

**Федеральное государственное бюджетное образовательное учреждение высшего
профессионального образования
«Владимирский государственный университет имени
Александра Григорьевича и Николая Григорьевича Столетовых»**

Р.И. МАКАРОВ

СПЕЦИАЛЬНЫЕ ГЛАВЫ МАТЕМАТИКИ

*Методические указания
к лабораторным занятиям*

Учебное электронное издание

ВЛАДИМИР 2012

Методические указания к лабораторным занятиям по дисциплине «Специальные главы математики» предназначены для подготовки магистров по направлению 230400 Информационные системы и технологии.

Дисциплина способствует формированию у обучаемых подходов к применению математических методов для решения специфических задач в области информационных систем и технологий.

Задания к практическим занятиям охватывают темы, изучаемые в первом учебном семестре: теория и методы принятия решений (2ч.), математические методы оптимизации (4ч.), математические основы нечетких систем (4ч.), нейронные сети для обработки информации (8ч.).

Задачами дисциплины являются:

- повысить уровень компетенции магистрантов за счет вооружения соответствующими знаниями и практическими навыками использования специальных глав математики для решения специфических задач в области информационных систем и информационных технологий;

- ознакомить магистрантов с методами разработки математических моделей на основе использования математического аппарата нечетких систем и нейронных сетей;

- изучить широкий круг вопросов, связанных с разработкой моделей предметных областей информационных систем.

Лабораторная работа №1

Оценка многокритериальных альтернатив

1.1 Цель работы

Ознакомление с методами принятия решений при многих критериях [1]. В случае небольшого числа заданных альтернатив усилия ЛПР направляются на сравнение заданных альтернатив с использованием Подхода Аналитической Иерархии (ПАИ). При подходе, направленном на Разработку Индексов Парного Сравнения Альтернатив (РИПСА), предпочтения ЛПР формируются при анализе проблемы, осуществляемой с помощью метода принятия решений. В работе рассматривается подход ПАИ в задачах принятия решений.

1.2 Теоретические сведения

Подход ПАИ состоит из следующих этапов:

- 1) Структуризация задачи в виде иерархической структуры;
- 2) Парные сравнения элементов каждого уровня. Перевод результатов сравнения в числа при помощи специальной таблицы;
- 3) Вычисление коэффициентов важности для элементов каждого уровня. Проверка согласованности суждений ЛПР;
- 4) Подсчет количественных индикаторов качества каждой из альтернатив и определение наилучшей альтернативы.

Пример. Выбрать наилучшую альтернативу с помощью метода аналитической иерархии, разработанного Т. Саати [1].

Дана общая цель – строительство аэропорта во Владимире.

Имеются четыре альтернативы A, B, C, D расположения площадок.

Альтернативы оцениваются с помощью трех критериев – C_1 стоимость строительства, C_2 время в пути от аэропорта до центра города, C_3 количество людей, подвергающихся шумовому воздействию.

Оценки вариантов по критериям: A(180, 70, 10), B(170, 40, 15), C(160, 55, 20), D(150, 50, 25).

Для парных сравнений в распоряжении ЛПР дается шкала словесных определений уровня важности (таблица 1):

Таблица 1.1- Шкала относительной важности

| Уровень важности | Количественное значение |
|--|-------------------------|
| Равная важность | 1 |
| Умеренное превосходство | 3 |
| Существенное или сильное превосходство | 5 |
| Значительное (большое) превосходство | 7 |
| Очень большое превосходство | 9 |

Структура задачи может быть представлена в виде, показанном на рисунке 1.1.

Парные сравнения элементов каждого уровня проводится ЛПР с использованием приведенные в таблице 1.1 определения. Матрица сравнений выбора места расположения

площадок приведена в таблице 1.2. Верхний треугольник матрицы заполняется по данным таблицы 1.2 с учетом относительной важности критериев строки над столбцами. Нижний треугольник матрицы заполняется симметрично относительно главной диагонали обратными величинами.

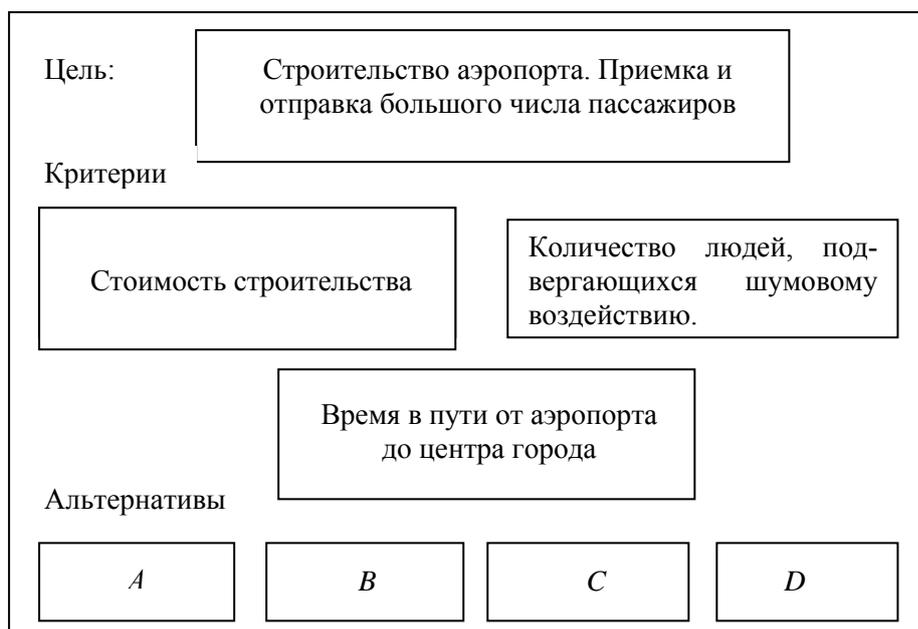


Рисунок 1.1- Структуризация задачи в виде иерархической структуры

Таблица 1.2- Матрица сравнений для критериев

| Критерий | C_1 | C_2 | C_3 | Собственный вектор | Вес критерия w_i |
|----------|-------|-------|-------|--------------------|--------------------|
| C_1 | 1 | 5 | 3 | 2,47 | 0,65 |
| C_2 | 1/5 | 1 | 3 | 0,84 | 0,22 |
| C_3 | 1/3 | 1/3 | 1 | 0,48 | 0,13 |

Собственные вектора матрицы вычисляются по формуле: извлекается корень n -ой степени из произведения элементов каждой строки. Веса критериев определяются нормировкой собственных векторов.

На нижнем уровне иерархической схемы (рис.1.1) сравниваются альтернативы по каждому критерию в отдельности. Результаты сравнения приведены в таблицах 1.3 – 1.5.

Для определения наилучшей альтернативы вычисляются показатели качества по формуле:

$$V_j = \sum_{i=1}^n w_i V_{ji} \quad (1.1)$$

$$V_1=0,65*0,04+0,22*0,05+0,13*0,56=0,11; V_2=0,215; V_3=0,241; V_4=0,431.$$

Таблица 1.3-Сравнение по критерию C_1

| Критерий | A | B | C | D | Собственный вектор | Вес критерия V_{1i} |
|----------|---|-----|-----|-----|--------------------|-----------------------|
| A | 1 | 1/5 | 1/7 | 1/9 | 0,23 | 0,04 |
| B | 5 | 1 | 1/3 | 1/5 | 0,76 | 0,13 |
| C | 7 | 3 | 1 | 1/3 | 1,63 | 0,27 |
| D | 9 | 5 | 3 | 1 | 3,4 | 0,56 |

Таблица 1.4-Сравнение по критерию C_2

| Критерий | A | B | C | D | Собственный вектор | Вес критерия V_{2i} |
|----------|---|-----|-----|-----|--------------------|-----------------------|
| A | 1 | 1/9 | 1/5 | 1/7 | 0,23 | 0,05 |
| B | 9 | 1 | 3 | 1 | 2,28 | 0,43 |
| C | 5 | 1/3 | 1 | 1 | 1,14 | 0,22 |
| D | 7 | 1 | 1 | 1 | 1,63 | 0,3 |

Таблица 1.5-Сравнение по критерию C_3

| Критерий | A | B | C | D | Собственный вектор | Вес критерия V_{3i} |
|----------|-----|-----|-----|---|--------------------|-----------------------|
| A | 1 | 3 | 5 | 9 | 3,4 | 0,56 |
| B | 1/3 | 1 | 3 | 7 | 1,63 | 0,27 |
| C | 1/5 | 1/3 | 1 | 5 | 0,76 | 0,13 |
| D | 1/9 | 1/7 | 1/5 | 1 | 0,23 | 0,04 |

Наилучшей является альтернатива D -четвертая, имеющая наибольшее значение показателя качества $V_4=0,431$.

1.3 Задание к лабораторной работе

Выбрать организацию работ по проектированию информационной системы с помощью метода аналитической иерархии, разработанного Т. Саати [1].

Дана общая цель – организация процессов разработки проекта информационной системы.

Имеются несколько типовых схем организации работ [2]: А- в одном лице выступают заказчик, разработчик и администратор, Б- функции разработчика отделяются от функций заказчика и администратора и выполняются другой организацией-разработчиком, В- заказчик привлекает фирму проектной интеграции.

Альтернативы оцениваются с помощью четырех критериев – C_1 стоимость разработки, C_2 продолжительность разработки, C_3 участие заказчика в разработке, C_4 использование аутсорсинга.

Для парных сравнений в распоряжении ЛПР дается шкала словесных определений уровня важности (таблица 1.6):

Оценки вариантов по критериям: указаны в индивидуальном задании (таблица 1.7-1.9)

Таблица 1.6 - Шкала относительной важности

| Уровень важности | Количественное значение |
|--|-------------------------|
| Равная важность | 1 |
| Умеренное превосходство | 3 |
| Существенное или сильное превосходство | 5 |
| Значительное (большое) превосходство | 7 |
| Очень большое превосходство | 9 |

1.4 Порядок выполнения лабораторной работы

- 1) Получить от преподавателя индивидуальное задание.
- 2) Ознакомиться с организацией работ по проектированию информационных систем по учебному пособию [2, стр. 277-285].
- 3) Выполнить структуризацию задачи в виде иерархической структуры;
- 4) Парно сравнить элементы каждого уровня. Перевести результаты сравнения в числа при помощи таблицы 1.6;
- 5) Вычислить коэффициенты важности для элементов каждого уровня.
- 6) Подсчитать количественные индикаторы качества каждой из альтернатив и определить наилучшую альтернативу.

1.5 Содержание отчета

- 1) Индивидуальное задание на лабораторную работу;
- 2) Структуризация задачи в виде иерархической структуры;
- 3) Матрица сравнений для критериев;
- 4) Таблицы сравнений по каждому критерию ;
- 5) Результаты вычисления показателей качества по каждой альтернативе;
- 6) Определения наилучшей альтернативы;
- 7) Выводы по работе.

1.6 Вопросы для самоконтроля

- 1) Основные этапы подхода аналитической иерархии;
- 2) Построение матрицы сравнений;
- 3) Вычисление коэффициентов важности;
- 4) Выбор лучшей альтернативы.

1.7 Список рекомендуемой литературы

- 1) Ларичев О.И. Теория и методы принятия решений, а также хроника событий в Волшебных странах: Учебник. Изд. второе, перераб. и доп. –М.: Логос, 2002. -392 с.-*ISBN 5-94010-180-1*.
- 2) Макаров Р.И. Методология проектирования информационных систем: учеб. пособие / Р.И. Макаров, Е.Р. Хорошева; Владим. гос. ун-т. Владимир: Изд-во Владим. гос. ун-та, 2008.-334с.- *ISBN 978-5-89368-817-7*.

1.8 Варианты заданий к лабораторной работе

Задание 1. Разработка информационной системы небольшой по стоимости и продолжительности

Таблица 1.7-Оценки разработки ИС небольшой по стоимости и продолжительности

| Организация работ по проектированию ИС | C_1 -стоимость разработки, тыс. руб. | C_2 -продолжительность разработки, месяцы | C_3 -участие заказчика в разработке,% | C_4 -использование аутсорсинга, % |
|---|--|---|---|-------------------------------------|
| А-силами заказчика | 150 | 12 | 100 | 0 |
| Б-с привлечением разработчика | 250 | 11 | 30 | 50 |
| В-с привлечением проектного интегратора | 200 | 10 | 50 | 100 |

Задание 2. Разработка большой и сложной информационной системы

Таблица 1.8-Оценки разработки большой и сложной ИС

| Организация работ по проектированию ИС | C_1 -стоимость разработки, тыс. руб. | C_2 -продолжительность разработки, месяцы | C_3 -участие заказчика в разработке,% | C_4 -использование аутсорсинга, % |
|---|--|---|---|-------------------------------------|
| А-силами заказчика | 300 | 32 | 100 | 0 |
| Б-с привлечением разработчика | 450 | 24 | 20 | 50 |
| В-с привлечением проектного интегратора | 400 | 18 | 50 | 100 |

Задание 3. Разработка большой и сложной ИС, система должна как можно быстрее дать экономический эффект

Таблица 1.9-Оценки разработки большой и сложной ИС, чтобы система как можно быстрее дала экономический эффект

| Организация работ по проектированию ИС | C_1 -стоимость разработки, тыс. руб. | C_2 -продолжительность разработки, месяцы | C_3 -участие заказчика в разработке,% | C_4 -использование аутсорсинга, % |
|---|--|---|---|-------------------------------------|
| А-силами заказчика | 300 | 32 | 100 | 0 |
| Б-с привлечением разработчика | 450 | 24 | 20 | 50 |
| В-с привлечением проектного интегратора | 450 | 16 | 50 | 100 |

Лабораторная работа №2

Математические методы оптимизации

2.1 Цель работы

Ознакомление магистрантов с численными методами многомерной оптимизации в программе *EXCEL*. Встроенная программа ПОИСК РЕШЕНИЯ позволяет решать многомерные оптимизационные задачи безусловной и условной оптимизации.

Решение задач линейного программирования симплекс-методом, дискретного программирования методом ветвей и границ, нелинейного программирования рассмотрено в методическом указании к практическим занятиям по дисциплине «Математические методы теории систем» [2].

