Ingeniería en Inteligencia Artificial,
Machine Learning
Sem: 2025-1, 5BM1, Práctica 8, Fecha de entrega: 11 de noviembre de 2024

Laboratorio 8: Clasificador Naive Bayes

Machine Learning

Grupo: 5BM1

Profesor: Andrés Floriano García

Integrantes:
Juan Manuel Alvarado Sandoval
Alexander Iain Crombie Esquinca
Herrera Saavedra Jorge Luis
Quiñones Mayorga Rodrigo

Contents

1 Introducción

En esta práctica, implementaremos y validaremos el Clasificador Naive Bayes, un modelo probabilístico utilizado ampliamente para tareas de clasificación. Su base se encuentra en el Teorema de Bayes y en la asunción de independencia condicional entre características. Este clasificador es efectivo y eficiente, especialmente cuando las características son condicionalmente independientes, lo que lo convierte en una opción adecuada para conjuntos de datos de alta dimensión.

Para esta práctica, utilizaremos tres conjuntos de datos estándar: *Banking*, *Titanic* y *Iris*. Los clasificadores serán validados mediante tres métodos de validación cruzada: Hold-Out (70/30 estratificado), 10-Fold Cross-Validation estratificado y Leave-One-Out. También se evaluará el rendimiento de los clasificadores usando métricas como *Accuracy* y la matriz de confusión.

2 Clasificador Naive Bayes

El clasificador Naive Bayes es un algoritmo de clasificación basado en el Teorema de Bayes, que describe la probabilidad posterior de una clase C dada un conjunto de características X. La fórmula general del Teorema de Bayes es:

$$P(C|X) = \frac{P(X|C) \cdot P(C)}{P(X)}$$

Donde:

- $P(C|X)$ es la probabilidad posterior de la clase C dada las características X.
- $P(X|C)$ es la probabilidad de observar X dado que la clase es C.
- $P(C)$ es la probabilidad a priori de la clase C.
- $P(X)$ es la probabilidad de las características X en general.

La suposición "naive" o ingenua se refiere a la simplificación de que todas las características son independientes entre sí, es decir, $P(X|C)$ puede descomponerse en el producto de las probabilidades de cada característica individual dada la clase:

$$P(X|C) = P(x_1|C) \cdot P(x_2|C) \cdot \cdots \cdot P(x_n|C)$$

Aunque esta suposición de independencia rara vez se cumple en la práctica, en muchos casos el algoritmo Naive Bayes ofrece un rendimiento sorprendentemente bueno, incluso cuando las características no son realmente independientes.

2.1 Laplace Smoothing

Uno de los desafíos de Naive Bayes, especialmente cuando se trabaja con datos de texto o categorías con frecuencia baja, es el problema de las probabilidades cero. Este problema ocurre cuando alguna de las características no aparece en el conjunto de entrenamiento para una clase específica. En este caso, $P(x_i|C)$ para una característica x_i sería igual a cero, lo que lleva a que toda la probabilidad $P(C|X)$ sea cero, independientemente de los demás factores.

Para resolver este problema, se emplea el *Laplace Smoothing* (también conocido como suavizado aditivo).

$$P(x_i|C) = \frac{\text{count}(x_i, C) + \alpha}{\text{count}(C) + \alpha \cdot |V|}$$

Donde:

- $\text{count}(x_i, C)$ es la cantidad de veces que la característica x_i aparece en el conjunto de datos para la clase C.
- $\text{count}(C)$ es la cantidad total de ocurrencias de la clase C.
- α es el parámetro de suavizado, comúnmente $\alpha = 1$ (suavizado de Laplace).
- $|V|$ es el número de características diferentes en el conjunto de datos (el tamaño del vocabulario).

El suavizado de Laplace asegura que todas las probabilidades sean mayores que cero.

3 Metodología

3.1 Conjuntos de Datos Utilizados

Para validar el clasificador Naive Bayes, se utilizaron tres datasets:

- **Banking**: Conjunto de datos que contiene información sobre campañas de telemarketing realizadas por una entidad bancaria. **banking**.
- **Titanic**: Conjunto de datos que incluye información sobre los pasajeros del Titanic, usado comúnmente para predicción de supervivencia. **titanic**.
- **Iris**: Conjunto de datos clásico para la clasificación de especies de flores según sus características físicas.

