TEMA 6: SUPPORT VECTOR MACHINES

ANÁLISIS MULTIVARIANTE (Grado en Estadística)

Departamento de Estadística e I.O. · UVa

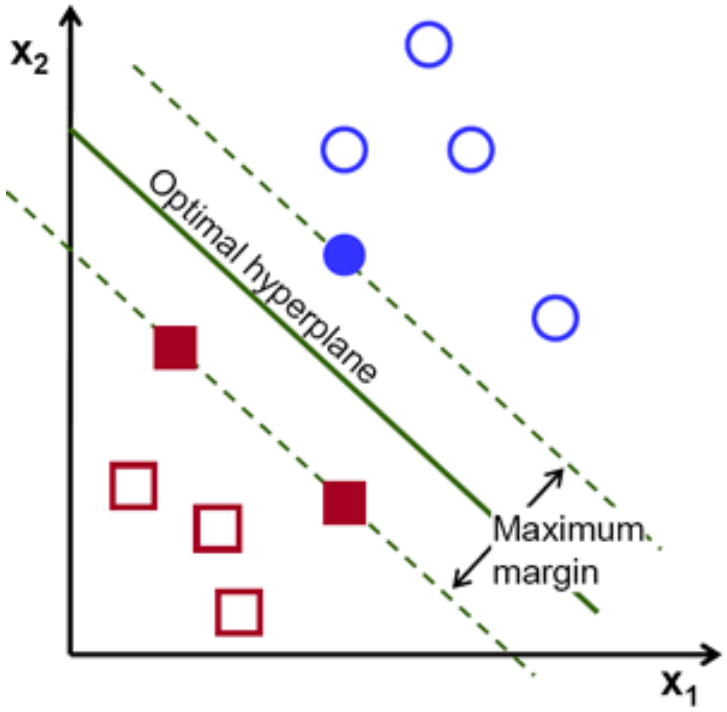
Separación Lineal

- **Datos "separables linealmente":** Suponer que estamos con n datos $x_i \in \mathbb{R}^p$, tenemos dos clases $y_i \in \{-1, 1\}$ y que existe un hiperplano $\{x : \beta_0 + \beta' x = 0\}$ que los **"separa"** perfectamente:
  - $\diamond$ $\boxed{y_i = 1}$ si $\boxed{\beta_0 + \beta' x_i > 0}$
  - $\diamond$ $\boxed{y_i = -1}$ si $\boxed{\beta_0 + \beta' x_i < 0}$

  (es decir, $(\beta_0 + \beta' x_i) y_i > 0$ para $i = 1, ..., n$).

- El LDA y la regresión logística buscan hiperplanos separantes lineales pero se puede ver que no siempre encuentran un hiperplano que separa perfectamente los grupos aunque éstos sí sean "linealmente separables".

- **Objetivo:** Si los datos son separables linealmente vamos a buscar un hiperplano que mejor los "separe" en el sentido de un mayor "margen"...



- La distancia del punto x a $\{x : \beta_0 + \beta' x = 0\}$ es $\pm \frac{1}{\|\beta\|}(\beta_0 + \beta' x)$

- Si tomamos $\|\beta\| = 1$ y los datos son separables linealmente entonces la "distancia del punto x_i al hiperplano separante" será $y_i(\beta_0 + \beta' x_i)$ (en transparencia anterior vimos que esa distancia era siempre positiva).

- Usaremos una regla $G(x) = \text{sign}\{\beta_0 + \beta' x\} \in \{-1, 1\}$ para β_0 y β que separen con mayor "margen" los grupos.

- **Mayor margen** M resolviendo el problema:

$$\max_{\beta_0,\beta \text{ con } \|\beta\|=1} M \text{ sujeto a } y_i(\beta_0 + \beta' x_i) \geq M$$

(distancia de todos los x_i al hiperplano mayores que el margén M...)

- En lugar de tomar $\|\beta\| = 1$ podemos tomar un β tal que la restricción $y_i(\beta_0 + \beta' x_i) \geq M$ simplifique a $y_i(\beta_0 + \beta' x_i) \geq 1$. En ese caso, se puede ver que la distancia de las observaciones en los "margenes" es $M = 1/\|\beta\|$. Luego maximizar $M(= 1/\|\beta\|)$ es equivalente minimizar $\|\beta\|$ y el problema puede reescribirse como

$$\min_{\beta_0,\beta} \frac{1}{2}\|\beta\|^2 \text{ sujeto a } y_i(\beta_0 + \beta' x_i) \geq 1$$

Karush-Kuhn-Tucker (extensión multiplicadores de Lagrange para restricciones con "$\geq$"...)

$$L(\beta_0, \beta, \alpha_1, ..., \alpha_n) = \frac{1}{2}\|\beta\|^2 - \sum_{i=1}^{n} \alpha_i(y_i(\beta_0 + \beta' x_i) - 1).$$

Derivando e igualando a 0 vemos que $0 = \sum_{i=1}^n \alpha_i y_i$ y $\beta = \sum_{i=1}^n \alpha_i y_i x_i$. Las condiciones Karush-Kuhn-Tucker implican $\alpha_i(y_i(\beta_0 + \beta' x_i) - 1) = 0$,...

