Tarea – Regresión lineal

Tema 1 - Aprendizaje Máquina (I). Master Ciencia de Datos UV

Jesús Martínez Leal y Miguel Muñoz Blat (Grupo 11)

2023-12-18

Índice

Ejercicio 1.

Considera la variable respuesta `Price` relacionándola con la variable `X` con la que tenga mayor relación lineal.

1. Evalúa el efecto de `X` sobre `Price`.
2. Obtén la recta de mínimos cuadrados. Interpreta los resultados obtenidos (coeficientes, significatividad, R2, contraste del modelo, etc...).
3. Dibuja el diagrama de dispersión, la recta de regresión y las bandas de confianza al 90 %.
4. Realiza un diagnóstico de los residuos. Si falla algunas de las condiciones, busca una posible solución.

Carga inicial de los datos necesarios.

Lo primero de todo será ver un poco en qué consiste nuestro conjunto de datos y en qué unidades se miden las distintas variables.

```
     Type Price MPG.city MPG.highway EngineSize Horsepower  RPM Rev.per.mile Fuel.tank.capacity
1   Small  15.9       25          31        1.8        140 6300         2890               13.2
2 Midsize  33.9       18          25        3.2        200 5500         2335               18.0
3 Compact  29.1       20          26        2.8        172 5500         2280               16.9
4 Midsize  37.7       19          26        2.8        172 5500         2535               21.1
5 Midsize  30.0       22          30        3.5        208 5700         2545               21.1
6 Midsize  15.7       22          31        2.2        110 5200         2565               16.4
  Passengers Length Wheelbase Width Weight  Origin
1          5    177       102    68   2705 non-USA
2          5    195       115    71   3560 non-USA
3          5    180       102    67   3375 non-USA
4          6    193       106    70   3405 non-USA
5          4    186       109    69   3640 non-USA
6          6    189       105    69   2880     USA
```

Tras usar el comando `?Cars93` en la terminal de RStudio, observamos que tenemos numerosas variables con características de 93 coches en venta en Estados Unidos en 1993. Vemos que por ejemplo la variable `Price` viene medida en miles de dólares y que `Horsepower` viene medida en HP (lo que serían 746 W en el Sistema Internacional).

Apartado 1.

Evalúa el efecto de `X` sobre `Price`.

Resolución.

Para evaluar el efecto de una variable `X` de nuestro conjunto sobre `Price` haremos una correlación entre los pares de variables. Se utiliza el método de Pearson para su cálculo.

La expresión de la correlación viene dada por:

$$r_{X,Y} = \frac{s_{x,y}}{s_x s_y}, \tag{1}$$

siendo $s_{x,y}$ la covarianza entre las variables X e Y y s_x, s_y las desviaciones estándar de las variables X e Y, respectivamente.

```
  Price   MPG.city MPG.highway EngineSize Horsepower          RPM Rev.per.mile Fuel.tank.capacity
1     1 -0.5945622  -0.5606804  0.5974254  0.7882176 -0.004954931   -0.4263951            0.61948
  Passengers    Length Wheelbase     Width   Weight
1 0.05786007 0.5036284 0.5008642 0.4560279 0.647179
```

Como se puede apreciar, la mayor correlación en valor absoluto más cercana a la unidad viene dada por la variable `var_max_value` con un valor de 0.7882176

Apartado 2.

Obtén la recta de mínimos cuadrados. Interpreta los resultados obtenidos (coeficientes, significatividad, R2, contraste del modelo, etc...).

Resolución.

Nuestro modelo de regresión lineal se considera de la forma:

$$Y = \beta_0 + \beta_1 X + \epsilon, \tag{2}$$

siendo en este caso $Y = Price$ y $X = Horsepower$.

```
Call:
lm(formula = Price ~ Horsepower, data = cars)

Residuals:
    Min      1Q  Median      3Q     Max
-16.413  -2.792  -0.821   1.803  31.753

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -1.3988     1.8200  -0.769    0.444
Horsepower    0.1454     0.0119  12.218   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.977 on 91 degrees of freedom
Multiple R-squared:  0.6213,    Adjusted R-squared:  0.6171
F-statistic: 149.3 on 1 and 91 DF,  p-value: < 2.2e-16


                   5 %      95 %
(Intercept) -4.4232199 1.6256817
Horsepower   0.1255998 0.1651427
```

En los coeficientes podemos apreciar que el valor estimado del `intercept` (β_0) es -1.3987691 y el de la pendiente (β_1) es 0.1453712. A pesar de que el intercepto posee un valor negativo (un precio negativo carecería de sentido) se nos indica que no hay significancia, por lo que no se descarta que fuera nulo. En contra, la pendiente sí posee significancia, indicando que efectivamente existe una relación (lineal) entre `Price` y `Horsepower` y que además es positiva. Esto es lo que cabría esperar: una subida del precio del vehículo conforme aumentan los caballos de potencia. Se muestran también los intervalos de confianza al 90% para intercepto y pendiente. Esto permite reafirmar lo comentado anteriormente: el intercepto no se descarta que sea 0, mientras que la pendiente sí es claramente distinta de 0 y además positiva.

Por otra parte, el valor obtenido para el coeficiente de determinación R^2 ha sido de 0.621287. Así pues, 62.128695 % sería el porcentaje de variabilidad explicado por el modelo para la variable dependiente `Precio`. El valor ajustado del R^2 viene dado por la expresión:

$$R^2_{\text{ajustado}} = 1 - \frac{n-1}{n-k-1}(1-R^2) \tag{3}$$

Este tiene en cuenta la penalización incluida por aumentar el número de predictores. El valor obtenido es ligeramente menor: 0.6171253.

Finalmente, el p-valor asociado al estadístico F es también sufucientemente pequeño (< 2.2e-16) como para concluir que al menos un coeficiente del modelo lineal es distinto de cero. En este caso, dado que solo hay un predictor, este test sería equivalente al contraste sobre la pendiente de la recta (por eso se obtiene lo mismo).

Apartado 3.

Dibuja el diagrama de dispersión, la recta de regresión y las bandas de confianza al 90 %.

Resolución.

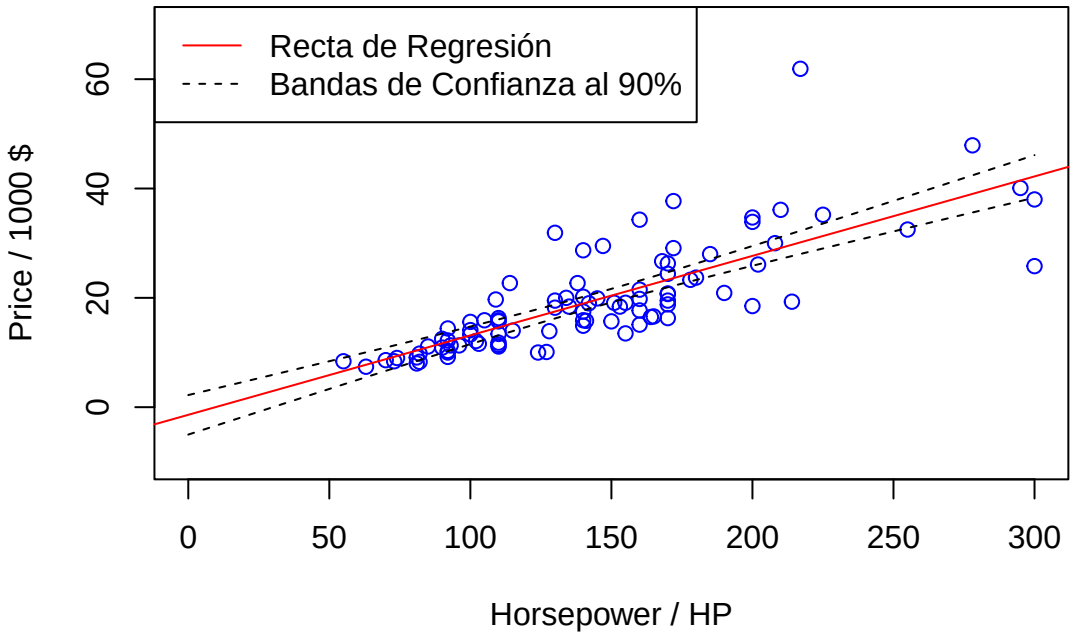

Figura 1. Gráfico de dispersión y recta de regresión para nuestro modelo

Observamos en la Figura anterior cómo la mayoría de puntos con los que contamos están agrupados en torno a 100 - 200 de Horsepower medido en HP. El modelo lineal realmente solo tiene relevancia en la zona de nuestros datos. Puede observarse cómo las bandas de confianza se acercan mucho más a la recta de regresión donde teníamos más valores agrupados. Es notable apreciar cómo hay un valor muy alejado de nuestra recta de regresión que seguramente está "tirando de ella" hacia arriba.

Apartado 4.

Realiza un diagnóstico de los residuos. Si falla algunas de las condiciones, busca una posible solución.

Resolución

Realizamos primero un diagrama de dispersión para nuestros residuos.

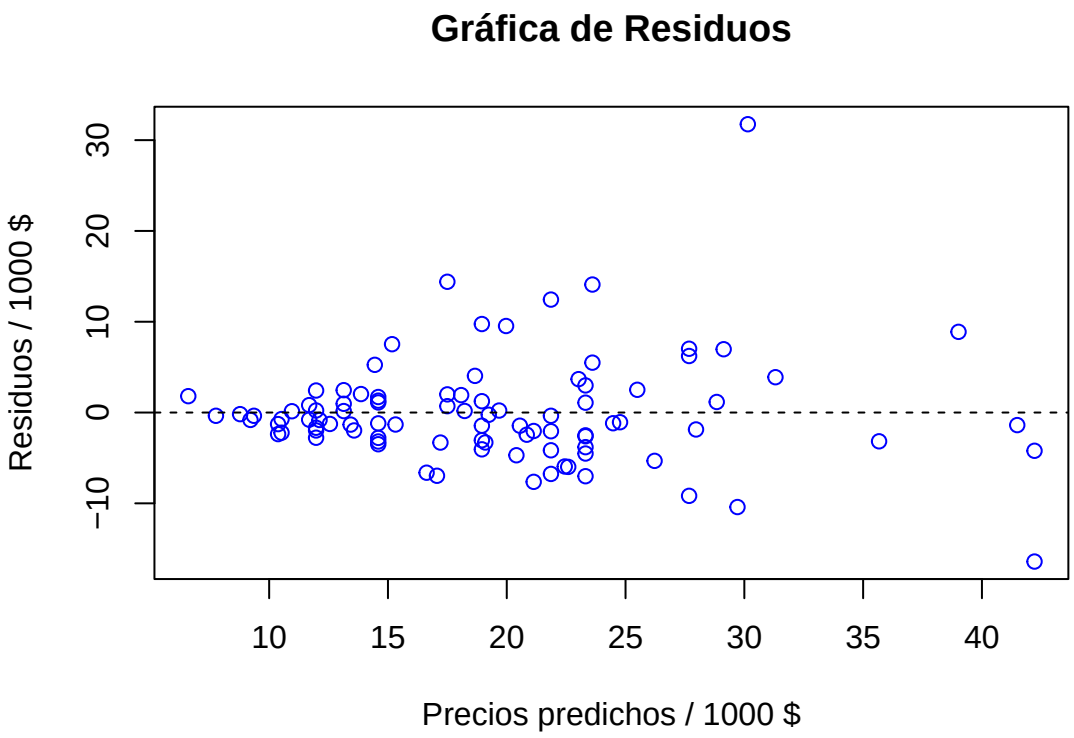

Figura 2. Gráfico de residuos en nuestro modelo

En el gráfico de residuos frente a valores predichos podemos ver que la variabilidad no es estrictamente constante. Para precios bajos encontramos una menor variabilidad que la hallada para precios intermedios y altos. Esto nos lleva a pensar que no tenemos una homocedasticidad clara. Para tratar de solucionar esto, podemos probar a realizar transformaciones sobre la variable dependiente (Precios). Por otra parte, la linealidad sí que parece adecuada: los residuos se distribuyen alrededor del 0.

En cuanto a la normalidad de los residuos encontramos esto:

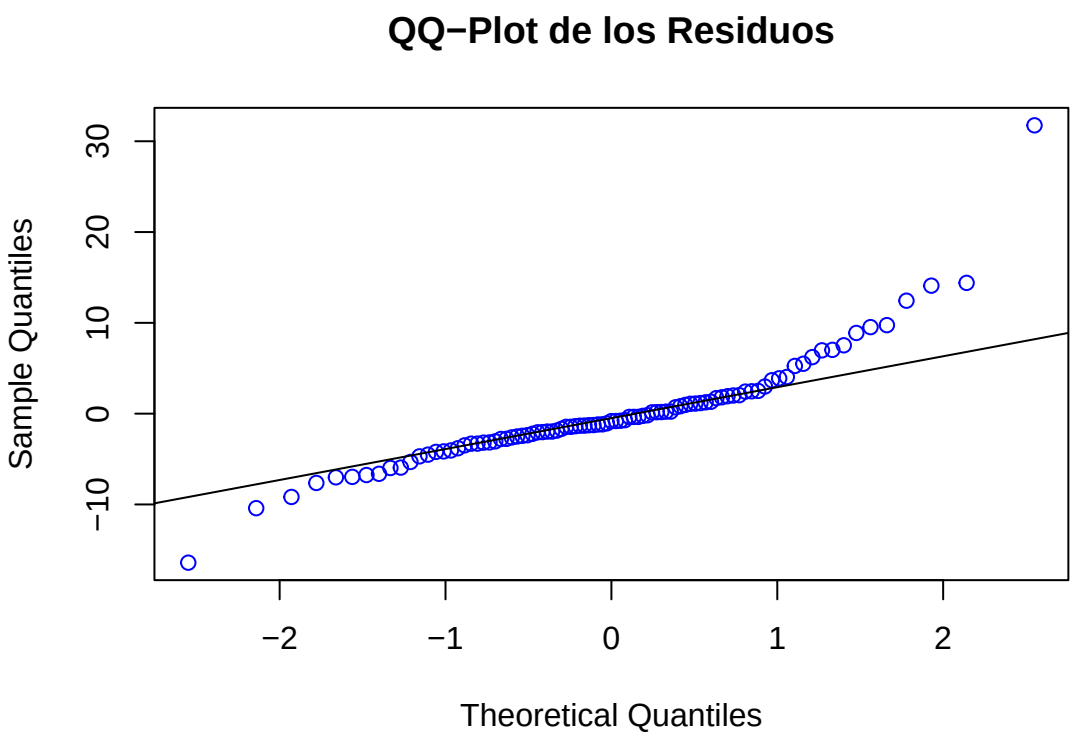

Figura 3. QQ-Plot de los residuos en nuestro modelo

```
    Shapiro-Wilk normality test

data:  reg$residuals
W = 0.859, p-value = 6.113e-08
```

Vemos cómo la distribución de residuos sigue un comportamiento normal pero difiere en gran medida en las colas. Utilizando el test de normalidad de Shapiro-Wilk obtenemos un p-valor de 6.1130946×10^{-8}, por lo que se rechaza la hipótesis nula de que la distribución sea normal para un nivel de significancia de 0.05.

Veremos ahora cómo cambian las cosas en nuestro modelo si aplicamos una transformación logarítmica.

