CoV2018

August 15, 2018

0.1 Cities on Volcanoes 2018

0.1.1 Code of the analysis and processing of seismic data of the work entitled "*The 2015 hurricane-induced lahars at Volcán de Colima, México: seismic characterization and numeric modeling*".

```
In [1]: using PyPlot
        using SAC
        using DSP
```

```
In [2]: pa = SAC.read("Patricia250.sac");
```

```
Info: Data are little-endian; byteswapping
```

```
In [3]: rmean!(pa);
        rtrend!(pa);
```

```
In [4]: #Conversion to physical units. V(m/s)= count*1.164153nV/22.8V/(m/s)
        pa.t = pa.t*((1.1641e-9)/(22.8));
```

```
In [5]: #Time vector
        ti = collect(0:pa.npts-1)*pa.delta;
        n = convert(Int32,floor(pa.npts/2)+1);
        tp = (DateTime(2015,10,23,21,0,0,1):(Dates.Millisecond(1))*4:DateTime(2015,10,24,4,0,0))
```

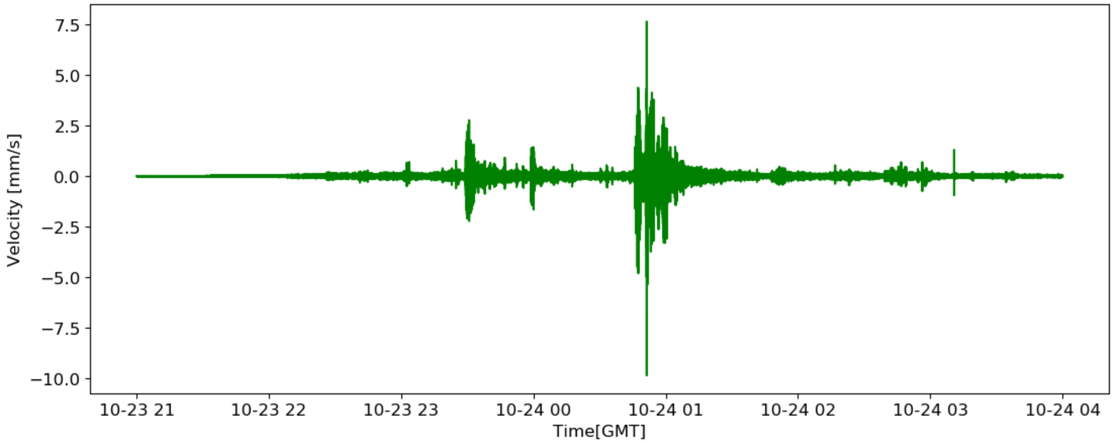
```
In [6]: #Frequency vector
        ds = 1/(pa.delta*pa.npts);
        f = collect(0:pa.npts-1)*ds;
```

```
In [13]: #Movil average function
         function MA(x,n)
             if size(x,1)==1
                 x = x'
             end
             y = zeros(length(x))
             sx = size(x,2)
             tape = NaN*(zeros(convert(Int,floor(n/2)),sx))
             x1 = [tape;x;tape]
             n1 = n-1
```

```
            for ii=1:size(y,1)
                sel = x1[ii+(0:n1),:]
                y[ii]=mean(sel[!isnan.(sel)]);
            end
            return y
        end
```

```
Out[13]: MA (generic function with 1 method)
```

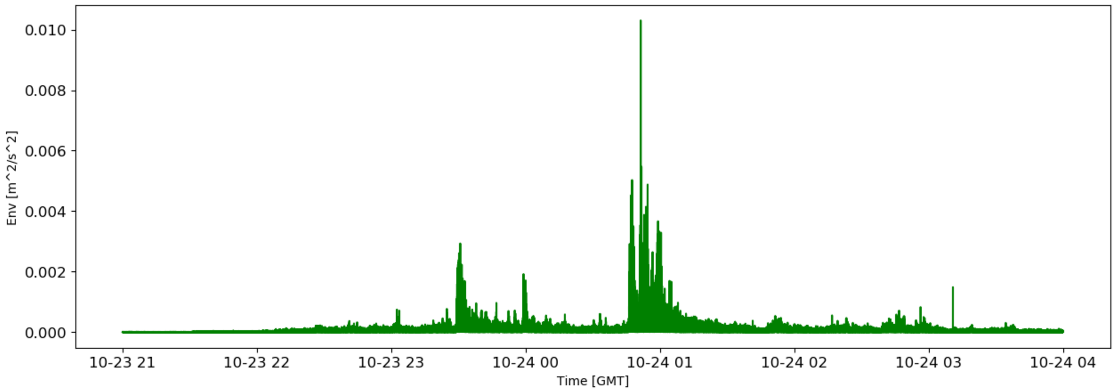
```
In [7]: fig = figure(figsize=(13,5))
        plot(tp,pa.t*1000,"g")
        xlabel("Time[GMT]", fontsize=12)
        xticks(fontsize=12)
        yticks(fontsize=12)
        ylabel("Velocity [mm/s]", fontsize=12)
        #PyPlot.savefig("time.png",dpi=400)
```



```
Out[7]: PyObject Text(25,0.5,'Velocity [mm/s]')
```

```
In [10]: env = abs.(hilbert(pa.t));
```

