Министерство образования Республики Беларусь

Учреждение образования

«Брестский Государственный технический университет»

Кафедра ИИТ

**Лабораторная работа №5**

По дисциплине «Логические основы интеллектуальных систем»

Тема: «Моделирование искусственной нейронной сети»

**Выполнил:**

Студент 2 курса

Группы ИИ-21

Кирилович А. А.

**Проверил:**

Скарубо А.О.

Брест 2023

**Цель работы:** Изучение пакета прикладных программ MatLab Neural Network ToolBox на примере моделирования искусственной нейронной сети для распознаваниясимволов латинского алфавита. А также, обзор основных возможностей GUI-интерфейса для NNT.

**Ход работы:**

*нейронную сеть, используя описанную выше методику.*

*полученную*

*Протестировать*

*Обучить нейронную сеть.*

*2.*

*3.*

*предопределенные*

*Neural Network*

*используя пакета*

***MatLab****, функции ToolBox.*

*сети* ***в режиме командной строки***

*Выполнить построение нейронной*

*1.*

Вариант №1

% Импортирование данных алфавита

[alphabet, targets] = prprob;

% Разделение данных на обучающую, тестовую и валидационную выборки

[trainInd, testInd, valInd] = dividerand(26, 0.7, 0.15, 0.15);

% Создание нейронной сети

net = feedforwardnet(10);

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

net.trainFcn = 'trainlm';

net.performFcn = 'mse';

net.trainParam.epochs = 500;

net.trainParam.showWindow = false;

% Обучение нейронной сети на идеальных векторах входа, используя функцию train:

net = train(net, alphabet(:, trainInd), targets(:, trainInd));

% Обучение сети на 10 наборах идеальных и зашумленных векторов входа, используя функцию trainbpx:

for i = 1:10

input = alphabet(:, [trainInd, valInd]) + randn(35, length([trainInd, valInd]))\*0.2;

target = targets(:, [trainInd, valInd]);

net = traingdx(net, input, target);

end

% Снова обучение нейронной сети на идеальных векторах входа, используя функцию train:

net = train(net, alphabet(:, trainInd), targets(:, trainInd));

% Тестирование нейронной сети на тестовой выборке

testOutputs = sim(net, alphabet(:, testInd));

% Вычисление производительности сети на тестовой выборке

% Чем меньше значение производительности, тем лучше качество обучения

testPerformance = perform(net, targets(:, testInd), testOutputs)

% Bизуализация работы нейронной сети:

% 1)

noisyJ = alphabet(:,10) + randn(35,1)\*0.2; % создаем зашумленный образ символа "J"

output = net(noisyJ); % прогнозируем выходные данные с помощью MLP

[maxVal, maxIndex] = max(output); % находим индекс класса с максимальной вероятностью

predictedLetter = char(maxIndex+64) % получаем предсказанный символ алфавита

%2)

J = alphabet(:,10); % создаем образ символа "J"

output = net(J); % прогнозируем выходные данные с помощью MLP

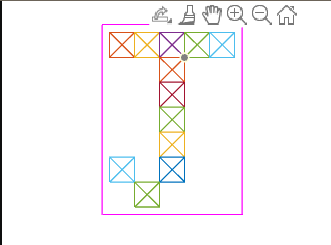
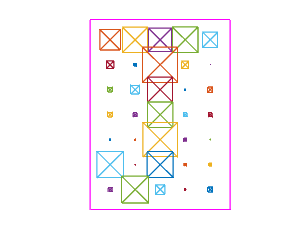
[maxVal, maxIndex] = max(output); % находим индекс класса с максимальной вероятностью

predictedLetter = char(maxIndex+64) % получаем предсказанный символ алфавита

testPerformance = 0.0632  
  
predictedLetter = 'J'  
predictedLetter = 'J'

plotchar(noisyJ)

plotchar(J)



**Вывод:** изучил пакет прикладных программ MatLab Neural Network ToolBox на примере моделирования искусственной нейронной сети для распознавания символов латинского алфавита. А также, сделал обзор основных возможностей GUI-интерфейса для NNT.