Clasificación

Estadística (M)

Clasificación

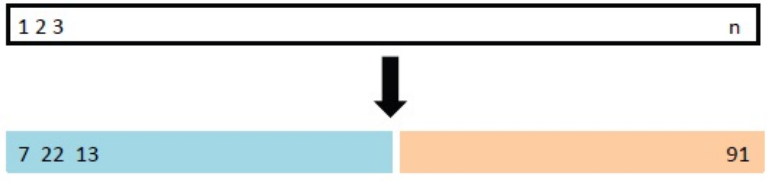
- Tarjeta de Crédito: clientes morosos o clientes pagadores
  - p_1 es la proporción de morosos
  - $p_2 = 1 - p_1$ de pagadores.

- Estamos interesados en determinar de qué tipo de cliente se trata en función del balance bancario mensual.
  - en morosos el balance es una v.a con distribución F_1
  - en pagadores el balance es una v.a con distribución F_2

La idea es que a partir de una muestra de entrenamiento aprenderemos a clasificar a los individuos, de manera de que al tener que clasificar a un nuevo cliente podamos hacerlo con la regla de clasificación aprendida. La eficacia de la regla se pone a prueba con una muestra de validación o testeo.

Validación Cruzada

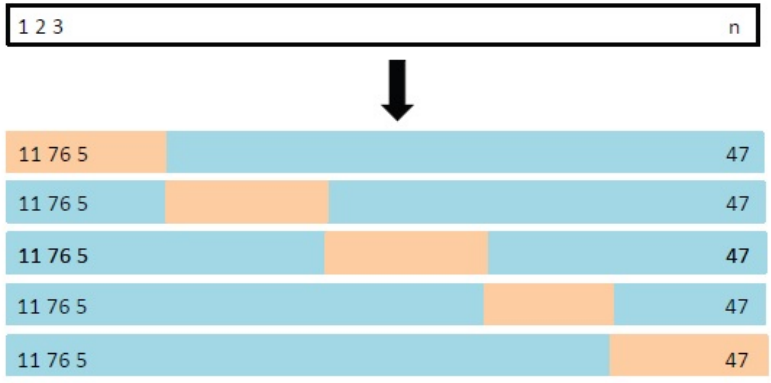
- Validación Cruzada o Cross–Validation: dividir al azar en 2 partes o folds a los datos: muestra de *validación* y muestra de *entrenamiento*. Con la muestra de *entrenamiento* se *entrena* a la regla, es decir se realizan todas las estimaciones para construir la regla. Se valida la regla construida con el primer fold computando el error de clasificación en la muestra de *validación*.



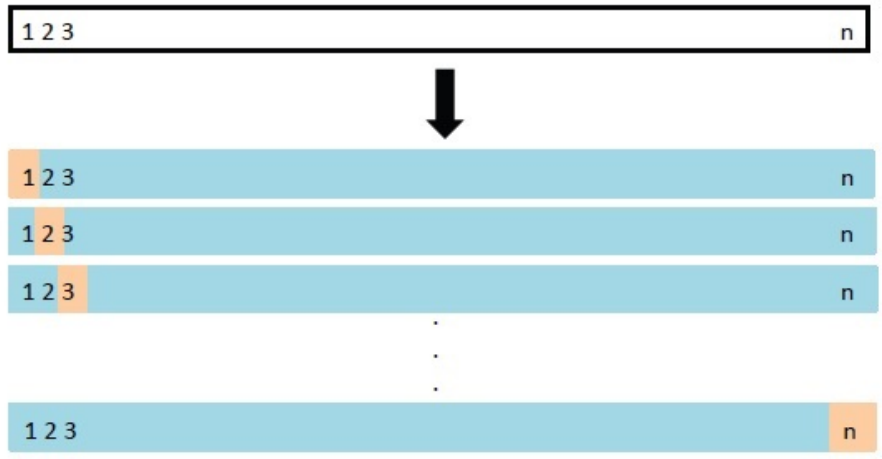
Validación Cruzada

- $k-$fold Cross–Validation: misma idea dividiendo en K partes, o sea, dividir al azar en K partes o folds a los datos, se separa al primer fold, se *entrena* a la regla con las $K-1$ muestras restantes y se valida la regla con el primer fold computando el error de clasificación. Luego se repite tomando como muestra j de validación a cada uno de los folds y se computa el error de clasificación.

