

Министерство образования Республики Беларусь
Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»

Кафедра информационных технологий автоматизированных систем

В.С. Муха, Т.В. Служанова

***СТАТИСТИЧЕСКИЕ МЕТОДЫ
ОБРАБОТКИ ДАННЫХ***

Лабораторный практикум
для студентов специальности 53 01 02
«Автоматизированные системы обработки информации»
всех форм обучения

Минск 2004

УДК 330 + 004.451.7 (075.8)
ББК 32.973.202.в6. я 73
М 92

Р е ц е н з е н т :
профессор кафедры высшей математики БГУИР,
кандидат физико-математических наук А.А. Карпук

Муха В.С.

М 92 Статистические методы обработки данных: Лаб. практикум для студ. спец. 53 01 02 «Автоматизированные системы обработки информации» всех форм обучения / В.С. Муха, Т.В. Слюянова. – Мн.: БГУИР, 2004. – 98 с.: ил.
ISBN 985-444-581-X.

Лабораторный практикум содержит описание десяти лабораторных работ. При выполнении работ предполагается использование системы программирования Matlab с пакетом прикладных программ по статистике «Statistics Toolbox». Лабораторные работы содержат теоретическую часть, относящуюся к статистическим методам, описание средств системы Matlab для решения поставленных задач и указания к порядку выполнения работ.

УДК 330 + 004.451.7 (075.8)
ББК 32.973.202.в6. я 73

ISBN 985-444-581-X

© Муха В.С., Слюянова Т.В., 2004
© БГУИР, 2004

ЛАБОРАТОРНАЯ РАБОТА № 1. ОЗНАКОМЛЕНИЕ С СИСТЕМОЙ MATLAB. ОДНОМЕРНЫЕ РАСПРЕДЕЛЕНИЯ ТЕОРИИ ВЕРОЯТНОСТЕЙ И МАТЕМАТИЧЕСКОЙ СТАТИСТИКИ

1.1. Цель работы

1.1.1. Ознакомление с системой программирования Matlab, приобретение навыков работы в ней.

1.1.2. Ознакомление с языком программирования системы Matlab.

1.1.3. Исследование с помощью средств Matlab одномерных распределений теории вероятностей и математической статистики.

1.2. Теоретические положения

1.2.1. Общие сведения о системе Matlab

Matlab (Matrix Laboratory – матричная лаборатория) – универсальная интегрированная система, предлагаемая ее разработчиками как язык программирования высокого уровня для технических вычислений.

Язык программирования Matlab является интерпретатором. Это значит, что каждая инструкция программы распознается и тут же исполняется. Этап компиляции полной программы отсутствует. Интерпретация означает, что Matlab не создает исполняемых конечных программ. Они существуют лишь в виде m-файлов (файлов с расширением m). Для выполнения программ необходимо находиться в среде Matlab. Однако для программ на языке Matlab созданы компиляторы, транслирующие программы на языке Matlab в коды языков программирования C и C++. Это решает проблему создания исполняемых программ, изначально написанных в среде Matlab.

1.2.2. Запуск системы Matlab

Matlab запускается нажатием левой клавиши мыши на ярлыке Matlab в рабочем меню операционной системы Windows. После этого появляется командное окно системы Matlab, и система готова к проведению вычислений в командном режиме. Полезно знать, что в начале запуска автоматически выполняется команда **matlabrc**, которая исполняет загрузочный файл **matlabrc.m** и файл **startup.m**, если таковой существует. Эти файлы выполняют начальную настройку терминала системы и задают ряд ее параметров. Для сохранения собственных m-файлов рекомендуется создать пользовательский каталог, например, каталог с именем USER на диске D. Доступ к этому каталогу необходимо обеспечить с помощью команды **path**, которая будет иметь вид:

path(path,'D:\USER').

Эту команду целесообразно включить в файл **startup.m**, который в свою очередь нужно создать и записать в один из каталогов системы Matlab, например в каталог, в котором размещается файл **matlabrc.m**.

1.2.3. Сеанс работы с Matlab

Сеанс работы с Matlab принято именовать сессией. Сессия, в сущности, является текущим документом, отражающим работу пользователя с системой Matlab. В ней имеются строки ввода, вывода и сообщений об ошибках. Строка ввода указывается с помощью приглашающего символа **>>**. В строке вывода символ **>>** отсутствует. Строка сообщений об ошибках начинается символами **???**. Входящие в сессию определения переменных и функций располагаются в рабочей области памяти (**workspace**). Команды набираются на клавиатуре с помощью обычных операций строчного редактирования. Особое назначение имеют клавиши **↑** и **↓**. Они используются для подстановки после приглашения **>>** ранее введенных строк, например, для их дублирования, исправления или дополнения.

Полезно сразу усвоить следующие команды:

clc – очищает экран и размещает курсор в левом верхнем углу пустого экрана;

clear – уничтожает в рабочем пространстве определения всех переменных;

clear x – уничтожает в рабочем пространстве определение переменной *x*;

clear a,b,c – уничтожает в рабочем пространстве определения переменных списка.

Уничтоженная (стертая в рабочем пространстве) переменная становится неопределенной. Использовать такие переменные нельзя, такие попытки сопровождаются выдачей сообщений об ошибке. По мере задания одних переменных и уничтожения других рабочая область перестает быть непрерывной и содержит «дыры» и всякий «мусор». Во избежание непроизводительных потерь памяти при работе с объемными данными следует использовать команду **pack**, осуществляющую дефрагментацию рабочей области.

1.2.4. Элементы программирования на языке Matlab

Система Matlab ориентирована на работу с матричными переменными. По умолчанию предполагается, что каждая заданная переменная – это матрица. Даже обычные константы и переменные рассматриваются в Matlab как матрицы размером 1×1 .

Простейшей конструкцией языка программирования является оператор присваивания:

Имя_переменной = Выражение

Типы переменных заранее не декларируются. Они определяются выражением, значение которого присваивается переменной. Так, если это выражение – вектор или матрица, то переменная будет векторной или матричной.

После набора оператора в командной строке и нажатия клавиши ENTER на

экран дисплея выводится вычисленное значение переменной. Для блокировки вывода результата вычислений на экран оператор нужно завершить символом ; (точка с запятой).

Пример

```
>> x=2;  
>> y=2;  
>> r=sqrt(x^2+y^2)  
  
r=  
    2.8284
```

Возможна также конструкция, состоящая только из выражения. В этом случае для результата вычислений Matlab назначает переменную с именем **ans**.

Пример

```
>> x=2;  
>> y=2;  
>> sqrt(x^2+y^2)  
  
ans=  
    2.8284
```

Для выполнения арифметических операций в системе Matlab применяются обычные символы: + (сложение), - (вычитание), * (умножение), / (деление), ^ (возведение в степень). Эти операции называются матричными, так как применяются и при работе с матрицами. Наряду с матричными операциями над массивами можно выполнять и поэлементные операции. Для обозначения поэлементных операций используется . (точка), предшествующая обычной (матричной) операции.

Для присваивания значений массиву необходимо значения элементов массива перечислить в квадратных скобках, разделяя их пробелами.

Пример

```
>> v=[1 5 3]  
  
v=
```

1 5 3

В этом примере мы задали вектор v (одномерный массив) со значениями элементов 1,5,3. Задание матрицы (двухмерного массива) требует указания различных строк. Для различения строк используется ; (точка с запятой).

Пример

```
>> m=[1 3 2; 5 6 4; 6 7 8]
```

```
m=
```

```
1 3 2
```

```
5 6 4
```

```
6 7 8
```

Для указания отдельного элемента массива используется имя массива и круглые скобки, внутри которых указываются индексы, разделенные запятыми.

Пример

```
>> m=[1 2 3; 4 5 6; 7 8 9];
```

```
>> m(1,1)=5;
```

```
>> m(3,3)=m(1,1)+m(3,3);
```

```
>> m
```

```
m=
```

```
5 2 3
```

```
4 5 6
```

```
7 8 14
```

Matlab допускает максимум 4096 символов в строке. Если для выражения не хватает одной строки или мы не желаем заходить в невидимую область окна, то выражение можно перенести на новую строку с помощью многоточия ... (3 или более точек). Комментарий в строке должен начинаться символом %.

Пример

```
>> % Пояснение переноса выражения и комментариев
```

```
>> x=2;
```

```
>> y=2;
```

```
>> r=sqrt(x^2+ ... % перенос выражения в следующую строку
y^2)
r=
    2.8284
```

Для формирования упорядоченных числовых последовательностей в Matlab применяется оператор : (двоеточие):

Начальное_значение: Шаг: Конечное_значение

Данная конструкция порождает последовательность (массив) чисел, которая начинается с начального значения, идет с заданным шагом и завершается конечным значением. Если шаг не задан, то он принимает значения 1 или -1.

Пример

```
>> i=1:6
i=
    1 2 3 4 5 6
>>x=0: 0.5: 3
x=
    0 0.5000 1.0000 1.5000 2.0000 2.500 3.0000
>> x=3: -0.5: 0
x=
    3.000 2.5000 2.0000 1.5000 1.0000 0.5000 0
```

1.2.5. Справочная система Matlab

Matlab имеет справочную систему, которая активизируется щелчком левой клавиши мыши на пункте ? главного меню Matlab. Справочная система позволяет ознакомиться с языком программирования Matlab, имеющимися в системе функциями, их назначением и описанием.

1.2.6. Создание m-файлов-сценариев

В Matlab имеется возможность написать программу и сохранить ее в виде m-файла с целью последующего многократного выполнения. **m-файл-сценарий**, именуемый также **script-файлом**, представляет собой, в отличие от **m-файлов-функций**, последовательность команд без входных и выходных параметров. Он имеет следующую структуру:

% Основной комментарий

%Дополнительный комментарий

Тело файла с любыми выражениями

Созданный m-файл включается в справочную систему. Комментарии в m-файле нужны для того, чтобы ознакомиться с назначением файла через справочную систему. Основной комментарий выводится при исполнении команд **lookfor** и **help имя_каталога**. Полный комментарий выводится при исполнении команды **help имя_файла**.

Для создания и отладки m-файла необходимо войти в редактор-отладчик Matlab, выбрав в основном меню командного окна Matlab пункт **Файл**, затем пункты **Создать** и **m-файл**. После раскрытия окна редактора-отладчика необходимо набрать нужные команды программы и сохранить полученный файл с помощью пунктов меню **Файл, Сохранить как...** редактора-отладчика. Для выполнения m-файла необходимо скопировать его в буфер с помощью пунктов **Правка, Выделить все** и **Правка, Копировать** меню редактора-отладчика, вставить файл из буфера в командное окно Matlab с помощью пунктов **Правка, Вставить** меню командного окна Matlab и нажать клавишу ENTER.

1.2.7. Построение графиков функций одной переменной

Для построения графиков функций одной переменной $y = f(x)$ в Matlab имеется функция **plot**. График строится в декартовой системе координат по за-

данным массивам значений аргумента и функции. Заданные этими массивами точки соединяются прямыми линиями. Имеется возможность изменять тип и цвет линии и тип узловых точек (маркер). Вызов этой функции осуществляется командой

plot(x,y,s)

где **x**, **y** – одномерные массивы одинаковой размерности; **x** – массив значений аргумента функции $y = f(x)$; **y** – массив значений функции $y = f(x)$; **s** – строковая константа, определяющая цвет линии, маркер узловых точек и тип линии. Эта константа может содержать от одного до трех символов.

Цвет линии определяется символами **y** (желтый), **m** (фиолетовый), **c** (голубой), **r** (красный), **g** (зеленый), **b** (синий), **w** (белый), **k** (черный).

Тип узловой точки определяется символами **.** (точка), **o** (окружность), **x** (крестик), **+** (плюс), ***** (звездочка), **s** (квадрат), **d** (ромб), **<** **>** **^** (треугольники различной направленности), **p** (пятиугольник), **h** (шестиугольник).

Тип линии определяется символами **-** (непрерывная), **:** (короткие штрихи), **-.** (штрихпунктир), **--** (длинные штрихи).

Символьную константу **s** можно опустить. В этом случае по умолчанию используется непрерывная линия желтого цвета.

Для построения в одном окне нескольких графиков можно использовать команду

plot(x1,y1,s1,x2,y2,s2,x3,y3,s3,...)

Пример

```
% графики функций sin x, cos x
```

```
x=0:0.1:2*pi;
```

```
y1=sin(x);
```

```
y2=cos(x);
```

```
plot(x,y1,'k-o',x,y2,'r--*')
```

В результате выполнения этой программы на экран монитора будет выведено графическое окно с графиками, представленными на рис. 1.1. Графики пред-

ставлены в черно-белой палитре, хотя в действительности график функции $\cos(x)$ выводится красным цветом.

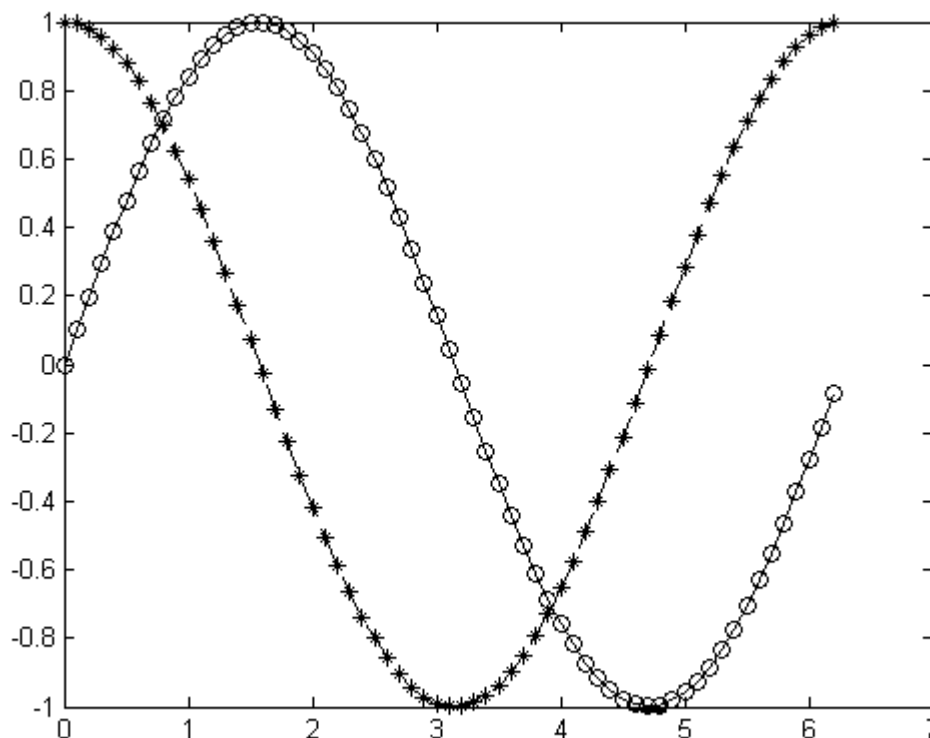


Рис. 1.1. Графики, выполненные с помощью программы plot

Созданный график можно скопировать в буфер Clipboard, активизировав в пункте **Edit** главного меню окна графики команду **Copy Figure**, с целью его дальнейшего редактирования в каком-либо графическом редакторе, например Paint.

1.2.8. Плотности вероятностей некоторых одномерных распределений

1.2.8.1. Равномерное распределение $U(a, b)$:

$$f_{\xi}(x) = \begin{cases} \frac{1}{b-a}, & a < x < b \\ 0, & x \leq a, x \geq b \end{cases}, \quad a, b \in R, a < b.$$

1.2.8.2. Нормальное (гауссовское) распределение $N(a, \sigma^2)$:

$$f_{\xi}(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-a)^2}{2\sigma^2}}, \quad a \in R, \sigma^2 > 0.$$

1.2.8.3. Экспоненциальное распределение $E(\lambda)$:

$$f_{\xi}(x) = \begin{cases} \lambda^{-1} e^{-\lambda^{-1}x}, & x \geq 0, \\ 0, & x < 0, \end{cases} \quad \lambda > 0.$$

1.2.8.4. Распределение χ^2 с k степенями свободы $H_1(k)$:

$$f_{\xi}(x) = \begin{cases} \frac{1}{2\Gamma\left(\frac{k}{2}\right)} \left(\frac{x}{2}\right)^{\frac{k}{2}-1} e^{-\frac{x}{2}}, & x > 0, k \in Z, \\ 0, & x \leq 0. \end{cases}$$

где $\Gamma\left(\frac{k}{2}\right)$ – гамма-функция, которая определяется выражением

$$\Gamma(x) = \int_0^{\infty} y^{x-1} e^{-y} dy.$$

Гамма-функция обладает следующими свойствами:

$$\Gamma(x+1) = x\Gamma(x), \quad \Gamma(k+1) = k!, \quad \Gamma(1) = \Gamma(2) = 1, \quad \Gamma\left(\frac{1}{2}\right) = \sqrt{\pi}.$$

1.2.8.5. Распределение Стьюдента с k степенями свободы $T_1(k)$:

$$f_{\xi}(x) = \frac{\Gamma\left(\frac{k+1}{2}\right)}{\sqrt{k\pi}\Gamma\left(\frac{k}{2}\right)} \left(1 + \frac{x^2}{k}\right)^{-\frac{k+1}{2}}, \quad k \in Z.$$

1.2.8.6. Распределение Фишера с m, k степенями свободы $F_1(m, k)$:

$$f_{\xi}(x) = \begin{cases} \frac{\Gamma\left(\frac{k+m}{2}\right)}{\Gamma\left(\frac{k}{2}\right)\Gamma\left(\frac{m}{2}\right)} k^{\frac{k}{2}} m^{\frac{m}{2}} \frac{x^{\frac{m}{2}-1}}{(k+mx)^{\frac{k+m}{2}}}, & x > 0, k, m \in Z, \\ 0, & x \leq 0. \end{cases}$$

1.2.8.7. Гамма-распределение $\Gamma_1(a, b)$:

$$f_{\xi}(x) = \begin{cases} \frac{1}{\Gamma(a)b^a} x^{a-1} e^{-\frac{x}{b}}, & x > 0, b > 0, a > 0, \\ 0, & x \leq 0. \end{cases}$$

1.2.8.8. Одномерное распределение Уишарта $W_1(k, \sigma^2)$

При $b = 2\sigma^2$, $a = \frac{k}{2}$ гамма-распределение представляет собой одномерное распределение Уишарта $W_1(k, \sigma^2)$ с плотностью вероятности вида

$$f_{\xi}(x) = \begin{cases} \frac{1}{2^{\frac{k}{2}} \Gamma\left(\frac{k}{2}\right) \sigma^k} x^{\frac{k}{2}-1} e^{-\frac{x}{2\sigma^2}}, & x > 0, \sigma^2 > 0, k \in Z \\ 0, & x \leq 0. \end{cases}$$

Легко заметить, что при $\sigma^2 = 1$ распределение $W_1(k, \sigma^2)$ представляет собой распределение H_k .

1.2.9. Средства Matlab для изучения одномерных распределений

В Matlab в пакете статистических программ **stats** (каталог `\Matlab\toolbox\stats`) имеются программы для расчета плотностей вероятности и функций распределения многих известных распределений. Имена функций для расчета плотностей вероятности оканчиваются буквами **pdf** (probability den-

sity function), а для расчета функций распределения – буквами **cdf** (cumulative distribution function).

Функции Matlab для расчета плотностей вероятности

y=chi2pdf(x,k) – расчет значения плотности вероятности распределения χ^2 с k степенями свободы в точке x .

y=exppdf(x,lambda) – расчет значения плотности вероятности экспоненциального распределения с параметром λ в точке x .

y=fpdf(x,m,k) – расчет значения плотности вероятности распределения Фишера с m , k степенями свободы в точке x .

y=gampdf(x,a,b) – расчет значения плотности вероятности гамма-распределения с параметрами a , b в точке x .

y=normpdf(x,a,sigma) – расчет значения плотности вероятности нормального распределения с параметрами a , σ в точке x , a – математическое ожидание, σ – среднее квадратичное отклонение.

y=tpdf(x,k) – расчет значения плотности вероятности распределения Стьюдента с k степенями свободы в точке x .

y=unifpdf(x,a,b) – расчет значения плотности вероятности равномерного в промежутке (a,b) распределения в точке x .

Функции Matlab для расчета функций распределения

y=chi2cdf(x,k) – расчет значения функции распределения распределения χ^2 с k степенями свободы в точке x .

y=expcdf(x,lambda) – расчет значения функции распределения экспоненциального распределения с параметром λ в точке x .

y=fcdf(x,m,k) – расчет значения функции распределения распределения Фишера с m , k степенями свободы в точке x .

