Lecture 1: Linear Algebra, S. Ch 1

n-vector **x** column vector $(x_1,x_2,\ldots,x_n)^T$ in $\boldsymbol{R}^n$ (or $\boldsymbol{C}^n$);
m x n matrix $\mathbf{A} = (a_{ij})$, i row index, j column index

A as linear operator: $\boldsymbol{R}^n \to \boldsymbol{R}^m$
$\mathbf{A} = (\mathbf{a}_1 \quad \mathbf{a}_2 \quad \ldots \quad \mathbf{a}_n), \mathbf{Ax} = x_1\mathbf{a}_1 + x_2\mathbf{a}_2 + \ldots + x_n\mathbf{a}_n$
linear combination is in the *column space* $V = R(\mathbf{A})$, spanned by the columns of **A**.

rank(**A**) = dim(V) = max. number of linearly independent columns

Matrix multiplication: $\boldsymbol{R}^n \xrightarrow[\text{mxn}]{\mathbf{A}} \boldsymbol{R}^m \xrightarrow[\text{kxm}]{\mathbf{B}} \boldsymbol{R}^k \qquad \boldsymbol{R}^n \xrightarrow[\text{kxn}]{\mathbf{C}} \boldsymbol{R}^k$

Cx = **B**(**Ax**) = (**BA**) **x** *Matrix multiplication is associative*

Different views:

1. $c_{ij} = \sum_{s=1}^{m} b_{is}a_{sj} = B(i,:) * A(:,j)$, scalar product of row i of **B** with column j of **A**

2. **C**(:,j) = **B**(:,1) $a_{1,j}$ + **B**(:,2) $a_{2,j}$ +…+ **B**(:,m) $a_{m,j}$, lin.comb of columns of **B**
Column space of **C** *no larger than column space of* **B**

3. **C** = **B**(:,1)***A**(1,:) + **B**(:,2)***A**(2,:) +…+ **B**(:,m)***A**(m,:)
"Outer product", **C** as sum of rank-one matrices

Ex.

$\mathbf{A} = \mathbf{I} + \mathbf{uv}^T$.
Solve **Ax** = **b**. How many solutions? Formula for $\mathbf{A}^{-1}$?

Main problems of Numerical Linear Algebra

I. Solve linear system

$$\mathbf{Ax} = \mathbf{b}, \mathbf{A}\ m\text{x}n, \text{ often } m = n$$

II. Eigenvalue problem: Find eigenvector(s) **x** and complex eigenvalue(s) λ

$$\mathbf{Ax} = \lambda\mathbf{x}$$

III. Optimization

1. "Linear programming"

$$\min_{\mathbf{x}} \mathbf{c}^T\mathbf{x} \text{ subject to } \mathbf{Ax} \geq \mathbf{b}, \mathbf{x} \geq 0$$

2. Least squares approximation

$$\min_{\mathbf{x}} \sum w_i r_i^2, \mathbf{r} = \mathbf{Ax} - \mathbf{b}$$

3. Energy minimization - equilibrium

$$\min_{\mathbf{x}} \frac{1}{2} \sum_{i,j} a_{ij} x_i x_j - \sum_i b_i x_i = \min_{\mathbf{x}} \frac{1}{2} \mathbf{x}^T \mathbf{Ax} - \mathbf{b}^T \mathbf{x}$$

Properties of **A**:

- Symmetry
- Sparsity
- Condition / singularity

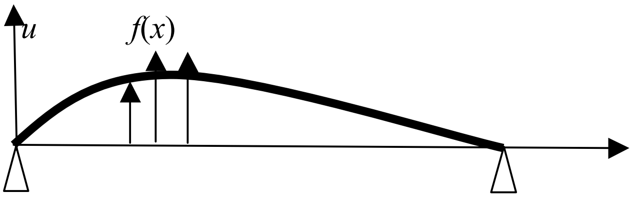
Sources of linear systems

- Discretization of differential equations
  - Finite differences & - volumes
  - Finite Elements
  - Spectral / Pseudo-spectral
- Network models & graphs
  - Electric circuits, mechanical trusses, hydraulic systems
  - Markov chains

