MSDS_458_X_OR_RT_commented code – full run 1

```
Calling the main() procedure from the program
Starting main() function

Calling welcome() function

starting the welcome() function
*****************************************************************************
*

Welcome to the Multilayer Perceptron Neural Network
  trained using the backpropagation method.
Version 1.0, 03/25/2017, A.J. Maren
For comments, questions, or bug-fixes, contact:
alianna.maren@northwestern.edu

*****************************************************************************
*

ending the welcome() function and returning to main

setting alpha, summedInput, maxNumIterations, and eta

alpha =  1.0 summedInput =  1
maxNumIterations =  2 eta =  0.5
Setting the debug_xxx_off statements to be false, so will print extra

calling the obtainNeuralNetworkSizeSpecs() function
Starting the obtainNeuralNetworkSizeSpecs() function

This network is set up to run the X-OR problem.
The numbers of nodes in the input, hidden, and output layers have been set to
2 each.
The number of input nodes is  2
The number of hidden nodes is  2
The number of output nodes is  2
Ending the obtainNeuralNetworkSizeSpecs() function

Flow-of-control trace: Back in main
I = number of nodes in input layer is 2
H = number of nodes in hidden layer is 2
O = number of nodes in output layer is 2

Initialize the trainingDataList to (0,0,0,0,0

Calling initializeWeightArray for input-to-hidden weights
starting the initializeWeightArray() function
initialize wt00 by calling initializeWeight() function
starting the InitializeWeight() function
weight =  -0.8743401897182881
ending the InitializWeight() function
```

```
initialize wt01 by calling initializeWeight() function
starting the InitializeWeight() function
weight =  0.021852356655784666
ending the InitializWeight() function

initialize wt02 by calling initializeWeight() function
starting the InitializeWeight() function
weight =  -0.5701931574863277
ending the InitializWeight() function

initialize wt03 by calling initializeWeight() function
starting the InitializeWeight() function
weight =  0.994528119889952
ending the InitializWeight() function


  Inside initializeWeightArray

    The weights just initialized are:
      weight00 = -0.8743,
      weight01 = 0.0219,
      weight10 = -0.5702,
      weight11 = 0.9945,

    The weightArray just established is:  [[-0.87434019 -0.57019316]
 [ 0.02185236  0.99452812]]

    Within this array:
       weight00 = -0.8743    weight10 = -0.5702
       weight01 = 0.0219    weight11 = 0.9945
  Returning to calling procedure

ending the initializeWeightArray() function and returning to main


Calling initializeWeightArray for Hidden-to-Output weights
starting the initializeWeightArray() function
initialize wt00 by calling initializeWeight() function
starting the InitializeWeight() function
weight =  -0.48096945530919366
ending the InitializWeight() function

initialize wt01 by calling initializeWeight() function
starting the InitializeWeight() function
weight =  -0.941926260530972
ending the InitializWeight() function

initialize wt02 by calling initializeWeight() function
starting the InitializeWeight() function
weight =  0.09592486977987913
ending the InitializWeight() function

initialize wt03 by calling initializeWeight() function
starting the InitializeWeight() function
weight =  -0.24989251579461058
ending the InitializWeight() function
```

```
  Inside initializeWeightArray

    The weights just initialized are:
      weight00 = -0.4810,
      weight01 = -0.9419,
      weight10 = 0.0959,
      weight11 = -0.2499,

    The weightArray just established is:  [[-0.48096946  0.09592487]
 [-0.94192626 -0.24989252]]

    Within this array:
       weight00 = -0.4810    weight10 = 0.0959
       weight01 = -0.9419    weight11 = -0.2499
  Returning to calling procedure

ending the initializeWeightArray() function and returning to main


calling initializeBiasWeightArray() function for Hidden nodes

starting initializeBiasWeightArray() function
initialize biasWeight0 by calling initializeWeight() function
starting the InitializeWeight() function
weight =  0.8312551787243612
ending the InitializWeight() function

initialize biasWeight1 by calling initializeWeight() function
starting the InitializeWeight() function
weight =  -0.5426447109691015
ending the InitializWeight() function


biasWeight0 =  0.8312551787243612
biasWeight1 =  -0.5426447109691015
The biasWeightArray =  [ 0.83125518 -0.54264471]
ending initializeBiasWeightArray() function
returning to main


calling initializeBiasWeightArray() function for Output nodes

starting initializeBiasWeightArray() function
initialize biasWeight0 by calling initializeWeight() function
starting the InitializeWeight() function
weight =  0.4909309716780714
ending the InitializWeight() function

initialize biasWeight1 by calling initializeWeight() function
starting the InitializeWeight() function
weight =  0.0042846390183264305
ending the InitializWeight() function


biasWeight0 =  0.4909309716780714
biasWeight1 =  0.0042846390183264305
```

```
The biasWeightArray =  [0.49093097 0.00428464]
ending initializeBiasWeightArray() function
returning to main


The initial weights for this neural network are:
       Input-to-Hidden                            Hidden-to-Output
  w(0,0) = -0.8743   w(1,0) = -0.5702         v(0,0) = -0.4810   v(1,0) =
0.0959
  w(0,1) = 0.0219   w(1,1) = 0.9945         v(0,1) = -0.9419   v(1,1) = -
0.2499

       Bias at Hidden Layer                          Bias at Output Layer
       b(hidden,0) = 0.8313                           b(output,0) = 0.4909
       b(hidden,1) = -0.5426                           b(output,1) = 0.0043

Establishing parameters - epsilon, iteration counter, SSE_InitialTotal

