CSI 436/536
Introduction to Machine Learning

Review of Linear Algebra (1)

Professor Siwei Lyu
Computer Science
University at Albany, State University of New York

Importance of linear algebra

- Linear algebra
  - provides superior notations (algebra)
    - many topics can be understood better with vector-matrix-space idea (e.g., Fourier)
  - has a consistent intuition (geometry)
    - what is true for low dimensional space is usually also true for high dimensional space
      - not usually the case in general
  - computes efficiently (numerical algorithms)
    - Almost all numerical computation requires support of linear algebra
    - LAPACK is the backbone of Matlab, NumPy, R

Algebra

- Fourier transform

$DFT(FFT)$:

$$X(k)=\sum_{n=0}^{N-1}x(n)\cdot e^{-j\left(\frac{2\pi}{N}\right)nk}\quad(k=0,1,\ldots,N-1)$$

$IDFT(IFFT)$:

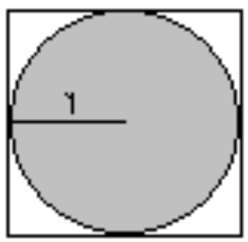
$$x(n)=\frac{1}{N}\sum_{k=0}^{N-1}X(k)\cdot e^{j\left(\frac{2\pi}{N}\right)nk}\quad(n=0,1,\ldots,N-1)$$

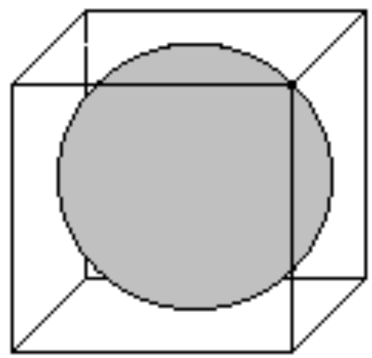
$$C=\begin{bmatrix} c_0 & c_{n-1} & \cdots & c_2 & c_1 \\ c_1 & c_0 & c_{n-1} & & c_2 \\ \vdots & c_1 & c_0 & \ddots & \vdots \\ c_{n-2} & & \ddots & \ddots & c_{n-1} \\ c_{n-1} & c_{n-2} & \cdots & c_1 & c_0 \end{bmatrix}.$$

- FT provide eigenvectors for circulant matrix (discrete case) or LTI operator (continuous case)
- proof of the fundamental convolution theorem (and its continuous version) becomes very easy