- Se puede probar que:
  - $\beta = \sum_{i=1}^{n} \alpha_i y_i x_i$
  - $\alpha_i \geq 0$
  - $\alpha_i > 0$ solo para los puntos verificando $y_i(\beta_0 + \beta' x_i) = 1 \Rightarrow$ Las únicas observaciones que contribuyen a calcular β son aquellas que están exactamente en los "margenes"
    $\Rightarrow$ **Puntos soporte** (**Support** Vector Machines)

- Nótese que los Support Vector Machines (SVM) ponen todo su esfuerzo clasificatorio centrandose en las observaciones "fronterizas" entre grupos ($\approx$ puntos soporte). El LDA, QDA y logística usan *todas* las observaciones en las clases aunque éstas fueran fácilmente separables...

El problema se reescribe de nuevo (...) en

$$\max_{\alpha_1,\ldots,\alpha_n} \left\{ \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{k=1}^{n} \alpha_i \alpha_k y_i y_k x_i' x_k \right\} \text{ sujeto a } \alpha_i \geq 0$$

y se debe verificar $\alpha_i(y_i(\beta_0 + \beta' x_i) - 1) = 0$ para $i = 1, \ldots, n$.

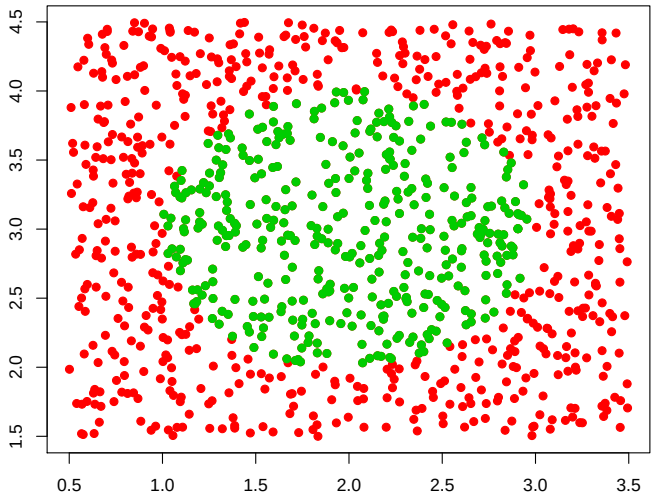
- **Grupos no separables linealmente:** Para grupos no separables podemos:

  1) Permitir observaciones mal-clasificadas e introducir variables de "holgura" ξ_i's (slack variables)

  2) Llevar los datos a dimensiones mayores ($x_i \mapsto \phi(x_i)$) donde los grupos estén separados.

  3) Combinar **1)** y **2)**

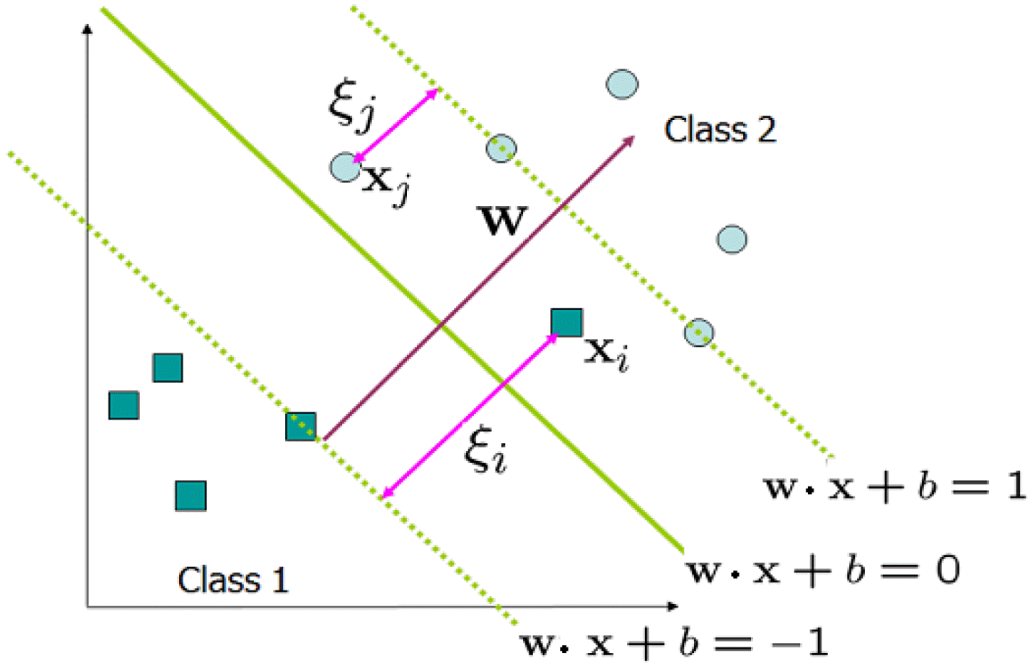
- **Ejemplo de 2):** G1 = "dentro bola $B((2,3)', 1)$" y G2 = "fuera bola"



