STAT 24310 Notes

Anthony Yoon

May 25, 2025

Abstract

Notes for STAT 24310 taught by Professor Yuehaw Khoo. Prior Linear Algebra and some basic multivariate calculus knowledge will be assumed. MATLAB notation for matrices will be extensively used.

Contents

1 Lecture 1: Introduction

This class is heavily based on Trefethen and Bau's Textbook. Take a look at it if you have the time.
When we run an algorithm, we often are interested in how long it takes to run the algorithm. For example, if we have a matrix $A \in \mathbb{R}^{n\times n}$ an $x, b \in \mathbb{R}^n$, and we are interested in solving

$$Ax = b$$

and n is very large, say $n = 100,000$, a concern is that the algorithm takes forever to run. In this case, we may be worried about the worst case time complexity, denoted as *big O notation*. In this case, solving $x = A^{-1}b$ is $\mathcal{O}(n^3)$. But what is this notation?

Definition 1.1. If there exists a $C \in \mathbb{R}$ such that for fg, where $g \geq 0$, where for all sufficiently t large enough, such that $|f(t)| \leq C \cdot g(t)$, then $f(t)\mathcal{O}(g(t))$

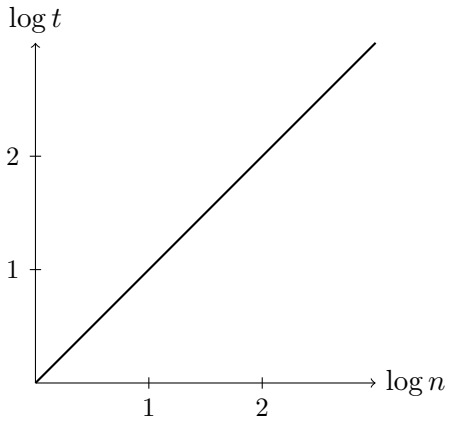
or equivantly:

Definition 1.2. If there exists a constant $C \in \mathbb{R}$ and a real number t_0 such that for all $t \geq t_0$, $|f(t)| \leq C \cdot g(t)$, where $g(t) \geq 0$, then we write $f(t) = \mathcal{O}(g(t))$ as $t \to \infty$.

Both are equivalent. For example, consider a problem that is $\mathcal{O}(n^3)$. If we were to increase the dimension of the problem, say a 100 times, then the algorithm would take 10000000 times longer to run. However, how do we visualize this? We do so by plotting the log-log plot, where we can note that:

$$t = \mathcal{O}(n^3)$$
$$\log t = 3 \log n + C$$

where C is some constant.



where we can see that the slope gives the exponent in time complexity.

1.1 Accuracy

When we are dealing with algorithms, we can also be concerned with how accurate it is. For example, consider $f : X \to Y$, where we are interested $f(x)$ for $x \in X$. However, pratically, we cannot calculate the exact values, take $\sqrt{x}$ for example, so a computer must make an approximation, usually denoted as $\tilde{f}(x)$. We can consider the relative accuracy, defined as follows:

Definition 1.3. Given an approximation of a function $\tilde{f}(x)$, we define relative accuracy as:

$$\frac{\|\tilde{f}(x) - f(x)\|}{\|f(x)\|}$$

where $\|\cdot\|$ is the norm of choosing.

A natural extension of accuracy are the follwing topics: **Stability and Conditional Number**. Intuively, stability is usually related to the algorithm ($\tilde{f}(x)$) used to solve the problem and the condition number is related to the actual setup of the problem ($f(x)$). We can consider the following defintions:

Definition 1.4. Backwards Stability: if $x \in X$. $\tilde{f}(x) = f(\tilde{x})$ for some $\frac{\|\tilde{x}-x\|}{\|x\|} = \mathcal{O}(\epsilon)$

Intuively, this above definition says that the problem will give the good enough answer to a slight deviation in the input. This definition is usually easier to prove.

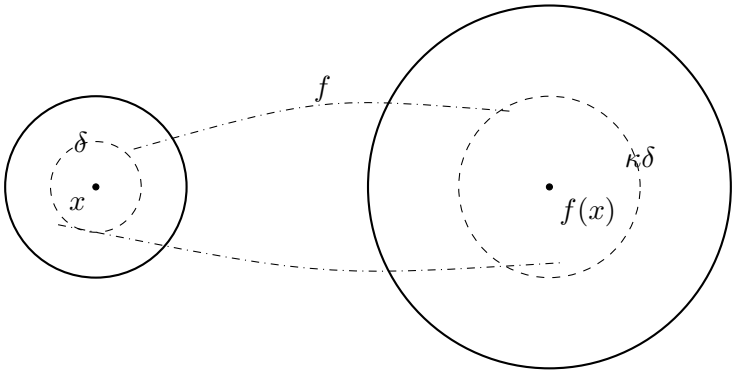
Definition 1.5. Absoulte Conditional Number: for (f, x)l we can consider the Absolute condition number as follows:

$$\hat{K}(f, x) = \lim_{\delta \to 0} \sup_{\|\delta x\| \leq \delta} \frac{\|f(x + \delta x) - f(x)\|}{\|\delta x\|}$$

Definition 1.6. Relative Conditional Number: for (f, x), we define the relative condition number as:

$$\lim_{\delta \to 0} \sup_{\frac{\|\delta x\|}{\|x\|} \leq \delta} \frac{\frac{\|f(x+\delta)-f(x)\|}{\|f(x)\|}}{\frac{\|\delta x\|}{\|x\|}} = \frac{\|f(x + \delta x) - f(x)\|\|x\|}{\|f(x)\|\|\delta x\|}$$

We can consider the diagram as follows:



We can see that the condition number measures how senstive the actual problem is to a change in x, as seen above. The larger the problem is, the more sensitive the problem is. Think of the diameter of the circles as how values go between each set. If the condition number is large, then the diameter is larger and thus, there is chance that the small pertrepuabation lands in a different area than original output.

