 áreas residenciales de Boston. Junto con el precio, se han registrado 13 variables adicionales.

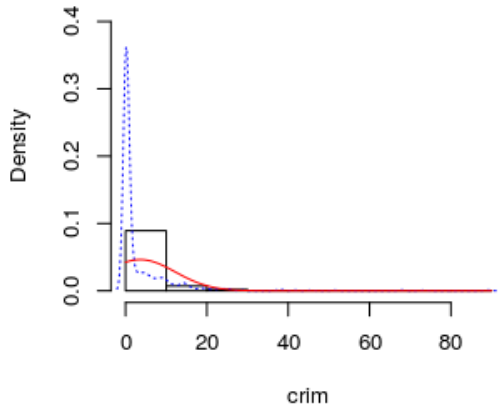
- crim: ratio de criminalidad per cápita de cada ciudad.
- zn: Proporción de zonas residenciales con edificaciones de más de 25.000 pies cuadrados.
- indus: proporción de zona industrializada.
- chas: Si hay río en la ciudad (= 1 si hay río; 0 no hay).
- nox: Concentración de óxidos de nitrógeno (partes per 10 millón).
- rm: promedio de habitaciones por vivienda.
- age: Proporción de viviendas ocupadas por el propietario construidas antes de 1940.
- dis: Media ponderada de la distancias a cinco centros de empleo de Boston.
- rad: Índice de accesibilidad a las autopistas radiales.
- tax: Tasa de impuesto a la propiedad en unidades de $10,000.
- ptratio: ratio de alumnos/profesor por ciudad.
- black: 1000(Bk - 0.63)^2 donde Bk es la proporción de gente de color por ciudad.
- lstat: porcentaje de población en condición de pobreza.
- medv: Valor mediano de las casas ocupadas por el dueño en unidades de $1000s.

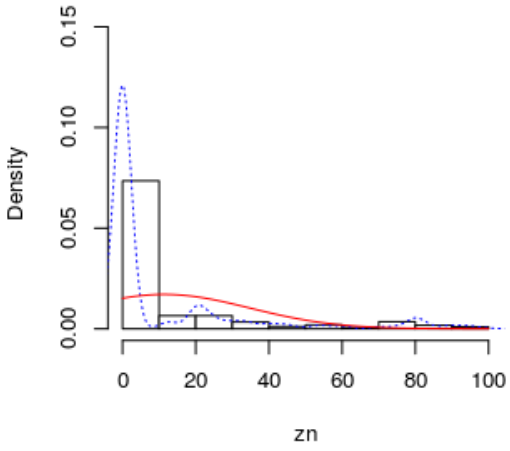
En primer lugar se realiza un análisis básico de los datos de forma numérica y gráfica.

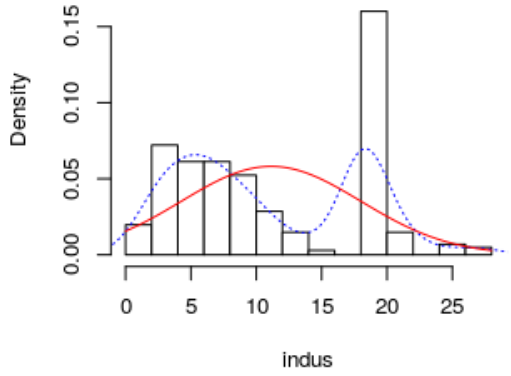
```
require(psych)
# La variable chas es una variable categórica por lo que se transforma a factor
Boston$chas <- as.factor(Boston$chas)
summary(Boston)
```

```
##       crim                 zn               indus        chas
##  Min.   : 0.00632   Min.   :  0.00   Min.   : 0.46   0:471
##  1st Qu.: 0.08204   1st Qu.:  0.00   1st Qu.: 5.19   1: 35
##  Median : 0.25651   Median :  0.00   Median : 9.69
##  Mean   : 3.61352   Mean   : 11.36   Mean   :11.14
##  3rd Qu.: 3.67708   3rd Qu.: 12.50   3rd Qu.:18.10
##  Max.   :88.97620   Max.   :100.00   Max.   :27.74
##       nox               rm              age               dis
##  Min.   :0.3850   Min.   :3.561   Min.   :  2.90   Min.   : 1.130
##  1st Qu.:0.4490   1st Qu.:5.886   1st Qu.: 45.02   1st Qu.: 2.100
##  Median :0.5380   Median :6.208   Median : 77.50   Median : 3.207
##  Mean   :0.5547   Mean   :6.285   Mean   : 68.57   Mean   : 3.795
##  3rd Qu.:0.6240   3rd Qu.:6.623   3rd Qu.: 94.08   3rd Qu.: 5.188
##  Max.   :0.8710   Max.   :8.780   Max.   :100.00   Max.   :12.127
##       rad              tax           ptratio          black
##  Min.   : 1.000   Min.   :187.0   Min.   :12.60   Min.   :  0.32
##  1st Qu.: 4.000   1st Qu.:279.0   1st Qu.:17.40   1st Qu.:375.38
##  Median : 5.000   Median :330.0   Median :19.05   Median :391.44
##  Mean   : 9.549   Mean   :408.2   Mean   :18.46   Mean   :356.67
##  3rd Qu.:24.000   3rd Qu.:666.0   3rd Qu.:20.20   3rd Qu.:396.23
##  Max.   :24.000   Max.   :711.0   Max.   :22.00   Max.   :396.90
##      lstat            medv
##  Min.   : 1.73   Min.   : 5.00
##  1st Qu.: 6.95   1st Qu.:17.02
##  Median :11.36   Median :21.20
##  Mean   :12.65   Mean   :22.53
##  3rd Qu.:16.95   3rd Qu.:25.00
##  Max.   :37.97   Max.   :50.00
```

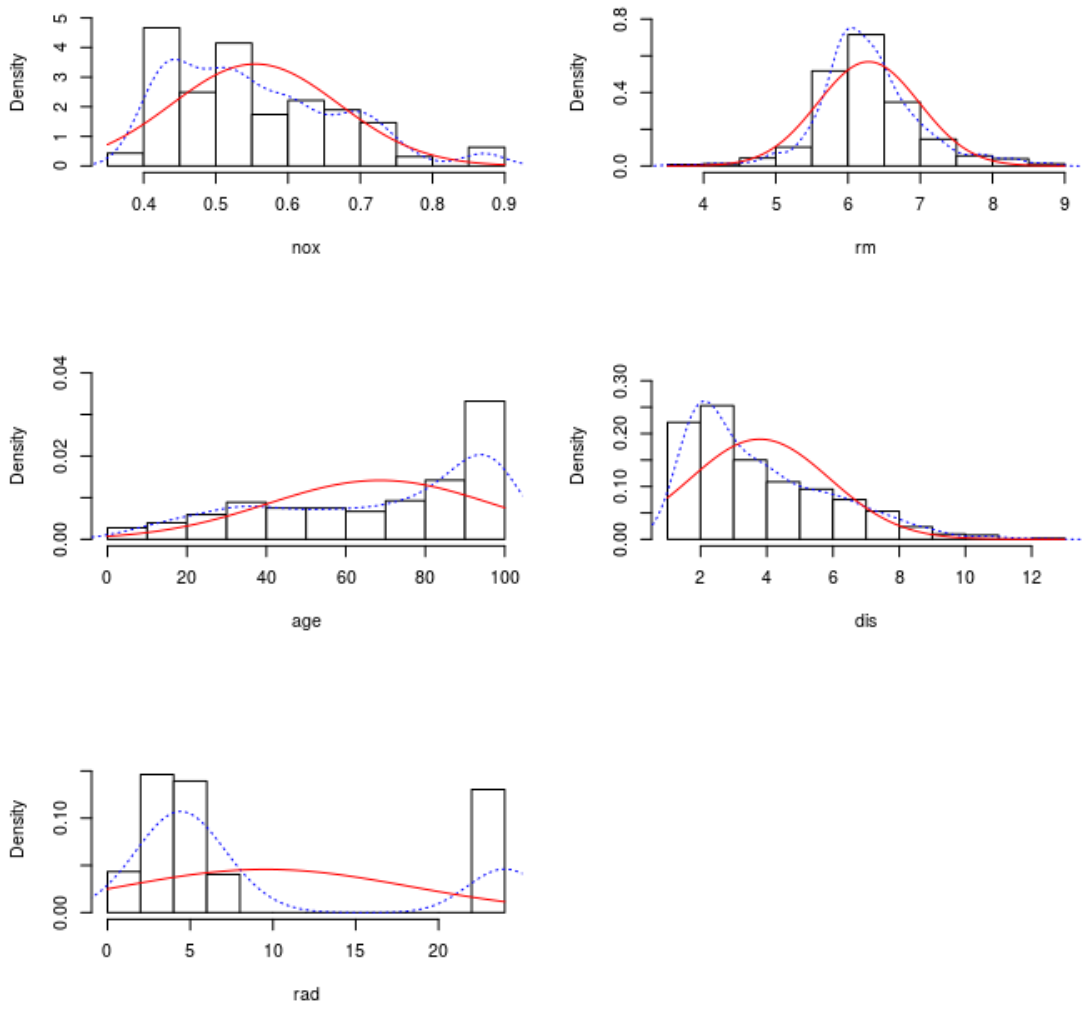
```
# Dado que hay muchas variables, se grafican por grupos de 4, excluyendo las
# categóricas
multi.hist(x = Boston[, 1:3], dcol = c("blue", "red"), dlty = c("dotted", "solid"),
           main = "")
```



