UNIVERSITY OF MINNESOTA TWIN CITIES

CSCI 5304 Fall 2021

COMPUTATIONAL ASPECTS OF MATRIX THEORY

Class time	:	*MW 4:00 – 5:15 pm*
Room	:	*Keller 3-230 or Online*
Instructor	:	*Daniel Boley*

Lecture notes: http://www-users.cselabs.umn.edu/classes/Fall-2021/csci5304/

September 3, 2021

About this class. Instructor and Teaching Assistant

- Me: Daniel Boley
- TA: Calvin Roth

– Course title:

"Computational Aspects of Matrix Theory"

What you will learn and why

– Alternative title:

“Theoretical Aspects of Matrix Computations”

... a longer title:

“Everything you need to know about computations involving matrices, that you can learn in one semester – from theory to algorithms, matlab implementations, and applications.”

➤ At the core of *most* disciplines requiring numerical computing..

➤ .. and gaining importance in Computer Science (machine learning, robotics, graphics, ...)

Objectives of this course

Set 1 Fundamentals of matrix theory :

- Matrices, subspaces, eigenvectors

- Norms, matrix norms, understanding errors, sensitivity [somewhat theoretical]

- Various decompositions, LU, QR, SVD, ..

set 2 Computational linear algebra / Algorithms

- Solving linear systems, LU factorization
- Solving least-squares problems, QR factorization
- Eigenvalue problems - computing eigenvalues, eigenvectors,

Set 3 Linear algebra in applications

- See how numerical linear algebra is used to solve problems in (a few) computer science-related applications.

- Examples: page-rank, applications in optimization, information retrieval, applications in machine learning, control, ...

Logistics:

➤ There is a cselabs class web-site for this course in :

`http://www-users.cselabs.umn.edu/classes/Fall-2021/csci5304`

➤ There you will find :

● Lecture notes, syllabus, schedule, a matlab folder, homework questions and some basic information [including link to Canvas]

● Most Everything else will be found on Canvas. [some info duplicated]

Please Note:

➤ Homeworks and solutions are copyrighted

- *Some solutions of HWs and tests may be provided to you [Canvas] with the hope of helping you understand the material better. By accessing them you agree not to send them to others, sell them (%#!!$), or otherwise (help) make them available via external web-sites.*

About lecture notes:

- Lecture notes (like this first set) will be posted on the class web-site – usually before the lecture.

- Note: format of notes used in class may be slightly different from the one posted – but contents are identical.

- Review them to get some understanding if possible before class.

- Read the relevant section (s) in one of the references listed

- Lecture note sets are grouped by topics rather than by lecture.

- In the notes the symbol ✍1 indicates suggested easy exercises or questions – often [not always] done in class.

- Each set of lecture notes may have a supplement which includes solutions to some of these exercises – possibly additional notes/comments.

Occasional in-class exercises (quizzes)

➤ Posted during class in the Homework folder

➤ Do them in class or by the due date (at the latest one day after being posted).

➤ These quizzes are not graded for correctness, but should be handed in. Full credit is given for any good-faith effort to answer the questions even if not correct.

Matlab

- You will need to use matlab or octave for testing algorithms.
- Some documentation is posted in the (class) matlab folder .
- Matlab should be available for free for all registered CSE students, but most everything done in the class can be done with Octave, a free Matlab look-alike.
- There are common alternatives to Matlab, namely Julia, Python (numpy, scipy), R, but they have very different syntax. Matlab notation is still very popular in technical writings.
- The Notes folder has pointers to side-by-side listing of common matrix expressions in Matlab, numpy, R, etc.
- Matlab diaries used for the demos (if any) will be posted in the Notes folder.

- If you do not know matlab at all and have difficulties with it talk to me or the TA at office hours. All you may need is some initial help to get you started.

Final remarks on lecture notes

➤ These notes are ‘evolving’. Please do not hesitate to report errors and/or provide feedback on content.

➤ On occasion I will repost lecture notes with changes/additions

How to study for this course:

1) Rely primarily on lecture notes as a starting point. Use other sources [e.g., (online) books] to get a deeper understanding.

2) Do the practice exercises indicated in lecture notes + the occasional practice exercise sets before class.

3) Ask questions! Participate in discussions (office hours, canvas, ...)

Zoom (& Canvas) etiquette

➤ Please make sure you are on mute when not communicating with class (make sure to unmute after you ask a question)

➤ You can also use 'chat' for quick questions [I try to check those regularly, but if I miss you, feel free to unmute yourself.]

➤ When possible, it is suggested that you switch the video on for interactive class activities. You can use zoom backgrounds to avoid sharing images of personal space.

