Aprendizaje Automático

Segundo Cuatrimestre de 2016

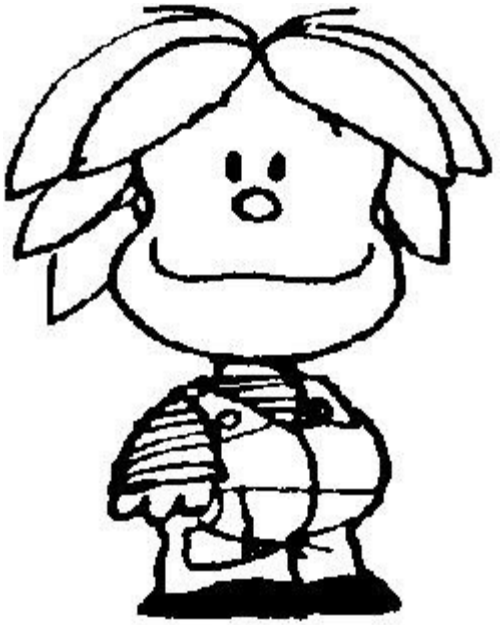
Clasificadores:

Naive Bayes, Vecinos Más Cercanos, SVM

DEPARTAMENTO
DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

Naive Bayes



Naive Bayes

Dada una nueva instancia con valores de atributos $a_1, a_2, ..., a_n$, su valor (o clase) más probable a posteriori v_{MAP} puede expresarse así:

$$v_{\text{MAP}} = \underset{v_j \in V}{\operatorname{argmax}} \; P(v_j \mid a_1, a_2, ..., a_n)$$

Bayes

$$= \underset{v_j \in V}{\operatorname{argmax}} \; \frac{P(a_1, a_2, ..., a_n \mid v_j) \cdot P(v_j)}{P(a_1, a_2, ..., a_n)}$$

$$= \underset{v_j \in V}{\operatorname{argmax}} \; P(a_1, a_2, ..., a_n \mid v_j) \cdot P(v_j)$$

Suposición "naive": los *n* atributos son independientes.

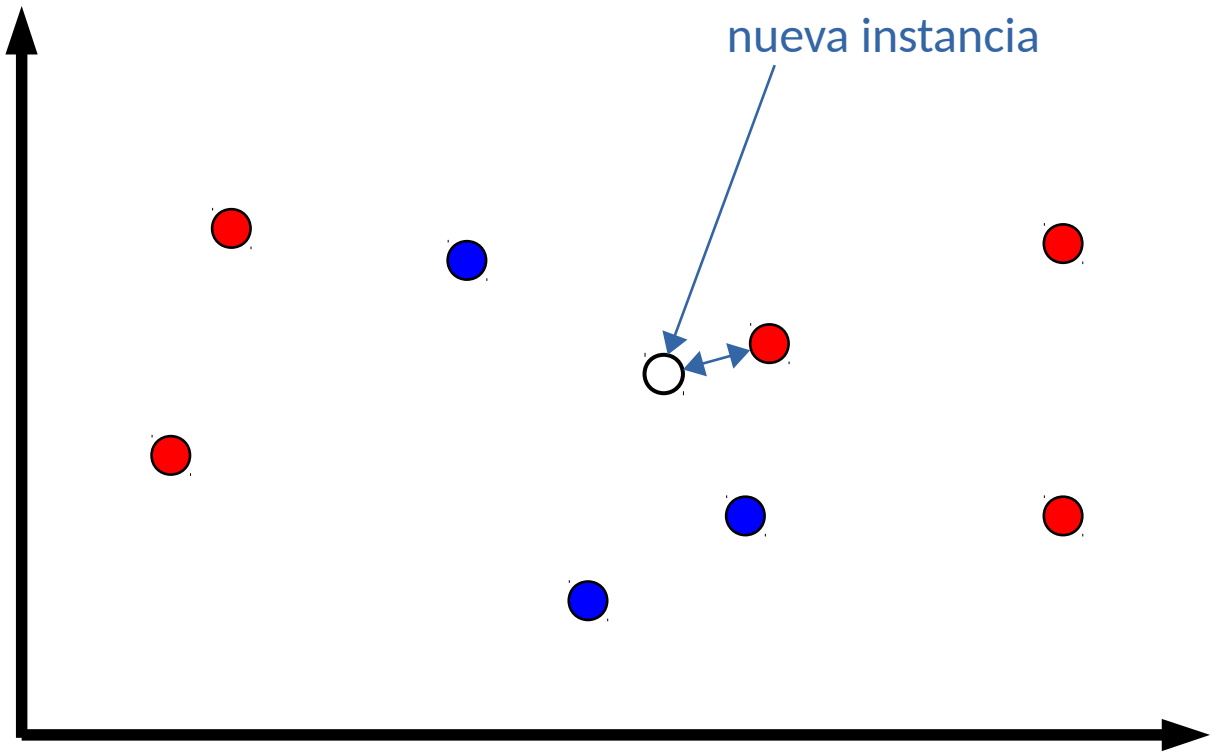
$$v_{\text{NB}} = \underset{v_j \in V}{\operatorname{argmax}} \; P(v_j) \cdot \prod_{i=i}^{n} P(a_i \mid v_j)$$

Vecinos Más Cercanos

Aprendizaje Basado en Instancias

Vecino Más Cercano:

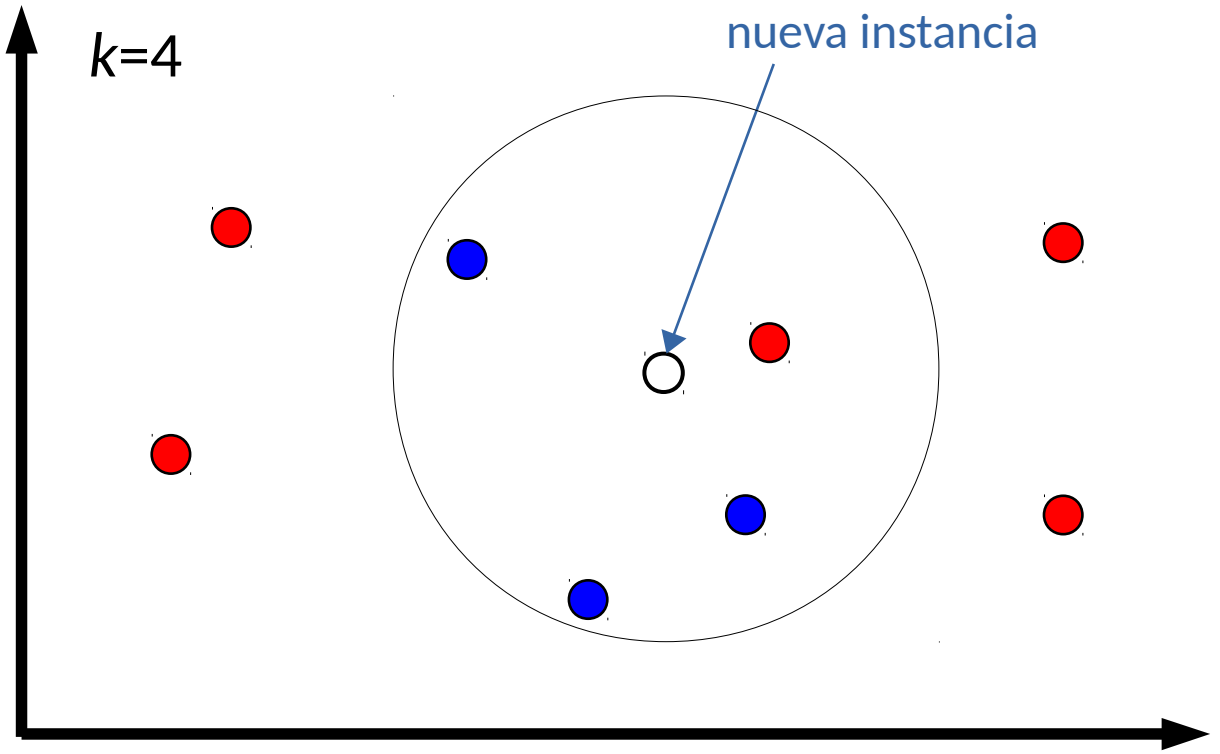
- Dada una nueva instancia , devolver la clase de la instancia más cercana en *D*.



