Mathematics for Engineers

Pál Burai

Numerical mathematics

This work was supported by the construction EFOP-3.4.3-16-2016-00021. The project was supported by the European Union, co-financed by the European Social Fund.

Examples

Let $a = 2,\ t = 4,\ k_- = -3,\ k_+ = 2.$

- Determine the floating point form of 0.1875! **Solution:**

$$0.1875 = 0 \cdot \frac{1}{2} + 0 \cdot \frac{1}{4} + 1 \cdot \frac{1}{8} + 1 \cdot \frac{1}{16} \implies 0.1875_{10} = 0.0011_2$$

  Normal form: $2^{-2} \cdot 0.1100$

- Determine the floating point form of $\frac{1}{3}$! **Solution:**

$$0.3333\cdots = 0\cdot\frac{1}{2}+1\cdot\frac{1}{4}+0\cdot\frac{1}{8}+1\cdot\frac{1}{16}+\cdots \implies 0.3333_{10} = 0.0101\ldots_2$$

  Normal form: $2^{-1} \cdot 0.1010\ldots_2$
  After truncation: $2^{-1} \cdot 0.1010$
  After rounding: $2^{-1} \cdot 0.1011$

Exercises

- Let $a = 2,\ t = 4,\ k_- = -3,\ k_+ = 3$. Determine the floating point form of the following numbers!

$$0.9375, \quad 0.1, \quad 0.6, \quad \frac{1}{32}, \quad 2.625$$

- Determine the largest floating point number (M_∞), the smallest positive floating point number (ε_0) and the left- and right neighbours of 1 if $a,\ t,\ k_-$ and k_+ are given!
- How many floating point numbers are representable if $a,\ t,\ k_-$ and k_+ are given?
- Let $a = 2,\ t = 4,\ k_- = -3,\ k_+ = 3$. Represent all the floating point numbers on the real line!

Exercises

- Read the help of the Matlab function `eps`.
- Examine the logical expressions: `0.4-0.5+0.1==0` and `0.1-0.5+0.4==0`! What will be their value?
- Let $x = \frac{1}{3}$. The expression $4x - 1 = x$ should return with $\frac{1}{3}$. Run the command `x = 4*x-1=x` at least 40 times in a `for` loop. What will be the value of x!
- Examine the value of the logical expressions `10^20+1==10^20`, `10^20+10==10^20`, `10^20+100==10^20`, `10^20+1000==10^20`, and `10^20+10000==10^20`!
- The algorithm below should return with x if x is an arbitrary positive real number. Run the algorithm with initial values $x = 1000$ and $x = 100$. What is your experience? Give an explanation for this phenomena!

```
x=1000;
for i = 1:60
      x = sqrt(x);
end
x
for i = 1:60
      x= x^2;
```

Exercises

- It is known that $\lim_{x \to 0} \frac{e^x - 1}{x} = 1$. Calculate with Matlab the ration for decreasing values! (Initialize x as 1 and divide it by 2 forty times, two hundred times, and two thousand times.) Give and explanation for the experienced phenomena!
- Solve with Matlab the linear system of equations $Ax = b$, where

$$A = \begin{bmatrix} 1 & 0.99 \\ 0.99 & 0.98 \end{bmatrix}, \quad b = \begin{bmatrix} 1.99 \\ 1.97 \end{bmatrix}, \quad \text{and} \quad b = \begin{bmatrix} 1.98 \\ 1.98 \end{bmatrix}.$$

Exercises

- Calculate the condition number of the 6×6 Hilbert-matrix! Use the `cond` and `hilb` Matlab functions! Let B be a 6×6 random matrix (use the `rand` function to generate it), calculate its condition number, try several times.
- Generate the 100×100 matrix A and the 100 dimensional vector b according to the expressions below. Solve the linear system of equations $Ax = b$ using the `backslash` operator. After this perturb b e.g. $b(100) = 1.00001$ instead of $b(100) = 1.0000$. Solve the system again, and calculate its condition number!

$$a_{ij} = \begin{cases} 1 & \text{if } i = j \\ -1 & \text{if } i < j \\ 0 & \text{otherwise,} \end{cases} \qquad b = \begin{bmatrix} -98 \\ -97 \\ \vdots \\ 0 \\ 1 \end{bmatrix}$$

Least squares problem, polynomial model

Example

Determine the straight line which gives the best fit to the data in least squares sense:

t_i	1	1.1	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2
f_i	8	8.9	9	9.8	10	11	11.5	11.5	12.5	13	13.7	14

Solution: use the `polyfit` function.

```
 t = [1 1.1 1.1:0.1:2];
f = [8 8.9 9 9.8 10 11 11.5 11.5 12.5 13 13.7 14];
p = polyfit(t,f,1)
x = linspace(0.5,2);
y = polyval(p,x);
plot(t,f,'*',x,y)
```

which gives first order polynomial which the best fit to the data in least squares sense. One can use any other positive integer instead of 1 for higher order least square approximation.