Initialize SSE_InitialArray to zeros
calling the computeSSE_Values() function to initialize the SSE_InitialArray
starting the computeSSE_Values() function

Compute a single feed-forward pass and obtain the Actual Outputs for zeroth
data set

Starting ComputeSingleFeedforwardPass() function
The inputs transferred in are:
Input0 =  0
Input1 =  0

The initial weights for this neural network are:
     Input-to-Hidden             Hidden-to-Output
w(0,0) = -0.874   w(0,1) = -0.570         v(0,0) = -0.481   v(0,1) = 0.096
w(1,0) = 0.022   w(1,1) = 0.995         v(1,0) = -0.942   v(1,1) = -0.250

  For hiddenActivation0 from input0, input1 =  0 ,  0

computing hiddenActivation0 by calling computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.6966202653352188
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:  0 ,  0
    The summed neuron input is 0.8313
    The activation (applied transfer function) for that neuron is 0.6966
ending computeSingleNeuronActivation() function and returning activation


  For hiddenActivation1 from input0, input1 =  0 ,  0
computing hiddenActivation1 by calling computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
```

```
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.3675725698433179
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:  0 ,  0
    The summed neuron input is -0.5426
    The activation (applied transfer function) for that neuron is 0.3676
ending computeSingleNeuronActivation() function and returning activation


  In computeSingleFeedforwardPass:
  Input node values:  0 ,  0
  The activations for the hidden nodes are:
    Hidden0 = 0.6966 Hidden1 = 0.3676

computing the activation of output node0 by calling
computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.5476393656209758
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:
0.6966202653352188 ,  0.3675725698433179
    The summed neuron input is 0.1911
    The activation (applied transfer function) for that neuron is 0.5476
ending computeSingleNeuronActivation() function and returning activation

computing the activation of output node1 by calling
computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.32218817918715204
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:
0.6966202653352188 ,  0.3675725698433179
    The summed neuron input is -0.7437
    The activation (applied transfer function) for that neuron is 0.3222
ending computeSingleNeuronActivation() function and returning activation


  Computing the output neuron activations

  Back in ComputeSingleFeedforwardPass (for hidden-to-output computations)
  The activations for the output nodes are:
    Output0 = 0.5476 Output1 = 0.3222

creating the actualAllNodesOutputList
the hiddenActivation0 =  0.6966202653352188
the hiddenActivation1 =  0.3675725698433179
the outputActivation0 =  0.5476393656209758
the outputActivation1 =  0.32218817918715204

ending ComputeSingleFeedforwardPass() function and returning to main

  In computeSSE_Values

  Actual Node Outputs for (0,0) training set:
      input0 =  0    input1 =  0
      actualOutput0 = 0.5476   actualOutput1 = 0.3222
      error0 =        -0.5476   error1 =        0.6778
   Initial SSE for (0,0) = 0.7593

Ending a single feed-forward pass to obtain the Actual Outputs for zeroth
data set

Compute a single feed-forward pass and obtain the Actual Outputs for first
data set

Starting ComputeSingleFeedforwardPass() function
The inputs transferred in are:
Input0 =  0
Input1 =  1

The initial weights for this neural network are:
     Input-to-Hidden             Hidden-to-Output
w(0,0) = -0.874   w(0,1) = -0.570         v(0,0) = -0.481   v(0,1) = 0.096
w(1,0) = 0.022   w(1,1) = 0.995         v(1,0) = -0.942   v(1,1) = -0.250

  For hiddenActivation0 from input0, input1 =  0 ,  1

computing hiddenActivation0 by calling computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.5648973422025058
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:  0 ,  1
    The summed neuron input is 0.2611
    The activation (applied transfer function) for that neuron is 0.5649
ending computeSingleNeuronActivation() function and returning activation


  For hiddenActivation1 from input0, input1 =  0 ,  1
computing hiddenActivation1 by calling computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()
```

```
Starting the computeTransferFnction() function
The activation is  0.6110869378709286
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:  0 ,  1
    The summed neuron input is 0.4519
    The activation (applied transfer function) for that neuron is 0.6111
ending computeSingleNeuronActivation() function and returning activation


  In computeSingleFeedforwardPass:
  Input node values:  0 ,  1
  The activations for the hidden nodes are:
    Hidden0 = 0.5649 Hidden1 = 0.6111

computing the activation of output node0 by calling
computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.5690192991279437
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:
0.5648973422025058 ,  0.6110869378709286
    The summed neuron input is 0.2779
    The activation (applied transfer function) for that neuron is 0.5690
ending computeSingleNeuronActivation() function and returning activation

computing the activation of output node1 by calling
computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.336146803794963
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:
0.5648973422025058 ,  0.6110869378709286
    The summed neuron input is -0.6805
    The activation (applied transfer function) for that neuron is 0.3361
ending computeSingleNeuronActivation() function and returning activation


  Computing the output neuron activations

  Back in ComputeSingleFeedforwardPass (for hidden-to-output computations)
