UNIVERSITAS Miguel Hernández

Tema 3. k-Nearest Neighbour (kNN).

José L. Sainz-Pardo Auñón

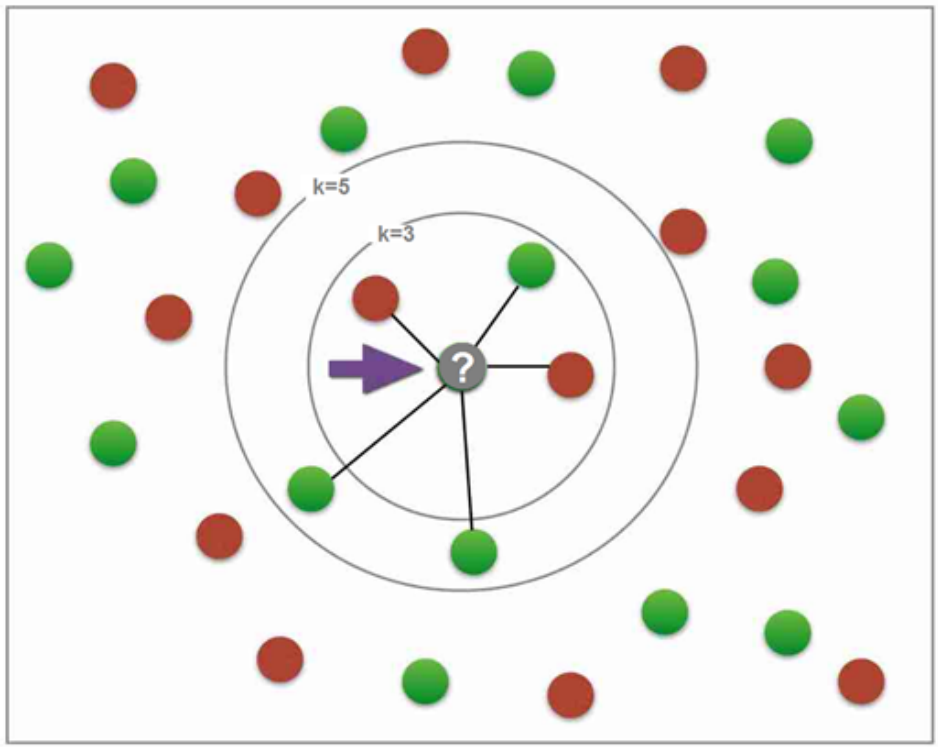
TÉCNICAS ESTADÍSTICAS PARA EL APRENDIZAJE II

Máster Universitario en Estadística Computacional y Ciencia de Datos para la Toma de Decisiones.

Índice

¿Qué es kNN?

- k-Nearest Neighbors (kNN) es un algoritmo basado en la semejanza de las instancias cercanas.
- Puede usarse tanto para problemas de clasificación como de regresión.
- Es un algoritmo **no paramétrico**, lo que significa que no asume una distribución previa de los datos.



Idea básica

Clasificación

Para clasificar un nuevo punto, el algoritmo busca los k puntos más cercanos en el conjunto de entrenamiento y decide la clase más común entre ellos.

Regresión

En problemas de regresión, se utiliza el promedio de los valores de los k vecinos más cercanos.

Medidas de Distancia

El concepto de cercanía entre los puntos es crucial en kNN, y se mide con diferentes métricas de distancia:

- **Distancia Euclidiana:**

$$d(p,q) = \sqrt{\sum_{i=1}^{n}(p_i - q_i)^2}$$

- **Distancia de Chebyshev:**

$$d_{p,q} = \max_{i=1}^{n} |x_i - y_i|$$

- **Distancia Manhattan:**

$$d(p,q) = \sum_{i=1}^{n} |p_i - q_i|$$

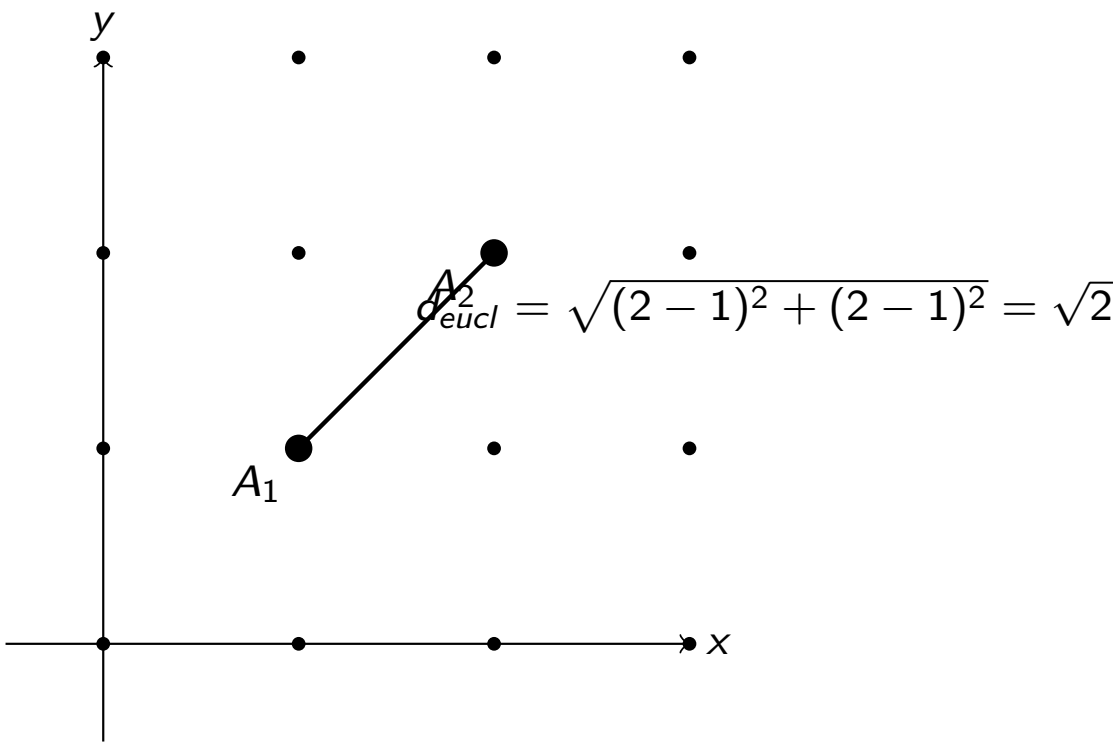
- **Distancia de Minkowski:**

$$d(p,q) = \left(\sum_{i=1}^{n} |p_i - q_i|^p \right)^{1/p}$$

- **Distancia Coseno:**

$$d(p,q) = 1 - \frac{p \cdot q}{\|p\|\|q\|}$$

Distancia Euclidiana

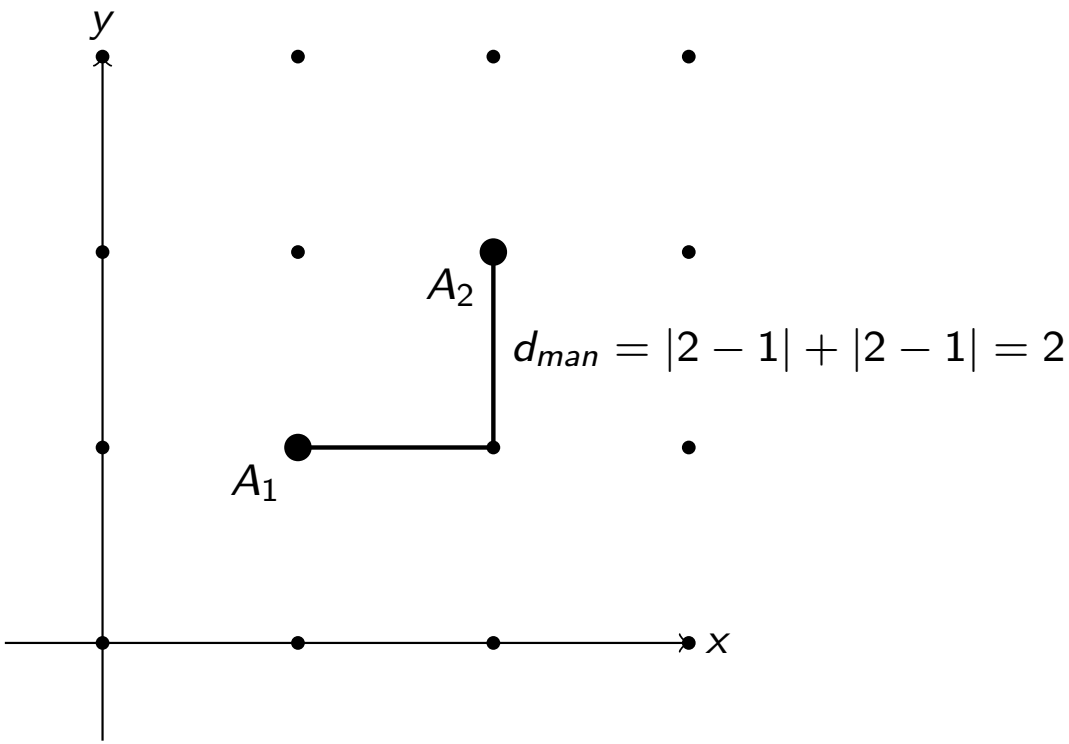


