Clasificación Mediante Modelos Lineales

Reconocimiento de Patrones

Departamento de Procesamiento de Señales
Instituto de Ingeniería Eléctrica
Facultad de Ingeniería, UdelaR

2018

Bishop, Cap. 4 – Duda et al., Cap. 5 – Hastie et al., Cap 4

Intro - Clasificación: definiciones

Clasificar: asignar cada vector de entrada $\mathbf{x} \in \mathbb{R}^d$ a una de las c clases ω_k, $k = 1, \ldots, c$.
Suponemos: clases incompatibles (regiones de decisión disjuntas).

Modelos lineales: las superficies de decisión son funciones lineales de los datos, hiperplanos de dimensión $d - 1$.

Datos linealmente separables: muestras de distintas clases separables por hiperplanos.

Clasificación: enfoques diversos

Tres enfoques:

- Funciones discriminantes: asignar directamente $\mathbf{x}$ a una clase. Forma de las funciones discriminantes conocida. Distribuciones cualesquiera.
- Modelado de las distribuciones a posteriori $p(\omega_k|\mathbf{x})$, luego clasificación/decisión óptima. Dos formas de estimar $p(\omega_k|\mathbf{x})$:
  - Modelo discriminativo: directamente, e.g. modelo paramétrico y determinación de parámetros con muestras de aprendizaje.
  - Modelo generativo: se modela $p(\mathbf{x}|\omega_k)$ y $P(\omega_k)$, luego Bayes (e.g. mezcla de Gaussianas)

Agenda

Funciones discriminantes

Para simplificar, $c = 2$ ($c > 2$ más adelante).

Discriminantes lineales:

- $g(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0$.
- Si $g(\mathbf{x}) \geq 0$, se asigna $\mathbf{x}$ a ω_1, si no a ω_2.
- Fronteras de decisión: hiperplanos

Modelos lineales generalizados:

- $g(\mathbf{x}) = f(\mathbf{w}^T\mathbf{x} + w_0)$, f no lineal "función de activación"
- Fronteras de decisión: $g(\mathbf{x}) = cte$, i.e. hiperplanos.

Más general: la misma teoría aplicada en un espacio transformado $\phi(\mathbf{x})$, ϕ no lineal.

- Frontera de decisión: hiperplanos en el espacio transformado, no lineales en el original

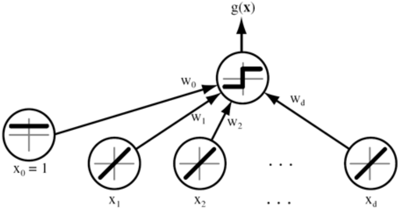
Funciones discriminantes lineales

Dos clases ($c = 2$).

$$g(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0$$

$\mathbf{w}$ - vector de pesos (para las características)
w_0 - valor del umbral

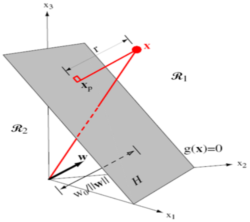
$$\text{Decisión: si } g(\mathbf{x}) \begin{cases} \geq 0 \\ < 0 \end{cases} \text{ entonces } \mathbf{x} \text{ de clase } \begin{cases} \omega_1 \\ \omega_2 \end{cases}$$



Frontera de decisión

$\mathcal{S} = \{\mathbf{x} \mid g(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0 = 0\}$: hiperplano de dim. $d-1$.

- $\mathbf{w}$ normal a $\mathcal{S}$: $\mathbf{x_1}, \mathbf{x_2} \in \mathcal{S} \Rightarrow \mathbf{w}^T(\mathbf{x_2} - \mathbf{x_1}) = 0$.



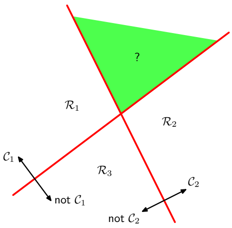
- Distancia del plano al origen: $\frac{\mathbf{w}^T}{\|\mathbf{w}\|}\mathbf{x} = -\frac{w_0}{\|\mathbf{w}\|}$.
- $\mathbf{x_p} \in \mathcal{S}$, $\mathbf{x} = \mathbf{x_p} + r\frac{\mathbf{w}}{\|\mathbf{w}\|}$
  $\Rightarrow g(\mathbf{x}) = \mathbf{w}^T\mathbf{x_p} + r\mathbf{w}^T\frac{\mathbf{w}}{\|\mathbf{w}\|} + w_0 = r\|\mathbf{w}\|$
  $\Rightarrow r = \frac{g(\mathbf{x})}{\|\mathbf{w}\|}$: $g(\mathbf{x})$ medida de distancia (signada) de $\mathbf{x}$ a $\mathcal{S}$.

Múltiples clases ($c > 2$)

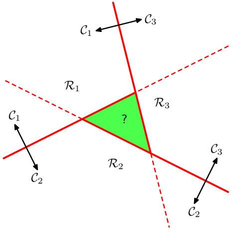
Estrategias posibles para extender el caso $c = 2$:

- $c - 1$ clasificadores binarios, ω_k o el resto ("uno contra todos")
- $c(c-1)/2$ parejas (i, j), ω_i o ω_j ("uno contra uno'), asignación por voto mayoritario.