Aprendizaje Basado en Instancias

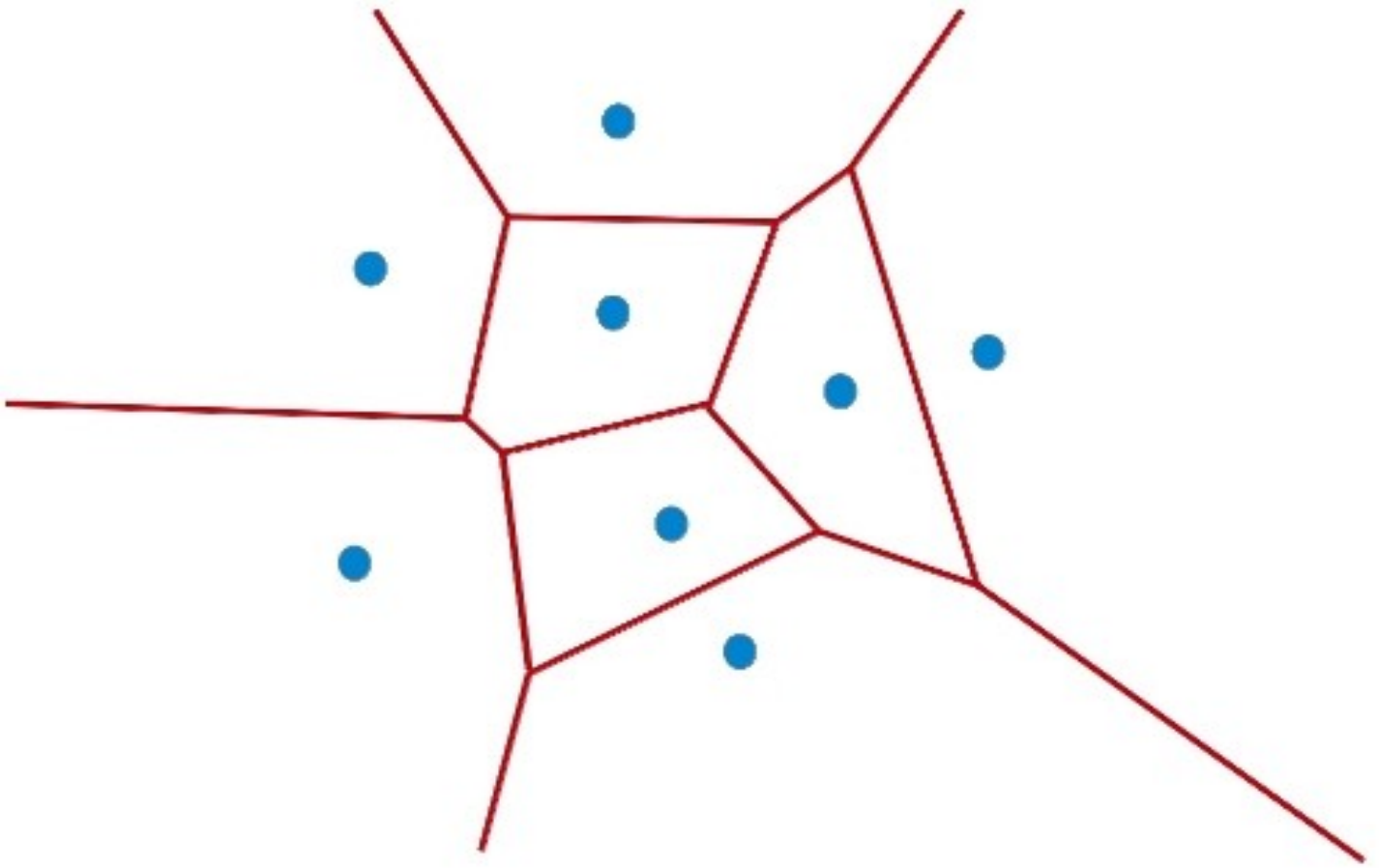
k Vecinos Más Cercanos (kNN):

- Dada una nueva instancia, devolver la clase más frecuente entre las *k* instancias más cercanas en *D*.



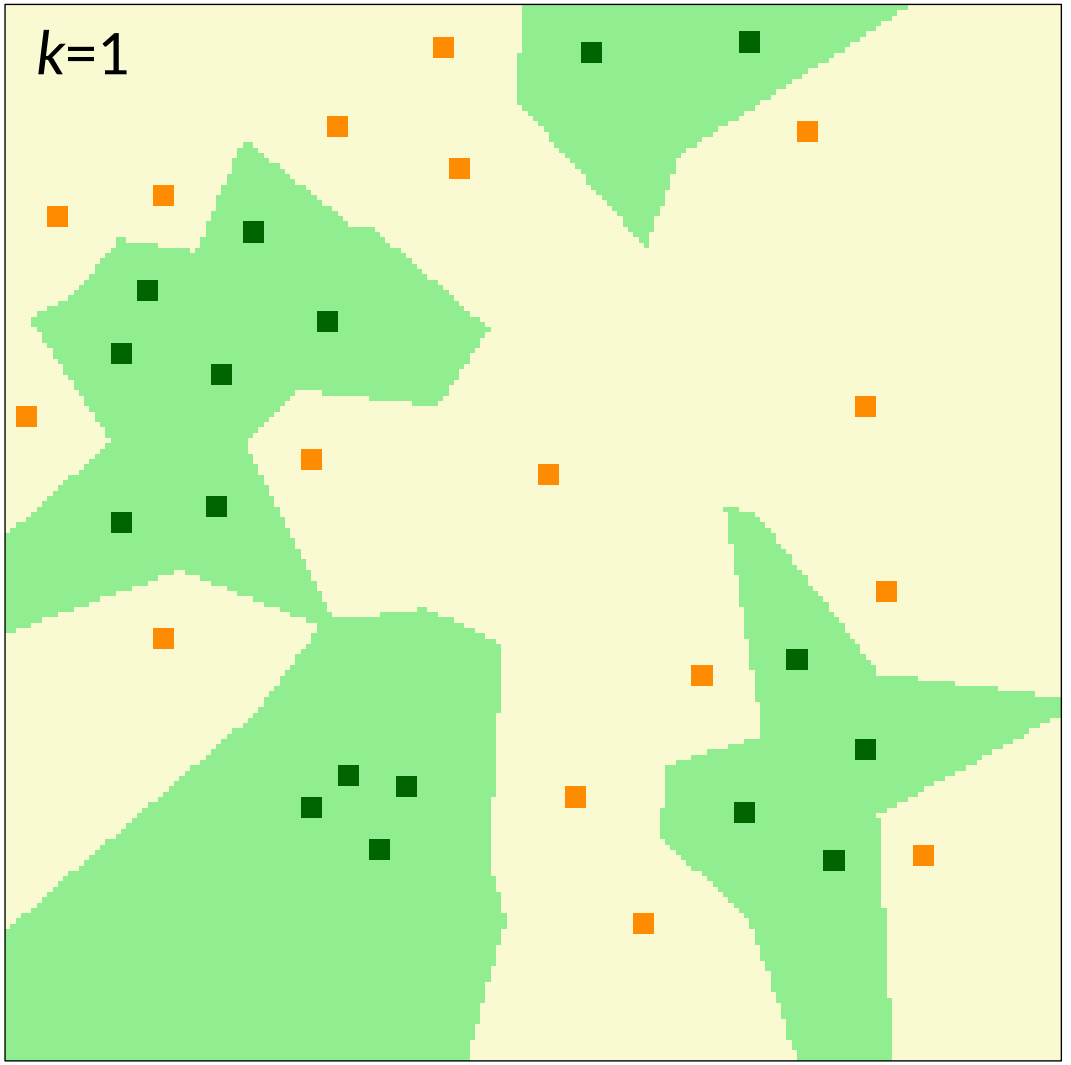
¿Qué significa un *k* más grande?

