CS 412 Introduction to Machine Learning

Linear Algebra for Machine Learning

Instructor: Wei Tang

Department of Computer Science

University of Illinois at Chicago

Chicago IL 60607

https://tangw.people.uic.edu

tangw@uic.edu

Slides credit: Sargur Srihari

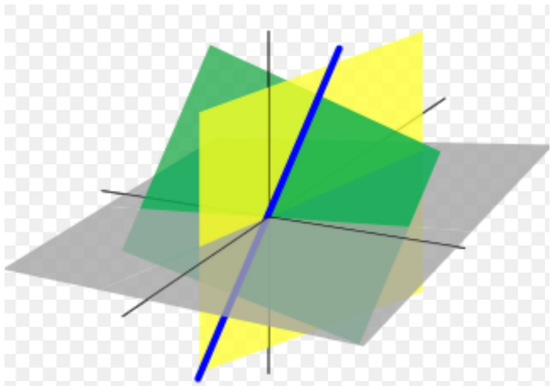
What is linear algebra?

- Linear algebra is the branch of mathematics concerning linear equations such as

$$a_1x_1 + \ldots\ldots + a_nx_n = b$$

  - In vector notation we say $\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x} = b$
  - Called a linear transformation of $\boldsymbol{x}$

- Linear algebra is fundamental to geometry, for defining objects such as lines, planes, rotations



Linear equation $a_1x_1 + \ldots\ldots + a_nx_n = b$ defines a plane in $(x_1,..,x_n)$ space

Why do we need to know it?

- Linear Algebra is used throughout engineering
- Essential for understanding ML algorithms
  - E.g., We convert input vectors $(x_1,..,x_\mathrm{n})$ into outputs by a series of linear transformations
- Here we discuss:
  - Concepts of linear algebra needed for ML
  - Omit other aspects of linear algebra

Linear Algebra Topics

- Scalars, Vectors, Matrices and Tensors
- Multiplying Matrices and Vectors
- Identity and Inverse Matrices
- Linear Dependence and Span
- Norms
- Special kinds of matrices and vectors
- Eigendecomposition
- Singular value decomposition
- The trace operator
- The determinant

Scalar

- Single number
  - In contrast to other objects in linear algebra, which are usually arrays of numbers
- Represented in lower-case italic x
  - They can be real-valued or be integers
    - E.g., let x be the slope of the line
      - Defining a real-valued scalar
    - E.g., let n be the number of units
      - Defining a natural number scalar

Vector

- An array of numbers arranged in order
- Each no. identified by an index
- Written in lower-case bold such as $\boldsymbol{x}$
  - its elements are in italics lower case, subscripted

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

- If each element is in R then $\boldsymbol{x}$ is in R^n
- We can think of vectors as points in space
  - Each element gives coordinate along an axis

