CC3045 – Inteligencia Aritificial

Laboratorio 1

Instrucciones

- Esta es una actividad en grupos de no más de 3 integrantes.
    - Recuerden **unirse al grupo de canvas**
- No se permitirá ni se aceptará cualquier indicio de copia. De presentarse, se procederá según el reglamento correspondiente.
- Tendrán hasta el día indicado en Canvas.
    - No se confíen, aprovechen el tiempo en clase para entender todos los ejercicios y avanzar lo más posible.
- **NOTA**: Limiten el uso de IA generativa. Intenten primero buscar en fuentes de internet y si en verdad necesitan usarla, asegúrense de colcar el prompt que utilizar para cada task donde corresponda, así como una explicación de por qué ese prompt funcionó.

Task 1 – El Dilema del Negocio

Imagine que usted ha sido contratado como Lead Data Scientist para tres startups diferentes. Para cada caso, responda las preguntas justificando su respuesta. No quiero definiciones de libro, quiero análisis de caso.

Caso A: "MediScan AI" (Diagnóstico Médico)

Están desarrollando un modelo para detectar un tipo de cáncer raro en etapas tempranas a partir de radiografías.

1. En este contexto, ¿qué es peor: un **Falso Positivo** (decirle a alguien sano que podría tener cáncer) o un **Falso Negativo** (decirle a alguien enfermo que está sano)?
2. Basado en lo anterior, si tuviera que optimizar el modelo priorizando una sola métrica entre **Precisión (Precision)** y **Sensibilidad (Recall/Sensitivity)**, ¿cuál escogería y por qué?
3. ¿Por qué el *Accuracy* sería una métrica peligrosa para presentar a los inversionistas en este caso específico?

Caso B: "SpamGuard" (Filtro de Correos)

Están creando un filtro de spam para una corporación grande.

1. ¿Qué error causaría más molestia y pérdida de productividad a los empleados: que un correo de spam llegue al Inbox (FP o FN dependiendo de su definición) o que un correo importante de un cliente se vaya a la carpeta de Spam?
2. ¿Qué métrica priorizaría aquí: **Precisión** o **Recall**? (Defina cuál es su clase positiva).

Caso C: "Zillow 2.0" (Predicción de Precios de Casas)

Están prediciendo el valor de mercado de propiedades. Tienen un modelo con las variables: *Metros Cuadrados, Ubicación, Número de Cuartos*. Ahora, un junior engineer sugiere agregar 50 variables nuevas (como "color de la puerta", "nombre del dueño anterior", etc.) y nota que el **R^2 (R-cuadrado)** subió ligeramente.

1. ¿Deberíamos confiar en ese aumento del R^2 para decir que el modelo es mejor?

2. ¿Qué métrica debería observar para saber si esas nuevas variables realmente aportan valor o son solo ruido? Justifique basándose en la diapositiva de "R^2 Ajustado".

Task 2 – Ingeniería de Datos

Utilizando Python (Pandas/NumPy), simule y procese un dataset. **No se permite usar funciones mágicas de limpieza automática** (como SimpleImputer de sklearn), deben hacerlo con lógica de programación para demostrar que entienden el proceso.

1. **Generación de Dataset Sucio**:
    a. Cree un DataFrame de 100 filas y 3 columnas: Edad, Salario y Compró_Producto (0 o 1).
    b. Introduzca intencionalmente valores NaN (nulos) en el 10% de la columna Edad.
    c. Genere un desbalance de clases en Compró_Producto: 90 filas deben ser '0' (No compró) y 10 filas '1' (Compró).
2. **Manejo de Datos Faltantes (Imputación):**
    a. Escriba un algoritmo que recorra la columna Edad.
    b. Si encuentra un valor faltante, rellénelo con el promedio de las edades existentes.
    c. Pregunta extra en código (comentario): ¿En qué situación usar el promedio sería una mala idea y sería mejor usar la mediana?
3. **Manejo de Datos Desbalanceados (Undersampling Manual)**:
    a. Dado que la clase '0' es mayoritaria, implemente una función que realice Undersampling:
        i. Debe mantener todas las filas de la clase minoritaria ('1').
        ii. Debe seleccionar aleatoriamente un número de filas de la clase mayoritaria ('0') igual al número de filas de la clase minoritaria.
        iii. El resultado debe ser un nuevo DataFrame balanceado (aprox. 20 filas en total).

Task 3 – Métricas de Desempeño

Escriba dos funciones en Python desde cero (usando math o numpy, pero sin usar sklearn.metrics) para calcular el error de dos listas de valores:

- y_real = [100, 150, 200, 250, 300] (Valores reales)
- y_pred = [110, 140, 210, 240, 500] (Predicciones - note el error masivo en el último dato).

En base a esto, realice:

1. Implemente la fórmula de RMSE vista en clase, como una función
2. Implemente la fórmula de MAE vista en clase, como una función
3. **Comparación**: Ejecute ambas funciones con los vectores dados.
    a. Imprima ambos resultados.
    b. Escriba un print final explicando: ¿Cuál de las dos métricas penalizó más el error del último dato (el 500)? ¿Por qué esto es importante si estamos prediciendo, por ejemplo, dosis de medicamentos?

Entregas en Canvas

1. Documento PDF con las respuestas a cada task
2. Archivo .py, o link a repositorio de GitHub (No se acepta entregas en otros medios)

Evaluación

1. [1.5 pt] Task 1
2. [1.5 pt] Task 2
3. [1.0 pt] Task 3

Total 4 pts