2016-2017

Aprendizaje Automático

3. Técnicas de optimización

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

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

Departament de Sistemas Informàtics i Computació (DSIC)

Universitat Politècnica de València (UPV)

Septiembre, 2016

Index

Clasificación, regresión y optimización

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

Optimización y aprendizaje automático

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

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

o bien:

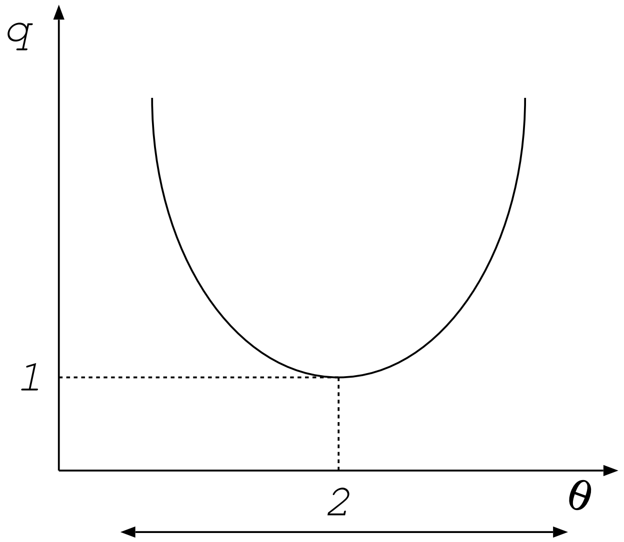
$$\hat{\mathbf{\Theta}} \equiv \mathbf{\Theta}^\star = \arg\max_{\mathbf{\Theta}} q_S(\mathbf{\Theta})$$

Técnicas generales de AA basadas en optimización

- f_Θ es una función cualquiera que depende de un vector de parámetros $\mathbf{\Theta}$: $q_S(\mathbf{\Theta})$ se basa en *funciones de error* y típicamente su optimización utiliza técnicas de *descenso/ascenso por gradiente* (caso particular de "hill-climbing").
- f_Θ es una función cualquiera dependiente de $\mathbf{\Theta}$, pero hay ciertas restricciones en los valores posibles de los parámetros $\mathbf{\Theta}$: *optimización con restricciones* de $q_S(\mathbf{\Theta})$ mediante la técnica de los *multiplicadores de Lagrange*.
- f_Θ 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_Θ 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

Optimización analítica: ejemplo



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

Optimización analítica: gradiente

- Dada una función *convexa* $q : \mathbb{R}^D \to \mathbb{R}$, calcular $\arg\min_{\boldsymbol{\Theta} \in \mathbb{R}^D} q(\boldsymbol{\Theta})$
- Procedimiento:
  1. Calcular el gradiente de q: $\boldsymbol{\nabla} q(\boldsymbol{\Theta}) \stackrel{\text{def}}{=} \left(\dfrac{\partial q(\boldsymbol{\Theta})}{\partial \Theta_1}, \ldots, \dfrac{\partial q(\boldsymbol{\Theta})}{\partial \Theta_D}\right)^t$
  2. Resolver $\boldsymbol{\nabla} q(\boldsymbol{\Theta}) = \mathbf{0}$;

     es decir, resolver el sistema de ecuaciones $\dfrac{\partial q(\boldsymbol{\Theta})}{\partial \Theta_i} = 0,\ 1 \le i \le D$.

     Sean $\Theta_1^\star, \ldots, \Theta_D^\star$ las soluciones obtenidas.
- Solución: $\boldsymbol{\Theta}^\star = (\Theta_1^\star, \ldots, \Theta_D^\star)^t$
- Si q es convexa, $\boldsymbol{\nabla} q(\boldsymbol{\Theta}^\star) = \mathbf{0}$ es una condición *necesaria y suficiente* para que $\boldsymbol{\Theta}^\star$ sea (la única) solución.

Ejercicios:

a) ¿Qué ocurre si q *no* es convexa?

b) Encontrar el vector $\boldsymbol{\theta}^\star \in \mathbb{R}^2$ que minimiza la función $q(\boldsymbol{\theta}) = (\theta_1 - 1)^2 + (\theta_2 - 2)^2$

Optimización analítica: otro ejemplo simple

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

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

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

$$q_S(\boldsymbol{\Theta}) \;\equiv\; L_S(\mu, \sigma) \;=\; \log \prod_{n=1}^{N} p(x_n \mid \mu, \sigma) \;=\; \sum_{n=1}^{N} \log p(x_n \mid \mu, \sigma)$$

$$=\; N\ \log \frac{1}{\sigma\sqrt{2\pi}} - \frac{1}{2\sigma^2} \sum_{n=1}^{N} (x_n - \mu)^2$$

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

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

[1] media y desviación típica

Optimización analítica: otro ejemplo algo menos simple

Ejercicio propuesto:

Estimar los parámetros $\boldsymbol{\Theta} \equiv (\mu_1, \mu_2)$ de una gaussiana bivariada (en $\mathbb{R}^2$), en la que la matriz de covarianza

$$\Sigma = \begin{bmatrix} \sigma_1 & \sigma_{12} \\ \sigma_{12} & \sigma_2 \end{bmatrix}$$

es conocida:

$$p(x_1, x_2 \mid \mu_1, \mu_2) = A \cdot \exp\left(-B \left[\frac{(x_1-\mu_1)^2}{\sigma_1^2} + \frac{(x_2-\mu_2)^2}{\sigma_2^2} - \frac{2\sigma_{12}(x_1-\mu_1)(x_2-\mu_2)}{(\sigma_1\sigma_2)^2}\right]\right)$$

donde

$$A = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-(\sigma_{12}/\sigma_1\sigma_2)^2}}, \qquad B = \frac{1}{2(1-(\sigma_{12}/\sigma_1\sigma_2)^2)}$$

