

## **AAI.EX.20220208**

Ejercicios elaborados con fines educativos, inspirados en los contenidos evaluados en el exámen del 08/02/2022 (convocatoria Feb-2022) 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.20220228.T.1**

#### **Enunciado AAI.EX.20220228.T.1**

Al entrenar un árbol de decisión, algunas implementaciones permiten ordenar previamente las características (por ejemplo, activando una opción tipo `presort`). ¿Qué efecto suele tener esto sobre el tiempo de entrenamiento?

- a) Puede acelerar el entrenamiento cuando el conjunto tiene pocas miles (o menos) instancias.
- b) Acelera el entrenamiento en cualquier caso.
- c) No cambia la velocidad.
- d) Siempre lo hace más lento.

#### **Solución AAI.EX.20220228.T.1**

Respuesta correcta: a)

Justificación: ordenar previamente puede ahorrar trabajo repetido en datasets pequeños, pero el coste de ordenar deja de compensar cuando el conjunto crece (además, no siempre está disponible en versiones modernas).

---

### **AAI.EX.20220228.T.2**

#### **Enunciado AAI.EX.20220228.T.2**

Si la estructura relevante de tus datos es claramente no lineal (por ejemplo, una variedad curva), ¿qué variante de PCA suele ser más adecuada para capturarla?

- a) PCA estándar.
- b) PCA con kernel.

c) PCA incremental.

d) PCA aleatorizado.

#### **Solución AAI.EX.20220228.T.2**

Respuesta correcta: b)

Justificación: Kernel PCA aplica el “truco del kernel” para proyectar implícitamente los datos a un espacio donde una proyección lineal puede capturar relaciones no lineales.

---

#### **AAI.EX.20220228.T.3**

##### **Enunciado AAI.EX.20220228.T.3**

¿Cuáles son dos problemas típicos que se resuelven con aprendizaje supervisado?

a) Regresión y clasificación.

b) Reducción de dimensionalidad y clasificación.

c) Regresión y clustering.

d) Ninguna de las anteriores.

#### **Solución AAI.EX.20220228.T.3**

Respuesta correcta: a)

Justificación: en supervisado hay etiquetas. La clasificación predice clases discretas y la regresión predice valores continuos.

---

#### **AAI.EX.20220228.T.4**

##### **Enunciado AAI.EX.20220228.T.4**

En un estimador de densidad por kernels (KDE) de `sklearn.neighbors`, se puede elegir entre varios kernels predefinidos. ¿Cuál de los siguientes **no** es un nombre de kernel válido ahí?

a) `epanechnikov`

b) `tophat`

c) `polynomial`

d) `exponential`

**Solución AAI.EX.20220228.T.4**

Respuesta correcta: c)

Justificación: en KernelDensity existen kernels como gaussian, tophat, epanechnikov, exponential, linear o cosine; polynomial no es una opción.

---

**AAI.EX.20220228.T.5****Enunciado AAI.EX.20220228.T.5**

Para un MLP usado en **regresión** (salida continua), ¿qué activación es la más habitual en la capa de salida?

- a) Ninguna (salida lineal).
- b) ReLU.
- c) Softmax.
- d) Sigmoide (logística).

**Solución AAI.EX.20220228.T.5**

Respuesta correcta: a)

Justificación: en regresión se usa salida lineal para permitir valores reales sin acotarlos artificialmente (sigmoide/softmax acotan; ReLU fuerza no negatividad).

---

**AAI.EX.20220228.T.6****Enunciado AAI.EX.20220228.T.6**

¿Qué variante de descenso de gradiente, en general, puede **converger al óptimo** con una tasa de aprendizaje fija razonable (sin necesidad de programar una disminución progresiva), asumiendo un problema bien condicionado y tiempo suficiente?

- a) Descenso por lotes completos (batch).
- b) Descenso por mini-lotes (mini-batch).
- c) a), b) y d).
- d) Descenso estocástico (SGD).

**Solución AAI.EX.20220228.T.6**

Respuesta correcta: a)

Justificación: con una tasa fija suficientemente pequeña, el batch GD puede converger de forma estable. En SGD y a menudo en mini-batch, para converger (no solo “oscilar cerca”) suele requerirse reducir el learning rate con el tiempo.