  Finalmente, se estima la tasa de error de clasificación promediando todos los errores estimados en cada fold.



Representación esquemática



En general...

Supongamos que disponemos de un conjunto de elementos que pueden venir de dos o más poblaciones distintas.

En cada elemento se observan p variables aleatorias:
$\mathbf{X} = (X_1, X_2, \ldots, X_p)^t$
Por ejemplo: registramos altura, ancho, peso, etc.

Se desea clasificar un nuevo elemento, con valores de las variables conocidas, a alguna de las poblaciones en estudio.

Clasificación

- X v.a. discreta $X = (x_1, x_2, \ldots, x_p) \in \mathcal{X}$
- Posibles *etiquetas*. Caso binario $\mathcal{Y} = \{0, 1\}$
- Clasificador: Regla que asigna a cada $x \in \mathcal{X}$ un elemento $y \in \mathcal{Y}$

Clasificación

- X v.a. discreta $X = (x_1, x_2, \ldots, x_p) \in \mathcal{X}$
- Posibles *etiquetas*. Caso binario $\mathcal{Y} = \{0, 1\}$
- Clasificador: Regla que asigna a cada $x \in \mathcal{X}$ un elemento $y \in \mathcal{Y}$
- H^{op} Optimo: Regla de Bayes - Caso binario

Regla Optima de Bayes

- X v.a. discreta $X = \{x_1, x_2, \ldots, x_p\} \in \mathcal{X}$
- Posibles *etiquetas*. Caso binario $\mathcal{Y} = \{0, 1\}$
- Clasificador: Regla que asigna a cada $x \in \mathcal{X}$ un elemento $y \in \mathcal{Y}$
- H^{op} Optimo: Regla de Bayes - Caso binario

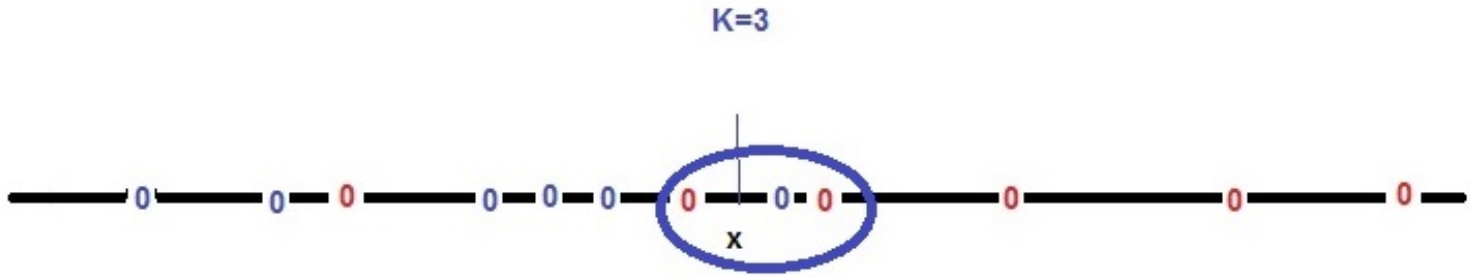
$$H^{op}(x) = \begin{cases} 1 & \text{si} \quad \mathbb{P}(Y = 1 \mid X = x) > \mathbb{P}(Y = 0 \mid X = x), \\ 0 & \text{si} \quad \mathbb{P}(Y = 0 \mid X = x) > \mathbb{P}(Y = 1 \mid X = x). \end{cases}$$