Ambas fallan (regiones ambiguas) ...



uno contra todos



uno contra uno

Múltiples clases ($c > 2$)

Idea: un único c-discriminante con c funciones lineales

- $g_k(\mathbf{x}) = \mathbf{w}_k^T \mathbf{x} + w_{k,0}, \quad k = 1, \ldots, c.$
- $\mathbf{x}$ se asigna a ω_k si $g_k(\mathbf{x}) > g_j(\mathbf{x}) \ \forall j \neq k$

Superficies de decisión: entre ω_k y ω_j: $g_k(\mathbf{x}) = g_j(\mathbf{x})$

$$\Leftrightarrow (\mathbf{w}_k - \mathbf{w}_j)^T \mathbf{x} + (w_{k,0} - w_{j,0}) = 0,$$ hiperplano dim.$d - 1$.

Compatible con $c = 2$.

Ejercicio Regiones de decisión: simplemente conexas y convexas:

$$\forall\ x_A, x_B \in \mathcal{R}_k, \forall\ \lambda \in [0,1], \text{ tenemos } (\lambda x_A + (1-\lambda) x_B) \in \mathcal{R}_k.$$

Funciones lineales generalizadas

Ejemplo: formas cuadráticas

$$g(\mathbf{x}) = \sum_{i=1}^{d}\sum_{j=1}^{d} w_{ij}x_ix_j + \sum_{i=1}^{d} w_ix_i + w_0 = \mathbf{x}^t\mathbf{W}\mathbf{x} + \mathbf{w}^t\mathbf{x} + w_0,$$

donde $w_{ij} = w_{ji}$.

- $d(d+1)/2$ parámetros adicionales (a estimar).
- Las superficies de decisión $\mathcal{S}$ son cuádricas.
- Si $\mathbf{W}$ no singular: recentrando extremo de $\mathcal{S}$ en origen:

$$\mathbf{x}^t\hat{\mathbf{W}}\mathbf{x} = 1, \text{ con } \hat{\mathbf{W}} = \frac{4\mathbf{W}}{\mathbf{w}^t\mathbf{W}^{-1}\mathbf{w} - 4w_0}.$$

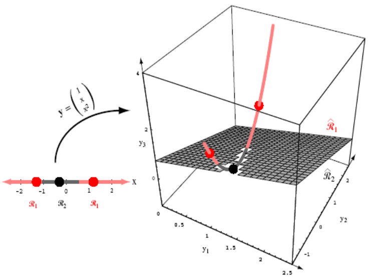
  - Si $\hat{\mathbf{W}}$ def positiva, $\mathcal{S}$ es un hiper-elipsoide.
  - Si no, $\mathcal{S}$ es alguna variedad de hiper-hiperboloide.

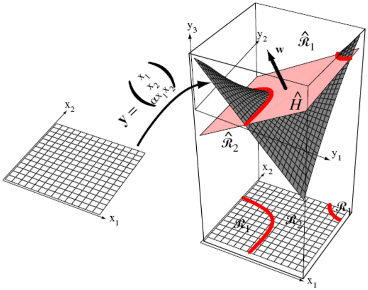
Discriminante polinomial, generalización

- Agregando combinaciones cúbicas o de mayor orden se tienen funciones discriminantes polinomiales.
- Caso general: $\mathbf{y} = \phi(\mathbf{x})$, con ϕ no lineal. La función discriminante ya no es lineal con $\mathbf{x}$ pero sí con $\mathbf{y}$.
- Se pasa a una **dimensión mayor** donde:
  - Es más fácil que los datos $\mathbf{y} = \phi(\mathbf{x})$ se vuelvan lo más linealmente separables.
  - Se usan discriminantes lineales en $\mathbf{y}$.

Discriminante polinomial: ejemplos

$$\mathbf{y} = \phi_1(x) = [1, x, x^2]^T \qquad \mathbf{y} = \phi_2(x_1, x_2) = [x_1, x_2, \alpha x_1 x_2]^T$$





Determinando los parámetros de los discriminantes

$$g_k(\mathbf{x}) = \mathbf{w}_k^T \mathbf{x} + w_{k,0}, \quad k = 1, \dots, c.$$

Datos: N muestras etiquetadas $\{\mathbf{x}_n, \mathcal{C}_n\}$, $\mathcal{C}_n \in \{\omega_1, \omega_2, \dots, \omega_c\}$.

Veremos tres formas de estimar los $(\mathbf{w}_k, w_{0,k})$:

- Mínimos cuadrados: minimizar error cuadrático para $\{\mathbf{x}_n, \mathbf{t}_n\}$, $\mathbf{t}_n = (t_{n,1}, \dots, t_{n,c})^T \in \{0,1\}^c$ con $t_{n,k} = \mathbb{1}_{\{\mathcal{C}_n = \omega_k\}}$.
- Discriminante de Fisher: buscar las direcciones de mayor separación de clases.
- Perceptrón: minimizar cantidad de muestras mal clasificadas (o variaciones).

