

**2020-2021**

## **Aprendizaje Automático**

### **3. Técnicas de optimización**



Francisco Casacuberta Nolla  
(fcn@dsic.upv.es)

Enrique Vidal Ruiz  
(evidal@dsic.upv.es)

Departament de Sistemes Informàtics i Computació (DSIC)

Universitat Politècnica de València (UPV)

# Index

- 1 Introducción ▷ 2
- 2 Optimización analítica: gradiente ▷ 6
- 3 Optimización con restricciones: multiplicadores de Lagrange y teorema Kuhn-Tucker ▷ 11
- 4 Técnicas de descenso por gradiente ▷ 22
- 5 Esperanza-Maximización (EM) ▷ 33
- 6 Notación ▷ 53

# Index

- 1 *Introducción* ▷ 2
- 2 Optimización analítica: gradiente ▷ 6
- 3 Optimización con restricciones: multiplicadores de Lagrange y teorema Kuhn-Tucker ▷ 11
- 4 Técnicas de descenso por gradiente ▷ 22
- 5 Esperanza-Maximización (EM) ▷ 33
- 6 Notación ▷ 53

# Clasificación, regresión y optimización

- Los modelos están parametrizados por un vector de parámetros  $\Theta$ ; es decir,  $\mathcal{F} = \{f_{\Theta} : \mathcal{X} \rightarrow \mathcal{Y}, \Theta \in \mathbb{R}^D\}$
- Clasificación:  $f_{\Theta} : \mathcal{X} \rightarrow \{1, \dots, C\}$ . En muchos problemas  $\mathcal{X} \equiv \mathbb{R}^d$
- Regresión:  $f_{\Theta} : \mathcal{X} \rightarrow \mathcal{Y}$ . Típicamente  $\mathcal{X} \equiv \mathbb{R}^d$  y  $\mathcal{Y} \equiv \mathbb{R}$ .
- Dos etapas:
  - **Aprendizaje**: Dado  $S \subset \mathcal{X} \times \mathcal{Y}$ , estimar  $\hat{\Theta} \in \mathbb{R}^D$   
Técnicas de optimización para aprendizaje:
    - \* Optimización analítica.
    - \* Optimización con restricciones: Multiplicadores de Lagrange.
    - \* Descenso (ascenso) por gradiente.
    - \* Optimización probabilística: Algoritmo EM.
  - **Búsqueda o inferencia**: Dados  $\Theta$  y  $x \in \mathcal{X}$ , estimar  $\hat{y} = f_{\Theta}(x)$ .  
Técnicas de optimización para búsqueda:
    - \* Exhaustiva: por ejemplo, clasificación, si  $C \ll$ .
    - \* Programación dinámica: algoritmo de Viterbi con modelos ocultos de Markov.
    - \* Inteligente: ramificación y poda,  $A^*$ , etc.

# Optimización y aprendizaje automático

- Datos:
  - $N$  muestras de aprendizaje:  
 $S = \{(x_1, y_1), \dots, (x_N, y_N)\}, (x_n, y_n) \in \mathcal{X} \times \mathcal{Y}, 1 \leq n \leq N,$
  - un clasificador o regresor,  $f_{\Theta} : \mathcal{X} \rightarrow \mathcal{Y}$ , parametrizado por  $\Theta \in \mathbb{R}^D$ ,
  - un criterio aprendizaje definido por una función objetivo,  $q_S : \mathbb{R}^D \rightarrow \mathbb{R}$
- estimar  $\hat{\Theta}$  mediante optimización de  $q_S$ ; es decir:

$$\hat{\Theta} \equiv \Theta^* = \arg \min_{\Theta} q_S(\Theta)$$

o bien:

$$\hat{\Theta} \equiv \Theta^* = \arg \max_{\Theta} q_S(\Theta)$$

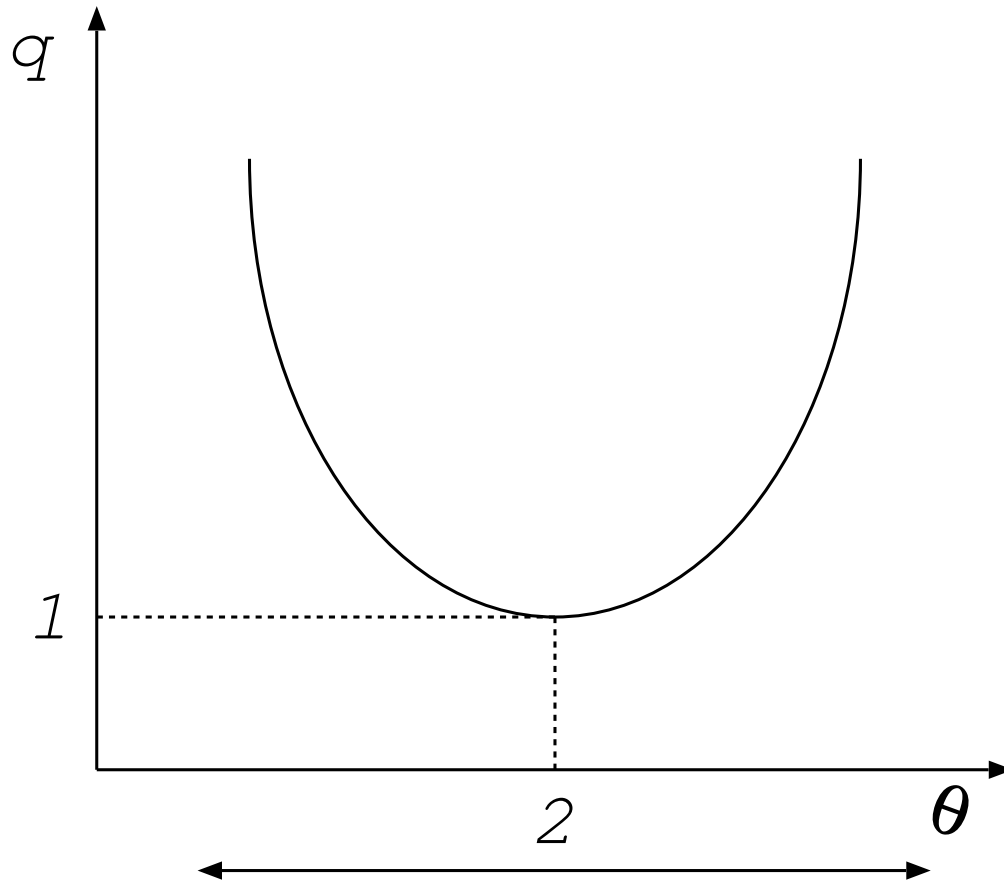
## Técnicas generales de AA basadas en optimización

- $f_{\Theta}$  es una función cualquiera que depende de un vector de parámetros  $\Theta$ :  $q_S(\Theta)$  se basa en *funciones de error* y típicamente su optimización utiliza técnicas de *descenso/ascenso por gradiente* (caso particular de “hill-climbing”).
- $f_{\Theta}$  es una función cualquiera dependiente de  $\Theta$ , pero hay ciertas restricciones en los valores posibles de los parámetros  $\Theta$ : *optimización con restricciones* de  $q_S(\Theta)$  mediante la técnica de los *multiplicadores de Lagrange*.
- $f_{\Theta}$  se basa en distribuciones (o densidades) de probabilidad: estimación de *máxima verosimilitud*. Frecuentemente hay restricciones en los parámetros a estimar y se requiere el uso de *multiplicadores de Lagrange*.
- $f_{\Theta}$  se basa en distribuciones (o densidades) de probabilidad, pero hay variables aleatorias “*latentes*” u “*ocultas*”: La estimación de *máxima verosimilitud* generalmente requiere una técnica de optimización llamada *esperanza-maximización* (EM).

# Index

- 1 Introducción ▷ 2
- 2 *Optimización analítica: gradiente* ▷ 6
- 3 Optimización con restricciones: multiplicadores de Lagrange y teorema Kuhn-Tucker ▷ 11
- 4 Técnicas de descenso por gradiente ▷ 22
- 5 Esperanza-Maximización (EM) ▷ 33
- 6 Notación ▷ 53

# Optimización analítica: ejemplo



- Dado  $q(\theta) = 1 + (\theta - 2)^2$
- Calcular  $\theta^* = \arg \min_{\theta \in \mathbb{R}} q(\theta)$
- Procedimiento:  $\frac{d q(\theta)}{d \theta} = 2 (\theta - 2) = 0$
- Solución:  $\theta^* = 2$



## Optimización analítica: gradiente

- Dada una función *convexa*  $q : \mathbb{R}^D \rightarrow \mathbb{R}$ , calcular  $\arg \min_{\Theta \in \mathbb{R}^D} q(\Theta)$
- Procedimiento:
  1. Calcular el gradiente de  $q$ :  $\nabla q(\Theta) \stackrel{\text{def}}{=} \left( \frac{\partial q(\Theta)}{\partial \Theta_1}, \dots, \frac{\partial q(\Theta)}{\partial \Theta_D} \right)^t$
  2. Resolver  $\nabla q(\Theta) = \mathbf{0}$ ;  
 es decir, resolver el sistema de ecuaciones  $\frac{\partial q(\Theta)}{\partial \Theta_i} = 0, \quad 1 \leq i \leq D$ .  
 Sean  $\Theta_1^*, \dots, \Theta_D^*$  las soluciones obtenidas.
- Solución:  $\Theta^* = (\Theta_1^*, \dots, \Theta_D^*)^t$
- Si  $q$  es convexa,  $\nabla q(\Theta^*) = \mathbf{0}$  es una condición *necesaria y suficiente* para que  $\Theta^*$  sea (la única) solución.

### Ejercicios:

- Encontrar el vector  $\theta^* \in \mathbb{R}^2$  que minimiza la función  $q(\theta) = (\theta_1 - 1)^2 + (\theta_2 - 2)^2$
- Encontrar el vector  $\theta^* \in \mathbb{R}^2$  que minimiza la función  $q(\theta) = (\theta_1 - 1)^2 + (\theta_2 - 2)^2 + \theta_1 \theta_2$
- ¿Qué ocurre si  $q$  no es convexa? Ejemplos:  $\frac{1}{10} x^3 + 20 x^2, \quad x^4 - 10 x^2$

## Optimización analítica: otro ejemplo simple

- Estimar los parámetros<sup>†</sup>  $\Theta \equiv (\mu, \sigma)$  de una gaussiana univariada (en  $\mathbb{R}^1$ ):

$$p(x; \Theta) \stackrel{\text{def}}{=} p(x; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right)$$

- Logaritmo de la verosimilitud de una muestra  $S = \{x_1, \dots, x_N\}$ :

$$\begin{aligned} q_S(\Theta) &\equiv L_S(\mu, \sigma) = \log \prod_{n=1}^N p(x_n; \mu, \sigma) = \sum_{n=1}^N \log p(x_n; \mu, \sigma) \\ &= N \log \frac{1}{\sigma\sqrt{2\pi}} - \frac{1}{2\sigma^2} \sum_{n=1}^N (x_n - \mu)^2 \end{aligned}$$

- Para una estimación de máxima verosimilitud basta hacer  $\nabla L_S(\Theta) = \mathbf{0}$ . En nuestro caso unidimensional ([ejercicio](#)):

$$\frac{\partial L_S(\mu, \sigma)}{\partial \mu} = 0 \Rightarrow \hat{\mu} = \frac{1}{N} \sum_{n=1}^N x_n; \quad \frac{\partial L_S(\mu, \sigma)}{\partial \sigma} = 0 \Rightarrow \hat{\sigma}^2 = \frac{1}{N} \sum_{n=1}^N (x_n - \hat{\mu})^2$$

<sup>†</sup> media y desviación típica

## Optimización analítica: otro ejemplo

- Estimar los parámetros de una gaussiana multivariada (en  $\mathbb{R}^D$ ), con  $\Sigma$  dada:

$$p(\mathbf{x}; \Theta) = (2\pi)^{-D/2} |\Sigma|^{-1/2} \exp \left( -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^t \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right)$$

donde  $\Theta \equiv (\boldsymbol{\mu}, \Sigma)$ . Si  $\Sigma$  está prefijada, entonces  $\Theta \equiv \boldsymbol{\mu} \in \mathbb{R}^D$

- Logaritmo de la verosimilitud de una muestra  $S = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ :

$$\begin{aligned} q_S(\Theta) \equiv L_S(\Theta) &= \log \prod_{n=1}^N p(\mathbf{x}_n; \Theta) = \sum_{n=1}^N \log p(\mathbf{x}_n; \Theta) \\ &= N \log \left( (2\pi)^{-D/2} |\Sigma|^{-1/2} \right) - \frac{1}{2} \sum_{n=1}^N (\mathbf{x}_n - \boldsymbol{\mu})^t \Sigma^{-1} (\mathbf{x}_n - \boldsymbol{\mu}) \end{aligned}$$

