M A T R I X C O M P U T A T I O N

MATH 544/CPST 531

A. GENZ

WASHINGTON STATE UNVERSITY

FALL 1988

10/10

P1.1-4: Show that if $A = [a_1, \cdots, a_n] \in \mathbb{R}^{m\times n}$ and $B = [b_1, \cdots, b_n] \in \mathbb{R}^{p\times n}$ then

$$AB^T = \sum_{i=1}^{n} a_i b_i^T$$

SHOW 1: $B = [b_1, \cdots, b_n]$, Then

$$B^T = \begin{bmatrix} b_1^T \\ b_2^T \\ \vdots \\ b_n^T \end{bmatrix}$$

$$AB^T = [a_1, \cdots, a_n] \cdot \begin{bmatrix} b_1^T \\ \vdots \\ b_n^T \end{bmatrix} = \sum_{i=1}^{n} a_i b_i^T$$

Q.E.D.

- - - - - - - - - - - - - - - - - - - -

SHOW 2: Let

$$B = [b_1, \cdots, b_n] = \begin{bmatrix} b_{11} & \cdots & b_{1n} \\ \vdots & \ddots & \vdots \\ b_{p1} & \cdots & b_{pn} \end{bmatrix}$$

Then
$$B^T = \begin{bmatrix} b_{11} & \cdots & b_{p1} \\ \vdots & \ddots & \vdots \\ b_{1n} & \cdots & b_{pn} \end{bmatrix}$$

And
$$A = [a_1, \cdots, a_n] = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix}$$

Thus,

$$AB^T = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix} \cdot \begin{bmatrix} b_{11} & \cdots & b_{p1} \\ \vdots & \ddots & \vdots \\ b_{1n} & \cdots & b_{pn} \end{bmatrix}$$

$$= \begin{bmatrix} \sum_{i=1}^{n} a_{1i} b_{1i} & \cdots & \sum_{i=1}^{n} a_{1i} b_{pi} \\ \vdots & \ddots & \vdots \\ \sum_{i=1}^{n} a_{mi} b_{1i} & \cdots & \sum_{i=1}^{n} a_{mi} b_{pi} \end{bmatrix}$$

$$= \sum_{i=1}^{n} \begin{bmatrix} a_{1i} b_{1i} & \cdots & a_{1i} b_{pi} \\ \vdots & \ddots & \vdots \\ a_{mi} b_{1i} & \cdots & a_{mi} b_{pi} \end{bmatrix}$$

$$= \sum_{i=1}^{n} a_i b_i^T$$

Q.E.D.

P1.1-7 Prove the following equation; Suppose $A \in \mathbb{R}^{n\times n}$ is nonsingular and u & $v \in \mathbb{R}^n$.

If $v^T A^{-1} u \neq -1$, then

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

PROOF: Considering

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

$$= AA^{-1} - \frac{AA^{-1}uv^T A^{-1}}{1+v^T A^{-1}u} + uv^T A^{-1} - \frac{uv^T A^{-1}u\,v^T A^{-1}}{1+v^T A^{-1}u}$$

$$= I - \frac{uv^T A^T}{1+v^T A^{-1}u} + uv^T A^{-1} - \frac{(v^T A^{-1}u)\cdot uv^T A^{-1}}{1+v^T A^{-1}u} \quad \cdots (1)$$

The reason why we can rewrite the last term in the eqation above is that $v^T A^{-1} u \rightarrow 1\times n \times n \times n \times n \times 1 = 1\times 1$ is a scalar. Thus

$$\text{eq.(1)} = I - \frac{1+v^T A^{-1}u}{1+v^T A^{-1}u}\cdot uv^T A^{-1} + uv^T A^{-1} = I$$

This implies that

$$(A+uv^T)^{-1} = A^{-1} - \frac{A^{-1}uv^T A^{-1}}{1+v^T A^{-1}u} \qquad \text{Q.E.D.}$$

P1.2-2 Show that if $S \subset \mathbb{R}^m$ is a subspace, then $\dim(S) + \dim(S^\perp) = m$.

SHOW: Let $\{v_1, \cdots, v_m\}$ be the orthogonal basis of $\mathbb{R}^m$ and without loss of generality, assume that $\{v_1, \cdots, v_k\}$ is the orthogonal basis of S. And the only thing we need to prove is $S^\perp = \text{span}\{v_{k+1}, \cdots, v_m\}$

(1) For any $x \in S^\perp$, $x^T \cdot v_i = 0$, $i = 1, \cdots, k$, thus $x \in \text{span}\{v_{k+1}, \cdots, v_m\}$

(2) For any $x \in \text{span}\{v_{k+1}, \cdots, v_m\}$, $x = \sum_{i=k+1}^{m} \alpha_i \cdot v_i$, $x^T = \sum_{i=k+1}^{m} \alpha_i \cdot v_i^T$

And for $y \in S$, $y = \sum_{j=1}^{k} \beta_j \cdot v_j \Rightarrow x^T y = \sum_{i=k+1}^{m} \alpha_i \cdot v_i^T \cdot \sum_{j=1}^{k} \beta_i \cdot v_j = 0$

$\therefore x \in S^\perp$. And this implies that $S^\perp = \text{span}\{v_{k+1}, \cdots, v_m\}$

And $\dim(S) + \dim(S^\perp) = k + (m-k) = m$ Q.E.D.

P.1.3-1 Suppose $A \in \mathbb{R}^{n\times n}$ has positive diagonal entires. Show that if both A and A^T are diagonally dominant then A is positive definite.

SHOW: Firstmost we are going to show that

L1: For a symmtric matrix $B_n \in \mathbb{R}^{n\times n}$ has positive diagonal entires. And B_n is also diagonally dominant, Then $\det(B_n) > 0$

SHOW: STEP 1: FOR $n=1$, $\det(B_1) = b_{11} > 0$

STEP 2: Assume that for $n=m-1$, $\det(B_{m-1}) > 0$

STEP 3: FOR $n=m$, Let

$$B_m = \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1m} \\ b_{12} & b_{22} & \cdots & b_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ b_{1m} & b_{2m} & \cdots & b_{mm} \end{bmatrix}$$

Then without loss of generlity, Assume $b_{11} = \max_i (b_{ii}, i=1,\cdots,m)$

$$\det(B_m) = \det \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1m} \\ b_{12} & b_{22} & \cdots & b_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ b_{1m} & b_{2m} & \cdots & b_{mm} \end{bmatrix}$$

$$= \det \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1m} \\ 0 & b_{22} - \frac{b_{12}}{b_{11}} \cdot b_{12} & \cdots & b_{2m} - \frac{b_{12}}{b_{11}} \cdot b_{1m} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & b_{2m} - \frac{b_{1m}}{b_{11}} \cdot b_{12} & \cdots & b_{mm} - \frac{b_{1m}}{b_{11}} \cdot b_{1m} \end{bmatrix} \quad \cdots (1)$$

$$\triangleq \det \begin{bmatrix} b_{11} & b^T \\ 0 & \bar{B}_{m-1} \end{bmatrix} = b_{11} \cdot \det(\bar{B}_{m-1}) \quad \cdots (2)$$

From (1), it is easy to see that $\bar{B}_{m-1}$ is symmetric.

And

$$\bar{b}_{ii} = b_{i+1\, i+1} - \frac{b_{1\, i+1}^2}{b_{11}} = \frac{1}{b_{11}} \left(b_{11} \cdot b_{i+1\, i+1} - b_{1\, i+1}^2 \right) > 0$$

This implies that $\bar{B}_{m-1}$ has positive diagonal entires.

And for the i th column of $\bar{B}_{m-1}$

$$\bar{b}_{ii} - \sum_{j\neq i} b_{ij} = b_{i+1\, i+1} - \frac{b_{1\, i+1}^2}{b_{11}} - \sum_{\substack{j\neq 1 \\ j\neq i+1}} b_{i+1, j} + \frac{b_{1\, i+1}}{b_{11}} \cdot \sum_{\substack{j\neq 1 \\ j\neq i+1}} b_{1,j}$$

$$= \left(b_{i+1\, i+1} - \sum_{j\neq i+1} b_{i+1, j} \right) + \frac{b_{1\, i+1}}{b_{11}} \left(b_{11} + \sum_{\substack{j\neq 1 \\ j\neq i+1}} b_{1,j} - b_{1\, i+1} \right) > 0$$

Applied the assumption in step 2 to eq. (2), we have

$$\det(B_m) = b_{11} \cdot \det(\bar{B}_{m-1}) > 0.$$

Thus we completed the proof of L1.

Then it is easy to see that all leading minors of $B_n > 0$

This implies that B_n is also a positive definite matrix.

Now, we are ready to show the own statement in P1.3-7. we see that

For any $x \neq 0$ and $x \in \mathbb{R}^n$

$$2x^T A x = x^T A x + x^T A^T x = x^T (A + A^T) x$$

Since A has positive diagonal entires, so does $A+A^T$.

$A+A^T$ is symmetric because $(A+A^T)^T = A+A^T$

and it is easy to show $A+A^T$ is also diagonally dominant if both A & A^T are diagonally dominant.

And then from L1, we know that $A+A^T$ is p.d.

$$x^T(A+A^T)x > 0$$

$$\therefore \quad x^T A x > 0$$

Thus, A is a positive definite matrix. Q.E.D.

P1.3-4 Show that the following subsets of $\mathbb{R}^{n\times n}$ are closed under inversion:

a) positive definite matrices.

Show (a): Assume that A is a p.d. matrix in $\mathbb{R}^{n\times n}$, but A^{-1} is not p.d.

Since A is p.d., that is for any $x \in \mathbb{R}^n$

$$x^TAx > 0 \quad \text{AND} \quad (x^TAx)^T = x^TA^Tx > 0$$

This implies that A^T is a p.d. matrix. ... (1)

However, if A^{-1} is not a p.d. matrix, there exists a non-zero vector $y \in \mathbb{R}^n$, such that

$$y^TA^{-1}y \leq 0$$

but $y^TA^{-1}y = y^TA^{-1}(A^{-1}A)^Ty = (A^{-T}y)^T A^T (A^{-T}y) \leq 0$

This implies that A^T is not a p.d. matrix. ... (2)

(1) and (2) have conflict. Thus A^{-1} is p.d. and subset a) is closed. Q.E.D.

c) unit triangular matrices. (we only prove the upper triangular)

Show (c): Let A is unit upper triangular and

$$A^{-1} = B = \begin{bmatrix} b_{11} & \cdots & b_{1n} \\ \vdots & \ddots & \vdots \\ b_{n1} & \cdots & b_{nn} \end{bmatrix}$$

$$BA = \begin{bmatrix} b_{11} & \cdots & b_{1n} \\ \vdots & \ddots & \vdots \\ b_{n1} & \cdots & b_{nn} \end{bmatrix} \cdot \begin{bmatrix} 1 & \cdots & a_{1n} \\ & \ddots & \vdots \\ 0 & & 1 \end{bmatrix} = \begin{bmatrix} b_{11} & \cdots & \sum_{i=1}^n b_{1i}a_{in} \\ \vdots & & \\ b_{n1} & \cdots & \sum_{i=1}^n b_{ni}\cdot a_{in} \end{bmatrix} = \begin{bmatrix} 1 & & 0 \\ & \ddots & \\ 0 & & 1 \end{bmatrix}$$

Then we have $b_{11} = 1$, $b_{21} = b_{31} = \cdots = b_{n1} = 0$.

We rewrite

$$BA = \begin{bmatrix} 1 & b^T \\ 0 & B_{n-1} \end{bmatrix} \cdot \begin{bmatrix} 1 & a^T \\ 0 & A_{n-1} \end{bmatrix} = \begin{bmatrix} 1 & a^T + b^TA_{n-1} \\ 0 & \underbrace{B_{n-1}A_{n-1}} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & \underbrace{I_{n-1}} \end{bmatrix}$$

Then we have $b_{22} = 1$, $b_{32} = b_{42} = \cdots = b_{n2} = 0$

With the same procedure, we will have a unit upper triangular matrix $B = A^{-1}$. Thus, subset c) is closed. Q.E.D.

P1.4-3: Show that if $A \in \mathbb{C}^{n\times n}$ is Hermitian, then $A = P + iQ$, where

$P^T = P \in \mathbb{R}^{n\times n}$, $-Q^T = Q \in \mathbb{R}^{n\times n}$, and $i^2 = -1$

SHOW: Since A is a complex matrix, we can write

$$A = \begin{bmatrix} p_{11}+i\cdot q_{11} & \cdots & p_{1n}+i\cdot q_{1n} \\ \vdots & \ddots & \vdots \\ p_{n1}+iq_{n1} & \cdots & p_{nn}+i\cdot q_{nn} \end{bmatrix}$$

$$= \begin{bmatrix} p_{11} & \cdots & p_{1n} \\ \vdots & \ddots & \vdots \\ p_{n1} & \cdots & p_{nn} \end{bmatrix} + i\cdot \begin{bmatrix} q_{11} & \cdots & q_{1n} \\ \vdots & \ddots & \vdots \\ q_{n1} & \cdots & q_{nn} \end{bmatrix} = P + i\cdot Q$$

$$A^H = \begin{bmatrix} p_{11}-i\cdot q_{11} & \cdots & p_{n1}-i\cdot q_{n1} \\ \vdots & \ddots & \vdots \\ p_{1n}-iq_{1n} & \cdots & p_{nn}-i\cdot q_{nn} \end{bmatrix}$$

$$= \begin{bmatrix} p_{11} & \cdots & p_{n1} \\ \vdots & \ddots & \vdots \\ p_{1n} & \cdots & p_{nn} \end{bmatrix} - i\cdot \begin{bmatrix} q_{11} & \cdots & q_{n1} \\ \vdots & \ddots & \vdots \\ q_{1n} & \cdots & q_{nn} \end{bmatrix} = P^T - i\cdot Q^T$$

And A is Hermitian,

$$A = A^H = P + iQ = P^T - i\cdot Q^T$$

So, $P = P^T \in \mathbb{R}^{n\times n}$ and $-Q^T = Q \in \mathbb{R}^{n\times n}$. Q.E.D.

P2.1-3 a) Verify that $\|\cdot\|_1$ is vector norm.

PROPERTY 1: $\|x\|_1 = |x_1| + \cdots + |x_n| > 0$ if $x \neq 0$ and $\|x\|_1 = 0$ if $x = 0$ CHK!

PROPERTY 2: $\|x+y\|_1 = |x_1+y_1| + \cdots + |x_n+y_n|$

$\leq |x_1| + |y_1| + \cdots + |x_n| + |y_n|$

$= (|x_1| + \cdots + |x_n|) + (|y_1| + \cdots + |y_n|) = \|x\|_1 + \|y\|_1$ CHECKED!

PROPERTY 3: $\|\alpha x\|_1 = |\alpha x_1| + \cdots + |\alpha x_n|$

$= |\alpha|\cdot|x_1| + \cdots + |\alpha|\cdot|x_n|$

$= |\alpha|\cdot(|x_1| + \cdots + |x_n|) = |\alpha|\cdot\|x\|_1$ CHK!

Thus $\|\cdot\|_1$ is a vector norm. Q.E.D.

P2.2-4 Verify (2.2-9) (Hint: $\|a\|_2 \le \|A\|_2$, where a is any column of A.)