Clasificación con mínimos cuadrados

Representación en coord. homogéneas:

$$g_k(\mathbf{x}) = \mathbf{w}_k^T\mathbf{x} + w_{k,0} = \underbrace{(w_{k,0}, \mathbf{w}_k^T)}_{\mathbf{a}_k^T} \cdot \underbrace{(1, \mathbf{x}^T)^T}_{\tilde{\mathbf{x}}}$$

$$\mathbf{g}(\mathbf{x}) = \begin{pmatrix} g_1(\mathbf{x}) \\ g_2(\mathbf{x}) \\ \vdots \\ g_c(\mathbf{x}) \end{pmatrix} = \begin{pmatrix} \mathbf{a}_1^T \\ \mathbf{a}_2^T \\ \vdots \\ \mathbf{a}_c^T \end{pmatrix} \tilde{\mathbf{x}} = \mathbf{A}^T\tilde{\mathbf{x}}$$

Objetivo:

determinar $\mathbf{A}$ que minimice $E(\mathbf{A}) = \frac{1}{2}\sum_{n=1}^{N} \|\mathbf{t}_n - \mathbf{A}^T\tilde{\mathbf{x}}_n\|_2^2$

Clasificación con mínimos cuadrados

Tenemos:

$$E(\mathbf{A}) = \frac{1}{2}\,\mathrm{Trace}\left\{(\tilde{\mathbf{X}}\mathbf{A} - \mathbf{T})^T(\tilde{\mathbf{X}}\mathbf{A} - \mathbf{T})\right\}$$

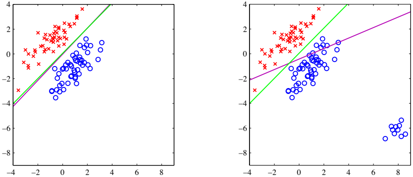
$$\text{con } \tilde{\mathbf{X}} = \begin{pmatrix} \tilde{\mathbf{x}}_1^T \\ \tilde{\mathbf{x}}_2^T \\ \vdots \\ \tilde{\mathbf{x}}_N^T \end{pmatrix}, \ \mathbf{T} = \begin{pmatrix} \mathbf{t}_1^T \\ \mathbf{t}_2^T \\ \vdots \\ \mathbf{t}_N^T \end{pmatrix}.$$

$$0 = \frac{\partial}{\partial \mathbf{A}} E(\mathbf{A}) = (\tilde{\mathbf{X}}^T\tilde{\mathbf{X}})\mathbf{A} - \tilde{\mathbf{X}}^T\mathbf{T}$$

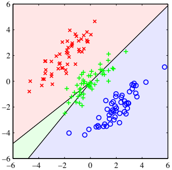
$$\Leftrightarrow \boxed{\mathbf{A} = (\tilde{\mathbf{X}}^T\tilde{\mathbf{X}})^{-1}\tilde{\mathbf{X}}^T\mathbf{T}}$$

Clasificación con mínimos cuadrados: resultados

Dos clases: min. cuadrados, target binarios (magenta), logistic regression (verde)



Tres clases: min. cuadrados, target binarios (izq.), logistic regression (der.)



Interpretar

Clasificación con mínimos cuadrados: limitantes

Limitantes:

- Poco robusto (puntos lejanos a la frontera tienen demasiado peso relativo)
- Comportamiento pobre para múltiples clases

Porqué falla?

- Mínimos cuadrados es un estimador óptimo para datos Gaussianos (estimador de máxima verosimilitud)
- Outliers y targets binarios lejos de la gaussianidad.

Discriminante lineal de Fisher

Comenzamos con $c = 2$.

N_1 puntos de la clase ω_1, N_2 puntos de la clase ω_2.

Idea: encarar el problema desde la óptica de reducción de dimensionalidad.

- Proyección el espacio de dim d a dim 1: $y = \mathbf{w}^T\mathbf{x}$.
- Decidimos por umbralización: si $y \geq w_0$, asignamos $\mathbf{x}$ a ω_1, de lo contrario a ω_2.

¿Entre los sub-espacios de dimensión 1 (generados por $\{\mathbf{w}\}$), cuál asegura mayor separación?

Discriminante lineal de Fisher

Necesitamos definir criterio de separación.

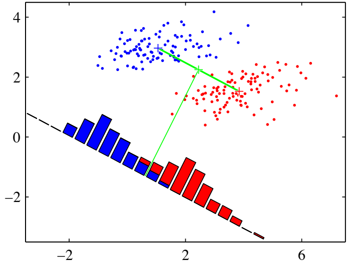
1er intento: $\mathbf{w}$ como la dirección de máxima distancia entre las medias $\mathbf{m}_1 = \frac{1}{N_1}\sum_{\mathbf{x}_n\in\omega_1}\mathbf{x}_n$ y $\mathbf{m}_2 = \frac{1}{N_2}\sum_{\mathbf{x}_n\in\omega_2}\mathbf{x}_n$ es máxima

$$\Rightarrow \arg\max_{\|\mathbf{w}\|=1}\left\{\mathbf{w}^T(\mathbf{m}_2-\mathbf{m}_1)\right\} = \frac{\mathbf{m}_2-\mathbf{m}_1}{\|\mathbf{m}_2-\mathbf{m}_1\|}.$$

Pregunta **Es un buen criterio?**

Discriminante lineal de Fisher

... Sabemos que no. Sólo tiene sentido cuando los datos de cada clase tienen matriz de covarianza proporcional a identidad.



Problema: gran dispersión de las clases en la proyección.

Ejercicio Proponer otro criterio

Discriminante lineal de Fisher

$= \sum w^T(x_n - m_k)(x_n - m_k)^T w$

2do intento: $\mathbf{w}$ como la dirección que permite a la vez:

$= w^T \left[\sum_{x_n \in \omega_k} (x_n - m_k)(x_n - m_k)^T\right] w$

(*)

- Mayor distancia entre las medias proyectadas,
- Menores dispersiones en la proyección.

