

TUTORIÁL4 - APLIKAČNÉ VYUŽITIE PRÁCE SO SÚBORMI V ÚLOHÁCH REGRESNEJ ANALÝZY V PROSTREDÍ MATLAB

NÁPLŇ

1. PRÁCA SO SÚBORMI
 - a. UKLADANIE NAČÍTAVANIE DÁT DO/Z MAT SÚBOROV
 - b. UKLADANIE A NAČÍTAVANIE DÁT DO/Z TEXTOVÝCH SÚBOROV
 - c. UKLADANIE A NAČÍTAVANIE DÁT DO/Z XLS SÚBOROV
2. APROXIMÁCIA
3. INTERPOLÁCIA
4. PRÍKLADY NA SAMOSTATNÉ RIEŠENIE

T3 - 1. PRÁCA SO SÚBORMI

Matlab umožňuje spracovanie súborov rôznych formátov. Nemusíme mať dáta uložené priamo v prostredí Matlab. V tomto programovom prostredí dokážeme exportovať a importovať dáta vo formáte aký si zvolíme. V tejto kapitole sa budeme venovať súborom typu MAT, textovým súborom a XLS súborom a prácou s nimi.

A. UKLADANIE A NAČÍTAVANIE DÁT DO/Z MAT SÚBOROV

Ukladanie dát

Na ukladanie dát do súboru bola v Matlabe vytvorená funkcia *save*. Táto funkcia ukladá premenné do binárneho súboru. Meno súboru do ktorého sa dáta uložia vieme ovplyvniť tým, že za názvom funkcie dopíšeme nami zvolený názov súbor. V prípade, že názov ne zadáme, Matlab nám premenné uloží do súboru *matlab.mat*. Keď chceme do súboru uložiť len niektoré premenné, je potrebné ich špecifikovať za názvom súboru, v opačnom prípade Matlab uloží do súboru všetky premenné Workspace.

Uloženie do súboru	matlab.mat	nazov.mat	nazov.mat
Uloženie premenných	všetkých	všetkých	b, c, d
Príkaz	a=1; b=4; c=3.8; d=b*c; save	a=1; b=4; c=3.8; d=b*c; save nazov.mat	a=1; b=4; c=3.8; d=b*c; save nazov.mat b c d

Pri zadávaní premenných, ktoré chceme uložiť je možné použiť aj tzv. žolíky označované ako *. Premenné, ktoré majú časť svojho názvu rovnakú nemusíme vypisovať jednotlivo, ale využijeme spomínaného žolíka.

```
a=1; prem1=4; prem2=2.1; b=6; prem3=0.3; prem4=6;
save nazovsuboru.mat prem*
```

Takto zadaný príkaz nám uloží do súboru *nazovsuboru.mat* len premenné *prem1*, *prem2*, *prem3*, *prem4*.

Matlab nám takto zadaným príkazom uloží dáta v binárnom formáte. Ďalšími príkazmi vieme zmeniť aj formát a to nasledovne :

```
save nazovsuboru.mat % štandardný binárny formát
save nazovsuboru.mat -ascii % 8-bitový textový výstup
save nazovsuboru.mat -ascii -tabs % 8-bitový textový výstup oddelený tabulátormi
save nazovsuboru.mat -ascii -double % 16-bitový textový výstup
save nazovsuboru.mat -ascii -double -tabs % 16-bitový textový výstup oddelený tabulátormi
```

Načítavanie dát

Import dát z Mat súborov je veľmi jednoduchý pomocou príkazu *load*. Tak ako pri ukladaní aj pri načítavaní môžeme načítať celý súbor, len určité premenné alebo využiť žolíka.

```
load nazovsuboru.mat
load nazovsuboru.mat prem1 prem2
load nazovsuboru.mat prem*
```

Predtým ako si dáta zo súboru načítame, môžeme sa pozrieť aké dáta v danom súbore sú uložené. Príkaz *whos* s prepínačom *file* a názvom súboru nám zobrazí názov premenných, veľkosť, počet bytov a triedu.

```
>> whos -file nazovsuboru.mat
Name           Size           Bytes  Class  Attributes
premi          1x1             8  double
premi2         1x1             8  double
premi3         1x1             8  double
premi4         1x1             8  double
```

B. UKLADANIE A NAČÍTAVANIE DÁT DO/Z TEXTOVÝCH SÚBOROV

Ukladanie dát

Matlab má niekoľko vstavaných funkcií pre ukladanie textových súborov. Ich použitie závisí od množstva ukladaných dát a od formátu súboru, do ktorého chceme dáta uložiť. Pri binárnych súboroch MAT sme využívali najjednoduchšiu funkciu *save*. Táto funkcia taktiež umožňuje ukladanie do ASCII súborov. Pri takomto uložení dát sa ako oddeľovač používa medzera.

```
a=2; b=3.1; C=[1,2,3;6,8,0];
save nazovsuboru.out -ASCII
```