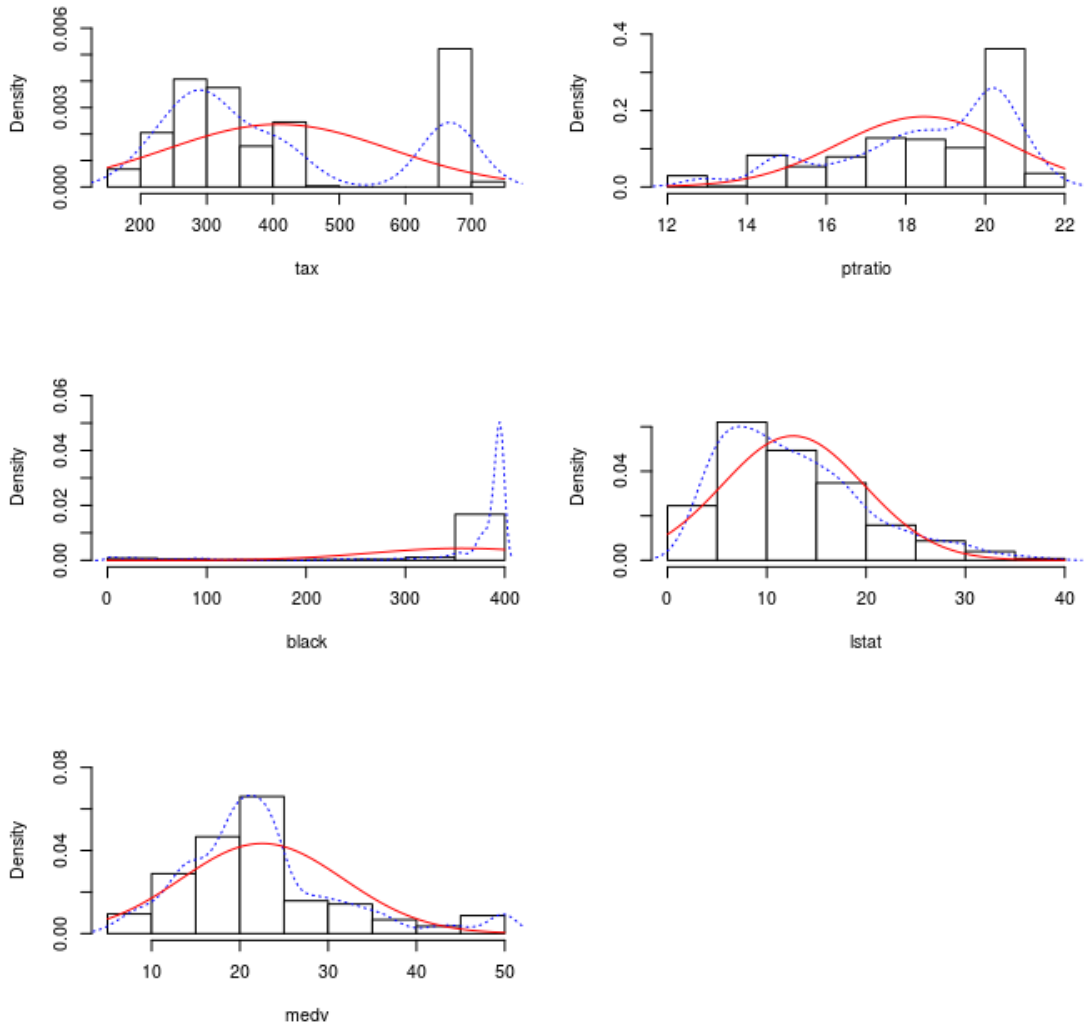





```
multi.hist(x = Boston[, 5:9], dcol = c("blue", "red"), dlty = c("dotted", "solid"),
           main = "")
```




```
multi.hist(x = Boston[, 10:14], dcol = c("blue", "red"),
           dlty = c("dotted",     "solid"), main = "")
```



Regresión lineal simple

Se pretende predecir el valor de la vivienda en función del porcentaje de pobreza de la población. Empleando la función `lm()` se genera un modelo de regresión lineal por mínimos cuadrados en el que la variable respuesta es *medv* y el predictor *lstat*.

```
modelo_simple <- lm(data = Boston, formula = medv ~ lstat)
```

La función `lm()` genera un objeto que almacena toda la información del modelo, para ver su contenido se emplea la función `names()` y para visualizar los principales parámetros del modelo generado se utiliza `summary()`.

```
names(modelo_simple)
```

```
##  [1] "coefficients"  "residuals"     "effects"       "rank"
##  [5] "fitted.values" "assign"        "qr"            "df.residual"
##  [9] "xlevels"       "call"          "terms"         "model"
```

```
summary(modelo_simple)
```

```
##
## Call:
## lm(formula = medv ~ lstat, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.168  -3.990  -1.318   2.034  24.500
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 34.55384    0.56263   61.41   <2e-16 ***
## lstat       -0.95005    0.03873  -24.53   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.216 on 504 degrees of freedom
## Multiple R-squared:  0.5441, Adjusted R-squared:  0.5432
## F-statistic: 601.6 on 1 and 504 DF,  p-value: < 2.2e-16
```

En la información devuelta por el *summary* se observa que el *p-value* del estadístico *F* es muy pequeño, indicando que al menos uno de los predictores del modelo está significativamente relacionado con la variable respuesta. Al tratarse de un modelo simple, el p-value de estadístico *F* es el mismo que el p-value del estadístico *t* del único predictor incluido en el modelo (*lstat*). La evaluación del modelo en conjunto puede hacerse a partir de los valores RSE o del valor R^2 devuelto en el *summary*.

- *Residual standar error* (RSE): En promedio, cualquier predicción del modelo se aleja 6.216 unidades del verdadero valor. Teniendo en cuenta que el valor promedio de la variable respuesta *medv* es de 22.53, RSE es de $\frac{6.216}{22.53} = 27\%$.

- R^2: El predictor *lstatus* empleado en el modelo es capaz de explicar el 54.44% de la variabilidad observada en el precio de las viviendas.

La ventaja de R^2 es que es independiente de la escala en la que se mida la variable respuesta, por lo que su interpretación es más sencilla.

Los dos coeficientes de regresión (β_0 y β_1) estimados por el modelo son significativos y se pueden interpretar como:

- Intercept(β_0): El valor promedio del precio de la vivienda cuando el *lstatus* es 0 es de 34.5538 unidades.
- Predictor lstat(β_1): por cada unidad que se incrementa el predictor *lstat* el precio de la vivienda disminuye en promedio 0.9500 unidades.

La estimación de todo coeficiente de regresión tiene asociada un error estándar, por lo tanto todo coeficiente de regresión tiene su correspondiente intervalo de confianza.

```
confint(modelo_simple, level = 0.95)
```

```
##                 2.5 %     97.5 %
## (Intercept) 33.448457 35.6592247
## lstat       -1.026148 -0.8739505
```

Como era de esperar dado que el *p-value* del predictor *lstat* ha resultado significativo para un $\alpha = 0.05$, su intervalo de confianza del 95% no contiene el valor 0.

Una vez generado el modelo, es posible predecir el valor de la vivienda sabiendo el estatus de la población en la que se encuentra. Toda predicción tiene asociado un error y por lo tanto un intervalo. Es importante diferenciar entre dos tipos de intervalo:

- Intervalo de confianza: Devuelve un intervalo para el valor promedio de todas las viviendas que se encuentren en una población con un determinado porcentaje de pobreza, supóngase *lstat=10*.

```
predict(object = modelo_simple, newdata = data.frame(lstat = c(10)),
        interval = "confidence", level = 0.95)
```

```
##        fit      lwr      upr
## 1 25.05335 24.47413 25.63256
```

- Intervalo de predicción: Devuelve un intervalo para el valor esperado de una vivienda en particular que se encuentre en una población con un determinado porcentaje de pobreza.

