Aux 9. Introducción a la Minería de Datos

Gastón L'Huillier[1,2], Richard Weber[2]
glhuilli@dcc.uchile.cl

[1]Departamento de Ciencias de la Computación
Universidad de Chile

[2]Departamento de Ingeniería Industrial
Universidad de Chile

2010

Riesgo Empírico vs Riesgo Esperado

Al aproximar una función, se pueden cometer dos tipos de errores:

Riesto Empírico

Error asociado a la base de datos (o la muestra) que observó.

$$R_{[emp]} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{2} |f(\mathbf{x}_i) - \hat{y}_i| \tag{1}$$

Riesgo Esperado

Error asociado al espacio total estudiado.

$$R = \int_{\mathcal{X}} \frac{1}{2} |f(\mathbf{x}_i) - \hat{y}_i| dP(\mathbf{x}, y) \tag{2}$$

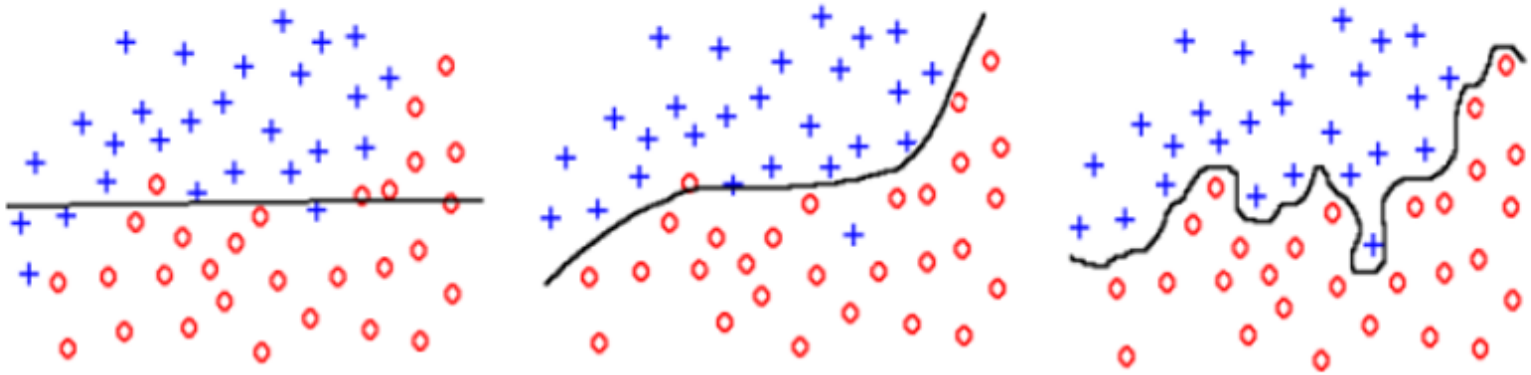
Riesgo Empírico vs Riesgo Esperado [2]

R.Empirico v/s R.Esperado

- Existe un comportamiento asintótico entre ambos riesgos. Sólo en el límite cuando la cantidad de observaciones es infinita, el riesgo empírico es igual al riesgo esperado.
- En general no existe garantía que se pueda encontrar una función que minimice el riesgo esperado utilizando el riesgo empírico, ya que son expresiones completamente distintas.
- Este resultado se puede ver en el hecho que el conjunto de entrenamiento tiene menor error que el de prueba.

Riesgo Empírico - Minimización

Si sólo minimizamos el riesgo empírico, nos podemos enfrentar con el siguiente problema:



Sub-ajuste | Ajuste | Sobre-ajuste

Riesgo Estructural - Minimización

¿Minimizar sólo el riesgo empírico?

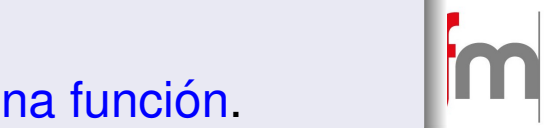
Para toda función f existe una aproximación f' tal que acierte a todo el conjunto de entrenamiento y falle en TODOS los demás puntos.
(*sobre-ajuste*)
$\Rightarrow$ No sirve minimizar solamente el riesgo empírico.

Teorema [Vapnik and Chervonenkis, 1971]

- Minimización de riesgo estructural esta determinado por la **capacidad de una función** y el **riesgo empírico**.
- Si $h < N$, es la dimensión VC de todas las funciones que se puedan estimar, entonces para todas las funciones de esta clase, con una probabilidad de al menos $1 - \eta$, se cumple:

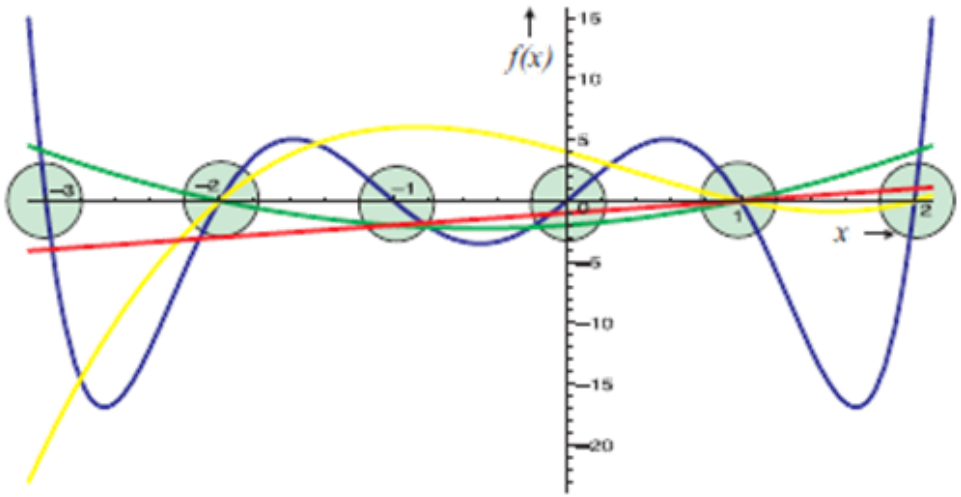
$$R(f) \leq R_{[emp]}(f) + \sqrt{\frac{h(log\frac{2m}{h} + 1) - log(\eta/4)}{N}} \tag{3}$$

- La dimensión VC (h) permite medir la *capacidad* de una función.



