Recitation Materials

for NYU Undergraduate Numerical Analysis

By

Ryan Shìjié Dù
Courant Institute of Mathematical Sciences - New York University

Fall, 2022

Contents

Chapter 0

Before the Course Materials

0.1 READ ME

This document is a compilation of the worksheets used in the recitation sessions for NYU Undergraduate Numerical Analysis in Fall of 2022. Some of the problems are taken from worksheets by Georg Stadler and Evan Toler.

The LaTeX files of this document, and some demonstration codes, can be found at https://github.com/Empyreal092/NA_Worksheet.

For students in my recitations: the problems marked with *** are not in the weekly version of the worksheets. They are extra problems.

0.2 Some tips about plotting

(2.a) Size your figure so that they fit neatly onto a page without having to scale them in LaTeX.

(2.b) Exporting as vector images.

(2.c) Set desired behaviors as default.

My bag of MATLAB tools: https://github.com/Empyreal092/MATLAB_Tool.

Chapter 1

Floating-Point Arithmetic

1.1 Finite digit arithmetic

For the convenience of understanding and calculation, in this worksheet, we model the round-off error as a finite digit arithmetic [BF10]. We will use k-digit chopping for numbers. For example, if we use 3-digit chopping, and write the finite digit representation as a function $fl()$, we then have

$$fl(\pi) = 3.14$$
$$fl(1239.6) = 1230.$$

We could also use k-digit rounding where we round the last number instead of chop.

(1.a) Show that for a k-digit chopping representation of numbers, we have

$$\frac{|y - fl(y)|}{|y|} \leq \frac{1}{0.1} \times 10^{-k} = 10^{-k+1}.$$

1.2 Quadratic formula: an example of catastrophic cancellation

For the quadratic problem ($a \neq 0$)

$$ax^2 + bx + c = 0$$

the quadratic formula gives the roots:

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}, \qquad x_x = \frac{-b - \sqrt{b^2 - 4ac}}{2a}.$$

Apply this formula to $x^2 + 62.10x + 1 = 0$ whose roots are approximately

$$x_1 \approx -0.01610723, \qquad x_2 \approx -62.08390.$$

(2.a) Use four-digit rounding arithmetic in the calculation to determine the root. Compute the relative error for calculating x_1

$$\frac{|fl(x_1) - x_1|}{|x_1|}.$$

You will find the relative error is large. Why is this the case?

(2.b) A similar calculation of x_2 using the quadratic formula produces a result with a small relative error.

(2.c) To produce a better approximation of x_1, we change the formula by "rationalizing the numerator":

$$x_1 = \frac{-2c}{b + \sqrt{b^2 - 4ac}}.$$

Show that this is an equivalent formula as the quadratic formula, for exact arithmetic.

(2.d) Use the new formula to calculate a new $fl(x_1)$, and calculate the relative error. Compare this with the one from using the quadratic formula, why has there been much improvement?

1.3 Horner's rule for evaluating polynomials

We will evaluate $f(x) = x^3 - 6.1x^2 + 3.2x + 1.5$ at $x = 4.71$ using three-digit chopping arithmetic.

(3.a) The most obvious way is to evaluate each term separately and sum them up. What is the relative error? How many floating point operations are needed (count sum and multiplications separately)?

(3.b) An alternative approach is using Horner's rule to write the polynomial as a nested expression:

$$f(x) = ((x - 6.1)x + 3.2)x + 1.5.$$

Use three-digit chopping arithmetic to evaluate the function. How about the relative error now? How many floating point operations are needed?

Chapter 2

Roots Finding

2.1 Fixed point methods

We want to find the roots of the nonlinear equation

$$f(x) := x^2 - x - 2 = 0$$

using the fixed point method.

(1.a) Plot $f(x)$ in your favorite programming language.

1. Remember to add a title and axis labels to your plot.
2. Try to "generalize" your code. For example, if we want to plot another function, is it straightforward to edit your code to adapt to the new task?

(1.b) The problem has two roots. What are they?

We will try to find the positive root (x^*) through the fixed point iteration of the form $x_{k+1} = g(x_k)$. We investigate two choices:

- $g_1(x) = x^2 - 2$
- $g_2(x) = \sqrt{x + 2}$

(1.c) Verify that x^* is indeed fixed points for the two functions. That is, $x^* = g(x^*)$.

1. You could verify this with a plot. (If there are two lines in a plot, put legends on them.)

(1.d) Will both choices work in the fixed point algorithm to find the root x^*? (Hint: consider the stability of the fixed point.)

1. Could you determine the stability using the plot you made in (1.c)?

(1.e) Implement the fixed point method for both choices of $g(x)$ above.

1. Set $x_0 = 5$.
2. What would be some good termination criteria for the fixed point algorithm?
3. Try to "generalize" your code. For example, what if we want to pick a new function $g(x)$?

Now consider the convergence behavior of the stable fixed point algorithm.

(1.f) From theory alone, how fast do you expect the convergence? (linear/superlinear/quadratic)? What is the expected convergence rate?

(1.g) Is this what your numerical results show? How do you verify the convergence behavior numerically? Try to show this with a plot.

1. Plot a representation of the error $|x_k - x^*|$ against iteration numbers. Which methods of plotting should we use? `plot`, `semilogy`, or `loglog`?
2. In more complicated examples we do not know x^*. What should we plot on the y-axis?
3. Did you remember to add a title and axis label to your plot?

(1.h) Could you diagnose the convergence rate numerically? Does it match the theoretical expectation?

(1.i) Is it possible to use $g_2(x)$ in the fixed-point iteration to find the negative root? Could we modify it so that it would work? In your spare time, you could verify the convergence behavior to the negative root as an exercise.

2.2 Newton's method and roots with higher multiplicity

In the HW we have an example showing that Newton's method no longer converges quadratically if the root has higher than one multiplicity. We explore some more in this direction.

We first define multiplicity: $r \in \mathbb{R}$ is a root of multiplicity m for the equation $f(x) = 0$ if there is a function $h(x)$ such that $h(r) \neq 0$ and $f(x) = (x - r)^m h(x)$.

