Computer Science 6915 - Winter 2019
Assignment 1

Implement from scratch in Python3 a KNN classifier. "Implementing from scratch" means that you are not allowed to use functions such as those provided by the sklearn.neighbors module to perform KNN, instead you need to write the necessary code yourself. Your program should be called KNN.py and it should run in Linux. You program should take three command-line arguments (given in the following order):

1. A filename specifying a tab-delimited plain-text file containing the training data. The file has a header with the name of the attributes and the last column indicates the output for that instance. Your program should be able to take a variable number of attributes. [You can assume the file is in the working directory.]
2. A filename specifying a tab-delimited plain-text file containing the unseen instances. The file has a header with the name of the attributes and it does not have a column corresponding to the output for that instance. The number of attributes in this file is the same as the number of attributes in the training data. [You can assume the file is in the working directory.]
3. An integer specifying the value of K. If not value is specified, the default value is 3.

For example, you program might be executed as follows:

$python3 KNN.py train.tsv test.tsv 5

where the $ indicates the terminal prompt. For each of the unseen instances, your program should print in the standard output (i.e., the terminal), the predicted class and estimated conditional probability of the predicted class (one prediction per line, the class and its conditional probability should be separated by a tab). The first two lines of a sample output look like this:

7 0.60
1 0.75

Note this is not the actual output you should get.

Data sets to be used are provided in D2L as tab-delimited text files. The task is to predict the type of a glass based on its chemical analysis. The training data (TrainingData_A1.tsv) consists of 204 instances, 9 attributes and 6 classes. You need to predict the class of the ten instances in the file TestData_A1.tsv.

For this assignment, submit through D2L the following (one submission per team):

a) Your python code in a single file called KNN.py
b) A one-page description of your implementation including the pseudo-code of your KNN implementation, the definition of the distance function you used, a description of any improvements you made to the standard KNN algorithm (if any), and explaining how you tested your code to make sure your implementation was correct. This description has to be submitted as a PDF file.

The assignment will be graded based on the correctness of the KNN implementation (including classification performance, code correctness and meeting specifications), and clarity of the description.

If you need, you can use these computer labs: https://www.mun.ca/computerscience/ugrad/labschedule.php