10 장 LU 분해법

A*x=b 를 가우스 소거법으로 풀려면 먼저 [A, b] 합성행렬을 만들어 풀어가므로 계수벡터(b)가 여러 가지가 나오는 경우에는 비효율적이다. 또 가우스 소거법은 전진소거 과정이 n^3에 비례하므로 계수벡터(b)를 합치지 않고 따로 풀어나갈 수 있으면 보다 효율적이 될 것이다.

이제 가우스 소거법을 조금 변형시켜 L, U 분해를 하여 합성행렬을 만들지 않고 근을 구하도록 해보자.

10.1 LU 분해법 개요, 10.2 가우스 소거법과의 관계

A = L * U

d = L ₩ b   % b 벡터에 가우스 전진소거 방법을 처리하면 d 벡터가 된다.

x = U ₩ d

즉, d 벡터는 가우스 소거법에서 전진소거 과정을 거친 합성행렬의 마지막 열이다.

$$\begin{bmatrix} A_{11} & A_{12} & A_{13} & b_1 \\ A_{21} & A_{22} & A_{23} & b_2 \\ A_{31} & A_{32} & A_{33} & b_3 \end{bmatrix} \rightarrow \begin{bmatrix} H_{11} & H_{12} & H_{13} & H_{14} \\ 0 & H_{22} & H_{23} & H_{24} \\ 0 & 0 & H_{33} & H_{34} \end{bmatrix}$$

A + b   U   d

L 하삼각행렬은 주대각 원소가 1이고 아래 원소엔 가우스 소거법의 'factor'가 들어간다.

$$\begin{bmatrix} 1 & 0 & 0 \\ f_{21} & 1 & 0 \\ f_{31} & f_{32} & 1 \end{bmatrix}$$ f21=a21/a11, f31=a31/a11, f32=a'32/a'22

A 행렬을 가우스 소거(전진 소거)처리하면 U 상삼각행렬이 되며 이는 L * U = A 로 쓴다.
b 벡터를 가우스 소거 처리한 것이 d 벡터이며 이는 L * d = b 이다.
따라서 A = L * U, d = L ₩ b, x = U ₩ d 이다.

예제 10.1
정말 손으로 풀 수는 없고(푸는 것은 쉬우나 그럴 수는 없다) 다음과 같이 매틀랩을 이용하여 바닥을 헤쳐가며 풀어 본다.

```
>> A=[3 -0.1 -0.2; 0.1 7 -0.3; 0.3 -0.2 10]
A =
    3.0000   -0.1000   -0.2000
    0.1000    7.0000   -0.3000
    0.3000   -0.2000   10.0000
```

```
>> U=A;
>> U(2,:)=U(2,:)-U(1,:).*U(2,1)/U(1,1)
U =
     3.0000    -0.1000    -0.2000
    -0.0000     7.0033    -0.2933
     0.3000    -0.2000    10.0000
>> U(3,:)=U(3,:)-U(1,:).*U(3,1)/U(1,1)
U =
     3.0000    -0.1000    -0.2000
    -0.0000     7.0033    -0.2933
          0    -0.1900    10.0200
```

이로서 첫 열을 모두 영으로 만들었다(물론 1행은 빼고).

factor 는 f21 = a21/a11 = 0.0333

```
>> f21 = A(2,1)/A(1,1)
f21 =
     0.0333
```

f31 = a31/a11 = 0.1

```
>> f31=A(3,1)/A(1,1)
f31 =
     0.1000
```

이제 다음 계산(전진소거와 L 하삼각행렬의 f32 구하기)으로 넘어간다.

3행2열의 계수를 제거해야 하나 이를 연산처리 하면 계수가 바뀌어 f32를 구할 수 없으므로 먼저 factor를 구해 기억/저장해 놓고 전진소거를 마무리 한다.

```
>> f32=U(3,2)/U(2,2)       % A행렬은 원 행렬로 계수 처리가 되어있지 않으므로 안 된다.
f32 =
    -0.0271

>> U(3,:)=U(3,:)-U(2,:)*U(3,2)/U(2,2)
U =
     3.0000    -0.1000    -0.2000
    -0.0000     7.0033    -0.2933
    -0.0000          0    10.0120
```

U 상삼각행렬은 완성되었고 이젠 L 하삼각행렬을 만들 차례이다.

