МИНОБРНАУКИ РОССИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»

Кафедра физики открытых систем

Применение нейронных сетей в исследовательских задачах классификации

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

Студента 4 курса 431 группы

направления 09.03.02 «Информационные системы и технологии»

факультета нелинейных процессов

Алимова Никиты Владимировича

Научный руководитель
д.ф.-м.н., профессор _______________ А.Н. Павлов

Заведующий кафедрой
д.ф.-м.н., профессор _______________ А.А. Короновский

Саратов 2019 год

Оглавление

Теоретическая часть

Введение

Нейронные сети успешно применяются в задачах классификации, прогнозирования и управления. При решении сложных многомерных задач нейроные сети незаменимы, поскольку их характеристики позволяют использовать нелинейное моделирование.

Поскольку нейронные сети нелинейны, они представляют мощный инструмент моделирования, который поможет воспроизвести сложные зависимости. Для линейного моделирования изобретено много процедур оптимизации, поэтому линейное моделирование достаточно часто использовалось на протяжении многих лет. Но в областях, в которых линейные методы не могут решать задачи с необходимой точностью, применяются нейросетевые методы.

Нейронные сети изучают на примерах, когда пользователь подбирает репрезентативную выборку, запускает алгоритм обучения, который автоматически воспринимает структуру данных. Пользователю необходимо знать какие данные использовать, как их подготовить, как выбирать архитектуру сети и интерпретировать результаты. Но если сравнивать уровень знаний, необходимый для применения основных и традиционных методов статистики, то для применения нейронных сетей необходимо знать гораздо меньше.

Краткие теоретические сведения.

Идея нейрона является обобщающей идеей: она синтезирует биологические представления с математическими образами. Нейрон – специальная нервная клетка, способная воспринимать, преобразовывать и распространять сигналы [1]. Искусственная нейронная сеть – математическая модель, смоделированная программно или аппаратно, построенная по принципу организации и функционирования биологических нейронных сетей – сетей нервных клеток живого организма. Данное определение появилось при изучении и попытке проектирования процессов, проходящих в мозге.

Общая модель нейрона состоит в следующем: нейрон имеет несколько каналов ввода информации – дендриты и канал вывода информации – аксон. Аксон нейрона соединен с дендритами других нейронов с помощью синапсов. При возбуждении нецрон посылает сигнал своему аксону. Через синапсы сингнал передается другим нейронам, которые, в свою очередь, могут возбуждаться или переходить в состояние торможения.

Нейрон возбуждается, если уровень сигналов, пришедших в него, в сумме привышает порог возбуждения или активации. Интенсивность сигнала, получаемого нейроном, зависит от активности синапсов. Графически модель нейрона можно представить в виде, показанном на рисунке 1. Из него видно, что на вход нейрона поступают сигналы x, каждый из которых умножается на вес w, для каждого сигнала имеется собственный вес. Затем производится сложение преобразованных сигналов и добавляется порог (блок сумматор). Результат преобразуется с помощью функции активации и подается на выход нейрона.

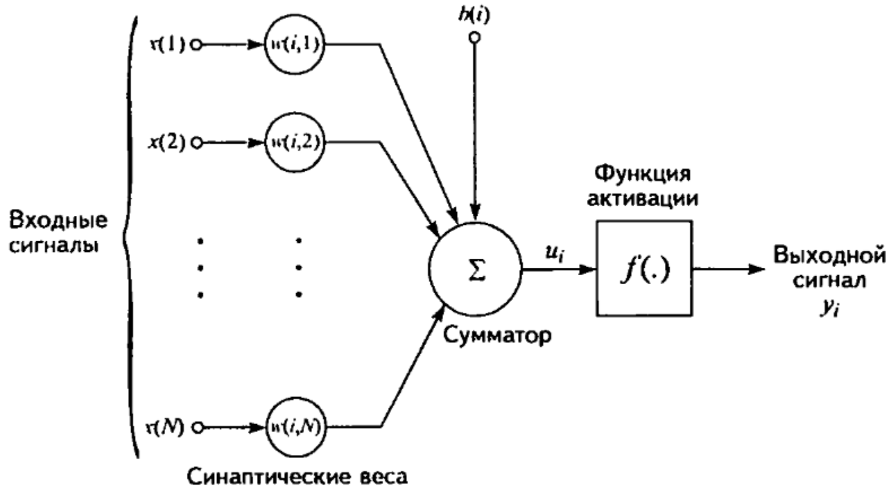

Рис. 1 Модель нейрона H1 [1]

Нейрон получает сигналы через несколько входных каналов (они указаны на схеме слева).

