1 Lecture 1

Rounding, Approximation, Well Posedness

Definition 1.1. We say that a problem is well posed if its solution exists, is unique and depends continuously on the inputs.

Definition 1.2. Rounding error refers to our representation of real numbers as floating point numbers.

Definition 1.3. Truncation error refers to our representation of infinite collections as finite ones. For example, the truncation of a Taylor series to a finite quantity of terms is an example of truncation.

Definition 1.4. Approximating a solution can be divided into two stages: approximation before the calculation and approximation during the calculation. Approximation before the calculation includes the following:

- How we model the problem.
- The accuracy of our input data.
  - How we measure the input data
  - How we compute the input data

Approximation during the calculation includes:

- Rounding Error
- Truncation Error

Remark 1.5. Recall that norms allow for constant pullout, satisfy the triangle inequality and output 0 if and only the input to the norm is also 0.

Example 1.6. It is easy to undestand rounding and truncation error. The first set of approximations can be understood as follows:

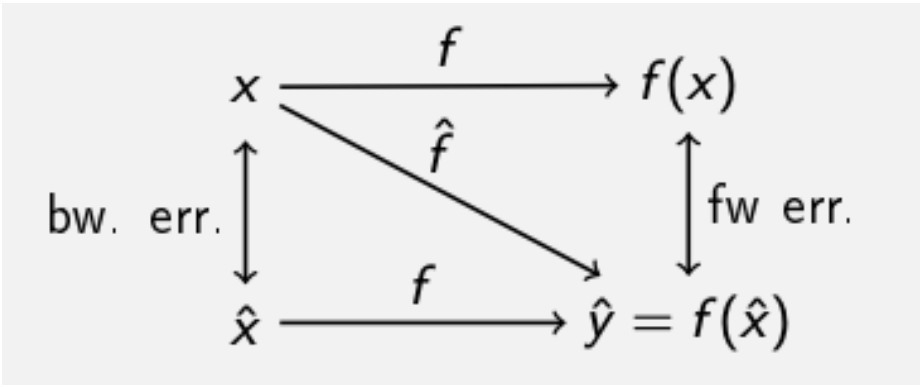
Suppose we use the surface area of the earth as an input. To calculate the surface area, we first assume that the earth is a sphere, then we somehow measure the radius and then we compute that area by multiplying the radius squared by π.

2 Lecture 2

Rounding and Error

Definition 2.1. Let x be some input, f some true function and $\hat{f}$ some estimated function. Absolute forward error is the difference $\left|f(x) - \hat{f}(x)\right|$. Relative forward error is $\frac{\left|f(x)-\hat{f}(x)\right|}{|f(x)|}$.

Definition 2.2. Backward error is the difference between the original input and the input that, under the true function f, would have given you $\hat{f}(x)$, the output of the original input under the approximate function.



Remark 2.3. Suppose I present to you two problems p_1 and p_2 that have the same forward error but the backward error of p_1, $b(p_1)$ is such that $b(p_1) > b(p_2)$. Is problem 1 or 2 more favorable? Is any more favorable? We can generally say that p_1 is more favorable, because it takes a larger input perturbation in p_1 to produce, under a true function f, the same output perturbation as p_2. This would suggest that p_1 is more *well conditioned*.

Remark 2.4. Truncation and rounding error make up the ways in which an operation is approximate. By truncation error, we refer to the process of replacing infinite sums and expressions with finite ones and by rounding error, we refer to the error introduced in representing real numbers with computer based number systems.

Lemma 2.5. This is sort of a lemma:

The relative error in the forward error tells us how many digits are significant in our estimate of the solution to a problem.

Example 2.6. Suppose we estimate $\sqrt{2} = 1.414\ldots$ as 1.4. Then the negative log of the relative error is $- \approx \log(0.01) = 2$, meaning that (correctly) our estimate is accurate up to 2 digits.

Definition 2.7. The condition number is

$$\sup_{x \in \text{domain}} \frac{\text{Relative Forward Error}}{\text{Relative backward error}}$$

Theorem 2.8. Assuming that a true function f is differentiable, then its condition number is given by

$$\sup_{x \in \text{Domain}} \frac{xf'(x)}{f(x)}$$

Proof.

$$\begin{aligned}\Delta y = \frac{f(x+\Delta x)-f(x)}{1} &= f'(x)\Delta(x)\\ \implies \frac{\Delta y}{y}\Big/\frac{\Delta x}{x}&\\ = \frac{f'(x)\Delta x}{y}\Big/\frac{\Delta x}{x}&\end{aligned}$$

and the latter simplies to the claimed expression. □

Problem Forward Error. Approximate $f(x) = 1/x$ by the taylor expansion $1-(x-1)$. What is the forward error when we let $x = 0.5$?

It is 0.5

Problem Backward Error. What about the backward error if we use $x = 0.5$

We need to find the value x' such that $f(x') = 1.5$. That is $x' = \frac{2}{3}$. Therefore, $|x - x'| = 0.16$.

Problem Condition Number. Determine the condition number of $\sin(x)$ on the interval $[0, \pi/2]$.

Proof. The formula gives us that $k(f) = \sup_{x\in[0,\pi/2]} \frac{x\cos x}{\sin x}$. Note that this is a decreasing function on. the given interval, so that the largest value is taken on at $x = 0$. By L'Hopital's rule, that gives us:

$$\frac{\cos x - x\sin x}{\cos x}.$$

which evaluates to 1 at $x = 0$. Thus, the condition number is 1. □

Problem Larger Condition Number. Over which interval is the condition number larger: $[0, 2\pi]$ or $[10^4\pi, 10^4\pi + 2\pi]$.

Proof. Recall that the formula is $\frac{xf'(x)}{f(x)}$, and that $f'(x)$ and $f(x)$ will assume the same values up to equivalence classes of $\sin$. Thus, the x in the numerator accounts for any potential difference, making the second interval have a greater condition number. □

Definition 2.9. A problem is well conditioned if, when considering a problem, small perturbations in an input do not result in large perturbations in the problem's output.

Definition 2.10. A method is stable if, when using it to solve a problem, the solution produced by the method is an exact solution to a nearby problem (ie the same problem applied to a different input). In other words, the error introduced by the method is no worse than the introduction of a little error into the input data.

Remark 2.11. We see that conditioning describes a problem, whereas stability describes a method used to solve that problem.

Definition 2.12. A method is accurate if it produces solutions that are very close to the true answer. A solution is accurate if both the method is stable and the problem well conditioned. If a problem has a stable algorithm but is not itself well conditioned, then the solution may not necessarily be accurate.

3 Lecture 3: Floating Point

The following questions were prelecture questions:

Problem Operator Conditioning. Let $x > 1$. Which of $f(x) = 3x, f(x) = 3/x, f(x) = 3 - x, f(x) = \sqrt{x}$ is ill posed (here, make the restrictive definition that a well posed problem must have a finite condition number.?