Funkcia *dlmwrite* tiež slúži na ukladanie dát do textového súboru. Na rozdiel do funkcie *save* si pri tejto funkcii môžeme sami zvoliť znak slúžiaci na oddeľovanie dát. Oddeľovač sa do príkazu pridáva v úvodzovkách.

```
A=[1 2 8; 6 3 7];
dlmwrite('data.out',A, ';')
```

Ďalšou funkciou slúžiacou na ukladanie textových súborov je funkcia *csvwrite*, ktorá je primárne určená pre dáta využívajúce tabuľkový procesor. Pri tejto funkcii sa oddeľovač nešpecifikuje, je to vždy čiarka.

```
A=[1 2 8; 6 3 7];
csvwrite('csvdata.out',A)
```

Poslednou funkciou na ukladanie dát do textového súboru je **diary**. Ukladá výstup príkazového riadku Matlabu.

```
diary data.out
A=[1 5 8 ; 2 4 7; 2 0 1];
A
diary off
type data.out
```

Príkazom uvedeným v poslednom riadku vypíšeme obsah súboru *data.out* do príkazového riadku.

Načítavanie dát

Na načítavanie dát taktiež existuje niekoľko vstavaných funkcií. Výber funkcie, ktorá sa na import dát použije závisí od naformátovanie dát v súbore. Textové dáta musia byť naformátované do rovnakého počtu stĺpcov v jednotlivých riadkoch. Ako oddeľovač dát slúži tzv. oddeľovací znak.

Najjednoduchší spôsob načítania dát je pomocou funkcie **load**. Táto funkcia sa využíva v prípade keď sú dáta uložené do obdĺžnika (v každom riadku je rovnaký počet stĺpcov). Dáta budú vložené do Workspace s menom totožným ako je názov súboru bez koncovky. Dáta je možné uložiť aj s názvom, ktorý si užívateľ zvolí sám (prípad v 2. riadku - dáta sa uložia do premennej A)

```
load data.txt
A=load('data.txt')
```

V prípade, že máme ako oddeľovač použitý iný znak ako je medzera, môžeme na načítanie dát použiť funkciu **dlmread**. Táto funkcia ignoruje medzery medzi dátami. Ako druhý parameter je potrebné zadať znak oddeľovača. Nepovinnými parametrami je posun, od ktorého chceme začať načítavanie. Indexovať sa začína od 0,0. V druhom riadku príkladu je uvedený príkaz na načítanie dát od prvku nachádzajúceho sa v 3. riadku 2. stĺpci.

```
A = dlmread('data.txt', ',');
A = dlmread('data.txt', ', ', 3, 2);
```

Čítanie dát z textových procesorov uložených v csv súboroch sa realizuje funkciou **csvread**. V takýchto súboroch ako oddeľovač slúži čiarka, preto sa v príkaze parameter oddeľovača nezadáva. Zvyšné parametre zostávajú také isté ako pri funkcii **dlmread**.

```
A=csvread('data.txt', 2, 3);
```

V prípade, že textový súbor obsahuje hlavičky k dátam, je potrebné použiť funkciu **textscan**, v tejto funkcii sa môže využiť parameter **headerlines**, ktorý umožňuje ignorovanie zadaného počtu riadkov od začiatku súboru.

Majme súbor data.txt, v ktorom sú nasledovné dáta :

Stĺpec1	Stĺpec2	Stĺpec3	Stĺpec4	Stĺpec5
2.154	1.256	3.785	24.125	2.536
3.256	1.401	2.869	36.254	1.258
3.987	1.569	3.028	12.548	3.685

V takomto prípade je potrebné najskôr otvoriť súbor na čítanie. Identifikátor súboru si vložíme do premennej idenf. Funkciou *textscan* načítame 3 riadky, znakom *%f* udávame, že chceme aby sa údaje uložili v desatinnom formáte. Ďalej využijeme parameter *headerlines* a za ním udáme číslo, ktoré nám označuje počet riadkov, ktoré sú v súbore uvedené ako hlavičky. Súbor je potrebné zavrieť a to príkazom *fclose*.

```
identif = fopen('data.txt','r');
data = textscan(identif, '%f %f %f %f %f',3,'headerlines',1);
fclose(identif);
```

Nie všetky textové súbory musia obsahovať výhradne číselné premenné. Uvedieme si príklad súboru v ktorom sa nachádzajú premenné rôzneho dátového typu. Majme súbor data.txt s nasledujúcim obsahom:

Vysoký	elektrikár	512.65	2	12	Áno
Chladná	lekárka	1254.35	1	10	Áno
Hravý	účtovník	845.23	2	15	Nie
Nováková	učiteľka	658.12	3	20	Áno

Takto usporiadané údaje je možné načítať pomocou funkcie *textread* takto:

```
>> [meno, profesia, mzda, deti, odpracovane_roky, odpoved]=...
textread('data.txt','%s %s %f %d %d %s', 4)
```

Údaje zo súboru sa rozdelia do jednotlivých premenných - meno, profesia, mzda....

