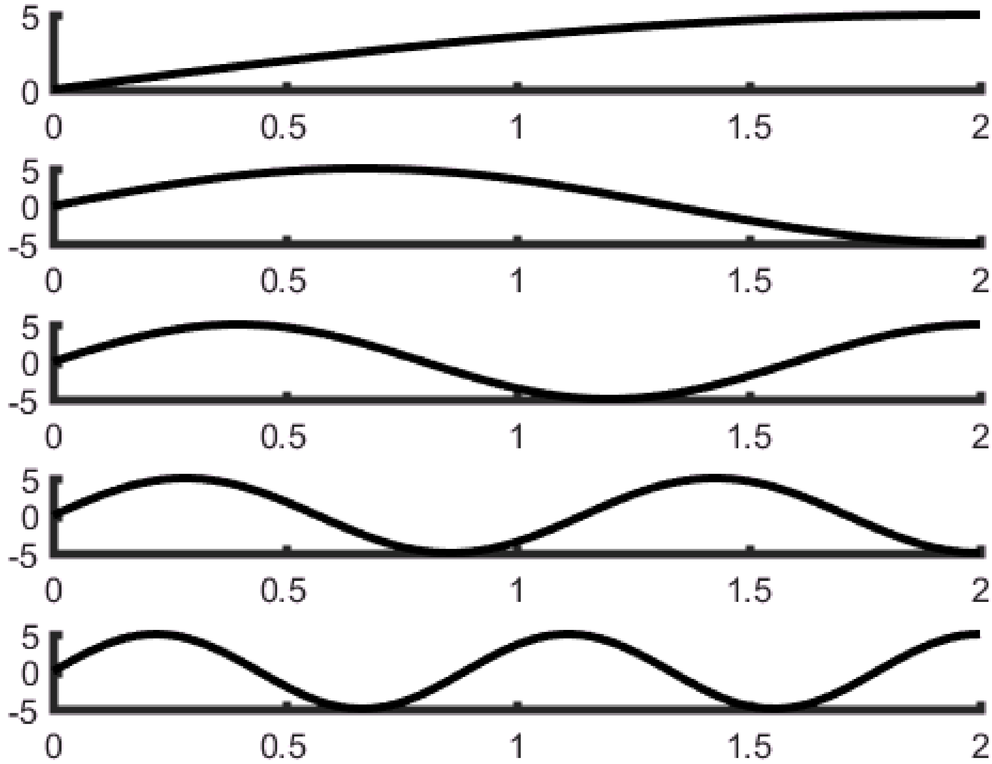
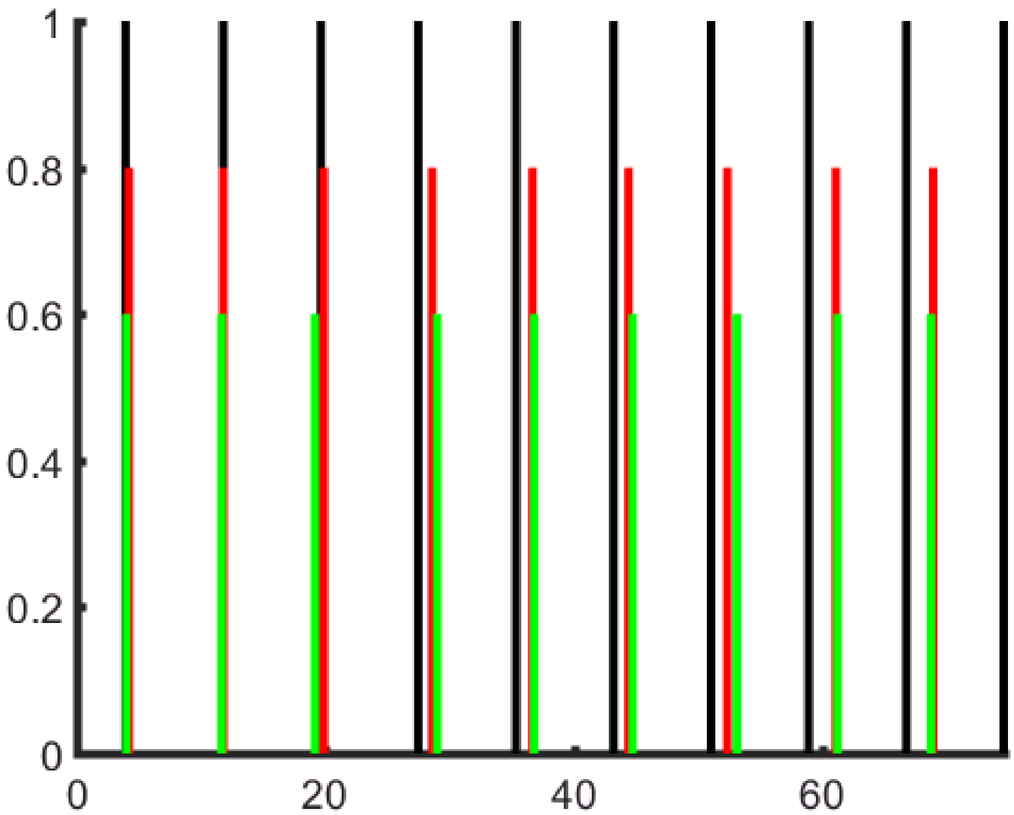
```
% gda12_14
% Supports Figure 12.21

% determine velocity structure v(x)=v0+v1(x)
% from frequencies of vibration

% unperturbed differetial equation -w^2 p = [v0+v1(x)]^2 d2p/dx2
% boundary conditions top: p=0 at x=0; bottom dp/dx=0 at x=h

% unperturbed solutions are analytic
% p(n,x) = A sin( (n-0.5) pi x / h ) n=1,2,3,...
