**Лабораторная работа № 7**

«Автоассоциативные сети с узким горлом».

Постановка задачи.

Целью работы является исследование свойств автоассоциативных сетей с узким горлом,

алгоритмов обучения, а также применение сетей для выполнения линейного и нелинейного анализа главных компонент набора данных.

Ход работы.

1. Использовать автоассоциативную сеть с узким горлом для отображения набора данных, выделяя первую главную компоненту данных.

2. Использовать автоассоциативную сеть с узким горлом для аппроксимации кривой на плоскости, выделяя первую нелинейную главную компоненту данных.

3. Применить автоассоциативную сеть с узким горлом для аппроксимации пространственной кривой, выделяя старшие нелинейные главные компоненты данных.

Теория.

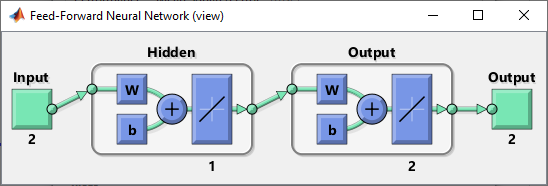
Автоассоциативной памятью — называют память, которая может завершить или исправить образ, но не может ассоциировать полученный образ с другим образом. Данный факт является результатом одноуровневой структуры ассоциативной памяти, в которой вектор появляется на выходе тех же нейронов, на которые поступает входной вектор.

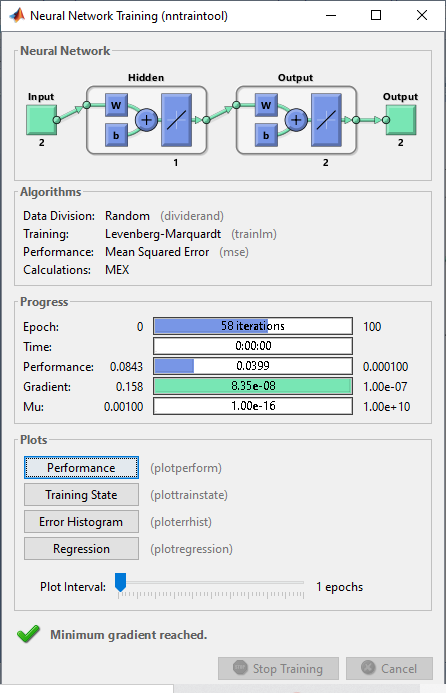
Стоит отметить, что сети с автоассоциативной памятью не устойчивы. Для устойчивых сетей последовательные итерации приводят к всё меньшим изменениям выхода, пока в конце концов выход не становится постоянным. Для многих сетей процесс никогда не заканчивается.

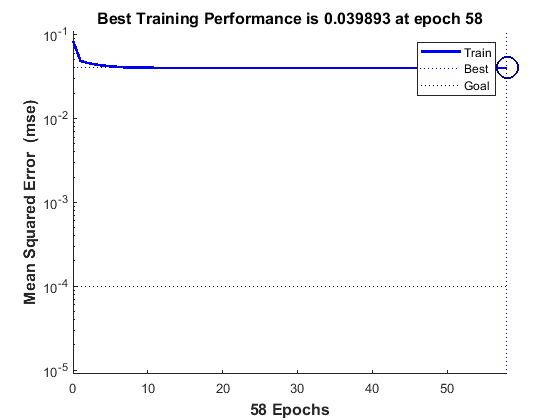
Входные данные и результаты.

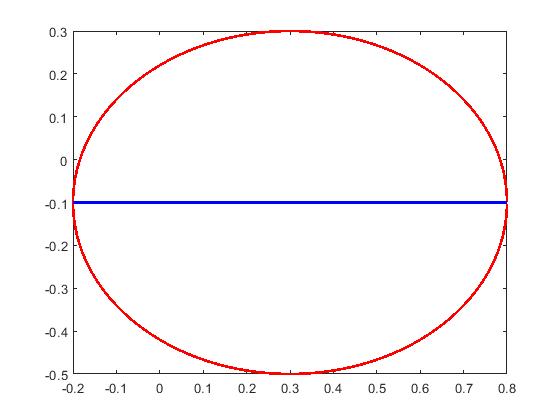
Вариант № 15.