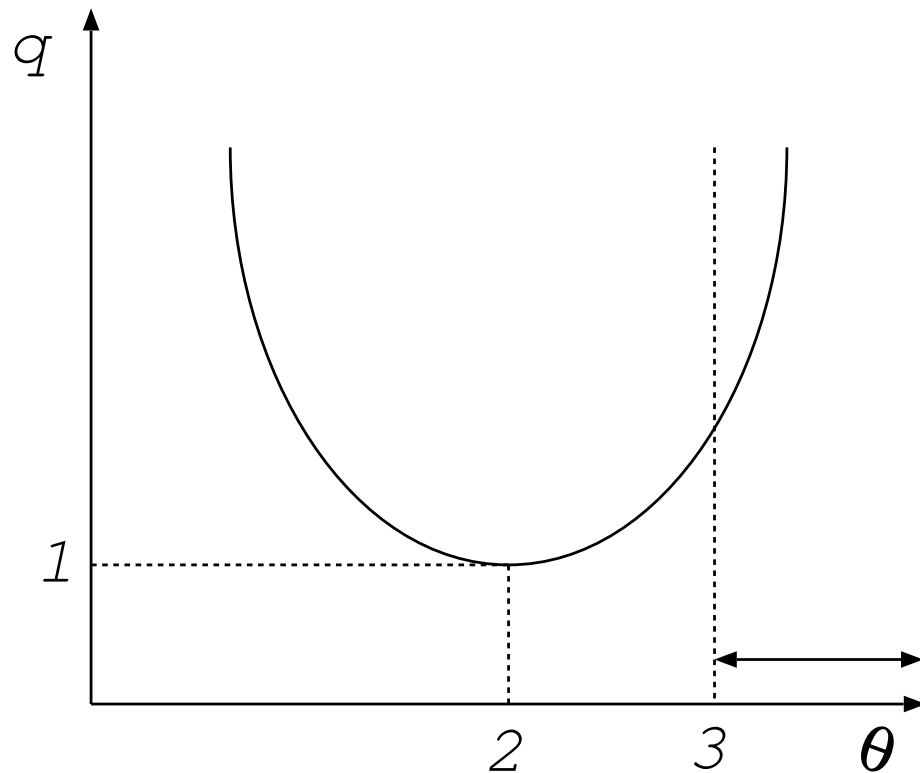
- Para una estimación de máxima verosimilitud basta hacer  $\nabla L_S(\Theta) = \mathbf{0}$ . Si  $\Sigma$  está prefijada, se obtiene (*[ejercicio](#)*):

$$\hat{\boldsymbol{\mu}} = \frac{1}{N} \sum_{n=1}^N \mathbf{x}_n$$

# Index

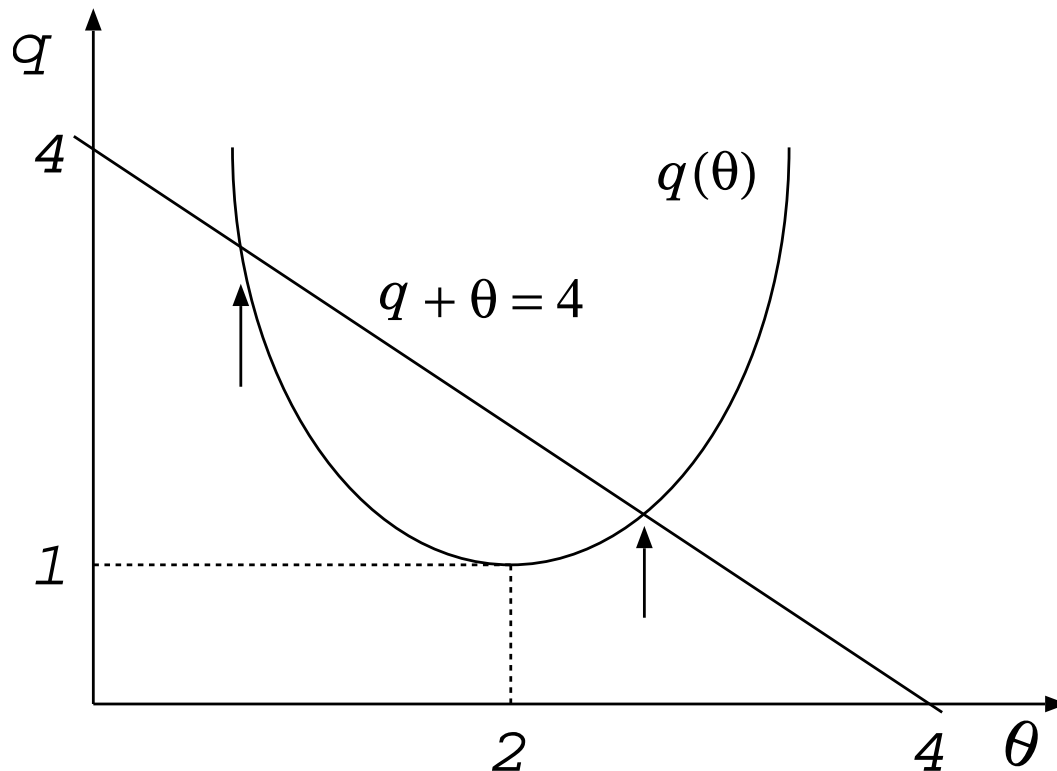
- 1 Introducción ▷ 2
- 2 Optimización analítica: gradiente ▷ 6
- 3 *Optimización con restricciones: multiplicadores de Lagrange y teorema Kuhn-Tucker* ▷ 11
- 4 Técnicas de descenso por gradiente ▷ 22
- 5 Esperanza-Maximización (EM) ▷ 33
- 6 Notación ▷ 53

# Optimización con restricciones: ejemplo simple 1



- Dado:  $q(\theta) = 1 + (\theta - 2)^2$ ,
- Calcular:  $\theta^* = \arg \min_{\theta: \theta \geq 3} q(\theta)$   
(restricción de desigualdad:  $\theta - 3 \geq 0$ )
- Solución: ??

## Optimización con restricciones: ejemplo simple 2



- Dado:  $q(\theta) = 1 + (\theta - 2)^2$ ,
- Calcular:  $\theta^* = \arg \min q(\theta)$   
 $\theta: q + \theta = 4$   
(restricción de igualdad:  $q + \theta - 4 = 0$ )
- Solución: ??

# Optimización con restricciones: multiplicadores de Lagrange

Consideremos un problema de optimización definido por:

$$\begin{array}{lll} \text{minimizar} & q(\Theta) & \Theta \in \mathbb{R}^D \\ \text{sujeto a} & v_i(\Theta) \geq 0 & 1 \leq i \leq k \\ & u_i(\Theta) = 0 & 1 \leq i \leq m \end{array}$$

donde  $q$  es una función convexa y  $v_i, u_i$  son funciones que expresan restricciones. Equivalentemente, el problema consiste en calcular:

$$\begin{aligned} \Theta^* = \arg \min_{\substack{\Theta \in \mathbb{R}^D \\ v_i(\Theta) \geq 0, 1 \leq i \leq k \\ u_i(\Theta) = 0, 1 \leq i \leq m}} q(\Theta) \end{aligned}$$

Para resolver este problema, se define la *función Lagrangiana*:

$$\Lambda(\Theta, \alpha, \beta) \stackrel{\text{def}}{=} q(\Theta) - \sum_{i=1}^k \alpha_i v_i(\Theta) + \sum_{i=1}^m \beta_i u_i(\Theta)$$

donde  $\alpha_i \geq 0, 1 \leq i \leq k$  y  $\beta_i, 1 \leq i \leq m$ , son los *multiplicadores de Lagrange*.

# La técnica de los multiplicadores de Lagrange

1. *Definir multiplicadores de Lagrange y Lagrangiana:*

$$\Lambda(\Theta, \alpha, \beta) \stackrel{\text{def}}{=} q(\Theta) - \sum_{i=1}^k \alpha_i v_i(\Theta) + \sum_{i=1}^m \beta_i u_i(\Theta)$$

2. *Obtener el minimizador  $\Theta^*$  de la Lagrangiana  $\Lambda(\Theta, \alpha, \beta)$ , en función de  $\alpha, \beta$  (resolviendo  $\nabla_{\Theta} \Lambda(\Theta, \alpha, \beta) = 0$ ):*

$$\Theta^*(\alpha, \beta) = \arg \min_{\Theta} \Lambda(\Theta, \alpha, \beta)$$

3. *Obtener la función dual de Lagrange* (sustituir  $\Theta$  por  $\Theta^*(\alpha, \beta)$  en  $\Lambda(\Theta, \alpha, \beta)$ ):

$$\Lambda_D(\alpha, \beta) \stackrel{\text{def}}{=} \Lambda(\Theta^*(\alpha, \beta), \alpha, \beta)$$

4. *Optimizar la función dual de Lagrange* (usualmente resolviendo  $\nabla \Lambda_D(\alpha, \beta) = 0$ ):

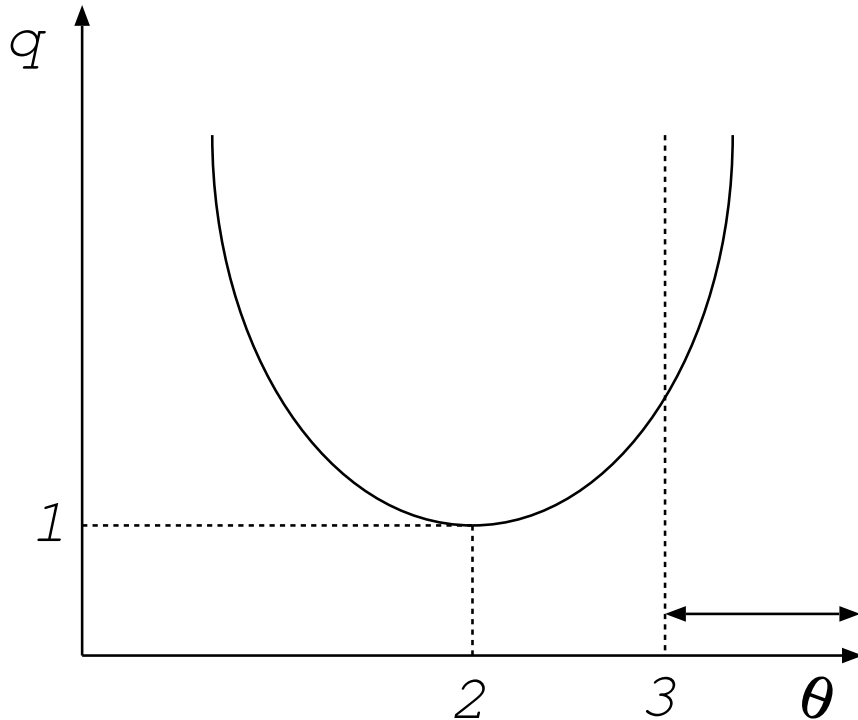
$$(\alpha^*, \beta^*) = \arg \max_{\alpha, \beta: \alpha_i \geq 0} \Lambda_D(\alpha, \beta)$$

5. *Solución final:*

$$\Theta^* = \Theta^*(\alpha^*, \beta^*)$$



# Multiplicadores de Lagrange: ejemplo



minimizar  $q(\theta) = 1 + (\theta - 2)^2$  con  $\theta \geq 3$

$$\Lambda(\theta, \alpha) = 1 + (\theta - 2)^2 - \alpha (\theta - 3)$$

$$\frac{\partial \Lambda(\theta, \alpha)}{\partial \theta} = 2(\theta - 2) - \alpha = 0 \Rightarrow \theta^*(\alpha) = 2 + \frac{\alpha}{2}$$

$$\Lambda_D(\alpha) = \Lambda(\theta^*(\alpha), \alpha) = 1 + \frac{\alpha^2}{4} - \alpha \left(\frac{\alpha}{2} - 1\right)$$

$$\frac{d \Lambda_D}{d \alpha} = \frac{\alpha}{2} - \frac{\alpha}{2} + 1 - \frac{\alpha}{2} = 1 - \frac{\alpha}{2} = 0 \Rightarrow$$

$$\alpha^* = 2 \geq 0 \rightarrow \theta^* = \theta^*(\alpha^*) = 3$$

## Ejercicios:

- a) minimizar  $q(\theta) = 1 + (\theta - 2)^2$  con la condición de desigualdad  $\theta \leq 3$ ?
- b) minimizar  $q(\theta) = 1 + (\theta - 2)^2$  con la condición de igualdad  $q(\theta) + \theta = 4$

## Multiplicadores de Lagrange: otro ejemplo

En una muestra  $S$  de una tarea de clasificación en *tres* clases se observan 4 datos de la clase  $c = 1$ , 2 datos de  $c = 2$  y 1 dato de  $c = 3$ . Estimar por *máxima verosimilitud* las probabilidades a priori de las clases,  $p_c, 1 \leq c \leq 3$ .

## Multiplicadores de Lagrange: otro ejemplo

En una muestra  $S$  de una tarea de clasificación en *tres* clases se observan 4 datos de la clase  $c = 1$ , 2 datos de  $c = 2$  y 1 dato de  $c = 3$ . Estimar por *máxima verosimilitud* las probabilidades a priori de las clases,  $p_c, 1 \leq c \leq 3$ .

- Modelo:  $P(c = 1) = p_1, P(c = 2) = p_2, P(c = 3) = p_3,$   
 $p_1 + p_2 + p_3 = 1, \quad \Theta \equiv (p_1, p_2, p_3)^t$

## Multiplicadores de Lagrange: otro ejemplo

En una muestra  $S$  de una tarea de clasificación en *tres* clases se observan 4 datos de la clase  $c = 1$ , 2 datos de  $c = 2$  y 1 dato de  $c = 3$ . Estimar por *máxima verosimilitud* las probabilidades a priori de las clases,  $p_c, 1 \leq c \leq 3$ .

- Modelo:  $P(c = 1) = p_1, P(c = 2) = p_2, P(c = 3) = p_3,$   
 $p_1 + p_2 + p_3 = 1, \quad \Theta \equiv (p_1, p_2, p_3)^t$

- Verosimilitud y logaritmo de la verosimilitud:

$$P(S \mid \Theta) = \prod_{i=1}^4 p_1 \prod_{j=1}^2 p_2 \prod_{k=1}^1 p_3 = p_1^4 p_2^2 p_3$$

$$q_S(\Theta) = L_S(\Theta) = \log P(S \mid \Theta) = 4 \log p_1 + 2 \log p_2 + \log p_3$$

## Multiplicadores de Lagrange: otro ejemplo

En una muestra  $S$  de una tarea de clasificación en *tres* clases se observan 4 datos de la clase  $c = 1$ , 2 datos de  $c = 2$  y 1 dato de  $c = 3$ . Estimar por *máxima verosimilitud* las probabilidades a priori de las clases,  $p_c, 1 \leq c \leq 3$ .

- Modelo:  $P(c = 1) = p_1, P(c = 2) = p_2, P(c = 3) = p_3,$   
 $p_1 + p_2 + p_3 = 1, \quad \Theta \equiv (p_1, p_2, p_3)^t$

- Verosimilitud y logaritmo de la verosimilitud:

$$P(S \mid \Theta) = \prod_{i=1}^4 p_1 \prod_{j=1}^2 p_2 \prod_{k=1}^1 p_3 = p_1^4 p_2^2 p_3$$

$$q_S(\Theta) = L_S(\Theta) = \log P(S \mid \Theta) = 4 \log p_1 + 2 \log p_2 + \log p_3$$

- Estimación de máxima verosimilitud:

$$\Theta^* = \arg \max_{\Theta} L_S(\Theta) = \arg \max_{p_1, p_2, p_3} (4 \log p_1 + 2 \log p_2 + \log p_3)$$

## Multiplicadores de Lagrange: otro ejemplo

En una muestra  $S$  de una tarea de clasificación en *tres* clases se observan 4 datos de la clase  $c = 1$ , 2 datos de  $c = 2$  y 1 dato de  $c = 3$ . Estimar por *máxima verosimilitud* las probabilidades a priori de las clases,  $p_c, 1 \leq c \leq 3$ .