Ex. The K, B, T, C –matrices of S. Ch 1

Transversally loaded string, small displacements

$$-S\frac{d^2u}{dx^2} = f(x), u(0) = u(L) = 0$$



Difference approximations

$$\Delta x \cdot u'(x_j) = \begin{cases} u_{j+1} - u_j + O(\Delta x^2) \\ (u_{j+1} - u_{j-1})/2 + O(\Delta x^3) \\ u_j - u_{j-1} + O(\Delta x^2) \end{cases}$$

$$\begin{cases} \begin{array}{ccccccccc} \hline 0 & \dots & 0 & 0 & -1 & 1 & 0 & \dots & 0 \\ \hline 0 & \dots & 0 & -1/2 & 0 & 1/2 & 0 & \dots & 0 \\ \hline 0 & \dots & 0 & -1 & 1 & 0 & 0 & \dots & 0 \\ \hline \end{array} \end{cases} * \begin{pmatrix} \dots \\ u_{j-1} \\ u_j \\ u_{j+1} \\ \dots \end{pmatrix}$$

The string:

$$-\Delta x^2 \cdot u''(x_j) = -u_{j-1} + 2u_j - u_{j+1} + O(\Delta x^4):$$

$$-u_{j-1} + 2u_j - u_{j+1} = \Delta x^2 f(x_j)/S, j = 1,2,\dots,n, \quad u_0 = u_{n+1} = 0$$

$$\mathbf{K}_n\mathbf{u} = \mathbf{f}$$

(Matlab): `u = K\f;`

Solution by Gaussian elimination

Step *k* subtracts a multiple of row *k* (also in RHS) from rows *k*+1,*k*+2,…,*n*,
“elementary row operation”

Leaves first *k* rows unchanged.

Preserves solution set.

Ex. $\mathbf{K}_3$

$$\begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix}\mathbf{u} = \begin{pmatrix} a \\ b \\ c \end{pmatrix} \Rightarrow \underbrace{\begin{pmatrix} 2 & -1 & 0 \\ 0 & 3/2 & -1 \\ 0 & 0 & 4/3 \end{pmatrix}}_{\mathbf{U}}\mathbf{u} = \underbrace{\begin{pmatrix} a \\ b + 1/2a \\ c + 1/3a + 2/3b \end{pmatrix}}_{\mathbf{L}^{-1}\mathbf{b}}, \mathbf{L}^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ 1/2 & 1 & 0 \\ 1/3 & 2/3 & 1 \end{pmatrix}$$

$$\mathbf{Uu} = \mathbf{L}^{-1}\mathbf{b}$$

Ex. Find $\mathbf{L} = (\mathbf{L}^{-1})^{-1}$ by GE:

- Inverse of triangular matrix is triangular
- pivots 2, 3/2, 4/3 all positive
- multipliers: -1/2, (0), -2/3 are subdiagonal elements in **L**.

There follows

- ***If there are n non-zero pivots***, unique solution for any RHS ;
- GE produces factorization **A** = **LU**,
    - $l_{ii} = 1$, l_{ij}, $i > j$, are the multipliers,
    - u_{ii} = pivots

GE with row interchanges:
If zero pivot in step k: exchange rows k and s where the element is non-zero. If no non-zero element in pivot column, matrix is singular.
"Partial pivoting": Take s for absolutely largest element in pivot column. Then

$$\left|l_{ij}\right| \le 1, i > j$$

There follows:
A matrix **A** is non-singular if and only if admits a factorization

$$\mathbf{PA} = \mathbf{LU}$$

with **P** a row reordering matrix, and $\left|l_{ij}\right| \le 1, i > j$.

Ex. Computation of determinants

$$\det \mathbf{A} = S \det \mathbf{U} = S \cdot \text{product of all pivots}, S = \det \mathbf{P} = \pm 1$$

Theorem
A is non-singular if and only if det**A** is non-zero.

Symmetric matrices
Observation:
A step of GE without row interchanges preserves symmetry:

$$\mathbf{A}(2:n,2:n) := \mathbf{A}(2:n,2:n) - \mathbf{A}(2:n,1) * \mathbf{A}(1,2:n) / a_{11}$$

and the two vectors are equal because of the symmetry of **A.** It follows that column k of **L** equals row k of **U**, divided by the kth pivot u_{kk}.:

Theorem
1. If GE can be carried out without row interchanges on the symmetric matrix **A**,

$$\mathbf{A} = \mathbf{LU} = \mathbf{LDL}^T,\ \mathbf{D} = \text{diag}(\mathbf{U}).$$

2. If additionally the pivots are positive, we may write

$$\mathbf{A} = \mathbf{L}_1\mathbf{L}_1^T,\ \mathbf{L}_1 = diag(\sqrt{u_{ii}}) \cdot \mathbf{L}$$

the Cholesky-factorization.

