Addendum

Vector Review



Coordinate Systems

Right handed coordinate system





Vector Arithmetic

$$\mathbf{a} = \begin{bmatrix} a_x & a_y & a_z \end{bmatrix}$$
$$\mathbf{b} = \begin{bmatrix} b_x & b_y & b_z \end{bmatrix}$$
$$\mathbf{a} + \mathbf{b} = \begin{bmatrix} a_x + b_x & a_y + b_y & a_z + b_z \end{bmatrix}$$
$$\mathbf{a} - \mathbf{b} = \begin{bmatrix} a_x - b_x & a_y - b_y & a_z - b_z \end{bmatrix}$$
$$-\mathbf{a} = \begin{bmatrix} -a_x & -a_y & -a_z \end{bmatrix}$$
$$s\mathbf{a} = \begin{bmatrix} sa_x & sa_y & sa_z \end{bmatrix}$$



Vector Magnitude

The magnitude (length) of a vector is:

$$\left|\mathbf{v}\right| = \sqrt{v_x^2 + v_y^2 + v_z^2}$$

A vector with length=1.0 is called a *unit vector* We can also *normalize* a vector to make it a unit vector:





Dot Product

$$\mathbf{a} \cdot \mathbf{b} = \sum a_i b_i$$
$$\mathbf{a} \cdot \mathbf{b} = a_x b_x + a_y b_y + a_z b_z$$

$\mathbf{a} \cdot \mathbf{b} = |\mathbf{a}| |\mathbf{b}| \cos \theta$



Dot Product

$$\mathbf{a} \cdot \mathbf{b} = \sum a_i b_i$$

$$\mathbf{a} \cdot \mathbf{b} = a_x b_x + a_y b_y + a_z b_z$$

$$\mathbf{a} \cdot \mathbf{b} = |\mathbf{a}| |\mathbf{b}| \cos \theta$$

$$\mathbf{a} \cdot \mathbf{b} = \mathbf{a}^T \mathbf{b}$$

$$\mathbf{a} \cdot \mathbf{b} = \begin{bmatrix} a_x & a_y & a_z \end{bmatrix} \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix}$$



Example: Angle Between Vectors

How do you find the angle θ between vectors **a** and **b**?





Example: Angle Between Vectors





Dot Products with General Vectors

The dot product is a scalar value that tells us something about the relationship between two vectors

- If $\mathbf{a} \cdot \mathbf{b} > 0$ then $\theta < 90^{\circ}$
- If $\mathbf{a} \cdot \mathbf{b} < 0$ then $\theta > 90^{\circ}$
- If $\mathbf{a} \cdot \mathbf{b} = 0$ then $\theta = 90^{\circ}$ (or one or more of the vectors is degenerate (0,0,0))



Dot Products with One Unit Vector

If $|\mathbf{u}|=1.0$ then $\mathbf{a} \cdot \mathbf{u}$ is the length of the *projection* of \mathbf{a} onto \mathbf{u}





Example: Distance to Plane

A plane is described by a point **p** on the plane and a unit normal **n**. Find the distance from point **x** to the plane





Example: Distance to Plane

The distance is the length of the projection of **x**-**p** onto **n**:





Dot Products with Unit Vectors





Cross Product

$$\mathbf{a} \times \mathbf{b} = \begin{vmatrix} i & j & k \\ a_x & a_y & a_z \\ b_x & b_y & b_z \end{vmatrix}$$

-

$$\mathbf{a} \times \mathbf{b} = \begin{bmatrix} a_y b_z - a_z b_y & a_z b_x - a_x b_z & a_x b_y - a_y b_x \end{bmatrix}$$



Properties of the Cross Product

$\mathbf{a} \times \mathbf{b}$ is a *vector* perpendicular to both **a** and **b**, in the direction defined by the right hand rule

$$|\mathbf{a} \times \mathbf{b}| = |\mathbf{a}| |\mathbf{b}| \sin \theta$$

 $|\mathbf{a} \times \mathbf{b}| = \text{area of parallelogram } \mathbf{ab}$
 $|\mathbf{a} \times \mathbf{b}| = 0$ if \mathbf{a} and \mathbf{b} are parallel



Example: Normal of a Triangle

Find the unit length normal of the triangle defined by 3D points **a**, **b**, and **c**





Example: Normal of a Triangle





Example: Area of a Triangle

Find the area of the triangle defined by 3D points **a**, **b**, and **c**





Example: Area of a Triangle

$$area = \frac{1}{2} |(\mathbf{b} - \mathbf{a}) \times (\mathbf{c} - \mathbf{a})|$$





Example: Alignment to Target

An object is at position \mathbf{p} with a unit length heading of \mathbf{h} . We want to rotate it so that the heading is facing some target \mathbf{t} . Find a unit axis \mathbf{a} and an angle θ to rotate around.





