Naïve Gauss Elimination

Ch.9

Naïve Gauss Elimination
Linear Algebra Review

Elementary Matrix Operations Needed for Elimination Methods:
• Multiply an equation in the system by a non-zero real number.
• Interchange the positions of two equation in the system.
• Replace an equation by the sum of itself and a multiple of another equation of the system.

Naïve Gauss Elimination
Similar to Elimination of Unknowns

1. Forward Elimination
2. Backward Substitution

Naïve because we don’t consider division by zero to be a possibility

\begin{align*}
a_1 x_1 + a_2 x_2 + a_3 x_3 &= b_1 \\
a_2 x_1 + a_2 x_2 + a_3 x_3 &= b_2 \\
a_3 x_1 + a_2 x_2 + a_3 x_3 &= b_3 \\
a_4 x_1 + a_2 x_2 + a_3 x_3 &= b_4
\end{align*}
Naïve Gauss Elimination

Similar to Elimination of Unknowns

1. Forward Elimination of Unknowns
   1. Reduce the coefficient matrix $[A]$ to an upper triangular system
   2. Eliminate $x_1$ from the 2nd to nth Eqns.
   3. Eliminate $x_2$ from the 3rd to nth Eqns.
   4. Continue process until the nth equation has only 1 Non-Zero coefficient

\[
\begin{align*}
\begin{bmatrix}
a_{11} & a_{12} & a_{13} \\
a_{21} & a_{22} & a_{23} \\
a_{31} & a_{32} & a_{33}
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3
\end{bmatrix}
&=
\begin{bmatrix}
b_1 \\
b_2 \\
b_3
\end{bmatrix} \\
\begin{bmatrix}
a_{11} & a_{12} & a_{13} \\
0 & a_{22} & a_{23} \\
0 & 0 & a_{33}
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3
\end{bmatrix}
&=
\begin{bmatrix}
b'_1 \\
b'_2 \\
b'_3
\end{bmatrix}
\end{align*}
\]

Naïve Gauss Elimination

1. Forward Elimination
   \[a_{11}x_1 + a_{12}x_2 + a_{13}x_3 = b_1 \] (1)
   \[a_{21}x_1 + a_{22}x_2 + a_{23}x_3 = b_2 \] (2)
   \[a_{31}x_1 + a_{32}x_2 + a_{33}x_3 = b_3 \] (3)

Eliminate $x_1$ from equation (2). Multiply (1) by $a_{21}/a_{11}$, then subtract the result from (2)

\[
\begin{align*}
\left(a_{22} - \frac{a_{21}a_{12}}{a_{11}}\right)x_2 + \left(a_{23} - \frac{a_{21}a_{13}}{a_{11}}\right)x_3
&= b_2 - \frac{a_{21}}{a_{11}}b_1 \\
\Rightarrow
a'_{22}x_2 + a'_{23}x_3 &= b'_2 \quad (2')
\end{align*}
\]

Naïve Gauss Elimination

1. Forward Elimination
   \[a_{11}x_1 + a_{12}x_2 + a_{13}x_3 = b_1 \] (1)
   \[a_{21}x_1 + a_{22}x_2 + a_{23}x_3 = b_2 \] (2)
   \[a_{31}x_1 + a_{32}x_2 + a_{33}x_3 = b_3 \] (3)

Eliminate $x_1$ from equation (2). Multiply (1) by $a_{21}/a_{11}$, then subtract the result from (2)

\[
\begin{align*}
\left(a_{22} - \frac{a_{21}a_{12}}{a_{11}}\right)x_2 + \left(a_{23} - \frac{a_{21}a_{13}}{a_{11}}\right)x_3
&= b_2 - \frac{a_{21}}{a_{11}}b_1 \\
\Rightarrow
a'_{22}x_2 + a'_{23}x_3 &= b'_2 \quad (2')
\end{align*}
\]
Naïve Gauss Elimination

1. Forward Elimination
   \[ a_1 x_1 + a_2 x_2 + a_3 x_3 = \bar{b}_1 \] (1)
   \[ + a'_2 x_2 + a'_3 x_3 = \bar{b}_2' \] (2')
   \[ a_2 x_2 + a'_3 x_3 = \bar{b}_3 \] (3) ---- Elimination Row

