

Numerične metode z Matlabom

Reševanje sistemov linearnih enačb in iskanje lastnih vrednosti

- Matrični sistem $A\mathbf{x} = \mathbf{b}$, kjer je $A \in \mathbb{R}^{m \times n}$, $\mathbf{b} \in \mathbb{R}^{m \times 1}$ in $\mathbf{x} \in \mathbb{R}^{n \times 1}$, v splošnem rešimo z ukazom $\backslash$ (backslash):

$$\mathbf{x} = A \backslash \mathbf{b}$$

Če je matrika nesingularna, je to algebraično (ne pa numerično!!!) ekvivalentno $\mathbf{x} = A^{-1}\mathbf{b}$.

Če je $m > n$ in je A polnega ranga, potem $\backslash$ vrne rešitev po metodi najmanjših kvadratov, torej tisti $\mathbf{x}$ za katerega je dosežen minimum

$$\min_{\mathbf{x} \in \mathbb{R}^n} \|A\mathbf{x} - \mathbf{b}\|_2.$$

- Pomemben ukaz je tudi `lu`, ki izračuna ***LU razcep z delnim pivotiranjem***. Za nesingularno matriko $A \in \mathbb{R}^{n \times n}$ je

$$PA = LU,$$

kjer je ***P*** permutacijska matrika, ***L*** spodnja trikotna matrika z enkami na diagonalni in ***U*** zgornja trikotna matrika. Klic je oblike

$$[L,U,P]=lu(A)$$

- Za simetrično pozitivno definitno matriko $A \in \mathbb{R}^{n \times n}$ je ekonomičneje uporabiti ***razcep Choleskega***

$$A = V^T V,$$

kjer je ***V*** zgornja trikotna matrika. Klic je oblike

$$V=chol(A)$$

- Za matriko $A \in \mathbb{R}^{m \times n}$ lahko izračunamo tudi t.i. **QR razcep**:

$$A = QR,$$

kjer je $Q \in \mathbb{R}^{m \times m}$ ortogonalna matrika ($Q^T Q = Q Q^T = I$) in $R \in \mathbb{R}^{m \times n}$ "zgornja trikotna" matrika. Klic je oblike

$$[Q, R] = \text{qr}(A)$$

- Med najpomembnejšimi je **singularni razcep**. Za matriko $A \in \mathbb{R}^{m \times n}$ je

$$A = U \Sigma V^T,$$

kjer sta $U \in \mathbb{R}^{m \times n}$ in $V \in \mathbb{R}^{n \times n}$ ortogonalni matriki ter $\Sigma \in \mathbb{R}^{m \times n}$ "diagonalna matrika" s singularnimi vrednostmi na diagonalni.

Klic je oblike

$$[U, S, V] = \text{svd}(A)$$

- Veliko pomembnih pojmov iz linearne algebre temelji na prejšnjih razcepih. Denimo **det** (determinanta), **rank** (rang), **null** (jedro), **orth** (ortogonalna baza), **det** (determinanta), ...
- Za kvadratno matriko $A \in \mathbb{R}^{n \times n}$ lahko poiščemo **lastne vrednosti** in **pripadajoče lastne vektorje**. Klic je oblike

$$[V,D]=\text{eig}(A)$$

Pri tem so stolpci matrike V lastni vektorji, diagonala matrike D pa lastne vrednosti.

Reševanje (sistemov) nelinearnih enačb

- Za reševanje ene enačbe uporabimo lahko ukaz **fzero**.
- Za reševanje sistema pa (če imamo "Optimization toolbox") **fsolve**.

Primer

Rešujemo enačbo

$$e^{-x} - x = 0$$

in sistem

$$\begin{aligned}x^2 + y^2 - 1 &= 0 \\ x^2 - y &= 0.\end{aligned}$$

- V prvem primeru uporabimo

```
f=inline('exp(-x)-x','x');  
xz=fzero(f,0.5);
```

v drugem pa

```
fsis=inline(' [X(1)^2+X(2)^2-1;X(1)^2-X(2)] ','X');  
xz=fsolve(fsis,[1;1]);
```

Seveda moramo poznati **dobre začetne približke**, da metoda res konvergira k željeni rešitvi.