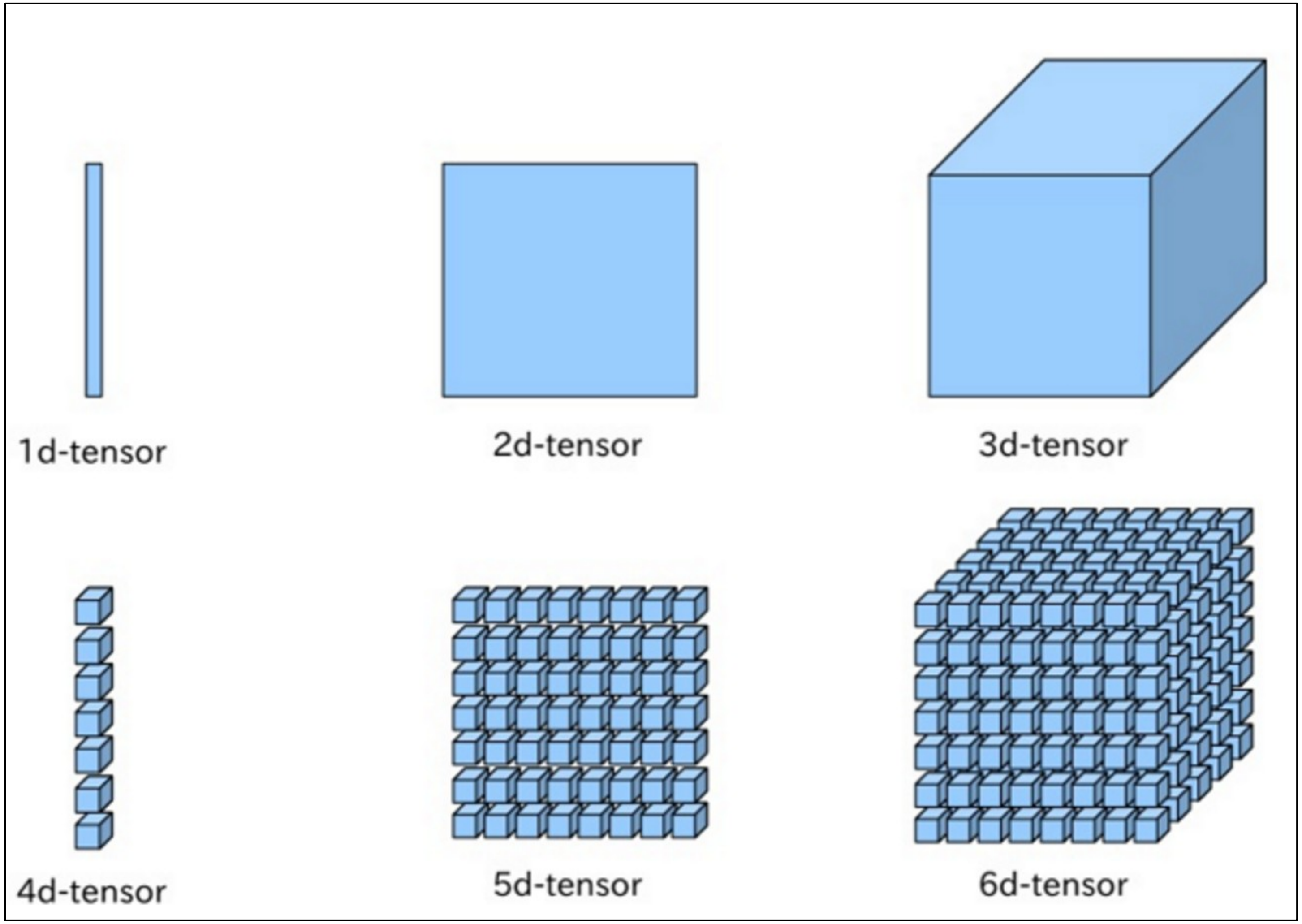
Matrices

- 2-D array of numbers
  - So each element identified by two indices
- Denoted by bold typeface $\boldsymbol{A}$
  - Elements indicated by name in italic but not bold
    - $A_{1,1}$ is the top left entry and $A_{m,n}$ is the bottom right entry
    - We can identify numbers in vertical column j by writing : for the horizontal coordinate
    - E.g., $\boldsymbol{A} = \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix}$
    - $A_{i:}$ is i^{th} row of A, $A_{:j}$ is j^{th} column of $\boldsymbol{A}$
- If A has shape of height m and width n with real-values then $\boldsymbol{A} \in \mathbb{R}^{m \times n}$

Tensor

- Sometimes need an array with more than two axes
  - E.g., an RGB color image has three axes
- A tensor is an array of numbers arranged on a regular grid with variable number of axes
  - See figure next
- Denote a tensor with this bold typeface: A
- Element (i,j,k) of tensor denoted by $\mathsf{A}_{i,j,k}$

Shapes of Tensors



Transpose of a Matrix

- An important operation on matrices
- The transpose of a matrix $\boldsymbol{A}$ is denoted as $\boldsymbol{A}^{\mathrm{T}}$
- Defined as

$$(\mathbf{A}^{\mathrm{T}})_{i,j} = A_{j,i}$$

  – The mirror image across a diagonal line
    - Called the main diagonal , running down to the right starting from upper left corner

$$A = \begin{bmatrix} A_{1,1} & A_{1,2} & A_{1,3} \\ A_{2,1} & A_{2,2} & A_{2,3} \\ A_{3,1} & A_{3,2} & A_{3,3} \end{bmatrix} \Rightarrow A^T = \begin{bmatrix} A_{1,1} & A_{2,1} & A_{3,1} \\ A_{1,2} & A_{2,2} & A_{3,2} \\ A_{1,3} & A_{2,3} & A_{3,3} \end{bmatrix}$$

$$A = \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \\ A_{3,1} & A_{3,2} \end{bmatrix} \Rightarrow A^T = \begin{bmatrix} A_{1,1} & A_{2,1} & A_{3,1} \\ A_{1,2} & A_{2,2} & A_{3,2} \end{bmatrix}$$

Vectors as special case of matrix

- Vectors are matrices with a single column
- Often written in-line using transpose

$$\boldsymbol{x} = [x_1,..,x_n]^{\mathrm{T}}$$

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \Rightarrow \boldsymbol{x}^T = [x_1, x_2, .. x_n]$$

- A scalar is a matrix with one element

$$a = a^{\mathrm{T}}$$

Matrix Addition

- We can add matrices to each other if they have the same shape, by adding corresponding elements
  - If A and B have same shape (height m, width n)

$$\boxed{C = A + B \Rightarrow C_{i,j} = A_{i,j} + B_{i,j}}$$

- A scalar can be added to a matrix or multiplied by a scalar $\boxed{D = aB + c \Rightarrow D_{i,j} = aB_{i,j} + c}$

Multiplying Matrices

- For product $\boldsymbol{C}=\boldsymbol{A}\boldsymbol{B}$ to be defined, $\boldsymbol{A}$ has to have the same no. of columns as the no. of rows of $\boldsymbol{B}$
  - If $\boldsymbol{A}$ is of shape mxn and B is of shape nxp then *matrix product* $\boldsymbol{C}$ is of shape mxp

$$C = AB \Rightarrow C_{i,j} = \sum_k A_{i,k} B_{k,j}$$

  - Note that the standard product of two matrices is not just the product of two individual elements
    - Such a product does exist and is called the element-wise product or the Hadamard product $\boldsymbol{A} ¤ \boldsymbol{B}$

