A take on digits recognition

By : Loay Hany : 1610
Mohamed Nader :1671
Ahmed Nabil: 1450

Character recognition is one of the most well known applications for machine learning, digit recognition follows the exact same patter, but with 10 discrete values to be recognized instead of 26 letters.

In the following report, we explain our take on digits recognition using python.

Data set:
We used Sklearn digits data set, it has 1797 images, each 8*8 and pixel values from 0 to 15 all natural numbers, originally they were 16*16 images with normal gray scale from 0 to 255, but for computation efficiency and better features, they used intensity windowing and averaging each 4 pixels in one, which gave very good results.

Algorithms:
We used support vector classifiers (SVC) based on support vector machine (SVM) algorithm. SVM is known for its efficiency, as well as its very good results in character recognition, in fact it is used as the main algorithm in OCR (optical charter recognition) beating even neural networks.

We also implemented KNN algorithm using cosine similarity as our similarity measure tool and taking multiples of eleven as nearest neighbors.

Testing:
We made multiple testing on the resulted model, we used cross validation with 10 cuts for SVC, as well as testing some random digit each time we run the algorithm and displaying it to make sure the tests are going correctly.
For KNN we split the data into training and testing with multiple sizes for datasets and also neighbors.

Results:
The SVM algorithm took very little time to converge scoring 98% in accuracy and 97% in precision.
The KNN algorithm took a little more time to converge scoring 94% in accuracy and 91% in precision
 N.P: we used accuracy and precision measurement not to be confused with precision and recall measurement.