  The activations for the output nodes are:
    Output0 = 0.5690 Output1 = 0.3361
```

```
creating the actualAllNodesOutputList
the hiddenActivation0 =  0.5648973422025058
the hiddenActivation1 =  0.6110869378709286
the outputActivation0 =  0.5690192991279437
the outputActivation1 =  0.336146803794963

ending ComputeSingleFeedforwardPass() function and returning to main

  Actual Node Outputs for (0,1) training set:
     input0 =  0    input1 =  1
     actualOutput0 = 0.5690   actualOutput1 = 0.3361
     error0 =        0.4310   error1 =        -0.3361
  Initial SSE for (0,1) = 0.2987

Ending a single feed-forward pass to obtain the Actual Outputs for first data
set

Compute a single feed-forward pass and obtain the Actual Outputs for second
data set

Starting ComputeSingleFeedforwardPass() function
The inputs transferred in are:
Input0 =  1
Input1 =  0

The initial weights for this neural network are:
     Input-to-Hidden             Hidden-to-Output
w(0,0) = -0.874   w(0,1) = -0.570         v(0,0) = -0.481   v(0,1) = 0.096
w(1,0) = 0.022   w(1,1) = 0.995         v(1,0) = -0.942   v(1,1) = -0.250

  For hiddenActivation0 from input0, input1 =  1 ,  0

computing hiddenActivation0 by calling computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.4892304131816207
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:  1 ,  0
    The summed neuron input is -0.0431
    The activation (applied transfer function) for that neuron is 0.4892
ending computeSingleNeuronActivation() function and returning activation


  For hiddenActivation1 from input0, input1 =  1 ,  0
computing hiddenActivation1 by calling computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.372666973421586
```

```
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:  1 ,  0
    The summed neuron input is -0.5208
    The activation (applied transfer function) for that neuron is 0.3727
ending computeSingleNeuronActivation() function and returning activation


  In computeSingleFeedforwardPass:
  Input node values:  1 ,  0
  The activations for the hidden nodes are:
    Hidden0 = 0.4892 Hidden1 = 0.3727

computing the activation of output node0 by calling
computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.5723325062488019
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:
0.4892304131816207 ,  0.372666973421586
    The summed neuron input is 0.2914
    The activation (applied transfer function) for that neuron is 0.5723
ending computeSingleNeuronActivation() function and returning activation

computing the activation of output node1 by calling
computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.36594305801668164
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:
0.4892304131816207 ,  0.372666973421586
    The summed neuron input is -0.5497
    The activation (applied transfer function) for that neuron is 0.3659
ending computeSingleNeuronActivation() function and returning activation


  Computing the output neuron activations

  Back in ComputeSingleFeedforwardPass (for hidden-to-output computations)
  The activations for the output nodes are:
    Output0 = 0.5723 Output1 = 0.3659

creating the actualAllNodesOutputList
the hiddenActivation0 =  0.4892304131816207
the hiddenActivation1 =  0.372666973421586
the outputActivation0 =  0.5723325062488019
the outputActivation1 =  0.36594305801668164

ending ComputeSingleFeedforwardPass() function and returning to main

  Actual Node Outputs for (1,0) training set:
     input0 =  1    input1 =  0
     actualOutput0 = 0.5723   actualOutput1 = 0.3659
     error0 =        0.4277   error1 =        -0.3659
  Initial SSE for (1,0) = 0.3168

Ending a single feed-forward pass to obtain the Actual Outputs for second
data set

Compute a single feed-forward pass and obtain the Actual Outputs for third
data set

Starting ComputeSingleFeedforwardPass() function
The inputs transferred in are:
Input0 =  1
Input1 =  1

The initial weights for this neural network are:
     Input-to-Hidden             Hidden-to-Output
w(0,0) = -0.874   w(0,1) = -0.570         v(0,0) = -0.481   v(0,1) = 0.096
w(1,0) = 0.022   w(1,1) = 0.995         v(1,0) = -0.942   v(1,1) = -0.250

  For hiddenActivation0 from input0, input1 =  1 ,  1

computing hiddenActivation0 by calling computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.35131176652597756
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:  1 ,  1
    The summed neuron input is -0.6133
    The activation (applied transfer function) for that neuron is 0.3513
ending computeSingleNeuronActivation() function and returning activation


  For hiddenActivation1 from input0, input1 =  1 ,  1
computing hiddenActivation1 by calling computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.6162675800048231
Ending the computeTransferFnction() function and returning the activation
```

```
  In computeSingleNeuronActivation with input0, input 1 given as:  1 ,  1
    The summed neuron input is 0.4737
    The activation (applied transfer function) for that neuron is 0.6163
ending computeSingleNeuronActivation() function and returning activation


  In computeSingleFeedforwardPass:
  Input node values:  1 ,  1
  The activations for the hidden nodes are:
    Hidden0 = 0.3513 Hidden1 = 0.6163

computing the activation of output node0 by calling
computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.5941326261750242
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:
0.35131176652597756 ,  0.6162675800048231
    The summed neuron input is 0.3811
    The activation (applied transfer function) for that neuron is 0.5941
ending computeSingleNeuronActivation() function and returning activation

computing the activation of output node1 by calling
computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.3821043636347127