9/9

(2.2-9) $\|A\|_2 \le \|A\|_F \le \sqrt{n} \cdot \|A\|_2$

VERIFICATION: Let a be the p-th column vector of A and vector x

$$x \triangleq [0, \cdots 0, \underset{p\text{-th}}{1}, 0, \cdots, 0]^T \Rightarrow Ax = a \ \& \ \|x\|_2 = 1$$

Then from definition of $\|A\|_2$, we have

$$\|A\|_2 \geq \frac{\|Ax\|_2}{\|x\|_2} = \|a\|_2$$

$$\|A\|_F^2 = \sum_{i=1}^{m}\sum_{j=1}^{n} |a_{ij}|^2 = \sum_{j=1}^{n}\sum_{i=1}^{m} |a_{ij}|^2$$

$$\le \sum_{j=1}^{n} \|a\|_2^2 = n \cdot \|a\|_2^2$$

Then, we have $\|A\|_F \le \sqrt{n} \cdot \|a\|_2 \le \sqrt{n} \cdot \|A\|_2$

From the lecture on 08-31-88, Wednesday, we have

$$\|A\|_2^2 = \sigma_1^2$$

$$\|A\|_F^2 = \sigma_1^2 + \sigma_2^2 + \cdots + \sigma_r^2 \qquad r \geq 1$$

$$\therefore \quad \|A\|_2 \le \|A\|_F$$

Thus, we verify that

$$\|A\|_2 \le \|A\|_F \le \sqrt{n} \cdot \|A\|_2 \qquad \text{Q.E.D.}$$

P2.2-7 b): Verify $\|A\|_\infty = \max_i \sum_{j=1}^{n} |a_{ij}|$

SHOW: According to the Lecture, we have

$$\|A\|_\infty = \sup_{\|x\|_\infty = 1} \|Ax\|_\infty$$

$$\|Ax\|_\infty = \max_i \left| \sum_{j=1}^{n} a_{ij} x_j \right|$$
$$\le \max_i \sum_{j=1}^{n} |a_{ij}| \cdot \|x\|_\infty$$
$$= \max_i \sum_{j=1}^{n} |a_{ij}|$$

$\therefore$ $\|Ax\|_\infty \le \max_i \sum_{j=1}^{n} |a_{ij}|$ for any $\|x\|_\infty = 1$

Let k is the index of the row of A, which has

$$\sum_{j=1}^{n} |a_{kj}| = \max_i \sum_{j=1}^{n} |a_{ij}|$$

And $x = [\operatorname{sign}(a_{k1}), \cdots, \operatorname{sign}(a_{kn})]^T$

where $\operatorname{sign}(a_{kj}) = \begin{cases} 1, & a_{kj} \geq 0 \\ -1, & a_{kj} < 0 \end{cases}$

$\|x\|_\infty = 1$

And

$$\|Ax\|_\infty = \max_i \sum_{j=1}^{n} |a_{ij}| = \sum_{j=1}^{n} |a_{kj}|$$

$\therefore$ $\|A\|_\infty = \max_i \sum_{j=1}^{n} |a_{ij}|$ Q.E.D.

P2.2-8 b) Verify (2.2-15)

$$\frac{1}{\sqrt{m}} \|A\|_1 \leq \|A\|_2 \leq \sqrt{n} \cdot \|A\|_1$$

VERIFICATION: From P2.2-4, we have

$$\|A\|_2 \geq \|a\|_2 \quad \cdots\cdots \quad (1)$$

where a is any column of A. And assume k is the index such that

$$\sum_{i=1}^{m} |a_{ik}| = \max_j \sum_{i=1}^{m} |a_{ij}| = \|A\|_1$$

From (1), we have

$$\|A\|_2^2 \geq \left[\sum_{i=1}^{m} |a_{ik}|^2\right]$$

$$\geq \left[\sum_{i=1}^{m} |a_{ik}|\right]^2 / m \quad \text{(can be proved easily by induction)}$$

$$= \|A\|_1^2 / m$$

$$\therefore \quad \|A\|_2 \geq \frac{1}{\sqrt{m}} \cdot \|A\|_1^2$$

$$\|A\|_F^2 = \sum_{j=1}^{n} \sum_{i=1}^{m} |a_{ij}|^2$$

$$\leq \sum_{j=1}^{n} \left[\sum_{i=1}^{m} |a_{ij}|\right]^2$$

$$\leq \sum_{j=1}^{n} \left[\sum_{i=1}^{m} |a_{ik}|\right]^2$$

$$= n \cdot \|A\|_1^2$$

$$\therefore \quad \|A\|_F \leq \sqrt{n} \cdot \|A\|_1$$

From problem 2.2-4, we have

$$\|A\|_2 \leq \|A\|_F \leq \sqrt{n} \cdot \|A\|_1$$

Thus, we have verified that

$$\frac{1}{\sqrt{m}} \|A\|_1 \leq \|A\|_2 \leq \sqrt{n} \cdot \|A\|_1 \qquad \text{Q.E.D.}$$

P.2.2-9 a) Verify (2.2-16)

$$\|QAZ\|_F = \|A\|_F$$

where Q and Z are orthogonal matrix.

VERIFICATION: Apply the singular value decomposition to A, that is

$\exists$ U And V. such that

$$A = U \Lambda V^T$$

Σ

Then

$$QAZ = QU\Sigma V^T Z = (QU)\Sigma V^T Z$$

we check that

$$(QU)\cdot(QU)^T = QU\cdot U^T\cdot Q^T = I$$

$$(V^T Z)\cdot(V^T Z)^T = V^T Z\cdot Z^T\cdot V = I$$

Thus QU, Σ, $Z^T V$ is also a set of decomposition of QAZ. And both A and QAZ have the same singular values. According to lectures, we know

$$\|A\|_F^2 = \|QAZ\|_F = \sigma_1^2 + \cdots + \sigma_r^2$$

$$\|A\|_F = \|QAZ\|_F \qquad \text{Q.E.D.}$$

P2.3.6. Show that if $A \in \mathbb{R}^{m\times n}$ has rank n, then $\|A(A^TA)^{-1}A^T\|_2 = 1$

SHOW: For such a matrix as A, there exists orthogonal matrices U and V. And

$$A = U\cdot\Sigma\cdot V^T$$

where

$$\Sigma = \begin{bmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_n \\ & \bigcirc & \end{bmatrix}$$

$$A^T\cdot A = V\cdot\Sigma^T\cdot U^TU\Sigma V^T$$

$$= V\cdot\Sigma^T\Sigma\cdot V^T = V\cdot\begin{bmatrix} \sigma_1^2 & & 0 \\ & \ddots & \\ 0 & & \sigma_n^2 \end{bmatrix} V^T$$

$$(A^TA)^{-1} = V\cdot\begin{bmatrix} \sigma_1^{-2} & & 0 \\ & \ddots & \\ 0 & & \sigma_n^{-2} \end{bmatrix} V^T \triangleq V\cdot\Lambda^{-1}\cdot V^T$$

$$B \triangleq A(A^TA)^{-1}A^T = U\Sigma V^T\cdot V\cdot\Lambda^{-1}\cdot V^T\cdot V\cdot\Sigma^T U^T$$

$$= U\Sigma\cdot\Lambda^{-1}\cdot\Sigma^T U^T$$

$$= U\cdot\begin{bmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_n \\ & \bigcirc & \end{bmatrix}\cdot\begin{bmatrix} \sigma_1^{-2} & & 0 \\ & \ddots & \\ 0 & & \sigma_n^{-2} \end{bmatrix}\cdot\begin{bmatrix} \sigma_1 & & 0 & \\ & \ddots & & 0 \\ 0 & & \sigma_n & \end{bmatrix} U^T$$

$$= U\cdot\begin{bmatrix} \sigma_1^{-1} & & 0 \\ & \ddots & \\ 0 & & \sigma_n^{-1} \\ & \bigcirc & \end{bmatrix}\cdot\begin{bmatrix} \sigma_1 & & 0 & \\ & \ddots & & 0 \\ 0 & & \sigma_n & \end{bmatrix} U^T$$

$$= U\cdot\begin{bmatrix} 1 & & 0 & \\ & \ddots & & 0 \\ 0 & & 1 & \\ & \bigcirc & & \end{bmatrix}\cdot U^T$$

$$= U\cdot\begin{bmatrix} 1 & & & & 0 \\ & \ddots & & & \\ & & 1 & & \\ & & & 0 & \\ 0 & & & & \ddots \; 0 \end{bmatrix}\cdot U^T$$

We see that the equation above is a set of singular value decomposition of $A(A^TA)^{-1}A^T \triangleq B$ and $\sigma_{1,B} = 1$

$\|A(A^TA)^{-1}A^T\|_2 = 1$.

Q.E.D.

P2.3.7 Use the SVD to establish the fundamental identity rank(A) = rank(A^T)

SHOW: From SVD, we have

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

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

$$\text{rank}(A) = \text{rank}(\Sigma)$$

And $\text{rank}(A^T) = \text{rank}(\Sigma^T)$

Assume

$$\Sigma = \begin{bmatrix} \sigma_1 & & 0 \\ 0 & \ddots & \sigma_n \\ & 00 & \end{bmatrix}$$

$$\Sigma^T = \begin{bmatrix} \sigma_1 & & 0 & 0 \\ & \ddots & & 0 \\ 0 & & \sigma_n & 0 \end{bmatrix}$$

It is easy to see that both Σ and Σ^T have the same ranks. Q.E.D.

P.2.4.1 Show that if P is an orthogonal projection, then $Q = I - 2P$ is orthogonal.

SHOW:
$$\begin{aligned} Q^T Q &= (I-2P)^T (I-2P) \\ &= (I-2P^T)(I-2P) \\ &= (I-2P)(I-2P) \\ &= I - 2P - 2P + 4P^2 \\ &= I - 4P + 4P \\ &= I \end{aligned}$$

Thus, we have shown that Q is orthogonal. Q.E.D.

Homework no: 2 matrix computations Ben M. Chen 07

P.2.5-1. Show that if $\|I\| \geq 1$, then $\kappa(A) \geq 1$

SHOW: According the difination of $\kappa(A)$

$$K(A) = \|A^{-1}\| \cdot \|A\| \geq \|A^{-1} \cdot A\| = \|I\| \geq 1 \qquad \text{Q.E.D.}$$

Additional Problem: Use digits (in order) from my student ID to form a 4×2 matrix (by rows). Use the LINPACK subroutine SSVDC to compute the singular value decomposition for this matrix. Print the resulting U and V matrices, the singular values and your condition number.

```
C        HOMEWORK NO:2          MATH 544 COMP LIN AL                BEN
C        PROGRAM FOR COMPUTING THE SINGULAR VALUE DECOMPOSITION OF MY
C
         REAL COND
         DIMENSION X(4,2),U(4,4),V(2,2),S(2),E(4),QRAUX(4),WORK(4)
         LDX=4
         LDU=4
         LDV=2
         DATA X / 6., 8., 3., 8., 0., 8., 8., 3./
         WRITE(6,*) 'FOLLOWING IS THE 4*2 MATRIX FROM MY ID NUMBER:'
         DO 05 I=1,4
         WRITE(6,*) (X(I,J),J=1,2)
05       CONTINUE
         CALL SSVDC(X,LDX,4,2,S,E,U,LDU,V,LDV,WORK,11,INFO)
         PRINT *,' '
         WRITE(6,*) 'LEFT DECOMPOSITION MATRIX  U='
         DO 10 I=1,4
         WRITE(6,*) (U(I,J),J=1,4)
10       CONTINUE
         PRINT *,' '
         WRITE(6,*) 'RIGHT DECOMPOSITION MATRIX V='
         DO 20 I=1,2
         WRITE(6,*)(V(I,J),J=1,2)
20       CONTINUE
         PRINT *,' '
         WRITE (6,*) 'SINGULAR VALUES OF MY ID MATRIX ARE:'
         WRITE (6,*) S(1), S(2)
         COND=S(1)/S(2)
         PRINT *, ' '
         WRITE(6,*)'THE CONDITION NUMBER IS:'
         WRITE(6,*) COND
         STOP
         END
```

```
FOLLOWING IS THE 4*2 MATRIX FROM MY ID NUMBER:
6.0 0.0
8.0 8.0
3.0 8.0
8.0 3.0

LEFT DECOMPOSITION MATRIX  U=
-.278738 .603621 .355453 -.656961
-.688341 -.139675 -.681629 -.205082
-.456059 -.642693 .61206 -6.58547E-02
-.490409 .45064 .185516 .722499

RIGHT DECOMPOSITION MATRIX V=
-.761143 .648583
-.648583 -.761143

SINGULAR VALUES OF MY ID MATRIX ARE:
16.384 6.44692

THE CONDITION NUMBER IS:
2.54137
```

Homework No: 3 MATH 544 COMP. LIN. AL. Ben M. Chen 02 11.0/11

P3.1-3: Suppose $X \in R^{n\times p}$ and $A \in R^{n\times n}$, with A symmetric. Specify an algorithm for computing Y=X'AX which requires pn(n+p/2) flops. Do not overwrite A with Y.

SOLUTION: Let A = ◹ + ◺ ,where ◹ = ◺^{T}

And then X'AX = X'(◹+◺)X = X'◹X + (X'◹X)'

Algorithm for P3.1-3

```
for i=1 to n                                   } n flops
    a_ii := a_ii/2

for i=1 to p
    for j=1 to p
        y_ij := 0

for i=1 to p                                   ]
    for j=1 to n                               ]
        s := 0                                 ] pn·(n/2) + O(pn) flops
        for k=1 to j                           ]   (pnp flops for the whole i, j loop)
            s := s + x_ki * a_kj               ]
        for k=1 to p
            y_ik := y_ik + s * x_jk

for i=1 to p                                   }
    for j=i to p                               } O(p²) flops
        y_ij := y_ij + y_ji                    }
        y_ji := y_ij                           }
```

Total flops = pn(n/2 + p) + O(pn) flops.

This is a slightly different result, but it is better for large n.

P3.2-5 Show through a 2-by-2 example that matrix multiplication in finite precision arithmetic need not have have small relative error

Show: Assuming that the unit roundoff in a particular computer is $u = 10^{-5}$; $\beta = 10$ and then $t = 6$

And matrices

$$A = \begin{bmatrix} 1.234564 & 2.469128 \\ 2.469128 & 4.938254 \end{bmatrix}$$

$$B = \begin{bmatrix} 4.444444 & -2.222223 \\ -2.222223 & 1.111111 \end{bmatrix}$$

$$A \cdot B = \begin{bmatrix} -2.47 \times 10^{-6} & 1.235 \times 10^{-6} \\ -5 \times 10^{-7} & -4.692 \times 10^{-6} \end{bmatrix}$$ This computed by using $t=10$

$$fl(A \cdot B) = \begin{bmatrix} -2 \times 10^{-5} & 2 \times 10^{-5} \\ 0 & -1 \times 10^{-5} \end{bmatrix}$$ This is computed by using $t = 6$

$$A \cdot B - fl(A \cdot B) = \begin{bmatrix} 1.753 \times 10^{-5} & 1.8765 \times 10^{-5} \\ -5 \times 10^{-7} & 5.308 \times 10^{-6} \end{bmatrix}$$

$$\text{Relative error}(AB) = \begin{bmatrix} 7.09717 & 15.19433 \\ 1 & 1.13129 \end{bmatrix}$$

We see that the relative error in the first row and second column is about 1519 %.