Least squares problem

The equation of the best fitting line is

$$f(t) = 5.8235t + 2.5338$$

We can plot the result and the date with the `refline` function. The first argument is the slope of the line, the second one is the constant term, and the last one is the marker type:

```
figure; plot(t,f,'*');
hold on; refline(p)
```

An another possibility is (applicable for an arbitrary polynomial):

```
xx = linspace(0.9,2.1);
yy = polyval(p,xx);
figure; plot(t,f,'*',xx,yy)
```

The `polyval` function calculates values of a polynomial with coefficients p at xx.

Least squares problem

Exercise

Determine the second order polynomial which gives the best fit to the data in least squares sense:

t_i	2.1	2.2	2.3	2.3	2.5	2.6	2.8	2.9
f_i	2.5	2	2.5	2.7	3	4	5.4	7

Example

Determine the function $F(t)$ which gives the best fit to the data in least squares sense and

$$F(t) = x_1 + x_2 \cos(\pi t) + x_3 \sin(\pi t).$$

t_i	0.1	0.5	1.2	1.5	2	2.1	2.4	3	3.2
f_i	3.9	2.6	-0.8	0.3	3.2	3.8	3.2	-0.7	-0.9

Least squares problem

Solution: The parameters are the solutions of the following linear system of equations, which is called the **(Gaussian) normal equation:**

$$A^T A x = A^T f,$$

where

$$A = \begin{bmatrix} 1 & \cos(\pi t_1) & \sin(\pi t_1) \\ 1 & \cos(\pi t_2) & \sin(\pi t_2) \\ \vdots & & \vdots \\ 1 & \cos(\pi t_9) & \sin(\pi t_9) \end{bmatrix}, \quad f = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_9 \end{bmatrix}, \quad b = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}.$$

Construct the matrix A from the given data, and solve the normal equation!

```
t = [0.1 0.5 1.2 1.5 2 2.1 2.4 3 3.2]';
f = [3.9 2.6 -0.8 0.3 3.2 3.8 3.2 -0.7 -0.9]';
A = [ones(9,1), cos(pi*t), sin(pi*t)];
x = (A'*A)\(A'*f)
```

Least squares problem

The model which fits the best and has the required form is

$$F(t) = 1.4372 + 2.0310\cos(\pi t) + 1.711\sin(\pi t).$$

Plot the data and the model function in one figure!

```
xx = linspace(0,3.3);
yy=x(1)+x(2)*cos(pi*xx)+x(3)*sin(pi*xx);
plot(t,f,'*',xx,yy)
```

Exercises

- Determine the straight line which gives the best fit to the data in least squares sense:

t_i	1	1.2	1.4	1.4	1.5	1.7	1.9	2
f_i	6.2	7	8	7.9	8.4	9.2	10	10.6

- Determine the third order polynomial which gives the best fit to the data in least squares sense:

t_i	0.5	0.7	1.1	1.3	1.5	1.7	1.9	2.1	2.3
f_i	2.5	2.3	1.8	1.3	0.9	0.4	0.1	-0.05	-0.01

- Determine the parameters of the model function $F(t) = a + \frac{b}{t}$ which gives the best fit to the data in least squares sense:

t_i	1	1.2	1.4	1.4	1.5	1.7	1.9	2	2.1	2.2
f_i	4.2	3.8	3.4	3.3	3.3	3	2.8	2.8	2.75	2.7

Exercises

- Determine the parameters of the model

$$F(t) = x_1 \sin(t) + x_2 \sin(2t) + x_3 \sin(3t)$$

which gives the best fit to the following data in least squares sense:

t_i	0.1	0.5	1.2	1.5	2	2.1	2.4	3	3.2	3.4	3.8	4	4.2	4.6	5
f_i	1	4.1	3	1	-1.5	-1.6	-1.7	-0.4	0.1	0.7	1.6	1.8	1.6	0.2	-2.5

- Determine the parameters of model

$$F(t) = x_1 + x_2 \ln(t)$$

which gives the best fit to the following data in least squares sense:

t_i	0.1	0.5	1.2	1.5	2	2.1	2.4	3	3.2
f_i	-0.6	1.5	2.5	2.9	3.2	3.3	3.5	3.8	3.9

Example

Determine the Lagrange interpolation polynomial, which fits to the following data: $(-2,-5),(-1,3),(0,1),(2,15)$.

