

Inteligencia Artificial: Práctica 5

1. Construcción de clasificadores en bases de datos sintéticas

- *Número de vecinos en k-nn. ¿Por qué siempre debe ser impar cuando hay dos clases?*

Debe ser impar pues, como vecinos próximos se basa en el máximo número de clases próximas al dato que queremos clasificar, si tomamos un número par de vecinos próximos, puede pasar que la mitad de los vecinos pertenezcan a una clase y la otra mitad a la otra clase, de modo que no podemos saber a cuál de las dos clases pertenecería nuestro dato. En cambio, si el número de vecinos próximos que tomamos es impar, siempre va a haber más de una clase que de otra, siendo más exacta nuestra clasificación.

- *Profundidad máxima de los árboles de decisión.*

Cuanta más profundidad pongamos, mejor será el resultado. Sin embargo, llega un punto en el que se alcanza un máximo y no sigue mejorando. Se puede observar que con una profundidad de 8, el resultado ya es mucho mejor que con 2.

- *Número de neuronas en la red neuronal y máximo número de épocas de entrenamiento.*

Se puede observar que aumentar el número de neuronas y aumentar el número de capas ocultas, varía muy poco los resultados. Los espacios sin clasificar en blanco se vuelven cada vez más pequeños. Pero la diferencia es tan pequeña con pocas capas que no merece la pena aumentarlas. Sin embargo, si bajamos el dato inicial sí se puede observar una gran diferencia a peor. Luego 50 neuronas es un buen dato.

2. Construcción de un clasificador en una base de datos real

- *¿Cuál es el mejor score que consigues con un k-nn y con qué k (valor de n_neighbours)?*

El mejor score que consigo es con 13 vecinos próximos pues sale 0.76 con un error de +/-0.04 de modo que como mucho alcanzaré 0.8 y como mínimo 0.72. Si seguimos aumentando el número de vecinos, el score empeora, sin embargo, entre 11, 13 y 15 vecinos próximos hay muy poca diferencia y a nivel estadístico no se notaría.

- *¿Cuál es el mejor score que consigues con un árbol de decisión y con qué profundidad máxima (valor de max_depth)?*

El mejor score es con una profundidad de 2 pues sale 0.74 con un error de 0.01. Si ponemos una profundidad de 5, también nos sale un score bueno y mejor, pero el error es

mayor, luego 2 es mejor profundidad. A partir de una profundidad de 9 el score se mantiene variando únicamente el error.

- ¿Cuál es el mejor score que consigues con una red neuronal y con qué configuración (valor de hidden_layer_sizes)?

Tras probar varias opciones, la mejor opción es con 150 neuronas, sin embargo, con una sola capa obtengo un error de 0.05. Si reparto esas 150 neuronas en 2 capas de modo que cada una tenga 75 neuronas, el resultado se mantiene igual pero los errores, aunque sean aleatorios cada vez y puedan variar, son menores a con una sola capa, estando alrededor del 0.67 con un error de 0.02. Si reparto las 150 neuronas en 3 capas el resultado sería más o menos igual.