Минобрнауки России

Федеральное государственное бюджетное
образовательное учреждение высшего
профессионального образования
«Рыбинский государственный авиационный
технический университет имени П. А. Соловьева»

ЗАОЧНАЯ ФОРМА ОБУЧЕНИЯ

КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ

*Методические указания к выполнению
контрольной работы*

чего делать?

РЫБИНСК
2013

Юдин В.В

УДК 62-83.001.573

Компьютерное моделирование: Методические указания к выполнению контрольной работы / Сост. В. В. Юдин; РГАТУ имени П. А. Соловьева. – Рыбинск, 2013. – 16 с. – (Заочная форма обучения / РГАТУ имени П. А. Соловьева).

Данные методические указания предназначены для выполнения контрольной работы студентами направления 140400 Электротехника и энергетика.

СОСТАВИТЕЛЬ
д-р техн. наук, профессор В. В. Юдин

ОБСУЖДЕНО
на заседании кафедры
«Электротехника и промышленная электроника»

РЕКОМЕНДОВАНО
Методическим Советом РГАТУ имени П. А. Соловьева

Зав. РИО М. А. Салкова

Подписано в печать 24.12.2013 г.
Формат 60×84 1/16. Уч.-изд. л. 1,0. Тираж 50. Заказ 5.

Множительная лаборатория РГАТУ имени П. А. Соловьева
152934, Рыбинск, ул. Пушкина, 53

© РГАТУ имени П. А. Соловьева, 2013

ВВЕДЕНИЕ

Целью изучения дисциплины является подготовка студентов направления подготовки 140400 Электротехника и энергетика в области компьютерного моделирования.

Основными задачами изучения дисциплины являются:

- получение студентами знаний о принципах и методах моделирования в электротехнике, электронной технике и электронике;
- приобретение студентами необходимых навыков в разработке математических моделей;
- применение студентами полученных знаний и навыков при решении практических задач в процессе разработки устройств и систем управления различного назначения.

Дисциплина связана с предшествующими ей курсами математики и физики и требует знания разделов: дифференциальное и интегральное исчисление, дифференциальные уравнения, теория вероятностей, электричество и магнетизм.

Изучение дисциплины рассчитано на 198 часов учебных занятий, из них аудиторных - 16 часов, самостоятельная работа - 186 час. Дисциплина изучается в шестом семестре. Учебным планом предусмотрено выполнение контрольной работы. Форма контроля знаний –экзамен.

Контрольная работа представляет собой комплексную задачу аппроксимации экспериментальных данных степенным полиномом. Задания на контрольную работу выдаются в период установочной сессии. Они представлены в вариантах. Студент выполняет тот вариант, номер которого соответствует его номеру в списке группы.

Контрольная работа представляется на проверку в полном объеме не позднее, чем за три дня до экзамена. Если работа не зачтена, преподаватель указывает, какую часть контрольных заданий переделать или выполнить вновь. Работа оценивается положительно, если правильно выполнено не менее 60% общего объема заданий.

СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

1. *Среда моделирования MATLAB*. Матрицы и работа с ними. Задание векторов и матриц. Использование двоеточия при задании векторов и матриц. Визуализация векторов и матриц. Сложение и умножение матриц. Возведение в степень и обращение матриц. Массивы и работа с ними. Операции поэлементного сложение, вычитание массивов, умножение, деления массивов Степени массивов. Операции отношения. Логические операции. Элементарные математические функции

КОНТРОЛЬНЫЕ ЗАДАНИЯ

Тема: Аппроксимация функциональных зависимостей

Для установления функциональной зависимости между двумя переменными x и y проведен эксперимент, в результате которого получена совокупность из m пар чисел (x_1,y_1), (x_2,y_2), … (x_m,y_m). Для аналитического описания такой зависимости предлагается некоторая функция

$$y = f(x, a_1, a_2, \dots a_n), \tag{1}$$

зависящая от n параметров, числовые значения которых требуется определить на основании эксперимента.

В зависимости от требований к свойствам аппроксимирующей функции можно предложить различные критерии определения коэффициентов. Рассмотрим основные методы, нашедшие наибольшее практическое применение.

