Solución Ejercicios Tema 2. Regresión lineal múltiple.

Máster en Ciencia de Datos. Módulo: Análisis exploratorio de datos

Ana Navarro Quiles

Curso 2022/2023

Ejercicio 1

Antes de que comience la construcción de un puente se pasa por una serie de etapas de producción, entre las que destaca su diseño. Esta fase se compone a su vez de varias actividades, por lo que suele ser de interés la predicción del tiempo de diseño a nivel de planificación presupuestaria. En el fichero **puentes** hay información sobre los proyectos de construcción de 45 puentes. A partir de dicha información trata de valorar el tiempo *Time* que se tarda en diseñar un puente en base a:

- Superficie de cubierta de puente (en miles de pies cuadrados), variable *DArea*
- Coste de construcción (en miles de dólares), variable *CCost*
- Número de planos estructurales, variable *DWGS*
- Longitud del puente (en pies), variable *Length*
- Número de tramos, variable *Spans*

Realiza el análisis indicando con todo detalle las características del modelo que vayas a emplear, las suposiciones que has de hacer y la validez de tus conclusiones. Con el modelo elegido responde a las siguientes preguntas

```
load('datosTema2.Rdata')
```

```
# Primero debemos eliminar la columna case, dado que no es una variable.
puentes2<-subset(puentes,select = c(-Case))
mod1 <- lm(Time ~ ., data=puentes2, na.action=na.exclude)
summary(mod1)
```

```
##
## Call:
## lm(formula = Time ~ ., data = puentes2, na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -81.816 -26.797  -9.674  24.882 180.443
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -34.83256   25.03837  -1.391    0.172
## DArea         0.24675    1.63170   0.151    0.881
## CCost        -0.02107    0.07143  -0.295    0.770
## Dwgs         19.68195    4.08583   4.817 2.23e-05 ***
## Length        0.05186    0.10378   0.500    0.620
## Spans        15.50454   10.14243   1.529    0.134
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 55.31 on 39 degrees of freedom
## Multiple R-squared:  0.7101, Adjusted R-squared:  0.6729
## F-statistic:  19.1 on 5 and 39 DF,  p-value: 1.435e-09
```

```
# Varias formas de resolver el ejercicio.

# Primera: Quitando variables una a una teniendo en cuenta el p-valor.
mod2<-update(mod1,~.-DArea)
summary(mod2)
```

```
##
## Call:
## lm(formula = Time ~ CCost + Dwgs + Length + Spans, data = puentes2,
##     na.action = na.exclude)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -80.16 -28.17 -10.03  25.23 179.45
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -35.25186   24.57855  -1.434   0.1593
## CCost        -0.01365    0.05125  -0.266   0.7914
## Dwgs         19.68023    4.03560   4.877 1.75e-05 ***
## Length        0.04765    0.09876   0.483   0.6321
## Spans        16.14138    9.11344   1.771   0.0842 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 54.64 on 40 degrees of freedom
## Multiple R-squared:  0.7099, Adjusted R-squared:  0.6809
## F-statistic: 24.47 on 4 and 40 DF,  p-value: 2.711e-10
```

```
mod3<-update(mod2,~.-CCost)
summary(mod3)
```

```
##
## Call:
## lm(formula = Time ~ Dwgs + Length + Spans, data = puentes2, na.action = na.exclude)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -76.53 -30.47 -10.19  24.62 181.54
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -33.83215   23.71973  -1.426   0.1613
## Dwgs         19.25041    3.65648   5.265 4.77e-06 ***
## Length        0.03274    0.08041   0.407   0.6860
## Spans        16.43716    8.94240   1.838   0.0733 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 54.01 on 41 degrees of freedom
## Multiple R-squared:  0.7094, Adjusted R-squared:  0.6881
```

```
## F-statistic: 33.36 on 3 and 41 DF,  p-value: 4.407e-11
```

```
mod4<-update(mod3,~.-Length)
summary(mod4)
```

```
##
## Call:
## lm(formula = Time ~ Dwgs + Spans, data = puentes2, na.action = na.exclude)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -75.97 -25.21 -10.37  24.60 180.27
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -37.527     21.696  -1.730  0.09104 .
## Dwgs          19.808      3.356   5.902 5.49e-07 ***
## Spans         19.154      5.895   3.249  0.00228 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 53.47 on 42 degrees of freedom
## Multiple R-squared:  0.7082, Adjusted R-squared:  0.6943
## F-statistic: 50.97 on 2 and 42 DF,  p-value: 5.847e-12
```

```
# Segunda forma de hacerlo:
modf <- step(mod1, direction = 'both',trace=0)
summary(modf)
```

```
##
## Call:
## lm(formula = Time ~ Dwgs + Spans, data = puentes2, na.action = na.exclude)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -75.97 -25.21 -10.37  24.60 180.27
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -37.527     21.696  -1.730  0.09104 .
## Dwgs          19.808      3.356   5.902 5.49e-07 ***
## Spans         19.154      5.895   3.249  0.00228 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 53.47 on 42 degrees of freedom
## Multiple R-squared:  0.7082, Adjusted R-squared:  0.6943
## F-statistic: 50.97 on 2 and 42 DF,  p-value: 5.847e-12
```

```
# Ambas formas nos ha dado el mismo resultado. Time ~ Dwgs + Spans
```

```
# En este punto podemos plantearnos si el intercept es significativo o no

modf_SinInter <- update(modf,~.-1)
summary(modf_SinInter)
```

```
##
```

```
## Call:
## lm(formula = Time ~ Dwgs + Spans - 1, data = puentes2, na.action = na.exclude)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -89.39 -32.91 -22.61  13.55 191.28
##
## Coefficients:
##       Estimate Std. Error t value Pr(>|t|)
## Dwgs    15.318      2.176   7.040 1.12e-08 ***
## Spans   19.439      6.028   3.225  0.00241 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 54.7 on 43 degrees of freedom
## Multiple R-squared:  0.9124, Adjusted R-squared:  0.9084
## F-statistic:   224 on 2 and 43 DF,  p-value: < 2.2e-16
```

```
AIC(modf,modf_SinInter)
```

```
##               df      AIC
## modf           4 490.7263
## modf_SinInter  3 491.8226
```

```
# Hemos ganado R^2 sin el intercepto, pero el AIC es mayor considerándolo.

# - El R^2 te dice como de explicativo es tu modelo. Es decir, que % de variabilidad
# de la variable Y queda explicado por el modelo.
# - El AIC es una medida de lo bien que el modelo se ajustará a nuevos datos, no a los datos existentes
# Imaginad que estamos tratando de predecir la salida a partir de algunas variables conocidas. Si se añ
# Si queréis profundizar en ambos criterios os recomiendo el libro:
# Introduction to Statistical Learning with R
# Lo dejo en el aula virtual

# Por tanto, depende de los fines de vuestra regresión (explicativos o predictivos)
# eligiremos un método u otro para su comparación.

# En este caso voy a resolver el problema considerando el Intercepto (modf) dado
# que es el que me daba el comando step directamente y el que menos AIC tiene.

# Sobre el Intercept: En general no es recomendable eliminar el Intercepto (a no ser que tengamos una r
# Por ejemplo, al hacer el Intercepto = 0 cambia la escala por completo y puede ser que
# el R^2 sea mayor por que la nube de puntos está muy lejana (algo similar a lo que pasa con los influy
# Hay casos particulares que puede ser interesante observar que sucede al forzarlo cero (por ejemplos c
```

a) ¿Cuál es el porcentaje de varianza explicada por tu modelo?¿Qué variables son relevantes?

70.82%. Dwgs y Spans.

b) ¿Cuál será el tiempo estimado según tu modelo para la construcción de un puente con los predictores en su valor promedio?¿Y cuál sería el intervalo de confianza para el promedio de tiempo predicho? ¿Y si se trata de un nuevo puente?

```
x0 <- data.frame(Dwgs=mean(puentes2$Dwgs), Spans=mean(puentes2$Spans))
predict(modf,newdata=x0,interval='confidence') # IC
```

```
##        fit      lwr      upr
```

```
## 1 153.3067 137.2199 169.3935
```

```
predict(modf,newdata=x0,interval='prediction') #nuevo puente
```

```
##        fit      lwr      upr
## 1 153.3067 44.20065 262.4127
```

