Introduction to Statistical Machine Learning CSC/DSCC 265/465

Lecture 2: Math Review

Cantay Caliskan

Notes and updates

Small syllabus update: Assignments

- **Problem Sets**
  - You will complete **on your own**
- **Midterm**
  - You will complete **on your own**
- **Challenge**
  - You will complete **in teams of 2 people (if you prefer)**
- **Final Project**
  - You will complete **in teams of 5 people**

Notes and updates

- **Class facebook** is uploaded on BlackBoard
- **Slack channel** is online!
  - Please join our channel ☺
- **Office hours** for our TAs will be posted today!
- **Problem Set 1** will be posted after this lecture!

- Enrolling in class:
  - Please contact Lisa Altman (lisa.altman@rochester.edu) to get on a waitlist to formally enroll in the class
  - If you are considering dropping, it would be good to do so before January 26 (Wednesday) – since we have 'quite a few' students waiting to join the class!

Grading guidelines for problem sets

- **Code**
  - Sufficient amount of comments should be provided in the code
  - Questions should be clearly numbered and the answers should be clearly structured
  - Coding files should be provided in **.ipynb** format (one of the *deliverables*)
  - Code should be included in your **.pdf** file (the other *deliverable*)
- **Technical Aspects**
  - Don’t make any logical mistakes
  - Show all of the work you have done in a concise and clear manner
    - Provide the solution step-by-step
    - If any citation needed, please provide the citation
- **Short Answers**
  - **Assignments must written with LaTeX or MS Word and exported as a .pdf file**
  - If you don’t provide an answer, no points will be given
  - Make sure that the short answers have been written carefully, have a good flow, and significant aspects of the analysis have been covered in the answer
  - **The answers provided should be your own work.**

Today and next class

- Prerequisites
- Linear algebra
- Probability review (next class)
- Supervised learning (next week)

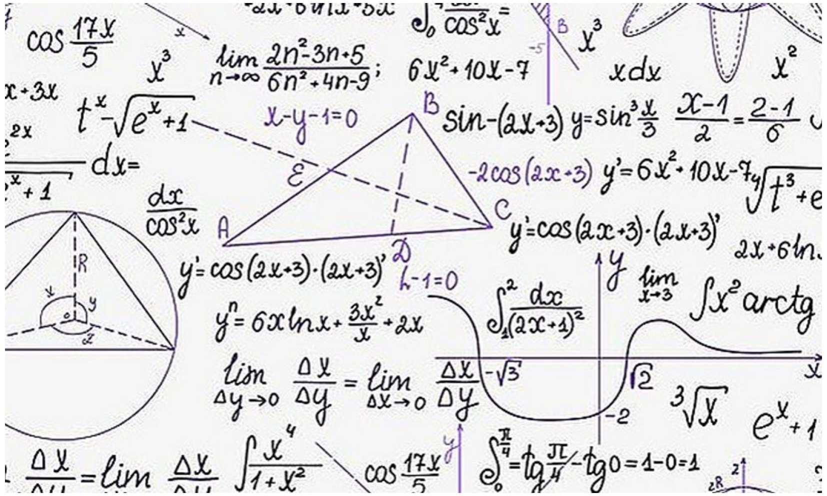
Prerequisites

Who should take this class?

- This is a difficult, fairly math- and programming-intensive class attended both by undergraduate and graduate students

- For most of the assignments, undergraduate and graduate students will be subject to same grading criteria

- If you haven’t heard about any ML algorithm at all, this may not be the course for you
- If you don’t have programming experience, you should take another course



Course prerequisites

- **DSCC / CSC / STAT 262:** Computational Intro to Statistics

or

- **STAT 212:** Applied Statistics I

or

- **STAT 213:** Elements of Probability and Math Statistics

and

- **CSC 161:** Introduction to Programming

or

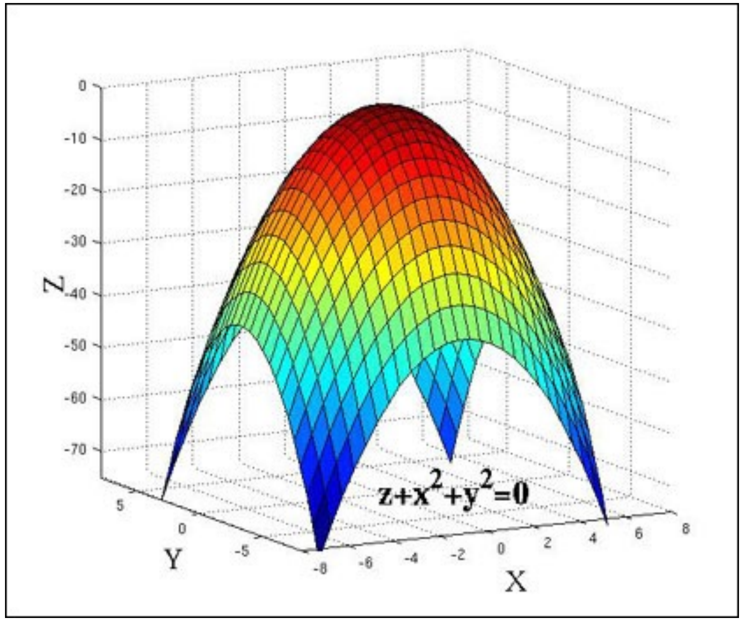
- **CSC 171:** Introduction to Computer Science