Example: Alignment to Target





Addendum

Cramer's Rule



This is a great method.

Cramer's Rule is a neat way to evaluate systems and if you put the work in now you'll do fine. It can be used for any size (2 by 2, 3 by 3 or even larger) system.

It is easy to memorize and fast.

I'm going to show you where Cramer's Rule comes from; but first, some definitions



Definitions

Determinant – a square array

2nd Order Determinant – a 2 by 2 array

3rd Order Determinant – a 3 by 3 array

Elements – The things in the array



What does a determinant look like?

A 2nd order determinant looks like this



Diagonal down right – diagonal down left ae-bd



Examples

Evaluate



Why is this useful for systems?

Lets work through an elimination example using all variables; then we can see how the determinant will be useful in solving.







If you apply the same process but eliminate x



So, what does Cramer's Rule say?





Examples

Solve using Cramer's Rule

1. 6x + 7y = -92. x - y = 55x + 4b = -1

$$2x - b = 10$$

WRIGHT STATE UNIVERSITY

Cramer's Rule in General

Cramer's Rule: For the system of equations Ax = y, where A is an *nxn* nonsingular matrix, the solution for the *i*th endogenous variable, x_i , is

$$\mathbf{x}_{i} = |\mathbf{A}_{i}|/|\mathbf{A}|$$

where the matrix A_i represents a matrix that is identical to the matrix A but for the replacement of the *i*th column with the *nx1* vector **y**.



Linear Systems in Matrix Form



Solution of Linear Systems

Each side of the equation

$$A \cdot \vec{x} = \vec{b} \quad (2)$$

Can be multiplied by A⁻¹:

$$A^{-1}A \cdot \vec{x} = A^{-1}\vec{b}$$

Due to the definition of A⁻¹: $A^{-1}A \cdot \vec{x} = I \cdot \vec{x} = \vec{x}$

Therefore the solution of (2) is:

$$x = A^{-1}\vec{b}$$



Consistency (Solvability)

- A⁻¹ does not exist for every A
- The linear system of equations A^{*}x=b has a solution, or said to be consistent IFF

Rank{A}=Rank{A|b}

• A system is inconsistent when

Rank{A}<Rank{A|b}

Rank{A} is the maximum number of linearly independent columns or rows of A. Rank can be found by using ERO (Elementary Row Oparations) or ECO (Elementary column operations).



Elementary row and column operations

The following operations applied to the augmented matrix [A|b], yield an equivalent linear system

- Interchanges: The order of two rows/columns can be changed
- Scaling: Multiplying a row/column by a nonzero constant
- Sum: The row can be replaced by the sum of that row and a nonzero multiple of any other row.

One can use ERO and ECO to find the Rank as follows: ERO \Rightarrow minimum # of rows with at least one nonzero entry or

ECO \Rightarrow minimum # of columns with at least one nonzero entry



An inconsistent example: Geometric interpretation



Uniqueness of solutions

The system has a unique solution IFF

Rank{A}=Rank{A|b}=n

n is the order of the system

• Such systems are called full-rank systems



Full-rank systems

 If Rank{A}=n $Det{A} \neq 0 \Rightarrow A^{-1} exists \Rightarrow Unique solution$ 1 equation 1 $\begin{vmatrix} 1 & 2 \\ 1 & -1 \end{vmatrix} \begin{vmatrix} x_1 \\ x_2 \end{vmatrix} = \begin{vmatrix} 4 \\ 2 \end{vmatrix}$ 0 equation 2 -1 -1 0 1 2 3 4 5 X₁



Rank deficient matrices

If Rank{A}=m<n

 $Det{A} = 0 \Rightarrow A$ is singular so not invertible

infinite number of solutions (n-m free variables)

under-determined system





Ill-conditioned system of equations

• A small deviation in the entries of A matrix, causes a large deviation in the solution.

$$\begin{bmatrix} 1 & 2 \\ 0.48 & 0.99 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 3 \\ 1.47 \end{bmatrix} \qquad \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$
$$\begin{bmatrix} 1 & 2 \\ 0.49 & 0.99 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 3 \\ 1.47 \end{bmatrix} \qquad \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 3 \\ 0 \end{bmatrix}$$



Ill-conditioned continued.....