2.2 Теоретические сведения

Численные методы оптимизации, численный эксперимент считаются важным разделом системного анализа [1]. Численные методы представляют собой последовательность однотипных шагов, итераций. Выделяют конечные и бесконечные численные методы.

Конечные методы позволяют получать решение за конечное, но обычно заранее неизвестное, число шагов (симплекс-метод, метод наискорейшего спуска, методы условной минимизации и др.).

Бесконечные методы основаны на построении последовательности все более точных приближений к решению. При этом, прекращая процесс на том или ином шаге итераций, можно получить решение с заданной точностью. К этой группе методов относят: метод Брауна - Робинсона, метод штрафных функций, или метод штрафов.

В современных условиях при использовании ЭВМ число методов расширяется. Существуют методы оптимизации, основанные на идее целенаправленного поиска минимума. При этом возможно задание ограничений на изменение независимых переменных.

При отсутствии ограничений используются методы безусловной оптимизации. Для решения таких задач алгоритм предусматривает последовательный поиск минимума функции одного переменного, при фиксированных значениях остальных. После проведения такого поиска для всех переменных осуществляется следующая итерация, при которой уточняется полученное значение. В программе ПОИСК РЕШЕНИЯ предусмотрен поиск по двум методам – методу сопряженных градиентов и методу Ньютона.

Пример. Пусть надо найти минимум функции $f(x_1, x_2) = (x_2 - x_1^2)^2 + (1 - x_1)^2$, ограничения не заданы. Ячейки A5 и B5 выделяем под искомые переменные x_1 и x_2 . В ячейку C5 записываем значение целевой функции $= (B5 - A5^2)^2 + (1 - A5)^2$

Выбрав в меню *EXCEL* Данные / Поиск решения задаем следующие параметры:

- Установить целевую ячейку: C5
- Равной: минимальному значению
- Изменяя ячейки: A5:B5
- В меню параметры можно указать:

максимальное время: 100 секунд; предельное число итераций: 100; относительная погрешность: 0,000001; допустимое отклонение: 5%; сходимость: 0,0001; оценки: квадратичная; разности: прямые; метод поиска: сопряженных градиентов.

Результат решения:(1;1). Задача решается аналитически, точное решение совпадает с численным решением.

Для решения задач условной оптимизации можно использовать встроенную программу EXCEL. Поиск успешно работает для некоторых видов ограничений: вида шара, координатного параллелепипеда, гиперплоскости.

Пример. Пусть надо найти минимум функции из предыдущего примера $f(x_1, x_2) = (x_2 - x_1^2)^2 + (1 - x_1)^2$. Ограничения заданы в виде функции $z(x_1, x_2) = 0,68 - (x_1^2 + x_2^2)$. В данном случае выделяем ячейки A6 и B6 под переменные x_1, x_2 . В ячейку C6 записываем значение целевой функции $=(B5-A5^2)^2+(1-A5)^2$. В ячейку D6- формулу ограничений $=0,68-(A5^2+B5^2)$. Далее совершаются те же действия, что и при безусловной многомерной оптимизации, только необходимо дополнительно установить ограничение: D6=0.

Для вывода отчетов с решениями в подменю «Результаты поиска решений» нажать кнопку «Сохранить найденное решение» и указать тип отчета: Результаты, Устойчивость, Пределы. Появятся листы EXCEL с тремя типами отчетов:

Microsoft Excel 12.0 Отчет по результатам

Рабочий лист: [Численные методы.xls]Условная оптимизация

Отчет создан: 05.07.2012 18:08:09

Целевая ячейка (Минимум)

| Ячейка | Имя | Исходное значение | Результат |
|--------|-----|-------------------|-------------|
| \$C\$6 | | 1 | 0,091214172 |

Изменяемые ячейки

| Ячейка | Имя | Исходное значение | Результат |
|--------|-----|-------------------|-------------|
| \$A\$6 | | 0 | 0,71223007 |
| \$B\$6 | | 0 | 0,415605762 |

Ограничения

| Ячейка | Имя | Значение | Формула | Статус | Разница |
|--------|-----|-------------|----------|------------|---------|
| \$D\$6 | | 1,77612E-07 | \$D\$6=0 | не связан. | 0 |

Microsoft Excel 12.0 Отчет по устойчивости

Рабочий лист: [Численные методы.xls]Условная оптимизация

Отчет создан: 05.07.2012 18:09:45

Изменяемые ячейки

| Ячейка | Имя | Результ. значение | Нормир. градиент |
|--------|-----|-------------------|------------------|
| \$A\$6 | | 0,71223007 | 0 |
| \$B\$6 | | 0,415605762 | 0 |

Ограничения

| Ячейка | Имя | Результ. значение | Лагранжа Множитель |
|--------|-----|-------------------|--------------------|
| \$D\$6 | | 1,77612E-07 | 1 |

Microsoft Excel 12.0 Отчет по пределам

Рабочий лист: [Численные методы.xls]Отчет по пределам 1

Отчет создан: 05.07.2012 18:09:45

| Целевое | | | | | | |
|------------|-----|-------------|---------------|-------------------|----------------|-------------------|
| Ячейка | Имя | Значение | | | | |
| \$D\$6 | | 1,77612E-07 | | | | |
| Изменяемое | | | Нижний предел | Целевой результат | Верхний предел | Целевой результат |
| Ячейка | Имя | Значение | | | | |
| \$A\$6 | | 0,71223007 | 0,71223007 | 1,7761E-07 | 0,71223007 | 1,7761E-07 |
| \$B\$6 | | 0,415605762 | 0,415605762 | 1,7761E-07 | 0,415605762 | 1,7761E-07 |

Метод штрафных функций, или метод штрафов, сводит решение к задаче оптимизации без ограничений.

Пример. Пусть надо найти минимум функции из предыдущего примера $f(x_1, x_2) = (x_2 - x_1)^2 + (1 - x_1)^2$. Ограничение задано в виде функции штрафа:

$$G(x_1) = \begin{cases} \infty, & \text{если } x_1 > 0,8 \\ 0, & \text{если } x_1 \leq 0,8 \end{cases}$$

Запишем функцию цели в виде: $z(x_1, x_2) = f(x_1, x_2) + G(x_1)$

Решение задачи без оптимизации $z(x_1, x_2)$ является решением задачи условной оптимизации. Ячейки A4 и B4 выделяем под искомые переменные x_1 и x_2 . В ячейку C4 записываем значение целевой функции: $=(B4-A4^2)^2+(1-A4)^2+ЕСЛИ(A4>0,8;1000;0)$

Выбрав в меню EXCEL Данные / Поиск решения задаем следующие параметры:

- Установить целевую ячейку: C4
- Равной: минимальному значению
- Изменяя ячейки: A4:B4
- В меню параметры можно указать:

максимальное время: 100 секунд; предельное число итераций: 100; относительная погрешность: 0,000001; допустимое отклонение: 5%; сходимость: 0,0001; оценки: квадратичная; разности: прямые; метод поиска: сопряженных градиентов.

Для вывода отчетов с решениями в подменю «Результаты поиска решений» нажать кнопку «Сохранить найденное решение» и указать тип отчета: Результаты, Устойчивость, Пределы. Появятся листы EXCEL с тремя типами отчетов:

Microsoft Excel 12.0 Отчет по результатам

Рабочий лист: [Численные методы.xls]Штрафная функция

Отчет создан: 05.07.2012 19:03:53

Целевая ячейка (Минимум)

| Ячейка | Имя | Исходное значение | Результат |
|--------|-----|-------------------|-------------|
| \$C\$4 | | 1 | 0,040619445 |

Изменяемые ячейки

| Ячейка | Имя | Исходное значение | Результат |
|--------|--|-------------------|-------------|
| \$A\$4 | Z=(x2-x1)^2+(1-x1)^2+ЕСЛИ(A4>0,8;1000;0) | 0 | 0,799999435 |
| \$B\$4 | | 0 | 0,824883552 |

Ограничения

НЕТ

Microsoft Excel 12.0 Отчет по устойчивости

Рабочий лист: [Численные методы.xls]Штрафная функция

Отчет создан: 05.07.2012 19:03:53

Изменяемые ячейки

| Ячейка | Имя | Результ. значение | Нормир. градиент |
|--------|---|-------------------|------------------|
| | $Z=(x_2-x_1)^2+(1-x_1)^2+\text{ЕСЛИ}(A_4>0,8;1000;0)$ | 0,799999435 | 0 |
| \$B\$4 | | 0,824883552 | 0 |

Ограничения

НЕТ

Microsoft Excel 12.0 Отчет по пределам

Рабочий лист: [Численные методы.xls]Отчет по пределам 2

Отчет создан: 05.07.2012 19:03:53

| Целевое | | | | | | |
|------------|---|-----------|---------------|-------------------|----------------|-------------------|
| Ячейка | Имя | Значение | | | | |
| \$C\$4 | | 0,040619 | | | | |
| Изменяемое | | | Нижний предел | Целевой результат | Верхний предел | Целевой результат |
| Ячейка | Имя | Значение | | | | |
| | $Z=(x_2-x_1)^2+(1-x_1)^2+\text{ЕСЛИ}(A_4>0,8;1000;0)$ | 0,7999994 | | | | |
| \$A\$4 | | 35 | #Н/Д | #Н/Д | #Н/Д | #Н/Д |
| \$B\$4 | | 0,824883 | #Н/Д | #Н/Д | #Н/Д | #Н/Д |

2.3 Задание к лабораторной работе

Решить численным методом задачу многомерной оптимизации в программе *EXCEL* при различных ограничениях на изменения независимых переменных: при отсутствии ограничений, при сложных видах ограничений. Построить графики – контурные кривые для нескольких значений целевой функции и ограничений. Составить отчеты в электронной форме и сделать выводы по решенной задаче.

2.4 Порядок выполнения лабораторной работы

- 1) Получить от преподавателя индивидуальное задание.
- 2) Ознакомиться с численными методами одномерной и многомерной оптимизации.
- 3) Решить задачу многомерной оптимизации, вывести отчеты в электронной форме.

4) Построить графики – контурные кривые для нескольких значений целевой функции и ограничений.

5) Провести анализ результатов решений, сделать выводы по работе.

2.5 Содержание отчета

1) Индивидуальное задание на лабораторную работу;

2) Графики контурных кривых для нескольких значений целевой функции и ограничений.

3) Отчеты программы *EXCEL* с результатами решения задачи при различных ограничениях на независимые переменные;

4) Анализ результатов решения задачи оптимизации при различных ограничениях на независимые переменные;

5) Выводы по работе.

2.6 Вопросы для самоконтроля

1) Конечные и бесконечные численные методы поиска оптимума многомерной оптимизации.

2) Многомерные задачи оптимизации. Безусловная оптимизация.

3) Многомерные задачи оптимизации. Условная оптимизация.

4) Многомерные задачи оптимизации. Метод штрафных функций.

5) Многомерные задачи оптимизации. Поиск экстремума методом сопряженных градиентов.

6) Многомерные задачи оптимизации. Поиск по методу Ньютона.

2.7 Список рекомендуемой литературы

1) Антонов А.В. Системный анализ. Учеб. для вузов. -М.: Высш. шк., 2004. -454с.- ISBN 5-06-004862-4.

2) Математические методы теории систем. Методические указания к практическим занятиям (часть 2-я). / Д.И. Левковский, Р.И. Макаров; Владимир. гос. ун-т. Владимир: Изд-во Владим. гос. ун-та, 2009.

2.8 Варианты заданий к лабораторной работе

Задание 1. Решить многомерную задачу оптимизации. Отыскать минимальное значение целевой функции $f(x_1, x_2) = x_1^2 + 0,5x_2^2$. Контурные кривые целевой функции представляют эллипсы. На изменения независимых переменных могут накладываться ограничения.

Рассмотреть два случая:

- на изменения независимых переменных не накладываются ограничения;

- на изменения независимых переменных накладываются ограничения

$$0,1 \leq x_1 \leq 1; 0,1 \leq x_2 \leq 1.$$

Задание 2. Решить многомерную задачу оптимизации. Отыскать минимальное значение целевой функции $f(x_1, x_2) = x_1^2 - 0,5x_2^2$. Контурные кривые целевой функции представляют гиперболы, вытянутые по оси x_2 . На изменения независимых переменных могут накладываться ограничения.

Рассмотреть два случая:

- на изменения независимых переменных не накладываются ограничения;
- на изменения независимой переменной x_1 накладываются ограничения $x_1 \geq 1,2$.

Задание 3. Решить многомерную задачу оптимизации. Отыскать минимальное и максимальное значения целевой функции $f(x_1, x_2) = 1 - 4x_1^2 + 10x_2$. Контурные кривые целевой функции представляют возрастающее возвышение вдоль оси x_2 . На изменения независимых переменных накладываются ограничения:

$$0 \leq x_1 \leq 1; 0 \leq x_2 \leq 0,5.$$

Задание 4. Решить многомерную задачу оптимизации. Отыскать минимальное значение целевой функции $f(x_1, x_2) = (x_2 - x_1)^2 + (1 - x_1)^2$. На изменения независимой переменной x_1 накладываются ограничения вида штрафной функции: $G(x_1) = \text{ЕСЛИ}(x_1 > 0,8; \infty; 0)$. Вместо знака ∞ в условии записывать большое число, например 1000.

Рассмотреть два случая:

- на изменения независимых переменных не накладываются ограничения;
- на изменения независимой переменной x_1 накладывается штрафная функция $G(x_1) = \text{ЕСЛИ}(x_1 > 0,8; 1000; 0)$. При решении задачи целевая функция должна записываться в виде:

$$z = f(x_1, x_2) + G(x_1) = (x_2 - x_1)^2 + (1 - x_1)^2 + \text{ЕСЛИ}(x_1 > 0,8; 1000; 0).$$

Задание 5. Решить многомерную задачу оптимизации. Отыскать минимальное значение целевой функции $f(x_1, x_2) = 0,5 + x_1^2 - x_2$. На изменения независимых переменных x_1 и x_2 накладывается ограничение: $x_1^2 + x_2^2 = 1$. Ограничения задать в виде штрафной функции:

$$G(x_1, x_2) = \text{ЕСЛИ}((x_1^2 + x_2^2) = 1; 0; \infty).$$

Вместо знака ∞ в условие записывать большое число, например 1000.

