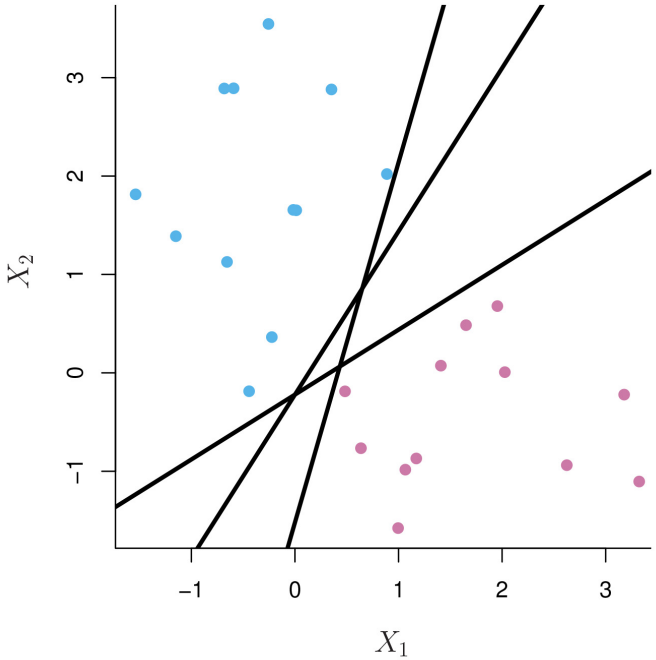
Support Vector Machines

Walter Sosa-Escudero

Universisad de San Andres y CONICET

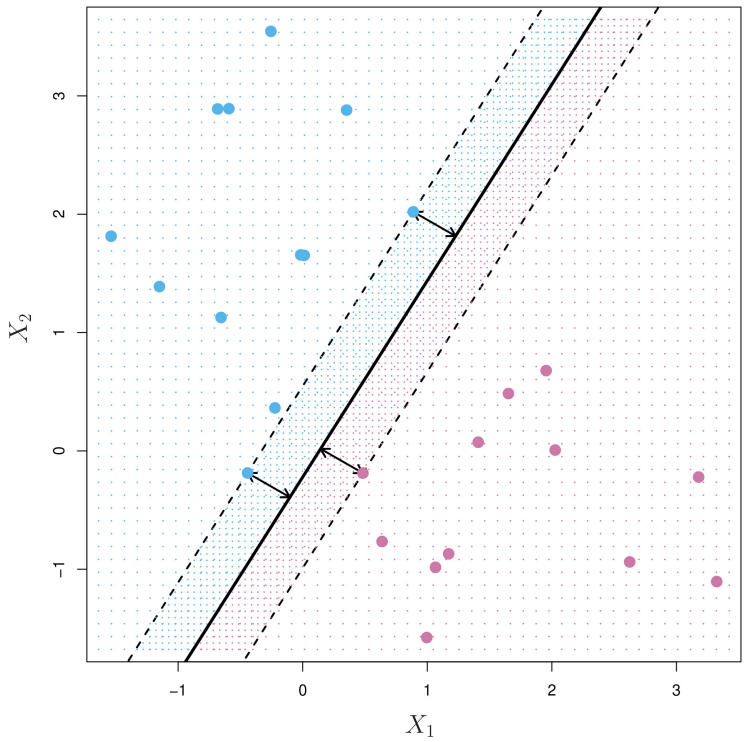
Fuente y graficos: An Introduction to Statistical Learning (Gareth, Witten, Hastie y Tibshirani, 2016)

Margin classifier: caso separable



- Azul = 1, rojo = -1
- Hiperplano separador
- Separable?
- Unicidad

Maximal margin hyperplane



- Margen
- Maximal margin hyperplane
- Vectores soporte
- Ventajas del 'margen'?
- Entrenamiento/test.

Hiperplano

En un espacio de dimension $p+1$, hiperplano:

$$\beta' x + \beta_0 = 0$$

$$\beta' = (\beta_1, \beta_2, \ldots, \beta_p)$$

Ej: $p = 2$, recta que no necesariamente pasa por el origen. $x_1\beta_1 + x_2\beta_2 + \beta_0 = 0$.

