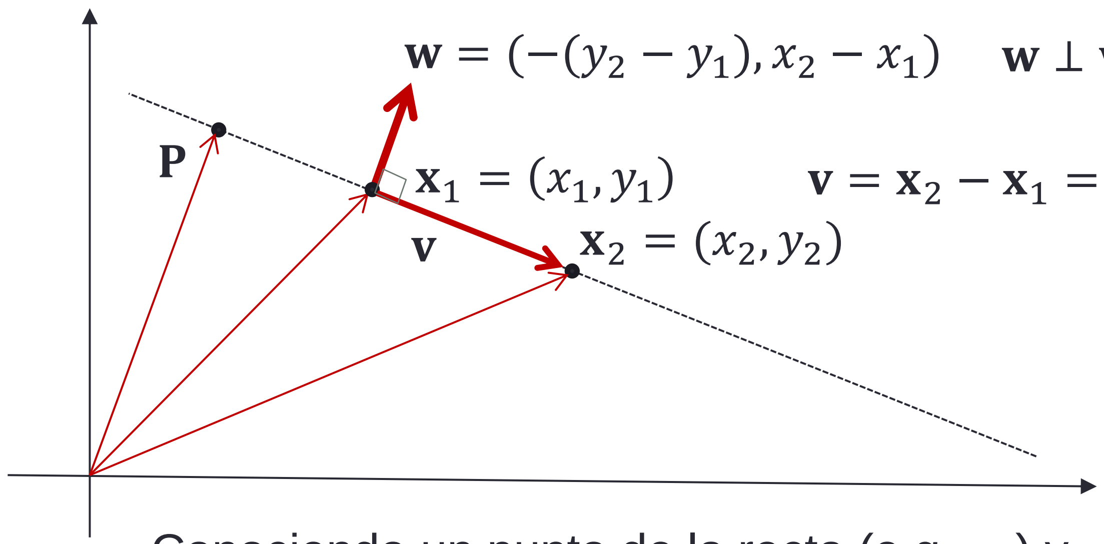
MAQUINAS DE VECTORES DE SOPORTE

Modelos Lineales

Dr. Jorge Hermosillo Valadez (jhermosillo@uaem.mx)
Depto. de Computación
CInC – IICBA, UAEM

Agosto - 2017

Geometría básica en el plano 2D



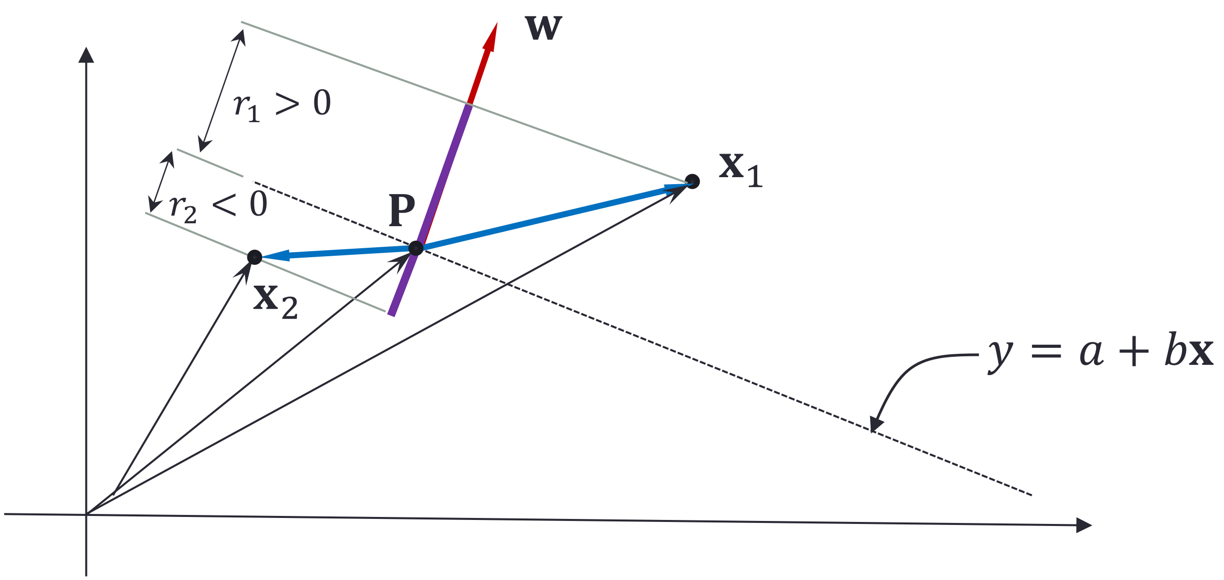
Conociendo un punto de la recta (e.g. $\mathbf{x}_1$) y $\mathbf{w}$ cualquier punto $\mathbf{P} = (x_P, y_P)$ que pertenezca a la recta satisface:

$$\mathbf{w} \cdot (\mathbf{P} - \mathbf{x}_1) = 0 \;\rightarrow (y_1 - y_2, x_2 - x_1) \cdot \; (x_P - x_1, y_P - y_1) = 0$$

$$(y_1 - y_2)(x_P - x_1) + (x_2 - x_1)(y_P - y_1) = 0$$

$$Ax_P + By_P + C = 0$$

Producto punto sobre vector perpendicular a una recta



Para todo punto $\mathbf{x} = (x, y)$ que no está sobre la recta: $y = a + b\mathbf{x}$

$$\langle \mathbf{w}, \mathbf{x} - \mathbf{P} \rangle = w_x(x - x_p) + w_y(y - y_p) = r \neq 0$$

$$\langle \mathbf{w}, \mathbf{x}_1 - \mathbf{P} \rangle = r_1 > 0$$

$$\langle \mathbf{w}, \mathbf{x}_2 - \mathbf{P} \rangle = \boldsymbol{r_2} < 0$$

donde $\langle \square, \square \rangle$ es el producto punto

Resumen del algoritmo Perceptron

- Datos de entrada:
  - $\mathbf{X} \coloneqq \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\} \subset \mathcal{X}(\text{datos}), \text{con } \mathbf{x}_i \coloneqq (x_1, x_2, \cdots, x_d) \subset \mathbb{R}^d$
  - $\mathbf{y} \coloneqq \{y_1, y_\mathbf{2}, \cdots, y_\mathbf{N}\} \subset \{\pm 1\}(\text{etiquetas})$
  - En coordenadas homogéneas:

$$\mathbf{X} \coloneqq \{(\mathbf{x}_1, 1), (\mathbf{x}_2, 1), \cdots, (\mathbf{x}_N, 1)\} \subset \mathbb{R}^N \times \mathbb{R}^{d+1}$$

- Parámetros:
  - $\mathbf{w} \coloneqq (w_1, w_2, \cdots, w_d) \subset \mathbb{R}^d$ (vector de pesos)
  - $u \in \mathbb{R}$ (umbral)
- Función (modelo lineal) en coord. homegéneas:
  - $y_i = \text{sign}(\langle \mathbf{w}, \mathbf{x}_i \rangle)$
  - $\mathbf{x}_i \coloneqq (x_1, x_2, \cdots, x_d, 1) \subset \mathbb{R}^{d+1}$,
  - $\mathbf{w} \coloneqq (w_1, w_2, \cdots, w_d, w_0) \subset \mathbb{R}^{d+1}$ donde $w_0 = u$

Algoritmo de Aprendizaje Perceptron

Entrada: Datos etiquetados de entrenamiento $\mathbf{X}$ en coordenadas homogéneas

Salida: Vector de pesos $\mathbf{w}$ que define al clasificador

$\mathbf{w} = \mathbf{0}$ *#Otras inicializaciones son posibles*

$converge = \boldsymbol{Falso}$

mientras $converge == \boldsymbol{Falso}$:

  $converge = \boldsymbol{Verdadero}$

  para i **en** $|\mathbf{X}|$:

    si $\mathrm{y_i}\langle \mathbf{w}, \mathbf{x}_i \rangle \leq 0$ **entonces**: #xi mal clasificado

      $\boldsymbol{w} = \boldsymbol{w} + y_i \eta \mathbf{x}_i$ #$0 < \eta \leq 1$ es la tasa de aprendizaje

      $converge = \boldsymbol{Falso}$

    fin

  fin

fin

Algoritmo de Aprendizaje Perceptron

Nota que el perceptrón básico aprende los pesos de los atributos

$$\begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_d \end{bmatrix} = y_1 \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \end{bmatrix}_1 + y_2 \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \end{bmatrix}_2 + \cdots + y_N \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \end{bmatrix}_N \Longrightarrow \mathbf{w} = \sum_{i=1}^{N} y_i x_i$$

- Luego del entrenamiento cada ejemplo estuvo mal clasificado 0 o α veces (en función del número de épocas; una época es una pasada completa de todos los datos).

$$\mathbf{w} = \sum_{j=1}^{N} \alpha_j y_j \mathbf{x}_j$$

Perceptron DUAL

- El vector de pesos es una combinación lineal de puntos de entrenamiento:

$$\mathbf{w} = \sum_{j=1}^{N} \alpha_j y_j \mathbf{x}_j$$

- En este caso la salida del clasificador $y_i = \operatorname{sign}(\langle \mathbf{w}, \mathbf{x}_i \rangle)$ se escribe:

$$y_i = \operatorname{sign}\left(\sum_{j=1}^{N} \alpha_j y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle\right)$$

- Es decir que podemos aprender los pesos α de las instancias $\mathbf{x}_\mathrm{i}$, en lugar de los pesos de los atributos.

- Por lo tanto la única información necesaria es la matriz n-por-n $\mathbf{G} = \mathbf{X}\mathbf{X}^\mathrm{T}$, llamada *matriz Gram*, que contiene todos los productos punto.

Algoritmo del Perceptron DUAL

$\boldsymbol{\alpha} \coloneqq (\alpha_1, \alpha_2, \cdots, \alpha_N) = 0$

$converge = \boldsymbol{Falso}$

mientras $converge == \boldsymbol{Falso}$:

$converge = \boldsymbol{Verdadero}$

para i **en** $|\mathbf{X}|$:

toma $(\mathbf{x}_i, y_i)$ de los datos

si $\mathrm{y_i} \sum_{j=1}^{|\mathbf{X}|} \alpha_j y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle \leq 0$ **entonces**: #xi mal clasificado

$\alpha_i = \alpha_i + 1$

$converge = \boldsymbol{Falso}$

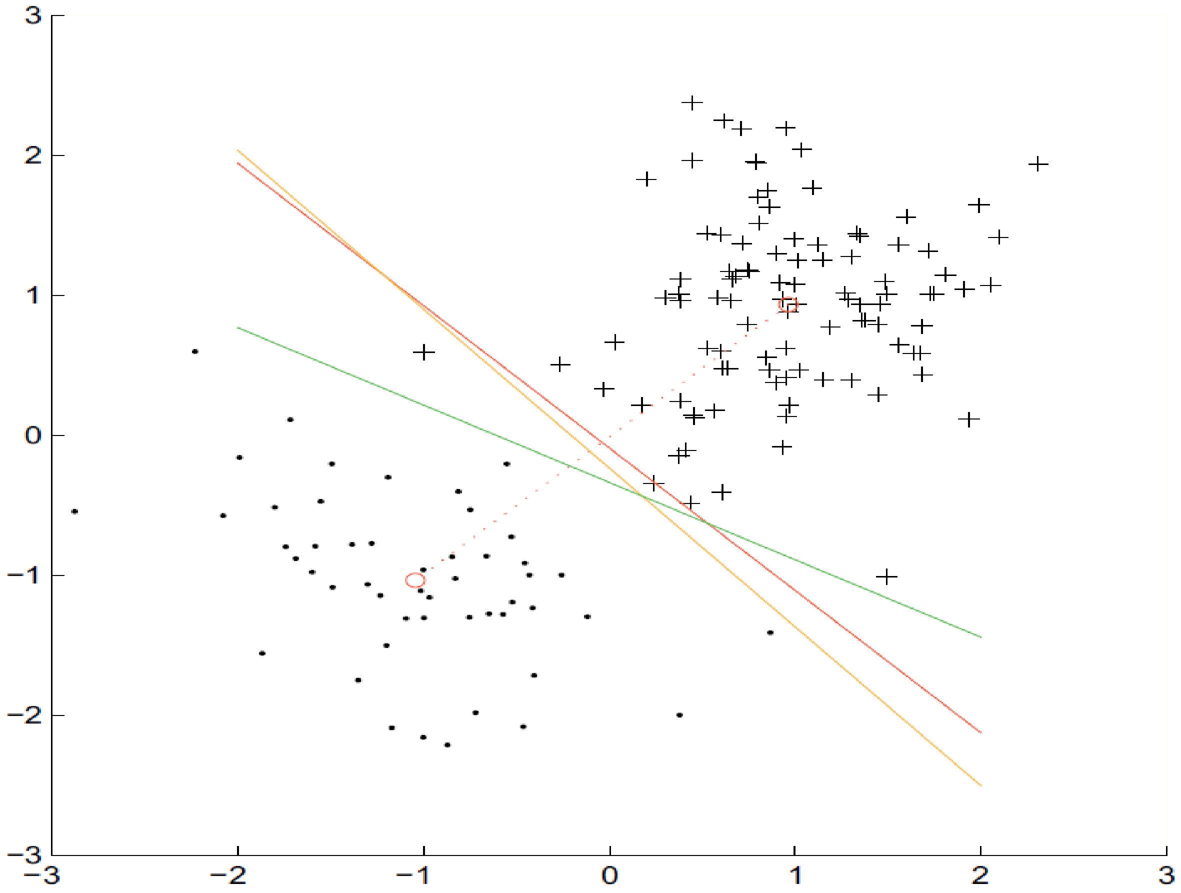
fin

fin

fin

¿Cuál es mejor?

