Лабораторная работа №1

Численное дифференцирование, простейший анализ функции и построение касательной к графику функции в MS Excel

Цель работы: научиться определять поведение функции на заданном участке, а также, строить касательную к графику функции в заданной точке с помощью MS Excel.

Численное дифференцирование и простейший анализ функции

Из курса математики известно, что формула производной в общем виде выглядит выглядит так:

$$f'(x) = \lim_{\Delta x \to 0} \frac{f(x + \Delta x) - f(x)}{\Delta x}, \tag{1}$$

где x – аргумент функции; Δx – приращение аргумента.

С помощью производной можно определить поведение функии (возрастание или убывание) и критические точки функции — максимумы, минимумы и точки перегиба. Если значение производной функции на некотором отрезке отрицательно, функция убывает; если положительно — возрастает; если равно нулю — имеется критическая точка.

Пример. Функция $f(x) = x^2 + 2x - 3$ задана в интервале $x \in [-5; 5]$. Исследовать поведение этой функции и построить касательную к её графику в точке $x_0 = 1$.

Последовательность действий:

1. Пусть $\Delta x = 0,00001$. В ячейку *A1* нужно ввести строку (без кавычек): «Dx=». Выделить букву «D» и выбрать шрифт *Symbol.* В ячейку B1 ввести число 0,00001.

2. В ячейках с A2 по F2 оформить шапку таблицы, как показано в табл. 1

3. В столбце *A*, начиная с третьей строки, будут содержаться значения x. В ячейки с *A3* по *A13* ввести значения от –5 до 5.

4. В ячейке *B3* записать формулу «=*A3*^2+2**A3*-3» и «растянуть» её до конечного значения x, т.е. до 13-ой строки.

5. Чтобы вычислить значения производной функции на заданном отрезке, необходимо сделать промежуточные вычисления. В ячейку *C3* ввести формулу суммы аргумента x и его приращения Δx. Формула будет иметь вид: «=*A3*+\$*B*\$*1*».

6. В ячейку *D3* записать формулу «=*C3*^2+2**C3*-3», по которой вычисляется значение функции f от $x + \Delta x$. «Растянуть» получившееся значение до

конечного значения аргумента.

7. В ячейку *E3* записать формулу производной (1), учитывая, что значения $f(x)$ находятся в ячейке *B3*, а значения $f(x+\Delta x)$ – в *D3*. Формула будет иметь вид: «=(*D3*-*B3*)/$*B*$*1*».

8. Определить поведение функции на заданном промежутке (возрастает, убывает или имеется критическая точка). Для этого в ячейку *F3* нужно записать формулу для определения поведения функции. Формула содержит три условия:

- если $f'(x) < 0$ – функция убывает;
- если $f'(x) > 0$ – функция возрастает;
- если $f'(x) = 0$ – имеется критическая точка[1]

и имеет вид: «=ЕСЛИ(F3<0;"убыв.";ЕСЛИ(F3=0;"к.т.";"возр."))»

Вычисление значений для построения касательной к графику в заданной точке

Уравнение касательной имеет вид:

$$y = f(x_0) + f'(x_0)(x - x_0) \tag{2}$$

Пример. Построить касательную к графику функции $y = x^2 + 2x - 3$, заданной в интервале $x \in [-5; 5]$, в точке $x_0 = 1$.

Последовательность действий

Для того, чтобы построить касательную к графику в заданной точке, нужно определиться с адресами ячеек, содержащих соответствующие значения.

Поскольку значение $x_0 = 1$ находится в ячейке *A9* (табл. 1), то значение $f(x_0)$ будет находиться ячейка *B9*, значение $f'(x)$ – в ячейка *E9*. Причём, в формуле должны использоваться абсолютные ссылки на эти ячейки. Значениям x будет соответствовать ячейка *A3*.

Таким образом, в соответствии с уравнением (2), в ячейке *G3* будет записана формула «=$*A*$*9*+$*E*$*9**(*A3*-$*A*$*9*)» (без кавычек).

Построение касательной к графику в заданной точке

Для того, чтобы построить графики функции, производной и касательной на одной координатной плоскости, нужно выделить три диапазона ячеек: *A2*:*B13*, *E2*:*E13* и *G2*:*G3*.

[1] Из-за слишком большой погрешности ввычислений, значение $f'(x)$ может быть неравным нулю.

Таблица 1: Таблица исходных данных

	A	B	C	D	E	F	G
1	$\Delta x =$	0,00001					
2	x	$f(x)$	$x + \Delta x$	$f(x + \Delta x)$	$f'(x)$	повед.	касат
3	-5	12	-4,9999	11,9992	-7,9999	убыв.	
4	-4	5	-3,9999	4,9994	-5,9999	убыв.	
5	-3	0	-2,9999	-0,0004	-3,9999	убыв.	
6	-2	-3	-1,9999	-3,0002	-1,9999	убыв.	
7	-1	-4	-0,9999	-4	1E-04	возр.	
8	0	-3	0,0001	-2,9998	2,0001	возр.	
9	1	0	1,0001	0,0004	4,0001	возр.	
10	2	5	2,0001	5,0006	6,0001	возр.	
11	3	12	3,0001	12,0008	8,0001	возр.	
12	4	21	4,0001	21,001	10,0001	возр.	
13	5	32	5,0001	32,0012	12,0001	возр.	

Для этого нужно, установив курсор в ячейку A2 и нажав левую кнопку мыши, выделить значения в столбце A, а затем, не отпуская кнопку мыши, выделить значения в столбце B. После этого, нажав и удерживая клавишу Ctrl на клавиатуре, выделить значения в столбцах E и G.

По значениям выделенных ячеек строится точечный график. Его примерный вид приведён на рис. 1.

Задания для самостоятельного выполнения

Построить графики функций, производных и касательных для следующих уравнений:

1. $f(x) = \frac{x^4}{4} - 2x^2 - \frac{9}{4}$, $x \in [-4; 4]$, $x_o = 1$

2. $f(x) = \frac{4}{x^2 - 2x + 2}$, $x \in [-5; 5]$, $x_0 = -3$

3. $f(x) = x^3 - 3x^2 + 2$, $x \in [-2; 4]$, $x_0 = 0$

4. $f(x) = x - \frac{4}{x^2 + 7}$, $x \in [-2; 3]$, $x_0 = 1$

Задания для самостоятельного изучения

1. Основные понятия: рабочий лист, ячейка.
2. Типы данных. Форматирование ячеек.

