Matrix A, B: A\*B 不一定等于 B\*A, matrix乘法direction matters

Identity Matrix: I\*A = A: 1 in diagonal position.

Why it works? Row information from indentity matrix 的row and column information from A的column, 比如[1,0] 乘以 or , 0 cancel out every elements 除了first term (a,b) in the column vector, 第二行[0,1] cancel out every elemts 除了second term(c, d)

**Inverse 2\*2:**

Calculate the inverse:

Determinant:

Indentiy matrix: I\*A = A, A\*I = A, 两个都满足的只有当A是square matrix的时候

**Inverse 3\*3: Gauss Jordan elimination (augment the matrix, operation: elementary row operation)**

Perform some operation 在left side and same operation on right side, 当have indentity matrix在left-hand side(变成indentiy matrix的形式 叫做reduced row echelon form ), right-hand side就是原来的invers

Row3 = row3 – row1

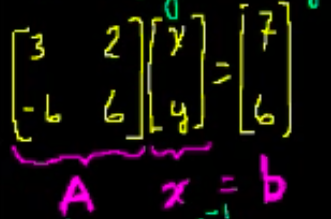
Swap row3 and row2

Row3 = row3 – 2\*row2

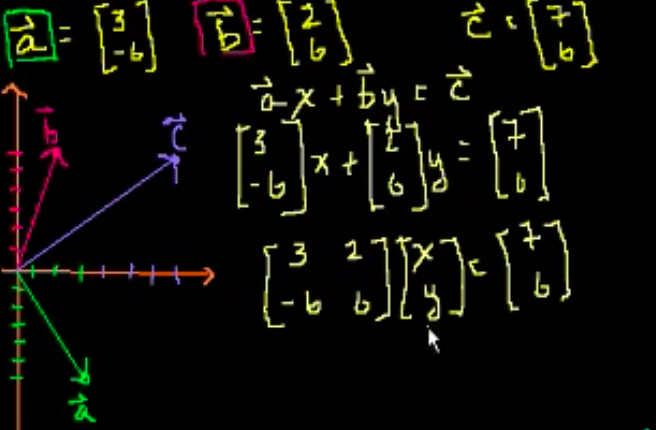
Row1 = row1 – row3

Hint why this work: 当对左面matrix进行操作可以想成乘以多个matrices，so we multiply matrix 得到indentity matrix, 乘以的多个matrices就是, 而我们知道identity matrix乘以任何matrix就是matrix itself

Solve system of equations (2\*2):



Ax = b 🡪 🡪 🡪

Matrices to solve vector combination: 可以想成matrix multiplication problem把两个vector合成一个vector

比如vector a = , b = , 得到 c =

A = \* =

, , 需要1个a,两个b

Singular matrices: 没有inverse的matrix

Prove: 比如

得到 => 如果 两条线平行, 不会相交没有解



如果从vector角度考虑，如下图 是重合的 if , 无法由, , 构成



Solve system of equations (3\*3):

通过row operation 变成

Vectors and Spans:

Set Colinear vectors: {c 比如vector在一条线上(slope),

Linear Combination: ; We can fill Any point in with combination of vector a and b: we can write , we can represent any vector in with some linear combination of a and b where a and b cannot be collinear (a,b 不能共线, 换种思维考虑: 如果共线了,组成的matrix没有inverse A\*c = b, A没有inverse). = (c)

比如 unit vector , 可以构建任何vector in by using these unit vectors

: The space of all of the combination of vectors

**Linearly Dependent set**: some vector in the set can be represented by some combinations of other vectors in the set, 比如是linearly dependent, 再比如 是linearly dependent,因为其中一个可以由另外两个构成构成

:

**V is Subspace of**  ( me vector from ) 必须满足：

1. V contains 0 vector
2. If in V then any scaler c: also in V (closure under scaler multiplication)
3. If in V and in V, + also in V (closure under addition)

同样如果满足这三个条件的也是subspace

e.g. v = {0} = : v 只有vector 0, v is subspace of

1. 满足条件1, vector 0 在v中

2. 满足条件2: c

3. 满足条件3:

e.g. S =: v is not subspace of

1. 满足条件1, vector 0 在v中

2. 不满足条件2: -1, -1\*a 为负数

3. 满足条件3: a+c是正数 given a>=0 and c>=0

e.g V = . is valid subspace of

1. 满足条件1:

2. 满足条件2: ; then ; can be arbitrary constant, 因为span是all linear combination 所以新的也在span当中

3. 满足条件3： ; , then , it also in span

e.g V = . is valid subspace of

1. 满足条件1:

2. 满足条件2: 就是span itself (combination of vector)

3. 满足条件3： ; 在span当中

**Span() of any vector is valid subspace,**

**Basis S= {}: 1. Span(all those linearly independent 2. = 0 when**

Basis(minimum set of vectors that spans the subspace): 如果用any vector in S 可以construct any vector in subspace V

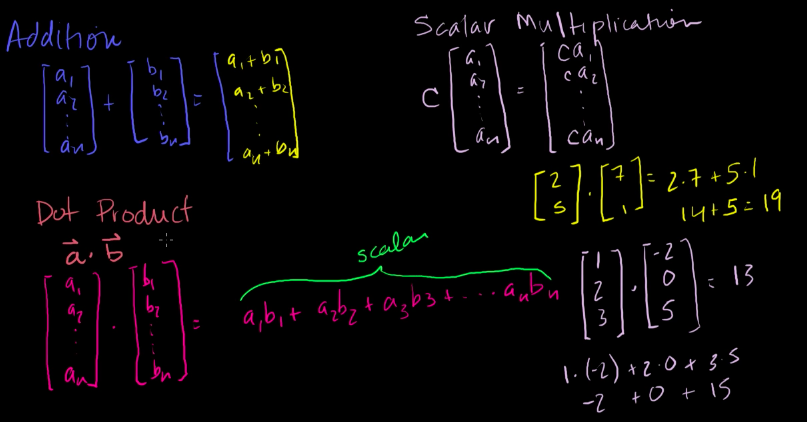
e.g. T = {, }, the span of T is still going to be subspace V but T is linearly dependent -> T is not basis for V ()

Basis: 比如, 需要两个non-redundant vector

Standard Basis for

Advantage of Basis: represent any vector in subspace by some unique combination of vectors in basis 比如Basis {} 是unique的

Vectors Dot Product



Length :

Communicative:

Distributive: ( =

Associative:

**Cauchy Schwarz Inequality**: If , only if two vector colinear, one vector 是另一个vector乘以的倍数 ()

Prove: suppose is non-zero vector

此时设

Take square root

When

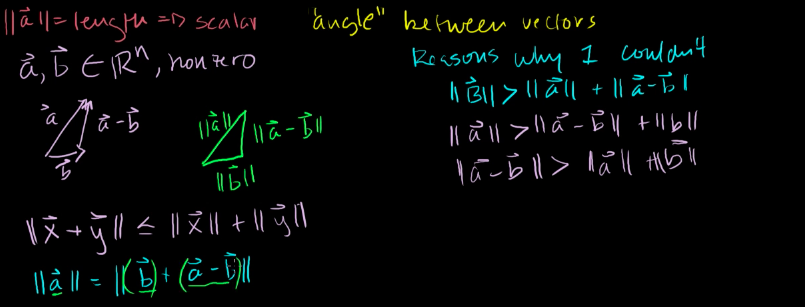
**Triangle Inequality**:

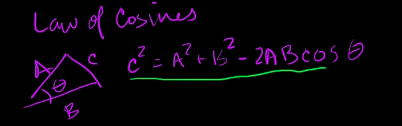
根据cauchy Schwarz inequality: dot product 可以是负数

当

doesn’t need to only 2-dimensional, 可以是n dimension

**Triangle Angle between vectors**:



Law of Cosine

Left-hand side

Left-hand side = right-hand side

If

Perpendicular

但是如果dot product = 0 不意味着垂直, 比如

但是当a, b都是nonzero vector, dot product 意味着垂直(perpendicular)

=> orthogonal, zero vector is orthogonal to everything; perpendicular is orthogonal, 但是othogonal 不是perpendicular

Dot Product:

Cross Product: only defined in ， 得到vector

Cross product 乘积是 orthogonal to and