Regla Optima de Bayes

- X v.a. discreta $X = \{x_1, x_2, \ldots, x_p\} \in \mathcal{X}$
- Posibles *etiquetas*. Caso binario $\mathcal{Y} = \{0, 1\}$
- Clasificador: Regla que asigna a cada $x \in \mathcal{X}$ un elemento $y \in \mathcal{Y}$
- H^{op} Optimo: Regla de Bayes - Caso binario

$$H^{op}(x) = \begin{cases} 1 & \text{si} \quad \mathbb{P}(Y = 1 \mid X = x) > \mathbb{P}(Y = 0 \mid X = x), \\ 0 & \text{si} \quad \mathbb{P}(Y = 0 \mid X = x) > \mathbb{P}(Y = 1 \mid X = x). \end{cases}$$

¿Cómo podríamos estimar $\mathbb{P}(Y = 1 \mid X = x)$ y $\mathbb{P}(Y = 0 \mid X = x)$?


K=3
0
0
0
0
0
0
0
0
0
0
0
0
0
x

$k-$Vecinos más cercanos (kNN: k-nearest neighbors)

El método de $k-$Vecinos más cercanos es uno de los métodos existentes para estimar la distribución condicional de Y dado X y para después clasificar una observación en la clase con la mayor probabilidad estimada.

- Elegimos k un entero positivo y un punto x para clasificar.
- El clasificador kNN identifica el conjunto de los k puntos más cercanos a x. Sea N_x dicho conjunto.
- Estima a $P(Y = 1 \mid X = x)$ por la fracción de puntos en N_x cuya etiqueta es igual a 1:

$$\widehat{\mathbb{P}}(Y = 1 \mid X = x) = \frac{1}{k} \sum_{i \in N_0} \mathcal{I}(y_i = 1)$$

- Análogamente estimamos $P(Y = 0 \mid X = x)$

El parámetro k de este método puede elegirse por Convalizadión Cruzada.

Otro Enfoque: Regresión Logística

La regla de óptima de Bayes depende de:

$$p(\mathbf{x}) = \mathbb{P}(Y = 1 \mid \mathbf{X} = \mathbf{x})$$

$\mathbf{X} = (X_1, \ldots, X_p)^t$ es un vector de p covariables.

¿Y si modelamos esta probabilidad en función de $\mathbf{X}$?

Otro Enfoque: Regresión Logística

La regla de óptima de Bayes depende de:

$$p(\mathbf{x}) = \mathbb{P}(Y = 1 \mid \mathbf{X} = \mathbf{x})$$

$\mathbf{X} = (X_1, \ldots, X_p)^t$ es un vector de p covariables.

¿Y si modelamos esta probabilidad en función de $\mathbf{X}$?

Tenemos que tener algunos cuidados...

Regresión Logística

Pensemos que tenemos una sola variable X
Notemos que

$$0 \quad \leq \quad p(x) \leq 1$$

Regresión Logística

Pensemos que tenemos una sola variable X
Notemos que

$$0 \quad \leq \quad p(x) \leq 1$$

Sin embargo, los **odds**

$$\frac{p(x)}{1-p(x)} \geq 0$$

Regresión Logística

Pensemos que tenemos una sola variable X
Notemos que

$$0 \quad \leq \quad p(x) \leq 1$$

Sin embargo, los **odds**

$$\frac{p(x)}{1-p(x)} \geq 0$$

Tomando logaritmo:

$$-\infty \quad < \quad \log\left(\frac{p(x)}{1-p(x)}\right) < \infty$$

Podríamos modelar:

$$\log\left(\frac{p(x)}{1-p(x)}\right) \quad = \quad \beta_1 + \beta_2\, x$$

Regresión Logística

Pensemos que tenemos una sola variable X
Notemos que

$$0 \quad \leq \quad p(x) \leq 1$$

Sin embargo, los **odds**

$$\frac{p(x)}{1-p(x)} \geq 0$$

Tomando logaritmo:

$$-\infty \quad < \quad \log\left(\frac{p(x)}{1-p(x)}\right) < \infty$$

Podríamos modelar:

$$logit(p(x)) = \log\left(\frac{p(x)}{1-p(x)}\right) \quad = \quad \beta_1 + \beta_2\, x$$