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:
0.35131176652597756 ,  0.6162675800048231
    The summed neuron input is -0.4806
    The activation (applied transfer function) for that neuron is 0.3821
ending computeSingleNeuronActivation() function and returning activation


  Computing the output neuron activations

  Back in ComputeSingleFeedforwardPass (for hidden-to-output computations)
  The activations for the output nodes are:
    Output0 = 0.5941 Output1 = 0.3821

creating the actualAllNodesOutputList
the hiddenActivation0 =  0.35131176652597756
the hiddenActivation1 =  0.6162675800048231
the outputActivation0 =  0.5941326261750242
the outputActivation1 =  0.3821043636347127
```

```
ending ComputeSingleFeedforwardPass() function and returning to main

  Actual Node Outputs for (1,1) training set:
     input0 =  1    input1 =  1
     actualOutput0 = 0.5941   actualOutput1 = 0.3821
     error0 =        -0.5941   error1 =        0.6179
  Initial SSE for (1,1) = 0.7348

Ending a single feed-forward pass to obtain the Actual Outputs for third data
set

Initializing array of SSE values
the SSE_InitialArray[0] =  0.7593377392113732 the SSE_InitialArray[1] =
0.2987390382257382
the SSE_InitialArray[2] =  0.3168138069220314 the SSE_InitialArray[3] =
0.7347885949248943

  The initial total of the SSEs is 2.1097

ending the computeSSE_Values() function and returning to main

The SSE_Array =  [0.7593377392113732, 0.2987390382257382, 0.3168138069220314,
0.7347885949248943, 2.1096791792840373]
The SSE_InitialTotal =  2.1096791792840373

In main, SSE computations completed, Total of all SSEs = 2.1097
  For input nodes (0,0), SSE_Array[0] = 0.7593
  For input nodes (0,1), SSE_Array[1] = 0.2987
  For input nodes (1,0), SSE_Array[2] = 0.3168
  For input nodes (1,1), SSE_Array[3] = 0.7348


About to enter the while loop for  2  iterations


starting at top of while loop for first time

create random trainingDataList by calling obtainRandomXORTrainingValues()
function
starting obtainRandomXORTrainingValues() function
The trainingDataList =  (0, 1, 1, 0, 1)
ending obtainRandomXORTrainingValues() function and returning to main

Iteration number  0

Randomly selected training data set number  1
The inputs and desired outputs for the X-OR problem from this data set are:
          Input0 =  0             Input1 =  1
 Desired Output0 =  1    Desired Output1 =  0

calling ComputeSingleFeedforwardPass to obtain actual outputs for training
set

Starting ComputeSingleFeedforwardPass() function
The inputs transferred in are:
Input0 =  0
```

```
Input1 =  1

The initial weights for this neural network are:
     Input-to-Hidden             Hidden-to-Output
w(0,0) = -0.874   w(0,1) = -0.570         v(0,0) = -0.481   v(0,1) = 0.096
w(1,0) = 0.022   w(1,1) = 0.995         v(1,0) = -0.942   v(1,1) = -0.250

  For hiddenActivation0 from input0, input1 =  0 ,  1

computing hiddenActivation0 by calling computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.5648973422025058
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:  0 ,  1
    The summed neuron input is 0.2611
    The activation (applied transfer function) for that neuron is 0.5649
ending computeSingleNeuronActivation() function and returning activation


  For hiddenActivation1 from input0, input1 =  0 ,  1
computing hiddenActivation1 by calling computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.6110869378709286
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:  0 ,  1
    The summed neuron input is 0.4519
    The activation (applied transfer function) for that neuron is 0.6111
ending computeSingleNeuronActivation() function and returning activation


  In computeSingleFeedforwardPass:
  Input node values:  0 ,  1
  The activations for the hidden nodes are:
    Hidden0 = 0.5649 Hidden1 = 0.6111

computing the activation of output node0 by calling
computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.5690192991279437
Ending the computeTransferFnction() function and returning the activation
```

```
  In computeSingleNeuronActivation with input0, input 1 given as:
0.5648973422025058 ,  0.6110869378709286
    The summed neuron input is 0.2779
    The activation (applied transfer function) for that neuron is 0.5690
ending computeSingleNeuronActivation() function and returning activation

computing the activation of output node1 by calling
computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.336146803794963
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:
0.5648973422025058 ,  0.6110869378709286
    The summed neuron input is -0.6805
    The activation (applied transfer function) for that neuron is 0.3361
ending computeSingleNeuronActivation() function and returning activation


  Computing the output neuron activations

  Back in ComputeSingleFeedforwardPass (for hidden-to-output computations)
  The activations for the output nodes are:
    Output0 = 0.5690 Output1 = 0.3361

creating the actualAllNodesOutputList
the hiddenActivation0 =  0.5648973422025058
the hiddenActivation1 =  0.6110869378709286
the outputActivation0 =  0.5690192991279437
the outputActivation1 =  0.336146803794963

ending ComputeSingleFeedforwardPass() function and returning to main

In main; have just completed a feedfoward pass with training set inputs 0 1
  The activations (actual outputs) for the two hidden neurons are:
    actualHiddenOutput0 = 0.5649
    actualHiddenOutput1 = 0.6111
  The activations (actual outputs) for the two output neurons are:
    actualOutput0 = 0.5690
    actualOutput1 = 0.3361'
  Initial SSE (before backpropagation) = 0.298739
  Corresponding SSE (from initial SSE determination) = 0.298739

Calling BackpropagateOutputToHidden() function