In many important cases it is known that **A** is "SPD" = symmetric and positive definite, i.e.

$$\mathbf{x}^T\mathbf{Ax} > 0 \text{ for all non-zero } \mathbf{x}$$

Ex.
The normal equations

$$\mathbf{A}^T\mathbf{Ax} = \mathbf{A}^T\mathbf{b}$$

of a least squares problem are SPD if the columns of **A** are linearly independent. The symmetry is obvious, as the semidefinitiveness:

$$\mathbf{x}^T\mathbf{A}^T\mathbf{Ax} = \mathbf{y}^T\mathbf{y} = \sum y_i^2 \ge 0, = 0 \text{ only if } \mathbf{y} \text{ is the zero - vector,}$$

$$\mathbf{Ax} = \mathbf{y}$$

But for **y** to be the zero-vector, so must **x** be all zeros, since the only linear combination of the columns of **A** has vanishing coefficients. Thus, the quadratic form vanishes only when **x** is 0.

Theorem:
A SPD matrix has all eigenvalues real and positive.
The first step is to establish reality of eigenvalues and eigenvectors of a real symmetric matrix. Define the *Hermitian transpose* $\mathbf{A}^H$ of a matrix as the complex conjugate of the transpose, $\mathbf{A}^H(i,j) = conj(\mathbf{A}(j,i))$. Note that

$$x^H x = \sum_i x_i conj(x_i) = \sum |x_i|^2 \geq 0$$

for a vector with real or complex elements.
Proof: 1. real … 2. positive…

Ex: Show that a symmetric matrix with positive pivots is positive definite.
Hint: Use $\mathbf{A} = \mathbf{L}\mathbf{L}^T$.
The converse is also true, but slightly harder to show, so

Theorem
A SPD matrix **A** can be factored $\mathbf{A} = \mathbf{LDL}^T$ without row exchanges and the pivots d_{ii} are positive.

Proof
Look at one step. The complete proof follows by induction.
1) The a_{11} element must be positive, because $a_{11} = \mathbf{e}_1^T\mathbf{A}\mathbf{e}_1 > 0$. We call it c.
2)

$$0 < \mathbf{x}^T\mathbf{A}\mathbf{x} = \begin{pmatrix} z & | & \mathbf{y}^T \end{pmatrix} \left(\begin{array}{c|c} c & \mathbf{a}^T \\ \hline \mathbf{a} & \mathbf{B} \end{array}\right) \left(\begin{array}{c} z \\ \hline \mathbf{y} \end{array}\right) = cz^2 + 2z\mathbf{a}^T\mathbf{y} + \mathbf{y}^T\mathbf{B}\mathbf{y} = c(z + \frac{1}{c}\mathbf{a}^T\mathbf{y})^2 + \mathbf{y}^T\mathbf{B}\mathbf{y} - \frac{\left(\mathbf{a}^T\mathbf{y}\right)^2}{c}$$

so

$$\mathbf{y}^T\mathbf{B}\mathbf{y} - \frac{\left(\mathbf{a}^T\mathbf{y}\right)^2}{c} > 0 \text{ for any vector } \mathbf{y}$$

The matrix in the next step becomes

$$\mathbf{C} = \mathbf{B} - \mathbf{a}\mathbf{a}^T / c; \mathbf{y}^T\mathbf{C}\mathbf{y} = \mathbf{y}^T\mathbf{B}\mathbf{y} - \mathbf{y}^T\mathbf{a}\mathbf{a}^T\mathbf{y} / c = \mathbf{y}^T\mathbf{B}\mathbf{y} - (\mathbf{a}^T\mathbf{y})^2 / c$$

so **C** is also SPD.

Example: Eigenvalue problem or solution of linear system?
Google's page rank algorithm, see e.g. article by C.Moler at Mathworks home page.

