

## TP 2

### Machine Learning

**Note :** Prepare a report (including the code source and the results) for the TP and send it to [nistor@lipn.univ-paris13.fr](mailto:nistor@lipn.univ-paris13.fr) . Answer to all questions.

#### Part I. K-Nearest Neighbor Algorithm

The nearest neighbor algorithm is a very simple classification algorithm based on the following principle: the class of each test data (to be classified) must be the closest (most similar) data class among the learning data.

List of useful functions:

*metrics.pairwise.euclidean\_distances*: calculate distances between data.

*argsort, argmin, argmax*: returns the indices of the ordered, minimum and maximum values.

*neighbors.KNeighborsClassifier*: the algorithm of the K Nearest Neighbors of sklearn.

1. Create a KNN (X, Y) function that takes X data and Y labels and returns a label, for each data, from the nearest neighbor of that data. Here we take each test data and we consider all others as learning. This allows us to test the power of the prediction algorithm according to the cross-validation method i.e. "leave-one-out".
2. The KNN function calculates the predicted class for each data item. Change the function to calculate and return the prediction error.
3. Test on Iris data.
4. Test the function of the k nearest neighbors of sklearn (with here  $K = 1$ ). Are the results different? Test with other values of K.
5. Change the KNN function so that it takes as input a number K of neighbors (instead of 1). The predicted class will then be the majority class among the K's neighbors. Test the function with a different numbers of K. What you can conclude?

#### Part II. Other classifications algorithms

1. Split the Iris dataset into two parts: learning subset (70%) and prediction subset (30%).
2. Use the scikit-learn library and apply the Support Vector Machine, Naive Bayesian Classifier and the Decision Trees on the iris dataset. Compute the prediction error for each method. Compute the confusion matrix and analyze the results by explaining the confusion matrix.