 A linear system of equations is said to be "ill-conditioned" if the coefficient matrix tends to be singular





Gaussian Elimination

- By using ERO, matrix A is transformed into an upper triangular matrix (all elements below diagonal 0)
- Back substitution is used to solve the upper-triangular system





Pivotal Element



The first coefficient of the first row (pivot) is used to zero out first coefficients of other rows. In terms of numerical stability it is usually best to use the largest element in the column, i.e. $\max_{k \le j \le n} |a_{jk}^{(k)}|$



First step of elimination

$$\begin{split} m_{2,1} &= a_{21}^{(1)} / a_{11}^{(1)} \\ m_{3,1} &= a_{31}^{(1)} / a_{11}^{(1)} \\ \vdots \\ m_{n,1} &= a_{n1}^{(1)} / a_{11}^{(1)} \\ \end{bmatrix} \begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2n}^{(2)} \\ 0 & a_{32}^{(2)} & a_{33}^{(2)} & \cdots & a_{3n}^{(2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & a_{n2}^{(2)} & a_{n3}^{(2)} & \cdots & a_{nn}^{(2)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1^{(1)} \\ b_2^{(2)} \\ b_3^{(2)} \\ \vdots \\ b_n^{(2)} \end{bmatrix} \end{split}$$

First row, multiplied by appropriate factor is subtracted from other rows.



Second step of elimination





Second step of elimination

$$m_{3,2} = a_{32}^{(2)} / a_{22}^{(2)} \begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2n}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & \cdots & a_{3n}^{(3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & a_{n3}^{(3)} & \cdots & a_{nn}^{(3)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1^{(1)} \\ b_2^{(2)} \\ b_3^{(3)} \\ \vdots \\ b_n^{(3)} \end{bmatrix}$$



Gaussion elimination algorithm

$$m_{r,p} = a_{rp}^{(p)} / a_{pp}^{(p)}$$

$$a_{rp}^{p} = 0$$

$$m_{r,p} = a_{rp}^{(p)} / a_{pp}^{(p)}$$

$$a_{rp}^{(p)} = 0$$

$$b_{r}^{(p+1)} = b_{r}^{(p)} - m_{r,p} \cdot b_{p}^{(p)}$$

$$a_{rc}^{(p+1)} = a_{rc}^{(p)} - m_{r,p} \cdot a_{pc}^{(p)}$$



Back substitution algorithm





Back substitution algorithm

The answer is obtained as following:





LU decomposition

If we now define a matrix R

$$R = \begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2n}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & \cdots & a_{3n}^{(3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & a_{n-1n-1}^{(n-1)} & a_{n-1n}^{(n-1)} \\ 0 & 0 & 0 & 0 & 0 & a_{nn}^{(n)} \end{bmatrix}$$

and another matrix $L = (l_{ij})$ with

$$l_{ik} = \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}$$

then we get: $A = L \cdot R$



LU decomposition

Note that the matrices R and L are upper and lower triangular matrices. Hence, we can solve the linear equation system in two steps:

$$L c = b$$
$$R x = c$$

Solving these two systems can be achieved similar to the using a similar algorithm we used for back substitution.

Also note that solving the system of linear equations for different solutions *b* does not require a repetition of the Gaussian elimination algorithm.



Cholesky decomposition

If the matrix *A* is symmetric, a solution for the LU decomposition is even easier since there exist the following decomposition for those matrices:

$$A = C^T C$$

where

$$C = \begin{pmatrix} c_{11} & \cdots & \cdots & c_{1n} \\ 0 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & c_{nn} \end{pmatrix}$$



Cholesky decomposition

How can we compute *C*? Let us look at the equation:

$$\begin{pmatrix} c_{11} & 0 & \cdots & 0 \\ c_{12} & \ddots & \ddots & \vdots \\ \vdots & & \ddots & 0 \\ c_{1n} & c_{2n} & \cdots & c_{nn} \end{pmatrix} \begin{pmatrix} c_{11} & \cdots & \cdots & c_{1n} \\ 0 & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & c_{nn} \end{pmatrix} = \begin{pmatrix} a_{11} & \cdots & \cdots & a_{1n} \\ \vdots & & & \vdots \\ a_{n1} & \cdots & \cdots & a_{nn} \end{pmatrix}$$

Hence:

$$a_{kk} = \sum_{j=1}^{k} c_{jk} c_{jk} \Longrightarrow c_{kk} = \sqrt{a_{kk} - \sum_{j=1}^{k-1} c_{jk}^2}$$
$$a_{kl} = \sum_{j=1}^{k} c_{jk} c_{jl} \Longrightarrow c_{kl} = \frac{a_{kl} - \sum_{j=1}^{k-1} c_{jk} c_{jl}}{c_{kk}} \quad \text{for} \quad k < l \le n$$

