SI231 - Matrix Computations, Fall 2020-21

Homework Set #3

Prof. Yue Qiu and Prof. Ziping Zhao

Name: **Major:** IE

Student No.: **E-mail:**

Acknowledgements:

1) Deadline: **2020-11-01 23:59:00**
2) Submit your homework at **Gradescope**. Entry Code: **MY3XBJ**. Homework #3 contains two parts, the theoretical part the and the programming part.
3) About the the theoretical part:
   (a) Submit your homework in **Homework 3** in gradescope. Make sure that you have correctly select pages for each problem. If not, you probably will get 0 point.
   (b) Your homework should be uploaded in the **PDF** format, and the naming format of the file is not specified.
   (c) You need to use LaTeXin principle.
   (d) Use the given template and give your solution in English. Solution in Chinese is not allowed.
4) About the programming part:
   (a) Submit your codes in **Homework 3 Programming part** in gradescope.
   (b) Detailed requirements see in Problem 2 and Probelm 3.
5) **No late submission is allowed.**

I. UNDERSTANDING PROJECTION

Problem 1. (5 points $\times$ 3)

Suppose that $\mathbf{P} \in \mathbb{R}^{n\times n}$ is a projector onto a subspace $\mathcal{U}$ along its orthogonal complement $\mathcal{U}^{\perp}$, then it is called the **orthogonal projector** onto $\mathcal{U}$.

1) Prove that an orthogonal projector must be singular if it is not an identity matrix.
2) What is the orthogonal projector onto $\mathcal{U}^{\perp}$ along the subspace $\mathcal{U}$?
3) Let $\mathcal{U}$ and $\mathcal{W}$ be two subspaces of a vector space $\mathcal{V}$, and denote $\mathbf{P}_{\mathcal{U}}$ and $\mathbf{P}_{\mathcal{W}}$ as the corresponding orthogonal projectors, respectively. Prove that $\mathbf{P}_{\mathcal{U}}\mathbf{P}_{\mathcal{W}} = 0$ if and only if $\mathcal{U} \perp \mathcal{W}$.

Solution.

1) P is idempotent; i.e., $p^2 = p$

   suppose the eigenvalues of P is λ and the associated eigenvector is ζ

   then, $P^2\zeta = \lambda p\zeta = \lambda^2\zeta$

   there also have $p^2\zeta = p\zeta = \lambda\zeta$

   $\therefore \quad \lambda^2 = \lambda \quad \Rightarrow \quad \lambda = 0$ or 1

   if the eigenvalues of P all are $\lambda_i = 1, \quad i = 1, 2, \cdots, n$

   and because $\lambda_i = 1 > 0, \quad i = 1, 2, \cdots, n,$

   so P is non-singular matrix. $\therefore P^{-1}P^2 = P^{-1}P \quad \therefore P = I$

   Otherwise s if there have an eigenvalues of P is 0, then the P singular. so orthogonal projector must be singular if it is not an identity matrix.

2) suppose $\alpha_1, \alpha_2, \cdots, \alpha_k$ are the basis of $\mathcal{U}$'s column vectors,

   let $A = [\alpha_1, \alpha_2, \ldots, \alpha_k]$

   then the orthogonal projector onto u is

$$P = A\left(A^{\top}A\right)^{-1}A^{\top}$$

   suppose $\mathcal{A} = \operatorname{span}\{\alpha_1, \alpha_2, \cdots, \alpha_k\}$, we have $\mathcal{A}^{\top} = \mathcal{U}^{\top}$

   so the orthogonal projector onto $\mathcal{U}^{\perp}$ is $P_A^{\perp} = I - A\left(A^{\top}A\right)^{-1}A^{-1}$

3) First suppose $P_U P_W = 0$. Suppose $w \in W$. Then

$$\begin{aligned} 0 &= P_U P_W w \\ &= P_U w \end{aligned}$$

   Hence $w \in \mathcal{N}(P_U)$. so $w \in U^{\perp}$. Thus $\langle u, w\rangle = 0$ for all $u \in U$ completing one direction of the proof.

   To prove the other direction, now suppose that $\langle u, w\rangle = 0$ for all $u \in U$ and all $w \in W$. Thus $U \subset W^{\perp}$ and $W \subset U^{\perp}$. If $w \in W$, then

$$\left(P_U P_W\right)(w) = P_U\left(P_W w\right) = P_U w = 0$$

   where the last equality holds because $w \in U^{\perp}$. If $v \in W^{\perp}$, then

$$\left(P_U P_W\right)(v) = P_U\left(P_W v\right) = P_U 0 = 0$$

since every element in V can be written as the sum of a vector in W and a vector in $W^{\perp}$,

suppose ν is an arbitrary vector in V, and $\nu = w_i + v_j$,

then $(P_U P_W)\nu = 0$ for every ν in V

So that $P_U P_W = 0$, as desired.

II. Least Square (LS) programming.

Problem 2. (10 points + 10 points + 5 points)

Write programs to solve the least square problem with specified methods, any programming language is suitable.

$$\mathbf{x} = \arg\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}), \quad f(\mathbf{x}) = ||\mathbf{y} - \mathbf{A}\mathbf{x}||_2^2$$

where $\mathbf{A} \in \mathbb{R}^{m\times n}$ is a matrix representing the predefined data set with m data samples of n dimensions (m=1000, n=210), and $\mathbf{y} \in \mathbb{R}^m$ represents the labels. The data samples are provided in the "data.txt" file, and the labels are provided in the "label.txt" file, you are supposed to load the data before solving the problem.

1) Solve the LS with gradient decent method.

   The gradient descent method for solving problem updates $\mathbf{x}$ as

   $$\mathbf{x} = \mathbf{x} - \gamma \cdot \nabla_{\mathbf{x}} f(\mathbf{x}),$$

   where γ is the step size of the gradient decent methods. We suggest that you can set $\gamma = 1e-5$.
2) Solve the LS by the method of normal equation with Cholesky decomposition and forward/backward substitution.
3) Compare two methods above.
   (a) Basing on the true running results from the program, count the number of "flops"*;
   (b) Compare gradient norm and loss $f(\mathbf{x})$ for results $\mathbf{x} = \mathbf{x}_{\mathbf{LS}}$ of above two algorithms.

Notation*: "flop": one flop means one floating point operation, i.e., one addition, subtraction, multiplication, or division of two floating-point numbers, in this problem each floating points operation $+, -, \times, \div, \sqrt{\cdot}$ counts as one "flop".

Hint for gradient decent programming:

1) **Step size selection**: to ensure the convergence of the method, γ is supposed to be selected properly (large step size may accelerate the convergence rate but also may lead to instability, A sufficiently small compensation always ensures that the algorithm converges).
2) **Terminal condition**: the gradient decent is an iteration algorithm that need a terminal condition. In this problem, the algorithm can stop when the gradient of the loss function $f(\mathbf{x})$ at current $\mathbf{x}$ is small enough.