Data Mining (DSCC 240/440) is not a formal prerequisite, but highly recommended before taking this class.

UNIVERSITY of ROCHESTER

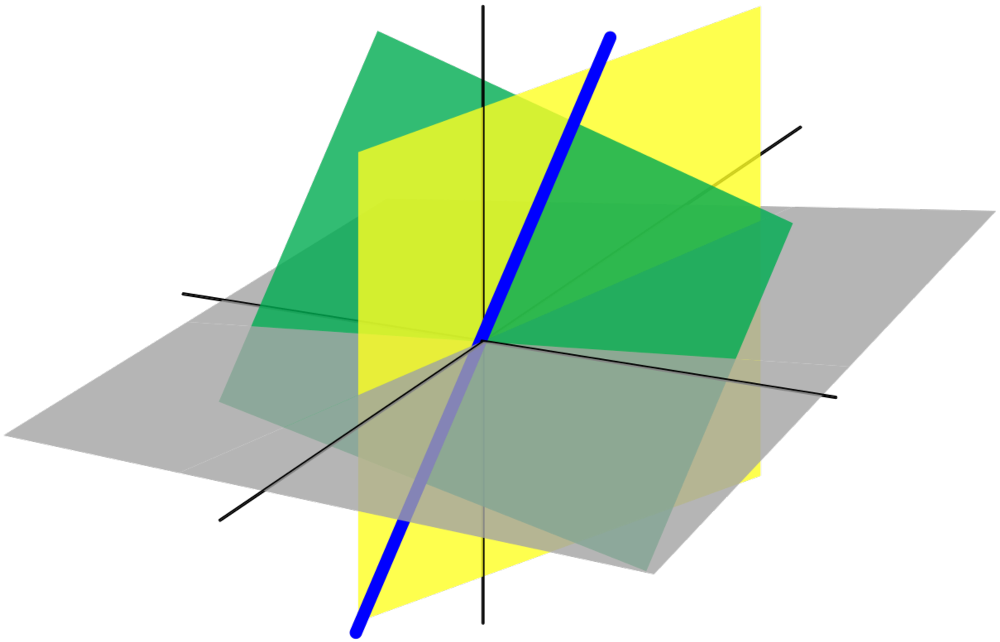
Course prerequisites

- **Multivariate Calculus**
  - Vectors; dot product
  - Determinants; cross product
  - Matrices; inverse matrices
  - Square systems; equations of planes
  - Parametric equations for lines and curves
  - Max-min problems; least squares
  - Second derivative test; boundaries and infinity
  - Level curves; partial derivatives; tangent plane approximation
  - Differentials; chain rule
  - Gradient; directional derivative; tangent plane
  - Lagrange multipliers
  - Non-independent variables
  - Double integrals
  - Change of variables



Useful to know

- **Linear algebra**
- Vectors and matrices
  - Basic matrix operations
  - Determinants, norms, trace
  - Special matrices
- Matrix inverse
- Matrix rank
- Eigenvalues and eigenvectors
- Matrix calculus

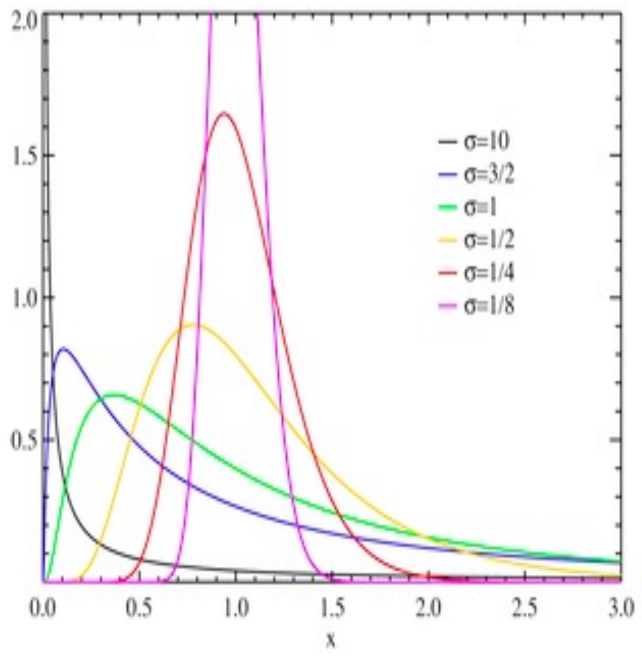


Useful to know

- **Probability**
  - Probability rules, conditional probability and independence, Bayes rule

