

# Самостоятельная лабораторная работа №1

## Исследование функций пакета Scilab для обработки и визуализации данных

### 1. Цель

Изучение среды численного моделирования Scilab и ее базовых функций, приобретение практических навыков моделирования в среде Scilab.

### 2. Основные теоретические положения

Scilab – это кроссплатформенный свободно распространяемый математический пакет, обладающий сходным с Matlab синтаксисом встроенного языка. Пакет программ Scilab предназначен для выполнения математических вычислений и численного моделирования широкого спектра информационных и управляющих систем.

Пакет можно скачать по адресу <https://www.scilab.org/>. После установки пакета Scilab для получения справочных сведений о его возможностях и командах необходимо выбрать пункт меню «Справка» или нажать кнопку F1. Откроется окно «Справочная система» (рисунок 2.1), в котором можно выполнять поиск необходимых сведений как с использованием разделов содержания справки, так и по ключевым словам.

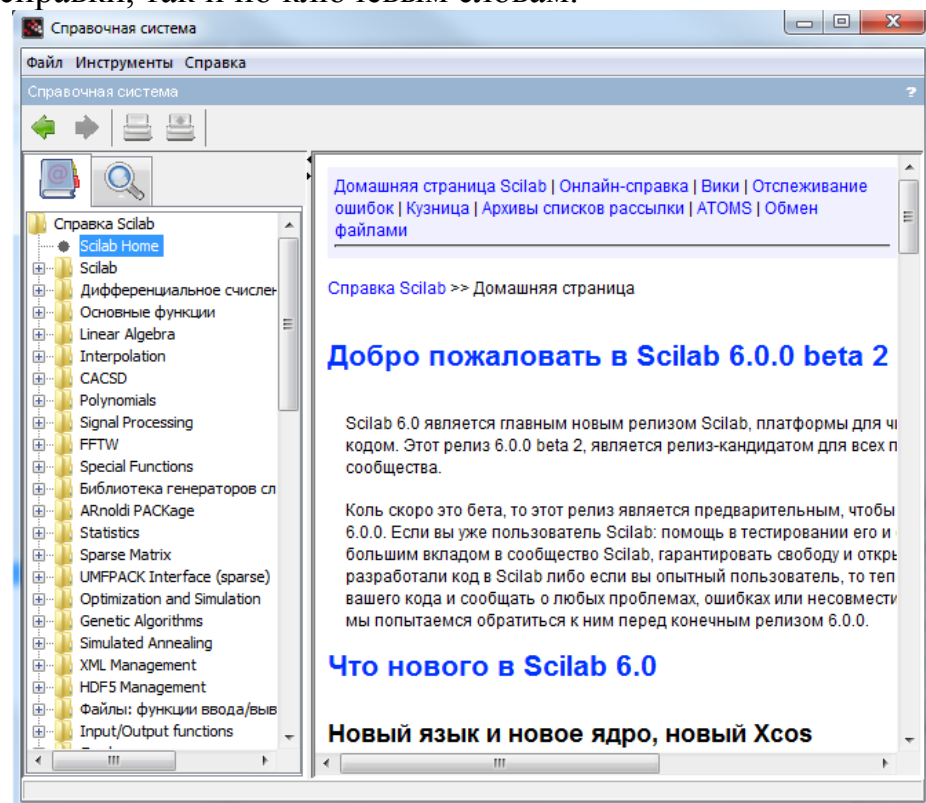


Рисунок 2.1 – Справочная система Scilab

Пакет Scilab позволяет выполнять команды в интерактивном режиме. Порядок ввода команд аналогичен пакету Матлаб. При этом используются следующие специфические обозначения:

```
// - начало однострочного комментария
% - начальный символ имени системных переменных, например:
    %i _мнимая единица ( $\sqrt{-1}$ );
    %pi _число  $\pi = 3.141592653589793$ ;
    %e _число  $e = 2.7182818$ ;
    %inf _машинный символ бесконечности ( $\infty$ );
    %NaN _неопределенный результат (0/0,  $\infty/\infty$  и т. п.);
    %eps _условный ноль %eps=2.220E-16;
    %t, %f – логические константы True и False.
```

Системные переменные используются в качестве констант в выражениях и не могут быть изменены пользователем. Ниже представлен пример ввода выражения, содержащего системные переменные и встроенные функции, и вычисления его значения в командном окне Scilab:

```
--> y= sin(%pi/4)+%e^2
y =
8.0961629
```

Scilab имеет расширенный перечень встроенных основных функций (см. п. меню «Справка»—> основные функции). Клавиша Tab может использоваться для автозавершения ввода имени функции или переменной (рисунок 2.2).

```
--> factor
```

Рисунок 2.2 – Пример автозавершения имени функции

Ввод в программу исходных данных с клавиатуры выполняется вызовом функции `x = input(message [, "string"])`, где `message` – строка, которая отображается на экране перед выполнением ввода, необязательный параметр `"string"` указывается при вводе строки, `x` – вещественное число или строка (если указан параметр `"string"`).

Для вывода на экран результатов вычислений может использоваться функция `disp`. Если требуется вывести комбинацию строковых значений и чисел, то требуется предварительно преобразовать числа в строки с помощью функции `string`:

```
--> t=19
--> disp("Средняя температура "+string(t)+" градусов")
```

Средняя температура 19 градусов

Для форматированного вывода на экран значений переменных может использоваться функция, аналогичная функции `printf` языка Си. Например:

```
--> printf("Результат: exp(3) = %f",exp(3))
Результат: exp(3) = 20.085537
```

Ввод-вывод данных из файлов и в файлы выполняется в Scilab с использованием функций `mopen`, `mfscanf`, `mfprintf`, которые полностью аналогичны функциям языка Си: `fopen`, `fscanf`, `fprintf`.

