APMA E4300: Introduction to Numerical Methods

Michael Cai

March 23, 2025

Abstract

This course introduces and analyzes fundamental numerical methods in scientific computation. The main topics include elementary numerical linear algebra, least-square methods, nonlinear equations and optimization, polynomial interpolation, numerical differentiation and integration, initial-value and boundary problems for ordinary differential equations and systems, eigenvalue problems. This course prepares students for more advanced studies in the field of numerical analysis and scientific computation (for instance, APMA courses E4301, E4302, and E6302).

Contents

Chapter 1

Linear Algebra Overview

Lecture 1

Jan 23 10:10

Remark.

$$Av = \lambda v$$

When $\{v_i\}_{i=1}^n$ is linearly independent, then we wil have that

$$Av_i = \lambda_i v_i$$

We can then write A as a product of two matrices such that we have

$$AQ = Q\alpha A = Q\alpha Q^{-1}$$

Remark. Suppose that some of the values do not contribute much in

$$A = Q\alpha Q^{-1}$$

Then we can throw away some nonzero elements given that they are small. Thus, we can create a approximation of the matrix which was a (256 x 256) matrix. We have converted it to a $n \times 2, 2 \times 2, 2 \times n$.

Example. We can have an image that is 256 x 256, we can compress it without losing too much information. However, there is never any free lunch. Any two dimensional object, 3d, etc. can be done. Thus, we can generalize it further in the course...

With λ being very small, it is negligible and can be omitted.

We would like to know how much error can be omitted.

Definition 1.0.1. l^2 norm is going to be denoted as $||\mathbf{x}||$ which is a negative number given by

$$\sqrt{\sum_{i=1}^{n} x_i^2}$$

Theorem 1.0.1. The l_2 norm satisfies the following properties:

- $||x||_{l_2} = 0 \Leftrightarrow x = 0$
- $||\alpha x|| = |\alpha| ||x||_{l_2}$

Lecture 2

Jan 28 10:10

Remark. Gaussian Elimination:
Suppose we had an $n \times n$ system with n unknowns and n equations denoted as

$$Ax = y, A \in \mathbb{R}^{\ltimes \times \ltimes}$$

Example. Given an example of the equation

$$y = a_0 + a_1 x + \cdots + a_{m-1} x^{m-1}$$

How do we figure out the coefficients. Now suppose that we are given a set of points, it is quite easy to solve a matrix for the given system of equations

$$a_0 + a_1 x + \cdots + a_{n-1} x^{n-1} = y_1$$

$$\ldots$$

$$a_0 + a_1 x_n + \cdots = y_m$$

This results in a matrix where we can solve for the values of the coefficients now.

$$\begin{pmatrix} 1 & x_1 & x_1^{m-1} \\ \ldots & & \\ 1 & x_m & x_m^{m-1} \end{pmatrix} \begin{pmatrix} a_0 \\ \ldots \\ a_m \end{pmatrix} = \begin{pmatrix} y_0 \\ \ldots \\ y_m \end{pmatrix}$$

In practice, we could measure data points which leads to a situation where there are more equations or x values than coefficients which is defined as a "overdetermined matrix". On the other hand, we can also have an undertermined problem.

Example. Suppose we have another example with a system of equations with $y = a_0 + a_1 \sin x + a_2 \sin 2x \ldots a_n \sin x$. If we continue this this is also a linear system of equations where the matrix of the equations will be

$$\begin{pmatrix} 1 & \sin x_1 & \sin(m-1)x_1 \\ \ldots & & \\ 1 & \sin x_m & \sin(m-1)x_m \end{pmatrix} \begin{pmatrix} a_0 \\ \ldots \\ a_m \end{pmatrix} = \begin{pmatrix} y_0 \\ \ldots \\ y_m \end{pmatrix}$$

One useful example is a gaussian mixture model where you can suerimpose a number of gaussian models to get the value of y. Thus, solving $Ax = y$ gives many possible applications.

Elementary Row Operations:

Definition 1.0.2. Elementary row operations can be done with multiplication. Scaling is done with the following matrix

$$S_i(p) = \begin{pmatrix} 1 & & \\ & p & \\ & & 1 \end{pmatrix}$$

which gives us a scaling factor on a row of the matrix. We simply see that the inverse of the matrix would be simply be $\frac{1}{p}$ for $S_i(p)^{-1}$

Definition 1.0.3. Interchange: interchanging row i and row j of $\vec{A}$ gives us

$$E_{ij}A = \begin{pmatrix} 1 & & & & \\ & 0 & 1 & & \\ & 1 & 0 & & \\ & & & 1 & \\ & & & & 1 \end{pmatrix} A$$

This showcases a exchange of rows i and j. Note that the inverse of such a matrix is just itself.

