Modelo lineal normal

Guillermo Ayala Gallego

2025-04-09

Table of contents

Introducción

¿Qué información tenemos?

- Tenemos información (numérica, categórica) sobre una serie de muestras u observaciones.
- Para la i-ésima observación tenemos p variables que recogemos en el vector $\mathbf{x}_i \in \mathbb{R}^p$ donde
$$\mathbf{x}_i = \begin{bmatrix} x_{i1} \\ \vdots \\ x_{ip} \end{bmatrix}.$$
- $\mathbf{x}_i^T = (x_{i1}, \ldots, x_{ip})$.
- Estas variables pueden ser númericas o categóricas.
- Pueden valores fijados por nosotros (dosis de medicación por ejemplo) o bien observados y por lo tanto realizaciones de variables aleatorias.

Variable respuesta

- En lo que sigue consideramos que hay una variable **importante**.

- La variable importante (para el experimentador) se le llama **variable respuesta**.
- En denominaciones más clásicas, **variable dependiente**.

- El resto de variables del estudio (las que recogemos en $\mathbf{x}_i$) son las variables predictoras o independientes.

- Nuestra información está formada por $(\mathbf{x}_i, y_i)$ con $i = 1, \ldots, n$.

¿Qué problema queremos resolver?

- Una respuesta fácil es decir que queremos conocer el valor de la variable respuesta utilizando las variables predictoras.
  - La respuesta anterior es **falsamente** simple.
  - ¿Solamente queremos conocer el valor de la respuesta?
  - Quizás estamos pensando en un futuro en donde conozcamos las variables predictoras y nos interese saber cuál será la respuesta correspondiente.
  - ¿El valor exacto de la respuesta?
  - ¿La media de la respuesta?
  - ¿Un valor numérico que aproxime cada una de estas cantidades o bien un intervalo que las contenga?

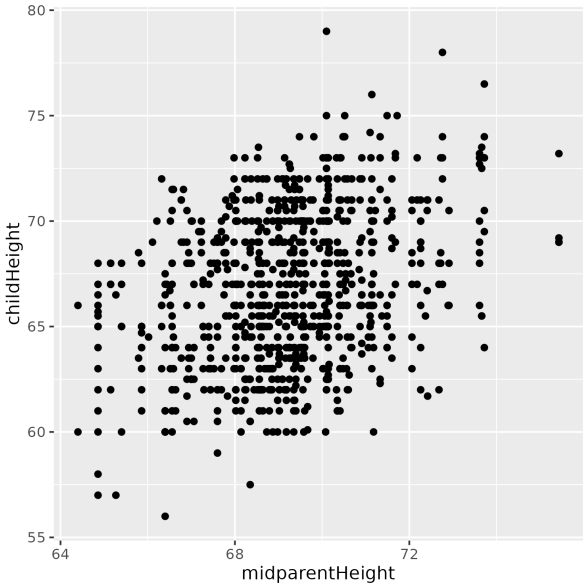
Media condicionada

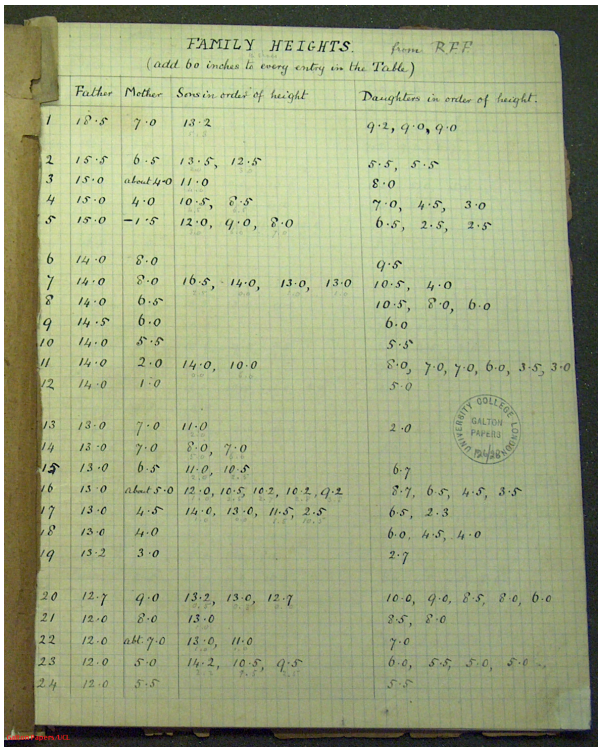
- Los valores observados y_i consideraremos que son realizaciones de una variable aleatoria Y.
  - Realmente en lo que sigue modelizaremos el comportamiento aleatorio de la variable Y_i condicionada a los valores observados $\mathbf{x}_i$.
  - Es decir, nuestro interés estará en **la distribución condicionada**.
  - Los valores Y_i serán independientes entre sí.
  - El interés se centrará (fundamentalmente) en la dependencia de la media de Y_i respecto de las covariables.
  - Nuestro interés fundamental (pero no único) estará en conocer las medias condicionadas $\mu_i = E[Y_i|\mathbf{x}_i]$.

Datos de Galton

- Históricamente podemos considerar el origen de este tipo de modelos.
- El problema que se plantea Galton es estudiar la posible relación que puede tener la estatura de un hijo o hija en edad adulta con las estaturas de sus padres.

Datos de Galton...




FAMILY HEIGHTS. from REF
(add 60 inches to every entry in the Table.)

	Father	Mother	Sons in order of height	Daughters in order of height.
1	18·5	7·0	13·2	9·2, 9·0, 9·0
2	15·5	6·5	13·5, 12·5	5·5, 5·5
3	15·0	about 4·0	11·0	8·0
4	15·0	4·0	10·5, 8·5	7·0, 4·5, 3·0
5	15·0	−1·5	12·0, 9·0, 8·0	6·5, 2·5, 2·5
6	14·0	8·0		9·5
7	14·0	8·0	16·5, 14·0, 13·0, 13·0	10·5, 4·0
8	14·0	6·5		10·5, 8·0, 6·0
9	14·5	6·0		6·0
10	14·0	5·5		5·5
11	14·0	2·0	14·0, 10·0	8·0, 7·0, 7·0, 6·0, 3·5, 3·0
12	14·0	1·0		5·0
13	13·0	7·0	11·0	2·0
14	13·0	7·0	8·0, 7·0	
15	13·0	6·5	11·0, 10·5	6·7
16	13·0	about 5·0	12·0, 10·5, 10·2, 10·2, 9·2	8·7, 6·5, 4·5, 3·5
17	13·0	4·5	14·0, 13·0, 11·5, 2·5	6·5, 2·3
18	13·0	4·0		6·0, 4·5, 4·0
19	13·2	3·0		2·7
20	12·7	9·0	13·2, 13·0, 12·7	10·0, 9·0, 8·5, 8·0, 6·0
21	12·0	8·0	13·0	8·5, 8·0
22	12·0	abt 7·0	13·0, 11·0	7·0
23	12·0	5·0	14·2, 10·5, 9·5	6·0, 5·5, 5·0, 5·0
24	12·0	5·5		5·5

