Metode minimizacije

Adamović, Matea

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Matea Adamović

METODE MINIMIZACIJE

ZAVRŠNI RAD

Varaždin, 2022.

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Matea Adamović

JMBAG: 0016141360

Studij: Poslovni sustavi

METODE MINIMIZACIJE

ZAVRŠNI RAD

Mentor/Mentorica:

Marija Jakuš, dipl. inž.

Varaždin, rujan 2022.

Matea Adamović

Izjava o izvornosti

Izjavljujem da je moj završni/diplomski rad izvorni rezultat mojeg rada te da se u izradi istoga nisam koristio drugim izvorima osim onima koji su u njemu navedeni. Za izradu rada su korištene etički prikladne i prihvatljive metode i tehnike rada.

Autor/Autorica potvrdio/potvrdila prihvaćanjem odredbi u sustavu FOI-radovi

---

Sažetak

Glavna tematika ovog završnog rada su metode minimizacije te njihova podjela. Kako bi potpunije razumjeli navedene metode pa tako i samu minimizaciju kao dio optimizacije, prvo su prikazani osnovni pojmovi potrebni za razumijevanje glavnog dijela rada. Proučavamo metode te njihov način funkcioniranja na realnim primjerima. Postupno su opisane metode jednodimenzionalne minimizacije, gradijentne metode i metode Newtonovog tipa kao jedne od najčešće korištenih minimizacijskih metoda. Svaka od ovih metoda problem minimizacije rješava na svoj način i bolje odgovara određenom tipu funkcija. Količina metoda potvrđuje važnost rješavanja problema minimizacije u svakodnevnim znanstvenim, ali i društvenim situacijama. Problemi raznih stručnih područja kao što su mehanika, ekonomija, teorija igara mogu se svesti na optimizacijski problem gdje vidimo njegovu široku primjenu danas.

Ključne riječi: konveksni skupovi; konveksne funkcije; optimizacija; minimizacija; lokalni minimum; stacionarna točka; gradijent funkcije

Sadržaj

1. Uvod

Metode minimizacije, kao glavna tematika ovog rada, dio su optimizacije koja ima široku primjenu u svim dijelovima života danas. Problemi raznih današnjih stručnih područja (mehanika, ekonomija, operacijska istraživanja) mogu se svesti na neki optimizacijski problem.

Nakon kratke obrade osnovnih pojmova minimizacijske metode svrstane su u tri skupine: metode jednodimenzionalne minimizacije, gradijentne metode i metode Newtonovog tipa. Minimizacija jednodimenzionalnih funkcija (funkcije s jednom nepoznanicom) obilježavaju najpoznatije metode poput metode polovljenja, metode zlatnog reza, metode parabole i Brentove metode.

Za višedimenzionalnu minimizaciju (funkcije više varijabli) najčešće se koriste gradijentne metode poput metode najbržeg spusta te metode zrcalnog spusta. Popularne su i metode Newtonovog tipa, kako za jednodimenzionalnu tako i za višedimenzionalnu minimizaciju.

Svaka metoda daje ispravne rezultate, tj. ispravne vrijednosti minimuma funkcije, ali o tipu funkcije ovisi efikasnost same metode. Za svaku funkciju pojedinačno je ispravnije odabrati najprikladniju metodu kako bi cijeli proces minimizacije bio što učinkovitiji.

2. Metode i tehnike rada

Za izradu ovog završnog rada korištena je literatura pronađena online pomoću Google i Google Scholar tražilica koja je u radu referencirana uz pomoć Mendeley programskog alata. Korišteni su razni znanstveni radovi, stručni članci i udžbenici za utvrđivanje definicija, pravila i teorema metoda minimizacije te svega vezanog uz njih. Usporedbom i kombinacijom sve dostupne literature, kao i filtriranjem pouzdane od nepouzdane nastao je ovaj rad vlastitom interpretacijom dostupne literature.

Za izradu primjera zadataka u kojima se pokazuje način funkcioniranja pojedine metode korišteni su alati poput Geogebre, MS Excel-a, raznih online kalkulatora te programski jezik Python.

3. Metode minimizacije

U ovom poglavlju bit će opisane odabrane metode minimizacije te definirani osnovni pojmovi vezani uz tu tematiku. Pregledom osnovnih pojmova na početku prikazat će se pojmovi i problemi na koje se fokusiraju odabrane metode. Nakon toga slijedi razrada metoda klasificiranih u tri skupine: metode jednodimenzionalne minimizacije, gradijentne metode i metode Newtonovog tipa.

3.1. Osnovni pojmovi

Kako bi razumjeli glavne metode jednodimenzionalne i višedimenzionalne minimizacije potrebno je definirati osnovne pojmove vezane uz te metode.

Infimum (najveća donja međa) u oznaci $\alpha = \inf S$ je najveći realni broj α sa svojstvom $\alpha \leq \beta$ za svaki broj β iz skupa S. Ukoliko skup S nije prazan, ali takav broj α ne postoji infimum je $-\infty$. U drugom slučaju ako je S prazan skup infimum je ∞. Ako infimum pripada skupu S on je najmanji element skupa S i naziva se minimumom od S, a ako ne pripada skupu onda skup nema najmanjeg elementa. [1]

Minimumom funkcije f se naziva najmanji od brojeva $f(x)$ funkcije f definirane na skupu S čije su vrijednosti realni brojevi. S obzirom na područje promatranja, minimum može biti lokalni ili globalni (apsolutni). [2]

Za zadanu funkciju $f: D \to \mathbb{R}, D \subseteq \mathbb{R}^n$ promatramo sljedeće probleme:

1. Odrediti $f^* = \inf\limits_{x \in D} f(x)$;
2. Ako funkcija f postiže svoj infimum na D, treba odrediti $argmin\, f(x)$, tj. točku $x^* \in D$ u kojoj funkcija f postiže svoj infimum;
3. Ako se $\inf f(x)$ ne postiže na D, treba odrediti niz $(x_n) \subset D$ za koji je $\lim\limits_{n \to \infty} f(x_n) = f^*$

Oznaka $\arg min\, f(x)$ predstavlja skup točaka (može biti jedna točka ili više točaka) u kojima zadana funkcija f postiže svoju minimalnu vrijednost. Tako oznaka $\arg\max f(x)$ predstavlja skup točaka u kojima funkcija f postiže svoju maksimalnu vrijednost. Bitno je primijetiti kako je dovoljno tražiti lokalni (globalni) minimum funkcije $f: D \to \mathbb{R}$ jer vrijedi

$$\underset{x \in D}{\operatorname{argmin}} f(x) = \underset{x \in D}{\operatorname{argmax}}(-f(x)) \ .$$

Prethodno opisani problemi optimizacije imaju široku primjenu u raznim granama poput mehanike, ekonomije, operacijskim istraživanjima i slično. [3]

Za numeričku optimizaciju bitnu ulogu ima konveksnost te je stoga bitno objasniti pojmove konveksnih skupova i konveksnih funkcija. Konveksan skup je skup $S \subseteq \mathbb{R}^n$ ako je

$$\lambda x_1 + (1-\lambda)x_2 \in S \qquad \text{za } \forall x_1, x_2 \in S \text{ i } \lambda \in [0,1].$$

Geometrijski gledano to znači da je **skup S konveksan** ako za svake dvije točke $x_1, x_2 \in S$ je i dužina koja spaja te točke sadržana u S. Dogovorno, prazan skup $\emptyset$ i skup $\mathbb{R}^n$ su konveksni. Također, iz definicije konveksnog skupa, može se zaključiti da su svi jednočlani skupovi konveksni. Primjeri konveksnih skupova su pravilni poligoni te pravilni poliedri. [4]

Skup $\mathbb{R}^{\boldsymbol{n}} = \{(\alpha_1, \dots, \alpha_n); \alpha_1, \dots, \alpha_n \in \mathbb{R}\}$ definiran je kao n-dimenzionalni realni vektorski prostor sa operacijama definiranim po komponentama. [5]

Na n-dimenzionalnom euklidskom prostoru $\mathbb{R}^n$ intuitivni pojam duljine vektora $x = (x_1, x_2, \dots, x_n)$ prikazujemo formulom

$$\|x\|^2 = \sqrt{x_1^2 + \cdots + x_n^2} \,.$$

Ta formula se naziva euklidskom normom koja zapravo izračunava udaljenost ishodišta od točke x. Bez koordinata normu možemo zapisati kao

$$\|x\| = \sqrt{x \cdot x} \,. \qquad [6]$$

Najmanji konveksni skup koji sadržava skup $D \subseteq \mathbb{R}^n$ naziva se konveksnom ljuskom skupa D. Na primjeru skupa D kojeg sačinjavaju tri različite točke iz $\mathbb{R}^n$ konveksna ljuska je trokut s vrhovima u tim trima točkama. [3]

Funkcija $f: \mathbb{R}^n \to \mathbb{R}$ je **konveksna** ako za sve $x_1, x_2 \in \mathbb{R}^n$ i za svaki $\lambda \in [0,1]$ vrijedi

$$f\big((1-\lambda)x_1 + \lambda x_2\big) \;\leq\; (1-\lambda)f(x_1) + \lambda f(x_2) \,. \qquad [3].$$

Kako bi potvrdili konveksnost funkcije na određenom intervalu možemo se poslužiti karakterizacijom konveksnosti pomoću druge derivacije. Taj postupak definiran je sljedećim koracima:

1) Određivanje domene funkcije f (ukoliko nemamo zadani interval)
2) Određivanje druge derivacije funkcije f''
3) Ako je $f''(x) > 0$ funkcija f je konveksna na području tog intervala, a ako je $f''(x) < 0$ funkcija f je konkavna. [8]

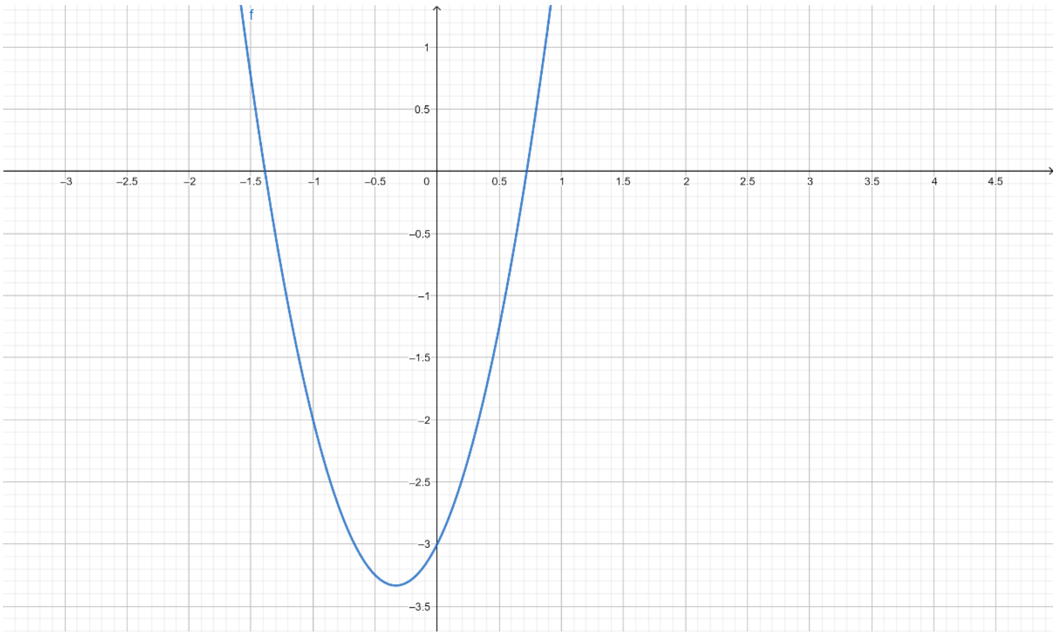
Razlikujemo konveksne, strogo konveksne i jako konveksne funkcije. Funkcija $f: \mathbb{R}^n \to \mathbb{R}$ je **strogo konveksna** ako za $\forall x_1, x_2 \in \mathbb{R}^n, x_1 \neq x_2$ vrijedi

$$f\big((1-\lambda)x_1 + \lambda x_2\big) < (1-\lambda)f(x_1) + \lambda f(x_2), \ \ \forall \lambda \epsilon (0,1).$$

Funkcija $f: \mathbb{R}^n \to \mathbb{R}$ je **jako konveksna** ako postoji realan broj $\gamma > 0$ za koji vrijedi

$$f\left(\frac{x_1 + x_2}{2}\right) \leq \frac{1}{2}\big(f(x_1) + f(x_2)\big) - \gamma \|x_1 - x_2\|^2, \ \ \forall x_1, x_2 \in \mathbb{R}^n.$$

Neka funkcija može istovremeno biti konveksna, strogo konveksna i jako konveksna što je vidljivo na primjeru kvadratne funkcije $f(x) = 3x^2 + 2x - 3$ s $\gamma = \frac{3}{4}$ prikazanoj na slici ispod. [1].