Dimensión Vapnik-Chervonenkis

- La capacidad de una función indica cuán compleja puede ser. (e.g. una función lineal tiene baja capacidad.)
- Minimización del riesgo estructural favorece estructuras simples. (navaja de Occam)



$$
\begin{aligned}
f_1(x) &= (x-1) \\
f_2(x) &= (x-1)(x+2) \\
f_3(x) &= (x-2)(x-1)(x+2) \\
f_6(x) &= (x-2)(x-1)\,x\,(x+1) \\
&\quad\ (x+2)(x+3)
\end{aligned}
$$

Riesgo Estructural - Navaja de Occam

William of Occam (1285 – 1349)

"Pluritas non est ponendas sin necesitade"
$\Rightarrow$ "Cantidad no debería ser agregada sin necesidad"

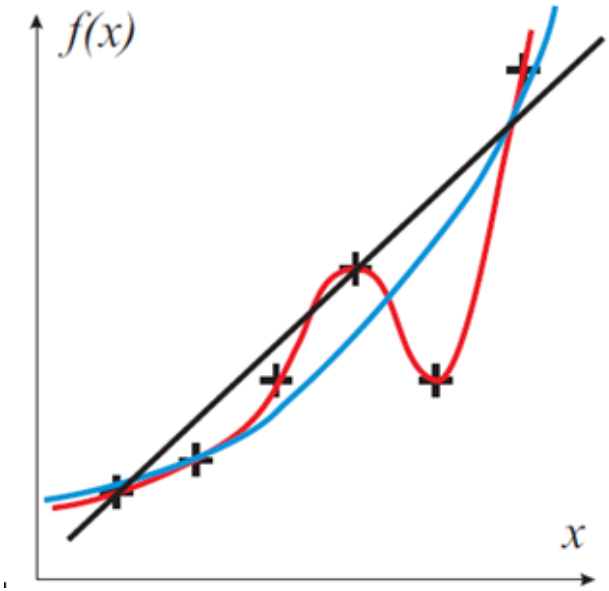
- Si dos teorías explican los hecho igual de bien, entonces la más simple debería ser considerada.
- Existe una menor cantidad de hipótesis simples que hipótesis complejas:
  - La coincidencia que una hipótesis simple se ajuste a los datos es baja.
  - La coincidencia que una hipótesis compleja se ajuste a los datos es alta.

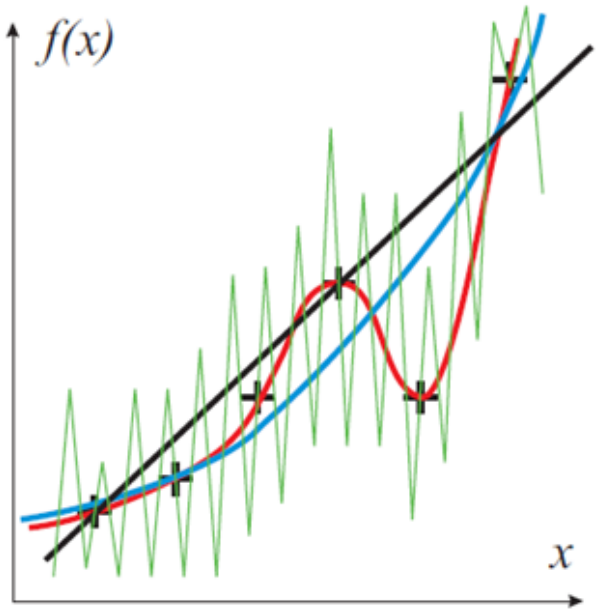
Riesgo Estructural - Navaja de Occam

William of Occam (1285 – 1349)

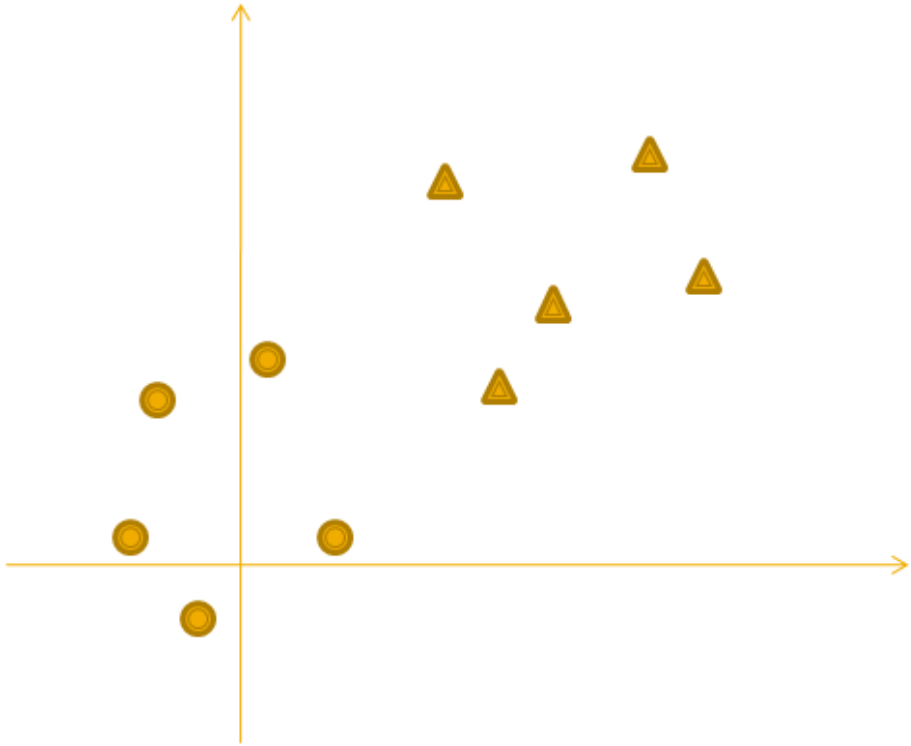
"Pluritas non est ponendas sin necesitade"

$\Rightarrow$ "Cantidad no debería ser agregada sin necesidad"

