# Trabajo 3: Programación

Patricia Córdoba Hidalgo

# Índice



# 1. Problema de clasificación

Este problema consiste en clasificar imágenes de dígitos escritos a mano, asignándole a cada una el dígito que representan. Nuestro espacio de características, $\chi$, está formado por datos de 64 características, cada una de ellas representando la intensidad de trazado de una de las casillas de una matriz $8 \times 8$, donde se ha trazado el dígito. El conjunto de etiquetas, $Y$, son los dígitos del 0 al 9. La función desconocida $f$ es aquella que a cada imagen (elemento de $\chi$) le asigna el dígito que representa (su etiqueta correspondiente).

Primero leemos los ficheros de datos con la función `readData`, que separa los datos de sus etiquetas. Los datos del fichero test serán los datos que usaremos para calcular el $E_{out}$ y los del fichero training los dividimos en datos de entrenamiento y datos de validación. Un 25 % de los datos del fichero de datos training serán usados para validar.

Antes de empezar a trabajar con los datos, se preprocesan. Primero se usa PCA para quedarnos con 42 de 64 características, que son capaces de explicar el 99 % de la distribución. Tras esto, escalamos los datos, ya que, si una característica tiene una varianza varios órdenes mayor que otra, puede tener repercusiones en el cálculo de la función objetivo y puede hacer que se aprenda correctamente del resto de características.
Estas transformaciones se ajustan a los datos de entrenamiento (con la función `fit`) y luego se usan las mismas sobre el resto de datos (tanto al conjunto de validación como al de test).

Para clasificar los datos, usaremos regresión logística multietiqueta. Por el teorema de "No-Free-Lunch" todos los algoritmos son iguales en media, no hay un algoritmo mejor que otro, hay problemas donde unos tienen mejor desempeño que otros y elegí usar este algoritmo inicialmente porque ya lo hemos trabajado en otras prácticas, es sencillo y hemos visto como usarlo en caso de clasificación multietiqueta. Los resultados obtenidos fueron buenos, por lo que concluí que era un algoritmo con buen desempeño en éste problema.

La implementación del algoritmo de *Regresión Logistica* está en la función `rl_sgd`. La función `rl_sgd` devuelve una lista de 10 vectores, cada uno de ellos calculado en una de las iteraciones del bucle principal (usando gradiente descendente estocástico y el error visto en teoría), que corresponde a los pesos de la función que separa esa clase del resto.
Antes de ejecutar esta función, es necesario ajustar el formato de las etiquetas, pasando de tener dígitos a tener vectores de 10 coordenadas. A la etiqueta que representa el dígito $i$ se le asocia el vector $e_i$, aquel que tiene ceros en todas sus componentes menos en la posición $i$, que tiene un 1. La etiqueta 0 pasa a ser [1 0 0 0 0 0 0 0 0 0], la etiqueta 1 pasa a ser [0 1 0 0 0 0 0 0 0 0], y así sucesivamente, hasta la etiqueta 9 que pasa a ser [0 0 0 0 0 0 0 0 0 1].

Los resultados obtenidos son: ....