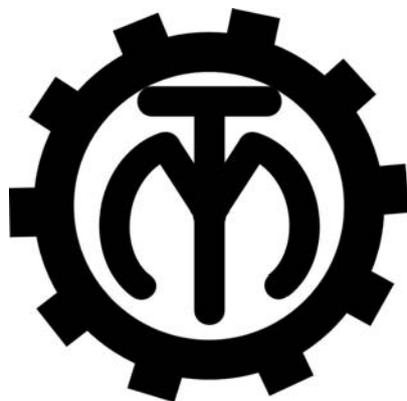


МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Кафедра «Технология машиностроения»

ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ В РОБОТОТЕХНИКЕ

*Методические рекомендации к практическим занятиям
для магистрантов направления подготовки
15.04.06 «Мехатроника и робототехника»
очной и заочной форм обучения*



Могилев 2023

УДК 621.865.8
ББК 32.816
И86

Рекомендовано к изданию
учебно-методическим отделом
Белорусско-Российского университета

Одобрено кафедрой «Технология машиностроения» «22» ноября 2022 г.,
протокол № 6

Составитель д-р техн. наук, проф. В. М. Пашкевич

Рецензент канд. техн. наук, доц. А. П. Прудников

В методических рекомендациях даны задания к практическим занятиям по дисциплине «Искусственный интеллект в робототехнике», приведены указания по их выполнению.

Учебно-методическое издание

ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ В РОБОТОТЕХНИКЕ

Ответственный за выпуск	В. М. Шеменков
Корректор	А. А. Подошевка
Компьютерная верстка	Н. П. Полевничая

Подписано в печать . Формат 60×84/16. Бумага офсетная. Гарнитура Таймс.
Печать трафаретная. Усл. печ. л. . Уч.-изд. л. . Тираж 36 экз. Заказ №

Издатель и полиграфическое исполнение:
Межгосударственное образовательное учреждение высшего образования
«Белорусско-Российский университет».
Свидетельство о государственной регистрации издателя,
изготовителя, распространителя печатных изданий
№ 1/156 от 07.03.2019.
Пр-т Мира, 43, 212022, г. Могилев.

© Белорусско-Российский
университет, 2023

Содержание

1 Искусственный интеллект и интеллектуальные агенты.....	4
2 Образы в искусственном интеллекте.....	5
3 Распознавание образов.....	7
4 Управление в робототехнических системах.....	10
5 Логические системы.....	12
6 Сети Петри в робототехнике.....	16
7 Поисковые алгоритмы в робототехнике.....	17
8 Обучение в робототехнике.....	18
9 Нейронные сети в робототехнике.....	21
Список литературы.....	33

1 Искусственный интеллект и интеллектуальные агенты

Задание

По заданию, выданному преподавателем, построить общее описание мира интеллектуального агента в соответствии со следующей схемой (рисунок 1).

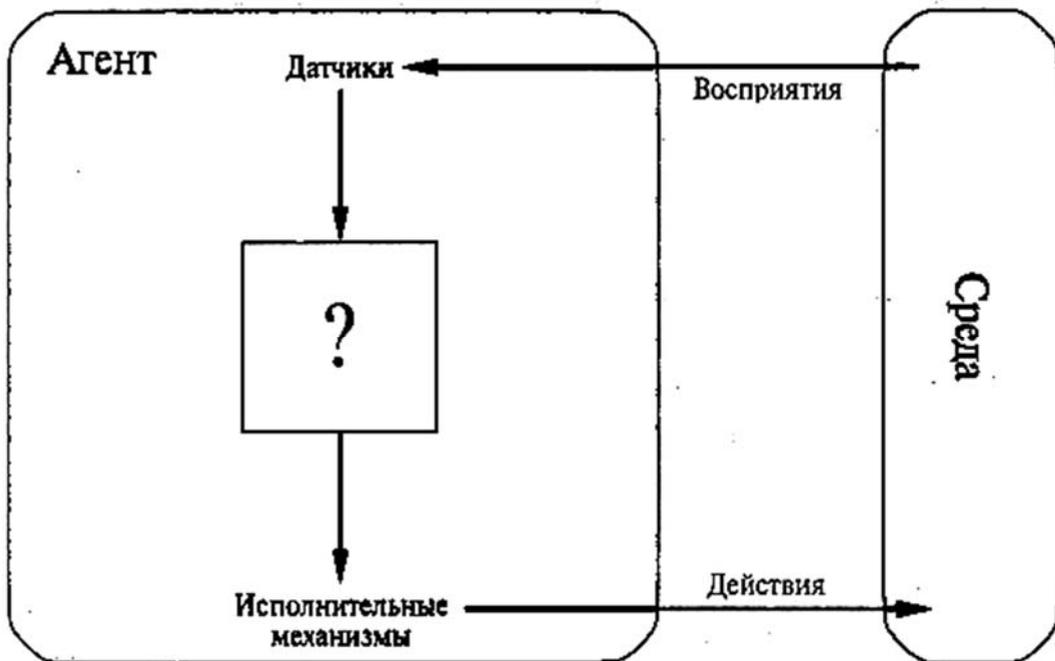


Рисунок 1 – К описанию мира интеллектуального агента

Порядок выполнения работы

1 Для варианта мобильного робота, выданного преподавателем, провести анализ и описать мир реализующего управление им интеллектуального агента:

- параметры окружающей среды (объекты и физические величины, несущие информацию о них), возможные состояния среды (случайные и детерминированные переходы состояний среды);
- каналы восприятия информации роботом (внутренние и внешние сигналы; физические величины; датчики или измерительные преобразователи; последовательности актов восприятия, их временные реализации);
- структуру взаимодействия программной среды робота и датчиков;
- исполнительные механизмы робота (предложить аппаратную реализацию исполнительных механизмов, описать принцип управления ими);
- описать набор возможных состояний робота (в зависимости от состояний окружающей среды и в зависимости от предпринятых роботом действий);
- описать пространство выполняемых роботом действий (элементарные действия; последовательности элементарных действий; временные задержки между действиями);

В качестве симптомов выбраны признаки (см. таблицу 1):

S_1 – повышенная вибрация на корпусе;

S_2 – превышение допускаемого уровня шума на 2...4 дБ.

В качестве диагностируемых состояний приняты:

D_1 – износ редуктора позиционера;

D_2 – увеличенный радиальный зазор подшипника;

D_3 – нормальное (работоспособное) состояние.

Порог принятия гипотезы выбрать равным 0,7.

Порядок выполнения работы

1 Разместить образы на рабочем листе *MS Excel* (рисунок 2).

2 Рассчитать априорные вероятности диагнозов и признаков, определив предварительно число передач, имеющих соответствующие признаки. Рекомендуется при расчете использовать встроенные функции *Excel* СЧЁТЗ, СУММЕСЛИ, СЧЁТЕСЛИМН.

3 Рассчитать вероятности принадлежности объектов классам D_1 , D_2 , D_3 по наблюдаемым *единичным* признакам S_1 и S_2 , а также по их отсутствию \bar{S}_1 и \bar{S}_2 по формуле

$$p(D_i | S_j) = p(D_i) \frac{p(S_j | D_i)}{\sum_{k=1}^3 p(S_k | D_i)}.$$

	A	B	C	D	E	F	G	H	I	J
1	№	1	2	3	4	5	6	7	8	9
2	S1	1	0	1	1	0	0	0	0	0
3	S2	0	1	0	0	1	1	1	1	0
4	D1	1	1	0	0	0	0	0	0	0
5	D2	0	0	1	1	1	1	1	1	0
6	D3	0	0	0	0	0	0	0	0	1
7										
8										
9	Наблюдений всего				n(D1)	n(D2)	n(D3)		n(S1)	n(S2)
10	40				2	6	32		3	7
11										
12					p(D1)	p(D2)	p(D3)		p(S1)	p(S2)
13					0,05	0,15	0,8		0,075	0,175
14										
15										
16	P(D1 S1)	P(D2 S1)	P(D3 S1)			p(D1 S1+S2)	p(D2 S1+S2)	p(D3 S1+S2)		
17	0,333333333	0,666666667	0			0,272727273	0,727272727	0		
18	P(D1 notS1)	P(D2 notS1)	P(D3 notS1)			p(D1 notS1+S2)	p(D2 notS1+S2)	p(D3 notS1+S2)		
19	0,666666667	0,333333333	1			0,096774194	0,516129032	0,387096774		
20	P(D1 S2)	P(D2 S2)	P(D3 S2)			p(D1 S1+notS2)	p(D2 S1+notS2)	p(D3 S1+notS2)		
21	0,142857143	0,666666667	0,285714286			0,428571429	0,571428571	0		
22	P(D1 notS2)	P(D2 notS2)	P(D3 notS2)			p(D1 notS1+notS2)	p(D2 notS1+notS2)	p(D3 notS1+notS2)		
23	0,857142857	0,333333333	0,714285714			0,015706806	0,041884817	0,942408377		

Рисунок 2 – Пример расчетного листа в *MS Excel*

4 Рассчитать вероятности принадлежности объектов классам D_1, D_2, D_3 по наблюдаемым *парным* признакам S_1 и S_2 , а также их отсутствию \bar{S}_1 и \bar{S}_2 по формуле

$$p(D_k | S_i S_j) = p(D_i) \frac{p(S_i S_j | D_k)}{p(S_i S_j)}.$$

5 Определить наиболее информативные комплексы признаков, превышающие порог принятия гипотезы.

Вопросы к защите

- 1 Каковы достоинства и недостатки метода Байеса?
- 2 Какие комплексы признаков наиболее информативны для метода Байеса?
- 3 Как строятся решающие правила по методу Байеса?

3 Распознавание образов

Задание

Для оценки возможности перемещения мобильного робота в ограниченном пространстве предложено использовать сигналы двух ультразвуковых преобразователей x_1 и x_2 , смонтированных, соответственно, над левой и правой колесными арками. Установлено, что на основе анализа этих сигналов можно разделять состояния среды на два следующих класса (рисунок 3):

- класс 1: коридор узкий, возможно застревание робота;
 - класс 2: коридор достаточной ширины, возможно дальнейшее перемещение.
- Первые 10 образцов принадлежат классу 1 ($E1$), остальные – к классу 2 ($E2$).

X1	1,5	1,9	3,8	2,5	3,8	2,5	2	1,3	3,8	2	?
X2	1	2,3	3,3	4,2	5,5	1,9	5	4	1	4,1	?
											E1

X1	12,8	10	8,4	12,2	10,8	13,3	8,2	10,4	10,5	13,3	?
X2	16,3	12	10,2	7,5	15,8	8,8	16,4	10,6	7,8	13,6	?
											E2

7	7
12	8
O1	O2

Рисунок 3 – Исходные данные

Центроидным методом определить эталоны (средние значения) E_1 и E_2 этих классов.

Определить состояние объектов O_1 и O_2 по сигналам преобразователей $x_1(O_1)$, $x_2(O_1)$ и $x_1(O_2)$, $x_2(O_2)$ на основе методики метрического распознавания образов.

Построить систему метрического распознавания образов и изучить ее чувствительность к изменению параметров распознавания.

Порядок выполнения работы

1 Построить графическое изображение образцов и рассчитанных эталонов E_1 и E_2 , а также объектов O_1 и O_2 в пространстве признаков x_1 и x_2 . Выделить маркерами разных типов образцы, центроиды и эталоны.

2 В качестве меры расстояния между эталоном и объектом принять обобщенную метрику

$$L = M \sqrt{\sum_{k=1}^2 |E_k - O_k|^N},$$

где 2 – размерность пространства признаков;

N, M – показатель степени обобщенного расстояния.

