Kody Quintana
CS 473
Artificial Neural Networks
April 10, 2019

Assignment #8

0.1 Given: Quiz problem # 2:
Find: Write Matlab/Python/C/C++ code to implement the forward propagation of the ANN. Calculate W's and $\hat{\vec{y}}$.

$$\vec{X} = [x_0 \ x_1 \ x_2]^T = [1.0 \ 0.7 \ 0.3]^T$$

$$\vec{y} = [y_0] = 1.0$$

This class takes the number of nodes per layer as arguments for its constructor. It generates random weight matrices and 0 filled nodes.

```python
import numpy as np


class NeuralNet(object):
    def __init__(self, *nodes_per_layer):
        self.n_inputs = nodes_per_layer[0]
        self.n_outputs = nodes_per_layer[len(nodes_per_layer) - 1]

        self.weight_matrix_list = [None] * (len(nodes_per_layer) -1)
        self.node_array_list = [None] * (len(nodes_per_layer))

        #Create n x m matrices for the weights
        for i in range(0, len(nodes_per_layer) - 1):
            self.weight_matrix_list[i] = np.random.rand(
                nodes_per_layer[i + 1], #Rows
                nodes_per_layer[i]) #Columns

        #Create 1 x n arrays for all nodes except inputs
        for i in range(0, len(nodes_per_layer)):
            self.node_array_list[i] = np.zeros((1, nodes_per_layer[i])).T


    def __str__(self):
        nodes = "\nLayers:\n"
        matrices = "\nMatrices\n"

        for i, matrix in enumerate(self.weight_matrix_list):
            matrices += str(i) + ":\n" + str(matrix) + "\n\n"
```

```python
29            for i, node in enumerate(self.node_array_list):
30                nodes += str(i) + ":\n" + str(node) + "\n\n"
31
32            return( matrices + nodes )
33
34        def set_inputs(self, X):
35            for i in range(len(X)):
36                self.node_array_list[0][i] = X[i]
37
38        def forward(self):
39            for i in range(len(self.weight_matrix_list)):
40                self.node_array_list[i+1] = np.matmul( self.weight_matrix_list[i],
                   ↪  self.node_array_list[i] )
41
42
43  TEST = NeuralNet(3, 3, 2, 1)
44  X = [1.0, 0.7, 0.3]
45  TEST.set_inputs(X)
46  print(TEST)
47  TEST.forward()
48  print(TEST)
```