Starting the BackpropagateOutputToHidden() function
Performs the backpropagation of weight changes onto hidden-to-output weigts

Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function
```

```
Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function
Calling the PrintAndTraceBackpropagateOutputToHidden() function

Starting PrintAndTraceBackpropagateOutputToHidden() function

In Print and Trace for Backpropagation: Hidden to Output Weights
  Assuming alpha = 1

  The hidden node activations are:
    Hidden node 0:    0.5649   Hidden node 1:    0.6111

  The output node activations are:
    Output node 0:    0.569    Output node 1:    0.336

  The transfer function derivatives are:
    Deriv-F(0):       0.245    Deriv-F(1):       0.223

The computed values for the deltas are:
                eta  *  error  *   trFncDeriv *   hidden
  deltaVWt00 =   0.50 * 0.4310  * 0.2452   * 0.5649
  deltaVWt01 =   0.50 * -0.3361  * 0.2232   * 0.5649
  deltaVWt10 =   0.50 * 0.4310  * 0.2452   * 0.6111
  deltaVWt11 =   0.50 * -0.3361  * 0.2232   * 0.6111

Values for the hidden-to-output connection weights:
           Old:     New:      eta*Delta:
[0,0]:   -0.4810   -0.4511   0.0299
[0,1]:   -0.9419   -0.9631   -0.0212
[1,0]:   0.0959   0.1282   0.0323
[1,1]:   -0.2499   -0.2728   -0.0229

Ending PrintAndTraceBackpropagateOutputToHidden() function

Back in the BackpropagateOutputToHidden() function

Ending the BackpropagateOutputToHidden() function

Calling BackpropagateBiasOutputWeights() function

Starting the BackpropagateBiasOutputWeights() function

Calling the computeTransferFnctnDeriv() function for outputNode0

Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function

Calling the computeTransferFnctnDeriv() function for outputNode1

Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function

The newBiasOutputWeightArray =  [ 0.54377704 -0.0332213 ]
```

```
Ending the BackpropagateBiasOutputWeights() function

Calling BackpropagateHiddenToInput() function

Starting the BackpropagateHiddenToInput() function
This Backpropagates the weight changes onto the input-to-hidden weights

Calling the computeTransferFnctnDeriv() function for hiddenNode0

Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function

Calling the computeTransferFnctnDeriv() function for hiddenNode1

Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function

Calling the computeTransferFnctnDeriv() function for outputNode0

Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function

Calling the computeTransferFnctnDeriv() function for outputNode1

Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function

Calling the PrintAndTraceBackpropagateHiddenToInput() function

Starting PrintAndTraceBackpropagateHiddenToInput() function
Traces the backpropagation of the input-to-hidden weights

In Print and Trace for Backpropagation: Input to Hidden Weights
  Assuming alpha = 1

  The hidden node activations are:
    Hidden node 0:    0.5649   Hidden node 1:    0.6111

  The output node activations are:
    Output node 0:    0.569    Output node 1:    0.336

  The transfer function derivatives at the hidden nodes are:
    Deriv-F(0):       0.246    Deriv-F(1):       0.238

The computed values for the deltas are:
                eta  *  error  *   trFncDeriv *   input    * SumTerm for
given H
  deltaWWt00 =   0.50 * 0.4310  * 0.2458   * 0.0000   * 0.1093
  deltaWWt01 =   0.50 * -0.3361  * 0.2377   * 0.0000   * 0.1253
  deltaWWt10 =   0.50 * 0.4310  * 0.2458   * 1.0000   * 0.1093
  deltaWWt11 =   0.50 * -0.3361  * 0.2377   * 1.0000   * 0.1253
```

```
Values for the input-to-hidden connection weights:
           Old:     New:      eta*Delta:
[0,0]:   -0.8743   -0.8743   0.0000
[0,1]:   0.0219   0.0219   0.0000
[1,0]:   -0.5702   -0.5678   0.0024
[1,1]:   0.9945   0.9980   0.0034

Ending the PrintAndTraceBackpropagateHiddenToInput() function

Ending the BackpropagateHiddenToInput() function

Calling BackpropagateBiasHiddenWeights() function

Starting the BackpropagateBiasHiddenWeights() function
This backpropagates weight changes onto the bias-to-hidden connection weights

Calling the computeTransferFnctnDeriv() function for outputNode0

Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function

Calling the computeTransferFnctnDeriv() function for outputNode1

Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function

Calling the computeTransferFnctnDeriv() function for hiddenNode0

Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function

Calling the computeTransferFnctnDeriv() function for hiddenNode1

Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function

The newBiasHiddenWeightArray =  [ 0.83369106 -0.5392125 ]

Ending the BackpropagateBiasHiddenWeights() function

The newBiasWeightArray =  [[0.5437770357373254, -0.033221298627139594],
[0.8336910603185961, -0.5392124998204]]

    The weights before backpropagation are:
         Input-to-Hidden                          Hidden-to-Output
    w(0,0) = -0.874   w(1,0) = -0.570         v(0,0) = -0.481   v(1,0) =
0.096
    w(0,1) = 0.022   w(1,1) = 0.995         v(0,1) = -0.942   v(1,1) = -0.250

    The weights after backpropagation are:
         Input-to-Hidden                          Hidden-to-Output
    w(0,0) = -0.874   w(1,0) = -0.568         v(0,0) = -0.451   v(1,0) =
0.128
```

```
    w(0,1) = 0.022   w(1,1) = 0.998         v(0,1) = -0.963   v(1,1) = -0.273

Calling the ComputeSingleFeedforwardPass() function

Starting ComputeSingleFeedforwardPass() function
The inputs transferred in are:
Input0 =  0
Input1 =  1

The initial weights for this neural network are:
     Input-to-Hidden             Hidden-to-Output
w(0,0) = -0.874   w(0,1) = -0.568         v(0,0) = -0.451   v(0,1) = 0.128
