Improving the Performance of the HHT using Ensemble Empirical Mode Decomposition

Daniel Bowman

January 23, 2013

Contents

1 Introduction

This vignette shows how the ensemble empirical mode decomposition method (EEMD) can improve the empirical mode decomposition (EMD) method and the resulting Hilbert spectrogram. The EEMD is a noise assisted version of the EMD method. It was developed to reduce aliasing and mode mixing that can occur during EMD. For more information, see Wu and Huang (2009) "Ensemble Empirical Mode Decomposition: A Noise Assisted Data Analysis Method."

This vignette examines three signals. The first signal is the same as Section 3 in the vignette titled "Investigating some Interesting Signals with Hilbert and Fourier Spectrograms." In this example, I demonstrate how the EEMD can improve the quality of intrinsic mode functions (IMFs), particularly at the beginning and end of the signal.

The second signal has two sinusoids with Gaussian white noise added. This example shows how the EEMD method can improve the IMF set and the resulting Hilbert spectrogram.

The third signal is a transient event recorded at Deception Island volcano, Antarctica. I show how the EEMD can extract meaningful information from a signal previously thought to be too short for spectral analysis. I also demonstrate the higher time frequency resolution of the Hilbert spectrogram as opposed to the Fourier spectrogram.

WARNING: The EEMD code in this vignette can take a very long time to run (minutes to hours).

2 Eliminating Spline Fitting Problems

The vignette titled "Investigating some Interesting Signals with Hilbert and Fourier Spectrograms" examines the EMD of the following signal:

$$x(t) = \sin(2\pi t + 0.5\sin(\pi t))e^{\frac{-(t-50)^2}{200}} + \sin(\frac{\pi}{10}t) \tag{1}$$

While the EMD successfully removes the nonlinear signal from the lower frequency sinusoid, the first and second IMF contain unusual oscillations in the first 20 and last 20 seconds of the signal (Fig. 1). These oscillations are not implied in the signal equation and likely represent problems with the spline fitting method in the EMD.

If low amplitude Gaussian white noise is added to the signal, the spline fitting issue is solved (Fig. 2). However, this creates a lot more IMFs, and the transient nonlinear signal $a_1 \sin(\omega_1 t + a_2 \sin(\omega_2 t))e^{\frac{-(t-50)^2}{200}}$ is now distributed across multiple IMFs rather than residing on a single IMF. This problem is called *mode mixing* and reflects the fact that random noise populates all time scales equally. Mode mixing can cause aliasing in the Hilbert transform because it disturbs local symmetry in the affected IMFs.

Each trial of the EEMD method adds a different white noise set to the signal and performs EMD. If this is done enough times, the averaged IMF

set from all the trials approaches the "true" signal . After EEMD, the spline fitting has been improved and mode mixing has been reduced (Fig. 3). The higher frequency signal is still present on several IMFs, but the signal less intermittent than it was in the noisy EMD trial (Fig. 2).

2.1 Code

EMD with no added noise

```
> library(hht)
> dt=0.01
> tt=seq_len(10000)*dt
> sig=sin(2*pi*tt+0.5*sin(pi*tt))*exp(-(tt-50)^2/100)+sin(pi*tt/10)
> emd_config=list()
> emd_config$max_sift=200
> emd_config$max_imf=100
> emd_config$tol=5
> emd_config$stop_rule="type5"
> emd_config$boundary="wave"
> emd_config$sm="none"
> emd_config$spar=NA
> emd_config$weight=20
> emd_result=sig2imf(sig, dt, emd_config)
> time_span=c(0, -1)
> imf_list=c(1:2)
> os=TRUE
> res=TRUE
> plot_imfs(emd_result, time_span, imf_list, os, res)
```

EMD with added noise

```
> set.seed(628)
> sig2=sin(2*pi*tt+0.5*sin(pi*tt))*exp(-(tt-50)^2/100)+
+     sin(pi*tt/10)+rnorm(length(sig), 0, 0.01)
> emd_result=sig2imf(sig2, dt, emd_config)
> time_span=c(0, -1)
> imf_list=c(1:emd_result$nimf)
> os=TRUE
> res=FALSE
> plot_imfs(emd_result, time_span, imf_list, os, res)
```

Perform EEMD

```
> trials=100
> nimf=11
> noise_amp=0.01
> trials_dir="example_1"
> eemd(sig2, dt, trials, nimf, noise_amp, emd_config, trials_dir)
> eemd_result=eemd_compile(trials_dir, trials, nimf)
```