Proof. We can compute the condition numbers using the formula $xf'(x)/f(x)$ since all functions are differentiable over the domain $x > 1$. This gives:

- $$\frac{3x}{3x} = 1$$
- $$\frac{x(-3/x^2)}{3/x} = \frac{(-3/x)}{3/x} = -1$$
- $$\frac{x(-1)}{3-x} \to \infty \text{ as } x \to 3$$
- $$\frac{-1/2}{\sqrt{x}}$$

□

Problem Forward and Backward Errors. Approximate $\sqrt{1+x}$ by $1+x/2$. What is the absolute backward error using $x = 8$.

Proof. To get $1 + 8/2 = 5$ using f, we need to let $x = 24$. Hence, the absolute backward error is $|8 - 24| = 16$. □

Problem Why choose a stable algorithm?. A stable algorithm always:

- produces an accurate result
  - False. If a problem is additionally well conditioned, then a stable algorithm applied to it may result in an accurate solution; otherwise, no
- insensitive to data error
  - No, a stable algorithm only assures us that an answer is a solution to a nearby problem
- improves the conditioning of the problem
  - No, this is distinct from stability. See the first retort.

- gives the correct answer to a nearby problem
  - This is exactly the definition.

Problem Condition Number of Square Root. At $x = 7$?

Proof.

$$\frac{1/2}{\sqrt{x}}x \Big/ \sqrt{x} = 1/2$$

□

Problem Backward Error. Given $x = 0.5$, approximate $\cos x$ by $1 - \frac{x^2}{2}$. What is the backward error?

Proof. Find $\hat{x}$ such that $\cos \hat{x} = 0.875$. Subtract that from 0.5. □

Problem Estimate Condition Number. Estimate $f(x) = \log(1 + x)$ (remember that log means natural log) with $x - \frac{x^2}{2} + \frac{x^3}{3}$. Suppose we know that $f(0.5168967) = \hat{f}(0.5)$. Then we know that the relative backward error is $\frac{0.516897-0.5}{0.5}$. Now use this information to approximate the condition number of f.

Proof. We already know the relative backward error, and the condition number is relative forward error over relative backward error. Hence, the condition number estimate is

$$\frac{\hat{f}(0.5) - f(0.5)}{f(0.5)} \Big/ \text{Relative Backward Error}$$

□

Definition 3.1. Suppose that we represent numbers using 64 bits. 32 bits will be dedicated to the whole portion of any number – we will progress under this system, going from rightmost bit to left most bit, from 2^0 to 2^32; similarly, 32 bits will be dedicated to the fractionalpart of any number – we will progress from rightmost bit to leftmost bit from 2^-32 to 2^-1.

Remark 3.2. Suppose that we wish to represent 2^{32}. Observe that largest number representable using this scheme is $2^{32} - 2^{-32}$.

It follows that the relative error of our represntation of 2^{32} is

$$\frac{2^{-32}}{2^{32}} = 2^{-64}$$

Now suppose that we wish to represent some number x in the range $(2^{-32}, 2^{-31})$. Assuming that 2^{-32} is closer to this number then 2^{-32}, the relative error of this representation is

$$\frac{\left|x - 2^{-32}\right|}{|x|} \leq \frac{\left|2^{-33}\right|}{|x|} \leq \frac{\left|2^{-33}\right|}{2^{-32}} = 2^{-1}$$

Observe that whereas the relative error in the former is good, in the latter it is poor. This motivates our definition of the double precision floating point system.

Example 3.3. Represent 13 in scientific notation in a binary number system. Require that the first and only digit before the decimal point (remember that this is scientific notation) is a 1; further require that 1 must occupy this position. You will find that the representation is uniquely $(1.101)_2 * 2^3$.

Remark 3.4. Suppose I told you that we we will represent all numbers as we did above. Leaving aside how we would represent 0, observe that since we require that the first and only digit before the decimal point is a 1, if we want to store such representations, there is no point in storing the 1. This allows us to say the following:

Definition 3.5.

- The double precision floating point system allocates 52 bits to store the fractional part of a number represented using the scheme above, where a 1 in the units place implicitly precedes the fractional part.
- 11 bits are allocated to store an exponent and 1 bit is allocated to store a sign bit.
  - The 11 exponent bits allow us to get integers in the range $[0, 2047]$. Assuming that we begin with an offest of -1023, meaning that 0 maps to -1023, we can represent an exponent in the range $[-1023, 1024]$.
- When the exponent becomes 0, instead of mapping to -1023, we to -1022 (yes, this contradicts the previous line, but introducting the contradiction and then correcting it is pedagogically more gradual). We also do away with the implicit 1 that precedes the fractional part of the significand (recall tha the fractional part of the significand is the only part which is stored in our representation). 0 is represented if we let all bits of the fractional part be 0.
  - Since we are no longer required to have an implicit, leading 1, notice that we can now drop down past $1 * 2^{-1022}$. Represent 2^{-1023} as $0.100\cdots \times 2^{-1022}$; represent 2^{-1024} as $0.0100\cdots \times 2^{-1022}$ and so forth. UNRESOLVED
- We map 2^{1024} to ∞ and if the fractional part contains any non-zero digit, we call that values ∞ instead. UNRESOLVED
- Define the UFL and OFL to be the smallest and greatest normal numbers that are not subnormal nor ∞.

Definition 3.6. Suppose that we have a number $x = b.\underbrace{b_1 b_2 \ldots b_{52}}_{\text{all useable digits consumed}} d_1 d_2 d_3$.

We need to somehow represent x in our floating point system. To do so, we round on the basis of $d_1 d_2 d_3$. Now we could make the scheme round to nearest, meaning that add 1 to b_{52} if $d_1 d_2 d_3 > 100$ and do nothing if $d_1 d_2 d_3 < 100$, but we must then choose what to do when $d_1 d_2 d_3 = 100$. What then? If we make the consistent, arbitrary decision to round up, we will produce statistical noise. Thus, the scheme employed is to round to even:

If b_{52} is 0, then do nothing, so that x is even; if b_{52} is 1, then add 1 to b_{52}.

Remark 3.7. Some floating point exercises `https://relate.cs.illinois.edu/course/cs450-s19/flow/inclass-floating-point/start/`

Definition 3.8. Let ϵ be the smallest number such that $\text{fl}(x(1+\epsilon)) > x$. If we use round to even, it is 2^{-52}; if we use round to nearest, with rounding up in case of a tie, it is 2^{-53} – in which case we also distinguish 2^{-52} with some title.

Lemma 3.9. The relative error in representing any number x within the normal range is at most ϵ.

Proof. Given x, there is an a such that

$$2^a \leq x \leq 2^{a+1}$$

Since we can represent any number $y = 2^a + \epsilon * n$ up to $y = 2^{a+1}$, it follows that there is at least one representable number in the range $[x, x+\epsilon]$. This allows us to say that the absolute error between this number and x is at most ϵ, which is formalized below:

$$\frac{|x - (1+\epsilon)x|}{|x|} = \frac{|x|\epsilon}{|x|} = \epsilon$$

□

4 Lecture 4

4.1 Quiz

Problem 1.

Estimating Output Error with Condition Numbers A function f(x) has a condition number of 2.3×10^2. For x=65 with an absolute error Δx=5×10−5, what is an upper bound on the relative error of the output?

