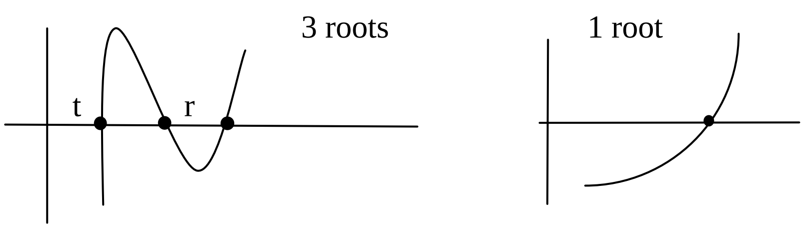
CS 314 Summer 2011: Week Two

<u>6/20</u>
-Order of convergence
-Assume that p is a zero of the function and E = p - p_n is the approximation of error.
If two positive constants exist when A != 0 and p = 0 exits,

$$\lim_{n\to\infty} \frac{|p-p_{n+1}|}{|p-p_n|R} = \lim_{n\to\infty} \frac{E_{n+1}}{|E_n|R} = \text{A}$$

The larger R, the faster the convergence
- If R=1, then the convergence is linear
- If R=2, then the convergence is quadratic

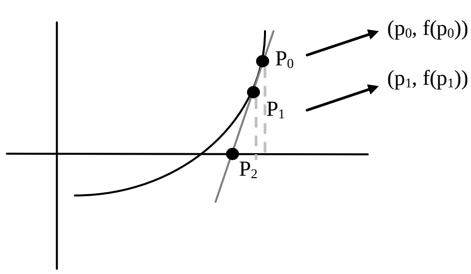
Usually, the more solutions (roots) the equation has, the slower the convergence



Finding a root with Newton Raphson has rate of convergence R=2 (quadratic) when the function has a single root.

Secant Method

- The Newton Raphson method requires to know the derivative and also it requires to evaluate both f(p_k) and f'(p_k)at every iteration
- The second method does not require a derivative and it needs only one evaluation of f(x)
- With a single root, the order of convergence is R=1.618
- Second method uses two initial points close to the root P_0, P_1



P_2 will be the intersection of the line with the x-axis

$$2.\ m = \frac{0 - f(p_1)}{p_2 - p_1}$$

$$1.\ m = \frac{f(p_1) - f(p_0)}{p_1 - p_0}$$

Assigning 1 and 2:

$$\frac{f(p_1) - f(p_0)}{p_1 - p_0} = \frac{0 - f(p_1)}{p_1 - p_0} \Rightarrow \quad p_2 - p_1 = \frac{-f(p_1)}{f(p_1) - f(p_0)}(p_1 - p_0)$$

$$\Rightarrow \quad p_2 = p_1 - \frac{f(p_1)}{f(p_1) - f(p_0)}(p_1 - p_0)$$

Secant Method

Example:

$f(x) = \sin(x) - .5 = 0$ $\qquad p_{k+1} = -\frac{f(p_k)(p_k - p_{k-1})}{f(p_k) - f(p_{k-1})}$

K	p_k	$F(p_k)$	P_{k+1}
0	$P_0 = 0$	-.5	1
1	$P_1 = 1$	.3415	.594
2	$P_2 = .594$	.0597	.50798
3	$P_3 = .50798$	-.0135	.5238

Exact solution: $x = 30^{\circ} * \pi/180$

Secant Method and Newton Raphson:

You can obtain the Newton Raphson method from the secant method by making $p_1 = p_0 + \Delta x$ and making $\Delta x \rightarrow 0$

$$p_{k+1} = p_k - \frac{f(p_k)(p_k - p_{k-1})}{f(p_k) - f(p_{k-1})}$$

$$p_k - p_{k-1} = \Delta x$$

$$p_{k+1} = p_k - f(p_k) \lim_{n \to \infty} \frac{\Delta x}{f(p_k) - f(p_k - \Delta x)}$$

$$p_{k+1} = p_k - \frac{f(p_k)}{f'(p_k)}$$

The Solution of linear systems - approximate solutions

A X = B

Ex. 6x + 3y + 2z = 29

3x + y + 2z = 17

2x + 3y + 2z = 21

$$\Rightarrow \begin{bmatrix} 6 & 3 & 2 \\ 3 & 1 & 2 \\ 2 & 3 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 29 \\ 17 \\ 21 \end{bmatrix}$$