Transformación logarítmica de la variable `Precio`

Realizamos los mismos pasos anteriores pero modificando `Precio` por `Log(Precio)` como variable dependiente.

```
Call:
lm(formula = log(Price) ~ Horsepower, data = cars)

Residuals:
     Min       1Q   Median       3Q      Max
-0.73316 -0.16831 -0.01999  0.14305  0.73621
```

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 1.8357418  0.0787413   23.31   <2e-16 ***
Horsepower  0.0071593  0.0005147   13.91   <2e-16 ***
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

Residual standard error: 0.2586 on 91 degrees of freedom
Multiple R-squared:  0.6801,    Adjusted R-squared:  0.6766
F-statistic: 193.4 on 1 and 91 DF,  p-value: < 2.2e-16


                    5 %        95 %
(Intercept) 1.704891775 1.966591865
Horsepower  0.006303923 0.008014708
```

Con el modelo logarítmico apreciamos que el valor de R^2 ha subido algo, teniendo 0.6800774, lo que indicaría en este caso una ligera mejoría de lo que teníamos anteriormente. Esto se debe principalmente a que el “outlier” que se ha comentado anteriormente ha sido reducido en importancia con esta transformación.

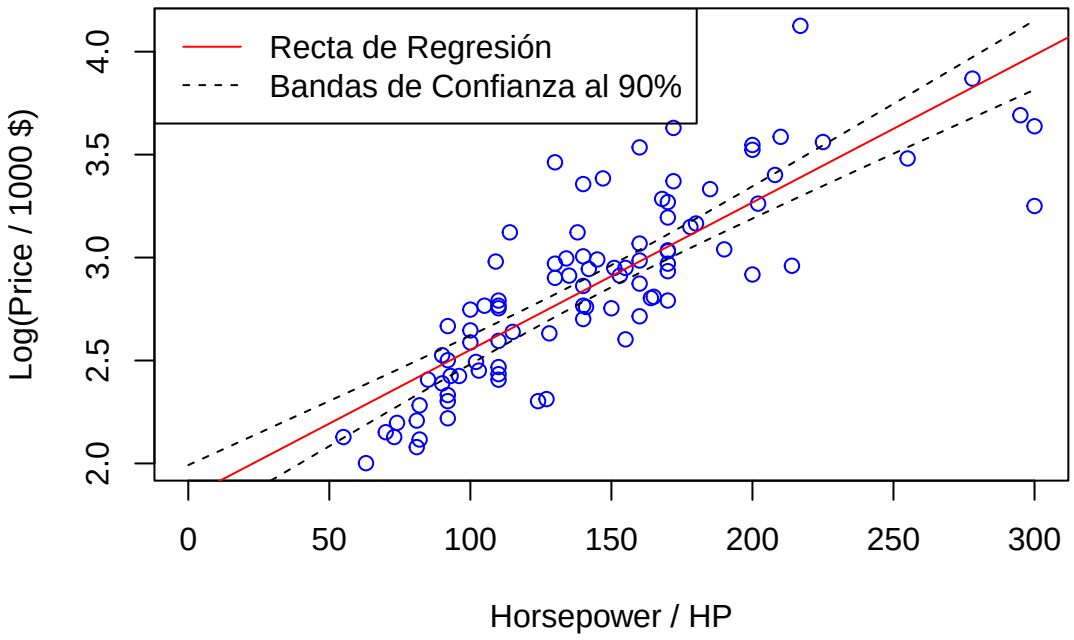

Figura 4. Gráfico de dispersión y recta de regresión para nuestro modelo tras la transformación logarítmica

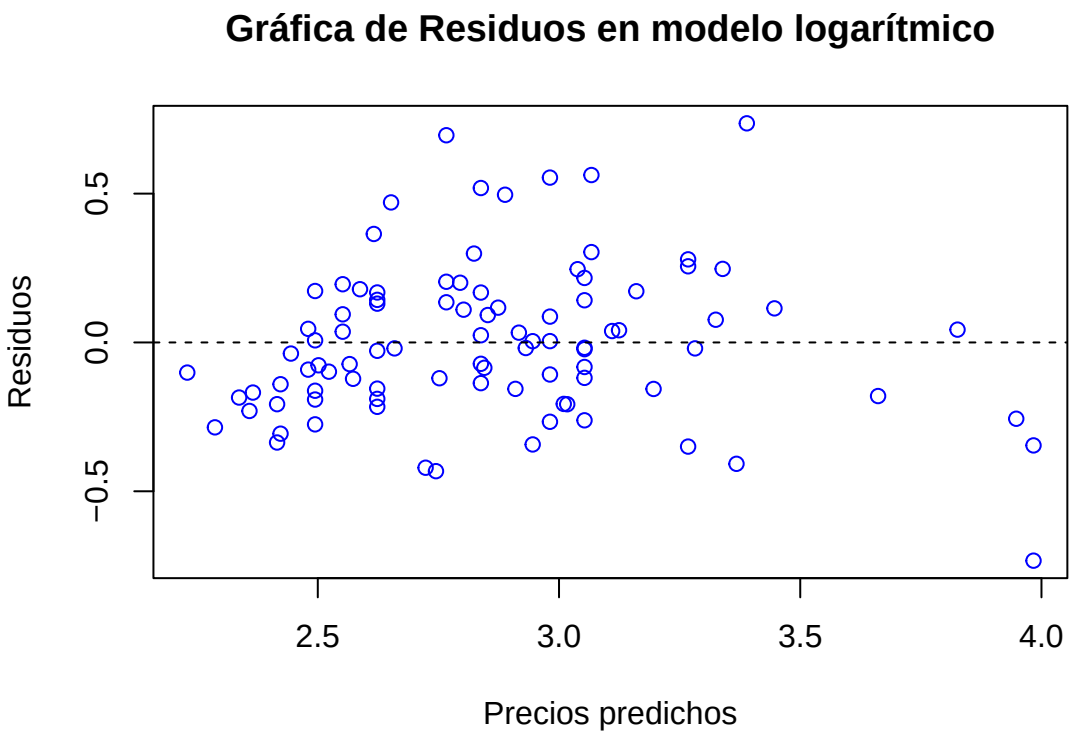

Figura 5. Gráfico de residuos en nuestro modelo tras la transformación logarítmica

Puede observarse cómo con el modelo logarítmico se ha podido disminuir ligeramente la heterocedasticidad, en el sentido de que no están tan agrupados los puntos iniciales entre sí.

```
    studentized Breusch-Pagan test

data:  logreg
BP = 6.5254, df = 1, p-value = 0.01063
```

Aún así, el p-valor asociado vemos que no es suficientemente alto, por lo que implicaría tener heterocedasticidad según este test.

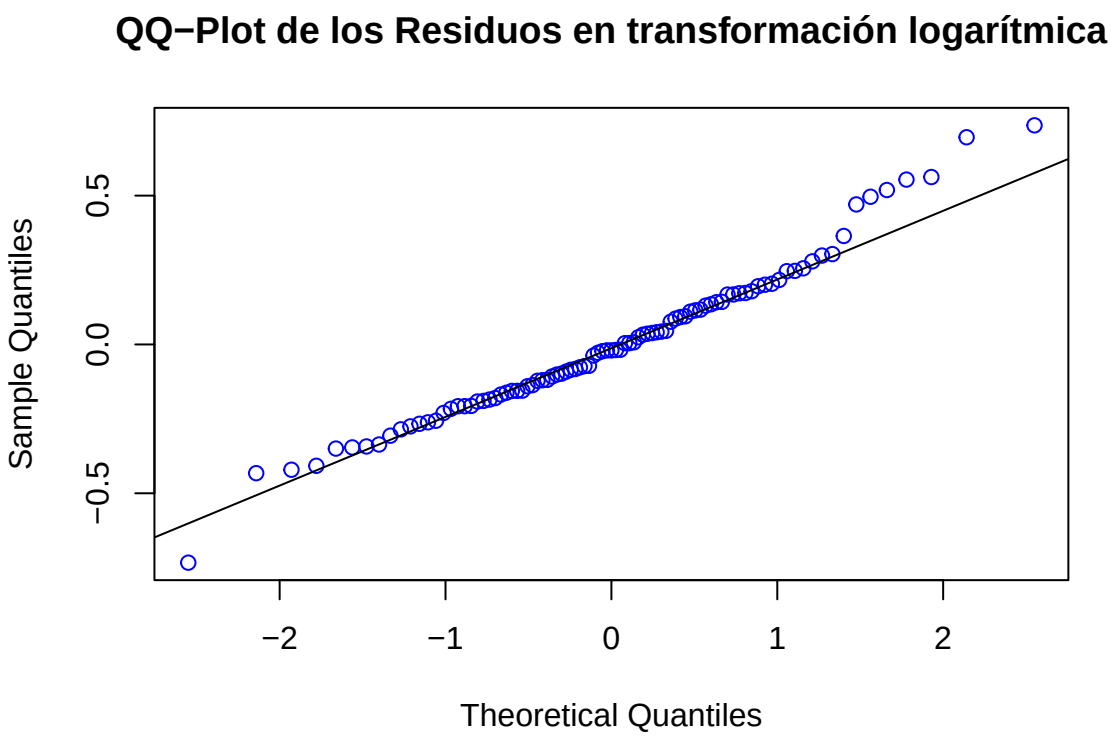

Figura 6. QQ-Plot de los residuos en nuestro modelo con transformación logarítmica

```
    Shapiro-Wilk normality test

data:  logreg$residuals
W = 0.97667, p-value = 0.0942
```

En este caso obtenemos que nuestra distribución de residuos se adapta más a una distribución normal, viéndose además reflejado esto en el shapiro test donde se obtiene un p-valor de $0.0941967 > 0.05$. Así pues, no puede rechazarse la hipótesis nula de normalidad.

Ejercicio 2.

Considera la variable respuesta Price relacionandola con el predictor MPG.city.

1. Evalúa el efecto de MPG.city sobre Price.
2. Obtén la recta mínimos cuadrados. Interpreta los resultados obtenidos (coeficientes, significatividad, R2, contraste del modelo, etc...).
3. Dibuja el diagrama de dispersión, la recta de regresión y las bandas de predicción al 90 %.
4. Realiza un análisis de los residuos.
5. ¿Te parece adecuado haber realizado regresión lineal o es preferible otro tipo de regresión?. Ajusta el modelo que te parezca más adecuado.

6. ¿Qué precio mínimo se espera para aquellos coches con un consumo de 12 litros a los 100 km por ciudad? Calcula e interpreta el intervalo de confianza y el de predicción.

Apartado 1.

Evalúa el efecto de MPG.city sobre Price.

Resolución.

En primer lugar, de ojear en `?Cars93` vemos que `MPG.city` es básicamente `Millas por galón estadounidense de ciudad`. Es una de las métricas usadas para evaluar la economía del combustible de un vehículo. Una vez sabido esto ya tenemos un poco más de contexto acerca de lo que estamos haciendo.

La correlación entre `Price` y `MPG.city` se muestra a continuación.

```
[1] -0.5945622
```

MPG.city y Price presentan una correlación lineal de -0.5945622. Su valor absoluto está algo alejado de la unidad, por lo que la relación no es del todo lineal. Por otra parte, el signo negativo indica que, en general, un aumento de MPG.city conllevaría una disminución del Precio.

Apartado 2.

Obtén la recta mínimos cuadrados. Interpreta los resultados obtenidos (coeficientes, significatividad, R2, contraste del modelo, etc...).

Resolución.

Nuestro modelo de regresión lineal se considera de la forma:

$$Y = \beta_0 + \beta_1 X + \epsilon, \tag{4}$$

siendo en este caso $Y = Price$ y $X = MPG.city$.

```
Call:
lm(formula = Price ~ MPG.city, data = cars)

Residuals:
    Min      1Q  Median      3Q     Max
-10.437  -4.871  -2.152   1.961  38.951

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  42.3661     3.3399  12.685  < 2e-16 ***
MPG.city     -1.0219     0.1449  -7.054 3.31e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.809 on 91 degrees of freedom
Multiple R-squared:  0.3535,    Adjusted R-squared:  0.3464
F-statistic: 49.76 on 1 and 91 DF,  p-value: 3.308e-10
```

```
                5 %       95 %
(Intercept) 36.815972 47.9161357
MPG.city    -1.262692 -0.7811955
```

En los coeficientes podemos apreciar que el valor estimado del `intercept` (β_0) es 42.3660541 y el de la pendiente (β_1) es -1.0219438. El intercepto posee un valor claramente distinto de 0 y además cuenta con significancia. El valor de la pendiente por su parte es negativo y cuenta con significancia. Se muestran también los intervalos de confianza al 90% para intercepto y pendiente. Esto permite reafirmar lo comentado anteriormente: el intercepto se descarta que sea nulo, mientras que la pendiente sí es claramente distinta de 0 y además negativa.

Por otra parte, el valor obtenido para el coeficiente de determinación R^2 ha sido de 0.3535042. Así pues, 35.3504166 % sería el porcentaje de variabilidad explicado por el modelo para la variable dependiente `Price`. Este valor es ciertamente bajo y está algo lejos de lo que sería lo usualmente aceptable.

El valor ajustado del R^2 viene dado por la expresión:

$$R^2_{\text{ajustado}} = 1 - \frac{n-1}{n-k-1}(1-R^2) \tag{5}$$

Este tiene en cuenta la penalización incluida por aumentar el número de predictores. El valor obtenido es ligeramente menor: 0.3463998.

Finalmente, el p-valor asociado al estadístico F es también sufucientemente pequeño ($<$ 3.31e-10) como para concluir que al menos un coeficiente del modelo lineal es distinto de cero. En este caso, dado que solo hay un predictor, este test sería equivalente al contraste sobre la pendiente de la recta (por eso se obtiene lo mismo).

Apartado 3.

Dibuja el diagrama de dispersión, la recta de regresión y las bandas de predicción al 90 %.

Resolución.

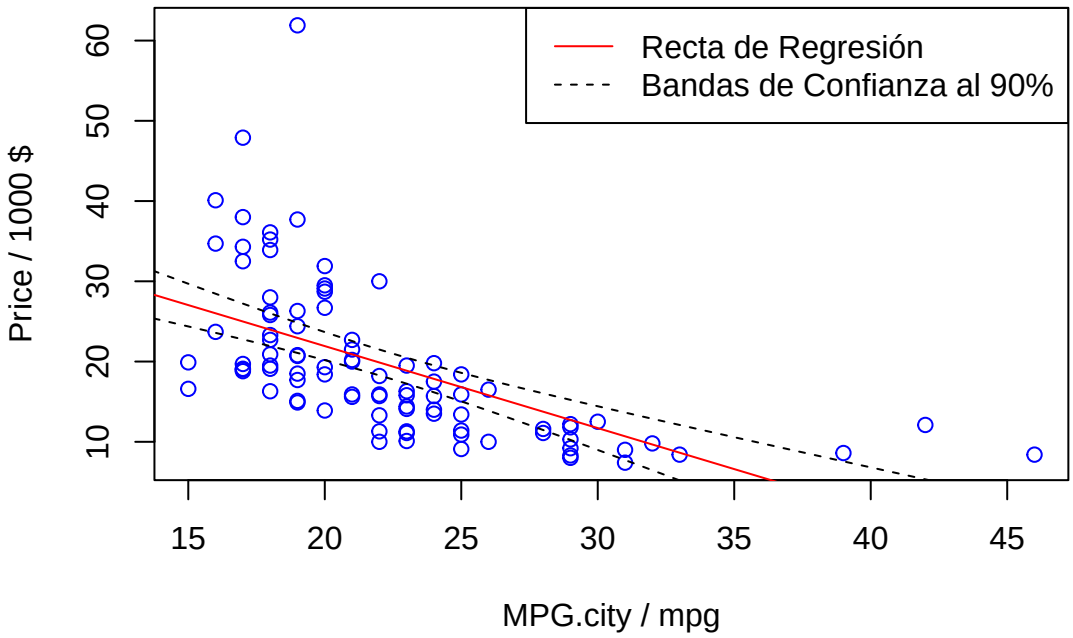

Figura 7. Gráfico de dispersión y recta de regresión para nuestro modelo

Se hace muy evidente con la representación que la variable `MPG.city` tiene un aspecto algo cuantizado, ya que no posee demasiados valores distintos. Esto hace que para un mismo valor de esta, haya muchos valores de `Price` diferentes.

Apartado 4.

Realiza un análisis de los residuos.

Resolución.

Realizamos primero un diagrama de dispersión para nuestros residuos.

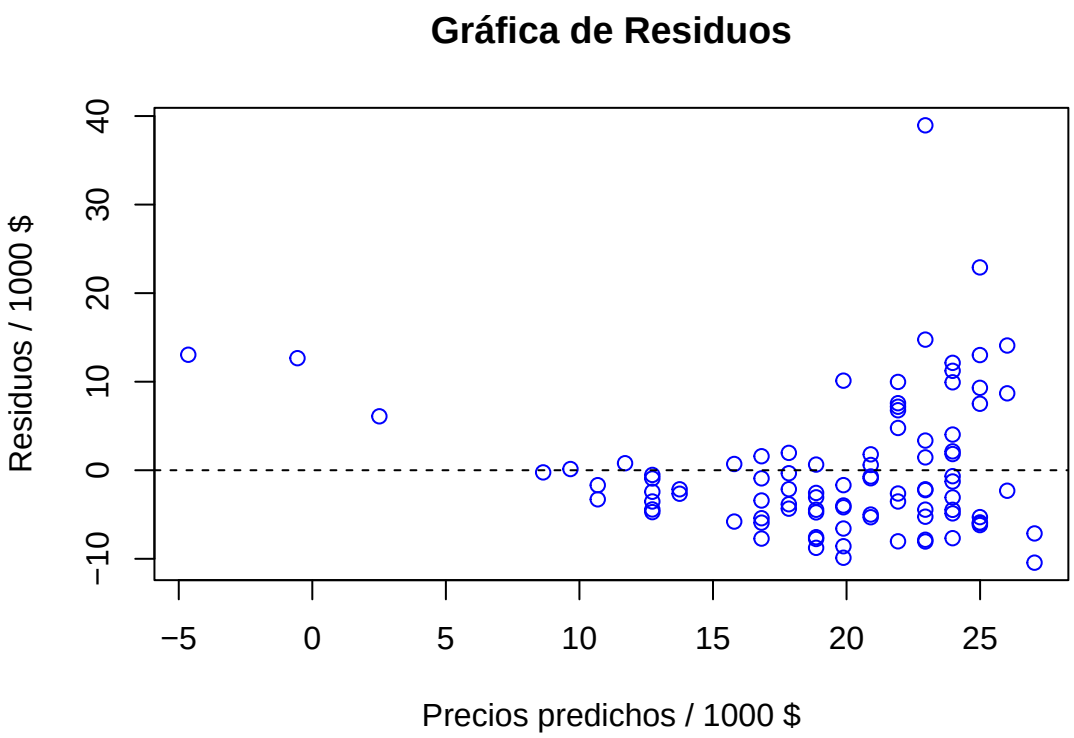

Figura 8. Gráfico de residuos en nuestro modelo

En la gráfica de residuos frente a valores predichos podemos ver que la relación entre ambas no es del todo lineal, tal y como podía verse tanto en la gráfica de dispersión anterior como en el valor absoluto de la correlación de Pearson. Vemos además que la variablidad no es constante por lo que también falla la homocedasticidad. En cuanto a la normalidad de los residuos encontramos esto:

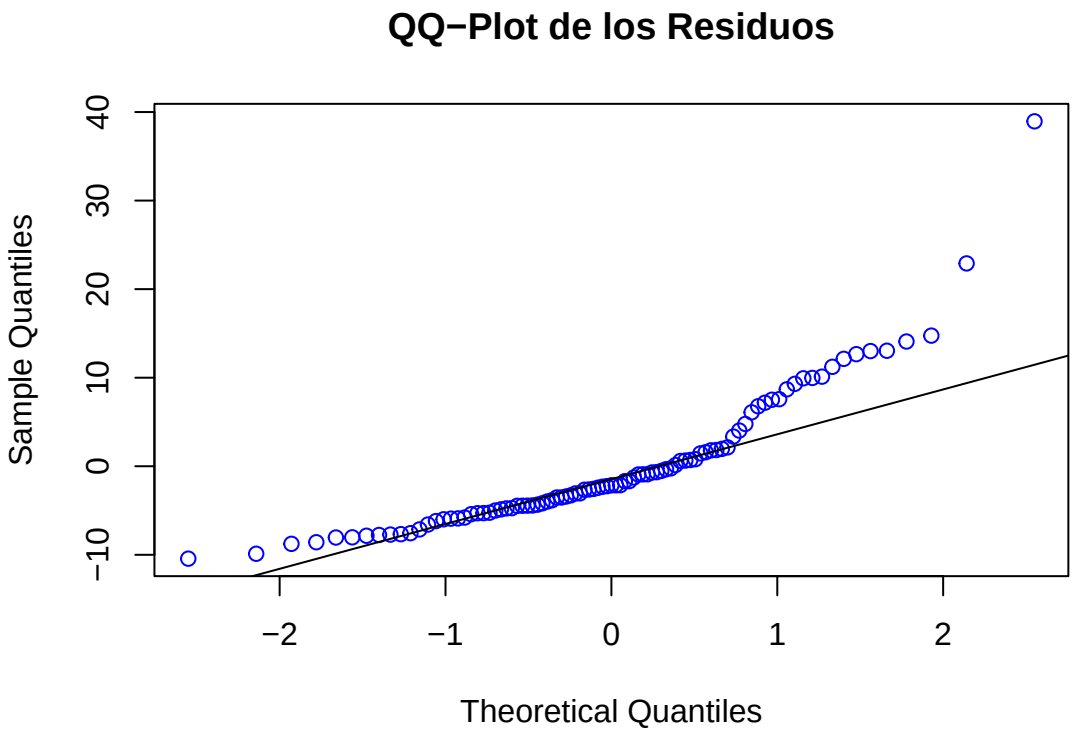

Figura 9. QQ-Plot de los residuos en nuestro modelo

