CS137 – Introduction to Scientific Computing
Winter Quarter 2004
Solutions to Homework #3

Felix Kwok

February 27, 2004

Written Problems

1. (Heath E3.10) Let B be an $n \times n$ matrix, and assume that B is both orthogonal and triangular.
   (a) Prove that B must be diagonal.
   (b) What are the diagonal entries of B?
   (c) Let A be $n \times n$ and non-singular. Use parts (a) and (b) to prove that the QR factorization of A is unique up to the signs of the diagonal entries of R. In particular, show that there exist unique matrices Q and R such that Q is orthogonal, R is upper triangular with positive entries on its main diagonal, and $A = QR$.

Solution. (a) We need two facts:

i. P,Q orthogonal $\implies$ PQ orthogonal
ii. P upper triangular $\implies$ P^{-1} upper triangular

(i) is easy: $(PQ)^T(PQ) = Q^T P^T PQ = Q^T IQ = Q^T Q = I$.
(ii) can be argued by considering the kth column p_k of P^{-1}. Then $P^{-1}e_k = p_k$, so $Pp_k = e_k$. Since P is upper triangular, we can use backward substitution to solve for p_k. Recall the formula for backward substitution for a general upper triangular system $Tx = b$:

$$x_k = \frac{1}{t_{kk}} \left(b_k - \sum_{j=k+1}^{n} t_{kj} b_j \right).$$

In our case, we see that since $(e_k)_j = 0$ for $j > k$, this implies $(p_k)_j = 0$ for $j > k$. In other words, the kth column of P^{-1} must be all zero below the kth row. This is to say P^{-1} is upper triangular.

Now we show that B both orthogonal and triangular implies B diagonal. Without loss of generality (i.e. replacing B by B^T if necessary), we can assume B is upper triangular. Then B^{-1} is also upper triangular by (ii). But $B^{-1} = B^T$ by orthogonality, and we know B^T is lower triangular. So B^T (and hence B) is both upper and lower triangular, implying that B is in fact diagonal.

(b) We know B is diagonal, so that $B = B^T$. Let $B = \mathrm{diag}(b_{11}, \ldots, b_{nn})$. Then $I = BB^T = \mathrm{diag}(b_{11}^2, \ldots, b_{nn}^2)$, so $b_{ii}^2 = 1$ for all i. Thus, the diagonal entries of B are ± 1.

(c) There is an existence part and a uniqueness part to this problem. In class we showed existence of a decomposition $A = QR$ where Q is orthogonal, and R is upper triangular, but not necessarily with positive entries on the diagonal, so we have to fix it up. We know that $r_{ii} \neq 0$ (otherwise R would be singular, contradicting the non-singularity of A), so we can define

$$D = \mathrm{diag}(\mathrm{sgn}(r_{11}), \ldots, \mathrm{sgn}(r_{nn})),$$

where

$$\operatorname{sgn}(x) = \begin{cases} 1, & x > 0, \\ 0, & x = 0, \\ -1, & x < 0. \end{cases}$$

Note that D is orthogonal, and $\tilde{R} = DR$ has positive diagonal. So if we define $\tilde{Q} = QD$ (orthogonal), then $A = \tilde{Q}\tilde{R}$ gives the required decomposition, so we have proved existence.
For uniqueness, suppose $A = Q_1R_1 = Q_2R_2$ are two such decompositions. Then $Q_2^TQ_1 = R_2R_1^{-1}$, so that the left-hand side is orthogonal and the right-hand side is upper triangular. By parts (a) and (b), this implies both sides are equal to a diagonal matrix with ± 1 as the only possible entries. But both R_1 and R_2 has positive diagonal, so $R_2R_1^{-1}$ must have positive diagonal (you should verify that for upper triangular matrices, the diagonal of the inverse is the inverse of the diagonal, and the diagonal of the product is the product of the diagonals). Thus, both sides are equal to a diagonal matrices with $+1$ on the diagonal, i.e. the identity. So $Q_2^TQ_1 = I \implies Q_1 = Q_2$, and $R_2R_1^{-1} = I \implies R_1 = R_2$. This shows uniqueness. □

