Christopher Wong
999234204
MAT 128B
Programming Project 3 **3/8/2017**

(a)

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

$\underline{\text{TOL} = 10^{-16}}$

$$Q = \begin{bmatrix} -5.0e-01 & 3.535533905932737e-01 & -3.535533905932737e-01 & 1.543033499620919e-01 \\ 0 & 7.071067811865475e-01 & 1.570092458683775e-16 & 6.172133998483677e-01 \\ 5.0e-01 & 3.535533905932737e-01 & -3.535533905932737e-01 & 1.543033499620919e-01 \\ 0 & 0 & -7.071067811865475e-01 & -6.172133998483675e-01 \\ -5.0e-01 & 3.535533905932737e-01 & 3.535533905932738e-01 & 3.086066999241838e-01 \\ -5.0e-01 & -3.535533905932737e-01 & -3.535533905932738e-01 & -3.086066999241838e-01 \end{bmatrix}$$

$$R_1 = \begin{bmatrix} 2.0e+00 & -4.0e+00 & 2.0e+00 & 1.0e+00 \\ 0 & 2.828427124746190e+00 & 2.828427124746190e+00 & 2.828427124746190e+00 \\ 0 & 0 & 2.828427124746190e+00 & 1.414213562373095e+00 \\ 0 & 0 & 0 & 3.597533769998863e-16 \end{bmatrix}$$

$R_2 =$ **Empty matrix: 4-by-0**

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

$\|Q^TQ - I\|_2 =$ **9.385906354489062e-01**

$\underline{\text{TOL} = 10^{-14}}$

$$Q = \begin{bmatrix} -5.000000000000000e-01 & 3.535533905932737e-01 & -3.535533905932737e-01 \\ 0 & 7.071067811865475e-01 & 1.570092458683775e-16 \\ 5.000000000000000e-01 & 3.535533905932737e-01 & -3.535533905932737e-01 \\ 0 & 0 & -7.071067811865475e-01 \\ -5.000000000000000e-01 & 3.535533905932737e-01 & 3.535533905932738e-01 \\ -5.000000000000000e-01 & -3.535533905932737e-01 & -3.535533905932738e-01 \end{bmatrix}$$

$$R_1 = \begin{bmatrix} 2.000000000000000e+00 & -4.000000000000000e+00 & 2.000000000000000e+00 \\ 0 & 2.828427124746190e+00 & 2.828427124746190e+00 \\ 0 & 0 & 2.828427124746190e+00 \end{bmatrix}$$

$$R_2 = \begin{bmatrix} 1.000000000000000e+00 \\ 2.828427124746190e+00 \\ 1.414213562373095e+00 \end{bmatrix}$$

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

$\|Q^TQ - I\|_2 =$ **4.224272170198187e-16**

TOL = $\mathbf{10^{-12}}$

$$Q = \begin{bmatrix} -5.000000000000000e-01 & 3.535533905932737e-01 & -3.535533905932737e-01 \\ 0 & 7.071067811865475e-01 & 1.570092458683775e-16 \\ 5.000000000000000e-01 & 3.535533905932737e-01 & -3.535533905932737e-01 \\ 0 & 0 & -7.071067811865475e-01 \\ -5.000000000000000e-01 & 3.535533905932737e-01 & 3.535533905932738e-01 \\ -5.000000000000000e-01 & -3.535533905932737e-01 & -3.535533905932738e-01 \end{bmatrix}$$

$$R_1 = \begin{bmatrix} 2.000000000000000e+00 & -4.000000000000000e+00 & 2.000000000000000e+00 \\ 0 & 2.828427124746190e+00 & 2.828427124746190e+00 \\ 0 & 0 & 2.828427124746190e+00 \end{bmatrix}$$

$$R_2 = \begin{bmatrix} 1.000000000000000e+00 \\ 2.828427124746190e+00 \\ 1.414213562373095e+00 \end{bmatrix}$$