Solution: Let's make the divided difference scheme:

Lagrange interpolation, exercises

Determine the Lagrange interpolation polynomial, which fits to the following data:

- $(-3,-6),(-2,-17),(-1,-8),(1,-2),(2,19)$;
- $(-3,-31),(-2,-8),(1,1),(2,22)$;
- $(-2,13),(-1,-4),(1,2)$;
- $(-2,-5),(-1,3),(0,1),(2,15)$;
- $(-1,4),(1,2),(2,10),(2,15)$;
- $(-2,38),(-1,5),(1,-1),(2,-10),(3,-7)$;
- $(-2,-33),(-1,-2),(1,6),(2,7),(3,-18)$;
- $(-3,-209),(-2,-43),(-1,-1),(1,-1),(2,-19)$.

- Find the Lagrangian interpolation polynomial which fits to the following data: $(-2,-6),(0,4),(1,-3),(2,-10)$. After the solution add the new point $(-1,2)$ to the previous points and calculate the minimal degree polynomial which fits to these data.

Lagrange interpolation with Matlab

The `polyfit(x,f,n-1)` function gives the coefficients of the polynomial of degree at most $n-1$, which fits to the data (x, f).

Example

Determine with Matlab the minimal degree polynomial which fits to the data $(-2,-5),(-1,3),(0,1),(2,15)$.

```
x = [-2 -1 0 2];
f = [-5 3 1 15];
p = polyfit(x,f,3)
```

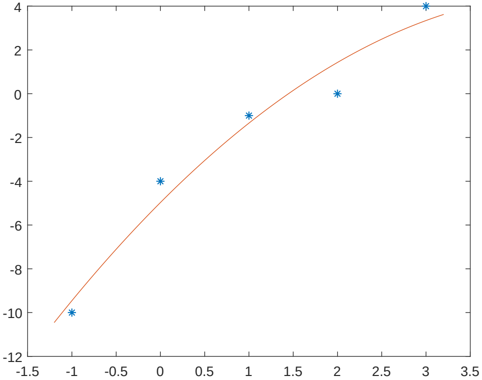
Plot the data and the polynomial in the same figure! Use the `polyval` function, which evaluates a polynomial in a given vector.

```
x = [-2 -1 0 2];
f = [-5 3 1 15]
p = polyfit(x,f,3);
xx = linspace(-2.5,2.5);
yy = polyval(p,xx);
figure; plot(x,f,'*',xx,yy)
```

Important remark

Be careful with the choice of the degree in `polyfit` function, otherwise the polynomial not necessarily fits to the data.

```
x = [-1 0 1 2 3]; f = [-10 -4 -1 0 4];
p = polyfit(x,f,2);xx = linspace(-1.2,3.2);
ff = polyval(p,xx);figure;plot(x,f,'*',xx,ff)
```

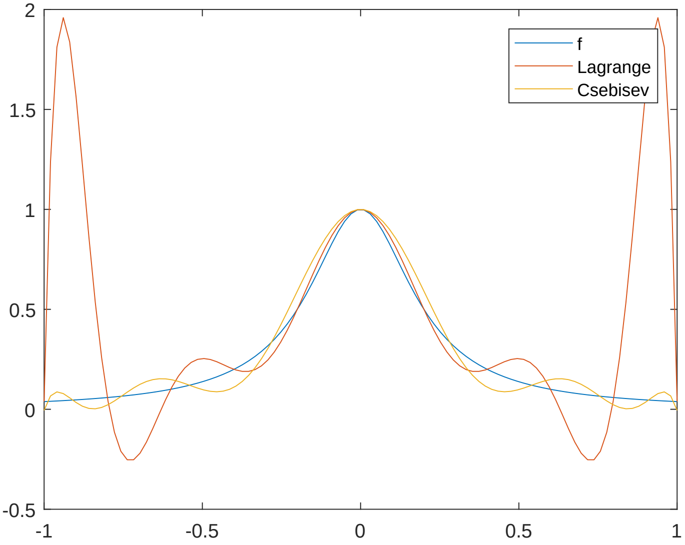


Exercise

Plot in one figure the following three functions:

- $$f(x) = \frac{1}{1+25x^2}$$
  on the interval $[-1, 1]$;
- the Lagrange polynomial of f belonging to the equidistant set of nodes with step size 0.2;
- the Lagrange polynomial of f belonging to the Chebyshev nodes:

$$x_k = \cos\left(\frac{2k-1}{22}\pi\right), \qquad k = 1, 2, \ldots, 11.$$


f
Lagrange
Csebisev
2
1.5
1
0.5
0
-0.5
-1
-0.5
0
0.5
1

Example

Determine the Hermitian interpolation polynomial, which fits to the following data:

x_i	-2	-1	1
$f(x_i)$	-10	-2	2
$f'(x_i)$	-20	10	10
$f''(x_i)$		-16	

Solution: We have seven fitting conditions, so the degree of the Hermite-polynomial will be at most six. Let's make the divided difference scheme.

-2	-10						
		-20					
-2	-10		$\frac{8-(-20)}{-1-(-2)} = 28$				
		$\frac{-2-(-10)}{-1-(-2)} = 8$		$\frac{2-28}{-1-(-2)} = -26$			
-1	-2		$\frac{10-8}{-1-(-2)} = 2$		16		
		10		$\frac{-8-2}{-1-(-2)} = -10$		-4	
-1	-2		$\frac{-16}{2}$		4		1
		10		$\frac{-4-(-8)}{1-(-1)} = 2$		-1	
-1	-2		$\frac{2-10}{-1-(-1)} = -4$		1		
		$\frac{2-(-2)}{1-(-1)} = 2$		$\frac{4-(-4)}{1-(-1)} = 4$			
1	2		$\frac{10-2}{1-(-1)} = 4$				
		10					
1	2						

$$\mathscr{H}(x) = -10 - 20(x+2) + 28(x+2)^2 - 26(x+2)^2(x+1) + 16(x+2)^2(x+1)^2$$

$$-4(x+2)^2(x+1)^3 + 1(x+2)^2(x+1)^3(x-1)$$

Exercises

Determine the Hermitian interpolation polynomial, which fits to the following data!

x_i	-1	1	2
$f(x_i)$	4	6	94
$f'(x_i)$	9	17	213

x_i	-2	-1	1
$f(x_i)$	13	3	7
$f'(x_i)$	-31	14	18
$f''(x_i)$		-40	

Exercise

Determine the equation of the tangent line at x_0 of an arbitrary $f : \mathbb{R} \to \mathbb{R}$ differentiable function!

Spline interpolation with Matlab

Example

Determine the third order spline interpolation polynomial which fits to the data below!

x_i	-2	-1	0	1	2	3
S	4	1	7	4	12	9
S'	15					8

Solution: Use the `spline` function!

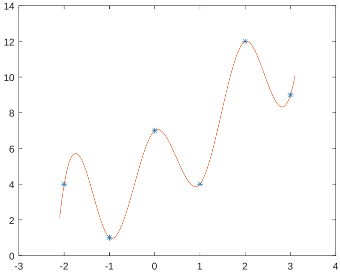
```
x = -2:3;
y = [15 4 1 7 4 12 9 8];
xx = linspace(-2.5,3.5);
p = spline(x,y)
p.coefs
plot(x,y(2:end-1),'*',xx,ppval(p,xx))
```

The `p.coefs` command gives the coefficients of the spline.