2. Let A be a real matrix. If there exists an orthogonal matrix Q such that $A = Q\begin{bmatrix} R \\ 0 \end{bmatrix}$, show that

$$A^TA = R^TR.$$

Solution. This simply involves multiplying the block matrices out:

$$A^TA = \begin{bmatrix} R^T & 0 \end{bmatrix} Q^TQ \begin{bmatrix} R \\ 0 \end{bmatrix} = \begin{bmatrix} R^T & 0 \end{bmatrix} \begin{bmatrix} R \\ 0 \end{bmatrix} = R^TR.$$

□

3. Let A be symmetric positive definite. Given an initial guess x_0, the method of steepest descent for solving $Ax = b$ is defined as

$$x_{k+1} = x_k + \alpha_k r_k,$$

where $r_k = b - Ax_k$ and α_k is chosen to minimize $f(x_{k+1})$, where

$$f(x) := \frac{1}{2}x^TAx - x^Tb.$$

(a) Show that

$$\alpha_k = \frac{r_k^Tr_k}{r_k^TAr_k}.$$

(b) Show that

$$r_{i+1}^Tr_i = 0.$$

(c) If $e_i := x - x_i \neq 0$, show that

$$e_{i+1}^TAe_{i+1} < e_i^TAe_i.$$

Hint: This is equivalent to showing $r_{i+1}^TA^{-1}r_{i+1} < r_i^TA^{-1}r_i$ (why?).

Solution. (a) To find out what α_k is, apply the recurrence relation to the definition of $f(x_{k+1})$:

$$\begin{aligned} f(x_{k+1}) &= \frac{1}{2}(x_k + \alpha_k r_k)^TA(x_k + \alpha_k r_k) - (x_k + \alpha_k r_k)^Tb \\ &= \frac{1}{2}(x_k^TAx_k + \alpha_k r_k^TAx_k + \alpha_k x_k^TAr_k + \alpha_k^2 r_k^TAr_k) - x_k^Tb - \alpha_k r_k^Tb \\ &= \frac{1}{2}(x_k^TAx_k + \alpha_k r_k^TAx_k + \alpha_k r_k^TA^Tx_k + \alpha_k^2 r_k^TAr_k) - x_k^Tb - \alpha_k r_k^Tb \\ &= \frac{1}{2}(x_k^TAx_k + \alpha_k r_k^TAx_k + \alpha_k r_k^TAx_k + \alpha_k^2 r_k^TAr_k) - x_k^Tb - \alpha_k r_k^Tb \\ &= \frac{1}{2}(x_k^TAx_k + 2\alpha_k r_k^TAx_k + \alpha_k^2 r_k^TAr_k) - x_k^Tb - \alpha_k r_k^Tb. \end{aligned}$$

Now differentiate with respect to α_k and set to zero:

$$\frac{d}{d\alpha_k} f(x_{k+1}(\alpha_k)) = r_k^T A x_k + \alpha r_k^T A r_k - r_k^T b = 0$$
$$\alpha_k r_k^T A r_k = r_k^T b - r_k^T A x_k = r_k^T r_k.$$

Isolating α_k yields the desired result. Note that $\alpha > 0$, since A is positive definite.

(b) It helps to first establish a recurrence relation involving the residuals only: note that

$$rk + 1 = b - Ax_{k+1} = b - A(x_k + \alpha_k r_k) = r_k - \alpha_k A r_k.$$

Then

$$\begin{aligned} r_{k+1}^T r_k &= (r_k - \alpha_k A r_k)^T r_k \\ &= r_k^T r_k - \left(\frac{r_k^T r_k}{r_k^T A r_k} \right) r_k^T A^T r_k \\ &= 0 \end{aligned}$$

since $A = A^T$.