A X = B

-Manipulating matrices easier than manipulating equation

Ex. Determining ranking for website

=> find maximum pages that point to page and base relevance on this

=> can create system of linear equation that can be evaluated by iterations

6/21

Definition of Vector:

$X = (x_1, x_2, \ldots, x_n)$

- $x_1, x_2, \ldots, x_n$ are called components of x

N-Dimensional space

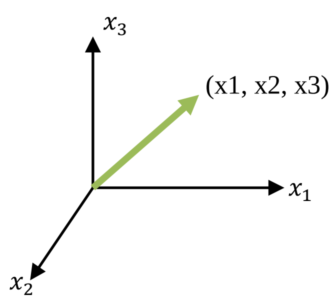
-the set of all N-dimensional vector

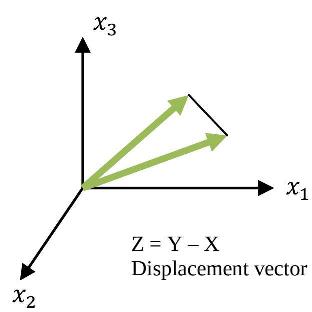
When a vector is used to determine a position it is called "position vector"

When a vector is used to denote movement between two points, it is called a displacement vector

X = Y if and only if

$X_J = Y_J$ for J = 1, 2, …, n





Sum:

$$X + Y = (x_1 + y_1, x_2 + y_2, \ldots, x_n + y_n)$$

Negative:

$$-X = (-x_1, -x_2, \ldots, -x_n)$$

Difference:

$$Y - X = Y + (-X)$$

Scalar Multiplycation:

$$cX = (cx_1, cx_2, \ldots, cx_n)$$

Linear Combination:

$$cX + dY = (cx_1 + dy_1, cx_2 + dy_2, \ldots, cx_n + dy_n)$$

Dot Product:

$$X \cdot Y = x_1y_1 + x_2y_2 + \cdots + x_ny_n = |x||y|cos\theta xy$$

The euclidean norm of a vector

$$||x|| = (x_1^2 + x_2^2 + \cdots + x_n^2)$$

-Scalar multiplication cX stretches a vector when c > 1 and shrinks

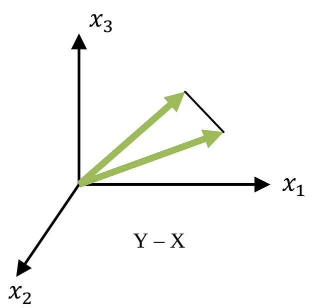
$$||cX|| = (c^2x_1^2 + c^2x_2^2 + \cdots + c^2x_n^2) = \text{c}(x_1^2 + x_2^2 + \cdots + x_n^2) = c||x||$$

Relationship between product and norm of a vector.

$$||x||^2 = x_1^2 + x_2^2 + \cdots + x_n^2 = X \cdot X$$

Displacement Vector

If x and y represent two points in the space, the displacement vector from x->y is given by the difference



$$||y - x|| = ((y_1^2 - x_1^2) + (y_2^2 - x_2^2) + \cdots + (y_n^2 - x_n^2))\text{^}(1/2)$$

Ex. X = (-1,3,2) and Y = (3,5,2)

Sum X+Y= ((-1+3), (3+5), (2+2))
Difference X-Y= (-1-3, 3-5, 2-2)
Scalar 2X = (2*-1, 2*3, 2*2) = (-2,6,4)
Length ||X|| = ((-1)^2+3^2+2^2)^(1/2) = (1+9+4)^(1/2)
Dot Product X ·Y = (-1*3, 3*5, 2*2) = (-3, 15, 4)
Displacement X-Y = (3-(-1), 5-3, 2,2) = (4, 2, 0)

