第一题

1. 对于 $r=1$,

$$
\begin{aligned}
u_{11} &= a_{11} &= 2\\
u_{12} &= a_{12} &= 3\\
u_{13} &= a_{13} &= 4\\
l_{21} &= \frac{a_{21}}{l_{11}} &= \frac{3}{2}\\
l_{31} &= \frac{a_{31}}{l_{11}} &= 2
\end{aligned}
$$

2. 对于 $r=2$,

$$
\begin{aligned}
u_{22} &= a_{22} - l_{21}u_{12} &= 5 - \frac{3}{2} \times 3 = \frac{1}{2}\\
u_{23} &= a_{23} - l_{21}u_{13} &= 2 - \frac{3}{2} \times 4 = -4\\
l_{32} &= \frac{a_{32} - l_{31}u_{12}}{u_{22}} &= \frac{3 - 2 \times 3}{\frac{1}{2}} = -6
\end{aligned}
$$

3. 对于 $r=3$,

$$
u_{33} = a_{33} - l_{31}u_{13} - l_{32}u_{23} = 30 - 2 \times 4 - (-6) \times (-4) = -2
$$

于是

$$
A = \begin{pmatrix} 1 & & \\ \frac{3}{2} & 1 & \\ 2 & -6 & 1 \end{pmatrix} \begin{pmatrix} 2 & 3 & 4 \\ & \frac{1}{2} & -4 \\ & & -2 \end{pmatrix} = LU
$$

4. 求解

由 $Ly=b$ 得到

$$
\begin{cases}
y_1 = b_1 & = 6 \\
y_2 = b_2 - l_{21}y_1 & = 5 - \frac{3}{2} \times 6 & = -4 \\
y_3 = b_3 - l_{31}y_1 - l_{32}y_2 & = 32 - 2 \times 6 - (-6) \times (-4) & = -4
\end{cases}
$$

从而 $y = (6, -4, -4)^T$

由 $Ux=y$ 得到

$$
\begin{cases}
x_3 = \frac{y_3}{u_{33}} & = \frac{-4}{-2} & = 2 \\
x_2 = \frac{y_2 - u_{23}x_2}{u_{22}} & = \frac{-4-(-4)\times 2}{\frac{1}{2}} & = 8 \\
x_1 = \frac{y_1 - u_{12}x_2 - u_{13}x_3}{u_{11}} & = \frac{6-3\times 8-4\times 2}{2} & = -13
\end{cases}
$$

从而 $x = (-13, 8, 2)^T$

教材第一章课后习题

4

因为 $3 \neq 7$，所以确定 Gauss 变换 L_1 形如

$$L_1 = \begin{pmatrix} 1 & & \\ -l_{21} & 1 & \\ -l_{31} & & 1 \end{pmatrix}$$

容易解得

$$l_{21} = -\frac{7-3}{2} = 2, l_{31} = -\frac{8-4}{2} = 2$$

从而

$$L_1 = \begin{pmatrix} 1 & & \\ 2 & 1 & \\ 2 & & 1 \end{pmatrix}$$

7

A 可以记为

$$A = \begin{pmatrix} a_{11} & a_1^T \\ a_1 & A_1 \end{pmatrix}$$

可得 Gauss 变换 L_1 为

$$L_1 = \begin{pmatrix} 1 & & & & \\ -\frac{a_{21}}{a_{11}} & 1 & & & \\ -\frac{a_{31}}{a_{11}} & & 1 & & \\ \vdots & & & \ddots & \\ -\frac{a_{n1}}{a_{11}} & & & & 1 \end{pmatrix} = \begin{pmatrix} 1 & & & & \\ -\frac{a_{12}}{a_{11}} & 1 & & & \\ -\frac{a_{13}}{a_{11}} & & 1 & & \\ \vdots & & & \ddots & \\ -\frac{a_{1n}}{a_{11}} & & & & 1 \end{pmatrix}$$

从而

$$A_2 = A_1 - \frac{a_1 a_1^T}{a_{11}}$$

显然对称

8

根据高斯变换，我们可以知道

$$(A_2)_{ij} = a_{ij} - \frac{a_{i1}}{a_{11}} a_{1j}$$

A_2 非对角线元素的绝对值之和为

$$\sum_{j=2, j\neq k}^{n} |(A_2)_{kj}| = \sum_{j=2, j\neq k}^{n} |a_{kj} - \frac{a_{k1}}{a_{11}} a_{1j}|$$