Slika 1. Grafički prikaz jako konveksne funkcije

Kako je glavna tema minimizacija, ključna je definicija lokalnog minimuma kojeg ćemo definirati za proizvoljnu funkciju $f: D \to \mathbb{R}, D \subseteq \mathbb{R}^n$.

Točka lokalnog minimuma $x^* \in D$ funkcije f na skupu D ima okolinu O tako da je $f(x) \geq f(x^*)$ za sve $x \in O \cap D$. Točku x^* zovemo točkom strogog minimuma ako postoji okolina O tko da vrijedi $f(x) > f(x^*)$ za sve $x \in O \cap D \ \setminus \{x^*\}$, a točkom globalnog minimuma funkcije f ako je $f(x) \geq f(x^*)$ za sve $x \in D$. Točka $x^* \in D \subseteq \mathbb{R}^n$ je stacionarna točka funkcije $f: D \to \mathbb{R}$ ako je $\nabla f(x^*) = 0$. [1].

Gradijent funkcije f u točki $x = (x_1, x_2, \dots, x_n)$ je vektor definiran sljedećom formulom

$$\nabla f(x) = \left(\frac{\partial f(x)}{\partial x_1}, \frac{\partial f(x)}{\partial x_2}, \dots, \frac{\partial f(x)}{\partial x_n}\right). \qquad [9]$$

3.2. Jednodimenzionalna minimizacija

Krećemo sa samom minimizacijom za zadanu funkciju $f : [a, b] \to \mathbb{R}$ koja na intervalu $[a, b]$ postiže $\inf\limits_{x\in[a,b]} f(x)$. Promatramo sljedeće probleme:

1. Određivanje barem jednog x^* tako da je $x^* \in \operatorname*{argmin}\limits_{x\in D} f(x)$ i $f^* = f(x^*)$
2. Određivanje $U = \{u \in [a, b] : f(u) = f^*\}$, tj. skupa svih točaka iz zadanog segmenta $[a, b]$ u kojima se postiže minimum zadane funkcije.

Za potrebe primjera jednodimenzionalne minimizacije promatrat ćemo široku klasu kvazikonveksnih funkcija koje svoj infimum postižu na segmentu $[a, b]$. Kod određivanja infimuma kvazikonveksnih funkcija ne zahtjeva se ni derivabilnost, ni neprekidnost funkcije. Definicija kvazikonveksne funkcije je:

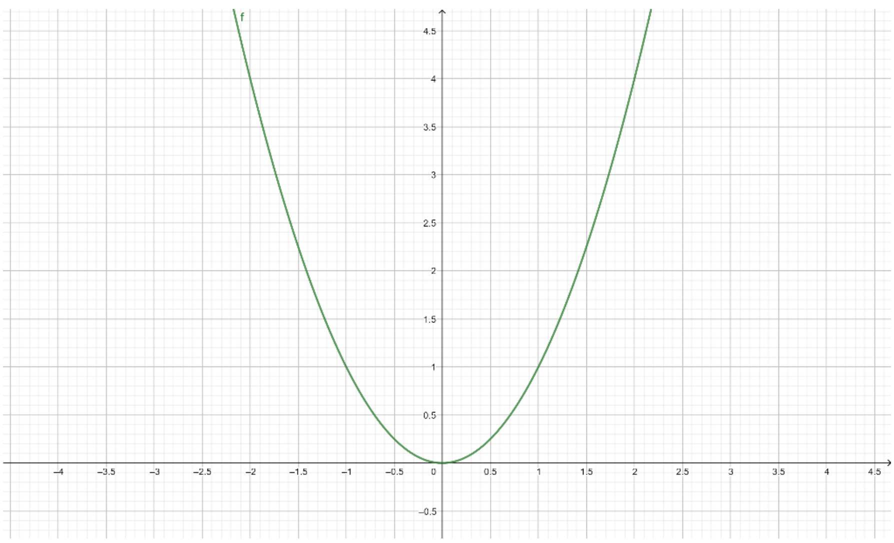
Funkcija $f : [a, b] \to \mathbb{R}$ je **strogo kvazikonveksna** na intervalu $[a, b]$ ako postiže svoj infumum $f^* = \inf\limits_{x\in[a,b]} f(x)$ u nekoj točki $x^* \in [a, b]$ i ako postoje $\alpha, \beta \in [a, b], a \le \alpha \le \beta \le b$ takvi da je:

i. f strogo monotono padajuća na $[a, \alpha)$, $a < \alpha$

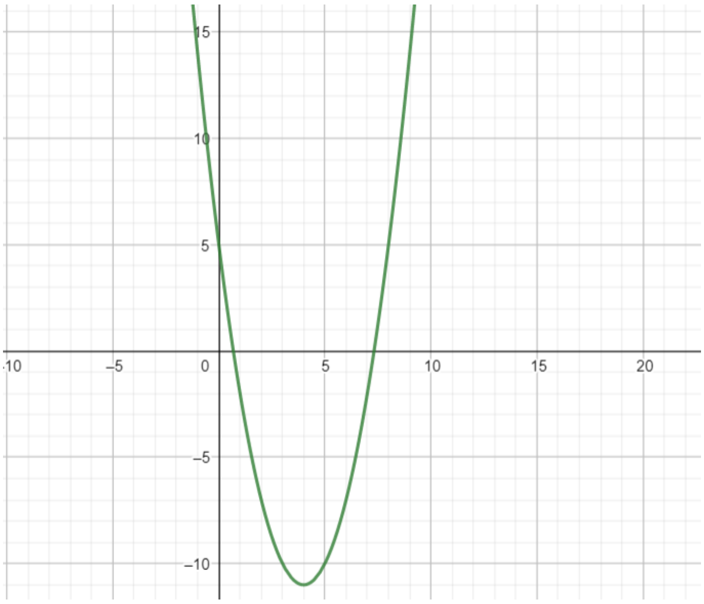
ii. f strogo monotono rastuća na $(\beta, b]$, $\beta < b$

iii. $f(u) = f^*$ za svaki $u \in (\alpha, \beta)$, $\alpha < \beta$. [1].

Na slikama ispod prikazani su grafovi primjera strogo kvazikonveksnih funkcija $f_1 : [a, b] \to \mathbb{R}, f_1(x) = x^2$ i $f_2 : [a, b] \to \mathbb{R}, f_2(x) = x^2 - 8x + 5$.



Slika 2. Grafički prikaz funkcije f_1



Slika 3. Grafički prikaz funkcije f_2

Obradit ćemo nekoliko najpoznatijih metoda jednodimenzionalne minimizacije i to metode kod kojih nije potrebno poznavanje derivacija zadanih funkcija. Spomenute metode su metoda polovljenja, metoda zlatnog reza, metoda parabole i Brentova metoda.

Metoda polovljenja se naziva tako jer u svakom koraku interval dijelimo na dvije polovice. Oznaka δ označava točnost rezultata koju želimo postići provođenjem koraka metode. Ako je δ manji, dijeljenje intervala je bliže raspolavljanju. Zadana je konveksna funkcija $f: [a, b] \to \mathbb{R}$ te je cilj odrediti točku $x^* = \underset{x \in D}{\operatorname{argmin}} f(x)$ s točnošću $\delta > 0$. Prije početka izvođenja same metode potrebno je označiti potrebne varijable po principu $x_a = a,\ x_b = b$, $x_m = \frac{1}{2}(a + b)$. Nakon ovog krećemo sa samom metodom prema sljedećim koracima:

1. Izračun: $f_a = f(x_a), f_b = f(x_b), f_m = f(x_m)$
2. U ovom koraku definiramo $x_l = \frac{1}{2}(x_a + x_m)\ i\ x_r = \frac{1}{2}(x_m + x_b)$ te izračunamo $f_l = f(x_l)$ i $f_r = f(x_r)$. Nakon toga odredimo $f_{min} = \min\{f_a, f_b, f_m, f_l, f_r\}$. Ovisno o vrijednosti f_{min} dodjeljujemo nove vrijednosti varijablama po sljedećim koracima:
    a. Ako je $f_{min} = f_a$ ili $f_{min} = f_l$ tada definiramo: $x_b = x_m,\ x_m = x_l,\ f_b = f_m$
    b. Inače, ako je $f_{min} = f_m$ tada definiramo: $x_a = x_l,\ x_b = x_r,\ f_a = f_l$, $f_b = f_r$
    c. Inače, ako je $f_{min} = f_r$ ili $f_{min} = f_b$ onda definiramo $x_a = x_m,\ x_m = x_r$, $f_a = f_m,\ f_m = f_r$

Korak 2. ponavljamo sve dok vrijednost $|x_b - x_a|$ nije manji od δ. Za minimum funkcije uzimamo onaj x^* čija vrijednost $f(x^*)$ u zadnjoj iteraciji metode bude odabrana kao f_{min}. [10]

Primjer 1.

Način funkcioniranja metode polovljenja prikazan je na primjeru odabrane funkcije $f(x) = x^3 + 4x^2 - 10$ na intervalu $[1,2]$. Karakterizacijom konveksnosti pomoću druge derivacije potvrđujemo da je promatrana funkcija konveksna. Taj podatak nam govori da minimum funkcije postoji. Postupak dokazivanja konveksnosti drugom derivacijom prikazan je ispod. Tablica pokazuje da je funkcija $f(x) = x^3 + 4x^2 - 10$ konveksna na intervalu $\langle -\frac{4}{3}, +\infty \rangle$ koji obuhvaća zadani interval $[1,2]$.

$$f''(x) = 6x + 8$$

$$f''(x) = 0, \quad 6x + 8 = 0, \quad x = -\frac{4}{3}$$

	$\langle -\infty, -\frac{4}{3} \rangle$	$\langle -\frac{4}{3}, +\infty \rangle$
$f''(x)$	$-$	$+$
$f(x)$	konkavna	konveksna

Na slici ispod vidljive su vrijednosti $x_a, x_b, x_m, f_a, f_b\ f_m, x_l, x_r, f_l, f_r$ te f_{min} kroz sve iteracije. Stupac provjera provjerava je li vrijednost $|x_b - x_a| > \delta$. Kada se tijekom izvođenja iteracija pojavi vrijednost FALSE znači da smo došli do našeg minimuma, tj. da je $|x_b - x_a| < \delta$.

Prvu iteraciju počinjemo tako da odredimo vrijednosti x_a i x_b pomoću zadanog intervala $[1,2]$.

$$x_a = 1,\, x_b = 2$$

Pomoću tih vrijednosti dolazimo do varijable $x_m = \frac{1+2}{2} = 1{,}5$ te računamo njihove vrijednosti funkcije f.

$$f_a = f(1) = -5,\ \ f_b = f(2) = 14,\, f_m = f(1.5) = 2.375$$

Dalje, kako je prikazano ispod, računamo $x_l, x_r, f_l\ i\ f_r$

$$x_l = \frac{x_a + x_m}{2} = 1.25\,, \qquad x_r = \frac{x_m + x_b}{2} = 1.75$$

$$f_l = f(1.25) = -1.79688\,, \quad f_r = f(1.75) = 7.609375\,.$$

Od vrijednosti f_a, f_b, f_m, f_l, f_r biramo najmanju te po tom dobivenom f_{min} dodjeljujemo nove vrijednosti varijablama za sljedeću iteraciju. Kako je u ovoj iteraciji $f_{min} = f_a$ mijenjamo varijable na sljedeći način

$$x_b = x_m = 1.5\ ,\ x_m = x_l = 1.25,\ f_b = f_m = 2.375\ .$$

Tim vrijednostima krećemo u drugu iteraciju i tako dalje sve dok vrijednost stupca provjera ne bude FALSE.

