

**Министерство науки и высшего образования Российской Федерации**  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
**«Владимирский государственный университет**  
**имени Александра Григорьевича и Николая Григорьевича Столетовых»**  
**(ВлГУ)**

Институт машиностроения и автомобильного транспорта  
Кафедра автомобильного транспорта, безопасности и управления качеством

Баженов М. Ю.

Методические указания к практическим работам по дисциплине  
«Аналитические и численные методы в планировании эксперимента»  
для студентов ВлГУ,  
обучающихся по направлению 23.04.03 Эксплуатация транспортно-технологических  
машин и комплексов

Владимир – 2022 г.

# Лабораторная работа №1.

## Основы работы в MATLAB. Решение уравнений.

Цель работы: приобретение навыков работы в пакете прикладных программ MATLAB: получение навыков в работе с переменными и символьными выражения, изучение способов решения уравнений, вычисление элементарных математических и тригонометрических функций.

### 1. Что такое MATLAB

MATLAB – матричная лаборатория – наиболее развитая система программирования для научно-технических расчетов, дополненная к настоящему времени несколькими десятками более частных приложений, относящихся к вычислительной математике, обработке информации, конструированию электронных приборов, экономике и ряду других разделов прикладной науки.

MATLAB предназначен прежде всего для программирования численных алгоритмов. Он разрабатывается уже более 15 лет и возник на основе более ранних прикладных пакетов LINPACK и EIGPACK, созданных в 1970-е гг. в США, и в свою очередь повлиял на появление таких систем, как MathCad, MAPLE и Mathematica. Совершенствование системы MATLAB происходило как в связи с достижениями в вычислительной математике, так и в связи с изменениями в архитектуре персональных компьютеров и развитием общесистемных средств. Со временем MATLAB был дополнен целым рядом приложений (toolboxes), далеко раздвинувших границы его применимости.

MATLAB – система программирования высокого уровня, работающая как интерпретатор и включающая большой набор инструкций (команд) для выполнения самых разнообразных вычислений, задания структур данных и графического представления информации. Команды эти разбиты на тематические группы, расположенные в различных директориях системы. Теперь в системе около 800 команд, и примерно половина из них вполне доступна начинающему пользователю. Команды с большим возможным объемом вычислений написаны на C, но много и таких команд, которые представлены в терминах этих первых. Поэтому система оказывается почти открытой для пользователя. Имеются большие возможности для вывода двумерной и трехмерной графики и средства управления ею. Пользователь может без особых затруднений добавлять свои

команды и писать программы в терминах уже существующих команд; несколько сложнее делать это в рамках Фортрана и С. Можно обмениваться данными с программами на этих языках, а из них обращаться к системе. Краткость и наглядность программирования и исключительные возможности визуализации результатов делают систему очень эффективной при поисках и апробации новых алгоритмов, при проведении разовых расчетов и в учебном процессе, поскольку ее можно осваивать без предварительного знакомства с основами программирования и выполнять такие сложные примеры, которые невозможно делать с использованием других систем.

## 2. Окна программы MATLAB

По умолчанию Рабочий стол включает в себя четыре окна.

Самое большое и наиболее важное окно - Command Window (Командное окно).

В верхней части окна Command Window (Командное окно) можно увидеть общую информацию о MATLAB, в окне видны некоторые инструкции по началу работы или по доступу к справке, но, что важнее всего, вы увидите приглашение командной строки (>>). Если окно Command Window (Командное окно) активно, то строка заголовка окна будет темной, а после приглашения командной строки (далее просто приглашение) будет отображен текстовый курсор (мигающая вертикальная линия). Это та область, где вы будете вводить команды программы MATLAB. Если же окно Command Window (Командное окно) не активно, щелкните на нем мышью в любом месте. На рисунке (Рис. 1) отображен пример открытого окна Рабочий стол вновь запущенной программы MATLAB.

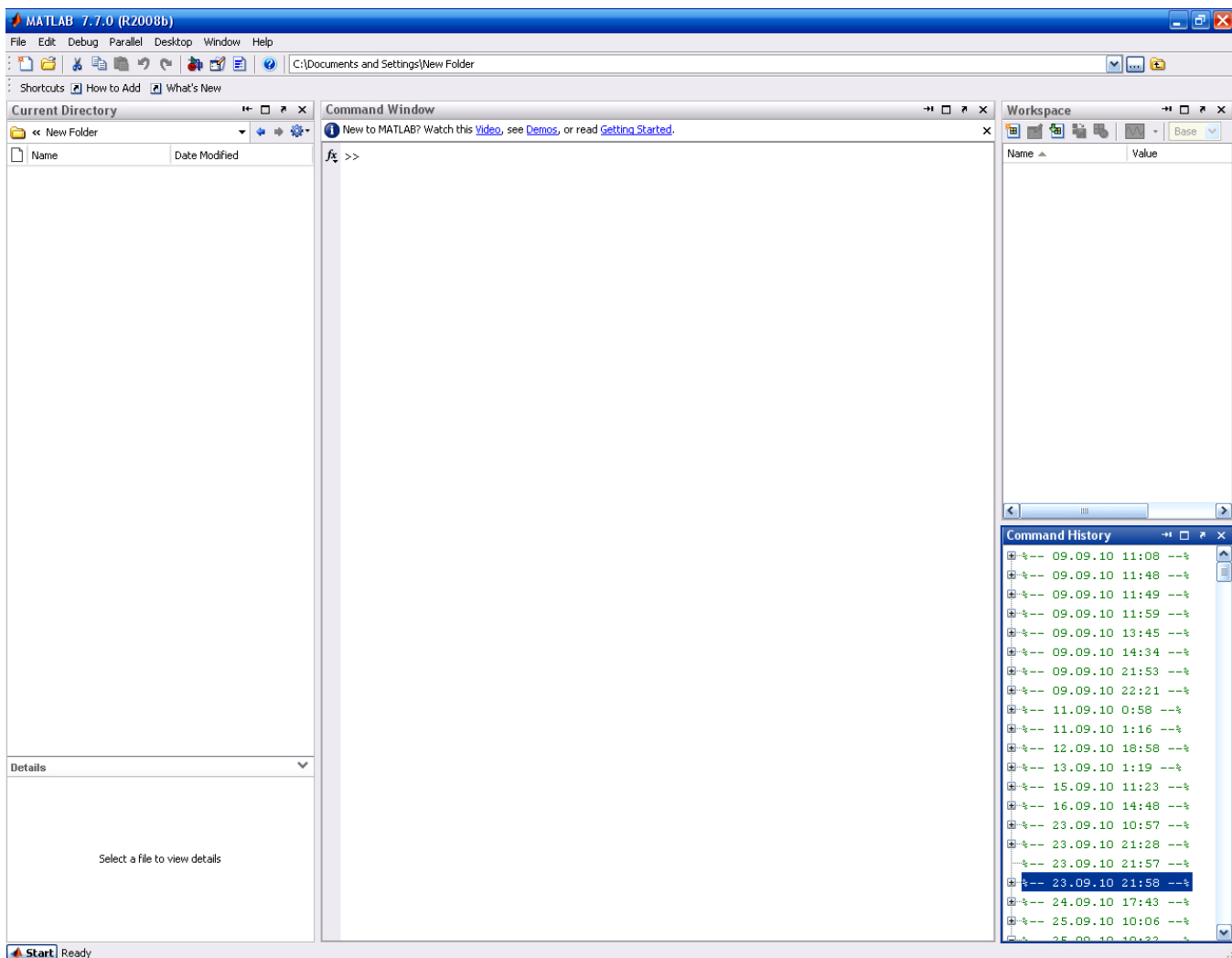


Рисунок 1 – Рабочий стол Matlab.

Следующее по важности окно называется Workspace (Рабочая область).

Окно Workspace (Рабочая область) содержит список текущих переменных и их размеры (но не значения переменных). Если вы дважды щелкнете мышью на переменной, значение переменной будет отображено в новом окне, называемом Array Editor (Редактор массива), которое вы можете использовать для редактирования отдельных элементов в векторах и матрицах. (Это окно можно также открыть, введя команду `openvar` и имя интересующей вас переменной.) Вы можете удалить переменную из «рабочей области», выделив ее в окне Workspace (Рабочая область) и нажав клавишу Delete.

Если вам необходимо прервать сессию и вы не хотите впоследствии вычислять все повторно, то вы можете сохранить текущую «рабочую область» с помощью команды `save`. Например, после ввода команды `save myfile` будут сохранены значения всех заданных текущих переменных в файле с именем `myfile.mat`. Чтобы сохранить только значения переменных X и Y, введите следующее:

```
>> save myfile X Y
```

Когда вы начинаете новую сессию и желаете восстановить значения этих переменных, используйте команду `load`. Например, введение команды `load myfile` восстановит значения всех переменных, сохраненных в файле `myfile.mat`.

Окно `Current Directory` (Текущий каталог).

Новые файлы, которые вы создаете в программе MATLAB, будут храниться в вашем текущем каталоге. Имя этого каталога отображается на панели инструментов Рабочего стола, а файлы и подкаталоги, которые содержит текущий каталог, отображаются в окне `Current Directory` (Текущий каталог). Отобразить имя текущего каталога вы можете также с помощью команды `pwd` («print working directory» (Отобразить рабочий каталог)) в окне `Command Window` (Командное окно), и можете также получить список содержимого текущего каталога, введя команду `dir` или `ls`.

Окно `Command History` (История команд).

Окно `Command History` (История команд) содержит текущую историю команд, которые вы вводите в окне `Command Window` (Командное окно). Это полезно в двух ситуациях. Во-первых, это позволяет вам с одного взгляда увидеть запись команд, которые вы ввели ранее. Во-вторых, это может сберечь вам время, затрачиваемое на ввод. Если вы щелкнете мышью на одной из записей (команд) в окне `Command History` (История команд), эта команда будет немедленно выполнена в окне `Command Window` (Командное окно). Однако зачастую вам понадобится редактировать предыдущую команду перед ее выполнением. Если вы щелкнете правой кнопкой мыши на записи команды в окне `Command History` (История команд), то команда будет выделена и на экране отобразится меню параметров. Вы можете выбрать команду `Copy` (Копировать), затем щелкнуть правой кнопкой мыши в окне `Command Window` (Командное окно) и выбрать команду `Paste` (Вставить), после чего выбранная вами команда будет отображена в командной строке и будет готова для редактирования.

### 3. Арифметика и переменные

Программу MATLAB можно использовать для арифметических расчетов, как калькулятор. Вы можете прибавлять с помощью знака `+`, отнимать со знаком `-`, умножать со знаком `*`, делить со знаком `/` и возводить в степень со знаком `^`. Например:

```
>> 7^2+(11-2)/2-6/2-20
```

```
ans =
```

30.5000

Программа MATLAB выводит ответ и обозначает значение в виде переменной, называемой *ans*. Если вы захотите выполнить дальнейшие вычисления с полученным ответом, вы можете использовать переменную *ans*, вместо того, чтобы заново вводить числовое значение. Например, вы можете вычислить сумму квадрата полученного ответа и его квадратного корня, как показано ниже:

```
>> ans^2 + sqrt(ans)
```

```
ans =
```

```
935.7727
```

Заметьте, что программа MATLAB присваивает новое значение переменной с каждым вычислением. Чтобы выполнять более сложные вычисления, вы можете присваивать вычисленные значения переменным по вашему выбору, например:

```
>> a = sin (10)
```

```
a =
```

```
-0.5440
```

```
>> b = cos (10)
```

```
b =
```

```
-0.8391
```

Теперь, где бы программа ни обнаружила буквы *a* и *b*, MATLAB будет подставлять значения -0.5440 и -0.8391 соответственно:

```
>> a^2 + b^2
```

```
ans =
```

```
1
```

Имя переменной или имя функции могут представлять собой любую строку или буквы, цифры, символы подчеркивания, при условии, что они начинаются с буквы (знаки пунктуации исключены). Программа MATLAB различает буквы

верхнего и нижнего регистров. Вам следует выбирать запоминающиеся имена, используя в основном буквы нижнего регистра.