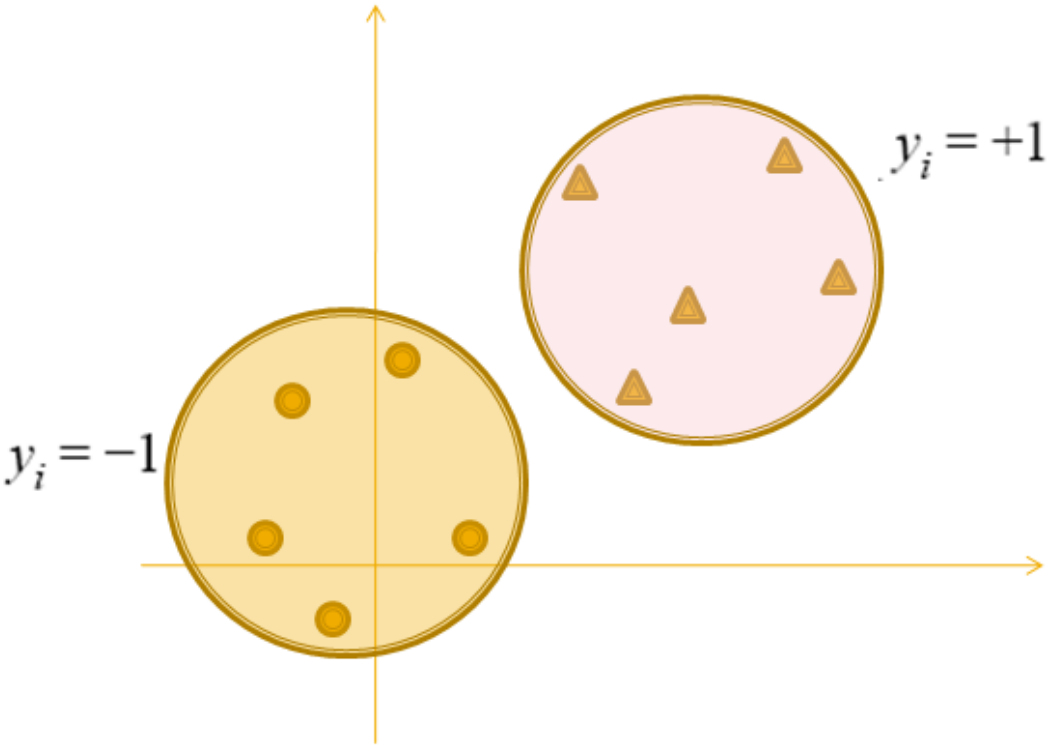




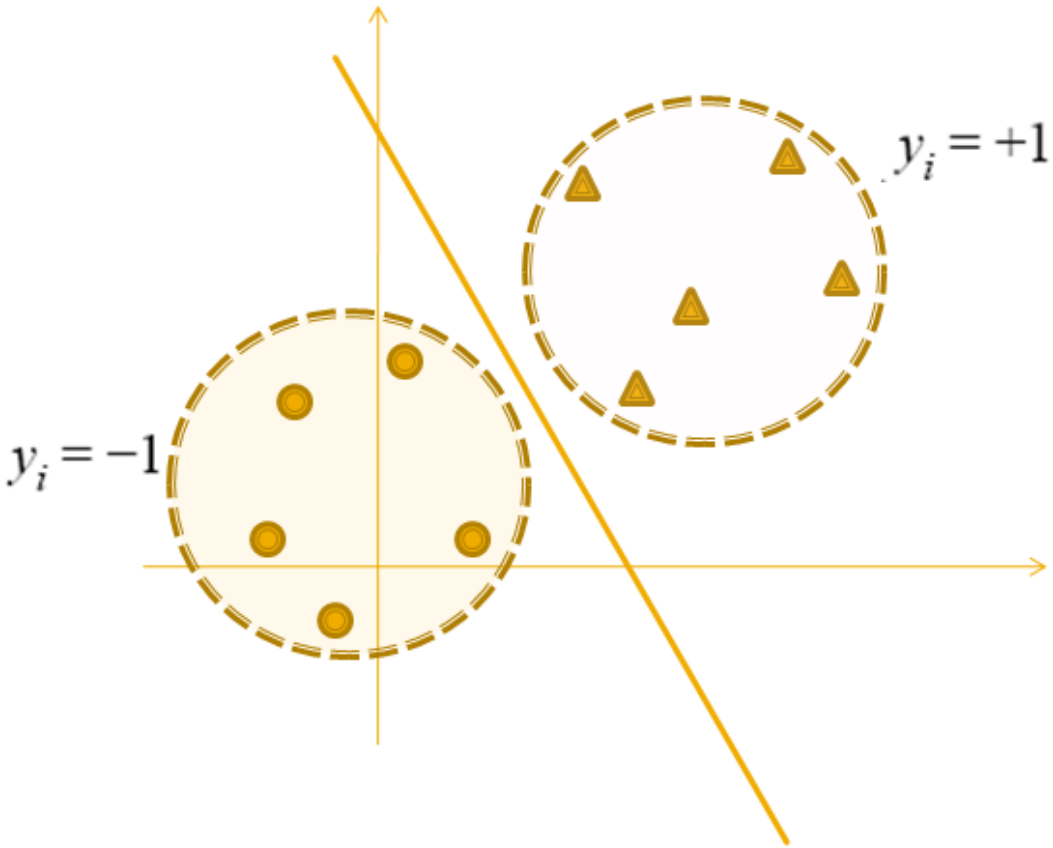
Support Vector Machines



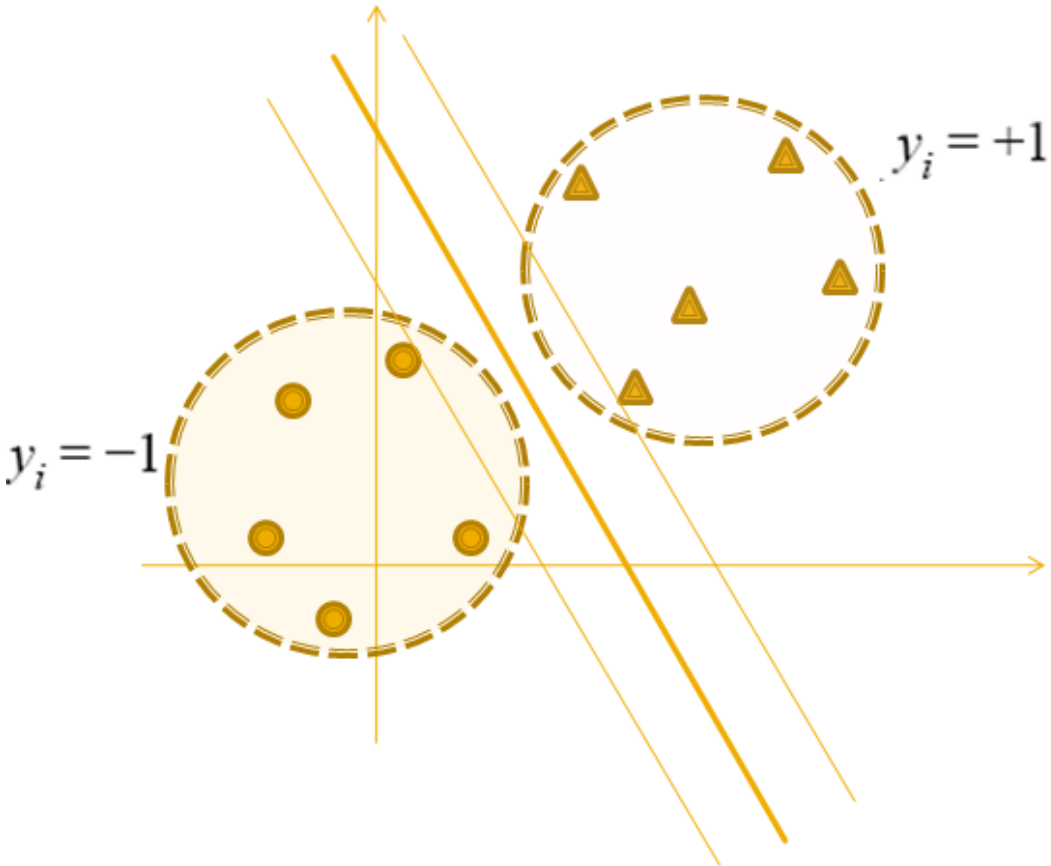
Support Vector Machines



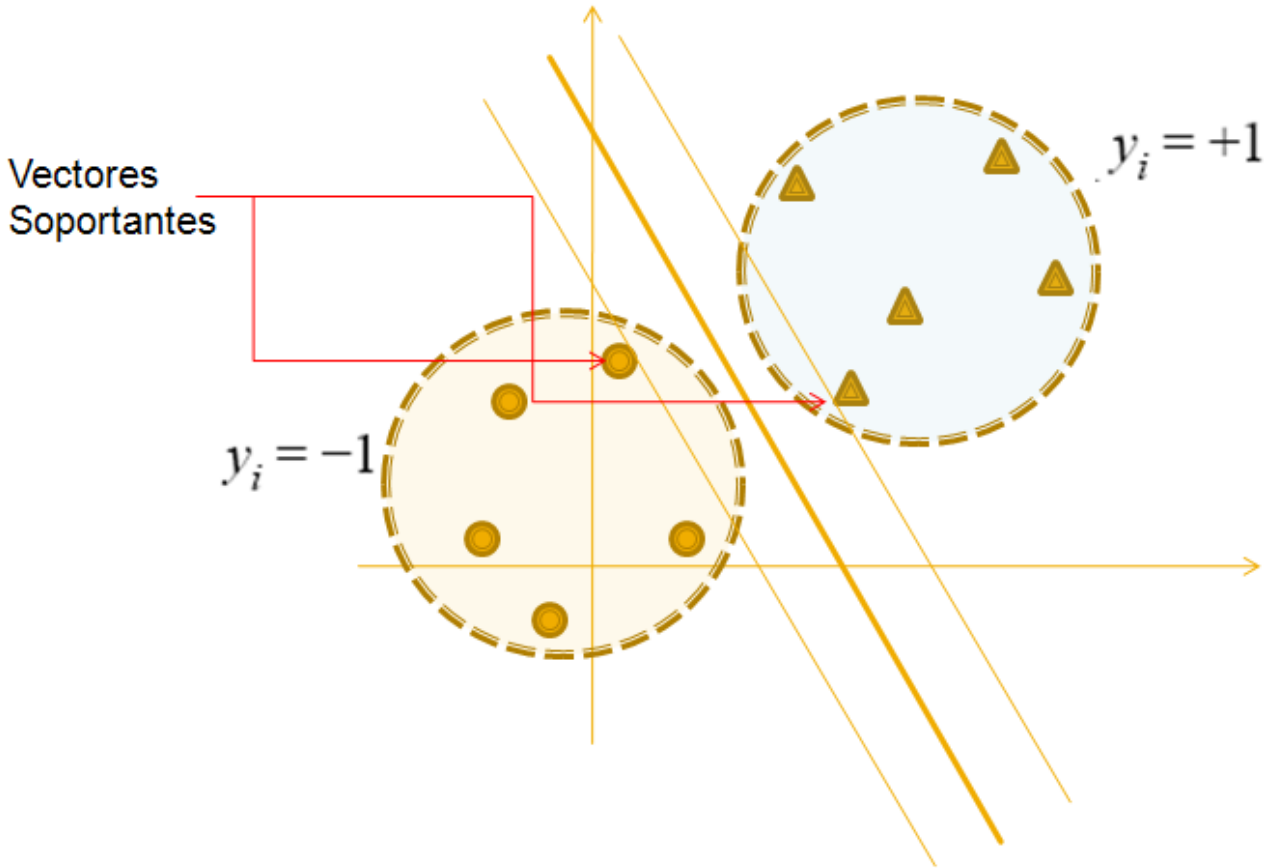
Support Vector Machines



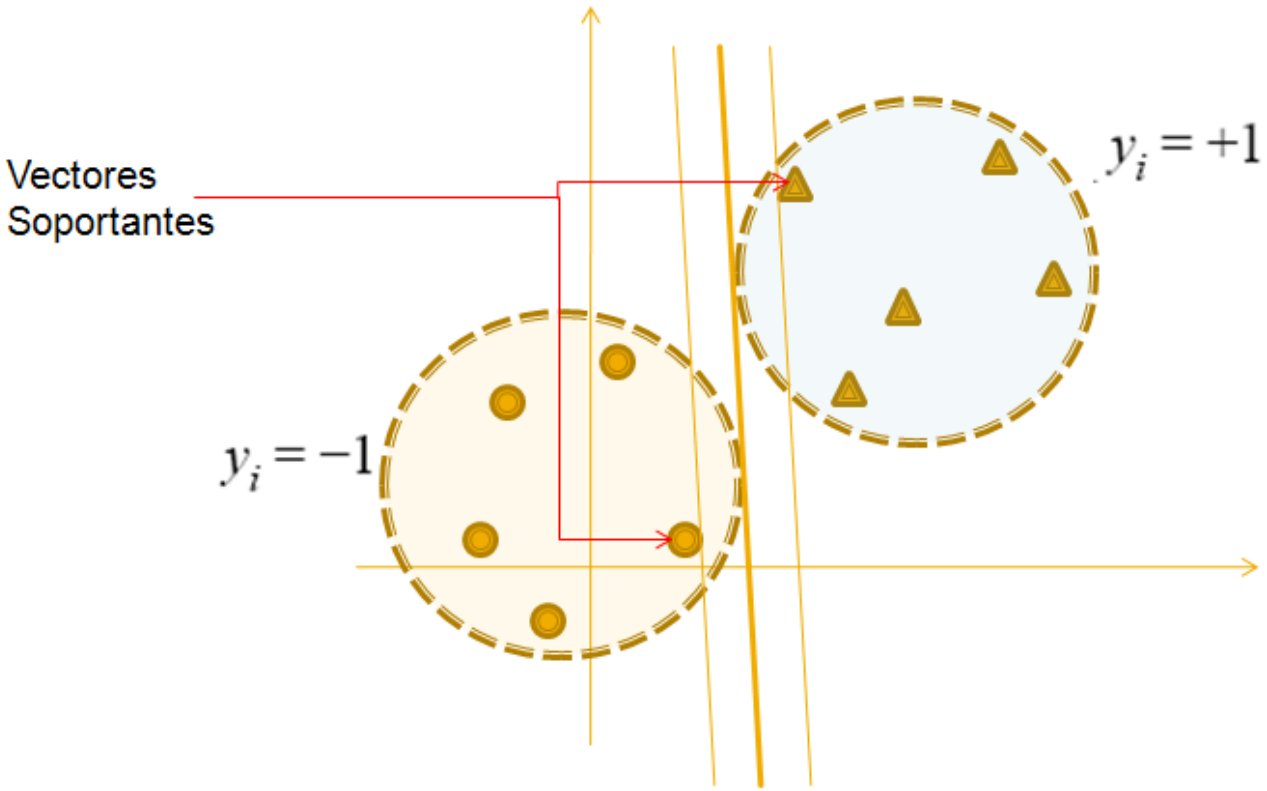
Support Vector Machines



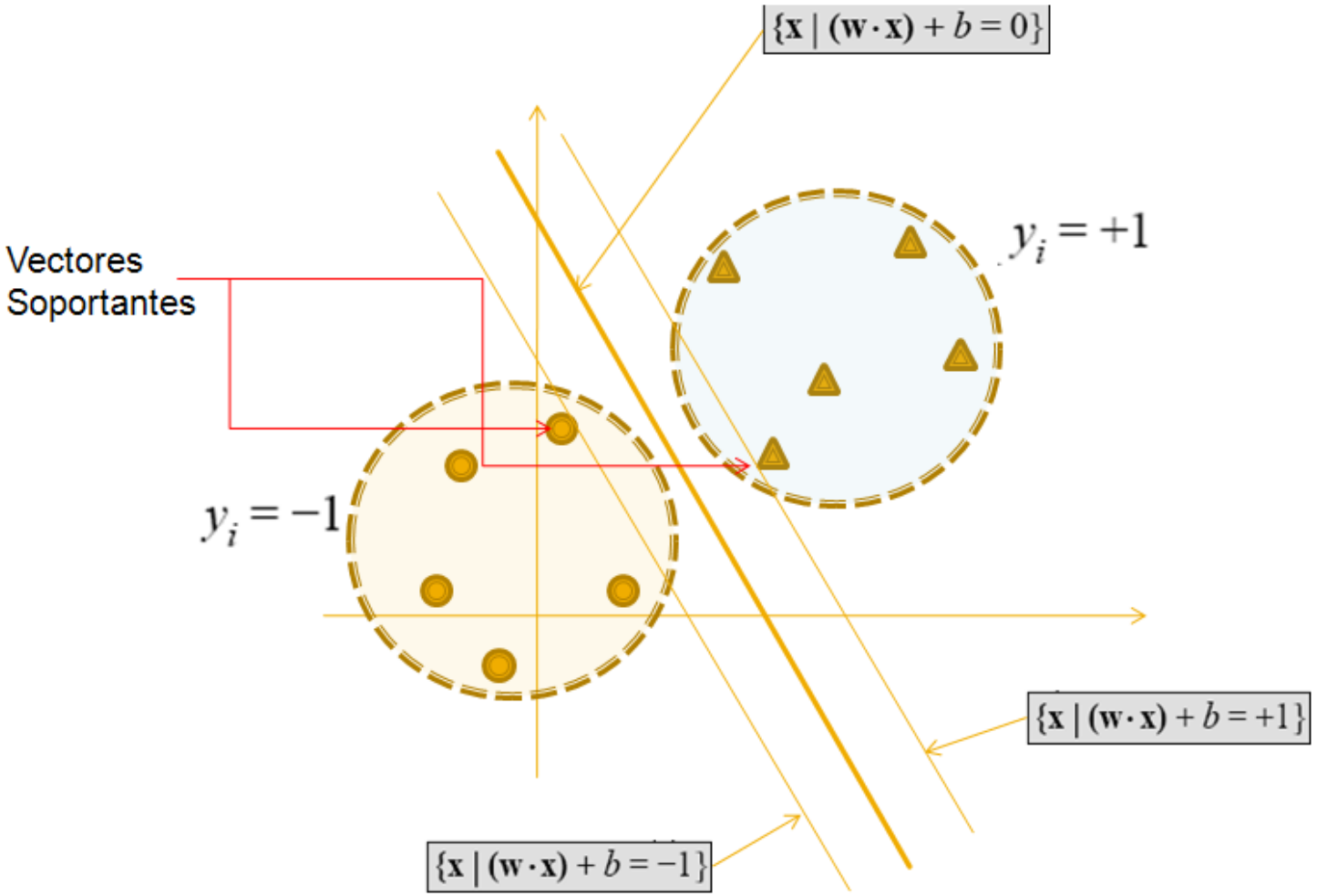
Support Vector Machines



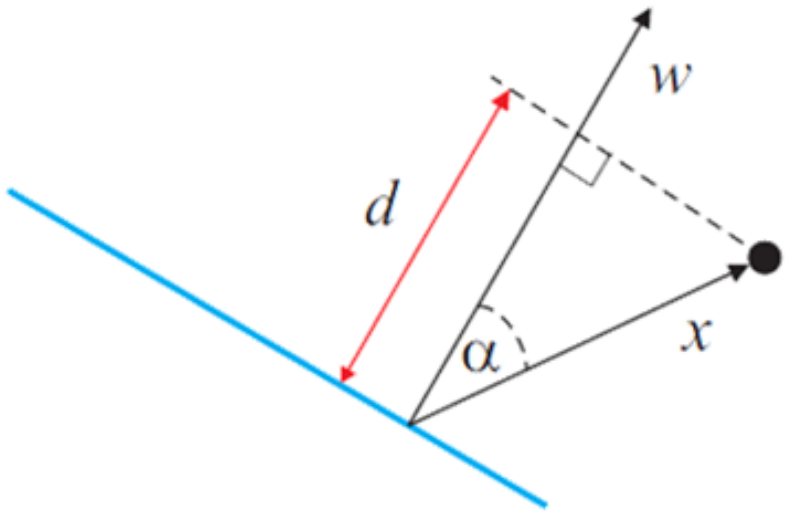
Support Vector Machines



Support Vector Machines



Support Vector Machines - Formulación



Geometría

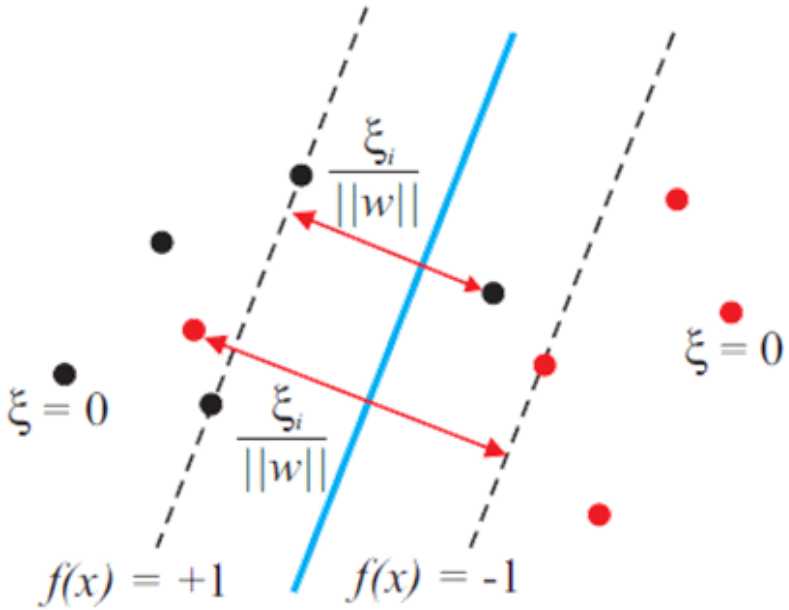
- Distancia entre $\mathbf{x}$ al hiperplano está dado por Eq. 4.
- El máximo margen es análogo a buscar el $||\mathbf{x}||$ mínimo.

$$cos(\alpha) = \frac{w^T \mathbf{x}}{||w|| \cdot ||\mathbf{x}||}, cos(\alpha) = \frac{d}{||\mathbf{x}||} \Rightarrow d = \frac{w^T \mathbf{x} + b}{||w||} \quad (4)$$