$y=\text{gamcdf}(x,a,b)$ – расчет значения функции распределения гамма- распределения с параметрами a , b в точке x .

$y=\text{normcdf}(x,a,\sigma)$ – расчет значения функции распределения нормально-го распределения с параметрами a , σ в точке x , a – математическое ожидание, σ – среднее квадратичное отклонение.

$y=\text{tcdf}(x,k)$ – расчет значения функции распределения распределения Стьюдента с k степенями свободы в точке x .

$y=\text{unifcdf}(x,a,b)$ – расчет значения функции распределения равномерного распределения в промежутке (a,b) распределения в точке x .

Для расчета значений гамма-функции в Matlab имеется функция $y=\text{gamma}(x)$.

1.3. Порядок выполнения работы

1.3.1. Для каждого распределения п. 1.2.8 вывести в одно графическое окно два графика плотности вероятности. Один из графиков плотности вероятности получить по собственной программе, написанной для расчета значений функции плотности вероятности по формулам п. 1.2.8, второй – с использованием функций системы Matlab. Исследовать их зависимость от параметров распределений.

1.3.2. Для каждого распределения п. 1.2.8 вывести в отдельное графическое окно график функции распределения с использованием функций системы Matlab. Исследовать их зависимость от параметров распределений.

ЛАБОРАТОРНАЯ РАБОТА № 2. МНОГОМЕРНЫЕ РАСПРЕДЕЛЕНИЯ ТЕОРИИ ВЕРОЯТНОСТЕЙ И МАТЕМАТИЧЕСКОЙ СТАТИСТИКИ

2.1. Цель работы

2.1.1. Изучение многомерных распределений теории вероятностей и математической статистики.

2.1.2. Исследование многомерных распределений теории вероятностей и математической статистики с помощью средств Matlab.

2.2. Теоретические положения. Плотности вероятностей некоторых многомерных распределений

2.2.1. Многомерное нормальное (гауссовское) распределение $N(A, R)$

Непрерывный случайный вектор $\bar{\xi} = (\xi_1, \dots, \xi_m)$ называется распределенным по нормальному закону, если его плотность вероятности имеет вид

$$f_{\bar{\xi}}(X) = \frac{1}{\sqrt{(2\pi)^m |R|}} \exp\left(-\frac{1}{2} \varphi(X)\right),$$

где

$$\varphi(X) = (X - A)^T R^{-1} (X - A).$$

Здесь приняты следующие обозначения: $X^T = (x_1, \dots, x_m)$ – m -мерный вектор аргументов плотности вероятности; $A^T = (a_1, \dots, a_m)$ – m -мерный вектор параметров; $R = (R_{i,j})$, $i, j = \overline{1, m}$, – симметричная положительно определенная $(m \times m)$ -матрица параметров; R^{-1} – матрица, обратная матрице R ; $|R|$ – определитель матрицы R . Символ T означает транспонирование, так что X – вектор-столбец, а X^T – вектор-строка. Параметры A и R распределения являются

соответственно математическим ожиданием и ковариационной матрицей вектора $\bar{\xi}$.

Уравнение

$$\varphi(X) = (X - A)^T R^{-1} (X - A) = c$$

определяет в R^m гиперповерхность, которая представляет собой эллипсоид. При $c = m + 2$ он называется эллипсоидом рассеяния нормального распределения. Содержательный смысл эллипсоида рассеяния состоит в том, что n -мерное равномерное в данном эллипсоиде распределение имеет то же математическое ожидание A и ту же ковариационную матрицу R , что и данное нормальное распределение. Многомерный (m -мерный) объем эллипсоида рассеяния пропорционален корню квадратному из определителя ковариационной матрицы:

$$V = (m + 2)^{m/2} \pi^{m/2} \sqrt{|R|} / \Gamma\left(\frac{m}{2} + 1\right).$$

При заданных дисперсиях компонент случайного вектора этот объем достигает своего максимума, когда компоненты не коррелированы (матрица R диагональная).

2.2.2. Двухмерное нормальное распределение

Если $\bar{\xi} = (\xi_1, \xi_2)$ – двухмерный случайный вектор, распределенный по нормальному закону, то мы имеем

$$X^T = (x_1, x_2), \quad A^T = (a_1, a_2), \quad R = \begin{pmatrix} R_{1,1} & R_{1,2} \\ R_{2,1} & R_{2,2} \end{pmatrix},$$

$$|R| = R_{1,1}R_{2,2} - R_{1,2}R_{2,1}, \quad R^{-1} = \frac{1}{|R|} \begin{pmatrix} R_{2,2} & -R_{1,2} \\ -R_{2,1} & R_{1,1} \end{pmatrix},$$

$$\begin{aligned} \varphi(X) &= (X - A)^T R^{-1} (X - A) = \\ &= R_{2,2}(x_1 - a_1)^2 - 2R_{1,2}(x_1 - a_1)(x_2 - a_2) + R_{1,1}(x_2 - a_2)^2. \end{aligned}$$

Если здесь обозначить $R_{1,1} = \sigma_1^2$, $R_{2,2} = \sigma_2^2$ и выразить коэффициент ковариации $R_{1,2} = R_{2,1} = \text{cov}(\xi_1, \xi_2)$ через коэффициент корреляции $r_{1,2}$ по формуле

$$R_{1,2} = R_{2,1} = r_{1,2}\sigma_1\sigma_2,$$

то функцию $\varphi(X)$ можно представить в виде

$$\varphi(x_1, x_2) = \left(\frac{1}{(1 - r_{1,2}^2)} \left(\frac{(x_1 - a_1)^2}{\sigma_1^2} - 2r_{1,2} \frac{(x_1 - a_1)(x_2 - a_2)}{\sigma_1\sigma_2} + \frac{(x_2 - a_2)^2}{\sigma_2^2} \right) \right),$$

а плотность вероятности двумерного нормального распределения – в виде

$$f_{\bar{\xi}}(x_1, x_2) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1 - r_{1,2}^2}} \exp\left(-\frac{1}{2}\varphi(x_1, x_2)\right).$$

Линии равного уровня двумерной плотности вероятности, определяемые уравнением

$$f_{\bar{\xi}}(x_1, x_2) = c_1$$

или уравнением

$$\varphi(x_1, x_2) = c,$$

где c_1 и c – некоторые константы,

представляют собой эллипсы в плоскости x_1ox_2 . Уравнение

$$\varphi(x_1, x_2) = 4$$

определяет эллипс рассеяния, площадь которого

$$S = 4\pi\sqrt{|R|} = 4\pi\sqrt{R_{1,1}R_{2,2} - R_{2,1}R_{1,2}}.$$

Для двумерного нормального распределения функция регрессии ξ_2 на ξ_1 определяется выражением

$$x_2 = a_2 + r_{1,2} \frac{\sigma_2}{\sigma_1} (x_1 - a_1),$$

а функция регрессии ξ_1 на ξ_2 – выражением

$$x_1 = a_1 + r_{1,2} \frac{\sigma_1}{\sigma_2} (x_2 - a_2).$$

2.2.3. Произведение одномерных гамма-распределений

$$\Gamma_m((a_1, b_1), \dots, (a_m, b_m))$$

$$f_{\bar{x}}(\bar{x}) = \begin{cases} \prod_{i=1}^m \frac{1}{\Gamma(a_i) b_i^{a_i}} x_i^{a_i-1} e^{-\frac{x_i}{b_i}}, & x_i > 0, b_i > 0, a_i > 0, \\ 0, & x_i \leq 0. \end{cases}$$

2.2.4. Произведение одномерных распределений Уишарта

$$W_m((k_1, \sigma_1^2), \dots, (k_m, \sigma_m^2))$$

При $b_i = 2\sigma_i^2$, $a_i = \frac{k_i}{2}$ произведение гамма-распределений представляет собой произведение одномерных распределений Уишарта.

2.2.5. Произведение одномерных распределений хи-квадрат

$$H_m(k_1, \dots, k_m)$$

При $b_i = 2$, $a_i = \frac{k_i}{2}$ произведение гамма-распределений представляет собой произведение распределений хи-квадрат.

2.2.6. Произведение одномерных экспоненциальных распределений

$$E_m(\lambda_1, \dots, \lambda_m)$$

При $b_i = \lambda_i$, $a_i = 1$ произведение гамма-распределений представляет собой произведение экспоненциальных распределений.

2.2.7. Равномерное распределение в гиперпрямоугольнике

$(a_1, b_1) \times (a_2, b_2) \times \dots \times (a_m, b_m) \ U_m((a_1, b_1), (a_2, b_2), \dots, (a_m, b_m))$

$$f_{\bar{\xi}}(\bar{x}) = \begin{cases} \prod_{i=1}^m \frac{1}{(b_i - a_i)}, & a_i < x_i < b_i, \ a_i < b_i, \\ 0, & \text{иначе.} \end{cases}$$

Многомерные распределения будем изучать с помощью средств трехмерной графики системы программирования Matlab.

2.3. Средства Matlab для изучения многомерных распределений

2.3.1. Создание массивов трехмерной графики

Трехмерные поверхности обычно описываются функцией двух переменных $z = f(x, y)$. Специфика построения трехмерных графиков требует не просто задания ряда значений x и y , то есть векторов \mathbf{x} и \mathbf{y} , а определения двумерных массивов \mathbf{X} и \mathbf{Y} . Для создания таких массивов служит функция **meshgrid**.

$[\mathbf{X}, \mathbf{Y}] = \text{meshgrid}(\mathbf{x}, \mathbf{y})$ – преобразует область, заданную векторами \mathbf{x} и \mathbf{y} , в двумерные массивы \mathbf{X} и \mathbf{Y} , которые могут быть использованы для вычисления значений функции двух переменных и построения трехмерных графиков. Эта функция формирует массивы \mathbf{X} и \mathbf{Y} таким образом, что строки выходного массива \mathbf{X} являются копиями вектора \mathbf{x} , а столбцы выходного массива \mathbf{Y} – копиями вектора \mathbf{y} .

Пример

```
[X,Y]=meshgrid(1:1:4,6:1:9)
```

X =

```
1  2  3  4
1  2  3  4
```

1	2	3	4
1	2	3	4

Y =

6	6	6	6
7	7	7	7
8	8	8	8
9	9	9	9

В этом примере формируются массивы X и Y для построения трехмерной поверхности при изменении x от 1 до 4 с шагом 1 и y от 6 до 9 с шагом 1.

2.3.2. Построение контурных графиков

Контурные графики являются попыткой отобразить на плоскость функцию двух переменных $z = f(x, y)$. Такую функцию можно представить в виде совокупности линий равного уровня, которые получаются, если трехмерная поверхность пересекается рядом плоскостей, расположенных параллельно друг другу. При этом контурный график представляет совокупность спроектированных на плоскость xy линий пересечения поверхности $z = f(x, y)$ плоскостями.

Для построения контурных графиков используется команда **contour**.

contour(x,y,z,n) строит контурный график по данным матрицы **z** с указанием спецификаций для **x** и **y** с заданием **n** линий равного уровня.

contour(x,y,z,v) строит линии равного уровня для высот, указанных значениями элементов вектора **v**.

Пример

```
[x,y]=meshgrid(-3: .2: 3, -3: .2: 3);
z=x.^2+y.^2;
contour(x,y,z,8)
grid on
```

По этой программе на экран монитора будет выведено восемь графиков (рис. 2.1). Сетка на рисунок нанесена с помощью команды `grid on`.

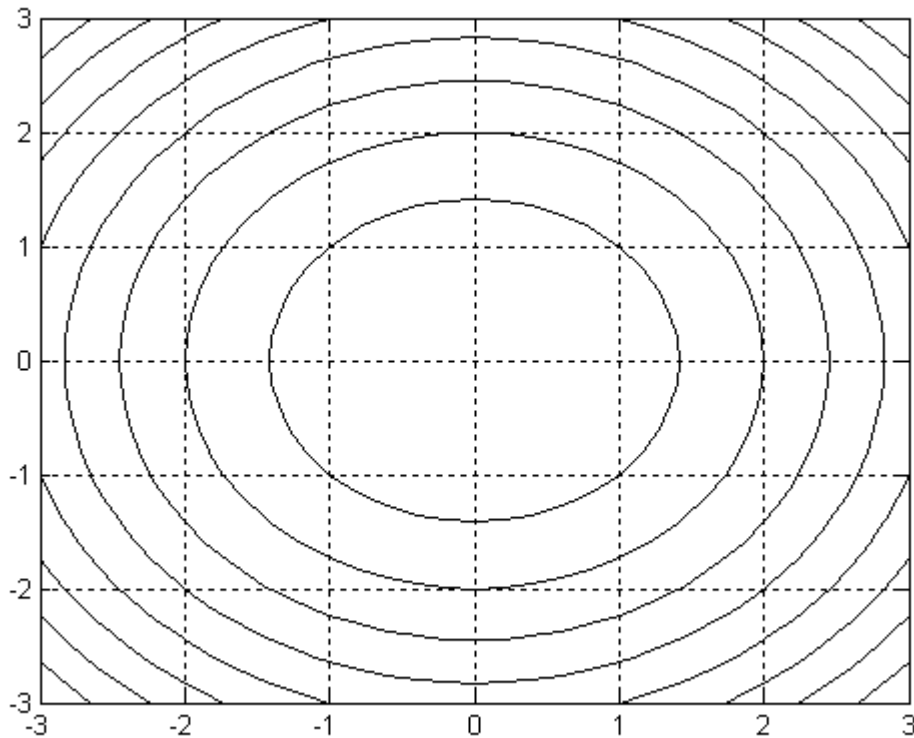


Рис. 2.1. Контурные графики, построенные с помощью функции `contour`

2.3.3. Построение графиков трехмерных поверхностей

Команда **plot3(...)** является аналогом команды **plot(...)**, но относится к функции двух переменных $z = f(x, y)$. Она строит аксонометрическое изображение трехмерных (3D) поверхностей.

plot3(X,Y,Z) вычерчивает различные строки, полученные из столбцов двухмерных массивов **X,Y,Z** одинаковой размерности. Массивы **X** и **Y** можно получить с помощью **meshgrid**.

Пример

```
[x,y]=meshgrid(-2:0.1:2,-2:0.1:2);
z=x.^2+y.^2;
plot3(x,y,z)
grid on
```

По этой программе будет выведена фигура, представленная на рис. 2.2.

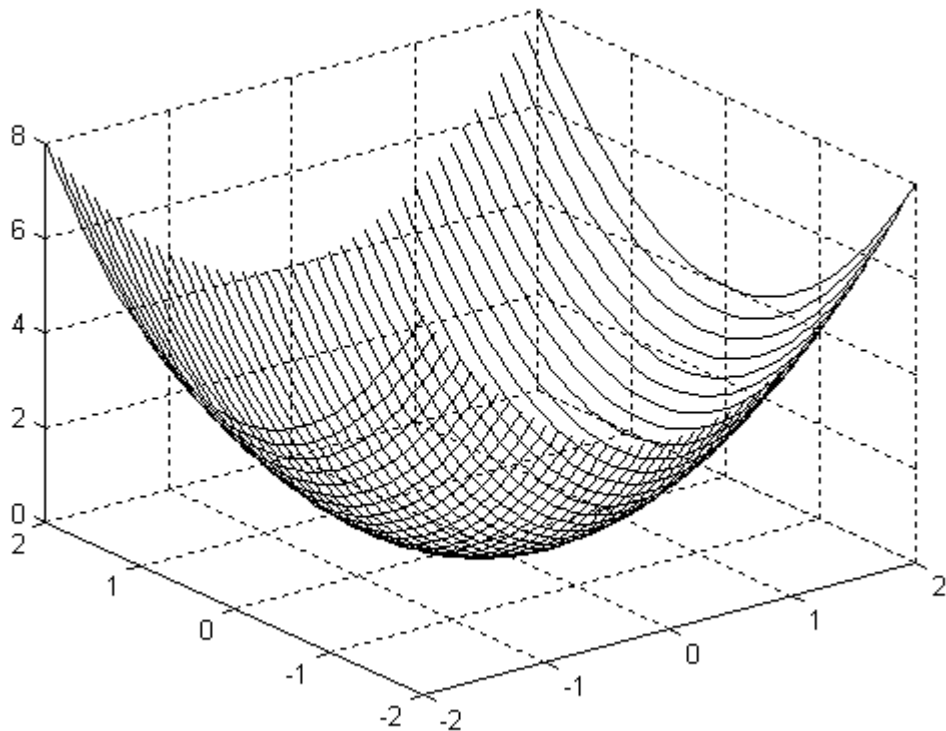


Рис. 2.2. Трехмерный график, построенный с помощью функции `plot3`

Однако более наглядными являются сеточные графики трехмерных поверхностей с заданной или функциональной окраской. Такие графики выполняются командой **mesh**.

mesh(X,Y,Z,C) – выводит в графическое окно сетчатую поверхность с цветами узлов поверхности, заданных массивом **C**.

mesh(X,Y,Z) – аналог предшествующей команды при **C=Z**, с использованием функциональной окраски, при которой цвет задается высотой поверхности.

Пример

```
[x,y]=meshgrid(-2:0.1:2,-2:0.1:2);
```

```
z=x.^2+y.^2;
```

```
mesh(x,y,z)
```

В результате выполнения этой программы на экран будет выведена фигура, представленная на рис. 2.3. Рисунок представлен в черно-белой палитре, хотя в действительности он формируется в цветной палитре.

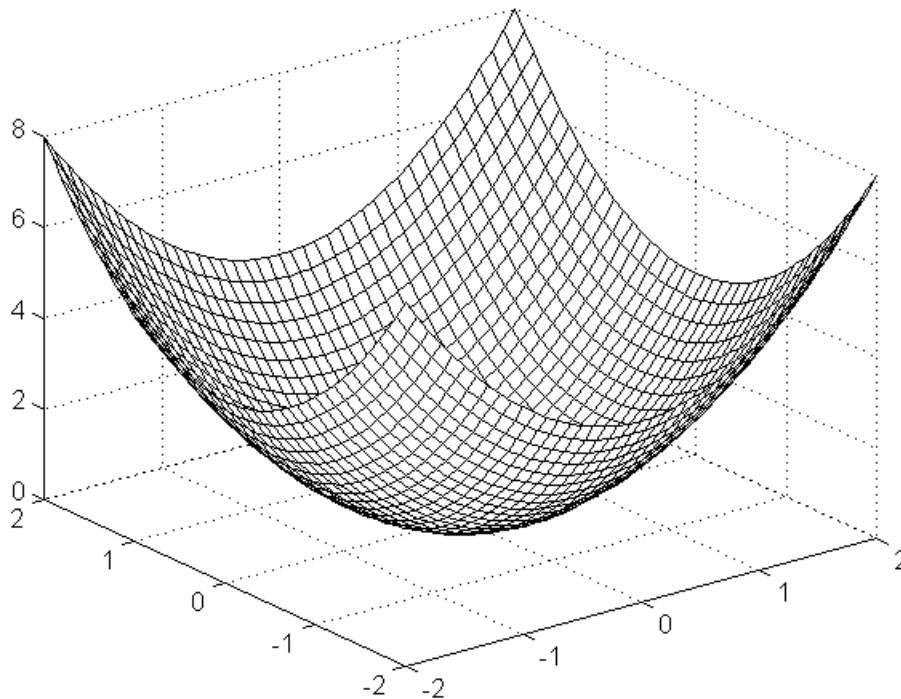


Рис. 2.3. Трехмерный график, построенный с помощью функции meshgrid

2.3.4. Продолжение построений графиков

Во многих случаях желательно построение ряда наложенных друг на друга графиков в одном и том же окне. Такую возможность обеспечивает команда продолжения графических построений **hold**.

hold on обеспечивает продолжение вывода графиков в текущее окно графики, что позволяет добавлять последующие графики к уже существующему.

hold off отменяет режим продолжения графических построений.

Пример

```
t=-3:0.2:3;
```

```
[x,y]=meshgrid(t,t);
```

```
z=x.^2+y.^2;
```

```
contour(x,y,z,8)
```

```
hold on
```

```
v=2*t;
```

```
plot(t,v,'k.-')
```

```
grid on
```

```
hold off
```


В этом примере на контурный график функции $z = x^2 + y^2$ наносится график прямой линии $y = 2x$ (рис. 2.4).