- Modelo:  $P(c = 1) = p_1, P(c = 2) = p_2, P(c = 3) = p_3,$   
 $p_1 + p_2 + p_3 = 1, \quad \Theta \equiv (p_1, p_2, p_3)^t$

- Verosimilitud y logaritmo de la verosimilitud:

$$P(S \mid \Theta) = \prod_{i=1}^4 p_1 \prod_{j=1}^2 p_2 \prod_{k=1}^1 p_3 = p_1^4 p_2^2 p_3$$

$$q_S(\Theta) = L_S(\Theta) = \log P(S \mid \Theta) = 4 \log p_1 + 2 \log p_2 + \log p_3$$

- Estimación de máxima verosimilitud:

$$\Theta^* = \arg \max_{\Theta} L_S(\Theta) = \arg \max_{p_1, p_2, p_3} (4 \log p_1 + 2 \log p_2 + \log p_3)$$

Solución: Resolver  $\nabla L_S(\Theta) = 0$

¿Es suficiente?

## Multiplicadores de Lagrange: otro ejemplo

En una muestra  $S$  de una tarea de clasificación en *tres* clases se observan 4 datos de la clase  $c = 1$ , 2 datos de  $c = 2$  y 1 dato de  $c = 3$ . Estimar por *máxima verosimilitud* las probabilidades a priori de las clases,  $p_c, 1 \leq c \leq 3$ .

- Modelo:  $P(c = 1) = p_1, P(c = 2) = p_2, P(c = 3) = p_3,$   
 $p_1 + p_2 + p_3 = 1, \Theta \equiv (p_1, p_2, p_3)^t$

- Verosimilitud y logaritmo de la verosimilitud:

$$P(S | \Theta) = \prod_{i=1}^4 p_1 \prod_{j=1}^2 p_2 \prod_{k=1}^1 p_3 = p_1^4 p_2^2 p_3$$

$$q_S(\Theta) = L_S(\Theta) = \log P(S | \Theta) = 4 \log p_1 + 2 \log p_2 + \log p_3$$

- Estimación de máxima verosimilitud:

$$\Theta^* = \arg \max_{\Theta} L_S(\Theta) = \arg \max_{\substack{p_1, p_2, p_3 \\ p_1 + p_2 + p_3 = 1}} (4 \log p_1 + 2 \log p_2 + \log p_3)$$

- Problema de optimización con restricciones al que aplicaremos la técnica de los multiplicadores de Lagrange

## Ejemplo: aplicación de la técnica de multiplicadores de Lagrange

- Lagrangiana:  $\Lambda(p_1, p_2, p_3, \beta) = 4 \log p_1 + 2 \log p_2 + \log p_3 + \beta (1 - p_1 - p_2 - p_3)$
- Soluciones óptimas en función del multiplicador de Lagrange:

$$\left. \begin{aligned} \frac{\partial \Lambda}{\partial p_1} &= \frac{4}{p_1} - \beta = 0 \\ \frac{\partial \Lambda}{\partial p_2} &= \frac{2}{p_2} - \beta = 0 \\ \frac{\partial \Lambda}{\partial p_3} &= \frac{1}{p_3} - \beta = 0 \end{aligned} \right\} \begin{aligned} p_1^*(\beta) &= \frac{4}{\beta} \\ p_2^*(\beta) &= \frac{2}{\beta} \\ p_3^*(\beta) &= \frac{1}{\beta} \end{aligned}$$

- Función dual de Lagrange:

$$\Lambda_D(\beta) = 4 \log \frac{4}{\beta} + 2 \log \frac{2}{\beta} + \log \frac{1}{\beta} + \beta \left(1 - \frac{4}{\beta} - \frac{2}{\beta} - \frac{1}{\beta}\right) = \beta - 7 \log \beta - 7 + 10 \log 2$$

- Valor óptimo del multiplicador de Lagrange:  $\frac{d\Lambda_D}{d\beta} = 1 - \frac{7}{\beta} = 0 \Rightarrow \beta^* = 7$

- Solución final:  $p_1^* = p_1^*(\beta^*) = \frac{4}{7} \quad p_2^* = p_2^*(\beta^*) = \frac{2}{7} \quad p_3^* = p_3^*(\beta^*) = \frac{1}{7}$

---

**EJERCICIO:** Demostrar que en cualquier problema de clasificación en  $C$  clases, la estimación de máxima verosimilitud de la probabilidad a priori de cada clase  $c$ ,  $1 \leq c \leq C$ , es  $\hat{p}_c = n_c/N$ , donde  $N = \sum_c n_c$  es el número total de datos observados y  $n_c$  es el número de datos de la clase  $c$ .



## Teorema de Kuhn-Tucker

Consideremos un problema de optimización,  $\mathcal{O}$ , definido por:

$$\begin{array}{ll} \text{minimizar} & q(\Theta), \quad \Theta \in \mathbb{R}^D \\ \text{sujeito a} & v_i(\Theta) \geq 0, \quad 1 \leq i \leq k \\ & u_i(\Theta) = 0, \quad 1 \leq i \leq m \end{array}$$

y la correspondiente función Lagrangiana:

$$\Lambda(\Theta, \alpha, \beta) = q(\Theta) - \sum_{i=1}^k \alpha_i v_i(\Theta) + \sum_{i=1}^m \beta_i u_i(\Theta)$$

*Teorema de Kuhn-Tucker:* si  $\exists \Theta^*, \alpha^*, \beta^*$  tales que:

$$\nabla_{\Theta} \Lambda(\Theta, \alpha^*, \beta^*)|_{\Theta^*} = 0;$$

$$\alpha_i^* \geq 0, \quad v_i(\Theta^*) \geq 0, \quad \alpha_i^* v_i(\Theta^*) = 0, \quad 1 \leq i \leq k,$$

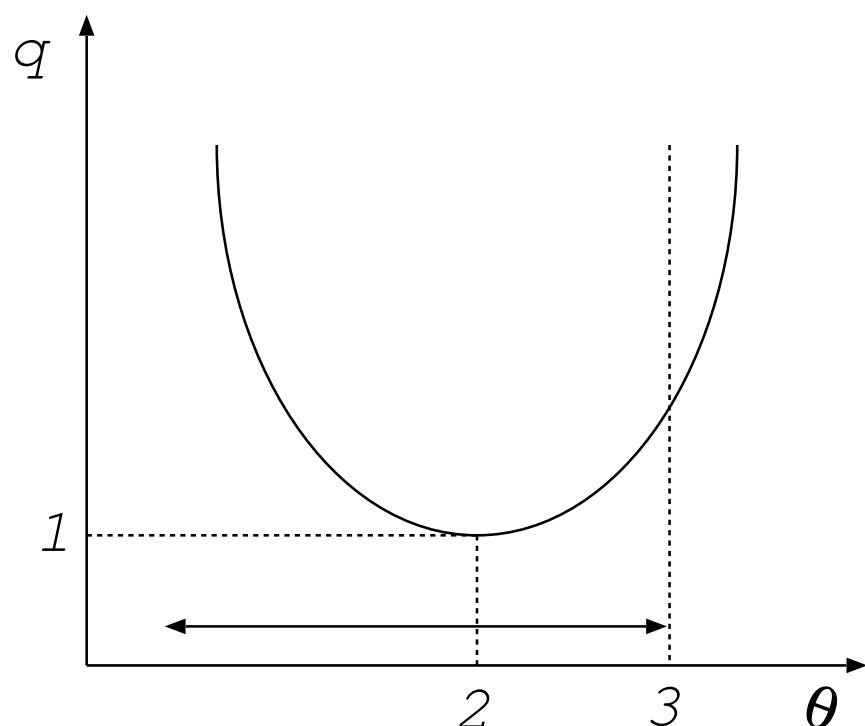
$$u_i(\Theta^*) = 0, \quad 1 \leq i \leq m$$

entonces  $q(\Theta^*)$  es solución al problema  $\mathcal{O}$ .

$\alpha_i^* v_i(\Theta^*) = 0, 1 \leq i \leq k$ : *condiciones complementarias de Karush-Kuhn-Tucker (KKT).*

## Multiplicadores de Lagrange y KKT: ejemplo

En el ejemplo anterior, ¿qué ocurre si la condición de desigualdad es  $\theta \leq 3$ ?



minimizar  $q(\theta) = 1 + (\theta - 2)^2$  con  $3 - \theta \geq 0$

$$\Lambda(\theta, \alpha) = 1 + (\theta - 2)^2 - \alpha (3 - \theta)$$

$$\frac{\partial \Lambda(\theta, \alpha)}{\partial \theta} = 2(\theta - 2) + \alpha = 0 \Rightarrow \theta^*(\alpha) = 2 - \frac{\alpha}{2}$$

$$\text{KKT: } \alpha^* v(\theta^*(\alpha^*)) = 0 \Rightarrow \alpha^* (3 - 2 + \frac{1}{2}\alpha^*) = 0$$

$$\Rightarrow \begin{cases} \alpha^* = 0 \\ \alpha^* = -2 < 0 \rightarrow \text{¡VIOLA } \alpha \geq 0! \end{cases}$$

$$\text{KKT} \Rightarrow \alpha^* = 0 \Rightarrow \theta^* = \theta^*(\alpha^*) = 2$$

*Ejercicio:* mediante el método de KKT, minimizar  $q(\theta) = 1 + (\theta - 2)^2$  con  $3 \leq \theta$ .

# Index

- 1 Introducción ▷ 2
- 2 Optimización analítica: gradiente ▷ 6
- 3 Optimización con restricciones: multiplicadores de Lagrange y teorema Kuhn-Tucker ▷ 11
- 4 *Técnicas de descenso por gradiente* ▷ 22
- 5 Esperanza-Maximización (EM) ▷ 33
- 6 Notación ▷ 53

## Descenso por gradiente

Problema: Minimización sin restricciones de una función objetivo  $q : \mathbb{R}^D \rightarrow \mathbb{R}$ , cuando una solución analítica no es viable:

$$\Theta^* = \arg \min_{\Theta} q(\Theta)$$

- Una solución: construir una secuencia de puntos  $\Theta(1), \dots, \Theta(k), \dots$ , que converja a  $\Theta^*$ .
- Cada valor  $\Theta(k)$  se contruye a partir del anterior  $\Theta(k-1)$  en la secuencia dependiendo de las derivadas de la función en el punto  $\Theta(k)$ .
- Recordatorio: Gradiente de  $q(\Theta)$  en el punto  $\Theta = \Theta(k)$ : vector formado por las derivadas parciales de la función calculadas en  $\Theta(k)$ :

$$\nabla q \big|_{\Theta=\Theta(k)} \equiv \left( \frac{\partial q}{\partial \theta_1} \bigg|_{\Theta=\Theta(k)}, \dots, \frac{\partial q}{\partial \theta_D} \bigg|_{\Theta=\Theta(k)} \right)^t$$

# Descenso por gradiente: algoritmo general

$$\Theta(1) = \text{arbitrario}$$

$$\Theta(k+1) = \Theta(k) - \rho_k \nabla q(\Theta) |_{\Theta=\Theta(k)}$$

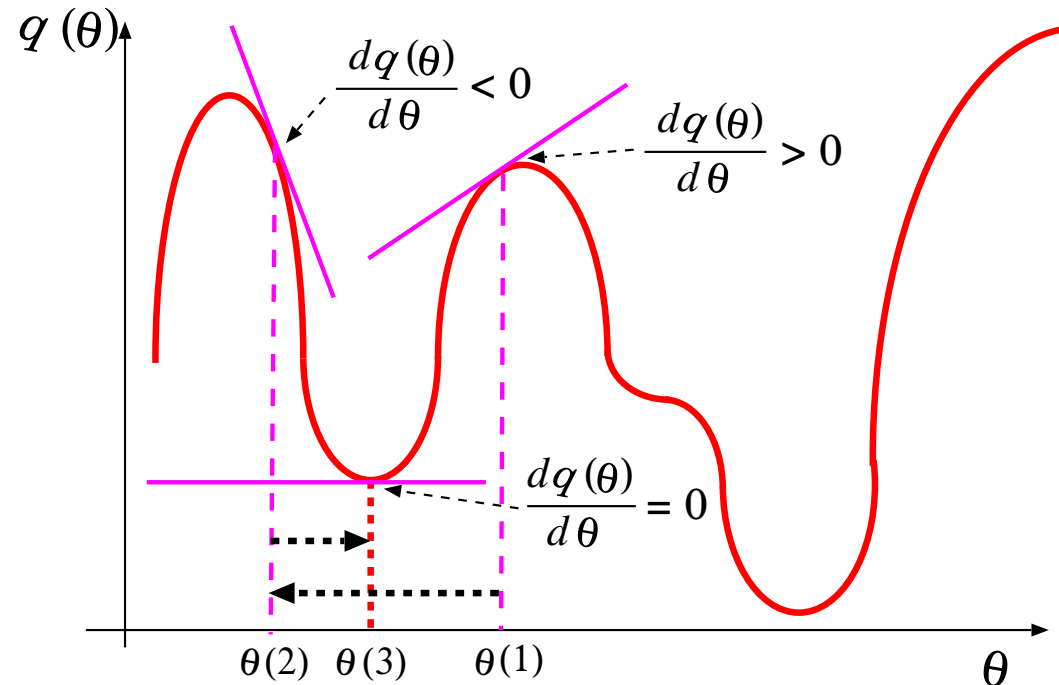
Donde  $\rho_k \in \mathbb{R}^{>0}$  se denomina *tamaño del paso de descenso*

Ejemplo en  $\mathbb{R}^1$  con  $\Theta \stackrel{\text{def}}{=} \theta$ :

