function [y1] = myNeuralNetworkFunction(x1)

%MYNEURALNETWORKFUNCTION neural network simulation function.

%

% Generated by Neural Network Toolbox function genFunction, 12-Apr-2020 16:41:11.

%

% [y1] = myNeuralNetworkFunction(x1) takes these arguments:

% x = 1xQ matrix, input #1

% and returns:

% y = 1xQ matrix, output #1

% where Q is the number of samples.

%#ok<\*RPMT0>

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

% Input 1

x1\_step1.xoffset = -0.04606841165;

x1\_step1.gain = 1.91192084353769;

x1\_step1.ymin = -1;

% Layer 1

b1 = [33.839995501107964;3.1407824128633339;5.7502542496827216;-7.6560147520724664;4.7176668304105291;2.2957246525230266;-1.5303109212128174;-4.5913506022583164;7.3918590790052692;-11.050586874868101;13.594398353989792;18.544784696789659];

IW1\_1 = [0.24756110699541978;-26.400921750917643;-19.675549263024813;7.175657154487225;-17.01386201911156;-16.680988151006854;-16.758292187425432;-16.798856090210066;16.254331816785651;-17.351952783978088;17.629008012603482;18.851096959664801];

% Layer 2

b2 = 5.5685719578126491;

LW2\_1 = [-3.9202200599660695 0.31010910420315368 -0.19217485432452863 2.5711467598903051 0.36768783943347189 -0.4559999777162832 0.0016147705657967362 -0.00033221715455616577 0.0027805506339536865 0.002794464575555362 0.016005443258574805 -0.0069356981249386416];

% Output 1

y1\_step1.ymin = -1;

y1\_step1.gain = 0.00909090909090909;

y1\_step1.xoffset = 0;

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

% Dimensions

Q = size(x1,2); % samples

% Input 1

xp1 = mapminmax\_apply(x1,x1\_step1);

% Layer 1

a1 = tansig\_apply(repmat(b1,1,Q) + IW1\_1\*xp1);

% Layer 2

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

% Output 1

y1 = mapminmax\_reverse(a2,y1\_step1);

end

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

% Map Minimum and Maximum Input Processing Function

function y = mapminmax\_apply(x,settings)

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)

x = bsxfun(@minus,y,settings.ymin);

x = bsxfun(@rdivide,x,settings.gain);

x = bsxfun(@plus,x,settings.xoffset);

end