HydroGeoSines

Signal In the Noise Exploration Software for hydrogeological datasets

Code demonstration

First, we import the sines package and create a new instance of a sines model.

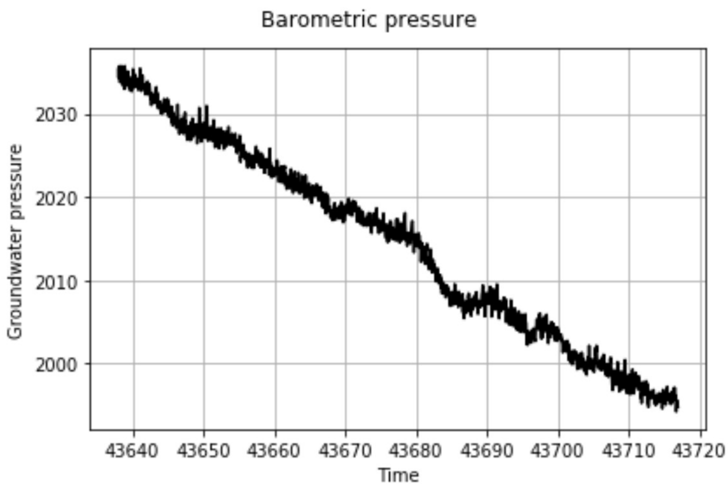
In [1]:
```
import hydrogeosines

s = hydrogeosines.model()
```

Next, we import groundwater pressure data, take a look at the first ten values, and then visualise the full dataset.

In addition to the barometric pressure dataset, we can assess one of three groundwater pressure datasets: RN027214, RN039613, or RN039617.

In [2]:
```
s.wd = 'hydrogeosines/test_data/port_keats/'
s.id = 'RN039613'

s.get_GW()

s.print_GW(10)

s.plot_GW()
```

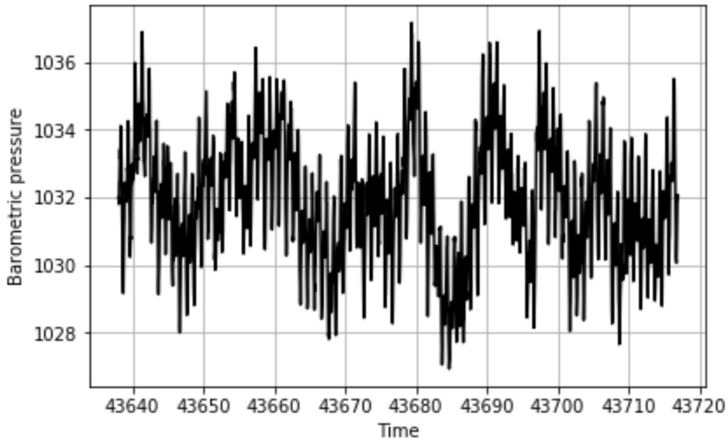
```
      time  pressure
 43638.000  2035.359
 43638.040  2034.306
 43638.080  2035.704
 43638.130  2034.636
 43638.170  2034.715
 43638.210  2035.108
 43638.250  2033.772
 43638.290  2035.249
 43638.330  2035.075
 43638.380  2035.768
```



Next, we import groundwater pressure data, take a look at the first ten values, and then visualise the full dataset.

In [3]:
```
s.get_BA()

s.print_BA(10)

s.plot_BA()
```

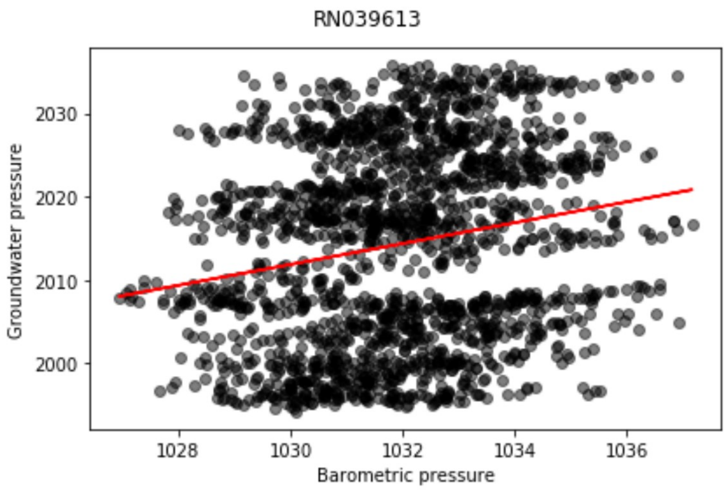
```
      time  pressure
 43638.000  1032.010
 43638.040  1032.004
 43638.080  1031.804
 43638.130  1032.470
 43638.170  1032.487
 43638.210  1033.031
 43638.250  1033.424
 43638.290  1033.127
 43638.330  1034.121
 43638.380  1033.933
```



Next, we calculate the linear regression of groundwater pressure versus barometric pressure, print the resulting parameters, and then plot the data and the estimated linear function.