3.2 Métodos de Validación

Se aplicaron los siguientes métodos de validación:

- **Hold-Out 70/30 estratificado**: Se divide el conjunto de datos en 70% para entrenamiento y 30% para prueba, manteniendo la proporción de clases.
- **10-Fold Cross-Validation estratificado**: Se divide el conjunto de datos en 10 partes, utilizando 9 partes para entrenamiento y una para prueba, repitiendo el proceso 10 veces.
- **Leave-One-Out**: Cada instancia se usa una vez como conjunto de prueba mientras el resto se usa para entrenar el modelo.

3.3 Medidas de Desempeño

Las medidas de desempeño empleadas fueron:

- **Accuracy**: Proporción de instancias correctamente clasificadas.
- **Matriz de Confusión**: Matriz que muestra el número de clasificaciones correctas e incorrectas para cada clase.

4 Resultados

4.1 Banking Dataset

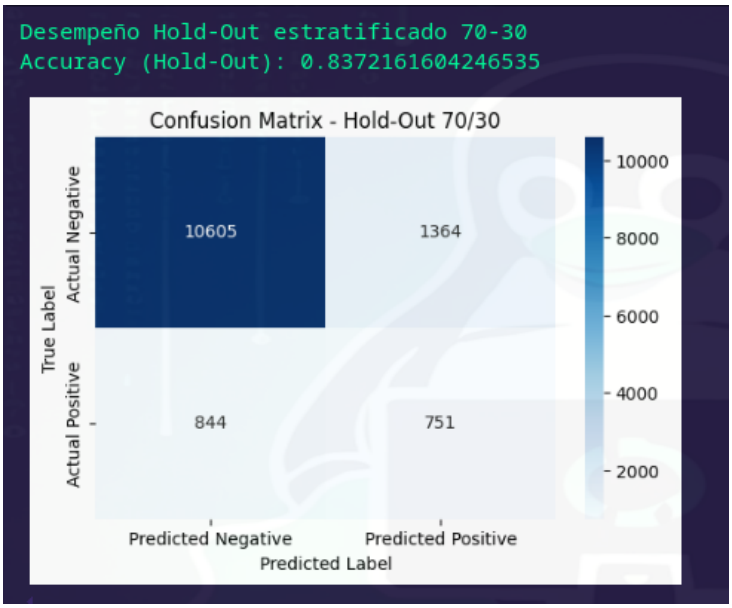

Figure 1: Desempeño de Clasificador Naive Bayes con Hold Out para Banking Dataset.

Figure 2: Desempeño de Clasificador Naive Bayes con K-Folds (10) para Banking Dataset.

Leave One Out no pudo ser aplicado en este dataset por cuestiones de tiempo, ya que este contenía más de 45,000 registros.

4.2 Titanic Dataset

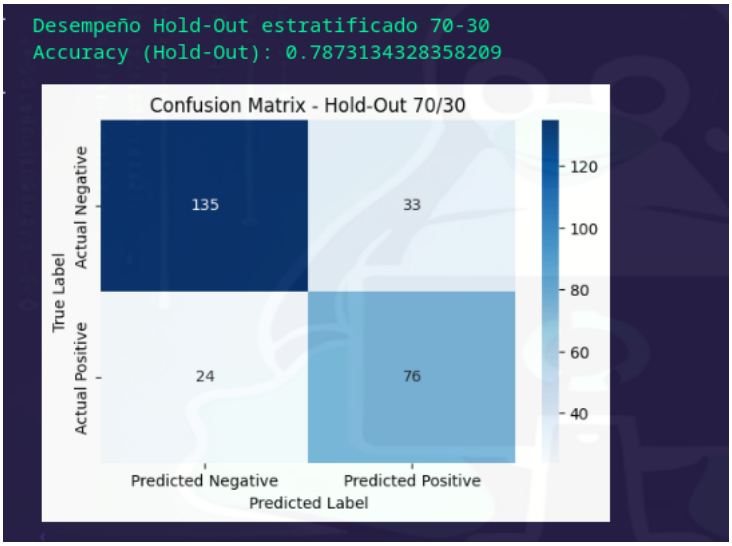

Figure 3: Desempeño de Clasificador Naive Bayes con Hold Out para Titanic Dataset.

