Numerical Methods 5633

Lecture 5

Michaelmas Term 2017

Marina Krstic Marinkovic
mmarina@maths.tcd.ie

School of Mathematics
Trinity College Dublin

Organisational

- Assignment 0 – discussion of the solutions

	To appear	Submission DL
Assignment 1	**online**	15.11
Assignment 2	6.12	10.1

Solving linear algebraic equations

- The linear systems problem: **Ax =b**

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \cdots & a_{mn} \end{bmatrix}$$

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \quad b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

- Some methods for solving linear systems and their features:
  - Gaussian Elimination (pivoting, scaling)
  - Matrix factorisation, LU decomposition
  - Back/Forward substitution

Linear algebra: recall

➡ The linear systems problem: **$Ax = b$**

<u>Theorem 1:</u> Given the matrix $\mathbf{A} \in \mathbf{R}^{n \times n}$ the following statements are equivalent

1. A is nonsingular;
2. The columns of **A** form an independent set of vectors;
3. The rows of **A** form an independent set of vectors;
4. The linear system $\mathbf{Ax=b}$ has a unique solution for all vectors $b \in \mathbf{R}^n$;
5. The linear system $\mathbf{Ax=0}$ has only the trivial solution $x=\mathbf{0}$;
6. The determinant of A is nonzero: **datA≠0.**

Gauss elimination method

- Write the linear system as a single augmented matrix: **A′=[A|b]**

- Apply the solution algorithm to A′– combination of ***elementary row operations***:

  1. multiply row with a nonzero scalar c
  2. interchange two rows
  3. multiply a row by a nonzero scalar c and add the result to another row

- If we can obtain a matrix A2 from A1 applying 1.–3. then A1 and A2 are told to be ***row equivalent***

- ***Backward substitution (back-solve):*** process of computing the unknowns from a system that is in upper-triangular form is called

Gauss elimination method

- Write the linear system as a single augmented matrix: **A′=[A|b]**

> Theorem 2. Let A′ be augmented matrix corresponding to a linear system Ax=b and suppose that **A′** is row equivalent to **A′′=[T|c]**. Then the two systems have the same solution sets.

- Special classes of matrices:

$$U = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 4 & 5 \\ 0 & 0 & 6 \end{bmatrix}$$

$$L = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 3 & 0 \\ 4 & 5 & 6 \end{bmatrix}$$

- Upper triangular(U):
- Lower triangular(L):

- Goal: Reduce augmented matrix **A′** to the new augmented matrix **A′′=[U|c]** where U is *upper triangular matrix*.

Naive Gauss elimination method

- See example shown in class and R code example in **tmp/NumericalMethods/TestingCode/Lecture7/**

```
#naive gaussian elimination
for (i in (1:(n-1)))  #last column sikpped - no elements bellow the diagonal
{
    for (j in ((i+1):n))            #down i-th column
        {
        m=a[j,i]/a[i,i]             #compute the multiplier
        for (k in ((i+1):n))
            a[j,k]=a[j,k]-m*a[i,k]  #eliminate a[j][i] element
        b[j]=b[j]-m*b[i]            #right hand side is modified appropriately
        }
}
#backward solution algorithm
x[n]=b[n]/a[n,n]
for (i in (n-1):1)
{
sum=0
for (j in (i+1):n)
    sum=sum+a[i,j]*x[j]
x[i]=(b[i]-sum)/a[i,i]
}
```

- Note that the algorithm does not create 0 in the lower triangular half of A –> that would be waste of computer time!
- Backward solution (*back-solve*): $x_i = \frac{1}{a_{ii}}\left(b_i - \sum_{j=i+1}^{n} a_{ij}x_j\right)$

Gauss elimination method - problems

- Possible problem: In the process of the backward reconstruction of the solution: we divide with the element on the diagonal
- If this element is 0, the algorithm breaks down
- Elements on the diagonal: **pivot elements**
- Solution: interchange two rows to avoid a zero element on the diagonal –> ***PIVOTING***
    - partial pivoting:entries bellow the diagonal in the same column are examined
    - complete pivoting: searches on current and all subsequent columns (more stable, but expensive to implement)
    - scaling (scaled pivoting): scaling the rows before pivoting to avoid catastrophic cancellations

Gauss elimination method - pivoting

Partial pivoting:

1. Suppose we are about to work on the i-th column. We search the portion of the i-th column bellow the diagonal and find an element with the <u>largest</u> absolute value. Assume this happens in the p-th row.
2. Interchange rows i and p
3. Proceed with elimination

Exercise 1: Add partial pivoting to the provided code for Gaussian elimination.

➡ Total number of operations(mul.+div.+~~sub.~~): $\frac{\overset{1}{\cancel{2}}}{3}n^3 + O(n^2)$

LU Decomposition

- Gaussian elimination –> altering right hand side **(b)**
- Several systems with the same coefficient matrix can be solved only if *all* right hand side vectors are known initially
- Cost of finding the inverse matrix: AX=I, X=A^{-1} in addition to solving a single Gauss elimination system: $n^3 + O(n^2)$
- Total number of operations(mul.+div.): $\frac{1}{3}n^3 + O(n^2)$
- If elimination already performed, #additional op.: $n^2 + O(n^2)$

LU Decomposition

➡ LU decomposition algorithm:

1. Factor the matrix A into the product of a lower triangular and upper triangular matrix: **A = LU**

2. Solve two triangular systems (instead of a full linear system): **Ax = b -> Ux = y**, where **Ly = b**

3. First solve **Ly = b** and then **Ux = y** to get the solution.

Exercises

Exercise 1: Add partial pivoting to the provided code for Gaussian elimination.

Exercise 2: Add partial pivoting to your LU decomposition code.

Exercise 3: Check if the code you wrote works properly using predefined R routine: library('Matrix');
expand(lu(a))