Support Vector Machines - Formulación [2]

Geometría

- Incluir la "Penalización" de malas clasificaciones en el modelo. Consideramos variables de holgura $\xi \geq 0$ para cada dato mal clasificado.
- Datos no son perfectos, debemos considerar errores para maximizar el margen.



Support Vector Machines - Formulación [3]

Finalmente tenemos lo siguiente:

- Minimizar el error en la separación de los objetos dados (del conjunto de entrenamiento)
- Maximizar el margen de separación (mejora la generalización del clasificador en conjunto de test)

La técnica Support Vector Machines (SVMs) permite cumplir con los dos objetivos anteriores, minimizando el riesgo estructural.

Support Vector Machines - Formulación [4]

Primal

$$\begin{aligned} \min_{\mathbf{w},\xi,b} \quad & \frac{1}{2}\sum_{i=1}^{A} w_i^2 + C\sum_{i=1}^{T}\xi_i \\ \text{subject to} \quad & y_i\left(w^T\mathbf{x}_i + b\right) \geq 1 - \xi_i \; \forall i \in \{1,\ldots,T\} \\ & \xi_i \geq 0 \; \forall i \in \{1,\ldots,T\} \end{aligned} \tag{5}$$

Función Lagrangeana

$$\begin{aligned} L(w,b,\alpha,\beta,\xi) = & \frac{1}{2}\sum_{i=1}^{A} w_i^2 + C\sum_{i=1}^{T}\xi_i \\ & - \sum_{i}^{N}\alpha_i\left(y_i\left(w^T\mathbf{x}_i + b\right) - 1 + \xi_i\right) - \sum_{i}^{N}\beta_i\xi_i \end{aligned} \tag{6}$$

Support Vector Machines - Formulación [5]

Condiciones de Optimalidad

$$\frac{\partial L}{\partial \mathbf{w}^*} = 0 \Rightarrow \mathbf{w}^* = \sum_i \alpha_i y_i \mathbf{x}_i \tag{7}$$

$$\frac{\partial L}{\partial b^*} = 0 \Rightarrow \sum_i \alpha_i y_i = 0 \tag{8}$$

$$\frac{\partial L}{\partial \xi^*} = 0 \Rightarrow \alpha_i + \beta_i = C \tag{9}$$

Condiciones de Holgura Complementaria (KKT)

