Problemas de Regresión

1. Se plantaron 8 pinos de 0.3 metros de altura en medios controlados y se los sometió a distintas intensidades de irrigación para simular el efecto de las diferentes precipitaciones pluviales. Al acabar el año se midieron las alturas. En la tabla siguiente se muestran las alturas medias (en metros) (y_i) al acabar el año y la cantidad de lluvia (en metros) simulada por cada valor x_i. Suponemos que Y, la altura del árbol al acabar el año, es una variable aleatoria con media $\beta_0 + \beta_1 x$, donde x es la precipitación, y con varianza constante σ^2 por todo valor de x. Hallar las mejores estimaciones lineales sin sesgo de β_0 y β_1 y hallar una estimación sin sesgo de σ^2.

y_i	x_i
0.4826	0.2540
0.5588	0.3556
0.6350	0.4572
0.7874	0.5588
0.8382	0.6604
0.9906	0.7620
1.1176	0.8636
1.1430	0.9652

   a) Estimar los valores b_0 y b_1 para la regresión lineal de la altura del pino en función de la cantidad de lluvia.
   b) Representa gráficamente los datos junto con la recta de regresión.
   c) Hallar un intervalo de confianza al 95% de confianza para los parámetros β_0 y β_1.
   d) Calcular la estimación de la varianza común de los errores de la regresión σ^2.
   e) Hallar el coeficiente de regresión y el coeficiente de regresión ajustado.
   f) Estudiar si el modelo es homocedástico gráficamente y usando el test correspondiente.
   g) Estudiar la normalidad de los residuos.
   h) Estudiar la correlación de los residuos.
   i) Hallar las observaciones "outliers", los "leverages" y las observaciones influyentes.

Solución

En primer lugar definimos las variables x (lluvia) e y (altura):

```
lluvia = c(0.4826, 0.5588, 0.6350, 0.7874, 0.8382, 0.9906, 1.1176, 1.1430)
altura = c(0.2540, 0.3556, 0.4572, 0.5588, 0.6604, 0.7620, 0.8636, 0.9652)
```

   a) Los valores b_0 y b_1 serán:

```
estudio.regresión = lm(altura ~ lluvia)
summary(lm(altura ~ lluvia))
```

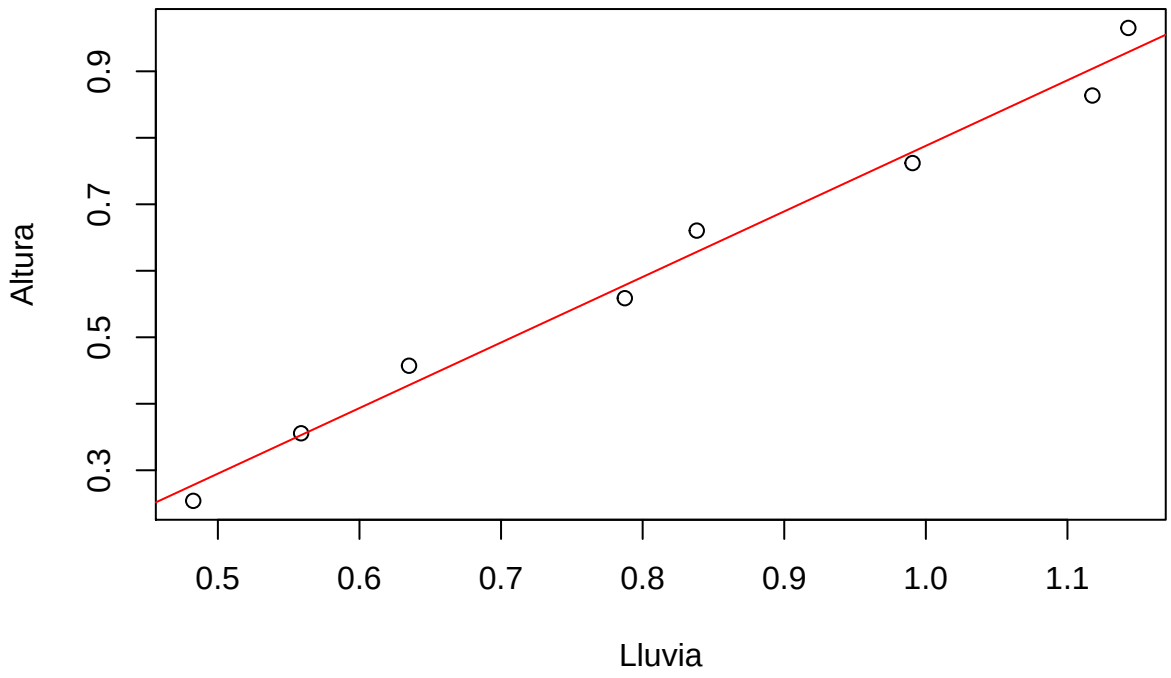
```
##
## Call:
## lm(formula = altura ~ lluvia)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.04029 -0.02056 -0.00697  0.02989  0.03626
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.19813    0.04078  -4.858  0.00283 **
## lluvia       0.98606    0.04786  20.601 8.51e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03174 on 6 degrees of freedom
## Multiple R-squared:  0.9861, Adjusted R-squared:  0.9837
## F-statistic: 424.4 on 1 and 6 DF,  p-value: 8.51e-07
```