(2.a) Suppose that a function f has m continuous derivative on the interval (a, b) containing c (i.e. $f(x) \in C^m_{(a,b)}$). Show that f has a zero of multiplicity m at c if and only if

$$0 = f(r) = f'(r) = \cdots = f^{(m-1)}(r)$$

and

$$f^m(r) \neq 0.$$

(2.b) Suppose r is a zero of multiplicity m of f, and $f(x) \in C^m_{(a,b)}$, $r \in (a, b)$. Show that the following fixed-point method has $g'(r) = 0$:

$$g(x) = x - m\frac{f(x)}{f'(x)}.$$

What can you say about the convergence behavior of this fixed-point method?

(2.c) Code this modified Newton's method to solve $f(x) = x^2 = 0$. Check the convergence numerically. How do you plan to show this?

Chapter 3

Solving Linear Systems

3.1 Solving $Ax = b$ and LU factorization

We will study the LU-factorization of the matrix

$$A := \begin{bmatrix} 3 & 3 & 0 \\ 6 & 4 & 7 \\ -6 & -8 & 9 \end{bmatrix}$$

into the product

$$A = LU = \begin{bmatrix} 1 & 0 & 0 \\ \ell_{21} & 1 & 0 \\ \ell_{31} & \ell_{32} & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix}$$

(1.a) MATLAB matrices creation and array indexing.

1. Create matrix A in your favorite coding language.
2. Print one element of the matrix via array indexing.

(1.b) In practical Gaussian elimination, the matrices L_k, are never formed and multiplied explicitly. The multipliers ℓ_{jk} are computed and stored directly into L, and the transformations L_k are then applied implicitly [TB97, p.151].

1. Verify that Gaussian elimination could be written as the following loop:

Algorithm 20.1. Gaussian Elimination without Pivoting

$U = A,\ L = I$
for $k = 1$ **to** $m - 1$
    for $j = k + 1$ **to** m
        $\ell_{jk} = u_{jk}/u_{kk}$
        $u_{j,k:m} = u_{j,k:m} - \ell_{jk} u_{k,k:m}$

2. Apply this loop at the matrix A and obtain the L and U matrices.

(1.c) Use the LU factorization to solve the linear system $Ax = b$ with $b = [1, 0, 0]^\top$ using one forward and one backward substitution.

(1.d) Use the LU factorization to compute the determinant of A. Recall that for two matrices of appropriate sizes, $\det(AB) = \det(A)\det(B)$.

(1.e) *** In the matrix A defined above, replace the $(2, 2)$-entry by 6. What is the rank of A after this modification? Attempt to compute the LU factorization of A. What do you observe? How might you "fix" the problem?

3.2 Block matrices and MATLAB matrix operations

(2.a) Let's practice creating matrices on the computer:

1. Create a matrix of all ones in your favorite coding language.
2. Create a matrix where its entries are independent standard Gaussian random variables (i.e.: with density $\mathcal{N}(0, 1)$).

(2.b) Now let's operate on these matrices:

1. Sum two matrices.
2. Multiply a matrix with an appropriately sized vector.
3. Multiply two matrices (Try some non-square matrices).

(2.c) Suppose we split up a matrix of size $\mathbb{R}^{(m+n)\times(m+n)}$ into blocks:

$$A = \left[\begin{array}{c|c} A_{11} & A_{12} \\ \hline A_{21} & A_{22} \end{array}\right]$$

where $A_{11} \in \mathbb{R}^{n\times n}$, $A_{22} \in \mathbb{R}^{m\times m}$, $A_{12} \in \mathbb{R}^{n\times m}$, and $A_{21} \in \mathbb{R}^{m\times n}$.

We have the block matrix multiplication formula:

$$AB = \left[\begin{array}{c|c} A_{11} & A_{12} \\ \hline A_{21} & A_{22} \end{array}\right] \cdot \left[\begin{array}{c|c} B_{11} & B_{12} \\ \hline B_{21} & B_{22} \end{array}\right] = \left[\begin{array}{c|c} A_{11}B_{11} + A_{12}B_{21} & A_{11}B_{12} + A_{12}B_{22} \\ \hline A_{21}B_{11} + A_{22}B_{21} & A_{21}B_{12} + A_{22}B_{22} \end{array}\right]$$

You could try to prove this at home. In session, we will verify this formula numerically by creating examples of A and B.

1. Make A and B to be Gaussian random matrices.
2. How do you compare two matrices numerically?

(2.d) (Challenge for you) Could you calculate the determinant of $A \in \mathbb{R}^{(m+n)\times(m+n)}$ by calculating the determinants of only m-by-m and n-by-n matrices?

Hint: try Gaussian elimination on block matrices. Then follow the procedure in (1.d).

3.3 Schur complement

Assume $M \in \mathbb{R}^{(m+n)\times(m+n)}$ and we split them into blocks

$$M = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$$

where $A \in \mathbb{R}^{n\times n}$, $D \in \mathbb{R}^{m\times m}$, $B \in \mathbb{R}^{n\times m}$, and $C \in \mathbb{R}^{m\times n}$. We also assume that M and all its leading submatrices are non-singular.

(3.a) Verify the formula

$$\begin{bmatrix} I & \\ -CA^{-1} & I \end{bmatrix} \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} A & B \\ & D - CA^{-1}B \end{bmatrix}$$

for "elimination" of the block C. The matrix $D - CA^{-1}B$ is known as the *Schur complement* of A in M.

(3.b) Explain the above decomposition as a form of "block LU".

Extra: Write down the block LDU decomposition.

3.4 Diagonally dominant matrix and pivoting

A matrix is called strictly (column) diagonal-dominant if the the absolute value of the diagonal entry in each column is larger than the sum of the absolute values of the other entries in that column; i.e., for all i:

$$|a_{ii}| > \sum_{j=1, j\neq i}^{n} |a_{ji}|$$

(4.a) Which of the following matrices is diagonally dominant?

$$B = \begin{bmatrix} -2 & 2 & 1 \\ 1 & 3 & 2 \\ 1 & -2 & 0 \end{bmatrix}, \qquad C = \begin{bmatrix} -4 & 2 & 1 \\ 1 & 6 & 2 \\ 1 & -2 & 5 \end{bmatrix}$$

(4.b) When computing the LU factorization of a strictly diagonally dominant matrix, why is pivoting never necessary?

