LAB-11: Machine Learning

Objective:

The objective of this lab is to apply use understand and use some well-known techniques of machine learning.

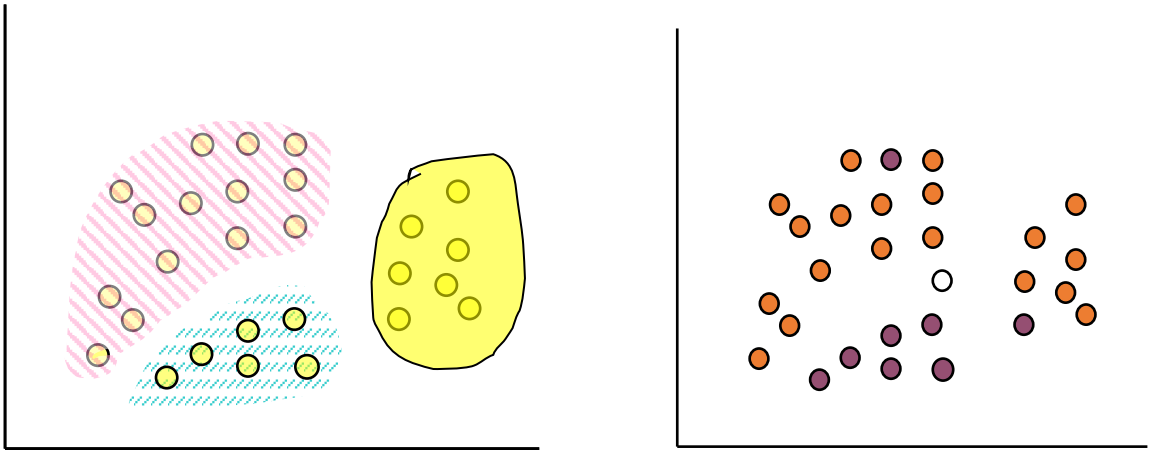
Theory:

Machine Learning is study of algorithms that improve their performance at some task with experience. It is to optimize a performance criterion using example data or past experience. According to Herbert Simon, learning is, "Any change in a System that allows it to perform better the second time on repetition of the same task or on another task drawn from the same population." [G. F. Luger and W. A. Stubblefield, *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*, The Benjamin/Cummings Publishing Company, Inc. 1989.]

Machine learning is programming computers to optimize a performance criterion using example data or past experience. Learning is used when:

- Human expertise does not exist (navigating on Mars),
- Humans are unable to explain their expertise (speech recognition)
- Solution changes in time (routing on a computer network)
- Solution needs to be adapted to particular cases (user biometrics)

Classification is one of the mostly used machine learning technique to enable any system to make proper decision. The purpose of classification is to decide proper class of unseen sample.

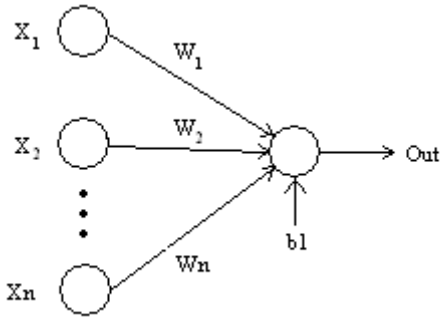


Minimum Distance Classification relies on calculating the distance of an unknown sample from all the classes and assigning the unknown sample to that class from which the distance is

minimum. The mean of every feature of every class is calculated and the distance of the unknown sample is calculated from this mean.

K-Nearest Neighbors classification checks k number of closest neighbors and then assigns the unknown sample to that class which has the most samples in the predefined number k.

A **perceptron** can be termed as the simplest neural network. A perceptron functions as a binary classifier meaning that it can differentiate between two classes. The final weights of a perceptron are calculated by running it for the entire test data and adjusting the weights accordingly until the weights are able to correctly classify all the test sample. A pictorial representation of a perceptron is as follows:



Some Useful Commands:

1. To read a csv file:

   import csv

   with open('some.csv', 'rb') as f:

   reader = csv.reader(f)

   for row in reader:

   my_list.append(row)

2. To randomly shuffle a list:

   import random

   shuffle (my_list)

3. To randomly generate a series of numbers: **random.sample(range(starting_point, ending_point), number_of_random_numbers_required)**

4. To use a kNN classifier: **scipy** and **scikit-learn** need to be installed first.

   from sklearn import neighbors

   my_classifier = neighbors.KNeighborsClassifier(n_neighbors=3)

   my_classifier.fit(feature_set, corresponding_classes)

my_classifier.predict_proba(feature_set_of_unknown_sample)

Lab Tasks:

Lab Task 1:

Using the dataset.csv file as input data, write the code for minimum distance classification.

Lab Task 2:

Shuffle the dataset.csv randomly and then split it into two equal parts namely Training Samples and Testing Samples. Now using the kNN algorithm, find out the accuracy when the number of k is varied. (Accuracy can be found by finding out the correct number of right guesses/correct predictions by the algorithm).

Lab Task 3:

A perceptron can be easily implemented by using the equation:

$$\textbf{margin} = -1\ [1\ w_1\ w_2\ \ldots\ w_n] \begin{bmatrix} 1 \\ x1 \\ x2 \\ \ldots \\ xn \end{bmatrix}$$

If margin > 0 then the next sample is tested. If it is less than zero then the weights are adjusted using the equation:

$$\Delta w = \eta\ y\ x$$

$$w_{new} = w + \Delta w$$

where η is 0.1, y is the label of the misclassified sample and x is the feature vector.

The process continues until all the samples yield a margin of greater than 0.

For testing a sample, the weights calculated after the training are multiplied with feature set of the unknown sample and then the value of the margin is checked.

Conclusion:

This lab gave an idea of different classification algorithms in supervised learning and their implementation.