Каждый сигнал проходит через соединение – синапс, имеющий вес w(I, j), который соответствует синаптической активности нейрона.

Коэффициенты w(I, j) называются весами синаптической связи, положительные значения которых соответствуют возбуждающим синапсам, отрицательные значения – тормозящим синапсам. Если w(I, j) = 0, то говорят, что связь между нейроном I и нейроном j отсутствует.

Величина b(i) называется пороговым значением (порогом активации).

Полученный нейроном сигнал преобразуется с помощью нелинейной функции активации или передоточной функции f в выходной сигнал

yi=f(ui) (2),

I – номер рассматриваемого нейрона в сети, j – номер синаптической связи.

Будем называть данную модель нейрона H1. Она является одной из первых моделей нейрона, предложенная МакКаллоком и Питсом в 1943г.

Есть иная форма записи формулы (1), в такой форме добавляется нулевой синапс с единичным сигналом:

Ui= sum j=0, n w(I,j)x(j) (3)

В такой модели для нелвеого веса полагаем w(I,0)=b(i) по определению. Таким образом вес нулевого синапса в модели (3) становится равен порогу модели (2).

Такую модель будем называть Н2. Графическое представление данной модели показано на рис. 2

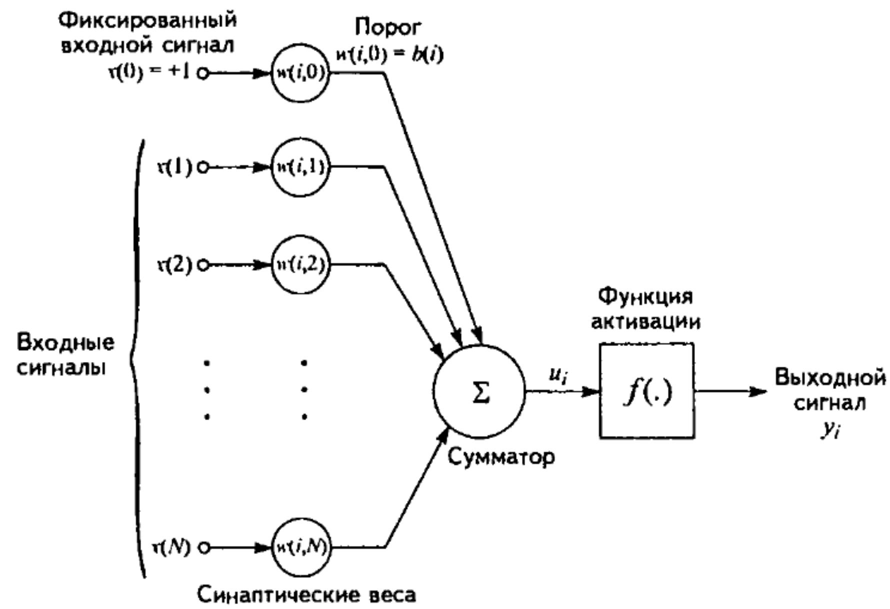

Рис. 2 Модель нейрона Н2 [1]

Если сравнивать модели Н1 и Н2, то можно проследить оновное отличие моделей – добавление входного сигнала с весом, равным порогу. Модели Н1 и Н2 являются эквивалентными. В данной работе будет рассматриваться модель Н1, в которой указан порог возбуждения в явном виде b(i).

Рассмотрены функции активации нейронов.

Всего существует несколько типов функций активаций:

1. Пороговая функция (threshold function) или функция единичного скачка, которая определяется равенствами

$F(u)=1,\ u=>0.\ F(u)=0,u<1$ (4)

2. Сигмоидальная функция (sigmoid function)

   $F(u)=1/1+e^{\wedge}-bu,\ b>0$ (5)

3. Кусочно-линейная функция (pice-wise-function)

   $F(u)=1,\ u>0.5;\ f(u)=|u|,\ |u|<0.5;\ f(u)=0,\ u<0.5$ (6)

4. Функция знак (сигнум)

   $F(u)=-1,u<0;\ f(u)=1,u>0$ (6)

Каждый нейрон – элемент сети. Он описывается своим набором пороговых значений и весов, а также функцией активации f.