---

#### **AAI.EX.20220228.T.7**

##### **Enunciado AAI.EX.20220228.T.7**

Si eliges hiperparámetros usando el conjunto de test (en vez de validación), ¿qué riesgo principal introduces?

- a) Terminar “aprendiéndote” el test (sobreajuste al test).
- b) Obtener una estimación demasiado optimista del rendimiento real.
- c) a) y b) son correctas.
- d) Forzar infraajuste específicamente al test.

##### **Solución AAI.EX.20220228.T.7**

Respuesta correcta: c)

Justificación: el test deja de ser una evaluación imparcial: ajustas decisiones mirando el test, lo que puede sobreajustarlo y hace que el rendimiento medido sea optimista.

---

#### **AAI.EX.20220228.T.8**

##### **Enunciado AAI.EX.20220228.T.8**

En detección de anomalías (no en “detección de novedades” con entrenamiento limpio), ¿qué se busca típicamente?

- a) Detectar valores atípicos solo en instancias nuevas, nunca en entrenamiento.
- b) Detectar valores atípicos solo en validación.
- c) Detectar valores atípicos tanto en el conjunto de entrenamiento como en instancias nuevas.
- d) Detectar valores atípicos solo dentro del conjunto de entrenamiento.

##### **Solución AAI.EX.20220228.T.8**

Respuesta correcta: c)

Justificación: en anomaly detection se asume que el training puede contener anomalías, así que interesa identificarlas en entrenamiento y también en produc-

ción. (En novelty detection se asume training “limpio” y se detecta principalmente en instancias nuevas.)

---

### **AAI.EX.20220228.T.9**

#### **Enunciado AAI.EX.20220228.T.9**

¿Qué tipo de forma/tamaño permiten modelar los clusters en un Modelo de Mezcla Gaussiana (GMM)?

- a) Elipses con tamaños y densidades similares.
- b) Círculos con tamaños similares.
- c) Elipses con tamaños y densidades potencialmente diferentes.
- d) No asume ninguna estructura probabilística.

#### **Solución AAI.EX.20220228.T.9**

Respuesta correcta: **c)**

Justificación: un GMM modela cada grupo como una gaussiana; con covarianzas generales, los clusters pueden ser elipsoidales y variar en tamaño y densidad (pesos y dispersión).

---

### **AAI.EX.20220228.T.10**

#### **Enunciado AAI.EX.20220228.T.10**

Pensando en rapidez para llegar “cerca” de un buen mínimo (aunque luego fluctúe alrededor), ¿qué método suele aproximarse antes a la vecindad de la solución óptima?

- a) Mini-batch gradient descent.
- b) Stochastic gradient descent (SGD).
- c) Batch gradient descent.
- d) a), b) y c).

#### **Solución AAI.EX.20220228.T.10**

Respuesta correcta: **b)**

Justificación: SGD realiza actualizaciones muy frecuentes (una instancia cada vez), lo que suele permitir progreso rápido al principio y alcanzar pronto una zona cercana al óptimo, aunque con más ruido que batch.

## **AAI.EX.20220208.D**

### **AAI.EX.20220208.D.1**

Puntuación máxima: 5 puntos Extensión máxima orientativa: 2 caras

#### **Enunciado AAI.EX.20220208.D.1**

Una empresa quiere abrir un concesionario de coches y necesita decidir en qué punto del mapa ubicarlo. Tras analizar aperturas recientes de concesionarios, se dispone de un conjunto de datos donde, para cada apertura, se registran:

- Coordenadas de la ubicación (latitud y longitud, ambas numéricas)
- Tamaño de la exposición (número de coches en exposición)
- Tamaño del equipo comercial (número de vendedores)
- Presupuesto inicial disponible
- Nivel de ventas mensuales alcanzado, no como cifra exacta sino como categoría: bajo, medio o alto

Para el nuevo concesionario se conocen el presupuesto inicial y el número de vendedores (y, si se desea, el tamaño de la exposición). El objetivo es recomendar una ubicación que maximice la probabilidad de alcanzar ventas altas.