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

$\|Q^TQ - I\|_2 =$ **4.224272170198187e-16**

Christopher Wong
999234204
MAT 128B

QR.m

```
% Part (a) QR factorization using Gram-Schmidt process applied to general
% matrices
clear Q R pivot R1 R2 i j k m n i0 normq qtilde r val
format long e

% cd 'C:\Users\Christopher\Desktop\MAT 128B\Project 3';

% Input a matrix A and a tolerance TOL >= 0
A = input('Input the matrix: ');
TOL = input('Input the tolerance: ');

if (TOL < 0)
  error('Tolerance must be zero or positive');
end

[m, n] = size(A);
Q = A;
R = zeros(n, n);
pivot = [1:n];
r = n;

for j = 1:n
  % 1)
  for i = j:n
    normq(i) = norm(Q(:,i), 2);
  end
  % 2)
  if (norm(normq(j:n), 2) <= TOL)
    r = j - 1;
    Q = Q(:,1:r);
    R = R(1:r,:);
    return
  end
  % 3)
  [val, i0] = max(normq((j+1):(n-1)));
  % 4)
  if (i0 > j)
    tmp = Q(:,j);
    Q(:,j) = Q(:,i0);
    Q(:,i0) = tmp;
    tmp = R(1:j-1,j);
    R(1:j-1,j) = R(1:j-1,i0);
    R(1:j-1,i0) = tmp;
    tmp = pivot(j);
    pivot(j) = pivot(i0);
    pivot(i0) = tmp;
```

```
  end
  % 5)
  R(j,j) = norm(Q(:,j), 2);
  % 6)
  qtilde = Q(:,j) / R(j,j);
  % 7)
  Q(:,j) = qtilde;
  % 8)
  for k = (j+1):n
    R(j,k) = qtilde'*Q(:,k);
    Q(:,k) = Q(:,k) - R(j,k)*qtilde;
  end
end

% Not sure why the program terminates before these lines for r < n
R1 = R(:,1:r);
R2 = R(:,(r+1):n);

fprintf('Done. Matrices Q, R (R1 and R2), and pivot vector created\n');
```

(b)

$$b = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

TOL = 10^{-16}

$$x_0 = \begin{bmatrix} -4.289142502257605e+14 \\ -8.578285004515210e+14 \\ -4.289142502257620e+14 \\ 8.578285004515231e+14 \end{bmatrix}$$

$\|b - Ax_0\|_2$ = **1.082531754730548e+00**

TOL = 10^{-14}

$$x_0 = \begin{bmatrix} 9.999999999999997e-01 \\ 1.999999999999999e+00 \\ -4.999999999999998e-01 \\ 0 \end{bmatrix}$$

$\|b - Ax_0\|_2$ = **9.999999999999999e-01**

TOL = 10^{-12}

$$x_0 = \begin{bmatrix} 9.999999999999997e-01 \\ 1.999999999999999e+00 \\ -4.999999999999998e-01 \\ 0 \end{bmatrix}$$

$\|b - Ax_0\|_2$ = **9.999999999999999e-01**

For TOL = 10^{-16}, we see that the entries of x_0 are very large which is alarming considering the entries of b are 1's and the entries of A are small integers. In fact, MATLAB prints an error that A is close to singular which we can see in cases TOL = 10^{-14} or TOL = 10^{-12} where the rank is 3 instead of 4 as in TOL = 10^{-16}. To trust our numerical results, we would use TOL = 10^{-14} or TOL = 10^{-12}.

Christopher Wong
999234204

QR_solve.m

```
% Part (b) solving the least squares problem using Q, R1, pivot

format long e

% cd 'C:\Users\Christopher\Desktop\MAT 128B\Project 3';

% Input matrices Q, R1, and the pivot vector
Q = input('Input the matrix Q: ');
R1 = input('Input the matrix R1: ');
pivot = input('Input the pivot vector: ');

% Setting up the pivot matrix
I = eye(size(pivot, 2));
P = I(:,pivot);

z1 = zeros(size(R1, 2), 1);
z1 = R1\(Q'*b);
z1sol = [z1; zeros(size(pivot, 2) - size(R1, 2), 1)];
x0 = P*z1sol;

fprintf('Done. Solution x0 is given.\n');
```

(c) Prove that the coefficient matrix $M = A^TA$ of the normal equations is symmetric positive definite if rank$A = n$.