1. First argue why the first column does not require pivoting. Then use Gaussian elimination to generate the required zeros in the first column
2. Show that, the submatrix you obtain when removing the first column and row is again strictly diagonally dominant.

(4.c) Let's show that an LU decomposition without pivoting exists in a different way:

1. Why are the leading principal submatrices of a strictly diagonally dominant matrix also strictly diagonally dominant?
2. Show that a diagonally dominant matrix is always invertible using the following argument: If A is not invertible, then there must exists a vector $\vec{v} \neq 0$ such that $A\vec{v} = \vec{0}$. Call r the largest (in absolute value) entry of $\vec{v}$ and consider multiplication of the r-th row.
3. Combine the previous two statements with a result from class to argue that the LU factorization of a strictly diagonally dominant matrix exists.

3.5 Calculating pivoted-LU

Compute by hand an LU factorization with pivoting ($PA = LU$) of the matrix:

$$A := \begin{bmatrix} -2 & 0 & 6 \\ -3 & 6 & 9 \\ -1 & 4 & 5 \end{bmatrix}.$$

Double-check your result using MATLAB's or Python's LU function!

Chapter 4

Conditioning and Stability

4.1 Matrix norms basics

(1.a) Compute $\|A\|_\infty$ and $\|A\|_1$ for the matrix

$$A = \begin{bmatrix} 1 & -1 & 2 & -3 \\ 7 & 2 & 3 & 5 \\ 2 & -4 & 3 & 8 \\ -3 & 5 & 3 & 1 \end{bmatrix}.$$

(1.b) Show that for symmetric positive definite (i.e., all eigenvalues are positive) matrices $A \in \mathbb{R}^{n\times n}$, the 2-norm condition number can also be computed as the ratio between the largest and the smallest eigenvalue of A, i.e.: $\kappa_2(A) = \lambda_{\max}/\lambda_{\min}$. Hint: Think about what the largest eigenvalue of A^{-1} is.

4.2 Norms Equivalency

Two norms in a finite-dimensional linear space X (e.g.: $\mathbb{R}^n$), $\|\cdot\|_a$ and $\|\cdot\|_b$, are called equivalent if there is a constant c such that for all x in X,

$$\|x\|_a \le c\,\|x\|_b\,, \qquad \|x\|_b \le c\,\|x\|_a\,. \tag{4.1}$$

(2.a) Suppose $\|\cdot\|_a$ and $\|\cdot\|_b$ are equivalent, and we know that an algorithm produces a sequence of vectors $\{e_n\}_{n\ge1}$, $\|e_n\|_a \to 0$ as $n \to \infty$. What could we conclude about $\|e_n\|_b$'s behavior for $n \to \infty$?

(2.b) We first show that the vector norms on $\mathbb{R}^n$, $\|\cdot\|_2$ and $\|\cdot\|_\infty$, are equivalent. To do this prove the inequality:

$$\|x\|_\infty \le \|x\|_2 \le \sqrt{n}\,\|x\|_\infty\,.$$

(2.c) The induced matrix norm on $\mathbb{R}^{n\times n}$: $\|\cdot\|_2$ and $\|\cdot\|_\infty$ are equivalent as well. Prove the inequality

$$\|A\|_\infty \le \sqrt{n}\,\|A\|_2\,,$$
$$\|A\|_2 \le \sqrt{n}\,\|A\|_\infty\,.$$

(2.d) (Challenge) Prove that: in a finite-dimensional linear space, all norms are equivalent; that is, any two satisfy (4.1) with some c, depending on the pair of norms [Lax07, p.217].

One inequality is relatively simple, the other one requires some big theorems from analysis. Read about the proof in Lax's book if you are interested.

4.3 Condition numbers based on different norms

(3.a) Let $A \in \mathbb{R}^{n\times n}$ be defined by

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Calculate $\kappa_1(A)$ and $\kappa_\infty(A)$. We see that a matrix can be well or ill-conditioned depending on the choice of norms.

(3.b) Indeed, we solve $A\boldsymbol{x} = \boldsymbol{b}$ and $A(\boldsymbol{x} + \Delta\boldsymbol{x}) = (\boldsymbol{b} + \Delta\boldsymbol{b})$ where

$$\boldsymbol{b} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}, \qquad \boldsymbol{x} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \text{and} \quad \Delta\boldsymbol{b} = \begin{bmatrix} \epsilon \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

Check that we have for both norms:

$$\frac{\|\Delta\boldsymbol{x}\|}{\|\boldsymbol{x}\|} \le \kappa(A)\frac{\|\Delta\boldsymbol{b}\|}{\|\boldsymbol{b}\|}.$$

4.4 Conditional number for the Hilbert matrix***

The Hilbert matrix $H \in \mathbb{R}^{n\times n}$ is a matrix with entries

$$h_{ij} = \frac{1}{i+j-1}.$$

(4.a) Using MATLAB or Python, compute the 2-norm-based condition numbers for $n = 3, 5, 10, 20, 25$.

(4.b) Let's consider a relative right hand side perturbation $\delta\boldsymbol{b}$ of a linear system with $\|\delta\boldsymbol{b}\|_2/\|\boldsymbol{b}\|_2 \approx 10^{-15}$. Write down the corresponding bounds $\|\delta\boldsymbol{x}\|_2/\|\boldsymbol{x}\|_2$ from the theory we discussed in class.

(4.c) Now, let's compute the actual error. Use the right-hand side vector with entries $b_i = \sum_{j=1}^{n}(j/(i+j-1))$ chosen such that the solution vector has entries $x_i = i$. Now, Compute the numerical solutions[1] $\boldsymbol{x}$, then re-compute $\boldsymbol{b} = H\boldsymbol{x}$ and compare the relative right-hand side error and the relative error in the solutions. How much are these better than the estimates you got from the condition number?

4.5 Condition numbers and pivoted LU

(5.a) Solve the matrix equation $A\boldsymbol{x} = \boldsymbol{b}$ with

$$A := \begin{bmatrix} 1 & 0 \\ 10^4 & 1 \end{bmatrix} \quad \text{and} \quad \boldsymbol{b} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

What is $\kappa_\infty(A)$?