TIP: On any quiz/exam question where you are asked to enter a number, you can also enter an expression, which will be evaluated for you. No need to bring out the calculator. (I.e. in the box below, you could simply enter "100*13", which would count the same as entering "1300".)

Make sure your answer has at least two accurate digits.

Solution 4.1. Relative Error <= Relative Input Error * Condition Number Relative Error <= (Machine Epsilon or some problem specific relative input error) * condition Number <= $(5 * 10^-5)/(65) * 2.3 * 10^2$

Problem 2.

Floating Point Rounding What is the value of $2\sqrt{\times}$ when calculated in a decimal floating point system with 3 digits that uses rounding? (Apply rounding to both the input data and the result of the arithmetic.)

Solution 4.2. $\sqrt{2} \approx 1.41$ and $\pi \approx 3.14$. Multiply both and round to 3 decimal places.

Problem 3.

In a normalized floating point system that supports subnormal numbers, which of the following binary operations on two positive numbers could lead to overflow?

Solution 4.3. Addition, Multiplication and Division can all lead numbers to explode outside of the normalized floating number range.

If two addends are already normalized floating point numbers, however, then there is no chance that subtracting these numbers can lead to a result that is not a normalized floating point[1]

Problem 4.

1 point Machine Epsilon and UFL Which of the following statements are true?

Select all that apply: Machine epsilon places a lower bound on the magnitude of normal floating point numbers. Machine epsilon is determined by the number of bits in the significand. Machine epsilon places an upper bound on the relative error of representing a real number in a floating point format. The UFL is determined by the number of bits in the exponent.

Solution 4.4. The bottom 3 are correct; the first is wrong, because the magnitude of normal floating point numbers is determined by the exponent range; the second is correct, because machine epsilon is $\beta^{-(p-1)}$ where β is a base and p a precision; the third is correct, because given a floating point number y, the smallest perturbation to y that results in a different floating point number is $\epsilon|y|$. The fourth is correct, because the order of magnitude of UFL is β^{L+1} where the exponent range falls in $[L, U]$.

Problem 5.

Floating Point Relative Error Which of the following is closest to the relative error from representing $e \times 2 - 145$ in IEEE single precision.

Solution 4.5. Notice that $e \times 2^{-145}$ is subnormal number. The underflow limit for single precision IEEE floating point numbers is 2^{-126}. Anything smaller requires that the leading 1 be shifted along the fractional part of a number. Since $e \times 2^{-145} \approx 2^{-144}$, it follows that we need to shift the leading 1 18 units to the right of the decimal. This will leave 5 digits (bits) in the fractional component for futher reprsentation. If we have n digits, then our relative error is approximately β^{-n}. Whence, the relative error here is 2^{-5}.

Problem 6.

Properties of Floating Point Operations Which of the following is true for floating point multiplication and addition? Select all that apply: Floating point addition is associative. Floating point multiplication is commutative. Floating point multiplication is associative. Floating point addition is commutative.

Solution 4.6. Floating point multiplication and addition are commutative but not associative.

I skip problem 7 – if you understood the solution to an earlier problem, you'll be okay.

Example 4.7. To perform floating point addition, suppose that you have $b_0.b_1 \ldots b_n \times 2^x$ and $d_0.d_1 \ldots d_n \times 2^y$ where $x > y$; make sure that you multiply the second number by 2^{x-y} and then perform grade school addition.

[1]Recall that a normalized floating point number has 1 in the fractional part before the significand.

Definition 4.8. Catastrophic cancellation takes place when the number of accurate digits in the result of an operation are far less than the number of accurate digits in the members of that operation.

Example 4.9. Supposing that we opereate in double precision and that a and b both have 12 digits of accuracy. Suppose that a and b agree up to their first 10 digits.

When a and b are subtracted, the first 10 digits will be cancelled and the final 2 digits will shift leftwards, becoming the 2, sole accurate digits. The remaining digits will be filled with garbage. From this, we see that whereas we had 12 digits of accuracy, we now only have 2 (ie our relative error is $10^{-}2$).

Definition 4.10. Given a vector norm $\|\cdot\|$, we define the matrix norm to be

$$\sup_{x\in\mathcal{D},x\neq 0}\frac{\|Ax\|}{\|x\|}$$

This is equivalent to

$$\sup_{x\in\mathcal{D},x\neq 0}\frac{\|Ax\|}{\|x\|}$$
$$\sup_{x\in\mathcal{D},x\neq 0}\frac{\left\|A\frac{x}{\|x\|}\right\|}{1}$$
$$\sup_{x\in\mathcal{D},\|x\|=1}\frac{\|Ax\|}{1}$$

We denote this norm by $\|A\|$.

Proposition 4.11.

$$\|A\|\|x\| \geq \|Ax\|$$

for all x that A can multiply.

Proof. Follows by definition. □

Proposition 4.12.

$$\|A\|_1 = \max_j \sum_i |A_{i,j}|$$

$$\|A\|_\infty = \max_i \sum_j |A_{i,j}|$$

An easy way to remember the 1 norm and ∞ norm is to note that these matrix norms also coincide for a vector, for which the 1 norm is the sum of absolute values, and the ∞ norm is the maximum value.

Definition 4.13. Suppose that A is invertible and that we wish to find the condition number of the problem of solving $Ax = b$ where b is the input and x the ouput; then the condition number is given by

$$\frac{\|\Delta x\|}{\|x\|} \Big/ \frac{\|\Delta b\|}{\|b\|}$$

$$\frac{\|\Delta x\|}{\|x\|} \Big/ \frac{\|\Delta b\|}{\|b\|}$$

Let $x = A^{-1}b$ and $\Delta x = A^{-1}\Delta b$

$$\frac{\|A^{-1}\Delta b\|}{\|A^{-1}b\|} \Big/ \frac{\|\Delta b\|}{\|b\|}$$

Now apply submultiplicativity

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

This inequality can be shown to be sharp for every matrix: for every matrix, there is a vector b for which this bound is attained, and so the condition number of the problem is $\|A\|\|A^{-1}\|$

We also call this quantity the condition number of the matrix.

Definition 4.14. If a matrix is not invertible, then we say that $k(A) = \infty$.

Proposition 4.15. $k(A) = k(A^{-1})$ and matrix vector multiplication has the same conditioning as solving a system of linear equations.

Proof. The first is easy. To find the y such that $Ax = y$, take x as the input of $A^{-1}y = x$. □

Proposition 4.16.