Priority: Maintain an inclusive and good learning atmosphere

➤ Be courteous. Everyone should feel comfortable asking questions or offering useful comments either on Zoom or in Canvas discussions.

➤ If there is an outage [zoom, internet,...] stay tuned + try to reconnect. In worst case: I will re-record lecture and post it.

⟶ Do read syllabus info on University rules on privacy (sharing of Zoom videos etc.) and Covid protocols.

➤ The Zoom office hours will be open to all without “waiting room”. We will see how this works and make changes as needed.

➤ For private office hours: contact us [email]

GENERAL INTRODUCTION

- **Background: Linear algebra and numerical linear algebra**
- **Types of problems to be seen in this course**
- **Mathematical background - matrices, eigenvalues, rank, ...**
- **Types of matrices, structutred matrices,**

Introduction

➤ This course is about *Matrix algorithms* or "matrix computations"

➤ It involves: algorithms for standard matrix computations (e.g. solving linear systems) - and their analysis (e.g., their cost, numerical behavior, ..)

➤ Matrix algorithms pervade most areas of science and engineering.

➤ In computer science: recent increase of interest in matrix algorithms for data mining, information retrieval, search engines, pattern recognition, graphics, ...

Examples

➤ Modern version of an old problem

A set of 12 coins containing nickels (5c each), dimes (10c each) and quarters (25c each) totals to $1.45. In addition, the total without the nickels amounts to $1.25. How many of each coin are there?

➤ Problem type: Linear system

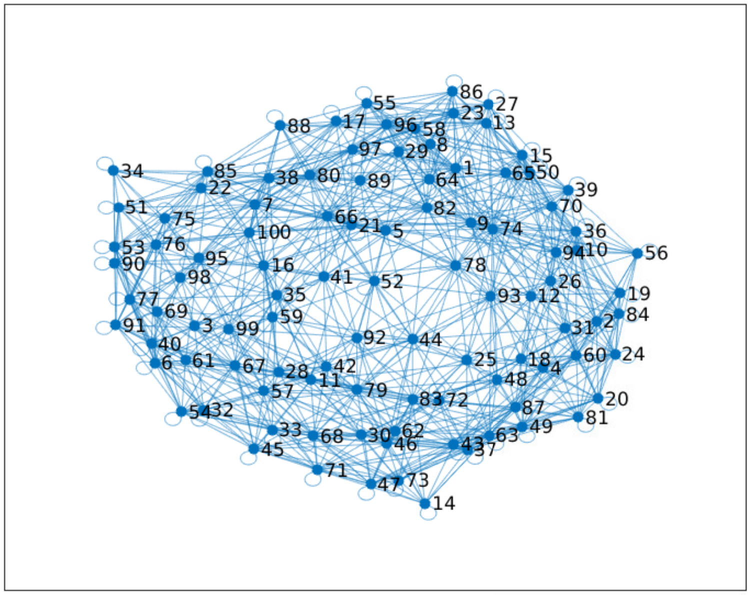
Solution: The system you get is: $$\begin{pmatrix} \mathbf{5} & \mathbf{10} & \mathbf{25} \\ \mathbf{1} & \mathbf{1} & \mathbf{1} \\ \mathbf{0} & \mathbf{10} & \mathbf{25} \end{pmatrix} \begin{pmatrix} \boldsymbol{x_n} \\ \boldsymbol{x_d} \\ \boldsymbol{x_q} \end{pmatrix} = \begin{pmatrix} \mathbf{145} \\ \mathbf{12} \\ \mathbf{125} \end{pmatrix}$$

where $\boldsymbol{x_n} = \#$ nickels, $\boldsymbol{x_d} = \#$ dimes, $\boldsymbol{x_q} = \#$ quarters

✍1 And the solution is: ?

➤ Pagerank of Webpages (21st cent AD)

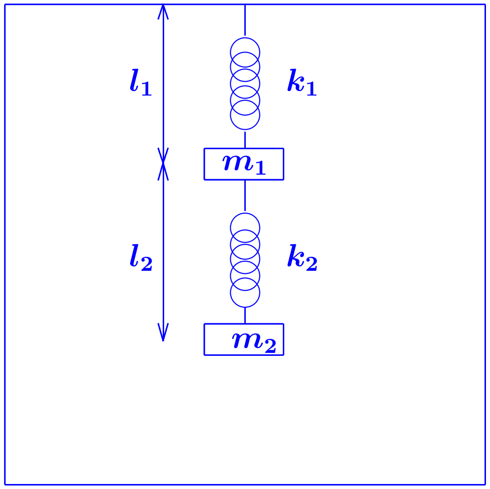
If one were to do a random walk from web page to web page, following each link on a given web page at random with equal likelihood, which are the pages to be encountered this way most often?