Remarks:

- The solution of the two methods should be printed in files named "sol1.txt" and "sol2.txt" and submitted in gradescope. The format should be same as the input file (210 rows plain text, each rows is a dimension of the final solution).
- Make sure that your codes are executable and are consistent with your solutions.

Solution.

1)
```
clear;
clc;
```

```
y = load('label.txt');
A = load('data.txt');
x=inv(A'*A)*A'*y;
fid=fopen(['E:\MatlabFile\LS\','sol.txt'],'w');
for m=1:length(x)
    fprintf(fid,'%.16f\r\n',x(m));
end
f_x=norm(y-A*x)^2
fclose(fid);
x1= Gradient_descent(A,y);
fid=fopen(['E:\MatlabFile\LS\','sol1.txt'],'w');
for i=1:length(x1)
    fprintf(fid,'%.16f\r\n',x1(i));
end
f_x1=norm(y-A*x1)^2
fclose(fid);
x2=Cholesky_LS(A,y);
fid=fopen(['E:\MatlabFile\LS\','sol2.txt'],'w');
for j=1:length(x2)
    fprintf(fid,'%.16f\r\n',x2(j));
end
f_x2= norm(y-A*x2)^2
fclose(fid);
```

```
function x = Gradient_descent(A,y)
tA = A;
x = ones(size(A,2),1);
k = 0;
while true
    gradient_descent = zeros(size(tA,2),1);
    z = A*x;
    gradient_descent = gradient_descent + 0.00001*A'*(y - z);
    x = x + gradient_descent;
    k = k+1;
    if ( norm(y - A*x)^2 ) <= 2.664261406344091e+04 | k>10000
        break;
```

```
    end
    fprintf('iterator times %d, error %f\n',k,( norm(y - A*x)^2 ));
end
```

2)

```
function x = Cholesky_LS(A,y)
C=A'*A;
L=Cholesky_decomposition(C);
y=LU(L,A'*y);
x=LU(L',y);
```

```
function L=Cholesky_decomposition(A)
[N,N]=size(A);
X=zeros(N,1);
Y=zeros(N,1);
for i=1:N
    A(i,i)=sqrt(A(i,i)-A(i,1:i-1)*A(i,1:i-1)');
    if A(i,i)==0
        'A is singular, no unique solution';
        break;
    end
    for j=i+1:N
        A(j,i)=(A(j,i)-A(j,1:i-1)*A(i,1:i-1)')/A(i,i);
    end
end
for x=1:N
    for y=1:N
        B(x,y)=A(x,y);
        if x<y
            B(x,y)=0;
            break;
        end
    end
end
L=B;
```

```
function LU_ans = LU(A,b)
N=length(A);
```

```
z = zeros(N,1);
LU_ans = zeros(N,1);
L=eye(N);%Let the L matrix be an identity matrix at first
for i=1:N-1
    for j=i+1:N
        L(j,i)=A(j,i)/A(i,i);
        A(j,:)=A(j,:)-(A(j,i)/A(i,i))*A(i,:);
    end
end
for p=1:1:N
    z(p)=b(p)/L(p,p);
    for q=1:1:p-1
        z(p)=z(p)-z(q)*L(p,q)/L(p,p);
    end
end
for p_2=N:-1:1
    LU_ans(p_2)=z(p_2)/A(p_2,p_2);
    for q_2=N:-1:p_2+1
        LU_ans(p_2)=LU_ans(p_2)-LU_ans(q_2)*A(p_2,q_2)/A(p_2,p_2);
    end
end
```

3) (a) Basing on the true running results from the program in the gradient descent method, it iterated 165 times to approximate x_{LS}, and in each approximation, if A is $m \times n$ matrix, then it uses $(mn^2+mn+m+3n)$ flops, so it totally uses $165 \times (1000 \times 210^2 + 1000 \times 210 + 1000 + 210 \times 3) = 44311630$ flops.
From the program in the Cholesky decomposition and forward/backward substitution method. In the $Function \ \ L = Cholesky\_decomposition(A)$, it can be computed in $\mathcal{O}\left(n^3/3\right)$, because $n = 210$, so it uses $\frac{1}{3}n^3 + m^2n + m \times n + \frac{2}{3}n^3 \times 2 = 225645000$ flops.
(b) In the code script Main.m, I solved that $f(x_1) = 2.664261406344088e{+}04$ and $f(x_2) = 2.664261406344087e{+}04$

III. Understanding the QR Factorization

Problem 3 [Understanding the Gram-Schmidt algorithm.]. (5 points + 7 points + 6 points + 7 points)

1) Consider the subspace $\mathcal{S}$ spaned by $\{\mathbf{a}_1, \dots, \mathbf{a}_4\}$, where

$$\mathbf{a}_1 = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}, \quad \mathbf{a}_2 = \begin{bmatrix} 2 \\ 3 \\ 4 \\ 5 \end{bmatrix}, \quad \mathbf{a}_3 = \begin{bmatrix} 3 \\ 4 \\ 5 \\ 6 \end{bmatrix}, \quad \mathbf{a}_4 = \begin{bmatrix} 3 \\ 5 \\ 7 \\ 11 \end{bmatrix}.$$

Use the **classical** Gram-Schimidt algorithm (See Algorithm 1), find a set of orthonormal basis $\{\mathbf{q}_i\}$ for $\mathcal{S}$ by hand (derivation is expected). Do not use decimals in your answers, fraction and n-th roots of numbers are accepted. Verify the orthonormality of the found basis.

Algorithm 1: Classical Gram-Schimidt algorithm

Input : A collection of linearly independent vectors $\mathbf{a}_1, \dots, \mathbf{a}_n$.

Initilization: $\widetilde{\mathbf{q}}_1 = \mathbf{a}_1, \mathbf{q}_1 = \widetilde{\mathbf{q}}_1 / \|\widetilde{\mathbf{q}}_1\|_2$

for $i = 2, \dots, n$ **do**

  $\widetilde{\mathbf{q}}_i = \mathbf{a}_i - \sum_{j=1}^{i-1} (\mathbf{q}_j^T \mathbf{a}_i)\mathbf{q}_j$

  $\mathbf{q}_i = \widetilde{\mathbf{q}}_i / \|\widetilde{\mathbf{q}}_i\|_2$

end

Output: $\mathbf{q}_1, \dots, \mathbf{q}_n$

---

2) Orthogonal projection of vector $\mathbf{a}$ onto a nonzero vector $\mathbf{b}$ is defined as

$$\text{proj}_{\mathbf{b}}(\mathbf{a}) = \frac{\langle \mathbf{a}, \mathbf{b} \rangle}{\langle \mathbf{b}, \mathbf{b} \rangle} \mathbf{b},$$

where $\langle , \rangle$ denotes the inner product of vectors. And for subspace $\mathcal{M}$ with orthonormal basis $\{\mathbf{u}_1, \dots, \mathbf{u}_k\}$, the orthogonal projector onto subspace $\mathcal{M}$ is given by

$$\mathbf{P} = \mathbf{U}\mathbf{U}^T, \quad \mathbf{U} = [\mathbf{u}_1 | \cdots | \mathbf{u}_k].$$

In the context of **projection of vector** and **projection onto subspace** respectively, can you give another two understandings of the classical Gram-Schmidt algorithm?