Входной слой нейронов служит для задания значений входных переменных, выходной слов – для вывода результатов. Последовательность слоев нейронов и их соединений называется архитектурой сети. Когда сеть задается, это значит, что задается архитектура сети и параметры нейронов.

Когда сеть работает, на вход подается сигнал (значения входных данных), затем возбуждаются нейроны первого промежуточного слоя, второго промежуточного слоя и тд.; в результате чего преобразованные сигнал поступает на выходной слой.

Преобразование сигнала нейроном сети проводится следующим естественным образом:

- Из взвешенной суммы выходов элементов предыдущего слоя вычитается пороговое значение;
- С помощью передаточной функции полученные значения преобразуются;
- В результате получается выходной сигнал нейрона, который поступает на вход следующего нейрона или подается на выход сети, если нейрон является конечным.

Обучение сети – подгонка свободных параметров с целью адаптации к внешним воздействиям.

Рассмотрим обучение с учителем.

Алгоритму обучения с учителем передается набор данных, содержащий признаки. Каждый пример снабжается меткой (классом). Например, в наборе данных Ирисы Фишера каждый пример (ирис) принадлежит своему виду. Алгоритм должен проанализировать набор Ирисы Фишера и научиться определять класс ириса по результатам измерений.

Обучение с учителем сводится к наблюдению нескольких примеров случайного вектора x, и ассоциированного с ним значения вектора y. Затем следует попытка предсказать y по x, зачастую в виде оценки p(y | x).

Подразумевается, что метка y предоставлена неким учителем, который объясняет системе машинного обучения как обрабатывать информацию. Отсюда происходит название «Обучение с учителем» или «Supervised learning».

Персептрон Розенблатта.

Рассмотрим модель H1.

Будем рассматривать классификацию с учителем.

Допустим, на вход нейрона поступает сигнал, принадлежащий либо классу A, либо классу B. Можно ли обучить нейронную сеть различать к какому именно классу принадлежит входной сигнал?

Работу алгоритма можно описать следующим образом: допустим, имеется сигнал +1 (плюс один) из класса A и сигналы -1 (минус 1) из класса B. Сигнал +1 может обозначать, например, напряжение, а сигнал -1 – расслабление.

Допустим, имеется обучающая выборка с известными резулттатами классификации, которая позволяет контролировать процесс обучения.

Вначале рассмотрим случай одномерных сигналов.

В стандартном нейроне существует один вход и порог. Следовательно, имеется два свободных параметра: вес w и порог активации b. Функцией активации в персептроне Розенблатта является стандартная функция сигнум (знак).

Работу алгоритм начинает с нулевых значений параметров.

- На шаге n поступает очередной сигнал из обучающей выборки.
- На шаге n вычисляется состояние нейрона w(n)x(n)+b(n).
- Пусть w(n)x(n)+b(n) >= 0, но объект отнесен к отрицательному классу B. Производится корректировака параметров по правилу: w(n+1)=w(n)-x(n), b(n+1)=b(n)-x(n).
- Пусть w(n)x(n)+b(n) < 0, но объект отнесен к классу A, производится корректировка параметров по правилу: w(n+1) = w(n)+x(n), b(n+1)=b(n)+x(n).
- В том случае, если классификация произведена правильно, корректировка не производится.

Данный алгоритм сходится за несколько шагов.

Также, алгоритм распространяется и на двумерные сигналы. Тогда в модели имеется 3 свободных параметра: вес w1, вес w2 и порог активации b.

Поскольку состояние такого нейрона описывается линейной функцией w1x1+w2x1+b, то можно правильно классифицировать лишь множества, разделенные прямой линией (лежащие по разные стороны от разделяющей линии). Такие множества называются линейно разделимыми (рис. 5).

