

Numerička matematika

7. predavanje

Saša Singer

`singer@math.hr`

`web.math.hr/~singer`

PMF – Matematički odjel, Zagreb

Sadržaj predavanja

- Interpolacija splajnovima (nastavak):
 - Po dijelovima kubična kvazihermiteova interpolacija.
 - Kubični splajn i neprekidnost druge derivacije.
 - Razne vrste rubnih uvjeta.
 - Ocjene pogreške za kubični splajn.
- Metoda najmanjih kvadrata:
 - Diskretni problem najmanjih kvadrata.
 - Normalne jednadžbe.
 - Linearizacija.
 - Matrična formulacija problema najmanjih kvadrata.

Informacije — kolokvij, zadaće

Službeni termin prvog kolokvija je:

• utorak, 13. 4., u 12 sati.

“Oživile” su i domaće zadaće iz NM.

• Realizacija ide “automatski” — preko web aplikacije.

Pogledajte na službeni web kolegija, pod “zadaće”.

• Tamo su početne upute.

Skraćeni link je

<http://web.math.hr/nastava/unm/zadace.php>

Direktni link na aplikaciju za zadaće je

<http://degiorgi.math.hr/nm/>

Informacije

Moja web stranica za Numeričku matematiku je

http://web.math.hr/~singer/num_mat/

Tamo su kompletna predavanja od prošle dvije godine, a stizati će i nova (kako nastaju).

🕒 Prva 2 su još nesređena — onako kako ste ih vidjeli!

Skraćena verzija skripte — 1. dio (prvih 7 tjedana):

http://web.math.hr/~singer/num_mat/num_mat1.pdf

Skraćena verzija skripte — 2. dio (drugih 6 tjedana):

http://web.math.hr/~singer/num_mat/num_mat2.pdf

Informacije — demonstratori

Kolegij “**Numerička matematika**” ima čak **3 demonstratora**:

- Ervin Duraković, Nika Kenda i Marin Mišur.

Za upute za dogovor i termine demonstratura

- pogledajte **oglase** na oglasnoj ploči i

- **web** stranicu kolegija — pod “**nastava**”.

Po dijelovima kubična kvazihermiteova interpolacija

Po dijelovima kubična kvazihermiteova interp.

Sad se možemo vratiti problemu kako napraviti **po dijelovima kubičnu** Hermiteovu interpolaciju, ako **nemamo** zadane derivacije.

U tom slučaju

- derivacije možemo **aproksimirati** na različite **načine**,
- a samu interpolaciju zvat ćemo **kvazihermiteova** po dijelovima kubična interpolacija.

Napomena. U slučaju **aproksimacije** derivacije, **greška** po dijelovima kubične interpolacije **ovisi** o tome

- koliko je “**dobra**” aproksimacija derivacije.

Podijeljene razlike unaprijed

Najjednostavnije je uzeti **podijeljene razlike** kao aproksimacije derivacija u čvorovima. One mogu biti

- **unaprijed** (do na posljednju), ili
- **unazad** (do na prvu).

Ako koristimo podijeljene razlike **unaprijed**, onda je

$$s_k = \begin{cases} \frac{f_{k+1} - f_k}{x_{k+1} - x_k}, & \text{za } k = 0, \dots, n-1, \\ \frac{f_n - f_{n-1}}{x_n - x_{n-1}}, & \text{za } k = n. \end{cases}$$

Podijeljene razlike unazad

a ako koristimo podijeljene razlike unazad, onda je

$$s_k = \begin{cases} \frac{f_1 - f_0}{x_1 - x_0}, & \text{za } k = 0, \\ \frac{f_k - f_{k-1}}{x_k - x_{k-1}}, & \text{za } k = 1, \dots, n. \end{cases}$$

Međutim, greška koju smo napravili takvom aproksimacijom derivacije je reda veličine

• $O(h)$ u derivaciji,

• odnosno $O(h^2)$ u funkcijskoj vrijednosti,

što je dosta loše.

Simetrične razlike kao aproksimacije derivacije

Ako su točke x_k ekvidistantne, možemo koristiti **simetričnu razliku** (osim na lijevom i desnom rubu, gdje to nije moguće). Uz oznaku $h = x_k - x_{k-1}$, imamo

$$s_k = \begin{cases} \frac{f_1 - f_0}{h}, & \text{za } k = 0, \\ \frac{f_{k+1} - f_{k-1}}{2h}, & \text{za } k = 1, \dots, n-1, \\ \frac{f_n - f_{n-1}}{h}, & \text{za } k = n. \end{cases}$$

Greška obzirom na obične podijeljene razlike

- ☛ će se **popraviti** tamo gdje se koristi simetrična razlika, ali
- ☛ **najveće** greške ostaju na **prvom** i **zadnjem** podintervalu.

Besselova aproksimacija derivacija

Postoje i **bolje** aproksimacije derivacija, a pripadni kvazihermiteovi kubični polinomi obično dobivaju **ime** po načinu aproksimacije derivacija.

Na pr., derivaciju u točki x_k **aproksimiramo** tako da povučemo

- **kvadratni** interpolacijski polinom kroz x_{k-1} , x_k i x_{k+1} ,
- a zatim ga **deriviramo**.

Pripadna kvazihermiteova interpolacija zove se **Besselova** po dijelovima kubična interpolacija.

U **prvoj** i **posljednjoj** točki **ne možemo** postupiti tako (jer nema lijeve, odnosno, desne točke).

Besselova aproksimacija derivacija — sredina

Derivaciju u x_0 aproksimiramo tako da povučemo

- kvadratni interpolacijski polinom kroz x_0 , x_1 i x_2 , i
- njega deriviramo u x_0 .

Slično, derivaciju u x_n aproksimiramo tako da povučemo

- kvadratni interpolacijski polinom kroz x_{n-2} , x_{n-1} i x_n , i
- njega deriviramo u x_n .

U unutrašnjim čvorovima x_k , za $k = 1, \dots, n - 1$, dobivamo

$$p_{2,k}(x) = f_{k-1} + f[x_{k-1}, x_k] (x - x_{k-1}) \\ + f[x_{k-1}, x_k, x_{k+1}] (x - x_{k-1})(x - x_k),$$

Besselova aproksimacija derivacija — sredina

a zatim, deriviranjem i uvrštavanjem x_k

$$s_k = p'_{2,k}(x_k) = f[x_{k-1}, x_k] + f[x_{k-1}, x_k, x_{k+1}] (x_k - x_{k-1}).$$

Uz oznaku $h_k = x_k - x_{k-1}$, $k = 1, \dots, n$, prethodna se formula može napisati i kao

$$\begin{aligned} s_k &= f[x_{k-1}, x_k] + h_k \frac{f[x_k, x_{k+1}] - f[x_{k-1}, x_k]}{h_k + h_{k+1}} \\ &= \frac{h_{k+1} f[x_{k-1}, x_k] + h_k f[x_k, x_{k+1}]}{h_k + h_{k+1}}, \end{aligned}$$

tj. s_k je **težinska srednja vrijednost** podijeljene razlike unaprijed i unatrag.

Besselova aproksimacija derivacija — početak

Za $k = 0$ pripadni polinom je

$$p_{2,1}(x) = f_0 + f[x_0, x_1](x - x_0) \\ + f[x_0, x_1, x_2](x - x_0)(x - x_1).$$

Deriviranjem, pa uvrštavanjem x_0 dobivamo

$$s_0 = p'_{2,1}(x_0) = f[x_0, x_1] + f[x_0, x_1, x_2](x_0 - x_1) \\ = f[x_0, x_1] - h_1 \frac{f[x_1, x_2] - f[x_0, x_1]}{h_1 + h_2} \\ = \frac{(2h_1 + h_2) f[x_0, x_1] - h_1 f[x_1, x_2]}{h_1 + h_2}.$$

Besselova aproksimacija derivacija — kraj

Za $k = n$ pripadni polinom je

$$p_{2,n-1}(x) = f_{n-2} + f[x_{n-2}, x_{n-1}](x - x_{n-2}) \\ + f[x_{n-2}, x_{n-1}, x_n](x - x_{n-2})(x - x_{n-1}).$$

Deriviranjem, pa uvrštavanjem x_n dobivamo

$$s_n = p'_{2,n-1}(x_n) = f[x_{n-2}, x_{n-1}] + f[x_{n-2}, x_{n-1}, x_n](x_n - x_{n-2}) \\ + f[x_{n-2}, x_{n-1}, x_n](x_n - x_{n-1}) \\ = f[x_{n-2}, x_{n-1}] + (h_{n-1} + 2h_n) \frac{f[x_{n-1}, x_n] - f[x_{n-2}, x_{n-1}]}{h_{n-1} + h_n} \\ = \frac{-h_n f[x_{n-2}, x_{n-1}] + (h_{n-1} + 2h_n) f[x_{n-1}, x_n]}{h_{n-1} + h_n}.$$

Besselova aproksimacija derivacija — greška

Dakle, za Besselovu po dijelovima kubičnu interpolaciju stavljamo

$$s_k = \begin{cases} \frac{(2h_1 + h_2) f[x_0, x_1] - h_1 f[x_1, x_2]}{h_1 + h_2}, & k = 0, \\ \frac{h_{k+1} f[x_{k-1}, x_k] + h_k f[x_k, x_{k+1}]}{h_k + h_{k+1}}, & k = 1, \dots, n-1, \\ \frac{-h_n f[x_{n-2}, x_{n-1}] + (h_{n-1} + 2h_n) f[x_{n-1}, x_n]}{h_{n-1} + h_n}, & k = n. \end{cases}$$

Greška

- u derivaciji je reda veličine $O(h^2)$,
- što znači da je greška u funkciji reda veličine $O(h^3)$.

