E3101: Introduction to Linear Algebra

Prof. Marc Spiegelman: 211 Mudd
Office Hours 1:45-2:45 Tu/Thurs

Course Website: www.ldeo.columbia.edu/~mspieg/e3101

Lecture 01: Introduction to Linear Algebra

- Introduction and Rationale
- Course Logistics
- Scalars and Vectors: properties and operations

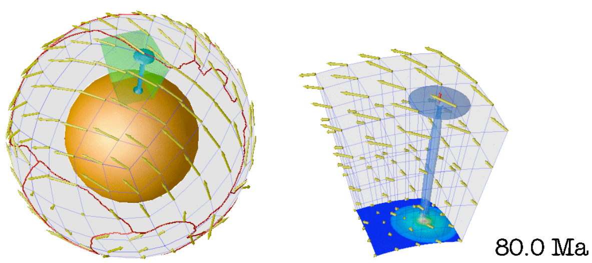
Why Linear Algebra?

It's Easy!
(at least the mechanics...)

It's Deep:
fundamental concepts (bases, vector spaces, orthogonality, projections, eigen values/vectors...) are essential for advanced mathematics (PDE's, Functional Analysis, Numerical analysis...).
But these ideas are much easier in Linear Algebra!

It's Practical: Central to many applications
Scientific Computing
Google Page Rank
Data Analysis
Computer Graphics
Network analysis (social/genomic)

An example application: Numerical models of the Hawaiin Hotspot (www.geodynamics.org)



Purpose of this course:

To teach all three aspects of linear algebra and develop a fundamental *fluency* in mechanics, theory and applications

Course Logistics:

2 Lectures per week
1 Homework per week (usually due tuesdays, 20% of grade)
Text: Strang - Introduction to Linear Algebra, 3rd Edition

2 Midterms (20% per quiz)
1 Final (40% of grade)

Want to do well? Do the work...

TA's: Matt Davis (primary) office hours TBD
My office hours Tu/Thurs 1:45-2:45 211 Mudd

All spelled out on web site
www.ldeo.columbia.edu/~mspieg/e3101

Lecture 01: meet the vectors

Scalars and Vectors
Basic properties and Notation
Basic operations
Vector addition
Scalar Multiplication
Linear Combinations of Vectors
The *Geometry* of Linear combinations
More operations
the dot product and vector transpose
Introduction to Matlab

Scalars: just numbers

Examples:

Notation:

Basic Operations: (4th grade arithmetic)

Commutative:

Associative:

Distributive:

Vectors: ordered sets of scalars

Examples & Notation:

Vectors: Basic Operations

Scalar Multiplication: create a new vector by multiplying by a scalar

Vectors: Basic Operations

Vector Addition: create a new vector by adding two (or more) vectors

Vectors: Basic Operations - putting it together

Linear Combinations of vectors: create new vectors by both scalar multiplication and vector addition

Examples:

Rules of Vectors Operations:

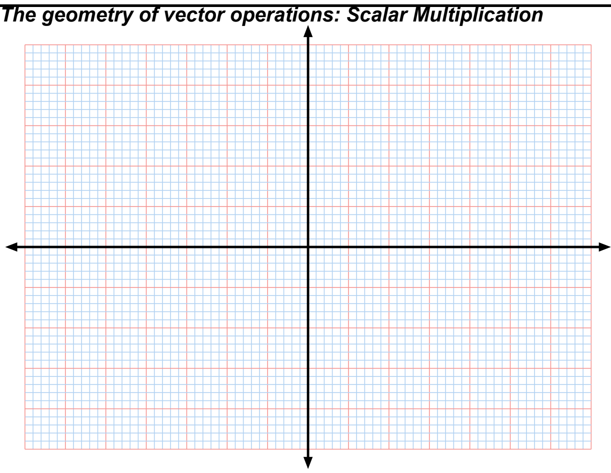
Follow from rules of scalar arithmetic

Commutative:

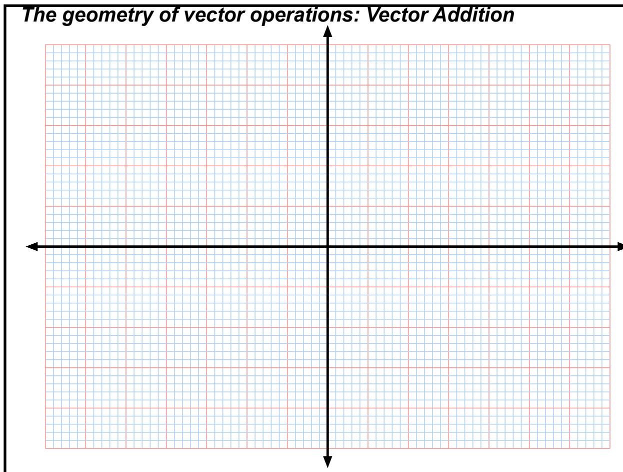
Associative:

Distributive (by Scalar Multiplication):