  - Random variables (expected value, variance)
  - Discrete and continuous variables
  - Density functions
  - Covariance
  - Joint distributions

  - Normal, Bernoulli, Binomial, Multinomial, Uniform distributions



Course prerequisites

python™

NumPy

pandas

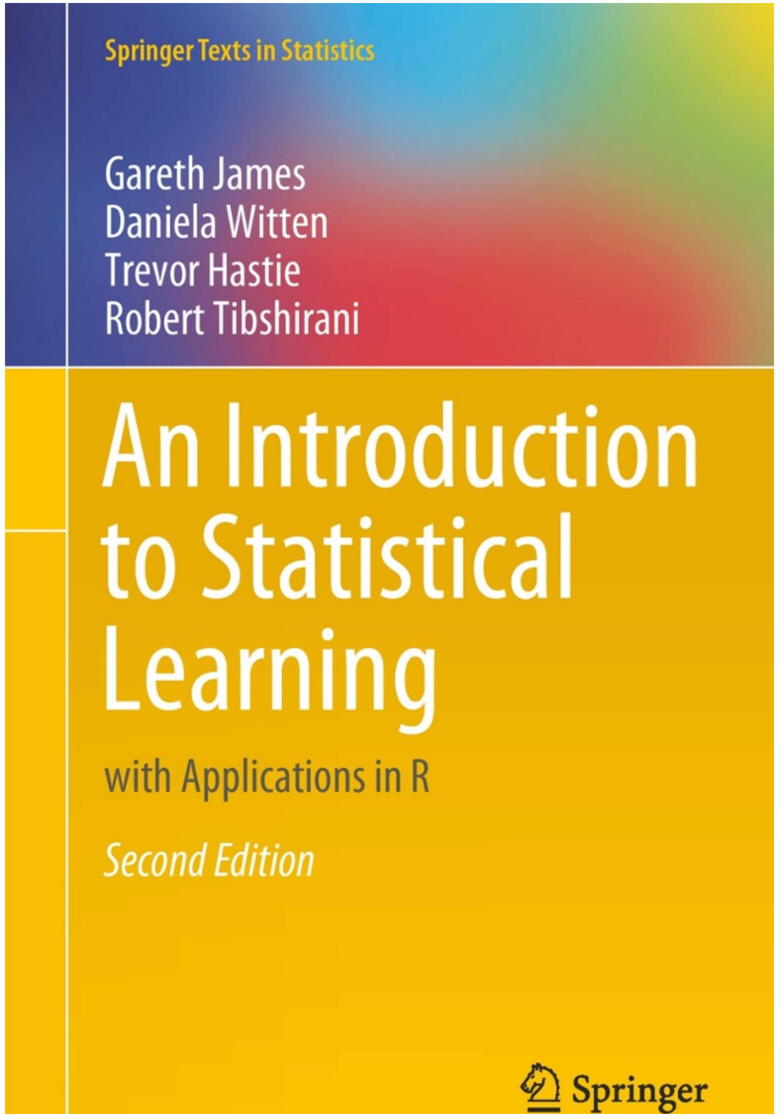
scikit learn

Course prerequisites

- **”I am not sure, but can I still take this course?”**
  - If you are not sure about the prerequisites or if you are not ready to learn ‘some new statistics’, you should not take this class
  - There won’t be enough time to teach yourself what you don’t know yet

UNIVERSITY of ROCHESTER

Read the book ☺



Let’s review stuff!

Matrix algebra

- Vectors and matrices
  - Basic Matrix Operations
  - Determinants, norms, trace
  - Special matrices
- Matrix inverse
- Matrix rank
- Eigenvalues and Eigenvectors
- Matrix Calculus

Matrix algebra

- Vectors and matrices
  - Basic Matrix Operations
  - Determinants, norms, trace
  - Special matrices

Let's take a look at these first.