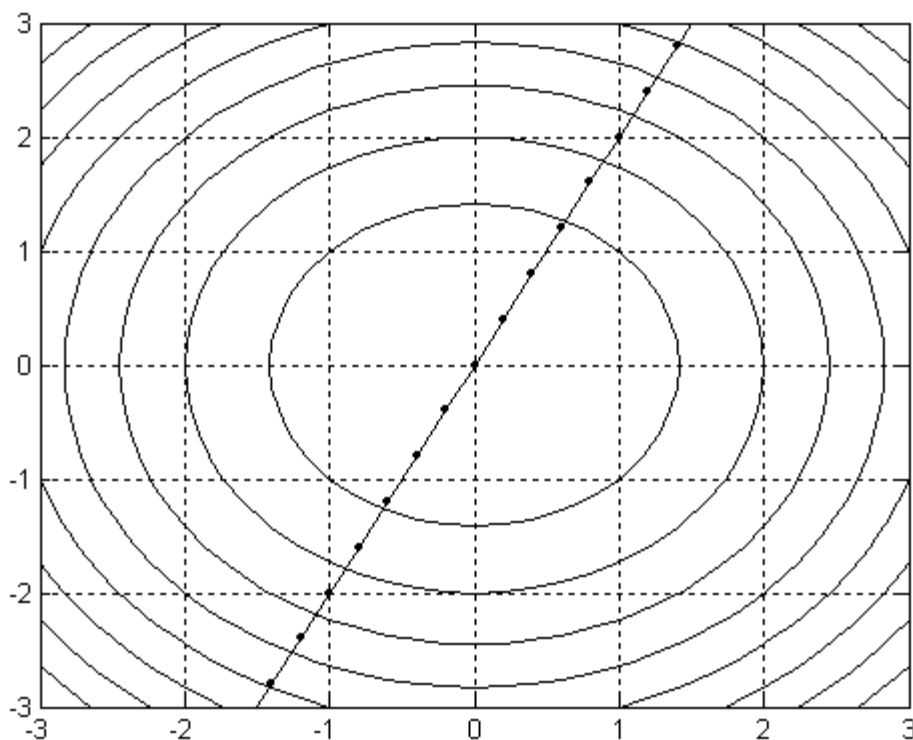


Рис. 2.4. Два графика в одном графическом окне

2.4. Порядок выполнения работы

2.4.1. Вывести на экран монитора графики поверхностей и линии равных уровней плотностей вероятности приведенных выше двумерных распределений (при $k = 2$) и исследовать их зависимость от параметров распределений.

2.4.2. Для нормального распределения в одно графическое окно вывести эллипс рассеяния и две функции регрессии. Исследовать зависимость формы и площади эллипса рассеяния от коэффициента корреляции при заданных дисперсиях компонент случайного вектора. Исследовать взаимное расположение функций регрессии и осей эллипса рассеяния (совпадают ли функции регрессии с осями эллипса?).

ЛАБОРАТОРНАЯ РАБОТА № 3. МОДЕЛИРОВАНИЕ ОДНОМЕРНЫХ СЛУЧАЙНЫХ ЧИСЕЛ

3.1. Цель работы

3.1.1. Изучение методов моделирования одномерных случайных чисел.

3.1.2. Приобретение навыков моделирования одномерных случайных чисел в системе Matlab.

3.2. Теоретические положения

Случайные числа с различными законами распределения обычно моделируются с помощью преобразований одного или нескольких независимых значений базовой случайной величины α . Базовая случайная величина α – это случайная величина с распределением $U(0,1)$ (равномерным распределением в интервале $(0,1)$). В любой системе программирования имеется стандартная программа моделирования базовой случайной величины. Независимые случайные величины с распределением $U(0,1)$ будем обозначать символами $\alpha_1, \alpha_2, \dots$.

3.2.1. Равномерное распределение $U(a,b)$

$$f_{\xi}(x) = \begin{cases} \frac{1}{b-a}, & a < x < b, \quad a, b \in R, \quad a < b \\ 0, & x \leq a, \quad x \geq b. \end{cases}$$

Алгоритм 1. $\xi = a + (b - a)\alpha$.

3.2.2. Нормальное (гауссовское) распределение $N(a, \sigma^2)$

$$f_{\xi}(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-a)^2}{2\sigma^2}}, \quad a \in R, \quad \sigma^2 > 0.$$

Алгоритм 1. Зарезервирована константа $c = 2\pi$. 1) $r = \sqrt{-2 \ln \alpha_1}$; 2) $\varphi = c\alpha_2$; 3) $\xi_1 = r \cos \varphi$, $\xi_2 = r \sin \varphi$; 4) $\xi_1 = a + \sigma\xi_1$, $\xi_2 = a + \sigma\xi_2$.

Алгоритм 2. 1) $v_1 = 2\alpha_1 - 1$, $v_2 = 2\alpha_2 - 1$; 2) $s = v_1^2 + v_2^2$; 3) Если $s \geq 1$, вернуться к п. 1; 4) $r = \sqrt{-(2 \ln s)/s}$; 5) $\xi_1 = v_1 r$, $\xi_2 = v_2 r$; 6) $\xi_1 = a + \sigma\xi_1$, $\xi_2 = a + \sigma\xi_2$.

В качестве случайного числа можно взять любое из чисел ξ_1 , ξ_2 .

3.2.3. Экспоненциальное распределение $E(\lambda)$

$$f_{\xi}(x) = \begin{cases} \lambda^{-1} e^{-\lambda^{-1}x}, & x \geq 0, \lambda > 0, \\ 0, & x < 0. \end{cases}$$

Алгоритм 1. $\xi = -\lambda \ln \alpha$.

3.2.4. Распределение χ^2 с k степенями свободы $H_1(k)$

$$f_{\xi}(x) = \begin{cases} \frac{1}{2\Gamma\left(\frac{k}{2}\right)} \left(\frac{x}{2}\right)^{\frac{k}{2}-1} e^{-\frac{x}{2}}, & x > 0, k \in Z, \\ 0, & x \leq 0. \end{cases}$$

Алгоритм 1. $\xi = \sum_{i=1}^k u_i^2$, где u_1, \dots, u_k – независимые случайные величины с распределением $N(0,1)$.

3.2.5. Распределение Стьюдента с k степенями свободы $T_1(k)$

$$f_{\xi}(x) = \frac{\Gamma\left(\frac{k+1}{2}\right)}{\sqrt{k\pi}\Gamma\left(\frac{k}{2}\right)} \left(1 + \frac{x^2}{k}\right)^{-\frac{k+1}{2}}, \quad k \in Z.$$

Алгоритм 1. $\xi = \frac{u}{\sqrt{v/k}}$, где u и v – независимые случайные величины, причем $u \in N(0,1)$, $v \in H_1(k)$.

3.2.6. Распределение Фишера с m, k степенями свободы $F_1(m, k)$

$$f_{\xi}(x) = \begin{cases} \frac{\Gamma\left(\frac{k+m}{2}\right)}{\Gamma\left(\frac{k}{2}\right)\Gamma\left(\frac{m}{2}\right)} k^{\frac{k}{2}} m^{\frac{m}{2}} \frac{x^{\frac{m}{2}-1}}{(k+mx)^{\frac{k+m}{2}}}, & x > 0, k, m \in Z, \\ 0, & x \leq 0. \end{cases}$$

Алгоритм 1. $\xi = \frac{v/m}{w/k}$, где v и w – независимые случайные величины с распределениями $H_1(m)$ и $H_1(k)$ соответственно.

3.2.7. Одномерное распределение Уишарта $W_1(k, \sigma^2)$

$$f_{\xi}(x) = \begin{cases} \frac{1}{2^{\frac{k}{2}} \Gamma\left(\frac{k}{2}\right) \sigma^k} x^{\frac{k}{2}-1} e^{-\frac{x}{2\sigma^2}}, & x > 0, \sigma^2 > 0, k \in Z, \\ 0, & x \leq 0. \end{cases}$$

Алгоритм 1. $\xi = \sum_{i=1}^k u_i^2$, где u_1, \dots, u_k – независимые случайные величины с распределением $N(0, \sigma^2)$.

3.3. Средства Matlab для моделирования одномерных случайных чисел

3.3.1. Создание m-файлов-функций

В Matlab имеется возможность написать программу и сохранить ее в ви-

де m-файла-сценария или m-файла-функции с целью последующего многократного выполнения. **М-файл-функция** является типичным объектом языка программирования системы Matlab. Структура m-файла-функции *с одним выходным параметром* выглядит следующим образом:

```
function var=f_name(список параметров)
```

```
% Основной комментарий
```

```
%Дополнительный комментарий
```

```
Тело файла с любыми выражениями
```

```
var=выражение
```

Здесь переменная **var** – выходной параметр, **f_name** – имя функции.

Функция возвращает свое значение **var** и может использоваться в математических выражениях в виде **f_name(список параметров)**.

Все переменные, имеющиеся в теле файла-функции, являются *локальными*, то есть действуют только в пределах тела функции, в отличие от файла-сценария, все переменные которого являются *глобальными*.

Правила вывода комментариев те же, что и у файлов-сценариев.

Последняя конструкция **var=выражение** вводится, если требуется, чтобы функция возвращала результат вычислений. Если m-файл-функция завершается строкой с точкой с запятой (;), то для возврата значения функции используется программный оператор **return**.

Если *выходных параметров больше одного*, то структура модуля имеет вид:

```
function [var1,var2,...]=f_name(список параметров)
```

```
%Основной комментарий
```

```
%Дополнительный комментарий
```

```
Тело файла с любыми выражениями
```

```
var1=выражение
```

```
var2=выражение
```

```
.....
```

Здесь **var1,var2,...** – имена переменных, которые являются выходными параметрами.

Такую функцию нельзя использовать в математических выражениях, поскольку она возвращает не один результат. Данная функция используется (вызывается) как отдельный элемент программы в виде:

[var1,var2,...]=f_name(список параметров).

Если такая функция используется в виде **f_name(список параметров)**, то возвращается значение только первого выходного параметра – переменной **var1**.

Если внутри функции целесообразно использовать глобальные переменные, то их нужно объявить с помощью команды

global var1 var2 ...

Начиная с версии 5.0, в функции системы Matlab можно включать подфункции. Они имеют такую же структуру, как и основная функция, и записываются в теле основной функции.

Для создания и отладки m-файла-функции необходимо войти в редактор-отладчик Matlab, выбрав в меню командного окна Matlab пункт **Файл**, затем пункты **Создать** и **M-файл**. После раскрытия окна редактора-отладчика необходимо набрать нужные команды программы, отредактировать их и сохранить полученный файл под именем **f_name** с помощью пунктов меню **Файл, Сохранить как...** редактора-отладчика.

3.3.2. Управляющие структуры языка программирования системы Matlab

3.3.2.1. Диалоговый ввод-вывод

Disp(X) отображает массив, не печатая имя массива. Если X – строка, то отображается текст.

Пример

```
x=[1 2 3];
```

```
disp(x)
```

```
1 2 3
```

```
disp('квадрат второго элемента=')
```

```
квадрат второго элемента=
```

```
disp(x(2)^2)
```

```
4
```

R=INPUT('Сколько яблок?') дает пользователю приглашение в текстовой строке и затем ожидает ввода с клавиатуры. Может быть введено любое Matlab-выражение, которое вычисляется, используя переменные в текущей рабочей области, и результат возвращается в R. Если пользователь нажимает клавишу возврата каретки, ничего не вводя, то вводится пустая матрица.

R=INPUT('введите ваше имя', 's') дает приглашение в текстовой строке и ожидает ввода символьной строки. Напечатанный текст не вычисляется; символы просто возвращаются как Matlab-строка.

Пример

```
r=input('введите угол в радианах')
```

```
введите угол в радианах 2*pi
```

```
r =
```

```
6.2832
```

```
r=input('введите ваше имя','s')
```

```
введите ваше имя 2*pi
```

```
r =
```

```
2*pi
```

3.3.2.2. Циклы типа for-end

Циклы типа **for-end** обычно используются для организации вычислений с

заданным числом повторений цикла. Конструкция такого цикла имеет вид:

```
for var=выражение  
    Инструкция,..., Инструкция  
end
```

Выражение чаще всего записывается в виде $b:s:e$, где b – начальное значение переменной цикла var , s – приращение (шаг) этой переменной и e – конечное значение управляющей переменной, при достижении которого цикл завершается. Возможна запись выражения в виде $b:e$, в этом случае $s=1$. Список выполняемых в цикле инструкций завершается оператором **end**.

Для досрочного выполнения цикла можно использовать оператор **break**. Как только этот оператор встречается в программе, цикл прерывается.

Возможно использование цикла в цикле.

Пример

```
for i=1:3  
    for j=1:3  
        a(i,j)=i+j;  
    end  
end
```

a

a =

```
2  3  4  
3  4  5  
4  5  6
```

3.3.2.3. Циклы типа while_end

```
while Условие  
    Инструкции  
end
```


Цикл типа **while** выполняется до тех пор, пока выполняется Условие. Для прекращения выполнения цикла можно использовать оператор **break**.

Пример

```
x=1;i=1;
  while x<=3
    y(i)=x;
    x=x+0.5;
    i=i+1;
  end
y
y =
  1.0000  1.5000  2.0000  2.5000  3.0000
```

3.3.2.4. Условный оператор if-elseif-else-end

Условный оператор **if** в общем виде записывается следующим образом:

```
if Условие
  Инструкции 1
elseif Условие
  Инструкции 2
else
  Инструкции 3
end
```

Эта конструкция допускает несколько частных вариантов. Простейший из них – следующий:

```
if Условие
  Инструкции
end
```

Данный оператор работает следующим образом. Пока Условие возвращает логическое значение 1 (то есть выполняется), выполняются Инструкции. Опе-

ратор **end** указывает на конец списка Инструкций. Инструкции в списке разделяются запятыми или точками с запятыми. Если Условие возвращает логическое значение 0 (то есть не выполняется), то Инструкции также не выполняются.

Еще один вариант:

if Условие

 Инструкции 1

else

 Инструкции 2

end

В этом варианте выполняются Инструкции 1, если выполняется Условие 1, или Инструкции 2 в противном случае.

Условие в операторе **if** записывается в виде:

Выражение_1 Оператор_отношения Выражение_2

В качестве Оператора_отношения используются следующие логические операторы: **==**, **<**, **>**, **<=**, **>=**, **~=**. Двойные символы не имеют между собой пробелов.

Пример

```
for i=1:3
    for j=1:3
        if i==j
            a(i,j)=2;
        elseif abs(i-j)==1
            a(i,j)=-1;
        else
            a(i,j)=0;
        end
    end
end
a
a =
```

```
2 -1 0
-1 2 -1
0 -1 2
```

3.3.2.5. Переключатель switch-case-otherwise-end

Для осуществления множественного выбора (или ветвления) используется конструкция с переключателем типа **switch**:

```
switch switch_Выражение
case case_Выражение
    Список_инструкций
case { case_Выражение1, case_Выражение2,... }
    Список_инструкций
...
otherwise,
    Список_инструкций
end
```

Выполняется первый оператор **case**, у которого case_Выражение соответствует switch_Выражению. Если ни одно из case_Выражений не соответствует switch_Выражению, то выполняется список инструкций после оператора **otherwise** (если он существует). Выполняется только один **case**, после чего выполнение продолжается с оператора после **end**.

Пример

Пусть существует m-файл-сценарий swit.m:

```
switch month
    case {1,2,3}
        disp('Первый квартал')
    case {4,5,6}
        disp('Второй квартал')
```

```
case {7,8,9}
    disp('Третий квартал')
case {10,11,12}
    disp('Четвертый квартал')
otherwise,
    disp('Ошибка в данных')
end
```

Эта программа в ответ на значения переменной `month` (номер месяца) определяет номер квартала и выводит сообщение. Как это происходит, видно из следующей программы:

```
month=3;

swit

Первый квартал

month=10;

swit

Четвертый квартал

month=13;

swit

Ошибка в данных
```

3.3.2.6. Создание паузы в вычислениях

Для остановки программы используется оператор **pause** в следующих формах:

pause – останавливает вычисления до нажатия любой клавиши.

pause(N) – останавливает вычисления на N секунд.

pause on – включает режим создания пауз.

pause off – выключает режим создания пауз.

3.3.3. Стандартные функции Matlab для моделирования одномерных случайных чисел

y=chi2rnd(k) – χ^2 -распределение.

y=exprnd(lambda) – экспоненциальное распределение.

y=frnd(m,k) – распределение Фишера.

y=gamrnd(a,b) – гамма-распределение.

y=normrnd(a,sigma) – нормальное распределение.

y=trnd(k) – распределение Стьюдента.

y=unifrnd(a,b) – равномерное распределение.

y=rand(m,k) – моделирует $(m \times k)$ -матрицу со случайными данными, выбранными из равномерного распределения в интервале $(0,1)$.

r=unidrnd(k) – возвращает матрицу случайных чисел, выбранных из набора $\{1,2,\dots,k\}$. Размер r является размером k .

r=unidrnd(k,mm,nn) – возвращает $(mm \times nn)$ -матрицу случайных чисел, выбранных из набора $\{1,2,\dots,k\}$.

3.4. Порядок выполнения работы

3.4.1. Выполнить моделирование случайных чисел с указанными в п. 3.2 распределениями. Для каждого распределения вывести по 100 случайных чисел, используя собственную программу, реализующую предложенный алгоритм, и стандартную программу Matlab. Собственные программы оформить в виде m -файлов-функций. Случайные числа вывести в виде точек на действительной прямой.

3.4.2. Для каждой выборки вычислить с помощью функции **function [xmean,s2,s3,s4,xmin,xmax,wtsun,wt,iwt,ifail]=g01aaf(x<,wt,iwt,ifail>)** выборочные среднее **xmean**, среднее квадратичное отклонение **s2**, коэффициент асимметрии **s3**, коэффициент эксцесса **s4**, минимальное значение выборки **xmin**, максимальное значение выборки **xmax**, сумму весов **wtsun** по данным x_1, x_2, \dots, x_n , помещенным в векторе **x** и имеющим соответствующие веса w_1, w_2, \dots, w_n , помещенные в векторе **wt**. Если присваивания весов не требуется, то параметр **wt** не указывается, при этом веса устанавливаются равными 1. Параметр **iwt=0**.

Пример

n=4;

iwt = 0;

x = [193.0 215.0 112.0 161.0];

[xmean,s2,s3,s4,xmin,xmax,wtsun,wt,iwt,ifail] = g01aaf(x);

xmean,s2,s3,s4,xmin,xmax,wtsun,ifail

ЛАБОРАТОРНАЯ РАБОТА № 4. МОДЕЛИРОВАНИЕ МНОГОМЕРНЫХ СЛУЧАЙНЫХ ЧИСЕЛ

4.1. Цель работы

4.1.1. Изучение методов моделирования многомерных случайных чисел.

4.1.2. Приобретение навыков моделирования многомерных случайных чисел в системе Matlab.

4.2. Теоретические положения

4.2.1. Моделирование случайных чисел с многомерным нормальным (гауссовским) распределением

Пусть требуется моделировать значения m -мерного случайного вектора $\bar{\xi} = (\xi_1, \dots, \xi_m)$, распределенного по нормальному закону $N(A, R)$ с математическим ожиданием A и ковариационной матрицей R ,

$$A = (a_i), \quad R = (\sigma_{i,j}), \quad i, j = \overline{1, m}.$$

Обозначим $\bar{\eta} = (\eta_1, \dots, \eta_m)$ стандартный гауссовский m -мерный случайный вектор, то есть случайный вектор, распределенный по нормальному закону $N(0, I)$, где I – единичная матрица. Методы моделирования базируются на следующем результате.

Теорема. Пусть $C = (c_{i,j})$ – действительная $(m \times m)$ -матрица, являющаяся решением матричного уравнения

$$CC^T = R. \quad (4.1)$$

Тогда случайный вектор $\bar{\xi}$, являющийся линейным преобразованием $\bar{\eta}$,

$$\bar{\xi} = C\bar{\eta} + A, \quad (4.2)$$

имеет нормальное распределение $N(A, R)$.

Моделирование $\bar{\eta}$ легко осуществляется. Действительно, компоненты этого вектора η_1, \dots, η_m некоррелированы, следовательно, и независимы, распределение отдельной компоненты η_i – стандартное нормальное $N(0,1)$. Поэтому моделирование $\bar{\eta}$ можно выполнить m -кратным обращением к функции моделирования случайного числа с одномерным стандартным нормальным распределением $N(0,1)$.