Multiplying Vectors

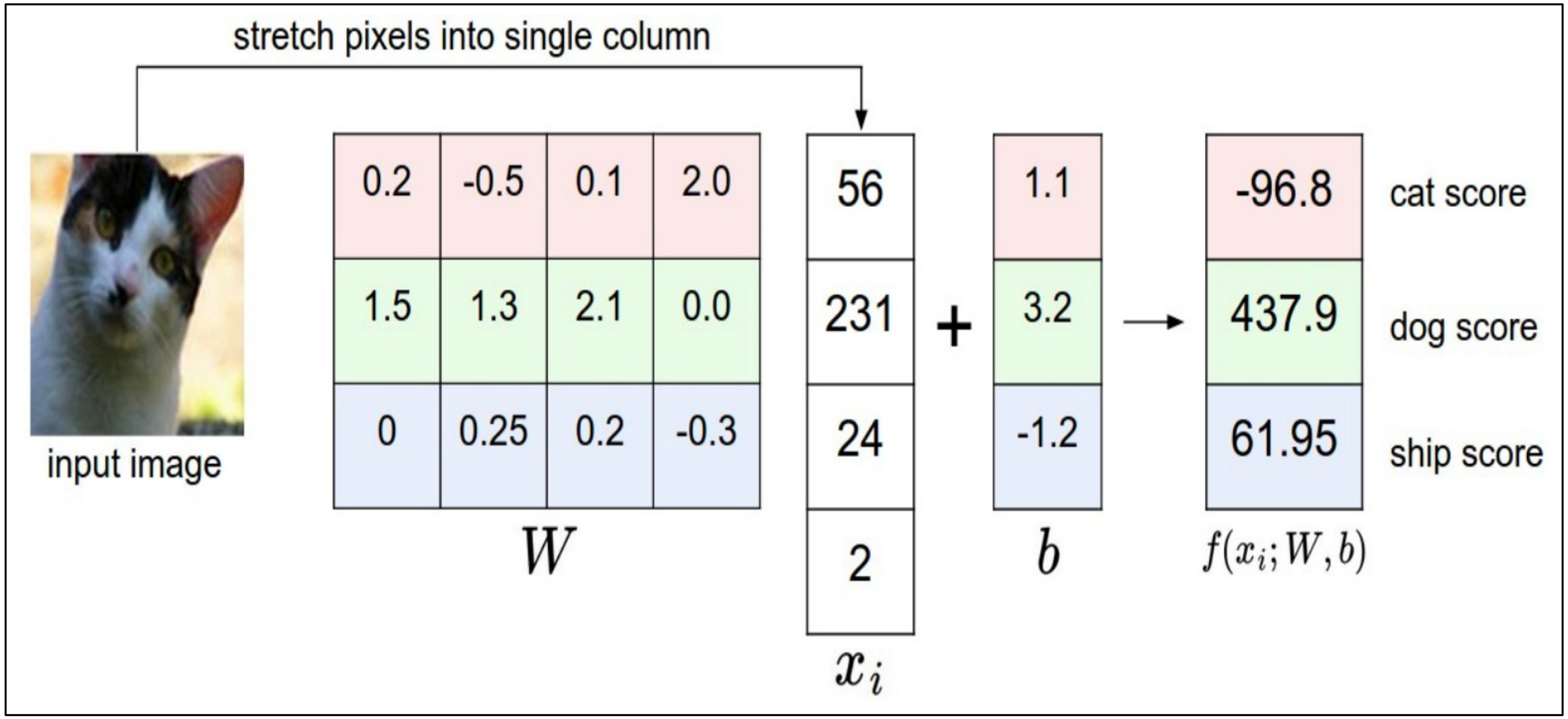
- Dot product between two vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ of same dimensionality is the matrix product $\boldsymbol{x}^{\mathrm{T}}\boldsymbol{y}$
- We can think of matrix product $C=AB$ as computing C_{ij} the dot product of row i of A and column j of B

Matrix Product Properties

- Distributivity over addition: $A(B+C)=AB+AC$
- Associativity: $A(BC)=(AB)C$
- Not commutative: $AB=BA$ is not always true
- Dot product between vectors is commutative: $\boldsymbol{x}^{\mathrm{T}}\boldsymbol{y}=\boldsymbol{y}^{\mathrm{T}}\boldsymbol{x}$
- Transpose of a matrix product has a simple form: $(AB)^{\mathrm{T}}=B^{\mathrm{T}}A^{\mathrm{T}}$

Example flow of tensors in ML

A linear classifier $y = Wx + b$



Identity and Inverse Matrices

- Matrix inversion is a powerful tool to analytically solve $A\boldsymbol{x}=\boldsymbol{b}$
- Needs concept of Identity matrix
- Identity matrix does not change value of vector when we multiply the vector by identity matrix
  - Denote identity matrix that preserves n-dimensional vectors as I_n
  - Formally $I_n \in \mathbb{R}^{n\times n}$ and $\forall \mathbf{x} \in \mathbb{R}^n, I_n \mathbf{x} = \mathbf{x}$
  - Example of I_3 $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Matrix Inverse

