Instituto Tecnológico de estudios superiores de Monterrey

Campus Estado de México

Departamento de Ingeniería

TC3006C

Inteligencia Artificial Avanzada

Grupo: 101

Profesor: Jorge Adolfo Ramírez Uresti

Momento de Retroalimentación

"*Módulo 2 Uso de framework o biblioteca de aprendizaje máquina para la implementación de una solución.*"

Fecha: 10/09/2025

Alumno:
Santiago Villazón Ponce de León A01746396

Introducción

En el presente trabajo se implementó un algoritmo de clasificación utilizando un framework de aprendizaje automático, en este caso **scikit-learn**, para resolver un problema de clasificación con datos categóricos. A diferencia de la entrega anterior, en la cual se programó manualmente un algoritmo de árboles de decisión y ensamble, en esta ocasión se emplearon directamente las herramientas provistas por la biblioteca, con el objetivo de demostrar su correcto uso y la configuración de los modelos.

El dataset seleccionado fue nuevamente **Mushrooms**, el cual contiene observaciones de distintos hongos descritos por atributos categóricos como el tipo de sombrero, color, olor, forma de la raíz y superficie del tallo. La variable objetivo, ubicada en la última columna y denominada *class*, distingue si un hongo es comestible (*edible*, e) o venenoso (*poisonous*, p). La clasificación de hongos resulta particularmente relevante, ya que un error al identificar un hongo venenoso podría tener consecuencias graves para la salud. (De igual manera se pueden probar los distintos datasets que están en el repositorio.)

El dataset original se dividió en dos subconjuntos estratificados: **80% para entrenamiento** y **20% para prueba**, lo que permitió asegurar que la distribución de clases se mantuviera en ambos conjuntos. A partir de este preprocesamiento, se guardaron los archivos ***train.csv*** y ***test.csv***, garantizando reproducibilidad y transparencia en la separación de datos.

Configuración del modelo

Para esta implementación se eligió un **Random Forest Classifier** de scikit-learn, configurado de la siguiente manera:

- **Algoritmo:** RandomForestClassifier (scikit-learn)
- **Número de árboles (n_estimators):** 200
- **Semilla de aleatoriedad (random_state):** 42
- **Criterio de división:** Gini (por defecto en scikit-learn)
- **Muestreo estratificado:** Sí (mantener balance de clases en train/test)

El preprocesamiento de variables categóricas se realizó utilizando **OneHotEncoder**, que transforma los valores categóricos en variables binarias, manejando automáticamente categorías no vistas en el conjunto de prueba. Este paso es crucial, ya que todos los atributos del dataset son categóricos.

Resultados

Tras ejecutar el modelo con el conjunto de datos *Mushrooms*, se obtuvieron los siguientes resultados en el conjunto de prueba ***(test.csv***):

Matriz de confusión (archivo ***matriz_confusion.csv***):

- Verdaderos Positivos (TP): 783
- Verdaderos Negativos (TN): 842
- Falsos Positivos (FP): 0
- Falsos Negativos (FN): 0

Métricas globales (archivo *metricas.csv*):

- Exactitud (Accuracy): **1.0000**
- Precisión macro (Precision_macro): **1.0000**
- Sensibilidad/Recall macro (Recall_macro): **1.0000**
- F1 macro: **1.0000**

Reporte de clasificación por clase:

- Para la clase *edible*: precisión 1.0, recall 1.0, f1-score 1.0
- Para la clase *poisonous*: precisión 1.0, recall 1.0, f1-score 1.0

El archivo ***predicciones.csv*** confirma que todas las instancias del conjunto de prueba fueron clasificadas correctamente por el modelo, sin errores en las etiquetas.

Análisis de los resultados

El modelo alcanzó un desempeño **perfecto** (100% en todas las métricas) al clasificar hongos como comestibles o venenosos. Estos resultados concuerdan con lo observado en la entrega anterior con la implementación manual: el dataset Mushrooms es altamente separable debido a la naturaleza de sus atributos categóricos. Variables como el olor o la forma del tallo contienen patrones tan distintivos que permiten separar las clases de manera determinista.

En consecuencia, tanto los métodos implementados manualmente como los modelos de scikit-learn alcanzan resultados idénticos, ya que las reglas necesarias para separar las clases son claras y consistentes.

Es importante señalar que este rendimiento perfecto **no implica necesariamente sobreajuste (overfitting)**, ya que el desempeño en entrenamiento y prueba es igualmente alto. Más bien, refleja que el dataset está diseñado con atributos muy discriminativos y sin ruido significativo.

No obstante, este tipo de resultados rara vez se replica en datasets reales más complejos, donde siempre existe cierto nivel de ambigüedad, ruido y solapamiento entre clases. Por ello, aunque este experimento demuestra el correcto funcionamiento de scikit-learn y del pipeline implementado, no debe generalizarse como un resultado típico en machine learning.

Conclusiones

La implementación de un **Random Forest** utilizando scikit-learn sobre el dataset Mushrooms logró un desempeño perfecto, confirmando la capacidad de este framework para manejar datos categóricos mediante preprocesamiento con OneHotEncoder y algoritmos robustos de clasificación.

La matriz de confusión evidenció que no hubo errores de predicción, y las métricas globales y por clase alcanzaron valores máximos. Estos resultados corroboran que el dataset Mushrooms es un caso didáctico, ideal para validar implementaciones y comprender el flujo de trabajo en machine learning supervisado.

En conclusión, este ejercicio permitió:

1. Reproducir el flujo completo de preprocesamiento, división de datos, entrenamiento y evaluación utilizando un framework.
2. Generar evidencias reproducibles (***train.csv***, ***test.csv***, ***predicciones.csv***, ***matriz_confusion.csv***, ***metricas.csv***).
3. Reflexionar sobre las limitaciones de los resultados perfectos y la importancia de evaluar los modelos en contextos más realistas.

El proyecto cumple así con los objetivos del módulo: demostrar el uso de un framework de aprendizaje automático, la correcta configuración de un algoritmo de clasificación y la documentación de sus resultados a través de métricas y análisis.