Calculate and Plot Averaged IMFs

```
> emd_config=list()
> emd_config$max_sift=200
> emd_config$max_imf=100
> emd_config$tol=5
> emd_config$stop_rule="type5"
> emd_config$boundary="wave"
> emd_config$sm="none"
> emd_config$spar=NA
> emd_config$weight=20
> resift_rule="max_var"
> resift_result=eemd_resift(eemd_result, emd_config, resift_rule)
> time_span=c(0, 100)
> imf_list=c(3:7)
> os=TRUE
> res=FALSE
> plot_imfs(resift_result, time_span, imf_list, os, res)
```
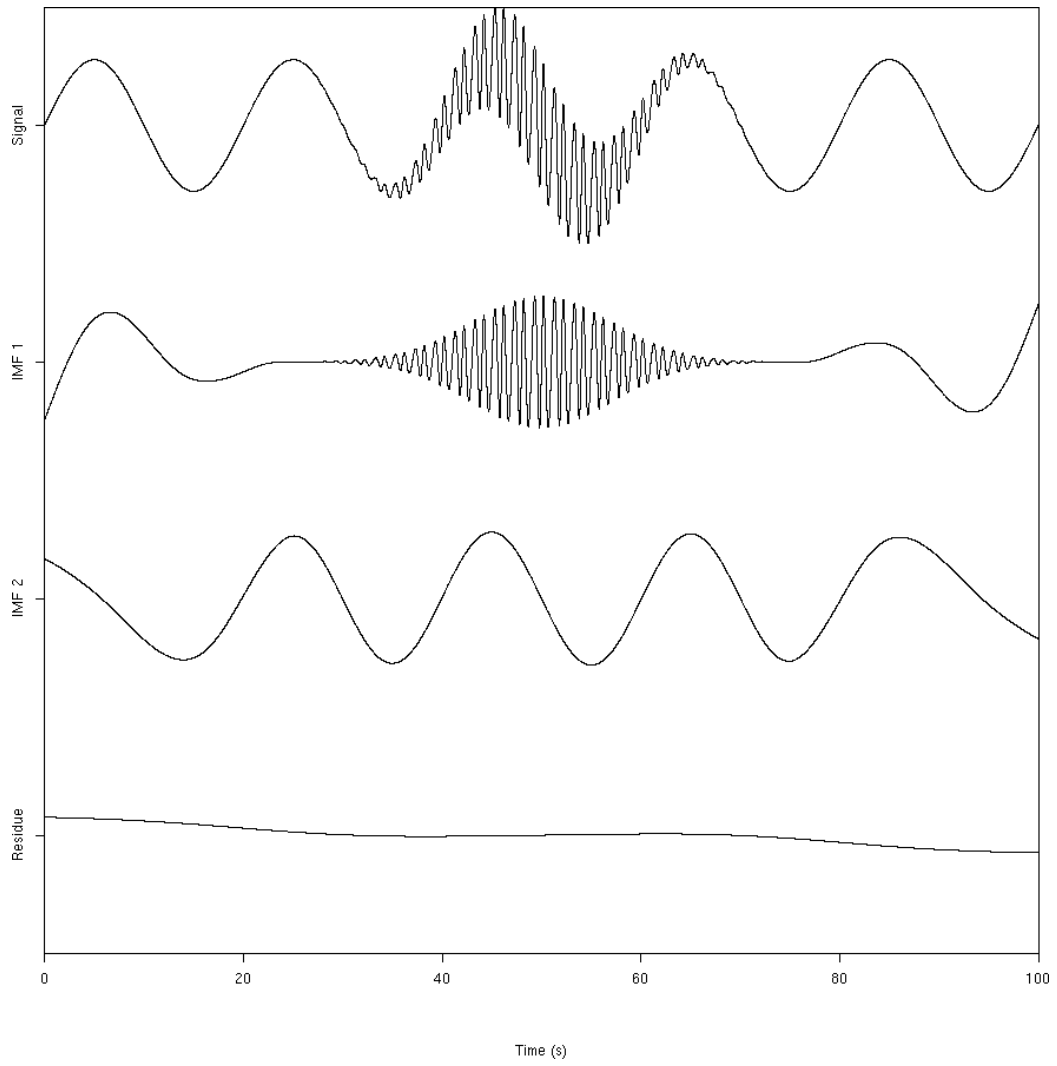
2.2 Figures

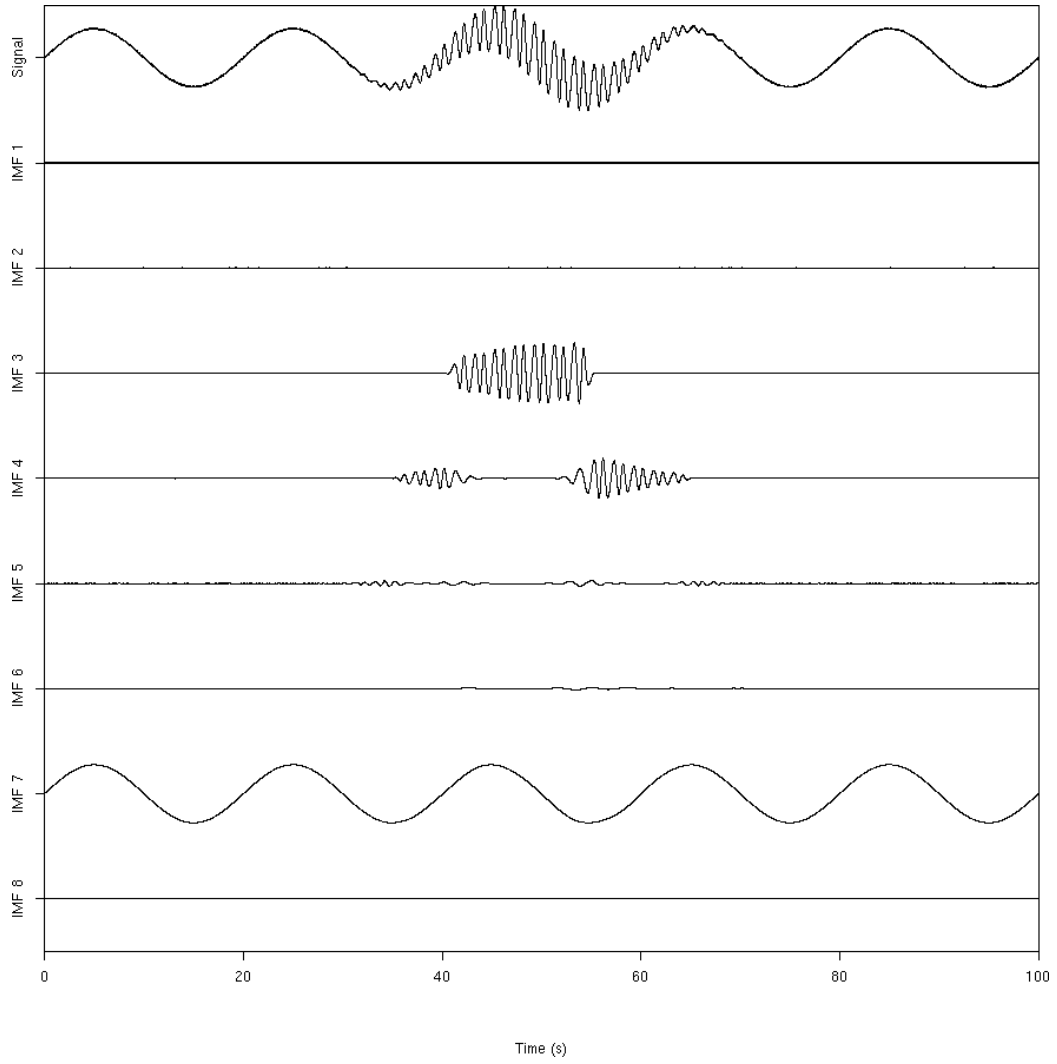Figure 1: EMD of signal in Equation 1. A total of 2 IMFs were returned.

