Inverse problems course, spring 2014 Exercise 2 solutions (January 28-31, 2014)
University of Helsinki
Department of Mathematics and Statistics
Samuli Siltanen, Esa Niemi and Teemu Saksala

[T1.] **Answer**

Let U be an orthogonal $(n \times n)$-matrix. Let $(e_i)_{i=i}^n$ be the standard orthonormal base for $\mathbb{R}^n$. Write matrix U with column vectors $(U_1, \ldots, U_n)$. Since we assume that $U^T = U^{-1}$ it holds that

$$I = UU^T = [U_i \cdot U_j] \Rightarrow U_i \cdot U_j = \delta_{ij}. \tag{1}$$

Here $[U_i \cdot U_j]$ is such a matrix that its elements are $U_i \cdot U_j$ and δ_{ij} is the Kronecker delta. Therefore we know that the vectors $(U_1, \ldots, U_n)$ are orthonormal. Let $y = \sum_{i=1}^n y_i e_i \in \mathbb{R}^n$. Remember the bilinearity of inner product and calculate the norm

$$||Uy||^2 = Uy \cdot Uy = \sum_{i.j=1}^n y_i y_j (Ue_i \cdot Ue_j) = \sum_{i.j=1}^n y_i y_j (U_i \cdot U_j) = \sum_{i=1}^n y_i^2 = ||y||^2. \tag{2}$$

Taking the square roots from the first and the last part of equation (2) we have proven the claim.

[T2.] **Answer**

We first recall that a real square matrix S is self-adjoint iff it is symmetric i.e. $S = S^T$. Let $A \in M(\mathbb{R}, k, n)$ i.e. A is a real $(k \times n)-$matrix. Calculate

$$(A^T A)^T = A^T (A^T)^T = A^T A \tag{3}$$

and notice that equation (3) shows that square matrix $(A^T A) \in M(\mathbb{R}, k, k)$ is selfadjoint.

Let $S : \mathbb{R}^k \to \mathbb{R}^k$ be that selfadjoint linear mapping which has matrix representation $A^T A$ wiht respect to standard euclidean basis $(e_i)_{i=1}^k$. Due the Spectral theorem of self-adjoint linear mappings it now holds that there exists an orthonormal basis $(\widetilde{e}_i)_{i=1}^k$ of $\mathbb{R}^k$ s.t. each $\widetilde{e}_i$ is an eigen vector of linear mapping S and in this basis L has matrix representation of diagonal matrix $D = \text{diag}(\lambda_1, \dots, \lambda_n)$ and λ_i is an eigen value related to vector $\widetilde{e}_i$. Let $V = (\widetilde{e}_1, \dots, \widetilde{e}_n)$ which is an orthogonal matrix. Now it also holds that

$$A^T A = VDV^T. \tag{4}$$

Let $L : \mathbb{R}^k \to \mathbb{R}^n$ be that linear mapping which has matrix representation A with respect to standard orthonormal basis $(e_i)_{i=1}^k$ of $\mathbb{R}^k$ and $(f_i)_{i=1}^n$ of $\mathbb{R}^n$. We say that linear mapping $L^* : \mathbb{R}^n \to \mathbb{R}^k$ is an adjoint of L if the following holds for all $u \in \mathbb{R}^k$ and $v \in \mathbb{R}^n$

$$L(u) \cdot v = u \cdot L^*(v). \tag{5}$$

Using matrix convention for linear mapping L in equation (5) it is easy to see that $L^* = A^T$.

Remember formula (5) and calculate

$$L(\widetilde{e}_i) \cdot L(\widetilde{e}_j) = \widetilde{e}_i \cdot L^*(L(\widetilde{e}_j)) = \widetilde{e}_i \cdot S(\widetilde{e}_j) = \lambda_j \widetilde{e}_i \cdot \widetilde{e}_j = \lambda_j \delta_{ij}. \tag{6}$$