  Dispersiones: $s_k^2 = \sum_{\mathbf{x}_n \in \omega_k} (w^T(\mathbf{x}_n - \mathbf{m}_k))^2, \quad k = 1, 2$

Criterio: maximizar $\boxed{J(\mathbf{w}) = \dfrac{\left(\mathbf{w}^T(\mathbf{m}_2 - \mathbf{m}_1)\right)^2}{s_1^2 + s_2^2} = \dfrac{\mathbf{w}^T \mathbf{S}_B \mathbf{w}}{\mathbf{w}^T \mathbf{S}_W \mathbf{w}}}$

con $\mathbf{S}_B$, $\mathbf{S}_W$: between-class and within-class covariance matrices

$$\mathbf{S}_B = (\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^T, \quad \mathbf{S}_W = \sum_{k=1}^{2} \sum_{\mathbf{x}_n \in \omega_k} (\mathbf{x}_n - \mathbf{m}_k)(\mathbf{x}_n - \mathbf{m}_k)^T$$

(*)

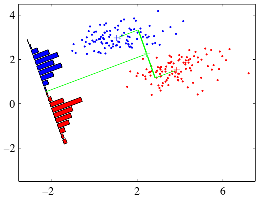
Ejercicio Encontrar $\mathbf{w}$

$= w^T \sum_{x_n \in \omega_1} (x_n - m_1)(x_n - m_1)^T w + w^T \sum_{x_n \in \omega_2} (x_n - m_2)(x_n - m_2)^T w = w^T S_B w$

Discriminante lineal de Fisher

$$\frac{\partial J}{\partial \mathbf{w}} = 0 \Leftrightarrow \quad (\mathbf{w}^T \mathbf{S}_B \mathbf{w}) \mathbf{S}_W \mathbf{w} = (\mathbf{w}^T \mathbf{S}_W \mathbf{w}) \mathbf{S}_B \mathbf{w} \propto \mathbf{m}_2 - \mathbf{m}_1$$

$$\Rightarrow \boxed{\mathbf{w} \propto \mathbf{S}_W^{-1} (\mathbf{m}_2 - \mathbf{m}_1)}$$



Cómo fijar w_0: e.g. umbral óptimo entre dos Gaussianas por máxima verosimilitud. Tiene cierto sentido (TCL): características independientes $\Rightarrow y = \mathbf{w}^T \mathbf{x}$ suma de VAs independientes.

Discriminante de Fisher: relación con mínimos cuadrados $(c = 2)$

N_1 puntos de la clase ω_1, N_2 puntos de la clase ω_2, $N = N_1 + N_2$.

En el enfoque de mínimos cuadrados:

$$(\mathbf{w}, w_0) = \arg\min \frac{1}{2} \sum_{n=1}^{N} \left(\mathbf{w}^T \mathbf{x}_n + w_0 - t_n\right)^2 .$$

Cond. de optimalidad (derivadas nulas):

$$\frac{\partial}{\partial w_0} : \quad \sum_{n=1}^{N} \left(\mathbf{w}^T \mathbf{x}_n + w_0 - t_n\right) = 0$$

$$\frac{\partial}{\partial \underline{w}} : \quad \sum_{n=1}^{N} \left(\mathbf{w}^T \mathbf{x}_n + w_0 - t_n\right) \mathbf{x}_n = 0$$

Discriminante de Fisher: relación con mínimos cuadrados $(c = 2)$

$$\text{Si definimos } t_n = \begin{cases} N/N1 & \text{si } \mathbf{x} \text{ es de la clase } \omega_1 \\ -N/N2 & \text{si } \mathbf{x} \text{ es de la clase } \omega_2 \end{cases},$$

Operando se obtiene:

$$w_0 = -\mathbf{w}^T \mathbf{m}$$

$$\left(\mathbf{S}_W + \frac{N_1 N_2}{N} \mathbf{S}_B\right) \mathbf{w} = N(\mathbf{m}_1 - \mathbf{m}_2) \Rightarrow \mathbf{w} \propto \mathbf{S}_W^{-1}(\mathbf{m}_2 - \mathbf{m}_1)$$

- Estrictamente vale por ende bajo hipótesis de Gaussianidad.
- Sufre de los mismos males que cualquier procedimiento de mínimos cuadrados.

Discriminante de Fisher: caso $c > 2$

- d' direcciones de proyección, $d' < d$
- Tiene sentido $d' \leq c - 1$, ideal $d' = c - 1$ Al igual que antes, $y_k = \mathbf{w}_k^T \mathbf{x},\ k = 1, \ldots, d'$, $\mu_k = w_k^T m_k$

$$\mathbf{y} = \begin{pmatrix} \mathbf{w}_1^T \\ \mathbf{w}_2^T \\ \vdots \\ \mathbf{w}_{d'}^T \end{pmatrix} \mathbf{x} = \mathbf{W}^T \mathbf{x}$$

- **Covarianzas en espacio proyectado (Inter e Intra-clase):**

$$\mathbf{s}_B = \sum_{k=1}^{c} N_k (\boldsymbol{\mu}_k - \boldsymbol{\mu})(\boldsymbol{\mu}_k - \boldsymbol{\mu})^T, \mathbf{s}_W = \sum_{k=1}^{c} \sum_{\mathbf{x}_n \in \omega_k} (\mathbf{y}_n - \boldsymbol{\mu}_k)(\mathbf{y}_n - \boldsymbol{\mu}_k)^T$$

Criterio posible: elegir los $\mathbf{w}_k$ para maximizar $\mathrm{Trace}\left\{\mathbf{s}_W^{-1}\mathbf{s}_B\right\}$