$$\alpha_i(y_i\left(w^T\mathbf{x}_i + b\right) - 1 + \xi_i) = 0, \forall i \in 1, \ldots, N \tag{10}$$

$$\beta_i\xi_i = (C - \alpha_i)\xi_i = 0, \forall i \in 1, \ldots, N \tag{11}$$

$$\tag{12}$$

Support Vector Machines - Formulación [6]

Primal

$$\begin{aligned} \min_{\mathbf{w},\xi,b} \quad & \frac{1}{2}\sum_{i=1}^{A} w_i^2 + C\sum_{i=1}^{T}\xi_i \\ \text{subject to} \quad & y_i\left(w^T\mathbf{x}_i + b\right) \geq 1 - \xi_i \; \forall i \in \{1,\ldots,T\} \\ & \xi_i \geq 0 \; \forall i \in \{1,\ldots,T\} \end{aligned} \tag{13}$$

Dual

$$\begin{aligned} \max_{\alpha} \quad & \sum_{i=1}^{T}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{T}\alpha_i\alpha_j y_i y_j \cdot (\mathbf{x}_i \cdot \mathbf{x}_j) \\ \text{subject to} \quad & \alpha_i \geq 0, \forall i \in \{1,\ldots,T\} \\ & \sum_{i=1}^{T}\alpha_i y_i = 0 \end{aligned} \tag{14}$$