Spline interpolation with Matlab

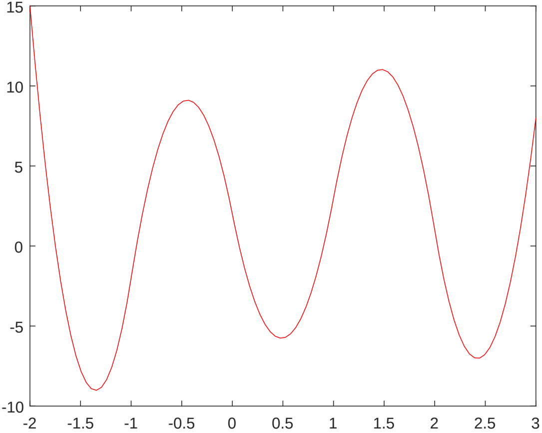
The `yy = spline(x,y,xx)` command gives the values of the spline, where `x` and `y` are the previously defined vectors, and `xx` contains those values where we would like to evaluate the function. In this case `yy` contains the evaluations.

```
x = -2:3;
y = [15 4 1 7 4 12 9 8];
xx = linspace(-2.1,3.1);
yy = spline(x,y,xx);
plot(x,y(2:end-1),'*',xx,yy)
```



Spline interpolation with Matlab

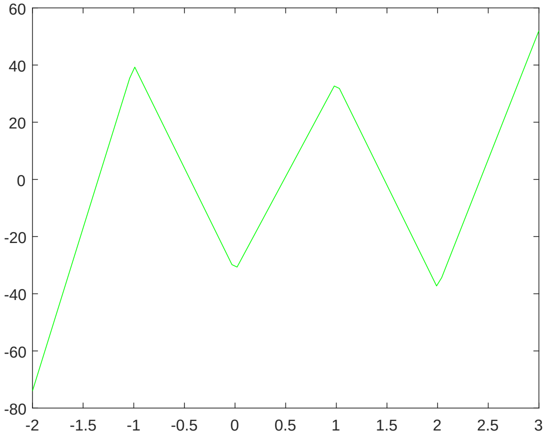
First derivative of the previously plotted spline:



This is continuously differentiable.

Spline interpolation with Matlab

Second derivative of the previously plotted spline:



This is continuous, however, it is not differentiable everywhere.

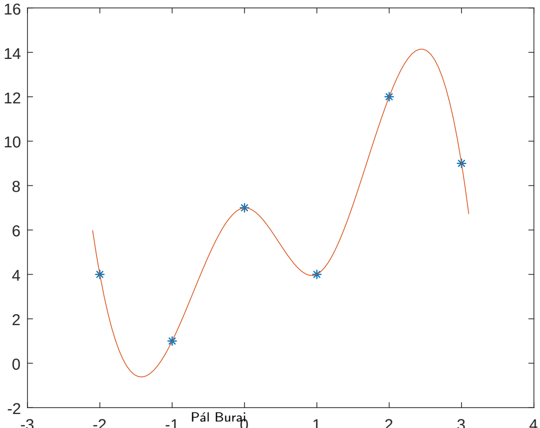
Remark

If the `spline` function is called with vectors of the same size, then the missing two conditions are substituted by Matlab with the continuity condition of the third derivative at the joining points of the first two and the last two subintervals.

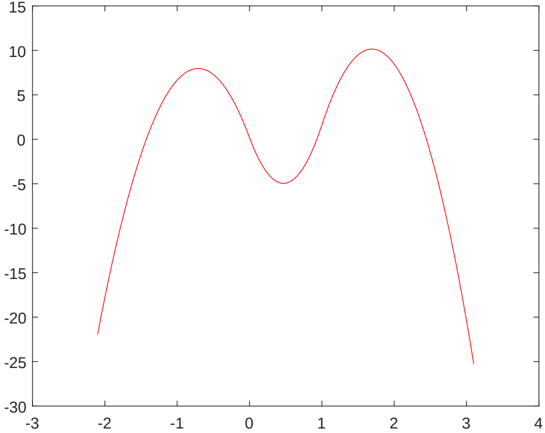
```
x = -2:3;
y = [4 1 7 4 12 9];
xx = linspace(-2.1,3.1);
yy = spline(x,y,xx);
plot(x,y,'*',xx,yy)
```

Spline interpolation with Matlab

```
x = -2:3;
y = [4 1 7 4 12 9];
xx = linspace(-2.1,3.1);
yy = spline(x,y,xx);
plot(x,y,'*',xx,yy)
```



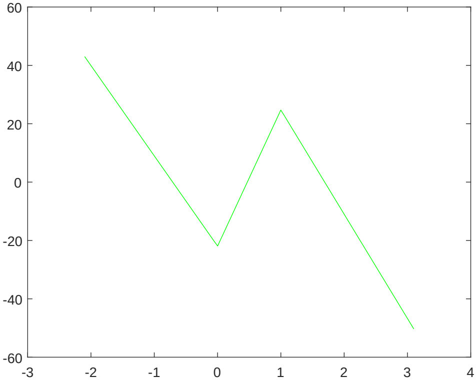
First derivative of the previously plotted spline:



This is continuously differentiable.

