Classes 6. Scilab: Symbolic and numerical calculations

Note: When copying and pasting, you need to delete pasted single quotation marks and type them manually for each occurrence.

6.1 Creating polynomials

There are two ways to create polynomials. First of them.

$p=poly([1\ 2\ 3],'z','coeff')$
$p =$

$1 + 2z + 3z^2$

where p is the polynomial of variable z with three coefficients 1,2,3.
It is possible to write 'c' in place of 'coeff'.
If we omit the third argument in the function call, it means that the generated polynomial has the elements given as the first argument here [1 2 3].

$p=poly([1\ 2\ 3],'z')$
$p =$

$- 6 + 11z - 6z^2 + z^3$

The above construction is used in case of increasing exponents. To generate any polynomial of the variable x

$\quad$ x = poly (0, 'x') \\

we create a polynomial with the variable x and we can now write this:

$p=1+x^2$
$p=1+x^2$

6.2 Roots of polynomials

Roots of the polynomial we find by using the command

$roots\ (p),$

where p is a predefined polynomial or polynomial written directly to the expression:

$p=poly([1\ 2\ 3],'z','c')$

$roots(p)$

$ans\ =$

$\quad - 0.3333333 + 0.4714045i$
$\quad - 0.3333333 - 0.4714045i$

6.3 Derivative of a polynomial

The command to calculate the derivative of a polynomial is derivat.

derivate(p)

The command

horner()

allows us to calculate the polynomial value at the selected point. For example,

horner(p, 3) is the value of polynomial *p* in point 3

6.4 Creating functions

Function creation takes place in Scilab in two ways: the first is simple but limited to simple mathematical expressions. We use command *deff()*

deff (*'function name and the method call'*, *'what calculates instructions for performing'*)

We call the function according to what we write in the first expression

deff('z=sincos(x)', 'z=cos(x)*sin(x)')

-->sincos(0)

ans =

0.

Alternatively, when we have more instructions to perform, we can put a variable and an expression in square brackets.

```
deff('[x]=mojeplus(y,z)','x=y+z ')
deff('[x]=mojemacro(y,z)',['a=3*y+1 '; 'x=a*z+y'])
```

The second way to define functions is to use the command:

function[*the arg list returned by the function*]=**name_function**(*call arguments*)
Instruction string…
endfunction

Example 1. Function for calculation of the product of natural numbers from 1 to *n:*

function[a]=iloczyn(n)

a=1

for x=1:n do

a=a*x

end

endfunction

-->b=iloczyn(4)
b=24

Example 2.

function[Area, Perimeter]= circle (radius)
--> Area=%pi* radius^2
--> Perimeter=2*%pi* radius
-->endfunction
- -> [P,O]= circle(6)

O = 37.699112
P = 113.09734

6.5 Function graph

Before drawing a graph, define the range for the variable - the function argument (usually x). We use the *linspace* command:

x=linspace(start value, end value, number of division points)

x =linspace(0,0.8,9)

x= 0. 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8

or

x= start value : step : end value

x=0: 0.3: 1

x = 0. 0.3 0.6 0.9

Simple graphics

To draw simple graphs of functions we can use the command *plot(x,y,x,y1)*. This command draw two graphs with arguments x and values of functions *y,y1*. We can use also command *plot2d(arg1, arg2, arg3)*, where *arg1* are the variables determining the domain, *arg2* are variables that store function values. The *arg3* values are shown in the following table:

Color	Parameter	Form	Parameter
Black	1	+	-1
Blue	2	×	-2
Green	3	⊕	-3
Blue 2	4	◆	-4
Red	5	◇	-5
Pink	6	△	-6
Red 2	7	▽	-7
White	8	⊕	-8
Navy blue	9	○	-9
Navy blue 2	10	✳	-10
Navy blue 3	11	□	-11

The following example illustrates the use of this command

x=linspace(0,2*%pi,101);

plot(x, exp(-x).*sin(4*x))

A command to draw graphs more than one function on a single graph looks

-->x=linspace(-1,1,61)';//wektor kolumnowy

-->x1=linspace(-3,1,61)';

-->y=x.^2;

-->yter=2*y;

-->ybis=1-x1.^2;

if the domain of all functions is the same:

plot2d([x x x],[y ybis yter])

