Neural Networks Assignment: Clustering with Competitive Neural Networks

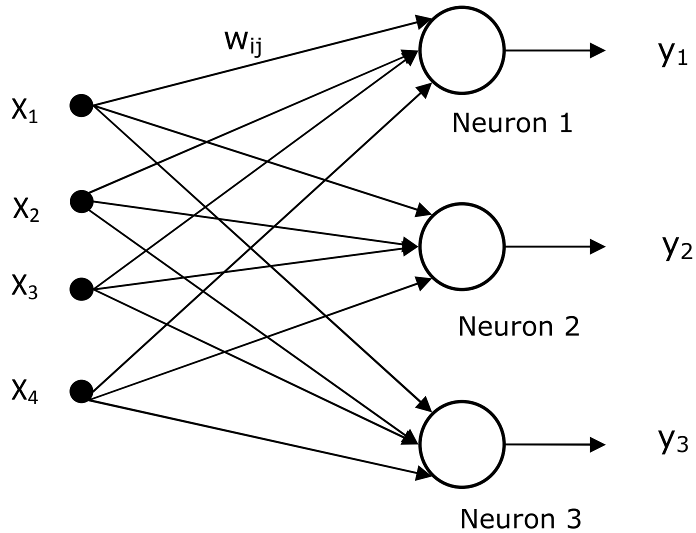

Figure 1

Q1)

I. Consider the network in Figure 1 and use it to cluster the Iris data found in iris.txt. Write a script to implement the following algorithm to do the clustering. In the iris.txt, the first four columns are the features and the last column is the class label.

> Let $X = \{x^{(1)}, x^{(2)}, \dots, x^{(l)}\}$ be the input iris dataset
> Algorithm:
> Initialize the weights randomly and normalize them
> Normalize each $x^{(p)} \in X$
> For each normalized $x^{(p)} \in X$
> find k such that $w_k . x^{(p)} \geq w_j . x^{(p)}, \forall 1 \leq j \leq m$ // This looks for the maximum dot product.
> assign $x^{(p)}$ to cluster (neuron) k
> $w_k = w_k + \rho(x^{(p)} - w_k)$ and normalize
>
> Repeat the for loop R times

In the above algorithm, w_k is the weight vector of neuron k. The iris data has 3 clusters (classes); therefore, the number of neurons (m) is 3. Repeat the loop 500 times (R=500).

II. Compute the purity of the clusters and report the values. Please include screen shots of your output. Please read the section at the end of the assignment for a definition of purity.

Q2)

Write a new script to cluster the same iris data using the same network in Figure 1, but this time using the following algorithm and report the purity of the clusters found. Let R=500. Please include screen shots of your output.

Let $X = \{x^{(1)}, x^{(2)}, \ldots, x^{(l)}\}$ be the input dataset
Algorithm :
Initialize the weights randomly
For each $x^{(p)} \in X$
find k such that $\lvert w_k - x^{(p)} \rvert < \lvert w_j - x^{(p)} \rvert, \forall 1 \leq j \leq m$ // This looks for the smallest Euclidian distance.
assign $x^{(p)}$ to cluster (neuron) k
$w_k = w_k + \rho(x^{(p)} - w_k)$
Repeat the for loop R times

Q3)

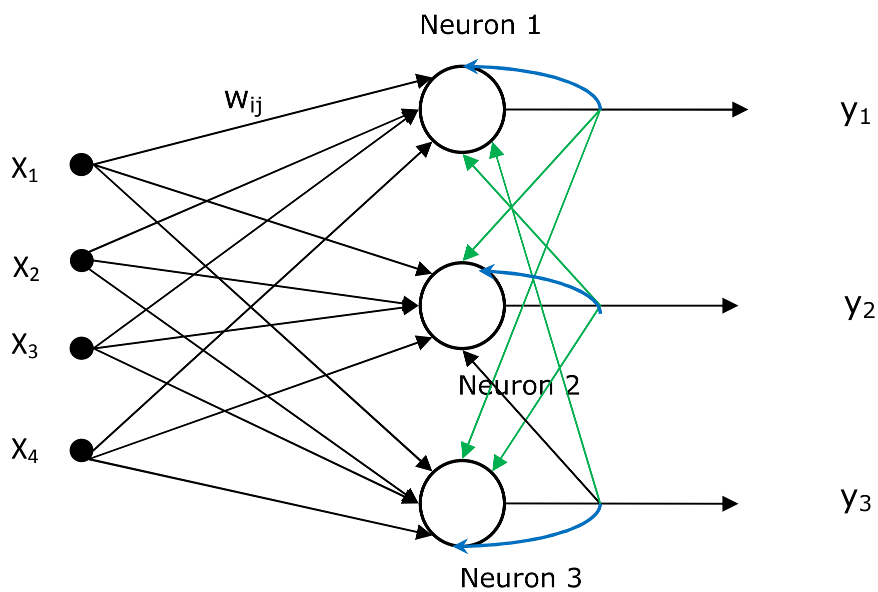

Figure 2

Consider the network in Figure 2, where the lateral connections have been added. Assume each neuron computes the weighted some of its inputs. For the feed-forward weights (w_{ij}'s), use the final weights learned in Q1. For the self feedback connections (blue ones), assume weights of 1. For the lateral inhibitory connections (green ones) assume weights of -ε, where 0 < ε <1/3 is a constant.

The expected behavior of this network is such that for any given normalized input vector $x^{(p)}$, only one of the output neurons will output a positive value.

I. Implement the network in Figure 2, and cluster the data in iris.txt (be sure to normalize the data as you did for Q1). What you need to do when clustering is, for each input vector $x^{(p)}$, assign it to the group represented by the neuron that has a positive value for it. Use ε = 0.1. Compute and report purity of the clusters.

II. Try using very low values (say, 0.001) and large values (say, 0.32) for ε and comment on the running time and the accuracy (purity) of the results.

Purity of clustering

Let $V=\{v_1, v_2,\ldots,v_K\}$ be a set of clusters and $C=\{ c_1, c_2,\ldots,c_M \}$ be the set of class labels in the dataset. Then the purity of this clustering is defined as below.

$$purity(V,C) = \frac{1}{N} \sum_{k=1}^{K} \underset{j}{Max}\{s_{jk}\} \quad , where\ s_{jk}\ is\ the\ number\ of\ data\ po\mathrm{int}\,s$$

$$in\ cluster\ k\ that\ belong\ to\ class\ j.\ N\ is\ the\ total\ number\ of\ data\ po\mathrm{int}\,s.$$

Please read the following link for more information in purity.

http://nlp.stanford.edu/IR-book/html/htmledition/evaluation-of-clustering-1.html