Module 6: Descriptive and Predictive Modeling

Exercise 1: K-Nearest Neighbors in depth

Preliminary note: this exercise is to be completed in groups of 2 students.

Based on the Python code examples used in class, groups are asked to choose one of the following data sets and describe what their fields represent:

A) "from sklearn.datasets import load_breast_cancer" (basic)

B) "from sklearn.datasets import load_digits" (advanced)

Departing from this selection, groups must address the following questions in order:

1) Describe the dataset in dimensions such as number of features, number of categories, and number of samples per category using Python.

2) Represent the statistical support of every feature graphically, resorting to Matplotlib's boxplot function. Are there any outliers that can be detected by simple visual inspection? If so, devise a handcrafted method to detect and isolate such examples.

3) Repeat each of the experiments seen in class with the K-Nearest Neighbors model, providing arguments for each of the steps taken along the process, and commenting on the partial results obtained with the selected dataset. Please use as many performance metrics as needed to illustrate the particularities of the selected dataset (e.g. imbalanced classes).

4) Read the Scikit-learn library documentation and configure the automated validation script so that the GridSearchCV() function uses leave-one-out cross-validation instead of k-fold. Which conclusions can be drawn from the mean cross-validation scores and the test scores using a k-neighbor model with optimized k?

5) Elaborate on the need for stratifying the cross-validation process analyzing the distribution of samples by class. If so, please show with empirical evidence what could occur if such a stratification was not performed, specially when decreasing the number of samples of the dataset.

6) Include in the set of hyper-parameters adjusted via cross-validation process the weights of the distance metric between samples according to the "weights" parameter of the model in Scikit-learn. Compute the model's performance when distance metric weights are fine-tuned within cross-validation with respect to only tuning the number of neighbors (K).

7) Following the same approach as in the last section, enter the type of distance metric ("metric" parameter) within the cross-validation process. Evaluates the results and gains / losses of generalizability of the model.

1) (ADVANCED) Repeat questions 1 to 7 with a dataset of your choice from the UCI repository, Choose a classification dataset with a "moderate" number of samples not to overload the computing resources)

Reports can be a DOC document, a PDF document (along with the Python scripts that generate the reported figures and results) or a Jupyter Notebook (with saved checkpoint). Other formats (e.g. link to Google Colab) must be agreed with the professor. When uploading the report, please indicate name, surname and ID (DNI number) of all members of the team.

Delivery deadline: March 7th, 2021