Support Vector Machines - Parámetros

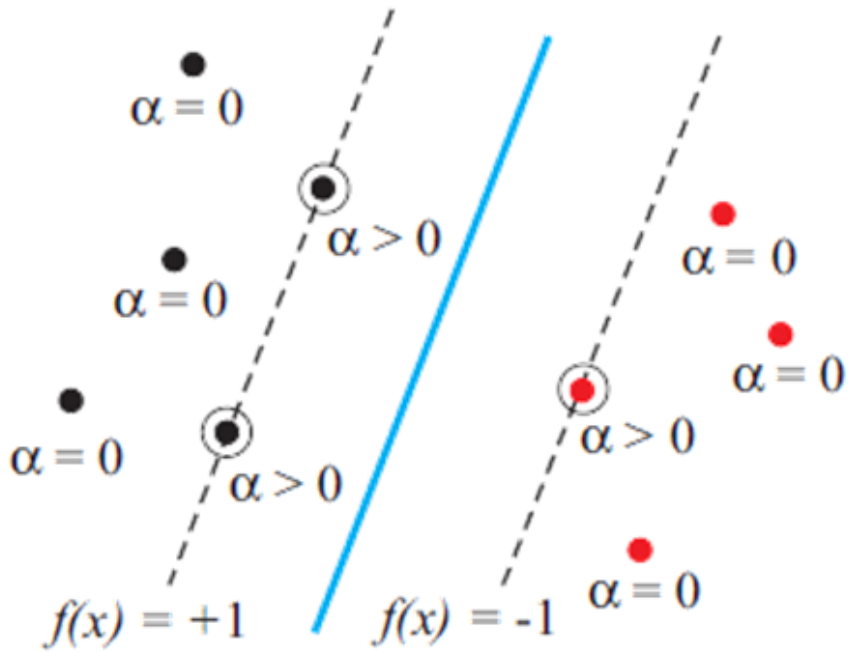
El parámetro C

- El parámetro C es quien acompaña al objetivo de minimizar el error ocasionado por la maximización del margen.
- Representa el trade-off entre el error de clasificación y la complejidad del modelo.
- El valor de C se considera como el parámetro de regularización en las SVMs.
- Referencias: [Blumer et al., 1989, Vapnik, 1999, Boser et al., 1992],

Valores grandes de C favorecen soluciones con pocos errores de clasificación.

Valores chicos de C expresa preferencias a modelos con menor complejidad.

Support Vector Machines - Propiedades del Dual



Geometría

- Resolución del problema de optimización cuadrático (QP) con algoritmo SMO (Sequential Minimal Optimization) [Platt, 1998]
- Datos con $\alpha_i \geq 0$ representan los vectores soportantes.

Modelos no lineales - Separabilidad

Resumiendo

- Si los datos no son linealmente separables se pueden considerar estrategias para lograr la separabilidad necesaria.
- Teorema: "Si el espacio es de dimensión infinita, todos los datos son linealmente separables"

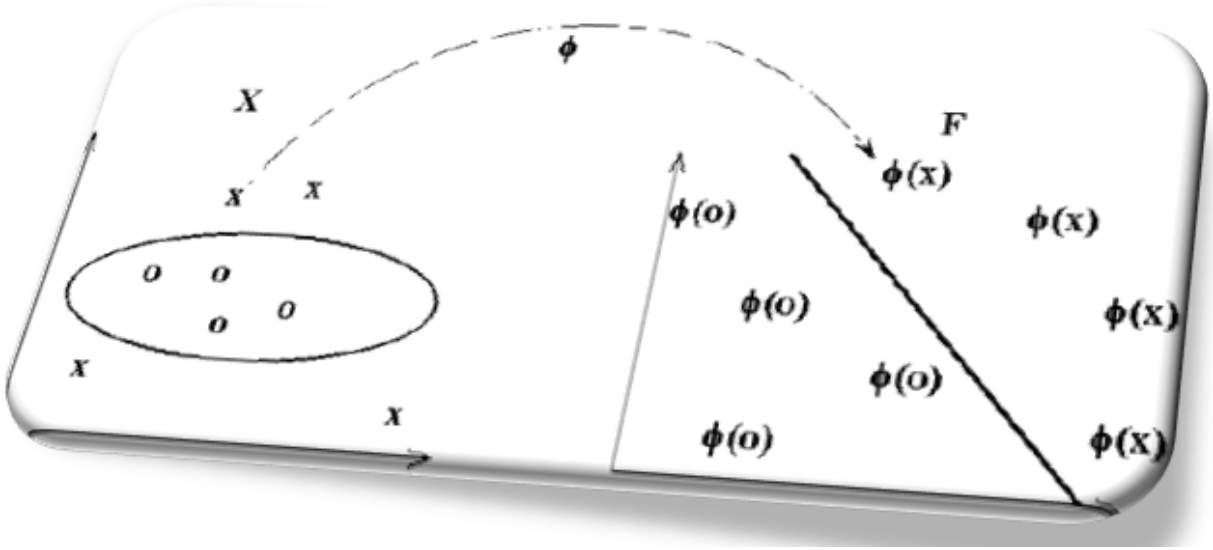
Solución: "Kernel Trick" (Regularización)

"Mapear" variables de un espacio a uno de mayor dimensión

Modelos no lineales - Kernel Trick

Kernel Trick

Cuando buscamos un hiperplano en un conjunto de datos no linealmente separable, definimos una transformación que mapea los datos en otro espacio. (“Kernel Trick”)

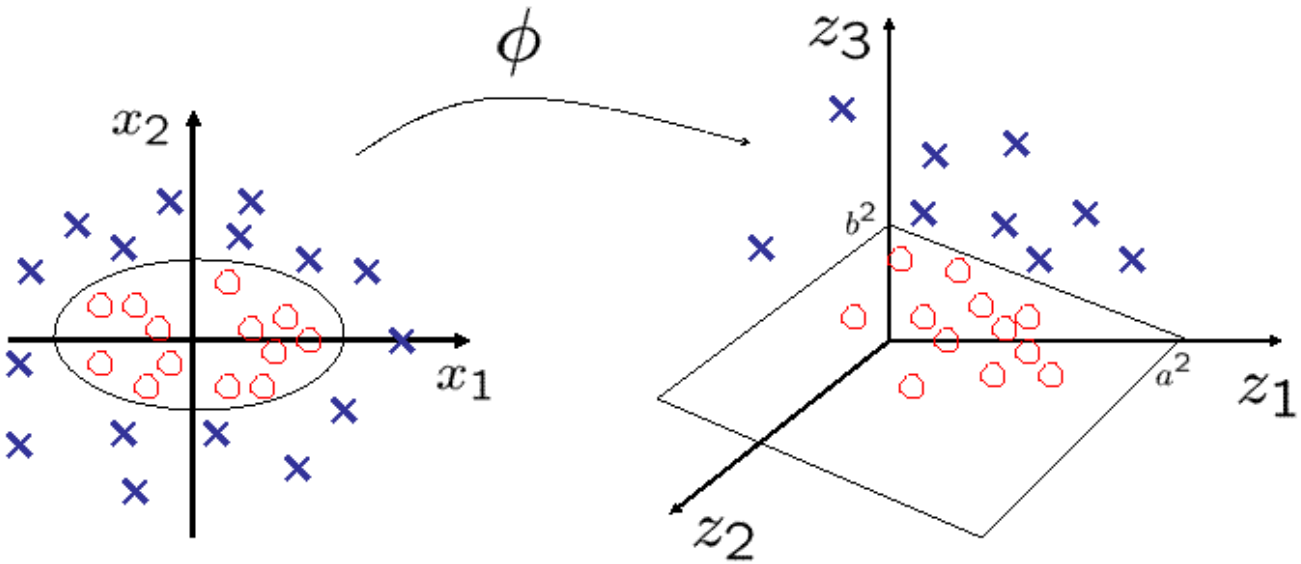


Modelos no lineales - Kernel Trick [2]

