Aprendizaje Estadístico - (84:04)

Alejandro Verri

Guía de Trabajos Prácticos 2021

Curso: Aprendizaje Estadístico (84.04)
Cátedra: Ing. Jemina García
Alumno: Ing. Alejandro Verri Kozlowski
Carrera: Doctorado en Ingeniería Civil
Lenguaje: R y RMarkdown [^1]
Librerías: `prettydoc`, `ggplot2`

Práctica 9/10

Ejercicio 1

Cargar los datos del paquete `cars` en el objeto `autos`

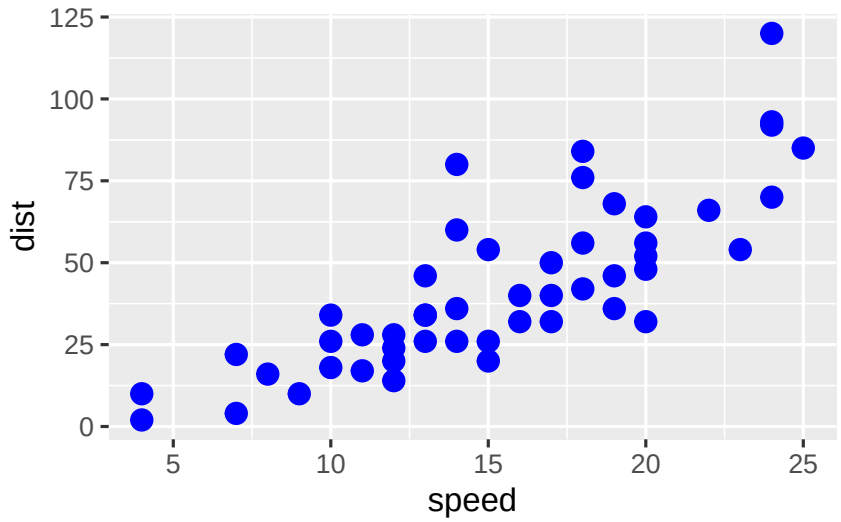
```
autos <- cars
str(autos)
```

```
## 'data.frame':    50 obs. of  2 variables:
##  $ speed: num  4 4 7 7 8 9 10 10 10 11 ...
##  $ dist : num  2 10 4 22 16 10 18 26 34 17 ...
```

Ejercicio 2

Realice el diagrama de dispersión para `X` vs. `Y` . ¿Qué observa?

```
ggplot(data= autos) +
  geom_point(mapping = aes(x=speed, y=dist),colour="blue",size=3)
```



Ejercicio 3

3. Estime la media y el desvío standard de cada una de las variables.

```
x <- autos$speed
y <- autos$dist
mX <- mean(x)
mY <- mean(y)
sX <- sd(x)
sY <- sd(y)
```

La media y desvío de las velocidades resultan iguales a `mX` = 15.4 y `sX` = 5.288 respectivamente. La distancia media de frenado y el desvío estándar, resultan en `mY` = 42.98 y `sY` = 25.77 respectivamente.

Ejercicio 4

Si se plantea un modelo* $E[Y_i|X_i] = \beta_o + \beta_1 X_i, i = 1, 2, .., 50$, halle los estimadores de mínimos cuadrados de β_o y β_1. Graficar la recta de cuadrados mínimos sobre el gráfico realizado en (2).

Si Y es una V.A. y $X = (X_1, X_2, ...X_n)$ es un vector aleatorio, la recta $g(X) = \beta_o + \beta_1 X$ hace mínimo el error cuadrático medio $ECM = E[(Y - g(X))^2]$. Derivando respecto de los estimadores e igualando a cero, la recta de cuadrados mínimos queda determinada según $\beta_o = E[Y] - \beta_1 E[X]$ y $\beta_1 = cov(X, Y)/var(X)$. Los estimadores resultan en

```
B <- c(
  bo = mY - cov(x,y)/var(x)*mX,
  b1 = cov(x,y)/var(x))
B
```

```
##         bo         b1
## -17.579095   3.932409
```

Debido a que X es un vector columna de una única variable aleatoria X_1, en este caso particular `cov(X,Y)` = 109.9 y `var(X)` = 27.96 son dos escalares.