Graphs
Set of vertices (nodes, ...) V and (directed) edges E.
Nodes are numbered 1:n, edges 1:m.
Node = web page, edge = hyperlink

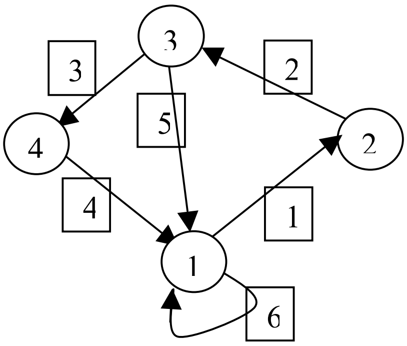
Representation:
1. *Edgelist*: L(k,1) = i, L(k,2) = j - an edge from node i to node j.
2. The *edge-node incidence* matrix **A** (S p143):
Edge k, from i to j: **A**(k,i) = -1, **A**(k,j) = +1, the rest zero. How represent edges from i to i? Store as sparse matrix.
3. The *node-node adjacency matrix* **W** (S p 142): **W**(i,j) = 1 if edge from i to j, the rest zeros.
(Out/In)*degree* of node i: number of out/in going edges

$$Od_i = \sum_j w_{ij}, Id_i = \sum_j w_{ji} \qquad \mathbf{1} = \texttt{ones(n,1)}$$

$$\mathbf{Od} = \mathbf{W1}, \mathbf{Id} = \mathbf{W}^T \mathbf{1},$$

Ex. Four nodes, five (six) edges

$$A = \begin{pmatrix} -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \\ 1 & 0 & 0 & -1 \\ 1 & 0 & -1 & 0 \\ ? & ? & \ldots & ? \end{pmatrix}$$



$$W = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}, Od = \begin{pmatrix} 2 \\ 1 \\ 2 \\ 1 \end{pmatrix}, Id = \begin{pmatrix} 3 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

Markov chains

"Random walker" X^n a stochastic variable which takes on values 1, 2, ..., n (the nodes). Each timestep X "jumps" along the edges at random, with given frequencies / probabilities :

$$P(X^n = i) = \sum_k P(X^{n-1} = k) \cdot \underbrace{P(X^n = i | X^{n-1} = k)}_{w_{ki}}$$

$$p_i^n = \sum_k w_{ki} p_k^{n-1} \Rightarrow \mathbf{p}^n = \mathbf{W}^T \mathbf{p}^{n-1}$$

Notes:

- $0 \le w_{ki} \le 1$
- $\mathbf{W1} = \mathbf{1}$ (the process goes to one of the nodes with probability 1)

It follows that the **p**-vector tends to a limit $\mathbf{p}\infty$ as n increases.
$\mathbf{p}\infty$ is

- the set of expected number of visitors to a node, or
- the average fraction of time spent at that node by a single process.

We must have

$$\mathbf{p}\infty = \mathbf{W}^\mathbf{T}\mathbf{p}\infty,$$

so $\mathbf{p}\infty$ is the (right) eigenvector of the eigenvalue 1 of $\mathbf{W}^\mathbf{T}$. (?)
We know that

- **W** has an eigenvalue 1 with eigenvector **1**.
- **...** so $\mathbf{W}^\mathrm{T}$ also has an eigenvalue 1, but what is its eigenvector?

The Web model: A random surfer

- chooses a random page with small probability *q/n*,
- follows a link on the current page k with equal probability, $p = (1\text{-}q)/\mathrm{Od}_k$

This defines the *very big* (n = 3G in 2003) ***full*** Markov matrix **W**. Check that it is a Markov matrix - that rows sum to 1.
The matrix of existing links is very sparse, so one might represent **W** as the sum of this sparse matrix and a rank-one correction $q/n\ \mathbf{1}\ \mathbf{1}^\mathrm{T}$ = `q/n*ones(n)` which is not stored.
Task: Compute $\mathbf{p}\infty$ and rank the pages according to decreasing component in $\mathbf{p}\infty$!

How?
0) Standard eigensolution by diagonalization
1) Power method for eigenvalue problem, or time-stepping $\mathbf{p}^n = \mathbf{W}^\mathrm{T}\ \mathbf{p}^{n-1}$
2) Solve $\mathbf{x} = \mathbf{W}^\mathrm{T}\mathbf{x}$ by faster iteration. ... singularity? can use $\mathbf{1}^\mathrm{T}\mathbf{x} = 1$ as "extra equation"
Diagonal elements are guaranteed to be $\ge q$, so no divide by zero problem. Gauss-Seidel faster than Jacobi (= 1). Even faster ?

How compute $\mathbf{W}^\mathrm{T}\mathbf{x}$??? Google secret?