Remark 1.1. Note if f is a function from $\mathbb{R}^n - \mathbb{R}^m$, we can define the following:

$$K_{abs} = J_f \quad K_{rel} = J_f \cdot \frac{\|x\|}{\|f(x)\|}$$

We can observe the following. If $\tilde{f}$ is backwards stable with accuracy ϵ, and if f has a relative condition number k. then $\tilde{f}$ is accurate with the follwoing relation:

$$\frac{\|\tilde{f}(x) - f(x)\|}{\|f(x)\|} = \mathcal{O}(k\epsilon)$$

Proof. If $\tilde{f}$ is backward stable, then there exists a point $y \in (x - \delta, x + \delta)$, with $\|\delta\| = \mathcal{O}(\epsilon\|x\|)$, such that:

$$\tilde{f}(x) = f(y)$$

This means the computed value $\tilde{f}(x)$ is the exact value of f evaluated at a slightly perturbed input y.
Now, using the definition of the relative condition number κ of f, we know that for small perturbations:

$$\frac{\|f(y) - f(x)\|}{\|f(x)\|} \leq \kappa \cdot \frac{\|y - x\|}{\|x\|} + \mathcal{O}\left(\left(\frac{\|y - x\|}{\|x\|}\right)^2\right)$$

Since $\|y - x\|/\|x\| = \mathcal{O}(\epsilon)$, it follows that:

$$\frac{\|\tilde{f}(x) - f(x)\|}{\|f(x)\|} = \frac{\|f(y) - f(x)\|}{\|f(x)\|} = \mathcal{O}(\kappa\epsilon)$$

□

2 Lecture 2: Norms

We begin with following definition of a norm in a general vector space.

Definition 2.1. Let X denote a vector space. A norm $\|\cdot\| : X \to R$ sastisfies

- Positivity: $\|x\| \geq 0$, $\|x\| = 0$ iff $x = 0$
- Homogenuity: $\|\alpha x\| = |alpha|\|x\|$
- Triangle inquality: $\|x + y\| \leq \|x\| + \|y\|$

We can also define the l_p norm

Definition 2.2. The l_p is defined as:

- $\|x\|_p = (\sum_{i=1}^{n} |x_i|^p)^{\frac{1}{p}}$, where $0 < p < \infty$
- $\|x\|_0$ refers to number of non-zero elements of X.
- $\|x\|_\infty = \max_j\{|x_j|\}$

We also have the following definitions:

Definition 2.3. Inner product on $\mathbb{C}^n$: $\langle x, y\rangle = \sum_{i=1}^{m} \overline{x}_i y_i$

Definition 2.4. Cauchy Schwartz inequality: $|\langle x, y\rangle| \leq \|x\|\|y\|$

Definition 2.5. Holder's inequality: $|\langle x, y\rangle \leq \|x\|_p\|y\|_q$ is $\frac{1}{p} + \frac{1}{q} = 1$

We can also define the equivalence of norms:

Definition 2.6. $\|\cdot\|_a, \|\cdot\|_b$ are equivalent if for $c_2 \geq c_1 \geq 0$:

$$C_1\|x\|_b \leq \|x\|_a \leq C_2\|x\|_b$$

We can prove that all norms are equivalent. This will be left as an exercise to the reader.

2.1 Matrix Norms

We can also introduce the notion of norms on matrices.

Definition 2.7. Frobenious norm: $\|A\|_F = \left(\sum_{i=1,j=1}^{m,n} |A_{ij}|^2\right)^{\frac{1}{2}}$

Definition 2.8. Induced norm: Given $A \in \mathbb{C}^{m\times n}, x \in \mathbb{C}^n$. We define the induced norm as follows:

$$\|A\|_{a\to b} = \sup_{x\neq 0} \frac{\|Ax\|_a}{\|x\|_b} = \sup_{\|x\|_b=1} \|Ax\|_a$$

Remark 2.1. By definition, we can see that:

$$\|Ax\|_a \leq \|A\|_{a\to b}\|x\|_b \quad \|A\|_a = \|A\|_{a\to a}$$

We also have the follwing properties:

Definition 2.9. We can also denote the following p norms.

- $\|A\|_1 = \max_{1\leq j\leq n} \|A(:,j)\|_1$, which is just the maximum column sum.
- $\|A\|_\infty = \max_{1\leq i\leq m} \|A(i,:)\|_1$ Which is the max of the row sum.

We also have the Cauchy Schwartz inequality for the matrix norm:

Definition 2.10. $\|AB\|_2 \leq \|A\|_2\|B\|_2$ and $\|AB\|_f \leq \|A\|_F\|B\|_f$ and $\|AB\|_f \leq \|A\|_2\|B\|_F$

We can prove the fact $\|AB\|_2 \leq \|A\|_2\|B\|_2$, which is as follows:

Proof. $\|AB\|_2 \leq \|A\|_2\|B\|_2$. To show this note,

$$\sup_{\|x\|=1} \|ABx\|_2 \leq \|Ay\|_2 \leq \|A\|_2\|y\|_2 \leq \|A\|_2$$

□

3 3: linear Transform

In numerical linear algebra, most problems and algorithms consist of linear maps.

Definition 3.1. Linear Transform is a function from $T : X \to Y$ where

- $T(\alpha x + \beta y) = \alpha T(x) + \beta T(y)$
- $T(v) = Av$ where $A \in \mathbb{C}^{m\times n}, v \in \mathbb{C}^n$

For example consider the following example.
Example
Let $P_k(x) = x^k$, where it is monomial, where $k \in \mathbb{N} \cup \{0\}$, where $X = span\{P_0, P_1, P_2\}$ and $Y = span\{P_0, P_1, P_2, P_3$. Note that if $q \in X$. this means that q is equivalent to a linear combination of the polynomials, where we can see that $Tq = (q+1)q$ It is trivial to show that T is a linear transform. Thus, we aim to find the matrix for this. Consider the following

$$Tp_1 = p_2 + p_1 \quad Tp_2 = p_3 + p_2$$

where we can construct the matrix as follows[1]

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

where columns vector is $\{p_0, p_1, p_2, p_3\}$ and row vectors are $\{p_0, p_1, p_2\}$.