Diagrama de Voronoi:

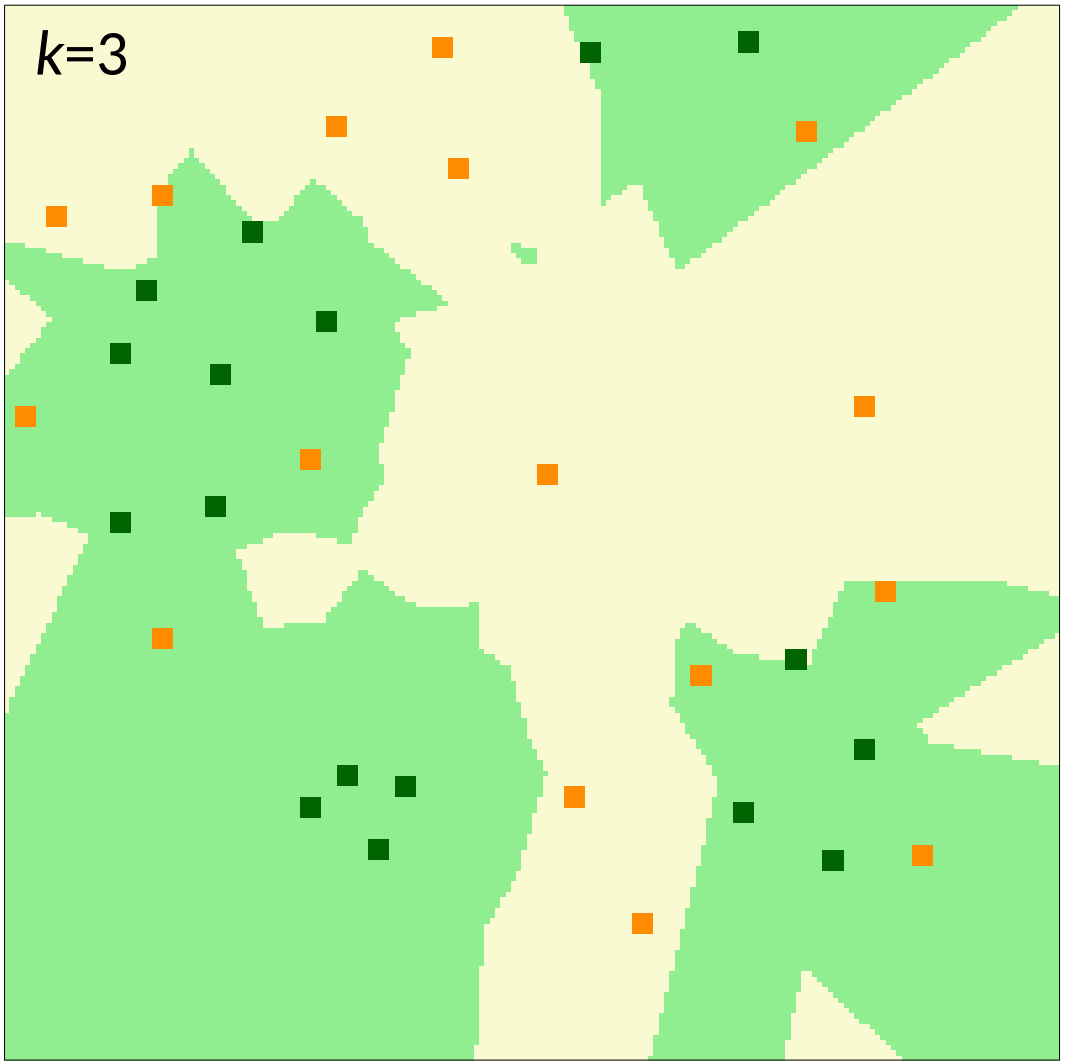


k=1

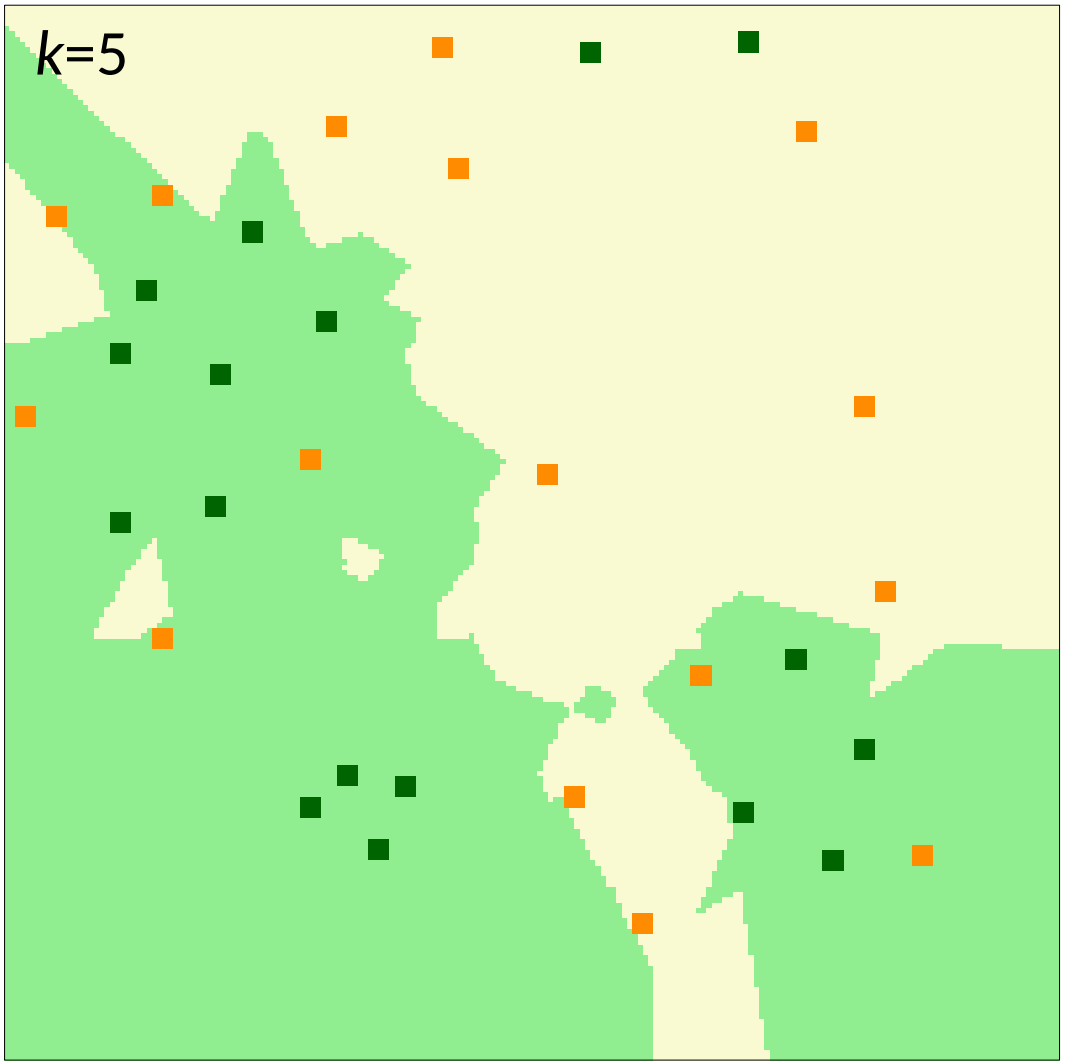
¿Qué significa un *k* más grande?