Definition 1.0.4. Replacement is also a method too. (finish later)

Remark. Gaussian Elimination:

We can start with an easy example

$$A = \begin{pmatrix} 2 & 1 & 1 & 0 \\ 4 & 3 & 3 & 1 \\ 8 & 7 & 9 & 5 \\ \dots & \dots & \dots & \dots \end{pmatrix}$$

We can then apply a matrix L_1 which is our linear operator we use to transform our matrix using gaussian elimination.

We can apply these operations which results in

$$L_3 L_2 L_1 A x = L_3 L_2 L_1 y$$

which will give us a upper triangular matrix. We can then solve and use backwards substitution toget our answer from the upper triangular matrix.

Let us denote this as

$$U = L_3 L_2 L_1 A$$

Thus, we can write everything as

$$Ux = L^{-1} y \Rightarrow LUx = y$$

where U is always an upper triangular matrix and the L is always a lower triangular matrix. This only works with elimination methods and not row swaps or replacements. This can be easily solvable by having

$$LUx = y \Rightarrow Lz = y$$

solve for z, do a backward substitution for U and we can easily get the coefficients for x

Lecture 3

Lecture 4: 10:10

Jan 30 10:10

Could not attend. Check lecture slides instead.

4 Feb 2025

Lecture 5: 10:10

6 Feb 2025

1.1 Linear Iterative Schemes for Linear Systems

Definition 1.1.1. Define the error at step k to be

$$e^k = x^k - x \Rightarrow e^{k+1} = Be^k$$

$$\Rightarrow e^k = B^k e^0$$

We see if B is greater than 1 , we are amplifying the noise of the answer. Thus, we want to use some method to get the size of the matrix $I - A$ to be less than 1

1.1.1 Jacobi Iteration

For the Jacobi Iteration we acn write that

$$A = P - N$$

$$Ax = y \Rightarrow Px = Nx + y$$

Remark. We have rewritten the equation in this form but is not in the form $x = Bx + y$. We can write instead

$$X = P^{-1}Nx + P^{-1}y$$

We can now perform the iteration here which makes the calculations much easier.

Recall that the Jacobi is going to consist of

$$P = D_A, N = L_A + U_A$$

Remark. Selecting $P = D_A$ is useful because the inverse of P is simply

$$\begin{pmatrix} \frac{1}{p_1} & & \\ & \frac{1}{p_2} & \\ & & \frac{1}{p_n} \end{pmatrix}$$

Thus given x_0,

$$x^{k+1} = D_A^{-1}(L_A + U_A)x^k + D_A^{-1}y, k \geq 0$$

Note that if we have the diagonal to be 0 for one part, the method is not possible. However, it is nice to see when the diagonal is very large.

In component form, we can write this as

$$x_1^{k+1} = \frac{y_1 - \sum_{j \neq 1} a_j x_j^K}{a_{11}}$$

$$\vdots$$

$$x_i^{k+1} = \frac{y_i - \sum_{i \neq j} a_{ij} x_j^k}{a_{ii}}$$

$$\vdots$$

$$x_n^{k+1} = \frac{y_n - \sum_{j \neq n} a_{nj} x_j^k}{a_{nn}}$$

Stopping Criteria We have to designate a stopping criteria for linear iterations. ONe of the methods is using a maximum amount of iterations and the other method is using the relative size of the residual. This is defined as the size of $y - Ax^k$ over the initial as

$$\frac{\|y - Ax^k\|_{l2}}{\|y - Ax^0\|_{l2}}$$

Lecture 6: 10:10

Could not attend. Check lecture slides instead. 11 Feb 2025

Lecture 7: 10:10

Could not attend. Check lecture slides instead. 13 Feb 2025

Lecture 8: 10:10

Could not attend. Check lecture slides instead. 18 Feb 2025

Lecture 9: 10:10

Could not attend. Check lecture slides instead. 20 Feb 2025

Lecture 10: 10:10

Could not attend. Check lecture slides instead. 25 Feb 2025

Lecture 11: 10:10

Could not attend. Check lecture slides instead. 4 Mar 2025

Lecture 12

6 Mar 10:10

Chapter 2

Interpolation

Consider the concept of interpolation where we are using extrapolation and generalization. Suppose we had a neural network where we have fitted it with data point $D(t)$. We are asking how general the neural network can be such that we can fit our data.