3.1 Inner Products

We can introduce the inner product as follos.

Definition 3.2. Given a inner product $\langle \cdot, \cdot \rangle$ where $v \in \mathbb{C}^n$, we can see that

- It sastifies bilinearity
- $\langle v, w \rangle = \langle w, v \rangle$
- $\langle v, v \rangle \geq 0$ if and only if $v = 0$

Note that $\langle v, v \rangle = \|v\|^2$. We can also define $\langle v, v \rangle_M = v^T M V$ where M is a positive definite matrix.

Definition 3.3. A matrix is positive definite if and only if all the eigenvalues are positive.

Definition 3.4. A matrix is negative definite if and only if all the eigenvalues are negative.

Definition 3.5. Hermitian matrix is the complex analgue to the transpose of a matrix. Symmmetric matrix is $A^T = A$ and Hermitian matrix is $A = A^*$ [2]

3.2 Different kinds of multiplication

Refer to my STAT 24300 notes. Covers all the definition in Lectures 1 and 5. The only extension we have to be concerned about is the complexity of each operation. We can characterize it as follows:

- Inner product $\mathcal{O}(n)$, as we are multiplying ad summing n times
- Matrix vector product is $\mathcal{O}(mn)$ as we are doing the dot product over m rows.
- Matrix Matrix product is $\mathcal{O}(mnp)$, where $A \in \mathbb{C}^{m \times n}, B \in \mathbb{C}^{n \times p}$, as we are doing matrix vector product p times.

[1] Professor Khoo did some weird proof about the change of basis here, and frankly I don't think its worth putting here as an FYI

[2] The * superscript denotes the complex conjugate, where every entry of the matrix is conjugated.

4 4: Conditioning of the Linear Transform

Let us consider $T(x) = Ax$. To analyze behavior of the matrix, we decompose is $A = F_1F_2 \dots F_n$. We have the following key decompositions.

- For a square matrix, we have Eigenvalue Decomposition (EVD), where if we have a diagnolzaible matrix A where[3]

$$A = V\Lambda V^{-1} \quad \Lambda = diag(\lambda_1, \lambda_2, \dots, \lambda_n)$$

  where V have columns are linearly independent vectors. Let v_i be a column of the eigenbasis, then we see that $Av_i = \lambda_i v_i$ as

$$\det(A - \lambda I) = 0 \implies \quad \exists \quad v\text{s.t} \quad (A - \lambda I)v = 0 \iff Av = \lambda v$$

- SVD, where we have $A \in \mathbb{C}^{m\times n}$ where $A = U\Sigma V^*$, where $U \in \mathbb{C}^{m\times m}$ and $\Sigma \in \mathbb{C}^{m\times n}$ and $V \in \mathbb{C}^{n\times n}$ and where $\Sigma = diag(\sigma_1, \sigma_2, \dots, \sigma_n)$.

The SVD is something that we discuss heavily in this class. For the time being, assume that U and V is unitary, which means that $U^TU = V^TV = I_n$. This will be covered later. For both cases, the number of eigenvaues or singular values equals the rank of the matrix.

4.1 Spectral Radius

Given a matrix $A \in \mathbb{C}^{n\times n}$, we can define the spectral norm

Definition 4.1. We define the spectral norm as

$$\rho(A) := \max_{j\in\{1,2,\dots,m\}} |\lambda_j(A)|$$

Remark 4.1. Note that $\|A\|_p \geq \rho(A)$. As let v be an eigenvector of A,

$$\|Av\|_p = \|\lambda v\|_p \leq \|A\|_p\|v\|_p$$
$$|\lambda| \leq \|A\|_p$$

We can prove that $\|A\|_2^2 = \sigma_1$.

Proof. Let $A \in \mathbb{C}^{m\times n}$, and let the singular value decomposition (SVD) of A be:

$$A = U\Sigma V^*$$

where:

[3]Diag is shorthand for a diagnol matrix

- $U \in \mathbb{C}^{m\times m}$, $U^*U = I_m$ (unitary),
- $V \in \mathbb{C}^{n\times n}$, $V^*V = I_n$ (unitary),
- $\Sigma \in \mathbb{R}^{m\times n}$ is diagonal with nonnegative entries $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$, the singular values of A.

Recall that the operator 2-norm (spectral norm) of A is defined by:

$$\|A\|_2 := \sup_{\|x\|_2=1} \|Ax\|_2$$

Using the SVD $A = U\Sigma V^*$, and noting that U and V are unitary (thus preserve the 2-norm), we compute:

$$\|A\|_2 = \sup_{\|x\|_2=1} \|Ax\|_2 = \sup_{\|x\|_2=1} \|U\Sigma V^*x\|_2 = \sup_{\|x\|_2=1} \|\Sigma V^*x\|_2$$

Let $y = V^*x$. Since V is unitary, $\|y\|_2 = \|x\|_2 = 1$. So:

$$\|A\|_2 = \sup_{\|y\|_2=1} \|\Sigma y\|_2$$

Now, Σ is a diagonal matrix with singular values $\sigma_1, \ldots, \sigma_r$, so for any unit vector $y = (y_1, \ldots, y_n)$, we have:

$$\|\Sigma y\|_2^2 = \sum_{i=1}^{r} \sigma_i^2 |y_i|^2 \leq \sigma_1^2 \sum_{i=1}^{r} |y_i|^2 \leq \sigma_1^2$$

The supremum is achieved when $y = e_1$, so:

$$\|A\|_2^2 = \sup_{\|y\|_2=1} \|\Sigma y\|_2^2 = \sigma_1^2$$

□

Using a very similar logic, we can see that $\|A^{-1}\|_2 = \frac{1}{\sigma_n}$

5 Lecture 5: Unitary Matrix

We begin with the proof of the unitary matrix.