Akimina aproksimacija derivacija — sredina

Još jedna varijanta aproksimacije derivacija “s imenom”.

Akima je 1970. godine dao sljedeću aproksimaciju koja

- **usrednjava** podijeljene razlike,
- s ciljem da se spriječe **oscilacije** interpolacijske funkcije φ :

$$s_k = \frac{w_{k+1}f[x_{k-1}, x_k] + w_{k-1}f[x_k, x_{k+1}]}{w_{k+1} + w_{k-1}}, \quad k = 0, \dots, n,$$

uz

$$w_k = |f[x_k, x_{k+1}] - f[x_{k-1}, x_k]|$$

i

$$w_{-1} = w_0 = w_1, \quad w_{n-1} = w_n = w_{n+1}.$$

Akimina aproksimacija derivacija — početak

Za $k = 0$ i $k = n$, formule se ne mogu **direktno** iskoristiti, bez dodatnih definicija.

Kraćenjem svih težina w_k u formuli za $k = 0$ dobivamo da je

$$s_0 = \frac{f[x_{-1}, x_0] + f[x_0, x_1]}{2}.$$

Treba još definirati što je $f[x_{-1}, x_0]$. Podijeljenu razliku $f[x_0, x_1]$ možemo interpretirati kao **sredinu** dvije susjedne **podijeljene razlike**, tj.

$$f[x_0, x_1] = \frac{f[x_{-1}, x_0] + f[x_1, x_2]}{2}.$$

Akimina aproksimacija derivacija — kraj

Odatle slijedi da je

$$f[x_{-1}, x_0] = 2f[x_0, x_1] - f[x_1, x_2],$$

odnosno

$$s_0 = \frac{3f[x_0, x_1] - f[x_1, x_2]}{2}.$$

Za $h_1 = h_2$ dobivamo **isto** kao i kod **Besselove** aproksimacije.
Inače — **ne!**

Na sličan način, možemo dobiti i relaciju za s_n

$$s_n = \frac{3f[x_{n-1}, x_n] - f[x_{n-2}, x_{n-1}]}{2}.$$

Akimina aproksimacija derivacija — greška

Akimin algoritam je

- popularan u praksi,
- nalazi se u standardnim numeričkim paketima, poput IMSL-a, iako je točnost ovih formula za aproksimaciju derivacije relativno slaba.

Za neekvidistantne točke, greška

- u derivaciji je reda veličine samo $O(h)$,
- a to znači samo $O(h^2)$ za funkcijske vrijednosti.

Ako su točke ekvidistantne, onda je greška

- reda veličine $O(h^2)$ za derivaciju,
- a $O(h^3)$ za funkciju, tj. kao i kod Besselove po dijelovima kvazihermitske interpolacije.

Akimina aproksimacija derivacija — osnovni cilj

Slabija točnost je potpuno u skladu s **osnovnim ciljem** Akimine aproksimacije derivacija. U mnogim primjenama

- želimo dobiti **geometrijski** ili vizuelno poželjan, oblik aproksimacijske funkcije φ ,
- tipičan primjer je (približno) **crtanje grafova** funkcija.

Ostaje još pitanje kako postići vizuelnu “**glatkoću**”?

- Heuristika — **izbjegavanje** naglih **promjena** u derivaciji.
- Dobivene podatke za derivaciju moramo “**izgladiti**”.
- **Problem izgladivanja** podataka je klasični problem numeričke analize.
- Najjednostavniji pristup je **zamjena** podatka **srednjom vrijednošću** podataka preko nekoliko susjednih točaka.

Druge aproksimacije derivacija

Aproksimacija derivacije mogla bi se napraviti još i **bolje**, ako

- povučemo interpolacijski polinom **stupnja 3** koji prolazi točkama x_k , x_{k-1} , x_{k+1} i **jednom** od točaka x_{k-2} ili x_{k+2} (**nesimetričnost!**)
- i njega **deriviramo** u x_k (uz pažljivo deriviranje na rubovima).

Takvim postupkom možemo dobiti **grešku**

- u **funkcijskoj vrijednosti** $O(h^4)$.

Primijetite da **bolja** aproksimacija derivacija **nije potrebna**, jer je greška kod po dijelovima Hermiteove kubične interpolacije također **reda veličine** $O(h^4)$.

Zaključak

Kvazihermiteova po dijelovima kubična interpolacija je, također,

- **lokalna**, tj. promjenom **jedne** točke promijenit će se samo **nekoliko susjednih** kubičnih polinoma.

Točno koliko, ovisi o tome koju smo aproksimaciju derivacije izabrali.

Kubična splajn interpolacija

Kubična splajn interpolacija

Brojeve s_0, \dots, s_n možemo odrediti i iz zahtjeva da

- φ ima neprekidnu drugu derivaciju u u unutarnjim čvorovima x_1, \dots, x_{n-1} , tj. da je klase $C^2[a, b]$.

Takva se interpolacija zove kubična splajn interpolacija.

Iz tih uvjeta ne možemo jednoznačno izračunati splajn, jer

- treba odrediti $4n$ koeficijenata kubičnih polinoma,
- a imamo $2n$ uvjeta interpolacije (svaki polinom mora interpolirati rubne točke svog podintervala),
- uvjeta ljepljenja prve i derivacije u unutarnjim točkama ima $n - 1$ (toliko je unutarnjih točaka),
- i $n - 1$ je uvjeta ljepljenja druge derivacije.

Broj nepoznatih parametara i broj uvjeta

Zaključak. Ukupno imamo

- $4n - 2$ uvjeta,
- a moramo odrediti $4n$ koeficijenata,
- pa vidimo da **nedostaju 2 uvjeta** da bismo te koeficijente mogli odrediti.

Za početak, **prva derivacija** se lijepi u unutarnjim točkama čim postavimo zahtjev da je

$$\varphi'(x_k) = s_k$$

u tim točkama, bez obzira na to što je s_k .

Ljepljenje druge derivacije

Ostaje postaviti uvjete ljepljenja **druge derivacije** u unutarnjim čvorovima. Zahtjev je

$$p_k''(x_k) = p_{k+1}''(x_k), \quad k = 1, \dots, n-1.$$

Ako polinome p_k pišemo u formi relativno obzirom na početnu točku podintervala, tj. ako je

$$p_k(x) = c_{0,k} + c_{1,k}(x - x_{k-1}) + c_{2,k}(x - x_{k-1})^2 + c_{3,k}(x - x_{k-1})^3,$$

onda je

$$p_k''(x) = 2c_{2,k} + 6c_{3,k}(x - x_{k-1})$$
$$p_{k+1}''(x) = 2c_{2,k+1} + 6c_{3,k+1}(x - x_k).$$

Ljepljenje druge derivacije

Uvrštavanjem x_k i dijeljenjem s 2 izlazi uvjet ljepljenja

$$c_{2,k} + 3c_{3,k}(x_k - x_{k-1}) = c_{2,k+1}.$$

Ostaje samo ispisati koeficijente $c_{i,k}$ u terminima f_k i s_k .

Ponovimo, za kubični polinom s 2 dvostruka čvora, imali smo

$$c_{3,k} = \frac{s_k + s_{k-1} - 2f[x_{k-1}, x_k]}{h_k^2},$$
$$c_{2,k} = \frac{f[x_{k-1}, x_k] - s_{k-1}}{h_k} - h_k c_{3,k}.$$

Sada još treba uvrstiti ove dvije relacije u uvjet ljepljenja druge derivacije.

Linearni sustav za kubičnu splajn interpolaciju

Sređivanjem izlazi

$$\frac{-3f[x_{k-1}, x_k] + s_{k-1} + 2s_k}{h_k} = \frac{3f[x_k, x_{k+1}] - 2s_k - s_{k+1}}{h_{k+1}}.$$

Pomnožimo li prethodnu relaciju s $h_k h_{k+1}$ i

- prebacimo li sve s_k na **lijevu** stranu,
- a članove koji nemaju s_k na **desnu** stranu,

za $k = 1, \dots, n - 1$, dobivamo

$$\begin{aligned} h_{k+1}s_{k-1} + 2(h_k + h_{k+1})s_k + h_k s_{k+1} \\ = 3(h_{k+1}f[x_{k-1}, x_k] + h_k f[x_k, x_{k+1}]). \end{aligned}$$

Linearni sustav za kubičnu splajn interpolaciju

Ovo je linearni sustav

• s $(n + 1)$ -om nepoznanicom i $(n - 1)$ -om jednažbom.

Ako zadamo nagibe s_0 i s_n , ostaje točno $n - 1$ nepoznanica.