При программировании в Scilab необходимо различать понятия **сценарий** и **функция, определяемая пользователем**.

**Сценарием** называют любую программу Scilab. Любой сценарий состоит из последовательности **инструкций** (команд), которые описывают конкретные действия с объектами Scilab. Сценарии создаются в окне редактора SciNotes, который может быть вызван из пункта меню «Инструменты». Сценарии сохраняются в \*.sce файлах.

**Функция, определяемая пользователем**, также является сценарием, но характеризуется наличием имени, благодаря которому к ней можно обратиться из любого места сценария. Имеется два способа определения функций: `function` и `deff`. Определение функции с использованием конструкции `function` аналогично большинству языков программирования и не требует пояснений. Например:

```
function z=f(x,y)
    z=1/(exp(-y)+exp(-x))
endfunction
```

Обычно такое определение функции сохраняется в файле \*.sce на диске (например, f.sce). Чтобы использовать функцию, необходимо загрузить её из файла. Для этого необходимо выполнить команду `exec`. Её единственным аргументом является строка с указанием положения сохраненного файла в файловой системе. Например, если файл f.sce был сохранен в домашнем каталоге пользователя, то команда загрузки будет выглядеть следующим образом: `exec('~/f.sce')`. После этого функция, описанная в файле, может быть вызвана в текущей сессии, например `f(1,2)`.

Определение функции с использованием оператора `deff` реализуется в соответствии с шаблоном:

```
deff('[имя1,...,имяN]=имя_функции(переменная_1,...,переменная_M)',
    'имя1=выражение1;...;имяN=выражениеN'),
```

где список `[имя1,...,имяN]` представляет имена выходных значений функции, `выражение1,..., выражениеN` — правила вычисления этих значений.

Листинг ниже демонстрирует определение функции  $z=f(x,y)$  с помощью оператора `deff` и использование `f` для вычисления значения функции:

```
--> deff('z=f(x,y)', 'z=1/(exp(-y)+exp(-x))');  
--> a=1; b=2; c=f(a,b)  
c =  
1.9872232
```

Scilab хранит установки, функции и переменные в рабочей области, называемой *Workspace*. Переменные рабочей области отображаются в окне обозревателя переменных. Для очистки рабочей области от всех пользовательских переменных используется команда `clear all`. Если требуется удалить одну переменную, то применяется команда `clear <имя переменной>`.

Система Scilab предназначена для выполнения математических вычислений. Обозначение математических операций и управляющих конструкций (`if`, `for`, `while` и др.) такое же, как и в системе Matlab. Рассмотрим примеры некоторых операций с одномерными (векторы) и двумерными массивами (матрицы).

Вектор-строка задается своими элементами, которые записываются внутри квадратных скобок и разделяются пробелами или запятыми. Для задания вектора-столбца его элементы следует разделять точкой с запятой (;). Например:

```
--> xves=[1 2 3] // вектор-строка  
xves =  
1. 2. 3.  
--> yves=[1;2;3] // вектор-столбец  
yves =  
1.  
2.  
3.
```

Векторы удобно создавать с помощью оператора “:”. Например, если требуется создать вектор из нечетных чисел от 1 до 10, то можно применить оператор:

```
--> x=1:2:10 // Xнач:Шаг:Xкон  
x =  
1. 3. 5. 7. 9.
```

Переменную заданную как массив можно использовать в арифметических выражениях в качестве аргумента стандартных математических функций. Результатом выполнения таких операторов являются массивы.

Для определения длины вектора используется функция `length`. Для получения *i*-го компонента вектора применяется команда `x(i)`, а чтобы получить компоненты вектора со 2-го по 4-й, используется инструкция

```
--> x(2:4)
ans =
    3.    5.    7.
```

Матрица представляется в виде набора векторов-строк, разделяемых точкой с запятой:

```
--> m=[1 2 3;4 5 6]
m =
    1.    2.    3.
    4.    5.    6.
```

Размер матрицы определяется с помощью функции `size`

```
--> size(m)
ans =
    2.    3.
```

Для выделения строки или столбца матрицы используют оператор “:”, например:

```
--> m(2,:)
ans =
    4.    5.    6.
--> m(:,3)
ans =
    3.
    6.
```

Матрицы и векторы можно формировать, составляя их из ранее заданных матриц и векторов:

```
--> v1=[1 2 3]; v2=[4 5 6]; v3=[7 8 9];
--> //Горизонтальная конкатенация векторов–строк:
--> V=[v1 v2 v3]
V = 1 2 3 4 5 6 7 8 9
--> //Вертикальная конкатенация векторов–строк,
--> //результат матрица:
--> V=[v1; v2; v3]
V =
    1 2 3
    4 5 6
    7 8 9
--> //Горизонтальная конкатенация матриц:
--> M=[V V V]
M =
```

```

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

```

--> //Вертикальная конкатенация матриц:

```
--> M=[V;V]
```

M =

```

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

```

Для работы с матрицами в Scilab применяются матричные операции: “+” — сложение; “-” — вычитание; “’” — транспонирование; “\*” — матричное умножение; “^” — возведение в степень; “\” — левое деление; “/” — правое деление; “.\*” — поэлементное умножение матриц; “.^” — поэлементное возведение в степень; “.\” — поэлементное левое деление; “./” — поэлементное правое деление. Эти операции аналогичны операциям Matlab.

Примеры действий над матрицами:

```
-->A=[1 2 0;-1 3 1;4 -2 5];
```

```
-->B=[-1 0 1;2 1 1;3 -1 -1];
```

```
--> //Вычислить  $(A^T+B)^2 - 2A(0.5B^T-A)$ 
```

```
-->(A'+B)^2-2*A*(1/2*B'-A)
```

ans =

```

10. 8. 24.
11. 20. 35.
63. - 30. 68.

