Review for Midterm 2

Contents

Preliminaries

Two Types of Errors

- absolute error
- relative error

Floating-Point Numbers

- binary scientific notation:

$$\pm\left(1+\frac{b_1}{2}+\frac{b_2}{2^2}+\cdots+\frac{b_d}{2^d}\right)2^E,$$

  where b_i is 0 or 1 and E is an integer.

  - d determines the *resolution*
  - the range of E determines the *scope* or *extent*

- IEEE Standard (double-precision; 64 bits)
  - $d = 52$ and $-1022 \leqslant E \leqslant 1023$
  - $\boxed{\text{eps}} = 2^{-52} \approx 2 \times 10^{-16}$
  - `realmin`, `realmax`

Floating-Point Numbers (cont')

- Key features
  - On any interval of the form $[2^E, 2^{E+1})$, there are 2^d evenly-spaced f-p numbers.
  - The spacing between two adjacent f-p numbers in $[2^E, 2^{E+1})$ is $2^{E-d} = 2^E \boxed{\text{eps}}$.
  - The gap between 1 and the next f-p number is $\boxed{\text{eps}}$, the machine epsilon.
  - Representation error (in relative sense) is bounded by $\frac{1}{2}\boxed{\text{eps}}$.

Conditioning (of a problem)

- The condition number measures the ratio of error in the result (or output) to error in the data (or input).
- Recall the definition of condition number $\kappa_f(x)$
- A large condition number implies that the error in a result may be much greater than the round-off error used to compute it.
- Catastrophic cancellation is one of the most common sources of loss of precision.

Stability (of an algorithm)

- When an algorithm produces much more error than can be explained by the condition number, the algorithm is unstable.

Square Linear Systems

Polynomial Interpolation

- Polynomial interpolation leads to a square linear system of equations with a Vandermonde matrix.

Gaussian Elimination and (P)LU Factorization

- A triangular linear system is solved by backward substitution or forward elimination.
- A general linear system is solved by Gaussian elimination.
- Gaussian elimination (with partial pivoting) is equivalent to (P)LU factorization.
- Solving a triangular linear system of size $n \times n$ takes $\sim n^2$ flops.
- PLU factorization takes $\sim \frac{2}{3}n^2$ flops.

Norms

A *norm* generalizes the notion of length for vectors and matrices.

- **Vector *p*-norm**

$$\|\mathbf{v}\|_p = \left(\sum_{i=1}^{n} |b_i|^p\right)^{1/p}, \quad p \in [1, \infty)$$

  and

$$\|\mathbf{v}\|_\infty = \max_i |v_i|$$

- **Matrix *p*-norm** (induced)

$$\|A\|_p = \max_{\|\mathbf{x}\|_p = 1} \|A\mathbf{x}\|_p, \quad p \in [1, \infty]$$

- **Frobenius norm** (non-induced)

$$\|A\|_F = \left(\sum_i \sum_j |a_{i,j}|^2\right)^{1/2}$$

- MATLAB: `norm` can calculate both vector and matrix norms

Row and Column Operations

Various row and column operations can be emulated by matrix multiplications.
("Left-multiplication for row actions, right-multiplication for column actions")

- row/column extraction (unit vector)
- row/column swap (elementary permutation matrix)
- row/column rearrangement (permutation matrix)
- row replacement $R_i \to R_i + cR_j$ (Gaussian transformation matrix)

Conditioning/Stability

- Partial pivoting is needed for numerical stability.
- The matrix condition number is equal to the condition number of solving a linear system of equations.

Programming Notes

- Built-in functionalities
  - backslash (\)
  - `lu`
  - `norm`
  - `cond`, `condest`, `linsolve`
- Demonstration/Instructional codes
  - `backsub` and `forelim`
  - `GEnp` and `GEpp`
  - `mylu` and `myplu`

Overdetermined Linear Systems

Polynomial Approximation

- The most common solution to overdetermined systems is obtained by *least squares*, which minimizes the 2-norm of the residual vector.
- Least squares is used to find fitting functions that depend linearly on the unknown parameters.
- Equivalence of the LLS problem and the normal equation
  - linear algebra proof
  - calculus proof

QR Factorization

- Orthogonal sets of vectors are preferred to nonorthogonal ones in computing. (no catastrophic cancellation)
- Matrices with orthonormal columns and orthogonal matrices enjoy many *nice* analytical properties.
- QR factorization plays a role in LLS similar to that of LU factorization in square linear systems.

Two Types of QR Factorization

For $A \in \mathbb{R}^{m \times n}$, $m \geqslant n$:

- Thick QR factorization: $A = QR$
  - $Q \in \mathbb{R}^{m \times m}$ orthogonal
  - $R \in \mathbb{R}^{m \times n}$ upper triangular
  - obtained by using successive Householder transformation matrices for *triangularization*
- Thin: $A = \widehat{Q}\widehat{R}$
  - $\widehat{Q} \in \mathbb{R}^{m \times n}$ orthonormal columns
  - $\widehat{R} \in \mathbb{R}^{n \times n}$ upper triangular
  - obtained by Gram-Schmidt *orthonormalization* procedure

Householder Transformation Matrices

- A Householder transformation matrix H (associated with a vector $\mathbf{z}$) is a *reflection* matrix which is
  - symmetric,
  - orthogonal, and
  - transforms $\mathbf{z}$ to $\pm \|\mathbf{z}\|_2 \, \mathbf{e}_1$.

Programming Notes

- Built-in functionalities
  - backslash (\)
  - `qr`
- Demonstration/Instructional codes
  - `lsqrfact`: solving least squares using QR
  - `gs`: Gram-Schmidt (for homework)