Рассмотреть два случая:

- на изменения независимых переменных не накладываются ограничения;
- на изменения независимых переменных x_1 и x_2 накладывается штрафная функция $G(x_1, x_2) = \text{ЕСЛИ}((x_1^2 + x_2^2) = 1; 0; 1000)$.

При решении задачи целевая функция должна записываться в виде:

$$z = f(x_1, x_2) + G(x_1, x_2) = 0,5 + x_1^2 - x_2 + \text{ЕСЛИ}((x_1^2 + x_2^2) = 1; 0; 1000).$$

Лабораторная работа №3

Построение нечетких моделей

3.1 Цель работы

Ознакомление магистрантов с методами разработки математических моделей на основе использования математического аппарата нечетких систем в программе *MATLAB 7.0.1*

3.2 Теоретические сведения

В *MATLAB* важная роль отводится специализированным группам программ, называемых *toolboxes*. *Toolboxes* – это всесторонняя коллекция функций *MATLAB* (*M*-файлов), которые позволяют решать частные классы задач [2].

Пакет прикладных программ *Fuzzy Logic* относится к теории нечетких (размытых) множеств. Обеспечивается поддержка современных методов нечеткой кластеризации и адаптивных нечетких нейронных сетей. Графические средства пакета позволяют интерактивно отслеживать особенности поведения системы.

Основные возможности пакета:

- 1) определение переменных, нечетких правил и функций принадлежности;
- 2) интерактивный просмотр нечеткого логического вывода;
- 3) современные методы: адаптивный нечеткий вывод с использованием нейронных сетей, нечеткая кластеризация;
- 4) интерактивное динамическое моделирование в *Simulink*.

Создание *m*-функции для моделирования. *M*-файлы являются обычными текстовыми файлами, которые создаются с помощью текстового редактора. Для операционной среды персонального компьютера система *MATLAB* поддерживает специальный встроенный редактор/отладчик, хотя можно использовать и любой другой текстовый редактор с *ASCII*-кодами.

M-функции являются *M*-файлами, которые допускают наличие входных и выходных аргументов. Они работают с переменными в пределах *собственной рабочей области*, отличной от рабочей области системы *MATLAB*.

Построение нечётких моделей слабо формализовано. В каждом случае применяются свои параметры построения модели. В методе построения модели с использованием *FCM*-кластеризации можно управлять размерами выборки (количество наблюдений в выборке, количество входных переменных), используемой для построения модели, а также видом модели (Сугэно / Мамдани).

Одним из недостатков нечётких моделей является эффект «переобучения», когда модель даёт минимальную ошибку на элементах обучающего

множества при большой ошибке на элементах тестирующего множества. Это означает, что модель «запомнила» обучающие наборы вместо того, чтобы научиться решать поставленную задачу. Для преодоления этого недостатка применяют разбиение исходной выборки на два подмножества: обучающую и проверочную. Обучающая используется для построения и настройки модели, проверочная – для определения пригодности модели. Большие проблемы могут возникнуть при использовании большого числа входных переменных. При этом образуется огромное число внутренних элементов модели, сама модель становится относительно более медленной, затрудняется или становится вообще невозможным (при имеющихся технических мощностях) обучение модели.

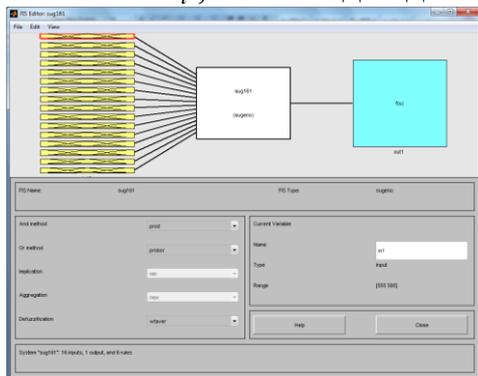
Пакет *MATLAB Fuzzy Logic Toolbox* позволяет строить модели тремя различными способами:

- 1) *genfis1* – построение нечёткой модели типа Сугэно, которую далее необходимо обучить с использованием *anfis*-обучения;
- 2) *genfis2* – генерация нечёткой модели с использованием алгоритма субтрактивной кластеризации;
- 3) *genfis3* – генерация нечёткой модели с использованием нечёткой (*FCM*) кластеризации.

Выбор способа построения нечеткой модели, определение параметров (отбор информативных входных переменных, размер обучающей выборки, величина радиуса центра кластеризации) для построения качественной модели должен проводиться путём постановки вычислительного эксперимента со статистическими данными, используемыми для построения модели [3, 4]. В качестве оценки адекватности модели можно использовать значение коэффициента парной корреляции, который будет показывать, насколько достоверно нечёткая модель моделирует выходную переменную.

При помощи инструмента *Fuzzy Logic* можно посмотреть сформированную структуру модели и нечеткие правила.

На рисунке 3.1 видна структура модели. На вход модели поступают 16 значений x_i , на выходе одно значение y .



Можно посмотреть сформированные нечеткие переменные и функции принадлежности (Рисунок 3.2). При этом заданы минимальные и максимальные значения для x_i , названия функций и переменных. Все функции принадлежности по умолчанию являются симметричными гауссовскими кривыми (*gaussmf*).

Рисунок 3.1 – Структура нечеткой модели

Также можно просмотреть сформированный набор правил (Рисунок 3.3). Видно, что все правила представляются видом:

Если (Выражение1) И (Выражение2)...И (Выражение16) то (Значение 1)(Значение 2)...(Значение m).

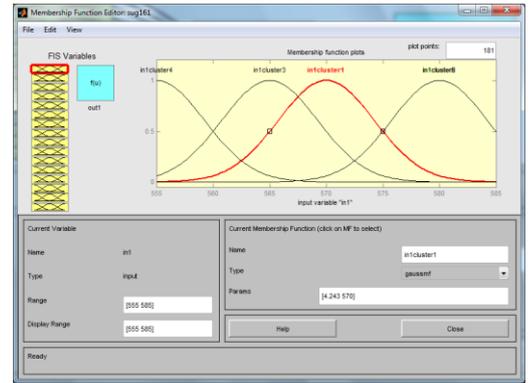


Рисунок 3.2 – Функции принадлежности нечеткой переменной x_1

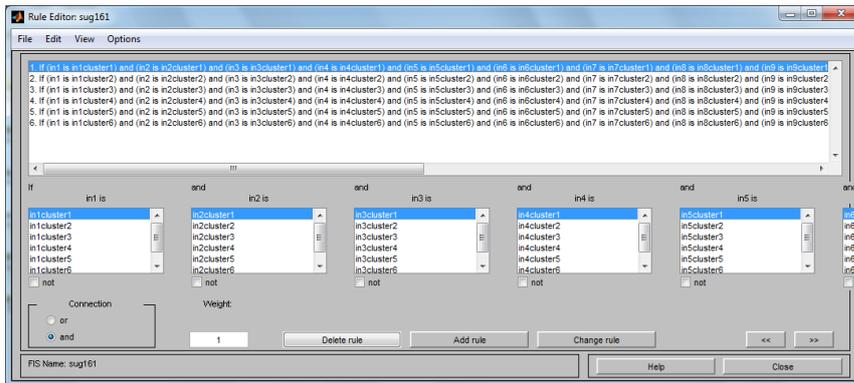


Рисунок 3.3 – Правила построения нечеткой модели

Пример. Имеются статистические данные функционирования многомерного объекта управления с семью входными переменными $X=(x_1, x_2, \dots, x_7)$ и с тремя выходными переменными $Y=(y_1, y_2, y_3)$ (таблица 3.1).

Таблица 3.1- Статистические данные объекта управления (обучающая выборка)

| №п/п | x_1 | x_2 | x_3 | x_4 | x_5 | x_6 | x_7 | y_1 | y_2 | y_3 |
|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 1 | 1 | 655 | 0,1 | 1,22 | 0,3 | 0,35 | 5 | 245 | 89 | 40 |
| 2 | 1 | 679 | 0,2 | 1,2 | 0,35 | 0,35 | 7 | 253 | 91 | 35 |
| 3 | 1 | 644 | 0,21 | 1,22 | 0,3 | 0,35 | 7 | 246 | 83 | 38 |
| 4 | 1 | 644 | 0,17 | 1,11 | 0,2 | 0,4 | 7 | 230 | 81 | 39 |
| 5 | 1 | 635 | 0,2 | 1,21 | 0,2 | 0,4 | 6 | 245 | 78 | 37 |
| 6 | 1 | 655 | 0,2 | 1,21 | 0,2 | 0,4 | 7 | 215 | 75 | 42 |
| 7 | 1 | 655 | 0,2 | 1,21 | 0,2 | 0,4 | 7 | 210 | 70 | 45 |
| 8 | 1 | 655 | 0,2 | 1,21 | 0,2 | 0,4 | 7 | 208 | 68 | 51 |
| 9 | 1 | 650 | 0,22 | 1,24 | 0,3 | 0,35 | 15 | 189 | 61 | 55 |
| 10 | 1 | 672 | 0,22 | 1,2 | 0,3 | 0,35 | 16 | 150 | 59 | 57 |
| 11 | 1 | 679 | 0,22 | 1,2 | 0,3 | 0,35 | 15 | 198 | 62 | 52 |
| 12 | 1 | 635 | 0,22 | 1,2 | 0,3 | 0,35 | 15 | 157 | 60 | 53 |
| 13 | 1 | 644 | 0,21 | 1,22 | 0,3 | 0,35 | 15 | 190 | 71 | 48 |
| 14 | 1 | 644 | 0,17 | 1,11 | 0,2 | 0,4 | 15 | 150 | 59 | 59 |
| 15 | 1 | 575 | 0,17 | 1,11 | 0,2 | 0,4 | 15 | 139 | 55 | 60 |
| 16 | 1 | 635 | 0,18 | 1,11 | 0,2 | 0,4 | 15 | 184 | 67 | 54 |
| 17 | 1 | 635 | 0,2 | 1,21 | 0,2 | 0,4 | 14 | 193 | 68 | 49 |

| | | | | | | | | | | |
|----|---|-----|-----|------|-----|------|----|-----|----|----|
| 18 | 1 | 655 | 0,2 | 1,21 | 0,2 | 0,4 | 14 | 200 | 71 | 37 |
| 19 | 2 | 650 | 0,2 | 1,21 | 0,3 | 0,35 | 17 | 145 | 59 | 46 |
| 20 | 2 | 650 | 0,2 | 1,21 | 0,3 | 0,35 | 17 | 137 | 55 | 53 |
| 21 | 2 | 650 | 0,2 | 1,21 | 0,3 | 0,35 | 17 | 124 | 51 | 58 |
| 22 | 2 | 650 | 0,2 | 1,21 | 0,3 | 0,35 | 18 | 132 | 55 | 64 |
| 23 | 2 | 650 | 0,2 | 1,21 | 0,4 | 0,4 | 18 | 126 | 54 | 62 |
| 24 | 2 | 650 | 0,2 | 1,21 | 0,4 | 0,4 | 18 | 117 | 51 | 67 |
| 25 | 2 | 650 | 0,2 | 1,21 | 0,4 | 0,4 | 18 | 113 | 49 | 65 |
| 26 | 2 | 650 | 0,2 | 1,21 | 0,4 | 0,4 | 18 | 119 | 52 | 66 |
| 27 | 2 | 650 | 0,2 | 1,21 | 0,4 | 0,4 | 18 | 127 | 61 | 67 |
| 28 | 2 | 650 | 0,2 | 1,21 | 0,4 | 0,4 | 18 | 118 | 53 | 64 |
| 29 | 2 | 650 | 0,2 | 1,21 | 0,4 | 0,4 | 18 | 125 | 51 | 68 |
| 30 | 2 | 650 | 0,2 | 1,21 | 0,4 | 0,4 | 18 | 131 | 53 | 69 |
| 31 | 2 | 650 | 0,2 | 1,21 | 0,4 | 0,4 | 18 | 120 | 54 | 65 |
| 32 | 2 | 650 | 0,2 | 1,21 | 0,4 | 0,4 | 18 | 121 | 49 | 69 |

Необходимо построить модель на нечетких множествах, описывающую зависимость выходных переменных y_1, y_2, y_3 от входных переменных x_1, x_2, \dots, x_7 . Проверить адекватность построенной модели и точность ее прогнозирования на проверочной выборке (таблица 3.2).

Таблица 3.2- Статистические данные объекта управления (проверочная выборка)

| №п/п | x_1 | x_2 | x_3 | x_4 | x_5 | x_6 | x_7 | y_1 | y_2 | y_3 |
|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 1 | 1 | 655 | 0,1 | 1,22 | 0,3 | 0,35 | 5 | 245 | 89 | 40 |
| 2 | 1 | 679 | 0,2 | 1,2 | 0,35 | 0,35 | 7 | 253 | 91 | 35 |
| 3 | 1 | 644 | 0,21 | 1,22 | 0,3 | 0,35 | 7 | 246 | 83 | 38 |
| 4 | 1 | 644 | 0,17 | 1,11 | 0,2 | 0,4 | 7 | 230 | 81 | 39 |
| 5 | 1 | 635 | 0,2 | 1,21 | 0,2 | 0,4 | 6 | 245 | 78 | 37 |
| 6 | 1 | 655 | 0,2 | 1,21 | 0,2 | 0,4 | 7 | 215 | 75 | 42 |
| 7 | 1 | 655 | 0,2 | 1,21 | 0,2 | 0,4 | 7 | 210 | 70 | 45 |
| 8 | 1 | 655 | 0,2 | 1,21 | 0,2 | 0,4 | 7 | 208 | 68 | 51 |
| 9 | 1 | 650 | 0,22 | 1,24 | 0,3 | 0,35 | 15 | 189 | 61 | 55 |
| 10 | 1 | 672 | 0,22 | 1,2 | 0,3 | 0,35 | 16 | 150 | 59 | 57 |

1) Вызвать программу *matlab*, создать новый файл с названием *primer* разместить в нем два файла: с входными переменными под названием *datin* и файл с выходными переменными y_1, y_2, y_3 - под названием *datout*. Исходные данные копировать с таблицы 3.1. Для создания нового файла с данными, в меню *matlab* вызывать *File\New\Variable*. При этом появится окно с пустым файлом *unnamed*. Вызвать контекстное меню командой *Rename* и переименовать файл *datin*. Нажав дважды мышкой раскрыть пустой файл. В появившуюся таблицу записать входные данные, скопированные с таблицы 3.1.