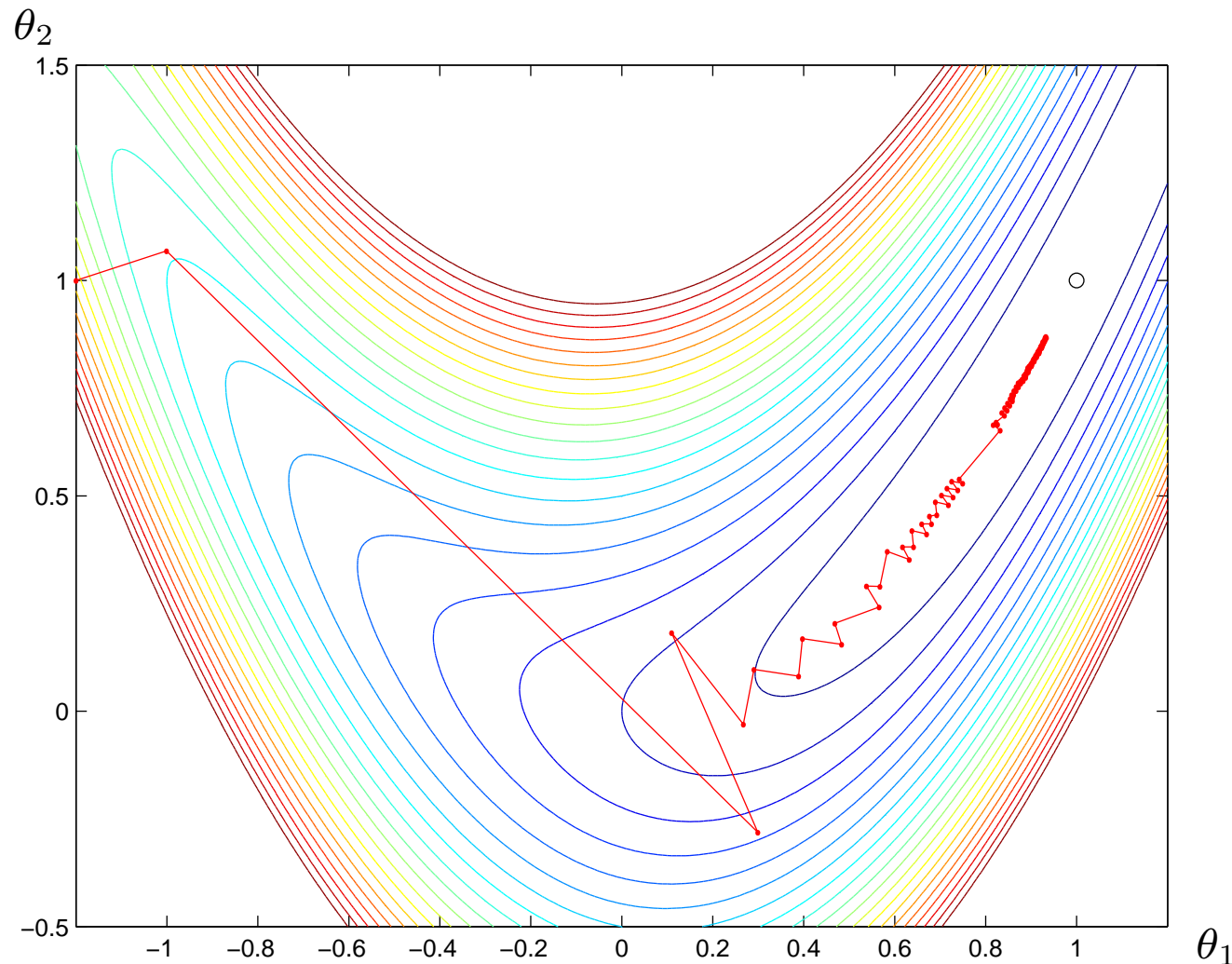
$$\theta(2) = \theta(1) - \rho_1 \left. \frac{dq}{d\theta} \right|_{\theta(1)}$$

$$\theta(3) = \theta(2) - \rho_2 \left. \frac{dq}{d\theta} \right|_{\theta(2)}$$

$$\theta(3) \equiv \theta^*, \quad \left. \frac{dq}{d\theta} \right|_{\theta(3)} = 0$$



# Descenso por gradiente: Ejemplo en $\mathbb{R}^2$



Curvas de nivel de la función de Rosenbrock<sup>1</sup>  $q(\theta_1, \theta_2) = 10(\theta_2 - \theta_1^2)^2 + (1 - \theta_1)^2$  y trayectoria seguida por el vector de parámetros  $\theta = (\theta_1, \theta_2)^t$ .

1. Figuras basadas en la presentación de R. Hauser [http://people.maths.ox.ac.uk/hauser/hauser\\_lecture2.pdf](http://people.maths.ox.ac.uk/hauser/hauser_lecture2.pdf).

## Convergencia y tamaño del paso

- **TEOREMA GENERAL DE CONVERGENCIA:**

Sea  $H(q, \Theta)$  la matriz de segundas derivadas (*Hessiana*) de  $q$  evaluada en  $\Theta$ :

$$H_{ij}(q, \Theta) \stackrel{\text{def}}{=} \frac{\partial^2 q(\Theta)}{\partial \theta_i \partial \theta_j}$$

Sean  $\lambda_l(k)$  los valores propios de  $H(q, \Theta(k))$  en el paso  $k$ -ésimo del algoritmo de *descenso por gradiente*.

*Si  $|1 - \lambda_l(k)\rho_k| < 1 \forall l$ , entonces  $\Theta(k)$  tiende a un mínimo local de  $q(\Theta)$  cuando  $k \rightarrow \infty$*

- **INFLUENCIA DEL TAMAÑO DEL PASO:**

- $\rho < 2/\lambda_{\max}$  garantiza la convergencia
- $\rho \gg \Rightarrow$  convergencia rápida y tendencia a oscilar
- $\rho \ll \Rightarrow$  convergencia lenta

## Ejemplo: clasificador lineal en dos clases

- Clasificador en *dos* clases basado *funciones discriminantes lineales* (FDL):

$$f(\mathbf{x}) = \arg \max_{1 \leq c \leq 2} \phi_c(\mathbf{x}), \quad \mathbf{x} = (x_1, \dots, x_d)^t \in \mathbb{R}^d, \quad \phi_c : \mathbb{R}^d \rightarrow \mathbb{R}, \quad 1 \leq c \leq 2$$

Cada FDL  $\phi_c$  está definida por un vector de pesos  $\theta_c \in \mathbb{R}^D$  donde  $D = d + 1$

En *notación homogénea* se añade una componente,  $x_0 \equiv 1$ , a  $\mathbf{x}$ , con lo que:

$$\phi_c(\mathbf{x}) = \sum_{j=1}^d \theta_{c_j} x_j + \theta_{c_0} = \sum_{j=0}^d \theta_{c_j} x_j = \boldsymbol{\theta}_c^t \mathbf{x}, \quad 1 \leq c \leq 2$$



## Ejemplo: clasificador lineal en dos clases

- Clasificador en *dos* clases basado *funciones discriminantes lineales* (FDL):

$$f(\mathbf{x}) = \arg \max_{1 \leq c \leq 2} \phi_c(\mathbf{x}), \quad \mathbf{x} = (x_1, \dots, x_d)^t \in \mathbb{R}^d, \quad \phi_c : \mathbb{R}^d \rightarrow \mathbb{R}, \quad 1 \leq c \leq 2$$

Cada FDL  $\phi_c$  está definida por un vector de pesos  $\theta_c \in \mathbb{R}^D$  donde  $D = d + 1$

En *notación homogénea* se añade una componente,  $x_0 \equiv 1$ , a  $\mathbf{x}$ , con lo que:

$$\phi_c(\mathbf{x}) = \sum_{j=1}^d \theta_{cj} x_j + \theta_{c0} = \sum_{j=0}^d \theta_{cj} x_j = \boldsymbol{\theta}_c^t \mathbf{x}, \quad 1 \leq c \leq 2$$

- Simplificación (si  $C = 2$ ): etiquetar las clases  $\{1, 2\}$  como  $\{+1, -1\}$  y usar un único vector de pesos  $\boldsymbol{\theta} = \boldsymbol{\theta}_1 - \boldsymbol{\theta}_2$ .

Clasificador,  $f_{\boldsymbol{\theta}} : \mathbb{R}^D \rightarrow \{-1, +1\}$ :

$$f_{\boldsymbol{\theta}}(\mathbf{x}) = \begin{cases} +1 & \text{si } \boldsymbol{\theta}^t \mathbf{x} \geq 0 \\ -1 & \text{si } \boldsymbol{\theta}^t \mathbf{x} < 0 \end{cases} \quad \begin{aligned} &(\phi_1(\mathbf{x}) \geq \phi_2(\mathbf{x}) \Rightarrow \boldsymbol{\theta}_1^t \mathbf{x} \geq \boldsymbol{\theta}_2^t \mathbf{x}) \\ &(\phi_1(\mathbf{x}) < \phi_2(\mathbf{x}) \Rightarrow \boldsymbol{\theta}_1^t \mathbf{x} < \boldsymbol{\theta}_2^t \mathbf{x}) \end{aligned}$$

## Aprendizaje de funciones discriminantes lineales

- Sea  $S = \{(\mathbf{x}_1, c_1), \dots, (\mathbf{x}_N, c_N)\}$ ,  $\mathbf{x}_n \in \mathbb{R}^D$ ,  $c_n \in \{+1, -1\}$  una muestra de entrenamiento.  $S$  es *linealmente separable* (LS) si  $\exists \boldsymbol{\theta} \in \mathbb{R}^D$  ( $D = d + 1$ ) tal que:

$$\forall n, 1 \leq n \leq N, \quad \boldsymbol{\theta}^t \mathbf{x}_n \begin{cases} \geq 0 & \text{if } c_n = +1 \\ < 0 & \text{if } c_n = -1 \end{cases}; \quad \text{es decir, } c_n \boldsymbol{\theta}^t \mathbf{x}_n \geq 0$$

- **Aprendizaje:** Dada  $S$ , encontrar un vector de pesos  $\hat{\boldsymbol{\theta}}$  que la separe; es decir, que satisfaga el sistema de  $N$  inecuaciones:

$$c_n \boldsymbol{\theta}^t \mathbf{x}_n \geq 0, \quad 1 \leq n \leq N$$

# Aprendizaje de funciones discriminantes lineales

- Sea  $S = \{(\mathbf{x}_1, c_1), \dots, (\mathbf{x}_N, c_N)\}$ ,  $\mathbf{x}_n \in \mathbb{R}^D$ ,  $c_n \in \{+1, -1\}$  una muestra de entrenamiento.  $S$  es *linealmente separable* (LS) si  $\exists \boldsymbol{\theta} \in \mathbb{R}^D$  ( $D = d + 1$ ) tal que:

$$\forall n, 1 \leq n \leq N, \quad \boldsymbol{\theta}^t \mathbf{x}_n \begin{cases} \geq 0 & \text{if } c_n = +1 \\ < 0 & \text{if } c_n = -1 \end{cases}; \quad \text{es decir, } c_n \boldsymbol{\theta}^t \mathbf{x}_n \geq 0$$

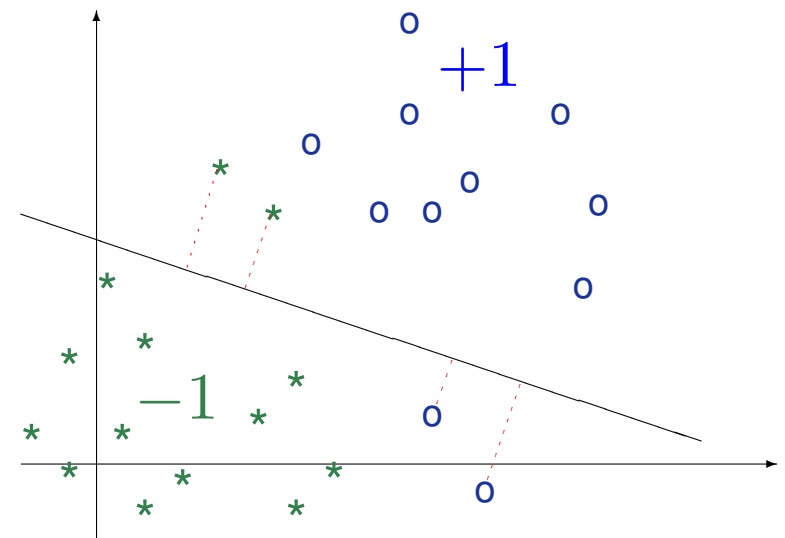
- Aprendizaje:** Dada  $S$ , encontrar un vector de pesos  $\hat{\boldsymbol{\theta}}$  que la separe; es decir, que satisfaga el sistema de  $N$  inecuaciones:

$$c_n \boldsymbol{\theta}^t \mathbf{x}_n \geq 0, \quad 1 \leq n \leq N$$

- Planteamiento equivalente:**  
minimizar la función:  $q_S : \mathbb{R}^D \rightarrow \mathbb{R}^{\geq 0}$ :

$$q_S(\boldsymbol{\theta}) = \sum_{\substack{(\mathbf{x}, c) \in S \\ c \boldsymbol{\theta}^t \mathbf{x} < 0}} -c \boldsymbol{\theta}^t \mathbf{x}$$

proporcional a la suma de *segmentos punteados*  $\longrightarrow$



# Aprendizaje de funciones discriminantes lineales

- Sea  $S = \{(\mathbf{x}_1, c_1), \dots, (\mathbf{x}_N, c_N)\}$ ,  $\mathbf{x}_n \in \mathbb{R}^D$ ,  $c_n \in \{+1, -1\}$  una muestra de entrenamiento.  $S$  es *linealmente separable* (LS) si  $\exists \boldsymbol{\theta} \in \mathbb{R}^D$  ( $D = d + 1$ ) tal que:

$$\forall n, 1 \leq n \leq N, \quad \boldsymbol{\theta}^t \mathbf{x}_n \begin{cases} \geq 0 & \text{if } c_n = +1 \\ < 0 & \text{if } c_n = -1 \end{cases}; \quad \text{es decir, } c_n \boldsymbol{\theta}^t \mathbf{x}_n \geq 0$$

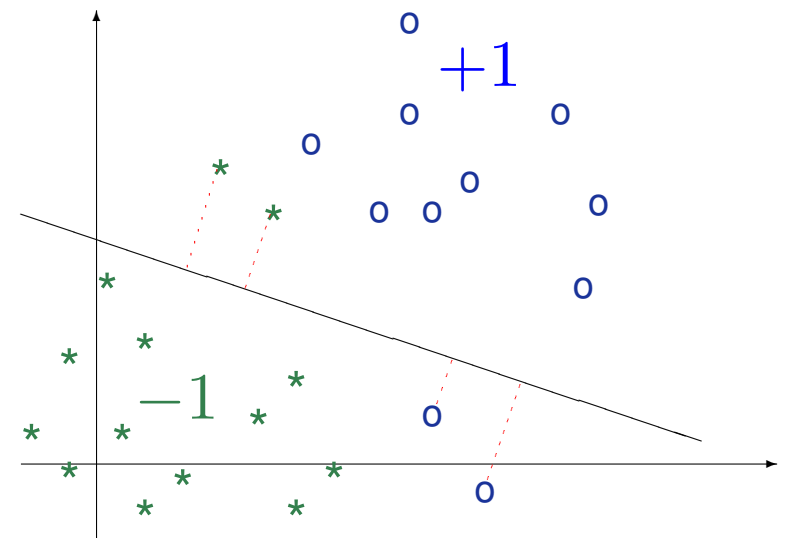
- Aprendizaje:** Dada  $S$ , encontrar un vector de pesos  $\hat{\boldsymbol{\theta}}$  que la separe; es decir, que satisfaga el sistema de  $N$  inecuaciones:

$$c_n \boldsymbol{\theta}^t \mathbf{x}_n \geq 0, \quad 1 \leq n \leq N$$

- Planteamiento equivalente:**  
minimizar la función:  $q_S : \mathbb{R}^D \rightarrow \mathbb{R}^{\geq 0}$ :

$$q_S(\boldsymbol{\theta}) = \sum_{\substack{(\mathbf{x}, c) \in S \\ c \boldsymbol{\theta}^t \mathbf{x} < 0}} -c \boldsymbol{\theta}^t \mathbf{x}$$

