AAI.EX.20230207

Ejercicios elaborados con fines educativos, inspirados en los contenidos evaluados en el exámen del 07/02/2023 (convocatoria Feb-2023) de Aprendizaje Automático 1 del MUICD de la UNED.

Este documento no es una copia ni una transcripción del examen oficial, sino una redacción propia de ejercicios conceptualmente equivalentes.

Test

Pregunta correcta: +0.5 puntos
Pregunta incorrecta: -0.1 puntos

AAI.EX.20230207.T.1

Enunciado AAI.EX.20230207.T.1
La técnica conocida como LLE (Local Linear Embedding) se caracteriza por ser:

a) Un método lineal de reducción de dimensionalidad basado en proyecciones globales.

b) Un método lineal que preserva relaciones lineales globales entre instancias.

c) Un método no lineal de reducción de dimensionalidad que preserva relaciones locales y funciona especialmente bien con datos en variedades curvas como el "rollo suizo".

d) Un método no lineal basado en proyecciones ortogonales.

Solución AAI.EX.20230207.T.1
Respuesta correcta: **c)**

Justificación: LLE es una técnica **no lineal** que preserva las relaciones locales entre vecinos cercanos, siendo muy adecuada para datos que viven en variedades no lineales.

AAI.EX.20230207.T.2

Enunciado AAI.EX.20230207.T.2
¿Con qué objetivo se utiliza el conjunto de prueba (test) en un flujo estándar de aprendizaje automático?

a) Para estimar el error de generalización antes del despliegue en producción.

b) Para elegir modelos y ajustar hiperparámetros.

c) Para ambas cosas.

d) Para ninguno de los fines anteriores.

Solución AAI.EX.20230207.T.2
Respuesta correcta: **a)**

Justificación: el conjunto de test se reserva para **evaluar el rendimiento final** del modelo; no debe usarse para selección ni ajuste, que se hacen con el conjunto de validación.

AAI.EX.20230207.T.3

Enunciado AAI.EX.20230207.T.3
¿Por qué es importante escalar las variables de entrada al usar Máquinas de Vectores de Soporte (SVM)?

a) Para reducir el sobreajuste.

b) Porque SVM es invariante a la escala.

c) Para reducir el subajuste.

d) Para evitar que las variables con valores pequeños queden infravaloradas frente a otras de mayor escala.

Solución AAI.EX.20230207.T.3
Respuesta correcta: **d)**

Justificación: SVM depende de distancias y productos escalares; si no se escalan los datos, las características con valores grandes dominan el modelo.

AAI.EX.20230207.T.4

Enunciado AAI.EX.20230207.T.4
En un árbol de decisión, ¿cómo suele compararse la impureza de Gini de un nodo hijo con la de su nodo padre?

a) Normalmente es mayor.

b) Normalmente es menor.

c) Siempre es menor.

d) No existe relación entre ambas.

Solución AAI.EX.20230207.T.4
Respuesta correcta: **b)**

Justificación: los criterios de división buscan **reducir la impureza**, por lo que, en general, los nodos hijos son más puros que su padre (aunque no siempre estrictamente).

AAI.EX.20230207.T.5

Enunciado AAI.EX.20230207.T.5
¿Cuáles de las siguientes son razones habituales para aplicar reducción de dimensionalidad?

a) Acelerar y mejorar algoritmos posteriores.

b) Aumentar el coste computacional.

c) Facilitar la interpretación de los datos transformados.

d) a) y c) son correctas.

Solución AAI.EX.20230207.T.5
Respuesta correcta: **d)**

Justificación: reducir dimensionalidad puede **mejorar eficiencia**, reducir ruido y facilitar la **visualización e interpretación**.

AAI.EX.20230207.T.6

Enunciado AAI.EX.20230207.T.6
¿Qué algoritmos de clustering se basan en la detección de regiones de alta densidad?

a) BIRCH y K-means.

b) Mean-Shift y DBSCAN.

c) Mean-Shift y K-means.

d) DBSCAN y BIRCH.

Solución AAI.EX.20230207.T.6
Respuesta correcta: **b)**

Justificación: **DBSCAN** y **Mean-Shift** identifican clusters como regiones densas separadas por zonas de baja densidad.

AAI.EX.20230207.T.7

Enunciado AAI.EX.20230207.T.7
En un MLP con 10 entradas, una capa oculta de 50 neuronas y una capa de salida de 3 neuronas, ¿qué dimensiones tiene la matriz de salida Y?

a) 3×50

b) $M \times 3$, siendo M el tamaño del lote

c) $3 \times M$, siendo M el tamaño del lote

d) 3×10

Solución AAI.EX.20230207.T.7
Respuesta correcta: **b)**

Justificación: cada fila de Y corresponde a una instancia del lote y cada columna a una neurona de salida, por lo que $Y \in \mathbb{R}^{M \times 3}$.