```

```
--> //Решить матричные уравнения  $A \cdot X=B$  и  $X \cdot A=B$ .
```

```
-->A=[3 2;4 3];
```

```
-->B=[-1 7;3 5];
```

```
--> //Решение матричного уравнения  $AX=B$ : решение  $X=A^{-1} B$ 
```

```
-->X=A\B
```

X =

```

- 9. 11.
13. - 13.

```

```
--> //Решение матричного уравнения  $XA=B$ : решение  $X=B A^{-1}$ 
```

```
-->X=B/A
```

X =

```

- 31. 23.
- 11. 9.

```

```
-->X*A-B // проверка
```

ans =

```

0. 0.
0. 0.

```

Для работы с матрицами и векторами в Scilab существуют специальные функции: ones(m,n) – создает матрицу из единиц; zeros(m,n) – создает матрицу

из нулей; `eye(m,n)` - создает единичную матрицу. Функции для вычисления различных числовых характеристик матриц:

- `sum` – сумма элементов массива;
- `prod` – произведение элементов массива;
- `max`, `min` – максимальное и минимальное значение массива;
- `mean`, `median` – среднее и медианное значение массива;
- `det` – определитель квадратной матрицы;
- `rank` – ранг матрицы;
- `norm` – норма квадратной матрицы;
- `cond` – число обусловленности матрицы (произведение нормы исходной матрицы на норму обратной матрицы);
- `spec` – *собственные значения и собственные векторы* квадратной матрицы;
- `inv` – вычисляет обратную матрицу;
- `pinv` – вычисляет псевдообратную матрицу;
- `linsolv` – решает систему линейных алгебраических уравнений;
- `svd` – выполняет сингулярное разложение матрицы и др.

Построение двумерного графика в отдельном графическом окне в координатах  $x, y$  осуществляется по команде `plot(x,y)`. Команда позволяет также задавать цвет и стиль изображения точек на графике, например команда `plot(1,2,"*r")` отобразит звездочку красным цветом в позиции с координатами (1,2). Для отображения графика функции вида  $y=f(x)$  удобно значения  $x$  задавать в виде вектора, который формируется с помощью вызова функции `x=linspace(a,b,n)`, где параметры  $a$  и  $b$  задают интервал определения функция, а  $n$  – число точек разбиения этого интервала. На рисунке 2.3 приведен пример программы, определяющей функции  $f(x)$  и  $g(x)$  и отображающей их графики в одной системе координат с помощью `plot`, соответственно красным и зеленым цветом:

```
function y=f(x)
    y=(x^2+2*x)*exp(-x/2)
endfunction
function y=g(x)
    y=cos(2*x)*exp(-x/2)
endfunction
x=linspace(-2,10,100);
plot(x,f,"r",x,g,"g")
xgrid //отображение сетки
// отображение надписей
xlabel('Функции f(x) и g(x)',
    'Позиция','Ускорение')
//легенда: 4 – правый нижний угол, %t - рамка
legend('f(x)', 'g(x)',4,%t)
```

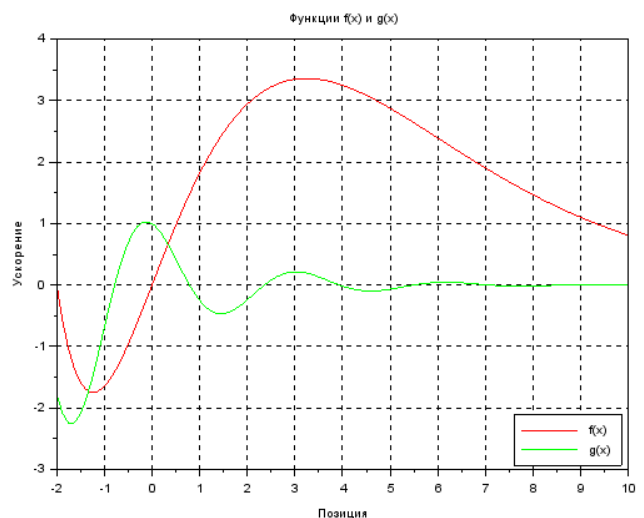


Рисунок 2.3 – Построение двумерных графиков с помощью `plot`

Функция `subplot(m,n,p)` разбивает графическое окно на  $m$  окон по вертикали и  $n$  окон по горизонтали, текущим окном становится окно с номером  $p$ . Применяется для отображения нескольких графиков – каждый в своем подокне.

Команда `figure(i)` делает активным  $i$ -ое окно. Для очистки текущего графического окна применяется функция `clf`. Функция `xdel(i)` закрывает графические окна, где  $i$  – вектор номеров окон.

Scilab позволяет отображать поверхности и кривые в 3-х мерном пространстве с помощью функции `surf`. Эта функция имеет три параметра:  $x$ ,  $y$  и  $z$ . Векторы  $x$  и  $y$ , размерами  $m$  и  $n$ , определяют координатные точки по осям  $Ox$  и  $Oy$ . Матрица  $z$ , размером  $m \times n$ , хранит высоты в координатных узлах. Для отображения поверхности, определяемой функцией  $z=f(x,y)$ , необходимо предварительно определить  $z$  во всех координатных узлах с помощью вызова функции `feval(x,y,f)`. Функция `feval` возвращает транспонированную матрицу, размером  $m \times n$ . Поэтому при вызове функции `surf` матрицу  $z$  требуется транспонировать. Удобнее 3-d поверхности строить с использованием функций `plot3d` и `plot3d1`. Сравнительный пример построения 3-D поверхности с помощью указанных функций представлен на рисунке 2.4.

```
function z=f(x, y)
    z=3*x^2+y^2;
endfunction
x=linspace(-1,1,50);
y=linspace(-2,2,100);
z=feval(x,y,f);
clf
figure(1)
subplot(1,3,1)
surf(x,y,z')
subplot(1,3,2)
plot3d(x,y,z)
subplot(1,3,3)
plot3d1(x,y,z)
```

