
```

%mp3_vig.m

%q1.1
figure(1);
x = zeros(1000,1); x(650:850) = 1;
plot(x),title('q1.1');

%q1.2
c = conv(x,x);
sprintf('The length of c %d',length(c))

%q1.3
figure(2);
plot(c),title('q1.3');

%q1.4
%{
    The Shape of x convolved with itself is a triangle offset to the right
of the center. The shape is due to the fact that as the two signals overlap,
the area increases to a maximum value when the two signals are directly on
top of one another and decreases as they move apart. The horizontal position
is because one of the signals is flipped. If they were not flipped, th
peak would be directly in the middle, since the two signals would overlap
perfectly when their indices line up.
%}

%q1.5
figure(3);
plot(conv(c,c)),title('q1.5');

%q1.6
%{
    This plot also rises to a peaked value at some point to the right of
the plot, but it also looks more like a gaussian than a triangle. Overall,
this function is smoother than the function of x convolved with itself.
This smoothing effect is because the convolution is essentially performing
a weighted average of sorts over the entire curve, where the weights are the
curve itself. Once again, the position of the peak is due to the flipping of
one of the convolved functions.
%}

%q1.7
u = fft(c);
plot(abs(u(1:2:end)), 'DisplayName', 'FFT(c)'); set(gca, 'xscale', 'log')

%q1.8
hold on;
plot(abs(fft(x).*fft(x)), 'DisplayName', 'FFT(x)*FFT(x)');title('q1.8');
legend()
xlim([0 100])
%{
    The two plots are nearly identical because the fourier transform of two

```

convolved functions is just the product of the fourier transforms of these two functions. This is an established property of the fourier transform and follows from the fact that you can split the fourier exponential term into two parts.

```
%}
```

```
%q1.9
```

```
%{
```

This is due to the discrete nature of our data. The convolved dataset is twice as long as the elementwise product dataset.

```
%}
```

```
load('PKD_RFs_MP3.mat')
```

```
%q2.1
```

```
figure(4);
```

```
hold on
```

```
plot(time,filtP,'b','DisplayName','P-Wave');
```

```
plot(time,filtSV,'r--','DisplayName','S-Wave');title('q2.1/q2.4');
```

```
xlabel('Time (s)'); ylabel('Intensity')
```

```
%{
```

It seems like there is a slight delay between the P-Wave and the S-wave, with the S-wave lagging behind the P-wave. The S wave is also slightly more complex because it has some late stage activity in the signal, whereas the S-wave simply dies away.

```
%}
```

```
%q2.2
```

```
figure(5);
```

```
hold on
```

```
plot(time-time(1),meanRF,'DisplayName','Receiver Function');title('q2.2');
```

```
xlabel('Lag Time (s)'); ylabel('Intensity');
```

```
legend();
```

```
%q2.3
```

```
%{
```

The largest signal feature occurs at roughly 5 seconds of lag time, whcih corresponds roughly to 50km of depth. The geologic feature that causes this signal is most likely the moho boundary between the crust and the mantle.

```
%}
```

```
%q2.4
```

```
figure(4);
```

```
v = conv(filtP,meanRF);
```

```
plot(time,v(1:length(filtP)), 'DisplayName','conv(filtP,meanRF)')
```

```
legend();
```

```
%{
```

This convolution looks very similar to the S-Wave signal

```
%}
```

```
%q2.5
```

