

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Таныгин Максим Олегович
Должность: и.о. декана факультета фундаментальной и прикладной информатики
Дата подписания: 21.09.2023 13:00:36
Уникальный программный ключ:
65ab2aa0d384efe840f6ca4c689ad1bc175e411a

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Юго-Западный государственный университет» (ЮЗГУ)

Кафедра программной инженерии



Утверждаю:

Проректор по учебной работе

Локтионова О.Г.

2021г.

СИСТЕМЫ КЛАССИФИКАЦИИ ИНФОРМАТИВНЫХ ПРИЗНАКОВ НА ОСНОВЕ НЕЙРОННОЙ СЕТИ КОХОНЕНА

Методические указания для выполнения лабораторной работы по дисциплине «Теория нейрокомпьютерных систем» для студентов направления подготовки 09.03.04 Программная инженерия

Курск 2021

УДК 004.932

Составитель: Р.А. Томакова

Рецензент

Кандидат технических наук, доцент *Е. И. Аникина*

Системы классификации информативных признаков на основе нейронной сети Кохонена: методические указания для проведения лабораторных работ и выполнения самостоятельной внеаудиторной работы по дисциплине «Теория нейрокомпьютерных систем» для студентов всех форм обучения направления подготовки бакалавров 09.03.04 «Программная инженерия»/ Юго-Зап. гос. ун-т; сост. Р.А. Томакова. Курск, 2021. –15с.

Рассмотрена методика классификации информативных признаков с помощью нейронной сети Кохонена. Приведено исследование механизмов самоорганизации двух основных классов, основанных на ассоциативном правиле Хебба, и механизме конкуренции между нейронами на базе обобщенного правила Кохонена в системе **MATLAB** с использованием графического инструмента пользователя **NNTOOL** из пакета **NeuralNetworkToolbox**.

Предназначены студентам всех форм обучения направления подготовки 09.03.04 Программная инженерия.

Текст печатается в авторской редакции

Подписано в печать . 2021 г. Формат 60×84 1/16.

Усл. печ. л. 0,7 . Уч.- изд. л. 0,6 .Тираж экз. Заказ 882. Бесплатно.

Юго-Западный государственный университет.

305040, г. Курск, ул. 50 лет Октября, 94.

СИСТЕМЫ КЛАССИФИКАЦИИ ОБРАЗОВ НА ОСНОВЕ НЕЙРОННОЙ СЕТИ КОХОНЕНА

Цель работы: изучение модели сети Кохонена и алгоритма обучения без учителя: самоорганизация, основанная на ассоциативном правиле Хебба, и механизм конкуренции между нейронами на базе обобщенного правила Кохонена; создание и исследование модели слоя Кохонена в системе **MATLAB**.

1. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

1.1 Постановка задачи

Самоорганизация нейронных сетей возможна на основе самоорганизующих операций, которые проводятся в различных локальных сегментах сети.

В соответствии с входными сигналами осуществляется активация нейронов, которые вследствие изменения значений синаптических весов адаптируются к поступающим обучающим выборкам. В процессе обучения наблюдается рост значений весов, при этом создается положительная обратная связь: *более мощные возбуждающие импульсы - более высокие значения весов - большая активность нейронов.*

При этом происходит расслоение нейронов на различные группы. Отдельные нейроны или их группы сотрудничают между собой и активизируются, подавляя своей активностью другие нейроны. При этом можно говорить как о сотрудничестве между нейронами внутри группы, так и о конкуренции между нейронами внутри группы и между различными группами.

Среди механизмов самоорганизации можно *выделить два основных класса: самоорганизация, основанная на ассоциативном правиле Хебба, и механизм конкуренции между нейронами на базе обобщенного правила Кохонена.*

Независимо от способов обучения самоорганизующихся сетей *большое значение имеет избыточность обучающих данных, без которой обучение невозможно.*

Широкий спектр обучающих данных, включающий многократные повторения похожих друг на друга выборок, образует "базу знаний" для обучения сети, из которой путем соответствующих сопоставлений выводятся решения по формированию на входе сети определенного вектора.