Sometimes it is useful to write vectors as columns instead of rows.

$$X = \begin{matrix} x1 \\ x2 \\ x3 \end{matrix} \quad Y = \begin{matrix} y1 \\ y2 \\ y3 \end{matrix}$$

We use superscript ‘ to convert or transpose a vector from row to column or vice versa

$$(x1, x2, \ldots xn)' = \begin{matrix} x1 \\ x2 \\ x3 \end{matrix}$$

Vector Algebra:

Y+X=X+Y	Commutative
X=0+X=X+0	Additive Identity
X-X=X+(-X)=0	Additive Inverse

(X+Y)+Z=X+(Y+Z) Associative Property
(a+b)*x=ax+bx Distributive Property w/ Scalar
a(x+y)=ax+ay Distributive Property for Vectors

Matrices

If A is a matrix, then the letter a_{ij} denotes the number in the location i,j (ith row and jth column)

$$A = \begin{matrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{matrix} = \text{3x3 matrix}$$

You can see an M*N matrix as M rows of N dimensional vectors

$$A = \begin{matrix} v_1 \\ \cdots \\ v_n \end{matrix} = [v1, \ldots, vn]^T$$

Or you can see A as N columns w/ M-dimensional vectors
A = [c1, c2, …, cn]

Operators

Equality:
A = B if and only if $a_{ij} = b_{ij}$

Sum:
$A+B=[a_{ij} + b_{ij}]_{M*N}$

Negation:
$-A=[-a_{ij}]_{M*N}$

Difference:
$A-B=[a_{ij}-b_{ij}]_{M*N}$

Scalar Multiplycation:
$cA=[ca_{ij}]_{M*N}$

$$\text{Ex. } A = \begin{matrix} 2 & 1 \\ 4 & 6 \\ 12 & 2 \end{matrix} \quad B = \begin{matrix} 1 & 7 \\ 6 & 2 \\ 2 & 1 \end{matrix}$$

3A+2B

$$3A = \begin{matrix} 6 & 3 \\ 12 & 18 \\ 9 & 6 \end{matrix} \quad 2B = \begin{matrix} 2 & 14 \\ 12 & 4 \\ 4 & 2 \end{matrix}$$

$$3A+2B = \begin{matrix} 8 & 17 \\ 24 & 22 \\ 13 & 8 \end{matrix}$$

Properties of Matrixes:

B+A = A+B Commutative
0+A=A+0 Additive Identy
A-A=A+(-A)=0 Additive Inverse
(A+B)+C=A+(B+C) Associative
(p+q)/A = pA+qA Distributive
p(A+B)=pA+pB Distributive for matrices

Matrix Multiplycation:

A*B = C Where $A=[a_{ij}]$ M*N
$B=[b_{ij}]$N*P
$C=[c_{ij}]$M*P

$C_{ij} = SUM(a_{ik}b_{kj})$
For i=1,2,…M
For j=1,2,…P

Ex. $A = \begin{matrix} 3 & 1 & 2 \\ 2 & 6 & 5 \end{matrix}$ $B = \begin{matrix} 6 & 4 & 6 & 4 \\ 2 & 1 & 3 & 1 \\ 1 & 2 & 1 & 0 \end{matrix}$

C = A*B =

$$\begin{matrix} 3*6+1*2+2*1 & 3*4+1*1+2*2 & 3*6+1*3+2*1 & 3*4+1*1+2*0 \\ 2*6+6*2+5*1 & 2*4+6*1+5*2 & 2*6+6*3+5*1 & 2*4+6*1+5*0 \end{matrix}$$

$$= \begin{matrix} 22 & 17 & 23 & 15 \\ 29 & 24 & 35 & 19 \end{matrix}$$

A*B != B*A No commutative property in multiplication

6/22

Special Matrices

$0=[0]_{M*N}$

$$0_{3*2}=\begin{matrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{matrix}$$