Определить расстояния от исследуемых объектов O_1 и O_2 до эталонов классов E_1 и E_2 при различных показателях степени M и N обобщенного расстояния, а также определить надежности принадлежности объектов классам 1 и 2.

3 Порог надежности для принятия гипотезы о принадлежности объекта выбранному классу состояний принять равным 0,7.

4 Оценить влияние показателей N, M (меры резкости пространства) на надежность распознавания.

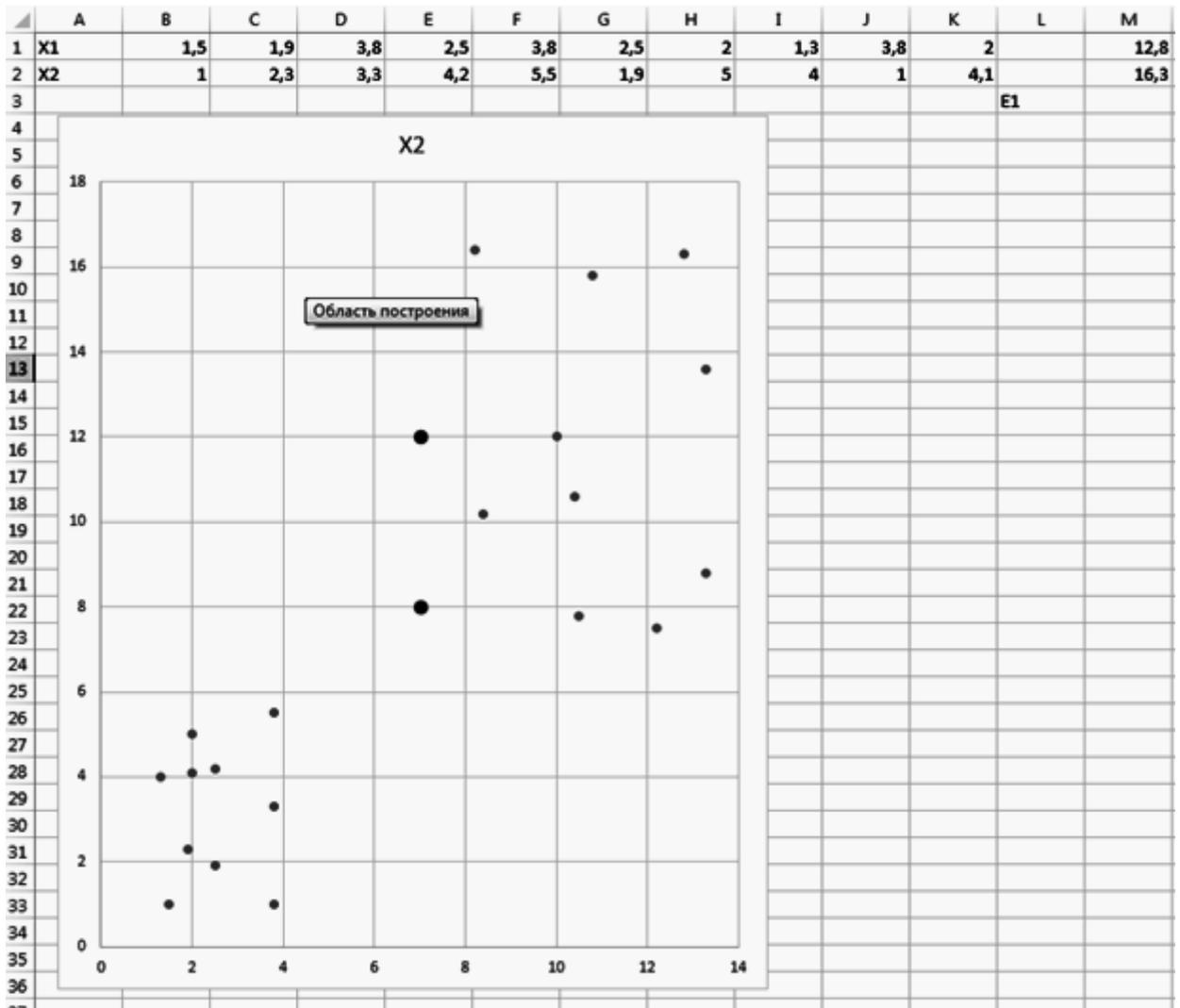
Пример отчета представлен на рисунке 4.

Вопросы к защите

1 В чем заключается процедура метрического распознавания образов?

2 Что такое метрика пространства?

3 Как влияет метрика пространства на надежность распознавания?



N	O	P	Q	R	S	T	U	V	W	X	Y
10	8,4	12,2	10,8	13,3	8,2	10,4	10,5	13,3		7	7
12	10,2	7,5	15,8	8,8	16,4	10,6	7,8	13,6		12	8
									E2	O1	O2
M											
2											
N											
2											
L(O1, E1)	13,89244		p(O1 ∈ E1)	0,5							
L(O1, E2)	13,89244		p(O1 ∈ E2)	0,5							
L(O2, E1)	10,63015		p(O2 ∈ E1)	0,5							
L(O2, E2)	10,63015		p(O2 ∈ E2)	0,5							

Рисунок 4 – Примерный вид отчета по работе

4 Управление в робототехнических системах

Задание

Построить конечный автомат, обеспечивающий управление процессом сборки узла механического редуктора. Сборочный робот, снабженный системой технического зрения, должен при этом обеспечить гибкий сценарий сборки (выполнять сборку с учетом произвольно появляющихся на конвейере деталей, а также с учетом деталей, уже установленных в сборочный узел).

Варианты заданий представлены в таблице 2 и выбираются по номеру студента в списке учебной группы. Схемы собираемых узлов приведены на рисунке 5.

Таблица 2 – Варианты заданий

Вторая цифра номера	Схема собираемого узла
0	Рисунок 5, а
1	Рисунок 5, б
2	Рисунок 5, в
3	Рисунок 5, г
4	Рисунок 5, д
5	Рисунок 5, е
6	Рисунок 5, ж
7	Рисунок 5, з
8	Рисунок 5, и
9	Рисунок 5, к

Порядок выполнения работы

Построить автоматную таблицу, содержащую набор команд для управления сборочным роботом, учитывающих возможные комбинации входных сигналов X (совокупность деталей на конвейере), **допустимых** внутренних состояний автомата Y (совокупность деталей, уже установленных в узле).

Базовой деталью является деталь, обозначенная на схеме позицией 0; по конвейеру движутся детали, обозначенные позициями 1–3.

Пример входных сигналов:

«000» – на сборочном конвейере нет деталей;

«100» – по конвейеру движется деталь, обозначенная на схеме позицией 1;

«011» – по конвейеру движутся детали, обозначенные позициями 2 и 3.

Пример внутренних состояний:

«000» – в узле нет установленных деталей;

«001» – в узел установлена деталь, обозначенная на схеме позицией 3;

«111» – все детали установлены в узел.

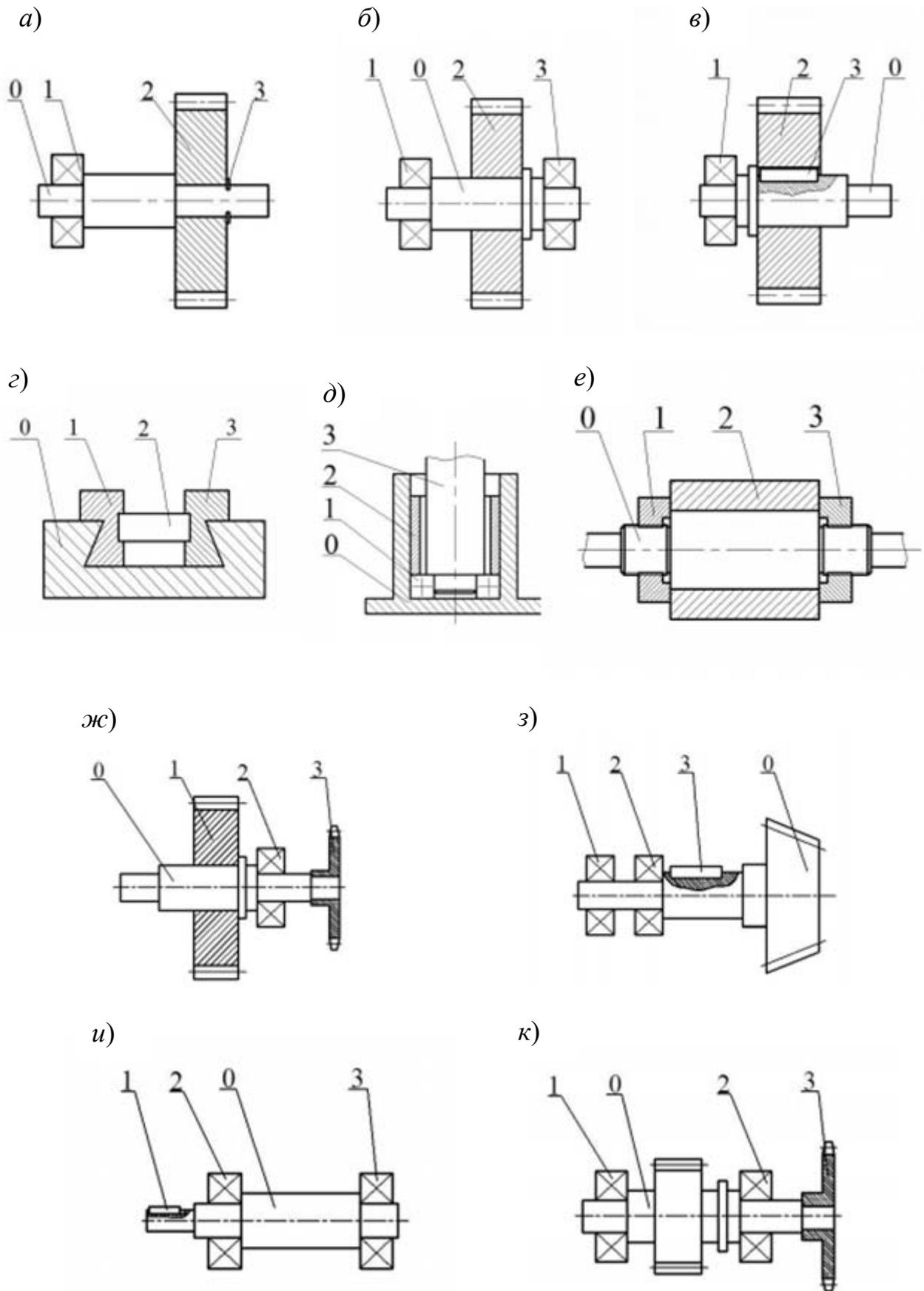


Рисунок 5 – Схемы собираемых узлов

Отобрать из общей совокупности внутренних состояний только те из них, которые допустимы (исходя из возможных последовательностей сборки).

Пример команд робота:

«W» (*wait*) – не выполнять каких-либо действий, ожидать изменения входного сигнала;

«R» (*release*) – снять собранный узел и установить в сборочное приспособление базовую деталь, обозначенную на схеме позицией 0 (выйти в исходное положение для сборки);

«1» – манипулятором № 1 взять деталь, обозначенную на схеме позицией 1, и установить на базовую деталь;

«2» – манипулятором № 2 взять деталь, обозначенную на схеме позицией 2, и установить на базовую деталь.

Вопросы к защите

1 В чем заключается основная идея автоматного описания?

2 Как формируется автоматная таблица?

3 Что обеспечивает гибкость автоматного описания?

