function [Y,Xf,Af] = myNeuralNetworkFunction(X,~,~)

%MYNEURALNETWORKFUNCTION neural network simulation function.

%

% Generated by Neural Network Toolbox function genFunction, 28-Mar-2017 16:22:38.

%

% [Y] = myNeuralNetworkFunction(X,~,~) takes these arguments:

%

% X = 1xTS cell, 1 inputs over TS timsteps

% Each X{1,ts} = 5xQ matrix, input #1 at timestep ts.

%

% and returns:

% Y = 1xTS cell of 1 outputs over TS timesteps.

% Each Y{1,ts} = 1xQ matrix, output #1 at timestep ts.

%

% where Q is number of samples (or series) and TS is the number of timesteps.

%#ok<\*RPMT0>

% ===== NEURAL NETWORK CONSTANTS =====

% Input 1

x1\_step1\_xoffset = [3;68;8;1613;1];

x1\_step1\_gain = [0.4;0.00516795865633075;0.00900900900900901;0.000567054153671676;1];

x1\_step1\_ymin = -1;

% Layer 1

b1 = [1.9740753133669324;2.5484159564326681;-4.0017887417243392;1.9022936791222163;2.0670439483615102;-0.32681398710440368;-0.98340285742764255;0.16278254884190541;0.087665790439591648;-0.55608502247102254;0.25528856856068871;-1.0437280577219259;0.040857427707629623;0.8612602273873774;0.9916187481967027;0.45776138729546961;1.6993117306139758;1.0996385958128752;2.7615780780515586;-2.0893371410206294];

IW1\_1 = [-0.080502667251245164 -4.1879057469534198 -3.1984193319503098 -0.07735698175523828 2.1041709322902995;-1.903124340065468 2.0558688492491584 1.2811956731058789 -0.13579006480593792 1.4337286756265508;1.2479036225405415 1.7142715673378268 -0.86873756423857584 0.41682254428265453 0.75839188224904319;-1.7614853190951245 -0.74710537706283386 0.26699337190007294 -0.85421898981124234 -0.33183207931131831;-1.6626379651882073 0.17913875324456394 -2.3981054102124193 -0.50790465146287489 -1.4243540710012457;0.4721125900635903 -0.54912408969548498 -0.68965588306881931 -1.9580129860406281 -1.4704353628802929;0.55396878421576901 0.64457232538914233 -0.89348159400098559 0.85288117076466885 -1.5764424613076604;0.49431578489228067 0.67927551390565377 -2.0644537046935052 1.695280386668427 1.0982755672024307;1.6035742843139096 -0.75465650448480281 1.4013933278841053 -1.155895903354146 1.6180300038956883;0.20864078635168268 -2.811066870846247 -2.3415946110632722 -0.055326139327297519 -0.31267922446538454;-1.5970024918540506 -1.8116128410245846 0.29543796092361113 -2.2178521888146907 -0.42642252793215074;1.6388987418992342 1.3960585717794958 -1.1660212697172745 2.4500048733181723 1.4838026035732059;-0.17501105139701126 -1.0530878393021521 0.27012778375450092 1.1075471243820127 1.2145061838565641;1.2344496150679127 -0.3489329090913898 -1.0365210142302996 -1.4586550476443012 0.56987656239860596;0.70383577433246991 0.51684347314487833 1.2702676498492247 1.2250490128045894 0.37409445304960215;0.8160972768193685 0.66990996070198061 0.33395191763897891 -2.618833301315064 0.67305036454617195;0.9562376932139296 1.8625980044658705 0.32419175710484022 0.35324466293024781 0.24291665592204381;1.181325988502328 -0.95147230098093005 2.5190605518955014 1.8698419587976673 -1.203233948120044;1.5244140184724624 -1.8342204267584279 -0.96273164225488517 -0.8218314930351861 -0.50888065320682696;-0.12429721826874922 0.31190733898458251 -1.2467068887156221 1.8995383254631106 -1.1281864999376392];

% Layer 2

b2 = -0.47468803200982285;

LW2\_1 = [1.0257240539382124 -0.47891268073115073 -0.35109408834988165 -0.25608840736630645 -0.050903662826292953 0.57875524594582362 -0.10449282154644506 0.93859723541292206 -0.029408389253740036 -1.3543921247651012 0.33708382953263949 -0.13865225441913043 -0.11076716571439101 -0.91719512899613309 0.0090064617511186329 0.23163695001440882 -0.42329542793550246 -0.21284508643251623 0.1690463179721039 -0.96324376949191781];

% Output 1

y1\_step1\_ymin = -1;

y1\_step1\_gain = 0.0531914893617021;

y1\_step1\_xoffset = 9;

% ===== SIMULATION ========

% Format Input Arguments

isCellX = iscell(X);

if ~isCellX, X = {X}; end;

% Dimensions

TS = size(X,2); % timesteps

if ~isempty(X)

Q = size(X{1},2); % samples/series

else

Q = 0;

end

% Allocate Outputs

Y = cell(1,TS);

% Time loop

for ts=1:TS

% Input 1

Xp1 = mapminmax\_apply(X{1,ts},x1\_step1\_gain,x1\_step1\_xoffset,x1\_step1\_ymin);

% Layer 1

a1 = tansig\_apply(repmat(b1,1,Q) + IW1\_1\*Xp1);

% Layer 2

a2 = repmat(b2,1,Q) + LW2\_1\*a1;

% Output 1

Y{1,ts} = mapminmax\_reverse(a2,y1\_step1\_gain,y1\_step1\_xoffset,y1\_step1\_ymin);

end

% Final Delay States

Xf = cell(1,0);

Af = cell(2,0);

% Format Output Arguments

if ~isCellX, Y = cell2mat(Y); end

end

% ===== MODULE FUNCTIONS ========

% Map Minimum and Maximum Input Processing Function

function y = mapminmax\_apply(x,settings\_gain,settings\_xoffset,settings\_ymin)

y = bsxfun(@minus,x,settings\_xoffset);

y = bsxfun(@times,y,settings\_gain);

y = bsxfun(@plus,y,settings\_ymin);

end

% Sigmoid Symmetric Transfer Function

function a = tansig\_apply(n)

a = 2 ./ (1 + exp(-2\*n)) - 1;

end

% Map Minimum and Maximum Output Reverse-Processing Function

function x = mapminmax\_reverse(y,settings\_gain,settings\_xoffset,settings\_ymin)

x = bsxfun(@minus,y,settings\_ymin);

x = bsxfun(@rdivide,x,settings\_gain);

x = bsxfun(@plus,x,settings\_xoffset);

end