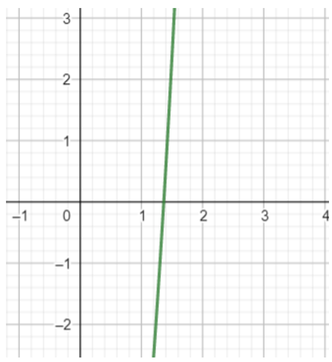
U ovom slučaju izvodimo 11 iteracija jer je vrijednost $|x_b - x_a|$ u 11. iteraciji manja od $\delta = 0.001$. Vrijednosti svih varijabli kroz iteracije prikazane su na slici 4. ispod. U zadnjem retku vrijednost $f_{min} = -5$ što je zapravo f_a tako da je x_a u ovom slučaju minimum. Konačno rješenje metode je dakle $x^* = 1$ na intervalu $[1,2]$.

Metoda polovljenja	$f(x) = x^3 + 4x^2 - 10$	interval	[1,2]	točnost	0,001

broj iteracija	xa	xb	xm	fa	fb	fm	xl	xr	fl	fr	fmin	provjera
1	1	2,00	1,5	-5	14	2,375	1,25	1,75	-1,79688	7,609375	-5	TRUE
2	1	1,5	1,25	-5	2,375	-1,79688	1,125	1,375	-3,51367	0,162109	-5	TRUE
3	1	1,25	1,125	-5	-1,79688	-3,51367	1,0625	1,1875	-4,28491	-2,68481	-5	TRUE
4	1	1,125	1,0625	-5	-3,51367	-4,28491	1,03125	1,09375	-4,64938	-3,9064	-5	TRUE
5	1	1,0625	1,03125	-5	-4,28491	-4,64938	1,015625	1,046875	-4,82641	-4,46889	-5	TRUE
6	1	1,03125	1,015625	-5	-4,64938	-4,82641	1,0078125	1,0234375	-4,91363	-4,73833	-5	TRUE
7	1	1,015625	1,0078125	-5	-4,82641	-4,91363	1,00390625	1,0117188	-4,95692	-4,87013	-5	TRUE
8	1	1,007813	1,0039063	-5	-4,91363	-4,95692	1,001953125	1,005859375	-4,97849	-4,93531	-5	TRUE
9	1	1,003906	1,0019531	-5	-4,95692	-4,97849	1,0009765625	1,0029296875	-4,98925	-4,96771	-5	TRUE
10	1	1,001953	1,0009766	-5	-4,97849	-4,98925	1,0004882813	1,0014648438	-4,99463	-4,98387	-5	TRUE
11	1	1,000977	1,0004883	-5	-4,98925	-4,99463	1,0002441406	1,0007324219	-4,99731	-4,99194	-5	FALSE

Slika 4. Primjer korištenja metode polovljenja

Na slici ispod prikazan je grafički prikaz zadane funkcije gdje se može provjeriti rezultat dobiven metodom polovljenja funkcije f na intervalu $[1,2]$ $x^* = 1$.



Slika 5. Grafički prikaz funkcije na zadanom intervalu

Metoda zlatnog reza slična je prethodnoj metodi polovljenja po načinu izvođenja, jedina razlika je u tome da točke oko polovišta zadanog intervala biramo u omjeru 'zlatnog reza'. Prema tom omjeru izražene su sljedeće formule kojima dobijemo $x_1 \; i \; x_2$ za zadanu funkciju $f: [a, b] \to \mathbb{R}$:

$$x_1 = a + \frac{3 - \sqrt{5}}{2}(b - a)$$

$$x_2 = a + b - x_1 = a + \frac{\sqrt{5} - 1}{2}(b - a).$$

Prvi korak metode je pomoću prethodnih formula definirati $x_1 \; i \; x_2$ te odrediti vrijednosti $f(x_1) \; i \; f(x_2)$. Ovisno o tim vrijednostima sljedeći korak je jedan od sljedeća dva:

1. Ako je $f(x_1) \leq f(x_2)$ tada eliminiramo sve desno, x_2 postaje novi b, a x_1 postaje x_2^* kao što je zapisano u nastavku

$$f(x_1) \leq f(x_2) \Rightarrow [a_2, b_2] = [a_1, x_2], x_2^* = x_1$$

2. Ako je $f(x_1) > f(x_2)$ tada eliminiramo sve lijevo, x_1 postaje novi a, a x_2 postaje x_2^* kao što je zapisano u nastavku

$$f(x_1) > f(x_2) \Rightarrow [a_2, b_2] = [x_1, b_1], x_2^* = x_2.$$

Sljedeći korak je simetrično s x_2^* izabrati točku $x_3 = a_2 + b_2 - x_2^*$ koja isto tako čini zlatni rez intervala $[a_2, b_2]$. Nakon ovoga ponavljamo proces iz prvog koraka gdje uspoređujemo vrijednosti $f(x_2^*) \; i \; f(x_3)$ te dobivamo novi interval $[a_3, b_3]$. Definirani koraci se ponavljaju određeni broj puta dok se ne postigne željena točnost, nakon čega se iz prosječne vrijednosti $x_1 \; i \; x_2$ varijabli, te iz prosječne vrijednosti $f(x_1) \; i \; f(x_2)$ definiraju x^* i y^* koordinata točke minimuma zadane funkcije. [3]

Primjer 2.

Ovu metodu prikazujemo na primjeru funkcije $f(x) = x^2 - 6x + 15$ sa zadanim intervalom $[0,4]$. Prije izvođenja same metode potvrđujemo da je promatrana funkcija konveksna. Kako se radi o kvadratnoj funkciji dovoljno je provjeriti je li vodeći koeficijent $a > 0$ što nam govori da je funkcija konveksnog oblika te da ima minimum. Ukoliko je $a < 0$ radi se o konkavnoj funkciji koja nema minimum. U našem primjeru $a = 1$ što potvrđuje da je funkcija $f(x) = x^2 - 6x + 15$ konveksna te da ima minimum na području domene funkcije.

Način rada ove metode implementirali smo kao Python programsko rješenje koje metodom zlatnog reza pronalazi minimum kvadratnih funkcija. Cijeli kod prikazan je na slici 6.

Metoda zlatnog reza.py - C:\Users\Korisnik\Desktop\Metoda zlatnog reza.py (3.9.5)

File Edit Format Run Options Window Help

```
#Metoda zlatnog reza

#Unos koeficijenata odabrane kvadratne funkcije
import math
print("Unos kvadratne funkcije")
fjaa=float(input("Unesite vodeći koeficijent kvadratne funkcije a: "))
if fjaa <= 0:
    print("Unesena funkcija nije konveksna, nema minimum")
    fjaa=float(input("Unesite vodeći koeficijent kvadratne funkcije a: "))
fjab=float(input("Unesite linearni koeficijent kvadratne funkcije b: "))
fjac=float(input("Unesite slobodni koeficijent kvadratne funkcije c: "))

#Unos intervala i željene točnosti za provođenje metode
print("Unos intervala")
a=round(float(input("Unesi a ")),5)
b=round(float(input("Unesi b ")),5)
print("Željena točnost")
e=float(input("Unesite točnost "))

#Računanje x1 i x2 prema formulama za omjer zlatnog reza
x1=round(a+((3-math.sqrt(5))/2)*(b-a),5)
x2=round(a+((math.sqrt(5)-1)/2)*(b-a),5)
i=0

#Ispis zaglavlja tablice za iteracije metode
print("------------------------------------------------------------------------")
print("Broj iteracije",'  ',"{:<10}{:<10}{:<10}{:<10}{:<10}{:<10}".format('a','b','x1','x2','fx1','fx2'))
print("------------------------------------------------------------------------")

#Provođenje metode (ispisuje podatke koje dobije u svakoj iteraciji)
while (x2-x1) >= e:
    i+=1
    fx1=round((x1**2)*fjaa+x1*fjab+fjac,5)
    fx2=round((x2**2)*fjaa+x2*fjab+fjac,5)
    print('     ',"{:<10}{:<10}{:<10}{:<10}{:<10}{:<10}{:<10}".format(i,a,b,x1,x2,fx1,fx2))
    if fx1 <= fx2:
        b=round(x2,5)
        x2=round(x1,5)
        x1=round(a+b-x2,5)
    else:
        a=round(x1,5)
        x1=round(x2,5)
        x2=round(a+b-x1,5)

#Konačni rezultat
x=round((x1+x2)/2,5)
y=round((fx1+fx2)/2,5)
print("------------------------------------------------------------------------")
print()
print("KONAČNI REZULTAT")
print("MIN(",x,',',y,')')
print()
print("Minimum zadane funkcije f(x)=",'(',fjaa,')',"x^2 +",'(',fjab,')',"x +",'(',fjac,')'," ima koordinate ",x," i ",y,".")
```

Slika 6. Programsko rješenje za metodu zlatnog reza

U nastavku su detaljno objašnjeni pojedini dijelovi programskog koda. Prije svega uključujemo Python matematički modul `math`. Program kreće od unosa same funkcije. Ovo programsko rješenje konstruirano je za pronalaženje minimuma kvadratnih funkcija tako da se na početku traži od korisnika da unese vrijednosti a, b i c koeficijenata funkcije. Kod upisa program provjerava je li vodeći koeficijent $a > 0$, tj. provjerava konveksnost funkcije. Ukoliko uneseni a nije veći od 0 program ponovno nudi upis vodećeg koeficijenta kako bi korisnik mogao pravilno unijeti koeficijente odabrane kvadratne funkcije kojoj želi pronaći minimum. Linije za unos tog dijela prikazane su ispod.

```
#Unos koeficijenata odabrane kvadratne funkcije
import math
print("Unos kvadratne funkcije")
fjaa=float(input("Unesite vodeći koeficijent kvadratne funkcije a: "))
if fjaa <= 0:
      print("Unesena funkcija nije konveksna, nema minimum")
      fjaa=float(input("Unesite vodeći koeficijent kvadratne funkcije a: "))
fjab=float(input("Unesite linearni koeficijent kvadratne funkcije b: "))
fjac=float(input("Unesite slobodni koeficijent kvadratne funkcije c: "))
```

Nakon unosa funkcije potrebno je unijeti podatke o intervalu (a, b) na kojem tražimo minimum prethodno unesene funkcije te točnost koju želimo postići završnim rezultatom metode. Kako bi nam bilo jednostavnije i elegantnije provođenje metode te vrijednosti odmah zaokružujemo kao što je vidljivo ispod s funkcijom `round`.

```
#Unos intervala i željene točnosti za provođenje metode
print("Unos intervala")
a=round(float(input("Unesi a ")),5)
b=round(float(input("Unesi b ")),5)
print("Željena točnost")
e=float(input("Unesite točnost "))
```

Dalje računamo x_1 i x_2 po prethodno navedenim formulama, te inicijaliziramo brojač `i` kako bi pratili broj iteracija tijekom izvođenja metode.

```
#Računanje x1 i x2 prema formulama za omjer zlatnog reza
x1=round(a+((3-math.sqrt(5))/2)*(b-a),5)
x2=round(a+((math.sqrt(5)-1)/2)*(b-a),5)
i=0
```

Prije samog izvođenja postavljamo zaglavlje tablice u koju će metoda ispisivati vrijednosti varijabli u svakoj iteraciji.

```
#Ispis zaglavlja tablice za iteracije metode
print("-----------------------------------------------------------------")
print("Broj iteracije", ' ', "{:<10}{:<10}{:<10}{:<10}{:<10}{:<10}".format(
'a','b','x1','x2','fx1','fx2'))
print("-----------------------------------------------------------------")
```

Ključni dio programa je izvođenje same metode koje je implementirano petljom `while`. Petlja će se izvoditi sve dok se ne postigne željena točnost završnog rezultata. U sklopu petlje provjeravamo je li $f(x_1) \leq f(x_2)$ ili $f(x_1) > f(x_2)$ te ovisno o tomu računamo nove vrijednosti varijabli kako je prikazano ispod.

```
#Provođenje metode (ispisuje podatke koje dobije u svakoj iteraciji)
while (x2-x1) >= e:
    i+=1
    fx1=round((x1**2)*fjaa+x1*fjab+fjac,5)
    fx2=round((x2**2)*fjaa+x2*fjab+fjac,5)
    print('',"{:<10}{:<10}{:<10}{:<10}{:<10}{:<10}{:<10}".format(i,a,b,x1,
x2,fx1,fx2))
    if fx1 <= fx2:
        b=round(x2,5)
        x2=round(x1,5)
        x1=round(a+b-x2,5)
    else:
        a=round(x1,5)
        x1=round(x2,5)
        x2=round(a+b-x1,5)
```