(c) First, note that

$$r_k = b - Ax_k = Ax - Ax_k = A(x - x_k) = Ae_k,$$

so that $r_k^T A^{-1} r_k = e_k^T A^T A^{-1} A e_k = e_k^T A e_k$, and similarly for r_{k+1}. Thus, the two inequalities are completely equivalent. Now we attack the inequality involving r:

$$\begin{aligned} r_{k+1}^T A^{-1} r_{k+1} &= r_{k+1}^T A^{-1} (r_k - \alpha_k A r_k) \\ &= r_{k+1}^T A^{-1} r_k - \underbrace{\alpha_k r_{k+1}^T r_k}_{0 \text{ by (b)}} \\ &= (r_k - \alpha_k A r_k)^T A^{-1} r_k \\ &= r_k A^{-1} r_k - \underbrace{\alpha_k r_k^T r_k}_{>0} \\ &< r_k A^{-1} r_k. \end{aligned}$$

□

Computer Problems

4. Let A be an $n \times n$ matrix of the form

$$A = \begin{bmatrix} 2+\sigma_1 h^2 & -1 & & & & & \\ -1 & 2+\sigma_2 h^2 & -1 & & & & \\ & -1 & \ddots & \ddots & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & \ddots & \ddots & -1 & \\ & & & & -1 & 2+\sigma_n h^2 \end{bmatrix}.$$

(a) Let $n = 25$, $h = 1/26$, and

$$\sigma_i = \begin{cases} 100, & i = 1, \ldots, 13, \\ 1 & i = 14, \ldots, 25. \end{cases}$$

Implement the conjugate gradient method and use it to solve the problem $Ax = h^2 b$, where b is a random vector with entries between -1 and 1. Use $x^{(0)} = 0$ as your starting vector, and iterate

until the relative residual $\|r^{(k)}\|_2/\|b\|_2$ is less than 10^{-5}. Plot the quantities $\|e^{(i)}\|_2$, $\|r^{(i)}\|_2$ and $\|e^{(i)}\|_A$. Which of these quantities do you expect to be monotonically decreasing? (Recall that $\|e^{(i)}\|_A := \left(e^{(i)T} A e^{(i)}\right)^{1/2}$).

(b) Transform the matrix problem into

$$(DAD)y = h^2 Db,$$

where $x = Dy$, and D is a diagonal matrix with

$$d_{ii} = (2 + \sigma_i h^2)^{-1/2}.$$

Repeat part (a) on the transformed problem. For which problem does CG converge faster?

(c) For this part assume $\sigma_i = 0$ for all i. Let

$$M = \begin{bmatrix} 1 & -1 & & & \\ -1 & 2 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -1 \\ & & & -1 & 2 \end{bmatrix}.$$

Apply conjugate gradients with M as a preconditioner as follows. First, compute the Cholesky factorization $M = LL^T$, and then solve the modified problem

$$(L^{-1} A L^{-T})y = L^{-1} b,$$

where $y = L^T x$. How many iteration does CG require to converge? Note that it is not necessary to compute L^{-1} and form $L^{-1}AL^{-T}$ explicitly. What should be done instead?

Solution. (a) The following MATLAB code implements the preconditioned conjugate gradient method with preconditioner M, where $M = P^T P$.

```
function [x,R] = mypcg(A,b,P,x0,tol)
% [x,R] = MYPCG(A,b,P,x0,tol)
% Solves (inv(P')*A*inv(P))(P*x) = inv(P')*b, where A is symmetric
% positive definite, using the conjugate gradient method. R(:,i) is the
% (i-1)st residual vector.

r = (P')\(b - A*x0);
initres = norm(r);
tol = tol * initres;
s = r;
y = P*x0;
count = 0;
R(:,1) = r;
while (norm(r) > tol),
    count = count + 1;
    As = (P')\(A*(P\s));
    alpha = (r'*r)/(s'*As);
    y = y + alpha * s;
    rnew = r - alpha * As;
    beta = (rnew'*rnew)/(r'*r);
    r = rnew;
    s = r + beta*s;
    R(:,count+1) = r;
end;
x = P\y;
disp(['CG converged in ',num2str(count),' iterations.']);
```

Figure 1 shows the results for the first test case. We get convergence in 25 iterations, as is predicted by the theory. We also see that $\|e^{(i)}\|_A$ is monotonically decreasing, as expected. The other two quantities need not decrease monotonically (e.g. see the curve for $\|r^{(i)}\|_2$).