Метод выборочных значений

В качестве критерия предлагается требование прохождения функциональной зависимости через конкретные экспериментальные точки. При этом возможны следующие случаи.

Если $n = m$, то для определения коэффициентов составляют систему уравнений

$$\begin{aligned} y_1 &= f(x_1, a_1, a_2, \dots a_n) \\ y_2 &= f(x_2, a_1, a_2, \dots a_n) \\ &\dots\dots\dots\dots\dots\dots \\ y_n &= f(x_n, a_1, a_2, \dots a_n) \end{aligned}$$

Если удается найти решение этой системы относительно коэффициентов $a_1, a_2, \dots a_n$, то оно и определяет конкретное значение функциональной зависимости, соответствующей эксперименту.

Если $n > m$, то в системе число переменных оказывается больше, числа уравнений и задача не имеет решения. Для определения коэффициентов требуются дополнительные данные.

Если $n < m$, то в системе количество определяемых переменных оказывается меньше, чем уравнений. В этом случае из m уравнений оставляют только n, выбирая наиболее важные из экспериментальных данных.

Полученная с помощью этого метода функциональная зависимость лбычно не обладает достаточной гладкостью, имея в промежуточных точках ($x \in [x_i, x_{i+1}]$, $i = 1, \dots n$) существенные отклонения. Это обстоятельство в ряде случаев служит существенным препятствием применения метода.

Метод наименьших квадратов

В соответствии с этим методом коэффициенты аппроксимирующей функции ищутся, исходя из требований ее максимального приближения ко всем экспериментальным точкам, не зависимо от их количества. При этом аппроксимирующая функция в принципе может и не проходить ни через одну точку экспериментальных данных.

В качестве критерия степени приближения возьмем величину среднего квадратичного отклонения аналитической зависимости от экспериментальных данных

$$\theta = \sum_{i=1}^{n} [y_i - f(a_1, a_2, \dots a_m, x_i)]^2 \,. \tag{2}$$

Минимальному значению критерия отвечают условия

$$\begin{aligned} \frac{\partial\theta}{\partial a_1} &= \sum_{i=1}^{n} [y_i - f(a_1, a_2, \dots a_m, x_i)] \frac{\partial}{\partial a_1} f(a_1, a_2, \dots a_m, x_i) = 0 \\ \frac{\partial\theta}{\partial a_2} &= \sum_{i=1}^{n} [y_i - f(a_1, a_2, \dots a_m, x_i)] \frac{\partial}{\partial a_2} f(a_1, a_2, \dots a_m, x_i) = 0 \\ &\dots \\ \frac{\partial\theta}{\partial a_m} &= \sum_{i=1}^{n} [y_i - f(a_1, a_2, \dots a_m, x_i)] \frac{\partial}{\partial a_m} f(a_1, a_2, \dots a_m, x_i) = 0 \end{aligned} \tag{3}$$

Часто аппроксимирующую функцию ищут в виде суммы некоторой совокупности более простых базовых функций $f_1(x), f_2(x), \dots f_m(x)$, т.е.

$$y = f(a_1, a_2, \dots a_m, x) = \sum_{j=1}^{n} a_j f_j(x) \,. \tag{4}$$

В этом случае система уравнений (3) принимает вид

$$\begin{aligned} &\sum_{i=1}^{n} [y_i - \sum_{j=1}^{m} a_j f_j(x_i)] f_1(x_i) = 0 \\ &\sum_{i=1}^{n} [y_i - \sum_{j=1}^{m} a_j f_j(x_i)] f_2(x_i) = 0 \\ &\dots \\ &\sum_{i=1}^{n} [y_i - \sum_{j=1}^{m} a_j f_j(x_i)] f_m(x_i) = 0. \end{aligned} \tag{5}$$