- The following properties hold:
  - $\|AB\| \leq \|A\|\|B\|$
    - For $\|ABx\| \leq \|A\|\|Bx\| \leq \|A\|\|B\|\|x\|$. The previous was true for all x; in particular, for all $x \neq 0$, this implies that

$$\|AB\| = \frac{\|ABx\|}{\|x\|} \leq \|A\|\|B\|$$

  - $\|A + B\| \leq \|A\| + \|B\|$.
  - $\|AA^{-1}\| = 1 \leq \|A\|\|A^{-1}\| \implies 1 \leq k(A)$
- $k(\gamma A) = k(A)$.
- $\|\gamma A\| = \gamma\|A\|$.
- $k(A) = \max_{x\neq 0} \frac{\|Ax\|}{\|x\|} (\min_{x\neq 0} \frac{\|Ax\|}{\|x\|})^{-1}$

–

By definition:

$$\left\|A^{-1}\right\| = \max_{x \neq 0} \frac{\left\|A^{-1}x\right\|}{\|x\|}$$

Let $y = A^{-1}x$, meaning that $Ay = x$. Then

$$= \max_{x \neq 0} \frac{\|y\|}{\|Ay\|}$$

Since $y = A^{-1}y$, we can consider the max over y instead of x:

$$= \max_{y \neq 0} \frac{\|y\|}{\|Ay\|}$$
$$= (\min_{y \neq 0} \frac{\|Ay\|}{\|y\|})^{-1}$$

- If D is a diagonal matrix, then $k(D) = \frac{\max d_{i,i}}{\min_{d_{i,i}}}$
- An orthogonal matrix has norm 1 in any norm.

Remark 4.17. A computational technique to evaluating a condition number is multiply a matrix A by many vectors that have unit norm and take the maximum norm of the resultant vectors divided by the minimum norm of the resultant vectors.

5 Lectures 5 and 6

5.1 Quiz Lecture 5

Problem 1.

Floating Point Cancellation For which of the following reasons is cancellation in floating-point computation usually bad?

Choice* The digits lost are the least significant. The digits lost are the most significant. The result is usually not exactly representable. Subsequent operations are likely to underflow or overflow.

Solution 5.1. The digits lost during catastrophic cancellation are the most significant ones (ie the left most ones). Since we're losing digits in the resultant answer, our answer is often exactly representable (remember that we have about $-\log(2^{-52}) \approx 16 = -\log(10^{-16})$ digits of relative accuracy).

Problem 2.

Cancellation and Rounding Which of the following statements is true regarding the relationship between rounding and cancellation?

Choice* In computation that is done with exact (not rounded) values, cancellation cannot occur. Catastrophic cancellation occurs when cancellation amplifies rounding errors in the input. Subtracting two rounded numbers will always amplify the error. Cancellation from subtracting two rounded inputs of similar magnitude and sign can be reduced by first converting the rounded inputs to higher precision.

Solution 5.2. Even with exact values, we can find that leading digits are chopped off, if the two numbers agree to a high number of decimal places. Yes, catastrophic cancellation is the phenemonon that takes places when we subtract so much from a number that whatever is shifted upwards to replace the nullified digits is inaccurate, rounding error. This does not mean that subtracting any two numbers will always amplify rounding error, however. Even if you convert numbers to higher precision, this will not change the fact that cancellation will take place – several initial numbers will still match; it will reduce the chances of catastrophic cancellation taking place, however.

Problem 3.

Cancellation and Relative Error Suppose a and b are stored in single precision and agree to four decimal digits. Assume a is known to seven decimal digits and b is known to five decimal digits.

Let c=b−a.

Accounting only for cancellation error, how many decimal digits of accuracy are in c?

How many decimal digits are in c after accounting for the relative error in a and b?

Solution 5.3. In single precision, we have $-\log(2^{-22}) \approx 7$ digits of accuracy. Thus, a and b are (ignoring their relative accuracies initially) known really to 7 digits, so that if subtract them and nullify their first 4 digist, then there are only 3 digits that remain. If we account for relative error in the subtraction, however, then there are only $\min 7-4, 5-4 = 1$ digits that are accurate.

Problem 4.

Changing the RHS You just solved a linear system Ax=b. Unfortunately, the RHS b that you solved it with was wrong.

Worried, you compute ‖Δb‖‖b‖10−12. The condition number of your matrix is about 10000.

What could your worst-case relative error in the solution x be due to your use of the wrong RHS?

Solution 5.4. Just multiply condition number by relative input error.

Problem 5.

Distance to Singularity Which of the following is a good indicator that a matrix is nearly (as measured by the matrix norm) singular?

Choice* Its norm is small. Its determinant is small. Its condition number is large. Its norm is large.

Solution 5.5. $k(\gamma A) = k(A)$. That is, no scaling of a matrix can change its condition number; all the other quantities can change very much, however.

Problem 6.

What is the 2 norm of a diagonal matrix:

Solution 5.6. It is the maximum of teh diagonal entries. The condition number is the ratio of the maximum diagonal entry in absolute value to the minimum one in absolute value.

5.2 Quiz for Lecture 6

Relative Residual Consider a matrix $A = \begin{bmatrix} -9 & -5 \\ 8 & 3 \end{bmatrix}$ and right-hand side vector b=[3−2]. Using the infinity norm, calculate the relative residual if elements of the solution vector x^ are rounded to one significant digit. Include at least three significant digits in your answer.

Solution 5.7. $\|A\| = 14$. Let $\hat{x} = A^{-1}b = \begin{bmatrix} -0.08 \\ 0.5 \end{bmatrix}$ if you round to 1 digit. Then $r = A\hat{x} - b = \begin{bmatrix} -0.22 \\ 0.14 \end{bmatrix}$

Meaning that $\|r\| = 0.22$. And $\|x\| = 0.5$

$$\frac{\|r\|}{\|x\|\|A\|} = \frac{0.22}{0.5 * 15}$$

which is correct.

Problem 2.

A stupid definition question.

Problem 3.

Gaussian Elimination In the following questions, consider Gaussian elimination with the prescribed pivoting strategy to generate the lower (L) and upper (U) triangular factors for the following matrix,

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

Know that with pivoting, we must first swap rows 1 and 3 and then perform gaussian elimination.

Problem 4.

Existence of LU Decomposition with no Pivoting For which of the following matrices does a LU factorization without pivoting not exist?

Solution 5.8. I have

Choice*

- $\begin{bmatrix} 1 & 2 & 4 \\ 1 & 3 & 7 \\ 2 & 4 & 1 \end{bmatrix}$
- $\begin{bmatrix} 1 & 2 & 6 \\ 3 & 0 & 9 \\ 1 & 3 & 7 \end{bmatrix}$
- $\begin{bmatrix} 1 & 2 & 4 \\ 2 & 4 & 1 \\ 1 & 3 & 7 \end{bmatrix}$
- $\begin{bmatrix} 1 & 3 & 7 \\ 2 & 4 & 1 \\ 1 & 2 & 4 \end{bmatrix}$

Sufficient Condition: If a pivot is 0 (assuming that we don't pivot the matrix

when performing gaussian elimination), then the matrix will fail to provide an LU factorization.

Problem 5.

Just apply

$$\text{rel error} \leq k(A)\frac{\|r\|}{\|A\|\|x\|}$$

Problem 6.

Elimination Matrices

1 point

Consider two 10×10 elimination matrices M_4 and M_7.

- M_4 only has off-diagonal entries (below the diagonal) in column 4.
- M_7 only has off-diagonal entries (below the diagonal) in column 7.

Which of the following is true?