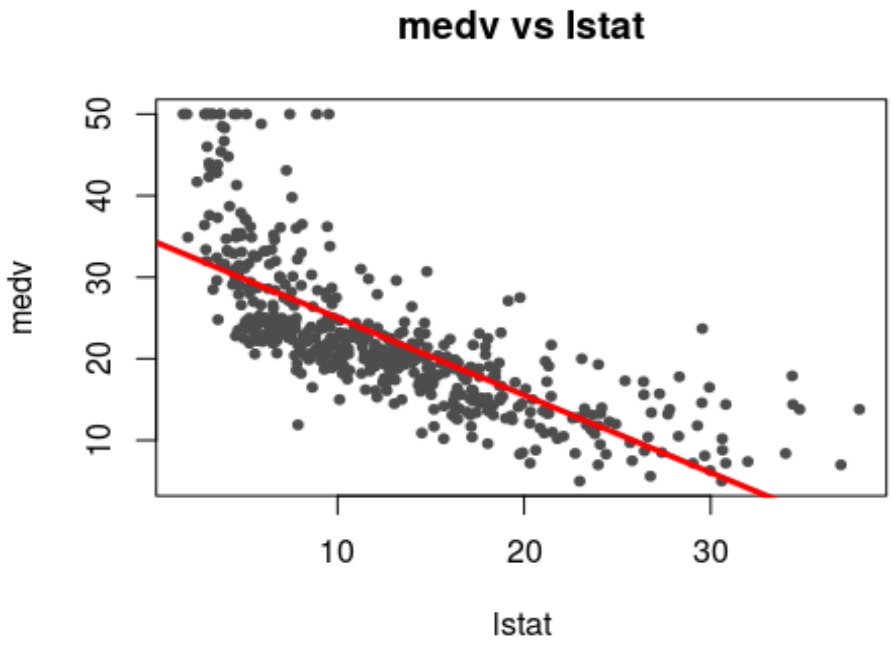
```
predict(object = modelo_simple, newdata = data.frame(lstat = c(10)), interval =
"prediction",
    level = 0.95)
```

```
##        fit      lwr      upr
## 1 25.05335 12.82763 37.27907
```

Como es de esperar ambos intervalos están centrados en torno al mismo valor. Si bien ambos parecen similares, la diferencia se encuentra en que los intervalos de confianza se aplican al valor promedio que se espera de *y* para un determinado valor de *x*, mientras que los intervalos de predicción no se aplican al promedio. Por esta razón los segundos siempre son más amplios que los segundos.

La creación de un modelo de regresión lineal simple suele acompañarse de una representación gráfica superponiendo las observaciones con el modelo. Además de ayudar a la interpretación, es el primer paso para identificar posibles violaciones de las condiciones de la regresión lineal.

```
attach(Boston)
plot(x = lstat, y = medv, main = "medv vs lstat", pch = 20, col = "grey30")
abline(modelo_simple, lwd = 3, col = "red")
```

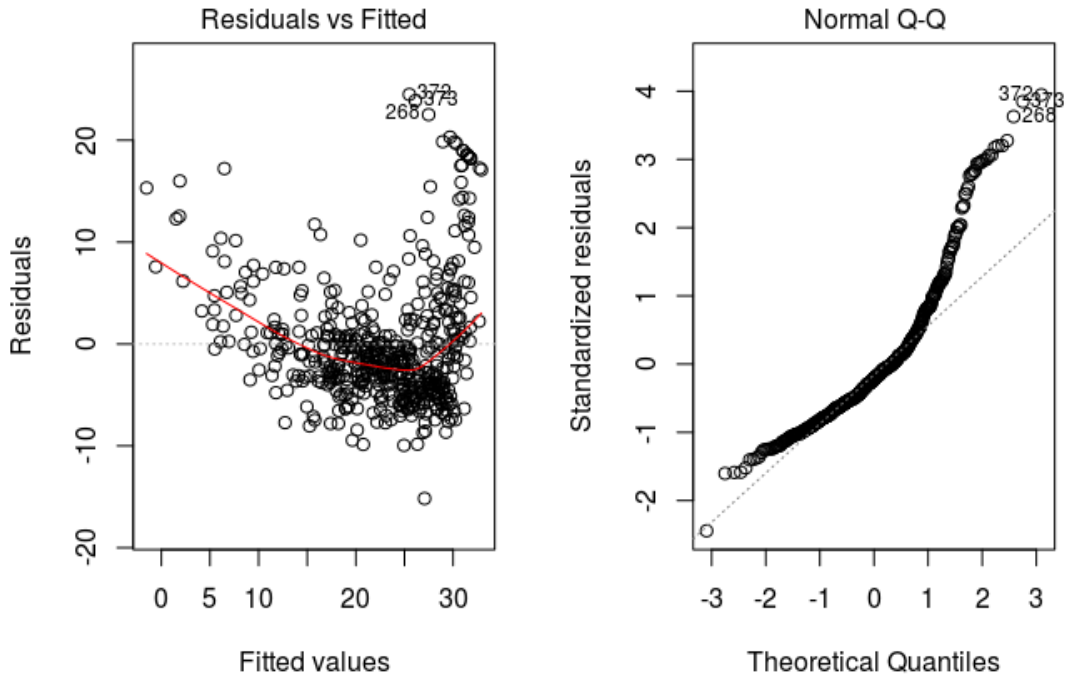


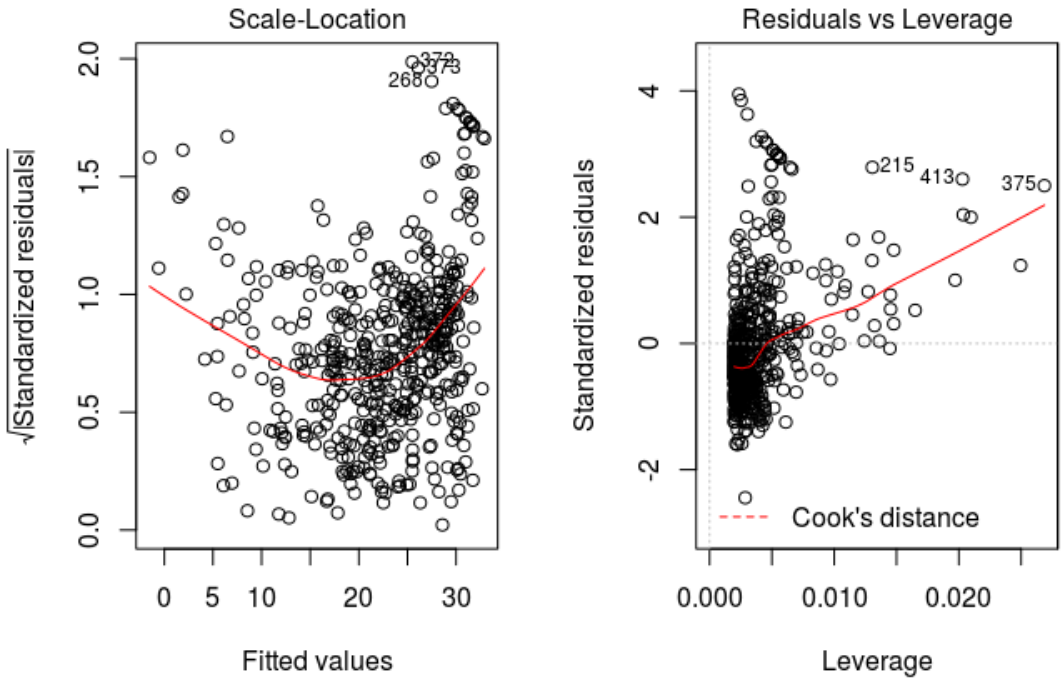
La representación gráfica de las observaciones muestra que la relación entre ambas variables estudiadas no es del todo lineal, lo que apunta a que otro tipo de modelo podría explicar mejor la relación. Aun así la aproximación no es mala.

Una de las mejores formas de confirmar que las condiciones necesarias para un modelo de regresión lineal simple por mínimos cuadrados se cumplen es mediante el estudio de los residuos del modelo.

En R, los residuos se almacenan dentro del modelo bajo el nombre de *residuals*. R genera automáticamente los gráficos más típicos para la evaluación de los residuos de un modelo.

```
par(mfrow = c(1, 2))
plot(modelo_simple)
```