Matrica tako dobivenog linearnog sustava je **trodijagonalna**

$$\begin{bmatrix} 2(h_1 + h_2) & h_1 & & & & \\ h_3 & 2(h_2 + h_3) & h_2 & & & \\ & \ddots & \ddots & \ddots & & \\ & & h_{n-1} & 2(h_{n-2} + h_{n-1}) & h_{n-2} & \\ & & & h_n & 2(h_{n-1} + h_n) & \end{bmatrix}$$

i **strogo dijagonalno dominantna** po recima, jer je

$$2(h_k + h_{k+1}) > h_k + h_{k+1},$$

pa je i **regularna**.

Rješenje linearnog sustava za splajn interpolaciju

Ovaj linearni sustav sigurno ima **jedinstveno rješenje**
 s_1, \dots, s_{n-1} .

Za rješavanje sustava možemo koristiti Gaussove eliminacije ili LR faktorizaciju **bez** pivotiranja.

Za trodijagonalnu matricu

$$A = \begin{bmatrix} d_1 & e_1 & & & & & \\ c_2 & d_2 & e_2 & & & & \\ & \ddots & \ddots & \ddots & & & \\ & & & c_{n-1} & d_{n-1} & e_{n-1} & \\ & & & & c_n & d_n & \end{bmatrix} .$$

pretpostavimo da **postoji LR faktorizacija** bez pivotiranja.

Rješenje linearnog sustava za splajn interpolaciju

Tada nije teško pokazati da su matrice L i R oblika

$$L = \begin{bmatrix} 1 & & & & & \\ l_2 & 1 & & & & \\ & \ddots & \ddots & & & \\ & & l_{n-1} & 1 & & \\ & & & l_n & 1 & \\ & & & & & 1 \end{bmatrix},$$
$$R = \begin{bmatrix} r_1 & e_1 & & & & \\ & r_2 & e_2 & & & \\ & & \ddots & \ddots & & \\ & & & r_{n-1} & e_{n-1} & \\ & & & & r_n & \end{bmatrix}.$$

Matrice A i R imaju **jednake** dijagonale **iznad** glavne.

Rješenje linearnog sustava za splajn interpolaciju

Ostale elemente matrica L i R računamo po sljedećim rekurzijama

$$r_1 = d_1,$$

za $i = 2, \dots, n$:

$$l_i = c_i / r_{i-1},$$

$$r_i = d_i - l_i e_{i-1}.$$

Primijetite, sada s_k

- nisu nezavisni, nego ovise jedan o drugom.
- To znači da aproksimacija više nije lokalna, jer se promjenom jedne točke mijenjaju svi polinomi.

Dva dodatna uvjeta

Posljednje otvoreno pitanje je kako možemo izabrati s_0 i s_n (nedostaju 2 uvjeta!)?

- Oni se ne zadaju direktno,
- uobičajeno se zadaju rubni uvjeti na funkciju φ iz kojih se određuju s_0 i s_n .

Postoji nekoliko tradicionalnih načina zadavanja rubnih uvjeta, odnosno jednačbi koje nedostaju.

Potpuni (kompletni) splajn

(a) Potpuni (kompletni) splajn

Poznato:

- derivacija funkcije f u rubovima, (recimo kod rješavanja rubnih problema za običnu diferencijalnu jednačbu).

Zadaje se:

$$s_0 = f'(x_0), \quad s_n = f'(x_n).$$

Greška aproksimacije u funkcijskoj vrijednosti je $O(h^4)$.

Zadana druga derivacija u rubovima

(b) Zadana druga derivacija u rubovima

Poznato:

• druga derivacija funkcije f u rubovima.

Zadaje se:

$$f''(x_0) = \varphi''(x_0) = p_1''(x_0), \quad f''(x_n) = \varphi''(x_n) = p_n''(x_n).$$

Treba još izraziti

• $p_1''(x_0)$ preko s_0, s_1 ,

• $p_n''(x_n)$ preko s_{n-1} i s_n .

Zadana druga derivacija u rubovima — početak

Znamo da je

$$c_{2,1} = \frac{p_1''(x_0)}{2} = \frac{f''(x_0)}{2}.$$

Iz izraza za $c_{2,1}$ izlazi

$$\frac{3f[x_0, x_1] - 2s_0 - s_1}{h_1} = \frac{f''(x_0)}{2},$$

ili, ako sredimo, dobivamo

$$2s_0 + s_1 = 3f[x_0, x_1] - \frac{h_1}{2}f''(x_0).$$

Ovu jednadžbu treba dodati kao **prvu** u linearni sustav.

Zadana druga derivacija u rubovima — kraj

Slično, korištenjem da je

$$p_n''(x_n) = 2c_{2,n} + 6c_{3,n}h_n,$$

te uvrštavanjem izraza za $c_{2,n}$ i $c_{3,n}$, izlazi i

$$s_{n-1} + 2s_n = 3f[x_{n-1}, x_n] + \frac{h_n}{2}f''(x_n).$$

Tu jednađbu dodajemo kao **zadnju** u linearni sustav.

Dobiveni linearni sustav

- ima $(n + 1)$ -u jednađbu i **isto** toliko nepoznanica,
- a može se pokazati da ima i **jedinstveno** rješenje.

Greška aproksimacije u funkcijskoj vrijednosti je $O(h^4)$.

Prirodni splajn

(c) Prirodni splajn

Poznato: tzv. slobodni krajevi,

$$\varphi''(x_0) = \varphi''(x_n) = 0.$$

Dodatne jednađbe: kao u (b), samo se stavi

$$f''(x_0) = f''(x_n) = 0$$

$$2s_0 + s_1 = 3f[x_0, x_1], \quad s_{n-1} + 2s_n = 3f[x_{n-1}, x_n].$$

Greška aproksimacije:

- ⦿ Ako f nema druge derivacije na rubu 0, onda je greška u funkcijskoj vrijednosti $O(h^2)$,
- ⦿ ako ih ima, onda je (kao u (b) slučaju) greška $O(h^4)$.

Numerička aproksimacija derivacija

(d) Numerička aproksimacija derivacija

Nepoznato: ponašanje derivacije funkcije f na rubovima.

Preostala dva parametra mogu se odrediti tako da

- **numerički** aproksimiramo φ' ili φ'' ili φ''' u rubovima.
- Kao aproksimaciju koristimo odgovarajuću **derivaciju** kubičnog interpolacijskog polinoma koji prolazi točkama x_0, \dots, x_3 , odnosno x_{n-3}, \dots, x_n .

Greška za bilo koju od ovih varijanti daje je reda $O(h^4)$ u funkcijskoj vrijednosti.

Not-a-knot (nije čvor) splajn

(e) Not-a-knot (nije čvor) splajn

Nepoznato: ponašanje derivacije funkcije f na rubovima.

Umjesto neke aproksimacije derivacije, koristimo tzv. “not-a-knot” (nije čvor) uvjet.

- Parametre s_0 i s_n biramo tako da su prva dva i posljednja dva kubična polinoma jednaka, tj. da je

$$p_1 = p_2, \quad p_{n-1} = p_n.$$

To znači da se u čvoru

- x_1 zalijepi i treća derivacija polinoma p_1 i p_2 ,
- x_{n-1} se zalijepi treća derivacija polinoma p_{n-1} i p_n .

Not-a-knot (nije čvor) splajn — početak

Te zahtjeve možemo pisati kao

$$p_1'''(x_1) = p_2'''(x_1), \quad p_{n-1}'''(x_{n-1}) = p_n'''(x_{n-1}).$$

Zahtjev $p_1'''(x_1) = p_2'''(x_1)$ znači da su vodeći koeficijenti polinoma p_1 i p_2 jednaki,

$$c_{3,1} = c_{3,2}.$$

Pridružimo li taj zahtjev zahtjevu ljepljenja druge derivacije,

$$c_{2,1} + 3c_{3,1}h_1 = c_{2,2},$$

dobivamo ...

Not-a-knot (nije čvor) splajn — početak

... prvu jednadžbu

$$\begin{aligned} \frac{f[x_0, x_1] - s_0}{h_1} + 2 \frac{s_1 + s_0 - 2f[x_0, x_1]}{h_1} \\ = \frac{f[x_1, x_2] - s_1}{h_2} - h_2 \frac{s_1 + s_0 - 2f[x_0, x_1]}{h_1^2}. \end{aligned}$$

Sređivanjem, izlazi

$$\begin{aligned} h_2 s_0 + (h_1 + h_2) s_1 \\ = \frac{(h_1 + 2(h_1 + h_2)) h_2 f[x_0, x_1] + h_1^2 f[x_1, x_2]}{h_1 + h_2}. \end{aligned}$$

Not-a-knot (nije čvor) splajn — kraj

Na sličan način dobivamo i **zadnju** jednadžbu

$$\begin{aligned} & (h_{n-1} + h_n)s_{n-1} + h_{n-1}s_n \\ &= \frac{(h_n + 2(h_{n-1} + h_n))h_{n-1} f[x_{n-1}, x_n] + h_n^2 f[x_{n-2}, x_{n-1}]}{h_{n-1} + h_n}. \end{aligned}$$

Greška aproksimacije za funkcijske vrijednosti je $O(h^4)$.