Tres distintos clasificadores lineales: ¿cómo estandarizar la ubicación del hyperplano?

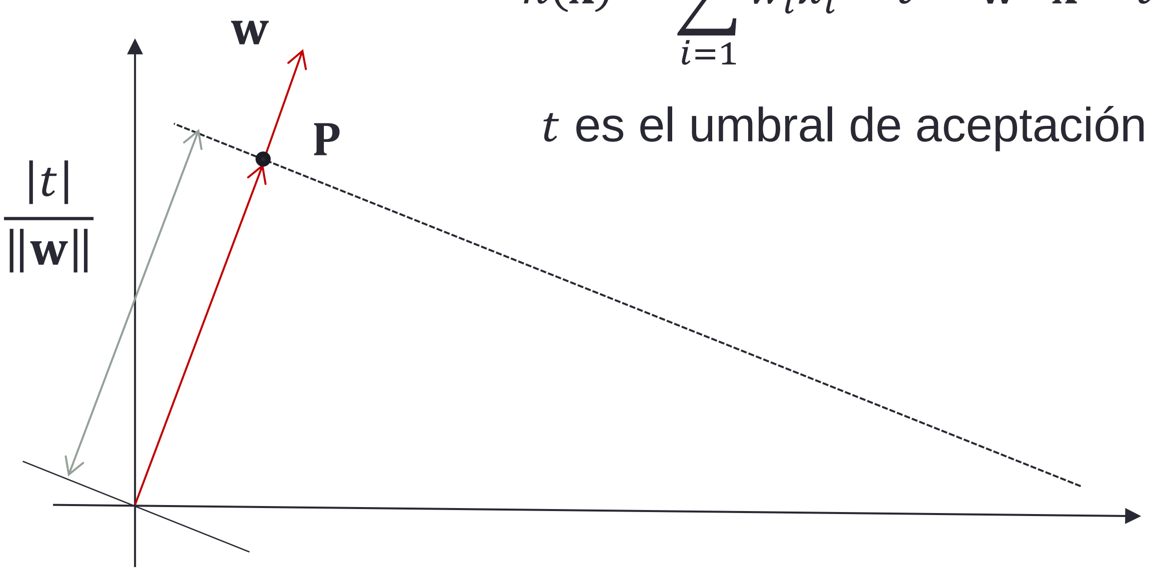


Propiedades del Perceptron

$$h(\mathbf{x}) = \sum_{i=1}^{d} w_i x_i - t = \mathbf{w} \cdot \mathbf{x} - t$$

t es el umbral de aceptación

$$\mathrm{d} = \frac{|t|}{\|\mathbf{w}\|}$$



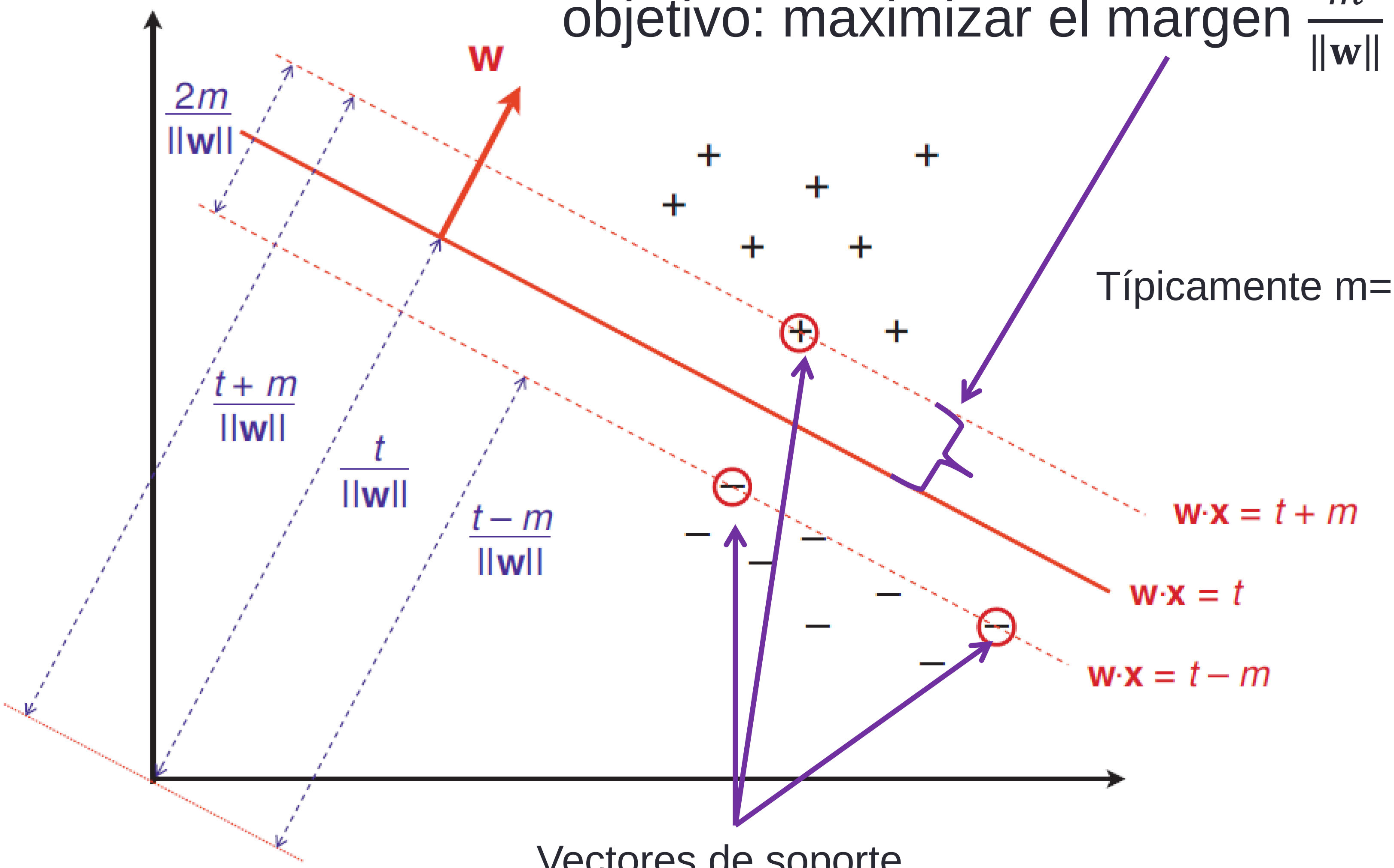
$$h(\mathbf{P}) = \mathbf{w} \cdot \mathbf{P} - t = 0$$

$$\|\mathbf{w}\| \|\mathbf{P}\| \cos \mathbf{0} = |t|$$

Clasificador por vectores de soporte

objetivo: maximizar el margen $\frac{m}{\|\mathbf{w}\|}$

Típicamente m=1



Vectores de soporte

Problema de optimización con restricciones

Maximizar el margen es equivalente a **<u>minimizar</u>** $\mathbf{w}$ o mejor aún $\frac{\mathbf{1}}{\mathbf{2}}\|\mathbf{w}\|^{\mathbf{2}}$

