Numerical linear Algebra...

Two techniques

1. Direct method - from before... (quadratic formula)
2. Iterative method.. (Newton's method)

Math 330 linear Algebra...

row op. → Gaussian elimination → $A=LU$

column op. → Gram-schmidt orthogonalization → $A=QR$

Householder's algorithm is also used for the same thing...

Cramer's rule..

These methods (with some modifications) work well on a computer ← to control rounding error...

Review Gaussian elimination (row operations)

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

$r_2 \leftarrow r_2 - 4r_1$

$r_3 \leftarrow r_3 - 2r_1$

$$\begin{bmatrix} 1 & 2 & 3 \\ 0 & -3 & -6 \\ 0 & -4 & -9 \end{bmatrix}$$

$r_3 \leftarrow r_3 - \frac{4}{3}r_2$

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

$$-9 - \frac{4}{3}(-6) = \frac{24-27}{3} = -1$$

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

If no mistakes

$$A = LU$$

```
julia> L=[1 0 0; 4 1 0; 2 4/3 1]
3×3 Matrix{Float64}:
 1.0  0.0      0.0
 4.0  1.0      0.0
 2.0  1.33333  1.0

julia> U=[1 2 3; 0 -3 -6; 0 0 -1]
3×3 Matrix{Int64}:
 1   2   3
 0  -3  -6
 0   0  -1

julia> L*U
3×3 Matrix{Float64}:
 1.0  2.0   3.0
 4.0  5.0   6.0
 2.0  0.0  -3.0
```

Similar notation to Matlab

Matrix-Matrix multiplication is built in.

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

When working by hand we swap rows sometimes either to make the arithmetic easier or because a 0 appeared where the pivot was supposed to be...

There is some flexibility in swapping rows. Idea is to swap rows in a way that reduces rounding error.

When working by hand we swap row to avoid fractions as much as possible... when reducing rounding error swap rows to make the denominator as big as possible..

Now try to maximize the denominators

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

$r_1 \leftrightarrow r_2$

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

$r_2 \leftarrow r_2 - \frac{1}{4} r_1$

$r_3 \leftarrow r_3 - \frac{1}{2} r_1$

$$\begin{bmatrix} 4 & 5 & 6 \\ 0 & 3/4 & 3/2 \\ 0 & -5/2 & -6 \end{bmatrix}$$

$2 - \frac{5}{4} = \frac{8-5}{4} = \frac{3}{4}$

$3 - \frac{6}{4} = \frac{6-3}{2} = \frac{3}{2}$

$0 - \frac{5}{2} = -5/2$

$-3 - \frac{6}{2} = -6$

$r_2 \leftrightarrow r_3$

$$\begin{bmatrix} 4 & 5 & 6 \\ 0 & -5/2 & -6 \\ 0 & 3/4 & 3/2 \end{bmatrix}$$

$r_3 \leftarrow r_3 + \frac{3/4}{5/2} r_2$

note: whatever rounding error were in r_2 they are reduced before being propagated to r_3.

$$U = \begin{bmatrix} 4 & 5 & 6 \\ 0 & -5/2 & -6 \\ 0 & 0 & -3/10 \end{bmatrix}$$

$$\frac{3}{2} + \frac{3}{10}(-6) = \frac{15-18}{10} = \frac{-3}{10}$$

```
julia> z=lu(A)
LU{Float64, Matrix{Float64}}
L factor:
3×3 Matrix{Float64}:
 1.0    0.0  0.0
 0.5    1.0  0.0
 0.25  -0.3  1.0
U factor:
3×3 Matrix{Float64}:
 4.0   5.0   6.0
 0.0  -2.5  -6.0
 0.0   0.0  -0.3
```

$$L = \begin{bmatrix} 1 & 0 & 0 \\ 1/2 & 1 & 0 \\ 1/4 & -3/10 & 1 \end{bmatrix}$$

```
julia> U=[4 5 6; 0 -5/2 -6; 0 0 -3/10]
3×3 Matrix{Float64}:
 4.0   5.0   6.0
 0.0  -2.5  -6.0
 0.0   0.0  -0.3

julia> L=[1 0 0; 1/2 1 0; 1/4 -3/10 1]
3×3 Matrix{Float64}:
 1.0    0.0  0.0
 0.5    1.0  0.0
 0.25  -0.3  1.0

julia> L*U
3×3 Matrix{Float64}:
 4.0  5.0   6.0
 2.0  0.0  -3.0
 1.0  2.0   3.0
```

Push "?" for help

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

Computers have no trouble doing the bookkeeping to figure out how the rows were permuted...

? lu to get

Component	Description
F.L	L (lower triangular) part of LU
F.U	U (upper triangular) part of LU
F.p	(right) permutation Vector
F.P	(right) permutation Matrix

```
julia> z.L
3×3 Matrix{Float64}:
 1.0    0.0  0.0
 0.5    1.0  0.0
 0.25  -0.3  1.0

julia> z.U
3×3 Matrix{Float64}:
 4.0   5.0   6.0
 0.0  -2.5  -6.0
 0.0   0.0  -0.3

julia> z.L*z.U
3×3 Matrix{Float64}:
 4.0  5.0   6.0
 2.0  0.0  -3.0
 1.0  2.0   3.0
```

```
julia> z.p
3-element Vector{Int64}:
 2
 3
 1

julia> A
3×3 Matrix{Int64}:
 1  2   3
 4  5   6
 2  0  -3

julia> A[z.p,:]
3×3 Matrix{Int64}:
 4  5   6
 2  0  -3
 1  2   3
```

← actually LU

Thus $PA = LU$ or $A = P^{-1}LU$

```
julia> z.P
3×3 Matrix{Float64}:
 0.0  1.0  0.0
 0.0  0.0  1.0
 1.0  0.0  0.0

julia> z.P'*z.L*z.U
3×3 Matrix{Float64}:
 1.0  2.0   3.0
 4.0  5.0   6.0
 2.0  0.0  -3.0
```

← keep track of how rows are swapped...

$$\Rightarrow \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 2 & 0 & -3 \end{bmatrix}$$