Rutgers-The State University of New Jersey

CS512 Final Project

Image Classification using k-Nearest Neighbours algorithm

Group 2: Abhishek Bhatt, Harsh Bhatt, Siva Harshini Dev Bonam, Tianzhi Cao

19th April 2020

OVERVIEW

The goal of our project is to build an application for image classification by implementing the k-Nearest Neighbour algorithm. For the purpose of development and evaluation, we will use the MNIST handwritten digit dataset. We aim to achieve comparable accuracy and complexity as some of the existing library implementations. Given a test image of a handwritten digit as an input, the application tells the number it represents as the output. On a high level, the algorithm exploits the idea of finding the k-nearest neighbors of the input image vector in a pre-labeled dataset, in order to classify it as the actual digit it represents. By the end of the project, we would have implemented a known algorithm not covered in class, and explored an application of the same.

GOALS

1. Implement the kNN algorithm to find the nearest neighbors of an input test image from the MNIST dataset with pre-labeled images.
2. Write an application on top of the algorithm that takes an image as input and returns the digit it contains as an output.
3. Evaluate the classification error of the application using available metrics and achieve values similar to some existing implementations.
4. Analyze the time and space complexity of the algorithm with the methodology learned in class.

SPECIFICATIONS

- DATA: MNIST datasets include 60,000 hand-written number images that were written by 250 different people. Each image is 28x28 pixels. We will split it into two parts: the training set and the test set. The data format is binary files of images and labels. The four files (images and labels for both train and test sets) total to around 55 MB when uncompressed. The data is at rest and can fit in a typical desktop.

- QUESTIONS: We would like to explore the training dataset to look for any visible trends, for example, the mean, mode, and median of the labels. For the user, we will answer the question of which digit does an input image contain. The application will have an interface where the user loads an image and gets the result.

- ALGORITHM: The main modules to be implemented in the application are: loading the data, vectorizing the input, dimensionality reduction (if needed), computing Euclidean distance between test input and pre-labeled image vectors, and classifying the input based on the mode of the k “nearest” labels to it.
  If we have n elements and the dimension of the data is d, then the time complexity of this algorithm is O(nd+kn) or O(ndk) depending on the preprocessing of the data. We have a preprocessed dataset so it would be of the order O(nd+kn).

- IMPLEMENTATION: We are doing an implementation of a popular algorithm used for data classification problems. We will use Python programming language for our development as it is widely used for image processing applications and data mining algorithms.
  Some implementations for the algorithm as built-in functions are available, for e.g. sklearn.neighbors package in the sci-kit learn library, that we can benchmark against for the results obtained.
  Given the team members have a good understanding of Python and data mining techniques, the project is feasible for completion within the stipulated time. We aim to achieve the given timeline by an efficient division of modules and tasks amongst the team members and finally combining each unit as the final application.

REFERENCES

http://yann.lecun.com/exdb/mnist/

https://blog.usejournal.com/a-quick-introduction-to-k-nearest-neighbors-algorithm-62214cea29c7