

## Tarea 4: Uso de bibliotecas para aprendizaje supervisado

### I Introducción

En esta tarea usted utilizará las bibliotecas Scikit-Learn y Keras para Python, y con ellas evaluará el desempeño de varios clasificadores usando datos reales de la base **MNIST**, que contiene alrededor de 70 000 imágenes de  $28 \times 28$  píxeles ilustrando variantes de los dígitos del 0 al 9 escritos a mano (ver figura 1).



**Figura 1:** Ejemplos de los dígitos en la base de datos **MNIST**.

#### I.1 Scikit Learn y SVM

1. Instale la biblioteca Scikit-Learn con todas sus dependencias en Python. Para ello puede utilizar `pip`, Anaconda, o cualquier otro manejador de paquetes de su elección.
2. Busque en el Internet algún ejemplo sencillo de uso de las máquinas de soporte vectorial (SVM) con Scikit-Learn para clasificar los dígitos escritos a mano de la base **MNIST**.
3. Repita el problema de clasificación de los datos de prueba utilizando:
  - 3.1. al menos 3 tipos de kernel (uno de los cuales debe ser de base radial)
  - 3.2. para cada kernel, variando el valor de los parámetros que tenga cada uno (por ejemplo,  $\gamma$  en el kernel de base radial)
  - 3.3. para cada configuración probada, variando el valor  $C$

Nótese que son bastantes pruebas que deben realizarse, cada una involucrando un proceso de entrenamiento, y un cálculo de matriz de confusión con datos de validación, por lo que se recomienda automatizar la ejecución de dichas pruebas. Sin embargo, deberá hacer varios experimentos manuales, para determinar rangos útiles de los parámetros a probar.

Si su acceso a computador es limitado, puede utilizar Google-Colab para correr sus experimentos, donde debe tomar en cuenta que los periodos disponibles no pueden exceder 12 h, es decir, su programa debe permitir retomar a partir de algún punto que usted le indique.

4. Calcule para cada caso la matriz de confusión y los valores derivados de esta.  
Observe que la base de datos **MNIST** ya trae datos para entrenamiento y datos para prueba. Asegúrese de usarlos correctamente.
5. Para presentar los resultados, utilice curvas de precisión contra exhaustividad, que se trazan variando un parámetro, dejando todos los otros constantes (puede buscar más información sobre curvas ROC (*receiver operator characteristic*), que es una técnica de análisis similar, que se diferencia por usar otras métricas de la matriz de confusión en los ejes).
6. Asegúrese de serializar la SVM entrenada, para que se pueda utilizar en la etapa de reconocimiento y evaluación.
7. Agregue un programa que le permita clasificar alguna imagen arbitraria de un dígito. Usted puede introducir el nombre de un archivo, o puede diseñar una interfaz sencilla en Python que permita al usuario dibujar el dígito a ser reconocido, y mostrar el resultado de la predicción.
8. Analice brevemente sus resultados.

## I.2 Keras y Deep Learning

1. Instale la biblioteca Keras con todas sus dependencias en su versión de Python. Para ello puede utilizar `pip`, Anaconda, o cualquier otro manejador de paquetes moderno.
2. Busque en el Internet algún ejemplo sencillo de red neuronal (profunda) para clasificar la base de datos **MNIST** (dígitos escritos a mano).
3. Calcule la matriz de confusión para un conjunto de datos de prueba.  
Observe que la base de datos **MNIST** ya trae datos para entrenamiento y datos para prueba. Asegúrese de usarlos correctamente.
4. Modifique el programa que le permite clasificar alguna imagen arbitraria de un dígito para este caso también (el programa debe permitir elegir el clasificador a utilizar)
5. Utilice el método *evaluate* del modelo para mostrar diversas estadísticas.
6. Analice los resultados y compárelos con los SVM

## II Entregables

1. Archivos de Python.
2. Modelos entrenados
3. Archivo README con instrucciones de cómo ejecutar su código.
4. Archivo PDF con las gráficas generadas y el análisis de resultados.

### III Notas

- Esta tarea se debe resolver en parejas.
- Es una tarea sencilla, pero interesa que se estudie a profundidad el código que se encuentre, y que se tome el tiempo para diseñar el experimento que prueba distintas configuraciones y parámetros.
- Los marcos de trabajo instalan la base de datos **MNIST** en sus ejemplos, por lo que posiblemente usted no requiera instalarlas aparte.
- Puesto que el entrenamiento tarda bastante y no siempre las versiones que tenga el profesor coinciden con las utilizadas, todos los resultados deben entregarse en el archivo pdf (la curvas y algunas matrices de confusión que considere relevante mostrar, para distintos parámetros de las SVM).