Обучение нейронной сети возможно осуществлять без учителя в тех случаях, когда в обучающей выборке неизвестны правильные (желаемые) выходные реакции.

Для реализации этого подхода необходимо выполнить два этапа:

1) *этап обучения – разработать методы разбиения образов на классы без учителя;*

2) *этап распознавания – выработать правила отнесения текущего входного образа к некоторому классу.*

Разбиение на классы может быть основано на использовании достаточно общих свойств классифицируемых информативных признаков объектов. В один класс попадают образы похожие в некотором смысле. В этом случае каждому классу в пространстве признаков соответствует обособленная группа точек.

Тогда задача классификации сводится к разделению в многомерном пространстве на части множества точек.

Процесс разбиения множества образов на классы называют кластеризацией.

Необходимо формализовать способ проведения границы между классами.

Существует тип нейронных сетей, основанный на *конкурентном обучении* (competitive learning), в котором нейроны выходного слоя соревнуются за право активации. При этом активным становится обычно один нейрон-победитель в сети нейронов.

Таким образом, в этом случае реализуется принцип **“победитель получает все”**.

1.2 Обучение нейронной сети по правилу Хебба

Д. Хебб, исследуя механизмы функционирования центральной нервной системы, установил, что связь между двумя клетками

усиливается, если обе клетки пробуждаются (становятся активными) в один и тот же момент времени.

Если j -я клетка с выходным сигналом Y_j связана с i -й клеткой, имеющей выходной сигнал Y_i связью с весом W_{ij} , то на силу связи этих клеток влияют значения выходных сигналов Y_i и Y_j .

Подстройка синапсов может проводиться только на основании информации доступной в нейроне, т.е. его состояние Y_j и имеющихся весовых коэффициентов W_{ij} .

По аналогии с известными принципами самоорганизации нервных клеток, был построен алгоритм обучения Хебба, основанный на следующем правиле коррекции весов:

$$W_{ij}(t) = W_{ij}(t-1) + \alpha Y_i^{(n-1)} \cdot Y_j^{(n)}, \quad (1)$$

где $Y_i^{(n-1)}$ – выходное значение i -го нейрона $(n-1)$ -го слоя;

$Y_j^{(n)}$ – выходное значение j -го нейрона (n) -го слоя;

$W_{ij}(t), W_{ij}(t-1)$ – весовые коэффициенты синапса, соединяющего i -й и j -й нейроны на итерациях t и $t-1$ соответственно;

α – коэффициент обучения.

Важно отметить, что при обучении данным методом усиливаются связи между соседними нейронами.

Существует также дифференциальный метод обучения Хебба, в котором коррекция весов производится по формуле:

$$W_{ij}(t) = W_{ij}(t-1) + \alpha [Y_i^{(n-1)}(t-1) - Y_i^{(n-1)}(t-1)] \cdot [Y_j^{(n)}(t) - Y_j^{(n)}(t-1)] \quad (2)$$

Из формулы следует, что наибольшие веса получают синапсы, соединяющие те нейроны, выходы которых наиболее динамично изменяются.

Полный алгоритм обучения имеет вид:

1. На стадии инициализации всем весовым коэффициентам присваиваются небольшие случайные значения.

2. На входы сети подается входной образ и сигнал возбуждения распространяется по всем слоям сети.

3. На основании полученных выходных значений Y_i, Y_j по формулам (1) или (2) производится изменение весовых

коэффициентов.

4. Переход на шаг 2, пока выходы сети не стабилизируются с заданной точностью. При этом на втором шаге цикла попеременно предъявляются все образы обучающего множества.

Вид отклика на каждый класс входных образов неизвестен заранее и представляет собой произвольное сочетание состояний нейронов, обусловленное случайным распределением весов на стадии инициализации. Несмотря на это сеть обобщает похожие образы, относя их к одному классу.

1.3 Обучение нейронной сети по правилу Кохонена

Этот алгоритм позволяет установление значений синаптических коэффициентов на основании предыдущей итерации

$$W_{ij}(t) = W_{ij}(t-1) + \alpha(Y_j^{(n-1)} - W_{ij}(t-1)) \quad (3)$$

Обучение нейронной сети сводится к минимизации разности между входными синапсами нейронов поступающих с выходов нейронов предыдущего слоя с весовыми коэффициентами его синапсов.

