

# Oblig1

BiHui Chen

## Oppgavel:

Implementasjon av konvolusjon i matlab.

```
%Forklaring av funksjon:
%Utfoere konvolusjon mellom signal h[n] og x[n], tilsvare
argument h og x.
%Argument ylen, tar i mot en int 0-1
function y = konvin3190 (h , x, ylen)
%Velge lengde av y, ved aa sender inn 0 eller 1 som tredje
argument
% kan man velge mellom lengden til konvolusjonen
if ylen == 0
    y = zeros(1,length(x));
elseif ylen == 1
    y = zeros(1,(length(h)+length(x)-1));
else
    disp("Feil argument: ylen")
    return;
end

%Utfoere konvolusjon
for n = 1:length(y)
    for k = 1:length(h)
        if (n-k+1) > 0 & (n-k+1) <= length(x)
            y(n) = y(n)+(h(k)*x(n-k+1));
        end
    end
end
end
```

### Oppgavelb:

Dette er implementasjon av Fourier transform i matlab.

```
function [X,f] = frekspekin3190(x,N,fs)
%Denne funksjonen tar imot signal x, antall sample punkt N
og
%samplefrekvens fs. Og renege frekvens response X og f som
er tilhoerende frekvens til X.
%Argument: x(array), N(int), fs(int)
%Return: X(array, lengde N), f(int)

%anta at x er en kausal signal, slik sampling gaa fra 1
til N, matlab index
%begynner fra 1.
omega = linspace(-pi,pi, N);
X = zeros(1,N);
for n = 1:N
    w = omega(n);
    sum = 0;
    for k = 0:length(x)-1
        if k == 0
            sum = sum + (x(1)*exp(-j*w*k));
        else
            sum = sum + (x(k+1)*exp(-j*w*k));
        end
    end
    X(n) = sum;
end
%f er den unormalisert frekvensspekteret med fysisk
frekvens i HZ
%f = normalisert frekvensspekteret * fs
f = (1/(2*pi))*omega*fs;
end
```

### Oppgave 1c:

Koden består av matlab kode for selv oppgaven, og en delta funksjon for å lage delta funksjon.

```
function d = delta(n,x)
%Dette funksjon ta imot en n vector og en integer x
%Return en vektor paa lengde n, der d(x+1) er 1 og resten
er 0.
d = zeros(1,length(n));
%siden matlab har array index fra 1 dermed maa vi gjoere
en variabel skift.
d(x+1) = 1;
end

clc;
f1 = 10; f2 = 20; fs = 100; t_lengde = 5;
%t_lengde = T*n = n/fs => t*fs = n
n = 0:(t_lengde*fs);

%Signal x, y, h
x = sin(((2*pi*f1)/fs).* n) + sin(((2*pi*f2)/fs).* n);
h =
(1/5)*(delta(n,0)+delta(n,1)+delta(n,2)+delta(n,3)+delta(n
,4));
y = konvin3190(h,x,1);

%Frekvensspekteret X,Y
[X,f_x] = Frekspekin3190(x,(t_lengde*fs),fs);
[Y,f_y] = Frekspekin3190(y,(t_lengde*fs),fs);
[H,f_h] = Frekspekin3190(h,(t_lengde*fs),fs);

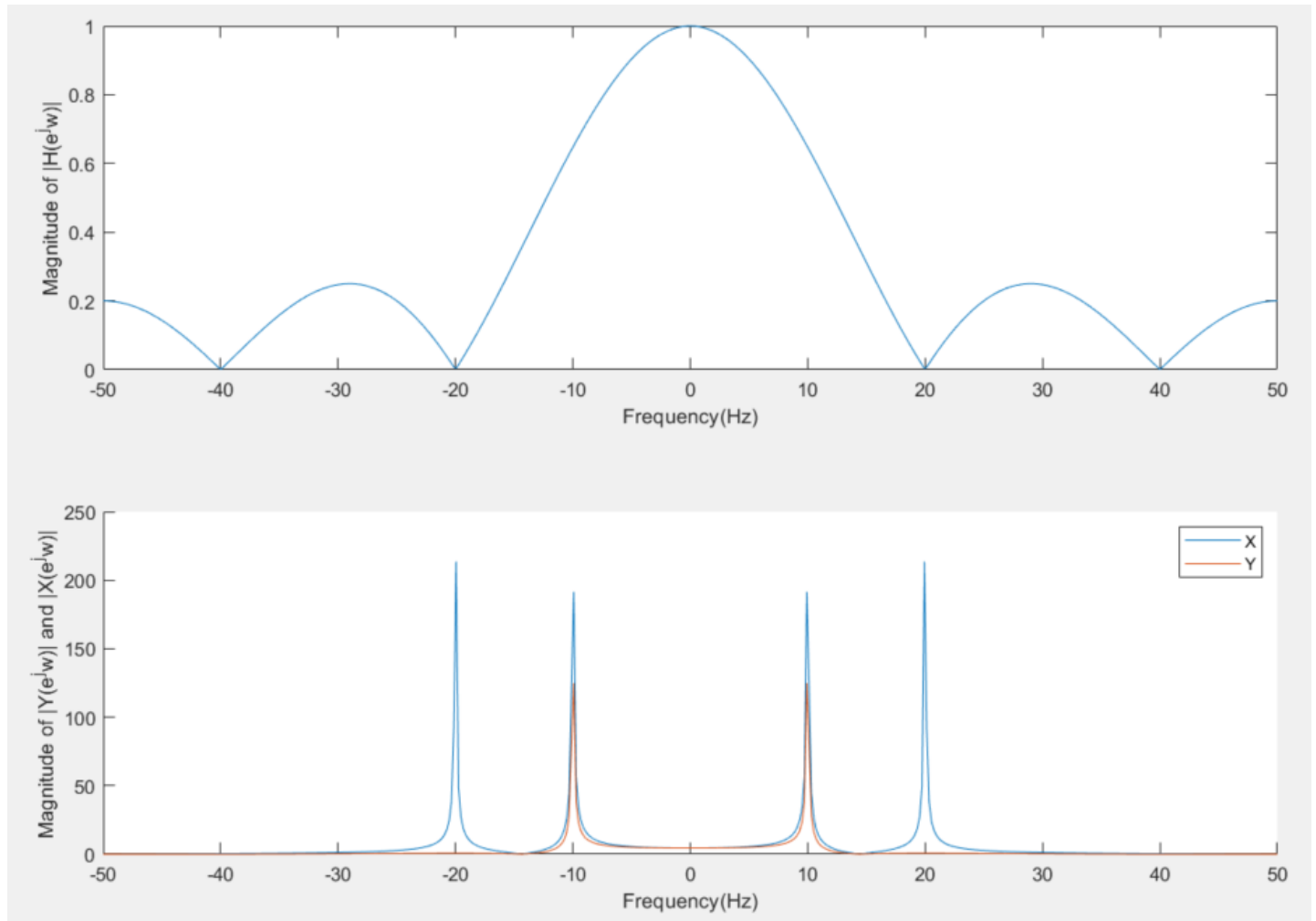
tiledlayout(2,1);
ax1 = nexttile;
plot(ax1,f_h,abs(H));
xlabel('Frequency(Hz)');
ylabel('Magnitude of |H(e^jw)|');

ax2=nexttile;
hold on;
plot(ax2,f_x,abs(X));
plot(ax2,f_y,abs(Y));
xlabel('Frequency(Hz)');
ylabel('Magnitude of |Y(e^jw)| and |X(e^jw)|');
legend('X','Y');
hold off;
```

%H = Y/X, saa ved 20hz, 40hz er system responsen minst, og Y er  
%veldig smaa mens X er veldig stor.

%Ved 0hz er system responsen stoerst, og Y/X ca 1.

Plott til oppgaven: Plottene er plottet fra -pi til pi, oppgitt i frekvens i x-aksen.

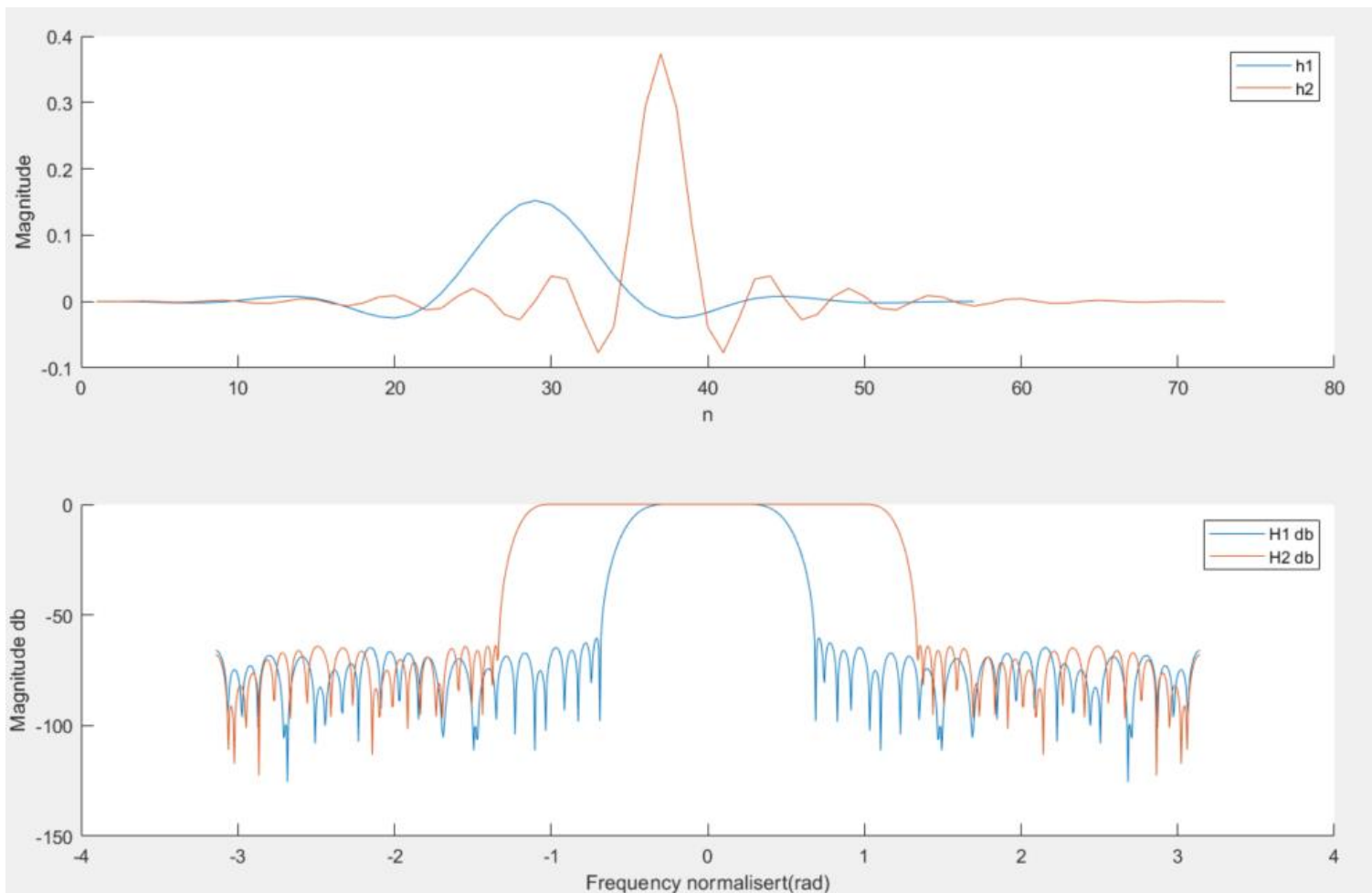


Fra frekvens spektret ser vi at frekvensen fra 0 til 10hz slipper lettere gjennom enn andre frekvensene. Det er et nullpunkt ved 20hz og et nullpunkt ved 40hz. Dermed ingen av disse 2 frekvensene slippes gjennom. Vi ser også fra figuren under. Ved 20hz er  $X(e^{jw})$  veldig høy, mens  $Y(e^{jw})$  er 0.