Просмотреть графики входных переменных с целью визуальной оценки наличия «грубых» данных: выбрать столбец с анализируемыми данными, вызвать контекстное меню, выполнить команду *plot(datin(1:32,n))*, где $n=1,2,\dots,7$ номер столбца с входными данными. При этом на экране появится график выбранной входной переменной. При обнаружении грубых данных их скорректировать путем замены на среднее арифметическое, вычисленное по «соседним» данным. Затем закрыть таблицу с данными.

Нажать иконку с названием *New Variable* и создать второй файл с названием *datout*, заполнив его выходными данными y_1, y_2, y_3 , скопированными с таблицы 3.1. Просмотреть график выходной переменной с целью оценки наличия грубых данных и их корректиров-

ки при необходимости. Сохранить оба созданных файла в папке с отчетом о выполняемой лабораторной работе под общим названием *primer* с помощью команды *matlab: File\Save\Workspase As....*

2) Создать модель на нечетких множествах. Для создания модели использовать функцию *genfis2*. Функция *genfis2* генерирует систему нечеткого логического вывода типа «Сугэно» из данных с использованием субтрактивной кластеризации. Экстракция правил из данных в функции *genfis2* происходит в два этапа. Вначале используется функция *subclust* для определения количества правил и мощностей терм-множеств выходных переменных. Затем с помощью метода наименьших квадратов определяется "то-" часть каждого правила. В результате этого получается система нечеткого логического вывода с базой правил, покрывающих всю предметную область.

Функция *genfis2* может иметь до пяти входных аргументов, первые три из которых обязательны:

- *Xin* – матрица, в которой каждая строка содержит значения входных переменных;
- *Xout* – матрица, в которой каждая строка содержит значения выходных переменных;

- *radii* – вектор, определяющий размеры области правил по каждой координате. Значения координат вектора *radii* должны находиться в диапазоне [0, 1] в связи с тем, что во время выполнения функции *subclust* данные масштабируются на единичный гиперкуб. Если значение *radii* задано скаляром, тогда все координаты считаются равноважными;

- *xBounds* – матрица диапазонов изменения данных, необходимая для их масштабирования на единичный гиперкуб. Каждый столбец матрицы задает диапазон изменения данных по одной координате, т.о. размер матрицы – $2 \times p$, где p – количество входных и выходных переменных. Если аргумент *xBounds* не задан, тогда диапазоны изменения данных рассчитываются функцией *subclust* по фактическим значениям матриц *Xin* и *Xout* [1].

M-файлы являются обычными текстовыми файлами, которые создаются с помощью текстового редактора.

Предварительно определить диапазон изменения переменных, используемых для построения модели. Для чего в командном окне *Command Window* набрать команды *File\New\M-File*. В текстовом редакторе *Editor – Untitled* набрать команды:

```
i=min(datin)
j=max(datin)
a=[i;j]
```

При этом в файл *matlab primer* занесутся файлы с названиями *i*, *j*, *a*. Открыть файл *a*, скопировать данные и записать их в таблицу с диапазоном изменения входных переменных (таблица 3.3).

Таблица 3.3- Диапазон изменения входных переменных *datin*

| Переменные | <i>x1</i> | <i>x2</i> | <i>x3</i> | <i>x4</i> | <i>x5</i> | <i>x6</i> | <i>x7</i> |
|-----------------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| Минимальное значение | 1 | 575 | 0,1 | 1,11 | 0,2 | 0,35 | 5 |
| Максимальное значение | 2 | 679 | 0,22 | 1,24 | 0,4 | 0,4 | 18 |

Аналогично определить диапазон изменения выходных переменных (таблица 3.4).

Таблица 3.4- Диапазон изменения выходных переменных *datout*

| Переменные | <i>y1</i> | <i>y2</i> | <i>y3</i> |
|-----------------------|-----------|-----------|-----------|
| Минимальное значение | 113 | 49 | 35 |
| Максимальное значение | 253 | 91 | 69 |

Построить модель на нечетких множествах в *M-File model*:
`expert =genfis2(datin,datout,0.44,[1 575 0.1 1.11 0.2 0.35 5 113
49 35; 2 679 0.22 1.24 0.4 0.4 18 253 91 69])`

3) Просмотреть структуру модели набрав в командном окне:

`Fuzzy (expert)`

При этом на экране *Fis Editor:sug 73* отобразится структура модели, аналогичная рисунку 3.1, с семью входными и тремя выходными переменными.

Просмотреть сформированные нечеткие переменные и функции принадлежности, набрав команду:

`Edit\Membership Functions`

При этом на экране отобразятся функции принадлежности, аналогичные рисунку 3.2, для каждой входной переменной, выбираемой клавишей *FIS Variables*.

Ознакомиться с сформированным набором правил вывода, набрав команду:

`Edit\Rules`

При этом на экране отобразятся правила построения нечеткой модели, аналогичные рисунку 3.3.

4) Рассчитать выходные значения переменных y_1, y_2, y_3 , записав их в файл *modout*, на полученной модели *expert*:

`modout = evalfis(datin, expert)`

Функция *evalfis* выполняет нечеткий логический вывод, может иметь три входных аргумента, два из которых обязательные:

- *datin* – матрица значений входных переменных, для которых необходимо выполнить нечеткий логический вывод. Матрица должна иметь размер $m \times n$, где n – количество входных переменных; m – количество входных данных. Каждая строка матрицы представляет один вектор значений входных переменных;

- *expert* – идентификатор системы нечеткого логического вывода.

Оценить качество разработанной модели, для чего построить совмещенные графики выходных переменных по фактическим *datout* и рассчитанным по модели *expert* данным *modout*:

`subplot(2,1,1);
plot(datout);
legend('show');`

`subplot(2,1,2);
plot(modout);
legend('show');`

Файлы с выходными данными *datout* и *modout* скопировать в *EXCEL* и рассчитать коэффициенты детерминации по выходным переменным по формуле:

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - y_{i,mod})^2}{\sum_{i=1}^n (y_i - y_{cp})^2}, \quad (3.1)$$

где y_i – фактическое значение выходной переменной в обучающей выборке, $i=1, 2, 3$; $y_{i,mod}$ – значение переменной на выходе модели, $i=1, 2, 3$; y_{cp} – среднее арифметическое значение выходной переменной в обучающей выборке.

Проверка значимости коэффициента детерминации осуществляется путем сравнения расчетного значения критерия Фишера:

$$F_{расч} = \frac{R^2 / n - m}{1 - R^2 / m - 1}, \quad (3.2)$$

с табличным $F_{табл}$. Табличное значение критерия определяется заданным уровнем значимости $\alpha = 0,05$ и степенями свободы $k_1 = m - 1$ и $k_2 = n - m$, где n – количество данных в обучающей выборке, $m = 7$ – количество входных переменных. Коэффициент R^2 значительно отличается от нуля, если выполняется неравенство

$$F_{расч} > F_{табл}.$$

Коэффициент детерминации показывает долю вариации выходной переменной, находящегося под воздействием входных переменных, т. е. определяет, какая доля вариации Y учтена в модели и обусловлена влиянием на него входных переменных. Чем ближе R^2 к 1, тем выше качество модели. Данный коэффициент является универсальным, так как он отражает тесноту связи и точность модели, а также может использоваться при любой форме связи переменных.

Коэффициент детерминации модели, рассчитанный по выходной переменной y_1 , равен $R^2 = 0,95$, статистически значим с уровнем значимости 0,05. Большое значение коэффициента множественной корреляции свидетельствует о том, что вариация переменной y_1 в обучающей выборке на 95% определяется вариацией входных переменных $x_1 - x_7$. Влияние неучтенных факторов незначительное, составляет 5%.

5) Определить пригодность разработанной модели на проверочной выборке, приведенной в таблице 3.2, предварительно создав файлы для входных переменных *chkdatin* и выходных переменных *chkdatout*. Рассчитать (прогнозировать) значения выходных переменных и записать их в файл *chkfuzout*:

$$chkfuzout = evalfis(chkdatin, expert);$$

Построить совмещенные графики выходных переменных по фактическим *chkdatout* и рассчитанным (прогнозированным) по модели *expert* данным *chkfuzout*:

$$\begin{aligned} & subplot(2,1,1); \\ & plot(chkdatout); \\ & legend('show'); \end{aligned}$$

$$\begin{aligned} & subplot(2,1,2); \\ & plot(chkfuzout); \\ & legend('show'); \end{aligned}$$

Файлы с выходными проверочными *chkdatout* и прогнозными *chkfuzout* данными скопировать в EXCEL. Рассчитать дисперсию погрешности модели s^2 и коэффициент парной корреляции r по выходным переменным y_1, y_2, y_3 по формулам:

$$s^2 = \frac{\sum_{i=1}^n (y_i - y_{i,mod})^2}{n}, \quad (3.3)$$

$$r = \frac{\sum_{i=1}^n (y_i - y_{cp}) (y_{i,mod} - y_{mod,cp})}{s \cdot s_{mod}}, \quad (3.4)$$

где y_i – значения выходной переменной в проверочной выборке, $i = 1, 2, 3$; $y_{i,mod}$ – прогнозное значение переменной на выходе модели, $i = 1, 2, 3$; y_{cp} – среднее арифметическое значение выходной переменной в проверочной выборке; $y_{mod,cp}$ – среднее арифметическое значение прогнозной переменной на выходе модели; s – стандартное отклонение переменных в проверочной выборке; s_{mod} – стандартное отклонение прогнозных переменных; n – количество данных в проверочной выборке.

Значимость коэффициента парной корреляции проверяется по t - критерию Стьюдента:

$$t_{\text{набл}} = \sqrt{\frac{r_{y,x}^2}{1-r_{y,x}^2}} (n-2) \quad (3.5)$$

Вычисленное по этой формуле значение $t_{\text{набл}}$ сравнивается с критическим значением t -критерия, которое берется из таблицы значений t Стьюдента с учетом заданного уровня значимости 0,05 и числа степеней свободы $(n-2)$.

Если $t_{\text{набл}} > t_{\text{кр}}$, то полученное значение коэффициента корреляции признается значимым (то есть нулевая гипотеза, утверждающая равенство нулю коэффициента корреляции, отвергается). И таким образом делается вывод о том, что между исследуемыми переменными есть тесная статистическая взаимосвязь.

Дисперсия погрешности модели, рассчитанная по выходной переменной y_1 , равна $s^2 = 150,5$. Коэффициент парной корреляции между y_1 и $y_{1\text{ мод}}$ равен $r = 0,92$, статистически значим с уровнем значимости 0,05. Большое значение коэффициента парной корреляции свидетельствует о том, что модель с высокой точностью описывает характер изменения выходной переменной y_1 на проверочной выборке. Это подтверждает высокую точность разработанной модели на нечетких множествах.

3.3 Задание к лабораторной работе

Разработать математическую модель на основе использования математического аппарата нечетких систем в программе *MATLAB 7.0.1*. Оценить адекватность разработанной модели на обучающей выборке. Оценить точность модели на проверочной выборке. Составить отчет по лабораторной работе и сделать выводы по результатам исследований.

3.4 Порядок выполнения лабораторной работы

Получить от преподавателя индивидуальное задание. Ознакомиться с методом построения нечеткой модели в программе *MATLAB 7.0.1*.

1) Создать в *MATLAB* файл с исходными данными. Проверить данные на наличие «грубых» ошибок.

2) Определить диапазон изменения переменных. Построить многомерную нечеткую модель, описывающую зависимость выходных переменных от множества входных переменных. Записать модель в m -файл.

3) Проанализировать структуру разработанной нечеткой модели, функции принадлежности и сформированный набор правил вывода.

4) Рассчитать выходные значения переменных y_1, y_2, y_3 на созданной нечеткой модели. Оценить качество разработанной модели по заданной выходной переменной.

5) Определить пригодность разработанной модели на проверочной выборке по заданной выходной переменной по величине дисперсии погрешности модели и коэффициенту парной корреляции между проверочными и модельными данными.

6) Составить отчет по лабораторной работе.

3.5 Содержание отчета

- 1) Индивидуальное задание на лабораторную работу.
- 2) Таблица с анализом исходных данных, используемых для построения модели на нечетких множествах.
- 3) Текстовый m - файл с m -функцией для моделирования многомерного объекта управления.
- 4) Структура разработанной нечеткой модели, функции принадлежности и сформированный набор правил вывода.
- 5) Расчеты значения коэффициента детерминации модели по заданной выходной переменной и оценка его значимости при уровне значимости 0,05.
- 6) Расчеты точности модели на проверочной выборке, оценка тесноты стохастической связи прогнозных данных с проверочными данными.
- 7) Выводы по результатам исследований разработанной модели объекта управления на нечетких множествах.

3.6 Вопросы для самоконтроля

- 1) В каких случаях используются нечеткие системы для разработки моделей объектов управления.
- 2) Что используется в качестве входных и выходных сигналов при разработке модели объекта управления.
- 3) Структура нечеткой системы с фуззификатором и дефуззификатором.
- 4) Фуззификатор, его функция в структуре нечеткой системы.
- 5) Назначение функции принадлежности в нечеткой системе и их виды.
- 6) Назначение дефуззификатора в структуре нечеткой системы.
- 7) Функция *genfis2* пакета *MATLAB Fuzzy Logic Toolbox*, используемая для построения модели. Входные аргументы функции.
- 8) Назначение функция *evalfis*. Обязательные аргументы функции.
- 9) Назначение обучающей и проверочной выборок при построении модели в нечеткой системе.
- 10) Оценка адекватности и точности модели объекта управления, созданной в нечеткой системе.

3.7 Список рекомендуемой литературы

- 1) Осовский С. Нейронные сети для обработки информации /Пер. с польского И.Д. Рудинского. -М.: Финансы и статистика, 2002. -344с. –ISBN 5-279-02567-4.
- 2) Перевод Конюшенко В.В. Начало работы с MATLAB. *Online* Документация по программному продукту. [Электронный ресурс]. – URL: <http://matlab.exponenta.ru/ml/book3/index.php>

3) Чушикина А.Ю. Исследование и разработка системы поддержки принятия решений на основе нечеткого обратного вывода /магистерская диссертация. Владим. гос. ун-т, Владимир, 2012.

4) Фоминой Н. А. Исследование эффективности применения моделей на нечетких множествах при управлении технологическими системами / магистерская диссертация. Владим. гос. ун-т, Владимир, 2011.

3.8 Варианты заданий к лабораторной работе

Задание 1. Построить нечеткую модель объекта управления с семью входными x_1 - x_7 и тремя выходными y_1 - y_3 переменными на основе использования математического аппарата нечетких систем в программе *MATLAB* 7.0.1. Оценить точность разработанной модели по выходной переменной y_1 . Исходные данные для построения модели и проверки точности приведены соответственно в таблицах 3.3 и 3.4.

Таблица 3.3- Статистические данные объекта управления (обучающая выборка)

| № п/п | x_1 | x_2 | x_3 | x_4 | x_5 | x_6 | x_7 | y_1 | y_2 | y_3 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 1 | 1 | 655 | 0,1 | 1,22 | 0,3 | 0,35 | 5 | 245 | 89 | 40 |
| 2 | 1 | 679 | 0,2 | 1,2 | 0,35 | 0,35 | 7 | 253 | 91 | 35 |
| 3 | 1 | 644 | 0,21 | 1,22 | 0,3 | 0,35 | 7 | 246 | 83 | 38 |
| 4 | 1 | 635 | 0,2 | 1,21 | 0,2 | 0,4 | 6 | 245 | 78 | 37 |
| 5 | 1 | 650 | 0,22 | 1,24 | 0,3 | 0,35 | 15 | 189 | 61 | 55 |
| 6 | 1 | 679 | 0,22 | 1,2 | 0,3 | 0,35 | 15 | 198 | 62 | 52 |
| 7 | 1 | 635 | 0,22 | 1,2 | 0,3 | 0,35 | 15 | 157 | 60 | 53 |
| 8 | 1 | 644 | 0,21 | 1,22 | 0,3 | 0,35 | 15 | 190 | 71 | 48 |
| 9 | 1 | 575 | 0,17 | 1,11 | 0,2 | 0,4 | 15 | 139 | 55 | 60 |
| 10 | 1 | 635 | 0,18 | 1,11 | 0,2 | 0,4 | 15 | 184 | 67 | 54 |
| 11 | 1 | 635 | 0,2 | 1,21 | 0,2 | 0,4 | 14 | 193 | 68 | 49 |
| 12 | 1 | 655 | 0,2 | 1,21 | 0,2 | 0,4 | 14 | 200 | 71 | 37 |
| 13 | 2 | 650 | 0,2 | 1,21 | 0,3 | 0,35 | 17 | 137 | 55 | 53 |
| 14 | 2 | 650 | 0,2 | 1,21 | 0,3 | 0,35 | 17 | 124 | 51 | 58 |
| 15 | 2 | 650 | 0,2 | 1,21 | 0,4 | 0,4 | 18 | 126 | 54 | 62 |
| 16 | 2 | 650 | 0,2 | 1,21 | 0,4 | 0,4 | 18 | 117 | 51 | 67 |
| 17 | 2 | 650 | 0,2 | 1,21 | 0,4 | 0,4 | 18 | 113 | 49 | 65 |
| 18 | 2 | 650 | 0,2 | 1,21 | 0,4 | 0,4 | 18 | 118 | 53 | 64 |
| 19 | 2 | 650 | 0,2 | 1,21 | 0,4 | 0,4 | 18 | 125 | 51 | 68 |
| 20 | 2 | 650 | 0,2 | 1,21 | 0,4 | 0,4 | 18 | 131 | 53 | 69 |
| 21 | 2 | 650 | 0,2 | 1,21 | 0,4 | 0,4 | 18 | 120 | 54 | 65 |
| 22 | 2 | 650 | 0,2 | 1,21 | 0,4 | 0,4 | 18 | 121 | 49 | 69 |

Таблица 3.4- Статистические данные объекта управления (проверочная выборка)

| № п/п | x_1 | x_2 | x_3 | x_4 | x_5 | x_6 | x_7 | y_1 | y_2 | y_3 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 1 | 1 | 644 | 0,17 | 1,11 | 0,2 | 0,4 | 15 | 150 | 59 | 59 |
| 2 | 2 | 650 | 0,2 | 1,21 | 0,3 | 0,35 | 17 | 145 | 59 | 46 |
| 3 | 2 | 650 | 0,2 | 1,21 | 0,3 | 0,35 | 18 | 132 | 55 | 64 |
| 4 | 1 | 644 | 0,17 | 1,11 | 0,2 | 0,4 | 7 | 230 | 81 | 39 |
| 5 | 2 | 650 | 0,2 | 1,21 | 0,4 | 0,4 | 18 | 119 | 52 | 66 |
| 6 | 1 | 655 | 0,2 | 1,21 | 0,2 | 0,4 | 7 | 215 | 75 | 42 |
| 7 | 1 | 655 | 0,2 | 1,21 | 0,2 | 0,4 | 7 | 210 | 70 | 45 |
| 8 | 1 | 655 | 0,2 | 1,21 | 0,2 | 0,4 | 7 | 208 | 68 | 51 |
| 9 | 2 | 650 | 0,2 | 1,21 | 0,4 | 0,4 | 18 | 127 | 61 | 67 |
| 10 | 1 | 672 | 0,22 | 1,2 | 0,3 | 0,35 | 16 | 150 | 59 | 57 |

Задание 2. Построить нечеткую модель объекта управления с семью входными x_1 - x_7 и тремя выходными y_1 - y_3 переменными на основе использования математического аппарата нечетких систем в программе *MATLAB* 7.0.1. Оценить точность разработанной модели по выходной переменной y_2 . Исходные данные для построения модели и проверки точности приведены соответственно в таблицах 3.3 и 3.4.

Задание 3. Построить нечеткую модель объекта управления с семью входными x_1 - x_7 и тремя выходными y_1 - y_3 переменными на основе использования математического аппарата нечетких систем в программе *MATLAB* 7.0.1. Оценить точность разработанной модели по выходной переменной y_3 . Исходные данные для построения модели и проверки точности приведены соответственно в таблицах 3.3 и 3.4.

Задание 4. Построить нечеткую модель объекта управления с семью входными x_1 - x_7 и двумя выходными y_1 - y_2 переменными на основе использования математического аппарата нечетких систем в программе *MATLAB* 7.0.1. Оценить точность разработанной модели по выходной переменной y_1 . Исходные данные для построения модели и проверки точности приведены соответственно в таблицах 3.3 и 3.4.

Задание 5. Построить нечеткую модель объекта управления с семью входными x_1 - x_7 и двумя выходными y_1 - y_2 переменными на основе использования математического аппарата нечетких систем в программе *MATLAB* 7.0.1. Оценить точность разработанной модели по выходной переменной y_2 . Исходные данные для построения модели и проверки точности приведены соответственно в таблицах 3.3 и 3.4.

Задание 6. Построить нечеткую модель объекта управления с семью входными x_1 - x_7 и двумя выходными y_1 - y_3 переменными на основе использования математического аппарата нечетких систем в программе *MATLAB* 7.0.1. Оценить точность разработанной модели по выходной переменной y_1 .

Исходные данные для построения модели и проверки точности приведены соответственно в таблицах 3.3 и 3.4.

Задание 7. Построить нечеткую модель объекта управления с семью входными x_1 - x_7 и двумя выходными y_1 - y_2 переменными на основе использования математического аппарата нечетких систем в программе *MATLAB* 7.0.1. Оценить точность разработанной модели по выходной переменной y_2 . Исходные данные для построения модели и проверки точности приведены соответственно в таблицах 3.3 и 3.4.

Задание 8. Построить нечеткую модель объекта управления с семью входными x_1 - x_7 и двумя выходными y_2 - y_3 переменными на основе использования математического аппарата нечетких систем в программе *MATLAB* 7.0.1. Оценить точность разработанной модели по выходной переменной y_2 . Исходные данные для построения модели и проверки точности приведены соответственно в таблицах 3.3 и 3.4.

Задание 9. Построить нечеткую модель объекта управления с семью входными x_1 - x_7 и двумя выходными y_2 - y_3 переменными на основе использования математического аппарата нечетких систем в программе *MATLAB* 7.0.1. Оценить точность разработанной модели по выходной переменной y_3 . Исходные данные для построения модели и проверки точности приведены соответственно в таблицах 3.3 и 3.4.

Задание 10. Построить нечеткую модель объекта управления с семью входными x_1 - x_7 и тремя выходными y_1 - y_3 переменными на основе использования математического аппарата нечетких систем в программе *MATLAB* 7.0.1. Оценить точность разработанной модели по выходной переменной y_2 . Исходные данные для построения модели и проверки точности приведены соответственно в таблицах 3.1 и 3.2.

Задание 11. Построить нечеткую модель объекта управления с семью входными x_1 - x_7 и тремя выходными y_1 - y_3 переменными на основе использования математического аппарата нечетких систем в программе *MATLAB* 7.0.1. Оценить точность разработанной модели по выходной переменной y_3 . Исходные данные для построения модели и проверки точности приведены соответственно в таблицах 3.1 и 3.2.

Лабораторная работа №4

Построение моделей на нейронных сетях

4.1 Цель работы

Ознакомление магистрантов с методами разработки математических моделей на основе использования нейронных сетей в программе *STATISTICA Neural Networks*.

4.2 Теоретические сведения

В большинстве случаев нейронные сети используются для предсказания (прогнозирования) значений одной или нескольких зависимых переменных от множества независимых переменных. Задачей временного ряда является предсказание зависимой (более поздней) переменной от ранних значений самой переменной.

В большинстве случаев нейронная сеть используется для предсказания одной переменной. При необходимости предсказания нескольких переменных используют несколько нейронных сетей с одним выходом. Опыт показывает, что сети с одной выходной переменной более предпочтительнее, чем сеть с многими выходными переменными, потому что взаимосвязи между выходными переменными создают затруднения при обучении такой сети.

Для моделирования объекта управления в лабораторной работе в качестве архитектуры нейронной сети выбирается многослойный персептрон, т.к. он позволяет решать широкий круг задач: прогнозирование, классификацию, кластеризацию, распознавание, аппроксимацию. Данная архитектура достаточно проста по структуре и в то же время широко используется для решения прикладных задач [1].

В качестве алгоритма обучения в работе выбран алгоритм обратного распространения. Алгоритм последовательно обучает сеть на данных из обучающего множества, уменьшает суммарную ошибку на всем обучающем множестве [2].

Процесс обучения нейронной сети может рассматриваться как настройка архитектуры и весов связей для эффективного выполнения поставленной задачи. Существуют три парадигмы обучения: с учителем, без учителя (самообучение) и смешанная.

Метод обучения «с учителем» заключается в последовательном предъявлении примеров значений входных переменных. На каждый входной пример существует требуемый ответ. Веса настраиваются таким образом, чтобы выходы сети были как можно более близкими к требуемым ответам. Более «жесткий» вариант обучения с учителем предполагает, что известна только критическая оценка правильности выхода нейронной сети, а не сами требуемые значения выхода. После многократного предъявления примеров веса се-

ти стабилизируются, причем сеть дает правильные ответы на все (или почти все) примеры из базы данных. В таком случае говорят, что сеть обучена. В программных реализациях можно видеть, что в процессе обучения величина ошибки (сумма квадратов ошибок по всем выходам) постепенно уменьшается. Когда величина ошибки достигает нуля или приемлемо малого уровня, обучение останавливают, и сеть готова к распознаванию.

Для обучения без учителя не нужно знания требуемых ответов на каждый пример обучающей выборки. В этом случае происходит распределение образцов по категориям (кластерам) в соответствии с внутренней структурой данных или степенью корреляции между образцами.

Нейронная сеть считается устойчивой, если после конечного числа итераций обучения ни один из примеров обучающей выборки не изменяет своей принадлежности в кластерах.

При смешанном обучении весовые коэффициенты одной группы нейронов настраиваются посредством обучения с учителем, а другой группы - на основе самообучения.

Важно отметить, что вся информация, которую сеть приобретает о задаче, содержится в наборе примеров. Поэтому качество обучения сети зависит от количества примеров в обучающей выборке, а также от того, насколько полно эти примеры описывают задачу. Считается, что для полноценной тренировки требуется хотя бы несколько десятков (а лучше сотен) примеров.

Для проверки правильности обучения построенной нейронной сети в нейромимитаторах предусмотрены специальные средства ее тестирования. В сеть вводится некоторый сигнал, который, как правило, не совпадает ни с одним из входных сигналов примеров обучающей выборки. Далее анализируется получившийся выходной сигнал сети [6].

Тестирование обученной сети может проводиться либо на одиночных входных сигналах, либо на тестовой выборке, которая имеет структуру, аналогичную обучающей выборке, и также состоит из пар (<вход>, <требуемый выход>). Обычно, обучающая и тестовая выборки не пересекаются. Тестовая выборка строится индивидуально для каждой решаемой задачи.

Одним из недостатков нейронных сетей является явление «переобучения». Оно заключается в том, что ошибки обучения на элементах обучающего множества оказываются очень малыми, но когда на вход сети подаются новые данные, то погрешность существенно возрастает. Это означает, что сеть «заучила» примеры, но не научилась решать задачу. Для преодоления эффекта переобучения используется разбиение исходных данных на три выборки. Обучающее подмножество используется для настройки параметров сети; контрольное – для того, чтобы контролировать, когда в сети появится

эффект переобучения; ошибка тестового множества применяется для сравнения различных моделей [3]. Задача определения длины выборок не формализована, решается постановкой вычислительного эксперимента в каждом конкретном случае.