Regresión Logística

Haciendo el camino inverso:

$$\frac{p(x)}{1-p(x)} = e^{\beta_1+\beta_2\ x}$$

$$p(x) = p(x, \boldsymbol{\beta}) = \frac{e^{\beta_1+\beta_2\ x}}{1+e^{\beta_1+\beta_2\ x}}$$

Regresión Logística

Haciendo el camino inverso:

$$\frac{p(x)}{1-p(x)} = e^{\beta_1+\beta_2\ x}$$

$$p(x) = p(x, \boldsymbol{\beta}) = \frac{e^{\beta_1+\beta_2\ x}}{1+e^{\beta_1+\beta_2\ x}} = \frac{1}{1+e^{-\beta_1-\beta_2\ x}}$$

Regresión Logística

Con una sola variable

$$p(x) = p(x, \boldsymbol{\beta}) \quad = \quad \frac{e^{\beta_1 + \beta_2\ x}}{1 + e^{\beta_1 + \beta_2\ x}}$$

Regresión Logística

Con una sola variable

$$p(x) = p(x, \boldsymbol{\beta}) \quad = \quad \frac{e^{\beta_1 + \beta_2\ x}}{1 + e^{\beta_1 + \beta_2\ x}}$$

Modelo de regresión logística:

$$\mathbb{P}(Y = 1 \mid \mathbf{X} = \mathbf{x}) = \frac{e^{\beta_1 x_1 + \cdots + \beta_p x_p}}{1 + e^{\beta_1 x_1 + \cdots + \beta_p x_p}} \tag{1}$$

Regresión Logística

El estimador clásico en este contexto se obtiene por el método de máxima verosimilitud, es decir hallando β_1, $\beta_2, \ldots,$ β_p que maximizan

$$L(\boldsymbol{\beta}; y_1, \ldots, y_n, \mathbf{x}_1, \ldots, \mathbf{x}_n) = \prod_{i=1}^{n} p(\mathbf{x}_i, \boldsymbol{\beta})^{y_i} (1 - p(\mathbf{x}_i, \boldsymbol{\beta}))^{1-y_i}$$

Regresión Logística

El estimador clásico en este contexto se obtiene por el método de máxima verosimilitud, es decir hallando β_1, $\beta_2, \ldots,$ β_p que maximizan

$$L(\boldsymbol{\beta}; y_1, \ldots, y_n, \mathbf{x}_1, \ldots, \mathbf{x}_n) = \prod_{i=1}^{n} p(\mathbf{x}_i, \boldsymbol{\beta})^{y_i} (1 - p(\mathbf{x}_i, \boldsymbol{\beta}))^{1-y_i}$$

Este estimador no tiene una expresión analítica. Se halla derivando e igualando a 0 la log–verosimilitud y resolviendo numéricamente la ecuación por el método de Newton-Raphson o Fisher-scoring.