➤ Problem type: (homogeneous) Linear system. Eigenvector problem.

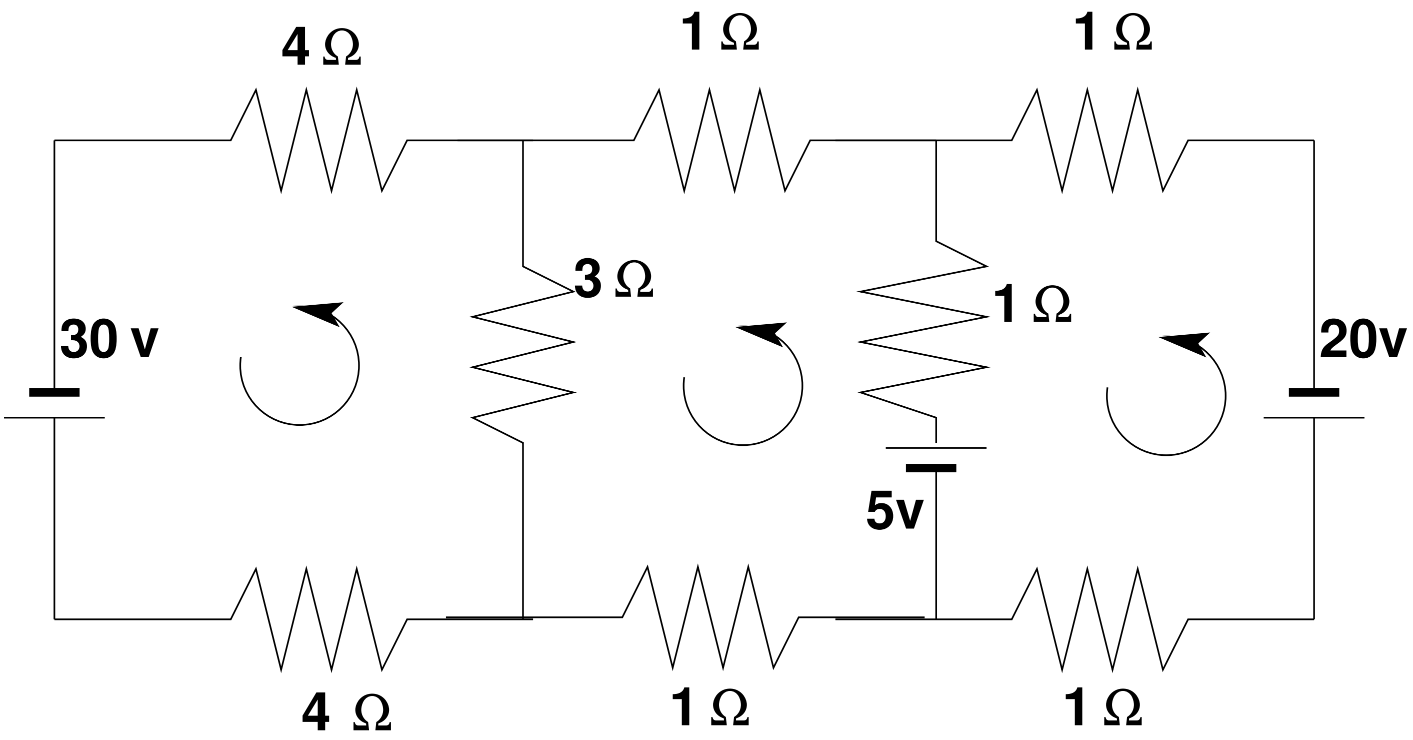
- Vibrations in mechanical systems. See: www.cs.umn.edu/~saad/eig_book_2ndEd.pdf

Problem: Determine the vibration modes of the mechanical system [to avoid resonance]. See details in Chapter 10 (sec. 10.2) of above reference.



- Problem type: Eigenvalue Problem

- Electrical circuits / Power networks, .. [Kirchhiff's voltage Law]



Problem: Determine the loop currents in a an electrical circuit - using Kirchhoff's Law $(\boldsymbol{V = RI})$

- Problem type: Linear System

Examples (cont.)

➤ Method of least-squares (inspired by first use of least squares ever, by Gauss around 1801)

A planet follows an elliptical orbit according to $ay^2 + bxy + cx + dy + e = x^2$ in cartesian coordinates. Given a set of noisy observations of (x, y) positions, compute a, b, c, d, e, and use to predict future positions of the planet. This least squares problem is nearly rank-deficient and hence very sensitive to perturbations in the observations.