Los estimadores de mínimos cuadrados pueden obtenerse también a partir de la matriz de diseño X asumiendo que $\hat{Y} = X\beta$. Haciendo mínimo al residuo $S(\beta) = \|Y - X\beta\|$ se obtiene el vector de estimadores $\beta = (X^T X)^{-1} X^T Y$. Aplicando la inversa de una matriz con la función `solve()` y la transpuesta con el operador `t()` se obtiene

```
n <- length(x)
X <- as.matrix(data.frame(I=rep(1,n),X=x))
```

```
B <- solve(t(X) %*% X) %*% t(X) %*% y
B
```

```
##          [,1]
## I -17.579095
## X   3.932409
```

Los estimadores pueden obtenerse también mediante la función de regresión lineal `lm()` de la librería `stats` de R, según

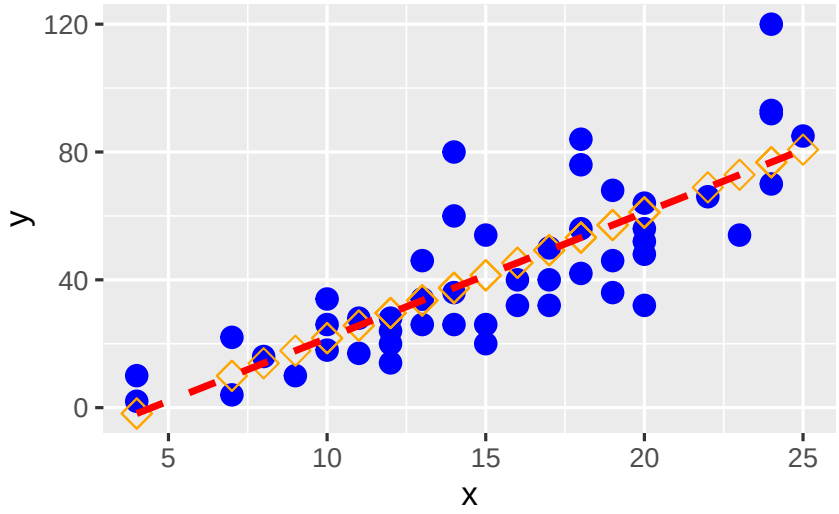
```
MODEL <- lm(formula = y ~ x)
B <- c(
  bo = MODEL$coefficients[[1]],
  b1 = MODEL$coefficients[[2]])
B
```

```
##          bo          b1
## -17.579095   3.932409
```

Ejercicio 5

Superponer sobre el gráfico anterior, en color naranja, los puntos correspondientes a los valores predichos.

```
yp <- B[1]+B[2]*x
ggplot() +
  geom_point(mapping = aes(x, y),colour="blue",size=3) +
  geom_point(mapping = aes(x, yp),colour="orange",size=3,shape=5)+
  geom_line(mapping = aes(x, yp),colour="red",size=1.1,linetype="dashed")
```



Ejercicio 6

¿Cuánto vale el estimador de σ^2 ?

Se puede demostrar que S^2 es un estimador insesgado de σ^2 si

$$S^2 = \frac{\|Y - \hat{Y}\|^2}{n - p}$$

donde p es el rango de β. Numéricamente, resulta igual a [1]

[1] El resultado de la operación matricial resulta en un array de 1x1 (singleton). El operador `()[[1]]` convierte el singleton en un escalar.

```
p <- length(B)
S2 <- ((y - yp)%*%(y - yp)/(n - p))[[1]]
S2
```

```
## [1] 236.5317
```

Ejercicio 7

Estime la matriz de covarianza de los estimadores obtenidos. La matriz de covarianza se puede obtener según $\Sigma_{\hat{\beta}} = \sigma^2 (X^T X)^{-1}$

```
S2*(solve(t(X)%*%X))
```

```
##           I          X
## I 45.676514 -2.6588234
## X -2.658823  0.1726509
```

¿Cuánto vale en este caso la matriz $X^T X$?