Typy týchto premenných určujú prepínače (%s - reťazec znakov, %f - reálne číslo, %d - celé číslo). Posledný parameter udáva počet riadkov, ktoré sa majú spracovať.

C. UKLADANIE A NAČÍTAVANIE DÁT DO/Z XLS SÚBOROV

Ukladanie dát

Funkcia *xlswrite* slúži na ukladanie dát vo formáte xls. Pri použití funkcie je potrebné zadať názov súboru, do ktorého sa budú dáta ukladať a maticu MxN dát, ktorá sa bude ukladať. Matlab má obmedzenie pre veľkosť takejto matice. Číslo M nesmie prekročiť hodnotu 65536 a číslo N nesmie prekročiť hodnotu 256. Matica môže obsahovať číselne a reťazcové hodnoty. Údaje sa zapíšu do prvého hárka na pozíciu A1. Pridaním parametrov vieme ovplyvniť názov hárka, do ktorého sa údaje uložia a pozíciu buniek od ktorej po ktorú sa údaje budú zapisovať (prvá bunka

označuje ľavý horný roh, druhá označuje pravý dolný roh). Obidve spomínané parametre zadávame ako reťazec.

```
M={'júl','teplota vzduchu','teplota vody';...  
1,31,23;2,29,22;3,33,24;4,31,25;5,30,21;6,34,24};  
xlswrite('Zosit.xls',M);  
xlswrite('Zosit.xls',M,'júl');  
xlswrite('Zosit.xls',M,'júl','B3:D8');
```

Načítavanie dát

Na načítavanie dát zo súboru xls bola vytvorená funkcia *xlsread*. Táto funkcia je určená na čítanie číselných údajov. Funkcia vracia maticu čísel dátového typu double.

```
A=xlsread('Názov_súboru.xls');  
% načítajú sa údaje zo súboru Názov_súboru  
% z prvého hárku od bunky A1  
B=xlsread('Názov_súboru.xls','názov_hárku');  
% načítajú sa údaje zo súboru Názov_súboru  
%z hárka názov_hárku od bunky A1  
C=xlsread('Názov_súboru.xls','názov_hárku','B3:D8');  
% načítajú sa údaje zo súboru Názov_súboru  
%z hárka názov_hárku od bunky B3 po bunku D8
```

T3 - 2. APROXIMÁCIA

Skúma funkčný vzťah (priebeh závislosti), podľa ktorého sa mení závislá premenná Y pri zmenách nezávislých veličín x_1, x_2, \dots, x_k .

Cieľom aproximácie je vyšetriť závislosť dvoch a viacerých premenných. Jej úlohou je nájsť vhodnú regresnú funkciu - funkčný predpis.

Regresná analýza pomáha :

- Riadiť a predpovedať chovanie sledovaných premenných
- Predpovedať hodnoty výstupných premenných aj tam, kde na výpočet nebolo dostatočné množstvo dát
- Zistiť body, ktoré sa výrazne odlišujú od očakávaného výsledku

Aproximácia funkcie priamkou - metóda najmenších štvorcov

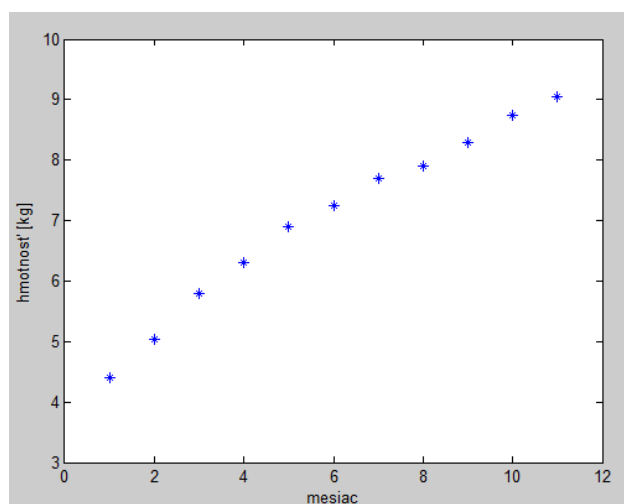
Majme množinu usporiadaných bodov $[x_i, y_i]$. táto metóda spočíva v hľadaní priamky $y=a_0 + a_1x$ s takými parametrami a_0, a_1 priamky, aby súčet druhých mocnín (štvorcov) vzdialeností jednotlivých bodov od tejto priamky bol čo najmenší. Pomocou metódy najmenších štvorcov dostaneme priamku využitím nasledujúcich rovníc pre aproximáciu:

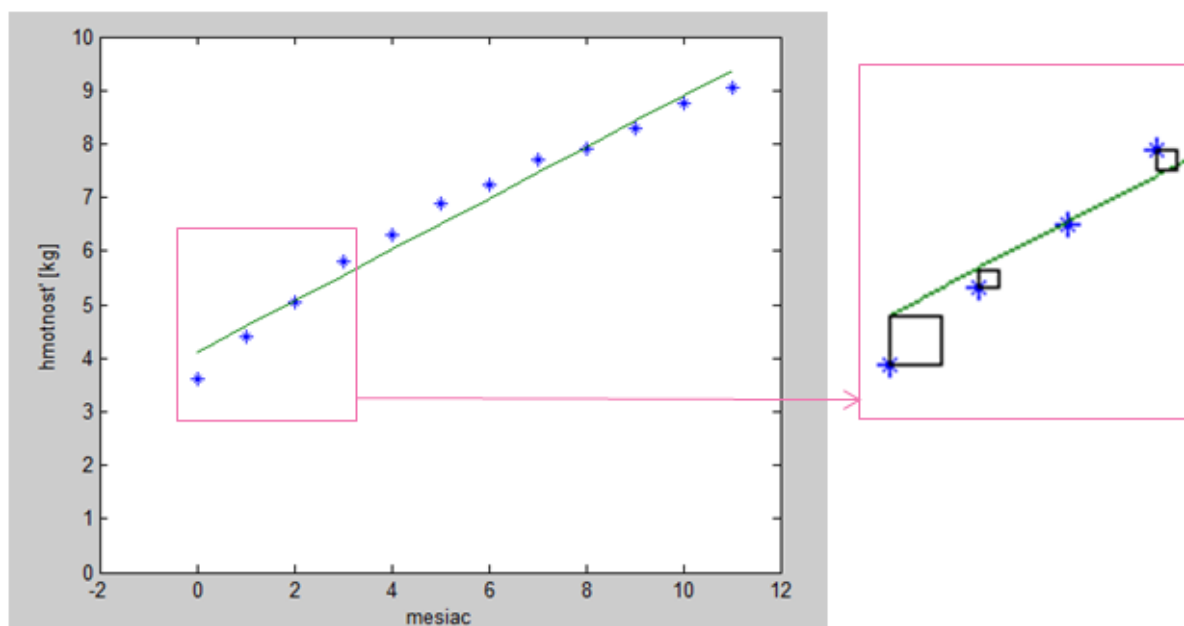
$$a_0(n+1) + a_1 \sum_{i=0}^n x_i = \sum_{i=0}^n y_i$$

$$a_0 \sum_{i=0}^n x_i + a_1 \sum_{i=0}^n x_i^2 = \sum_{i=0}^n x_i y_i$$

Príklad - V tabuľke je uvedená meniac sa hmotnosť dieťaťa vzhľadom na vek dieťaťa v prvých 12-tich mesiacoch. Našou úlohou bude zistiť aká je závislosť medzi hmotnosťou [kilogram] dieťaťa a jeho vekom [mesiac]. Tabuľkové hodnoty vieme vložiť do grafu.

Mesiac (x)	Hmotnosť dieťaťa (y)
0.	3,6
1.	4,4
2.	5,03
3.	5,8
4.	6,3
5.	6,9
6.	7,25
7.	7,7
8.	7,9
9.	8,3
10.	8,75
11.	9,05





Riešenie

$$12a_0 + 67a_1 = 80,98$$

$$67a_0 + 580,1784a_1 = 513,91$$

Následne vypočítame jednotlivé hodnoty parametrov priamky:

$$a_0 = 0,4791$$

$$a_1 = 4,11295$$

Výsledná priamka je v tvare:

$$y = 0,4791x + 4,11295$$

V Matlabe môžeme tvar priamky nájsť nasledovným príkazom :