c) Uno de los constructores indica que, en su experiencia, se tarda lo mismo en construir un puente de 1,2 o 3 tramos, y algo más en construir puentes de más de tres tramos ¿Podrías construir un modelo de regresión para comprobar la hipótesis del constructor ?¿Te parece acertada dicha hipótesis en función de la bondad de ajuste?

```
# Vamos a hacer las 4 categorías (1,2,3, y más de 3 (hasta el 7 que es el máximo))
puentes2$Spans_c<-cut(puentes2$Spans, breaks=c(0,1,2,3,7))
puentes3<-subset(puentes2,select = c(-Spans))
mod1c <- lm(Time ~ ., data=puentes3, na.action=na.exclude)
summary(mod1c)
```

```
##
## Call:
## lm(formula = Time ~ ., data = puentes3, na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -80.835 -27.921  -9.197  26.781 176.498
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -20.35433   25.74598  -0.791 0.434226
## DArea          0.23019    1.68721   0.136 0.892221
## CCost         -0.03986    0.07309  -0.545 0.588755
## Dwgs          19.77814    4.72538   4.186 0.000168 ***
## Length         0.07422    0.09819   0.756 0.454519
## Spans_c(1,2]  15.72239   25.85810   0.608 0.546885
## Spans_c(2,3]  34.98546   29.36079   1.192 0.241017
## Spans_c(3,7]  70.00347   43.68505   1.602 0.117558
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 56.28 on 37 degrees of freedom
## Multiple R-squared:  0.7153, Adjusted R-squared:  0.6614
## F-statistic: 13.28 on 7 and 37 DF,  p-value: 2.038e-08
```

```
mod2c<-update(mod1c,~.-DArea)
summary(mod2c)
```

```
##
## Call:
## lm(formula = Time ~ CCost + Dwgs + Length + Spans_c, data = puentes3,
##     na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -81.624 -31.403  -9.152  26.645 175.658
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)  -20.23624   25.39698  -0.797 0.430519
## CCost         -0.03340    0.05494  -0.608 0.546838
## Dwgs          19.83250    4.64736   4.267 0.000127 ***
## Length         0.07084    0.09379   0.755 0.454689
## Spans_c(1,2]  15.43403   25.43661   0.607 0.547616
## Spans_c(2,3]  35.55884   28.68075   1.240 0.222646
## Spans_c(3,7]  72.27669   39.85775   1.813 0.077679 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 55.55 on 38 degrees of freedom
## Multiple R-squared:  0.7151, Adjusted R-squared:  0.6701
## F-statistic:  15.9 on 6 and 38 DF,  p-value: 4.86e-09
```

```
mod3c<-update(mod2c,~.-CCost)
summary(mod3c)
```

```
##
## Call:
## lm(formula = Time ~ Dwgs + Length + Spans_c, data = puentes3,
##     na.action = na.exclude)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -81.06 -30.84 -10.84  25.17 177.96
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -15.54178   23.99830  -0.648   0.5210
## Dwgs          18.68410    4.21169   4.436 7.28e-05 ***
## Length         0.04661    0.08421   0.554   0.5830
## Spans_c(1,2]  11.49969   24.39999   0.471   0.6401
## Spans_c(2,3]  36.87059   28.36739   1.300   0.2013
## Spans_c(3,7]  67.98083   38.90807   1.747   0.0885 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 55.09 on 39 degrees of freedom
## Multiple R-squared:  0.7124, Adjusted R-squared:  0.6755
## F-statistic: 19.32 on 5 and 39 DF,  p-value: 1.234e-09
```

```
mod4c<-update(mod3c,~.-Length)
summary(mod4c)
```

```
##
## Call:
## lm(formula = Time ~ Dwgs + Spans_c, data = puentes3, na.action = na.exclude)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -87.42 -26.10 -10.87  24.05 179.91
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -17.929     23.402  -0.766   0.4481
```

```
## Dwgs           19.820      3.646   5.436 2.93e-06 ***
## Spans_c(1,2]   13.390     23.950   0.559   0.5792
## Spans_c(2,3]   38.095     28.035   1.359   0.1818
## Spans_c(3,7]   83.512     26.721   3.125   0.0033 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 54.62 on 40 degrees of freedom
## Multiple R-squared:  0.7101, Adjusted R-squared:  0.6811
## F-statistic: 24.49 on 4 and 40 DF,  p-value: 2.673e-10
```

```
# Vemos que en la variable dummy, el único factor significativo es el de (3,7)
```

```
# otra forma de hacerlo
puentes2$Spans_2c<-factor(puentes2$Spans)
puentes3<-subset(puentes2,select = c(-Spans,-Spans_c))
mod1c <- lm(Time ~ ., data=puentes3, na.action=na.exclude)
summary(mod1c)
```

```
##
## Call:
## lm(formula = Time ~ ., data = puentes3, na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -92.669 -32.999  -3.492  22.520 167.788
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -47.5013    29.4804  -1.611    0.116
## DArea          1.3365     1.8274   0.731    0.470
## CCost         -0.1150     0.0923  -1.246    0.221
## Dwgs          23.5090     5.2154   4.508  7.4e-05 ***
## Length         0.2029     0.1376   1.475    0.149
## Spans_2c2      9.7268    25.7756   0.377    0.708
## Spans_2c3     10.4329    31.7263   0.329    0.744
## Spans_2c4     10.5726    59.5016   0.178    0.860
## Spans_2c5     35.1505    53.6653   0.655    0.517
## Spans_2c6     62.8414    57.7390   1.088    0.284
## Spans_2c7   -115.6267   115.1117  -1.004    0.322
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 55.46 on 34 degrees of freedom
## Multiple R-squared:  0.7459, Adjusted R-squared:  0.6712
## F-statistic: 9.982 on 10 and 34 DF,  p-value: 1.532e-07
```

```
mod2c<-update(mod1c,~.-DArea)
summary(mod2c)
```

```
##
## Call:
## lm(formula = Time ~ CCost + Dwgs + Length + Spans_2c, data = puentes3,
##     na.action = na.exclude)
##
## Residuals:
```

```
##     Min      1Q  Median      3Q     Max
## -95.410 -29.620  -4.365  24.173 164.848
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -42.64840   28.53255  -1.495    0.144
## CCost        -0.06792    0.06564  -1.035    0.308
## Dwgs         23.13404    5.15556   4.487 7.47e-05 ***
## Length        0.16640    0.12733   1.307    0.200
## Spans_2c2     8.58821   25.55702   0.336    0.739
## Spans_2c3    16.83654   30.29098   0.556    0.582
## Spans_2c4    27.86065   54.24115   0.514    0.611
## Spans_2c5    47.36390   50.66112   0.935    0.356
## Spans_2c6    82.08383   51.05400   1.608    0.117
## Spans_2c7   -77.08061  101.65633  -0.758    0.453
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 55.09 on 35 degrees of freedom
## Multiple R-squared:  0.7419, Adjusted R-squared:  0.6756
## F-statistic: 11.18 on 9 and 35 DF,  p-value: 5.59e-08
```

```
mod3c<-update(mod2c,~.-CCost)
summary(mod3c)
```

```
##
## Call:
## lm(formula = Time ~ Dwgs + Length + Spans_2c, data = puentes3,
##     na.action = na.exclude)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -101.041  -23.297   -7.408   24.696  171.703
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -31.32956   26.37790  -1.188   0.2427
## Dwgs         20.94701    4.70693   4.450 7.95e-05 ***
## Length        0.08605    0.10101   0.852   0.3999
## Spans_2c2     3.12719   25.03057   0.125   0.9013
## Spans_2c3    22.78589   29.76936   0.765   0.4490
## Spans_2c4    16.59676   53.18948   0.312   0.7568
## Spans_2c5    48.25907   50.70334   0.952   0.3475
## Spans_2c6    89.55922   50.58970   1.770   0.0851 .
## Spans_2c7   -35.18971   93.33725  -0.377   0.7084
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 55.14 on 36 degrees of freedom
## Multiple R-squared:  0.734,  Adjusted R-squared:  0.6749
## F-statistic: 12.42 on 8 and 36 DF,  p-value: 2.47e-08
```

```
mod4c<-update(mod3c,~.-Length)
summary(mod4c)
```

```
##
## Call:
## lm(formula = Time ~ Dwgs + Spans_2c, data = puentes3, na.action = na.exclude)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -104.341  -23.877   -0.056   28.466  175.742
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -34.064     26.085  -1.306   0.1996
## Dwgs          22.741      4.194   5.422  3.8e-06 ***
## Spans_2c2      7.252     24.467   0.296   0.7686
## Spans_2c3     26.273     29.377   0.894   0.3769
## Spans_2c4     34.073     48.892   0.697   0.4902
## Spans_2c5     81.627     32.077   2.545   0.0152 *
## Spans_2c6    113.437     41.959   2.704   0.0103 *
## Spans_2c7     18.250     68.856   0.265   0.7924
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 54.94 on 37 degrees of freedom
## Multiple R-squared:  0.7287, Adjusted R-squared:  0.6774
## F-statistic:  14.2 on 7 and 37 DF,  p-value: 8.718e-09
```

```
# Y vemos que los tramos 5 y 6 son significativos. Cuidado con quitar el intercepto.
# Vemos que con estos factores el R^2 ajustado es menor que el anterior, dado que hemos introducido más
```

```
# Veamos que sucede si quito el intercepto en este caso:
mod4c_Sin<-update(mod3c,~.-Length-1)
summary(mod4c_Sin)
```

```
##
## Call:
## lm(formula = Time ~ Dwgs + Spans_2c - 1, data = puentes3, na.action = na.exclude)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -104.341  -23.877   -0.056   28.466  175.742
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## Dwgs       22.740909   4.193973   5.422  3.8e-06 ***
## Spans_2c1 -34.064068  26.084630  -1.306    0.200
## Spans_2c2 -26.811931  37.415266  -0.717    0.478
## Spans_2c3  -7.791557  45.194468  -0.172    0.864
## Spans_2c4   0.009093  63.575520   0.000    1.000
## Spans_2c5  47.562501  44.177654   1.077    0.289
## Spans_2c6  79.372729  51.329273   1.546    0.131
## Spans_2c7 -15.813634  83.519824  -0.189    0.851
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 54.94 on 37 degrees of freedom
## Multiple R-squared:  0.924,  Adjusted R-squared:  0.9076
```

```
## F-statistic: 56.23 on 8 and 37 DF,  p-value: < 2.2e-16
```

```
# Como se puede observar lo que estaba en el intercepto pasa a ser lo que se indica en la primera categ

# Fijaros que ahora sale que ninguna categoría de Span es significativa, esto se debe a que hemos resta
```

```
puentes2$Spans_2c<-cut(puentes2$Spans, breaks=c(0,3,7))
puentes4<-subset(puentes2,select = c(-Spans,-Spans_c))
mod1c <- lm(Time ~ ., data=puentes4, na.action=na.exclude)
summary(mod1c)
```

```
##
## Call:
## lm(formula = Time ~ ., data = puentes4, na.action = na.exclude)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -84.08 -30.08 -11.99  23.84 190.52
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -26.50823   24.48108  -1.083    0.286
## DArea           0.46680    1.64222   0.284    0.778
## CCost          -0.04782    0.06804  -0.703    0.486
## Dwgs           21.95546    4.30383   5.101 9.09e-06 ***
## Length          0.08873    0.09649   0.920    0.363
## Spans_2c(3,7]  47.04808   38.57609   1.220    0.230
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 55.89 on 39 degrees of freedom
## Multiple R-squared:  0.704,  Adjusted R-squared:  0.666
## F-statistic: 18.55 on 5 and 39 DF,  p-value: 2.125e-09
```

```
mod2c<-update(mod1c,~.-DArea)
summary(mod2c)
```

```
##
## Call:
## lm(formula = Time ~ CCost + Dwgs + Length + Spans_2c, data = puentes4,
##     na.action = na.exclude)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -85.84 -30.54 -10.95  24.23 189.54
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -26.68091   24.19071  -1.103    0.277
## CCost          -0.03554    0.05196  -0.684    0.498
## Dwgs           22.15653    4.19624   5.280 4.83e-06 ***
## Length          0.08235    0.09277   0.888    0.380
## Spans_2c(3,7]  51.51338   34.82544   1.479    0.147
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 55.25 on 40 degrees of freedom
## Multiple R-squared:  0.7034, Adjusted R-squared:  0.6737
## F-statistic: 23.71 on 4 and 40 DF,  p-value: 4.194e-10
```

```
mod3c<-update(mod2c,~.-CCost)
summary(mod3c)
```

```
##
## Call:
## lm(formula = Time ~ Dwgs + Length + Spans_2c, data = puentes4,
##     na.action = na.exclude)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -85.49 -29.31 -10.27  23.91 194.13
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -22.41537   23.22113  -0.965    0.340
## Dwgs           20.97797    3.80136   5.519 2.09e-06 ***
## Length          0.05482    0.08304   0.660    0.513
## Spans_2c(3,7]  48.15565   34.25338   1.406    0.167
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 54.89 on 41 degrees of freedom
## Multiple R-squared:  0.6999, Adjusted R-squared:  0.6779
## F-statistic: 31.87 on 3 and 41 DF,  p-value: 8.457e-11
```

```
mod4c<-update(mod3c,~.-Length)
summary(mod4c)
```

```
##
## Call:
## lm(formula = Time ~ Dwgs + Spans_2c, data = puentes4, na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -93.266 -29.316  -5.228  23.822 196.523
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -25.371     22.632  -1.121  0.26864
## Dwgs            22.450      3.058   7.341 4.75e-09 ***
## Spans_2c(3,7]   65.239     22.293   2.926  0.00551 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 54.52 on 42 degrees of freedom
## Multiple R-squared:  0.6967, Adjusted R-squared:  0.6823
## F-statistic: 48.24 on 2 and 42 DF,  p-value: 1.316e-11
```

```
# Y vemos que la categoria >3 es significativa. Cuidado con quitar el intercepto
# cambiarían los resultados por completo.
```

Ejercicio 2

En el banco de datos **diabetes**, que contiene datos sobre la mortalidad por dicha enfermedad se pretende estudiar el efecto del hábito tabáquico *TABAC* sobre la edad de diagnóstico de la diabetes *EDATDIAG* . Justifica la elección de variables explicativas de entre las disponibles:

- Mortalidad por diabetes, variable *MORT*
- Tiempo de vida en meses tras el diagnóstico, variable *TEMPSVIU*
- Edad del paciente, variable *EDAT*
- Índice de masa corporal, variable *BMI*
- Resultado del electrocardiograma, variable *ECG*
- Antecedentes coronarios, variable *CHD*
- Presión arterial sistólica y diastólica, variables *SBP* y *DBP*, respectivamente

```
# Quitamos las variables que sabemos que no son necesarias
diabetes2<-subset(diabetes,select = c(-NUMPACIE,-MORT))
```

a) Ajusta un modelo simple para contestar la pregunta de investigación. Indica la bondad del ajuste e interpreta el efecto.

```
mod1 <- lm(EDATDIAG ~ TABAC, data=diabetes2, na.action=na.exclude)
summary(mod1)
```

```
##
## Call:
## lm(formula = EDATDIAG ~ TABAC, data = diabetes2, na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -21.488  -6.353  -1.488   6.491  32.491
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)       48.509      1.367  35.480  < 2e-16 ***
## TABACfumador       0.979      2.114   0.463    0.644
## TABACex-fumador   -8.156      1.990  -4.099 6.85e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.32 on 146 degrees of freedom
## Multiple R-squared:  0.1377, Adjusted R-squared:  0.1259
## F-statistic: 11.66 on 2 and 146 DF,  p-value: 2.012e-05
```

```
# Se espera que el tiempo promedio de diagnóstico en no fumadores sea 48.50877

# Si tenemos dos pacientes se espera que el tiempo promedio de la edad de diagnóstico del que fuma sea

# Si tenemos dos pacientes se espera que el tiempo promedio de la edad de diagnóstico del exfumador sea

# En este caso la bondad de ajuste es de 0.1259. Es decir, el 12.59% de variabilidad de la edad de diag
```

b) Los resultados del modelo anterior sugieren alguna simplificación de la variable explicativa? Si es así realizala.

```
library(dplyr)
# Como la variable fumador no es significativa, se puede simplificar en dos variables.
# Primero vamos a considerar los siguientes grupos {No fumador} y {fumador, ex-fumador}
diabetes2$TABAC2<-recode(diabetes$TABAC, 'No fumador'="No fumador", .default="Otro")
```

```r
mod2<-lm(EDATDIAG~TABAC2, data=diabetes2, na.action=na.exclude)
summary(mod2)
```

```
##
## Call:
## lm(formula = EDATDIAG ~ TABAC2, data = diabetes2, na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -20.509  -7.509  -1.424   5.491  35.576
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   48.509      1.443  33.611   <2e-16 ***
## TABAC2Otro    -4.085      1.837  -2.224   0.0277 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.9 on 147 degrees of freedom
## Multiple R-squared:  0.03255,    Adjusted R-squared:  0.02597
## F-statistic: 4.946 on 1 and 147 DF,  p-value: 0.02767
```

```r
AIC(mod1,mod2)
```

```
##      df      AIC
## mod1  4 1123.435
## mod2  3 1138.576
```

```r
# Observamos que aunque ahora no hay ninguna categoria no significativa, el modelo es menos explicativo

diabetes2$TABAC22<-recode(diabetes$TABAC, 'ex-fumador'="ex-fumador", .default="Otro")
mod3<-lm(EDATDIAG~TABAC22, data=diabetes2, na.action=na.exclude)
summary(mod3)
```

```
##
## Call:
## lm(formula = EDATDIAG ~ TABAC22, data = diabetes2, na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -20.918  -5.918  -1.353   6.082  32.082
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)         48.918      1.040  47.041  < 2e-16 ***
## TABAC22ex-fumador   -8.565      1.777  -4.819 3.56e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.29 on 147 degrees of freedom
## Multiple R-squared:  0.1364, Adjusted R-squared:  0.1305
## F-statistic: 23.22 on 1 and 147 DF,  p-value: 3.561e-06
```

```r
# Vemos que el R^2 ha disminuido con respuesto al modelo sin simplificar, pero el R^2 ajustado ha mejor
AIC(mod1,mod2,mod3)
```

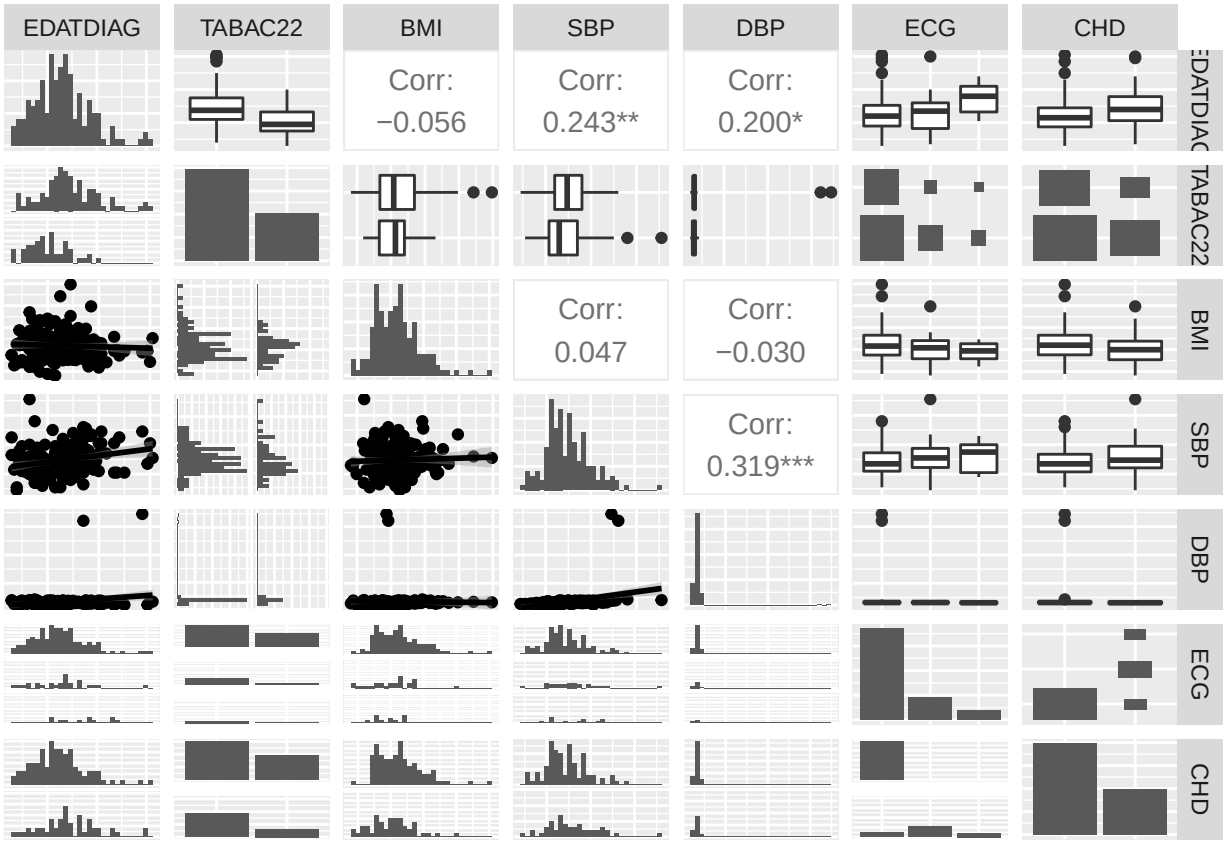
```
##      df      AIC
## mod1  4 1123.435
## mod2  3 1138.576
## mod3  3 1121.654
```

```
# Vemos que el tercer modelo no solo mejora el R^2 ajustado si no que también el AIC, por lo que nos qu
```

c) Valora qué variables de la base de datos deberían ser consideradas como potenciales confusores y evalúa la posible confusión causada por cada una de ellas. ¿Cuál es tu modelo final?

```
library(GGally)
```

```
# Confusores son aquellos que afectan a ambas variables X=TABAC22 e Y=EDATDIAG
ggpairs(diabetes2[,c('EDATDIAG','TABAC22', 'BMI','SBP','DBP','ECG','CHD')],
        lower = list(continuous = "smooth"),
        diag = list(continuous = "barDiag"), axisLabels = "none")
```