Spline interpolation with Matlab

Second derivative of the previously plotted spline:



This is continuous, however, it is not differentiable everywhere. Moreover, at the first and at the last node (in -1 and in 2) it has no corner.

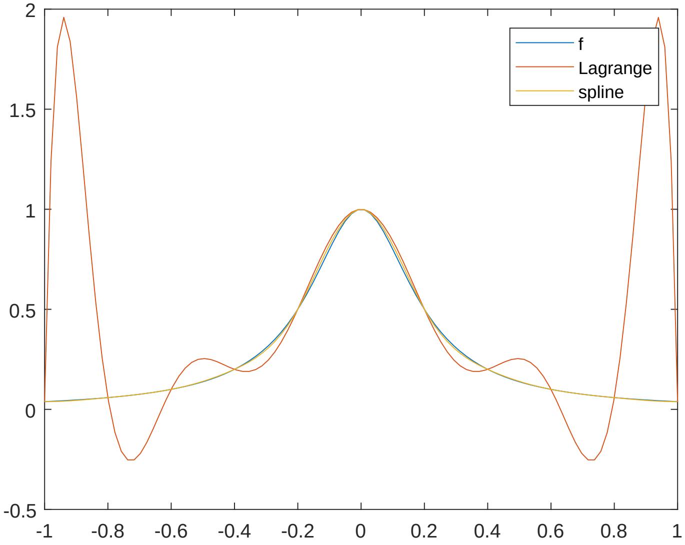
Exercise

Plot in one figure the following three functions:

- $$f(x) = \frac{1}{1+25x^2}$$

  on the interval $[-1, 1]$;
- the Lagrange polynomial of f belonging to the equidistant set of nodes with step size 0.2;
- the third order spline interpolation polynomial belonging to the the equidistant set of nodes with step size 0.2. (At the ends the value of the derivative is defined as zero.)

Spline interpolation with Matlab



Anonymous functions, functions handle

Definition of a function in the command line

```
f1 = @(x) x.*sin(x);
```

This defines the function $f1(x) = x\sin(x)$. We can call it with the command: `y=f1(pi/4)`

The variables of the function (in the current case this is only x) are after the symbol @, this is followed by the function itself (anonymous function). The variable (it is $f1$ here) on the left hand side of $=$ is a "function handle" type variable.
One can define a multivariable function in this way, e.g.:

```
f2 = @(x,y) x.^2+x.*y-y+3
```

Numerical integration

One can integrate a one-variable function with the `integral` command.

Example

Calculate the following integral with Matlab!

$$\int_0^3 x\sqrt{1+x}\,\mathrm{dx}$$

Solution:

```
f = @(x) x.*sqrt(1+x);
integral(f,0,3)
```

The call of the `integral` function in general:

```
integral(fv,xmin,xmax)
```

where `fv` is the integrand (it is a handle type variable), `xmin` and `xmax` are the lower and upper limit of the integral.

Numerical integration, Parametric integration

The `integral` function uses adaptive quadrature rule, the default values of the absolute and the relative error are 10^{-10} and 10^{-6} respectively. These values are controllable:

```
integral(f,0,3,'RelTol',1e-8,'AbsTol',1e-13)
```

One can write the function into the `integral` command:

```
integral(@(x) x.*sqrt(1+x),0,3)
```

If we defined the function earlier, then it can be passed it as a function handle, e.g.:

```
function y = myfnc(x)
y = x.*sqrt(1+x)
end
integral(@myfnc,0,3)
```

One can use Matlab's inner functions in a pretty similar way, e.g.:

```
integral(@sin,0,pi)
```

Numerical integration, Improper integrals

The limits of the integral can be $-\infty$ and/or ∞:

```
f = @(x) exp(-x);
integral(f,0,Inf)
```

This works also in the case when the function is not defined at the endpoints of an interval:

```
f = @(x) 1./sqrt(1-x.^2);
integral(f,-1,1)
```

If the integrand contains a parameter, we can calculate the integral if a value is assigned to the parameter.

$$\int_0^5 x^2 - cx + 3\,\mathrm{dx}$$

Here c is a parameter.

```
f = @(x,c) x.^2-c*x+3;
```

The value of the integral if $c = 4.5$ can be found with the command

```
integral(@(x) f(x,4.5),0,5)
```

Numerical integration, Unknown function

Assume that we know the values of a function only at some points. One can use the `trapz` function to calculate the integral in this case.

