Laboratorio 6

Bioestadistica 2020

Regresión lineal

A modo de ejemplo utilizaremos el dataset llamado *airquality* que viene ya en *R*.

Este dataset consiste en mediciones de 4 variables atmosfericas.

```
?airquality
```

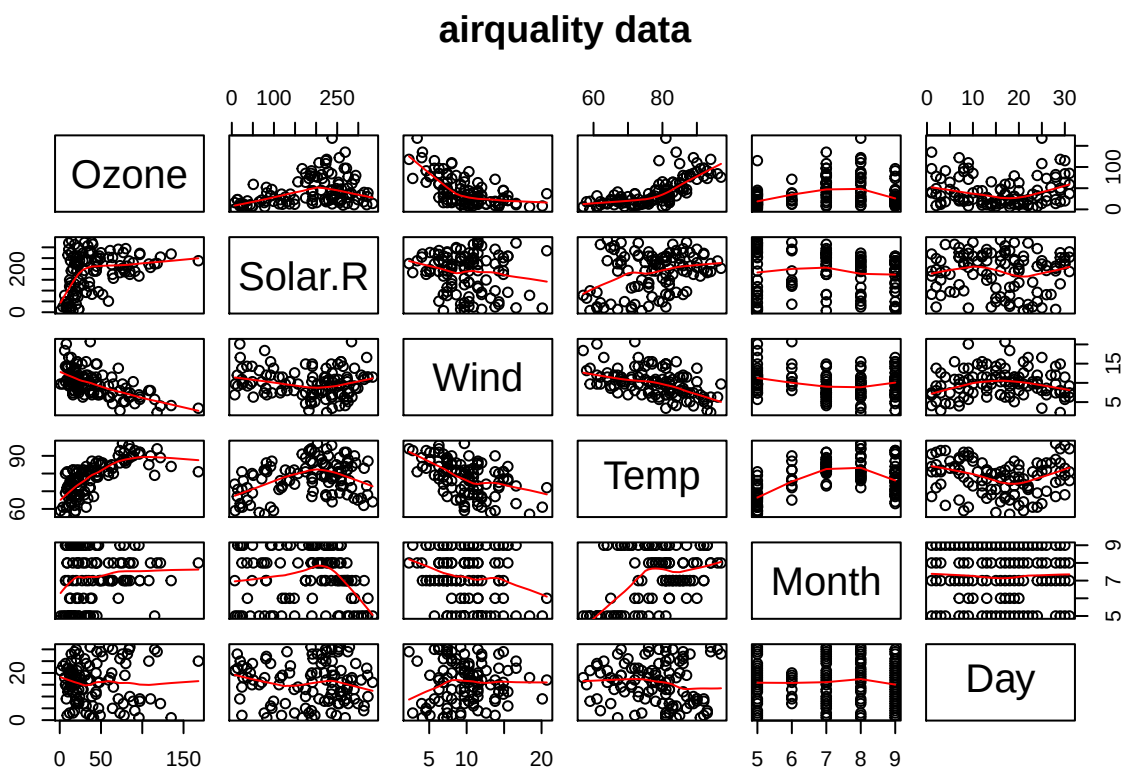
```
## starting httpd help server ... done
```

Una cosa importante al momento de trabajar con nuestros datos, es en el caso de que haya filas con datos faltantes, los cuales aparecen como NA en la tabla, lo que podemos hacer es quedarnos solo con las filas que no tienen datos faltantes.

```
datos=airquality[complete.cases(airquality),]
```

Una manera de visualizar los datos de a pares para tener una idea sobre en que situaciones se podría ajustar un modelo lineal es mediante el siguiente,

```
pairs(datos, panel=panel.smooth, main='airquality data')
```



Una vez visualizados los datos, nos preguntamos si hay una dependencia lineal del viento con respecto a la temperatura, para lo cual hacemos una regresión lineal con el comando *lm()*

```
modeloRegresion = lm(Wind ~ Temp,datos)
```

Un resumen de los resultados puede verse aplicando la función *summary* al modelo ajustado.