Consider a small perturbation $\Delta\boldsymbol{b} = [10^{-3}, 0]^\top$ being added to the right-hand side, and solve again. Repeat with $\Delta\boldsymbol{b} = [0, 10^{-3}]^\top$. You should see that small perturbation can, but does not have to have a large effect even for badly conditioned systems.

(5.b) Verify the following LU decomposition of a matrix A without pivoting:

$$A := \begin{bmatrix} 10^{-4} & 1 \\ 1 & 1 \end{bmatrix} = LU = \begin{bmatrix} 1 & 0 \\ 10^4 & 1 \end{bmatrix} \begin{bmatrix} 10^{-4} & 1 \\ 0 & 1 - 10^4 \end{bmatrix}$$

We have seen in the previous problem that solving a system with the matrix L is sensitive to errors, i.e., it is poorly conditioned. However, the original A matrix is well-conditioned.

Now the LU factorization of A with pivoting is

$$PA = \begin{bmatrix} 1 & 1 \\ 10^{-4} & 1 \end{bmatrix} = LU = \begin{bmatrix} 1 & 0 \\ 10^{-4} & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & 1 - 10^{-4} \end{bmatrix}$$

We see that the LU factors with pivoting are better conditioned.

4.6 Condition number for solving linear system

(6.a) Find $\|A\|_2$ for the matrix

$$A = \begin{bmatrix} 1 & \varepsilon \\ \varepsilon & 1 \end{bmatrix},$$

where $\varepsilon \in (0, 1)$ (Hint: for symmetric matrices A, the eigenvalues of $A^T A$ are simply the squares of the eigenvalues of A).

[1] Note that all these computations contain tiny errors due to the final precision of computer computations.

(6.b) Continued from the previous item: Suppose that you have two systems

$$\begin{array}{l} x_1 + \varepsilon x_2 = b_1 \\ \varepsilon x_1 + x_2 = b_2 \end{array} \quad \text{and} \quad \begin{array}{l} \tilde{x}_1 + \varepsilon \tilde{x}_2 = \tilde{b}_1 \\ \varepsilon \tilde{x}_1 + \tilde{x}_2 = \tilde{b}_2 \end{array}$$

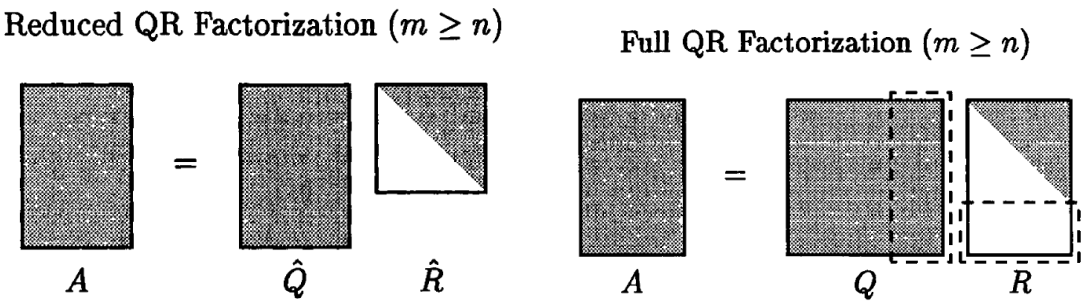
where $\tilde{\boldsymbol{b}} = (\tilde{b}_1, \tilde{b}_2)^T$ is approximately equal to $\boldsymbol{b} = (b_1, b_2)^T$, with a 5% relative error, that is $\frac{\|\tilde{\boldsymbol{b}} - \boldsymbol{b}\|_2}{\|\boldsymbol{b}\|_2} \leq 0.05$. Find an upper bound for the relative error $\frac{\|\tilde{\boldsymbol{x}} - \boldsymbol{x}\|_2}{\|\boldsymbol{x}\|_2}$ where $\tilde{\boldsymbol{x}} = (\tilde{x}_1, \tilde{x}_2)^T$ and $\boldsymbol{x} = (x_1, x_2)^T$. This upper bound will depend on ε.

Chapter 5

QR Factorization and Least Squares

5.1 Two forms of QR

(1.a) We have two forms of QR:



(1.b) We can interpret the formula for the solution of the least-squares problem

$$\hat{R}\boldsymbol{x} = \hat{Q}^\top \boldsymbol{b}$$

by using the full form of QR.

5.2 Projectors

A projector is a square matrix P that satisfies

$$P^2 = P.$$

(2.a) Assume P is a projector, and show that $I - P$ is also a projector.

(2.b) We can show that

$$\begin{aligned}&\text{range}(I-P)=\text{null}(P);\\&\text{null}(I-P)=\text{range}(P);\\&\text{range}(P)\cap\text{null}(P)=0.\end{aligned}$$

An orthogonal projector is a projector whose has the subspaces range(P) and null(P) orthogonal.

n.b.: An orthogonal projector P is not an orthogonal matrix! Why?

(2.c) Show that if $P=P^\top$ symmetric, the projector P is orthogonal (Hint: take one vector in range(P) and one in null(P), show that they must be orthogonal to each other).

The reverse direction holds as well. Therefore the two definitions are equivalent.

(2.d) A special case of orthogonal projection is the projection onto a vector:

$$P_v=\frac{\boldsymbol{v}\boldsymbol{v}^\top}{\boldsymbol{v}^\top\boldsymbol{v}}.$$

Show that it is indeed an orthogonal projector with range span($\boldsymbol{v}$).

(2.e) Another orthogonal projection is

$$P_{\perp v}=I-\frac{\boldsymbol{v}\boldsymbol{v}^\top}{\boldsymbol{v}^\top\boldsymbol{v}}.$$

What is its null space? What is its range?

5.3 Geometric interpretation of Householder reflectors

(3.a) Name $H(\boldsymbol{v})$ the linear subspace orthogonal to the vector $\boldsymbol{v}$. A reflector across $H(\boldsymbol{v})$ is

$$F_{\boldsymbol{v}}=I-2\frac{\boldsymbol{v}\boldsymbol{v}^\top}{\boldsymbol{v}^\top\boldsymbol{v}}.$$

Compare this with P_v and $P_{\perp v}$, and interpret the formula geometrically.