➤ Problem type: Least-Squares system

Read Wikipedia's article on planet ceres:
`http://en.wikipedia.org/wiki/Ceres_(dwarf_planet)`

Dynamical systems and epidemiology

A set of variables that fill a vector $\boldsymbol{y}$ are governed by the equation

$$\boxed{\frac{d\boldsymbol{y}}{dt} = \boldsymbol{A}\boldsymbol{y}}$$

Determine $\boldsymbol{y}(\boldsymbol{t})$ for $\boldsymbol{t} > \boldsymbol{0}$, given $\boldsymbol{y}(\boldsymbol{0})$ [called 'orbit' of $\boldsymbol{y}$]

➤ Problem type: (Linear) system of ordinary differential equations.

Solution: $$\boldsymbol{y(t) = e^{tA} y(0)}$$

➤ Involves exponential of $\boldsymbol{A}$ [think Taylor series], i.e., a matrix function

- This is the simplest form of dynamical systems (linear).
- Consider the slightly more complex system:

$$\frac{d\boldsymbol{y}}{dt} = \boldsymbol{A}(\boldsymbol{y})\boldsymbol{y}$$

- Nonlinear. Requires ‘integration scheme’.
- Next: a little digression into our interesting times...

Example: The SIR model in epidemiology

A population of $\boldsymbol{N}$ individuals, with $\boldsymbol{N = S + I + R}$ where:

S Susceptible population. These are susecptible to being contaminated by others (not immune).

I Infectious population: will contaminate susceptible individuals.

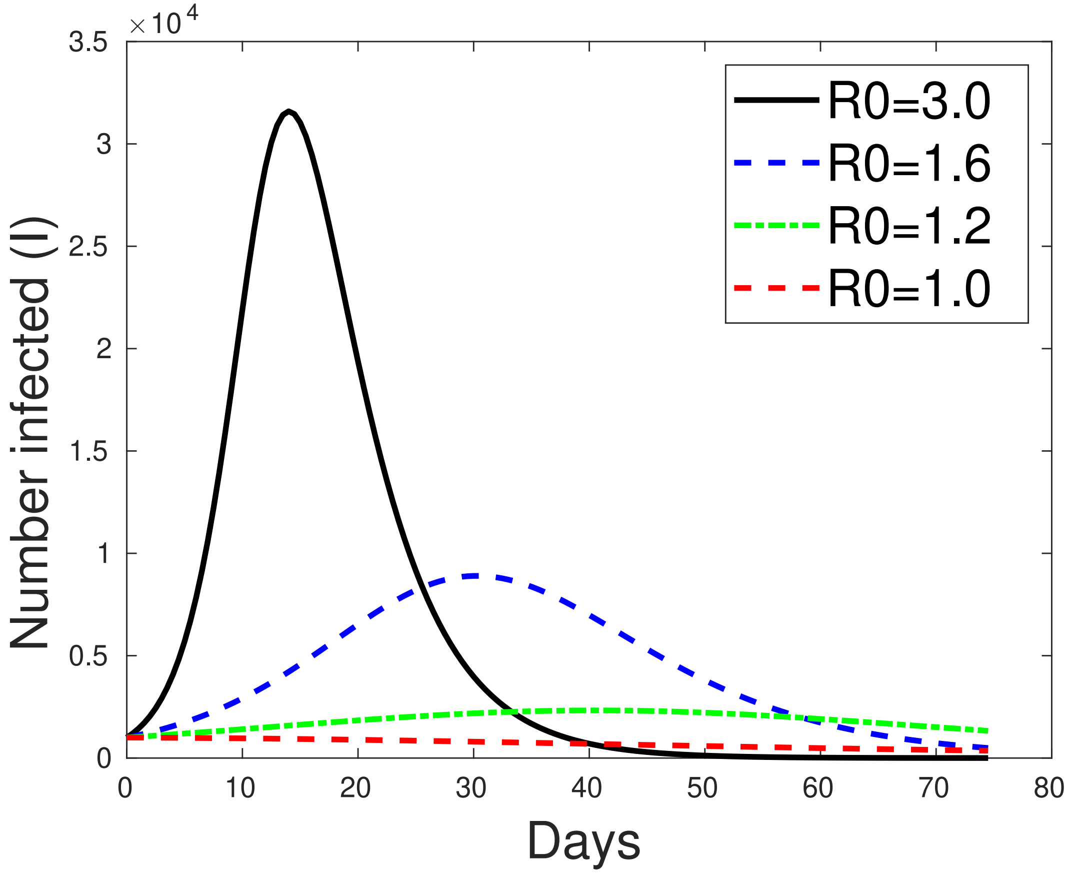
R 'Removed' population: either deceased or recovered. These will no longer contaminate others.