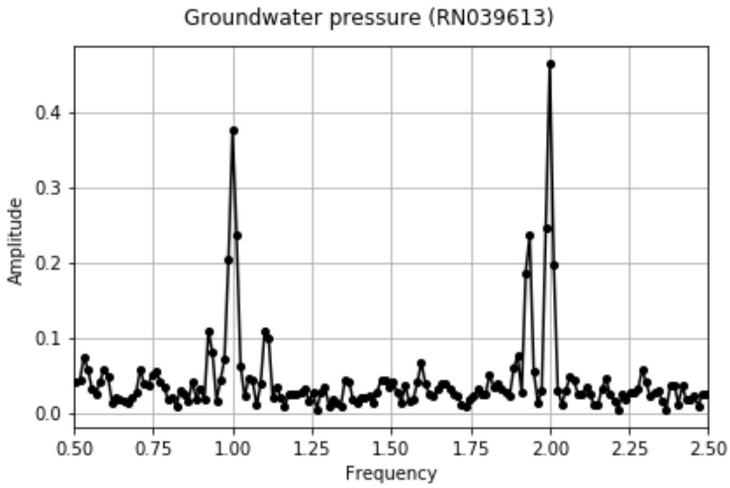
In [4]:
```
s.calc_linear_GW()

%matplotlib inline
s.plot_linregress(pname=None)
```

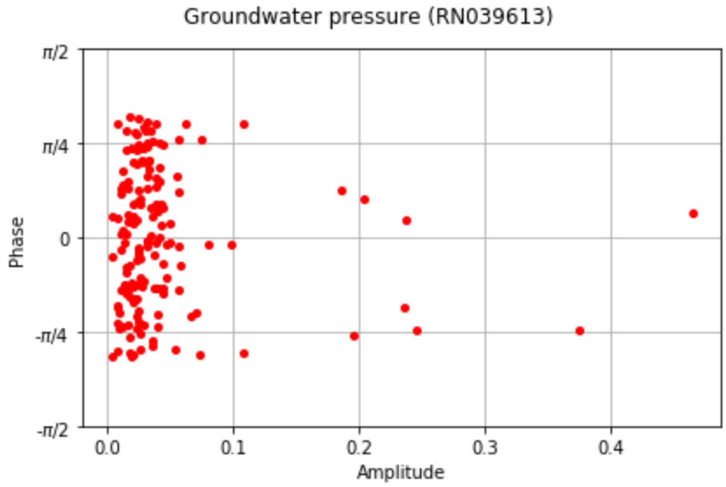
```
     1.251   723.807
```



Next, we calculate the discrete Fourier transform of the groundwater pressure dataset and plot the resulting amplitude spectrum, as well as an amplitude versus phase plot, in order to identify dominant component frequencies.

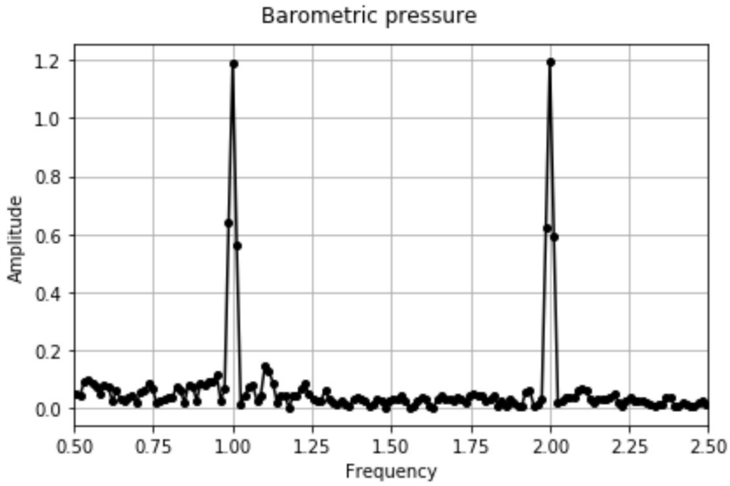
In [5]:
```
s.calc_ft_GW()
%matplotlib inline
s.plot_ft_avf_GW(pname=None)
s.plot_ft_pva_GW(pname=None)
```