Proof. Consider the definition of the relative condition number, where we see that

$$K = \lim_{\delta\to 0} \sup_{\|\delta x\|\leq\delta} \frac{\|x\|}{\|Ax\|} \frac{\|A\delta x\|}{\|\delta x\|}$$

and we see that the above is equivalent to:

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

Note that if $\|\cdot\| = \|\cdot\|_2$, then we see that:

$$k_{rel} = \frac{\sigma_1}{\sigma_n}$$

□

now we can proceed to the properties of the Unitary matrix. First, we can introduce the properties of an orthogonal matrix:

Definition 5.1. Let $Q \in \mathbb{R}^{m\times n}$ with $m > n$, where q_i denotes the columns of Q. we see that $\forall q_i$, $\|q_i\| = 1$, $\langle q_1, q_j\rangle = \delta_{ij}$. A such matrix is called orthonormal.

Definition 5.2. A matrix that is orthonormal and square is unitary.

We can see the following properties of orthonormal and orthogonal matrices, where $Q^*Q = I_n$, $K(Q) = 1$ and $\|Qx\|_2 = \|x\|_2$

5.1 Basis interpretation of Q being orthogonal

Let $x = Qz$, where $z \in \mathbb{C}^n$. We assume that Q is an unitary matrix and $x \in Range(Q)$. Thus, we see that by the definition of matrix vector multiplication, we see that:

$$x = \sum_{i=1}^{n} z_i Q_i$$

However, note that:

$$\langle Q_i, x\rangle = \langle Q_i, \sum_{i=1}^{n} z_i Q_i\rangle = \sum_{i=1}^{n} z_i \langle Q_i, Q_j\rangle$$

Thus, using this idea, we can see that

$$\begin{aligned} x &= \sum_{i=1}^{n} z_i Q_i \\ &= \sum_{i=1}^{n} Q_i \langle Q_i, x\rangle \\ &= \sum_{i=1}^{n} Q_i Q_i^* x \end{aligned}$$

thus, we can see that QQ^* acts like an indentity matrix.

Definition 5.3. We can see that if we let Q to be orthonormal, we find that QQ^* is the orthogonal projector to the subspace $span(Q_1, Q_2, \ldots, Q_n) = range(Q)$ if $Q \notin range(Q)$.

Remark 5.1. We can see that $(I - QQ^*)v$ represents the perpendicular aspect of the vector, or rather the part that is not projected into the subspace.

6 Lecture 6: QR decomposition

Refer to this article, does a very good of explaning it. The intution is we keep removing all perpednicualr parts of each vector in the basis to ensure we have an orthonormla basis. We have the following algorthim to solve it.

Algorithm 1 Gram-Schmidt Orthogonalization (Matrix Form)

Require: Linearly independent vectors $\{a_1, a_2, \ldots, a_n\} \subseteq \mathbb{C}^m$

Ensure: Orthonormal matrix $Q = [q_1 \ q_2 \ \ldots \ q_n] \in \mathbb{C}^{m \times n}$ such that $\text{span}(q_1, \ldots, q_k) = \text{span}(a_1, \ldots, a_k)$ for all k

1: $q_1 \leftarrow \dfrac{a_1}{\|a_1\|}$
2: **for** $j = 2$ to n **do**
3: Let $Q_{1:j-1} \leftarrow [q_1 \ q_2 \ \ldots \ q_{j-1}]$
4: $v_j \leftarrow \left(I - Q_{1:j-1}Q_{1:j-1}^*\right) a_j$
5: $q_j \leftarrow \dfrac{v_j}{\|v_j\|}$
6: **end for**
7: **return** $Q = [q_1 \ q_2 \ \ldots \ q_n]$

we can also find an upper triangular matrix of A, whose columns are $\{a_1, a_2, \ldots a_n\}$, where we can consider the following algorithm[4]

[4]Not explicity explained by Professor Khoo, but nice to know

Algorithm 2 QR Decomposition via Gram-Schmidt (Matrix Form)

Require: Linearly independent vectors $\{a_1, a_2, \ldots, a_n\} \subseteq \mathbb{C}^m$

Ensure: $Q \in \mathbb{C}^{m\times n}$ with orthonormal columns, and upper triangular $R \in \mathbb{C}^{n\times n}$ such that $A = QR$

Initialize $Q \leftarrow 0_{m\times n}$, $R \leftarrow 0_{n\times n}$
for $j = 1$ to n **do**
    $v_j \leftarrow a_j$
    for $i = 1$ to $j-1$ **do**
        $R_{i,j} \leftarrow \langle q_i, a_j \rangle$
        $v_j \leftarrow v_j - R_{i,j} q_i$
    end for
    $R_{j,j} \leftarrow \|v_j\|$
    $q_j \leftarrow \dfrac{v_j}{R_{j,j}}$
    Set column j of Q to q_j
end for
return (Q, R) such that $A = QR$

The complexity of calculating the norm is $\mathcal{O}(m)$. Note the matrix multiplciation of $Q_{1:j-1}Q^*_{1:j-1}$ is $\mathcal{O}(m(j-1))$. Thus, performing we can see that:

$$\mathcal{O}(m) + \mathcal{O}(m(j-1)) + \mathcal{O}(m(j-1)) = \mathcal{O}(m + 2m(j-1)) = \mathcal{O}(m(j-1)) = \mathcal{O}(mn)$$

note we do the operation n times via the for loop, and we see that the final compleixty is $\mathcal{O}\left(mn^2\right)$. However, note the division sign. This means that algorithm is unstable as values could exponentially increase geivn a small enough norm magnitude. From here, we can introduce a more stable procedure, the Householder Reflection.