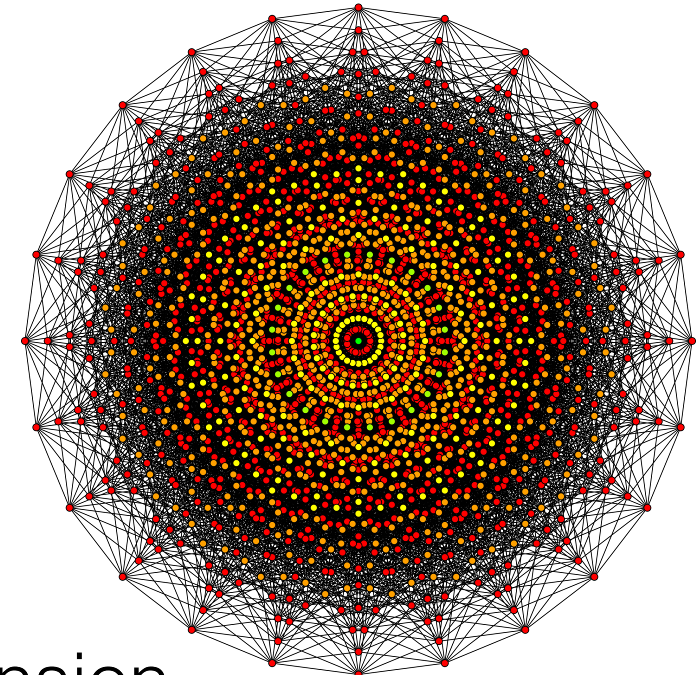
curse of dimensionality

- sphere inscribed in cube

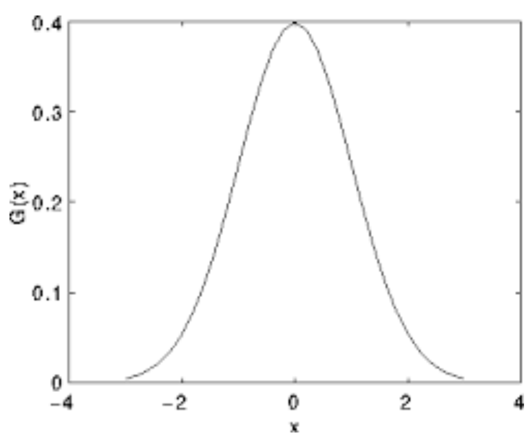


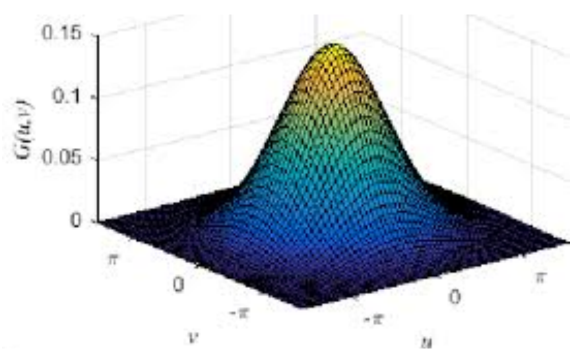


as $d \to \infty$.

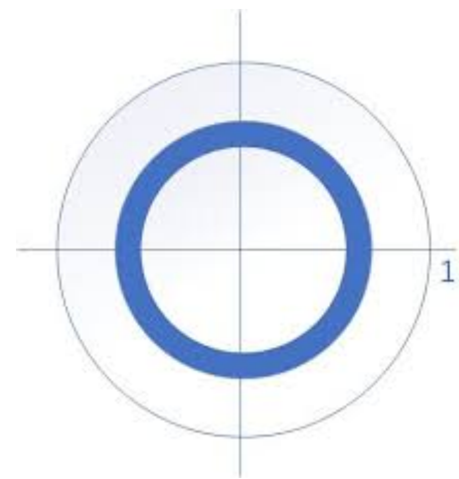


- Gaussian distribution in high dimension





as $d \to \infty$.



Numerical linear algebra

- NLA is behind the majority of numerical procedures for machine learning
  - The majority of ML algorithms are optimization problems [there is a small fraction is about integration instead of optimization]
  - All optimization problems are practically solved as a sequence of quadratic optimization problems
  - All quadratic optimization problems are solved as linear equations or eigenvalues

Overview

- Objects in linear algebra
  - vectors, linear spaces, matrices, linear transforms
- Problems in linear algebra
  - linear equation $\mathbf{Ax} = \mathbf{b}$
  - eigenvalue equation $\mathbf{Ax} = \lambda\mathbf{x}$
- Techniques in linear algebra
  - Matrix factorizations: LU decomposition, eigen decomposition, QR decomposition, etc
- Mostly we will work with
  - Symmetric positive (semi)definite matrices

vectors, space and transforms

- Vectors are list of numbers over a field (real space)
  - Geometrically correspond to points
  - we use column vector by default
  - vector can add/subtract/scale

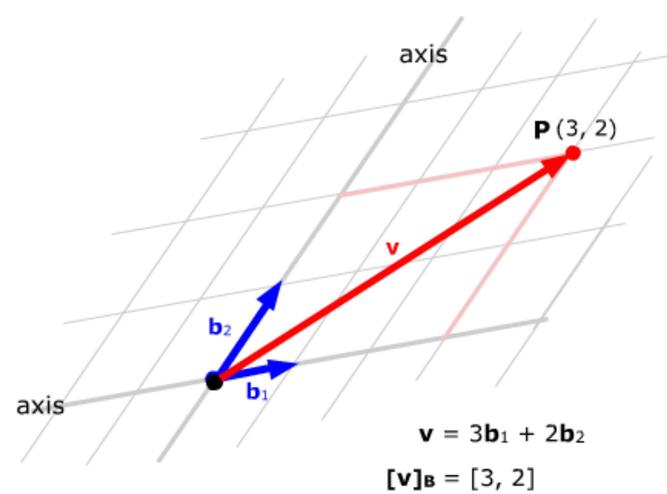
$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

