About this class

• Instructor and Teaching Assistant:

➤ Me: Yousef Saad
➤ TA: Ivan Radkevich

• Course title: “Computational Aspects of Matrix Theory”

... Class aims to cover:

“Everything that you can learn in one semester about computations that involve matrices, from theory, to algorithms, matlab/Python implementations, and applications.”

➤ Subject is at the core of *most* disciplines requiring numerical computing..

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

1-2
Who is in this class today?

Out of ≈ 62 [Including Unite]

Ugrad  (Total = 23)
- BS/BA-CS: 18
- BS-DS: 3
- BS-math: 1
- B. Comp. Eng: 1

Grad:  (Total = 39)
- ME: 19
- CS: 12
- ECE: 5
- AEM: 2
- IndSyst: 1
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:

- Lecture notes, syllabus, schedule, a matlab/Python folder, and some basic information, can be found here:
  
  http://www-users.cse.umn.edu/~saad/csci5304

- Everything else will be found on Canvas: Homeworks, exam/quizzes info, your grades, etc..

  ➢ The two sites have links that point to each other
Please Note:

Homeworks, tests, and their solutions are copyrighted

- Solutions of HWs and tests are 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 \( \text{1} \) indicates suggested easy exercises or questions sometimes solved in class.

- Each set will include a supplement with solutions to some of these exercises + possibly additional notes/comments (an evolving document).
Occasional in-class practice exercises

- Posted in advance – [Canvas]
- Do them before class. No need to turn in anything.
- ... will be discussed - typically at beginning of class
Matlab and/or Python

➢ You will need to use matlab or Python+numpy for testing algorithms.

➢ In the past: only matlab -

➢ For those interested you can turn in your codes for assignments in Python+numpy. + demos often in both

➢ Some documentation is posted in the (class) matlab folder – No documentation for Python

➢ Important: I post the matlab diaries used for the demos (if any)...

➢ ... something similar for Python [under IPython]

- If you do not know matlab at all and have difficulties with it - and you do not know python - talk to me or the TA at office hours. You may need is some initial help to get you started with matlab.
Final remarks on lecture notes

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, ...)

Classes are all in-person. Zoom will be used only when necessary.

If you are sick *please* do not come to class [there is really no need to] !!!!

If I get sick - I will schedule the class on Zoom [Assuming I can!] –

Office hours: See posted information for details (schedule, zoom option, etc.)
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, structured 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, ...
Example: Linear Systems

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:

\[
\begin{pmatrix}
5 & 10 & 25 \\
1 & 1 & 1 \\
0 & 10 & 25
\end{pmatrix}
\begin{pmatrix}
x_n \\
x_d \\
x_q
\end{pmatrix}
=
\begin{pmatrix}
145 \\
12 \\
125
\end{pmatrix}
\]

where \( x_n = \#\) nickels, \( x_d = \#\) dimes, \( x_q = \#\) quarters

And the solution is: ?
Example: Pagerank

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.
Example: Vibrations


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
Problem: Determine the loop currents in an electrical circuit - using Kirchhoff’s Law \( (V = RI) \)

➢ Problem type: Linear System

Example: Electrical circuits / Power networks, Kirchhoff’s Law
Example: Method of least-squares

Inspired by first use of least squares ever - by Gauss around 1801

A planet follows an elliptical orbit according to $a y^2 + b x y + c x + d y + e = x^2$ 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)
Example: Dynamical systems and epidemiology

A set of variables that fill a vector $y$ are governed by the equation

$$\frac{dy}{dt} = Ay$$

Determine $y(t)$ for $t > 0$, given $y(0)$ [called ‘orbit’ of $y$]

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

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

Involves exponential of $A$ [think Taylor series], i.e., a matrix function
This is the simplest form of dynamical systems (linear).

Consider the slightly more complex system:

\[
\frac{dy}{dt} = A(y)y
\]

Nonlinear. Requires ‘integration scheme’.

Next: a little digression into our interesting times...
The SIR model in epidemiology

A population of $N$ individuals, with $N = S + I + R$ where:

- **S** Susceptible population. These are susceptible 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:

\[
\frac{dS}{dt} = -\beta IS; \quad \frac{dI}{dt} = (\beta S - \mu)I; \quad \frac{dR}{dt} = \mu I
\]

$1/\mu =$ infection period [e.g. 5 days].