Modelos sobre la media

- Nos planteamos conocer la media de la respuesta aleatoria.
- Pretendemos conocer la media de la respuesta aleatoria **condicionada** a los valores de las variables predictoras.
- Denotamos

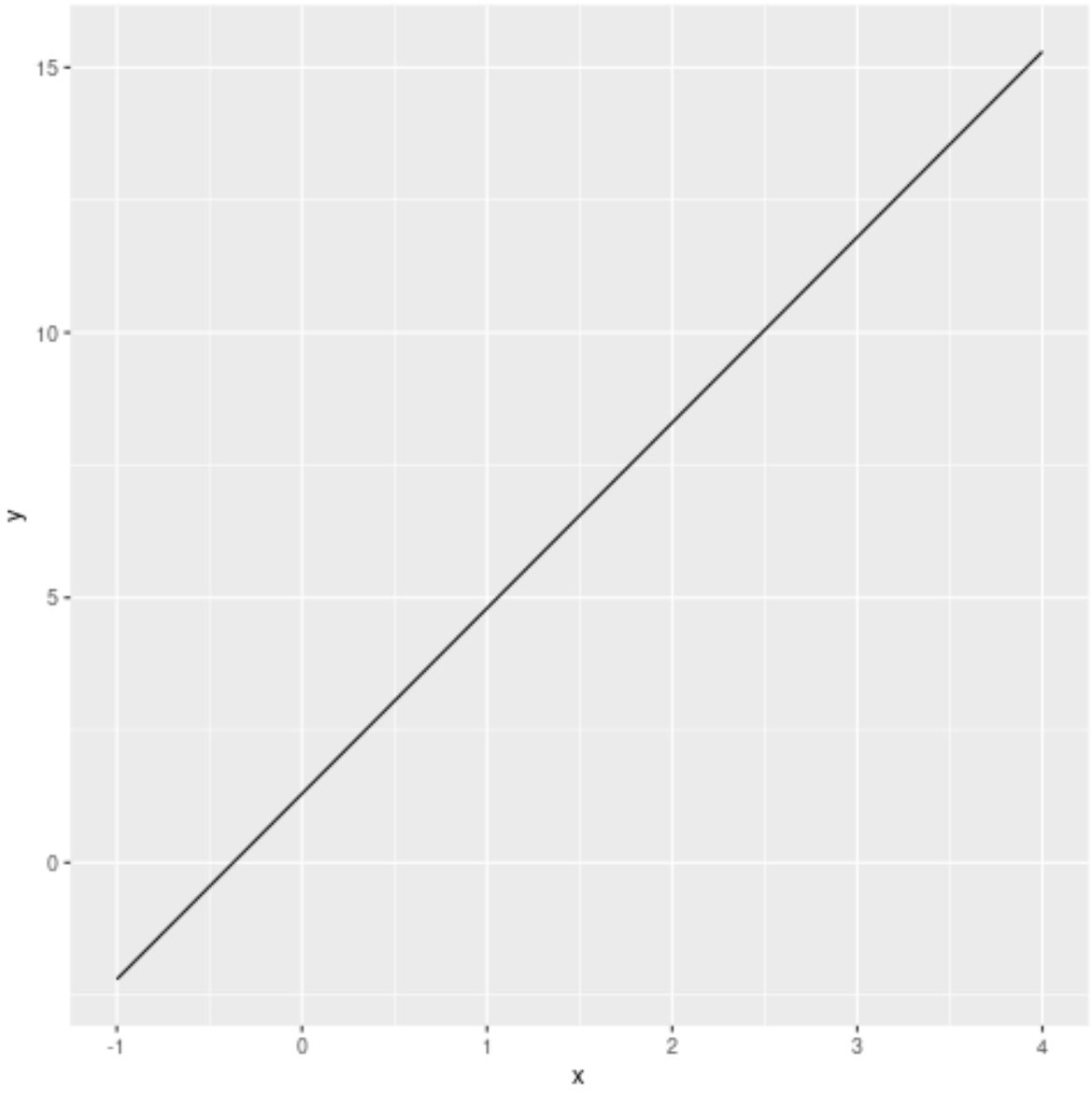
$$\mu_i = E[Y_i|\mathbf{x}_i].$$

Dependencia lineal

¿Qué tipos de dependencia vamos a considerar?

- En la mayor parte de los casos dependencias de tipo lineal.
- Suponemos dos predictores $\mathbf{x} = (x_1, x_2)^T$ tales que x_1 es numérico y el segundo es una variable categórica binaria codificada con 1 y 0.
- ¿Cómo modelizamos la dependencia de la media condicionada respecto de x_1?
- Una dependencia lineal vendría dada como

$$\mu_i = \beta_0 + \beta_1 x_{i1}.$$



¿Y la dependencia de las μ_i respecto de la variable binaria?

- Obviamente simplemente tenemos dos valores.
- Un modo simple es

$$\mu_i = \beta_0 + \beta_2 x_{i2}.$$

- Cuando $x_{i2} = 0$

$$\mu_i = \beta_0.$$

- Cuando $x_{i2} = 1$

$$\mu_i = \beta_0 + \beta_2.$$

¿Y las dos variables predictoras conjuntamente consideradas?

- El modelo más sencillo sería:

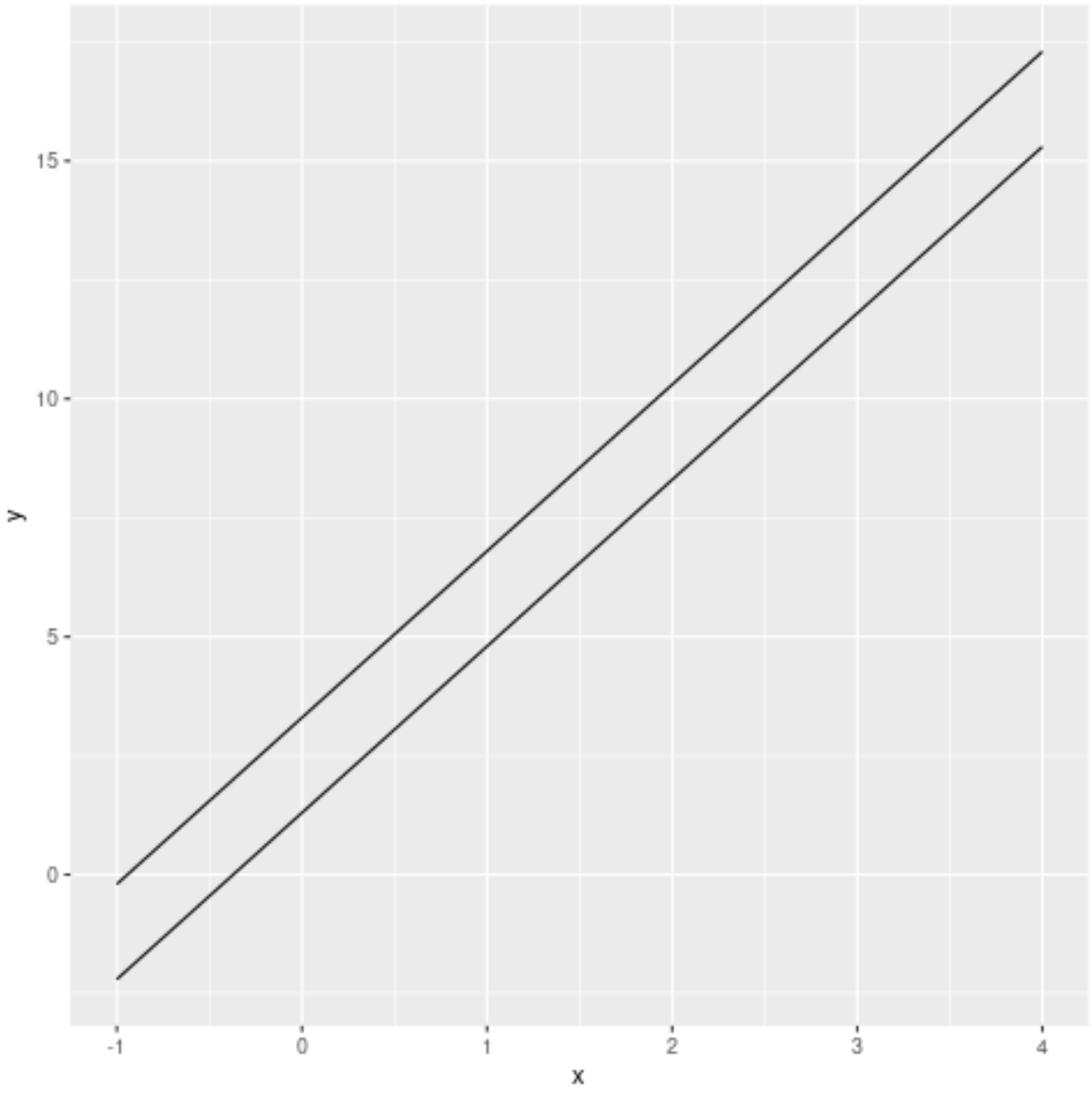
$$\mu_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2}.$$

- Cuando $x_{i2} = 0$ entonces

$$\mu_i = \beta_0 + \beta_1 x_{i1}.$$

- Cuando $x_{i2} = 1$ entonces

$$\mu_i = \beta_0 + \beta_2 + \beta_1 x_{i1}.$$



¿Como podemos expresar la dependencia de ambas covariables?

- Otra vez recurrimos a la opción de expresarlo de un modo lineal.

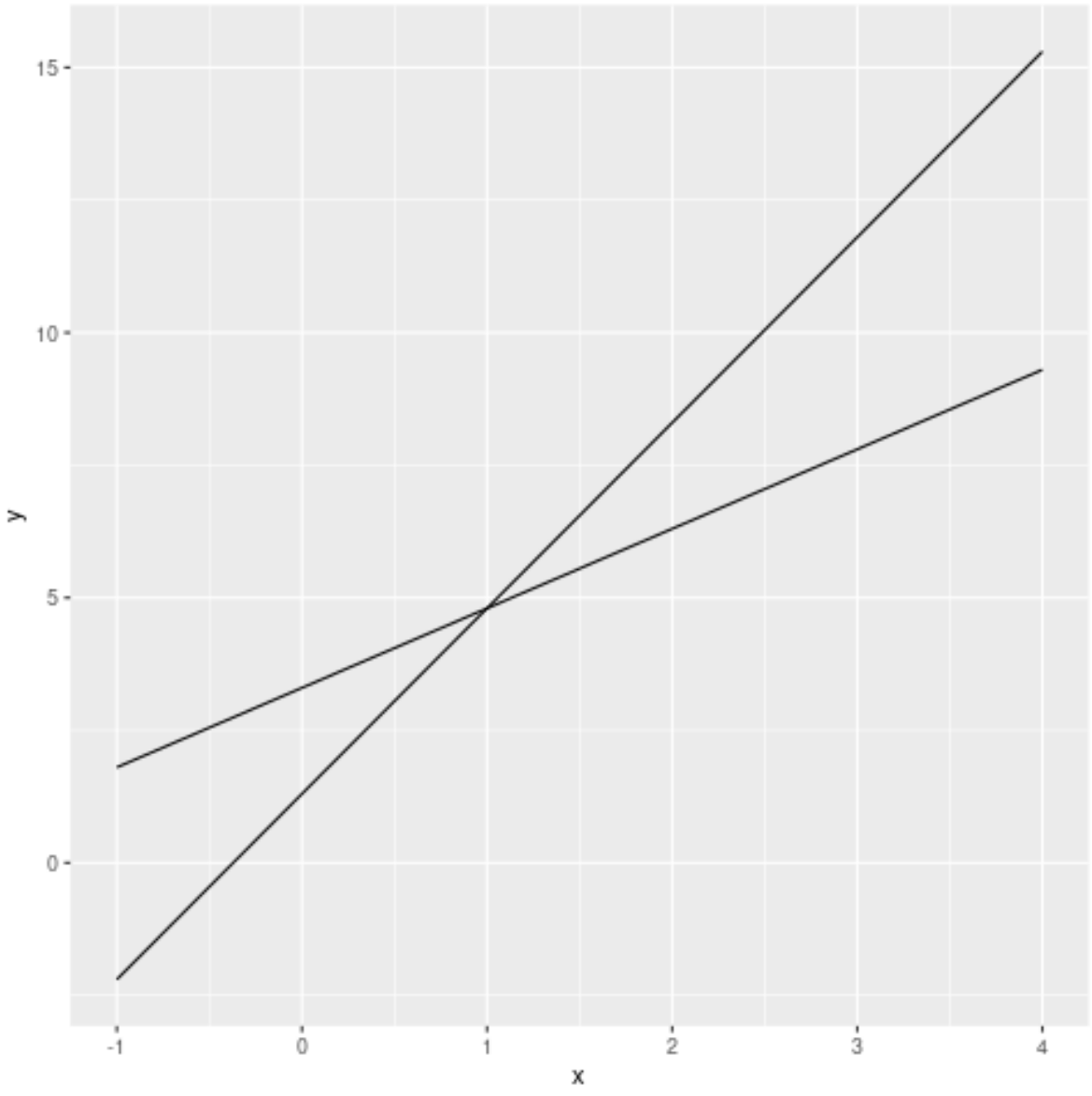
$$\mu_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \beta_3 x_{i1} x_{i2}$$

- Cuando $x_{i2} = 0$ tenemos

$$\mu_i = \beta_0 + \beta_1 x_{i1}.$$

  Cuando $x_{i2} = 1$ tenemos

$$\mu_i = (\beta_0 + \beta_2) + (\beta_1 + \beta_3) x_{i1}.$$



Predictor categórico y variables dummy

- Cuando consideramos una variable predictora categórica con más de dos categorías entonces es habitual codificarla utilizando variables tontas.
- Si la variable predictora categórica tiene I categorías entonces se elige una categoría de referencia (por ejemplo, la primera).
- Las variables binarias asociadas a la variable original x serían: $v_1 = 1$ si $x = 2$ y cero en otro caso; $v_2 = 1$ si $x = 3$ y cero en otro caso; ...; $v_{I-1} = 1$ si $x = I$ y cero en otro caso.
- Obviamente cuando todas las variables v son nulas estamos en la primera categoría.