- Linear space is the set of vectors closed under addition and scalar product
  - Subspace is a subset of a space including zero
  - A space can be **spanned** by a set of vectors

    for $\alpha_1, \cdots, \alpha_k \in \mathcal{R}$, $\sum_{i=1}^{k} \alpha_i \mathbf{x}_i$
- A linear transform is a mapping between points in two spaces that keeps linearity

linear independence of vectors

- a set of vectors is linear independent if for any $\vec{u}_i$ is not in $\text{span}(\vec{u}_1, \cdots, \vec{u}_{i-1}, \vec{u}_{i+1}, \cdots, \vec{u}_n)$
- A set of bases of a space V is a set of independent vectors that also span it
  - Canonical basis is the basis that are orthornormal
- Coordinates are coefficients on basis
- the *max* number of vectors that are linearly independent in a space is its dimension
- Dimension of a space may not be the same as the dimension of an individual vector in it



additional structures of space

- distance between two vectors: **metric**
  - metric space
- length of a vector: **norm**
  - norm space
- angle between two vectors: **inner product**
  - inner product space (Hilbert space)
- parallelogram by two vectors: **exterior product**
  - Grassmann space

Vector metrics (distance)

- L2 (Euclidean) metric

$$\|x - y\|_2 = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

- L1 (Manhattan) metric

$$\|x - y\|_1 = \sum_{i=1}^{n} |x_i - y_i|$$

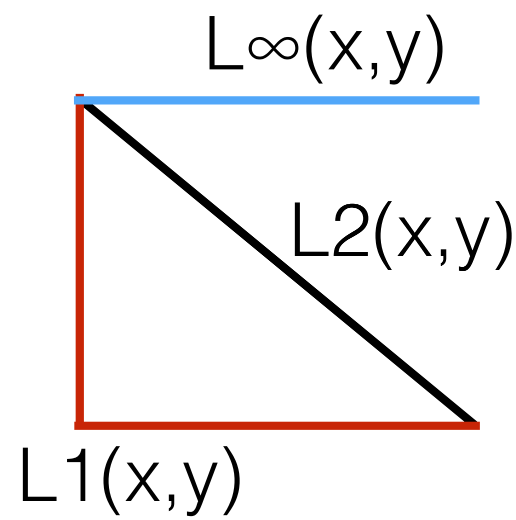
- L∞ (Chebeshev) metric

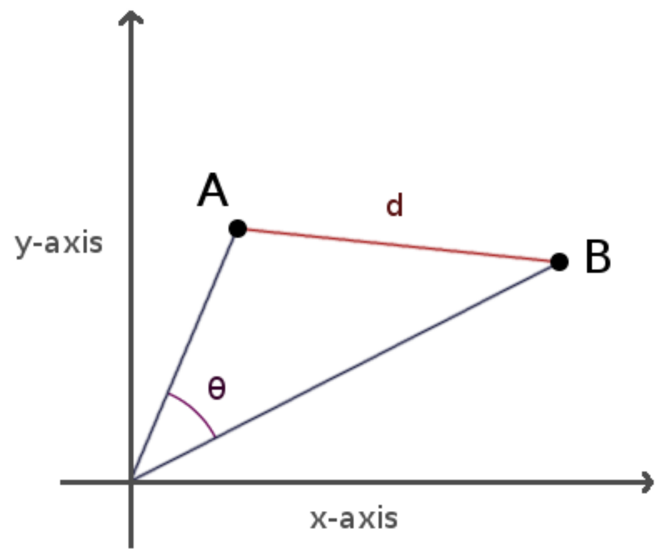
$$\|x - y\|_1 = \max_i |x_i - y_i|$$

- Lp metric (p ≥ 1)