Различные методы моделирования $\bar{\xi}$, известные в литературе, отличаются лишь способом построения матрицы C .

Один из методов использует неединственность решения (4.1) и требует, чтобы C была нижней треугольной матрицей: $c_{i,j} = 0$, если $j > i$. Ненулевые элементы $c_{i,j}$ определяются рекуррентно. Действительно, в силу (4.2) $\xi_1 = c_{1,1}\eta_1 + a_1$. Выражение (4.1) дает соотношение $c_{1,1}^2 = \sigma_{1,1}$. Следовательно, $c_{1,1} = \sqrt{\sigma_{1,1}}$. Теперь из (4.2) получим

$$\xi_2 = c_{2,1}\eta_1 + c_{2,2}\eta_2 + a_2,$$

а из (2.1) имеем

$$\begin{aligned} c_{2,1}^2 + c_{2,2}^2 &= \sigma_{2,2}, \\ c_{2,1}c_{1,1} &= \sigma_{1,2}. \end{aligned}$$

Отсюда

$$c_{2,1} = \frac{\sigma_{1,2}}{c_{1,1}}, \quad c_{2,2} = \sqrt{\sigma_{2,2} - c_{2,1}^2},$$

и т.д. Справедлива общая рекуррентная формула

$$c_{i,j} = (\sigma_{i,j} - \sum_{v=1}^{j-1} c_{i,v}c_{j,v}) / \sqrt{\sigma_{j,j} - \sum_{v=1}^{j-1} c_{j,v}^2}. \quad (4.3)$$

Здесь

$$\sum_{v=1}^0 (\cdot) = 0$$

и вычисления по рекуррентной формуле (4.3) осуществляются по строкам мат-

рицы C , то есть в следующем порядке: $c_{1,1}, c_{2,1}, c_{2,2}, c_{3,1}, c_{3,2}, c_{3,3}, c_{4,1}, \dots, c_{m,m}$.

Таким образом, моделирующий алгоритм определяется формулами (4.3), (4.2).

4.2.2. Моделирование случайных чисел с многомерным распределением, равным произведению одномерных гамма-распределений $\Gamma_m((a_1, b_1), \dots, (a_m, b_m))$

$$f_{\bar{\xi}}(\bar{x}) = \begin{cases} \prod_{i=1}^m \frac{1}{\Gamma(a_i) b_i^{a_i}} x_i^{a_i-1} e^{-\frac{x_i}{b_i}}, & x_i > 0, b_i > 0, a_i > 0, \\ 0, & x_i \leq 0. \end{cases}$$

4.2.3. Моделирование случайных чисел с многомерным распределением, равным произведению одномерных распределений Уишарта $W_m((k_1, \sigma_1^2), \dots, (k_m, \sigma_m^2))$

При $b_i = 2\sigma_i^2$, $a_i = \frac{k_i}{2}$ произведение гамма-распределений представляет собой произведение одномерных распределений Уишарта.

4.2.4. Моделирование случайных чисел с многомерным распределением, равным произведению одномерных распределений хи-квадрат $H_m(k_1, \dots, k_m)$

При $b_i = 2$, $a_i = \frac{k_i}{2}$ произведение гамма-распределений представляет собой произведение распределений хи-квадрат.

4.2.5. Моделирование случайных чисел с многомерным распределением, равным произведению одномерных экспоненциальных распределений
 $E_m(\lambda_1, \dots, \lambda_m)$

При $b_i = \lambda_i$, $a_i = 1$ произведение гамма-распределений представляет собой произведение экспоненциальных распределений.

4.2.6. Моделирование случайных чисел с многомерным равномерным распределением в гиперпрямоугольнике $(a_1, b_1) \times (a_2, b_2) \times \dots \times (a_m, b_m)$
 $U_m((a_1, b_1), (a_2, b_2), \dots, (a_m, b_m))$

$$f_{\bar{\xi}}(\bar{x}) = \begin{cases} \prod_{i=1}^m \frac{1}{(b_i - a_i)}, & a_i < x_i < b_i, \quad a_i < b_i, \\ 0, & \text{иначе.} \end{cases}$$

4.3. Средства Matlab для моделирования многомерных случайных чисел

В Matlab имеется программа для моделирования многомерных случайных чисел с нормальным распределением.

r=mvnrnd(mu,sigma,cases) возвращает матрицу случайных чисел, выбранных из многомерного нормального распределения с вектором средних **mu** и ковариационной матрицей **sigma**. Параметр **cases** является количеством строк в **r** (количеством многомерных случайных чисел). Описание этой функций можно найти в справочной системе в разделе "**Инструментарий статистики**" (каталог `\MATLAB\toolbox\stats\`)

Для моделирования многомерных случайных чисел с распределениями, описанными в пп. 4.2.2 – 4.2.6, необходимо пользоваться программами моделирования скалярных случайных чисел, приведенными в работе № 3.

4.4. Порядок выполнения работы

4.4.1. Выполнить моделирование случайных чисел с указанными в пп. 4.2.1– 4.2.6 распределениями. Для каждого распределения при $m = 2$ вывести диаграмму рассеивания, на которую нанести 100 случайных чисел, используя собственную программу, реализующую предложенный алгоритм, и стандартную программу Matlab. Собственные программы оформить в виде m-файлов-функций. Диаграмма рассеивания – это рисунок, на который нанесены смоделированные значения двумерного случайного вектора.

4.4.2. На диаграмму рассеивания двумерного нормального закона вывести также функцию регрессии

$$y = a_y + r_{x,y} \frac{\sigma_y}{\sigma_x} (x - a_x).$$

Здесь a_x, σ_x – математическое ожидание и среднее квадратичное отклонение аргумента; a_y, σ_y – математическое ожидание и среднее квадратичное отклонение функции; $r_{x,y}$ – коэффициент корреляции между аргументом и функцией.

4.4.3. Исследовать изменение диаграмм рассеивания в зависимости от параметров распределений.

ЛАБОРАТОРНАЯ РАБОТА № 5. ОЦЕНИВАНИЕ ЗАКОНОВ РАСПРЕДЕЛЕНИЯ СКАЛЯРНЫХ СЛУЧАЙНЫХ ВЕЛИЧИН

5.1. Цель работы

5.1.1. Изучение оценок законов распределения скалярных случайных величин.

5.1.2. Приобретение навыков получения оценок законов распределения скалярных случайных величин с помощью системы программирования Matlab.

5.2. Теоретические положения

5.2.1. Эмпирическая функция распределения

Пусть имеется выборка x_1, \dots, x_n из распределения $F_\xi(x)$. Простейший взгляд на нее состоит в том, что числа x_1, \dots, x_n считаются возможными значениями некоторой дискретной случайной величины ξ^* , причем вероятности этих значений одинаковы и равны $1/n$. Ряд распределения этой случайной величины имеет вид табл. 5.1.

Таблица 5.1

Ряд распределения случайной величины ξ^*

x_i	x_1	x_2	...	x_n
p_i	$1/n$	$1/n$...	$1/n$

Эмпирической или выборочной функцией распределения $F_\xi^*(x)$ называется функция распределения дискретной случайной величины ξ^* :

$$F_\xi^*(x) = F_{\xi^*}(x).$$

В соответствии с этим определением эмпирическая функция распределения задается формулой

$$F_{\xi}^*(x) = \frac{m}{n},$$

где m – количество выборочных значений, меньших x ; n – объем выборки.

Эмпирическую функцию распределения удобно строить с использованием порядковых статистик. В этом случае она определяется формулой

$$F_{\xi}^*(x) = \begin{cases} 0, & \text{если } x \leq x_{(1)}, \\ \frac{i}{n}, & \text{если } x_{(i)} < x \leq x_{(i+1)}, \quad i = \overline{1, n-1}, \\ 1, & \text{если } x > x_{(n)}. \end{cases}$$

В этой формуле $x_{(i)}$ – i -я порядковая статистика, $i = \overline{1, n}$.

Эмпирическая функция распределения $F_{\xi}^*(x)$ представляет собой ступенчатую функцию, поскольку это функция распределения дискретной случайной величины (рис. 5.1).

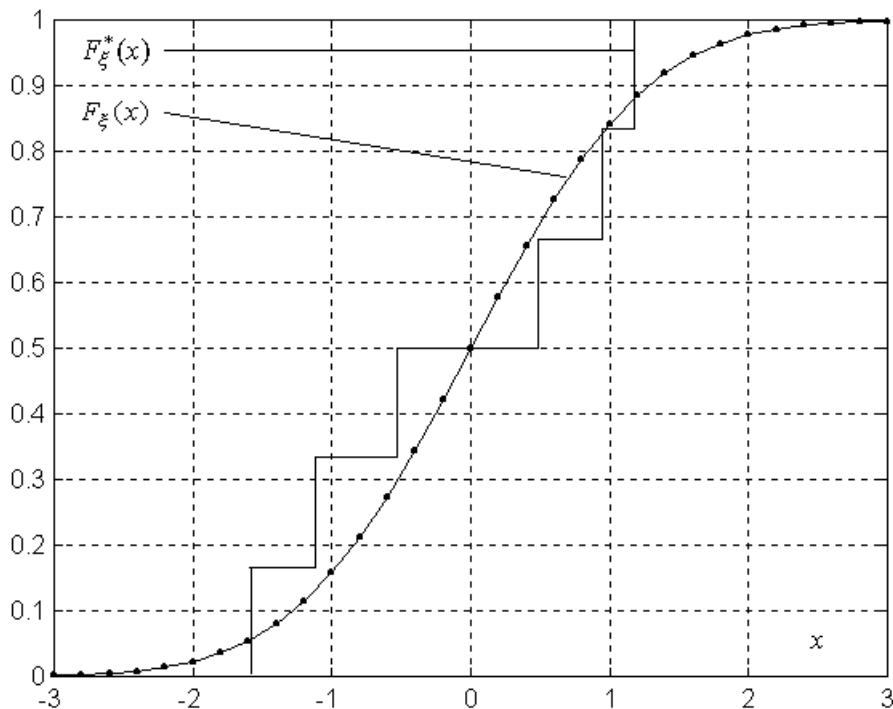


Рис. 5.1. Эмпирическая функция распределения

Эмпирическая функция распределения является состоятельной оценкой генеральной (теоретической) функции распределения $F_{\xi}(x)$. Более того, согласно теореме Гливленко–Кантелли имеет место следующая сходимость почти наверное:

$$\sup_x |F_{\xi}^*(x) - F_{\xi}(x)| \xrightarrow[n \rightarrow \infty]{n.н.} 0.$$

5.2.2. Гистограмма

Гистограмма – это фигура $f_{\xi}^*(x)$, контур которой является оценкой генеральной плотности вероятности $f_{\xi}(x)$. Гистограмма строится следующим образом. Весь интервал выборочных значений $[x_{(1)}, x_{(n)}]$ делится на некоторое количество l непересекающихся интервалов длиной Δ_i , $i = \overline{1, l}$, и подсчитывается количество выборочных значений m_i , попавших в i -й интервал. Если на каждом интервале как на основании построить прямоугольник высотой

$$h_i = \frac{m_i}{n\Delta_i},$$

то мы получим фигуру, которая называется гистограммой. Вид гистограммы приведен на рис. 5.2. Гистограмма является состоятельной оценкой генеральной плотности вероятности при увеличении объема выборки n и числа интервалов l , если только при этом стремится к нулю максимальная из длин интервалов разбиения.

Существуют два способа построения гистограммы.

1. **Равноинтервальный способ.** Выбирают количество интервалов l , а длину Δ каждого интервала определяют по формуле

$$\Delta = \frac{x_{(n)} - x_{(1)}}{l}. \quad (5.1)$$

2. Равновероятный способ. Выбирают количество выборочных значений m , попавших в каждый интервал. Объем выборки должен быть кратен m . Тогда число интервалов $l = n/m$, и интервалы будут следующими: $[x_{(1)}, x_{(m)}]$, $[x_{(m)}, x_{(2m)}]$, ..., $[x_{((l-1)m)}, x_{(lm)}]$, где $x_{(i)}$ – i -я порядковая статистика. При этом способе интервалы имеют различную длину, и границы интервалов попадают на выборочные значения. Принято считать, что граничное значение делится поровну между двумя интервалами, то есть $1/2$ значения попадает в левый интервал и $1/2$ – в правый. Понятно, что при этом в крайний левый интервал попадает $m - 1/2$ значений, в крайний правый – $m + 1/2$ значений, а в средние интервалы – по m значений.

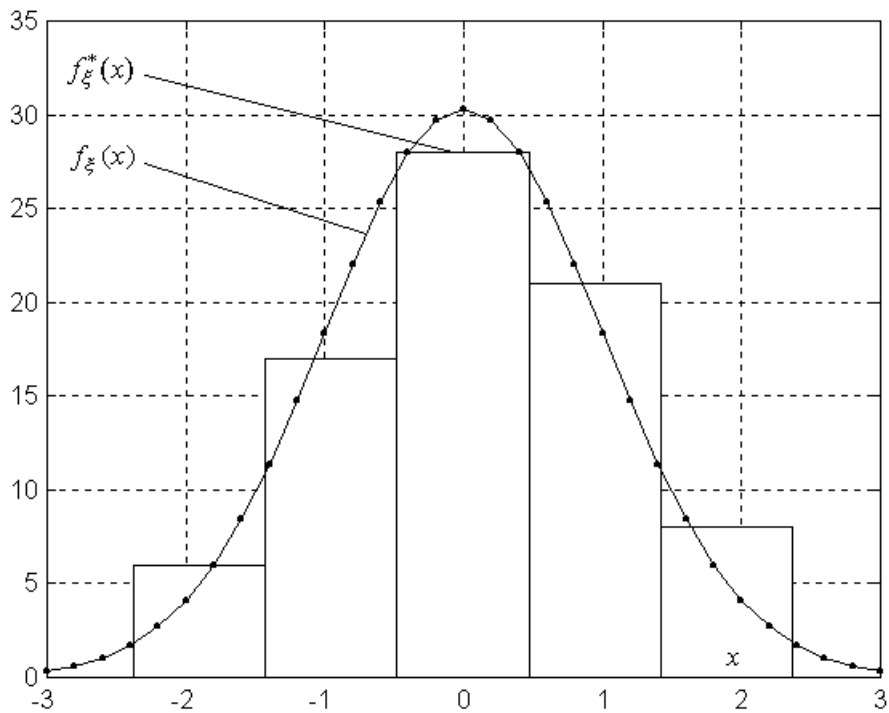


Рис. 5.2. Гистограмма

5.3. Средства Matlab для получения и исследования оценок законов распределения скалярных случайных величин

Для моделирования выборок из различных распределений используются программы, описанные в лабораторной работе № 3. Для моделирования различных плотностей вероятности и функций распределения используются программы, описанные в лабораторной работе № 1. Опишем также программы для получения и отображения на экран оценок законов распределения.

5.3.1. Сортировка в Matlab

$y = \text{sort}(x)$ сортирует элементы вектора x в возрастающем порядке. Здесь x – исходный вектор, y – отсортированный вектор. В случае, когда x – матрица, функция $y = \text{sort}(x)$ сортирует каждый столбец x в возрастающем порядке.

$y = \text{sort}(x, d)$ сортирует матрицу x вдоль измерения d .

$[y, i] = \text{sort}(x, d)$ возвращает также индексную матрицу i . Если x – вектор, то элементы индексной матрицы указывают номера элементов вектора y в исходном векторе x (см. `toolbox\matlab\datafun`).

Программа сортировки используется для формирования вариационного ряда из имеющейся выборки.

5.3.2. Лестничные графики в Matlab

$\text{stairs}(y)$ строит лестничный (ступенчатый) график по значениям элементов вектора y .

$\text{stairs}(x, y)$ строит лестничный график по значениям элементов вектора y в точках скачков, определенных в x . Значения x должны располагаться в возрастающем порядке (см. `toolbox\matlab\specgraph`).

Функция **stairs(x,y)** используется для получения и графического отображения эмпирической функции распределения. В этом случае x – вариационный ряд, а $y(i) = i/n$, $i = \overline{1, n}$.

5.3.3. Гистограммы в Matlab

n=hist(y) распределяет элементы вектора y в 10 интервалов одинаковой длины Δ (5.1) и возвращает количество элементов, попавших в каждый интервал, в виде вектора n . Если y – матрица, то **hist** работает со столбцами.

n=hist(y,l), где l – скаляр, использует l интервалов одинаковой длины (5.1).

n=hist(y,x), где x – вектор, возвращает количество элементов вектора y , попавших в интервалы с центрами, заданными вектором x . Число интервалов в этом случае равно числу элементов вектора x .

[n,x]=hist(...) возвращает числа попаданий в интервалы (в векторе n), а также положения центров интервалов (в векторе x).

hist(...) строит гистограмму без возвращения параметров, то есть строит прямоугольники высотой

$$h_i = m_i,$$

где m_i – число элементов, попавших в i -й интервал, $i = \overline{1, l}$ (см. **toolbox\matlab\datafun**).

Функция **hist** используется для получения и отображения гистограммы.

5.4. Порядок выполнения работы

5.4.1. Получить (смоделировать) выборки из приведенных в п. 1.2.8 лабораторной работы № 1 одномерных распределений. Для этого использовать программы, описанные в п. 3.3.2.7 лабораторной работы № 3.

5.4.2. Для каждого распределения вывести на экран в одно графическое окно гистограмму и генеральную плотность вероятности, а в другое графическое ок-

но – эмпирическую функцию распределения и генеральную функцию распределения. Для вывода генеральных плотностей вероятности и функций распределения использовать программы, описанные в п. 1.2.9 работы № 1.

Для согласования масштабов гистограммы и генеральной плотности вероятности необходимо генеральную плотность вероятности умножить на коэффициент

$$k = n\Delta = n \frac{x_{(n)} - x_{(1)}}{l}.$$

5.4.3. Исследовать сходимость эмпирических распределений к генеральным при увеличении объема выборки n .

ЛАБОРАТОРНАЯ РАБОТА № 6. ПОЛУЧЕНИЕ ТОЧЕЧНЫХ ОЦЕНОК ПАРАМЕТРОВ РАСПРЕДЕЛЕНИЙ

6.1. Цель работы

6.1.1. Изучение методов получения точечных оценок параметров распределений.

6.1.2. Приобретение навыков получения точечных оценок параметров распределений в системе Matlab.

6.2. Теоретические положения

6.2.1. Методы нахождения точечных оценок параметров распределений

Задача точечного оценивания формулируется следующим образом.

Известна плотность вероятности генеральной совокупности с точностью до векторного параметра $\bar{\theta} = (\theta_1, \dots, \theta_m)$, что мы будем обозначать как $f_{\xi}(x, \bar{\theta})$. Требуется по выборке (x_1, \dots, x_n) из этого распределения найти оценку $\hat{\bar{\theta}} = (\hat{\theta}_1, \dots, \hat{\theta}_m)$ параметра $\bar{\theta}$.

Изложим два метода решения этой задачи.

6.2.1.1. Метод моментов

Этот метод заключается в следующем. Находим m начальных теоретических моментов

$$\nu_j = E(\xi^j) = \int_{-\infty}^{\infty} x^j f_{\xi}(x, \bar{\theta}) dx, \quad j = \overline{1, m}.$$

Из формулы видно, что теоретические моменты являются функциями неизвестных параметров, то есть $\nu_j = \nu_j(\theta_1, \dots, \theta_m)$. Далее находим m выборочных начальных моментов

$$\bar{v}_j = \frac{1}{n} \sum_{i=1}^n x_i^j, \quad j = \overline{1, m}.$$

Приравнявая соответствующие теоретические и выборочные моменты, получаем систему m уравнений

$$v_j(\theta_1, \dots, \theta_m) = \bar{v}_j, \quad j = \overline{1, m}.$$

Оценки $\hat{\theta}_1, \dots, \hat{\theta}_m$ определяются как решение этой системы.

Достоинство метода – простота. Недостаток – неизвестно, являются ли полученные оценки достаточно хорошими. Этот вопрос придется решать отдельно. Метод рекомендуется для получения оценок не более двух-трех параметров.