Otro ejercicio algo más complejo propuesto: Asumir que Σ tampoco es conocida; es decir estimar: $\boldsymbol{\Theta} \equiv (\mu_1, \mu_2, \sigma_1, \sigma_2, \sigma_{12})$

Optimización analítica: otro ejemplo

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

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

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

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

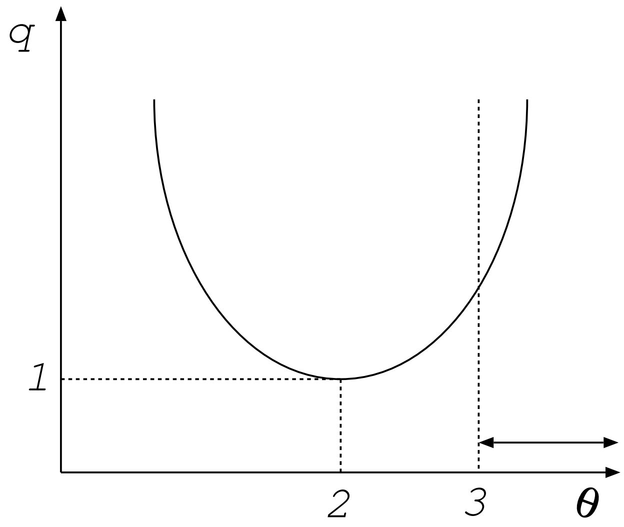
$$\begin{aligned} q_S(\boldsymbol{\Theta}) \equiv L_S(\boldsymbol{\Theta}) &= \log \prod_{n=1}^{N} p(\boldsymbol{x}_n \mid \boldsymbol{\Theta}) = \sum_{n=1}^{N} \log p(\boldsymbol{x}_n \mid \boldsymbol{\Theta}) \\ &= N \log\left((2\pi)^{-D/2}|\Sigma|^{-1/2}\right) - \frac{1}{2}\sum_{n=1}^{N}(\boldsymbol{x}_n-\boldsymbol{\mu})^t\Sigma^{-1}(\boldsymbol{x}_n-\bolds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

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

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

Index

Optimización con restricciones: ejemplo



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

Optimización con restricciones: multiplicadores de Lagrange

Consideremos un problema de optimización definido por:

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

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

$$\boldsymbol{\Theta}^\star = \mathop{\arg\min}_{\substack{\boldsymbol{\Theta} \in \mathbb{R}^D \\ v_i(\boldsymbol{\Theta}) \geq 0,\ 1 \leq i \leq k \\ u_i(\boldsymbol{\Theta}) = 0,\ 1 \leq i \leq m}} q(\boldsymbol{\Theta})$$

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

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

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:*

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

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

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

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

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

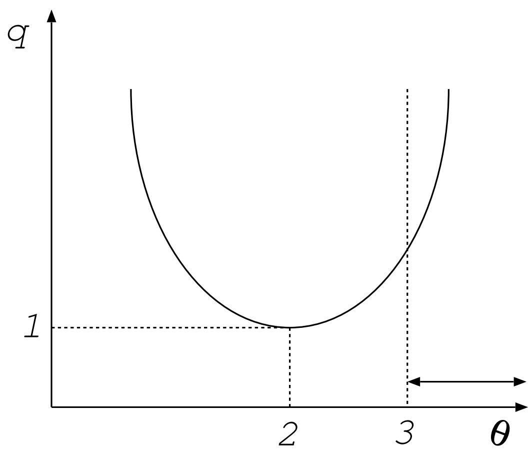
4. *Optimizar la función dual de Lagrange* (usualmente resolviendo $\boldsymbol{\nabla} \Lambda_D(\boldsymbol{\alpha}, \boldsymbol{\beta}) = 0$):

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

5. *Solución final*:

$$\boldsymbol{\Theta}^\star = \boldsymbol{\Theta}^\star(\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star)$$

Multiplicadores de Lagrange: ejemplo



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

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

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

$$\Lambda_D(\alpha) = \Lambda(\theta^\star(\alpha), \alpha) = 1 + \frac{\alpha^2}{4} - \alpha\,(2 + \frac{\alpha}{2} - 3)$$

$$\frac{d\,\Lambda_D}{d\,\alpha} = \frac{\alpha}{2} - 2 - \frac{\alpha}{2} + 3 - \frac{\alpha}{2} = 1 - \frac{\alpha}{2} = 0 \Rightarrow$$

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

Ejercicio:

a) ¿Y qué ocurre si la condición de desigualdad es $\theta \leq 3$?

b) minimizar $q(\theta) = 1 + (\theta - 2)^2$ con $q(\theta) + \theta = 4$

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, $p_c, 1 \leq c \leq 3$.

- Modelo: $P(c=1) = p_1, \;\; P(c=2) = p_2, \;\; P(c=3) = p_3,$
  $p_1 + p_2 + p_3 = 1, \quad \mathbf{\Theta} \equiv (p_1, p_2, p_3)^t$
- Verosimilitud y logaritmo de la verosimilitud:

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

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

- Estimación de máxima verosimilitud:

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

- 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:

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

- Función dual de Lagrange:

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

- Valor óptimo del multiplicador de Lagrange: $\dfrac{d\Lambda_D}{d\beta} \;=\; 1 - \dfrac{7}{\beta} \;=\; 0 \;\Rightarrow\; \beta^\star \;=\; 7$
- Solución final: $p_1^\star = p_1^\star(\beta^\star) = \dfrac{4}{7} \qquad p_2^\star = p_2^\star(\beta^\star) = \dfrac{2}{7} \qquad p_3^\star = p_3^\star(\beta^\star) = \dfrac{1}{7}$

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:

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

y la correspondiente función Lagrangiana:

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

Teorema de Kuhn-Tucker: si $\exists\ \boldsymbol{\Theta}^\star, \boldsymbol{\alpha}^\star$, $\boldsymbol{\beta}^\star$ tales que:

$$\boldsymbol{\nabla}_{\boldsymbol{\Theta}}\, \Lambda(\boldsymbol{\Theta}, \boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star)|_{\boldsymbol{\Theta}^\star} \;=\; \mathbf{0};$$

$$\alpha_i^\star \geq 0, \quad v_i(\boldsymbol{\Theta}^\star) \geq 0, \quad \alpha_i^\star\, v_i(\boldsymbol{\Theta}^\star) = 0, \quad 1 \leq i \leq k,$$

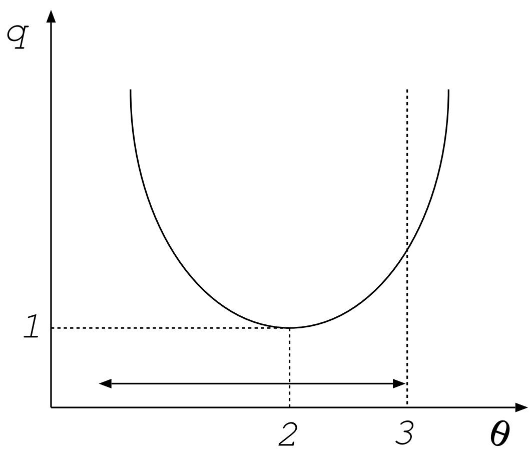
$$u_i(\boldsymbol{\Theta}^\star) = 0, \quad 1 \leq i \leq m$$

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

$\alpha_i^\star v_i(\boldsymbol{\Theta}^\star) = 0,\ 1 \leq i \leq k$: *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 $3 - \theta \geq 0$

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

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

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

$$\Rightarrow \begin{cases} \alpha^\star = 0 \\ \alpha^\star = 2(2-3) \;<\; 0 \;\rightarrow\; \text{¡VIOLA } \alpha \geq 0\,! \end{cases}$$

KKT $\Rightarrow\ \alpha^\star = 0 \ \Rightarrow\ \theta^\star = \theta^\star(\alpha^\star) = 2$

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

Index

Descenso por gradiente

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

$$\mathbf{\Theta}^\star = \arg\min_{\mathbf{\Theta}} \; q(\mathbf{\Theta})$$

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

$$\boldsymbol{\nabla} q \,\Big|_{\mathbf{\Theta}=\mathbf{\Theta}(k)} \equiv \left(\frac{\partial q}{\partial \Theta_1}\Big|_{\mathbf{\Theta}=\mathbf{\Theta}(k)}, \dots, \frac{\partial q}{\partial \Theta_D}\Big|_{\mathbf{\Theta}=\mathbf{\Theta}(k)} \right)^t$$

Descenso por gradiente: algoritmo general

$$\begin{array}{lcl} \boldsymbol{\Theta}(1) & = & \text{arbitrario} \\ \boldsymbol{\Theta}(k+1) & = & \boldsymbol{\Theta}(k) - \rho_k \boldsymbol{\nabla} q(\boldsymbol{\Theta}) \,|_{\boldsymbol{\Theta}=\boldsymbol{\Theta}(k)} \end{array}$$

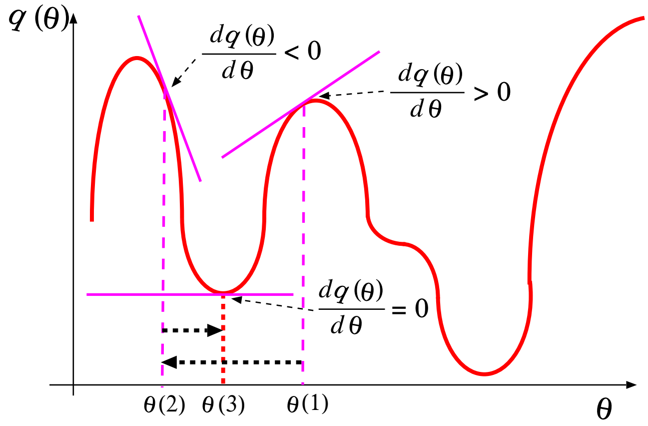
Donde $\rho_k \in \mathbb{R}^{>0}$ es un *factor de aprendizaje*

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

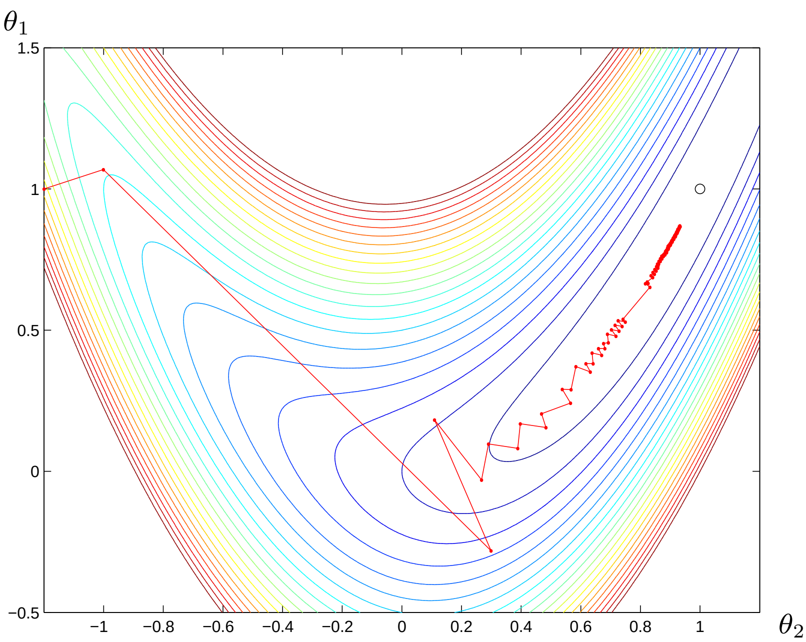
$$\theta(2) = \theta(1) - \rho_1 \frac{dq}{d\theta}\bigg|_{\theta(1)}$$

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

$$\theta(3) \equiv \theta^\star, \quad \frac{dq}{d\theta}\bigg|_{\theta(3)} = 0$$



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



Curvas de nivel de la función de Rosenbrock[1] $q(\theta_1, \theta_2) = 10(\theta_1 - \theta_2^2)^2 + (\theta_1 - 1)^2$ y trayectoria seguida por el vector de parámetros $\boldsymbol{\theta} = (\theta_1, \theta_2)^t$.