Программа MATLAB использует арифметику двойной точности с плавающей точкой, при чем точность сохраняется приблизительно до 15 знака; однако по умолчанию MATLAB отображает только 5 знаков. Чтобы отобразить больше знаков, введите команду *format long*. Тогда все последующие числовые результаты будут иметь 15 знаков после точки. Чтобы вернуться к пятизначному отображению, введите команду *format short*.

#### 4. Символьные вычисления

Используя инструмент Symbolic Math Toolbox (Инструментарий символьной математики), можно осуществлять алгебраические или символьные вычисления, такие как разложение на множители или решение алгебраических уравнений.

Для выполнения символьных вычислений вам необходимо использовать команду *syms*, чтобы задать переменные, которые вы планируете использовать в качестве символьных. Рассмотрим следующую серию команд:

```
>> syms x y
```

```
>> (x-y)*(x-y)^2
```

```
ans =
```

```
(x - y)^3
```

```
>> expand (ans)
```

```
ans =
```

```
x^3 - 3*x^2*y + 3*x*y^2 - y^3
```

```
>> factor (ans)
```

```
ans =
```

```
(x - y)^3
```

Команда *expand* заставляет программу MATLAB представить выражение в виде многочлена, а команда *factor* - восстановить краткую прежнюю форму.

Программа MATLAB имеет команду, называемую *simplify*, которую вы иногда можете использовать для максимально простого выражения формулы, например:

```
>> simplify ((x^3-y^3)/(x-y))
```

```
ans =
```

```
x^2 + x*y + y^2
```

Когда идет работа с символьными выражениями, часто бывает необходимо заменять числовое значение или даже другое символьное выражение одной (или более) исходной переменной в выражении. Это делается с помощью команды *subs*. Например, предположим, что задано символьное выражение *w*, включающее в себя символьную переменную *u*. Тогда команда *subs (w, u, 2)* заменит значением 2 переменную *u* в выражении *w*.

Так же в символьной форме можно складывать обыкновенные дроби:

```
>> sym ('1/3')+ sym ('1/2')
```

```
ans =
```

```
5/6
```

Наконец, для того чтобы выполнять арифметические вычисления переменной точности существует команда *vpa*. Например, чтобы получить результат вычисления  $\sqrt{2}$  с точностью до 50 знака после запятой, нужно ввести следующее:

```
>> vpa ('sqrt (2)', 50)
```

```
ans =
```

```
1.4142135623730950488016887242096980785696718753769
```

## 5. Встроенные функции

Программа MATLAB имеет много встроенных функций. В их число входят функции *sqrt*, *cos*, *sin*, *tan*, *log*, *exp* и *atan* (для функции арктангенс), а также более специализированные математические функции, такие как *gamma*, *erf* и *besselj*. Программа MATLAB имеет также некоторые встроенные константы, включая *pi* (число  $\pi$ ), *i* (комплексное число  $i = \sqrt{-1}$ ) и *Inf* ( $\infty$  - бесконечность). Ниже показано несколько примеров:

```
>> log (exp(4))
```



```
ans =
```

```
4
```

Функция *log* является натуральным логарифмом и во многих текстах называется *ln*.

```
>> sin (3*pi/4)
```

```
ans =
```

```
0.7071
```

Чтобы получить точный ответ, вам необходимо использовать символьный аргумент.

```
>> sin (sym ('3*pi/4'))
```

```
ans =
```

```
2^(1/2)/2
```

## 6. Решение уравнений

Можно решать уравнения, содержащие переменные, с помощью команды *solve*. Например, чтобы решить квадратное уравнение  $x^2 - 3x - 9 = 0$ , введите следующее:

```
>> solve ('x^2-3*x-9 = 0')
```

```
ans =
```

```
3/2 - (3*5^(1/2))/2  
(3*5^(1/2))/2 + 3/2
```

Обратите внимание, что уравнение, которое требуется решить, задано как строка, то есть взято в одинарные кавычки. Ответ представляет собой точное (символьное) решение. Для получения числовых решений введите *double (ans)* или *vpa (ans)*, чтобы отобразить больше знаков.

Ввод с командой *solve* может также быть символьным выражением, но в этом случае программа MATLAB потребует, чтобы правая часть выражения была заключена в скобки, и решения уравнения  $x^2 - 4x = -10$  будет выглядеть так:

```
>> syms x
>> solve (x^2 - 4*x + 10)
```

ans =

```
2 - 6^(1/2)*i
6^(1/2)*i + 2
```

Ответ представляет собой точное (символьное) решение (сложные числа, где буква *i* в ответе ставится для мнимой единицы  $\sqrt{-1}$ ). Для получения числовых решений введите *double (ans)* или *vpa (ans)*, чтобы отобразить больше знаков.

С помощью команды *solve* можно решать высокоуровневые полиномиальные (многочленные) уравнения, равно как и многие другие типы уравнений. Можно также решать уравнения, содержащие более чем одну переменную. Если уравнений меньше, чем переменных, вам следует определить (как строки), какую переменную (переменные) требуется вычислить. Например, введите *solve ('2\*x - log (y) = 1', 'y')*, чтобы решить уравнение  $2x - \log y = 1$  для переменной *y* при условии *x*. Подобным образом вы можете определить более чем одно уравнение. Например:

```
>> [x, y] = solve ('x^2 - y = 2', 'y - 2*x = 5')
```

x =

```
2*2^(1/2) + 1
1 - 2*2^(1/2)
```

y =

```
4*2^(1/2) + 7
7 - 4*2^(1/2)
```

Эта система уравнений имеет два решения. Программа MATLAB выдает решение, выводя два значения *x* и два значения *y* для этих решений. Таким образом, первое решение состоит из первого значения *x* и первого значения *y*. Вы можете извлечь эти значения, введя в командную строку *x(1)* и *y(1)*:

```
>> x (1)
```

```
ans =
```

```
2*2^(1/2) + 1
```

```
>> y (1)
```

```
ans =
```

```
4*2^(1/2) + 7
```

Второе решение можно извлечь, введя  $x(2)$  и  $y(2)$ .

Обратите внимание, что в предыдущей команде *solve* мы назначили вывод в векторной форме  $[x, y]$ . Если вы используете команду *solve* в системе уравнений, не задавая вывод в векторной форме, в этом случае программа MATLAB не отображает автоматически значения решения:

```
>> solv = solve ('x^2 - y = 2', 'y - 2*x = 5')
```

```
solv =
```

```
  x: [2x1 sym]
```

```
  y: [2x1 sym]
```

Чтобы увидеть векторы значений  $x$  и  $y$ , введите  $\text{solv.x}$  и  $\text{solv.y}$ . Чтобы увидеть отдельные значения, введите  $\text{solv.x}(1)$  и  $\text{solv.y}(1)$ , и т.п.

Некоторые уравнения нельзя решить символически, и в таких случаях команда *solve* пытается найти числовой ответ. Например:

```
>> solve ('sin(x)=2-x')
```

```
ans =
```

```
x = 1.1060601577062719106167372970301
```

Варианты заданий для работы

Вариант №1

1. Вычислите следующие величины:

а)  $367 - 143$ ;

б)  $e^{12}$  и  $51810 \cdot \pi$ , с точностью до 15 знака. Сравните, какое из значений больше;

в)  $1567/478$ ,  $7613/2533$ ,  $417/117$ . Сравните, какая из дробей больше.

2. Вычислите следующие тригонометрические величины:

а)  $\sin(0.3)$ ;

б)  $\ln(5)$ ;

в)  $\arctan(1/4)$ .

3. Решите в символьной форме систему уравнений:

$$\begin{cases} x^2 + y^2 = 2 \\ y + 2 \cdot x = 3. \end{cases}$$

4. Разложите на множители многочлен:

$$x^4 - 1.$$

5. Представьте выражение в виде многочлена:

$$(x - 3 \cdot y) \cdot (x^2 - 4 \cdot y).$$

1. Найдите точное решение уравнения:

а)  $37 \cdot x - 22 = 0$ ;

б)  $3 \cdot x^4 + 6 \cdot x^3 - 9 \cdot x^2 = 0$ .

Вариант №2

2. Вычислите следующие величины:

а)  $717 - 356$ ;

б)  $\ln(12)$  и  $7,8/\pi$ , с точностью до 15 знака. Сравните, какое из значений больше;

в)  $3012/477$ ,  $567/423$ ,  $2319/215$ . Сравните, какая из дробей больше.

3. Вычислите следующие тригонометрические величины:

а)  $\cos(0.7)$ ;

б)  $\cosh(0.2)$ ;

в)  $\arctan(1/6)$ .

4. Решите в символьной форме систему уравнений:

$$\begin{cases} x + y = 3 \\ x^2 + 2y^2 - xy + 2x - 3y = 3. \end{cases}$$

5. Разложите на множители многочлен:

$$a^2 + b^2.$$

6. Представьте выражение в виде многочлена:

$$(x + 1) \cdot (x - 1)(x^2 + 1).$$

7. Найдите точное решение уравнения:

а)  $17 \cdot x = 33$ ;

б)  $x^4 - 13 \cdot x^2 + 30 = 0$ .

## Лабораторная работа №2

### Обработка матриц

Цель работы: приобретение навыков работы в пакете прикладных программ MATLAB: изучение способов формирования векторов, матриц и подматриц.

#### 1. Формы задания переменных

В рабочем окне MATLAB можно задавать переменные, представляющие собой числа, векторы-строки, векторы-столбцы и матрицы размерностью [nхn].

Например, вводим переменную A=5. В рабочем окне выводится:

```
>> A=5
```

```
A =
```

```
5
```

Если на конце выражения поставить знак «;», то выражение сразу будет записано в память без вывода на экран. Это позволяет сокращать время выполнения вычислительных программ и операций.

Числа в обозначении вектора-столбца отделяются друг от друга точкой с запятой, числа в обозначении вектора-строки отделяются друг от друга запятой, запятая в обозначении десятичных дробей обозначается точкой. При вводе векторов и матриц используются квадратные скобки.

Пример ввода в рабочем окне MATLAB вектора-столбца размерностью [7х1]:

```
>> A=[3;4;1.7;16;11;8.33;10]
```

```
A =
```

```
3.0000  
4.0000  
1.7000  
16.0000  
11.0000  
8.3300  
10.0000
```

Пример ввода в рабочем окне MATLAB вектора-строки размерностью [1х5]:

```
>> A=[0.5,4,3.7,6,12.75]
```

```
A =
```

```
0.5000 4.0000 3.7000 6.0000 12.7500
```

Пример ввода в рабочем окне MATLAB матрицы размерностью [3x3]:

```
>> A=[1.4,2,3;7,9,11;0.2,7,3.2]
```

A =

```
1.4000 2.0000 3.0000
7.0000 9.0000 11.0000
0.2000 7.0000 3.2000
```

Обратите внимание, что элементы матрицы в строке отделяются друг от друга запятыми, а сами строки разделяются точкой с запятой. Элементы в строке можно также отделять друг от друга пробелами.

## 2. Формирование векторов и подматриц

Оператор «:» является очень полезным оператором языка MATLAB. Он применяется для формирования векторов и матриц или для выделения из них подвекторов, подматриц, подблоков массива.