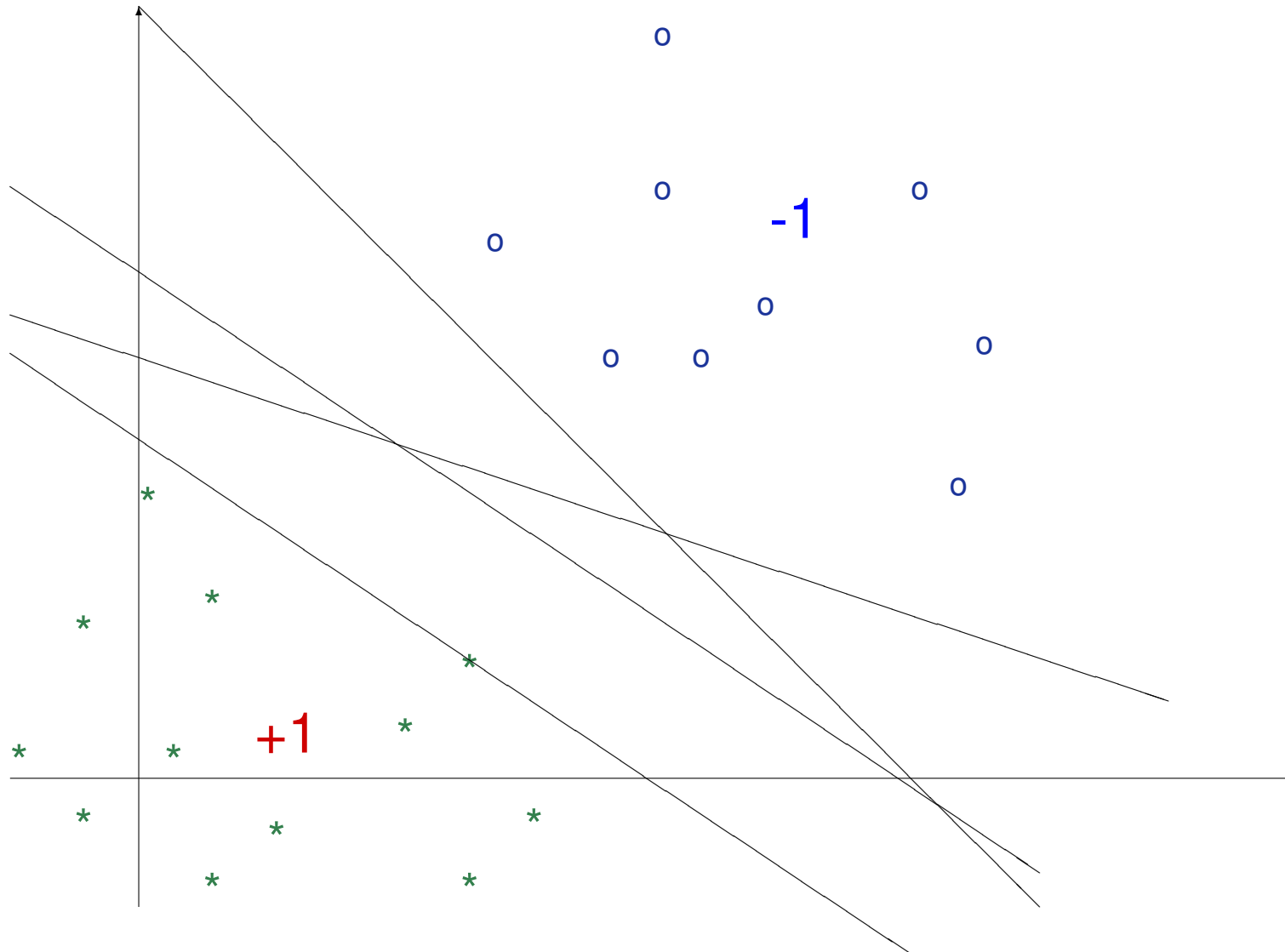
proporcional a la suma de *segmentos punteados*  $\longrightarrow$



- Sea  $\boldsymbol{\theta}^* = \arg \min_{\boldsymbol{\theta}} q_S(\boldsymbol{\theta})$ . Si  $S$  es LS, entonces  $q_S(\boldsymbol{\theta}^*) = 0$ ,  $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}^*$ .

# Aprendizaje de funciones discriminantes lineales

Ejemplo de muestras linealmente separables en  $\mathbb{R}^2$  y posibles soluciones



# Algoritmo perceptrón

$$\nabla q_S(\boldsymbol{\theta}) = \nabla \sum_{\substack{(\mathbf{x},c) \in S \\ c \boldsymbol{\theta}^t \mathbf{x} < 0}} -c \boldsymbol{\theta}^t \mathbf{x} = \sum_{\substack{(\mathbf{x},c) \in S \\ c \boldsymbol{\theta}^t \mathbf{x} < 0}} -c \mathbf{x}$$

$$\boldsymbol{\theta}(1) = \text{arbitrario}$$

$$\boldsymbol{\theta}(k+1) = \boldsymbol{\theta}(k) + \rho_k \sum_{\substack{(\mathbf{x},c) \in S \\ c \boldsymbol{\theta}(k)^t \mathbf{x} < 0}} c \mathbf{x}$$

El *tamaño del paso* aquí se denomina **factor de aprendizaje**

# Algoritmo perceptrón

$$\nabla q_S(\boldsymbol{\theta}) = \nabla \sum_{\substack{(\mathbf{x},c) \in S \\ c \boldsymbol{\theta}^t \mathbf{x} < 0}} -c \boldsymbol{\theta}^t \mathbf{x} = \sum_{\substack{(\mathbf{x},c) \in S \\ c \boldsymbol{\theta}^t \mathbf{x} < 0}} -c \mathbf{x}$$

$$\boldsymbol{\theta}(1) = \text{arbitrario}$$

$$\boldsymbol{\theta}(k+1) = \boldsymbol{\theta}(k) + \rho_k \sum_{\substack{(\mathbf{x},c) \in S \\ c \boldsymbol{\theta}(k)^t \mathbf{x} < 0}} c \mathbf{x}$$

El *tamaño del paso* aquí se denomina **factor de aprendizaje**

*Algoritmo perceptrón muestra a muestra (“online”): (DEMO)*

$$\boldsymbol{\theta}(1) = \text{arbitrario}$$

$$\boldsymbol{\theta}(k+1) = \begin{cases} \boldsymbol{\theta}(k) & c(k) \boldsymbol{\theta}^t \mathbf{x}(k) \geq 0 \\ \boldsymbol{\theta}(k) + \rho_k c(k) \mathbf{x}(k) & c(k) \boldsymbol{\theta}^t \mathbf{x}(k) < 0 \end{cases}$$

## Algoritmo perceptrón

$$\nabla q_S(\boldsymbol{\theta}) = \nabla \sum_{\substack{(\mathbf{x},c) \in S \\ c \boldsymbol{\theta}^t \mathbf{x} < 0}} -c \boldsymbol{\theta}^t \mathbf{x} = \sum_{\substack{(\mathbf{x},c) \in S \\ c \boldsymbol{\theta}^t \mathbf{x} < 0}} -c \mathbf{x}$$

$$\boldsymbol{\theta}(1) = \text{arbitrario}$$

$$\boldsymbol{\theta}(k+1) = \boldsymbol{\theta}(k) + \rho_k \sum_{\substack{(\mathbf{x},c) \in S \\ c \boldsymbol{\theta}(k)^t \mathbf{x} < 0}} c \mathbf{x}$$

El *tamaño del paso* aquí se denomina **factor de aprendizaje**

*Algoritmo perceptrón muestra a muestra (“online”):* (DEMO)

$$\boldsymbol{\theta}(1) = \text{arbitrario}$$

$$\boldsymbol{\theta}(k+1) = \begin{cases} \boldsymbol{\theta}(k) & c(k) \boldsymbol{\theta}^t \mathbf{x}(k) \geq 0 \\ \boldsymbol{\theta}(k) + \rho_k c(k) \mathbf{x}(k) & c(k) \boldsymbol{\theta}^t \mathbf{x}(k) < 0 \end{cases}$$

### TEOREMA DEL PERCEPTRÓN:

*Si  $S$  es LS y  $\rho_k$  es positivo y decreciente o creciente sublinealmente con  $k$ , el algoritmo perceptrón converge a una solución en un número finito de iteraciones*



# Regresión lineal mediante descenso por gradiente

- Sea  $f_{\boldsymbol{\theta}} : \mathbb{R}^D \rightarrow \mathbb{R}$  una función *lineal* ( $D = d + 1$ ):

$$f_{\boldsymbol{\theta}}(\mathbf{x}) \stackrel{\text{def}}{=} \boldsymbol{\theta}^t \mathbf{x}, \quad \mathbf{x}, \boldsymbol{\theta} \in \mathbb{R}^D$$

y  $S$  una muestra de entrenamiento:

$$S = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)\}, \quad \mathbf{x}_n \in \mathbb{R}^D, \quad y_n \in \mathbb{R}$$

- Aprendizaje: Calcular  $\hat{\boldsymbol{\theta}}$  tal que;

$$\hat{\boldsymbol{\theta}}^t \mathbf{x}_n \approx y_n, \quad 1 \leq n \leq N$$

- *Aproximación por mínimos cuadrados*:  
minimizar la **función de Widrow-Hoff**:

$$q_S(\boldsymbol{\theta}) = \frac{1}{2} \sum_{n=1}^N (\boldsymbol{\theta}^t \mathbf{x}_n - y_n)^2$$

- Solución: descenso por gradiente.

## Algoritmo de Widrow-Hoff (Adaline)

$$\nabla q_S(\boldsymbol{\theta}) = \nabla \frac{1}{2} \sum_{n=1}^N (\boldsymbol{\theta}^t \mathbf{x}_n - y_n)^2 = \sum_{n=1}^N (\boldsymbol{\theta}^t \mathbf{x}_n - y_n) \mathbf{x}_n$$

$$\boldsymbol{\theta}(1) = \text{arbitrario}$$

$$\boldsymbol{\theta}(k+1) = \boldsymbol{\theta}(k) + \rho_k \sum_{n=1}^N (y_n - \boldsymbol{\theta}(k)^t \mathbf{x}_n) \mathbf{x}_n$$

## Algoritmo de Widrow-Hoff (Adaline)

$$\nabla q_S(\boldsymbol{\theta}) = \nabla \frac{1}{2} \sum_{n=1}^N (\boldsymbol{\theta}^t \mathbf{x}_n - y_n)^2 = \sum_{n=1}^N (\boldsymbol{\theta}^t \mathbf{x}_n - y_n) \mathbf{x}_n$$

$$\boldsymbol{\theta}(1) = \text{arbitrario}$$

$$\boldsymbol{\theta}(k+1) = \boldsymbol{\theta}(k) + \rho_k \sum_{n=1}^N (y_n - \boldsymbol{\theta}(k)^t \mathbf{x}_n) \mathbf{x}_n$$

*Algoritmo muestra a muestra:* (DEMO)

$$\boldsymbol{\theta}(1) = \text{arbitrario}$$

$$\boldsymbol{\theta}(k+1) = \boldsymbol{\theta}(k) + \rho_k \left( y(k) - \boldsymbol{\theta}(k)^t \mathbf{x}(k) \right) \mathbf{x}(k)$$

# Algoritmo de Widrow-Hoff (Adaline)

$$\nabla q_S(\boldsymbol{\theta}) = \nabla \frac{1}{2} \sum_{n=1}^N (\boldsymbol{\theta}^t \mathbf{x}_n - y_n)^2 = \sum_{n=1}^N (\boldsymbol{\theta}^t \mathbf{x}_n - y_n) \mathbf{x}_n$$

$$\boldsymbol{\theta}(1) = \text{arbitrario}$$

$$\boldsymbol{\theta}(k+1) = \boldsymbol{\theta}(k) + \rho_k \sum_{n=1}^N (y_n - \boldsymbol{\theta}(k)^t \mathbf{x}_n) \mathbf{x}_n$$

*Algoritmo muestra a muestra:* (DEMO)

$$\boldsymbol{\theta}(1) = \text{arbitrario}$$

$$\boldsymbol{\theta}(k+1) = \boldsymbol{\theta}(k) + \rho_k \left( y(k) - \boldsymbol{\theta}(k)^t \mathbf{x}(k) \right) \mathbf{x}(k)$$

TEOREMA:

Si  $\rho_k = \rho_1/k$ ,  $\rho_1 > 0$ ,  $\hat{\boldsymbol{\theta}} = \lim_{k \rightarrow \infty} \boldsymbol{\theta}(k)$  *satisface*  $\nabla q_S(\boldsymbol{\theta})|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} = 0$

# Index

- 1 Introducción ▷ 2
- 2 Optimización analítica: gradiente ▷ 6
- 3 Optimización con restricciones: multiplicadores de Lagrange y teorema Kuhn-Tucker ▷ 11
- 4 Técnicas de descenso por gradiente ▷ 22
- 5 *Esperanza-Maximización (EM)* ▷ 33
- 6 Notación ▷ 53