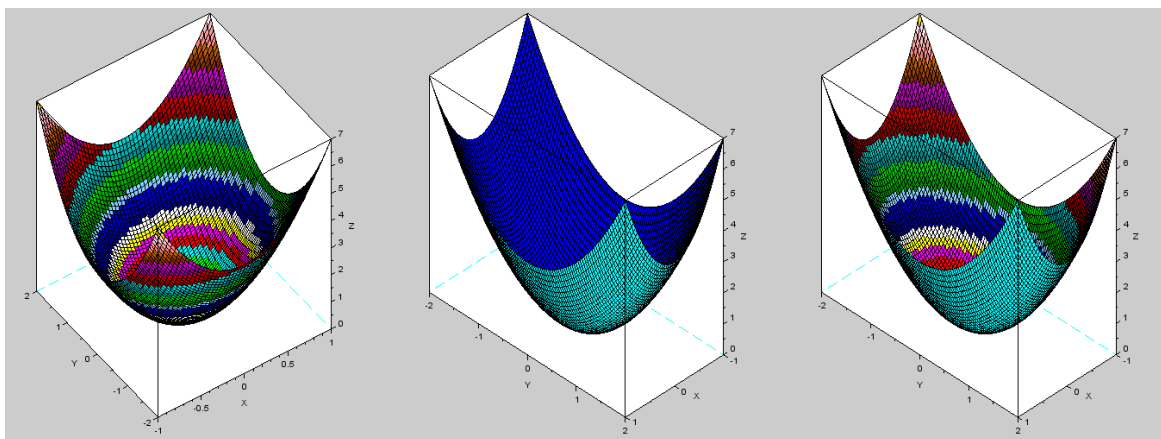


Рисунок 2.4 – 3-D поверхности, построенные функциями `surf`, `plot3d`, `plot3d1`



В Scilab имеются и другие функции для построения 2-х и 3-х мерных графиков, например: plot2d, mesh, contour, plot3d2, plot3d3, param3d и др.

Scilab позволяет выполнять статистическую обработку данных в массивах. Функция mean(M) вычисляет среднее всех элементов матрицы M, mean(M,'r') – среднее по столбцам, а mean(M,'c') – среднее по строкам. Функция stdev(M) вычисляет стандартное отклонение, а функция variance(M) – дисперсию всех элементов массива M. Специальная функция tabul(M) вычисляет частоты появления значений в массиве M. Для генерации псевдослучайных чисел могут использоваться функции grand и rand. Функция rand(n1,n2,...nn[, p]) - формирует многомерную матрицу случайных чисел, необязательный параметр p - это символьная переменная, с помощью которой можно задать тип распределения случайной величины ('uniform' - равномерное, 'normal'- гауссовское). На рисунке 2.4 изображен листинг программы для генерации матрицы размером 10x10 из нормально распределенных случайных чисел с нулевым средним и единичной дисперсией, а также построена гистограмма распределения.

```
--> x=rand(10,10,'normal');  
--> mean(x)  
ans =  
-0.0218062  
--> variance(x)  
ans =  
1.1118125  
--> histplot(12,x);
```

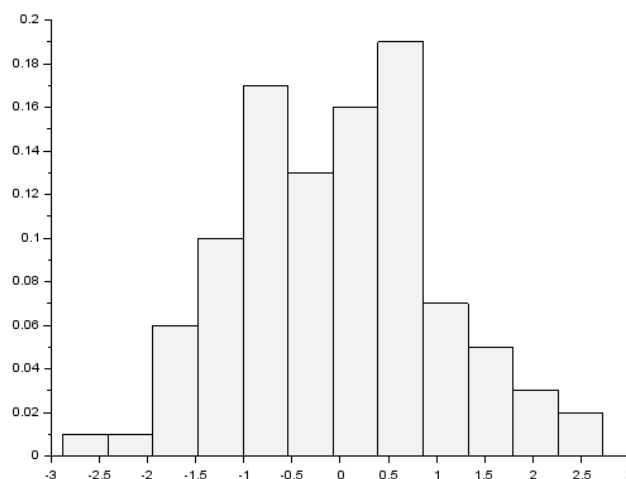


Рисунок 2.4 – Пример использования функций rand и histplot

В функции grand доступны более лучшие генераторы случайных чисел в том смысле, что они имеют как более длинный период, так и более лучшие статистические свойства. Кроме этого, функция grand обладает большими функциональными возможностями, в частности, позволяет генерировать случайные числа по различным законам распределения. Ниже приведены некоторые примеры вызова функции grand:

```
x=grand(400,500,'nor',0,1); //нормальное распределение  
x=grand(1,1000,'unf',-1,1); // равномерное распред., числа в диапазоне [-1,1]  
x=grand(1,1000,'exp',3); // экспоненциальное распределение, лямбда=3  
x=grand(1,5000,'chi',10); // распределение хи квадрат с 10-ю степенями  
свободы  
x=grand(1,5000,'poi',2); // распределение Пуассона с мат. ожиданием 2
```

### 3 Общее описание пакета Scilab

В лабораторных работах используется пакет Scilab с дополнительным модулем моделирования нейронных сетей Neuralnetworks 2.0.

При запуске Scilab появляется главное окно, показанное на рисунке 3.1.