$$\|x - y\|_p = \left(\sum_{i=1}^{n} (x_i - y_i)^p \right)^{1/p}$$

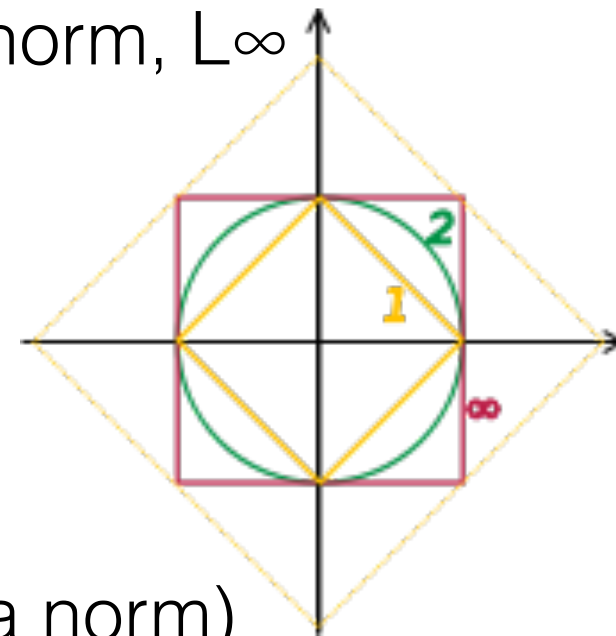
- All metrics satisfy
  - symmetric: d(x,y) = d(y,x)
  - non-negativity: d(x,x) ≥ 0
  - triangle inequality: d(x,y) + d(y,z) ≥ d(x,z)





Norms

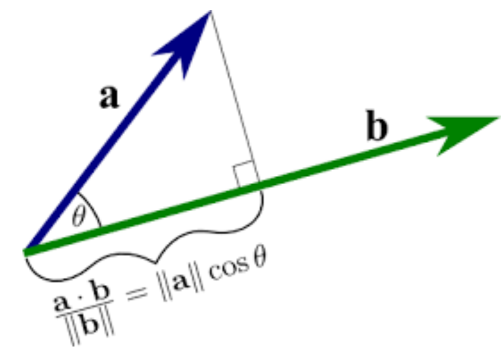
- L2 (Euclidean) norm, L1 (Manhattan) norm, L∞ norm, Lp norm ($p \geq 1$)
- All norms satisfy
  - non-negativity: $|x| \geq 0$
  - triangle inequality: $|x| + |y| \geq |x+y|$
- Normalization to unit vectors (w.r.t. to a norm)
  - Projections onto unit spheres (w.r.t. to a norm)
- Given a norm, we can define metric (distance) as the norm of the different vector
- due norm: $||x||_{p^*} = \max\{s^T x | \; ||s||_p \leq 1\}$, L2 is self-dual, L1 is dual of L∞



Vector products

- inner (scalar) product: (**v**,**v**) →a number $\mathbf{x}^T\mathbf{y} = \sum_{i=1}^{n} x_i y_i$
  - geometrically related with angles
  - Cauchy-Schwartz inequality

$$|\langle \vec{u}, \vec{v} \rangle| \leq \|\vec{u}\| \|\vec{v}\|$$



  - <u,v> = 0 iff u and v orthogonal
- rect (Dirac) product: (**v**,**v**) →a vector