- Si solamente tenemos la variable categórica como predictora entonces la media sería función lineal de x del siguiente modo:

$$\mu_i = \beta_0 + \beta_1 v_1 + ... + \beta_{I-1} v_{I-1}.$$

- Por ejemplo, supongamos una numérica x y una categórica v con I categorías. Construimos las variables tontas siendo I la de referencia. Podemos considerar modelos como

$$\mu_i = \beta_0 + \beta_1 x_i,$$

que lo expresamos como

$$y \sim x$$

- Un modelo que contiene solamente a v sería el dado previamente

$$\mu_i = \beta_0 + \beta_1 v_{i1} + ... + \beta_{I-1} v_{i,I-1}.$$

y lo expresamos como

$$y \sim v$$

- Un modelo que contempla ambas variables puede ser

$$\mu_i = \beta_0 + \beta_1 x_i + \beta_2 v_{i1} + ... + \beta_I v_{i,I-1}$$

Este modelo lo podemos abreviar como

$$y \sim x + v$$

- Un modelo más completo que contempla la posible interacción sería

$$\mu_i = \beta_0 + \beta_1 x_i + \beta_2 v_{i1} + ... + \beta_I v_{i,I-1} + \beta_{I+1} x_i v_{i1} + ... + \beta_{2I-1} x_i v_{i,(I-1)}$$

Esto lo indicaremos como

$$y \sim x * v$$
$$y \sim x + v + x : v$$

Recta de mínimos cuadrados

- Nuestros datos son (x_i, y_i) con $i = 1, ..., n$.
- Un modelo sencillo podría ser

$$y_i = \beta_0 + \beta_1 x_i, \ i = 1, ..., n$$

- No es posible.
- Una **buena** solución aproximada.
- Nos conformamos con

$$y_i \approx \beta_0 + \beta_1 x_i, \ i = 1, ..., n$$

- Una posibilidad para determinar unos **buenos** valores para β_0 y β_1 es considerar

$$S_t = \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2.$$

- Podemos obtener los valores que minimizan la función anterior igualando las derivadas parciales a cero.

$$\frac{\partial S_t}{\partial \beta_0} = -2 \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i),$$

$$\frac{\partial S_t}{\partial \beta_1} = -2 \sum_{i=1}^{n} x_i (y_i - \beta_0 - \beta_1 x_i),$$

- Igualando a cero

$$\sum_{i=1}^{n} y_i - n\hat{\beta}_0 - \hat{\beta}_1 \sum_{i=1}^{n} x_i = 0,$$

$$\sum_{i=1}^{n} x_i y_i - \hat{\beta}_0 \sum_{i=1}^{n} x_i - \hat{\beta}_1 \sum_{i=1}^{n} x_i^2 = 0,$$

- Haciendo algún cálculo se obtiene

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n} x_i y_i - (\sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i)/n}{\sum_{i=1}^{n} x_i^2 - (\sum_{i=1}^{n} x_i)^2/n}.$$

- Se sigue que

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}.$$

Recta de mínimos cuadrados ...

- Si denotamos

$$S_{xy} = \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y}),$$

y

$$S_{xx} = \sum_{i=1}^{n} (x_i - \bar{x})^2, \quad S_{yy} = \sum_{i=1}^{n} (y_i - \bar{y})^2,$$

entonces

$$\hat{\beta}_1 = \frac{S_{xy}}{S_{xx}}.$$

Predicción y residuo

- Nos planteamos encontrar unas buenas aproximaciones para y_i utilizando x_i.
- Lo natural sería tomar el siguiente valor la **predicción** o **valor ajustado** para y_i

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i.$$

- El **residuo** es

$$e_i = y_i - \hat{y}_i.$$

Modelo

Regresión lineal simple

- $Y_i = \beta_0 + \beta_1 x_i + \epsilon_i$ para $i = 1, \ldots, n$.
- $\epsilon_i \sim N(0, \sigma^2)$.
- Los errores ϵ_i son independientes.

Regresión lineal simple...

- Estamos asumiendo que la distribución **condicionada** de Y_i al predictor x_i es normal con media $\beta_0 + \beta_1 x_i$ y varianza σ^2,

$$Y_i | x_i \sim N(\beta_0 + \beta_1 x_i, \sigma^2),$$

y que los distintos Y_i son independientes entre sí.

Representación matricial

- Tenemos el **vector de respuestas aleatorias**, $\mathbf{Y}$; la **matriz de modelo**, $\mathbf{X}$; el **vector de coeficientes**, y el **vector de errores aleatorios**, dados por

$$\mathbf{Y} = \begin{bmatrix} Y_1 \\ \vdots \\ Y_n \end{bmatrix}; \quad \mathbf{X} = \begin{bmatrix} 1 & x_1 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix}; \quad = \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}; \quad = \begin{bmatrix} \epsilon_1 \\ \vdots \\ \epsilon_n \end{bmatrix}$$

- El modelo de regresión lineal simple se puede formular como

$$\mathbf{Y} = \mathbf{X} \ + \ .$$

- El vector de errores aleatorios sigue una distribución normal multivariante

$$\sim N_2(\mathbf{0}_p, \sigma^2 \mathbf{I}_p).$$

Verosimilitud

- La función de verosimilitud viene dada por

$$L(\beta_0, \beta_1, \sigma^2; y_1, \dots, y_n) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}(y_i - \beta_0 - \beta_1 x_i)^2} = \frac{1}{(2\pi)^{n/2}\sigma^n} e^{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \beta_0 - \beta_1 x_i)^2} \quad (1)$$

- Los estimadores máximo verosímiles de los coeficientes corresponden con los estimadores mínimo cuadráticos.

Ejemplo

Paquetes

- Cargamos los paquetes que necesitamos.

```
pacman::p_load(Biobase,ggplot2)
```

Datos

- Leemos los datos.
- La primera es la más ortodoxa.

```
data(gse25171,package="tamidata2")
sel0 = which("261892_at"==fData(gse25171)[,"PROBEID"])
df0 = data.frame(pData(gse25171)[,c("time","Pi")],
                 expression=exprs(gse25171)[sel0,])
```

```
(p = ggplot(df0,aes(x=time,y=expression)) + geom_point())
```

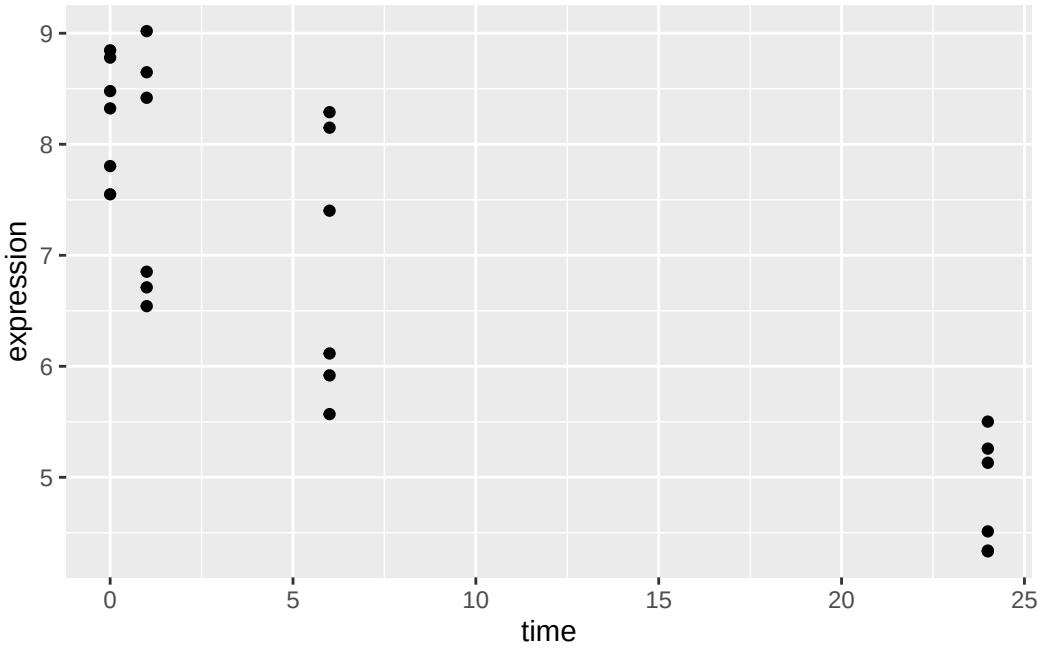


Figure 1: Expression vs time

- Podemos querer guardar el gráfico.

```
ggsave("gse25171_261892_at.png",p)
```

Recta de mínimos cuadrados

- Ajustamos.

```
fit = lm(expression ~ time, data = df0)
```

- ¿Qué clase de objeto tenemos?

```r
class(fit)
```

```
[1] "lm"
```

- Vemos lo que el método `print` nos muestra de un objeto de clase `lm`.

```r
fit
```

```

Call:
lm(formula = expression ~ time, data = df0)

Coefficients:
(Intercept)         time
     7.9684      -0.1331
```

- Los coeficientes de la recta de mínimos cuadrados los podemos obtener con el método `coef` o `coefficients`.

```r
coef(fit)
```

```
(Intercept)        time
  7.9684362  -0.1330703
```

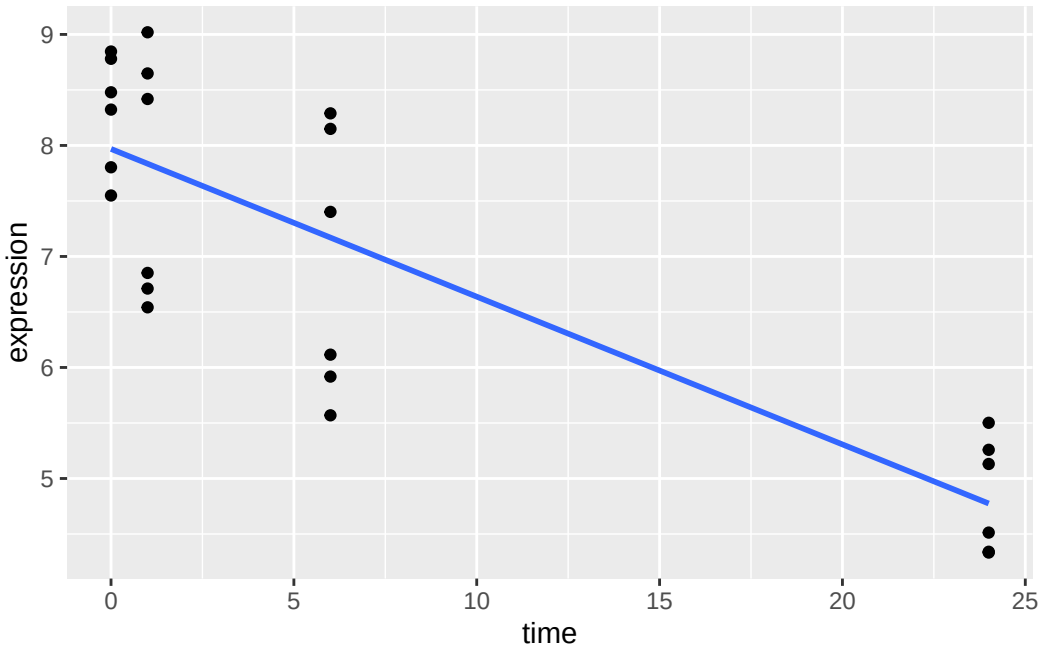
```r
coefficients(fit)
```

```
(Intercept)        time
  7.9684362  -0.1330703
```

- Añadimos la recta de mínimos cuadrados al diagrama de puntos.

```r
(p = ggplot(df0,aes(x=time,y=expression))+geom_point() +
    geom_smooth(method='lm', se = FALSE))
```

```
`geom_smooth()` using formula = 'y ~ x'
```



- El método `predict` aplicado al objeto de clase `lm` nos proporciona las predicciones para las observaciones.

```
predict(fit)
```

```
GSM618324.CEL.gz GSM618325.CEL.gz GSM618326.CEL.gz GSM618327.CEL.gz
        7.968436         7.968436         7.835366         7.835366
GSM618328.CEL.gz GSM618329.CEL.gz GSM618330.CEL.gz GSM618331.CEL.gz
        7.170014         7.170014         4.774749         4.774749
GSM618332.CEL.gz GSM618333.CEL.gz GSM618334.CEL.gz GSM618335.CEL.gz
        7.968436         7.968436         7.835366         7.835366
GSM618336.CEL.gz GSM618337.CEL.gz GSM618338.CEL.gz GSM618339.CEL.gz
        7.170014         7.170014         4.774749         4.774749
GSM618340.CEL.gz GSM618341.CEL.gz GSM618342.CEL.gz GSM618343.CEL.gz
        7.968436         7.968436         7.835366         7.835366
GSM618344.CEL.gz GSM618345.CEL.gz GSM618346.CEL.gz GSM618347.CEL.gz
        7.170014         7.170014         4.774749         4.774749
```