Из (5) следует

$$\sum_{i=1}^{n}[\sum_{j=1}^{m} a_j f_j(x_i) f_1(x_i)] = \sum_{i=1}^{n} y_i f_1(x_i) \qquad (6)$$

$$\sum_{i=1}^{n}[\sum_{j=1}^{m} a_j f_j(x_i) f_2(x_i)] = \sum_{i=1}^{n} y_i f_2(x_i)$$

$$\ldots$$

$$\sum_{i=1}^{n}[\sum_{j=1}^{m} a_j f_j(x_i) f_m(x_i)] = \sum_{i=1}^{n} y_i f_m(x_i)$$

Для получения более компактной формы записи аналитических соотношений введем следующие матричные обозначения. Систему базовых функций $f_1(x), f_2(x), \ldots f_m(x)$ будем рассматривать как матричную функцию скалярного аргумента

$$f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ \ldots \\ f_m(x) \end{bmatrix}.$$

Для числовых значений аргументов, функций и коэффициентов аппроксимирующей зависимости определим соответственно векторы

$$X = [x_1\ x_2\ \ldots\ x_n]$$
$$Y = [y_1\ y_2\ \ldots\ y_n].$$
$$A = [a_1\ a_2\ \ldots\ a_m]$$

Для всей системы базовых функций определим матрицу

$$F = \begin{bmatrix} f_1(x_1) & f_1(x_2) & \ldots & f_1(x_n) \\ f_2(x_1) & f_2(x_2) & \ldots & f_2(x_n) \\ & & \ldots & \\ f_m(x_1) & f_m(x_2) & \ldots & f_m(x_n) \end{bmatrix}. \qquad (7)$$

Минимуму квадратичного отклонения соответствуют коэффициенты, рассчитанные по формуле

$$A = (FF^T)^{-1} FY^T. \qquad (8)$$

При этом величина среднего квадратичного отклонения

$$\theta = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \tilde{y}_i)^2} = \sqrt{\frac{1}{n}\Delta\ \Delta^T} = \sqrt{\frac{1}{n}(Y-\tilde{Y})(Y-\tilde{Y})^T}. \qquad (9)$$

может быть определена по формуле

$$\theta = \sqrt{\frac{1}{n}\left(Y - A^T F\right)\left(Y - A^T F\right)^T} \qquad (10)$$

Метод наименьших квадратов для степенного полинома

В том случае, когда аппроксимирующая функция является степенным полиномом и имеет вид

$$y = a_0 + a_1 x + a_2 x^2 + \ldots + a_{m-1} x^{m-1} = \sum_{j=1}^{m} a_{j-1} x^{j-1}. \qquad (11)$$

система базовых функций определяется матрицей

$$F = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ x_1 & x_2 & \ldots & x_n \\ x_1^2 & x_2^2 & \ldots & x_n^2 \\ & \ldots & & \\ x_1^{m-1} & x_2^{m-1} & \ldots & x_n^{m-1} \end{bmatrix}. \qquad (12)$$

Для произведения матриц $\boldsymbol{F}\,\boldsymbol{F}^T$ и матрицы произведения $\boldsymbol{F}\,\boldsymbol{Y}^T$ при этом получим

$$F\,F^T = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ x_1 & x_2 & \ldots & x_n \\ x_1^2 & x_2^2 & \ldots & x_n^2 \\ & \ldots & & \\ x_1^{m-1} & x_2^{m-1} & \ldots & x_n^{m-1} \end{bmatrix} \begin{bmatrix} 1\ x_1\ x_1^2 \ldots x_1^{m-1} \\ 1\ x_2\ x_2^2 \ldots x_2^{m-1} \\ \ldots \\ 1\ x_n\ x_n^2 \ldots x_n^{m-1} \end{bmatrix} = .$$

$$= \begin{bmatrix} n & \sum\limits_{i=1}^{n} x_i & \sum\limits_{i=1}^{n} x_i^2 & \dots & \sum\limits_{i=1}^{n} x_i^{m-1} \\ \sum\limits_{i=1}^{n} x_i & \sum\limits_{i=1}^{n} x_i^2 & \sum\limits_{i=1}^{n} x_i^3 & \dots & \sum\limits_{i=1}^{n} x_i^{m} \\ \sum\limits_{i=1}^{n} x_i^2 & \sum\limits_{i=1}^{n} x_i^3 & \sum\limits_{i=1}^{n} x_i^4 & \dots & \sum\limits_{i=1}^{n} x_i^{m+1} \\ & & \dots & & \\ \sum\limits_{i=1}^{n} x_i^{m-1} & \sum\limits_{i=1}^{n} x_i^{m} & \sum\limits_{i=1}^{n} x_i^{m+1} & \dots & \sum\limits_{i=1}^{n} x_i^{2m-2} \end{bmatrix}.$$