В этом случае реализуется принцип *«победитель получает все»*. Такой подход в обучении нейронной сети называют *конкурирующим обучением*.

Приведенная процедура разбивает множество входных образов на кластеры присущие входным данным.

В правильно обученной сети для текущего входного образа X активизируется только один выходной нейрон-победитель, соответствующий кластеру, который попадает в текущий входной образ X .

1.4 Методы определения нейрона-победителя

Существует два метода определения нейрона победителя.

Первый метод основан на вычислении скалярного произведения $\bar{X} \cdot \bar{W}$, т.е. максимальное значение этого произведения определяет нейрон. При этом обычно векторы \bar{X} и \bar{W} нормализуются.

Для каждого выходного вектора вычисляется скалярное произведение

$$Y_0 = \sum_i W_{i0} \cdot X_i = \overline{W_0} \cdot \overline{X}, \quad \forall 0 \neq k, \quad Y_0 \leq Y_k. \quad (4)$$

Затем определяется нейрон-победитель. Далее устанавливаются значения:

для каждого нейрона победителя полагают $Y_k = 1$, а для остальных устанавливаются $Y = 0$.

Как только определяется победитель, его веса корректируются по формуле:

$$W_k(t+1) = \frac{W_k(t) + \alpha(x(t) - W_k(t))}{|W_k(t) + \alpha(x(t) - W_k(t))|}. \quad (5)$$

В этом случае деление на модуль позволяет нормализовать вектора W_k .

Согласно этому выражению вектор весов вращается («подкручивается») в сторону вектора X (рис. 1).

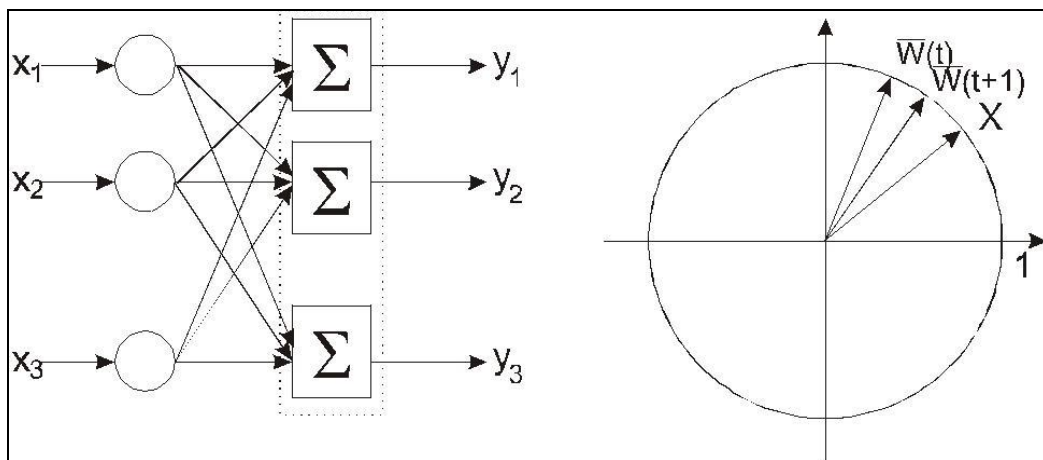


Рисунок 1. Вращение весового вектора в процессе обучения

В процессе обучения, при предъявлении текущего вектора \overline{X} , определяется ближайший к нему весовой вектор нейрона победителя, который «подкручивается» по направлению к \overline{X} .

В результате вектора весов нейронов вращаются по направлению к тем областям, где находится много входных образов (т.е. кластеров), что продемонстрировано на рисунке 2.