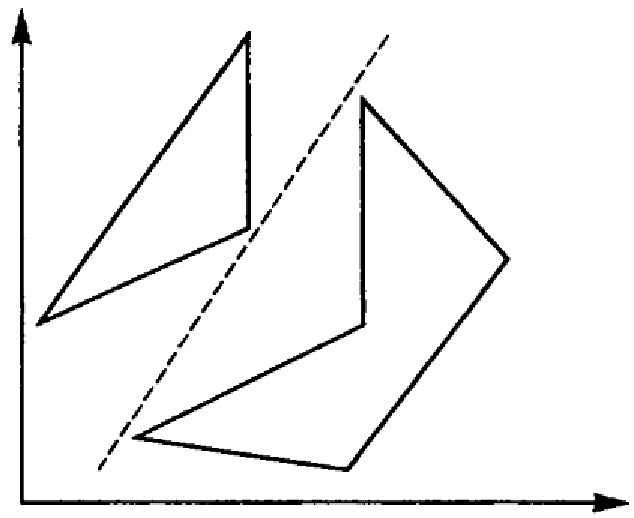

Рис. 5 Линейно разделимые множества

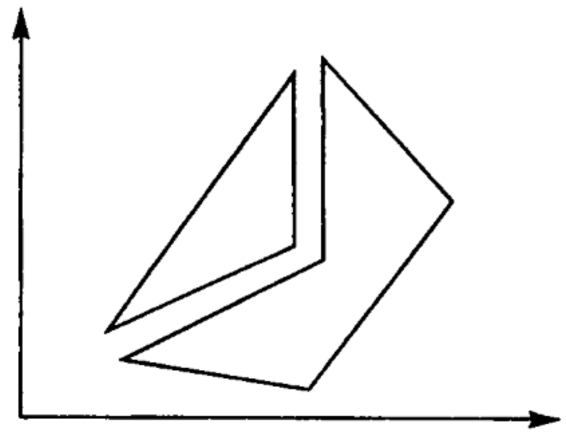

Рис. 6 Линейно неразделимые множества

В случае трех входных сигналов нужна плоскость, разделяющая два множества в пространстве. Для n-мерного пространства необходим (n-1) мерный разделитель.

Естественное обобщение перспетрона Розенблатта приводит к появлению понятия многослойного персептрона.

Многослойный персептрон имеет входной слой, выходной слой и слой (слои) скрытых нейронов, которые расположены между входным и выходным слоем.

Сигнал распространяется от входного слоя к выходному, проходя через скрытые слои нейронов. Скрытые слои играют ключевую роль в обучении нейронов.

Многослойный персептрон отличается от однослойного функцией активации и высокой степенью связности нейронов (когда каждый нейрон скрытого слоя связан с нейронами предыдущего слоя).

Функция активации в многослойном персептроне более гладкая, а не скачкообразная.

На примерах ниже можно рассмотреть архитектуры персептронов, отличающихся количеством скрытых слоёв.

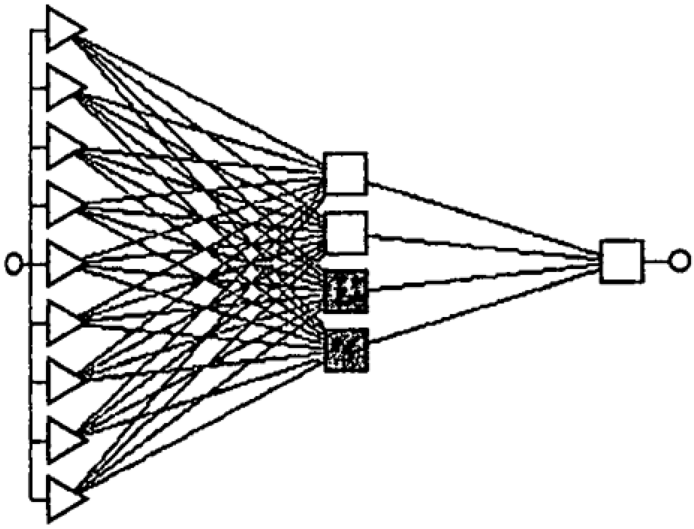

Рис. № 7. Архитектура персептрона с одним скрытым слоем

Обучение персептрона с такой архитектурой происходит при помощи алгоритма обратного распространения ошибки. Данный алгоритм является модификацией алгоритма градиентного спуска, а термин «обратное распространение» связан с распространением вектора ошибки в обратном направлении от направления распространения входного сигнала.

Практическая часть

Пример: «Задача исключаю щего ИЛИ»

Задача исключающего или представляет интерес с точки зрения обучения нейронной сети решению нелинейной задачи. Существует набор данных, где есть 2 бинарные переменные на входе – Вход_1 и Вход_2, а также бинарная выходная переменная XOR. Результаты XOR истины, если Вход_1 и Вход_2 отличаются, иначе – ложны.