- La distancia euclidiana se calcula usando la fórmula:

$$d_{eucl} = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

- Es la distancia en línea recta entre dos puntos en el espacio.

Distancia de Manhattan

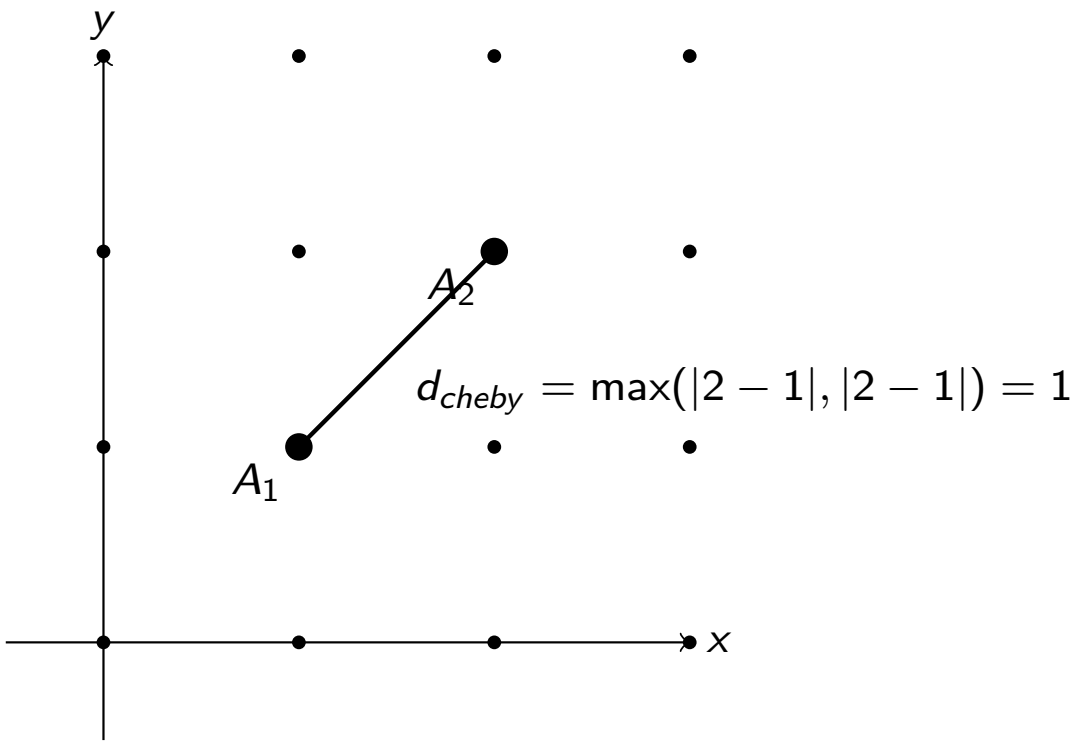


- La distancia de Manhattan se calcula usando la fórmula:

$$d_{man} = \sum_{i=1}^{n} |x_i - y_i|$$

- También conocida como distancia de ciudad o taxista, se basa en movimientos en línea recta a lo largo de ejes.