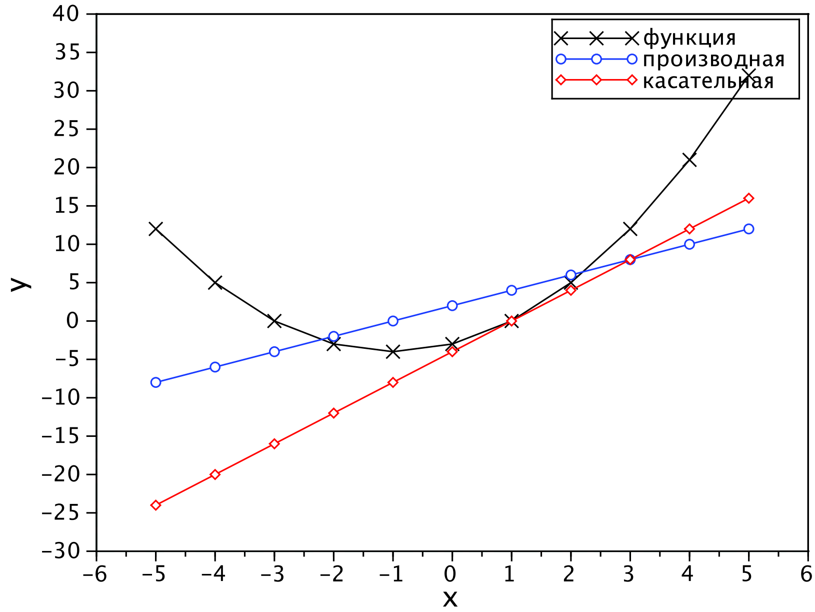

Рис. 1: Графики функции, производной и касательной

3. Ввод формул в ячейки. Абсолютные и относительные ссылки.
4. Мастер функций. Графические возможности Excel.

Вопросы для самоконтроля

1. Раскрыть понятие электронной таблицы.
2. Табличный процессор и электронная таблица — это одно и то же?
3. Раскрыть понятие рабочего листа и ячейки. Как называется совокупность рабочих листов?
4. С данными каких типов может работать Excel?
5. Раскрыть понятие относительной и абсолютной ссылок. В чём их различия?
6. Что такое Мастер функций? Каково его назначение?
7. Какие типы графиков поддерживаются в Excel? Чем отличается диаграмма типа График от Точечной диаграммы?

Лабораторная работа №2

Начало работы в Scilab. Простейшие вычисления

Цель работы: ознакомиться с базовыми возможностями программы Scilab.

Scilab — пакет прикладных математических программ, предоставляющий мощное открытое окружение для инженерных (технических) и научных расчётов.

Scilab содержит сотни математических функций, и есть возможность добавления новых, написанных на различных языках (C, C++, Fortran и т. д.). Также имеются разнообразные структуры данных (списки, полиномы, рациональные функции, линейные системы), интерпретатор и язык высокого уровня.

В системе доступно множество инструментов:

- двумерные и трёхмерные графики, анимация;
- линейная алгебра;
- разреженные матрицы (sparse matrices);
- полиномиальные и рациональные функции;
- интерполяция, аппроксимация;
- симуляция: решение ОДУ и ДУ.
- Xcos — гибрид системы моделирования динамических систем и симуляции;
- дифференциальные и не дифференциальные оптимизации;
- обработка сигналов;
- параллельная работа;
- статистика;
- работа с компьютерной алгеброй;
- интерфейс к Fortran, Tcl/Tk, C, C++, Java, LabVIEW.

Scilab имеет схожий с MATLAB язык программирования. Некоторые наиболее часто употребляемые математические опреции приведены в табл. 2. В табл. приведены некоторые математические константы. Также Scilab позволяет работать с элементарными (табл. 4) и большим числом специальных функций

(Бесселя, Неймана, интегральные функции), имеет мощные средства работы с матрицами, полиномами (в том числе и символьно), производить численные вычисления (например, численное интегрирование) и решение задач линейной алгебры, оптимизации и симуляции, мощные статистические функции, а также средство для построения и работы с графиками.

Таблица 2: Некоторые наиболее часто употребляемые математические опреции

Оператор	Действие
`a+b`	сложение
`a-b`	вычитание
`a*b`	умножение
`a/b`	деление
`a^b`	возведение в степень
`m'`	транспонирование матрицы или вектора

Таблица 3: Некоторые математические константы

Константа	Описание
%pi	Число π (3,14159265...)
%e	Число e (2,71828...)
%i	Мнимая единица в записи комплексного числа
%j	

В состав пакета также входит XCos — инструмент для редактирования блочных диаграмм и симуляции (аналог Simulink в пакете MATLAB).

Для хранения исходных данных и промежуточных результатов используются переменные.

Пример. Даны два числа: $a = 33$, $b = 20$. Найти значения суммы и разности этих чисел и присвоить их переменным S и R соответственно.

Решение задачи будет выглядеть так:

```
a=33
b=20
S=a+b
R=a-b
```

Таблица 4: Некоторые наиболее часто употребляемые функции

Имя функции	Значение
`sin(x)`	синус x
`cos(x)`	косинус x
`tan(x)`	тангенс x
`cot(x)`	котангенс x
`asin(x)`	арксинус x
`acos(x)`	арккосинус x
`atan(x)`	арктангенс x
`acot(x)`	арккотангенс x
`abs(x)`	модуль числа (абсолютная величина) числа x
`sqrt(x)`	квадратный корень числа x
exp(x)	эквивалентно e^x

Задания для выполнения лабораторной работы

Все неизвестные величины задать самостоятельно.

0. Длина отрезка составляет p миль (1 миля=1,609 км). Перевести её в километры и метры.

1. Даны значения зарядов q_1, q_2, q_3 и расстояние между зарядами r_1 и r_2. Найти силу F_3, если известно, что $F_1 = k\frac{|q_1||q_3|}{r_1^2}$, $F_2 = k\frac{|q_2||q_3|}{r_2^2}$, $F_3 = \sqrt{F_1^2 + F_2^2}$, $k = 9 \cdot 10^9$.

2. Дан прямоугольный треугольник с катетами a и b. Найти гипотинузу треугольника c.

3. Известен радиус окружности R. Найти длину окружности C и площадь круга S.

4. Известны стороны прямоугольника a и b. Найти площадь прямоугольника S, периметр P и длину диаганали L.

5. Известен радиус сферы R и высота шарового сегмента Н. Найти площадь поверхности сферы S, объём шара V_1, и объём шарового сегмента V_2. Формула для расчёта: $S = 4\pi R^2$, $V_1 = \frac{4}{3}\pi R^3$, $V_2 = \pi H^2 \left(R - \frac{1}{3}H\right)$.

6. Известны радиус основания конуса R и его высота H. Найти объём конуса $V = \frac{1}{3}(\pi R^2 H)$.

7. Известен радиус цилиндра R и высота H. Найти объём цилиндра V, площадь его боковой поыерхности S, и полную площадь P. Формулы для расчёта: $V = \pi R^2 H$, $S = 2\pi RH$, $P = 2\pi R(H + R)$.