If the function domain is different in this case, the *ybis* function has an area within the vector range *x1*.
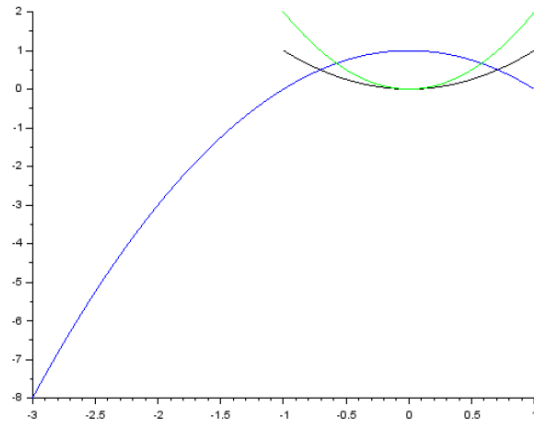
plot2d([x x1 x],[y ybis yter])

Fig 1. A graph of three functions with different domains

6.6. Numerical integration

The calculation of the value of the definite integral is relatively simple and it is reduced to using the command *integrate()*:

[x]=integrate('expr','x',x0,x1), where *expr* is expression under the integral

x – variable of integration

x0,x1 are integration limits

or command *intg()*:

[v,err]=intg(a,b,f)

where *v* is the value of the integral

err is a mistake

a and *b* are integration limits

f is a function name e.g. defined with a command *deff()*.

6.7. Finding of derivatives

The command *numdiff()* performs finding derivatives of one or more variables (determining the gradient of a function).

Example 2.

-->function[y]=fun(x)

*-->y=(x-3).^2+4*x.^2*

-->endfunction

-->numdiff(fun,2);

-->numdiff(fun,2)

ans = 14.

numdiff(fun,0)

ans = - 6.

6.8.   Systems of nonlinear equations

Use the command *fsolve(punkt start, funkcja)*. First we determine what vectors we will have, that is, order the equations. The same equations can be written in matrix form.

Example 3.

$$x_2 - \cos(x_1) - 8 = 0,$$
$$x_2 - 2\,x_1 - x_1^2 - 4 = 0;$$
$$x_2 - \cos(x_1) - 8 = y_1,$$
$$x_2 - 2\,x_1 - x_1^2 - 4 = y_2.$$

$$\begin{bmatrix} 0 & 1 \\ -2 & 1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ -1 & 0 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^2 + \begin{bmatrix} -1 & 0 \\ 0 & 0 \end{bmatrix}\cos\left(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\right) + \begin{bmatrix} -8 \\ -4 \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$

or

$$a\,X + b\,X^2 + c\,\cos(X) + d = Y,$$

where

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, a = \begin{bmatrix} 0 & 1 \\ -2 & 1 \end{bmatrix}, b = \begin{bmatrix} 0 & 0 \\ -1 & 0 \end{bmatrix}, c = \begin{bmatrix} -1 & 0 \\ 0 & 0 \end{bmatrix}, d = \begin{bmatrix} -8 \\ -4 \end{bmatrix}, Y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$

function [Y]=fst(X)

// we accept X=[x1;x2], Y=[y1;y2]

a=[0,1;-2,1]; b=[0,0;-1,0]; c=[-1,0;0,0]; d=[-8;-4];

Y= a*X +b*X^2 + c*cos(X) + d

endfunction

// or another way

function [Y]=fst(X)

Y= [X(2)-cos(X(1))-8; X(2)-2*X(1)-X(1)^2-4]

endfunction

// the initial solution is the point (0,0)

xy1=fsolve([0;0],fst);

Solution:

xy1=[1.2959523; 8.2713968]

the starting point (-3,0)

xy2=fsolve([-3;0],fst);

Example 4.

$$\begin{cases} y - x^2 + 1 = 0, \\ x - \dfrac{1}{3}(2y - y^2) = 0. \end{cases}$$

The variables x and y are inserted into the vector **x**, and the results stores the vector z.

function [z]=zadanie(x)

z(1)=x(2)-x(1).^2+1;

z(2)=x(1)-(2*x(2)-x(2).^2)/3;

endfunction

// axis setting in the middle of the graph:

ster_axe=gca();

ster_axe.y_location="origin";

ster_axe.x_location="origin";

// sketching the graphs:

x1=linspace(-3,3,101);

y1=x1.^2-1;

y2=linspace(-3,5,101);

x2=(2*y2-y2.^2)/3;