	Вход_1	Вход_2	XOR
1	0	0	0
2	0	1	1
3	1	0	1
4	1	1	0

Рисунок 8. Таблица истинности XOR

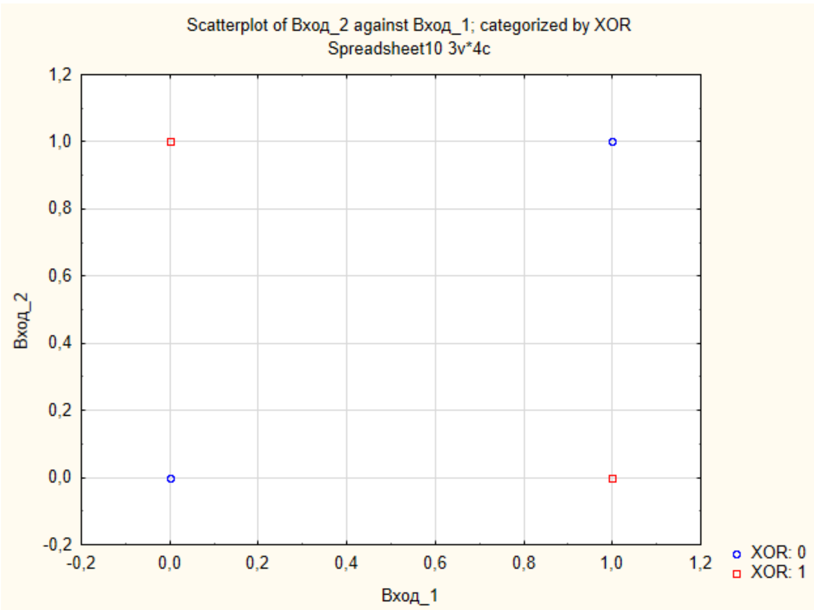

Рисунок 9. Отображение результатов XOR переменной на плоскости

На данном графике по оси абсцисс отложены значения Вход_1, а по оси ординат отложены значения Вход_2. Если рассмотреть график, можно увидеть, что множество линейно неразделимо, то есть нельзя провести линию так, чтобы по одну сторону линии оказались случаи истинные, а по другую сторону ложные. Данную задачу решать линейными методами нецелесообразно. Это свойство представляет интерес для обучения нейронной сети, так как линейными методами решить её невозможно. Для решения таких задач был разработан многослойный персептрон. Для корректного решения данной задачи необходимо, чтобы на скрытом слое было 2 блока (иначе – 2 скрытых слоя), как указано в литературе [2]. Данный пример является простейшей задачей из большого класса линейно неотделимых задач.

Если рассмотреть статистику обучения с одним скрытым слоем (одним блоком скрытого слоя), то можно заметить, что эффективность обучения приближается к значению 0.5. Погрешность распознавания будет слишком высокой. Данный пример показателен, так как результаты обучения совпадают с результатами обучения, представленными в книге [2].

Index	Net. name	Training perf.	Test
10	MLP 4-1-1	0,577350	
11	MLP 4-1-1	0,577350	
12	MLP 4-1-1	0,577350	
13	MLP 4-1-1	0,577350	

Рисунок 10. Эффективность обучения нейронных сетей с одним скрытым блоком.

У данной сети алгоритм обучения BFGS, функция ошибки – сумма квадратов, активация скрытого слоя – гиперболический тангенс.

В следующем эксперименте данным нейронным сетям необходимо было распознать значения из «новой» таблицы истинности, заданной в другом порядке.

Cases	10.XOR_(t)	11.XOR_(t)	12.XOR_(t)	13.XOR_(t)	Вход_1	Вход_2
1	-0,000000	-0,000000	0,666667	0,666668	1	1
2	0,666667	0,666667	0,666667	0,666668	1	0
3	0,666667	0,666667	0,666667	0,666668	0	1
4	0,666667	0,666667	0,000001	0,000008	0	0
5	0,666667	0,666667	0,666667	0,666668	1	0
6	0,666667	0,666667	0,000001	0,000008	0	0
7	-0,000000	-0,000000	0,666667	0,666668	1	1
8	0,666667	0,666667	0,666667	0,666668	0	1

Рисунок 11. Результат распознавания нейронных сетей значений «новой» таблицы истинности, поданной на вход данным сетям.