8. Известны стороны прямоугольного параллелепипеда a, b и c. Найти его объём V, площадь боковой поверхности S и полную площадь поверхности P.

9. Длина отрезка l задана в дюймах. Выразить значение длины в метрах,

сантиметрах и миллиметрах, если 1 дюйм = 2,54 см.

Задания для самостоятельного изучения

1. Понятие численных вычислений.
2. Типы данных в программе Scilab. Запись чисел.
3. Понятие переменной.
4. Математические операции и функции. Функции `sqrt` и `abs`.

Вопросы для самоконтроля

1. Какие Вы знаете математические операции и функции?
2. Как в Scilab найти квадратный корень числа?
3. Какая функция используется для нахождения абсолютной величины?

Лабораторная работа №3

Работа с векторами и матрицами

Цель работы: научиться работать с векторами и матрицами в Scilab.

Основной тип данных в Scilab — матрица. Даже простые числа рассматриваются программой как матрицы, состоящие из одной строки и одного столбца. Частным случаем матриц являются векторы, которые рассматриваются программой как матрицы, состоящие из одной строки и нескольких столбцов или одного столбца и нескольких строк. Таким образом, можно выделить два типа векторов: вектор-строка и вектор-столбец.

Векторы в Scilab задаются в виде совокупности чисел, окружённой квадратными скобками. Элементы вектора-строки отделяются друг от друга запятыми или пробелами. В векторе-столбце элементы отделяются друг от друга символом «точка с запятой»:

```
a=[2 1 -2 4 3]; // это вектор-строка
b=[3, 2, 5, -1, 0]; // и это вектор-строка
c=[3; 2; 4; 5; -2; 1]; // а это вектор-столбец
```

Важно заметить, что вектор-строка и вектор-столбец в понимании Scilab – не одно и то же.

Кроме того, Scilab позволяет создавать векторы, содержащие последовательности чисел. Операторы и функции для создания последовательностей приведеды в табл. 5.

Таблица 5: Операторы и функции для создания векторов – последовательностей чисел.

Операторы и функции	Значение
`m:n`	оператор создаёт последовательность чисел от m до n с шагом 1
`m:k:n`	оператор создаёт последовательность чисел от m до n с шагом k
`linspace(m, n)`	функция создаёт последовательность чисел от m до n с количеством элементов в последовательности – 100
`linspace(m, n, t)`	функция создаёт последовательность чисел от m до n с количеством элементов в последовательности – t

Матрицы, как и векторы, задаются в виде совокупности чисел, окружённой квадратными скобками. Элементы в строках отделяются друг от друга запятыми или пробелами. Строки отделяются друг от друга символами «точка с запятой»:

```
m=[2, 3, -2; 4, 0, 2; 1, 4, 3]; // Элементы в строках разделены
d=[2 1 3; 4 2 3; 0 0 1]; // Элементы в строках разделены пробела
```

Таким образом, матрицу можно рассматривать как вектор-строку, элементоми которого являюся векторы-столбцы.

Все математические функции, приведённые в табл. 4 могут применяться как при работе с числами, так и при работе с матрицами. Например, если в качестве аргумента функции `sin(x)` передать матрицу, то в качестве результата функция вернёт матрицу того же размера, что и исходная, но её элементами будут значения синусов соответствующих элементов исходной матрицы.

Scilab имеются специальные функции для работы с матрицами (табл. 6).

Таблица 6: Некоторые функции для работы с матрицами

Функция	Действие
`inv(M)`	Вычисление обратной матрицы
`det(M)`	Вычисление определителя матрицы
`size(M)`	Возвращает два числа – количество строк и количество столбцов

Для того, чтобы узнать длину вектора, можно воспользоваться как функцией `size(v)`, так и функцией `length(v)`.

Математические операции умножение, деления и возведения в степень, приведённые в табл. 2 являются матричными. Например, результатом оператора умножения «*» будет матричное произведение. Другими словами, две матрицы будут перемножены по правилам матричного произведения.

Кроме матричных, в Scilab есть и поэлементные математические операции. Как следует из названия, эти операции применяются к соответствующим элементам двух матриц или векторов. К поэлементным операциям относятся операции сложения и вычитания (табл. 2), а также операции, приведённые в табл. 7.

Таблица 7: Поэлементные математические операции

Оператор	Действие
`a.*b`	Поэлементное умножение
`a./b`	Поэлементное деление
`a.^b`	Поэлементное возведение в степень

Умножение и деление вектора или матрицы на число выполняется с помощью матричных операций. Также, допускается сложение матрицы и числа или вычитание числа из матрицы или матрицы из числа.

Задания для выполнения лабораторной работы

Выполнить следующие вычисления:

1. $a = 15$, $b = (3, 2, 4, -2, 0)$. Вычислить $a \cdot b$
2. $a = (2, 3, 1, 4)$, $b = \begin{pmatrix} -1 \\ 2 \\ 0 \end{pmatrix}$. Вычислить $a \cdot b$
3. $a = \begin{pmatrix} 2 & -3 & 0 \\ 3 & 2 & 0 \\ 1 & 3 & -1 \end{pmatrix}$, $b = \begin{pmatrix} 2 & 3 & 2 \\ 1 & -2 & 3 \\ 3 & 0 & 1 \end{pmatrix}$. Вычислить поэлементное произведение a и b
4. $a = (2, 3, 1)$, $b = 1$. Вычислить $a - b$ и $b - a$.

Задания для самостоятельного изучения

1. Понятие вектора и матрицы.

2. Представление чисел в Scilab.

3. Операции и функции при работе с матрицами и векторами: поэлементные и матричные операции. Обращение матриц, транспонирование, определитель.

Вопросы для самоконтроля

1. Раскрыть понятие вектора и матрицы.
2. Как в Scilab записываются векторы и матрицы?
3. Какие типы векторов поддерживаются в Scilab?
4. Какие Вы знаете матричные и поэлементные операции?
5. Чем отличаются матричные операции от поэлементных?
6. Какие функции используются для транспонирования матрицы, обращения матрицы и нахождения определителя? Применима ли операция транспонирования для векторов?

Лабораторная работа №4

Решение систем линейных уравнений

Цель работы: научиться решать системы линейных уравнений в программе Scilab.

Система m линейных алгебраических уравнений с n неизвестными (СЛАУ) в линейной алгебре — это система уравнений вида:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + ... + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + ... + a_{2n}x_n = b_2 \\ ... \\ a_{m1}x_1 + a_{m2}x_2 + ... + a_{mn}x_n = b_m \end{cases}$$

где m – количество уравнений; n – количество неизвестных; $x_1, x_2, \ldots, x_n$ – неизвестные, которые надо определить; $a_{11}, a_{12}, \ldots, a_{mn}$ – коэффициенты системы; $b_1, b_2, \ldots\ b_m$ – свободные члены.