Nakon što petlja, tj. metoda odradi sve potrebne iteracije računamo prosječne vrijednosti x^* i y^* te ispisujemo konačni rezultat.

```
#Konačni rezultat
x=round((x1+x2)/2,5)
y=round((fx1+fx2)/2,5)
print("-----------------------------------------------------------------")
print()
print("KONAČNI REZULTAT")
print("MIN(",x,',',y,')')
print()
print("Minimum zadane funkcije f(x)=",'(',fjaa,')',"x^2 +",'(',fjab,')',"x
+",'(',fjac,')'," ima koordinate ",x," i ",y,".")
```

Rad ovog programskog rješenja provjeren je na prethodno spomenutom primjeru funkcije $f(x) = x^2 - 6x + 15$ sa zadanim intervalom $[0,4]$ i točnosti $\epsilon = 0{,}001$. Prvo su uneseni koeficijenti kvadratne funkcije $a = 1, b = -6, c = 15$, granice zadanog intervala $a = 0, b = 4$ te željena točnost $\epsilon = 0{,}001$. Nakon tih unosa u tablici je za svaku iteraciju ispisan broj iteracije te vrijednosti varijabli a, b, x_1, x_2, $f(x_1)\ i\ f(x_2)$. Nakon što je postignuta željena točnost metoda se zaustavlja te ispisuje konačan rezultat minimuma zadane funkcije. Konačan rezultat dobije se tako da se iz prosječne vrijednosti $x_1\ i\ x_2$ varijabli, te iz prosječne vrijednosti $f(x_1)\ i\ f(x_2)$ definiraju x^* i y^* koordinate točke minimuma zadane funkcije Cijelo rješenje programa vidljivo je ispod na slici 7.

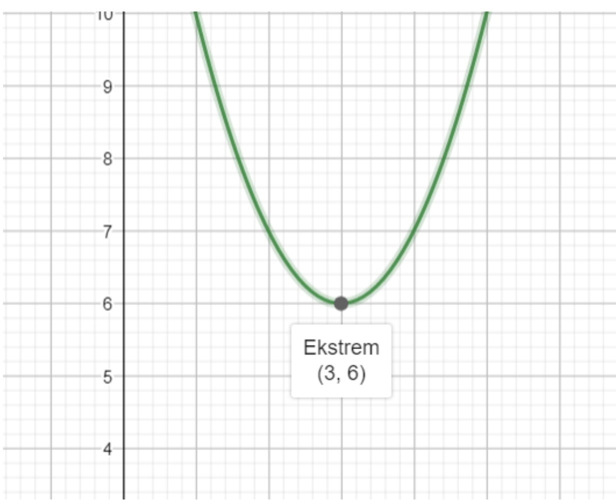
```
IDLE Shell 3.9.5
File  Edit  Shell  Debug  Options  Window  Help
Python 3.9.5 (tags/v3.9.5:0a7dcbd, May  3 2021, 17:27:52) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
=========== RESTART: C:\Users\Korisnik\Desktop\Metoda zlatnog reza.py ==========
Unos kvadratne funkcije
Unesite vodeći koeficijent kvadratne funkcije a: -2
Unesena funkcija nije konveksna, nema minimum
Unesite vodeći koeficijent kvadratne funkcije a: 1
Unesite linearni koeficijent kvadratne funkcije b: -6
Unesite slobodni koeficijent kvadratne funkcije c: 15
Unos intervala
Unesi a 0
Unesi b 4
Željena točnost
Unesite točnost 0.001
--------------------------------------------------------------------------
Broj iteracije    a          b          x1         x2         fx1        fx2
--------------------------------------------------------------------------
      1         0.0        4.0        1.52786    2.47214    8.1672     6.27864
      2         1.52786    4.0        2.47214    3.05572    6.27864    6.0031
      3         2.47214    4.0        3.05572    3.41642    6.0031     6.17341
      4         2.47214    3.41642    2.83284    3.05572    6.02794    6.0031
      5         2.83284    3.41642    3.05572    3.19354    6.0031     6.03746
      6         2.83284    3.19354    2.97066    3.05572    6.00086    6.0031
      7         2.83284    3.05572    2.9179     2.97066    6.00674    6.00086
      8         2.9179     3.05572    2.97066    3.00296    6.00086    6.00001
      9         2.97066    3.05572    3.00296    3.02342    6.00001    6.00055
      10        2.97066    3.02342    2.99112    3.00296    6.00008    6.00001
      11        2.99112    3.02342    3.00296    3.01158    6.00001    6.00013
      12        2.99112    3.01158    2.99974    3.00296    6.0        6.00001
      13        2.99112    3.00296    2.99434    2.99974    6.00003    6.0
--------------------------------------------------------------------------

KONAČNI REZULTAT
MIN( 2.99865 , 6.00001 )

Minimum zadane funkcije f(x)= ( 1.0 ) x^2 + ( -6.0 ) x + ( 15.0 )  ima koordinate  2.99865  i  6.00001 .
>>>
```

Slika 7. Primjer rješenja programskim rješenjem metode zlatnog reza

Na slici 8. nalazi se grafički prikaz zadane funkcije $f(x) = x^2 - 6x + 15$ gdje možemo vidjeti da je rezultat metode zlatnog reza $\min(2.99865, 6{,}00001) \approx (3{,}6)$ ispravan, tj. da točka minimuma zadane funkcije ima koordinate (3,6).



Slika 8. Grafički prikaz korištene funkcije

Metoda parabole je još jedna jednostavna metoda za određivanje lokalnog minimuma funkcije $f: [a, c] \to \mathbb{R}$. Za početak izvođenja metode potrebno je u zadanom intervalu $[a, c]$ odabrati točku b u kojoj funkcija prima manju vrijednost nego za točke a i c. Podacima o točkama $(a, f(a)), (b, f(b))\ i\ (c, f(c))$ određujemo interpolacijski polinom pomoću Lagrangeove formule kako je prikazano formulama prikazanim ispod.

$$P_2 = f(a)p_a(x) + f(b)p_b(x) + f(c)p_c(x)$$

$$p_a(x) = \frac{(x-c)(x-b)}{(a-c)(a-b)}, \quad p_b(x) = \frac{(x-a)(x-c)}{(b-a)(b-c)}, \quad p_c(x) = \frac{(x-a)(x-b)}{(c-a)(c-b)}$$

Dobiveni polinom P_2 zapravo je aproksimacija funkcije f na zadanom intervalu $[a, c]$ te zbog toga tražimo njegov minimum, a ne minimum zadane funkcije. Stacionarnu točku dobivenog interpolacijskog polinoma drugog stupnja dobijemo sljedećom formulom

$$x_{min} = \frac{1}{2}\frac{a^2\big(f(b)-f(c)\big)+b^2\big(f(c)-f(a)\big)+c^2\big(f(a)-f(b)\big)}{a\big(f(b)-f(c)\big)+b\big(f(c)-f(a)\big)+c\big(f(a)-f(b)\big)}\ . \qquad [3]$$

Ukoliko je interpolacijski polinom dobra aproksimacija funkcije f, x minimum od polinoma P_2 bi trebao biti dobra aproksimacija x^* minimuma od f. Iako sa zadanom točnosti taj proces možemo poboljšati. Točnost $\delta,\ \varepsilon$ ili $xtol$ zadajemo u ovisnosti o željenoj preciznosti rezultata. Iterativni postupak metode započinjemo tako da odredimo vrijednost varijable dx po formuli

$$dx = \max\{a, b, c\} - \min\{a, b, c\}\ .$$

Na temelju tog odabiremo sljedeći korak metode. Ako je dx manji od zadane točnosti rezultata, tj. ako je $dx < xtol$ dobivenu vrijednost x_{min} uzimamo kao završni rezultat minimuma funkcije

$$x^* = x_{min}\ .$$

Ako je dx veći od zadane točnosti, tj. $dx > xtol$ moramo napraviti zamjenu vrijednosti varijabli za sljedeću iteraciju po principu

$$a = b, \qquad b = c, \ \ c = x\ .$$

Za taj novo dobiveni skup podataka ponovno računamo stacionarnu točku interpolacijskog polinoma x_{min} i ponavljamo prethodno definirane korake. Proces se može ponavljati dok se ne postigne zadana točnost ili dok ne uočimo male promjene u rezultatima. [11]

Primjer 3.

Opisani postupak za funkciju $f(x) = x^2 \cos(\frac{x+3\pi}{4})$ na intervalu $[1,10]$ prikazan je na slici ispod. Prije samog postupka metode potvrđujemo da je spomenuta funkcija konveksna pomoću druge derivacije kako je prikazano ispod. Uočavamo da nam je funkcija konveksna samo na drugom intervalu tako da za provedbu metode možemo uzeti manji interval od početnog $[1,10]$ pa uzimamo interval $[5,10]$.

$$f''(x) = 2\cos\left(\frac{x+3\pi}{4}\right) - x \cdot sin\left(\frac{x+3\pi}{4}\right) - \frac{x^2\cos\left(\frac{x+3\pi}{4}\right)}{16}$$

$$f''(x) = 0, \qquad x = 4.0714847379986 \approx 4.07148$$

	$\langle -\infty, 4.07148 \rangle$	$\langle 4.07148, +\infty \rangle$
$f''(x)$	$-$	$+$
$f(x)$	konkavna	konveksna

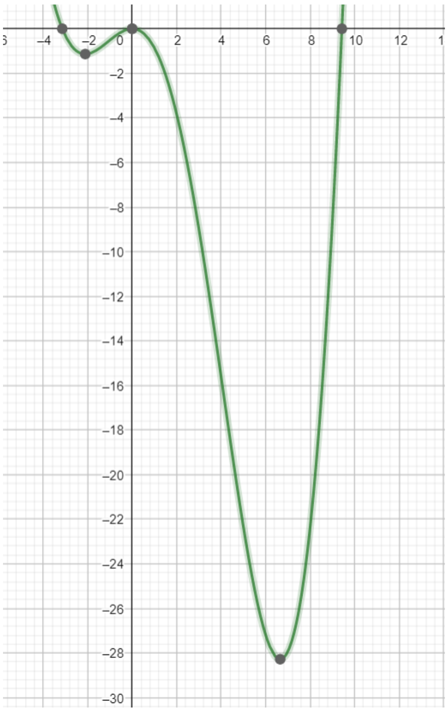
Za vrijednosti a, b, c iz intervala $[5,10]$ izračunate su vrijednosti $f(a), f(b)\ i\ f(c)$ pomoću kojih onda računamo brojnik i nazivnik za prethodno spomenutu formulu stacionarne točke interpolacijskog polinoma što je prikazano u stupcu x. Nakon toga računamo vrijednost varijable dx koja nam govori trebamo li izvoditi još iteracija. Ta postignuta točnost se provjerava u stupcu $dx < xtol$. Na kraju iteracije još računamo vrijednost $f(x)$ što je zapravo y^* varijabla minimuma funkcije $f(x) = x^2 \cos(\frac{x+3\pi}{4})$ na intervalu $[5,10]$. Kada je zadana točnost postignuta vrijednost u ćeliji je TRUE što vidimo u 10. ponavljanju te je to naš konačni rezultat, točka $(6.7, -28.3)$. Također možemo uočiti male promjene u vrijednostima zadnjih par iteracija što je dodatna potvrda da je izvođenje metode gotovo.

Jednadžba	xtol	Interval	a	b	c
$f(x) = x^2 \cos(\frac{x+3\pi}{4})$	0,001	[5,10]	5	7,5	10

Broj iteracija	a	b	c	f(a)	f(b)	f(c)	brojnik	nazivnik	dx	dx<xtol	x	f(x)
1	0	7,5	10	0	-26,094	14,213	3408,890505	367,53807	10	FALSE	4,637466	-20,0290665
2	7,5	10	4,637466	-26,094	14,213	-20,029	1665,772685	130,54302	5,362534	FALSE	6,380167	-28,11278175
3	10	4,637466	6,380167	14,213	-20,029	-28,113	1291,981764	103,02257	5,362534	FALSE	6,270382	-27,92373429
4	4,637466	6,380167	6,270382	-20,029	-28,113	-27,924	-7,597239769	-0,558021	1,742702	FALSE	6,807308	-28,24874725
5	6,380167	6,270382	6,807308	-28,113	-27,924	-28,249	-0,876052263	-0,065823	0,536926	FALSE	6,654619	-28,31572473
6	6,270382	6,807308	6,654619	-27,924	-28,249	-28,316	-1,138362145	-0,085588	0,536926	FALSE	6,65024	-28,31567969
7	6,807308	6,654619	6,65024	-28,249	-28,316	-28,316	-0,003994484	-0,0003	0,157068	FALSE	6,654229	-28,31572508
8	6,654619	6,65024	6,654229	-28,316	-28,316	-28,316	2,53957E-07	1,908E-08	0,004379	FALSE	6,654265	-28,31572508
9	6,65024	6,654229	6,654265	-28,316	-28,316	-28,316	2,1641E-08	1,626E-09	0,004025	FALSE	6,654265	-28,31572508
10	6,654229	6,654265	6,654265	-28,316	-28,316	-28,316	4,64137E-15	3,488E-16	3,62E-05	TRUE	6,654265	-28,31572508
11	6,654265	6,654265	6,654265	-28,316	-28,316	-28,316	-7,90481E-21	-5,94E-22	9,48E-08	TRUE	6,654265	-28,31572508
12	6,654265	6,654265	6,654265	-28,316	-28,316	-28,316	-1,23728E-20	-9,3E-22	8,07E-08	TRUE	6,654265	-28,31572508

Slika 9. Primjer korištenja metode parabole

Na slici ispod nalazi se grafički prikaz odabrane funkcije $f(x) = x^2 \cos(\frac{x+3\pi}{4})$ gdje možemo uočiti da je metoda parabole brz i efikasan način do ispravnog rezultata. Lokalni minimum na intervalu $[5,10]$ uistinu je točka $(6.7, -28.3)$.



Slika 10. Grafički prikaz funkcije za metodu parabole

Brentova metoda za jednodimenzionalnu minimizaciju koja se najčešće koristi u praksi kombinira metodu zlatnog reza i metodu parabole. Iako je metoda zlatnog reza pogodna i za minimizaciju funkcije u najgorim mogućim uvjetima, u nekim slučajevima metoda parabole to može brže odraditi. Tako u slučaju funkcije paraboličnog oblika u okolini minimuma metoda parabole konvergira minimumu u svega nekoliko iteracija. Također metoda zlatnog reza garantirano nalazi minimum funkcije, dok to kod metode parabole nije sigurno. Tako ih Brentova metoda kompromisno koristi obje ovisno o tome koja se metoda u danom koraku učini pogodnijom. Kod kombiniranja metoda važno je paziti na nepotrebno dodatno izračunavanje vrijednosti funkcije prelaska s metode na metodu, pratiti zaustavni kriterij te snažno pravilo za odabir metode za sljedeću iteraciju. [3]

Ključno je prilikom optimizacijskog procesa metode pratiti sljedećih 6 točaka:

- a – lijeva granica intervala (a, b)
- b – desna granica intervala (a, b)
- x – najmanja vrijednost funkcije
- w – 2. najmanja vrijednost (nakon x)

- v – vrijednost prije w (3. najmanja vrijednost)
- u – točka u kojoj je vrijednost funkcije izračunata zadnji put.

Može se pratiti i točka m koja je u sredini intervala (a, b), tj. polovište intervala, ali se u njoj vrijednost funkcije ne mora računati. [3]

Interpolacija metodom parabole radi se kroz točke x, v i w, a kako bi korak dobiven na taj način bio prihvatljiv moraju se zadovoljiti sljedeći uvjeti:

- Sljedeća aproksimacija mora biti unutar intervala (a, b)
- Udaljenost sljedeće aproksimacije od x mora biti manja od polovine udaljenosti zadnje dvije aproksimacije, ukoliko to nije slučaj vraća se na metodu zlatnog reza.

Tipični zaustavi kriterij za ovu metodu je da su vrijednosti a i b udaljenje $2 \cdot x \cdot \varepsilon$. [3]

Primjer 4.

Najčešće se prve iteracije rješavaju metodom zlatnog reza dok točke v, w i x ne poprime različite vrijednosti kako bi mogli rješavati sljedeće iteracije metodom parabole. To je vidljivo i na primjeru funkcije $f(x) = \frac{x^3}{3} - \frac{x^2}{2} - x - 1$ prikazanom u tablici ispod. Zadana funkcija je konveksna što nam govori da minimum na odabranom intervalu postoji. Minimum tražimo na zadanom intervalu $[1,2]$ s točnosti $\varepsilon = 10^{-6}$. Vidimo kako su prve dvije iteracije rješavane metodom zlatnom reza, a sve iduće metodom parabole. Konačno rješenje je $x^* = 1{,}618034$ pronađeno u šestoj iteraciji metodom parabole. Također uočavamo da je metoda minimum našla već u prvoj iteraciji, ali to ne zna dok se interval dovoljno ne smanji. [12]

Tablica 1. Primjer minimizacije Brentovom metodom

Iteracija	u	Odabrana metoda
1	1,618034	Metoda zlatnog reza
2	1,763932	Metoda zlatnog reza
3	1,612758	Metoda parabole
4	1,617924	Metoda parabole
5	1,618034	Metoda parabole
6	1,618034	Metoda parabole

(Izvor: Veliz, 2020.)

3.3. Uvod u višedimenzionalnu minimizaciju

Za početak prikazane su osnovne, dobro poznate definicije i činjenice o ekstremima funkcije više varijabli.

Neka je zadana funkcija $f\colon S \to \mathbb{R}, S \subseteq \mathbb{R}^n, n \geq 2, n \in \mathbb{N}$. Funkcija f ima u točki $c \in S$ **lokalni minimum** ako postoji okolina O_c oko točke c gdje je za svaki $x = (x_{1,}x_2, \dots, x_n) \in O_c$ vrijedi $f(c) \leq f(x)$. Funkcija je **diferencijabilna** na zadanom skupu S ako postoje sve prve parcijalne derivacije i one su neprekidne. Za odabrani x , vektor

$$\nabla f(x) = (\frac{\partial f(x)}{\partial x_1}, \frac{\partial f(x)}{\partial x_2}, \dots, \frac{\partial f(x)}{\partial x_n})$$

je **gradijent funkcije** f u točki x. Iz ove definicije možemo uočiti da su skalarne projekcije gradijenta parcijalne derivacije prvog reda zadane funkcije f. [9]

Ako je točka c lokalni ekstrem funkcije f njezin gradijent je nulvektor $\nabla f(c) = \vec{0}$. Sve točke kojima je gradijent nulvektor nazivamo **stacionarnim točkama** . Iz toga slijedi nužan, ali ne i dovoljan **uvjet lokalnih ekstrema**. Ekstrem funkcije mora biti stacionarna točka, ali stacionarna točka ne mora nužno biti ekstrem funkcije f. Takve točke kojima je gradijent nulvektor, a nisu ekstremi nazivamo **sedlastim točkama ili točkama pregiba** u kojima se mijenja zakrivljenost funkcije. U tim točkama funkcija prelazi iz konkavne u konveksnu ili obrnuto. Ispod je prikazana Hesseova matrica u kojoj se nalaze elementi parcijalne derivacije drugog reda funkcije f. [9]

$$H(x_1, x_2, \dots, x_n) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}$$

Prethodno definiranim činjenicama o ekstremima funkcije više varijabli skupili smo potrebne podatke za nalaženje tih ekstrema. Dakle, ako je c stacionarna točka dva puta diferencijabilne funkcije f i D_k $(k \in \mathbb{N})$ glavne subdeterminante pridružene Hesseovoj matrici u točki c onda funkcija f

a) ako su $D_k > 0$ ima lokalni minimum u c

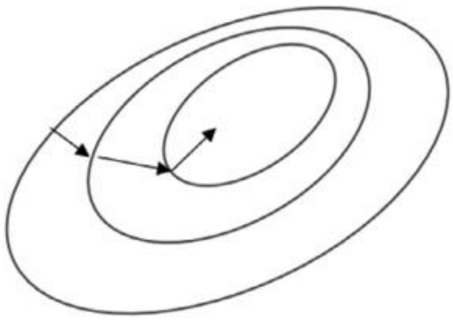
b) za parne k, $D_k > 0$, a za neparne k, $D_k < 0$ ima lokalni maksimum u točki c

c) ako za barem jedan k $D_k < 0$ ili postoje neparni k i l $D_k > 0$ i $D_l < 0$ nema lokalnih ekstrema. [9]

3.4. Gradijentne metode

Općenito, gradijentne metode su numeričke metode optimizacije u kojima se koristi gradijent. Geometrijski gledano gradijent je okomit na nivo plohu $f(x) = a$ iz čega proizlazi činjenica da gradijent funkcije $\nabla f(x)$ ima smjer i orijentaciju najbržeg rasta funkcije. To znači da $-\nabla f(x)$ ima smjer i orijentaciju najbržeg pada funkcije. [9]

Cilj gradijentnih metoda je zadanom početnom aproksimacijom doći do sljedeće aproksimacije koristeći gradijent. Vizualni prikaz te ideje vidljiv je na slici ispod.



Slika 11. Vizualni prikaz ideje gradijentnih metoda

Korak aproksimacije za minimum dan je sljedećom formulom

$$x^{(k+1)} = x^k - \lambda_k \nabla f(x^k).$$

Postoje različite mogućnosti odabira koraka λ_k kako bi se osigurala konvergencija niza iteracija prema kritičnoj točki zadane funkcije. Ukoliko izaberemo čvrsti korak u svim iteracijama dobivamo metodu konstantnog spusta. Ključno je da kod računanja minimuma vrijednost funkcije u svakoj sljedećoj iteraciji bude manja, tj. da se gradijent sve više bliži nuli. Kriterij zaustavljanja algoritma može biti formula ispod.

$$\sqrt{\sum_{i=1}^{n} \left(\frac{\partial f}{\partial x_i}(x^{(k)}) \right)^2} \leq \varepsilon$$

Ako je u $x^{k+1} = x^k - \lambda_k \nabla f(x^k)$ korak $\lambda_k > 0$ biramo tako da nekom od metoda jednodimenzionalne optimizacije minimiziramo funkciju jedne varijable konstruiranu formulom ispod. [9]

$$g(\lambda) = f\left(x^{(k)} - \lambda \nabla f\left(x^{(k)}\right)\right), \lambda \geq 0$$

Na taj način dobiva se **metoda najbržeg spusta** koju prikazujemo u primjeru ispod.

Primjer 5.

