APRENDIZAJE BASADO EN INSTANCIAS

- *Instance-Based Learning* (IBL)
- La idea es crear un clasificador 'perezoso' (*lazy*)
- Para clasificar una nueva instancia, utilizo las que más se le parecen de las que ya conozco.
- Fases:
  1. Entrenamiento: No se entrena, sino que se almacena todo el conjunto de datos disponibles
     - No se realiza ningún cómputo
  2. Generalización: dado un nuevo dato, se extraen de memoria un conjunto de datos similares, que son utilizados para clasificar el nuevo dato
     - Aquí es donde se realiza todo el cómputo

Aprendizaje basado en Instancias

- ¿Cómo definimos el concepto de similitud?
- ¿Coste de clasificación?
  - ¡Todo el cómputo se realiza en tiempo de clasificación!
    - Aprendizaje Perezoso (*Lazy Learning*)
- Ventajas:
  - Para cada nueva instancia puedo obtener un clasificador diferente.
  - La descripción de las instancias puede ser tan compleja como quiera.
- Desventajas:
  - El costo de clasificación puede ser alto.
  - Atrib. irrelevantes pueden afectar la medida de similitud.

APRENDIZAJE BASADO EN INSTANCIAS

- k-NN (*k-Nearest Neighbour*)
  - Un nuevo caso se va a clasificar en la clase más frecuente a la que pertenecen sus k vecinos más cercanos
    - También para realizar regresión
  - Idea muy simple e intuitiva
  - Fácil implementación
  - No hay un modelo explícito

APRENDIZAJE BASADO EN INSTANCIAS

- k-NN (*k-Nearest Neighbour*)
  - Todas las instancias corresponden con puntos en un espacio de dimensión n (R^n)
    - Cada instancia está caracterizada por n valores
      - Por tanto, cada instancia es un punto con n coordenadas
  - Se puede calcular la distancia entre dos instancias por medio de la distancia euclídea:

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^{n} (x_i[r] - x_j[r])^2}$$

APRENDIZAJE BASADO EN INSTANCIAS

- k-NN (*k-Nearest Neighbour*)
  - Se puede calcular la distancia entre dos instancias por medio de la distancia euclídea:

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^{n} (x_i[r] - x_j[r])^2}$$

    - Si los atributos tienen diferentes rangos, se normaliza
      - Paso crucial
      - ¿Qué ocurriría si no se normalizase?
    - Si los atributos tienen valores simbólicos, ¿se numeran?
    - Se calcula así la distancia de una instancia al resto

APRENDIZAJE BASADO EN INSTANCIAS

- k-NN (*k-Nearest Neighbour*)
  - Otras medidas de distancia:
    - Manhattan: $\sum_{i=1}^{k} |x_i - y_i|$
    - Minkowski: $\left(\sum_{i=1}^{k} (|x_i - y_i|)^q \right)^{1/q}$
    - Coseno:
      - 1 – (coseno del ángulo entre los dos puntos)
        - Se toman los puntos como vectores
    - Estas (junto con la euclídea) son válidas sólo para variables continuas

APRENDIZAJE BASADO EN INSTANCIAS

- k-NN (*k-Nearest Neighbour*)
  - Otras medidas de distancia:
    - Para variables categóricas
      - Distancia de Hamming:

Hamming Distance

$$D_H = \sum_{i=1}^{k} |x_i - y_i|$$

$$x = y \Rightarrow D = 0$$

$$x \neq y \Rightarrow D = 1$$

      - También utilizada cuando las instancias son datos binarios

X	Y	Distance
Male	Male	0
Male	Female	1

APRENDIZAJE BASADO EN INSTANCIAS

- k-NN (*k-Nearest Neighbour*)
  - Otras medidas de distancia:
    - Correlación
      - 1 – (correlación entre las instancias)
        - En este caso, la lista de valores de variables que conforma una instancia se toma como una secuencia de valores

APRENDIZAJE BASADO EN INSTANCIAS

- El parámetro k identifica cuántos vecinos se utilizan para la decisión
- Función objetivo
  - Dada una instancia x, y sus k vecinos: $x_1, \ldots, x_k$
    - Caso discreto: Clasificación:

      Se asigna el valor más común de entre los k más cercanos:

      f: $R^n \rightarrow V$, $V = \{v_1,\ldots,v_s\}$

$$\hat{f}(x) = \underset{v \in V}{\text{argmáx}} \sum_{i=1}^{k} \delta(v, f(x_i))$$

      donde $\delta(a,b)=1$ si a=b, y $\delta(a,b)=0$ en cualquier otro caso

      - Es decir, aquella etiqueta con la que haya mayor coincidencia:

Clase v tal que $\sum_{i=1}^{k} \delta(v, f(x_i))$ sea la más alta

APRENDIZAJE BASADO EN INSTANCIAS

- El parámetro k identifica cuántos vecinos se utilizan para la decisión
- Función objetivo
  - Dada una instancia x, y sus k vecinos: $x_1, \ldots, x_k$
    - Caso continuo: Regresión:

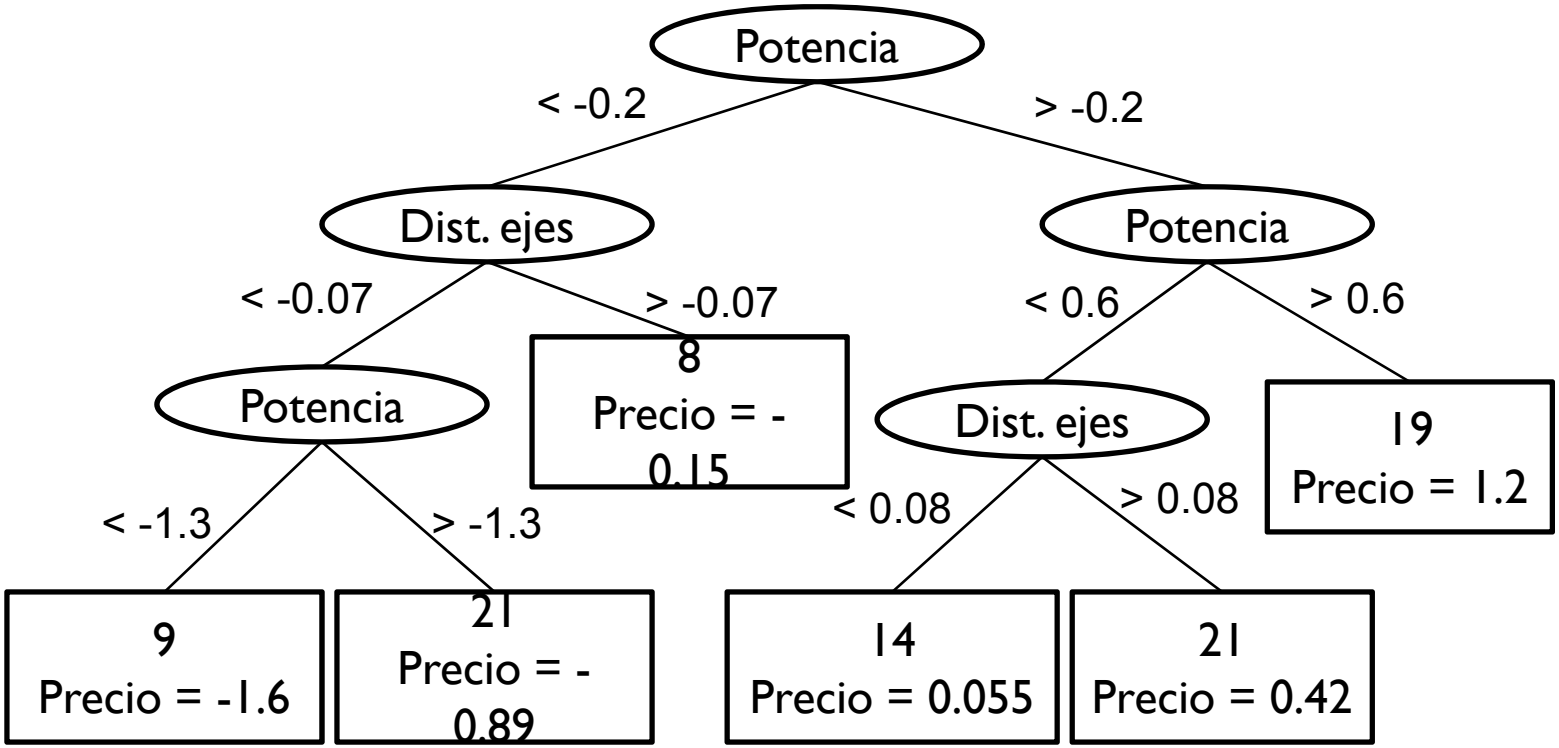
      Se asigna el valor medio de entre los k más cercanos:

      f: $R^n \rightarrow R$

$$\hat{f}(x) = \frac{\sum_{i=1}^{k} f(x_i)}{k}$$

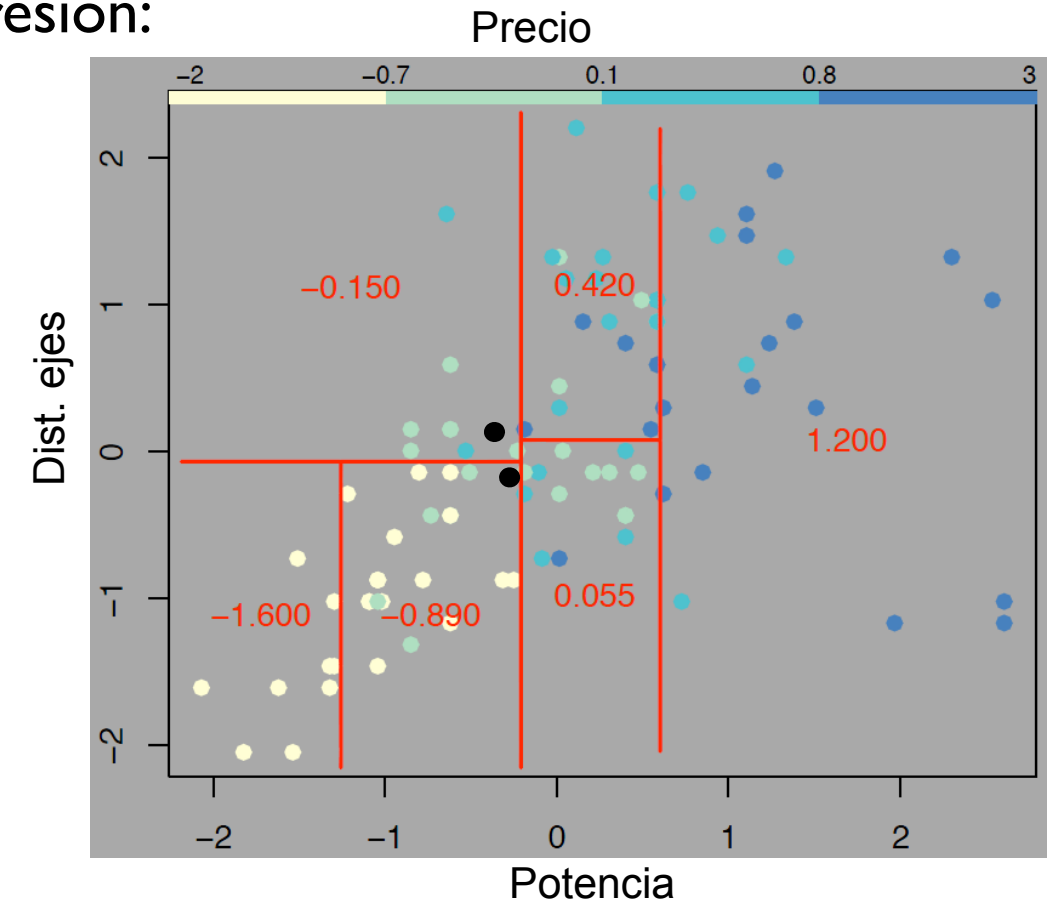
APRENDIZAJE BASADO EN INSTANCIAS

- Ejemplo: caso continuo:
  - Precios de coches a partir de potencia y dist. ejes:
  - Árbol de regresión:



APRENDIZAJE BASADO EN INSTANCIAS

- Ejemplo: caso continuo:
  - Precios de coches a partir de potencia y dist. ejes:
  - Árbol de regresión:



Nuevas instancias:

Valores resultantes:
-0.150 y -0.89

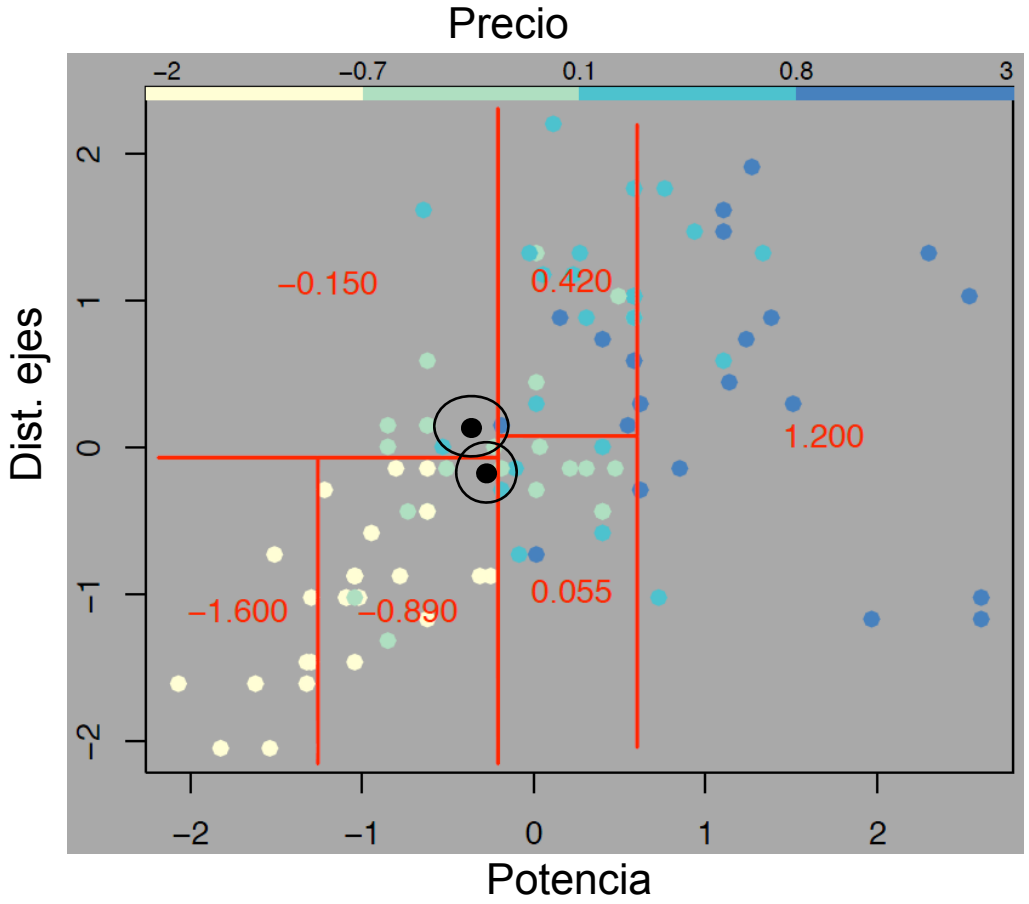
Discontinuidad muy grande

APRENDIZAJE BASADO EN INSTANCIAS

- Ejemplo: caso continuo:
  - Precios de coches a partir de potencia y dist. ejes:
  - kNN (k=3):

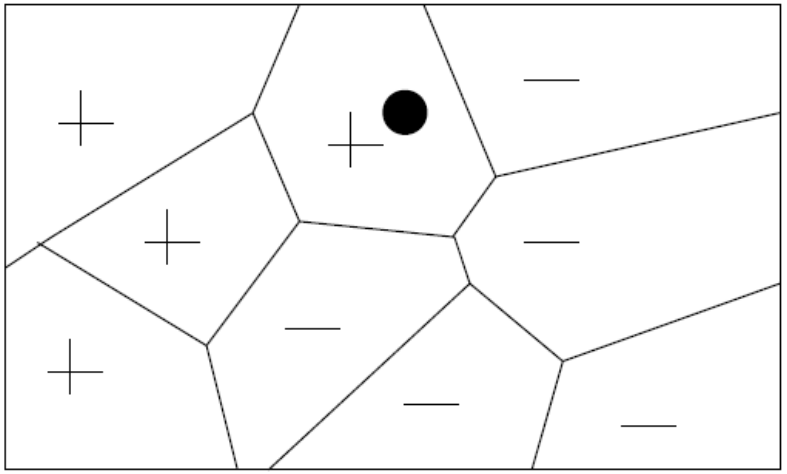
Nuevas instancias:

Valores resultantes:
Media de los valores de las 3 instancias más cercanas
Valor con menos discontinuidad

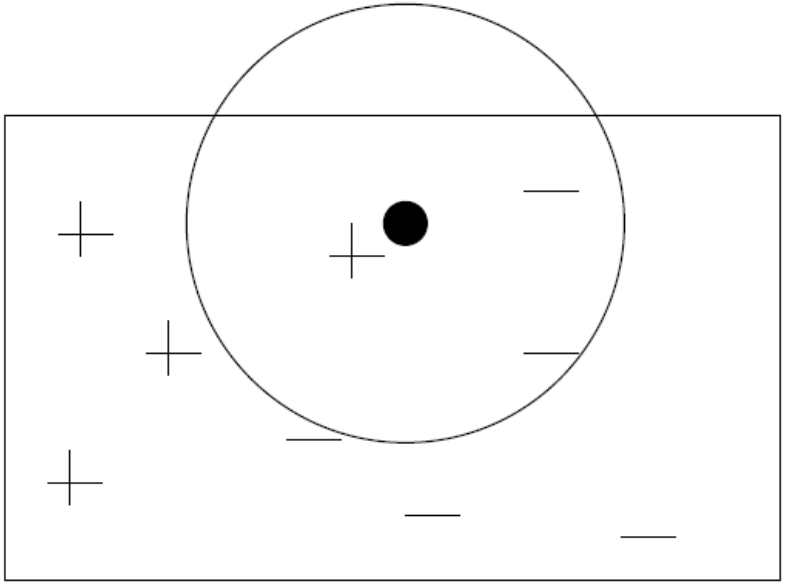


Aprendizaje basado en Instancias

- El parámetro k identifica cuántos vecinos se utilizan para la decisión



1-NN



3-NN

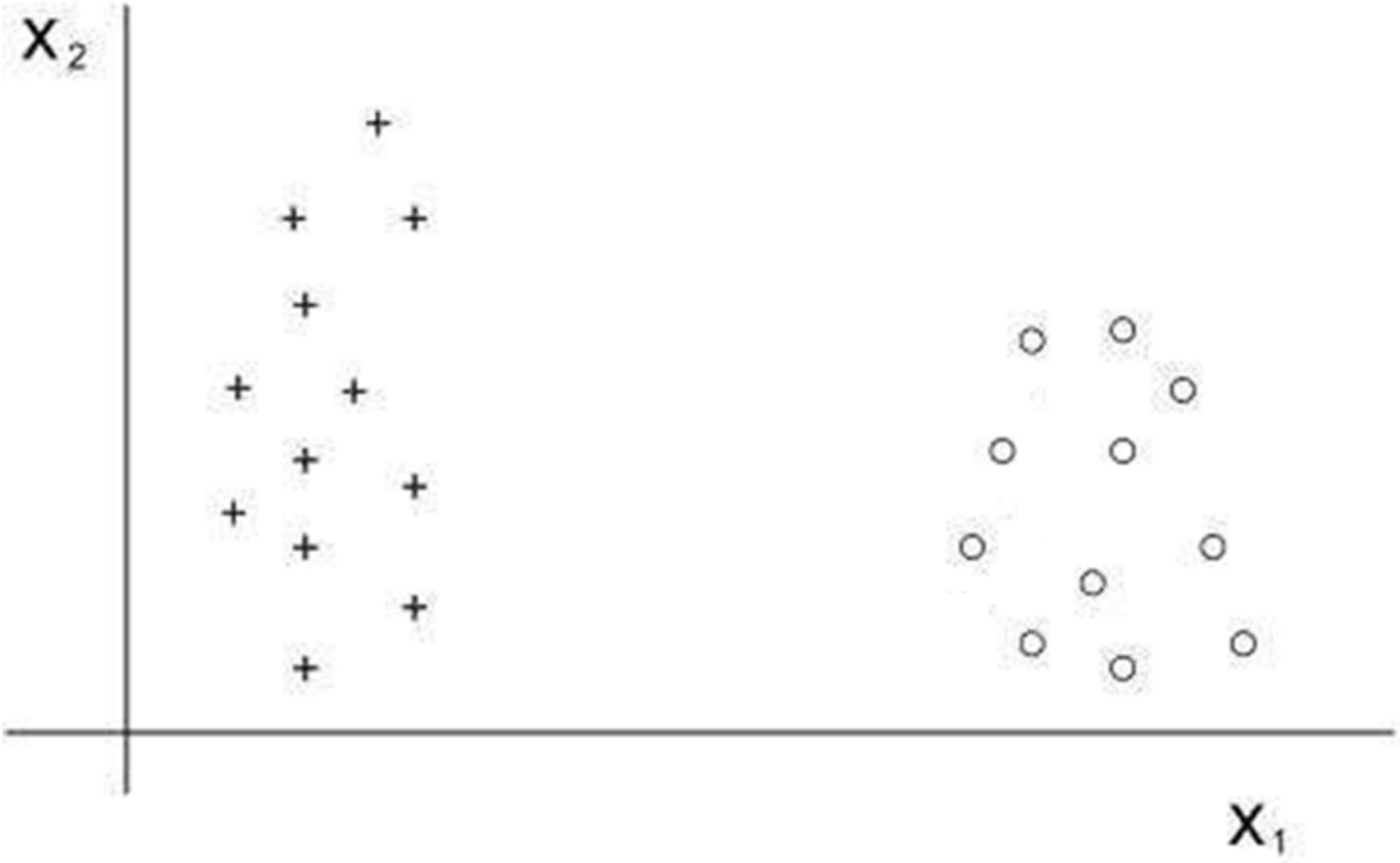
- ¿Qué k se escoge?
  - Proceso prueba y error, dado que depende del problema
  - Las regiones que se forman con 1-NN se denominan regiones de Voronoi

APRENDIZAJE BASADO EN INSTANCIAS

- Algoritmo k-NN:
  - Entradas:
    - D = $\{(x_1, c_1),\ldots,(x_N,c_N)\}$ conjunto de patrones
    - x: nuevo caso a clasificar
    - k
  - k-NN(D, x, k)
    - Para todo objeto (x_i, c_i) en D,
      - Calcular $d_i = d(x_i,x)$
    - Ordenar d_i en orden ascendente
    - Quedarnos con los k casos D_x^k ya clasificados más cercanos a x
    - Asignar a x la clase más frecuente en D_x^k

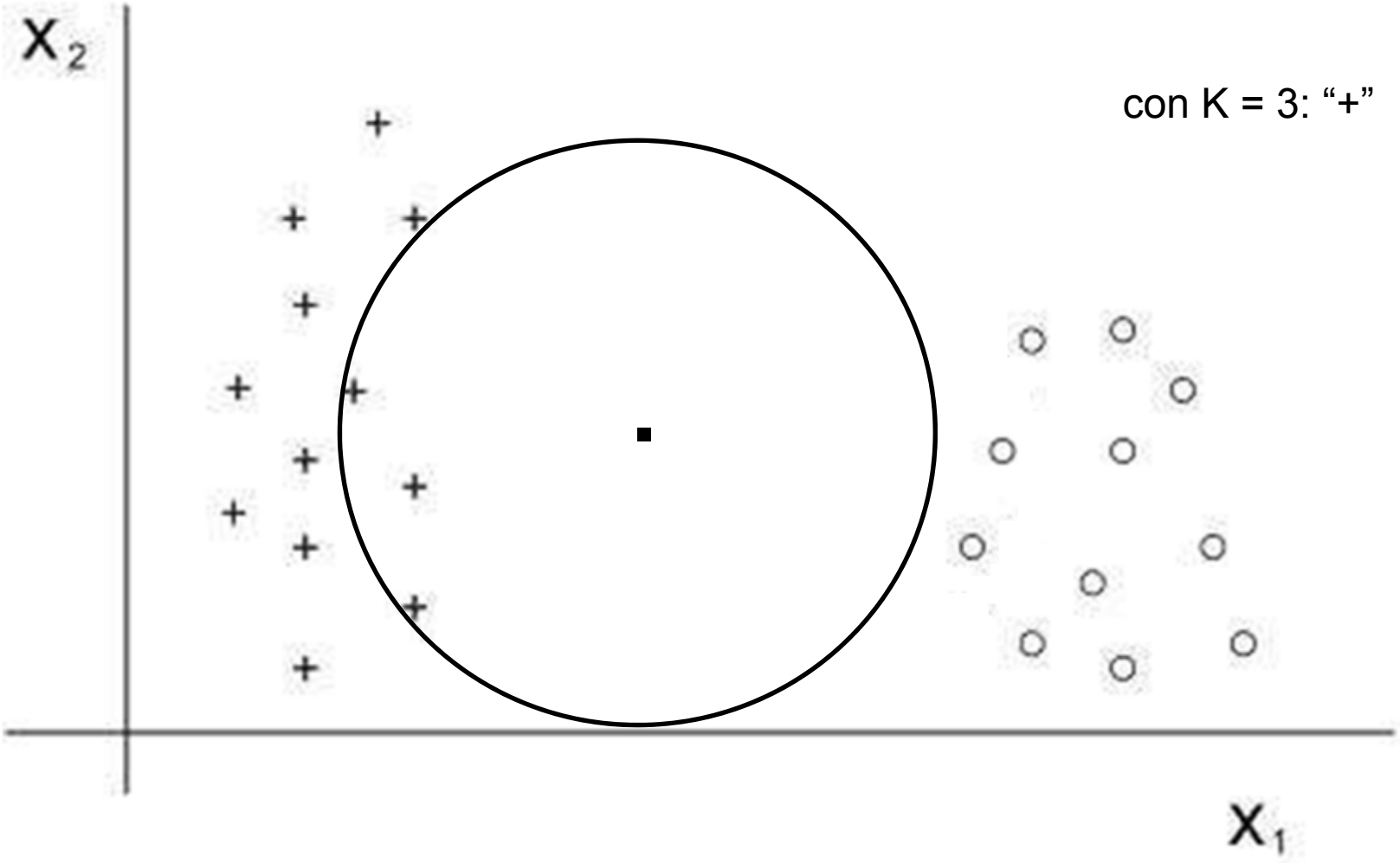
Aprendizaje basado en Instancias

- k-NN: Ejemplo:



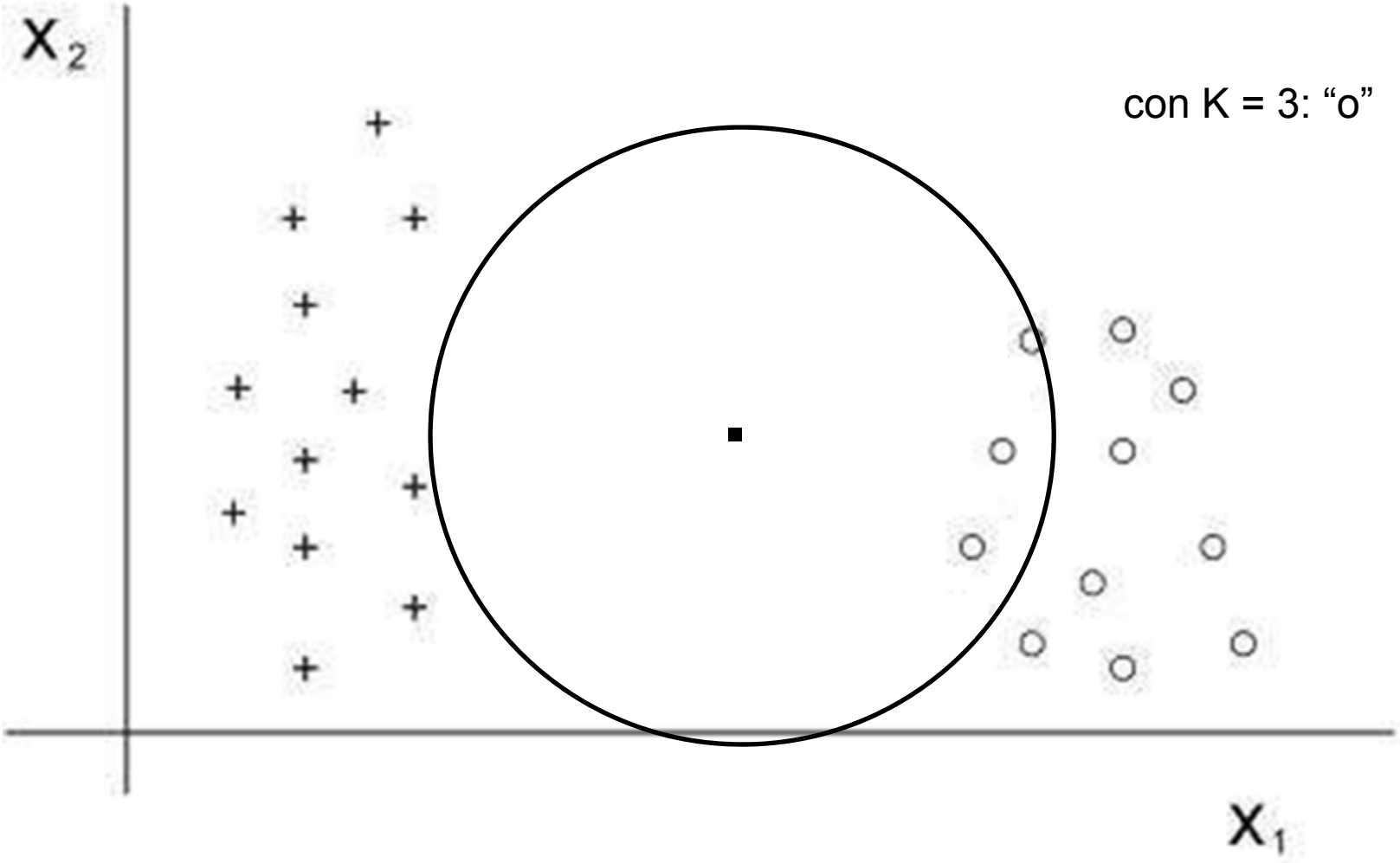
APRENDIZAJE BASADO EN INSTANCIAS

- k-NN: Ejemplo:



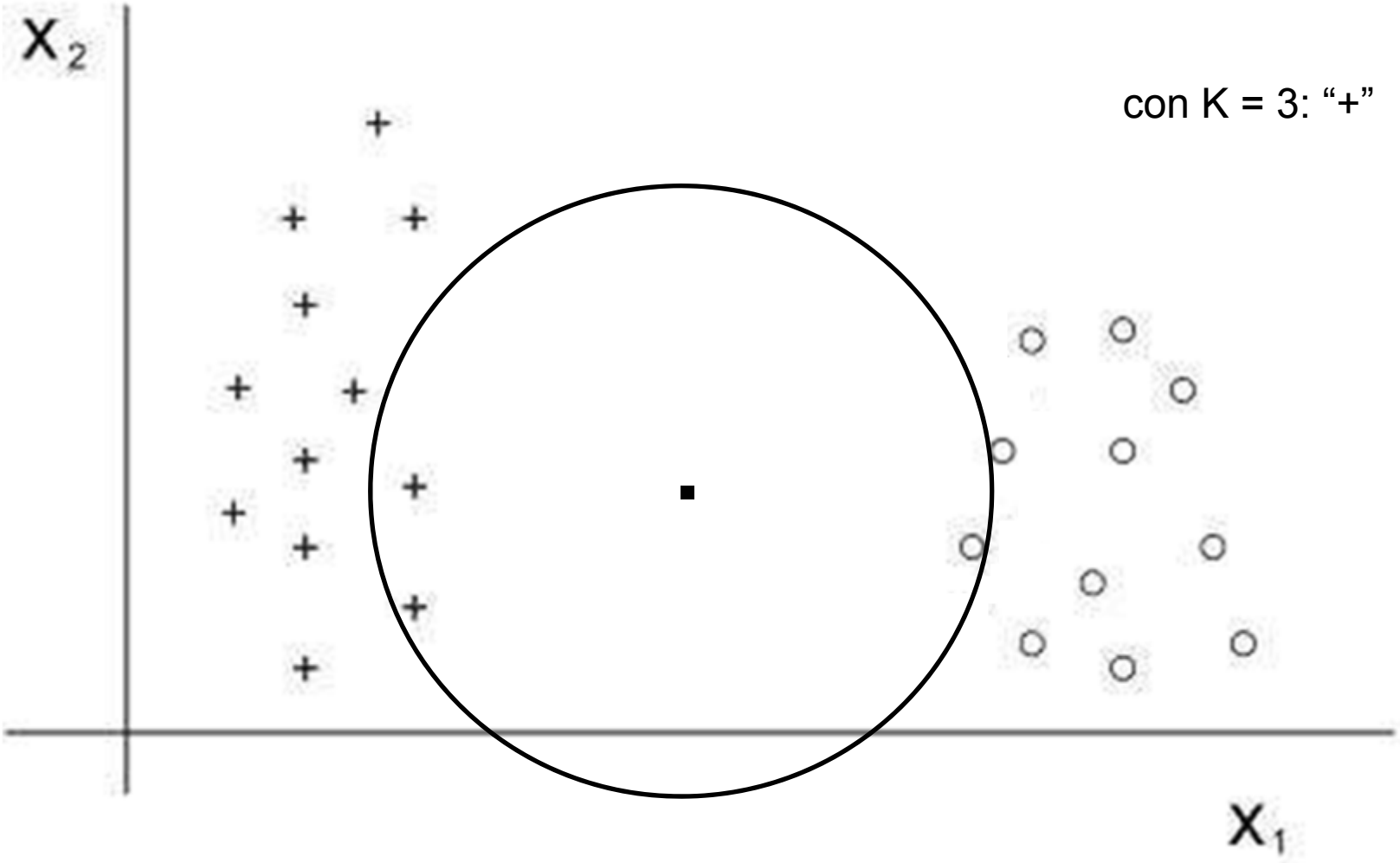
APRENDIZAJE BASADO EN INSTANCIAS

- k-NN: Ejemplo:



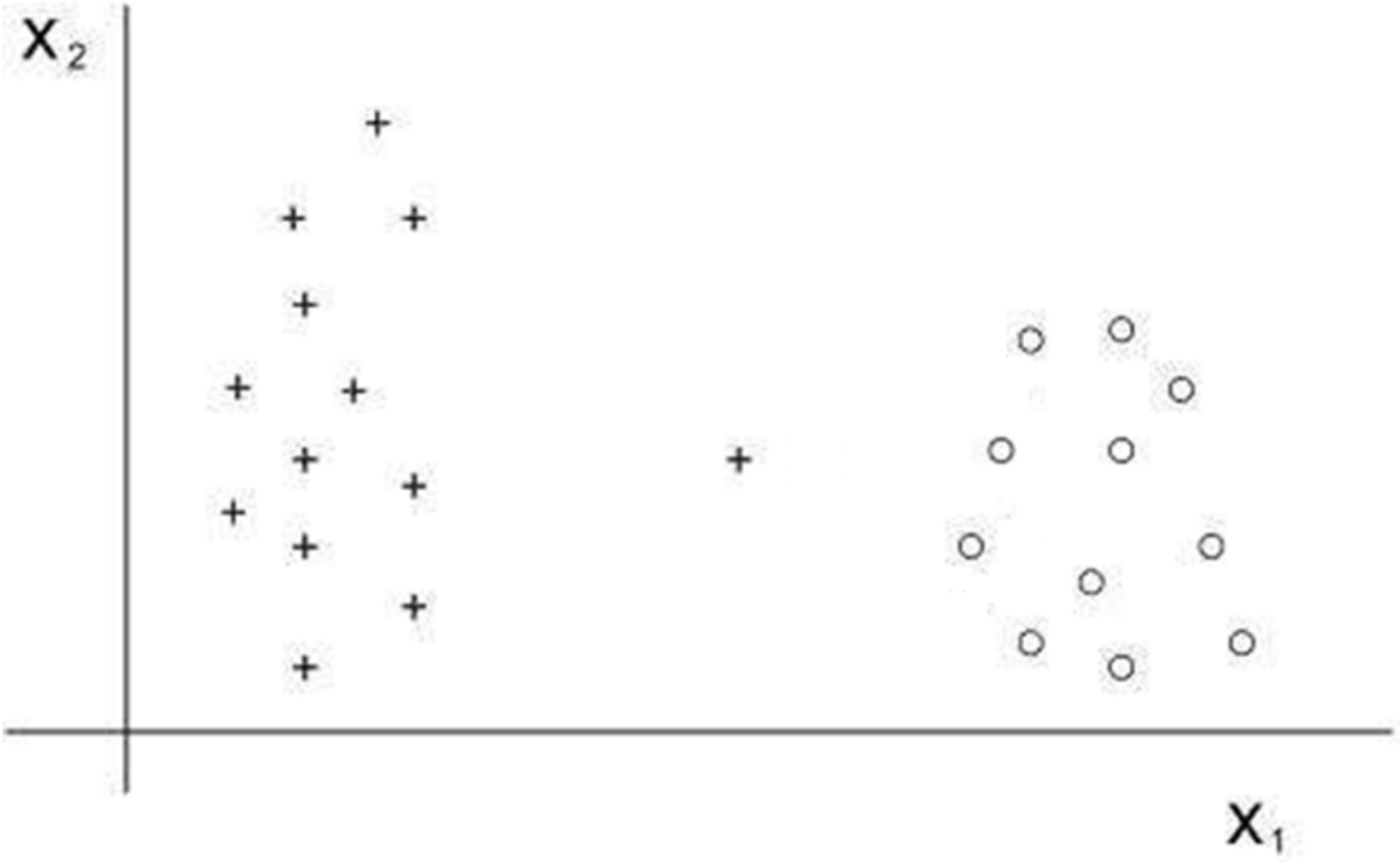
APRENDIZAJE BASADO EN INSTANCIAS

- k-NN: Ejemplo:



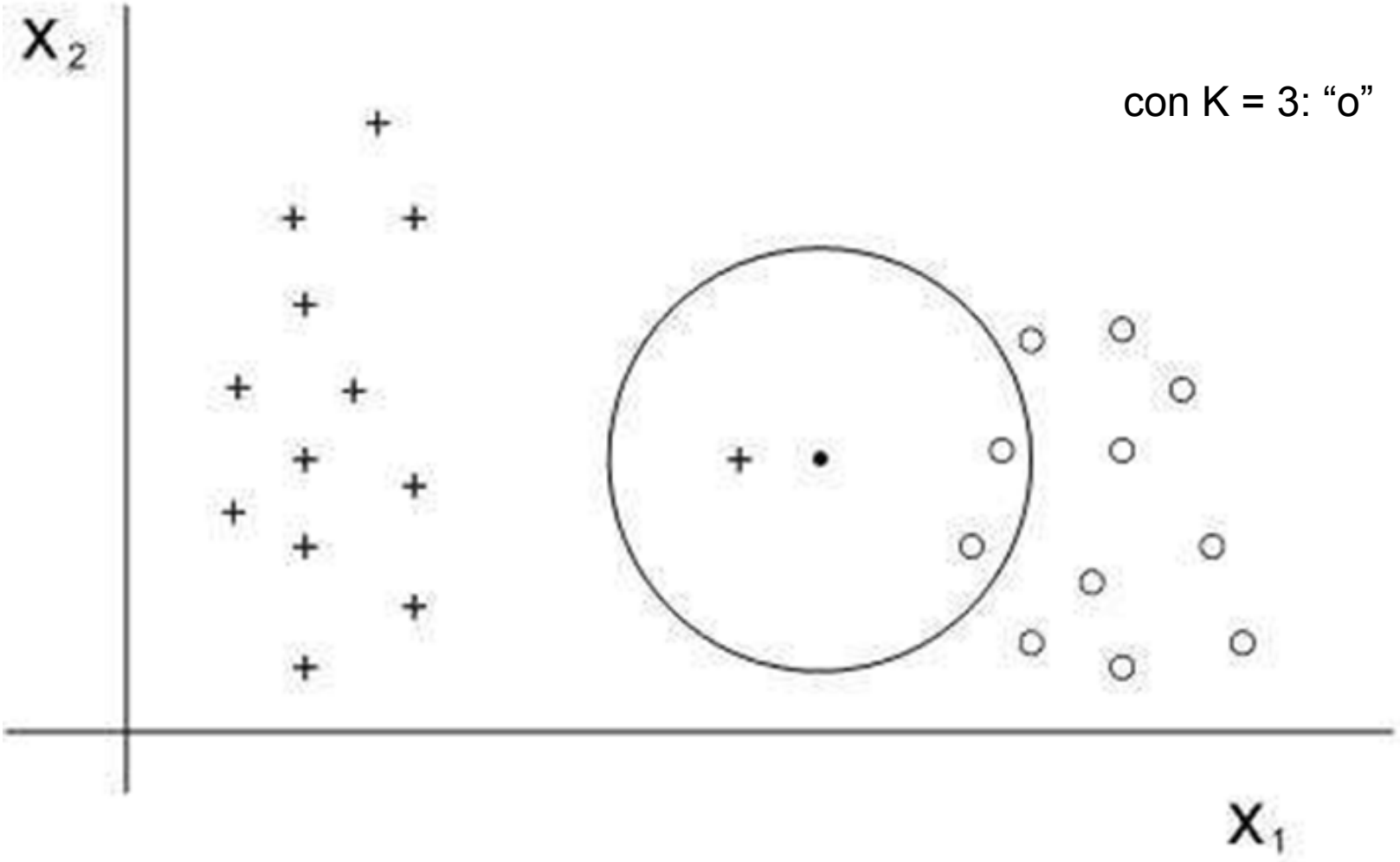
Aprendizaje basado en Instancias

- k-NN: Ejemplo:



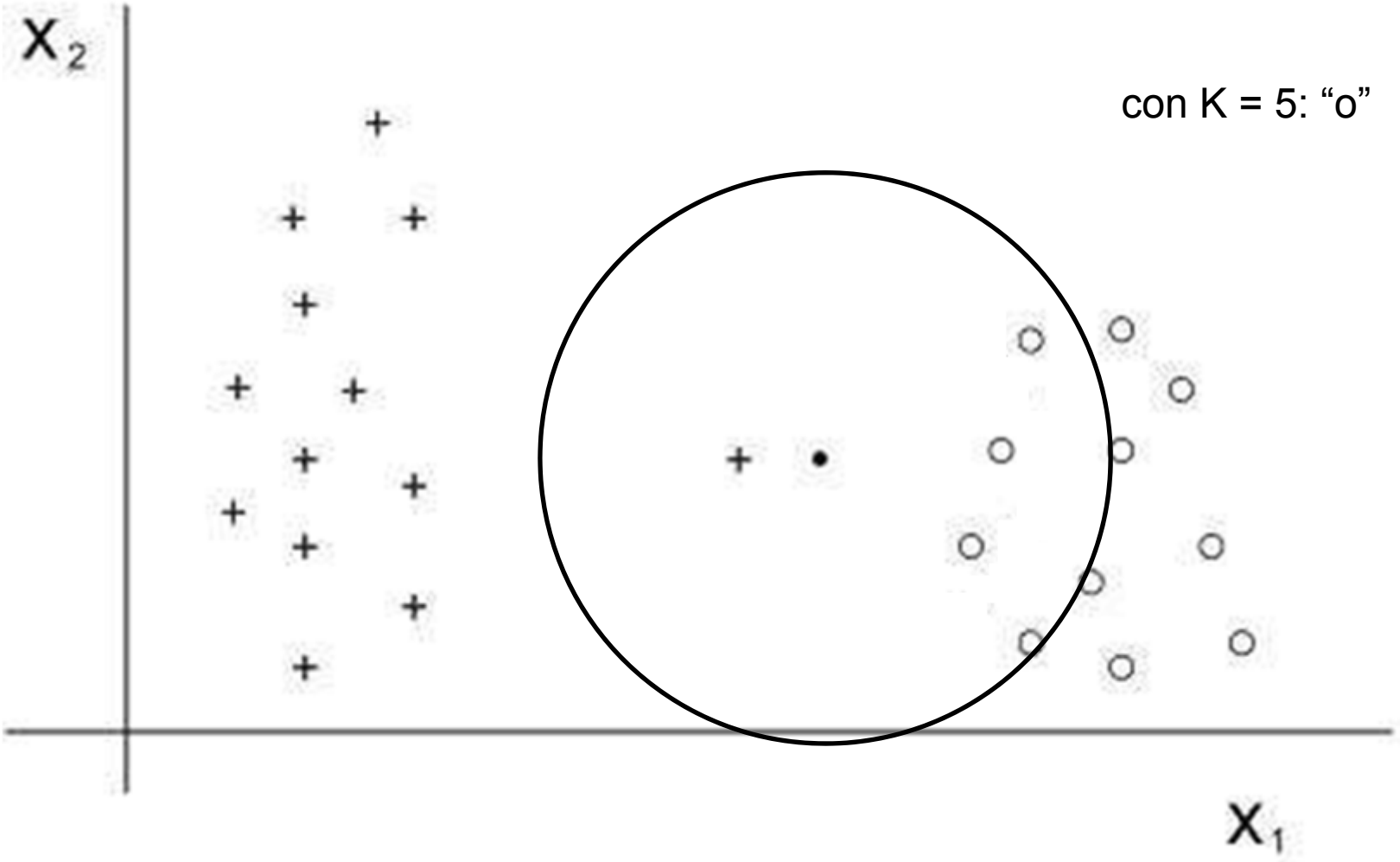
APRENDIZAJE BASADO EN INSTANCIAS

- k-NN: Ejemplo:



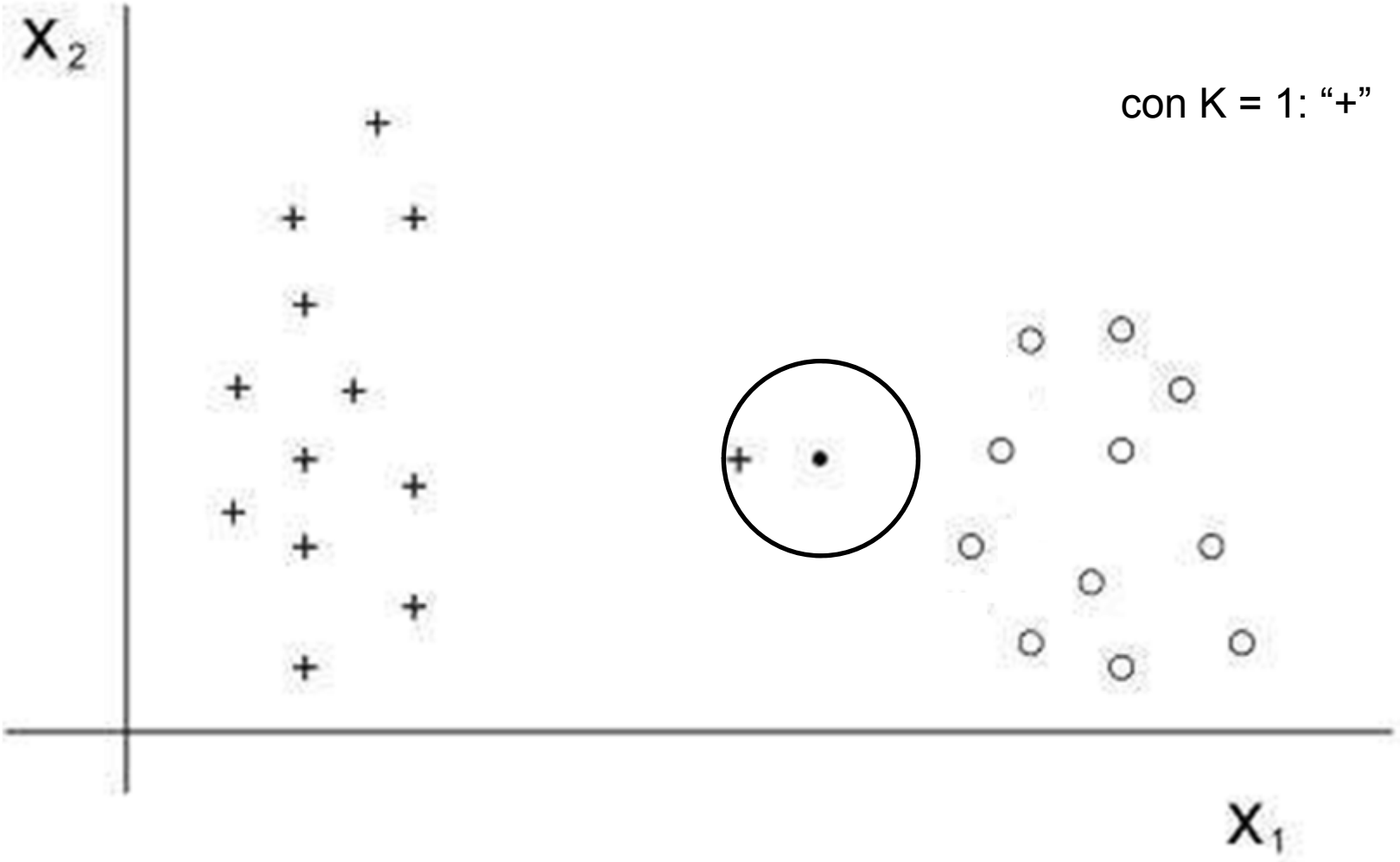
Aprendizaje basado en Instancias

- k-NN: Ejemplo:



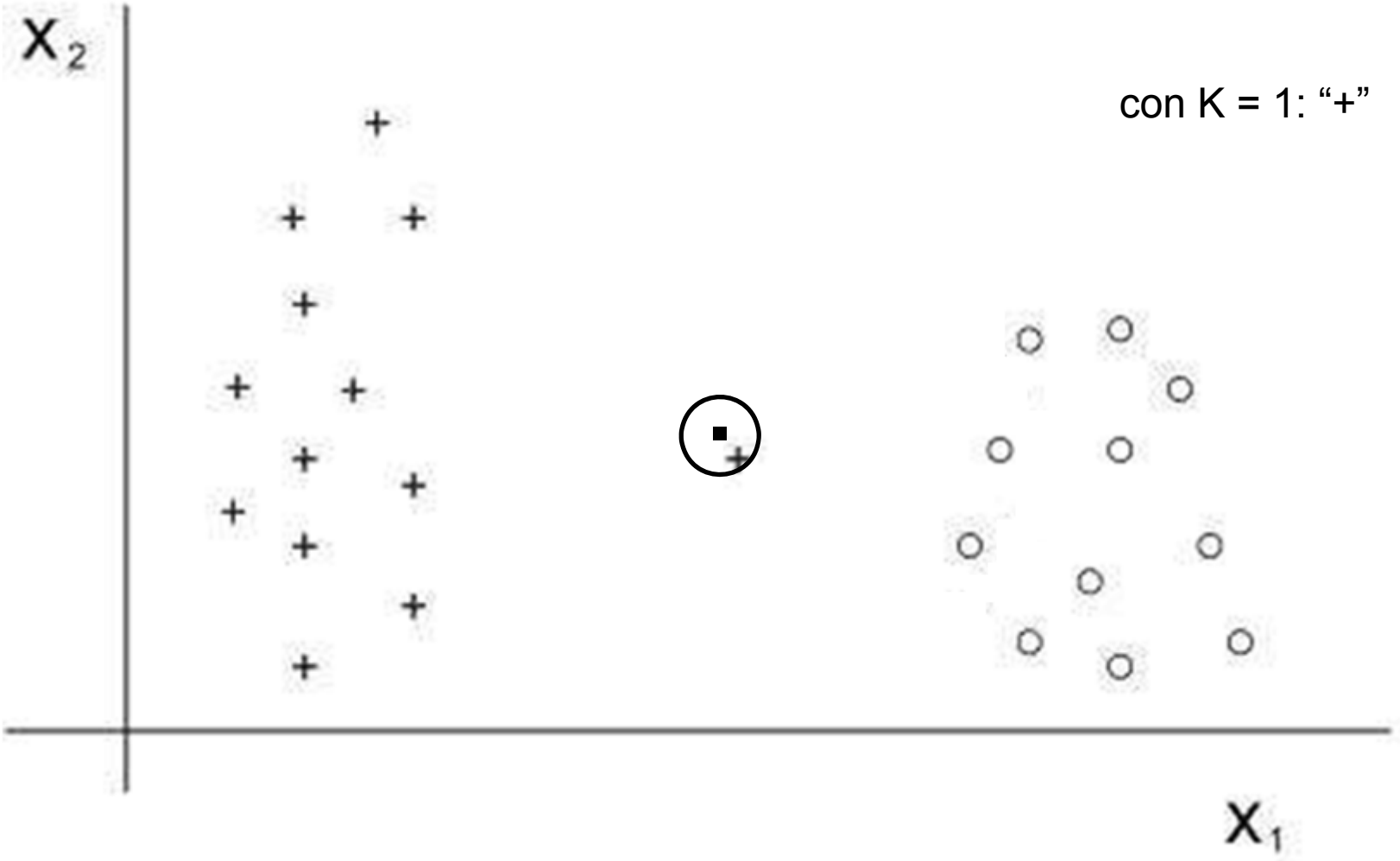
APRENDIZAJE BASADO EN INSTANCIAS

- k-NN: Ejemplo:



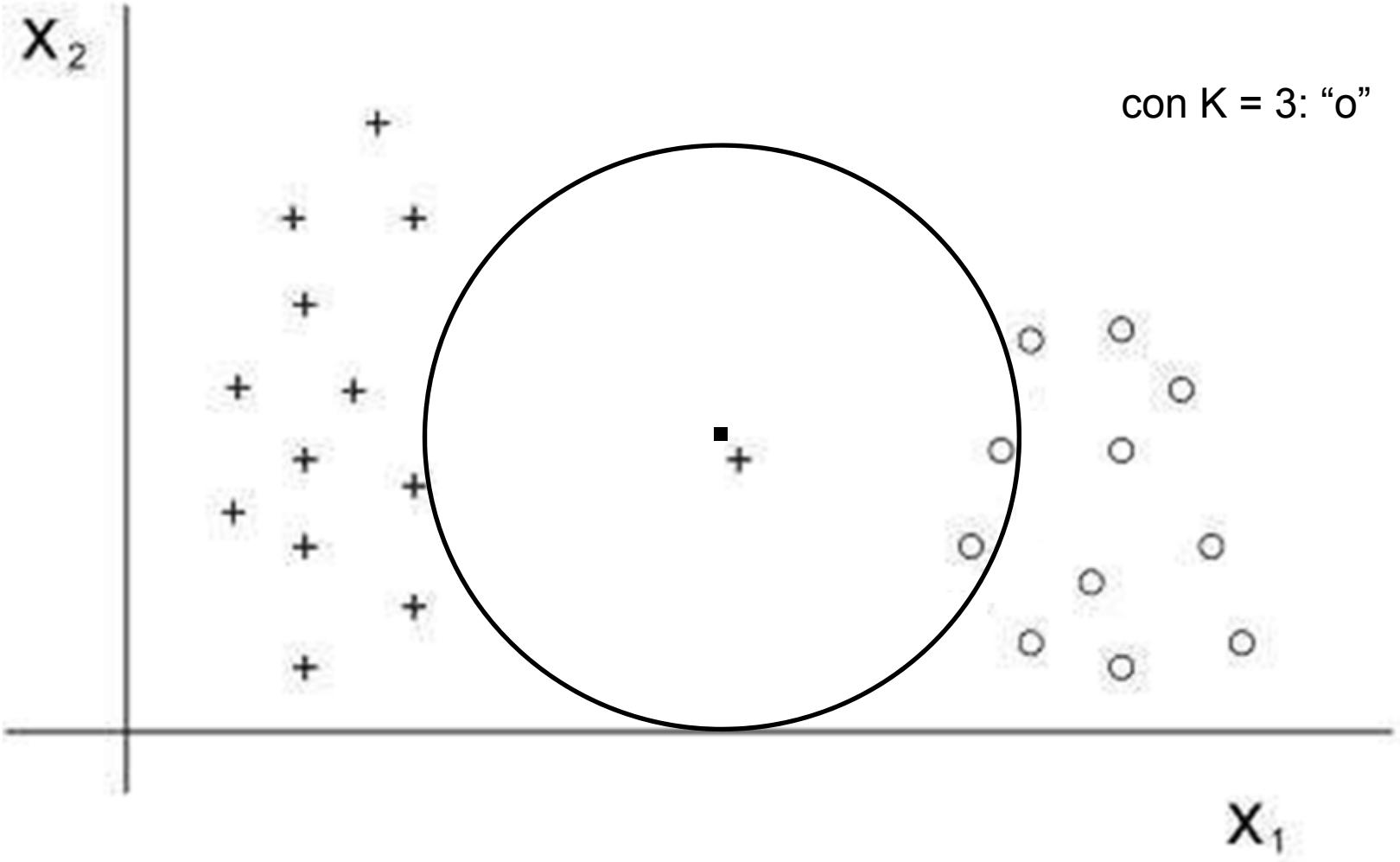
APRENDIZAJE BASADO EN INSTANCIAS

- k-NN: Ejemplo:



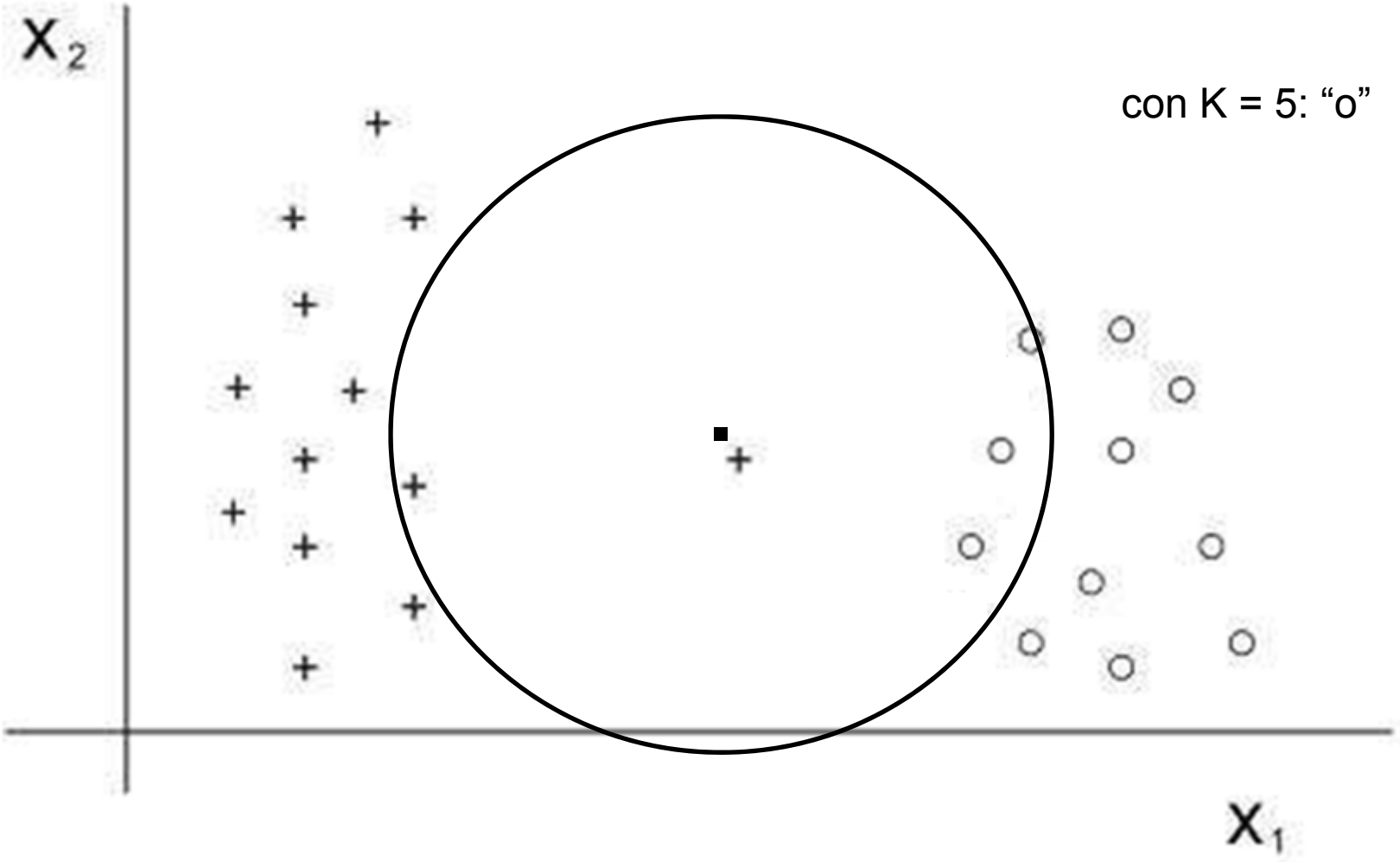
APRENDIZAJE BASADO EN INSTANCIAS

- k-NN: Ejemplo:



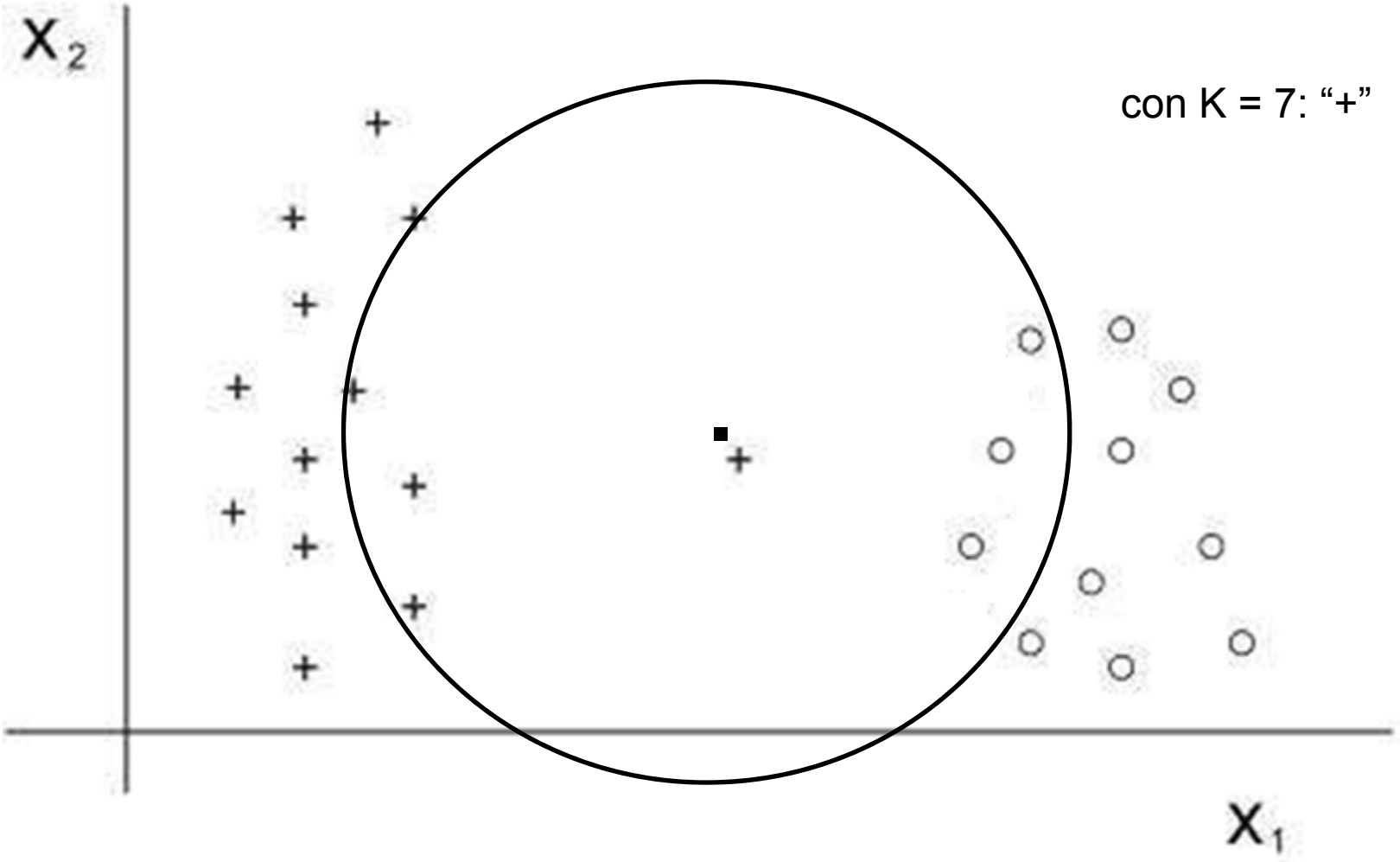
APRENDIZAJE BASADO EN INSTANCIAS

- k-NN: Ejemplo:



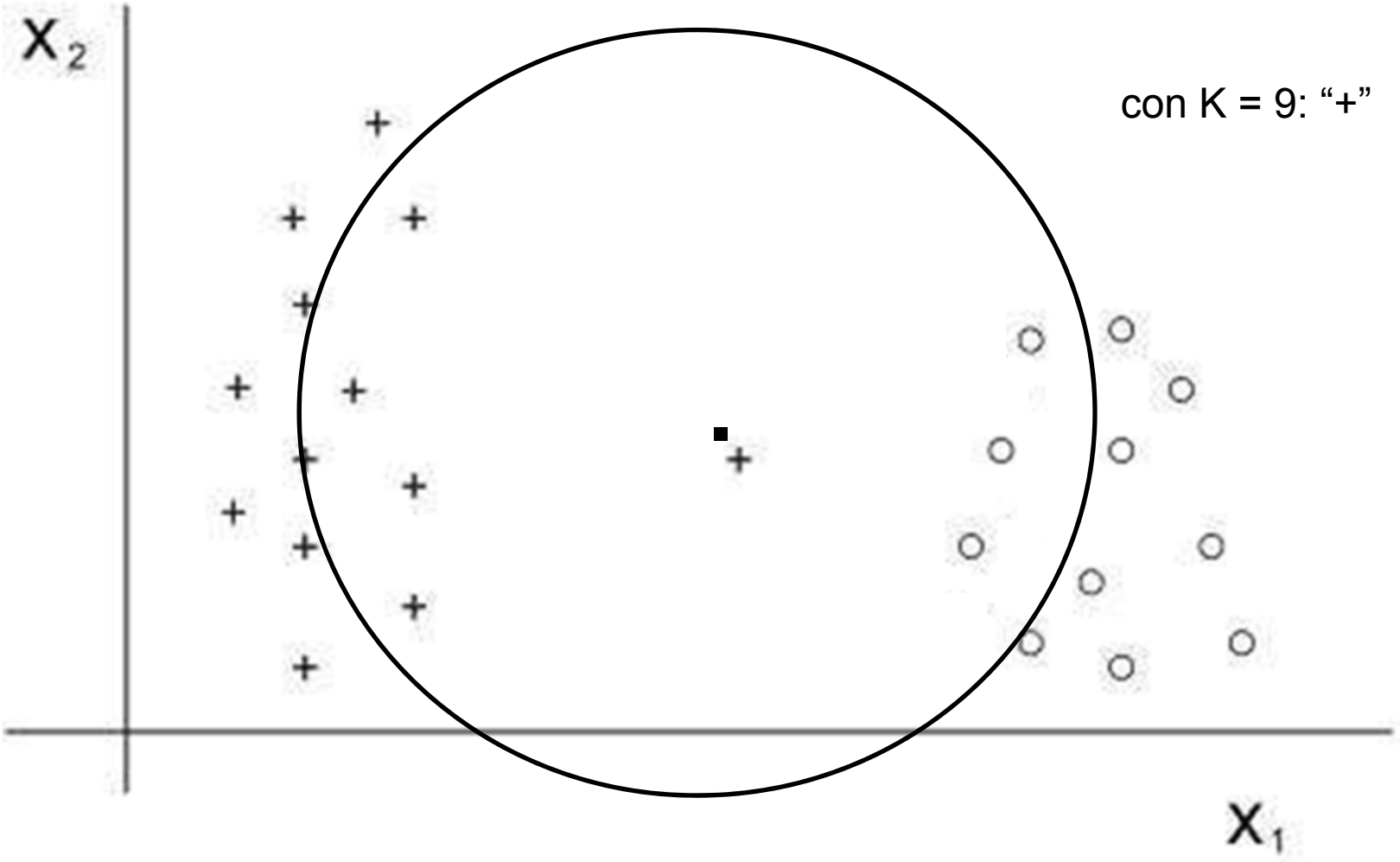
APRENDIZAJE BASADO EN INSTANCIAS

- k-NN: Ejemplo:



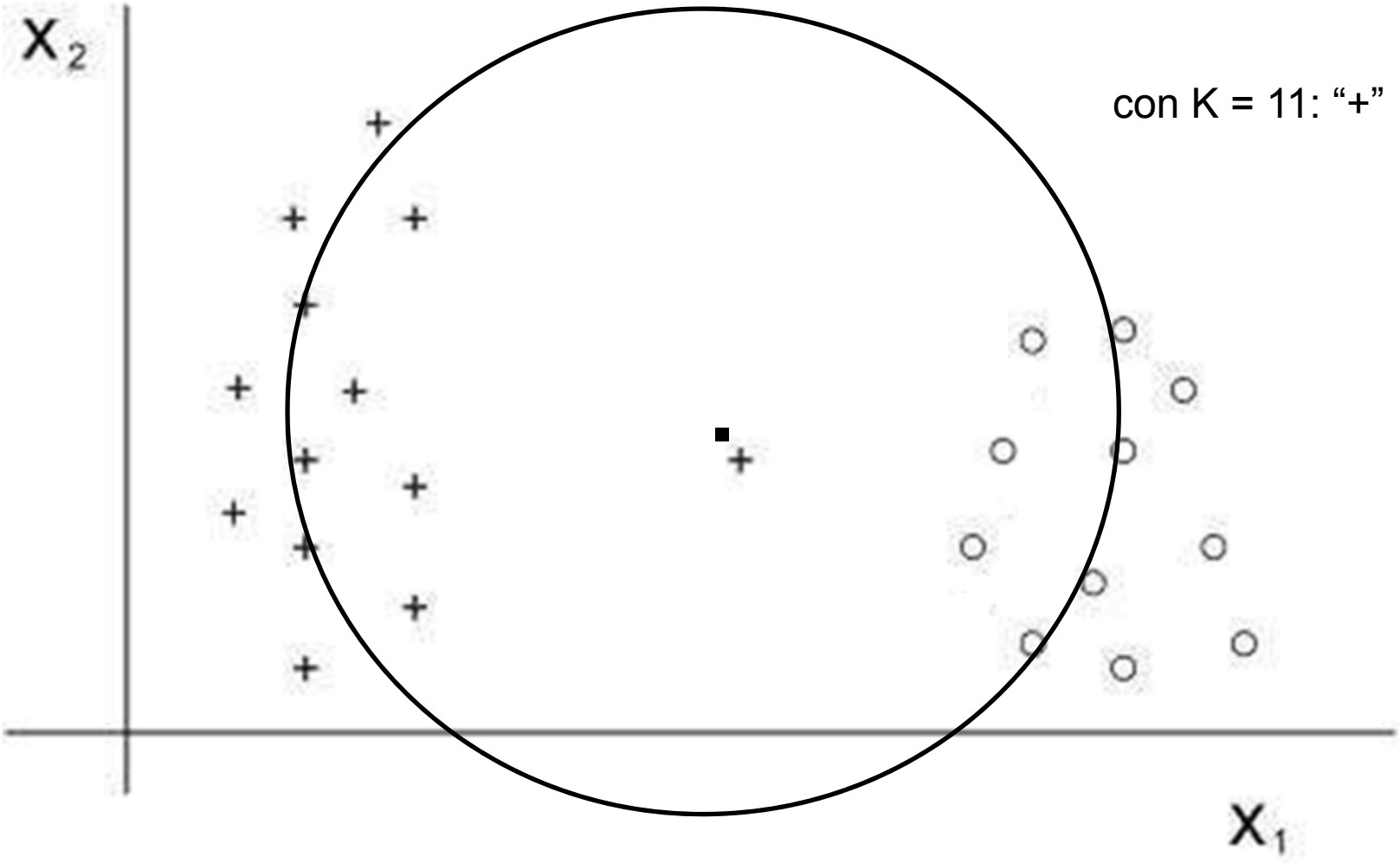
APRENDIZAJE BASADO EN INSTANCIAS

- k-NN: Ejemplo:



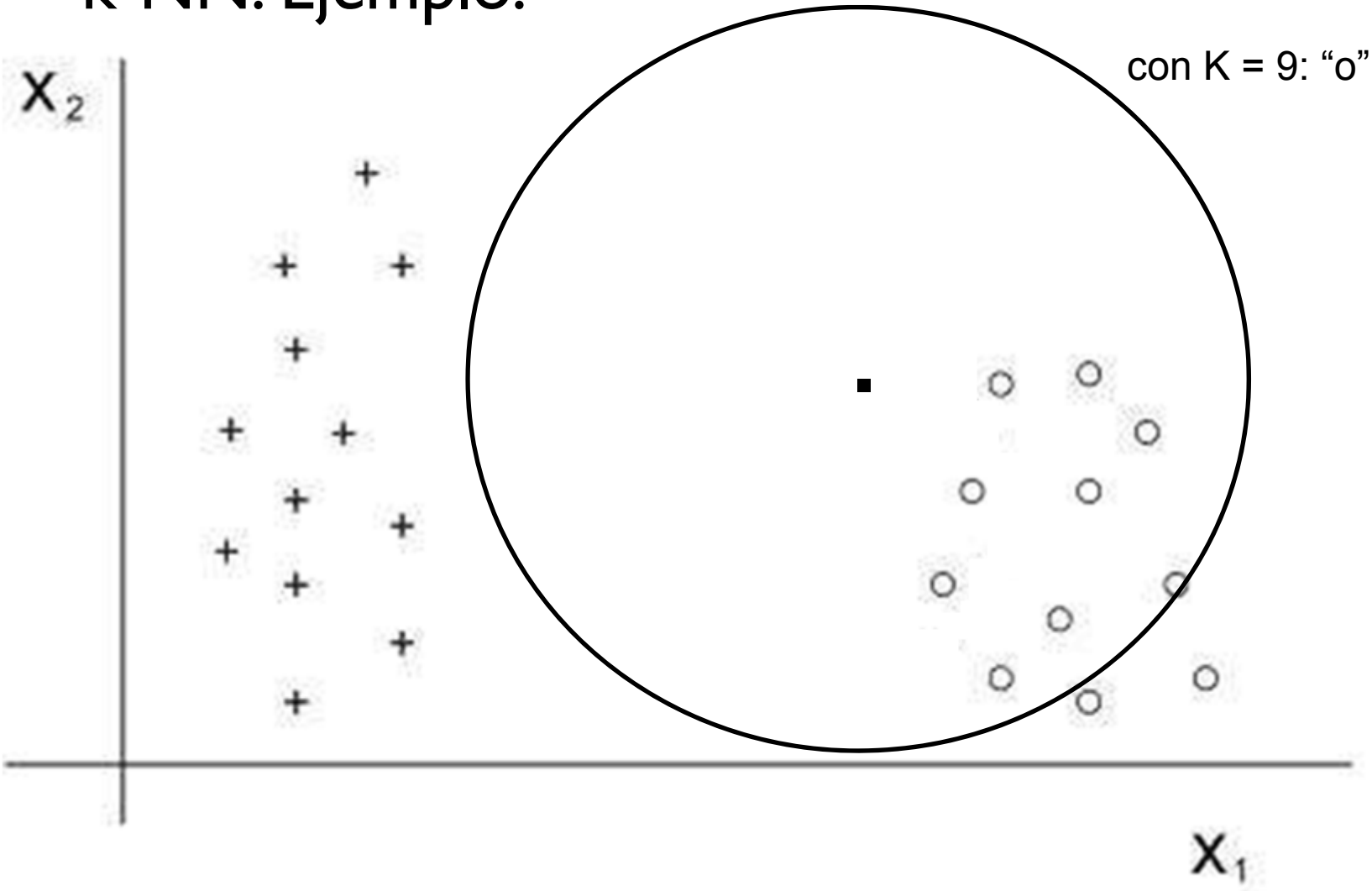
APRENDIZAJE BASADO EN INSTANCIAS

- k-NN: Ejemplo:



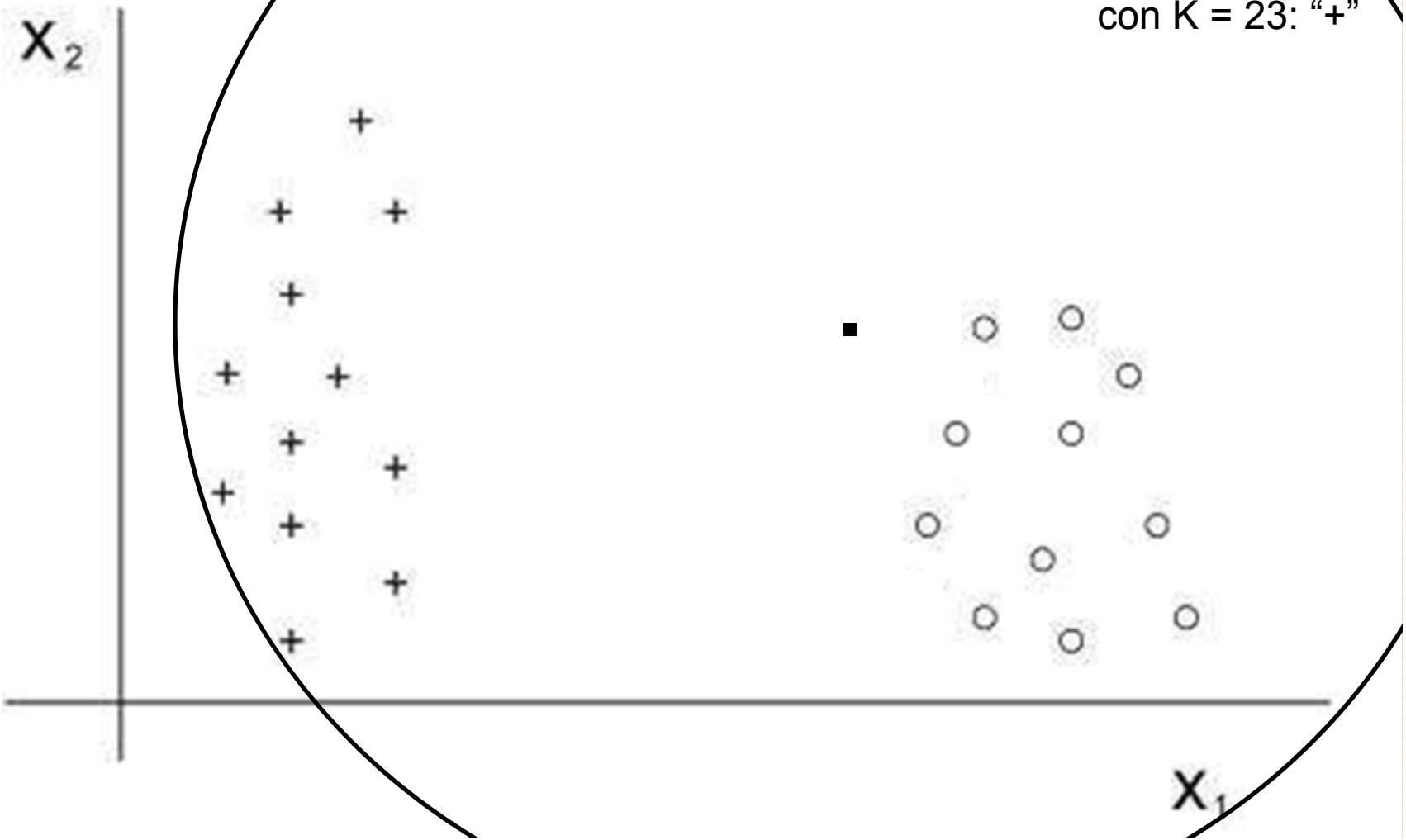
APRENDIZAJE BASADO EN INSTANCIAS

- k-NN: Ejemplo:



APRENDIZAJE BASADO EN INSTANCIAS

- k-NN: Ejemplo:

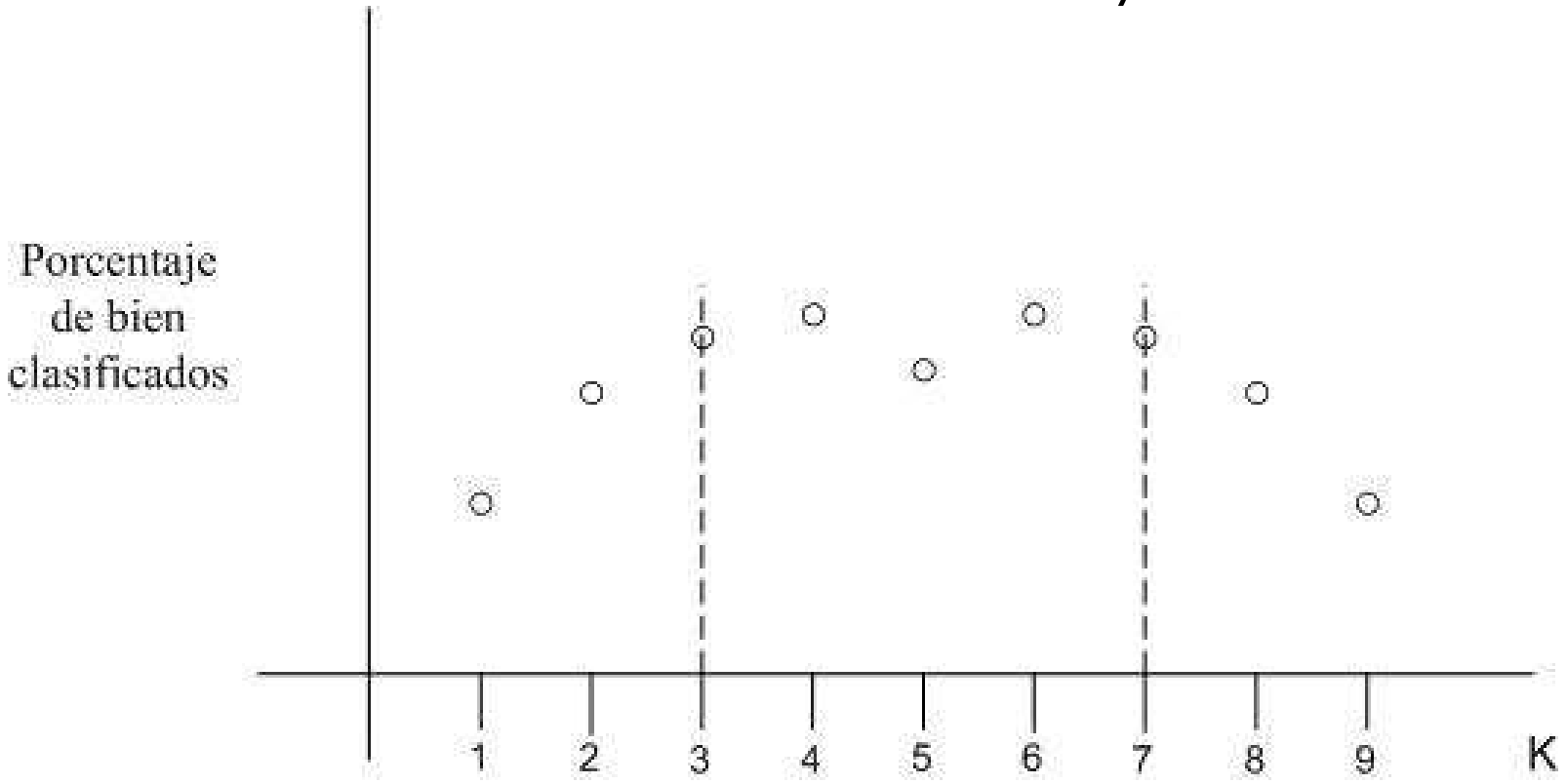


Aprendizaje basado en Instancias

- k-NN: Valor de k:
  - Si se elige k muy bajo, el resultado es muy sensible al ruido.
  - Si es muy alto, las zonas que tengan muchos ejemplos pueden acaparar a zonas que tengan menos.
  - Una forma de estimar k es probando distintos valores, midiendo los resultados dejando un elemento del conjunto fuera y clasificando con el resto
    - 1-out-cross-validation

APRENDIZAJE BASADO EN INSTANCIAS

- k-NN: Valor de k:
  - Se constata empíricamente que el porcentaje de casos bien clasificados es no monótono con respecto de k
    - Una buena elección son valores de k entre 3 y 7



Aprendizaje basado en Instancias

- k-NN: Valor de k:
  - Por lo general, se elige un k impar para no tener problemas de empate.
    - Los valores usuales son bajos: 1, 3 y 5.
    - Para el caso discreto (clasificación), cuando se tienen 2 clases
      - Si se tienen más de 2 clases se puede dar empate con valores pares e impares
- En caso de que se produzca un empate entre dos o más clases, conviene tener una regla heurística para su ruptura, por ejemplo:
  - Seleccionar la clase que contenga al vecino más próximo
  - Seleccionar la clase con distancia media menor
  - etc.

APRENDIZAJE BASADO EN INSTANCIAS

- El paradigma K-NN es un tanto atípico si se compara con el resto de paradigmas clasificatorios:
  - En el resto de paradigmas la clasificación de un nuevo caso se lleva a cabo a partir de dos tareas:
    1. Inducción del modelo clasificatorio
    2. La posterior deducción (o aplicación) sobre el nuevo caso,
  - En cambio, en el paradigma K-NN, al no existir modelo explícito, las dos tareas anteriores se encuentran colapsadas en lo que se acostumbra a denominar **transinducción**.

Aprendizaje basado en Instancias

- Variantes del k-NN:
  - k-NN con rechazo
  - k-NN con distancia media
  - k-NN con ponderación de vecinos
  - k-NN con distancia mínima
  - k-NN con ponderación de variables

APRENDIZAJE BASADO EN INSTANCIAS

- Variantes del k-NN:
  - **k-NN con rechazo**
  - k-NN con distancia media
  - k-NN con ponderación de vecinos
  - k-NN con distancia mínima
  - k-NN con ponderación de variables

APRENDIZAJE BASADO EN INSTANCIAS

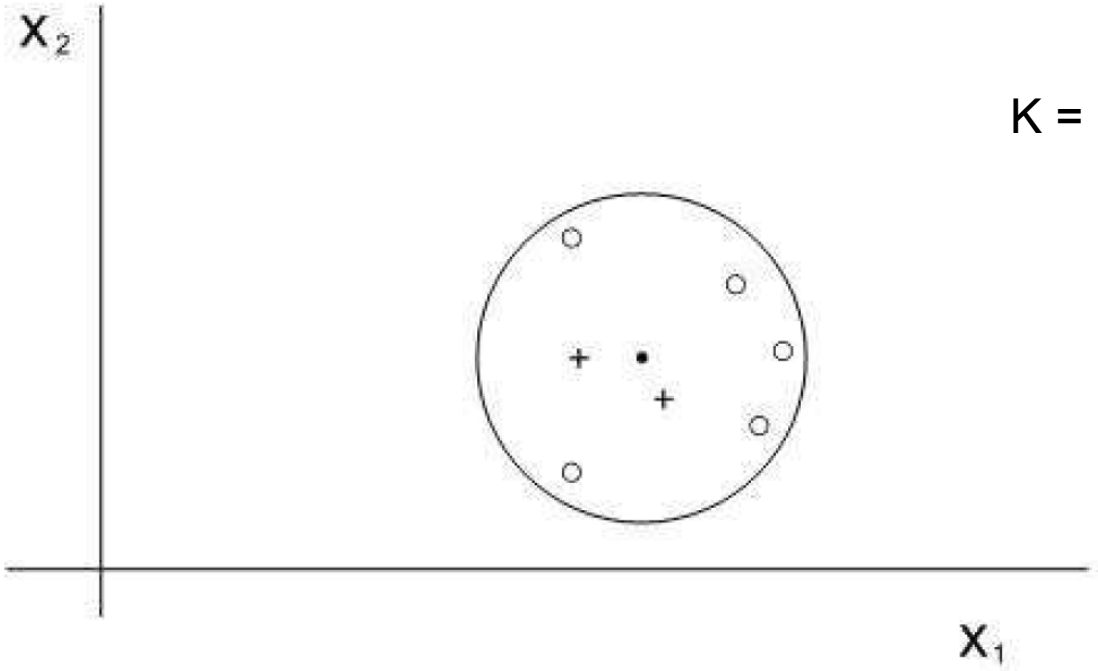
- Variantes: k-NN con rechazo:
  - Para poder clasificar un caso hay que tener ciertas garantías
    - Puede ocurrir que un caso quede sin clasificar, si no hay esas garantías de que se asigne a la clase correcta
    - Ejemplos de garantías:
      - El número de votos obtenidos por la clase deberá superar un umbral prefijado.
        - Por ejemplo, si k=10, con 2 clases, el umbral podría ser 6
      - El número de votos obtenidos por la clase más votada deberá superar a la segunda más votada en cierta cantidad
        - Por ejemplo, si k=20, con 4 clases, se asignará clase si la diferencia de votos entre la más votada y la siguiente sea mayor que 3
      - Para asignar una clase, debe de haber mayoría absoluta.

APRENDIZAJE BASADO EN INSTANCIAS

- Variantes del k-NN:
  - k-NN con rechazo
  - **k-NN con distancia media**
  - k-NN con ponderación de vecinos
  - k-NN con distancia mínima
  - k-NN con ponderación de variables

APRENDIZAJE BASADO EN INSTANCIAS

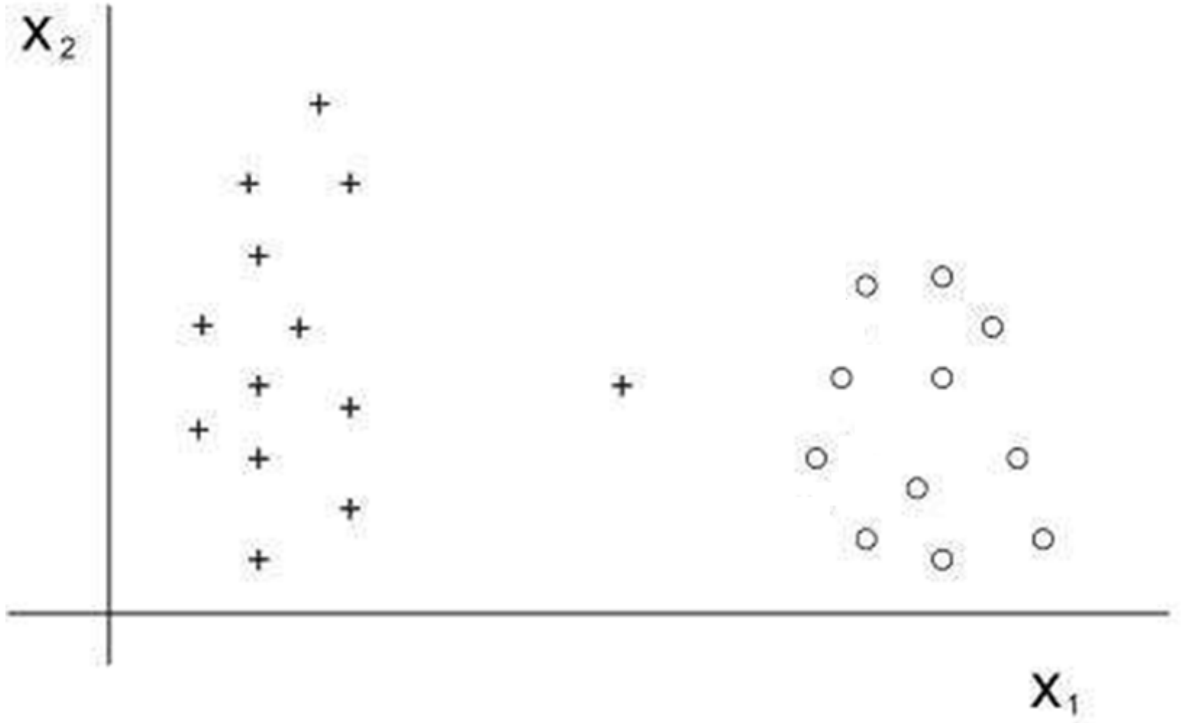
- Variantes: k-NN con distancia media:
  - Asignar un nuevo caso a la clase cuya distancia media sea menor



- En este caso, la nueva instancia se clasifica como "+", porque la distancia media a los dos casos de ejemplo de clase "+" es menor que la distancia media a los 5 casos "o"

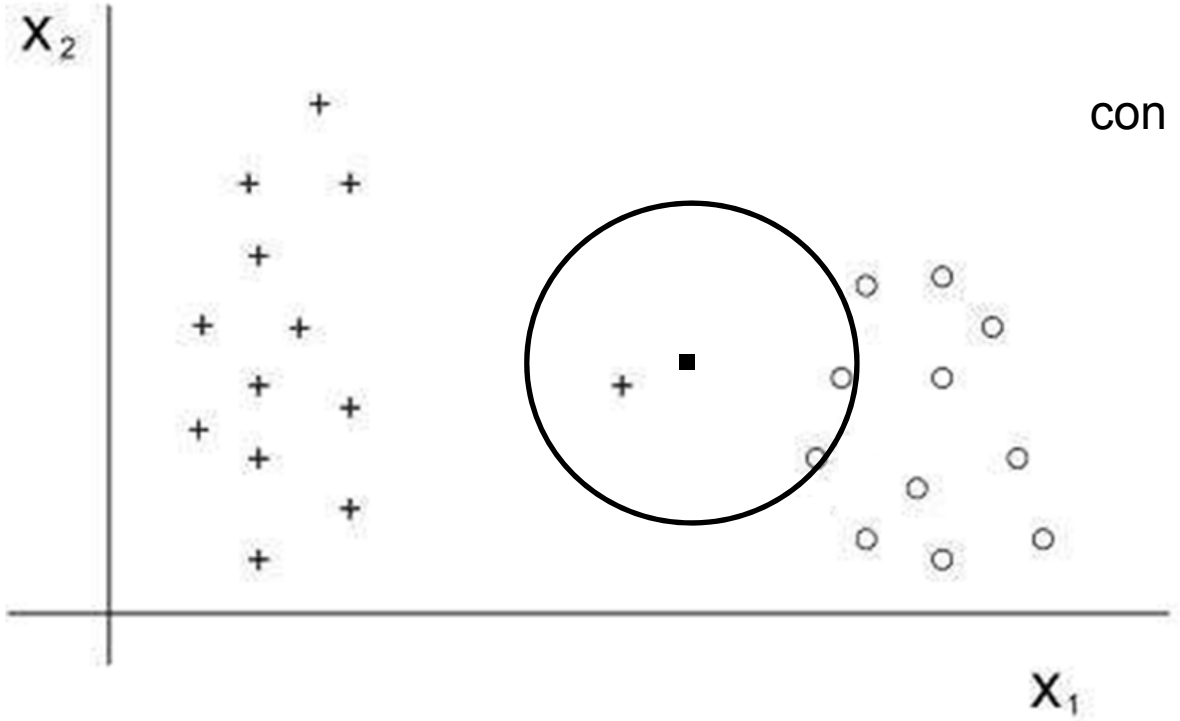
APRENDIZAJE BASADO EN INSTANCIAS

- Variantes: k-NN con distancia media:
  - Asignar un nuevo caso a la clase cuya distancia media sea menor
    - Problema: mucha sensibilidad al ruido: Ejemplo:



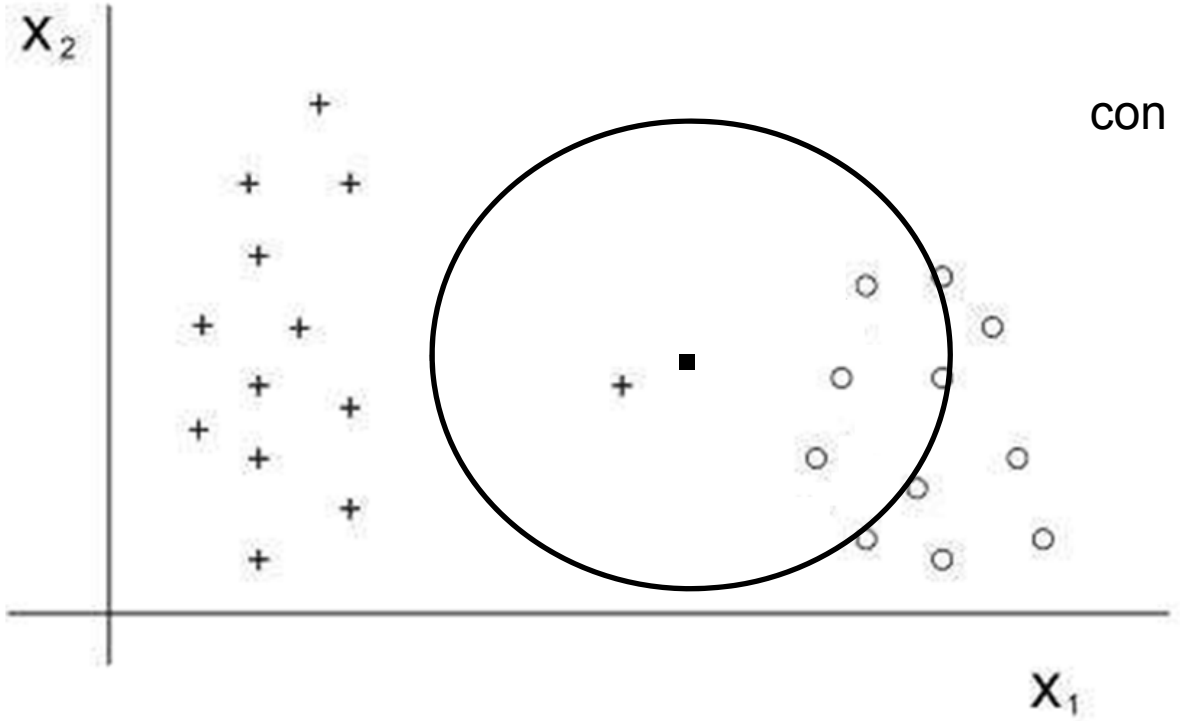
APRENDIZAJE BASADO EN INSTANCIAS

- Variantes: k-NN con distancia media:
  - Asignar un nuevo caso a la clase cuya distancia media sea menor
    - Problema: mucha sensibilidad al ruido: Ejemplo:



APRENDIZAJE BASADO EN INSTANCIAS

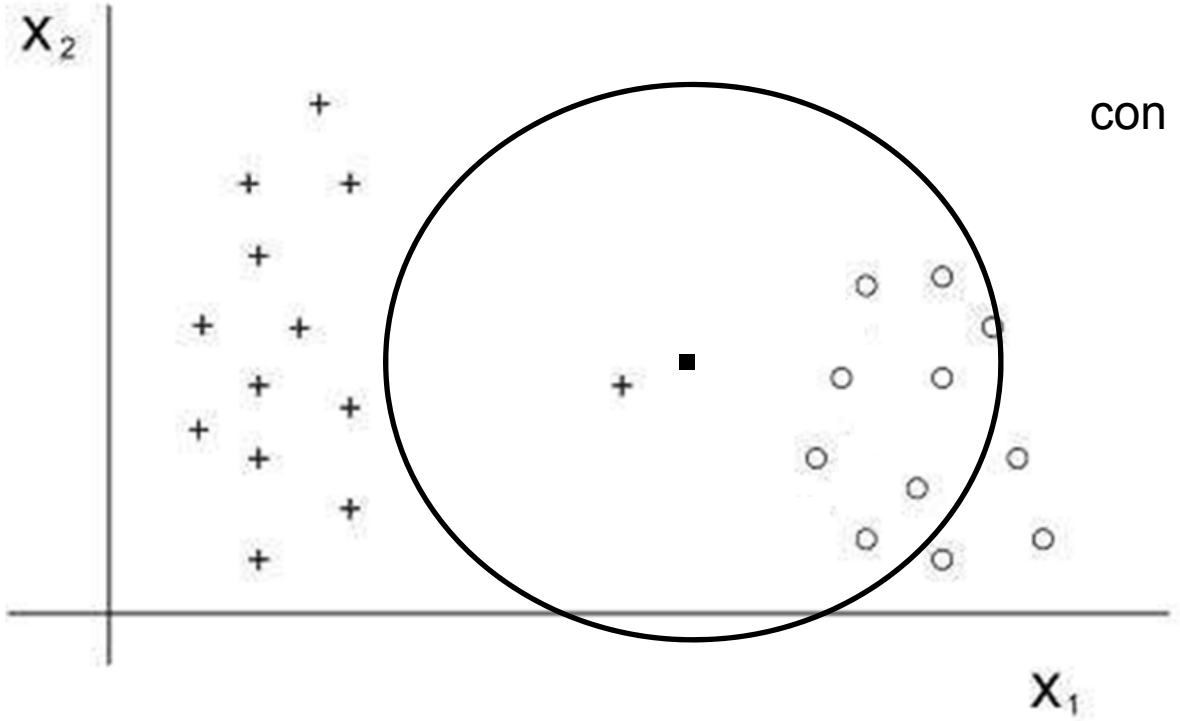
- Variantes: k-NN con distancia media:
  - Asignar un nuevo caso a la clase cuya distancia media sea menor
    - Problema: mucha sensibilidad al ruido: Ejemplo:



con K = 5: “+”

APRENDIZAJE BASADO EN INSTANCIAS

- Variantes: k-NN con distancia media:
  - Asignar un nuevo caso a la clase cuya distancia media sea menor
    - Problema: mucha sensibilidad al ruido: Ejemplo:



con K = 9: “+”

APRENDIZAJE BASADO EN INSTANCIAS

- Variantes del k-NN:
  - k-NN con rechazo
  - k-NN con distancia media
  - **k-NN con ponderación de vecinos**
  - k-NN con distancia mínima
  - k-NN con ponderación de variables

APRENDIZAJE BASADO EN INSTANCIAS

- Variantes: k-NN con ponderación (pesado) de vecinos
  - *Distance-Weighted k-NN*:
  - Ponderar la importancia que aporta cada vecino al valor de la función objetivo, en función de su distancia
    - Ejemplo: inverso de la distancia
    - Ejemplo: inverso de la distancia al cuadrado

APRENDIZAJE BASADO EN INSTANCIAS

- Variantes: k-NN con ponderación (pesado) de vecinos
  - Dada una instancia x, y sus k vecinos: $x_1,\ldots,x_k$
    - Caso discreto:
      - Función objetivo:

$$\hat{f}(x) = \underset{v \in V}{\text{argmáx}} \sum_{i=1}^{k} w_i \delta(v, f(x_i))$$

donde $w_i = 1 / d(x,x_i)^2$

- Si algún x_i coincide exactamente con x, se utiliza la versión no ponderada para los que cumplan dicha condición.
  - O devolver la clase de x_i

APRENDIZAJE BASADO EN INSTANCIAS

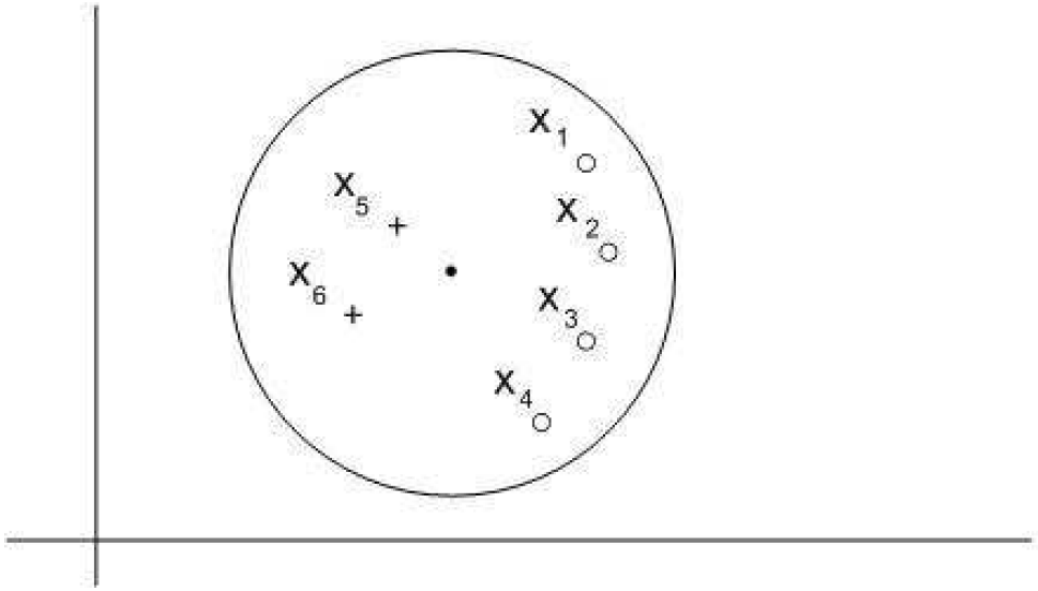
- Variantes: k-NN con ponderación (pesado) de vecinos
  - Dada una instancia x, y sus k vecinos: $x_1,\ldots,x_k$
    - Caso continuo:
      - Función objetivo:

$$\hat{f}(x) = \frac{\sum_{i=1}^{k} w_i f(x_i)}{\sum_{i=1}^{k} w_i}$$

donde $w_i = 1 / d(x,x_i)^2$

APRENDIZAJE BASADO EN INSTANCIAS

- Variantes: k-NN con ponderación (pesado) de vecinos



	$d(\mathbf{x}_i, \mathbf{x})$	w_i
$\mathbf{x}_1$	2	0,5
$\mathbf{x}_2$	2	0,5
$\mathbf{x}_3$	2	0,5
$\mathbf{x}_4$	2	0,5
$\mathbf{x}_5$	0,7	1/0,7
$\mathbf{x}_6$	0,8	1/0,8

$$w_i = \frac{1}{d(x_i, x)}$$

  - En este caso, los pesos relativos a la clase "o" suman 2, y los pesos relativos a la clase "+" suman 2.67
    - Se clasifica como "+" a pesar de haber más vecinos de clase "o"

APRENDIZAJE BASADO EN INSTANCIAS

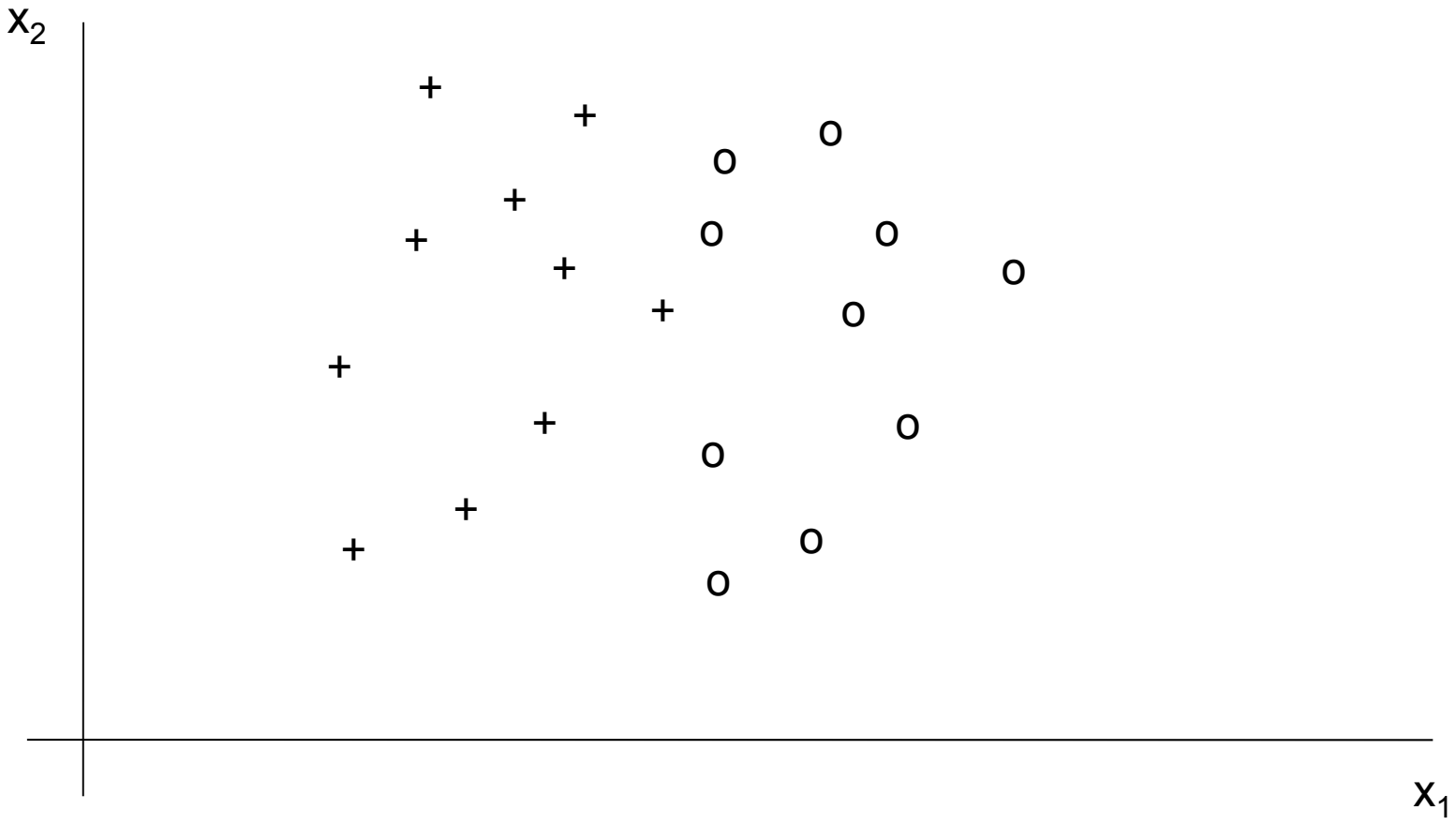
- Variantes del k-NN:
  - k-NN con rechazo
  - k-NN con distancia media
  - k-NN con ponderación de vecinos
  - **k-NN con distancia mínima**
  - k-NN con ponderación de variables

APRENDIZAJE BASADO EN INSTANCIAS

- Variantes: k-NN con distancia mínima:
  - Se comienza seleccionando **un caso por clase**
    - Generalmente, el caso más cercano al baricentro o centroide de todos los elementos de dicha clase
    - De esta forma, se reduce la dimensión del fichero de casos a almacenar del número de ejemplos al número de clases
  - Dado un nuevo caso a clasificar, se asigna este nuevo caso a la clase cuyo representante esté más cercano
    - Es como hacer un 1-NN al conjunto con solo un caso por clase

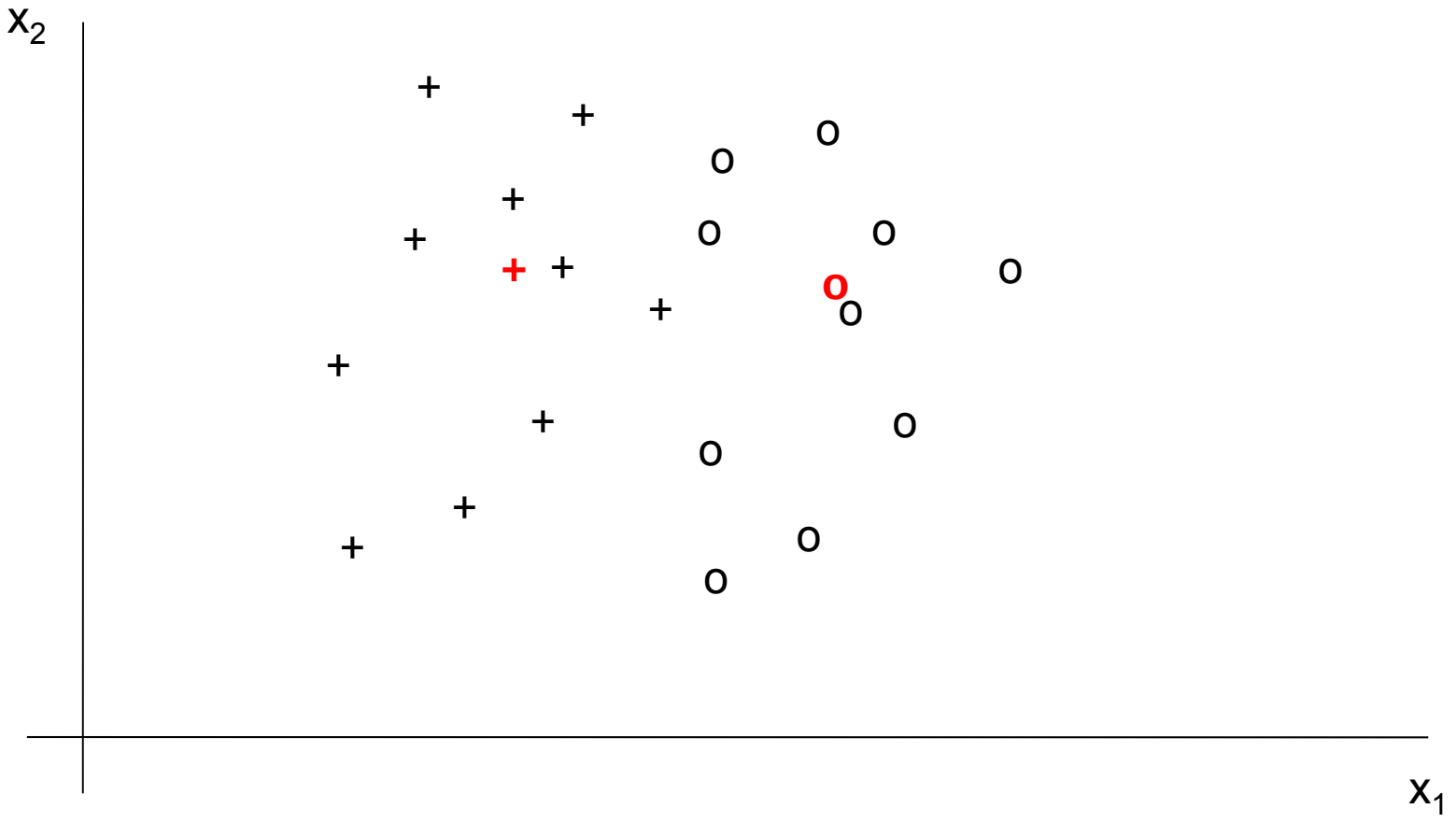
APRENDIZAJE BASADO EN INSTANCIAS

- Variantes: k-NN con distancia mínima:
  - Patrones:



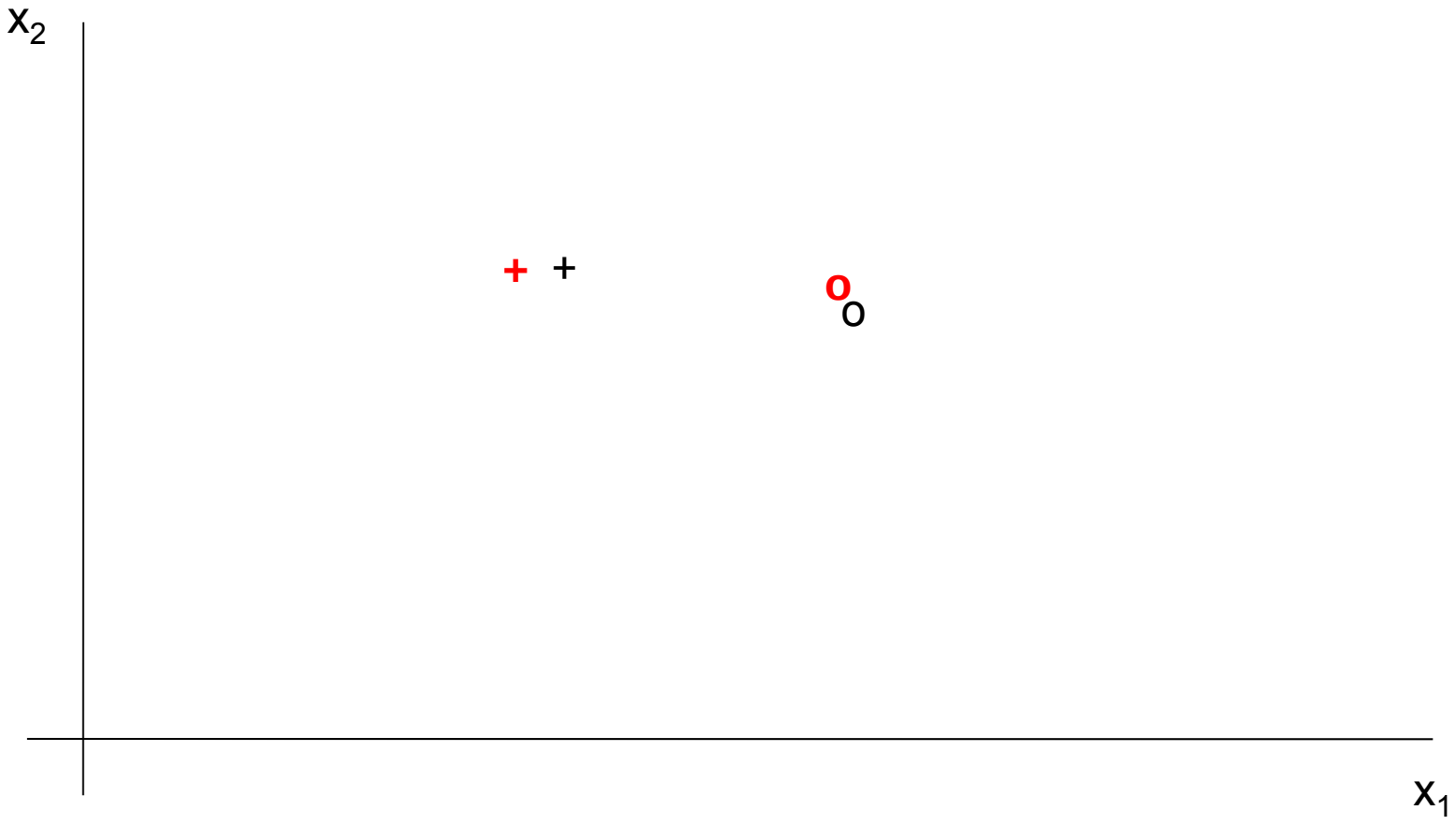
APRENDIZAJE BASADO EN INSTANCIAS

- Variantes: k-NN con distancia mínima:
  - Centroides de cada clase:



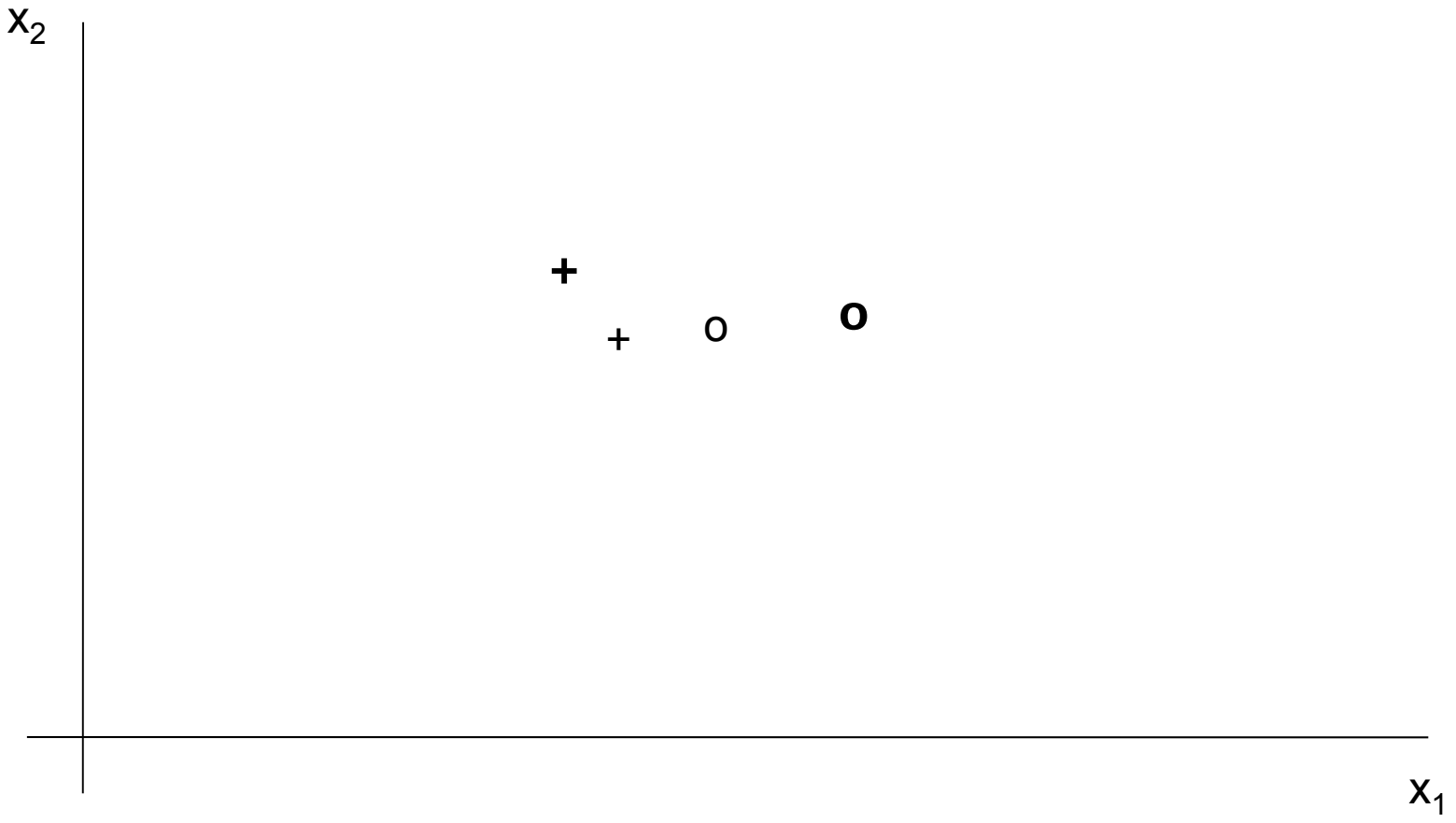
APRENDIZAJE BASADO EN INSTANCIAS

- Variantes: k-NN con distancia mínima:
  - Instancias más cercanas a los centroides:



APRENDIZAJE BASADO EN INSTANCIAS

- Variantes: k-NN con distancia mínima:
  - Clasificación de nuevas instancias:

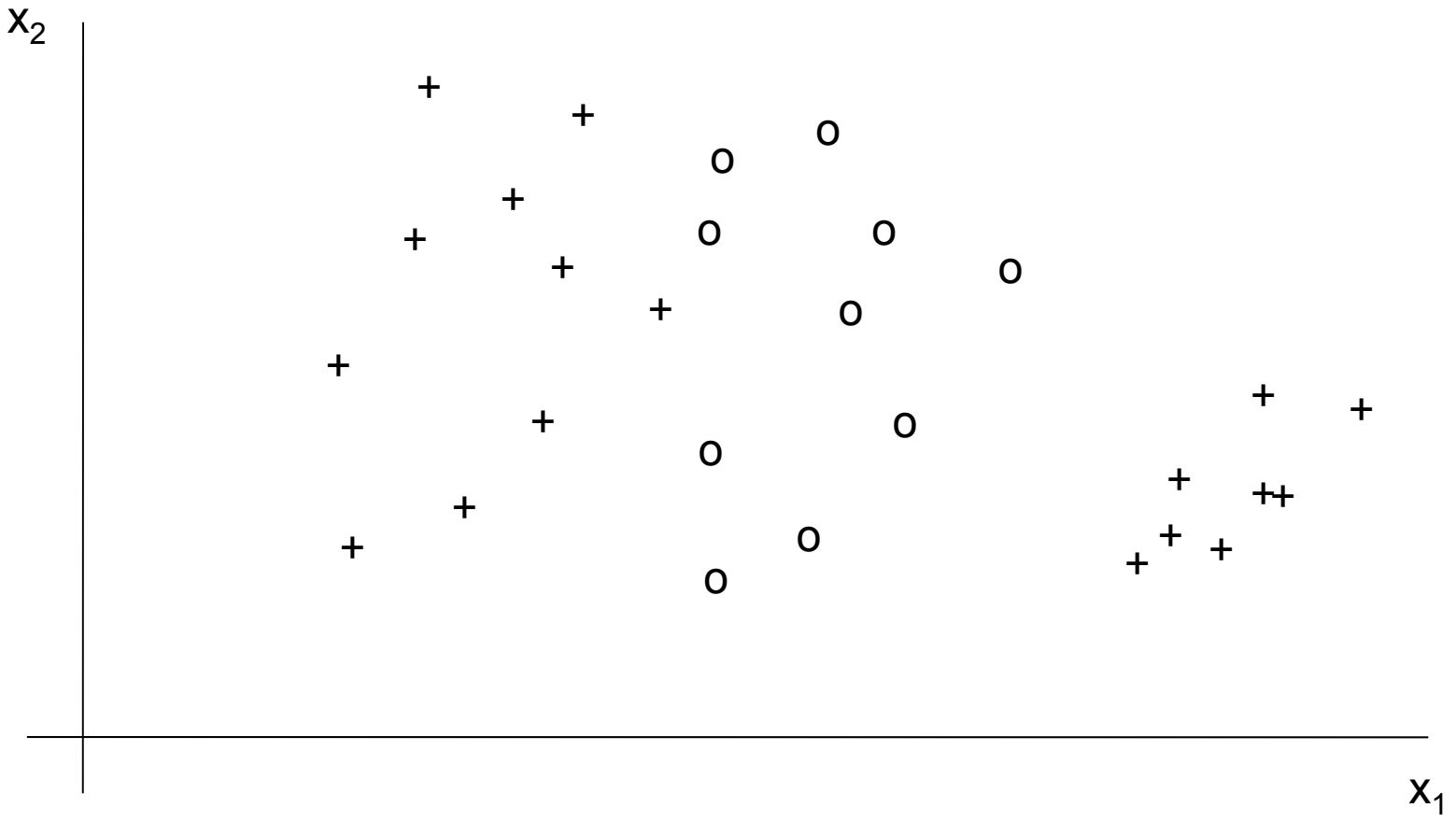


Aprendizaje basado en Instancias

- Variantes: k-NN con distancia mínima:
  - Ventaja:
    - El coste computacional es inferior al k-NN genérico
  - Desventaja:
    - Su efectividad está condicionada a la homogeneidad dentro de las clases
      - Cuanto más homogéneas, más efectivo

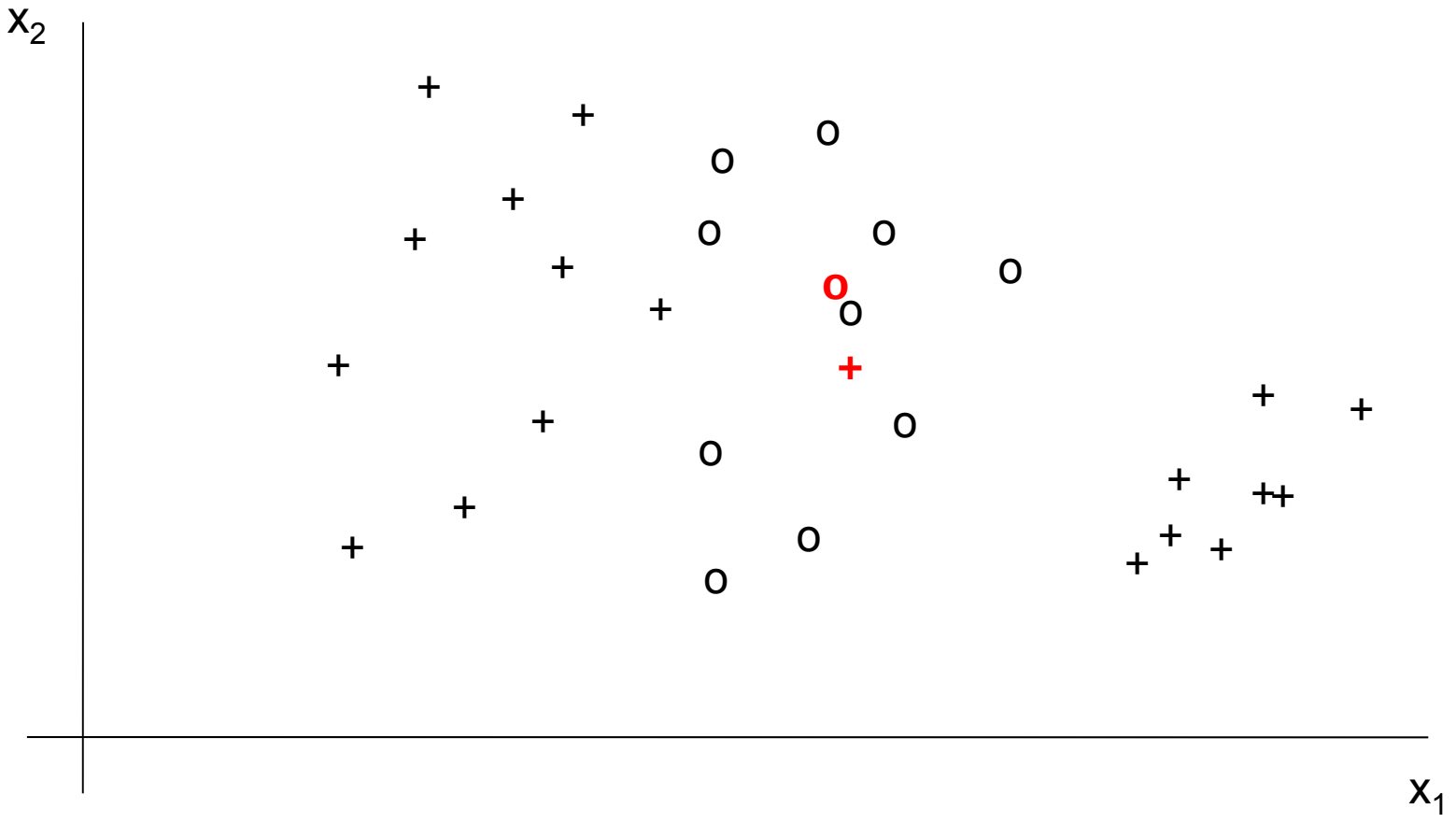
APRENDIZAJE BASADO EN INSTANCIAS

- Variantes: k-NN con distancia mínima:
  - Con conjuntos no homogéneos la técnica fallaría:



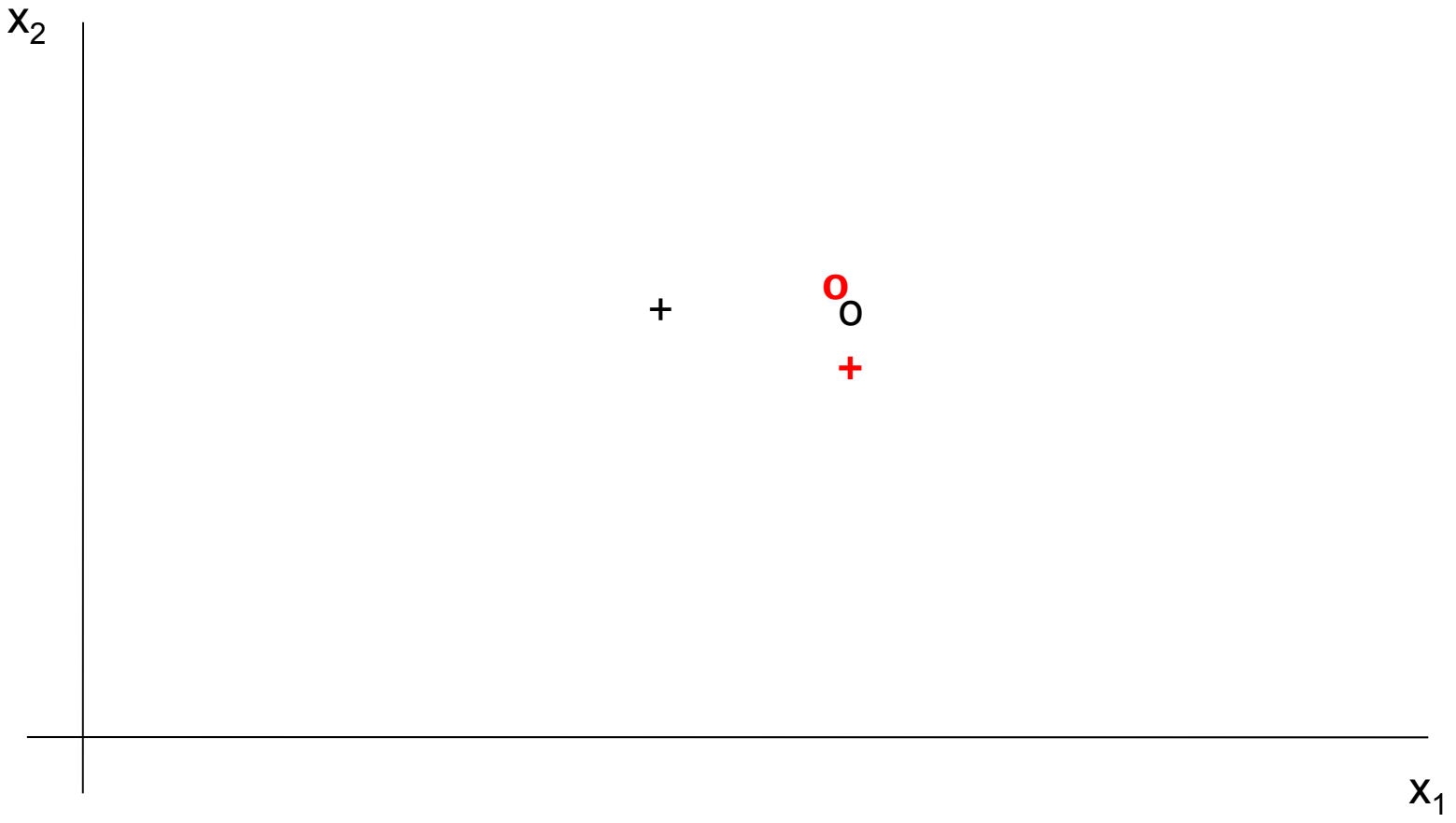
APRENDIZAJE BASADO EN INSTANCIAS

- Variantes: k-NN con distancia mínima:
  - Con conjuntos no homogéneos la técnica fallaría:



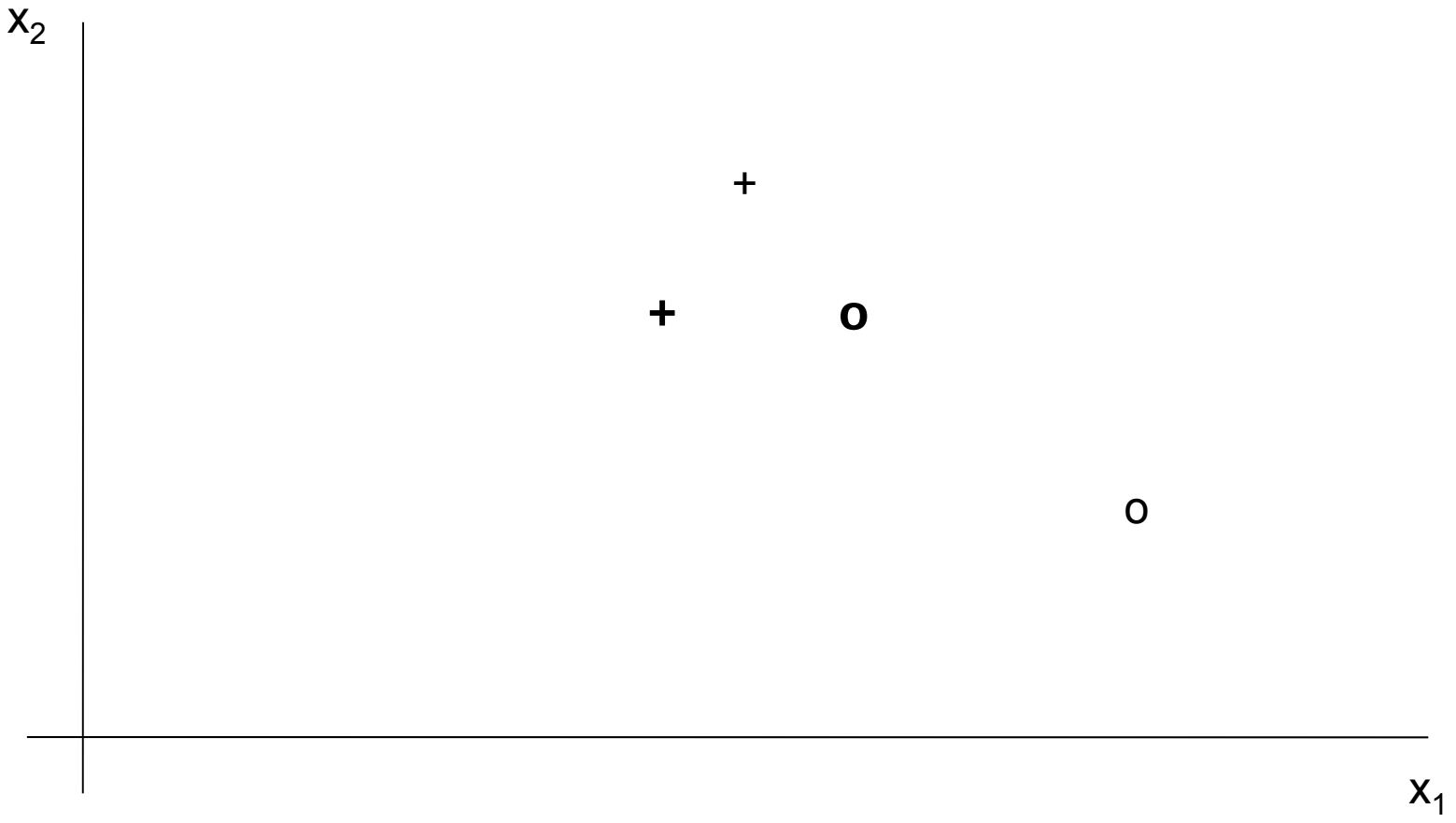
APRENDIZAJE BASADO EN INSTANCIAS

- Variantes: k-NN con distancia mínima:
  - Con conjuntos no homogéneos la técnica fallaría:



APRENDIZAJE BASADO EN INSTANCIAS

- Variantes: k-NN con distancia mínima:
  - Con conjuntos no homogéneos la técnica fallaría:



APRENDIZAJE BASADO EN INSTANCIAS

- Variantes del k-NN:
  - k-NN con rechazo
  - k-NN con distancia media
  - k-NN con ponderación de vecinos
  - k-NN con distancia mínima
  - **k-NN con ponderación de variables**

APRENDIZAJE BASADO EN INSTANCIAS

- Variantes: k-NN con ponderación (pesado) de variables
  - Hasta ahora, el cálculo de las distancias pondera de la misma manera todas las variables
    - ¿Son todos los atributos o características igual de relevantes?
    - ¿Depende esa relevancia de la zona del espacio?
  - Si hay, por ejemplo, 20 variables, y solo dos de ellas son relevantes, instancias que en realidad son muy diferentes pueden estar muy próximas en el espacio
    - "Maldición de las dimensiones"

APRENDIZAJE BASADO EN INSTANCIAS

- Variantes: k-NN con ponderación (pesado) de variables
  - Hasta ahora, el cálculo de las distancias pondera de la misma manera todas las variables
    - Por ejemplo: distancia euclídea:

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^{n} (x_i[r] - x_j[r])^2}$$

n: número de variables

APRENDIZAJE BASADO EN INSTANCIAS

- Variantes: k-NN con ponderación (pesado) de variables
  - La distancia euclídea

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^{n} (x_i[r] - x_j[r])^2}$$

    otorga la misma importancia a todas las variables
    - Puede ser peligroso si hay alguna variable irrelevante
    - Solución: ponderar cada una de las variables
      - Corresponde a modificar el "largo" de los ejes en el espacio
      - Por ejemplo, con la fórmula anterior:

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^{n} w_r (x_i[r] - x_j[r])^2}$$

        w_r asigna un peso a la variable r

APRENDIZAJE BASADO EN INSTANCIAS

- Variantes: k-NN con ponderación (pesado) de variables
  - Ejemplo:
    - Variable x_1 es irrelevante para C
    - Variable x_2 es relevante para C
  - ¿Cómo calcular los términos w_r?
    - Una forma podría ser a partir de la **medida de información mínima** $I(x_i, C)$ entre la variable x_i y la variable de clase C

$$I(X_i, C) = \sum_{x_i, c} p_{(X_i,C)}(x_i, c) \log \frac{p_{(X_i,C)}(x_i, c)}{p_{X_i}(x_i) \cdot p_C(c)}$$

X_1	X_2	C
0	0	1
0	0	1
0	0	1
1	0	1
1	0	1
1	1	1
0	1	0
0	1	0
0	1	0
1	1	0
1	1	0
1	0	0

APRENDIZAJE BASADO EN INSTANCIAS

- Variantes: k-NN con ponderación (pesado) de variables
  - Medida de información mínima entre dos variables
    - Reducción en la incertidumbre sobre una de las variables cuando se conoce el valor de la otra variable
    - Cuanto mayor sea la medida de información mutua entre las variables, mayor será la "dependencia" existente entre las mismas
    - En este caso, se calcula la medida de información mínima entre cada variable y la variable de clase
      - El peso w_r asociado a la variable x_r será proporcional a la medida de información mutua $I(x_r, C)$