I X og Y plotten, ved 10 hz ser vi at  $|X|$  er større enn  $|Y|$ . Det forklare også at  $|H|$  i disse områdene er under 1. Siden for en input X gir output Y, som er mindre enn X.

## Oppgave2a:

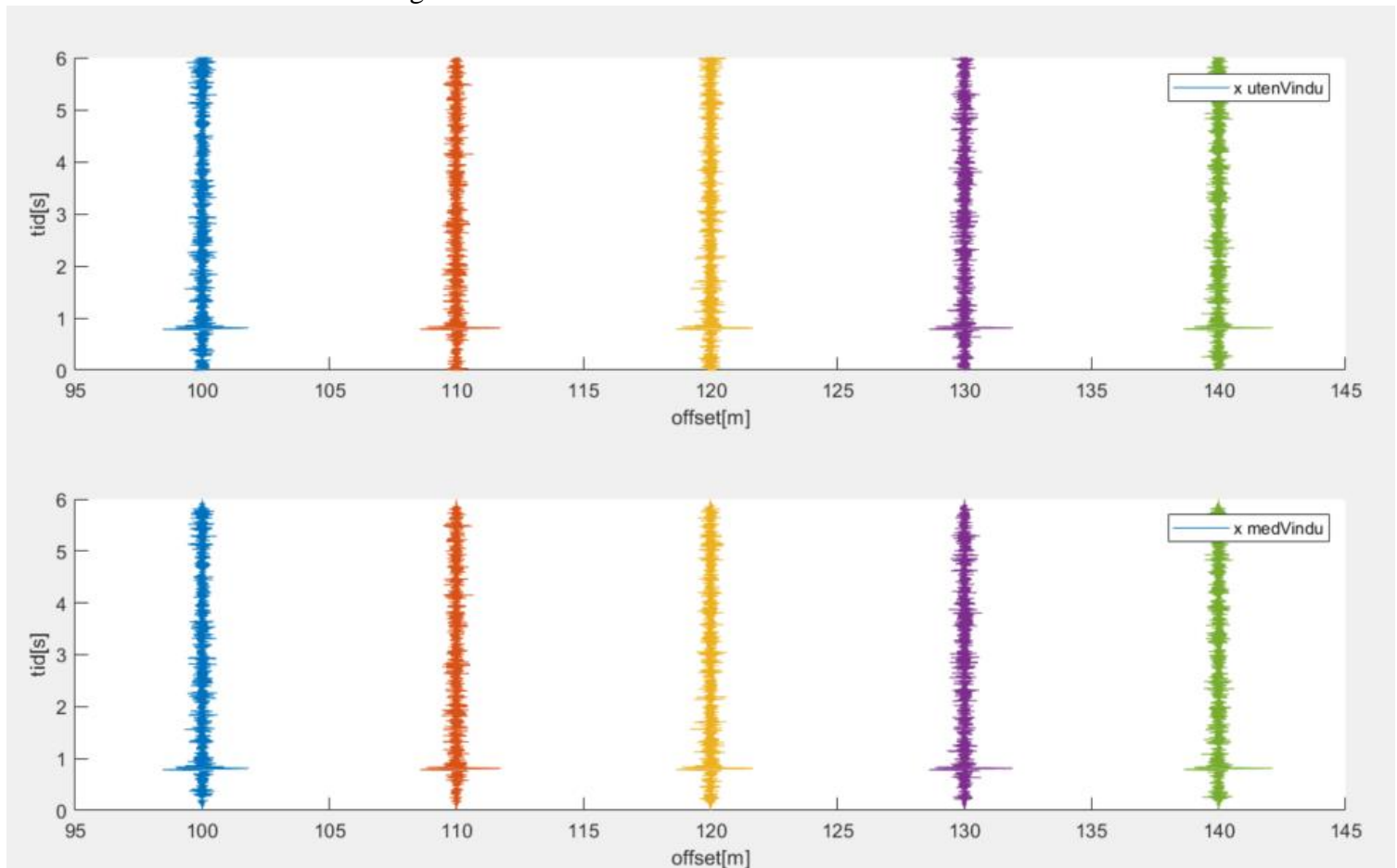
Plott til oppgave2:



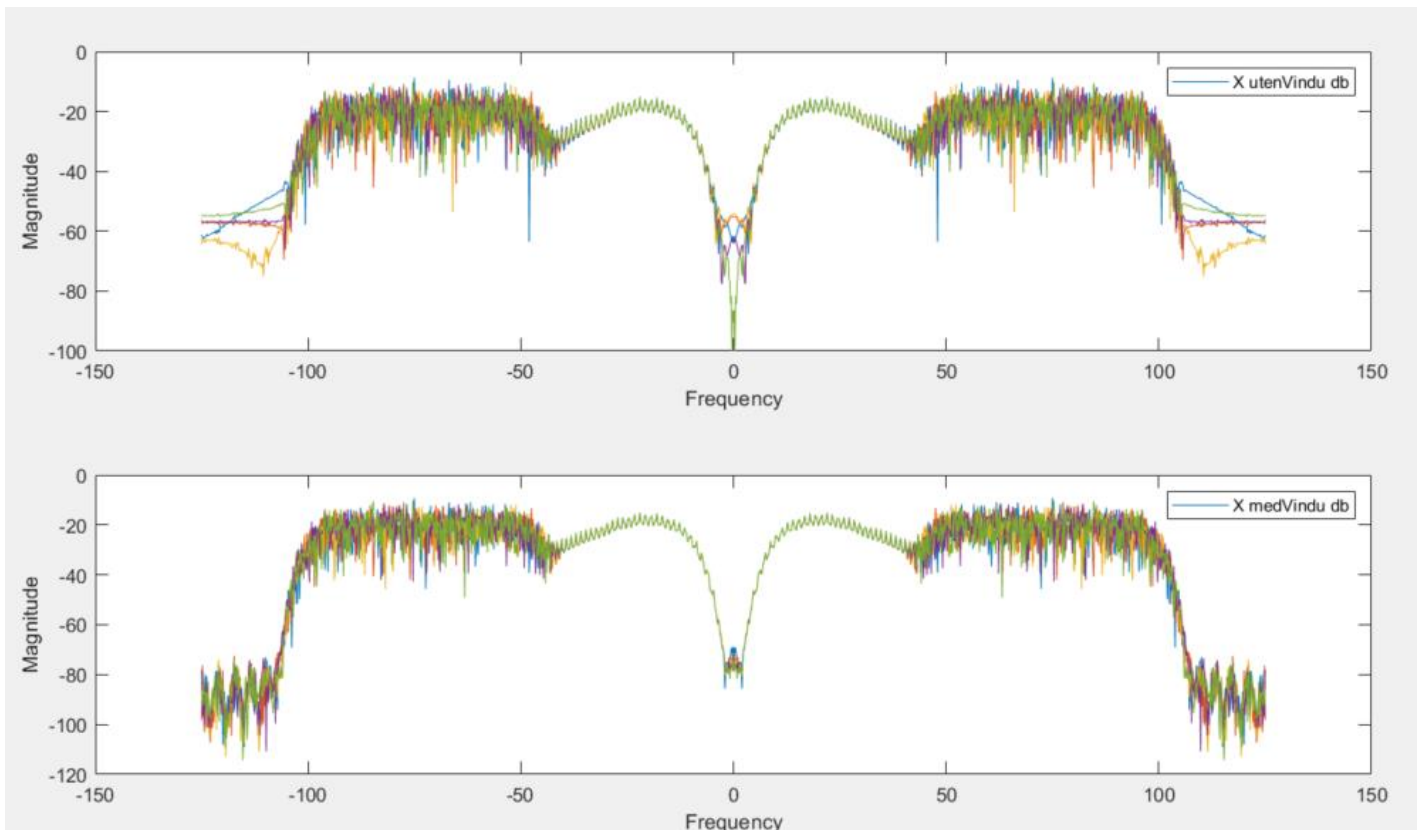
Fra figuren ser vi at filter H1 har mindre passband enn filter H2. Det vil si at noen høye frekvens kan passere filter h2, men disse passere ikke filter h1. Dersom vi ønsker bare veldig lave frekvenser, så er det bedre å bruke h1 enn h2. Siden h2 vil slipper gjennom de lave frekvensene, samtidig slipper gjennom noen høye frekvenser som vi ikke ønsker.

## Oppgave2b:

Plott av x uten vindu og x med vindu:



Plott av X uten vindu og X med vindu.



Siden høyre frekvensen vil absorberes av de dypere lagene. Når vi skal se på seismiske signal så skal vi se på refleksjon av sedimentær lagene om inneholder de lave frekvensene. Dermed er delen til høyre og venstre i frekvens spektret støy og delen nærmest midten er signal.

