
RESULTADO DE APRENDIZAJE

RdA de la asignatura:

- **RdA 2:** Aplicar modelos de aprendizaje automático supervisado y no supervisado, así como su validación y optimización, en la resolución de problemas tanto reales como simulados.

RdA de la clase:

- Comprender los fundamentos del algoritmo kNN, sus ventajas, desventajas y parámetros principales.
- Implementar el algoritmo kNN en Python y evaluar su desempeño en un problema práctico.

INTRODUCCIÓN

Pregunta inicial: Imagina que estás organizando una fiesta y no sabes si alguien prefiere pizza o sushi. Si conoces las preferencias de sus amigos más cercanos, ¿podrías predecir su elección? ¿Cómo lo harías?

DESARROLLO

Actividad 1: Socialización de la Clase Invertida

Revisión de los conceptos trabajados individualmente con ChatGPT sobre los conceptos de aprendizaje supervisado.

¿Cómo lo haremos?

- **Dinámica de discusión:** Se seleccionarán estudiantes al azar para definir los siguientes conceptos basándose en su interacción previa con el asistente virtual:
 1. ¿Dónde se puede aplicar aprendizaje supervisado?
 2. ¿Qué modelos existen de aprendizaje supervisado para regresión?
 3. ¿Qué modelos existen de aprendizaje supervisado para clasificación?
 4. ¿Cuál es el significado de «underfit»?
 5. ¿Cuál es el significado de «overfit»?

Actividad 2: Introducción al algoritmo kNN

En esta actividad los estudiantes conocerán el algoritmo kNN mediante visualización de video, interacción con ChatGPT y clase magistral, explorando sus fundamentos, ventajas, desventajas y parámetros clave.

¿Cómo lo haremos?

- **Proyección de video:** Se presentará un video educativo que explique el algoritmo k-Nearest Neighbors.

Enlace al video: [K Nearest Neighbors](#).

- **Interacción con ChatGPT:** Después del video, cada estudiante formulará preguntas a ChatGPT relacionadas con los términos o conceptos nuevos.
- **Clase magistral:** Se explicará cómo funciona kNN en detalle:
 - Concepto de «vecinos más cercanos».
 - Parámetros importantes (valor de k , peso por distancia).
 - Ventajas y desventajas del algoritmo.
- **Implementación en Python:**
- **Experimentación:**

Actividad 2: Implementación práctica del algoritmo kNN

En esta actividad los estudiantes implementarán el algoritmo kNN mediante exploración de cuaderno de Jupyter, trabajando con un ejemplo guiado y experimentando con diferentes valores de k , métricas de distancia y tamaños de conjunto de prueba.

¿Cómo lo haremos?

- **Explicación de parámetros:** Se explicarán los parámetros clave del algoritmo: valor de k (número de vecinos), métricas de distancia (euclidean, manhattan) y su impacto en el rendimiento del modelo.
- **Implementación en Python:** Los estudiantes accederán a un cuaderno de Jupyter previamente preparado.

Enlace al cuaderno: [10-kNN.ipynb](#).

- **Experimentación:**

CIERRE

Verificación de aprendizaje:

1. ¿Cómo decide kNN la clase de un punto nuevo cuando $k=3$?
2. ¿Por qué es importante normalizar los datos antes de aplicar kNN?
3. ¿Cuáles son las principales ventajas y desventajas del algoritmo kNN?

Preguntas tipo entrevista:

1. Si aumento k de 1 a 100, ¿siempre mejorará la generalización del modelo porque considera más vecinos?

Tarea: Desarrollar los ejercicios planteados en el siguiente cuaderno y entregarlo por el aula virtual:

Enlace al cuaderno: [05-kNN.ipynb](#).

Pregunta de investigación:

1. ¿Qué sucede con el rendimiento de kNN cuando el tamaño del dataset crece significativamente, y cómo podrías abordar este problema?
2. ¿Qué tan sensible es kNN a datos con ruido o valores atípicos, y cómo podrías preprocesar los datos para mitigar estos efectos?

Para la próxima clase:

- Visualizar el video [¿Qué es una SVM?](#).
- Leer las secciones 10.1, 10.2, 10.3, 10.4 y 10.5 del artículo [Beginning with machine learning: a comprehensive primer](#).