% w(n) = (n-0.5)*pi*v0/h
% normalization so that (integral 0 to h) p(n,x) p(m,x) v0^(-2) dx = delta(n,m)
% A = sqrt( 2 v0^2 / h )

clear all;

% independent variable x
h=2;
v0=5;
Nx = 101;
xmin=0;
xmax=h;
Dx = (xmax-xmin)/(Nx-1);
x = xmin + Dx*[0:Nx-1]';

% plot some wave p's and check normalization
figure(2);
clf;
verbose=0;
if( verbose )
    fprintf('normalization (should be unity)\n');
end
for n =[1:5]
    subplot(5,1,n);
    set(gca,'LineWidth',3);
    set(gca,'FontSize',12);
    hold on;
    p = sqrt(2*(v0^2)/h) * sin( (n-0.5) * pi * x / h );
    plot(x,p,'k-','LineWidth',3);
    I = Dx*sum(p.*p)/(v0^2);
    if( verbose )
        fprintf('    n %d area %f\n',n,I);
    end
end
```



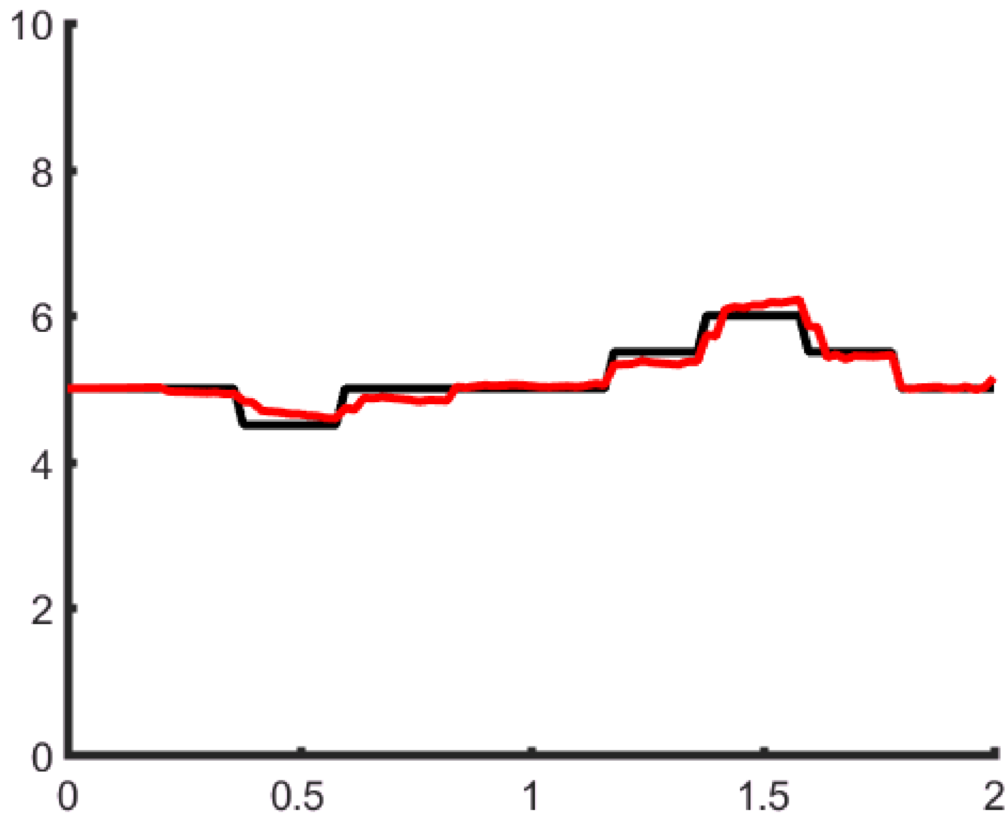

```
% Figure. Organ pipe example. First several eigenfunctions.

% build a true v1
v1true = zeros(Nx,1);
v1true(20:30)=v1true(20:30)-(v0/10);
v1true(60:90)=v1true(60:90)+(v0/10);
v1true(70:80)=v1true(70:80)+(v0/10);
figure(3);
clf;
set(gca,'LineWidth',3);
set(gca,'FontSize',14);
hold on;
axis( [xmin, xmax, 0, 10] );
plot( x, v0+v1true, 'k-', 'LineWidth', 3 );

% unperturbed frequencies
N=200;
w0 = ([1:N]'-0.5)*pi*v0/h;

% perturbed frequencies
w1true = zeros(N,1);
for n =[1:N]
    p = sqrt(2*(v0^2)/h) * sin( (n-0.5) * pi * x / h );
    w1true(n)=w0(n)*Dx*sum(v1true.*p.*p)/(v0^3);
end

% observations are perturbed frequencies plus random noise
sigmad = 0.1*w0(1);
dobs = w1true + random('Normal',0.0,sigmad,N,1);
```

```
% ladder diagram; plot only a few frequencies
Nfew=10;
figure(4);
clf;
set(gca,'LineWidth',3);
set(gca,'FontSize',14);
axis( [0, max(w0(Nfew)), 0, 1] );
hold on
for n =[1:Nfew]
    plot( [w0(n),w0(n)], [0, 1] ,'k-', 'LineWidth', 3);
    plot( [w0(n)+w1true(n),w0(n)+w1true(n)],   [0, 0.8] ,'r-', 'LineWidth', 3);
    plot( [w0(n)+dobs(n),w0(n)+dobs(n)],   [0, 0.6] ,'g-', 'LineWidth', 3);
end
```

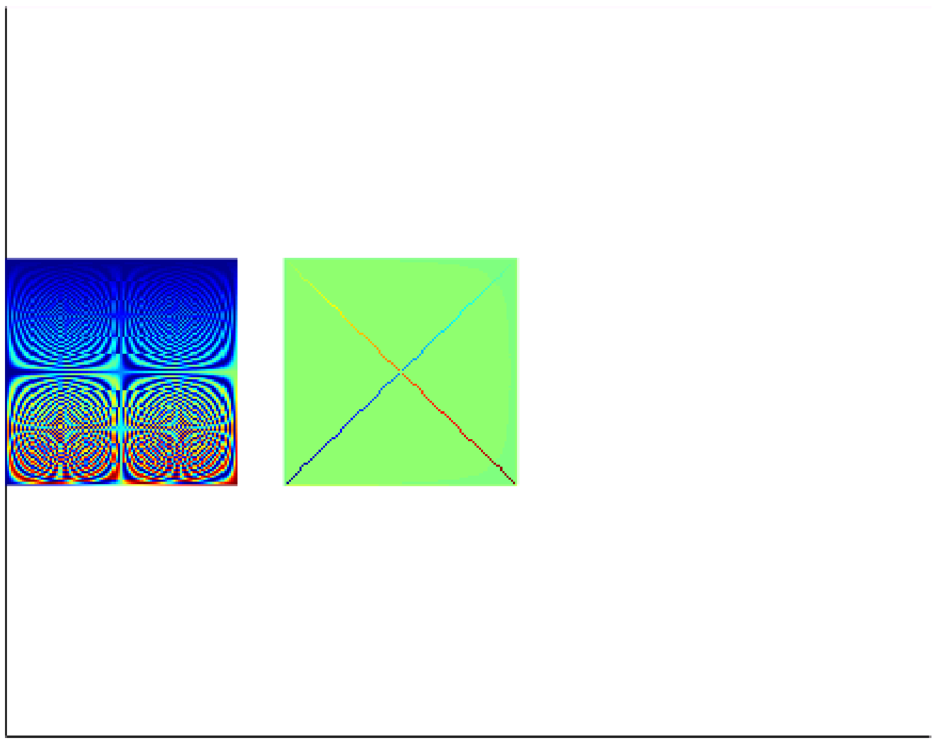



```
% Figure 12.21 Organ pipe example. (A) Ladder diagram for unperturbed (black), true (red),
% and observed (green) eigenfrequencies of the organ pipe. MatLab script gda12_14.

% model parameters are velocity perturbations

M=Nx;
mest = zeros(M,1);

% data kernel is integral linking them
if( 0 ) % brute force way
    G = zeros(N,M);
    for i=[1:N]
    for j=[1:M]
        pij = sqrt(2*(v0^2)/h) * sin( (i-0.5) * pi * x(j) / h );
        G(i,j) = Dx*w0(i)*(pij^2)/(v0^3);
    end
```

```
    end
else % tensor product way
    pijm = sqrt(2*(v0^2)/h) * sin( pi * ([1:N]'-0.5)*x' / h );
    G = Dx*(w0*ones(1,M)).*(pijm.*pijm)/(v0^3);
end

% damped least squares solution, but weight lower frequencies more
e2 = 1.0e-6;
GMG = (G'*G+e2*diag([1:M].^-1))\(G');
mest = GMG*dobs;
figure(3);
plot( x, v0+mest, 'r-', 'LineWidth', 3 );
```




```
% Figure 12.21 Organ pipe example. (B) True (black) and estimated (red) velocity structure.

% resolution matrix
R = (GMG*G);
gda_draw(G,' ',R);
```



```
% Figure 12.21 Organ pipe example. (C) Data kernel G. (D) Model resolution matrix R.
```