5 Логические системы

Задание

Для управления роботом-сборщиком построить на основе коэффициентов определенности и базы данных о состоянии яблок сорта «Синап» по результатам спектрального анализа их окраски систему для распознавания двух состояний:

D_1 – сорт высший (отсутствуют видимые дефекты);

D_2 – бессортовые (присутствуют заметные пятна гнили, парши, мелкие яблоки).

В качестве критериев распознавания были выбраны следующие признаки:

S_1 – диаметральный размер более 9 см;

S_2 – примесь цвета на поверхности – коричневая ($RGB = 115; 66; 34$), более 5 %;

S_3 – отсутствие значительных примесей черного цвета на поверхности ($RGB = 40; 40; 40$), менее 5 %;

S_4 – преобладающая цветовая гамма – желтая ($RGB = 255; 219; 139$) или красная ($RGB = 213; 48; 50$), более 90 %.

В таблице 3 приведена база данных с наблюдениями за яблоками, соответствующими состояниям D_1 и D_2 . При этом значение «1» соответствует наличию признака S_i , а значение «0» – его отсутствию.

Таблица 3 – Наблюдения за яблоками сорта «Синап» с признаками $S_1 \dots S_4$

№	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
S_1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
S_2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
S_3	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
S_4	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
D_1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
D_2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Продолжение таблицы 3

№	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
S_1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
S_2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
S_3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
S_4	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
D_1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
D_2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Продолжение таблицы 3

№	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
S_1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0
S_2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	1	1
S_3	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
S_4	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1
D_1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D_2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Порядок выполнения работы

1 Разместить данные на рабочем листе *MS Excel* (рисунок 6).

2 Приняв в качестве базового состояния D_1 (высший сорт), рассчитать значения коэффициентов определенности для каждого из признаков S_i , используя функции СЧЁТЕСЛИМН.

3 Проверить качество распознавания по полученным коэффициентам определенности, оценив с их помощью принадлежность объектов состояниям D_1 и D_2 . В качестве порога для принятия гипотезы выбрать положительное (отрицательное) значение коэффициента определенности для комбинации признаков.

	A	B	C	D	E	F	G	H	I
1	№	1	2	3	4	5	6	7	8
2	S1	1	1	1	1	1	1	1	1
3	S2	0	0	0	0	0	0	0	0
4	S3	1	1	1	1	1	1	1	1
5	S4	0	0	0	0	0	0	0	0
6	D1	1	1	1	1	1	1	1	1
7	D2	0	0	0	0	0	0	0	0
8	D-?	0,9746875	0,9746875	0,9746875	0,974688	0,974688	0,974688	0,974688	0,97
9									
10									
11									
12									
13		S1	S2	S3	S4	п общ			
14	D1								
15	D2								
16									
17	МД								
18	МНД								
19									
20	CF								

Рисунок 6 – Пример расчетного листа в *MS Excel*

Для автоматизации процедуры можно использовать *VBA*-макрос следующего содержания:

```
Sub Main()
```

```
CF1 = 0.8
CF2 = -0.5
CF3 = -0.25
CF4 = 0.15
```

```
For i = 1 To 60
```

```
    ' значения признаков S и диагноза D
```

```
    S1 = Range("a2").Offset(0, i)
```

```
    S2 = Range("a3").Offset(0, i)
```

```
    S3 = Range("a4").Offset(0, i)
```

```
    S4 = Range("a5").Offset(0, i)
```

```
    D1 = Range("a6").Offset(0, i)
```

```
    ' значение коэффициента определенности CFSi для текущего
```

```
    ' признака Si
```

```
        CFS1 = CFS(CF1, S1)
```

```
        CFS2 = CFS(CF2, S2)
```

```
        CFS3 = CFS(CF3, S3)
```

CFS4 = CFS(CF4, S4)

' комбинация четырех коэффициентов определенности

c12 = CF12(CFS1, CFS2)

c123 = CF12(c12, CFS3)

c1234 = CF12(c123, CFS4)

' записать вычисленный коэффициент определенности для D1
Range("a8").Offset(0, i) = c1234

Next i

End Sub

Function CFS(CF, S) *' функция для вычисления k-та определенности*

If S = 1 Then

CFS = CF

Else

If CF >= 0 Then CFS = 1 - CF

If CF < 0 Then CFS = 1 + CF

End If

End Function

Function CF12(CF1, CF2) *' функция для расчета комбинации*
' коэффициентов определенности

CF12 = CF1 + CF2 * (1 - Abs(CF1))

End Function

4 Определить величину ошибки системы распознавания, если для принятия гипотезы используется значение коэффициента определенности для комбинации признаков, превышающее 0,7.

Вопросы к защите

1 Что такое решающее правило?

2 В каком случае решающее правило не может быть построено?

3 При каких условиях можно применять решающее правило?

6 Сети Петри в робототехнике

Задание

Для модели РТК в форме сети Петри описать условия коллаборативного взаимодействия станка с ЧПУ и обслуживающего робота, описываемых сетью (рисунок 7), с ограничением по объему буфера-накопителя (B, B').

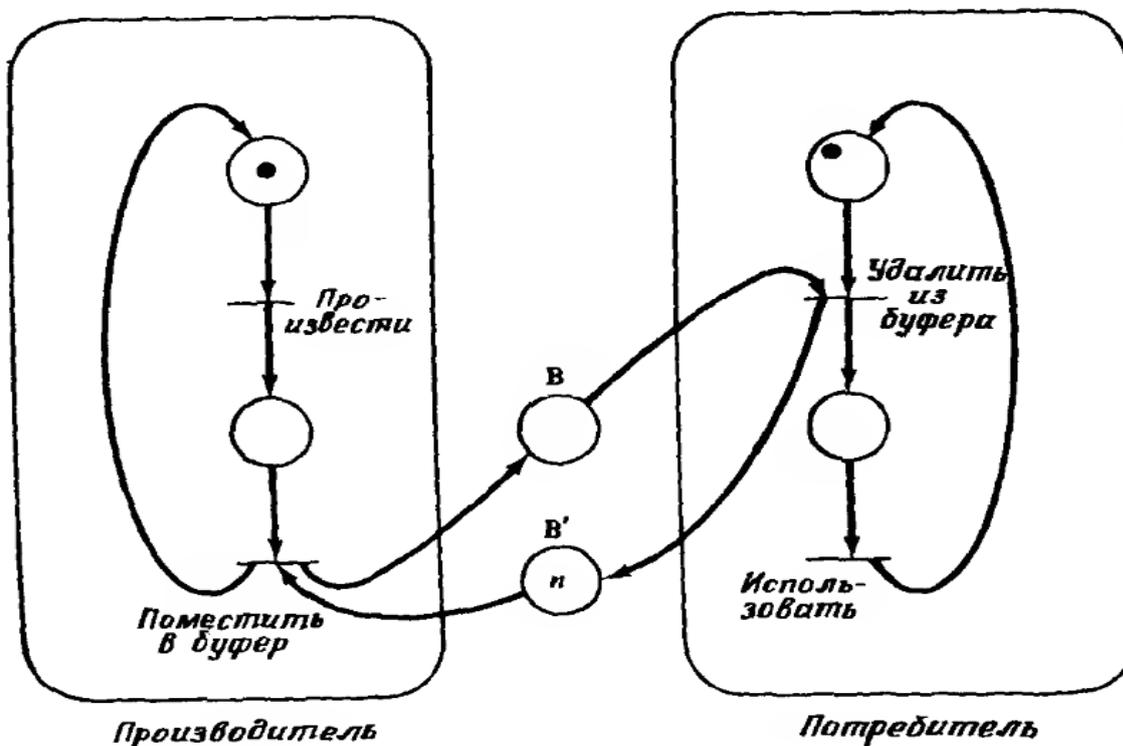


Рисунок 7 – Задача о взаимодействии потребителя и производителя

Порядок выполнения работы

- 1 Описать последовательность действий коллаборативной системы при наличии буфера единичного объема ($n = 1$) и при большем объеме (по заданию преподавателя).
- 2 Показать последовательность выполняемых операций, включая перемещение маркеров, описание переходов.
- 3 Определить поведение системы при наличии простоев и заделов.
- 4 Построить модель динамической системы в форме сети Петри для описания процессов обслуживания роботом-загрузчиком разнотипных станков, объединенных в РТК.

Вопросы к защите

- 1 Какие типы вершин применяются в сетях Петри?
- 2 Какие переходы являются допустимыми?
- 3 Как формируется последовательность действий в сетях Петри?

7 Поисковые алгоритмы в робототехнике

Задание

Для пространства состояний для мира робота-пылесоса (рисунок 8) описать алгоритмы поиска для очистки пространства. Дуги переходов на рисунке 8 обозначают действия: L – переместиться влево; R – переместиться вправо; S – удалить мусор.

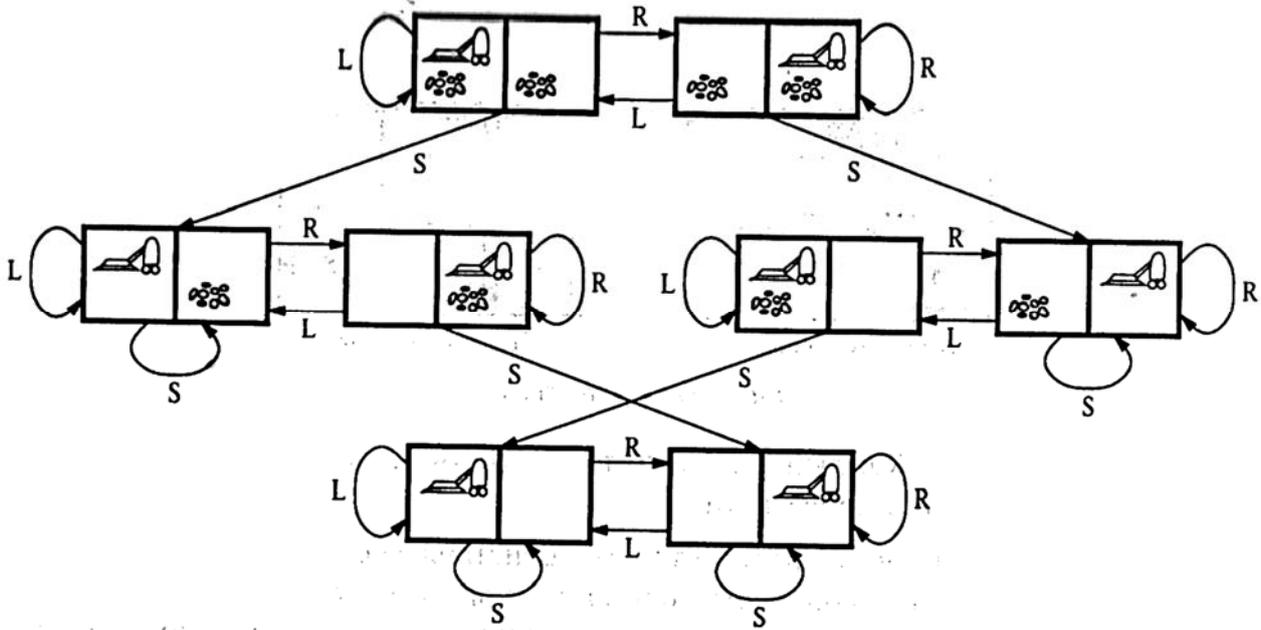


Рисунок 8 – Пространство состояний для мира робота-пылесоса

Порядок выполнения работы

Описать возможные *состояния* интеллектуального агента, если комната условно разделена на квадраты.

