RBFNN

1. Changing cluster centres does not seem to have a significant effect on the performance of the network. The network was tested on one instance with cluster centres: -
   [[-0.19801968 0.43890627 0.33536348]
   [0.95654173 -1.49216101 -1.40554509]
   [0.32899446 1.0874627   1.27473459]
   [-1.34629474 0.13790367 -0.02453557]]
   And again with: -
   [[0.32899446 1.0874627   1.27473459]
   [0.95654173 -1.49216101 -1.40554509]
   [-1.34629474 0.13790367 -0.02453557]
   [-0.19801968 0.43890627 0.33536348]]
   Both times 80% accuracy was achieved.
2. Test size = 33%, Accuracy = 20%.
   Test size = 40%, Accuracy = 63.33%.
   Test size = 50%, Accuracy = 62.67%
   [Number of cluster centres = 4]
3. The weight matrix for the final layer was directly calculated using $W = (G^TG)^{-1}G^T\Phi$
4. 4 cluster centres → 62.67% accuracy
   6 cluster centres → 72% accuracy
   15 cluster centres → 65.33% accuracy
   30 cluster centres → 70.66% accuracy
5. Given the same number of iterations, the algorithm performed similarly on both networks. While the Perceptron took a few iterations to converge, the output accuracy was 100% very fast. However, the RBFNN was faster in processing.

MNIST Dataset

1. Changing initial clusters had some effect on the network. Accuracy varied between 14% and 30%.
2. Training and Test sets were taken as given in MNIST.
3. 30 cluster centres → 20% accuracy
   20 cluster centres → 14% accuracy
   100 clsuter centres → 25% accuracy
   500 cluster cntres → 35% accuracy
4. The perceptron neural network as well as the RBFNN took a long time to process the data. However, for a dataset as large as this we observed the benefits of RBFNN over perceptron. The RBFNN converged much faster than the neural network, however, the performance was significantly better for the neural network.