6.1 HouseHolder Relfection

Refer to the visualzaition for intuition of the Householder transformation.

$$\underbrace{\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots \\ a_{21} & a_{22} & a_{23} & \cdots \\ a_{31} & a_{32} & a_{33} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}}_{A} \xrightarrow{H_1} \underbrace{\begin{bmatrix} * & * & * & \cdots \\ 0 & * & * & \cdots \\ 0 & * & * & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}}_{H_1 A} \quad \text{Zeroed by } H_1$$

With the householder reflection decomposition, we aim to reduce problem of the QR de-

composition to

$$A^{(1)} = Q^{(1)}A^{(0)} \quad A^{(2)} = Q^{(2)}A^{(1)}$$
$$Q^{(n)}Q^{(n-1)} \dots Q^{(1)}A = R$$

where if $Q^{(i)}$ is an unitary matrix, that we are left with a QR factorization.
We want our string of unitary matrices to be in the following form:

$$Q^{(1)} = \begin{bmatrix} I_{j-1} & 0 \\ 0 & F^{(1)} \end{bmatrix}$$

where F is $(m-j+1) \times (m-j+1)$. We want this F to act on the lower half of A to ensure that we elimnate the entries below to ensure that we have the RREF like structure of the above diagram. Or more specfically,

$$A^{(j-1)} = \begin{bmatrix} U^{(j-1)} \\ L^{(j-1)} \end{bmatrix}$$

where $U^{(j-1)}$ is $(j-1) \times (j-1)$ and $L^{(j-1)}$ is $(m-j+1) \times (m-j+1)$. Thus, we can see that:

$$A^{(j-1)} = Q^{(j)}A^{(j-1)} = \begin{bmatrix} U^{(j-1)} \\ F^{(j)}L^{(j-1)} \end{bmatrix}$$

We also want to make F an orthonomral matrix to enforce that the bigger Q. However, to reinforce the RREF like nature, we have the following subproblem.

We want to solve the problem of $Fx = z\|x\|e_1$

where $|z| = 1$. Let $v = \|x\|e - x$, which we call the householder vector. The geonetric intuition of this is that v points in the direction of $\|x\|e$ to from x, and we can reflect the point accross the line to get the vector mapped onto $\|x\|e$, which is equivalent to zeroing out entries.

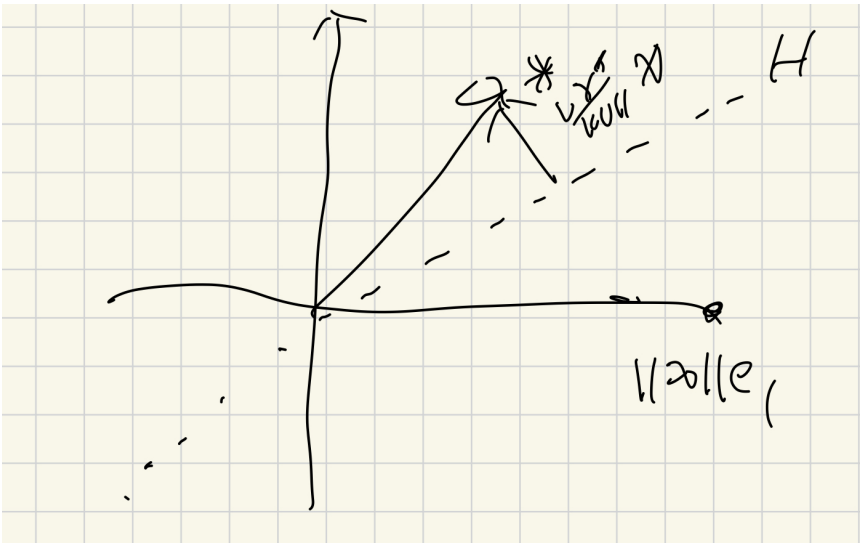


Figure 1: Intution of the householder reflection

However, we are interested in the quantity $v = \|x\|e - x$. Thus, we find the reflection matrix as follows:

$$I - \frac{2vv^*}{\|v\|^2}$$

where the multiplication of two is derived from the fact that we are interested in a vector pointing in the opposite direction original vector. Now, a natural question that may arise from this is *what is the optimal choice of z?*. To ensure numeric stabilitty, we choose $z \in \{-1, 1\}$ such that z has the opposite sign of x. The pseudocode for this decomposition is as follows:

Algorithm 3 Householder Relfection QR decomposition

Require: $A \in \mathbb{C}^{m \times n}$
Ensure: $Q_1 Q_2 \dots Q_n A = R$
Let $A^0 = A$
Let $Q_0 = I_m$
Intialize
for j in $1 : min(m, n)$ **do**
    $x = A^{j-1}_{(j:m,j)}$
    Let z be opposite sign of x
    Let $\tilde{v}^j = \frac{v^j}{\|v\|^j}$
    $\gamma = I - 2\tilde{v}^j(\tilde{v}^j)^T)A^{j-1}_{j:m,j:m}$
    Let $H = I_{m \times m}$
    Let $H_{j:m,j:m} = \gamma$
    $A^j = HA^{j-1}$
    $Q^j = Q^{j-1}H^T$
end for
return A, Q where $A = R$ in the QR decomposition.

The complexity of this algorithm is $\mathcal{O}(mn^2)$ as we have matrix multiplicatio and a for loop.

7 Lecture 7: Linear System

When we want to solve a system of general linear equations, we can generalize out the the definition of the Pseudoinverse.

Definition 7.1. The Moore Rose Pseudoinerse is $A = V\Sigma^T U^T$ where $A = U\Sigma V^T$, or the SVD decomposition. If A is invertible, than $A^{-1} = A^\dagger$

Remark 7.1. It can be shown that $A^\dagger$ can be used to solve the Least Squares Problem. This will be shown later in the future

7.1 Naive solution to solving system of linear equations

We could find a solution by analyzing solving for A^{-1} or $A^\dagger$, but this solution is very computationally expensive. Thus, we can proceed with a more optimized method: Gaussian Elimination.