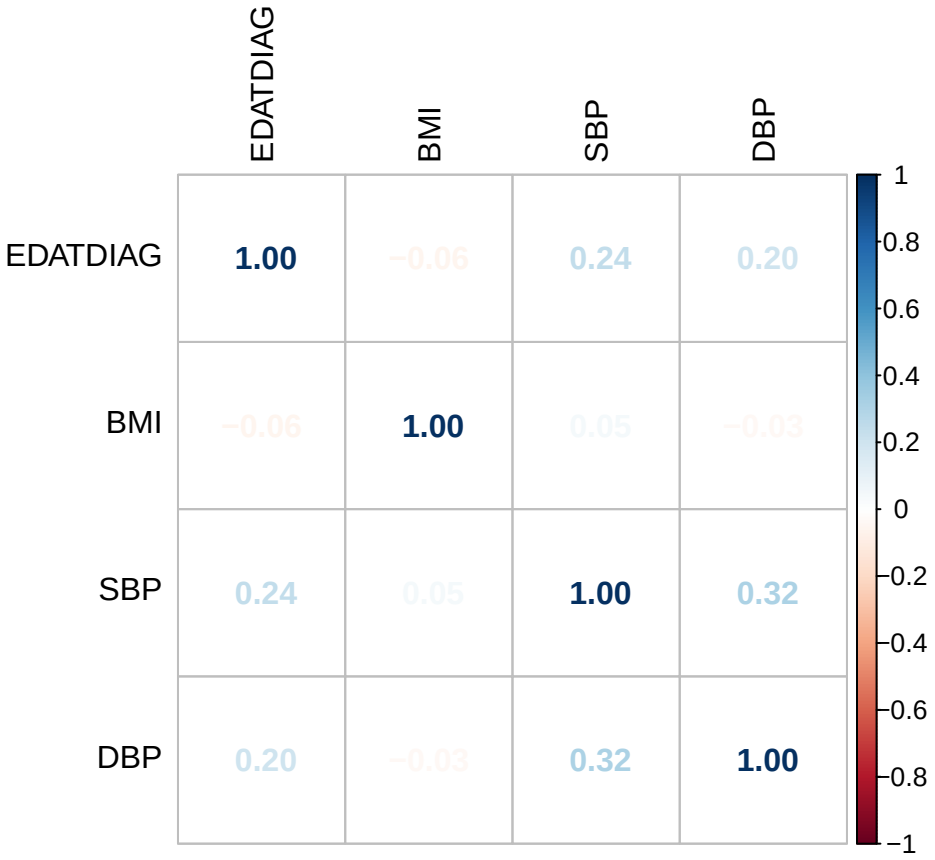



```
# Descartamos BMI como confusora dado que la correlación con la variable de interés no es alta. No hay
```

```
library(corrplot)
```

```
## Warning: package 'corrplot' was built under R version 4.1.3
```

```
corrplot(cor(diabetes2[,c('EDATDIAG','BMI','SBP','DBP')]), method = "number", tl.col = "black")
```




```
# vamos a ver que variables pueden confundir. Lo que vamos a hacer es ver
# como afectan al coeficiente del Tabac22.ex-fumador
mod3a<-update(mod3, ~.+SBP)
summary(mod3a)
```

```
##
## Call:
## lm(formula = EDATDIAG ~ TABAC22 + SBP, data = diabetes2, na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -21.430  -5.720  -1.441   6.374  31.807
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       31.63363    5.78445   5.469 1.92e-07 ***
## TABAC22ex-fumador -8.33132    1.73155  -4.811 3.70e-06 ***
## SBP                0.12366    0.04075   3.035  0.00285 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.02 on 146 degrees of freedom
## Multiple R-squared:  0.1877, Adjusted R-squared:  0.1765
## F-statistic: 16.86 on 2 and 146 DF,  p-value: 2.574e-07
```

```r
100*abs(coef(mod3a)["TABAC22ex-fumador"]-coef(mod3)["TABAC22ex-fumador"])/abs(coef(mod3))["TABAC22ex-fu
```

```
## TABAC22ex-fumador
##          2.733187
```

```r
# SBP no confunde: apenas un cambio del 3%

mod3a<-update(mod3, ~.+DBP)
summary(mod3a)
```

```
##
## Call:
## lm(formula = EDATDIAG ~ TABAC22 + DBP, data = diabetes2, na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -20.332  -6.332  -1.375   6.239  32.239
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       46.870229   1.372269  34.155  < 2e-16 ***
## TABAC22ex-fumador -8.234811   1.759656  -4.680 6.49e-06 ***
## DBP                0.021490   0.009563   2.247   0.0261 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.16 on 146 degrees of freedom
## Multiple R-squared:  0.1653, Adjusted R-squared:  0.1539
## F-statistic: 14.46 on 2 and 146 DF,  p-value: 1.871e-06
```

```r
100*abs(coef(mod3a)["TABAC22ex-fumador"]-coef(mod3)["TABAC22ex-fumador"])/abs(coef(mod3))["TABAC22ex-fu
```

```
## TABAC22ex-fumador
##          3.859885
```

```r
# DBP no confunde: apenas un cambio del 4%


mod3a<-update(mod3, ~.+ECG)
summary(mod3a)
```

```
##
## Call:
## lm(formula = EDATDIAG ~ TABAC22 + ECG, data = diabetes2, na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -20.407  -5.951  -1.407   6.491  32.593
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        48.4066     1.1664  41.502  < 2e-16 ***
## TABAC22ex-fumador  -8.4554     1.7582  -4.809 3.76e-06 ***
## ECGFrontera        -0.8979     2.1832  -0.411  0.68147
## ECGAnormal          8.6267     3.1959   2.699  0.00778 **
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.09 on 145 degrees of freedom
## Multiple R-squared:  0.1811, Adjusted R-squared:  0.1642
## F-statistic: 10.69 on 3 and 145 DF,  p-value: 2.163e-06
```

```
100*abs(coef(mod3a)["TABAC22ex-fumador"]-coef(mod3)["TABAC22ex-fumador"])/abs(coef(mod3))["TABAC22ex-fu
```

```
## TABAC22ex-fumador
##          1.284141
```

```
#ECG no confunde: un cambio de poco más del 1%


mod3a<-update(mod3, ~.+CHD)
summary(mod3a)
```

```
##
## Call:
## lm(formula = EDATDIAG ~ TABAC22 + CHD, data = diabetes2, na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -22.867  -6.686  -1.555   5.445  33.264
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)         47.736      1.234  38.697  < 2e-16 ***
## TABAC22ex-fumador   -8.181      1.779  -4.600  9.1e-06 ***
## CHDSi                3.131      1.787   1.752   0.0819 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.22 on 146 degrees of freedom
## Multiple R-squared:  0.1542, Adjusted R-squared:  0.1426
## F-statistic: 13.31 on 2 and 146 DF,  p-value: 4.906e-06
```

```
100*abs(coef(mod3a)["TABAC22ex-fumador"]-coef(mod3)["TABAC22ex-fumador"])/abs(coef(mod3))["TABAC22ex-fu
```

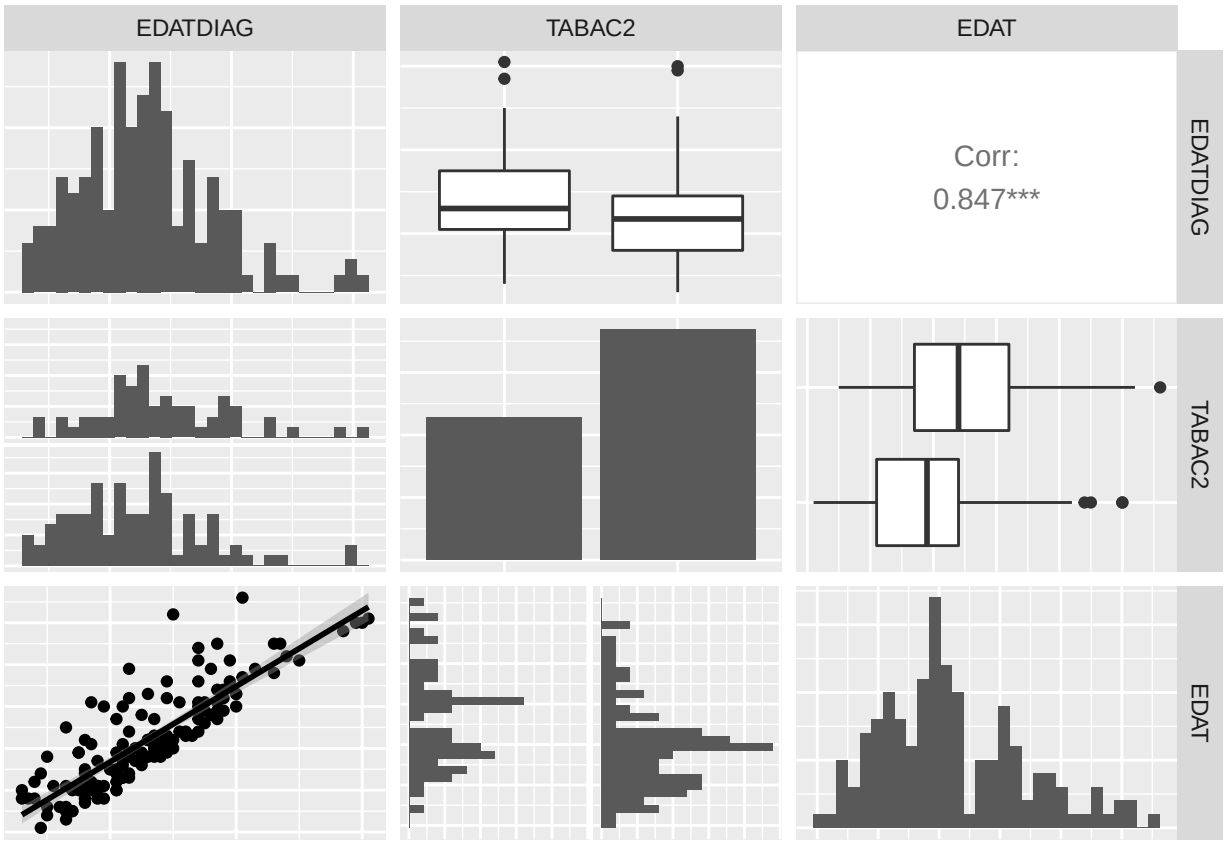
```
## TABAC22ex-fumador
##          4.483113
```

```
#CHD no confunde: apenas un cambio del 4.5%
```

d) Caso de ser lícito considerar la variable *EDAT* como potencial confusor, analiza su efecto.

```
diabetes2$EDAT<-diabetes$EDAT

ggpairs(diabetes2[,c('EDATDIAG','TABAC2', 'EDAT')],
        lower = list(continuous = "smooth"), diag = list(continuous = "barDiag"),
        axisLabels = "none")
```




```
# Podría ser lícito, por la gran correlación que tiene con EDATDIAG y el efecto con TABAC2

mod3a<-update(mod3, ~.+EDAT)
summary(mod3a)
```

```
##
## Call:
## lm(formula = EDATDIAG ~ TABAC22 + EDAT, data = diabetes2, na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -19.579  -2.731   1.269   3.919  12.204
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        5.35139    2.58186   2.073    0.040 *
## TABAC22ex-fumador -0.65993    1.11681  -0.591    0.555
## EDAT               0.78326    0.04516  17.343   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.905 on 146 degrees of freedom
## Multiple R-squared:  0.7178, Adjusted R-squared:  0.7139
## F-statistic: 185.7 on 2 and 146 DF,  p-value: < 2.2e-16
```

```r
100*abs(coef(mod3a)["TABAC22ex-fumador"]-coef(mod3)["TABAC22ex-fumador"])/abs(coef(mod3))["TABAC22ex-fu
```

```
## TABAC22ex-fumador
##          92.29544
```

```r
# Confunde un 92%
```

Ejercicio 3

Usando la base de datos **deportistas**, valora e interpreta la existencia de interacción entre *MCMagra* y *Genero* en la explicación de la variable *PrctGrasa*

```r
mod1<-lm(PrctGrasa~ MCMagra+Genero, data=deportistas, na.action=na.exclude )
summary(mod1)
```

```
##
## Call:
## lm(formula = PrctGrasa ~ MCMagra + Genero, data = deportistas,
##     na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.1959 -2.2760 -0.5097  1.7914 17.9355
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   7.72462    1.94170   3.978 9.72e-05 ***
## MCMagra       0.18443    0.03454   5.339 2.53e-07 ***
## Generomale  -12.24299    0.90078 -13.591  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.176 on 199 degrees of freedom
## Multiple R-squared:  0.5493, Adjusted R-squared:  0.5448
## F-statistic: 121.3 on 2 and 199 DF,  p-value: < 2.2e-16
```

```r
modI<-lm(PrctGrasa~ MCMagra*Genero, data=deportistas, na.action=na.exclude )
summary(modI)
```

```
##
## Call:
## lm(formula = PrctGrasa ~ MCMagra * Genero, data = deportistas,
##     na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.3472 -2.4071 -0.7003  1.7473 18.1300
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)         0.28459    3.30006   0.086  0.93137
## MCMagra             0.31997    0.05965   5.364 2.26e-07 ***
## Generomale          0.05873    4.53697   0.013  0.98969
## MCMagra:Generomale -0.20065    0.07258  -2.765  0.00624 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 4.108 on 198 degrees of freedom
## Multiple R-squared:  0.5661, Adjusted R-squared:  0.5595
## F-statistic: 86.09 on 3 and 198 DF,  p-value: < 2.2e-16
```

```
# Este ejercicio lo hemos visto en teoría.
# Si quisieramos tener el modelo solo con la interacciónу MCMagra (Es decir, sin Genero sola)
modI_2<-lm(PrctGrasa~ MCMagra*Genero-Genero, data=deportistas, na.action=na.exclude )
summary(modI_2)
```

```
##
## Call:
## lm(formula = PrctGrasa ~ MCMagra * Genero - Genero, data = deportistas,
##     na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.3430 -2.4017 -0.6981  1.7461 18.1287
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)         0.31566    2.25895   0.140    0.889
## MCMagra             0.31941    0.04118   7.756 4.47e-13 ***
## MCMagra:Generomale -0.19973    0.01414 -14.126  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.098 on 199 degrees of freedom
## Multiple R-squared:  0.5661, Adjusted R-squared:  0.5617
## F-statistic: 129.8 on 2 and 199 DF,  p-value: < 2.2e-16
```