Datos: (y_i, x_i), $y_i \in \{-1, 1\}$, $x_i \in \Re^p$

Bajo separacion perfecta, hiperplano separador:

$$y_i \left(\beta' x_i + \beta_0\right) > 0, \qquad \forall\, i = 1, \ldots, n$$

- Problema: bajo separabilidad, en general no existe un unico hiperplano separador.
- Margen: minima distancia de cada punto x_i al hiperplano.
- Maximal margin hyperplane: hiperplano separador mas *lejano* a todos los puntos.

Distancia al hiperplano

Hiperplano: $\beta' x + \beta_0 = 0$

Normalizacion: $||\beta|| = \sum_{j=1}^{p} \beta_j^2 = 1$

Minima distancia de x_0 al hiperplano (minima distancia al cuadrado):

$$\min_{x} \ (x_0 - x)'(x_0 - x) \ ; \quad \text{sa} \quad \beta' x + \beta_0 = 0$$

Intuicion? x mas cercano a x_0, x en el hiperplano.

Lagrange: $L(x, \lambda) = (x_0 - x)'(x_0 - x) - \lambda \left(\beta' x + \beta_0\right)$

FOC (con respecto a x): $2(x_0 - x) - \lambda\beta = 0$

$$2(x_0 - x) - \lambda\beta = 0$$

Premultiplicando por β' (con $\beta'\beta = 1$)

$$2\beta'(x_0 - x) = \lambda$$

Premultiplicando por $(x_0 - x)'$

$$2(x_0 - x)'(x_0 - x) = \lambda(x - x_0)'\beta$$

Reemplazando λ:

$$2(x_0 - x)'(x_0 - x) = \left(2\beta'(x_0 - x)\right)\left((x_0 - x)'\beta\right)$$

$$2(x_0 - x)'(x_0 - x) = \left(2\beta'(x_0 - x)\right)\left((x_0 - x)'\beta\right)$$

Entonces, la distancia minima (al cuadrado) al hiperplano es

$$(x_0 - x)'(x_0 - x) = \left(\beta'(x_0 - x)\right)^2$$

No hace falta saber x solo la distancia!

Minima distancia (raiz cuadrada):

$$\beta'(x_0 - x) = \beta' x_0 - \beta' x$$

Como $\beta' x + \beta_0 = 0$, minima distancia:

$$\beta' x_0 + \beta_0$$

Maximal margin classifier (MMC)

La solucion a este problema produce el MMC:

$$\max_{\beta,\beta_0} M, \text{ sa:}$$

1. $||\beta|| = 1$
2. $y_i \left(\beta' x_i + \beta_0\right) \geq M, \qquad i = 1, \ldots, n, \quad M > 0$

Clasificador $\hat{y}_i = \text{sgn}\left(\beta' x_i + \beta_0\right)$

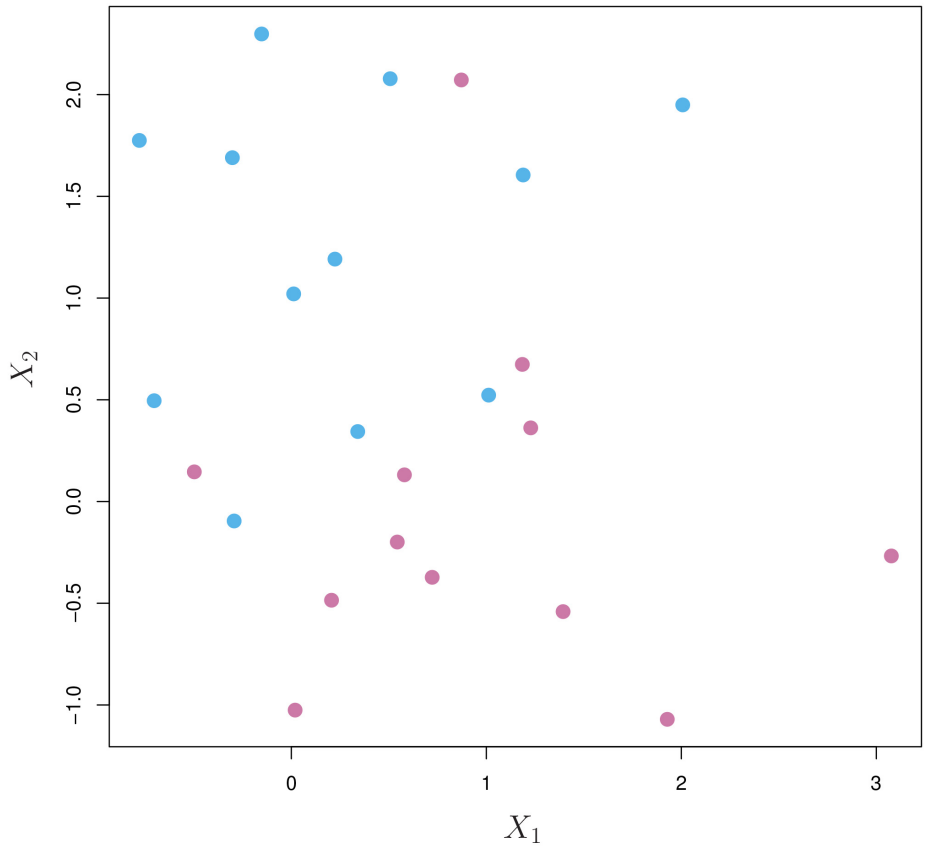
Condicion 2):