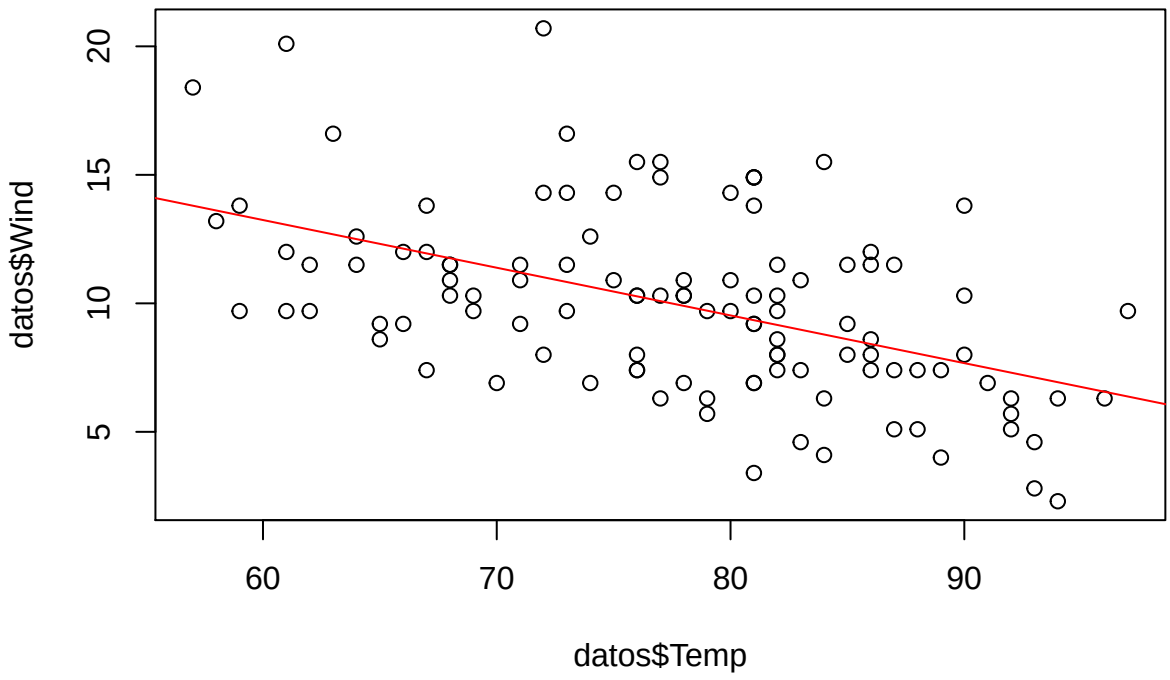
```
summary(modeloRegresion)
```

```
##
## Call:
## lm(formula = Wind ~ Temp, data = datos)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.9443 -2.3584 -0.3005  1.6136  9.6852
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 24.37877    2.43137  10.027  < 2e-16 ***
## Temp        -0.18561    0.03102  -5.983 2.84e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.101 on 109 degrees of freedom
## Multiple R-squared:  0.2472, Adjusted R-squared:  0.2403
## F-statistic: 35.79 on 1 and 109 DF,  p-value: 2.842e-08
```

Esto devuelve varios valores de interés. Lo primero es el summary de los residuos $\hat{Y}_i - Y_i$ (los cuartiles). Observar que el valor esperado de los residuos es 0 por lo tanto la mediana debería dar próximo a 0 si se cumplen las hipótesis de nuestro modelo. Los valores que se obtienen de $\hat{a}_n$ y $\hat{b_n}$ están en la columna Intercept y son -0.18561 y 24.37877 respectivamente. Esto significa que la pendiente de la recta de regresión es negativa. Es decir un aumento de la temperatura disminuye el valor medio del viento ya que el valor. Además se sigue que aumentar en una unidad la temperatura (en este caso 1 grado Farenheit) produce una diminución de 0.18561 millas por hora, en la velocidad media de los vientos. Por otro lado el valor 24,378 es el coeficiente $\hat{b_n}$ que nos da el valor de la intersección de la recta en 0 (nos estaría diciendo que nuestro modelo, no el real sino el que hallamos, estima la velocidad media de los vientos en 24.37877 si la temperatura es 0). Los valores 2.43137 y 0.03102 son estimaciones de la raiz de la varianza de $\hat{b_n}$ y $\hat{a_n}$ respectivamente. La última columna da el p-valor para la pruebas $H_0 \ : \ b_n = 0$ y para la prueba $H_0 \ : \ a_n = 0$. El primero de dichos valores da casi 0, con lo cual rechazamos a nivel por ejemplo 0,05 la hipótesis de que $b_n = 0$ (y de hecho se rechaza a casi cualquier nivel razonable). El otro p valor es pequeño también, por lo tanto también rechazamos $H_0 \ : \ a_n = 0$, esto último nos dice que hay una relación lineal entre las variables viento y temperatura. Finalmente el valor que obtenemos para el R^2 es el que se denomina Multiple R-squared y da 0.2472. Esto nos dice que la variación explicada por el modelo es aproximadamente 1/4 de la variación total, con lo cual el ajuste no es muy bueno.