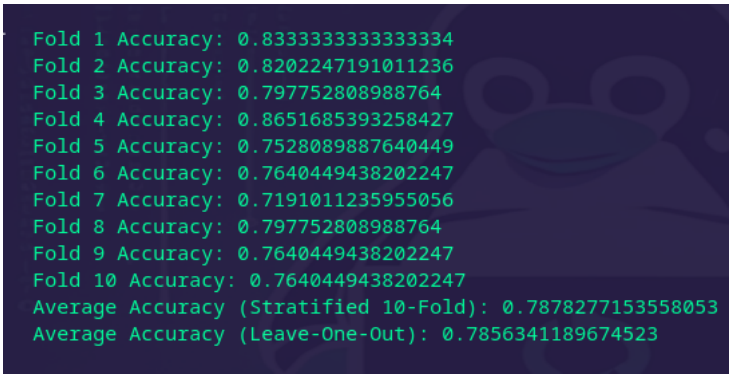

Figure 4: Desempeño de Clasificador Naive Bayes con K-Folds (10) y Leave One Out
Titanic Dataset.

4.3 Iris Dataset

Figure 5: Desempeño de Clasificador Naive Bayes con Hold Out para Iris Dataset.

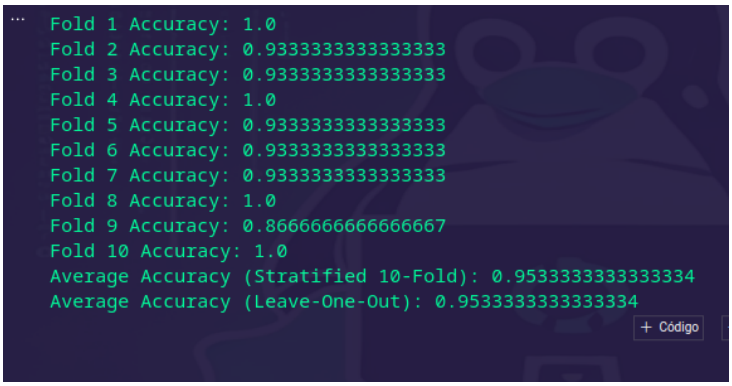

Figure 6: Desempeño de Clasificador Naive Bayes con K-Folds (10) y Leave One Out
Iris Dataset.

4.4 Justificación para el valor de K

El valor de K en el método de validación cruzada K-Fold se justifica con base en los resultados de desempeño obtenidos en los tres conjuntos de datos. Se ha seleccionado $K = 10$ debido a que proporciona un equilibrio adecuado entre sesgo y varianza, mostrando una consistencia razonable en las tasas de *accuracy*. Para el dataset Banking, la precisión promedio fue de 0.836, mientras que para el Titanic fue de 0.788 y para el Iris alcanzó 0.836.

Estos resultados indican que el clasificador Naïve Bayes mantiene un rendimiento estable a lo largo de los diferentes subconjuntos del conjunto de datos, con un promedio de *accuracy* que varía entre 0.787 y 0.848. Utilizar un valor de $K = 10$ permite que los subconjuntos de entrenamiento sean lo suficientemente grandes como para representar adecuadamente la distribución de los datos, mientras que el subconjunto de prueba sigue siendo útil para evaluar el rendimiento del modelo sin sobreajuste. Así, $K = 10$ resulta ser una opción adecuada para todos los datasets evaluados.

5 Conclusión

El clasificador Naive Bayes es una técnica eficaz y sencilla para la clasificación de datos. A pesar de su suposición de independencia entre características, puede ofrecer buenos resultados, como se observó en los conjuntos de datos utilizados.

Los resultados mostraron que la precisión del clasificador varía según el conjunto de datos y el método de validación utilizado. El método de validación 10-Fold Cross-Validation demostró ser un buen balance entre eficiencia y precisión en comparación con el Hold-Out y Leave-One-Out, que pueden tener limitaciones en términos de tiempo de procesamiento o variabilidad de los resultados.

Este laboratorio permitió observar la efectividad de Naive Bayes en diferentes contextos, destacando la importancia de una correcta selección de métodos de validación y el análisis detallado de las características del dataset para optimizar el rendimiento del clasificador.

6 Enlace al Repositorio

Haz click para seguir el enlace

7 Referencias

- Moro, S., Cortez, P., & Rita, P. (2014). A Data-Driven Approach to Predict the Success of Bank Telemarketing. *Decision Support Systems*, 62, 22-31. Recuperado de: https://archive.ics.uci.edu/ml/datasets/bank+marketing
- Will Cukierski. (2012). Titanic - Machine Learning from Disaster. Kaggle. Recuperado de: https://kaggle.com/competitions/titanic
- Iris dataset. Recuperado de: https://archive.ics.uci.edu/ml/datasets/iris
- Scikit-learn: Naive Bayes classification. Recuperado de: https://scikit-learn.org/stable/modules/naive_bayes.html