Universidad Nacional de Ingeniería
Facultad de Ciencias
Escuela Profesional de Ciencia de la Computación

Práctica Calificada I

Minería de Datos

CC442 **16/01/2026** **Ciclo: 2025-III**

Puntaje: 20 puntos. **Duración**. 120 minutos

PARTE I (9 puntos)

Completar con código en Python, el notebook ***Parte-I_PC1_CC442.ipynb***

PARTE II (11 puntos)

El Dilema de la Simplicidad en el Mercado Inmobiliario

Contexto

Has sido contratado por la startup ***UNIPropTech*** que busca predecir el valor de las viviendas en California. El CEO de la empresa cree que cuanta más información se recolecte de los sensores y censos (aunque sea irrelevante), mejor será su predicción.

Sin embargo, sabes que "lanzar todas las variables al modelo" es un error costoso. Tu misión hoy es demostrarle al equipo técnico, con datos en mano, que **menos es más**.

Su desafío

Fase 1: La Preparación del Terreno

Antes de empezar, debemos asegurarnos de que nuestras variables "hablen el mismo idioma".

- Carga los datos de viviendas que te hemos proporcionado. (**en la Nota**)
- Aplica una técnica de **Estandarización**.

Pregunta: ¿Por qué no podemos comparar una variable como "Número de habitaciones" con "Ingreso promedio" sin escalarlas primero?

Fase 2: El Torneo de Modelos - Búsqueda Exhaustiva

No vamos a adivinar. Vamos a usar la técnica de **Fuerza Bruta (Exhaustive Search)** utilizando la librería ***mlxtend***.

- Tu tarea es ***programar un algoritmo en Python***, que evalúe **todas las combinaciones posibles** de predictores (desde 1 hasta 5 variables).
- Queremos encontrar el subconjunto "dorado".

Fase 3: El Juicio de los Tres Jueces (15 min)

Para convencer al CEO, no basta con mostrar el R^2 (sabemos que ese número miente cuando añadimos variables basura). Debes presentar un informe comparativo usando los tres jueces que aprendimos en la **Sección 6.4** del libro del curso:

1. **El R-Cuadrado Ajustado:** ¿Realmente mejora el modelo al añadir esa variable extra?
2. **El AIC (Akaike):** El juez que busca el equilibrio.
3. **El BIC (Bayesiano):** El juez más estricto y tacaño con las variables.

Fase 4: La Reunión de Directorio

Genera una visualización (o una tabla detallada si el sistema gráfico falla) que muestre la evolución de estos tres jueces. Al final, debes redactar una conclusión breve:

- ¿Cuál es el número óptimo de variables para nuestro producto?
- Si el BIC nos dice que usemos 3 variables y el AIC nos sugiere 5, ¿qué decisión tomarías tú basándote en el concepto de **Parsimonia**?

Nota:

```
# --- IMPORTS OBLIGATORIOS ---
from sklearn.datasets import fetch_california_housing
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import StandardScaler
from mlxtend.feature_selection import ExhaustiveFeatureSelector as EFS
```

Cuidado con el Overfitting: Si tu modelo es demasiado complejo, fallará cuando intentemos predecir casas nuevas fuera de nuestro dataset.