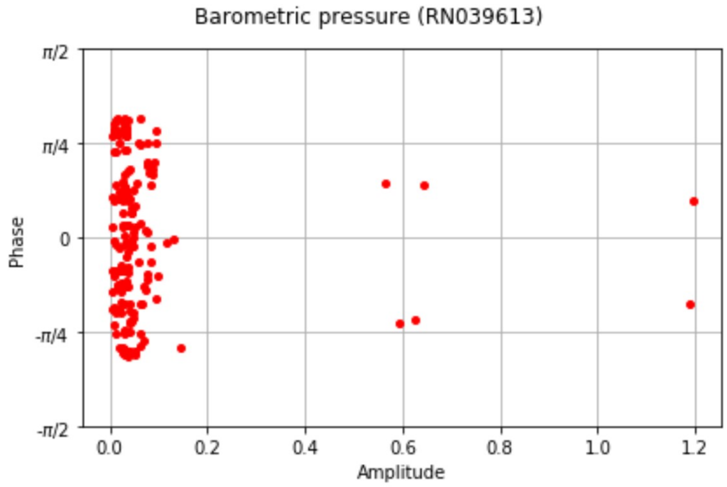




Next, we repeat the analysis for the barometric pressure dataset.

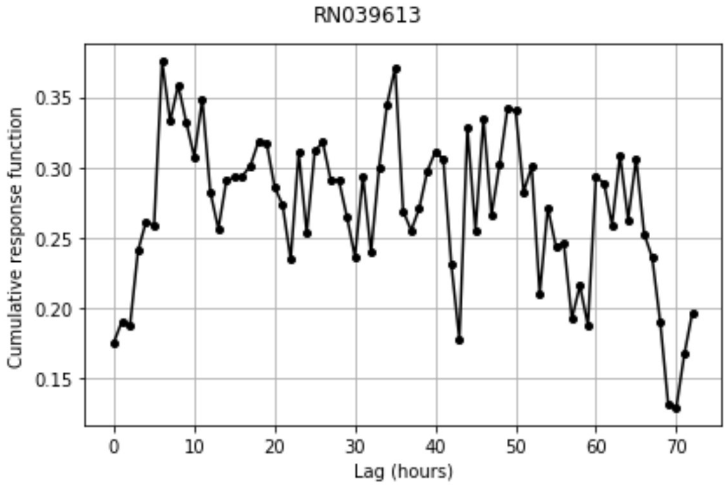
```
In [6]: s.calc_ft_BA()
        %matplotlib inline
        s.plot_ft_avf_BA(pname=None)
        s.plot_ft_pva_BA(pname=None)
```

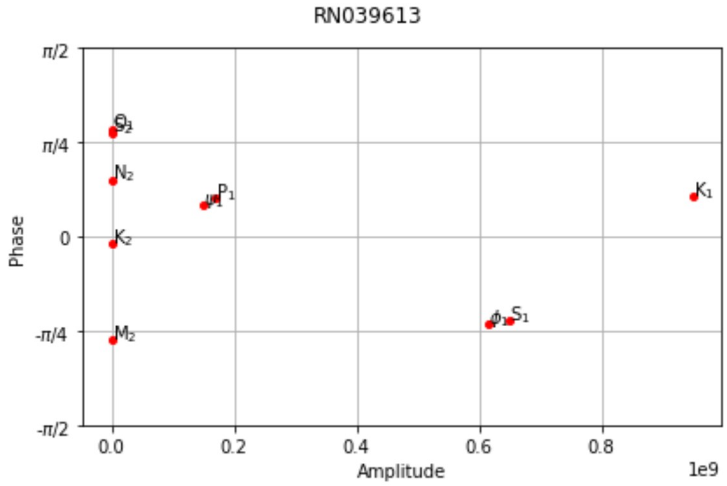




Next, we use regression deconvolution to estimate the amplitudes and phases of specific frequencies of interest, and to estimate a cumulative response function.

In [7]:
```
s.calc_regress_deconv()
s.plot_regress_deconv_crf(pname=None)
s.plot_regress_deconv_pva(pname=None)
```





In [ ]: