# LINEAR ALGEBRA: ANSWER TO HOMEWORK 4

VISHVAS VASUKI

## 1. 8.2

Write a MATLAB function [Q,R] = mgs(A) which returns the reduced QR factorization. Output variables are Q and R.

Code was emailed to the TA.

```
function [Q, R] = QR(A)

[m, n] = size(A);

% Modified gram schmidt
V=A;
for j=[1:n]
    R(j,j)=norm(V(:,j),2);
    Q(:,j)=V(:,j)/R(j,j);
    for i=[j+1:n]
        R(j,i) = Q(:,j)'*V(:,i);
        V(:,i)=V(:,i)-R(j,i)*Q(:,j);
    end
end

end
```

## 2. 10.2

2.1. **a.** A is $m \times n, m \geq n$.

Write [W, R] = house(A). Use Householder reflections which computes the implicit representation of a full QR factorization of A. W is $m \times n$, lower triangular, its columns are the vectors $v_k$ defining the successive Householder reflections. R is a $n \times n$ triangular matrix.

Code was emailed to the TA.

```
% Householder triangularization.
% Gives:
% R upper triangular matrix
% W $m\times n$, lower triangular, its columns are the vectors $v_{k}$
% defining the successive Householder reflections.

function [W, R] = house(A)
[m, n] = size(A);
R = A;
W = zeros(m,n);
```

```
for k = 1:n
    x = R(k:m,k);
    I = eye(m-k+1);
    e_1 = I(:,1);
    sgn = sign(x(1,1));
    if(sgn == 0)
        sgn = 1;
    end
    v = -sgn*norm(x).*e_1 - x;
    v = v./norm(v);
    R(k:m,k:n) = R(k:m,k:n) - 2.*v*v'*R(k:m,k:n);
    W(k:m,k) = W(k:m,k) + v;
end

end
```

2.2. **b.** Write a MATLAB function Q = formQ(W) that takes the matrix W produced by house as input and generates a corresponding $m \times m$ orthogonal matrix Q.

Code was emailed to the TA.

```
% Q = formQ(W) that takes the matrix W produced by house as input and
% generates a corresponding $m \times m$ orthogonal matrix Q.

function Q = formQ(W)
[m, n] = size(W);
Q = eye(m,m);
for k = 1:n
    Q_k = eye(m,m);
    v = W(k:m,k);
    F = eye(m-k+1,m-k+1) - 2*v*v';
    Q_k(k:m,k:m) = F;
    Q = Q*Q_k;
end

end
```

## 3. 11.3

Very long matlab question comparing mgs, qr, house.

Let m = 50, n = 12. Use linspace, to define t = m-vector of linearly spaced grid points from 0 to 1. Use vander and fliplr to define A to be matrix associated with least squares fitting on thi sgrid by polynomial of degree n-1. b = cos(4t). Print to 16 digit precision the least squares coefficient vector x by six methods.

3.1. **a.** Solving normal equations using Matlab's \.

```
 1.000000009828539
-0.000003058556934
-7.999881599423152
-0.001794968011434
10.680787070393082
```

```
-0.065291258825856
-5.500576191037059
-0.344542637686066 :Inaccurate: compare with SVD result.
 2.012711257119547 :Inaccurate: compare with SVD result.
-0.229531584456158 :Inaccurate: compare with SVD result.
-0.275744751507555
 0.070224085955164
```

3.2. **b.** Using QR factorization by mgs.

```
 0.999999996532329
 0.000000389043255
-8.000011138708601
 0.000118975453461
10.666096448792814
 0.001163513676147
-5.689404852048028
 0.001960173714906
 1.602552673895843
 0.072894805227406 :Inaccurate: compare with SVD result.
-0.402066671638750 :Inaccurate: compare with SVD result.
 0.093052068368706
```

3.3. **c.** Using QR factorization by house.

```
 1.000000000996605
-0.000000422742926
-7.999981235688280
-0.000318763197465
10.669430795645004
-0.013820286887481
-5.647075630351267
-0.075316019060198
 1.693606957535560
 0.006032112954680
-0.374241705130035
 0.088040576364041
```

3.4. **d.** Using QR factorization by qr.

```
 1.000000000996609
-0.000000422742968
-7.999981235687960
-0.000318763203417
10.669430795705138
-0.013820287197629
-5.647075629418464
-0.075316020815308
 1.693606959644216
 0.006032111376332
-0.374241704457822
 0.088040576239507
```

3.5. **e.** $x = A\backslash b$ in Matlab.

```
 1.000000000996608
-0.000000422743228
-7.999981235679865
-0.000318763301264
10.669430796332966
-0.013820289609699
-5.647075623539531
-0.075316030120297
 1.693606969166215
 0.006032105309351
-0.374241702274479
 0.088040575901462
```

3.6. **f.** Matlab's svd.

```
 1.000000000996608
-0.000000422742965
-7.999981235688125
-0.000318763200829
10.669430795684868
-0.013820287102762
-5.647075629701436
-0.075316020265539
 1.693606958952933
 0.006032111917519
-0.374241704697128
 0.088040576285090
```

3.7. **g.** The numbers which appear very wrong ($error > 0.1$) are marked above. Normal equations exhibit instability, as does the algorithm which uses mgs (to a lesser extant).

## 4. 10.1

4.1. **a.** Eigenvalues of a Hoseholder reflector F. Give geometric argument as well as algebraic proof.

**Theorem 4.1.1.** *Let F be a $m \times m$ Householder reflector. The eigenvalues of F are +1 and -1.*

*Remark* 4.1.2. Geometric argument: F reflects all vectors in $C^n$ accross a (m-1) dimensional hyperplane, P. For vectors x inside P, F has no effect as $Fx = x$. This eigenvalue has multiplicity n-1, as the corresponding eigenspace is m-1 dimensional. For all vectors passing through the 1 dimensional vector space orthogonal to P, $Fx = -x$. So, -1 is the only other eigenvalue. It has multiplicity 1.

*Proof.* The proof follows from the following lemmata. As 1 is an eigenvalue of $n \times n$ F with multiplicity n-1 and -1 is another eigenvalue with multiplicity atleast 1, there are no other eigenvalues. □

**Lemma 4.1.3.** *1 is an eigenvalue of $F = (I - 2vv^*)x$. If F is $n \times n$, 1 has multiplicity n-1.*

*Proof.* Fx = x whenever $v^*x = 0$, where k is a constant:

(1) $$(I - 2vv^*)x = x - 2vv^*x$$
(2) $$= x \text{ as } v^*x = 0$$
(3)

If F is $n \times n$, the multiples of v or kv can cover only a space of 1 dimension. So, the dimension of the space perpendicular to v is n-1. Hence, the eigenspace is n-1 dimensional, and the multiplicity of the eigenvalue 1 is n-1. ☐

**Lemma 4.1.4.** *-1 is an eigenvector of $F = (I - 2vv^*)x$.*

*Proof.* Fx = -x whenever x = kv, where k is a constant:

(4) $$(I - 2vv^*)kv = kv - 2kvv^*v$$
(5) $$= kv - 2kv$$
(6) $$= -kv$$
(7)

So, -1 is an eigenvalue. ☐

4.2. **b.**

**Theorem 4.2.1.** *The determinant of a m*m Hoseholder reflector F is -1.*

*Proof.* As proved in the previous theorem, the eigenvalues of F are 1 (with multiplicity m-1) and -1. As the determinant of a matrix is equal to the determinant of its Eigenvalue matrix, det(F)=-1. ☐

4.3. **c.**

**Theorem 4.3.1.** *Singular Values of a m*m Hoseholder reflector F are all 1.*

*Remark* 4.3.2. Geometric argument: The Householder reflector maps vectors in the unit sphere to other vectors of equal length in the unit sphere.

*Proof.* Proof by construction of SVD.

$Let F = I - 2vv^*$.

We have shown in an earlier lemma (regarding eigenvalues of F) that Fx = x for all $x \perp v$, and that such x span a n-1 dimensional space. One can use the Gram schmidt method to find the orthogonal basis $F_1$, which is $m \times m$, but has rank (m-1), for such a space.

We have shown in an earlier lemma that Fx= -x when x=kv , where k is a constant. So, we observe that F(-v) = v. We note that -v $\perp$ column space $C(F_1)$.

So, we can construct the matrix $U = [F_1\ (-v)]$, and $V = [F_1\ v]$. Then, we see that FU=VI. Hence, we have the SVD of F, and all singular values of F are 1. ☐

## 5. 10.4

s= sin t, c = cos t for some t.

F:

```
-c s
 s c
```

det(F) = -1. F is a reflector, a special case of a Householder reflector in $C^2$.

J:

```
 c s
-s c
```

det(J) = 1. J is a rotator. Called 'Givens rotation'.

5.1. **a.**

*Remark* 5.1.1. Geometric effects of left multiplication by F on the plane $R^2$: Upon solving the equation Fx = x in order to find the reflecting line, we find $x^*$ = k[$\frac{1-c}{s}$ 1], where k is any scalar constant. Thus, F reflects every vector in $R^2$ accross the space (line) spanned by this vector.

*Remark* 5.1.2. Geometric effects of left multiplications by J on the plane $R^2$: Consider a vector $x^* = [\|x\| \cos A \;\; \|x\| \sin A]$. We see that $(Jx)^* = \|x\| [\cos(A - t) \;\; \sin(A - t)]$. So, J rotates vectors clockwise if t is positive, and counterclockwise if t is negative.

5.2. **b.** Make an algorithm for QR factorization that is analogous to Alg 10 (Triangularizing by introducing zeros), but based on Givens rotations instead of Householder reflections.

*Algorithm* 5.2.1. **findJ**
Input: Vector x in $R^2$.
Output: Givens rotation matrix J which takes x to $\|x\| e_1$.

- Let $c = \frac{x_1}{\|x\|}$.
- $s = \frac{x_2}{\|x\|}$.
- Make matrix J with s and c.

*Algorithm* 5.2.2. **makeR**
for k=1 to n
for i = m-1 to k

- $x = A_{i:i+1,k}$.
- Find the corresponding $J_{k,i}$ using findJ.
- $A_{i:i+1,k:n} = J_{k,i}A_{i:i+1,k:n}$.

5.3. **c.**

**Theorem 5.3.1.** ***makeR*** *involves 6 flops per entry operated on rather than 4; asymptotic operation count is 50% greater than work for householder orthogonalization.*

*Proof.* Take arbitrary $A_{i,j}$, such that it is not in the first row of A. Suppose that the index in the outermost loop is $k < j$. Now, $A_{i,j}$ is involved in two multiplications: $A_{i:i+1,j} = J_{k,i}A_{i:i+1,j}$ and $A_{i-1:i,j} = J_{k,i-1}A_{i:i,j}$. Each of these matrix multiplications involve 2 multiplications and 1 addition per entry. These operations dominate the others when m and n are large. So, asymptotically, 6 flops are required per entry, compared to 4 flops per entry required in Householder orthogonalization. Hence, 50% more work is required.

*Remark* 5.3.2. findJ itself requires 6 flops, but this is compensated by the fact that $A_{i:i+1,k} = J_{k,i}A_{i:i+1,k}$ can be direcly assigned to $(\|A_{i:i+1,k}\|, 0)$, without requiring further calculation.

∎