Trabajo Práctico 2

Laboratorio de Datos
Facultad de Ciencias Exactas y Naturales - Universidad de Buenos Aires

Segundo Cuatrimestre 2024

El objetivo del presente trabajo práctico es aplicar algunas herramientas de lo aprendido sobre clasificación y selección de modelos con validación cruzada.

El trabajo práctico se realiza de a 3 integrantes, sin excepción. Se asume que los grupos serán los mismos que de TP01. Si por alguna razón necesitan cambiar la composición del grupo, por favor avisen a la cátedra.

1. Introducción

Dataset. En el presente TP trabajaremos con el conjunto de datos de imágenes (de 28x28 píxeles en escala de grises con valores 0-255) denominado `MNIST-C` en su versión "Motion Blur". Cada imagen del *set* de datos representa un dígito escrito a mano entre 0 y 9, ambos inclusive. Se trata de una versión ligeramente corrompida de un dataset muy famoso llamado `MNIST`[1]. En la sección Referencias hay links al dataset `MNIST-C` completo y el trabajo donde lo presentan. Ahí pueden ver el tipo de alteración que elegimos y todas las que hay disponibles.

Clasificación por atributos. Este dataset está compuesto por imágenes, lo que plantea una diferencia frente a los datos que utilizamos en las clases. En `titanic`, por ejemplo, cada elemento del dataset estaba definido por atributos con una interpretación muy concreta (sexo, edad, etc). En este trabajo, en cambio, los atributos son el valor de cada píxel en la imagen ($28 \times 28 = 784$ atributos). Tengan en cuenta esto al realizar la exploración de los datos y utilicen gráficos para ayudar a la visualización!

Archivos. Para comenzar deben descargar del campus de la materia el conjunto de datos que se encuentra en formato `npy` (archivo binario para guardar arrays de `numpy`). Son dos archivos: `mnistc_images.npy` contiene las imágenes y `mnistc_labels.npy` contiene el dígito que corresponde a cada una. Encontrarán también un código mínimo para arrancar.

2. Ejercicios

2.1. Análisis exploratorio

Deben analizar la cantidad de datos, cantidad y tipos de atributos, cantidad de clases de la variable de interés (dígitos) y otras características que consideren relevantes. Además se espera que con su análisis puedan responder las siguientes preguntas:

I. ¿Cuáles parecen ser los atributos (i.e., píxeles) más relevantes para predecir el dígito al que corresponde la imagen? ¿Cuáles no? ¿Creen que se pueden descartar atributos?

II. ¿Hay dígitos que son parecidos entre sí? Por ejemplo, ¿qué es más fácil de diferenciar: las imágenes correspondientes a los dígitos 0 y 1, ó las imágenes de 5 y 6?

III. Tomen una de las clases, por ejemplo el dígito 7. ¿Son todas las imágenes muy similares entre sí?

Importante: las respuestas deben ser justificadas en base a gráficos de distinto tipo.

[1] `https://en.wikipedia.org/wiki/MNIST_database`

2.2. Clasificación multiclase

Dada una imagen se desea responder la siguiente pregunta: ¿a cuál de los dígitos corresponde la imagen?

I. Deben elegir cinco dígitos para trabajar correspondientes al número del grupo (ver tabla). Primero seleccionar solo los datos correspondientes a esos dígitos. Luego separar el conjunto de datos en desarrollo (*dev*) y validación (*hold-out*). Para los incisos II. y III. utilizar el conjunto de datos de desarrollo solamente. Dejar apartado el conjunto *hold-out* para utilizar recién en el inciso IV..

Grupo	Dígitos
1	0, 1, 5, 6, 9
2	1, 3, 6, 7, 8
3	0, 1, 2, 3, 9
4	3, 4, 6, 7, 9
5	0, 1, 2, 4, 6
6	0, 1, 2, 8, 9
7	1, 2, 3, 7, 8
8	2, 4, 6, 7, 9
9	1, 4, 5, 8, 9
10	1, 3, 4, 5, 6
11	3, 4, 6, 8, 9
12	1, 2, 3, 7, 9
13	1, 2, 3, 4, 9
14	0, 2, 4, 6, 7

II. Ajustar un modelo de árbol de decisión. Explorar y reportar el rendimiento con distintas profundidades máximas.

III. Realizar un experimento para comparar y seleccionar distintos árboles de decisión con distintos hiperparámetos. Para esto, utilizar validación cruzada con k-folding. ¿Cuál fue el mejor modelo? Documentar cuál configuración de hiperparámetros es la mejor y qué performance tiene.