Figure 2: EMD of signal in Equation 1 with random Gaussian noise added.
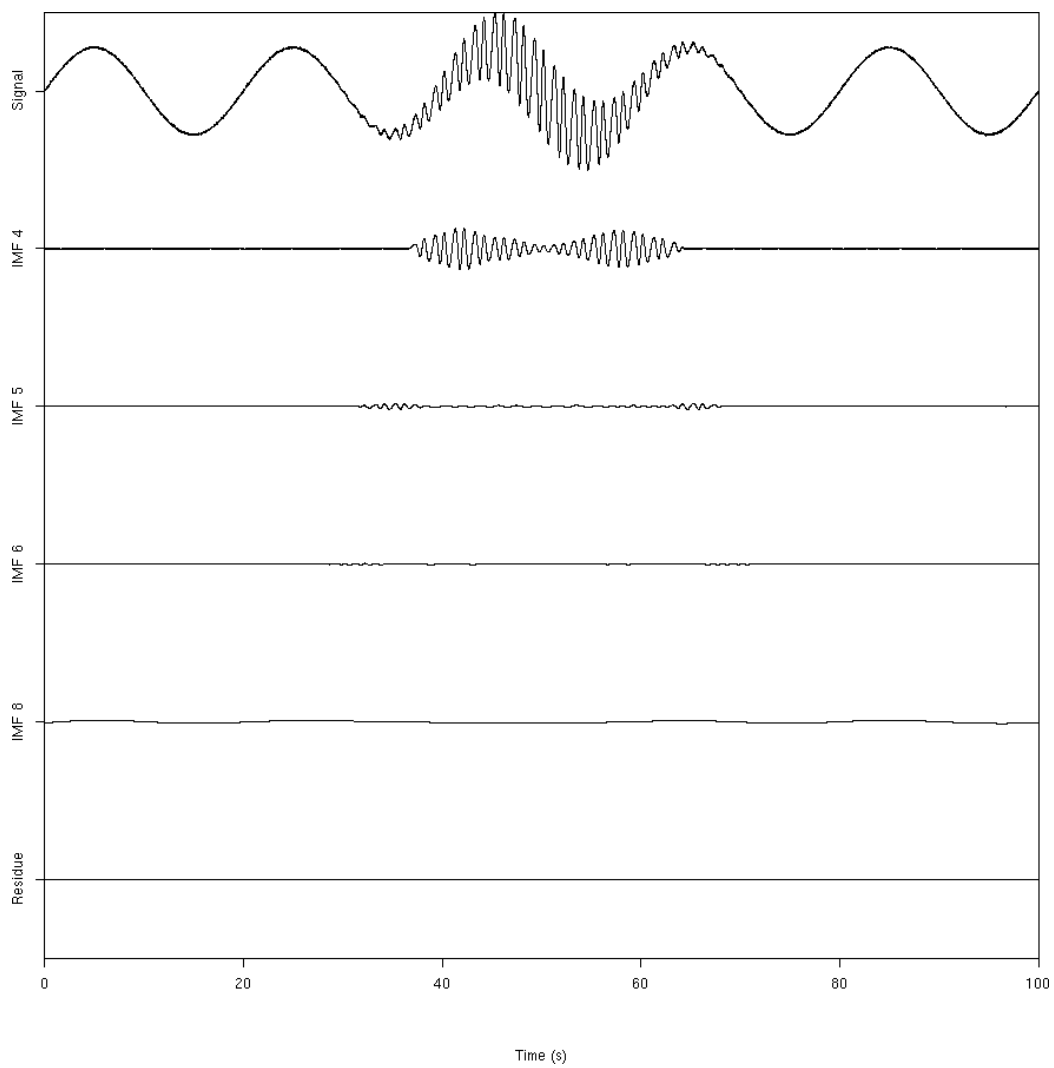
Figure 3: Major IMFs from EEMD of signal in Equation 1

.

3   Reducing Mode Mixing

Even a well behaved stationary time series will have problems if there is random noise in addition to the signal. Fortunately, the EEMD is able to counteract the effect of a noisy signal by averaging a large number of trials together. As the number of trials grows, the signal tends to emerge from the averaged IMF set.

This example's time series has the form

$$x(t) = 2\sin(2\pi t) + 0.5\sin(12\pi t) + \mathcal{N}(0, 0.4) \qquad (2)$$

where

$$\mathcal{N}(0, 0.4)$$

produces random Gaussian noise with a mean of 0 and a standard deviation of 0.4. In theory, the EMD should produce one IMF showing the $2\sin(2\pi t)$ component and one IMF showing the $0.5\sin(12\pi t)$ component. However, the presence of noise creates many more IMFs and also causes severe mode mixing. The most obvious mode mixing occurs where portions of the high frequency sinusoid switch back and forth between IMF 2 and IMF 3. IMF 4 and IMF 5 both contain portions of the lower frequency, creating severe distortion in that signal band as well (Fig. 4).

The Hilbert spectrogram of this time series should show strong continuous frequency bands at 0.5 Hz and 6 Hz. However, the mode mixing produces a chaotic spectrogram with an intermittent low frequency signal around 0.5 Hz and a scattered high frequency signal between 4 and 8 Hz (Fig. 5).

After a 100 trial EEMD run, the averaged IMF set looks better than the original EMD (Fig. 6). While IMF 2 still has evidence of mode mixing, IMF 3 is more continuous than it was before. IMF 5 is much improved compared to the original IMF set.

The EEMD Hilbert spectrogram is a marked improvement over the original Hilbert spectrogram (Fig. 7). There are two continuous bands of energy that correspond with the sinusoids in Equation 2. While the 6 Hz band is still noisy, it is less scattered than in the original spectrogram in Figure 5.