```
    Shapiro-Wilk normality test

data:  reg$residuals
W = 0.84144, p-value = 1.434e-08
```

Por otro lado, en la gráfica de los cuantiles de los residuos frente a los cuantiles teóricos de una distribución normal podemos ver cómo en los extremos los cuantiles se desvían considerablemente, por lo que tampoco parece que se cumpla la normalidad de los residuos. Esto puede verse también con un contraste de hipótesis que nos brinda el test de normalidad de Shapiro-Wilk, en el que se obtiene un p-valor de 1.4337855×10^{-8}, por lo que se rechaza la hipótesis nula de que la distribución sea normal para un nivel de significancia de 0.05.

En definitiva, no se cumplen las hipótesis de partida del modelo lineal, por lo que no sería lo más apropiado para describir la relación entre ambas variables. En todo caso, podría plantearse la transformación de alguna de las variables con tal de obtener mayor linealidad, homocedasticidad y normalidad. También podría plantearse otro modelo de regresión no lineal como el **modelo potencial**, que parece más adecuado para describir la relación entre estas varaibles.

Apartado 5.

¿Te parece adecuado haber realizado regresión lineal o es preferible otro tipo de regresión? Ajusta el modelo que te parezca más adecuado.

Resolución.

Se probarán distintos modelos para ver con cuál se obtienen resultados más óptimos.

Paso a un modelo potencial.

El modelo potencial que proponemos es de la forma:

$$Y = \beta_0 X^{\beta_1} \tag{6}$$

Este puede expresarse como un modelo lineal tomando logaritmos:

$$\log(Y) = \log(\beta_0) + \beta_1 \log(X) \tag{7}$$

Pasan a realizarse los pasos seguidos anteriormente aplicando esta última transformación.

```
[1] -0.7444919
```

Vemos ya que la correlación ha aumentado (en valor absoluto) entre estas dos variables, lo que es una buena señal.

```
Call:
lm(formula = log(Price) ~ log_MPG.city, data = cars)

Residuals:
     Min       1Q   Median       3Q      Max
-0.61991 -0.21337 -0.03462  0.19766  1.05362

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)    7.5237     0.4390   17.14   <2e-16 ***
log_MPG.city  -1.5119     0.1421  -10.64   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3052 on 91 degrees of freedom
Multiple R-squared:  0.5543,    Adjusted R-squared:  0.5494
F-statistic: 113.2 on 1 and 91 DF,  p-value: < 2.2e-16


                   5 %      95 %
(Intercept)   6.794114  8.253305
log_MPG.city -1.748126 -1.275747
```

Observamos ya una subida en el valor de $R^2 = $0.5542682, indicando que este modelo puede ajustarse de una mejor manera a nuestros datos.

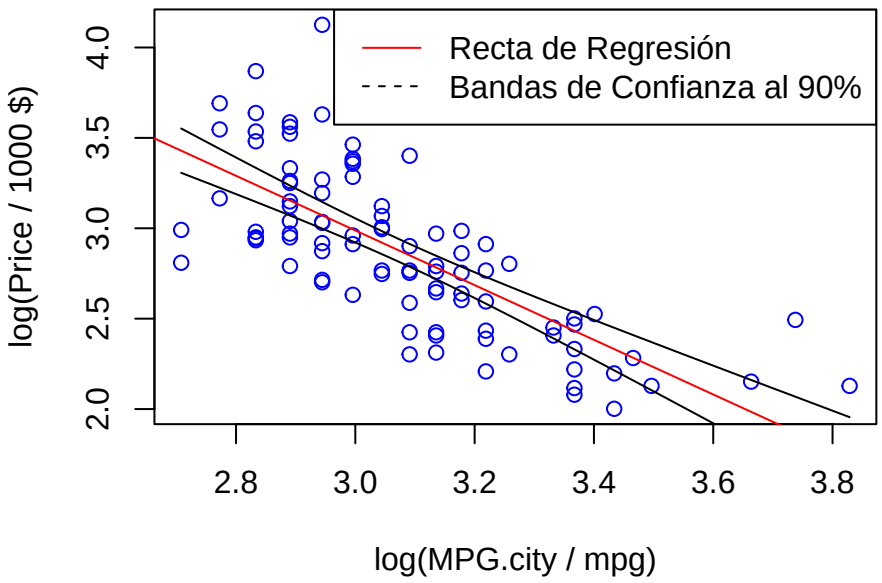

Figura 10. Gráfico de dispersión y recta de regresión para nuestro modelo potencial

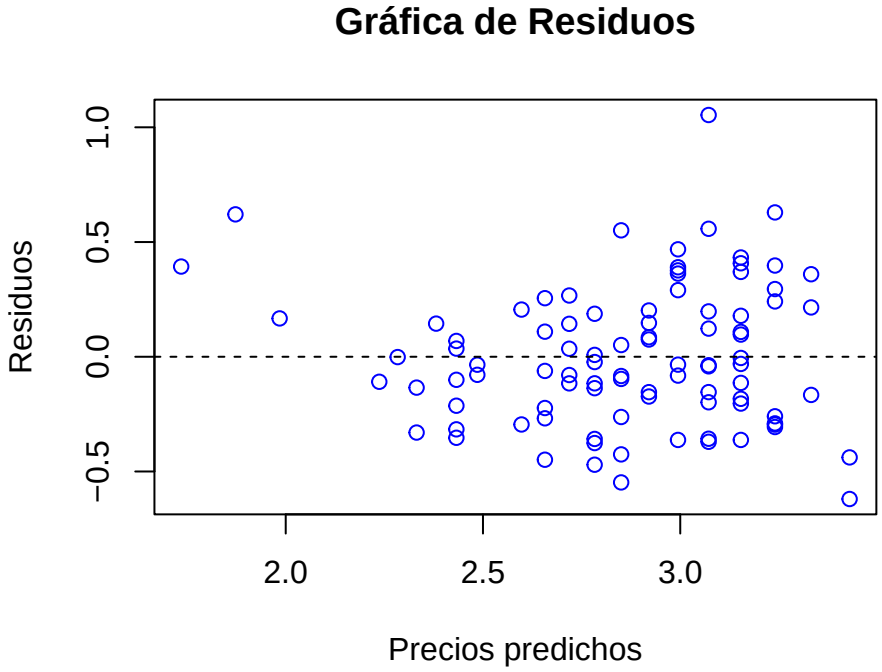

Figura 11. Gráfico de residuos en nuestro nuevo modelo potencial

Vemos cómo el problema de la heterocedasticidad se ha ido en gran medida, al igual que el de la linealidad.

En cuanto a la normalidad de los residuos podemos aplicar el contraste que nos ofrece el shapiro test nuevamente, obteniendo:

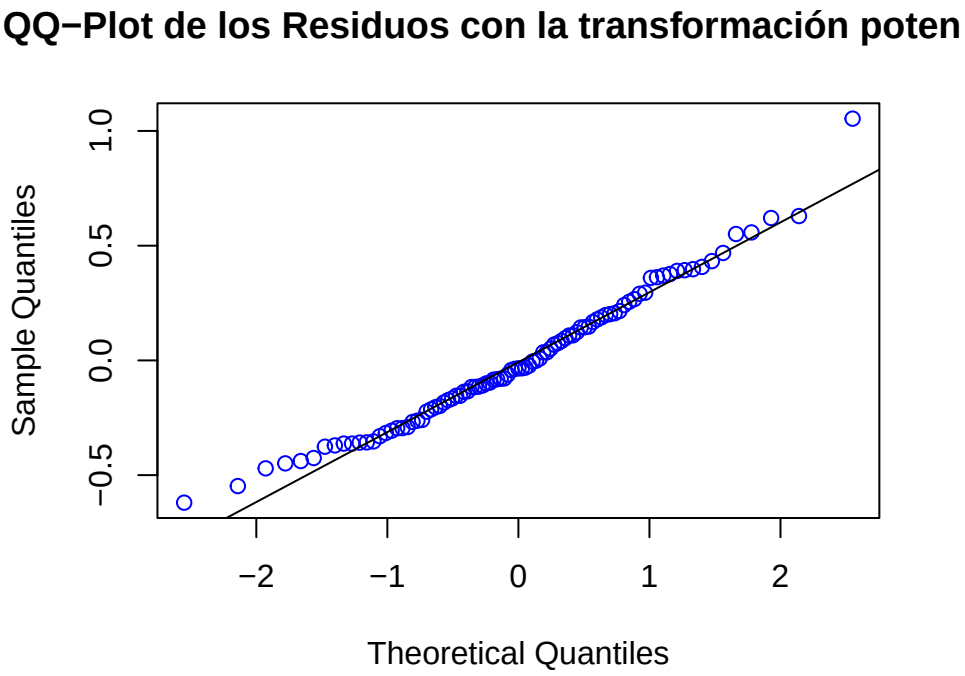

Figura 12. QQ-Plot de los residuos en nuestro modelo con transformación potencial

```
    Shapiro-Wilk normality test

data:  logreg$residuals
W = 0.97787, p-value = 0.1154
```

Ahora sí vemos que los residuos parecen mucho más normales, obteniendo un p-valor de 0.1153595: no se puede rechazar la hipótesis nula de normalidad.

Apartado 6.

¿Qué precio mínimo se espera para aquellos coches con un consumo de 12 litros a los 100 km por ciudad? Calcula e interpreta el intervalo de confianza y el de predicción.

Resolución

Para hacer este cálculo debemos primero conocer la conversión a las unidades que estamos tratando con nuestros datos. Una búsqueda en Convertidor nos muestra que:

$$1 \text{ km / L} = 2.3521 \text{ mpg} \tag{8}$$

Así pues, realizando una operación sencilla con tenemos:

$$\frac{100}{12} \text{ km / L} \times 2.3521 \text{ mpg} = 19.60\text{mpg} \tag{9}$$

Hay que tener algo de cuidado, ya que tenemos un modelo de la forma log-log. Así pues, además de introducir en la variable X que tenemos log(19.60), obtendremos valores para `log(Price)`. Es por eso que aplicamos la función inversa a la transformación implementada: en este caso la exponencial.

```
Para un MPG.city de 19.6 mpg, la estimación puntual de Price (en miles de dólares) es
 20.592 con un intervalo de confianza: ( 19.208 , 22.075 ).
```

```
Para un MPG.city de 19.6 mpg, la estimación puntual de Price (en miles de dólares) es
 20.592 con un intervalo de predicción: ( 11.186 , 37.908 ).
```

Puede observarse fácilmente cómo el intervalo de predicción es mucho más ancho que el de confianza. Esto se debe a que en el intervalo de predicción se considera tanto la incertidumbre en la estimación del parámetro de la población como la variabilidad de los puntos de datos individuales alrededor de la línea de regresión.

Para responder a la pregunta que se nos hace conviene utilizar el que ofrece la predicción. El precio mínimo esperado será de 11.186 miles de dólares.

Ejercicio 3.

1. Considerando un tope de 10 variables, encuentra el número óptimo de variables a incluir en un modelo predictivo de Price, según los criterios R2, BIC y AIC.

- ¿Qué variables incluye el modelo obtenido? (seleccionar el criterio que más te guste). Interpreta los coeficientes obtenidos, ¿consideras que tienen sentido?.

2. Selecciona el mejor modelo con el método stepwise.
3. Selecciona el mejor modelo con el método stepwise considerando la variable Passengers como factor. Contesta a las siguientes preguntas:

- ¿Qué % de la varianza de Price explica el modelo?
- ¿Podrías depurar el modelo?
- ¿Cuál es el efecto de la variable Origin sobre Price?

4. ¿Qué modelo de los apartados anteriores es mejor? Con el que te quedes, realiza el diagnóstico de tu modelo, sin emprender ninguna acción, e indica los problemas que presenta.
5. Emprende ahora las acciones que te parezcan oportunas e indica los problemas que has conseguido solucionar o mejorar un poco.
6. Obtén la predicción del precio para un coche en la mediana de los predictores en el modelo escogido. Notar que las variables categóricas se tratan de diferente manera, no hay mediana.

Apartado 1.

Considerando un tope de 10 variables, encuentra el número óptimo de variables a incluir en un modelo predictivo de Price, según los criterios R2, BIC y AIC. - ¿Qué variables incluye el modelo obtenido? (seleccionar el criterio que más te guste). Interpreta los coeficientes obtenidos, ¿consideras que tienen sentido?.

Resolución.

La función `regsubsets()` del paquete `leaps` nos permitirá realizar regresiones por subconjuntos, lo que implica la posibilidad de poder ajustar modelos lineales con todas las combinaciones posibles de un conjunto dado de variables predictoras. Podremos con esta además identificar los valores que toman los modelos según los distintos criterios propuestos (R^2, BIC, AIC (llamado aquí CP)).

A continuación se muestra lo que devuelve esta función al aplicarse a nuestros datos. Obtendremos primeramente una especie de tabla en el que se muestran los mejores modelos (`nbest = 1`) según el número de variables inmiscuidas (de 1 hasta `nvmax`), junto con las variables que este emplea. Se ha podido comprobar cómo se obtienen mejores resultados si no se tienen en cuenta en los cálculos posibles dependencias cuadráticas en las variables predictoras.

