Lab 3

Bharath Shekar
IIT Bombay
28/8/2018

Processing flow

- Throughout the class, you have to take data through a processing flow. So far, we have worked on the near-trace gather (we will continue to do so). Verify that you have taken the near-trace gather through the following steps:
    - Reading the data, geometry, sorting
    - Amplitude gain
    - Statics

Processing flow

- In order to maintain clarity, you could adopt a naming convention. For example:
  - Reading the data, geometry, sorting — ntgather.su
  - Amplitude gain — agc_ntgather.su
  - Statics — stat_agc_ntgather.su*

* You can cut short some names in order to avoid unnecessarily long names.. (for e.g. you can cut out “agc” in the next step)

Processing flow

- Reading the data, geometry, sorting
- Amplitude gain
- Statics
- Deconvolution: several flavours: next step… But before that, a few preliminaries.

Signature deconvolution

- The far-field signature is provided as a segy file “farfield.sgy” — download it from moodle
- The sampling interval for the wavelet is 1 ms
- Read in the data:

```
segyread tape=farfield.sgy verbose=1
| sushw key=dt a=1000 > farfield.su
```

Signature deconvolution

- Plot the spectrum of the farfield wavelet:

```
suxwigb < farfield.su  &

sufft < farfield.su | suamp
mode=amp | suxwigb &
```

Resampling

- The farfield wavelet is sampled at 1 ms and the seismic data is sampled at 4 ms. We should resample the farfield wavelet. But before we resample, we need to apply a low-pass filter (why?) How should I design the low-pass filter?

```
sufilter < farfield.su f=90,110 amps=1,0 >
filt_farfield.su

suxwigb < filt_farfield.su &

sufft < filt_farfield.su | suamp mode=amp |
suxwigb &
```

Resampling

- Compare the spectrum of the far-field wavelet before and after low-pass.
- Now that we have removed high frequencies from slightly-below-Nyquist onwards, we are ready to do resampling:

```
suresamp < filt_farfield.su dt=0.004
nt=150 > resamp_farfield.su
```

Compare the resampled wavelet with the original in time and frequency domain

Signature deconvolution

- By inspection, you might have observed that the far-field wavelet is not minimum phase. (To check rigorously, you need what is called as “Kolmogorov factorisation”)

- We now convert the wavelet into minimum phase:

```
suminphase < resamp_farfield.su
sign2=-1 > minphs_farfield.su
```

Compare the min. phased wavelet with the resampled wavelet in time and frequency domain..

Signature deconvolution

- By inspection, you might have observed that the far-field wavelet is not minimum phase. (To check rigorously, you need what is called as “Kolmogorov factorisation”)

- We now convert the wavelet into minimum phase:

```
suminphase < resamp_farfield.su
sign2=-1 > minphs_farfield.su
```

Compare the min. phased wavelet with the resampled wavelet in time and frequency domain..

Shaping the data

$$d(t) = s(t) * w(t) * e(t)$$

$$s(t) * f(t) = s_{\min}(t)$$

Minimum phase equivalent of s(t)

The same filter should be applied to the data to convert it to minimum phase. This shaping operation is also called signature deconvolution.

Signature deconvolution

- We are now ready to do signature deconvolution:

```
sushape < stat_agc_ntgather.su
wfile=resamp_farfield.su
dfile=minphs_farfield.su showshaper=1
nshape=1500 2>shaper.asc >
minphs_ntgather.su
```

Can you explain how the command sushape works? Hint: Previous slide.
Now compare the ntg in time and frequency domain before and after minimum phasing

Predictive error filtering

- We are now ready to do predictive error filtering.
- We need to work out the period of repetition of the multiples. Take auto-correlation of the data:

```
suacor < ntgather.su ntout=1001 |
suximage f1=-2.0 perc=99 &
```

Predictive error filtering

- Try to do predictive error filtering of the data (exercise).