El valor de b_0 es $b_0 = -0.1981312$ y el valor de b_1 es $b_1 = 0.9860602$.

b) La representación de los datos junto con la recta de regresión es la siguiente:

```
plot(lluvia,altura,xlab="Lluvia",ylab="Altura")
abline(estudio.regresión,col="red")
```



c) Los intervalos de confianza pedidos son los siguientes:

```
confint(estudio.regresión)
```

```
##                 2.5 %      97.5 %
## (Intercept) -0.2979199 -0.09834245
## lluvia       0.8689423  1.10317806
```

d) La estimación de la varianza común de los errores σ^2 es:

```
errores=estudio.regresión$residuals
n=length(lluvia)
(S2 = sum(errores^2)/(n-2))
```

```
## [1] 0.001007264
```

e) El coeficiente de regresión y el ajustado son los siguientes:

```
(R2 = summary(estudio.regresión)$r.squared)
```

```
## [1] 0.9860602
```

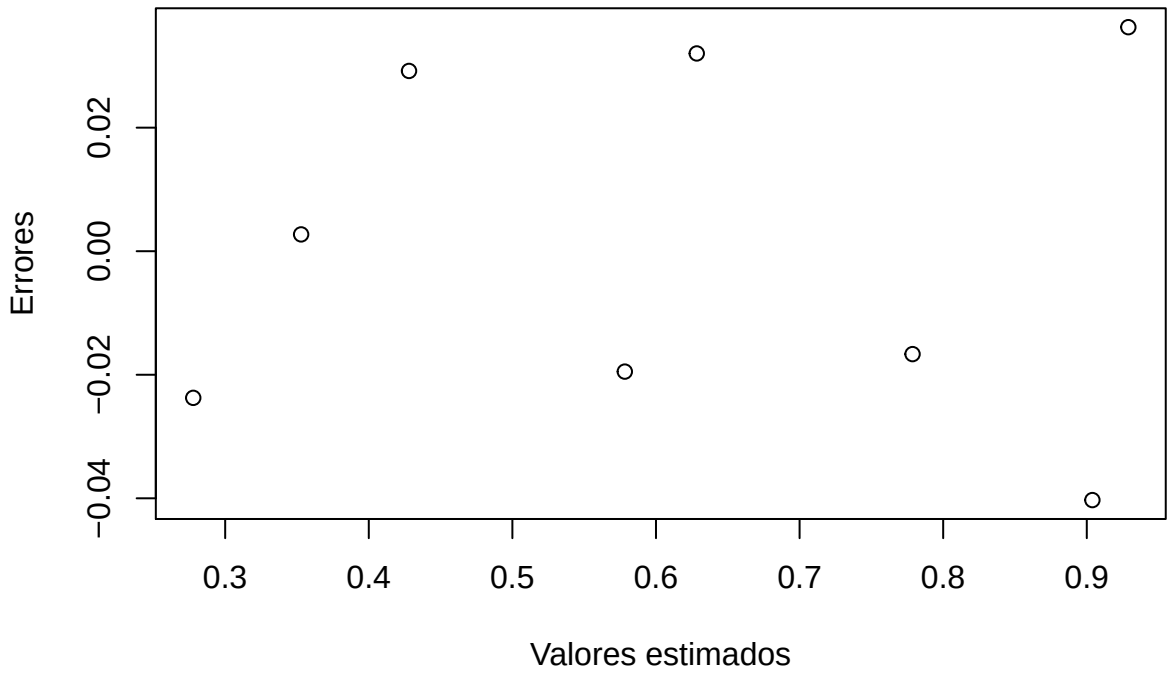
```
(R2.ajustado = summary(estudio.regresión)$adj.r.squared)
```

```
## [1] 0.9837369
```

Podemos observar que el ajuste es bastante bueno.

f) Para ver si el modelo es homocedástico hay que realizar el gráfico de los errores en función de los valores estimados y ver si dicho gráfico se parece a un "cielo estrellado":

```
plot(estudio.regresión$fitted.values,estudio.regresión$residuals,xlab="Valores estimados",ylab="Errores")
```



En principio, no se observa ningún patrón. Apliquemos el test de White para comprobar la homocedasticidad:

```r
library(lmtest)
bptest(estudio.regresión,~ lluvia+I(lluvia^2))
```

```
##
##  studentized Breusch-Pagan test
##
## data:  estudio.regresión
## BP = 3.8788, df = 2, p-value = 0.1438
```

Como el valor es bastante grande, no tenemos indicios para rechazar la homocedasticidad de los residuos.

