Homework 2

Due: Wednesday 10/14/2015.

Reading assignment: Chapter 6, chapter 7, and sections 8.1, 8.2, 8.3 of the textbook.

Homework problems

1. Exercise 1.20.
2. Exercise 1.21.
3. Exercise 2.2.
4. Exercise 2.9.
6. Tomography. Download the file tomography.mat from the class webpage and load it in MATLAB using the command load tomography. This creates a matrix \( A \) of size \( 576 \times 784 \) and a vector \( b \) of length 576. The matrix \( A \) and the vector \( b \) describe a toy tomography example and were constructed using the MATLAB AIR Tools package that can be found at www2.compute.dtu.dk/~pcha/AIRtools. (You do not need this package for the exercise.) The geometry is shown in the figure below.

![Tomography Example](image-url)
The image at the center is a black-and-white image of size $28 \times 28$. The figure shows a random image but in the actual problem we used one of the images of handwritten digits of homework 1. The green dots are twelve source locations. For each source location we generate 48 rays emanating from the source. (The figure shows the rays for two sources only.) For each ray, we calculate the line integral of the pixel intensities along the ray. This gives $12 \cdot 48 = 576$ linear equations

$$
\sum_{j=1}^{784} A_{ij} x_j = b_i, \quad i = 1, \ldots, 576.
$$

Here $x_j$ denotes the intensity in pixel $j$ of the image (images are stored as vectors of length 784, as in homework 1), $A_{ij}$ is the length of the intersection of ray $i$ with pixel $j$, and $b_i$ is the value of the line integral along ray $i$. These equations can be written in matrix form as $Ax = b$ where $A$ and $b$ are the data in `tomography.mat`, or as

$$
a_i^T x = b_i, \quad i = 1, \ldots, 576,
$$

where $a_i^T$ is row $i$ of $A$.

The purpose is to reconstruct the image $x$ from the line integral measurements. We will use Kaczmarz’s iterative algorithm (lecture 2, page 38) for this purpose. Note however that this is a small problem and very easy to solve using the standard non-iterative methods that are discussed later in the course.

Kaczmarz’s algorithm starts at an arbitrary point $x$ (for example, a zero vector) and then cycles through the equations. At each iteration we take a new equation, and update $x$ by replacing it with its projection on the hyperplane defined by the equation. If $K$ is the number of cycles and $m = 576$ is the number of equations, the algorithm can be summarized as follows.

**Initialize $x$.**

For $k = 1, \ldots, K$:

For $i = 1, \ldots, m$:

Project $x$ on the $i$th hyperplane:

$$
x := x - \frac{a_i^T x - b_i}{\|a_i\|^2} a_i.
$$

end

end

Run the algorithm for $K = 10$ cycles, starting at $x = 0$ (a black image). Compute the error $\|Ax - b\|/\|b\|$ after each cycle. Also display the reconstructed image $x$ (using the command `imshow(reshape(x, 28, 28))`).