Choice*

- $M_4M_7 = M_4 + M_7 - I$ (where I is the identity matrix)
- $M_7M_4 = M_4 + M_7 - I$ (where I is the identity matrix)
- $M_4 = M_7$
- None of these

Note that elimination matrices must progress from left to right, meaning that elimination matrices M_1 and M_2 must be such that the off diagonal entry of M_1 is left of the off diagonal entry of M_2, if we want M_1M_2 to merge.

The following is the definition of the residual.

$$\boldsymbol{r = b - A\hat{x}.}$$

Remark 5.9. The residual itself does not reveal much. Suppose we calculate $r = b - Ax$. Now solve for $kAx = kb$ and the residual required to solve that is k times as great. This is why we define the relative residual:

$$\frac{\|r\|}{\|A\| \cdot \|\hat{x}\|}$$

We can obtain a bound on the relative forward error required to solve $Ax = b$ in terms of r.

$$\|\Delta \boldsymbol{x}\| = \|\hat{\boldsymbol{x}} - \boldsymbol{x}\| = \|\boldsymbol{A}^{-1}(\boldsymbol{A}\hat{\boldsymbol{x}} - \boldsymbol{b})\| = \| - \boldsymbol{A}^{-1}\boldsymbol{r}\| \leq \|\boldsymbol{A}^{-1}\| \cdot \|\boldsymbol{r}\|.$$

Dividing both sides by $\|\hat{\boldsymbol{x}}\|$ and using the definition of $\text{cond}(\boldsymbol{A})$, we then have

$$\frac{\|\Delta \boldsymbol{x}\|}{\|\hat{\boldsymbol{x}}\|} \leq \text{cond}(\boldsymbol{A}) \frac{\|\boldsymbol{r}\|}{\|\boldsymbol{A}\| \cdot \|\hat{\boldsymbol{x}}\|}.$$

Remark 5.10. This bound tells us that if the residual is small and the matrix and well conditioned, then the relative error is low.

Example 2.8 Small Residual. Consider the linear system

$$\boldsymbol{A}\boldsymbol{x} = \begin{bmatrix} 0.913 & 0.659 \\ 0.457 & 0.330 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0.254 \\ 0.127 \end{bmatrix} = \boldsymbol{b},$$

whose matrix we saw in Example 2.7. Consider two approximate solutions

$$\hat{\boldsymbol{x}}_1 = \begin{bmatrix} 0.6391 \\ -0.5 \end{bmatrix} \quad \text{and} \quad \hat{\boldsymbol{x}}_2 = \begin{bmatrix} 0.999 \\ -1.001 \end{bmatrix}.$$

The norms of their respective residuals are

$$\|\boldsymbol{r}_1\|_1 = 7.0 \times 10^{-5} \quad \text{and} \quad \|\boldsymbol{r}_2\|_1 = 2.4 \times 10^{-2}.$$

So which is the better solution? We are tempted to say $\hat{\boldsymbol{x}}_1$ because of its much smaller residual. But the exact solution to this system is $\boldsymbol{x} = [1, \ -1]^T$, as is easily confirmed, so $\hat{\boldsymbol{x}}_2$ is actually much more accurate than $\hat{\boldsymbol{x}}_1$. The reason for this surprising behavior is that the matrix $\boldsymbol{A}$ is ill-conditioned, as we saw in Example 2.7, and because of its large condition number, a small residual does not imply a small error in the solution. To see how $\hat{\boldsymbol{x}}_1$ was obtained, see Example 2.17.

Demo: Vanilla Gaussian Elimination

What do we get by doing Gaussian Elimination?

> *Row Echelon Form.*

How is that different from being upper triangular?

> Zeros allowed on and above the diagonal.

What if we do not just eliminate downward but also upward?

> That's called *Gauss-Jordan elimination*. Turns out to be computationally inefficient. We won't look at it.

Remark 5.11. Also note that a matrix is in row echelon form if the first non-zero entry of each row (what was the pivot during gaussian elimination) is to the right of the first non-zero entry of any preceding row; moreover, entries in rows above the pivot (but in the same column) must be 0.

Elimination Matrices

What does this matrix do?

$$\begin{pmatrix} 1 & & & & \\ & 1 & & & \\ -\frac{1}{2} & & 1 & & \\ & & & 1 & \\ & & & & 1 \end{pmatrix} \begin{pmatrix} * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \end{pmatrix}$$

- Add $(-1/2)\times$ the first row to the third row.
- One elementary step in Gaussian elimination
- Matrices like this are called *Elimination Matrices*

Remark 5.12. If we add k to the identity matrix at entry i, j, and left multiply the resultant matrix C by some matrix of interest A, then the result is to take the j th row of A multiply it by k and then add it to i. We can undo this process by using the same matrix but, in place of k, using $-k$. This second matrix is the inverse to the elimination matrix C.

Elimination Matrices

What does this matrix do?

$$\begin{pmatrix} 1 & & & & \\ & 1 & & & \\ -\frac{1}{2} & & 1 & & \\ & & & 1 & \\ & & & & 1 \end{pmatrix} \begin{pmatrix} * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \\ * & * & * & * & * \end{pmatrix}$$

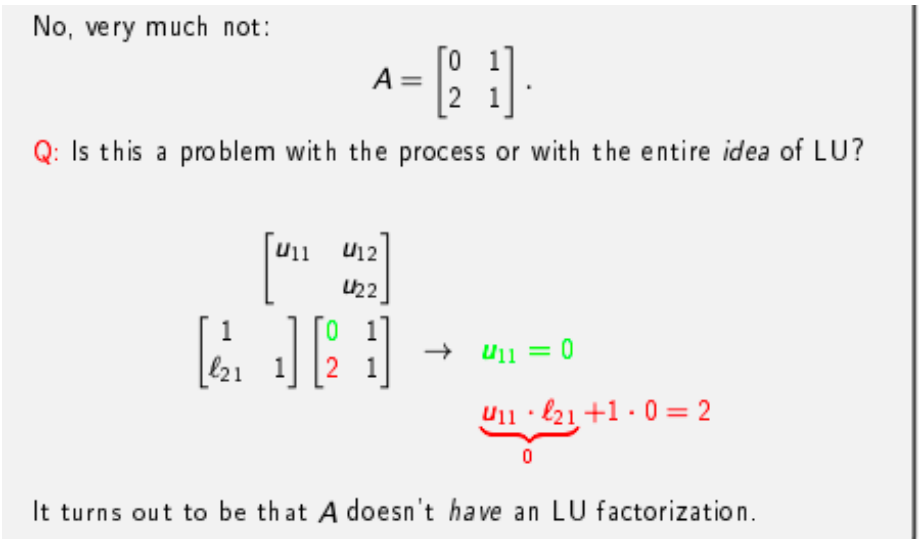
- Add $(-1/2)\times$ the first row to the third row.
- One elementary step in Gaussian elimination
- Matrices like this are called *Elimination Matrices*

Remark 5.13. Suppose that we multiply A by an elimination matrix M_1, then by M_2 up to M_l, where M_l is the last matrix required to turn A into Row Echelon Form. Eventually, we will have

$$(M_l \dots M_1)A = U \implies A = (M_l \dots M_1)^{-1}U$$

At first glance, this is okay, because it turns out that left multiplication of an elimination matrix X by Y such that X has a non-zero off diagonal at column i and Y has a non-zero off diagonal at column j where $i < j$ results in an elimination matrix that just merges X and Y [2]

For whatever reason, pivoting foils this attempt:



The solution is to repeatedly apply permutations to A (in the form of permutation matrices) so that the pivot is the largest element in terms of absolute value in its column.