На рисунке № 11 правильные распознанные значения отмечены красным. Остальные распознанные числа либо не являются точными, либо являются ложными. Как можно заметить из результатов эксперимента, погрешность распознавания и предсказания корректного значения действительно высока. Результат обучения не имеет смысла применять практически. Для этого создадим нейронную сеть с двумя скрытыми слоями.

Нейронные сети с двумя скрытыми слоями обучались решать данную задачу распознавания. Архитектура сетей следующая: алгоритм обучения – BSFG, функция ошибки – сумма квадратов, функция активации – гиперболический тангенс.

Результаты обучения наглядно продемонстрированы на рисунке 12 .

Index	Net. name	Training perf.	Test
14	MLP 4-2-1	0,999999	
16	MLP 4-2-1	1,000000	

Рисунок № 12. Эффективность обучения нейронных сетей с двумя скрытыми слоями.

Значения эффективности корректного распознавания примерно равна единице. Это означает, что данные сети обучились корректно, либо с незначительной погрешностью. Для распознавания корректных значений данным сетям были поданы на вход те же данные, что и предыдущей.

Результаты предсказаний данных сетей наглядно продемонстрированы на рисунке № 12.

Cases	14.XOR_(t)	16.XOR_(t)	Вход_1	Вход_2
1	0,000400	0,000488	1	1
2	0,999112	1,000317	1	0
3	1,000087	0,999770	0	1
4	-0,000673	0,000429	0	0
5	0,000400	0,000488	1	1
6	0,999112	1,000317	1	0
7	1,000087	0,999770	0	1
8	-0,000673	0,000429	0	0

Рисунок № 13. Результат распознавания нейронных сетей с двумя блоками на скрытом слое.

Из рисунка № заметно, что все значения распознанны корректно, либо с незначительной погрешностью, которой можно принебречь. Значения получились корректные, такие как и описано в [2].

Ирисы Фишера

Теперь рассмотрим классификацию на задаче «Ирисы Фишера».

Данную задачу классификации впервые предложил Р. Фишер в 1936 году. Этот набор данных часто используется для изучения алгоритмов классификации, распознавания образов, статистического анализа и т.п.

Имеется 3 типа ирисов: Setosa, Virginic, Versicol по 50 представителей каждого вида. Для каждого типа ирисов имеется длина и ширина чашелистика, длина и ширина лепестка. Необходимо обучить нейронную сеть правильно классифицировать новый «неизвестный» тип ирисов по данным из этого набора. Данные поделены на 2 подвыборки: тренировочная и тестовая.

В этой задаче есть особенность: один тип ирисов – Setosa, линейно отделим от других. Два другие типа – Virginic и Versicol имеют пересечения. То есть некоторые экземпляры выбиваются из общего набора данных, значит, есть возможность неправильной классификации, отнесения к другому классу.

