UNIVERSIDAD SAN SEBASTIAN
VOCACIÓN POR LA EXCELENCIA

Implementación del algoritmo *K-Nearest Neighbor* (KNN)

Inteligencia Artificial y Ciencia de Datos - DRBC 0008

Margot Inés Paco Chipana

Santiago de Chile, 1 de noviembre de 2025

1. Introducción

El algoritmo *k-Nearest Neighbors* (KNN) es un método de clasificación supervisado basado en instancias que asigna una clase a un nuevo ejemplo según la mayoría de sus k vecinos más cercanos. Su simplicidad, efectividad y bajo costo de entrenamiento lo convierten en una herramienta fundamental para el aprendizaje supervisado.

En esta práctica se implementó KNN desde cero en Python, sin librerías de aprendizaje automático, aplicándolo al conjunto de datos *Wine*. Se analizaron tres métricas de distancia (Euclidiana, Manhattan y Coseno), distintos valores de k, y dos esquemas de validación: validación cruzada 5-fold y *Leave-One-Out* (LOOCV). Finalmente, se comparó la implementación manual con la versión del clasificador de `scikit-learn` y con el algoritmo KStar disponible en Weka.

2. Metodología

- **Implementación manual:** se programó el algoritmo KNN calculando manualmente las distancias, el voto mayoritario y la resolución de empates.
- **Normalización:** se aplicó la transformación min–max a cada atributo:

$$a_i' = \frac{a_i - \text{mín}(a_i)}{\text{máx}(a_i) - \text{mín}(a_i)}$$

  dentro de cada pliegue de validación para evitar fuga de información.
- **Métricas evaluadas:**
  1. Distancia Euclidiana
  2. Distancia Manhattan
  3. Distancia del Coseno

- **Selección del parámetro k:** se probaron valores impares entre 1 y 17 para evitar empates.
- **Validación:** se utilizó validación cruzada estratificada 5-fold y validación LOOCV.
- **Comparación:** se contrastaron los resultados con el modelo `KNeighborsClassifier` de `scikit-learn` y el clasificador `KStar` de Weka (medida de entropía).

3. Resultados

Tabla 1: Resultados de validación cruzada y LOOCV (implementación manual y sklearn)

Distancia	k óptimo	Accuracy (CV)	Accuracy (LOOCV)	Accuracy sklearn (CV)
Euclidiana	15	0.9776	0.9775	0.9775
Manhattan	17	0.9778	0.9775	0.9719
Coseno	3	0.9722	0.9663	0.9551

Las tres métricas muestran una precisión superior al 95 %. Las distancias Euclidiana y Manhattan presentan el mejor rendimiento (≈97.7 %), mientras que la métrica del Coseno desciende ligeramente (≈95.5–97.2 %). Esto indica que las clases del dataset *Wine* se separan principalmente por magnitud, no por orientación angular. .

3.1. Matrices de confusión (sklearn, acumuladas 5-fold)

Euclidiana (k=15):

$$\begin{bmatrix} 58 & 1 & 0 \\ 2 & 68 & 1 \\ 0 & 0 & 48 \end{bmatrix}$$

Manhattan (k=17):

$$\begin{bmatrix} 59 & 0 & 0 \\ 3 & 66 & 2 \\ 0 & 0 & 48 \end{bmatrix}$$

Coseno (k=3):

$$\begin{bmatrix} 59 & 0 & 0 \\ 6 & 63 & 2 \\ 0 & 0 & 48 \end{bmatrix}$$

En todas las matrices se observa que las clases 1 y 3 son clasificadas correctamente en la mayoría de los casos, mientras que la clase 2 concentra la mayoría de los errores. En las distancias Euclidiana y Manhattan los errores son mínimos (4–5 instancias mal clasificadas), lo que explica el alto desempeño del modelo. Sin embargo, al emplear la distancia del Coseno, el número de errores en la clase 2 aumenta, reduciendo la precisión general a aproximadamente 95 %.

4. Evaluación del desempeño

El desempeño se evaluó mediante la exactitud promedio (accuracy), las matrices de confusión y el análisis de estabilidad entre validaciones. Los valores de *accuracy* y *F1-score* promedio oscilaron entre 0.95 y 0.98, indicando un modelo estable y con baja varianza. La clase 2 fue la más difícil de distinguir, generando la mayoría de los errores de clasificación.

En comparación con la versión de `scikit-learn`, los resultados son prácticamente idénticos (diferencias menores al 1 %), lo que valida la correcta implementación del algoritmo desde cero. Asimismo, los resultados de LOOCV confirman que el modelo generaliza correctamente y no presenta sobreajuste.

5. Comparación con KStar (Weka)

El clasificador **KStar** de Weka fue evaluado con parámetro `B=20` y validación cruzada 5-fold. El resultado fue una exactitud de **73.0 %** y una matriz de confusión con predominio de errores en la clase 2:

$$\begin{bmatrix} 43 & 14 & 2 \\ 9 & 54 & 8 \\ 3 & 12 & 33 \end{bmatrix}$$

Este rendimiento, significativamente menor al obtenido con KNN, se explica por el tipo de métrica utilizada: KStar mide la similitud mediante una *función de entropía de transformación* en lugar de una distancia geométrica. Dicha medida es más robusta en datasets ruidosos o mixtos, pero menos eficiente en conjuntos puramente numéricos y bien separados como el *Wine*. Por ello, el KNN tradicional supera a KStar en este caso.

6. Conclusiones

- La implementación manual del algoritmo KNN replicó con precisión el comportamiento del modelo de `scikit-learn`, confirmando su validez.
- Las distancias Euclidiana y Manhattan mostraron los mejores resultados (≈97.7 %), mientras que la métrica del Coseno presentó una ligera caída.
- El modelo demostró estabilidad entre los esquemas de validación cruzada y LOOCV, sin evidencias de sobreajuste.
- El clasificador KStar de Weka obtuvo un desempeño inferior (≈73 %), debido a su medida basada en entropía, menos adecuada para datos numéricos limpios.

Enlace al repositorio:
https://github.com/MargotPC/KNN-from-Scratch-wine-