Assignment 3 - NN & SVM Comparison

Conclusions Drawn

Conclusion : Dataset has two feature vectors, labelling them as **x** and **y**. **The dataset has 40000 entries.**

Given : Please note that the last column of the attached data has the target (label) information. This label has no physical significance. You may change it at will provided you do it consistently across the board.

Labelling the three columns as x, y, class for first feature(0), second feature(1) and target(2) respectively.

Number of unique classes : 4

Labelled them as : Class A, Class B, Class C, Class D.

Class	Target Value
Class A	1
Class B	-1
Class C	2
Class D	-2

Neural Network

Creating a neural network (of an appropriate architecture) to classify the samples.

- The architecture of the Neural Network created has 1 hidden layer with 16 nodes and an output layer with 4 nodes (for 4 unique values in the dataset target values).
- Using too few nodes in the hidden layer would cause underfitting.

- Used **mean-square-error** as loss function.
- Used **stochastic gradient descent** as optimizer.

Note : Initially loss decreases with the number of epochs, thereafter decrease in loss per epoch is very less. Therefore no need to train on a large number of epochs.

Accuracy - 97.5%.

Support Vector Machine

- Used a multi-class SVM approach to perform the classification.
- With SVM, compared both one-one and one-rest strategies.
- Also did the timing test between these two schemes.
- SVM with a linear kernel is useful when the number of feature vectors is large and the number of training examples is less.
- But our dataset had only two feature vectors therefore an SVM with gaussian kernel gave better results.

Time Comparisons: One-One strategy takes more time as compared to One-Rest strategy.

- One-Rest strategy : 2.7998199462890625
- One-One strategy : 5.3945066928863525

Accuracy : One-One strategy is more accurate as compared to One-Rest strategy.

- One vs Rest strategy : 97.225%
- One vs One strategy : 97.525%

Did a comparative analysis between the neural network and the better SVM strategy.

- Neural networks perform better when they feed upon a large dataset as compared to our dataset.
- Our dataset consisted of **40000 rows and 2 feature vectors**. Thus both neural networks and SVM perform reasonably well and both of their performance was comparable.
- Neural networks took a lot more time in training, the training time of a neural network varies as it depends on the number of epochs, batch size etc.
- SVM performed a lot better in terms of less training time but **as the size of the dataset will increase neural networks will definitely perform better.**
- SVM's have one more benefit, as SVM is a convex optimization problem, it is sure that we will reach global minima whereas when training neural networks we can get stuck at local minima.
