Lab Topic 07 - Classification using k-Nearest Neighbors

CNM Peralta

Background

The **k-Nearest Neighbors classification algorithm** is a **non-parametric machine learning algorithm** that classifies new data based by finding its k nearest neighbors in the data set, and choosing the classification that is represented the most.

The Algorithm

Given:

$$\begin{array}{cccccccc}
x_{00} & x_{01} & x_{02} & \cdots & x_{0n} & \rightarrow & y_0 \\
x_{10} & x_{11} & x_{12} & \cdots & x_{1n} & \rightarrow & y_1 \\
x_{20} & x_{21} & x_{22} & \cdots & x_{2n} & \rightarrow & y_2 \\
 & & & \vdots & & & \\
x_{m0} & x_{m1} & x_{m2} & \cdots & x_{mn} & \rightarrow & y_m
\end{array}$$

The classification of a new feature vector $x = (x_0, x_1, x_2, ..., x_n)$ is determined by computing its distance from all the other feature vectors in the training data set, and choosing the *k* nearest neighbors, where *k* is a given value. Thus we have the following steps:

- Given *k* and a feature vector, *x*:
    - For each feature vector, *v*, in the training data set:
        - Compute the distance from *x* to *v* using Euclidean distance.

$$d = \sqrt{\sum_{i}^{n}(x_i - v_i)^{2}}$$

        - If it is one of the k nearest neighbors so far, remember it.
    - Go through the k nearest neighbors and count the number of times each class occurs.
    - The class with the maximum count will be the classification of x.

Exercise

Create a program that classifies feature vectors using k-Nearest Neighbors.

A dataset containing diabetes information will be used. The task is to classify the next points from diabetes.csv. The test file contains information regarding the number of pregnancies, glucose value, blood pressure, skin thickness, insulin value, bmi, diabetes pedigree function, age, and outcome of a person. The person can be classified as either diabetic (outcome=1) or non-diabetic (outcome=0). The program must read input.in which contains unlabelled data points.

Write the output of your program to a text file (**output.txt**) with the following format:

```
3.0, 70.0, 34.0, 31.0, 77.0, 31.0, 0.59, 24.0, 0
4.0, 135.0, 91.0, 0.0, 33.0, 34.6, 0.190, 25.0, 0
8.0, 140.0, 80.0, 0.0, 0.0, 23.1, 1.85, 49.0, 0
4.0, 174.0, 70.0, 55.0, 135.0, 52.2, 0.75, 22.0, 1
1.0, 100.0, 63.0, 33.0, 117.0, 28.5, 0.5, 30.0, 0
2.0, 120.0, 65.0, 20.0, 119.0, 32.9, 0.876, 25.0, 0
0.0, 189.0, 62.0, 19.0, 199.0, 35.1, 0.7, 24.0, 1
```

***The last column is the classification. In this dataset, 1:Diabetic or 0:Non-diabetic.*
Rows and columns in training data may vary. Your code should be able to handle other datasets.

NOTE: Newly-classified input data should be considered for the classification of the rest of the input data.

Scoring

- The criteria for the exercise is as follows:

Criteria	**Points**
Read training dataset correctly	2
Read input.in correctly	2
Compute distances correctly	2
Identify k-Nearest Neighbors correctly	2
Write output.txt correctly	2
Total	10

Reference

Stuart Russell and Peter Norvig. 2009. *Artificial Intelligence: A Modern Approach* (3rd ed.). Prentice Hall Press, Upper Saddle River, NJ, USA.