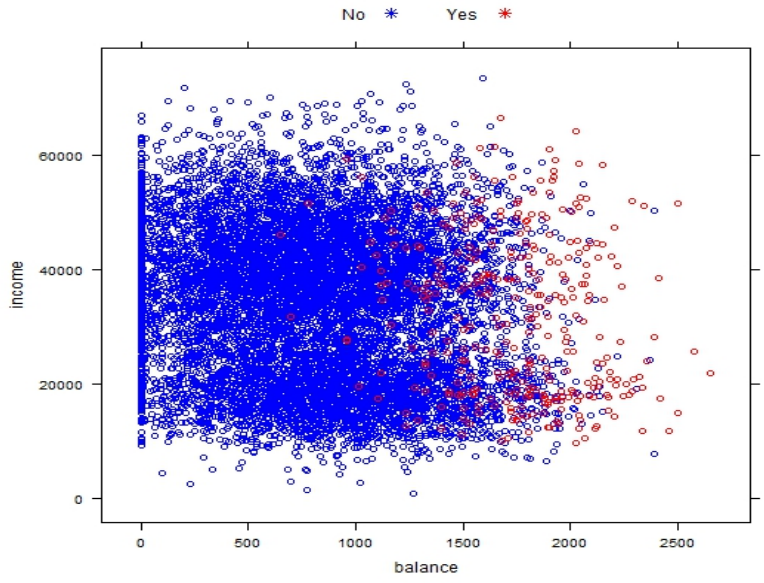
Vayamos a un ejemplo: datos de default

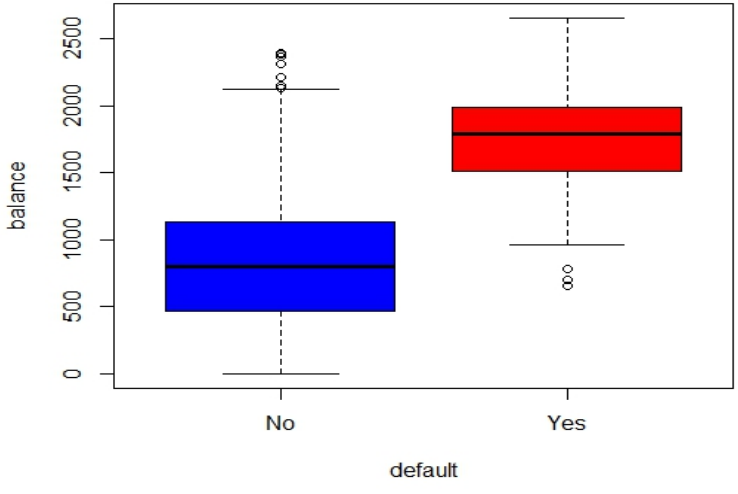
Variables:

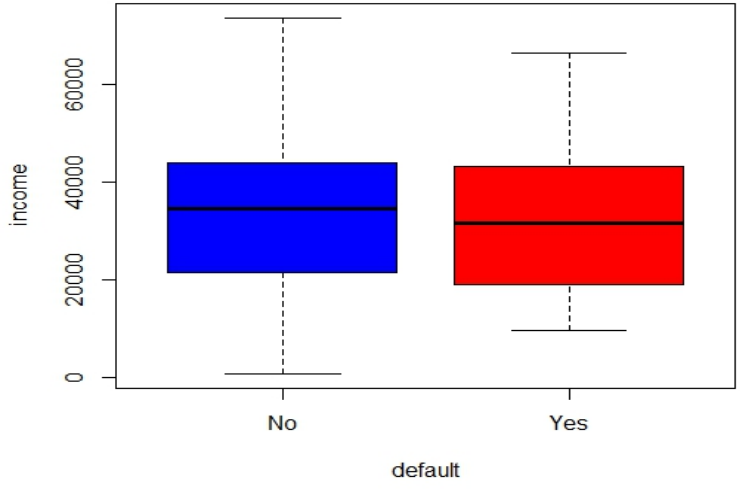
- **default** (yes o no)
- **student** (yes o no)
- **balance**: balance mensual de la tarjeta
- **income**: ingreso anual

```
library(ISLR)
attach(Default)
names(Default)

library(lattice)

xyplot(income~balance,groups=default, type="p", col=c("blue","red"),
key=list(columns=2,text=list(levels(default)),
points=list(col=c("blue","red"))))

plot(balance~default,col=c("blue","red"))
plot(income~default,col=c("blue","red"))
```

Gráficos de datos de default







Gráficos de datos de default

Llamemos

$$p(balance) = \mathbb{P}(default = Yes \mid balance)$$

Ajustamos el modelo

$$logit(p(balance_i)) = \beta_1 + \beta_2\, balance_i$$

```
proba.hat=glm(default~balance,family=binomial)$fitted

default01=1*(default=="Yes")

plot(balance,default01)
points(balance,proba.hat,col="blue")
```