P3.3-1. Execute Algorithm 3.3-1 with $x = (1, 7, 2, 3, -1)^T$, $k = 1$, $j = 5$

Solution:

$m := \max\{1, 7, 2, 3, 1\} = 7$

$\alpha := 0$

For $i = 1$ to 5

[for = 1 ; $V_1 = 1/7 = 0.142857$, $\alpha = 0.020408$

for = 2 ; $V_2 = 1$, $\alpha = 1.020408$

for = 3 ; $V_3 = 2/7 = 0.285714$, $\alpha = 1.102041$

for = 4 ; $V_4 = 3/7 = 0.428571$, $\alpha = 1.285714$

for = 5 ; $V_5 = -1/7 = -0.142857$, $\alpha = 1.306122$]

$\alpha = 1.142857$

$\beta = 1/(\alpha(\alpha + |V_1|)) = 0.680556$

$V_1 = 0.142857 + 1.142857 = 1.285714$

Check:

$$Px = x - \beta v v^T x = \begin{bmatrix} 1 \\ 7 \\ 2 \\ 3 \\ -1 \end{bmatrix} - 0.680556 \begin{bmatrix} 1.285714 \\ 1.00000 \\ 0.285714 \\ 0.428571 \\ -0.142857 \end{bmatrix} \cdot$$

$$\cdot [1.285714 , 1.000000, 0.285714, 0.428571, -.142857] \cdot \begin{bmatrix} 1 \\ 7 \\ 2 \\ 3 \\ -1 \end{bmatrix}$$

$$= \begin{bmatrix} -8.000002 \\ -0.000003 \\ 0.000001 \\ 0.000002 \\ -0.000001 \end{bmatrix}$$

P3.3-4: Give an algorithm for the update $A := AP$, where $A \in R^{n\times q}$, $P \in \mathbb{R}^{q\times q}$, and P is a Householder matrix.

Solution: Since P is a Householder matrix. $\exists\, V \in \mathbb{R}^{q\times 1}$

$$P = I - 2VV^T/V^TV$$

$$\frac{VV^T}{V^TV} = \begin{pmatrix} \frac{V_1^2}{V^TV} & & \text{else} \\ & \ddots & \\ \text{else} & & \frac{V_q^2}{V^TV} \end{pmatrix} = \frac{1}{2}(I-P)$$

Let $V^TV = 1$, we have

$$\begin{bmatrix} V_1^2 & V_1V_2 & \cdots & V_1V_q \\ V_2V_1 & V_2^2 & \cdots & V_2V_q \\ \vdots & \vdots & \ddots & \vdots \\ V_qV_1 & V_qV_2 & \cdots & V_q^2 \end{bmatrix} = \frac{1}{2}(I-P) \triangleq \begin{bmatrix} (1-P_{11})/2 & -P_{12}/2 & \cdots & -P_{1q}/2 \\ -P_{21}/2 & (1-P_{22})/2 & \cdots & -P_{2q}/2 \\ \vdots & & \ddots & \vdots \\ -P_{q1}/2 & \cdots & \cdots & (1-P_{qq})/2 \end{bmatrix}$$

Let $\sqrt{2}V_1 = \sqrt{(1-P_{11})}$,

$\sqrt{2}V_i = -P_{1i}/(\sqrt{2}V_1)$, $i = 2, \cdots, q$

Then

$$AP = A(I - 2VV^T)$$

$$= A - 2AVV^T = A - A(\sqrt{2}V)(\sqrt{2}V)^T$$

Algorithm: $P_{q1} := \sqrt{(1-P_{11})}$

For $i = 2, \cdots, q$

$P_{qi} := -P_{1i}/P_{q1}$

For $i = 1, \cdots, n$

$S := 0$

For $j = 1, \cdots, q$

$S := S + a_{ij} * P_{qj}$

For $j = 1, \cdots, q$

$a_{ij} := a_{ij} - S * P_{qj}$

Requires $2nq + O(n)$ flops for this algorithm.

P3.4-2 Suppose x and y are unit vectors in R^n. Give an algorithm using Givens transformations which computes an orthogonal Q such that $Qx = y$

Solution: Recall $y = J(i,k,\theta)x$

$$y_i = cx_i + sx_k \Rightarrow x_i y_i = cx_i^2 + sx_i x_k$$

$$y_k = -sx_i + cx_k \Rightarrow x_k y_k = -sx_i x_k + cx_k^2$$

$$c(x_i^2 + x_k^2) = x_i y_i + x_k y_k$$

$$c = (x_i y_i + x_k y_k)/(x_i^2 + x_k^2)$$

$$s = (x_k y_i - x_i y_k)/(x_i^2 + x_k^2)$$

$$c^2 + s^2 = (y_i^2 + y_k^2)/(x_i^2 + x_k^2) = 1$$

$\therefore$ we pick $y_i = \pm\sqrt{x_i^2 + x_k^2 - y_k^2}$ ... (1)

In order to obtain a reasonable y_i, we must choose

$x_i^2 + x_k^2 - y_k^2 \geq 0$ .... (2)

So, our algorithm begins with the match of the smallest $|y_i|$, $i = 1, \cdots, n$ $\triangleq y_k$, then pick a x_i such that the condition in (2) is held.

Good

Assuming a Given's transformation is given as below:

$$Q' = \begin{bmatrix} 1 & & & & & & 0 \\ & \ddots & & & & & \\ & & c & \cdots & s & & \\ & & \vdots & 1 & \vdots & & \\ & & -s & \cdots & c & & \\ & & & & & \ddots & \\ 0 & & & & & & 1 \end{bmatrix} = \begin{array}{c} \begin{array}{ccccc} & p & & k & \end{array} \\ \begin{bmatrix} I & 0 & 0 & 0 & 0 \\ 0 & c & 0 & s & 0 \\ 0 & 0 & I & 0 & 0 \\ 0 & -s & 0 & c & 0 \\ 0 & 0 & 0 & 0 & I \end{bmatrix} \end{array} \begin{array}{l} \\ \cdots i \\ \\ \cdots k \\ \\ \end{array} \quad (p)$$

$$Q = \begin{bmatrix} Q_{11} & Q_{12} & Q_{13} & Q_{14} & Q_{15} \\ Q_{21} & Q_{22} & Q_{23} & Q_{24} & Q_{25} \\ Q_{31} & Q_{32} & Q_{33} & Q_{34} & Q_{35} \\ Q_{41} & Q_{42} & Q_{43} & Q_{44} & Q_{45} \\ Q_{51} & Q_{52} & Q_{53} & Q_{54} & Q_{55} \end{bmatrix}, \quad Q'Q = \begin{bmatrix} Q_{11} & Q_{12} & Q_{13} & Q_{14} & Q_{15} \\ \times & \times & \times & \times & \times \\ Q_{31} & Q_{32} & Q_{33} & Q_{34} & Q_{35} \\ \times & \times & \times & \times & \times \\ Q_{51} & Q_{52} & Q_{53} & Q_{54} & Q_{55} \end{bmatrix} \begin{array}{l} \\ \cdots i \\ \\ \cdots k \\ \\ \end{array}$$

We see that we need only to compute two new rows.

To be continued on →

P3.4-2 (CONT.)

```
Algorithm:   for i=1 to n
                for j=1 to n
                   if i=j then  q_ij :=1  else  q_ij :=0

      BEN:   m:=0
             for i=1 to n
                if  x_i = y_i  then  m:=m+1
             if m=n then  stop the algorithm.
                else  a:=1
                      for i=1 to n
                         if  x_i ≠ y_i  and  |y_i|<a
                             then  a:=|y_i|
                                   k:=i
                      for i=1,2,...,k-1,k+1,...,n
                         if  x_i ≠ y_i
                             then  b:= x_i^2 + x_K^2 - y_K^2        3 flops
                                   if  b≥0
                                      then  p:=i
                                            goto  CHEN

     CHEN:   r:=sign(y_p)*√b
             c:=(x_p * y_p + x_K * r)/(x_p^2 + x_K^2 )
             s:=(x_K * r - x_p * y_K )/(x_p^2 + x_K^2 )
             x_K:= y_K
             x_p:= r
             for i=1 to n
                h_i:= q_pi * c + q_Ki * s     } 4 flops
                z_i:= q_Ki * c - q_pi * s
             for i=1 to n
                q_pi:= h_i
                q_Ki:= z_i
             goto BEN
```

Example: Following example is computed based on the algorithm above. $n = 5$, X and Y are given as

$$X = \begin{bmatrix} -0.231453 \\ 0.015694 \\ -0.346730 \\ 0.236744 \\ 0.877447 \end{bmatrix} \qquad Y = \begin{bmatrix} 0.345678 \\ -0.700721 \\ 0.000123 \\ -0.014562 \\ 0.623927 \end{bmatrix}$$

The worst case in this algorithm requires $7n^2 + O(n)$ flops.

SOLUTION TO EXAMPLE:

$$X = \begin{bmatrix} -0.231453 \\ 0.015694 \\ -0.346730 \\ 0.236744 \\ 0.877447 \end{bmatrix} \qquad Y = \begin{bmatrix} 0.345678 \\ -0.700721 \\ 0.000123 \\ -0.014562 \\ 0.623927 \end{bmatrix}$$

$$\begin{bmatrix} -0.555443 & 0.000000 & -0.831554 & 0.000000 & 0.000000 \\ 0.000000 & 1.000000 & 0.000000 & 0.000000 & 0.000000 \\ 0.831554 & 0.000000 & -0.555443 & 0.000000 & 0.000000 \\ 0.000000 & 0.000000 & 0.000000 & 1.000000 & 0.000000 \\ 0.000000 & 0.000000 & 0.000000 & 0.000000 & 1.000000 \end{bmatrix} * \begin{bmatrix} -0.231453 \\ 0.015694 \\ -0.346730 \\ 0.236744 \\ 0.877447 \end{bmatrix} = \begin{bmatrix} 0.416884 \\ 0.015694 \\ 0.000123 \\ 0.236744 \\ 0.877447 \end{bmatrix}$$

$$\begin{bmatrix} -0.474440 & 0.000000 & -0.710285 & 0.520002 & 0.000000 \\ 0.000000 & 1.000000 & 0.000000 & 0.000000 & 0.000000 \\ 0.831554 & 0.000000 & -0.555443 & 0.000000 & 0.000000 \\ 0.288832 & 0.000000 & 0.432410 & 0.854165 & 0.000000 \\ 0.000000 & 0.000000 & 0.000000 & 0.000000 & 1.000000 \end{bmatrix} * \begin{bmatrix} -0.231453 \\ 0.015694 \\ -0.346730 \\ 0.236744 \\ 0.877447 \end{bmatrix} = \begin{bmatrix} 0.479195 \\ 0.015694 \\ 0.000123 \\ -0.014562 \\ 0.877447 \end{bmatrix}$$

$$\begin{bmatrix} -0.331120 & 0.716179 & -0.495720 & 0.362918 & 0.000000 \\ 0.339784 & 0.697917 & 0.508691 & -0.372414 & 0.000000 \\ 0.831554 & 0.000000 & -0.555443 & 0.000000 & 0.000000 \\ 0.288832 & 0.000000 & 0.432410 & 0.854165 & 0.000000 \\ 0.000000 & 0.000000 & 0.000000 & 0.000000 & 1.000000 \end{bmatrix} * \begin{bmatrix} -0.231453 \\ 0.015694 \\ -0.346730 \\ 0.236744 \\ 0.877447 \end{bmatrix} = \begin{bmatrix} 0.345678 \\ -0.332236 \\ 0.000123 \\ -0.014562 \\ 0.877447 \end{bmatrix}$$

$$\underbrace{\begin{bmatrix} -0.331120 & 0.716179 & -0.495720 & 0.362918 & 0.000000 \\ 0.301175 & 0.618613 & 0.450889 & -0.330097 & -0.462975 \\ 0.831554 & 0.000000 & -0.555443 & 0.000000 & 0.000000 \\ 0.288832 & 0.000000 & 0.432410 & 0.854165 & 0.000000 \\ 0.157312 & 0.323118 & 0.235511 & -0.172419 & 0.886371 \end{bmatrix}}_{Q} * \underbrace{\begin{bmatrix} -0.231453 \\ 0.015694 \\ -0.346730 \\ 0.236744 \\ 0.877447 \end{bmatrix}}_{X} = \underbrace{\begin{bmatrix} 0.345678 \\ -0.700721 \\ 0.000123 \\ -0.014562 \\ 0.623927 \end{bmatrix}}_{Y}$$

V. good: extra credit +1

P3.5-3 Determine a 3-by-3 Gauss transformation M such that

$$M \begin{bmatrix} 2 \\ 3 \\ 4 \end{bmatrix} = \begin{bmatrix} 2 \\ 7 \\ 8 \end{bmatrix}$$

Solution: We see that the first elemente in both vectors are the same. So, we do not need to apply some Xformation to them. Then for the second elements, we assume

$$M \begin{bmatrix} 2 \\ 3 \\ 4 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ -\alpha_1 & 1 & 0 \\ -\alpha_2 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 2 \\ 3 \\ 4 \end{bmatrix} = \begin{bmatrix} 2 \\ 3-2\alpha_1 \\ 4-2\alpha_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 7 \\ 8 \end{bmatrix}$$

$$\Rightarrow \alpha_1 = -2 \qquad \alpha_2 = -2 \qquad \Rightarrow \alpha = (0, -2, -2)^T$$

we have Gauss transformation

$$M = \begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 2 & 0 & 1 \end{bmatrix}$$

$$= I - \begin{bmatrix} 0 \\ -2 \\ -2 \end{bmatrix} \cdot (1, 0, 0)$$

$$= I - \alpha \cdot e_1^T$$

Note: the definition of Gauss transformation in text is a matrix which has the form $I - \alpha e_k^T$. But it transforms all elements to 0 when the indices are less than k. Here, the M transforms the elements to nonzeros.

P4.1-2 Interchange the order of the loops in Algorithms 4.1-1 and 4.1-2.

Algorithm 4.1-1:

For $i = n, n-1, \dots, 1$

$y_i := b_i$

For $j = i-1, \dots, 2, 1$

$p := -l_{ij} / l_{jj}$

This should be outside this loop

$y_i := y_i - b_j * p$

For $k = j-1, \dots, 2, 1$

$l_{ik} := l_{ik} + l_{jk} * p$

This is not necessary

$y_i := y_i / l_{ii}$

This algorithm requires $\frac{n^3}{6} + O(n^2)$ flops.

-1

Algorithm 4.1-2:

For $i = 1, 2, \dots, n$

$x_i := y_i$

For $j = n, \dots, i+1$

$p := -u_{ij} / u_{jj}$

$x_i := x_i - b_j * p$

For $k = n$ to $j+1$

$u_{ik} := u_{ik} + u_{jk} * p$.

$x_i := x_i / u_{ii}$

This algorithm requires $\frac{1}{6} n^3 + O(n^2)$ flops.

Algorithms should use $\frac{n^2}{2}$ flops

P4.2-3 Suppose

$$A = \begin{array}{cc} \left[\begin{array}{cc} A_{11} & A_{12} \\ A_{21} & A_{22} \end{array}\right] & \begin{array}{l} k \\ m-k \end{array} \\ \begin{array}{cc} k \quad & n-k \end{array} & \end{array}$$

and that A_{11} is nonsingular. The matrix

$$S = A_{22} - A_{21} A_{11}^{-1} A_{12}$$

is called the schur complement of A_{11} in A. Show that if A_{11} has an L-U decomposition, then after k steps of Gaussian elimination S equals the matrix $A_{22}^{(k)}$ in (4.2,-1)

SHOW: Because A_{11} has an L-U decomposition and A_{11} is an k-by-k matrix. This implies that we can performance at lease k Gaussian elimination to matrix A. And at the k-the Gaussian elimination.