Command Window

```
>> x=0:11;
>> y=[3.6 4.4 5.03 5.8 6.3 6.9 7.25 7.7 7.9 8.3 8.75 9.05];
>> f=polyfit(x,y,1)
```

f =

```
0.4792    4.1129
```

Aproximácia funkcie polynómom

Niektoré závislosti x a y nie je možné preložiť priamkou, ale je potrebné preložiť ich polynómom. Pri aproximácii funkcie polynómom n -tého stupňa je analytické riešenie zdĺhavé. Pri aproximácii funkcie polynómom druhého stupňa metódu najmenších štvorcov pre ručný výpočet sú potrebné nasledovná vzorce:

$$a_0(n+1) + a_1 \sum_{i=0}^n x_i + a_2 \sum_{i=0}^n x_i^2 = \sum_{i=0}^n y_i,$$

$$a_0 \sum_{i=0}^n x_i + a_1 \sum_{i=0}^n x_i^2 + a_2 \sum_{i=0}^n x_i^3 = \sum_{i=0}^n x_i y_i,$$

$$a_0 \sum_{i=0}^n x_i^2 + a_1 \sum_{i=0}^n x_i^3 + a_2 \sum_{i=0}^n x_i^4 = \sum_{i=0}^n x_i^2 y_i.$$

Pre aproximáciu funkcie vyšším stupňom sa vzorce ešte viac rozširujú.

V Matlabe je možné zadané (namerané) údaje aproximovať polynómom n -tého stupňa. Ide o nájdenie takých koeficientov polynómu, ktoré zabezpečia funkčné hodnoty polynómu čo najbližšie k funkčným hodnotám aproximovanej funkcie. Vykonáva sa to pomocou funkcie **polyfit**, ktorá používa metódu najmenších štvorcov. Jej syntax je nasledovná

$$\text{polyfit}(x, y, n),$$

kde x , y predstavujú súradnice bodov, ktoré aproximujeme a n je najvyšší stupeň polynómu ktorým aproximujeme.

Pri aproximácii je taktiež potrebná funkcia **polyval**, vypočíta funkčnú hodnotu pre nami zvolenú hodnotu premennej x .