- Matrix inverse
- Matrix rank
- Eigenvalues and Eigenvectors
- Matrix Calculus

Scalar

- An element of a field which is used to define a ***vector space***
  - *Multiple scalars combined make a **vector***

  - Scalars can be real numbers or complex numbers

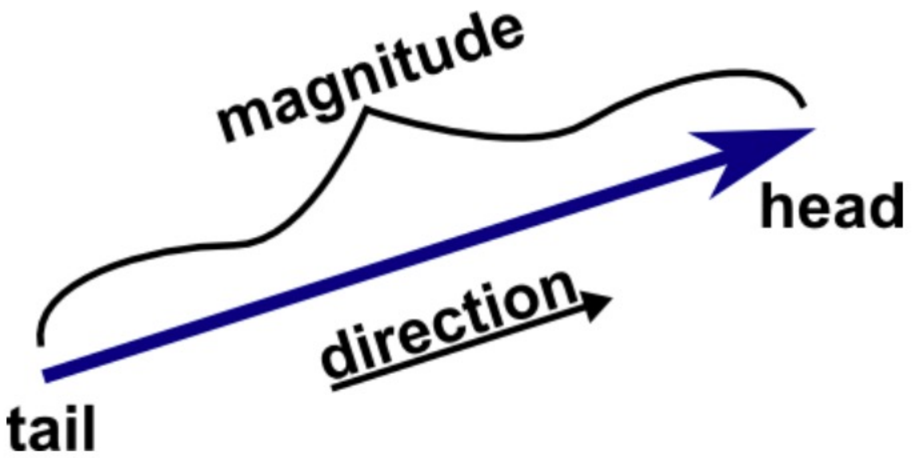
  - Is actually a ***1x1*** matrix (has one row, one column)
    - <u>Example:</u> A single number is a scalar

  - Is used to informally refer to a ***vector***, ***matrix***, ***tensor*** (because of the above)

  - Comes from the <u>Latin</u> Word ***scala*** (means 'ladder') ☺

Vector

- A mathematical object that has both a ***magnitude*** and ***direction***
- Can be represented with a 'collection' of numbers
- Can be in any *n-dimensional* feature space where *n > 0*



- Two vectors are the same, if they have:
  - The same *magnitude*
  - The same *direction*
  - If they are the same: their ***cosine similarity is 1***

Vector

- A <u>column</u> vector $v \;\in R^{m\;x\;1}$
  - ***m***: #rows
  - ***n***: *#columns* **(and specifically n = 1 here)**

$$\boldsymbol{v} = \begin{bmatrix} v_1 \\ v_2 \\ . \\ . \\ . \\ v_m \end{bmatrix}$$

By default, you should assume that a vector is a column vector.

- A row vector $v^T \;\in R^{1\;x\;n}$ **Rows and columns are switched here!**

$$\boldsymbol{v^T} = [v_1\; v_2 \;\; \ldots \;\; v_n]$$

- T denotes the transpose operation

Matrix

- A matrix $\boldsymbol{A} \in R^{m \times n}$ is an array of numbers sized **'m by n'**
  - ***m***: #rows
  - ***n****: #columns*

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{a_{11}} & \boldsymbol{a_{12}} & \boldsymbol{a_{13}} & \dots & \boldsymbol{a_{1n}} \\ \boldsymbol{a_{21}} & \boldsymbol{a_{22}} & \boldsymbol{a_{23}} & \dots & \boldsymbol{a_{2n}} \\ . & . & . & & . \\ . & . & . & & . \\ . & . & . & & . \\ \boldsymbol{a_{m1}} & \boldsymbol{a_{m2}} & \boldsymbol{a_{m3}} & \dots & \boldsymbol{a_{mn}} \end{bmatrix}$$

  - If $m = n$, we then A is square ⟵ **And, we have a square matrix!**

  - Question: Can a matrix have more than two dimensions?

Basic Matrix Operations

- Things we can fairly easily do:
  - Addition
  - Scaling
  - Dot product
  - Multiplication
  - Transpose
  - Inverse / pseudoinverse
  - Determinant / trace

Matrix operations are at the center of 'optimization'

Basic Matrix Operations

Reading Matrices

- Let's say we have the following matrix ***A***:

$$\begin{bmatrix} 1 & 9 & -13 \\ 20 & 5 & -6 \end{bmatrix}$$

- This is a **2 x 3** (2 by 3) matrix

- A_{ij} = '*i,j*' entry in i^{th} row, j^{th} column

- <u>Question</u>: What is A_{23} ?
- <u>Question</u>: What is A_{32} ? **Doesn't exist**

Reading Vectors

- Let's say we have the following vector ***v***:

$$\begin{bmatrix} 460 \\ 232 \\ 315 \\ 178 \end{bmatrix}$$

1-indexed vs 0-indexed:

- $\boldsymbol{v_i}$ = $\boldsymbol{i^{th}}$ element

$$y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} \qquad y = \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \end{bmatrix}$$