# Aprendizaje de modelos probabilísticos con variables latentes

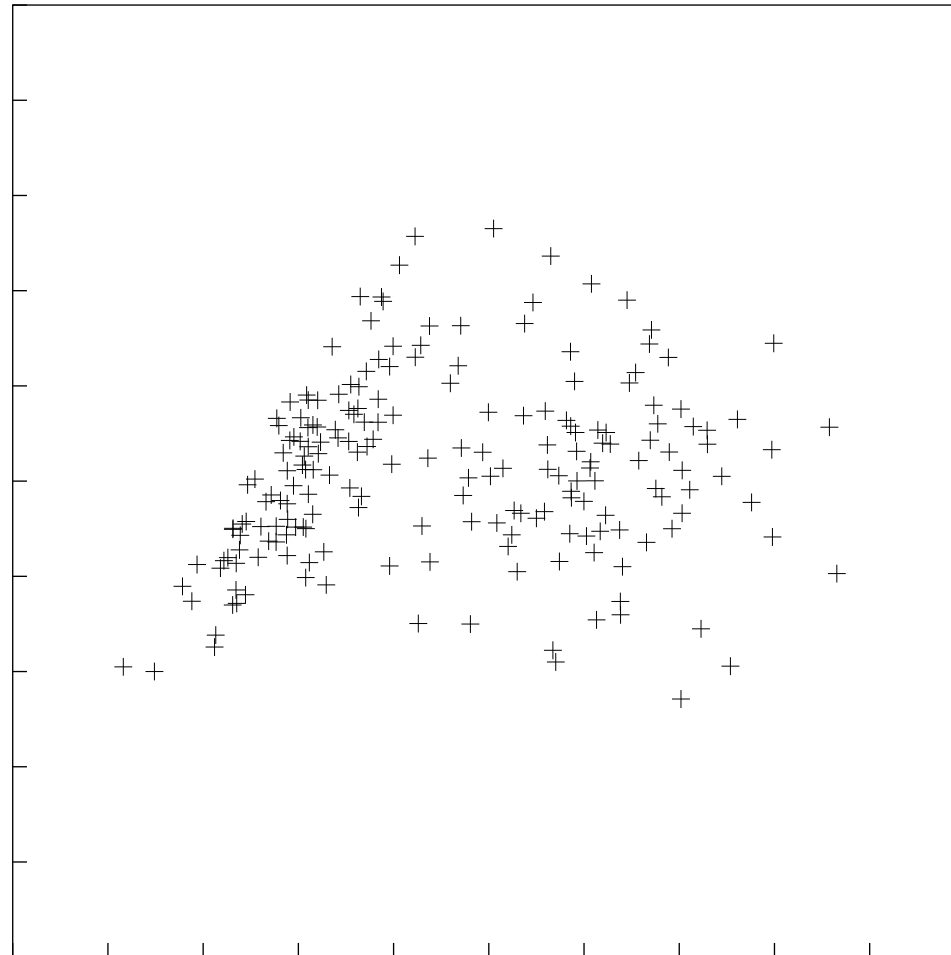
- Se suele usar el criterio de *máxima verosimilitud*; es decir,  $q_S(\Theta) \equiv L_S(\Theta)$ .
- En ocasiones los datos observados *no* contienen suficiente información sobre cómo han sido generados por los modelos probabilísticos asumidos.
- Por ejemplo, en los modelos ocultos de Markov los datos de entrenamiento son cadenas de símbolos, sin información sobre qué *secuencia de estados* ha producido cada cadena.
- Otro ejemplo típico son los modelos definidos como combinación lineal (“mezcla” o “mixtura”) de distribuciones de probabilidad. Los coeficientes de combinación son parámetros a aprender, pero los datos de entrenamiento no contienen información sobre la distribución con que se ha generado cada dato.
- La información ausente en los datos de entrenamiento generalmente se denomina datos *perdidos*, o variables *latentes* u *ocultas*.
- Las técnicas simples de optimización resultan insuficientes para la estimación de los parámetros de estos modelos.

## Ejemplo: mezcla de 2 gaussianas en $\mathbb{R}^2$ y modelo generador

- En el caso de una única gaussiana las muestras se generan en un paso:
  1. Escoger  $\mathbf{x} \in \mathbb{R}^2$ , de acuerdo con la distribución  $p(\mathbf{x}; \boldsymbol{\mu}, \Sigma)$
- Si el modelo es una *mezcla* de 2 gaussianas, el proceso de generación se compone de dos etapas:
  1. Con probabilidad  $P(1) = \alpha_1 \equiv \alpha$  escoger la primera componente de la mezcla o con  $P(2) = \alpha_2 = 1 - \alpha$  escoger la segunda componente con la que se va a generar  $\mathbf{x}$ . Sea  $k \in \{1, 2\}$  la distribución escogida.
  2. Escoger  $\mathbf{x}$ , según la distribución definida por la  $k$ -ésima gaussiana,  $p(\mathbf{x}; \boldsymbol{\mu}_k, \Sigma_k)$ 
    - $\mathbf{x}$  es el dato *observable* y  $k$  es el valor de una variable *oculta*  $z$ . Los datos observables junto con los ocultos se denominan *datos completos*.
    - Probabilidad con la que se genera  $\mathbf{x}$  según este proceso:

$$\begin{aligned}
 p(\mathbf{x}) &= \sum_{k=1}^2 p(z = k, \mathbf{x}) = \sum_{k=1}^2 P(z = k) p(\mathbf{x} | k) \equiv \alpha p(\mathbf{x}; \boldsymbol{\mu}_1, \Sigma_1) + (1 - \alpha) p(\mathbf{x}; \boldsymbol{\mu}_2, \Sigma_2) \\
 &\equiv p(\mathbf{x}; \boldsymbol{\Theta}); \quad \boldsymbol{\Theta} \stackrel{\text{def}}{=} [\alpha, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \Sigma_1, \Sigma_2]
 \end{aligned}$$

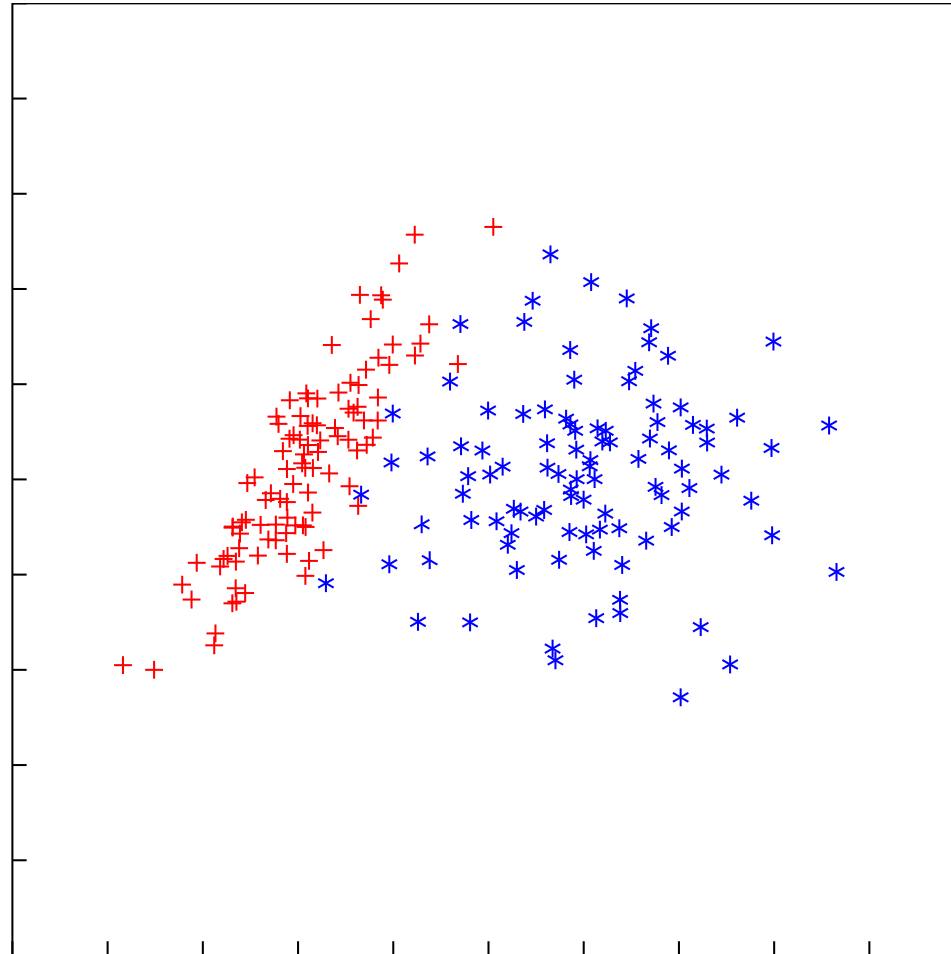
## Mezcla de gaussianas: ilustración en $\mathbb{R}^2$



Datos *observables* generados por una mezcla de dos gaussianas.

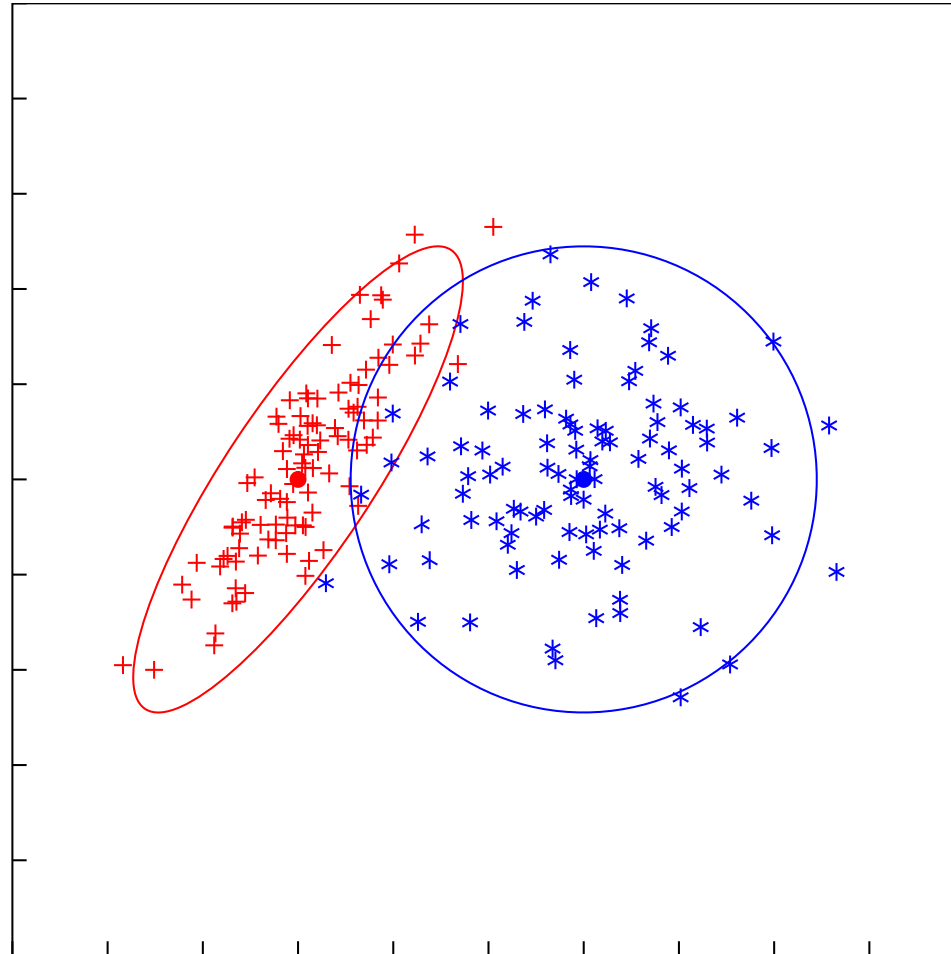


# Mezcla de gaussianas: ilustración en $\mathbb{R}^2$



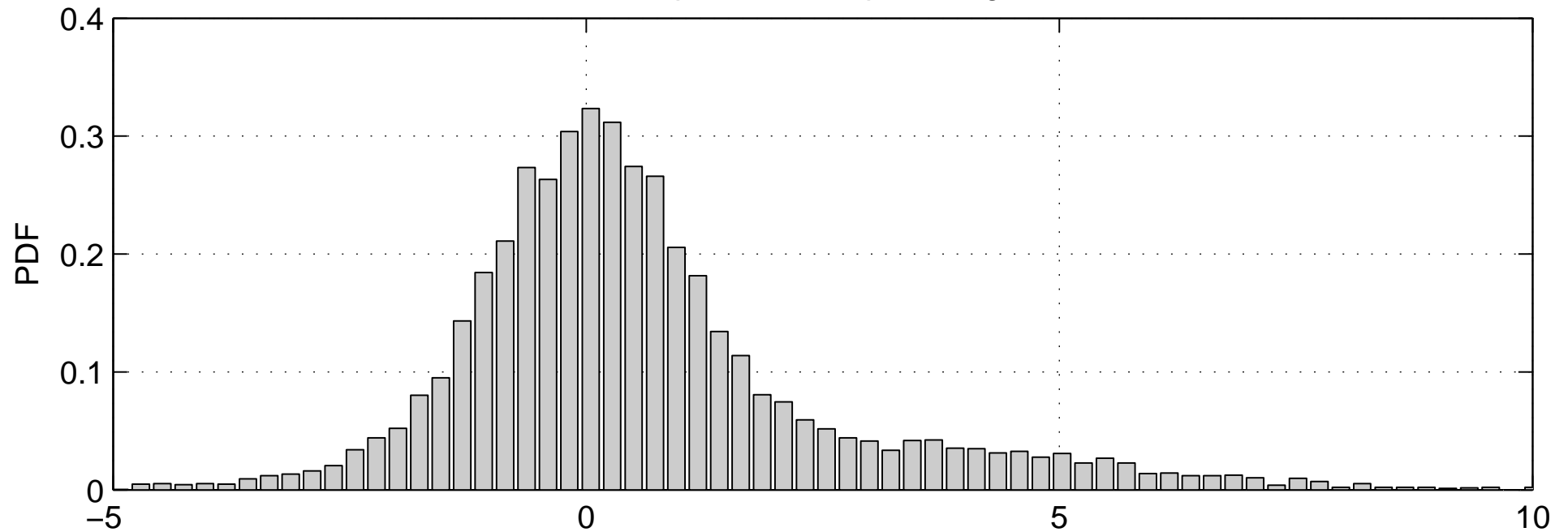
Datos de una mezcla de dos gaussianas con la variable oculta expuesta.