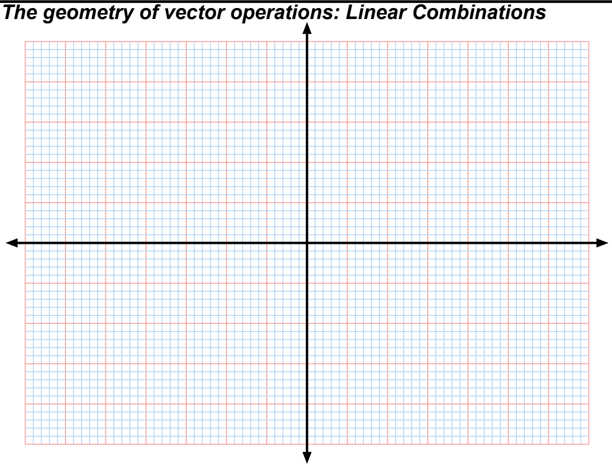
The geometry of vector operations: Scalar Multiplication



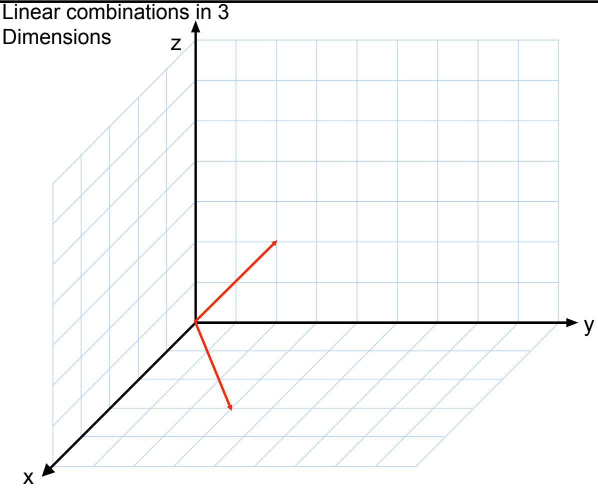
The geometry of vector operations: Vector Addition



The geometry of vector operations: Linear Combinations



Linear combinations in 3 Dimensions



The geometry of linear combinations

Linear combinations of vectors are geometric objects

All Linear combinations of 1 vector
form a

Linear combinations of 2 vectors
form a

Linear combinations of 3 vectors
form a

n vectors ?

Other Operations: The Dot Product

The Dot Product: maps 2 vectors to a scalar

Definition:

Examples:

Other Operations:

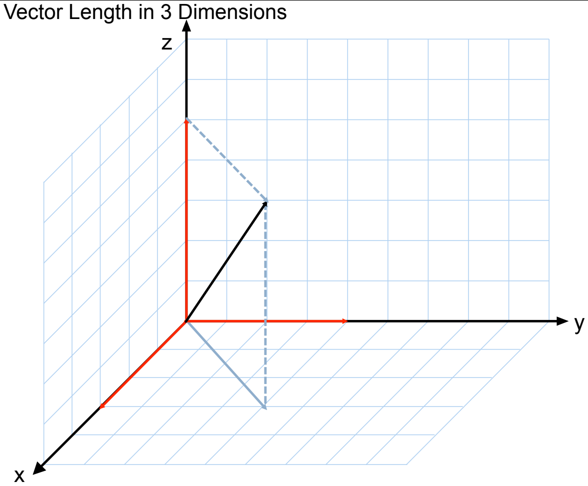
Dot products and the length of a vector

Consider the dot product of a vector with itself

Examples:

Definition: Vector Length (vector norm)

Vector Length in 3 Dimensions



Unit Vectors:

Definition: n-dimensional vectors of length $||\underline{q}||=1$

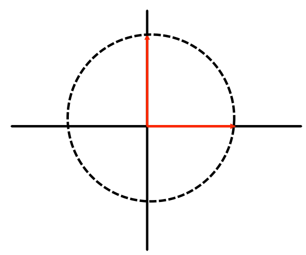
Any vector can be transformed into a unit vector pointing in the same direction

Examples:

Dot product of Unit vectors:

Theorem: the dot product is invariant to rotation

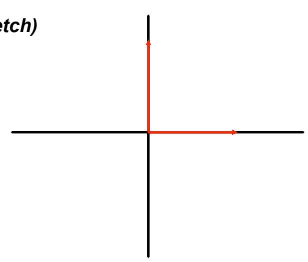
Proof: (2-D sketch)



Dot product of Unit vectors:

Theorem: the dot product of two unit vectors is cosine of the angle between them (in all dimensions)

Proof: (2-D sketch)



Dot product of Unit vectors in 3-D: Proof by Basketball

Dot product of General Vectors:

Theorem: the dot product of any two vectors v and w

Schwartz Inequality:

Orthogonal Vectors:

Dot product and the Vector transpose:

Definition: Transpose operator- turns column vectors into row vectors (and vice versa)

Definition of dot product using transpose:

Linear Algebra and Matlab®:

Basic Syntax:

```
% define a scalar
a = 2
c = 3;

% define vectors

x = [ 1 ; 2 ; 3 ]  % column vector
x = [ 1 2 3 ]      % row vector
x = [ 1 2 3 ]'     % column vector using transpose

y = [ 1 1 -1]'
```

Linear Algebra and Matlab®:

Basic Operations:

```
% scalar multiplication
z = 3*x
z = a*x

% vector addition and linear combinations

z = x+y
z = 2*x + 3*y
z = a*x +c*y

% dot product

c = dot(x,y) % use dot function
c = x'*y     % use transpose x^Ty

% vector length & unit vectors
normV = sqrt(x'*x);
normV = norm(x);
q = x/norm(x);  % check norm(q)=1

% angle between vectors

theta = acos(x'*y/norm(x)/norm(y))
```