Three equations:

$$\boldsymbol{\frac{dS}{dt} = -\beta I S; \quad \frac{dI}{dt} = (\beta S - \mu) I; \quad \frac{dR}{dt} = \mu I}$$

$\boldsymbol{1/\mu}$ = infection period [e.g. 5 days].
$\boldsymbol{\beta = \mu R_0 / N}$ where $\boldsymbol{R_0}$ = reproduction number. Details in class.

➤ The importance of reducing R_0 (a.k.a. “social distancing”):



Typical Large-scale problem (e.g. Fluid flow)

Physical Model

↓

Nonlinear PDEs

↓

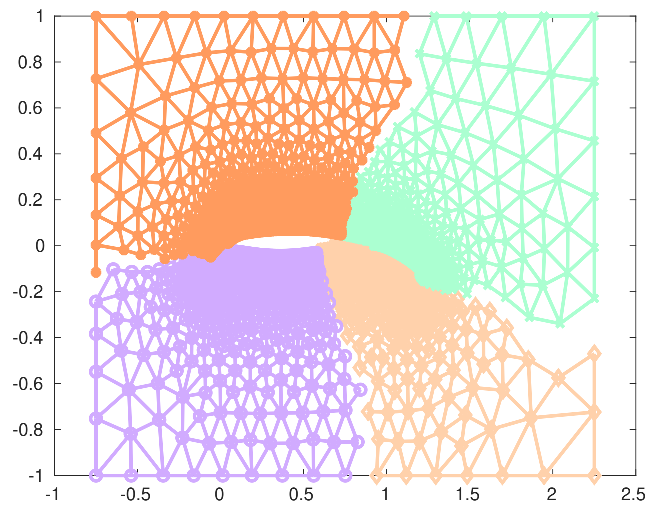
Discretization

↓

Linearization (Newton)

↓

Sparse Linear Systems $Ax = b$



Background in linear algebra

- Review vector spaces.

- A vector subspace of $\mathbb{R}^n$ is a subset of $\mathbb{R}^n$ that is also a real vector space. The set of all linear combinations of a set of vectors $G = \{a_1, a_2, \ldots, a_q\}$ of $\mathbb{R}^n$ is a vector subspace called the linear span of G,

- If the a_i's are linearly independent, then each vector of $\mathbf{span}\{G\}$ admits a unique expression as a linear combination of the a_i's. The set G is then called a *basis*.

✍2 Recommended reading: Sections 1.1 – 1.6 of

www.cs.umn.edu/~saad/eig_book_2ndEd.pdf

Matrices

➤ A real $m \times n$ matrix A is an $m \times n$ array of real numbers

$$a_{ij}, \quad i = 1, \ldots, m, \; j = 1, \ldots, n.$$

Set of $m \times n$ matrices is a real vector space denoted by $\mathbb{R}^{m \times n}$.

➤ Complex matrices defined similarly.

➤ A matrix represents a linear mapping between two vector spaces of finite dimension n and m:

$$\boxed{x \in \mathbb{R}^n \quad \longrightarrow \quad y = Ax \in \mathbb{R}^m}$$

➤ Recall: this mapping is linear [what does it mean?]

➤ Recall: Any linear mapping from $\mathbb{R}^n$ to $\mathbb{R}^m$ *is* a matrix vector product

Operations:

Addition: $\boldsymbol{C = A + B}$, where $\boldsymbol{A, B, C} \in \mathbb{R}^{\boldsymbol{m \times n}}$ and

$$\boldsymbol{c_{ij} = a_{ij} + b_{ij}, \quad i = 1, 2, \ldots m, \quad j = 1, 2, \ldots n.}$$

Multiplication by a scalar: $\boldsymbol{C = \alpha A}$, where

$$\boldsymbol{c_{ij} = \alpha \, a_{ij}, \quad i = 1, 2, \ldots m, \quad j = 1, 2, \ldots n.}$$

Multiplication by another matrix:

$$\boldsymbol{C = AB,}$$

where $\boldsymbol{A} \in \mathbb{R}^{\boldsymbol{m\times n}}$, $\boldsymbol{B} \in \mathbb{R}^{\boldsymbol{n\times p}}$, $\boldsymbol{C} \in \mathbb{R}^{\boldsymbol{m\times p}}$, and

$$\boldsymbol{c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}.}$$

Transposition: If $A \in \mathbb{R}^{m\times n}$ then its transpose is a matrix $C \in \mathbb{R}^{n\times m}$ with entries

$$c_{ij} = a_{ji}, i = 1, \ldots, n, \; j = 1, \ldots, m$$

Notation : A^T.