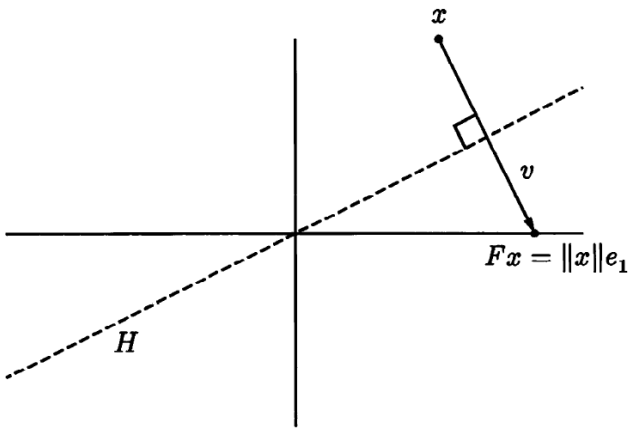


Figure 10.1. *A Householder reflection.*

(3.b) To use Householder for QR decomposition, we want $F\boldsymbol{x} = c\boldsymbol{e}_1$. We know that $c = \pm \|\boldsymbol{x}\|_2$. Explain this geometrically.

(3.c) From this, we have that

$$\boldsymbol{v} = \boldsymbol{x} - F\boldsymbol{x} = \boldsymbol{x} \pm \|\boldsymbol{x}\| \, \boldsymbol{e}_1.$$

From a geometric point of view, why is this the correct formula?

5.4 QR decomposition via Householder

(4.a) Construct the QR factorization of the following matrix using Householder reflectors (the algebra is not simple. You might use MATLAB as a calculator):

$$A = \begin{bmatrix} 1 & 0 & 1 \\ 2 & 1 & 3 \\ 0 & 2 & 4 \end{bmatrix}.$$

(4.b) Use the factorization to determine $|\det(A)|$

5.5 Least squares and infections disease

Let us assume an infectious disease with the following reported new infections I_i on each day t_i, for $i = 1, \ldots, 10$. Using least squares fitting, we would like to understand the nature of this growth. We consider two models to describe the connection between time (i.e., days) t and the number of new infections, both with 3 unknown parameters (a, b, c):

$$I(t) = a + bt + ct^2 \qquad \text{(polynomial model)}$$

Table 5.1: Number of new infections I_i on days t_i.

t_i:	1	2	3	4	5	6	7	8	9	10
I_i:	14	20	21	24	15	45	67	150	422	987

$$I(t) = a + bt + c\exp(t) \qquad \text{(exponential model)}$$

Our goal is to figure out which model describes the progression of the infections better, and we use least squares fitting to figure that out. Note that if a model would fit the data perfectly, $I(t_i) = I_i$ for all i. In general, you will not be able to find parameters that satisfy this, and thus have to use least squares fitting (sometimes this is also called *regression*).

(5.a) Formulate, assuming the polynomial model, the least squares problem for the parameters $\boldsymbol{x} = [a, b, c]^T$ by specifying the matrices A and the vector $\boldsymbol{b}$:

$$\min_{\boldsymbol{x}\in\mathbb{R}^3} \|A\boldsymbol{x} - \boldsymbol{b}\|_2^2$$

(5.b) Same as above, but for the exponential model.

(5.c) Use a QR-factorization in MATLAB or Python to solve these problems and plot the data as points, as well as the model as a line. Repeat using the normal equations $A^TA\boldsymbol{x} = A^T\boldsymbol{b}$.

(5.d) To decide which model describes the data better, we need to compute the distance between the model and the data points. Take a look at the proof from class for how the QR factorization can be used to solve least squares problems. In particular, we found that:

$$\|A\boldsymbol{x} - \boldsymbol{b}\|_2^2 \geq \|\boldsymbol{b}_2\|_2^2,$$

where $\boldsymbol{b}_2 = \hat{\hat{Q}}^\top\boldsymbol{b}$. We also found that this inequality is equality if $\boldsymbol{x}$ solves the least squares problem. Thus, the norm or $\boldsymbol{b}_2$ is a measure of how well the model fits the data. Use this to decide which of the two models above describes the data better.

Chapter 6

Eigen-Problems

6.1 Gershgorin disks and the power method

Consider the matrix

$$A = \begin{bmatrix} -6 & 2 & 0.3 & 0 & -0.7 \\ 2 & -4 & 0.1 & 0.05 & 0 \\ 0.3 & 0.1 & 2 & 0.1 & 0.1 \\ 0 & 0.05 & 0.1 & 4 & 0 \\ -0.7 & 0 & 0.1 & 0 & 6 \end{bmatrix}$$

and recall the definition of the Gershgorin disks:

$$D_i = \{z \in \mathbb{C} \mid |z - a_{ii}| \leq \sum_{j \neq i} |a_{ij}|\}.$$

(1.a) Argue that all eigenvalues of A are real.

(1.b) What are the Gershgorin disks for A? Use them to give a set, $D \subset \mathbb{R}$, that contains all eigenvalues of A.

(1.c) Can you conclude that the eigenvalue with the largest absolute value is simple?

(1.d) Argue that A is invertible. Conclude that all diagonally dominant matrix is invertible.

(1.e) True or False? Let $A \in \mathbb{R}^{n \times n}$ and D_i, $i = 1, 2, \ldots, n$, be the Gerschgorin disks of A. If $0 \in \bigcup_{i=1}^{n} D_i$ then A is singular.

(1.f) Write down the first iteration of the power method starting from $\boldsymbol{x}_0 = (0, 0, 0, 0, 1)^T$. You don't need to normalize. Explain why $\boldsymbol{x}_0 = \mathbf{0}$ is not a suitable starting point.

(1.g) The eigenvalues of A, after rounding, are $\{-7, -3, 2, 4, 6\}$. Which eigenvalue direction will the sequence of the previous question converge to?

6.2 Eigenvectors as stationary points of Rayleigh quotient

For H a real symmetric matrix, we define Rayleigh quotient as a function $\mathbb{R}^n \to \mathbb{R}$:

$$R(\boldsymbol{x}) = \frac{\boldsymbol{x}^\top H \boldsymbol{x}}{\boldsymbol{x}^\top \boldsymbol{x}} = \frac{q(\boldsymbol{x})}{p(\boldsymbol{x})}.$$

We will show that $\boldsymbol{v}$ is a stationary point (i.e.: $\nabla R(\boldsymbol{v}) = 0$) of the Rayleigh quotient if and only if it is an eigenvector of H (cf. [Lax07, p.114-116] and [TB97, p.203-204]).

(2.a) To characterize a point such that $\nabla R(\boldsymbol{v}) = 0$, we need to know the gradient of $R(\boldsymbol{x})$ at $\boldsymbol{v}$. We could do this, but an alternative approach is to take $t \in \mathbb{R}$ and calculate

$$\frac{\mathrm{d}}{\mathrm{d}t} R(\boldsymbol{v} + t\boldsymbol{y})\Big|_{t=0}$$

for all $\boldsymbol{y} \in \mathbb{R}^n$. In particular, we can get the gradient by picking $\boldsymbol{y} = \boldsymbol{e}_i$.

(2.b) Using the above calculation, show the iff claim in the main text of the problem.

6.3 Computing eigenvalues via the Power Iteration

Given is the following matrix:

$$A = \begin{bmatrix} -2 & 1 & 4 \\ 1 & 1 & 1 \\ 4 & 1 & -2 \end{bmatrix},$$

It has eigenvalues and eigenvectors:

$$\lambda_1 = 0,\ \boldsymbol{v}_1 = \begin{bmatrix} 0.41 \\ -0.82 \\ 0.41 \end{bmatrix}, \quad \lambda_2 = -6,\ \boldsymbol{v}_2 = \begin{bmatrix} 0.71 \\ 0.0 \\ -0.71 \end{bmatrix}, \quad \lambda_3 = 3,\ \boldsymbol{v}_3 = \begin{bmatrix} -0.58 \\ -0.58 \\ -0.58 \end{bmatrix}.$$

(3.a) Calculate the first iterate of the power method when $\boldsymbol{x}_0 = (0, 1, 1)^T$.

(3.b) Which eigenvalue direction will the sequence defined in (3.a) converge to?

(3.c) Give an initialization vector such that the power method does *not* converge to the direction of the largest (in absolute value) eigenvalue.

(3.d) Write a simple program implementing the power method for the matrix A.

1. Use the Rayleigh quotient to calculate estimates of the eigenvalues for each iteration.
2. What is the order of convergence of the eigenvector estimates and the eigenvalue estimates? What is the speed of convergence?
3. Could you explain the relationship between the two convergence speeds?
   (Hint: last week we showed that eigenvectors $\boldsymbol{v}$ are stationary points of the Rayleigh quotient)

6.4 The Inverse Iteration

Take A to be the matrix above, and let $\theta \in \mathbb{R}$ and let $\boldsymbol{x}_0 \in \mathbb{R}^3$.

(4.a) Define the *Inverse Iteration* (also called *Inverse Power Method*) to calculate eigenvectors of A near θ.

(4.b) If $\theta = 2$, where will the sequence defined in (i) converge to and why?

(4.c) If $\theta = -2$, where will the sequence defined in (i) converge to and why?

(4.d) Write a simple program implementing the inverse power method. Do the same sub-tasks as the ones in (3.d).

6.5 The Rayleigh Quotient Iteration

It is irresistible to use the eigenvalues estimates from the Rayleigh quotient to update θ for each step in the inverse iteration. The resulting algorithm is the Rayleigh Quotient Iteration [TB97]:

Algorithm 27.3. Rayleigh Quotient Iteration

$v^{(0)}$ = some vector with $\|v^{(0)}\| = 1$
$\lambda^{(0)} = (v^{(0)})^T A v^{(0)}$ = corresponding Rayleigh quotient
for $k = 1, 2, \ldots$

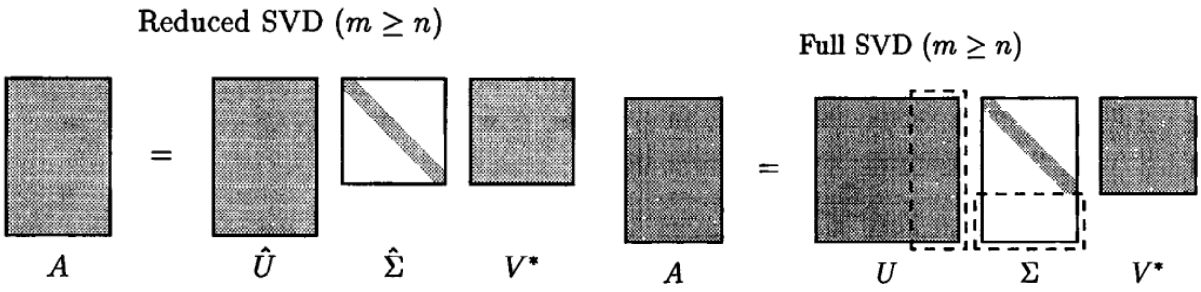
Solve $(A - \lambda^{(k-1)} I) w = v^{(k-1)}$ for w	apply $(A - \lambda^{(k-1)} I)^{-1}$
$v^{(k)} = w / \|w\|$	normalize
$\lambda^{(k)} = (v^{(k)})^T A v^{(k)}$	Rayleigh quotient

(5.a) Implement the Rayleigh Quotient Iteration.

1. What is the order of convergence of the eigenvector estimates and the eigenvalue estimates?
2. Could you explain this (high) order of convergence?

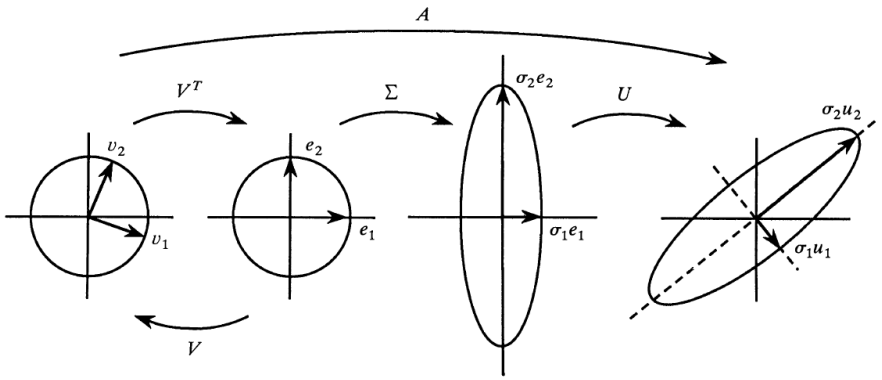
6.6 Singular Value Decomposition (SVD) basics

(6.a) Like QR, SVD has the full and reduced form [TB97]:



Note that the full form has U which spans the whole of $\mathbb{R}^n$.