Distancia de Chebyshev

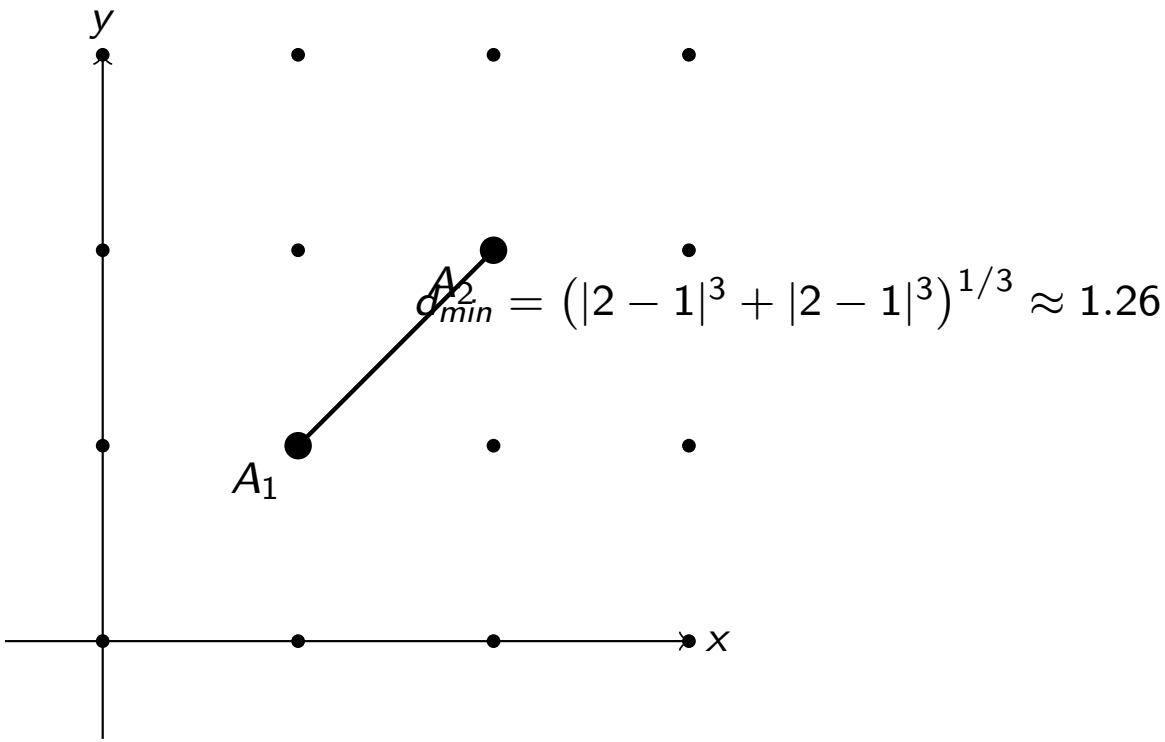


- La distancia de Chebyshev se calcula usando la fórmula:

$$d_{cheby} = \max_{i=1}^{n} |x_i - y_i|$$

- Mide la máxima distancia en cualquiera de las dimensiones entre los puntos.

Distancia de Minkowski

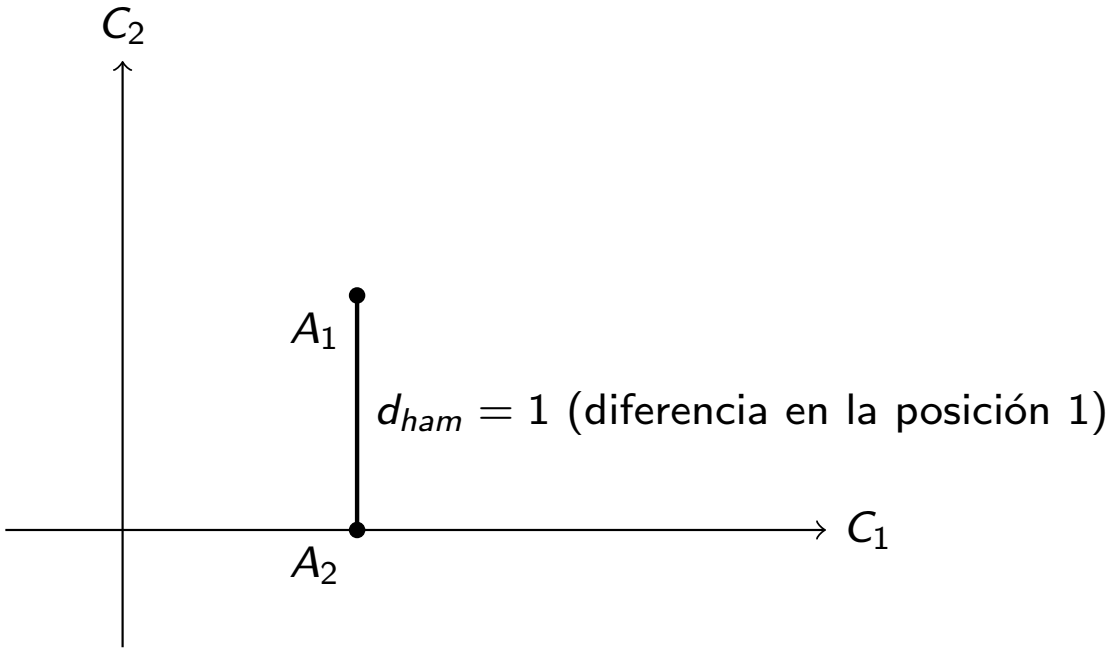


- La distancia de Minkowski se calcula como:

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

- Generaliza las distancias, dependiendo del valor de p.

Distancia de Hamming

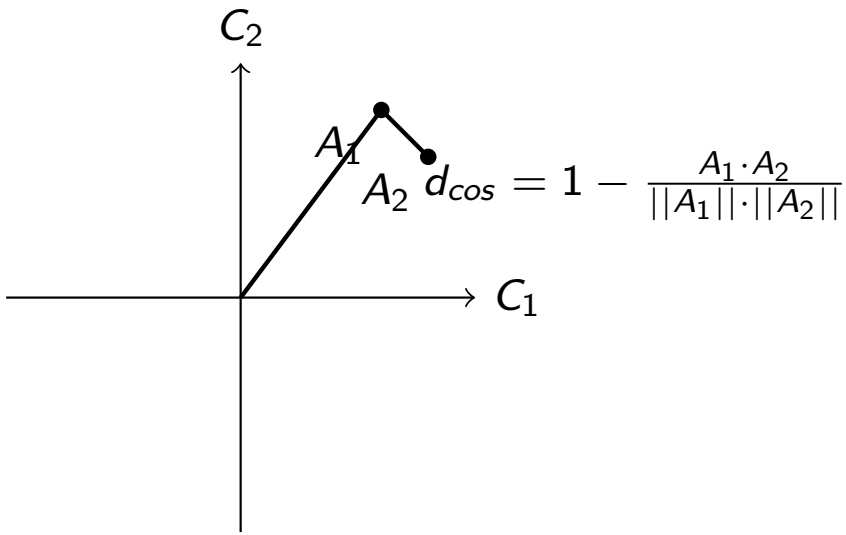


- La distancia de Hamming se calcula como:

$$d_{ham} = \frac{1}{n} \sum_{i=1}^{n} I(x_i \neq y_i)$$

- Se utiliza principalmente para variables categóricas o binarias y mide la cantidad de diferencias.

Distancia de Coseno



- La distancia de coseno se calcula como:

$$d_{cos} = 1 - \frac{A_1 \cdot A_2}{||A_1|| \cdot ||A_2||}$$

- Mide la similitud entre dos vectores en función del ángulo entre ellos.

Estandarización y Preprocesamiento de Datos

- kNN es sensible a la escala de los datos, ya que la mayoría de las métricas de distancia dependen de las magnitudes de los atributos.
- Si los atributos tienen diferentes escalas (por ejemplo, uno en metros y otro en milímetros), las distancias pueden estar sesgadas.

Estandarización de Datos

¿Es necesaria la estandarización?

- La estandarización o normalización de los datos es esencial cuando los atributos tienen diferentes escalas.
- Técnicas comunes:
  - **Estandarización:** Cada atributo se convierte en una variable con media 0 y desviación estándar 1:

$$z = \frac{x - \mu}{\sigma}$$

  - Normalización Min-Max: Escala los valores entre 0 y 1:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

Otras Técnicas de Preprocesamiento

- **Manejo de valores faltantes:**
  - Completar valores faltantes con la media, mediana o usando métodos más avanzados como kNN imputación.
- **Reducción de dimensionalidad:**
  - Métodos como PCA (Análisis de Componentes Principales) pueden reducir el número de dimensiones manteniendo la mayoría de la varianza, lo que mejora el rendimiento de kNN.
- **Selección de características:**
  - Eliminación de características irrelevantes o redundantes, utilizando técnicas como Selección basada en información, importancia de características o métodos tipo Forward, Backward y Stepwise.