7.2 Gaussian Elimination

Consider the follwing analogy. We want to solve for

$$\left[\begin{array}{cc|c} 2 & 1 & 5 \\ 4 & -6 & -2 \end{array}\right] \xrightarrow{R_2 \leftarrow R_2 - 2R_1} \left[\begin{array}{cc|c} 2 & 1 & 5 \\ 0 & -8 & -12 \end{array}\right]$$

$$\left[\begin{array}{cc|c} 2 & 1 & 5 \\ 0 & -8 & -12 \end{array}\right] \xrightarrow[R_2 \leftarrow -\frac{1}{8}R_2]{R_1 \leftarrow \frac{1}{2}R_1} \left[\begin{array}{cc|c} 1 & \frac{1}{2} & \frac{5}{2} \\ 0 & 1 & \frac{3}{2} \end{array}\right]$$

$$\left[\begin{array}{cc|c} 1 & \frac{1}{2} & \frac{5}{2} \\ 0 & 1 & \frac{3}{2} \end{array}\right] \xrightarrow{R_1 \leftarrow R_1 - \frac{1}{2}R_2} \left[\begin{array}{cc|c} 1 & 0 & 1 \\ 0 & 1 & \frac{3}{2} \end{array}\right]$$

We aim to replicate the above by creating a triangular matrix and utilizing back substitution to solve for the other variables. The whole idea is we eliminate entries from the augmented matrix and then utilize back substitution to solve for all variables. We do so by using each pivot point. The following is algorthim for this.

Algorithm 4 Gaussian Elimination using Back Substitution with no Pivoting

Require: $A \in \mathbb{R}^{n \times n}$, $y \in \mathbb{R}^n$
Ensure: x such that $x = A^{-1}y$

Let $U \leftarrow A$, $L \leftarrow I_n$ ▷ LU decomposition
for $j = 1$ to $n - 1$ **do**
  for $i = j + 1$ to n **do**
    if $U_{j,j} = 0$ **then**
      stop the program ▷ Zero pivot element
    end if
    $L_{i,j} \leftarrow \frac{U_{i,j}}{U_{j,j}}$
    $U_{i,j:n} \leftarrow U_{i,j:n} - L_{i,j} \cdot U_{j,j:n}$
  end for
end for
Initialize $z \in \mathbb{R}^n$ ▷ Forward substitution to solve $Lz = y$
for $i = 1$ to n **do**
  $z_i \leftarrow y_i - \sum_{j=1}^{i-1} L_{i,j} z_j$
end for
Initialize $x \in \mathbb{R}^n$ ▷ Back substitution to solve $Ux = z$
for $i = n$ to 1 with step size -1 **do**
  $x_i \leftarrow \left(z_i - \sum_{j=i+1}^{n} U_{i,j} x_j\right) / U_{i,i}$
end for

Remark 7.2. Here, we comment on the structure on what the L matrix looks like during the Gaussian Elimination. We can see hat the inverse of L is also lower triangular.

Do later. Just know that it replcaites the operations done to reduce a matrix

7.3 Characteristics of the LU decomposition

We finally get that $A = LU$, where we could use this find solutions to linear systems. We can also find the complexity of each step of the algorithm, where we see that in the Gaussian Elmination step, we can take some assumptions about the structure of the matrix to find that the complexity of this step.
Assume each matrix is apporximatly $n \times n$. We see that we utilize matrix multiplication, which yields $\mathcal{O}\left(n^2\right)$ per j row. However, we do j n times, and we can see that the total cost would be $\mathcal{O}\left(n^3\right)$. We can find a similar relation with the backwards substitution, which would be $\mathcal{O}\left(n^2\right)$ as we can see that we also do multiplication n times.

8 Lecture 8: LSQ

Consider the following problem:

$$Ax = b$$

where $A \in \mathbb{R}^{n \times n}$. As noted previously, we could use the LU decomposition to solve this as we can

1. Solve $A = LU$
2. Set up the equation as $LUx = b$
3. Let $Ux = y$ and solve $Ly = b$ for y
4. Then solve $Lx = y$ where x is the solution to the system.

Note the above only holds when the $A_{1:k,1:k}$ is always nonsingular. Now we ask the question where we change $A \in \mathbb{R}^{n \times n}$ to $A \in \mathbb{R}^{m \times n}$. Now we have an issue where we may have more measurement than what we need, or an *overdetermined system*. This means that there is a case where there may not exist an x within the range of A that causes this happen. This means that we have to find a line of best fit that may minimize error, which introduces the following problem:

8.1 Least Squares

We aim to

$$\min \|Ax - b_2^2$$

Remark 8.1. The above problem will always have a solution, but not always unique. However, if A is full rank, than the above problem will have an unique solution.

We can prove the second part of the remark as follows:

Proof. We begin with the contrapositive proof of the statement. So we aim to prove that if A has less than rank n then there exists no solution. Assume that $rank(A) \neq n$, which means that $ker(A) \neq 0$, which implies that there exists a z such that $Az = 0$ Assume that x^* is the solution to the LS squares. Then we know that $\|Ax^* - b\|$ is the minimum. Therefore, we can see that:

$$\begin{aligned} \|Ax^* - b\| &= \|Ax^* - Az - b\| \\ &= \|A(x^* + z) - b\| \end{aligned}$$

we see that there exists more than one minimizing argumenet. □

Remark 8.2. Note that the least squares approximation is one of the most widely done things

- Even with a very noisy b, we can make an approximation
- Calculating the exact inverse is also expensive, but we can approximate it but pay some price in approximation.