## Mezcla de gaussianas: ilustración en $\mathbb{R}^2$



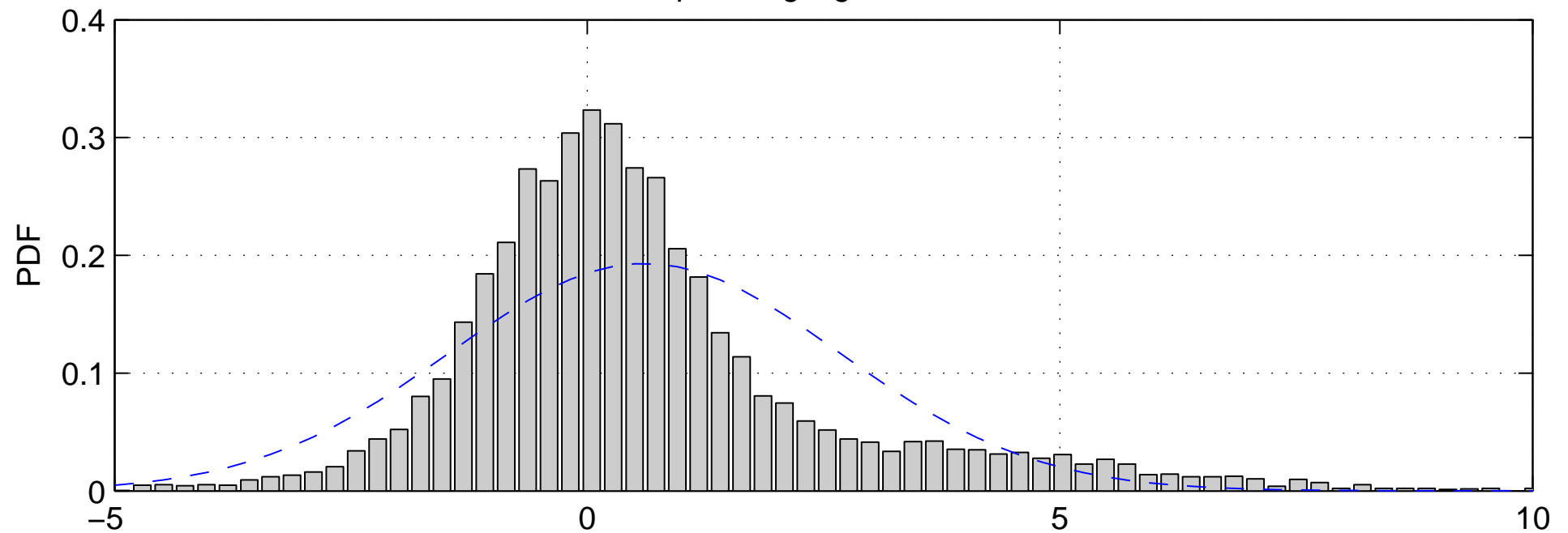
Datos de una mezcla de dos gaussianas con la variable oculta expuesta.  
Las elipses muestran los parámetros de las gaussianas del modelo generador.

# Mezcla de gaussianas: otro ejemplo en $\mathbb{R}$



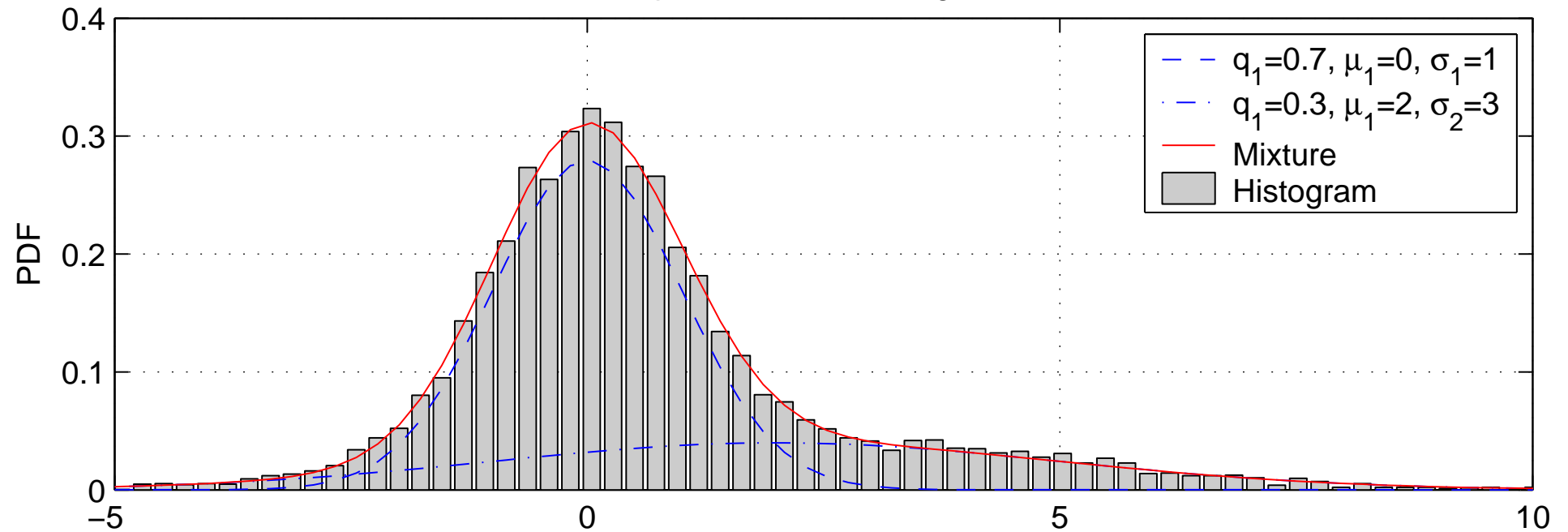
Histograma de una muestra de una variable unidimensional

# Mezcla de gaussianas: otro ejemplo en $\mathbb{R}$



Una única gaussiana estimada a partir de la muestra

# Mezcla de gaussianas: otro ejemplo en $\mathbb{R}$



Mezcla de dos gaussianas con la que se ha generado la muestra

# Aprendizaje de modelos probabilísticos con variables latentes

- El logaritmo de la verosimilitud de la muestra  $S = \{x_1, \dots, x_N\}$  con  $x_i \in \mathbb{R}^d$  para  $1 \leq i \leq N$  y variables latentes  $\{z_1, \dots, z_N\}$ , es:

$$L_S(\Theta) = \log \prod_{n=1}^N P(x_n; \Theta) = \sum_{n=1}^N \log \left( \sum_{z_n} P(x_n, z_n; \Theta) \right)$$

- Problema: Estimar de  $\Theta$  por máxima verosimilitud:  $\Theta^* = \arg \max_{\Theta} L_S(\Theta)$
- Se define una **función auxiliar**  $Q(\Theta, \Theta')$

$$Q(\Theta, \Theta') = \sum_{n=1}^N \sum_{z_n} P(z_n | x_n; \Theta') \log P(x_n, z_n; \Theta)$$

- Teorema:  $Q(\Theta, \Theta') \leq L_S(\Theta)$  para cualquier  $\Theta'$ .
- Estimación de  $\Theta$ : El algoritmo EM utilizando  $Q(\Theta, \Theta')$ .

## Algoritmo esperanza-maximización (EM)

- Inicialización:  $t = 1$ ,  $\Theta(1)$  arbitrario.
- Iteración hasta la convergencia
  - Paso E (esperanza de las variables ocultas). A partir de un conjunto de parámetros dado  $\Theta(t)$  y para todo  $1 \leq n \leq N$ ,

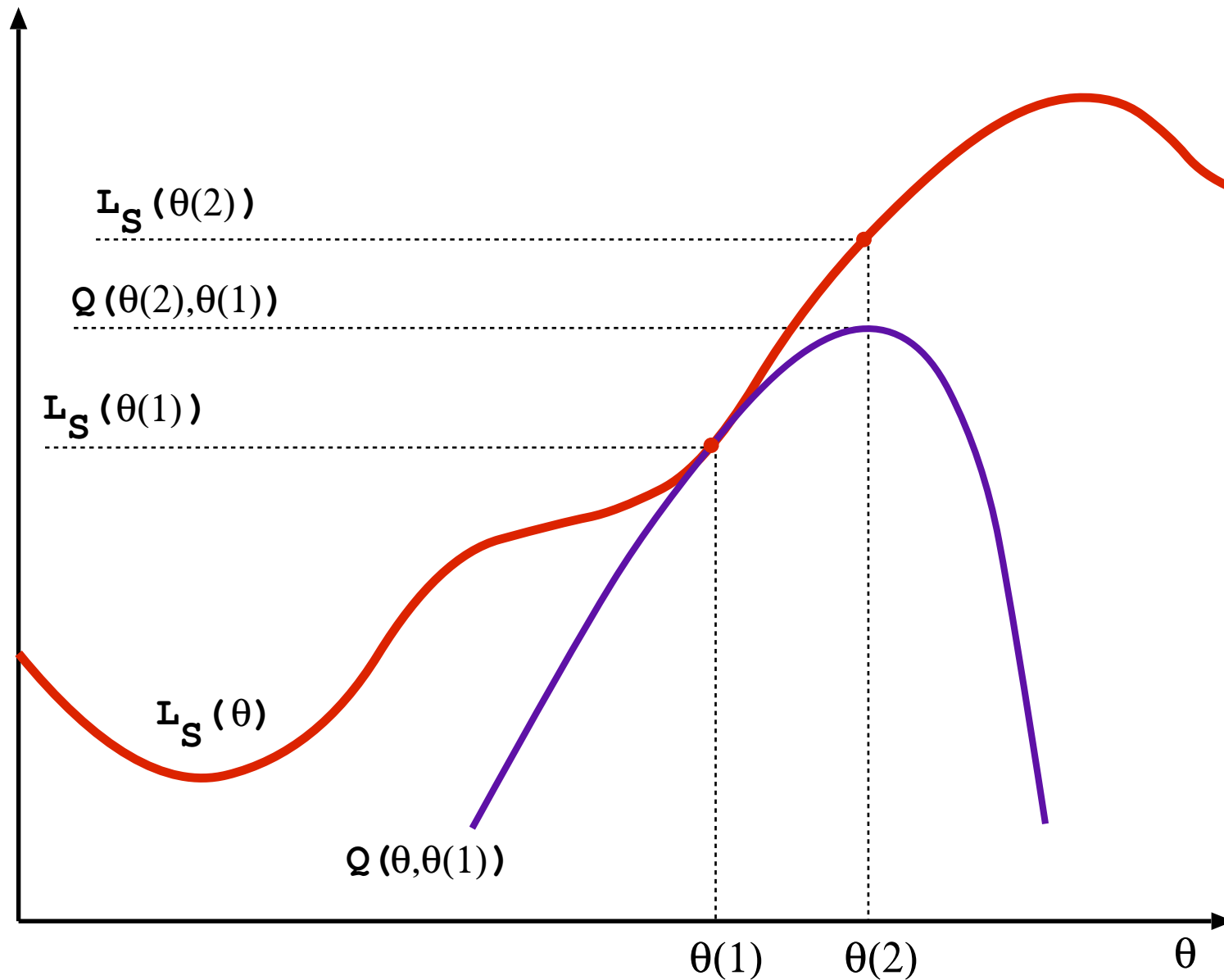
$$P(z_n | x_n; \Theta(t)) = \frac{P(x_n | z_n; \Theta(t))P(z_n; \Theta(t))}{P(x_n)}$$

- Paso M (maximización de la función auxiliar  $Q(\Theta, \Theta(t))$  con respecto a  $\Theta$ )

$$\begin{aligned}\Theta(t+1) &= \arg \max_{\Theta} Q(\Theta, \Theta(t)) \\ &= \arg \max_{\Theta} \sum_{n=1}^N \sum_{z_n} P(z_n | x_n; \Theta(t)) \log P(x_n, z_n; \Theta)\end{aligned}$$