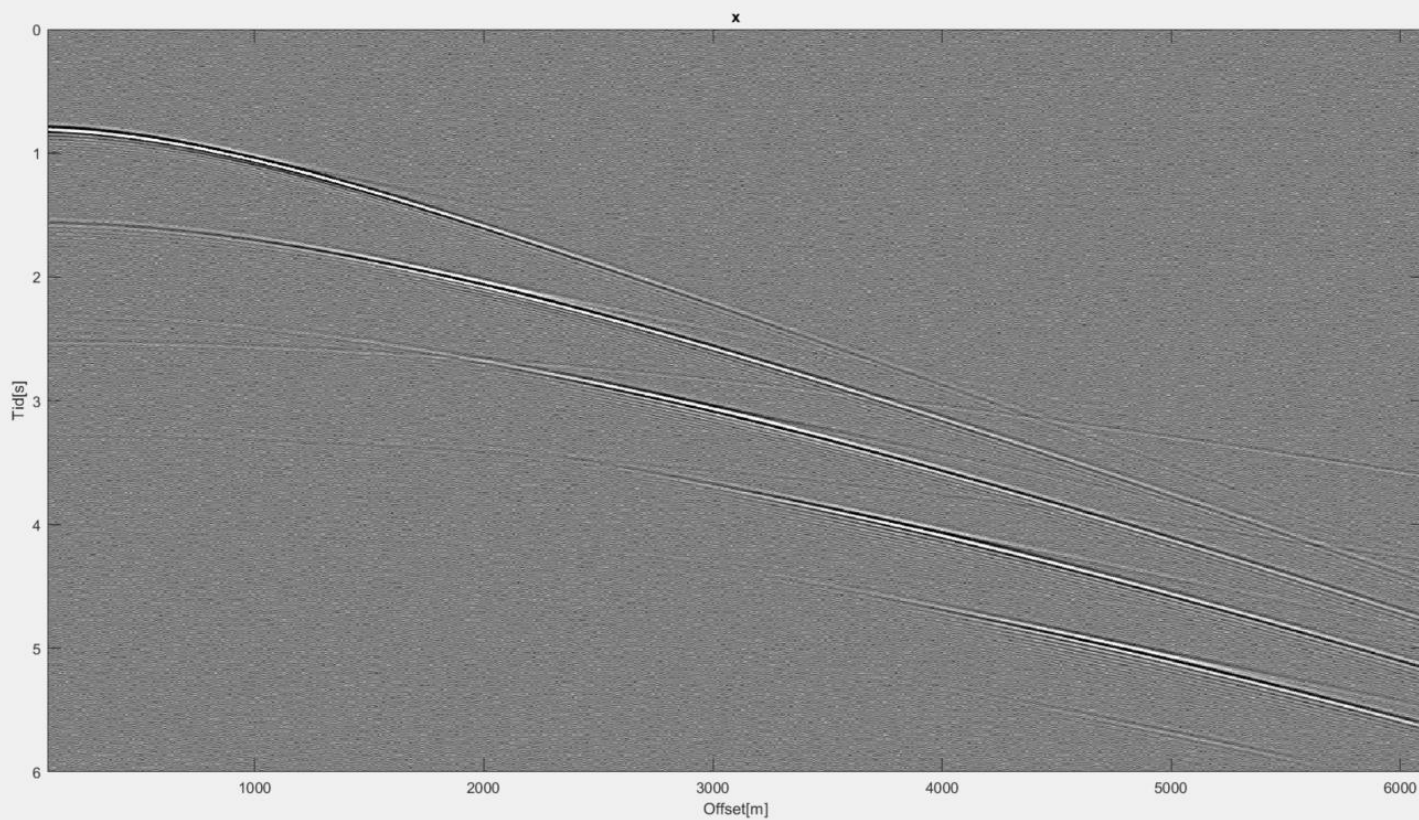
Vindu funksjonen jeg brukte i denne oppgaven:

```
W = tukeywin(length(t),0.1);
```

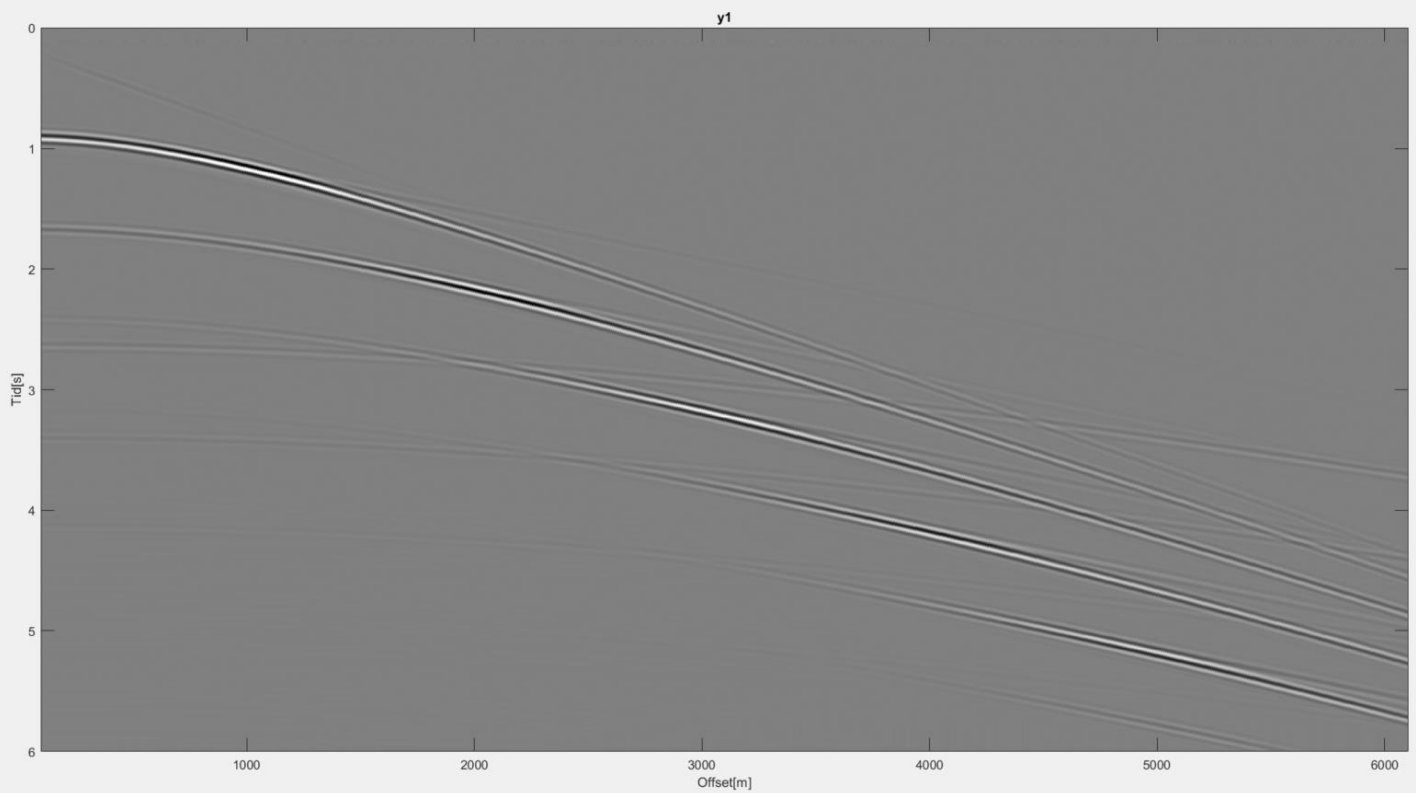


### Oppgave2c:

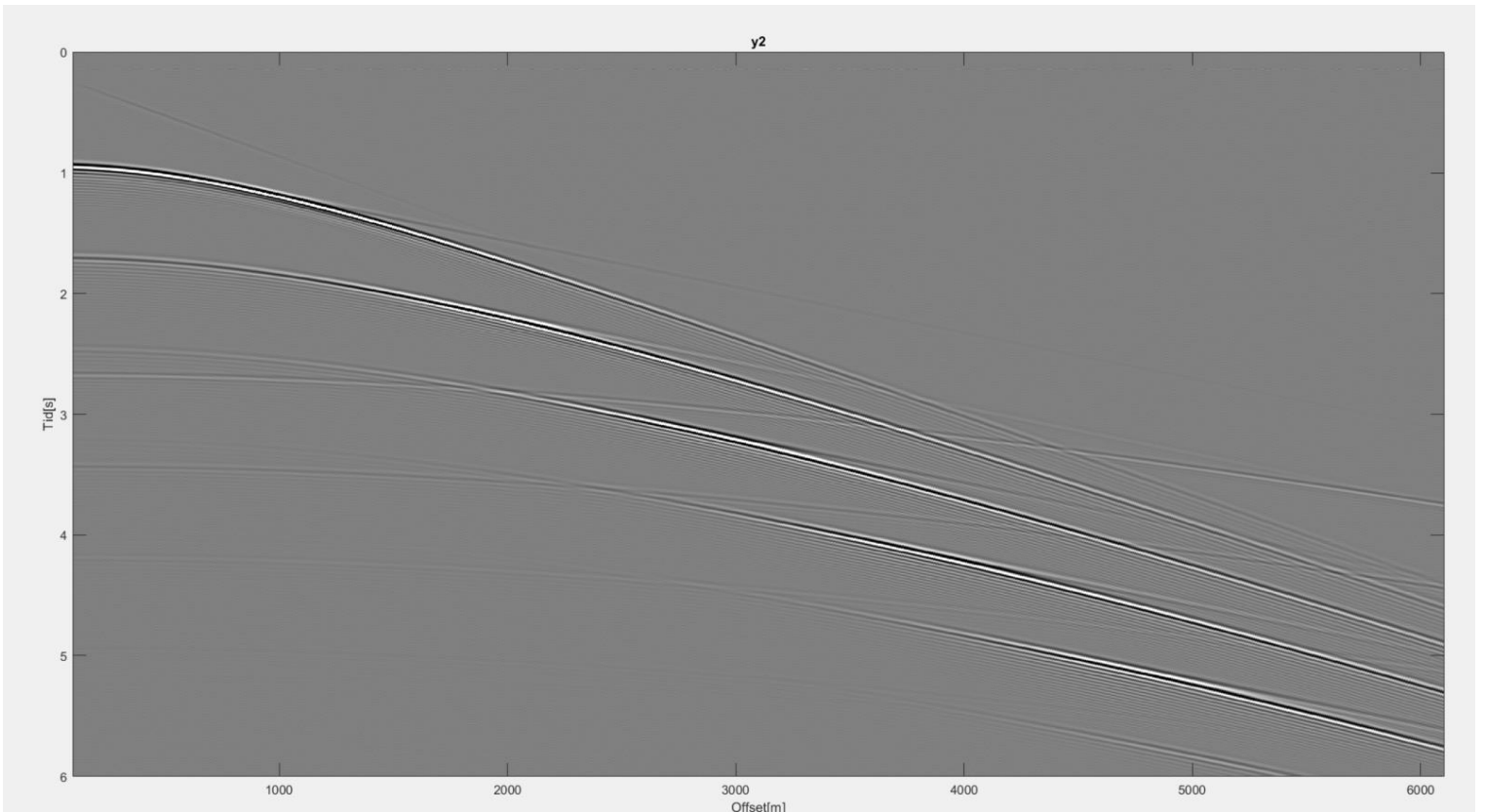
**Plott av signal x:** Som er ufiltrert signal, med støy. Støyen gjør blant annet signalene er vanskeligere å lese fra grafen, sammenlikner med filtret versjon av x signalet.



**Plott av y1:** Som er den filtret versjon av signal x med filter h1. Slik som forventet, gir y1 best resultat av x og y2 plott. Siden den filtrere vekk mest støy som er på grunn av høyere frekvens.