Предположим, вы хотите создать вектор со значениями от 1 до 9. Пример, как это сделать, не вводя последовательно каждое число:

```
>> X=1:9
```

X =

```
1 2 3 4 5 6 7 8 9
```

Условное выражение 1:9 используется для представления вектора чисел от 1 до 9 с приращением на 1. Приращение может быть задано в виде среднего из трех аргументов:

```
>> X=1:3:10
```

X =

```
1 4 7 10
```

Приращение может быть дробным или отрицательным, например, 0:0.2:1 или 100:-2:50.

Зададим матрицу A следующим образом:

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

A =

```
1 2 3 4 5
5 4 3 2 1
7 5 3 1 9
2 4 6 8 2
1 3 2 5 4
```

и выделим из него подблок, включающий в себя строки со второй по четвертую столбцов с первого по третий следующим образом:

```
>> A(2:4,1:3)
```

```
ans =
    5    4    3
    7    5    3
    2    4    6
```

Оператор  $A(4, :)$  позволяет выделить все столбцы четвертой строки массива  $A$

```
>> A(4,:)
```

```
ans =
    2    4    6    8    2
```

Оператор  $A(:, 4)$  позволяет выделить все строки четвертого столбца массива  $A$

```
>> A(:,4)
```

```
ans =
    4
    2
    1
    8
    5
```

Операторы умножения «.\*» и правого деления «./» с точкой используются при перемножении или делении массивов (каждое число первого массива умножается/делится на соответствующее число второго массива).

Например:

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

```
A =
```

```
1 2 3
6 5 4
7 8 9
```

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



B =

```
1 3 5
2 4 6
9 8 7
```

>> A.\*B

ans =

```
1 6 15
12 20 24
63 64 63
```

Операторы умножения и правого деления без точки применяются при перемножении или делении матриц по правилам линейной алгебры.

Например:

>> A\*B

ans =

```
32 35 38
52 70 88
104 125 146
```

При применении оператора левого деления с точкой «.\» и оператора без точки «\» выполняется решение систем линейных уравнений вида  $AX=B$  по методу наименьших квадратов для матриц и векторов. Например:

>> A\B

ans =

```
-1.3164 0.2425 1.8014
2.6329 -0.4850 -3.6029
-1.3164 0.2425 1.8014
```

>> A.\B

ans =

```
1.0000 1.5000 1.6667
0.3333 0.8000 1.5000
1.2857 1.0000 0.7778
```

При транспонировании массива (операция «.'») строки просто заменяются столбцами

```
>> A.'
ans =
     1     6     7
     2     5     8
     3     4     9
```

Для возведения каждого элемента в степень нужно произвести следующую операцию  $A.^n$ ,  $n$  в данном случае значение степени, в которую будут возводиться элементы матрицы  $A$ .

При транспонировании матрицы (операция «'» без точки) результатом является транспонированная матрица, для комплексных чисел выполняется операция комплексного сопряжения.

Над матрицами можно производить разнообразные операции.

Например, с помощью команды *diag* возможно формировать или извлекать диагонали матрицы.

Функция  $X=diag(v)$  формирует квадратную матрицу  $X$  с вектором  $v$  на главной диагонали;

Функция  $X=diag(v,k)$  формирует квадратную матрицу  $X$  порядка  $length(v)+abs(k)$  с вектором  $v$  на  $k$ -той диагонали.

Функция  $v=diag(X,k)$  извлекает из матрицы  $X$  диагональ с номером  $k$ ; при  $k>0$  – это номер  $k$ -й верхней диагонали, при  $k<0$  – это номер  $k$ -й нижней диагонали.

Функция  $B=reshape(A,m,n)$  возвращает матрицу размером  $[m \times n]$ , сформированную из элементов матрицы  $A$  путем их последовательной выборки по столбцам.

Функция  $sum(A)$  позволяет найти сумму всех элементов матрицы  $A$ .

Функции  $max(A)$  и  $min(A)$  возвращают соответственно максимальный и минимальный элементы вектора  $A$ .

Для генерирования матриц полезны следующие простые функции:

*zeros(m,n)* - нулевая матрица

*ones(m,n)* - матрица, состоящая из одних единиц

*eye(m,n)* - матрица с 1 на диагонали и 0 вне диагонали

*rand(m,n)* - матрица со случайными элементами, равномерно распределенными на отрезке  $[0, 1]$

*randn(m,n)* - матрица со случайными элементами, распределенными по нормальному закону с математическим ожиданием, равным 0, и средним квадратическим отклонением, равным 1.

В приведенных выше функциях параметры  $m$  и  $n$  определяют число строк и столбцов матрицы соответственно. Каждую из предыдущих функций можно вызвать с одним параметром — в этом случае генерируется квадратная матрица указанного порядка с соответствующим свойством. Например, команда *eye(m)* генерирует единичную матрицу порядка  $m$ .

### Задание для выполнения работы.

1. В рабочем окне Matlab сгенерировать квадратную матрицу  $A$  размерностью  $6 \times 6$ .
2. Выделить из матрицы  $A$  подматрицу  $B$  размерностью  $4 \times 4$ .
3. Выделить из матрицы  $A$  подматрицу  $C$  размерностью  $4 \times 4$  и не равную  $B$ .
4. Извлечь из матрицы  $A$  диагональ  $D$ .
5. Умножить матрицу  $B$  на матрицу  $C$  с точной и без точки.
6. Разделить матрицу  $B$  на матрицу  $C$  левым делением с точной и без точки.
7. Разделить матрицу  $B$  на матрицу  $C$  правым делением с точной и без точки.
8. Транспонировать матрицу  $A$ .
9. Найти сумму элементов матрицы  $A$ .
10. Найти матрицу, каждый элемент которой равен квадрату соответствующего числа матрицы  $A$

## Лабораторная работа №3

### Построение графиков и гистограмм в Matlab

Цель работы: изучение основных команд черчения в программе MATLAB для построения графиков и гистограмм.

#### Построение графиков командой *ezplot*

Простейший способ построить график функции с одной переменной - использовать команду *ezplot*, которая может работать со строкой, символьным выражением или анонимной функцией, представляющих функцию для графического вывода. Например, чтобы построить график функции  $2x^2 + 3x + 1$  в интервале от -2 до 2 (используя строчную форму команды *ezplot*), введите следующее:

```
>> ezplot ('2*x^2+3*x+1', [-2;2])
```

Чертеж графика будет отображен на экране в новом окне, озаглавленном «Figure 1».

Используя символьное выражение, вы можете воспроизвести чертеж на Рис. 1, введя в командную строку следующее:

```
>> syms x, ezplot ('2*x^2+3*x+1', [-2;2])
```

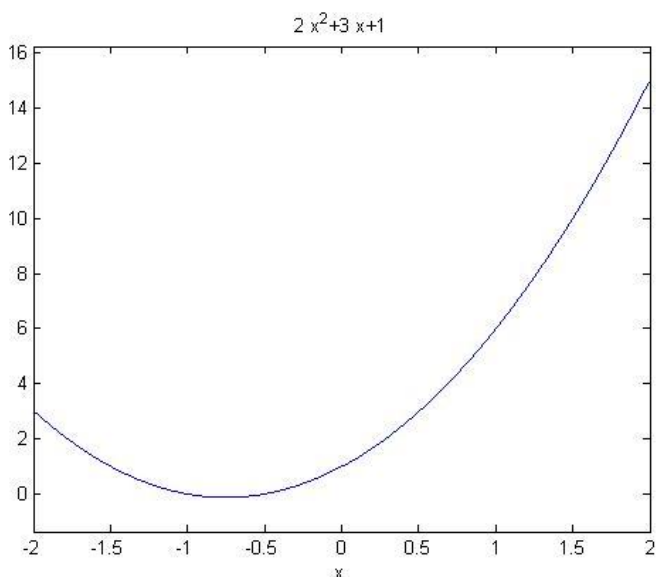


Рис. 1. Парабола  $y = 2x^2 + 3x + 1$

## Построение графиков командой `plot`

Команда `plot` работает с векторами числовых данных. Синтаксис команды представляет собой: `plot (X, Y)`, где `X` и `Y` являются векторами одинаковой длины. Например:

```
>> X = [1 3 5]; Y = [4 8 7]; plot (X,Y)
```

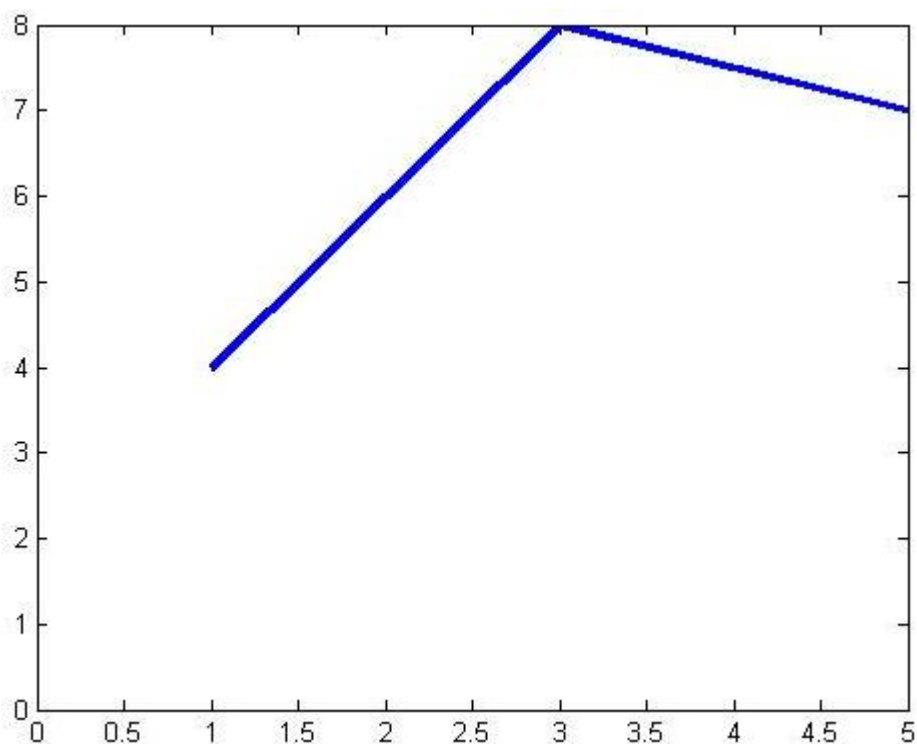


Рис. 2. Построение линейных сегментов

В этом случае мы отделили несколько команд в одной строке с помощью точки с запятой, вместо запятой. Обратите внимание, что вывод команд, предшествующих знаку точка с запятой, запрещается.

Команда `plot` рассматривает вектора `X` и `Y`, как перечни координат последовательных точек на графике, и соединяет точки в виде линейных сегментов. Таким образом, на Рис. 2 показано, как программа MATLAB соединяет точки с координатами (1, 4), (3, 8) и (5, 7).

Чтобы начертить график функции  $x^2$  в интервале от -1 до 2, сначала требуется создать перечень `X` из значений `x`, а затем ввести `plot (X, X.^2)`. Точка в данном выражении обязательна, так как `X.^2` представляет собой поэлементное возведение в квадрат вектора `X`, но не матричный квадрат. Нам необходимо использовать достаточное количество значений `x` для уверенности в том, что результирующий график, нарисованный путем соединения точек, будет выглядеть

нормально (плавная, а не ломаная линия). Мы используем приращение в размере 0.01. Таким образом, чтобы отобразить график параболы, введите:

Результат отображен на Рис. 3. Обратите внимание, что мы использовали точку с запятой, чтобы запретить вывод вектора X из 301 элемента.