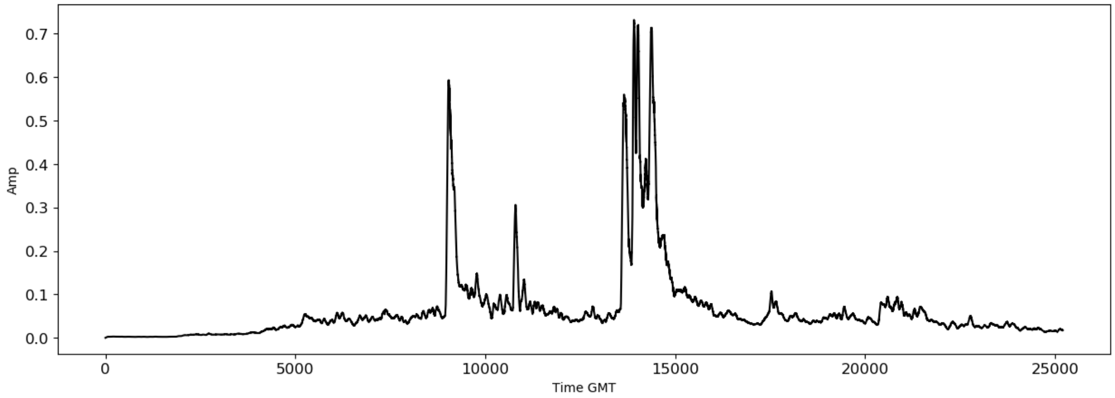
```
In [13]: fig = figure(figsize=(15,5))
         plot(tp, env,"g")
         xlabel("Time [GMT]")
         xticks(fontsize=12)
         ylabel("Env [m^2/s^2]")
         yticks(fontsize=12)
```



```
Out[13]: ([-0.002, 0.0, 0.002, 0.004, 0.006, 0.008, 0.01, 0.012], PyCall.PyObject[PyObject Text(
```

```
In [11]: b = ones(15000)*(1/15000);
         env_1m = filt(b,1,env);
```

```
In [12]: fig = figure(figsize=(15,5))
         plot(ti,(abs.(env_1m)).*1000,"k")
         xlabel("Time GMT")
         xticks(fontsize=12)
         ylabel("Amp")
         yticks(fontsize=12)
```

