Numerical linear algebra

Purdue University
CS 51500

~~Fall 2019~~
Fall 2020
David Gleich

David F. Gleich

Call me …
“Prof Gleich”
“Dr. Gleich”

Please not
“Hey matrix guy!”

Numerical linear algebra

Or

Matrix computations

Purpose

Matrix computations underlie much (most?) of applied computations.

It's the language of computational algorithms.

PageRank (from the paper)

$$
\begin{aligned}
R_0 &\leftarrow S \\
\text{loop :} & \\
R_{i+1} &\leftarrow AR_i \\
d &\leftarrow ||R_i||_1 - ||R_{i+1}||_1 \\
R_{i+1} &\leftarrow R_{i+1} + dE \\
\delta &\leftarrow ||R_{i+1} - R_i||_1 \\
\text{while } \delta > \epsilon &
\end{aligned}
$$

BFGS

3. The Generalized Method

In this method (Broyden, 1967) the vector $\mathbf{p}_i$ is given by

$$\mathbf{p}_i = -\mathbf{H}_i\mathbf{f}_i, \tag{3.1}$$

where $\mathbf{H}_i$ is positive definite. $\mathbf{H}_1$ is chosen to be an arbitrary positive definite matrix (often the unit matrix) and $\mathbf{H}_{i+1}$ is given by

$$\mathbf{H}_{i+1} = \mathbf{H}_i - \mathbf{H}_i\mathbf{y}_i\mathbf{w}_i^T + \mathbf{p}_i t_i \mathbf{q}_i^T, \qquad i = 1, 2, \ldots, \tag{3.2a}$$

where

$$\mathbf{y}_i = \mathbf{f}_{i+1} - \mathbf{f}_i, \tag{3.2b}$$

$$\mathbf{q}_i^T = \alpha_i \mathbf{p}_i^T - \beta_i \mathbf{y}_i^T \mathbf{H}_i, \tag{3.2c}$$

$$\mathbf{w}_i^T = \gamma_i \mathbf{y}_i^T \mathbf{H}_i + \beta_i t_i \mathbf{p}_i^T, \tag{3.2d}$$

$$\alpha_i = (1 + \beta_i \mathbf{y}_i^T \mathbf{H}_i \mathbf{y}_i)/\mathbf{p}_i^T \mathbf{y}_i, \tag{3.2e}$$

$$\gamma_i = (1 - \beta_i t_i \mathbf{p}_i^T \mathbf{y}_i)/\mathbf{y}_i^T \mathbf{H}_i \mathbf{y}_i. \tag{3.2f}$$

The parameter β_i is arbitrary and setting it equal to zero gives the DFP method (Fletcher & Powell, 1963). It was shown by Broyden (1967) that the matrices $\mathbf{H}_i$ constructed in this way are always positive definite if $\beta_i \geqslant 0$.

Circular antennae design

B. Port Description of Array

Since relatively few of the elements of V are nonzero some reduction in (3) is possible. Only those columns of Y which correspond to indices of triangles centered at the dipole mid-points need be retained in (3); denote as Y_R the rectangular matrix obtained by deleting all columns of Y not having such a column index. Then,

$$I = Y_R V_T \tag{8}$$

where V_T is the N vector formed by deleting all identically zero elements of V. Y_R is denoted the "reduced admittance matrix."

Furthermore, if only the feed-point currents are of interest, a similar reduction may be performed on rows of Y_R and I to yield,

$$I_T = Y_T V_T \tag{9}$$

Dynamic mode decomposition

1. Split the time series of data in V_1^N into the two matrices V_1^{N-1} and V_2^N.
2. Compute the SVD of $V_1^{N-1} = U\Sigma W^T$.
3. Form the matrix $\tilde{S} = U^T V_2^N W \Sigma^{-1}$, and compute its eigenvalues λ_i and eigenvectors y_i.
4. The i-th DMD eigenvalues is the λ_i and the i-th DMD mode is the $U y_i$.

Electrical circuits

“A matrix version of Kirchhoff’s circuit law is the basis of most circuit simulation software”

-- Wikipedia

Other applications

Biology

PDEs/Mechanical Engineering/AeroAstro

Machine learning

Statistics

Graphics

Purpose

The purpose this class is to teach you how to "speak matrix computations like a native" so that you can understand, implement, interpret, and extend work that uses them.

Examples

Why should we avoid the “normal equations”?

Why do I get strange looks if I talk about the SVD of a symmetric positive definite matrix?

Why not write things element-wise?

Please pay attention for a second, this next bit is important!

The new class schedule

Basic Problems

- Least Squares, Linear Systems, Singular Values, Eigenvalues, Sparse Matrices,

Simple Algorithms

- Gradient descent, power method
- Convergence analysis

Finite Termination

- Coordinate fixing -> Cholesky
- LU with pivoting
- QR factorization

Conditioning & Stability (after midterm)

- How to choose algorithms?

Advanced Problems

- Sequences of linear systems
- Generalized eigenvalue problems

Krylov Methods

- Arnoldi, Lanczos

Eigenvalue algorithms

- All eigenvalues
- Some eigenvalues

Getting high performance, randomized?

Why did I change this?

- One weakness of a classic presentation is that it discourages interplay between pre/post midterm.
- The new presentation makes the class more exciting and highlights the interplay between materials.
- One downside, it doesn’t really follow an existing book.

Textbooks

No **best** reference.

Golub and van Loan – “The Bible” – but sometimes a bit terse

Trefethen & Bau, Numerical Linear Algebra

Demmel, Applied Numerical Linear Algebra

Saad, Iterative Methods for Sparse Linear Systems

Background books

Strang, Linear Algebra and its Applications

Meyer, Matrix Analysis

Why I like Julia & Matlab

Julia Designed as a technical computing language

Matlab it's a modeling language for matrix methods!

The power method described in Wikipedia

$$b_{k+1} = \frac{Ab_k}{\|Ab_k\|}.$$

```
while 1
  a = b;
  b = A*b;
  b = b/norm(b);
  if test_converge(a,b); break; end
end
```

Matlab & Julia code

```
x = b # make a reference to A
y = zeros(length(b)) # allocate
while 1
  A_mul_B!(y,A,x)     # y = Ax
  scale!(y,1/norm(x)) # scale
  if test_converge(x,y); break; en
  x,y == y,x # swap pointers
end
```

Super efficient Julia code

Software

You will have to write matrix programs in class.

Julia & Atom my recommendation (what I use!)
Julia & Jupyter notebook my 2nd recommendation
Julia & Text Editor (your call!)

Matlab what I used to use
SciPy, NumPy okay (look at spyder/pythonxy)

R not recommended, best to avoid
Scilab you're on your own
C/C++ with LAPACK okay, but ill-advised
Fortran (same!)

THE SYLLABUS

Cut to website!

www.cs.purdue.edu/homes/dgleich/cs515-2020

Quiz

- Write down any questions, concerns, issues, etc. you think you have after hearing about the class logistics.