```
          TypeLarge TypeMidsize TypeSmall TypeSporty TypeVan MPG.city MPG.highway EngineSize
1  ( 1 )  " "       " "         " "       " "        " "     " "      " "         " "
2  ( 1 )  " "       "*"         " "       " "        " "     " "      " "         " "
3  ( 1 )  " "       " "         " "       " "        " "     " "      " "         " "
4  ( 1 )  " "       "*"         " "       " "        " "     " "      " "         " "
5  ( 1 )  " "       "*"         " "       " "        " "     " "      " "         " "
6  ( 1 )  " "       "*"         " "       " "        " "     " "      " "         " "
7  ( 1 )  " "       "*"         " "       "*"        " "     " "      " "         " "
8  ( 1 )  " "       "*"         " "       "*"        "*"     " "      " "         " "
9  ( 1 )  " "       "*"         " "       "*"        "*"     " "      "*"         " "
10  ( 1 ) " "       "*"         " "       "*"        "*"     " "      "*"         " "
          Horsepower RPM Rev.per.mile Fuel.tank.capacity Passengers Length Wheelbase Width Weight
1  ( 1 )  "*"         " " " "          " "                " "        " "    " "       " "   " "
2  ( 1 )  "*"         " " " "          " "                " "        " "    " "       " "   " "
3  ( 1 )  "*"         " " " "          " "                " "        " "    "*"       "*"   " "
4  ( 1 )  "*"         " " " "          " "                " "        " "    "*"       "*"   " "
5  ( 1 )  "*"         "*" " "          " "                " "        " "    "*"       "*"   " "
6  ( 1 )  "*"         "*" " "          " "                " "        " "    "*"       "*"   " "
7  ( 1 )  "*"         "*" " "          " "                " "        " "    "*"       "*"   " "
8  ( 1 )  "*"         "*" " "          " "                " "        " "    "*"       "*"   " "
9  ( 1 )  "*"         "*" " "          " "                " "        " "    "*"       "*"   " "
10  ( 1 ) "*"         "*" "*"          " "                " "        " "    "*"       "*"   " "
          OriginUSA
1  ( 1 )  " "
2  ( 1 )  " "
3  ( 1 )  " "
4  ( 1 )  " "
5  ( 1 )  " "
6  ( 1 )  "*"
7  ( 1 )  "*"
8  ( 1 )  "*"
9  ( 1 )  "*"
10  ( 1 ) "*"

            Rsq    RsqAdj        Cp       BIC
```

```
 [1,] 0.6212870 0.6171253 34.698727 -81.23561
 [2,] 0.6656906 0.6582615 22.195204 -88.30122
 [3,] 0.7006332 0.6905422 12.781933 -94.03557
 [4,] 0.7212477 0.7085771  8.048629 -96.13815
 [5,] 0.7367518 0.7216226  4.984548 -96.92759
 [6,] 0.7515821 0.7342506  2.140541 -97.78758
 [7,] 0.7572900 0.7373021  2.276161 -95.41679
 [8,] 0.7620670 0.7394067  2.715864 -92.73285
 [9,] 0.7649217 0.7394313  3.783436 -89.32280
[10,] 0.7677334 0.7394082  4.865050 -85.90926
```

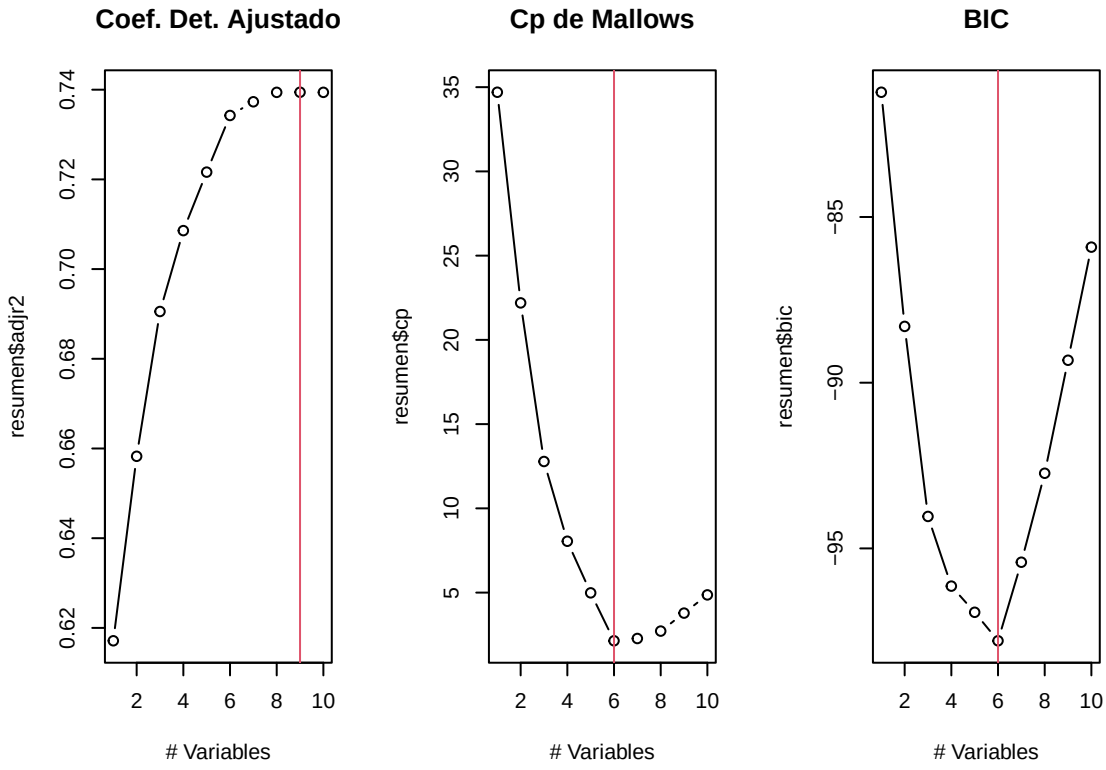

Figura 13. Valores obtenidos para distintos criterios en el mejor modelo según el número de variables

A la vista de lo obtenido hemos visto adecuado seleccionar el mejor modelo en función de los criterios `CP` o `BIC`, ya que nos permitan reducir algo la complejidad (tendríamos 6 variables en lugar de 10) respecto a si lo hiciéramos con el de `RsqAdj`.

```
Call:
lm(formula = formula, data = cars)

Residuals:
     Min       1Q   Median       3Q      Max
-11.2403  -2.6302  -0.1117   2.0195  22.6970

Coefficients:
```

```
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 46.649677  27.321819   1.707 0.091529 .
TypeLarge    1.299684   2.914914   0.446 0.656864
TypeMidsize  3.961813   1.914233   2.070 0.041632 *
TypeSmall    0.334407   2.088371   0.160 0.873174
TypeSporty   3.015877   2.299341   1.312 0.193307
TypeVan     -1.849234   2.874896  -0.643 0.521866
Horsepower   0.160788   0.017971   8.947 9.17e-14 ***
RPM         -0.003695   0.001373  -2.691 0.008630 **
Wheelbase    0.663732   0.212168   3.128 0.002434 **
Width       -1.434571   0.388931  -3.688 0.000404 ***
OriginUSA   -3.217185   1.296097  -2.482 0.015098 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.984 on 82 degrees of freedom
Multiple R-squared:  0.7627,    Adjusted R-squared:  0.7338
F-statistic: 26.36 on 10 and 82 DF,  p-value: < 2.2e-16
```

Las variables que incluye el mejor modelo seleccionado son: {Type, Horsepower, RPM, Wheelbase, Width, Origin} (resultando significativos sus coeficientes para el modelo, como bien se muestra arriba). Nótese que las variables `Type` y `Origin` han sido consideradas de tipo factor, por lo que, a pesar de que en el mejor modelo obtenido con Cp se seleccionaran solo ciertas categorías, en el modelo seleccionado final debe estar la columna pertinente completa.

Los resultados obtenidos resultan lógicos si se piensa en el contexto.

- Un incremento en potencia (`Horsepower`) conllevaría un mayor `Price` en líneas generales.
- Un incremento en la distancia entre centros de ruedas delanteras y traseras (`Wheelbase`) suele indicar que el coche sea algo más grande, lo que puede llevar a un mayor precio.

En cuanto al resto, resulta muy complicado deducir de manera lógica el por qué el coeficiente es positivo o negativo. Se necesitarían explorar otros factores importantes como pudiera ser la demanda de dicho año (1993) en USA de vehículos. Además, una muestra de 93 vehículos no consideramos que sea del todo suficiente (al menos con la percepción actual de la gran cantidad de vehículos que existen).

Una cosa que sí llama la atención es que el precio disminuye con la variable `RPM`, ya que mayores Revoluciones Por Minuto proporcionan mayor potencia y uno esperaría un mayor precio. Por ello, podemos estudiar si RPM es realmente significativa realizando un test F parcial, cuya hipótesis nula es que el coeficiente RPM es cero.

```
Analysis of Variance Table

Model 1: Price ~ Type + Horsepower + Wheelbase + Width + Origin
Model 2: Price ~ Type + Horsepower + RPM + Wheelbase + Width + Origin
  Res.Df    RSS Df Sum of Sq      F  Pr(>F)
1     83 2216.4
2     82 2036.6  1    179.86 7.2419 0.00863 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

En este caso podemos rechazar la hipótesis nula para un nivel de significancia de 0.05 por lo que podemos mantener RPM como predictor.

Apartado 2.

Selecciona el mejor modelo con el método stepwise.

Resolución.

Otro método de selección de un modelo óptimo es el que plantea `stepwise`. Existen dos enfoques:

- **Forward Selection (selección hacia delante)**. Se parte aquí de un modelo vacío y se van agregando una a una las variables predictoras, evaluando el impacto de cada adición en el rendimiento del modelo. En cada caso, se agrega la variable que más mejora la métrica de desempeño que estemos utilizando (como AIC), hasta que no se mejore más o se alcance algún criterio de detención.
- **Backward Elimination (eliminación hacia atrás)**. Se empieza con un modelo que incluye todas las variables predictoras y, en cada paso, elimina la variable menos significativa según algún criterio de evaluación (como AIC) hasta que eliminar más variables empeore el modelo o se alcance un criterio de detención.

Existen alternativas que mezclarían los dos métodos.

Forward stepwise.

El recorrido se muestra a continuación (Ver código, en el PDF ocupa mucho).

Vemos que el mejor modelo al que acaba llegando el proceso iterativo en modo `forward` es al que posee las variables predictoras Price, Horsepower, Type, Origin, Wheelbase, Width, RPM. El valor final obtenido para `AIC` ha sido de 309.0368868.

Backward stepwise.

El recorrido se muestra a continuación (Ver código, en el PDF ocupa mucho).

Vemos que el mejor modelo al que acaba llegando el proceso iterativo en modo `backward` es al que posee las variables predictoras Price, Horsepower, RPM, Length, Wheelbase, Width, Origin. El valor final obtenido para `AIC` ha sido de 310.0595247.

Método híbrido.

El recorrido se muestra a continuación (Ver código, en el PDF ocupa mucho).

Vemos que el mejor modelo al que acaba llegando el proceso iterativo en modo `both` es al que posee las variables predictoras Price, Horsepower, RPM, Length, Wheelbase, Width, Origin. El valor final obtenido para `AIC` ha sido de 310.0595247.

El modelo seleccionado será el que obtenga el valor de `AIC` más bajo.

El método seleccionado es el perteneciente a forward, donde se obtuvo un `AIC` de 309.0368868 y utilizando como variables predictoras Horsepower, Type, Origin, Wheelbase, Width, RPM.

Apartado 3.

Selecciona el mejor modelo con el método stepwise considerando la variable Passengers como factor. Contesta a las siguientes preguntas:

- ¿Qué % de la varianza de Price explica el modelo?
- ¿Podrías depurar el modelo?
- ¿Cuál es el efecto de la variable Origin sobre Price?

Resolución.

Convertimos a factor de manera manual la variable `Passengers` en primer lugar. Hacemos una copia de nuestro dataset para no modificarlo: `cars2`.

Se debe ahora repetir lo hecho en el apartado anterior (ver código .Rmd para mostrar las trazas.)

El método seleccionado es el perteneciente a both, donde se obtuvo un `AIC` de 309.0368868 y utilizando como variables predictoras Type, Horsepower, RPM, Wheelbase, Width, Origin. Ahora que tenemos el mejor modelo podemos responder a las preguntas realizadas.

- ¿Qué % de la varianza de Price explica el modelo?

```
Call:
lm(formula = Price ~ Type + Horsepower + RPM + Wheelbase + Width +
    Origin, data = cars2)

Residuals:
     Min       1Q   Median       3Q      Max
-11.2403  -2.6302  -0.1117   2.0195  22.6970

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 46.649677  27.321819   1.707 0.091529 .
TypeLarge    1.299684   2.914914   0.446 0.656864
TypeMidsize  3.961813   1.914233   2.070 0.041632 *
TypeSmall    0.334407   2.088371   0.160 0.873174
TypeSporty   3.015877   2.299341   1.312 0.193307
TypeVan     -1.849234   2.874896  -0.643 0.521866
Horsepower   0.160788   0.017971   8.947 9.17e-14 ***
RPM         -0.003695   0.001373  -2.691 0.008630 **
Wheelbase    0.663732   0.212168   3.128 0.002434 **
Width       -1.434571   0.388931  -3.688 0.000404 ***
OriginUSA   -3.217185   1.296097  -2.482 0.015098 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.984 on 82 degrees of freedom
Multiple R-squared:  0.7627,    Adjusted R-squared:  0.7338
F-statistic: 26.36 on 10 and 82 DF,  p-value: < 2.2e-16
```

El valor obtenido para el coeficiente de determinación R^2 ha sido de 0.7627498. Así pues, 76.2749832 % sería el porcentaje de variabilidad explicado por el modelo para la variable dependiente `Precio`.

- ¿Podrías depurar el modelo?

Para depurar el modelo podemos en primer lugar comprobar si existe interacción entre el factor Type y el resto de varables numéricas.

```
Call:
lm(formula = Price ~ Horsepower * Type + RPM + Wheelbase + Width +
    Origin, data = cars2)
```

```
Residuals:
     Min       1Q   Median       3Q      Max
-11.6781  -2.4510  -0.0023   1.8266  22.3651

Coefficients:
                        Estimate Std. Error t value Pr(>|t|)
(Intercept)            42.706728  32.041668   1.333  0.18651
Horsepower              0.131645   0.062747   2.098  0.03918 *
TypeLarge               2.857140  15.413662   0.185  0.85343
TypeMidsize            -3.717117   8.783656  -0.423  0.67334
TypeSmall              -0.086591   9.696753  -0.009  0.99290
TypeSporty              0.923248   8.665213   0.107  0.91543
TypeVan                -0.104830  18.219851  -0.006  0.99542
RPM                    -0.003313   0.001494  -2.218  0.02951 *
Wheelbase               0.573754   0.259554   2.211  0.03004 *
Width                  -1.212680   0.428715  -2.829  0.00596 **
OriginUSA              -3.146679   1.365638  -2.304  0.02391 *
Horsepower:TypeLarge   -0.003529   0.095190  -0.037  0.97052
Horsepower:TypeMidsize  0.049582   0.062788   0.790  0.43214
Horsepower:TypeSmall   -0.010562   0.082597  -0.128  0.89858
Horsepower:TypeSporty   0.012830   0.064095   0.200  0.84187
Horsepower:TypeVan     -0.009579   0.121726  -0.079  0.93748
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.068 on 77 degrees of freedom
Multiple R-squared:  0.7696,    Adjusted R-squared:  0.7247
F-statistic: 17.15 on 15 and 77 DF,  p-value: < 2.2e-16



Call:
lm(formula = Price ~ Horsepower + RPM * Type + Wheelbase + Width +
    Origin, data = cars2)

Residuals:
     Min       1Q   Median       3Q      Max
-10.5060  -2.5012   0.0308   2.1003  22.3196

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)     51.4542824 32.2227840   1.597 0.114400
Horsepower       0.1627970  0.0192556   8.455 1.35e-12 ***
RPM             -0.0028926  0.0030218  -0.957 0.341430
TypeLarge       16.9718799 21.6175516   0.785 0.434806
TypeMidsize      0.1630394 18.4103991   0.009 0.992957
TypeSmall        1.3125023 21.0167692   0.062 0.950366
TypeSporty      20.1241094 20.8686930   0.964 0.337904
TypeVan         13.0063318 32.1842443   0.404 0.687244
Wheelbase        0.6567124  0.2197635   2.988 0.003763 **
Width           -1.5656102  0.4241473  -3.691 0.000415 ***
OriginUSA       -2.8489464  1.3669441  -2.084 0.040461 *
RPM:TypeLarge   -0.0030751  0.0042247  -0.728 0.468885
RPM:TypeMidsize  0.0007881  0.0034077   0.231 0.817730
```

```
RPM:TypeSmall   -0.0002445  0.0038076  -0.064 0.948959
RPM:TypeSporty  -0.0031534  0.0038551  -0.818 0.415881
RPM:TypeVan     -0.0028643  0.0065171  -0.440 0.661529
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.068 on 77 degrees of freedom
Multiple R-squared:  0.7696,    Adjusted R-squared:  0.7248
F-statistic: 17.15 on 15 and 77 DF,  p-value: < 2.2e-16


Call:
lm(formula = Price ~ Horsepower + RPM + Wheelbase * Type + Width +
    Origin, data = cars2)

Residuals:
     Min       1Q   Median       3Q      Max
-11.5691  -2.5836  -0.1384   1.8653  21.9852

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)           1.394e+02  7.651e+01   1.822  0.07232 .
Horsepower            1.639e-01  2.131e-02   7.694 3.95e-11 ***
RPM                  -3.860e-03  1.523e-03  -2.534  0.01329 *
Wheelbase            -1.260e-01  6.778e-01  -0.186  0.85305
TypeLarge            -9.145e+01  1.078e+02  -0.848  0.39881
TypeMidsize          -8.869e+01  7.924e+01  -1.119  0.26651
TypeSmall            -9.522e+01  7.831e+01  -1.216  0.22770
TypeSporty           -1.099e+02  9.114e+01  -1.206  0.23162
TypeVan              -5.582e+01  9.901e+01  -0.564  0.57455
Width                -1.600e+00  4.875e-01  -3.282  0.00155 **
OriginUSA            -3.309e+00  1.352e+00  -2.448  0.01662 *
Wheelbase:TypeLarge   9.012e-01  9.981e-01   0.903  0.36940
Wheelbase:TypeMidsize 9.008e-01  7.637e-01   1.179  0.24184
Wheelbase:TypeSmall   9.373e-01  7.725e-01   1.213  0.22875
Wheelbase:TypeSporty  1.116e+00  9.062e-01   1.232  0.22187
Wheelbase:TypeVan     5.555e-01  9.196e-01   0.604  0.54758
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.079 on 77 degrees of freedom
Multiple R-squared:  0.7686,    Adjusted R-squared:  0.7235
F-statistic: 17.05 on 15 and 77 DF,  p-value: < 2.2e-16


Call:
lm(formula = Price ~ Horsepower + RPM + Wheelbase + Width * Type +
    Origin, data = cars2)

Residuals:
     Min       1Q   Median       3Q      Max
-10.7274  -2.3580   0.0681   2.0476  21.9864

Coefficients:
```

```
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)        1.418e+02  1.181e+02   1.201  0.23351
Horsepower         1.620e-01  1.972e-02   8.218 3.87e-12 ***
RPM               -4.092e-03  1.494e-03  -2.739  0.00765 **
Wheelbase          6.573e-01  2.529e-01   2.599  0.01120 *
Width             -2.809e+00  1.720e+00  -1.634  0.10643
TypeLarge         -5.829e+01  1.228e+02  -0.475  0.63633
TypeMidsize       -6.035e+01  1.265e+02  -0.477  0.63476
TypeSmall         -1.151e+02  1.227e+02  -0.938  0.35129
TypeSporty        -9.937e+01  1.179e+02  -0.843  0.40209
TypeVan           -1.051e+02  1.316e+02  -0.799  0.42684
OriginUSA         -3.404e+00  1.379e+00  -2.468  0.01580 *
Width:TypeLarge    9.328e-01  1.802e+00   0.518  0.60618
Width:TypeMidsize  9.760e-01  1.867e+00   0.523  0.60257
Width:TypeSmall    1.728e+00  1.829e+00   0.945  0.34773
Width:TypeSporty   1.518e+00  1.751e+00   0.867  0.38865
Width:TypeVan      1.520e+00  1.914e+00   0.794  0.42969
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 5.078 on 77 degrees of freedom
Multiple R-squared:  0.7687,    Adjusted R-squared:  0.7236
F-statistic: 17.06 on 15 and 77 DF,  p-value: < 2.2e-16
```

Ninguna de las interacciones es significativa y la variabilidad explicada es prácticamente igual por lo que descartamos los modelos anteriores, ya que es preferible tener un modelo con menos parámetros.

