



Práctica Digit Recognition.

MNIST ("Modified National Institute of Standards and Technology") es el conjunto de datos de facto de "hola mundo" de la visión artificial. Desde su lanzamiento en 1999, este conjunto de datos clásico de imágenes escritas a mano ha servido como base para la evaluación comparativa de algoritmos de clasificación. A medida que surgen nuevas técnicas de aprendizaje automático, MNIST sigue siendo un recurso confiable tanto para investigadores como para estudiantes.

En esta competencia, su objetivo es identificar correctamente los dígitos de un conjunto de datos de decenas de miles de imágenes escritas a mano. Hemos seleccionado un conjunto de núcleos de estilo tutorial que cubren todo, desde la regresión hasta las redes neuronales. Lo alentamos a experimentar con diferentes algoritmos para aprender de primera mano qué funciona bien y cómo se comparan las técnicas.

Puedes ampliar información sobre la practica en <https://www.kaggle.com/>

Descripción del conjunto de datos.

Los archivos de datos train.csv y test.csv contienen imágenes en escala de grises de dígitos dibujados a mano, del cero al nueve.

Cada imagen tiene 28 píxeles de alto y 28 píxeles de ancho, lo que da un total de 784 píxeles. Cada píxel tiene un único valor de píxel asociado, que indica la claridad u oscuridad de ese píxel; los números más altos significan más oscuro. Este valor de píxel es un número entero entre 0 y 255, ambos incluidos.

El conjunto de datos de entrenamiento (train.csv) tiene 785 columnas. La primera columna, llamada "etiqueta", es el dígito dibujado por el usuario. El resto de las columnas contienen los valores de los píxeles de la imagen asociada.

Cada columna de píxeles del conjunto de entrenamiento tiene un nombre como pixelx, donde x es un número entero entre 0 y 783, ambos inclusive.



Práctica Digit Recognition.

Para ubicar este píxel en la imagen, supongamos que hemos descompuesto x como $x = i * 28 + j$, donde i y j son números enteros entre 0 y 27, ambos inclusive. Entonces, $\text{pixel}x$ se ubica en la fila i y la columna j de una matriz de 28×28 (indexada por cero).

Por ejemplo, $\text{pixel}31$ indica el píxel que está en la cuarta columna desde la izquierda y la segunda fila desde arriba, como en el diagrama ascii que se muestra a continuación.

Visualmente, si omitimos el prefijo "píxel", los píxeles forman la imagen de esta manera:

```
000 001 002 003 ... 026 027
028 029 030 031 ... 054 055
056 057 058 059 ... 082 083
|   |   |   |   ...   |   |
728 729 730 731 ... 754 755

756 757 758 759 ... 782 783
```

Objetivo de la Práctica.

Determinar el modelo con mayor Accuracy y menor consumo de recursos capaz de reconocer dígitos tal y como se define en los apartados anteriores. Para cumplir el primer objetivo podemos usar cualquiera de las técnicas (por ejemplo, tune) y parámetros (por ejemplo, parámetro de complejidad cp , número de neuronas $size$ o coste C) que se han estudiado. En cuanto al segundo objetivo, se puede usar cualquier técnica como *importance*, *PCA* o *encoders*. También, se puede usar técnicas que permitan disminuir el número de variables mediante la manipulación de la imagen..

Los modelos que se permiten usar son los que se han estudiado en la asignatura o combinación de ellos (ensemble).



Práctica Digit Recognition.

Entregables

1. Un informe que explique los pasos seguidos en la práctica tanto en la elección del modelo como en la elección de las técnicas para reducir la complejidad del problema. Planificación de las tareas y relación de tareas y componente del grupo que la realiza.
2. Todos los programas en R que se citen en la práctica.
3. Se adjuntará el modelo final entrenado con los 100 primeros dígitos del data frame (usa el comando `save` para guardar el modelo).

Los alumnos realizarán una presentación de su proyecto en el aula.