根据绝对值三角不等式，可得

$$\begin{aligned}\sum_{j=2, j\neq k}^{n} |a_{kj} - \frac{a_{k1}}{a_{11}} a_{1j}| &\leq \sum_{j=2, j\neq k}^{n} \left(|a_{kj}| + |\frac{a_{k1}}{a_{11}} a_{1j}| \right) \\ &= \sum_{j=2, j\neq k}^{n} |a_{kj}| + |\frac{a_{k1}}{a_{11}}| \sum_{j=2, j\neq k}^{n} |a_{1j}|\end{aligned}$$

由于 A 是严格对角占优阵，所以

$$|a_{kk}| > \sum_{j=1, j\neq k}^{n} |a_{kj}|$$

进一步可得

$$|a_{11}| - |a_{1k}| > \sum_{j=2, j\neq k}^{n} |a_{1j}|$$

也就是说

$$|\frac{a_{k1}}{a_{11}}| \sum_{j=2, j\neq k}^{n} |a_{1j}| < |a_{k1}| - |\frac{a_{1k} a_{k1}}{a_{11}}|$$

最终可得

$$\sum_{j=2, j\neq k}^{n} |a_{kj} - \frac{a_{k1}}{a_{11}} a_{1j}| < \sum_{j=2, j\neq k}^{n} |a_{kj}| + |a_{k1}| - |\frac{a_{1k} a_{k1}}{a_{11}}|$$

回到 A_2，可以发现

$$\begin{aligned}|(A_2)_{kk}| &= |a_{kk} - \frac{a_{k1}}{a_{11}} a_{1k}| \\ &\geq |a_{kk}| - |\frac{a_{1k} a_{k1}}{a_{11}}| \\ &> \sum_{j=2, j\neq k}^{n} |a_{kj}| + |a_{k1}| - |\frac{a_{1k} a_{k1}}{a_{11}}| > \sum_{j=2, j\neq k}^{n} |(A_2)_{kj}|\end{aligned}$$

也就是说，A_2 是严格对角占优阵。

10

设原正定矩阵的分块形式为

$$A = \begin{pmatrix} a_{11} & a_1^T \\ a_1 & A_1 \end{pmatrix}, a_{11} > 0$$

经过一步 Gauss 消去后，可得

$$A_2 = A_1 - \frac{a_1 a_1^T}{a_{11}}$$

令 H_k 表示A_2 的前 k 行前 k 列子矩阵，则

$$H_k = G_k - \frac{h_k h_k^T}{a_{11}}$$

其中 G_k 是 A_1 的前 k 行前 k 列子矩阵，h_k 是 a_1 的前 k 个元素组成的向量

令 M_{k+1} 表示 A 的前 $k+1$ 行前 $k+1$ 列子矩阵

矩阵 M_{k+1} 用分块形式可以表示为

$$M_{k+1} = \begin{pmatrix} a_{11} & h_k^T \\ h_k & G_k \end{pmatrix}$$

利用分块矩阵的行列式求法，可得

$$\det(M_{k+1}) = a_{11} \det\left(G_k - \frac{1}{a_{11}} h_k h_k^T\right) = a_{11} \det(H_k)$$