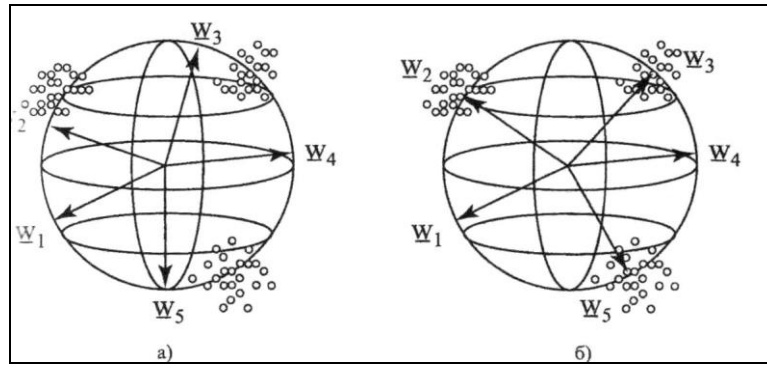


Рисунок 2. Установление весовых векторов нейронов в центры кластеров

Второй метод основан на вычислении евклидова расстояния. При реализации этого метода определяется k -й нейрон-победитель, для которого выполняется условие:

$$\|\bar{W}_k - \bar{X}\| \leq \|\bar{W}_0 - \bar{X}\|, \quad (6)$$

Очевидно, что если вектора \bar{X} и \bar{W} нормализованы, то этот метод дает результаты аналогичные, результатам на основании первого метода.

После определения нейрона-победителя осуществляется корректировка его веса по формуле:

$$\bar{W}_k(t+1) = \bar{W}_k(t) + \alpha(\bar{x}(t) - \bar{W}_k(t)). \quad (7)$$

1.4 Пример выполнения работы

1. В окне “**CommandWindow**” среды **MATLAB** введите команду **nntool**. Откроется окно графического инструмента пользователя **nntool**.
2. Нажмите кнопку “**New**”.
3. В текстовом поле ввода “**Name**” введите имя нейронной сети, например **nnclass**.
4. В выпадающем списке “**NetworkType**” выберите позицию “**Competitive**” (рис. 3).