- Inverse of square matrix A defined as $A^{-1}A = I_n$
- We can now solve $A\boldsymbol{x}=\boldsymbol{b}$ as follows:

$$A\boldsymbol{x} = \boldsymbol{b}$$
$$A^{-1}A\boldsymbol{x} = A^{-1}\boldsymbol{b}$$
$$I_n\boldsymbol{x} = A^{-1}\boldsymbol{b}$$
$$\boldsymbol{x} = A^{-1}\boldsymbol{b}$$

- This depends on being able to find A^{-1}
- If A^{-1} exists there are several methods for finding it

Solving Simultaneous equations

- $\boldsymbol{A x} = \boldsymbol{b}$

  where $\boldsymbol{A}$ is N x N

  $\boldsymbol{x}$ is N x 1: set of weights to be determined

  $\boldsymbol{b}$ is N x 1

Closed-form solutions

- Two closed-form solutions

  1. Matrix inversion $\boldsymbol{x}=\mathrm{A}^{-1}\boldsymbol{b}$
  2. Gaussian elimination

Linear Equations: Closed-Form Solutions

1. Matrix Formulation: $Ax=b$
Solution: $x=A^{-1}b$

$$\begin{aligned} a_{11}x_1 + \ a_{12}x_2 + \cdots + \ a_{1n}x_n &= b_1 \\ a_{21}x_1 + \ a_{22}x_2 + \cdots + \ a_{2n}x_n &= b_2 \\ \vdots \qquad\quad \vdots \qquad\qquad\quad \vdots \quad &\quad \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n &= b_m \end{aligned}$$

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}$$

2. Gaussian Elimination
followed by back-substitution

$$\begin{aligned} x + 3y - 2z &= 5 \\ 3x + 5y + 6z &= 7 \\ 2x + 4y + 3z &= 8 \end{aligned}$$

$$\left[\begin{array}{ccc|c} 1 & 3 & -2 & 5 \\ 3 & 5 & 6 & 7 \\ 2 & 4 & 3 & 8 \end{array}\right] \sim \left[\begin{array}{ccc|c} 1 & 3 & -2 & 5 \\ 0 & -4 & 12 & -8 \\ 2 & 4 & 3 & 8 \end{array}\right] \sim \left[\begin{array}{ccc|c} 1 & 3 & -2 & 5 \\ 0 & -4 & 12 & -8 \\ 0 & -2 & 7 & -2 \end{array}\right] \sim \left[\begin{array}{ccc|c} 1 & 3 & -2 & 5 \\ 0 & 1 & -3 & 2 \\ 0 & -2 & 7 & -2 \end{array}\right]$$

$$\sim \left[\begin{array}{ccc|c} 1 & 3 & -2 & 5 \\ 0 & 1 & -3 & 2 \\ 0 & 0 & 1 & 2 \end{array}\right] \sim \left[\begin{array}{ccc|c} 1 & 3 & -2 & 5 \\ 0 & 1 & 0 & 8 \\ 0 & 0 & 1 & 2 \end{array}\right] \sim \left[\begin{array}{ccc|c} 1 & 3 & 0 & 9 \\ 0 & 1 & 0 & 8 \\ 0 & 0 & 1 & 2 \end{array}\right] \sim \left[\begin{array}{ccc|c} 1 & 0 & 0 & -15 \\ 0 & 1 & 0 & 8 \\ 0 & 0 & 1 & 2 \end{array}\right]$$

Disadvantage of closed-form solutions

- If A^{-1} exists, the same A^{-1} can be used for any given $\boldsymbol{b}$
  - But A^{-1} cannot be represented with sufficient precision
  - It is not used in practice
- Gaussian elimination also has disadvantages
  - numerical instability (division by small no.)
  - $O(n^3)$ for n x n matrix
- Software solutions use value of $\boldsymbol{b}$ in finding $\boldsymbol{x}$
  - E.g., difference (derivative) between $\boldsymbol{b}$ and output is used iteratively

Span of a set of vectors

- Span of a set of vectors: set of points obtained by a *linear combination* of those vectors
  - A linear combination of vectors $\{\boldsymbol{v}^{(1)},.., \boldsymbol{v}^{(n)}\}$ with coefficients c_i is $\boxed{\sum_i c_i \boldsymbol{v}^{(i)}}$
  - System of equations is $\boldsymbol{Ax}=\boldsymbol{b}$

$$\boxed{\boldsymbol{A}\boldsymbol{x} = \sum_i x_i A_{:,i}}$$

    - This is a linear combination of vectors
  - Thus determining whether $\boldsymbol{Ax}=\boldsymbol{b}$ has a solution is equivalent to determining whether $\boldsymbol{b}$ is in the span of columns of $\boldsymbol{A}$
    - This span is referred to as *column space* or *range* of $\boldsymbol{A}$

