

## **Machine learning Lab Report06**

*Name: Raman Kumar*

*Reg NO: BL.EN.P2DSC22009*

*Course: Mtech Data Science*

**1. Compare the results of the SVM classifier with those obtained in earlier classes from k-NN and MLP classifiers. Provide comparative accuracies along with the time taken to train and test.**

	Accuracy(%)	Time Taken(sec)
SVM	78.70	170 sec
MLP	69.89	0.53 sec
KNN	70.24	0.22 sec

KNN vs SVM vs MLP:

MLP classifier: For this model, it calculated an accuracy of 69.89% when using the SGD algorithm with the ReLU activation function and learning rate of 0.05. The time taken by the classifier to classify the labels is 0.53 seconds which is not better than SVM.

KNN Classifier: For this model, it calculated an accuracy of 70.24% upon using the Euclidean distance measure and the nearest neighbor value used for it was k=5 and the time taken for the model was 0.22 seconds.

SVM Classifier: For this model, it calculated an accuracy of 78.70% when regularised parameter \*C was taken as 1 and kernel function opted as linearSVC. The time calculated for this model was most of all i.e 170 sec.

**2. Which classifier would you choose for your classification problem? Provide justifications for your choice.**

KNN vs SVM: SVM wins

- SVM takes better care of outliers than KNN.
- If training data << large features, then SVM gives better results.

KNN vs Neural: KNN wins

- Neural needs large training data compared to KNN to achieve sufficient accuracy.
- Neural needs a lot of parameter tuning.

SVM vs NN: SVM wins

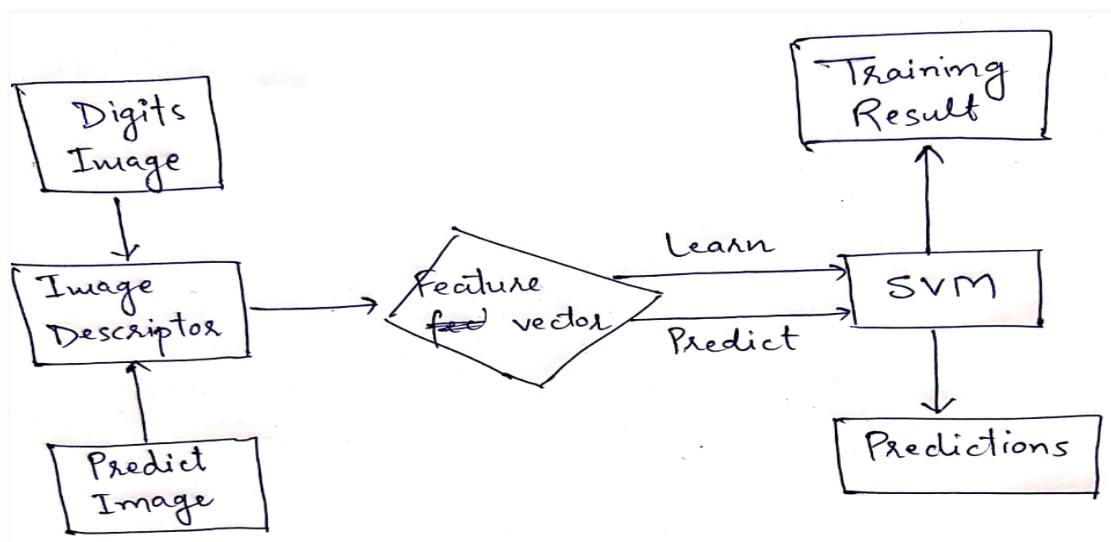
- SVM has a convex parameter fun.
- Whereas, NN will be stuck in local minima only
- SVM performs better when less data there, but for NN we need a lot more of data.
- But NN can do multiclass classification better in comparison to SVM. SVM needs multiple models to classify.

So, I think SVM is actually better in terms of accuracy but if we consider the time taken to model it is not that good.

So if time constraints or limits will be there, we can always go with the KNN classifier.

**3. A scanned image of a paper containing multiple printed digits (0 to 9) available in boxes. Each box contains a single digit. Design an SVC classifier for recognizing these printed digits. The designed system should be represented with:**

- a flow diagram for the system detailing various action steps
- parameters associated with each action.
- Short description of each action step of the flow diagram



a). Image acquisition:

- Obtaining text images from the scanner in format PNG, BMP, etc.

b). Pre-Processing:

- To make the image readable for the recognition systems.

c). Feature Extraction/Vector:

- (Reduce the redundant data)

Pixel value 255 - white

Pixel value 0 - black

$Y(i, j) = \begin{cases} 0 & , \text{ if } Y(i, j) > T(\text{threshold value}) \end{cases}$

$\{ 255 \text{ otherwise}$

d). Train algorithm using SVC classifier