TAREA 1: ANÁLISIS DE DATOS PRÁCTICO UTILIZANDO PYTHON (E/E)

A. Ferreira, W. Gomez, E. Castro
Departamento de Ingeniería Matemática
Universidad de La Frontera

03 de abril de 2023

Para esta Tarea, a cada estudiante se le asignarán 3 algoritmos de clasificación (de los vistos en clases), seleccionados aleatoriamente y utilizando la semilla de asignación. Para esto, a continuación se presenta un programa que tomará la semilla asignada y entregará los 3 algoritmos con los cuales cada estudiante deberá abordar esta tarea.

```
import random

mi_semilla_asignada = 13
random.seed(mi_semilla_asignada)
algoritmos = ['Perceptron','KNN','Regresion Logistica','SVM Kernel lineal',
              'Random Forest','SVM Kernel no lineal','Arbol de Decision']
asignados = random.sample(algoritmos,k=3)
print("Mis algoritmos asignados son: \t",asignados)
```

Por otro lado, la Base de Datos $desafio1.csv$ contiene 17 variables (columnas). De estas, sólo la última variable es Categórica (representa etiquetas de clase). El resto de las variables son continuas. A partir de este conjunto de datos:

Pregunta 1 (2 pts)

Preparar los datos para ajustar modelos de clasificación. Para esto:

1. Leer la base de datos y almacenarla como un objeto *DataFrame* de Pandas. Mostrar estadísticas descriptivas de los datos y generar un gráfico de tortas que muestre el porcentaje del total de datos que pertenece a cada clase y la cantidad de datos en las clases.

2. Generar los conjuntos (X, y) y realizar un muestreo aleatorio simple y estratificado, tomando el 30% de los datos para construir el conjunto de test. Con el 70% de datos restantes, construir el conjunto de entrenamiento. Mostrar en gráficos de tortas las distribuciones de los datos en las clases de los conjuntos de test y de entrenamiento.

Pregunta 2 (3 pts)

Utilizar los tres algoritmos de clasificación asignados y responder a los siguientes items.

1. Con el conjunto de entrenamiento generado en la pregunta anterior, obtener tres modelos de clasificación usando los algoritmos asignados. Describir los hiperparámetros utilizados. Recuerde estandarizar los datos en el caso que sea necesario.
2. Utilizar los modelos obtenidos en el item anterior para predecir la clasificación de los datos incluidos en el conjunto de test. Generar un nuevo DataFrame con los datos del conjunto test, incluyendo la clasificación real y la clasificación predicha por los modelos.
3. Finalmente, mostrar el total de datos bien clasificados y el total de datos mal clasificados por los modelos y mostrar los gráficos de tortas con las distribuciones predichas por estos, utilizando en este item solo los datos del conjunto test.

Pregunta 3 (1 pts)

Utilice la semilla asignada y el código del bloque siguiente para extraer dos características de la base de datos y generar una nueva base de datos (X_2, y).

```
import os
import pandas as pd

mi_semilla_asignada = 13
df = pd.read_csv(os.path.join(os.getcwd(), 'desafio1.csv'))
auxY, auxX = df['class'], df.drop('class', axis= 'columns')
auxX = auxX.sample(n= 2, axis= 'columns', random_state= mi_semilla_asignada)
auxX.columns = [i for i in range(auxX.shape[1])]
X = pd.concat([auxX[auxY == 0][:100], auxX[auxY != 0][:100]])
X.reset_index(inplace= True, drop= True)
y = pd.Series([0] * 100 + [1] * 100)
X2 = X.sample(frac= 1, random_state= mi_semilla_asignada)
y = y.sample(frac= 1, random_state= mi_semilla_asignada)
```

Utilizar el algoritmo *AdalineSGD* implementado en clases para ajustar un modelo en el conjunto de datos (X_2, y). Mostrar los umbrales de decisión para la primera época de entrenamiento, alguna época intermedia y el umbral de decisión del modelo final. Adicionalmente, mostrar el comportamiento de la pérdida en función de las épocas de entrenamiento. ¿Qué puede concluir?

Cada estudiante debe desarrollar la tarea de forma individual y preparar un Notebook denominado *Nombre_Apellido_Tarea*1.*ipynb*. El Notebook debe incluir la explicación de cada paso implementado, los códigos correspondientes y los resultados obtenidos. En caso de haber copia, se sancionará a los estudiantes implicados con la nota mínima. Cada estudiante deberá subir su solución al campus virtual antes del día martes 11 de abril a las 23:30 hrs.