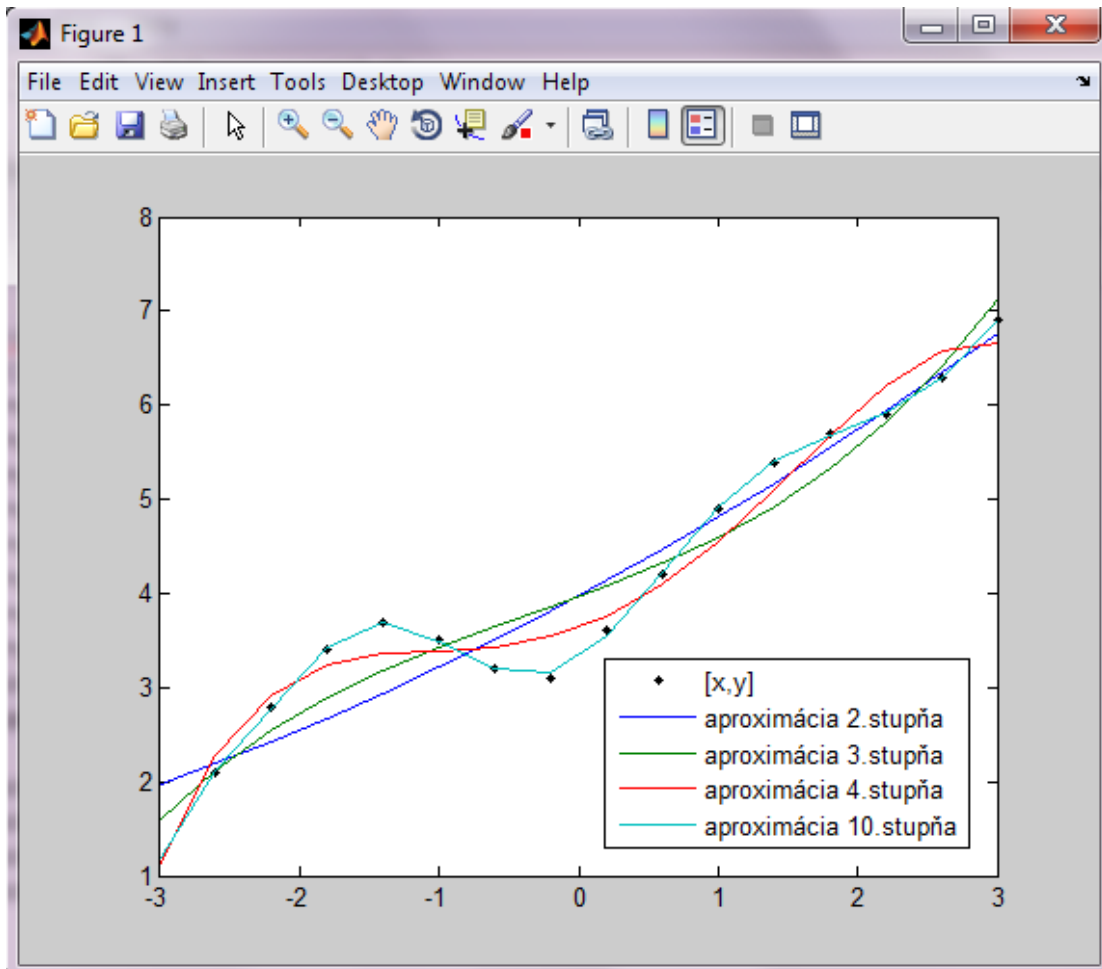
$$\text{polyval}(p, x),$$

kde p je polynóm.

Riešený príklad - Aproximujte údaje x a y polynómami 2., 3., 4., 5. stupňa. Vykreslite do grafu príslušné priebehy. Vypočítajte chybu odchýliek pre každú aproximáciu. Pričom $x=-3:0.4:3$, $y=[1.2 \ 2.1 \ 2.8 \ 3.4 \ 3.7 \ 3.5 \ 3.2 \ 3.1 \ 3.6 \ 4.2 \ 4.9 \ 5.4 \ 5.7 \ 5.9 \ 6.3 \ 6.9]$.

```
x=-3:0.4:3;
y=[1.2 2.1 2.8 3.4 3.7 3.5 3.2 3.1 3.6 4.2 4.9 5.4 5.7 5.9 6.3 6.9];
a1=polyfit(x,y,2);
a2=polyval(a1,x);
b1=polyfit(x,y,3);
b2=polyval(b1,x);
c1=polyfit(x,y,4);
c2=polyval(c1,x);
d1=polyfit(x,y,10);
d2=polyval(d1,x);
plot(x,y,'k.',x,a2,x,b2,x,c2,x,d2)
legend(' [x,y] ', 'aproximácia 2.stupňa', 'aproximácia 3.stupňa', ...
      'aproximácia 4.stupňa', 'aproximácia 10.stupňa')
odchylka2=sum(sum(y-a2).^2);
odchylka3=sum(sum(y-b2).^2);
odchylka4=sum(sum(y-c2).^2);
odchylka10=sum(sum(y-d2).^2);
```

Tutoriál4 : Aplikačné využitie práce so súbormi v úlohách regresnej analýzy v prostredí Matlab



T3 - 3. INTERPOLÁCIA

Interpolácia

Hlavnou myšlienkou interpolácie je nájsť, funkciu (polynóm) $P_n(x)$, ktorá sa bude zhodovať, s funkciou $f(x)$ v n rôznych uzlových bodoch x_i , tj.

$$P_n(x) = f(x_i) = f_i = y_i \quad i = 0; 1; \dots; n$$

Lagrangeov interpolačný polynóm

Pre $i=0, 1, \dots, n$ má polynóm $L_n(x)$ tvar:

$$L_n(x) = \sum_{i=0}^n \frac{(x - x_0) \dots (x - x_{i-1})(x - x_{i+1}) \dots (x - x_n)}{(x_i - x_0) \dots (x_i - x_{i-1})(x_i - x_{i+1}) \dots (x_i - x_n)} f(x_i)$$

Príklad

Majme zadané hodnoty funkcie $f : \langle 1;4 \rangle \rightarrow \mathbb{R}$ tabuľkou:

x	1	2	4
f(x)	1	4	16

Nahradením funkcie pomocou polynómu $L_2(x)$ sa vypočíta približná hodnota funkcie f v bode $x=3$ pomocou $L_2(3)$

$$L_2(x) = \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} \cdot f(x_0) + \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} \cdot f(x_1) + \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)} \cdot f(x_2).$$

Dosadením konkrétnych hodnôt do rovnice dostaneme vzťah:

$$L_2(3) = \frac{(3 - 2)(3 - 4)}{(1 - 2)(1 - 4)} \cdot 1 + \frac{(3 - 1)(3 - 4)}{(2 - 1)(2 - 4)} \cdot 4 + \frac{(3 - 1)(3 - 2)}{(4 - 1)(4 - 2)} \cdot 16 = 9.$$

Pre interpoláciu funkčných hodnôt y v závislosti na x ($y=f(x)$) existuje v Matlabe funkcia `interp1`. Pre interpoláciu funkčných hodnôt z v závislosti od x,y ($z=f(x,y)$) sa využíva funkcia `interp2`. Pre závislosť $v=f(x,y,z)$ je k dispozícii funkcia `interp3`. Budeme sa zaoberať možnosťami interpolácie v rovine (2D) a v priestore (3D).