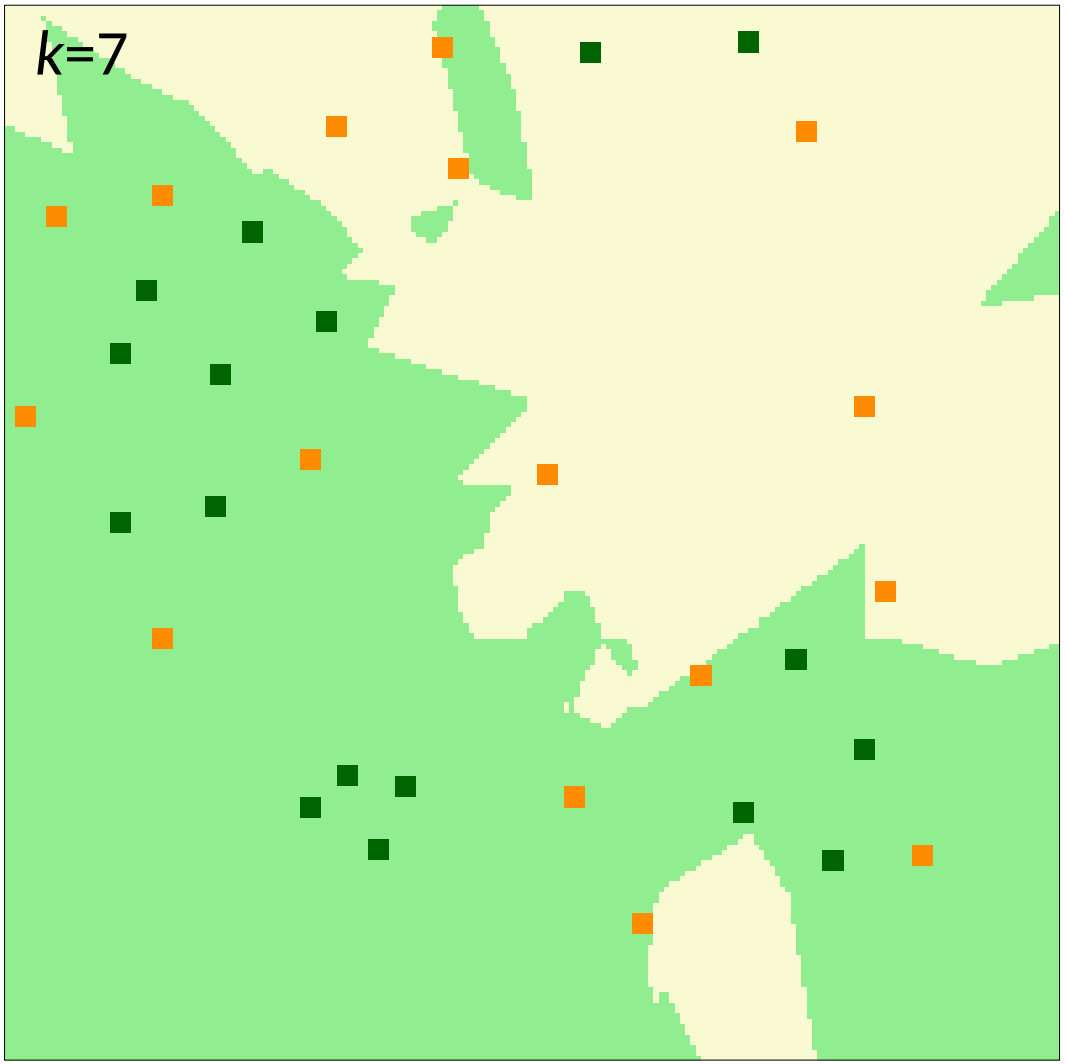
¿Qué significa un *k* más grande?



¿Qué significa un *k* más grande?



¿Qué significa un *k* más grande?



... ¿y si *k=n*?

Distance-Weighted kNN

Podríamos querer que los vecinos más cercanos tengan más influencia en la votación…

Cada vecino x_i aporta w_i votos (y no 1 como en kNN), donde

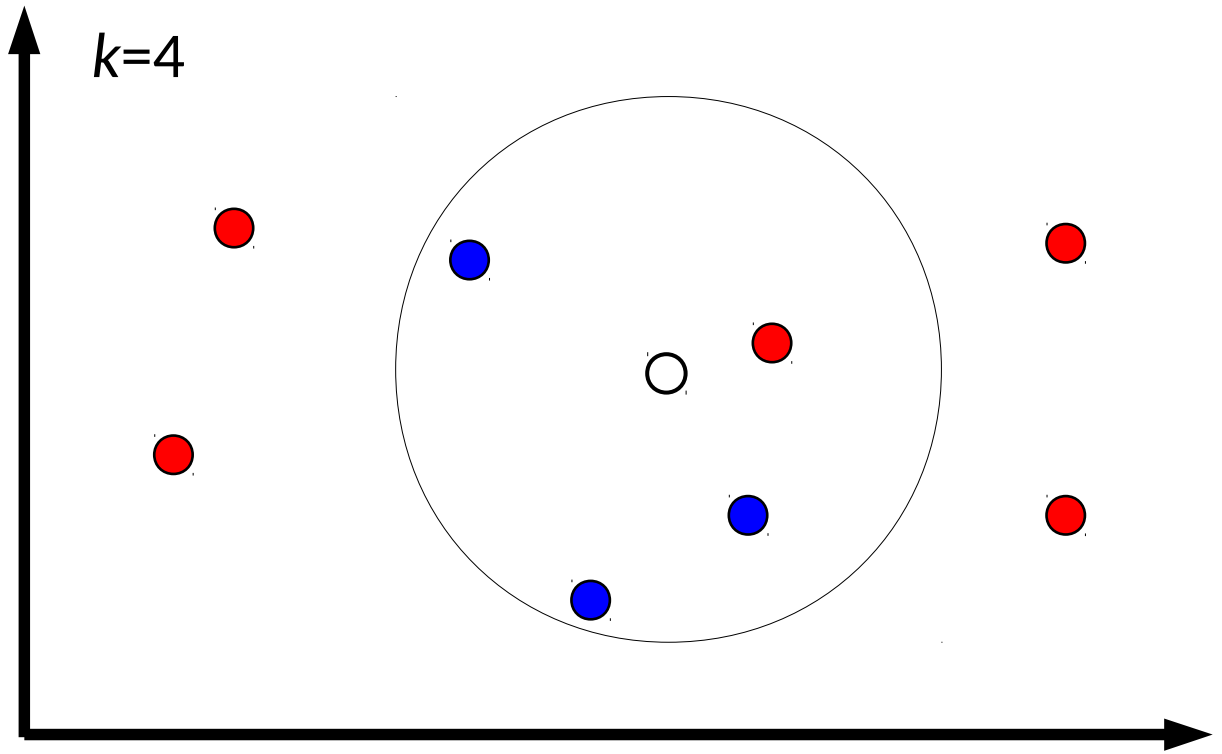
$$w_i = \frac{1}{d(x_q, x_i)^2}$$

x_q es la instancia a clasificar, y $d(\cdot,\cdot)$ es la distancia entre dos instancias.