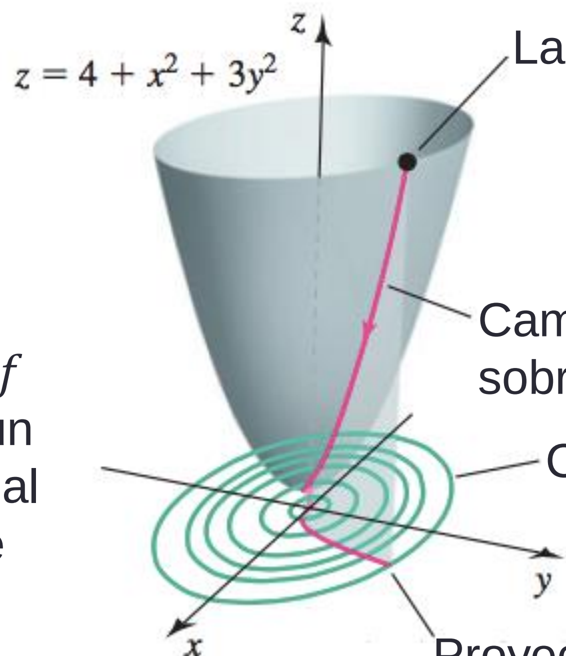
$$\mathbf{w}^*, t^* = \underset{\mathbf{w},t}{\operatorname{argmin}} \frac{1}{2}\|\boldsymbol{w}\|^2 \text{ sujeto a } y_i\left(\langle \mathbf{w}, \mathbf{x}_i \rangle - t\right) \geq 1, \qquad 1 \leq i \leq N$$

Para ello usaremos el metodo de Multiplicadores de Lagrange

Fuentes de consulta:

- http://www-mtl.mit.edu/Courses/6.050/2004/unit9/wyatt.apr.7.pdf
- https://www.khanacademy.org/math/multivariable-calculus/applications-of-multivariable-derivatives/constrained-optimization/a/lagrange-multipliers-single-constraint

Descenso de gradiente



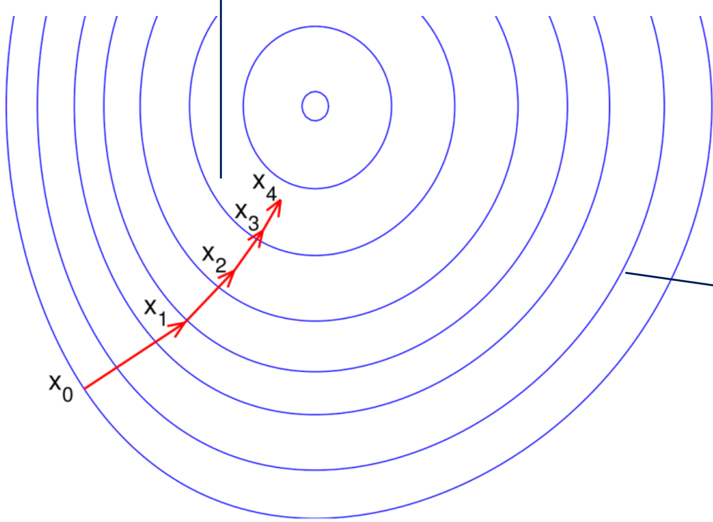
La pelota se suelta en (3,4,61)

Camino de descenso más pronunciado sobre la superficie

Curvas de nivel o de contorno

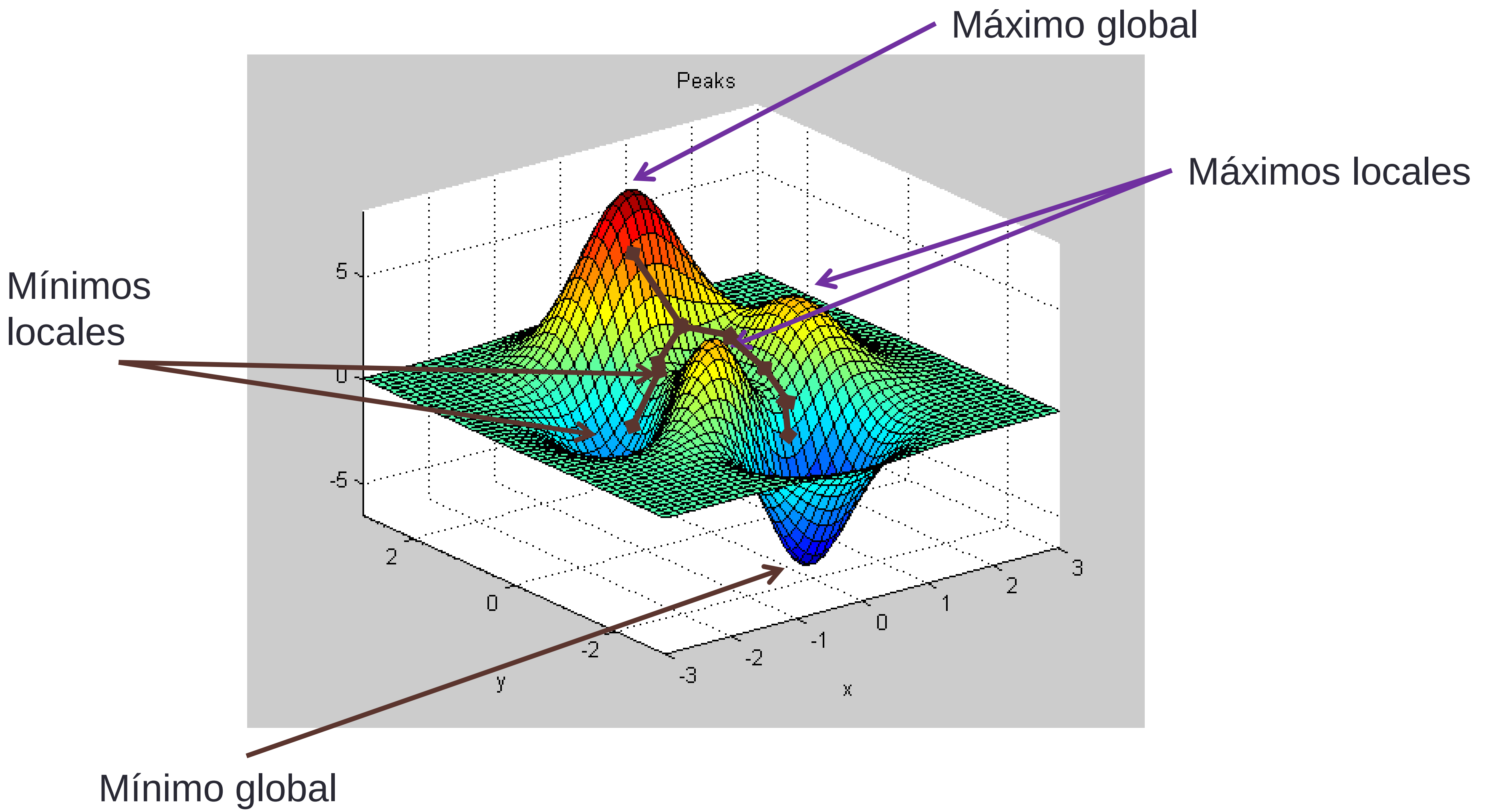
Proyección del camino de descenso más pronunciado en el plano xy.