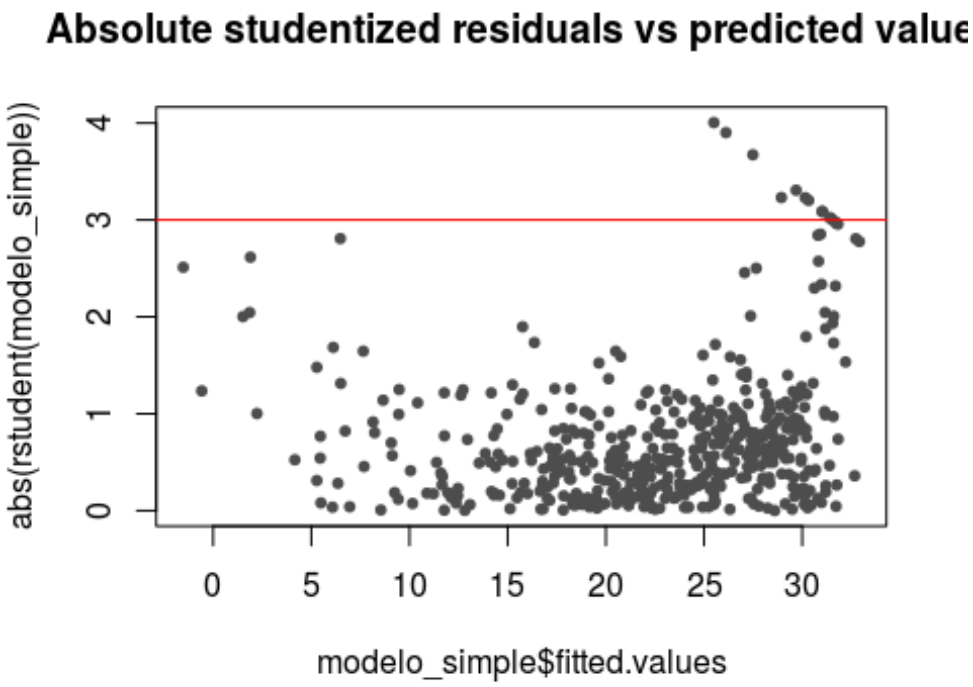




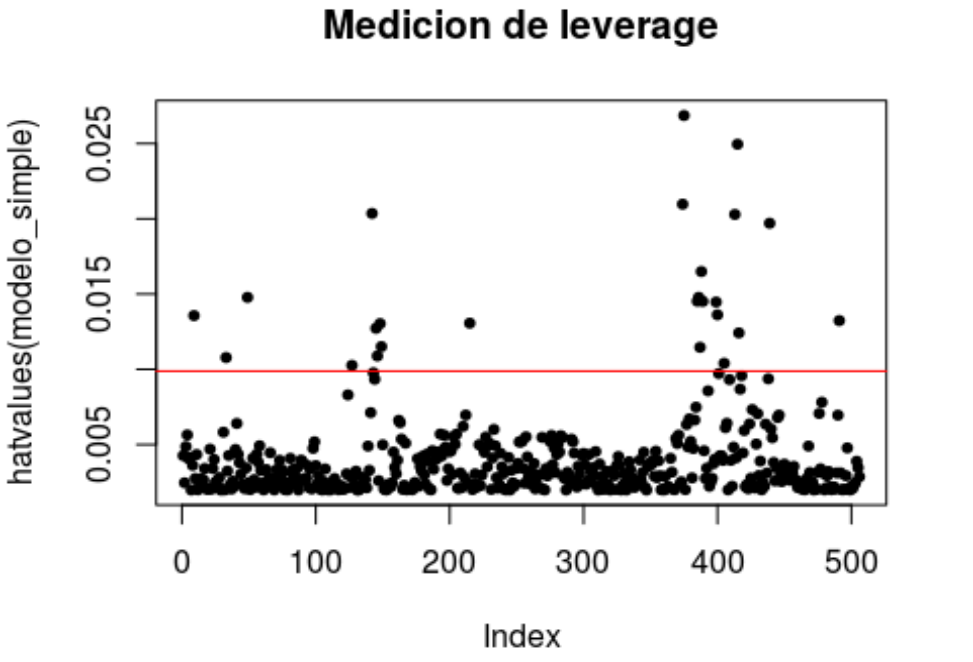
Los residuos confirman que los datos no se distribuyen de forma lineal, ni su varianza constante (plot1). Además se observa que la distribución de los residuos no es normal (plot2). Algunas observaciones tienen un residuo estandarizado absoluto mayor de 3 (1.73 si se considera la raíz cuadrada) lo que es indicativo de observación atípica (plot3). Valores de *Leverages (hat)* mayores que $2.5x((p+1)/n)$, siendo p el número de predictores y n el número de observaciones, o valores de *Cook* mayores de 1 se consideran influyentes (plot4). Todo ello reduce en gran medida la robustez de la estimación del error estándar de los coeficientes de correlación estimados y con ello la del modelo es su conjunto.

Otra forma de identificar las observaciones que puedan ser *outliers* o puntos con alta influencia (*leverage*) es emplear las funciones `rstudent()` y `hatvalues()`.

```
plot(x = modelo_simple$fitted.values, y = abs(rstudent(modelo_simple)),
     main = "Absolute studentized residuals vs predicted values",  pch = 20,
     col = "grey30")
abline(h = 3, col = "red")
```




```
plot(hatvalues(modelo_simple), main = "Medición de leverage", pch = 20)
# Se añade una linea en el threshold de influencia acorde a la regla 2.5x((p+1)/n)
abline(h = 2.5 *((dim(modelo_simple$model)[2] -1 + 1)/dim(modelo_simple$model)[1]),
       col = "red")
```



En este caso muchos de los valores parecen posibles *outliers* o puntos con alta influencia porque los datos realmente no se distribuyen de forma lineal en los extremos.

Modelo

$$precio\ medio\ vivienda = 34.55 - 0.95 * lstat$$

Regresión múltiple

Se desea generar un modelo que permita explicar el precio de la vivienda de una población empleando para ello cualquiera de las variables disponibles en el *dataset* Boston y que resulten útiles en el modelo.

R permite crear un modelo con todas las variables incluidas en un *data.frame* de la siguiente forma:

```
modelo_multiple <- lm(formula = medv ~ ., data = Boston)
# También se pueden especificar una a una
summary(modelo_multiple)
```

```
##
## Call:
## lm(formula = medv ~ ., data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.595  -2.730  -0.518   1.777  26.199
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.646e+01  5.103e+00   7.144 3.28e-12 ***
## crim        -1.080e-01  3.286e-02  -3.287 0.001087 **
## zn           4.642e-02  1.373e-02   3.382 0.000778 ***
## indus        2.056e-02  6.150e-02   0.334 0.738288
## chas1        2.687e+00  8.616e-01   3.118 0.001925 **
## nox         -1.777e+01  3.820e+00  -4.651 4.25e-06 ***
## rm           3.810e+00  4.179e-01   9.116  < 2e-16 ***
## age          6.922e-04  1.321e-02   0.052 0.958229
```

```
## dis         -1.476e+00  1.995e-01  -7.398 6.01e-13 ***
## rad          3.060e-01  6.635e-02   4.613 5.07e-06 ***
## tax         -1.233e-02  3.760e-03  -3.280 0.001112 **
## ptratio     -9.527e-01  1.308e-01  -7.283 1.31e-12 ***
## black        9.312e-03  2.686e-03   3.467 0.000573 ***
## lstat       -5.248e-01  5.072e-02 -10.347  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.745 on 492 degrees of freedom
## Multiple R-squared:  0.7406, Adjusted R-squared:  0.7338
## F-statistic: 108.1 on 13 and 492 DF,  p-value: < 2.2e-16
```

El *p-value* obtenido para el estadístico F es muy pequeño (< 2.2e-16) lo que indica que al menos uno de los predictores introducidos en el modelo está relacionado con la variable respuesta *medv*. El modelo es capaz de explicar el 74% de la variabilidad observada en el precio de la vivienda ($R^2 = 0.74$)

En el *summary* se puede observar que algunos predictores tienen *p-values* muy altos, sugiriendo que no contribuyen al modelo por lo que deben ser excluidos, por ejemplo *age* e *indus*. La exclusión de predictores basándose en *p-values* no es aconsejable, en su lugar se recomienda emplear métodos de *best subset selection*, *stepwise selection (forward, backward e hybrid)* o *Shrinkage/regularization*. Para una descripción detallada de cada uno ver capítulo *Selección de predictores y mejor modelo: Subset selection, Ridge, Lasso y dimension reduction*.

```
step(modelo_multiple, direction = "both", trace = 0)
```

```
##
## Call:
## lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad +
##     tax + ptratio + black + lstat, data = Boston)
##
## Coefficients:
## (Intercept)         crim           zn        chas1          nox
##   36.341145    -0.108413     0.045845     2.718716   -17.376023
##          rm          dis          rad          tax      ptratio
##    3.801579    -1.492711     0.299608    -0.011778    -0.946525
##       black        lstat
##    0.009291    -0.522553
```