3) Consider the subspace $\mathcal{S}$ spaned by $\{\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3\}$,

$$\mathbf{a}_1 = \begin{bmatrix} 1 \\ \epsilon \\ \epsilon \end{bmatrix}, \quad \mathbf{a}_2 = \begin{bmatrix} 1 \\ \epsilon \\ 0 \end{bmatrix}, \quad \mathbf{a}_3 = \begin{bmatrix} 1 \\ 0 \\ \epsilon \end{bmatrix},$$

where ϵ is a small real number such that $1 + k\epsilon^2 = 1$ ($k \in \mathbb{N}^+$). First complete the pseudo algorithm in Algorithm 2. Then use the **classical** Gram-Schimidt algorithm and the **modified** Gram-Schimidt algorithm respectively, find two sets of basis for $\mathcal{S}$ by hand (derivation is expected). Are the two sets of basis the same? If not, which one is the desired orthonormal basis? Report what you have found.

Algorithm 2: Modified Gram-Schmidt algorithm

Input : A collection of linearly independent vectors $\mathbf{a}_1, \ldots, \mathbf{a}_n$.

Initilization: $\tilde{q}_1 = a_1, \hat{q}_2 = a_2 \quad \cdots, \tilde{q}_n = a_n$

for $i = 1, \ldots, n$ **do**

  $\widetilde{\mathbf{q}}_i = \frac{\tilde{\mathbf{q}}_i}{\|\widetilde{\mathbf{q}}_i\|^2}$

  for $j = i + 1, \ldots, n$ **do**

    $\widetilde{\mathbf{q}}_j = \widetilde{\mathbf{q}}_j - (\widetilde{\mathbf{q}}_j \mathbf{q}_i)\mathbf{q}_i$

  end

end

Output: $\mathbf{q}_1, \ldots, \mathbf{q}_n$

4) **Programming part:** In this part, you are required to code both the **classical Gram-Schmidt** and **the modified Gram-Schmidt** algorithms. For $\epsilon = 1\mathrm{e}{-4}$ and $\epsilon = 1\mathrm{e}{-9}$ in sub-problem 2), give the outputs of two algorithms and calculate $\|\mathbf{Q}^T\mathbf{Q} - \mathbf{I}\|_{\mathrm{F}}$, where $\mathbf{Q} = [\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3]$.

Remarks:

- Coding languages are restricted, but do not use built-in function such as `qr`.
- When handing in your homework in gradescope, package all your codes into your_student_id+hw3_code.zip and upload. In the package, you also need to include a file named README.txt/md to clearly identify the function of each file.
- Make sure that your codes can run and are consistent with your solutions.

Solution.

1)

$$\tilde{q}_1 = a_1, \quad q_1 = \frac{\hat{q}_1}{\|\hat{q}_1\|_2} = \left[\frac{1}{\sqrt{30}}, \frac{2}{\sqrt{30}}, \frac{3}{\sqrt{30}}, \frac{4}{\sqrt{30}}\right]^\top$$

$$\tilde{q}_2 = a_2 - \left(q_1^\top a_2\right) q_1 = \begin{bmatrix} 2 \\ 3 \\ 4 \\ 5 \end{bmatrix} - \frac{2 + 6 + 12 + 20}{\sqrt{30}} \cdot \begin{bmatrix} \frac{1}{\sqrt{30}} \\ \frac{2}{\sqrt{30}} \\ \frac{3}{\sqrt{30}} \\ \frac{4}{\sqrt{30}} \end{bmatrix} = \left[\frac{2}{3}, \frac{1}{3}, 0, -\frac{1}{3}\right]$$

$$\therefore q_2 = \frac{\tilde{q}_2}{\|\tilde{q}_2\|} = \left[\frac{2}{\sqrt{6}}, \frac{1}{\sqrt{6}}, 0, -\frac{1}{\sqrt{6}}\right]^\top$$

$$\tilde{q}_3 = a_3 - \left(q_1^\top a_3\right) q_1 - \left(q_2^\top a_3\right) q_2 = \begin{bmatrix} 3 \\ 4 \\ 5 \\ 6 \end{bmatrix} - \frac{3 + 8 + 15 + 24}{\sqrt{30}} \begin{bmatrix} \frac{1}{\sqrt{30}} \\ \frac{2}{\sqrt{30}} \\ \frac{3}{\sqrt{30}} \\ \frac{4}{\sqrt{30}} \end{bmatrix} - \frac{6 + 4 - 6}{\sqrt{6}} \begin{bmatrix} \frac{2}{\sqrt{6}} \\ \frac{1}{\sqrt{6}} \\ 0 \\ -\frac{1}{\sqrt{6}} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$q_3 = \tilde{q}_3 = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^\top$

$$\tilde{q}_4 = a_4 - \left(q_1^\top a_4\right) q_1 - \left(q_2^\top a_4\right) q_2 - \left(q_3^\top a_4\right) q_3 = \begin{bmatrix} 3 \\ 5 \\ 7 \\ 11 \end{bmatrix} - \frac{3+10+21+44}{\sqrt{30}} \begin{bmatrix} \frac{1}{\sqrt{30}} \\ \frac{1}{\sqrt{30}} \\ \frac{1}{\sqrt{30}} \\ \frac{1}{\sqrt{30}} \end{bmatrix} = \begin{bmatrix} \frac{2}{5} \\ -\frac{1}{5} \\ -\frac{4}{5} \\ \frac{3}{5} \end{bmatrix}$$

$\therefore q_4 = \frac{\tilde{q}_4}{\|\tilde{q}_4\|} = \left[\frac{2}{\sqrt{30}}, -\frac{1}{\sqrt{30}}, -\frac{4}{\sqrt{30}}, \frac{3}{\sqrt{30}}\right]^\top$

2) In the context of projection of vector, we can have a geometrical understanding of the classical Gram-Schmidt algorithm:

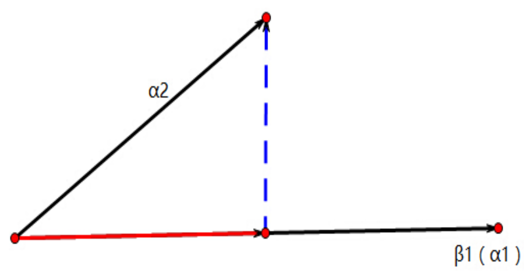
Given a basis $\alpha_1, \alpha_2, \ldots, \alpha_n$, convert it to another orthonormal basis $\beta_1, \beta_2, \ldots, \beta_n$ so that the two bases are equivalent.