Use of a Vector in Regression

- A sample matrix
  - N samples, D features

	# hours studied	# hours playing games	# classes missed
Student #1	10	3	0
Student #2	8	20	2
Student #3	5	1	5

Grade
87
75
63

    - Feature vector has three dimensions

- This is a regression problem

Norms

- Used for measuring the size of a vector
- Norms map vectors to non-negative values
- Norm of vector $\boldsymbol{x} = [x_1,..,x_n]^T$ is distance from origin to $\boldsymbol{x}$
  - It is any function f that satisfies:

$$f(x) = 0 \Rightarrow \boldsymbol{x} = 0$$

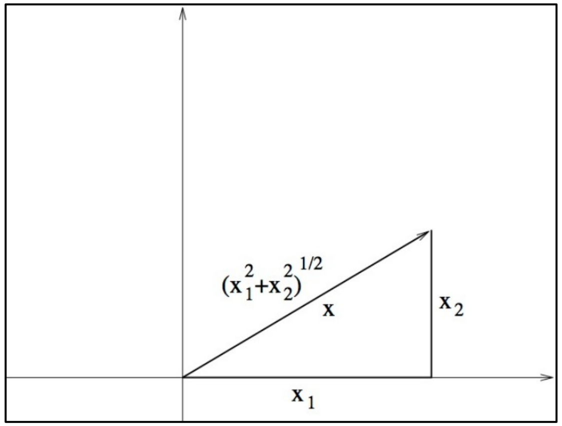
$$f(x+y) \leq f(x) + f(y) \quad \text{Triangle Inequality}$$

$$\forall \alpha \in R \quad f(\alpha x) = |\alpha| f(x)$$

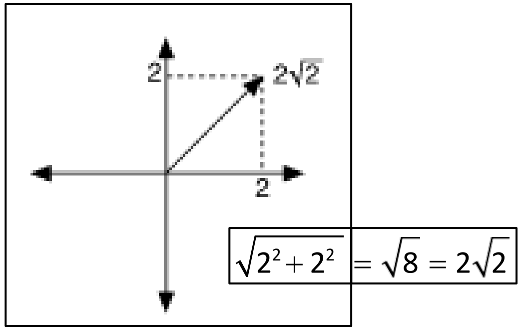
L^P Norm

- Definition:

$$\|\boldsymbol{x}\|_p = \left(\sum_i |x_i|^p \right)^{\frac{1}{p}}$$



- L^2 Norm
  - Called Euclidean norm
    - Simply the Euclidean distance between the origin and the point $\boldsymbol{x}$
    - written simply as $||x||$
    - Squared Euclidean norm is same as $\boldsymbol{x}^{\mathrm{T}}\boldsymbol{x}$



- L^1 Norm

- L^∞ Norm

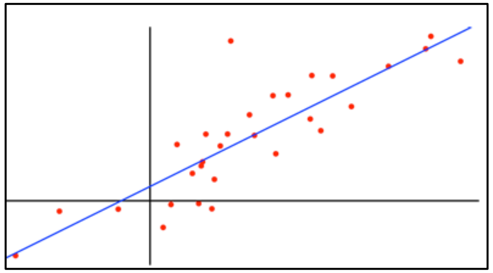
$$\|\boldsymbol{x}\|_\infty = \max_i |x_i|$$

  - Called max norm

Use of norm in Regression

- Linear Regression

  $\boldsymbol{x}$: a vector, $\boldsymbol{w}$: weight vector



$$y(\boldsymbol{x},\boldsymbol{w}) = w_0 + w_1 x_1 + .. + w_d x_d = \boldsymbol{w}^T \boldsymbol{x}$$

- Loss Function

$$\tilde{E}(\boldsymbol{w}) = \frac{1}{2}\sum_{n=1}^{N}\{y(\boldsymbol{x}_n,\boldsymbol{w}) - t_n\}^2$$

Size of a Matrix: Frobenius Norm

- Similar to L^2 norm

$$\|A\|_F = \left(\sum_{i,j} A_{i,j}^2\right)^{\frac{1}{2}}$$

$$A = \begin{bmatrix} 2 & -1 & 5 \\ 0 & 2 & 1 \\ 3 & 1 & 1 \end{bmatrix} \quad \|A\| = \sqrt{4+1+25+..+1} = \sqrt{46}$$

- Frobenius in ML
  - Layers of neural network involve matrix multiplication
  - Regularization:
    - minimize Frobenius of weight matrices $\|W(i)\|$ over L layers

$$J_R = J + \lambda \sum_{i=1}^{L} \left\|W^{(i)}\right\|_F$$

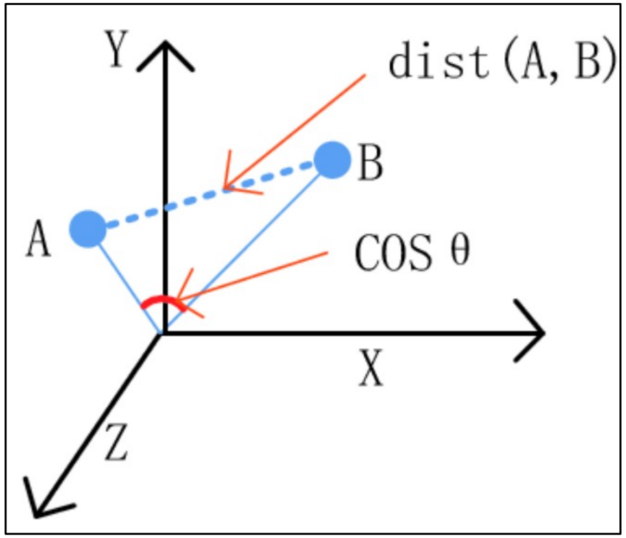
Angle between Vectors

- Dot product of two vectors can be written in terms of their L^2 norms and angle θ between them