w(1,0) = 0.022   w(1,1) = 0.998         v(1,0) = -0.963   v(1,1) = -0.273

  For hiddenActivation0 from input0, input1 =  0 ,  1

computing hiddenActivation0 by calling computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.5654959585571423
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:  0 ,  1
    The summed neuron input is 0.2635
    The activation (applied transfer function) for that neuron is 0.5655
ending computeSingleNeuronActivation() function and returning activation


  For hiddenActivation1 from input0, input1 =  0 ,  1
computing hiddenActivation1 by calling computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.6119023244302164
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:  0 ,  1
    The summed neuron input is 0.4553
    The activation (applied transfer function) for that neuron is 0.6119
ending computeSingleNeuronActivation() function and returning activation


  In computeSingleFeedforwardPass:
  Input node values:  0 ,  1
  The activations for the hidden nodes are:
    Hidden0 = 0.5655 Hidden1 = 0.6119

computing the activation of output node0 by calling
computeSingleNeuronActivation
```

```
starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.5779304340535262
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:
0.5654959585571423 ,  0.6119023244302164
    The summed neuron input is 0.3143
    The activation (applied transfer function) for that neuron is 0.5779
ending computeSingleNeuronActivation() function and returning activation

computing the activation of output node1 by calling
computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.33019875492840167
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:
0.5654959585571423 ,  0.6119023244302164
    The summed neuron input is -0.7073
    The activation (applied transfer function) for that neuron is 0.3302
ending computeSingleNeuronActivation() function and returning activation


  Computing the output neuron activations

  Back in ComputeSingleFeedforwardPass (for hidden-to-output computations)
  The activations for the output nodes are:
    Output0 = 0.5779 Output1 = 0.3302

creating the actualAllNodesOutputList
the hiddenActivation0 =  0.5654959585571423
the hiddenActivation1 =  0.6119023244302164
the outputActivation0 =  0.5779304340535262
the outputActivation1 =  0.33019875492840167

ending ComputeSingleFeedforwardPass() function and returning to main

In main; have just completed a single step of backpropagation with inputs 0 1
    The new SSE (after backpropagation) = 0.287174
    Error(0) = 0.4221,   Error(1) = -0.3302
    SSE(0) =   0.1781,   SSE(1) =   0.1090
  The difference in initial and the resulting SSEs is: 0.0116

   The training has resulted in improving the total SSEs

  The previous SSE Total was 2.1097
  The new SSE Total was 2.0981
    For node 0: Desired Output =  1  New Output = 0.5779
    For node 1: Desired Output =  0  New Output = 0.3302
    Error(0) = 0.4221,   Error(1) = -0.3302
    SSE0(0) =   0.1781,   SSE(1) =   0.1090
  Delta in the SSEs is 0.0116
SSE improvement

Iteration number  0

starting at the top of the while loop again

create random trainingDataList by calling obtainRandomXORTrainingValues()
function
starting obtainRandomXORTrainingValues() function
The trainingDataList =  (0, 1, 1, 0, 1)
ending obtainRandomXORTrainingValues() function and returning to main

Iteration number  1

Randomly selected training data set number  1
The inputs and desired outputs for the X-OR problem from this data set are:
          Input0 =  0             Input1 =  1
 Desired Output0 =  1    Desired Output1 =  0

calling ComputeSingleFeedforwardPass to obtain actual outputs for training
set

Starting ComputeSingleFeedforwardPass() function
The inputs transferred in are:
Input0 =  0
Input1 =  1

The initial weights for this neural network are:
     Input-to-Hidden             Hidden-to-Output
w(0,0) = -0.874   w(0,1) = -0.568         v(0,0) = -0.451   v(0,1) = 0.128
w(1,0) = 0.022   w(1,1) = 0.998         v(1,0) = -0.963   v(1,1) = -0.273

  For hiddenActivation0 from input0, input1 =  0 ,  1

computing hiddenActivation0 by calling computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.5654959585571423
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:  0 ,  1
    The summed neuron input is 0.2635
    The activation (applied transfer function) for that neuron is 0.5655
ending computeSingleNeuronActivation() function and returning activation


  For hiddenActivation1 from input0, input1 =  0 ,  1
computing hiddenActivation1 by calling computeSingleNeuronActivation
```

```
starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.6119023244302164
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:  0 ,  1
    The summed neuron input is 0.4553
    The activation (applied transfer function) for that neuron is 0.6119
ending computeSingleNeuronActivation() function and returning activation


  In computeSingleFeedforwardPass:
  Input node values:  0 ,  1
  The activations for the hidden nodes are:
    Hidden0 = 0.5655 Hidden1 = 0.6119

computing the activation of output node0 by calling
computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.5779304340535262
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:
0.5654959585571423 ,  0.6119023244302164
    The summed neuron input is 0.3143
    The activation (applied transfer function) for that neuron is 0.5779
ending computeSingleNeuronActivation() function and returning activation

computing the activation of output node1 by calling
computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.33019875492840167
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:
0.5654959585571423 ,  0.6119023244302164
    The summed neuron input is -0.7073
    The activation (applied transfer function) for that neuron is 0.3302
