

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

Due: Friday December 20, 2002

Motivation

In this homework, you will write MATLAB code to implement both the “brute force” and dynamic programming solution to a single input, single output, scalar state variable model, such as

$$x_{j+1} = mx_j + g_j$$

You will also explore the use of higher order regularization.

1) Implement the dynamic programming method that produces the values in Table 1.2 of the textbook. For this case,  $m = 1$  and  $b = 0.2$ . Use the values of  $d$  given in Table 1.1 (there is an error in the  $d$  values in Table 1.2). You might want to try and duplicate the results of both Tables 1.1 and 1.2 if you are having troubles, since they give some intermediate variable values.

2) Write a program (m-file) to implement the “brute-force” method of solving the dynamic equations. For the data given in Table 1.1, estimate the  $g_j$ 's. You should get the same answers as in problem 1.

3) Now write a program (m-file) that will construct a system with added noise and solve it with both the brute force or the dynamic programming methods. The inputs to the program should be:

- points = the number of sample points
- sigma = the standard deviation of the noise
- b = the value of the regularization parameter
- m = the value of  $m$
- x1 = the initial state

Your program must be able to handle variations in any of these input parameters! In order to generate some data, use something like the following pieces of code so you will know the true input.

```
%
% -----
% first test case, a sine wave over one period
% -----
%
% tpoints = points+1;
% d = zeros(tpoints,1);
% d(1) = x1;
% t = linspace(0,2*pi,points);
% g = sin(t);
% for i=2:tpoints
```

```

%     d(i) = m*d(i-1)+g(i-1);
% end;
%
% -----
% second test case, a pulse wave. each pulse of length points
% -----
%
%     t = ones(points,1);
%     g = [0*t; t; 0*t; t; 0*t; t];
%     p = length(g);
%     tpoints = p+1;
%     d = zeros(tpoints,1);
%     d(1) = x1;
%     for i=2:tpoints
%         d(i) = m*d(i-1)+g(i-1);
%     end;
%
% add some noise to the measurements
%
%     d = d+sigma*randn(tpoints,1);
%

```

a) Show that, for a few sample points, the results with dynamic programming and the brute force method produce the same results for the same inputs.

b) Explore the results for the two different sample problems with different values of the regularization parameter  $b$  for the case with  $m = 1$ ,  $\sigma = 0.5$  with at least 30 sample points. (Note that for the on/off input, your value of *points* will be much smaller than 30.) Compare the regularized versus unregularized solutions (i.e., the least squares solution). I want you to produce plots for this part that contain (1) the true input, (2) the least squares estimate, and (3) the regularized estimate. Try at least three values of  $b$  and try to find a good value.

4) Using the brute force method, incorporate second order Tikhonov regularization by penalizing the second derivative of the  $g_j$ 's (but don't penalize the estimate of the initial state!) Recall we have the difference expressions:

$$\begin{aligned}
 \ddot{f}(n) &\approx f(n+1) - 2f(n) + f(n-1) \\
 \ddot{f}(1) &\approx 2f(1) - 5f(2) + 4f(3) - f(4) \\
 \ddot{f}(N) &\approx 2f(N) - 5f(N-1) + 4f(N-2) - f(N-3)
 \end{aligned}$$

Compare results of second order regularization with zero order regularization and least square estimates for at least three different values of  $b$ . Try to find a reasonable value of  $b$  for each test case.

**Write-up** Your write up for this homework should be neat and organized. Discuss your results and findings. You will receive extra credit for creativity and exploring different things that were not required, such as differing noise levels, different test inputs, etc. You will only learn how to solve these problems by experience!