Machine Learning Assignment 1
Report

By : M.Sri Sailesh(S20170010093),Tanay Rathore (S20170010163)

KNN-Classifier:

KNN-Classifier is an algorithm used to classify given data based on the attributes of other data points present in the dataset. More Specifically, we use the Minkowski Distance, as a metric to find out who the k nearest neighbors are, and subsequently assign the most common class among the k neighbors to the current datapoint in question.

KNN is a lazy learning algorithm. All of the computation is only performed when classification needs to be done.

Cross Validation(r-Fold):

Cross-validation is primarily used in machine learning to estimate the skill of a machine learning model on unseen data. That is, to use a limited sample in order to estimate how the model is expected to perform in general when used to make predictions on data not used during the training of the model.

The general procedure is as follows:

- Shuffle the dataset randomly.
- Split the dataset into k groups
- For each unique group:
    - Take the group as a hold out or test data set
    - Take the remaining groups as a training data set
    - Evaluate the test data after training with the training set

Dataset-1: Wheat Seeds Dataset.

No of examples: 210

No of Attributes: 7

No of Classes: 3

Metadata stored in seeds_meta.h.

Code: KNN_seeds.c

Compilation & running instructions:

$ gcc KNN_seeds.c -lm

$ ./a.out

Results:

Program was run for 5 times.Due to limited size of the dataset(210),and random shuffling of data, Much variation was observed between runs, Nevertheless the accuracy values were between 89-92%.

- Optimum Values: K:1 P:2 Accuracy:0.900000
- Optimum Values: K:1 P:2 Accuracy:0.914286
- Optimum Values: K:10 P:3 Accuracy:0.919048
- Optimum Values: K:7 P:1 Accuracy:0.914286
- Optimum Values: K:5 P:1 Accuracy:0.909524

Dataset-2: Optical Recognition of Handwritten numbers.

No of examples: 3823

No of Attributes: 64

No of Classes: 10

Metadata stored in ocr_meta.h.

Code: KNN_seeds.c

Compilation & running instructions:

$ gcc KNN_ocr.c -lm

$ ./a.out

Results:

Program was run for 3 times. In these 3 trials, K and P were 3,3 always. However, we can expect slight variations because of the random shuffling. Accuracy values are around 97-98%

- Optimum Values: K:3 P:3 Accuracy:0.986660
- Optimum Values: K:3 P:3 Accuracy:0.987444
- Optimum Values: K:3 P:3 Accuracy:0.984286