3.1   Code

EMD of noisy signal

```
> library(hht)
> set.seed(628)
> dt=0.01
> tt=seq_len(2000)*dt
> sig=sin(2*pi*tt) + 2*sin(12*pi*tt)+rnorm(length(tt), 0, 0.4)
> emd_config=list()
> emd_config$max_sift=200
> emd_config$max_imf=100
> emd_config$tol=5
> emd_config$stop_rule="type5"
> emd_config$boundary="wave"
> emd_config$sm="none"
> emd_config$spar=NA
> emd_config$weight=20
> emd_result=sig2imf(sig, dt, emd_config)
> time_span=c(0, 20)
> imf_list=1:5
> os=TRUE
> res=FALSE
> plot_imfs(emd_result, time_span, imf_list, os, res)
```

Hilbert spectrogram of noisy signal

```
> emd_result=hhtransform(emd_result)
> max_freq=10
> freq_step=0.1
> hspec=hh_render(emd_result, max_freq, freq_step)
> time_span=c(0, 20)
> freq_span=c(0, 10)
> amp_span=c(0.75, 3)
> hhspec_image(hspec, time_span, freq_span, amp_span)
```

Run EEMD

```
> trials=100
> nimf=9
> noise_amp=0.4
> trials_dir="example_2"
> eemd(sig, dt, trials, nimf, noise_amp, emd_config, trials_dir)
> eemd_result=eemd_compile(trials_dir, trials, nimf)
```

Generate averaged IMF set

```
> trials=100
> nimf=9
> trials_dir="example_2"
> eemd_result=eemd_compile(trials_dir, trials, nimf)
> emd_config=list()
> emd_config$max_sift=200
> emd_config$max_imf=100
> emd_config$tol=5
> emd_config$stop_rule="type5"
> emd_config$boundary="wave"
> emd_config$sm="none"
> emd_config$spar=NA
> emd_config$weight=20
> resift_rule="max_var"
> resift_result=eemd_resift(eemd_result, emd_config, resift_rule)
> time_span=c(0, 20)
> imf_list=1:5
> os=TRUE
> res=FALSE
> plot_imfs(resift_result, time_span, imf_list, os, res)
```

Render and plot EEMD Hilbert spectrogram

```
> trials_dir="example_2"
> trials=100
> nimf=9
> eemd_result=eemd_compile(trials_dir, trials, nimf)
> max_freq=10
> freq_step=0.1
> hspec=hh_render(eemd_result, max_freq, freq_step)
> time_span=c(0, 20)
> freq_span=c(0, 10)
> amp_span=c(0.75, 3)
> cluster_span=c(8, 100)
> hhspec_image(hspec, time_span, freq_span, amp_span, cluster_span)
```
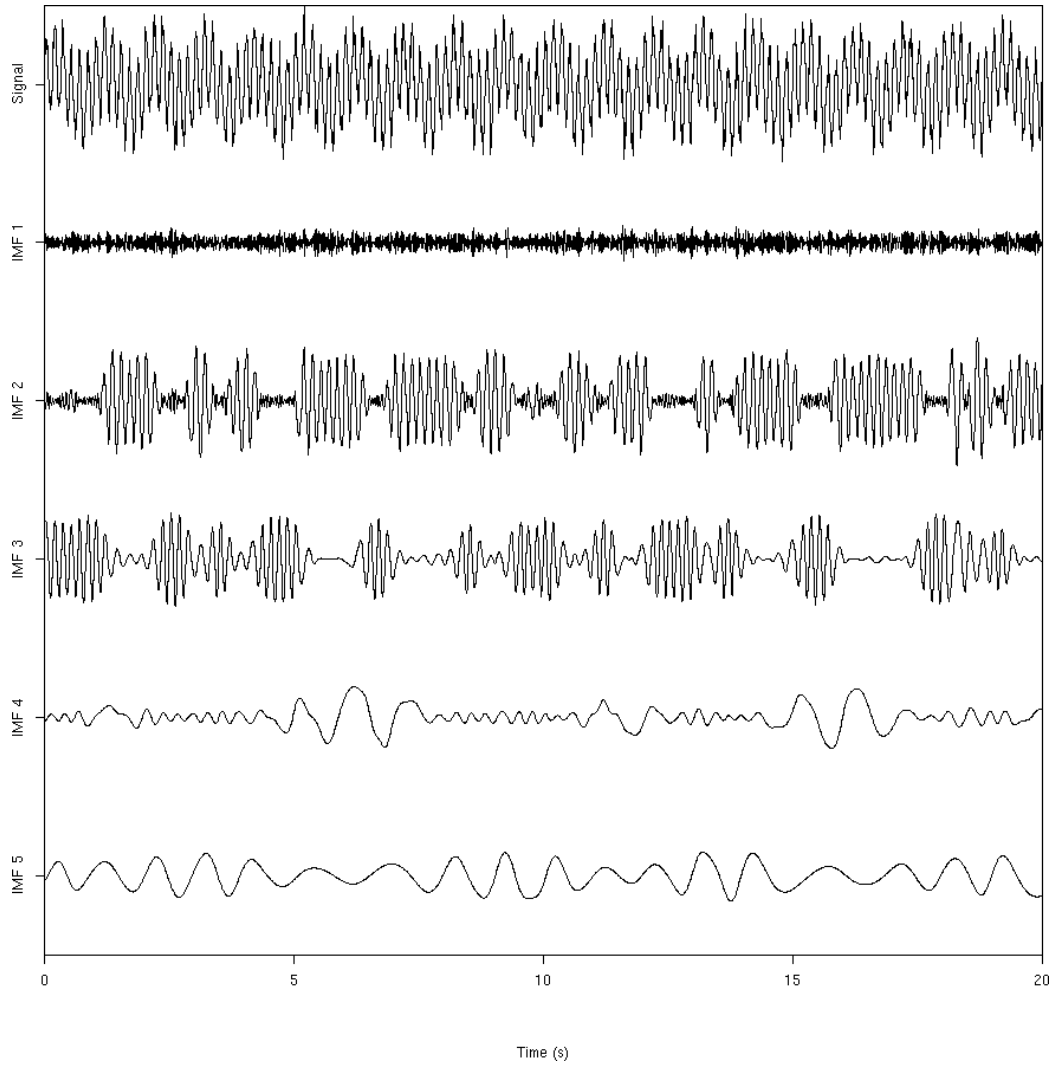
3.2   Figures