$$\boxed{x^T y \Rightarrow ||x||_2 ||y||_2 \cos\theta}$$

- Cosine between two vectors is a measure of their similarity

$$\boxed{\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}}}$$



Special kind of Matrix: Diagonal

- Diagonal Matrix has mostly zeros, with non-zero entries only in diagonal
  - E.g., identity matrix, where all diagonal entries are 1

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

  - E.g., covariance matrix with independent features

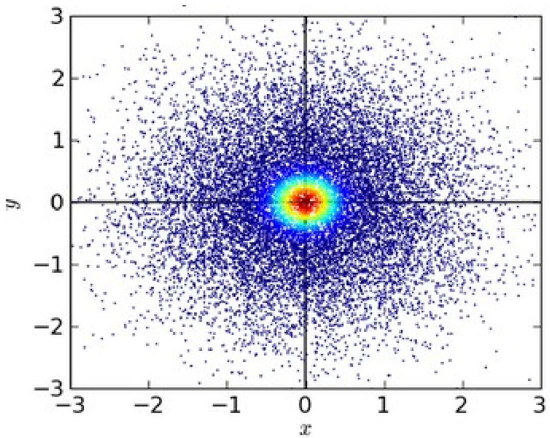
$$Cov(X,Y) = \sigma_{XY} = E[(X - \mu_X)(Y - \mu_Y)]$$

$$\text{Covariance} = \frac{\sum (x_i - x_{avg})(y_i - y_{avg})}{n-1}$$

$$\text{Covariance} = \frac{-64.57}{8}$$

$$\text{Covariance} = \boxed{-8.07}$$

$$\begin{bmatrix} \sigma_1^2 & 0 & \dots & 0 \\ 0 & \sigma_2^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma_K^2 \end{bmatrix}$$



If $\mathrm{Cov}(X,Y)=0$ then $\mathrm{E}(XY)=\mathrm{E}(X)\mathrm{E}(Y)$

$$N(\mathbf{x} \mid \boldsymbol{\mu}, \Sigma) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{x}-\boldsymbol{\mu})\right\}$$

Efficiency of Diagonal Matrix

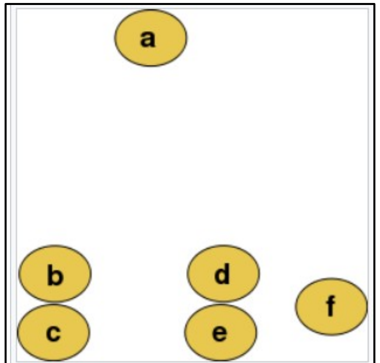
- $\text{diag}\,(\boldsymbol{v})$ denotes a square diagonal matrix with diagonal elements given by entries of vector $\boldsymbol{v}$
- Multiplying vector $\boldsymbol{x}$ by a diagonal matrix is efficient
  - To compute $\text{diag}(\boldsymbol{v})\boldsymbol{x}$ we only need to scale each x_i by v_i

$$\boxed{\text{diag}(\boldsymbol{v})\boldsymbol{x} = \boldsymbol{v} \odot \boldsymbol{x}}$$

- Inverting a square diagonal matrix is efficient
  - Inverse exists iff every diagonal entry is nonzero, in which case $\text{diag}\,(\boldsymbol{v})^{-1} = \text{diag}\,([1/v_1,..,1/v_n]^T)$

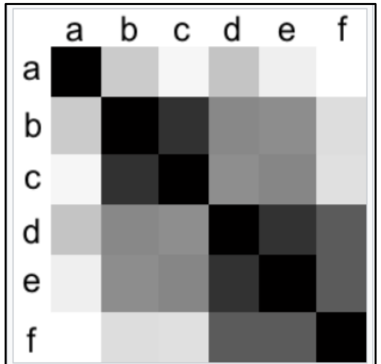
Special kind of Matrix: Symmetric

- A symmetric matrix equals its transpose: $A = A^T$
  - E.g., a distance matrix is symmetric with $A_{ij} = A_{ji}$



Raw data

	a	b	c	d	e	f
a	0	184	222	177	216	231
b	184	0	45	123	128	200
c	222	45	0	129	121	203
d	177	123	129	0	46	83
e	216	128	121	46	0	83
f	231	200	203	83	83	0