Ejercicio 4

En el banco de datos **Boston** del paquete de R MASS, que contiene datos sobre los suburbios de Boston, queremos analizar el precio medio de la vivienda *medv* respecto del estatus de la población *lstat*. Para ello:

```
library(MASS)
```

```
## Warning: package 'MASS' was built under R version 4.1.3
```

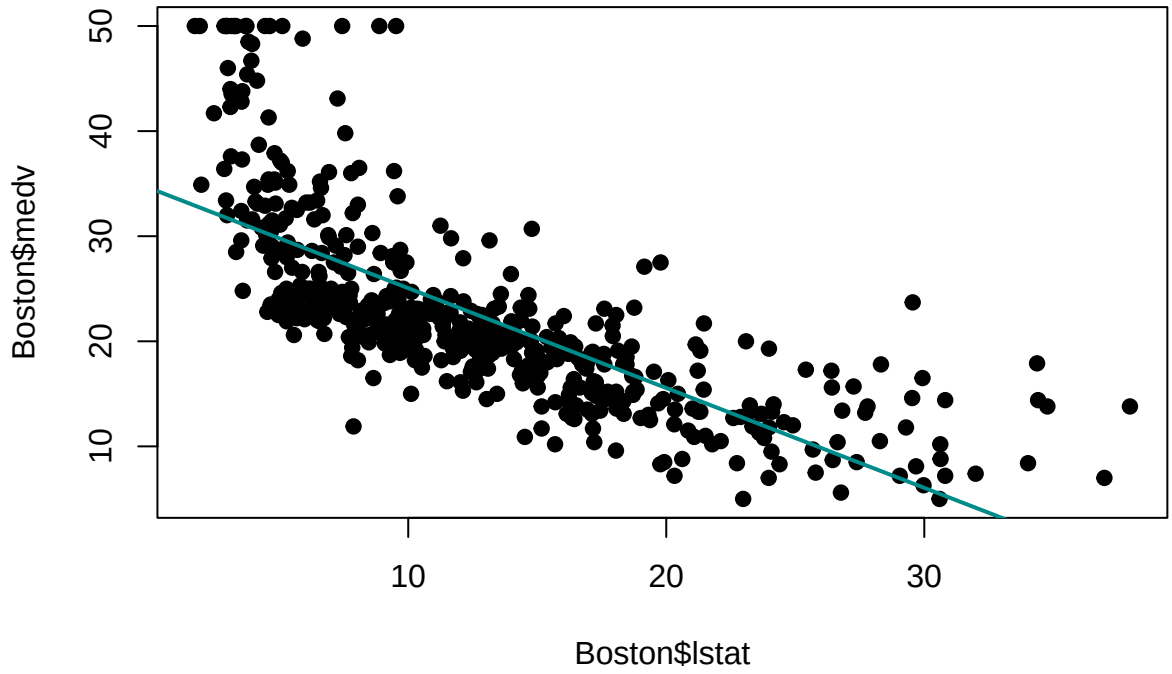
a) Ajusta una recta de regresión a los datos y representa el ajuste ¿Qué comentarías?

```
mod1<-lm(medv~ lstat, data=Boston, na.action=na.exclude )
summary(mod1)
```

```
##
## Call:
## lm(formula = medv ~ lstat, data = Boston, na.action = na.exclude)
##
## 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
```

```
plot(Boston$lstat,Boston$medv, pch=19,type="p")
abline(coef=coef(mod1),col="darkcyan",lwd=2)
```




```
# Aunque los coeficientes salen significativos, gráficamente observamos que sería mejor una regresión n
```

b) Ajusta un modelo parabólico directamente y mediante polinomios ortogonales, compara numérica y gráficamente los dos modelos entre sí y con el modelo lineal (comandos `anova` y `AIC`)

```
mod2a<-lm(medv~lstat+I(lstat^2),data=Boston, na.action=na.exclude)
summary(mod2a)
```

```
##
## Call:
## lm(formula = medv ~ lstat + I(lstat^2), data = Boston, na.action = na.exclude)
##
## 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) 42.862007   0.872084   49.15   <2e-16 ***
## lstat       -2.332821   0.123803  -18.84   <2e-16 ***
## I(lstat^2)   0.043547   0.003745   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
```

```
anova(mod2a,mod1) # El p-valor es pequeño, por lo que rechazamos la hipótesis nula y el coeficiente cua
```

```
## Analysis of Variance Table
##
## Model 1: medv ~ lstat + I(lstat^2)
## Model 2: medv ~ lstat
##   Res.Df   RSS Df Sum of Sq     F    Pr(>F)
## 1    503 15347
## 2    504 19472 -1   -4125.1 135.2 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

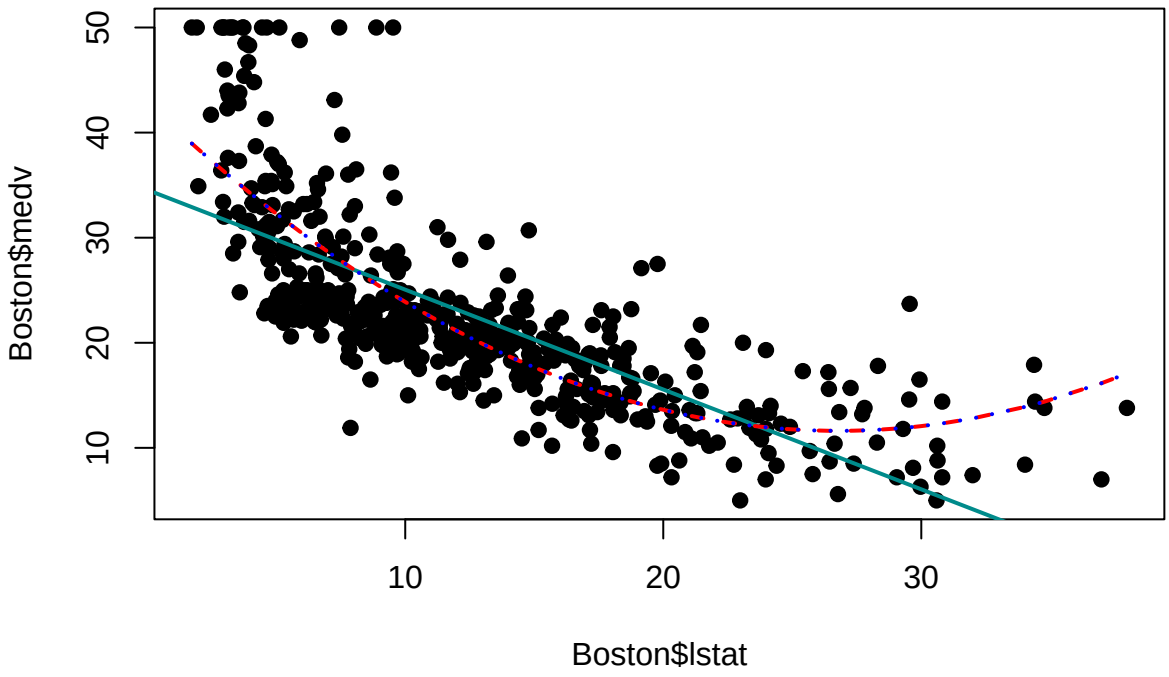
```
mod2b<-lm(medv~poly(lstat,2),data=Boston, na.action=na.exclude)
summary(mod2b)
```

```
##
## Call:
## lm(formula = medv ~ poly(lstat, 2), data = Boston, na.action = na.exclude)
##
## 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
```

```
anova(mod2b,mod1)
```

```
## Analysis of Variance Table
##
## Model 1: medv ~ poly(lstat, 2)
## Model 2: medv ~ lstat
##   Res.Df   RSS Df Sum of Sq     F    Pr(>F)
## 1    503 15347
## 2    504 19472 -1   -4125.1 135.2 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
plot(Boston$lstat,Boston$medv, pch=19,type="p")
abline(coef=coef(mod1),col="darkcyan",lwd=2)
lines(sort(Boston$lstat),fitted(mod2a)[order(Boston$lstat)],col="red", lwd=2,lty=2)
lines(sort(Boston$lstat),fitted(mod2b)[order(Boston$lstat)],col="blue", lwd=2,lty=3)
```




```
AIC(mod1);AIC(mod2a);AIC(mod2b)
```

```
## [1] 3288.975
```

```
## [1] 3170.516
```

```
## [1] 3170.516
```

c) ¿Mejoraría el modelo con un polinomio de orden superior? Inténtalo usando el comando poly y representa el ajuste del modelo polinómico elegido.

```
#Lo hacemos con el b, para comparar la introducción de polinomios ortogonales.
mod3b<-lm(medv~poly(lstat,3),data=Boston, na.action=na.exclude)
anova(mod2b,mod3b,test="F")# El p-valor es pequeño, por lo que rechazamos la hipótesis nula y el coefic
```

```
## Analysis of Variance Table
##
## Model 1: medv ~ poly(lstat, 2)
## Model 2: medv ~ poly(lstat, 3)
##   Res.Df   RSS Df Sum of Sq      F    Pr(>F)
## 1    503 15347
## 2    502 14616  1    731.76 25.134 7.428e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
mod4b<-lm(medv~poly(lstat,4),data=Boston, na.action=na.exclude)
anova(mod4b,mod3b,test="F")# El p-valor es pequeño, por lo que rechazamos la hipótesis nula y el coefic
```

```
## Analysis of Variance Table
##
## Model 1: medv ~ poly(lstat, 4)
## Model 2: medv ~ poly(lstat, 3)
##   Res.Df   RSS Df Sum of Sq      F    Pr(>F)
## 1    501 13968
## 2    502 14616 -1   -647.79 23.235 1.904e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
mod5b<-lm(medv~poly(lstat,5),data=Boston, na.action=na.exclude)
anova(mod4b,mod5b,test="F")# El p-valor es pequeño, por lo que rechazamos la hipótesis nula y el coefic
```

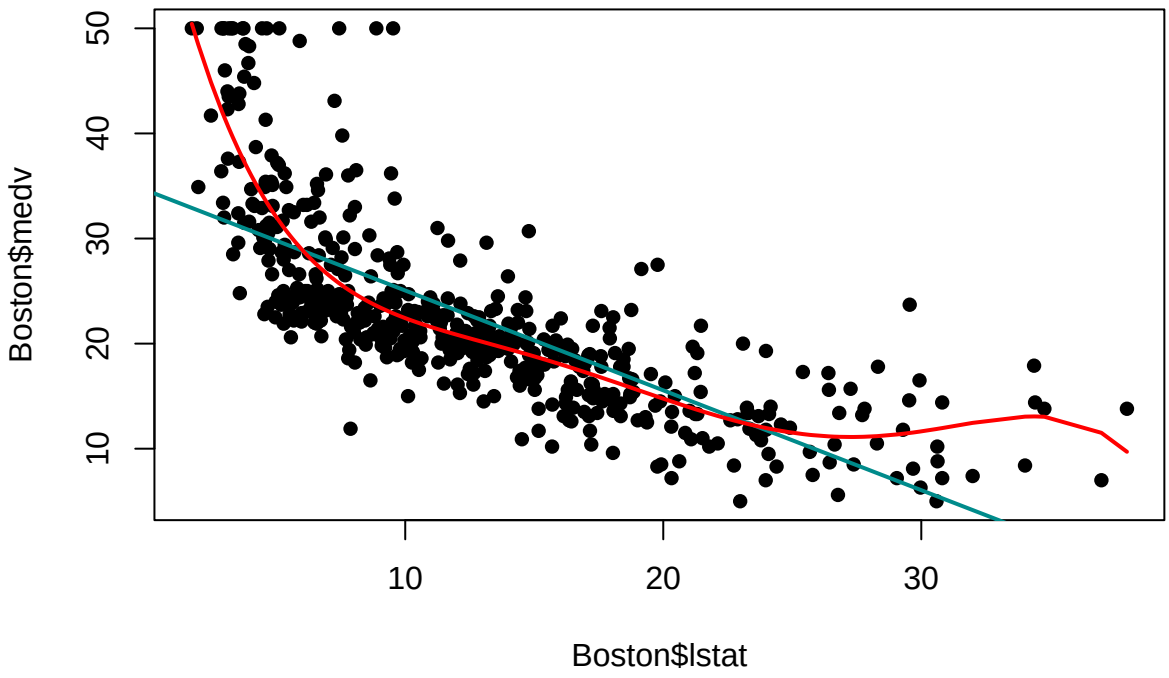
```
## Analysis of Variance Table
##
## Model 1: medv ~ poly(lstat, 4)
## Model 2: medv ~ poly(lstat, 5)
##   Res.Df   RSS Df Sum of Sq     F    Pr(>F)
## 1    501 13968
## 2    500 13597  1    370.66 13.63 0.0002471 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
mod6b<-lm(medv~poly(lstat,6),data=Boston, na.action=na.exclude)
anova(mod6b,mod5b,test="F")# El p-valor es grande, por lo que no rechazamos la hipótesis nula y el coef
```

```
## Analysis of Variance Table
##
## Model 1: medv ~ poly(lstat, 6)
## Model 2: medv ~ poly(lstat, 5)
##   Res.Df   RSS Df Sum of Sq      F Pr(>F)
## 1    499 13555
## 2    500 13597 -1   -42.364 1.5596 0.2123
```

```r
# Podríamos haber hecho un bucle con un criterio de parada adecuado.
```

```r
# Nos quedamos con el modelo de orden 5
plot(Boston$lstat,Boston$medv, pch=16,type="p")
abline(coef=coef(mod1),col="darkcyan",lwd=2)
lines(sort(Boston$lstat),fitted(mod5b)[order(Boston$lstat)],col="red", lwd=2)
```



Ejercicio 5

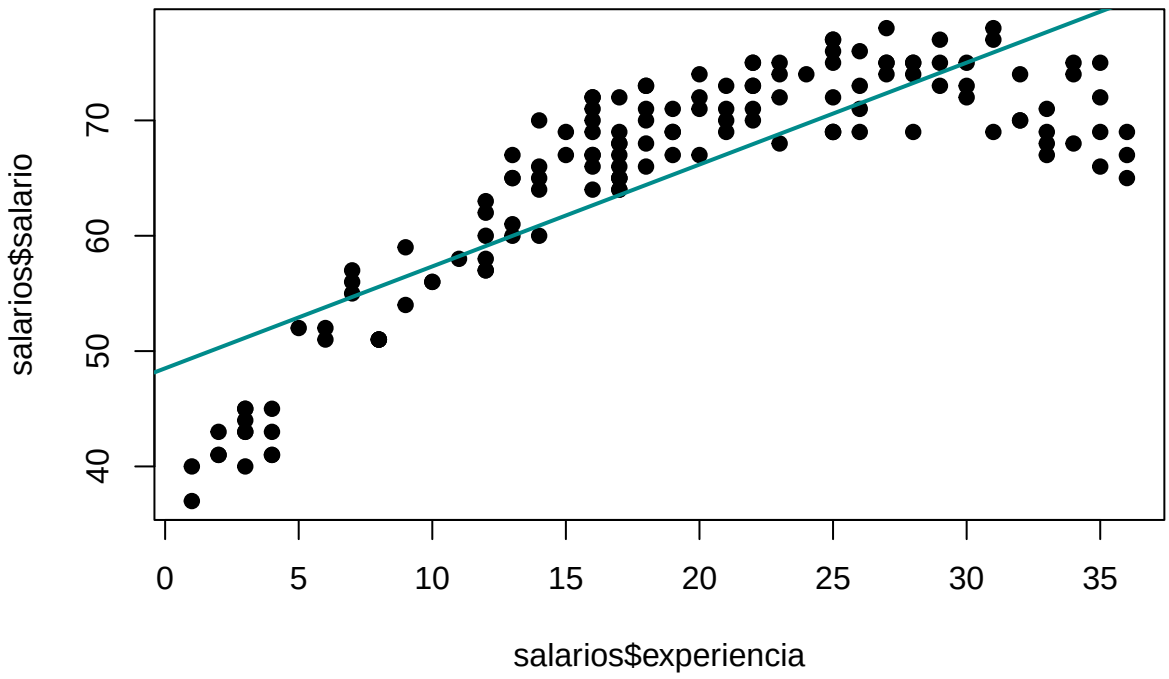
Las organizaciones profesionales de contables, ingenieros, etc., realizan encuestas regularmente entre sus miembros para conseguir información relativa a los salarios, las pensiones y las condiciones de empleo. Uno de los resultados de estas encuestas es la llamada curva de salario, que relaciona el sueldo con los años de experiencia. La curva salarial muestra el salario "típico'' de los profesionales con un determinado número de años de experiencia. Es de gran interés para los miembros de la profesión, pues les gusta saber dónde están situados entre sus pares. También es útil para los departamentos de personal de las empresas, para realizar ajustes de sueldos o para contratar a nuevos profesionales. Modeliza la curva salarial, con los datos de la base **salarios**.

```
mod1<-lm(salario~experiencia,data=salarios, na.action=na.exclude)
summary(mod1)
```

```
##
## Call:
## lm(formula = salario ~ experiencia, data = salarios, na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.310  -3.893   1.408   4.442   9.359
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 48.50593    1.08810   44.58   <2e-16 ***
## experiencia  0.88345    0.05158   17.13   <2e-16 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.828 on 141 degrees of freedom
## Multiple R-squared:  0.6754, Adjusted R-squared:  0.6731
## F-statistic: 293.3 on 1 and 141 DF,  p-value: < 2.2e-16
```

```
plot(salarios$experiencia,salarios$salario, pch=19,type="p")
abline(coef=coef(mod1),col="darkcyan",lwd=2)
```




```
mod2<-lm(salario~experiencia+I(experiencia^2),data=salarios, na.action=na.exclude)
anova(mod2,mod1)
```

```
## Analysis of Variance Table
##
## Model 1: salario ~ experiencia + I(experiencia^2)
## Model 2: salario ~ experiencia
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1    140 1111.2
## 2    141 4789.0 -1   -3677.9 463.38 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
mod3<-lm(salario~experiencia+I(experiencia^2)+I(experiencia^3),data=salarios, na.action=na.exclude)
anova(mod2,mod3)
```

```
## Analysis of Variance Table
##
## Model 1: salario ~ experiencia + I(experiencia^2)
```

```
## Model 2: salario ~ experiencia + I(experiencia^2) + I(experiencia^3)
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1    140 1111.2
## 2    139 1096.4  1      14.8 1.8764  0.173
```

```
#Nos quedamos con el ajuste parabólico (gráficamente se podía intuir).
```

Ejercicio 6

Usando la base **diabetes**,

a) Ajusta el mejor modelo posible para predecir la edad al diagnóstico *EDATDIAG*, usando el comando `regsubset` y basándote en el criterio de mínimo Akaike.

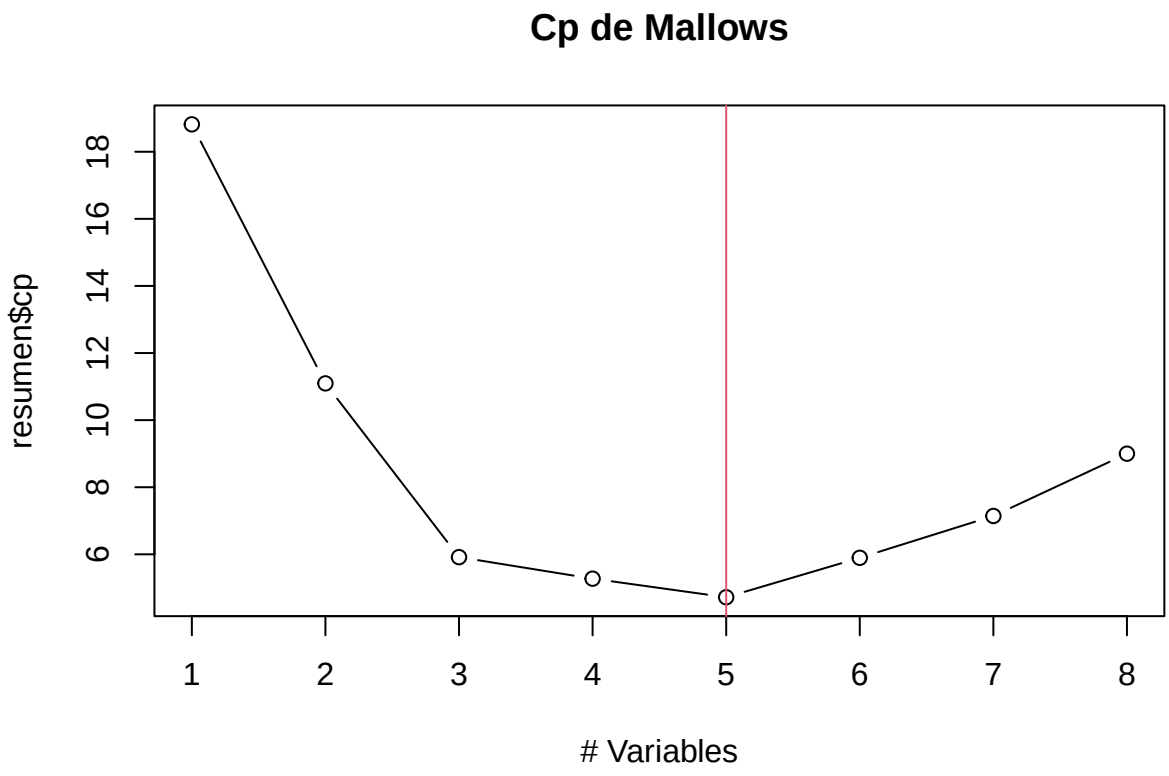
```
library(leaps)
```

```
## Warning: package 'leaps' was built under R version 4.1.3
```

```
sel_lm <- regsubsets(EDATDIAG ~ .-EDAT-TEMPSVIU-MORT-NUMPACIE , data=diabetes, nvmax=10)
resumen<-summary(sel_lm)

resultado <- cbind(resumen$rsq,resumen$adjr2,resumen$cp,resumen$bic)
colnames(resultado) <- c('Rsq','RsqAdj','Cp','BIC')

# Dibujamos los valores obtenidos del Cp (que es el que se basa en el criterio
# de Akaike)
plot(1:8, resumen$cp, xlab = "# Variables", main = "Cp de Mallows",
     type='b')
abline(v = which.min(resumen$cp), col = 2)
```




```
colnames(resumen$which)[resumen$which[5,]==T] # Para saber cuales son las variables
```

```
## [1] "(Intercept)"     "TABACex-fumador" "SBP"             "DBP"
## [5] "ECGFrontera"     "CHDSi"
```

```
bestAIC<-lm(EDATDIAG ~TABAC+SBP+DBP+ECG+CHD,data=diabetes,na.action=na.exclude )
AIC(bestAIC)
```

```
## [1] 1111.087
```

```
summary(bestAIC)
```

```
##
## Call:
## lm(formula = EDATDIAG ~ TABAC + SBP + DBP + ECG + CHD, data = diabetes,
##     na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -19.635  -5.589  -0.889   5.887  33.287
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     33.13372    5.73367   5.779 4.63e-08 ***
## TABACfumador     0.64811    2.02654   0.320   0.7496
## TABACex-fumador -7.69934    1.90823  -4.035 8.92e-05 ***
## SBP              0.09224    0.04255   2.168   0.0319 *
## DBP              0.01683    0.00987   1.705   0.0904 .
```

```
## ECGFrontera     -5.73298    3.39926  -1.687   0.0939 .
## ECGAnormal       3.75533    4.07571   0.921   0.3584
## CHDSi            5.13468    3.00611   1.708   0.0898 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.745 on 141 degrees of freedom
## Multiple R-squared:  0.2578, Adjusted R-squared:  0.2209
## F-statistic: 6.996 on 7 and 141 DF,  p-value: 3.715e-07
```

```
# Lo siguiente es para ver que efectivamente la variable ECG es significativa.
prueba<-update(bestAIC,~.-ECG)
anova(prueba,bestAIC,test="F")
```

```
## Analysis of Variance Table
##
## Model 1: EDATDIAG ~ TABAC + SBP + DBP + CHD
## Model 2: EDATDIAG ~ TABAC + SBP + DBP + ECG + CHD
##   Res.Df   RSS Df Sum of Sq      F  Pr(>F)
## 1    143 14171
## 2    141 13390  2    781.69 4.1158 0.01831 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Notar que las variables TABAC y ECG son significativas y tienen
# tres categorías cada una. Es conveniente, como hemos hecho anteriormente,
# crear una base de datos con las variables adecuadas.
# En un ejercicio anterior hemos creado la variable TABAC22 con dos categorías.
bestAIC<-lm(EDATDIAG ~TABAC22+SBP+DBP+ECG+CHD,data=diabetes2,na.action=na.exclude )
prueba<-update(bestAIC,~.-ECG)
anova(prueba,bestAIC,test="F")
```

```
## Analysis of Variance Table
##
## Model 1: EDATDIAG ~ TABAC22 + SBP + DBP + CHD
## Model 2: EDATDIAG ~ TABAC22 + SBP + DBP + ECG + CHD
##   Res.Df   RSS Df Sum of Sq   F Pr(>F)
## 1    144 14192
## 2    142 13399  2    792.62 4.2 0.0169 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(bestAIC)
```

```
##
## Call:
## lm(formula = EDATDIAG ~ TABAC22 + SBP + DBP + ECG + CHD, data = diabetes2,
##     na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -19.907  -5.474  -0.885   5.644  33.024
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       33.29975    5.69204   5.850 3.24e-08 ***
```

```
## TABAC22ex-fumador -7.97272    1.70064  -4.688 6.40e-06 ***
## SBP                0.09317    0.04232   2.202   0.0293 *
## DBP                0.01643    0.00976   1.684   0.0945 .
## ECGFrontera       -5.81952    3.37775  -1.723   0.0871 .
## ECGAnormal         3.70035    4.05919   0.912   0.3635
## CHDSi              5.23038    2.98171   1.754   0.0816 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.714 on 142 degrees of freedom
## Multiple R-squared:  0.2572, Adjusted R-squared:  0.2259
## F-statistic: 8.197 on 6 and 142 DF,  p-value: 1.274e-07
```

```
# Creamos también una variable dicotómica para la variable ECG
diabetes2$ECG2<-recode(diabetes$ECG, 'Normal'="Normal", 'Frontera'="Other",'Anormal'="Other")
prueba1<-update(bestAIC,~.-ECG+ECG2)
summary(prueba1)
```

```
##
## Call:
## lm(formula = EDATDIAG ~ TABAC22 + SBP + DBP + CHD + ECG2, data = diabetes2,
##     na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -19.882  -6.128  -1.011   5.702  33.060
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       33.143531   5.819698   5.695 6.76e-08 ***
## TABAC22ex-fumador -7.862251   1.738383  -4.523 1.27e-05 ***
## SBP                0.094184   0.043266   2.177   0.0311 *
## DBP                0.016166   0.009979   1.620   0.1074
## CHDSi              5.227998   3.048737   1.715   0.0885 .
## ECG2Other         -3.056160   3.295992  -0.927   0.3554
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.932 on 143 degrees of freedom
## Multiple R-squared:  0.218,  Adjusted R-squared:  0.1907
## F-statistic: 7.973 on 5 and 143 DF,  p-value: 1.181e-06
```

```
AIC(bestAIC,prueba1)
```

```
##         df      AIC
## bestAIC  8 1109.195
## prueba1  7 1114.865
```

```
# La categoría Other no es significativa. Veamos que pasa si ponemos por un lado Frontera y por el otro
diabetes2$ECG2<-recode(diabetes$ECG, 'Frontera'="Frontera", .default="Other")
prueba2<-update(bestAIC,~.-ECG+ECG2)
AIC(bestAIC,prueba2)
```

```
##         df      AIC
## bestAIC  8 1109.195
## prueba2  7 1108.065
```

```r
#Hemos ganado

bestAIC<-prueba2
summary(bestAIC)
```

```
##
## Call:
## lm(formula = EDATDIAG ~ TABAC22 + SBP + DBP + CHD + ECG2, data = diabetes2,
##     na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -19.946  -5.768  -0.730   5.865  33.000
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)       33.129373   5.685611   5.827 3.59e-08 ***
## TABAC22ex-fumador -8.013095   1.699063  -4.716 5.65e-06 ***
## SBP                0.094643   0.042261   2.240  0.02667 *
## DBP                0.016261   0.009753   1.667  0.09764 .
## CHDSi              6.986441   2.274564   3.072  0.00255 **
## ECG2Frontera      -7.594142   2.758705  -2.753  0.00668 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.708 on 143 degrees of freedom
## Multiple R-squared:  0.2529, Adjusted R-squared:  0.2268
## F-statistic: 9.681 on 5 and 143 DF,  p-value: 5.576e-08
```

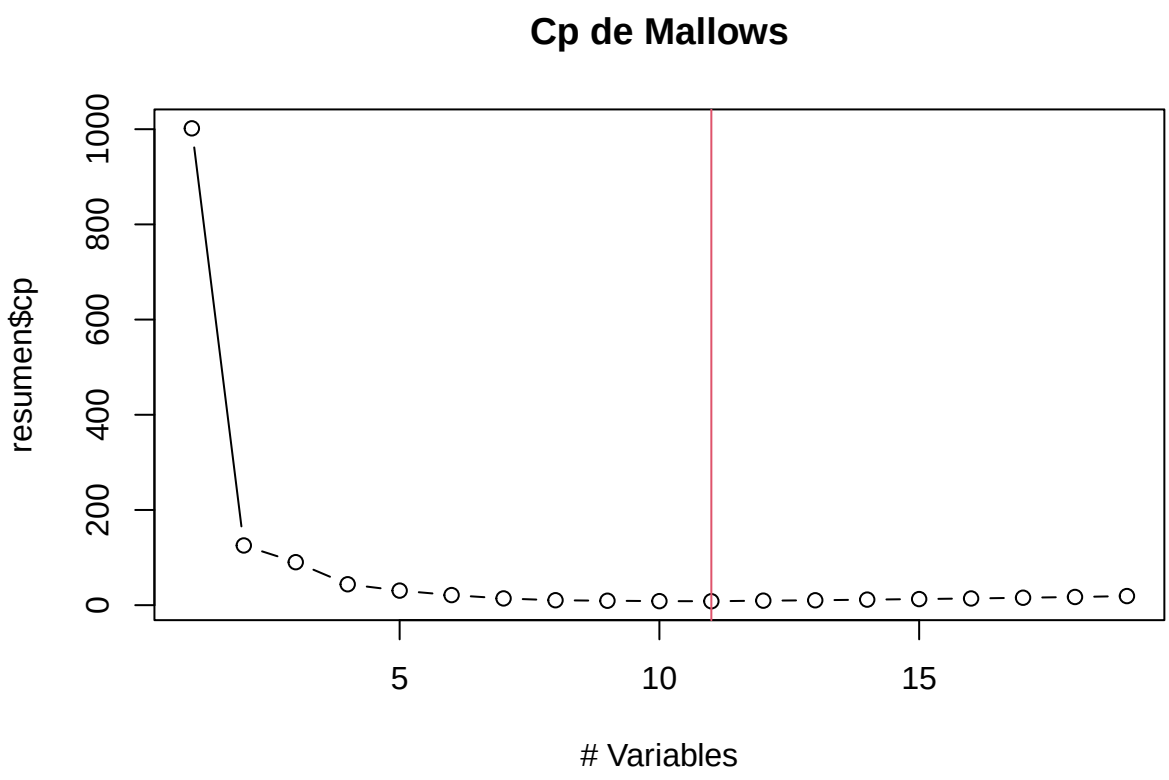
b) Describe el modelo que has seleccionado.

Ejercicio 7

En la base **deportistas**, pretendemos ajustar el mejor modelo predictivo del porcentaje de grasa, usando las variables disponibles. Elige entre los dos métodos `regsubsets` y `step` ¿Cuál has elegido y por qué?