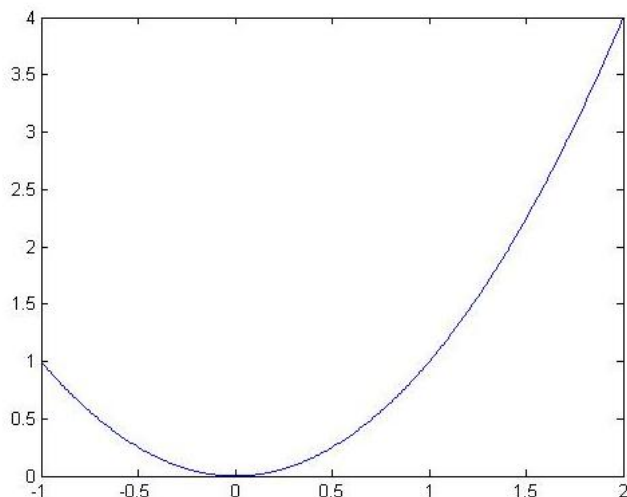


Рис. 3. Построенная парабола

### Построение нескольких кривых

Каждый раз, когда вы выполняете команду построения (черчения), программа MATLAB стирает старый чертеж и рисует новый. Если вы хотите произвести наложение двух или более чертежей, используйте команду *hold on*. Эта команда дает программе MATLAB инструкцию – сохранять старые изображения и рисовать любые новые изображения поверх старых. Эта инструкция будет выполняться до тех пор, пока вы не введете команду *hold off*. Ниже приведен пример с использованием команды *ezplot*:

```
>> ezplot('exp(-2*x)', [0 10])
```

```
>> hold on
```

```
>> ezplot('cos(x)', [0 10])
```

```
>> hold off
```

```
>> title 'exp(-2x) and cos(x)'
```

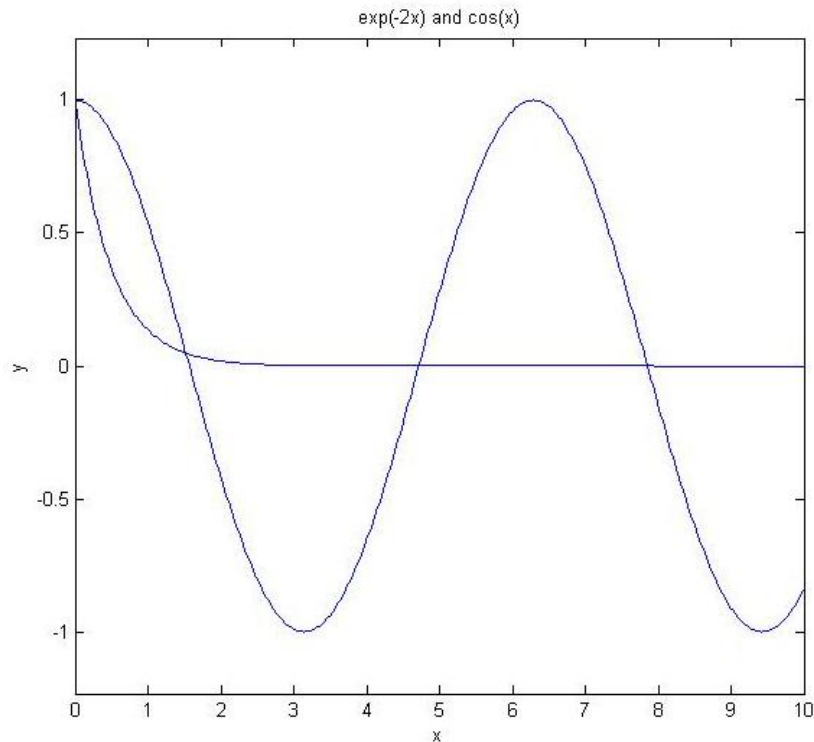


Рис. 4. Две пересекающиеся кривые

С помощью команды *plot* вы можете сразу чертить несколько кривых. Например:

```
>> X = 0:0.01:10; plot (X, exp(-2*X), X, cos(X))
```

Обратите внимание, что вектор координат оси *x* должен задаваться один раз для каждой функции, которая будет выводиться в виде графика.

### Видоизменение графиков

Вы можете изменять график различными способами. Вы можете изменить название графика, введя следующее (в окне Command Window (Командное окно), а не в окне рисунка):

```
>> title 'желаемое название графика'
```

Такое же изменение можно осуществить прямо в окне рисунка, выбрав команду Edit >> Axes Properties (Редактирование >> Свойства осей) в верхней части этого окна. (Просто введите новое название в поле, обозначенное Title (Название)). Чтобы вставлять названия или текст в изображение, вы можете использовать команды *text*, *xlabel*, *ylabel*, *zlabel* и *legend* в дополнение к команде *title*. Команды *xlabel*, *ylabel*, *zlabel* добавляют текст рядом с координатными осями, команда *legend* отображает «Легенду» изображения, а команда *text* добавляет текст

в заданной точке. Эти команды имеют опциональные аргументы, которые можно использовать для изменения типа или размера шрифта текста.

Можно также изменить вертикальный и горизонтальный диапазоны графика с помощью команды *axis*. Например, чтобы ограничить вертикальный диапазон интервалом от -1 до 4, введите:

```
>> axis ([-2 2 -1 4])
```

Первые два числа представляют диапазон горизонтальной оси; в команду должны быть включены оба диапазона, даже если хотите изменить только один.

Чтобы создать изображение квадрата графика, введите *axis square*; при этом шкалы будут одинаковыми на обеих осях, если диапазоны x и y имеют равную длину. Для диапазонов любой длины вы можете создать одинаковую шкалу на обеих осях, не изменяя изображение, вводом команды *axis equal*. Обычно эта команда расширяет один из диапазонов, если это необходимо. Однако, если вы нечаянно обрежете часть графика, эта утраченная часть не исчезнет из памяти программы MATLAB. Вы можете вновь настроить диапазоны с помощью команды *axis* или ввести команду *axis tight*, чтобы автоматически задать диапазоны, которые будут включать график полностью. Чтобы узнать о дополнительных возможностях, введите *help axis*. (Не забудьте сначала ввести *more on*, если вы хотите прочитать сразу всю страницу).

Другим немаловажным способом изменения типа графики является модифицирование цвета или типа линий чертежа, или изменение пометок и маркировки на осях. С помощью команды *plot* вы можете изменять цвет графика, чертить штриховой или точечной линией, помечать начерченные точки символами, просто добавляя третий аргумент в строку для каждой пары x-y. Символы для цветов следующие: 'y' -желтый, 'm' -пурпурный, 'c' -голубой, 'r' -красный, 'g' -зеленый, 'b' -синий, 'w' -белый и 'k' -черный. Символы маркеров точек включают в себя 'o' -кружок, 'x' -крест, '+' -плюс и '\*' -звездочка. Символы типов линий включают в себя '-' -сплошная линия, ':' -точечная линия и '--' -штриховая линия. Если задан тип точки, но нет типа линии, тогда чертятся только точки, но между ними не чертится соединяющая их линия.

### **Панель инструментов изображения**

Каждое окно изображения имеет панель инструментов, расположенную под строкой меню и содержащую значки (ярлыки) для нескольких элементов меню, в том числе и значки для открытия, сохранения и распечатки изображений. В



середине находится несколько значков, соответствующих нескольким элементам меню Tools (Инструменты).

Два значка со знаками плюса и минуса служат для управления масштабом (увеличение и уменьшение соответственно). Чтобы увеличить какую-либо точку на вашем графике, щелкните мышью на значке со знаком плюса, а затем щелкните мышью на необходимой вам точке. Вы можете делать это много раз; каждый раз масштаб будет изменяться по обеим осям приблизительно в два раза. Однако не следует щелкать слишком быстро, так как двойной щелчок мышью возвращает график к исходному состоянию. Аналогично, щелкая мышью на значке со знаком минуса, можно постепенно уменьшать изображение. Программа MATLAB 7 имеет также значок с изображением руки, который позволяет передвигать мышью график горизонтально и вертикально внутри текущих осей (панорамирование). К дополнительным параметрам масштабирования можно получить доступ, щелкнув правой кнопкой мыши на окне изображения при включенном режиме масштабирования, используя подменю Options (Параметры) в меню Tools (Инструменты) или применяя команду.

Чтобы модифицировать объекты в окне изображения с помощью мыши, в этом окне должен быть активирован режим редактирования. В программе MATLAB 6 и более поздних вы можете включить или отключить режим редактирования, выбрав команду меню Tools -> Edit Plot (Инструменты -> Редактировать чертеж) или щелкнув мышью на значке со стрелкой справа от значка печати. Когда режим редактирования включен, этот значок со стрелкой выделен, и установлен флажок рядом с пунктом Edit Plot (Редактировать чертеж) в меню Tools (Инструменты). Далее рекомендуется щелкнуть мышью на объекте в окне изображения для его редактирования. При щелчке мышью на объекте в окне изображения весь объект должен выделяться маленькими черными квадратами. Если этого не произойдет, тогда вам потребуется включить режим редактирования.

## Построение гистограмм

Классическая гистограмма характеризует числа попаданий значений элементов вектора  $Y$  в  $M$  интервалов с представлением этих чисел в виде столбцовой диаграммы. Для получения данных для гистограммы служит функция *hist*, записываемая в следующем виде:

- $N=\text{hist}(Y)$  — возвращает вектор чисел попаданий для 10 интервалов, выбираемых автоматически. Если  $Y$  — матрица, то выдается массив данных о числе попаданий для каждого из ее столбцов;
- $N=\text{hist}(Y,M)$  — аналогична вышерассмотренной, но используется  $M$  интервалов ( $M$  — скаляр);
- $N=\text{hist}(Y,X)$  — возвращает числа попаданий элементов вектора  $Y$  в интервалы, центры которых заданы элементами вектора  $X$ ;
- $[N,X]=\text{HIST}(\dots)$  — возвращает числа попаданий в интервалы и данные о центрах интервалов.

Команда  $\text{hist}(\dots)$  с синтаксисом, аналогичным приведенному выше, строит график гистограммы. В следующем примере строится гистограмма для 1000 случайных чисел и выводится вектор с данными о числах их попаданий в интервалы, заданные вектором  $x$ :

```
>> x=-3:0.2:3;
>> y=randn(1000,1);
>> hist(y,x)
>> h=hist(y,x)
h =
Columns 1 through 14
    4    0    0    6   11    9   14   21   34   46   52   57   61   68
Columns 15 through 28
   90   73   74   90   66   63   48   29   22   18   15   10    4    3
Columns 29 through 31
    5    3    4
```

Построенная гистограмма представлена на рис. 5.

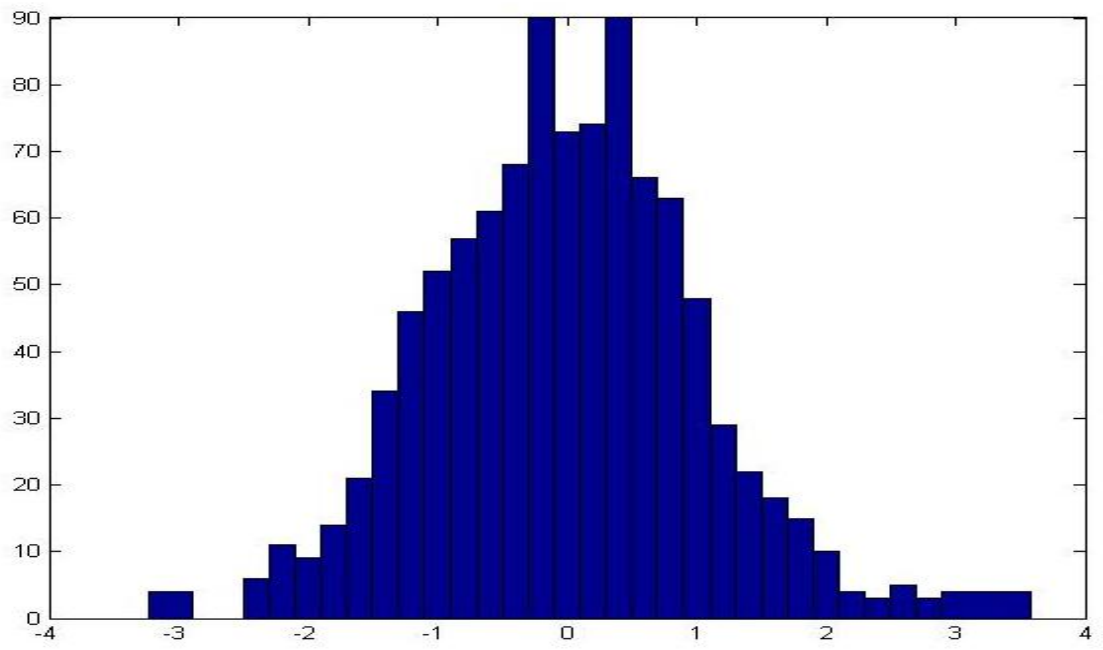


Рис. 5. Пример построения гистограммы.