1. Figuras basadas en la presentación de R. Hauser http://people.maths.ox.ac.uk/hauser/hauser_lecture2.pdf.

Convergencia y factor de aprendizaje

- TEOREMA GENERAL DE CONVERGENCIA:

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

$$H_{ij}(q, \mathbf{\Theta}) \stackrel{\text{def}}{=} \frac{\partial^2 q(\mathbf{\Theta})}{\partial \Theta_i \, \partial \Theta_j}$$

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

  Si $|1-\lambda_l(k)\rho_k|<1 \; \forall l$, *entonces* $\mathbf{\Theta}(k)$ *tiende a un minimo local de* $q(\mathbf{\Theta})$ *cuando* $k \to \infty$

- INFLUENCIA DEL FACTOR DE APRENDIZAJE:
  - $\rho < 2/\lambda_{\max}$ 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):

$$f(\boldsymbol{x}) = \underset{1 \le c \le 2}{\arg\max} \; \phi_c(\boldsymbol{x}), \quad \boldsymbol{x} = (x_1, \ldots, x_d)^t \in \mathbb{R}^d, \quad \phi_c : \mathbb{R}^d \to \mathbb{R}, \; 1 \le c \le 2$$

  Cada FDL ϕ_c está definida por un vector de pesos $\boldsymbol{\theta}_c \in \mathbb{R}^D$ donde $D = d+1$

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

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

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

$$f_{\boldsymbol{\theta}}(\boldsymbol{x}) \;=\; \begin{cases} +1 & \text{si } \; \boldsymbol{\theta}^t \boldsymbol{x} \ge 0 \\ -1 & \text{si } \; \boldsymbol{\theta}^t \boldsymbol{x} < 0 \end{cases}$$

Aprendizaje de funciones discriminantes lineales

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

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

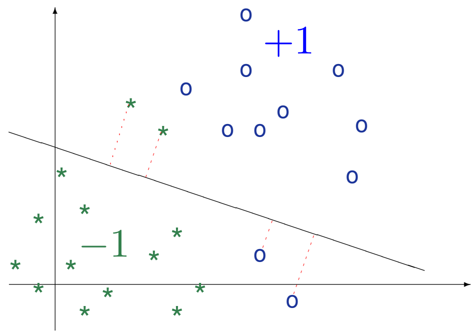
- Aprendizaje: Dada S, encontrar un vector de pesos $\hat{\boldsymbol{\theta}}$ que la separe; es decir, que satisfaga el sistema de N inecuaciones:

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

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

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

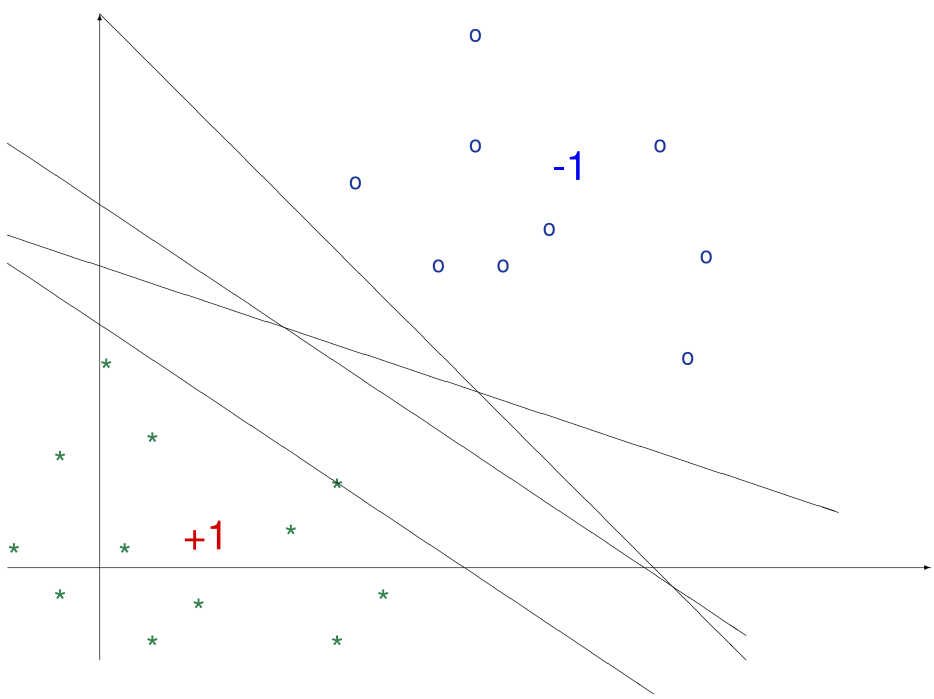
  proporcional a la suma de
  segmentos punteados $\longrightarrow$



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

Aprendizaje de funciones discriminantes lineales

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



Algoritmo perceptrón

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

$$\boxed{\begin{aligned} \boldsymbol{\theta}(1) &= \text{arbitrario} \\ \boldsymbol{\theta}(k+1) &= \boldsymbol{\theta}(k) + \rho_k \sum_{\substack{(\boldsymbol{x},c)\in S \\ c\ \boldsymbol{\theta}^t\boldsymbol{x} < 0}} c\,\boldsymbol{x} \end{aligned}}$$

Algoritmo perceptrón muestra a muestra ("online"):

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

TEOREMA DEL PERCEPTRÓN:

Si S es LS y ρ_k es positivo y decreciente o creciente sublinealmente con k, el algorithmo perceptrón converge a una solución en un número finito de iteraciones

Regresión lineal mediante descenso por gradiente

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

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

  y S una muestra de entrenamiento:

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

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

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

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

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

- Solución: descenso por gradiente.