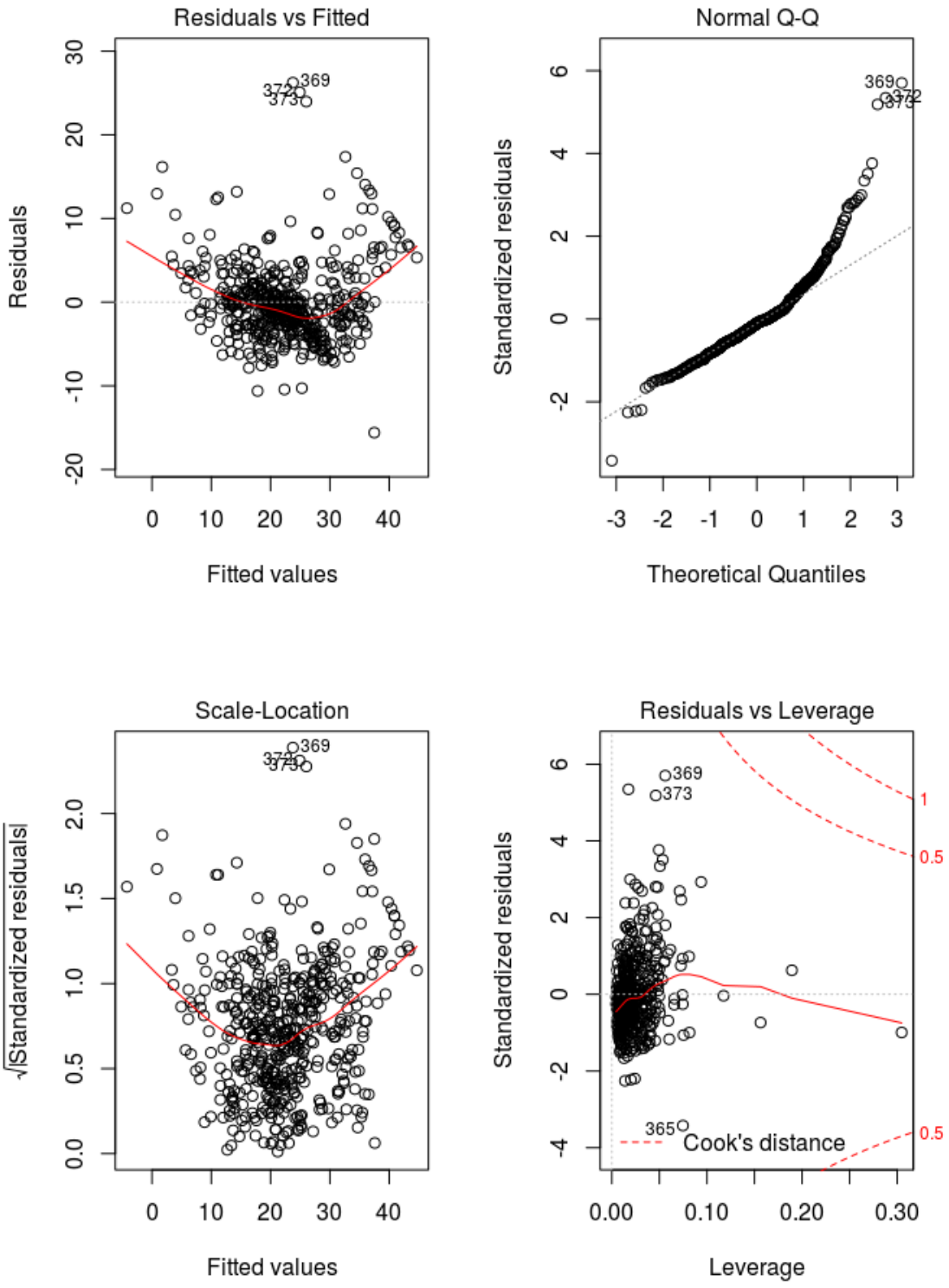
La selección de predictores empleando *stepwise selection (hybrid/doble)* ha identificado como mejor modelo el formado por los predictores *crim, zn, chas, nox, rm, dis, rad, tax, ptratio, black, lstat*.

```
modelo_multiple <- lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad +
                      tax + ptratio + black + lstat, data = Boston)
# También se pueden indicar todas las variables de un data.frame y exluir
# algunas modelo_multiple <- lm(formula = medv~. -age -indus, data = Boston)
summary(modelo_multiple)
```

```
##
## Call:
## lm(formula = medv ~ crim + zn + chas + nox + rm + dis + rad +
##     tax + ptratio + black + lstat, data = Boston)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.5984  -2.7386  -0.5046   1.7273  26.2373
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  36.341145   5.067492   7.171 2.73e-12 ***
## crim         -0.108413   0.032779  -3.307 0.001010 **
## zn            0.045845   0.013523   3.390 0.000754 ***
## chas1         2.718716   0.854240   3.183 0.001551 **
## nox         -17.376023   3.535243  -4.915 1.21e-06 ***
## rm            3.801579   0.406316   9.356  < 2e-16 ***
## dis          -1.492711   0.185731  -8.037 6.84e-15 ***
## rad           0.299608   0.063402   4.726 3.00e-06 ***
## tax          -0.011778   0.003372  -3.493 0.000521 ***
## ptratio      -0.946525   0.129066  -7.334 9.24e-13 ***
## black         0.009291   0.002674   3.475 0.000557 ***
## lstat        -0.522553   0.047424 -11.019  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.736 on 494 degrees of freedom
## Multiple R-squared:  0.7406, Adjusted R-squared:  0.7348
## F-statistic: 128.2 on 11 and 494 DF,  p-value: < 2.2e-16
```

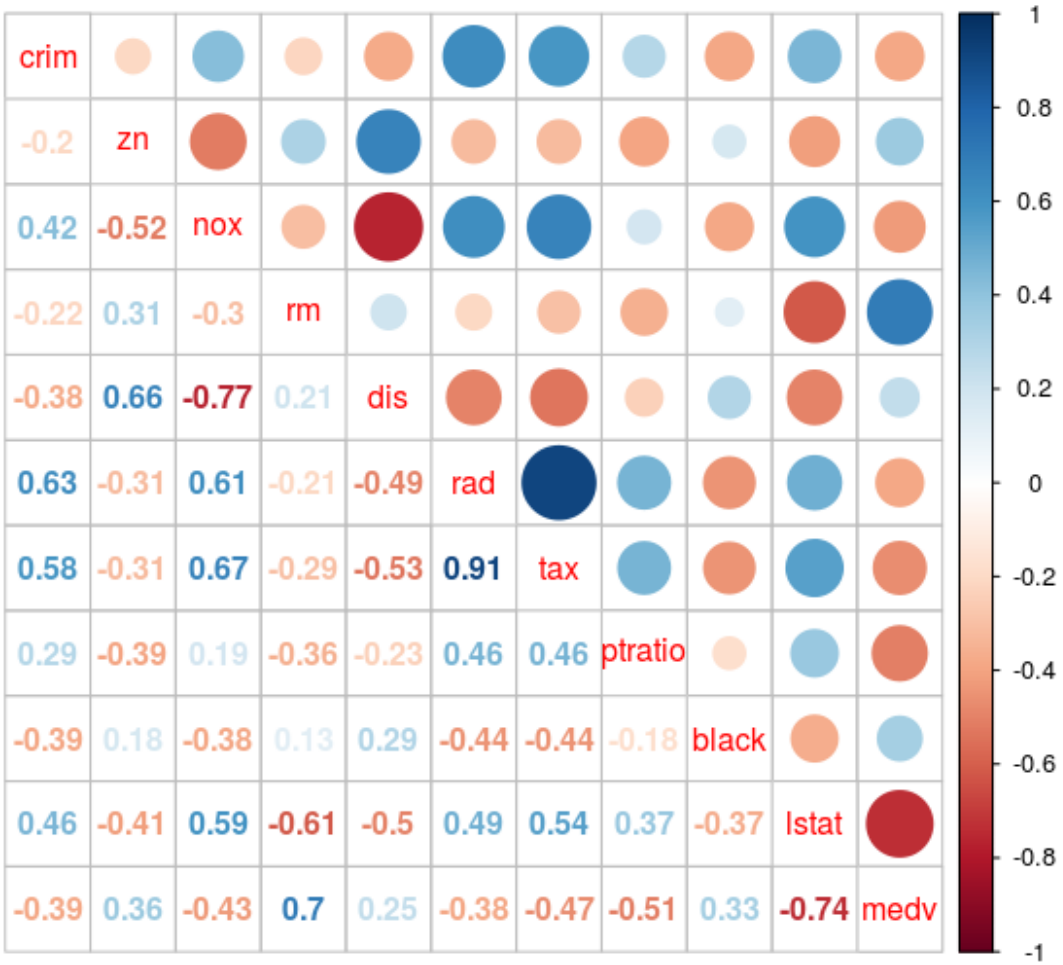
En los modelos de regresión lineal con múltiples predictores, además del estudio de los residuos vistos en el modelo simple, es necesario descartar colinealidad o multicolinealidad entre variables.

```
par(mfrow = c(1, 2))
plot(modelo_multiple)
```