6.2.1.2. Метод максимума правдоподобия

Этот метод использует понятие функции правдоподобия. Функцией правдоподобия называется совместная плотность вероятности $f(x_1, \dots, x_n)$ выборочных значений x_1, \dots, x_n , рассматриваемых как случайные величины. Функция правдоподобия зависит как от переменных x_1, \dots, x_n , так и от неизвестных параметров $\theta_1, \dots, \theta_m$. Обычно она обозначается зависящей только от неизвестных параметров в виде $L(\theta_1, \dots, \theta_m)$. Для простого случайного выбора функция правдоподобия рассчитывается по формуле

$$L(\theta_1, \dots, \theta_m) = \prod_{i=1}^n f_{\xi}(x_i, \theta_1, \dots, \theta_m),$$

где $f_{\xi}(x_i, \theta_1, \dots, \theta_m)$ – плотность вероятности генеральной совокупности, в которую вместо аргумента x подставлено x_i .

Метод максимума правдоподобия заключается в том, что оценки отыскиваются из условия максимума функции правдоподобия:

$$L(\theta_1, \dots, \theta_m) \rightarrow \max_{\theta_1, \dots, \theta_m}.$$

Полученные таким образом оценки называются максимально правдоподобными, или м.п.-оценками.

Часто решение задачи упрощается с помощью следующего приема. Поскольку любая функция и ее логарифм достигают экстремума на одних и тех же значениях аргументов, то можно максимизировать не функцию правдоподобия, а ее натуральный логарифм, то есть логарифмическую функцию правдоподобия:

$$\ln L(\theta_1, \dots, \theta_m) \rightarrow \max_{\theta_1, \dots, \theta_m} .$$

Чтобы найти м.п.-оценки, необходимо приравнять к нулю частные производные функции правдоподобия или логарифмической функции правдоподобия и решить полученную систему уравнений:

$$\frac{\partial}{\partial \theta_j} L(\theta_1, \dots, \theta_m) = 0, \quad j = \overline{1, m}, \quad (6.1)$$

или

$$\frac{\partial}{\partial \theta_j} \ln L(\theta_1, \dots, \theta_m) = 0, \quad j = \overline{1, m}. \quad (6.2)$$

Если учесть, что логарифмическая функция правдоподобия представляется в виде суммы

$$\ln L(\theta_1, \dots, \theta_m) = \sum_{i=1}^n f_{\xi}(x_i, \theta_1, \dots, \theta_m),$$

то последняя система преобразуется к виду

$$\sum_{i=1}^n \frac{\partial}{\partial \theta_j} \ln f_{\xi}(x_i, \theta_1, \dots, \theta_m) = 0, \quad j = \overline{1, m}. \quad (6.3)$$

Доказано, что м.п.-оценки являются состоятельными, асимптотически несмещенными и асимптотически эффективными.

Пример. Найти оценки параметров a и σ^2 нормальной генеральной совокупности $N(a, \sigma^2)$.

Решение. Нам известна плотность вероятности генеральной совокупности с точностью до двух параметров a, σ^2 :

$$f_{\xi}(x, a, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-a)^2}{2\sigma^2}}.$$

Оценки по методу моментов получаем весьма просто. Теоретические моменты $\nu_1 = E(\xi) = a, \mu_2 = D(\xi) = \sigma^2$. Приравнивая их к соответствующим выборочным моментам, получим

$$\bar{a} = \frac{1}{n} \sum_{i=1}^n x_i = \bar{x},$$

$$\bar{\sigma}^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2 = \bar{s}^2.$$

Воспользуемся теперь методом максимума правдоподобия. Будем максимизировать логарифмическую функцию правдоподобия, для чего найдем

$$\ln f_{\xi}(x_i, a, \sigma^2) = \ln\left(\frac{1}{\sqrt{2\pi}}\right) - \frac{1}{2} \ln \sigma^2 - \frac{(x_i - a)^2}{2\sigma^2}.$$

Найдем частные производные по оцениваемым параметрам

$$\frac{\partial}{\partial a} (\ln f_{\xi}(x_i, a, \sigma^2)) = \frac{x_i - a}{\sigma^2},$$

$$\frac{\partial}{\partial \sigma^2} \ln f_{\xi}(x_i, a, \sigma^2) = -\frac{1}{2\sigma^2} + \frac{(x_i - a)^2}{2\sigma^4}.$$

Для получения оценок необходимо решать систему уравнений (6.3), которая имеет вид

$$\begin{cases} \sum_{i=1}^n \frac{x_i - a}{\sigma^2} = 0, \\ \sum_{i=1}^n \frac{(x_i - a)^2 - \sigma^2}{2\sigma^4} = 0. \end{cases}$$

Из первого уравнения находим

$$\bar{a} = \frac{1}{n} \sum_{i=1}^n x_i = \bar{x}.$$

Подставляя \bar{x} вместо a во второе уравнение, получим

$$\hat{\sigma}^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2 = \bar{s}^2.$$

6.3. Средства Matlab для получения точечных оценок параметров распределений

normfit(x,alpha) возвращает оценки параметров нормального распределения по выборке, размещенной в **x**.

[muhat,sigmahat,muci,sgmaci]=normfit(x,alpha) возвращает оценки параметров нормального распределения и 100(1-alpha)-процентные доверительные интервалы. По умолчанию необязательный параметр alpha=0,05, что соответствует 95-процентным доверительным интервалам.

expfit(x) возвращает максимально правдоподобную оценку параметра экспоненциального распределения по данным в **x**.

[muhat, muci]=expfit(x,альфа) дает максимально правдоподобную оценку и 100(1-alpha)-процентный интервал доверия. По умолчанию alpha=0,05, что соответствует 95-процентному интервалу доверия.

gamfit(x) возвращает максимально правдоподобные оценки параметров гамма-распределения по данным в **x**.

[phat,pci]=gamfit(x,alpha) дает максимально правдоподобные оценки и 100(1-alpha)-процентные интервалы доверия. По умолчанию alpha=0,05, что соответствует 95-процентному интервалу доверия.

unifit(x,alpha) возвращает максимально правдоподобные оценки параметров равномерного распределения по данным в **x**.

[ahat,bhat,aci,bci]=unifit(x,alpha) дает максимально правдоподобные оценки и 100(1-alpha)-процентные интервалы доверия. **alpha** – необязательный пара-

метр. По умолчанию $\alpha=0,05$, что соответствует 95-процентному интервалу доверия.

binofit(x,n) возвращает максимально правдоподобную оценку вероятности успеха для биномиального распределения по данным в векторе **x** (**x** – число успехов в **n** испытаниях).

[phat,pci]=binofit(x,n,alpha) дает максимально правдоподобную оценку и 100(1-alpha)-процентный интервал доверия. **alpha** – необязательный параметр. По умолчанию $\alpha=0,05$, что соответствует 95-процентному интервалу доверия.

phat=mle('dist',data) возвращает максимально правдоподобные оценки параметров распределения, определенного в **dist**, по данным в векторе **data**.

[phat,pci]=mle(dist,data,alpha,p1) возвращает максимально правдоподобную оценку и 100(1-alpha)-процентные интервалы доверия. По умолчанию $\alpha=0,05$, что соответствует 95-процентному интервалу доверия. **p1** – дополнительный параметр для использования с биномиальным распределением для задания количества испытаний.

function [x,options]=fmins(funfcn,x,options,grad,varargin) минимизирует функцию многих переменных.

x=fmins('f',x0) пытается вернуть вектор **x**, который является локальным минимумом функции **f(x)** около стартового вектора **x0**. **'f'** – строка, содержащая имя функции, которая должна минимизироваться. **f(x)** должна быть скалярной функцией векторной переменной.

x=fmins('f',x0,options) использует вектор управляющих параметров. Если **options(1)** является положительным, то отображаются промежуточные шаги решения. По умолчанию **options(1)=0**. **options(2)** является допуском завершения для **x**; по умолчанию опция равна $1.e-4$. **options(3)** – допуск завершения для **f(x)**; по умолчанию – $1.e-4$. **options(14)** является максимальным количеством оценок функции; по умолчанию **options(14) = 200*n**, где **n** – длина **x**. Другие компоненты **options** функцией **fmins** не используются.

$x=fmins('f',x0,options,[],p1,p2,...)$ предусматривает дополнительные аргументы, которые передаются в минимизируемую функцию, $f(x,p1,p2,...)$.

$[x,options]=fmins(...)$ возвращает количество оценок функции в **options(10)**.

fmins использует симплекс-метод (прямой поиск) Нелдера–Мида.

Пример. Пусть сформирована следующая m-файл-функция, содержащая минимизируемую функцию:

```
function y=funobj(x)
y=(x(1)-1)^2+(x(2)-2)^2+(x(3)-1)^2;
return
```

Минимизация этой функции может быть выполнена с помощью следующей программы:

```
x=[0 0 0];
y = fmins('funobj',x)
y =
    1.0000    2.0000    1.0000
```

6.4. Порядок выполнения работы

6.4.1. Для приведенных в п. 1.2.8 лабораторной работы № 1 распределений записать функции правдоподобия и получить м.п.-оценки параметров, оформив функции правдоподобия в виде m-файлов-функций и максимизировав их. Максимизация функции $f(x)$ эквивалентна минимизации функции $(-f(x))$.

6.4.2. Оценки сравнить с оценками, полученными с помощью стандартных функций Matlab.

ЛАБОРАТОРНАЯ РАБОТА № 7. ПОЛУЧЕНИЕ ИНТЕРВАЛЬНЫХ ОЦЕНОК ПАРАМЕТРОВ РАСПРЕДЕЛЕНИЙ

7.1. Цель работы

7.1.1. Изучение задачи получения интервальных оценок параметров распределений.

7.1.2. Приобретение навыков получения интервальных оценок параметров распределений в системе Matlab.

7.2. Теоретические положения

Интервальные оценки параметров распределений

Доверительным интервалом для некоторого параметра θ называется интервал (θ_n, θ_g) , накрывающий параметр θ с доверительной вероятностью γ :

$$P(\theta_n < \theta < \theta_g) = \gamma. \quad (7.1)$$

Задача получения интервальной оценки параметра распределения заключается в определении по выборке (x_1, x_2, \dots, x_n) нижней и верхней границ интервала θ_n, θ_g . Доверительная вероятность γ выбирается близкой к 1 из набора чисел $\{0,9; 0,95; 0,975\}$.

Для придания задаче однозначности уравнение (7.1) представляют в виде двух уравнений

$$\begin{cases} P(\theta > \theta_g) = \alpha_1, \\ P(\theta < \theta_n) = \alpha_2, \end{cases} \quad (7.2)$$

где $\alpha_1 + \alpha_2 = 1 - \gamma$.

Доверительный интервал называется симметричным, если в (7.2) $\alpha_1 = \alpha_2 = \alpha = (1 - \gamma)/2$. Таким образом, симметричный доверительный интервал удовлетворяет следующей системе уравнений:

$$\begin{cases} P(\theta > \theta_{\epsilon}) = \frac{1-\gamma}{2}, \\ P(\theta < \theta_{\eta}) = \frac{1-\gamma}{2}. \end{cases} \quad (7.3)$$

Для построения симметричного доверительного интервала для неизвестного параметра θ обычно используется статистика, представляющая собой точечную оценку $\hat{\theta}$ этого параметра, или некоторая функция точечной оценки $g = g(\hat{\theta})$. Должен быть известен закон распределения этой статистики, то есть плотность вероятности $f_g(x)$. В силу того что параметр θ нам неизвестен, эта плотность вероятности будет зависеть от θ , то есть нам известна плотность вероятности статистики с точностью до параметра $f_g(x, \theta)$. Для построения доверительного интервала для параметра θ строят «доверительный интервал» для статистики g , то есть находят g_{η} , g_{ϵ} из системы уравнений

$$\begin{cases} P(g > g_{\epsilon}) = \frac{1-\gamma}{2}, \\ P(g < g_{\eta}) = \frac{1-\gamma}{2}. \end{cases}$$

Решение этой системы уравнений относительно g_{η} , g_{ϵ} эквивалентно решению системы уравнений (7.3) относительно θ_{η} , θ_{ϵ} .

Приведем доверительные интервалы для некоторых параметров.

7.2.1. Доверительный интервал для математического ожидания a нормальной генеральной совокупности $N(a, \sigma^2)$ при известной дисперсии σ^2

$$\bar{x} - u_{\frac{1-\gamma}{2}} \frac{\sigma}{\sqrt{n}} < a < \bar{x} + u_{\frac{1-\gamma}{2}} \frac{\sigma}{\sqrt{n}},$$

где

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i,$$

$u_{\frac{1-\gamma}{2}}$ – $100\frac{1-\gamma}{2}$ -процентное отклонение нормального распределения $N(0,1)$

(рис. 7.1).

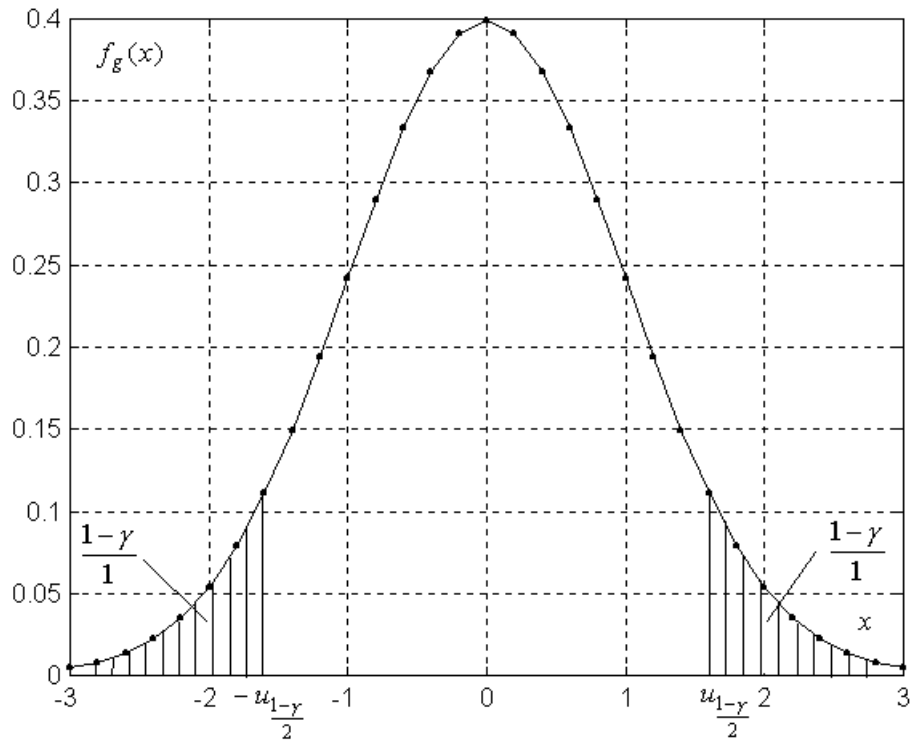


Рис. 7.1. Иллюстрация $100\frac{1-\gamma}{2}$ -процентного отклонения нормального распределения

7.2.2. Доверительный интервал для математического ожидания a нормальной генеральной совокупности $N(a, \sigma^2)$ при неизвестной дисперсии σ^2

$$\bar{x} - t_{\frac{1-\gamma}{2}} \frac{\bar{s}}{\sqrt{n-1}} < a < \bar{x} + t_{\frac{1-\gamma}{2}} \frac{\bar{s}}{\sqrt{n-1}},$$

где

$$\bar{s}^2 = \frac{1}{n} \sum_{n=1}^n (x_i - \bar{x})^2, \quad \bar{s} = \sqrt{\bar{s}^2},$$

$t_{\frac{1-\gamma}{2}}$ – $100\frac{1-\gamma}{2}$ -процентное отклонение распределения $T_1(n-1)$. В силу симметричности распределения Стьюдента $T_1(n-1)$ величина $t_{\frac{1-\gamma}{2}}$ имеет ту же графическую иллюстрацию, что и величина $u_{\frac{1-\gamma}{2}}$ (рис. 7.1).

7.2.3. Доверительный интервал для дисперсии σ^2 нормальной генеральной совокупности $N(a, \sigma^2)$ при известном математическом ожидании a

$$\frac{n\bar{s}_0^2}{v_{\frac{1-\gamma}{2}}} < \sigma^2 < \frac{n\bar{s}_0^2}{v_{\frac{1+\gamma}{2}}},$$

где

$$\bar{s}_0^2 = \frac{1}{n} \sum_{i=1}^n (x_i - a)^2,$$

$v_{\frac{1-\gamma}{2}}, v_{\frac{1+\gamma}{2}}$ – $100\frac{1-\gamma}{2}$ - и $100\frac{1+\gamma}{2}$ -процентные отклонения распределения $H_1(n)$ (рис. 7.2).

7.2.4. Доверительный интервал для дисперсии σ^2 нормальной генеральной совокупности $N(a, \sigma^2)$ при неизвестном математическом ожидании a

$$\frac{n\bar{s}^2}{w_{\frac{1-\gamma}{2}}} < \sigma^2 < \frac{n\bar{s}^2}{w_{\frac{1+\gamma}{2}}},$$

где $w_{\frac{1-\gamma}{2}}, w_{\frac{1+\gamma}{2}}$ – $100\frac{1-\gamma}{2}$ - и $100\frac{1+\gamma}{2}$ -процентные отклонения распределения $H_1(n-1)$ (рис. 7.2).

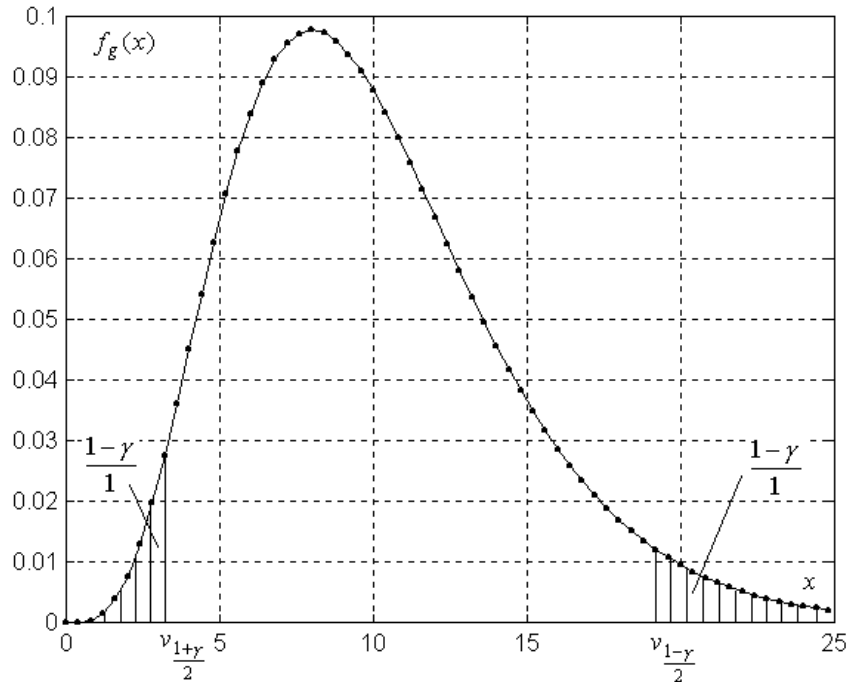


Рис. 7.2. Иллюстрация $100\frac{1-\gamma}{2}$ - и $100\frac{1+\gamma}{2}$ -процентных отклонений распределения хи-квадрат

7.2.5. Доверительный интервал для вероятности появления случайного события A

Пусть выполнено n независимых испытаний Бернулли, в результате которых событие A появилось m раз. Границы доверительного интервала для вероятности $p = P(A)$ события A определяются выражением

$$\frac{n}{n + u_{\frac{1-\gamma}{2}}^2} \left(\hat{p} + \frac{u_{\frac{1-\gamma}{2}}^2}{2n} \pm u_{\frac{1-\gamma}{2}} \sqrt{\frac{\hat{p}\hat{q}}{n} + \frac{u_{\frac{1-\gamma}{2}}^2}{4n^2}} \right),$$

где $u_{\frac{1-\gamma}{2}}$ – $100\frac{1-\gamma}{2}$ -процентное отклонение распределения $N(0,1)$, $p = \hat{p} = \frac{m}{n}$ – частота события, $\hat{q} = 1 - \hat{p}$. При знаке минус эта формула дает нижний доверительный предел, а при знаке плюс – верхний.

Приведем также приближенное выражение для доверительного интервала:

$$\hat{p} - u_{\frac{1-\gamma}{2}} \sqrt{\frac{\hat{p}\hat{q}}{n}} < p < \hat{p} + u_{\frac{1-\gamma}{2}} \sqrt{\frac{\hat{p}\hat{q}}{n}} .$$

7.3. Средства Matlab для получения интервальных оценок параметров распределений

7.3.1. Интервальные оценки параметров распределений

normfit(x,alpha) возвращает оценки параметров нормального распределения по выборке, размещенной в **x**.