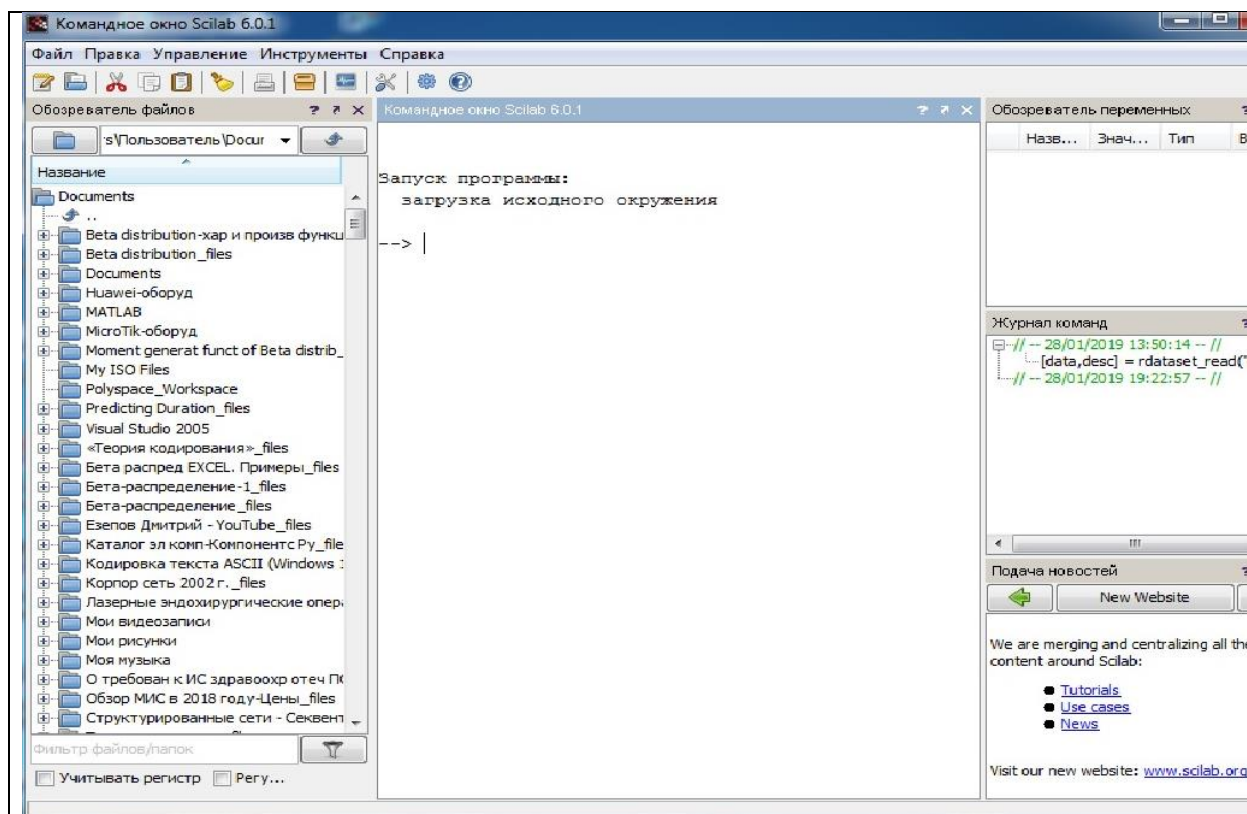


Рисунок 3.1 – Главное окно системы Scilab

Оно содержит окно обозревателя файлов, окно обозревателя переменных, журнал команд, окно новостей и командное окно. При выборе одного из окон его рамка окрашивается синим цветом.

Управление Scilab осуществляется через систему меню и командное окно. Появление стрелки вида `-->` (*prompt*) свидетельствует о готовности Scilab воспринимать команду.

Для установки модуля моделирования нейронных сетей необходимо в командном окне ввести команду инсталляции модуля:

```
atomsInstall("neuralnetwork")
```

В этом случае из сети Интернет автоматически будет загружен и установлен Neural Network Module 2.0. При необходимости модуль может быть установлен вручную. Необходимые компоненты располагаются по адресу <https://atoms.scilab.org/toolboxes/neuralnetwork/2.0>

## 4 Варианты заданий и программа работы

4.1 Ознакомиться с особенностями построения системы Scilab, основными командами и функциями, а также правилами построения сценариев [2].

4.2 Выбрать вариант задания, взяв остаток от деления номера студента в списке группы на 15. Выполнить задания 4.3-4.6 в соответствии с выбранным вариантом.

4.3 Определить функцию  $f(x)$  и вычислить N её значений на заданном отрезке. На экран вывести значения аргумента и значения функции. Построить график функции  $f(x)$ , снабдив его всеми необходимыми надписями и координатной сеткой.

1. Функция  $f(x) = \frac{\sin x \cos x}{x + \cos x}$ , N=15, на отрезке  $[0, 2\pi]$ .

2. Функция  $f(x) = \ln(2x - 1)\sqrt{e^{-x} + 4e^x}$ , N=10, на отрезке  $[0.7, 4]$ .

3. Функция  $f(x) = (\sin x - 1)\sqrt{e^{-\cos x} + \operatorname{tg} x e^x}$ , N=20, на отрезке  $[0.05, 1]$ .

4. Функция  $f(x) = x \sin x + \frac{e^{-x} - e^x}{e^{-x} + e^x}$ , N=30, на отрезке  $[0, 1]$ .

5. Функция  $f(x) = x (\sin x + \cos x) \frac{x - \operatorname{ctg} 2x}{1 + \sin^2 x}$ , N=10, на отрезке  $[0, \pi/2]$ .

6. Функция  $f(x) = \frac{x}{1 + \sqrt[3]{x+1}}$ , N=30, на отрезке  $[10, 100]$ .

7. Функция  $f(x) = \frac{\operatorname{sh} x + \operatorname{ch} x}{\operatorname{th} x + 5}$ , N=20, на отрезке  $[-3, 3]$ .