Algoritmo de Widrow-Hoff (Adaline)

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

$$\boxed{\begin{aligned} \boldsymbol{\theta}(1) &= \text{arbitrario} \\ \boldsymbol{\theta}(k+1) &= \boldsymbol{\theta}(k) + \rho_k \sum_{n=1}^{N} (y_n - \boldsymbol{\theta}(k)^t \boldsymbol{x}_n)\, \boldsymbol{x}_n \end{aligned}}$$

Algoritmo muestra a muestra:

$$\begin{aligned} \boldsymbol{\theta}(1) &= \text{arbitrario} \\ \boldsymbol{\theta}(k+1) &= \boldsymbol{\theta}(k) + \rho_k \left(y(k) - \boldsymbol{\theta}(k)^t \boldsymbol{x}(k) \right) \boldsymbol{x}(k) \end{aligned}$$

TEOREMA:

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

Index

Aprendizaje de modelos probabilísticos con variables latentes

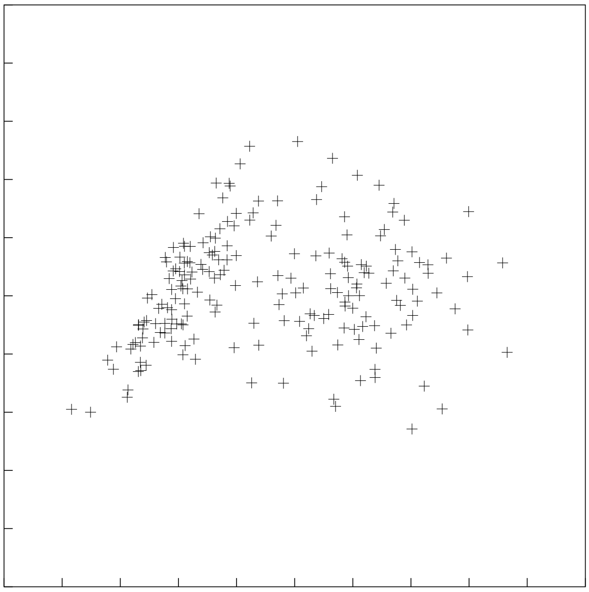
- Se suele usar el criterio de *máxima verosimilitud*; es decir, $q_S(\mathbf{\Theta}) \equiv L_S(\mathbf{\Theta})$
- En ocasiones los datos observados *no* contienen suficiente información sobre cómo han sido generados por los modelos probabilísticos asumidos
- Por ejemplo, en los modelos ocultos de 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 gaussianas y modelo generador

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

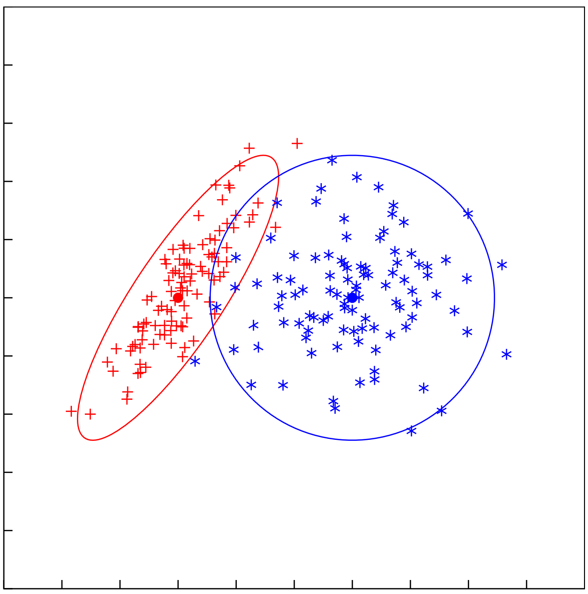
$$
\begin{aligned}
p(\boldsymbol{x}) &= \sum_{k=1}^{K} p(k, \boldsymbol{x}) = \sum_{k=1}^{K} P(k)\, p(\boldsymbol{x} \mid k) \equiv \sum_{k=1}^{K} \pi_k\, p(\boldsymbol{x} \mid \boldsymbol{\mu}_k, \Sigma_k) \\
&\equiv p(\boldsymbol{x} \mid \mathbf{\Theta}); \quad \mathbf{\Theta} \stackrel{\text{def}}{=} [\pi_1, \ldots, \pi_k, \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k, \Sigma_1 \ldots, \Sigma_k]
\end{aligned}
$$

Mezcla de gaussianas: ilustración



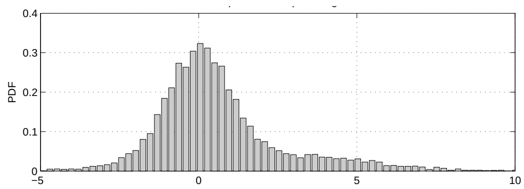
Datos *observables* generados por una mezcla de dos gaussianas.

Mezcla de gaussianas: ilustración

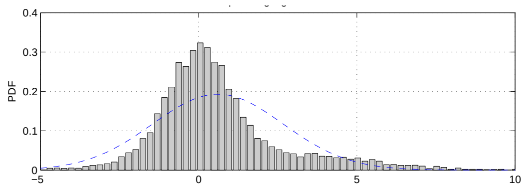


Datos de una mezcla de dos gaussianas con la variable oculta expuesta.
Las elipses muestran los parámetros de las gaussianas del modelo generador.