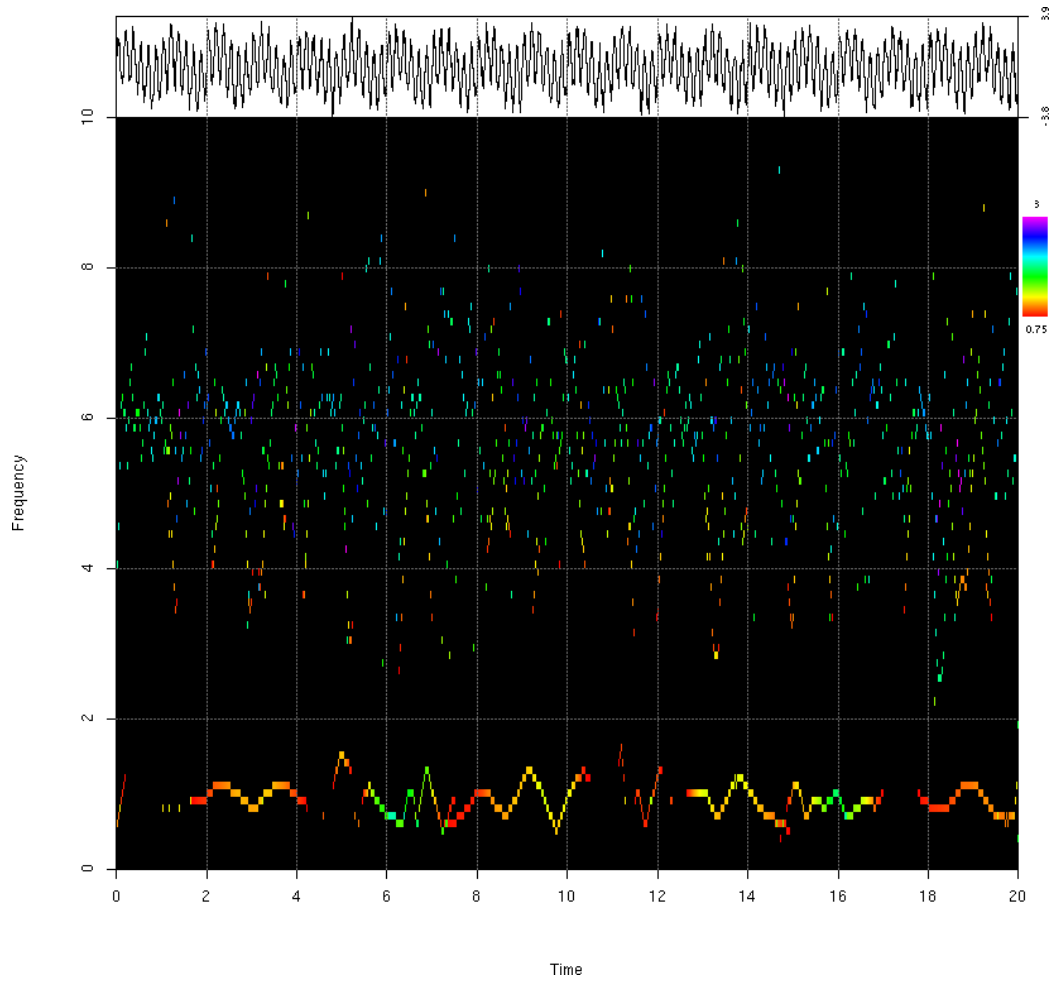Figure 4: EMD of signal in Equation 2.

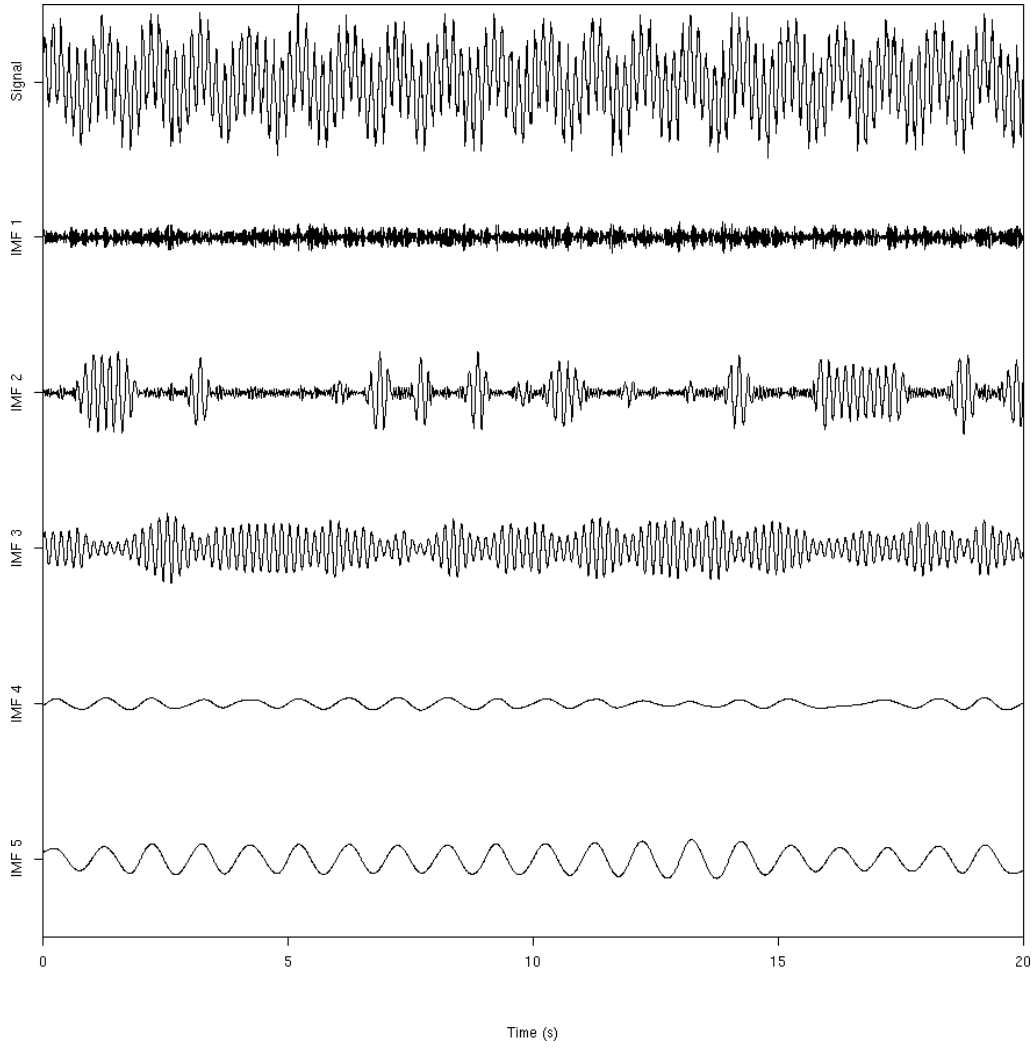Figure 5: Hilbert spectrogram of signal in Equation 2.