- Signo?
- Nivel?

- Clasificacion: sgn $(\beta' x_i + \beta_0)$.
- Support vector: puntos que satisfacen 2) como igualdad. Puntos que estan *sobre* hiperplanos separadores.
- Importante: la solucion depende *solamente* de los vectores soporte.

Support Vector Classifier: caso no separable

Y ahora?



No existe ningun hiperplano separador.

Support vector classifier

$$\max_{\beta_0,\beta,\epsilon} M, \qquad \text{sa:}$$

1. $||\beta|| = 1$
2. $y_i \left(\beta' x_i + \beta_0\right) \geq M(1 - \epsilon_i)$
3. $\epsilon_i \geq 0$
4. $\sum_{i=1}^{n} \epsilon_i \leq C, \quad C \geq 0$

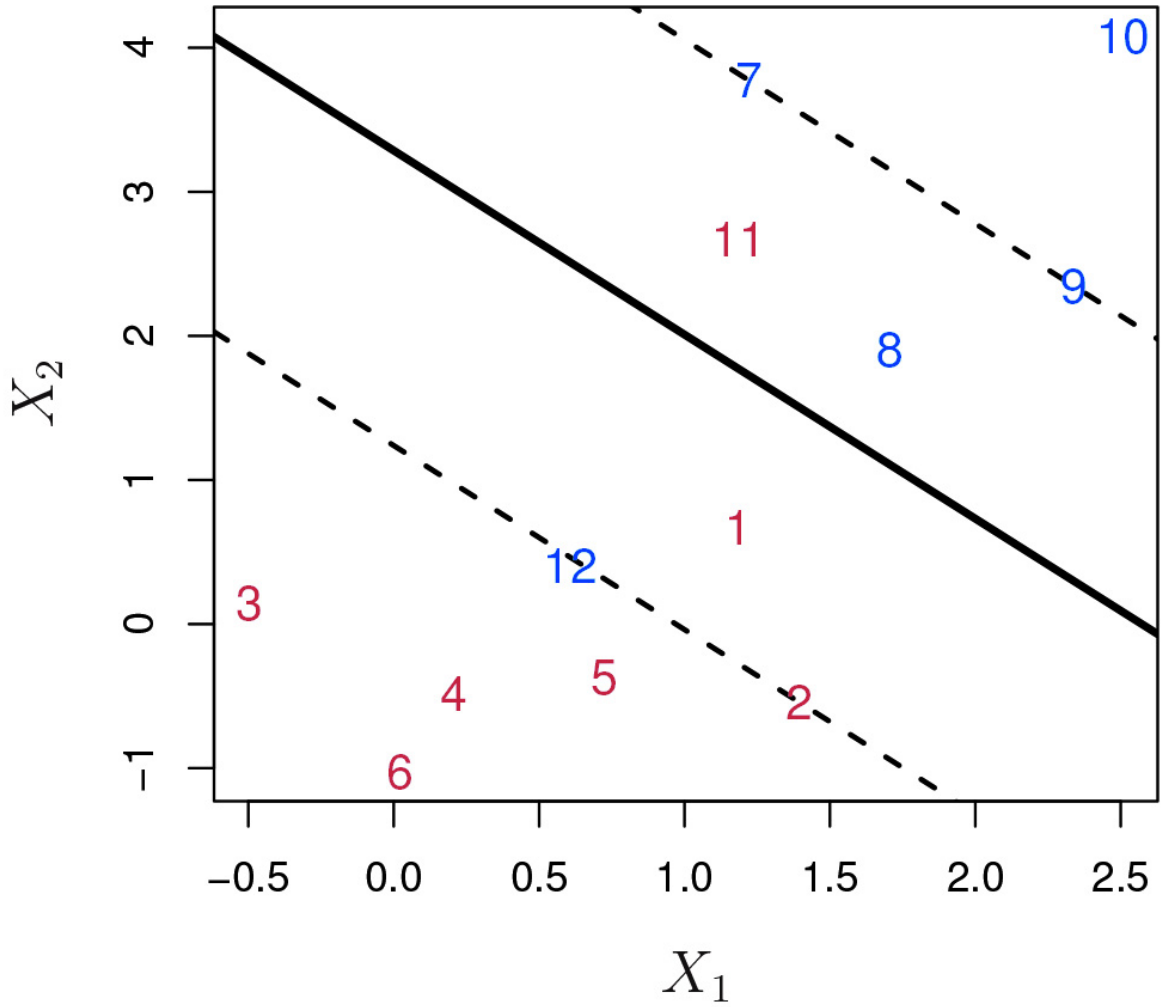
$C =$ 'costo'.