// 3 and 5 represents the color of the graph

plot2d(x1,y1,3);

plot2d(x2,y2,5);

// finding solutions:

init=[-5;8];

// the value close to solution on a sketch chart

sol=fsolve(init, task)

disp(sol)

plot2d(sol(1:1), sol(2:2),-5)

```
init=[3;-4];
sol=fsolve(init, task)
disp(sol)
plot2d(sol(1:1), sol(2:2),-5)
// sol(1:1) – first coordinate sol,
// sol(2:2) – second coordinate sol;
//-5 is a diamond sign
```

6.9   Exercises

Exercise 1. Solution of equations:

a) Find the roots of a polynomial $x^4 - 0.7\,x^3 + 7.1\,x^2 - 4.9\,x + 0.7$;

b) Find the solution of the equation $y(x) = 0$, where $y = 16\,x^3 - 4.8\,x^2 + 5.6\,x - 9.6$;

c) Find the solution of the equation $\left(x - \frac{3}{2}\right)^{1/5} - \left(x + \frac{59}{2}\right)^{\frac{1}{5}} + 1 = 0$;

d) Find the real solution of the system of equations $\begin{cases} x^2 - y^2 = 3, \\ x^4 - 16 = 0. \end{cases}$ Draw graphs of functions and make visualization of solutions.

e) Find the real solution of the system of equations $\begin{cases} \sin\left(y + \frac{1}{2}\right) = x^2, \\ 3x^2 + y^2 = 2. \end{cases}$ Draw graphs of functions and make visualization of solutions.

What kind of equations are these?

Exercise 2. Calculation values of derivatives in a point:

a) Calculate value of the derivative $f'(2)$, where $f(x) = (x - 3)^2 + 4x$;

b) Calculate the gradient of a function at the point (1,1): $y(x_1, x_2) = x_1^2 x_2^3 + x_2$;

c) Calculate partial derivatives (gradient of function y)
$\frac{\partial y}{\partial x_1}(2, -3,1)$, $\frac{\partial y}{\partial x_2}(2, -3,1)$, $\frac{\partial y}{\partial x_3}(2, -3,1)$, where $y(x_1, x_2, x_3) = x_1^2 x_2^{x_3} + x_1 x_3^3$.

Exercise 3. Find the definite integrals:

a) $\int_2^7 \sqrt{4x - 3}\, dx$;

b) $\int_0^{\pi/2} \sin(3x)\, dx$.

Exercise 4. For arbitrary variables x, y, z calculate the value of the expressions a and b:

a) $a = \sqrt{|x-1|} - \sqrt[3]{|y|}$, $b = x\left(\arctan z + e^{-(x+3)}\right)$;

b) $a = \frac{3+e^{y-1}}{|y-\tan z|}$, $b = 1 + |y - x| + \frac{(y-x)^2}{2} + \frac{|y-x|^3}{3}$;

c) $a = (1 + y)\frac{x+y/(x^2+4)}{e^{-x-2}+1/(x^2+4)}$, $b = \frac{1+\cos(y-2)}{x^4+\sin^2 z}$;

d) $a = \frac{2\cos(x-\pi/6)}{1/2+\sin^2 y}$, $b = 1 + \tan^2\frac{z}{2}$;

e) $a = \ln\left|(y - \sqrt{|x|})\left(x - \frac{y}{z+x^2/4}\right)\right|$, $b = \cos^2\left(\arctan\frac{1}{z}\right)$.

Helpful functions:

abs	the absolute value	acos	arccos
exp	exponent function	asin	arcsin
log	natural logarithm	atan	arctg
log10	logarithm of base 10	cosh	hyperbolic cosine
cos	cosine (argument in radians)	sinh	hyperbolic sine
sin	sine (argument in radians)	tanh	hyperbolic tangent
sinc	$\frac{\sin x}{x}$	acosh	argch
tan	tangent (argument in radians)	asinh	argsh
cotg	cotangent (argument in radians)	sqrt	square root

Exercise 5. Determine the sum value

$$1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \frac{1}{5} - \frac{1}{6} + \frac{1}{7} - \frac{1}{8} + \frac{1}{9} - \frac{1}{10} + \frac{1}{11} - \frac{1}{12}\ .$$

How to write this action on several lines? How to calculate the value of an analog expression of 20 elements?