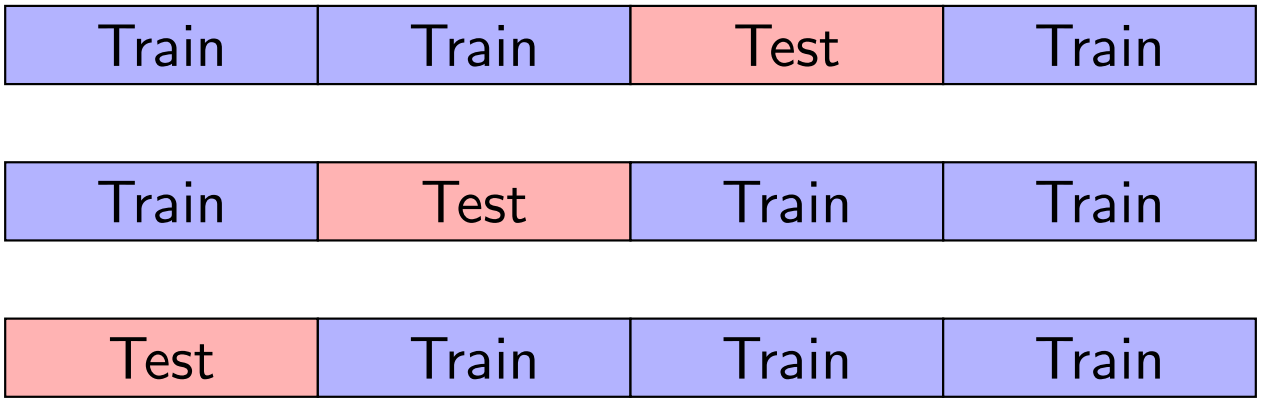
Elección del Parámetro k

- El valor de k es crítico para el rendimiento del algoritmo.
- Si k es demasiado pequeño, el modelo puede ser sensible al ruido (overfitting).
- Si k es demasiado grande, puede incluir puntos que no sean representativos (underfitting).
- **Método común:** Usar validación cruzada para encontrar el mejor valor de k.

¿Qué es la Validación Cruzada?

- La **validación cruzada** es una técnica de evaluación del rendimiento de un modelo, usada para evitar sobreajuste y subajuste.
- Consiste en dividir los datos en varios subconjuntos (o "folds") y evaluar el modelo en diferentes combinaciones de entrenamiento y prueba.
- El tipo más común es la **validación cruzada p-fold**:
  - El conjunto de datos se divide en k partes.
  - Se entrena el modelo en $k-1$ partes y se prueba en la parte restante.
  - Este proceso se repite k veces, variando el conjunto de prueba cada vez.
- Finalmente, se calcula el rendimiento promedio sobre todas las iteraciones.

Visualización de la Validación Cruzada k-fold



Repetir este proceso k veces

Aplicación de Validación Cruzada en kNN

- La validación cruzada es especialmente útil para seleccionar el valor óptimo del parámetro k en kNN.
- Proceso para encontrar el mejor k:
  1. Probar varios valores de k (por ejemplo, $k = 1, 3, 5, 7, 9$).
  2. Para cada valor de k, aplicar validación cruzada k-fold.
  3. Promediar el rendimiento de cada valor de k a lo largo de todos los folds.
  4. Seleccionar el k que maximice el rendimiento promedio.
- Esto evita problemas de sobreajuste o subajuste al encontrar un valor de k adecuado para el conjunto de datos.

Ventajas de la Validación Cruzada

- **Generalización robusta:** Reduce la variabilidad de las particiones aleatorias al evaluar el rendimiento en múltiples subconjuntos.
- **Mejora en la selección de parámetros:** Permite optimizar parámetros del modelo, como el valor de k en kNN, de manera sistemática.
- **Previene overfitting:** Asegura que el modelo no se ajuste excesivamente a un solo conjunto de datos, ya que se prueba en diferentes subconjuntos.
- **Aplicable a cualquier modelo:** La validación cruzada no es exclusiva de kNN; puede usarse para ajustar y evaluar cualquier tipo de modelo.

Selección de Prototipos

- Si se tienen excesivos datos, es importante reducir el tamaño del conjunto de entrenamiento para mejorar la eficiencia. A los individuos del conjunto de entrenamiento reducido se les denominaa **prototipos**.
- Técnicas como **Condensed Nearest Neighbor (CNN)** y **Edited Nearest Neighbor (ENN)** ayudan a seleccionar un subconjunto de los datos.
- **CNN:** Se enfoca en reducir el tamaño del conjunto de entrenamiento manteniendo la capacidad de clasificación.
- **ENN:** Elimina puntos del conjunto de entrenamiento que son clasificados incorrectamente por sus vecinos.