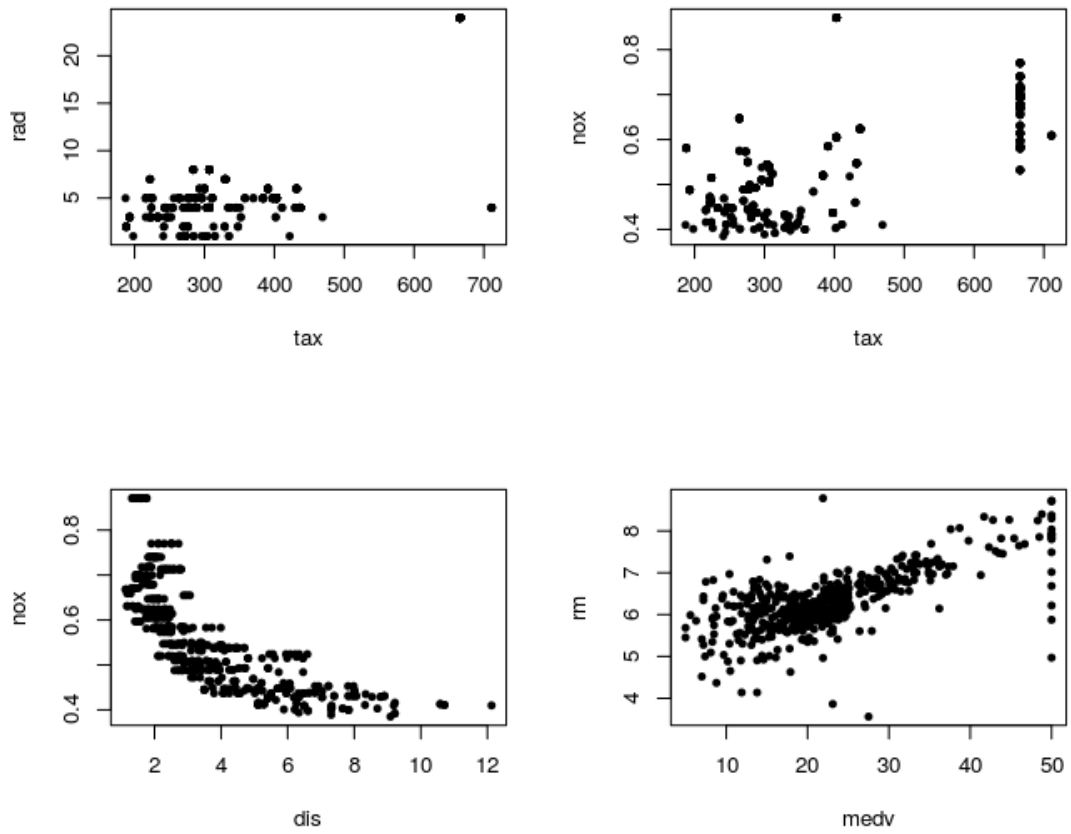
Para la colinealidad se recomienda calcular el coeficiente de correlación entre cada par de predictores incluidos en el modelo:

```
require(corrplot)
corrplot.mixed(corr = cor(Boston[, c("crim", "zn", "nox", "rm", "dis", "rad",
                                     "tax", "ptratio", "black", "lstat", "medv")],
                          method = "pearson"))
```



El análisis muestra correlaciones muy altas entre los predictores *rad* y *tax* (positiva) y entre *dis* y *nox* (negativa).

```
attach(Boston)
par(mfrow = c(2, 2))
plot(x = tax, y = rad, pch = 20)
plot(x = tax, y = nox, pch = 20)
plot(x = dis, y = nox, pch = 20)
plot(x = medv, y = rm, pch = 20)
```



Si la correlación es alta y por lo tanto las variables aportan información redundante, es recomendable analizar si el modelo mejora o no empeora excluyendo alguno de estos predictores.

Para el estudio de la multicolinealidad una de las medidas más utilizadas es el *factor de inflación de varianza VIF*. Puede calcularse mediante la función `vif()` del paquete *car*.

```
require(car)
vif(modelo_multiple)
```

```
##     crim       zn     chas      nox       rm      dis      rad      tax
## 1.789704 2.239229 1.059819 3.778011 1.834806 3.443420 6.861126 7.272386
##  ptratio    black    lstat
## 1.757681 1.341559 2.581984
```

Los índices VIF son bajos o moderados, valores entre 5 y 10 indican posibles problemas y valores mayores o iguales a 10 se consideran muy problemáticos.

Interacción entre predictores

Una de las asunciones del modelo de regresión lineal múltiple es la de aditividad, según la cual los efectos que causan sobre la variable respuesta Y variaciones en el predictor X_i son independientes del valor que tomen los otros predictores. Se conoce como *efecto de interacción* cuando el efecto de un predictor varía dependiendo del valor que adquiera otro predictor. Si esto ocurre, el modelo mejorará si se incluye dicha interacción. Si en un modelo se incorpora una interacción, se deben incluir también los predictores individuales que forman la interacción aun cuando por sí solos no sean significativos.

Supóngase que empleando las variables *age* y *lstat* del data set Boston se quiere generar un modelo lineal que permita predecir el valor medio de la vivienda *medv*.

El modelo lineal múltiple empleando ambos predictores resulta en:

```
modelo <- lm(medv ~ lstat + age, data = Boston)
summary(modelo)
```

```
##
## Call:
## lm(formula = medv ~ lstat + age, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.981  -3.978  -1.283   1.968  23.158
```

```
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 33.22276    0.73085  45.458  < 2e-16 ***
## lstat       -1.03207    0.04819 -21.416  < 2e-16 ***
## age          0.03454    0.01223   2.826  0.00491 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.173 on 503 degrees of freedom
## Multiple R-squared:  0.5513, Adjusted R-squared:  0.5495
## F-statistic:   309 on 2 and 503 DF,  p-value: < 2.2e-16
```

Dado que es un modelo con dos predictores continuos se puede representar el plano de regresión.

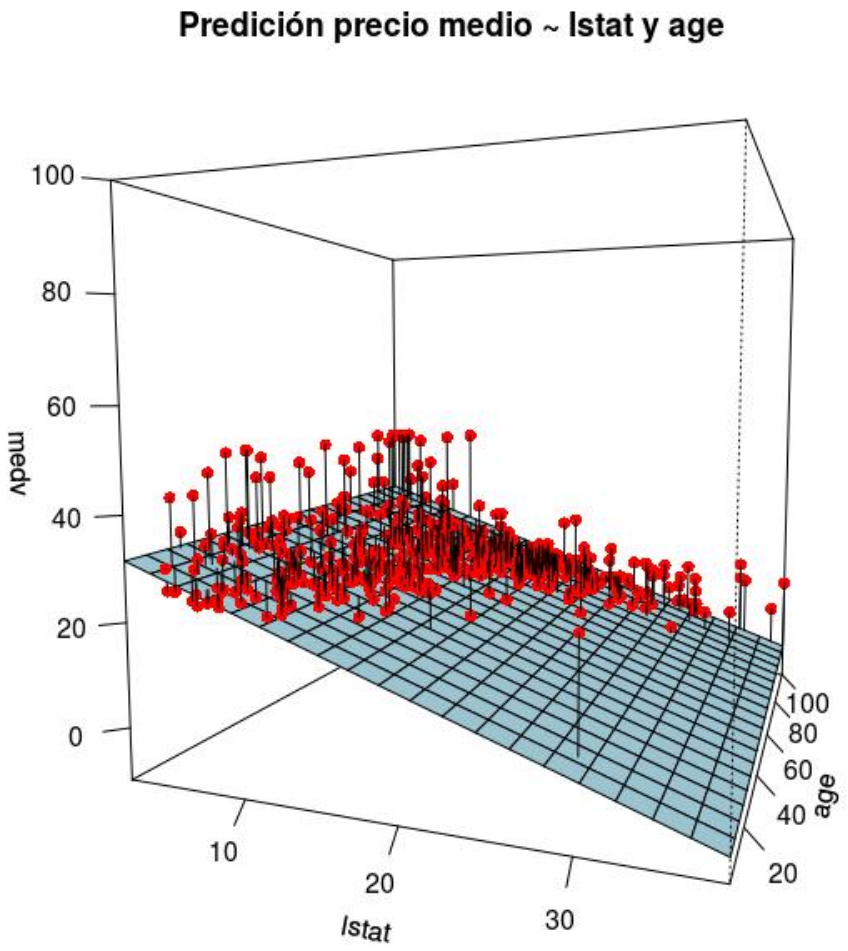
```
rango_lstat <- range(Boston$lstat)
nuevos_valores_lstat <- seq(from = rango_lstat[1], to = rango_lstat[2],

                            length.out     = 20)
rango_age <- range(Boston$age)
nuevos_valores_age <- seq(from = rango_age[1], to = rango_age[2], length.out = 20)

predicciones <- outer(X = nuevos_valores_lstat, Y = nuevos_valores_age,
                      FUN = function(lstat, age) {
                              predict(object = modelo,
                                      newdata = data.frame(lstat, age))
                       })

superficie <- persp(x = nuevos_valores_lstat, y = nuevos_valores_age,
                    z = predicciones,theta = 20, phi = 5, col = "lightblue",
                    shade = 0.1, zlim = range(-10, 100), xlab = "lstat",
                    ylab = "age", zlab = "medv", ticktype = "detailed",
                    main = "Predición precio medio ~ lstat y age")

observaciones <- trans3d(Boston$lstat, Boston$age, Boston$medv, superficie)
error <- trans3d(Boston$lstat, Boston$age, fitted(modelo), superficie)
points(observaciones, col = "red", pch = 16)
segments(observaciones$x, observaciones$y, error$x, error$y)
```



En R se puede generar un modelo con interacción de dos formas: indicando de forma explícita los predictores individuales y entre que predictores se quiere evaluar la interacción.

```
lm(medv ~ lstat + age + lstat:age, data = Boston)
```