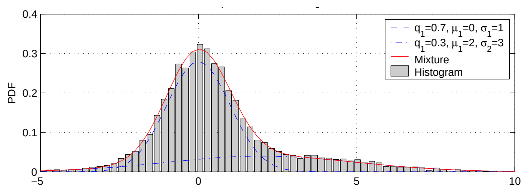
Mezcla de gaussianas: otro ejemplo en 1D



Histograma de una muestra de una variable unidimensional



Una única gaussiana estimada a partir de la muestra



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

Parámetros de una mezcla de gaussianas

- Para simplificar, suponemos que la matriz de covarianzas, Σ, de todas las K gaussianas de la mezcla es la misma y es fija y conocida:

$$
\begin{aligned}
p(\boldsymbol{x} \mid \boldsymbol{\Theta}) &= \sum_{k=1}^{K} \pi_k \, p(\boldsymbol{x} \mid \boldsymbol{\mu}_k) \\
&= (2\pi)^{-d/2} |\Sigma|^{-1/2} \sum_{k=1}^{K} \pi_k \, \exp \left\{ -\frac{1}{2} (\boldsymbol{x} - \boldsymbol{\mu}_k)^t \Sigma^{-1} (\boldsymbol{x} - \boldsymbol{\mu}_k) \right\}
\end{aligned}
$$

donde $\boldsymbol{\Theta} \equiv [\pi_1, \ldots, \pi_K, \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K]$, $\displaystyle\sum_{k=1}^{K} \pi_k = 1$

Mezcla de K gaussianas en $\mathbb{R}^d$: variables latentes

Sea $\boldsymbol{\Theta} = (\pi_k, \boldsymbol{\mu}_k, 1 \leq k \leq K)$ los parámetros de la mezcla de gaussianas $p(\boldsymbol{x} \mid \boldsymbol{\Theta})$ y sea $S = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$, $\boldsymbol{x}_n \in \mathbb{R}^d, 1 \leq n \leq N$, una muestra de esta distribución.

- Para cada dato $\boldsymbol{x}_n$, hay K *variables latentes* $z_{kn} \in \{0,1\}, 1 \leq k \leq K$
- $z_{kn} = 1 \Leftrightarrow \boldsymbol{x}_n$ ha sido generado por la gaussiana k; por tanto: $\sum_{k=1}^{K} z_{kn} = 1 \;\; \forall n$ y π_k es la prob. *"a priori"* (independiente de $\boldsymbol{x}_n$) de que $z_{kn} = 1$
- *Valor esperado de* z_{kn}, dado $\boldsymbol{x}_n$ y $\boldsymbol{\Theta}$:

$$\begin{aligned}
\hat{z}_{kn} \stackrel{\text{def}}{=} E(z_{kn}) &= \sum_{z_{kn}=0}^{1} z_{kn} P(z_{kn} \mid \boldsymbol{x}_n, \boldsymbol{\Theta}) \\
= P(z_{kn} = 1 \mid \boldsymbol{x}_n, \boldsymbol{\Theta}) &= \frac{P(z_{kn} = 1 \mid \boldsymbol{\Theta})\, p(\boldsymbol{x}_n \mid z_{kn} = 1, \boldsymbol{\Theta})}{p(\boldsymbol{x}_n \mid \boldsymbol{\Theta})} \\
= \frac{\pi_k\, p(\boldsymbol{x}_n \mid \boldsymbol{\mu}_k)}{p(\boldsymbol{x}_n \mid \boldsymbol{\Theta})} &= \frac{\pi_k\, p(\boldsymbol{x}_n \mid \boldsymbol{\mu}_k)}{\sum_{k'=1}^{K} \pi_{k'}\, p(\boldsymbol{x}_n \mid \boldsymbol{\mu}_{k'})}
\end{aligned}$$

Estimación de parámetros de una mezcla de gaussianas

- El logaritmo de la verosimilitud de la muestra $S = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$, dada una estimación de las variables latentes $\{\hat{\boldsymbol{z}}_1, \ldots, \hat{\boldsymbol{z}}_N\}$ es:

$$L'_S(\boldsymbol{\Theta}) = \log \prod_{n=1}^{N} p(\boldsymbol{x}_n \mid \hat{\boldsymbol{z}}_n; \boldsymbol{\Theta}) = \sum_{n=1}^{N} \log \sum_{k=1}^{K} \pi_k\, p(\boldsymbol{x}_n \mid \hat{\boldsymbol{z}}_n; \boldsymbol{\mu}_k)$$

- Estimación de $\boldsymbol{\Theta} \equiv [\pi_1, \ldots, \pi_K, \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K]$ por máxima verosimilitud:

$$\hat{\boldsymbol{\Theta}} = \underset{\substack{\boldsymbol{\Theta} \\ \sum_{k=1}^{K} \pi_k = 1}}{\arg\max}\; L'_S(\boldsymbol{\Theta})$$

Optimización analítica:

$$1 \leq k \leq N: \quad \boldsymbol{\nabla}_{\boldsymbol{\mu}_k} L'_S(\boldsymbol{\Theta}) = 0 \;\rightarrow\; \boldsymbol{\mu}_k = \frac{1}{\pi_k N} \sum_{n=1}^{N} \hat{z}_{kn}\, \boldsymbol{x}_n$$

Multiplicadores de Lagrange [$\Lambda(\boldsymbol{\Theta}, \beta) = L'_S(\boldsymbol{\Theta}) + \beta(1 - \sum_{k=1}^{K} \pi_k)$]:

$$1 \leq k \leq K: \; \frac{\partial \Lambda}{\partial \pi_k} = 0 \;\rightarrow\; \pi_k = \sum_{n=1}^{N} \frac{\hat{z}_{kn}}{\beta} \rightarrow [\frac{\partial \Lambda_D}{\partial \beta} = 0] \rightarrow \; \pi_k = \frac{1}{N} \sum_{n=1}^{N} \hat{z}_{kn}$$