[muhat,sigmahat,muci,signiaci]=normfit(x,alpha) возвращает оценки параметров нормального распределения и 100(1-alpha)-процентные доверительные интервалы. По умолчанию необязательный параметр alpha=0,05, что соответствует 95-процентным доверительным интервалам.

expfit(x) возвращает максимально правдоподобную оценку параметра экспоненциального распределения по данным в **x**.

[muhat,muci]=expfit(x,alpha) дает максимально правдоподобную оценку и 100(1-alpha)-процентный интервал доверия. По умолчанию alpha=0,05, что соответствует 95-процентному интервалу доверия.

gamfit(x) возвращает максимально правдоподобные оценки параметров гамма-распределения по данным в **x**.

[phat,pci]=gamfit(x,alpha) дает максимально правдоподобные оценки и 100(1-alpha)-процентные интервалы доверия. По умолчанию alpha=0,05, что соответствует 95-процентному интервалу доверия.

unifit(x,alpha) возвращает максимально правдоподобные оценки параметров равномерного распределения по данным в **x**.

[ahat,bhat,aci,bci]=unifit(x,alpha) дает максимально правдоподобные оценки и 100(1-alpha)-процентные интервалы доверия. **alpha** – необязательный пара-

метр. По умолчанию $\alpha=0,05$, что соответствует 95-процентному интервалу доверия.

binofit(x,n) возвращает максимально правдоподобную оценку вероятности успеха для биномиального распределения по данным в векторе **x**. (**x** – число успехов в **n** испытаниях).

[phat,pci]=binofit(x,n,alpha) дает максимально правдоподобную оценку и 100(1-alpha)-процентный интервал доверия. **alpha** – необязательный параметр. По умолчанию $\alpha=0,05$, что соответствует 95-процентному интервалу доверия.

phat=mle('dist',data) возвращает максимально правдоподобные оценки параметров распределения, определенного в **dist**, по данным в векторе **data**.

[phat,pci]=mle('dist',data,alpha,p1) возвращает максимально правдоподобную оценку и 100(1-alpha)-процентные интервалы доверия. По умолчанию $\alpha=0,5$, что соответствует 95-процентному интервалу доверия. **p1** – дополнительный параметр для использования с биномиальным распределением для задания количества испытаний.

7.3.2. Определение процентных отклонений распределений

x=norminv(p,mu,sigma) возвращает значение аргумента функции нормального распределения с математическим ожиданием **mu** и средним квадратичным отклонением **sigma** по значениям функции в **p**.

x=chi2inv(p,n) возвращает значение аргумента функции распределения хи-квадрат с **n** свободами по значениям функции в **p**.

x=tinv(p,n) возвращает значение аргумента функции распределения Стьюдента с **n** степенями свободы по значениям функции в **p**.

x=finv(p,m,n) возвращает значение аргумента функции распределения Фишера с **m**, **n** степенями свободы по значениям функции в **p**.

Замечание. Для определения 100α -процентного отклонения любого распределения необходимо использовать значение $p = 1 - \alpha$.

7.4. Порядок выполнения работы

7.4.1. Смоделировать выборку из нормального распределения $N(a, \sigma^2)$ с самостоятельно выбранными значениями параметров a, σ^2 и получить приведенные в п. 7.2.1 доверительные интервалы для этих параметров, создав для этого собственные m-файлы-функции.

7.4.2. Результаты сравнить с оценками, полученными стандартными средствами Matlab.

ЛАБОРАТОРНАЯ РАБОТА № 8. ПРОВЕРКА ГИПОТЕЗЫ О ЗАКОНЕ РАСПРЕДЕЛЕНИЯ

8.1. Цель работы

8.1.1 Изучение методов проверки гипотезы о законе распределения.

8.1.2. Получение навыков проверки гипотезы о законе распределения в системе Matlab.

8.2. Теоретические положения

8.2.1. Понятие статистической гипотезы. Классификация гипотез

Статистической гипотезой называется любое непротиворечивое множество утверждений

$$H = \{H_0, H_1, \dots, H_{k-1}\}$$

относительно распределения генеральной совокупности. Такая гипотеза называется k -альтернативной. Каждое утверждение гипотезы H_i , $i = \overline{0, k-1}$, называется альтернативой k -альтернативной гипотезы или также гипотезой.

Проверить гипотезу – это значит по выборке x_1, \dots, x_n из генеральной совокупности принять обоснованное решение об истинности одной из альтернатив.

Если какая-то из альтернатив принята, то все остальные альтернативы отклоняются, то есть считаются ложными.

Гипотеза проверяется на основе так называемого критерия проверки гипотезы. Критерий – это правило, позволяющее принять или отклонить ту или иную альтернативу по имеющейся выборке. Обычно принимают или отклоняют нулевую гипотезу H_0 .

Альтернатива H_i называется параметрической, если она задает значение некоторого параметра θ распределения. В противном случае она называется непараметрической.

Многоальтернативная гипотеза H называется параметрической, если все ее альтернативы параметрические, и непараметрической, если хотя бы одна альтернатива непараметрическая.

Альтернатива H_i называется простой, если она однозначно определяет распределение генеральной совокупности, и сложной – в противном случае.

Многоальтернативная гипотеза H называется простой, если все ее альтернативы простые, и сложной, если хотя бы одна из альтернатив сложная.

8.2.2. Критерий значимости

Пусть проверяется двухальтернативная сложная гипотеза $\{H_0, H_1\}$, где H_0 – простая гипотеза, а H_1 – сложная. Большинство таких гипотез проверяется с помощью так называемого критерия значимости.

В основе критерия значимости лежит некоторая статистика $g = g(x_1, \dots, x_n)$, которая представляет собой отклонение эмпирических (выборочных) данных от гипотетических.

Пусть $f_g(x)$ – плотность вероятности статистики. Эта плотность вероятности должна быть известной. Критерий значимости имеет вид

$$P(|g| > g_{\alpha/2}) = \alpha, \quad (8.1)$$

или

$$P(g > g_{\alpha}) = \alpha, \quad (8.2)$$

или

$$P(g < g_{\alpha}) = \alpha, \quad (8.3)$$

где α – вероятность, которая выбирается из следующего набора малых чисел: $\{0,1; 0,05; 0,025; 0,01\}$. Событие, имеющее такую вероятность, можно считать практически невозможным, то есть не появляющимся в результате одного эксперимента. Величины $g_{\alpha/2}, g_{\alpha}$ называются пределами значимости, α – уров-

нем значимости. Области, определяемые условиями $|g| > g_{\alpha/2}$, или $g > g_{\alpha}$, или $g < g_{\alpha}$, называются критическими областями. Эти области отмечены на рис. 8.1–8.3 штриховкой.

Критерий (8.1) называется двусторонним или критерием с двусторонней критической областью. Критерий (8.2) – правосторонний. Критерий (8.3) – левосторонний. Гипотеза проверяется следующим образом. Выбирается уровень значимости α . По таблицам распределения статистики g определяется предел значимости $g_{\alpha/2}$ или g_{α} , в зависимости от вида критерия. Затем по имеющейся выборке и формуле для статистики g подсчитывают эмпирическое значение статистики $g_{\text{э}}$. Если окажется, что $|g_{\text{э}}| > g_{\alpha/2}$ для двустороннего критерия (8.1), или $g_{\text{э}} > g_{\alpha}$ для правостороннего критерия (8.2), или $g_{\text{э}} < g_{\alpha}$ для левостороннего критерия (8.3), то проверяемая гипотеза H_0 отклоняется. Иначе говоря, если эмпирическое значение статистики $g_{\text{э}}$ попадает в критическую область, то проверяемая гипотеза H_0 отклоняется.

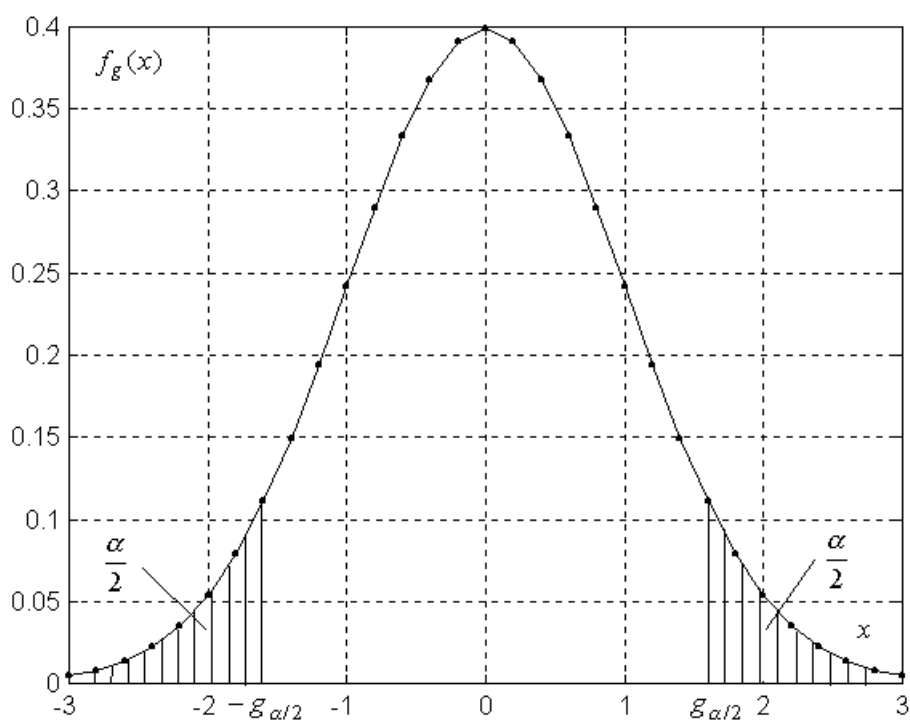


Рис. 8.1. Критические области для двустороннего критерия значимости

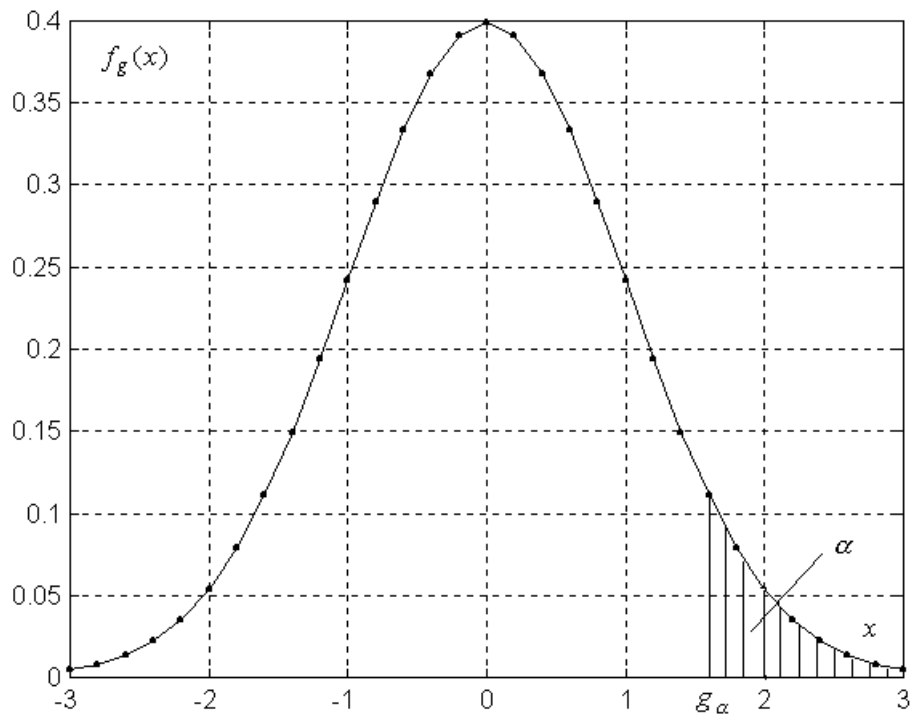


Рис. 8.2. Критическая область для правостороннего критерия значимости

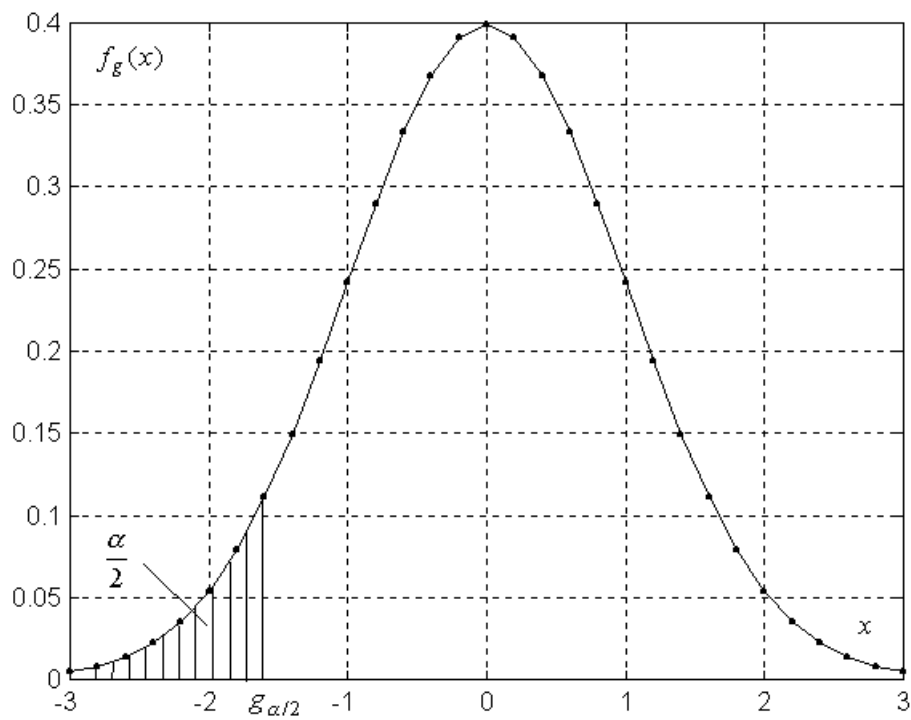


Рис. 8.3. Критическая область для левостороннего критерия значимости

Отклонение гипотезы осуществляется в силу того, что имеется противоречие между гипотетическими и эмпирическими данными: произошло событие, которое не должно было произойти в результате единичного эксперимента.

8.2.3. Проверка гипотезы о законе распределения

Пусть по выборке x_1, \dots, x_n из некоторой генеральной совокупности нужно проверить гипотезу о том, что генеральная совокупность имеет заданное распределение. Критерии для проверки такой гипотезы получили название критериев согласия.

8.2.3.1. Критерий согласия χ^2 (Пирсона)

Пусть $f_\xi(x)$ – плотность вероятности генеральной совокупности, $f_0(x, \theta_1, \dots, \theta_m)$ – гипотетическая плотность вероятности, известная с точностью до m параметров $\theta_1, \dots, \theta_m$, причем m может быть равным нулю. Требуется проверить двухальтернативную непараметрическую сложную гипотезу

$$\{H_0 : f_\xi(x) = f_0(x, \theta_1, \dots, \theta_m), H_1 : f_\xi(x) \neq f_0(x, \theta_1, \dots, \theta_m)\}.$$

Это гипотеза о том, что наша выборка извлечена из распределения $f_0(x, \theta_1, \dots, \theta_m)$. Для проверки этой гипотезы критерием χ^2 множество возможных значений случайной величины ξ разбивается на l интервалов и подсчитывается количество выборочных значений m_i , попавших в каждый интервал (как при построении гистограммы). Для проверки гипотезы используется статистика

$$v = \sum_{i=1}^l \frac{(m_i - n\hat{p}_i)^2}{n\hat{p}_i}, \quad (8.4)$$

где \hat{p}_i – гипотетическая вероятность попадания случайной величины ξ в i -й интервал. Она определяется по формуле

$$\bar{p}_i = \int_{\Delta_i} f_0(x, \hat{\theta}_1, \dots, \hat{\theta}_m) dx.$$

Интегрирование в этой формуле осуществляется по i -му интервалу Δ_i . Здесь $f_0(x, \hat{\theta}_1, \dots, \hat{\theta}_m)$ – гипотетическая плотность вероятности, в которую вместо неизвестных параметров подставлены их м.п.-оценки $\hat{\theta}_1, \dots, \hat{\theta}_m$.

В случае выполнения гипотезы H_0 статистика (8.4) имеет распределение, которое при $n \rightarrow \infty$ приближается к распределению $H_1(l - m - 1)$ (хи-квадрат с $(l - m - 1)$ степенями свободы).

Критерий значимости для проверки этой гипотезы – это правосторонний критерий вида

$$P(v > v_\alpha) = \alpha,$$

где v_α – 100α -процентное отклонение распределения $H_1(l - m - 1)$.

Если гипотетическая плотность вероятности известна полностью, то необходимо считать $m = 0$, то есть воспользоваться таблицами распределения $H_1(l - 1)$.

8.2.3.2. Критерий согласия λ (Колмогорова)

Проверяется гипотеза

$$H_0 : F_\xi(x) = F_0(x)$$

против альтернативы

$$H_1 : F_\xi(x) \neq F_0(x),$$

где $F_\xi(x)$ – функция распределения генеральной совокупности, $F_0(x)$ – гипотетическая функция распределения (полностью известная функция). Она предполагается непрерывной.

Для проверки гипотезы используется статистика

$$\lambda = \Delta \sqrt{n}, \tag{8.5}$$

где

$$\Delta = \max_x |F_0(x) - F_\xi^*(x)| -$$

максимальный модуль отклонения гипотетической функции распределения $F_0(x)$ от эмпирической функции распределения $F_\xi^*(x)$.

Если гипотеза H_0 верна, то статистика λ (8.5) имеет распределение, приближающееся при $n \rightarrow \infty$ к распределению Колмогорова. Критерий для проверки гипотезы имеет следующий вид:

$$P(\lambda > \lambda_\alpha) = \alpha,$$

где $\lambda_\alpha - 100\alpha$ -процентное отклонение распределения Колмогорова (табл.8.1).

Таблица 8.1

Процентные отклонения распределения Колмогорова, $P(\lambda > \lambda_\alpha) = \alpha$

α	0,01	0,02	0,03	0,04	0,05
λ_α	1,627	1,520	1,45	1,40	1,358

8.2.3.3. Критерий согласия ω^2 (Мизеса–Смирнова)

Здесь количественной мерой отклонения эмпирических данных от гипотетических служит величина

$$\omega^2 = \int_{-\infty}^{\infty} [F_\xi^*(x) - F_0(x)]^2 dF(x) = \frac{1}{12n^2} + \frac{1}{n} \sum_{k=1}^n [F_0(x_{(k)}) - \frac{2k-1}{2n}]^2,$$

где $x_{(k)}$ – порядковая статистика. Статистика критерия ω^2 имеет вид

$$z = n\omega^2. \quad (8.6)$$

Для статистики z (8.6) при $n \rightarrow \infty$ существует предельное распределение, для которого составлены таблицы (табл. 8.2). Критерий ω^2 является правосторонним.

Таблица 8.2

Процентные отклонения предельного распределения статистики z ,

$$P(z > z_\alpha) = \alpha$$

α	0,01	0,02	0,03	0,04	0,05
z_α	0,74	0,62	0,55	0,50	0,46

8.3. Средства Matlab для проверки гипотезы о законе распределения

Критерий согласия хи-квадрат

**function [chisq,p,ndf,eval,chisqi,ifail]=g08cggf(ifreq,cint,dist,par,npest,...
prob,ifail)** предназначена для проверки гипотезы о законе распределения с помощью критерия согласия хи-квадрат для стандартных непрерывных распределений. Проверяется нулевая гипотеза о том, что выборочные данные принадлежат определенному распределению, против альтернативной гипотезы, что данные не принадлежат этому распределению. Выборочные данные (x_1, \dots, x_n) должны быть сгруппированы в k классов. Гипотетические вероятности попадания в классы вычисляются в программе или поставляются пользователем. В пределах этой программы доступны следующие распределения: нормальное, равномерное, экспоненциальное, хи-квадрат, гамма.

