24. Conditioning of linear systems and iterative methods

Last time

- Singular-value decomposition
- Eigen-decomposition
- Power iteration for eigenvectors

Goals for today

- Condition number for solving a linear system
- Iterative methods for solving linear systems

Sensitivity of solving linear systems

- How sensitive to perturbations is solving $A\mathbf{x} = \mathbf{b}$?
- To answer this question we need to be able to talk about the **norm** (“size”) of vectors and of matrices

Sensitivity of solving linear systems

- How sensitive to perturbations is solving $A\mathbf{x} = \mathbf{b}$?
- To answer this question we need to be able to talk about the **norm** (“size”) of vectors and of matrices

- For vector $\mathbf{v} \in \mathbb{R}^m$ we will use 2-norm:

$$\|\mathbf{v}\|_2 := \sqrt{\sum_{i=1}^{m} v_i^2}$$

Induced matrix norm

- There are various ways of measuring the size (norm) of a matrix that are useful for different purposes
- We will focus on the so-called **induced** matrix norm, induced from a given norm on vectors

Induced matrix norm

- There are various ways of measuring the size (norm) of a matrix that are useful for different purposes
- We will focus on the so-called **induced** matrix norm, induced from a given norm on vectors

- Suppose A is a square $(m \times m)$ matrix
- For each **x** can measure length **x** and of its image A**x** under A
- Natural to ask *how much* A can stretch **x**

Induced matrix norm II

- Define induced matrix norm as smallest C such that

$$\|\mathsf{A}\| \leq C\|\mathbf{x}\| \quad \forall \mathbf{x} \in \mathbb{R}^m$$

Induced matrix norm II

- Define induced matrix norm as smallest C such that

$$\|\mathsf{A}\| \leq C\|\mathbf{x}\| \quad \forall \mathbf{x} \in \mathbb{R}^m$$

- Hence

$$\|\mathsf{A}\| = \sup_{\|\mathbf{x}\|=1} \|\mathsf{A}\mathbf{x}\|$$

- i.e. biggest stretching on unit circle
- So for 2-norm, $\|\mathsf{A}\| = \sigma_1$, largest singular value

. . .

- Note that $\|\mathsf{A}\mathbf{x}\| \leq \|\mathsf{A}\|\|\mathbf{x}\|$

Condition number

- How sensitive is solution of the problem

  solve $A\mathbf{x} = \mathbf{b}$

  to perturbations *of the problem*?
- I.e. perturbations in A or $\mathbf{b}$?

Condition number

- How sensitive is solution of the problem

  solve $A\mathbf{x} = \mathbf{b}$

  to perturbations *of the problem*?
- I.e. perturbations in A or $\mathbf{b}$?

- Simplest case: Perturb *input* $\mathbf{b}$ to $\mathbf{b} + \Delta\mathbf{b}$
- How much is *output* $\mathbf{x}$ perturbed?

Condition number II

- We have

$$\mathsf{A}(\mathbf{x} + \Delta\mathbf{x}) = \mathbf{b} + \Delta\mathbf{b}$$

- Since $\mathsf{A}\mathbf{x} = \mathbf{b}$, we find

$$\mathsf{A}(\Delta\mathbf{x}) \simeq \Delta\mathbf{b}$$

- Hence to first order in perturbations:

$$\Delta\mathbf{x} = \mathsf{A}^{-1}(\Delta\mathbf{b})$$

Condition number III

- So relative condition number is

$$\kappa = \frac{\|\Delta \mathbf{x}\|/\|\mathbf{x}\|}{\|\Delta \mathbf{b}\|/\|\mathbf{b}\|}$$

$\cdot \quad \cdot \quad \cdot$

- Using $\mathbf{b} = \mathsf{A}\mathbf{x}$ we get

$$= \frac{\|\mathsf{A}^{-1}\Delta \mathbf{b}\|\|\mathsf{A}\mathbf{x}\|}{\|\mathbf{x}\|\|\Delta \mathbf{b}\|}$$

