
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.

Resultados específicos:

- Comprender el método de retropropagación y su deducción matemática.
- Revisar las técnicas de optimización en el entrenamiento de redes neuronales.

INTRODUCCIÓN

Pregunta inicial: ¿Qué desafíos enfrenta el entrenamiento de redes neuronales y cómo podemos mitigarlos para mejorar su desempeño?

DESARROLLO

Actividad 1: Deducción del método de retropropagación

¿Cómo lo haremos?

- **Clase magistral:** Explicación detallada del método de retropropagación:
 - Introducción a la retropropagación y su propósito en redes neuronales.
 - Deducción paso a paso del algoritmo, desde la función de pérdida hasta el cálculo de gradientes usando la regla de la cadena.

Verificación de aprendizaje:

- ¿Cómo se utiliza la regla de la cadena en el cálculo de gradientes?
- ¿Qué rol desempeña la función de pérdida en la retropropagación?
- ¿Por qué es fundamental calcular los gradientes de manera eficiente en redes neuronales?

Actividad 2: Regularización y optimización en redes neuronales

¿Cómo lo haremos?

- **Clase magistral:** Presentación de conceptos clave:
 - Regularización: L1, L2 y Dropout.
 - Early stopping como estrategia para evitar el sobreajuste.
 - Diferentes optimizadores: SGD, Adam, RMSProp, etc.
 - Learning rate adaptativo y su impacto en el entrenamiento.
- **Ejercicio práctico:** Implementación guiada en un notebook de Python:
 - Configurar y entrenar una red neuronal básica.
 - Aplicar técnicas de regularización y evaluar resultados.
 - Comparar el rendimiento utilizando diferentes optimizadores.
 - Implementar early stopping y observar su impacto en el sobreajuste.

Verificación de aprendizaje:

- ¿Cómo afecta la regularización al sobreajuste?
- ¿Cuáles son las diferencias principales entre SGD y Adam como optimizadores?
- ¿Qué ventajas ofrece el learning rate adaptativo frente a un learning rate constante?

CIERRE

Tarea: Desarrollar los ejercicios planteados en el siguiente cuaderno, usando mejoras en el entrenamiento de la red, y entregarlo por el aula virtual:

Enlace al cuaderno: [07-Perceptron.ipynb](#).

Preguntas para reflexionar:

1. ¿Qué técnicas de regularización funcionan mejor en diferentes escenarios?
2. ¿Cómo podrías mejorar el desempeño de una red neuronal sin cambiar su arquitectura?
3. ¿Cómo afecta el tamaño del dataset al rendimiento del modelo y la necesidad de regularización?

Para la próxima clase: Exploración de arquitecturas avanzadas como redes convolucionales y recurrentes. Previsualizar el notebook: [Arquitecturas Avanzadas](#).