Matrix Addition

- Two matrices ***A*** and ***B*** that are to be added need to have the <u>equal</u> number of rows and columns

$$\mathbf{A}+\mathbf{B}=\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ b_{21} & b_{22} & \cdots & b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ b_{m1} & b_{m2} & \cdots & b_{mn} \end{bmatrix}$$

$$= \begin{bmatrix} a_{11}+b_{11} & a_{12}+b_{12} & \cdots & a_{1n}+b_{1n} \\ a_{21}+b_{21} & a_{22}+b_{22} & \cdots & a_{2n}+b_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1}+b_{m1} & a_{m2}+b_{m2} & \cdots & a_{mn}+b_{mn} \end{bmatrix}$$

Matrix Addition

- Two matrices ***A*** and ***B*** that are to be added need to have the <u>equal</u> number of **rows and columns**

$$\begin{bmatrix} 1 & 3 \\ 1 & 0 \\ 1 & 2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 7 & 5 \\ 2 & 1 \end{bmatrix} = \begin{bmatrix} 1+0 & 3+0 \\ 1+7 & 0+5 \\ 1+2 & 2+1 \end{bmatrix} = \begin{bmatrix} 1 & 3 \\ 8 & 5 \\ 3 & 3 \end{bmatrix}$$

Scalar Multiplication

- Each element in the matrix is multiplied with the given ***scalar***
- Let's find ***4*A***, where:

$$A = \begin{bmatrix} 2 & 1 \\ 3 & -2 \end{bmatrix}$$

$$4\begin{bmatrix} 2 & 1 \\ 3 & -2 \end{bmatrix} = \begin{bmatrix} 4 \cdot 2 & 4 \cdot 1 \\ 4 \cdot 3 & 4 \cdot (-2) \end{bmatrix} = \begin{bmatrix} 8 & 4 \\ 12 & -8 \end{bmatrix}$$

- Follow the same rule for division

Matrix multiplication

- Let's say we have three matrices ***A***, ***B***, and ***C*** such that ***AB = C***
- The number of the columns in ***A*** needs to be equal to number of the rows in ***B***
- If ***A*** is *m x n* matrix, and ***B*** is an *n x p* matrix
- The resulting ***C*** will be a *m x p* matrix

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix} \quad \mathbf{B} = \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{pmatrix} \quad \mathbf{C} = \begin{pmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mp} \end{pmatrix}$$

$$c_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + \cdots + a_{in}b_{nj} = \sum_{k=1}^{n} a_{ik}b_{kj}$$

Matrix multiplication

- Let's take a look at an example:

$$\begin{bmatrix} -4 & -3 & -2 \\ 6 & 0 & -1 \\ 2 & 1 & 3 \end{bmatrix}\begin{bmatrix} 5 & 4 \\ 6 & 7 \\ -4 & -3 \end{bmatrix} = \begin{bmatrix} -4\times 5 + -3\times 6 + -2\times -4 & -4\times 4 + -3\times 7 + -2\times -3 \\ 6\times 5 + 0\times 6 + -1\times -4 & 6\times 4 + 0\times 7 + -1\times -3 \\ 2\times 5 + 1\times 6 + 3\times -4 & 2\times 4 + 1\times 7 + 3\times -3 \end{bmatrix}$$

$$= \begin{bmatrix} -30 & -31 \\ 34 & 27 \\ 4 & 6 \end{bmatrix}$$

Matrix multiplication

- What about multiplying with ***Identity*** matrix?

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{bmatrix}$$

Vectors

- **Norm** (of a vector)

$$\|x\|_2 = \sqrt{\sum_{i=1}^{n} x_i^2}$$

What type of a norm is this?

Answer: Euclidean norm

- To be specific, norm is a function $f : R^n \rightarrow R$ that satisfies four properties:
    - ***Non-negativity:*** For all $x \in R^n, f(x) \geq 0$
    - ***Definiteness:*** $f(x) = 0 \; iff \; x = 0$
    - ***Homogeneity:*** For all $x \in R^n, t \in R \rightarrow f(tx) = |t| f(x)$
    - ***Triangle inequality:*** For all $x, y \in R^n, f(x+y) \leq f(x) + f(y)$