Metodom najbržeg spusta minimizirajmo funkciju $f(x,y) = x^2 + 2x + y^2 - 4y + 3$ počevši od točke $(-3,-3)$. Prije izvođenja metode dokazano je da je zadana funkcija konveksna.

$$\text{gradijent } \nabla f(x,y) = \begin{bmatrix} 2x+2 \\ 2y-4 \end{bmatrix} \qquad \nabla f(-3,-3) = \begin{bmatrix} 2\cdot(-3)+2 \\ 2\cdot(-3)-4 \end{bmatrix} = \begin{bmatrix} -4 \\ -10 \end{bmatrix}$$

$$\begin{bmatrix} x^1 \\ y^1 \end{bmatrix} = \begin{bmatrix} x^0 \\ y^0 \end{bmatrix} - \lambda \nabla f(x^0,y^0) = \begin{bmatrix} -3 \\ -3 \end{bmatrix} - \lambda \begin{bmatrix} -4 \\ -10 \end{bmatrix} = \begin{bmatrix} -3+4\lambda \\ -3+10\lambda \end{bmatrix}$$

$$\begin{aligned} g(\lambda) = f(x^1,y^1) &= (-3+4\lambda)^2 + 2\cdot(-3+4\lambda) + (-3+10\lambda)^2 - 4(-3+10\lambda) + 3 \\ &= 116\lambda^2 - 116\lambda + 27 \end{aligned}$$

$$g'(\lambda) = 232\lambda - 116$$

$$g'(\lambda) = 0 \text{ za } \lambda = \frac{1}{2}$$

Za korak $\lambda = \frac{1}{2}$ dalje računamo

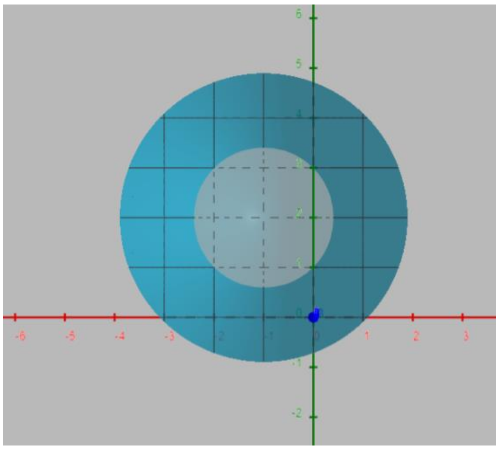
$$\begin{bmatrix} x^1 \\ y^1 \end{bmatrix} = \begin{bmatrix} -3+4\cdot\frac{1}{2} \\ -3+10\cdot\frac{1}{2} \end{bmatrix} = \begin{bmatrix} -1 \\ 2 \end{bmatrix}.$$

Provjerimo gradijent funkcije u točki $(-1,2)$:

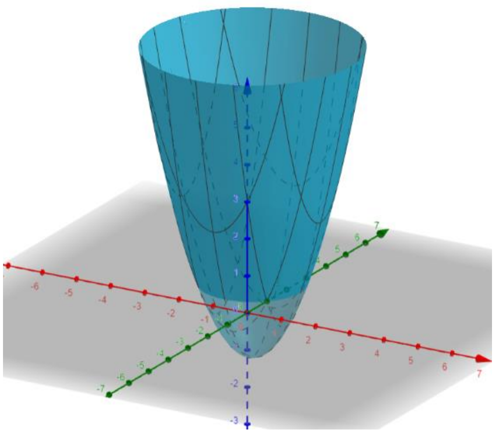
$$\nabla f(-1,2) = \begin{bmatrix} 2\cdot(-1)+2 \\ 2\cdot 2-4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

Očito je minimum funkcije u točki $(-1,2)$ i iznosi

$$f(-1,2) = (-1)^2 + 2\cdot(-1) + 2^2 - 4\cdot 2 + 3 = -2\ .$$



Slika 12. Prikaz funkcije na xy osima



Slika 13. Prikaz funkcije u prostoru

Primjer 6.

Riješimo isti primjer koristeći istu početnu točku i korak $\lambda = 0.1$ (metoda konstantnog spusta).

$$\text{gradijent } \nabla f(x,y) = \begin{bmatrix} 2x+2 \\ 2y-4 \end{bmatrix} \qquad \nabla f(-3,-3) = \begin{bmatrix} 2\cdot(-3)+2 \\ 2\cdot(-3)-4 \end{bmatrix} = \begin{bmatrix} -4 \\ -10 \end{bmatrix}$$

$$\begin{bmatrix} x^1 \\ y^1 \end{bmatrix} = \begin{bmatrix} x^0 \\ y^0 \end{bmatrix} - \lambda \nabla f(x^0, y^0) = \begin{bmatrix} -3 \\ -3 \end{bmatrix} - 0.1 \begin{bmatrix} -4 \\ -10 \end{bmatrix} = \begin{bmatrix} -3+0.4 \\ -3+1 \end{bmatrix} = \begin{bmatrix} -2.6 \\ -2 \end{bmatrix}$$

Usporedimo vrijednost funkcije u početnoj i dobivenoj aproksimaciji

$$f(-3,-3) = (-3)^2 + 2\cdot(-3) + (-3)^2 - 4\cdot(-3) + 3 = 27$$

$$f(-2.6,-2) = (-2.6)^2 + 2\cdot(-2.6) + (-2)^2 - 4\cdot(-2) + 3 = 16.56$$

Očito se vrijednost funkcije smanjuje, tako da prelazimo na sljedeću aproksimaciju.

$$\begin{bmatrix} x^2 \\ y^2 \end{bmatrix} = \begin{bmatrix} x^1 \\ y^1 \end{bmatrix} - \lambda \nabla f(x^1, y^1) = \begin{bmatrix} -2.6 \\ -2 \end{bmatrix} - 0.1 \begin{bmatrix} 2\cdot(-2.6)+2 \\ 2\cdot(-2)-4 \end{bmatrix} = \begin{bmatrix} -2.6+0.32 \\ -2+0.8 \end{bmatrix} = \begin{bmatrix} -2.28 \\ -1.2 \end{bmatrix}$$

Usporedimo vrijednosti funkcije u zadnje dvije aproksimacije:

$$f(-2.28,-1.2) = (-2.28)^2 + 2\cdot(-2.28) + (-1.2)^2 - 4\cdot(-1.2) + 3 = 9.8484$$

$$f(x^2,y^2) < f(x^1,y^1).$$

$$\begin{bmatrix} x^3 \\ y^3 \end{bmatrix} = \begin{bmatrix} x^2 \\ y^2 \end{bmatrix} - \lambda \nabla f(x^2, y^2) = \begin{bmatrix} -2.28 \\ -1.2 \end{bmatrix} - 0.1 \begin{bmatrix} 2\cdot(-2.28)+2 \\ 2\cdot(-1.2)-4 \end{bmatrix} = \begin{bmatrix} -2.28+0.256 \\ -1.2+0.64 \end{bmatrix} = \begin{bmatrix} -2.024 \\ -0.56 \end{bmatrix}$$

Izračunajmo vrijednost funkcije u x^3, y^3

$$f(-2.024,-0.56) = (-2.024)^2 + 2\cdot(-2.024) + (-0.56)^2 - 4\cdot(-0.56) + 3 = 5.602176$$

Četvrta aproksimacija je

$$\begin{bmatrix} x^4 \\ y^4 \end{bmatrix} = \begin{bmatrix} x^3 \\ y^3 \end{bmatrix} - \lambda \nabla f(x^3, y^3) = \begin{bmatrix} -2.024 \\ -0.56 \end{bmatrix} - 0.1 \begin{bmatrix} 2\cdot(-2.024)+2 \\ 2\cdot(-0.56)-4 \end{bmatrix} = \begin{bmatrix} -2.024+0.2048 \\ -0.56+0.512 \end{bmatrix}$$
$$= \begin{bmatrix} -1.8192 \\ -0.048 \end{bmatrix}$$

$$f(-1.8192,-0.048) = (-1.8192)^2 + 2\cdot(-1.8192) + (-0.048)^2 - 4\cdot(-0.048) + 3$$
$$= 2.865393\ .$$

Preostale aproksimacije vidljive na slici ispod dokazuju da je gradijent u svakoj sljedećoj aproksimaciji bliži i bliži nuli (nulvektoru). Vrijednosti x i y sve se više približavaju stvarnoj točki minimuma $(-1{,}2)$ za koju vrijednost funkcije iznosi -2.

Gradijentna metoda konstantnog spusta

$$f(x, y) = x^2 + 2x + y^2 - 4y + 3$$

Početna točka	Korak λ
T(-3,-3)	0,1

Broj aproksimacije k	x^{k-1}	y^{k-1}	$\nabla f(x^{k-1})$	$\nabla f(y^{k-1})$	x^k	y^k	$f(x^k, y^k)$
1	-3	-3	-4	-10	-2,6	-2	16,56
2	-2,6	-2	-3,2	-8	-2,28	-1,2	9,878400
3	-2,28	-1,2	-2,56	-6,4	-2,024	-0,56	5,602176
4	-2,024	-0,56	-2,048	-5,12	-1,8192	-0,048	2,865393
5	-1,8192	-0,048	-1,6384	-4,096	-1,65536	0,3616	1,113851
6	-1,65536	0,3616	-1,31072	-3,2768	-1,52429	0,68928	-0,007135
7	-1,52429	0,68928	-1,048576	-2,62144	-1,41943	0,951424	-0,724567
8	-1,41943	0,951424	-0,8388608	-2,097152	-1,33554	1,161139	-1,183723
9	-1,33554	1,161139	-0,67108864	-1,6777216	-1,26844	1,328911	-1,477582
10	-1,26844	1,328911	-0,53687091	-1,34217728	-1,21475	1,463129	-1,665653
11	-1,21475	1,463129	-0,42949673	-1,07374182	-1,1718	1,570503	-1,786018
12	-1,1718	1,570503	-0,34359738	-0,85899346	-1,13744	1,656403	-1,863051
13	-1,13744	1,656403	-0,27487791	-0,68719477	-1,10995	1,725122	-1,912353
14	-1,10995	1,725122	-0,21990233	-0,54975581	-1,08796	1,780098	-1,943906
15	-1,08796	1,780098	-0,17592186	-0,43980465	-1,07037	1,824078	-1,964100
16	-1,07037	1,824078	-0,14073749	-0,35184372	-1,05629	1,859263	-1,977024
17	-1,05629	1,859263	-0,11258999	-0,28147498	-1,04504	1,88741	-1,985295
18	-1,04504	1,88741	-0,09007199	-0,22517998	-1,03603	1,909928	-1,990589
19	-1,03603	1,909928	-0,07205759	-0,18014399	-1,02882	1,927942	-1,993977
20	-1,02882	1,927942	-0,05764608	-0,14411519	-1,02306	1,942354	-1,996145
21	-1,02306	1,942354	-0,04611686	-0,11529215	-1,01845	1,953883	-1,997533
22	-1,01845	1,953883	-0,03689349	-0,09223372	-1,01476	1,963107	-1,998421
23	-1,01476	1,963107	-0,02951479	-0,07378698	-1,01181	1,970485	-1,998989
24	-1,01181	1,970485	-0,02361183	-0,05902958	-1,00944	1,976388	-1,999353
25	-1,00944	1,976388	-0,01888947	-0,04722366	-1,00756	1,981111	-1,999586
26	-1,00756	1,981111	-0,01511157	-0,03777893	-1,00604	1,984888	-1,999735
27	-1,00604	1,984888	-0,01208926	-0,03022315	-1,00484	1,987911	-1,999830
28	-1,00484	1,987911	-0,00967141	-0,02417852	-1,00387	1,990329	-1,999891
29	-1,00387	1,990329	-0,00773713	-0,01934281	-1,00309	1,992263	-1,999931
30	-1,00309	1,992263	-0,0061897	-0,01547425	-1,00248	1,99381	-1,999956
31	-1,00248	1,99381	-0,00495176	-0,0123794	-1,00198	1,995048	-1,999972
32	-1,00198	1,995048	-0,00396141	-0,00990352	-1,00158	1,996039	-1,999982
33	-1,00158	1,996039	-0,00316913	-0,00792282	-1,00127	1,996831	-1,999988
34	-1,00127	1,996831	-0,0025353	-0,00633825	-1,00101	1,997465	-1,999993
35	-1,00101	1,997465	-0,00202824	-0,0050706	-1,00081	1,997972	-1,999995
36	-1,00081	1,997972	-0,00162259	-0,00405648	-1,00065	1,998377	-1,999997
37	-1,00065	1,998377	-0,00129807	-0,00324519	-1,00052	1,998702	-1,999998
38	-1,00052	1,998702	-0,00103846	-0,00259615	-1,00042	1,998962	-1,999999
39	-1,00042	1,998962	-0,00083077	-0,00207692	-1,00033	1,999169	-1,999999
40	-1,00033	1,999169	-0,00066461	-0,00166153	-1,00027	1,999335	-1,999999

Slika 14. Izračun primjera metode konstantnog spusta

Metoda zrcalnog spusta

Krenimo s poznatom pretpostavkom da je f konveksna funkcija u varijabli $x \in S$ gdje je S konveksan i zatvoren skup. Za bolje razumijevanje ove metode potrebno je definirati neke pojmove prikazane u nastavku.