```r
library(leaps)
sel_lm1 <- regsubsets(PrctGrasa ~ . , data=deportistas,nvmax=19)
resumen<-summary(sel_lm1)
resultado <- cbind(resumen$rsq,resumen$adjr2,resumen$cp,resumen$bic)
colnames(resultado) <- c('Rsq','RsqAdj','Cp','BIC')


# Indica el mejor modelo predictivo por lo que nos basamos en el criterio AIC
plot(1:19, resumen$cp, xlab = "# Variables", main = "Cp de Mallows",
     type='b')
abline(v = which.min(resumen$cp), col = 2)
```




```
which.min(resumen$cp)
```

```
## [1] 11
```

```
colnames(resumen$which)[resumen$which[11,]==T]
```

```
##  [1] "(Intercept)"   "SumPliegues"   "MCMagra"       "Peso"
##  [5] "DeporteField"  "DeporteGym"    "DeporteSwim"   "DeporteT_400"
##  [9] "DeporteT_Sprn" "DeporteTennis" "DeporteW_Polo" "Generomale"
```

```
reg_sub<-lm(PrctGrasa ~ SumPliegues+MCMagra+Peso+Deporte+Genero,data=deportistas)
summary(reg_sub)
```

```
##
## Call:
## lm(formula = PrctGrasa ~ SumPliegues + MCMagra + Peso + Deporte +
##     Genero, data = deportistas)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.07388 -0.36600 -0.01114  0.40121  2.43089
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    11.128255   0.568009  19.592  < 2e-16 ***
## SumPliegues     0.037471   0.007493   5.001 1.30e-06 ***
## MCMagra        -0.918983   0.054092 -16.989  < 2e-16 ***
## Peso            0.803138   0.048431  16.583  < 2e-16 ***
```

```
## DeporteField   -0.561987   0.214202  -2.624  0.00941 **
## DeporteGym     -1.200832   0.404845  -2.966  0.00341 **
## DeporteNetball  0.248727   0.216383   1.149  0.25182
## DeporteRow      0.184073   0.176654   1.042  0.29875
## DeporteSwim    -0.382379   0.203511  -1.879  0.06180 .
## DeporteT_400   -0.948195   0.215669  -4.397 1.84e-05 ***
## DeporteT_Sprn  -0.590206   0.237634  -2.484  0.01388 *
## DeporteTennis  -0.337134   0.257544  -1.309  0.19212
## DeporteW_Polo  -0.221257   0.223472  -0.990  0.32340
## Generomale     -1.104796   0.218563  -5.055 1.02e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6765 on 188 degrees of freedom
## Multiple R-squared:  0.9888, Adjusted R-squared:  0.9881
## F-statistic:  1280 on 13 and 188 DF,  p-value: < 2.2e-16
```

```
### El método de regsubsets considerada muchas categorías por separado, demasiadas variables y separa f
```

```
lm_completo<-lm(PrctGrasa ~ . , data=deportistas, na.action=na.exclude)
sel_lm <- step(lm_completo , direction = 'both',trace=0)
summary(sel_lm)
```

```
##
## Call:
## lm(formula = PrctGrasa ~ SumPliegues + MCMagra + Peso + Deporte +
##     Genero, data = deportistas, na.action = na.exclude)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.07388 -0.36600 -0.01114  0.40121  2.43089
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    11.128255   0.568009  19.592  < 2e-16 ***
## SumPliegues     0.037471   0.007493   5.001 1.30e-06 ***
## MCMagra        -0.918983   0.054092 -16.989  < 2e-16 ***
## Peso            0.803138   0.048431  16.583  < 2e-16 ***
## DeporteField   -0.561987   0.214202  -2.624  0.00941 **
## DeporteGym     -1.200832   0.404845  -2.966  0.00341 **
## DeporteNetball  0.248727   0.216383   1.149  0.25182
## DeporteRow      0.184073   0.176654   1.042  0.29875
## DeporteSwim    -0.382379   0.203511  -1.879  0.06180 .
## DeporteT_400   -0.948195   0.215669  -4.397 1.84e-05 ***
## DeporteT_Sprn  -0.590206   0.237634  -2.484  0.01388 *
## DeporteTennis  -0.337134   0.257544  -1.309  0.19212
## DeporteW_Polo  -0.221257   0.223472  -0.990  0.32340
## Generomale     -1.104796   0.218563  -5.055 1.02e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6765 on 188 degrees of freedom
## Multiple R-squared:  0.9888, Adjusted R-squared:  0.9881
## F-statistic:  1280 on 13 and 188 DF,  p-value: < 2.2e-16
```

```
# Al final obtenemos los mismos resultados
```

Ejercicio 8

Usando la base de datos **Puentes**, ajusta el mejor modelo predictivo del coste de construcción *CCost*. ¿Qué variable de las disponibles tiene mayor capacidad predictiva?

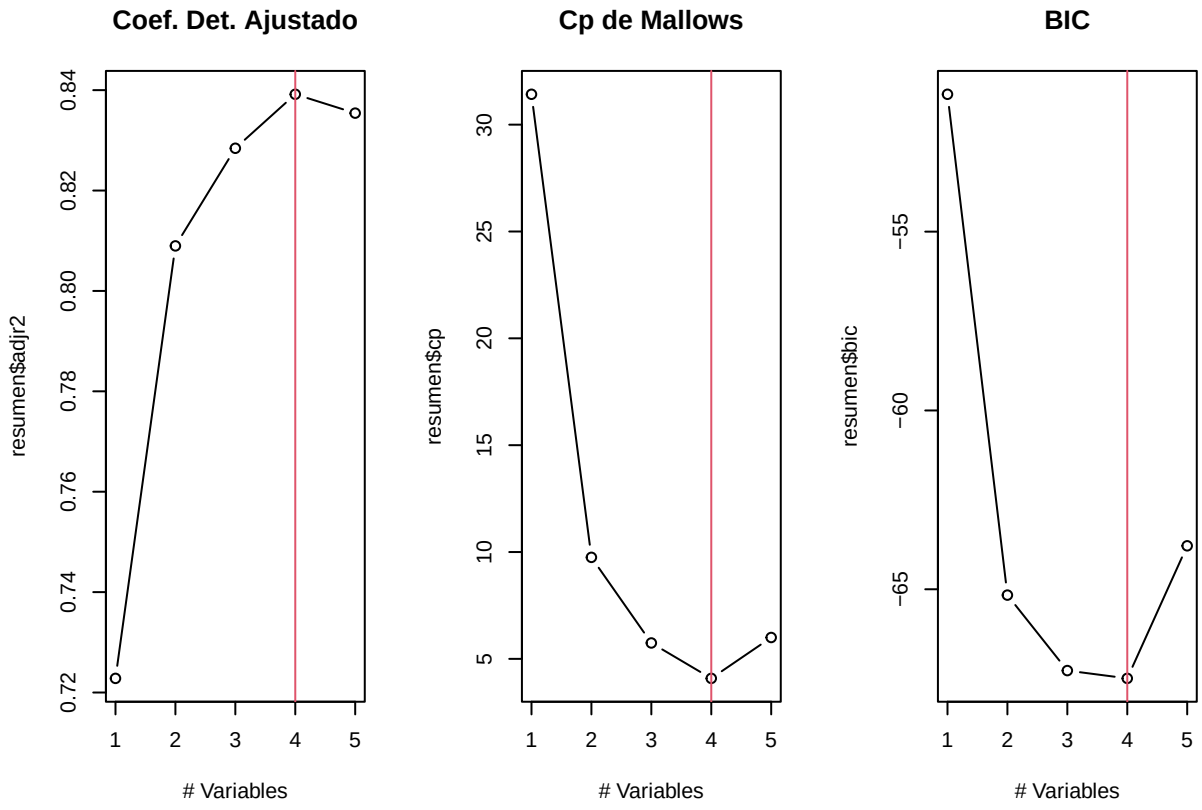
```
library(leaps)
sel_lm <- regsubsets(CCost ~. -Case , data=puentes,nvmax=8)
summary(sel_lm)
```

```
## Subset selection object
## Call: regsubsets.formula(CCost ~ . - Case, data = puentes, nvmax = 8)
## 5 Variables  (and intercept)
##        Forced in Forced out
## Time       FALSE      FALSE
## DArea      FALSE      FALSE
## Dwgs       FALSE      FALSE
## Length     FALSE      FALSE
## Spans      FALSE      FALSE
## 1 subsets of each size up to 5
## Selection Algorithm: exhaustive
##          Time DArea Dwgs Length Spans
## 1  ( 1 ) " "  "*"   " "  " "    " "
## 2  ( 1 ) " "  "*"   " "  "*"    " "
## 3  ( 1 ) " "  "*"   " "  "*"    "*"
## 4  ( 1 ) " "  "*"   "*"  "*"    "*"
## 5  ( 1 ) "*"  "*"   "*"  "*"    "*"
```

```
resumen<-summary(sel_lm)

resultado <- cbind(resumen$rsq,resumen$adjr2,resumen$cp,resumen$bic)
colnames(resultado) <- c('Rsq','RsqAdj','Cp','BIC')

par(mfrow = c(1,3))
plot(1:5, resumen$adjr2, xlab = "# Variables", main = "Coef. Det. Ajustado",
     type="b")
abline(v = which.max(resumen$adjr2), col = 2)
plot(1:5, resumen$cp, xlab = "# Variables", main = "Cp de Mallows",
     type='b')
abline(v = which.min(resumen$cp), col = 2)
plot(1:5, resumen$bic, xlab = "# Variables", main = "BIC",
     type = "b")
abline(v = which.min(resumen$bic), col = 2)
```




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

# Coinciden: son necesarias 4 variables.


colnames(resumen$which)[resumen$which[4,]==T] #Vemos las variables necesarias
```

```
## [1] "(Intercept)" "DArea"       "Dwgs"        "Length"      "Spans"
```

```
best<-lm(CCost ~DArea+Dwgs+Length+Spans  ,data=puentes,na.action=na.exclude )
summary(best)
```

```
##
## Call:
## lm(formula = CCost ~ DArea + Dwgs + Length + Spans, data = puentes,
##     na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -335.21  -55.18  -16.05   29.07  353.83
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -28.2249    55.2427  -0.511   0.6122
## DArea        15.6990     2.6236   5.984 4.98e-07 ***
## Dwgs         16.7306     8.6485   1.935   0.0601 .
## Length        0.8441     0.1870   4.515 5.48e-05 ***
```

```
## Spans       -51.9553    20.8934  -2.487   0.0172 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 122.4 on 40 degrees of freedom
## Multiple R-squared:  0.8538, Adjusted R-squared:  0.8392
## F-statistic:  58.4 on 4 and 40 DF,  p-value: 3.598e-16
```

```
# Este es un caso en el que puede ser interesante hacer 0 el intercepto, dado que si todas las variable

best_Sin<-lm(CCost ~DArea+Dwgs+Length+Spans-1  ,data=puentes,na.action=na.exclude )
summary(best_Sin)
```

```
##
## Call:
## lm(formula = CCost ~ DArea + Dwgs + Length + Spans - 1, data = puentes,
##     na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -322.49  -60.90  -17.96   20.25  346.36
##
## Coefficients:
##        Estimate Std. Error t value Pr(>|t|)
## DArea   16.0064     2.5306   6.325 1.49e-07 ***
## Dwgs    13.1192     4.9382   2.657  0.01119 *
## Length   0.8739     0.1760   4.965 1.26e-05 ***
## Spans  -55.2524    19.6920  -2.806  0.00764 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 121.3 on 41 degrees of freedom
## Multiple R-squared:  0.9268, Adjusted R-squared:  0.9197
## F-statistic: 129.8 on 4 and 41 DF,  p-value: < 2.2e-16
```

```
AIC(best,best_Sin);BIC(best,best_Sin);anova(best,best_Sin);
```

```
##          df      AIC
## best      6 567.0901
## best_Sin  5 565.3829
```

```
##          df      BIC
## best      6 577.9301
## best_Sin  5 574.4162
```

```
## Analysis of Variance Table
##
## Model 1: CCost ~ DArea + Dwgs + Length + Spans
## Model 2: CCost ~ DArea + Dwgs + Length + Spans - 1
##   Res.Df    RSS Df Sum of Sq     F Pr(>F)
## 1     40 599662
## 2     41 603576 -1   -3913.5 0.261 0.6122
```