Se puede demostrar que la matriz $X^T X$ para el caso de un vector aleatorio con una única variable aleatoria $X_1 = [x_1 x_2 ... x_n]$ tiene como componentes

$$X^T X = \begin{pmatrix} n & \sum_{i=1}^{n} x_i \\ \sum_{i=1}^{n} x_i & \sum_{i=1}^{n} {x_i}^2 \end{pmatrix}$$

En R, la matriz anterior resulta igual a:

```
array(c(n,sum(x),sum(x),sum(x^2)),dim=c(2,2))
```

```
##      [,1]  [,2]
## [1,]   50   770
## [2,]  770 13228
```

Esta matriz, reporta los mismos valores que los que se obtienen a partir de la matriz de diseño:

```
(t(X)%*%X)
```

```
##     I     X
## I  50   770
## X 770 13228
```

Ejercicio 8

Verifique que $\sum_{i=1}^{n} (Y_i - \hat{Y}_i) = 0$

```
 sum(yp-y)
```

```
## [1] 2.877698e-13
```

Ejercicio 9

Centre las observaciones X_i 's y recalcule los estimadores de los parámetros.

```
mX <- mean(x)
z <- x - mX
n <- length(z)
Z <- as.matrix(data.frame(I=rep(1,n),X=z))
Bz <- solve(t(Z) %*% Z) %*% t(Z) %*% y
Bz
```

```
##        [,1]
## I 42.980000
## X  3.932409
```

Los estimadores son diferentes. Sin embargo, la predicción $\hat{Y}$ es la misma, ya que la diferencia entre ambas es despreciable $\|g(X) - g(X - \eta_X)\|^2 \approx 0$

```
yp <- B[1]+B[2]*x
yq <- Bz[1]+Bz[2]*z
(t(yp-yq)%*%(yp-yq))[[1]]
```

```
## [1] 4.687727e-26
```

¿Cambia el estimador de σ^2 ? Recalcule la estimación de la matriz de covarianza de los estimadores y compárela con la obtenida en (6).

Por su definición, el estimador de σ^2 dado por $S^2 = \|Y - \hat{Y}\|^2/(n - p)$, no cambia porque la predicción $\hat{Y}$ no cambió. Sin embargo la matriz de covarianza $\Sigma_{\hat{\beta}} = \sigma^2(Z^T Z)^{-1}$ será diferente

```
S2*(solve(t(Z)%*%Z))
```

```
##              I            X
## I 4.730634e+00 6.256467e-16
## X 6.256467e-16 1.726509e-01
```

Sin embargo, el determinante de la matriz de covarianza expresada en el cambio de variables $Z = X - E[X]$ es el mismo que la misma matriz en la variable X

```
det(S2*(solve(t(Z)%*%Z)))/det(S2*(solve(t(X)%*%X)))
```

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

Ejercicio 10

Ajustar un modelo polinomial que prediga y usando x y x^2

Para incluir un término más en la regresión, se incorpora una columna más en la matriz de diseño

```
X <- as.matrix(data.frame(I=rep(1,n), X1=x, X2=x^2))
B <- solve(t(X) %*% X) %*% t(X) %*% y
B
```

```
##         [,1]
## I  2.4701378
## X1 0.9132876
## X2 0.0999593
```

¿Encuentra alguna evidencia de que el término cuadrático mejora el ajuste del modelo?

La manera de medir la eficiencia del nuevo ajuste es a través del estimador de la varianza de la mediana condicional

```
p <- length(B)
S2_old <- S2
S2 <- ((y - yp)%*%(y - yp)/(n - p))[[1]]
S2/S2_old
```

```
## [1] 1.021277
```

La nueva varianza condicional es un 2% mayor a la varianza del modelo lineal con un único término. Luego, el modelo con un término cuadrático no mejora la eficiencia de la predicción.

Graficar la curva obtenida sobre el gráfico realizado en (2).

```
yp_old <- yp
yp <- B[1]+B[2]*x+B[3]*x^2
ggplot() +
  geom_point(mapping = aes(x, y),colour="blue",size=2) +
  geom_point(mapping = aes(x, yp_old),colour="orange",size=2,shape=5)+
  geom_line(mapping = aes(x, yp_old),colour="red",size=1.1,linetype=2)+
  geom_point(mapping = aes(x, yp),colour="orange",size=3,shape=8)+
geom_line(mapping = aes(x, yp),colour="darkred",size=1.1,linetype=4)
```