$A_{11}^{(k)}$ will be upper triangular matrix.

$A_{21}^{(k)}$ will be 'all zeros' matrix

Then

$$S^{(k)} = A_{22}^{(k)} - A_{21}^{(k)} (A_{11}^{(k)})^{-1} A_{12}^{(k)}$$

$$= A_{22}^{(k)} - 0 \cdot (A_{11}^{(k)})^{-1} A_{12}^{(k)}$$

$$= A_{22}^{(k)}$$

where $A_{22}^{(k)}$ is exactlly the same as that in (4.2,-1).

Q.E.D.

P4.2-8 Matrices in $\mathbb{R}^{n\times n}$ of the form $N(y,k) = I + ye_k^T$ where $y \in \mathbb{R}^n$ are said to be Gauss-Jordan transformations. (a) Give a formula for $N(y,k)^{-1}$ assuming it exists. (b) Given $x \in \mathbb{R}^n$, under what conditions can y be found so $N(y,k)x = e_k$? (c) Give an algorithm using Gauss-Jordan transformations that overwrites A with A^{-1}. What conditions on A ensure the success of your algorithm

---

Solution: (a) In order to get $N(y,k)^{-1}$, we must assume that $y_k \neq -1$. Otherwise, the rank of $I + ye_k^T$ drops to $n-1$. Under this assumption $N(y,k)^{-1}$ is given as

$$N(y,k)^{-1} = I - ye_k^T/(1+y_k)$$

We check that

$$(I + ye_k^T)[I - ye_k^T/(1+y_k)]$$

$$= I - ye_k^T/(1+y_k) + ye_k^T - y\underbrace{e_k^T y}_{y_k}e_k^T/(1+y_k)$$

$$= I + \frac{y_k\, ye_k^T}{1+y_k} - \frac{y_k\, ye_k^T}{1+y_k}$$

$$= I$$

CHECKED!

(b) y can be found to satisfy $N(y,k)x = e_k$ if and on if $x_k \neq 0$

PROOF:

$$N(y,k)x = (I + ye_k^T)x$$

$$= x + ye_k^T x$$

$$= x + y\cdot x_k = e_k$$

$$y \cdot x_k = e_k - x$$

$$y = (e_k - x)/x_k \quad \text{if } x_k \neq 0$$

And if y exists, x_k must be nonzero.

(to be contined.)

P4.2-8 (CONT.)

(c) Let

$$A = [a_1, a_2, \cdots, a_n] = A^{(1)}$$

From (b) part, we see that if $a_{11} \neq 0$, there exists

$$y^{(1)} = (e_1 - a_1)/a_{11}$$

such that

$$N(y^{(1)}, 1) \cdot A = [e_1, a_2^{(2)}, \cdots, a_n^{(2)}] = A^{(2)}$$

then we see that we can overwrite $y^{(1)}$ in e_1 positions, and

$$N(y, 2) \cdot e_1 = e_1 + y \underbrace{e_2^T e_1}_{0} = e_1$$

So, we can apply another Gauss-Jordon transformations to $A^{(2)}$ if $a_{22}^{(2)} \neq 0$.

And then so on, we will be able to have

$$N(y^{(n)}, n) \cdot N(y^{(n-1)}, n-1), \cdots, N(y^{(1)}, 1) \cdot A = [e_1, e_2, \cdots, e_n] = I$$

Then

$$A^{-1} = N(y^{(n)}, n) \cdot N(y^{(n-1)}, n-1) \cdots N(y^{(1)}, 1)$$
$$= (I + y^{(n)} e_n^T) \cdot (I + y^{(n-1)} e_{n-1}^T) \cdots (I + y^{(1)} e_1^T)$$

And

$$N(y^{(n)}, n) N(y^{(n-1)}, n-1) = \begin{bmatrix} 1, & 0, & \cdots, & 0, & y_1^{(n)} \\ 0, & 1, & \cdots, & 0, & y_2^{(n)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & y_{n-1}^{(n)} \\ 0 & 0 & \cdots & 0 & 1 + y_n^{(n)} \end{bmatrix}$$

$$\cdot \begin{bmatrix} 1, & 0, & \cdots, & y_1^{(n-1)}, & 0 \\ 0, & 1, & \cdots, & y_2^{(n-1)}, & 0 \\ \vdots & \vdots & \ddots & \vdots & \\ 0 & 0 & \cdots & 1 + y_{n-1}^{(n-1)} & 0 \\ 0 & 0 & \cdots & y_n^{(n-1)} & 1 \end{bmatrix} = \begin{bmatrix} 1, & 0, & \cdots, & y_1^{(n-1)} + y_1^{(n)} y_n^{(n-1)} & y_1^{(n)} \\ 0, & 1, & \cdots, & y_2^{(n-1)} + y_2^{(n)} y_n^{(n-1)} & y_2^{(n)} \\ \vdots & & & & \vdots \\ 0, & 0, & \cdots, & 1 + y_{n-1}^{(n-1)} + y_{n-1}^{(n)} y_n^{(n-1)} & y_{n-1}^{(n)} \\ 0, & 0, & \cdots, & y_n^{(n-1)} + y_n^{(n)} y_n^{(n-1)} & 1 + y_n^{(n)} \end{bmatrix}$$

$$\triangleq \begin{bmatrix} 1, & 0, & \cdots, & \bar{y}_1^{(n-1)} & \bar{y}_1^{(n)} \\ 0, & 1, & \cdots, & \bar{y}_2^{(n-1)} & \bar{y}_2^{(n)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0, & 0, & \cdots, & 1 + \bar{y}_{n-1}^{(n-1)} & \bar{y}_{n-1}^{(n)} \\ 0, & 0, & \cdots, & \bar{y}_n^{(n-1)} & 1 + \bar{y}_n^{(n)} \end{bmatrix}$$

(to be continued)

P4.2−8 (CONT.)

C. (CONT.)

$N(y^{(n)}, n) \cdot N(y^{(n-1)}, n-1) \cdot N(y^{(n-2)}, n-2) =$

$$\begin{bmatrix} 1, & 0, & \cdots & 0, & \bar{y}_1^{(n-1)} & \bar{y}_1^{(n)} \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ 0, & 0, & \cdots, & 1 & \bar{y}_{n-2}^{(n-1)} & \bar{y}_{n-2}^{(n)} \\ 0, & 0, & \cdots, & 0 & 1+\bar{y}_{n-1}^{(n-1)} & \bar{y}_{n-1}^{(n)} \\ 0, & 0, & \cdots, & 0, & \bar{y}_{n-1}^{(n-1)} & 1+y_n^{(n)} \end{bmatrix} \cdot \begin{bmatrix} 1, & 0, & \cdots, & y_1^{(n-2)}, & 0, & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ 0, & 0, & \cdots, & 1+y_{n-2}^{(n-2)}, & 0, & 0 \\ 0, & 0, & \cdots, & y_{n-1}^{(n-2)}, & 1, & 0 \\ 0, & 0, & \cdots, & y_n^{(n-2)}, & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1, & 0, & \cdots & y_1^{(n-2)} + \bar{y}_1^{(n-1)} \cdot y_{n-1}^{(n-2)} + \bar{y}_1^{(n)} \cdot y_n^{(n-2)} & \bar{y}_1^{(n-1)} & \bar{y}_1^{(n)} \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ 0, & 0, & \cdots & 1 + y_{n-2}^{(n-2)} + \bar{y}_{n-2}^{(n-1)} \cdot y_{n-1}^{(n-2)} + \bar{y}_{n-2}^{(n)} \cdot y_n^{(n-2)} & \bar{y}_{n-2}^{(n-1)} & \bar{y}_{n-2}^{(n)} \\ 0, & 0, & \cdots & y_{n-1}^{(n-2)} + \bar{y}_{n-1}^{(n-1)} \cdot y_{n-1}^{(n-2)} + \bar{y}_{n-1}^{(n)} y_n^{(n-2)} & 1+\bar{y}_{n-1}^{(n-1)} & \bar{y}_{n-1}^{(n)} \\ 0, & 0, & \cdots & y_n^{(n-2)} + \bar{y}_n^{(n-1)} \cdot y_{n-1}^{(n-2)} + \bar{y}_n^{(n)} \cdot y_n^{(n-2)} & \bar{y}_n^{(n-1)} & 1+\bar{y}_n^{(n)} \end{bmatrix}$$

And so on . . . .

The conditions to ensure the success of my algorithm is the leading principal submatries of A are nonsingular.

Following shows the statement for 2-by-2 case.

Let, $A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$, then $y_1 = \begin{bmatrix} \frac{1-a_{11}}{a_{11}} \\ -\frac{a_{21}}{a_{11}} \end{bmatrix}$; a_{11} must be nonzero

$$(I + y_1 e_1^T) A = \begin{bmatrix} 1 & a_{12}/a_{11} \\ 0 & (a_{22}a_{11} - a_{21}a_{12})/a_{11} \end{bmatrix} = \begin{bmatrix} 1 & a_{12}/a_{11} \\ 0 & \det(A)/a_{11} \end{bmatrix}$$

in order to performence the second Gauss-Jardon transformation, det (A) must be nonzero.;

Thus, all the leading principal submatrices of A must be nonsingular.

(to be continued.)

P4.2-8 (C) [CONT.]

ALGORITHM:

```
For i=1 to n
    p := 1/a_ii
    For j=1 to n
        a_ji := -a_ji *p
    a_ii := p + a_ii
    For k=i+1 to n
        p := a_ik
        For j=1 to n
            a_jk := a_jk + a_ji * p
For i=n-1, n-2, ..., 1
    For j=1 to n
        b_j := a_ji
        For k=i+1 to n
            b_j := b_j + a_jk * a_ki
        For j=1 to n
            a_ji := b_j
For i=1 to n
    a_ii := 1 + a_ii
```

$$① = \sum_{i=1}^{n} \sum_{k=i+1}^{n} \sum_{j=1}^{n} 1 = \sum_{i=1}^{n} \sum_{k=i+1}^{n} n = \sum_{i=1}^{n} n\cdot(n-i) = \frac{n^3}{2} + O(n^2) \text{ flops}$$

$$② = \frac{n^3}{2} + O(n^2) \text{ flops}$$

Following gives axample for n=6:

Results obtained based on the algorithm above.

This algorithm requires $n^3 + O(n^2)$ flops.

$$A_{old} = \begin{bmatrix} -0.04321 & -0.46814 & 0.79860 & 0.05229 & -0.11330 & -0.05259 \\ 0.40780 & -0.97423 & -0.08081 & -0.31885 & -0.28988 & 0.17655 \\ -0.08690 & 0.09425 & -0.03488 & -0.07713 & -0.28128 & 0.01940 \\ 0.00453 & 0.01777 & -0.65656 & -0.10736 & 0.26433 & -0.62195 \\ 0.39280 & 0.07262 & 0.48151 & -0.13622 & -0.34560 & 0.52026 \\ 0.03338 & 0.00525 & 0.46136 & 0.19660 & -0.41628 & 0.14333 \end{bmatrix}$$

$$A_{new} = \begin{bmatrix} -0.51578 & 0.16382 & -2.51785 & 1.70257 & 1.61773 & 1.46563 \\ -0.09517 & -0.85063 & 0.48558 & 0.56920 & 0.98783 & -0.16851 \\ 1.24958 & -0.53274 & 0.09913 & 0.26090 & 0.75149 & -0.49434 \\ -1.49064 & 0.29796 & -3.30085 & -0.34636 & -1.49264 & 3.44783 \\ 0.31688 & -0.32722 & -1.74102 & -0.40844 & 0.11650 & -1.44029 \\ -0.93361 & 0.34877 & -0.27929 & -1.96839 & -0.44613 & -0.67955 \end{bmatrix}$$

$$A_{old} * A_{new} = \begin{bmatrix} 1.00000 & 0.00000 & 0.00000 & -0.00000 & -0.00000 & 0.00000 \\ 0.00000 & 1.00000 & -0.00000 & -0.00000 & -0.00000 & 0.00000 \\ -0.00000 & -0.00000 & 1.00000 & 0.00000 & 0.00000 & -0.00000 \\ 0.00000 & 0.00000 & -0.00000 & 1.00000 & 0.00000 & -0.00000 \\ 0.00000 & 0.00000 & -0.00000 & 0.00000 & 1.00000 & 0.00000 \\ 0.00000 & -0.00000 & -0.00000 & 0.00000 & -0.00000 & 1.00000 \end{bmatrix}$$

10/10

P4.3-5. *Using $\beta = 10$, $t = 2$, chopped arithmetic, compute the L-U factorization of*

$$A = \begin{bmatrix} 7 & 6 \\ 9 & 8 \end{bmatrix}$$

For this example, what is the matrix H in (4.3-3)?

Solution:

$$M_1 = I - \alpha^{(1)} e_1^T = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 0 \\ 9/7 \end{bmatrix} \cdot [1, 0]$$

$$= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 0 & 0 \\ 1.2 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -1.2 & 1 \end{bmatrix}$$

$$M_1 \cdot A = \begin{bmatrix} 7 & 6 \\ 0.6 & 0.8 \end{bmatrix}$$

$$L = \begin{bmatrix} 1 & 0 \\ 1.2 & 1 \end{bmatrix} \qquad U = \begin{bmatrix} 7 & 6 \\ 0 & 0.8 \end{bmatrix}$$

$$\hat{A} = L \cdot U = \begin{bmatrix} 1 & 0 \\ 1.2 & 1 \end{bmatrix} \cdot \begin{bmatrix} 7 & 6 \\ 0 & 0.8 \end{bmatrix} = \begin{bmatrix} 7 & 6 \\ 8.4 & 8 \end{bmatrix}$$

$$= \begin{bmatrix} 7 & 6 \\ 9 & 8 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0.6 & 0 \end{bmatrix}$$

Thus,

$$H = \begin{bmatrix} 0 & 0 \\ 0.6 & 0 \end{bmatrix}$$

P4.4-4. Show that if $A^T \in R^{n\times n}$ is diagonally dominant, then A has a decomposition $A = LU$ with $|l_{ij}| \le 1$.

SHOW: A^T is diagonally dominant, then

$$|a_{jj}| > \Sigma_{i\neq j} |a_{ij}| \quad \text{for all } j$$

let

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}$$

$$M_1 = I - \alpha^{(1)} e_1^T$$

$$\alpha^{(1)} = (0, a_{21}/a_{11}, \cdots, a_{n1}/a_{11})$$
$$= (0, l_{21}, \cdots, l_{n1}) \quad ; \quad |l_{i1}| \le 1 \text{ for } i = 1, 2, \cdots, n$$
$$\& \ |l_{21}| + |l_{31}| + \cdots + |l_{n1}| \le 1$$

$$M_1 A = A - \alpha^{(1)} \cdot e_1^T A$$

$$= A - \begin{bmatrix} 0 \\ l_{21} \\ \vdots \\ l_{n1} \end{bmatrix} \cdot [a_{11} \ a_{12} \cdots a_{1n}]$$

$$= \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ 0 & a_{22} - l_{21}\cdot a_{12} & & a_{2n} - l_{21}\cdot a_{1n} \\ \vdots & \vdots & & \vdots \\ 0 & a_{n2} - l_{n1}\cdot a_{12} & & a_{nn} - l_{n1}\cdot a_{1n} \end{bmatrix} \triangleq \begin{bmatrix} a_{11} & \beta \\ 0 & A_{n-1} \end{bmatrix}$$

For any column in A_{n-1}, say column j, we have

$$\sum_{\substack{i=1 \\ i\neq j}}^{n-1} |a_{ij}^{(n-1)}| = |a_{2j} - l_{21}\cdot a_{1j}| + \cdots + |a_{nj} - l_{n1}\cdot a_{1j}| \quad (\text{exclude } j)$$
$$\le (|a_{2j}| + \cdots + |a_{j-1j}| + |a_{j+1j}| + \cdots + |a_{nj}|) + |a_{1j}| \cdot \sum_{\substack{i=2 \\ i\neq j}}^{n} |l_{ij}|$$
$$\le |a_{jj}| - |a_{1j}| + |a_{1j}|(1 - |l_{j1}|)$$
$$= |a_{jj}| - |a_{1j}| \cdot |l_{j1}| \le |a_{jj} - l_{j1}\cdot a_{1j}| = |a_{jj}^{(n-1)}|$$

Thus, A_{n-1}^T is also diagonally dominant. Then apply Gaussian Transformation to A_{n-1}, we have $|l_{22}| + |l_{32}| + \cdots + |l_{n2}| \le 1$. And by induction, we showed the statement.