Первое задание

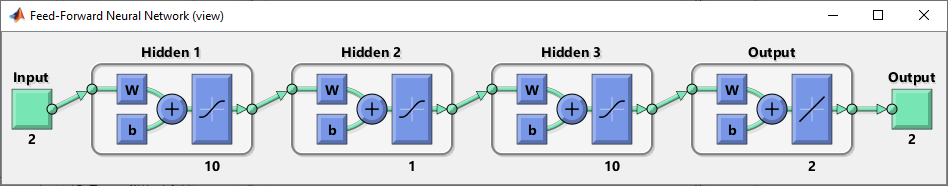


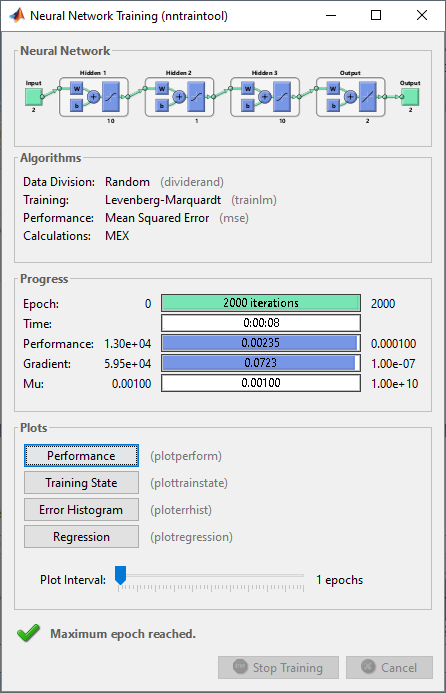


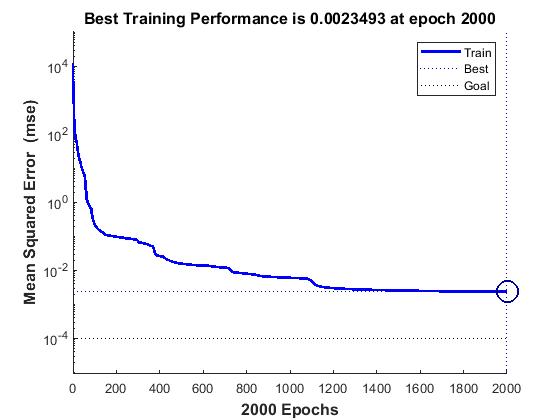


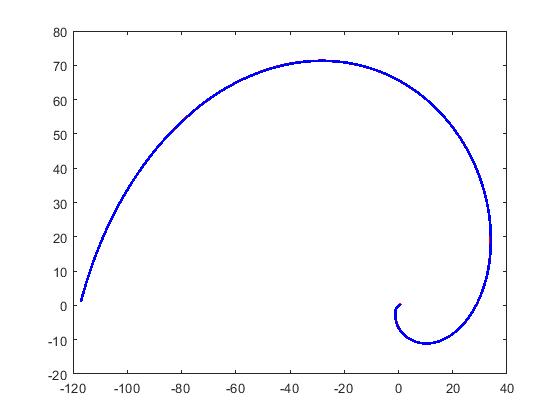


Второе задание

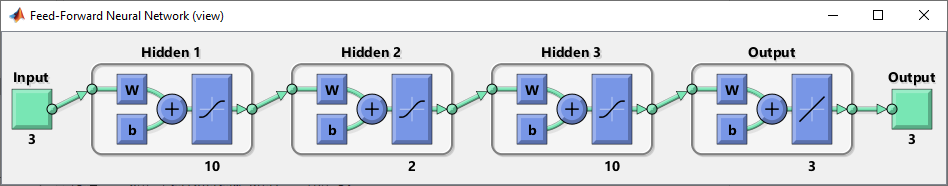


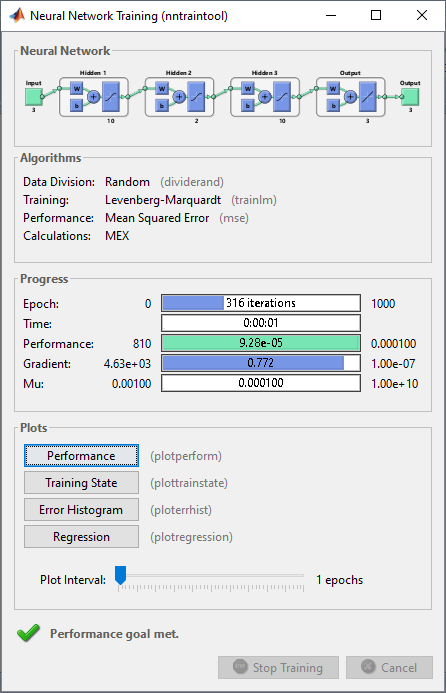


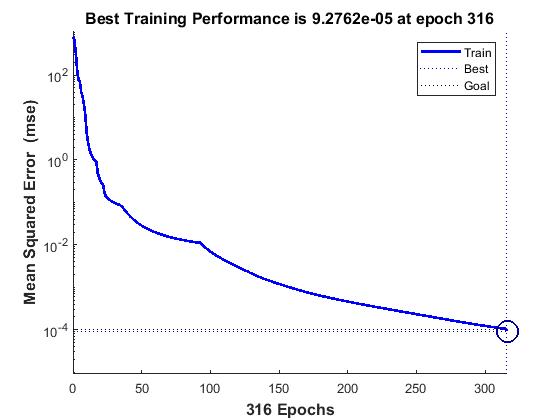


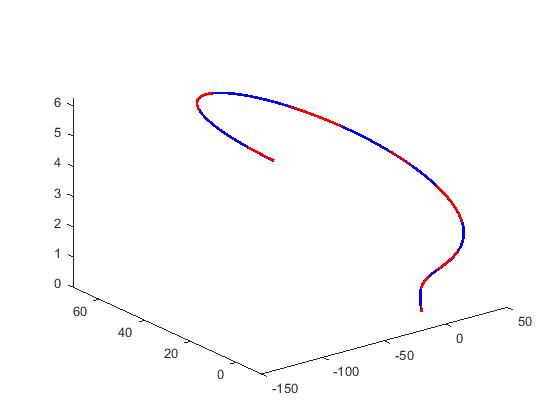


Третее задание









Исходный код.

function [ o ] = ellipse( t, a, b, ShiftX, ShiftY, alpha )

%UNTITLED3 Summary of this function goes here

%   Detailed explanation goes here

x1 = a \* cos(t);

y1 = b \* sin(t);

o = [x1 \* cos(alpha) - y1 \* sin(alpha) + ShiftX;

     x1 \* sin(alpha) + y1 \* cos(alpha) + ShiftY];

end

function [ o ] = secondFunc( alpha )

%UNTITLED3 Summary of this function goes here

%   Detailed explanation goes here

r = 1 - 3 \* (alpha .^ 2);

o = [ r .\* cos(alpha); r .\* sin(alpha)];

end

function [ o ] = thirdFunc( alpha )

%UNTITLED3 Summary of this function goes here

%   Detailed explanation goes here

r = 1 - 3 \* (alpha .^ 2);

o = [ r .\* cos(alpha); r .\* sin(alpha); alpha];

end

clear;

clc;

t = 0 : 0.025 : 2 \* pi;

x = ellipse(t, 0.5, 0.4, 0.3, -0.1, 0);

xseq = con2seq(x);

net = feedforwardnet(1, 'trainlm');

net.layers{1}.transferFcn = 'purelin';

net = configure(net, xseq, xseq);

net = init(net);

view(net);

net.trainParam.epochs = 100;

net.trainParam.goal = 10e-5;

net = train(net, xseq, xseq);

yseq = sim(net, xseq);

y = cell2mat(yseq);

plot(x(1, :), x(2, :), '-r', y(1, :), y(2, :), '-b', 'LineWidth', 2);

clear;

clc;

phi = 0 : 0.025 : 2 \* pi;

x = secondFunc(phi);

xseq = con2seq(x);

net = feedforwardnet([10, 1, 10], 'trainlm');

net.layers{1}.transferFcn = 'tansig';

net.layers{2}.transferFcn = 'tansig';