Определить возможные *начальные состояния*.

Описать *функцию определения приемника*, которое является следствием попыток выполнения трех действий (L , R , S).

Определить *функцию проверки достижения цели* (очистка всех квадратов).

Определить *функцию оценки стоимости пути* (стоимость перемещения для достижения цели).

Рассмотреть решение задачи различными *методами поиска*: поиском в глубину; в ширину; поиском с возвратом; случайным поиском.

Вопросы к защите

- 1 В чем состоит смысл поиска решения интеллектуальными агентами?
- 2 Какие методы поиска могут быть использованы, в чем их достоинства и недостатки?
- 3 Зачем используется функция оценки стоимости пути?

8 Обучение в робототехнике

Задание

Используя метод, базирующийся на процедуре обучения, обучить систему распознаванию двух классов состояний.

При этом в качестве представителей первого класса выбраны объекты P_1 и P_2 , а в качестве представителей второго класса – объекты P_3 и P_4 . Объекты характеризуются двумерным набором признаков – x_1 и x_2 .

Варианты заданий приведены в таблице 4.

Таблица 4 – Исходные данные

Класс 1				Класс 2			
P_1		P_2		P_3		P_4	
x_1	x_2	x_1	x_2	x_1	x_2	x_1	x_2
0	1	2	1	6	7	8	9

Порядок выполнения работы

1 Расположить данные, весовые коэффициенты и норму обучения в ячейках рабочего листа, построить точечные графики классов, выделив их рядами разных цветов.

2 Добавить к диаграмме ряд данных в форме, соответствующий дискриминантной функции $d = w_0 + w_1x_1 + w_2x_2$. Считая, что для построенной диаграммы $x = x_1$, а $y = x_2$, построить в интервале изменения переменной x_1 в пределах обоих классов (т. е. 0...8) соответствующую дискриминантной функции линию

$$y = -\frac{w_0}{w_2} - \frac{w_1}{w_2}x_1.$$

Приблизительный вид рабочего листа приведен на рисунке 9.

3 Пользуясь алгоритмом обучения, найти выражение для дискриминантной функции, контролируя процесс обучения визуально. Вычисление функции принадлежности ведется по формуле

$$y = \text{sign}(d) = \text{sign}(w_0 + w_1x_1 + w_2x_2),$$

которая равна +1 для класса 1 и –1 для класса 2, а корректировка весовых коэффициентов – по выражению

$$w_i^{n+1} = w_i^n + c \cdot (y_i - y_r) \cdot x_i,$$

где c – мера обучения, $c = 0...1$;

Y_i и Y_r – требуемое и действительное значения функции принадлежности.

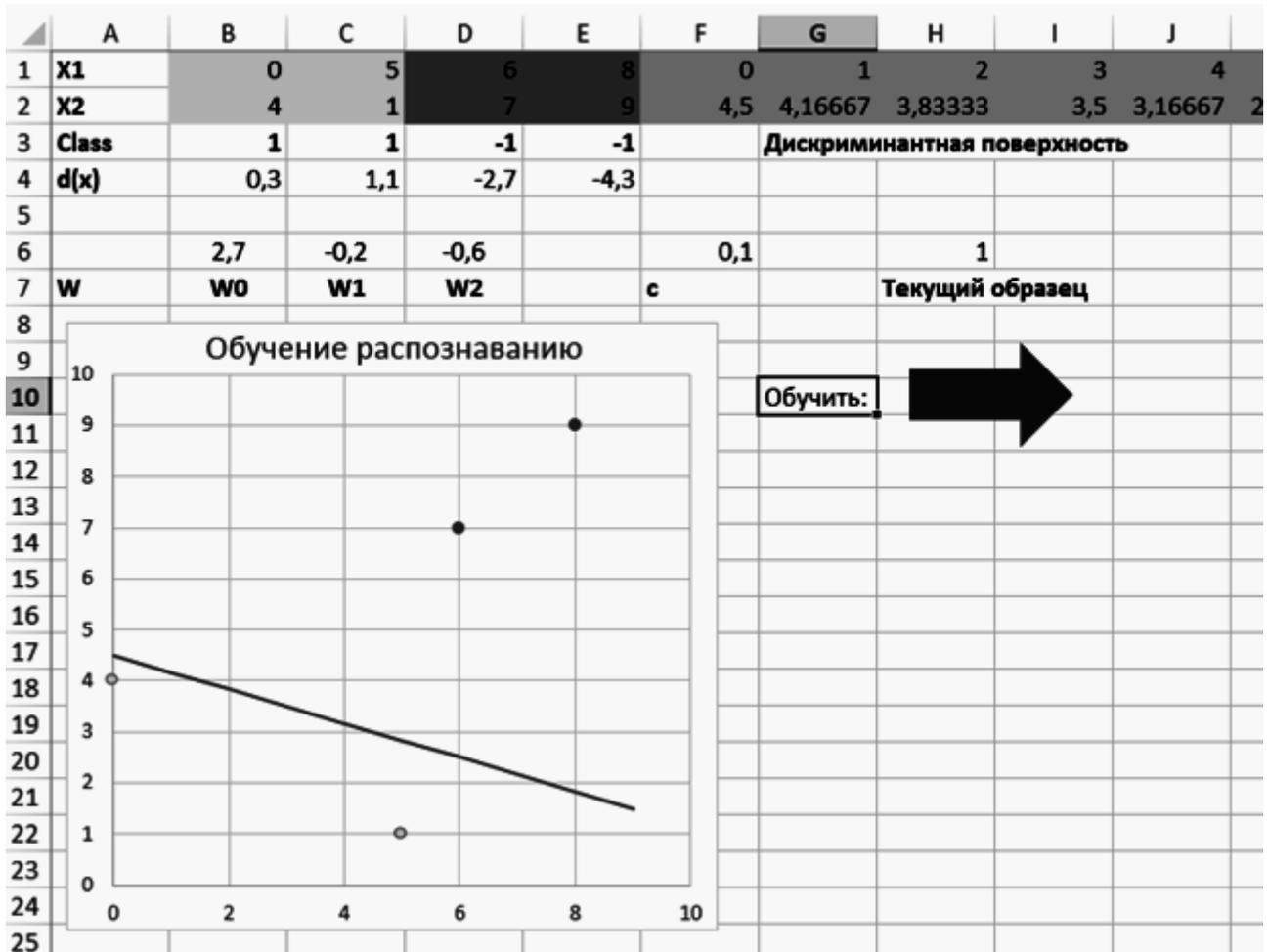


Рисунок 9 – Примерный вид отчета по работе

Для автоматизации вычислений рекомендуется использовать VBA-макрос следующего вида:

```
Sub Main()
```

```
    ' Загрузка данных
```

```
    Curr_item = Range("h6")           ' номер текущего образца
```

```
    x1 = Range("a1").Offset(0, Curr_item)
```

```
    x2 = Range("a2").Offset(0, Curr_item)
```

```
    c = Range("f6")                   ' скорость обучения
```

```
    w0 = Range("b6")                  ' коэффициенты дискриминантной функции
```

```
    w1 = Range("c6")
```

```
    w2 = Range("d6")
```

```
    d_targ = Range("a3").Offset(0, Curr_item)
```

```

' требуемое значение дискр.функции для текущего образца
  d_real = Range("a4").Offset(0, Curr_item)
' значение дискр.функции для текущего образца (без sign)
  If d_real = 0 Then ' защита от деления на ноль
    d_real = 0
  Else
    d_real = d_real / Abs(d_real) ' функция sign
  End If

```

```

' Проверка правильности распознавания

```

```

  If d_real = d_targ Then ' распознано верно
    MsgBox ("Образец № " & Curr_item & " распознан верно!")

    If Curr_item = 4 Then ' перейти к следующему образцу
      Curr_item = 1
    Else
      Curr_item = Curr_item + 1
    End If

```

```

    Range("h6") = Curr_item
' записать номер следующего образца для распознавания
  Else

```

```

' Скорректировать весовые k-ты

```

```

    MsgBox ("Образец № " & Curr_item & " распознан неверно!" & Chr(13)
& "Корректируем дискриминантную функцию !")

```

```

    w0 = w0 + c * (d_targ - d_real) * 1
    w1 = w1 + c * (d_targ - d_real) * x1
    w2 = w2 + c * (d_targ - d_real) * x2

```

```

    Range("b6") = w0
' записать скорректированные коэфф.дискриминантной функции
    Range("c6") = w1
    Range("d6") = w2

```

```

  If Curr_item = 4 Then ' перейти к следующему образцу
    Curr_item = 1
  Else
    Curr_item = Curr_item + 1
  End If

```

```

  Range("h6") = Curr_item

```

' записать номер следующего образца для распознавания

End If

End Sub

4 Изучить скорость схождения алгоритма при изменении в пределах $0,01 \dots 1$ меры обучения s .

Вопросы к защите

1 В чем заключается смысл обучения распознаванию образов?

2 Что такое дискриминантная функция?

3 В чем заключается проблема линейной делимости образов?

9 Нейронные сети в робототехнике

Задание

С помощью нейронной сети, состоящей из одного нейрона с одним входом x_1 (рисунок 10), построить процедуру, позволяющую системе поднастройки станка предсказать изменение размера токарного резца l во времени t , связанное с его износом. Для представленного нейрона принять

$$\Sigma = w_0 + w_1 x_1; \quad y = f(\Sigma) = \Sigma \cdot 1.$$

Предполагается, что размер резца (расстояние от некоторой измерительной базы на передней поверхности резца до его режущей кромки) изменяется прямо пропорционально времени при обработке заготовки с постоянными режимами резания (т. е. когда путь резания пропорционален времени).

Таким образом, вход нейрона x_1 соответствует времени t ; выходная переменная y – размеру l .

Экспериментальные данные приведены в таблице 5.

Таблица 5 – Данные для обучения

t	$l_{экс}$	t	$l_{экс}$
0	10,5	15	9,1
5	9,9	20	8,3
10	9,5	25	7,8

Принять в качестве модели износа зависимость в форме

$$\hat{l} = l_0 + kt, \quad (1)$$

где \hat{l} – текущий размер резца;
 l_0 – размер резца в начальный момент времени;
 k – интенсивность износа во времени;
 t – текущий момент времени.

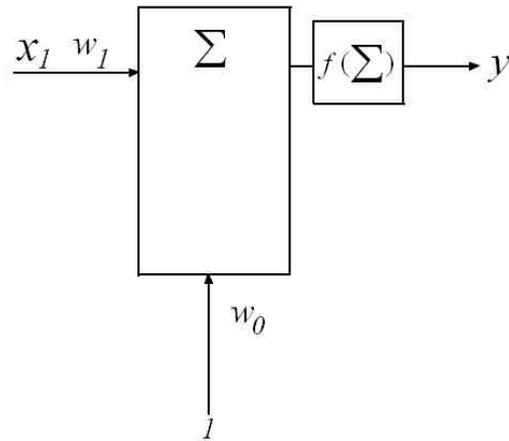


Рисунок 10 – Схема нейрона для моделирования износа резца

Порядок выполнения работы

1 Перенести данные на рабочий лист, построить их график, разместить также на нем ячейки для нормы обучения c и коэффициентов l_0 и k .

Примерный вид отчета приведен на рисунке 11.

