IP Project 1

Gravity Surveying Inverse Problem Analysis and Solution

Due Thursday February 16 (at class)

- If you have questions, **ask them in class rather in private**.
- Resources: Use any resources you choose, except other classmates and faculty. You may find information in books, papers, and on the web.
- Explain your answers for each problem, show Matlab code used in a problem **(that you wrote)**, and use Matlab graphics when appropriate. Please do not turn in a lot of computational results. **Be brief and summarize your computations with plots and tables**.
- Turn in a **compact**, documented paper. If you decide to email me your paper, **please** send one file, preferably of type pdf. Also, **please** use the following naming system yourlastname_IPproj1_SP12.pdf

  If you cannot easily put together the pdf file, hard copy will be fine.

Problem Setup

- Refer to the emails and handouts on the gravity surveying inverse problem and the code **gravity.m**. Use the code to generate A, b, and the true solution x using:

$$[A, b, x] = gravity(n, example, 0, 1, d).$$

  The matrix A will be $n \times n$ and Toeplitz. Include in your tests below $n = 32$, example = 2, and depth $d = .5$. Experiment with various input parameters, not just those above.

Solving the Inverse Problem

For each computed solution xc you obtain in problems 4 through 7 below, report the relative error:

$$\frac{\|xc - x\|_2}{\|x\|_2}.$$

1. Study experimentally how the **right-hand side** b **varies with the depth** d, for a fixed value of n. You might look at some plots of the vectors b.
2. Compute the **condition number, and examine how it varies** with n, for a fixed value of the depth d.
3. Then keep n fixed at, say 32, and study how the **condition number varies with depth** d; try to explain the observed behavior.
4. Compute xc first using the **backslash command**.

5. Now, solve for xc using the **truncated SVD** method. In order to "try" to find a good singular value truncation point in the SVD. You might use the command svds. Try various truncation points, including the use of the first 16 singular values only.

6. Compute xc using **Tikhonov regularization**. Try each of the following regularization parameters: $\lambda = .0001, .001, .01,$ and $.1$. Of course you cannot use Greek letters easily in Matlab, so you might use p for λ.

   First, se the normal equations method with the backslash command for,

   $$(A^T A + \lambda^2 I)xc = A^T b.$$

   Also, solve the least squares problem

   $$min\|\begin{bmatrix} A \\ \lambda I \end{bmatrix} xc - \begin{bmatrix} b \\ 0 \end{bmatrix}\|_2$$

   using, say, the Matlab command lsqlin. (Use doc lsqlin to see how to use the command.)

   Which regularization parameter λ gives the smallest relative error in each solution method?

7. Finally, **add noise** to the right hand side b using the following Matlab commands: $s = randn(32, 1)$ and $noise = eps * 10^k$, with k = 5, 10, 12, 14 and 16. Then use

   $$bn = b + s * noise$$

   for the new right hand side, that is, replace b by bn.

   Now, repeat problems 4, 5 and 6 above using the 5 noisy bn vectors.

   Again, which regularization paramter λ gives the smallest relative error in each solution method?