L. Vandenberghe
ECE133A

5/1/2021

Homework 4 solutions

1. *Exercise A5.6 (b).* Denote the columns of X by X_1, X_2, $\ldots$, X_n, and the columns of B by B_1, B_2, $\ldots$, B_n. If we write the equation in terms of the columns of X and B we get

$$\begin{array}{rcl} LX_1 + L_{11}X_1 &=& B_1 \\ LX_2 + L_{21}X_1 + L_{22}X_2 &=& B_2 \\ LX_3 + L_{31}X_1 + L_{32}X_2 + L_{33}X_3 &=& B_2 \\ &\vdots& \\ LX_n + L_{n1}X_1 + L_{n2}X_2 + \cdots + L_{nn}X_n &=& B_n. \end{array}$$

In other words,

$$\begin{array}{rcl} (L + L_{11}I)X_1 &=& B_1 \\ (L + L_{22}I)X_2 &=& B_2 - L_{21}X_1 \\ (L + L_{33}I)X_3 &=& B_2 - L_{31}X_3 - L_{32}X_2 \\ &\vdots& \\ (L + L_{nn}I)X_n &=& B_n - L_{n1}X_1 - L_{n2}X_2 - \cdots - L_{n,n-1}X_{n-1}. \end{array}$$

We can solve for X_1 first, then evaluate the right-hand side of the second equation and solve for X_2, then substitute X_1 and X_2 in the third right-hand side and solve for X_3, et cetera. The matrices on the left-hand sides are upper triangular with nonzero diagonal elements (since we are given that $L_{ii} + L_{jj} \neq 0$ for all i and j), so each of the n equations can be solved by forward substitution.

To estimate the cost we look at the equation for the kth column:

$$(L + L_{kk}I)X_k = B_k - L_{k1}X_1 - L_{k2}X_2 - \cdots - L_{k,k-1}X_{k-1}.$$

The vector on the right-hand side can be computed in $2(k-1)n$ operations: $(k-1)n$ for the scalar-vector multiplications and $(k-1)n$ for the vector subtractions. Adding L_{kk} to the diagonal elements of L takes n flops. Solving the equation costs n^2. The complexity of step k is therefore $n^2 + (2k-1)n$ flops. The total complexity for the n columns is

$$\begin{array}{rcl} n^3 + n\displaystyle\sum_{k=1}^{n}(2k-1) &=& n^3 + n(1 + 3 + 5 + \cdots + (2n-1)) \\ &=& 2n^3. \end{array}$$

2. *Exercise A6.3.*

   (a) The result in the hint follows from

   $$x^T Sx = (x^T Sx)^T = x^T S^T x = -x^T Sx,$$

   hence $x^T Sx = 0$. This can also be seen by noting that

   $$x^T Sx = \sum_{i=1}^n \sum_{j=1}^n S_{ij} x_i x_j = \sum_{i=1}^n (S_{ii} x_i^2 + \sum_{j=i+1}^n (S_{ij} + S_{ji}) x_i x_j).$$

   Since $S_{ii} = 0$ and $S_{ij} = -S_{ji}$ for a skew-symmetric matrix the sum is zero.
   To prove that $I - S$ is nonsingular we show that $(I-S)x = 0$ implies $x = 0$:

   $$\begin{aligned} (I-S)x = 0 &\implies x^T x - x^T Sx = x^T (I-S)x = 0 \\ &\implies x^T x = 0 \\ &\implies x = 0. \end{aligned}$$

   (b) We have (for any matrix S)

   $$(I-S)(I+S) = I - S + S - S^2 = (I+S)(I-S).$$

   Multiplying with $(I-S)^{-1}$ on both sides gives

   $$(I+S)(I-S)^{-1} = (I-S)^{-1}(I+S).$$

   (c) We show that $A^T A = I$:

   $$\begin{aligned} A^T A &= (I-S)^{-T}(I+S)^T(I+S)(I-S)^{-1} \\ &= (I+S)^{-1}(I-S)(I+S)(I-S)^{-1} \\ &= (I+S)^{-1}(I-S)(I-S)^{-1}(I+S) \\ &= I. \end{aligned}$$

   On line 2 we use the skew-symmetry of S. Line 3 follows from part (b).