Neka je f konveksna funkcija na $\mathbb{R}^n, \partial f(x)$ je njezin subgradijent u x ako je

$$f(y) \geq f(x) + \langle \partial f(x), y - x \rangle, \ \forall y \in \mathbb{R}^n.$$

Funkcija f je **Lipschitz neprekidna** s konstantom K na konveksnom skupu S ako za $\forall x \epsilon S$ vrijedi $\|\partial f(x)\|_* \leq K$ gdje je ∂f bilo koji subgradijent od f [13]

Funkcija $d: \mathbb{R}^n \times \mathbb{R}^n \to [0, \infty\rangle$ je **kvazimetrička funkcija** ako vrijedi $\{\forall x, y \in \mathbb{R}^n\}$ $d(x, y) = 0 \Leftrightarrow x = y$. Sada definiramo iteraciju metode zrcalnog spusta s duljinom $\alpha > 0$ sljedećim izrazom

$$Mirr(\alpha, x, \partial f) = argmin_{y \in S}\{d(x, y) + \langle \alpha \partial f(x), y - x \rangle\}$$

gdje je ∂f subgradijent neke funkcije.

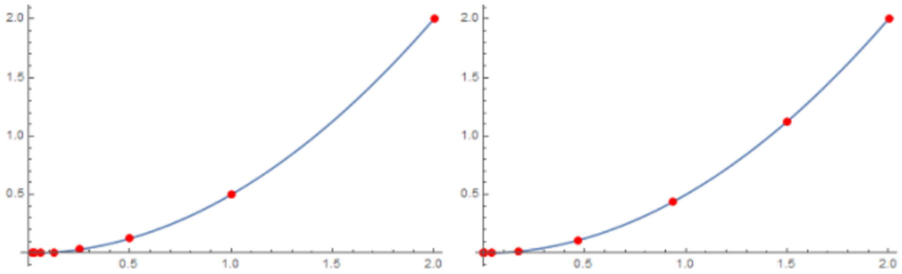
Teorem za ocjenjivanje pogreške metode

Vrijedi $f(\bar{x}) - f(x^*) \leq \frac{\sqrt{2\Theta}K}{\sqrt{k}}$ gdje ε-aproksimaciju točke x^* postižemo u $\Omega(\frac{2\Theta K^2}{\varepsilon^2})$ iteracija (aritmetička sredina prvih k iteracija $\overline{x} = \frac{1}{k}\sum_{i=0}^{k-1} x_i$, Θ bilo koja gornja međa na $d(x_0, x^*)$, točka minimuma x^* , $\alpha = \frac{\sqrt{2\Theta}}{K\sqrt{k}}$).

Za razliku od gradijentne metode najbržeg spusta koja brzo konvergira dok je trenutna iteracija daleko od završnog, optimalnog rješenja, metoda zrcalnog spusta konvergira brzo kada je blizu optimalnog rješenja. Iz tog razloga često se kombiniraju u tzv. uparenu metodu [13].

Primjer 7.

Na slici 15. ispod na grafovima prikazane su iteracije metoda najbržeg spusta i zrcalnog spusta za funkciju $f(x) = \frac{x^2}{2}$ s početnim $x_0 = 2$ na intervalu $[0,2]$. Uočavamo kako je metoda najbržeg spusta brža od metode zrcalnog spusta u prvih nekoliko iteracija, a sporija nakon prvih nekoliko iteracija. [13]



Slika 15. Iteracije metode najbržeg spusta (lijevo) i metode zrcalnog spusta (desno)

3.5. Metode Newtonovog tipa

Newtonova metoda jednodimenzionalne minimizacije je jedna od najčešće korištenih metoda za određivanje lokalnog minimuma derivabilne funkcije. Pretpostavljamo da je $x^* \in [a,b]$ jedinstvena točka lokalnog minimuma funkcije $f \in C^2[a,b]$. Za početak izaberemo $x_0 \in (a,b)$ i na osnovni Taylorove formule funkciju f u okolini odabrane točke x_0 aproksimiramo kvadratnom funkcijom

$$k_1(x) = f(x_0) + f'(x_0)(x - x_0) + \tfrac{1}{2} f''(x_0)(x - x_0)^2.$$

Sljedeću aproksimaciju x_1 točke x^* biramo tako da odredimo minimum kvadratne funkcije k_1

$$x_1 = x_0 - \tfrac{f'(x_0)}{f''(x_0)}\ .$$

Ponavljajući taj postupak dobijemo niz $x_0, x_1, \dots, x_n$ koji je zadan rekurzivnom formulom

$$x_{n+1} = x_n - \frac{f'(x_n)}{f''(x_n)}, n = 0,1 \dots$$

Definirani niz postepeno konvergira prema x^*. Ova obična Newtonova metoda minimizacije formalno se podudara s Newtonovom metodom tangenti za rješavanje jednadžbe $f'(x) = 0$. [3]

Sljedeći teoremi potvrđuju prethodno navedene podatke.

Neka je $S = [a,b]$ interval realnih brojeva i $\phi: S \to S$. Točku $x \in S$ takvu da je $\phi(x) = x$ zovemo **čvrstom točkom** od ϕ.

Kontrakcijom se naziva preslikavanje $\phi: S \to S$ ako postoji pozitivni broj $q < 1$ za koji za svake dvije točke x^1 i x^2 iz S vrijedi

$$|\phi(x^1) - \phi(x^2)| \le q|x^1 - x^2|.$$

Ako je $\phi: S \to S$ kontrakcija na $S = [a, b] \subset \mathbb{R}$ i $x^0 \in S, x^{k+1} = \phi(x^k)$ za $k \in \mathbb{N}_0$ postoji jedinstvena čvrsta točka x_* preslikavanja ϕ te niz $\{x^k\}$ konvergira u x_* te

$$\left|x^{k+1} - x_*\right| \leq (q)^{k+1}|x^0 - x_*|, k = 0,1, \dots \ .$$

Neka funkcija $f \in C^2(R^n)$ zadovoljava uvjet

$$m\|y\|^2 \leq (f'(x)y, y) \leq M\|y\|^2,\ x, y, \in \mathbb{R}^n \text{ za neke } M > m > 0$$

te neka se parametri duljine koraka α_k biraju iz uvjeta

$$f(x) - f(x_k) \leq \alpha f'(x_k)p_k, \quad p_k = -\frac{f'(x_k)}{f''(x_k)} \ .$$

Slijedi da neovisno o izboru početne aproksimacije iterativni proces

$$x_{k+1} = x_k + \alpha_k p_k,\ \ k = 0,1, \dots$$

konvergira prema jedinstvenoj točki minimuma x^*. Vrijedi ispod navedeni zapis. [10]

$$\|x_{k+1} - x^*\| \leq \alpha_k \|x_k - x^*\| \text{ gdje } \alpha_k \to 0\ za\ k \to \infty$$

Primjer 8.

Način funkcioniranja opisane metode prikazan je na slici 16. ispod za primjer funkcije $f(x) = x^3 - 3x - 3$ s početnom aproksimacijom $x_0 = 2$. Pomoću druge derivacije potvrđujemo da je promatrana funkcija konveksna na intervalu $\langle 0, +\infty\rangle$. Taj podatak nam govori da minimum funkcije postoji na pozitivnom dijelu osi x.

$$f''(x) = 6x$$

$$f''(x) = 0, \quad x = 0$$

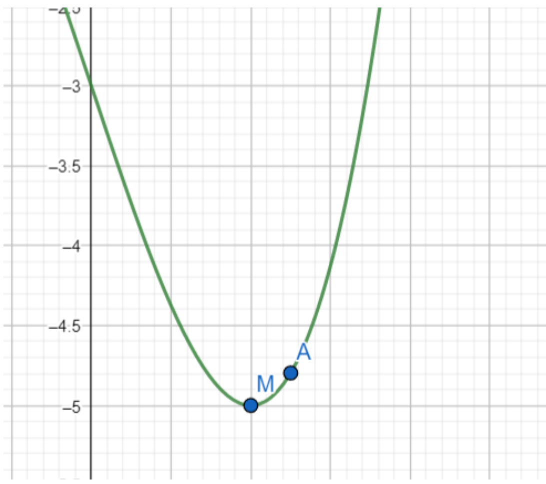
	$\langle -\infty, 0\rangle$	$\langle 0, +\infty\rangle$
$f''(x)$	$-$	$+$
$f(x)$	konkavna	konveksna

Newtonova metoda - jednodimenzionalna						
f(x)=	$x^3 - 3x - 3$				x0	Točnost
f'(x)=	$3x^2 - 3$				2	0,0005
f''(x)=	$6x$					

k	x_k	$f(x_k)$	$f'(x_k)$	Stop
1	1,25	-4,796875	1,6875	FALSE
2	1,025	-4,998109	0,151875	FALSE
3	1,00030488	-5	0,00183	FALSE
4	1,00000005	-5	2,79E-07	FALSE
5	1	-5	0	TRUE

Slika 16. Primjer korištenja Newtonove metode - jednodimenzionalna minimizacija

Na grafu funkcije $f(x) = x^3 - 3x - 3$ prikazanom na slici 17. ispod potvrđujemo da je rezultat metode ispravan, tj. da je minimum funkcije točka s koordinatama $M(1, -5)$. To dobro označava brzinu i efikasnost same metode. Točka A na grafu označava početnu točku, tj. točku iz prve iteracije metode, dok točka M označava minimum pronađen u 5. iteraciji.



Slika 17. Grafički prikaz funkcije iz primjera

Newtonova metoda višedimenzionalne minimizacije

Općeniti oblik iterativnog procesa za traženje lokalnog minimuma neprekidno diferencijabilne funkcije $f: \mathbb{R}^n \to \mathbb{R}$ je

$$x_{k+1} = x_k + \alpha_k p_k, \quad k = 0,1 \dots .$$

Kao što smo već spomenuli x_0 je početna aproksimacija, p_k vektor smjera kretanja od točke x_k u točku x_{k+1}, a $\alpha_k > 0$ duljina koraka u smjeru vektora p_k.

Cilj je ostvariti kretanje od točke x_k do x_{k+1} tako da se vrijednost funkcije svakim korakom smanjuje, tj. da bude $f(x_{k+1}) < f(x_n)$. Taj cilj osiguravamo tako da vektor p_k izaberemo da bude $\nabla f(x_k, p_k) < 0$. Već je poznato da je $\nabla f(x_k)$ gradijent funkcije f u točki x_k. Ukoliko za funkciju f napišemo Taylorovu formulu u okolini točke x_k

$$f(x) = f(x_k) + (\nabla f_k, x - x_k) + \frac{1}{2}\big(f''\big(x_k + \vartheta(x - x_k)\big)(x - x_k), x - x_k\big), \qquad \vartheta \in (0,1)$$

i stavimo $x = x_k + \alpha_k p_k$ uz oznaku $x^* = x_k + \vartheta(x - x_k)$ dobivamo

$$f(x_k + \alpha_k p_k) = f(x_k) + \alpha_k(\nabla f_k, p_k) + \tfrac{\alpha_k^2}{2}(f''(x^*)p_k, p_k), \quad \vartheta \in (0,1). \quad [3]$$

Kao u i Newtonovoj metodi jednodimenzionalne minimizacije u okolini točke x_0 funkciju aproksimiramo kvadratnom funkcijom dobivenom iz Taylorove formule

$$\psi(x) = f(x_0) + (\nabla f(x_0), x - x_0) + \frac{1}{2}(f''(x_0)(x - x_0), x - x_0).$$

Funkcija ψ je strogo konveksna i njezin minimum postiže se u točki $x_1 = x_0 - [f'(x_0)]^{-1}\nabla f(x_0)$ što je zapravo sljedeća aproksimacija do koje smo došli preko vektora smjera kretanja.