Porijeklo naziva “not-a-knot”:

- kubični splajn uobičajeno ima neprekidne **druge** derivacije u unutarnjim čvorovima x_1, \dots, x_{n-1} .
- **Treća** derivacija funkcije φ , općenito, “puca”.
- Kod “not-a-knot” splajna, u x_1 i x_{n-1} **ne puca** treća derivacija, pa to **nisu** “pravi” čvorovi splajna.

Ostali rubni uvjeti

(f) Ostali rubni uvjeti

Svi dosad opisani načini zadavanja rubnih uvjeta “čuvaju”

- trodijagonalnu strukturu linearnog sustava za nepoznate parametre s_k .

Za aproksimaciju periodičkih funkcija (na intervalu koji odgovara periodu) zahtijeva se periodičnost prve i druge derivacije u rubovima

$$\varphi'(x_0) = \varphi'(x_n), \quad \varphi''(x_0) = \varphi''(x_n),$$

što vodi na jednadžbe

$$p'_1(x_0) = p'_n(x_n), \quad p''_1(x_0) = p''_n(x_n).$$

Dobiveni linearni sustav više nije trodijagonalan.

Greška kubične splajn interpolacije

Neka je $f \in C^2[a, b]$ i pretpostavimo da

- f ima ograničenu i integrabilnu četvrtu derivaciju na svakom podintervalu $[x_{k-1}, x_k]$.

Tada je

$$\|f(x) - \varphi(x)\|_{\infty} \leq \frac{5}{384} h^4 \|f^{(4)}\|_{\infty},$$

$$\|f'(x) - \varphi'(x)\|_{\infty} \leq \frac{1}{24} h^3 \|f^{(4)}\|_{\infty},$$

$$\|f''(x) - \varphi''(x)\|_{\infty} \leq \frac{3}{8} h^2 \|f^{(4)}\|_{\infty},$$

$$\|f^{(3)}(x) - \varphi^{(3)}(x)\|_{\infty} \leq \frac{1}{2} \left(\frac{1}{\beta} + \beta \right) h \|f^{(4)}\|_{\infty},$$

gdje je $\beta := (\max_k h_k) / (\min_k h_k)$ mjera neuniformnosti mreže.

Greška kubičnog splajna i rubni uvjeti

Ove ocjene greške, naravno, vrijede **samo** uz pretpostavku

- da su i **rubni uvjeti** dovoljno **točni**,
- tj. i oni **zadovoljavaju** odgovarajuću **ocjenu** greške.

U protivnom, **gubimo točnost** pri **rubovima**.

Napomena.

- Dozvoljeno je **kombinirati** **razne** oblike rubnih uvjeta u **jednom** i **drugom** rubu.

Postoji cijela teorija **splajn** funkcija — ne samo polinomnih.

- **Vektorski** prostor, “lokalna” **baza** — **B-splajnovi**, itd.

Demo — Not-a-knot kubična splajn interp.

Pokazati kako izgleda Not-a-knot kubična splajn interpolacija na primjeru funkcije Runge:

$$f(x) = \frac{1}{1+x^2}, \quad x \in [-5, 5],$$

na ekvidistantnim mrežama s parnim brojem podintervala.

 `Num_Pas\Interp\Comp_Spl\GnuPlot\SplRung.plt`

Primjer — prirodni splajn

Primjer. Neka je

$$f(x) = \sin(\pi x).$$

Nađite prirodni splajn koji aproksimira funkciju f na $[0, 1]$ s čvorovima interpolacije $x_k = 0.2k$, za $k = 0, \dots, 5$.

Izračunajte vrijednost tog splajna u točki 0.55.

Budući da su točke ekvidistantne s razmakom $h = 0.2$, “srednje” jednadžbe linearnog sustava za splajn su

$$hs_{k-1} + 4hs_k + hs_{k+1} = 3(hf[x_{k-1}, x_k] + hf[x_k, x_{k+1}]),$$

$$k = 1, \dots, 4.$$

Primjer — prirodni splajn

Dodatne jednadžbe (prva i zadnja) za prirodni splajn su

$$2s_0 + s_1 = 3f[x_0, x_1], \quad s_4 + 2s_5 = 3f[x_4, x_5].$$

Za desnu stranu sustava trebamo **prve** podijeljene razlike

x_k	$f[x_k]$	$f[x_k, x_{k+1}]$
0.0	0.0000000000	2.9389262615
0.2	0.5877852523	1.8163563200
0.4	0.9510565163	0.0000000000
0.6	0.9510565163	-1.8163563200
0.8	0.5877852523	-2.9389262615
1.0	0.0000000000	

Primjer — prirodni splajn

Iz svih ovih podataka dobivamo linearni sustav

$$\begin{bmatrix} 0.4 & 0.2 & & & & \\ 0.2 & 0.8 & 0.2 & & & \\ & 0.2 & 0.8 & 0.2 & & \\ & & 0.2 & 0.8 & 0.2 & \\ & & & 0.2 & 0.8 & 0.2 \\ & & & & 0.2 & 0.4 \end{bmatrix} \cdot \begin{bmatrix} s_0 \\ s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \end{bmatrix} = \begin{bmatrix} 1.7633557569 \\ 2.8531695489 \\ 1.0898137920 \\ -1.0898137920 \\ -2.8531695489 \\ -1.7633557569 \end{bmatrix}$$

Rješenje tog linearnog sustava za “nagibe” je

$$s_0 = -s_5 = 3.1387417029,$$

$$s_1 = -s_4 = 2.5392953786,$$

$$s_2 = -s_3 = 0.9699245271.$$

Primjer — prirodni splajn

Budući da se točka $x = 0.55$ nalazi u intervalu $[x_2, x_3] = [0.4, 0.6]$, **restrikcija** splajna na taj interval je polinom p_3 , kojeg nalazimo iz tablice podijeljenih razlika

t_k	$f[t_k]$	$f[t_k, t_{k+1}]$	$f[t_k, t_{k+1}, t_{k+2}]$	$f[t_k, \dots, t_{k+3}]$
0.4	0.9510565163			
		0.9699245271		
0.4	0.9510565163		-4.8496226357	
		0.0000000000		0.0000000000
0.6	0.9510565163		-4.8496226357	
		-0.9699245271		
0.6	0.9510565163			

Oдавde odmah slijedi da je p_3 , zapravo, **kvadratni** polinom

$$p_3(x) = 0.9510565163 + 0.9699245271(x - 0.4) - 4.8496226357(x - 0.4)^2.$$

Primjer — prirodni splajn

Pogledajmo još aproksimacije za funkciju, prvu i drugu derivaciju u točki 0.55.

	funkcija $j = 0$	prva derivacija $j = 1$	druga derivacija $j = 2$
$f^{(j)}(0.55)$	0.9876883406	-0.4914533661	-9.7480931932
$\varphi^{(j)}(0.55)$	0.9874286861	-0.4849622636	-9.6992452715
greška	0.0002596545	-0.0064911026	-0.0488479218

Aproksimacije su vrlo točne, iako je h relativno velik, jer funkcija $\sin(\pi x)$ zadovoljava prirodne rubne uvjete.

Greška aproksimacije funkcije je reda veličine $O(h^4)$, prve derivacije $O(h^3)$, a druge derivacije $O(h^2)$.

Usporedba raznih vrsta interpolacije

Demo — Interpolacija izmjerenih podataka

Pokazati kako izgleda **usporedba** raznih vrsta interpolacije:

- interpolacija **polinomima**,
- **Akimina** po dijelovima kubična **kvazihermiteova** interpolacija,
- interpolacija **Not-a-knot** kubičnim **splajnom**,

na skupu **izmjerenih** podataka u **praksi**,

- s **raznim** izborima **čvorova** interpolacije.

Ovo je poznato “**težak**” primjer za interpolaciju!

- `Num_Pas\Interp\C_Akim_1\GnuPlot\C_Akim_1.plt`

Diskretni problem najmanjih kvadrata

Minimizacija vektora pogreške

Neka je funkcija f

• zadana na diskretnom skupu točaka x_0, \dots, x_n .

Točaka x_0, \dots, x_n ima mnogo više nego nepoznatih parametara a_0, \dots, a_m aproksimacijske funkcije φ , tj. $n \gg m$.

Aproksimacijska funkcija

$$\varphi(x, a_0, \dots, a_m)$$

određuje se iz uvjeta da je 2-norma vektora pogrešaka u čvorovima aproksimacije najmanja moguća, tj. minimizira se

$$S = \sum_{k=0}^n (f(x_k) - \varphi(x_k))^2 \rightarrow \min.$$

Sustav normalnih jednadžbi

Uočimo da je

- uvijek $S \geq 0$, bez obzira kakvi su parametri, jer se radi o zbroju kvadrata.
- Funkcija S minimizira se kao funkcija više varijabli a_0, \dots, a_m .
- S je dovoljno glatka funkcija, jer je funkcija u parametrima a_k , pa je nužni uvjet ekstrema

$$\frac{\partial S}{\partial a_k} = 0, \quad k = 0, \dots, m.$$

Takav pristup vodi na tzv. sustav normalnih jednadžbi.