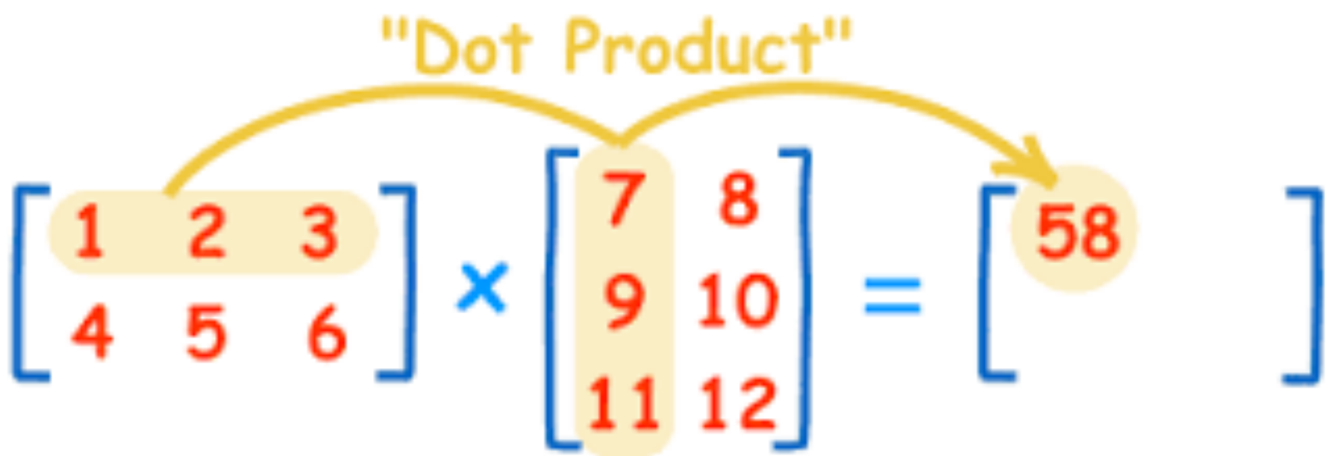
$$\mathbf{x} \odot \mathbf{y} = \begin{pmatrix} x_1 \cdot y_1 \\ x_2 \cdot y_2 \\ \vdots \\ x_n \cdot y_n \end{pmatrix}$$

- exterior (cross/wedge) product: (**v**,**v**) →a vector
- outer (tensor) product: (**v**,**v**) →a matrix (actually a tensor)

$$\mathbf{x}\mathbf{y}^T = \begin{pmatrix} x_1 \cdot y_1 & x_1 \cdot y_2 & \cdots & x_1 \cdot y_m \\ x_2 \cdot y_1 & x_2 \cdot y_2 & \cdots & x_2 \cdot y_m \\ \vdots & \vdots & \ddots & \vdots \\ x_n \cdot y_1 & x_n \cdot y_2 & \cdots & x_n \cdot y_m \end{pmatrix}$$

matrix

- matrix is 2D table of numbers
  - all matrices of the same dim form a vector space
- the transpose of a matrix A, denoted A^T , is the matrix whose (i,j) entry equals the (j,i) entry of A
- Matrix multiplication
- non communicative multiplication, $AB \neq BA$ usually



Matrix multiplication

- as outer product of "inner products"

$$\begin{pmatrix} - & a_1^T & - \\ - & a_2^T & - \\ - & a_3^T & - \end{pmatrix} \begin{pmatrix} | & | & | \\ b_1 & b_2 & b_3 \\ | & | & | \end{pmatrix} = \begin{pmatrix} a_1^T b_1 & a_1^T b_2 & a_1^T b_3 \\ a_2^T b_1 & a_2^T b_2 & a_2^T b_3 \\ a_3^T b_1 & a_3^T b_2 & a_3^T b_3 \end{pmatrix}$$

- as inner product of "outer products"

$$\begin{pmatrix} | & | & | \\ b_1 & b_2 & b_3 \\ | & | & | \end{pmatrix} \begin{pmatrix} - & a_1^T & - \\ - & a_2^T & - \\ - & a_3^T & - \end{pmatrix} = b_1 a_1^T + b_2 a_2^T + b_3 a_3^T$$

Some special matrices

- Square and rectangular matrices
- Diagonal and identity matrices

$$\begin{pmatrix} 1 & 0 \\ 0 & 4 \end{pmatrix} \qquad \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

- Upper and lower triangular matrices

$$\begin{pmatrix} 1 & 0 \\ 2 & 4 \end{pmatrix} \qquad \begin{pmatrix} 1 & 2 \\ 0 & 4 \end{pmatrix}$$

- Symmetric matrices $A^T = A$
- skew-symmetric matrices $A^T = - A$
- Matrix inverse $A^{-1}A = AA^{-1} = I$
- orthogonal matrices: $A^TA = AA^T = I$, or $A^T = A^{-1}$

Solving linear equations

- The most important problem in LA is solving the linear equation: Ax = b, b is a known vector (dim n), x is unknown vector (dim m)
- A is a matrix (dim n x m): collection of m vectors

$$A = \begin{pmatrix} | & | & \cdots & | \\ a_1 & a_2 & \cdots, & a_m \\ | & | & \cdots & | \end{pmatrix} = \text{col}(a_1 a_2 \cdots, a_m)$$