Transpose Conjugate: for complex matrices, the **transpose conjugate** matrix denoted by A^H is more relevant: $A^H = \bar{A}^T = \overline{A^T}$.

✍ 3 $(A^T)^T = ??$

✍ 4 $(AB)^T = ??$

✍ 5 $(A^H)^H = ??$

✍ 6 $(A^H)^T = ??$

✍ 7 $(ABC)^T = ??$

✍ 8 True/False: $(AB)C = A(BC)$

✍ 9 True/False: $AB = BA$

✍ 10 True/False: $AA^T = A^TA$

Review: Matrix-matrix and Matrix-vector producs

- Recall definition of $C = A \times B$: $c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}$.
- Recall what C represents [in terms of mappings]..
- Can do the product column-wise [Matlab notation used]:

$$C_{:,j} = \sum_{k=1}^{n} b_{kj} A_{:,k}$$

- Can do it row-wise:

$$C_{i,:} = \sum_{k=1}^{n} a_{ik} B_{k,:}$$

➤ Can do it as a sum of 'outer-product' matrices:

$$\boxed{C = \sum_{k=1}^{n} A_{:,k} B_{k,:}}$$

✍11 Verify all 3 formulas above..

✍12 Complexity? [number of multiplications and additions]

✍13 What happens to these 3 different approches to matrix-matrix multiplication when B has one column $(p = 1)$?

✍14 Characterize the matrices AA^T and A^TA when A is of dimension $n \times 1$.

Range and null space (for $A \in \mathbb{R}^{m\times n}$)

- Range: $\mathbf{Ran}(A) = \{Ax \mid x \in \mathbb{R}^n\}$ $\subseteq \mathbb{R}^m$
- Null Space: $\mathbf{Null(A)} = \{x \in \mathbb{R}^n \mid Ax = \mathbf{0}\ \}$ $\subseteq \mathbb{R}^n$
- Range $=$ linear span of the columns of A
- Rank of a matrix $\mathbf{rank}(A) = \mathbf{dim}(\mathbf{Ran}(A))$ $\leq n$

- Ran$(A) \subseteq \mathbb{R}^m$ $\rightarrow$ rank $(A) \le m \rightarrow$

$$\boxed{\text{rank }(A) \le \min\{m, n\}}$$

- rank (A) = number of linearly independent columns of A = number of linearly independent rows of A

- A is of full rank if $\mathbf{rank}(A) = \mathbf{min}\{m, n\}$. Otherwise it is rank-deficient.

Rank+Nullity theorem for an $m \times n$ matrix:

$$dim(Ran(A)) + dim(Null(A)) = n$$

Apply to A^T: $dim(Ran(A^T)) + dim(Null(A^T)) = m \rightarrow$

$$\text{rank}(A) + dim(Null(A^T)) = m$$

➤ Terminology:

- $dim(Null(A))$ is the Nullity of A [Another term: co-rank]

✍15 Show that $A \in \mathbb{R}^{m\times n}$ is of rank one iff [if and only if] there exist two nonzero vectors $u \in \mathbb{R}^m$ and $v \in \mathbb{R}^n$ such that

$$A = uv^T.$$

What are the eigenvalues and eigenvectors of A?

✍16 Is it true that

$$\mathbf{rank}(A) = \mathbf{rank}(\bar{A}) = \mathbf{rank}(A^T) = \mathbf{rank}(A^H) \ ?$$

✍17 Matlab exercise: explore the matlab function `rank`.

✍18 Matlab exercise: explore the matlab function `rref`.

✍19 Find the range and null space of the following matrix:

$$\begin{pmatrix} -1 & 1 & 0 \\ 1 & 2 & 3 \\ 1 & -2 & -1 \\ 2 & -1 & 1 \end{pmatrix}$$

Then verify your result with matlab.

Square matrices, matrix inversion, eigenvalues

- Square matrix: $n = m$, i.e., $A \in \mathbb{R}^{n\times n}$
- Identity matrix: square matrix with

$$a_{ij} = \begin{cases} 1 \text{ if } i = j \\ 0 \text{ otherwise} \end{cases}$$

- Notation: I.
- Property: $AI = IA = A$
- Inverse of A (when it exists) is a matrix C such that

$$AC = CA = I$$

Notation: A^{-1} .

Eigenvalues and eigenvectors

A complex scalar $\boldsymbol{\lambda}$ is called an eigenvalue of a square matrix $\boldsymbol{A}$ if there exists a nonzero vector $\boldsymbol{u}$ in $\mathbb{C}^n$ such that $\boldsymbol{Au} = \boldsymbol{\lambda u}$. The vector $\boldsymbol{u}$ is called an eigenvector of $\boldsymbol{A}$ associated with $\boldsymbol{\lambda}$. The set of all eigenvalues of $\boldsymbol{A}$ is the 'spectrum' of $\boldsymbol{A}$. Notation: $\boldsymbol{\Lambda(A)}$.