Модели, реализуемые программами-имитаторами нейронов, чрезвычайно разнообразны. Для построения нейронной сети, ориентированной на решение конкретной задачи, используются процедуры формирования нейронных сетей, которые обеспечивают ввод указанных характеристик моделей нейронов и структур нейронных сетей [5]. Каждая группа моделей нейронных сетей может быть использована для решения лишь некоторого ограниченного класса практических задач. Так, многослойные и полносвязные нейронные сети с сигмоидальными передаточными функциями используются для распознавания образов и адаптивного управления; нейронные сети с локальными связями - для обработки изображений и некоторых других частных задач. Для решения задач линейной алгебры используются многослойные сети с особыми передаточными функциями [6].

Лишь для небольшого числа моделей нейронных сетей существует строгое математическое обоснование возможности их применения для решения конкретных практических задач. В наибольшей степени теоретически проработаны двухслойные нейронные сети с сигмоидальными передаточными функциями [6].

Наиболее известный пакет для создания и изучения нейросетевых моделей – *STATISTICA Neural Networks*. Данный программный продукт содержит самые современные и мощные алгоритмы обучения сетей, открытый *API* – интерфейс и встроенные функции Мастер решения задач, а также Генетический алгоритм отбора входных данных.

Пакет *STATISTICA Neural Networks* прост в использовании. Нейронные сети учатся на примерах. Пользователь нейронной сети подбирает представительные данные, а затем запускает алгоритм обучения, который автоматически воспринимает структуру данных. При этом от пользователя, конечно, требуется какой-то набор эвристических знаний о том, как следует отбирать и подготавливать данные, выбирать нужную архитектуру сети и интерпретировать результаты.

Автоматический конструктор сети самостоятельно выбирает подходящий тип сети и вариант архитектуры (т.е. число скрытых элементов). В стандартном варианте большинство пользователей просто проводят серию экспериментов с сетями различных типов и архитектур, выделяя на эвристическом уровне наиболее многообещающие, после чего именно на них и сосредотачивается все внимание. Автоматический конструктор сети действует по такой же схеме, и даже более формально. Задачу определения архитектуры сети он рас-

смаатривает как задачу оптимизации и для исследования возможных вариантов архитектур использует довольно сложные методы поиска (в том числе линейный поиск и вариант метода «отжига»). При этом работа алгоритма может занимать весьма длительное время, поскольку он перебирает множество различных вариантов архитектур. Однако времени у него уйдет не больше, чем у человека, выполняющего ту же серию экспериментов, а преимущество этого метода в том, что в процессе работы он не требует присутствия пользователя. Кроме того, этот алгоритм осуществляет перебор гораздо большего числа вариантов, чем способен выполнить человек, и как следствие, его результаты часто оказываются лучше.

Одна из наиболее трудных задач при построении нейронной сети – выбор архитектуры, а после того, как архитектура выбрана, - задание ее свободных параметров в большой степени зависит от сложности задачи, которую нужно решить, и поскольку заранее неизвестно, насколько задача сложна, необходим ряд экспериментов. Самое трудное – выбор числа промежуточных слоев и числа элементов в них.

Существуют следующие правила определения промежуточных слоев в многослойном персептроне *MPL*:

- При использовании *MPL* для большинства практических задач достаточно одного промежуточного слоя. Если даже при большом числе скрытых элементов не удастся уменьшить ошибку до приемлемого уровня, можно попробовать сеть с двумя промежуточными слоями.

- Чем больше число скрытых элементов в сети, тем более сложную задачу она может моделировать, но при этом потребуются более длительное обучение и возникнет опасность переобучения. При экспериментировании нужно следить за тем, *как меняются обучающая и контрольная ошибки*. Если при добавлении новых элементов уменьшаются обе ошибки, то, по-видимому, сеть пока слишком мала. Если же контрольная ошибка стала намного больше, чем ошибка обучения (и, в частности, если она увеличивается при дальнейшем обучении), то, вероятно, что сеть слишком большая.

- Общее число весов и пороговых коэффициентов в сети должно быть меньше, а лучше – намного меньше, чем число обучающих наблюдений. В идеале нужно *иметь наблюдений в десять или двадцать раз больше, чем имеющееся количество весов*. Если обучающих наблюдений недостаточно, следует ограничиться сетями небольших размеров, поскольку для моделирования сложной функции попросту не хватает данных. *Если число наблюдений меньше, чем число входов, умноженное на число выходов, следует использовать только линейные модели*.

- Для оценки числа нейронов в скрытых слоях однородных нейронных сетей можно воспользоваться формулой (данная формула является следствием

из теоремы Колмогорова-Арнольда-Хехт-Нильсена о представимости любой многомерной функции нескольких переменных с помощью нейронной сети фиксированной размерности) [7]. Для оценки необходимого числа синаптических весов W в многослойной сети с сигмоидальными передаточными функциями можно воспользоваться формулой:

$$\frac{mN}{1 + \log_2 N} \leq W \leq m \left(\frac{N}{n} + 1 \right) (n + m + 1) + m \quad (4.1)$$

где n – размерность входного сигнала, m – размерность выходного сигнала, N – число элементов обучающей выборки.

Оценив необходимое число весов, можно рассчитать число нейронов в скрытых слоях. Например, для двуслойной сети это число составит:

$$L = \frac{W}{n + m} \quad (4.2)$$

Известны и другие формулы для оценки, например:

$$2(n + L + m) \leq N \leq 10(n + L + m), \quad (4.3)$$

$$\frac{N}{10} - n - m \leq L \leq \frac{N}{2} - n - m. \quad (4.4)$$

Точно так же можно рассчитать число нейронов в сетях с большим числом нейронов.

Пример. Имеются статистические данные функционирования объекта управления – технологического процесса производства листового стекла флоат-способом [4]. Данные приведены в файле *steklo.excel*. Используя выборку с 1 по 100 строку построить нейросетевую модель, описывающую зависимость оптических искажений, видимых в проходящем свете, измеренные по методу "Зебра", от значений режимных переменных технологической линии.

Для обучения нейронных сетей используются выборки из 50 обучающих, 25 контрольных и 25 тестовых наблюдений. Контрольное множество данных будет участвовать в кросс-проверке сетей при их обучении. При отсутствии данной проверки, нейронные сети будут слишком близко следовать обучающим данным («чрезмерная подгонка») и воспринимать не столько структуру данных, сколько содержащиеся в них помехи. Кроме того, нейронные сети, которые обучались без кросс-проверки обладают малой способностью к обобщению данных (т.е. показывать хорошие результаты на совершенно новых данных). Тестовое множество данных, которое вообще никак не будет участвовать в обучении сетей, будет использоваться только после того, как будут сформированы окончательные варианты обученных сетей.

Для решения поставленной задачи выберем типовую нейронную сеть – многослойный перцептрон *MPL* с прямой передачей сигнала и с сигмоидальными передаточными функциями. Входной слой служит для ввода значений входных переменных. Каждый из скрытых и выходных нейронов соединен со всеми элементами предыдущего слоя. Можно было бы рассмотреть сети, в которых нейроны связаны только с некоторыми из нейронов предыдущего слоя; однако, для большинства приложений предпочтительны сети с полной системой связей.

А. Рассмотрим построение нейронной сети, создаваемой автоматическим конструктором, когда число входных переменных определено структурой линейной регрессионной моделью [4]:

$$Zlb=f(Tcm2, Tza, Tzc, Tolsc)$$

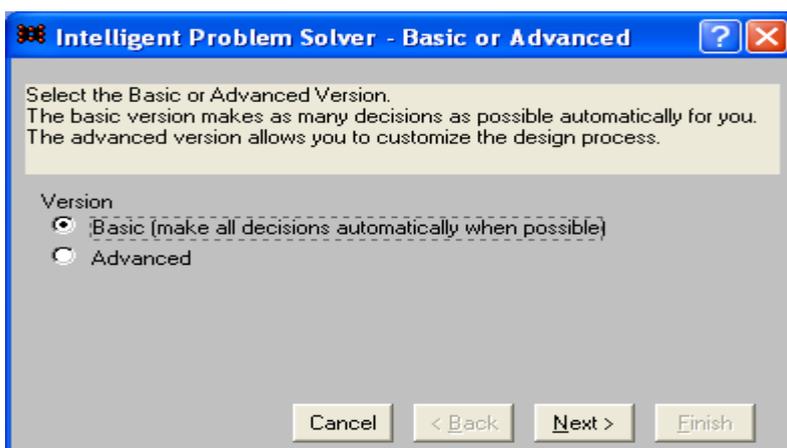
1) Для моделирования нейронной сети вводим данные в пакет *STATISTICA Neural Networks*. Исходные данные для нейронной сети, моделирующей качество вырабатываемого стекла, представлены в следующем виде:

Таблица 4.1-Статистические данные функционирования объекта управления

| VAR1 | VAR2 | VAR3 | VAR4 | VAR5 | VAR6 | VAR7 | VAR8 | VAR9 | VAR10 | VAR11 | VAR12 | VAR13 | VAR14 |
|-------------------|---------------------|--------------------|---------------------|---------------------|--------------------|------------------|------------------|---------------------|--------------------|--------------------|------------------|--------------------|--------------------|
| Tcm2 ₁ | Tcanal ₁ | Tin1R ₁ | Tin12R ₁ | Tin20R ₁ | ToutC ₁ | Tza ₁ | Tzc ₁ | Oxygen ₁ | Tolsc ₁ | Razn1 ₁ | Zlb ₁ | R1dol ₁ | R2dol ₁ |
| Tcm2 ₂ | Tcanal ₂ | Tin1R ₂ | Tin12R ₂ | Tin20R ₂ | ToutC ₂ | Tza ₂ | Tzc ₂ | Oxygen ₂ | Tolsc ₂ | Razn1 ₂ | Zlb ₂ | R1dol ₂ | R2dol ₂ |
| ... | ... | ... | ... | ... | ... | | | | | | | | |

Таблица содержит выборку из 100 строк с данными функционирования объекта управления. Первые 10 колонок заполнены входными переменными, последние с 11 по 14 колонку содержат выходные переменные. Оптических свойств вырабатываемого стекла, измеренные по методу «Зебра» находятся в колонке VAR12.

2) Создание нейронных сетей автоматическим конструктором начинается с открытия диалогового окна *Basic or Advanced* (рисунок ниже):



Используем второй вариант (*Advanced*) построения нейронных сетей, поскольку он отличается от первого расширенными настройками. Далее нажав клавишу *Next* выбираем тип проблемы *Standart*. Затем выбираем мышкой показатель качества стекла *Zlb*, который находится в колонке VAR12. Повторно нажимая клавишу *Next* выбираем входные переменные,

расположенные в колонках VAR1, VAR7, VAR8, VAR10.

На следующем шаге автоматического конструктора нейронной сети указываем количество обучающих (*Training*) 50, контрольных (*Verification*) 25 и тестовых (*Test*) 25 наблюдений.

Далее в окне *Type of Network* нужно выбрать типы сетей, которые будут анализироваться. Они упорядочены в порядке сложности обучения (рисунок ниже). Линейные (*Linear*) сети почти не требуют обучения и включены сюда потому, что дают хорошую точку отсчета для сравнения эффективности различных методов. Сети *PNN* и *GRNN* также довольно просты, радиальные базисные функции (*Radial Basis Functions*) устроены несколько сложнее, а трех- и четырехслойные многослойные перцептроны (*Multiplayer Perceptrons*) – это сложные конструкции.

Для трехслойного *MPL* поиск сводится к выбору числа элементов в скрытом слое. По существу, это линейный поиск для функции, содержащей помехи (при разных прогонах обучения сети, даже с одним и тем же числом элементов, могут получаться немного различные результаты).

3) На следующем шаге определяем сложность нейронной сети. Сложность нейронной сети зависит от количества нейронов (L) в скрытых слоях, числа синаптических связей (W) в сети и числа элементов обучающей выборки (N). Из расчетов числа нейронов в промежуточном слое имеем следующее: при выбранном $N=50$, получаем:

$$W \leq 82 \text{ по формуле (4.1),}$$

$$L \leq 16 \text{ по формуле (4.2).}$$

Таким образом, для быстроты анализа автоматическому конструктору в окне *Hidden Units* задаем число синаптических связей 40 и среднее значение числа нейронов в промежуточном слое, равное 8.

4) На следующих этапах задаем время в течение, которого будет идти поиск *Duration of design process-Timed=10* минут. В окне *Saving Networks* выбираем максимальное число лучших сетей 10, задаем режимы: *Balance performance...* и *Increase...* В следующем окне *Results Shown* помечаем все три окна, обеспечивающие сохранение листинга с результатами, суммарной статистики и с анализом чувствительности лучшей сети.

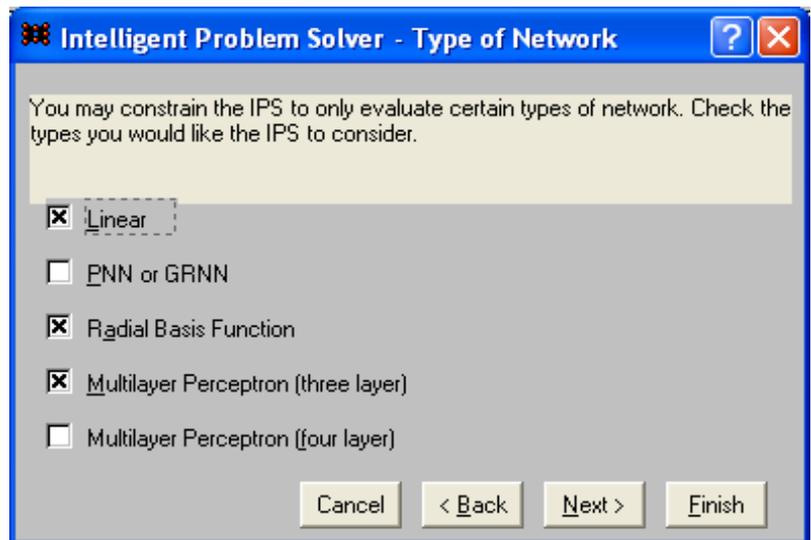
Затем нажимаем клавишу *Finish*.

5) Автоматический анализатор выдает информацию, содержащуюся в окнах.

- В окне *Intelligent Problem Solver Messages* сообщается о том, что было протестировано 9807 нейронных сетей с различными архитектурами. Определены 10 лучших. Самая лучшая из протестированных сетей обладает следующими характеристиками (коэффициенты регрессии 0.275, корреляции 0.962).

