

**Course: IT Workshop – III (CSP 211)**

**PROJECT REPORT**

**Submitted By: Submitted To:**

Vishvas Bansal (2019KUEC2031) Dr. Ankush Vashishtha

Achalesh Lakhotiya (2019KUEC2036)

Shalin Gupta (2019KUEC2032)

Bhavesh Soni (2019KUEC2029)

**What is Machine Learning and How it works:**

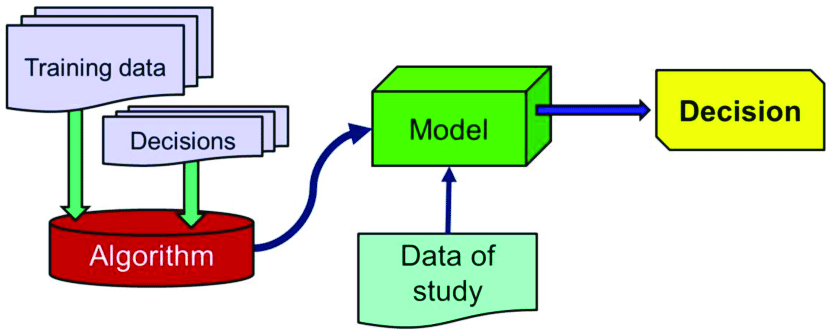
Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed.

In it, first we train the model for a certain dataset (In our case it is MNIST dataset). After that we apply the algorithm for the dataset and then check test cases with the training data to predict the most appropriate output.

**Abstract:**

Our project is **Handwriting Classification**.

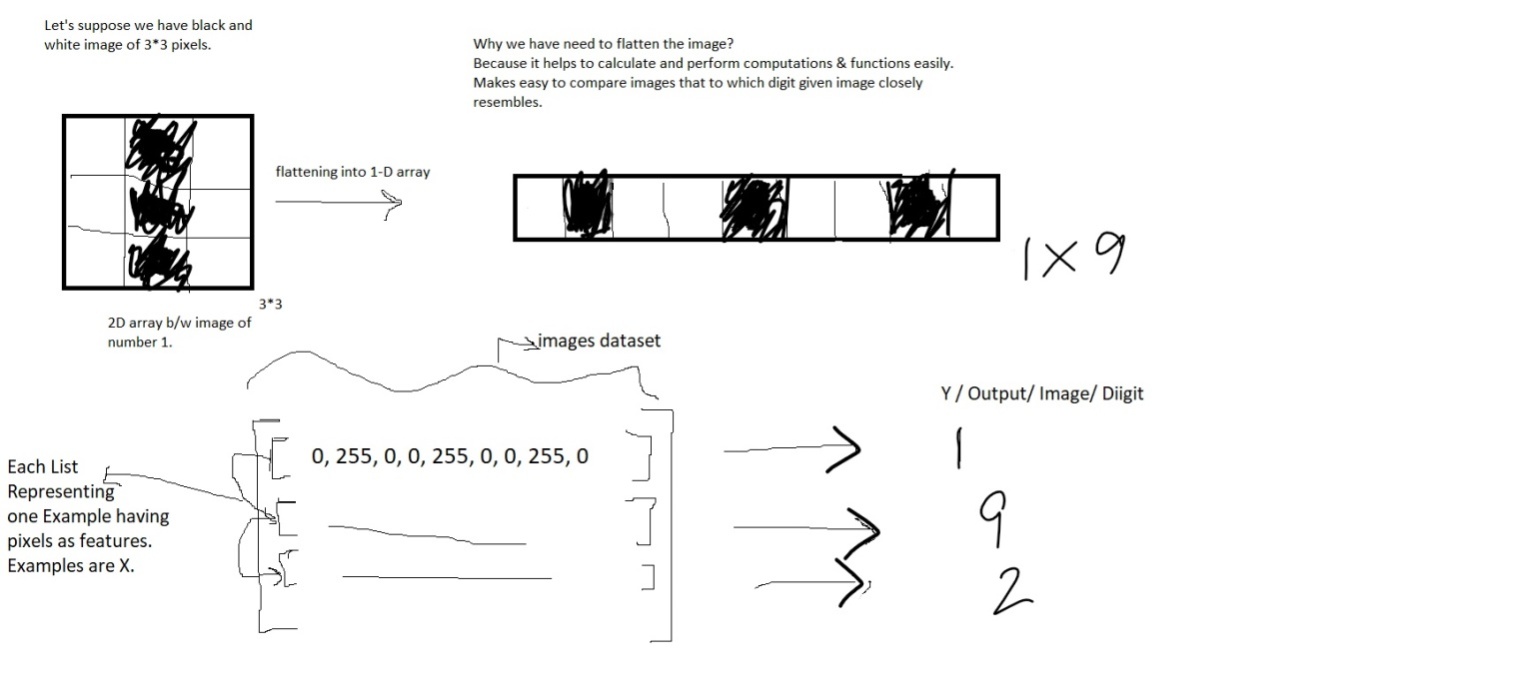
In this project we do supervised machine learning, i.e. we have input and output variables and we use the algorithm to learn the mapping of input on output. The goal being to use the mapping so well that when we have new input data we can predict the output for that data. It’s called supervised learning because the process of an algorithm learning from the dataset can be thought of as a teacher teaching the student. We know the correct answers and the algorithm iteratively makes predictions on the training data and is corrected by the dataset and the learning stops when the algorithm achieves an acceptable accuracy.