Proof. Let $M = A^TA$. We first show that M is symmetric. It is clear that $M^T = (A^TA)^T = A^TA = M$. Thus, M is symmetric.

To show that M is positive definite, we first recall the definition of a positive definite matrix. A matrix M is positive definite iff $x^TMX > 0 \; \forall x \neq 0$. So we can write

$$\begin{aligned} x^TMX &= x^TA^TAx \\ &= (Ax)^T(Ax) \\ &> 0. \end{aligned}$$

The last line comes from the fact that Ax is a vector and its inner (dot) product with itself is always positive. $(Ax)^T(Ax)$ is also equal to the Euclidean norm of Ax squared which is also positive (for $x \neq 0$). $\square$

(d)

Cholesky_normal_equations.m

```
% Part (d)
% Using L and D from M = A'A = LDLT to solve the normal equations.
% Normal equations are A'Ax = A'b where ' denotes transpose.

format long e

% cd 'C:\Users\Christopher\Desktop\MAT 128B\Project 3';

% Input matrices A, L, and D
A = input('Input the matrix A: ');
L = input('Input the matrix L: ');
D = input('Input the matrix D: ');

M = L*D*L';

% Solving the normal equations
% Mx = LDLTx = A'b
x = M\(A'*b);

fprintf('Done. Solution x is given.\n');
```

(e) Using $A \in \mathbb{R}^{17\times 13}$ and $b \in \mathbb{R}^{17}$ given in the Matlab file prog3e.mat.

TOL = 10^{-14}

QR

$$x = \begin{bmatrix} -1.546070222421405e+11 \\ -9.297487747885947e+10 \\ 1.493693105771897e+11 \\ 1.066527222191558e+11 \\ -2.537212627334844e+11 \\ 1.606536022062050e+11 \\ 8.872212574108492e+10 \\ -7.871640990844116e+10 \\ 1.954514525566362e+11 \\ -2.671291704890820e+10 \\ 1.242255913092082e+11 \\ -2.309933234896546e+11 \\ -9.934598662102525e+10 \end{bmatrix}$$

$\|b - Ax\|_2$ = **4.678432159367829e-05**

LDL^T

$$x = \begin{bmatrix} -4.112121011942238e-01 \\ 5.472316353121219e-01 \\ -5.173550705393407e-01 \\ 4.551617465426449e-01 \\ -2.703669723434581e-01 \\ 5.772905380891625e-01 \\ -5.275351589993031e-01 \\ -8.210281428254646e-01 \\ -8.095436455955173e-01 \\ 8.320110605315605e-02 \\ -4.355899005820045e-02 \\ 7.370565021068745e-02 \\ -1.517278149777153e-01 \\ -6.337289071975470e-02 \\ 1.486174812005002e-01 \\ 2.815813274135337e-01 \\ 1.787279076014642e-01 \end{bmatrix}$$

$\|b - Ax\|_2$ = **6.685060541230082e-01**

TOL = 10^{-12}

QR

$$x = \begin{bmatrix} -1.546070222421405e+11 \\ -9.297487747885947e+10 \\ 1.493693105771897e+11 \\ 1.066527222191558e+11 \\ -2.537212627334844e+11 \\ 1.606536022062050e+11 \\ 8.872212574108492e+10 \\ -7.871640990844116e+10 \\ 1.954514525566362e+11 \\ -2.671291704890820e+10 \\ 1.242255913092082e+11 \\ -2.309933234896546e+11 \\ -9.934598662102525e+10 \end{bmatrix}$$