Система называется однородной, если все её свободные члены равны нулю Иначе — неоднородной.

Система называется квадратной, если число m уравнений равно числу n неизвестных.

Решение системы — это совокупность n чисел $c_1, c_2, \ldots, c_n$, таких что подстановка каждого c_i вместо x_i в систему обращает все её уравнения в тождества.

Система называется совместной, если она имеет хотя бы одно решение, и несовместной, если у неё нет ни одного решения.

Система линейных уравнений может быть представлена в матричной форме как:

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

или

$$Ax = b$$

где A – матрица системы; x – столбец неизвестных; b – столбец свободных членов.

Методы решения систем линейных уравнений

Метод Крамера (правило Крамера) — способ решения квадратных систем линейных алгебраических уравнений с ненулевым определителем основной матрицы (причём для таких уравнений решение существует и единственно). Назван по имени Габриэля Крамера (1704–1752), придумавшего метод.

Для системы линейных уравнений с неизвестными с определителем матрицы системы Δ, отличным от нуля, решение записывается в виде

$$x_i = \frac{1}{\Delta} \begin{bmatrix} a_{11} & \cdots & a_{1,i-1} & b_1 & a_{1,i+1} & \cdots & a_{1n} \\ a_{21} & \cdots & a_{2,i-1} & b_2 & a_{2,i+1} & \cdots & a_{2n} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{n-1,1} & \cdots & a_{n-1,i-1} & b_{n-1} & a_{n-1,i+1} & \cdots & a_{n-1,n} \\ a_{n1} & \cdots & a_{n,i-1} & b_n & a_{n,i+1} & \cdots & a_{nn} \end{bmatrix}$$

(i-ый столбец матрицы системы заменяется столбцом свободных членов).

В другой форме правило Крамера формулируется так: для любых коэффициентов c_1, c_2, …, c_n справедливо равенство:

$$(c_1x_1 + c_2x_2 + \ldots + c_nx_n) \cdot \Delta = - \begin{bmatrix} a_{11} & a_{12} & \ldots & a_{1n} & b_1 \\ a_{21} & a_{22} & \ldots & a_{2n} & b_2 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ a_{n1} & a_{n2} & \ldots & a_{nn} & b_n \\ c_1 & c_2 & \ldots & c_n & 0 \end{bmatrix}$$

Пример. Дана система линейных уравнений:

$$\begin{cases} 2x_1 + 5x_2 + 4x_3 = 30 \\ x_1 + 3x_2 + 2x_3 = 150 \\ 2x_1 + 10x_2 + 9x_3 = 110 \end{cases}$$

Определители:

$$\Delta = \begin{bmatrix} 2 & 5 & 4 \\ 1 & 3 & 2 \\ 2 & 10 & 9 \end{bmatrix}, \Delta_1 = \begin{bmatrix} 30 & 5 & 4 \\ 150 & 3 & 2 \\ 110 & 10 & 9 \end{bmatrix},$$

$$\Delta_2 = \begin{bmatrix} 2 & 30 & 4 \\ 1 & 150 & 2 \\ 2 & 110 & 9 \end{bmatrix}, \Delta_3 = \begin{bmatrix} 2 & 5 & 30 \\ 1 & 3 & 150 \\ 2 & 10 & 110 \end{bmatrix}$$

Решение:

$$x_1 = \frac{\Delta_1}{\Delta}, x_2 = \frac{\Delta_2}{\Delta}, x_3 = \frac{\Delta_3}{\Delta}$$

Поскольку в программе Scilab имеется встроенная функция det для нахождения определителя матрицы, решение задачи будет выглядеть так:

```
d=det([2 5 4; 1 3 2; 2 10 9])
d1=det([30 5 4; 150 3 2; 110 10 9])
d2=det([2 30 4; 1 150 2; 2 110 9])
d3=det([2 5 30; 1 3 150; 2 10 110])
x1=d1/d
x2=d2/d
x3=d3/d
```

Матричный метод решения (метод решения через обратную матрицу) систем линейных алгебраических уравнений с ненулевым определителем состоит в следующем. Пусть дана система линейных уравнений с неизвестными:

$$\begin{cases} a_{11}x_1 + \ldots + a_{1n}x_n = b_1 \\ \ldots \\ a_{n1}x_1 + \ldots + a_{nn}x_n = b_n \end{cases}$$

Тогда её можно переписать в матричной форме: $Ax = b$, где A – основная матрица системы, b и x – столбцы свободных членов и решений системы соответственно:

$$A = \begin{pmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \ldots & a_{nn} \end{pmatrix}, b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}, x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

Умножим это матричное уравнение слева на A^{-1} – матрицу, обратную к матрице A: $A^{-1}(Ax) = A^{-1}b$. Так как $A^{-1}A = E$ (E – единичная матрица), получаем

$$x = A^{-1}b. \tag{3}$$

Правая часть этого уравнения даст столбец решений исходной системы. Условием применимости данного метода (как и вообще существования решения неоднородной системы линейных уравнений с числом уравнений, равным числу неизвестных) является невырожденность матрицы A. Необходимым и достаточным условием этого является неравенство нулю определителя матрицы A ($\det A \neq 0$).

С помощью функции `inv` можно найти матрицу, обратную к заданной, а затем и корни системы линейных уравнений по формуле (3).

Пример. Дана система линейных уравнений:

$$\begin{cases} 3x_1 + 2x_2 - x_3 = 4 \\ 2x_1 - x_2 + 5x_3 = 23 \\ x_1 + 7x_2 - x_3 = 5 \end{cases}$$

Найти корни этой системы.

Решение задачи:

```
// задаётся матрица
A=[3 2 -1; 2 -1 5; 1 7 -1]
// задаётся вектор-столбец свободных членов
b=[4; 23; 5]
// нужно проверить, не вырожденная ли система.
// определитель не должен быть равен нулю
det(A)
// решение системы уравнений
x=inv(A)*b
// проверка
A(1,1)*x(1)+A(1,2)*x2+A(1,3)*x(3)
// в результате на экран будет выдано число 4.
// Это означает, что корни системы найдены верно
```

Метод решения системы линейных уравнений с помощью операции левого матричного деления. Операция левого матричного деления обозначается знаком «\» и является решением для систем уравнений вида $Ax = b$. Если A – квадратная и невырожденная, то `x=A\b` эквивалентно `x=inv(A)*b`.