Diskretna metoda najmanjih kvadrata za pravac

Primjer. Zadane su točke $(x_0, f_0), \dots, (x_n, f_n)$, koje treba po diskretnoj metodi najmanjih kvadrata aproksimirati **pravcem**

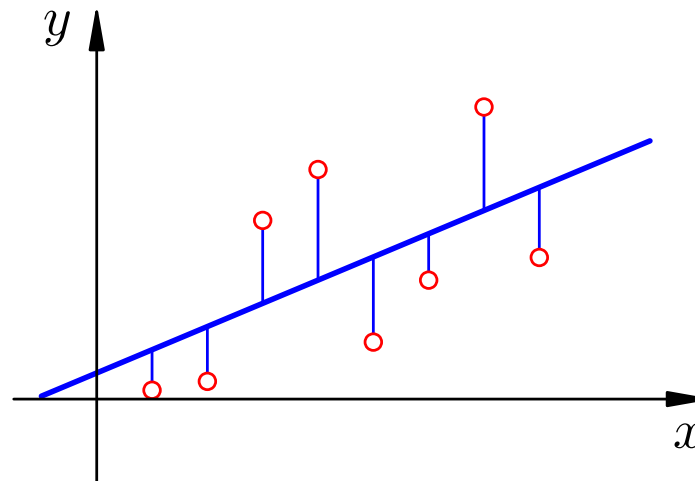
$$\varphi(x) = a_0 + a_1x.$$

Greška aproksimacije (u čvorovima), koju **minimiziramo** je

$$\begin{aligned} S &= S(a_0, a_1) = \sum_{k=0}^n (f_k - \varphi(x_k))^2 \\ &= \sum_{k=0}^n (f_k - a_0 - a_1x_k)^2 \rightarrow \min. \end{aligned}$$

Diskretna metoda najmanjih kvadrata za pravac

Slika **zadanih** točaka u ravnini i **pravca** koji ih aproksimira.



Uočiti da se **greška** u svakoj točki “mjeri” u **smjeru** osi y

$$S = S(a_0, a_1) = \sum_{k=0}^n (f_k - a_0 - a_1 x_k)^2 \rightarrow \min .$$

Diskretna metoda najmanjih kvadrata za pravac

Parcijalne derivacije po parametrima a_0 i a_1 su:

$$0 = \frac{\partial S}{\partial a_0} = -2 \sum_{k=0}^n (f_k - a_0 - a_1 x_k),$$

$$0 = \frac{\partial S}{\partial a_1} = -2 \sum_{k=0}^n (f_k - a_0 - a_1 x_k) x_k.$$

Dijeljenjem s -2 i sređivanjem po nepoznanicama a_0 , a_1 , dobivamo **linearni sustav**

$$a_0(n + 1) + a_1 \sum_{k=0}^n x_k = \sum_{k=0}^n f_k$$

$$a_0 \sum_{k=0}^n x_k + a_1 \sum_{k=0}^n x_k^2 = \sum_{k=0}^n f_k x_k.$$

Diskretna metoda najmanjih kvadrata za pravac

Uvedemo li standardne skraćene oznake

$$s_\ell = \sum_{k=0}^n x_k^\ell, \quad t_\ell = \sum_{k=0}^n f_k x_k^\ell, \quad \ell \geq 0,$$

onda linearni sustav možemo pisati kao

$$s_0 a_0 + s_1 a_1 = t_0$$

$$s_1 a_0 + s_2 a_1 = t_1.$$

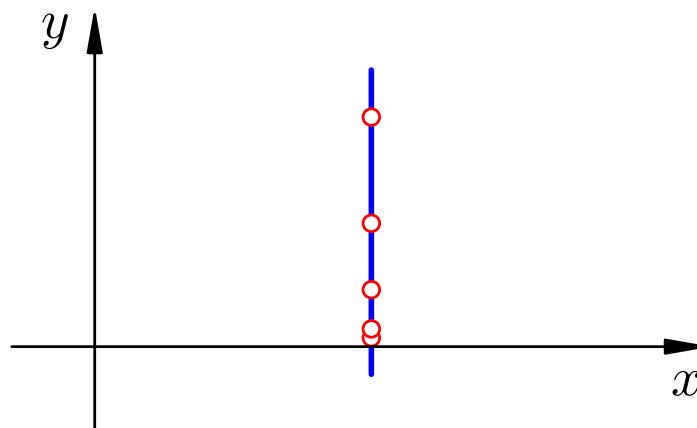
Matrica sustava je **regularna**, što slijedi iz linearne nezavisnosti vektora

$$(1, 1, \dots, 1)^T \quad \text{i} \quad (x_0, x_1, \dots, x_n)^T,$$

uz uvjet da imamo **barem dvije** različite točke x_k , pa postoji **jedinstveno** rješenje sustava.

Diskretna metoda najmanjih kvadrata za pravac

Slika **situacije** u kojoj problem najmanjih kvadrata za pravac **nema** rješenja, s tim da je $n \geq m = 1$, tj. imamo barem **dvije** točke.



Ako imamo **više različitih** podataka u **jednoj** jednoj točki x_0 ,

- aproksimacijski **pravac** (očito) **postoji** i jedinstven je,
- ali je **okomit** na x -os,
- pa njegova jednadžba **nema** oblik $y = a_0 + a_1x$.

Minimalnost rješenja?

Je li to zaista **minimum**?

- To nije teško pokazati, korištenjem **drugih parcijalnih derivacija** (**dovoljan** uvjet minimuma je **pozitivna definitnost** Hesseove matrice).

Provjera je li to minimum, može i puno **lakše**, jer se radi o **zbroju kvadrata**, pa

- S predstavlja **paraboloid** s otvorom prema **gore**, u varijablama a_0, a_1 , pa je jasno da takvi paraboloidi imaju minimum.

Zbog toga se nikad ni **ne provjerava** je li dobiveno rješenje minimum za S .

Najmanji kvadrati za polinome

Za funkciju φ mogli bismo uzeti i polinom višeg stupnja,

$$\varphi(x) = a_0 + a_1x + \cdots + a_mx^m,$$

ali postoji **opasnost** da je za malo veće m ($m \approx 10$)

- dobiveni sustav **vrlo loše uvjetovan**, pa dobiveni **rezultati** mogu biti jako **pogrešni**.

U praksi se to **nikada** direktno ne radi (na ovaj način), već za $m \geq 2, 3$.

Ako se koriste aproksimacije **polinomima viših stupnjeva**,

- onda se to radi korištenjem **ortogonalnih polinoma** (vidjeti kasnije).

Najmanji kvadrati za opće linearne funkcije

Linearni model diskretnih najmanjih kvadrata je potpuno primjenjiv na **opću linearnu funkciju**

$$\varphi(x) = a_0\varphi_0(x) + \cdots + a_m\varphi_m(x),$$

gdje su $\varphi_0, \dots, \varphi_m$ poznate (zadane) funkcije.

Zadatak. Zadane su točke $(x_0, f_0), \dots, (x_n, f_n)$, koje treba po diskretnoj metodi najmanjih kvadrata aproksimirati funkcijom oblika

$$\varphi(x) = a_0\varphi_0(x) + a_1\varphi_1(x).$$

Rješenje. Ide sasvim analogno (pogledati u skripti).

Što s nelinearnim funkcijama?

Što ako φ **nelinearno** ovisi o parametrima?

- Dobivamo **nelinearni** sustav jednačbi, koji se relativno **teško** rješava.
- Problem postaje **ozbiljan** optimizacijski problem, koji se može **približno** rješavati.
- Metode koje se najčešće koriste su **metode pretraživanja** ili **Levenberg–Marquardt** metoda.

Postoji i **drugi** pristup.

- Katkad se jednostavnim **transformacijama** problem može transformirati u **linearni** problem najmanjih kvadrata.
- Rješenja lineariziranog i nelinearnog problema **nisu jednaka**, jer je i greška (**nelinearno**) transformirana!

Primjer nelinearnih najmanjih kvadrata

Primjer. Zadane su točke $(x_0, f_0), \dots, (x_n, f_n)$, koje po diskretnoj metodi najmanjih kvadrata treba aproksimirati funkcijom oblika

$$\varphi(x) = a_0 e^{a_1 x}.$$

Uočite da φ **nelinearno** ovisi o parametru a_1 .

Direktni pristup problemu vodi na minimizaciju

$$\begin{aligned} S &= S(a_0, a_1) = \sum_{k=0}^n (f_k - \varphi(x_k))^2 \\ &= \sum_{k=0}^n (f_k - a_0 e^{a_1 x_k})^2 \rightarrow \min. \end{aligned}$$

Primjer nelinearnih najmanjih kvadrata

Parcijalnim deriviranjem po varijablama a_0 i a_1 dobivamo

$$0 = \frac{\partial S}{\partial a_0} = -2 \sum_{k=0}^n (f_k - a_0 e^{a_1 x_k}) e^{a_1 x_k},$$

$$0 = \frac{\partial S}{\partial a_1} = -2 \sum_{k=0}^n (f_k - a_0 e^{a_1 x_k}) a_0 x_k e^{a_1 x_k}.$$

To je **nelinearan** sustav jednažbi, kojeg ne znamo riješiti!

