Neural and Evolutionary Computation (NEC)

A2: Unsupervised learning with PCA, t-SNE, k-means, AHC and SOM

Objective

Apply and compare the results from five different unsupervised learning techniques:

- **Principal Component Analysis** (**PCA**)
- **t-distributed Neighbor Stochastic Embedding** (**t-SNE**)
- **k-means**
- **Agglomerative Hierarchical Clustering** (**AHC**)
- **Self-Organizing Maps** (**SOM**)

Deliverables

This assignment can be done alone or in pairs (groups of two)

For this activity you must deliver **one PDF document** that includes:

- A link to the Github repository where the code of all the activity is accessible. More details on the code in the following sections.
- Explanations of the different questions that are detailed in the section below, including all the relevant implementation decisions taken during the process.
- The name of the file should be **A3-Name1Surname1-Name2Surname2.pdf**

Part 1: Selecting and analyzing the datasets

The unsupervised learning techniques must be applied on two datasets:

1. Synthetic dataset (*A3-data.txt*):
    - Features**:** 4 variables, 1 class
    - Patterns**:** 360 patterns
    - The class information must "not" be used in the unsupervised learning, only to identify the classes in the plots

2. Search a dataset from the Internet, with the following characteristics:
    - Features: at least 6 variables, and a class attribute
    - The class attribute must refer to, at least, 4 different classes
    - Patterns: at least 200 patterns
    - The class information must "not" be used in the unsupervised learning, only to identify the classes in the plots

As an output of this part, **you should include in your report the following analysis**:

- For dataset 2, describe the details of the dataset and **the link to the source webpage where it has been download.** If you have done any type of data preprocessing also include this information in the dataset description.

Part 2: Comparing unsupervised learning algorithms

We are going to perform unsupervised learning of the two datasets using some of the algorithms that we have seen in class. In this assignment you do not need to implement the algorithms, just use already programmed algorithms in open-source libraries. All the code (and notebooks) used in the analysis should be included in the github repository.

- **PCA**: find and plot the PCA projection in two dimensions, using a different color for each class. For each PCA analysis you should include **2 plots**: **a colored scatter plot of the first two principal components,** and a **scree plot with the accumulated variance.**

- **t-SNE**: find and plot the t-SNE projection in two dimensions, using a different color for each class. You can play with different parameters of the t-SNE (perplexity, …). For each set of parameters you should include a colored scatter plot visualization with the description of the used parameters.

- **k-means**: use k-means to classify the patterns in $k = 2, 3, \ldots, K$ classes, and compare the obtained classes with the real ones. For each value, include a scatter plot of the data (e.g., using as x and y coordinates the PCA reduction), and color the points according to the classes they belong to. If the value of k is equal to the real number of classes, you can use a confusion matrix to compare the results obtained.

- **AHC**: use the unweighted average (UPGMA) and complete linkage (CL) methods of Agglomerative Hierarchical Clustering (AHC), using as input the matrix of Euclidean distances between the original patterns, and use different colors to represent the patterns in each original class. **Plot the resulting dendrograms**.

- **SOM**: use Self-Organizing Maps (SOM) to visualize the data, using different settings (topology and size of the map, learning rate, neighborhood function, etc.). The minimum size for the SOM architecture must have at least 100 neurons. Examples of visualizations of SOM are a heatmap of the most represented class in each position, or the u-matrix. For the "best" map, also calculate and plot the component planes.