$\beta = \mu R_0/N$ where $R_0 =$ reproduction number.
The importance of reducing $R_0$ (a.k.a. “social distancing”):
General Problems in Numerical Linear Algebra (dense & sparse)

- Linear systems: $Ax = b$. Often: $A$ is large and sparse
- Least-squares problems $\min ||b - Ax||_2$
- Eigenvalue problem $Ax = \lambda x$. Several variations -
  - SVD .. and
  - ... Low-rank approximation
- Tensors and low-rank tensor approximation
- Matrix equations: Sylvester, Lyapunov, Riccati, ..
- Nonlinear equations – acceleration methods
- Matrix functions and applications
- Many many more ...
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 $\text{span}\{G\}$ admits a unique expression as a linear combination of the $a_i$’s. The set $G$ is then called a *basis*.

Recommended reading: Sections 1.1 – 1.6 of

A real \( m \times n \) matrix \( A \) is an \( m \times n \) array of real numbers

\[
a_{ij}, \quad i = 1, \ldots, m, \quad j = 1, \ldots, n.
\]

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 \( m \):

\[
\begin{align*}
x \in \mathbb{R}^n & \quad \longrightarrow \quad y = Ax \in \mathbb{R}^m
\end{align*}
\]

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

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

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

**Multiplication by a scalar:** \( C = \alpha A \), where

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

**Multiplication by another matrix:** \( C = AB \),

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

\[
c_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj}.
\]
**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: \( A^H = \bar{A}^T = \bar{A}^T \).

1. \((A^T)^T = ??\)
2. \((AB)^T = ??\)
3. \((A^H)^H = ??\)
4. \((A^H)^T = ??\)
5. \((ABC)^T = ??\)
6. True/False: \((AB)C = A(BC)\)
7. True/False: \(AB = BA\)
8. True/False: \(AA^T = A^TA\)

GvL: 1.1–1.3, 2.1. – Background
Review: Matrix-matrix and Matrix-vector products

- 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:

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

Verify all 3 formulas above.

Complexity? [number of multiplications and additions]

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

Characterize the matrices $AA^T$ and $A^T A$ when $A$ is of dimension $n \times 1$. 
Matrix-vector product: computing $y = Ax$

**Column-form:**
Linear combination of columns $A(:, j)$ with coefficients $x_j$
yields $y$

**Row-form:**
Dot product of $A(i, :)$ and $x$
gives $y_i$
**Kronecker products of matrices**

This is a special product of matrices that can be quite useful in some situations.

**Definition**

For $A \in \mathbb{R}^{m \times n}$, $B \in \mathbb{R}^{p \times q}$ define:

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

(A matrix of size $(mp) \times (nq)$).

In Matlab: `kron (A, B)`

Note that the dimensions $m, n, p, q$, can be any ($> 0$) integers.

Show that for 2 vectors $u, v$ we have $v^T \otimes u = uv^T$

The Kronecker sum of matrices also arises in some applications. If $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{m \times m}$ then their Kronecker sum is: $A \oplus B = A \otimes I_{m,m} + I_{n,n} \otimes B$
Range and null space (for $A \in \mathbb{R}^{m \times n}$)

- **Range**: $\text{Ran}(A) = \{Ax | x \in \mathbb{R}^n\} \subseteq \mathbb{R}^m$

- **Null Space**: $\text{Null}(A) = \{x \in \mathbb{R}^n | Ax = 0\} \subseteq \mathbb{R}^n$

- Range = linear span of the columns of $A$

- **Rank of a matrix**: $\text{rank}(A) = \text{dim}(\text{Ran}(A)) \leq n$

- $\text{Ran}(A) \subseteq \mathbb{R}^m \rightarrow \text{rank } (A) \leq m \rightarrow$
  $$\text{rank } (A) \leq \min\{m, n\}$$

- $\text{rank } (A) = \text{number of linearly independent columns of } A = \text{number of linearly independent rows of } A$

- **$A$ is of full rank** if $\text{rank}(A) = \min\{m, n\}$. Otherwise it is rank-deficient.
Rank+Nullity theorem for an $m \times n$ matrix:

$$\dim(\text{Ran}(A)) + \dim(\text{Null}(A)) = n$$

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

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

Terminology:

- $\dim(\text{Null}(A))$ is the Nullity of $A$ [Another term: co-rank]
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 \( A \)?

Is it true that
\[
\text{rank}(A) = \text{rank}(\bar{A}) = \text{rank}(A^T) = \text{rank}(A^H)?
\]

Matlab exercise: explore the matlab function \texttt{rank}.

Matlab exercise: explore the matlab function \texttt{rref}.