IV. Entrenar el modelo elegido a partir del inciso previo, ahora en todo el conjunto de desarrollo. Utilizarlo para predecir las clases en el conjunto *hold-out* y reportar la performance.

Observación: Al realizar la evaluación utilizar métricas de clasificación multiclase como, por ejemplo, la exactitud. Además pueden realizar una matriz de confusión y evaluar los distintos tipos de errores para las clases.

3. Entrega

Importante: ¡No deben entregar los archivos del dataset!
La entrega comprende los siguientes CUATRO archivos:

I. Un archivo llamado `numero_grupo_analisis.py` con el código principal. Este archivo puede complementarse con otros archivos `.py` donde figure parte del código, y que sean importados y utilizados desde el archivo principal.

Como siempre, ordenar el código de la siguiente manera:

- Al inicio, una descripción que contemple: el nombre del grupo, los nombres de lxs participantes, contenido del archivo y cualquier otro dato relevante que considere importante.
- Luego la sección de los `import`.
- A continuación, la carga de datos.
- Siguiendo, las funciones propias que hayan definido.

- Finalmente, el código que no está dentro de funciones.

El código debe estar modularizado (separando bloques con `#%%`) para permitir su ejecución por fragmentos.

Todo lo que figure en el informe debe deducirse de los resultados del código.

II. Un archivo `README.txt` con los requerimientos de bibliotecas utilizadas e instrucciones sobre cómo ejecutar el código.

III. Un archivo llamado `numero_grupo_informe_tp2.pdf` que tenga un informe breve (no más de 10 carillas, de las cuales la mayoría deberían ser gráficos) en LaTeXcompilado (pdf).

Ordenar el informe de la siguiente manera:

- Carátula con título, nombres de lxs integrantes con sus correos electrónicos, número de grupo, materia, cuatrimestre, etc.
- Breve introducción al problema donde digan qué se propusieron hacer, qué base de datos van a utilizar, etc.
- Resultados del análisis exploratorio realizado, incluyendo los gráficos que lo muestran. Importante: **redacten** lo que se supone que quieren mostrar con dichos gráficos. Ayuden quien lea el informe a interpretar lo que ustedes están incluyendo/viendo/presentando.
- Explicación sobre los experimentos realizados, incluyendo los gráficos que consideren convenientes. Acá pueden incluir algo de detalle metodológico, como por ejemplo si utilizaron validación cruzada o no, qué porcentaje de los datos separaron para desarrollo (entrenamiento/validación) y testing (*hold-out*), qué biblioteca/función están utilizando, etc.
- Conclusiones, incluyendo los resultados relevantes de los modelos desarrollados.

IV. La planilla de autoevaluación que se explica a continuación.

4. Autoevaluación

Aquí tienen una herramienta para chequear la calidad del análisis realizado y el informe que prepararon. Al finalizar la entrega y antes de enviar el TP-02, realizar lo siguiente:

I. Copiar la siguiente planilla de autoevaluación (una sola a nivel grupal) a una carpeta personal:
`https://docs.google.com/spreadsheets/d/1XVvxSp0oU43YgyczajcwWKOZJbOK0gU_i9wSzTAkQwM`

II. Completarla.

III. Descargarla como pdf y agregarla al envío virtual, con nombre `numero_grupo_autoeval.pdf`.

Tomen esto como una herramienta: si ven que pueden mejorar el puntaje en algún ítem de la autoevaluación, obviamente pueden rehacer o mejorar la parte correspondiente antes de entregar.

5. Referencias

Dataset completo: `https://zenodo.org/records/3239543`
Trabajo original donde se presenta el dataset (preprint): `https://arxiv.org/abs/1906.02337`