El gradiente ∇f de $f(y,x)$ es un vector ortogonal a las líneas de contorno

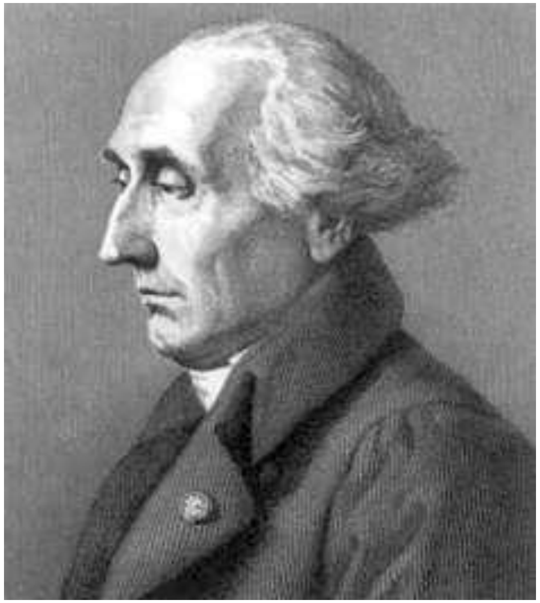


Mapa de contorno: $f(y,x) = d$

Mínimos/Máximos locales y globales



Método de Lagrange



Joseph Louis de Lagrange

Turín, 25 de enero de 1736-París, 10 de abril de 1813), fue un físico, matemático y astrónomo italiano naturalizado francés, que después de formarse en su Italia natal pasó la mayor parte de su vida en Prusia y Francia.

Método de Lagrange

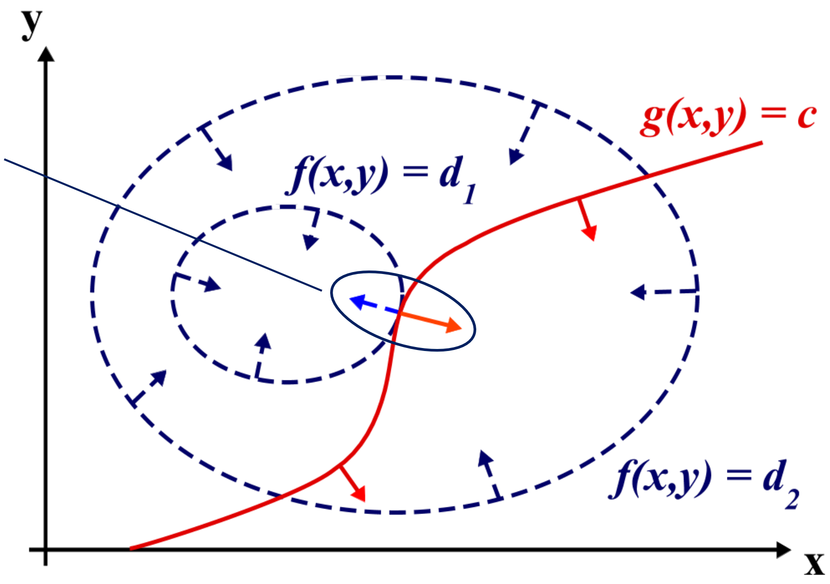
- La técnica del multiplicador de Lagrange permite encontrar el máximo o mínimo de una función multivariable $f(x_1, x_2, \dots, x_N)$ cuando hay alguna restricción en los valores de entrada del tipo:

$$\mathrm{g}(x_1, x_2, \dots, x_N) = c \qquad c \in \mathbb{R}$$

- La idea central es buscar puntos en los que las líneas de contorno de $f(x_1, x_2, \dots, x_N)$ y $\mathrm{g}(x_1, x_2, \dots, x_N)$ son tangentes entre sí.

puntos donde los vectores de gradiente de ambas curvas son paralelos entre sí.

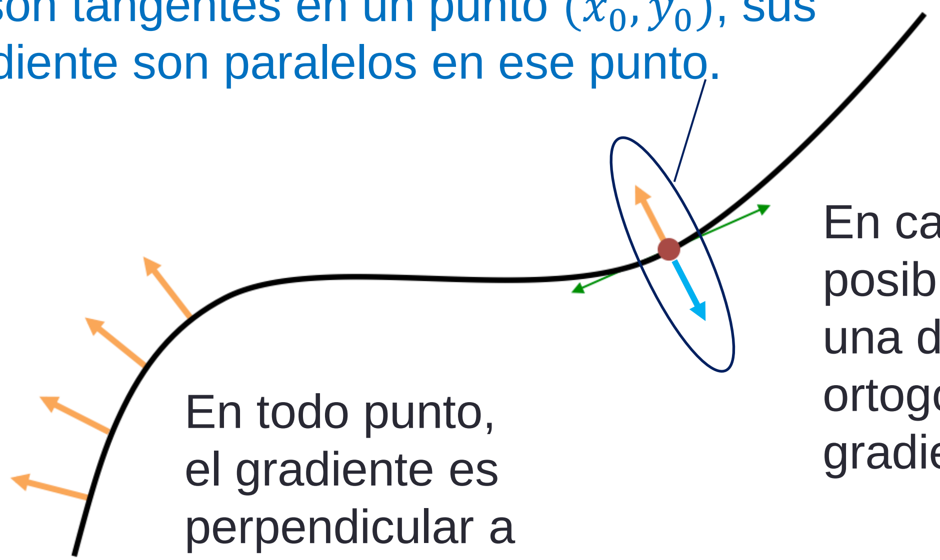
¡Veamos el gradiente!



Método de Lagrange

- Como hemos mencionado el gradiente ∇f de f evaluado en un punto (x_0, y_0) siempre da un vector perpendicular a la línea de contorno que pasa por ese punto.

Esto significa que cuando las líneas de contorno de dos funciones f y g son tangentes en un punto (x_0, y_0), sus vectores de gradiente son paralelos en ese punto.