Definition 2.0.1. Interpolant: Given a discrete set of values y_i at locations x_i an interpolant is a piecewise continuous function $f(x)$ that passes through the data such that

$$f(x_i) = y_i \quad 1 \leq i \leq n$$

After we fit the function we want to evaluate the new data points. Notice that there is a factor that we can't control for such that we don't know the form of $D(t)$ where $D(t) = P_n t^n + P_{n-1} t^{n-1} \cdots + P_1$

Example. Suppose we had data points that looked like a linear line. Then we can guess the form

$$D(t) = a + bt$$

Given our plots we have a linear set of equations

$$a + bt_1 = D_1$$

$$a + bt_2 = D_2$$

$$\cdots$$

Example. If we want a quadratic interpolation we need three data points
Let $P_2(x) = p_0 + p_1 x + p_2 x^2$, then we have the requirements

$$\sum_{i=1}^{3} p_0 + p_1 x_i + p_2 x_i^2 = y_i$$

which can be put into matrix form and solved which will give a form in terms of fractions.

In general, given $n+1$ data points we can generate an n-th order polynomial of the form

$$\begin{pmatrix} 1 & x_0 & \ldots & x_0^n \\ 1 & x_1 & \ldots & x_1^n \\ & & & \\ 1 & & \ldots & x_n^n \end{pmatrix} \begin{pmatrix} p_0 \\ \vdots \\ p_n \end{pmatrix} = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}$$

$$Ap = y$$

Theorem 2.0.1. Weierstrass Approximation Theorem: Let $f \in \mathcal{C}([a,b])$ be given. For $\forall \epsilon > 0, \exists$ a polynomial $P(x)$ such that

$$|P(x) - f(x)| \leq \epsilon, \quad \forall x \in [a,b]$$

2.1 Lagrange Interpolation

Theorem 2.1.1. Suppose we had $n+1$ distinct points. THen there exists a unique polynomial of degreen n such that $P(x)$ pass through these points. We must have

$$P(x) = \sum_{k=0}^{n} f(x_k) L_{n,k}(x), \quad L_{n,k}(x) = \prod_{i=0, i\neq k}^{n} \frac{x - x_i}{x_k - x_i}$$

where L are the Lagrange interpolating polynomials at node x_k

Remark. Recall the coefficients of a nice structure in terms of fractions. This structure can go to arbitrary orders which lead to the lagrange interpolation. Thus we get the following form nice form where

$$P(x) = \sum_{k=0, x_n \neq x_k}^{n} f(x_k) \frac{x - x_0}{x_k - x_0} \frac{x - x_1}{x_k - x_1} \cdots \frac{x - x_n}{x_k - x_n}$$

This term is given as $L_{n,k}(x)$. Thus we get the form

$$P(x) = y_0 L_{n,0}(x) + y_1 L_{n,1}(x) + \cdots + y_n L_{n,n}(x)$$

Observe that

$$L_{n,k}(x_j) = \delta_{kj} = \begin{cases} 1, k = j; \\ 0, k \neq j; \end{cases}$$

Theorem 2.1.2. Error of Lagrange Interpolation: If $f \in \mathcal{C}^{n+1}(pa, b)$, then we have that

$$f(x) = \sum_{k=0}^{n} f(x_k) L_{n,x}(x) + \frac{f^{(n+1)}(\eta)}{(n+1)!} \prod_{i=0}^{n} (x - x_i)$$

where the second term is defined to be the error. We can see a taylor expansion where this is the residual terms.

Lecture 13: 10:10

11 Mar 2025

2.2 Lagrange Interpolation Proof

Proof. The Lagnrange polynomial $L_{nk}(x)$ is given as

$$L_{nk}(x) = \prod_{i=0, i\neq k}^{n} \frac{x - x_i}{x_k - x_i}$$

If $L_{nk}(x_j) = \delta_{kj}$ then if $k = j$ the whole expression cancels to be one. And 0 if otherwise. Let us now show that

$$P(x_j) = \sum_{k=0}^{n} f(x_k) L_{nk}(x_j) = f(x_j), \quad 0 \leq j \leq n$$

■

Lecture 14: 10:10

Could not attend. Check lecture slides instead. 13 Mar 2025