Quizá podemos usar *todas* las instancias en *D*, en lugar de sólo las *k* más cercanas.

kNN – Devolviendo Probabilidades

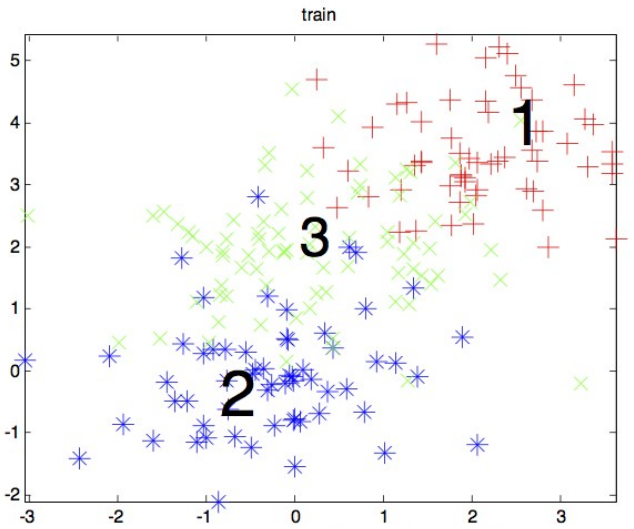
$$P(f(x_q) = y) = \sum_{x_j \in Vec(x_q, k, D)} I(f(x_j) = y) \cdot \frac{1}{k}$$

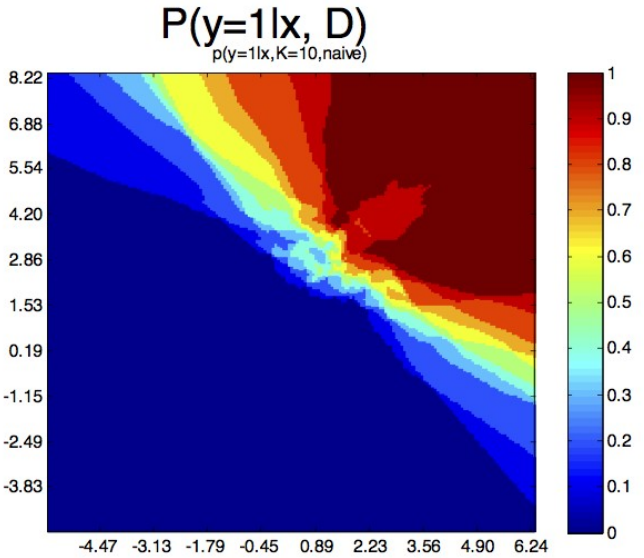


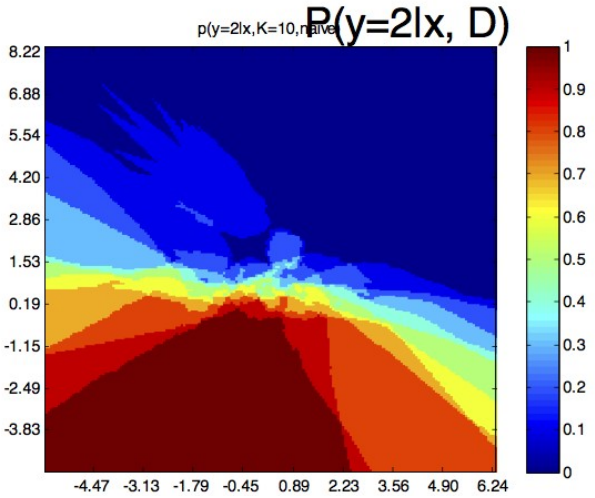
Para la nueva instancia: *P(rojo)* = ¼; *P(azul)* = ¾.

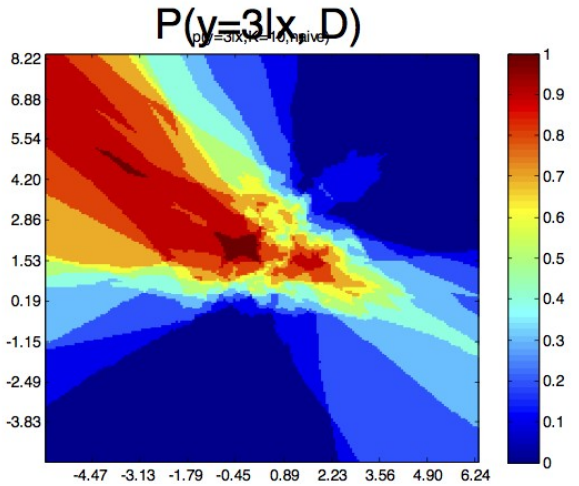
kNN – Devolviendo Probabilidades

$$P(f(x_q) = y) = \sum_{x_j \in Vec(x_q, k, D)} I(f(x_j) = y) \cdot \frac{1}{k}$$