ending computeSingleNeuronActivation() function and returning activation


  Computing the output neuron activations

  Back in ComputeSingleFeedforwardPass (for hidden-to-output computations)
  The activations for the output nodes are:
    Output0 = 0.5779 Output1 = 0.3302

creating the actualAllNodesOutputList
the hiddenActivation0 =  0.5654959585571423
the hiddenActivation1 =  0.6119023244302164
the outputActivation0 =  0.5779304340535262
the outputActivation1 =  0.33019875492840167

ending ComputeSingleFeedforwardPass() function and returning to main

In main; have just completed a feedfoward pass with training set inputs 0 1
  The activations (actual outputs) for the two hidden neurons are:
    actualHiddenOutput0 = 0.5655
    actualHiddenOutput1 = 0.6119
  The activations (actual outputs) for the two output neurons are:
    actualOutput0 = 0.5779
    actualOutput1 = 0.3302'
  Initial SSE (before backpropagation) = 0.287174
  Corresponding SSE (from initial SSE determination) = 0.287174

Calling BackpropagateOutputToHidden() function

Starting the BackpropagateOutputToHidden() function
Performs the backpropagation of weight changes onto hidden-to-output weigts

Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function

Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function
Calling the PrintAndTraceBackpropagateOutputToHidden() function

Starting PrintAndTraceBackpropagateOutputToHidden() function

In Print and Trace for Backpropagation: Hidden to Output Weights
  Assuming alpha = 1

  The hidden node activations are:
    Hidden node 0:    0.5655   Hidden node 1:    0.6119

  The output node activations are:
    Output node 0:    0.578    Output node 1:    0.330

  The transfer function derivatives are:
    Deriv-F(0):       0.244    Deriv-F(1):       0.221

The computed values for the deltas are:
                eta  *  error  *   trFncDeriv *   hidden
  deltaVWt00 =   0.50 * 0.4221  * 0.2439   * 0.5655
  deltaVWt01 =   0.50 * -0.3302  * 0.2212   * 0.5655
  deltaVWt10 =   0.50 * 0.4221  * 0.2439   * 0.6119
  deltaVWt11 =   0.50 * -0.3302  * 0.2212   * 0.6119

Values for the hidden-to-output connection weights:
```

```
           Old:     New:      eta*Delta:
[0,0]:   -0.4511   -0.4220   0.0291
[0,1]:   -0.9631   -0.9838   -0.0206
[1,0]:   0.1282   0.1597   0.0315
[1,1]:   -0.2728   -0.2952   -0.0223

Ending PrintAndTraceBackpropagateOutputToHidden() function

Back in the BackpropagateOutputToHidden() function

Ending the BackpropagateOutputToHidden() function

Calling BackpropagateBiasOutputWeights() function

Starting the BackpropagateBiasOutputWeights() function

Calling the computeTransferFnctnDeriv() function for outputNode0

Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function

Calling the computeTransferFnctnDeriv() function for outputNode1

Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function

The newBiasOutputWeightArray =  [ 0.54240802 -0.03222998]

Ending the BackpropagateBiasOutputWeights() function

Calling BackpropagateHiddenToInput() function

Starting the BackpropagateHiddenToInput() function
This Backpropagates the weight changes onto the input-to-hidden weights

Calling the computeTransferFnctnDeriv() function for hiddenNode0

Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function

Calling the computeTransferFnctnDeriv() function for hiddenNode1

Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function

Calling the computeTransferFnctnDeriv() function for outputNode0

Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function

Calling the computeTransferFnctnDeriv() function for outputNode1
```

```
Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function

Calling the PrintAndTraceBackpropagateHiddenToInput() function

Starting PrintAndTraceBackpropagateHiddenToInput() function
Traces the backpropagation of the input-to-hidden weights

In Print and Trace for Backpropagation: Input to Hidden Weights
  Assuming alpha = 1

  The hidden node activations are:
    Hidden node 0:    0.5655   Hidden node 1:    0.6119

  The output node activations are:
    Output node 0:    0.578    Output node 1:    0.330

  The transfer function derivatives at the hidden nodes are:
    Deriv-F(0):       0.246    Deriv-F(1):       0.237

The computed values for the deltas are:
                eta  *  error  *   trFncDeriv *   input    * SumTerm for
given H
  deltaWWt00 =   0.50 * 0.4221  * 0.2457   * 0.0000   * 0.1276
  deltaWWt01 =   0.50 * -0.3302  * 0.2375   * 0.0000   * 0.1442
  deltaWWt10 =   0.50 * 0.4221  * 0.2457   * 1.0000   * 0.1276
  deltaWWt11 =   0.50 * -0.3302  * 0.2375   * 1.0000   * 0.1442

Values for the input-to-hidden connection weights:
           Old:     New:      eta*Delta:
[0,0]:   -0.8743   -0.8743   0.0000
[0,1]:   0.0219   0.0219   0.0000
[1,0]:   -0.5678   -0.5648   0.0029
[1,1]:   0.9980   1.0019   0.0039

Ending the PrintAndTraceBackpropagateHiddenToInput() function

Ending the BackpropagateHiddenToInput() function

Calling BackpropagateBiasHiddenWeights() function

Starting the BackpropagateBiasHiddenWeights() function
This backpropagates weight changes onto the bias-to-hidden connection weights

Calling the computeTransferFnctnDeriv() function for outputNode0

Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function

Calling the computeTransferFnctnDeriv() function for outputNode1

Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function
```

```
Calling the computeTransferFnctnDeriv() function for hiddenNode0

Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function

Calling the computeTransferFnctnDeriv() function for hiddenNode1

Starting the computeTransferFnctnDeriv() function
This computes the derivate of the transfer function