P4.4-6: Suppose $A \in R^{n\times n}$ has an L-U decomposition and that L and U are known. Give an algorithm which can compute the (i,j) entry of A^{-1} in approximately $\frac{1}{2}(n-j)^2 + \frac{1}{2}(n-i)^2$ flops.

Solution: Let $A^{-1} = (x_1, x_2, \cdots, x_n)$; $x_i \in R^{n\times 1}$, $i = 1, 2, \cdots, n$. Then

$$AA^{-1} = LU(x_1, x_2, \cdots, x_n) = I = (e_1, e_2, \cdots, e_n)$$

And

$$L\underbrace{U x_j}_{y} = e_j$$

$$Ly = \left[\quad\right] \cdot \begin{bmatrix} y_1 \\ \vdots \\ y_j \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix} \Rightarrow y_1 = y_2 = \cdots = y_{j-1} = 0$$

$$U x_j = \left[\quad\right] \cdot \begin{bmatrix} a_{1j}^{(-1)} \\ \vdots \\ a_{ij}^{(-1)} \\ \vdots \\ a_{nj}^{(-1)} \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ y_j \\ \vdots \\ y_n \end{bmatrix}$$

Algorithm:

$y_j := 1/\ell_{jj}$

For $k = j+1, \cdots, n$

  $y_k := 0$

  For $h = j, \cdots, k-1$

    $y_k := y_k - \ell_{kh} \cdot y_h$ } $k-j$ flops

  $y_k := y_k / \ell_{kk}$

} $\approx \frac{1}{2}(n-j)^2$ flops

For $k = n, \cdots, i$

  $a_{kj} := y_k$

  For $h = k+1, \cdots, n$

    $a_{kj} := a_{kj} - u_{kh} \cdot a_{kh}$ } $n-k$ flops

  $a_{kj} := a_{kj} / u_{kk}$

} $\approx \frac{1}{2}(n-i)^2$ flops

Total flops $= \sum_{k=j+1}^{n} (k-j) + \sum_{k=i}^{n} (n-k) = \frac{1}{2}(n-j)^2 + \frac{1}{2}(n-i)^2 + O(n)$ flops.

P4.5-2: Using $\beta=10$, $t=2$, chopped arithmeti, solve

$$\begin{bmatrix} 11 & 15 \\ 5 & 7 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 7 \\ 3 \end{bmatrix}$$

Using Gaussian elimenation with partial pivoting. Do one step of iterative improvement using $t=4$ arithmetic to compute the residual.

Solution: $P_1 = I_2$ and

$$M_1 = \begin{bmatrix} 1 & 0 \\ -0.45 & 1 \end{bmatrix} \qquad M_1 \cdot \begin{bmatrix} 15 \\ 7 \end{bmatrix} = \begin{bmatrix} 15 \\ 0.30 \end{bmatrix}$$

$$L = \begin{bmatrix} 1 & 0 \\ 0.45 & 1 \end{bmatrix} \qquad U = \begin{bmatrix} 11 & 15 \\ 0 & 0.30 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 \\ 0.45 & 1 \end{bmatrix} \cdot \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 7 \\ 3 \end{bmatrix} \Rightarrow \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 7 \\ -0.10 \end{bmatrix}$$

$$\begin{bmatrix} 11 & 15 \\ 0 & 0.30 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 7 \\ -0.10 \end{bmatrix} \Rightarrow \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{bmatrix} = \begin{bmatrix} 1.0 \\ -0.33 \end{bmatrix}$$

Iterative improvement:

$$r = \begin{bmatrix} 7 \\ 3 \end{bmatrix} - \begin{bmatrix} 11 & 15 \\ 5 & 7 \end{bmatrix} \cdot \begin{bmatrix} 1.0 \\ -0.33 \end{bmatrix} = \begin{bmatrix} 0.95 \\ 0.31 \end{bmatrix}$$

$$Ly = r : \begin{bmatrix} 1 & 0 \\ 0.45 & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 0.95 \\ 0.31 \end{bmatrix} \Rightarrow \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 0.95 \\ -0.11 \end{bmatrix}$$

$$Uz = y : \begin{bmatrix} 11 & 15 \\ 0 & 0.30 \end{bmatrix} \cdot \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} 0.95 \\ -0.11 \end{bmatrix} \Rightarrow \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} 0.57 \\ -0.36 \end{bmatrix}$$

$$x_{new} = \begin{bmatrix} 1.0 \\ -0.33 \end{bmatrix} + \begin{bmatrix} 0.57 \\ -0.36 \end{bmatrix}$$

$$= \begin{bmatrix} 1.5 \\ -0.69 \end{bmatrix}$$

The actual solution is

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 2.0 \\ -1.0 \end{bmatrix}$$

One more step will make the solution much better.

P5.14 (Symmetric storage) Suppose the n-by-n symmetrix matrix $A=(a_{ij})$ is stored in an $n(n+1)/2$ vector c as follows:

$$c=(a_{11}, a_{21}, a_{22}, a_{31}, \cdots, a_{n1}, a_{n2}, \cdots, a_{nn})$$

Rewrite Algorithm 5.1-2 with A stored in this fashion.

Solution: We note that for any a_{ij}, the corresponding element in Vector c is $c_{(i-1)i/2+j}$

Algorithm:

For $k=1, \cdots, n$

$s := k(k+1)/2 \quad : \quad d_k := c_s$

For $p=1, \cdots, k-1$

$s := (k-1)\cdot k/2+p$

$\gamma_p := d_p \cdot c_s$

$d_k := d_k - c_s \cdot \gamma_p$

If $d_k = 0$

then quit

else

For $i=k+1, \cdots, n$

$t := (i-1)\cdot i/2+k$

For $p=1, \cdots, k-1$

$s := (i-1)\cdot i/2 + p$

$c_t := c_t - c_s \cdot \gamma_p$

$c_t := c_t / d_k$

P5.2-1: Show that if

$$A = \begin{array}{c} \left[\begin{array}{cc} \alpha & W^T \\ V & B \end{array}\right] \begin{array}{c} 1 \\ n-1 \end{array} \\ \begin{array}{cc} 1 & n-1 \end{array} \end{array}$$

is positive definite, then so is $B - VW^T/\alpha$.

Show:

$$A = \begin{bmatrix} 1 & 0 \\ V/\alpha & I_{n-1} \end{bmatrix} \cdot \begin{bmatrix} \alpha & 0 \\ 0 & B - VW^T/\alpha \end{bmatrix} \cdot \begin{bmatrix} 1 & W^T/\alpha \\ 0 & I_{n-1} \end{bmatrix}$$

Let y be any $n-1$ vector and

$$a = -y^T V/\alpha$$

Since A is p.d.

$$[a \quad y^T]\, A \begin{bmatrix} a \\ y \end{bmatrix} > 0$$

That is

$$[a \quad y^T] \cdot \begin{bmatrix} 1 & 0 \\ V/\alpha & I_{n-1} \end{bmatrix} \cdot \begin{bmatrix} \alpha & 0 \\ 0 & B-VW^T/\alpha \end{bmatrix} \cdot \begin{bmatrix} 1 & W^T/\alpha \\ 0 & I_{n-1} \end{bmatrix} \cdot \begin{bmatrix} a \\ y \end{bmatrix}$$

$$= [0 \quad y^T] \cdot \begin{bmatrix} \alpha & 0 \\ 0 & B-VW^T/\alpha \end{bmatrix} \cdot \begin{bmatrix} a + W^T y/\alpha \\ y \end{bmatrix}$$

$$= [0 \quad y^T(B - VW^T/\alpha)] \begin{bmatrix} a + W^T y/\alpha \\ y \end{bmatrix}$$

$$= y^T (B - VW^T/\alpha)\, y > 0$$

Thus $B - VW^T/\alpha$ is positive definite. Q.E.D.

P5.2 — 3 See the statement on page 90 in text.

a) It is easy to check that

$$A=\begin{bmatrix}1 & 0\\ V_1/\alpha_1 & I_{n-1}\end{bmatrix}\cdot\begin{bmatrix}\alpha_1 & 0\\ 0 & A_1-V_1V_1^T/\alpha_1\end{bmatrix}\begin{bmatrix}1 & V_1^T/\alpha_1\\ 0 & I_{n-1}\end{bmatrix}$$

$$=\begin{bmatrix}1 & 0 & 0\\ V_1/\alpha & 1 & 0\\ & V_2/\alpha & I_{n-2}\end{bmatrix}\cdot\begin{bmatrix}\alpha_1 & & 0\\ & \alpha_2 & \\ 0 & & A_2-V_2V_2^T/\alpha_2\end{bmatrix}\cdot\begin{bmatrix}1 & & V_1^T/\alpha_1\\ & 1 & V_2^T/\alpha_2\\ 0 & & I_{n-1}\end{bmatrix}$$

Algorithm: overwrite A with L

For $j=1,2,\cdots,n-1$

$d_j := a_{jj}$

For $i=j+1,\cdots,n$

$a_{ij} := a_{ij}/d_j$

$\frac{1}{2}(n-j)^2$ flops $\approx$ { For $i=j+1,\cdots,n$

For $k=j+1,\cdots,i$

$a_{ik} := a_{ik} - a_{ij}\cdot a_{kj}\cdot d_j$ }

$$\text{Total flops} \doteq \sum_{j=1}^{n-1}\frac{1}{2}(n-j)^2 \doteq \frac{1}{2}n^3-\frac{1}{2}n^3+\frac{1}{2}\cdot\frac{1}{3}n^3=\frac{1}{6}n^3+O(n^2)$$

b). Since the largest element in a positive definite matrix occurs on the diagonal.

$$P_1^TAP_1 = L_1\cdot\begin{bmatrix}d_1 & 0\\ 0 & A_1\end{bmatrix}L_1^T$$

$$\rightarrow\quad L_1^{-1}P_1^TAP_1L_1^{-T}=\begin{bmatrix}d_1 & 0\\ 0 & A_1\end{bmatrix}$$

$$\rightarrow\quad P_2^TL_1^{-1}P_1^TAP_1\cdot L_1^{-T}P_2 = L_2\cdot\begin{bmatrix}d_1 & & \\ & d_2 & \\ & & A_2\end{bmatrix}\cdot L_2^T$$

$$\rightarrow\quad\cdots\quad\rightarrow$$

$$L_{n-1}P_{n-1}^T L_{n-2}^{-1}P_{n-2}^T\cdots L_1^{-1}P_1^TA\cdot P_1\cdot L_1^{-T}\cdots L_{n-1}^{-T}=\begin{bmatrix}d_1 & & \\ & d_2 & \\ & & \ddots & \\ & & & d_n\end{bmatrix}\triangleq D$$

$$A=(P_1L_1\cdot P_2L_2\cdots P_{n-1}L_{n-1})D\cdot(P_1L_1\cdot P_2L_2\cdots P_{n-1}L_{n-1})^T$$

Let $P=P_1P_2\cdots P_{n-1}$, Then

$$P^TAP=(P_{n-1}\cdots P_2\cdot P_1\cdot P_1L_1P_2L_2\cdots P_{n-1}L_{n-1})\cdot D\cdot(P_1L_1\cdots P_{n-1}L_{n-1})^T\cdot P.$$

It can be shown that

$$L \triangleq P_{n-1} \cdots P_2 \cdot P_1 \cdot P_1 L_1 P_2 \cdots P_{n-1} L_{n-1} \quad \text{is lower tragular.}$$

$$P^T A P = L \cdot D \cdot L^T \quad , \quad \text{where}$$

$$D = \begin{bmatrix} d_1 & & & \\ & d_2 & & \\ & & \ddots & \\ & & & d_n \end{bmatrix} \quad , \quad d_1 \geq d_2 \geq \cdots d_n > 0$$

<u>Algorithm:</u>

For $k = 1, \cdots, n-1$ $\{ \text{or} = \text{rank}(A) \text{ in part C} \}$

Determine indice $p \in \{k, \cdots, n\}$ such that

$$a_{pp} = \max_{k \leq i \leq n} a_{ii}$$

$r_k := p$

Swap a_{kj} and a_{pj} $(j = k, \cdots, n)$

Swap a_{ik} and a_{ip} $(i = 1, \cdots, m)$

$d_k := a_{kk}$

For $i = k+1, \cdots, n$

$a_{ik} := a_{ik} / d_k$

For $i = k+1, \cdots, n$

For $j = k+1, \cdots, n$

$a_{ij} := a_{ij} - a_{ik} \cdot a_{jk} \cdot d_k$

C) If rank$(A) = r$, then from the algorithm in part b), it is clear to see that after r steps $P = P_1 \cdot P_2 \cdots P_r$

$$P^T A P =$$

$$\begin{bmatrix} 1 & & & & 0 \\ l_{21} & \ddots & & & 0 \\ \vdots & & 1 & 0 & 0 \\ \vdots & & \vdots & 1 & 0 \\ l_{n1} & & l_{nr} & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} d_1 & & & & \\ & d_2 & & & \\ & & \ddots d_r & & \\ & & & 0 & \\ & & & & \ddots 0 \end{bmatrix} L^T$$

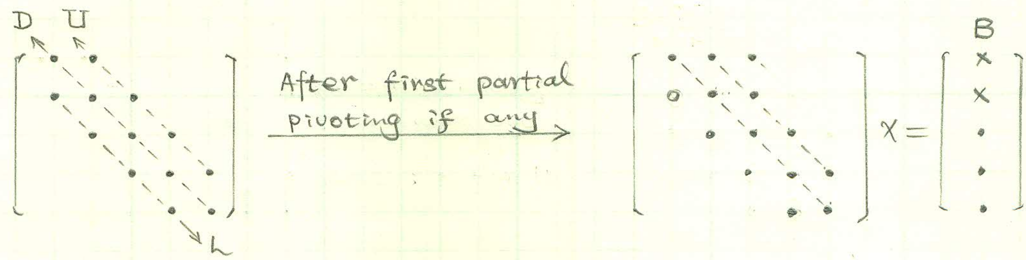
$$= \begin{bmatrix} L_{11} & 0 \\ L_{12} & I_{n-r} \end{bmatrix} \begin{bmatrix} D^* & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} L_{11}^T & L_{12}^T \\ 0 & I_{n-r} \end{bmatrix}$$

$$= \begin{bmatrix} L_{11} D L_{11}^T & L_{11} D L_{12}^T \\ L_{12} D L_{11}^T & L_{12} D L_{12}^T \end{bmatrix} = \begin{bmatrix} L_{11} \\ L_{12} \end{bmatrix} D \cdot \begin{bmatrix} L_{11}^T & L_{12}^T \end{bmatrix} \triangleq L^* D^* L^{*T}$$

Q.E.D.