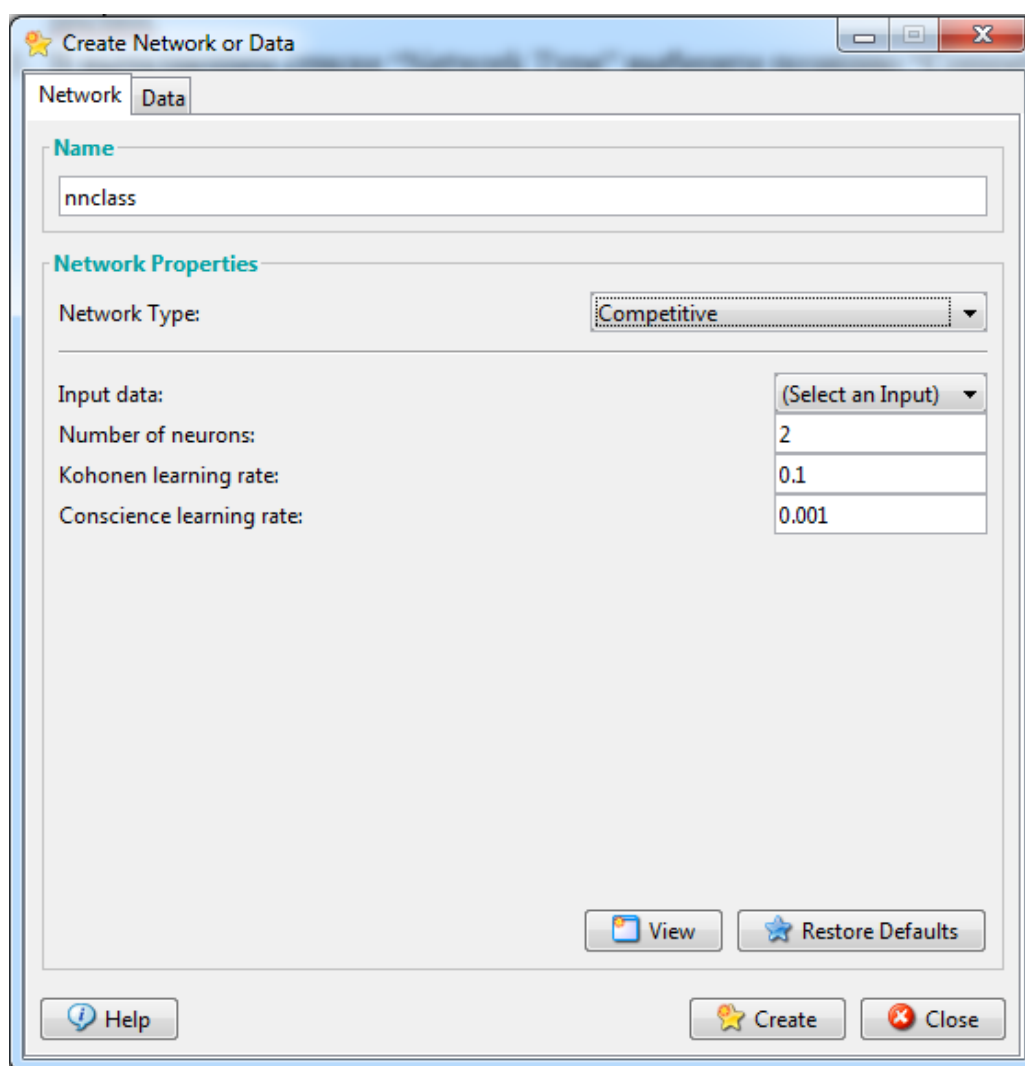


Рисунок 3. Выбор позиции “**Competitive**” в выпадающем списке “**NetworkType**” окна создания нейронной сети графического инструмента пользователя **nntool**.

5. Открыв на вкладку **“Data”**, создайте входные данные P, которые будут использованы для обучения сети, например [1 .8 .1 .9; .2 .9 .1 .8]и входные данные A, которые будут использоваться для симуляции функционирования нейронной сети, например [0 1 -1; 0 1 -1].

6. Откройте вкладку **“Network”**. В выпадающем списке **“Inputdata”** выберите позицию P. В текстовом поле ввода **“Numberofneurons”** введите количество нейронов в скрытом слое равное двум (рис.4).