Kernel Trick

$$\phi : (x_1, x_2) \rightarrow (x_1^2, \sqrt{2}x_1 \cdot x_2, x_2^2)$$

$$\left(\frac{x_1}{a}\right)^2 + \left(\frac{x_2}{b}\right)^2 = 1 \rightarrow \frac{z_1}{a^2} + \frac{z_3}{b^2} = 1$$



Modelos no lineales - Kernel Trick [3]

Definimos la función de Mapeo:

$$\begin{aligned} \phi : \mathbb{R}^A &\rightarrow \mathcal{F} \\ \mathbf{x} &\rightarrow \phi(\mathbf{x}) \end{aligned}$$

Definimos la función de Kernel:

$$\kappa(\mathbf{x}, \mathbf{x}') := \phi(\mathbf{x}) \cdot \phi(\mathbf{x}')$$

$$\begin{aligned} \kappa : \mathbb{R}^A \times \mathbb{R}^A &\rightarrow \mathbb{R} \\ (\mathbf{x}, \mathbf{x}') &\rightarrow \kappa(\mathbf{x}, \mathbf{x}') \end{aligned}$$

Función de Kernel Trick [3]

Nuevo Dual:

Actualizamos el dual en el producto punto con nuestra nueva función de Kernel κ

$$\begin{aligned} \max_{\alpha} \quad & \sum_{i=1}^{T} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{T} \alpha_i \alpha_j y_i y_j \cdot \kappa(\mathbf{x}_i, \mathbf{x}_j) \\ \text{subject to} \quad & \alpha_i \geq 0, \forall i \in \{1, \ldots, T\} \\ & \sum_{i=1}^{T} \alpha_i y_i = 0 \end{aligned} \tag{15}$$

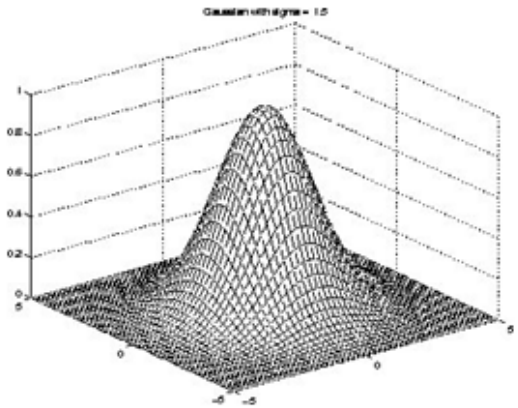
Función de decisión:

Finalmente tenemos la siguiente función de decisión:

$$f(\mathbf{x}_j) = sgn\left(\sum_{i=1}^{N} \alpha_i y_i \kappa(\mathbf{x}_i, \mathbf{x}_j) + b \right)$$

Función de Kernel [1]

$$K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{\sigma^2}\right)$$

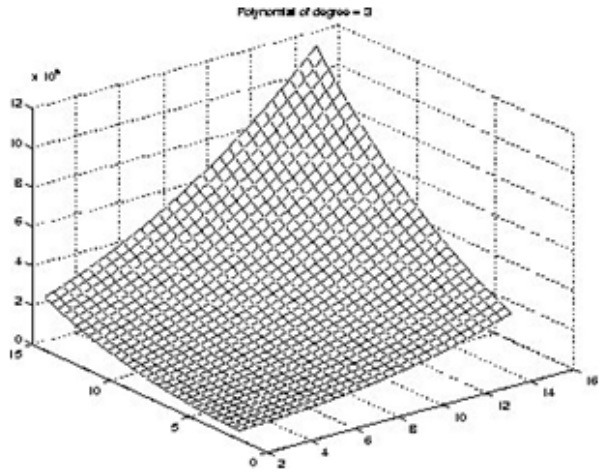


Kernel Radial Basis Function (RBF)

- Parametro σ permite ajustar una curvatura a los datos.
- Espacio característico $\mathcal{F}$ es un espacio de Hilbert de dimensión infinita. Esta propiedad permite regularizar bien el problema de separación.
- Espacio de Hilbert es un espacio vectorial en el cuál está definido un producto punto (que permite calcular el largo de un vector y angulos entre ellos).

Función de Kernel [2]

$$K(x_i, x_j) = (x_i \cdot x_j + 1)^b$$



Kernel Polinomial

- Generalización de hiperplano lineal ($w\mathbf{x} + b$)

Función de Kernel [3]

Ventajas

- Teoremas de minimización estructural y regularización son el estado del arte en la teoría de aprendizaje estadístico.
- Se puede aplicar a datos representados en cualquier espacio de Hilbert (donde pueda definir una medida de distancia).
- Relativamente pocos parámetros a estimar.
- Se pueden formular nuevas extensiones (flexibilidad).

Desventajas

- Determinar el Kernel a utilizar es complejo.
- Sólo es un clasificador binario. (¿Qué ocurre cuando tenemos un problema de más de dos clases?)

References I


Blumer, A., Ehrenfeucht, A., Haussler, D., and Warmuth, M. K. (1989).
Learnability and the vapnik-chervonenkis dimension.
J. ACM, 36(4):929–965.


Boser, B. E., Guyon, I. M., and Vapnik, V.Ñ. (1992).
A training algorithm for optimal margin classifiers.
In *COLT '92: Proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152, New York, NY, USA. ACM.


Platt, J. (1998).
Sequential minimal optimization: A fast algorithm for training support vector machines.


Vapnik, V. and Chervonenkis, A. (1971).
On the uniform convergence of relative frequencies of events to their probabilities.

Theory of Probability and its Applications, 16:264–280.


Vapnik, V.Ñ. (1999).
The Nature of Statistical Learning Theory (Information Science and Statistics).
Springer.