Пользователь должен поставить массив частот **ifreq** длиной k и массив границ классов (интервалов) **cint** длиной k , где k – число классов. Этот набор данных может быть вычислен с помощью программы **g01aef**. **dist** – строка, содержащая гипотетическое распределение: 'u' – равномерное, 'n' – нормальное, 'e' – экспоненциальное, 'c' – хи-квадрат, 'g' – гамма. **par** – массив, содержащий значения параметров распределения, **npest=0**.

g08cggf возвращает статистику хи-квадрат ν (8.4) в **chisq**, число степеней свободы в **ndf** и вероятность превышения случайной величиной значения статистики **chisq** в **p**.

function [cint,ifreq,xmin,xmax,ifail]=g01aef(x,iclass,cint,ifail) определяет частоты для исходных данных. Данные состоят из выборки объема n , помещенной в векторе x . Параметр **iclass** определяет, как формируются границы классов (интервалов) разбиения выборки. При $iclass=0$ границы классов определяются программно, при $iclass=1$ границы классов берутся из массива **cint**. **cint** – действительный массив длиной k , где k – число классов, содержит границы классов (интервалов) y_1, \dots, y_{k-1} . **ifreq** – массив частот длиной k , в котором содержится число выборочных значений, попавших в каждый интервал (частоты). **xmin** – минимальное выборочное значение, **xmax** – максимальное выборочное значение. По умолчанию устанавливаются входные значения для дополнительных аргументов: $n=\text{длина}(x)$; $k=\text{длина}(\text{cint})$.

Пусть $a = \min(x_1, \dots, x_n)$ и $b = \max(x_1, \dots, x_n)$. Программа создает распределение частот в k классах $f_i, i = \overline{1, k}$. Границы классов $y_i, i = \overline{1, k-1}$, могут быть поставлены пользователем или получены программно. Если значения границ классов получены программно, то они определяются одним из следующих способов. Если $k > 2$, область значений x разделяется на $k-2$ интервала равной длины и два экстремальных крайних интервала. Если $k = 2$, то $y_1 = (a + b)/2$. Частоты классов формируются следующим образом: f_1 равно числу значений x в интервале $(-\infty, y_1)$; f_i равно числу значений x в интервале $[y_{i-1}, y_i)$, $i = \overline{1, k-1}$; f_k равно числу значений x в интервале $[y_{k-1}, \infty)$. Если границы классов рассчитаны программно и $k > 2$, то $f_1 = f_k = 0$ и y_1 и y_{k-1} выбираются так, что $y_1 < a$ и $y_{k-1} > b$.

Если распределение частоты необходимо для дискретной переменной, то желательно, чтобы границы классов поставил пользователь.

Пример использования программы g01aef

$x=[22.3 \ 21.6 \ 22.6 \ 22.4 \ 22.4 \ 22.4 \ 22.1 \ 21.9 \ 23.1 \ 23.4];$

```

iclass = 0; % 0 – границы классов определяются программно, 1 – поставляются
%пользователем
cint =[0 0 0 0 0]; % число нулей=числу классов  $k$ 
[cint,ifreq,xmin,xmax,ifail] = g01aef(x,iclass,cint) % расчет частот
cint =
    21.5991
    22.1997
    22.8003
    23.4009
     0
ifreq =
     0
     3
     5
     2
     0
xmin =
    21.6000
xmax =
    23.4000
ifail =
     0

```

Пример проверки гипотезы с помощью программы **g08cgf**

```

n=100; %задается объем выборки
alpha=0.05 %задается уровень значимости
cdist='e' %задается гипотетическое распределение
par(1)=1; %задается 1-й параметр гипотетического распределения
par(2)=0; %задается 2-й параметр гипотетического распределения
npest=0;
for i=1:n % формируется выборка

```

```

    x(i)=exprnd(1/par(1));
end
iclass=0; % 0 – классы формируются программно, 1 – поставляются %пользо-
вателем
cint=[0 0 0 0 0 0 0 0 0 0]; % количество нулевых элементов равно количеству
% интервалов k
k=10;
[cint,ifreq,xmin,xmax,ifail] = g01aef(x,iclass,cint); %формирование интервалов и
%частот
cint
ifreq
[chisq,p,ndf,eval,chisqi,ifail]=g08cggf(ifreq,cint(1:k-1),cdist,par,npest); %вычисление
%статистики
chisq %значение статистики
ndf %число степеней свободы
p % вероятность того, что  $k\hat{s}_i > \text{chisq}$ 
if p>alpha % выносится решение
    disp('гипотеза принимается')
else
    disp('гипотеза отклоняется')
end

alpha=
0.05
cdist =
e
cint =
0.0094
0.5859

```

1.1624
1.7389
2.3154
2.8919
3.4684
4.0449
4.6214
0

ifreq =

0
47
24
18
4
2
0
2
3
0

** 3 classes have expected frequency less than one.

** ABNORMAL EXIT from NAG Library routine G08CGF: IFAIL = 10

** NAG soft failure – control returned

chisq =

15.8252

ndf =

9

p =

0.0706

гипотеза принимается

8.4. Порядок выполнения работы

8.4.1. Получить выборки объема $n = 100$ из распределений, приведенных в п. 1.2.8 лабораторной работы № 1. Проверить гипотезу о законе распределения приведенными в п. 8.2.2 данной работы критериями, написав собственные m -файлы.

8.4.2. Сравнить результаты проверки гипотезы по критерию хи-квадрат с результатами проверки с помощью стандартной программы Matlab **g08cgf**.

Замечание 1. В программе **g08cgf** плотность вероятности экспоненциального распределения предполагается заданной в виде $f(x) = \lambda e^{-\lambda x}$, в то время как в программах **exprnd**, **expcdf**, **expddf**, **expfit** – в виде $f(x) = \lambda^{-1} e^{-\lambda^{-1}x}$.

Замечание 2. При написании собственной программы проверки гипотезы для определения процентных отклонений распределения хи-квадрат необходимо использовать программу **chi2inv**, описанную в п. 7.3.2 работы № 7. Matlab не имеет программ для определения процентных отклонений распределений Колмогорова и ω^2 . В связи с этим при проверке гипотезы критериями Колмогорова и ω^2 процентное отклонение необходимо взять из табл. 8.1, 8.2 данной работы.

ЛАБОРАТОРНАЯ РАБОТА № 9. ОБРАБОТКА РЕЗУЛЬТАТОВ КОСВЕННЫХ ИЗМЕРЕНИЙ: КЛАССИЧЕСКАЯ ЗАДАЧА О МЕТОДЕ НАИМЕНЬШИХ КВАДРАТОВ

9.1. Цель работы

9.1.1. Изучение задачи и методов обработки результатов косвенных измерений.

9.1.1. Исследование в системе Matlab задачи оценивания местоположения объекта по измерениям пеленгов.

9.2. Теоретические положения

9.2.1. Классическая задача о методе наименьших квадратов (МНК)

Классическая задача оценивания векторных параметров по косвенным измерениям предполагает, что результаты измерений (показания приборов) z_i функционально связаны с параметрами $\theta_1, \dots, \theta_m$ вектором параметров:

$$z_i = \psi_i(\theta_1, \dots, \theta_m, x_{i,1}, \dots, x_{i,l}) + e_i, \quad i = \overline{1, M}, \quad (9.1)$$

где $\psi_i(\theta_1, \dots, \theta_m, x_{i,1}, \dots, x_{i,l})$ – некоторые известные скалярные функции; e_i – ошибки измерений; $x_{i,1}, \dots, x_{i,l}$ – входные переменные, которые измеряются точно или отсутствуют.

Требуется по измерениям z_i найти оценки $\hat{\theta}_1, \dots, \hat{\theta}_m$ неизвестных параметров $\theta_1, \dots, \theta_m$.

Задача в таком виде была сформулирована Гауссом. Для ее решения Гаусс предложил свой знаменитый метод наименьших квадратов (МНК).

В настоящее время эта задача формулируется и решается с использованием векторно-матричного подхода. Зависимости (9.1) записывают в векторной форме:

$$Z = \Psi(\Theta, X) + E, \quad (9.2)$$

где $Z^T = (z_1, \dots, z_M)$, $\Psi^T(\Theta, X) = (\psi_1(\Theta, X_1), \dots, \psi_M(\Theta, X_M))$, $\Theta^T = (\theta_1, \dots, \theta_m)$; $X = (X_1, \dots, X_M)$; $E^T = (e_1, \dots, e_M)$. Вектор ошибок E считается распределенным по нормальному закону $N_M(0, R_E)$.

Требуется по результатам измерений Z , X найти оценку $\hat{\Theta}$ вектора параметров Θ .

После линеаризации функции $\Psi(\Theta, X)$ в окрестности некоторой опорной точки Θ_0 получают МНК-оценку в виде

$$\hat{\Theta}_{NK} = \Theta_0 + (Q^T R_E^{-1} Q)^{-1} Q^T R_E^{-1} (Z - \Psi(\Theta_0, X)), \quad (9.3)$$

где

$$Q = \frac{d\Psi(\Theta_0, X)}{d\Theta_0}, \quad (9.4)$$

а ковариационная матрица оценки определяется выражением

$$R_{\hat{\Theta}_{NK}} = (Q^T R_E^{-1} Q)^{-1}.$$

9.2.2. Оценивание координат объекта по измерениям пеленгов

Полученную оценку (9.3) используем в задаче оценивания декартовых координат объекта на плоскости $хоу$ угломерным способом в многопозиционных локационных системах. Из M базовых точек (позиций) $A_1 = (x_1, y_1), \dots, A_M = (x_M, y_M)$ измеряются углы на объект, в результате чего получают значения углов $\alpha_1, \dots, \alpha_M$ (рис. 9.1). Величины $\alpha_1, \dots, \alpha_M$ независимы и распределены по нормальному закону $N(a_i, \sigma^2)$, где a_i – точное значение i -го угла, σ^2 – дисперсия ошибок измерений углов. Координаты x_i, y_i базовых точек будем

считать известными. По имеющимся измерениям необходимо оценить вектор координат объекта $\Theta^T = (x_c, y_c)$.

Опишем алгоритм расчета МНК-оценки (9.3). Соотношения (9.1) в данном случае имеют вид

$$\alpha_i = \arctg \frac{y_c - y_i}{x_c - x_i} + e_i, \quad i = \overline{1, M}, \quad (9.5)$$

так что

$$\psi_i(x_c, y_c) = \arctg \frac{y_c - y_i}{x_c - x_i}, \quad i = \overline{1, M}. \quad (9.6)$$

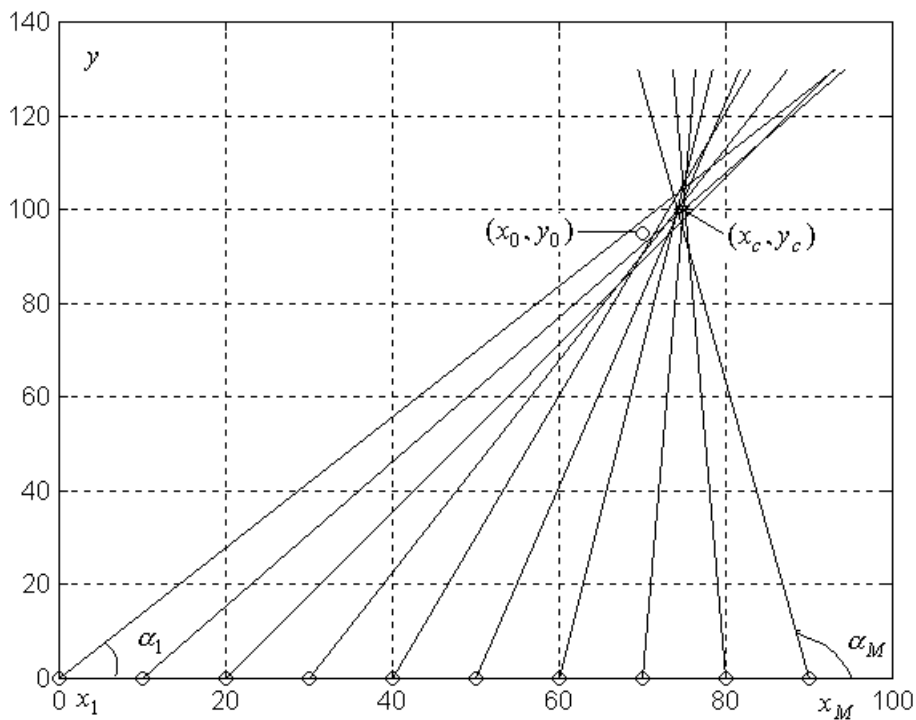


Рис. 9.1. Графическая иллюстрация задачи оценивания местоположения объекта по измерениям пеленгов

В векторной форме записи (9.2) будем иметь следующие обозначения:

$$Z^T = (\alpha_i), \quad \Psi^T(x_c, y_c) = (\psi_i(x_c, y_c)), \quad E^T = (e_i), \quad i = \overline{1, M}.$$

Поскольку ошибки измерений e_i независимы и имеют одну и ту же дисперсию σ^2 , то ковариационная матрица вектора E равна

$$R_E = \sigma^2 I,$$

где I – единичная $(M \times M)$ -матрица.

В этом случае оценка (9.3) приводится к виду

$$\hat{\Theta}_{NK} = \Theta_0 + (Q^T Q)^{-1} Q^T (Z - \Psi(\Theta_0)), \quad (9.7)$$

где $\Theta_0^T = (x_0, y_0)$ – опорная точка.

Матрица Q в соответствии с формулами (9.4), (9.6) формируется в виде

$$Q = (q_{i,j}), \quad i = \overline{1, M}, \quad j = \overline{1, 2},$$

где

$$q_{i,1} = \frac{d\psi_i(x_0, y_0)}{dx_0} = \frac{y_0 - y_i}{d_i},$$

$$q_{i,2} = \frac{d\psi_i(x_0, y_0)}{dy_0} = \frac{x_0 - x_i}{d_i},$$

$$d_i = (x_0 - x_i)^2 + (y_0 - y_i)^2.$$

Таким образом, в выражении (9.7) $\hat{\Theta}_{NK}$, Θ_0 , Z , $\Psi(\Theta_0)$ – векторы-столбцы, Q – $(M \times 2)$ -матрица и Q^T – $(2 \times M)$ -матрица.

9.3. Средства Matlab для выполнения задания

Matlab обладает весьма удобными средствами для работы с матрицами. Для транспонирования матрицы применяется символ «'» (кавычка), так что A' – это транспонированная матрица A . Используются также следующие символы: «+» (плюс) или «-» (минус) для сложения или вычитания матриц, «*» (звездочка) для умножения матриц, «/» (слэш) или «\» (обратный слэш) для деления матриц. Например, $A \setminus B$ означает левое деление B на A , соответствующее математическому произведению $A^{-1}B$, где A^{-1} – обратная матрица. Запись A / B означает правое деление матрицы A на матрицу B , что соответствует произведению AB^{-1} . Использование деления вместо умножения на обратную матрицу

имеет преимущества по точности и времени выполнения, так как использует метод исключения Гаусса без явного обращения матриц.

9.4. Порядок выполнения работы

9.4.1. Выполнить статистическое компьютерное моделирование задачи при 10 базовых точках ($M = 10$), расположенных на оси абсцисс с равномерным шагом $l = 10$ км, так что

$$x_l = 0, x_{k+l} = x_k + l, k = \overline{1, M-1}, \quad (9.8)$$

$$y_k = 0, k = \overline{1, M}. \quad (9.9)$$

Для объекта с координатами

$$(x_c, y_c) = (75, 100) \quad (9.10)$$

вычислить 10 оценок

$$\hat{\Theta}_{NK, \nu}^T = (\hat{x}_{c, \nu}, \hat{y}_{c, \nu}), \quad \nu = \overline{1, 10}. \quad (9.11)$$

Опорную точку (x_0, y_0) для получения каждой МНК-оценки выбрать равной наблюдению, полученному по первым двум пеленгам:

$$x_0 = \frac{(y_2 - y_1) + (x_1 \operatorname{tg} \alpha_1 - x_2 \operatorname{tg} \alpha_2)}{\operatorname{tg} \alpha_1 - \operatorname{tg} \alpha_2}, \quad (9.12)$$

$$y_0 = (x_0 - x_1) \operatorname{tg} \alpha_1 + y_1. \quad (9.13)$$

Дисперсию ошибок измерений углов принять равной $\sigma^2 = 0,001$ рад.

9.4.2. В одно окно вывести графическую иллюстрацию, включающую: базовые точки (9.8, 9.9), точку-объект (9.10), лучи пеленгов из базовых точек на объект под углами (9.5), точки-оценки координат объекта (9.7, 9.11). Лучи пеленгов можно построить следующим образом. Поскольку уравнение прямой, выходящей из точки $(x_i, 0)$ под углом α_i , имеет вид

$$y = \operatorname{tg} \alpha_i (x - x_i),$$

то, выбрав, например, $y = 1,3y_c = y_l$, получим

$$xl = \frac{yl + tg\alpha_i x_i}{tg\alpha_i}.$$

Лучи проводятся с помощью функции **plot** через две точки $(x_i, 0)$ и (xl, yl) , то есть по векторам $u = (x_i, xl)$ и $v = (0, yl)$.

9.4.3. Исследовать зависимость точности оценивания от дисперсии ошибок измерений углов σ^2 , от выбора опорной точки (x_0, y_0) , предусмотрев ее произвольный выбор, от расположения на плоскости оцениваемого объекта по отношению к базовым точкам.

ЛАБОРАТОРНАЯ РАБОТА № 10. РЕГРЕССИОННЫЙ АНАЛИЗ

10.1. Цель работы

10.1.1. Изучение методов решения задач регрессионного анализа.

10.1.2. Приобретение навыков решения задач регрессионного анализа с помощью системы Matlab.

10.2. Теоретические положения

10.2.1. Постановка задачи

В регрессионном анализе изучается взаимосвязь между случайными и неслучайными величинами на основе экспериментальных данных. Ограничения, накладываемые на изучаемые величины, менее обременительны по сравнению с корреляционным анализом. В связи с этим считается, что задачи регрессионного анализа чаще встречаются на практике.

Рассмотрим сразу случай многих переменных (множественную нелинейную регрессию) в матричной форме, поскольку он включает и простую линейную и нелинейную регрессии.

Считается, что случайная величина η зависит от векторной неслучайной переменной $X = (x_1, \dots, x_q)$ таким образом, что условная плотность вероятности величины η нормальная следующего вида:

$$f(\eta/x) = N(\varphi(x, \bar{\theta}), \sigma^2),$$

где $\bar{\theta} = (\theta_1, \dots, \theta_m)$ – вектор неизвестных параметров функции регрессии $y = \varphi(x, \bar{\theta})$.

Эквивалентной является следующая модель данных:

$$\eta = \varphi(X, \bar{\theta}) + \xi,$$

где $\xi \in N(0, \sigma^2)$.

Выбирая некоторые значения X_1, \dots, X_n векторной переменной X , можно получить значения $y_{o,1}, \dots, y_{o,n}$ случайной величины η в виде

$$y_{oi} = \varphi(x_i, \bar{\theta}) + z_i, \quad i = \overline{1, n},$$

где z_i – независимые случайные величины (значения величины ξ).

Требуется по наблюдениям $(X_1, y_{o,1}), \dots, (X_n, y_{o,n})$ получить оценки $\hat{\bar{\theta}}, \hat{\sigma}^2$ неизвестных параметров $\bar{\theta}, \sigma^2$ и сделать статистические выводы относительно этих параметров.

10.2.2. Точечные оценки параметров

Решим задачу в матричных обозначениях. Функция регрессии $y = \varphi(x, \bar{\theta})$, как линейная, так и нелинейная относительно X , может быть представлена следующим образом:

$$y = \varphi(X, \bar{\theta}) = H^T \bar{\theta},$$

где

$$H^T = H^T(X) = (h_1(X), \dots, h_m(X)) = (h_j(X)), \quad j = \overline{1, m}.$$

Здесь $h_j(X)$ – некоторые функции вектора X ,

$$\bar{\theta}^T = (\theta_1, \dots, \theta_m) = (\theta_j), \quad j = \overline{1, m},$$

где $\bar{\theta}^T$ – вектор-столбец параметров.

Класс функций, описываемых таким образом, называется классом функций, линейных по параметрам. Желая получить линейную функцию регрессии одной переменной, мы можем выбрать

$$H^T = (1, x), \quad \bar{\theta}^T = (\alpha, \beta).$$

Тогда

$$y = \varphi(x, \alpha, \beta) = (1, x) \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \alpha + \beta x.$$

Желая получить нелинейную функцию регрессии в виде $y = \alpha + \beta x_1 + \gamma x_2 + \tau x_1^2$, достаточно выбрать $H^T = (1, x_1, x_2, x_1^2)$, $\bar{\theta}^T = (\alpha, \beta, \gamma, \tau)$.