3. *Exercise A6.9.*

   (a) For $k = 1$, the result is obvious because S^{k-1} and $\mathbf{diag}(We_1)$ are both equal to the $n \times n$ identity matrix.
   The columns of WS^{k-1} are the columns of W, shifted circularly to the left over $k-1$ positions. For $2 \le k \le n$, this gives

   $$\begin{aligned} &WS^{k-1} \\ &= \begin{bmatrix} 1 & \cdots & 1 & 1 & 1 & \cdots & 1 \\ \omega^{-(k-1)} & \cdots & \omega^{-(n-1)} & 1 & \omega^{-1} & \cdots & \omega^{-(k-2)} \\ \omega^{-2(k-1)} & \cdots & \omega^{-2(n-1)} & 1 & \omega^{-2} & \cdots & \omega^{-2(k-2)} \\ \vdots & & \vdots & \vdots & \vdots & & \vdots \\ \omega^{-(n-1)(k-1)} & \cdots & \omega^{-(n-1)(n-1)} & 1 & \omega^{-(n-1)} & \cdots & \omega^{-(n-1)(k-2)} \end{bmatrix}. \end{aligned}$$

The column of ones on the right-hand side is column $n-k+2$.

The kth column of W is the vector $We_k = (1, \omega^{-(k-1)}, \omega^{-2(k-1)}, \ldots, \omega^{-(n-1)(k-1)})$, so the matrix $\mathbf{diag}(We_k)W$ is

$$\mathbf{diag}(We_k)W =$$
$$\begin{bmatrix} 1 & \cdots & 1 & 1 & 1 & \cdots & 1 \\ \omega^{-(k-1)} & \cdots & \omega^{-(n-1)} & \omega^{-n} & \omega^{-(n+1)} & \cdots & \omega^{-(n+k-2)} \\ \omega^{-2(k-1)} & \cdots & \omega^{-2(n-1)} & \omega^{-2n} & \omega^{-2(n+1)} & \cdots & \omega^{-2(n+k-2)} \\ \vdots & & \vdots & \vdots & \vdots & & \vdots \\ \omega^{-(n-1)(k-1)} & \cdots & \omega^{-(n-1)(n-1)} & \omega^{-(n-1)n} & \omega^{-(n-1)(n+1)} & \cdots & \omega^{-(n-1)(n+k-2)} \end{bmatrix}.$$

The column

$$(1, \omega^{-n}, \omega^{-2n}, \ldots, \omega^{-(n-1)n})$$

on the right-hand side is column $n-k+2$ of this matrix.

Comparing the expressions for WS^{k-1} and $\mathbf{diag}(We_k)W$, we see that the two matrices are equal because $\omega^{-n} = 1$.

(b) Column k of $T(a)$ is

$$\begin{aligned} S^{k-1}a &= \frac{1}{n} W^H \, \mathbf{diag}(We_k)Wa \\ &= \frac{1}{n} W^H \left((We_k) \circ (Wa)\right) \\ &= \frac{1}{n} W^H \, \mathbf{diag}(Wa)We_k. \end{aligned}$$

On the second line we use $u \circ v$ to denote the componentwise vector product (lecture 1, page 1.13). Therefore

$$\begin{aligned} T(a) &= \begin{bmatrix} a & Sa & S^2a & \cdots & S^{n-1}a \end{bmatrix} \\ &= \frac{1}{n} W^H \, \mathbf{diag}(Wa) \begin{bmatrix} We_1 & We_2 & We_3 & \cdots & We_n \end{bmatrix} \\ &= \frac{1}{n} W^H \, \mathbf{diag}(Wa)W. \end{aligned}$$

(c) To evaluate

$$\begin{aligned} T(a)x &= \frac{1}{n} W^H \, \mathbf{diag}(Wa)W \\ &= W^{-1} \left((Wa) \circ (Wx)\right), \end{aligned}$$

we compute the discrete Fourier transforms Wa and Wx of the vectors a and x, multiply them componentwise to get $y = (Wa) \circ (Wx)$ and take the inverse DFT $W^{-1}y$ of y. This requires two DFTs, one inverse DFT, and a componentwise product of length n. The complexity is order $n \log n$. In MATLAB or Julia, this is implemented as: `ifft( fft(a) .* fft(x) )`.

