

2017

# Fundamentos de los Sistemas Inteligentes

MEMORIA DE PRÁCTICAS

ZABAI ARMAS HERRERA Y HÉCTOR DÉNIZ ÁLVAREZ

# Índice

1. [Introducción](#)
2. [Prácticas](#)
  - A. [Búsquedas](#)
  - B. [Redes Neuronales](#)
  - C. [Aprendizaje por refuerzo](#)
3. [Conclusión](#)

## 1. Introducción

En esta memoria de las prácticas de la asignatura explicaremos brevemente y a groso modo nuestro desarrollo en las implementaciones. Esto implica los algoritmos, métodos y puntos de vista de nosotros.

## 2. Prácticas

El código fuente de todas las prácticas e incluso este documento, se pueden clonar del siguiente repositorio GitHub:

<https://github.com/Zabai/FSL.git>

### 2.A Búsquedas

Esta práctica consta de 2 partes: la búsqueda no informada (ramificación y acotación) y la informada (ramificación y acotación con subestimación).

A partir del código base de ramificación y acotación tuvimos que implementar una nueva clase de lista llamada “Sorting”, porque en esencia lo único que varía es que antes de expandir los nodos que contenga los ordena por el coste que tengan.

En la segunda parte creamos nuestro propio “graph\_search” para eliminar la lista cerrada, además de otra clase lista “Sorting2” (sí, somos muy originales con los nombres) que ordena los nodos por su coste real y heurístico.

Aquí dejamos una tabla comparativa de los nodos expandidos según distintas rutas y algoritmos de búsqueda utilizados.

|       | Anchura | Profundidad | Ramificación y salto | Ramificación y salto con subestimación |
|-------|---------|-------------|----------------------|--|
| A → B | 10      | 9           | 43                   | 5                                      |
| F → Z | 9       | 17          | 14                   | 4                                      |
| V → R | 11      | 13          | 28                   | 6                                      |
| I → A | 14      | 8           | 232                  | 15                                     |

## 2.B Redes Neuronales

Al igual que la práctica anterior, esta también consta de 2 tareas: clasificación del iris y utilización del conjunto de datos MNIST.

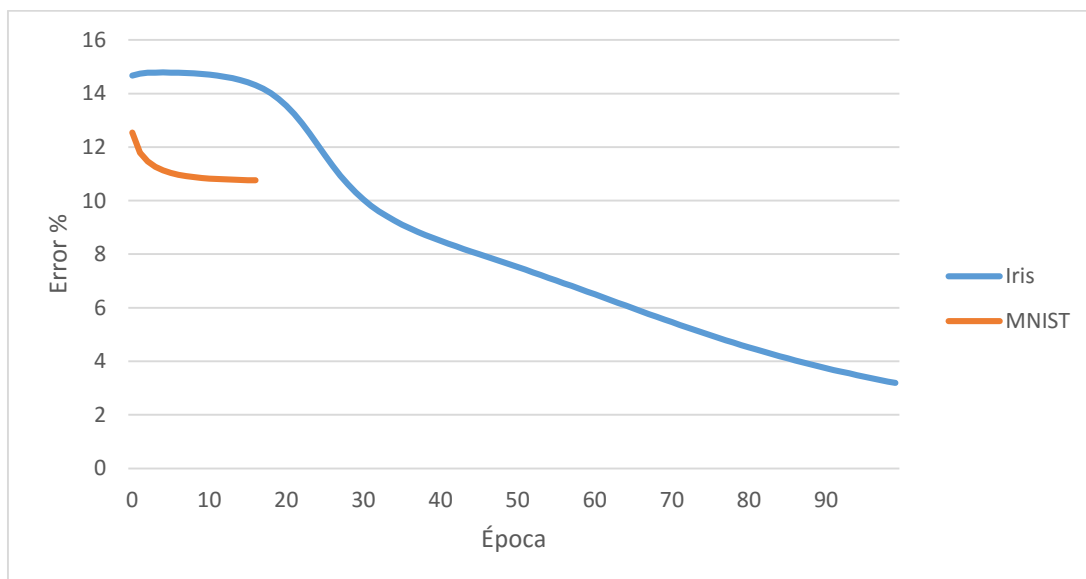
El procedimiento de estas tareas se puede resumir en los siguientes pasos:

1. Diseñar la estructura de la red neuronal (entradas, neuronas, capas)
2. Adaptar el conjunto de datos y dividirlo
3. Empezar a entrenar a la vez que validar
4. Una vez terminado el entrenamiento, efectuar las pruebas que determinan la tasa de fallos

La estructura del iris se define como 4 entradas (alto y ancho del sépalo; alto y ancho del pétalo) que van a 5 neuronas y las salidas de estas van a otras 3 neuronas.

Con el MNIST se empieza a tener un grado de complejidad ya que, posee unas  $28 \times 28$  entradas (píxeles de las imágenes) que van a 10 neuronas. Esta, por el contrario que el iris, no tiene más capas.

Las siguientes gráficas representan el aprendizaje, es decir, la tasa de fallos por época.



## 2.C Aprendizaje por refuerzo

La última práctica consiste en implementar políticas de aprendizaje, en concreto: greedy y e-greedy.

La política greedy se centra en aprender si no conoce una ruta mejor, ya que se dedicará a explotar las que tengan mayor porcentaje. En cambio, la e-greedy tiene un porcentaje que tendrá en cuenta para explorar otras opciones.

En la siguiente gráfica vemos el promedio de acciones por episodio según las políticas de aprendizaje por refuerzo.



### **3. Conclusión**

Este conjunto de prácticas da al estudiante la correcta introducción sobre varios campos de la inteligencia artificial. Empezando desde las búsquedas informadas con heurísticas hasta el aprendizaje por refuerzo.

Es cierto que nos gustaría dar en profundidad estos temas, pero como siempre el tiempo es el principal problema.