O bien de forma directa

```
modelo_interaccion <- lm(medv ~ lstat * age, data = Boston)
summary(modelo_interaccion)
```

```
## Call:
## lm(formula = medv ~ lstat * age, data = Boston)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.806  -4.045  -1.333   2.085  27.552
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 36.0885359  1.4698355  24.553  < 2e-16 ***
## lstat       -1.3921168  0.1674555  -8.313 8.78e-16 ***
## age         -0.0007209  0.0198792  -0.036   0.9711
## lstat:age    0.0041560  0.0018518   2.244   0.0252 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.149 on 502 degrees of freedom
## Multiple R-squared:  0.5557, Adjusted R-squared:  0.5531
## F-statistic: 209.3 on 3 and 502 DF,  p-value: < 2.2e-16
```

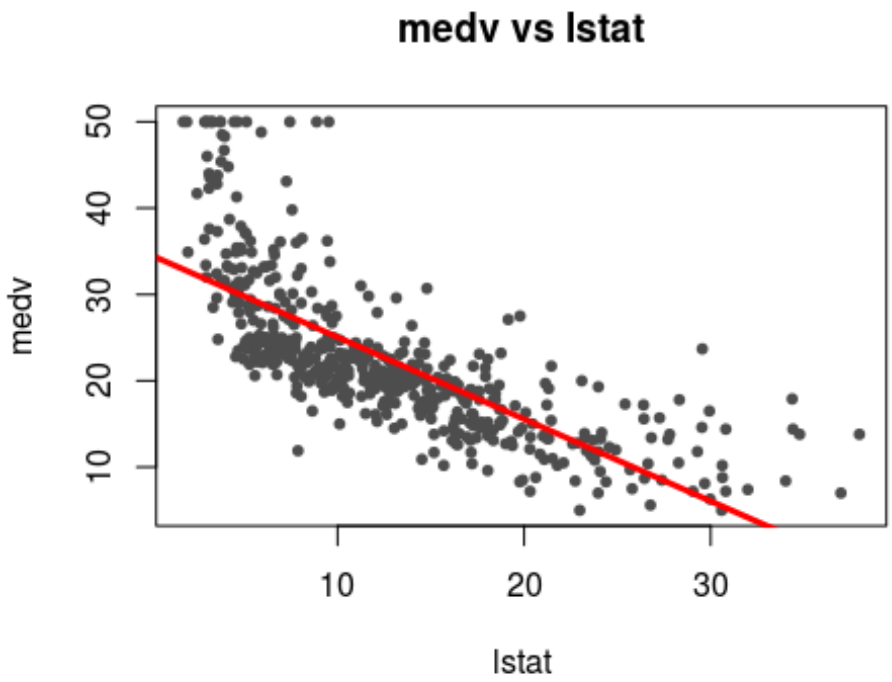
La interacción, aunque significativa, apenas aporta mejora al modelo, *R-squared sin interacción = 0.5513* y *R-squared con interacción = 0.5557*. Por lo tanto, siguiendo el principio de parsimonia, el modelo más adecuado (dentro de lo limitado ya que solo explica el 55% de variabilidad) es el modelo sin interacción.

Regresión Polinomial: incorporar no-linealidad a los modelos lineales.

La Regresión Polinomial, aunque permite describir relaciones no lineales, se trata de un modelo lineal en el que se incorporan nuevos predictores elevando el valor de los ya existentes a diferentes potencias.

Cuando se intenta predecir el valor de la vivienda en función del estatus de la población, el modelo lineal generado no se ajusta del todo bien debido a que las observaciones muestran una relación entre ambas variables con cierta curvatura.

```
attach(Boston)
plot(x = lstat, y = medv, main = "medv vs lstat", pch = 20, col = "grey30")
abline(modelo_simple, lwd = 3, col = "red")
```



La curvatura descrita apunta a una posible relación cuadrática, por lo que un polinomio de segundo grado podría capturar mejor la relación entre las variables. En R se pueden generar modelos de regresión polinómica de diferentes formas:

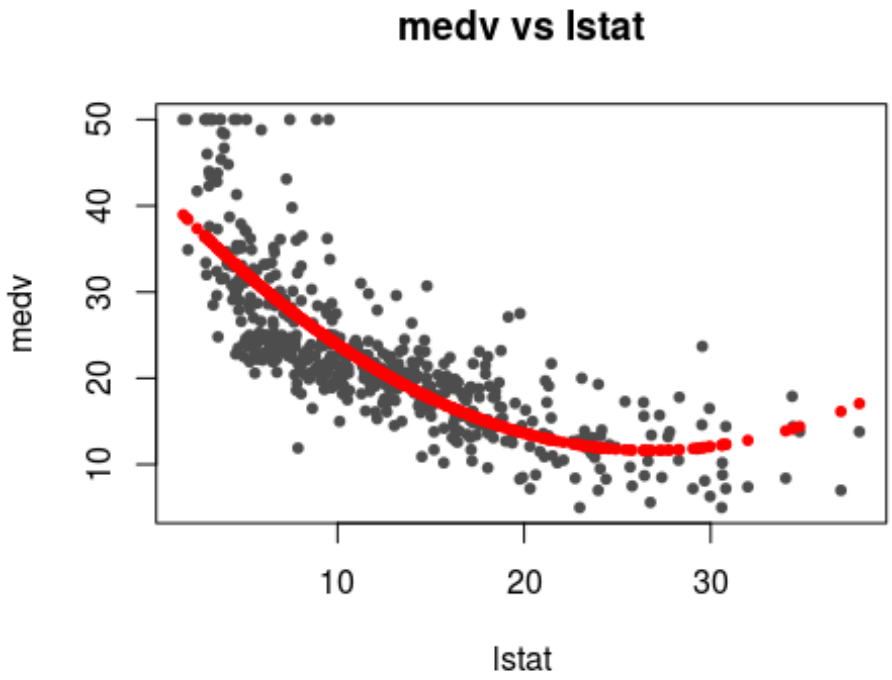
- Identificando cada elemento del polinomio: *modelo_pol2* `<- lm(formula = medv ~ lstat + I(lstat^2), data = Boston)` El uso de `I()` es necesario ya que el símbolo ^ tiene otra función dentro de las formula de R.
- Con la función poly(): `lm(formula = medv ~ poly(lstat, 2), data = Boston)`

```
modelo_pol2 <- lm(formula = medv ~ poly(lstat, 2), data = Boston)
summary(modelo_pol2)
```

```
##
## Call:
## lm(formula = medv ~ poly(lstat, 2), data = Boston)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.2834  -3.8313  -0.5295   2.3095  25.4148
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       22.5328     0.2456   91.76   <2e-16 ***
## poly(lstat, 2)1 -152.4595     5.5237  -27.60   <2e-16 ***
## poly(lstat, 2)2   64.2272     5.5237   11.63   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.524 on 503 degrees of freedom
## Multiple R-squared:  0.6407, Adjusted R-squared:  0.6393
## F-statistic: 448.5 on 2 and 503 DF,  p-value: < 2.2e-16
```

El *p-value* próximo a 0 del predictor cuadrático de *lstat* indica que contribuye a mejorar el modelo.

```
attach(Boston)
plot(x = lstat, y = medv, main = "medv vs lstat", pch = 20, col = "grey30")
points(lstat, fitted(modelo_pol2), col = "red", pch = 20)
```



A la hora de comparar dos modelos, se pueden evaluar sus R^2. En este caso el modelo cuadrático es capaz de explicar un 64% de variabilidad frente al 54% del modelo lineal. En el caso particular de que los modelos a comparar sean anidados (el modelo de menor tamaño está formado por un subset de predictores del modelo de mayor tamaño), se puede saber si el modelo mayor aporta una mejora sustancial estudiando si los coeficientes de regresión de los predictores adicionales son distintos a cero. El test estadístico empleado para hacerlo es el ANOVA.

$$Modelo_{menor}: \ y = \beta_0 + \beta_1 x_1 + \ldots + \beta_k x_k$$

$$Modelo_{mayor}: \ y = \beta_0 + \beta_1 x_1 + \ldots + \beta_k x_k + \beta_{k+1} x_{k+1} + \ldots + \beta_p x_p$$

La hipótesis a contrastar es que todos los coeficientes de regresión de los predictores adicionales son igual a cero, frente a la hipótesis alternativa de que al menos uno es distinto.

$$H_0: \beta_{k+1} = \ldots = \beta_p$$

El estadístico empleado es:

$$F = \frac{(SEE_{Modelo_{menor}} - SEE_{Modelo_{mayor}})/(p-k)}{SEE_{Modelo_{mayor}}/(n-p-1)}$$

Dado que un polinomio de orden *n* siempre va a estar anidado a uno de orden *n+1*, se pueden comparar modelos polinómicos dentro un rango de grados haciendo comparaciones secuenciales.

```
anova(modelo_simple, modelo_pol2)
```

```
## Analysis of Variance Table
##
## Model 1: medv ~ lstat
## Model 2: medv ~ poly(lstat, 2)
##   Res.Df   RSS Df Sum of Sq     F    Pr(>F)
## 1    504 19472
## 2    503 15347  1    4125.1 135.2 < 2.2e-16 ***
## ---
```

El *p-value* obtenido para el estadístico F confirma que el modelo cuadrático es superior.

