BFGS

Nikola Ajzenhamer

Matematički fakultet,
Univerzitet u Beogradu

4. juli 2018.

Sadržaj

Uvod

- Broyden-Fletcher-Goldfarb-Shanno
- Jedan od Kvazi-Njutnovih metoda:
  - Zahtevaju gradijent ciljne funkcije pri svakoj iteraciji
  - Konstruišu model ciljne funkcije
    koji je dovoljno dobar da daje superlinearnu konvergenciju
  - S obzirom da ne zahtevaju izvode drugog reda,
    ponekad su brži od Njutnovih metoda

BFGS metod

- Model ciljne funkcije:

$$m_k(p) = f_k + \nabla f_k^T p + \frac{1}{2} p^T B_k p \qquad (1)$$

- B_k je:
  - Simetrična pozitivno definitna matrica dimenzija $n \times n$
  - Ažurira se pri svakoj iteraciji
  - Predstavlja aproksimaciju Hesijana

BFGS metod (2)

- Minimizator modela 1:

$$p_k = -B_k^{-1} \nabla f_k \tag{2}$$

- Koristi se kao pravac pretrage:

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

- Kasnije ćemo razmotriti određivanje koraka α_k

BFGS metod (3)

- Može se izvesti naredna formula
  koja se koristi za izračunavanje naredne aproksimacije B_{k+1}^{-1}
  ukoliko je poznata prethodna aproksimacija B_k^{-1}:

$$B_{k+1}^{-1} = (I - \rho_k s_k y_k^T) B_k^{-1} (I - \rho_k y_k s_k^T) + \rho_k s_k s_k^T \tag{3}$$

  gde važi

$$s_k = x_{k+1} - x_k, \quad y_k = \nabla f_{k+1} - \nabla f_k, \quad \rho_k = \frac{1}{y_k^T s_k}$$

- Za početnu aproksimaciju može se uzeti $B_0^{-1} = I$

Određivanje koraka α_k

- Određivanje α_k se vrši tako da zadovoljava Wolfe uslove:

$$\begin{aligned} f(x_k + \alpha_k p_k) &\leq f(x_k) + c_1 \alpha_k \nabla f_k^T p_k \\ \nabla f(x_k + \alpha_k p_k)^T p_k &\geq c_2 \nabla f_k^T p_k \end{aligned} \tag{4}$$

  gde važi $0 < c_1 < c_2 < 1$.
- Prvi uslov govori da α_k treba da daje *neophodan spust* ciljne funkcije f (*Armijo condition*)
- Drugi uslov eliminiše premale korake (*curvature condition*)

Određivanje koraka α_k

- Algoritam linijske pretrage za određivanje koraka
  vrši se u dve faze:
  1. Faza odsecanja – pronalazi se interval koji sadrži prihvatljive vrednosti
  2. Faza selekcije – pronalazi se finalni korak (algoritam `Zoom`)

Algoritam `Linijska_pretraga`

- Ulaz:
  - Funkcija $\phi(\alpha) = f(x_k + \alpha p_k)$
- Izlaz:
  - Parametar α^* koji zadovoljava Wolfe uslove 4

Algoritam `Linijska_pretraga`

```
Postavi α_0 ← 0, odaberi α_1 > 0 i α_max;
i ← 1;
repeat
    Izračunaj φ(α_i);
    if φ(α_i) > φ(0) + c_1 α_i φ'(α_0) or [φ(α_i) ≥ φ(α_{i-1}) and i > 1]
        Postavi α* ← Zoom(α_{i-1}, α_i); Zaustavi;
    Izračunaj φ'(α_i);
    if |φ'(α_i)| ≤ -c_2 φ'(0)
        Postavi α* ← α_i; Zaustavi;
    if φ'(α_i) ≥ 0
        Postavi α* ← Zoom(α_i, α_{i-1}); Zaustavi;
    Izaberi α_{i+1} ∈ (α_i, α_max);
    i ← i + 1;
end (repeat)
```

Algoritam `Zoom`

- Ulaz:
  - Funkcija $\phi(\alpha) = f(x_k + \alpha p_k)$
  - Vrednosti α_{lo} i α_{hi}
- Izlaz:
  - Finalni korak α^* iz intervala $[\alpha_{lo}, \alpha_{hi}]$

Algoritam Zoom

repeat
    Pronađi probni korak α_j između α_{lo} i α_{hi};
    Izračunaj $\phi(\alpha_j)$;
    if $\phi(\alpha_j) > \phi(0) + c_1\alpha_j\phi'(\alpha_0)$ or $\phi(\alpha_j) \geq \phi(\alpha_{lo})$
        Postavi $\alpha_{hi} \leftarrow \alpha_j$;
    else
        Izračunaj $\phi'(\alpha_j)$;
        if $|\phi'(\alpha_j)| \leq -c_2\phi'(0)$
            Postavi $\alpha^* \leftarrow \alpha_j$; Zaustavi;
        if $\phi'(\alpha_j)(\alpha_{hi} - \alpha_{lo}) \geq 0$
            Postavi $\alpha_{hi} \leftarrow \alpha_{lo}$;
        Postavi $\alpha_{lo} \leftarrow \alpha_j$;
end (repeat)

Algoritam BFGS

- Ulaz:
  - Ciljna funkcija f
  - Gradijent funkcije ∇f
  - Početna tačka x_0
  - Parametar konvergencije ϵ
  - Inverz aproksimacije Hesijana B_0^{-1}
- Izlaz:
  - Aproksimacija minimuma x^*

Algoritam BFGS

$k \leftarrow 0$;
`while` $||\nabla f_k|| > \epsilon$
  `Izračunaj pravac pretrage` p_k `prema formuli 2;`
  `Postavi` $\alpha_k \leftarrow$ `Linijska_pretraga(`f, x_k, p_k`);`
  `Postavi` $x_{k+1} \leftarrow x_k + \alpha_k p_k$;
  `Postavi` $s_k \leftarrow x_{k+1} - x_k$;
  `Postavi` $y_k \leftarrow \nabla f_{k+1} - \nabla f_k$;
  `Izračunaj` B_{k+1}^{-1} `prema formuli 3;`
  $k \leftarrow k + 1$;
`end (repeat)`

Ocena složenosti

- S obzirom da nema rešavanja sistema linearnih jednačina i matrica-matrica operacija, ne postoje ni operacije složenosti $O(n^3)$
- Formula 3 može se prezapisati tako da se izvrše samo množenja $(n \times n) \cdot (n \times 1)$, $(n \times 1) \cdot (1 \times n)$ i $(1 \times n) \cdot (n \times n)$, čije su složenosti $O(n^2)$
- Dakle, svaka iteracija se izvodi po cenu od $O(n^2)$ uz cenu izračunavanja vrednosti funkcija i gradijenata

Primeri izvršavanja algoritma BFGS

Svi primeri sadrže neke zajedničke parametre:

- Parametar konvergencije: $\epsilon = 0.001$
- Inverz aproksimacije Hesijana $B_0^{-1} = I$

Dodatno:

- Broj koraka u algoritmu je ograničen na 30
- Koordinate početnih tačaka su nasumično odabrane iz intervala $[-100, 100]$

Primeri izvršavanja algoritma BFGS – Primer 1

Ulaz:

- Funkcija $f = x^2 + (y-5)^2 + z^2 + \sin^2(x)$
- Početna tačka: $x_0 = (-80, 2, 21)$

Osobine funkcije:

- Nema globalni minimum
- Lokalni minimum: $x_{lmin} = (0, 5, 0)$

Izlaz:

- Aproksimacija minimuma: $x^* = (2.26262201 \cdot 10^{-9}, 5.00000000 \cdot 10^{0}, 3.08331127 \cdot 10^{-8})$
- Broj koraka: 6

Primeri izvršavanja algoritma BFGS – Primer 2

Ulaz:

- Funkcija $g = -(5 + 3x - 4y - x^2 + xy - y^2)$
- Početna tačka: $x_0 = (-26, -13)$

Osobine funkcije:

- Globalni minimum: $x_{gmin} = (\frac{2}{3}, -\frac{5}{3})$

Izlaz:

- Aproksimacija minimuma: $x^* = (0.66666667, -1.66666667)$
- Broj koraka: 5

Primeri izvršavanja algoritma BFGS – Primer 3

Ulaz:

- Funkcija $h = (4 - x^2 - 2y^2)$
- Početna tačka: $x_0 = (16, -1)$

Osobine funkcije:

- Globalni minimum: Svaka tačka x_{gmin} koja zadovoljava $x^2 + 2y^2 = 4$

Izlaz:

- Aproksimacija minimuma: $x^* = (0.17758343, 1.40862772)$[1]
- Broj koraka: 7

[1] Uvrštanjem vrednosti x^* u izraz $x^2 + 2y^2$ dobija se vrednost 3.999999981715361.

Hvala na pažnji!

Pitanja?

Dodatak

Literatura

- Jorge Nocedal & S. Wright. *Numerical Optimization*.
  Springer-Verlag New York, 2006.

Repozitorijum

https://github.com/theikeofficial/BFGS