Thus, we now have

$$(M_l P_l \dots M_1 P_1)A = U \implies A = (M_l P_l \dots M_1 P_1)^{-1} U$$

However, what should be L above is not always left triangular. It can be shown that a factorization of $(M_l P_l \dots M_1 P_1)^{-1}$ does, however, give us a lower triangular system.

[2] Note that merging also takes place if we multiply two elimination matrices that have their off diagonal non-zero entry in the same column as each other.

Sort out what LU with pivoting looks like. Have: $M_3 P_3 M_2 P_2 M_1 P_1 A = U$.

Define: $L_3 := M_3$
Define $L_2 := P_3 M_2 P_3^{-1}$
Define $L_1 := P_3 P_2 M_1 P_2^{-1} P_3^{-1}$

$$\begin{aligned}&(L_3 L_2 L_1)(P_3 P_2 P_1)\\ =&M_3 (P_3 M_2 P_3^{-1})(P_3 P_2 M_1 P_2^{-1} P_3^{-1}) P_3 P_2 P_1\\ =&M_3 P_3 M_2 P_2 M_1 P_1 \quad (!)\end{aligned}$$

$$\underbrace{P_3 P_2 P_1}_{P} A = \underbrace{L_1^{-1} L_2^{-1} L_3^{-1}}_{L} U.$$

$L_1, \ldots, L_3$ are still lower triangular!

Q: Outline the solve process with pivoted LU

Changing Condition Numbers

Once we have a matrix A in a linear system $A\mathbf{x} = \mathbf{b}$, are we stuck with its condition number? Or could we improve it?

Diagonal scaling is a simple strategy that sometimes helps.

- Row-wise: $DA\mathbf{x} = D\mathbf{b}$
- Column-wise: $AD\hat{\mathbf{x}} = \mathbf{b}$
  Different $\hat{\mathbf{x}}$: Recover $\mathbf{x} = D\hat{\mathbf{x}}$.

What is this called as a general concept?

Preconditioning

- Left preconditioning: $MA\mathbf{x} = M\mathbf{b}$
- Right preconditioning: $AM\hat{\mathbf{x}} = \mathbf{b}$
  Different $\hat{\mathbf{x}}$: Recover $\mathbf{x} = M\hat{\mathbf{x}}$.

Remark 5.14. Suppose that D above satisfies $k(D) \approx 1$. Then

$$k(DA) = \|DA\|\|(DA)^{-1}\| \leq \|D\|\|A\|\|A^{-1}\|\|D^{-1}\| \leq k(A)$$

so that the condition number of $K(DA)$ is no greater than the condition number of A.

Assuming that D is invertible, then the set of x satisfying $Ax = b$ is precisely the set of x satisfying $Ax = b$. Left multiplication by D of A is called, understandably, left preconditioning and scales A in a row-wise manner; right multiplication by D of A is called right preconditioning.

Remark 5.15.

Computational Cost

What is the computational cost of multiplying two $n \times n$ matrices?

$O(n^3)$

What is the computational cost of carrying out LU factorization on an $n \times n$ matrix?

Recall

$$M_3 P_3 M_2 P_2 M_1 P_1 A = U \ldots$$

so $O(n^4)$?!!!

Fortunately not: Multiplications with permuation matrices and elimination matrices only cost $O(n^2)$.

So overall cost of LU is just $O(n^3)$.

Demo: Complexity of Mat-Mat multiplication and LU

Multiplication by a permutation matrix is only an n operation, since it involves switching rows. Multiplication by an elimination matrix simply involves scaling one row and mulitplying it by another, and this process is done at most n times for any one elimination matrix (making it $O(n^2)$ as well). Since these transformations are applied at most n times, the process of getting a matrix into LU form is only $O(n^3)$.

Remark 5.16.

LU on Blocks: The Schur Complement

Given a matrix

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

can we do 'block LU' to get a *block triangular matrix*?

Multiply the top row by $-CA^{-1}$, add to second row, gives:

$$\begin{bmatrix} A & B \\ 0 & D - CA^{-1}B \end{bmatrix}.$$

$D - CA^{-1}B$ is called the Schur complement. Block pivoting is also possible if needed.

Not sure why this is significant.

Remark 5.17. Unresolved

Example 2.15 Small Pivots. Using finite-precision arithmetic, we must avoid not only zero pivots but also *small* pivots in order to prevent unacceptable error growth, as shown in the following example. Let

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

where ϵ is a positive number smaller than the unit roundoff ϵ_{mach} in a given floating-point system. If we do not interchange rows, then the pivot is ϵ and the resulting

multiplier is $-1/\epsilon$, so that we get the elimination matrix

$$\boldsymbol{M} = \begin{bmatrix} 1 & 0 \\ -1/\epsilon & 1 \end{bmatrix},$$

and hence

$$\boldsymbol{L} = \begin{bmatrix} 1 & 0 \\ 1/\epsilon & 1 \end{bmatrix} \quad \text{and} \quad \boldsymbol{U} = \begin{bmatrix} \epsilon & 1 \\ 0 & 1 - 1/\epsilon \end{bmatrix} = \begin{bmatrix} \epsilon & 1 \\ 0 & -1/\epsilon \end{bmatrix}$$

in floating-point arithmetic. But then

$$\boldsymbol{LU} = \begin{bmatrix} 1 & 0 \\ 1/\epsilon & 1 \end{bmatrix} \begin{bmatrix} \epsilon & 1 \\ 0 & -1/\epsilon \end{bmatrix} = \begin{bmatrix} \epsilon & 1 \\ 1 & 0 \end{bmatrix} \neq \boldsymbol{A}.$$

Using a small pivot, and a correspondingly large multiplier, has caused an unrecoverable loss of information in the transformed matrix. If we interchange rows, on the other hand, then the pivot is 1 and the resulting multiplier is $-\epsilon$, so that we get the elimination matrix

$$\boldsymbol{M} = \begin{bmatrix} 1 & 0 \\ -\epsilon & 1 \end{bmatrix},$$

and hence

$$\boldsymbol{L} = \begin{bmatrix} 1 & 0 \\ \epsilon & 1 \end{bmatrix} \quad \text{and} \quad \boldsymbol{U} = \begin{bmatrix} 1 & 1 \\ 0 & 1 - \epsilon \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

in floating-point arithmetic. We therefore have

$$\boldsymbol{LU} = \begin{bmatrix} 1 & 0 \\ \epsilon & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ \epsilon & 1 \end{bmatrix},$$

which is the

Remark 5.18. Notice that if we already have an LU factorization, then computing a rank 1 update is just an $O(n^2)$ operation.

Changing matrices

Seen: LU cheap to re-solve if RHS changes. (Able to keep the expensive bit, the LU factorization) What if the *matrix* changes?