- Ax represents all vectors in the **column space** of A

$$Ax = x_1 a_1 + x_2 a_2 + \cdots + x_m a_m$$

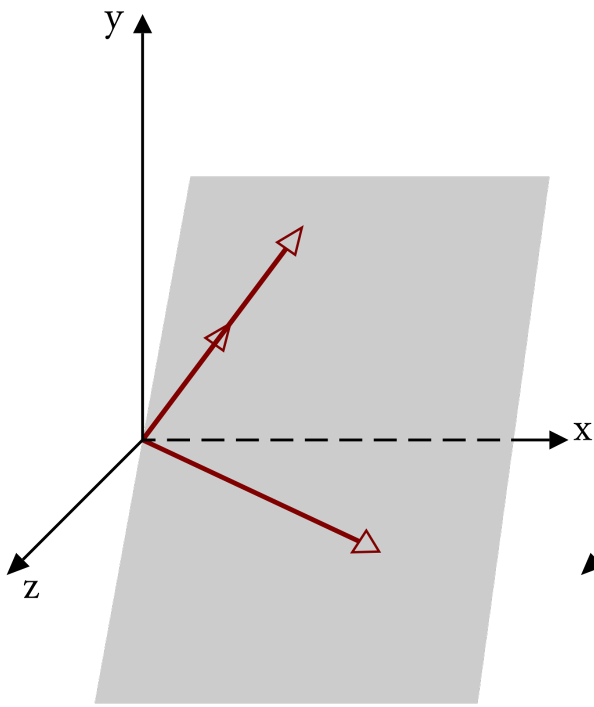
- Ax = 0 is the **null space** of A, with x = 0 always in it
- Column space determines the existence of the solution, null space determines the uniqueness of the solution

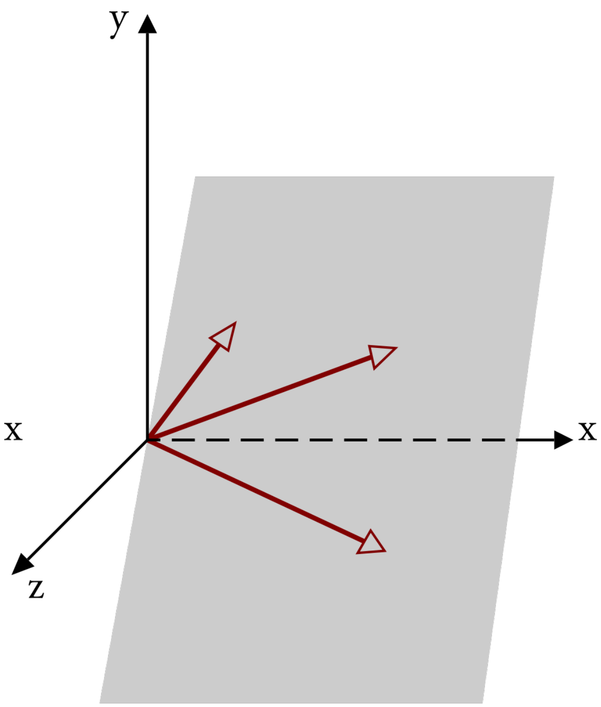
Geometric interpretation

- To solve Ax = b is equivalent to find a representation of b in the column space of A

$$Ax = x_1 a_1 + x_2 a_2 + \cdots + x_m a_m = b$$

  - If b is in col(A), solution exists
  - If null(A) = {0}, solution is unique





Solve Ax = b

- case 1: matrix A is square and full ranked
  n = m, # of equations = # of unknowns
  ⇒ complete problem ⇒ unique solution

- case 2: matrix X is tall & thin
  n > m, # of equations > # of unknowns
  ⇒ over-complete problem ⇒ no solution