(d) The factorization of $T(a)$ show that $T(a)$ is nonsingular if and only if the elements of Wa are nonzero. The inverse of $T(a)$ is

$$T(a)^{-1} = W^{-1}\,\mathbf{diag}(Wa)^{-1}W.$$

We can therefore evaluate $T(a)^{-1}b = W^{-1}\,\mathbf{diag}(Wa)^{-1}Wb$ by computing the DFTs Wa and Wb of a and b, then dividing Wb componentwise by Wa, and computing the inverse DFT of the result. In MATLAB notation:

```
x = ifft( fft(b) ./ fft(a))).
```

We test this algorithm in MATLAB for $n = 10000$.

```
>> n = 10000;  a = randn(n,1);  b = randn(n,1);
>> A = toeplitz(a, [a(1), flipud(a(2:n))']);
>> % solve using standard method
>> tic, x1 = A\b; toc
Elapsed time is 9.560306 seconds.
>> % solve using fast method
>> tic, x2 = ifft( fft(b) ./ fft(a)); toc
Elapsed time is 0.000545 seconds.
>> % compare the results
>> norm(x1-x2)
ans =
   1.7278e-12
```

The results in Julia are similar.

```
julia> using FFTW
julia> n = 10000;  a = randn(n,1);  b = randn(n,1);
julia> A = hcat( [ circshift(a,k) for k = 0:n-1 ]... );
julia> @time x1 = A\b;
  9.421464 seconds (11 allocations: 763.092 MiB, 0.91% gc time)
julia> @time x2 = ifft( fft(b) ./ fft(a));
  0.000934 seconds (206 allocations: 948.656 KiB)
julia> norm(x1-x2)
2.0815737580478035e-12
```

4. *Exercise A6.13.*

(a) We verify that $A^TA = I$.

$$\begin{aligned} A^TA &= \begin{bmatrix} aa^T & I - aa^T \\ I - aa^T & aa^T \end{bmatrix} \begin{bmatrix} aa^T & I - aa^T \\ I - aa^T & aa^T \end{bmatrix} \\ &= \begin{bmatrix} aa^Taa^T + I - 2aa^T + aa^Taa^T & 2(aa^T - aa^Taa^T) \\ 2(aa^T - aa^Taa^T) & I - 2aa^T + aa^Taa^T + aa^Taa^T \end{bmatrix} \end{aligned}$$

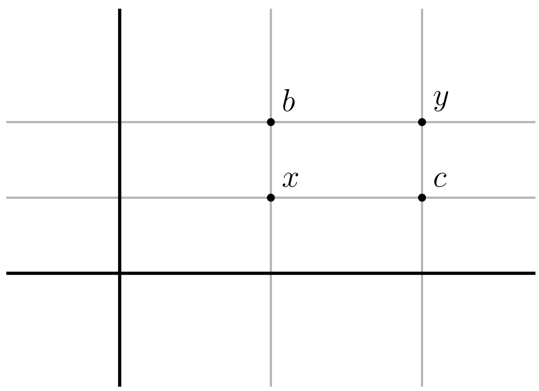
$$
\begin{aligned}
&= \begin{bmatrix} aa^T + I - 2aa^T + aa^T & 2(aa^T - aa^T) \\ 2(aa^T - aa^T) & I - 2aa^T + aa^T + aa^T \end{bmatrix} \\
&= \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}.
\end{aligned}
$$

The simplifications on line 3 follow because $a^Ta = 1$.

(b) Since $A^{-1} = A$, the solution is

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} aa^T & I - aa^T \\ I - aa^T & aa^T \end{bmatrix} \begin{bmatrix} b \\ c \end{bmatrix} = \begin{bmatrix} aa^Tb + (I - aa^T)c \\ (I - aa^T)b + (aa^T)c \end{bmatrix}$$

The vectors aa^Tb and aa^Tc are the projections of b and c on the line through a on the origin. The vectors $(I - aa^T)b$ and $(I - aa^T)c$ are the projections on the line orthogonal to a.