и

$$F Y^T = \begin{bmatrix} 1 & 1 & \dots & 1 \\ x_1 & x_2 & \dots & x_n \\ x_1^2 & x_2^2 & \dots & x_n^2 \\ & \dots & & \\ x_1^{m-1} & x_2^{m-1} & \dots & x_n^{m-1} \end{bmatrix} \cdot \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \dots \\ y_n \end{bmatrix} = \begin{bmatrix} \Sigma y_i \\ \Sigma x_i y_i \\ \Sigma x_i^2 y_i \\ \dots \\ \Sigma x_i^{m-1} y_i \end{bmatrix}.$$

На основании полученных произведений определим выражение для матрицы коэффициентов аппроксимирующей полиномиальной функции

$$A = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ \dots \\ a_{m-1} \end{bmatrix} = \begin{bmatrix} n & \sum\limits_{i=1}^{n} x_i & \sum\limits_{i=1}^{n} x_i^2 & \dots & \sum\limits_{i=1}^{n} x_i^{m-1} \\ \sum\limits_{i=1}^{n} x_i & \sum\limits_{i=1}^{n} x_i^2 & \sum\limits_{i=1}^{n} x_i^3 & \dots & \sum\limits_{i=1}^{n} x_i^{m} \\ \sum\limits_{i=1}^{n} x_i^2 & \sum\limits_{i=1}^{n} x_i^3 & \sum\limits_{i=1}^{n} x_i^4 & \dots & \sum\limits_{i=1}^{n} x_i^{m+1} \\ & & \dots & & \\ \sum\limits_{i=1}^{n} x_i^{m-1} & \sum\limits_{i=1}^{n} x_i^{m} & \sum\limits_{i=1}^{n} x_i^{m+1} & \dots & \sum\limits_{i=1}^{n} x_i^{2m-2} \end{bmatrix}^{-1} \cdot \begin{bmatrix} \Sigma y_i \\ \Sigma x_i y_i \\ \Sigma x_i^2 y_i \\ \dots \\ \Sigma x_i^{m-1} y_i \end{bmatrix}$$

Для практических вычислений в среде MATLAB удобно пользоваться специальной матрицей, которая носит название матрицы Вандермонда и определяется функцией ***vander(X)***.

Векторный аргумент

$$X = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix}$$

с компонентами

$$w_{ij} = x_i^{n-j}, \quad i, j = 1, 2, \dots n$$

эта функция преобразует в матрицу, столбцы которой являются степенями (в смысле поэлементного возведения) исходного вектора

$$W(X) = [X.\hat{}(n-1) \;\; \dots \;\; X.\hat{}(1) \;\; X.\hat{}(0)] =$$

$$= \begin{pmatrix} x_1^{n-1} & \dots & x_1^1 & x_1^0 \\ x_2^{n-1} & \dots & x_2^1 & x_2^0 \\ \dots & \dots & \dots & \dots \\ x_n^{n-1} & \dots & x_n^1 & x_n^0 \end{pmatrix} = \begin{pmatrix} x_1^{n-1} & \dots & x_1 & 1 \\ x_2^{n-1} & \dots & x_2 & 1 \\ \dots & \dots & \dots & \dots \\ x_n^{n-1} & \dots & x_n & 1 \end{pmatrix}.$$

Приняв во внимание вышеизложенное, вычисление матрицы F может быть осуществлено посредством транспонирования матрицы Вандермонда, с последующим выделением из нее m столбцов т.е.

$$W = vander(X)'; \quad F = W(:,1:m).$$

Например при $m = 2$ для

$$X = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

Имеем

$$W = vander\left(\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}\right)' = \begin{bmatrix} 1 & 1 & 1 \\ 4 & 2 & 1 \\ 9 & 3 & 1 \end{bmatrix}, \quad F = W(:,1:2) = \begin{bmatrix} 1 & 1 \\ 4 & 2 \\ 9 & 3 \end{bmatrix}.$$

Использование этого приема позволяет существенно упростить все дальнейшие вычисления.