13

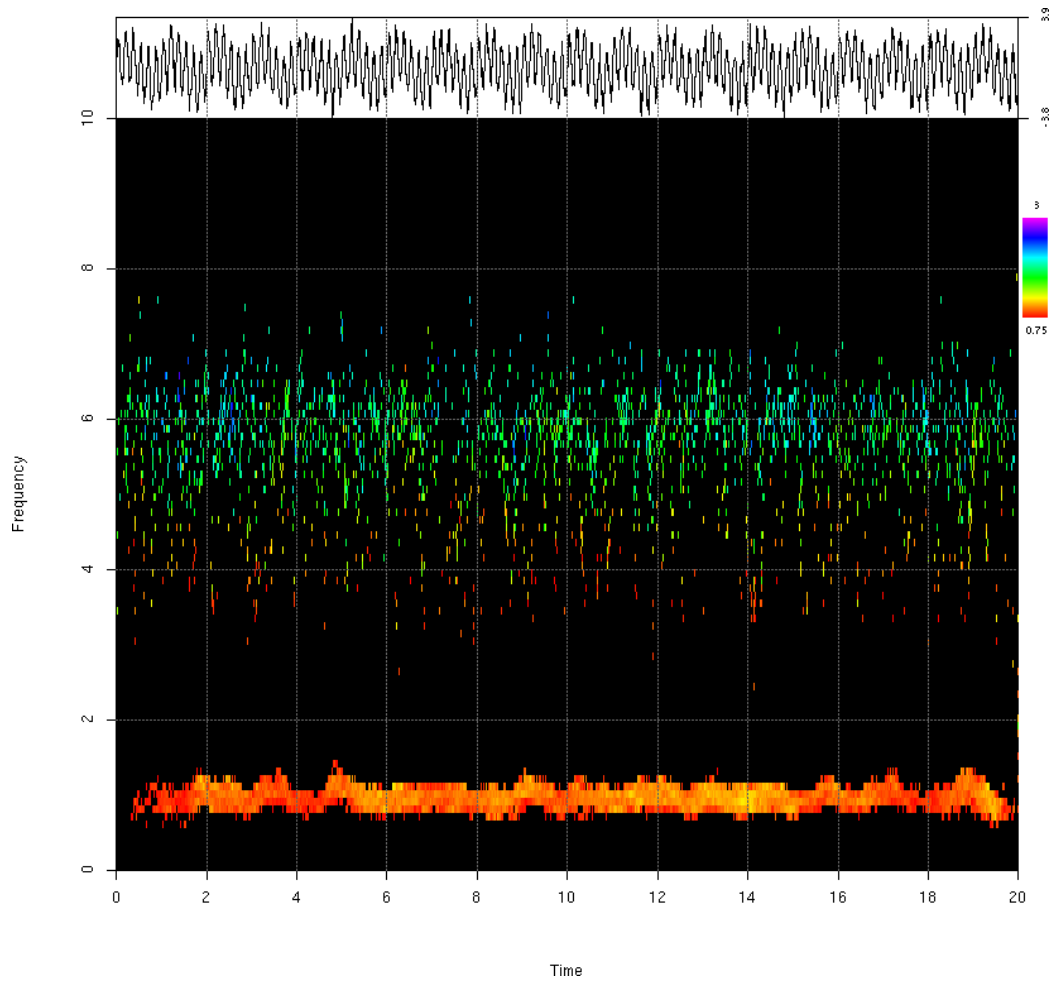Figure 6: Averaged IMF set generated from EEMD of signal in Equation 2.

Figure 7: Hilbert spectrogram generated from EEMD of signal in Equation 2.

4   Transient Seismic Event Analyzed with EEMD

A network of ocean bottom seismometers was deployed in Port Foster (the flooded caldera of Deception Island volcano) in 2005. These instruments recorded numerous transient seismic events. One of these events (and its 10 trial EEMD) has been included as an example data set in this R package. This vignette regenerates this EEMD, but with 100 trials.

The EEMD method recovers three main IMFs: a high amplitude, high frequency component, a high amplitude, low frequency component and a low frequency, low amplitude component (Fig. 8). While the presence of a high frequency component is clear from the original signal, the spindle shape of this component is not obvious until it is decomposed into an IMF. Although this could have been discovered by using a bandpass filter, the signal would likely have been distorted because of the transient nature of the waveform and the fact that this component has frequency modulation. Therefore, the EEMD has already recovered potentially valuable information on the different components of this signal.

The EEMD Hilbert spectrogram shows that each of the three components varies in frequency throughout the signal (Fig. 9). The high frequency component has energy between 15 and 20 Hz, dropping to around 13 Hz in about a quarter of a second. The low frequency, high amplitude component starts at around 6 Hz, glides to below 5 Hz over a half second, then rises in frequency again for another second or so before tapering off. The low frequency, low amplitude component decreases from around 5 Hz to around 2 Hz over less than a second.

The Fourier spectrogram cannot resolve such fine details in this signal (Fig. 10). It is difficult to determine if there is frequency gliding in the Fourier spectrogram due to the coarseness of the frequency resolution. This problem could be alleviated by shortening the time window, but that would mask the contribution of lower frequency elements of the signal. Also, the Fourier spectrogram shows energy arriving slightly before 6.5 seconds, whereas the seismogram shows the main signal starting just after 6.5 seconds. However, the Hilbert spectrogram shows spectral energy arriving just after 6.5 seconds, thus providing better agreement with the seismogram.