$\|b - Ax\|_2 =$ **4.678432159367829e-05**

LDL^T

$$x = \begin{bmatrix} -1.084212758253467e+07 \\ -5.341898337215079e+07 \\ 5.402236389254467e+06 \\ -3.078563540776220e+07 \\ -7.317060218039244e+06 \\ 7.979441421510610e+07 \\ -4.078790139396015e+07 \\ -7.445778967454656e+05 \\ 4.631027351273581e+07 \\ 6.041363102650469e+06 \\ 3.048194590502472e+05 \\ -2.752119725406089e+07 \\ -1.462576296252692e+07 \end{bmatrix}$$

$\|b - Ax\|_2 =$ **6.685060541230082e-01**

Christopher Wong
999234204

TOL = 10^{-10}

QR

$$x = \begin{bmatrix} -1.279333791949196e+08 \\ 7.989637293852357e+08 \\ 1.788154841354073e+09 \\ 9.606338899267700e+08 \\ 1.092935711880787e+08 \\ -2.566553685570429e+08 \\ -3.070062117655145e+09 \\ 2.373770402763600e+09 \\ -7.070470804156628e+08 \\ -1.541995609142110e+09 \\ -2.129347940478388e+09 \\ 0 \\ 0 \end{bmatrix}$$

$\|b - Ax\|_2$ = **5.565509038236765e-01**

$\underline{LDL^T}$ **Does not exist for this tolerance because at this level, rank(A) = 11 < 13. That is, A is not full rank at TOL = 10^{-10}.**

(f)

Theory 1

$$f_1(t) = c_0 + c_1 t e^{-t} + c_2 t^2 e^{-2t}$$

$$x = \begin{bmatrix} c_0 \\ c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 9.989466497393511e-01 \\ -9.982112073329603e-01 \\ -9.965726661416167e-01 \end{bmatrix}$$

Theory 2

$$f_2(t) = c_0 + c_1 \sqrt{t} e^{-\sqrt{t}} + c_2 t e^{-2\sqrt{t}}$$

$$x = \begin{bmatrix} c_0 \\ c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 5.845428014785350e-01 \\ 4.247465387007817e+00 \\ -1.204915107205322e+01 \end{bmatrix}$$

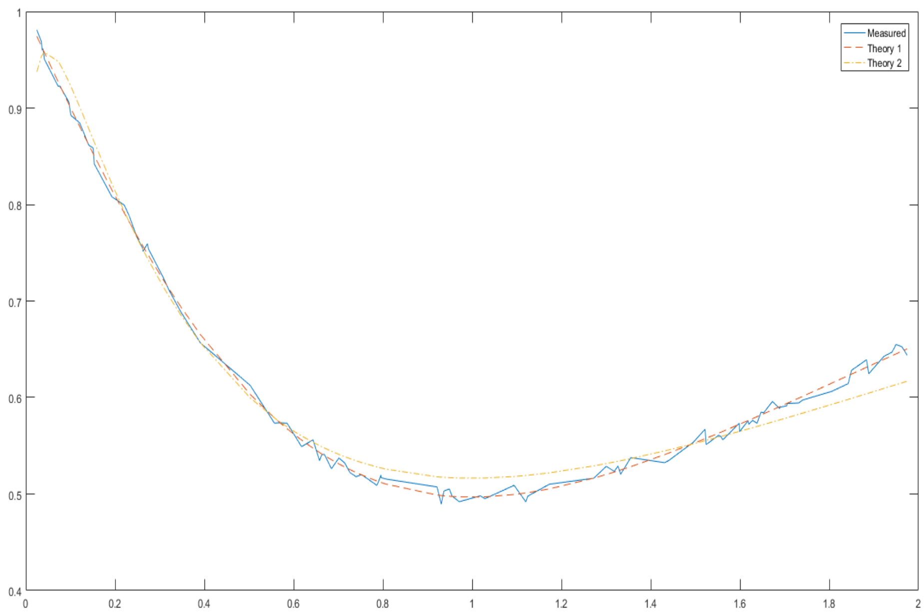

FIGURE 1.

Theory 1 is more appropriate for the given data because its least squares line follows the data better than Theory 2.