Multiplicador de Lagrange

- La condición de paralelismo se traduce en :

$$\nabla f(x_1, x_2, \dots, x_N) = \lambda_0 \nabla \mathrm{g}(x_1, x_2, \dots, x_N) \qquad (1)$$

- Para una función $f(x_1, x_2, \dots, x_N)$:

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_N} \end{bmatrix}$$

- Notación práctica: $\nabla f = \left(\partial_{x_1} f, \partial_{x_2} f, \cdots, \partial_{x_N} f\right)$

- En esta forma (1) se puede desglosar como:

$$\begin{aligned} \partial_{x_1} f &= \lambda_0 \partial_{x_1} g \\ \partial_{x_2} f &= \lambda_0 \partial_{x_2} g \\ &\vdots \\ \partial_{x_N} f &= \lambda_0 \partial_{x_N} g \end{aligned}$$

Lagrangiano

- Lagrange propuso una nueva función especial (el Lagrangiano) que recoge las mismas variables de entrada que f y g, junto con λ que ahora se considera una variable en lugar de una constante:

$$\mathcal{L}(x_1, x_2, \dots, x_N, \lambda) = f(x_1, x_2, \dots, x_N) - \lambda(\mathrm{g}(x_1, x_2, \dots, x_N) - c)$$

- Los puntos de tangencia se encuentran calculando:

$$\nabla\mathcal{L}(x_1, x_2, \dots, x_N, \lambda) = 0$$

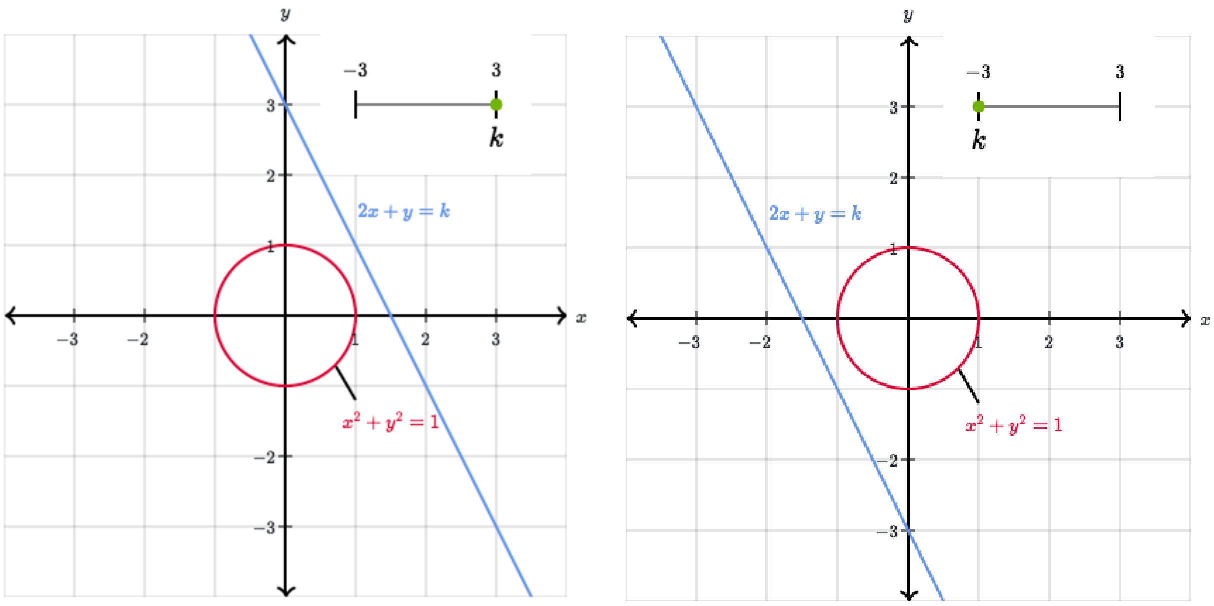
Ejemplo

- Encuentre el valor máximo de las lineas de contorno f:

$$f(x,y) = 2x + y = k$$

con la restricción sobre la entrada (x,y) tal que éstas deben cumplir con:

$$g(x,y) = x^2 + y^2 = 1$$



Ejemplo (Lagrangiano)

- Para este ejemplo el Lagrangiano se escribe:

$$\begin{aligned}\mathcal{L}(x,y,\lambda) &= f(x,y) - \lambda(\mathrm{g}(x,y) - 1)\\ &= 2x + y - \lambda(x^2 + y^2 - 1)\end{aligned}$$

- Podemos ver que nuestra restricción se escribe:

$$\partial_\lambda \mathcal{L}(x,y,\lambda) = 0 \implies \mathrm{g}(x,y) = 1$$

- También vemos que $\nabla\mathcal{L}$ tiene dos componentes:

$$\nabla f = \begin{bmatrix}2\\1\end{bmatrix} \text{ y } \nabla g = \begin{bmatrix}2x\\2y\end{bmatrix}$$

- Por lo que la condición de tangencia $\nabla\mathcal{L} = 0$ se escribe:

$$\begin{bmatrix}2\\1\end{bmatrix} = \lambda_0 \begin{bmatrix}2x_0\\2y_0\end{bmatrix}$$

Ejemplo (planteamiento de la solución)

- Resumiendo buscamos:

  - Un punto (x_0, y_0) tal que: $\mathrm{g}(x_0, y_0) = 1$ que para el ejemplo eso es:

$$x_0^2 + y_0^2 = 1$$

  - Además este punto es tal que:

$$\begin{cases} 2\lambda_0 x_0 = 2 \\ 2\lambda_0 y_0 = 1 \end{cases}$$

- Calcula (x_0, y_0) y λ_0:

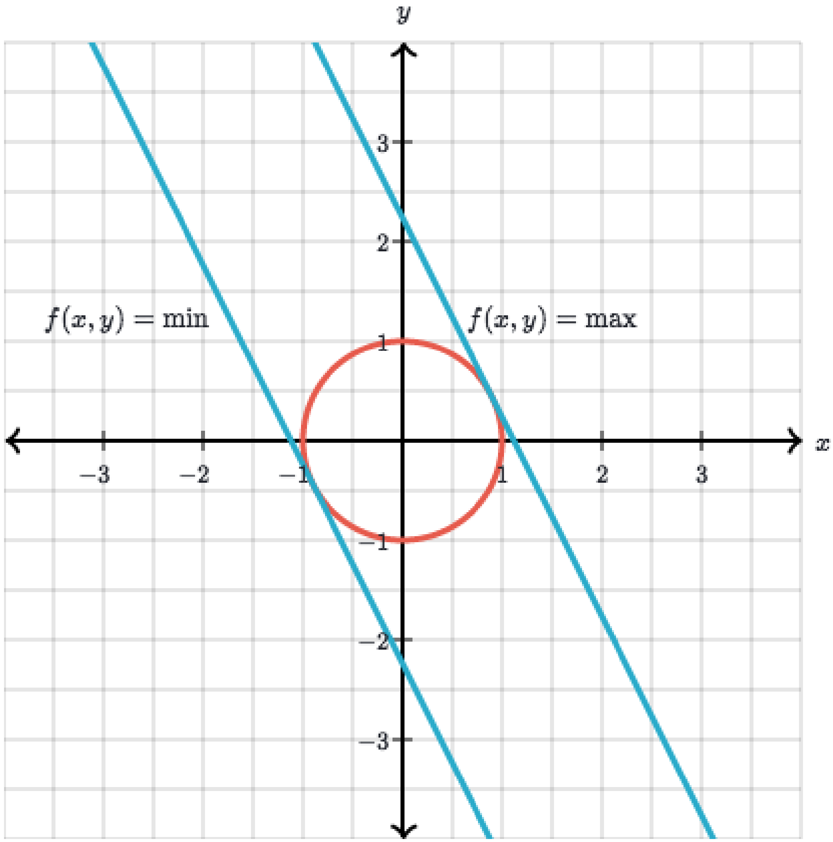
Ejemplo (solución)