g) Para estudiar la normalidad de los residuos, apliquemos el test de Shapiro-Wilks:

```r
shapiro.test(estudio.regresión$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  estudio.regresión$residuals
## W = 0.89569, p-value = 0.2641
```

El p-valor es bastante grande, por tanto no tenemos evidencias para rechazar la normalidad de los residuos.

h) Para estudiar la correlación de los residuos, apliquemos el test de Durbin-Watson:

```
dwtest(estudio.regresión,alternative='greater')
```

```
##
##  Durbin-Watson test
##
## data:  estudio.regresión
## DW = 2.5169, p-value = 0.6254
## alternative hypothesis: true autocorrelation is greater than 0
```

```
dwtest(estudio.regresión,alternative='less')
```

```
##
##  Durbin-Watson test
##
## data:  estudio.regresión
## DW = 2.5169, p-value = 0.3746
## alternative hypothesis: true autocorrelation is less than 0
```

Como los p-valores son grandes, no tenemos evidencias suficientes para rechazar que no haya autocorrelación entre los errores. Es decir, concluimos que no hay ni autocorrelación positiva ni negativa.

i) Miremos si hay outliers en nuestra tabla de datos:

```
library(car)
```

```
## Warning: package 'car' was built under R version 3.6.3
```

```
## Loading required package: carData
```

```
outlierTest(estudio.regresión)
```

```
## No Studentized residuals with Bonferroni p < 0.05
## Largest |rstudent|:
##    rstudent unadjusted p-value Bonferroni p
## 7 -1.823628            0.12781           NA
```

La única observación candidata a outlier es la número 7 pero el p-valor nos dice que de hecho no lo es.

Hallemos los posibles "leverages":

```
(valores.hat = hatvalues(estudio.regresión))
```

```
##         1         2         3         4         5         6         7         8
## 0.3826119 0.2791636 0.2021277 0.1272927 0.1258254 0.1918562 0.3275862 0.3635363
```

```
which(valores.hat > 2*2/n)
```

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

No hay observaciones "leverages".

Por último, estudiemos si hay observaciones influyentes:

```
(distancias.cook=cooks.distance(estudio.regresión))
```

```
##           1           2           3           4           5           6
## 0.280855743 0.001974211 0.134228571 0.031523348 0.083776214 0.040473998
##           7           8
## 0.583802949 0.585853116
```

```
which(distancias.cook > 4/(n-2))
```

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

Tampoco tenemos observaciones influyentes.

2. Los siguientes datos relacionan la producción de biomasa de soja con la radiación solar interceptada acumulada durante un período de ocho semanas después de la emergencia. La producción de biomasa es el peso seco medio en gramos de muestras independientes de cuatro plantas.

X (Radiación solar)	Y Biomasa de la planta
29.7	16.6
68.4	49.1
120.7	121.7
217.2	219.6
313.5	375.5
419.1	570.8
535.9	648.2
641.5	755.6

a) Estimar los valores b_0 y b_1 para la regresión lineal de la biomasa de la planta en función de la radiación solar.
b) Representa gráficamente los datos junto con la recta de regresión.
c) Hallar un intervalo de confianza al 95% de confianza para los parámetros β_0 y β_1.
d) Calcular la estimación de la varianza común de los errores de la regresión σ^2.
e) Hallar el coeficiente de regresión y el coeficiente de regresión ajustado.
f) Estudiar si el modelo es homocedástico gráficamente y usando el test correspondiente.
g) Estudiar la normalidad de los residuos.
h) Estudiar la correlación de los residuos.
i) Hallar las observaciones "outliers", los "leverages" y las observaciones influyentes.

Solución

En primer lugar definimos las variables x (radiación) e y (biomasa):

```
radiación = c(29.7, 68.4, 120.7, 217.2, 313.5, 419.1, 535.9, 641.5)
biomasa = c(16.6, 49.1, 121.7, 219.6, 375.5, 570.8, 648.2, 755.6)
```

a) Los valores b_0 y b_1 serán:

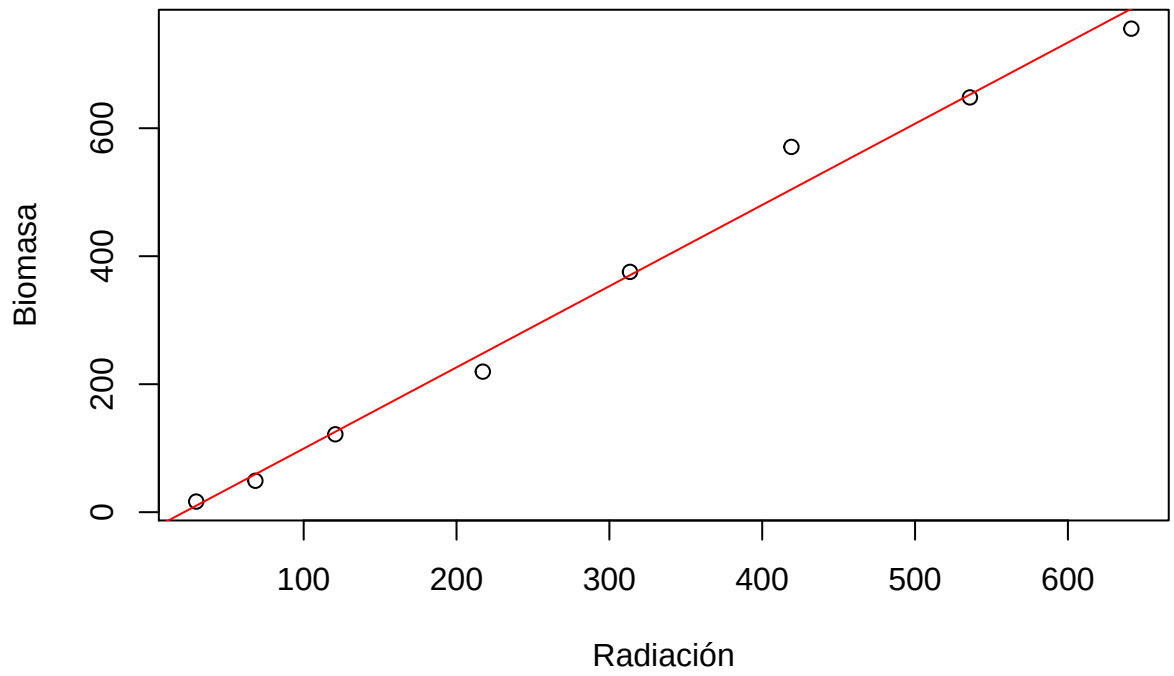
```
estudio.regresión = lm(biomasa ~ radiación)
summary(lm(biomasa ~ radiación))
```

```
##
## Call:
## lm(formula = biomasa ~ radiación)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -31.052 -14.738  -4.175   5.488  66.428
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -27.56895   19.84220  -1.389    0.214
## radiación     1.26925    0.05503  23.063 4.36e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 32.65 on 6 degrees of freedom
## Multiple R-squared:  0.9888, Adjusted R-squared:  0.987
## F-statistic: 531.9 on 1 and 6 DF,  p-value: 4.355e-07
```

El valor de b_0 es $b_0 = -27.5689519$ y el valor de b_1 es $b_1 = 1.2692462$.

b) La representación de los datos junto con la recta de regresión es la siguiente:

```
plot(radiación,biomasa,xlab="Radiación",ylab="Biomasa")
abline(estudio.regresión,col="red")
```



c) Los intervalos de confianza pedidos son los siguientes:

```
confint(estudio.regresión)
```

```
##                   2.5 %    97.5 %
## (Intercept) -76.121061 20.983157
## radiación     1.134583  1.403909
```

d) La estimación de la varianza común de los errores σ^2 es:

```
errores=estudio.regresión$residuals
n=length(lluvia)
(S2 = sum(errores^2)/(n-2))
```

```
## [1] 1066.047
```

e) El coeficiente de regresión y el ajustado son los siguientes:

```
(R2 = summary(estudio.regresión)$r.squared)
```

```
## [1] 0.9888456
```

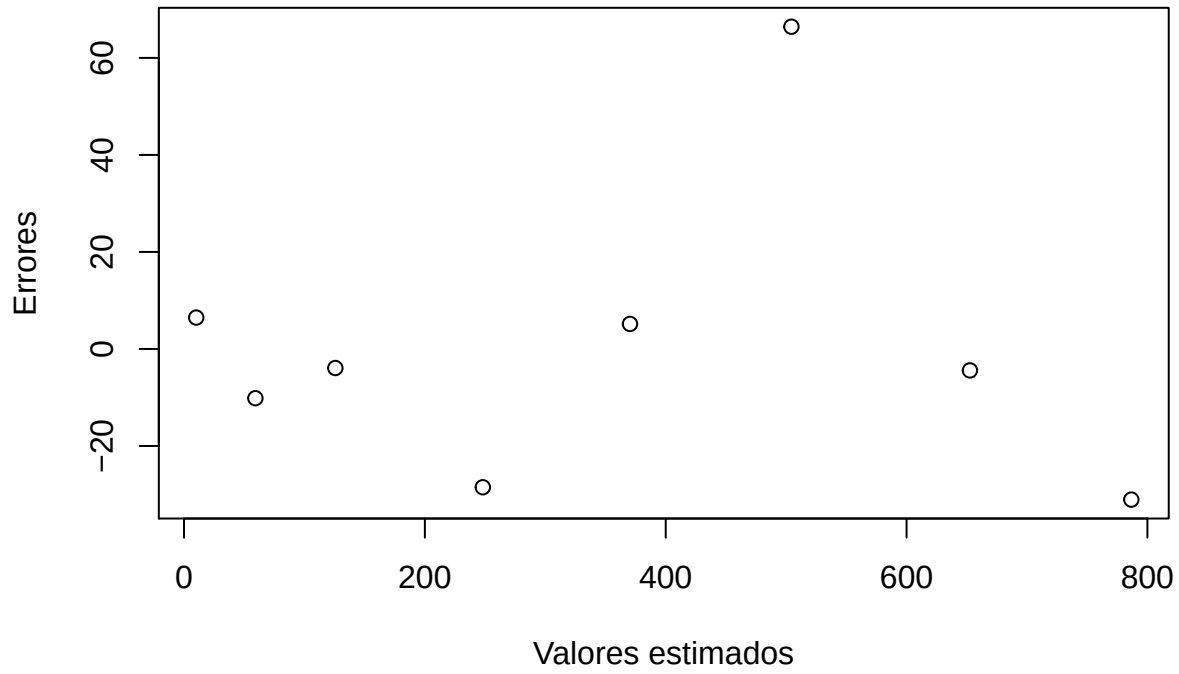
```r
(R2.ajustado = summary(estudio.regresión)$adj.r.squared)
```

```
## [1] 0.9869865
```