8. Функция  $f(x) = x^3 |\sin x + \cos x| + x - \operatorname{tg} 2x$ , N=10, на отрезке  $[0, \pi/2]$ .

9. Функция  $f(x) = \log_2(\sin x + 1)\sqrt{e^{-\cos x}}$ , N=30, на отрезке  $[0, \pi/2]$ .

10. Функция  $f(x) = \frac{\sqrt[3]{x^2+1}}{\sqrt{|x|+0.5}} \frac{x}{1 + \sin^2 x}$ , N=10, на отрезке  $[-\pi/2, \pi/2]$ .

11. Функция  $f(x) = \frac{\sin^2(2x + \frac{\pi}{2})}{\sqrt{\frac{|x|+1}{4}}} + \frac{x \ln x}{1+x^2}$ , N=20, на отрезке  $[-\pi/2, \pi/2]$ .

12. Функция  $f(x) = \frac{\sqrt{(3x+1)^3}}{x+2} + \sin^2(\frac{2x-1}{x+1})$ , N=15, на отрезке  $[0, 1]$ .

13. Функция  $f(x) = \sqrt{\log_2(\sin x + 1)} + \sqrt{1 + x^{3.5}}$ , N=30, на отрезке  $[0, \pi/2]$ .

14. Функция  $f(x) = 4 \cos^2\left(\frac{x+\pi}{x+2}\right) - \ln\left(\frac{e^{-x} - e^x}{e^{-x} + e^x}\right)$ , N=20, на отрезке  $[0, 1]$ .

15. Функция  $f(x) = \ln\left(\sqrt{\frac{2x-1}{x+1}}\right) + \sin(\sqrt{2+x})$ , N=10, на отрезке  $[0.7, 4]$ .

#### 4.4 Если возможно, то вычислить матрицу обратную D

1.  $D = 2(A^2 + B)(2B - A)$ , где

$$A = \begin{pmatrix} 2 & 3 & -1 \\ 4 & 5 & 2 \\ -1 & 0 & 7 \end{pmatrix}, \quad B = \begin{pmatrix} -1 & 0 & 5 \\ 0 & 1 & 3 \\ 2 & -2 & 4 \end{pmatrix}$$

2.  $D = 3A - (A + 2B)B^2$ , где

$$A = \begin{pmatrix} 4 & 5 & -2 \\ 3 & -1 & 0 \\ 4 & 2 & 7 \end{pmatrix}, \quad B = \begin{pmatrix} 2 & 1 & -1 \\ 0 & 1 & 3 \\ 5 & 7 & 3 \end{pmatrix}$$

3.  $D = 3A^2 - (A + 2B)B$ , где

$$A = \begin{pmatrix} 4 & 5 & -2 \\ 3 & -1 & 0 \\ 4 & 2 & 7 \end{pmatrix}, \quad B = \begin{pmatrix} 2 & 1 & -1 \\ 0 & 1 & 3 \\ 5 & 7 & 3 \end{pmatrix}$$

4.  $D = (A - B^2)2(2A + B^3)$ , где

$$A = \begin{pmatrix} 5 & 2 & 0 \\ 10 & 4 & 1 \\ 7 & 3 & 2 \end{pmatrix}, \quad B = \begin{pmatrix} 3 & 6 & -1 \\ -1 & -2 & 0 \\ 2 & 1 & 3 \end{pmatrix}$$

5.  $D = 2(A - B)(A^2 + B)$ , где

$$A = \begin{pmatrix} 5 & 1 & 7 \\ -10 & -2 & 1 \\ 0 & 1 & 2 \end{pmatrix}, \quad B = \begin{pmatrix} 2 & 4 & 1 \\ 3 & 1 & 0 \\ 7 & 2 & 1 \end{pmatrix}$$

6.  $D = (A - B)^2 A + 2B$ , где

$$A = \begin{pmatrix} 5 & -1 & 3 \\ 0 & 2 & -1 \\ -2 & -1 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 3 & 7 & -2 \\ 1 & 1 & -2 \\ 0 & 1 & 3 \end{pmatrix}$$

7.  $D = (A^2 - B^2)(A + B^2)$ , где

$$A = \begin{pmatrix} 7 & 2 & 0 \\ -7 & -2 & 1 \\ 1 & 1 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} 0 & 2 & 3 \\ 1 & 0 & -2 \\ 3 & 1 & 1 \end{pmatrix}$$

8.  $D = 2(A - B)(A^2 + B)$ , где

$$A = \begin{pmatrix} 5 & 1 & 7 \\ -10 & -2 & 1 \\ 0 & 1 & 2 \end{pmatrix}, \quad B = \begin{pmatrix} 2 & 4 & 1 \\ 3 & 1 & 0 \\ 7 & 2 & 1 \end{pmatrix}$$

9.  $D = 2A - (A^2 + B)B$ , где

$$A = \begin{pmatrix} 1 & 4 & 2 \\ 2 & 1 & -2 \\ 0 & 1 & -1 \end{pmatrix}, \quad B = \begin{pmatrix} 4 & 6 & -2 \\ 4 & 10 & 1 \\ 2 & 4 & -5 \end{pmatrix}$$

10.  $D = 2(A - 0,5B) + A^3B$ , где

$$A = \begin{pmatrix} 5 & 3 & -1 \\ 2 & 0 & 4 \\ 3 & 5 & -1 \end{pmatrix}, \quad B = \begin{pmatrix} 1 & 4 & 16 \\ -3 & -2 & 0 \\ 5 & 7 & 2 \end{pmatrix}$$