Ejercicios propuestos

1.Describe the null hypotheses to which the p-values given in Table 3.4 correspond. Explain what conclusions you can draw based on these p-values. Your explanation should be phrased in terms of sales, TV, age, and newspaper, rather than in terms of the coefficients of the linear model.

Describe a que hipótesis nula pertenecen los p-values de la tabla. Explica las conclusiones que se pueden extraer de ellos.

..	Estimate	Std. Error	t value	Pr(>
(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

Cuando se genera un modelo de regresión lineal simple o múltiple se estima, para cada predictor introducido en el modelo, un coeficiente de correlación que mide la influencia de cada uno de ellos sobre la variable respuesta. Los valores estimados son tales que el modelo generado se ajusta lo mejor posible a las observaciones a partir de las que se creado (uno de los métodos más comúnmente empleado es el de mínimos cuadrados). Dado que se trata de estimaciones, cada coeficiente de correlación calculado tiene su error asociado.

Para determinar si cada uno de los predictores incluidos en un modelo está realmente asociado con la variable respuesta se emplea un test de significancia en contra de la hipótesis nula de que el valor de su coeficiente de correlación estimado es 0. El test estadístico empleado es un t-test.

La tabla 3.4 recoge el resumen de los coeficientes de correlación estimados para un modelo que contiene como predictores *tv*, *age* y *newspaper* y variable respuesta *sales*. *Los p-value* de los predictores *tv* y *age* son muy próximos a cero, lo que significa una fuerte evidencia en contra de la hipótesis nula de que sus coeficientes de correlación son cero, por lo tanto, sí se puede afirmar que están relacionados con la variable *sales*. En contraposición, el p-value del predictor *newspaper* es alto (0.8599) por lo que no hay evidencia suficiente para rechazar la hipótesis nula de que el verdadero valor de su coeficiente de correlación es 0. No se puede afirmar que exista una relación entre *newspaper* y *sales*.

2.Carefully explain the differences between the KNN classifier and KNN regression methods.

Explicar la diferencia entre clasificador-KNN y regresión-KNN

El *KNN clasifier* (K-Nearest Neighbours Clasifier) es un método estadístico empleado para asignar observaciones a uno de los diferentes niveles de una variable cualitativa cuando el valor de esta se desconoce. Consiste en identificar las *k* observaciones más cercanas y calcular con ellas la fracción de las mismas que pertenecen a cada uno de los niveles. La nueva observación se asigna al nivel para el que la proporción sea más alta.

La *KNN regression* es un método no barométrico de regresión. Consiste en identificar las *k* observaciones más cercanas al punto que se quiere predecir y asignarle como valor predicho el promedio de las observaciones vecinas.

En ambos casos, cuanto menor sea el número *k* de observaciones vecinas empleadas mayor será la flexibilidad del método y a su vez mayor la influencia que tiene cada observación empleada como vecina (peligro de *overfitting*).

2.Suppose we have a data set with five predictors, x_1 = GPA, x_2 = IQ, x_3= Gender (1 for Female and 0 for Male), x_4 = Interaction between GPA and IQ, and x_5 = Interaction between GPA and Gender. The response is starting salary after

graduation (in thousand of dollars). Suppose we use least squares to fit the model, and get $\hat{\beta}_0$=50, $\hat{\beta}_1$=20, $\hat{\beta}_2$=0.07, $\hat{\beta}_3$=35, $\hat{\beta}_4$=0.01, $\hat{\beta}_5$=−10.

Which answer is correct, and why?

1. For a fixed value of IQ and GPA, males earn more on average than females.

2. For a fixed value of IQ and GPA, females earn more on average than males.

3. For a fixed value of IQ and GPA, males earn more on average than females provided that the GPA is high enough.

4. For a fixed value of IQ and GPA, females earn more on average than males provided that the GPA is high enough.

La ecuación de la línea de mínimos cuadrados del modelo resultante acorde al enunciado es:

$$\hat{y} = 50 + 20GPA + 0.07IQ + 35Gender + 0.01GPA \times IQ - 10GPA \times Gender$$

Para hombres (gender=0) queda como:

$$\hat{y} = 50 + 20GPA + 0.07IQ + 0.01GPA \times IQ$$

Para mujeres (gender=1) queda como:

$$\hat{y} = 50 + 20GPA + 0.07IQ + 35 + 0.01GPA \times IQ - 10GPA$$

$$\hat{y} = 85 + 10GPA + 0.07IQ + 0.01GPA \times IQ$$

Para valores de GPA menores de 3.5 el salario inicial promedio de las mujeres es mayor. Cuando el valor de GPA es igual a 3.5 los salarios iniciales promedio son iguales para ambos sexos. Si el valor de GPA es mayor que 3.5, el salario inicial promedio de los hombres es mayor.

Predict the salary of a female with IQ of 110 and a GPA of 4.0.

$$\hat{y} = 85 + 10 * 4 + 0.07 * 110 + 0.01 * 4 * 110 = 137.1$$

True or false: Since the coefficient for the GPA/IQ interaction term is very small, there is very little evidence of an interaction effect. Justify your answer.

El coeficiente de correlación de un predictor es una estimación que indica cuanto varía en promedio la variable respuesta por cada unidad que se incrementa el predictor (manteniendo constante el resto). Cuantifica por lo tanto el tamaño de la influencia del predictor pero no si dicha influencia observada es real o solo por azar. Para este segundo fin se emplea el *p-value* de un t-test que contraste la hipótesis nula de que el valor del coeficiente de correlación es 0. Así pues, el valor de un coeficiente de correlación puede ser pequeño pero no por ello deja de ser real (siempre que su *p-value* así lo indique).

3. I collect a set of data (n=100 observations) containing a single predictor and a quantitative response. I then fit a linear regression model to the data, as well as a separate cubic regression, i.e.

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \varepsilon$$

Suppose that the true relationship between X and Y is linear, i.e. $Y = \beta_0 + \beta_1 X + \varepsilon$**. Consider the training residual sum of squares (RSS) for the linear regression, and also the training RSS for the cubic regression. Would we expect one to be lower than the other, would we expect them to be the same, or is there not enough information to tell? Justify your answer.**

El *training residual sum of squares (RSS)* es el sumatorio de los cuadrados de los errores cuando se emplean las observaciones utilizadas para crear el modelo, es una de las medidas de bondad de ajuste del modelo. Si la verdadera relación entre la variable Y y el predictor X es lineal, el modelo de regresión lineal, que precisamente asume este tipo de función $f(X)$, se va a poder ajustar mejor a los datos que el modelo cuadrático, por lo tanto el RSS del modelo lineal será menor.

Answer (a) using test rather than training RSS.

El *test residual sum of squares (TSS)* es el sumatorio de los cuadrados de los errores obtenido al emplear un set de observaciones distintas a las utilizadas para crear el modelo. El *TSS* es importante para determinar la utilidad del modelo generado. Un modelo con *overfitting* se ajustará muy bien a las observaciones empleadas para entrenarlo por lo que su *RSS* será pequeño, pero puede que para nuevas observaciones no lo haga, disparando el *TSS*. A este problema se le conoce como el equilibrio entre bias y varianza.

En el caso presentado por el ejercicio, siendo la verdadera relación entre ambas variables lineal, los nuevos sets de datos es de esperar que mantengan esta relación, por lo que el modelo lineal seguirá ajustándose mejor generando un menor *TSS*. El valor exacto dependerá de cada nuevo set, por lo que, si bien en promedio el *TSS* del modelo lineal será menor, no es posible confirmarlo con total seguridad para todos los casos.

Suppose that the true relationship between X and Y is not linear, but we don't know how far it is from linear. Consider the training RSS for the linear regression, and also the training RSS for the cubic regression. Would we expect one to be lower than the other, would we expect them to be the same, or is there not enough information to tell? Justify your answer.

Un modelo de regresión cúbico permite mayor flexibilidad que un modelo lineal. En este caso se desconoce cómo es la relación pero se sabe que no es lineal por lo que independientemente de cómo de exacto sea el ajuste mediante un modelo cúbico, siempre va a ser mejor que uno lineal y por lo tanto menor su *RSS*.

Answer (c) using test rather than training RSS.

Dado que no se conoce cuál es la verdadera relación entre las dos variables, no es posible determinar cuál de los dos modelos tendrá menor **TSS**. Esto se debe a que el valor del *TSS* es el resultado del equilibrio entre bias y varianza. El modelo cúbico es más susceptible a la varianza debido a su flexibilidad, lo que aumentaría el *TSS*. El modelo lineal por su parte es más susceptible a bias al ser menos flexible, lo que también aumenta el *TSS*.