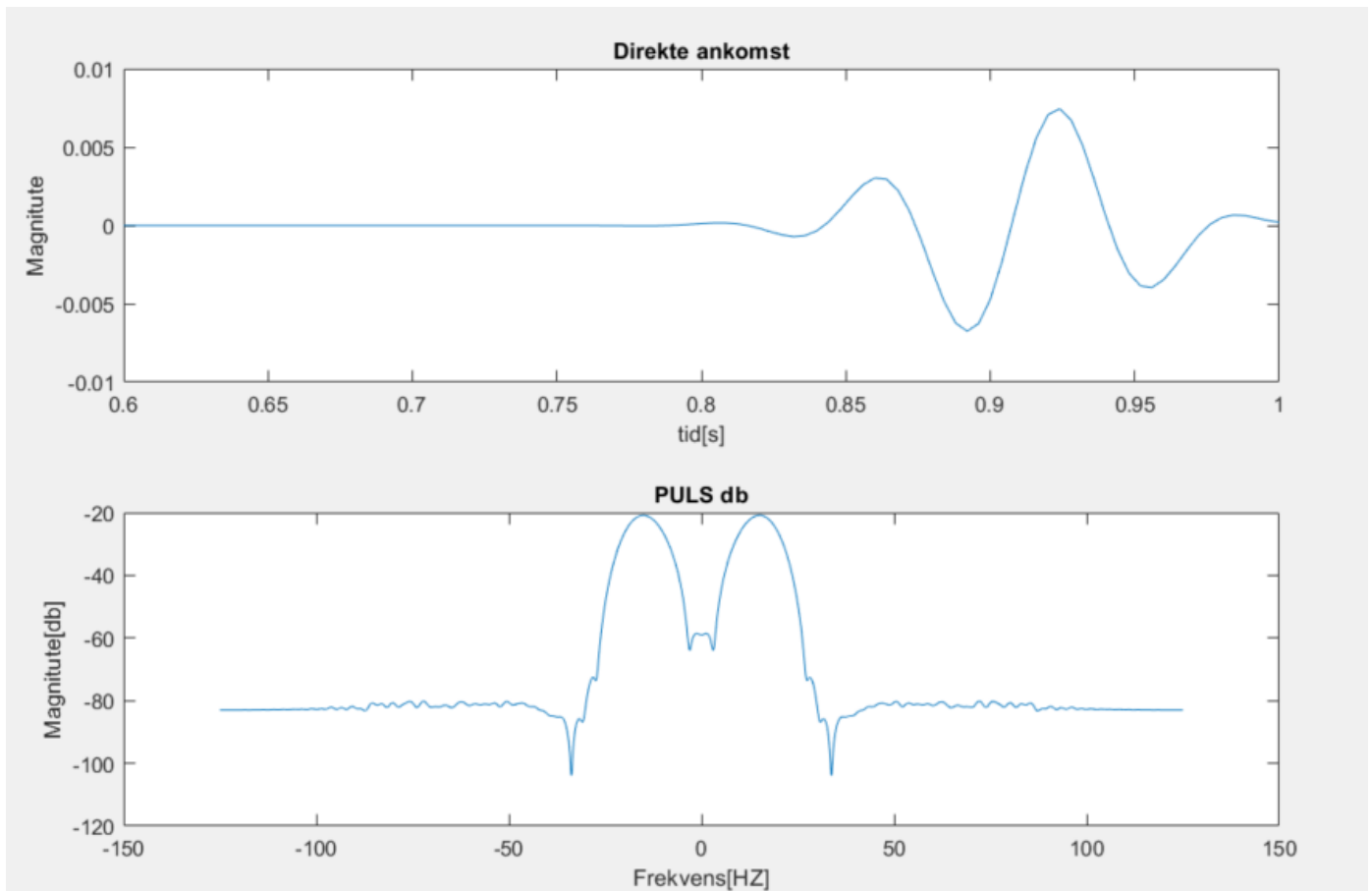
**Plott av y2:** Dette er filtret versjon av signal x med å bruke filter h2. Siden filter h2 slipper noen høy frekvens gjennom, som ikke slippes av h1. Derfor ser vi at den har en del støy sammen med refleksjonene som plottes. Men vi ser at støyen har blitt redusert sammenlikner med plotten av x signalet.



### Oppgave 3a:

Trasen ved offset 1.

Plott av direkte ankomst og frekvensen av direkte ankomst i db:

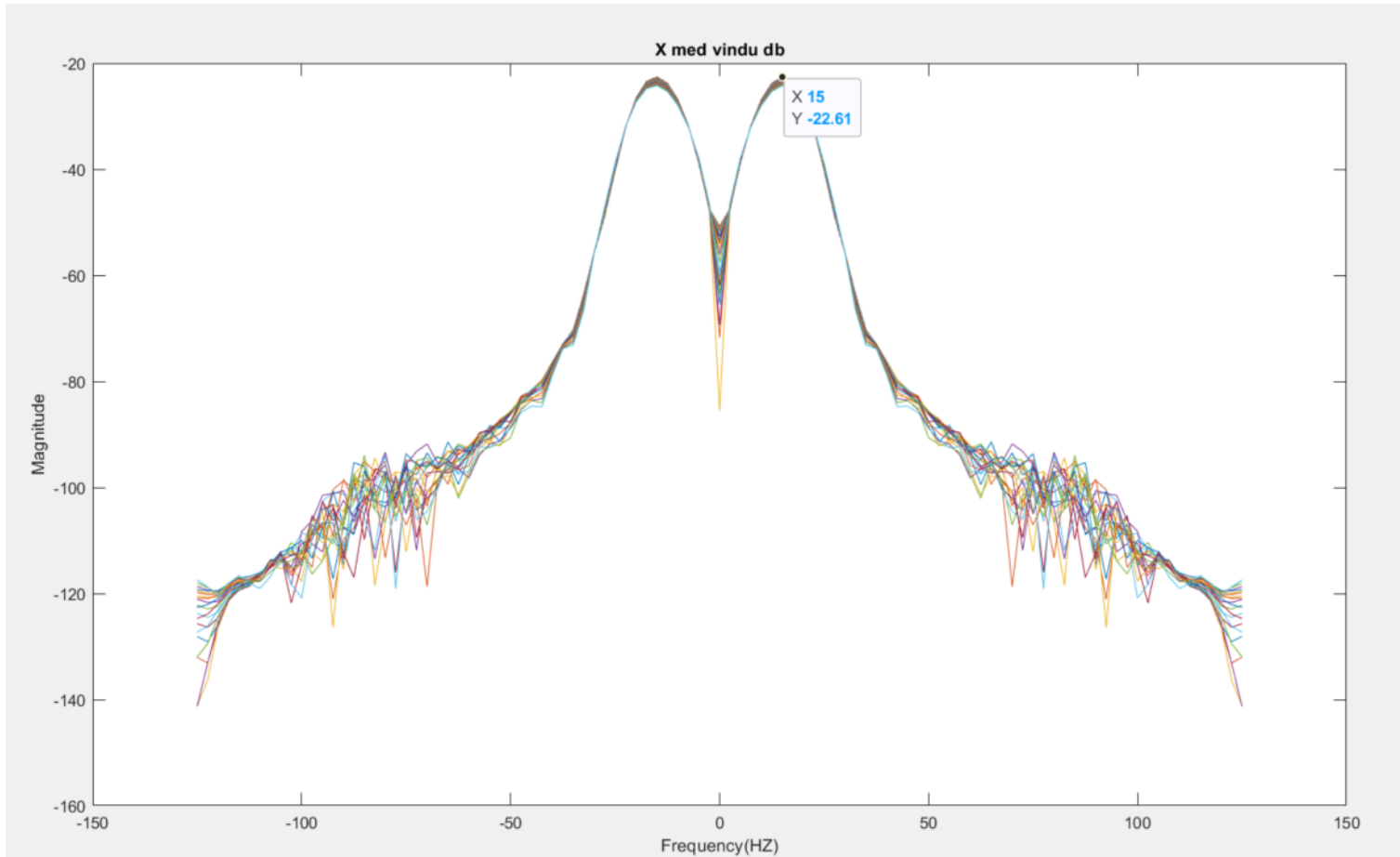


Her så valgte jeg tidsvindu fra 0.6s-1s. Under ser vi en plott av frekvens spektret i db med  $20\log(|H(e^{j\omega})|)$ . Fra dette spektret, ser vi at signalet inneholde mest frekvens mellom 5hz-25hz.

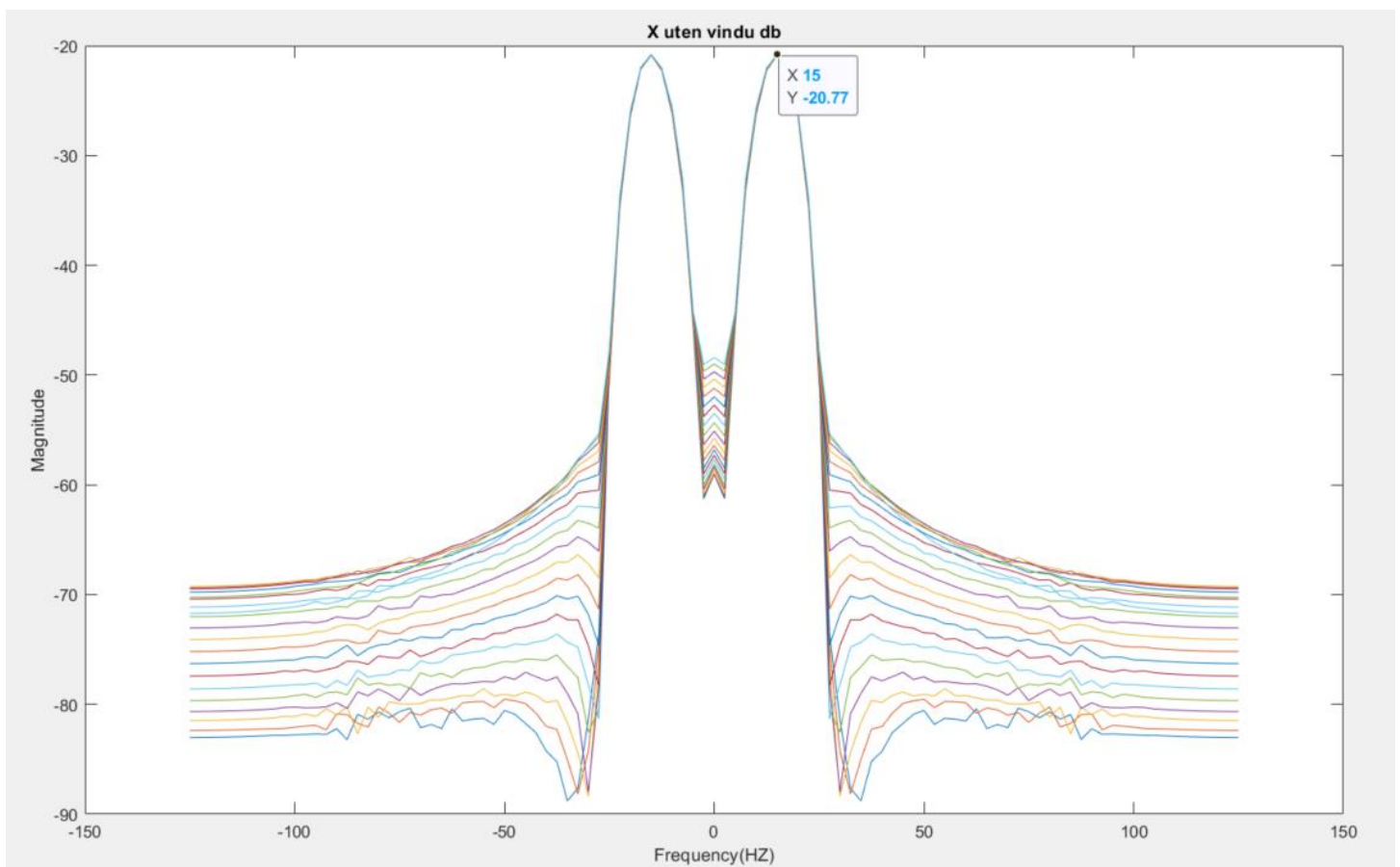
### Oppgave 3b:

Vindu: `W = tukeywin(length(x_utenVindu), 0.5);`

Plott av `X_db` med vindu:



Plott av X\_db uten vindu:



Vi ser at ved bruk av vindu kan vi redusere magnituden til frekvensene vi ikke ønsker å se på. Men samtidig så vil valg av vindu redusere magnituden til toppen vi ønsker ser på. Dermed er parameter som vi sender inn for å lage vindu viktig.