With that we begin with an analysis of the problem. Let $x^* = \arg\min \|Ax - b\|$. If $x^* \in range(A)$, then we know that there exists some solution such that

$$\|Ax-b\|^2 = \|p(Ax-b)^2\|+\|(I-P)(Ax-b)\|^2 = \|p(Ax-b)\|^2+\|(I-p)b\|^2 = \|Ax-pb\|^2+C$$

where p is the projection matrix onto the vector A and C is a constant. Thus, to solve the least squares problem is equivalent to$range(A) \perp Ax - b$ which is equivalent $Ax - b \in null(A)$ This implies that

$$A^*(Ax - b) = A^*Ax = A^*b$$

Note that if A has full rank, this means that A^*A has full rank. This implies that

$$x^* = (A^*A)^{-1}A^*b$$

We can define $(A^*A)^{-1}A^*$ as the Moorse Penrose Pseudo-inverse. We can calculate this in one of two ways

8.2 Calculating the Pseudo-inverse via the SVD

We define that $A = U\Sigma V^*$ where we can see that

$$A^*A = V^*\Sigma^*\Sigma V$$

9 Lecture 9: Introduction to Optimization

Last class, we showed that there existed a solution to the Least Squares problem, or rather $\min \|Ax - b\|_2^2$ through a combination of the QR and LU decomposition. We can consider the following general optimization problem, where we define $f : \Omega \to \mathbb{R}$:

$$x^* = \arg\min_{x \in \Omega} f(x)$$

where we say that Ω is the objective domain. The intuition is that we want to find the input that would minimize the value of the function.

9.1 Classes of Optimization Problems

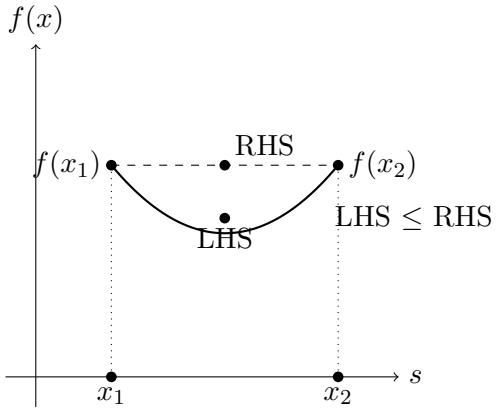
Given a domain Ω that we want to optimize over, we have the following scenarios:

- a set of discrete points, which is discrete optimization
- $\Omega \subseteq \mathbb{R}^n, \subseteq \mathbb{R}^n$, a constrained optimization problem. An example of this would be induced norms.
- $\Omega = \mathbb{R}^n$, a unconstrained optimization problem. An example of this would be the Least Squares Regression Problem.

We can often times utilize properties of the function, which is where we can introduce the idea of convexity.

Definition 9.1. f is convex over S if for any $x_1, x_2 \in S$ and $t \in [0, 1]$, $f(tx_1 + (1-t)x_2) \leq tf(x_1) + (1-t)f(x_2)$

This concept can be illustrated in the following diagram:



We can then expand the definition of convexity as follows:

Definition 9.2. f is strongly convex over S if for any $x_1, x_2 \in S$ and $t \in [0,1]$, $f(tx_1 + (1-t)x_2) < tf(x_1) + (1-t)f(x_2)$

A natural expansion of this is when f is convex, then $f\left(\sum_{i=1}^n t_i x_i\right) \leq \sum_{i=1}^n t_i f(x_i)$ where $\sum_{i=1}^n t_i = 1$ such that $t_i \geq 0$. We can also consider the following definition of convexity:

Definition 9.3. Locally Convex: $f : \Omega \to \mathbb{R}$, where f is convex over some $S \subseteq \Omega$

9.2 Taylor's Theorem

We begin with an introduction of smoothness.

Definition 9.4. A function is said to be C^n if it is contionus and differentiable n number of times.

Also a refresher on what the Taylor's Theorem in 1D is.

Definition 9.5. if $f : \mathbb{R} \to \mathbb{R}$ is $(k+1)$ times differentiable at $x = a$, then

$$f(x) = f(a) + \frac{\partial f(a)}{\partial x}(x-a) + \cdots + \frac{1}{k!}\frac{\partial^k f(x)}{\partial x^k}(x-a)^k + \frac{L}{(k+1)!}(x-a)^{k+1}$$

where

$$L = \sup_{\xi \in [x,a]} \left| \frac{\partial^{k+1} f(x)}{\partial x^k} \right|$$

note that if $|x-a|$ is sufficiently small, we can discard the remainder the term (the one with the L in it). Note that also that if the $(k+1)$ derivative is bounded, then we can approximate f with the kth order polynomial. We can also consider the case of the special case of the third order taylor polynomial expansion.

Definition 9.6. For $f : \mathbb{R}^n \to \mathbb{R}$, and f has a bounded 3rd deriative, the expansion is as follows

$$f(x) = f(a) + \frac{\partial f(a)}{\partial x}(x-a) + \frac{1}{2}(x-a^T)H(a)(x-a) + \mathcal{O}(\|x-a\|^3)$$

where we define

$$H(a)_{ij} = \frac{\partial f(a)}{\partial x_i \partial x_j}$$

where $H(a)$ is $n \times n$

To illustrate the nature of the Hessian Matrix, we can visualize f locally to see that we can use a contour map. The Hessian usually shows the directions of the biggest changes in direction, given by the Eigenvalues. Need to fix
However, we can see that if our function is convex and smooth, then *optimality is guarantted based on the local deriative.* We can see this in the following proof:

Proof. if f is convex and differentiable, we can see that if we let a, b be in the domain of f. we can see that we are left with the following inequality.

$$f(b) \geq f(a) + \frac{\partial f(a)}{\partial x}(b - a)$$

by the definition of convexity. If we choose a such that $f'(a) = 0$, we see that we have guarnteed an optimal solution. □

However, we can see that if we are working with a strongly convex differentiable solution, we can use the same assumptions as above to see in the following proof.

Proof. We are given that f is strongly convex and differentiable. Thus, we can see that for the following taylor expansions

$$f(b) \geq f(a) + \frac{\partial f(a)}{\partial x}(b - a) + \frac{a}{2}\|b - a\|_2^2$$

We see that we can choose a point such that $\frac{\partial f(a)}{\partial x} = 0$, which implies that

$$\begin{aligned} f(b) &\geq f(a) + \frac{a}{2}\|b - a\|^2 \\ \implies f(b) &> f(a), \forall b, b \neq a \end{aligned}$$

Since we know that the function is strongly convex and thus, we know that for any point b and a. the last inequality holds, we can see that we have indeed foound the unique optimizer. □

Remark 9.1. Strongly convex implies strictly convex, but not the other way around.

