

THUẬT TOÁN

Phương pháp Gauss Seidel giải hệ phương trình $Ax = b$

Input: Ma trận mở rộng $[A|B]$, sai số ϵ

Output: Nghiệm x

Bước 1: Nhập dữ liệu đầu vào

Bước 2: Kiểm tra tính chéo trội của ma trận A

- + Dùng gói `row_diagonal()` để kiểm tra tính chéo trội hàng
- + Nếu thỏa mãn thì tiếp tục thuật toán ở bước 3
- + Nếu không thỏa mãn thì dùng gói `col_diagonal()` kiểm tra tính chéo trội cột
 - Nếu ma trận A chéo trội cột thì tiếp tục thuật toán ở bước 3
 - Nếu không thỏa mãn thì kết luận ma trận A không chéo trội và dừng thuật toán

Bước 3: Biến đổi $Ax = b$ về dạng $x = Cx + d$

- + Xác định ma trận C
 - Dùng gói `C_row_matrix()` nếu A chéo trội hàng
 - Dùng gói `C_col_matrix()` nếu A chéo trội cột
- + Xác định vecto d
 - Nếu chéo trội hàng thì
for $i = 0$ to n : $d[i] = b[i] / a[i][i]$
 - Nếu chéo trội cột thì
for $i = 0$ to n : $d[i] = b[i]$

Bước 4: Xác định hệ số co

- + Nếu chéo trội hàng thì dùng gói `lamda_cal()` để tìm hệ số co λ
- + Nếu chéo trội cột thì dùng gói `zeta_cal()` để tìm hệ số co ζ
- + Trong trường hợp chéo trội cột thì cần dùng gói `s_cal()` để tìm thêm hệ số s

Bước 5: Tìm nghiệm:

- + Dùng gói `row_solve()` để tìm nghiệm khi chéo trội hàng
- + Dùng gói `col_solve()` để tìm nghiệm khi chéo trội cột

Bước 6: In nghiệm ra ngoài màn hình

Gói kiểm tra tính chéo trội hàng *row_diagonal()*

Input: Ma trận A

Output: *row_diagonal()* = 1 (nếu true) hay 0 (nếu false)

Bước 1: Tính:

$$temp = \sum_{j=0}^n |a[i][j]|$$

Bước 2: Kiểm tra điều kiện: for i = 0 to n-1:

Nếu $2 * |a_{ii}| > temp$ thì return true

ngược lại return false

Gói kiểm tra tính chéo trội cột *col_diagonal()*

Input: Ma trận A

Output: *col_diagonal()* = 1 (nếu true) hay 0 (nếu false)

Bước 1: Tính:

$$temp = \sum_{i=0}^n |a[i][j]|$$

Bước 2: Kiểm tra điều kiện: for j = 0 to n-1:

Nếu $2 * |a[j][j]| > temp$ thì return true

ngược lại return false

Gói xác định ma trận C khi chéo trội hàng *C_row_matrix()*

Input: Ma trận A

Output: Ma trận C

Bước : Dùng vòng lặp for để tìm $c[i][j]$

for i = 0 to n-1:

for j = 0 to n-1:

Nếu $i == j$ thì $c[i][j] = 0$

Ngược lại thì $c[i][j] = -a[i][j] / a[i][i]$

Gói xác định ma trận C khi chéo trội cột C_col_matrix()

Input: Ma trận A

Output: Ma trận C

Bước : Dùng vòng lặp for để tìm $c[i][j]$

for $i = 0$ to $n-1$:

for $j = 0$ to $n-1$:

Nếu $i == j$ thì $c[i][j] = 0$

Ngược lại thì $c[i][j] = -a[i][j] / a[j][j]$

Gói tính hệ số co lamda khi chéo trội hàng lamda_cal()

Input: Ma trận C

Output: lamda

Bước 1: for $i=0$ to $n-1$: tính

$$beta = \sum_{j=0}^i |c[i][j]|$$

$$gamma = \sum_{j=i}^n |c[i][j]|$$

Bước 2: Tìm lamda: Với $i = 0$ to $n-1$:

$$lamda = \max \frac{gamma}{1 - beta}$$

Gói tính hệ số co zeta khi chéo trội cột zeta_cal()

Input: Ma trận C

Output: zeta

Bước 1: for $j=0$ to $n-1$: tính

$$beta = \sum_{i=0}^j |c[i][j]|$$

$$gamma = \sum_{i=j}^n |c[i][j]|$$

Bước 2: Tìm lamda: Với $j = 0$ to $n-1$:

$$zeta = \max \frac{beta}{1 - gamma}$$

Gói tính hệ số s khi chéo trội cột

Input: ma trận C

Output: s

Bước 1: for $j = 0$ to $n-1$:

$$temp = \sum_{i=0}^j |c[i][j]|$$

Bước 2: Với $j = 0$ to $n-1$: $s = \max temp$

Gói tìm nghiệm khi chéo trội hàng row_solve()

Input: ma trận A, C, lamda, epsi

Output: x

Bước 1: Khởi tạo vector $x[i] = 0$ với $i = 0$ to $n-1$

Bước 2: for $i = 0$ to $n-1$: đặt $z[i] = x[i]$

Bước 3: Tính $x[i]$ với $i=0$ to $n-1$ và j khác i

$$x[i] = b[i] / a[i][i] + \sum_{j=0}^{n-1} c[i][j] * x[j]$$

Bước 4: Kiểm tra điều kiện dừng

Nếu $\|z - x\| < epsi * (1 - lamda) / lamda$ thì đưa nghiệm $x[i]$ ra màn hình

Ngược lại thì quay lại bước 2

Gói tìm nghiệm khi chéo trội cột col_solve()

Input: ma trận A, C, zeta, s, epsi

Output: x

Bước 1: Khởi tạo vector $x[i] = 0$ với $i = 0$ to $n-1$

Bước 2: for $i = 0$ to $n-1$: đặt $y[i] = x[i] * a[i][i]$

Bước 3: for $i = 0$ to $n-1$: đặt $z[i] = y[i]$

Bước 4: Tính $y[i]$ với $i=0$ to $n-1$ và j khác i

$$y[i] = b[i] + \sum_{j=0}^{n-1} c[i][j] * y[j]$$

Bước 5: Tính $x[i] = y[i]/a[i][i]$ với $i=0$ to $n-1$

Bước 6: Kiểm tra điều kiện dừng

Nếu $\| z - y \| < \textit{epsi} * (1 - s) * (1 - \textit{zeta}) / \textit{zeta}$ thì kết luận nghiệm là $x[i]$

Ngược lại thì quay lại bước 3