Por ultimo podemos graficar nuestros datos y el modelo de regresión, mediante

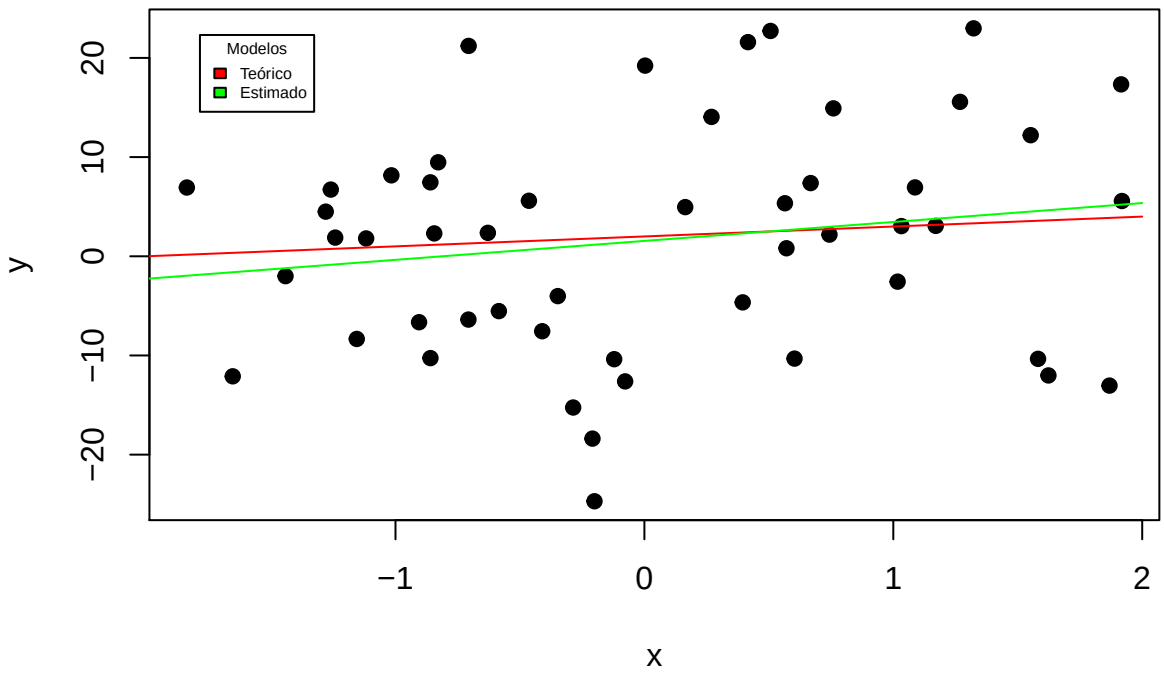
```
plot(datos$Temp,datos$Wind)
f=function(x){return(modeloRegresion$coefficients[2]*x+modeloRegresion$coefficients[1])}
lines(55:100,f(55:100),col='red')
```



Veamos ciertos casos especiales para ver que podría pasar con los modelos lineales.

Comencemos estudiando quÃ© sucede si tenemos pocos datos y σ es muy grande como en **??**

```
n = 50
x = runif(n, -2, 2)
a = 1
b = 2
e = rnorm(n, 0, 10)
y = a*x + b + e
plot(x, y, pch = 19)
eje_x = seq(-2,2, length.out = 10000)
lines(eje_x, a*eje_x + b, col = "red")
regresion = lm(y ~ x)
a_hat = regresion$coefficients[2]
b_hat = regresion$coefficients[1]
lines(eje_x, a_hat*eje_x + b_hat, col = "green")
legend("topleft", inset=.05, title="Modelos", c("Teórico", "Estimado"),
       fill= c("red", "green"), cex = 0.5)
```




```
summary(regresion)
```