- El método `residuals` o `resid` nos proporciona los residuos.

```
residuals(fit)
```

```
GSM618324.CEL.gz GSM618325.CEL.gz GSM618326.CEL.gz GSM618327.CEL.gz
       0.3545741        0.8777147       -0.9836174        0.5831095
GSM618328.CEL.gz GSM618329.CEL.gz GSM618330.CEL.gz GSM618331.CEL.gz
      -1.2520196        0.9791937       -0.4345729        0.3560575
GSM618332.CEL.gz GSM618333.CEL.gz GSM618334.CEL.gz GSM618335.CEL.gz
      -0.1650657        0.5102758       -1.2935869        0.8132667
GSM618336.CEL.gz GSM618337.CEL.gz GSM618338.CEL.gz GSM618339.CEL.gz
      -1.6008608        1.1191170       -0.4417852       -0.2617990
GSM618340.CEL.gz GSM618341.CEL.gz GSM618342.CEL.gz GSM618343.CEL.gz
      -0.4191948        0.8119677       -1.1244602        1.1839258
GSM618344.CEL.gz GSM618345.CEL.gz GSM618346.CEL.gz GSM618347.CEL.gz
      -1.0545087        0.2315680        0.4836551        0.7270455
```

```
resid(fit)
```

```
GSM618324.CEL.gz GSM618325.CEL.gz GSM618326.CEL.gz GSM618327.CEL.gz
       0.3545741        0.8777147       -0.9836174        0.5831095
GSM618328.CEL.gz GSM618329.CEL.gz GSM618330.CEL.gz GSM618331.CEL.gz
      -1.2520196        0.9791937       -0.4345729        0.3560575
GSM618332.CEL.gz GSM618333.CEL.gz GSM618334.CEL.gz GSM618335.CEL.gz
      -0.1650657        0.5102758       -1.2935869        0.8132667
GSM618336.CEL.gz GSM618337.CEL.gz GSM618338.CEL.gz GSM618339.CEL.gz
      -1.6008608        1.1191170       -0.4417852       -0.2617990
GSM618340.CEL.gz GSM618341.CEL.gz GSM618342.CEL.gz GSM618343.CEL.gz
      -0.4191948        0.8119677       -1.1244602        1.1839258
GSM618344.CEL.gz GSM618345.CEL.gz GSM618346.CEL.gz GSM618347.CEL.gz
      -1.0545087        0.2315680        0.4836551        0.7270455
```

- El método `summary` aplicado a un objeto `lm`.

```
(fit.s = summary(fit))
```

```
Call:
lm(formula = expression ~ time, data = df0)

Residuals:
    Min      1Q  Median      3Q     Max
-1.6009 -0.5772  0.2931  0.7483  1.1839

Coefficients:
```

```
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  7.96844    0.23130  34.451  < 2e-16 ***
time        -0.13307    0.01868  -7.122 3.84e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.8836 on 22 degrees of freedom
Multiple R-squared:  0.6975,    Adjusted R-squared:  0.6837
F-statistic: 50.73 on 1 and 22 DF,  p-value: 3.842e-07
```

- Tenemos un objeto de clase

```
class(fit.s)
```

```
[1] "summary.lm"
```

- Algo de la información que contiene el objeto es

```
attributes(fit.s)
```

```
$names
 [1] "call"          "terms"         "residuals"     "coefficients"
 [5] "aliased"       "sigma"         "df"            "r.squared"
 [9] "adj.r.squared" "fstatistic"    "cov.unscaled"

$class
[1] "summary.lm"
```

- La matriz de modelo se obtiene con

```
model.matrix(fit)
```

```
                (Intercept) time
GSM618324.CEL.gz           1    0
GSM618325.CEL.gz           1    0
GSM618326.CEL.gz           1    1
GSM618327.CEL.gz           1    1
GSM618328.CEL.gz           1    6
GSM618329.CEL.gz           1    6
GSM618330.CEL.gz           1   24
GSM618331.CEL.gz           1   24
```

```
GSM618332.CEL.gz           1    0
GSM618333.CEL.gz           1    0
GSM618334.CEL.gz           1    1
GSM618335.CEL.gz           1    1
GSM618336.CEL.gz           1    6
GSM618337.CEL.gz           1    6
GSM618338.CEL.gz           1   24
GSM618339.CEL.gz           1   24
GSM618340.CEL.gz           1    0
GSM618341.CEL.gz           1    0
GSM618342.CEL.gz           1    1
GSM618343.CEL.gz           1    1
GSM618344.CEL.gz           1    6
GSM618345.CEL.gz           1    6
GSM618346.CEL.gz           1   24
GSM618347.CEL.gz           1   24
attr(,"assign")
[1] 0 1
```

- En el `summary` tenemos los contrastes.

```
summary(fit)
```

```
Call:
lm(formula = expression ~ time, data = df0)

Residuals:
    Min      1Q  Median      3Q     Max
-1.6009 -0.5772  0.2931  0.7483  1.1839

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  7.96844    0.23130  34.451  < 2e-16 ***
time        -0.13307    0.01868  -7.122 3.84e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.8836 on 22 degrees of freedom
Multiple R-squared:  0.6975,    Adjusted R-squared:  0.6837
F-statistic: 50.73 on 1 and 22 DF,  p-value: 3.842e-07
```

Análisis de la varianza de una vía

- Tenemos una variable de interés Y y pretendemos estudiar su posible dependencia de un factor experimental.
- El experimentador tiene un **factor** de interés con distintos niveles y se pretende evaluar su influencia en la variable respuesta.
- Si el factor tiene dos niveles entonces podemos comparar las medias mediante un test de la t (t-test).
- Con más de dos niveles del factor hemos de utilizar otras herramientas.
- Nos ocupamos (de un modo muy simple) de lo que se conoce como **experimentos con un solo factor completamente aleatorizado**.

Comparando grupos

- Tenemos I condiciones distintas.
- En cada una de ellas n_i muestras.
- $\sum_{i=1}^{I} n_i = n$ es el total de muestras.
- Y_{ij} denota la respuesta aleatoria en la j-ésima muestra de la i-ésima condición.

Modelo

- Suponemos que Y_{ij} con $j = 1, \ldots, n_i$ son independientes y con la misma distribución.
- El modelo de análisis de la varianza de una vía es

$$Y_{ij} = \beta_0 + \beta_i + \epsilon_{ij}, \tag{2}$$

- Donde se asume que $\epsilon_{ij} \sim N(0, \sigma^2)$ y son independientes entre si para los distintos grupos y dentro de cada grupo.
- Estamos asumiendo que $Y_{ij} \sim N(\beta_0 + \beta_i, \sigma^2)$.

Interpretación de los parámetros

- β_0 es la media global de todos los grupos.
- β_i sería la diferencia de la media del grupo i respecto de esta media global.
- ϵ_{ij} sería el error aleatorio de la observación (i, j) respecto del valor medio $\beta_0 + \beta_i$.

- Se trata de evaluar si hay diferencias entre grupos.
- Bajo el modelo que acabamos de formular se traduce en la siguiente hipótesis nula

$$H_0 : \beta_1 = \beta_2 = ... = \beta_I = 0.$$

frente a que algún par de los β_i sean distintos.