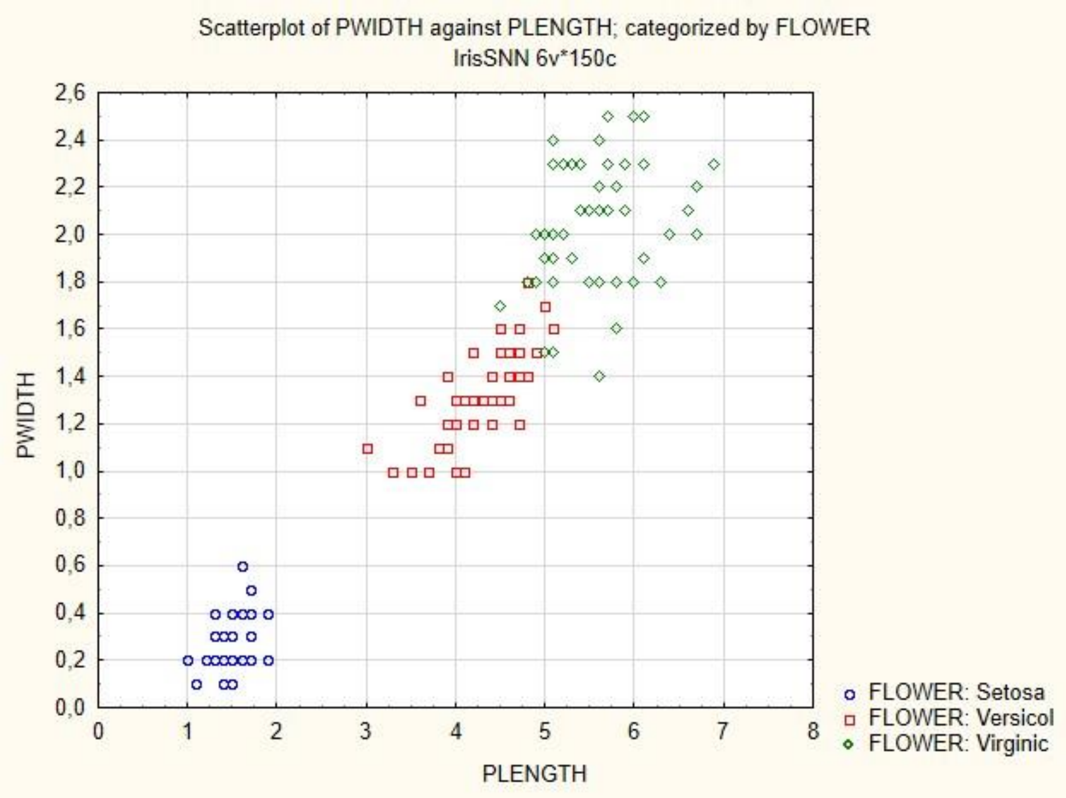

Рисунок № 14. Распределение типов ириса по длине и ширине лепестка.

Нейронная сеть обучалась и тестировалась на соответствующих подвыборках.

На результатах обучения видно, что нейронная сеть MLP 4-13-3 показала наилучший результат. Все наиболее успешные модели представлены на графике ниже.

Summary of active networks (IrisSNN)

Index	Net. name	Training perf.	Test perf.	Validation perf.	Training algorithm	Error function	Hidden activation
1	MLP 4-13-3	100,0000	97,14286		BFGS 24	SOS	Exponential
2	MLP 4-9-3	98,7500	97,14286		BFGS 27	SOS	Tanh
3	MLP 4-19-3	97,5000	97,14286		BFGS 11	Entropy	Identity
4	MLP 4-8-3	98,7500	97,14286		BFGS 34	SOS	Tanh

Пример обучения: «Digit»

На примере данной задачи можно рассмотреть каким образом нейронные сети возможно применить в распознавании образов. Существует набор данных, имитирующий «цифры», отображающеся экране неисправного калькулятора. Классы зависимостей переменной Digit соответствовали цифрам 0-9, ввод которых осуществляется с помощью клавиатуры калькулятора.

В задаче имеется семь предикторов, каждый отвечает за свою линию. Из этих линий образуется цифра.

Каждый предиктор имеет два состояния: 0 – отсутствует, 1 – присутствует. Состояние показывает, высвечивалось на экране соответствующая предиктору линия или не высвечивалась.

Описание предикторных переменных:

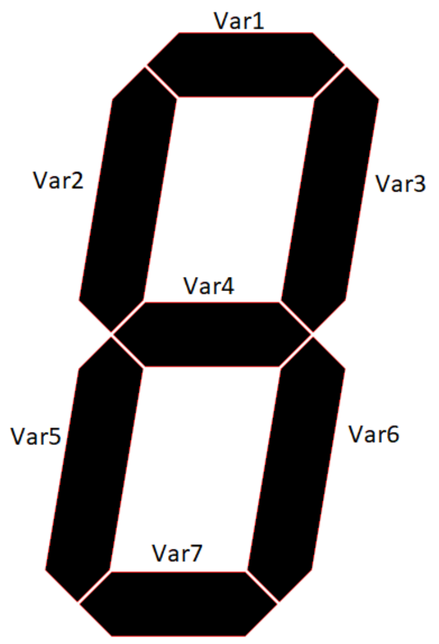

Рис № 16. Название предикторов и соответствующие им линии

Var1 – верхняя горизонтальная, Var2 – верхняя левая, Var3 – верхняя правая, Var4 – средняя горизонтальная, Var5 – нижняя левая вертикальная, Var6 – нижняя левая горизонтальная, Var7 – нижняя горизонтальная (рисунок №)

Интерес задачи представляет то, что калькулятор неисправен. При нажатии любой кнопки на экране не всегда отображается корректная комбинация линий.