También podemos plantearnos agrupar categorías del factor Type como por ejemplo Compact y Small, ya que Small presenta el p-valor más alto y esto podría deberse a que aporta lo mismo que la categoría Compact.

```
Call:
lm(formula = Price ~ Type + Horsepower + RPM + Wheelbase + Width +
    Origin, data = cars2)

Residuals:
    Min      1Q  Median      3Q     Max
-11.120  -2.630  -0.114   2.103  22.788

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)   51.318196  27.545924   1.863 0.065999 .
TypeLarge     -2.576501   2.175834  -1.184 0.239736
TypeSmallsize -3.909959   1.875537  -2.085 0.040169 *
TypeSporty    -1.098310   2.265448  -0.485 0.629089
TypeVan       -5.748031   2.256693  -2.547 0.012707 *
Horsepower     0.159996   0.017175   9.316 1.53e-14 ***
RPM           -0.003642   0.001325  -2.749 0.007332 **
Wheelbase      0.646626   0.182232   3.548 0.000641 ***
Width         -1.420593   0.376778  -3.770 0.000304 ***
OriginUSA     -3.221486   1.288190  -2.501 0.014358 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 4.954 on 83 degrees of freedom
Multiple R-squared:  0.7627,    Adjusted R-squared:  0.7369
F-statistic: 29.64 on 9 and 83 DF,  p-value: < 2.2e-16
```

Con este cambio hemos podido obtener un resultado algo mejor, por lo que se convertirá en nuestro `mejor_modelo`

Para depurar más a fondo el modelo, se podría realizar un diagnóstico del mismo y solucionar posibles problemas de linealidad, homocedasticidad, normalidad, outliers... Esto se llevará a cabo en los próximos apartados.

El mejor modelo (tras el proceso de depuración anterior) resulta entonces en:

```
Call:
lm(formula = Price ~ Type + Horsepower + RPM + Wheelbase + Width +
    Origin, data = cars2)

Residuals:
    Min      1Q  Median      3Q     Max
-11.120  -2.630  -0.114   2.103  22.788

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   51.318196  27.545924   1.863 0.065999 .
TypeLarge     -2.576501   2.175834  -1.184 0.239736
TypeSmallsize -3.909959   1.875537  -2.085 0.040169 *
TypeSporty    -1.098310   2.265448  -0.485 0.629089
TypeVan       -5.748031   2.256693  -2.547 0.012707 *
Horsepower     0.159996   0.017175   9.316 1.53e-14 ***
RPM           -0.003642   0.001325  -2.749 0.007332 **
Wheelbase      0.646626   0.182232   3.548 0.000641 ***
Width         -1.420593   0.376778  -3.770 0.000304 ***
OriginUSA     -3.221486   1.288190  -2.501 0.014358 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.954 on 83 degrees of freedom
Multiple R-squared:  0.7627,    Adjusted R-squared:  0.7369
F-statistic: 29.64 on 9 and 83 DF,  p-value: < 2.2e-16
```

- ¿Cuál es el efecto de la vairable `Origin` sobre `Price`?

Tal y como se observa, el mejor modelo obtenido sí tiene presente dicha variable. Al ser una variable categórica esta viene descompuesta en $k-1$ predictores codificados como 0 o 1 (variables *dummy*). En este caso, al tener 2 categorías dispondría de una sola variable dummy: `OriginUSA`. Puede apreciarse que esta posee un coeficiente significativo y que se estima como negativo, indicando que si el coche es de origen estadounidense la variable `Price` debería tender a ser menor (con el resto de variables constantes). Esto puede llegar a tener algo de lógica, ya que coches importados suelen ser más caros por motivos de impuestos, costes de transporte, etc.

Apartado 4.

¿Qué modelo de los apartados anteriores es mejor? Con el que te quedes, realiza el diagnóstico de tu modelo, sin emprender ninguna acción, e indica los problemas que presenta.

Resolución.

El mejor modelo obtenido es el de la forma: **lm(formula = Price ~ Type + Horsepower + RPM + Wheelbase + Width + Origin, data = cars2)**.

Haciendo uso del comando `plot` sobre nuestro objeto que contiene el modelo de regresión obtenemos una serie de gráficos diagnósticos que nos proporcionan información sobre la idoneidad de nuestro modelo.

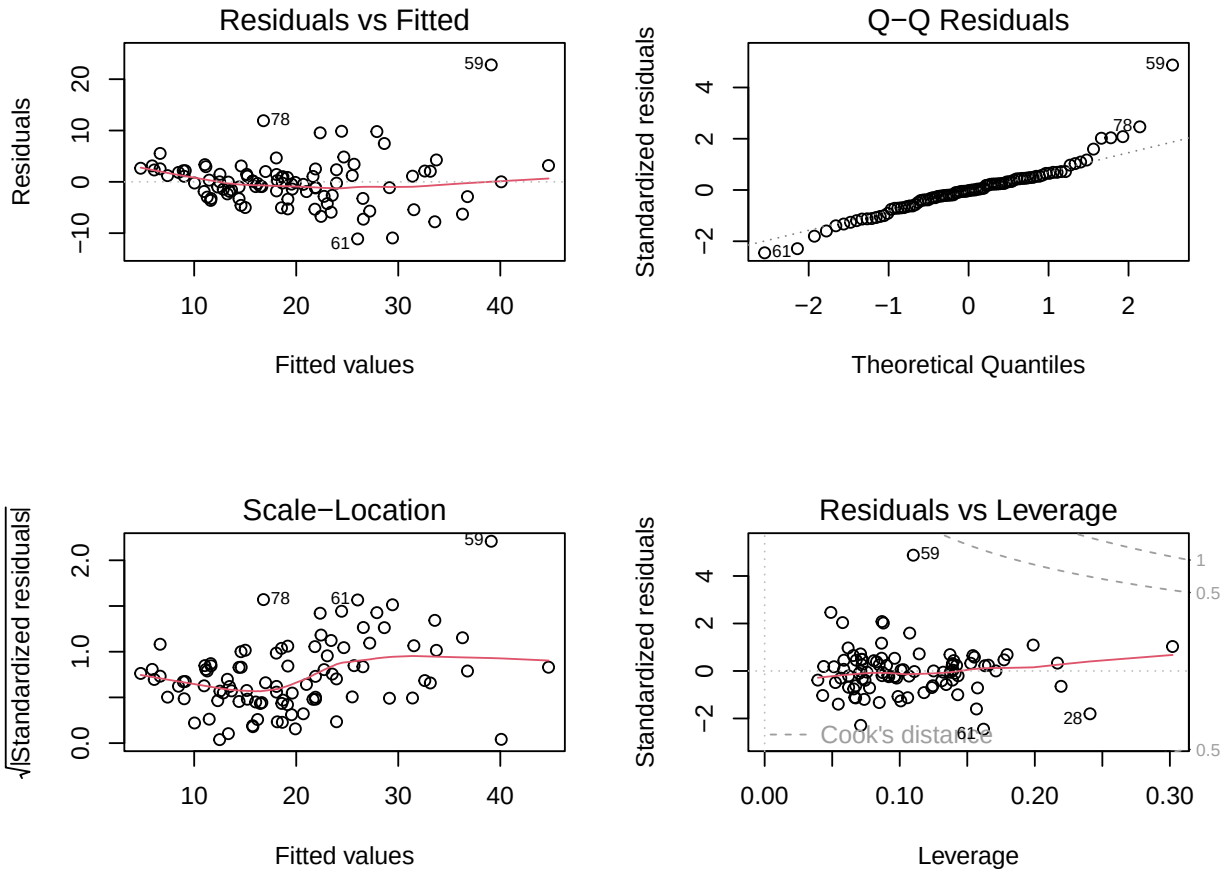

Figura 14. Gráficos de diagnóstico del mejor modelo obtenido.

Podemos observar en el gráfico de arriba a la izquierda cómo los residuos se colocan alrededor de la línea horizontal de cero, indicando que hay más o menos linealidad. Lo que no parece tan claro es la homocedasticidad, ya que hacia la parte central y final crece un poco la variabilidad. Algo similar se observa en el gráfico de `Scale-Location` con los residuos estandarizados (y en valor absoluto). Una gráfica de un estilo similar con los residuos studentizados se muestra en mayor tamaño más abajo.

Por otra parte, en el QQ-plot de los residuos estandarizados obtenemos que varios puntos se van considerablemente de la recta que indica la normalidad. Más adelante, se hará el test de Shapiro-Wilk para verificar si pueden considerarse que siguen una distribución normal o no.

Finalmente, en el gráfico de abajo a la derecha se nos muestran las observaciones que tienen un alto efecto

en los coeficientes de regresión y que, por tanto, pueden influir significativamente en la forma del modelo. Más concretamente, se señalan las observaciones atípicas (residuos fuera de [-2, 2]) e influyentes a posteriori (estadístico de Cook > 0.5 y > 1).

Representamos los residuos studentizados frente a los valores predichos para analizar la linealidad y la homocedasticidad del modelo.

```
`geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```

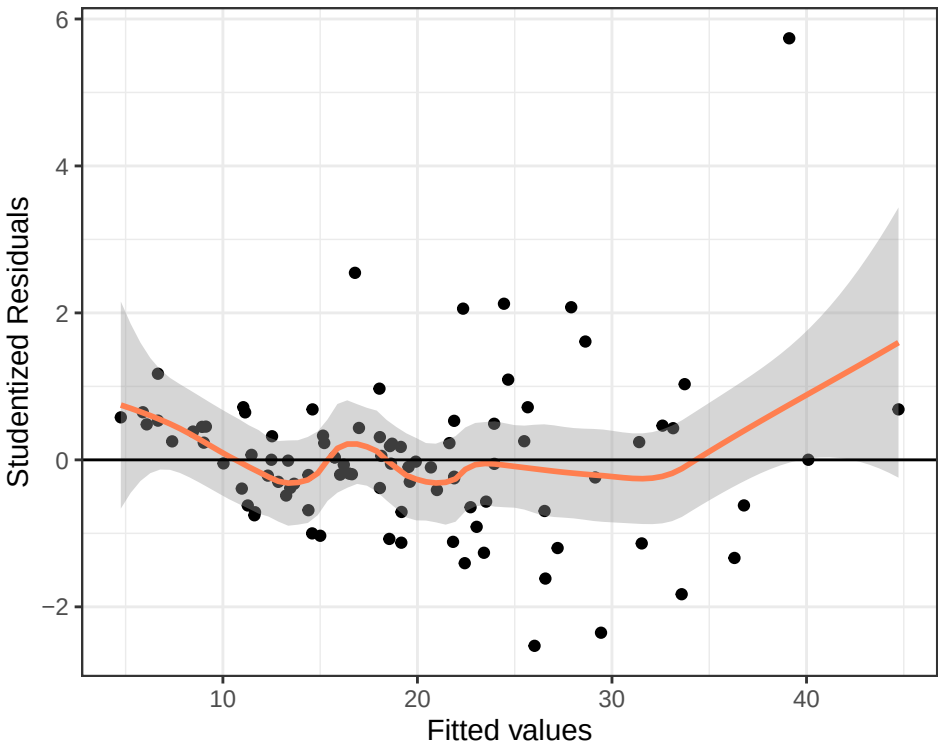

Figura 15. Residuos estudentizados para nuestro modelo.

De nuevo, apreciamos cómo no tenemos una distribución de los residuos del todo uniforme alrededor del cero, apreciando en los extremos una subida.

Puede hacerse el test de Breusch-Pagan para homocedasticidad del modelo. La hipótesis nula de este test es que no hay heterocedasticidad en los residuos.

```
    studentized Breusch-Pagan test

data:  mejor_modelo
BP = 17.238, df = 9, p-value = 0.04511
```

Se ha obtenido un p-valor de 0.045111, por lo que se podría decir que rechazamos la hipótesis nula: hay heterocedasticidad con un nivel de confianza de 0.05.

A continuación se muestran algunas gráficas de residuos parciales, para tratar de evaluar la contribución individual de cada variable independiente en nuestro modelo. Son los residuos al eliminar cada predictor.

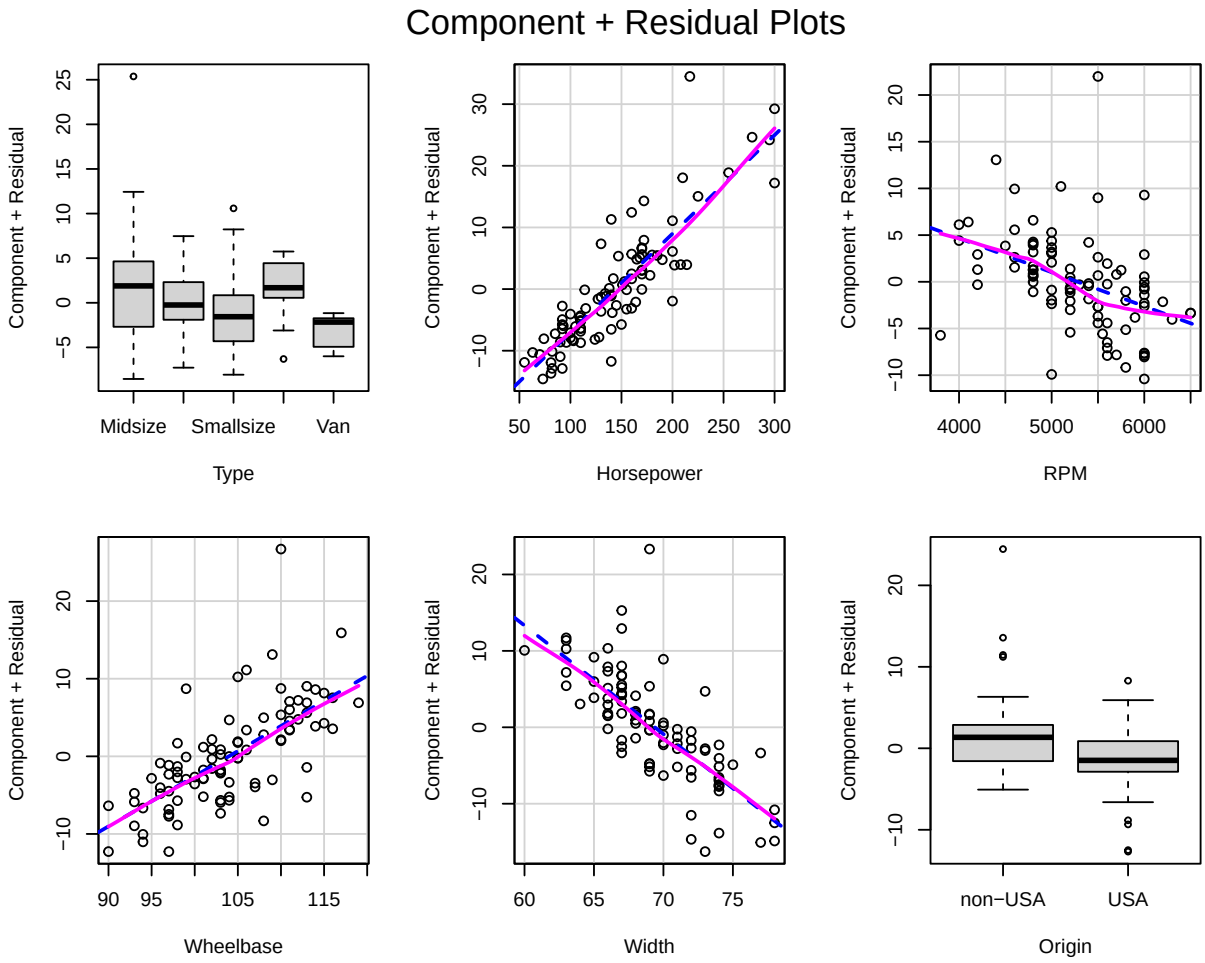

Figura 16. Gráficos parciales de residuos (componentes + residuos) de nuestro modelo.

No se identifican patrones no lineales ni violaciones de la homocedasticidad en estos gráficos.

Los residuos studentizados se muestran a continuación, junto con el test de shapiro para determinar de forma algo más rigurosa si pueden considerarse normales o no.

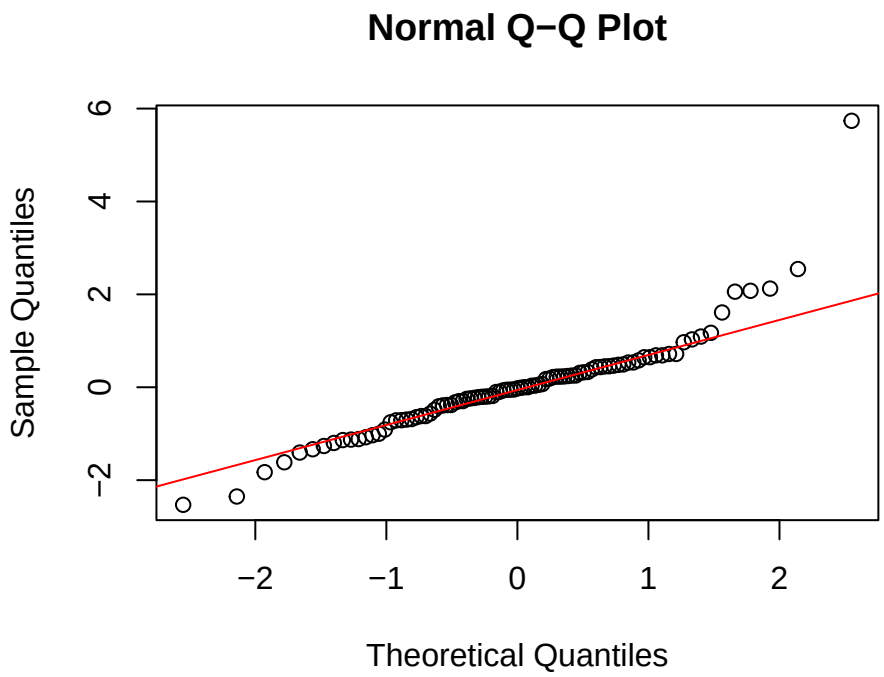

Figura 17. Normalidad en los residuos studentizados.