AAI.EX.20230207.T.8

Enunciado AAI.EX.20230207.T.8
En la clase `KernelDensity` del paquete `sklearn.neighbors`, ¿cuál es el kernel que se utiliza por defecto?

a) `tophat`

b) `epanechnikov`

c) `exponential`

d) `gaussian`

Solución AAI.EX.20230207.T.8
Respuesta correcta: **d)**

Justificación: el kernel **gaussiano** es la opción por defecto en `KernelDensity`.

AAI.EX.20230207.T.9

Enunciado AAI.EX.20230207.T.9
¿Bajar siempre en la dirección de mayor pendiente para llegar al fondo de un valle es una analogía clásica de qué algoritmo de optimización?

a) Descenso del gradiente.

b) Algoritmo voraz.

c) Optimización aleatoria.

d) Algoritmo genético.

Solución AAI.EX.20230207.T.9
Respuesta correcta: **a)**

Justificación: el descenso del gradiente avanza iterativamente en la dirección de máxima pendiente descendente para minimizar una función.

AAI.EX.20230207.T.10

Enunciado AAI.EX.20230207.T.10
En un problema de clasificación multiclase con pocos datos, atributos independientes y distribuciones normales, ¿qué algoritmo es más adecuado si se prioriza una inferencia muy rápida?

a) SVM.

b) Árboles de decisión.

c) Naive Bayes.

d) Redes neuronales.

Solución AAI.EX.20230207.T.10
Respuesta correcta: **c)**

Justificación: **Naive Bayes** tiene inferencia extremadamente rápida y funciona bien bajo hipótesis de independencia y normalidad de las características.

Desarrollo

AAI.EX.20230207.P.1

Enunciado AAI.EX.20230207.P.1 Una compañía dedicada a la distribución mayorista de frutas cítricas establece los precios de venta de forma sem-

anal en función del tipo de producto. Los artículos se agrupan en cuatro clases principales: naranjas, mandarinas, limones y pomelos.

La empresa dispone de un histórico de datos de campañas anteriores que incluye, para cada registro, información como: tipo de fruta, semana del año, cantidad recolectada semanalmente, agricultor responsable, tamaño medio del fruto, tonalidad y precio unitario por kilogramo.

El criterio de fijación de precios es semanal y por tipo de producto. Esto implica que, dentro de una misma semana y categoría, toda la producción se comercializa al mismo precio, con independencia del productor o del volumen cosechado.

El objetivo de la empresa es desarrollar una herramienta informática capaz de aprender, a partir de los datos históricos, el comportamiento de los precios y sugerir valores adecuados para la campaña actual. No es necesario que el sistema proporcione explicaciones interpretables sobre cómo se generaban los precios en el pasado, pero sí que sea eficiente en tiempo de respuesta, incluso cuando se trabaja con grandes volúmenes de información.

Para el año en curso, se facilita un conjunto de datos que no incluye el precio por kilogramo ni la etiqueta de la categoría del producto. Se pide definir uno o varios enfoques de aprendizaje automático que permitan estimar un precio semanal por tipo de producto, incluso cuando dicha categoría no esté previamente identificada.

En este contexto, ¿resultaría imprescindible determinar primero la categoría del producto antes de estimar el precio?

Solución AAI.EX.20230207.P.1 Contexto del problema

- Históricos: variables X (semana, cantidad, agricultor, tamaño, tonalidad, tipo) y objetivo continuo y (precio €/kg) ⇒ *aprendizaje supervisado* de *regresin* (Tema 3).
- Año actual: falta el precio y y falta la categoría/tipo c (naranja/mandarina/limón/pomelo) ⇒ hay que combinar:
  - *aprendizaje no supervisado* para inferir c (Tema 4),
  - y *regresin* para estimar el precio (Tema 3).
- Restricción de negocio: un único precio por semana y tipo:

$$p = p(s, c)$$

  es decir, para todos los registros con la misma semana s y tipo c, el precio es el mismo.

Enfoque A (recomendado): Clustering (Paso A) + Árbol de decisión para regresión (Paso B)

Paso A: inferencia de la categoría mediante clustering (Tema 4)

Objetivo:

- Estimar una categoría $\hat{c}$ para cada registro del año actual a partir de variables que separen tipos.

Variables para agrupar (ejemplos):

- tamaño medio del fruto
- tonalidad
- (opcional) otras variables físico-productivas si existen

Modelo de clustering:

- K-means con $K = 4$ (porque hay 4 clases principales).
- Justificación:
  - muy eficiente y escalable con grandes volúmenes,
  - asignación rápida en inferencia.

Resultado:

- Para cada registro i, el clustering devuelve una etiqueta de grupo:

$$\hat{c}_i \in \{1, 2, 3, 4\}$$

  (después se puede mapear a naranja/mandarina/limón/pomelo usando el histórico, por ejemplo, viendo qué tipo real domina en cada cluster).

Paso B: estimación del precio semanal por producto (Tema 3)

Como la política fija un único precio por (s, c), el núcleo del modelo puede ser:

$$\hat{p} = f(s, c)$$

donde c será la categoría real en histórico y la categoría inferida $\hat{c}$ en el año actual.

