ECE 497-3: Inverse Problems in Engineering
Homework #5

Due: Friday January 17, 2003

<u>*Motivation*</u>
In this homework, we examine

- plotting an L-curve
- using CRESO to choose b
- using the maximum curvature method to choose b

1) In this problem, we will determine how to use CRESO for the case of higher order Tikhonov regularization. As for zero order regularization, CRESO maximizes the difference between the derivative of the smoothing term $b||\mathbf{R}\underline{g}||^2$ and the derivative of the fit to the data $||\underline{x} - \underline{d}||^2$.
a) By performing the required minimization, show that the $\underline{g}$ that is the solution to the regularization problem

$$\begin{aligned} \min \Pi &= ||\underline{x} - \underline{d}||^2 + b||\mathbf{R}\underline{g}||^2 \\ \text{subject to } \underline{x} &= \mathbf{Q}\underline{g} \end{aligned}$$

is given by

$$\underline{g} = \left(\mathbf{Q}^T\mathbf{Q} + b\mathbf{R}^T\mathbf{R}\right)^{-1}\mathbf{Q}^T\underline{d}$$

b) Define

$$B(b) = b||\mathbf{R}\underline{g}||^2 - ||\underline{x} - \underline{d}||^2$$

Show that

$$\frac{dB(b)}{db} = ||\mathbf{R}\underline{g}||^2 + 2b\frac{d}{db}||\mathbf{R}\underline{g}||^2 = C(b)$$

c) Using the generalized singular value decomposition (gsvd), we can write

$$\begin{aligned} \mathbf{Q} &= \mathbf{U}\mathbf{C}\mathbf{X}^T \\ \mathbf{R} &= \mathbf{V}\mathbf{S}\mathbf{X}^T \end{aligned}$$

where $\mathbf{U}$ and $\mathbf{V}$ are *unitary* matrices, $\mathbf{C}$ and $\mathbf{S}$ are *diagonal* matrices with elements s_i and c_i on the diagonals, and

$$\mathbf{C}^T\mathbf{C} + \mathbf{S}^T\mathbf{S} = \mathbf{I}$$

Assuming $\mathbf{X}^{-1}$ exists, and defining

$$\begin{aligned} \underline{g} &= \mathbf{X}^{-T}\underline{\tilde{g}} \\ \underline{\tilde{d}} &= \mathbf{U}^T\underline{d} \end{aligned}$$

show that we can write $C(b)$ as

$$C(b) = \sum_{i=1}^{i=N} \frac{s_i^2 c_i^2 \tilde{d}_i^2}{(c_i^2 + b s_i^2)^2} \left[1 - \frac{4 b s_i^2}{c_i^2 + b s_i^2} \right]$$

The idea of CRESO is then to find a b to maximize $C(b)$.

2) A different method for choosing the regularization parameter b is to find the point on the L-curve that has the maximum curvature. In this problem we will derive a set of formulas to do this for zero order Tikhonov regularization.

From problem one, the $\underline{g}$ that is the solution to the regularization problem

$$\begin{aligned} \min \Pi &= ||\underline{x} - \underline{d}||^2 + b||\underline{g}||^2 \\ \text{subject to } \underline{x} &= \mathbf{Q}\underline{g} \end{aligned}$$

is given by

$$\underline{g} = \left(\mathbf{Q}^T \mathbf{Q} + b \mathbf{I} \right)^{-1} \mathbf{Q}^T \underline{d}$$

Now assume we compute the singular value decomposition of $\mathbf{Q}$

$$\mathbf{Q} = \mathbf{U}\mathbf{S}\mathbf{V}^T$$

where $\mathbf{U}$ and $\mathbf{V}$ are *unitary* and $\mathbf{S}$ is a diagonal matrix with singular values σ_i on the diagonals. Assume then

$$\begin{aligned} \underline{d} &= \mathbf{U}\underline{\alpha} \\ \underline{g} &= \mathbf{V}\underline{\beta} \end{aligned}$$

a) Show that

$$\begin{aligned} \underline{\alpha} &= \mathbf{U}^T \underline{d} \\ \beta_i &= \frac{\sigma_i \alpha_i}{\sigma_i^2 + b} \end{aligned}$$

b) Now define

$$\begin{aligned} \eta &= ||\underline{g}||^2 \\ \rho &= ||\mathbf{Q}\underline{g} - \underline{d}||^2 \end{aligned}$$