Podemos observar que el ajuste es bastante bueno.

f) Para ver si el modelo es homocedástico hay que realizar el gráfico de los errores en función de los valores estimados y ver si dicho gráfico se parece a un "cielo estrellado":

```
plot(estudio.regresión$fitted.values,estudio.regresión$residuals,xlab="Valores estimados",ylab="Errores")
```



Se observa una especie de cuña para los valores de radiación estimados entre 200 y 800 pero como son pocos datos, no se puede concluir nada seguro. Apliquemos el test de White para comprobar la homocedasticidad:

```
library(lmtest)
bptest(estudio.regresión,~ radiación+I(radiación^2))
```

```
##
##  studentized Breusch-Pagan test
##
## data:  estudio.regresión
## BP = 1.8186, df = 2, p-value = 0.4028
```

Como el valor es bastante grande, no tenemos indicios para rechazar la homocedasticidad de los residuos. Lo que observamos antes, se debía a la aleatoriedad.

g) Para estudiar la normalidad de los residuos, apliquemos el test de Shapiro-Wilks:

```
shapiro.test(estudio.regresión$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  estudio.regresión$residuals
## W = 0.82756, p-value = 0.056
```

El p-valor está en la zona de penumbra, por tanto no podemos tomar una decisión clara sobre la normalidad de los residuos.

h) Para estudiar la correlación de los residuos, apliquemos el test de Durbin-Watson:

```
dwtest(estudio.regresión,alternative='greater')
```

```
##
##  Durbin-Watson test
##
## data:  estudio.regresión
## DW = 1.8035, p-value = 0.2058
## alternative hypothesis: true autocorrelation is greater than 0
```

```
dwtest(estudio.regresión,alternative='less')
```

```
##
##  Durbin-Watson test
##
## data:  estudio.regresión
## DW = 1.8035, p-value = 0.7942
## alternative hypothesis: true autocorrelation is less than 0
```

Como los p-valores son grandes, no tenemos evidencias suficientes para rechazar que no haya autocorrelación entre los errores. Es decir, concluimos que no hay ni autocorrelación positiva ni negativa.

i) Miremos si hay outliers en nuestra tabla de datos:

```
library(car)
outlierTest(estudio.regresión)
```

```
##   rstudent unadjusted p-value Bonferroni p
## 6 4.961544          0.0042425      0.03394
```

La única observación candidata a outlier es la número 6 y los p-valores obtenidos la confirman como outlier.

Hallemos los posibles "leverages":

```
(valores.hat = hatvalues(estudio.regresión))
```

```
##         1         2         3         4         5         6         7         8
## 0.3223376 0.2686381 0.2095890 0.1414317 0.1261650 0.1699977 0.2922801 0.4695609
```

```
which(valores.hat > 2*2/n)
```

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

No hay observaciones "leverages".

Por último, estudiemos si hay observaciones influyentes:

```
(distancias.cook=cooks.distance(estudio.regresión))
```

```
##           1           2           3           4           5           6
## 0.013791149 0.024255699 0.002429041 0.073151901 0.002063563 0.510715182
##           7           8
## 0.005347283 0.754757802
```

```r
which(distancias.cook > 4/(n-2))
```

```
## 8
## 8
```

Vemos que la observación número 8 es una observación influyente.

3. Se probó un modelo de simulación para el flujo máximo de agua de las cuencas hidrográficas comparando el flujo máximo medido de 10 tormentas con predicciones del flujo máximo obtenido del modelo de simulación. Q_o y Q_p son los flujos máximos observados y pronosticados, respectivamente. Se registraron cuatro variables independientes:

   - X_1: area de la cuenca (m^2),
   - X_2: pendiente promedio de la cuenca (en porcentaje),
   - X_3: índice de absorbencia superficial (0 = absorbencia completa, 100 = sin absorbencia), y
   - X_4: intensidad de pico de lluvia calculada en intervalos de media hora.