Задания для выполнения лабораторной работы

Для всех систем выполнить проверку

1. Решить систему линейных уравнений методом Крамера:

$$\begin{cases} x_1 + 2x_2 - x_3 = 0 \\ 2x_1 - 3x_2 + x_3 = 3 \\ x_1 + x_2 - 3x_3 = 1 \end{cases}$$

2. Решить систему линейных уравнений с помощью обратной матрицы:

$$\begin{cases} 2x_1 + 3x_2 - 5x_3 + x_4 = 10 \\ 5x_1 - 2x_2 + 4x_3 - 2x_4 = 5 \\ x_1 + 3x_2 - x_3 + 2x_4 = 7 \\ 3x_1 - x_2 + 2x_3 - x_4 = 3 \end{cases}$$

3. Решить систему линейных уравнений методом левого деления:

$$\begin{cases} x_1 + 2x_2 - x_3 + x_5 = 3 \\ x_2 - x_1 + 2x_3 + x_4 - 5x_5 = -4 \\ x_3 + x_5 = 7 \\ x_1 - x_2 + x_4 = 3 \\ 2x_1 - 3x_4 + x_2 - x_3 = 4 \end{cases}$$

Задания для самостоятельного изучения

Системы линейных алгебраических уравнений. Методы их решения

Вопросы для самоконтроля

1. Какие методы решения систем линейных алгебраических уравнений вы знаете?

2. Для чего применяется операция левого матричного деления? Чему она эквивалентна?

Лабораторная работа №5

Построение двумерных графиков

<u>Цель работы</u>: научиться строить графики в программе Scilab.

Для построения двумерных графиков применятеся функция plot, которая принимает два или более аргументов. Первый аргумент — вектор значений по оси абсцисс. Второй аргумент — вектор значений по оси ординат. Третий аргумент — форматная строка, в которой указывается цвет линии (табл. 8), тип линии (табл. 9) и тип маркеров (табл. 10). Для задания векторов значений можно применять операторы и функции из табл. 5.

Таблица 8: Цвета линий графиков

Обозначение	Цвет
r	красный
g	зелёный
b	синий
c	ярко-голубой
m	ярко-розовый
y	жёлтый
k	чёрный

Таблица 9: Типы линий графиков

обозначение	тип линии
-	сплошная
--	штриховая
.-	штрих-пунктирная
:	точечная

Таблица 10: Типы маркеров

Обозначение	Тип маркера
+	в виде знака «+»
*	звёздочки
буква «o»	кружки
. (точка)	закрашенный кружки
буква «x»	крестики
буква «s»	квадраты
буква «d»	ромбы

Пример. Построить график функции $y = 2x^2 + 4x - 5$, при $x \in [-5; 5]$.

```
x=-5:5;
```

```
y=2*x^2+4*x-5;
plot(x,y,'k-o')
```

В результате будет построен график, приведённый на рис. 2, на котором будет изображена сплошная линия чёрного цвета с маркерами в виде кружков.

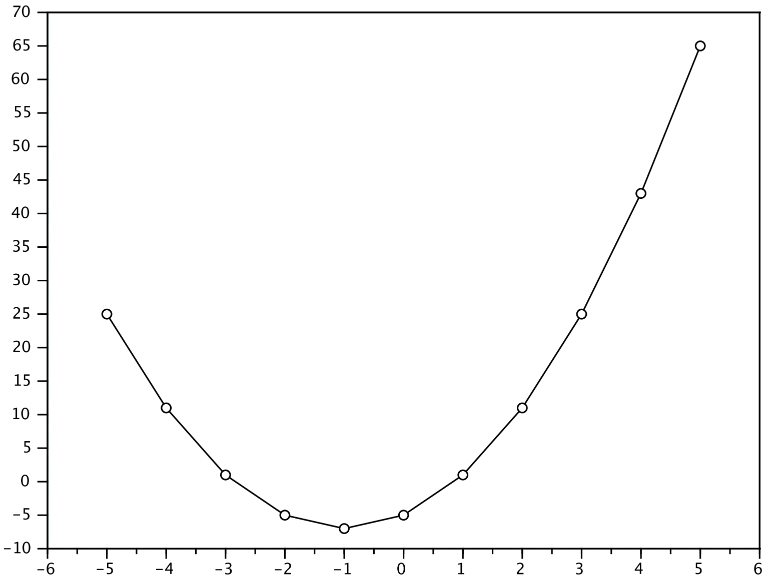

Рис. 2: График функции $y = 2x^2 + 4x - 5$

Чтобы на графике отобразить координатную сетку, можно использовать функцию `xgrid`.

В Scilab есть функция `legend` для отображения на графике легенды. Функция принимает количество аргументов, равное количеству линий на графике. Все аргументы строковые (т. е., записываются в кавычках). Также, функция `legend` может принимать дополнительный строковый аргумент `pos`, который задаёт положение легенды на графике (табл. 11).

Пример. Построить график функции $y = x^3 - 4x^2 + 2x - 10$, при $x \in [-5; 5]$. Отобразить координатную сетку добавить легенду.

```
x=-5:5;
y=x^3-4*x^2+2*x-10;
plot(x,y)
xgrid
legend("$y=x^3-4x^2+2x-10$",pos='in_upper_left');
```

Грфик приведён на рис. 3.

Таблица 11: Возможные значения аргумента `pos` функции `legend`

Обозначени	Значение
"in_upper_right"	в правом верхнем углу внутри графика
"in_upper_left"	в левом верхнем углу внутри графика
"in_lower_left"	в левом нижнем углу внутри графика
"in_lower_right"	в правом нижнем углу внутри графика
"out_upper_right"	в правом верхнем углу вне графика
"out_upper_left"	в левом верхнем углу вне графика
"out_lower_left"	в левом нижнем углу вне графика
"out_lower_right"	в правом нижнем углу вне графика
"upper_caption"	над верхним левым углом графика
"lower_caption"	под нижним левым углом графика

Если в функции `legend` строку разместить между знаками «$...$», то на графике эта строка будет отображаться в полиграфическом качестве. В табл. 12 приведены некоторые конструкции для форматирования формул в полиграфическом качестве.