$$\lambda_0 = \frac{\pm\sqrt{5}}{2} \qquad (x_0, y_0) = \left(\frac{1}{\lambda_0}, \frac{1}{2\lambda_0}\right)$$

$$(x_0, y_0) = \left(\frac{2}{\sqrt{5}}, \frac{1}{\sqrt{5}}\right) \Longrightarrow f(x_0, y_0) = \max$$

o bien

$$(x_0, y_0) = \left(\frac{-2}{\sqrt{5}}, \frac{-1}{\sqrt{5}}\right) \Longrightarrow f(x_0, y_0) = \min$$



Método de optimización por multiplicadores de Lagrange

- Paso 1: Define el lagrangiano

$$\mathcal{L}(x, y, \cdots, \lambda) = f(x, y, \ldots) - \lambda(\mathrm{g}(x, y, \ldots) - c)$$

- Paso 2: Plantea las ecuaciones de soluciones óptimas

$$\nabla \mathcal{L}(x, y, \cdots, \lambda) = 0$$

- Paso 3:
  - Considera cada solución, que se verá como $({}^{1}x_0, {}^{2}x_0, \ldots, {}^{N}x_0, \lambda_0)$.
  - Introduce cada punto sin λ en f como entrada.
  - Cualquiera que dé el mayor (o más pequeño) valor es el punto máximo (o mínimo) que estás buscando.

El problema de optimizacion en SVM's

- Nuestro objetivo es:

$$\mathbf{w}^*, t^* = \underset{\mathbf{w}, t}{\operatorname{argmin}} \frac{1}{2} \|\boldsymbol{w}\|^2$$

- Sujeto a las siguientes N restricciones:

$$y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle - t) \geq 1, \qquad 1 \leq i \leq N$$

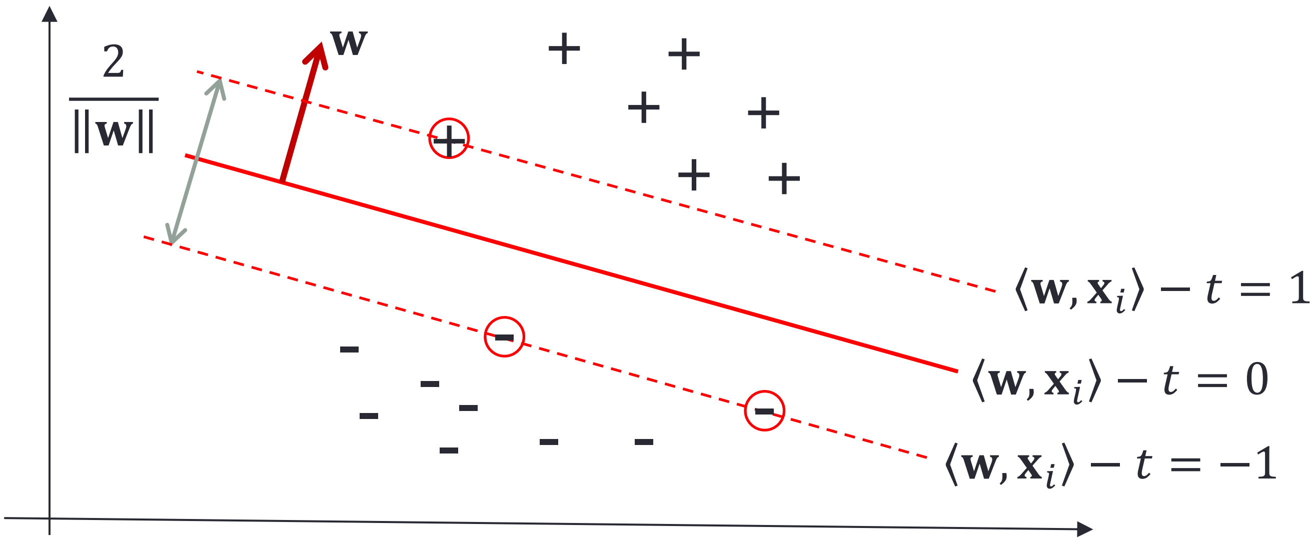
El problema de optimizacion en SVM's

- Nuestro objetivo es:

$$\mathbf{w}^*, t^* = \underset{\mathbf{w},t}{\operatorname{argmin}} \frac{1}{2} \|\boldsymbol{w}\|^2$$

- Sujeto a las siguientes N restricciones:

$$y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle - t) \geq 1, \qquad 1 \leq i \leq N$$



El problema de optimizacion en SVM's

- Definimos el lagrangiano

$$\mathcal{L}_P(\mathbf{w}, t, \alpha_1, \cdots, \alpha_N) = \frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_{i=1}^{N} \alpha_i (yi\,(\langle \mathbf{w}, \mathbf{x}_i \rangle - t\,) - \, 1)$$

$$= \frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_{i=1}^{N} \alpha_i y_i \langle \mathbf{w}, \mathbf{x}_i \rangle + \sum_{i=1}^{N} \alpha_i y_i t + \sum_{i=1}^{N} \alpha_i$$

$$= \frac{1}{2}\langle \mathbf{w}, \mathbf{w} \rangle - \left\langle \mathbf{w}, \sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i \right\rangle + t\left(\sum_{i=1}^{N} \alpha_i y_i\right) + \sum_{i=1}^{N} \alpha_i$$

- Para un t óptimo $\partial_t \mathcal{L}_P = 0 \Longrightarrow \sum_{i=1}^{N} \alpha_i y_i = 0$
- Para pesos óptimos $\partial_{\mathbf{w}} \mathcal{L}_P = 0 \Longrightarrow \mathbf{w} = \sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i$

Modelo dual de optimización

- Reinsertando estas expresiones en $\mathcal{L}_P$ obtenemos $\mathcal{L}_D$ el lagrangiano del problema dual:

$$\mathcal{L}_D(\alpha_1, \cdots, \alpha_N) = -\frac{1}{2}\left\langle \sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i , \sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i \right\rangle + \sum_{i=1}^{N} \alpha_i$$

$$= -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle + \sum_{i=1}^{N} \alpha_i$$

Modelo dual de optimización

- El problema de optimización dual es el siguiente:

$$\alpha_1^*, \cdots, \alpha_N^* \;=\; \underset{\alpha_1,\cdots,\alpha_N}{\operatorname{argmax}} -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\,\alpha_j y_i y_j\,\langle \mathbf{x}_i, \mathbf{x}_j\rangle + \sum_{i=1}^{N}\alpha_i$$