Задачи исследования

На основании исходных данных, полученных в приложении 1 и содержащих абсциссы x_1, x_2, x_3, x_4, x_5, x_6 и ординаты y_1, y_2, y_3, y_4, y_5, y_6 шести точек, выполнить аппроксимацию степенными полиномами различной степени, показав влияния степени полинома на его аппроксимирующие свойства.

Для этого необходимо:

- определить с помощью метода наименьших квадратов коэффициенты аппроксимирующих полиномов для различных степеней,
- построить совмещенное графические изображения полиномов и исходных точек..

Кроме того, рекомендуется исследовать зависимость погрешности аппроксимации от степени полинома. Последнее задания не является обязательным и служат для повышения рейтинга студента.

В качестве примера в приложении 2 приведен текст файла *aproxw.m*, содержащего программу аппроксимации экспериментальных данных полиномом пятого порядка. Программа разработана на основании приведенной методики и соответствует варианту 25.

В программе приняты следующие обозначения:

X – матрица абсцисс точек эксперимента,

Y – матрица их ординат,

m – степень аппроксимирующего степенного полинома,

W – матрица Вандермонда,

F – матрица системы базовых функций,

A – матрица коэффициентов аппроксимирующего степенного полинома,

xmax – максимальное значение абсциссы точек эксперимента,

DY – матрица разностей фактических значений ординат и их расчетных значений, вычисленных с помощью аппроксимирующего полинома,

n – число точек эксперимента (число элементов в X и Y),

tetamin – величина среднего квадратичного отклонения расчетных значений ординат аппроксимирующей функции от их фактических значений в точках эксперимента,

kt – количество точек в графике выводимой аппроксимирующей функции,

XN, YN, FN и *YN* – матрицы, соответствующие ранее рассмотренным матрицам *X, Y, F* и *Y*, но определенные для выводимой аппроксимирующей функции,

s1 – символьная переменная, содержащая информацию о порядке полинома,

s2 – символьная переменная, содержащая запятую,

s3 – символьная переменная, содержащая информацию о среднем квадратичном отклонении.

Программы производит вычисление коэффициентов аппроксимирующего полинома, вычисляет значения этого полинома во всем диапазоне изменения аргумента в 40 точках, вычисляет величину среднего квадратичного отклонения и строит график аппроксимирующей функции. Одновременно на график наносятся исходные точки эксперимента в виде звездочек.

В приложении 3 приведен другой вариант программы анализа аппроксимирующих свойств степенного полинома. В основу программы положены следующие функции полиномиальной алгебры, входящие в состав MATLAB:

P=polyfit(XT,YT,N) – функция, определяющая коэффициенты *P* степенного полинома *N*-й степени, наилучшим образом соответствующие исходным данным, заданным векторами *XT* и *YT;*

Y=polyval(P,X) - функция, определяющая значение ординат *Y* степенного полинома, заданного вектором *P*, в точке с абсциссой *X*.

Текст программы находится в файле *aproxp.m,* формирующей цикл аппроксимаций с разными значениями степеней полинома. Результаты расчетов выводятся в различные окна, сформированные функцией *subplot.*

В процесс вычислений программа обращается к функции *aprox*(*XT,YT,N*), осуществляющей решение задачи аппроксимации исходных данных *XT* и *YT* полиномом заданной степени *N* с последующим вычислением значений полинома в промежуточных точках диапазона [*min*(XT),*max*(*XT*)].

С целью уменьшения общей трудоемкости работы при разработке программы анализа студентами могут быть использованы приведенные функции или их фрагменты.