Den dominante frekvensen ser vi er ca 15HZ, lest fra grafen.

**Oppgave 3c:**

Vertikal oppløsning =  $\lambda/8$

Der  $\lambda = (3000\text{m/s})/15\text{HZ} = 200\text{m}$

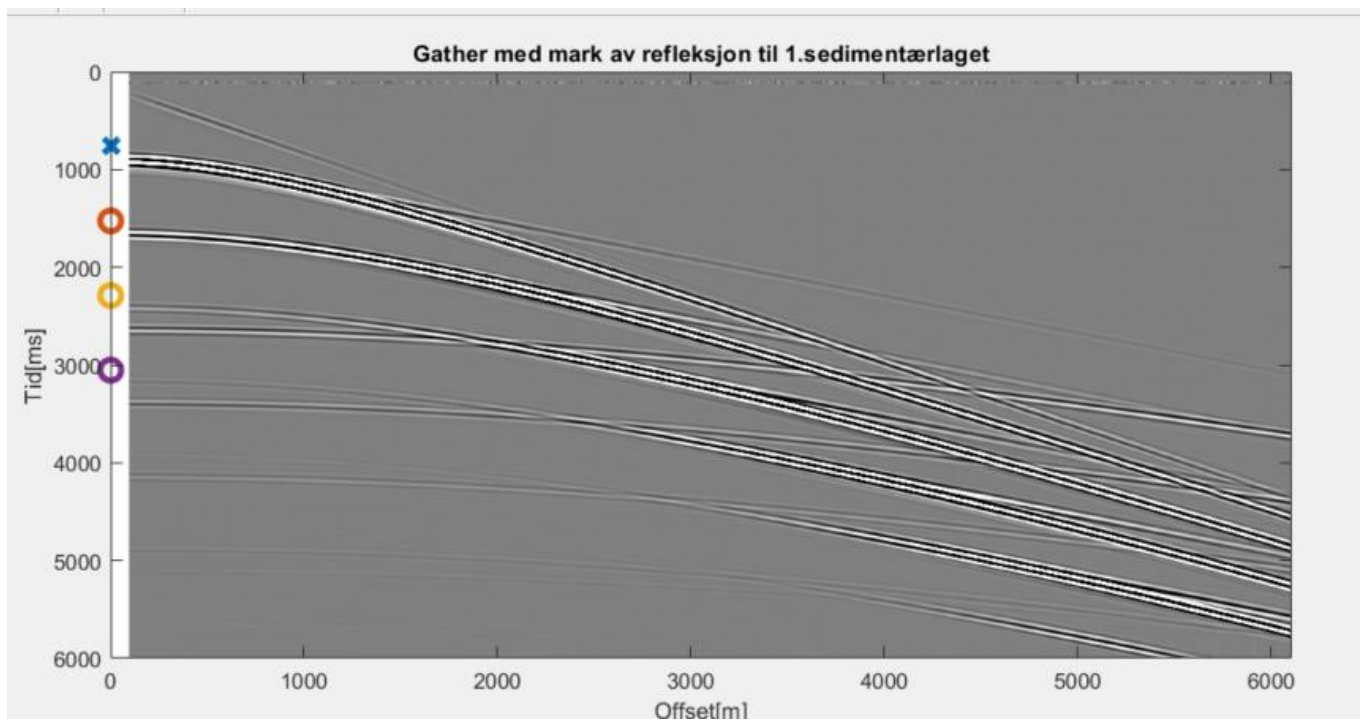
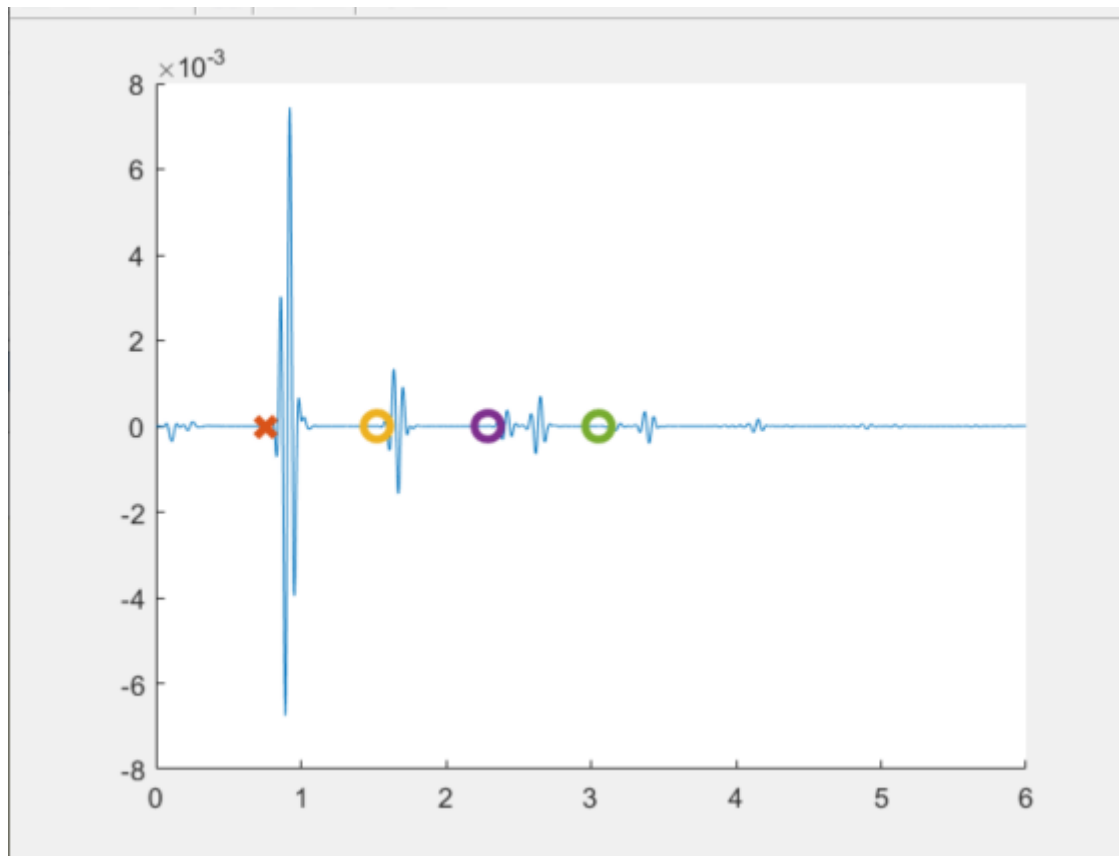
Vertikal oppløsning =  $200\text{m}/8 = 25\text{m}$

Nøyaktighet vertikalt skiller to lag er ca.25m.

#### Oppgave 4a:

Plotten av refleksjon av første sedimentærlaget. Der hoved refleksjon er merket med kryss.

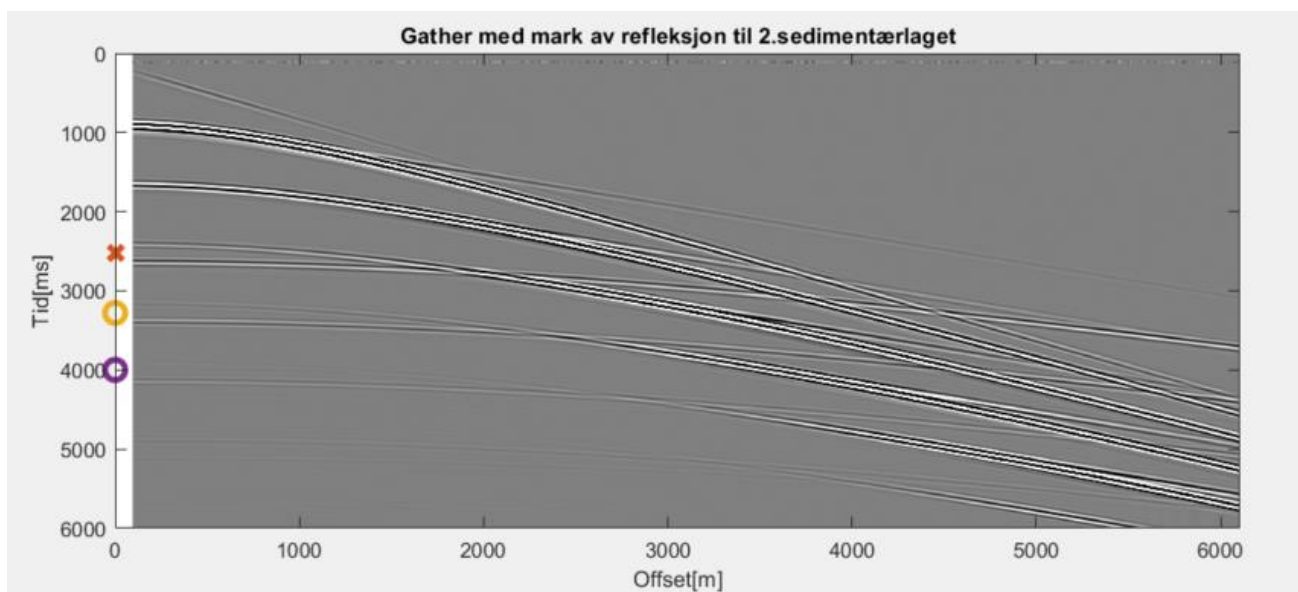
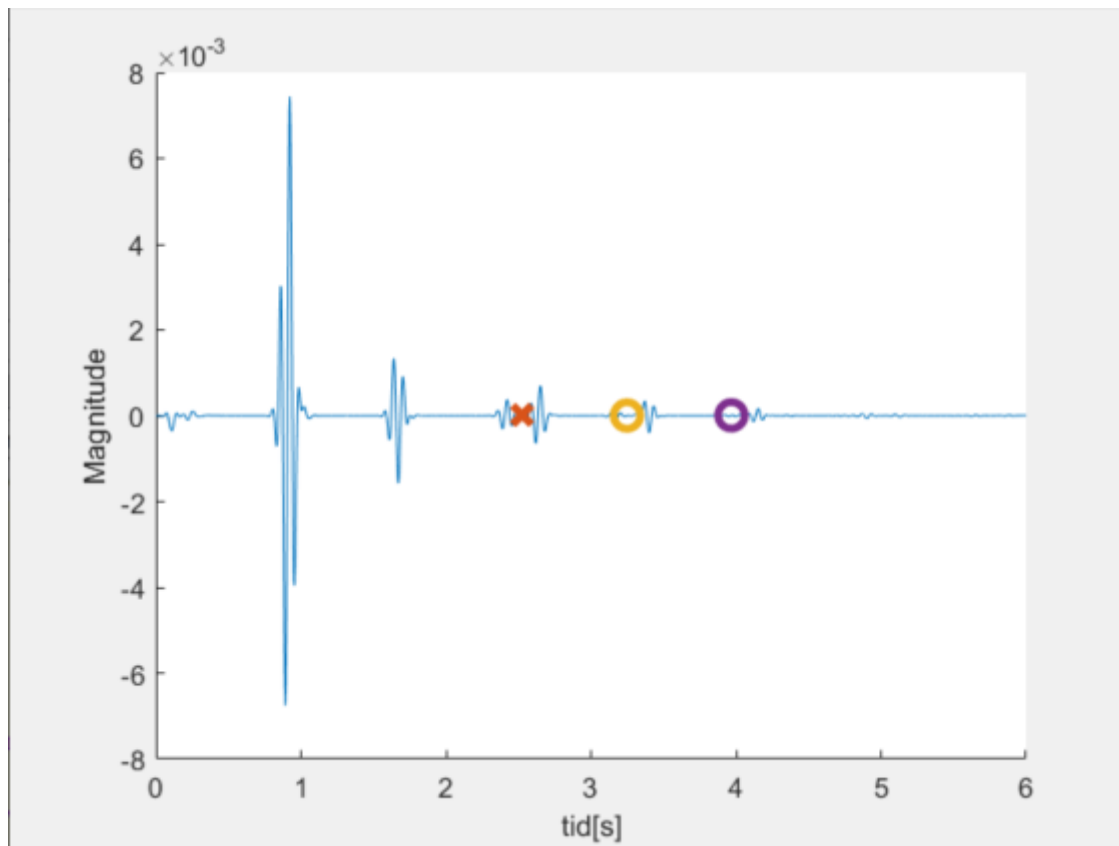
$t_w = 0.76s$  og  $c_w = 0.1426$