Contraste

- ¿Cómo podemos contrastar la hipótesis nula anterior?
- Denotamos por y_{ij} la j-ésima muestra observada bajo la condición i ($i = 1, ..., I$ y $j = 1, ..., n_i$).
- Denotamos las medias muestrales para cada grupo como

$$\bar{y}_{i\cdot} = \sum_{j=1}^{n_i} \frac{y_{ij}}{n_i}.$$

- La media de todas las observaciones o media total como

$$\bar{y}_{\cdot\cdot} = \sum_{i=1}^{I} \sum_{j=1}^{n_i} \frac{y_{ij}}{n}.$$

- Definimos la **suma de cuadrados intra** o **del error** como

$$SS(Within) = \sum_{i=1}^{I} \sum_{j=1}^{n_i} (y_{ij} - \bar{y}_{i\cdot})^2,$$

y la **suma de cuadrados entre** como

$$SS(Between) = \sum_{i=1}^{I} n_i (\bar{y}_{i\cdot} - \bar{y}_{\cdot\cdot})^2.$$

- El estadístico para contrastar esta hipótesis nula es

$$F = \frac{SS(Between)/(I-1)}{SS(Within)/(n-I)}.$$

Tabla de análisis de la varianza

Source	SS	df	MS	F	p
Between	SS(B)	I-1	$\frac{SS(B)}{I-1}$	$\frac{SS(B)/(I-1)}{SS(W)/(n-I)}$	$P(>F)$
Within	SS(W)	n-I	$\frac{SS(W)}{n-I}$		
Total	SS(B) + SS(W)				

Contraste...

- **Bajo la hipótesis nula de que todas las medias son la misma** (y puesto que asumimos una misma varianza) tendríamos una distribución común bajo todas las condiciones.
- Asumiendo la hipótesis nula el estadístico F se distribuye como un F con $I-1$ y $n-I$ grados de libertad,
$$F \sim F_{I-1,n-I}.$$
- Bajo la hipótesis alternativa, los valores de F tenderán a ser **grandes** o mayores que los esperables bajo la hipótesis nula.
- La **región crítica** (donde rechazamos la hipótesis nula) será un intervalo de la forma $[c, +\infty)$.
- Si tomamos como valor c el valor observado tendremos el p-valor.

Otra formulación del modelo

- Consideramos las variables que indican las categorías y que consideran como categoría de referencia el primer grupo.
- $$E[Y_{1j}] = \beta_0$$
  y
  $$E[Y_{ij}] = \beta_0 + \beta_i$$
  para $i = 2, \ldots, I$. De un modo conjunto:
  $$E[Y_{ij}] = \beta_0 + \beta_2 v_{2j} + \ldots + \beta_I v_{Ij}$$
  donde $v_{ij} = 1$ si estamos en el grupo i y cero en otro caso.
- La hipótesis nula de que no hay diferencias entre las medias de los distintos grupos vendría formulada como
$$H_0 : \beta_2 = \ldots = \beta_I = 0.$$

tamidata2::gse25171

- Analizamos los datos correspondientes a la sonda `261892_at`.
- Consideramos las variables fenotípicas `time2` y `Pi`.

```
pacman::p_load(Biobase)
data(gse25171,package="tamidata2")
head(pData(gse25171),n=2)
```

```
                time time2        Pi replication
GSM618324.CEL.gz    0 Short Treatment           1
GSM618325.CEL.gz    0 Short   Control           2
```

```
sel0 = which("261892_at"==fData(gse25171)[,"PROBEID"])
df0 = data.frame(pData(gse25171)[,c("time","Pi")],
                 expression=exprs(gse25171)[sel0,])
```

- Construimos una variable categórica combinando las dos predictoras previas.

```
time2Pi = vector("list",ncol(gse25171))
for(i in seq_along(time2Pi))
  time2Pi[[i]] = paste0(pData(gse25171)[,"time2"][i],
                        pData(gse25171)[,"Pi"][i])
time2Pi = factor(unlist(time2Pi))
levels(time2Pi)
```

```
[1] "MediumControl"   "MediumTreatment" "ShortControl"    "ShortTreatment"
```

- Construimos un `data.frame` en el que consideramos la expresión de la sonda y la variable que acabamos de construir.

```
sel0 = which("261892_at"==fData(gse25171)[,"PROBEID"])
df1 = data.frame(time2Pi,expression=exprs(gse25171)[sel0,])
summary(df1[,"time2Pi"])
```

```
  MediumControl MediumTreatment    ShortControl  ShortTreatment
              6               6               6               6
```

- Realizamos el ajuste del modelo y vemos la tabla anova.

```r
fit2 = lm(expression ~ time2Pi,data=df1)
summary(fit2)
```

```
Call:
lm(formula = expression ~ time2Pi, data = df1)

Residuals:
     Min       1Q   Median       3Q      Max
-1.98463 -0.62805  0.04225  0.54559  1.79155

Coefficients:
                       Estimate Std. Error t value Pr(>|t|)
(Intercept)              6.4976     0.4007  16.217 5.66e-13 ***
time2PiMediumTreatment  -1.2419     0.5666  -2.192 0.040407 *
time2PiShortControl      2.2010     0.5666   3.884 0.000922 ***
time2PiShortTreatment    0.7991     0.5666   1.410 0.173828
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.9814 on 20 degrees of freedom
Multiple R-squared:  0.6607,    Adjusted R-squared:  0.6098
F-statistic: 12.98 on 3 and 20 DF,  p-value: 6.219e-05
```

```r
fit2 = aov(expression ~ time2Pi,data=df1)
summary(fit2)
```

```
            Df Sum Sq Mean Sq F value   Pr(>F)
time2Pi      3  37.52  12.505   12.98 6.22e-05 ***
Residuals   20  19.26   0.963
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Mínimos cuadrados

- Disponemos de los datos $(\mathbf{x}_i, y_i)$ con $i = 1, \dots, n$ siendo $\mathbf{y} = (y_1, \dots, y_n)^T$, las respuestas observadas; $\mathbf{x}_i = (x_{i1}, \dots, x_{ip})^T$, los predictores correspondientes a la i-ésima observación.

- La matriz de modelo que recoge los valores de los predictores:

$$\mathbf{X} = \begin{bmatrix} x_{11} & \dots & x_{1p} \\ \vdots & \vdots & \vdots \\ x_{n1} & \dots & x_{np} \end{bmatrix}$$

- Asumimos

$$= \mathbf{X} \, .$$

- ¿Cómo estimamos ?

Planteamiento

Estimadores mínimo cuadráticos

- Una opción clásica son los mínimos cuadrados en donde pretendemos que

$$\|\mathbf{y} - \hat{}\|^2$$

- Lo que estamos haciendo es sustituir el vector de medias desconocido por los valores observados.
- Consideramos la función

$$L(\ ) = \sum_{i=1}^{n} (y_i - \mu_i)^2 = \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 .$$

- Los estimadores mínimo cuadráticos minimizan la función $L(\ )$.
-

$$\hat{} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}.$$

Predicciones y la matriz H

- Las medias estimadas las obtenemos con

$$\hat{} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}.$$

- Si denotamos

$$\mathbf{H} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T,$$

  entonces

$$\hat{} = \mathbf{H}\mathbf{y}.$$

-

$$E[\hat{}] = \ .$$

-

$$var[\hat{}] = \sigma^2(\mathbf{X}^T\mathbf{X})^{-1},$$

Residuos y estimación de la varianza

- Los residuos son
$$\mathbf{e} = \mathbf{y} - \mathbf{X}\hat{}.$$
- Se tiene
$$E\sum_{i=1}^{n}\frac{(Y_i - \hat{\mu}_i)^2}{n-p} = \sigma^2.$$

tamidata2::gse25171

Con `time` y `Pi`

- Analizamos los datos correspondientes a la sonda `261892_at`.
- Consideramos las variables fenotípicas `time` y `Pi`.

