Comparación de Regresión y Regresión Logística

Instrucciones generales: La presente tarea-laboratorio consta de dos partes. En la primera se basa principalmente en conocimiento teóricos, por lo que se deberá responder lo solicitado. En la segunda parte, es la que se basa en poner en práctica lo aprendido durante esta semana.

|  |  |
| --- | --- |
| Parte 1 | Preguntas Teóricas |

Instrucciones: Responda de forma clara y lo más breve posible cada una de las siguientes preguntas

1. ¿Qué es un perceptrón y cómo funciona?

Es un modelo de clasificación binaria, base de las redes neuronales.

Está compuesto por entradas (features), pesos, sesgo (bias) y función de activación.

Las entradas son características de los datos, los pesos indican la importancia de c/entrada, el sesgo permite ajustar el modelo independientemente de las entradas y la función de activación convierte la combinación lineal de entrads y pesos en una decisión (0 ó 1).

1. ¿Qué diferencia a un perceptrón de un perceptrón multicapa (MLP)?

Un MLP tiene pocas capas ocultas, modela relaciones no lineales simples. Mientras que un perceptrón simple solo tiene una capa de procesamiento y solo puede aprender relaciones lineales en las entradas y salida.

1. ¿Qué es la función de activación en un perceptrón y por qué es importante?

La función de activación en un perceptrón es la encargada de decidir si una neurona “se activa” o no en función del valor calculado a partir de las entradas y sus pesos.

1. ¿Por qué un perceptrón simple no puede resolver problemas no lineales como el problema XOR?

El perceptrón simple solo puede separar clases con una línea recta, solo puede resolver problemas linealmente separables. XOR requiere al menos dos fronteras de decisión combinadas.

|  |  |
| --- | --- |
| Parte 2 | Práctica |



Task único– Perceptrón

Este conjunto de datos fue extraído de la base de datos del Censo de 1994 por Ronny

Kohavi y Barry Becker (especialistas en minería de datos y visualización en Silicon Graphics). Se seleccionaron registros relativamente limpios aplicando las siguientes condiciones: personas mayores de 16 años (AAGE > 16), con ingresos superiores a 100 dólares (AGI > 100), un peso final mayor a 1 (AFNLWGT > 1) y que trabajen al menos una hora (HRSWK > 0). El objetivo es predecir si una persona gana más de 50,000 dólares al año.

.

Link del dataset: [https://www.kaggle.com/datasets/uciml/adult-censusincome?select=adult.csv](https://www.kaggle.com/datasets/uciml/adult-census-income?select=adult.csv)

El objetivo es utilizar el conjunto de datos del Censo de 1994 para construir y evaluar un modelo de perceptrón simple y un perceptrón multicapa (MLP), utilizando bibliotecas de Python con el fin de predecir si una persona gana m[as de 50K dólares al año.

A continuación, se detallan las instrucciones para construir los modelos utilizando Python y Jupyter Notebook, asegurándonos de realizar un adecuado preprocesamiento de datos y una división correcta en conjuntos de entrenamiento y prueba.

Instrucciones para Crear un Modelo de Perceptrónx

1. Preparación del entorno
   1. Asegúrate de tener Python instalado, junto con las bibliotecas necesarias: NumPy, Pandas, scikit-learn.
   2. Crea un entorno de trabajo limpio para el desarrollo del proyecto..
2. Carga y Preprocesamiento del Conjunto de Datos
   1. Carga el conjunto de datos proporcionado (en formato CSV u otro compatible).
   2. Identifica las variables predictoras (características) y la variable objetivo (etiqueta de clase).
   3. Realiza un análisis exploratorio básico para identificar valores nulos, datos atípicos y la distribución de las variables.
   4. Convierte las variables categóricas en numéricas mediante técnicas de codificación, como la codificación de etiquetas o variables ficticias.
   5. Estandariza o normaliza las variables numéricas para mejorar el rendimiento del modelo.
   6. Divide el conjunto de datos en conjuntos de entrenamiento y prueba, utilizando una proporción adecuada (por ejemplo, 80% entrenamiento y 20% prueba).
3. Construcción del Modelo de Perceptrón Simple
   1. Utiliza un clasificador de perceptrón simple para entrenar el modelo con el conjunto de datos de entrenamiento.
   2. Ajusta los parámetros de entrenamiento, como la tasa de aprendizaje y el número máximo de iteraciones.
   3. Evalúa el modelo utilizando el conjunto de prueba para obtener métricas de rendimiento como la matriz de confusión, precisión, recall y F1-score.
4. Construcción del Perceptrón Multicapa (MLP)
   1. Implementa un perceptrón multicapa utilizando un clasificador adecuado.
   2. Define la arquitectura de la red neuronal, incluyendo el número de capas ocultas, el número de neuronas por capa y la función de activación.
   3. Ajusta los hiperparámetros relevantes, como el algoritmo de optimización, la tasa de aprendizaje y el número de iteraciones.
   4. Evalúa el modelo con el conjunto de prueba y analiza las métricas de rendimiento obtenidas.

1. Comparación de Resultados
   1. Compara el rendimiento del perceptrón simple y el MLP en función de las métricas de evaluación.
   2. Analiza las diferencias en la capacidad de generalización de ambos modelos.
   3. Reflexiona sobre cómo la complejidad del modelo afecta su capacidad para resolver problemas lineales y no lineales.
2. Análisis y Conclusiones
   1. Describe las principales observaciones obtenidas durante el desarrollo del ejercicio.
   2. Explica qué modelo mostró un mejor rendimiento y por qué.
   3. Identifica posibles mejoras para optimizar el desempeño de los modelos, como el ajuste de hiperparámetros o la ingeniería de características.

Entrega:

Presenta un informe que incluya el análisis de los datos, la metodología utilizada, los resultados obtenidos y las conclusiones finales.

Incluye gráficos o tablas relevantes que respalden tu análisis.

Recuerda presentar también el Jupyter Notebook usado. (el informe puede ser parte del mismo notebook)

*Fecha de entrega: 15 días de plazo*