```
    Shapiro-Wilk normality test

data:  res_stu
W = 0.87558, p-value = 2.672e-07
```

El p-valor del Shapiro-Wilk test de normalidad es muy pequeño (2.6718323×10^{-7}) e indica que la distribución de los residuos no es normal. Esto puede verse en la gráfica de los cuantiles de los residuos frente a los cuantiles teóricos de una distribución normal, ya que en los extremos se alejan bastante de la distribución normal, especialmente el último cuantil de la muestra. Falla por tanto la normalidad del modelo.

Pasamos ahora sí a analizar los posibles `outliers`. Usualmente un dato será anómalo si el residuo studentizado está fuera de rango (-2, 2) y extremo si está fuera de (-3, 3). Utilizaremos la función `outlierTest()` para encontrarlos.

```
   rstudent unadjusted p-value Bonferroni p
59 5.737169         1.5633e-07   1.4539e-05
```

```
      Type Price MPG.city MPG.highway EngineSize Horsepower  RPM Rev.per.mile Fuel.tank.capacity
59 Midsize  61.9       19          25        3.2        217 5500         2220               18.5
   Passengers Length Wheelbase Width Weight  Origin
59          5    187       110    69   3525 non-USA
```

Observamos que la única observación catalogada como outlier ha sido la número 59, la cual ya habíamos observado previamente en la gráfica de residuos studentizados frente a valores predichos de la variable

dependiente. Vemos que su valor de residuo studentizado es muy elevado (5.7371693).

Podemos averiguar ahora si este outlier es una información influyente. Para ello, en primer lugar representamos la distancia de Cook, que es una medida de la influencia de una observación en la estimación de los coeficientes del modelo, frente a los valores ajustados.

Una observación será influyente si tiene valores para la distancia de Cooks que cumplen:

$$D_i > \frac{4}{n-p-1}, \tag{10}$$

siendo n el número de muestras, p el número de coeficientes estimados en el modelo (incluyendo el intercepto).

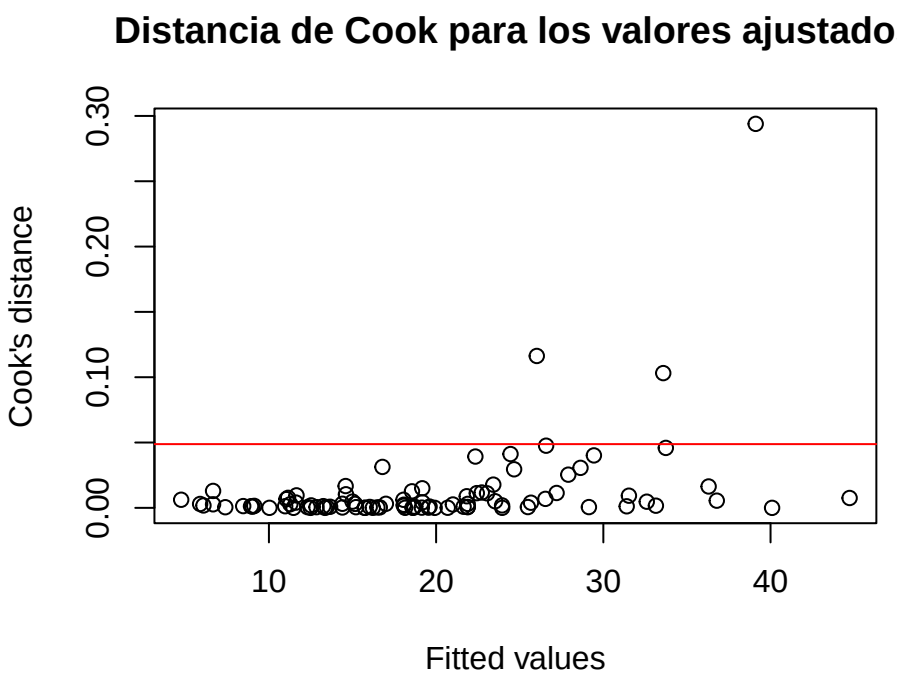

Figura 18. Distancias de Cook para los valores ajustados en nuestro modelo..

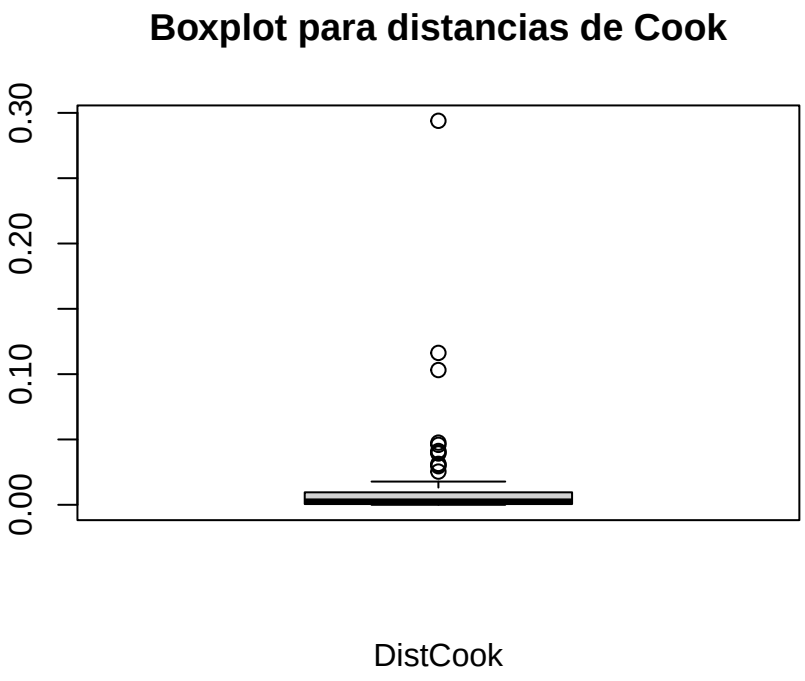

Figura 19. Distancias de Cook para los valores ajustados en nuestro modelo..

```
     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
3.000e-08 5.672e-04 2.563e-03 1.234e-02 9.561e-03 2.939e-01
```

Vemos como hay dos valores con distancia de Cook especialmente grandes, siendo el valor máximo de 0.2939342 mientras que la mediana es 0.0025627. Si bien el consenso teórico es que una distancia de Cook lo bastante grande si es mayor que uno, podemos considerar que estos valores (en especial el máximo) son suficientemente grandes como para ser analizados individualmente.

A continuación, usamos la función de `influencePlot`. Esta función representa los residuos studentizados frente a los `hat-values` (valores de `leverage`), que son medidas que indican cuánto una observación específica influye en la estimación de sus propios valores ajustados y con un código tamaño y color dado por la distancia de Cook.

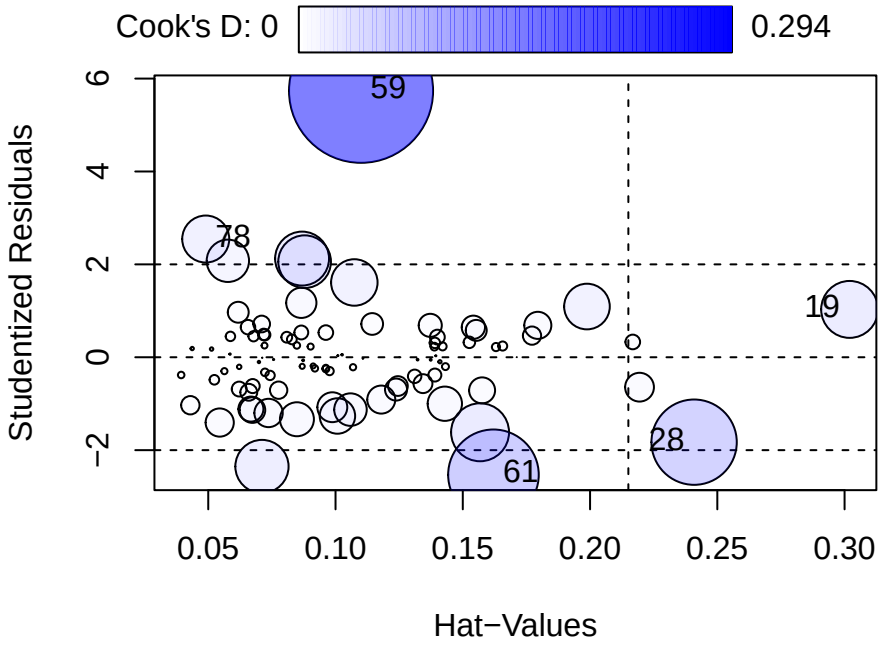

Figura 20. Gráfico de influencia para nuestro modelo (Distancia de cook y leverage).

```
     StudRes        Hat      CookD
19  1.030295 0.30197843 0.04588896
28 -1.828194 0.24087983 0.10314507
59  5.737169 0.11003409 0.29393423
61 -2.530600 0.16204942 0.11627398
78  2.545428 0.04901218 0.03132475
```

Vemos como el outlier que habíamos localizado, correspondiente a la observación 59, es una observación influyente con la mayor distancia de Cook. Un resumen de las medidas de influencias se muestra a continuación.

```
Potentially influential observations of
     lm(formula = Price ~ Type + Horsepower + RPM + Wheelbase + Width +      Origin, data = cars2) :

   dfb.1_ dfb.TypL dfb.TypSm dfb.TypSp dfb.TypV dfb.Hrsp dfb.RPM dfb.Whlb dfb.Wdth dfb.OUSA dffit
11   0.00    0.00      0.00      0.00      0.00     0.00     0.00    0.00     0.00     0.00     0.00
17  -0.02   -0.03      0.01     -0.04      0.07    -0.01    -0.02   -0.08     0.09    -0.01     0.17
19   0.19   -0.01      0.08      0.01      0.11     0.44    -0.23   -0.32     0.05     0.06     0.68
36   0.01    0.03     -0.05     -0.12     -0.16    -0.02    -0.02   -0.21     0.15    -0.12    -0.34
59   0.73   -0.37     -0.51     -0.16     -0.41     1.08_*  -0.64    0.78    -1.22_*  -0.26     2.02_*
61  -0.05    0.64      0.08      0.01      0.60     0.02     0.66   -0.36     0.11     0.02    -1.11_*
76  -0.15    0.32      0.24      0.29      0.21    -0.24     0.15    0.05     0.09    -0.26    -0.65
78  -0.17    0.04      0.25     -0.03      0.03    -0.02     0.20   -0.12     0.22    -0.19     0.58
   cov.r   cook.d hat
11  1.36_*  0.00   0.17
17  1.42_*  0.00   0.22
19  1.42_*  0.05   0.30
```

```
36  1.38_*  0.01   0.22
59  0.04_*  0.29   0.11
61  0.64_*  0.12   0.16
76  0.63_*  0.04   0.07
78  0.55_*  0.03   0.05
```

Las observaciones influyentes dadas por la función influence.measures también incluyen este outlier, que además es la observación que presenta mayor número de valores significativos para los DfBetas (dfb). Las DfBetas muestran las diferencias en los coeficientes estimados al excluir la observación, por lo que este outlier podría estar afectando bastante al modelo.

Finalmente, vamos a analizar la colinealidad de los predictores.

La colinealidad es cuando dos o más predictores están muy correlacionados. Se encuentra lo siguiente cuando la hay:

- Es difícil separar el efecto individual de cada predictor.
- Las varianzas de los predictores estarán infladas, afectando al test t y al intervalo de confianza.

Veamos primero gráficos por pares para las variables presentes en nuestro modelo:

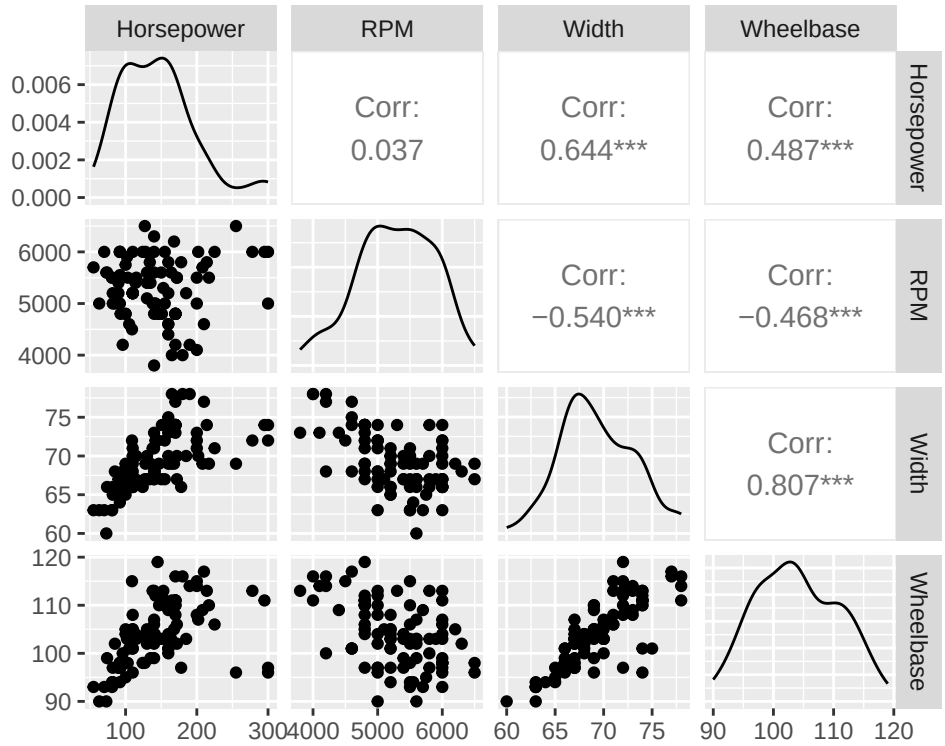

Figura 21. Gráficos por pares para las variables numéricas presentes en nuestro modelo.

Representando por pares las variables numéricas de nuestro modelo vemos que existe una relación lineal entre varias variables, especialmente entre las variables `Wheelbase` y `Width`. Esto era de esperar, ya que la distancia entre ruedas traseras y delanteras intuitivamente puede estar relacionada con la anchura del propio vehículo. Es coherente que las dimensiones de los coches mantengan una determinada proporción por motivos de diseño, reflejándose en esta relación.

```
           Horsepower          RPM      Width  Wheelbase
Horsepower 1.00000000  0.03668821  0.6444134  0.4868542
RPM        0.03668821  1.00000000 -0.5397211 -0.4678123
Width      0.64441342 -0.53972113  1.0000000  0.8072134
Wheelbase  0.48685421 -0.46781229  0.8072134  1.0000000
```

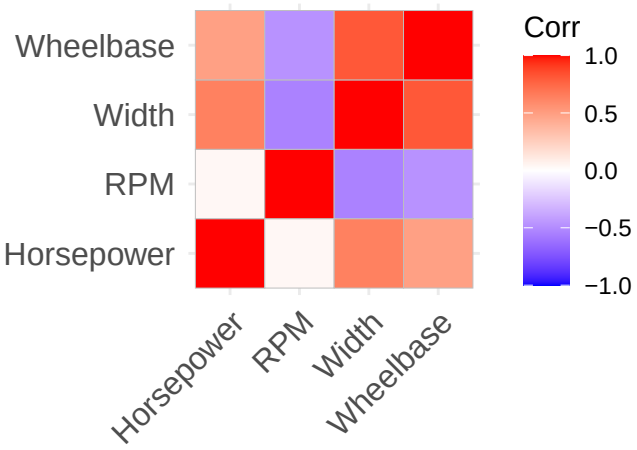

Figura 22. Mapa de correlaciones para las variables numéricas en el modelo.

Tal y como reflejaban las gráficas, la correlación entre estas variables es algo alta, siendo la mayor correlación entre las variables `Wheelbase` y `Width`, lo cual tiene sentido por lo explicado anteriormente.

Para ver la multicolinealidad, calculamos el factor de inflación de la varianza (`VIF`). El VIF se calcula para cada variable independiente en nuestro modelo y proporciona una medida de cuánto aumenta la varianza de un coeficiente estimado debido a la multicolinealidad con las otras variables predictoras. La expresión es la siguiente:

$$\text{VIF}_j = (1 - R_j^2)^{-1}, \tag{11}$$

donde R_j^2 es el coeficiente de determinación de la regresión X_j, como variable respuesta, frente a los demás predictores.

Nos deberíamos preocupar cuando $\text{VIF}_j > 5$.

```
               GVIF Df GVIF^(1/(2*Df))
Type       7.422073  4        1.284741
Horsepower 3.032956  1        1.741538
RPM        2.343236  1        1.530763
Wheelbase  5.789085  1        2.406052
Width      7.598963  1        2.756622
Origin     1.570276  1        1.253106
```

Sumado a la alta correlación de la variable `Width` con el resto de variables, tenemos que esta es la que tiene el VIF más alto, por lo que la multicolinealidad es significativa.

Apartado 5.

Emprende ahora las acciones que te parezcan oportunas e indica los problemas que has conseguido solucionar o mejorar un poco.

Resolución.

En primer lugar, para mejorar la homocedasticidad del modelo podemos realizar alguna transformación sobre la variable dependiente. En concreto, vamos a buscar una transformación de Box-Cox. Esta es posible de ser realizada puesto que nuestra variable dependiente `Price` es no negativa. Posee la siguiente expresión:

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

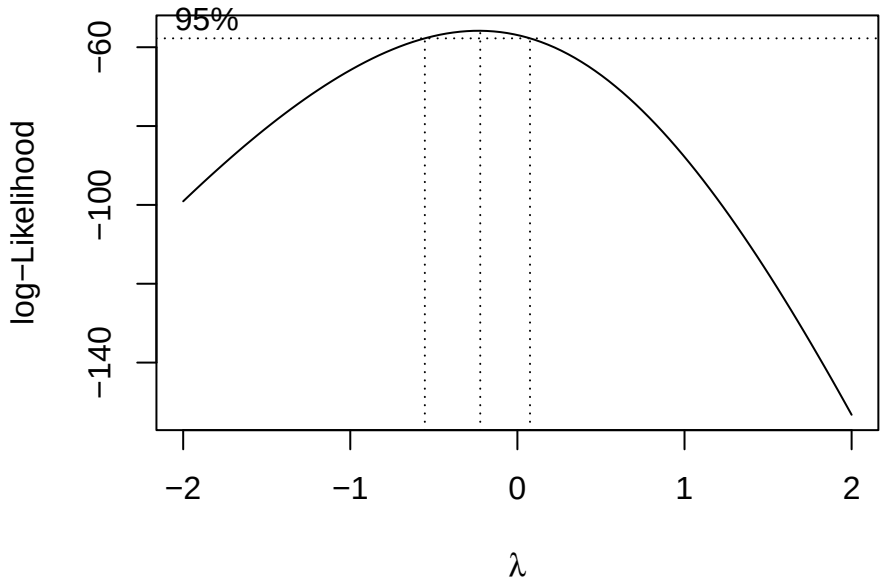

Figura 23. Gráfica para estimar el valor de lambda necesario en la transformación Box-cox.

```
[1] -0.2222222
```

En este caso la λ que maximiza la verosimilitud sería -0.22. Este es un valor bastante próximo a 0 y en las transformaciones de Box-Cox se toma el logaritmo cuando lambda tiende a cero. Por tanto, por simplicidad podemos tomar directamente una transformación logarítmica, que además tiene una interpretación más directa. Se comparará también con el valor que se obtendría para el R^2 ajustado con la transformación exacta.

Por otro lado, eliminamos la observación 59, que es un outlier y una observación influyente que puede estar afectando demasiado a los coeficietes del modelo.

Finalmente, como hemos visto que `Width` presenta una alta correlación con otras variables como `Length` y `Wheelbase` y una alta multicolinealidad y podría estar aportando información redundante, nos planteamos un modelo sin esta variable. Veremos si mejora el R^2 ajustado o por si el contrario no lo hace.

```
Call:
lm(formula = log(Price) ~ Type + Horsepower + Width + RPM + Wheelbase +
    Origin, data = cars3)

Residuals:
     Min       1Q   Median       3Q      Max
-0.40412 -0.10574 -0.00454  0.09046  0.52128

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.051e+00  1.049e+00   1.955  0.05401 .
TypeLarge    4.226e-02  1.113e-01   0.380  0.70518
TypeMidsize  9.772e-02  7.353e-02   1.329  0.18760
TypeSmall   -2.241e-01  8.034e-02  -2.790  0.00657 **
TypeSporty   7.263e-02  8.838e-02   0.822  0.41361
TypeVan     -9.365e-02  1.098e-01  -0.853  0.39618
Horsepower   5.600e-03  7.020e-04   7.976 8.36e-12 ***
Width       -2.676e-02  1.526e-02  -1.754  0.08327 .
RPM         -7.834e-05  5.294e-05  -1.480  0.14278
Wheelbase    2.287e-02  8.232e-03   2.778  0.00679 **
OriginUSA   -1.598e-01  4.954e-02  -3.226  0.00181 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1903 on 81 degrees of freedom
Multiple R-squared:  0.8315,    Adjusted R-squared:  0.8107
F-statistic: 39.98 on 10 and 81 DF,  p-value: < 2.2e-16



Call:
lm(formula = (Price^(-0.22) - 1)/-0.22 ~ Type + Horsepower +
    Width + RPM + Wheelbase + Origin, data = cars3)

Residuals:
      Min        1Q    Median        3Q       Max
-0.204763 -0.053906 -0.002908  0.051293  0.267835

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.461e+00  5.494e-01   2.659  0.00944 **
TypeLarge    1.434e-02  5.831e-02   0.246  0.80636
TypeMidsize  4.401e-02  3.852e-02   1.143  0.25658
TypeSmall   -1.390e-01  4.209e-02  -3.303  0.00142 **
TypeSporty   3.764e-02  4.630e-02   0.813  0.41858
TypeVan     -5.334e-02  5.751e-02  -0.928  0.35642
Horsepower   2.766e-03  3.678e-04   7.520 6.58e-11 ***
Width       -1.122e-02  7.995e-03  -1.404  0.16423
RPM         -3.315e-05  2.773e-05  -1.195  0.23545
Wheelbase    1.220e-02  4.312e-03   2.830  0.00586 **
OriginUSA   -8.481e-02  2.595e-02  -3.268  0.00159 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 0.09969 on 81 degrees of freedom
Multiple R-squared:  0.8376,    Adjusted R-squared:  0.8175
F-statistic: 41.77 on 10 and 81 DF,  p-value: < 2.2e-16
```

No consideramos que merezca la pena complicar tanto el modelo por una ganancia tan escasa en el R^2 ajustado, por lo que continuaremos con la transformación logarítmica. Ahora bien, pasamos ahora a ver qué sucede si quitamos la variable predictora `Width`.

```
Call:
lm(formula = log(Price) ~ Type + Horsepower + RPM + Wheelbase +
    Origin, data = cars3)

Residuals:
     Min       1Q   Median       3Q      Max
-0.37468 -0.10117 -0.01469  0.09525  0.49749

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  8.552e-01  8.071e-01   1.060 0.292447
TypeLarge   -6.440e-03  1.091e-01  -0.059 0.953098
TypeMidsize  6.943e-02  7.264e-02   0.956 0.341994
TypeSmall   -2.586e-01  7.888e-02  -3.279 0.001531 **
TypeSporty   1.031e-02  8.194e-02   0.126 0.900140
TypeVan     -1.440e-01  1.073e-01  -1.342 0.183325
Horsepower   4.834e-03  5.565e-04   8.686 3.03e-13 ***
RPM         -3.738e-05  4.810e-05  -0.777 0.439340
Wheelbase    1.593e-02  7.307e-03   2.179 0.032164 *
OriginUSA   -1.828e-01  4.837e-02  -3.780 0.000296 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1927 on 82 degrees of freedom
Multiple R-squared:  0.8251,    Adjusted R-squared:  0.806
F-statistic:    43 on 9 and 82 DF,  p-value: < 2.2e-16
```

Observamos una caída en el valor de R^2 ajustado. Por tanto, preferimos continuar capturando en nuestro modelo dicha variable predictora.

Empezamos mostrando el gráfico general para el diagnóstico con este nuevo modelo:

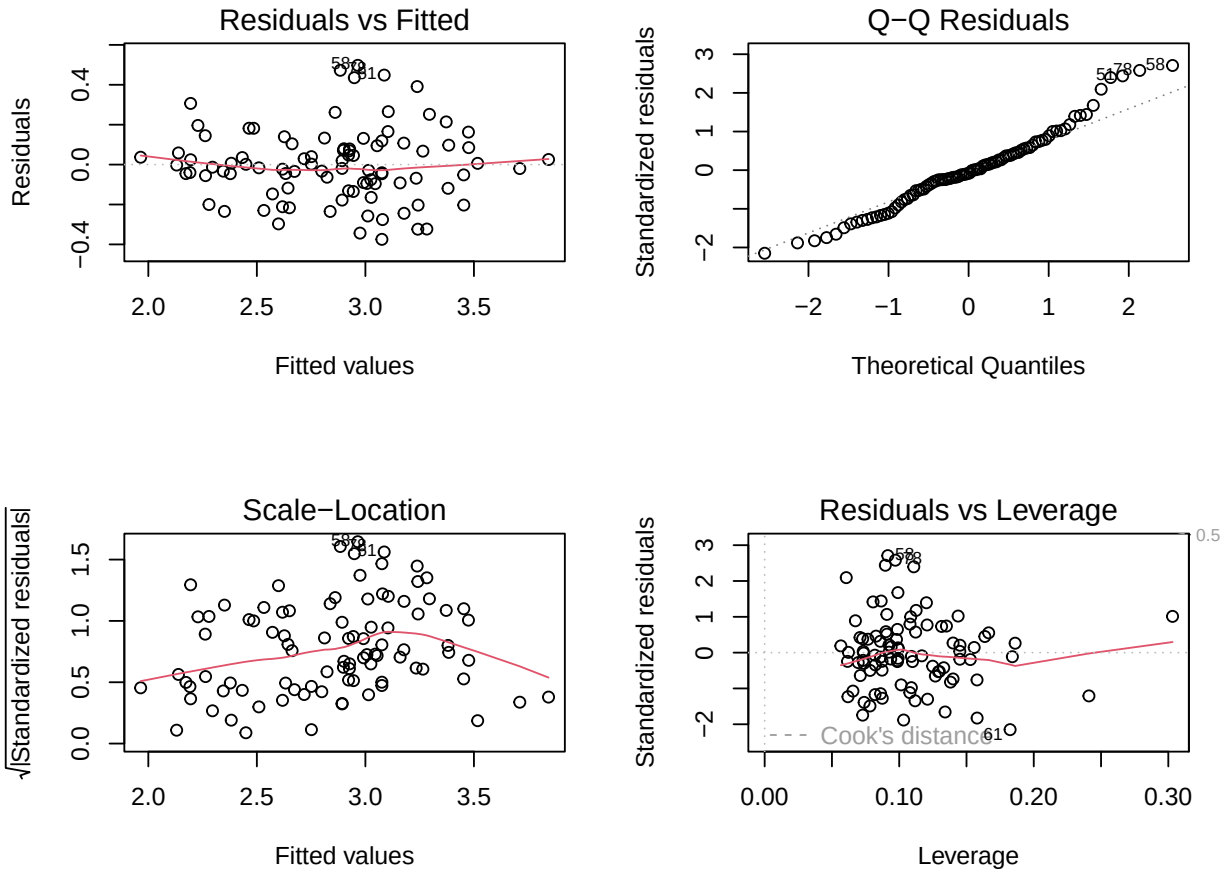

Figura 24. Gráficos de diagnóstico del modelo modificado.

Puede observarse a simple vista que la gráfica de residuos frente a valores ajustados ha mejorado. Veamos las cosas con mayor detalle.

```
‘geom_smooth()‘ using method = ’loess’ and formula = ’y ~ x’
```

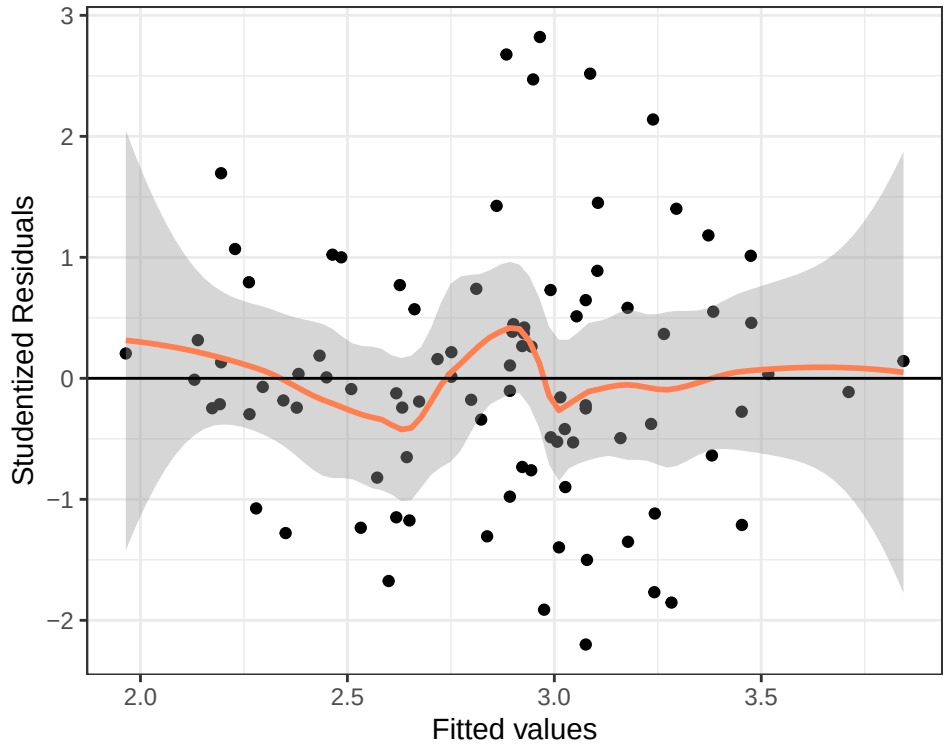

Figura 25. Residuos estudentizados para nuestro modelo modificado.

Ahora los residuos studentizados parecen distribuirse de forma más uniforme entorno al cero y la variablidad también es más constante. La ausencia del outlier de la observación 59 también proporciona mejor resultado en la parte derecha del gráfico.