Aprendizaje basado en Instancias

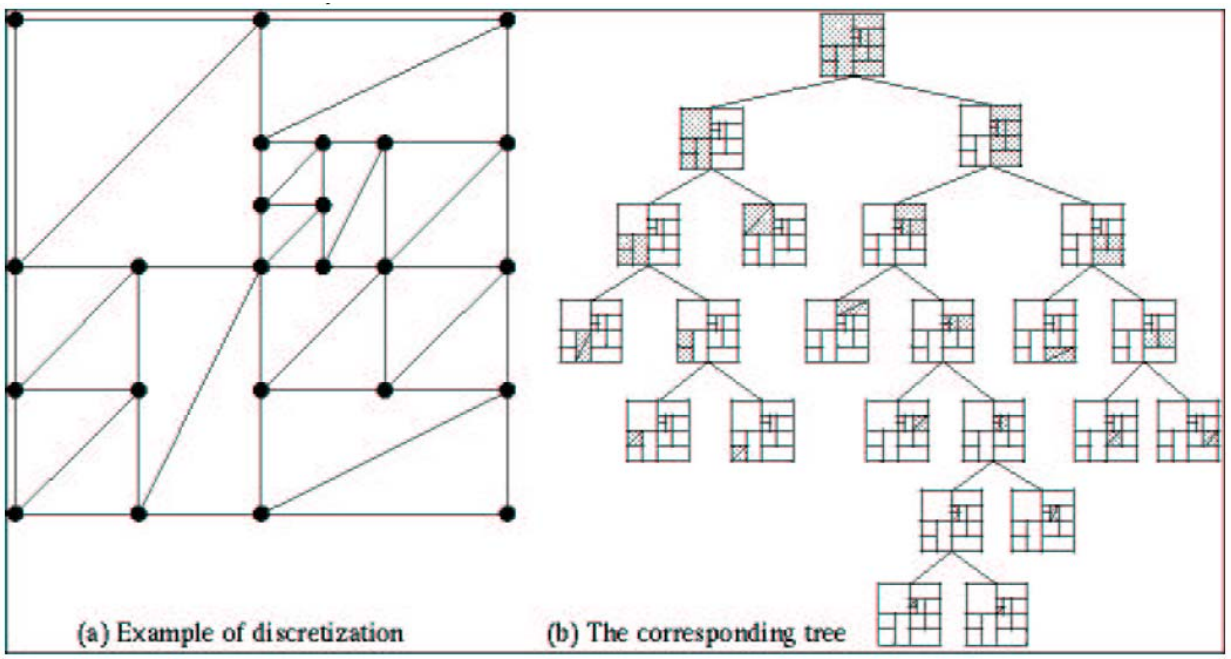
- Siempre hay problemas cuando el conjunto de instancias es muy grande:
  - Problemas de almacenamiento
  - Problemas de cálculo de vecinos
- Soluciones:
  - Indexación
  - Selección de instancias
  - Reemplazo de instancias

APRENDIZAJE BASADO EN INSTANCIAS

- Siempre hay problemas cuando el conjunto de instancias es muy grande:
  - Problemas de almacenamiento
  - Problemas de cálculo de vecinos
- Soluciones:
  - **Indexación**
  - Selección de instancias
  - Reemplazo de instancias

APRENDIZAJE BASADO EN INSTANCIAS

- Soluciones: Indexación:
  - Árboles KD (*Locally Weighted Regression*)
    - A.W. Moore



  - Agrupación o clustering
    - Aprendizaje no supervisado

Aprendizaje basado en Instancias

- Siempre hay problemas cuando el conjunto de instancias es muy grande:
  - Problemas de almacenamiento
  - Problemas de cálculo de vecinos
- Soluciones:
  - Indexación
  - **Selección de instancias**
  - Reemplazo de instancias

APRENDIZAJE BASADO EN INSTANCIAS

- Soluciones: Selección de instancias:
  - Elegir un grupo reducido de instancias (prototipos) (S) que mantengan la misma información que el conjunto total (T)
  - Métodos:
    - Incremental:
      - Comenzar con un conjunto S de prototipos vacío
      - Ir añadiendo instancias al conjunto S a partir de las instancias en T, siempre y cuando cumplan un determinado criterio
        - Por ejemplo, cuando al intentar clasificarlo, se clasifica de forma distinta a su clase
          - Condensación de Hart

Aprendizaje basado en Instancias

- Soluciones: Selección de instancias:
  - Elegir un grupo reducido de instancias (prototipos) (S) que mantengan la misma información que el conjunto total (T)
  - Métodos:
    - Decremental:
      - Comenzar con un conjunto S=T de prototipos
      - Ir eliminando instancias al conjunto S, siempre y cuando cumplan un determinado criterio
        - Por ejemplo, que al extraerlo del conjunto, se sigue clasificando correctamente
          - Edición de Wilson

Aprendizaje basado en Instancias

- Siempre hay problemas cuando el conjunto de instancias es muy grande:
  - Problemas de almacenamiento
  - Problemas de cálculo de vecinos
- Soluciones:
  - Indexación
  - Selección de instancias
  - **Reemplazo de instancias**

APRENDIZAJE BASADO EN INSTANCIAS

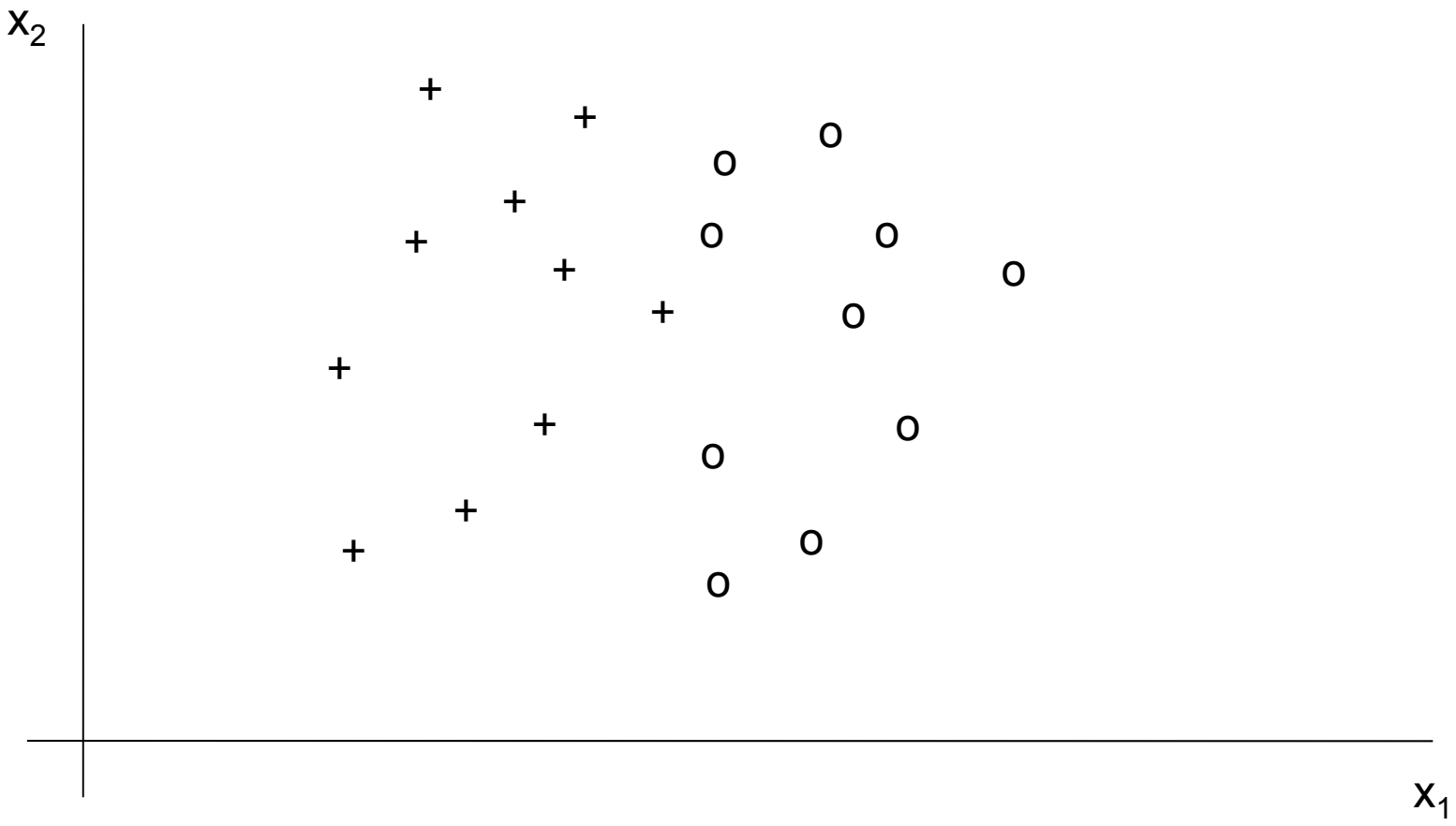
- Soluciones: Reemplazo de instancias:
  - Calcular **prototipos** a partir del conjunto de entrenamiento
  - Normalmente 1-NN
  - *Learning Vector Quantization* (LVQ):
    - Comenzar con un conjunto de prototipos $S = \{s_1,\ldots,s_M\}$
    - Repetir:
      - Elegir una nueva instancia x
      - Obtener el prototipo más cercano de S, $s = \text{argmin}_i(d(x,s_i))$
      - Actualizar la posición de s_i:

        $s_i = s_i + \alpha[x - s_i]$ si s_i y x pertenecen a la misma clase

        $s_i = s_i - \alpha[x - s_i]$ si s_i y x pertenecen distinta clase
  - Problema de LVQ: Definición del número de prototipos a utilizar

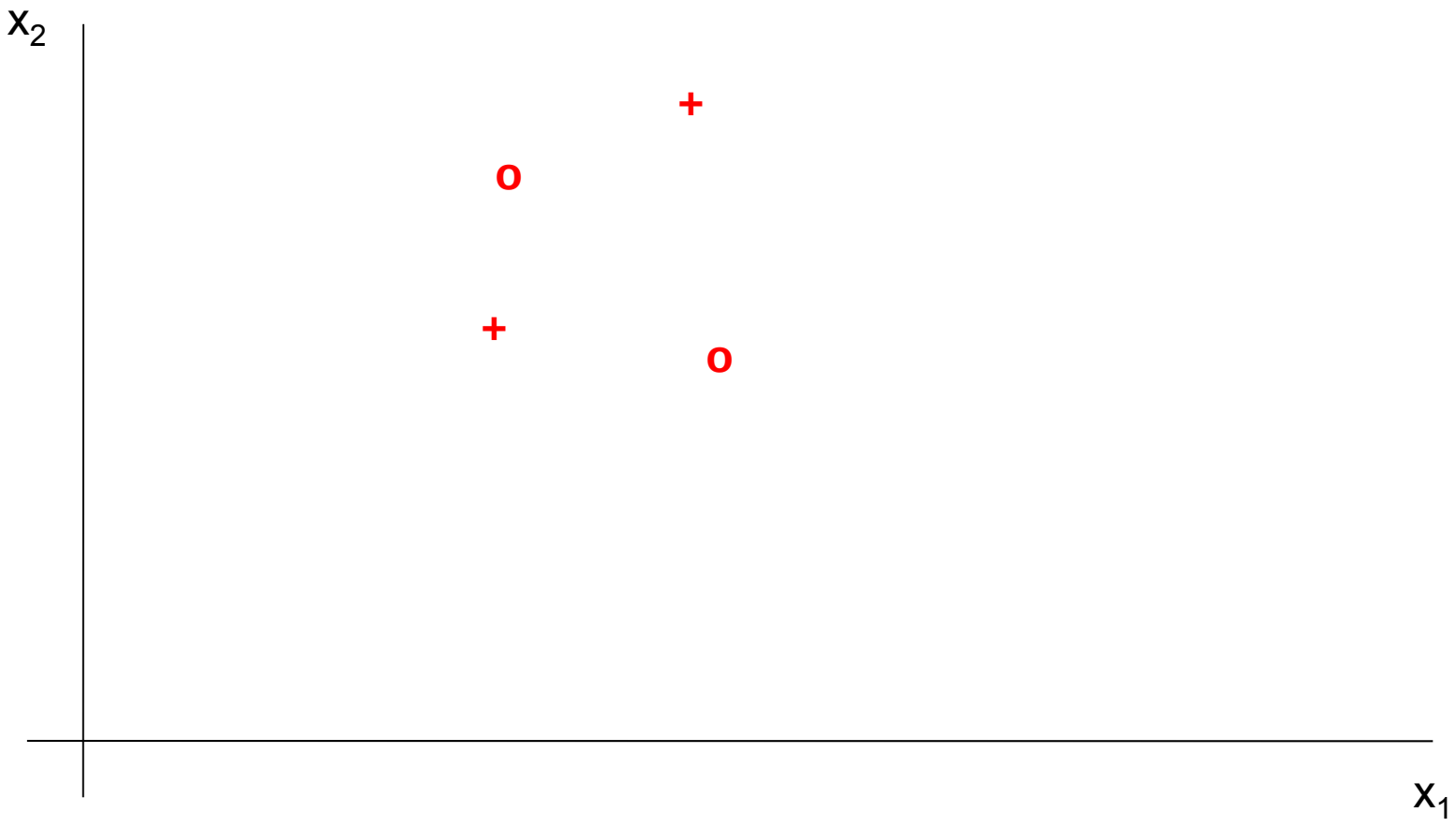
APRENDIZAJE BASADO EN INSTANCIAS

- Soluciones: Reemplazo de instancias:
  - Patrones:



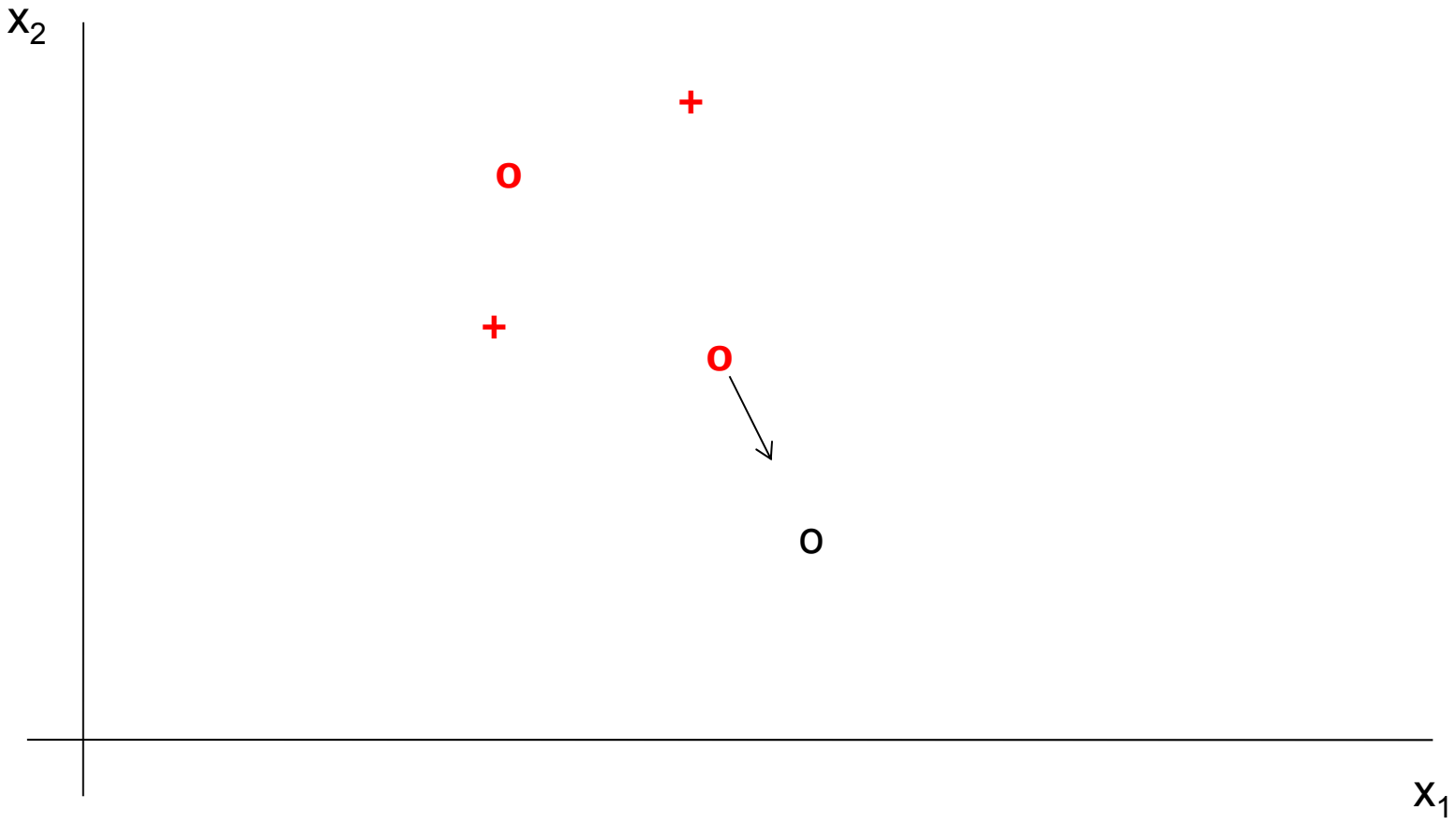
APRENDIZAJE BASADO EN INSTANCIAS

- Soluciones: Reemplazo de instancias:
  - Prototipos:



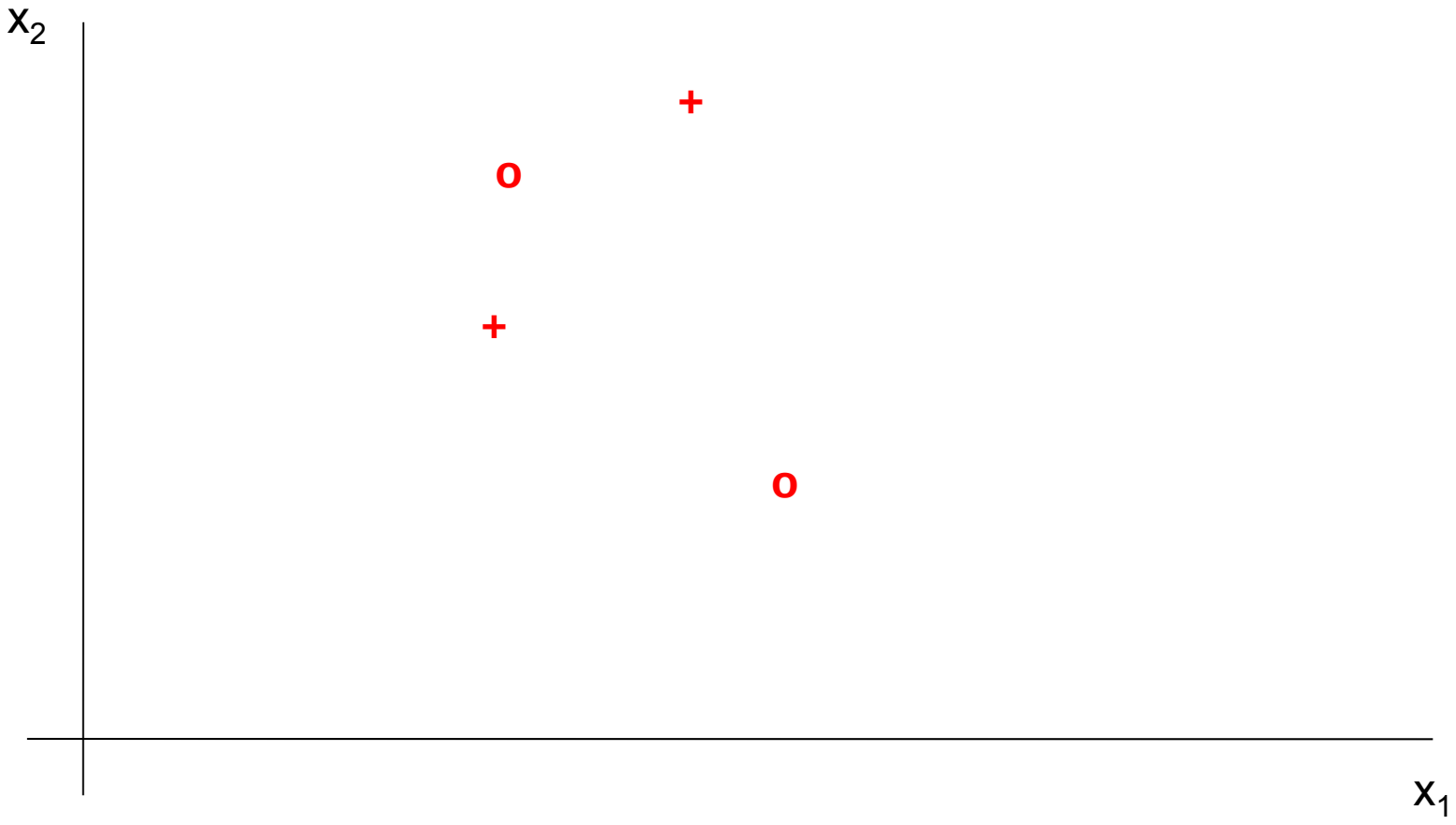
APRENDIZAJE BASADO EN INSTANCIAS

- Soluciones: Reemplazo de instancias:
  - Ante esta instancia, se acerca el prototipo más cercano:



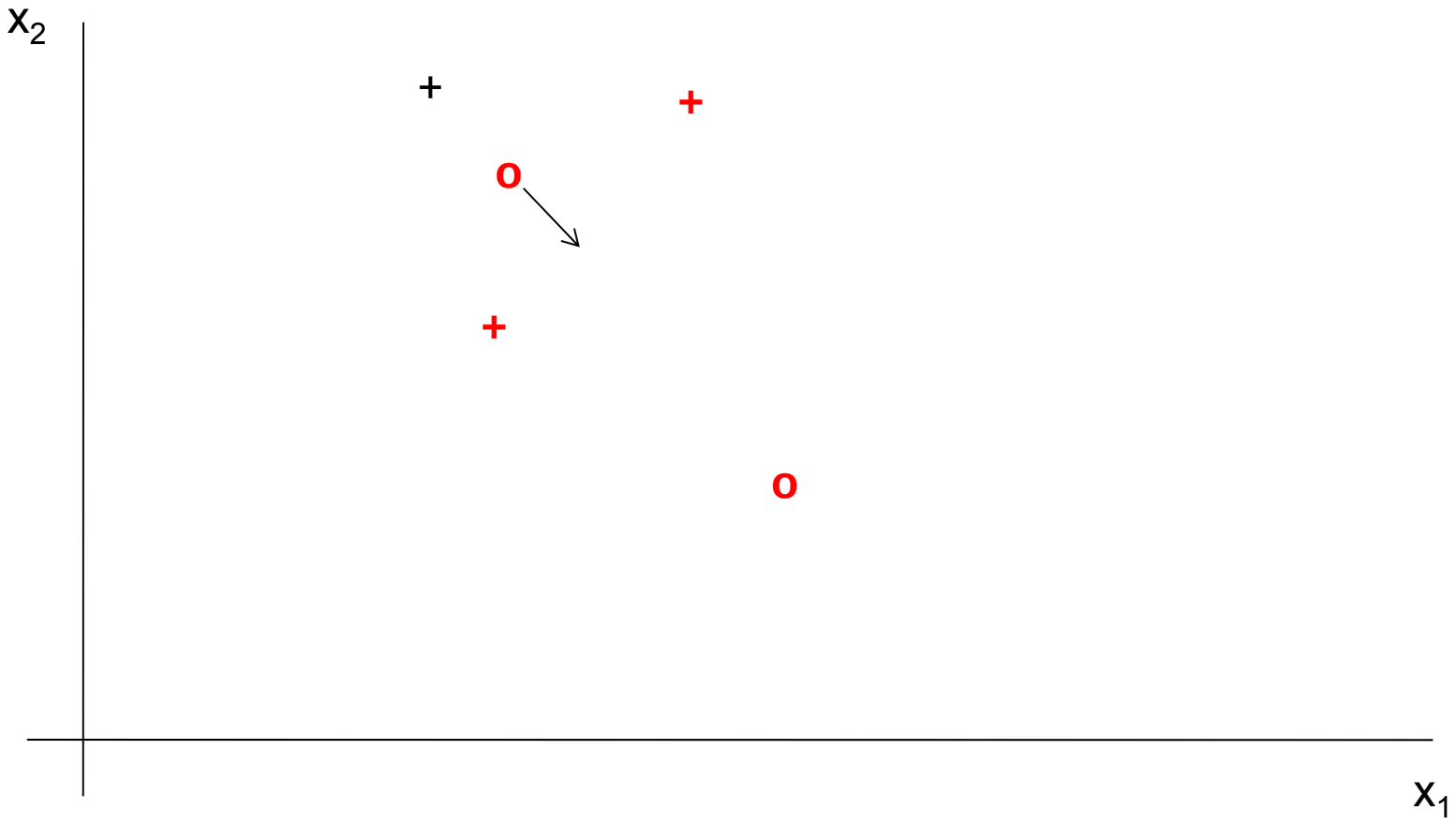
APRENDIZAJE BASADO EN INSTANCIAS

- Soluciones: Reemplazo de instancias:



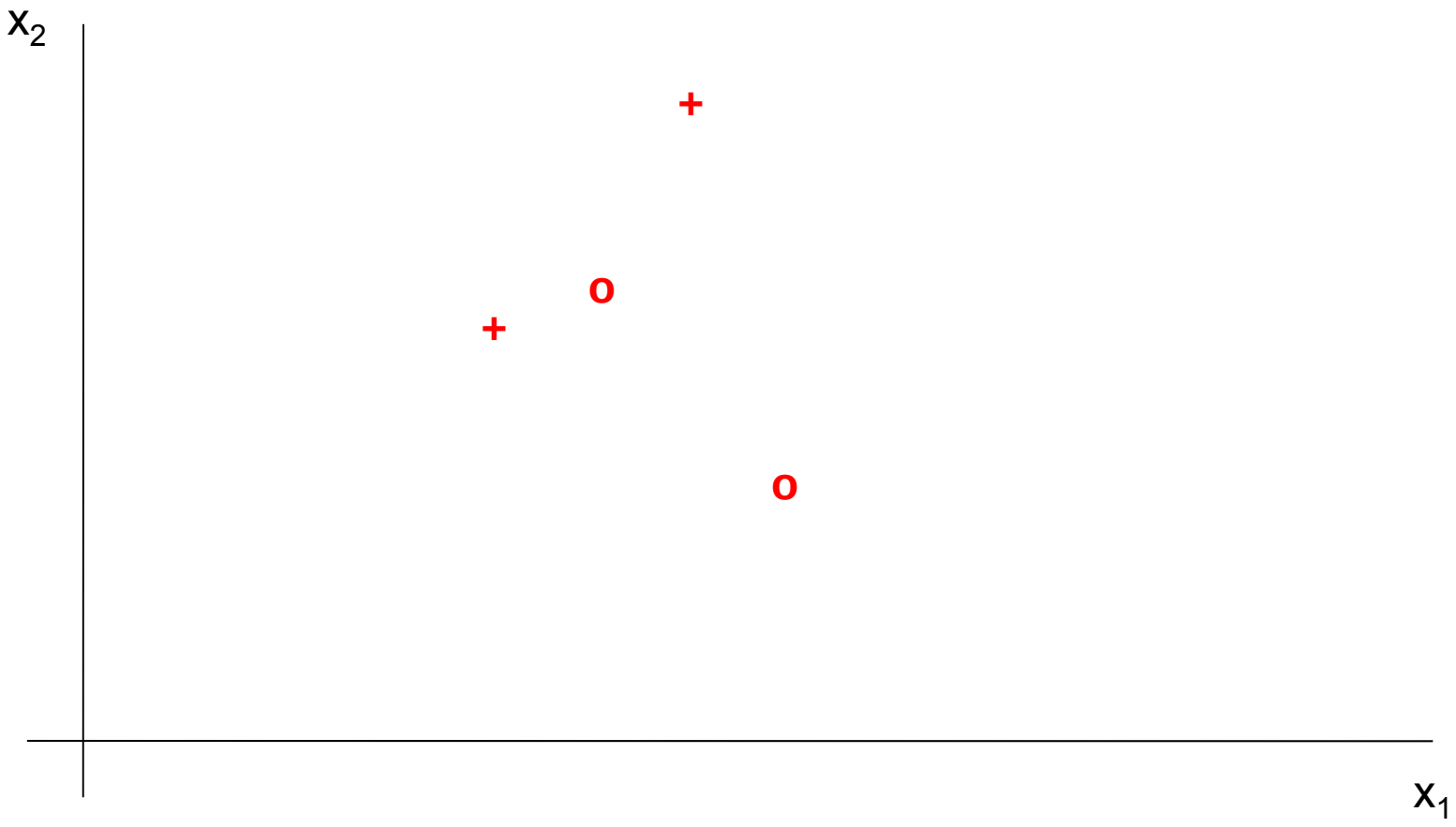
APRENDIZAJE BASADO EN INSTANCIAS

- Soluciones: Reemplazo de instancias:
  - Ante esta instancia, se aleja el prototipo más cercano:



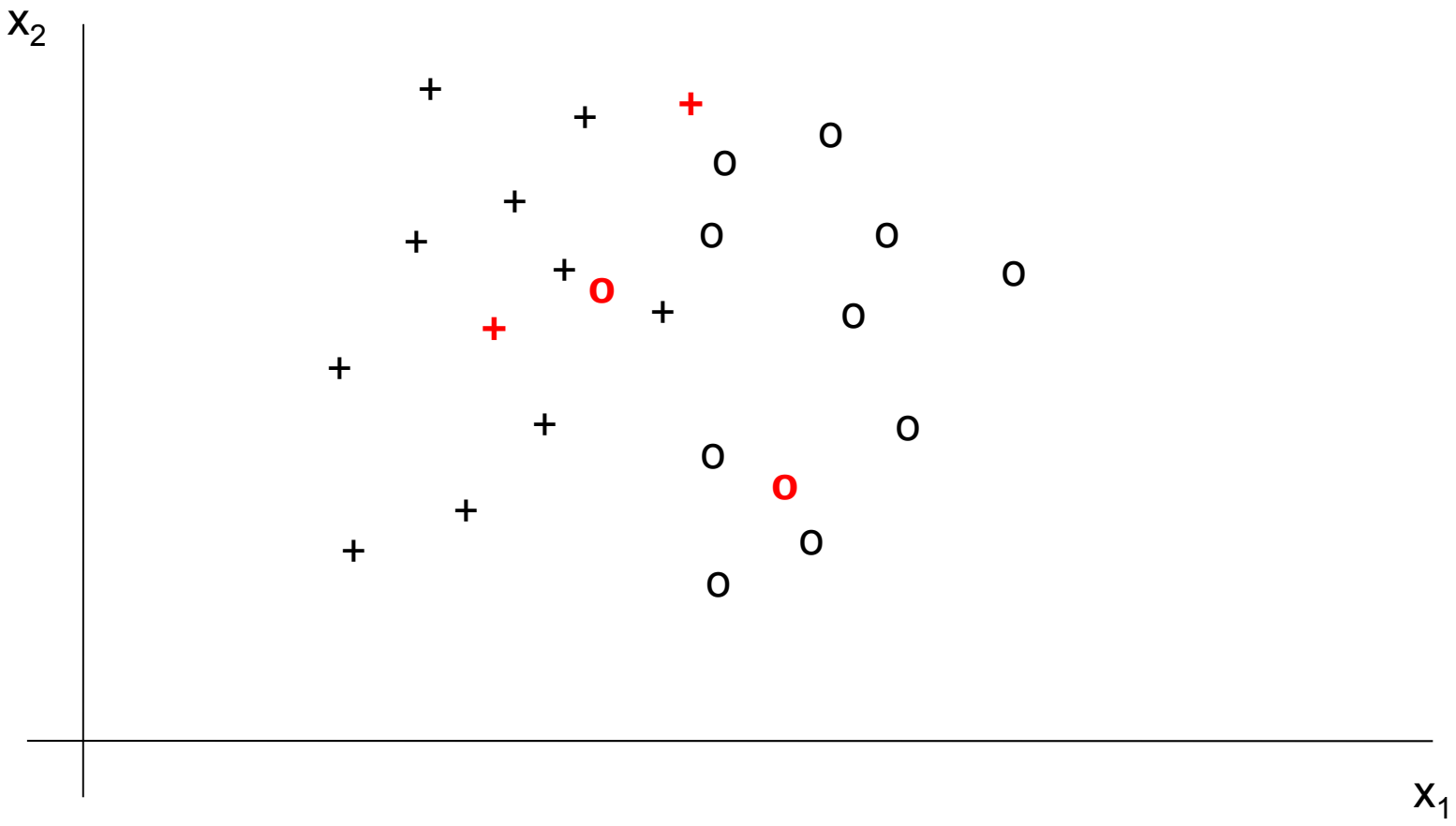
APRENDIZAJE BASADO EN INSTANCIAS

- Soluciones: Reemplazo de instancias:



APRENDIZAJE BASADO EN INSTANCIAS

- Soluciones: Reemplazo de instancias:
  - Patrones junto con los prototipos movidos con esas dos:



Aprendizaje basado en Instancias

- Selección de características
  - ¿Son todos los atributos o características relevantes para el problema de clasificación?
  - Selección de características:
    - Determinar qué características son las interesantes, y eliminar las restantes
    - En los árboles de decisión esto está resuelto por el propio mecanismo de construcción de los árboles
  - Soluciones: Métodos estadísticos, algoritmos genéticos, …
  - Fases:
    - Selección de características
    - Limpieza de datos
    - k-NN

Aprendizaje basado en Instancias

- Métodos relacionados:
  - Regresión local ponderada
    - *Locally Weighted Regression*
  - Razonamiento basado en casos
    - *Case-Based Reasoning*

APRENDIZAJE BASADO EN INSTANCIAS

- Métodos relacionados:
  - **Regresión local ponderada**
    - *Locally Weighted Regression*
  - Razonamiento basado en casos
    - *Case-Based Reasoning*

Aprendizaje basado en Instancias

- Métodos relacionados:
  - Regresión local ponderada
    - Usar kNN para realizar regresión
    - Basado en la construcción de modelos lineales de forma local en zonas del espacio
    - Construcción del modelo cuando se realiza la consulta de un nuevo caso
      - Almacenamiento previo de todos los casos atributos-valor
    - Ante una nueva consulta, se obtienen los k vecinos
      - De cada vecino, se toma su valor

Aprendizaje basado en Instancias

- Métodos relacionados:
  - Regresión local ponderada
    - A partir de los valores de los vecinos, se construye un modelo de regresión lineal:

$$\hat{f}(x) = a_0 + a_1 x[1] + \ldots + a_n x[n]$$

    - Se devuelve esa salida y se elimina el modelo
      - Ante una nueva instancia, se construye otro modelo
    - Aplicable con cualquier otro método de aproximación:
      - Modelo no lineal: cuadrático, cúbico, etc.
      - Redes de neuronas
      - etc.

APRENDIZAJE BASADO EN INSTANCIAS

- Métodos relacionados:
  - Regresión local ponderada
    - *Locally Weighted Regression*
  - **Razonamiento basado en casos**
    - *Case-Based Reasoning*

APRENDIZAJE BASADO EN INSTANCIAS

- Métodos relacionados:
  - Razonamiento basado en casos
    - ¿Qué sucede cuando las instancias son representadas de forma más compleja?
    - Ejemplo:
      - Transporte:<ómnibus>
      - Tiempo:<lluvioso>
      - Predicción:<dia+1: nublado, dia+2=soleado, dia+3=?>
      - Lugar:<casa[cuartos=2, sin piscina], centro=lejos>
      - Personas:<adultos[hombres=1, mujeres=1], niños=4>
      - Satisfacción: ????

APRENDIZAJE BASADO EN INSTANCIAS

- Métodos relacionados:
  - Razonamiento basado en casos
    - CBR (*Case-Based Reasoning*)
    - Es una técnica de Inteligencia Artificial que se basa en la utilización de experiencias previas para resolver nuevos problemas mediante la hipótesis:
      - Problemas similares tienen soluciones similares.
      - Típico en aprendizaje humano
        - Aprendizaje por analogía
          - Tema 1

Aprendizaje basado en Instancias

- Métodos relacionados:
  - Razonamiento basado en casos
    - Dado un problema a resolver, el CBR busca, en una base de datos llamada Base de Casos, problemas similares que anteriormente se hayan resuelto con éxito, llamados casos, y adapta las soluciones para dar una solución al problema actual.
    - Este mecanismo de razonamiento es utilizado por los humanos en múltiples problemas y permite que sea un sistema de fácil comprensión.

APRENDIZAJE BASADO EN INSTANCIAS

- Métodos relacionados:
  - Razonamiento basado en casos
    - El CBR involucra toda una metodología con un ciclo de actividades que además de solucionar nuevos problemas nos permita aprender de las buenas soluciones obtenidas por los nuevos problemas:
      - Recuperar
      - Reutilizar
      - Revisar
      - Retener

Aprendizaje basado en Instancias

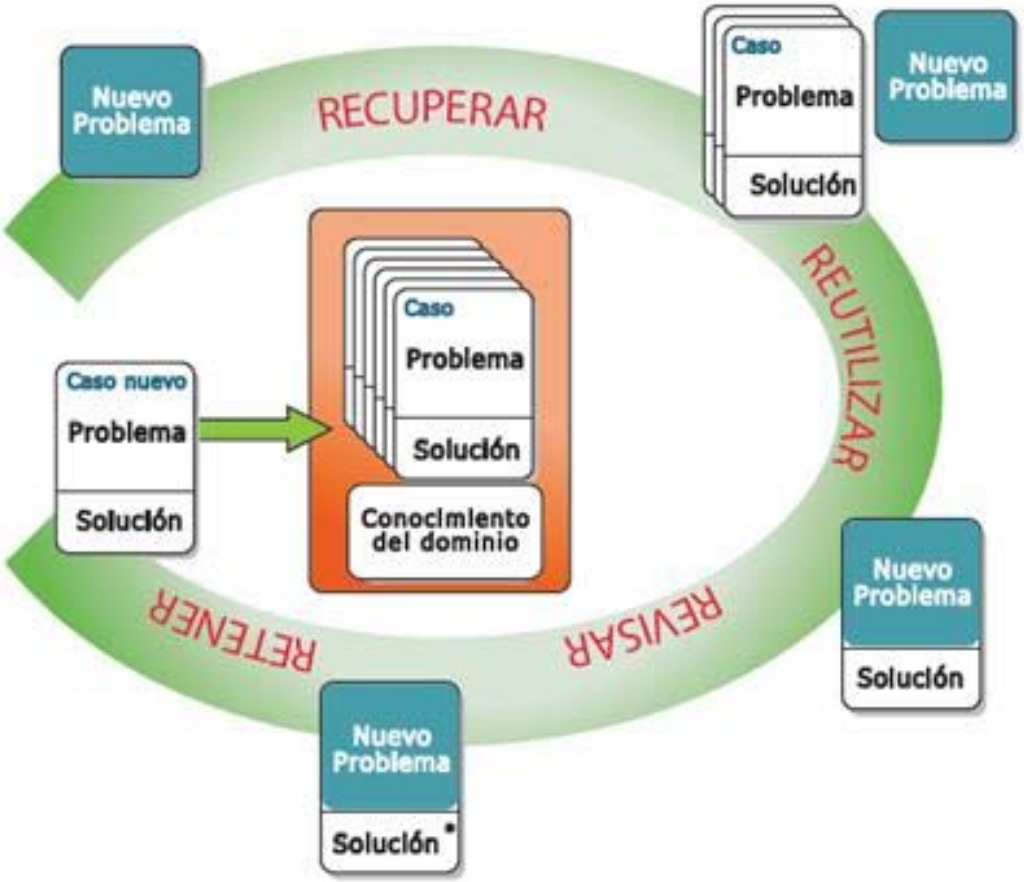
- Métodos relacionados:
  - Razonamiento basado en casos
    - Recuperar (*retrieve*):
      - Dado un problema, se recuperan los casos más similares de la Base de Casos
        - Un caso es un problema anterior con su solución.
    - Reutilizar (*reuse*):
      - Extraer la solución del caso seleccionado para utilizarla
      - Esto puede implicar adaptar la solución a la nueva situación.

Aprendizaje basado en Instancias

- Métodos relacionados:
  - Razonamiento basado en casos
    - Revisar (*revise*):
      - Se debe analizar si la nueva solución es aceptable y si es necesario revisarla.
      - Si bien se suele entender que en el proceso de reutilización se lleva a cabo toda la problemática de adaptación del caso ó casos recuperados para el nuevo problema, en muchas aplicaciones prácticas las fases de reutilización y revisión apenas se distinguen, y muchos investigadores hablan de fase de **adaptación**, que combina ambas.
    - Retener (*retain*):
      - Después de haber aplicado la solución con éxito, se debe almacenar la experiencia como un nuevo caso en la Base de Casos.

APRENDIZAJE BASADO EN INSTANCIAS

- Métodos relacionados:
  - Razonamiento basado en casos

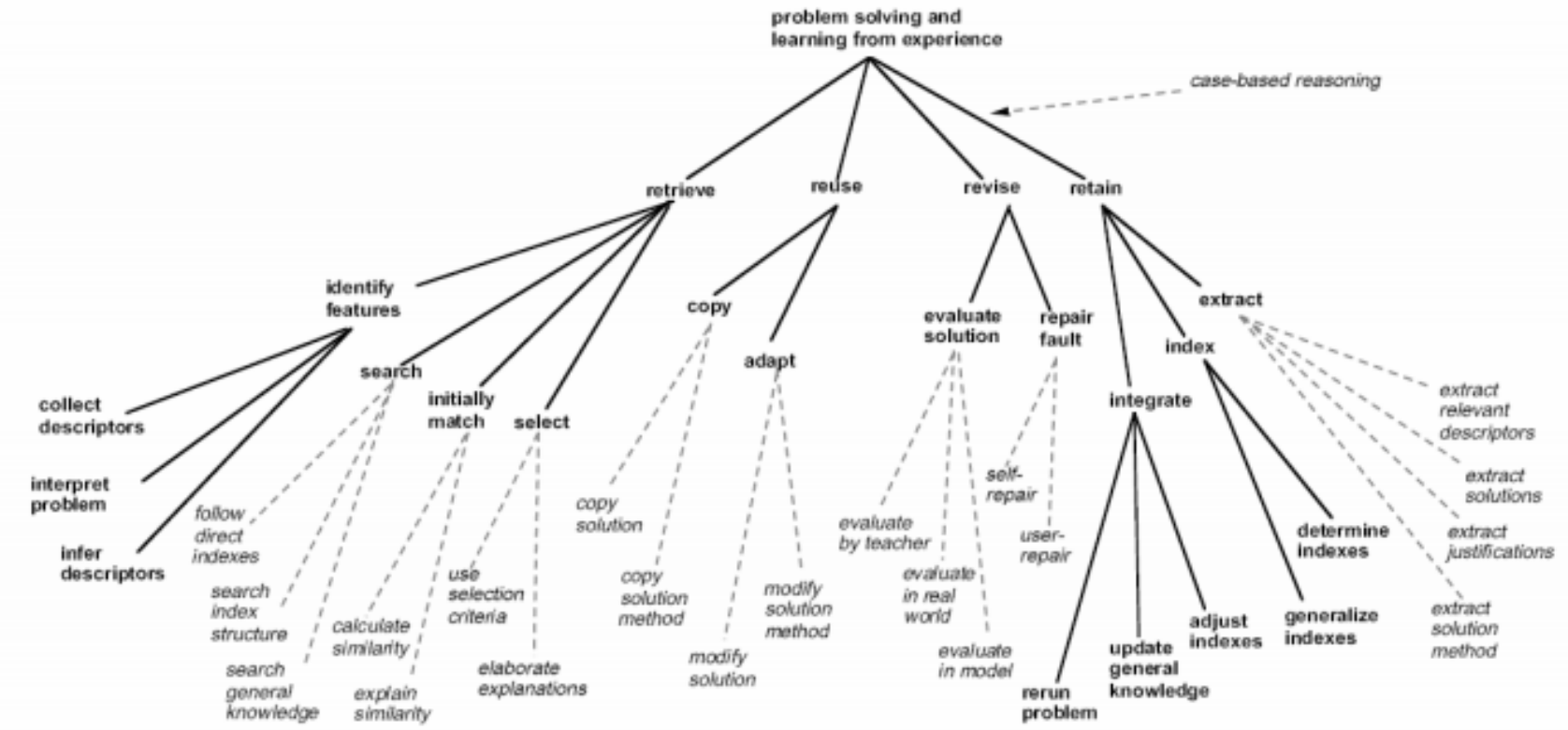


Aprendizaje basado en Instancias

- Métodos relacionados:
  - Razonamiento basado en casos
    - Los cuatro procesos no son tareas únicas, es decir, cada uno de ellos implica llevar a cabo una serie de tareas más específicas.
      - Jerarquía de tareas

APRENDIZAJE BASADO EN INSTANCIAS

- Métodos relacionados:
  - Razonamiento basado en casos



Aprendizaje basado en Instancias

- Métodos relacionados:
  - Razonamiento basado en casos
    - Basado en los mismos principios que kNN
      - Se clasifica una instancia en base a casos parecidos
      - La diferencia es que, en lugar de utilizar puntos en un espacio euclídeo, representamos las instancias con atributos más complejos
    - Se debe buscar una métrica de similitud que depende del dominio de trabajo
    - La solución se basa en combinaciones complejas y específicas al dominio de aplicación

Aprendizaje basado en Instancias

- Métodos relacionados:
  - Razonamiento basado en casos
    - Representaciones de la información mucho más desarrolladas:
      - Imágenes
      - Documentos
      - Planes
      - etc.

Aprendizaje basado en Instancias

- Paradigma que puede usarse en otro tipo de espacios:
  - En lugar de instancias representadas en el espacio euclídeo
    - Representaciones más complejas
  - Por ejemplo: clasificación de imágenes:
    - Base de datos con imágenes de distintas clases
    - Mediante la creación de un modelo (RNA,SVM,etc):
      - Extraer características de las imágenes
      - Crear base de datos con esas características y las clases
      - Ajustar un modelo de aprendizaje máquina
      - Ante una nueva instancia, extraer características y aplicar el modelo

Aprendizaje basado en Instancias

- Paradigma que puede usarse en otro tipo de espacios:
  - En lugar de instancias representadas en el espacio euclídeo
    - Representaciones más complejas
  - Por ejemplo: clasificación de imágenes:
    - Base de datos con imágenes de distintas clases
    - Mediante el paradigma del aprendizaje basado en instancias:
      - Desarrollar una función que, entre dos imágenes, de una medida de similitud entre las mismas
      - Ante una nueva imagen, calcular las más cercanas y a partir de ellas, calcular la clase de pertenencia

APRENDIZAJE BASADO EN INSTANCIAS

- Algoritmos perezosos (*lazy*) vs. voraces (o ansiosos, ávidos, etc.) (*greedy*)
  - Los algoritmos perezosos [KNN, Regresión Local...] retrasan el cálculo de una hipótesis hasta la llegada de una nueva consulta.
    - Computan una aproximación local de la función objetivo para responder cada nueva consulta.
    - En otras palabras, utilizan múltiples aproximaciones locales para modelar la función objetivo [global].
  - Los algoritmos voraces pueden utilizar también aproximaciones; sin embargo, éstas quedan "fijas" al conjunto de entrenamiento.
    - Se elabora un modelo
  - Dado un mismo espacio de hipótesis, los algoritmos perezosos tienen un mayor poder de adaptación a una nueva consulta.

Aprendizaje basado en Instancias

- Resumen:
  - Los métodos basados en casos posponen la creación de una hipótesis hasta el momento de una nueva clasificación.
  - Esto les permite generar una aproximación **local** para cada una de las nuevas instancias, aproximando el objetivo con funciones complejas.
  - Desventajas:
    - El costo de cálculo de estas aproximaciones
    - Definición de una métrica apropiada
    - Almacenamiento