2 Найти зависимость размера резца от времени, обучив нейронную сеть. При нахождении коэффициентов l_0 и k использовать упрощенные формулы, например

$$l_{0\ n+1} = l_{0\ n} + c \cdot (l_{\text{экс}} - l_{\text{мод}});$$

$$k_{n+1} = k_n + c \cdot (l_{\text{экс}} - l_{\text{мод}}),$$

где k_{n+1} и k_n – значения коэффициента k после и до корректировки;

c – коэффициент скорости обучения (норма обучения);

$l_{\text{экс}}$ – экспериментальное (требуемое) значение размера, приведенное в таблице 5;

$l_{\text{мод}}$ – модельное (расчетное) значение размера, полученное по формуле (1).

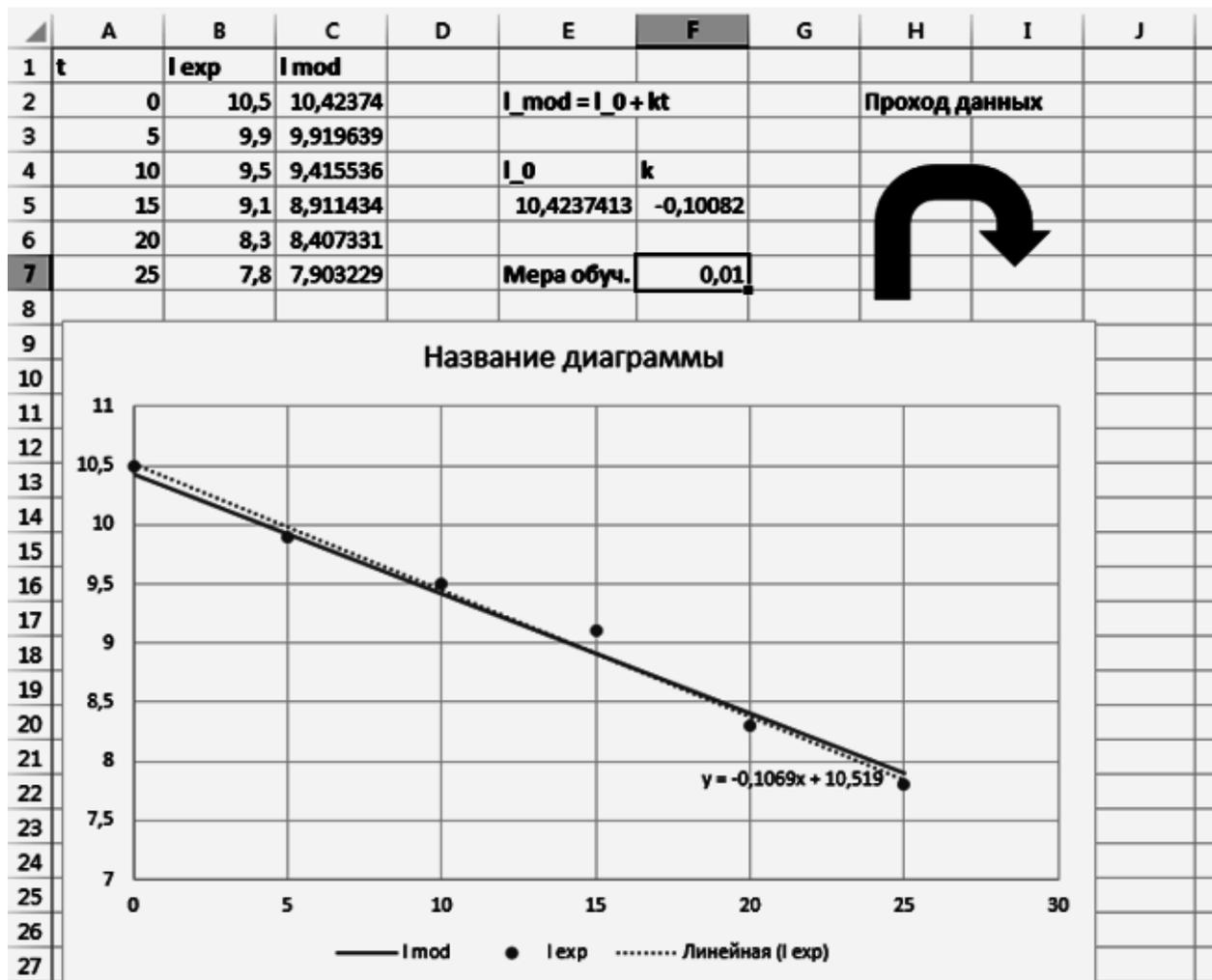


Рисунок 11 – Примерный вид отчета по работе

В качестве начальных приближений принять значения коэффициентов, равные нулю. Коэффициент скорости обучения принять равным 0–1. Обучение вести с помощью VBA-макроса, например

```
Sub Main()
```

```
    c = Range("f7")      ' норма обучения
```

```
    L0 = Range("e5")     ' начальный размер
```

```
    k = Range("f5")     ' коэффициент интенсивности износа
```

```
    For i = 1 To 6
```

```
        t = Range("a1").Offset(i, 0)      ' экспериментальные данные
```

```
        Iexp = Range("b1").Offset(i, 0)
```

```
        Imod = L0 + k * t                  ' модельное значение
```

```
        L0 = L0 + c * (Iexp - Imod)       ' корректировка k-та L0
```

```
    Next i
```

```
Range("e5") = L0
```

```
Range("f5") = k
```

```
For i = 1 To 6 ' обновить таблицу с модельными данными
```

```
  t = Range("a1").Offset(i, 0)
```

```
  lmod = L0 + k * t
```

```
  Range("c1").Offset(i, 0) = ymod
```

```
Next i
```

```
For i = 1 To 6
```

```
  t = Range("a1").Offset(i, 0)
```

```
  lexp = Range("b1").Offset(i, 0)
```

```
  ymod = L0 + k * t
```

```
  k = k + c * (lexp - lmod) ' корректировка k-та k
```

```
Next i
```

```
Range("e5") = L0
```

```
Range("f5") = k
```

```
For i = 1 To 6 ' обновить таблицу с модельными данными
```

```
  t = Range("a1").Offset(i, 0)
```

```
  lmod = L0 + k * t
```

```
  Range("c1").Offset(i, 0) = lmod
```

```
Next i
```

```
End Sub
```

3 Сравнить результаты, полученные в п. 2, с решением, полученным методом наименьших квадратов (используя опцию *Добавить линию тренда* для графика).

4 Исследовать влияние нормы обучения c на скорость сходимости и точность результата.

5 Предложить условия останова обучения.

Вопросы к защите

1 Какова структура искусственного нейрона?

2 Какую структуру имеют сети архитектуры *MLP*?

3 В чем сущность обучения нейронных сетей?

Задание

Используя trial-версию пакета *Statistica Neural Networks (SNN)*, построить для распознавания данных сеть архитектуры *MLP* и обучить ее различными методами.

Порядок выполнения работы

1 Запустить программу *SNN* и загрузить файл с тестовыми данными *Irisdat.sta* («Ирисы Фишера») из папки *STATISTICA 10\Examples\Datasets*.

Тестовая задача заключается в построении системы распознавания трех линейно неразделимых классов (трех видов ирисов) – *I.Setosa*, *I.Virginic*, *I.Versicolor* – по четырем признакам – длина чашелистика (переменная *Sepallen*), ширина чашелистика (переменная *Sepalwid*), длина лепестка (переменная *Petalen*), ширина лепестка (переменная *Petalwid*). В данных присутствуют экземпляры, являющиеся выбросами, т. е. выбивающиеся из общей картины данных.

2 Выбрать пакет *Neural Networks* на закладке *Data Mining*.

Выбрать тип задачи – *Classification* на панели *New Analysis* (рисунок 12).

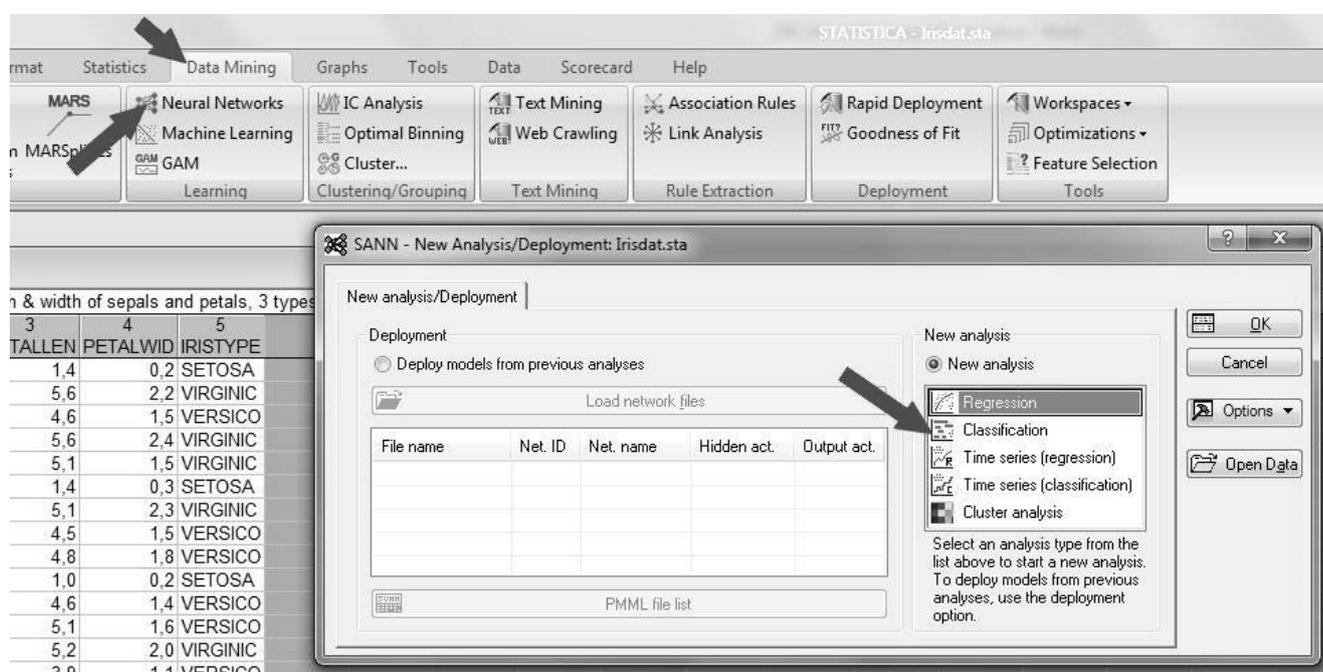


Рисунок 12 – Запуск пакета *SNN* и выбор типа задачи

Выбрать на закладке *Quick* определение типа переменных (*Variables*) (рисунок 13).

Назначить переменным тип:

IRISTYPE – *Categorical target* (качественная выходная переменная);

SEPALLEN, *SEPALWID*, *PETALLEN*, *PETALWID* – *Continuous inputs* (непрерывная числовая входная переменная) (рисунок 14).

На закладке *Quick* (Быстрый конструктор сетей) определить тип сети – *MLP* (*Multy-Layer Perceptron*, многослойный персептрон); число скрытых слоев сетей (*Min & Max hidden units*) в сетях; число сетей, автоматически генерируемых для обучения (*Networks to train*); количество отбираемых, наилучших сетей (*Networks to retain*); тип функции для контроля за ошибкой обучения – сумма квадратов невязок (*Sum of Squares*) (рисунок 15).