Identity Matrix

$I_n = [\delta_{ij}]_{N*N}$ where $\delta_{ij}= \{^{1\ i=j}_{0\ i\ !=\ j}$

Ex. $I2 = \begin{matrix} 1 & 0 \\ 0 & 1 \end{matrix}$

$$\begin{matrix} 1 & 0 \\ 0 & 1 \end{matrix} \begin{matrix} a11 & a12 & a13 \\ a21 & a22 & a23 \end{matrix} = \begin{matrix} a11 & a12 & a13 \\ a21 & a22 & a23 \end{matrix}$$

Matrix Multiplycation Properties

(AB)C = A(BC)	Associative matrix
IA = AI=A	Identity matrix
A(B+C)=AB+AC	Left distributive property
(A+B)C=AC+BC	Right distributive property
C(AB)=(CA)B=ACB	Scalar associative property (c is scalar)

*Notice no commutative property

Inverse of a Non Singular Matrix

N*N matrix A is called nonsingular or invertible if there exits an N*N matrix B such that:

AB=BA=I

If there is such B,A is said to be singular and B is said to be the inverse of A

$B=A^{-1}$

$A A^{-1}= A^{-1}A$

If A represents a system of linear equations, then A non-singular -> A has a unique solution

A -> singular Ax=d has more than one solution or no solution at all

Upper Triangle System

A matrix A = $[a_{ij}]$ is called upper triangle if $a_{ij}=0$ when i>j

$a_{11}x_1 + a_{12}x_{2\ +\dots+}\ a_Nx_N=b_1$

$a_{22}x_{2\ +\dots+}\ a2_Nx_N=b_2$

$a_{nn}x_n= b_n$

To solve this system we use back substitution

Xn = bn/ann

$X_{n-1} = b_{n-1} – a_{n-1}$, N*N / $a_{n-1,}$ N-1

…

$X_1=b_1-a_{1N*N}-a_{1N*1}x_{1N-1}\dots a_{12}x_2$ / a_{11}

Example: 3x + 4y + z = 2

3y + 2z = 1

5z = 10

Z = 10/5 = 2

3y + 2(2) = 1 -> y = (1-4)/3 -> -1

3x + 4(-1) + 2 = 2 -> x = (2 – 4(-1) – 2) / 3 = (2 + 4 – 2) / 3 = 4 / 3

Most system of linear equations are not upper triangular. However, we can make them upper triangular using the following method called Gauss Elimination. It uses elementary transformation.

1. Interchange – changing the order of two equations does not affect the solution
2. Scaling – Multiplying an equation by a constant does not affect the solution
3. Replacement – A equation can be replaced by the sum of itself and a non-zero multiple of another equation

   We will use these transformations to convert any system of linear equations to upper triangular and then solve it with back substitution.

   1x+3y+4z = 19

   8x+9y+3z = 35

   x+y+z = 6

   Augmented matrix:

1.	1	3	4	19
2.	8	9	3	35
3	1	1	1	6

Step 1: 2. <- 1. * (-8) + 2.

1.	1	3	4	19
2.	0	−15	−29	−117
3	1	1	1	6

Step 2: 3. <- 1. * -1 + 3.

$$\begin{matrix} 1. & 1 & 3 & 4 & 19 \\ 2. & 0 & -15 & -29 & -117 \\ 3 & 0 & -2 & -3 & -13 \end{matrix}$$

Step 3: 3. + 2. * (2/-15) -> 3

$$\begin{matrix} 1. & 1 & 3 & 4 & 19 \\ 2. & 0 & -15 & -29 & -117 \\ 3 & 0 & 0 & .86666 & 2.6 \end{matrix}$$

Back substitution:
Z = 2.6/.866 = 3
Y = (-117 + 29*3) / -15 = 2
X = 19 – 4*3 – 3*2 = 1