Таблица 12: Некоторые знаки для форматирования формул

Конструкция	Значение
a^b	Возведение в степень: a^b
a_b	Подстрочный индекс: a_b
\frac{a}{b}	Дробь: $\frac{a}{b}$
\sqrt{a}	Квадратный корень: $\sqrt{a}$
\int_a^b	Интеграл: $\int_a^b$
\sum_a^b	Сумма: $\sum_a^b$
\pi	Число π

Задания для выполнения лабораторной работы

Построить графики функций

1. $y = \frac{x^4}{4} - 2x^2 - \frac{9}{4}$, $x \in [-4; 4]$
2. $y = \frac{4}{x^2-2x+2}$, $x \in [-5; 5]$
3. $y = x^3 - 3x^2 + 2$, $x \in [-2; 4]$

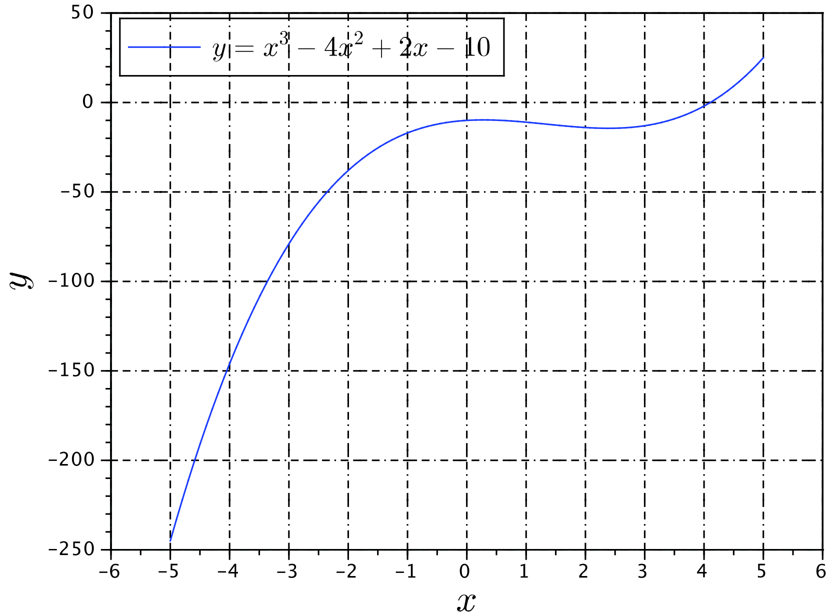

Рис. 3: График функции $y = x^3 - 4x^2 + 2x - 10$

4. $y = \frac{x-4}{x^2+7}$, $x \in [-2; 3]$

Задания для самостоятельного изучения

1. Графические возможности программы Scilab.
2. Плоские и объёмные графики.
3. Графики в декартовых и полярных системах координат.

Вопросы для самоконтроля

1. Какая функция используется для построения двумерного графика в прямоугольных координатах? Какие аргументы она принимает?

2. Как добавить легенду на график?

3. Как изменить параметры графика?

4. Что такое форматная строка? Из чего она состоит?

Лабораторная работа №6

Построение поверхностей

Цель работы: научиться строить поверхности в программе Scilab.

Для построения поверхностей в Scilab используются функции `mesh` и `surf`. Отличие их состоит в том, что функция `mesh` строит поверхность в виде сетки. Функция `surf` строит цветную поверхность.

Обе функции могут принимать по одному или по три аргумента. В первом случае аргументом является матрица, по значениям которой будет построена поверхность. При этом, подписи к координатным осям будут сделаны автоматически. Во втором случае, первый и второй аргументы — векторы, значения которых будут использоваться в качестве подписей к координатным осям. Третий аргумент — матрица, по значениям которой будет построена поверхность.

Для формирования матриц исходных данных можно применять функцию `meshgrid`, которая принимает один или два аргумента и возвращает две матрицы со значениями x и y. В качестве аргументов функция принимает векторы, содержащие последовательности чисел. Если функция вызывается с одним аргументом – вектороим с размерностью a, то в результате будут созданы две квадратные матрицы, размерностью $a \times a$. Если же функция вызывается с двумя аргументами – векторами, имеющими размерность a и b, то в результате будут созданы две матрицы с размерностями $a \times b$.

Пример. Построить поверхность для функции $z = x^3 - y^2$, при $x \in [-5; 5]$, $y \in [-4; 6]$.

```
x=linspace(-5,5,50);
y=linspace(-4,6,50);
[xm,ym]=meshgrid(x,y);
zm=xm.^3-ym.^2; // используются поэлементные операции!
surf(x,y,zm)
```

Полученная поверхность приведена на рис. 4.

Важно! При вычислении значений для построения поверхности должны использоваться поэлементные операции умножения («.*»), деления («./») и возведения в степень («.^»).

Здания для выполнения лабораторной работы

Построить поверхности для следующих функций

1. $z = 2x + y^2$, $x \in [-5; 5]$, $y \in [-5; 5]$

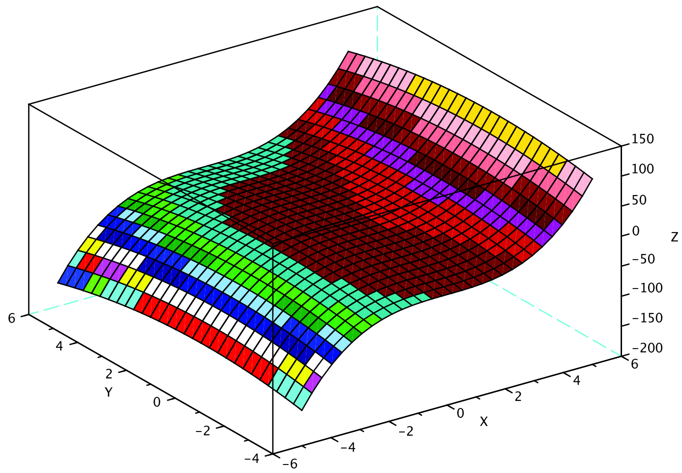

Рис. 4: Поверхность функции $z = x^3 - y^2$

2. $z = x^2 + 0,5y^2$, $x \in [-5; 6]$, $y \in [-3; 7]$

3. $z = x^{1,3} - y^{0,3}$, $x \in [-5; 5]$, $y \in [-4; 6]$

Вопросы для самоконтроля

Какие функции используются для построения поверхностей? Какие аргументы они принимают?

Лабораторная работа №7

Построение плоских и объёных графиков в полярных координатах

Цель работы: научиться строить плоские и объёмные полярные графики в программе Scilab.

Полярная система координат — двухмерная система координат, в которой каждая точка на плоскости определяется двумя числами — полярным углом и полярным радиусом. Полярная система координат особенно полезна в случаях, когда отношения между точками проще изобразить в виде радиусов и углов; в более распространённой, декартовой или прямоугольной системе координат, такие отношения можно установить только путём применения тригонометрических уравнений.

Для построения графиков в полярных координатах в Scilab применяются следующие функции:

- `polarplot` – функция для построения плоского полярного графика;
- `comet3d` – функция для построения графиков в пространстве, в том числе и полярных.

Пример. Построить график функции $\rho = \phi \cdot sin\ \phi$, при $\phi \in [0; 2\pi]$.

```
phi=linspace(0, 2*%pi);
rho=phi.*sin(phi);
polarplot(phi,rho)
```

Полученный график представлен на рис. 5.