Estimación de parámetros de una mezcla de gaussianas: Algoritmo EM

1. Inicializar $i = 0$ y $\boldsymbol{\Theta}(i) \equiv (\pi_k(i), \boldsymbol{\mu}_k(i),\ 1 \le k \le K)$ con valores adecuados

2. Iterar hasta convergencia:

   - **Paso E** (*"expectación"* o cálculo de la esperanza de variables latentes):

$$\left.\begin{array}{c} 1 \le n \le N \\ 1 \le k \le K \end{array}\right\} : \quad \hat{z}_{kn} \;=\; \frac{\pi_k(i)\ p(\boldsymbol{x}_n \mid \boldsymbol{\mu}_k(i))}{\sum_{k'} \pi_{k'}(i)\ p(\boldsymbol{x}_n \mid \boldsymbol{\mu}_{k'}(i))}$$

   - **Paso M** (*"maximización"* de la log-verosimilitud, $L_S(\boldsymbol{\Theta})$):

$$1 \le k \le K : \quad \left\{ \begin{array}{rcl} \pi_k(i+1) & = & \dfrac{1}{N} \displaystyle\sum_{n=1}^{N} \hat{z}_{kn} \\[2ex] \boldsymbol{\mu}_k(i+1) & = & \dfrac{1}{\pi_k N} \displaystyle\sum_{n=1}^{N} \hat{z}_{kn}\ \boldsymbol{x}_n \end{array} \right.$$

   - $i \leftarrow i + 1$

Algoritmo EM basado en función Q

- Se define una función de $\boldsymbol{\Theta}$ a partir de valores dados (previos) de $\boldsymbol{\Theta}'$:

$$Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}') \stackrel{\text{def}}{=} E_{\boldsymbol{z}|\boldsymbol{x},\boldsymbol{\Theta}'}(\log P(\boldsymbol{x}, \boldsymbol{z} \mid \boldsymbol{\Theta})) \;=\; \sum_{\boldsymbol{z}} \sum_{n=1}^{N} \log P(\boldsymbol{x}_n, \boldsymbol{z} \mid \boldsymbol{\Theta})\ P(\boldsymbol{z} \mid \boldsymbol{x}_n, \boldsymbol{\Theta}')$$

- Propiedades: 1) $Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}')$ es convexa; 2) $Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}') \le L_S(\boldsymbol{\Theta})\ \forall \boldsymbol{\Theta}, \boldsymbol{\Theta}'$

Algoritmo EM basado en Q:

1. Inicializar $\boldsymbol{\Theta}(0)$; $i = 0$

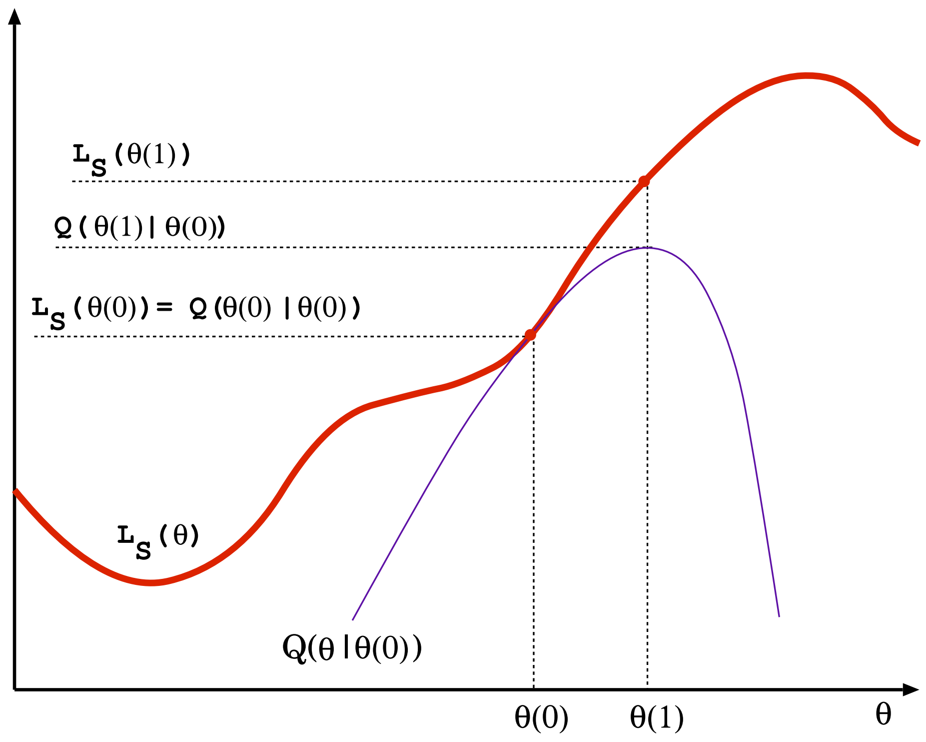
2. Repetir, hasta convergencia:

   - Paso E (*esperanza*): Obtener $Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}(i))$ a partir de $\boldsymbol{\Theta}(i)$

