

# Week 4 - k Nearest Neighbours

MACHINE INTELLIGENCE LABORATORY- PESU UE19CS305



Teaching Assistants-

[vighneshkamath43@gmail.com](mailto:vighneshkamath43@gmail.com)

[sarasharish2000@gmail.com](mailto:sarasharish2000@gmail.com)

[abaksy@gmail.com](mailto:abaksy@gmail.com)

k-Nearest Neighbours is one of the most famous supervised classification and regression algorithms. The beauty of the algorithm is that we can observe it in our real life, i.e., our nature is to be close with people who are the most like us.

In this assignment you are required to prepare a Python class KNN which can be used by anyone for **classification**. The detailed instructions are given to you in this document.

**Your task is to complete the code for the class KNN and its methods**

**You are provided with the following files:**

1. **week4.py**
2. **SampleTest.py**

Note: These sample test cases are just for your reference.

## SAMPLE TEST CASE Data

The training data is given in the form below (N=10, D=2)

	x1	x2	y
0	2.781084	2.550537	0.0
1	1.465489	2.362125	0.0
2	3.396562	4.400294	0.0
3	1.388070	1.850220	0.0
4	3.064072	3.005306	0.0
5	7.627531	2.759262	1.0
6	5.332441	2.088627	1.0
7	6.922597	1.771064	1.0
8	8.675419	-0.242069	1.0
9	7.673756	3.508563	1.0

The dataset here consists of N points each of D dimensions and the class that the data points belong to (x1 and x2 represent the dimensions, and y represents the class/category that the point belongs to). All dimensions are **numerical** and **continuous** in nature; hence you do not need to encode the data in any way.

Note that there may be more than 2 classes. Do not assume that number of classes will be 2 always!

## Important Points:

1. Please do not make changes to the function definitions that are provided to you. Use the skeleton as it has been given. Also do not make changes to the sample test file provided to you. Run it as it is.
2. You are free to write any helper functions that can be called in any of these predefined functions given to you. Helper functions must be only in the file named 'YOUR\_SRN.py'.
3. **Note that the code is for classification only.**
4. **Your code will be auto evaluated by our testing script and our dataset and test cases will not be revealed. Please ensure you take care of all edge cases!**
5. **The experiment is subject to zero tolerance for plagiarism. Your code will be tested for plagiarism against every code of all the sections and if found plagiarized both the receiver and provider will get zero marks without any scope for explanation.**
6. **Kindly do not change variable names or use any other techniques to escape from plagiarism, as the plagiarism checker is able to catch such plagiarism**
7. Hidden test cases will not be revealed post evaluation.

- $$W_{x_i} = \frac{1}{d(x_i, q)}$$

- ✓ You are provided with structure of class KNN.
- ✓ The class KNN contains one constructor and five methods.
- ✓ Your task is to write code for the four of these five methods.

1. You may write your own helper functions if needed
2. You can import libraries that come built-in with python 3.7
3. You cannot change the skeleton of the code
4. Note that the target value is an int

1. This will help you check your code.
2. **Passing the cases in this does not ensure full marks, you will need to take care of edge cases**
3. Name your code file as YOUR\_SRN.py
4. Run the command  
**python3 SampleTest.py --SRN YOUR SRN**

**if import error occurs due to any libraries that is mentioned in the skeleton code try:**  
**python3.7 SampleTest.py --SRN YOUR\_SRN**

**SUBMISSION FORMAT and DEADLINE:**

You are required to complete code in week4.py and **rename it to SRN.py**

**Failing to submit in the right format will lead to zero marks without a chance for correction**

Example: If your SRN is PES1201801819 your file should be PES1201801819.py

- Delete all print statements if used for debugging before submission
- Ensure you run your file against sample test before submitting
- Check for syntax and indentation errors in your file, and make sure that tabs and spaces are used uniformly for indentation (i.e., if tabs are used then only tabs must be used throughout the file, and similarly for spaces)
- All the helper functions should be in the same file (SRN.py), make sure you run the sample test cases as indicated above before submitting.

**Deadline: on or before 19/09/2021 10:00PM**

**Where to submit: Edmodo**