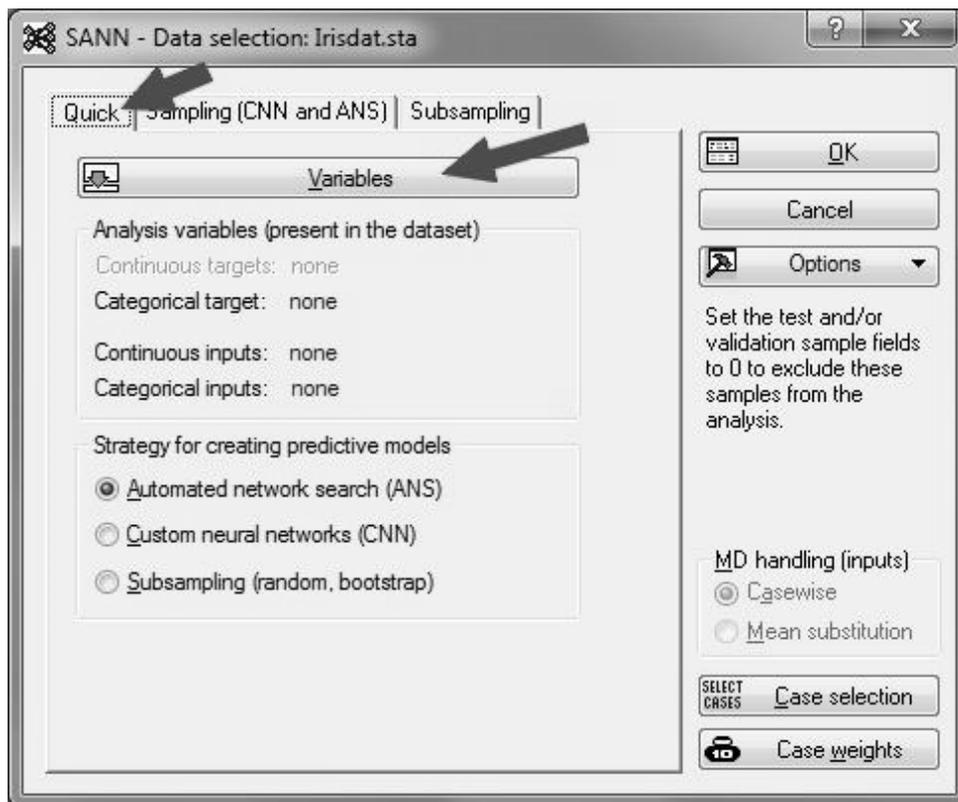


Рисунок 13 – Определение типа переменных

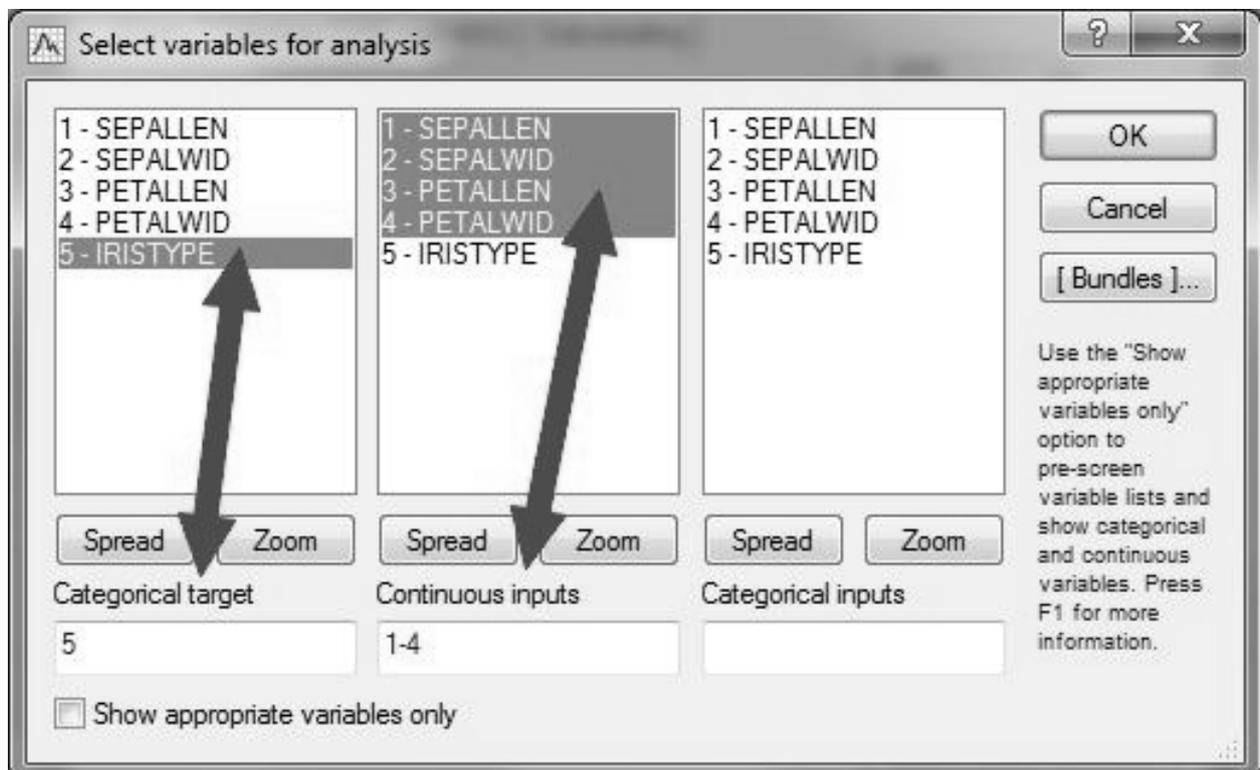


Рисунок 14 – Назначение типов переменных

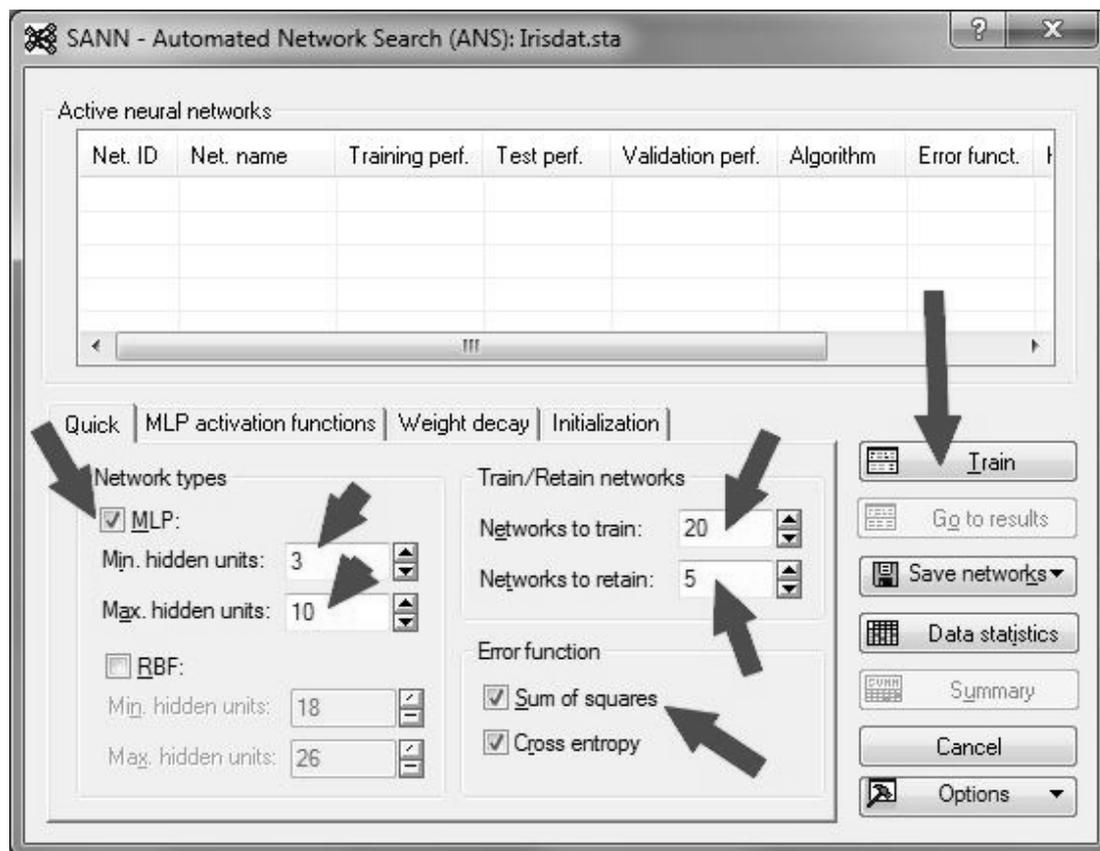


Рисунок 15 – Выбор типа и структуры сети

На закладке *MLP Activation Function* выбрать тип функции активации нейронов. Наиболее часто с этой целью применяется сигмоидальная (логистическая) функция. Также могут быть выбраны функции *Identity* (без сжатия исходных данных), *Tanh* (гиперболический тангенс), *Exponential* (экспоненциальная) (рисунок 16).

Выбрать на закладке *Weight Decay* (затухание весов) норму обучения, характеризующую скорость изменения весов в скрытых и выходном слое сети (рисунок 17).

Запустить процедуру генерации сетей и их обучения, нажав на кнопку *Train*.

Результаты обучения выводятся в окне *Results* (рисунок 18). Наилучший результат распознавания указан для каждой сети в поле *Training perf.* В этом же окне можно построить двухмерный (*X and Y*) или трехмерный (*X, Y and Z*) график входных и выходных переменных сети. Графики строятся совмещенными со значениями выходного сигнала сети, могут вращаться, что делает представление информации весьма наглядным и удобным для анализа (рисунок 19).

Кроме того, в режиме конструктора сетей возможно также построение в реальном масштабе времени графиков ошибок обучения и ошибок распознавания (запуск кнопкой *Training Error* на рисунке 18) в режиме кросс-проверки (см. рисунок 19). При этом строятся как график ошибки обучения (*Train error*), так и ошибки проверки обучения на тестовом множестве (*Test error*).