Graphical View

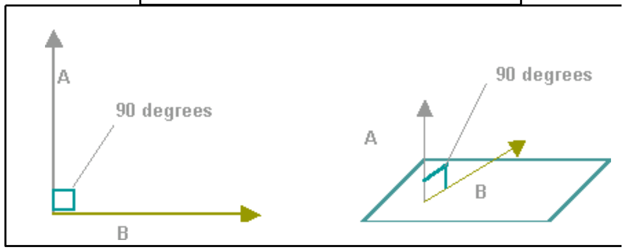
  - E.g., covariance matrices are symmetric

$$\Sigma = \begin{pmatrix} 1 & .5 & .15 & .15 & 0 & 0 \\ .5 & 1 & .15 & .15 & 0 & 0 \\ .15 & .15 & 1 & .25 & 0 & 0 \\ .15 & .15 & .25 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & .10 \\ 0 & 0 & 0 & 0 & .10 & 1 \end{pmatrix}.$$

Special Kinds of Vectors

- Unit Vector
  - A vector with unit norm $\|x\|_2 = 1$
- Orthogonal Vectors
  - A vector $\boldsymbol{x}$ and a vector $\boldsymbol{y}$ are orthogonal to each other if $\boldsymbol{x}^{\mathrm{T}}\boldsymbol{y} = 0$
    - If vectors have nonzero norm, vectors at 90 degrees to each other

$$\begin{bmatrix} 2 \\ -3 \\ -2 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 4 \\ -2 \\ 7 \end{bmatrix}$$



  - Orthonormal Vectors
    - Vectors are orthogonal & have unit norm
    - Orthogonal Matrix
      - A square matrix whose rows are mutually orthonormal: $A^{\mathrm{T}}A = AA^{\mathrm{T}} = I$
      - $A^{-1} = A^{T}$

Orthogonal matrices are of interest because their inverse is very cheap to compute

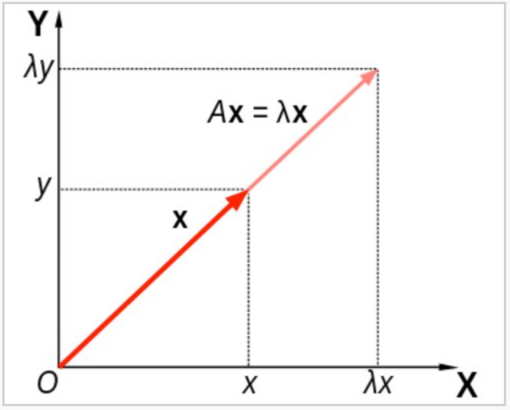
Eigenvector

- An eigenvector of a square matrix A is a non-zero vector v such that multiplication by A only changes the scale of v

$$Av = \lambda v$$

  - The scalar λ is known as eigenvalue

- If v is an eigenvector of A, so is any rescaled vector sv. Moreover sv still has the same eigen value. Thus look for a unit eigenvector



Matrix *A* acts by stretching the vector *x*, not changing its direction, so *x* is an eigenvector of *A*.

Wikipedia

Eigenvalue and Characteristic Polynomial

- Consider $Av=w$
- If v and w are scalar multiples, i.e., if $Av=\lambda v$
  - then v is an eigenvector of the linear transformation A and the scale factor λ is the eigenvalue corresponding to the eigen vector
- This is the *eigenvalue equation* of matrix A
  - Stated equivalently as $(A-\lambda I)v=0$
  - This has a non-zero solution if $|A-\lambda I|=0$ as
    - The polynomial of degree n can be factored as

$$|A-\lambda I| = (\lambda_1-\lambda)(\lambda_2-\lambda)\dots(\lambda_n-\lambda)$$

    - The $\lambda_1, \lambda_2 \dots \lambda_n$ are roots of the polynomial and are eigenvalues of A

Example of Eigenvalue/Eigenvector

- Consider the matrix $$A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$
- Taking determinant of $(A\text{-}\lambda \mathrm{I})$, the char poly is

$$\boxed{|A - \lambda I| = \begin{bmatrix} 2-\lambda & 1 \\ 1 & 2-\lambda \end{bmatrix} = 3 - 4\lambda + \lambda^2}$$

- It has roots $\lambda = 1$ and $\lambda = 3$ which are the two eigenvalues of A
- The eigenvectors are found by solving for $\boldsymbol{v}$ in $A\boldsymbol{v} = \lambda\boldsymbol{v}$, which are

$$\boxed{v_{\lambda=1} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}, v_{\lambda=3} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}}$$

Eigendecomposition

- Suppose that matrix A has n linearly independent eigenvectors $\{v^{(1)},..,v^{(n)}\}$ with eigenvalues $\{\lambda_1,..,\lambda_n\}$
- Concatenate eigenvectors to form matrix V
- Concatenate eigenvalues to form vector $\lambda=[\lambda_1,..,\lambda_n]$
- Eigendecomposition of A is given by

$$A = V\mathrm{diag}(\lambda)V^{-1}$$

Decomposition of Symmetric Matrix

- Every real symmetric matrix A can be decomposed into real-valued eigenvectors and eigenvalues

$$A = Q \Lambda Q^{\mathrm{T}}$$

where Q is an orthogonal matrix composed of eigenvectors of A: $\{v^{(1)},..,v^{(n)}\}$