We have created a model which classifies and recognises images of handwritten digits.

So First we flatten the image which in this case is a 28 by 28 matrix making the calculations easy.

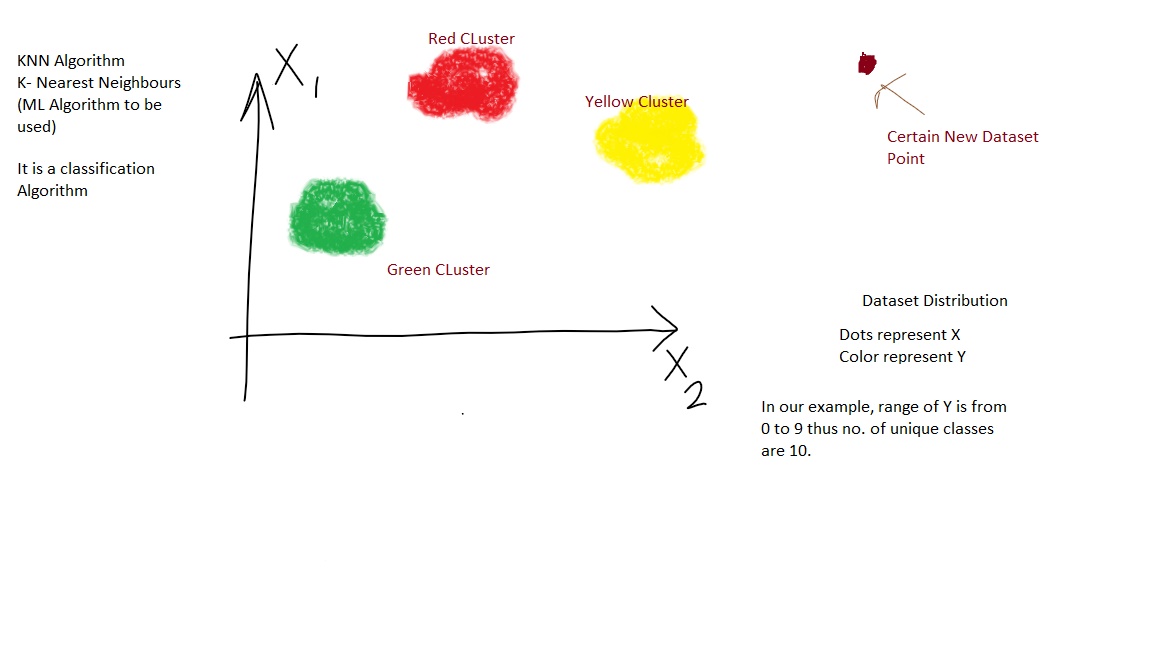
Flattening the image means to append the rows containing pixels one after the other. Pixel is a minute area of illumination on a display screen, one of many from which an image is composed.