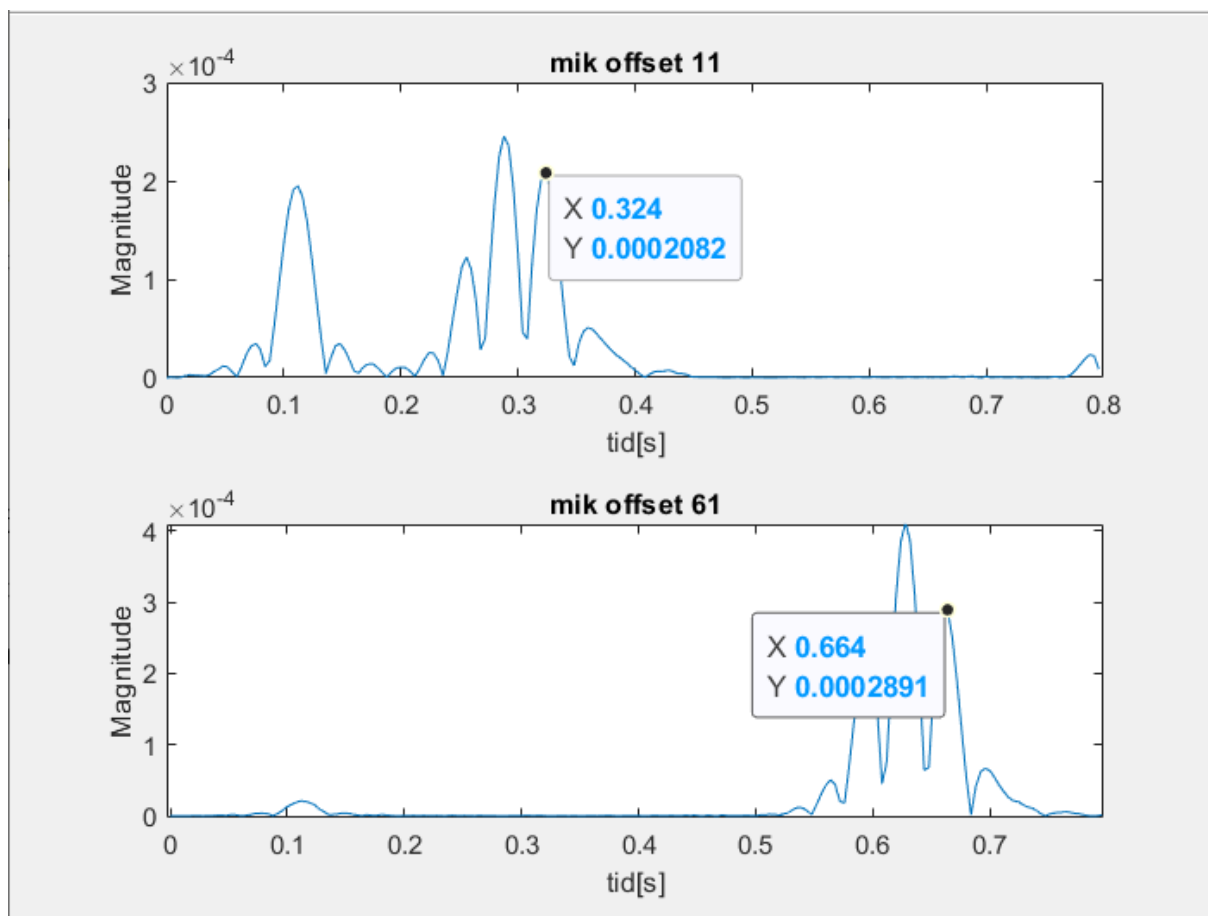
#### Oppgave 4b:

Plott av refleksjon av andre sedimentærlaget. Der hoved refleksjon er plottet med x.



### Oppgave 5:

Plotten av absoluttverdien av direkte ankomst i offset 11 og 61.



Tid forskjell er 0.34s

Avstand forskjell er 500m

Dermed gir det ca.  $500\text{m} / 0.34\text{s} = 1470\text{m/s}$ , som er lydhastighet i vann.

Her ser vi også jo større avstand mellom to mikrofoner, jo nøyaktigere svar blir.

### Oppgave 6a:

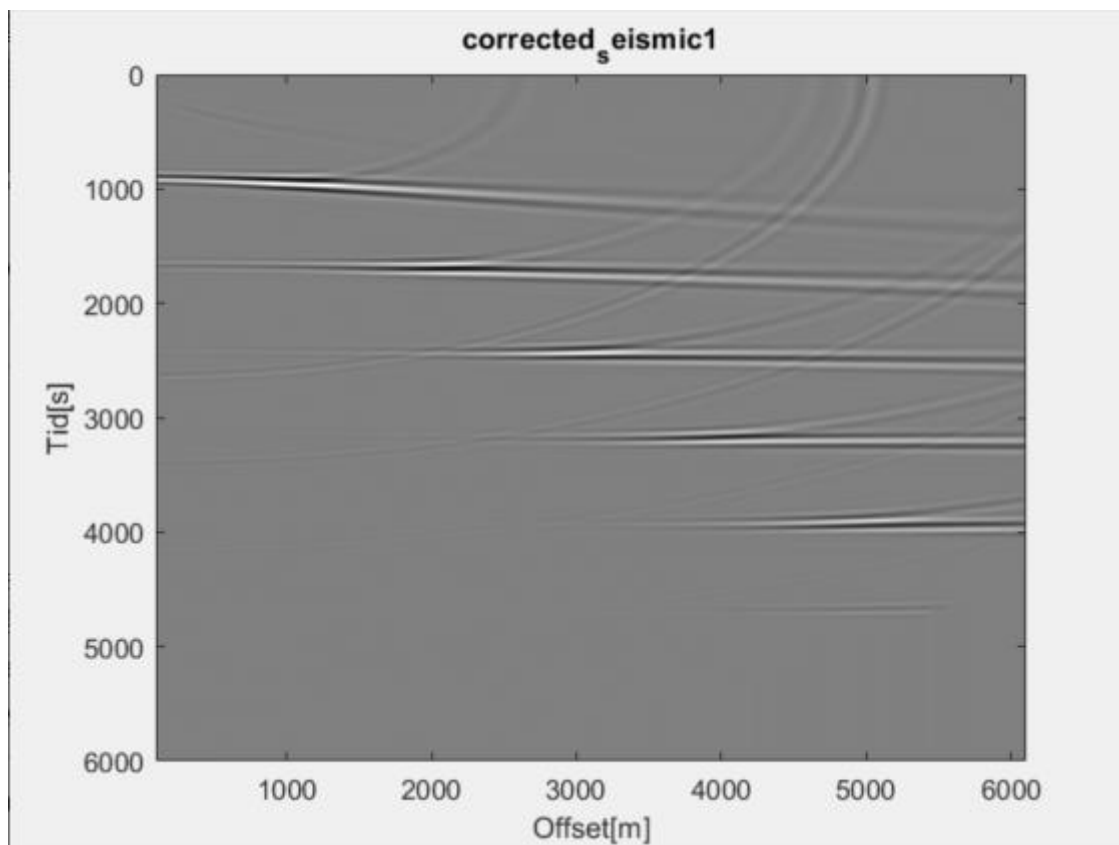
Med å prøve ut med 3 forskjellig offset:

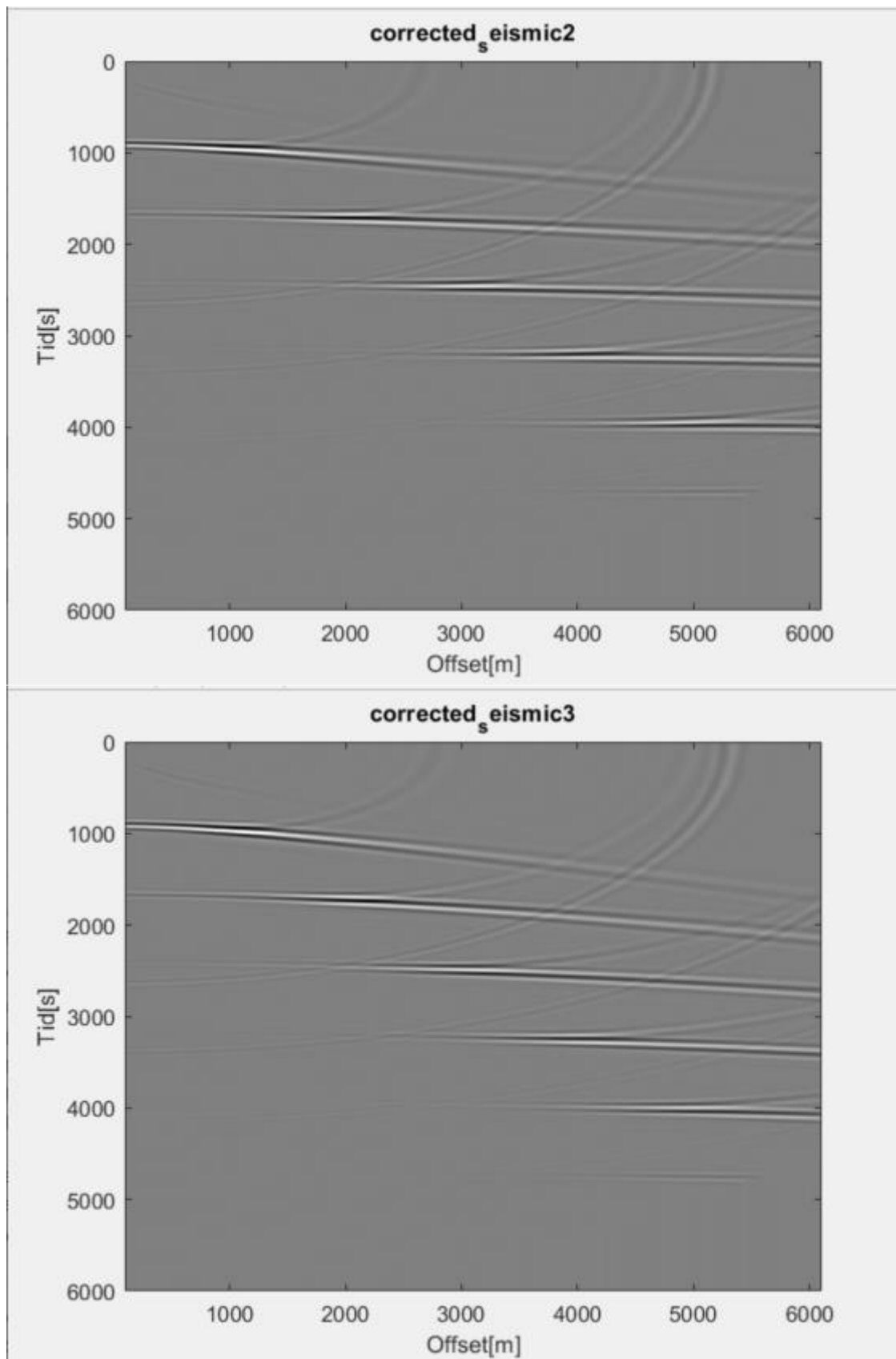
Med å prøve ut med forskjellige verdiene, vi ser allerede ved ca 20 m/s forskjell begynner, grafen til refleksjonen til første sedimentærlaget å bøye seg. Dermed er ca.  $\pm 20$  m/s.

```
corrected_seismic1 =  
nmocorrection2(t,0.0040,offset1,y1_all,1470);
```

```
corrected_seismic2 =  
nmocorrection2(t,0.0040,offset1,y1_all,1490);
```

```
corrected_seismic3 =  
nmocorrection2(t,0.0040,offset1,y1_all,1520);
```

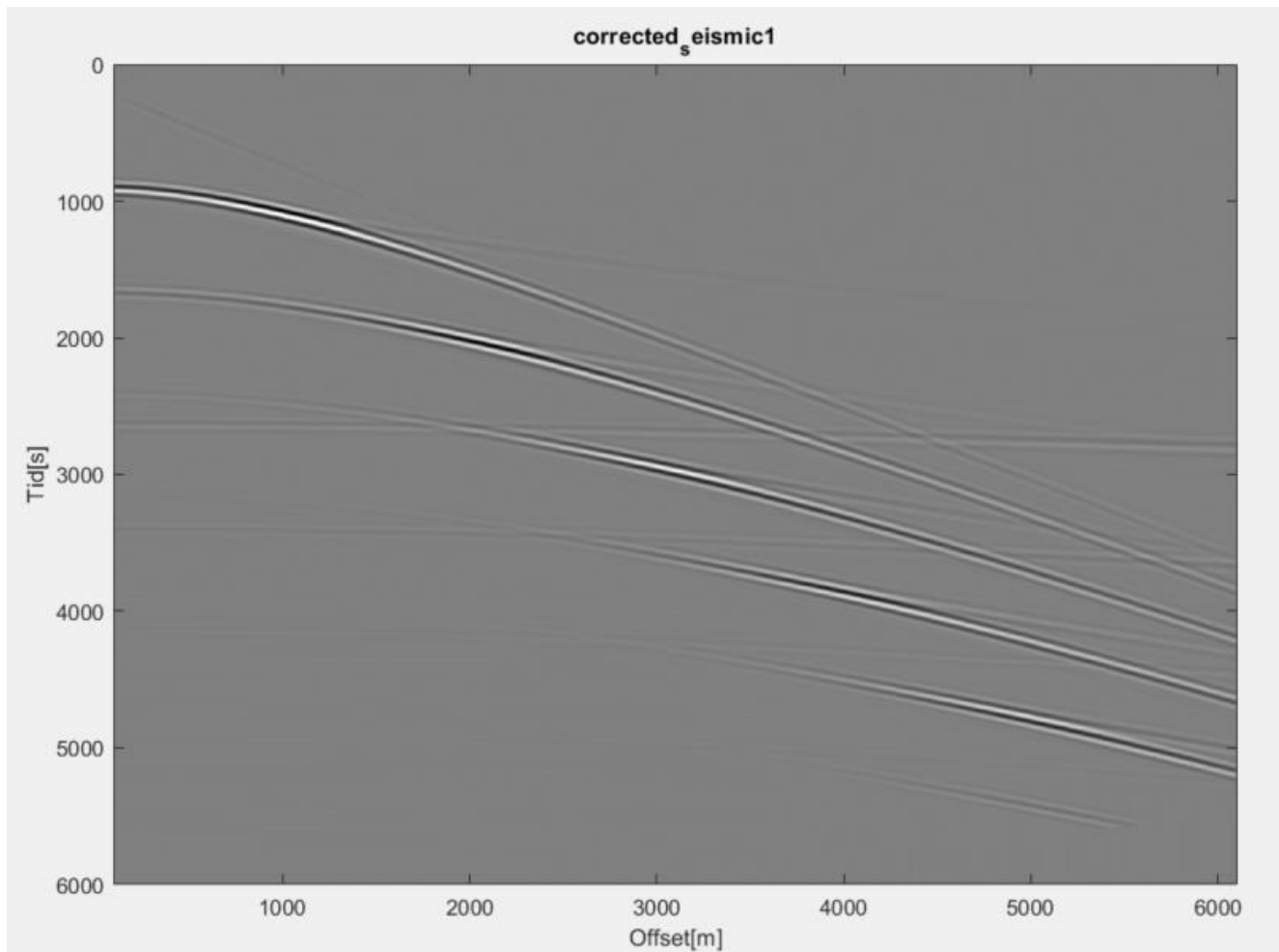


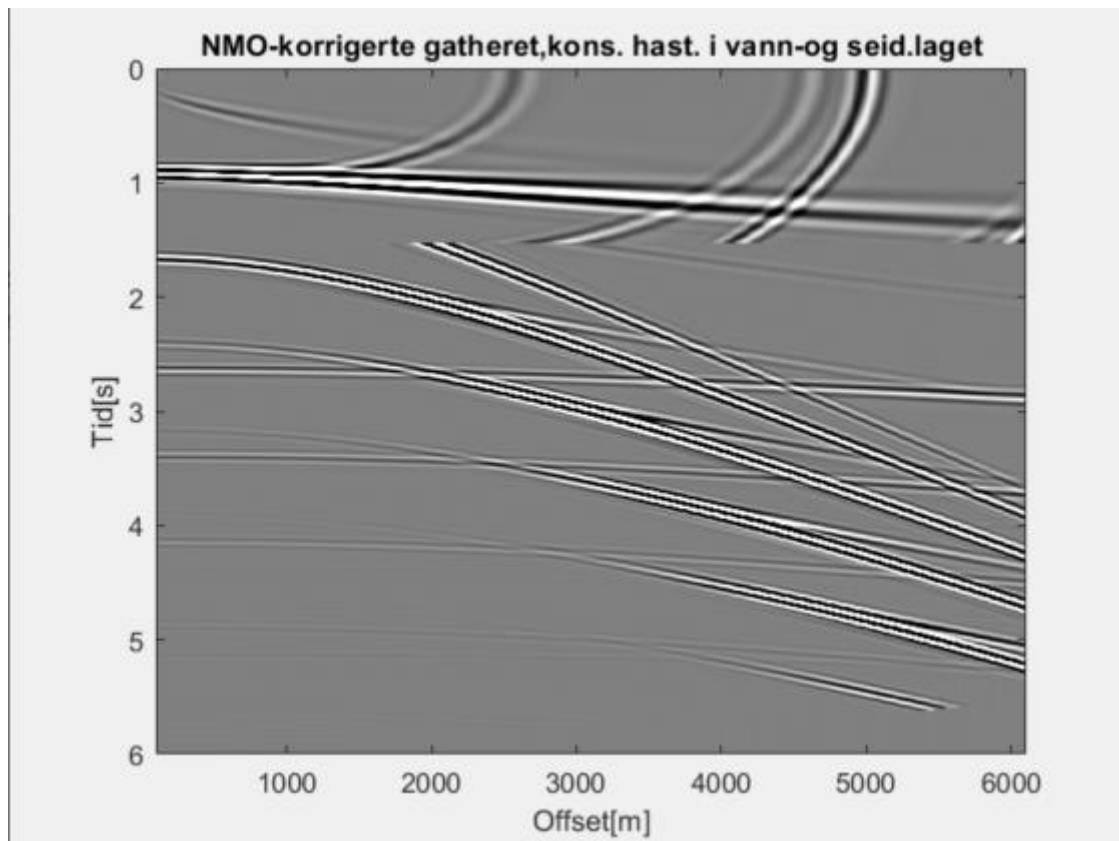


### Oppgave 6b:

Med å prøve ut ulike hastighet, observerer jeg at refleksjon fra andre sedimentærlaget er mest rett ved hastighet  $V_{\text{NOM}} = 2500\text{m/s}$ .

```
corrected_seismic1 =  
nmocorrection2(t,0.0040,offset1,y1_all,2500);
```