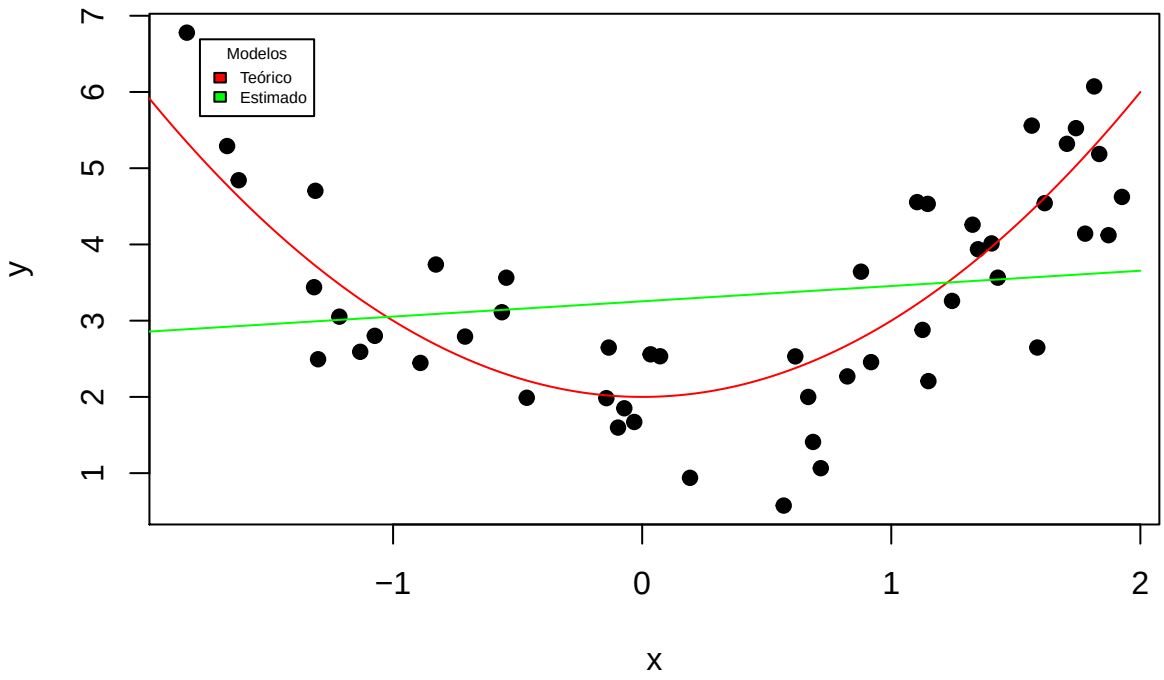
```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -25.865  -7.497   1.188   7.669  21.010
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.549      1.611   0.961    0.341
## x              1.908      1.574   1.212    0.231
##
## Residual standard error: 11.39 on 48 degrees of freedom
## Multiple R-squared:  0.0297, Adjusted R-squared:  0.009482
## F-statistic: 1.469 on 1 and 48 DF,  p-value: 0.2314
```

Está claro que, si disponemos de pocos datos, o si las X_i toman valores cercanos, un valor grande de σ puede oscurecer la relación lineal existente entre las X_i y las Y_i.

Veamos qué sucede si la relación entre X_i y Y_i no es lineal.

```
n = 50
x = runif(n, -2, 2)
a = 1
b = 2
```

```r
e = rnorm(n, 0, 1)
y = a*x^2 + b + e
plot(x, y, pch = 19)
eje_x = seq(-2,2, length.out = 10000)
lines(eje_x, a*eje_x^2 + b, col = "red")
regnolin = lm(y ~ x)
a_hat = regnolin$coefficients[2]
b_hat = regnolin$coefficients[1]
lines(eje_x, a_hat*eje_x + b_hat, col = "green")
legend("topleft", inset=.05, title="Modelos",c("Teórico","Estimado"),
       fill= c("red", "green"), cex = 0.5)
```




```r
summary(regnolin)
```