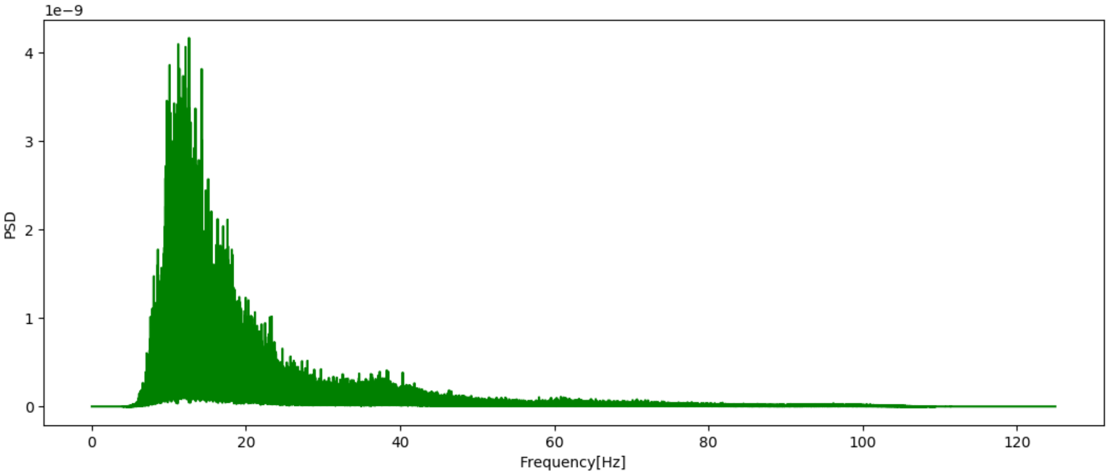


```
Out[12]: ([-0.1, 0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8], PyCall.PyObject[PyObject Text(0,-
```

```
In [14]: #Butterworth filter
         resp = Bandpass(6,124,fs=250)
         desig = Butterworth(4)
         fil = filt(digitalfilter(resp,desig),pa.t);
```

```
In [15]: specf = welch_pgram(fil,fs=250);
```

```
In [14]: fig = figure(figsize=(13,5))
         plot(specf.freq,specf.power,"g")
         xlabel("Frequency[Hz]")
         ylabel("PSD")
```



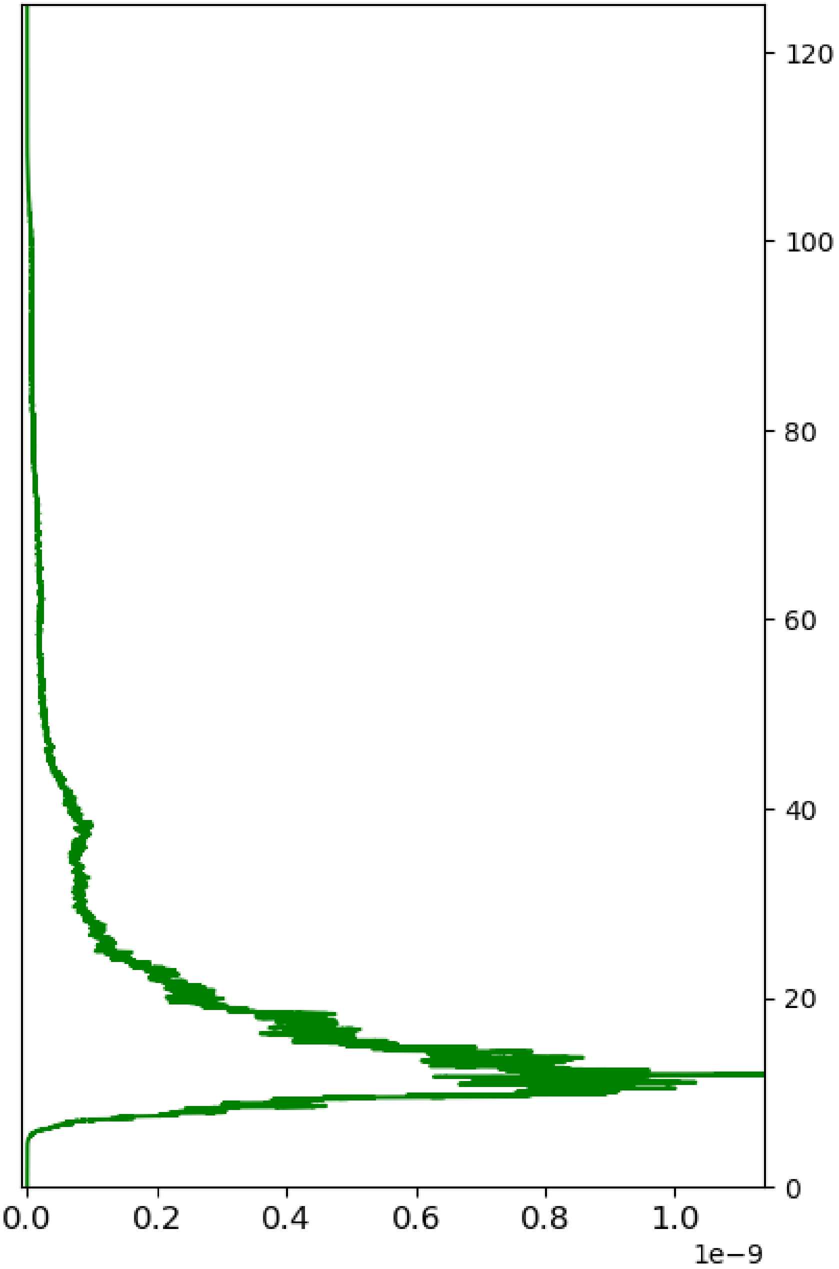
```
Out[14]: PyObject Text(24,0.5,'PSD')
```

```
In [10]: av_spec = MA(specf.power,400);
```

```
WARNING: !(B::BitArray) is deprecated, use .!(B) instead.
Stacktrace:
 [1] depwarn(::String, ::Symbol) at ./deprecated.jl:70
 [2] !(::BitArray{2}) at ./deprecated.jl:57
 [3] MA(::Array{Float64,1}, ::Int64) at ./In[7]:13
 [4] include_string(::String, ::String) at ./loading.jl:522
 [5] include_string(::Module, ::String, ::String) at /home/marv/.julia/v0.6/Compat/src/Compat.jl
 [6] execute_request(::ZMQ.Socket, ::IJulia.Msg) at /home/marv/.julia/v0.6/IJulia/src/execute_re
 [7] (::Compat.#inner#17{Array{Any,1},IJulia.#execute_request,Tuple{ZMQ.Socket,IJulia.Msg}})() a
 [8] eventloop(::ZMQ.Socket) at /home/marv/.julia/v0.6/IJulia/src/eventloop.jl:8
 [9] (::IJulia.##14#17)() at ./task.jl:335
while loading In[10], in expression starting on line 1
```

```
In [17]: fig = figure(figsize=(5,8))
         plot(av_spec,specf.freq,"g")
         xticks(fontsize=12)
         yticks(fontsize=12)
         axis("tight")
         ax=gca()
```

