HW #1 - due 10/14/2019

The goal of this assignment is to illustrate issues related with inversion of a diffusion process in the presence of noisy data and the need for regularization. You are encouraged to use any software available in your favorite programming language. In particular, Matlab provides a friendly syntax for implementation.

Mathematical Background:

Consider a linear system of equations

$$\mathbf{Ax} = \mathbf{d} \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^n$ is an unknown vector (transmitted data), $\mathbf{d} \in \mathbb{R}^n$ is a given vector (observed/received data) and $\mathbf{A} \in \mathbb{R}^{n \times n}$ is a given nonsingular matrix (transformation of data).
In many applications the observed/received data is corrupted by unknown noise or measurement errors, such that we only have access to

$$\hat{\mathbf{d}} = \mathbf{d} + \boldsymbol{\xi}$$

where $\xi \in \mathbb{R}^n$ is an unknown noise vector. The problem is then formulated as follows:

Given the matrix $\mathbf{A}$ and a vector $\hat{\mathbf{d}}$, provide an approximation of the unknown vector $\mathbf{x}$.

Practical difficulties arise when $\mathbf{A}$ is ill-conditioned. In such case, an attempt to provide an approximation $\hat{\mathbf{x}} \approx \mathbf{x}$ by solving

$$\mathbf{A}\hat{\mathbf{x}} = \hat{\mathbf{d}} \tag{2}$$

will not work since $\hat{\mathbf{x}}$ will be drastically corrupted by noise,

$$\hat{\mathbf{x}} = \mathbf{A}^{-1}\hat{\mathbf{d}} = \mathbf{A}^{-1}(\mathbf{d} + \boldsymbol{\xi}) = \mathbf{x} + \underbrace{\mathbf{A}^{-1}\boldsymbol{\xi}}_{amplified\ noise} \tag{3}$$

Homework content

Consider an image represented as a matrix $\mathbf{X} \in \mathbb{R}^{n \times m}$ and a blurring process represented by the matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$. Whereas the true image is the solution to the matrix equation

$$\mathbf{AX} = \mathbf{D}$$

we want to reconstruct an approximation $\widehat{\mathbf{X}}$ to the image $\mathbf{X}$ given the blurring matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ and a matrix of received noisy data

$$\widehat{\mathbf{D}} = \mathbf{AX} + \boldsymbol{\xi} \tag{4}$$

The direct inversion approach provides

$$\widehat{\mathbf{X}} = \mathbf{A}^{-1}\widehat{\mathbf{D}} \tag{5}$$

The data file clown1.m contains a 200×320 matrix $\mathbf{X}$ ("the truth") that represents the picture clown1.jpg. Alternatively, in Matlab use "load clown" since "clown.mat" should be available in the Matlab built in data files. The image is loaded by default in a matrix X. Then use "A=X; imagesc(A)", "colormap(gray)".

Blurring (data transmission process) is represented by a diffusion process as follows: Consider a 200×200 symmetric tridiagonal matrix $\mathbf{B}$ with entries [1]

$$B(i,i) = 1 - 2L, \ i = 1, 2, \ldots n$$

$$B(i, i+1) = L, \ i = 1, 2, \ldots n - 1; \quad B(i+1, i) = L, \ i = 1, 2, \ldots n - 1$$

where $L = 0.1$. Then the blurring operator is $\mathbf{B}$ to the power k

$$\mathbf{A} = \mathbf{B}^k$$

such that higher values of k increase the amount of diffusion. We will use a range of values $k = 1, 2, \ldots 20$.

The noise $\boldsymbol{\xi}$ in received data is represented as a matrix with entries random numbers taken from a normal distribution with mean 0 and standard deviation 0.01,

$$\boldsymbol{\xi} = 0.01 * randn(200, 320)$$

- **Task 1 (10 points)** Provide the plot of the condition number of the matrix $\mathbf{A} = \mathbf{B}^k$ for $k = 1 : 20$.

- **Task 2 (20 points)** Generate the noisy receive data image $\widehat{\mathbf{D}}$ as in (4) and the reconstructed image $\widehat{\mathbf{X}}$ given by the direct inversion approach (5). Provide the plots of the absolute error and the relative error in the reconstructed image $\widehat{\mathbf{X}}$ for $k = 1 : 20$,

$$\|\mathbf{X} - \widehat{\mathbf{X}}\|, \qquad \frac{\|\mathbf{X} - \widehat{\mathbf{X}}\|}{\|\mathbf{X}\|}$$

using the Frobenius (Euclidean) matrix norm.

- **Task 3 (10 points)** Provide the reconstructed image for $k = 1$, $k = 5$ and $k = 20$

- **Task 4 (10 points)** Repeat tasks 2 and 3 for a noise matrix with standard deviation 0.1,

$$\boldsymbol{\xi} = 0.1 * randn(200, 320)$$

---

[1]such matrix results from numerical discretization of the 1-D heat equation $u_t - u_{xx} = 0$