Modelo elegido: Árbol de decisión para regresión (DT-regresión)

Variables:

- Entradas X:
  - semana s (numérica)
  - categoría c (codificada, por ejemplo one-hot o como entero si la implementación lo permite)
- Salida y:
  - precio €/kg

Justificación:

- Eficiente en predicción (muy rápida, adecuada para grandes volúmenes).
- Captura patrones no lineales por tramos (subidas/bajadas bruscas entre semanas).
- Encaja con la estructura real: si el precio depende sobre todo de (s, c), un árbol lo aprende con pocas particiones.

Diseño del proyecto (metodología completa)

Paso 1. Construcción del dataset histórico para el Paso B

- Si en histórico hay múltiples registros por productor con el mismo precio "oficial", es recomendable consolidar:
  - crear una tabla por pares (s, c) con un único precio objetivo (p.ej. media/mediana del precio registrado).
- Así el modelo aprende directamente la política de precios.

Paso 2. Preprocesado

- Codificar c.
- No hace falta escalado para árboles.

Paso 3. Validación

- Si hay varios años: validación temporal (entrenar en años anteriores, validar en el último).
- Métricas típicas:
  - MAE (error absoluto medio),
  - RMSE (penaliza más grandes errores).

Paso 4. Entrenamiento

- Ajustar DT-regresión en el histórico (consolidado o no).

Paso 5. Predicción en el año actual

- Para cada registro con semana s_i:
  - inferir $\hat{c}_i$ con K-means,
  - predecir $\hat{p}_i = f(s_i, \hat{c}_i)$.

Paso 6. Generación del precio "oficial" por semana y tipo (imposición de la regla de negocio)

- Para cada par (s, c) (usando $\hat{c}$ en el año actual):

$$\hat{p}(s, c) = \text{mediana}\{\hat{p}_i : \ s_i = s, \ \hat{c}_i = c\}$$

  (mediana es robusta si hay ruido).

Hiperparámetros relevantes (y efecto)

- En K-means:
  - $K = 4$ (fijado por el problema).
  - inicialización y número de reinicios (mejora estabilidad de solución).
- En DT-regresión:
  - profundidad máxima: controla complejidad; demasiada profundidad $\Rightarrow$ sobreajuste.
  - mínimo de muestras por hoja: suaviza la predicción; más alto $\Rightarrow$ menos varianza.

  - criterio de partición (MSE/MAE según implementación): cambia sensibilidad a outliers.

Enfoque B (alternativo y aún más eficiente en inferencia): "Tabla de precios por (semana, tipo)" + Paso A

Si se acepta que el precio depende únicamente de (s, c), no hace falta un modelo complejo en Paso B:

- Paso A: clustering para inferir $\hat{c}$ (igual que antes).
- Paso B:
  - del histórico se construye directamente una tabla:

$$\hat{p}(s, c) = \text{mediana del precio histórico en } (s, c)$$

  - en el año actual:

$$\hat{p}_i = \hat{p}(s_i, \hat{c}_i)$$

Ventajas:

- Respuesta extremadamente rápida (simple consulta).
- Muy robusto si el comportamiento se repite campaña a campaña.

Desventaja:

- Si hay deriva fuerte (cambio estructural de precios), generaliza peor que un modelo que use más variables.

Enfoque C (sin "clasificar primero" de forma dura): mezcla suave con GMM (Tema 4) + DT-regresión por tipo

Si en Paso A se usa un modelo de mezclas gaussianas (GMM) en lugar de K-means, se obtienen probabilidades de pertenencia:

$$P(c = k \mid \mathbf{x})$$

Entonces se puede estimar el precio sin asignar una categoría única:

- Entrenar un modelo de precio por tipo $f_k(s)$ (p.ej. DT-regresión con entradas semana y tipo fijo k).
- Predicción final:

$$\hat{p} = \sum_{k=1}^{4} P(c = k \mid \mathbf{x}) \; f_k(s)$$

Ventaja:

- Maneja incertidumbre cuando el producto está "entre dos tipos" por características.
- Evita errores bruscos por mala asignación dura.

¿Es imprescindible determinar primero la categoría antes de estimar el precio?

No es imprescindible determinarla "primero" y de forma dura, pero sí es imprescindible incorporar información equivalente al tipo.

- Si el precio se fija por (s, c), entonces el tipo c es una variable latente clave:
  - O bien se estima una etiqueta $\hat{c}$ (clustering) y luego se aplica $f(s, \hat{c})$.
  - O bien se estima una distribución $P(c \mid \mathbf{x})$ (p.ej. GMM) y se calcula un precio esperado ponderado.

Conclusión:

- **No** es obligatorio "clasificar primero" como paso separado con una etiqueta única.
- **Sí** es necesario, de alguna manera, inferir el tipo (explícita o implícitamente) porque sin esa información el problema queda mal especificado: una misma semana puede tener precios distintos según el producto.