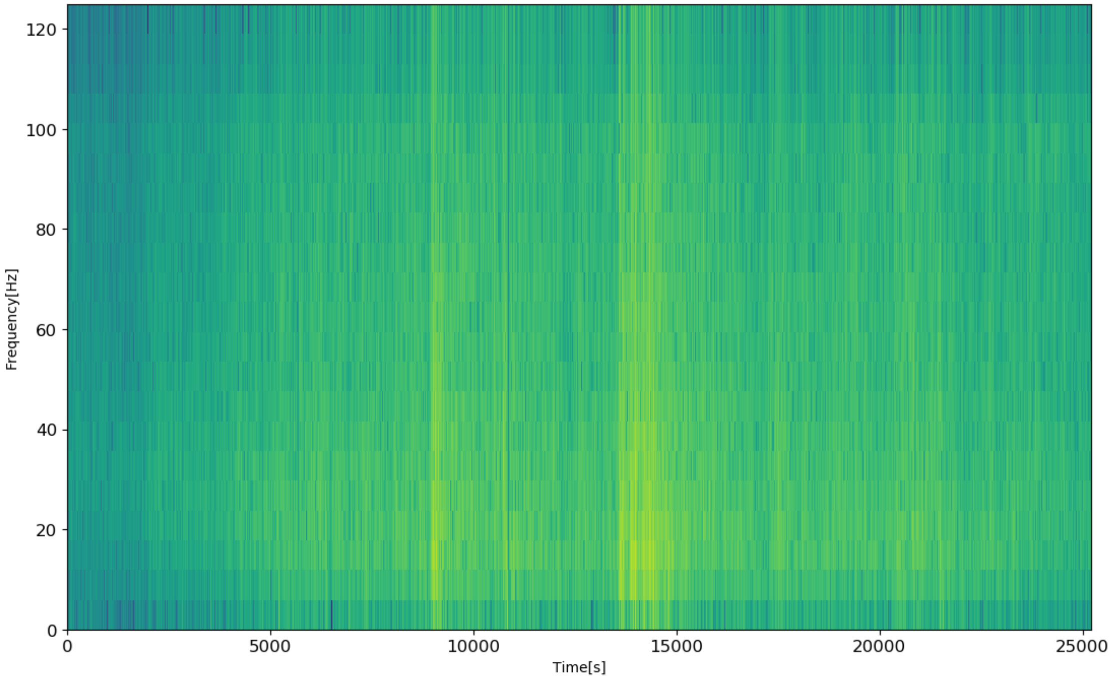
```
        ax[:yaxis][:set_ticks_position]("right")
        xlim(minimum(av_spec)-0.01e-9,maximum(av_spec))
        ylim(minimum(specf.freq),maximum(specf.freq))
        #PyPlot.savefig("psd_covs.png",dpi=400)
```



```
Out[17]: (0.0, 125.0)
```

```
In [21]: fig = figure(figsize=(13,8))
         specgram(pa.t,125,250,pad_to=40,noverlap=50);
```

```
        xlabel("Time[s]")
        xticks(fontsize=12)
        ylabel("Frequency[Hz]")
        yticks(fontsize=12)
        PyPlot.savefig("spec.png",dpi=300)
```