net.layers{3}.transferFcn = 'tansig';

net.layers{4}.transferFcn = 'purelin';

net = configure(net, xseq, xseq);

net = init(net);

view(net);

net.trainParam.epochs = 2000;

net.trainParam.goal = 10e-5;

net = train(net, xseq, xseq);

yseq = sim(net, xseq);

y = cell2mat(yseq);

plot(x(1, :), x(2, :), '-r', y(1, :), y(2, :), '-b', 'LineWidth', 2);

clear;

clc;

phi = 0 : 0.025 : 2 \* pi;

x = thirdFunc(phi);

xseq = con2seq(x);

net = feedforwardnet([10, 2, 10], 'trainlm');

net.layers{1}.transferFcn = 'tansig';

net.layers{2}.transferFcn = 'tansig';

net.layers{3}.transferFcn = 'tansig';

net.layers{4}.transferFcn = 'purelin';

net = configure(net, xseq, xseq);

net = init(net);

view(net);

net.trainParam.epochs = 1000;

net.trainParam.goal = 10e-5;

net = train(net, xseq, xseq);

yseq = sim(net, xseq);

y = cell2mat(yseq);

plot3(x(1, :), x(2, :), x(3, :), '-r', y(1, :), y(2, :), y(3, :), '-b', 'LineWidth', 2);

Выводы:

Выполнив лабораторную работу, научился применять автоассоциативную сеть с узким горлом для аппроксимации функций и отображения данных, выделяя линейные и нелинейные компоненты данных.

Итеративную автоассоциативную сеть с узким горлом используют по двум причинам:

* Нейронные алгоритмы легко обобщаются на случай нелинейного сжатия информации, когда никаких явных решений уже не существует. Никто не мешает нам заменить линейные нейроны в описанных выше сетях - нелинейными.
* Иногда обучение необходимо проводить в режиме «на лету», т. е. уметь адаптироваться к потоку данных. Примером может служить борьба с нестационарными помехами в каналах связи. Итерационные методы идеально подходят в этой ситуации, когда нет возможности собрать воедино весь набор примеров и произвести необходимые матричные операции над ним.