Štruktúra funkcia ***interp1*** je:

```
interp1(x, y, xi, 'metóda'),
```

Tutoriál4 : Aplikačné využitie práce so súbormi v úlohách regresnej analýzy v prostredí Matlab

kde x a y sú vektory predstavujúce interpolované dáta, x_i je vektor predstavujúci body interpolácie. Parameter ,metóda' nie je povinný, predstavuje akou metódou sa má interpolácia vykonať.

Možnosti metód:

- ,nearest' - interpolácia susedných bodov
- ,linear' - lineárna interpolácia
- ,spline' - kubická splajnová interpolácia
- ,cubic' - štvorcová interpolácia

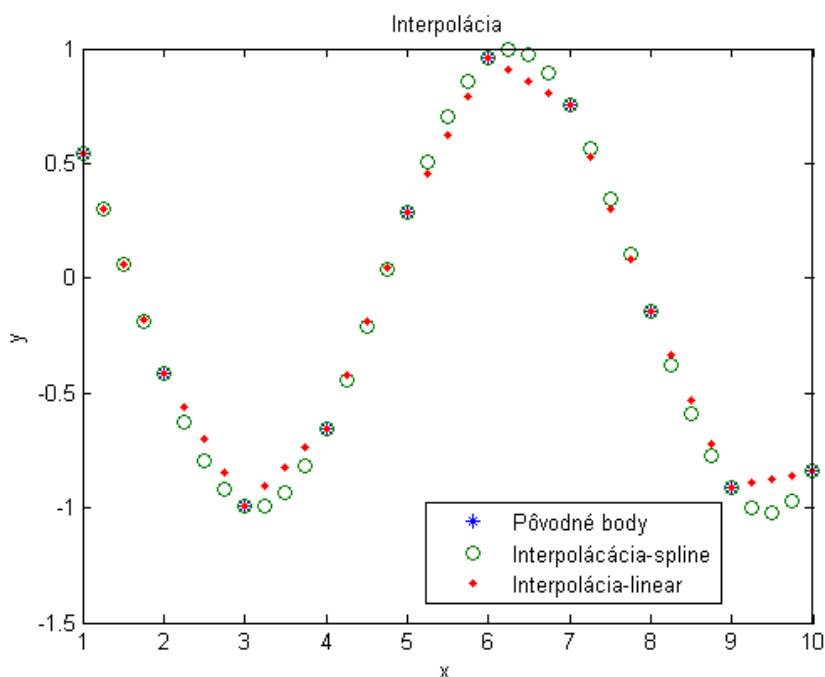
Štruktúra funkcia *interp2* je:

```
interp2(x,y,z,xi,yi,'metóda'),
```

kde z je premenná hodnoty dvojrozmernej interpolácie, x a y sú súradnice interpolácie, x_i a y_i súbor bodov interpolácie. Pre parameter ,metóda' platí to, čo pri funkcii *interp1*.

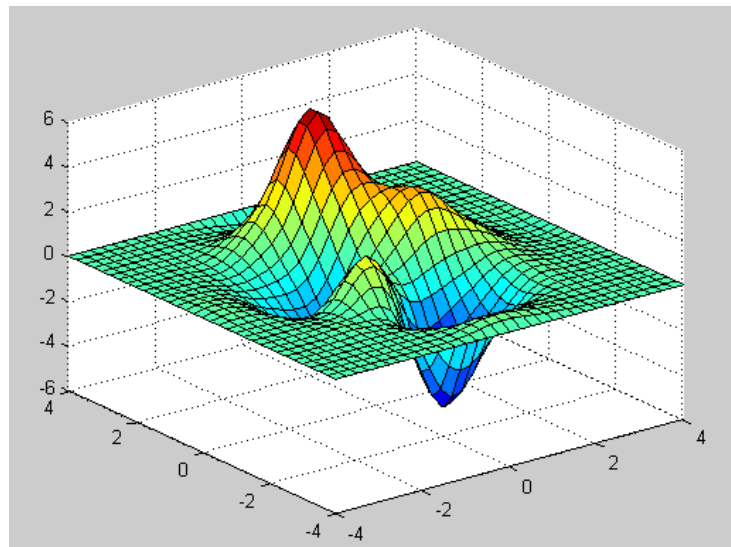
Riešený príklad - Interpolujte dáta x a y so štvornásobnou periódou vzorkovania (so štvrtinovým krokom). Výsledok vykreslite v grafe. Kde $x=0:10$, $y=\cos(x)$.

```
x=1:10;
y=cos(x);
xi=1:0.25:10;
yi1=interp1(x,y,xi,'spline');
yi2=interp1(x,y,xi,'linear');
plot(x,y,'*',xi,yi1,'o',xi,yi2,'.')
title('Interpolácia')
xlabel('x')
ylabel('y')
legend('Pôvodné body', 'Interpolácia-spline', 'Interpolácia-linear')
```