На рисунке№ приведены первые 30 наблюдений

	1 DIGIT	2 VAR1	3 VAR2	4 VAR3	5 VAR4	6 VAR5	7 VAR6	8 VAR7
1	seven	ONE	ZERO	ONE	ZERO	ZERO	ONE	ZERO
2	one	ZERO	ZERO	ONE	ZERO	ZERO	ONE	ZERO
3	four	ZERO	ONE	ONE	ONE	ZERO	ONE	ZERO
4	two	ONE	ONE	ONE	ONE	ONE	ZERO	ZERO
5	eight	ZERO	ONE	ONE	ONE	ONE	ONE	ONE
6	one	ZERO	ZERO	ONE	ZERO	ZERO	ONE	ZERO
7	five	ONE	ONE	ZERO	ONE	ZERO	ONE	ONE
8	six	ONE	ZERO	ZERO	ONE	ONE	ONE	ONE
9	two	ONE	ZERO	ONE	ONE	ONE	ZERO	ONE
10	eight	ONE	ONE	ONE	ONE	ZERO	ONE	ONE
11	one	ZERO	ZERO	ONE	ZERO	ONE	ONE	ZERO
12	eight	ONE	ONE	ONE	ONE	ONE	ONE	ONE
13	one	ONE	ZERO	ONE	ZERO	ONE	ONE	ZERO
14	seven	ONE	ZERO	ONE	ZERO	ZERO	ONE	ZERO
15	seven	ZERO	ZERO	ONE	ZERO	ONE	ONE	ZERO
16	six	ONE	ONE	ZERO	ONE	ONE	ONE	ONE
17	zero	ONE	ZERO	ONE	ZERO	ONE	ONE	ZERO
18	one	ZERO	ZERO	ONE	ZERO	ZERO	ONE	ONE
19	four	ZERO	ONE	ONE	ONE	ZERO	ONE	ZERO
20	six	ONE	ONE	ZERO	ONE	ONE	ONE	ONE
21	four	ONE	ONE	ONE	ONE	ZERO	ONE	ZERO
22	nine	ONE	ONE	ONE	ONE	ZERO	ONE	ONE
23	five	ONE	ONE	ZERO	ONE	ZERO	ONE	ONE
24	nine	ONE	ONE	ZERO	ONE	ZERO	ONE	ONE
25	two	ONE	ZERO	ONE	ONE	ZERO	ZERO	ONE
26	zero	ONE	ONE	ONE	ONE	ONE	ONE	ONE
27	two	ONE	ZERO	ONE	ZERO	ONE	ZERO	ONE
28	three	ONE	ZERO	ZERO	ONE	ZERO	ONE	ONE
29	one	ZERO	ZERO	ONE	ZERO	ZERO	ONE	ZERO
30	six	ONE	ONE	ZERO	ONE	ONE	ONE	ONE

Рисунок № 17. Пример наблюдений

В переменной DIGIT записаны результаты распознавания цифры. Предикторы записаны в соответствующие переменные Var1 – Var7.

Требования к модели:

1) Должна быть возможность у сети экстраполировать за область обучающих данных (если комбинация предикторов сильно отличается от обучающего множества – ответ должен быть правильный)

2) Прогноз должен занимать минимум времени

В данной задаче обучался многослойный персептрон, так как по сравнению с другими алгоритмами он более экономичный по памяти, быстро достигает нужного значения ошибки. Следует учесть, что к точному минимуму ошибки данный алгоритм может ходиться довольно медленно.

Анализ результатов:

Наибольшее количество ошибок в распознавании приходится на цифру 9. Также был провзеден анализ чувствительности к предикторам. Анализ чувствительности позволяет сделать вывод об относительной важности входных переменных для конкретной нейронной сети и при необходимости удалить входы с низкими показателями чувствительности. Результаты анализа можно пронаблюдать в таблице (рис№)

Список литературы:

[1] Нейронные сети. STATISTICA Neural Networks: Методология и технологии современного анализа данных/ под редакцией В. П. Боровикова – 2-е изд., перераб и доп. – М.: Горячая линия – Телеком, 2008. – 392с