

# Minería de Datos

## IIC2433

KNN

Vicente Domínguez

# ¿Qué veremos esta clase?

- Otra forma de clasificar: KNN

# KNN

## *k* Nearest Neighbors (*k* vecinos cercanos)

- *k* es un número natural
  - $k = 1, 2, 3, 4, \dots$  etc
- Si  $k = 1$ , hablamos de 1NN (1 vecino cercano)
- Si  $k = 2$ , hablamos de 2NN (2 vecinos cercano)

etc...

# Métricas de distancia

## *Distancia Euclidiana (la más conocida)*

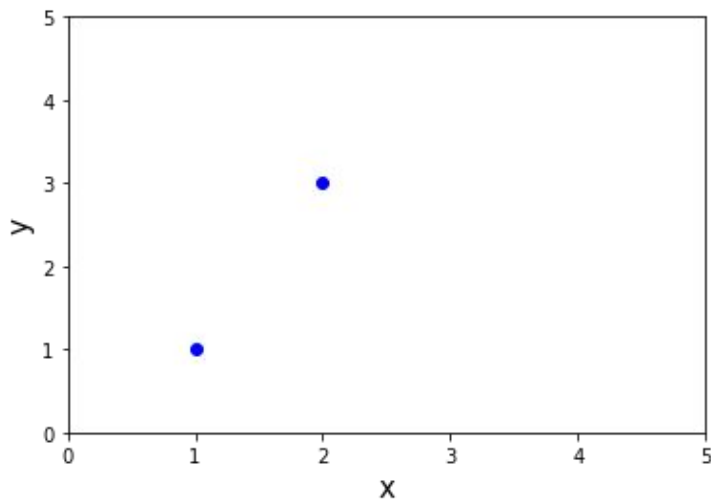
“La **distancia euclidiana** entre dos puntos es longitud del segmento que los une”

$$\begin{aligned}d(\mathbf{p}, \mathbf{q}) &= d(\mathbf{q}, \mathbf{p}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2} \\&= \sqrt{\sum_{i=1}^n (q_i - p_i)^2}.\end{aligned}$$

# Distancia Euclidiana

## *Ejemplos*

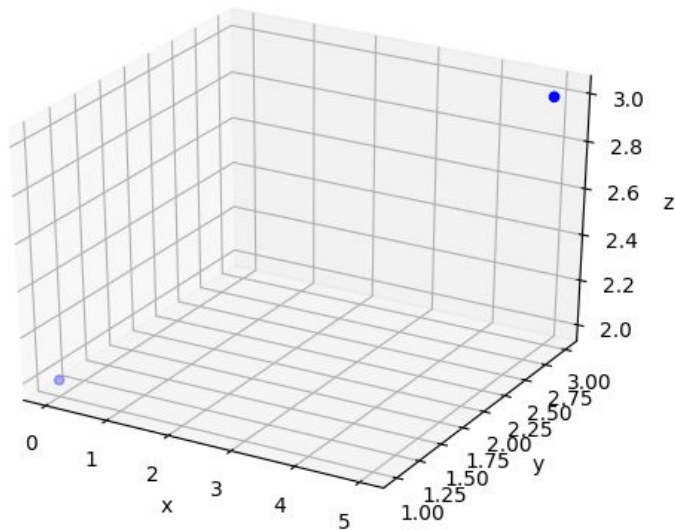
Calcular la distancia euclidiana entre los puntos  $(1, 1)$  y  $(2, 3)$



# Distancia Euclidiana

## *Ejemplos*

Calcular la distancia euclidiana entre los puntos  $(0, 1, 2)$  y  $(5, 3, 3)$



# Distancia Euclidiana

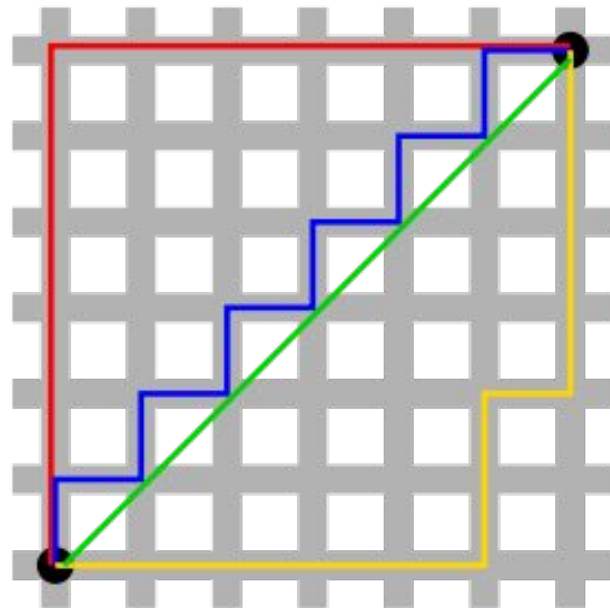
## *Ejemplos*

Calcular la distancia euclidiana entre los puntos  $(0, 1, 1, 1)$  y  $(1, 2, 2, 2)$