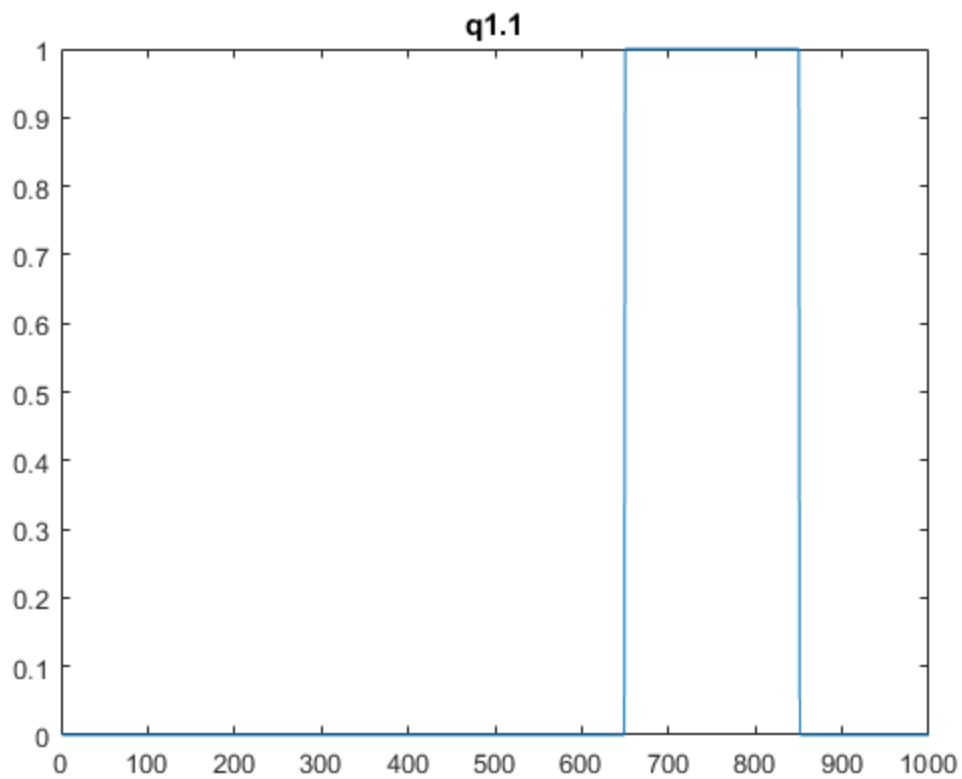
```
%{
```

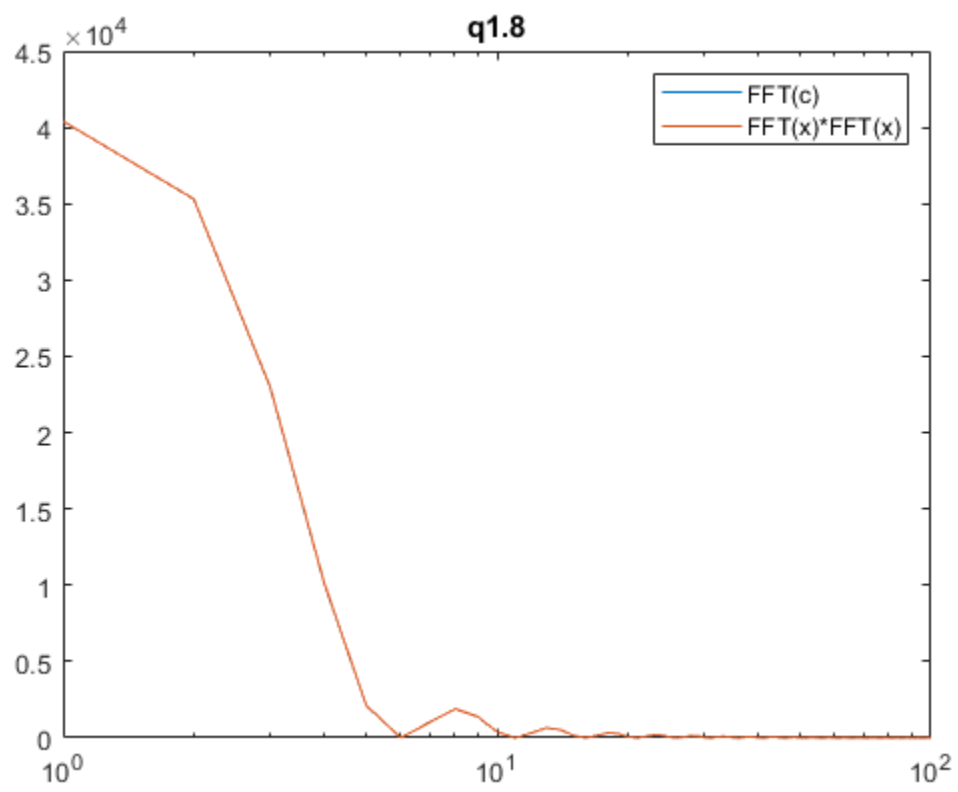
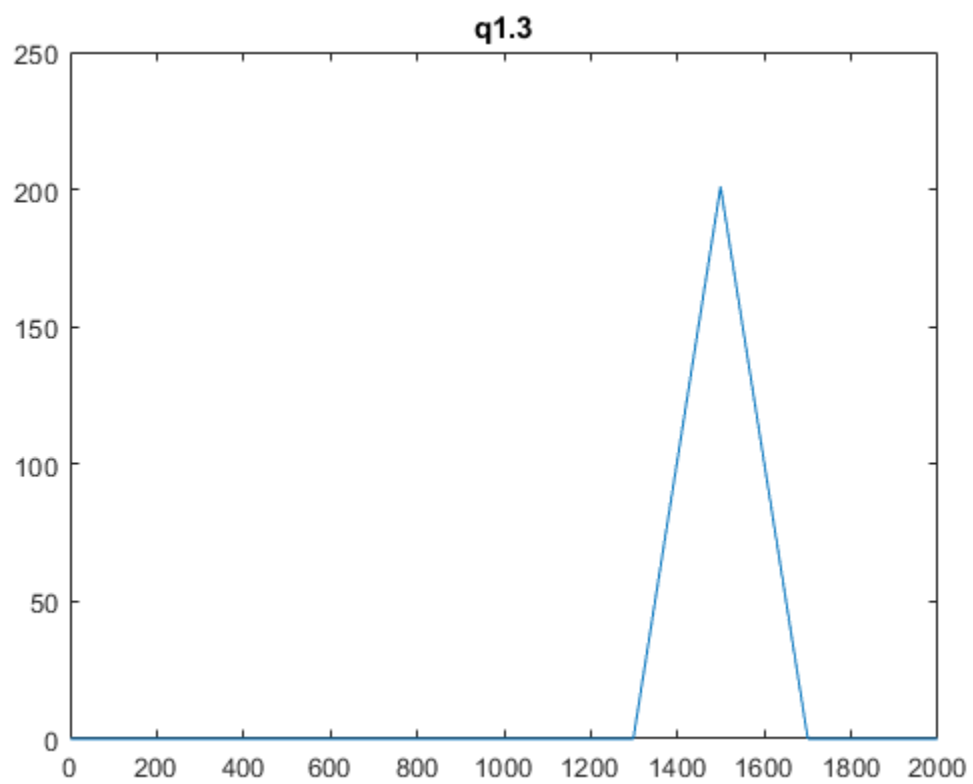
In some spots, I was able to glean some general information about the receiver function, but in other spots, I was not able to. For example, I could have predicted that the largest peak was not quite at the beginning since the S-Wave seemed to lag behind. I would have predicted the RF to also have a late stage peak, similat to the S-wave signal, but this is not the case.