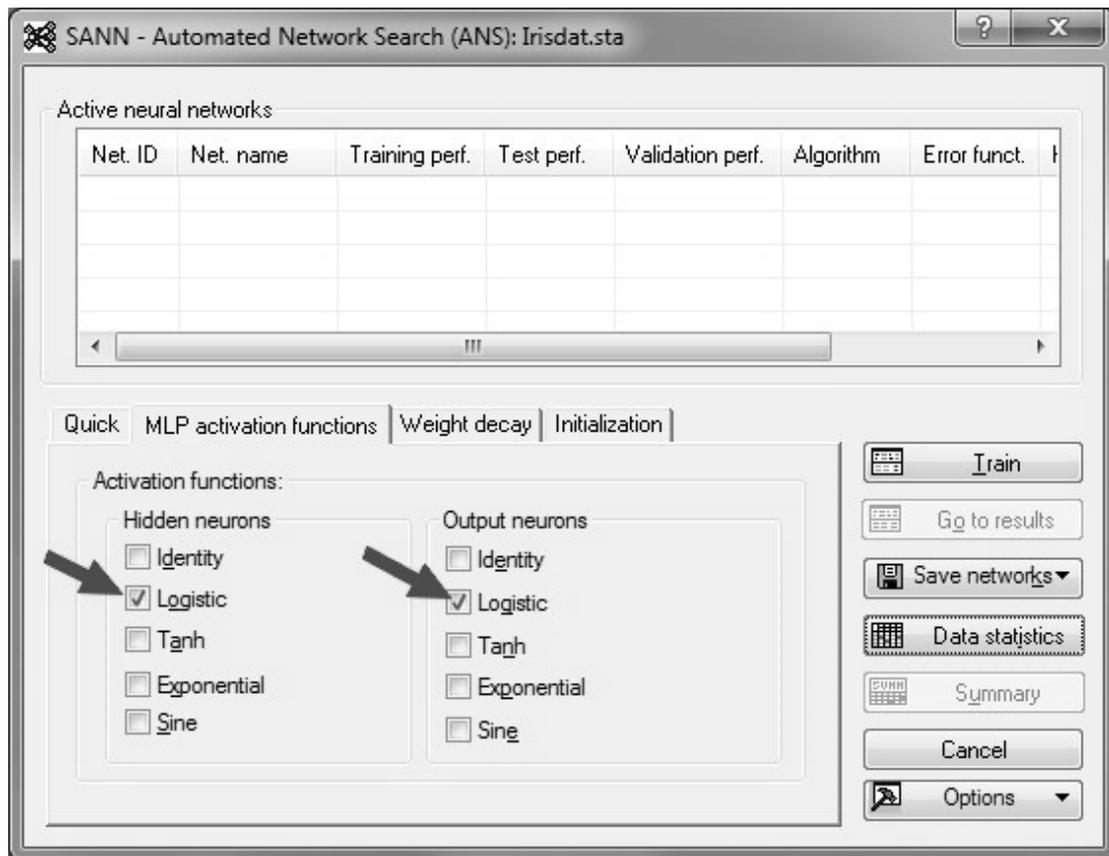


Рисунок 16 – Выбор функции активации нейронов

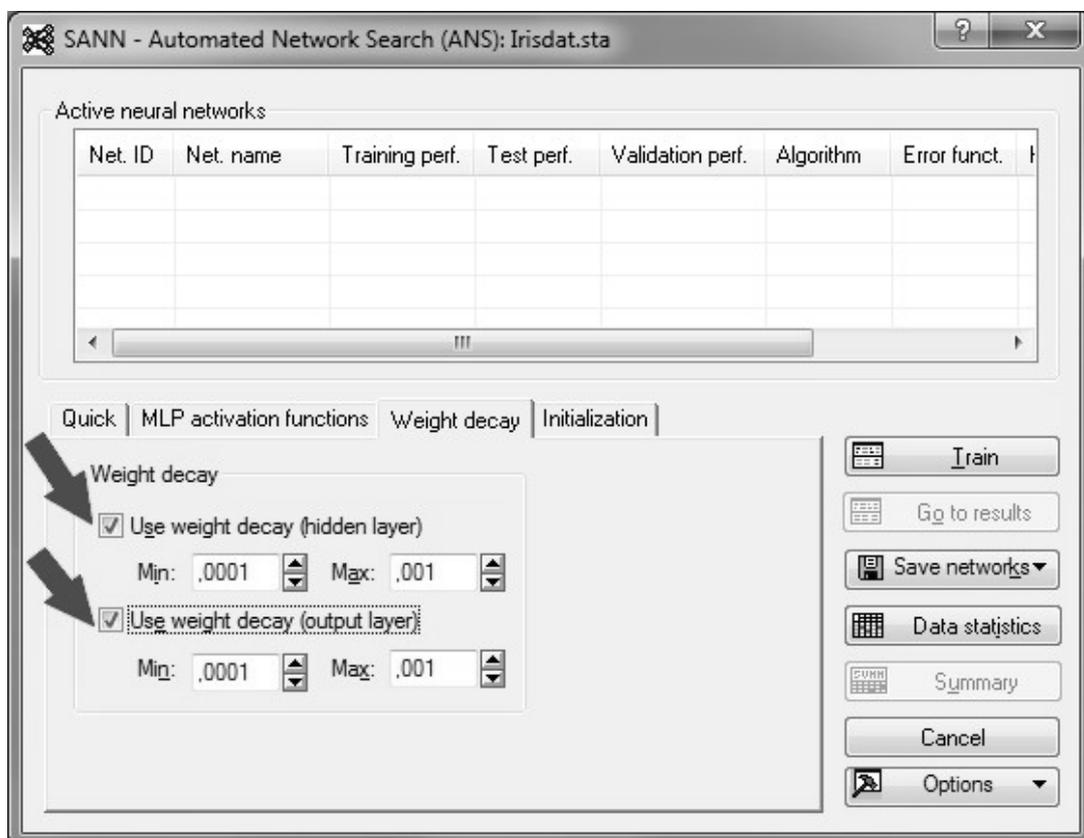


Рисунок 17 – Выбор нормы обучения для весов нейронов

SANN - Results: Irisdat.sta

Active neural networks

Net. ID	Net. name	Training perf.	Test perf.	Validation perf.	Algorithm	Error funct.
31	MLP 4-3-3	97,169811	100,000...	100,000000	BFGS 19	CE
32	MLP 4-7-3	93,396226	100,000...	100,000000	BFGS 12	SOS
33	MLP 4-5-3	96,226415	100,000...	100,000000	BFGS 35	CE
34	MLP 4-3-3	94,339622	100,000...	100,000000	BFGS 21	CE

Select/Deselect active networks Delete networks

Build models with CNN Build models with ANS Build models with Subsampling

Predictions Graphs Details Liftcharts Custom predictions

Target variable: IRISTYPE Case names (2-D and 3-D)

X-axis: SETOSA (conf.), VERSICOL (conf.), VIRGINIC (conf.), **SEPALLEN**, SEPALWID, PETALLEN, PETALWID

Y-axis: SETOSA (conf.), VERSICOL (conf.), VIRGINIC (conf.), SEPALLEN, SEPALWID, PETALLEN, **PETALWID**

Z-axis: Target, Output, Accuracy, SETOSA (conf.), VERSICOL (conf.), VIRGINIC (conf.), **SEPALLEN**

Histograms of X X and Y X, Y and Z Training error

Summary Save networks Cancel Options

Samples: Train, Test, Validation, Missing

Рисунок 18 – Результаты обучения сети

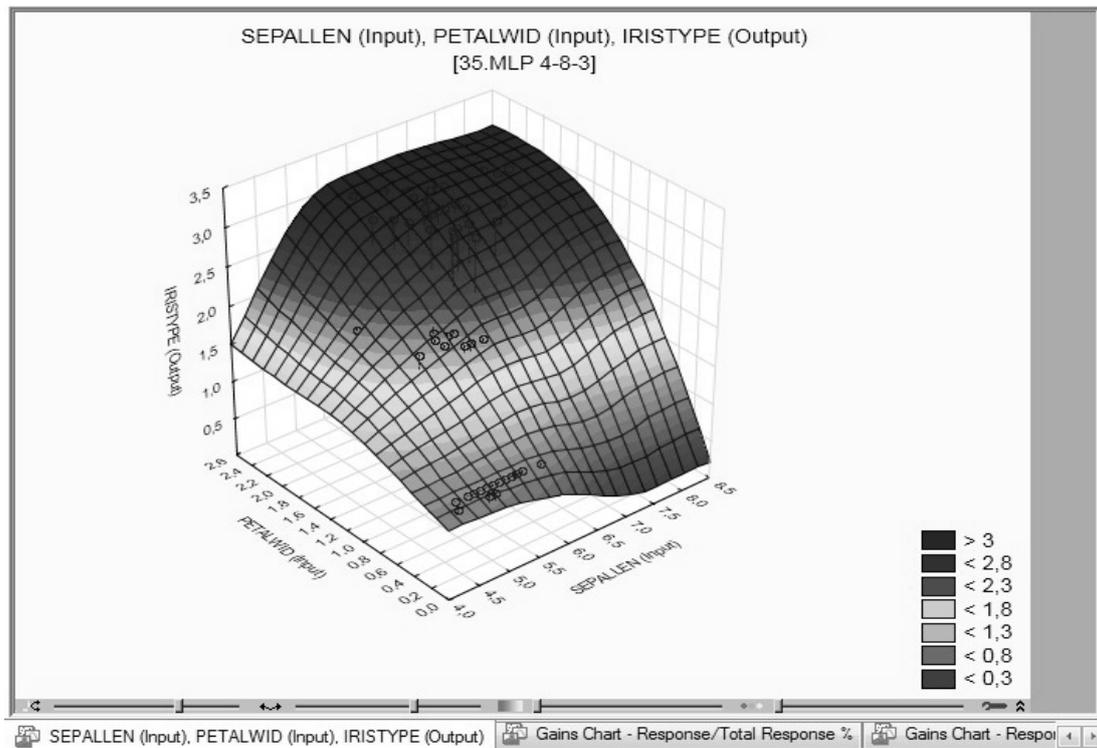


Рисунок 19 – 3D-графики выходных переменных

Дополнительно отобранные сети могут быть модифицированы с помощью клиентского конструктора нейронных сетей (закладка *Build models with CNN*) и в автоматическом режиме (закладка *Build models with ANN*). При использовании автоматического режима при обучении доступен алгоритм обратного распространения ошибки; при клиентском режиме можно выбрать методы обучения второго порядка в закладке *MLP* (рисунок 20).

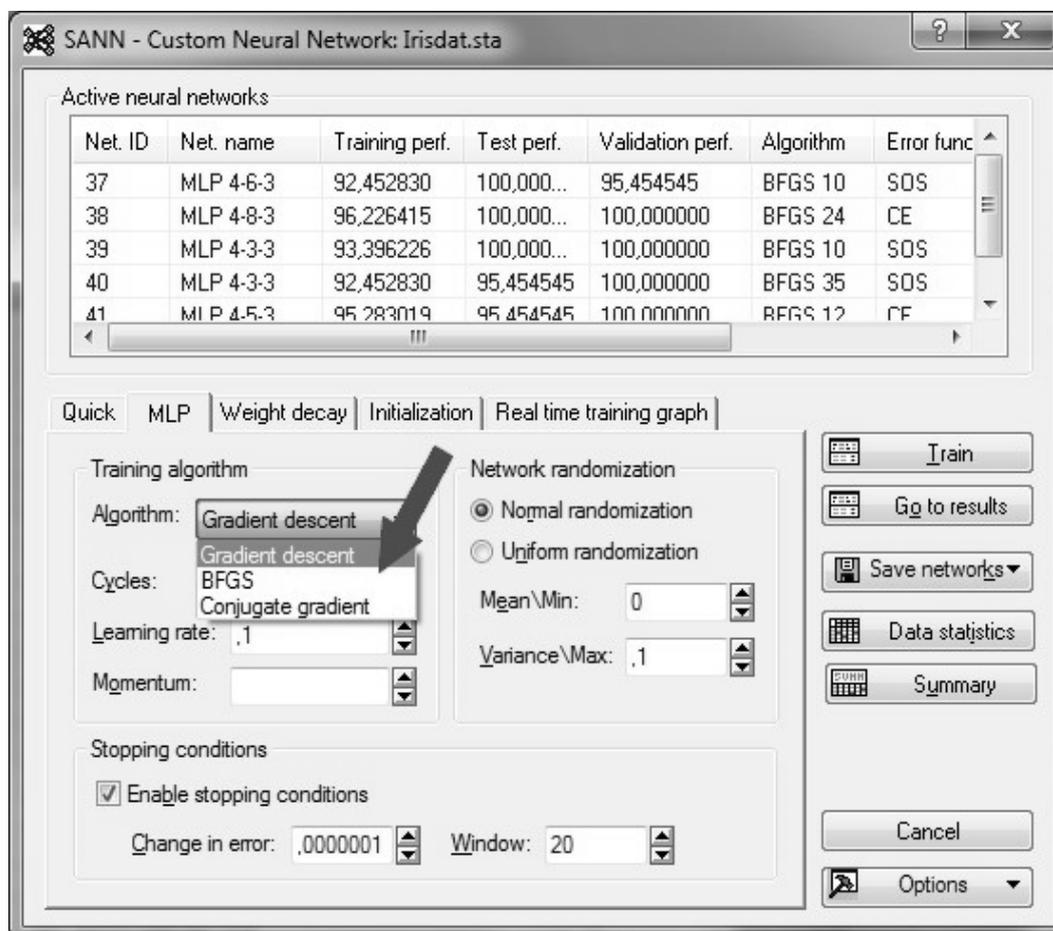


Рисунок 20 – Выбор метода обучения в режиме конструктора

Методы обучения представлены тремя вариантами: *Gradient descent* (метод градиентного спуска); *BFGS* (метод *Broyden-Fletcher-Goldfarb-Shanno*, или квази-ньютоновский метод); *Conjugate gradient* (метод сопряженных градиентов).