$\beta_1 = \alpha_1$

$\beta_2 = \alpha_2 - \frac{(\alpha_2,\beta_1)}{(\beta_1,\beta_1)}\beta_1 \ \ldots$

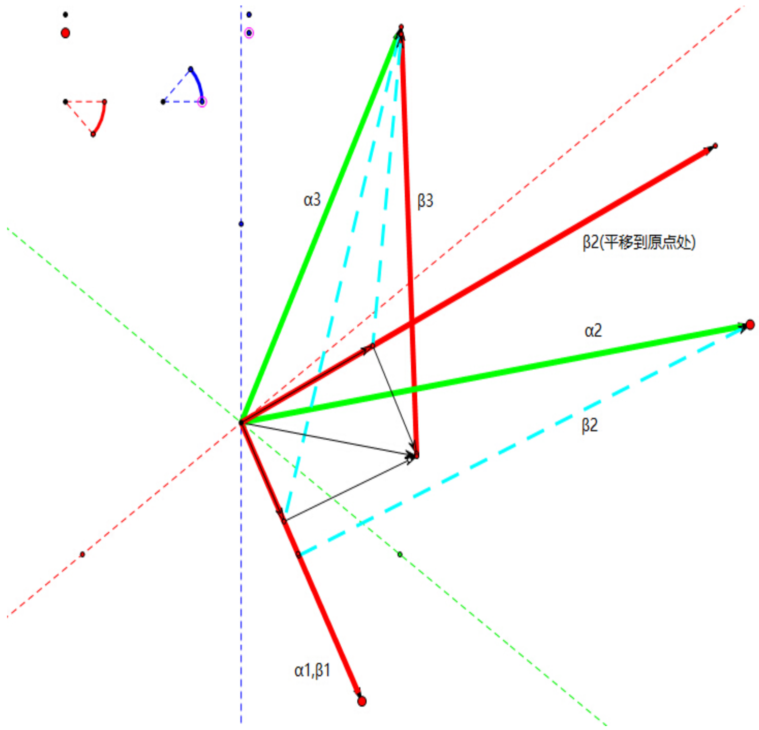
$\beta_n = \alpha_n - \frac{(\alpha_n,\beta_1)}{(\beta_1,\beta_1)}\beta_1 - \frac{(\alpha_n,\beta_2)}{(\beta_2,\beta_2)}\beta_2 - \ldots - \frac{(\alpha_n,\beta_{n-1})}{(\beta_{n-1},\beta_{n-1})}\beta_{n-1}$

First of all,Orthogonal projection of vector α onto a nonzero vector β is defined as $\mathrm{proj}_\beta(\alpha) = \frac{\langle\alpha,\beta\rangle}{\langle\beta,\beta\rangle}\beta$,



The red vector is the projection part, so the blue vector is $\alpha_2 - \frac{(\alpha_2,\beta_1)}{(\beta_1,\beta_1)}\beta$, that corresponds to Gram-Schmidt algorithm's β_2. So β_2 is perpendicular to β_1

When the number of vectors is 3, the geometric interpretation of the corresponding three-dimensional space



is shown in the figure:

The green vector is the original base α_i, α_1 is red because α_1 is also β_1, I'm going to shift β_2 to the origin, then the orthogonal projection of α_3 onto XoY plane is $\alpha_3 - \frac{(\alpha_3,\beta_1)}{(\beta_1,\beta_1)}\beta_1 - \frac{(\alpha_3,\beta_2)}{(\beta_2,\beta_2)}\beta_2$

We can see that β_1 β_2 β_3 are orthoogonal, it can also be extended to Euclidean space above three dimensions.

3) Use the **classical** Gram-Schimidt algorithm:

Suppose ε is positive,

$\tilde{q}_1 = a_1 \quad q_1 = \frac{\hat{q}_1}{\|q_1\|} = \left[\frac{1}{\sqrt{1+2\varepsilon^2}}, \frac{\varepsilon}{\sqrt{1+2\varepsilon^2}}, \frac{\varepsilon}{\sqrt{1+2\varepsilon^2}}\right]^\top = \begin{bmatrix} 1 & \varepsilon & \varepsilon \end{bmatrix}^\top$

$$\tilde{q}_2 = a_2 - \left(q_1^\top a_2\right) q_1 = \begin{bmatrix} 1 \\ \varepsilon \\ 0 \end{bmatrix} - \left(1+\varepsilon^2\right) \begin{bmatrix} 1 \\ \varepsilon \\ \varepsilon \end{bmatrix} = \begin{bmatrix} -\varepsilon^2 \\ -\varepsilon^3 \\ -\varepsilon - \varepsilon^3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -\varepsilon \end{bmatrix}, \|\tilde{q}_i\| = \varepsilon$$

$$q_2 = \frac{\tilde{q}_2}{\|\tilde{q}_n\|} = \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix}$$

$$\tilde{q}_3 = a_3 - \left(q_1^\top a_3\right) q_1 - \left(q_2^\top u_3\right) q_2 = \begin{bmatrix} 1 \\ 0 \\ \varepsilon \end{bmatrix} - \left(1+\varepsilon^2\right) \begin{bmatrix} 1 \\ \varepsilon \\ \varepsilon \end{bmatrix} - (-\varepsilon) \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ -\varepsilon \\ -\varepsilon \end{bmatrix}$$

$q_3 = \frac{\tilde{q}_3}{\|\tilde{q}_3\|} = \begin{bmatrix} 0 & -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}^\top$

Use the **modified** Gram-Schimidt algorithm:

$\tilde{q}_1 = a_1 \quad , \quad q_1 = \frac{\hat{q}_1}{\|\hat{q}_1\|} = \left[\frac{1}{\sqrt{1+2\varepsilon^2}}, \frac{\varepsilon}{\sqrt{1+2\varepsilon^2}}, \frac{\varepsilon}{\sqrt{1+2\varepsilon^2}}\right]^\top = \begin{bmatrix} 1 & \varepsilon & \varepsilon \end{bmatrix}^\top$

$$\tilde{q}_2 = a_2 - \left(q_1^T a_2\right) q_1 = \begin{bmatrix} 1 \\ \varepsilon \\ 0 \end{bmatrix} - \left(1+\varepsilon^2\right) \begin{bmatrix} 1 \\ \varepsilon \\ \varepsilon \end{bmatrix} = \begin{bmatrix} -\varepsilon^2 \\ -\varepsilon^3 \\ -\varepsilon - \varepsilon^3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -\varepsilon \end{bmatrix}$$