4.1   Code

Run EEMD

```
> library(hht)
> data(port_foster_event)
> set.seed(628)
> emd_config=list()
> emd_config$max_sift=200
> emd_config$max_imf=100
> emd_config$tol=5
> emd_config$stop_rule="type5"
> emd_config$boundary="wave"
> emd_config$sm="none"
> emd_config$spar=NA
> emd_config$weight=20
> trials=100
> nimf=8
> noise_amp=6.4e-07
> trials_dir="example_3"
> eemd(sig, dt, trials, nimf, noise_amp, emd_config, trials_dir)
> eemd_result=eemd_compile(trials_dir, trials, nimf)
```

Create averaged IMF set

```
> emd_config=list()
> emd_config$max_sift=200
> emd_config$max_imf=100
> emd_config$tol=5
> emd_config$stop_rule="type5"
> emd_config$boundary="wave"
> emd_config$sm="none"
> emd_config$spar=NA
> emd_config$weight=20
> resift_rule="max_var"
> resift_result=eemd_resift(eemd_result, emd_config, resift_rule)
> time_span=c(5, 10)
> imf_list=1:3
> os=TRUE
> res=FALSE
> fit_line=TRUE
> plot_imfs(resift_result, time_span, imf_list, os, res, fit_line)
```

Render and display Hilbert spectrogram

```
> trials_dir="example_3"
> trials=100
> nimf=11
> eemd_result=eemd_compile(trials_dir, trials, nimf)
> max_freq=30
> freq_step=0.05
> hspec=hh_render(eemd_result, max_freq, freq_step)
> time_span=c(5, 9)
> freq_span=c(0, 25)
> amp_span=c(1e-06, 3e-05)
> cluster_span=c(4, 100)
> hhspec_image(hspec, time_span, freq_span, amp_span, cluster_span)
```

Render and display Fourier spectrogram

```
> data(port_foster_event)
> ft=list()
> ft$xt=sig
> ft$dt=dt
> ft$nfft=4096
> ft$ns=30
> ft$nov=29
> time_span=c(5, 10)
> freq_span=c(0, 25)
> amp_span=c(0.00001, 0.001)
> ftspec_image(ft, time_span, freq_span, amp_span)
```
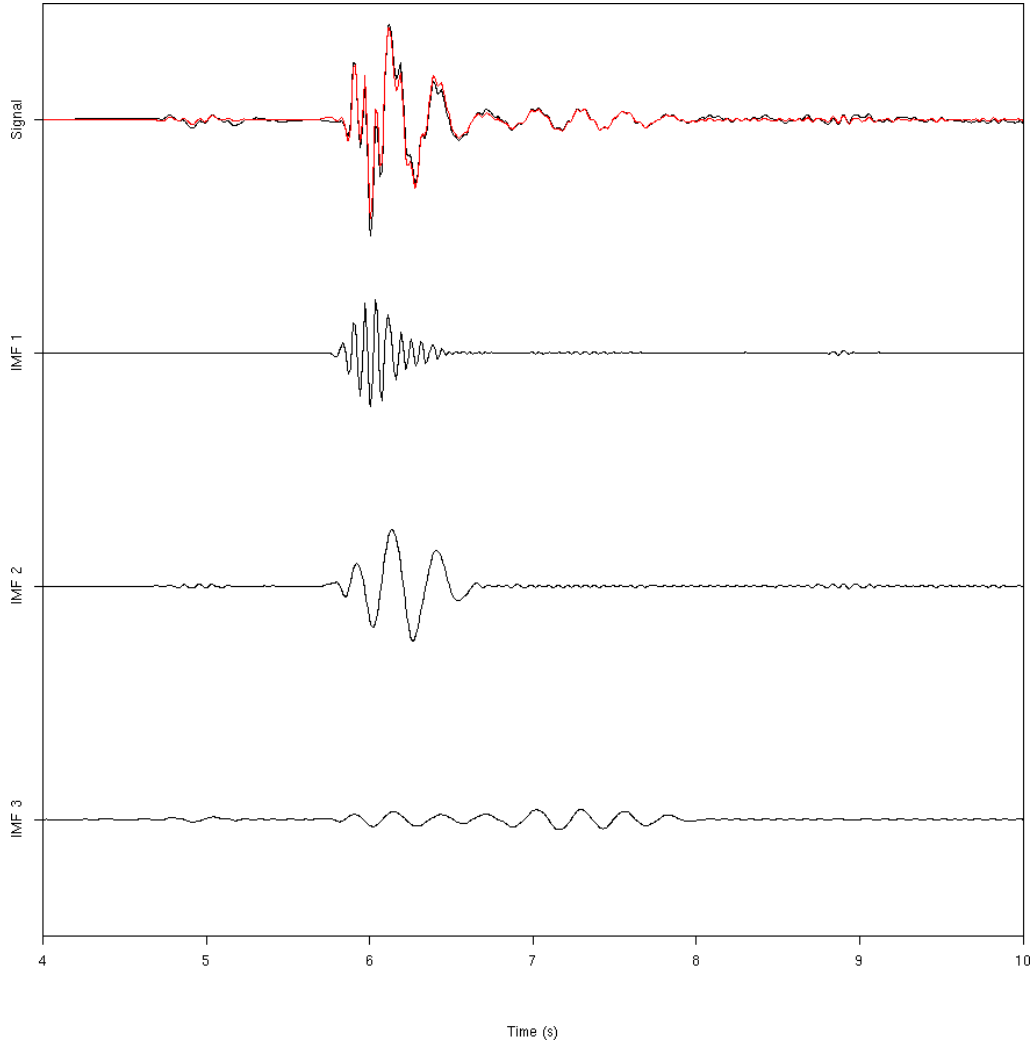
4.2   Figures