- Sujeto a las restricciones:

$$\alpha_i > 0\;,\; 1 \leq i \leq N \qquad \text{y} \qquad \sum_{i=1}^{N}\alpha_i y_i = 0$$

Aspectos importantes

- La forma dual del problema de optimización de las SVM ilustra dos aspectos importantes :

$$\alpha_1^*, \cdots, \alpha_N^* = \underset{\alpha_1,\cdots,\alpha_N}{\operatorname{argmax}} -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \, \alpha_j y_i y_j \left\langle \mathbf{x}_i, \mathbf{x}_j \right\rangle + \sum_{i=1}^{N} \alpha_i$$

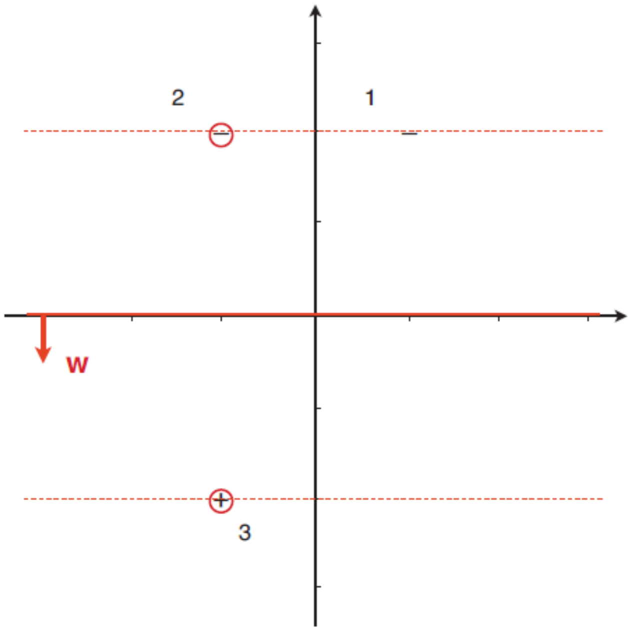
1. Maximizar el margen es equivalente a encontrar los vectores de soporte; es decir, los puntos para los cuales los multiplicadores de Lagrange son no nulos.
2. El problema de optimización está completamente definido por el producto punto de pares de instancias de entrenamiento: las entradas de la matriz Gram.

Ejercicio

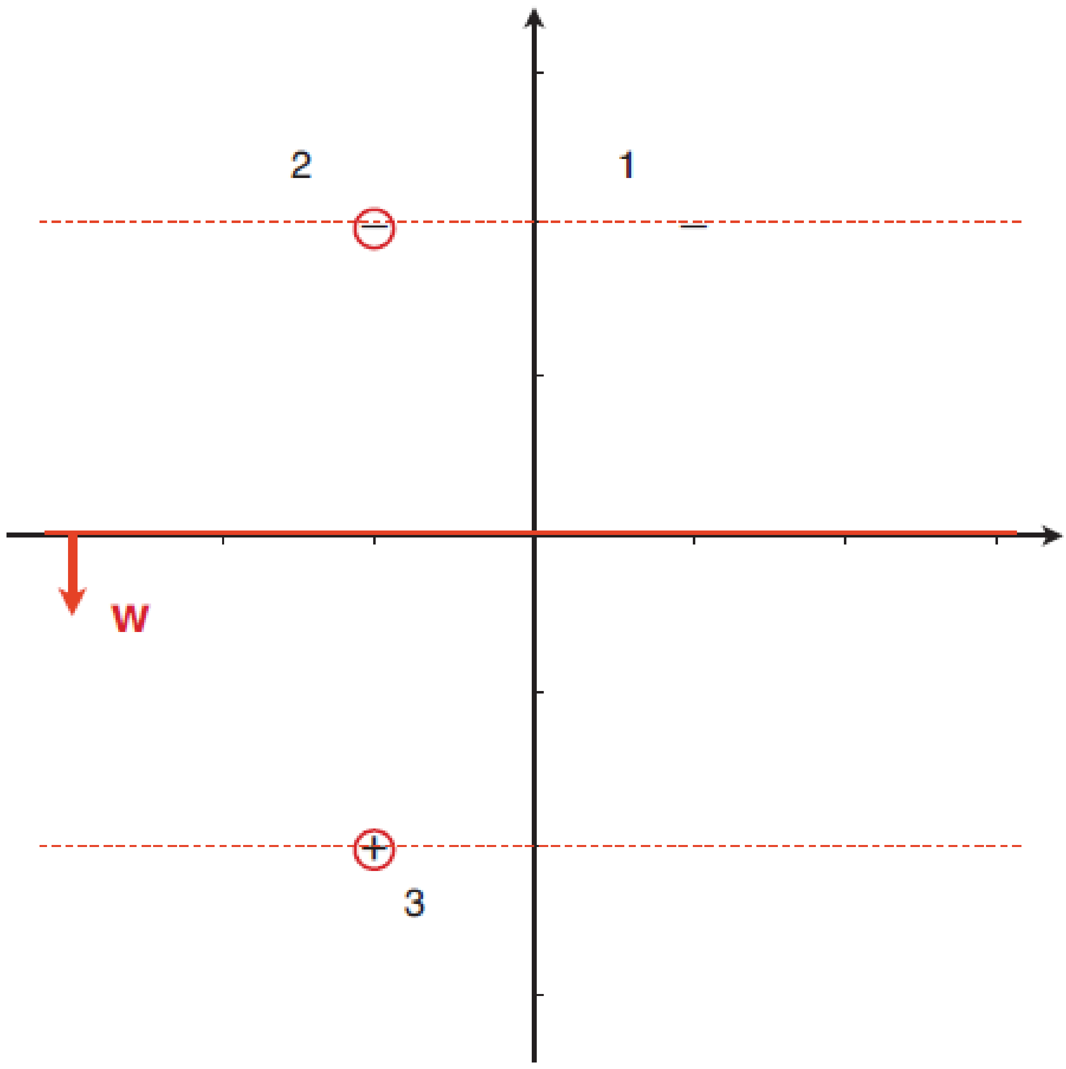
Let the data points and labels be as follows (see Figure):

$$\mathbf{X} = \begin{pmatrix} 1 & 2 \\ -1 & 2 \\ -1 & -2 \end{pmatrix} \qquad \mathbf{y} = \begin{pmatrix} -1 \\ -1 \\ +1 \end{pmatrix} \qquad \mathbf{X}' = \begin{pmatrix} -1 & -2 \\ 1 & -2 \\ -1 & -2 \end{pmatrix}$$

The matrix $\mathbf{X}'$ on the right incorporates the class labels; i.e., the rows are $y_i\mathbf{x}_i$.



1. Encuentre la matriz Gram
2. Exprese el problema de optimización Dual
3. Encuentre los vectores de soporte



2
1
w
3