$$q_2 = \frac{\tilde{q}_2}{\|\tilde{q}_2\|} = \begin{bmatrix} 0 & 0 & -1 \end{bmatrix}^\top$$

$$a_3^{(1)} = a_3 - \left(q_1^\top a_3\right) q_1 = \begin{bmatrix} 1 \\ 0 \\ \varepsilon \end{bmatrix} - \left(1+\varepsilon^2\right) \begin{bmatrix} 1 \\ \varepsilon \\ \varepsilon \end{bmatrix} = \begin{bmatrix} 0 \\ -\varepsilon \\ 0 \end{bmatrix}$$

$$a_3^{(2)} = a_3^{(1)} - \left(q_2^\top a_3^{(1)}\right) q_2 = \begin{bmatrix} 0 \\ -\varepsilon \\ 0 \end{bmatrix} - 0 = \begin{bmatrix} 0 \\ -\varepsilon \\ 0 \end{bmatrix} \therefore q_3 = \frac{a_3^{(2)}}{\left\|a_3^{(2)}\right\|} = \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix}$$

so the two sets of basis are not same,the one that use the modified Gram-Schimidt algorithm is the desired orthonormal basis.In the classical Gram-Schimidt algorithm's answer, we get $q_2^\top q_3 = \frac{1}{\sqrt{2}} \neq 0$, so q_2 and q_3 are not orthogonal. But in the classical Gram-Schimidt algorithm's answer, we get $q_1^\top q_2 = -\varepsilon^2 = 0, \quad q_1^\top q_3 = -\varepsilon, \quad q_2^\top q_3 = 0$, so q_1, q_1, q_1 are orthogonal except a slight error ε.

4)
```
clear all;clc;
% a1=[1;2;3;4];
% a2=[2;3;4;5];
% a3=[3;4;5;6];
% a4=[3;5;7;11];
% A=[a1 a2 a3 a4];
e=1e-4;
a1=[1;e;e];
a2=[1;e;0];
a3=[1;0;e];
A=[a1 a2 a3];
Q_classical= Classical_GS(A)
normF_1=norm(Q_classical,'fro')
Q_modified= Modified_GS(A)
normF_2=norm(Q_modified,'fro')
e=1e-9;
Q_classical_1= Classical_GS(A)
normF_11=norm(Q_classical_1,'fro')
Q_modified_2= Modified_GS(A)
```

```
normF_2=norm(Q_modified_2,'fro')
```

The output is:

```
Q_classical =                    %e=1e-4
    0.999999990000000    0.000099999998782    0.000099999999892
    0.000099999999000    0.000000010000000   -0.999999995000000
    0.000099999999000   -0.999999995000000    0.000000002820298
normF_1 =
      3.988504106090590e-09
Q_modified =
    0.999999990000000    0.000099999998782    0.000099999999782
    0.000099999999000    0.000000010000000   -0.999999995000000
    0.000099999999000   -0.999999995000000    0.000000000000000
normF_2 =
      5.640596652684282e-13
Q_classical_1 =                 %e=1e-9
    0.999999990000000    0.000099999998782    0.000099999999892
    0.000099999999000    0.000000010000000   -0.999999995000000
    0.000099999999000   -0.999999995000000    0.000000002820298
normF_11 =
      3.988504106090590e-09
Q_modified_2 =
    0.999999990000000    0.000099999998782    0.000099999999782
    0.000099999999000    0.000000010000000   -0.999999995000000
    0.000099999999000   -0.999999995000000    0.000000000000000
normF_2 =
      5.640596652684282e-13
```

IV. SOLVING LS VIA QR FACTORIZATION AND NORMAL EQUATION

Problem 4 [Understanding the influence of the condition number to the solution.]. (4 points + 5 points + 4 points + 4 points + 3 points points)

Consider such two LS problems:

$$\min_{\mathbf{x}\in\mathbb{R}^n} \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2 \tag{1}$$

$$\min_{\mathbf{x}\in\mathbb{R}^n} \|\mathbf{A}\mathbf{x}-(\mathbf{b}+\delta\mathbf{b})\|_2^2$$

with $\mathbf{A}\in\mathbb{R}^{m\times n}$. For $\mathbf{b}=\begin{bmatrix}1 & 3/2 & 3 & 6\end{bmatrix}^T$ and $\delta\mathbf{b}=\begin{bmatrix}1/10 & 0 & 0 & 0\end{bmatrix}^T$,

1) Computing solution to the problem (1) via QR decomposition when
$$\mathbf{A}=\begin{bmatrix}1 & 2 & 3\\ 2 & 3 & 5\\ 3 & 4 & 7\\ 4 & 5 & 11\end{bmatrix}.$$
2) For a full-rank matrix $\mathbf{A}$, consider the equation $\mathbf{A}\mathbf{x}=\mathbf{b}$, after adding some noise $\delta\mathbf{b}$ to $\mathbf{b}$, we have $\mathbf{A}(\mathbf{x}+\delta\mathbf{x})=\mathbf{b}+\delta\mathbf{b}$, and then proof
$$\frac{1}{\|\mathbf{A}\|\|\mathbf{A}^\dagger\|}\frac{\|\delta\mathbf{b}\|}{\|\mathbf{b}\|}\le\frac{\|\delta\mathbf{x}\|}{\|\mathbf{x}\|}\le\|\mathbf{A}\|\|\mathbf{A}^\dagger\|\frac{\|\delta\mathbf{b}\|}{\|\mathbf{b}\|},$$
and give it a plain interpretation.
3) Computing the solutions to the two LS problems via the normal equation $\mathbf{A}^T\mathbf{A}\mathbf{x}_{LS}=\mathbf{A}^T\mathbf{b}$ when
$$\mathbf{A}=\begin{bmatrix}1 & 2 & 2\\ 2 & 2 & 2\\ 3 & 3 & 3\\ 1 & 1 & 0\end{bmatrix}.$$
4) Computing the solutions to the two LS problems via the normal equation $\mathbf{A}^T\mathbf{A}\mathbf{x}_{LS}=\mathbf{A}^T\mathbf{b}$ when
$$\mathbf{A}=\begin{bmatrix}1 & 1 & 1\\ 1 & 2 & 4\\ 1 & 3 & 9\\ 1 & 4 & 16\end{bmatrix}.$$
5) Compare the 2-norm condition number $\|\mathbf{A}\|\|\mathbf{A}^\dagger\|$ for $\mathbf{A}$ in 3) and 4) and the influence on the solution to problem (1) resulted by the additional noise $\delta\mathbf{b}$.

   Hint: Show the influence on the solution by $\frac{\|\delta\mathbf{x}\|}{\|\mathbf{x}\|}$.

Remarks: You can use MATLAB for some matrix computations (deviation is expected) in 3), 4), 5). Do not use decimals in your answers, fraction and n-th roots of numbers are accepted.

Solution.