Show that we can write

$$\eta = \sum_{i=1}^{i=N} \frac{\sigma_i^2 \alpha_i^2}{(\sigma_i^2 + b)^2}$$

$$\rho = \sum_{i=1}^{i=N} \frac{b^2 \alpha_i^2}{(\sigma_i^2 + b)^2}$$

$$\dot{\eta} = \frac{d\eta}{db} = -2 \sum_{i=1}^{i=N} \frac{\sigma_i^2 \alpha_i^2}{(\sigma_i^2 + b)^3}$$

$$\dot{\rho} = \frac{d\rho}{db} = 2 \sum_{i=1}^{i=N} \frac{b \sigma_i^2 \alpha_i^2}{(\sigma_i^2 + b)^3}$$

where N is equal to the minimum number of either the σ_i or α_i.

c) Show that, from the above relations, we can conclude

$$\begin{aligned} \dot{\rho} &= -b\dot{\eta} \\ \ddot{\rho} &= -\dot{\eta} - b\ddot{\eta} \end{aligned}$$

d) Now let's define

$$\begin{aligned} \hat{\eta} &= \log \eta \\ \hat{\rho} &= \log \rho \end{aligned}$$

Show that

$$\begin{aligned} \dot{\hat{\rho}} &= \frac{\dot{\rho}}{\rho} \\ \dot{\hat{\eta}} &= \frac{\dot{\eta}}{\eta} \\ \ddot{\hat{\rho}} &= \frac{\ddot{\rho}\rho - (\dot{\rho})^2}{\rho^2} \\ \ddot{\hat{\eta}} &= \frac{\ddot{\eta}\eta - (\dot{\eta})^2}{\eta^2} \end{aligned}$$

e) The curvature of the L-curve (on the log-log scale) is defined as

$$\kappa = 2 \frac{\dot{\hat{\rho}}\ddot{\hat{\eta}} - \ddot{\hat{\rho}}\dot{\hat{\eta}}}{[(\dot{\hat{\rho}})^2 + (\dot{\hat{\eta}})^2]^{\frac{3}{2}}}$$

Show that the numerator and denominator can be written as

$$\begin{aligned} \dot{\hat{\rho}}\ddot{\hat{\eta}} - \ddot{\hat{\rho}}\dot{\hat{\eta}} &= [b^2 \eta \dot{\eta} + b\rho\dot{\eta} + \rho\eta] \frac{(\dot{\eta})^2}{\rho^2 \eta^2} \\ \left[(\dot{\hat{\rho}})^2 + (\dot{\hat{\eta}})^2\right]^{\frac{3}{2}} &= (b^2 \eta^2 + \rho^2)^{\frac{3}{2}} \frac{(\dot{\eta})^3}{\eta^3 \rho^3} \end{aligned}$$

f) Show that the equation for the curvature is

$$\kappa = 2 \frac{\eta\rho \left[b\rho\dot{\eta} + \rho\eta + b^2 \eta \dot{\eta}\right]}{\dot{\eta} \left[b^2 \eta^2 + \rho^2\right]^{\frac{3}{2}}}$$

where

$$\eta = \sum_{i=1}^{i=N} \frac{\sigma_i^2 \alpha_i^2}{(\sigma_i^2 + b)^2}$$

$$\rho = \sum_{i=1}^{i=N} \frac{b^2 \alpha_i^2}{(\sigma_i^2 + b)^2}$$

$$\dot{\eta} = \frac{d\eta}{db} = -2 \sum_{i=1}^{i=N} \frac{\sigma_i^2 \alpha_i^2}{(\sigma_i^2 + b)^3}$$

We then find b to maximize $|\kappa|$.