- В окне *Network Set Editor* приводятся структур 10-и лучших сетей с характеристиками: среднеквадратической ошибки, число входных переменных, количество нейронов в скрытом слое, *performance*. В рассматриваемом примере наилучшей выбрана модель типа *MLP* с двумя входными переменными (из заданных четырех переменных), два нейрона в скрытом слое, среднеквадратическая ошибка 3,06, *performance* 0,38.

- В окне *Sensitivity Analysis* приводятся результаты анализа чувствительности нейронной сети к входным данным для обучающей и контрольной выборки. Переменные ранжированы по чувствительности. В рассматриваемом примере видно, что нейронная сеть, описывающая оптические искажения, видимые в проходящем свете, измеренные по методу "Зебра, наиболее чувствительна к изменению значений режимных переменных VAR10 (*Tolsc* толщина ленты стекла) и VAR1 (*Tcm2* температура стекломассы в стекловаренной печи).



| | VAR1 | VAR10 |
|-------|----------|----------|
| Rank | 2 | 1 |
| Error | 4.201558 | 7.250218 |
| Ratio | 1.825048 | 3.149307 |
| Rank | 2 | 1 |
| Error | 4.347205 | 7.871148 |
| Ratio | 1.420043 | 2.571163 |

Обозначения:

Error – значение среднеквадратической ошибки сети в случае если рассматриваемая переменная не была бы в входных данных;

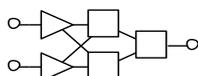
Ratio – отношение значения *Error* к значению *Baseline Error*

Baseline Error - значение среднеквадратичной ошибки сети в случае, если все переменные (их было четыре), будут входными данными

2,3 3,06

Черным цветом помечены результаты на обучающей выборке, красным - на контрольной.

- Ниже показана структура лучшей нейронной сети типа *MLP*, прогнозирующей оптические свойств вырабатываемого стекла, измеряемых по методу «Зебра».



Оценка качества данной нейронной сети представлена в диалоговом окне (Регрессионная статистика).

- В окне *Regression Statistics* (статистики регрессии, рисунок 4.1) подсчитываются среднее *Data Mean* и стандартное отклонение *Data S.D.* выходной переменной, среднее значение *Error Mean* и стандартное отклонение *Error S.D.* ошибки сети, а также отношение стандартного отклонения ошибки к стандартному отклонению данных *S.D.*

Ratio и коэффициент корреляции между спрогнозированным и фактическими значениями *Correlation* на обучающей, контрольной и тестирующих выборках. Обычно значение средней ошибки (*Error Mean*) получается близким к нулю, а стандартное отклонение прогноза оказывается существенно меньше стандартного отклонения данных.

Если величина *S.D.Ratio* меньше 0,1, это означает хорошее качество регрессии.

| | Tr.VAR12 | Ve.VAR12 | Te.VAR12 |
|-------------|-----------|-----------|-----------|
| Data Mean | 50.8 | 52.2 | 50.04 |
| Data S.D. | 7.920524 | 8.170067 | 7.103286 |
| Error Mean | 0.3565578 | -0.3496 | 1.419502 |
| Error S.D. | 2.297474 | 3.104001 | 2.729342 |
| Abs E. Mean | 1.881901 | 2.443777 | 2.307839 |
| S.D. Ratio | 0.290066 | 0.3799235 | 0.3842365 |
| Correlation | 0.9571925 | 0.9258903 | 0.9347443 |

Рисунок 4.1 – Окно *Regression Statistics*

| | RMS Error Train | 2.3 | Verify | 3.06 | Test | 3.0 |
|----|-----------------|--------|-----------|-----------|------|-----|
| | VAR12 | TVAR12 | EVAR12 | Error | | |
| 01 | 66.90376 | 66 | 0.9037552 | 0.03013 | | |
| 02 | 64.86102 | 67 | -2.138984 | 0.0713 | | |
| 03 | 64.73468 | 66 | -1.265321 | 0.04218 | | |
| 04 | 64.70245 | 63 | 1.702451 | 0.05675 | | |
| 05 | 64.70245 | 65 | -0.297549 | 0.0099183 | | |
| 06 | 64.70245 | 65 | -0.297549 | 0.0099183 | | |
| 07 | 64.73468 | 62 | 2.734679 | 0.09116 | | |
| 08 | 64.73468 | 64 | 0.7346787 | 0.02449 | | |
| 09 | 64.73468 | 64 | 0.7346787 | 0.02449 | | |
| 10 | 60.06796 | 60 | 0.06796 | 0.002265 | | |
| 11 | 60.13151 | 61 | -0.868486 | 0.02895 | | |
| 12 | 54.58704 | 52 | 2.587039 | 0.08623 | | |
| 13 | 55.00663 | 55 | 0.006633 | 0.0002211 | | |
| 14 | 54.92309 | 56 | -1.076908 | 0.0359 | | |
| 15 | 49.5371 | 51 | -1.462903 | 0.04876 | | |
| 16 | 49.44996 | 50 | -0.550036 | 0.01833 | | |
| 17 | 49.71162 | 49 | 0.7116171 | 0.02372 | | |
| 18 | 45.26036 | 48 | -2.739643 | 0.09132 | | |

- В окне *Run Data Set* (прогнать набор данных) содержатся следующие значения (перечисленные слева направо): фактические выходы сети, целевые выходные значения, ошибки, т.е. разности между первым и вторым, и суммарная ошибка по каждому наблюдению. Над таблицей выдается итоговая среднеквадратичная ошибка (СКО) – *RMS error* сети на наборе: обучающих -50, контрольных -25, тестовых -25 данных (Рисунок 4.2).

Рисунок 4.2 - Окно *Run Data Set*

Б) Рассмотрим построение нейронной сети, создаваемых автоматическим конструктором, когда входные переменные не конкретизированы и заданы множеством всех 10-и входных переменных.

Создание нейронных сетей автоматическим конструктором начинается с открытия диалогового окна *Basic or Advanced*. Помечаются мышкой выходная переменная VAR12 и все 10 входных переменных VAR1 - VAR10.

Автоматический анализатор выдает информацию, содержащуюся в окнах.

- В окне *Intelligent Problem Solver Messages* сообщается о том, что было протестировано 2839 нейронных сетей с различными архитектурами. Определены 10 лучших. Самая лучшая из протестированных сетей обладает следующими характеристиками (коэффициенты регрессии 0.2, корреляции 0.98).

- В окне *Network Set Editor* приводятся структур 10-и лучших сетей с характеристиками: среднеквадратической ошибки, число входных переменных, количество нейронов в скрытом слое, *performance*. В рассматриваемом примере наилучшей выбрана модель типа *MLP* с 9-ю входными переменными, 7 нейронов в скрытом слое, среднеквадратическая ошибка 1,75, *performance* 0,2.

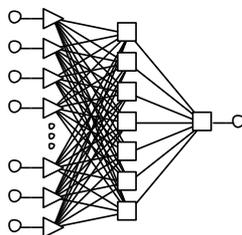
- В окне *Sensitivity Analysis* (рисунок 4.3) приводятся результаты анализа чувствительности нейронной сети к входным данным для обучающей и контрольной выборки.

| | VAR1 | VAR2 | VAR3 | VAR4 | VAR6 | VAR7 | VAR8 | VAR9 | VAR10 |
|-------|------|------|------|------|------|------|------|------|-------|
| Rank | 4 | 3 | 2 | 7 | 9 | 5 | 6 | 8 | 1 |
| Error | 2.43 | 2.96 | 3.17 | 1.96 | 1.48 | 2.43 | 2.25 | 1.50 | 6.60 |
| Ratio | 1.91 | 2.33 | 2.49 | 1.54 | 1.16 | 1.91 | 1.77 | 1.18 | 5.18 |
| Rank | 3 | 5 | 2 | 6 | 9 | 4 | 7 | 8 | 1 |
| Error | 3.06 | 2.88 | 3.49 | 2.85 | 1.69 | 3.02 | 2.68 | 2.17 | 9.90 |
| Ratio | 1.74 | 1.64 | 1.98 | 1.62 | 0.96 | 1.72 | 1.53 | 1.24 | 5.64 |

Рисунок 4.3 - Окно *Sensitivity Analysis*

Входные переменные ранжированы по чувствительности. В рассматриваемом примере видно, что нейронная сеть, описывающая оптические свойства вырабатываемого стекла по методу «Зебра», наиболее чувствительна к изменению значений режимных переменных VAR10 (*Tolsc* толщина ленты стекла), VAR3 (*Tin1R* температура в первом пролете флотат-ванны), VAR2 (*Tcanal* температура стекломассы в выработочном канале стекловаренной печи), VAR1 (*Tcm2* температура стекломассы в стекловаренной печи).

- Ниже показана структура лучшей нейронной сети типа *MLP*, прогнозирующей оптические свойств вырабатываемого стекла, измеряемых по методу «Зебра».



- В окне *Regression Statistics* (рисунок 4.4) приводятся среднее *Data Mean* и стандартное отклонение *Data S.D.* выходной переменной, среднее значение *Error Mean* и стандартное отклонение *Error S.D.* ошибки сети, а также отношение стандартного отклонения ошибки к стандартному отклонению данных *S.D. Ratio*, коэффициент корреляции между спрогнозированным и фактическими значениями *Correlation* на обучающей, контрольной и тестирующих выборках.

| | Tr.VAR12 | Ve.VAR12 | Te.VAR12 |
|-------------|-----------|-----------|-----------|
| Data Mean | 49.68 | 53.72 | 50.76 |
| Data S.D. | 5.902058 | 8.715503 | 9.452866 |
| Error Mean | -0.0022 | -0.02658 | 0.4997012 |
| Error S.D. | 1.286572 | 1.790656 | 3.536783 |
| Abs E. Mean | 0.9531708 | 1.242784 | 2.74766 |
| S.D. Ratio | 0.2179871 | 0.2054564 | 0.3741493 |
| Correlation | 0.9759938 | 0.9816402 | 0.928383 |

Рисунок 4.4 – Окно *Regression Statistics*

- В окне *Run Data Set* содержатся фактические значения выходных переменных, выходные переменные сети, ошибки сети и суммарная ошибка по каждому наблюдению. Над таблицей выдается итоговая среднеквадратичная ошибка *RMS error* сети на наборе: обучающих -50, контрольных -25, тестовых -25 данных (Рисунок 4.5).

| | RMS Error Train | 1.2 | Verify | 1.75 | Test | 3.5 |
|----|-----------------|--------|-----------|-----------|------|-----|
| | VAR12 | TVAR12 | EVAR12 | Error | | |
| 01 | 73.11785 | 66 | 7.11785 | 0.2542089 | | |
| 02 | 62.75665 | 67 | -4.243352 | 0.1515483 | | |
| 03 | 62.59275 | 66 | -3.40725 | 0.1216875 | | |
| 04 | 62.76189 | 63 | -0.238112 | 0.008504 | | |
| 05 | 64.70936 | 65 | -0.2906 | 0.01038 | | |
| 06 | 65.56636 | 65 | 0.5663574 | 0.02023 | | |
| 07 | 62.88834 | 62 | 0.8883444 | 0.03173 | | |
| 08 | 62.68308 | 64 | -1.316916 | 0.04703 | | |
| 09 | 64.40124 | 64 | 0.401244 | 0.01433 | | |
| 10 | 56.18116 | 60 | -3.81884 | 0.1363872 | | |
| 11 | 60.72658 | 61 | -0.273418 | 0.009765 | | |
| 12 | 53.01855 | 52 | 1.018553 | 0.03638 | | |
| 13 | 54.7895 | 55 | -0.2105 | 0.007518 | | |
| 14 | 57.35329 | 56 | 1.353293 | 0.04833 | | |
| 15 | 50.4636 | 51 | -0.5364 | 0.01916 | | |
| 16 | 50.59852 | 50 | 0.5985198 | 0.02138 | | |
| 17 | 49.24055 | 49 | 0.2405537 | 0.008591 | | |
| 18 | 44.49586 | 48 | -3.504142 | 0.1251479 | | |

Сравнительный анализ характеристик нейронных сетей, основанных на регрессионных моделях и характеристик нейронных сетей, созданных с автоматическим определением значимых входных переменных, показал, что последние нейронные сети обладают более лучшими характеристиками прогнозирования т.е. имеют более низкий показатель *Regression Ratio* (отношение стандартного отклонения ошибки к стандартному отклонению данных) и более высокий коэффициент корреляции между прогнозируемыми и фактическими значениями (Таблица 4.2).

Рисунок 4.5 - Окно *Run Data Set*

Таблица 4.2 – Сравнительные характеристики нейронных сетей при моделировании оптических свойств вырабатываемого стекла, измеряемых по методу «Зебра»

| Тип нейронной сети | Количество нейронов | | | Характеристики сети | | |
|--------------------|---------------------|----------------------|-----------|-----------------------|------------------------|-----------------------------|
| | на входе | в промежуточном слое | на выходе | Коэффициент регрессии | Коэффициент корреляции | Среднеквадратическая ошибка |
| <i>MLP</i> | 2 | 2 | 1 | 0,27 | 0,96 | 3,06 |
| <i>MLP</i> | 9 | 7 | 1 | 0,2 | 0,98 | 1,75 |

В качестве информационной модели процесса формования ленты стекла на расплаве олова выбираем нейронную сеть *MLP* с числом нейронов на входе - 9, в промежуточном слое -7, на выходе -1. Фактические данные изменения оптических искажений, видимых в проходящем свете, измеренные по методу "Зебра» VAR12 и результаты моделирования T.VAR12 приведены на рисунке 4.6.