1) let $r_{ii} = \|\tilde{\mathbf{q}}_i\|_2, r_{ji} = \mathbf{q}_j^T\mathbf{a}_i$ for $j = 1, \ldots, i-1$ we see that $\mathbf{a}_i = \sum_{j=1}^{i} r_{ji}\mathbf{q}_i$ for all i, or, equivalently,

$$\begin{bmatrix} \mathbf{a}_1 & \mathbf{a}_2 & \ldots & \mathbf{a}_n \end{bmatrix} = \begin{bmatrix} \mathbf{q}_1 & \mathbf{q}_2 & \ldots & \mathbf{q}_n \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ & r_{22} & \cdots & r_{2n} \\ & & \ddots & \vdots \\ & & & r_{nn} \end{bmatrix}$$

use the solution above, we can get

$$\tilde{q}_1 = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}, q_1 = \begin{bmatrix} \frac{1}{\sqrt{30}} \\ \frac{2}{\sqrt{30}} \\ \frac{3}{\sqrt{30}} \\ \frac{4}{\sqrt{30}} \end{bmatrix}, \quad \tilde{q}_2 = \begin{bmatrix} \frac{2}{3} \\ \frac{1}{3} \\ 0 \\ -\frac{1}{3} \end{bmatrix}, q_2 = \begin{bmatrix} \frac{2}{\sqrt{b}} \\ \frac{1}{\sqrt{6}} \\ 0 \\ -\frac{1}{\sqrt{6}} \end{bmatrix} \tilde{q}_3 = \begin{bmatrix} \frac{2}{5} \\ -\frac{1}{5} \\ -\frac{4}{5} \\ \frac{3}{5} \end{bmatrix} q_3 = \begin{bmatrix} \frac{2}{\sqrt{30}} \\ -\frac{1}{\sqrt{30}} \\ -\frac{4}{\sqrt{30}} \\ \frac{3}{\sqrt{30}} \end{bmatrix}$$

$$\text{So} \quad Q_1 = \begin{bmatrix} q_1 & q_2 & q_3 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{30}} & \frac{2}{\sqrt{6}} & \frac{2}{\sqrt{30}} \\ \frac{2}{\sqrt{30}} & \frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{30}} \\ \frac{3}{\sqrt{30}} & 0 & -\frac{4}{\sqrt{30}} \\ \frac{4}{\sqrt{30}} & -\frac{1}{\sqrt{6}} & \frac{3}{\sqrt{30}} \end{bmatrix}, \quad R_1 = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ & & \ddots & \vdots \\ & & & r_{nn} \end{bmatrix} = \begin{bmatrix} \sqrt{3}_0 & \frac{40}{\sqrt{30}} & \frac{78}{\sqrt{30}} \\ 0 & \frac{2}{\sqrt{6}} & 0 \\ 0 & 0 & \frac{\sqrt{30}}{5} \end{bmatrix}$$

$min\{\|Ax-b\|_2^2\} = min\{\left\|Q_1^\top b - R_1 x\right\|_2^2\}$ (because $Q_2 = 0$)

$$z = Q_1^T b = \begin{bmatrix} \frac{1}{\sqrt{30}} & \frac{2}{\sqrt{30}} & \frac{3}{\sqrt{30}} & \frac{4}{\sqrt{30}} \\ \frac{2}{\sqrt{6}} & \frac{1}{\sqrt{6}} & 0 & -\frac{1}{\sqrt{6}} \\ \frac{2}{\sqrt{20}} & -\frac{1}{\sqrt{30}} & -\frac{4}{\sqrt{30}} & \frac{3}{\sqrt{30}} \end{bmatrix} \begin{bmatrix} 1 \\ \frac{3}{2} \\ 3 \\ 6 \end{bmatrix} = \left[\frac{37}{\sqrt{30}} - \frac{5}{2\sqrt{6}}, \frac{13}{2\sqrt{30}}\right]^\top$$

then solve $\quad R_1 x = z$

$x = R_1^{-1} z = \begin{bmatrix} \frac{1}{12} & -\frac{5}{4} & \frac{13}{12} \end{bmatrix}^\top$

2) $\because A(x+\delta x) = b + \delta b$

$\therefore x(1+\delta) = A^\dagger(1+\delta)b$

$\therefore \delta x = \delta A^\dagger b$

$\therefore \|\delta x\| = \left\|\delta A^\dagger b\right\| \leqslant \left\|A^\dagger\right\| \|\delta b\|$

$\therefore Ax = b$

$\therefore \|b\| = \|Ax\| \leqslant \|A\|\|x\|$

$\frac{1}{\|x\|} \leqslant \frac{\|A\|}{\|b\|}$

multiply $(*)$ and $(**)$, we can get $\frac{\|\delta x\|}{\|x\|} \leqslant \|A\| \left\|A^\dagger\right\| \frac{\|\delta b\|}{\|b\|}$

$\because \delta Ax = \delta b$

$\therefore \|\delta b\| \leqslant \|\delta Ax\| \leqslant \|A\|\|\delta x\|$

$\therefore \frac{\|\delta b\|}{\|A\|} \leqslant \|\delta x\|$

$\because x = A^\dagger b$

$\therefore \|x\| \leqslant \left\|A^\dagger\right\| \|b\|$

$\therefore \frac{1}{\|A^{\dagger}\|\|b\|} \leqslant \frac{1}{\|x\|}$

multiply (1) and (2), we can get $\frac{1}{\|\mathbf{A}\|\|\mathbf{A}^{+}\|}\frac{\|\delta\mathbf{b}\|}{\|\mathbf{b}\|} \leqslant \frac{\|\delta\mathbf{x}\|}{\|\mathbf{x}\|}$

so $\frac{1}{\|\mathbf{A}\|\|\mathbf{A}^{+}\|}\frac{\|\delta\mathbf{b}\|}{\|\mathbf{b}\|} \leq \frac{\|\delta\mathbf{x}\|}{\|\mathbf{x}\|} \leq \|\mathbf{A}\|\left\|\mathbf{A}^{\dagger}\right\|\frac{\|\delta\mathbf{b}\|}{\|\mathbf{b}\|}$

3) $x_1 = \left(A^{\top}A\right)^{-1}A^{\top}b = \begin{bmatrix} \frac{11}{13} & \frac{67}{13} & -\frac{66}{13} \end{bmatrix}^{\top}$

$x_2 = \left(A^{\top}A\right)^{-1}A^{\top}(b+\delta b) = \left[\frac{97}{130}\,\frac{683}{130} - \frac{66}{13}\right]^{\top}$

4) $x_1 = \left(A^{\top}A\right)^{-1}A^{\top}b = \left[\frac{15}{8} - \frac{59}{40} \quad \frac{5}{8}\right]^{\top}$