Special cases allow something to be done (a so-called *rank-one update*):

$$\hat{A} = A + \mathbf{u}\mathbf{v}^T$$

The Sherman-Morrison formula gives us

$$(A + \mathbf{u}\mathbf{v}^T)^{-1} = A^{-1} - \frac{A^{-1}\mathbf{u}\mathbf{v}^T A^{-1}}{1 + \mathbf{v}^T A^{-1}\mathbf{u}}.$$

Proof: Multiply the above by $\hat{A}$ get the identity.
FYI: There is a rank-k analog called the Sherman-Morrison-Woodbury formula.

Demo: Sherman-Morrison

For

$$\left(A + uv^T\right)^{-1} b = A^{-1}b - \frac{\left(A^{-1}u\right) v^T A^{-1} b}{1 + v^T A^{-1} u}$$

And $A^{-1}x$ for any x is an $O(n^2)$ operation. The only other operation in this formula is to compute a dot product.

Remark 5.19.

LU: Special cases

What happens if we feed a non-invertible matrix to LU?

$$PA = LU$$

(invertible, not invertible) (Why?)

What happens if we feed LU an $m \times n$ non-square matrices?

Think carefully about sizes of factors and columns/rows that do/don't matter. Two cases:

- $m > n$ (tall&skinny): $L : m \times n$, $U : n \times n$
- $m < n$ (short&fat): $L : m \times m$, $U : m \times n$

This is called reduced LU factorization.

A matrix A always admits an LU factorization, even if A is singular. First, observe that every column of A must contain at least one non-zero number – or else, why would the column be part of A. Thus, if a pivot entry does not contain a non-zero value, we can rotate rows so that the pivot entry does have

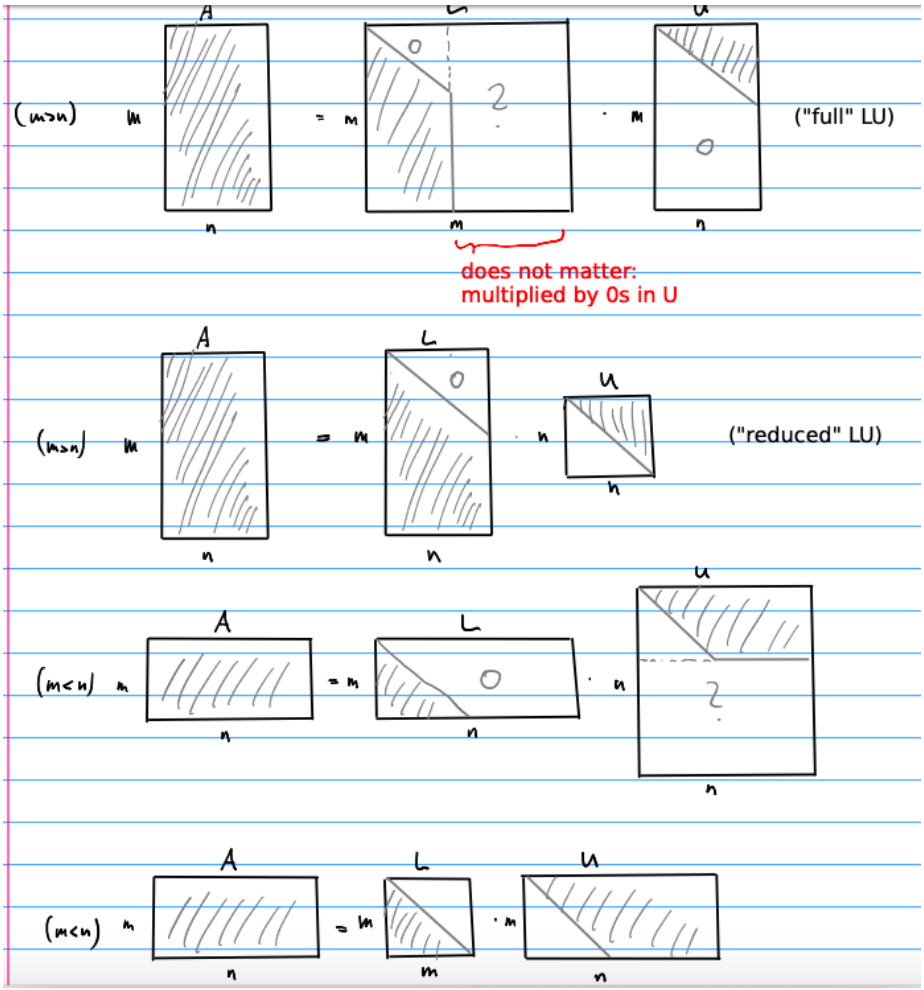
a non-zero value. We then can apply elimination matrices, as needed, until all row values in the pivot's column are 0.

The foregoing tells us that there still exists a sequence of permutations and elimination matrices that bring A into upper triangular form. Since these matrices are each invertible, it follows that L is still invertible. Since $|PA| = 0$, we must have, therefore, $|U| = 0$, which means that 0 must occupy some diagonal entry of U.

Why? A matrix fails to be invertible iff 0 is an eigenvalue. 0 is an eigenvalue iff some entry on the diagonal is 0, because the eigenvalues of a triangular matrix are precisely its diagonal entries.

Why are permutation matrices invertible? Group theory promises us that some power of a permutation brings it back to the identity. That is $P^k = I$ for some I. Thus $P^{-1} = P^{k-1}$.

Remark 5.20. Why can we take an LU decomposition and then reduce it, as shown below?



If $m > n$, applying elimination matrices that use row $n + 1$ is a no-op, since the use of row n has (along with prior use of rows $1 \ldots n - 1$) already made

row $n+1$ be entirely 0. or if $n > m$. As a consequence, in the unreduced LU factorization (the first and third pictures above), we see that every column in $\{n+1 \dots m\}$ is really just 0 except at a diagonal entry (where it is 1). As the picture points out, however, determinining what exists in the unreduced L is not useful, since $A = LU$ where $L = [QB]$ and $U = \begin{bmatrix} Q' \\ B' \end{bmatrix}$ where $Q = m \times n$, $B = m \times m - n$, $Q' = n \times n$ and $B' = m - n \times n$. Since $BB' = \mathbb{0}$, there is no need to store B or B'.

Using similar reasoning, we can understand the case that $n > m$.

6 Lecture 7

Least Squares

6.1 Quiz

Problem 1.

Gaussian Elimination with Partial Pivoting Under what conditions will Gaussian elimination with partial pivoting succeed in computing the LU factorization of an n×n matrix A?

Solution 6.1. Always.

Problem 2.

Basic Linear Algebra Subprograms Which of the following is true?

Select all that apply: Most BLAS level 3 functions can be composed out of multiple lower level functions BLAS level 1 functions do not involve matrices BLAS level 2 functions generally do more arithmetic operations per matrix/vector entry than level 3 functions BLAS level 2 functions do not involve vectors

Solution 6.2. Recall that level 1 is vector-vector operations; level 2 is matrix vector operations and level 3 is matrix matrix operations.