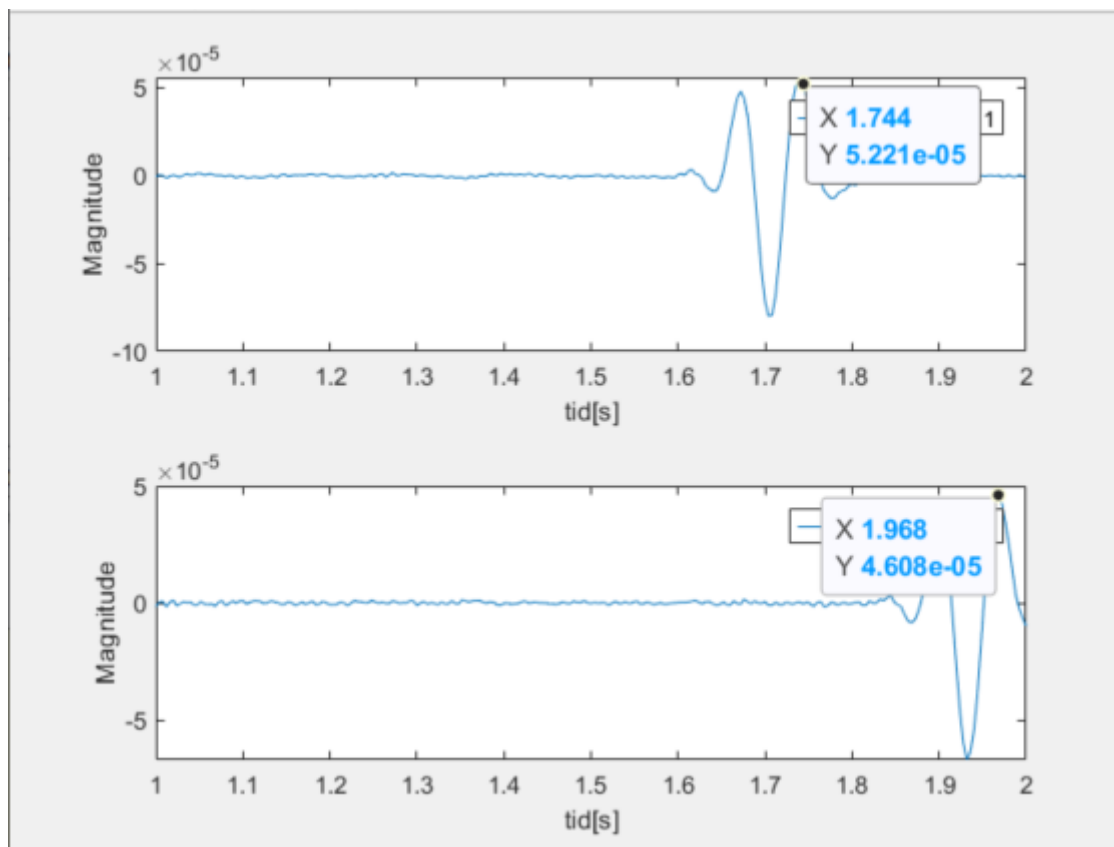


Dersom man glattere, så vil det som er mellom 1s til 2.6s blir redusert, og plottene blir finere. Men usikker på hvordan man gjør det, dermed ikke gjort det her.

### Oppgave 7a:

Med å se på tidsintervall 1s-2s og offset 5000m-5800m.

Finne refraksjon med ved offset 500 og 560.



Dette gir

$$V = 600\text{m}/0.228\text{s} = 2631.5\text{m/s}$$

Dette tilsvarer ca. 100m/s forskjellen med svaret jeg fant i oppgave 6b.

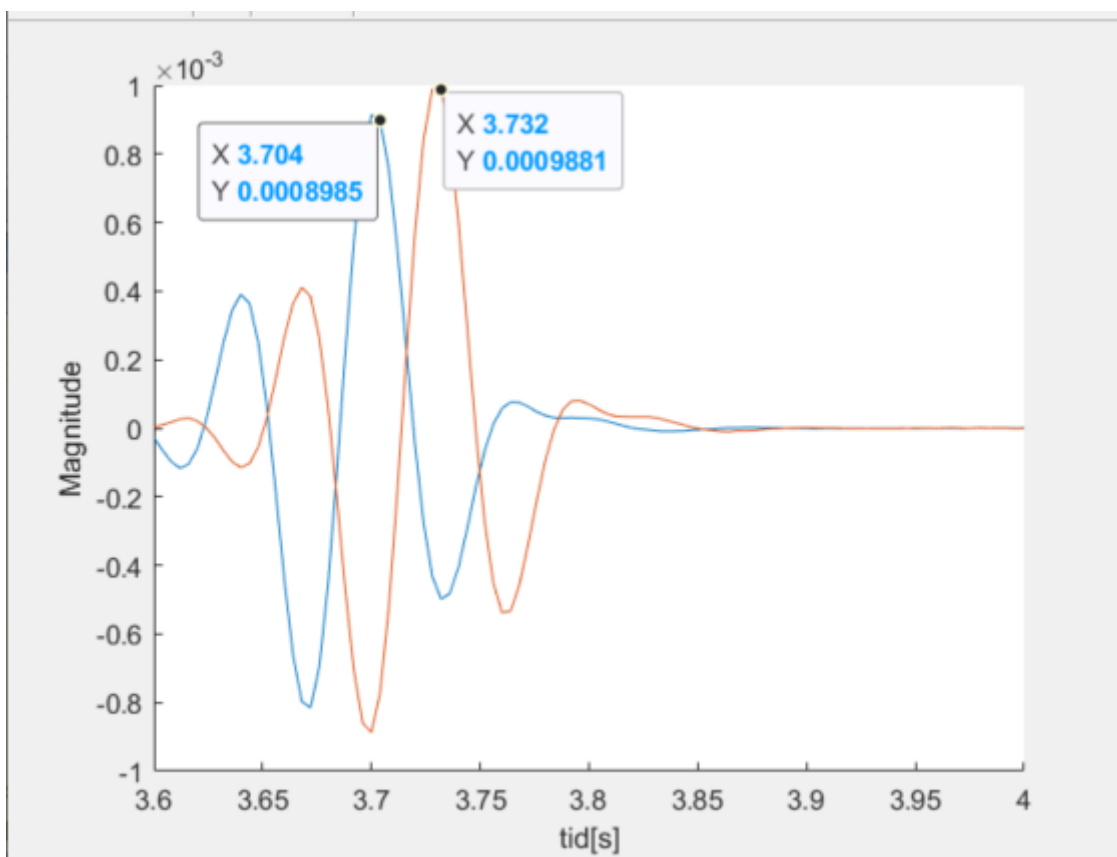
### Oppgave 7b:

Siden jeg ser ikke refraksjon til 2. sedimentærlaget. Dermed velger en over estimering av refraksjonen basert på tangensen til refleksjon til 2.sedimentærlaget.

Ved å se på tangenten til offset 590-601, tid 3,5s-3,8s. Finner jeg en hastighet ca. 4000m/s. Men jeg tror dette er litt dårlig estimering med stor unøyaktighet. Dermed prøvde jeg å plote signal ved 2 forskjellige offset.

Jeg valgte offset 591 og 601, som gir avstand 100m. Og signalet til refleksjonen brukte 0.028 sekund for å bevege seg over 100m. Dette gir ca. 3570m/s. Dette er likevel en grov estimering av lyd hastigheten til 2.sedimentærlaget. Den virkelige hastigheten er lavere enn det.

Plott av refleksjon signal i offset 591 og 601.



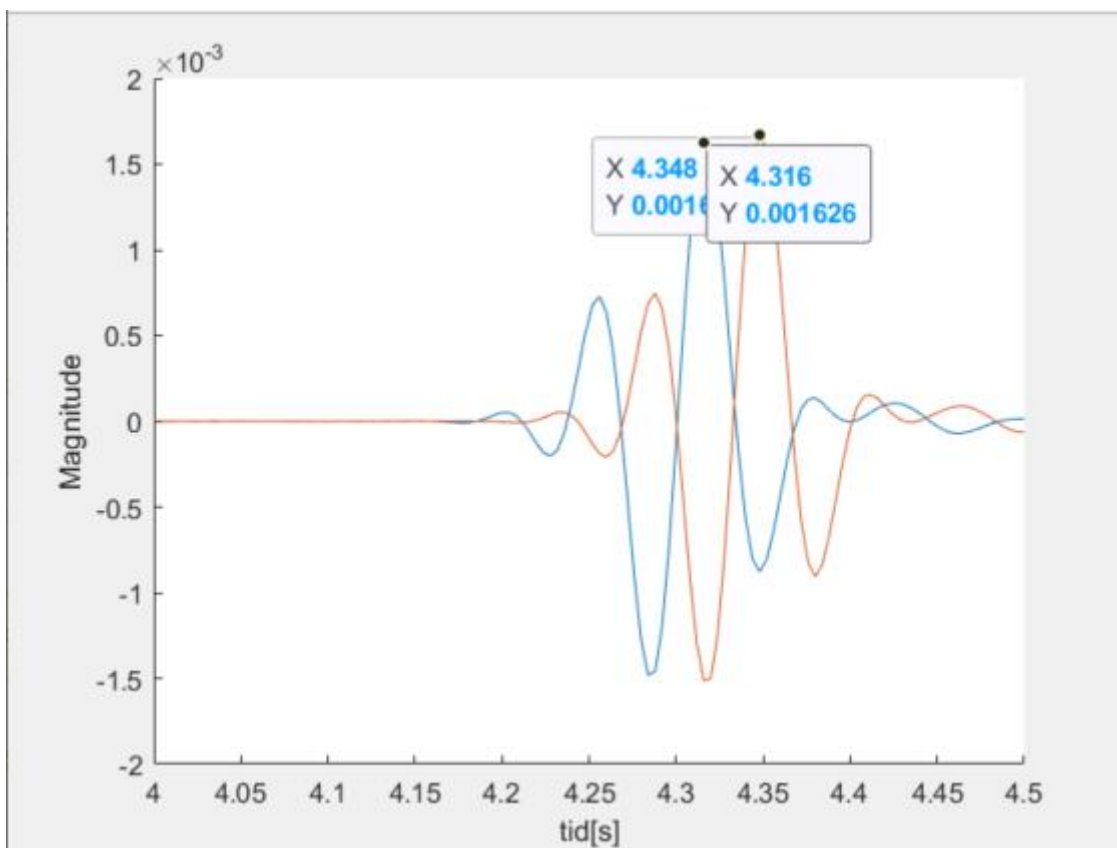


### Oppgave 7c:

I denne oppgaven så byter vi data seg, slik den har en offset over 8000m. I dette data, klarer jeg likevel ikke å se refraksjon til 2.sedimentærlaget. Vi bruker samme måte å estimere lydhastigheten, som vi gjorde i oppgave7b. Henter ut offset i 791 og 801.

Vi ser at avstanden mellom er 100m, og tidsforskjellen er 0.032s. Derfor gir lydhastighet i 2.sedimentærlaget ca. 3125m/s. Dette er en bedre estimering i forhold til 4000m/s og 3570m/s som vi har funnet i oppgave7b. Siden refraksjonen vil følge refleksjonen til et punkt og splitte ut. Dermed jo nærmere vi er til det splitt punktet, jo nøyaktigere er estimeringen. Dermed er 3125m/s en mer nøyaktigere estimering til lydhastighet i 2.sedimentærlaget.

Plott av refleksjon signal i offset 791 og 801.



**Oppgave 7d:**

Dybden til sedimentærlag 1:

Tiden det tar for hoved refleksjonen å komme tilbake er 0.76s.

Tiden det tar for en vei er dermed  $0.76s/2 = 0.38s$ .

Lydhastighet i vann: 1470m/s.

Dermed  $s=v*t = 558.6m$ .

Dybden til sedimentærlag 2:

Tiden det tar for hoved refleksjon til 2. sedimentærlaget til å komme tilbake er 2.528s.

Tiden det tar for en vei er dermed  $2.528s/2 = 1.264s$ .

$$t_{tot} = t_1 + t_2 = 1.264s$$

Fra dybden til sedimentærlag 1,  $t_1 = 0.38s$  og gir  $s_1 = 558.6m$

Dermed er  $t_2 = 0.884s$ , og lydhastighet i 2. sedimentærlaget er 2631m/s. Dermed gir det  $s_2 = 2325.8m$

$s_{tot} = s_1 + s_2 = 2884.4m$ , som er dybden til 2.sedimentærlaget.