Selección de Prototipos

- El rendimiento de kNN depende del tamaño y la calidad del conjunto de entrenamiento.
- Seleccionar prototipos es una forma de reducir el tamaño del conjunto de datos sin sacrificar el rendimiento.
- Dos técnicas comunes para la selección de prototipos:
  - **CNN (Condensed Nearest Neighbor)**
  - **ENN (Edited Nearest Neighbor)**
- Objetivo: Eliminar ejemplos redundantes o ruidosos y mantener los ejemplos representativos.

Condensed Nearest Neighbor (CNN)

- Propuesta por Hart en 1968, la técnica **CNN** reduce el conjunto de entrenamiento seleccionando solo los ejemplos más representativos.
- Proceso:
  1. Inicia con un subconjunto pequeño, usualmente un ejemplo de cada clase.
  2. Se recorre el conjunto de entrenamiento y se agregan al subconjunto los ejemplos mal clasificados por kNN.
  3. Este proceso se repite hasta que no se agregan nuevos ejemplos.
- **Objetivo**: Mantener un subconjunto reducido que aún permita clasificar correctamente los ejemplos del conjunto original.

Edited Nearest Neighbor (ENN)

- Propuesta por Wilson en 1972, la técnica **ENN** se enfoca en mejorar la calidad del conjunto de entrenamiento eliminando ejemplos ruidosos.
- Proceso:
  1. Se aplica kNN (usualmente con $k = 3$) a cada ejemplo del conjunto de entrenamiento.
  2. Si el ejemplo está mal clasificado por sus k vecinos, se elimina del conjunto de datos.
- **Objetivo**: Eliminar ejemplos que están mal clasificados por sus vecinos cercanos, generalmente debido a ruido o errores de etiquetado.

Comparación: CNN vs ENN

- **Objetivo principal**:
  - **CNN**: Reducir el tamaño del conjunto de datos manteniendo los ejemplos más representativos.
  - **ENN**: Mejorar la calidad eliminando ejemplos ruidosos o mal clasificados.
- **CNN**:
  - Menor conjunto de datos, pero con riesgo de incluir ejemplos ruidosos.
  - Útil para reducir el costo computacional en grandes conjuntos de datos.
- **ENN**:
  - Conjunto más limpio y de mejor calidad.
  - Puede eliminar demasiados puntos, especialmente en zonas de frontera entre clases.
- **Combinación**:
  - En la práctica, a veces se combinan ambas técnicas (CNN + ENN) para obtener un conjunto de datos reducido y limpio.

Ventajas y Desventajas

Ventajas

- Algoritmo sencillo y fácil de implementar.
- No requiere entrenamiento explícito.
- Funciona bien en problemas con límites de decisión no lineales.

Desventajas

- Alto costo computacional en tiempo de predicción.
- Sensible a la escala de los datos.
- La elección de k y la métrica de distancia pueden ser difíciles.

Aplicaciones de kNN

- **Reconocimiento de patrones:** Reconocimiento de escritura, clasificación de imágenes.
- **Sistemas de recomendación:** Basado en la similitud de usuarios o productos.
- **Detección de anomalías:** Detección de fraudes, intrusiones o errores.

kNN como Método de Regresión

Uso de kNN para Regresión:

- kNN no solo se usa para clasificación, sino también para predecir variables continuas.
- En lugar de asignar la clase más frecuente entre los vecinos, el valor de la variable continua se predice como la **media** o **mediana** de los valores de los k vecinos más cercanos.

kNN como Método de Regresión

Algoritmo:

- Para un nuevo punto de datos:
  - Se calculan las distancias a todos los puntos del conjunto de entrenamiento.
  - Se seleccionan los **k** vecinos más cercanos.
  - La predicción se obtiene calculando la **media** (o a veces la mediana) de las etiquetas continuas de estos vecinos.
  - Para medir la calidad de la predicción sigue siendo una buena medida el R^2.

kNN como Método de Regresión

Ventajas de kNN para Regresión:

- Modelo no paramétrico, sin suposiciones sobre la forma de la función.
- Fácil de implementar y entender.

Desventajas:

- Sensible a la escala de los datos y a la elección de k.
- Requiere un número significativo de datos para obtener buenas predicciones.


UNIVERSITAS
Miguel Hernández