Assignment 4

CMSC 478 – Machine Learning

October 9, 2024

Item	Summary
Assigned	Oct 9,
Due	Oct 22, 11:59 PM
Topic	KMeans, KNN
Points	100

You are to complete this assignment on your own: that is, the code and writeup you submit must be entirely your own. However, you may discuss the assignment at a high level with other students or on the discussion board. Note at the top of your assignment who you discussed this with or what resources you used (beyond course staff, any course materials). For the programming part of this assignment, code must be written in python. You may use google colab / ,ipynb notebook.

1. KNN (50 points) (Programming part):
You will design a k-nearest neighbor classifier for this homework assignment and use it to classify some data. Attached are two datasets. One is for training (train.txt) and the other is for testing (test.txt). Each row in the data is one observation. All observations belong to one of the two classes (class 0 and class 1). The first two columns are 2 dimensional observations/features of the data and the third column contains class labels. Hint: use "numpy.loadtxt()" to load the data.

Task 1): It's always a good idea to visualize the data whenever you can, so the first task involves looking at the data. Plot the two datasets (using the scatter plot function in matplotlib). Show class 0 in red and class 1 in blue. Make sure to properly label your plots.

Task 2): Design a k-nearest neighbor classifier to classify the test dataset. (Do not use a built-in knn library). Using your classifier, try out 5 different values for k (k = 7, k = 51, k= 151, k = 201 and k = 251) and report classification accuracies on the test dataset.

You should notice the accuracy scores are generally decreasing as k increases. **Explain why that is happening in a sentence or two.**

2. K-Means (50 points) (Written part):

Suppose you have a dataset in which the instances are 1-dimensional. The instances are 1, 2, 4, 5, 10, 11, 12, 25. Run k-means clustering on this dataset for k = 3 with initial centroids on the first 3 instances in the dataset (i.e., 1, 2, 4).

Draw the points on a number line and show which points belong to which clusters for each iteration of k-means. Either point out the clusters or use the Ci = j notation. Run the algorithm until two consecutive iterations yield the same assignment of points to clusters.

3. Bonus question (10 points): Draw the elbow curve with inertia as the evaluation criteria. Mention the optimal number of clusters for the above problem. (For this bonus question, you may solve it either using code or by hand).