$x_2 = \left(A^{\top}A\right)^{-1}A^{\top}(b+\delta b) = \left[\frac{21}{10} - \frac{163}{100}\,\frac{13}{20}\right]^{\top}$

5) $A^{\dagger} = \left(A^{\top}A\right)^{-1}A^{\top}$

For problem $(3):\quad \|A\| = 6.982 \quad \left\|A^{\dagger}\right\| = 1.908$

$So, \quad \|A\|\left\|A^{\dagger}\right\| = 13.325$

$\delta x = x_2 - x_1 = \left[-\frac{1}{10}, \frac{1}{10}, 0\right]^{\top}$

$\frac{\|\delta x\|}{\|x\|} = \frac{\|x_2 - x_1\|}{\|x_1\|} = \frac{125}{6438} = 0.0194$

For problem $(4):\quad \|A\| = 19.621 \quad \left\|A^{\dagger}\right\| = 3.756$

$So, \quad \|A\|\left\|A^{\dagger}\right\| = 73.694$

$\delta x = x_2 - x_1 = \left[\frac{9}{40}, -\frac{31}{200}, \frac{1}{40}\right]^{\top}$

$\frac{\|\delta x\|}{\|x\|} = \frac{\|x_2 - x_1\|}{\|x_1\|} = \frac{1231}{11065} = 0.1113$

V. Underdetermined System

Problem 5 [Solving Underdetermined System by QR]. (10 points + 5 points)

Consider the following underdetermined system $\mathbf{Ax} = \mathbf{b}$ with $\mathbf{A} \in \mathbb{R}^{m\times n}$ and $m < n$. Let

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & 2 & 0 \\ 0 & -2 & 2 & 1 \\ 2 & 5 & 6 & 1 \end{bmatrix}, \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix},$$

1) Use Householder reflection to give the full QR decomposition of tall $\mathbf{A}^T$, i.e., $\mathbf{A}^T = \mathbf{QR}$ with $\mathbf{Q}$ being a square matrix with orthonormal columns.
2) Give one possible solution via QR decomposition of $\mathbf{A}^T$, write down your solution using $\mathbf{b}$.

Solution.

1) $A^T = \begin{bmatrix} 1 & 0 & 2 \\ 2 & -2 & 5 \\ 2 & 2 & 6 \\ 0 & 1 & 1 \end{bmatrix} = [\alpha_1, \alpha_2, \alpha_3]$

$v_1 = a_1 - \|\alpha_1\| e_1 = \begin{bmatrix} -2 & 2 & 2 & 0 \end{bmatrix}^\top$

$H_1 = I - \frac{2}{\|v_1\|_2^2} v_1 v_1^\top$

Then $A^{(1)} = H_1 A^\top = \begin{bmatrix} 3 & 0 & 8 \\ 0 & -2 & -1 \\ 0 & 2 & 0 \\ 0 & 1 & 1 \end{bmatrix}$

$u_2 = \begin{bmatrix} -2 & 2 & 1 \end{bmatrix}^1$

$\therefore v_2 = u_2 - \|u_2\| e_1 = \begin{bmatrix} -5 & 2 & 1 \end{bmatrix}^\top$

$\therefore \hat{H}_2 = I - \frac{2}{\|v_2\|_2^2} v_2 v_2^\top = \begin{bmatrix} -\frac{2}{3} & \frac{2}{3} & \frac{1}{3} \\ \frac{2}{3} & \frac{11}{15} & -\frac{2}{15} \\ \frac{1}{3} & -\frac{2}{15} & \frac{14}{15} \end{bmatrix}$

$H_2 = \begin{bmatrix} 1 & 0 \\ 0 & \tilde{H}_2 \end{bmatrix}$

$A^{(2)} = H_2 A^{(1)} = H_2 H_1 A^\top = \begin{bmatrix} 3 & 0 & 8 \\ 0 & 3 & 1 \\ 0 & 0 & -\frac{4}{5} \\ 0 & 0 & \frac{3}{5} \end{bmatrix}$

$u_3 = [-\frac{4}{5} \quad \frac{3}{5}]^\top$

$\therefore v_3 = u_3 - \|u_3\| e_1 = [-\frac{9}{5} \quad \frac{3}{5}]^\top$

$\therefore \tilde{H}_3 = I - \frac{2}{\|V_3\|_2^2} v_3 v_3^\top = \begin{bmatrix} -\frac{4}{5} & \frac{3}{5} \\ \frac{3}{5} & \frac{4}{5} \end{bmatrix}$

$$\therefore H_3 = \begin{bmatrix} I_2 & 0 \\ 0 & \tilde{H}_3 \end{bmatrix}$$

$$then \quad A^{(3)} = H_3A^{(2)} = H_3H_2H_1A^T = \begin{bmatrix} 3 & 0 & 8 \\ 0 & 3 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} = R$$

$$Q = H_1H_2H_3 = \begin{bmatrix} \frac{1}{3} & 0 & -\frac{2}{3} & \frac{2}{3} \\ \frac{2}{3} & -\frac{2}{3} & \frac{1}{3} & 0 \\ \frac{2}{3} & \frac{2}{3} & 0 & -\frac{1}{3} \\ 0 & \frac{1}{3} & \frac{2}{3} & \frac{2}{3} \end{bmatrix}$$

2) For $\mathbf{A} \in \mathbb{R}^{m\times n}$ with $m < n$ and $\operatorname{rank}(\mathbf{A}) = m$, we have

$$\mathbf{A}^T = \mathbf{QR} = \begin{bmatrix} \mathbf{Q}_1 & \mathbf{Q}_2 \end{bmatrix} \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{0} \end{bmatrix} = \mathbf{Q}_1\mathbf{R}_1 + \mathbf{Q}_2\mathbf{0}$$

- note

$$\mathbf{Ax} = \mathbf{R}_1^T\mathbf{Q}_1^T\mathbf{x} + \mathbf{0}^T\mathbf{Q}_2^T\mathbf{x} = \mathbf{b}$$

which indicates

$$\mathbf{Q}_1^T\mathbf{x} = \mathbf{R}_1^{-T}\mathbf{b}$$

and $\mathrm{Q}_2^T\mathrm{x}$ can be anything, which we set to be d. Then we have

$$\begin{bmatrix} \mathbf{Q}_1^T\mathbf{x} \\ \mathbf{Q}_2^T\mathbf{x} \end{bmatrix} = \mathbf{Q}^T\mathbf{x} = \begin{bmatrix} \mathbf{R}_1^{-T}\mathbf{b} \\ \mathbf{d} \end{bmatrix}$$