Contesta razonadamente:

- Diseña una red neuronal sencilla para abordar el problema.
- Explica qué es una neurona artificial y qué es un perceptrón multicapa.
- Dado que latitud/longitud pueden ser negativas, ¿qué harías si solo quisieras sugerir ubicaciones con coordenadas positivas?

### **Solución AAI.EX.20220208.D.1**

#### **Planteamiento:**

Se dispone de datos de concesionarios recientes con variables numéricas (longitude, latitud, coches en exposición, nº vendedores, presupuesto) y una etiqueta categórica del rendimiento de ventas mensual (alto/medio/bajo). El cliente fija presupuesto y nº de vendedores (y, si se desea, coches en exposición) y quiere una localización que maximice la probabilidad de “alto”.

Una forma práctica de plantearlo con redes neuronales es:

- 1) entrenar un modelo que aprenda  $P(\text{ventas} \in \{\text{alto, medio, bajo}\} | \text{features})$
- 2) para un perfil dado del cliente, buscar (longitude, latitud) que maximicen  $P(\text{alto})$

---

#### **1) Red neuronal sencilla para resolver el problema:**

**\*\*Entrada (features)\*\*:**

- Longitud, Latitud
- Coches en exposición
- N° vendedores
- Presupuesto inicial

**\*\*Salida\*\*:**

- 3 neuronas (alto/medio/bajo) con activación softmax para obtener probabilidades:  
 $\hat{p} = \text{softmax}(z)$

**\*\*Arquitectura mínima (MLP)\*\*:**

- Capa de entrada: 5 variables numéricas
- 1 o 2 capas ocultas densas (por ejemplo 16 y 8 neuronas) con ReLU
- Capa de salida: 3 neuronas con softmax

**\*\*Entrenamiento\*\*:**

- Pérdida: entropía cruzada (cross-entropy)
- Optimización: Adam
- Preprocesado: estandarizar variables numéricas (muy importante en redes)

**\*\*Cómo recomendar localización\*\***

Con presupuesto y vendedores fijados por el cliente:

- Se define un rango plausible de long/lat (por ejemplo, el "bounding box" observado o zonas)
- Se evalúa el modelo en una rejilla de puntos o se usa un optimizador para maximizar:  
 $\max_{\text{lon}, \text{lat}} P(\text{alto} | \text{lon}, \text{lat}, \text{vendedores}, \text{presupuesto}, \text{exposición})$
- Se devuelve la localización con mayor probabilidad de "alto" (o las top-k localizaciones)

**2) ¿Qué es una unidad neuronal? ¿Qué es un perceptrón multicapa?**

**\*\*Unidad neuronal (neurona artificial)\*\***

Es un bloque que calcula una combinación lineal de entradas y aplica una función no lineal:

```
$$
z = w^T x + b, \quad a = \phi(z)
$$
```

- \$x\$ son las entradas, \$w\$ los pesos, \$b\$ el sesgo
- \$\phi\$ es la activación (ReLU, sigmoide, etc.)

**\*\*Perceptrón multicapa (MLP)\*\***

Es una red formada por varias capas de neuronas:

- una capa de entrada
- una o más capas ocultas (con activaciones no lineales)

- una capa de salida

Al apilar capas, el MLP puede aproximar relaciones no lineales complejas entre variables (p)

3) Si solo se desean localizaciones con long/lat positivas, ¿qué habría que hacer?

Hay dos enfoques típicos:

\*\*A) Restringir el espacio de búsqueda (recomendación)\*\*

Durante la fase de recomendación (optimización o búsqueda en rejilla), imponer la restricción

- longitud > 0
- latitud > 0

Así, aunque el modelo pueda evaluar cualquier valor, solo se proponen ubicaciones positivas.

\*\*B) Transformar la salida para forzar positividad\*\*

Si se quiere que la red "genere" directamente long/lat (en un modelo generativo/inverso), se

- usar activaciones que produzcan valores no negativos, como ReLU o softplus en la salida:  
\$\$\text{softplus}(z)=\log(1+e^z)\$\$  
Esto garantiza  $\text{long/lat} \geq 0$ .

En este enunciado (modelo que puntúa probabilidad de "alto"), lo más simple y correcto es la