- λ is an eigenvalue iff the columns of $A - \lambda I$ are linearly dependent.

- ... equivalent to saying that its rows are linearly dependent. So: there is a nonzero vector w such that

$$w^H(A - \lambda I) = 0$$

- w is a left eigenvector of A (u= right eigenvector)

- λ is an eigenvalue iff $\boxed{\det(A - \lambda I) = 0}$

Eigenvalues/vectors

- An eigenvalue is a root of the Characteristic polynomial: $\boldsymbol{p_A(\lambda) = \det(A - \lambda I)}$

- So there are n eigenvalues (counted with their multiplicities).

- The multiplicity of these eigenvalues as roots of p_A are called algebraic multiplicities.

- The geometric multiplicity of an eigenvalue λ_i is the number of linearly independent eigenvectors associated with λ_i.

- Geometric multiplicity is $\leq$ algebraic multiplicity.

- An eigenvalue is simple if its (algebraic) multiplicity is one. It is semi-simple if its geometric and algebraic multiplicities are equal.

➤ Two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ are similar if there exists a nonsingular matrix $\boldsymbol{X}$ such that $\boldsymbol{A = XBX^{-1}}$

✍21 Eigenvalues of $\boldsymbol{A}$ and $\boldsymbol{B}$ are the same. What about eigenvectors?

➤ Note: $\boldsymbol{A}$ and $\boldsymbol{B}$ represent the same mapping using 2 different bases.

Fundamental Problem: Given $\boldsymbol{A}$, find $\boldsymbol{X}$ so that $\boldsymbol{B}$ has a simpler structure (e.g., diagonal) $\longrightarrow$ Eigenvalues of $\boldsymbol{B}$ easier to compute

Definition: $\boldsymbol{A}$ is diagonalizable if it is similar to a diagonal matrix

➤ We will revisit these notions later in the semester

✍22 Given a polynomial $\boldsymbol{p(t)}$ how would you define $\boldsymbol{p(A)}$?

✍23 Given a function $\boldsymbol{f(t)}$ (e.g., $\boldsymbol{e^t}$) how would you define $\boldsymbol{f(A)}$? [Leave the full justification for next chapter]

✍24 If A is nonsingular what are the eigenvalues/eigenvectors of A^{-1}?

✍25 What are the eigenvalues/eigenvectors of A^k for a given integer power k?

✍26 What are the eigenvalues/eigenvectors of $p(A)$ for a polynomial p?

✍27 What are the eigenvalues/eigenvectors of $f(A)$ for a function f? [Diagonalizable case]

✍28 For two $n \times n$ matrices A and B are the eigenvalues of AB and BA the same?

✍29 Review the Jordan canonical form. [Short description in sec. 1.8.2 of My eigenvalue book] Define the eigenvalues, and eigenvectors from the Jordan form.

- Spectral radius = The maximum modulus of the eigenvalues

$$\rho(A) = \max_{\lambda \in \lambda(A)} \ |\lambda|.$$

- Trace of A = sum of diagonal elements of A.

$$\text{Tr}\,(A) = \sum_{i=1}^{n} a_{ii}.$$

- $\mathbf{tr}(A)$ = sum of all the eigenvalues of A counted with their multiplicities.

➤ Recall that $\det(A)$ = product of all the eigenvalues of A counted with their multiplicities.

✍30 Trace, spectral radius, and determinant of

$$A = \begin{pmatrix} 2 & 1 \\ 3 & 0 \end{pmatrix}.$$

Types of (square) matrices

- Symmetric $A^T = A$. • Skew-symmetric $A^T = -A$.
- Hermitian $A^H = A$. • Skew-Hermitian $A^H = -A$.

- Normal $A^H A = AA^H$.

- Nonnegative $a_{ij} \geq 0, \ i,j = 1, \ldots, n$

- Similarly for nonpositive, positive, and negative matrices

- Unitary $Q^H Q = I$. (for complex matrices)

[Note: Common useage restricts this definition to complex matrices. An *orthogonal matrix* is a unitary *real* matrix – not very natural]

- Orthogonal $\boldsymbol{Q^TQ = I}$ [orthonormal columns]

[I will sometimes call unitary matrix a square matrix with orthonormal columns, regardless on whether it is real or complex]

➤ The term "orthonormal" matrix is rarely used.