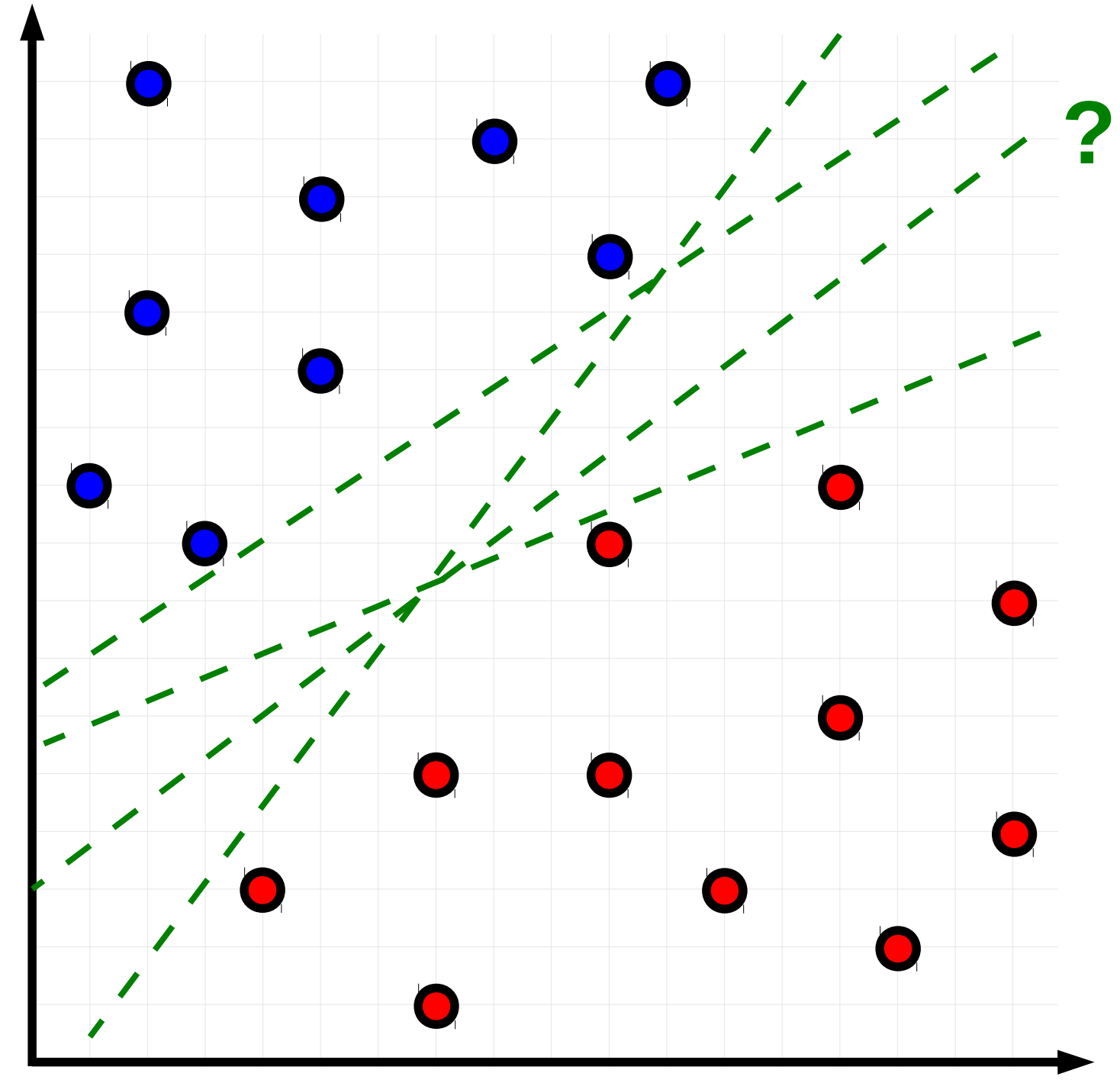


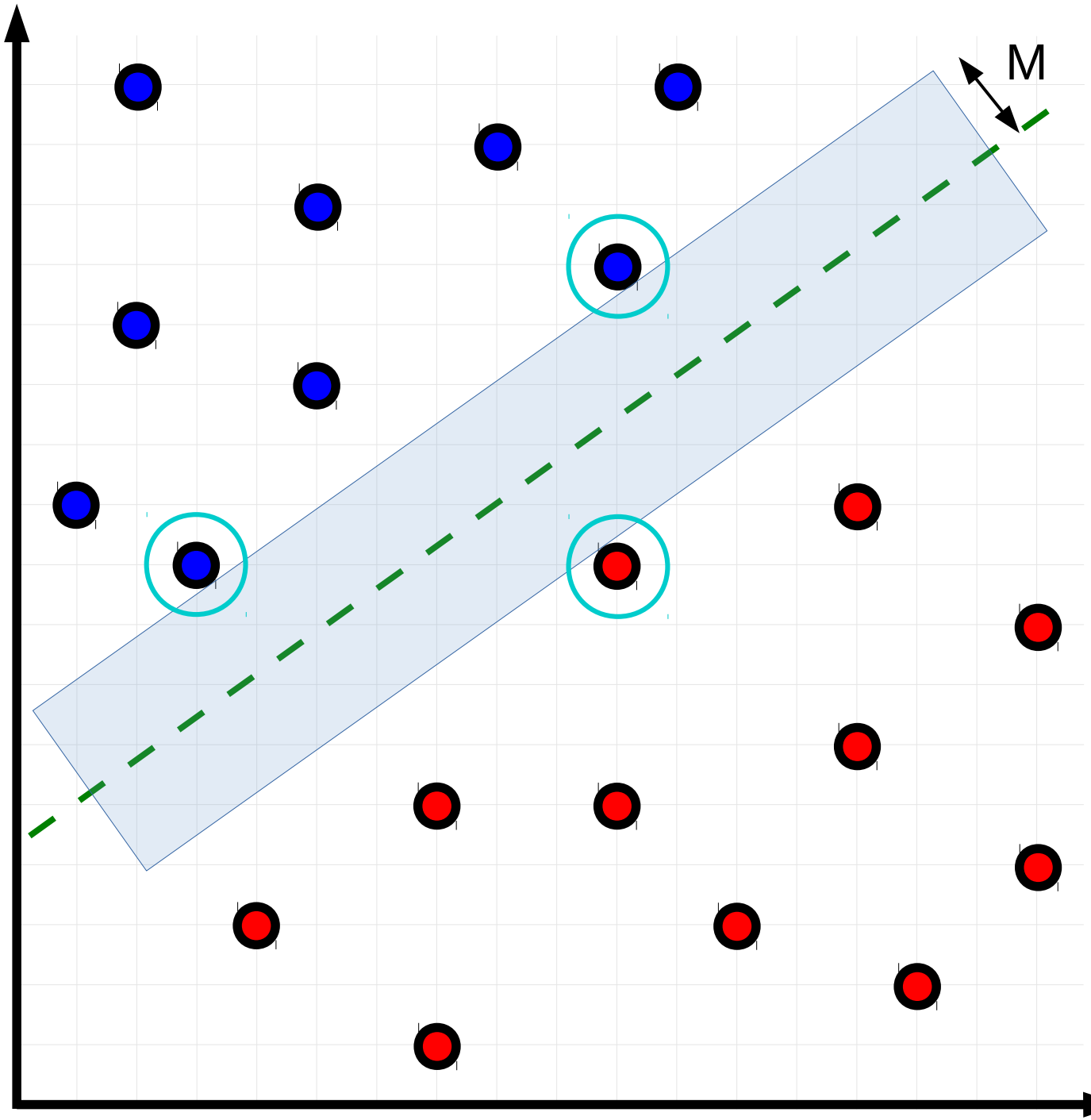


kNN

- 👍 Técnica simple que a veces permite aproximar conceptos muy complejos.
- 👍 El entrenamiento es muy rápido. (¿Entrenamiento?)
- 👎 La consulta es muy lenta. (AyED eficientes.)
- 👎 El modelo (¿modelo?) ocupa mucho espacio en disco.
- ? Para pensar: La distancia se calcula con todos los atributos. ¿Qué pasa si algunos son irrelevantes?

Support Vector Machines





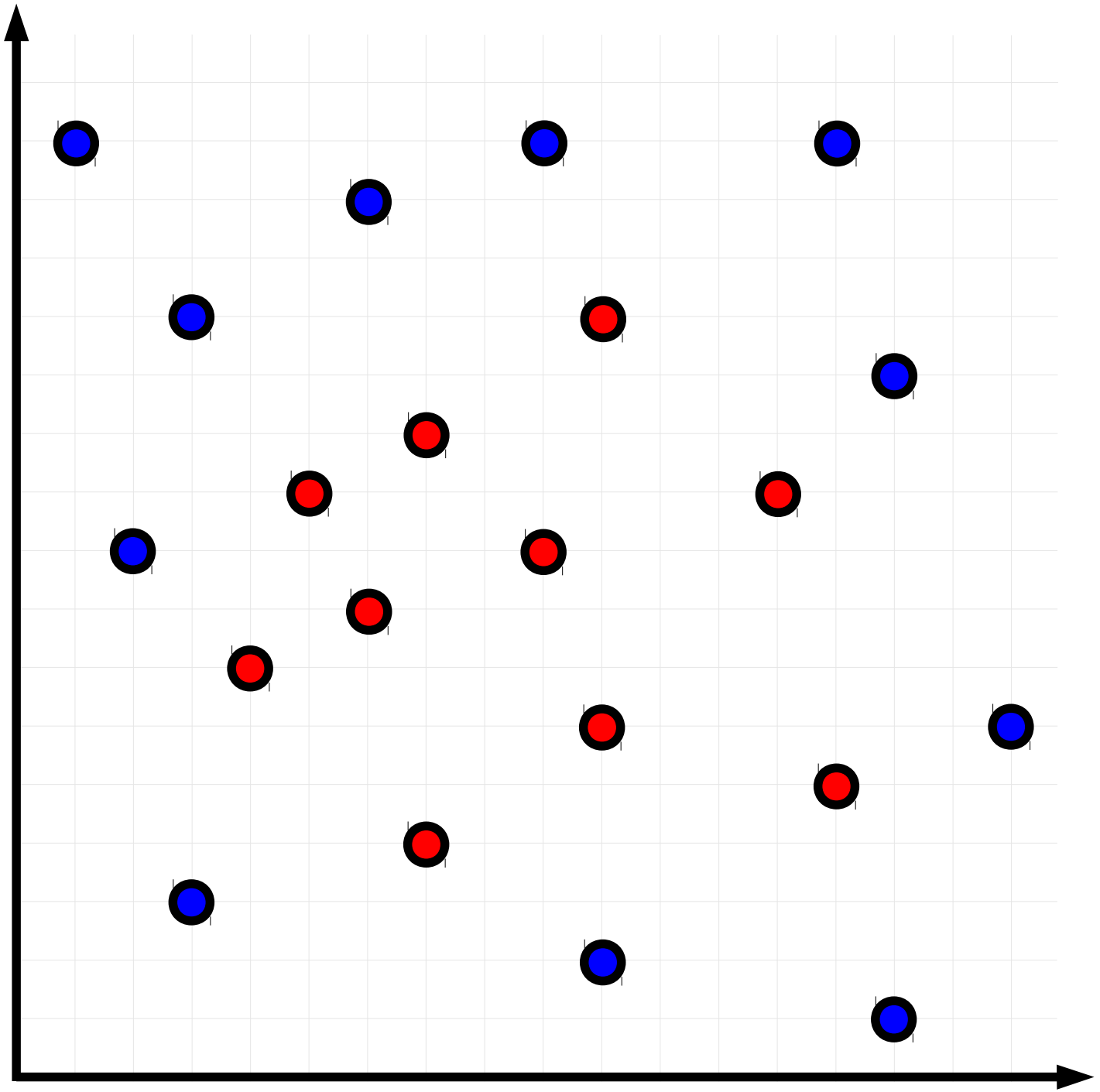
Margen M: Distancia de las instancias más cercanas a la línea de decisión (hiperplano).