## Задание

### Вариант №1

1. Используйте команды `plot` или `ezplot`, чтобы построить графики следующих функций:

$$y = x^3 - x^2 \text{ и } y = e^{x^2} - 5 \text{ для } -2 < x < 2$$

на одной и той же системе координат.

С полученный график отредактировать следующим образом:

- А) с помощью команды `axis` сделать более удобными оси координат;
- Б) С помощью команд `title`, `xlabel` и `ylabel` изменить название графика и осей;
- В) сделать линии двух функций разного цвета, толщины и с различными маркерами.

2. Постройте гистограмму для 500 случайных чисел, интервалы попадания задаются следующим вектором  $-2,5 < x < 3,5$ .

**Лабораторная работа № 4**  
**Подбор формул по данным опыта**  
**методом наименьших квадратов в MATLAB**

**Цель работы:**

- изучить математический аппарат, используемый при обработке опытных данных;
- получить практические навыки обработки экспериментальных данных в MATLAB.

**1. Общие положения**

В практической работе часто зависимость между переменными величинами получается в результате опыта (измерений). Обычно в таком случае эта зависимость оказывается заданной в виде таблицы. Функции, заданные таким образом, могут входить в дальнейшие операции и расчеты. Для удобства пользования такими зависимостями необходимо сперва подобрать формулу, хорошо описывающую опытные данные. Подбор формулы, описывающей результаты опыта, является существенной частью обработки экспериментальных данных. Одним из методов получения таких формул является способ наименьших квадратов.

**1.1. Метод наименьших квадратов**

Пусть в результате опытов найдены некоторые значения  $x_i$  и соответствующие им значения  $y_i$ , которые заданы таблицей (табл.1).

Таблица 1

$x$	$x_1$	$x_2$	...	$x_i$	...	$x_n$
$y$	$y_1$	$y_2$	...	$y_i$	...	$y_n$

Требуется найти зависимость  $y = f(x)$ . Такой зависимостью может быть:

$$y = ax + b \text{ – линейная,}$$

$$y = bx^a \text{ – степенная,}$$

$$y = be^{ax} \text{ – показательная,}$$

$$y = b + a \ln x \text{ – логарифмическая,}$$

$$y = \frac{a}{x} + b \text{ – гиперболическая и т.д.}$$

Метод наименьших квадратов позволяет подобрать более точные значения параметров  $a$  и  $b$ . Предварительно необходимо установить общий

вид аналитической функции, который можно выявить по опытным данным, если их нанести на плоскость с координатами X-Y (рис.1).

Зависимость  $y$  от  $x$ , изображаемая аналитической функцией  $y = f(x)$ , не может совпадать с экспериментальными значениями во всех  $n$  точках. Это означает, что для всех или некоторых точек имеем разность  $\Delta_i = y_i - f(x_i)$ , отличную от нуля.

Метод наименьших квадратов заключается в том, что подбираются параметры  $a$  и  $b$  таким образом, чтобы сумма квадратов разностей была наименьшей, т.е.

$$z = \sum_{i=1}^n \Delta_i^2 = \sum_{i=1}^n [y_i - f(x_i)]^2 \rightarrow \min$$

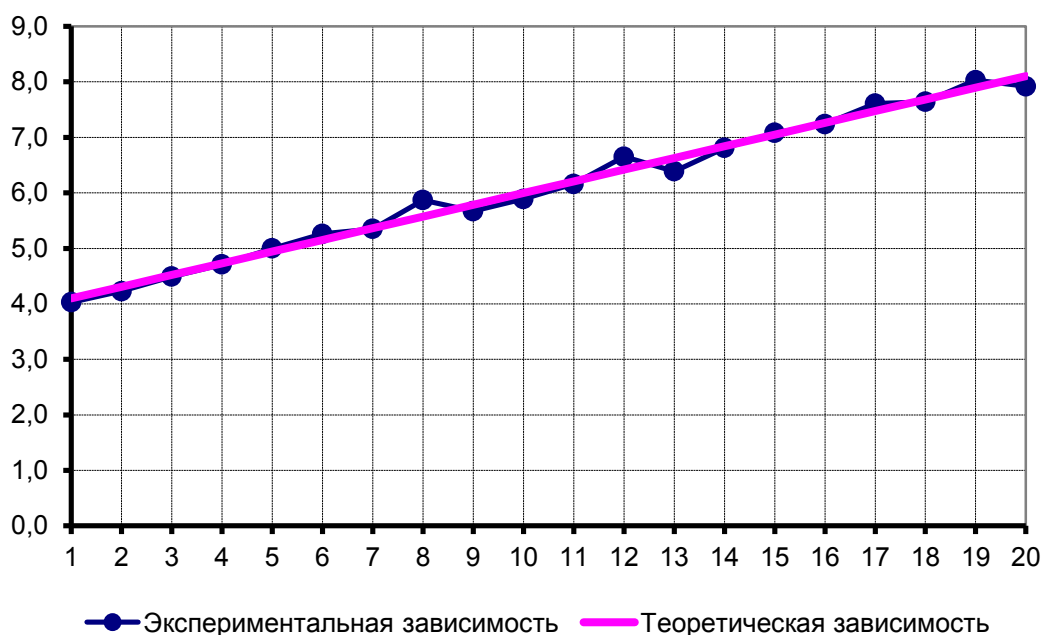


Рис. 1. Теоретическая и экспериментальная зависимости

Если вид функции  $y = f(x)$  установлен, то ее можно представить в виде

$$y = f(x) = \varphi(x, a, b),$$

где  $a$  и  $b$  – искомые параметры, тогда

$$z = \sum_{i=1}^n [y_i - \varphi(x_i, a, b)]^2 \rightarrow \min . \quad (1)$$

Для нахождения минимума выражения (1) вычислим частные производные по аргументам  $a$  и  $b$  и приравняем эти производные к нулю, получим:

$$\begin{cases} \frac{\partial z}{\partial a} = 2 \sum_{i=1}^n [y_i - \varphi(x_i, a, b)] \varphi'_a(x_i, a, b) = 0, \\ \frac{\partial z}{\partial b} = 2 \sum_{i=1}^n [y_i - \varphi(x_i, a, b)] \varphi'_b(x_i, a, b) = 0. \end{cases} \quad (2)$$

Система (2) содержит два уравнения с двумя неизвестными  $a$  и  $b$ . Решив систему (2), найдем значения параметров  $a$  и  $b$ . При найденных значениях параметров величина  $z$  будет наименьшей, т.е. аналитическая зависимость будет наилучшим образом описывать экспериментальные данные.

## 1.2. Линейная регрессия

Пусть эмпирические данные необходимо описать зависимостью  $y = ax + b$ , т.е.  $y = \varphi(x, a, b) = ax + b$ .

Тогда, согласно методу наименьших квадратов, запишем

$$z = \sum_{i=1}^n (y_i - ax_i - b)^2 \rightarrow \min \quad (3)$$

Выбираем числа  $a$  и  $b$  так, чтобы величина  $z$  была наименьшей, для чего частные производные выражения (3) по  $a$  и  $b$  получим

$$\begin{cases} \frac{\partial z}{\partial a} = 2 \sum_{i=1}^n (y_i - ax_i - b)(-x_i) = 0, \\ \frac{\partial z}{\partial b} = 2 \sum_{i=1}^n (y_i - ax_i - b)(-1) = 0. \end{cases} \quad (4)$$

Эти два условия дают следующую систему уравнений

$$\begin{cases} \sum_{i=1}^n x_i y_i - a \sum_{i=1}^n x_i^2 - b \sum_{i=1}^n x_i = 0, \\ \sum_{i=1}^n y_i - a \sum_{i=1}^n x_i - nb = 0. \end{cases} \quad (5)$$

Откуда

$$\begin{cases} a = \frac{\sum x_i y_i - \frac{\sum x_i \sum y_i}{n}}{\sum x_i^2 - \frac{(\sum x_i)^2}{n}}; \\ b = \frac{\sum y_i}{n} - \frac{a \sum x_i}{n}. \end{cases} \quad (6)$$

Для определения численной величины параметров  $a$  и  $b$  составляется расчетная таблица (табл. 2), программа для расчета на ЭВМ или выполняется расчет средствами MS Excel.

Таблица 2

$i$	$x_i$	$y_i$	$x_i y_i$	$x_i^2$
1	$x_1$	$y_1$	$x_1 y_1$	$x_1^2$
2	$x_2$	$y_2$	$x_2 y_2$	$x_2^2$
...	...	...	...	...
$n$	$x_n$	$y_n$	$x_n y_n$	$x_n^2$
	$\sum_{i=1}^n x_i$	$\sum_{i=1}^n y_i$	$\sum_{i=1}^n x_i y_i$	$\sum_{i=1}^n x_i^2$

## 2. Выполнение работы

Используя опытные данные, рассчитать коэффициенты регрессии с помощью программы MATLAB.

Одномерный массив вводится следующим образом:

**X=[1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20]**

Перемножение каждого элемента массива на такой же элемент другого массива:

**XY=X.\*Y**

Возведение каждого элемента массива X в квадрат:

**X2=X.^2**

Суммирование массива производится с помощью команды **sum(X)**.

Подсчет количества элементов массива:

**n=length(X)**

График строится с помощью команды **plot(X, Y)**

Редактирование осей: **axis([x<sub>min</sub> x<sub>max</sub> y<sub>min</sub> y<sub>max</sub>])**.

Чтобы построить два графика на одной системе координат воспользуемся командой **hold on**. Пример:

```
plot(X, Y)
axis([0 20 0 4])
hold on
plot(X.*a+b)
```

## 3. Задание к лабораторной работе

1. Для заданного статистического ряда найти коэффициенты регрессии  $a$  и  $b$ .
2. По полученным данным построить график теоретической и экспериментальной зависимостей (см рис. 2).
3. Сделать выводы по работе.



