Machine Learning, Spring 2023: Project 1

KAMAL SUBEDI(G35111003)

Platform: Jupyter Notebook  
Device: MacOS M2  
Programming language: Python3, Python libraries  
Dataset: <https://www.kaggle.com/c/digit-recognizer/data>

**Data Description:**

It is the collection of 28x28 grayscale images of handwritten digits, along with their corresponding labels. The dataset contains a training set of 42,000 images and a test set of 28,000 images. Each image in the dataset is labeled with the digit it represents, ranging from 0 to 9. The training set is provided as a CSV file with 42,000 rows and 785 columns. The first column of the CSV file contains the label (the digit it represents), while the remaining 784 columns contain the pixel values of the corresponding image (28x28=784). Each pixel value is an integer between 0 and 255, representing the grayscale intensity of the pixel. The test set is provided as a CSV file with 28,000 rows and 784 columns, containing only the pixel values of the corresponding images. The dataset is commonly used for image classification tasks.

**Algorithm Description**

First, load the train.csv file and split it into train and test in the ratio of 80:20. As 80:20 split will provide enough data to train and to evaluate its performance. Then define a function to compute KNN of a given data point from training dataset using Euclidean distance metric. Then define. A function to predict the label of a given test data using the label of its K nearest neighbor based on majority vote. After that, I use K value as 5 to get the prediction and its accuracy and confusion matrices. Then I use k values from 5 to 15 to get the different accuracy to measure its performance. After getting a good accuracy rate, I use this model to test the data from test datset, randomly choose 20 data points to evaluate.

**Algorithm Result:**

Here, for k = 5, we got 92.8% accuracy rate of our algorithm along with confusion matrix.

  
Then, calculate the confusion for k = 5, and got this:  
True Positives(TP) = 91

True Negatives(TN) = 110

False Positives(FP) = 0

False Negatives(FN) = 0

Then, I took the accuracy value for k range. From 5 to 15 and plot the graph between K values and accuracy as below:

Chart, line chart

Description automatically generated

Fig: Graph between different values of K and Accuracy to get the optimal one.

From the graph, we can see that, we got the maximum accuracy which is 93.1% when K=8 and 9.

Then, I test the algorithm using test.csv file and got it as output.

Graphical user interface, chart

Description automatically generated with medium confidence

Fig: Output showing the label of the handwritten digit.

**Runtime:**