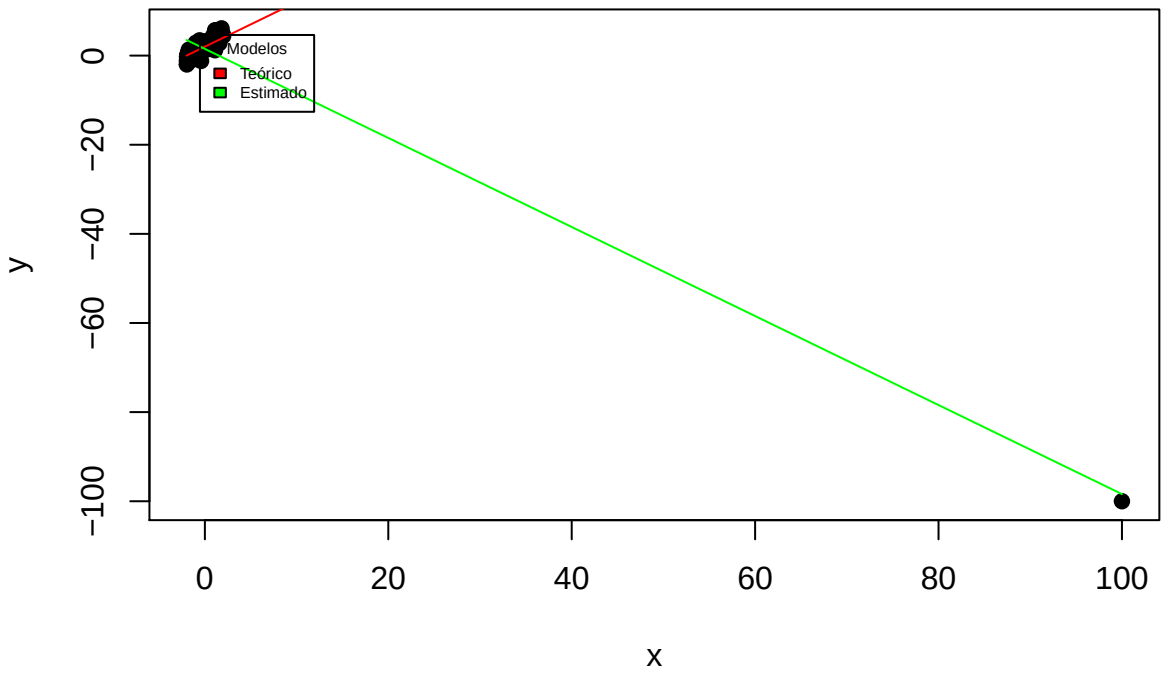
```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.7935 -0.9678 -0.1342  0.9074  3.8904
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.2541     0.2097  15.515   <2e-16 ***
## x             0.2005     0.1773   1.131    0.264
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.413 on 48 degrees of freedom
## Multiple R-squared:  0.02594,    Adjusted R-squared:  0.005646
## F-statistic: 1.278 on 1 and 48 DF,  p-value: 0.2638
```

En teoría, para el rango en el que se encuentran las X_i y su relación cuadrática con las Y_i, debería suceder que $\hat{a} = 0$. Sin embargo, esto va a depender de dónde se encuentren las X_i, y de los valores de e_i.

Veamos ahora el caso de los *outliers*, o sea, datos atípicos o erroneos en nuestra base de datos.

```
n = 50
x = runif(n, -2, 2)
a = 1
b = 2
e = rnorm(n, 0, 1)
y = a*x + b + e
x[1] = 100
y[1] = -100 # El primer par (X_i, Y_i) es un outlier y no respeta el modelo lineal
plot(x, y, pch = 19)
eje_x = seq(-2,100, length.out = 10000)
lines(eje_x, a*eje_x + b, col = "red")
regconoutlier= lm(y ~ x)
a_hat = regconoutlier$coefficients[2]
b_hat = regconoutlier$coefficients[1]
lines(eje_x, a_hat*eje_x + b_hat, col = "green")
legend("topleft",inset=.05,title="Modelos",c("Teórico","Estimado"),
       fill= c("red", "green"), cex = 0.5)
```




```
summary(regconoutlier)
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.4605 -2.4334 -0.3393  2.1843  6.4247
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.50600    0.42848   3.515 0.000971 ***
## x           -0.99903    0.03019 -33.094  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.007 on 48 degrees of freedom
## Multiple R-squared:  0.958,  Adjusted R-squared:  0.9571
## F-statistic:  1095 on 1 and 48 DF,  p-value: < 2.2e-16
```

Podemos ver que el peso en el modelo del outlier es mucho mayor al del resto de los datos. Esto se debe a que, en la ecuación, el sumando del error cuadrático del outlier es mucho más sensible a las elecciones de $\hat{a}$ y $\hat{b}$ que el resto de los sumandos.