By previous equation it holds that $\lambda_j \geq 0$. Reorder basis $(\widetilde{e}_i)_{i=1}^k$ if necessary and choose $l \leq \min\{k, n\}$ s.t. $\lambda_i \geq \lambda_{i+1} > 0$ for every $i \leq l - 1$. Define

$$\widetilde{f}_i := \frac{L(\widetilde{e}_i)}{||L(\widetilde{e}_i)||} = \frac{L(\widetilde{e}_i)}{\sqrt{\lambda_i}} \tag{7}$$

and note that by formula (6) set $(\widetilde{f}_i)_{i=1}^l$ is orthonormal. Next choose vectors $(\widetilde{f}_i)_{i=l}^n$ s.t. set $(\widetilde{f}_i)_{i=1}^n$ is an orthonormal basis for $\mathbb{R}^n$. By formula (6) it also holds for any $i \in l + 1, \dots, k$ that

$$L(\widetilde{e}_i) = 0,$$

Finally choose that $\sigma_i = \sqrt{\delta_i}$. Note that, if we now choose $[L_{ij}]$ as $[A_{ij}]$ in exercise paper, we have shown that with respect to orthonormal basis $(\widetilde{e}_i)_{i=1}^k$ and $(\widetilde{f}_i)_{i=1}^n$ $[L_{ij}]$ is the matrix of mapping L. If $U = (\widetilde{f}_1, \dots, \widetilde{f}_n)$ we have also shown that

$$A = U[L_{ij}]V^T.$$

[T3.] **Answer**

Let

$$\mathbf{f} = \begin{pmatrix} f_7 & f_8 & f_9 \\ f_4 & f_5 & f_6 \\ f_1 & f_2 & f_3 \end{pmatrix}$$

be the attenuation values of given square and $\mathbf{m} = (m_1, \dots m_6) \in \mathbb{R}^6$ our maesure data of 6 X-ray lines. Here we think that the bottommost line of the frist picture is L_1 and L_6 is the topmost line of the second picture. Next we have to find the matrix $\mathbf{A} \in M(\mathbb{R}, 9, 6)$ s.t.

$$\mathbf{m} = \mathbf{A}\mathbf{f}. \tag{8}$$

We use the following facts to construct $\mathbf{A}$.

- The length of the side of each pixel is 1.
- Entry A_{ij} of matrix $\mathbf{A}$ is the distance that ray L_i travels in the jth pixel.

Looking the fist picture we note that if ray L_i travels through j^{th} pixel the corresponding number

$$A_{ij} = \sqrt{1^2 + \left(\frac{1}{3}\right)^2} = \sqrt{\frac{10}{9}} = \frac{\sqrt{10}}{3}.$$

Looking the second picture we note that if ray L_i travels through j^{th} pixel the corresponding number

$$A_{ij} = \sqrt{1^2 + 1^2} = \sqrt{2}.$$

Let us build the rows 1, 4 and 5 in detail. Ray L_1 travels through pixels 1, 2 and 3. Therefore we have that row

$$A_1 = (A_{1,j})_{J=1}^9 = (\frac{\sqrt{10}}{3}, \frac{\sqrt{10}}{3}, \frac{\sqrt{10}}{3}, 0, \dots, 0).$$

Ray L_4 travels through pixels 2 and 6. Therefore we have that row

$$A_4 = (A_{1,j})_{J=1}^9 = (0, \sqrt{2}, 0, 0, 0, \sqrt{2}, 0, 0, 0).$$

Ray L_5 travels through pixels 1, 5 and 9. Therefore we have that row

$$A_5 = (A_{1,j})_{J=1}^9 = (\sqrt{2}, 0, 0, 0, \sqrt{2}, 0, 0, 0, \sqrt{2}).$$

Now it holds that matrix

$$\mathbf{A} = \begin{pmatrix} \frac{\sqrt{10}}{3}\ \frac{\sqrt{10}}{3}\ \frac{\sqrt{10}}{3}\ 0\ 0\ 0\ 0\ 0\ 0 \\ 0\ 0\ 0\ \frac{\sqrt{10}}{3}\ \frac{\sqrt{10}}{3}\ \frac{\sqrt{10}}{3}\ 0\ 0\ 0 \\ 0\ 0\ 0\ 0\ 0\ 0\ \frac{\sqrt{10}}{3}\ \frac{\sqrt{10}}{3}\ \frac{\sqrt{10}}{3} \\ 0\ \sqrt{2}\ 0\ 0\ 0\ \sqrt{2}\ 0\ 0\ 0 \\ \sqrt{2}\ 0\ 0\ 0\ \sqrt{2}\ 0\ 0\ 0\ \sqrt{2} \\ 0\ 0\ 0\ \sqrt{2}\ 0\ 0\ 0\ \sqrt{2}\ 0 \end{pmatrix}$$