```
    studentized Breusch-Pagan test

data:  modelo_mod
BP = 11.627, df = 9, p-value = 0.2352
```

Ahora el test de homocedasticidad proporciona un p-valor bastante más alto (0.2351541), lo que confirma la mejora en este aspecto. Ahora no se podría rechazar que haya homocedasticidad.

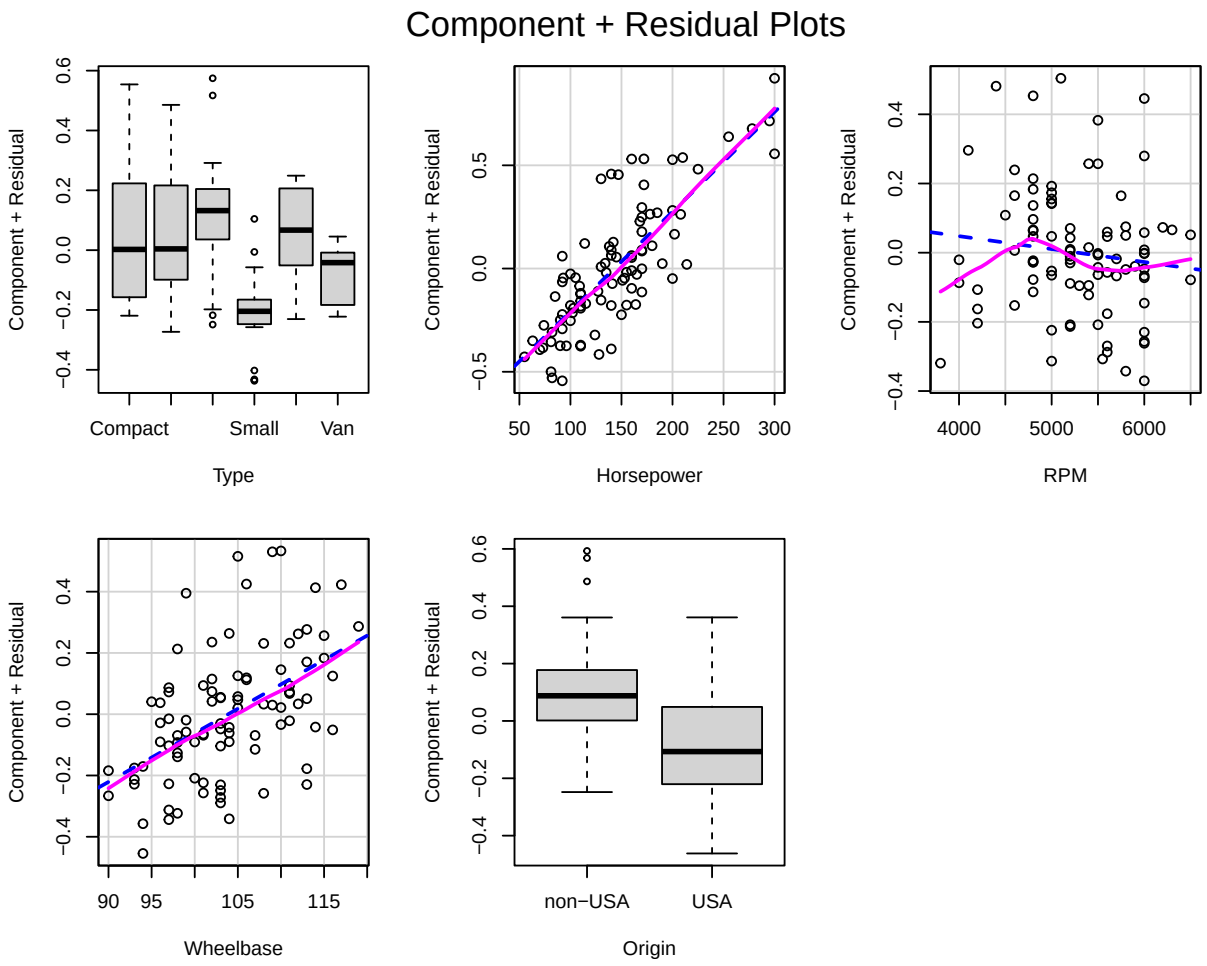

Figura 26. Gráficos parciales de residuos (componentes + residuos) de nuestro modelo modificado.

Los residuos parciales siguen sin mostrar grandes problemas de linealidad ni homocedasticidad. El único comportamiento algo extraño se sigue dando por la variable `RPM`.

Pasamos a comprobar de nuevo la normalidad de los residuos studentizados.

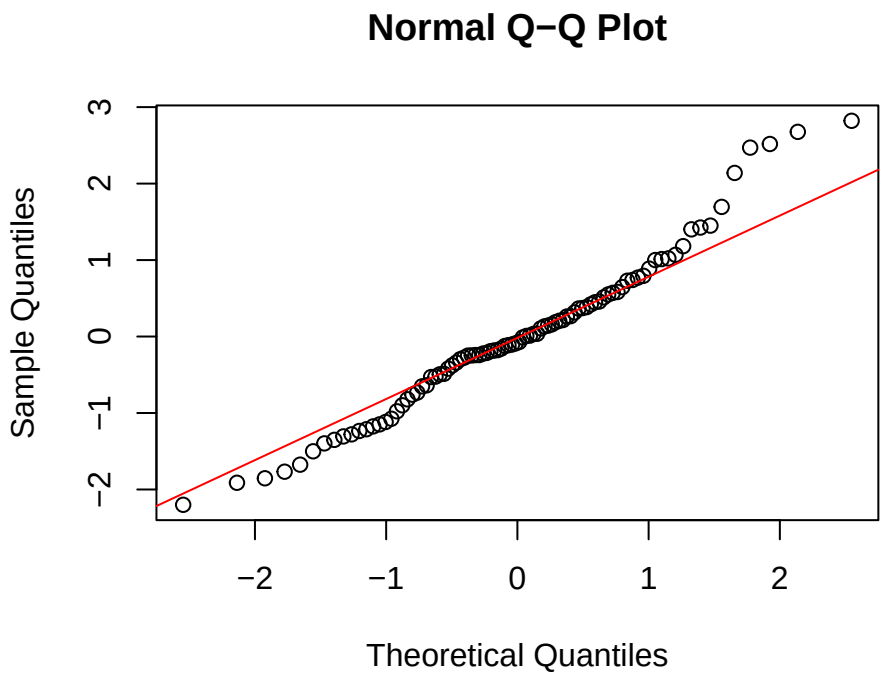

Figura 27. Normalidad en los residuos studentizados.

```
    Shapiro-Wilk normality test

data:  res_stu
W = 0.9734, p-value = 0.05659
```

El Shapiro-Wilk test ahora no nos permite rechazar la hipótesis nula de que la distribución es normal con un nivel de significancia de 0.05 (el p-valor obtenido ha sido 0.0565868). Si bien los cuantiles de los residuos studentizados todavía presentan desviaciones en los extremos, estas desviaciones son algo menos pronunciadas. Sobre todo, ya no observamos ese valor claramente más desviado que el resto, que presumiblemente se correspondía al outlier eliminado y empeoraba bastante la normalidad de los residuos.

Analizamos ahora los outliers del nuevo modelo:

```
No Studentized residuals with Bonferroni p < 0.05
Largest |rstudent|:
   rstudent unadjusted p-value Bonferroni p
58 2.821104          0.0060172      0.55358
```

```
      Type Price MPG.city MPG.highway EngineSize Horsepower  RPM Rev.per.mile Fuel.tank.capacity
58 Compact  31.9       20          29        2.3        130 5100         2425               14.5
   Passengers Length Wheelbase Width Weight  Origin
58          5    175       105    67   2920 non-USA
```

Esta vez no se han encontrado outliers con el test de Bonferroni, como era de esperar. Aún así, la función `outlierTest()` ha retornado el valor que mayor residuo studentizado posee. Vemos que es incluso inferior a

3. Por tanto, considerarlo como un outlier puede ser demasiado arbitrario. Analizamos ahora si este posible outlier puede ser una observación influyente.

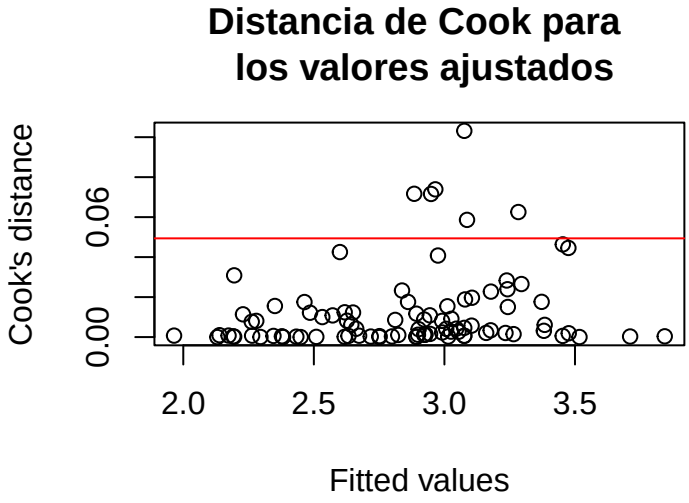

Figura 28. Distancias de Cook para los valores ajustados en nuestro modelo modificado.

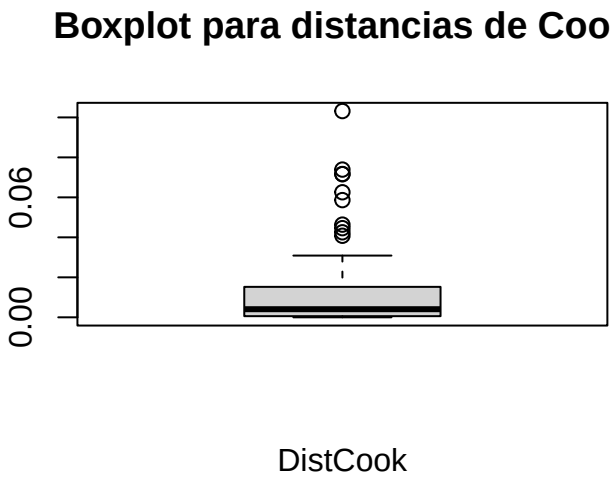

Figura 29. Distancias de Cook para los valores ajustados en nuestro modelo modificado.

```
     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
0.0000004 0.0005765 0.0040468 0.0123683 0.0151465 0.1031695
```

En esta ocasión ya no encontramos observaciones con una distancia de Cook tan grande como antes, sino que ahora el valor máximo se queda en 0.1031695.

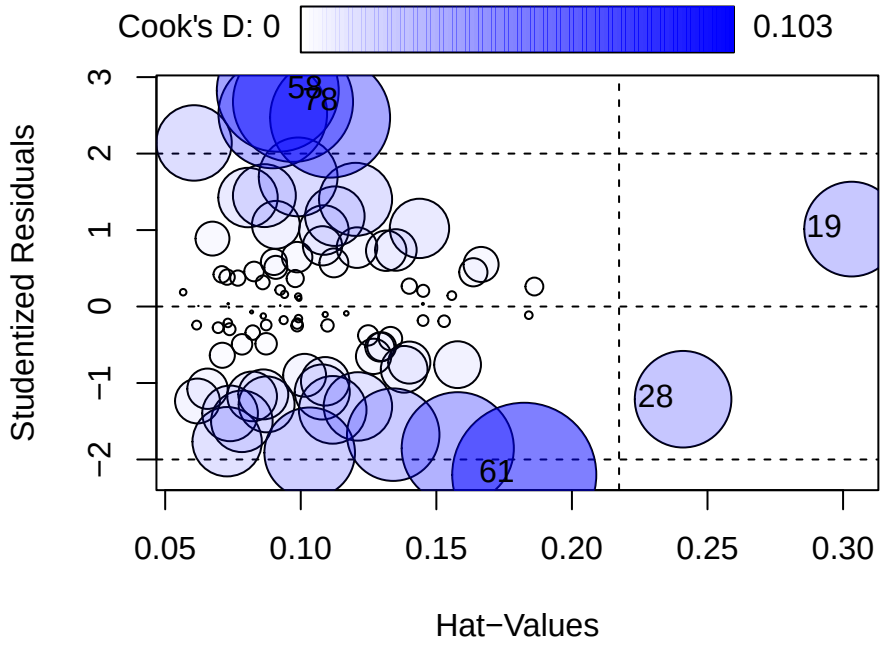

Figura 30. Gráfico de influencia para nuestro modelo modificado(Distancia de cook y leverage).

```
     StudRes        Hat      CookD
19  1.012596 0.30315255 0.04459246
28 -1.211962 0.24099789 0.04637376
58  2.821104 0.09151767 0.07390134
61 -2.200121 0.18241814 0.10316953
78  2.676465 0.09713493 0.07168047
```

Tenemos ahora que las observaciones no poseen en ningún caso una distancia de Cook que nos pueda preocupar. Además, las que poseen un valor más alto tienen un leverage relativamente bajo.

```
Potentially influential observations of
     lm(formula = log(Price) ~ Type + Horsepower + RPM + Wheelbase +      Origin, data = cars3) :

   dfb.1_ dfb.TypL dfb.TypM dfb.TypSm dfb.TypSp dfb.TypV dfb.Hrsp dfb.RPM dfb.Whlb dfb.OUSA dffit
11   0.01    0.01     0.00    -0.01      0.01      0.01    -0.03    -0.01   -0.01    -0.02    -0.05
19   0.28   -0.05    -0.06     0.06     -0.01      0.04     0.56    -0.27   -0.26     0.07     0.67
22   0.20    0.66     0.17    -0.21     -0.07      0.21    -0.19     0.13   -0.24     0.07     0.87
36  -0.07   -0.04    -0.02     0.03      0.03      0.02    -0.03     0.03    0.07     0.03     0.13
51   0.05   -0.23     0.26     0.15      0.02     -0.18     0.09    -0.42    0.08     0.09     0.79
58  -0.09   -0.49    -0.60    -0.29     -0.30     -0.55     0.01    -0.28    0.25    -0.42     0.90
78   0.28    0.02    -0.31    -0.66     -0.55     -0.03     0.02     0.24   -0.33    -0.11     0.88
   cov.r   cook.d hat
11  1.38_*  0.00   0.18
19  1.43_*  0.04   0.30
22  0.61_*  0.07   0.11
36  1.38_*  0.00   0.19
```

```
51  0.58_*  0.06   0.09
58  0.49_*  0.07   0.09
78  0.54_*  0.07   0.10
```

La función influence.measures muestra que esta observación tampoco afecta de forma importante a los coeficientes, ya que los DfBetas no son significativos. Por tanto, podemos conservar esta observación, ya que si bien el criterio utilizado la indica como outlier, no tiene una influencia desmesurada en nuestro modelo por lo que no debería ser un problema.

Finalmente, vamos a analizar la colinealidad:

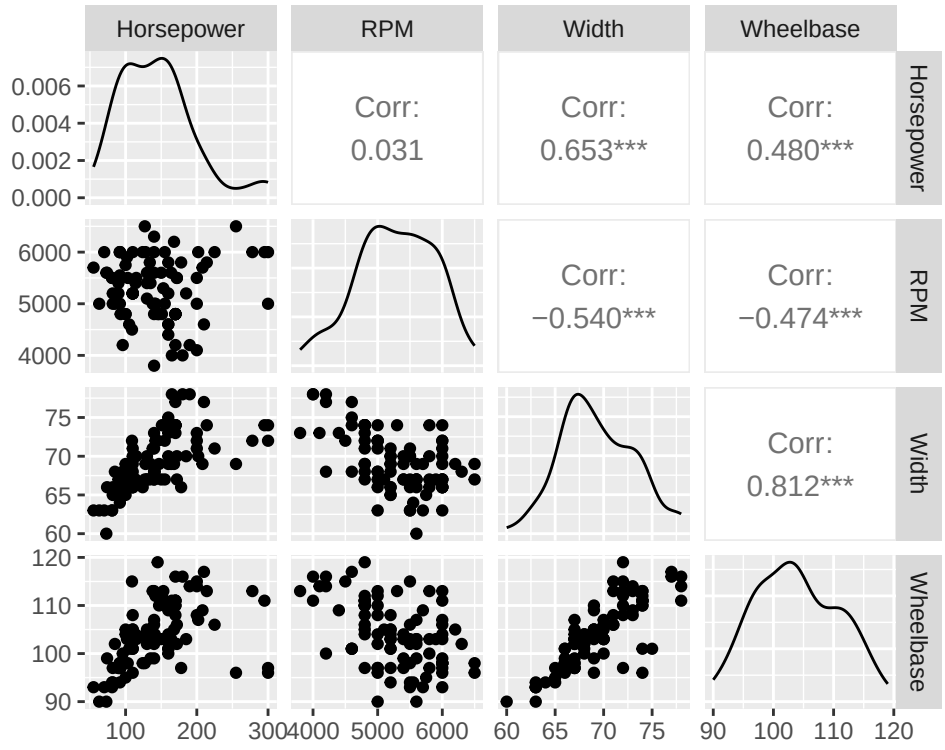

Figura 31. Gráficos por pares para las variables numéricas presentes en nuestro modelo.

```
            Horsepower         RPM      Width  Wheelbase
Horsepower 1.00000000  0.03140701  0.6530179  0.4804696
RPM        0.03140701  1.00000000 -0.5397491 -0.4738036
Width      0.65301787 -0.53974912  1.0000000  0.8117506
Wheelbase  0.48046959 -0.47380364  0.8117506  1.0000000
```

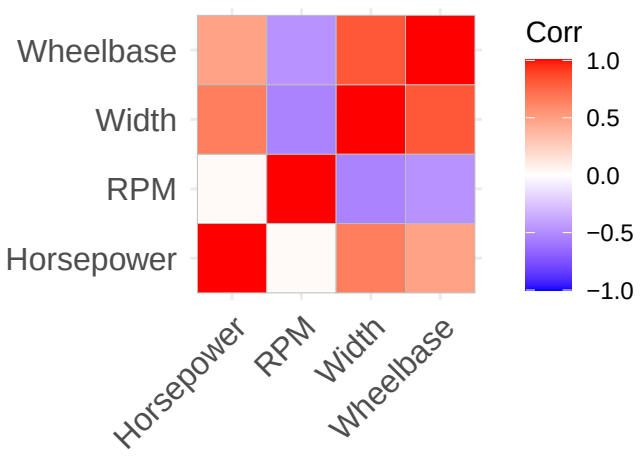

Figura 32. Mapa de correlaciones para las variables numéricas en el modelo modificado.

Obviamente las correlaciones entre las variables se mantienen iguales ya que sólo hemos eliminado una observación de nuestro conjunto de datos.

```
               GVIF Df GVIF^(1/(2*Df))
Type       9.144085  5        1.247711
Horsepower 2.059261  1        1.435013
RPM        2.038264  1        1.427678
Wheelbase  6.098590  1        2.469532
Origin     1.446241  1        1.202598
```

Por otro lado, al eliminar la variable `Width` del modelo, encontramos una mejoría apreciable en VIF. La única con un VIF algo superior a 5 sería `Wheelbase`, pero esto no resulta demasiado preocupante. (`Type` posee algo más, pero tiene 5 grados de libertad).

En definitiva, con las modificaciones del modelo se han podido solucionar los problemas de homocedasticidad, normalidad de los residuos, además de la colinealidad de las variables del modelo. Hay que tener claro que el hecho de quitar el outlier antes se hace para comprobar la validez del modelo y si el resto de datos pueden adecuarse más con este. Estrictamente, en un problema real no podríamos eliminar este outlier así como así en nuestro modelo, ya que podría no ser un dato erróneo.

Apartado 6

Obtén la predicción del precio para un coche en la mediana de los predictores en el modelo escogido. *Notar que las variables categóricas se tratan de diferente manera, no hay mediana.*

Resolución.

Con el modelo adquirido final (llamado `modelo_mod` en el código), podemos predecir ahora el precio de un coche en la situación que se nos pide. Para el caso de las variables de tipo categórico hemos escogido la moda como representante. Recordemos que nuestro modelo es de la forma:

```
Las características del coche para la predicción vienen dadas por:

     Type Horsepower  RPM Length Wheelbase Origin
1 Midsize        140 5200     69       103    USA
```

```
La estimación puntual de Price (en miles de dólares) para un coche con esas características es
 17.54 con un intervalo de predicción: ( 11.75 , 26.182 ).
```