- case 3: matrix A is short & fat
  n < m, # of equations < # of unknowns
  ⇒ under-complete problem ⇒ non-unique solution

matrix inverse

- for square matrix A, if $\det(A) \neq 0$, then A^{-1} is defined as the matrix satisfying $A^{-1}A = AA^{-1} = I$
  - matrix A is invertible, otherwise, it is singular
  - For a 2 x 2 matrix, inverses can be computed as

$$\mathbf{B} = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$$

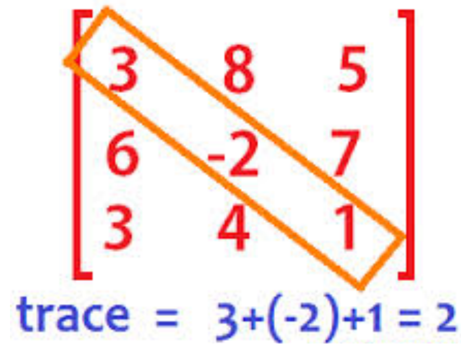
If $AD - BC \neq 0$, then B has an inverse, denoted B^{-1}

$$\mathbf{B}^{-1} = \frac{1}{AD-BC} \begin{bmatrix} D & -B \\ -C & A \end{bmatrix}$$

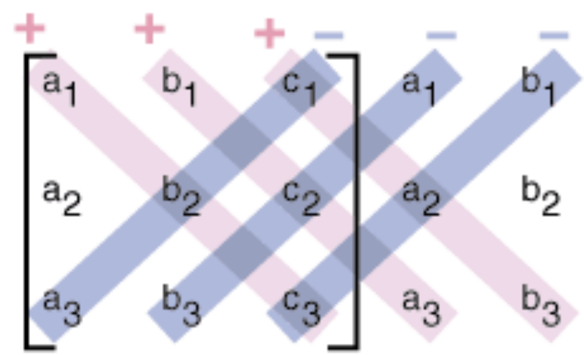
- for rectangular matrix A
  - its left Moore-Penrose pseudo inverse $(A^T A)^{-1} A^T$
  - its right Moore-Penrose pseudo inverse $A^T (A A^T)^{-1}$

Matrix trace & determinant

- trace
  - property: tr(AB) = tr(BA^T)



- determinant
  - computation involves Levi-Civita tensor



$$\det A = (a_1 b_2 c_3 + b_1 c_2 a_3 + c_1 a_2 b_3) - (a_3 b_2 c_1 + b_3 c_2 a_1 + c_3 a_2 b_1)$$

  - det(aA) = a^{n}det(A), det(AB) = det(A)det(B), det(A^{-1}) = det(A)$^{-1}$
  - A not invertible, then det(A) = 0, and vice versa

Solve Ax = b using matrix inverse

- for square matrix A, if det(A) ≠ 0, then A^{-1} is defined as the matrix satisfying $A^{-1}A = AA^{-1} = I$
  - matrix A is invertible, otherwise, it is singular

$$\begin{aligned} 2x + 3y &= 6 \\ 4x + 9y &= 15 \end{aligned} \Rightarrow \begin{bmatrix} 2 & 3 \\ 4 & 9 \end{bmatrix} \bullet \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 6 \\ 15 \end{bmatrix} \Rightarrow \begin{bmatrix} x \\ y \end{bmatrix} = A^{-1} \begin{bmatrix} 6 \\ 15 \end{bmatrix}$$

- Why is this not a good way to solve linear equation
  - Running time is $O(n^3)$
  - Numerically unstable
  - Lose of good structure in A, e.g., sparsity

$$\begin{pmatrix} 1 & 0 \\ 0 & \epsilon \end{pmatrix}^{-1} = \begin{pmatrix} 1 & 0 \\ 0 & \frac{1}{\epsilon} \end{pmatrix}$$

- On modern computers, for matrix smaller than 1000 dimension, direct inverse is feasible.