Instancias más cercanas: "support vectors".

SVM: Busca maximizar M.

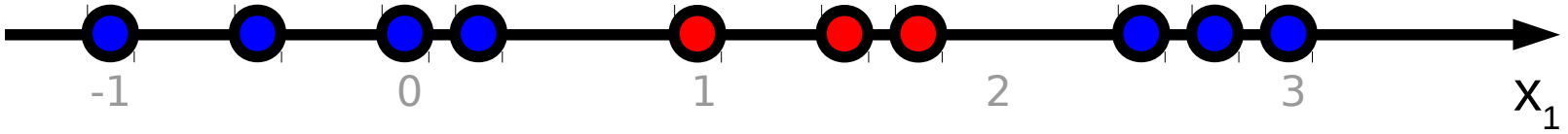
Problema de optimización.

Solución eficiente: programación cuadrática (QP).



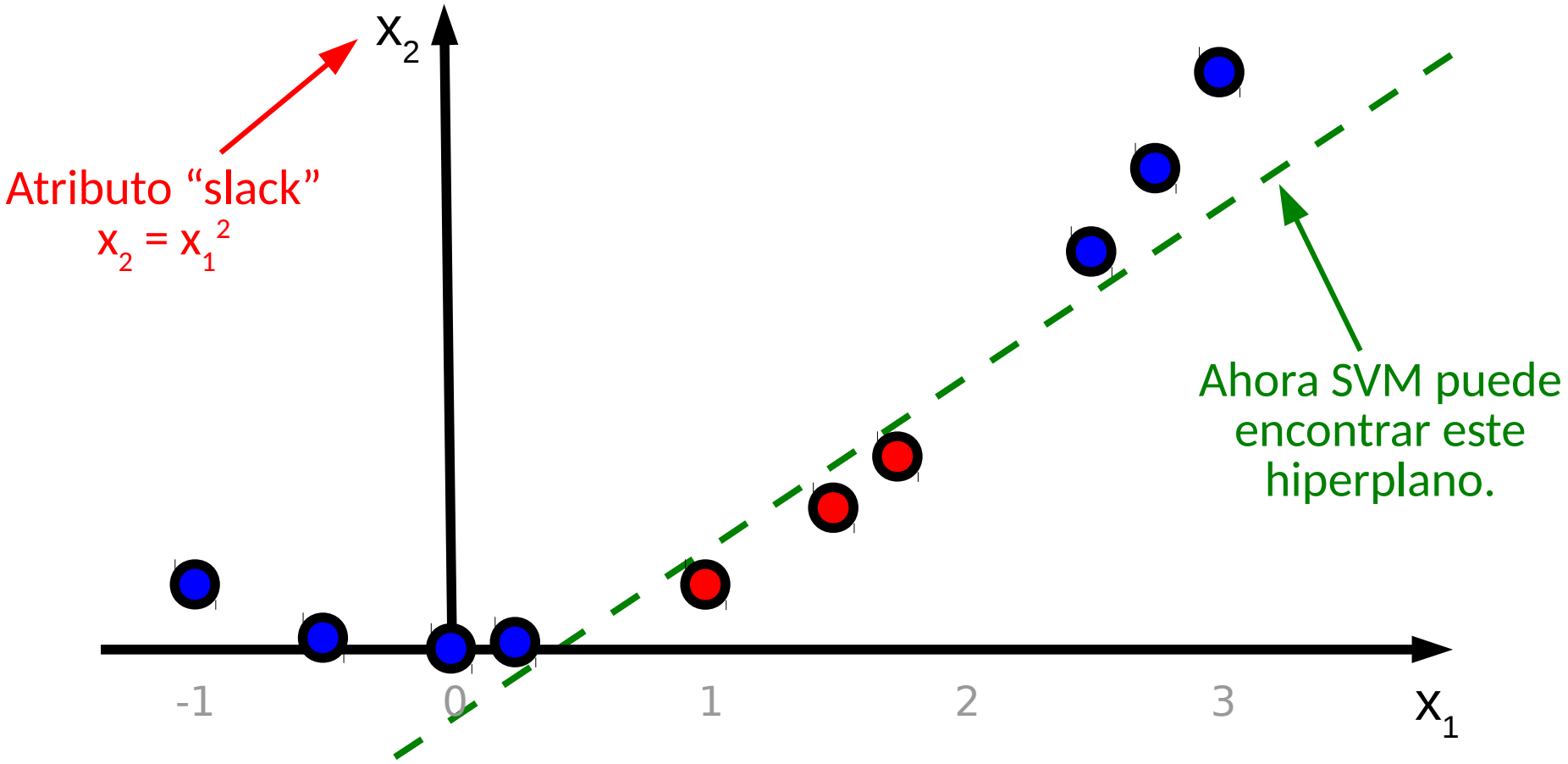
¿Qué hacemos si las instancias no son linealmente separables?

Datos originales:



Un único atributo x_1. Instancias no linealmente separables.

Datos transformados:



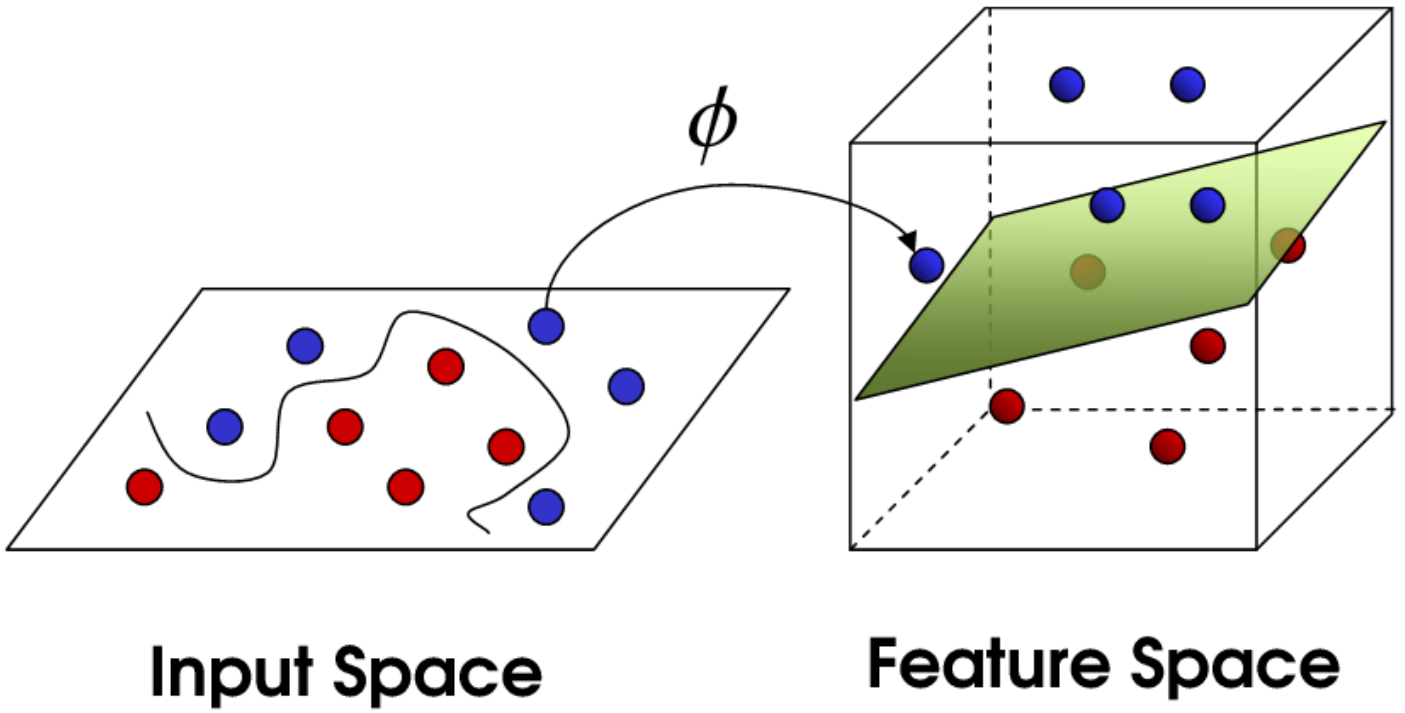
“Kernel Trick”