먼저 단위행렬을 준비한다.

```
>> L=eye(3)
```

```
L =
        1        0        0
        0        1        0
        0        0        1
```

전에 저장/기억해 놓은 계수 곱하기, 나누기(factor)를 불러 넣는다.

```
>> L(2,1)=f21; L(3,1)=f31; L(3,2)=f32
L =
      1.0000              0              0
      0.0333      1.0000              0
      0.1000     -0.0271       1.0000
```

이제 L*U=A 인지 확인해보자.

```
>> L*U
ans =
      3.0000     -0.1000     -0.2000
      0.1000      7.0000     -0.3000
      0.3000     -0.2000     10.0000
>> A
A =
      3.0000     -0.1000     -0.2000
      0.1000      7.0000     -0.3000
      0.3000     -0.2000     10.0000
```

예제 10.2 앞 문제를 풀어 근을 구하는 데 정말 손으로 할 순 없고 그냥 매틀랩을 쓴다.

```
A = L * U          % 이렇게 L,U로 분해 했고
d = L ₩ b
x = U ₩ d          % 이러면 해를 구하게 된다.
```

```
>> b=[7.85 -19.3 71.4]';
>> d=L₩b
d =
      7.8500
   -19.5617
     70.0843
>> x=U₩d
x =
      3.0000
     -2.5000
```

```
    7.0000
```

A * x = b 가 되는지 확인한다.

```
>> A*x
ans =
    7.8500
  -19.3000
   71.4000
```

10.2.1 피보팅을 이용한 LU 분해법

```
P*A = L*U          % A not pivoted, L&U pivoted, P unit matrix pivoted
L*d = P*b          % b도 pivoting 해야 한다.
U*x = d            % x는 U₩d하면 된다
```

10.2.2 매틀랩 함수: lu

10장 연습문제 10.5번으로 풀어 본다.

```
>> A=[2 -6 -1; -3 -1 6; -8 1 -2];
>> b=[-38 -34 -40]';
>> Ap=[A(3,:);A(1,:);A(2,:)] % pivoting 한 A 행렬
Ap =
    -8     1    -2
     2    -6    -1
    -3    -1     6
>> P=eye(3);
>> P=[P(3,:);P(1,:);P(2,:)]    %P행렬에 피보팅 순서를 넣는다
P =
     0     0     1
     1     0     0
     0     1     0
>> [L U]=lu(Ap);              % 피보팅 한 A 행렬을 L U 분해한다.
>> Pb=P*b                     % 상수 벡터를 똑같이 피보팅한다(계수벡터와 행 교환 순서가 같게).
Pb =
   -40
   -38
   -34
>> d=L₩Pb;                    % 피보팅한 상수벡터에 가우스 소거 처리를 하고
>> x=U₩d                      % 드뎌 해를 구한다
```

```
x =
     6.3425
     8.6239
    -1.0581
```

10.3 Cholesky 분해법

대칭행렬은 촐레스키 분해법으로 분해하여 근을 구하는 것이 효과적이다.

A = U' * U

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} u_{11} & 0 & 0 \\ u_{12} & u_{22} & 0 \\ u_{13} & u_{23} & u_{33} \end{bmatrix} * \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix}$$

따라서 다음과 같은 과정으로 U(나 U^T)를 구할 수 있다.

u11^2 = a11, u11 = sqrt(a11)

u11*u12 = a12, u12 = a12/u11

u11*u13 = a13, u13 = a13/u11

u12^2   u22^2 = a22, u22 = sqrt(a22 – u12^2)

u12*u13 + u22*u23 = a23, u23 = (a23 – u12*u13)/u22

u13^2 + u23^2 + u33^2 = a33, u33 = sqrt(a33 – (u13^2 + u23^2))

일단 분해되면 다음은 LU 분해법과 동일하다.

d = U' ₩ b

x = U ₩ d

예제 10.5

```
>> A=[6 15 55; 15 55 225; 55 225 979]
A =
      6     15     55
     15     55    225
     55    225    979
>> A'              % 대각행렬을 확인
ans =
      6     15     55
     15     55    225
     55    225    979

>> U=zeros(3)
```

```
U =
     0     0     0
     0     0     0
     0     0     0

>> U(1,1)=sqrt(A(1,1))
U =
    2.4495         0         0
         0         0         0
         0         0         0

>> U(1,2:end)=A(1,2:end)/U(1,1)
U =
    2.4495    6.1237   22.4537
         0         0         0
         0         0         0

>> U(2,2)=sqrt(A(2,2)-U(1,2)^2)
U =
    2.4495    6.1237   22.4537
         0    4.1833         0
         0         0         0

>> U(2,3)=(A(2,3)-U(1,2)*U(1,3))/U(2,2)
U =
    2.4495    6.1237   22.4537
         0    4.1833   20.9165
         0         0         0

>> U(3,3)=sqrt(A(3,3)-(U(1,3)^2+U(2,3)^2))
U =
    2.4495    6.1237   22.4537
         0    4.1833   20.9165
         0         0    6.1101
```

10.3.1 매틀랩 함수: chol()

위 예제를 chol() 함수로 풀어본다.

```
>> A
```

```
A =
     6    15    55
    15    55   225
    55   225   979

>> U=chol(A)
U =
    2.4495    6.1237   22.4537
         0    4.1833   20.9165
         0         0    6.1101
>> b
b =
    7.8500
  -19.3000
   71.4000
>> d=U'₩b
d =
    3.2047
   -9.3049
   31.7616
>> x=U₩d
x =
   24.1964
  -28.2154
    5.1982
>> A*x
ans =
    7.8500
  -19.3000
   71.4000
```

10.4 매틀랩 왼쪽 나눗셈

2014-11-7, 곽노태