

Centro universitario de ciencias exactas e ingeniería



DIVISIÓN ELECTRÓNICA Y COMPUTACIÓN



Práctica 1.- Ejercicio 3

Division de tecnologías para la integración Ciber-Humana

Departamento de Ciencias Computacionales.

Materia: Seminario de Inteligencia Artificial 2

Profesor: Diego Campos Peña

Alumno: Jose Emmanuel Vazquez Haro

Código: 219747476

Carrera: Ingeniería en Computación

Sección: D05

Ciclo: 2024A

Fecha: 15/04/2024



Introducción:

Realizar el ejercicio 3 del archivo Práctica1.pdf.

- Realiza los programas en el lenguaje de tu elección.
- Sube un reporte en formato PDF y los programas correspondientes en un repositorio de git.

Se realizará un algoritmo de retropropagación se repite a lo largo de múltiples iteraciones o épocas de entrenamiento hasta que el error de predicción alcance un nivel aceptable o converja hacia un mínimo. Es importante destacar que el éxito del entrenamiento de una red neuronal.

Desarrollo:

Como primer paso del problema desarrollaremos un algoritmo de retropropagación para nuestro problema de un perceptrón multicapa y para el desarrollo es necesario contar con una serie de especificaciones para su buena ejemplificación.

- `concentlite.csv`: este archivo que nos brindó el profesor contiene dos clases distribuidas de forma concéntrica.

Es necesario comprender que lo que esperamos de nuestro algoritmo es que sea capaz de entrenar y probar dataset que le estamos mandando a llamar para que tenga la capacidad de clasificar el perceptrón multicapa.

Partes del programa:

- Como primera parte del programa después de la carga de las dependencias necesarias es la carga y preparación de datos el cual como se mencionó anteriormente es desde el archivo anexado en la actividad, luego dividimos en características y etiquetas.
- Como segunda parte de la estructura de la red neuronal establecemos la arquitectura de la red, donde incluimos el tamaño de la capa de entrada, las ocultas y las de salida.



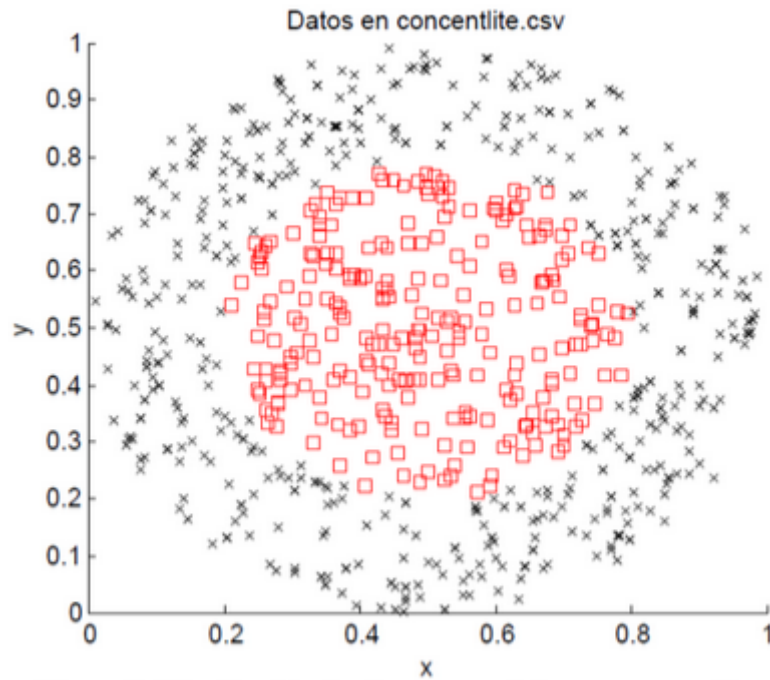
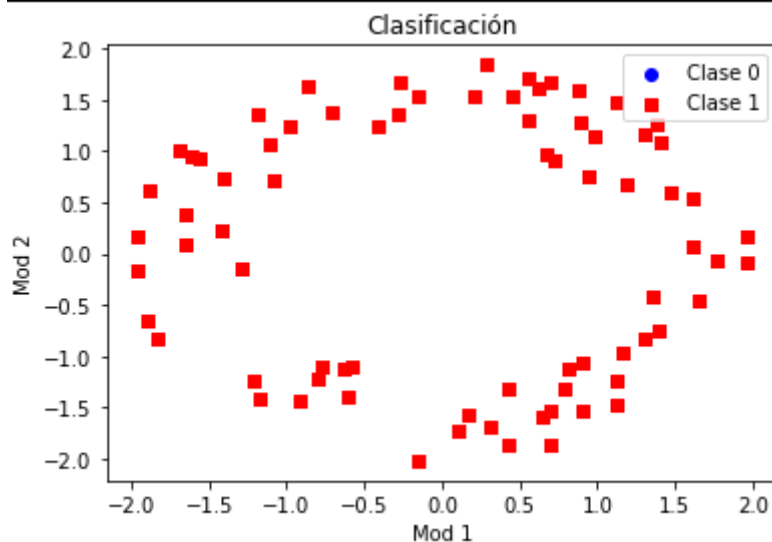


Figura 2. Distribución de clases para el dataset concentrite.

- Inicialización de pesos y sesgos de forma aleatoria.
- Cómo cuarta parte realizaremos el entrenamiento de la red mediante un bucle que itera sobre un número fijo de épocas.
- Después del entrenamiento utilizamos la red entrenada para hacer la predicción de conjunto de prueba y finalmente visualizamos los resultados en un gráfico de dispersión.



Conclusión:

Como conclusión el algoritmo desarrollado nos proporciona una implementación de una red neuronal de retropropagación para la clasificación binaria. En el cual al cargarle nuestro conjunto de datos nos permite dividir lo entrenado con lo probado, esto como vimos es esencial para entrenar perceptrones multicapa al ajustar los pesos en los que la red pueda aprender y con esto realizar tareas automáticas. Lo aprendido en esta práctica es la capacidad que tiene de aprender de manera eficiente a partir del conjunto de datos, el ajuste que realizamos a los pesos y sesgos en función a la salida del error propagando las capas de la red.

Repositorio de Git: <https://github.com/EHaroo/SemIA2.git>

