

Inteligencia Computacional

Guía de trabajos prácticos 2

Perceptrón multicapa

1. Objetivos

- Profundizar en los conceptos teóricos relacionados con la retropropagación del error.
- Analizar y utilizar un algoritmo de entrenamiento completo para el perceptrón multicapa.
- Aplicar una arquitectura de redes neuronales a la clasificación automática de datos reales.
- Conocer más alternativas de las técnicas de validación cruzada.

2. Trabajos prácticos

Ejercicio 1: Resuelva el problema de clasificación XOR mediante un perceptrón multicapa con la mínima cantidad posible de neuronas. Calcule el valor de los pesos a partir del análisis de las regiones de clasificación mediante hiperplanos y pruebe la red neuronal mediante un programa computacional.

Ejercicio 2: Realice una búsqueda de implementaciones del perceptrón multicapa, con código abierto y en el lenguaje de su elección. Seleccione una que permita elegir libremente la cantidad de capas y de neuronas en cada capa de la red, y analice *en detalle* el código correspondiente al algoritmo de retropropagación. Para entrenar y probar el algoritmo utilice dos clases con las siguientes distribuciones:

- rectángulo de vértices $(2,2)$, $(-2,2)$, $(2,-2)$ y $(-2,-2)$ con un hueco de vértices $(1,1)$, $(-1,1)$, $(1,-1)$ y $(-1,-1)$,
- círculo de radio $\sqrt{2}/2$ centrado en $(0,0)$.

Una vez realizado el entrenamiento genere patrones al azar dentro de un área rectangular de vértices $(3,3)$, $(-3,3)$, $(3,-3)$ y $(-3,-3)$ y represente gráficamente, con diferentes colores, el resultado de la clasificación realizada por el perceptrón multicapa.

Ejercicio 3: Verifique experimentalmente la influencia de la incorporación del término de momento en la ecuación de adaptación de los pesos. Para ello utilice el mismo código del ejemplo anterior y, de ser necesario, modifíquelo para incluir el término de momento. Compare la velocidad de convergencia del algoritmo para el caso de la base de datos `concent.csv`, que consiste en dos clases distribuidas en forma concéntrica como muestra la Figura 1.

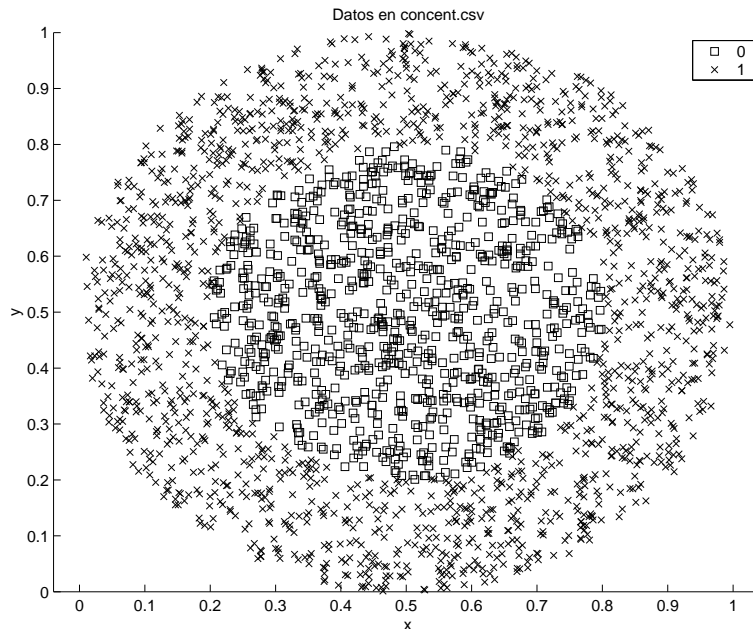


Figura 1: Distribución de clases para la base de datos `concent.csv`.

Ejercicio 4: *Iris* es el género una planta herbácea con flores que se utilizan en decoración. Dentro de este género existen muy diversas especies entre las que se han estudiado la *Iris setosa*, la *Iris versicolor* y la *Iris virginica* (ver Figura 2).

Estas tres especies pueden distinguirse según las dimensiones de sus pétalos y sépalos. Un grupo de investigadores ha recopilado la información correspondiente a las longitudes y anchos de los pétalos y sépalos de 50 plantas de cada especie. En el archivo `iris.csv` se encuentran estas mediciones (en cm.) junto con un valor numérico que indica la especie reconocida por los investigadores (0=setosa; 1=versicolor; 2=virginica). Para la clasificación

de una gran cantidad de estas plantas se desea crear un programa que aprenda de estos 150 patrones para luego realizar la tarea de forma automática.

Para la validación utilice los métodos *leave-k-out* y *leave-one-out* con un perceptrón multicapa como clasificador. Estime error de clasificación promedio y la desviación estándar según los dos métodos.



Figura 2: Muestra de la especie *Iris virginica*.

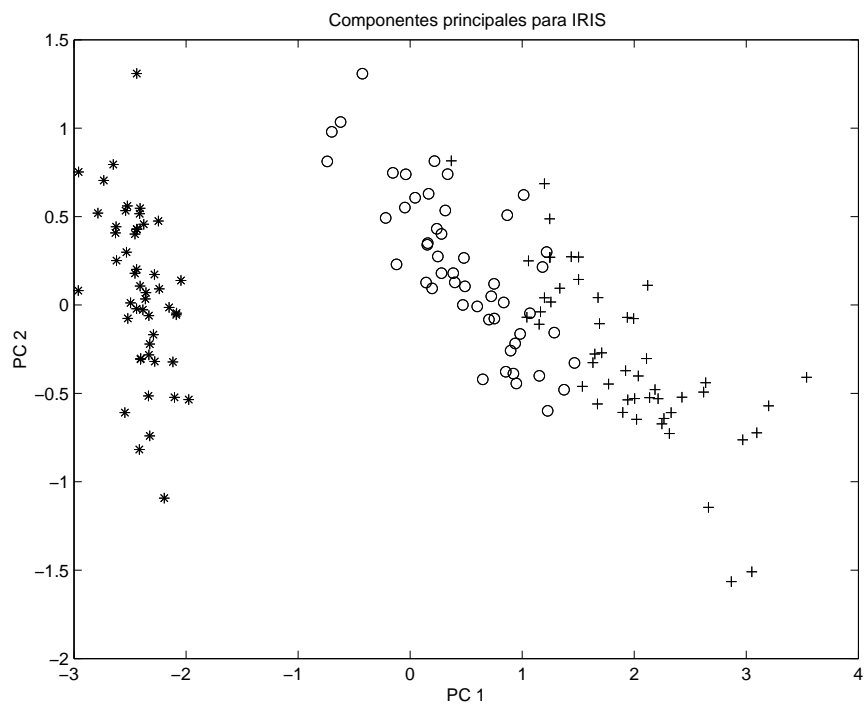


Figura 3: Proyección en \mathbb{R}^2 de la distribución de clases para la base de datos `iris.csv`.