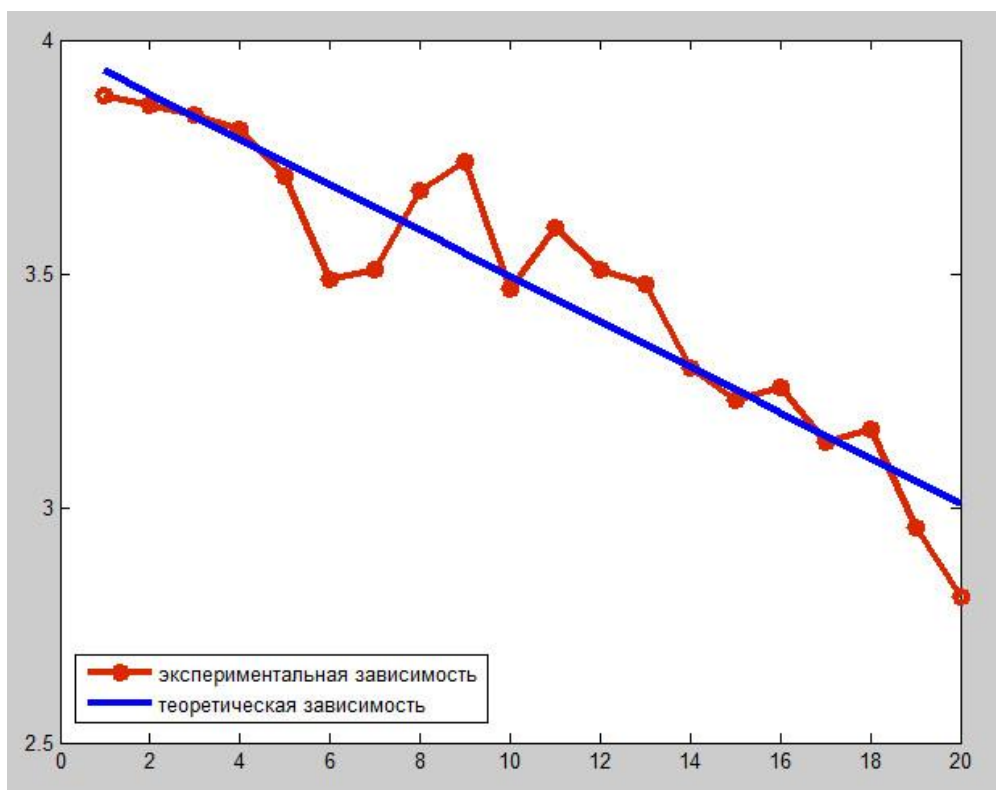


Рис. 2. Полученный график в MATLAB

#### 4. Варианты заданий к лабораторной работе

$X_i = i$	Значения $Y_i = Y(X_i)$							
	Вариант 1	Вариант 2	Вариант 3	Вариант 4	Вариант 5	Вариант 6	Вариант 7	Вариант 8
1	3,88	4,08	3,90	4,03	3,82	5,998	6,030	5,850
2	3,86	4,18	3,82	4,23	3,44	5,820	6,072	5,619
3	3,84	4,38	3,60	4,49	3,16	5,754	6,297	5,569
4	3,81	4,46	3,47	4,71	2,95	5,828	6,428	5,426
5	3,71	4,44	3,31	5,00	2,73	5,627	6,425	5,237
6	3,49	4,55	3,05	5,26	2,40	5,597	6,473	5,025
7	3,51	4,67	3,14	5,35	2,27	5,693	6,592	4,988
8	3,68	4,89	2,89	5,87	1,85	5,469	6,815	5,037
9	3,74	4,86	2,66	5,67	1,88	5,413	6,786	4,586
10	3,47	5,04	2,53	5,89	1,32	5,526	6,925	4,575
11	3,60	5,22	2,35	6,16	1,18	5,344	7,116	4,445
12	3,51	4,99	2,49	6,65	1,15	5,304	7,053	4,353
13	3,48	5,39	2,19	6,39	0,85	5,352	7,224	3,933
14	3,30	5,56	1,82	6,81	0,48	5,301	7,439	3,899
15	3,23	5,42	1,69	7,08	0,18	5,424	7,302	3,793
16	3,26	5,85	1,54	7,24	-0,01	4,966	7,426	3,473
17	3,14	5,99	1,22	7,61	-0,12	5,080	7,797	3,551
18	3,17	5,85	1,17	7,64	-0,60	5,256	7,871	3,171
19	2,96	6,01	1,04	8,03	-0,68	5,090	7,929	3,330
20	2,81	5,97	1,12	7,92	-0,54	5,053	8,060	3,044

	Вариант 9	Вариант 10	Вариант 11	Вариант 12	Вариант 13	Вариант 14	Вариант 15	Вариант 16
1	6,310	5,650	20,5	20,9	2,02	19,9	22,3	1,35
2	6,308	5,431	19,4	20,5	1,98	20,3	22,9	1,87
3	6,546	5,250	19,2	21,9	1,67	22,0	22,7	1,88
4	6,855	5,000	18,7	21,8	1,65	23,9	26,2	2,30
5	7,073	4,790	17,7	21,7	1,57	21,9	27,2	2,82
6	7,770	4,569	18,8	22,7	1,42	26,1	28,2	2,87
7	7,225	4,296	17,1	25,8	1,37	26,5	31,3	3,07
8	7,739	4,065	16,0	27,3	1,07	26,0	34,9	3,74
9	7,995	3,837	15,6	28,2	0,85	25,5	38,2	3,97
10	8,063	3,519	14,0	30,4	0,48	24,9	39,5	4,36
11	8,247	3,281	15,0	30,3	0,35	25,0	42,2	4,40
12	8,472	2,926	12,6	34,5	-0,30	25,2	44,8	4,51
13	8,627	2,801	9,9	36,2	-0,61	24,4	50,6	4,57
14	8,936	2,546	9,7	38,5	-1,20	23,5	55,0	5,00
15	9,082	2,232	9,1	41,9	-1,39	22,6	56,8	5,60
16	9,076	2,016	7,1	44,5	-1,76	21,9	61,9	5,69
17	9,363	1,794	4,3	48,5	-2,28	22,4	64,2	5,70
18	9,679	1,663	5,4	50,6	-2,81	23,4	70,4	5,83
19	9,846	1,375	1,9	56,3	-3,57	19,6	75,7	6,16
20	10,013	1,217	0,1	59,1	-4,06	21,9	81,0	7,82

	Вариант 17	Вариант 18	Вариант 19	Вариант 20
1	5,95	1,3	7,5	0,12
2	6,37	1,6	10,6	0,13
3	6,70	1,7	11,0	0,15
4	6,88	1,8	11,4	0,18
5	7,06	1,8	15,2	0,18
6	7,50	1,9	16,2	0,19
7	7,66	2,1	17,3	0,20
8	8,10	2,1	17,4	0,22
9	8,15	2,2	18,0	0,24
10	8,47	2,4	18,2	0,24
11	8,59	2,5	21,2	0,24
12	8,59	2,5	21,5	0,24
13	8,69	2,7	22,1	0,25
14	8,76	2,7	23,0	0,26
15	9,95	2,7	25,0	0,27
16	11,16	2,8	25,9	0,27
17	11,50	2,9	28,4	0,28
18	11,84	2,9	29,0	0,29
19	13,97	3,0	29,1	0,30
20	15,12	3,2	29,7	0,31

## Лабораторная работа № 5.

### Законы распределения дискретной случайной величины

Цель работы:

- изучить основные характеристики биномиального закона и закона Пуассона;
- освоить методику построения многоугольника распределения и графика функции распределения дискретной случайной величины;
- получить практические навыки расчета вероятностных задач на ЭВМ в пакете прикладных программ Matlab.

#### 1. Общие положения

Случайной величиной называется такая величина, которая в результате опыта может принять то или иное значение, которое с точностью нельзя предсказать до опыта.

Между частными значениями случайной величины и вероятностями их появления существует определенная зависимость. Указанная зависимость называется законом распределения дискретной случайной величины. Закон распределения случайной величины может задаваться в виде таблицы, графика или формулы.

Все случайные величины делятся на дискретные и непрерывные. Дискретная случайная величина принимает фиксированные значения на отрезке  $[a, b]$ . Непрерывная случайная величина принимает на отрезке  $[a, b]$  любое значение.

Основными вероятностями закона распределения дискретной случайной величины являются биномиальный закон и закон Пуассона.

#### 2. Биномиальный закон распределения

Биномиальное распределение возникает при выполнении следующих условий:

- в результате одного испытания может появиться одно из двух противоположных событий  $\bar{A}$  или  $\bar{A} = B$ ;
- вероятности указанных событий от опыта к опыту не меняются и составляют  $P(A) = p$  и  $P(B) = q$ ;
- производится  $n$  независимых испытаний.

При выполнении указанных условий возникают различные комбинации таких событий, вероятность появления которых определяется по формуле, называемой биномиальным законом распределения

$$P(m, n) = \frac{n!}{m!(n-m)!} q^{n-m} p^m = C_n^m q^{n-m} p^m; \quad (1)$$

где  $n$  – число независимых испытаний;

$P_{(m, n)}$  – вероятность того, что при  $n$  испытаниях событие  $A$  появится ровно  $m$  раз;

$p$  и  $q$  – вероятности появления соответственно событий  $A$  и  $B$ , где  $q = 1 - p$ ;

$C_n^m$  – число сочетаний из  $n$  элементов по  $m$ .

Для биномиального распределения числовых характеристик: математическое ожидание  $M(m)$  и дисперсия  $D(m)$  выражаются с помощью формул:

$$M[m] = np \quad \text{и} \quad D[m] = npq .$$

### 3. Распределение Пуассона

Распределение Пуассона представляет собой предельный случай биномиального распределения для условий, когда  $P \rightarrow 0$ ;  $n \rightarrow \infty$  и  $np = a$ .

Преобразуя выражение биномиального закона при приведенных выше условиях, получим формулу распределения Пуассона

$$P(m, n) = \frac{a^m}{m!} e^{-a}, \quad (2)$$

где  $n$  – число испытаний;

$m$  – число появления события  $A$  ( $m = 0, 1, 2, \dots$ );

$P_{(m, n)}$  – вероятность того, что при  $n$  испытаниях событие  $A$  появится ровно  $m$  раз;

$a$  – параметр закона ( $a = np$ );

$P$  – вероятность появления события  $A$  в одном испытании.

В связи с тем, что вероятность появления отдельных событий в распределении Пуассона характеризуется малой вероятностью ( $P \rightarrow 0$ ), закон Пуассона называют законом редких явлений.

Математическое ожидание  $M(m)$  и дисперсия  $D(m)$  для распределения Пуассона равны и определяются по выражению

$$M[m] = D[m] = np = a.$$

Закон Пуассона описывает:

- поток требований в зону ремонта и ТО;
- поток заявок на запасные части, узлы, агрегаты;
- случайное число отказов в течение фиксированной наработки.

#### 4. Дискретная случайная величина

Дискретная случайная величина, кроме формул 1 и 2, также может быть задана:

а) рядом распределения вероятности (табл. 1);

Таблица 1

Частные значения события А	$m_A$	0	1	2	3	...	$m$
Вероятности, отвечающие частным значениям появления случайной величины	$P(m,n)$	$P(0,n)$	$P(1,n)$	$P(2,n)$	$P(3,n)$	...	$P(m,n)$

б) многоугольником распределения вероятности появления события А равно: 0, 1, 2, 3, ...,  $m$  раз (рис. 1);

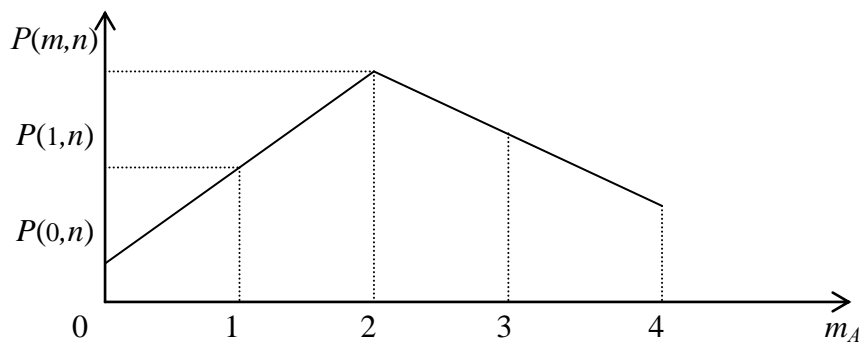


Рис. 1. Многоугольник распределения вероятности дискретной случайной величины

в) графиком функции распределения вероятности.