and

$$(m_1, m_2, m_3, m_4, m_5, m_6) = \begin{pmatrix} \frac{\sqrt{10}}{3} \; \frac{\sqrt{10}}{3} \; \frac{\sqrt{10}}{3} \; 0\,0\,0\,0\,0\,0 \\ 0\,0\,0 \; \frac{\sqrt{10}}{3} \; \frac{\sqrt{10}}{3} \; \frac{\sqrt{10}}{3} \; 0\,0\,0 \\ 0\,0\,0\,0\,0\,0 \; \frac{\sqrt{10}}{3} \; \frac{\sqrt{10}}{3} \; \frac{\sqrt{10}}{3} \\ 0 \; \sqrt{2} \; 0\,0\,0 \; \sqrt{2} \; 0\,0\,0 \\ \sqrt{2} \; 0\,0\,0 \; \sqrt{2} \; 0\,0\,0 \; \sqrt{2} \\ 0\,0\,0 \; \sqrt{2} \; 0\,0\,0 \; \sqrt{2} \; 0 \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ f_6 \\ f_7 \\ f_8 \\ f_9 \end{pmatrix}.$$

[M1.] **Answer**

After running the files `DC1_cont_data_comp.m`, `DC2_discrete_data_comp.m` and `DC4_truncSVD_comp.m` one has the matrices `U, D` and `V`, i.e. the SVD of `A`, stored in the Matlab Workspace. The condition number of `A` can then be computed e.g. by the command `D(1,1)/D(n,n)` either in the m-file `DC4_truncSVD_comp.m` or in the Command Window. The condition numbers for resolutions `n=100,200,300,400` are `1.0944e+03, 5.2812e+04, 1.0201e+05` and `1.4951e+06`, respectively. In other words, the condition number of `A` becomes larger as `n` grows. This means that the more precisely one models (discretizes) the convolution, the more ill-posed deconvolution problem one gets!

Note: The condition number of a matrix can also be computed by Matlab's built-in function `cond.m`, i.e. the same condition numbers as above can be obtained using the command `cond(A)`. (However, since we already have computed the SVD of `A`, it is computationally more efficient to compute the condition number as `D(1,1)/D(n,n)`.)

[M2.] **Answer**

(b) (Note that here the diagonal matrix `D` is not a square matrix as in the previous exercise.)

One can compute the condition number of `A` by, e.g., the command `D(1,1)/D(min(size(D)),min(size(D)))` to get `1.0152e+05`.

(c) The minimum relative error is 64%. (If you modify the code to compute the relative error for every number of singular vectors, you will get a minimum relative error of 63.4954% with 500 singular vectors.)

[M3.] **Answer**

(a) Simply replace the number 180 by number 90 on line 23 in `XRM1_matrix_comp.m` and on line 27 in `XRM3_NoCrimeData_comp.m`.

(b) Compute the condition number in the same manner as in M2(b) to get cond(A)
=9.9649e+05. Compared to M2(b), the condition number here is larger, meaning that the limited angle (90°) CT problem is more ill-posed than the "corresponding" full angle (180°) problem.

(c) The minimum relative error is 74%, obtained by 220, 293 or 366 singular vectors. (If you modify the code to compute the relative error for every number of singular vectors, you will get a minimum relative error of 73.5357% with 230 singular vectors). Compared to M2(b), the minimum relative error here is larger. Also, the minimum error is attained at a lower number of singular vectors.

The reconstruction is arguably worse in this limited-angle case. It contains certain details of the phantom but certain details might be totally missing; more precisely, shapes in the direction of the x-ray projections are reconstructed relatively well while shapes perpendicular to the direction of the x-rays are missing or poorly reconstructed.

In the limited-angle case the singular vectors are not as symmetric as they are in the full-angle case, rather they seem to be "stretched" in the direction of the x-rays, similarly to the shapes in the limited-angle reconstructions.