Jakobijev algoritam za pronalaženje sopstvenih vrednosti

Isidora Đurđević

Matematički fakultet,
Univerzitet u Beogradu

13. septembar 2019.

Sadržaj

Uvod

- Jakobijev algoritam je iterativna metoda za pronalaženje sopstvenih vrednosti i sopstvenih vektora matrice
- Naziv je dobio po matematičaru Gustav Karl Jakobiju
- Poznat kao proces dijagonalizacije

Jakobijev algoritam

- Rešenje je garantovano za realne i simetrične matrice
- Zasniva se na iterativnoj primeni matrica rotacija kako bi se anulirali elementi van dijagonale
- Rotacije se zovu Jakobijeve ili Givensove rotacije
- Rezultat je dijagonalna matrica na čijoj dijagonali se nalaze sopstvene vrednosti
- Proizvodom primenjenih matrica rotacija dobija se matrica čije kolone čine sopstvene vektore

Jakobijev algoritam (2)

- Neka je `S` simetrična matrica, i `G = G(i, j, `θ`)` Givensova matrica rotacije. Tada je:

$$\texttt{S'} = \texttt{GS}G^T$$

- simetrična matrica i slična je matrici `S`
- matrica `S'` se izračunava sledećim formulama:

$$\begin{array}{rcl}
S'_{ii} & = & c^2 S_{ii} - 2scS_{ij} + s^2 S_{jj} \\
S'_{jj} & = & s^2 S_{ii} - 2scS_{ij} + c^2 S_{jj} \\
S'_{ij} = S'_{ji} & = & (c^2 - s^2)S_{ij} + sc(S_{ii} - S_{jj}) \\
S'_{ik} = S'_{ki} & = & cS_{ik} - sS_{jk} \qquad \mathrm{k} \neq \mathrm{i, j} \\
S'_{jk} = S'_{kj} & = & sS_{ik} - cS_{jk} \qquad \mathrm{k} \neq \mathrm{i, j} \\
S'_{kl} & = & S_{kl} \qquad \mathrm{k, l} \neq \mathrm{i, j}
\end{array}$$

Jakobijev algoritam (3)

- Gde je `s = sin(`θ`)` i `c = cos(`θ`)`
- Pošto je `G` ortogonalna matrica, `S` i `S’` imaju istu normu
- Ako izaberemo θ tako da je S'_{ij} `= 0` tada je `S’`:

$$S'_{ij} = cos(2\theta)S_{ij} + \frac{1}{2}sin(2\theta)(S_{ii} - S_{jj}) \tag{1}$$

Jakobijev algoritam (4)

- Izjednačavanjem prethodne formule na 0 i sređivanjem dobijamo:

$$tan(2\theta) = \frac{2S_{ij}}{S_{jj} - S_{ii}} \tag{2}$$

- S_{ij} je pivot i predstavlja najveći vandijagonalni element
- Matrica rotacije:

$$\begin{bmatrix} cos\theta & sin\theta \\ -sin\theta & cos\theta \end{bmatrix}$$

Jakobijev algoritam - koraci

- Pronađi najveći vandijagonalni element
- Ukoliko je najveći vandijagonalni element manji od unapred definisane vrednosti ε - `break` -
- Inače izvrši rotaciju tako da se anulira najveći vandijagonalni element

Ocena složenosti

- Složenost rotacije je `O(n)` ukoliko je poznat najveći vandijagonalni element, inače `O(`n^2`)`
- Moguće je svesti na `O(n)` ukoliko se čuva niz sa vrednostima vandijagonalnih elemenata, gde je `i`-ti element niza najveći element u `i`-tom redu matrice
- Rotacija predstavlja množenje matrica i složenost je `O(`n^3`)`

Hvala na pažnji!

Pitanja?

Dodatak

Literatura

Repozitorijum

https://github.com/theikeofficial/BFGS