Norms

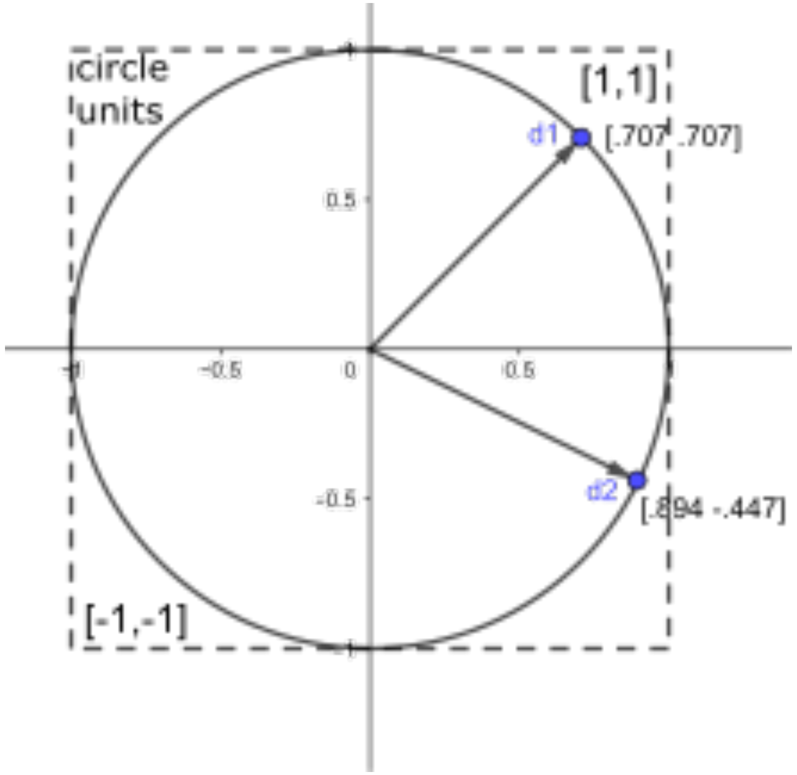
- $\|x\|_1 = \sqrt{\sum_{i=1}^{n} x_i^1}$ **Manhattan norm**
- $\|x\|_2 = \sqrt{\sum_{i=1}^{n} x_i^2}$ **Euclidean norm**
- …
- $\|x\|_\infty = max_i|x_i|$ **Easy proof through triangle inequality!**

- General l_p norms: $\|x\|_p = (\sum_{i=1}^{n}|x_i|^p)^{1/p}$ **Minkowski norm**

Matrix Operations

- What is a ***unit vector***?
  - **= A vector of length 1**
- How do we convert a vector into a ***unit vector***?
  - $v_N = \frac{v}{|v|}$

All 2D unit vectors can be found in a circle with radius = 1

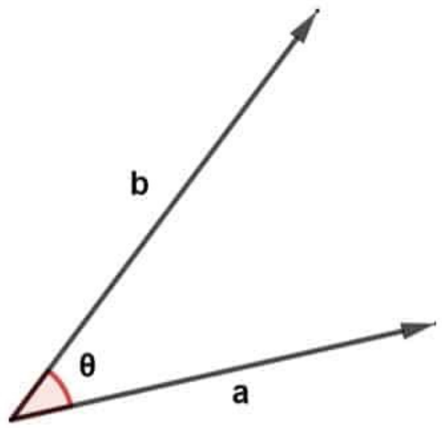


  - Also called ***normalized vector***

Matrix Operations

- ***Dot product*** (inner product) of vectors
  - Multiply corresponding entries of two vectors and add up the result
  - How is this useful?
    - A proxy for the ***angle*** between two vectors

Angle Between Two Vectors



- $x^T y = [x_1 \ \ \dots \ \ x_n] \begin{bmatrix} x_1 \\ x_2 \\ . \\ . \\ . \\ x_n \end{bmatrix} = \sum_{i=1}^{n} x_i y_i$

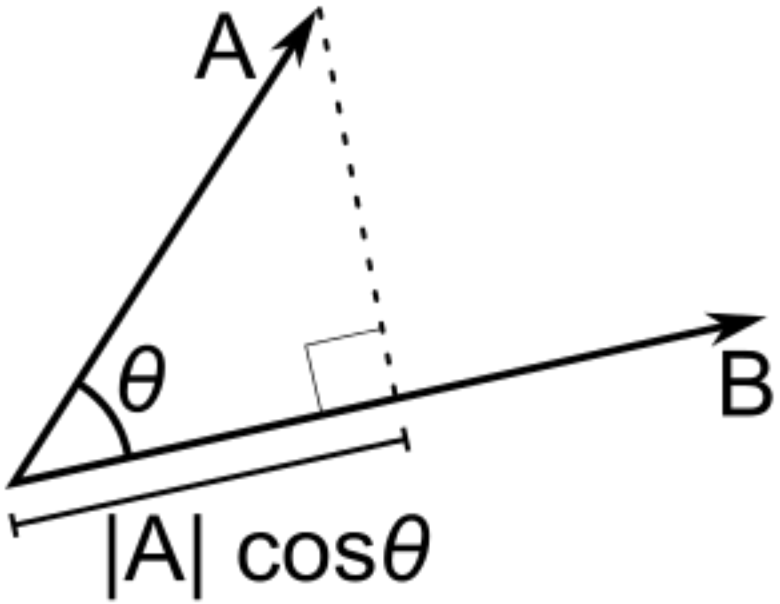
What do we get as the result?