Iskanje ekstremov skalarnih polj

- Lokalni minimum funkcije **ene spremenljivke**, torej

$$\min_{x \in [a,b]} f(x),$$

poiščemo z ukazom **fminbnd**. Klic je oblike

```
[xmin,fmin]=fminbnd(f,a,b)
```

Primer

Poiščimo minimum funkcije $f(x) = \sin x - \cos x$ na intervalu $[-\pi, \pi]$:

```
f=inline('sin(x)-cos(x)','x');  
[xmin,fmin]=fminbnd(f,-pi,pi)
```

- Lokalne **ekstreme funkcij več spremenljivk lahko poiščemo s funkcijo `fminsearch`**. Za funkcijo $F : \mathcal{D} \subseteq \mathbb{R}^n \rightarrow \mathbb{R}$ želimo torej poiskati

$$\min_{\mathbf{x} \in \mathcal{D}} F(\mathbf{x}).$$

Uporabimo klic oblike

```
[xmin,fmin]=fminsearch(F,x0)
```

Primer

Poiščimo minimum skalarne polja $f(x, y) = x^2 + y^2 - x y$ na domeni $[-1, 1] \times [-1, 1]$:

```
F=inline('X(1)^2+X(2)^2-X(1)*X(2)','X');  
[Xmin,Fmin]=fminbnd(f,[1;1])
```

- Podobna ukaza sta še `fminunc` in `fmincon`.

Interpolacija in aproksimacija s polinomi

- Tabelo podatkov (x_i, y_i) , $i = 0, 1, \dots, m$, **interpoliramo s polinomom p_n stopnje $\leq n \leq m$** z ukazom `polyfit`. Klic je oblike
`p=polyfit(x,y,n)`

Če je $n < m$, se uporabi **metoda najmanjših kvadratov**, torej poišče tak polinom p , za katerega je dosežen minimum

$$\min_{p \in \mathcal{P}_n} \sum_{i=0}^m (p(x_i) - y_i)^2,$$

kjer je $\mathcal{P}_n$ prostor polinomov stopnje ne več kot n .

- Matlab podpira tudi računanje z **zlepki**. Za tabelo podatkov iz prejšnje točke in predpisane abscise t_j , $j = 1, 2, \dots, N$, lahko **interpolacijski zlepek** dobimo z ukazom

`ft=interp1(x,y,t,metoda)`

Numerično integriranje

- Integralov

$$\int_a^b f(x) dx$$

pogosto ne moremo analitično izračunati.

- V Matlabu imamo na voljo dve standardni metodi **trapz** (sestavljeno trapezno pravilo) in **quad** (adaptivno Simpsonovo pravilo): Klica sta

```
I=trapz(x,y)
```

```
I=quad(f,a,b)
```

- Podobni ukazi so še **quadgk**, **quadl**, **dblquad**, **triplequad**,...

Reševanje navadnih diferencialnih enačb

- Rešujemo začetni problem

$$y'(x) = f(x), \quad y(x_0) = y_0.$$

Uporabimo lahko eno od vgrajenih Matlabovih funkcij, denimo **ode45**:

$$[x, y] = \text{ode45}(f, [x_0, b], y_0)$$

- Če je diferencialna enačba višjega reda,

$$y^{(k)}(x) = f(x, y(x), y'(x), \dots, y^{(k-1)}(x)),$$

$$y(x_0) = y_0, y'(x_0) = y'_0, \dots, y^{(k-1)}(x_0) = y_0^{(k-1)}.$$

jo **prevedemo na sistem enačb prvega reda**:

Uvedemo nove spremenljivke $y_1 = y, y_2 = y', \dots, y_k = y^{(k-1)}$.
Zapišemo sistem

$$y_1'(x) = y_2(x)$$

$$y_2'(x) = y_3(x)$$

$$\vdots$$

$$y_k'(x) = f(x, y_1(x), \dots, y_{k-1}(x)),$$

z začetnimi pogoji $y_1(x_0) = y_0, y_2(x_0) = y_0', \dots, y_k(x_0) = y_0^{(k-1)}$.

Primer

Numerično izračunajte višino in hitrost pri navpičnem padanju telesa, na katerega delujeta sili teže in upora. Upoštevajte kvadratični zakon upora. Podani sta začetna višina in začetna hitrost.

Odgovorite na nekaj naslednjih vprašanj:

- a) Kakšna je hitrost telesa po dolgem času?*
- b) Kdaj telo pade na tla?*
- c) Kdaj je telo na polovici začetne višine?*

Namig: Izpeljite diferencialno enačbo drugega reda, jo prepisite v sistem prvega in ga rešite z Matlabovo funkcijo `ode45`. Odgovore na zgornja vprašanja poičite na podlagi rezultatov. Morda boste morali uporabiti še kakšne metode iz reševanja nelinearnih enačb, ...