CSE514 Fall 18 – Datamining Homework 4

Assigned date: 4/10/2018 **Due date: 4/24/2018**, before class starts

**P1**: Autoencoder (24 pt)

**The problem**:

We discuss in class that autoencoder can automatically discover the binary code for decimal numbers. The case we consider in class is the binary code for 8 decimal numbers from 0 to 7. Recall that in this autoencoder we use a position encoding scheme (check your class notes to see what that is).

We now construct a few autoencoders for finding binary code for 16 decimal numbers from 0 to 15. We are going to use the same position encoding scheme here.

For these autoencoders, the input layers and the output layers all have 16 nodes. We will use the sigmod function (i.e., f(x) = 1/(1+e^(-x))) for perceptron activation and consider the following architectures (for problems 1-4):

1. A 3-layer NN (with layers for input, hidden and output). The hidden layer has 5 perceptrons. (4 pts)
2. A 3-layer NN. The hidden layer has 4 perceptrons. (4 pts)
3. A 3-layer NN. The hidden layer has 3 perceptrons. (4 pts)
4. A 5-layer NN. The 1st, 2nd, and 3rd hidden layers have 8, 4, and 8 perceptrons, respectively. (4 pts)
5. Repeat 1) and 4) above, but this time use the Rectified Linear Unit (ReLU), i.e., f(u) = max(0, u), for all perceptron activation. Compare the average running time of 5 runs of the two methods (correspondingly, each with the same initial random weights). (8 pts)

What to report: Run your program 5 times with different initial weights and compare the stable states of **all hidden layers** from all these 5 runs on each autoencoder architecture. Report and **compare** the stable states of all hidden layers after you train the autoencoders **using different initial weights**. Discuss the correspondence of these states and the input values. For problem 5) above, compare and discuss the stable states using the two different activation functions. Another factor to consider it the running time using the two activation functions – the number of steps to convergence and total running time. From these comparison, what conclusion can you derive regarding which activation function to use? If you cannot get stable states for one or both of these activation functions, explain the reason(s) why you cannot have stable states.