На основании ряда распределения или многоугольника распределения может быть построен график функции распределения (рис. 2).

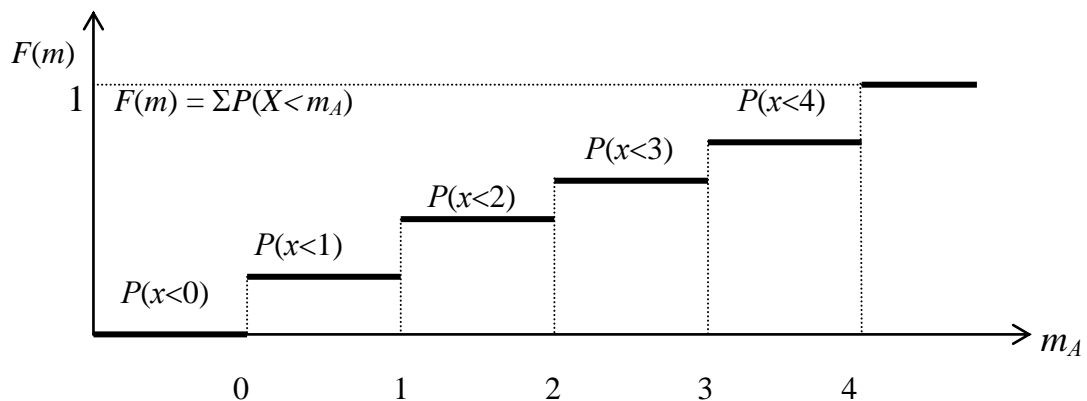


Рис. 2. График распределения вероятности дискретной случайной величины

## 5. Факториал в Matlab

Для выполнения данной лабораторной работы необходимо знать, как считается факториал в пакете прикладных программ Matlab. Для этого используется команда *factorial* (*n*), что является последовательным произведением всех натуральных чисел от 1 до *n*.

Например:

```
>> factorial (3)
```

```
ans =
```

```
6
```

## 6. Варианты заданий к лабораторной работе

Для выданного преподавателем варианта построить таблицу распределения вероятности, многоугольник распределения вероятности и график функции распределения вероятности.

**Вариант 1.** Вероятность того, что при текущем ремонте автомобиля требуется замена ведомого диска сцепления, равна  $p = 0,4$ . Определить вероятность того, что при 3-х текущих ремонтах потребуется 0, 1, 2, 3 диска.

**Вариант 2.** Для нормальной работы автобазы на линии должно быть не менее восьми автомобилей (в наличии имеется десять). Вероятность невыхода каждого автомобиля на линию равна 0,1. Найти вероятность нормальной работы автобазы на ближайший день.

**Вариант 3.** Дискретная случайная величина  $X$  – число выходов из строя конденсатора системы зажигания в течение одного года эксплуатации распределена по биномиальному закону. Вероятность выхода из строя конденсатора составляет 0,3. Рассматривается 4 автомобиля. Определить вероятность выхода из строя конденсатора у этих автомобилей.

**Вариант 4.** Автомобиль проходит ТО. Число неисправностей, обнаруженных во время ТО, распределяется по закону Пуассона с параметром  $a = 2$ . Если неисправностей не обнаружено, ТО автомобиля продолжается в среднем 2 часа. Если обнаружены 1 или 2 неисправности, то на устранение каждой из них требуется еще в среднем полчаса. Если обнаружено более 2-х неисправностей, то автомобиль становится на профилактический ремонт, где находится в среднем 4 часа. Определить закон распределения среднего времени  $T$  обслуживания и ремонта.

**Вариант 5.** Авторемонтная мастерская обслуживает 100 автомобилей. Вероятность того, что в течение дня поступит одна заявка на ремонт, равна 0,01. Найти вероятность того, что в течение дня поступит: а) три заявки; б) менее трех заявок; в) более трех заявок; г) хотя бы одна заявка.

**Вариант 6.** Устройство состоит из трёх независимо работающих элементов. Вероятность отказа каждого элемента в одном испытании равна 0,1. Составить закон распределения числа отказавших элементов в одном испытании.

**Вариант 7.** В партии 10% нестандартных деталей. Наудачу отобраны четыре детали. Написать биномиальный закон распределения дискретной случайной величины  $X$  – числа нестандартных деталей среди четырёх отобранных.

**Вариант 8.** Автомобиль выпущен тиражом 100 000 экземпляров. Вероятность того, что он собран с браком, равна 0,0001. Найти вероятность того, что тираж содержит 1, 2, 3, 4, 5 бракованных автомобилей.

**Вариант 9.** Устройство состоит из 1000 элементов, работающих независимо один от другого. Вероятность отказа любого элемента в течении времени  $T$  равна 0,002. Найти вероятность того, что за время  $T$  откажут 0, 1, 2, 3 элемента.

**Вариант 10.** Станок-автомат штампует детали. Вероятность того, что изготовленная деталь окажется бракованной, равна 0,01. Найти вероятность того, что среди 200 деталей окажется 0, 1, 2, 3, 4 бракованных.

**Вариант 11.** Завод отправил на базу 500 деталей. Вероятность повреждения детали в пути равна 0,002. Найти вероятности того, что в пути будет повреждено деталей: а) ровно три; б) менее трех; в) более трех; г) хотя бы одна.

**Вариант 12.** Пусть дискретная случайная величина  $X$  – число схода с линии автобусов по технической неисправности в течение одной смены работы распределена по биномиальному закону. Вероятность схода с линии составляет 0,2. Рассматривается 10 автомобилей. Определить вероятность схода с линии этих автобусов.

**Вариант 13.** Вероятность того, что при диагностировании автомобиля потребуются дальнейший текущий ремонт, равна 0,3. Определить вероятность того, что при диагностировании 3-х автомобилей возникнет необходимость в 0, 1, 2, 3 текущих ремонтах.

**Вариант 14.** В небольшом районном городе есть автобусный парк из 20 автобусов. Для успешного перевозочного процесса необходимо чтобы на линии ежедневно находилось не менее 16 автобусов. Вероятность невыхода каждого автомобиля на линию равна 0,05. Найти вероятность нормальной работы автобусного парка на ближайший день.

**Вариант 15.** Слесарь изготавливает детали. Вероятность того, что изготовленная деталь окажется бракованной, равна 0,01. Найти вероятность того, что среди 100 деталей окажется 0, 1, 2, 3 бракованных.



## **Лабораторная работа № 6**

### **Обработка экспериментальных данных в MATLAB**

#### **Цель работы:**

- изучить последовательность обработки экспериментальных данных;
- получить практические навыки обработки экспериментальных данных в MATLAB.

#### **1. Общие положения**

Задачи, возникающие при изучении процессов автомобильного транспорта, требуют знаний основных положений теории вероятностей и математической статистики.

Если математическая статистика изучает методы сбора и обработки результатов наблюдений случайных процессов, то теория вероятностей изучает их закономерности.

Случайные процессы – это такие процессы, которые при неоднократном воспроизведении одного и того же опыта протекают каждый раз несколько по другому.

При решении задач математической статистики и теории вероятностей приходится сталкиваться с понятием генеральной и выборочной совокупностей.

Генеральной совокупностью называют совокупность всех объектов (элементов), подлежащих изучению. Очевидно, что подвергать исследованию всю генеральную совокупность затруднительно или нецелесообразно. В связи с этим из генеральной совокупности извлекают лишь некоторую ее часть, называемую выборочной совокупностью (выборкой).

При изучении процессов автомобильного транспорта нередко возникает необходимость обработки больших объемов информации, представленной в виде статистических рядов. Например, пробег шин до замены, величина износа протектора шины у различных автомобилей на определенном пробеге, величина люфта рулевого управления автомобилей в процессе эксплуатации и т.д.

Для автоматизации обработки таких статистических рядов возможно использование программы MATLAB. Программа содержит сотни команд для работы в области математики. Использовать MATLAB можно для построения графиков функций, решения уравнений, выполнения статистических тестов и многого другого. Это высокоуровневый язык программирования, который способен взаимодействовать с другими языками программирования, например, Fortran и C. Можно создавать звук и анимационную графику. Можно производить симуляцию и моделирование. Можно подготавливать

материалы для экспортирования в Интернет. Кроме того, возможно использовать программу MATLAB для объединения математических вычислений с текстом и графикой с целью создания совершенных, интегрированных, интерактивных документов.

## 2. Выполнение лабораторной работы

Пусть имеем доброкачественный объем выборки (статистический ряд). Порядок обработки его следующий:

- зарегистрированные значения рассматриваемого признака  $X_i$  расположим в возрастающем порядке. Для этого воспользуемся командой **sort (A)**, где **A** – наш статистический ряд;
- найдем наибольшее  $X_{max}$  и наименьшее  $X_{min}$  значения параметра с помощью команд **max(A)** и **min(A)**;
- определим размах измерения значений параметра  $R = X_{max} - X_{min}$ ;
- вычислим число интервалов  $K$  в зависимости от объема выборки  $n$

$$K=1+3,32 \lg n$$

Сначала найдем объем выборки  $n$  с помощью команды **length(A)**; затем в командном окне напишем: **K=1+3.32\*log10(n)**;

- определим ширину частичного интервала  $h$

$$h = \frac{R}{k} ;$$

- определим границы интервалов, для чего установим нулевое (крайнее) значение интервала  $X_0$ :  $X_0 = X_{min} - h/2$ .

Следующие границы интервалов определяются последовательным прибавлением ширины интервала  $h$  к предыдущему значению границы:  $X_1 = X_0 + h$ ,  $X_2 = X_1 + h$  и т.д. до тех пор пока  $X_k$  не будет больше  $X_{max}$ . В MATLAB это можно сделать таким образом:

```
x=X0;i=1;  
while x < max (A)  
y(i)=x;  
x=x+h;  
i=i+1;  
end
```

Присвоим полученные значения обозначениям границ: **X0=y(1)**; **X(1)=y(2)**; и т.д. до последней границы;

- определим число элементов значений признаков, попавших в  $i$ -й интервал (эту величину называют опытной частотой  $m_i$ \* данного интервала):

$$m_i = \mathbf{length}(\mathbf{find}(\mathbf{A} > X_{i-1} \ \& \ \mathbf{A} < X_i));$$

- определим относительную величину частоты, которую называют частотой  $i$ -го интервала  $W_i$ :  $W_i = m_i/n$ .

– определим накопленную частоту  $W_i^H$ .

$$W_i^H = \sum_{i=1}^i W_i$$

Накопление частоты  $W^H$  получается путём последовательного прибавления частоты  $W_i$  очередного интервала:  $W_1^H = W_1$ ,  $W_2^H = W_1^H + W_2$ ,  $W_3^H = W_2^H + W_3$ , и т.д., для последнего интервала  $W_n^H = 1$ ;

– определим основные числовые характеристики выборки:

а) среднее арифметическое – **mean(A)**;

б) статистическая дисперсия – **var(A)**;

в) среднее квадратическое отклонение – **std(A)**.

– построим гистограмму распределения признака с помощью команды **hist(A, Xs)**, где Xs – вектор середин интервалов (середины интервалов нужно найти и сформировать вектор их значений). Гистограмма строится по опыт-ным частотам  $m_i$  или частотам  $W_i$  (рис.1). С помощью команд **axis**, **grid**, **xlabel**, **ylabel** оформите гистограмму как представлено на рисунке.