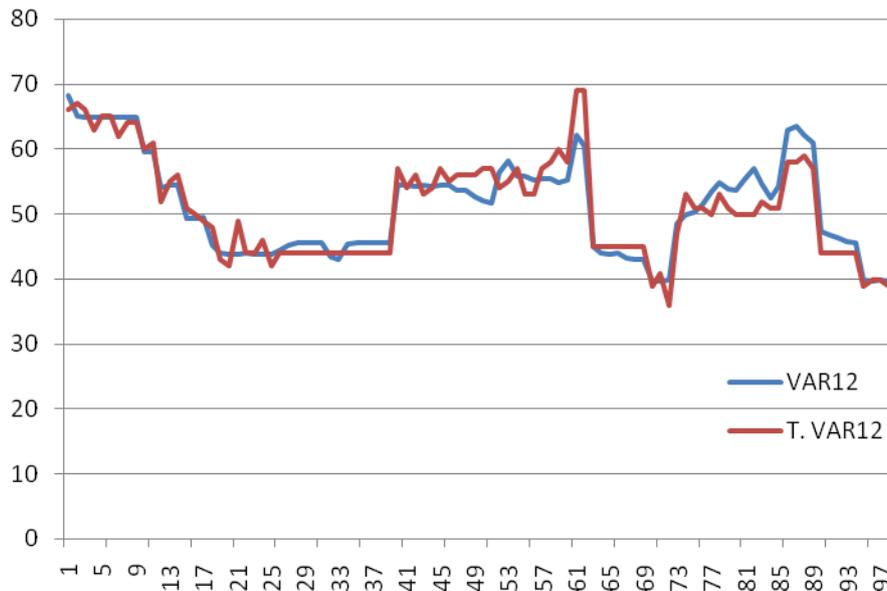


Рисунок 4.6 - Фактические данные изменения оптических искажений, видимых в проходящем свете, измеренные по методу "Зебра" VAR12 и результаты моделирования T.VAR12

4.3 Задание к лабораторной работе

Разработать математическую модель объекта управления на основе использования нейронных сетей в программе *STATISTICA Neural Networks*. Оценить адекватность разработанной модели на обучающей, а точность на контрольной выборке. Составить отчет по лабораторной работе и сделать выводы по результатам исследований.

4.4 Порядок выполнения лабораторной работы

Получить от преподавателя индивидуальное задание. Ознакомиться с методом построения нейронных сетей в программе *STATISTICA Neural Networks*.

- 1) Создать в программе *STATISTICA Neural Networks* таблицу с исходными данными путем копирования данных из файла *steklo.exe*.
- 2) Построить нейронную сеть с помощью автоматического конструктора при заданном числе входных переменных, определяемом структурой линейной регрессионной модели [4].
- 3) Проанализировать структуру разработанной нейронной сети.
- 4) Построить второй вариант нейронной сети с помощью автоматического конструктора с использованием 10-и входных переменных.
- 5) Оценить качество двух разработанных нейронных сетей и их пригодность для математического описания объекта управления по величине среднеквадратичной ошибки модели и коэффициенту парной корреляции

между фактическими и модельными данными. Выбрать лучшую нейронную сеть.

б) Построить совмещенные графики изменения выходной переменной с использованием фактических и модельных данных.

7) Сделать выводы по выполненной лабораторной работе.

4.5 Содержание отчета

1) Индивидуальное задание на лабораторную работу.

2) Нейронная сеть, построенная с помощью автоматического конструктора при заданном числе входных переменных, определяемом структурой линейной регрессионной модели. Структура сети и статистические характеристики.

3) Второй вариант нейронной сети, построенный с помощью автоматического конструктора с использованием 10-и входных переменных. Структура сети и статистические характеристики.

4) Результаты сравнительной оценки качества разработанных нейронных сетей по величине среднеквадратичной ошибки модели и коэффициенту парной корреляции между фактическими и модельными данными.

5) Обоснование выбора лучшей нейронной сети для математического описания объекта управления.

б) Совмещенные графики изменения выходной переменной с использованием фактических и модельных данных.

7) Выводы по результатам лабораторной работы.

4.6 Вопросы для самоконтроля

1) Назначение нейронных сетей, область их применения.

2) Процесс обучения нейронной сети. Методы обучения сетей, их особенности.

3) От чего зависит качество обучения нейронной сети?

4) «Переобучение» нейронной сети. Преодоление эффекта переобучения.

5) Назначение тестирования нейронной сети. Технология тестирования обученной сети.

б) Модели, реализуемые программами-имитаторами нейронов. Области использования многослойных и полносвязных нейронных сетей с сигмоидальными передаточными функциями, нейронных сетей с локальными связями, многослойных сетей с особыми передаточными функциями.

7) Особенности пакета *STATISTICA Neural Networks*. Назначение пакета и его содержание.

8) Правила определения промежуточных слоев в многослойном персептроне *MPL*: оценка числа нейронов в скрытых слоях однородных нейронных сетей.

9) Метод построения нейронной сети с помощью автоматического конструктора при заданном числе входных переменных.

10) Метод построения нейронной сети с помощью автоматического конструктора, когда входные переменные не конкретизированы и заданы множеством переменных.

11) Оценка адекватности и точности моделей, построенных на нейронных сетях.

4.7 Список рекомендуемой литературы

1) Галушкин А.И. Теория нейронных сетей. Кн.1: Учеб. пособие для вузов. – М.: ИПРЖР, 2001. -385 с.

2) Короткий С. Нейронные сети: алгоритм обратного распространения.//<http://www.neuropower/ole/rus/book/index.html>

3) Медведев, В.С. Нейронные сети. *MATLAB 6*/ В.С. Медведев, В.Г. Потемкин; под общ. ред. В.Г. Потемкина. – М.: ДИАЛОГ-МИФИ, 2002. – 496 с. – ISBN 5–86404–163–7.

4) Р.И. Макаров, В.В. Тарбеев, Е.Р. Хорошева и др. Управление качеством листового стекла (флоат-способ): Учебное пособие. -М.: Издательство Ассоциации строительных вузов, 2004.-152с. –ISBN-5-93093-261-1.

5) Барский А.Б. Нейронные сети: распознавание, управление, принятие решений. – М.: Финансы и статистика, 2004. – 176с. -ISBN 5-279-02757-X.

6) Борисов Ю, Кашкаров В., Сорокин С. Нейросетевые методы обработки информации и средства их программно-аппаратной поддержки. – Открытые системы. – 1997.-№4.

7) Круглов В.В., Борисов В.В. Искусственные нейронные сети. Теория и практика – 2-е изд. – М.: Горячая линия – Телеком, 2002. – 382 с.

4.8 Варианты заданий к лабораторной работе

Задание 1. Построить математическую модель объекта управления с десятью входными и четырьмя выходными переменными на основе использования пакета для создания и изучения нейросетевых моделей – *STATISTICA Neural Networks*. Оценить точность разработанной модели по выходной переменной *Zlb*, характеризующей оптические искажения, видимые в проходящем свете, измеренные по методу "Зебра". Исходные данные для построения модели приведены в файле *steklo.exe*. Для построения модели использовать выборку из таблицы функционирования объекта управления, строки с 101 по

200. Регрессионная модель, описывающая «Зебру» имеет следующую структуру: $Zlb=f(Tcm2, Tza, Tzc, Tolsc)$.

Задание 2. Построить математическую модель объекта управления с десятью входными и четырьмя выходными переменными на основе использования пакета для создания и изучения нейросетевых моделей – *STATISTICA Neural Networks*. Оценить точность разработанной модели по выходной переменной *Zlb*, характеризующей оптические искажения, видимые в проходящем свете, измеренные по методу "Зебра". Исходные данные для построения модели приведены в файле *steklo.exe*. Для построения модели использовать выборку из таблицы функционирования объекта управления, строки с 201 по 300. Регрессионная модель, описывающая «Зебру» имеет следующую структуру: $Zlb=f(Tcm2, Tza, Tzc, Tolsc)$.

Задание 3. Построить математическую модель объекта управления с десятью входными и четырьмя выходными переменными на основе использования пакета для создания и изучения нейросетевых моделей – *STATISTICA Neural Networks*. Оценить точность разработанной модели по выходной переменной *Zlb*, характеризующей оптические искажения, видимые в проходящем свете, измеренные по методу "Зебра". Исходные данные для построения модели приведены в файле *steklo.exe*. Для построения модели использовать выборку из таблицы функционирования объекта управления, строки с 264 по 365. Регрессионная модель, описывающая «Зебру» имеет следующую структуру: $Zlb=f(Tcm2, Tza, Tzc, Tolsc)$.

Задание 4. Построить математическую модель объекта управления с десятью входными и четырьмя выходными переменными на основе использования пакета для создания и изучения нейросетевых моделей – *STATISTICA Neural Networks*. Оценить точность разработанной модели по выходной переменной *R1dol*, характеризующей оптические искажения, видимые в отраженном свете, измеренные в первой долянке. Исходные данные для построения модели приведены в файле *steklo.exe*. Для построения модели использовать выборку из таблицы функционирования объекта управления, строки с 1 по 100. Регрессионная модель, описывающая *R1dol* имеет следующую структуру: $R1dol=f(Tcanal, Tin1R, Tin20R, ToutC)$.

Задание 5. Построить математическую модель объекта управления с десятью входными и четырьмя выходными переменными на основе использования пакета для создания и изучения нейросетевых моделей – *STATISTICA Neural Networks*. Оценить точность разработанной модели по выходной переменной *R1dol*, характеризующей оптические искажения, видимые в отраженном свете, измеренные в первой долянке. Исходные данные для построения модели приведены в файле *steklo.exe*. Для построения модели использовать выборку из таблицы функционирования объекта управления, строки с

101 по 200. Регрессионная модель, описывающая $R1dol$ имеет следующую структуру: $R1dol = f(Tcanal, Tin1R, Tin20R, ToutC)$.

Задание 6. Построить математическую модель объекта управления с десятью входными и четырьмя выходными переменными на основе использования пакета для создания и изучения нейросетевых моделей – *STATISTICA Neural Networks*. Оценить точность разработанной модели по выходной переменной $R1dol$, характеризующей оптические искажения, видимые в отраженном свете, измеренные в первой долянке. Исходные данные для построения модели приведены в файле *steklo.exe*. Для построения модели использовать выборку из таблицы функционирования объекта управления, строки с 201 по 300. Регрессионная модель, описывающая $R1dol$ имеет следующую структуру: $R1dol = f(Tcanal, Tin1R, Tin20R, ToutC)$.

Задание 7. Построить математическую модель объекта управления с десятью входными и четырьмя выходными переменными на основе использования пакета для создания и изучения нейросетевых моделей – *STATISTICA Neural Networks*. Оценить точность разработанной модели по выходной переменной $R2dol$, характеризующей оптические искажения, видимые в отраженном свете, измеренные во второй долянке. Исходные данные для построения модели приведены в файле *steklo.exe*. Для построения модели использовать выборку из таблицы функционирования объекта управления, строки с 1 по 100. Регрессионная модель, описывающая $R2dol$ имеет следующую структуру: $R2dol = f(Tcanal, Tin1R, Tin20R, ToutC)$.

Задание 8. Построить математическую модель объекта управления с десятью входными и четырьмя выходными переменными на основе использования пакета для создания и изучения нейросетевых моделей – *STATISTICA Neural Networks*. Оценить точность разработанной модели по выходной переменной $R2dol$, характеризующей оптические искажения, видимые в отраженном свете, измеренные во второй долянке. Исходные данные для построения модели приведены в файле *steklo.exe*. Для построения модели использовать выборку из таблицы функционирования объекта управления, строки с 101 по 200. Регрессионная модель, описывающая $R2dol$ имеет следующую структуру: $R2dol = f(Tcanal, Tin1R, Tin20R, ToutC)$.

Задание 9. Построить математическую модель объекта управления с десятью входными и четырьмя выходными переменными на основе использования пакета для создания и изучения нейросетевых моделей – *STATISTICA Neural Networks*. Оценить точность разработанной модели по выходной переменной $R2dol$, характеризующей оптические искажения, видимые в отраженном свете, измеренные во второй долянке. Исходные данные для построения модели приведены в файле *steklo.exe*. Для построения модели использовать выборку из таблицы функционирования объекта управления, строки с

201 по 300. Регрессионная модель, описывающая $R2dol$ имеет следующую структуру: $R2dol = f(Tcanal, Tin1R, Tin20R, ToutC)$.

Задание 10. Построить математическую модель объекта управления с десятью входными и четырьмя выходными переменными на основе использования пакета для создания и изучения нейросетевых моделей – *STATISTICA Neural Networks*. Оценить точность разработанной модели по выходной переменной $Razn1$, характеризующей разнотолщинность ленты стекла в первой долянке. Исходные данные для построения модели приведены в файле *steklo.exe*. Для построения модели использовать выборку из таблицы функционирования объекта управления, строки с 1 по 100. Регрессионная модель, описывающая $Razn1$ имеет следующую структуру:

$$Razn1 = f(Tin1R, Tin12R, Tin20R, Tolsc).$$

Задание 11. Построить математическую модель объекта управления с десятью входными и четырьмя выходными переменными на основе использования пакета для создания и изучения нейросетевых моделей – *STATISTICA Neural Networks*. Оценить точность разработанной модели по выходной переменной $Razn1$, характеризующей разнотолщинность ленты стекла в первой долянке. Исходные данные для построения модели приведены в файле *steklo.exe*. Для построения модели использовать выборку из таблицы функционирования объекта управления, строки с 101 по 200. Регрессионная модель, описывающая $Razn1$ имеет следующую структуру:

$$Razn1 = f(Tin1R, Tin12R, Tin20R, Tolsc).$$

Задание 12. Построить математическую модель объекта управления с десятью входными и четырьмя выходными переменными на основе использования пакета для создания и изучения нейросетевых моделей – *STATISTICA Neural Networks*. Оценить точность разработанной модели по выходной переменной $Razn1$, характеризующей разнотолщинность ленты стекла в первой долянке. Исходные данные для построения модели приведены в файле *steklo.exe*. Для построения модели использовать выборку из таблицы функционирования объекта управления, строки с 201 по 300. Регрессионная модель, описывающая $Razn1$ имеет следующую структуру:

$$Razn1 = f(Tin1R, Tin12R, Tin20R, Tolsc).$$

Содержание

| | |
|--|----|
| 1. Лабораторная работа №1. Оценка многокритериальных альтернатив | 3 |
| 2. Лабораторная работа №2. Математические методы оптимизации..... | 8 |
| 3. Лабораторная работа №3. Построение нечетких моделей..... | 14 |
| 4. Лабораторная работа №4. Построение моделей на нейронных сетях | 26 |