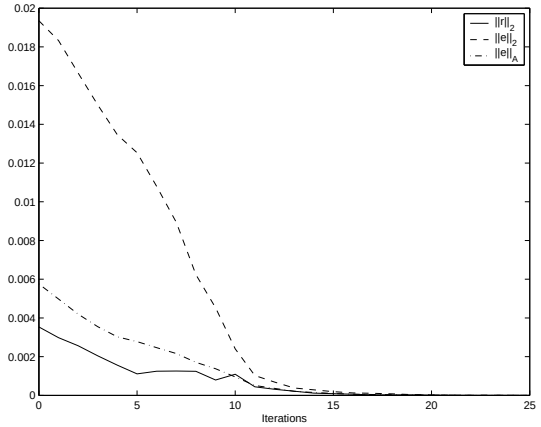

Figure 1: Error plots for 4(a).

(b) With diagonal scaling, we also get convergence in 25 iterations, i.e. no improvement. This is likely because the variation in the diagonal is too insignificant for diagonal scaling to have any effect. The error and residual norms are shown in Figure 2.

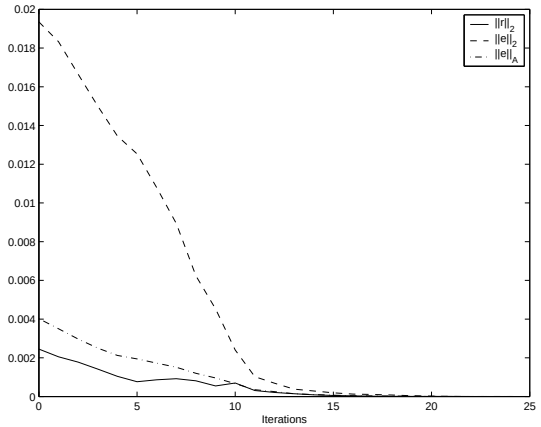

Figure 2: Error plots for 4(b).

(c) Here we expect the convergence to be in two steps because M is a rank-one change from A, and this is confirmed by the error plots in Figure 3. Note that in the CG code, it is not necessary to compute L^{-1} and form $L^{-1}AL^{-T}$ explicitly. Instead, there are two solves and a matrix-vector multiply at the beginning of each iteration, and these solves are generally implemented as a forward/backward substitution (since L is triangular).

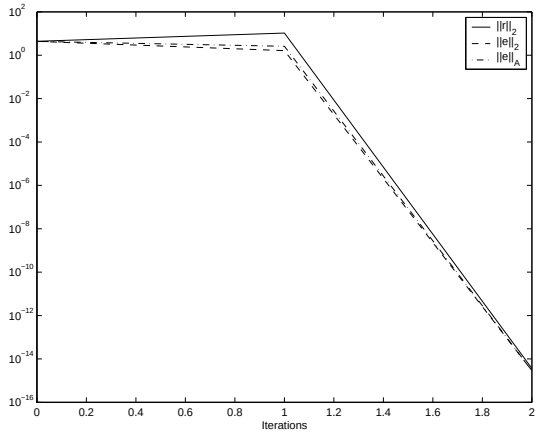

Figure 3: Error plot for 4(c).

□

5. We are given the following data for the total population of the United States, as determined by the U. S. Census, for the years 1900 to 2000. The units are millions of people.

t	y
1900	75.995
1910	91.972
1920	105.711
1930	123.203
1940	131.669
1950	150.697
1960	179.323
1970	203.212
1980	226.505
1990	249.633
2000	281.422

Suppose we model the population growth by

$$y \approx \beta_1 t^3 + \beta_2 t^2 + \beta_3 t + \beta_4.$$

(a) Use normal equations for computing β. Plot the resulting polynomial and the exact values $\mathbf{y}$ in the same graph.

(b) Use the QR factorization to obtain β for the same problem. Plot the resulting polynomial and the exact values, as well as the polynomial in part (a), in the same graph. Also compare your coefficients with those obtained in part (a).

(c) Suppose we translate and scale the time variable t by

$$s = (t - 1950)/50$$

and use the model

$$y \approx \beta_1 s^3 + \beta_2 s^2 + \beta_3 s + \beta_4.$$

Now solve for the coefficients β and plot the polynomial and the exact values in the same graph. Which of the polynomials in part (a) through (c) gives the best fit to the data?