Приложение 1

Варианты задания

Вари-ант	Значения функции y_i					
	y_1	y_2	y_3	y_4	y_5	y_6
Вар.1	0.1	3.2	2.4	4.1	4.9	5.4
Вар.2	0.2	3.1	2.2	4.3	4.8	4.9
Вар.3	0.1	2.9	1.8	4.7	5.0	5.1
Вар.4	0.2	3.2	1.9	3.8	5.1	5.4
Вар.5	0.1	3.1	2.4	3.7	4.9	4.9
Вар.6	0.2	2.9	2.2	4.1	4.8	5.1
Вар.7	0.1	3.2	1.8	4.3	5.0	5.4
Вар.8	0.2	3.1	1.9	4.7	5.1	4.9
Вар.9	0.1	2.9	2.4	3.8	4.9	5.1
Вар.10	0.2	3.2	2.2	3.7	4.8	5.4
Вар.11	0.1	3.1	1.8	4.1	5.0	4.9
Вар.12	0.2	2.9	1.9	4.3	5.1	5.1
Вар.13	0.1	3.2	2.4	4.7	4.9	5.4
Вар.14	0.2	3.1	2.2	3.8	4.8	4.9
Вар.15	0.1	2.9	1.8	3.7	5.0	5.1
Вар.16	0.2	3.2	1.9	4.1	5.1	5.4
Вар.17	0.1	3.1	2.4	4.3	4.9	4.9
Вар.18	0.2	2.9	2.2	4.7	4.8	5.1
Вар.19	0.1	3.2	1.8	3.8	5.0	5.4
Вар.20	0.2	3.1	1.9	3.7	5.1	4.9
Вар.21	0.1	2.9	2.4	4.1	4.9	5.1
Вар.22	0.2	3.2	2.2	4.3	4.8	5.4
Вар.23	0.1	3.1	1.8	4.7	5.0	4.9
Вар.24	0.2	2.9	1.9	3.8	5.1	5.1
Вар.25	0.1	3.2	2.7	4.6	4.9	5.1

шьо вио

Приложение 2

Текстовый файл *aproxw.m* программы аппроксимации данных полиномом пятого порядка для варианта 25.

```
%                <<< aproxw >>>
X=[1 2 3 4 5 6];
Y=[0.1 3.2 2.7 4.6 4.9 5.1];
m=5;
W=flipud(vander(X)');
F=W(1:m,:);
A=(((F*F')^-1)*F)*Y';
xmax=max(X);
DY=Y-A'*F;
n=length(Y);
tetamin=sqrt(DY*DY'/n);
kt=40;
XN=1:xmax/kt:xmax;
WN=flipud(vander(XN)');
FN=WN(1:m,:);
YN=A'*FN;
plot(X,Y,'*',XN,YN,'LineWidth',3)
grid
s1=strcat('m =  ',num2str(m));
s2=',';
s3=strcat('    tetamin =   ',num2str(tetamin));
title(strcat(s1,s2,s3))
```

либо этот расчет здесь 2 шт.

Приложение3

прежде сначала нужно это

```
%================================
%          <<< aproxp >>>
% SUB: aprox
%================================
XT=[1 2 3 4 5 6 7 8 9];
YT=[1 2 4 7 5 3 2 1 0];
for i=1:8
   subplot(2,4,i)
   P=aprox(XT,YT,i);
   tit=strcat('N=',num2str(i));
   title(tit)
end

%-------------------------------------------------------------
%                    <<< aprox >>>
%   Аппроксимация степенным понимномом
%    Построение графика и исходных точек
%    IN: XT,YT - абсциссы и ордигаты точек
%                N - стапень полинома
% OUT: P - коэффициенты полинома
%-------------------------------------------------------------
function P=aprox(XT,YT,N)
P=polyfit(XT,YT,N);
ki=20;
xmax=max(XT);
xmin=min(XT);
dx=(xmax-xmin)/ki;
X=xmin:dx:xmax;
Y=polyval(P,X);
plot(X,Y,XT,YT,XT,YT,'o')
grid
```