```
% gda03_09
% example of data smoothing and gap filling with
% weighted damped least squares
% supports Figure 3.11

clear all;

% need these two commands to use bconj() biconjugate gradient solver
clear F;
global F;

% initialize figure
figure(1);
clf;

% the data is a sinusoid in an auxillary varible z;
M=101;
Dz = 1.0;
z = Dz*[0:M-1]'; % z is an auxiallty variable
zmax = max(z);
mtrue = sin( 3*pi*z/zmax );

% Evaluate sine wave at the following randomly-chosen z indices
index = [1, unidrnd(M,1,floor(M/2)),M]';
N = length(index);
zobs=z(index);

% observed data is sinusoid plus random noise
sigmad=0.05;
dobs = sin( 3*pi*zobs/zmax ) + random('Normal',0,sigmad,N,1);

% PART 1:  First derivative smoothing

% the N data equations are just m = dobs.  The only trick is lining
% up the corresponding elements of m and dobs, since they are not of
% the same length
F = spalloc( N+M-1, M, 3*N );
f = zeros(N+M-1, 1);
for i = [1:N]
    F(i, index(i)) = 1;
    f(i) = dobs(i);
end

% now implement 1st derivative flatness constraints but the last m
% (that makes M-1 rows)
epsilon=1.0;
rDz = 1/Dz;
rDz2 = 1/Dz^2;
for i = [1:M-1]
    F(i+N, i) = -epsilon*rDz;
    F(i+N, i+1) = epsilon*rDz;
    f(i+N) = 0;
end

% least squares solution, using bicg()
tol = 1e-6;
maxit = 3*M;
mest = bicg( @weightedleastsquaresfcn, F'*f, tol, maxit );
```

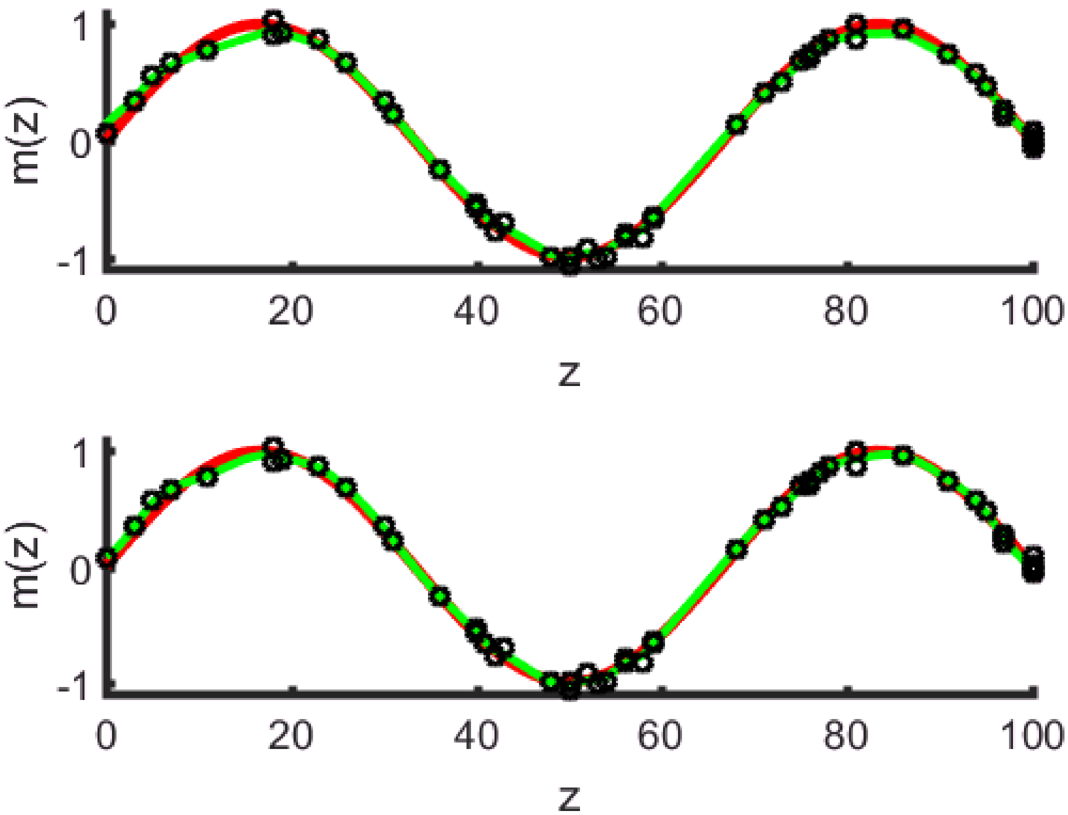
```
bicg converged at iteration 44 to a solution with relative residual 5.4e-07.
```

```
% plot
subplot(2,1,1);
set(gca,'LineWidth',2);set(gca,'LineWidth',3);
set(gca,'FontSize',14);hold on;
axis( [0, zmax, -1.1, 1.1 ] );
plot( z, mtrue, 'r-', 'LineWidth', 4 );
plot( z, mest, 'g-', 'LineWidth', 3 );
plot( zobs, dobs, 'ko', 'LineWidth', 2 );
xlabel('z');
ylabel('m(z)');

% PART 2:  Second derivative smoothing

% the N data equations are just m = dobs.  The only trick is lining
% up the corresponding elements of m and dobs, since they are not of
% the same length
F = spalloc( N+M, M, 3*N );
f = zeros(N+M, 1);
for i = [1:N]
    F(i, index(i)) = 1;
    f(i) = dobs(i);
end

% now implement 2nd derivative smoothness constraints of all interior m's
epsilon=1.0;
rDz = 1/Dz;
rDz2 = 1/Dz^2;
for i = [1:M-2]
    F(i+N, i) = epsilon*rDz2;
    F(i+N, i+1) = -2*epsilon*rDz2;
    F(i+N, i+2) = epsilon*rDz2;
    f(i+N) = 0;
end

% now implement 1st derivative flatness constraints for m's at the edges
F(N+M-1, 1) = -epsilon*rDz; F(N+M-1, 2) = epsilon*rDz;
f(N+M-1) = 0;
F(N+M, M-1) = -epsilon*rDz; F(N+M, M) = epsilon*rDz;
f(N+M) = 0;

% least squares solution, using bicg()
tol = 1e-6;
maxit = 3*M;
mest = bicg( @weightedleastsquaresfcn, F'*f, tol, maxit );
```

```
bicg converged at iteration 85 to a solution with relative residual 7e-07.
```

```
% plot
subplot(2,1,2);
set(gca,'LineWidth',3);
set(gca,'FontSize',14);
hold on;
axis( [0, zmax, -1.1, 1.1 ] );
plot( z, mtrue, 'r-', 'LineWidth', 4 );
plot( z, mest, 'g-', 'LineWidth', 3 );
```

```
plot( zobs, dobs, 'ko', 'LineWidth', 2 );
xlabel('z');
ylabel('m(z)');
```




```
% Figure 3.11.  Examples of weighted damped least squares applied to the problem of
% filling in data gaps, for two different types of prior information. (A) (Red curve)
% The true model is a sinusoid sampled with Δz=1. (Black circles) The data are noisy
% observations of the model at just a few points. (Green curve) The estimated model is
% reconstructed from the data using the prior information of flatness (small first derivative)
% (B) Same as (A) except using the prior information of smoothness (small second derivative)
% in the interior of the (0,100) interval and flatness at its ends. MatLab script gda03_09.
```