S **druge** strane, ako **logaritmujemo** relaciju

$$\varphi(x) = a_0 e^{a_1 x},$$

dobivamo

$$\ln \varphi(x) = \ln(a_0) + a_1 x.$$

Primjer nelinearnih najmanjih kvadrata

Moramo **logaritmirati** još i vrijednosti funkcije f u točkama x_k , pa uz supstitucije

$$h(x) = \ln f(x), \quad h_k = h(x_k) = \ln f_k, \quad k = 0, \dots, n,$$

i

$$\psi(x) = \ln \varphi(x) = b_0 + b_1 x,$$

gdje je

$$b_0 = \ln a_0, \quad b_1 = a_1,$$

dobivamo **linearni** problem najmanjih kvadrata

$$\begin{aligned} \tilde{S} &= \tilde{S}(b_0, b_1) = \sum_{k=0}^n (h_k - \psi(x_k))^2 \\ &= \sum_{k=0}^n (h_k - b_0 - b_1 x_k)^2 \rightarrow \min. \end{aligned}$$

Primjer nelinearnih najmanjih kvadrata

Iz rješenja b_0 i b_1 , lako očitamo a_0 i a_1

$$a_0 = e^{b_0}, \quad a_1 = b_1.$$

Napomene uz **linearizaciju**:

- Pri linearizaciji smo pretpostavili da je $f_k > 0$, da bismo mogli **logaritmirati**.
- Ovako dobiveno rješenje **uvijek** daje **pozitivan** a_0 , tj. linearizirani $\varphi(x)$ će uvijek biti veći od 0.
- Kad su neki $f_k \leq 0$, korištenjem **translacije** svih podataka treba dobiti $f_k + \text{translacija} > 0$, pa onda linearizirati.
- Pokušajte **korektno** formulirati takvu **linearizaciju**!

Tipične linearizacije — opća potencija

Kratki popis funkcija koje su često u upotrebi i njihovih **standardnih linearizacija** u problemu najmanjih kvadrata.

(a) Funkcija

$$\varphi(x) = a_0 x^{a_1}$$

linearizira se logaritmiranjem

$$\psi(x) = \log \varphi(x) = \log(a_0) + a_1 \log x,$$

$$h_k = \log f_k, \quad k = 0, \dots, n.$$

Uvedimo oznake

$$b_0 = \log(a_0), \quad b_1 = a_1.$$

Tipične linearizacije — opća potencija

Linearizirani problem najmanjih kvadrata glasi

$$\tilde{S} = \tilde{S}(b_0, b_1) = \sum_{k=0}^n (h_k - b_0 - b_1 \log(x_k))^2 \rightarrow \min,$$

U ovom slučaju, da bismo mogli provesti linearizaciju,

● mora biti i $x_k > 0$ i $f_k > 0$.

Tipične linearizacije — 1 / linearna funkcija

(b) Funkcija

$$\varphi(x) = \frac{1}{a_0 + a_1 x}$$

linearizira se na sljedeći način

$$\psi(x) = \frac{1}{\varphi(x)} = a_0 + a_1 x,$$

$$h_k = \frac{1}{f_k}, \quad k = 0, \dots, n.$$

Pripadni **linearni problem** najmanjih kvadrata je

$$\tilde{S} = \tilde{S}(a_0, a_1) = \sum_{k=0}^n (h_k - a_0 - a_1 x_k)^2 \rightarrow \min.$$

Tipične linearizacije — x / linearna funkcija

(c) Funkciju

$$\varphi(x) = \frac{x}{a_0 + a_1 x}$$

možemo linearizirati na **više** načina.

1. način:

$$\psi(x) = \frac{1}{\varphi(x)} = a_0 \frac{1}{x} + a_1,$$

$$h_k = \frac{1}{f_k}, \quad k = 0, \dots, n.$$

Pripadni **linearni** problem najmanjih kvadrata je

$$\tilde{S} = \tilde{S}(a_0, a_1) = \sum_{k=0}^n \left(h_k - a_0 \frac{1}{x_k} - a_1 \right)^2 \rightarrow \min.$$

Tipične linearizacije — x / linearna funkcija

2. način:

$$\psi(x) = \frac{x}{\varphi(x)} = a_0 + a_1x,$$

$$h_k = \frac{x_k}{f_k}, \quad k = 0, \dots, n.$$

Pripadni **linearni** problem najmanjih kvadrata je

$$\tilde{S} = \tilde{S}(a_0, a_1) = \sum_{k=0}^n (h_k - a_0 - a_1x_k)^2 \rightarrow \min.$$

Tipične linearizacije — još jedan primjer

(d) Funkcija

$$\varphi(x) = \frac{1}{a_0 + a_1 e^{-x}}$$

linearizira se stavljanjem

$$\psi(x) = \frac{1}{\varphi(x)} = a_0 + a_1 e^{-x},$$

$$h_k = \frac{1}{f_k}, \quad k = 0, \dots, n.$$

Pripadni **linearni** problem najmanjih kvadrata je

$$\tilde{S} = \tilde{S}(a_0, a_1) = \sum_{k=0}^n (h_k - a_0 - a_1 e^{-x_k})^2 \rightarrow \min.$$

Primjer

Primjer. Uvaženi znanstvenik **dr. Zurić**, Ulica astronoma 69, dobio je ideju da se Zemlja giba oko Sunca po **eliptičnoj orbiti**, sa Suncem u jednom fokusu.

Nakon niza opažanja i mjerenja (uz dosta računa), dobio je slijedeće podatke

$x [^\circ]$	0	45	90	135	180
$r [10^6 \text{ km}]$	147	148	150	151	152

u kojima je

- r **udaljenost** od Zemlje do Sunca (u 10^6 km),
- a x je **kut** između **spojnice** Zemlja–Sunce i **glavne osi** elipse (u **stupnjevima**):

Primjer (nastavak)

Dr. Zurić, naravno, zna da se **elipsa** može opisati formulom

$$r = \frac{\rho}{1 + \varepsilon \cos x}.$$

Pomognite mu da nađe ρ i ε , **diskretnom linearnom** metodom najmanjih kvadrata, nakon **preuređenja** ove formule.

Rješenje. Pomnožimo formulu s **nazivnikom** funkcije, pa dobivamo

$$r(1 + \varepsilon \cos x) = \rho,$$

odnosno

$$-\varepsilon r \cos x + \rho = r.$$

Primjer (nastavak)

Relaciju $-\varepsilon r \cos x + \rho = r$ gledamo kao funkciju oblika

$$au + b = v,$$

gdje je

$$u = r \cos x, \quad v = r, \quad a = -\varepsilon, \quad b = \rho.$$

Zatim se primijeni **linearna** metoda najmanjih kvadrata za **pravac**.

Prema tome, treba minimizirati

$$S = \sum_{i=0}^4 (v_i - au_i - b)^2 \rightarrow \min.$$

Primjer (nastavak)

Deriviranjem izlazi

$$\frac{\partial S}{\partial a} = -2 \sum_{i=0}^4 (v_i - au_i - b)u_i = 0$$

$$\frac{\partial S}{\partial b} = -2 \sum_{i=0}^4 (v_i - au_i - b) = 0.$$

Nakon sređivanja, uz $n + 1 = 5$, imamo

$$\begin{bmatrix} \sum_{i=0}^4 u_i^2 & \sum_{i=0}^4 u_i \\ \sum_{i=0}^4 u_i & 5 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \sum_{i=0}^4 u_i v_i \\ \sum_{i=0}^4 v_i \end{bmatrix}.$$

Primjer (nastavak)

Kad se radi “na ruke”, traženi podaci se obično slože u **tablicu**

i	$v_i = r_i$	$u_i = r_i \cos x_i$	u_i^2	$u_i v_i$
0	147	147	21609	21609
1	148	104.6518036	10952	15488.46693
2	150	0	0	0
3	151	-106.7731240	11400.5	-16122.74172
4	152	-152	23104	-23104
Σ	748	-7.1213204	67065.5	-2129.27479

Primjer (nastavak)

Linearni sustav za koeficijente je

$$\begin{bmatrix} 67065.5 & -7.1213204 \\ -7.1213204 & 5 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} -2129.27479 \\ 748 \end{bmatrix}.$$

Rješenje tog sustava je

$$a = -1.58663722 \cdot 10^{-2}, \quad b = 149.5774021,$$

pa je

$$\varepsilon = 1.58663722 \cdot 10^{-2}, \quad \rho = 149.5774021.$$

Značenja:

- ε je **ekscentricitet** elipse,
- ρ je “**srednja**” udaljenost od Zemlje do Sunca.

Izbor aproksimacijske funkcije

Kad smo jednom odabrali oblik aproksimacijske funkcije pitamo se — jesmo li **dobro** izabrali njezin oblik?

- Kad nađemo aproksimaciju, moramo pogledati **graf pogreške**.
- Ako on “**jednoliko**” oscilira oko **nule**, onda je aproksimacijska funkcija **dobro** odabrana.

Primjer uklanjanja slučajne greške

Metoda najmanjih kvadrata uklanja i slučajne greške (recimo, kod mjerenja). To joj je osnovna svrha u statistici!

Primjer. Eksperimentalni podaci uzeti su tako da se y koordinate točaka na pravcu

$$y(x) = 4x + 3$$

za $x = 0, 1, \dots, 100$, perturbiraju za

• uniformno distribuirani slučajni broj, između -1 i 1 .