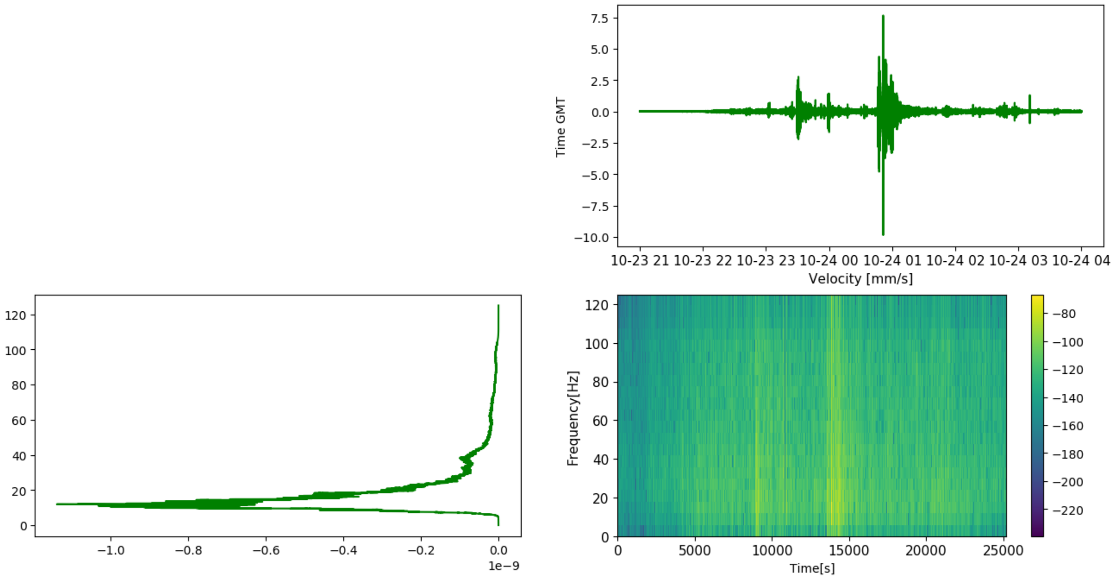


```
In [33]: fig = figure(figsize=(13,5))
         sp = specgram(pa.t,125,250,pad_to=40,noverlap=50);
         xlabel("Time[s]")
         xticks(fontsize=12)
         ylabel("Frequency[Hz]")
         yticks(fontsize=12)
```

```
        UndefVarError: heatmap not defined



        Stacktrace:

         [1] include_string(::String, ::String) at ./loading.jl:522
```

```
In [40]: fig = figure(figsize=(16,8))
         subplot(2,2,2)
```

```
#fig = figure(figsize=(10,3))
plot(tp,pa.t*1000,"g")
xlabel("Velocity [mm/s]",fontsize=11)
xticks(fontsize=11)
ylabel("Time GMT",fontsize=10)
yticks(fontsize=10)
subplot(2,2,3)
#fig = figure(figsize=(10,5))
plot(av_spec.*-1,specf.freq,"g")
#axis("tight")
#ax=gca()
#ax[:spines]["left"][:set_position]("center")
#xlabel("Frecuencia[Hz]")
#ylabel("PSD")
subplot(2,2,4)
#fig = figure(figsize=(10,5))
specgram(pa.t,125,250,pad_to=40,noverlap=50);
xlabel("Time[s]")
xticks(fontsize=11)
ylabel("Frequency[Hz]", fontsize=11)
yticks(fontsize=11)
colorbar()
```



```
Out[40]: PyObject <matplotlib.colorbar.Colorbar object at 0x7f1d3954b198>
```