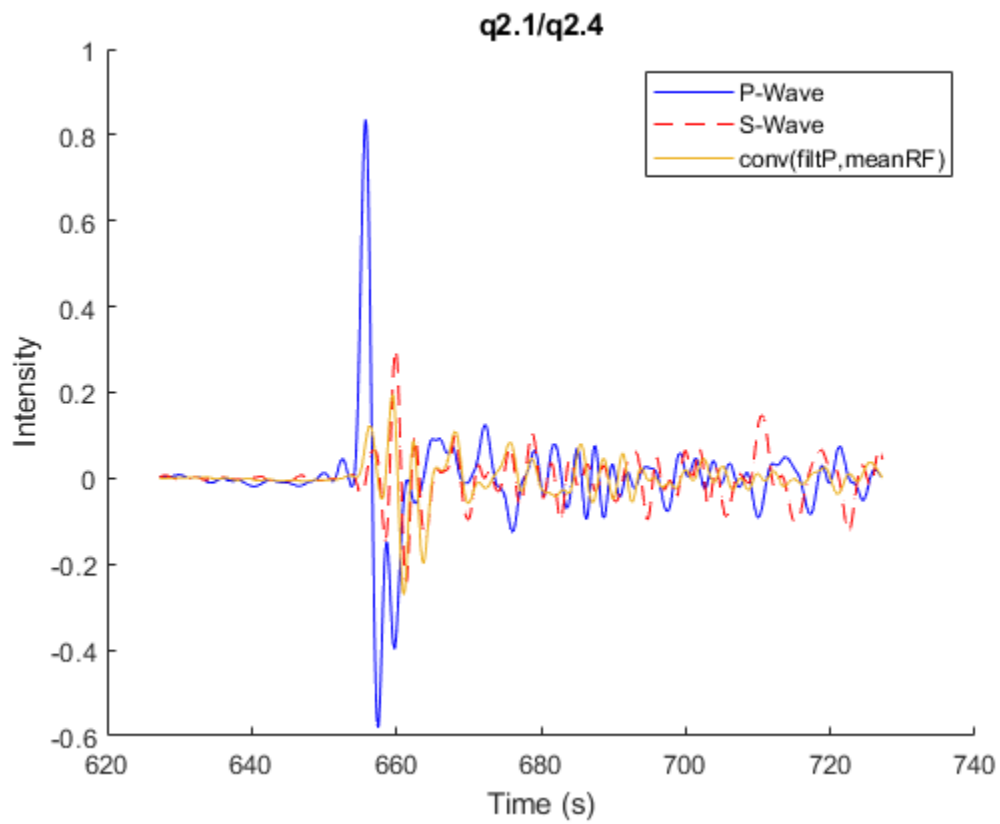
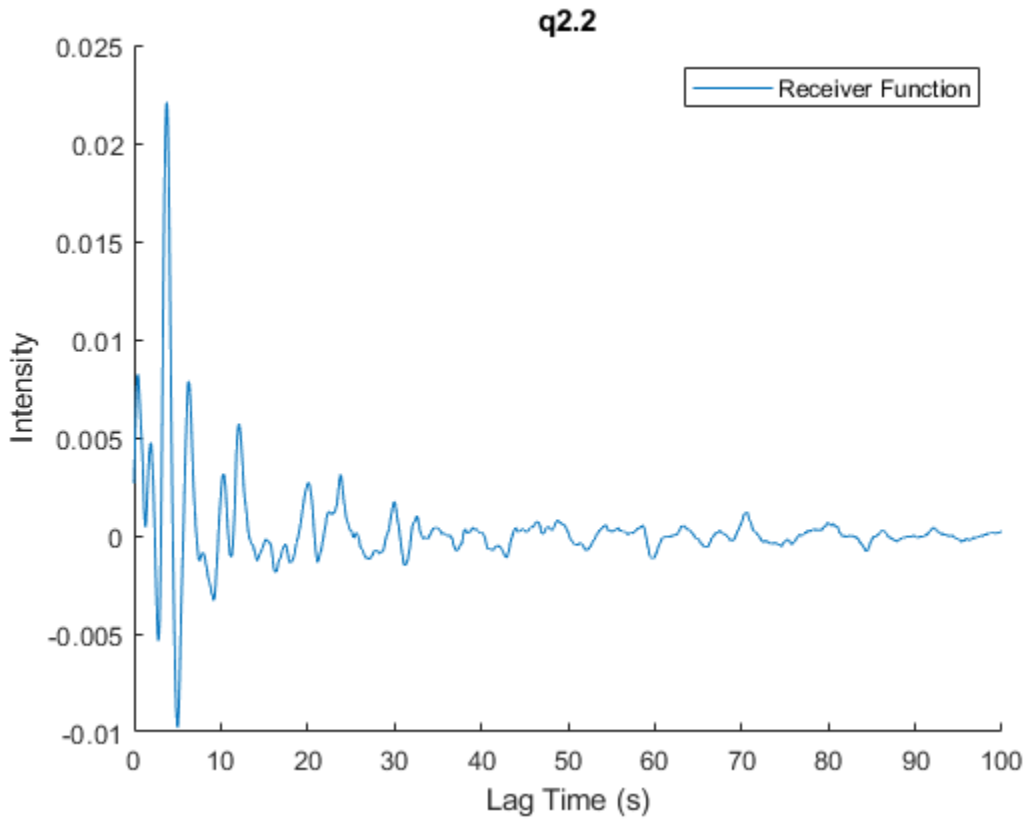
%}

ans =

'The length of c 1999'







Published with MATLAB® R2023b