Ejercicio 1

Importar la base de datos del laboratorio 4 e investigar que variables se ajustan a un modelo de regresión lineal.

Predicción

Una vez confecciónado el modelo de regresión lineal, podemos realizar predicciones sobre el comportamiento de nuestros datos, utilizando el modelo lineal. Para eso usamos el comando *predict()*.

```
predict(modeloRegresion,
        newdata = data.frame(Temp = 98))
```

```
##        1
## 6.188978
```

También podemos calcular los intervalos de confianza agregando parámetros a la función *predict()*:

```
predict(modeloRegresion,
        newdata = data.frame(Temp = 98),
        interval = "confidence",
        level = .95)
```

```
##        fit      lwr      upr
## 1 6.188978 4.816317 7.561639
```

Análisis de varianza

El análisis de varianza (comunmente conocido como ANOVA por sus siglas en inglés), es un caso particular del modelo lineal que estudiamos antes, pero en el caso en que las X_i son 0 o 1. Veamos por medio de un ejemplo por qu'e merece ser estudiado en particular.

Supongamos que se quiere comparar el rendimiento de un cultivo (medido en kilogramos por hectárea) en tres tipos de suelo (arenoso, arcilloso, limoso) teniendo 10 parcelas de cada uno. La variable rendimiento Y depende del tipo de suelo usado, y de la parcela en cuestión, por lo tanto tenemos $Y_{1,1}, \ldots, Y_{1,10}$ rendimientos para el tipo de suelo arenoso, para cada una de las 10 parcelas, análogamente tenemos $Y_{2,1}, \ldots, Y_{2,10}$ para el tipo de suelo arcilloso y finalmente $Y_{3,1}, \ldots, Y_{3,10}$ para el tipo de suelo limoso. Supondremos que podemos modelar las $Y_{i,1}, \ldots, Y_{i,10}$ como $Y_{i,j} = \mu + \alpha_i + e_{i,j}$ para $j = 1, \ldots, 10$, donde μ es una constante. Supondremos que los errores e_i son independientes, todos ellos con media 0 y varianza σ^2. La hipótesis que podríamos querer contrastar es $H_0 : \alpha_1 = \alpha_2 = \alpha_3 = 0$ o no. Si *no* se rechaza H_0 tenemos indicios de que el rendimiento medio, en cada tipo de suelo, es el mismo, ya que estamos suponiendo $E(e_{i,j}) = 0$ y por lo tanto $E(Y_{i,j}) = \mu$. Es decir *no* hay diferencias en el rendimiento.

Supongamos que tenemos la tabla:\

arenoso	arcilloso	limoso
6	17	13
10	15	16
8	3	9
6	11	12
14	14	15
17	12	16
9	12	17
11	8	13
7	10	18
11	13	14

Podemos cargarla a R mediante

```
arenoso<-c(6,10,8,6,14,17,9,11,7,11)
arcilloso<-c(17,15,3,11,14,12,12,8,10,13)
limoso<-c(13,16,9,12,15,16,17,13,18,14)
datos<-data.frame(cbind(arenoso,arcilloso,limoso))
stackeddata<-stack(datos)
```

Este último comando transforma los datos en dos columnas, una con los valores (denominada values) de rendimiento dados por la tabla, y la otra con el tipo de suelo al que corresponde dicho valor, denominada ind. Ahora ejecutamos

```
summary(aov(values~ind,data=stackeddata))
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## ind          2   99.2   49.60   4.245  0.025 *
## Residuals   27  315.5   11.69
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Obtuvimos una tabla con varios valores, entre ellos el valor del estadístico, que en este caso es 4.245, por lo tanto para realizar el test a nivel por ejemplo $\alpha = 0.05$, tenemos que comparar dicho valor con el que devuelve el comando qf(0.95,2,27):

```
qf(0.95,2,27)
```

```
## [1] 3.354131
```

Como $4.245 > 3.354$ se rechaza H_0 a nivel $\alpha = 0.05$. Esto significa que a nivel $\alpha = 0.05$ hay indicios de que el tipo de suelo influye sobre el rendimiento medio en los mismos.

El summary también devuelve el p-valor en la columna Pr(>F). En este caso 0.025, lo cual significa que $P_{H_0}(F_{2,27} > 4.245) = 0.025$. Por lo tanto como $0.025 < 0.05$ nuevamente rechazamos H_0.