9.3 Applications of convexity

Consider the following:

$$f(x) = \frac{1}{2}ax^2 - bx, a > 0$$

and consider the following:

$$\begin{aligned} f(x) &= \frac{1}{2}ax^2 - bx \\ f(y) - f(x) &= \frac{1}{2}a(y^2 - x^2) - b(y - x) \end{aligned}$$

10 Lecture 10: Introduction to Gradient Descent

As noted before, we can see that when we are given a convex function, say $f(x) = 0.5ax^2 - bx$ where $a > 0, b > 0$, we are left with a convex function where we can use the deriative of the function to solve for the extrema values. Now we aim to expand this out to a multivariate case. The anaolgue of a positive value matrix is a *positive definite matrix*, or where all the eigenvalues are all positive. We can show that if M is a positive definite symmetric matrix, than f is a strongly convex function.

Proof. We are given the function $f(x) = \frac{1}{2}M - b^T x$, which is a natural analogue to the 1-d case. We can see that $M = U\Lambda U^T$, where U is orthogonal. Thus, we can see that

$$f(x) = \frac{1}{2}x^T U\Lambda U^T x - b^T(UU^T)x$$

and let $\tilde{x} = U^T x$ and $\tilde{b} = U^T b$. Thus, we are left with the following:

$$\sum_{i=1}^{n} \frac{1}{2}\lambda_{ii}\tilde{x}_i^2 - \tilde{b}_i\tilde{x}_i$$

which is equiavlent to the 1d case, where

$$\sum_{i=1}^{n} f_i(\tilde{x}_i)$$

where each f_i is strongly convex. Since the sum of strongly convex functions are still strongly convex, then we know that f is strongly convex. □

Remark 10.1. We are familar with the least squares problem, or when

$$x = \arg\min_x \|Ax - b\|^2$$

Fill in this in later after confirming

10.1 How to solve for the extrema

If we are given a strongly convex function where there exists a x such that $\nabla f(x) = 0$. then x must be the unique minimizer. On a computer, we have the following methods:

- **Netwon's Type Method**: Note that we are interested in solving the system of equations where
$$\nabla f(x) = Mx - b$$
where M denotes the Jacobian of a matrix. To compute this problem, we see that we have to solve for $x^* = M^{-1}b$, which is compuationally expensive, as proved before it takes $\mathcal{O}(n^3)$ to solve such a system.

- **Gradient based minimization thorugh iterative algorithm**: We use the gradient multiplized by some arbitary value to travel down the fucntion to see if we can converge on the minima value. For example, we want a solution that converges to the optimal value, say x^*.

Ultimately for the gradient method, we want to create a sequence of x that converegs to the optimal value x^*. To do so, we must define some sort of relationship between successive values in the sequence. Therefore, we must introduce an *updating scheme*. We define the updating scheme as follows:

$$x^{k+1} = x^k - s_k Df(x^k)$$

where s_k is the step size. Now a natural question that may follow is "How do we choose the step size?".
This depends heavily on the shape and structure of the function. Consider the strongly convex framework that we are very familar with. We can define an objective function, where we want to minimize the difference in function values of the sequence values and the actual minimizer. Consider the following:

$$f(x) = \frac{1}{2}(x - x^k)^T < (x - x^k) + \xi$$

where ξ is some constant. We can see that the above simplifes to:

$$f(x) = \frac{1}{2}\|x - x^k\|_M^2 + \xi$$

without a loss of generality, we can set the above function as $f(x) = \frac{1}{2}\|x - x^k\|_M^2$. If we want to show that the fucntion linear converges to some magnitude, we can see that:

$$\|x^{k+1} - x^*\| \leq C\|x^k - x^*\|^q$$

We have two cases, linear convergence and quadratic convergence. Linear convergence is when $q = 1$ and $C \in [0, 1]$. Similarly, quadratic convergence is when $q = 2$ and $C > 0$. We begin to prove that this gradient descent has linear convergence is we pick a step size such that $s_k \in (0, \frac{1}{\lambda_{max}(M)})$

Proof. For notational sake, let M be the Jacobian matrix and. Note that:

$$\nabla f(x) = Mx - b$$

Thus,

$$\begin{aligned}
\|x^{k+1} - x^*\|^2 &= \|x^k - s(Mx^k - b) - x^*\|^2 \\
&= \|x^k - x^* - s(Mx^k - Mx^*)\|^2 \\
&= \|(I - sM)(x^k - x^*)\|^2
\end{aligned}$$

Note that:

$$\|(I - sM)(x^k - x^*)\|^2 \leq \|I - sM\|^2 \|(x^k - x^*)\|^2$$

By the property of the spectral norm, we can bound the above equation as:

$$\|(I - sM)(x^k - x^*)\|^2 \leq \|I - sM\|^2 \|(x^k - x^*)\|^2 \leq \lambda(I_n - sM)^2 \|x^k - x^*\|$$

Note that $\lambda(I_n - sM)^2$ will be less than 1 if $s \in (0, \frac{2}{\lambda_{max}(M)})$, thus implying convergence. $\square$

Let $s = \frac{1}{\lambda_{max}(M)}$. We can see that:

$$1 - s\lambda_{max}(M) = 1 - \frac{\lambda_{min}(M)}{\lambda_{max}(M)} = 1 - \frac{1}{\kappa}$$

11 Lectures 11, 12: Conjugate Gradient Descent

Previously, we introduced the notion of gradient descent methods, which try to minimize the function after some iterations of convergence. A remark that I would like to make is that conjugate gradient descent refers to the method of solving linear systems using this method.