

## PRÁCTICA: BASADOS EN INSTANCIAS

En el módulo Sklearn de Python hay un conjunto de datos de dígitos manuscritos. Con él se va a realizar esta práctica. En síntesis, cada instancia es una matriz de 8x8 pixels correspondiente a dígitos manuscritos de 0 a 9. La magnitud de cada atributo (pixel) sería la intensidad en una escala de grises que va desde cero (pixel en blanco), hasta un valor máximo que depende de la implementación (DESCR). El almacenamiento de cada ejemplo (dígito) es en un vector de 64 componentes proveniente del **aplanado** de la mencionada matriz de pixeles de 8x8.

Antes de seguir adelante, deberá separar un conjunto de muestras para aprendizaje y otro para test basándose en el método de reserva tomando el porcentaje habitual de  $\frac{2}{3}$  y  $\frac{1}{3}$ .

A continuación, se harán varios experimentos de clasificación aplicando el método de los k vecinos más próximos, que en python se lleva a cabo con:

- `sklearn.neighbors.KNeighborsClassifier`

De entre sus argumentos, aparte del número k, aparece otro denominado “algorithm”. Aquí hay que usar el denominado “brute”, ya que corresponde con el más simple descrito en la clase de teoría.

Para calcular la tasa de aciertos para un k dado, se efectuará la media aplicando el método de reserva (Hold-Out) repetido 10 veces.

### PARTE I:

Variando el k entre 1 y 15, ejecutar el algoritmo y obtener su tasa de aciertos para el conjunto de test. Representar este valor en un diagrama de barras, calcular el k óptimo y dar su correspondiente tasa de aciertos.

### PARTE II:

Lo mismo que lo anterior, sólo que los píxeles hay que reducirlos a bits, esto es, poner uno si el pixel no era cero y dejar el cero, si ya lo era.

### ENTREGA:

Un fichero jupyter-notebook/lab, con todo el código y las explicaciones pertinentes para seguir los pasos de esta práctica.