11.  $D = (A - B)A^2 + 3B$ , где

$$A = \begin{pmatrix} 3 & 2 & -5 \\ 4 & 2 & 0 \\ 1 & 1 & 2 \end{pmatrix}, \quad B = \begin{pmatrix} -1 & 2 & 4 \\ 0 & 3 & 2 \\ -1 & -3 & 4 \end{pmatrix}$$

12.  $D = 3(A^2 + B^2) - 2AB$ , где

$$A = \begin{pmatrix} 4 & 2 & 1 \\ 3 & -2 & 0 \\ 0 & -1 & 2 \end{pmatrix}, \quad B = \begin{pmatrix} 2 & 0 & 2 \\ 5 & -7 & -2 \\ 1 & 0 & -1 \end{pmatrix}$$

13.  $D = 2A^3 + 3B(AB - 2A)$ , где

$$A = \begin{pmatrix} 1 & -1 & 0 \\ 2 & 0 & -1 \\ 1 & 1 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} 5 & 3 & 1 \\ -1 & 2 & 0 \\ -3 & 0 & 0 \end{pmatrix}$$

14.  $D = A(A^2 - B) - 2(B + A)B$ , где

$$A = \begin{pmatrix} 2 & 3 & 1 \\ -1 & 2 & 4 \\ 5 & 3 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 2 & 7 & 13 \\ -1 & 0 & 5 \\ 5 & 13 & 21 \end{pmatrix}$$

15.  $D = (2A - B)(3A + B) - 2A^2B$ , где

$$A = \begin{pmatrix} 1 & 0 & 3 \\ -2 & 0 & 1 \\ -1 & 3 & 1 \end{pmatrix}, \quad B = \begin{pmatrix} 7 & 5 & 2 \\ 0 & 1 & 2 \\ -3 & -1 & -1 \end{pmatrix}$$

4.5. Вычислить 3 значения функции на заданном отрезке. Вывести значения аргумента и значения функции на экран. Построить 3-D график функции.

1. Функция  $f(x, y) = \frac{\sin x \cos y}{x + \cos x}$ , на отрезке  $x \in [0, \pi], y \in [0, \pi]$
2. Функция  $f(x, y) = x \sin xy + \frac{e^{-x} - e^y}{e^{-y} + e^x}$ , на отрезке  $x \in [0, \pi], y \in [0, 1]$
3. Функция  $f(x, y) = \left(\frac{e^{-x} - e^y}{2}\right)^2 + \left(\frac{e^{-y} + e^x}{2}\right)^2$ , на отрезке  $x, y \in [-3, 3]$
4. Функция  $f(x, y) = 4xy - \cos^2\left(\frac{e^{-y} - e^x}{e^{-x} + e^y}\right)$ , на отрезке  $x, y \in [0, 1]$
5. Функция  $f(x, y) = e^{\frac{y+5}{x^2}} - \sqrt{x+y}$ , на отрезке  $x, y \in [0, 3]$
6. Функция  $f(x, y) = \operatorname{tg}\left(\sqrt{\frac{2x-y}{x+y+1}}\right) + \sqrt{y+x}$ , на отрезке  $x, y \in [0.7, 4]$
7. Функция  $f(x, y) = \frac{x+2y}{1+\sqrt[3]{xy+1}}$ , на отрезке  $x, y \in [10, 100]$
8. Функция  $f(x, y) = \frac{\sin^2(2xy + \frac{\pi}{2})}{\sqrt{\frac{|xy|+1}{4}}}$ , на отрезке  $x, y \in [-\pi/2, \pi/2]$
9. Функция  $f(x, y) = \log_2(\sin x + \sin y + 1)\sqrt{e^{-\cos x}}$ , на отрезке  $x, y \in [0, \pi/2]$
10. Функция  $f(x, y) = |\sin x + \cos y| + x + y$ , на отрезке  $x, y \in [0, \pi/2]$
11. Функция  $f(x, y) = \frac{\operatorname{sh} x + \operatorname{ch} y}{\operatorname{th} xy + 5}$ , на отрезке  $x, y \in [-3, 3]$
12. Функция  $f(x, y) = 4\cos^2\left(\frac{x+y+\frac{\pi}{2}}{xy+2}\right)$ , на отрезке  $x, y \in [0, 1]$
13. Функция  $f(x, y) = x(\sin y + \cos x)\frac{y}{1+\sin^2 y}$ , на отрезке  $x, y \in [0, \pi/2]$
14. Функция  $f(x, y) = 5x^2 - 2y^2 + xy$ , на отрезке  $x, y \in [0, 3]$
15. Функция  $f(x, y) = (\sin x^2 + \cos y^2)^y$ , на отрезке  $x, y \in [-1, 1]$

4.6 Сгенерировать матрицу из  $(N+10) \times (100 \times N)$  одинаково распределенных случайных чисел, где  $N$  – номер студента в списке группы. Построить гистограмму распределения этих чисел. Вычислить среднее, дисперсию и стандартное отклонение, медиану. Варианты распределений случайных чисел:

1. Равновероятное распределение на интервале  $(0,1)$ .
2. Нормальное распределение с мат. ожиданием 1 и дисперсией 2.
3. Равновероятное распределение на интервале  $(-5,5)$ .
4. Нормальное распределение с мат.ожиданием 5 и дисперсией 0.25.
5. Экспоненциальное распределение с параметром  $\lambda=5$ .
6. Экспоненциальное распределение с параметром  $\lambda=10$ .

7. Распределение Пирсона хи-квадрат с 5-ю степенями свободы.
8. Распределение Пирсона хи-квадрат с 20-ю степенями свободы.
9. Распределение Пирсона хи-квадрат с 1, 10, 30-ю степенями свободы.
10. Распределение Пирсона хи-квадрат с 1, 5, 20-ю степенями свободы.
11. Распределение Пуассона с мат. ожиданием 2.
12. Распределение Пуассона с мат. ожиданием 10.
13. Нормальное распределение с мат. ожиданием -5 и дисперсией 5.
14. Равновероятное распределение на интервале (-3,10).
15. Экспоненциальное распределение с параметром лямбда=1.

