

INTERPOLACE A APROXIMACE

Interpolace polynomem

```
>> x = 0 : 4; % x = [ 0 1 2 3 4 ]
>> y = [ 3 1 2 6 8 ];
>> xx = 0 : 0.1 : 4; % deleni intervalu < 0 ; 4 > s krokem 0.1
>> n = length(x)-1; % n ... stupen polynomu
>> p = polyfit(x,y,n) % p ... vektor koeficientu polynomu
>> yy = polyval(p,xx); % yy ... hodnoty polynomu spoctene s krokem 0.1
>> plot(x,y,'ro',xx,yy,'b') % graf polynomu - dane body jsou cervene krouzky, graf je modry
```

Porovnání interpolace polynomem a kubickým splajnem:

Nechte proběhnout předchozí příklad, nezavírejte okno s obrázkem a zadejte tyto příkazy:

```
>> yy1 = spline(x,y,xx); % yy1 ... hodnoty kubického splajnu
>> hold on % chceme kreslit do stejného obrazku
>> plot(x,y,'ro',xx,yy1,'g') % zelený graf splajnu
```

Na následujícím příkladu lépe vynikne výhodnost interpolace splajnem:

```
>> x = -4 : 4; % x = [ -4 -3 -2 -1 0 1 2 3 4 ]
>> y = [ 0 0 0 0 1 2 6 7 7];
>> xx = -4 : 0.1 : 4; % deleni intervalu < -4 ; 4 > s krokem 0.1
>> n = length(x)-1; % n ... stupen polynomu
>> p = polyfit(x,y,n) % p ... vektor koeficientu polynomu
>> yy = polyval(p,xx); % yy ... hodnoty polynomu spoctene s krokem 0.1
>> hold off % chceme kreslit do noveho obrazku
>> plot(x,y,'ro',xx,yy,'b') % graf polynomu - dane hodnoty cervene krouzky, graf modry
>> yy1 = spline(x,y,xx); % yy1 ... hodnoty kubického splajnu
>> hold on % chceme kreslit do stejného obrazku
>> plot(x,y,'ro',xx,yy1,'g') % zelený graf splajnu
```

Aproximace polynomem metodou nejmenších čtverců

Proložení přímky

```
>> x = -4 : 4; % x = [ -4 -3 -2 -1 0 1 2 3 4 ]
>> y = [ 0 0 0 0 1 2 6 7 7];
>> xx = -4 : 0.1 : 4; % deleni intervalu < -4 ; 4 > s krokem 0.1
>> p = polyfit(x,y,1) % proložení přímky metodou nejmenších čtverců
>> yy = polyval(p,xx); % yy ... hodnoty na přímce spoctene s krokem 0.1
>> plot(x,y,'ro',xx,yy,'b') % graf - dane body jsou cervene krouzky, přímka je modrá
```

Funkce pro postupnou aproximaci dat polynomy všech stupňů od přímky až po interpolační polynom:

```
function mnc( X, Y, krok)
% mnc( X, Y, krok)
% vstup:
%   X - radkový vektor x-ových souřadnic daných bodů (aspoň dva body, nekontroluje se)
%   Y - radkový vektor y-ových souřadnic daných bodů (správný rozměr se nekontroluje)
%   krok>0 - se kterým se budou kreslit grafy polynomu
% vystup:
%   obrazek s aproximacními polynomy od přímky po interpolaci
%   další polynom se nakreslí vždy po stisku mezerníku
n = length(unique(X)); % počet různých hodnot v X
% deleni intervalu s daným krokem:
minx = min(X);
maxx = max(X);
xx = linspace(minx, maxx, (maxx-minx)/krok);
p = polyfit(X,Y,1); % přímka
yy = polyval(p,xx); % body na grafu přímky
hold off; % nový obrazek
plot(X,Y,'ro',xx,yy);
hold on; % dál se bude kreslit do stejného obrazku
for i = 2 : n-1
    pause % čekání na stisk mezerníku
    p = polyfit(X,Y,i);
    yy = polyval(p,xx);
    plot(xx,yy);
end % for
end % function
```

Funkci můžeme zavolat takto (vždy po stisknutí mezery nakreslí další polynom):

```
>> x = -4 : 4;
>> y = [ 0 0 0 0 1 2 6 7 7];
>> mnc(x, y, 0.2)
```