因此

$$\det(H_k) = \frac{\det(M_{k+1})}{a_{11}} > 0$$

综上，A_2 是正定矩阵。

教材第一章上机习题 1

```cpp
#include <format>
#include <iostream>

using namespace std;

void solve(double** A, double* b, const int n, double** L, double** U,
```

```
            double* y, double* x) {
  double** tmp = new double*[n + 1];
  for (int i = 0; i <= n; i++) tmp[i] = new double[n + 1];
  for (int i = 0; i <= n; i++)
    for (int j = 0; j <= n; j++) tmp[i][j] = A[i][j];

  for (size_t k = 1; k <= n - 1; ++k) {
    size_t p = k;
    for (size_t i = k; i <= n; ++i)
      if (A[i][k] > A[p][k]) p = i;
    for (size_t j = 1; j <= n; ++j) swap(A[k][j], A[p][j]);
    if (A[k][k]) {
      for (size_t i = k + 1; i <= n; ++i) {
        A[i][k] /= A[k][k];
        for (size_t j = k + 1; j <= n; ++j) A[i][j] -= A[i][k] * A[k][j];
      }
    } else {
      break;
    }
  }

  for (size_t i = 1; i <= n; ++i) {
    for (size_t j = 1; j <= n; ++j) {
      L[i][j] = ((i == j) ? 1 : ((i > j) ? A[i][j] : 0));
      U[i][j] = ((i <= j) ? A[i][j] : 0);
      A[i][j] = tmp[i][j];
    }
    delete[] tmp[i];
  }
  delete[] tmp;

  for (size_t i = 1; i <= n; ++i) {
    y[i] = b[i];
    for (size_t j = 1; j <= i - 1; ++j) y[i] -= L[i][j] * y[j];
  }

  for (size_t i = n; i >= 1; --i) {
    x[i] = y[i];
    for (size_t j = i + 1; j <= n; ++j) x[i] -= U[i][j] * x[j];
    x[i] /= U[i][i];
  }
}
```

```cpp

int main() {
  const int n = 84;
  // const int n = 4;

  double** A = new double*[n + 1];
  double* b = new double[n + 1];
  double** L = new double*[n + 1];
  double** U = new double*[n + 1];
  double* y = new double[n + 1];
  double* x = new double[n + 1];

  for (int i = 1; i <= n; i++) {
    A[i] = new double[n + 1];
    L[i] = new double[n + 1];
    U[i] = new double[n + 1];
  }

  for (int i = 1; i <= n; i++) {
    for (int j = 1; j <= n; j++) A[i][j] = 0;
    A[i][i] = 6;
    A[i][i - 1] = 8;
    A[i - 1][i] = 1;
    b[i] = 15;
  }
  b[1] = 7;
  b[n] = 14;

  solve(A, b, n, L, U, y, x);

  constexpr auto fmt_str = "{:10.5f}";

  cout << "A" << endl;
  for (int i = 1; i <= n; i++) {
    for (int j = 1; j <= n; j++) cout << format(fmt_str, A[i][j]);
    cout << endl;
  }

  cout << "b" << endl;
  for (int i = 1; i <= n; i++) cout << format(fmt_str, b[i]);

  cout << endl << "L" << endl;
```

```
  for (int i = 1; i <= n; i++) {
    for (int j = 1; j <= n; j++) cout << format(fmt_str, L[i][j]);
    cout << endl;
  }
  cout << "U" << endl;
  for (int i = 1; i <= n; i++) {
    for (int j = 1; j <= n; j++) cout << format(fmt_str, U[i][j]);
    cout << endl;
  }

  cout << "y" << endl;
  for (int i = 1; i <= n; i++) cout << format(fmt_str, y[i]);

  cout << endl << "x" << endl;
  for (int i = 1; i <= n; i++) cout << format(fmt_str, x[i]);
  cout << endl;

  for (int i = 0; i < n; i++) {
    delete[] A[i];
    delete[] L[i];
    delete[] U[i];
  }
  delete[] A;
  delete[] b;
  delete[] L;
  delete[] U;
  delete[] y;
  delete[] x;

  return 0;
}
```

```
 week 03 ⚡  dev
> .\run.ps1
A
    6.00000   1.00000   0.00000   0.00000
    8.00000   6.00000   1.00000   0.00000
    0.00000   8.00000   6.00000   1.00000
    0.00000   0.00000   8.00000   6.00000
b
    7.00000  15.00000  15.00000  14.00000
L
    1.00000   0.00000   0.00000   0.00000
    0.00000   1.00000   0.00000   0.00000
    0.00000   0.00000   1.00000   0.00000
    0.75000  -0.43750   0.23438   1.00000
U
    8.00000   6.00000   1.00000   0.00000
    0.00000   8.00000   6.00000   1.00000
    0.00000   0.00000   8.00000   6.00000
    0.00000   0.00000   0.00000  -0.96875
y
    7.00000  15.00000  15.00000  11.79688
x
    3.14315  -4.85887  11.00806 -12.17742
```

图 1　四阶矩阵的运行结果