В этом же окне может быть выбрано условие останова обучения (*Stopping condition*) – по величине минимального изменения ошибки обучения (*Change in error*) в течение нескольких последних циклов (задается параметром *Window*).

На закладке *Real time training graph* может быть выбрано построение соответствующих графиков в режиме кросс-проверки (рисунок 21).

Пример таких графиков для обучения методом обратного распространения ошибки приведен на рисунке 22.

Обученная сеть может быть сохранена для дальнейшей работы с данными (для распознавания образов) – *Save networks* (рисунок 23).

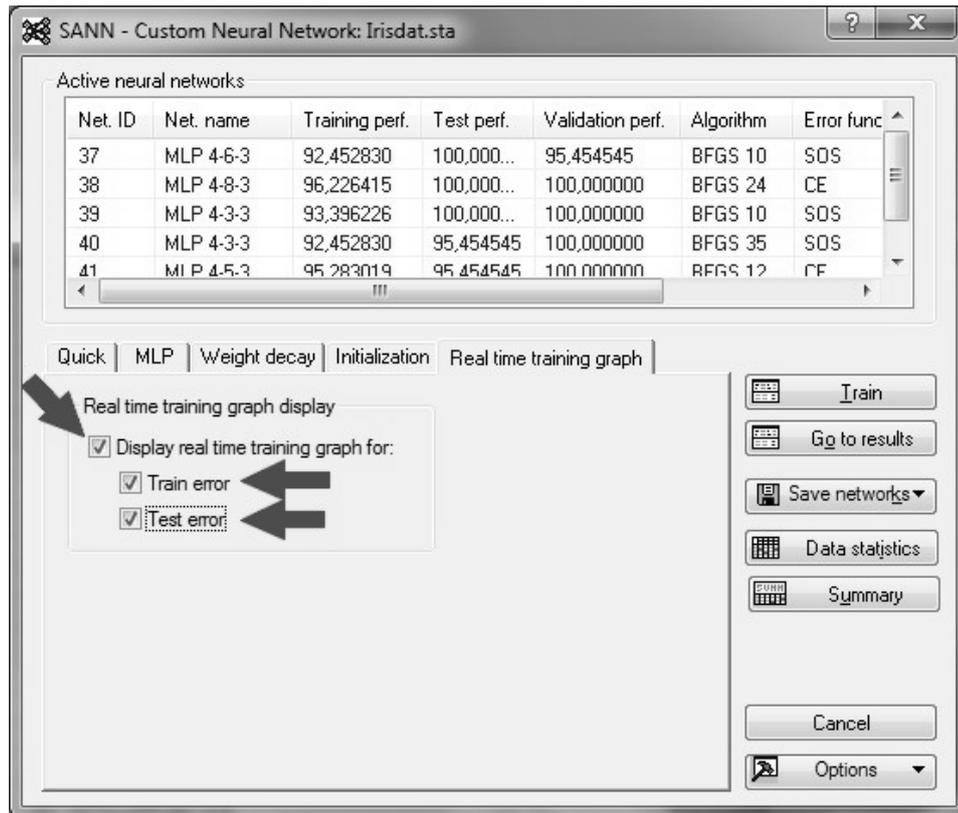


Рисунок 21 – Построение графиков ошибок обучения и кросс-проверки

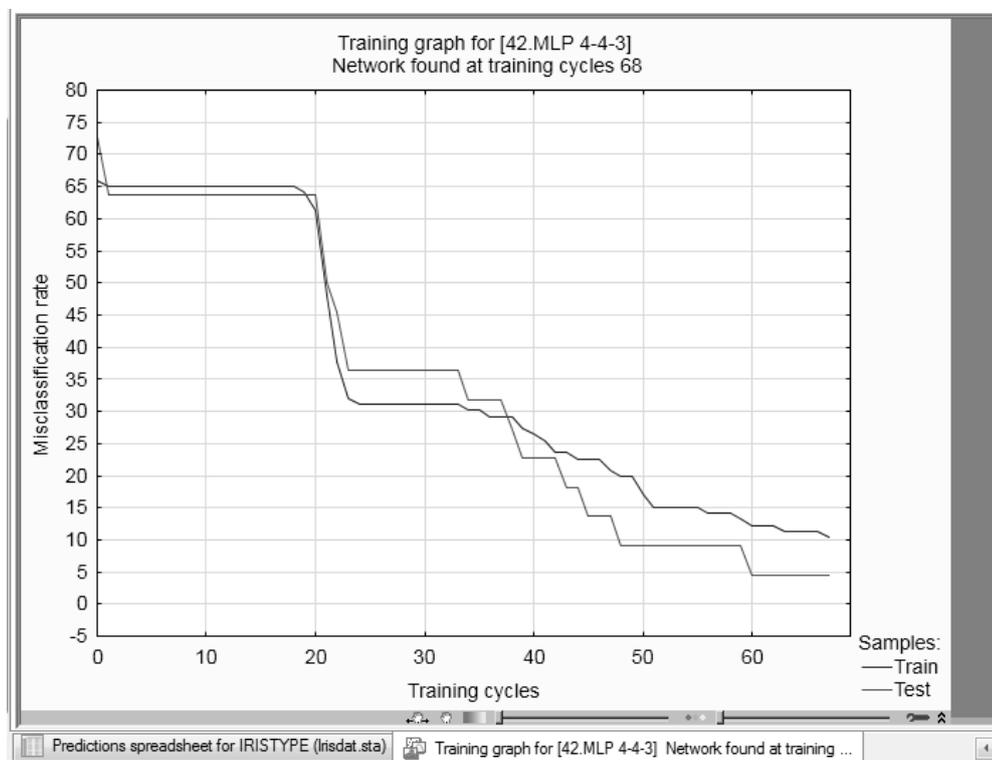


Рисунок 22 – График кросс-проверки при обучении сети

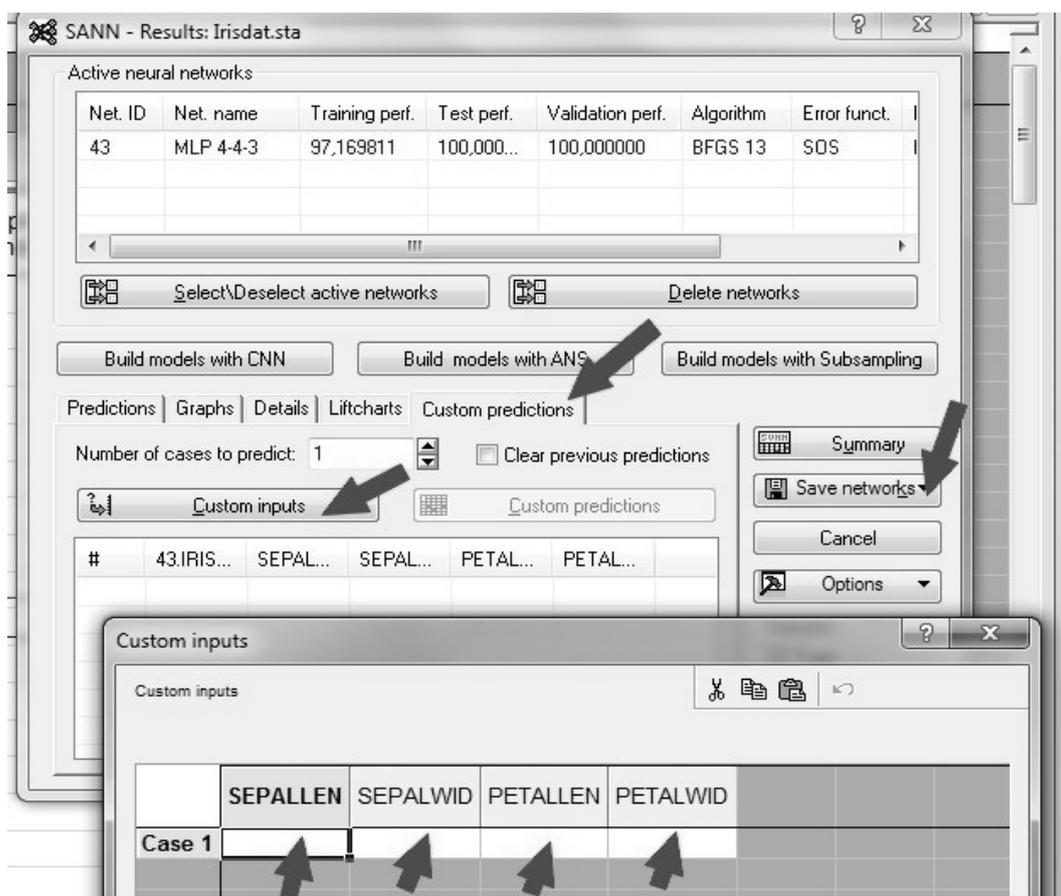


Рисунок 23 – Работа с обученной сетью

На закладке *Custom prediction* (см. рисунок 23) можно также использовать обученные сети для решения новых задач. Входные переменные вводятся при этом в соответствующие поля таблицы *Custom inputs*.

3 Провести исследования скорости и точности обучения методом обратного распространения ошибки (*Back Propagation*), а также влияния на точность распознавания числа слоев и количества нейронов. Использовать для каждого параметра несколько нейронных сетей, проанализировать различия в результатах обучения на отдельных сетях.

Провести оценку результативности обучения путем анализа графиков кросс-проверки.

4 Провести обучение методами градиентного спуска, сопряженных градиентов и квази-ньютоновским методом, сравнить их эффективность.

Вопросы к защите

1 Какие методы первого порядка применяются на практике; каковы их достоинства и недостатки?

2 Какие методы второго порядка применяются на практике; каковы их достоинства и недостатки?

3 Какие условия остановки обучения применяются на практике?

Список литературы

1 **Масленникова, О. Е.** Основы искусственного интеллекта: учебное пособие / О. Е. Масленникова, И. В. Гаврилова. – 2-е изд., стер. – Москва: ФЛИНТА, 2013. – 282 с.

2 **Осипов, Г. В.** Методы искусственного интеллекта / Г. В. Осипов. – Москва: Физматлит, 2011. – 296 с.

3 **Рутковская, Д.** Нейронные сети, генетические алгоритмы и нечеткие системы / Д. Рутковская, М. Пилиньский, Л. Рутковский. – Москва: Горячая линия-Телеком, 2007. – 452 с.

4 **Ярушкина, Н. Г.** Основы теории нечетких и гибридных систем: учебное пособие / Н. Г. Ярушкина. – Москва: Финансы и статистика, 2004. – 320 с.

5 Нейронные сети. STATISTICA Neural Networks: пер. с англ. – Москва: Горячая линия-Телеком, 2000. – 182 с.

6 **Биргер, И. А.** Техническая диагностика / И. А. Биргер. – Москва: Машиностроение, 1978. – 240 с.