Riešený príklad - Vytvorte matice x a y ktoré budú obsahovať sieť hodnôt v rozmedzí od $\langle -3;3 \rangle$. Sieť pre interpoláciu vytvorte s krokom 0,25. Dáta interpolujte druhým stupňom interpolácie. Výsledok zobrazte v grafe pomocou `surf()`.

```
[x,y]=meshgrid(-4:1:4);  
z=peaks(x,y);  
% Vytvorenie siete pre interpoláciu:  
[xi,yi]=meshgrid(-4:0.25:4);  
% Interpolácia štvorcovou metódou  
zi1=interp2(x,y,z,xi,yi,'cubic');  
surf(xi,yi,zi1)
```



T3 - 4. PRÍKLADY NA SAMOTATNÉ RIEŠENIE

1. Naplňte premenné *prem1*, *prem2*, *prem3*, *prem4*, *prem5*, *a*, *b*, *c*, *d* číselnými hodnotami. Uložte vytvorené premenné *prem1*, *prem2*, *prem3*, *prem4*, *prem5*, *b*, *d* do súboru *uloha1.mat*. Pri zadávaní premenných využite žolíka.
2. Vyčistite Workspace. Zo súboru *uloha1.mat* načítajte všetky premenné.
3. Vyčistite Workspace. Zo súboru *uloha1.mat* načítajte nasledovné premenné - *a,c,d,prem1*.
4. Vytvorte maticu 4x2. Uložte ju do textového súboru , v ktorom ako oddeľovač bude slúžiť čiarka.
5. Vyčistite Workspace. Do premennej *A* načítajte obsah súboru, ktorý ste vytvorili v predošlej úlohe.
6. Vytvorte si textový súbor, ktorý bude obsahovať „tabuľku“ údajov o vašich spolužiakoch, a to : meno, evidenčné číslo, počet zapísaných predmetov v minulom semestri, počet úspešne zvládnutých predmetov v minulom semestri, odpoveď na otázku, či sú spokojný so svojim výberom vysokej školy. Načítajte tieto údaje do Workspace.
7. Vytvorte súbor v programe Microsoft Excel, do ktorého vložíte informácie o meniacom sa kurze americký dolár vzhľadom na eura v čase 1.2.2013-15.2.2013 (1.stĺpec - poradové číslo dňa v mesiaci február, 2.stĺpec - predaj, 3.stĺpec - nákup). Načítajte tieto údaje do Workspace.
8. Pomocou ľubovoľného príkazu na zobrazovanie grafov zobrazte funkciu $z=f(2x^2-y^3)$, pričom $x=3:20$ a $y=-3:-20$. Graf popíšte.
9. POTREBNÉ PRE ĎALŠIE PRÍKLADY
 - A) Vytvorte súbor *data1*, kde uložíte premenné:
 $x = [-5, -4.5, -3, -2, 0.5, 2.5, 6, 10, 12.5, 16, 20.5, 22]$
 $y = [0.257, 0.968, 1.254, 1.685, 2.002, 2.685, 2.986, 3.574, 3.867, 4.012, 3.986, 4.876]$.
 - B) Vytvorte súbor *data2*, kde uložíte premenné:
 $x = [-4, -2, 0, 2, 4, 6, 8, 10, 12, 14, 15, 16, 17, 18, 19, 20, 21]$
 $y = [-0.112, -0.058, 0.218, 0.294, 0.351, 0.486, 0.687, 0.788, 1.007, 1.315, 1.419, 1.225, 1.254, 1.321, 1.457]$
10. Aproximujte údaje zo súboru *data1* polynómami 2., 3. a 4. stupňa. Vykreslite do grafu príslušné priebehy. Vypočítajte sumu štvorcov odchýliek pre každú aproximáciu.
11. Aproximujte údaje zo súboru *data1* polynómom *n*- tého stupňa, pričom *n* sa zadá z klávesnice. Aproximované údaje uložte do súboru. Vykreslite do grafu priebeh aproximácie.
12. Interpolujte namerané dáta so súboru *data2*. Použite tretinový krok vzorkovania a na interpolovanie použite metódu 'cubic' a 'linear'. Výsledky zobrazte v dvoch grafoch ktoré budú vedľa seba.
13. Načítajte dáta z súboru *data1*. Tieto hodnoty interpolujte zo štvrtinovým krokom vzorkovania a použite na to funkciu *interp*. Následne interpolujte dáta pomocou metódy 'spline' a zobrazte výsledok v grafe.