Gauss 소거법

1. 미지수 소거법

- 이 방법은 연립방정식에서 미지수를 제거하는 과정으로, 다음과 같은 단계로 진행됩니다.

1. 하나의 미지수를 제거:

- 연립방정식에서 하나의 미지수를 제거하기 위해 한 방정식을 다른 방정식에 대해 연산합니다.
- 예를 들어, 첫 번째 방정식에서 x_1을 제거하고, 이를 통해 두 번째 방정식을 단순화할 수 있습니다.
- 이 과정은 궁극적으로 연립방정식의 크기를 줄이기 위한 단계입니다.

1. 소거된 방정식을 활용한 해 구하기:

- 하나의 방정식을 단순화하면, 남은 방정식에 대해 다시 미지수를 소거하는 과정을 반복합니다.
- 마지막에는 하나의 방정식과 미지수만 남게 되며, 이를 해결하면 나머지 미지수들도 역으로 대입하여 구할 수 있습니다.
- 이를 **후진 대입(Back Substitution)**이라고 합니다.

2. 가우스 소거법

- 가우스 소거법은 연립방정식을 행렬 형태로 변환하여 해결하는 방법으로, 두 단계로 나뉩니다.

2-1. 전진 소거**(Forward Elimination):**

- 미지수를 차례대로 제거하여 상삼각 행렬 형태로 변환하는 과정입니다. 이를 통해 연립방정식의 미지수 수를 하나씩 줄입니다.
- 주어진 연립방정식을 아래와 같은 행렬로 표현할 수 있습니다.

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & b_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ a_{31} & a_{32} & a_{33} & b_3 \end{bmatrix}$$

- 먼저, 첫 번째 방정식에서 x_1을 제거하고 나머지 방정식에 대해 계산을 수행합니다.
- 그 후 두 번째 방정식에서 x_2를 제거하고, 마지막 방정식에서 x_3를 남기게 합니다.
- 이 단계가 완료되면 방정식이 상삼각 행렬 형태가 됩니다.

2-2. 후진 대입**(Back Substitution):**

- 전진 소거 과정에서 상삼각 행렬을 얻으면, 마지막 방정식부터 미지수 값을 구하기 시작합니다.
- 상삼각 행렬을 사용하여 미지수를 역순으로 대입하여 각 값을 계산합니다.
- 예를 들어, 마지막 방정식에서 x_3을 구한 후, 이를 두 번째 방정식에 대입하여 x_2를 구하고, 마지막으로 첫 번째 방정식에 대입하여 x_1을 구하는 방식입니다.

$$x_3 = \frac{b'_3}{a'_{33}}$$

$$x_2 = \frac{b'_2 - a'_{23}x_3}{a'_{22}}$$

$$x_1 = \frac{b_1 - a_{12}x_2 - a_{13}x_3}{a_{11}}$$

예제 설명

다음과 같은 연립방정식을 가우스 소거법으로 풉니다:

$$\begin{aligned} 3x_1 - 0.1x_2 - 0.2x_3 &= 7.85 \\ 0.1x_1 + 7x_2 - 0.3x_3 &= -19.3 \\ 0.3x_1 - 0.2x_2 + 10x_3 &= 71.4 \end{aligned}$$

```
% 주어진 계수 행렬과 상수 항을 포함한 확장 행렬 G
G = [
    3, -0.1, -0.2, 7.85;
    0.1, 7, -0.3, -19.3;
    0.3, -0.2, 10, 71.4
];

% 행렬의 크기 (N은 미지수의 개수)
N = 3;

% 전진 소거 과정
for i = 1:N-1
    for j = i+1:N
        % i번째 행을 j번째 행에 대해 소거하는 과정
        G(j,:) = G(j,:) - (G(j,i) / G(i,i)) * G(i,:);
        %G(j, i:N+1) = G(j, i:N+1) - (G(j, i) / G(i, i)) * G(i, i:N+1);
    end
end

% 후진 대입을 통해 해를 구하는 과정
X = zeros(N,1); % 결과를 저장할 벡터
for i = N:-1:1
    % 후진 대입을 통해 각 미지수의 값을 계산
    X(i) = (G(i,N+1) - G(i,i+1:N) * X(i+1:N)) / G(i,i);
end

% 결과 출력
disp('해는 다음과 같습니다:');
```

```
해는 다음과 같습니다:
```

```
disp(X);
```

```
    3.0000
   -2.5000
    7.0000
```

1. 연산 합계 공식

- 주어진 두 식은 각각 다음과 같은 합계를 나타냅니다.

1. 첫 번째 합계:

$$\sum_{k=1}^{m} k = \frac{m(m+1)}{2} = \frac{m^2}{2} + O(m)$$

이 식은 1부터 m까지의 정수 합을 나타냅니다. 최종적으로 이 값은 $m^2/2$에 비례하고, 작은 항목들은 무시할 수 있다는 $O(m)$ 표기법을 사용합니다.

1. 두 번째 합계:

$$\sum_{k=1}^{m} k^2 = \frac{m(m+1)(2m+1)}{6} = \frac{m^3}{3} + O(m^2)$$

이 식은 1부터 m까지의 제곱합을 나타내며, 최종적으로 $m^3/3$에 비례하는 결과를 얻습니다. 여기서도 작은 항들은 무시합니다.

2. 전진 소거 연산 횟수

- 전진 소거 과정에서의 연산 횟수는 $\frac{m^3}{3} + O(m^2)$입니다.
- 이는 가우스 소거법에서 $n \times n$ 크기의 행렬에 대해 전진 소거를 수행할 때의 복잡도를 나타내며, 전체 연산량은 대략 $O(m^3)$에 비례합니다.

3. 후진 대입 연산 횟수

- 후진 대입 과정에서의 연산 횟수는 $\frac{m^2}{2} + O(m)$입니다.
- 후진 대입은 전진 소거 이후에 남은 상삼각 행렬에서 미지수들을 차례로 계산하는 과정이며, 전진 소거에 비해 연산량이 작습니다.

4. 총 계산 시간 분석

- 시스템의 크기 m이 커질수록 계산 시간은 전진 소거 단계에서 주로 발생하며, 이는 $O(m^3)$에 비례하여 증가합니다.

- 즉, 대부분의 연산이 전진 소거 단계에서 발생하고, 후진 대입 단계에서는 상대적으로 적은 연산이 필요합니다.