   Eliminate \( x_1 \) from (3). Multiply (1) by \( a_3 a_1 / a_1 \), then subtract the result from (3)
   \[ \left( a_3 - \frac{a_3}{a_1} a_1 \right) x_2 + \left( a'_3 - \frac{a_3}{a_1} a_1 \right) x_3 = \bar{b}_3 - \frac{a_3}{a_1} \bar{b}_1 \]
   \[ a'_1 x_1 + a'_3 x_3 = \bar{b}'_3 \]

2. Pivot Equation
   \[ a_1 x_1 + a_2 x_2 + a_3 x_3 = \bar{b}_1 \] (1)
   \[ + a'_2 x_2 + a'_3 x_3 = \bar{b}_2' \] (2')
   \[ a_2 x_2 + a'_3 x_3 = \bar{b}_3 \] (3') ---- Elimination Row

   Eliminate \( x_2 \) from (3'). Multiply (2') by \( a'_3 a_2 / a_2 \), then subtract the result from (3')
   \[ \left( a'_3 - \frac{a'_3}{a_2} a_2 \right) x_2 + \left( a'_2 - \frac{a'_3}{a_2} a_2 \right) x_3 = \bar{b}_3' - \frac{a'_3}{a_2} \bar{b}_2' \]
   \[ a'_1 x_1 + a'_3 x_3 = \bar{b}'_3 \]

3. Solve for \( x_3 \)
   \[ x_3 = \frac{\bar{b}'_3}{a'_3} \]

---

Naïve Gauss Elimination

1. Forward Elimination
   \[ a_1 x_1 + a_2 x_2 + a_3 x_3 = \bar{b}_1 \] (1)
   \[ + a'_2 x_2 + a'_3 x_3 = \bar{b}_2' \] (2')
   \[ a_2 x_2 + a'_3 x_3 = \bar{b}_3 \] (3') ---- Elimination Row

   Eliminate \( x_1 \) from (3). Multiply (1) by \( a_3 a_1 / a_1 \), then subtract the result from (3)
   \[ \left( a_3 - \frac{a_3}{a_1} a_1 \right) x_2 + \left( a'_3 - \frac{a_3}{a_1} a_1 \right) x_3 = \bar{b}_3 - \frac{a_3}{a_1} \bar{b}_1 \]
   \[ a'_1 x_1 + a'_3 x_3 = \bar{b}'_3 \]

2. Pivot Equation
   \[ a_1 x_1 + a_2 x_2 + a_3 x_3 = \bar{b}_1 \] (1)
   \[ + a'_2 x_2 + a'_3 x_3 = \bar{b}_2' \] (2')
   \[ a_2 x_2 + a'_3 x_3 = \bar{b}_3 \] (3') ---- Elimination Row

   Eliminate \( x_2 \) from (3'). Multiply (2') by \( a'_3 a_2 / a_2 \), then subtract the result from (3')
   \[ \left( a'_3 - \frac{a'_3}{a_2} a_2 \right) x_2 + \left( a'_2 - \frac{a'_3}{a_2} a_2 \right) x_3 = \bar{b}_3' - \frac{a'_3}{a_2} \bar{b}_2' \]
   \[ a'_1 x_1 + a'_3 x_3 = \bar{b}'_3 \]

3. Solve for \( x_3 \)
   \[ x_3 = \frac{\bar{b}'_3}{a'_3} \]
Naïve Gauss Elimination

2. Backwards substitution:
   \[ a_{12} x_2 + a_{13} x_3 = b_1 \] (1)
   \[ a_{22} x_2 + a_{23} x_3 = b_2 \] (2')
   \[ a_{33} x_3 = b_{33} \] (3')
   \[ x_3 = \frac{b_{33}}{a_{33}} \]
   From (2')
   \[ x_2 = \frac{b_2 - a_{23} x_3}{a_{22}} \]
   From (1)
   \[ x_1 = \frac{b_1 - a_{12} x_2 - a_{13} x_3}{a_{11}} \]

Naïve Gauss Elimination

In General, the last equation should reduce to:
\[
x_i = \frac{b^{(i-1)}_{i}}{a^{(i-1)}_{ii}} - \sum_{j=i+1}^{n} a^{(i-1)}_{ij} x_j
\]
\[
x_i = \frac{b^{(i-1)}_{i}}{a^{(i-1)}_{ii}}
\]
General form is how we will numerically implement.

