

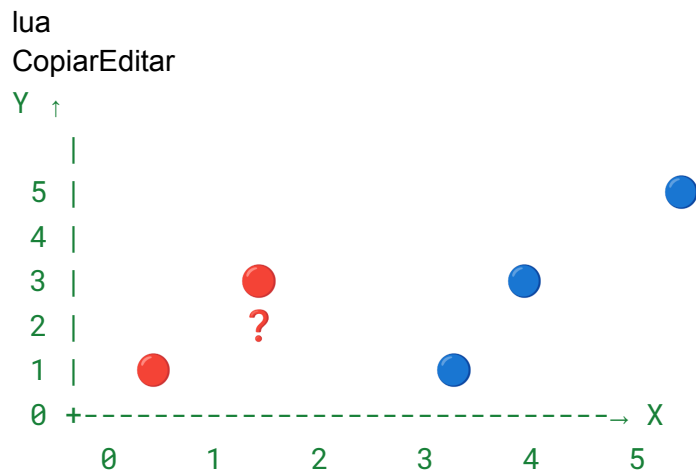





Imagina que estás viendo un plano de coordenadas (como una hoja cuadriculada). Tienes dos tipos de puntos:

-  **Clase A (Rojo)**
-  **Clase B (Azul)**

Ahora imagina este plano:



- Tienes 3 puntos  rojos (Clase A) y 3 puntos  azules (Clase B).
- El punto  en (1,2) es el que quieres clasificar. ¿Será rojo o azul?

---




### Paso 1: Elegir $k=3$

Vamos a ver los **3 vecinos más cercanos** al punto  (1,2).

---

### Paso 2: Calcular distancias (a simple vista)

Los vecinos más cercanos al punto (1,2) son:

-  en (1,1) → distancia pequeña
-  en (0,1) → distancia un poco más
-  en (2,1) → distancia similar

- Los otros puntos están más lejos (como el de arriba en (4,5), etc.)





### Paso 3: Contar vecinos por clase

Entre los 3 más cercanos:



-  Rojo (1,1)
-  Rojo (0,1)
-  Azul (2,1)

Votación:

-  Clase A: 2 votos
-  Clase B: 1 voto



### Paso 4: Resultado

Como **ganó la clase Roja** () , el punto  se clasifica como **Clase A (Rojo)**.

NOTAS

---



### k-NN no necesita entrenamiento

- k-NN es un algoritmo **"lazy learner"** (aprendiz perezoso), lo que significa que **no construye un modelo explícito ni ajusta parámetros durante una fase de entrenamiento**.
- En lugar de eso, **almacena todos los datos de entrenamiento y realiza los cálculos de predicción en el momento** que se consulta (es decir, al hacer una predicción).



### Entonces, ¿cuándo "converge" k-NN?

Podemos hablar de "convergencia" en dos sentidos muy específicos:

---

## 1. Convergencia del error al aumentar el tamaño del conjunto de entrenamiento

- A medida que aumentamos la cantidad de datos, el rendimiento del modelo (ej. tasa de error) tiende a **estabilizarse**.
- Si los datos están bien distribuidos y  $k$  es elegido apropiadamente, el modelo se acerca a una clasificación más robusta.
- Esto **no es una convergencia del algoritmo**, sino **una convergencia del desempeño**.

---

## 2. Convergencia empírica para elegir el mejor $k$

- Cuando se prueba con varios valores de  $k$  (usualmente mediante validación cruzada), se puede observar que a partir de cierto punto **el rendimiento se estabiliza**, indicando un valor óptimo.
- De nuevo, esto **no es convergencia de un proceso iterativo**, sino una **optimización de hiperparámetro**.

---

El algoritmo k-NN **no converge en el sentido clásico** porque **no entrena**. La única "convergencia" observable es la **estabilización del rendimiento** al aumentar los datos o ajustar el valor de  $k$ .