5. *Exercise A6.16.* This follows from $Q = AR^{-1}$ and the fact that R^{-1} is upper triangular. Therefore the kth column of Q is a linear combination of the first k columns of A.

6. *Exercise A7.5.* We evaluate the expression as $x = b + y + v + w$ where

$$y = A^{-1}b, \qquad v = A^{-2}b = A^{-1}y, \qquad w = A^{-3}c = A^{-1}v.$$

(a) LU factorization $A = PLU$ ($(2/3)n^3$ flops).

(b) Calculate $y = A^{-1}b$ by solving $PLUy = b$.

   i. Calculate $\tilde{v} = P^Tb$ (0 flops, because P^Tb is a permutation of b).

   ii. Solve $L\tilde{w} = \tilde{v}$ by forward substitution (n^2 flops).

   iii. Solve $Uy = \tilde{w}$ by backward substitution (n^2 flops).

(c) Calculate $v = A^{-2}b = A^{-1}y$ by solving $PLUv = y$ ($2n^2$).

(d) Calculate $w = A^{-3}b = A^{-1}v$ by solving $PLUw = v$ ($2n^2$).

(e) $x = b + y + v + w$ ($3n$).

Total: $(2/3)n^3 + 6n^2 + 3n$.

7. *Exercise A7.30.*

(a) Define $u = A^{-1}c$. The coefficients of u express c as a linear combination of the columns of A:

$$c = Au = \sum_{j=1}^{n} u_j a_j.$$

If $u_i = (A^{-1}c)_i = 0$, then the vector a_i does not appear in this sum, and we can write it as

$$0 = \sum_{j=1}^{i-1} u_j a_j - c + \sum_{j=i+1}^{n} u_j a_j = \tilde{A}x$$

with $x_i = -1$ and $x_j = u_j$ for $j \neq i$. This proves that the columns of $\tilde{A}$ are linearly dependent.

Alternatively, multiplying $\tilde{A}$ on the left with A^{-1} gives the matrix

$$A^{-1}\tilde{A} = I + (A^{-1}c - e_i)e_i^T = (I - e_i e_i^T) + A^{-1}ce_i^T.$$

This is the identity matrix with its ith column replaced by $A^{-1}c$. Since $(A^{-1}c)_i = 0$, the matrix $A^{-1}\tilde{A}$ has a zero ith row, and therefore is singular. If $A^{-1}\tilde{A}$ is singular, then $\tilde{A}$ is singular.

(b) We multiply $\tilde{A}$ with the proposed inverse, and verify that the product is the identity matrix.

$$\begin{aligned}
&(A + (c - a_i)e_i^T)(A^{-1} - \frac{1}{e_i^T A^{-1}c}(A^{-1}c - e_i)e_i^T A^{-1}) \\
&= I + (c - a_i)e_i^T A^{-1} - \frac{1}{e_i^T A^{-1}c}(c - a_i)e_i^T A^{-1} - \frac{(e_i^T A^{-1}c - 1)}{e_i^T A^{-1}c}(c - a_i)e_i^T A^{-1} \\
&= I + (1 - \frac{1}{e_i^T A^{-1}c} - (1 - \frac{1}{e_i^T A^{-1}c}))(c - a_i)e_i^T A^{-1} \\
&= I.
\end{aligned}$$

(c) From the expression in part (b),

$$\tilde{A}^{-1}b = A^{-1}b - \frac{(A^{-1}b)_i}{(A^{-1}c)_i}(A^{-1}c - e_i).$$

This shows we need $x = A^{-1}b$ and $z = A^{-1}c$, and a few vector operations.

- LU factorization $A = PLU$. $(2/3)n^3$ flops.
- Solve $PLUx = b$ by forward and backward substitution. $2n^2$ flops.
- Solve $PLUz = c$ by forward and backward substitution. $2n^2$ flops.
- Compute $\alpha = x_i/z_i$. 1 flop.
- Compute $y = x - \alpha(z - e_i)$. $2n + 1$ flops.

The total is $(2/3)n^3$ plus lower order terms.