P5.3.3 Give an algorithm for solving unsymmetric tridiagonal systems $Ax = b$ that use Gaussian elimination with partial pivoting and which requires only 4 n-vectors of floating point storage.

Solution:



$$P_1 A = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ l_1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \times & \times & ? & 0 & 0 \\ 0 & \times & \times & 0 & 0 \\ 0 & \times & \times & \times & 0 \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} \times \\ \times \\ \times \\ \times \\ \times \end{bmatrix} = P_1 B \triangleq \hat{B}$$

Solve $Ly = \hat{B} \Rightarrow$

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_5 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 - l_1 \cdot b_1 \\ b_3 \\ \vdots \\ b_5 \end{bmatrix}$$

Then, we can overwrite l_1 with ?, because it is no longer used.

And then for the second, third, ..., until the n-1 th step, we can actually use only 4 n-vectors of floating point storage.

The algorithm is given on next page.

The algorithm has been tested working O.K. and it requires about $O(n)$ flops.

Algorithm: For $k = 1, \cdots, n-1$

If $d_k < l_k$

then

$S := d_k$; $d_k := l_k$; $l_k := S$

$S := u_k$; $u_k := d_{k+1}$; $d_{k+1} := S$

$S := b_k$; $b_k := b_{k+1}$; $b_{k+1} := S$

$S := u_{k+1}$

Partial Pivoting

else

$S := 0$

$l_k := l_k / d_k$

$b_{k+1} := b_{k+1} - l_k \cdot b_k$

$d_{k+1} := d_{k+1} - l_k \cdot u_k$

If $S \neq 0$ then $u_{k+1} := -l_k \cdot S$

$l_k := S$ (overwrited)

$b_n := b_n / d_n$

$b_{n-1} := (b_{n-1} - u_{n-1} \cdot b_n) / d_{n-1}$

For $i = n-2, \cdots, 1$

$b_i := (b_i - u_i \cdot b_{i+1} - l_i \cdot b_{i+2}) / d_i$

solving for x

Note: Original tridiagonal system $Ax = b$

the lower sub-diagonal are stored in vector L

the diagonal are stored in vector D

the upper sub-diagonal are stored in vector U

And the final solution are overwrited in b.

P6.1-2 Define the function $\phi: \mathbb{R}^n \to \mathbb{R}$ by $\phi(x) = \frac{1}{2}\|Ax-b\|_2^2$, show that the gradient $\nabla\phi(x)$ is given by $\nabla\phi(x) = A^T(Ax-b)$

Show:

$$\phi(x) = \frac{1}{2}\|Ax-b\|_2^2 = \frac{1}{2}(Ax-b)^T\cdot(Ax-b)$$

$$= \frac{1}{2}\left[x^T\cdot A^T\cdot A\cdot x - x^TA^T\cdot b - b^T\cdot A\cdot x + b^Tb\right]$$

$$\nabla\phi(x) = \frac{1}{2}\left[\nabla(x^TA^TAx) - 2\nabla(x^TA^Tb)\right]$$

Assume: A is $m\times n$ matrix; b is $m\times 1$

then $A^T\cdot b$: $n\times m \times m\times 1 = n\times 1$

Let $A^Tb = [\gamma_1, \cdots, \gamma_n]^T$

$$x^T\cdot A^Tb = (x_1, \cdots, x_n)\begin{bmatrix}\gamma_1\\ \vdots\\ \gamma_n\end{bmatrix} = \gamma_1x_1 + \cdots + \gamma_nx_n$$

Then

$$\nabla(x^TA^Tb) = (\gamma_1, \cdots, \gamma_n)^T = A^Tb$$

Let

$$A^TA = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & \ddots & & \\ a_{1n} & a_{2n} & \cdots & a_{nn}\end{bmatrix}$$

$$x^TA^TAx = \left[\sum_{i=1}^n a_{1i}\cdot x_i, \cdots, \sum_{i=1}^n a_{ni}\cdot x_i\right]\cdot\begin{bmatrix}x_1\\ \vdots\\ x_n\end{bmatrix}$$

$$= a_{11}\cdot x_1^2 + \left(\sum_{i=2}^n a_{1i}x_i\right)x_1 + \cdots$$

$$+ \left(\sum_{i=1}^{n-1} a_{ni}x_i\right)\cdot x_n + a_{nn}\cdot x_n^2$$

$$\therefore \nabla(x^TA^TAx) = \Big[2a_{11}\cdot x_1 + \sum_{i=2}^n a_{1i}x_i + a_{12}\cdot x_2 + \cdots + a_{1n}\cdot x_n,$$

$$\cdots, 2a_{nn}x_n + \sum_{i=1}^{n-1} a_{in}\cdot x_i + a_{1n}x_1 + \cdots + a_{n-1n}x_{n-1}\Big]^T$$

$$= 2A^TAx$$

$$\therefore \nabla\phi(x) = \frac{1}{2}\left[2A^TAx - 2A^Tb\right] = A^T(Ax-b)$$

Q.E.D.

P6.1-8 Use the method of normal equations to solve the LS problem where

$$A=\begin{bmatrix}1&2\\3&4\\5&6\end{bmatrix} \qquad b=\begin{bmatrix}1\\1\\1\end{bmatrix}$$

$$A^TA=\begin{bmatrix}1&3&5\\2&4&6\end{bmatrix}\cdot\begin{bmatrix}1&2\\3&4\\5&6\end{bmatrix}=\begin{bmatrix}35&44\\44&56\end{bmatrix}$$

$$A^Tb=\begin{bmatrix}1&3&5\\2&4&6\end{bmatrix}\cdot\begin{bmatrix}1\\1\\1\end{bmatrix}=\begin{bmatrix}9\\12\end{bmatrix}$$

and

$$A^TA=\begin{bmatrix}5.91608&0\\7.43736&0.82808\end{bmatrix}\cdot\begin{bmatrix}5.91608&7.43736\\0&0.82808\end{bmatrix}$$

Solve:

$$\begin{bmatrix}5.91608&0\\7.43736&0.82808\end{bmatrix}\cdot\begin{bmatrix}y_1\\y_2\end{bmatrix}=\begin{bmatrix}9\\12\end{bmatrix}$$

$$\begin{bmatrix}y_1\\y_2\end{bmatrix}=\begin{bmatrix}1.52128\\0.82807\end{bmatrix}$$

Solve:

$$\begin{bmatrix}5.91608&7.43736\\0&0.82808\end{bmatrix}\cdot\begin{bmatrix}x_{1LS}\\x_{2LS}\end{bmatrix}=\begin{bmatrix}1.52128\\0.82807\end{bmatrix}$$

We have

$$x_{LS}=\begin{bmatrix}-1.00000\\1.00000\end{bmatrix}$$

P 6.2-2 Compute the Q-R factorization of $A = \begin{bmatrix} 5 & 9 \\ 12 & 7 \end{bmatrix}$.

Solution: $V_1 = \begin{bmatrix} 18 \\ 12 \end{bmatrix}$

$$P_1 = I - 2V_1V_1^T / V_1^TV_1$$

$$= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \frac{1}{234} \begin{bmatrix} 324 & 216 \\ 216 & 144 \end{bmatrix}$$

$$= \begin{bmatrix} -0.38462 & -0.92308 \\ -0.92308 & 0.38462 \end{bmatrix}$$

$$P_1 A = \begin{bmatrix} -13.00006 & -9.92314 \\ 0.00004 & -5.61538 \end{bmatrix}$$

$$Q = \begin{bmatrix} -.38462 & -.92308 \\ -.92308 & 0.38462 \end{bmatrix}$$

$$R = \begin{bmatrix} -13.00006 & -9.92314 \\ 0.00004 & -5.61538 \end{bmatrix}$$

$$QR = \begin{bmatrix} 5.00005 & 9.00008 \\ 12.00011 & 7.00006 \end{bmatrix}$$

CHECK !

P6.2-7 The matrix $C = (A^TA)^{-1}$, where rank $(A) = n$, arises in many statistical applications and is known as the variance-covariance matrix. This problem shows how the decomposition $A = QR$ is useful in computations involving C.

(a) Show $C = (R^TR)^{-1}$

Assume $A = QR$, then $A^T = R^TQ^T$

$$A^T \cdot A = R^TQ^T \cdot Q \cdot R = R^TR$$

Then $C = (A^TA)^{-1} = (R^T \cdot R)^{-1}$ checked.

(b) Give an algorithm for computing $c_{11}, \cdots, c_{nn}$ requiring $n^3/6$ flops.

Solution: I think in this problem, we may assume we alreay Q, R for A.

And
$$R = \begin{bmatrix} R_1 \\ 0 \end{bmatrix}.$$

Then $R^T \cdot R = R_1^T R_1$. And then solve $(R_1^T R_1) c_i = e_i$

$$\begin{bmatrix} r_{11} & 0 & \cdots & 0 \\ r_{12} & r_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ r_{1n} & r_{2n} & \cdots & r_{nn} \end{bmatrix} \cdot \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ 0 & r_{22} & \cdots & r_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & r_{nn} \end{bmatrix} \cdot \begin{bmatrix} c_{1i} \\ \vdots \\ c_{ii} \\ \vdots \\ c_{ni} \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix}$$

Solve:
$$\begin{bmatrix} r_{11} & & & \\ r_{12} & r_{22} & & \\ \vdots & \vdots & \ddots & \\ r_{1n} & r_{2n} & \cdots & r_{nn} \end{bmatrix} \cdot \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix} i \Rightarrow \begin{bmatrix} y_1 \\ \vdots \\ y_i \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ x \\ \vdots \\ x \end{bmatrix} i$$

Solve:
$$\begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ & r_{22} & \cdots & r_{2n} \\ & & \ddots & \vdots \\ & & & r_{nn} \end{bmatrix} \cdot \begin{bmatrix} c_{1i} \\ \vdots \\ c_{ii} \\ \vdots \\ c_{ni} \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ x \\ \vdots \\ x \end{bmatrix}$$

To be continued.

P6.2-7 (b) (CONT.)

Algorithm:

For $k = 1, 2, \cdots, n$

$y_K := 1/r_{KK}$

For $i = k+1, \cdots, n$

$y_i := 0$

For $j = k, \cdots, i-1$

$y_i := y_i - r_{ji} \cdot y_j$

$y_i := y_i / r_{ii}$

$\doteq \frac{1}{2}(n-k)^2$ flops

For $i = n, \cdots, k$

For $j = i+1, \cdots, n$

$y_i := y_i - r_{ij} \cdot y_j$

$y_i := y_i / r_{ii}$

$\doteq \frac{1}{2}(n-k)^2$ flops

$C_{KK} := y_K$

Total flops $= \sum_{K=1}^{n} (n-k)^2 = \frac{n^3}{3} + O(n^2) \longrightarrow$?

$-.5$

For more efficient algorithm compute R^{-1} first

P6.3-2 Algorithm 6.3-2 is easily adapted to solve the weighted least squares problem

$$\min \| D(AX - b)\|_2$$

where $A \in R^{m\times n}$, $b \in R^m$, and $D = \mathrm{diag}(d_i)$, $d_i > 0$. Indeed, if

$$MD^{-2}M^T = \tilde{D} = \mathrm{diag}(\tilde{d}_i) \qquad \tilde{d}_i > 0$$

and

$$MA = \begin{array}{c c} \begin{bmatrix} S_1 \\ 0 \end{bmatrix} & \begin{array}{c} n \\ m-n \end{array} \end{array} \qquad Mb = \begin{array}{c c} \begin{bmatrix} C \\ d \end{bmatrix} & \begin{array}{c} n \\ m-n \end{array} \end{array}$$

then the solution to the weighted LS problem is obtained by solving $S_1 X = C$. Show this and indicate changes in Algorithm 6.3-2 are required to handle this problem.

Solution: Define the orthogonal transformation

$$Q \triangleq \tilde{D}^{-\frac{1}{2}} \cdot M \cdot D^{-1}$$

$$QQ^T = \tilde{D}^{-\frac{1}{2}} M D^{-1} \cdot D^{-1} \cdot M^T \cdot \tilde{D}^{-\frac{1}{2}} = \tilde{D}^{-\frac{1}{2}} \cdot \tilde{D} \cdot \tilde{D}^{-\frac{1}{2}} = I$$

Then

$$\|D(AX-b)\|_2 = \|QD(AX-b)\|_2$$

$$= \|\tilde{D}^{-\frac{1}{2}}(MAx - Mb)\|_2$$

$$= \left\| \tilde{D}^{-\frac{1}{2}} \left(\begin{bmatrix} S_1 \\ 0 \end{bmatrix} x - \begin{bmatrix} c \\ d \end{bmatrix} \right) \right\|_2$$

for any $x \in R^n$, Clearly, X_{LS} is obtained by solving $S_1 X = C$.

- - - - - - - - - - - -

Change the first statement to $\tilde{d}_i := 1/d_i^2$

And all $dx \Rightarrow \tilde{d}x$ in the original algorithm.

P6.4-6 Show that if

$$A = \begin{bmatrix} R & w \\ 0 & v \end{bmatrix} \begin{array}{l} k \\ m-k \end{array} \qquad b = \begin{bmatrix} c \\ d \end{bmatrix} \begin{array}{l} k \\ m-k \end{array}$$

and A has full rank, then

$$\min \|AX - b\|_2^2 = \|d\|_2^2 - (v^T d / \|v\|_2)^2$$

Show : Since A is full rank,

$$AX = \begin{bmatrix} R & w \\ 0 & v \end{bmatrix} \cdot \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \begin{bmatrix} c \\ d \end{bmatrix}$$

Solve $v X_2 = d$, to minimize $\|v X_2 - d\|_2^2$

$$v^T v \cdot X_2 = v^T \cdot d$$

$$X_2 = v^T \cdot d / v^T \cdot v = v^T \cdot d / \|v\|_2^2 \quad \cdots\cdots (1)$$

And $\quad R X_1 + w X_2 = c$

$$R X_1 = c - w X_2 = c - w v^T \cdot d / \|v\|_2^2 \quad \cdots\cdots (2)$$

Since A is full rank , we will have no error to solve (2)

$$\therefore \min \|AX - b\|_2^2 = \min \|v X_2 - b\|_2^2$$

$$= \| v v^T \cdot d / \|v\|_2^2 - d \|_2^2$$

$$= (d^T - (v^T d) \cdot v^T / \|v\|_2^2) \cdot (d - (v^T d) \cdot v / \|v\|_2^2)$$

$$= \|d\|_2^2 - (v^T d / \|v\|_2)^2$$

P6.5-1 Givens transformations can be used to reduce an upper triangular matrix R to upper bidiagonal form. The form of the algorithms is as follows:

For $j = n, \cdots, 3$

For $i = 1, \cdots, j-2$

$\boxed{\text{SPACE 1:}}$ $R := UR$ (U rotates rows i and $i+1$ and zeros r_{ij})

$\boxed{\text{SPACE 2:}}$ $R := RV$ (V rotates columns i and $i+1$ & zeros $r_{i+1,i}$)

Fill in the details of this algorithm and given, an operation count.