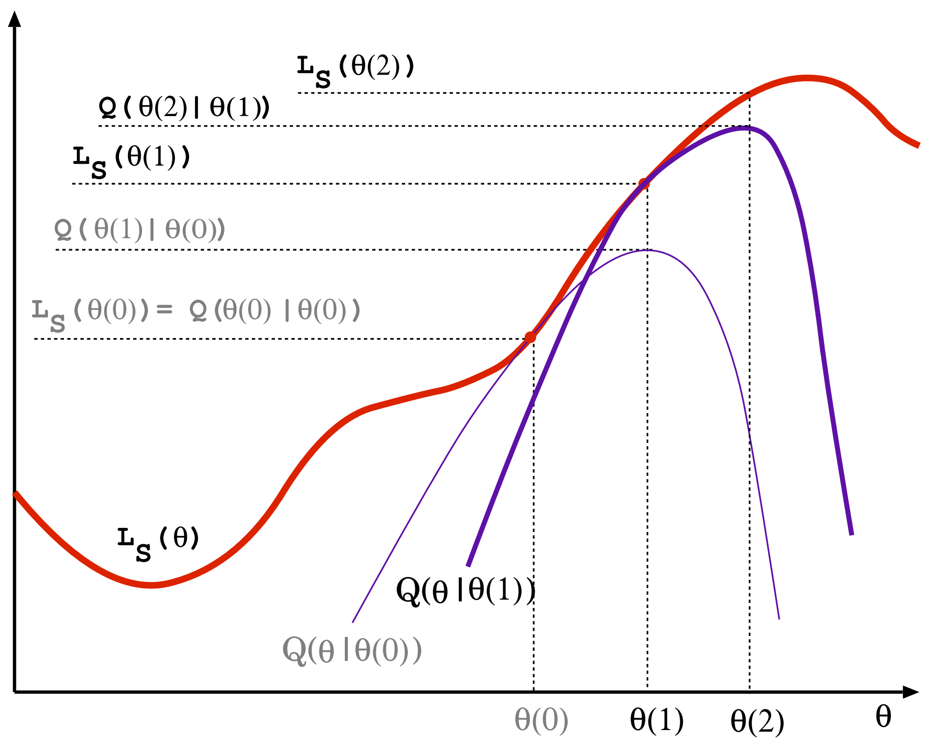
   - Paso M (*maximización*): Calcular $\boldsymbol{\Theta}(i+1) \;=\; \arg\max_{\boldsymbol{\Theta}} Q(\boldsymbol{\Theta}, \boldsymbol{\Theta}(i))$

   - $i = i + 1$

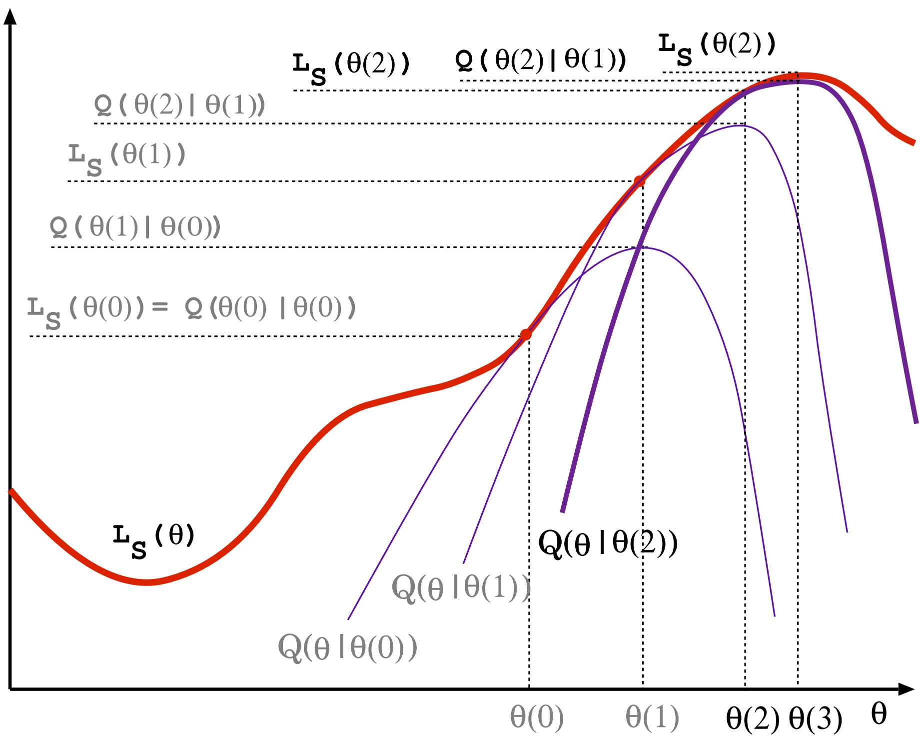
Propiedades y convergencia del EM



Propiedades y convergencia del EM



Propiedades y convergencia del EM



Notación

- $\boldsymbol{\Theta} = (\Theta_1, \ldots, \Theta_D)^t$: vector de parámetros. Como los vectores son matrices columna, para representar las componentes en fila se usa la t (*"transpuesta"*).

- $q_S(\boldsymbol{\Theta})$: función objetivo a optimizar definida sobre un conjunto de entrenamiento S, cuyos de parámetros son $\boldsymbol{\Theta}$

- $\boldsymbol{\nabla} q(\boldsymbol{\Theta}) \stackrel{\text{def}}{=} \left(\frac{\partial q(\boldsymbol{\Theta})}{\partial \Theta_1}, \ldots, \frac{\partial q(\boldsymbol{\Theta})}{\partial \Theta_D} \right)^t$: gradiente de la función q_s
  (vector de derivadas parciales con respecto a cada componente de $\boldsymbol{\Theta}$)

- $\boldsymbol{\nabla} q(\boldsymbol{\Theta}) \mid_{\boldsymbol{\Theta}=\boldsymbol{\Theta}(k)} \equiv \left(\frac{\partial q}{\partial \Theta_1} \Big|_{\boldsymbol{\Theta}=\boldsymbol{\Theta}(k)}, \ldots, \frac{\partial q}{\partial \Theta_D} \Big|_{\boldsymbol{\Theta}=\boldsymbol{\Theta}(k)} \right)^t$: gradiente de la función q calculado en $\boldsymbol{\Theta}(k)$

- Σ : matriz de covarianzas de una distribución gaussiana