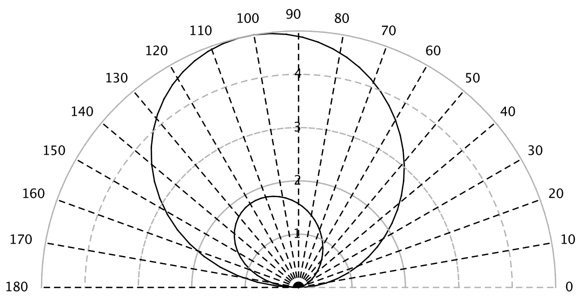

Рис. 5: Построить график функции $\rho = \phi \cdot sin\ \phi$ в полярных коодинатах

Для построения объёмных графиков в полярных координатах можно использовать функцию `comet3d`, которая принимает три аргумента: первый и второй – координаты по осям x и y, третий – координата по оси z.

Пример. Построить объёмный график в виде спирали.

Для построения этого графика можно воспользоваться параметрическими уравнениями, которые будут описывать положение точек в пространстве посредством тригонометрических функций `sin` и `cos`. Функция `cos` будет описывать изменение графика по оси x, а функция `sin` – по оси y.

```
t=0,2*$pi;
n=2; // количество витков спирали
x=cos(t);
y=sin(t);
comet3d(x*n,y*n,t)
```

График представлен на рис. 6.

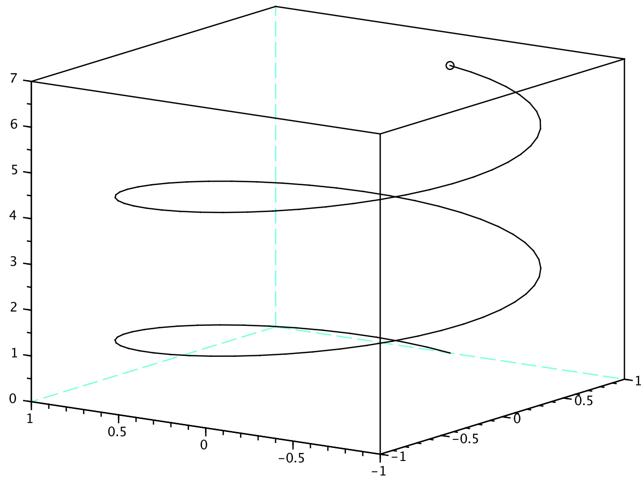

Рис. 6: Объёмный график в полярных координатах

Здания для выполнения лабораторной работы

Построить плоские графики функций в полярных координатах

1. $\rho = 2\sin 4\phi$, $\phi \in [0; 2\pi]$

2. $\rho = -\phi^2$, $\phi \in [-\pi; \pi]$

3. $\rho = \cos(\cos x)$, $\phi \in [-\pi; \pi]$

4. $\rho = \sin 4x + 2\cos 3x$, $\phi \in [0; 2\pi]$

Вопросы для самоконтроля

1. В чём отличие прямоугольной системы координат от полярной?
2. Какие типы графиков и диаграмм поддерживаются в Scilab?
3. Какая функция применяется для построения полярного графика? Какие аргументы она принимает?

Лабораторная работа №8

Нахождение корней многочленов от одной переменной

Цель работы: научиться находить корни многочленов от одной переменной в программе Scilab.

Многочлен (или полином) от одной переменной — это конечная формальная сумма вида:

$$c_0 + c_1x^1 + \ldots + x_nx^n$$

где c_i – фиксированные коэффициенты, а x – переменная.

В программе Scilab есть функция для определения многочлена — `poly`, которая принимает два или три аргумента:

- вектор коэффициентов при неизвестных или корней многочлена (зависит от третьего аргумента);
- имя неизвестного — строковый параметр (записывается в кавычках);
- тип первого аргумента. Необязательный строковый параметр. Может иметь одно из двух значений:
  - ’r’ или ’roots’, если многочлен задан корнями. В этом случае первый аргумент функции `poly` представляет собой вектор корней полинома.
  - ’c’ или ’coeff’, если многочлен задан коэффициентами при неизвестных. В этом случае первый аргумент функции `poly` представляет собой вектор коэффициентов при неизвестных.

Третий аргумент не является обязательным и может отсутствовать. В этом случае подразумевается, что многочлен задан корнями (т. е. первый аргумент функции `poly` представляет собой вектор корней полинома).

Пример. Задать многочлен $x^2 + 2x - 3$. Для того, чтобы воспользоваться функцией `poly`, нужно создать вектор-строку коэффициентов при неизвестном. Причём, коэффициенты должны быть отсортированы в порядке возрастания степеней неизвестного и начинаться от степени 0. Таким образом, указанный многочлен можно переписать так: $-3x^0 + 2x^1 + 1x^2$. Следовательно, вектор коэффициентов при неизвестных будет иметь вид: `[-3 2 1]`, а многочлен в Scilab будет иметь вид:

```
v=[-3 2 1]
p=poly(v, 'x', 'c')
```

Над многочленами в Scilab можно выполнять различные математические действия: сложение и вычитание двух многочленов, умножение или деление многочлена на число, произведение двух многочленов.

```
b=p-1 // вычитание числа из многочлена
s=p+b // сложение двух многочленов
p=p*b // произведение двух многочленов
c=p*2 // умножение многочлена на число
```

Для нахождения корней многочленов используется функция `roots`, принимающая в качестве аргумента многочлен.

```
x=roots(p)
```

Задания для выполнения лабораторной работы

Найти корни многочленов. Проверить найденные корни графически

1. $x^3 - 2x^2 + 4x - 3 = 0$

2. $x^2 + 2x^3 - 16 + 2x = 0$

3. $x^5 - 2x^3 + 5x - 14 = 0$

Задания для самостоятельного изучения

1. Понятие многочлена.
2. Коэффициенты и корни многочлена.
3. Операции над многочленами.

Вопросы для самоконтроля

1. Раскрыть понятие многочлена.
2. Как многочлены записываются в Scilab?
3 .Как найти корни многочлена?
4. Какие операции над многочленами поддерживаются в Scilab?

Лабораторная работа №9

Пользовательские функции. Решение нелинейных уравнений

Цель работы: научаться работать с пользовательскими функциями в программе Scilab, а также, решать нелинейные урвнения.

Scilab позволяет пользователю задавать собственные функции. Чтобы задать функцию, содержащую уравнение, можно воспользоваться специальной встренной функцией `deff`, поторая принимает два строковых аргумента. Первый описывает общий вид будущей функции, например $y = f(x)$, где f – имя функции. Второй – описывает поведение будущей функции.

Пример. Создать функцию $y = f(x)$, где $f(x) = x^2 + 2x - 3$.

```
deff("y=f(x)", "y=x^2+2*x-3")
// теперь функцию f(x) можно использовать в вычислениях
// обращаясь к ней по имени f
t=f(1)
// в результате t будет равно 0
```

Для решения нелинейных уравнений в Scilab используется функция `fsolve`, принимающая два аргумента — приблизительное значение и пользовательскую функцию, содержащую заданное уравнение.