3) We now want to implement a program for plotting L-curves and choosing the regularization parameter. We will look at the following examples:

```
%
%================
%  Test 1
%================
%

  Q1 = [-0.2042 0.0559 0.3715 -0.2077;
         0.1347 0.2208 0.5294 0.1336;
         0.4803 0.3889 0.6862 0.4670;
         0.6078 0.6088 1.2159 0.6081];
  g1 = [1; 1; 1; 1];
  Y = Q1*g1;
  RMS = norm(Y)/sqrt(length(Y));
  sigma = RMS/(10^(0.05*SNR));
  noise = sigma*randn(length(Y),1);
  d = Y + noise;
%
%================
%  Test 2
%================
%
  Q2 = [-0.0505 0.0683 0.1175 0.0538 0.1709;
         0.2215 0.1729 0.0005 0.2794 0.2963;
         0.4663 0.4675 -0.0623 0.4237 0.3403;
        -0.1871 0.0701 0.2650 0.0460 0.3136;
         0.2109 0.0203 0.5130 -0.0603 0.7171];
  g2 = [1; 1; 1; 1; 1];
  Y = Q2*g2;
  RMS = norm(Y)/sqrt(length(Y));
  sigma = RMS/(10^(0.05*SNR));
  noise = sigma*randn(length(Y),1);
```

```
  d = Y + noise;
%
%==================
%  Test 3
%=================
%
  Q3 = [1.001  1.967  3.055  3.985  4.991  6.001;
         7.000  8.026  8.957 10.010 11.007 12.001;
        13.000 14.010 14.982 16.007 17.004 17.998;
        19.000 20.011 20.982 22.002 29.004 30.000;
        24.998 26.005 26.990 28.008 29.004 30.000];
  g3 = [1; 1; 1; 1; 1; 1];
  Y = Q3*g3;
  RMS = norm(Y)/sqrt(length(Y));
  sigma = RMS/(10^(0.05*SNR));
  noise = sigma*randn(length(Y),1);
  d = Y + noise;
```

a) Your code should have at least five arguments, something like:

- bmin = the minimum expected value of b
- bmax = the maximum expected value of b
- N = the number of values of b between bmin and bmax. ($N \geq 1000$))
- SNR = signal to noise level
- which = the case to run (of the three above)

You may want to put the command

```
randn('state',0);
```

at the beginning of your code, so you will get the same noise each time. This is important since you need a different range of b for different noise levels and noise realizations.

The Matlab command

```
s = diag(S);
```

will give you a vector s of the diagonal of matrix S.

Start off by writing code that will produce and L-curve on a log-log scale, i.e., it should plot

$$\log(||\mathbf{Q}\underline{g} - \underline{d}||^2) \quad vs. \quad \log(||\underline{g}||^2)$$

You should find the b's between bmin and bmax using the logspace command

```
b = logspace(log10(bmin),log10(bmax),N);
```

You should, at this point, play around with a range of b's to be sure you can get the characteristic 'L' shape for the L-curve for various signal to noise levels. It may be difficult at small SNR levels (3 db or less) to get a very good L. At larger SNR's you should be able to do better.

b) Implement the CRESO method for choosing the regularization parameter for **zero** order Tikhonov regularization. Use it to estimate the value of b and then estimate $\underline{g}$ for various signal to noise levels. Plot the chosen value of b on the L-curve. I do not expect you to really write an optimization routine. Just search though the values of b you have and find the best one.

From class notes, for zero order Tikhonov regularization

$$\begin{aligned} \mathbf{Q} &= \mathbf{U}\mathbf{S}\mathbf{V}^T \\ \underline{d} &= \mathbf{U}\underline{\alpha} \\ \underline{g} &= \mathbf{V}\underline{\beta} \\ \beta_i &= \frac{\sigma_i \alpha_i}{\sigma_i^2 + b} \end{aligned}$$

We want to find b to maximize

$$C(b) = \sum_{i=1}^{i=N} \frac{\sigma_i^2 \alpha_i^2}{(\sigma_i^2 + b)^2} \left[1 - \frac{4b}{\sigma_i^2 + b} \right]$$

c) Implement the maximum curvature method for choosing the regularization parameter for **zero** order Tikhonov regularization. Use it to estimate the value of b and then estimate $\underline{g}$ for various signal to noise levels. Plot the chosen value of b on the L-curve. I do not expect you to really write an optimization routine. Just search though the values of b you have and find the best one.

d) Find the optimal value of the regularization parameter (you can define optimal in any reasonable way) for **zero** order Tikhonov regularization. Use it to estimate the value of b and then estimate $\underline{g}$ for various signal to noise levels. Plot the chosen value of b on the L-curve. I do not expect you to really write an optimization routine. Just search though the values of b you have and find the best one.

Note: Parts (a) - (d) should all be shown on the same plot with the same realization of noise. Your code should be written so that for each realization of noise with a certain SNR, you can plot the L curve, the location on the L curve chosen by CRESO, by the maximum curvature method, and the optimal choice. For each realization of noise you will get a different plot. We will uses these matrices (and CRESO) again, so save your code!