(6.b) We can interpret the full form of SVD as a change of basis, then a scaling, and then another change of basis (Figure from [Str93]).



6.7 Some properties of SVD

We list (and attempt to prove) some properties of SVD:

(7.a) $\text{range}(A) = \text{span}(\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_r)$ and $\text{null}(A) = \text{span}(\boldsymbol{v}_{r+1}, \ldots, \boldsymbol{v}_n)$.

(7.b) The rank of A is r, the number of nonzero singular values.

(7.c) We have $\|A\|_2 = \sigma_1$, the largest singular value. And $\|A\|_F = \sqrt{\sigma_1^2 + \sigma_2^2 + \cdots + \sigma_r^2}$. You will show the second property in your homework.

6.8 Low-rank approximation using SVD

(8.a) Show that A is the sum of r *rank-one* matrices:

$$A = \sum_{j=1}^{r} \sigma_j \boldsymbol{u}_j \boldsymbol{v}_j^\top .$$

(8.b) For any ν with $0 \le \nu \le r$, define

$$A_\nu = \sum_{j=1}^{\nu} \sigma_j \boldsymbol{u}_j \boldsymbol{v}_j^\top .$$

Then we have

$$\|A - A_\nu\|_2 = \inf_{\substack{B \in \mathbb{R}^{m \times n} \\ \operatorname{rank}(B) \le \nu}} \|A - B\|_2 = \sigma_{\nu+1}.$$

Note that a similar theorem is also true for the Frobenius norm.

Chapter 7

Interpolations and Quadrature

7.1 Polynomial interpolation and linear algebra

In lecture you have learned the theorem which states:

For $n \geq 1$ and distinct $n+1$ data pairs $(x_0, y_0), \ldots, (x_n, y_n)$ there exists a unique $p_n(x) \in P_n$, an n-th order polynomial such that $p_n(x_i) = y_i$ for $i = 0, \ldots, n$.

We will try to prove this using linear algebra.

(1.a) We can write an n-th order polynomial as

$$p_n(x) = a_0 + a_1 x + \cdots + a_n x^n.$$

This gives us $n+1$ free variables to solve. Frame the problem of finding $p_n(x)$ s.t. $p_n(x_i) = y_i$ for $i = 0, \ldots, n$ as a matrix problem $X\boldsymbol{a} = \boldsymbol{y}$.

(1.b) Show that since $x_i \neq x_j$ for all i, j, we have the matrix X we constructed has full rank.

(1.c) Think about the uniqueness and existence claim in the theorem in linear algebra language.

7.2 Interpolation and quadrature basics***

- True or False? For the nodes $x_0 = 0, x_1 = 1, x_2 = 2$, the Lagrange interpolation polynomial $L_0(x)$ is $-x^2 + 1$.
- True or False? We compute the Hermite interpolant with 3 distinct nodes of a function f that is a polynomial of degree 4. Then this Hermite interpolant is identical to f. (In short: Hermite interpolation with 3 nodes is exact for polynomials of degree 4.)
- True or False? Hermite interpolation with 4 distinct nodes is exact for polynomials of degree 6.

- True or False? Gauss quadrature with 3 integration nodes (and corresponding weights) is exact for polynomials of degree 7.
- A Gauss quadrature rule with how many points is required to integrated polynomials of degree 10 exactly?
- What is the result obtained with Simpson's rule for integrating $f(x) = x^2$ over the interval $[0, 1]$?
- True or false: Let p_n be the Lagrange interpolant to a function f with $n + 1$ interpolation points, and $e_n(x) = |p_n(x) - f(x)|$. The interpolation error $\|e_n\|_\infty$ *always* gets arbitrarily small for large n, i.e., $\|e_n\|_\infty \to 0$ as $n \to \infty$.

7.3 Lagrange interpolation polynomial example

Let $x_0, \ldots, x_n$ be distinct interpolation nodes, and let

$$p_n(x) = \sum_{k=0}^{n} L_k(x)(x_k)^j,$$

where j is an integer and $n \geq j > 0$. What is the $p_n(x)$ function? What are the values of $p_n(0)$ and $p_n(1)$?

7.4 Hermite interpolation polynomial example

Recall that the Hermite interpolation of a function f at the points x_0, x_1, x_2 has the form

$$p(x) = \sum_{j=0}^{2} H_j(x) f(x_j) + \sum_{j=0}^{2} K_j(x) f'(x_j).$$

(4.a) Show that the polynomial

$$-\frac{1}{\pi}x^2 + x$$

is the Hermite interpolation polynomial of $f(x) := \sin(x)$ based on the nodes $x_0 = 0$, $x_1 = \pi$.

(4.b) Show that the polynomial $K_2(x)$ in this representation for $x_0 = 0, x_1 = 1, x_2 = 2$ is given by

$$\frac{1}{4}x^5 - x^4 + \frac{5}{4}x^3 - \frac{1}{2}x^2.$$

7.5 Error bound for interpolation***

This requires MATLAB or Python: we interpolate the function $f : [0, 1] \to \mathbb{R}$ defined by $f(x) = \exp(3x)$ using the nodes $x_i = i/2$, $i = 0, 1, 2$ by a quadratic polynomial $p_2 \in \boldsymbol{P}_2$. Compare the

exact interpolation error $E_f(x) := f(x) - p_2(x)$ at $x = 3/4$ with the estimate

$$|E_f(x)| \leq \frac{M_{n+1}}{(n+1)!}|\pi_{n+1}(x)|,$$

where $M_{n+1} = \max_{z\in[0,1]} |f^{(n+1)}(z)|$, $f^{(n+1)}$ is the $(n+1)$st derivative of f, and $\pi_{n+1}(x) = (x - x_0)(x - x_1)(x - x_2)$.