orthogonal matrix: components are orthogonal or $v^{(i)\mathrm{T}} v^{(j)} = 0$

Λ is a diagonal matrix of eigenvalues $\{\lambda_1,..,\lambda_n\}$

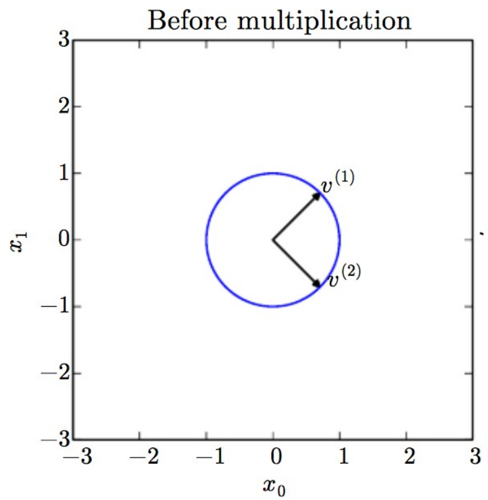
- We can think of A as scaling space by λ_i in direction $v^{(i)}$
  - See figure on next slide

Effect of Eigenvectors and Eigenvalues

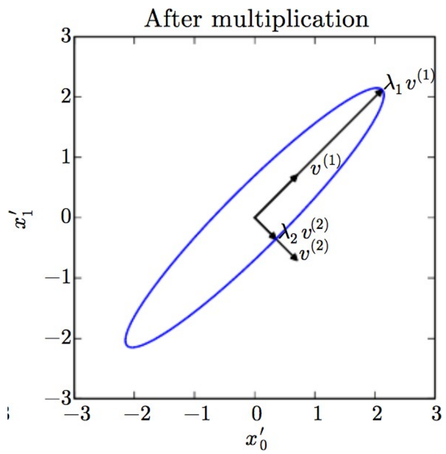
- Example of 2×2 matrix
- Matrix A with two orthonormal eigenvectors
  - $v^{(1)}$ with eigenvalue λ_1, $v^{(2)}$ with eigenvalue λ_2

Plot of unit vectors $\boldsymbol{u} \in \mathbb{R}^2$ (circle)

Plot of vectors $A\boldsymbol{u}$ (ellipse)



with two variables x_1 and x_2



Eigendecomposition is not unique

- Eigendecomposition is $A = Q\Lambda Q^T$
  - where Q is an orthogonal matrix composed of eigenvectors of A
- Decomposition is not unique when two eigenvalues are the same
- By convention order entries of Λ in descending order:
  - Under this convention, eigendecomposition is unique if all eigenvalues are unique

What does eigendecomposition tell us?

- Tells us useful facts about the matrix:
  1. Matrix is *singular* if & only if any eigenvalue is zero
  2. Useful to optimize quadratic expressions of form

     $f(\boldsymbol{x}) = \boldsymbol{x}^T A \boldsymbol{x}$ subject to $||\boldsymbol{x}||_2 = 1$

     Whenever $\boldsymbol{x}$ is equal to an eigenvector, f is equal to the corresponding eigenvalue

     Maximum value of f is max eigen value, minimum value is min eigen value

Positive Definite Matrix

- A matrix whose eigenvalues are all positive is called *positive definite*
  - Positive or zero is called *positive semidefinite*
- If eigen values are all negative it is *negative definite*
  - Positive definite matrices guarantee that $\boldsymbol{x}^T A \boldsymbol{x} \geq \boldsymbol{0}$

Singular Value Decomposition (SVD)

- Eigendecomposition has form: $A = V\mathrm{diag}(\lambda)V^{-1}$
  - If A is not square, eigendecomposition is undefined
- SVD is a decomposition of the form $A = UDV^T$
- SVD is more general than eigendecomposition
  - Used with any matrix rather than square ones
  - Every real matrix has a SVD
    - Same is not true of eigen decomposition

SVD Definition

- Write A as a product of 3 matrices: $A = UDV^{\mathrm{T}}$
  - If A is $m \times n$, then U is $m \times m$, D is $m \times n$, V is $n \times n$
- Each of these matrices have a special structure
  - U and V are orthogonal matrices
  - D is a diagonal matrix not necessarily square
    - Elements of Diagonal of D are called *singular values of* A
    - Columns of U are called *left singular vectors*
    - Columns of V are called *right singular vectors*
- SVD interpreted in terms of *eigendecomposition*
  - Left singular vectors of A are eigenvectors of AA^{T}
  - Right singular vectors of A are eigenvectors of $A^{\mathrm{T}}A$
  - Nonzero singular values of A are square roots of eigen values of $A^{\mathrm{T}}A$. Same is true of AA^{T}

Trace of a Matrix

- Trace operator gives the sum of the elements along the diagonal

$$\boxed{Tr(A) = \sum_{i,i} A_{i,i}}$$

- Frobenius norm of a matrix can be represented as

$$\boxed{\|A\|_F = \left(Tr(A)\right)^{\frac{1}{2}}}$$

Determinant of a Matrix

- Determinant of a square matrix $\det(A)$ is a mapping to a scalar
- It is equal to the product of all eigenvalues of the matrix
- Measures how much multiplication by the matrix expands or contracts space