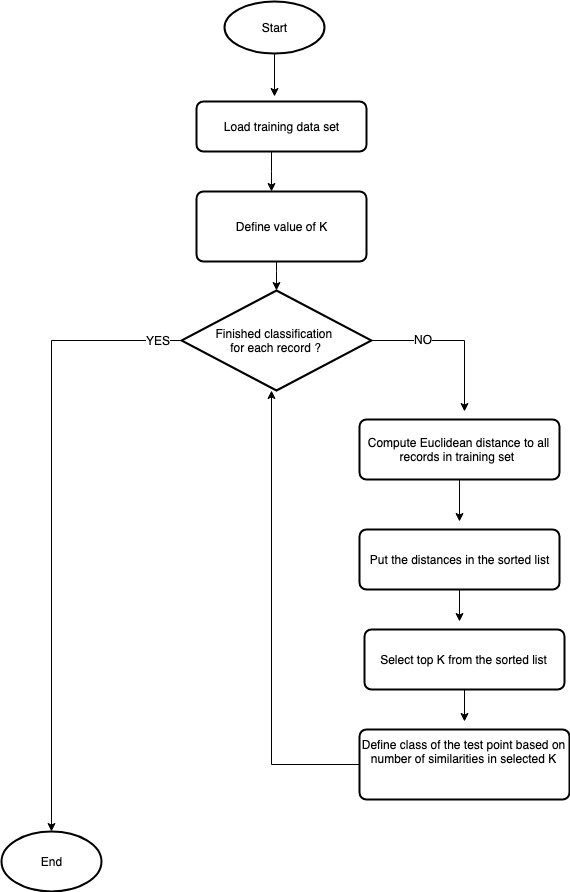


**KNN (K- Nearest Neighbours) Algorithm:**

It is a brute force algorithm as we compare point with every point. In this:

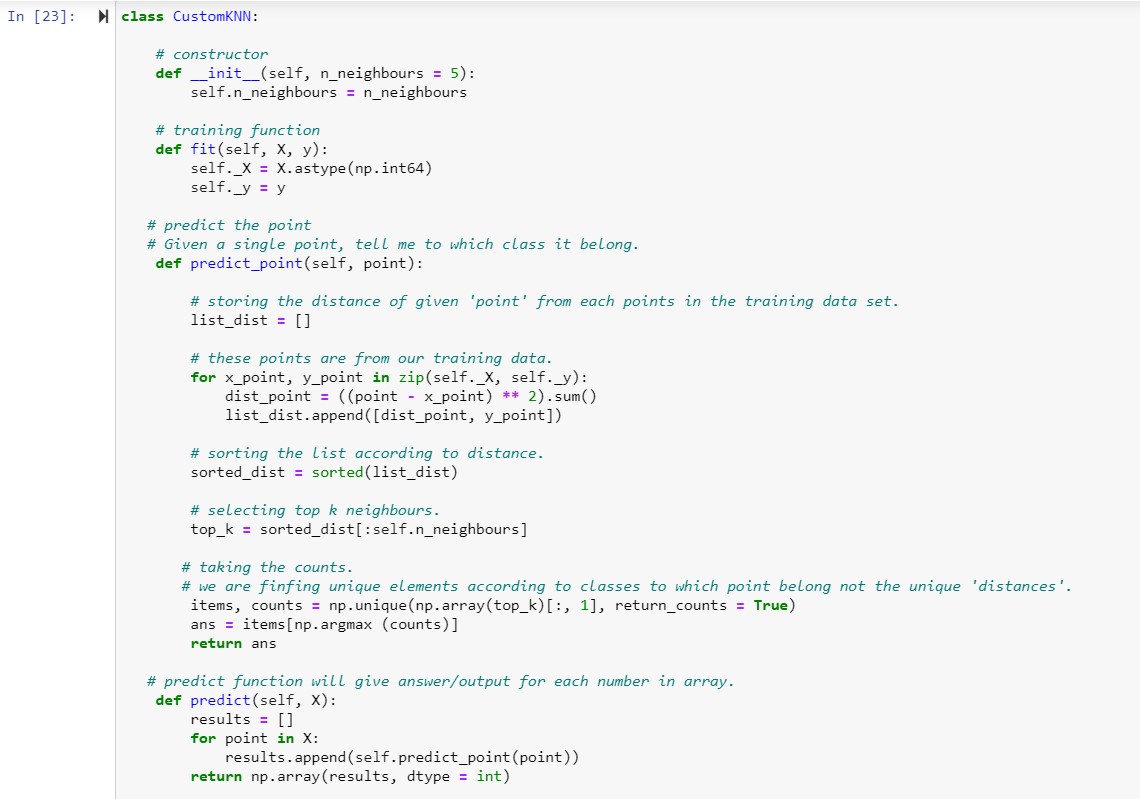
1. First we take the distances of given point from the every point in every cluster.
2. We get something like this { [2,Y],[5,R],[3,G],[1,R],[4,Y],[2,R],[6,G],[4,R] }
3. Sort the distances
4. Not it looks like this { [1,R],[2,R],[2,Y],[3,G],[4,R],[4,Y],[5,R],[6,G] }
5. Now choose some top K cases, like we took K=5
6. Now we find the most occurring cluster, here it is R





**Our Source Code:**

We have made both custom class for the algorithm and also used inbuilt class available to apply KNN algorithm on the MNIST dataset. As can be seen in these blocks of code





**Note:** You can also find the Source Code files and Assignments on following link too:

<https://github.com/achalesh27022003/Python-Assignments-IWT>

**THANK YOU**