- Transformación de vectores de atributos. Ej: $\Phi(x_1) = (x_1, x_1^2)$
- Expandir las transf's explícitamente suele ser muy costoso.
- Lo evitamos mediante kernels.
  - Truco de álgebra lineal que nos permite operar con múltiples atributos en forma implícita:

$$K(x, y) = \langle \Phi(x), \Phi(y) \rangle$$

  - Si un algoritmo (ej. SVM) puede expresarse en términos de productos internos entre vectores, reemplazamos las apariciones de <x, y> por K(x, y).
  - Así, ejecutamos SVM implícitamente en dimensiones superiores.

Support Vector Machines



- Kernels más usados: lineal, polinomial, sigmoideo, RBF.
  - Souza, C. R. "Kernel Functions for Machine Learning Applications"

Support Vector Machines

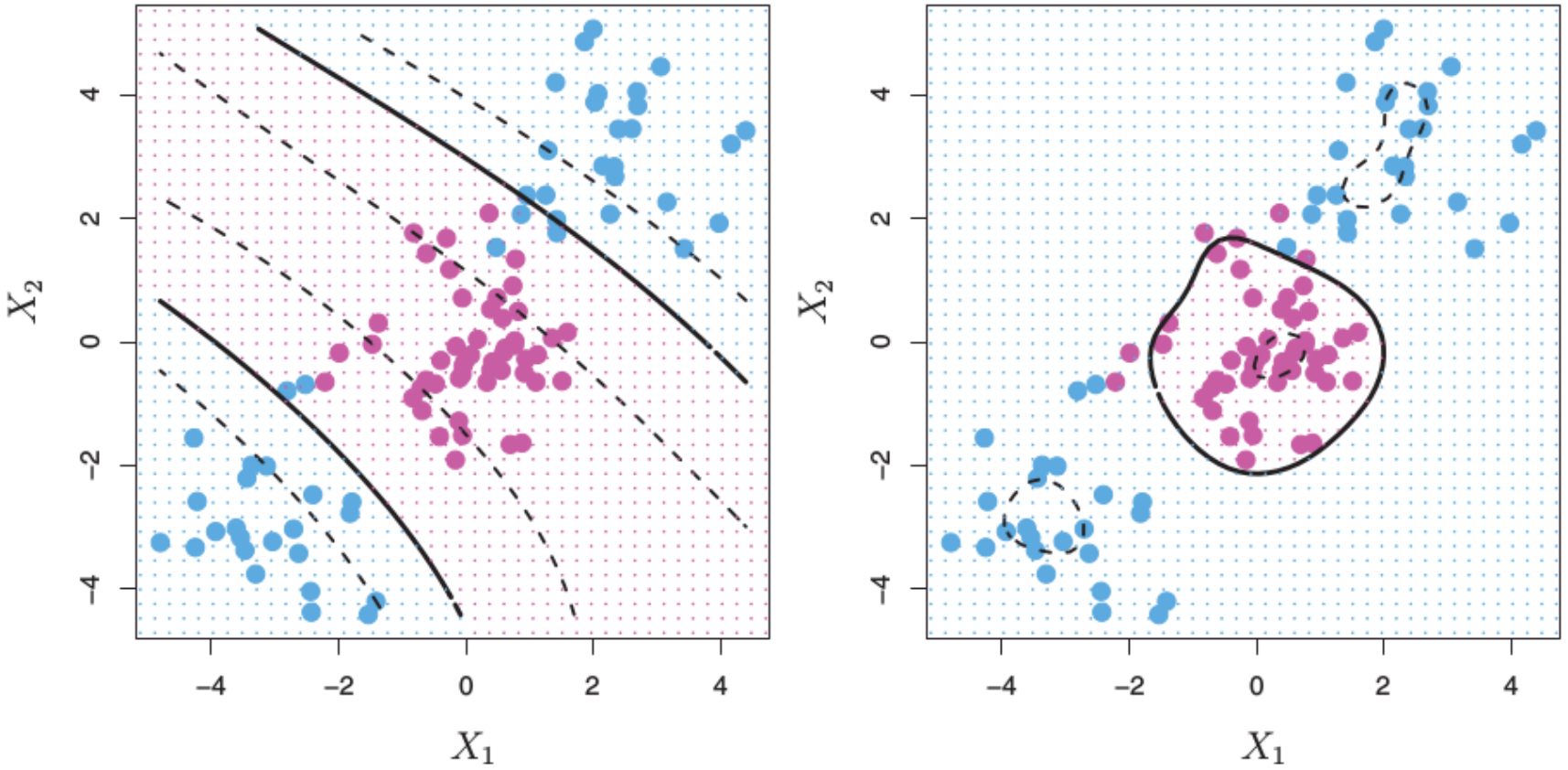


FIGURE 9.9. Left: *An SVM with a polynomial kernel of degree 3 is applied to the non-linear data from Figure 9.8, resulting in a far more appropriate decision rule.* Right: *An SVM with a radial kernel is applied. In this example, either kernel is capable of capturing the decision boundary.*

Fuente: G. James et al., "An Introduction to Statistical Learning with Applications in R", Springer, 6th edition, 2015.

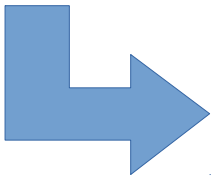
Support Vector Machines

- Complejidad computacional:
  - Entrenamiento costoso; consulta eficiente.
- ¿Espacio de hipótesis?
- ¿Sesgo inductivo?

(a) L. Bottou, C.J. Lin. "Support vector machine solvers." Large scale kernel machines, pp. 301-320, 2007.

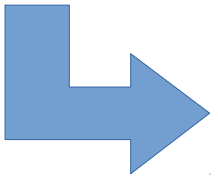
SVM – Atributos Categóricos

- EstadoCivil: {Soltero, Casado, Viudo, Divorciado, Otro}



EstadoCivil_Soltero: {0, 1}
EstadoCivil_Casado: {0, 1}
EstadoCivil_Viudo: {0, 1}
EstadoCivil_Divorciado: {0, 1}
EstadoCivil_Otro: {0, 1}

- Palabras: BagOfWords



Palabras_hola: $\mathbb{N}$
Palabras_mundo: $\mathbb{N}$
Palabras_la: $\mathbb{N}$
Palabras_y: $\mathbb{N}$
Palabras_cuando: $\mathbb{N}$
…

SVM – Clases múltiples

- Hasta ahora: clasificación binaria.
- *N* clases:
  - Para cada clase C_i, entrenar un SVM para discriminar C_i del resto (clasificación *OVA*: *one-versus-all*).
  - Para una nueva instancia, correr los *N* clasificadores y retornar la clase con mayor margen (i.e., con mayor confianza).

Repaso

- Clasificadores:
  - Naive Bayes
  - K Vecinos más Cercanos (KNN)
  - Support Vector Machines (SVM)

- Próximo tema: Conjuntos de Clasificadores.