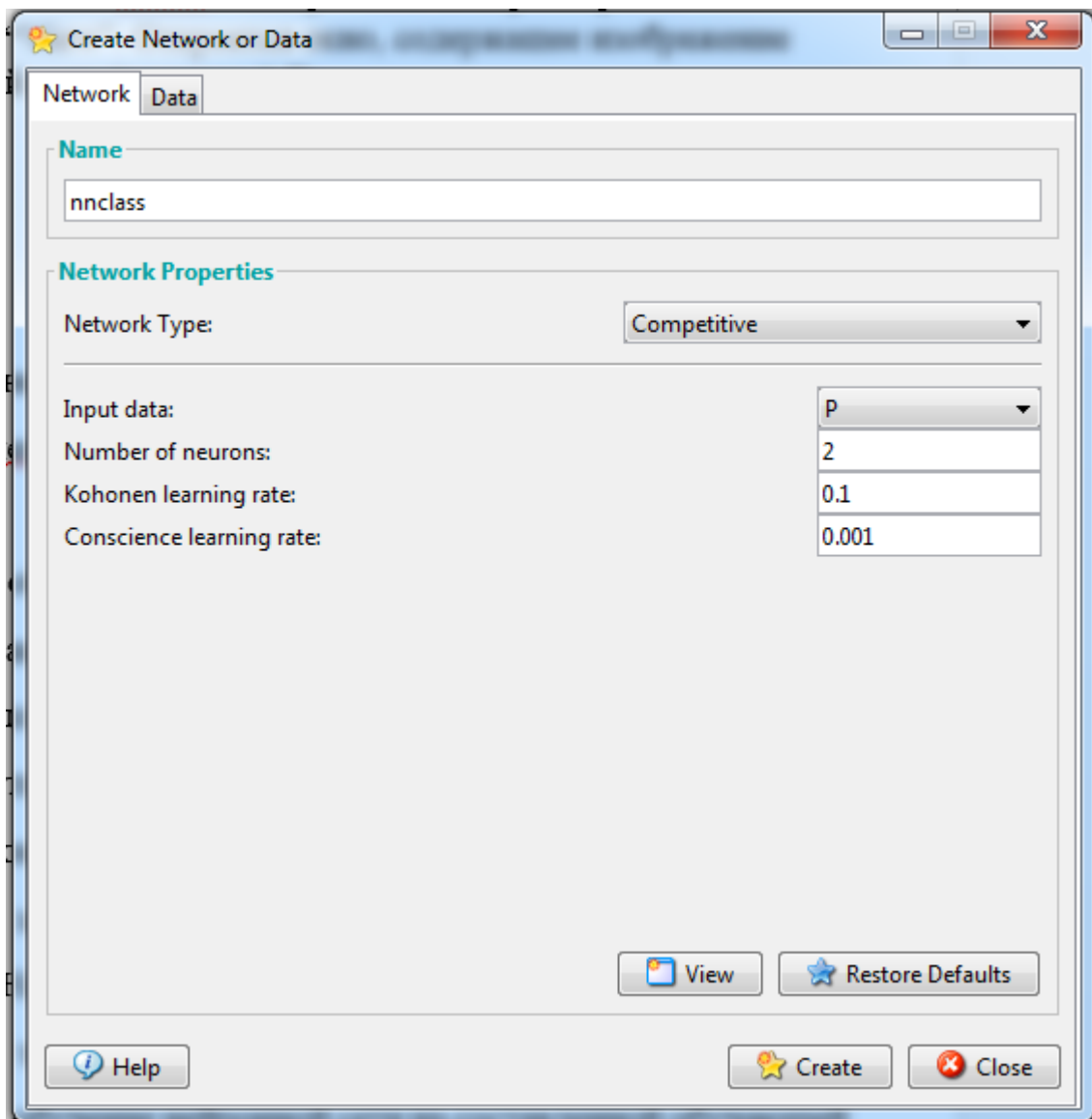


Рисунок 4. Окно создания нейронной сети графического инструмента пользователя **nntool** с настроенными параметрами.

7. Нажмите кнопку “**View**”. Откроется окно, содержащее изображение модели нейронной сети (рис. 5).

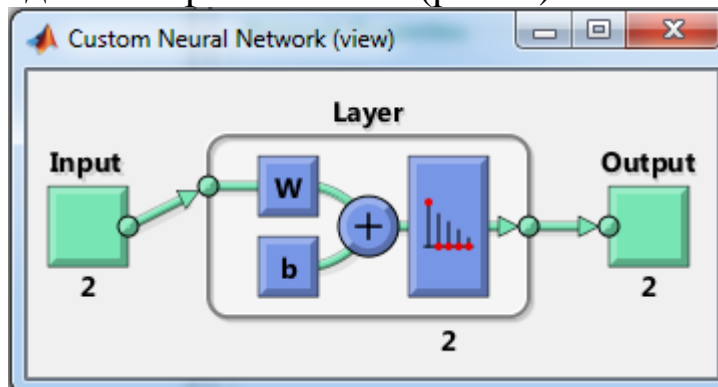


Рисунок 5. Модель нейронной сети

8. Закройте окно с изображением модели нейронной сети и нажмите кнопку “**Create**”. Нажмите кнопку “**Close**”.

9. Дважды кликните мышкой по наименованию нейронной сети в окне графического инструмента пользователя **nntool**.

10. В открывшемся окне откройте вкладку “**Train**”. В выпадающем списке “**Inputs**” выберите позицию Р. Нажмите кнопку “**TrainNetwork**”. Откроется окно обучения нейронной сети (рис.к б).