Solution: SPACE 1:

$$c := r_{ij} / (r_{ij}^2 + r_{i+1\,j}^2)^{1/2}$$

$$s := r_{i+1\,j} / (r_{ij}^2 + r_{i+1\,j}^2)^{1/2}$$

For $k = i, \cdots, j$

$\alpha := r_{ik}$; $\beta := r_{i+1\,k}$

$r_{ik} := \alpha * c + \beta * s$

$r_{i+1\,k} := \beta * c - \alpha * s$ $\Big\}$ 4 flops

$\Big\} \approx 4(j-i)$ flops

$$U = \begin{array}{c} \begin{array}{cccccc} & & i & i+1 & & \end{array} \\ \left[\begin{array}{cccccc} 1 & & & & & \\ & \ddots & & & & \\ & & c & s & & \\ & & -s & c & & \\ & & & & \ddots & \\ & & & & & 1 \end{array}\right] \begin{array}{c} \\ \\ i \\ i+1 \\ \\ \\ \end{array} \end{array}$$

SPACE 2:

$$c := r_{i+1\,i+1} / (r_{i+1\,i}^2 + r_{i+1\,i+1}^2)^{1/2}$$

$$s := r_{i+1\,i} / (r_{i+1\,i}^2 + r_{i+1\,i+1}^2)^{1/2}$$

For $k = 1, \cdots, i+1$

$\alpha := r_{ki}$; $\beta := r_{k,i+1}$

$r_{ki} := \alpha * c - \beta * s$

$r_{k,i+1} := \alpha * s + \beta * c$ $\Big\}$ 4 flops

$\Big\} \approx 4i$ flops

$$V = \begin{array}{c} \begin{array}{cccccc} & & i & i+1 & & \end{array} \\ \left[\begin{array}{cccccc} 1 & & & & & \\ & \ddots & & & & \\ & & c & s & & \\ & & -s & c & & \\ & & & & \ddots & \\ & & & & & 1 \end{array}\right] \begin{array}{c} \\ \\ i \\ i+1 \\ \\ \\ \end{array} \end{array}$$

Total flops $\cong \sum_{j=3}^{n} \sum_{i=1}^{j-2} 4j \cong \frac{4}{3} n^3$ flops.

P7.1-3 Show that if $T=\begin{bmatrix} P & R \\ 0 & S \end{bmatrix}$ with P and S square, then $\lambda(T)=\lambda(P) \cup \lambda(S)$.

8/8

Show: For any

$$W=\begin{bmatrix} A & B \\ 0 & C \end{bmatrix}=\begin{bmatrix} I & 0 \\ 0 & C \end{bmatrix}\cdot\begin{bmatrix} I & B \\ 0 & I \end{bmatrix}\cdot\begin{bmatrix} A & 0 \\ 0 & I \end{bmatrix}$$

Then, $\det(W)=\det(C)\cdot\det(A)$

$$\lambda I-T=\lambda I-\begin{bmatrix} P & R \\ 0 & S \end{bmatrix}$$

$$=\begin{bmatrix} \lambda I_p-P & -R \\ 0 & \lambda I_s-S \end{bmatrix}$$

$$\det(\lambda I-T)=\det(\lambda I_p-P)\cdot\det(\lambda I_s-S)$$

This implies that $\lambda(T)=\lambda(P)\cup\lambda(S)$. Q.E.D.

P7.1-8 Suppose $A\in\mathbb{C}^{n\times n}$ has distinct eigenvalues, show that if $Q^HAQ=T$ is its schur decomposition and $AB=BA$, the Q^HBQ is upper trangular.

Show: $AB=BA$ Then $Q^HAQ\cdot Q^HBQ=Q^HBQQ^HAQ$

Let $Q^HBQ=R \Rightarrow TR=RT$

Assume

$$T=\begin{array}{cc} \begin{array}{cc} 1 & n-1 \end{array} & \\ \begin{bmatrix} \lambda_1 & \times \\ 0 & T_1 \end{bmatrix} & \begin{array}{c} 1 \\ n-1 \end{array} \end{array}, \quad R=\begin{array}{cc} \begin{array}{cc} 1 & n-1 \end{array} & \\ \begin{bmatrix} \gamma_1 & \times \\ \gamma & R_1 \end{bmatrix} & \begin{array}{c} 1 \\ n-1 \end{array} \end{array}$$

$$TR=\begin{bmatrix} \times & \times \\ T_1\gamma & \times \end{bmatrix}=RT=\begin{bmatrix} \times & \times \\ \gamma\lambda_1 & \times \end{bmatrix}$$

We have $T_1\gamma=\gamma t_1$, that is

$$(\lambda_1 I_{n-1}-T_1)\gamma=0$$

Since T_1 is upper trangulor with eigenvalues of A (excluded λ_1) in diagonal, and we know that A has distinct eigenvalues. that is $\lambda_1 I_{n-1}-T_1$ is nonsingular. Thus $\gamma\equiv 0$. By induction we proved the statement.

P7.2-5 Show that if $A=\begin{bmatrix} a & c \\ 0 & b \end{bmatrix}$, then $s(a)=s(b)=(1+|c/(a-b)|^2)^{-1}$.

SHOW: $\lambda_1 = a$, $\lambda_2 = b$

$$(\lambda_1 I - A)x_a = \begin{bmatrix} 0 & -c \\ 0 & a-b \end{bmatrix}\begin{bmatrix} x_{1a} \\ x_{2a} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

So, we can pick $x_a = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$

$$(\lambda_2 I - A)x_b = \begin{bmatrix} b-a & -c \\ 0 & 0 \end{bmatrix}\cdot\begin{bmatrix} x_{1b} \\ x_{2b} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Pick $x_b = \frac{1}{\sqrt{c^2+(b-a)^2}}\begin{bmatrix} c \\ b-a \end{bmatrix}$

$$y_a^H(\lambda_1 I - A) = [y_{1a}, y_{2a}]\cdot\begin{bmatrix} 0 & -c \\ 0 & a-b \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Pick $y_a = \frac{1}{\sqrt{c^2+(a-b)^2}}\begin{bmatrix} a-b \\ c \end{bmatrix}$ ✓

$$y_b^H(\lambda_2 I - A) = [y_{1b}, y_{2b}]\cdot\begin{bmatrix} b-a & -c \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Pick $y_b = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

Then $s(a) \equiv \left| \frac{1}{\sqrt{c^2+(a-b)^2}}[a-b, c]\cdot\begin{bmatrix} 1 \\ 0 \end{bmatrix} \right|$

$$= \left| \frac{a-b}{\sqrt{c^2+(a-b)^2}} \right| = \left[\frac{|a-b|^2}{c^2+(a-b)^2} \right]^{1/2}$$

$$s(b) \equiv \left| [0, 1]\cdot\frac{1}{\sqrt{c^2+(a-b)^2}}\begin{bmatrix} c \\ b-a \end{bmatrix} \right|$$

$$= \frac{|a-b|}{\sqrt{c^2+(a-b)^2}} = [1+|c/(a-b)|^2]^{-1/2} \equiv s(a)$$ ✓

But $\left(1+\frac{c^2}{(a+b)^2}\right)^{-1} = \frac{(a+b)^2}{c^2+(a+b)^2}$?

This correct

P7.3-3 Verify that (7.3-10) calculates the matrices T_K defined by (7.3-9)

VERIFICATION:

From (7.3-10): $T_{K-1} = L_K R_K$

$$\Rightarrow R_K = L_K^{-1} \cdot T_{K-1}$$

$$\Rightarrow T_K = R_K \cdot L_K = L_K^{-1} \cdot T_{K-1} \cdot L_K$$

$$= L_K^{-1} \cdot L_{K-1}^{-1} \cdot T_{K-2} \cdot L_{K-1} \cdot L_K$$

$$= \underbrace{(L_0 \cdot L_1 \cdots L_K)}_{G_K}{}^{-1} \cdot A \cdot (L_0 \cdot L_1 \cdots L_K)$$

$$= G_K^{-1} \cdot A \cdot G_K$$

P. 7.3-4 Suppose the power method (7.3-3) is applied to $A = \begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix}$ with $V^{(0)} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$, what is $\lambda^{(20)}$?

Answer: $\lambda^{(20)} = 2$

proof: $z^{(1)} = \begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$

$\lambda^{(1)} = 2$

$V^{(1)} = \begin{bmatrix} 2 \\ 2 \end{bmatrix} / 2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

And repeat again and again.

Note: $V^{(0)}$ is already an eigenvector of A associated to the eigenvalue $\lambda = 2$.

P7.4-3. In some situations it is necessary to solve the linear system $(A + z\cdot I)x = b$ for many different values of $z \in \mathbb{R}$ and $b \in \mathbb{R}^n$. Show how this problem can be effiently and stably solved through application of Algorithms 5.3-4 and 7.4-2.

SHOW: (1) Using Algorithm 7.4-2 to find the orthogonal matrix U_0 such that

$$U_0^T A U_0 = H \qquad \text{requires: } \tfrac{5}{3} n^3 \text{ flops}$$

Then

$$U_0^T (A + zI) U_0 = H + (U_0^T I U_0)\cdot z$$

$$= H + z\cdot I \; ; \quad \text{computer } U_0 b \text{ requires: } n^2 \text{ flops}$$

This means that we can always use to same U_0 for all different values of z.

(2) We note that $H + z\cdot I$ is also a Hessenberg matrix. Then use Algorithm 5.3-4 to obtain the upper trangular form. requires: $n^2/2$ flops + $n^2/2$ flops work done on vector b.

(3) Using Back-subtitution to colve the system for $U_0^T x$ requires: $n^2/2$ flops

(4) Final result: $x = U_0 \cdot U_0^T x$: n^2 flops

Exclude the initial work: $\frac{5}{3}$ flops.

The total work for solving each of these systems is about $3.5\, n^2$ flops. Compared to the work required for solving each system seperately ($n^3/6$ flops), this method is effient. ※

P7.4-8 Suppose λ and x are known eigenvalue-eigenvector pair for the upper Hessenberg matrix $H \in \mathbb{R}^{n\times n}$. Give an algorithm for computing orthogonal

$$P^T H P = \begin{bmatrix} \lambda & \omega^T \\ 0 & H_1 \end{bmatrix}$$

where $H_1 \in \mathbb{R}^{(n-1)\times(n-1)}$ is upper Hessenberg. (Hint: P is a product of Jacobi rotations.)

Solution:

Let
$$P = \begin{bmatrix} x_1 & 0 & 0 & \cdots & 0 \\ x_2 & d_2 & u_2 & & \\ x_3 & l_3 & d_3 & \ddots & \\ \vdots & & \ddots & \ddots & u_{n-1} \\ x_n & & & l_n & d_n \end{bmatrix}$$

then

$$P^T = \begin{bmatrix} x_1 & x_2 & x_3 & \cdots & x_n \\ 0 & d_2 & l_3 & & \\ 0 & u_2 & d_3 & \ddots & \\ \vdots & & \ddots & \ddots & l_n \\ 0 & & & u_{n-1} & d_n \end{bmatrix}$$

where $x' = [x_1, x_2, \cdots, x_n]$ is the normalized eigenvector of H associated to λ. That is

$$x^T \cdot x = 1 \quad , \quad x^T H x = \lambda$$

$$P^T \cdot P = \begin{bmatrix} 1 & d_2x_2 + l_3x_3 & u_2x_2 + d_3x_3 + l_4x_4 & \cdots \\ d_2x_2 + l_3x_3 & d_2^2 + l_3^2 & d_2 \cdot u_2 + d_3 l_3 & \cdots \\ u_2x_2 + d_3x_3 + l_4x_4 & d_2u_2 + d_3l_3 & u_2^2 + d_3^2 + l_4^2 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

Let $\begin{cases} d_2 x_2 + l_3 x_3 = 0 \\ d_2^2 + l_3^2 = 1 \end{cases}$

x_2, x_3 are known, then solve for d_2, l_3.

then from $\begin{cases} d_2 u_2 + d_3 l_3 = 0 \\ u_2 x_2 + d_3 x_3 + l_4 x_4 = 0 \\ u_2^2 + d_3^2 + l_4^2 = 1 \end{cases}$

Solving for u_2, d_3, l_4, since x_2, x_3, x_4, d_2, l_3 are known.

And so on. Finally we will have an orthogonal P:

$$P = \begin{bmatrix} x_1 & 0 & 0 & \cdots & 0 \\ x_2 & d_2 & u_2 & \cdots & 0 \\ x_3 & l_3 & d_3 & \cdots & 0 \\ \vdots & & \ddots & \ddots & u_{n-1} \\ x_n & \cdots & & l_n & d_n \end{bmatrix}$$

$$P^T \cdot H \cdot P = \begin{bmatrix} x_1 & x_2 & x_3 & \cdots & x_n \\ 0 & d_2 & l_3 & \cdots & 0 \\ 0 & u_2 & d_3 & \cdots & 0 \\ \vdots & & \ddots & \ddots & l_n \\ 0 & \cdots & & u_{n-1} & d_n \end{bmatrix} \cdot$$

$$\begin{bmatrix} h_{11} & h_{12} & h_{13} & h_{1n} \\ h_{21} & h_{22} & h_{23} & h_{2n} \\ 0 & h_{32} & h_{33} & \cdots h_{3n} \\ & \ddots & \ddots & \\ 0 & & h_{nn-1} & h_{nn} \end{bmatrix} \cdot \begin{bmatrix} x_1 & 0 & 0 & \cdots & 0 \\ x_2 & d_2 & u_2 & & 0 \\ x_3 & l_3 & d_3 & \ddots & \\ \vdots & & \ddots & \ddots & u_{n-1} \\ x_n & 0 & & l_n & d_n \end{bmatrix}$$

$$= \begin{bmatrix} \lambda & \omega_1 & \omega_2 & \cdots & \omega_n \\ 0 & h'_{22} & h'_{23} & \cdots & h'_{2n} \\ 0 & h'_{32} & h'_{33} & \cdots & h'_{3n} \\ 0 & & \ddots & \ddots & \vdots \\ \vdots & 0 & & \ddots & \\ 0 & & & h'_{nn-1} & h'_{nn} \end{bmatrix} = \begin{bmatrix} \lambda & \omega^T \\ 0 & H_1 \end{bmatrix}$$

8.6/10

P7.5-3. Explain how the single-shift QR step $H-\mu I=UR$, $\bar{H}=RU+\mu I$ can be carried out implicitly. That is, show how the transition from H to $\bar{H}$ can be carried out without subtracting the shift μ from the diagonal of H.

Explaination: INITIALIZATION

You may assume that H is upper Hessenberg

Find Q-R decomposition of $H-\mu I$

$$H-\mu I = UR$$

Then repeat following to find U

$$\bar{H} := U^T H U$$

Explaination: Assuming

$$H = \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1n} \\ h_{21} & h_{22} & \cdots & h_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ h_{n1} & h_{n2} & \cdots & h_{nn} \end{bmatrix}$$

(i) Find an orthogonal matrix U_1 so that

How is μ used?

$$U_1 H = \begin{bmatrix} h_{11}' & h_{12}' & \cdots & h_{1n}' \\ 0 & h_{22}' & \cdots & h_{2n}' \\ \vdots & \vdots & \ddots & \vdots \\ 0 & h_{n2}' & \cdots & h_{nn}' \end{bmatrix}$$

−1

Then

$$U_1 H U_1 = \begin{bmatrix} h_{11}'' & h_{12}'' & \cdots & h_{1n}'' \\ h_{21}'' & h_{22}'' & \cdots & h_{2n}'' \\ \vdots & \vdots & \ddots & \vdots \\ h_{n1}'' & h_{n2}'' & \cdots & h_{nn}'' \end{bmatrix}$$

(ii) Find an orthogonal matrix U_2 so that

$$U_2 U_1 H U_1 = \begin{bmatrix} h_{11}''' & h_{12}''' & \cdots & h_{1n}''' \\ h_{21}''' & h_{22}''' & \cdots & h_{2n}''' \\ \vdots & 0 & & \\ h_{n1}''' & 0 & \cdots & h_{nn}''' \end{bmatrix}$$

And so on. Finally Let $U = U_1 U_2 \cdots U_{n-1}$ And

$$\bar{H} = U^T H U$$

P7.5-4. Suppose H is upper Hessenberg and that we compute the factorization $PH = LU$ via Gaussian elimination with partial pivoting. (see Algorithm 5.3-4) Show that $\bar{H} = U(P^T L)$ is upper Hessenberg and similar to H.