Implement Gauss Elimination in Matlab

```
Function x =gauss(A,B)
%Input       –A is N*N nonsingular matrix
%            - B is a N*1 matrix
%Output      - X is a N*1 matrxi with the solution Ax=B
%Get dimensions of A
[N N] = size(A);
%Initialize x w/ zeros
X = zeros(N,1)
%Obtain augmented matrix
Aug = [A B]
%Gauss elimination
%for all rows
For p = 1: N
    %Choose the pivot, we ignore the case when pivot becomes zero
    Piv = Aug(p,p)
    %divide pivotal row by pivot
    For k=p:N+1
        Aug(p,k) = Aug(p,k)/piv
    End
End
```

6/23

```
%You can also use the short notation
%Aug(p,p:N+1) = Aug(p,p:N+1)/piv
%Make zeros the other elements in the pivotal column
For k=p+1:N %for all remaining rows after the pivot
    M=aug(k,p) %rows after pivot
    For i=p:N+1 %element at row k in pivot column
        Aug(k,i) = aug(k,i) – m*aug(p,i)
    End
End
```

```
%at this point we have an upper triangle matrix
%now we solve with back substitution
For k=N:1:-1 %Goes backward
        %Accumulate all elements in upper triangle
        Sum = 0;
        For i=k+1:N
                Sum = sum + aug(k,i)*x(i,1)
        End
        X(k,1)=aug(k,N+1)
End
```

____

End gauss.m

What will happen if pivot == 0?
-If pivot ==0, we switch rows to prevent division by zero
-To reduce computing errors, you can choose as the next pivot the largest number in the pivotal column, under the current pivotal row.

```
1 7 5 1 8 ->    1 7 5 1 8
  2 5 1 7         9 3 1 2
  9 3 1 2         2 5 1 7
    4 2 1           4 2 1
      1 0             1 0
```

Triangular factorization

In many cases you have multiple sets of equations that have the same A but different B->B1, B2, B3…

Ax1 = B1
Ax2 = B2
Ax3 = B3
…
Axn = Bn

If this is the case, instead of using gauss elimination for every system, we use LR factorization that reduces the amount of work.

A matrix A has a triangular factorization if it can be expressed as the product of a lower-triangular matrix L and an upper triangular matrix U s,t

A = L*U

$$\underbrace{\begin{bmatrix} a11 & \cdots & a1n \\ \vdots & \ddots & \vdots \\ an1 & \cdots & ann \end{bmatrix}}_{A} = \underbrace{\begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ mn1 & \cdots & 1 \end{bmatrix}}_{L} \underbrace{\begin{bmatrix} u11 & \cdots & u1n \\ \vdots & \ddots & \vdots \\ 0 & \cdots & unn \end{bmatrix}}_{U}$$

-Assume that there is a linear system Ax=B such that has a triangular factorization. Then

1. Ax = B
2. A = LU then substituting 2. In 1.
3. LUx = B

Then we can define Y=Ux, then substituting 4. In 3.

4. LY = B -> $\begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ mij & \cdots & 1 \end{bmatrix} \begin{matrix} y1 \\ y2 \\ yn \end{matrix} = \begin{matrix} b1 \\ b2 \\ bn \end{matrix}$

So we can solve x by first solving Y in LY=B using forward substitution and then solve X in Ux=Y using backwards substitution.

$$\begin{bmatrix} u11 & \cdots & u1n \\ \vdots & \ddots & \vdots \\ 0 & \cdots & unn \end{bmatrix} \begin{matrix} x1 \\ xn \end{matrix} = \begin{matrix} y1 \\ yn \end{matrix}$$

Using the LU matrix to solve multiple set of equations that share the same A simplifies work, however, we need to find L and U such that A=L*U

6/24

LU Factorization

We want to solve the following 2 systems of eq.