No \texttt{rref} function in numpy – [see sympy]
Find the range and null space of the following matrix:

Verify your result with matlab [hint: use null, rank, rref]
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: $A^{-1}$.
A complex scalar $\lambda$ is called an eigenvalue of a square matrix $A$ if there exists a nonzero vector $u$ in $\mathbb{C}^n$ such that $Au = \lambda u$. The vector $u$ is called an eigenvector of $A$ associated with $\lambda$. The set of all eigenvalues of $A$ is the ‘spectrum’ of $A$. Notation: $\Lambda(A)$.

- $\lambda$ 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 $A$ ($u = \text{right eigenvector}$)
- $\lambda$ is an eigenvalue iff $\det(A - \lambda I) = 0$
An eigenvalue is a root of the Characteristic polynomial:

$$p_A(\lambda) = \det(A - \lambda I)$$

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 $\lambda_i$ is the number of linearly independent eigenvectors associated with $\lambda_i$.

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 $A$ and $B$ are similar if there exists a nonsingular matrix $X$ such that 
\[ A = XBX^{-1} \]

Eigenvalues of $A$ and $B$ are the same. What about eigenvectors?

Note: $A$ and $B$ represent the same mapping using 2 different bases.

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

Definition: $A$ is diagonalizable if it is similar to a diagonal matrix

We will revisit these notions later in the semester

Given a polynomial $p(t)$ how would you define $p(A)$?

Given a function $f(t)$ (e.g., $e^t$) how would you define $f(A)$? [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?

- Spectral radius = The maximum modulus of the eigenvalues
  \[ \rho(A) = \max_{\lambda \in \sigma(A)} |\lambda|. \]

- Trace of \( A \) = sum of diagonal elements of \( A \).
  \[ \text{Tr}(A) = \sum_{i=1}^{n} a_{ii}. \]

- \( \text{tr}(A) \) = 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.

- 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 usage restricts this definition to complex matrices. An *orthogonal matrix* is a unitary *real* matrix – not very natural]

- Orthogonal \( Q^T Q = I \) [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.
What is the inverse of a unitary (complex) or orthogonal (real) matrix?

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

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

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

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

Find all real $2 \times 2$ matrices that are normal.

Show that a triangular matrix that is normal is diagonal.
**Matrices with structure**

- **Diagonal** \( a_{ij} = 0 \) for \( j \neq i \). Notation:
  \[
  A = \text{diag} (a_{11}, a_{22}, \ldots, a_{nn}).
  \]

- **Upper triangular** \( a_{ij} = 0 \) for \( i > j \).

- **Lower triangular** \( a_{ij} = 0 \) for \( i < j \).

- **Upper bidiagonal** \( a_{ij} = 0 \) for \( j \neq i \) and \( j \neq i + 1 \).

- **Lower bidiagonal** \( a_{ij} = 0 \) for \( j \neq i \) and \( j \neq i - 1 \).

- **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.
**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 \(V\):

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

Try the matlab function `vander`

What does the matrix-vector product \(Va\) represent?

Interpret the solution of the linear system \(Va = y\) where \(a\) is the unknown. Sketch a ‘fast’ solution method based on this.
Entries are constant along diagonals, i.e., $a_{ij} = r_{j-i}$.

Determined by $m + n - 1$ values $r_{j-i}$.

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

Toeplitz systems ($m = n$) can be solved in $O(n^2)$ ops.

The whole inverse (!) can be determined in $O(n^2)$ ops.

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 $h_{j+i-1}$.

Explore `hankel(c,r)` in matlab.
A circulant matrix is a matrix in which each row vector is rotated one element to the right relative to the preceding row vector. Entries in a row are cyclically right-shifted to form the next row. Determined by $n$ values.

$$C = \begin{pmatrix}
c_1 & c_2 & c_3 & c_4 & c_5 \\
c_5 & c_1 & c_2 & c_3 & c_4 \\
c_4 & c_5 & c_1 & c_2 & c_3 \\
c_3 & c_4 & c_5 & c_1 & c_2 \\
c_2 & c_3 & c_4 & c_5 & c_1\
\end{pmatrix}$$

How can you generate a circulant matrix in MATLAB?

If $C$ is circulant (real) and symmetric, what can be said about the $c_i$'s?

A simple and important circulant matrix is the up-shift matrix $S_5$:

$$S_5 = \begin{bmatrix}
0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 \\
1 & 0 & 0 & 0 & 0
\end{bmatrix}$$
What is the result of multiplying \( S_n \) by a vector? What are the powers of \( S_n \)? What is the inverse of \( S_n \)?

Show that

\[ C = c_1 I + c_2 S_n + c_3 S^2_n + \cdots + c_n S^{n-1}_n \]

As a result show that all circulant matrices of the same size commute.

(Continuation) Use the result of the previous exercise to show that the product of two circulant matrices is circulant.
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.