Ending the computeTransferFnctnDeriv() function

The newBiasHiddenWeightArray =  [ 0.83419032 -0.53871162]

Ending the BackpropagateBiasHiddenWeights() function

The newBiasWeightArray =  [[0.5424080209906543, -0.03222998368408356], 
[0.8341903238759356, -0.5387116193191024]]

    The weights before backpropagation are:
         Input-to-Hidden                           Hidden-to-Output
    w(0,0) = -0.874   w(1,0) = -0.568         v(0,0) = -0.451   v(1,0) = 
0.128
    w(0,1) = 0.022   w(1,1) = 0.998        v(0,1) = -0.963   v(1,1) = -0.273

    The weights after backpropagation are:
         Input-to-Hidden                           Hidden-to-Output
    w(0,0) = -0.874   w(1,0) = -0.565         v(0,0) = -0.422   v(1,0) = 
0.160
    w(0,1) = 0.022   w(1,1) = 1.002        v(0,1) = -0.984   v(1,1) = -0.295

Calling the ComputeSingleFeedforwardPass() function

Starting ComputeSingleFeedforwardPass() function
The inputs transferred in are:
Input0 =  0
Input1 =  1

The initial weights for this neural network are:
     Input-to-Hidden             Hidden-to-Output
w(0,0) = -0.874   w(0,1) = -0.565         v(0,0) = -0.422   v(0,1) = 0.160
w(1,0) = 0.022   w(1,1) = 1.002        v(1,0) = -0.984   v(1,1) = -0.295

  For hiddenActivation0 from input0, input1 =  0 ,  1

computing hiddenActivation0 by calling computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.5662170147588406
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:  0 ,  1
    The summed neuron input is 0.2664
    The activation (applied transfer function) for that neuron is 0.5662
ending computeSingleNeuronActivation() function and returning activation


  For hiddenActivation1 from input0, input1 =  0 ,  1
computing hiddenActivation1 by calling computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.6128359345509735
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:  0 ,  1
    The summed neuron input is 0.4592
    The activation (applied transfer function) for that neuron is 0.6128
ending computeSingleNeuronActivation() function and returning activation


  In computeSingleFeedforwardPass:
  Input node values:  0 ,  1
  The activations for the hidden nodes are:
    Hidden0 = 0.5662 Hidden1 = 0.6128

computing the activation of output node0 by calling
computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.586584615580249
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:
0.5662170147588406 ,  0.6128359345509735
    The summed neuron input is 0.3499
    The activation (applied transfer function) for that neuron is 0.5866
ending computeSingleNeuronActivation() function and returning activation

computing the activation of output node1 by calling
computeSingleNeuronActivation

starting computeSingleNeuronActivation() function
compute the activation by calling computeTransferFnctn()

Starting the computeTransferFnction() function
The activation is  0.32440084935542585
Ending the computeTransferFnction() function and returning the activation


  In computeSingleNeuronActivation with input0, input 1 given as:
0.5662170147588406 ,  0.6128359345509735
    The summed neuron input is -0.7336
    The activation (applied transfer function) for that neuron is 0.3244
ending computeSingleNeuronActivation() function and returning activation


  Computing the output neuron activations

  Back in ComputeSingleFeedforwardPass (for hidden-to-output computations)
  The activations for the output nodes are:
    Output0 = 0.5866 Output1 = 0.3244

creating the actualAllNodesOutputList
the hiddenActivation0 =  0.5662170147588406
the hiddenActivation1 =  0.6128359345509735
the outputActivation0 =  0.586584615580249
the outputActivation1 =  0.32440084935542585

ending ComputeSingleFeedforwardPass() function and returning to main

In main; have just completed a single step of backpropagation with inputs 0 1
    The new SSE (after backpropagation) = 0.276148
    Error(0) = 0.4134,   Error(1) = -0.3244
    SSE(0) =   0.1709,   SSE(1) =   0.1052
  The difference in initial and the resulting SSEs is: 0.0110

   The training has resulted in improving the total SSEs

  The previous SSE Total was 2.0981
  The new SSE Total was 2.0871
    For node 0: Desired Output =  1  New Output = 0.5866
    For node 1: Desired Output =  0  New Output = 0.3244
    Error(0) = 0.4134,   Error(1) = -0.3244
    SSE0(0) =   0.1709,   SSE(1) =   0.1052
  Delta in the SSEs is 0.0110
SSE improvement

Iteration number  1

Out of while loop
  Initial Total SSE = 2.1097
  Final Total SSE = 2.0871
  Delta in the SSEs is 0.0226
SSE total improvement

The initial weights for this neural network are:
     Input-to-Hidden                       Hidden-to-Output
w(0,0) = -0.874   w(0,1) = -0.570         v(0,0) = -0.481   v(0,1) = 0.096
w(1,0) = 0.022   w(1,1) = 0.995         v(1,0) = -0.942   v(1,1) = -0.250

The final weights for this neural network are:
     Input-to-Hidden                       Hidden-to-Output
w(0,0) = -0.874   w(0,1) = -0.565         v(0,0) = -0.422   v(0,1) = 0.160
w(1,0) = 0.022   w(1,1) = 1.002         v(1,0) = -0.984   v(1,1) = -0.295

The SSE values at the beginning of training were:
  SSE_Initial[0] = 0.7593
  SSE_Initial[1] = 0.2987
  SSE_Initial[2] = 0.3168
  SSE_Initial[3] = 0.7348

The total of the SSE values at the beginning of training is 2.1097

The SSE values at the end of training were:
  SSE[0] = 0.7593
  SSE[1] = 0.2761
  SSE[2] = 0.3168
  SSE[3] = 0.7348

The total of the SSE values at the end of training is 2.0871

Values for the new outputs compared with previous, given only a partial
backpropagation training:
     Old:     New:     nu*Delta:
Output 0:  Desired =  1 Old actual =  0.5779 Newactual  0.5866
Output 1:  Desired =  0 Old actual =  0.3302 Newactual  0.3244

ending main() function
```