

## **DETALLES SOBRE EL REPOSITORIO**

Este repositorio contiene dos algoritmos de clasificación con sus respectivos datasets. El primero de ellos es una regresión lineal simple que clasifica a personas como "Obesa" o "No obesa" dependiendo de la relación entre su peso y su altura. Por otro lado, el segundo es una red neuronal de dos capas, una oculta y otra de salida, con cuatro neuronas en la capa intermedia y solo una en la final. El objetivo de esta es clasificar de igual manera a personas como "Obesa" o "No obesa". Cabe destacar que ambos funcionan correctamente, sin embargo, el segundo algoritmo es más complejo que el primero.

### **1. REGRESIÓN LINEAL**

El modelo realizado con regresión lineal entrena con un set de entrenamiento bastante pequeño, y posteriormente se prueba con un set de prueba aún más corto. No existen datos de validación en este caso ya que al ser un problema muy simple, no es necesario contar con ellos. Además, si así fuera los sets de prueba y entrenamiento serían aún más pequeños, y es que gracias al tamaño de ambos, los resultados no son para nada buenos. Alcanza una precisión de 0.79 con los datos de entrenamiento, pero esta disminuye a 0.54 con los datos de prueba. Los valores para la sensibilidad y el f1-score tampoco son tan altos como se buscaba. Esto se debe a que en ambos casos, confunde bastante a la gente "No obesa", clasificándola como "Obesa". En conclusión es un modelo descente que busca reducir el error entre lo predicho y lo real, pero aún se queda lejos de tener excelentes resultados.

### **2. RED NEURONAL**

El segundo modelo, implementa una red neuronal de dos capas con cuatro neuronas en la capa oculta, una en la capa de salida, y una función de activación sigmoide por neurona. Lo primero que hace es inicializar los pesos y sesgos de la red para que en cada iteración del entrenamiento, se realice la propagación hacia adelante para calcular las salidas. Posteriormente, se ejecuta la retropropagación para actualizar los parámetros utilizando el descenso de gradiente. Tras entrenar la red, se evalúa su desempeño en los conjuntos de entrenamiento, validación y prueba, calculando y mostrando sus respectivas matrices de confusión, y métricas como precisión, sensibilidad y F1 score. El código se realiza con la ayuda de funciones, esto con la finalidad de optimizar y simplificar el proceso de entrenamiento. Cabe destacar que este modelo funciona bastante mejor que el primero, ya que alcanza una precisión de 0.9178, 0.9357 y 0.91 para los datos de entrenamiento, validación y prueba en ese respectivo orden. Además, los valores de sensibilidad y F1 score están por arriba de 0.9 en los tres conjuntos. En conclusión, esta red neuronal proporciona un modelo más eficiente que el de regresión lineal, pero a su vez, es más complejo.