Пример. Найти корни уравнения $x^2 + 2,125x - 4,5 = 0$. Чтобы решить данную задачу, нужно создать пользовательскую функцию, которая будет содержать данное уравнение. Затем, заданную функцию нужно передать в качестве аргумента в функцию `fsolve`.

```
deff("y=f(x)", "y=x^2+2.125*x-4.5")
fsolve(0, f)
// в результате на экране отобразится число 1.3100316
```

После того, как корень найден, желательно проверить его правильность с помощью графика. Для этого нужно выбрать интервал, в который будет входить найденный корень. В данном случае подойдёт интервал $[-4; 2]$.

```
x=linspace(-4,2);
plot(x,f(x))
xgrid
```

В результате будет построен график (рис. 7), по которому видно, что, во-первых, корень найден верно, а во-вторых, имеется ещё один корень где-то между -3,5 и 3. В этом случае можно найти и второй корень, указав в качестве приблизительного значения, наример $-3,5$.

```
fsolve(-3.5, f)
// в результате на экране отобразится число -3.4350316
```

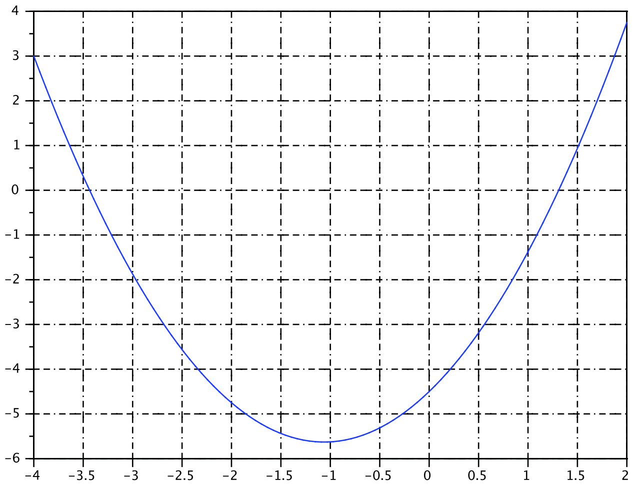

Рис. 7: График функции $y = x^2 + 2,125x - 4,5$

Также, в качестве первого аргумента можно указать вектор, состоящий, в данном случае, из двух приблизительных значений. В результате на экране появятся два числа — корни заданного уравнения.

```
fsolve([-3.5 0], f)
// на экране появятся два числа: -3.4350316 и 1.3100316
```

Важно! Функция `fsolve` не всегда верно находит корни уравнения. Желательно всегда проверять найденный с её помощью результат графически.

Задания для выполнения лабораторной работы

Решить нелинейные уравнение. Проверить найденные корни графически

1. $(x-1)^2 - x(x+2) = 2x - 3$
2. $2\ln(x^2+1) = 0$
3. $e^{\sin\frac{x}{2}} - \cos x = 0$

Задания для самостоятельного изучения

1. Пользовательские функции. Оператор deff.
2. Решение нелинейных уравнений.

Вопросы для самоконтроля

1. Что такое пользовательская функция?

2. Для каких целей применяются пользовательские функции?

3. С помощью каких функций можно найти корни нелинейного уравнения в Scilab?

4. Нужно ли результат проверять графически?

Лабораторная работа №10

Численное дифференцирование в Scilab

Цель работы: закрепить навыки работы с пользовательскими функциями в Scilab на примере численного дифференцирования.

Уравнение производной имеет следующий вид:

$$f'(x) = \lim_{\Delta x \to 0} \frac{f(x + \Delta x) - f(x)}{\Delta x}$$

где x – аргумент функции, Δx – приращение аргумента.

Уравнение касательной к графику в заданной точке имеет вид:

$$y = f(x_0) + f'(x_0)(x - x_0)$$

где x_0 – точка касания.

Пример. Дана функция $y = x^2 + 2x - 3$, определённая в интервале $x \in [-5; 5]$. Продифференцировать численно эту функцию, построить график этой функции, график производной и касательную в точке $x_0 = 1$.

Для решения этой задачи нужно представить заданную функцию, производную этой функции и функцию касательной к графику в виде пользовательских функций Scilab.

```
// функция
deff("y=f(x)", "y=x^2+2*x-3")
// производная
deff("y=df(x)", "dx=0.0001, y=(f(x+dx)-f(x))/dx")
// касательная
deff("y=k(x, x0)", "y=f(x0)+df(x0)*(x-x0)")
```

Теперь можно переходить к вычислениям и построению графиков.

```
x=-5:5;
// вывод на экран значений в виде таблицы
[x' f(x)' df(x)' k(x,1)']
```

```
Построение графиков
plot(x,f(x),"k-o")
plot(x,df(x),"b-.")
plot(x,k(x,1),"r-*")
legend("$y=f(x)$", "$y=f'(x)$", "$y=k(x,x_0)$")
```

Полученный график приведён на рис. 8

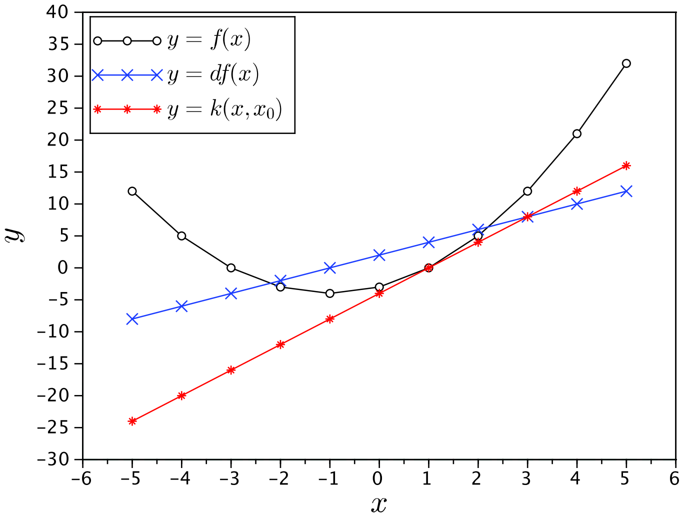

Рис. 8: График функции, производной и касательной в заданной точке

Задания для выполнения лабораторной работы

Построить графики фукций, производных и касательных

1. $y = \frac{x^4}{4} - 2x^2 - \frac{9}{4}$, $x \in [-4; 4]$, $x_0 = 0$
2. $y = \frac{4}{x^2-2x+2}$, $x \in [-5; 5]$, $x_0 = -1$
3. $y = x^3 - 3x^2 + 2$, $x \in [-2; 4]$, $x_0 = 0$
4. $y = \frac{x-4}{x^2+7}$, $x \in [-2; 3]$, $x_0 = -0,75$

Лабораторная работы №11

Основы работы в LibreCAD

Цель работы: получить навыки работы в программе LibreCAD.