Vektor smjera kretanja definiran je s $p = -[f''(x_0)]^{-1}\nabla f(x_0)$ te definira smjer spusta iz točke x_0 jer zbog pozitivne definitnosti Hessijana f ispunjava dolje navedeni uvjet spusta

$$(\nabla f(x_0), p) = (\nabla f(x_0), -[f''(x_0)]^{-1}\nabla f(x_0) = -(f''(x_0)p, p) < 0. \qquad [3]$$

Tim postupom dolazimo do iterativne metode

$$x_{k+1} = x_k - [f''(x_k)]^{-1}\nabla f(x_k), \quad k = 0,1 \ldots.$$

koja se naziva običnom Newtonovom ili Newton-Raphsonovom metodom. Računanje inverzne matrice $[f''(x_k)]^{-1}$ izbjegava se tako da umjesto toga rješavamo sustav linearnih jednadžbi. S obzirom na to iterativni postupak je definiran kao

$$x_{k+1} = x_k + p_k, \quad (p_k \text{ rješenje sustava})$$

$$f''(x_k)p_k = -\nabla f(x_k).$$

Ako uzmemo u obzir da kretanje točke x_k u smjeru vektora p_k donosi smanjenje vrijednosti funkcije možemo uvesti duljinu koraka s čime definiramo Newtonovu metodu s regulacijom koraka

$$x_{k+1} = x_k - \alpha_k[f''(x_k)]^{-1}\nabla f(x_k), \quad k = 0,1 \ldots$$

koju u implementaciji definiramo kao

$$x_{k+1} = x_k + \alpha_k p_k,$$

$$f''(x_k)p_k = -\nabla f(x_k).$$

Jedan od načina definiranja duljine koraka je Goldstein-ov princip. [3]

Konvergencija Newtonove metode

Uz pretpostavku da je promatrana funkcija $f: \mathbb{R}^n \to \mathbb{R}$ dvostruko neprekidno diferencijabilna i jako konveksna, tj.

$$m\|y\|^2 \leq (f''(x)y, y) \leq M\|y\|^2, \qquad M > m > 0, \quad \forall x, y \in \mathbb{R}^n$$

takva funkcija je ograničena odozdo i posjeduje jedinstvenu točku minimuma.

Također vrijedi

$$\frac{m}{M^2}\|y\|^2 \leq ([f''(x)]^{-1}y, y) \leq \frac{1}{m}\|y\|^2, \quad M > m > 0, \quad , \quad \forall x, y \in \mathbb{R}^n$$

iz čega slijedi ocjena

$$(\nabla f_k, p_k) = -(\nabla f_k, [f''(x_k)]^{-1}\nabla f_k \leq -\frac{m}{M^2}\|\nabla f_k\|^2.$$

Ukoliko je funkcija $f \in C^2(\mathbb{R}^n)$ dvostruko neprekidna i jako konveksna te se parametri duljine koraka α_k biraju prema Goldstein-ovom principu, ona konvergira prema jedinstvenoj točki minimuma x^* superlinearnom brzinom

$$\|x_{k+1} - x^*\| \leq \lambda_k\|x_k - x^*\|$$

gdje $\lambda_k \to 0$ za $k \to \infty$. [3]

Quasi-Newtonove metode

Kako problem minimizacije diferencijabilne funkcije $\mathbb{R}^n \to \mathbb{R}$ možemo gledati kao problem rješavanja jednadžbe

$$F(x) = 0, \qquad F: \mathbb{R}^n \to \mathbb{R}^n$$

gdje je $F(x) - f'(x)$, a F je zapravo Jacobijeva matrica funkcije f.

Ako je

$$f(x) - \left(f_1(x), \dots, f_n(x)\right)^T, x = (x_1, \dots, x_n)^T$$

Jacobijeva matrica sastoji se od parcijalnih derivacija funkcije f_i kako je prikazano na slici 18. Općenito zaključujemo kako se sustav nelinearnih jednadžbi te problem minimizacije funkcije više varijabli rješava iterativnim metodama. [3]

$$f' = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{pmatrix}$$

Slika 18. Jacobijeva matrica parcijalnih derivacija funkcije

Primjer 9.

Na primjeru konveksne funkcije $f(x) = x_1^2 x_2^3 - x_1 x_2^3 - 1, x_1^3 - x_1 x_2^3 - 4$ s početnom točkom $x_0 = (1,1)$ provodimo Newtonovu metodu funkcije više varijabli.

Na slici ispod vidimo Jacobijevu matricu parcijalnih derivacija ove funkcije te rezultate koje dobijemo iteriranjem metode. Uočavamo da proces brzo konvergira do minimuma funkcije. [14]

$$f(x) = \begin{cases} x_1^2 x_2^3 - x_1 x_2^3 - 1 \\ x_1^3 - x_1 x_2^3 - 4 \end{cases} \qquad x_0 = (1,1)$$

$$\begin{pmatrix} 2x_1x_2^3 - x_2^3 & 3x_1^2x_2^2 - 3x_1x_2^2 \\ 3x_1^2 - x_2^3 & -3x_1x_2^2 \end{pmatrix}$$

k	$\mathbf{x}^{[k]}$	$f(\mathbf{x}^{[k]})$
1	(1.00000, 1.00000)	(–1.00000, –4.00000)
2	(2.00000, 0.33333)	(–0.92593, 3.92593)
3	(1.96192, 0.55253)	(–0.68167, 3.22078)
4	(1.75067, 1.00719)	(0.34273, –0.42320)
5	(1.74779, 0.92334)	(0.02884, –0.03679)
6	(1.74762, 0.91481)	(0.00027, –0.00035)
7	(1.74762, 0.91472)	(0.00000, 0.00000)

Slika 19. Rezultati Newtonove metode funkcije više varijabli

4. Zaključak

Optimizacija ima široku primjenu u raznim stručnim područjima (mehanika, ekonomija, operacijska istraživanja). Tu uočavamo i važnost minimizacije te njezinih metoda za rješavanje svakodnevnih problema.

Tri skupine metoda minimizacije obrađene u ovom radu su metode jednodimenzionalne minimizacije, gradijentne metode i metode Newtonovog tipa. Korisne metode jednodimenzionalne minimizacije su metoda polovljenja, metoda zlatnog reza, metoda parabole i Brentova metoda.

Što se tiče višedimenzionalne minimizacije popularne su aproksimativne gradijentne metode koje imaju cilj početnom aproksimacijom rezultata doći do sljedeće aproksimacije te tako iterativnim procesom do konačnog rezultata. Obrađene gradijentne metode su metoda najbržeg spusta (unutar nje i metoda konstantnog spusta) te metoda zrcalnog spusta.

Jedne od najčešće korištenih metoda su metode Newtonovog tipa jednodimenzionalne i višedimenzionalne minimizacije koje određuju lokalni minimum derivabilne funkcije.

U svrhu boljeg razumijevanja metoda prikazani su primjeri zadataka riješeni uz pomoć programskih alata poput Geogebre, MS Excel-a, raznih online kalkulatora te programskog jezika Python.

Zaključno, svaka od odrađenih metoda je dobra jer donosi ispravan rezultat, ali efikasnost ovisi o tipu funkcije. Ne može se odabrati najbolja metoda kad svaka bolje odgovara određenom tipu funkcije tako da bi najispravnije bilo odabrati najprikladniju metodu za baš tu funkciju. Tako pridonosimo učinkovitosti minimizacije.

Popis literature

[1] "infimum | Hrvatska enciklopedija." https://www.enciklopedija.hr/natuknica.aspx?ID=27393 (accessed Aug. 11, 2022).

[2] "minimum | Hrvatska enciklopedija." https://www.enciklopedija.hr/natuknica.aspx?id=41038 (accessed Aug. 11, 2022).

[3] R. Scitovski, N. Truhar, and Z. Tomljanović, *Metode optimizacije*. Osijek: Sveučilište Josipa Jurja Strossmayera u Osijeku - Odjel za matematiku, 2014. Accessed: Feb. 08, 2022. [Online]. Available: https://www.mathos.unios.hr/mo/Metode_optimizacije_2014.pdf

[4] P. Nujić, "Konveksni skupovi," Diplomski rad, Sveučilište Josipa Jurja Strossmayera u Osijeku - Odjel za matematiku, Osijek, 2017. Accessed: Feb. 08, 2022. [Online]. Available: https://urn.nsk.hr/urn:nbn:hr:126:403773

[5] B. Guljaš predavanja Zagreb, "OSNOVE MATEMATIČKEMATEMATIˇMATEMATIČKE ANALIZE".

[6] "Norm (mathematics) - Wikipedia." https://en.wikipedia.org/wiki/Norm_(mathematics) (accessed Aug. 11, 2022).

[7] E. Đaferović, "Hijerarhija konveksnih funkcija," Diplomski rad, Sveučilište u Zagrebu, Zagreb, 2018. Accessed: Feb. 08, 2022. [Online]. Available: https://repozitorij.pmf.unizg.hr/islandora/object/pmf:5123

[8] "Konveksnost i konkavnost | Sustav." https://sustav.sino.com.hr/konveksnost-i-konkavnost (accessed Aug. 29, 2022).

[9] M. Čančarević, Š. Zlopaša, and D. Čulina, "Optimizacija pomoću gradijentne metode".

[10] Slišković M, "Konveksnost i optimizacija," Diplomski rad, Sveučilište u Zagrebu, Zagreb, 2017. Accessed: Feb. 17, 2022. [Online]. Available: https://urn.nsk.hr/urn:nbn:hr:217:406951

[11] "Optimization: Minimization by Parabolic Interpolation MATH2070: Numerical Methods in Scientific Computing I How do we find the lowest point?".

[12] "Brent's Minimization Method - YouTube." https://www.youtube.com/watch?v=BQm7uTYC0sg (accessed Aug. 17, 2022).

[13] L. Borozan, S. Jelić, D. Matijević, and D. J. Ševerdija Sveučilište Strossmayera Osijeku, "math.e Uparena optimizacijska metoda gradijentni i zrcalni spust hibridna ili uparena metoda konveksna optimizacija".

[14] "Multivariate Newton's Method - Value-at-Risk: Theory and Practice." https://www.value-at-risk.net/multivariate-newtons-method/#exercise_2_21 (accessed Jul. 14, 2022).

Popis slika

Popis tablica

Prilog 1

U nastavku prilažem Python programski kod za metodu zlatnog reza koji je opisan kroz rad.

Metoda zlatnog reza.py

Metoda zlatnog reza.py - C:\Users\Korisnik\Desktop\Metoda zlatnog reza.py (3.9.5)

File Edit Format Run Options Window Help

```python
#Metoda zlatnog reza

#Unos koeficijenata odabrane kvadratne funkcije
import math
print("Unos kvadratne funkcije")
fjaa=float(input("Unesite vodeći koeficijent kvadratne funkcije a: "))
if fjaa <= 0:
    print("Unesena funkcija nije konveksna, nema minimum")
    fjaa=float(input("Unesite vodeći koeficijent kvadratne funkcije a: "))
fjab=float(input("Unesite linearni koeficijent kvadratne funkcije b: "))
fjac=float(input("Unesite slobodni koeficijent kvadratne funkcije c: "))

#Unos intervala i željene točnosti za provođenje metode
print("Unos intervala")
a=round(float(input("Unesi a ")),5)
b=round(float(input("Unesi b ")),5)
print("Željena točnost")
e=float(input("Unesite točnost "))

#Računanje x1 i x2 prema formulama za omjer zlatnog reza
x1=round(a+((3-math.sqrt(5))/2)*(b-a),5)
x2=round(a+((math.sqrt(5)-1)/2)*(b-a),5)
i=0

#Ispis zaglavlja tablice za iteracije metode
print("--------------------------------------------------------------------------")
print("Broj iteracije",'  ',"{:<10}{:<10}{:<10}{:<10}{:<10}{:<10}".format('a','b','x1','x2','fx1','fx2'))
print("--------------------------------------------------------------------------")

#Provođenje metode (ispisuje podatke koje dobije u svakoj iteraciji)
while (x2-x1) >= e:
    i+=1
    fx1=round((x1**2)*fjaa+x1*fjab+fjac,5)
    fx2=round((x2**2)*fjaa+x2*fjab+fjac,5)
    print('     ',"{:<10}{:<10}{:<10}{:<10}{:<10}{:<10}{:<10}".format(i,a,b,x1,x2,fx1,fx2))
    if fx1 <= fx2:
        b=round(x2,5)
        x2=round(x1,5)
        x1=round(a+b-x2,5)
    else:
        a=round(x1,5)
        x1=round(x2,5)
        x2=round(a+b-x1,5)

#Konačni rezultat
x=round((x1+x2)/2,5)
y=round((fx1+fx2)/2,5)
print("--------------------------------------------------------------------------")
print()
print("KONAČNI REZULTAT")
print("MIN(",x,',',y,')')
print()
print("Minimum zadane funkcije f(x)=",'(',fjaa,')',"x^2 +",'(',fjab,')',"x +",'(',fjac,')'," ima koordinate ",x," i ",y,".")
```