Таким образом, мы рассматриваем зависимости вида

$$y_{oi} = H_i^T \bar{\theta} + z_i, \quad i = \overline{1, n},$$

где

$$H_i^T = (h_1(X_i), \dots, h_m(X_i)) = (h_j(X_i)), \quad j = \overline{1, m}, \quad i = \overline{1, n},$$

и считаем, что векторы H_1, \dots, H_n линейно независимы.

Воспользовавшись методом максимума правдоподобия, можно получить следующие оценки:

$$\hat{\theta}^T = \left(\sum_{j=1}^n H_j H_j^T \right)^{-1} \left(\sum_{i=1}^n H_i y_{o,i} \right),$$

$$\hat{\sigma}^2 = \frac{1}{n} \sum_{i=1}^n (y_{o,i} - H_i^T \hat{\theta})^2.$$

Обычно эти оценки записывают в другой форме. Если рассмотреть весь набор данных эксперимента, то рассматриваемую модель можно записать в виде

$$Y_o = F \bar{\theta} + Z,$$

где

$$Y_o^T = (y_{o,i}), \quad F = (f_{i,j}) = (h_j(X_i)), \quad \bar{\theta}^T = (\theta_j), \quad Z^T = (z_i), \quad i = \overline{1, n}, \quad j = \overline{1, m}.$$

В этом случае оценки запишутся следующим образом:

$$\hat{\theta} = (F^T F)^{-1} (F^T Y_o), \tag{10.1}$$

$$\hat{\sigma}^2 = \frac{1}{n} (Y_o - F \hat{\theta})^T (Y_o - F \hat{\theta}). \tag{10.2}$$

10.2.3. Свойства оценок

Оценка $\hat{\theta}$ – несмещенная, состоятельная, эффективная. Оценка $\hat{\sigma}^2$ – асимптотически несмещенная, состоятельная, асимптотически эффективная. Исправленная оценка

$$\hat{\sigma}_1^2 = \frac{n}{n-m} \hat{\sigma}^2 = \frac{1}{n-m} (Y_o - F\hat{\theta})^T (Y_o - F\hat{\theta}),$$

где m – количество оцениваемых параметров,

является несмещенной, состоятельной и эффективной. Оценки $\hat{\theta}$ и $\hat{\sigma}^2$ независимы.

Рассмотрим распределения оценок. Доказано, что

$$\hat{\theta} \in N_m(\bar{\theta}, \sigma^2 (F^T F)^{-1}), \quad (10.3)$$

$$v = \frac{n\hat{\sigma}^2}{\sigma^2} = \frac{(n-m)\hat{\sigma}_1^2}{\sigma^2} \in H_{n-m}.$$

Матрица

$$\sigma^2 (F^T F)^{-1} \quad (10.4)$$

является ковариационной матрицей оценки.

Введем обозначения:

$$A = F^T F = (a_{i,j}), \quad A^{-1} = (F^T F)^{-1} = (a^{i,j}), \quad i, j = \overline{1, m}.$$

Используя эти обозначения и распределение вектора $\hat{\theta}$, можно записать, что

$$u_i = \frac{\hat{\theta}_i - \theta_i}{\sqrt{\sigma^2 a^{i,i}}} \in N(0,1), \quad i = \overline{1, m},$$

$$t_i = \frac{\hat{\theta}_i - \theta_i}{\sqrt{n\sigma^2 a^{i,i}}} \sqrt{n-m} = \frac{\hat{\theta}_i - \theta_i}{\sqrt{\sigma_1^2 a^{i,i}}} \in T_{n-m}, \quad i = \overline{1, m}.$$

Имея оценку $\hat{\theta}$, мы можем получить оценку ординаты функции регрессии \hat{y} для любой точки X в виде

$$\hat{y} = H^T \hat{\theta}.$$

Можно показать, что

$$\hat{y} = H^T \hat{\theta} \in N(H^T \bar{\theta}, \sigma^2 H^T A^{-1} H)$$

и не зависит от $\hat{\sigma}^2$. Тогда статистика

$$t_y = \frac{(\hat{y} - y)\sqrt{n-m}}{\sqrt{H^T A^{-1} H n \hat{\sigma}^2}} = \frac{(\hat{y} - y)}{\sqrt{H^T A^{-1} H \hat{\sigma}_1^2}} \in T_{n-m}.$$

10.2.4. Доверительные интервалы для параметров и функции регрессии

На основе статистики t_i получаем доверительный интервал для параметра θ_i , $i = \overline{1, m}$:

$$\hat{\theta}_i - t_{\frac{1-\gamma}{2}} \sqrt{\hat{\sigma}_1^2 a^{i,i}} < \theta_i < \hat{\theta}_i + t_{\frac{1-\gamma}{2}} \sqrt{\hat{\sigma}_1^2 a^{i,i}}.$$

На основе статистики t_y получаем доверительный интервал для истинного значения функции регрессии $y = H^T \bar{\theta}$:

$$\hat{y} - t_{\frac{1-\gamma}{2}} \sqrt{\hat{\sigma}_1^2 H^T A^{-1} H} < y < \hat{y} + t_{\frac{1-\gamma}{2}} \sqrt{\hat{\sigma}_1^2 H^T A^{-1} H}.$$

В этих выражениях $t_{\frac{1-\gamma}{2}}$ – 100 $\frac{1-\gamma}{2}$ -процентное отклонение распределения T_{n-m}

(Стьюдента с $n-m$ степенями свободы).

На основе статистики v получаем доверительный интервал для дисперсии σ^2 :

$$\frac{(n-m)\hat{\sigma}_1^2}{v_{\frac{1-\gamma}{2}}} < \sigma^2 < \frac{(n-m)\hat{\sigma}_1^2}{v_{\frac{1+\gamma}{2}}},$$

где $v_{\frac{1-\gamma}{2}}$, $v_{\frac{1+\gamma}{2}}$ – 100 $\frac{1-\gamma}{2}$ - и 100 $\frac{1+\gamma}{2}$ -процентные отклонения распределения

H_{n-m} (хи-квадрат с $n-m$ степенями свободы).

Статистики t_i , v можно использовать для проверки гипотез о параметрах

θ_i, σ^2 , например, гипотезы $\{H_0 : \theta_i = \theta_{i,0}, H_1 : \theta_i \neq \theta_{i,0}\}$, $i = \overline{1, m}$, или гипотезы $\{H_0 : \sigma^2 = \sigma_0^2, H_1 : \sigma^2 \neq \sigma_0^2\}$.

Статистика t_y может применяться для проверки гипотезы относительно истинного значения ординаты $y = H^T \bar{\theta}$ функции регрессии.

10.3. Средства Matlab для выполнения работы

При создании собственных m-файлов используются операции над матрицами, рассмотренные в работе № 9. Для проверки гипотез о значимости параметров необходимо вычислять процентные отклонения распределения Стьюдента. Это можно сделать с помощью программы **x=tinv(p,n)**, которая возвращает значение аргумента функции распределения Стьюдента с **n** степенями свободы по значению **p** функции распределения. Для определения 100α -процентного отклонения необходимо выполнить обращение: **x=tinv(1-alpha,n)**.

10.4. Порядок выполнения работы

10.4.1. Выполнить моделирование задачи для функции регрессии вида

$$\varphi(x, \alpha, \beta, \gamma) = \alpha + \beta x + \gamma x^2$$

при выбранных самостоятельно значениях параметров α, β, γ , дисперсии ошибок измерений σ^2 и объема выборки n . Значения $x_i, i = \overline{1, n}$, переменной x смоделировать с равномерным шагом из некоторого интервала (a, b) .

10.4.2. Получить точечные оценки параметров (10.1, 10.2). Вывести графическую иллюстрацию в виде теоретической функции регрессии, эмпирической функции регрессии и наблюдений (поля рассеивания).

10.4.3. Проверить гипотезы о значимости параметров α, β, γ функции регрессии.

10.4.4. Построить и вывести в виде графика 95-процентный доверительный интервал для выходной переменной y функции регрессии.

ЛИТЕРАТУРА

1. Крамер Г. Математические методы статистики. – М.: Мир, 1975. – 648 с.
2. Харин Ю.С., Степанова М.Д. Практикум на ЭВМ по математической статистике. – Мн.: Университетское, 1987. – 304 с.
3. Муха В.С. Теория вероятностей и математическая статистика. Ч. 3: Математическая статистика: Тексты лекций для студентов радиотехн. специальностей. – Мн.: БГУИР, 1998. – 71 с.
4. Смирнов Н.В., Дунин-Барковский И.В. Курс теории вероятностей и математической статистики для технических приложений. – М.: Наука, 1969. – 511 с.
5. Боровков А.А. Математическая статистика. – М.: Наука, 1984. – 472 с.
6. Большев Л.Н., Смирнов Н.В. Таблицы математической статистики. – М.: Наука, 1983. – 416 с.
7. Жевняк Р.М. и др. Теория вероятностей и математическая статистика. – Мн.: Харвест, 2000. – 384 с.
8. Муха В.С. Теория вероятностей: Учеб. пособие для студентов техн. специальностей высш. учеб. заведений. – Мн.: БГУИР, 2001. – 167 с.
9. Гаусс Ф.К. Избранные геодезические сочинения. Т. 1: Способ наименьших квадратов. – М.: Изд-во геод. литературы, 1957. – 157 с.
10. Дьяконов В.П., Абраменкова И.В. Матлаб 5.0/5.3. Система символьной математики. – М.: Нолидж, 1999. – 640 с.
11. Муха В.С., Птичкин В.А. Введение в MATLAB: Метод. пособие для выполнения лаб. работ по курсам «Статистические методы обработки данных» и «Теория автоматического управления» для студ. спец. 53 01 02 «Автоматизированные системы обработки информации». – Мн.: БГУИР, 2002. – 40 с.

СОДЕРЖАНИЕ

ЛАБОРАТОРНАЯ РАБОТА № 1. ОЗНАКОМЛЕНИЕ С СИСТЕМОЙ МАТЛАВ. ОДНОМЕРНЫЕ РАСПРЕДЕЛЕНИЯ ТЕОРИИ ВЕРОЯТНОСТЕЙ И МАТЕМАТИЧЕСКОЙ СТАТИСТИКИ.....	1
1.1. Цель работы	3
1.2. Теоретические положения	3
1.2.1. Общие сведения о системе Matlab	3
1.2.2. Запуск системы Matlab	4
1.2.3. Сеанс работы с Matlab	4
1.2.4. Элементы программирования на языке Matlab.....	5
1.2.5. Справочная система Matlab.....	8
1.2.6. Создание m-файлов-сценариев	9
1.2.7. Построение графиков функций одной переменной.....	9
1.2.8. Плотности вероятностей некоторых одномерных распределений	11
1.2.9. Средства Matlab для изучения одномерных распределений.....	13
1.3. Порядок выполнения работы.....	15
ЛАБОРАТОРНАЯ РАБОТА № 2. МНОГОМЕРНЫЕ РАСПРЕДЕЛЕНИЯ ТЕОРИИ ВЕРОЯТНОСТЕЙ И МАТЕМАТИЧЕСКОЙ СТАТИСТИКИ ...	16
2.1. Цель работы	16
2.2. Теоретические положения. Плотности вероятностей некоторых многомерных распределений	16
2.2.1. Многомерное нормальное (гауссовское) распределение $N(A, R)$	16
2.2.2. Двухмерное нормальное распределение	17
2.2.3. Произведение одномерных гамма-распределений $\Gamma_m((a_1, b_1), \dots, (a_m, b_m))$	19
2.2.4. Произведение одномерных распределений Уишарта $W_m((k_1, \sigma_1^2), \dots, (k_m, \sigma_m^2))$	19
2.2.5. Произведение одномерных распределений хи-квадрат $H_m(k_1, \dots, k_m)$..	19
2.2.6. Произведение одномерных экспоненциальных распределений $E_m(\lambda_1, \dots, \lambda_m)$	19
2.2.7. Равномерное распределение в гиперпрямоугольнике $(a_1, b_1) \times (a_2, b_2) \times \dots \times (a_m, b_m)$ $U_m((a_1, b_1), (a_2, b_2), \dots, (a_m, b_m))$	20
2.3. Средства Matlab для изучения многомерных распределений.....	20
2.3.1. Создание массивов трехмерной графики	20
2.3.2. Построение контурных графиков	21

2.3.3. Построение графиков трехмерных поверхностей	22
2.3.4. Продолжение построений графиков.....	24
2.4. Порядок выполнения работы.....	25

ЛАБОРАТОРНАЯ РАБОТА № 3. МОДЕЛИРОВАНИЕ ОДНОМЕРНЫХ СЛУЧАЙНЫХ ЧИСЕЛ..... 26

3.1. Цель работы	26
3.2. Теоретические положения	26
3.2.1. Равномерное распределение $U(a, b)$	26
3.2.2. Нормальное (гауссовское) распределение $N(a, \sigma^2)$	26
3.2.3. Экспоненциальное распределение $E(\lambda)$	27
3.2.4. Распределение χ^2 с k степенями свободы $H_1(k)$	27
3.2.5. Распределение Стьюдента с k степенями свободы $T_1(k)$	27
3.2.6. Распределение Фишера с m, k степенями свободы $F_1(m, k)$	28
3.2.7. Одномерное распределение Уишарта $W_1(k, \sigma^2)$	28
3.3. Средства Matlab для моделирования одномерных случайных чисел... 28	28
3.3.1. Создание m-файлов-функций	28
3.3.2. Управляющие структуры языка программирования системы Matlab	30
3.3.3. Стандартные функции Matlab для моделирования одномерных случайных чисел.....	37
3.4. Порядок выполнения работы..... 37	37

ЛАБОРАТОРНАЯ РАБОТА № 4. МОДЕЛИРОВАНИЕ МНОГОМЕРНЫХ СЛУЧАЙНЫХ ЧИСЕЛ..... 39

4.1. Цель работы	39
4.2. Теоретические положения	39
4.2.1. Моделирование случайных чисел с многомерным нормальным (гауссовским) распределением	39
4.2.2. Моделирование случайных чисел с многомерным распределением, равным произведению одномерных гамма-распределений $\Gamma_m((a_1, b_1), \dots, (a_m, b_m))$	41
4.2.3. Моделирование случайных чисел с многомерным распределением, равным произведению одномерных распределений Уишарта $W_m((k_1, \sigma_1^2), \dots, (k_m, \sigma_m^2))$	41

4.2.4. Моделирование случайных чисел с многомерным распределением, равным произведению одномерных распределений хи-квадрат $H_m(k_1, \dots, k_m)$	41
4.2.5. Моделирование случайных чисел с многомерным распределением, равным произведению одномерных экспоненциальных распределений $E_m(\lambda_1, \dots, \lambda_m)$	42
4.2.6. Моделирование случайных чисел с многомерным равномерным распределением в гиперпрямоугольнике $(a_1, b_1) \times (a_2, b_2) \times \dots \times (a_m, b_m)$ $U_m((a_1, b_1), (a_2, b_2), \dots, (a_m, b_m))$	42
4.3. Средства Matlab для моделирования многомерных случайных чисел.....	42
4.4. Порядок выполнения работы.....	43
ЛАБОРАТОРНАЯ РАБОТА № 5. ОЦЕНИВАНИЕ ЗАКОНОВ РАСПРЕДЕЛЕНИЯ СКАЛЯРНЫХ СЛУЧАЙНЫХ ВЕЛИЧИН	44
5.1. Цель работы	44
5.2 Теоретические положения	44
5.2.1. Эмпирическая функция распределения	44
5.2.2. Гистограмма.....	46
5.3. Средства Matlab для получения и исследования оценок законов распределения скалярных случайных величин.....	48
5.3.1. Сортировка в Matlab	48
5.3.2. Лестничные графики в Matlab	48
5.3.3. Гистограммы в Matlab	49
5.4. Порядок выполнения работы.....	49
ЛАБОРАТОРНАЯ РАБОТА № 6. ПОЛУЧЕНИЕ ТОЧЕЧНЫХ ОЦЕНОК ПАРАМЕТРОВ РАСПРЕДЕЛЕНИЙ	51
6.1. Цель работы	51
6.2. Теоретические положения	51
6.2.1. Методы нахождения точечных оценок параметров распределений.....	51
6.3. Средства Matlab для получения точечных оценок параметров распределений.....	55
6.4. Порядок выполнения работы.....	57

ЛАБОРАТОРНАЯ РАБОТА № 7. ПОЛУЧЕНИЕ ИНТЕРВАЛЬНЫХ ОЦЕНОК ПАРАМЕТРОВ РАСПРЕДЕЛЕНИЙ.....	58
7.1. Цель работы	58
7.2. Теоретические положения	58
7.2.1. Доверительный интервал для математического ожидания a нормальной генеральной совокупности $N(a, \sigma^2)$ при известной дисперсии σ^2	59
7.2.2. Доверительный интервал для математического ожидания a нормальной генеральной совокупности $N(a, \sigma^2)$ при неизвестной дисперсии σ^2	60
7.2.3. Доверительный интервал для дисперсии σ^2 нормальной генеральной совокупности $N(a, \sigma^2)$ при известном математическом ожидании a	61
7.2.4. Доверительный интервал для дисперсии σ^2 нормальной генеральной совокупности $N(a, \sigma^2)$ при неизвестном математическом ожидании a	61
7.2.5. Доверительный интервал для вероятности появления случайного события A	62
7.3. Средства Matlab для получения интервальных оценок параметров распределений.....	63
7.3.1. Интервальные оценки параметров распределений.....	63
7.3.2. Определение процентных отклонений распределений	64
7.4. Порядок выполнения работы.....	65
ЛАБОРАТОРНАЯ РАБОТА № 8. ПРОВЕРКА ГИПОТЕЗЫ О ЗАКОНЕ РАСПРЕДЕЛЕНИЯ	66
8.1. Цель работы	66
8.2. Теоретические положения	66
8.2.1. Понятие статистической гипотезы. Классификация гипотез.....	66
8.2.2. Критерий значимости.....	67
8.2.3. Проверка гипотезы о законе распределения.....	70
8.3. Средства Matlab для проверки гипотезы о законе распределения.....	73
8.4. Порядок выполнения работы.....	78
ЛАБОРАТОРНАЯ РАБОТА № 9. ОБРАБОТКА РЕЗУЛЬТАТОВ КОСВЕННЫХ ИЗМЕРЕНИЙ: КЛАССИЧЕСКАЯ ЗАДАЧА О МЕТОДЕ НАИМЕНЬШИХ КВАДРАТОВ.....	79
9.1. Цель работы	79

9.2. Теоретические положения	79
9.2.1. Классическая задача о методе наименьших квадратов (МНК).....	79
9.2.2. Оценивание координат объекта по измерениям пеленгов	80
9.3. Средства Matlab для выполнения задания	82
9.4. Порядок выполнения работы.....	83
ЛАБОРАТОРНАЯ РАБОТА № 10. РЕГРЕССИОННЫЙ АНАЛИЗ	85
10.1. Цель работы	85
10.2. Теоретические положения	85
10.2.1. Постановка задачи	85
10.2.2. Точечные оценки параметров	86
10.2.3. Свойства оценок	88
10.2.4. Доверительные интервалы для параметров и функции регрессии	89
10.3. Средства Matlab для выполнения работы	90
10.4. Порядок выполнения работы.....	90
ЛИТЕРАТУРА.....	92

Учебное издание

**Муха Владимир Степанович,
Слуянова Татьяна Викторовна**

**СТАТИСТИЧЕСКИЕ МЕТОДЫ
ОБРАБОТКИ ДАННЫХ**

Лабораторный практикум
для студентов специальности 53 01 02
«Автоматизированные системы обработки информации»
всех форм обучения

Редактор Т.А. Лейко
Корректор Е.Н. Батурчик

Подписано в печать .01.2004.
Печать ризографическая.
Уч.-изд. л. 4,6.

Формат 60×84 1/16.
Гарнитура «Таймс».
Тираж 150 экз.

Бумага офсетная.
Усл. печ. л.
Заказ 544.

Издатель и полиграфическое исполнение:
Учреждение образования
«Белорусский государственный университет информатики и радиоэлектроники».
Лицензия ЛП № 156 от 30.12.2002.
Лицензия ЛВ № 509 от 03.08.2001.
220013, Минск, П. Бровка, 6