Q_o	Q_p	X_1	X_2	X_3	X_4
28	32	.03	3.0	70	.6
112	142	.03	3.0	80	1.8
398	502	.13	6.5	65	2.0
772	790	1.00	15.0	60	.4
2294	3075	1.00	15.0	65	2.3
2484	3230	3.00	7.0	67	1.0
2586	3535	5.00	6.0	62	.9
3024	4265	7.00	6.5	56	1.1
4179	6529	7.00	6.5	56	1.4
710	935	7.00	6.5	56	.7

   Consideramos $Y = \ln\left(\frac{Q_o}{Q_p}\right)$ como variable dependiente, consideramos la regresión de Y como función de X_1, X_2, X_3 y X_4. Se pide:

a) Estimar los valores b_0, b_1, b_2, b_3, b_4 para la regresión lineal de Y en función de X_i, $i = 1, 2, 3, 4$.
b) Hallar un intervalo de confianza al 95% de confianza para los parámetros β_i, $i = 0, 1, 2, 3, 4$.
c) Calcular la estimación de la varianza común de los errores de la regresión σ^2.
d) Hallar el coeficiente de regresión y el coeficiente de regresión ajustado.
e) Estudiar si el modelo es homocedástico gráficamente y usando el test correspondiente.
f) Estudiar la normalidad de los residuos.
g) Estudiar la correlación de los residuos.
h) Contrastar la linealidad y la aditividad del modelo.
i) Hallar las observaciones "outliers", los "leverages" y las observaciones influyentes.

Solución

En primer lugar definimos las variables X_1, X_2, X_3, X_4 e Y:

```
X1=c(.03, .03, .13, 1.00, 1.00, 3.00, 5.00, 7.00, 7.00, 7.00)
X2=c(3.0, 3.0, 6.5, 15.0, 15.0, 7.0, 6.0, 6.5, 6.5, 6.5)
X3=c(70, 80, 65, 60, 65, 67, 62, 56, 56, 56)
X4=c(.6, 1.8, 2.0, .4, 2.3, 1.0, .9, 1.1, 1.4, .7)
Qo =c(28, 112, 398, 772, 2294, 2484, 2586, 3024, 4179, 710)
Qp =c(32, 142, 502, 790, 3075, 3230, 3535, 4265, 6529, 935)
(Y=log(Qo/Qp))
```

```
##  [1] -0.1335314 -0.2373282 -0.2321481 -0.0230484 -0.2930079 -0.2626120
##  [7] -0.3126010 -0.3438617 -0.4461818 -0.2752816
```

a) Los valores b_i, $i = 0, 1, 2, 3, 4$ serán:

```
estudio.regresión = lm(Y ~ X1+X2+X3+X4)
summary(lm(Y ~ X1+X2+X3+X4))
```

```
##
## Call:
## lm(formula = Y ~ X1 + X2 + X3 + X4)
##
## Residuals:
##       1       2       3       4       5       6       7       8
## -0.02800  0.03143  0.01892  0.01694 -0.01530 -0.02860 -0.02467  0.02501
##       9      10
## -0.04129  0.04556
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.087321   0.311931   0.280  0.79074
## X1          -0.035384   0.009336  -3.790  0.01276 *
## X2           0.004726   0.004765   0.992  0.36677
## X3          -0.001913   0.004189  -0.457  0.66702
## X4          -0.120080   0.023810  -5.043  0.00396 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.04119 on 5 degrees of freedom
## Multiple R-squared:  0.9287, Adjusted R-squared:  0.8717
## F-statistic: 16.29 on 4 and 5 DF,  p-value: 0.004502
```

El valor del vector $(b_0, b_1, b_2, b_3, b_4)^\top$ es el siguiente:

```
estudio.regresión$coefficients
```

```
##  (Intercept)           X1           X2           X3           X4
##  0.087321331 -0.035384099  0.004726196 -0.001913147 -0.120079975
```

b) Los intervalos de confianza pedidos son los siguientes:

```
confint(estudio.regresión)
```

```
##                    2.5 %      97.5 %
## (Intercept) -0.714523782  0.88916644
## X1          -0.059384200 -0.01138400
## X2          -0.007521599  0.01697399
## X3          -0.012680693  0.00885440
## X4          -0.181284500 -0.05887545
```

c) La estimación de la varianza común de los errores σ^2 es:

```
errores=estudio.regresión$residuals
n=length(X1)
k=4
(S2 = sum(errores^2)/(n-k-1))
```

```
## [1] 0.001696714
```

d) El coeficiente de regresión y el ajustado son los siguientes:

```
(R2 = summary(estudio.regresión)$r.squared)
```

```
## [1] 0.9287455
```