- the solution is

$$\mathbf{x} = \mathbf{Q}\begin{bmatrix} \mathbf{R}_1^{-T}\mathbf{b} \\ \mathbf{d} \end{bmatrix} = \mathbf{Q}_1\mathbf{R}_1^{-T}\mathbf{b} + \mathbf{Q}_2\mathbf{d}$$

where to get the minimum norm solution, we can set $\mathrm{d} = 0$. so,

$$\mathbf{x} = \mathbf{Q}\begin{bmatrix} \mathbf{R}_1^{-T}\mathbf{b} \\ \mathbf{d} \end{bmatrix} = \mathbf{Q}_1\mathbf{R}_1^{-T}\mathbf{b}$$

$$\because Q_1 = \begin{bmatrix} \frac{1}{3} & 0 & -\frac{2}{3} \\ \frac{2}{3} & -\frac{2}{3} & \frac{1}{3} \\ \frac{2}{3} & \frac{2}{3} & 0 \\ 0 & \frac{1}{3} & \frac{2}{3} \end{bmatrix}, \quad R_1 = \begin{bmatrix} 3 & 0 & 8 \\ 0 & 3 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\therefore x = \begin{bmatrix} \frac{17}{9} & \frac{2}{9} & -\frac{2}{3} \\ -\frac{2}{3} & -\frac{1}{3} & \frac{1}{3} \\ \frac{2}{9} & \frac{2}{9} & 0 \\ -\frac{16}{9} & -\frac{1}{9} & \frac{2}{3} \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} \frac{17}{9}b_1 + \frac{2}{9}b_2 - \frac{2}{3}b_3 \\ -\frac{2}{3}b_1 - \frac{1}{3}b_2 + \frac{1}{3}b_3 \\ \frac{2}{9}b_1 + \frac{2}{9}b_2 \\ -\frac{16}{9}b_1 - \frac{1}{9}b_2 + \frac{2}{3}b_3 \end{bmatrix}$$

VI. Solving LS via Projection

Problem 6. (Bonus question, 6 points + 4 points)

Consider the Least Square (LS) problem:

$$\min_{\mathbf{x}\in\mathbb{R}^n} \|\mathbf{A}\mathbf{x}-\mathbf{y}\|_2^2 \tag{2}$$

where $\mathbf{A} \in \mathbb{R}^{m\times n}$ $(m > n)$ may not be full rank. Denote

$$X_{\text{LS}} = \left\{\mathbf{x} \in \mathbb{R}^n | \mathbf{A}^T\mathbf{A}\mathbf{x} = \mathbf{A}^T\mathbf{y}\right\}$$

as the set of all solutions to (2), and

$$\mathbf{x}_{\text{LS}} = \mathbf{A}^{\dagger}\mathbf{y}$$

where $\mathbf{A}^{\dagger} \in \mathbb{R}^{n\times m}$ is the *pseudo inverse of* $\mathbf{A}$ satisfies the following properties:

1) $\mathbf{A}\mathbf{A}^{\dagger}\mathbf{A} = \mathbf{A}$.
2) $\mathbf{A}^{\dagger}\mathbf{A}\mathbf{A}^{\dagger} = \mathbf{A}^{\dagger}$.
3) $(\mathbf{A}\mathbf{A}^{\dagger})^T = \mathbf{A}\mathbf{A}^{\dagger}$.
4) $(\mathbf{A}^{\dagger}\mathbf{A})^T = \mathbf{A}^{\dagger}\mathbf{A}$.

Answer the following questions:

1) Prove that $\mathbf{x}_{\text{LS}}$ is a solution to (2) and is of minimum 2-norm in X_{LS}, that is

$$\mathbf{x}_{\text{LS}} = \min\{\|\mathbf{x}\|_2^2 |\ \mathbf{x} \in X_{\text{LS}}\}\,.$$

Hint. Notice that the orthogonal projection onto $\mathcal{N}(A)$ is given by

$$\mathbf{\Pi}_{\mathcal{N}(A)} = \mathbf{I} - \mathbf{A}^{\dagger}\mathbf{A}$$

2) Prove that $X_{\text{LS}} = \{\mathbf{x}_{\text{LS}}\}$ if and only if $\text{rank}(\mathbf{A}) = n$.

Solution.

1) $\because X_{LS} = \left\{x \in \mathbb{R}^n \mid A^\top Ax = A^\top y\right\}$

$x_{LS} = A^{\dagger}y \in \mathbb{R}^n$

$A^\top Ax_{LS} = A^\top AA^{\dagger}y = A^\top\left(AA^{\dagger}\right)^\top y = A^\top\left(A^{\dagger}\right)^\top A^\top y = \left(A^{\dagger}A\right)^\top A^\top y = \left(AA^{\dagger}A\right)^\top y = Ay$

$\therefore x_{LS} \in X_{LS}$

$\therefore x_{LS}$ is a solution to (2)

Suppose $\hat{x}$ is a solution of $A^\top Ax = A^\top y$

$i\cdot e\cdot \hat{x} \in X_{LS}$ then $\begin{cases} A^\top A\hat{x} = A^\top y \quad\Rightarrow\quad A^\top A\left(\hat{x}-x_{LS}\right) = 0 \\ A^\top Ax_{LS} = A^\top y \end{cases}$

$\therefore \hat{x} - x_{LS} \in \mathcal{N}\left(A^\top A\right) = N(A)$

$$\begin{aligned}\because \quad \Pi_{N(A)}x_{LS} &= \left(I - A^{\dagger}A\right)x_{LS} = x_{LS} - A^{\dagger}Ax_{LS} = x_{LS} - A^{\dagger}AA^{\dagger}y \\ &= x_{LS} - A^{\dagger}y = 0\end{aligned}$$

$\therefore \quad x_{LS} \in R(A)$

$\therefore \quad x_{LS} \perp (\hat{x} - x_{LS})$

$\therefore \|\hat{x}\|^2 = \|(\hat{x} - x_{LS}) + x_{LS}\|^2 = \|\hat{x} - x_{LS}\|^2 + \|x_{LS}\|^2 \geqslant \|x_{LS}\|^2$

So $\mathrm{x_{LS}} = \arg\min_{\mathrm{x} \in X_{\mathrm{LS}}} \|\mathrm{x}\|_2$

2) If $X_{\mathrm{LS}} = \{\mathrm{x_{LS}}\}$

Then $\mathrm{A}^T \mathrm{A} X = \mathrm{A}^T y$ has a unique solution.

$\therefore \mathrm{A^T A}$ is column full rank.

$\because \mathcal{R}(\mathbf{A}) = \mathcal{R}\left(\mathbf{A^T A}\right)$, then $(\mathbf{A}) = n$

If rank $(\mathbf{A}) = n, then$ $\mathbf{A^T A}$ is invertible, the solution of equation $\mathbf{A}^T \mathbf{A} \mathbf{x} = \mathbf{A}^T y$ is unique, it means that $X_{\mathrm{LS}} = \{\mathrm{x_{LS}}\}$

Above all, $X_{\mathrm{LS}} = \{\mathrm{x_{LS}}\}$ if and only if $\mathrm{rank}(\mathrm{A}) = n$