Intuiciones

$$2)\ y_i\ (\beta' x_i + \beta_0) \geq M(1-\epsilon_i), \qquad 3)\ \epsilon_i \geq 0, \qquad 4)\ \sum_{i=1}^{n} \epsilon_i \leq C$$

- Clasificador: sgn $(\beta' x_i + \beta_0)$
- $\epsilon_i = 0$. Caso anterior. No existe hiperplano separador.
- $0 < \epsilon_i \leq 1$: *dentro* del margen pero bien clasificada.
- $\epsilon > 1$: mal clasificada.

Idea: Caso separable: clasificacion correcta y ninguna observacion dentro del margen. No separable: 'Sacrificar' observaciones para seguir clasificando correctamente el resto.



ϵ_i para cada caso?

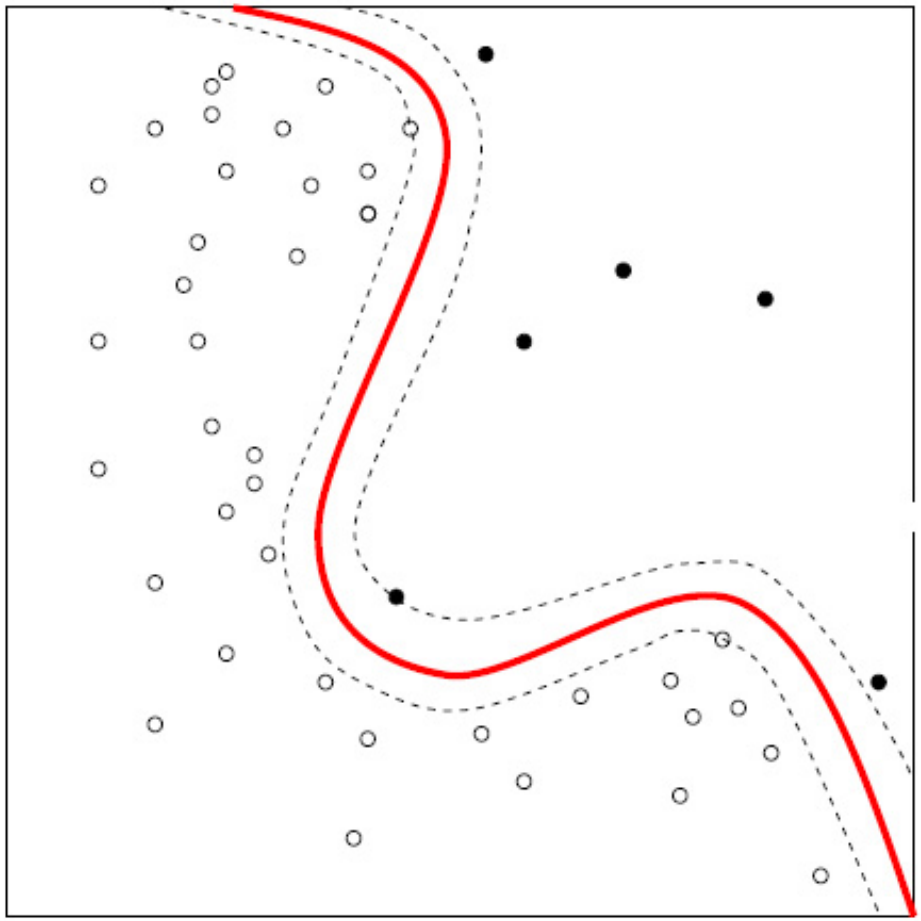
Support vector property

- Hiperplano solucion determinado exclusivamente por las observaciones que caen sobre o en el lado incorrecto del margen.
- Observaciones con $\epsilon_i \neq 0$
- Formalmente: Kuhn-Tucker

C, hiperparametro elegido por cross validation. C chico?. C muy grande? Trade off.

Ejercicio: ver discusion pp. 346-349 en ISL(2016)

Support vector machine



Dualidad

$$\min_{x \in \Re^p} f(x) \text{ sa } h(x) \leq 0$$

$$L(x, \lambda) = f(x) + \lambda \, h(x)$$

Alternativa:

1. Construir $L^D(\lambda) = \inf_x L(x, \lambda)$
2. $\max_\lambda L^D(\lambda)$ sa $\lambda \geq 0$

Bajo algunas condiciones, la solucion a ambos problemas deberia coincidir (strong duality). Al primer problema se lo llama primal y al segundo, dual. $L^D(\lambda) =$ funcion dual de Lagrange.