SHOW: Since H is upper Hessenberg.

$$H = \begin{bmatrix} x & x & x & x & x \\ x & x & x & x & x \\ 0 & x & x & x & x \\ 0 & 0 & x & x & x \\ 0 & 0 & 0 & x & x \end{bmatrix}$$

We know from theorem given in the lecture that $P^T L$ is 1 bounded. $P^T L$ is of the form

$$P^T L = \begin{bmatrix} x & & & & & \\ x & x & & & 0 & \\ & x & x & & & \\ 0 & & x & x & & \\ & & & x & x & \end{bmatrix}$$

Then

(i) $$\bar{H} = U(P^T L) = \begin{bmatrix} x & x & x & \cdots & x \\ 0 & x & x & \cdots & x \\ 0 & 0 & x & \cdots & x \\ \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & x \end{bmatrix} \cdot \begin{bmatrix} x & 0 & 0 & & \\ x & x & 0 & & \\ 0 & x & x & & \\ & & \ddots & \ddots & \\ & & & x & x \end{bmatrix}$$

$$= \begin{bmatrix} x & x & x & \cdots & x \\ x & x & x & \cdots & x \\ 0 & x & x & \cdots & x \\ 0 & 0 & x & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \\ 0 & 0 & 0 & x & x \end{bmatrix}$$ is upper Hessenberg

(ii) $$PHP^T = LUP^T = L(UP^T) = L\bar{H}L^{-1}$$

$$H = (P^T L)\bar{H}(P^T L)^{-1}$$

Thus H and $\bar{H}$ is similar.

P8.2-1 Let $A = \begin{bmatrix} w & x \\ x & z \end{bmatrix}$ be real and suppose we perform the following shifted QR step:

$$A - zI = UR$$
$$\bar{A} = RU + zI$$

Show that if $\bar{A} = \begin{bmatrix} \bar{w} & \bar{x} \\ \bar{x} & \bar{z} \end{bmatrix}$ then

$$\bar{w} = w + x^2(w-z)/[(w-z)^2 + x^2]$$
$$\bar{z} = z - x^2(w-z)/[(w-z)^2 + x^2]$$
$$\bar{x} = -x^3/[(w-z)^2 + x^2]$$

SHOW:

$$A - zI = \begin{bmatrix} w-z & x \\ x & 0 \end{bmatrix}$$

Let $U^T = \dfrac{1}{[(w-z)^2 + x^2]^{1/2}} \begin{bmatrix} w-z & x \\ -x & w-z \end{bmatrix}$

$$R = U^T(A - zI) = \frac{1}{[(w-z)^2+x^2]^{1/2}} \cdot \begin{bmatrix} (w-z)^2+x^2 & x(w-z) \\ 0 & -x^2 \end{bmatrix}$$

Then

$$\bar{A} = \frac{1}{[(w-z)^2 + x^2]} \begin{bmatrix} (w-z)^2+x^2 & x(w-z) \\ 0 & -x^2 \end{bmatrix} \cdot \begin{bmatrix} w-z & -x \\ x & w-z \end{bmatrix}$$

$$= \frac{1}{(w-z)^2 + x^2} \cdot \begin{bmatrix} [(w-z)^2+x^2]\cdot(w-z)+x^2(w-z) & -x^3 \\ -x^3 & -x^2(w-z) \end{bmatrix} + zI$$

$$= \begin{bmatrix} [(w-z)^2+x^2](w-z+z)+x^2(w-z) & -x^3 \\ -x^3 & [(w-z)^2+x^2]\cdot z - x^2(w-z) \end{bmatrix} \frac{1}{(w-z)^2+x^2}$$

$$= \begin{bmatrix} w + x^2(w-z)/[(w-z)^2+x^2] & -x^3/[(w-z)^2+x^2] \\ -x^3/[(w-z)^2+x^2] & z - x^2(w-z)/[(w-z)^2+x^2] \end{bmatrix}$$

#

P8.2-3 Suppose $A \in \mathbb{R}^{n\times n}$ is skew symmetric ($A^T = -A$). Show how construct Householder matrices $P_1, \cdots, P_{n-2}$ such that $(P_1 \cdots P_{n-2})^T A (P_1 \cdots P_{n-2})$ is tridiagonal. How many flops are required by your algorithm.

Solution:

$$A^T = -A \Rightarrow \text{all diagonal elements in } A = 0$$

$$A = \begin{bmatrix} 0 & x & x & \cdots & x \\ -x & 0 & x & \cdots & x \\ -x & -x & 0 & \cdots & x \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -x & -x & -x & \cdots & 0 \end{bmatrix}$$

Then when we creat a P_1, so that

$$P_1^T \cdot A = \begin{bmatrix} 0 & x & x & \cdots & x \\ -x & x & \ddots & & x \\ 0 & -x & & \ddots & \\ \vdots & \vdots & & & \\ 0 & \vdots & x & & x \end{bmatrix}$$

And then applied P_1 to left side of $P_1^T A$, we have

$$P_1^T A P_1 = \begin{bmatrix} 0 & x & 0 & \cdots & 0 \\ -x & 0 & x & \cdots & x \\ 0 & x & 0 & & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & x & x & \cdots & 0 \end{bmatrix}$$

Since $(P_1^T A P_1)^T = P_1^T \cdot A^T \cdot P_1 = -(P_1^T A P_1)$, so

$P_1^T A P_1$ is also a skew symmetric matrix.

The algorithm is given in next page.

P8.2-3 (CONT.)

Algorithm:

For $i = 1$ to $n-2$

1. Construct P_i^T according to Algorithm 3.3-1 on p.40 in text for $a_{i+1,i}, \cdots, a_{ni}$. requires $2(n-i)$ flops

2. Update A with Algorithm 3.3-2 on page 41 in text to perform $P_i^T A$. requires $2n(n-i)$ flops

3. Following algorithm compute $P_i^T A \cdot P_i$

$\frac{3}{2}(n^2+i^2) - 3ni$ flops

For $p = i, \cdots, n-2$

$s := v_{i+1} \cdot a_{p\,i+1} + \cdots + v_n \cdot a_{pn}$ } $(n-i)$ flops

$s := \beta s$

For $k = p+1, \cdots, n$

$a_{pk} := a_{pk} - s \cdot v_k$

$a_{kp} := -a_{pk}$

} $(n-p)$ flops

-.2

$$\text{Total flops} \cong \sum_{i=1}^{n-2} \left[2n^2 - 2ni + \frac{3}{2}n^2 + \frac{3}{2}i^2 - 3ni \right]$$

$$\cong \frac{3}{2} n^3 + O(n^2) \text{ flops}$$

Use algorithm 8.2-1 to reduce work to $2n^3/3$.

P8.2-5 Show that if $A = B + i\cdot C$ is Hermitian, then $M = \begin{bmatrix} B & -C \\ C & B \end{bmatrix}$ is symmetric. Relate the eigenvalues and eigenvectors of A and M.

SHOW: A is Hermitian, then

$$A^T = B^T + iC^T = \bar{A} = B - iC$$

We have

$$B^T = B \quad \text{and} \quad C^T = -C$$

$$M^T = \begin{bmatrix} B^T & C^T \\ -C^T & B^T \end{bmatrix} = \begin{bmatrix} B & -C \\ C & B \end{bmatrix} = M$$

Thus, M is symmetric.

Assume that λ is eigenvalue of A and x is an eigenvector associated to it, then

$$Ax = Bx + iCx = \lambda x$$

Then

$$\begin{bmatrix} B & -C \\ C & B \end{bmatrix} \cdot \begin{bmatrix} x \\ -ix \end{bmatrix} = \begin{bmatrix} Bx + iCx \\ Cx - iBx \end{bmatrix} = \begin{bmatrix} \lambda x \\ -i(Bx + iCx) \end{bmatrix}$$

$$= \begin{bmatrix} \lambda x \\ -i\cdot\lambda x \end{bmatrix} = \lambda \cdot \begin{bmatrix} x \\ -ix \end{bmatrix}$$

This implies that λ is also a eigenvalue of M with an eigenvector $[x^T \ -ix^T]^T$ associated to it.

This also means that all eigenvalues of A are real. Because all eigenvelues of M are real due to symmetric M.

*

P8.3-2 Give formulae for the eigenvectors of $\begin{bmatrix} 0 & A^T \\ A & 0 \end{bmatrix}$ in terms of the singular vectors of $A \in \mathbb{R}^{m\times n}$, where $m \geq n$

Assuming A has the singular value decomposition

$$A = U\Sigma V^T \quad , \quad U = [u_1, \cdots, u_m] \ \& \ V = [v_1, v_2, \cdots, v_n]$$

where

$$\Sigma = \begin{array}{cc} n & \\ \begin{bmatrix} \sigma_1 & & & 0 \\ & \sigma_2 & & \\ & & \ddots & \sigma_n \\ & 0 & & \end{bmatrix} & m \end{array}$$

According Eq. (8.3-3) in text, we have following:

Let $U = [\underset{n}{U_1} \quad \underset{m-n}{U_2}]$ and

$$Q = \frac{1}{\sqrt{2}} \begin{bmatrix} V & V & 0 \\ U_1 & -U_1 & \sqrt{2}\,U_2 \end{bmatrix} \quad \text{is orthogonal}$$

Then we have

$$Q^T \begin{bmatrix} 0 & A^T \\ A & 0 \end{bmatrix} Q = \operatorname{diag}(\sigma_1, \cdots, \sigma_n, -\sigma_1, \cdots, -\sigma_n, 0, \cdots, 0)$$

Rewrite the equation above, we get

$$\begin{bmatrix} 0 & A^T \\ A & 0 \end{bmatrix} Q = Q \cdot \operatorname{diag}(\sigma_1, \cdots, \sigma_n, -\sigma_1, \cdots, -\sigma_n, 0, \cdots, 0)$$

Thus, we obtain the all eigenvalues of $\begin{bmatrix} 0 & A^T \\ A & 0 \end{bmatrix}$:

$$\sigma_1, \sigma_2, \cdots, \sigma_n, -\sigma_1, \cdots, -\sigma_n, 0, \cdots, 0$$

And the eigenvectors associated to these eigenvalues, respectively,

$$[g_1, g_2, \cdots, g_{n+m}] = Q = \frac{1}{\sqrt{2}} \begin{bmatrix} V & V & 0 \\ U_1 & -U_1 & \sqrt{2}\,U_2 \end{bmatrix}$$

OR

$$g_i = \begin{cases} \frac{1}{\sqrt{2}} \begin{bmatrix} v_i \\ u_i \end{bmatrix} & \text{for } i \leq n \\ \frac{1}{\sqrt{2}} \begin{bmatrix} v_{i-n} \\ -u_{i-n} \end{bmatrix} & \text{for } n < i \leq 2n \\ \begin{bmatrix} 0 \\ u_{i-n} \end{bmatrix} & \text{for } 2n < i \leq m+n \end{cases}$$

P8.5-4 How many positive eigenvalues does the following tridiagonal matrix have?

$$A = \begin{bmatrix} 2 & 1 & & & \\ 1 & 0 & 3 & & \\ & 3 & 4 & 1 & \\ & & 1 & 1 & 2 \\ & & & 2 & 7 \end{bmatrix}$$

Solution: From the given, we have $n = 5$ and

$$a_1 = 2, \quad a_2 = 0, \quad a_3 = 4, \quad a_4 = 1, \quad a_5 = 7$$

$$b_2 = 1, \quad b_3 = 3, \quad b_4 = 1, \quad b_5 = 2$$

Then,

$$P_0(x) \equiv 1 \quad ; \quad P_0(0) = 1$$

$$P_1(x) = 2 - x \quad ; \quad P_1(0) = 2$$

$$P_2(x) = (-x)(2-x) - 1 \cdot 1 = x^2 - 2x - 1 \quad ; \quad P_2(0) = -1$$

$$P_3(x) = (4-x)(x^2-2x-1) - 9(2-x)$$

$$P_3(0) = -22$$

$$P_4(0) = 1 \cdot (-22) - 1 \cdot (-1) = -21$$

$$P_5(0) = 7 \cdot (-21) - 4 \cdot (-22) = -59$$

$$\therefore \{P_0(0), P_1(0), \cdots P_5(0)\} = \{1, 2, -1, -22, -21, -59\}$$

Then $\alpha(0) = 1$, That is

We have $5 - 1 = 4$ positive eigenvalues for A.

P 8.6-1 Suppose $A \in \mathbb{R}^{n \times n}$ is symmetric and $G \in \mathbb{R}^{n \times n}$ is lower triangular and non-singular. Give an algorithm for computing $C = G^{-1} A G^{-T}$ in $\frac{n^3}{3}$ flops.

Solution: Since A is symmetric, we can rewrite A as

$$A = B + B^T \quad \text{where } B \text{ is lower triangular}$$

Then

$$C = G^{-1} A G^{-T} = G^{-1}(B + B^T) G^{-T}$$

$$= G^{-1} \cdot B \cdot G^{-T} + G^{-1} B^T G^{-T}$$

$$= G^{-1} B G^{-T} + (G^{-1} B G^{-T})^T \triangleq D + D^T$$

where $D = G^{-1} B G^{-T}$

$$G D G^T = B$$

G and B are lower triangular, so is DG^T

(i) Solving $G \cdot X = B$. Then

$DG^T = X$, X is lower triangular.

$GD^T = X^T$

(ii) solving $GD^T = X^T$, we get D^T

(iii) $C = D + D^T$

The following algorithm follows these procedures.

P8.6-1 (CONT.) *Algorithm*

$n^3/3$ flops:

For $k = 1, \cdots, n$
  $a_{kk} := a_{kk}/2$
  for $i = k, \cdots, n$
    for $j = 1, \cdots, i-1$
      $a_{ik} := a_{ik} - g_{ij} * a_{jk}$
    $a_{ik} := a_{ik} / g_{ii}$

$n^3/2$ flops:

for $k = 1, \cdots, n$
  for $i = 1, \cdots, n$
    if $i > k$ then $a_{ki} := 0$ else $a_{ki} := a_{ik}$
    for $j = 1, \cdots, i-1$
      $a_{ki} := a_{ki} - g_{ij} * a_{kj}$
    $a_{ki} := a_{ki} / g_{ii}$

$O(n^2)$ flops:

for $i = 1, \cdots, n$
  for $j = 1, \cdots, n$
    $c_{ij} := a_{ij} + a_{ji}$

Note : (i) Unfortunately, it takes $5n^3/6$ flops in my algorithm to compute

$$C = G^{-1} A G^{-T}.$$

(ii) If we compute $A = LDL^T$ first ($n^3/6$ flops), then

$$C = (G^{-1}L) \cdot D \cdot (G^{-1}L)^T$$

And it requires $n^3/3$ flops for computing $G^{-1}L$, And then it takes about $n^3/2$ more to finish the computations. (Total n^3 flops)

Good try

$-.2$