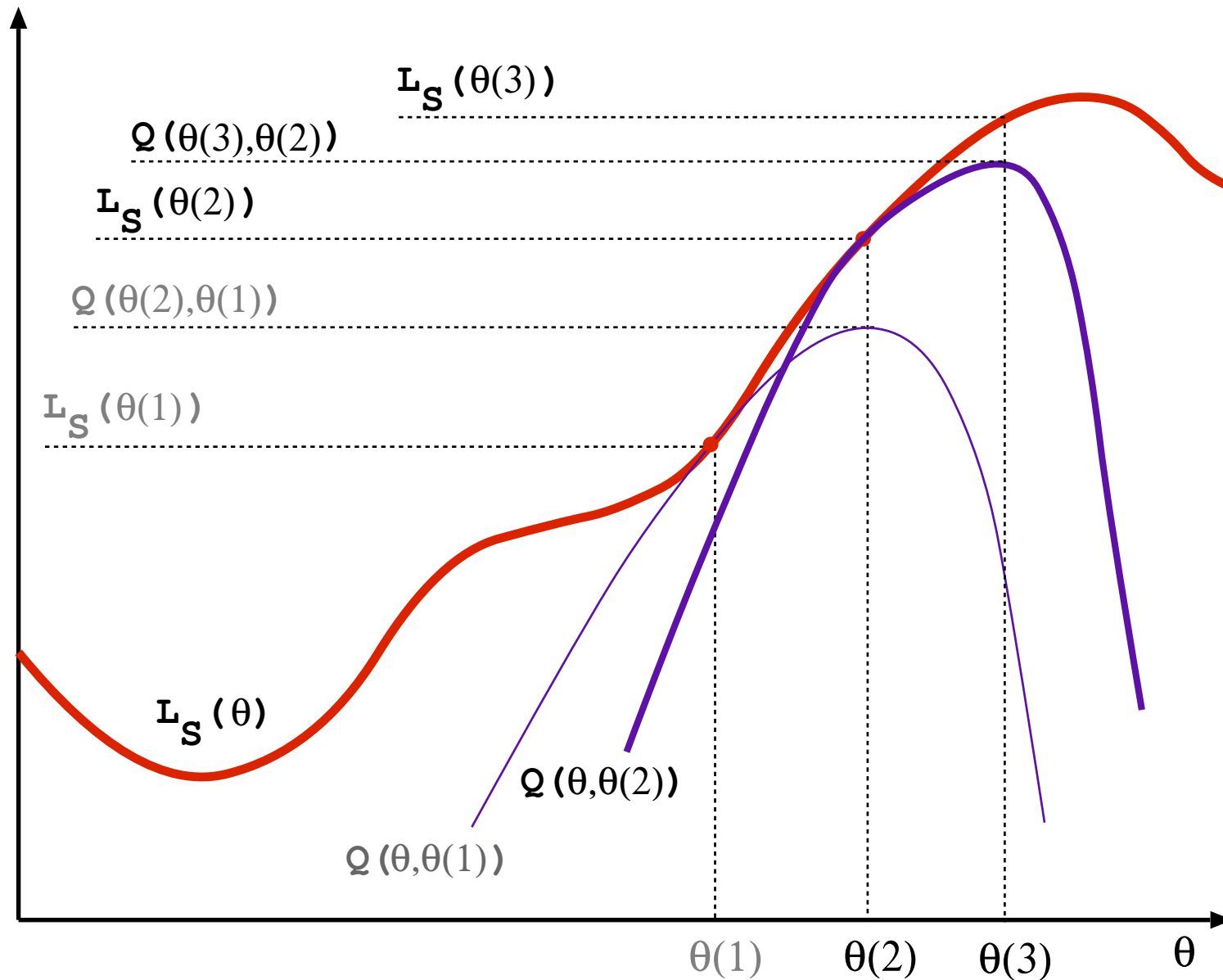
- $t = t + 1$

# Propiedades y convergencia del EM

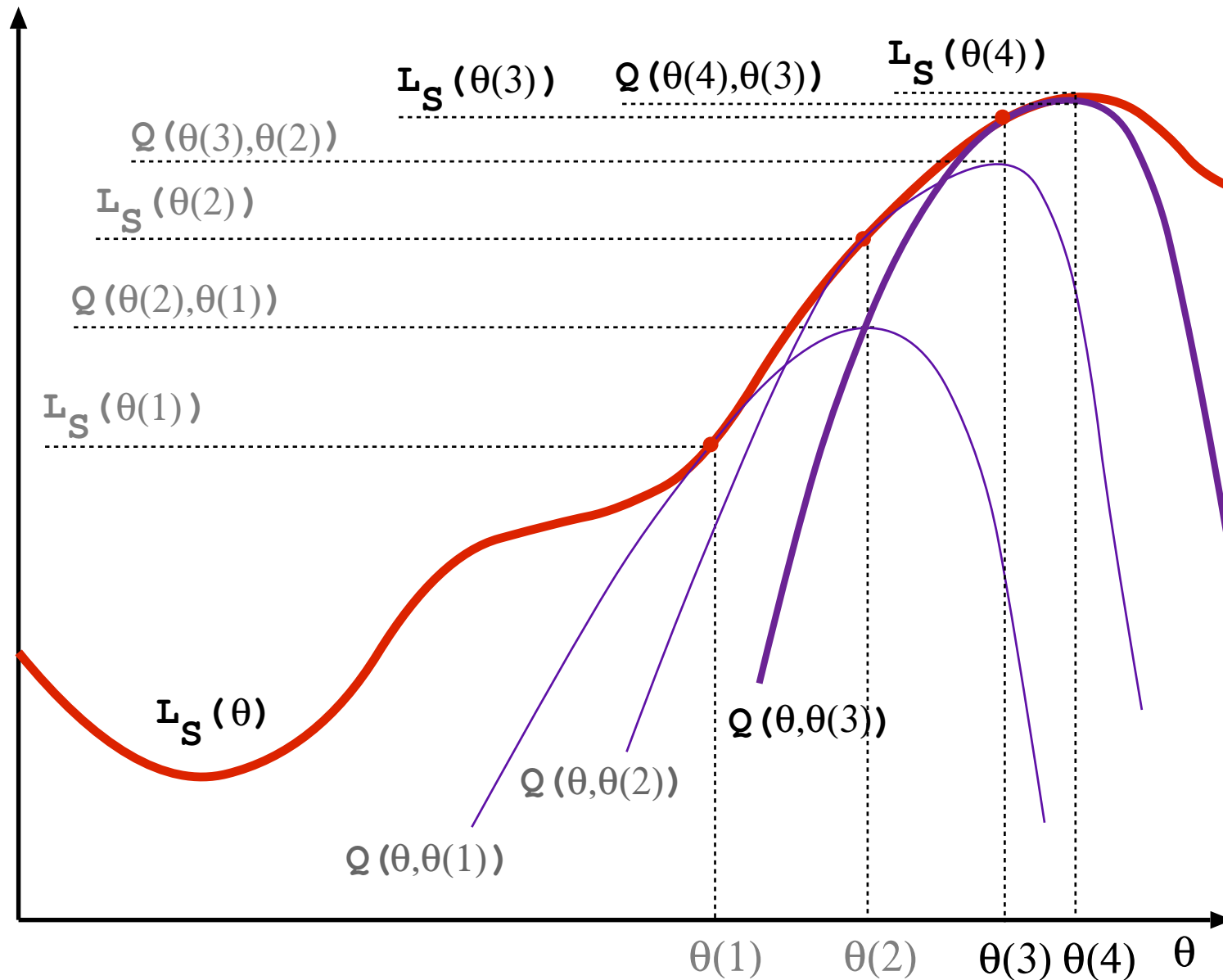




# Propiedades y convergencia del EM



# Propiedades y convergencia del EM



## Ejemplo: mezcla de 2 gaussianas en $\mathbb{R}$

- Dada una muestra  $S = \{x_1, \dots, x_N\}$  con  $x_i \in \mathbb{R}$  para  $1 \leq i \leq N$ , el problema es estimar  $\Theta \stackrel{\text{def}}{=} (\alpha, \mu_1, \mu_2)^t$  (suponemos que  $\sigma_1$  y  $\sigma_2$  están dados)

$$\begin{aligned}
 (\alpha^*, \mu_1^*, \mu_2^*) &= \arg \max_{\mu_1, \mu_2, \alpha} L_S(\alpha, \mu_1, \mu_2) \\
 &= \arg \max_{\mu_1, \mu_2, \alpha} \sum_{n=1}^N \log p(x_n; \mu_1, \mu_2, \alpha) \\
 &= \arg \max_{\mu_1, \mu_2, \alpha} \sum_{n=1}^N \log \sum_{i=1}^2 p(x_n, z_n = i; \mu_1, \mu_2, \alpha) \\
 &= \arg \max_{\mu_1, \mu_2, \alpha} \sum_{n=1}^N \log \sum_{i=1}^2 P(z_n = i; \alpha) p(x_n \mid z_n = i; \mu_i)
 \end{aligned}$$

- Recordemos que,

$$\begin{aligned}
 P(z_n = 1; \alpha) &= \alpha \quad ; \quad P(z_n = 2; \alpha) = (1 - \alpha) \\
 p(x_n \mid z_n = i; \mu_i) &= p(x_n; \mu_i) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{(x - \mu_i)^2}{2\sigma_i^2}\right) \quad \text{para } i = 1, 2 \\
 \log p(x_n \mid z_n = i; \mu_i) &= -\log \sqrt{2\pi}\sigma_i - \frac{(x - \mu_i)^2}{2\sigma_i^2}
 \end{aligned}$$

## Ejemplo: mezcla de 2 gaussianas en $\mathbb{R}$

- Apliquemos el paso E para  $i = 1, 2$  en la iteración  $t$  para  $1 \leq n \leq N$  e  $i = 1, 2$ :

$$P(z_n = i \mid x_n; \Theta(t)) = \frac{P(z_n; \Theta(t)) p(x_n \mid z_n; \Theta(t))}{\sum_{i'=1}^2 P(z_n = i'; \Theta(t)) p(x_n \mid z_n = i'; \Theta(t))}$$

- Sustituyendo:

$$P(z_n = 1 \mid x_n; \Theta(t)) = \frac{\alpha(t) p(x_n; \mu_1(t))}{\alpha(t) p(x_n; \mu_1(t)) + (1 - \alpha(t)) p(x_n; \mu_2(t))}$$

$$P(z_n = 2 \mid x_n; \Theta(t)) = \frac{(1 - \alpha(t)) p(x_n; \mu_2(t))}{\alpha(t) p(x_n; \mu_1(t)) + (1 - \alpha(t)) p(x_n; \mu_2(t))}$$

## Ejemplo: mezcla de 2 gaussianas en $\mathbb{R}$

- Cálculo de  $Q(\Theta, \Theta(t))$  en  $t$  que toma la forma de (ejercicio):

$$\begin{aligned}
 Q(\Theta, \Theta(t)) &= \sum_{n=1}^N \sum_{z_n} P(z_n \mid x_n; \Theta(t)) \log p(x_n, z_n; \Theta) \\
 &= \sum_{n=1}^N \left[ P(z_n = 1 \mid x_n; \Theta(t)) \left( \log \alpha - \log \sqrt{2\pi} \sigma_1 - \frac{(x_n - \mu_1)^2}{2\sigma_1^2} \right) + \right. \\
 &\quad \left. P(z_n = 2 \mid x_n; \Theta(t)) \left( \log(1 - \alpha) - \log \sqrt{2\pi} \sigma_2 - \frac{(x_n - \mu_2)^2}{2\sigma_2^2} \right) \right]
 \end{aligned}$$

- Apliquemos el paso M, maximizando  $Q(\Theta, \Theta(t))$  con respecto a  $\Theta = (\mu_1, \mu_2, \alpha)^t$  (ejercicio)

$$\text{Resolver } \frac{\partial Q(\Theta, \Theta(t))}{\partial \alpha} = 0, \quad \frac{\partial Q(\Theta, \Theta(t))}{\partial \mu_1} = 0, \quad \frac{\partial Q(\Theta, \Theta(t))}{\partial \mu_2} = 0$$

## Algoritmo EM para la mezcla de 2 gaussianas en $\mathbb{R}$

- Inicialización:  $t = 1$ ,  $\Theta(1)$  arbitrarios ( $\Theta(1) = (\mu_1(1), \mu_2(1), \alpha(1))^t$ ).
- Iteración hasta la convergencia
  - Paso E (esperanza). A partir de  $\Theta(t) = (\mu_1(t), \mu_2(t), \alpha(t))^t$  y para  $1 \leq n \leq N$ :

$$P(z_n = 1 \mid x_n; \Theta(t)) = \frac{\alpha(t) p(x_n; \mu_1(t))}{\alpha(t) p(x_n; \mu_1(t)) + (1 - \alpha(t)) p(x_n; \mu_2(t))}$$

$$P(z_n = 2 \mid x_n; \Theta(t)) = \frac{(1 - \alpha(t)) p(x_n; \mu_2(t))}{\alpha(t) p(x_n; \mu_1(t)) + (1 - \alpha(t)) p(x_n; \mu_2(t))}$$

- Paso M (maximización)

$$\alpha(t+1) = \frac{1}{N} \sum_{n=1}^N P(z_n = 1 \mid x_n; \Theta(t))$$

$$\mu_1(t+1) = \frac{1}{N \alpha(t+1)} \sum_{n=1}^N P(z_n = 1 \mid x_n; \Theta(t)) x_n$$

$$\mu_2(t+1) = \frac{1}{N (1 - \alpha(t+1))} \sum_{n=1}^N P(z_n = 2 \mid x_n; \Theta(t)) x_n$$

- $t = t + 1$

## Ejemplo: mezcla de gaussianas y modelo generador en $\mathbb{R}^D$

- En el caso de una única gaussiana las muestras se generan en un paso:
  1. Escoger  $x$ , de acuerdo con la distribución  $p(x \mid \mu, \Sigma)$
- Si el modelo es una *mezcla* de  $K$  gaussianas, el proceso de generación se compone de dos etapas:
  1. De acuerdo con la distribución  $P(k) = \alpha_k$ , escoger la componente  $k$ -ésima de la mezcla con la que se va a generar  $x$
  2. Escoger  $x$ , según la distribución definida por la  $k$ -ésima gaussiana,  $p(x \mid \mu_k, \Sigma_k)$ 
    - $x$  es el dato *observable* y  $k$  es el valor de una variable *oculta*  $z$ . Los datos observables junto con los ocultos se denominan *datos completos*
    - Probabilidad con la que se genera  $x$  según este proceso:

$$\begin{aligned}
 p(x) &= \sum_{k=1}^K p(z = k, x) = \sum_{k=1}^K P(z = k) p(x \mid k) \equiv \sum_{k=1}^K \alpha_k p(x; \mu_k, \Sigma_k) \\
 &\equiv p(x; \Theta); \quad \Theta \stackrel{\text{def}}{=} (\mu_1, \dots, \mu_K, \Sigma_1, \dots, \Sigma_K, \alpha_1, \dots, \alpha_K)^t
 \end{aligned}$$

# Algoritmo EM para la mezcla de $K$ gaussianas

- $\Theta \stackrel{\text{def}}{=} (\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K, \alpha_1, \dots, \alpha_K)^t$ , supondremos que  $\Sigma_1 \dots, \Sigma_K$  están dados.
- Inicialización:  $t = 1$ ,  $\Theta(1) = (\boldsymbol{\mu}_1(1), \dots, \boldsymbol{\mu}_K(1), \alpha_1(1), \dots, \alpha_K(1))^t$  arbitrarios.
- Iteración hasta la convergencia
  - Paso E (esperanza). A partir de  $\Theta(t)$  y para  $1 \leq n \leq N$  y  $1 \leq k \leq K$ :

$$P(z_n = k \mid \mathbf{x}_n; \Theta(t)) = \frac{\alpha_k(t) p(\mathbf{x}_n; \boldsymbol{\mu}_k(t))}{\sum_{k'} \alpha_{k'}(t) p(\mathbf{x}_n; \boldsymbol{\mu}_{k'}(t))}$$

- Paso M (maximización). Para todo  $1 \leq k \leq K$ :

$$\alpha_k(t+1) = \frac{1}{N} \sum_{n=1}^N P(z_n = k \mid \mathbf{x}_n; \Theta(t))$$

$$\boldsymbol{\mu}_k(t+1) = \frac{1}{N \alpha_k(t+1)} \sum_{n=1}^N P(z_n = k \mid \mathbf{x}_n; \Theta(t)) \mathbf{x}_n$$

- $t = t + 1$



# Index

- 1 Introducción ▷ 2
- 2 Optimización analítica: gradiente ▷ 6
- 3 Optimización con restricciones: multiplicadores de Lagrange y teorema Kuhn-Tucker ▷ 11
- 4 Técnicas de descenso por gradiente ▷ 22
- 5 Esperanza-Maximización (EM) ▷ 33
- 6 *Notación* ▷ 53

## Notación

- $\Theta = (\Theta_1, \dots, \Theta_D)^t$ : vector de parámetros. Como los vectores son matrices columna, para representar las componentes en fila se usa la  $t$  (“*transpuesta*”).
- $q_S(\Theta)$ : función objetivo a optimizar definida sobre un conjunto de entrenamiento  $S$ , cuyos de parámetros son  $\Theta$
- $\nabla q(\Theta) \stackrel{\text{def}}{=} \left( \frac{\partial q(\Theta)}{\partial \Theta_1}, \dots, \frac{\partial q(\Theta)}{\partial \Theta_D} \right)^t$ : gradiente de la función  $q_s$   
(vector de derivadas parciales con respecto a cada componente de  $\Theta$ )
- $\nabla q(\Theta) \big|_{\Theta=\Theta(k)} \equiv \left( \frac{\partial q}{\partial \Theta_1} \bigg|_{\Theta=\Theta(k)}, \dots, \frac{\partial q}{\partial \Theta_D} \bigg|_{\Theta=\Theta(k)} \right)^t$ : gradiente de la función  $q$   
calculado en  $\Theta(k)$
- $\Sigma$ : matriz de covarianzas de una distribución gaussiana