Tako se dobiju podaci

$$f_i = 4x_i + 3 + \text{slučajna perturbacija između } -1 \text{ i } 1,$$
$$i = 0, \dots, 100.$$

Primjer uklanjanja slučajne greške

Prvih nekoliko podataka izgleda ovako:

x_i	$y(x_i)$	f_i
0	3	3.481757957246973
1	7	7.905987449877890
2	11	11.931070097690015
3	15	15.495131876084549
4	19	18.681441353019998
5	23	22.984820207108194

Kad se metodom najmanjih kvadrata za **pravac** $\varphi(x) = ax + b$ izračunaju parametri, oni su

$$a = 3.99598, \quad b = 3.20791.$$

Primjer uklanjanja slučajne greške

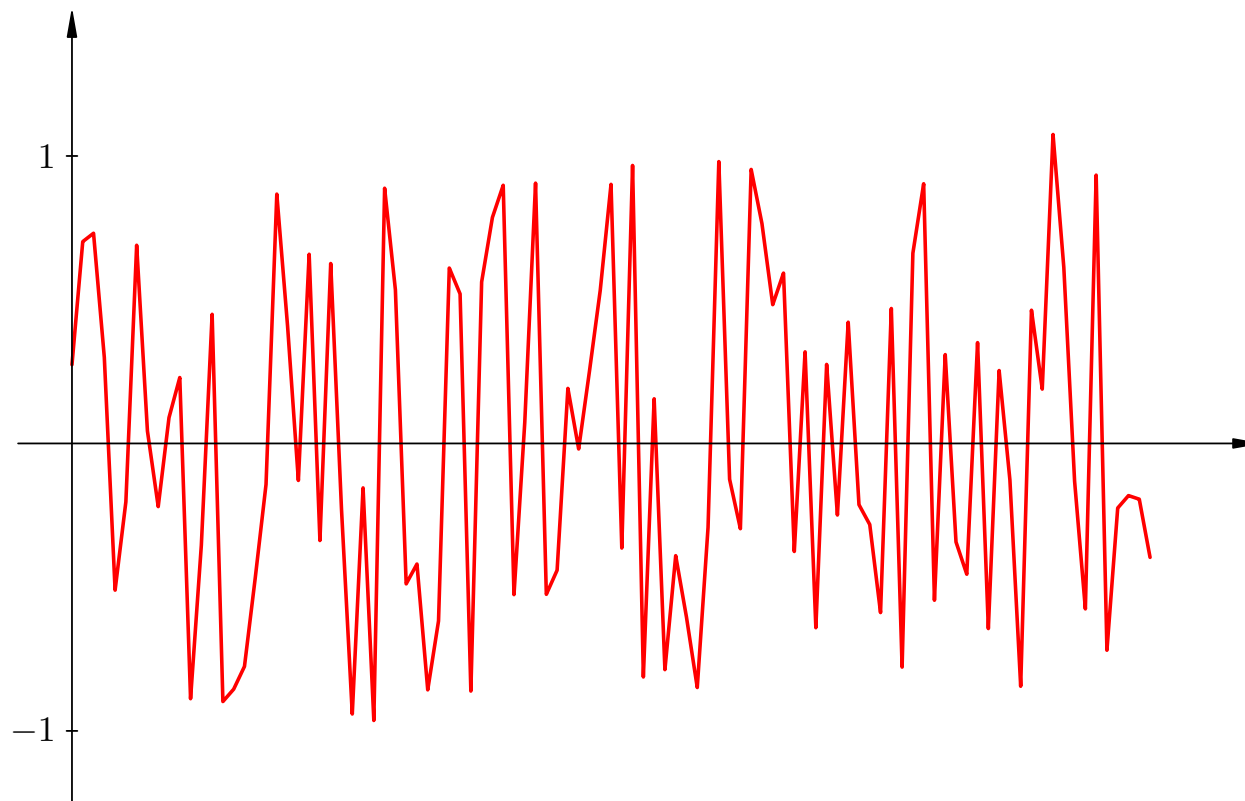
Pogledajmo što su aproksimacije vrijednosti f_i za prvih nekoliko podataka:

x_i	$y(x_i)$	$\varphi(x_i)$
0	3	3.207905163100534
1	7	7.203881519200112
2	11	11.199857875299690
3	15	15.195834231399269
4	19	19.191810587498847
5	23	23.187786943598425

Uočite da su greške $\varphi(x_i)$ obzirom na $y(x_i)$ **znatno manje** nego greške f_i obzirom na $y(x_i)$.

Primjer uklanjanja slučajne greške

Pogledajmo kako se ponaša greška $f_i - \varphi(x_i)$.



Greška izgleda kao slučajna uniformna funkcija između -1 i 1 , što znači da smo uklonili slučajnu grešku.

Demo primjeri

GnuPlot demo za prethodni problem.

- `Num_Pas\Mls\GnuPlot\Pravac.plt`

Diskretnom metodom najmanjih kvadrata aproksimiraju se podaci za **viskoznost 40%** etilnog alkohola.

- Primjer pokazuje **različita** rješenja ako problem **lineariziramo** ili ako ga **ne lineariziramo**.
- Također, pokazan je **način izbora** aproksimacijske funkcije.
- `Num_Pas\Mls\GnuPlot\Etil.plt`

Matrična formulacija linearnog problema najmanjih kvadrata

Matrična formulacija

Diskretni **linearni** problem najmanjih kvadrata najčešće se rješava u **matričnom** obliku.

Da bismo formirali **matrični zapis** linearnog problema najmanjih kvadrata, zgodno je **preimenovati** nepoznanice,

- tako da **matricu**,
- vektor **desne** strane i
- **nepoznanice** u linearnom sustavu

pišemo u uobičajenoj formi:

- **standardno** su nepoznanice x_1, \dots, x_m ,
- a ne a_0, \dots, a_m .

Matrična formulacija

Pretpostavimo da skup podataka (t_k, y_k) , za $k = 1, \dots, n$, želimo aproksimirati **linearnom** funkcijom

$$\varphi(t) = x_1\varphi_1(t) + \dots + x_m\varphi_m(t).$$

Želimo pronaći parametre x_j tako da podaci (t_k, y_k) zadovoljavaju

$$y_k = \sum_{j=1}^m x_j\varphi_j(t_k), \quad k = 1, \dots, n.$$

Primijetite da to nije uvijek moguće, jer je podataka uobičajeno **znatno više** nego parametara.

Matrična formulacija

Ako označimo

$$a_{kj} = \varphi_j(t_k), \quad b_k = y_k,$$

onda prethodne jednađbe možemo napisati u **matričnom obliku**

$$Ax = b.$$

Budući da je matrica A “**dugačka**”, postavlja se pitanje što je **najbolje** rješenje ovog sustava.

Najčešće određujemo x tako da se minimizira **rezidual** $r = Ax - b$, tj. tražimo rješenje problema

$$\min_x \|r\|_2 = \min_x \|Ax - b\|_2, \quad A \in \mathbb{R}^{n \times m}, \quad b \in \mathbb{R}^n.$$

Komentar

Ako smo dobro birali bazične funkcije φ_i , onda je razumno pretpostaviti da su one **linearno nezavisne** na **zadanim podacima**, pa

- matrica A ima **puni stupčani rang**, tj. $\text{rang}(A) = m$.

S druge strane, ako je $\text{rang}(A) < m$, onda

- rješenje x **nije jedinstveno**, jer mu možemo dodati bilo koji vektor iz nul-potprostora od A , a da se rezidual ne promijeni.

Odsad nadalje, pretpostavimo (ako drugačije nije rečeno) da A ima **puni stupčani rang**.

Veza s problemom najmanjih kvadrata

Teorem. Skup svih rješenja problema $\min_x \|r\|_2$ označimo s

$$\mathcal{S} = \{x \in \mathbb{R}^m \mid \|Ax - b\|_2 = \min\}.$$

Tada je $x \in \mathcal{S}$ ako i samo ako vrijedi sljedeća relacija ortogonalnosti

$$A^T(b - Ax) = 0,$$

koju obično nazivamo **sustav normalnih jednažbi** i pišemo u obliku

$$A^T Ax = A^T b.$$

Veza s problemom najmanjih kvadrata

Dokaz. Pretpostavimo da \hat{x} zadovoljava normalne jednadžbe

$$A^T \hat{r} = 0, \quad \hat{r} = b - A\hat{x}.$$

Tada za bilo koji $x \in \mathbb{R}^m$ imamo

$$r = b - Ax = \hat{r} + A\hat{x} - Ax = \hat{r} - A(x - \hat{x}).$$

Ako označimo $e = x - \hat{x}$, onda je $r = \hat{r} - Ae$, pa imamo

$$\begin{aligned} \|r\|_2^2 &= r^T r = (\hat{r} - Ae)^T (\hat{r} - Ae) \\ &= \hat{r}^T \hat{r} - \hat{r}^T Ae - (Ae)^T \hat{r} + (Ae)^T Ae \\ &= \|\hat{r}\|_2^2 - (A^T \hat{r})^T e - e^T (A^T \hat{r}) + \|Ae\|_2^2 = \{A^T \hat{r} = 0\} \\ &= \|\hat{r}\|_2^2 + \|Ae\|_2^2 = \|\hat{r}\|_2^2 + \|A(x - \hat{x})\|_2^2, \end{aligned}$$

što je **minimizirano** kad je $e = 0$, tj. $x = \hat{x}$.