Note: Since we normalize with the pivot element, if it is zero, we have a problem → Naïve method

Naïve Gauss Elimination

Example:
\[
2x_1 + x_2 + 3x_3 = 1 \quad (1)
4x_1 + 4x_2 + 7x_3 = 1 \quad (2)
2x_1 + 5x_2 + 9x_3 = 3 \quad (3)
\]
Naïve Gauss Elimination – Numerically Implementing

\[
\begin{bmatrix}
  a_{11} & a_{12} & a_{13} \\
  a_{21} & a_{22} & a_{23} \\
  a_{31} & a_{32} & a_{33}
\end{bmatrix}
\begin{bmatrix}
  x_1 \\
  x_2 \\
  x_3
\end{bmatrix}
= 
\begin{bmatrix}
  b_1 \\
  b_2 \\
  b_3
\end{bmatrix}
\]

3 Main Loops: Forward Elimination
1. Pivot Row – from 1st row to the n-1 row, move down, we will call the pivot row, row k.
2. Elimination Row – Rows below Pivot row, where eliminations take place (top down), call this the ith row.
3. Element transform Loop – columns, jth column. Move left to right.

What we do: Forward Elimination
A. Normalization Step: multiply the kth row elements \(a_{kj}\) by \(-a_{ik}/a_{kk}\)
B. Add the result of step A, to \(a_{ij}\)
C. Calculate the new b’s or right hand side terms

\[
\begin{bmatrix}
  a_{11} & a_{12} & a_{13} \\
  a_{21} & a_{22} & a_{23} \\
  a_{31} & a_{32} & a_{33}
\end{bmatrix}
\begin{bmatrix}
  x_1 \\
  x_2 \\
  x_3
\end{bmatrix}
= 
\begin{bmatrix}
  b_1 \\
  b_2 \\
  b_3
\end{bmatrix}
\]

Instead of saving \(a'_{ij}\)
we save as \(a_{ij}\)

What we do: Back Substitution
For the nth Row:

\[
x_n = \frac{b_n}{a_{nn}}
\]

Now, work backwards row by row, right to left (n-1 row and n column)

\[
x_i = \frac{b_i - \sum a_{ij}x_j}{a_i}
\]
Naïve Gauss Elimination – Pseudocode

%Forward Elimination to build an upper triangular matrix
for k=1:n-1
    for i=k+1:n
        factor = a(i,k)/a(k,k); %normalizing factor (Step A)
        for j=k+1:n %move across the columns loop
            a(i,j) = a(i,j) - factor*a(k,j); % (Step B)
        end
        b(i) = b(i) - factor*b(k); % (Step C)
    end
end

%Backward Substitution
x(n)=b(n)/a(n,n); %solve for the last x value
for i=n-1:-1:1
    sum = 0;
    for j=i+1:n
        sum = sum + a(i,j)*x(j);
    end
    x(i)=(b(i)-sum)/a(i,i);
end

Problems with Naïve Elimination Methods

1. Division by zero
   • If a_{11} = 0, then the 1st elimination step yields division by zero.
   • "pivoting" technique will be used to avoid this problem

2. R.O. Error
   • Every result is dependant on previous results → R.O. error can propagate
   • “Rule of Thumb” – if n > 100
   • Double precision will help

3. Ill Conditioned Systems (D~0)
   • Small changes in the coefficient (a_i) matrix result in large changes in the solution
   • Or, alternatively a wide range of answers (x_i’s) satisfy the equations
   • R.O. error can produce small changes in coefficients that can lead to large errors, (Check by slightly changing the coefficients and seeing the effect on the results)

4. Singular Systems: (D=0)
   • One or more equations are identical
   • We have (n-f) equations and n unknowns

QUICK way to check D
• After the forward elimination evaluate the determinant of the modified coefficient matrix

\[
\begin{vmatrix}
  a_{11} & a_{12} & a_{13} \\
  0 & a'_{22} & a'_{23} \\
  0 & 0 & a'_{33}
\end{vmatrix}
\]

\[
D = \begin{vmatrix}
  a_{11} & a_{12} & a_{13} \\
  0 & a'_{22} & a'_{23} \\
  0 & 0 & a'_{33}
\end{vmatrix} = a_{11} \cdot a'_{22} \cdot a'_{33}
\]
Problems with Naïve Elimination Methods

When checking D, how small is too small? Solution: Standardize the determinant.