```
pacman::p_load(Biobase)
data(gse25171,package="tamidata2")
head(pData(gse25171),n=2)
```

```
                time time2        Pi replication
GSM618324.CEL.gz    0 Short Treatment           1
GSM618325.CEL.gz    0 Short   Control           2
```

```
sel0 = which("261892_at"==fData(gse25171)[,"PROBEID"])
df0 = data.frame(pData(gse25171)[,c("time","Pi")],
                 expression=exprs(gse25171)[sel0,])
```

- Ajustamos un modelo con dos variables predictoras.

```
fit4 = lm(expression~ time + Pi,data=df0)
```

- La matriz modelo es

```
head(model.matrix(fit4),n=5)
```

```
                (Intercept) time PiTreatment
GSM618324.CEL.gz           1    0           1
GSM618325.CEL.gz           1    0           0
GSM618326.CEL.gz           1    1           1
GSM618327.CEL.gz           1    1           0
GSM618328.CEL.gz           1    6           1
```

- Los coeficientes los tenemos con

```
coef(fit4)
```

```
(Intercept)        time PiTreatment
  8.6293898  -0.1330703  -1.3219071
```

- Los residuos los tenemos con

```
resid(fit4)
```

```
GSM618324.CEL.gz GSM618325.CEL.gz GSM618326.CEL.gz GSM618327.CEL.gz
      1.01552765       0.21676116      -0.32266386      -0.07784405
GSM618328.CEL.gz GSM618329.CEL.gz GSM618330.CEL.gz GSM618331.CEL.gz
     -0.59106601       0.31824015       0.22638066      -0.30489608
GSM618332.CEL.gz GSM618333.CEL.gz GSM618334.CEL.gz GSM618335.CEL.gz
      0.49588789      -0.15067778      -0.63263328       0.15231308
GSM618336.CEL.gz GSM618337.CEL.gz GSM618338.CEL.gz GSM618339.CEL.gz
     -0.93990720       0.45816343       0.21916840      -0.92275255
GSM618340.CEL.gz GSM618341.CEL.gz GSM618342.CEL.gz GSM618343.CEL.gz
      0.24175878       0.15101414      -0.46350659       0.52297219
GSM618344.CEL.gz GSM618345.CEL.gz GSM618346.CEL.gz GSM618347.CEL.gz
     -0.39355515      -0.42938559       1.14460870       0.06609189
```

- El error estándar residual lo tenemos con

```
summary(fit4)$sigma
```

```
[1] 0.5644856
```

Con `time2` y `Pi`

- Analizamos los datos correspondientes a la sonda `261892_at`.
- Consideramos las variables fenotípicas `time2` y `Pi`.

```
pacman::p_load(Biobase)
data(gse25171,package="tamidata2")
head(pData(gse25171),n=2)
```

```
                time time2        Pi replication
GSM618324.CEL.gz    0 Short Treatment           1
GSM618325.CEL.gz    0 Short   Control           2
```

```
sel0 = which("261892_at"==fData(gse25171)[,"PROBEID"])
df0 = data.frame(pData(gse25171)[,c("time2","Pi")],
                 expression=exprs(gse25171)[sel0,])
```

- Ajustamos un modelo con dos variables predictoras.

```
fit5 = aov(expression~ time2 + Pi,data=df0)
```

- La matriz modelo es

```
head(model.matrix(fit5),n=5)
```

```
                (Intercept) time2Medium PiTreatment
GSM618324.CEL.gz           1           0           1
GSM618325.CEL.gz           1           0           0
GSM618326.CEL.gz           1           0           1
GSM618327.CEL.gz           1           0           0
GSM618328.CEL.gz           1           1           1
```

- La tabla de análisis de la varianza la tenemos con

```
summary(aov(fit5))
```

```
            Df Sum Sq Mean Sq F value   Pr(>F)
time2        1  26.99  26.992   29.36 2.24e-05 ***
Pi           1  10.48  10.485   11.41  0.00285 **
Residuals   21  19.30   0.919
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- ¿Y un modelo con interacción?

```
fit6 = aov(expression~ time2 * Pi,data=df0)
```

- Vemos la tabla anova.

```
summary(fit6)
```

```
            Df Sum Sq Mean Sq F value   Pr(>F)
time2        1 26.992  26.992   28.02 3.51e-05 ***
Pi           1 10.485  10.485   10.88  0.00358 **
time2:Pi     1  0.038   0.038    0.04  0.84370
Residuals   20 19.264   0.963
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- No parece que tengamos interacción.

Muchos modelos

Un modelo por fila con `GSEAlm`

- Hemos elegido trabajar con una sonda elegida de un modo arbitrario.
- Tenemos interés en todas las sondas.
- Hemos de ajustar un modelo lineal para cada una de estas sondas.
- Los predictores serán variables fenotípicas compartidas por todas las sondas.
- Los distintos modelos comparten los predictores y difieren en la variable respuesta.

```
pacman::p_load(GSEAlm)
data(gse25171,package="tamidata2")
fits1 = GSEAlm::lmPerGene(gse25171,~ time + Pi)
```

- La matriz de modelo común a todos los ajustes la tenemos con

```
head(fits1$x)
```

```
                (Intercept) time PiTreatment
GSM618324.CEL.gz           1    0           1
GSM618325.CEL.gz           1    0           0
GSM618326.CEL.gz           1    1           1
GSM618327.CEL.gz           1    1           0
GSM618328.CEL.gz           1    6           1
GSM618329.CEL.gz           1    6           0
```

- Los coeficientes de cada ajuste con

```
fits1$coefficients
```

- Para la primera sonda tenemos los coeficientes

```
fits1$coefficients[,1]
```

```
 (Intercept)         time  PiTreatment
5.086290213 -0.004315513 -0.116650336
```

- Las varianzas del error estimadas son

```
fits1$sigmaSqr
```

Y los estadísticos correspondientes a los tests de coeficientes nulos con

```
fits1$tstat
```

Un modelo por fila con `limma`

```
pacman::p_load(limma)
```

- Ajustamos los modelos.

```
design = model.matrix(~ pData(gse25171)[,"time"] +  pData(gse25171)[,"Pi"])
fits2 = limma::lmFit(gse25171,design)
```

- Los coeficientes los tenemos con

```
head(fits2$coefficients,n=2)
```

```
            (Intercept) pData(gse25171)[, "time"]
244919_at      5.086290              -0.004315513
244920_s_at    8.371483              -0.004977996
            pData(gse25171)[, "Pi"]Treatment
244919_at                         -0.11665034
244920_s_at                       -0.08573999
```

- Los coeficientes del ajuste para la primera sonda son

```
fits2$coefficients[1,]
```

```
                     (Intercept)         pData(gse25171)[, "time"]
                     5.086290213                      -0.004315513
pData(gse25171)[, "Pi"]Treatment
                    -0.116650336
```