A scalar.

Matrix Operations

- ***Dot product*** (inner product) of vectors

- If ***B*** is a unit vector, then ***A*B*** gives the length of ***A*** which lies in the direction of ***B***



Matrix Operations

- ***Product*** of two matrices
- Matrix multiplication is associative: ***(AB)C = A(BC)***
- Matrix multiplication is distributive: ***A(B+C) = AB + AC***
- In most cases, matrix multiplication is not commutative:
  - ***AB ≠ BA***
  - Note: We can only multiply two matrices ***A*** and ***B*** if the #columns in ***A*** is equal to #rows in ***B***
  - $\boldsymbol{A} \in \boldsymbol{R}^{m\ x\ n}\ and\ \boldsymbol{B} \in \boldsymbol{R}^{n\ x\ q}$

UNIVERSITY of ROCHESTER

Matrix Operations

- ***Powers***

- We can multiply a matrix with itself
  - $\boldsymbol{A*A}$
  - $\boldsymbol{A^2}$
  - $\boldsymbol{A^3}$ is possible

  - Question: When is $\boldsymbol{A*A}$ possible?
    - When $\boldsymbol{A}$ is a square matrix

Transpose

- **Transpose**: A matrix flipped over its diagonal
- Example:

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}^T = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \end{bmatrix}$$

A

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$$

- Useful properties:

$$(ABC)^T = C^T B^T A^T$$
$$(A^T)^T = A$$
$$(A + B)^T = A^T + B^T$$
$$(A^T)^{-1} = (A^{-1})^T$$

How is transpose useful?

Rotation
Scaling
Data manipulation

Determinant

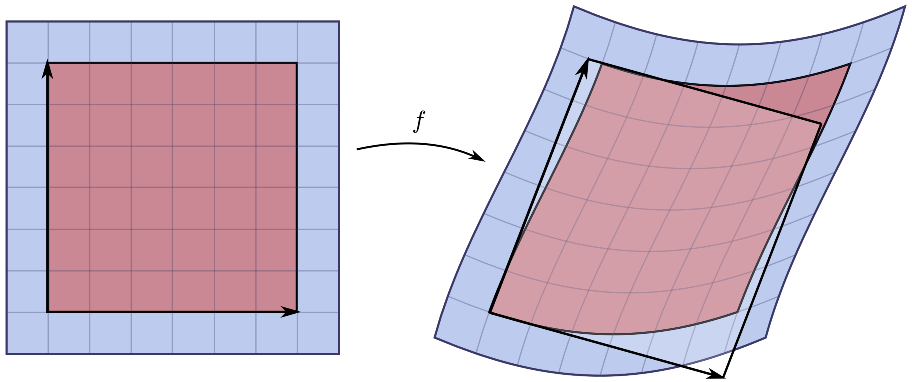
- A ***scalar*** value
- Can only be calculated for <u>square matrices</u>
- Denoted as **det(A)**, **det A** or $|A|$
- **For a *2x2*** matrix, we have: $$|A| = \begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc.$$
- For a ***3x3*** matrix, we have: $$|A| = \begin{vmatrix} a & b & c \\ d & e & f \\ g & h & i \end{vmatrix} = a\begin{vmatrix} e & f \\ h & i \end{vmatrix} - b\begin{vmatrix} d & f \\ g & i \end{vmatrix} + c\begin{vmatrix} d & e \\ g & h \end{vmatrix}$$
- Properties:
  - $\det(AB) = \det(BA)$
  - $\det(A^{-1}) = \frac{1}{\det(A)}$
  - $\det(A^T) = \det(A)$
  - $\det(A) = 0 \; iff \; A \; is \; singular$