Scale Equations – such that the maximum coefficient for any equation is 1.

\[3x_1 + 2x_2 = 12\]
\[-x_1 + 3x_2 = 3\]
\[\rightarrow D = 11\]

Divide (1) by 3 and (2) by 3.

\[x_1 + \frac{2}{3}x_2 = 4\]
\[-\frac{1}{3}x_1 + x_2 = 1\]
\[\rightarrow D = 1.22\]

Problems with Naïve Elimination Methods

Now for a 2x2 matrix D takes on Values:

\[a_{11}x_1 + a_{12}x_2 = b_1\]
\[a_{21}x_1 + a_{22}x_2 = b_2\]
\[D = a_{11}a_{22} - a_{12}a_{21}\]
\[|D| \leq 2\]

Methods for Improving Solutions

1. Use More Significant digits
2. Partial Pivoting
   - Avoid division by zero or vary small numbers
     a) Before normalizing in Gauss elimination, find the largest element (absolute value) in the first column
     b) Reorder the equations so that the largest element is the pivot element
     c) Repeat for each elimination step – i.e., 2nd application would find the largest element in the 2nd column (below the 1st Equation) and seek the largest pivot element.
Methods for Improving Solutions – Partial Pivoting

code

```
p=k;                     %assume row with largest coefficient
big=abs(a(k,k))         %assume the diagonal term is largest
for ii = k+1:n          %move down the rows to check elements
    dummy=abs(a(ii,k));
    if dummy > big      %if the element is bigger swap it out
        big=dummy;
        p=ii;           %rename the largest row
    end
end %end for loop

%if p is not equal to k, we need to swap row k with row p
%if p is equal to k, then we don't do anything
if p~=k
    for jj=k:n %move across columns to swap coefficient values
        dummy=a(p,jj); %temporarily store the element
        a(p,jj)=a(k,jj);
        a(k,jj)=dummy;
    end
    dummy = b(p); %now swap right hand side values
    b(p) = b(k);
    b(k) = dummy;
end
```

Show Matlab example

### Methods for Improving Solutions

3. **Scaling** – helps make pivoting decisions

\[
\begin{align*}
3x_1 + 70,000x_2 &= 40,000 \\
-0.0000428x_1 + 1x_2 &= 0.5714 \\
-0.0000428x_1 + 1x_2 &= 0.5714
\end{align*}
\]

What problem could arise here

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scale</td>
<td>Scale all rows so that the maximum coefficient value in any row is one.</td>
</tr>
</tbody>
</table>

### Methods for Improving Solutions

3. **Scaling** –

- We have seen that adding and subtracting of numbers with very different magnitudes can result in RO error.
- Scale all rows so that the maximum coefficient value in any row is one.
- **Note:** Scaling by very large numbers can potentially introduce RO error.

**Suggestion:**
- Employ scaling only to make a decision regarding pivoting.
- Comparison & row switching are not subject to RO error.
- Complete solution using original coefficients.
Gauss-Jordan Elimination

- Variation of Gauss Elimination
- When an unknown is eliminated it is eliminated from all equations, not just subsequent ones (Diagonal Matrix Results)
- All rows are normalized by their pivot element
- Identity Matrix results

\[
\begin{bmatrix}
\begin{array}{ccc|c}
 a_{11} & a_{12} & a_{13} & b_1 \\
 a_{21} & a_{22} & a_{23} & b_2 \\
 a_{31} & a_{32} & a_{33} & b_3 \\
\end{array}
\end{bmatrix}
\begin{bmatrix}
 x_1 \\
 x_2 \\
 x_3 \\
\end{bmatrix}
=
\begin{bmatrix}
 b_1 \\
 b_2 \\
 b_3 \\
\end{bmatrix}
\]

- Almost identical to Gauss Elimination but, more operations are required
- No back substitution step

Methods for Improving Solutions – Partial Pivoting

Overhead Gauss Jordan example