Discriminante de Fisher: caso $c > 2$

Obs: relación entre covarianzas en espacio original y proyectado

$$\mathbf{s}_W = \mathbf{W}^T \mathbf{S}_W \mathbf{W}, \;\; \mathbf{s}_B = \mathbf{W}^T \mathbf{S}_B \mathbf{W}, \;\; \mathbf{W} \in \mathbb{R}^{d \times d'}$$

con

$$\mathbf{S}_B = \sum_{k=1}^{c} N_k (\mathbf{m}_k - \mathbf{m})(\mathbf{m}_k - \mathbf{m})^T, \;\; \mathbf{S}_W = \sum_{k=1}^{c} \sum_{\mathbf{x}_n \in \omega_k} (\mathbf{x}_n - \mathbf{m}_k)(\mathbf{x}_n - \mathbf{m}_k)^T$$

Ejercicio Utilizando la descomposición de Cholesky para $\mathbf{S}_W$, determinar $\mathbf{W}$ óptima en función de $\mathbf{S}_W$ y $\mathbf{S}_B$.

Cholesky: $\mathbf{W}^T \mathbf{S}_W \mathbf{W} = \mathbf{W}^T (\mathbf{S}_W^{\frac{1}{2}})(\mathbf{S}_W^{\frac{1}{2}})^T \mathbf{W}$

Cambio de coordenadas: $\mathbf{V} = (\mathbf{S}_W^{\frac{1}{2}})^T \mathbf{W}, \;\; \mathbf{W} = (\mathbf{S}_W^{-\frac{1}{2}})^T \mathbf{V}$

Discriminante de Fisher: caso $c > 2$

$$\mathbf{s}_W^{-1}\mathbf{s}_B = \underbrace{(\mathbf{V}^T\mathbf{V})^{-1}}_{\text{normalización}} \mathbf{V}^T \underbrace{\mathbf{S}_W^{-\frac{1}{2}}\mathbf{S}_B(\mathbf{S}_W^{-\frac{1}{2}})^T}_{M} \mathbf{V}$$

$\Rightarrow$ Para maximizar la traza, las cols de $\mathbf{V}$ deben ser los d' vectores propios dominantes de $\mathbf{M}$:

$$\mathbf{M}\mathbf{V} = \mathbf{V}\mathrm{Diag}(\lambda_1, \ldots, \lambda_{d'})$$
$$\mathbf{M}(\mathbf{S}_W^{\frac{1}{2}})^T\mathbf{W} = (\mathbf{S}_W^{\frac{1}{2}})^T\mathbf{W}\mathrm{Diag}(\lambda_1, \ldots, \lambda_{d'})$$
$$\Longrightarrow \quad \mathbf{S}_B\mathbf{W} = \mathbf{S}_W\mathbf{W}\mathrm{Diag}(\lambda_1, \ldots, \lambda_{d'})$$

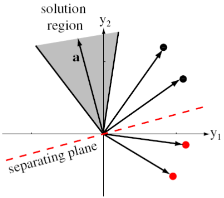
(Remplazando M)

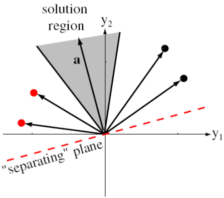
$\Rightarrow$ **Cols de $\mathbf{W}$: los d' autovectores dominantes de $\mathbf{S}_W^{-1}\mathbf{S}_B$**

Perceptrón de Rosenblatt y variaciones (caso $c = 2$)

Caso linealmente separable → Queremos clasificación sin error:

- $g(\mathbf{x}_n) = \mathbf{w}^T \phi(\mathbf{x}_n) + w_0 = \mathbf{a}^T \mathbf{y}_n \begin{cases} \geq 0 & \text{si } \mathbf{x}_n \text{ de la clase } \omega_1 \\ < 0 & \text{si } \mathbf{x}_n \text{ de la clase } \omega_2 \end{cases}$
- $t_n = \begin{cases} +1 & \text{si } \mathbf{x}_n \text{ de la clase } \omega_1 \\ -1 & \text{si } \mathbf{x}_n \text{ de la clase } \omega_2 \end{cases}$
- Región de solución $\mathcal{A}$: $\mathbf{a}$ tal que $t_n \mathbf{a}^T \mathbf{y}_n > 0$ para todo n





- Más robustez, margen (reduce $\mathcal{A}$): $t_n \mathbf{a}^T \mathbf{y}_n > b > 0$

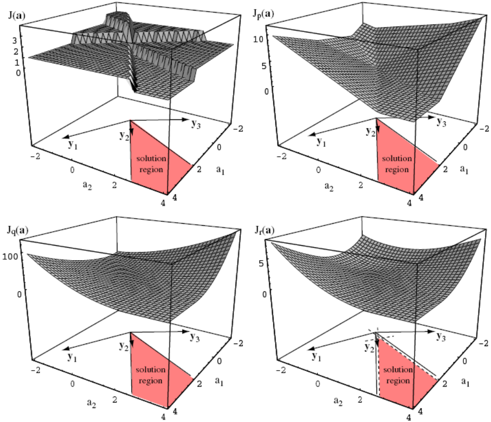
Perceptrón de Rosenblatt y variaciones (caso $c = 2$)

- $J(\mathbf{a})$: costo que se minimiza cuando $\mathbf{a}$ es una solución.
- $\mathcal{M}(\mathbf{a})$: indices de patrones mal clasificados para ese $\mathbf{a}$