Solution. (a) The least squares problem is

$$\min \|y - A\beta\|$$

where

$$A = \begin{bmatrix} t_1^3 & t_1^2 & t_1 & 1 \\ \vdots & & & \vdots \\ t_n^3 & t_n^2 & t_n & 1 \end{bmatrix},$$

$n = 11$, $t_i = 1900 + 10(i-1)$. We form the normal equations $A^T A\beta = A^T y$ and solve for β to obtain

```
beta =

    1.010415596011712e-005
   -4.961885780449666e-002
    8.025770365215973e+001
   -4.259196447217581e+004
```

The plot is shown in Figure 4. Here we have $\|y - A\beta\|_2 = 10.1$.

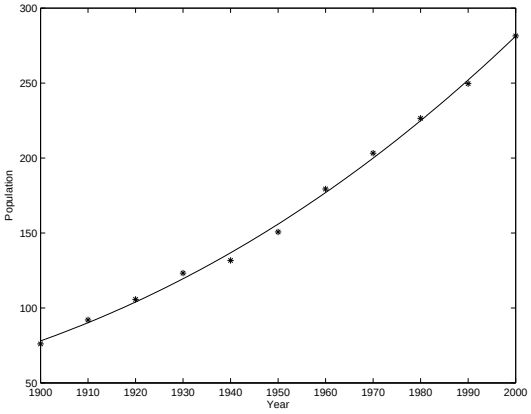

Figure 4: Least squares fit for 5(a).

(b) Now we use the QR factorization to obtain `beta2`:

```
>> [Q,R] = qr(A);
>> b = Q'*y;
>> beta2 = R(1:4,1:4)\b(1:4)

beta2 =

    1.010353535409117e-005
   -4.961522727598729e-002
    8.025062525889830e+001
   -4.258736497384948e+004
```

So $\|\texttt{beta2} - \texttt{beta}\|_2 = 4.60$. The norm of the residual is essentially the same as in part (a), with a difference of 1.08×10^{-10}. This shows the least-squares problem is very ill-conditioned, since a small change in the residual yields a relatively large change in the solution vector. However, Figure 5 shows that the two fits are hardly distinguishable on a graph (mostly because the residuals are both small).

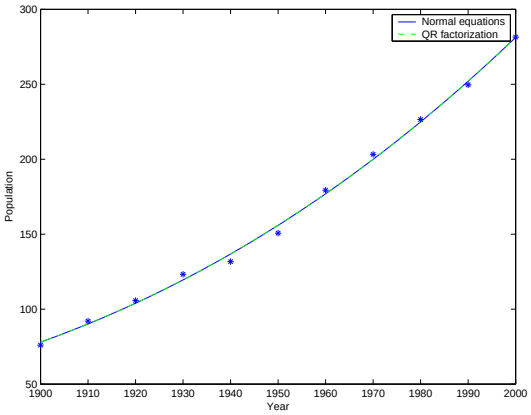

Figure 5: Least squares fit for 5(b).

(c) We now perform a change of the independent variable

$$s = (t - 1950)/50$$

so that $s \in [-1, 1]$. We again set up the least square system and solve:

```
>> [Q,R] = qr(A);
>> b = Q'*b;
>> b = Q'*y;
>> beta3 = R(1:4,1:4)\b(1:4);
>> beta3

beta3 =

    1.262941919191900e+000
    2.372613636363639e+001
    1.003659217171717e+002
    1.559042727272727e+002
```

Note that the coefficients are different because we are using a different basis. The residual is again essentially the same as parts (a) and (b) (the difference is 1.54×10^{-11}). So even though the coefficients are quite different, the quality of the fit is essentially the same (even though (c) is more accurate by just a tiny bit). □

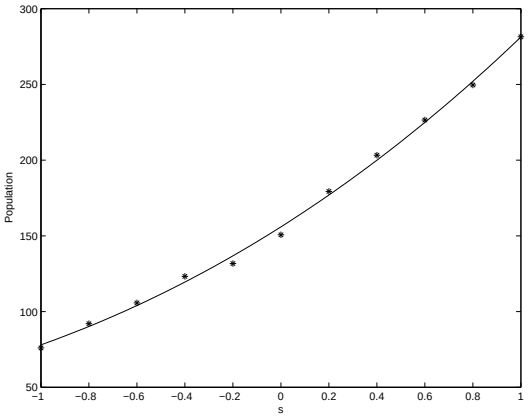

Figure 6: Least squares fit for 5(c).