7.6 Deriving a new quadrature rule

Given $f : [0, 1] \to \mathbb{R}$, you want to derive a new quadrature rule that does uses not only function values, but also gradient values:

$$\int_0^1 f(x)\,\mathrm{d}x \approx \alpha_0 f(0) + \alpha_1 f'(0) + \alpha_2 f(1). \tag{7.1}$$

(6.a) First, find polynomials $J_0, J_1, J_2 \in \mathcal{P}_2$, with the following properties:

$$\begin{aligned} J_0(0) &= 1, \quad J_0'(0) = 0, \quad J_0(1) = 0 \\ J_1(0) &= 0, \quad J_1'(0) = 1, \quad J_1(1) = 0 \\ J_2(0) &= 0, \quad J_2'(0) = 0, \quad J_2(1) = 1. \end{aligned}$$

(*Hint:* For each J_i, make an ansatz for a quadratic polynomial using the monomial basis.)

Given f, you can now define a polynomial approximation $p \in \mathcal{P}_2$ via

$$p(x) = f(0)J_0(x) + f'(0)J_1(x) + f(1)J_2(x). \tag{7.2}$$

The polynomial p is an approximation to f in the sense that $p(0) = f(0)$, $p'(0) = f'(0)$ and $p(1) = f(1)$.

(6.b) Use the polynomial p derived in (7.2) and the same method used to derive the Newton-Cotes quadrature rules, to find the coefficients α_0, α_1 and α_2 in (7.1).

(6.c) Use your new quadrature rule to approximate $\displaystyle\int_0^1 \exp(2x)\sin^2(x)\,\mathrm{d}x$, and also compare with Simpson's rule. The exact value of this integral is $1.2668\ldots$.

7.7 Trapezoidal rule for smooth periodic functions

We investigate how the (composite) trapezoidal rule performs for smooth, periodic functions. Consider integrating the smooth, periodic function $f(x) = e^{\sin x}$ over a single period. The exact value of the integral is

$$I(f) = \int_0^{2\pi} e^{\sin x}\,\mathrm{d}x = 7.95492652101284527\ldots.$$

(7.a) Write down the composite trapezoidal rule $T_N(f)$ on equispaced nodes $0 = x_0 \leq \cdots \leq x_N = 2\pi$ for estimating the value of this integral.

(7.b) Simplify your expression for $T_N(f)$ using the periodicity of f.

(7.c) Show that $T_N(f)$ is equivalent to both a left-endpoint Riemann sum and a right-endpoint Riemann sum approximation to $I(f)$.

(7.d) Compute $T_N(f)$ for various progressively larger N. Plot the quadrature errors against N on (i) a log-log plot, and (ii) a `semilogy` plot. What is the order of accuracy of the trapezoidal rule for smooth, periodic functions?

7.8 Convergence order of quadrature

(8.a) We would like to integrate a function on $[0, 1]$ using the composite trapezoid rule with sub-interval size h. We name the result T_h. How does the error ($e_h = |T_h - I|$) scale with h?

(8.b) In real application, we do not know the true value of the integral. To verify the order of convergence of our code, we can calculate this quantity:

$$\frac{T_h - T_{h/2}}{T_{h/2} - T_{h/4}}.$$

How does this quantity scale with h?

(8.c) We will show this using the code from last session.

7.9 Inner product

(9.a) Let's define a new inner product in $\mathbb{R}^3$ with a symmetric positive-definite matrix $W \in \mathbb{R}^{3\times 3}$ as follows:

$$\langle \boldsymbol{u}, \boldsymbol{v} \rangle_W := \boldsymbol{u}^\top W \boldsymbol{v}$$

Show that this defines an inner product on $\mathbb{R}^3$ and write down the induced norm. If

$$W = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix},$$

give a vector that is orthogonal to $[1, 0, 0]^\top$ in the W-inner product.

(9.b) Now let's consider a symmetric positive semi-definite matrix. Does the above definition still define an inner product and why/why not?

7.10 Orthogonal polynomial and Gauss quadrature

We assume an inner product on $[-1, 1]$ with weight $\omega(x) := 1 - x^2$, i.e.,

$$\langle p, q \rangle = \int_{-1}^{1} \omega(x) p(x) q(x) \, \mathrm{d}x.$$

We define the following polynomials, which are orthogonal with respect to this inner product:

$$\varphi_0(x) = 1, \quad \varphi_1(x) = 2x, \quad \varphi_2(x) = 5x^2 - 1$$

(10.a) Verify that φ_0 and φ_1 are orthogonal on $[-1, 1]$ with respect to the weight ω.

(10.b) Are the φ_j's orthonormal under this inner product?

(10.c) Find the polynomial $p \in \boldsymbol{P}_2$ for which $\displaystyle\int_{-1}^{1} (1 - x^2)(p(x) - x^3)^2 \, \mathrm{d}x$ is minimal.

(10.d) *** Use this family of orthogonal polynomials to compute the quadrature nodes of a 2-point Gauss quadrature rule. Compute the corresponding quadrature weights by using that constant and linear functions must be integrated exactly by this rule.

(10.e) *** Up to what polynomial order is this quadrature rule exact for integrals over $[-1, 1]$ weighted by ω?

7.11 More Gauss quadrature***

Let $\omega : [-1, 1] \to \mathbb{R}_{>0}$ be a weight function for the inner product $\langle f, g \rangle := \int_{-1}^{1} \omega(x) f(x) g(x) \, dx$. Assume that ω is continuous and strictly monotonically increasing. We want to compute a Gauss quadrature formula with a single node x_0. Show that $x_0 > 0$.

Bibliography

[BF10] Richard L. Burden and J. Douglas Faires. *Numerical Analysis*. Cengage Learning, August 2010.

[Lax07] Peter D. Lax. *Linear Algebra and Its Applications*. John Wiley & Sons, September 2007.

[Str93] Gilbert Strang. The Fundamental Theorem of Linear Algebra. *The American Mathematical Monthly*, 100(9):848–855, 1993.

[TB97] Lloyd N. Trefethen and David III Bau. *Numerical Linear Algebra*. SIAM, June 1997.