Candidatos a criterios:

- Cantidad de patrones mal clasificados: $J(\mathbf{a}) = \#\mathcal{M}(\mathbf{a})$.
  Difícil de optimizar constante a trozos, no diferenciable).
- Perceptrón: $J_p(\mathbf{a}) = \sum_{n\in\mathcal{M}(\mathbf{a})}(-t_n\mathbf{a}^T\mathbf{y}_n)$.
  Lineal a trozos, continuo pero no C^1 (no es problemático en este caso)
- Relajación cuadrática: $J_q(\mathbf{a}) = \sum_{n\in\mathcal{M}(\mathbf{a})}(\mathbf{a}^T\mathbf{y}_n)^2$
  Es C^1. Dos problemas: Admite como solución $\mathbf{a} = 0$ y no trata igual a todas las muestras (residuo dominado por patrones con $\|\mathbf{y}_n\|$ grande)
- Relajación cuadrática con margen: $J_r(\mathbf{a}) = \frac{1}{2}\sum_{n\in\mathcal{M}(\mathbf{a})}\frac{(t_n\mathbf{a}^T\mathbf{y}_n - b)^2}{\|\mathbf{y}_n\|^2}$
  Evita los problemas de la relajación cuadrática simple.

Perceptrón de Rosenblatt y variaciones (caso $c = 2$)



Métodos de optimización (diferenciable)

Descenso por el gradiente

Nos movemos en la dirección de máxima pendiente

Algoritmo:

- Inicialización: $\mathbf{a}(0)$, θ, η
- $\mathbf{a}(k+1) = \mathbf{a}(k) - \eta(k)\nabla J(\mathbf{a}(\mathbf{k}))$
- hasta que $\eta(k)\|\nabla J(\mathbf{a}(\mathbf{k}))\| < \theta$

¿Cómo elegir $\eta(k)$?

- Demasiado pequeño: convergencia lenta
- Demasiado grande: oscilación o posible divergencia (si pb. no convexo)

Métodos de optimización (diferenciable)

Método de Newton

Se hace una aproximación de segundo orden con la matriz Hessiana.

$$J(\mathbf{a}) \approx J(\mathbf{a}(k)) + \nabla J^t(\mathbf{a} - \mathbf{a}(k)) + \frac{1}{2}(\mathbf{a} - \mathbf{a}(k))^t H(\mathbf{a} - \mathbf{a}(k))$$

Algoritmo:

- $\mathbf{a}(k+1) = \mathbf{a}(k) - 2H^{-1}\nabla J(\mathbf{a}(k))$
- hasta que $H^{-1}\nabla J(\mathbf{a}(k))$ sea menor a un umbral

Ventajas/desventajas:

- Converge en muchos menos pasos
- Calcular H^{-1} es muy costoso

Optimización del criterio Perceptrón

- Consideramos descenso por el gradiente
- Otra opción: programación lineal (e.g. método del Simplex)

El gradiente es entonces:

$$\nabla J_p = - \sum_{n \in \mathcal{M}(\mathbf{a})} t_n \mathbf{y}_n$$

El descenso es:

- Inicialización: $\mathbf{a}(0)$, θ, η
- $\mathbf{a}(k+1) = \mathbf{a}(k) + \eta(k) \sum_{n \in \mathcal{M}(\mathbf{a})} t_n \mathbf{y}_n$
- hasta que $\eta(k) \| \sum_{n \in \mathcal{M}(\mathbf{a})} t_n \mathbf{y}_n \| < \theta$

Perceptrón: corrección de a una muestra

- Se considera el algoritmo con un incremento fijo ($\eta(k) = \eta$ constante). Como el criterio no cambia si se escala $\mathbf{a}$, basta con tomar $\eta = 1$
- Se actualiza $\mathbf{a}$ con cada patrón mal clasificado

Algoritmo:

Inicializar $\mathbf{a}$, $k = 0$

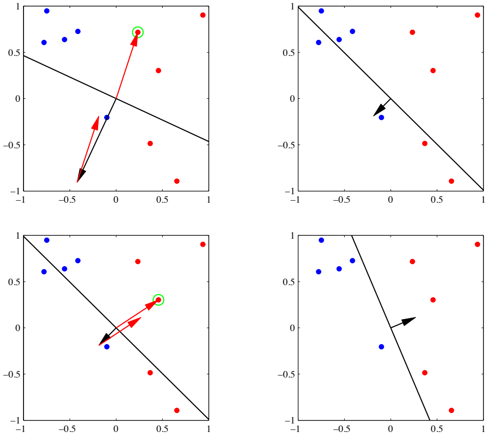
Repetir

- $k \leftarrow (k+1) \bmod N$
- Si $\mathbf{y}_k$ mal clasificado, entonces $\mathbf{a} \leftarrow \mathbf{a} + t_k \mathbf{y}_k$

hasta que todos los patrones estén bien clasificados.

Perceptrón: corrección de a una muestra

Ejemplo:



Perceptrón: convergencia

Teorema:

- Si los patrones son linealmente separables, el algoritmo converge a una solución exacta en una cantidad finita de pasos.
- Si los datos no son linealmente separables, el algoritmo no converge.

Prueba: ver e.g. Duda & Hart

Observaciones:

- La cantidad de pasos puede ser grande. Es imposible distinguir no separabilidad de convergencia lenta.
- Las relajaciones de mínimos cuadrados J_q y J_r siempre dan una solución, aún en el caso no separable.