Solve Ax = b using decomposition

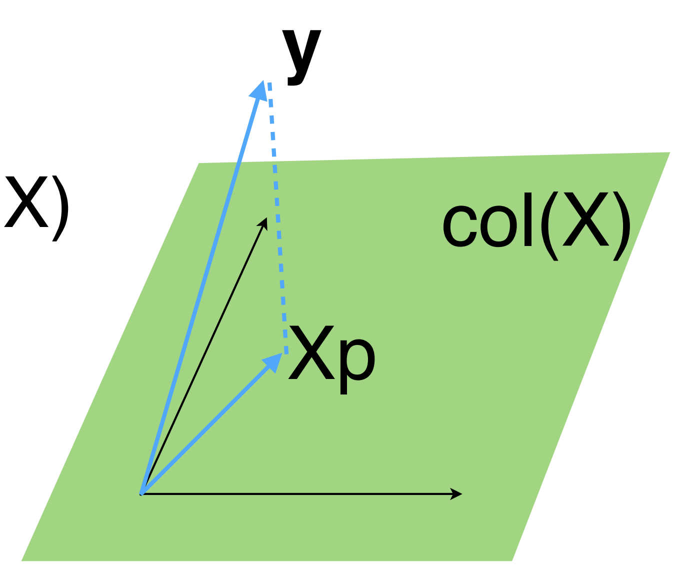
- We can decompose a square matrix A = LDU, where L and U are a lower triangular and upper matrices with diagonal 1, and D is a diagonal matrix with pivots
- If A is not invertible, then one of the pivot is zero
- Solving Ax = b becomes LDUx = b, then two steps Ly = b (forward elimination), DUx = y (backward elimination)
    - This is known as Gaussian elimination
    - Solution time is $O(n^2)$, and numerically it is very stable (caveat: if the pivots are chosen right)
    - It is numerically stable (only divide by pivot)

Projection

- for col(X) as a 2D subspace of the 3D space
- least squares problem is equivalent to finding the projection of vector **y** in col(X)

The transform $\Pi_X(\mathbf{y})$ is known as the *projection* of $\mathbf{y}$ on X. The geometrical interpretation of $\Pi_X(\mathbf{y})$ is that it is the vector in col(X) that has the minimum ℓ_2 distance to $\mathbf{y}$.

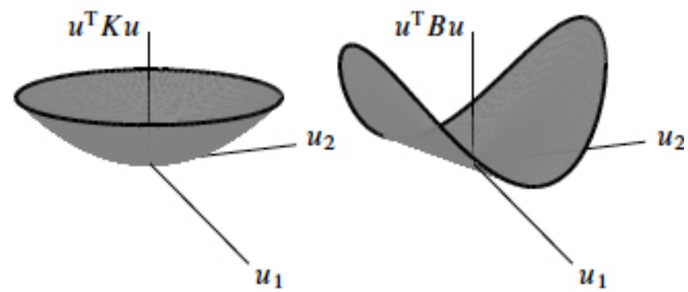
$$\Pi_X(y) = X(X^T X)^{-1} X^T y$$



- idempotent $\Pi_X(\mathbf{x}) = x$, for $x \in$ col(X)
- orthogonality $\mathbf{y} - \Pi_X(\mathbf{y}) \perp X$
- Householder transform
  mirror reflection
  $H(\mathbf{y}) = 2\,\Pi_X(\mathbf{y}) - \mathbf{y}$

Positive definite matrix

- A is a square matrix, for any $x \neq 0$, we form a quadratic form using A and x, x^TAx, then if
  - $x^TAx > 0$, A is a positive definite matrix
  - $x^TAx < 0$, A is a negative definite matrix
  - $x^TAx \geq 0$, A is a positive semi-definite matrix
  - $x^TAx \leq 0$, A is a negative semi-definite matrix
  - otherwise, A is indefinite
- Geometrical interpretation



- Symmetric positive (semi)definite matrices play a very important role in machine learning and optimization

Matrix inversion lemma

- Woodsbury identity: when A and D are invertible
  $$(A+BDC^T)^{-1} = A^{-1} - A^{-1}C(D^{-1}+CA^{-1}B^T)^{-1}B^TA^{-1}$$
  - Proof: multiply the matrix on both sides
- important special case
  - B=C=z, a vector, D=I
    $$(A+zz^T)^{-1} = A^{-1} - (A^{-1}zz^TA^{-1})/(1+z^TA^{-1}z)$$
  - B=-C=z, a vector, D=I
    $$(A-zz^T)^{-1} = A^{-1} + (A^{-1}zz^TA^{-1})/(1-z^TA^{-1}z)$$
  - caching A^{-1} and computing the inversion recursively, typical inversion will take $O(n^3)$, while this special case it is O(n)