Problem 4.

Rank One Change If an n×n linear system Ax=b has already been solved by LU factorization, and then the matrix is changed by adding a matrix of rank 1, how much work is required to solve the new linear system with the same right-hand side?

Solution 6.3. The Sherman Morrison Formula tells us that this cost is $O(n^2)$.

Problem 5.

If we reduce a 9×24 matrix, then the shapes work out to be:

Solution 6.4.

$$(9 \times 9) \times (9 \times 24)$$

Problem 6.

Which problem requires the largest amount of work:
kkkkkk

Remark 6.5. We assume that we work with tall, skinny matrices that have full column rank.

Remark 6.6.

Properties of Least-Squares

Consider LSQ problem $\mathbf{Ax} \cong \mathbf{b}$ and its associated *objective function* $\varphi(x) = \|\mathbf{b} - \mathbf{Ax}\|_2^2$. Does this always have a solution?

Yes. $\varphi \geqslant 0$, $\varphi \to \infty$ as $\|\mathbf{x}\| \to \infty$, φ continuous $\Rightarrow$ has a minimum.

Is it always unique?

No, for example if A has a nullspace.

Examine the objective function, find its minimum.

$$\begin{aligned} \varphi(\mathbf{x}) &= (\mathbf{b} - A\mathbf{x})^T(\mathbf{b} - A\mathbf{x}) \\ &\quad \mathbf{b}^T\mathbf{b} - 2\mathbf{x}^T A^T\mathbf{b} + \mathbf{x}^T A^T A\mathbf{x} \\ \nabla\varphi(\mathbf{x}) &= -2A^T\mathbf{b} + 2A^T A\mathbf{x} \end{aligned}$$

$\nabla\varphi(\mathbf{x}) = \mathbf{0}$ yields $A^T A\mathbf{x} = A^T\mathbf{b}$. Called the *normal equations*.

The textbook proves that there is always a unique vector $y \in \text{Span}(A)$ such that $\phi(y) = \|b - y\|^2$ is minimal; as a consequence, there is at least one vector $x \in \mathbb{R}^m$ where A is $\mathbb{R}^{n\times m}$ that minimizes $Ax \approx b$. This vector x is unique iff A is full rank.

Definition 6.7. A matrix P is a projection if $P^2 = P$. A matrix is an orthogonal projection if $P^2 = P$ and $P^T = P$.

Proposition 6.8. If P is an orthogonal projection, then the span of $P_\perp = (I - P)$ is orthogonal to the span of P.

Proof. Given $x, y \in \mathbb{R}^n$, we see that

$$\begin{aligned} &\langle Px, (I - P)y\rangle \\ &= \langle Px, y - Py\rangle \\ &= \langle Px, y\rangle - \langle Px, Py\rangle \\ &= \langle Px, y\rangle - \langle x, Py\rangle \\ &= 0 \end{aligned}$$

□

Corollary 6.9. Given an orthogonal projection P, any vector x can be expressed as $x = Px + P_\perp x$.

Proposition 6.10. The vector x satisfying $\min_{x\in\mathbb{R}^n} \|Ax - b\|_2$ is precisely the x such that $Ax = Pb$ where P is a projection onto A.

Proof. Note that in what follows, all norms refer to the 2 norm.

$$\|Ax - b\| = \|P(Ax - b) + P_\perp(Ax - b)\|$$

Since P and $P_\perp$ map to orthogonal subspaces, we can apply the Pythagorean theorem

$$\begin{aligned} &= \|P(Ax - b)\| + \|P_\perp(Ax - b)\| \\ &= \|P(Ax - b)\| + \|-P_\perp b\| \\ &= \|P(Ax - b)\| + \|P_\perp b\| \\ &= \|(Ax - Pb)\| + \|P_\perp b\| \end{aligned}$$

The RHS is fixed, so we can only minimize the LHS

□

Corollary 6.11. The x aforementioned is $(A^TA)^{-1}A^Tb$

Proof.

$$\begin{aligned} Ax &= Pb \\ \iff A^TAx &= A^TPb \\ \iff A^TAx &= (PA)^Tb \\ \iff A^TAx &= (A)^Tb \\ \iff x &= (A^TA)^{-1}(A)^Tb \end{aligned}$$

□

Proposition 6.12. $P = (A^TA)^{-1}A^T$ is an orthogonal projection, assuming that A has full column rank.

Proof. Verify to yourself that it is symmetric and $P^2 = P$. Also verify that $\text{span}(P) = \text{span}(A)$. □

Corollary 6.13. The x aformentioned is orthogonal to $b - Ax$, the residual.

Proof.

$$b = Pb + P_\perp b$$

Substitute the definition of x and P found above

$$b = Ax + (b - Ax)$$

□

Proposition 6.14. Suppose we know that the columns of $Q \in \mathbb{R}^{m \times n}$ form an orthonormal basis for span(A). Then QQ^T is an orthogonal projector for A.

Proof. $(QQ^T)(QQ^T) = QQ^T$. Thus, this matrix is a projection; is it also clearly symmetric; finally, note that its span is precisely the span of A. □

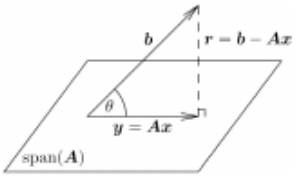
Corollary 6.15. With Q as above and $P = QQ^T$, then the optimal x satisfying $Ax \approx b$ is $Ax = Pb$. Leftmulitply both sides by Q^T to obtain

$$Q^T Ax = Q^T b$$

If we do this, we can avoid the hassle of using the normal equations.

Definition 6.16. I take the following as definitions. No time to look into their proofs:

Sensitivity and Conditioning of Least Squares



Define

$$\cos(\theta) = \frac{\|A\mathbf{x}\|_2}{\|\mathbf{b}\|_2},$$

then

$$\frac{\|\Delta\mathbf{x}\|_2}{\|\mathbf{x}\|_2} \leqslant \text{cond}(A)\frac{1}{\cos(\theta)} \cdot \frac{\|\Delta\mathbf{b}\|_2}{\|\mathbf{b}\|_2}.$$

What values of θ are bad?

$\mathbf{b} \perp \text{colspan}(A)$, i.e. $\theta \approx \pi/2$.

Sensitivity and Conditioning of Least Squares (II)

Any comments regarding dependencies?

Unlike for $A\mathbf{x} = \mathbf{b}$, the sensitivity of least squares solution depends on both A and $\mathbf{b}$.

What about changes in the matrix?

$$\frac{\|\Delta\mathbf{x}\|_2}{\|\mathbf{x}\|_2} \leqslant [\text{cond}(A)^2 \tan(\theta) + \text{cond}(A)] \cdot \frac{\|\Delta A\|_2}{\|A\|_2}.$$

Two behaviors:

- If $\tan(\theta) \approx 0$, condition number is $\text{cond}(A)$.
- Otherwise, $\text{cond}(A)^2$.

7 Lecture 8

Problem Transformations

7.1 Quiz

Remark 7.1. If Q is orthogonal, then $\|v\| = \|Qv\|$.