6x1+x2-4x3 = 3

5x1+3x2-2x3 = 21

1x1+4x2+3x3 = 10

And

6x1 + x2 -4x3 = 3

5x1 + 3x2 + 2x3 = 10

1x1 – 4x2 + 3x3 = 0

$\begin{matrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{matrix} * \begin{matrix} 6 & 1 & -4 \\ 5 & 3 & 2 \\ 1 & -4 & 3 \end{matrix}$ => to get A21 => $\begin{matrix} 1 & 0 & 0 \\ 5/6 & 1 & 0 \\ 0 & 0 & 1 \end{matrix} * \begin{matrix} 6 & 1 & -4 \\ 0 & 3-(5/6) & 2+(20/6) \\ 1 & -4 & 3 \end{matrix}$

We want I→I and A→U

$\begin{matrix} 1 & 0 & 0 \\ 5/6 & 1 & 0 \\ 1/6 & 0 & 1 \end{matrix} \quad \begin{matrix} 6 & 1 & -4 \\ 0 & 13/6 & 16/3 \\ 1 & -4-(1/6) & 3+(3/2) \end{matrix}$

$\begin{matrix} 1 & 0 & 0 \\ 5/6 & 1 & 0 \\ 1/6 & -25/13 & 1 \end{matrix} * \begin{matrix} 6 & 1 & -4 \\ 0 & 13/6 & 13/6 \\ 0 & 0 & 1086/78 \end{matrix}$ → so we have L and U Ax=b, A=LU

LUx=b, Ux=Y and LY=B

First solve: Ly=B

$\begin{matrix} 1 & 0 & 0 \\ 5/6 & 1 & 0 \\ 1/6 & -25/13 & 1 \end{matrix} \begin{matrix} y1 \\ y2 \\ y3 \end{matrix} = \begin{matrix} 3 \\ 21 \\ 10 \end{matrix}$ → y1= 3, y2 = 18.5, y3 = 45.0769

Now we solve Ux = Y

$$\begin{bmatrix} 6 & 1 & -4 \\ 0 & 13/6 & 13/6 \\ 0 & 0 & 1086/78 \end{bmatrix} \begin{bmatrix} x1 \\ x2 \\ x3 \end{bmatrix} = \begin{bmatrix} 3 \\ 18.5 \\ 45.0764 \end{bmatrix}$$

X3 = 45.0764 (98/1086) = 3.2376
X2 = [18.5 – ((32/6) * 3.2376)](6/13) = .5695
X1 = [3 – (1*.5695) + (-4 * 3.2376)](1/6) = 2.5635

Use also LU factorization to solve:

6x1 + 1x2 + 4x3 = 3
5x1 + 3x2 + 2x3 = 10
1x1 – 4x1 + 3x3 = 0

$$L = \begin{bmatrix} 1 & 0 & 0 \\ 5/6 & 1 & 0 \\ 1/6 & -35/13 & 1 \end{bmatrix}, \quad U = \begin{bmatrix} 6 & 1 & -4 \\ 0 & 13/6 & 32/6 \\ 0 & 0 & 1086/78 \end{bmatrix}$$

Ax=b, LUx=b, $b = \begin{bmatrix} 3 \\ 10 \\ 0 \end{bmatrix}$

Solve first for Y and then solve Ux=Y using forward substitution
Solve LY=b using forward substitution

Factor A into LU = $O(n^3)$
Gauss Elimination = $O(n^3)$
Back Substitution = $O(n^2)$
So for M systems of linear equations if we use only Gauss elimination

$MO(n^3) \rightarrow O(MN^3)$

If we use LU factorization once and the M forward and M backward substitution

$O(N^3) + M(O(N^3)) + O(N^2)$
LU forward backward

$O(N^3) + 2MO(N^2)$
$O(N^3) + MO(MN^2)$
If M=N

1. Gauss elimination for $O(NN^3) = O(N^4)$
2. LU factorization + backward/forward substitution
   $O(N^3) + O(N^3) \sim O(N^3)$ ← this is faster