Condition number IV

Have $\kappa = \frac{\|A^{-1}\Delta\mathbf{b}\|\|A\mathbf{x}\|}{\|\mathbf{x}\|\|\Delta\mathbf{b}\|}$

- Now use $\|A\mathbf{x}\| \leq \|A\|\|\mathbf{x}\|$ to get

$$\kappa \leq \|A\|\|A^{-1}\|$$

Condition number IV

Have $\kappa = \frac{\|\mathsf{A}^{-1}\Delta\mathbf{b}\|\|\mathsf{A}\mathbf{x}\|}{\|\mathbf{x}\|\|\Delta\mathbf{b}\|}$

- Now use $\|\mathsf{A}\mathbf{x}\| \leq \|\mathsf{A}\|\|\mathbf{x}\|$ to get

$$\kappa \leq \|\mathsf{A}\|\|\mathsf{A}^{-1}\|$$

- Define **condition number of matrix** A as

$$\kappa(\mathsf{A}) := \|\mathsf{A}\|\|\mathsf{A}^{-1}\|$$

Residual

- Suppose we are solving $A\mathbf{x} = \mathbf{b}$
- Suppose have approximate solution $\widehat{\mathbf{x}}$

Residual

- Suppose we are solving $A\mathbf{x} = \mathbf{b}$
- Suppose have approximate solution $\hat{\mathbf{x}}$
- As usual, do not know how far $\mathbf{x}$ is from true solution
- i.e. **forward error** $\hat{\mathbf{x}} - \mathbf{x}$
- Instead, only know the **residual**

$$\mathbf{r} := A\hat{\mathbf{x}} - \mathbf{b}$$

Backward error

- We see that $\hat{\mathbf{x}}$ **exactly** solves a **perturbed problem**:

$$A\hat{\mathbf{x}} = \mathbf{b} - \mathbf{r}$$

- Relative error is

$$\frac{\|\Delta \mathbf{x}\|}{\|\mathbf{x}\|} = \frac{\|\hat{\mathbf{x}} - \mathbf{x}\|}{\|\mathbf{x}\|} \leq \kappa(A)\frac{\|\mathbf{r}\|}{\|\mathbf{b}\|}$$

Backward error

- We see that $\hat{\mathbf{x}}$ **exactly** solves a **perturbed problem**:

$$\mathsf{A}\hat{\mathbf{x}} = \mathbf{b} - \mathbf{r}$$

- Relative error is

$$\frac{\|\Delta \mathbf{x}\|}{\|\mathbf{x}\|} = \frac{\|\hat{\mathbf{x}} - \mathbf{x}\|}{\|\mathbf{x}\|} \leq \kappa(A) \frac{\|\mathbf{r}\|}{\|\mathbf{b}\|}$$

- So forward error can be large even when backward error is small, if $\kappa(A)$ is large

Iterative methods for solving linear systems

Need for iterative methods

- Although we have at least two exact, finite methods for solving linear systems, it may be useful to turn to an **iterative** method

Need for iterative methods

- Although we have at least two exact, finite methods for solving linear systems, it may be useful to turn to an **iterative** method
- Can get approximate solution faster
- May not even be able to store matrix since too large, e.g. discretization of PDE
- Some iterative methods just require to be able to compute $A\mathbf{x}$ . . .

- How can we turn $A\mathbf{x} = \mathbf{b}$ into an iterative method?
- Recall from start of course:

Rearrange the equation to have more than one occur-

Splitting

- To do this we try to **split** A into two pieces:

$$\mathrm{A}\mathbf{x} = (\mathrm{A}_1 + \mathrm{A}_2)\mathbf{x} = \mathbf{b}$$

- Hence

$$\mathrm{A}_1\mathbf{x} = \mathbf{b} - \mathrm{A}_2\mathbf{x}$$

Splitting

- To do this we try to **split** A into two pieces:

$$\mathsf{A}\mathbf{x} = (\mathsf{A}_1 + \mathsf{A}_2)\mathbf{x} = \mathbf{b}$$

- Hence

$$\mathsf{A}_1\mathbf{x} = \mathbf{b} - \mathsf{A}_2\mathbf{x}$$

- Convert into an iterative method:

$$\mathsf{A}_1\mathbf{x}_{n+1} = \mathbf{b} - \mathsf{A}_2\mathbf{x}_n$$

Splitting

- To do this we try to **split** A into two pieces:

$$\mathsf{A}\mathbf{x} = (\mathsf{A}_1 + \mathsf{A}_2)\mathbf{x} = \mathbf{b}$$

- Hence

$$\mathsf{A}_1\mathbf{x} = \mathbf{b} - \mathsf{A}_2\mathbf{x}$$

- Convert into an iterative method:

$$\mathsf{A}_1\mathbf{x}_{n+1} = \mathbf{b} - \mathsf{A}_2\mathbf{x}_n$$

- Seems like harder problem than we started with!

Splitting

- To do this we try to **split** A into two pieces:

$$\mathsf{A}\mathbf{x} = (\mathsf{A}_1 + \mathsf{A}_2)\mathbf{x} = \mathbf{b}$$

- Hence

$$\mathsf{A}_1\mathbf{x} = \mathbf{b} - \mathsf{A}_2\mathbf{x}$$

- Convert into an iterative method:

$$\mathsf{A}_1\mathbf{x}_{n+1} = \mathbf{b} - \mathsf{A}_2\mathbf{x}_n$$

- Seems like harder problem than we started with!
- But easier if choose A_1 to be *easy to invert / solve*

Convergence

- We have

$$\mathbf{x}_{n+1} = \mathsf{A}_1^{-1}(-\mathsf{A}_2\mathbf{x}_n + b)$$

- Convergence depends on properties of $\mathsf{R} := \mathsf{A}_1^{-1}\mathsf{A}_2$
- Namely how much it stretches vectors

Convergence

- We have

$$\mathbf{x}_{n+1} = \mathsf{A}_1^{-1}(-\mathsf{A}_2\mathbf{x}_n + b)$$

- Convergence depends on properties of $\mathsf{R} := \mathsf{A}_1^{-1}\mathsf{A}_2$
- Namely how much it stretches vectors
- Can show that convergence if $\rho(\mathsf{R}) < 1$
- Where $\rho(\mathsf{R})$ is **spectral radius** of R

$$\rho(\mathsf{R}) = \max_i |\lambda_i(\mathsf{R})|$$

- $\lambda_i(\mathsf{R})$ are eigenvalues of R

Jacobi method

- E.g. choose $A_1 := D$ and $A_2 := A - D$ where $D := \mathrm{diag}(A)$ is diagonal part of A

Jacobi method

- E.g. choose $A_1 := D$ and $A_2 := A - D$
  where $D := \mathrm{diag}(A)$ is diagonal part of A
- **Jacobi method**

Jacobi method

- E.g. choose $\mathsf{A}_1 := \mathsf{D}$ and $\mathsf{A}_2 := \mathsf{A} - \mathsf{D}$

  where $\mathsf{D} := \mathrm{diag}(\mathsf{A})$ is diagonal part of A
- **Jacobi method**
- Each iteration is $\mathcal{O}(n^2)$ steps

Jacobi method

- E.g. choose $\mathsf{A}_1 := \mathsf{D}$ and $\mathsf{A}_2 := \mathsf{A} - \mathsf{D}$
  where $\mathsf{D} := \mathrm{diag}(\mathsf{A})$ is diagonal part of A
- **Jacobi method**
- Each iteration is $\mathcal{O}(n^2)$ steps
- Only works for certain matrices A
- In particular when **diagonally dominant**:

$$|A_{i,i}| > \sum_{j \neq i} |A_{i,j}|$$

- I.e. diagonal term is $>$ sum of off-diagonals in same row

Gauss–Seidel

- Instead decompose $A = L + U$
- Same as Jacobi except uses already updated x_is; twice as fast

Krylov methods

- Methods that use only $A\mathbf{x}$

Summary

- Defined induced norm for a matrix
- Condition number for solving linear system is

$$\kappa = \|\mathsf{A}\| \|\mathsf{A}^{-1}\|$$

- Can find iterative methods for solving linear systems