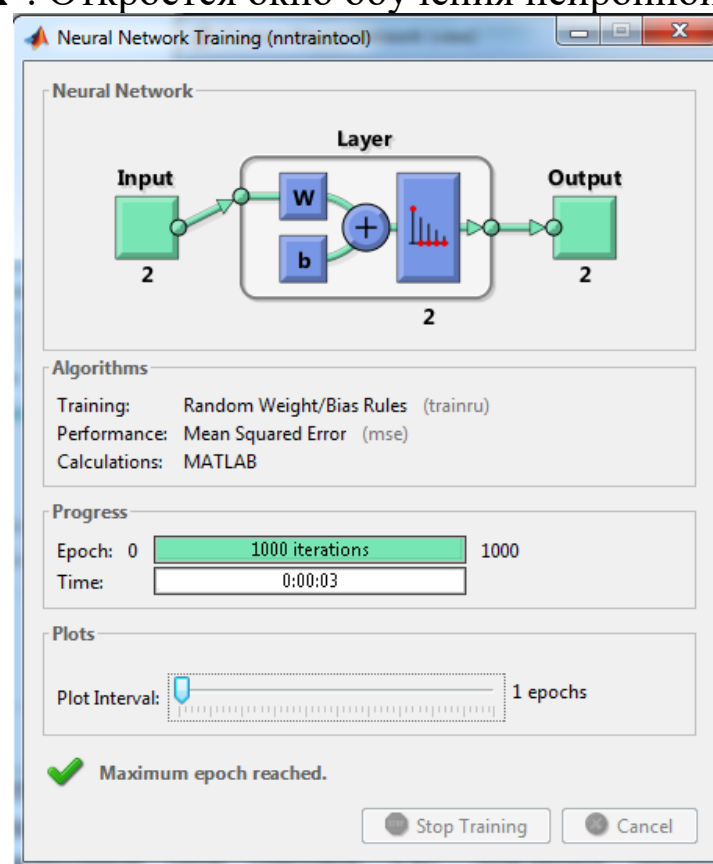


Рисунок 6. Окно обучения нейронной сети

11. Перейдите на вкладку “**Simulate**”.

12. В выпадающем списке “**Inputs**” выберите позицию А. Нажмите кнопку “**Simulate Network**”.

Дождитесь окончания прогона нейронной сети и закройте это окно.

13. Рассмотрите полученный результат.

14. Сделайте выводы.

2. ЗАДАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОГО ВЫПОЛНЕНИЯ

1. Создать нейронную сеть, которая производит классификацию на основе слоя Кохонена и произвести начальную инициализацию весов нейронной сети.

2. Построить обучающую выборку, которая позволяет правильно классифицировать заданную проверочную выборку. Координаты точек обучающей выборки должны быть подобраны геометрически. Для этого первоначально нанести точки проверочной выборки на график, ориентировочно построить вокруг каждой из этих точек несколько точек обучающей выборки, затем записать координаты этих точек в массив обучающей выборки. На каждую из точек проверочной выборки должно приходиться не менее 3 точек обучающей выборки.

3. Произвести обучение нейронной сети на составленной обучающей выборке.

4. Используя обученную нейронную сеть, произвести классификацию проверочного множества.

5. Предоставить отчет. Содержание отчета:

- титульный лист;
- тема и цель работы;
- формальная постановка задачи классификации;
- основные теоретические сведения;
- задание, учитывая свой вариант;
- представить графическое и табличное представление обучающего множества;
- представить графическое и табличное представление проверочных точек;
- выводы.

Таблица 1

Варианты индивидуальных заданий

№ варианта	Количество классов	Координаты точек множества, номер класса, к которому принадлежит данная точка
1	3	$[0;0]$ -1; $[1;1]$ -2; $[-1;-1]$ -3;
2	4	$[2;1]$ -1; $[1;0]$ -2; $[0;-1]$ -3;
3	5	$[0;0]$ -1; $[1;1]$ -1; $[-1;-1]$ -3;
4	6	$[1;0]$ -1; $[-1;1]$ -2; $[-1;-1]$ -2;
5	7	$[0;0]$ -1; $[1;1]$ -2; $[-1;-1]$ -3;
6	8	$[0;0]$ -1; $[1;1]$ -1; $[-1;-1]$ -1;
7	3	$[-1;0]$ -1; $[-1;1]$ -2; $[-1;-1]$ -3;
8	4	$[0;1]$ -1; $[1;-1]$ -2; $[-1;-1]$ -3;
9	5	$[2;0]$ -1; $[1;1]$ -2; $[-1;1]$ -2;
10	6	$[0;0]$ -1; $[1;1]$ -2; $[-1;-1]$ -1;
11	7	$[0;0]$ -1; $[1;1]$ -1; $[-1;-1]$ -2;
12	8	$[0;0]$ -1; $[1;1]$ -2; $[-1;-1]$ -3;
13	3	$[0;0]$ -1; $[1;1]$ -1; $[-1;-1]$ -1;
14	4	$[0;1]$ -1; $[1;-1]$ -2; $[-1;-1]$ -3;

3. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. В чем заключается правило обучения Хебба?
2. Как осуществляется подстройка синапсов?
3. Как осуществляется коррекция весов?
4. На каком принципе основан метод обучения Хебба?
5. Как выполняется дифференциальный метод обучения Хебба?
6. Сформулируйте алгоритм обучения Хебба.
7. Что такое кластеризация информативных признаков?
8. Какие этапы выполняются для реализации обучения без учителя?
9. В чем заключается задача классификации?
10. В чем заключается идея обучения НС по правилу Кохонена?
11. В чем заключается идея конкурентирования нейронов?
12. Какие методы определения нейрона-победителя существуют?
13. Как осуществляется уточнение весов для нейрона-победителя?
14. Привести геометрическую интерпретацию обучения весов по методу Кохонена.

РЕКОМЕНДАТЕЛЬНЫЙ СПИСОК ЛИТЕРАТУРЫ

1. Борисовский, С.А. Нейросетевые модели с иерархическим пространством информативных признаков для сегментации плохоструктурированных изображений / С.А. Борисовский, А.Н. Брежнева, Р.А. Томакова // Биомедицинская радиоэлектроника, 2010, – № 2. – С. 49-53.
2. Дж Ту, Гонсалес, Р. Принципы распознавания образов / Р. Гонсалес, Дж Ту – М.: Мир, 1978. – 411 с.
3. Корневский, Н.А. Нейронные сети с макрослоями для классификации и прогнозирования патологий сетчатки глаза / Н.А. Корневский, Р.А. Томакова, С.П. Серегин, А.Ф. Рыбочкин // Медицинская техника, 2013. – № 4. – С. 16-18.
4. Томакова Р.А. Теоретические основы и методы обработки и анализа микроскопических изображений биоматериалов: монография

/ Р.А. Томакова, С.Г. Емельянов, С.А. Филист. – Курск, Юго-Зап. гос. ун-т, 2011. – 202с.

5. Томакова Р.А. Интеллектуальные технологии сегментации и классификации биомедицинских изображений: монография / Р.А. Томакова, С.Г. Емельянов, С.А. Филист. – Курск, Юго-Зап. гос. ун-т, 2012. – 222с.

6. Томакова, Р.А. Структурно-функциональные решения нечетких нейронных сетей для интеллектуальных систем анализа разнотипных признаков/ Р.А. Томакова, С.А. Филист, В.В. Жилин, С.А. Горбатенко//Фундаментальные и прикладные проблемы техники и технологии, 2011. – № 1. – С. 85-91.

7. Томакова, Р.А. Универсальные сетевые модели для задач классификации биомедицинских данных / Р.А. Томакова, С.А. Филист, Яа Зар До // Известия Юго-Западного государственного университета, 2012. – № 4-2(43). – С.44-50.

8. Томакова, Р.А. Метод обработки и анализа сложноструктурируемых изображений на основе встроенных функций среды MATLAB / Р.А. Томакова, С.А. Филист //Вестник Читинского государственного университета. – 2012. – № 1 (80). – С. 3-9.

9. Томакова, Р.А. Нейросетевые модели принятия решений для диагностики заболеваний легких на основе анализа флюорограмм грудной клетки / Р.А. Томакова, М.В. Дюдин, М.В. Томаков // Биомедицинская радиоэлектроника. – 2014. – №9. – С. 12-15.

10. Хайкин, С. Нейронные сети: полный курс / С. Хайкин. – М.: ООО «И.Д. Вильямс», 2006. – 1104с.