Ideas

$$\min_{x \in \Re^p} f(x) \text{ sa } h(x) \leq 0$$

$$L(x, \lambda) = f(x) + \lambda \, h(x)$$

$$L^D(\lambda) = \inf_x L(x, \lambda)$$

Si p^* es el valor minimo del primal:

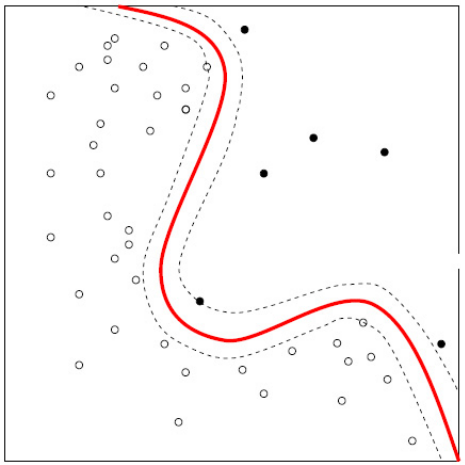
$$L^D(\lambda) \leq p^* \qquad \text{Por que?}$$

- Dual: maxima cota inferior.
- Intuicion: minimizacion de costo sujeto a presupuesto. Maximizar el precio sombra de la restriccion presupuestaria.

Ideas

- Muchas veces el dual es mas tratable o intuitivo que el primal.
- Strong duality: ambos problemas conducen a la misma solucion. Convexidad y 'cualificacion de restricciones' (Slater).
- Desigualdades via Kuhn-Tucker.
- Detalles: Boyd y Vandenberghe (2004, Convex Optimization)

De vuelta a SVM...



- Problema: separacion no lineal
- Solucion?: Agregar terminos no lineales (polinomio, interacciones)
- Solucion SVM: agregarlas en el *dual*.
- Por que?: computacionalmente mucho mas eficiente.

El dual de MMC

$$\max_{\beta,\beta_0} M, \text{ sa: } ||\beta|| = 1, \quad y_i \left(\beta' x_i + \beta_0\right) \geq M, \quad M > 0$$

Truco: $||\beta|| = 1/M$. El siguiente problema es equivalente

$$\min_{\beta,\beta_0} \frac{1}{2}||\beta||^2, \text{ sa: } y_i \left(\beta' x_i + \beta_0\right) \geq 1$$

Idea: armaremos el primal y el dual

Primal:

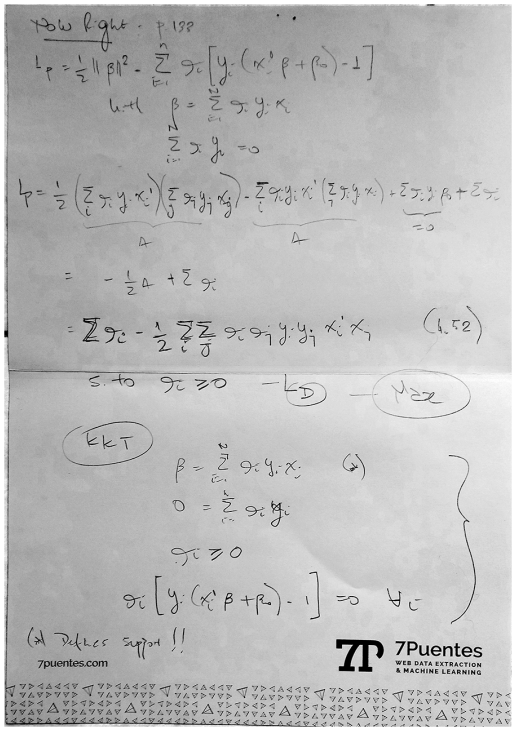
$$L(\beta, \beta_0, \lambda) = \frac{1}{2}||\beta||^2 - \sum_{i=1}^{n} \lambda_i \left[y_i(x_i'\beta + \beta_0) - 1 \right]$$

Dual: minimizar con respecto a β, β_0, reemplazar en L, luego maximizar con respecto a λ. FOC con respecto a β, β_0

$$\begin{aligned} \beta &= \sum_{i=1}^{n} \lambda_i y_i x_i \\ 0 &= \sum_{i=1}^{n} \lambda_i y_i \end{aligned}$$

Reemplazando en L obtenemos L^D

$$L^D(\lambda) = \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j \; y_i y_j \; x_i' x_j$$



- Dual: desaparecieron los β, β_0
- Termino clave: $x_i' x_j$.
- El problema depende de x solo a traves del producto interno.
- Kernel trick: 'no-linealizar' el producto interno.

Kernel trick

$$L^D(\lambda) = \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j \; y_i y_j \; K(x_i, x_j)$$

$K(x_i, x_j)$: 'kernel'. Funcion simetrica y no negativa.
Generalizacion del producto interno.

- Lineal: $K(x_i, x_j) = x_i' x_j$
- Cuadratico: $K(x_i, x_j) = (1 + x_i' x_j)^2$
- Radial: $K(x_i, x_j) = \exp(-\gamma ||x_i - x_j||^2)$
- Red neuronal: $K(x_i, x_j) = \tanh(\gamma_1 \; x_i' x_j + \gamma_2)$

Ejemplo: Kernel cuadratico

$$K(x_i, x_j) = (1 + x_i' x_j)^2$$

Dos predictores, $x_i' = (z_i \;\; r_i)$

$$\begin{aligned} K(x_i.x_j) &= (1 + x_i' x_j)^2 \\ &= (1 + z_i z_j + r_i r_j)^2 \\ &= 1 + 2z_i z_j + 2r_i r_j + (z_i z_j)^2 + (r_i r_j)^2 + 2r_i r_j z_i z_j \\ &= h_i' h_j \end{aligned}$$

con $h_s' \equiv \left(1, \sqrt{2}\, z_s, \sqrt{2}\, r_s, \; z_s^2, \; r_s^2, \; \sqrt{2}\, z_s r_s\right)$

Idea: es como si hubiesemos introducido los niveles, los cuadrados y la interaccion en el problema general!

Ideas y extensiones

- Support vector machine: misma logica, aplicada sobre el dual del problema de optimizacion de soft margin.
- Ganancia computacional: 'pocas cuentas' calculadas en los vectores soporte.
- Kernel trick: no-linealidades en la generalizacion del producto interno, no en las x originales.
- Kernel como un producto interno de vectores de funciones no lineales de los predictores originales?: Mercer kernel. Idea: 'raiz cuadrada' de una matriz simetrica pd (matriz de Gram).
- Kernel: idea mucho mas generica, capaz de generar un espacio de funciones (Kernel reproducing Hilbert space).

Referencias

- Hastie, Tibshirani, Friedman, *The Elements of statistical Learning*.
- Murphy, *Machine learning*

Ver citas en el programa de la materia.