✍31 What is the inverse of a unitary (complex) or orthogonal (real) matrix?

✍32 What can you say about the diagonal entries of a skew-symmetric (real) matrix?

✍33 What can you say about the diagonal entries of a Hermitian (complex) matrix?

✍34 What can you say about the diagonal entries of a skew-Hermitian (complex) matrix?

✍35 Which matrices of the following type are also normal: real symmetric, real skew-symmetric, Hermitian, skew-Hermitian, complex symmetric, complex skew-symmetric matrices.

✍36 Find all real 2×2 matrices that are normal.

✍37 Show that a triangular matrix that is normal is diagonal.

Matrices with structure

- Diagonal $a_{ij} = 0$ for $j \neq i$. Notation :

$$A = \mathrm{diag}\,(a_{11}, a_{22}, \ldots, a_{nn})\,.$$

- Upper triangular $a_{ij} = 0$ for $i > j$.
- Lower triangular $a_{ij} = 0$ for $i < j$.
- Upper bidiagonal $a_{ij} = 0$ for $j \neq i$ or $j \neq i + 1$.
- Lower bidiagonal $a_{ij} = 0$ for $j \neq i$ or $j \neq i - 1$.
- Tridiagonal $a_{ij} = 0$ when $|i - j| > 1$.

- Banded $a_{ij} \neq 0$ only when $i - m_l \leq j \leq i + m_u$, 'Bandwidth' $= m_l + m_u + 1$.

- Upper Hessenberg $a_{ij} = 0$ when $i > j + 1$. Lower Hessenberg matrices can be defined similarly.

- Outer product $A = uv^T$, where both u and v are vectors.

- Block tridiagonal generalizes tridiagonal matrices by replacing each nonzero entry by a square matrix.

Special matrices

Vandermonde :

➤ Given a column of entries $[x_0, x_1, \cdots, x_n]^T$ put its (component-wise) powers into the columns of a matrix V:

$$V = \begin{pmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^2 \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^n \end{pmatrix}$$

✍38 Try the matlab function *vander*

✍39 What does the matrix-vector product Va represent?

✍40 Interpret the solution of the linear system $Va = y$ where a is the unknown. Sketch a 'fast' solution method based on this.

Toeplitz :

- Entries are constant along diagonals, i.e., $a_{ij} = r_{j-i}$.
- Determined by $m + n - 1$ values r_{j-i}.

$$T = \underbrace{\begin{pmatrix} r_0 & r_1 & r_2 & r_3 & r_4 \\ r_{-1} & r_0 & r_1 & r_2 & r_3 \\ r_{-2} & r_{-1} & r_0 & r_1 & r_2 \\ r_{-3} & r_{-2} & r_{-1} & r_0 & r_1 \\ r_{-4} & r_{-3} & r_{-2} & r_{-1} & r_0 \end{pmatrix}}_{Toeplitz}$$

- Toeplitz systems $(m = n)$ can be solved in $O(n^2)$ ops.
- The whole inverse (!) can be determined in $O(n^2)$ ops.

✍$_{41}$ Explore *toeplitz(c,r)* in matlab.

Hankel : Entries are constant along anti-diagonals, i.e., $a_{ij} = h_{j+i-1}$. Determined by $m+n-1$ values h_{j+i-1}.

$$H = \underbrace{\begin{pmatrix} h_1 & h_2 & h_3 & h_4 & h_5 \\ h_2 & h_3 & h_4 & h_5 & h_6 \\ h_3 & h_4 & h_5 & h_6 & h_7 \\ h_4 & h_5 & h_6 & h_7 & h_8 \\ h_5 & h_6 & h_7 & h_8 & h_9 \end{pmatrix}}_{Hankel}$$

✍42 Explore *hankel(c,r)* in matlab.

Circulant : Entries in a row are cyclically right-shifted to form next row. Determined by n values.

$$C = \underbrace{\begin{pmatrix} v_1 & v_2 & v_3 & v_4 & v_5 \\ v_5 & v_1 & v_2 & v_3 & v_4 \\ v_4 & v_5 & v_1 & v_2 & v_3 \\ v_3 & v_4 & v_5 & v_1 & v_2 \\ v_2 & v_3 & v_4 & v_5 & v_1 \end{pmatrix}}_{Circulant}$$

✍43 How can you generate a circulant matrix in matlab?

✍44 If C is circulant (real) and symmetric, what can be said about the v_i's?

Sparse matrices

➤ Matrices with very few nonzero entries – so few that this can be exploited.

➤ Many of the large matrices encountered in applications are sparse.

➤ Main idea of “sparse matrix techniques” is not to represent the zeros.

➤ This will be covered in some detail at the end of the course.