Algoritmo de Ho-Kashyap para mínimos cuadrados

Modificación del mínimos cuadrados que:

- Si las muestras son linealmente separables, entrega una solución exacta en un número finito de pasos.
- Permite saber si el problema no es linealmente separable.

Idea: En el caso linealmente separable, existen $\hat{\mathbf{a}}$ y $\hat{\mathbf{b}}$ t.q. $\mathbf{Y}\hat{\mathbf{a}} = \hat{\mathbf{b}} > 0$. La idea es optimizar $\mathbf{Ya} - \mathbf{b}$ tanto en $\mathbf{a}$ y en $\mathbf{b}$, con la restricción que $\mathbf{b} > 0$.

Algorithm 11 (Ho-Kashyap)

<u>begin</u> <u>initialize</u> $\mathbf{a}, \mathbf{b}, \eta(\cdot) < 1$, criteria b_{min}, k_{max}
  <u>do</u> $k \leftarrow k + 1$
    $\mathbf{e} \leftarrow \mathbf{Ya} - \mathbf{b}$
    $\mathbf{e}^{+} \leftarrow 1/2(\mathbf{e} + \mathrm{Abs}[\mathbf{e}])$
    $\mathbf{b} \leftarrow \mathbf{a} + 2\eta(k)\mathbf{e}^{+}$
    $\mathbf{a} \leftarrow \mathbf{Y}^{\dagger}\mathbf{b}$
    <u>if</u> $\mathrm{Abs}[\mathbf{e}] \leq b_{min}$ <u>then</u> <u>return</u> $\mathbf{a}, \mathbf{b}$ *and* <u>exit</u>
  <u>until</u> $k = k_{max}$
  Print `NO SOLUTION FOUND`
<u>end</u>

Ejercicio Estudiar el algoritmo en Duda & Hart, Cap. 5, y entender qué hace exactamente el algoritmo.

Regresión logística: problema de dos clases

Comenzamos por el problema de dos clases.
Sean C_1, C_2, las clases a la que pertenecen las muestras $\mathbf{x}$.

Definimos las distribuciones a posteriori como:

$$p(C_1|\mathbf{x}) = y(\mathbf{x}) := \sigma(\mathbf{w}^T\mathbf{x}),$$
$$p(C_2|\mathbf{x}) = 1 - p(C_1|\mathbf{x}).$$

La función $\sigma(a) := \frac{1}{1+\exp(-a)}$ se conoce como *sigmoide*.
Verifica las siguientes propiedades:

- $\sigma(a)$ es creciente y su rango es $[0, 1)$. Puede ser interpretada como una probabilidad,
- $\sigma(-a) = 1 - \sigma(a)$,
- $a = \ln\left(\frac{\sigma}{1-\sigma}\right)$, llamada función *logit*, representa el log del cociente de las probabilidades a posteriori.

Este modelo probabilístico generativo se conoce como *regresión logística*, si bien es un modelo para clasificación más que para regresión.

Ventaja computacional de la regresión logística

Comparemos con otro modelo generativo clásico ya visto: mezcla de Gaussianas (MoG). Supongamos que el espacio de características es de dimensión M.

- El model de regresión logística involucra M parámetros (dimensión de $\mathbf{w}$).
- La clasificación por MoG con misma matriz de covarianza, ajustando las Gaussianas y luego decidiendo por ML involucra $2M$ parámetros para las medias, $M(M+1)/2$ para la matriz de covarianza y 1 para el prior $p(C_1)$. Total: $M(M+5)/2+1$ parámetros.

De aquí en más, nos dedicaremos a estudiar cómo determinar el vector de parámetros $\mathbf{w}$ para el modelo de regresión logística.

Determinación de los parámetros de la regresión logística

Disponemos de N muestras **independientes** etiquetadas, $\{\mathbf{x}_n, t_n\}$, $n = 1, 2, \ldots, N$, con $t_n \in \{0, 1\}$. Definimos $\mathbf{t} = (t_1, t_2, \ldots, t_N)^T$.

Cada muestra $\mathbf{x}_n$ sigue una distribución de Bernoulli de parámetro $y_n = p(C_1|\mathbf{x}_n)(= \sigma(\mathbf{w}^T\mathbf{x}_n))$, y debido a la independencia tenemos que la verosimilitud es

$$p(\mathbf{t}|\mathbf{w}) = \prod_{n=1}^{N} y_n^{t_n}(1 - y_n)^{1-t_n}.$$

Para obtener el $\mathbf{w}$ óptimo debemos maximizar esta expresión, o podemos minimizar el costo

$$\begin{aligned} E(\mathbf{w}) &= -\ln p(\mathbf{t}|\mathbf{w}) \\ &= -\sum_{n=1}^{N} t_n \ln y_n + (1 - t_n)\ln(1 - y_n). \end{aligned}$$

Determinación de los parámetros de la regresión logística (cont.)

Recordando que $y_n = \sigma(\mathbf{w}^T\mathbf{x}_n)$, y observando que $\frac{d\sigma}{da} = \sigma(1-\sigma)$, obtenemos operando:

$$\begin{aligned}\nabla E(\mathbf{w}) &= -\sum_{n=1}^{N} \frac{d}{d\mathbf{w}} \left\{ t_n \ln \sigma(\mathbf{w}^T\mathbf{x}_n) + (1-t_n)\ln(1-\sigma(\mathbf{w}^T\mathbf{x}_n)) \right\} \\ &= \ldots = \sum_{n=1}^{N} (\underbrace{\sigma(\mathbf{w}^T\mathbf{x}_n)}_{y_n} - t_n)\mathbf{x}_n.\end{aligned}$$

Obs. 1: el gradiente de $E(\mathbf{w})$ es la suma ponderada de las muestras $\mathbf{x}_n$ ponderadas por los “errores" correspondientes $(y_n - t_n)$. Cuanto menor es la magnitud de este error, menos influencia tiene la muestra en la dirección y la magnitud del gradiente.