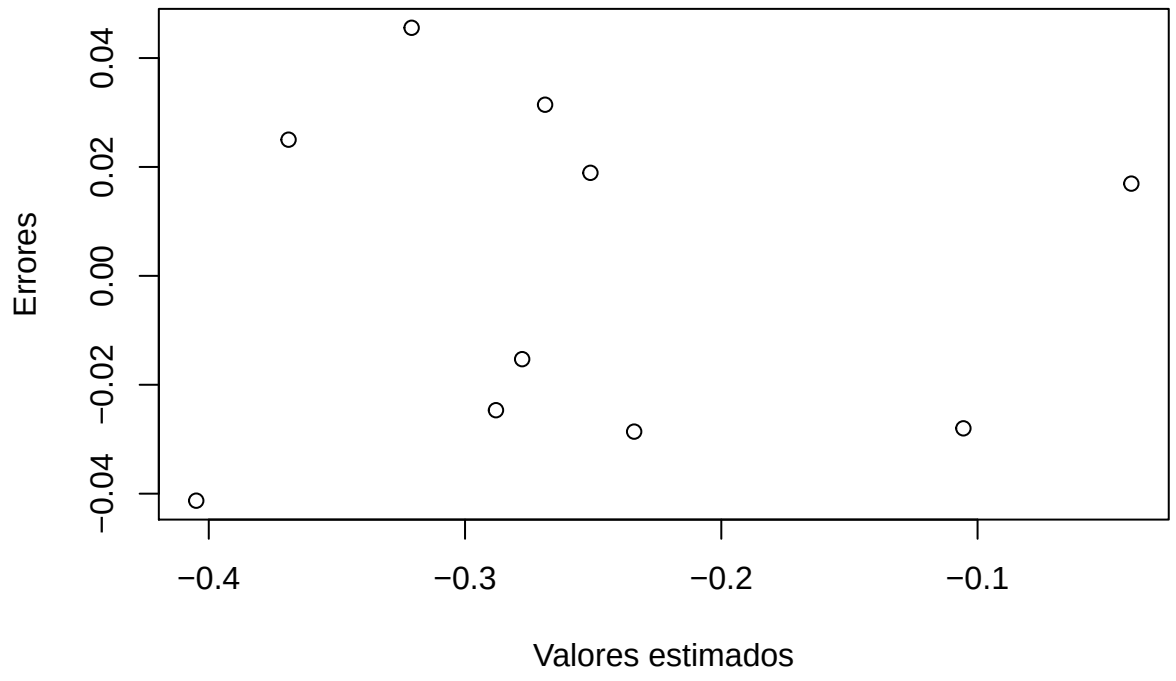
```
(R2.ajustado = summary(estudio.regresión)$adj.r.squared)
```

```
## [1] 0.8717419
```

Podemos observar que el ajuste es bastante bueno.

e) Para ver si el modelo es homocedástico hay que realizar el gráfico de los errores en función de los valores estimados y ver si dicho gráfico se parece a un "cielo estrellado":

```
plot(estudio.regresión$fitted.values,estudio.regresión$residuals,xlab="Valores estimados",ylab="Errores")
```



No observamos ningún patrón visible. Apliquemos el test de White para comprobar la homocedasticidad:

```
library(lmtest)
X=cbind(X1,X2,X3,X4)
bptest(estudio.regresión,~ X+I(X^2))
```

```
##
##  studentized Breusch-Pagan test
##
## data:  estudio.regresión
## BP = 7.3259, df = 8, p-value = 0.5019
```

Como el valor es bastante grande, no tenemos indicios para rechazar la homocedasticidad de los residuos. Lo que observamos antes, se debía a la aleatoriedad.

f) Para estudiar la normalidad de los residuos, apliquemos el test de Shapiro-Wilks:

```
shapiro.test(estudio.regresión$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  estudio.regresión$residuals
## W = 0.90416, p-value = 0.2432
```

El p-valor es bastante grande, por tanto no tenemos evidencias para rechazar la normalidad de los residuos.

g) Para estudiar la correlación de los residuos, apliquemos el test de Durbin-Watson:

```
dwtest(estudio.regresión,alternative='greater')
```

```
##
##  Durbin-Watson test
##
## data:  estudio.regresión
## DW = 2.2785, p-value = 0.5415
## alternative hypothesis: true autocorrelation is greater than 0
```

```
dwtest(estudio.regresión,alternative='less')
```

```
##
##  Durbin-Watson test
##
## data:  estudio.regresión
## DW = 2.2785, p-value = 0.4585
## alternative hypothesis: true autocorrelation is less than 0
```

Como los p-valores son grandes, no tenemos evidencias suficientes para rechazar que no haya autocorrelación entre los errores. Es decir, concluimos que no hay ni autocorrelación positiva ni negativa.

h) Para estudiar la aditividad, usamos el test de Tukey:

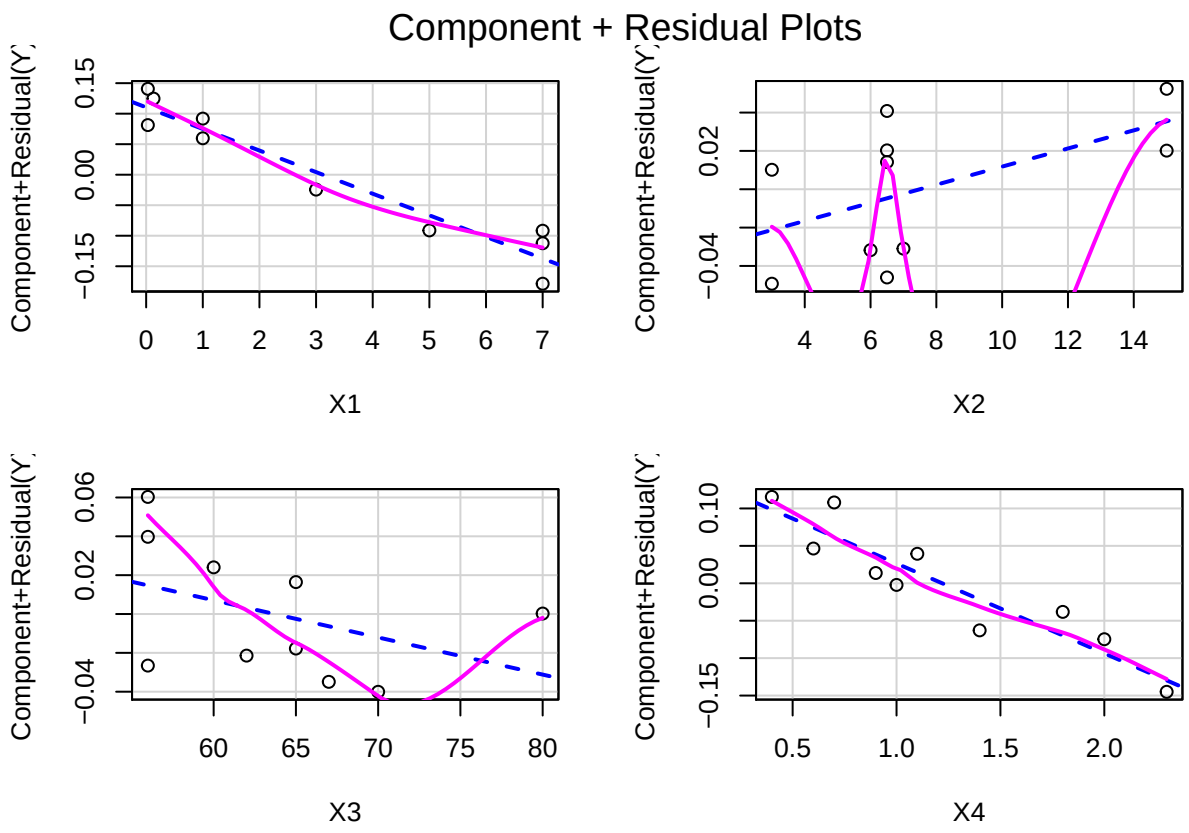
```
residualPlots(estudio.regresión,plot=FALSE)
```

```
##            Test stat Pr(>|Test stat|)
## X1            1.2725           0.2721
## X2            0.1202           0.9101
## X3            1.7156           0.1614
## X4            0.6508           0.5507
## Tukey test   -0.2150           0.8298
```

Como los p-valores son grandes, no tenemos evidencias para rechazar la aditividad del modelo.

Para estudiar la linealidad, realizamos los gráficos de residuos parciales:

```
crPlots(estudio.regresión)
```



Observamos que la variable que se ajusta menos a la linealidad es la X_2. También observamos que en la X_3 tenemos indicios de no linealidad. Todas las demás presentan un ajuste bastante aceptable al modelo lineal.

i) Miremos si hay outliers en nuestra tabla de datos:

```
outlierTest(estudio.regresión)
```

```
## No Studentized residuals with Bonferroni p < 0.05
## Largest |rstudent|:
##   rstudent unadjusted p-value Bonferroni p
## 2 1.939199            0.12449           NA
```

La única observación candidata a outlier es la número 2 pero el p-valor obtenido la descarta como outlier.

Hallemos los posibles "leverages":

```
(valores.hat = hatvalues(estudio.regresión))
```

```
##         1         2         3         4         5         6         7         8
## 0.5995377 0.7597133 0.7940579 0.7859826 0.7344332 0.2130011 0.1781053 0.2871882
##         9        10
## 0.3470313 0.3009494
```

```
which(valores.hat > 2*(k+1)/n)
```

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

No hay observaciones "leverages".

Por último, estudiemos si hay observaciones influyentes:

```
(distancias.cook=cooks.distance(estudio.regresión))
```

```
##          1          2          3          4          5          6          7
## 0.34552452 1.53205840 0.79038857 0.58059899 0.28733341 0.03316623 0.01891793
##          8          9         10
## 0.04167251 0.16353949 0.15066820
```

```
which(distancias.cook > 4/(n-k-1))
```

```
## 2
## 2
```

Vemos que la observación número 2 es una observación influyente.