20/09

Хранение разреженных матриц

Опр. Портретом разреженной матрицы называется множество:

$$P_A = \{(i, j) : a_{ij} \neq 0\}$$

Возможны следующие случаи:

1. Матрица несимметрична и ее портрет несимметричен:

$$\exists\ (i, j) \in P_A : (j, i) \notin P_A \Leftrightarrow A \neq A^{\mathrm{T}}$$

$$\begin{bmatrix} 1 & 2 \\ 0 & 0 \end{bmatrix} \Rightarrow \begin{bmatrix} * & * \\ 0 & 0 \end{bmatrix}$$

2. Матрица несимметрична но её портрет симметричен:

$$A \neq A^{\mathrm{T}},\ \text{но}\ (i, j) \in P_A, (j, i) \in P_A, a_{ij} \neq a_{ij}$$

$$\begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix} \Rightarrow \begin{bmatrix} * & 0 \\ 0 & * \end{bmatrix}$$

3. Если матрица симметрична ($a_{ji} = a_{ij}$), то её портрет симметричен:

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \Rightarrow \begin{bmatrix} * & 0 \\ 0 & * \end{bmatrix}$$

Форматы хранения разреженных матриц:
CSR - compressed sparse row
CSC - compressed sparse column
CSLR - compressed sparse low-triangle row

Рассмотрим формат CSR:

1. aelem - массив, хранящий все $a_{ij} \neq 0$ в строках

2. jptr - массив размерности aelem, указывает номер N_j элемента a_{ij}

3. iptr - массив размерности $n+1$ (n - размерность СЛАУ), хранит число элементов $a_{ij} \neq 0$ в строке

   iptr$[i+1]$ - iptr$[i]$ - число элементов в i-ой строке

   iptr$[n+1]$ - число элементов в aelem $+\ 1$

   Пример:

$$A = \begin{bmatrix} 9 & 0 & 0 & 3 & 1 & 0 & 1 \\ 0 & 11 & 2 & 1 & 0 & 0 & 2 \\ 0 & 1 & 10 & 2 & 0 & 0 & 0 \\ 2 & 1 & 2 & 9 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 12 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 8 & 0 \\ 2 & 2 & 0 & 0 & 3 & 0 & 8 \end{bmatrix}$$

Матрица A несимметрична, а P_A симметричен, т.к. если $a_{ij} \neq 0$, то $a_{ji} \neq 0$.

aelem: $[[9,3,1,1], [11,2,1,2], [1,10,2], [2,1,2,9,1], [1,1,12,1], [8], [2,2,3,8]]$.

jptr: $[[1,4,5,7], [2,3,4,7], [2,3,4], [1,2,3,4,5], [1,4,5,7], [6], [1,2,5,7]]$.

iptr: $[1,5,9,12,17,21,22,26]$.

Рассмотрим формат CSLR:

Обычно такой формат используется для хранения симметричных матриц, т.е. в памяти хранятся только элементы верхнего (или нижнего) треугольника. Если матрица несимметрична, то храним ещё и элементы нижнего треугольника, но уже по столбцам (для сохранения структуры).

В нашем примере:

adiag: $[9,11,10,9,12,8,8]$ – все диагональные элементы.

altr: $[1,2,1,2,1,1,2,2,3]$ – элементы нижнего треугольника.

autr: $[2,3,1,2,1,1,1,2,1]$ – элементы верхнего треугольника.

jptr: $[2,1,2,3,1,4,1,2,5]$.

iptr: $[1,1,1,2,5,7,7,10]$.

Входные данные: $\overline{x}$, aelem, jptr, iptr, n.

$$z = A\overline{x},\ A - CSR$$

Листинг 1 - алгоритм составления вектора z

```
i: 1,.., n
{
    z[i]=0;
    j: iptr[i],..,iptr[i+1]-1
    {
        z[i]=z[i]+x[jptr[j]]*aelem[j];
    }
}
```

Учет граничных условий

Граничные условия I рода

Зададим температуру в узле a_{33}:

$$A = \begin{bmatrix} \cdots & & \cdots & & \cdots \\ \cdots & 0 & 1 & 0 & \cdots \\ \cdots & & \cdots & & \cdots \end{bmatrix}$$

Чтобы не испортилась симметричность портрета, искусственно обнулим необходимые элементы массива aelem.

Листинг 2 - алгоритм задания граничных условий

```
i = iptr[k],..,iptr[k+1]-1
{
    if jptr[i]=k:
        aelem[jptr[k]]=1;
    else:
        aelem[jptr[k]]=0;
}
```

Метод Холецкого

$$A \approx LU$$
$$A = LU + R$$

где L - нижнетреугольная матрица;

U - верхнетреугольная матрица;

R - ошибка округления.

$$\begin{cases} Ax = b \\ LUx = b \\ Ly = b \Rightarrow Ux = y \end{cases}$$

Предобусловливание

Число обусловленности: $\frac{\lambda_{max}}{\lambda_{min}}$

Если число обусловленности велико, система плохо обусловлена, и небольшие ошибки в данных могут привести к большим ошибкам в решении. Если число обусловленности близко к 1, система хорошо обусловлена, и решения будут стабильными.

Рассмотрим СЛАУ $Ax = b$, где A – плохо обусловленная матрица.

Пусть M – невырожденная матрица размерности $n \times n$. Домножим СЛАУ на матрицу, обратную M:

$$M^{-1}Ax = M^{-1}b$$

$$A^{*}x = b^{*}$$

M - матрица предобусловливания

1. M должна быть по возможности близка к A (пример: M = diag(A)).
2. M должна быть легко вычислима.
3. M должна быть легко обратима.