Obs. 2: $\nabla E(\mathbf{w}) = 0$ no admite una solución analítica cerrada (debido a la no linealidad de la sigmoide). Sin embargo, veremos que es $E(\mathbf{w})$ es una función convexa y por lo tanto admite una solución única.

Solución única $\mathbf{w}$ al problema de regresión logística

Operando (usando $d\sigma/da = \sigma(1-\sigma)$) se obtiene que la matriz Hessiana $\mathbf{H} = \nabla\nabla E(\mathbf{w})$ vale

$$\mathbf{H} = \sum_{n=1}^{N} y_n(1-y_n)\mathbf{x}_n\mathbf{x}_n^T.$$

Como y_n y $1-y_n$ son positivos, H es definida positiva y por ende $E(\mathbf{w})$ es estrictamente convexa y admite un único mínimo global.

Podemos minimizar $E(\mathbf{w})$ mediante la iteración de Newton-Raphson:

$$\mathbf{w}^{(new)} = \mathbf{w}^{(old)} - \mathbf{H}^{-1}\nabla E(\mathbf{w}^{(old)}).$$

Solución única $\mathbf{w}$ al problema de regresión logística

Si llamamos

- $\mathbf{y} = (y_1, y_2, \ldots, y_N)^T$, $\mathbf{t} = (t_1, t_2, \ldots, t_N)^T$,
- $\mathbf{X} = [\mathbf{x}_1 | \mathbf{x}_2 | \ldots | \mathbf{x}_N]$, $\mathbf{W}(\mathbf{w}) = \mathrm{Diag}\left(\{y_n(1-y_n)\}_{n=1}^N\right)$,

tenemos

- $\nabla E(\mathbf{w}) = \mathbf{X}^T(\mathbf{y} - \mathbf{t})$, $\mathbf{H}(\mathbf{w}) = \mathbf{X}^T\mathbf{W}\mathbf{X}$, (Ambos dependen de $\mathbf{w}$).

$$\Rightarrow \mathbf{w}^{(new)} = \mathbf{w}^{(old)} - \left(\mathbf{X}^T\mathbf{W}^{(old)}\mathbf{X}\right)^{-1}\mathbf{X}^T(\mathbf{y}^{(old)} - \mathbf{t})$$

$$= \left(\mathbf{X}^T\mathbf{W}^{(old)}\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{W}^{(old)}\underbrace{\left(\mathbf{X}\mathbf{w}^{(old)} - \mathbf{W}^{(old)^{-1}}(\mathbf{y}^{(old)} - \mathbf{t})\right)}_{\mathbf{z}^{(old)}}$$

$$\boxed{\mathbf{w}^{(new)} = \left(\mathbf{X}^T\mathbf{W}^{(old)}\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{W}^{(old)}\mathbf{z}^{(old)}}$$

Iterative Reweighted Least Squares (IRLS)

Regresión logística multiclase

- K clases C_k, N muestras $\{\mathbf{x}_n, \mathbf{t}_n\}$, con $\mathbf{t}_n = (t_{n1}, t_{n2}, \dots, t_{nK})$, con $t_{nk} = 1$ si $\mathbf{x}_n \in C_k$, 0 sino.
- $p(C_k|\mathbf{x}_n) = y_{nk} := \sigma(\mathbf{w}_k^T \mathbf{x}_n)$.
- $\sigma(\mathbf{w}_k^T \mathbf{x}) := \frac{\exp(\mathbf{w}_k^T \mathbf{x})}{\sum_{j=1}^K \exp(\mathbf{w}_j^T \mathbf{x})}$ función *softmax*. Si $K = 2$ es la sigmoide.

Verosimilitud: $p(\mathbf{t}_1, \dots, t_N | \mathbf{w}_1, \dots, \mathbf{w}_K) = \prod_{n=1}^N \prod_{k=1}^K (y_{nk})^{t_{nk}}$.

$(-1)\times$log-verosimilitud: $E(\mathbf{w}_1, \dots, \mathbf{w}_K) = -\sum_{n=1}^N \sum_{k=1}^K t_{nk} \ln y_{nk}$.

Usando que $\frac{\partial y_{nk}}{\partial \mathbf{w}_j} = y_{nk}(\delta_{kj} - y_{nj})\mathbf{x}_n$[1]:

$$\nabla_{\mathbf{w}_j} E(\mathbf{w}_1, \dots, \mathbf{w}_K) = \sum_{n=1}^N (y_{nj} - t_{nj})\mathbf{x}_n,$$

$$\nabla_{\mathbf{w}_k} \nabla_{\mathbf{w}_j} E(\mathbf{w}_1, \dots, \mathbf{w}_K) = -\sum_{n=1}^N y_{nk}(\delta_{kj} - y_{nj})\mathbf{x}_n \mathbf{x}_n^T.$$

Los $\mathbf{w}_k$ se calculan mediante IRLS.

[1] δ_{kj} es la delta de Kronecker