## 5 Методические указания по выполнению ознакомительной работы

5.1. При выполнении задания 4.3 рекомендуется воспользоваться функцией **plot2d**. Синтаксис функции (параметры в квадратных скобках не обязательны):

```
plot2d([logflag,][x,],y[,style[,strf[,leg[,rect[,nax]]]]])
plot2d([logflag,][x,],y,<opt_args>)
```

Здесь  $x$  –действительная матрица или вектор; если  $x$  отсутствует, то предполагается, что вектор изменяется от 1:n, где  $n$  - число пунктов кривой, определяемых параметром  $y$ ;

$y$  - действительная матрица или вектор;

<opt\_args> - последовательность ключей key1=value1, key2=value2,..., где key1, key2,... могут иметь значения:

logflag - устанавливает масштаб вдоль осей (линейный или логарифмический), возможные значения: "nn", "nl", "ln" и "ll";

style - устанавливает стиль для каждой кривой, целочисленное положительное или отрицательное значение;

strf - управляет отображением заголовка, задается в виде последовательности из трех цифр "xyz" (по умолчанию strf = "081").

Значения этих цифр и остальных ключей можно посмотреть в справке к функции. Ниже приведен пример сценария для построения графиков 3-х синусоидальных функций с использованием plot2d:

```
clf;
x=[0:0.1:2*%pi]';
plot2d(x,[sin(x) sin(2*x) sin(3*x)])
```

5.3 В ходе выполнения задания 4.4. используйте встроенные функции для работы с матрицами, описанные на стр. 9

5.4. При построении 3-D графиков (задание 4.5) рекомендуется изучить возможности пункта меню «Инструменты» графического окна, в частности, вращения изображения и увеличения области, а также возможностей

контекстного меню, которое появляется при нажатии правой кнопки мышки. Используйте эти инструменты для более детального анализа свойств функции и редактирования изображения и надписей.

5.5 При построении гистограмм в соответствии с заданием 4.6 воспользуйтесь расширенными возможностями функции `histplot`, приведенными в примере:

```
d=rand(1,10000,'normal');  
clf; histplot(20,d)  
xgrid  
clf; histplot(20,d,normalization=%f)  
clf; histplot(20,d,leg='rand(1,10000,"normal")',style=5)  
clf; histplot(20,d,leg='rand(1,10000,"normal")',style=16, rect=[-  
3,0,3,0.5]);
```

## **6 Содержание отчета**

- 6.1 Цель работы.
- 6.2 Вариант задания, описание используемых формул.
- 6.3 Листинги программ с комментариями.
- 6.4 Результаты вычислений и графики функций.
- 6.5 Выводы по результатам исследований.

## **7 Контрольные вопросы**

- 7.1 Дайте общую характеристику системы моделирования Scilab и приведите основные отличия от системы Matlab.
- 7.2 Какие типы данных, используются в Scilab и каким образом задаются константы в программах для Scilab?
- 7.3 Как в Scilab определяются пользовательские функции?
- 7.4 Приведите пример задания вектора-строки, вектора столбца и матрицы.
- 7.5 Поясните, как осуществляются операции конкатенации матриц и как они реализуются инструкциями Scilab?
- 7.6 Как математически записываются комплексные числа и какие операции существуют в Scilab для работы с такими числами?
- 7.7 Какие основные статистические функции имеются в Scilab для обработки матриц?
- 7.8 Каким образом можно сохранить данные в файле или ввести их в Scilab из файла?
- 7.9 Какие основные команды Scilab используются для построения двумерных графиков?



- 7.10 Как сделать, чтобы вывод нового графика не стирал предыдущий график?
- 7.11 Как отобразить на графике координатную сетку, подписать рисунок и названия осей координат?
- 7.12 Как построить в Scilab 3-D поверхность?
- 7.13 Напишите сценарий генерирования случайной последовательности с нормальным законом распределения и построения гистограммы распределения.
- 7.14 Как можно разделить графическое окно на отдельные подокна?
- 7.15 Как можно разделить переменные на равномерные интервалы? Приведите пример.
- 7.16 Напишите сценарий генерирования смеси гармонического сигнала с шумом и графического отображения этого процесса.
- 7.17 Напишите сценарий генерации случайных чисел с экспоненциальным распределением и построения гистограммы такого распределения.

### **Список рекомендованной литературы**

1. Бондарев В.Н. Искусственный интеллект: Учеб. пособие для студентов вузов / В. Н. Бондарев, Ф. Г. Аде. — Севастополь: Изд-во СевНТУ, 2002. — 613 с.
2. Ерин С.В. Scilab – примеры и задачи: практическое пособие / С.В. Ерин – М.: Лаборатория «Знания будущего», 2017. – 154 с.
3. Медведев, В.С. Нейронные сети. MATLAB 6 / В.С. Медведев, В.Г. Потемкин; под общ. ред. В.Г. Потемкина. — М.: ДИАЛОГ-МИФИ, 2002. — 496 с.
4. Хайкин С. Нейронные сети: Полный курс. Пер. С англ. / С. Хайкин. — М.: Изд. «Вильямс», 2006. — 1104 с.
5. Hagan M.T. Neural Network Design. The 2nd edition [Электронный ресурс] /М.Т.Нagan, Н.В. Demuth, М.Н. Beale, O.D. Jesus. . — Frisco, Texas, 2014 . — 1012 p. Режим доступа: <https://www.hagan.okstate.edu/NNDesign.pdf>. —Последний доступ: 14.01.2019. —Название с экрана.