Figure 8: Averaged IMFs from EEMD of transient seismic signal. The red line shows sum of displayed IMFs, black line shows original signal.
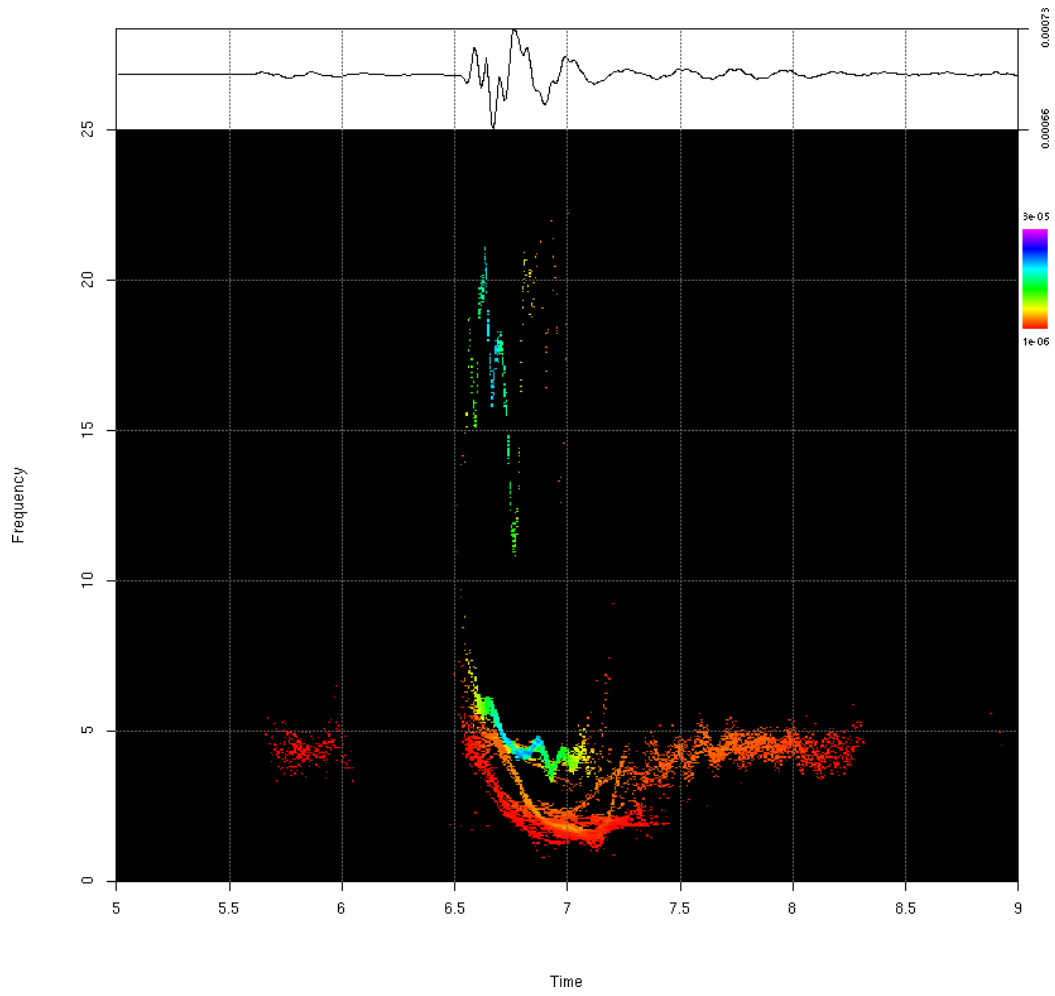
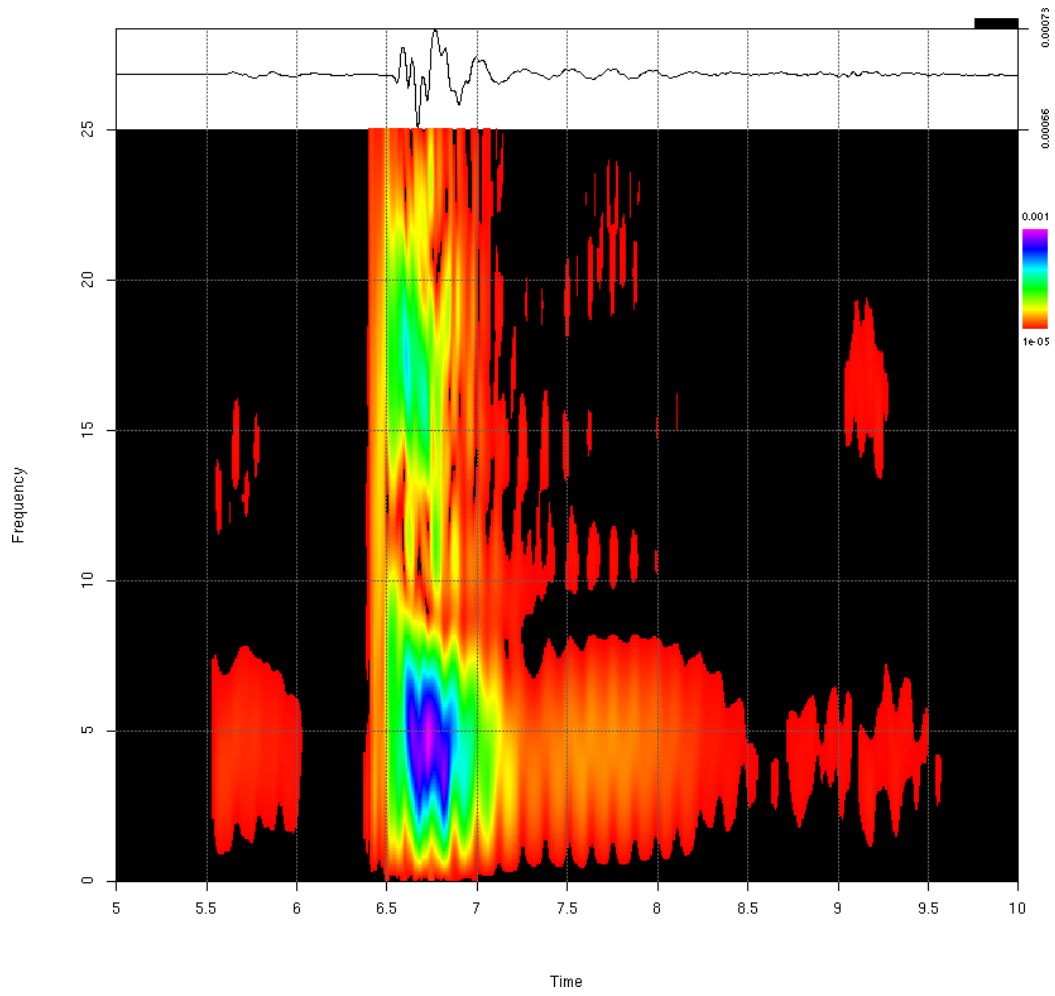Figure 9: Hilbert spectrogram of EEMD of transient seismic signal.

Figure 10: Fourier spectrogram of EEMD of transient seismic signal.