G1 y G2 no separables linealmente con $\{x_1, x_2\}$ y sí con $\{x_1, x_2, x_1^2, x_2^2\}$

- **Admitir "mal-clasificados":** Permitimos observaciones x_i en lados no verificando los "margenes" y penalizando el tamaño de esa "mala-clasificación" con un "coste" dado por C:

$$\min_{\beta_0,\beta} \frac{1}{2}\|\beta\|^2 + C\sum_{i=1}^{n}\xi_i \text{ sujeto a } \xi_i \geq 0 \text{ y } y_i(\beta_0 + \beta' x_i) \geq 1 - \xi_i$$



Computacionalmente (...) se añade la restricción $0 \leq \alpha_i \leq C$

- **Trabajar en dimensiones mayores y uso de nucleos (kernels):** Llevar los datos a un espacio de dimensión mayor

  $x_i \in \mathbb{R}^p$ (data space) $\mapsto \Phi(x_i) = (\Phi_1(x_i), ..., \Phi_r(x_i))' \in \mathbb{R}^r$ (feature space)

  con $r \geq p$ donde sí son separables.

- Realizamos un SVM basado en $\{(\Phi(x_i), y_i)\}_{i=1}^n$. Dado un nuevo $x \in \mathbb{R}^p$

$$G(x) \hookleftarrow G_{\text{aux}}(\Phi(x)) = \text{sign}\{\beta_0 + \beta'\Phi(x)\}$$

y tendremos un $\beta = \sum_{i=1}^n \alpha_i y_i \Phi(x_i)$ (ahora con $\beta \in \mathbb{R}^r$) tal que:

$$G_{\text{aux}}(x) = \text{sign}\left\{\beta_0 + \sum_{i=1}^n \alpha_i y_i \Phi(x_i)'\Phi(x)\right\} =$$

$$= \text{sign}\left\{\beta_0 + \sum_{i=1}^n \alpha_i y_i \langle \Phi(x_i), \Phi(x)\rangle\right\} \in \{-1, 1\}$$

Recordar que $u'v = \langle u, v\rangle$

- **Kernels:** Para simplificar el coste computacional y evitar tener que fijar explicitamente las funciones $\Phi_1(\cdot)$,..., $\Phi_r(\cdot)$ se usan **nucleos**:

$$K(x, x') = \langle \Phi(x), \Phi(x') \rangle.$$

Así, la regla anterior se convierte en:

$$G(x) = \text{sign} \left\{ \beta_0 + \sum_{i=1}^{n} \alpha_i y_i K(x_i, x) \right\}$$

- **Kernels más usados:**
  - ◇ *Polinómico:* $K(x, x') = (1 + \langle x, x' \rangle)^d$
  - ◇ *Bases radiales:* $K(x, x') = \exp(-\gamma \|x - x'\|^2)$

Nótese que la función objetivo también se expresa en terminos de evaluaciones del nucleo K ya que la nueva función objetivo $\sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{k=1}^{n} \alpha_i \alpha_k y_i y_k \Phi(x_i)' \Phi(x_k)$ se escribe como $\sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{k=1}^{n} \alpha_i \alpha_k y_i y_k K(x_i, x_k)$...

- **Ejemplo:** Kernel polinómico con $d = 2$:

$$K(x, x') = (1 + \langle x, x' \rangle)^2 = (1 + x_1 x_1' + x_2 x_2')^2$$

$$= 1 + 2x_1 x_1' + 2x_2 x_2' + (x_1 x_1')^2 + (x_2 x_2')^2 + 2x_1 x_1' x_2 x_2'.$$

  Luego para $x = (x_1, x_2)'$ podemos tomar

$$\Phi(x) = (\Phi_1(x), \Phi_2(x), ..., \Phi_6(x))' = (1, \sqrt{2}x_1, \sqrt{2}x_2, x_1^2, x_2^2, \sqrt{2}x_1 x_2)'$$

  con lo que $K(x, x') = \langle \Phi(x), \Phi(x') \rangle$ y la dimensión del "feature space" es $r = 6 (> p = 2)$.

- Los parametros C (coste mal-clasificación) y d (nucleo polinómico) o γ (bases radiales) se eligen por **validación cruzada** (elegir (C, d) o (C, γ) minimizando ese error):

  ◇ C mayor lleva a mayor sobreajuste
  ◇ d mayor lleva a mayor sobreajuste
  ◇ γ mayor lleva a mayor sobreajuste

- La librería `e1071` en `R` ajusta SVM y realiza esa validación cruzada
- No es invariante frente a transformaciones de escala por lo que hay que escalar los datos
- El problema de $q > 2$ clases se aborda, por ejemplo, realizando $q(q-1)/2$ problemas SVM de dos clases y realizando una votación...
- Existen nucleos $K(\cdot, \cdot)$ especialmente adaptados a problemas muy diferentes como, por ejemplo, en "text mininig"...