# Métricas de distancia

## *Distancia Manhattan*

“La **distancia manhattan** entre dos puntos es longitud del camino entre ellos dando pasos estrictamente verticales u horizontales”





# Otras métricas de distancia

## *Mahalanobis*

$$D_M(\vec{x}) = \sqrt{(\vec{x} - \vec{\mu})^T S^{-1} (\vec{x} - \vec{\mu})}.$$

## *Minkowski*

$$\left( \sum_{i=1}^n |x_i - y_i|^p \right)^{1/p}$$

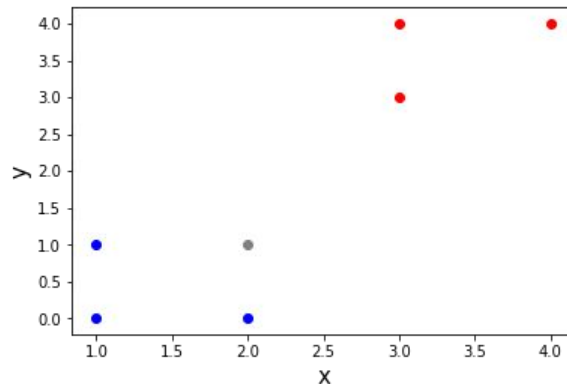
# 1NN (1 vecino más cercano)

$$K = 1$$

¿Cuál es la clase del punto (2, 1) ? ¿Rojo o azul?

Le asignaremos la clase de su vecino más cercano.

Para saber cuál es el vecino más cercano el computador debe calcular su distancia frente a todos los puntos del dataset.



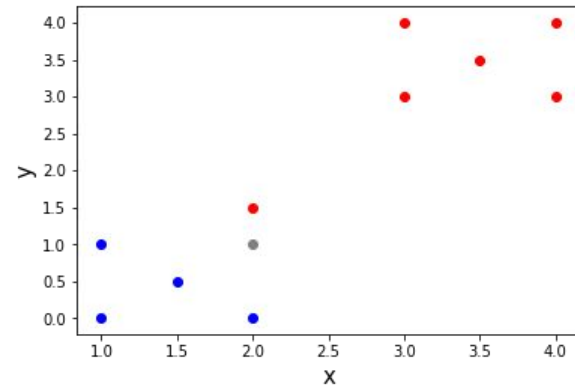
# 3NN (3 vecinos más cercanos)

$$K = 3$$

¿Cuál es la clase del punto (2, 1) ? ¿Rojo o azul?

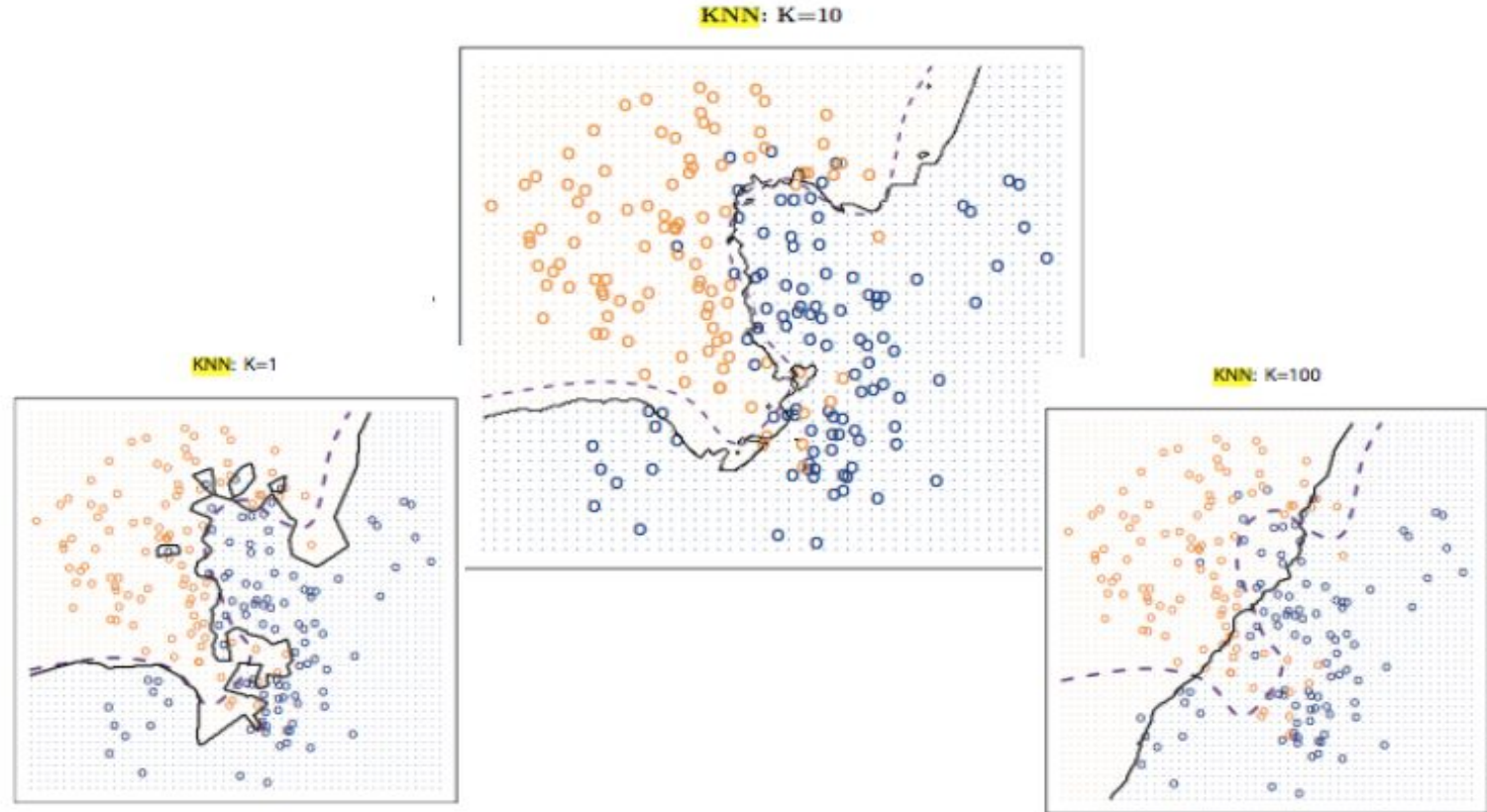
Tomaremos la clase de sus tres vecinos más cercanos y le asignaremos la clase más

Para saber cuál es el vecino más cercano el computador debe calcular su distancia frente a todos los puntos del dataset.



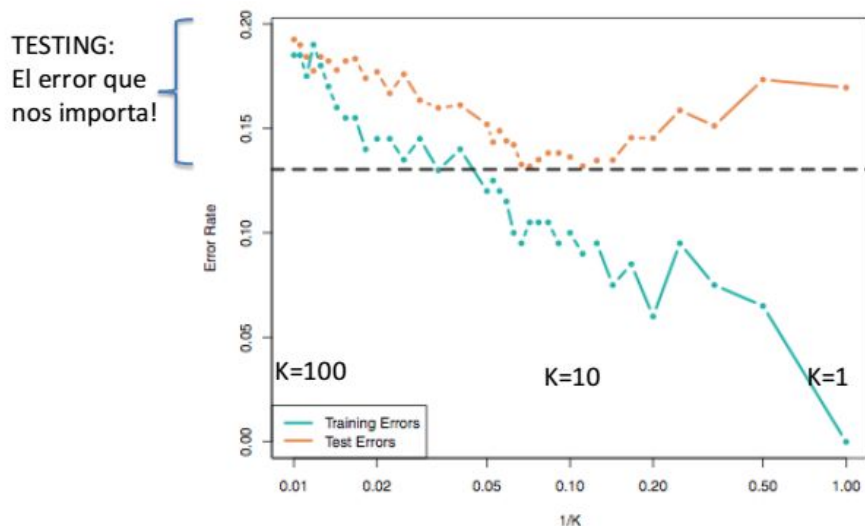
# KNN

*¿Qué es mejor un valor de  $k$  alto o bajo?*



# KNN

*¿Qué es mejor un valor de  $k$  alto o bajo?*



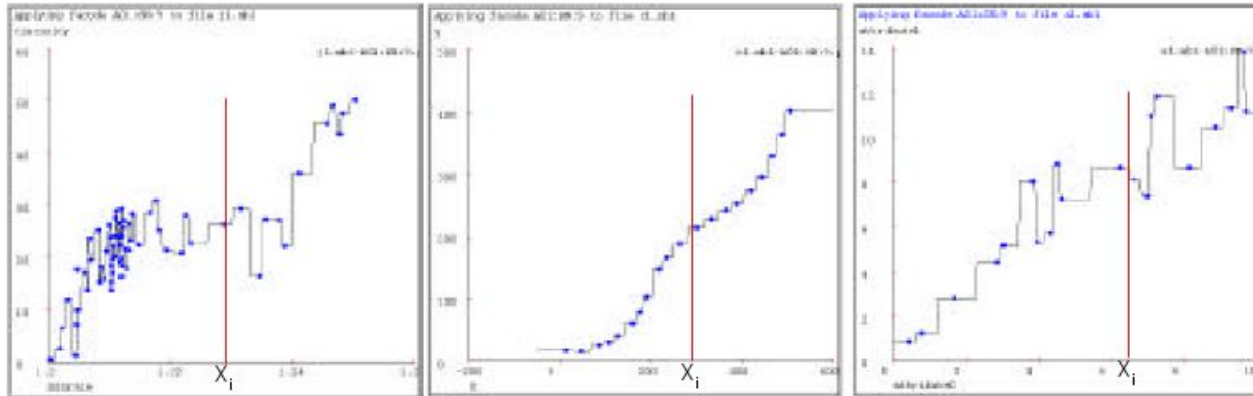
**FIGURE 2.17.** The **KNN** training error rate (blue, 200 observations) and test error rate (orange, 5,000 observations) on the data from Figure 2.13, as the level of flexibility (assessed using  $1/K$ ) increases, or equivalently as the number of neighbors  $K$  decreases. The black dashed line indicates the Bayes error rate. The jumpiness of the curves is due to the small size of the training data set.

# Regresiones usando KNN

$$K = 1$$

¿Cuál es el valor de  $y$ , dado un valor de  $x$ ?

Usamos el valor de  $y$  del punto más cercano en  $x$



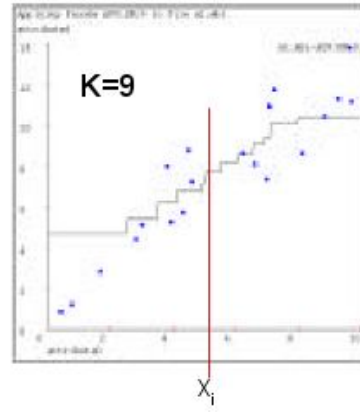
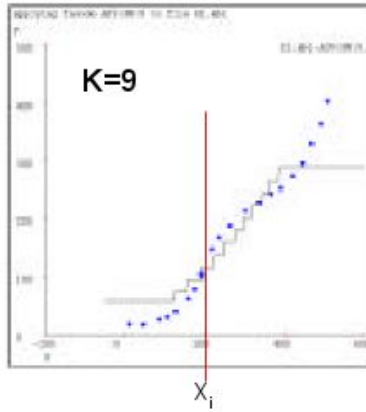
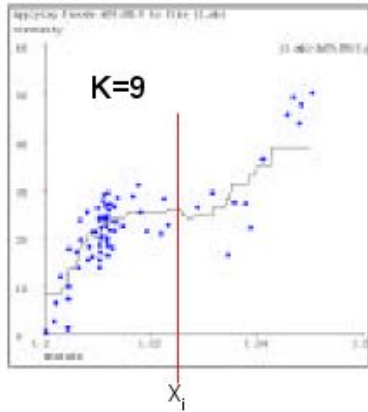
# Regresiones usando KNN

$$K = 9$$

¿Cuál es el valor de  $y$ , dado un valor de  $x$ ?

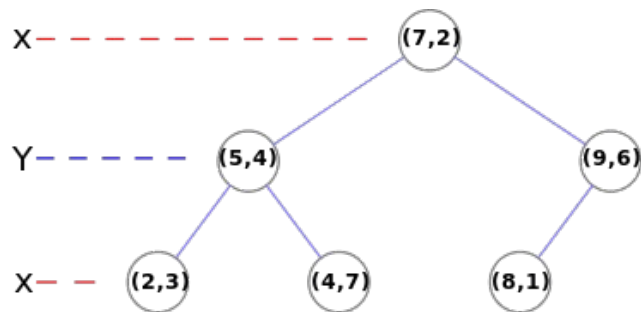
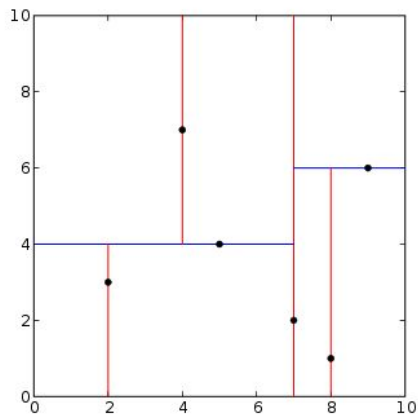
Usamos el el promedio de los valores  $y$ , de los 9 puntos con  $x$  más cercano.

(Podemos usar promedio simple o ponderado)



# KD-Tree

*Permite encontrar los  $k$ -vecinos cercanos sin calcular las distancias hacia todos los datos.*





# Trabajar con atributos nominales

Es necesario transformar los datos nominales a numéricos

## Nominal a entero

Id	Pais
0	Rusia
1	Alemania
2	Chile
3	Argentina
4	Chile

## Nominal a One-hot Encoding

Id	Alemania	Argentina	Chile	Rusia
0	0	0	0	1
1	1	0	0	0
2	0	0	1	0
3	0	1	0	0
4	0	0	1	0