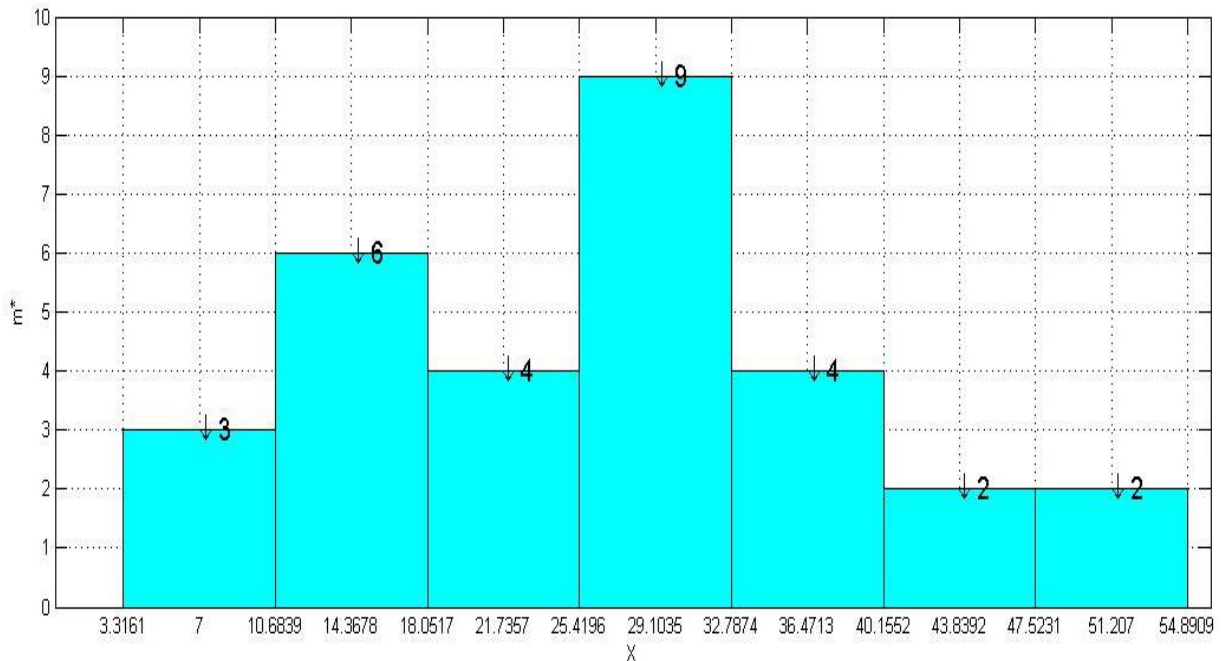
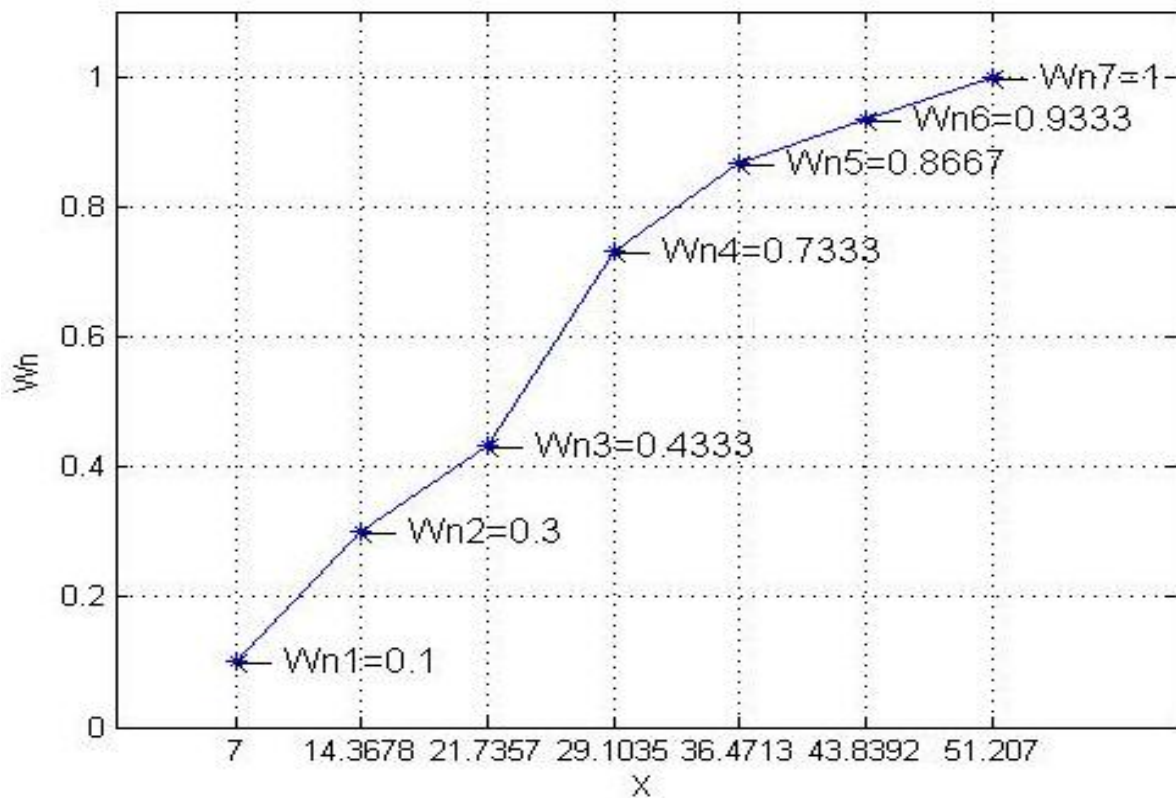


Рис. 1. Гистограмма распределения признака

Кумулятивная кривая строится по накопленным частотам  $W_i^H$ , она соответствует опытной функции распределения признака  $W_i^H = F_{on}^*(x)$  (рис.2). По оси X отложены середины интервалов.

**plot(Xs,Wn),**

где Xs – вектор середин интервалов, Wn – вектор накопленных частотей. С помощью команд **axis**, **grid**, **xlabel**, **ylabel** оформите график как представлено на рисунке.



*Рис. 2. Кумулятивная кривая*

### **3. Задание к лабораторной работе**

1. Обработать экспериментальные данные по предложенной выше методике.
2. По данным интервального вариационного ряда:
  - а) вычислить числовые характеристики ряда;
  - б) построить гистограмму и кумулятивную кривую.
3. Сделать выводы по работе.

### 3. Варианты заданий к лабораторной работе

Вариант 1	Вариант 2	Вариант 3	Вариант 4	Вариант 5	Вариант 6	Вариант 7	Вариант 8
7,0	4,7	0,03	9	4,8	31,8	36,9	0,14
7,8	7,5	0,20	6	5,7	31,7	70,8	0,22
10,0	8,0	0,10	14	5,9	32,1	207,0	0,36
12,0	8,4	0,13	24	6,7	32,6	83,0	0,23
11,4	11,0	0,14	8	6,8	32,7	91,8	0,18
16,4	11,5	0,08	8	12,0	37,3	120,0	0,30
16,7	13,0	0,07	12	12,7	38,4	39,2	0,11
17,8	13,5	0,13	17	12,4	39,7	70,9	0,19
18,0	14,3	0,22	8	16,8	40,4	84,3	0,11
18,8	16,1	0,16	12	16,9	43,1	94,6	0,19
22,0	17,4	0,17	18	18,3	107,0	42,7	0,19
23,5	18,2	0,12	10	18,7	7,8	71,3	0,22
25,0	20,7	0,15	7	18,9	23,5	84,7	0,15
25,9	21,5	0,09	13	36,2	31,8	94,7	0,15
26,6	21,7	0,14	8	36,8	29,2	133,9	0,23
28,2	25,1	0,21	9	37,4	47,5	53,3	0,20
28,7	25,9	0,12	9	37,6	59,3	71,3	0,35
28,7	28,4	0,05	14	37,7	12,2	85,1	0,24
30,7	29,2	0,10	9	38,1	25,0	96,3	0,25
31,8	29,3	0,18	7	38,5	32,1	59,5	0,24
32,0	29,4	0,19	9	39,7	50,6	71,8	0,30
32,1	29,7	0,08	14	42,4	30,0	85,4	0,18
35,0	30,0	0,14	7	43,0	50,4	96,6	0,26
35,9	31,9	0,17	10	46,4	66,5	61,7	0,29
36,5	30,2	0,19	10	48,7	16,4	72,0	0,21
37,4	30,7	0,09	16	134,6	25,9	85,5	0,26
43,6	32,0	0,14	14	50,7	35,9	98,4	0,28
45,1	33,0	0,13	11	54,8	55,1	136,5	0,14
48,0	34,0	0,14	16	57,0	35,9	63,7	0,18
50,5	35,9	0,14	12	59,3	52,3	75,0	0,25

Вариант 9	Вариант 10	Вариант 11	Вариант 12	Вариант 13	Вариант 14	Вариант 15	Вариант 16
7,5	132	722	1,6	0,9	299	3,17	1,35
10,6	325	834	0,8	0,6	836	3,18	4,40
15,2	446	848	0,3	1,0	326	1,35	5,69
16,2	490	887	0,5	0,8	327	4,40	2,30
17,3	493	1201	0,4	2,4	331	5,69	3,07
17,4	519	526	0,6	0,7	245	2,30	4,51
18,0	690	536	0,5	0,6	253	3,07	7,82
18,2	1124	570	0,7	0,6	256	4,51	1,87
21,2	545	600	0,6	1,0	257	7,82	4,57
21,5	551	615	0,6	0,9	268	1,87	5,83
22,1	560	650	3,4	1,5	269	4,57	2,82
23,0	570	677	1,5	0,9	270	5,83	3,74
25,0	616	697	1,8	1,5	274	2,82	5,70
25,9	651	705	2,2	0,9	277	3,74	5,00
28,4	670	722	1,7	1,5	297	5,70	3,97

29,0	76	397	1,3	1,4	593	0,50	5,60
11,0	782	410	1,1	0,6	632	3,97	1,88
29,1	105	420	1,2	1,1	674	5,60	2,87
11,4	321	430	2,2	1,8	721	1,88	4,36
29,7	350	440	1,5	1,0	778	2,87	6,16
13,5	359	451	1,8	0,9	184	4,36	3,59
30,0	365	458	1,7	1,1	197	6,16	5,23
14,3	374	475	1,2	0,9	198	3,59	7,20
31,3	436	488	1,0	0,9	241	5,23	1,78
15,8	451	500	0,8	1,4	385	7,20	2,66
31,9	480	514	1,6	0,9	397	1,78	3,65
17,2	505	294	1,8	0,7	424	2,66	5,60
35,9	541	300	1,8	0,9	430	3,65	3,61
18,0	235	319	1,6	1,4	464	5,50	5,51
43,6	250	302	1,9	0,7	487	3,61	1,80

Вариант 17	Вариант 18	Вариант 19	Вариант 20
8,69	7,5	0,18	2,2
11,50	10,6	0,26	2,7
13,97	15,2	0,22	2,8
7,06	16,2	0,24	2,1
8,15	17,3	0,20	2,4
8,76	17,4	0,27	1,3
11,84	18,0	0,24	2,7
15,12	18,2	0,31	1,7
8,47	21,2	0,13	2,7
6,37	21,5	0,30	1,6
7,50	22,1	0,18	2,9
8,59	23,0	0,24	2,9
9,95	25,0	0,28	3,0
6,70	25,9	0,12	1,8
7,66	28,4	0,27	3,2
8,59	29,0	0,15	1,8
11,16	11,0	0,19	1,9
6,88	29,1	0,24	2,1
8,10	11,4	0,29	2,5
5,95	29,7	0,25	2,5
7,18	13,5	0,17	1,8
8,54	30,0	0,21	2,6
9,66	14,3	0,25	3,2
6,17	31,1	0,21	2,5
7,20	15,8	0,25	2,4
8,55	31,9	0,20	2,0
9,84	17,2	0,21	1,7
13,65	35,9	0,29	1,7
4,27	18,0	0,25	2,4
7,13	43,6	0,17	2,2