<u>Question</u>: What is a <u>singular</u> matrix?
A matrix where det(A) = 0 ☺

Determinant

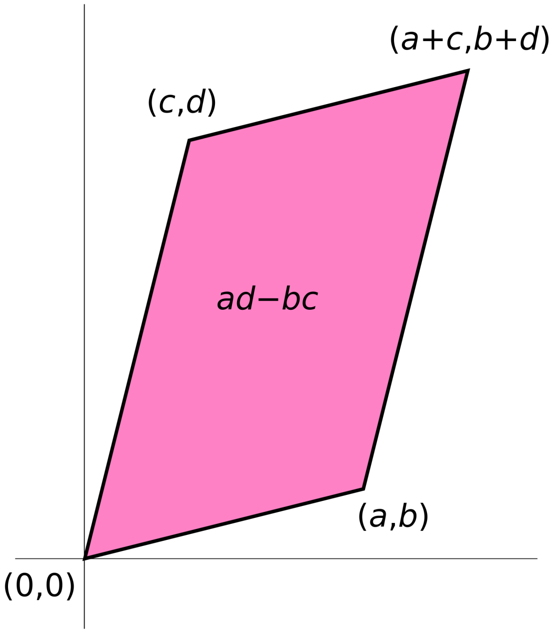
- Closely related to ***eigenvalues*** and ***characteristic polynomial*** of a matrix
  - Remember: Most ML is about ‘summarizing’ data

- Formally: The area of an n-dimensional parallelogram is the determinant of its basis vectors

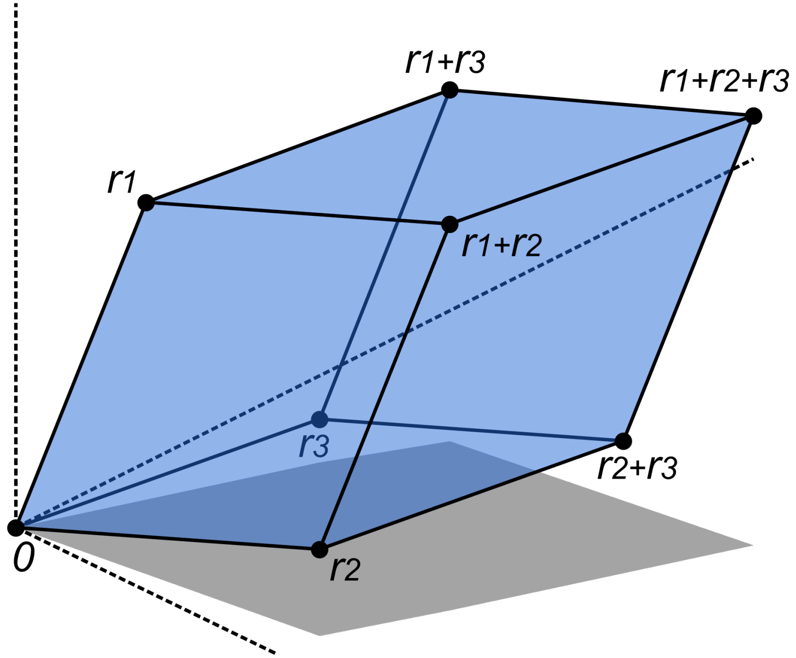


With two vectors, a small square is converted to…

Recall: Volume in 2D is ‘area’



A parallelogram in 2D



A parallelepiped in 3D

Trace

- ***Trace*** is the sum of diagonal elements

- Example: $Tr\left(\begin{bmatrix} 2 & 4 \\ 6 & 8 \end{bmatrix}\right) = 2 + 8 = 10$

- Trace is invariant to a lot of transformations. So, it is used sometimes in proofs.

- Some ***trace*** properties:

$$Tr(AB) = Tr(BA)$$
$$Tr(A + B) = Tr(A) + Tr(B)$$

Useful in ***eigendecomposition***!

Special Matrices

- ***Symmetric*** matrix:
  - $A \text{ is symmetric} \leftrightarrow A = A^T$
  - Example: Some social media networks are ***symmetric*** matrices

$$A = \begin{bmatrix} 1 & 7 & 3 \\ 7 & 4 & 5 \\ 3 & 5 & 0 \end{bmatrix}$$

- ***Skew-symmetric*** matrix:
  - $A \text{ is skew} - \text{symmetric} \leftrightarrow A = -A^T$
  - Example: Some trade networks are ***skew-symmetric*** matrices

$$\begin{bmatrix} 0 & -2 & -5 \\ 2 & 0 & -7 \\ 5 & 7 & 0 \end{bmatrix}$$

- ***Identity*** matrix:
  - A matrix that is equivalent to mathematical **1**
  - Other matrices are 'invariant' to multiplication by ***Identity*** matrix

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

- ***Diagonal*** matrix:
  - A matrix that is equal to zero (0) everywhere except for its ***diagonal***
  - Example: A social network consisting of ***loops***

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & -2 \end{bmatrix}$$

Please do the following until next week!

- Review what we have just gone through (matrix algebra)

- Start with your problem set (due date ***Wednesday, January 26, 11:59 PM***)

- Read ***Chapter 3*** from our book

- And reminder:
  - Please contact Lisa (lisa.altman@rochester.edu) to get on waitlist
  - If you are planning to drop, please do so at your earliest opportunity