Inverse problems course — Exercise 3 (February 3–5, 2015)
University of Helsinki
Department of Mathematics and Statistics
Samuli Siltanen and Andreas Hauptmann
Related book sections (Mueller & Siltanen 2012):

Theoretical exercises:

T1. Set

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

(a) Draw the range of A as a subset of $\mathbb{R}^2$. Draw also the point $(0, 1)$ to the same image.

(b) Find the least-squares solution(s) of equation $Af = m$, where $m = [0, 1]^T$.

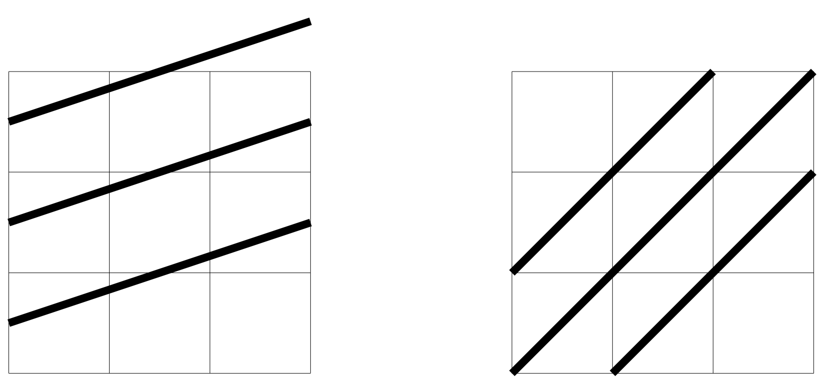
(c) Determine the minimum-norm solution of equation $Af = m$ by geometric arguments. (Analyze the triangles involved.)

T2. (a) Diagonalize (by hand, not computer) the symmetric matrix $A^T A$, where A is as above. Make sure that the eigenvectors are orthonormal.

(b) Follow the method of Problem T3 of Exercise 2 and calculate the singular value decomposition of A by hand.

(c) Find the minimum-norm solution of equation $Af = m$ by Moore-Penrose pseudoinverse. Do you get the same answer than in Problem T1?

T3. Thin lines depict pixels and thick lines X-rays in this image:



Give a numbering to the nine pixels ($f \in \mathbb{R}^9$) and to the six X-rays ($m \in \mathbb{R}^6$), and construct the matrix A for the measurement model $m = Af$. The length of the side of a pixel is one.

Matlab exercises:

M1. Consider equations $x_1 + x_2 = 1$, $x_2 = -2$ and $-\frac{1}{3}x_1 + x_2 = -2$.

(a) Write the equations in the matrix form $Ax = y$. (That is, specify the elements in the 3×2 matrix A and the vector $y \in \mathbb{R}^3$.)

(b) Use Matlab to compute the singular value decomposition $A = UDV^T$.

(c) Using the result of (b), construct D^+ and the minimum norm solution $x^+ := VD^+U^Ty$ in Matlab. Draw the three lines specified by the equations and the point x^+ in the (x_1, x_2)-plane. Discuss the result.

M2. The *condition number* of a square matrix A of size $n \times n$ is defined by

$$\mathrm{cond}(A) := \frac{d_1}{d_n},$$

where d_1 and d_n are the first and last singular values of A, respectively.

Download the Matlab routine `XR01_buildA.m` from the course website. There, you can choose two crucial numbers: `N` determines the size of the reconstruction (which is $N \times N$), and `T` is the number of projection directions evenly distributed between 0 and 180 degrees.

(a) Let `T` be constant and increase `N` step by step. Use the command `spy(A)` to see the structure of the measurement matrix. What happens to $\mathrm{cond}(A)$ when `N` grows? Why?

(b) Let `N` be constant and increase `T` step by step. Use the command `spy(A)` to see the structure of the measurement matrix. What happens to $\mathrm{cond}(A)$ when `T` grows? Why?