Veza s problemom najmanjih kvadrata

S druge strane, pretpostavimo da \hat{x} **minimizira** normu reziduala, ali **ne zadovoljava** sustav normalnih jednažbi

$$A^T \hat{r} = z \neq 0.$$

Tada možemo definirati

$$x = \hat{x} + \varepsilon z,$$

za proizvoljni dovoljno **mali** broj $\varepsilon > 0$, pa je

$$r = \hat{r} - \varepsilon Az$$

i

$$\|r\|_2^2 = r^T r = \hat{r}^T \hat{r} - 2\varepsilon z^T z + \varepsilon^2 (Az)^T (Az) < \hat{r}^T \hat{r} = \|\hat{r}\|_2^2,$$

što znači da, **protivno** pretpostavci, \hat{x} **nije** rješenje problema minimizacije reziduala $\|r\|_2$, za **svaki** dovoljno **mali** $\varepsilon > 0$. ■

Karakterizacija rješenja

Prethodni teorem, zapravo je rekao da je rješenje problema minimizacije $\|r\|_2$ isto što i rješenje sustava normalnih jednadžbi

$$A^T A x = A^T b.$$

Sada odmah vidimo:

- matrica $A^T A$ je simetrična i pozitivno semidefinitna, jer za svaki vektor x vrijedi

$$x^T A^T A x = (x^T A^T)(Ax) = (Ax)^T(Ax) = \|Ax\|_2^2 \geq 0,$$

- sustav normalnih jednadžbi uvijek ima rješenje, jer je

$$A^T b \in \mathcal{R}(A^T) = \mathcal{R}(A^T A),$$

(v. teorem Kronecker–Capelli).

Karakterizacija rješenja

Vrijedi čak i jače.

Teorem. Matrica $A^T A$ je **pozitivno definitna** (pa onda i regularna) ako i samo ako su stupci od A **linearno nezavisni**, tj. ako je $\text{rang}(A) = m$.

Dokaz. Ako su stupci od A **linearno nezavisni**, tada za svaki $x \neq 0$ vrijedi $Ax \neq 0$ (definicija linearne nezavisnosti), pa je za takav x

$$x^T A^T A x = \|Ax\|_2^2 > 0,$$

tj. $A^T A$ je **pozitivno definitna**.

Karakterizacija rješenja

S druge strane, ako su stupci **linearno zavisni**, tada postoji $x_0 \neq 0$ takav da je $Ax_0 = 0$, pa je za takav x_0

$$x_0^T A^T Ax_0 = \|Ax_0\|_2^2 = 0.$$

Ako je x takav da je $Ax \neq 0$, onda je

$$x^T A^T Ax = \|Ax\|_2^2 > 0,$$

pa je $A^T A$ **pozitivno semidefinitna**. ■

Karakterizacija rješenja

Ako je $A^T A$ pozitivno definitna, tj. ako je $\text{rang}(A) = m$, onda postoji **jedinstveno** rješenje problema najmanjih kvadrata (matrica sustava je regularna)!

$$A^T A x = A^T b,$$

koje je dano s

$$x = (A^T A)^{-1} A^T b,$$

s tim da mu je **rezidual**

$$r = b - A(A^T A)^{-1} A^T b.$$

Geometrijska interpretacija rješenja

Desna strana b može se napisati kao

$$b = Ax + r,$$

pri čemu je $Ax \in \mathcal{R}(A)$. Nadalje, iz sustava normalnih jednažbi

$$A^T(b - Ax) = -A^T r = 0,$$

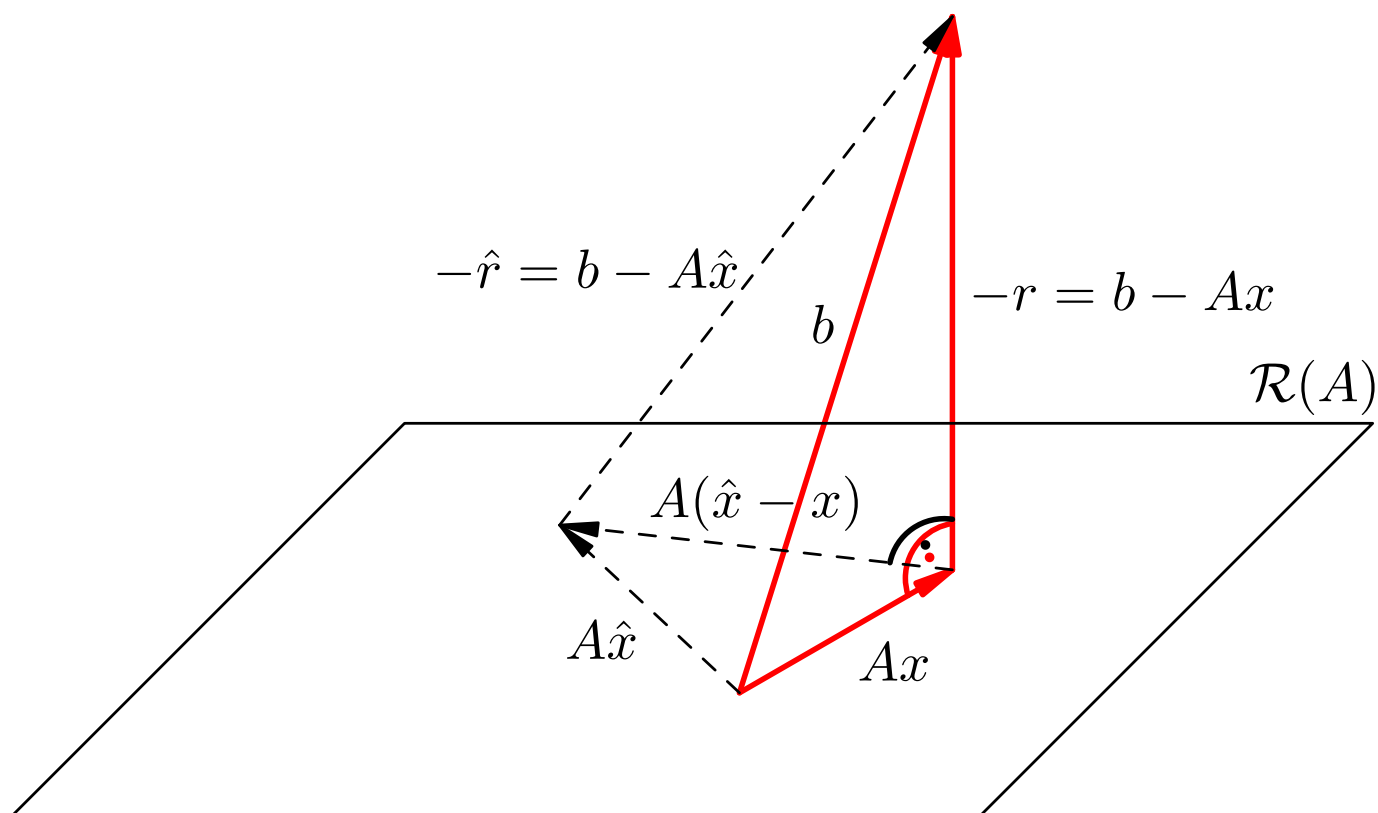
izlazi da je $r \in \mathcal{N}(A^T)$. Prisjetimo li se da je

$$\mathcal{R}(A) \oplus \mathcal{N}(A^T) = \mathbb{R}^n$$

dobivamo **geometrijsku** interpretaciju problema najmanjih kvadrata.

Geometrijska interpretacija rješenja

Rješenje problema najmanjih kvadrata dobivamo tako da napravimo **ortogonalnu projekciju** vektora b na potprostor $\mathcal{R}(A)$.



Rješenje problema najmanjih kvadrata

Treba još samo pronaći način kako jednostavno “pročitati” rješenje. Jasno je da se matrica $A^T A$ ne invertira, nego se rješava linearni sustav

$$A^T A x = A^T b.$$

Ovaj **pozitivno definitni** sustav normalnih jednažbi mogli bismo riješiti tako da iskoristimo **faktorizaciju Choleskog**.

Prednosti/nedostaci metode:

- ukupan broj aritmetičkih operacija za rješenje je $nm^2 + \frac{1}{3}m^3 + O(m^2)$, što je **brzo**,
- ali, rješavanje na ovaj način **nije naročito točno**.

Korištenje QR faktorizacije

Ponovno, neka je $A^T A$ pozitivno definitna. Promatramo problem minimizacije

$$\|Ax - b\|_2 \rightarrow \min.$$

Prisjetite se, za proizvoljnu ortogonalnu matricu Q^T vrijedi da čuva skalarni produkt, (onda i kvadrat norme, pa i normu).

Dakle, rješenje problem minimizacije možemo lako zapisati kao

$$\|Ax - b\|_2 = \|Q^T(Ax - b)\|_2 = \|Q^T Ax - Q^T b\|_2 \rightarrow \min.$$

Pitanje je samo kako naći pogodan Q^T tako da lako pročitamo rješenje.

Odgovor: korištenjem QR faktorizacije.