Example

The velocity of a vehicle is measured during one minute at every five second:

t (sec)	0	5	10	15	20	25	30	35	40	45	50	55	60
v (m/sec)	2.2	2.8	3	3	2.7	2.5	2.4	2.9	3.3	3.5	3.5	3.3	3

Let us estimate the length of the covered distance.

Numerical integration, Unknown function

Solution: Using the known expression for the covered distance, our task is to estimate the value of the following integral.

$$\int_0^a v(t)\,\mathrm{dt}$$

```
x = 0:5:60;
f = [2.2 2.8 3 3 2.7 2.5 2.4 2.9 3.3 3.5 3.5 3.3
    3];
trapz(x,f)
y = cumtrapz(x,f)
```

where the ith coordinate of y is the the covered distance till the ith instant of time.

Exercises

- $$\int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} x \sin(x^2)\,\mathrm{dx}$$
- $$\int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} e^{\frac{-x^2}{2}}\,\mathrm{dx}$$
- $$\int_{-1}^{1} \sqrt{1-x^2}\,\mathrm{dx}$$

The `fzero` function

It approximates such a root of f where the function changes sign in a neighborhood of the root. If the function is given as a handle, then

- The command

```
x = fzero(f,x0)
```

  gives an approximation of a root x starting the iteration from $x0$.

- The command

```
[x,fval,exitflag,output] = fzero(f,x0)
```

  starts the approximation of the root from $x0$, it gives the value of the function (fval) at the end of the iteration, the reason of termination (exitflag), and some details of the process (output).

Example

Find with Matlab the root of $f(x) = e^x - 3x^2$ in the interval $[0, 1]$.

Solution:

```
f = @(x) exp(x)-3*x^2;
[x,fval,exitflag,output]=fzero(f,0.5)
```

The rounded root is 0.7148, if we would like to know more digits change the format of the result.

```
format long
x
```

If we would like to get more information about the process use the command:

```
[x,fval,exitflag,output]=fzero(f,0.5)
```

Nonlinear equations with Matlab

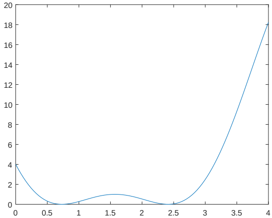
One can fasten the process if an interval is known where the function change its sign. It is possible to call the fzero function with this interval instead of $x0$.

```
[x,fval,exitflag,output]=fzero(f,[0,1])
```

The fzero function doesn't find the root $\bar{x}$ if f doesn't change its sign in a neighbourhood of $\bar{x}$.

```
f = @(x) 13-9*cos(x).^2-12*sin(x);
x = fzero(f,0)
```

Plot the function on the interval $[0, 4]$. Our guess is that the function has two roots close to 0.5 and 2.5, but there is no sign change there.



Nonlinear equations with Matlab

The `fminbnd` command

It finds the minimum of the function in a given interval.

```
x = fminbnd(f,xmin,ymin)
[x,fval,exitflag, output] = fminbnd(f,xmin,xmax)
```

Example:

```
f = @(x) 13-9*cos(x).^2-12*sin(x);
[x,fval] = fminbnd(f,0,1)
[xx,fval] = fminbnd(f,2,3)
```

If the function f has a root which is a local maximum as well, then we should call the `fminbnd` function with $-f$.

Important remark

If we looking for a root with `fminbnd`, then check always that it is really a root or not.

Nonlinear equations with Matlab

Roots of polynomials with `roots` command

Let us find the roots of the polynomial $x^4 + x^3 - x - 1$.

```
p = [1 1 0 -1 -1];
roots(p)
```

Exercises

Approximate the solutions of the following equations:

- $3x = \cos(x)$,
- $3x^3 - 12x + 4 = 0$,
- $e^x = \sin(x)$,
- $\log(x) = 2 - x$,
- $\cos^2(x) + 2\sin(x) = 2$,
- $x^4 - x^3 - 2x^2 - 2x + 4 = 0$.