

# trial1

July 10, 2020

## Trial Envi Contour Plot with WDRT

```
[1]: import numpy as np
import scipy.io
import WDRT.ESSC as ESSC
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
import scipy.interpolate as interp
import WDRT.longTermExtreme as lte
import WDRT.shortTermExtreme as ste
import h5py
import os
import csv
```

```
[38]: # Load data from example_envSampling.py
#envFile = h5py.File(os.path.join(r'data', 'NDBC46022.h5'), 'r')
#envFile = h5py.File(os.path.join(r'data/NDBC46022.h5'), 'r')
#Hs_Return = np.array(envFile['ReturnContours/Hs_Return'])
#T_Return = np.array(envFile['ReturnContours/T_Return'])
#Hs_sample = np.array(envFile['Samples_ContourApproach/Hs_SampleCA'])
#T_sample = np.array(envFile['Samples_ContourApproach/T_SampleCA'])
data=np.genfromtxt('siteTestdata.txt', delimiter=" ", skip_header=1)
```

```
[39]: # Some characteristics of the data
print(data[:5]) #first 5 rows
print(data.ndim) # of dimensions e.g. 2 for 2-D array
print(data.size)# size of dimension
print(data.shape)# product of this gives size size
data.dtype # data type(s)
```

```
[[0.439 7.011 1.43 ]
 [0.483 7.017 2.07 ]
 [0.493 7.036 2.93 ]
 [0.476 7.035 3.21 ]
 [0.533 7.014 3.74 ]]
```

2

43824

(14608, 3)

```
[39]: dtype('float64')
```

```
[42]: #Slicing up the data
swh=data[:,0] #significant wave height
pwp=data[:,1] #significant wave period
wind=data[:,2]
print(swh)
print(pwp)
print(wind)
print(swh.shape)
print(pwp.shape)
print(wind.shape)
```

```
[0.439 0.483 0.493 ... 0.114 0.16 0.094]
[7.011 7.017 7.036 ... 8.246 8.196 8.129]
[1.43 2.07 2.93 ... 2.49 3.91 5.09]
(14608,)
(14608,)
(14608,)
```

```
[40]: # Create buoy object
buoy16 = ESSC.Buoy('16', 'NDBC')
```

```
[44]: #buoy16.loadFromTxt(r'NBDC46022')

buoy16.Hs=swh
buoy16.Tp=pwp
```

```
[ ]: # Create PCA EA object for the buoy
pca16 = ESSC.PCA(buoy16)
```

```
[45]: # Declare required parameters
Time_SS = 1. # Sea state duration (hrs)
Time_R = 100 # Return periods (yrs) of interest
```

```
[46]: # Create PCA EA object for the buoy
pca16 = ESSC.PCA(buoy16)
```

↳ -----

AttributeError Traceback (most recent call↳  
↳last)

```
<ipython-input-46-b40fddcc8326> in <module>
1 # Create PCA EA object for the buoy
```

```

----> 2 pca16 = ESSC.PCA(buoy16)

c:
↪ \users\awhite15\appdata\local\programs\python\python36\lib\site-packages\WDRT\ESSC.
↪ py in __init__(self, buoy, size_bin)
    927         self.Weight_points = None
    928
--> 929         self.coeff, self.shift, self.comp1_params, self.sigma_param,
↪ self.mu_param = self.__generateParams(self.size_bin)
    930
    931     def __generateParams(self, size_bin=250.0):

c:
↪ \users\awhite15\appdata\local\programs\python\python36\lib\site-packages\WDRT\ESSC.
↪ py in __generateParams(self, size_bin)
    931     def __generateParams(self, size_bin=250.0):
    932         pca = skPCA(n_components=2)
--> 933         pca.fit(np.array((self.buoy.Hs - self.buoy.Hs.mean(axis=0),
↪ self.buoy.T - self.buoy.T.mean(axis=0))).T)
    934         coeff = abs(pca.components_) # Apply correct/expected sign
↪ convention
    935         coeff[1, 1] = -1.0 * coeff[1, 1] # Apply correct/expected
↪ sign convention

```

AttributeError: 'list' object has no attribute 'mean'

```
[34]: # Calculate contour using PCA method
pca_Hs_Return, pca_T_Return = pca16.getContours(Time_SS, Time_R)
```

```
[36]: import ipyml
%matplotlib widget
```

```
[37]: # Show a plot of the data
pca16.plotData()
```

Canvas(toolbar=Toolbar(toolitems=[('Home', 'Reset original view', 'home', 'home'), ('Back', 'B

```
[47]: #%lsmagic
```

```
[48]: #pwd
```

```
[50]: #matplotlib inline
```

```
[53]: #Testing Matplotlib
      # "Sample Demo"
      #import numpy as np
      #import matplotlib.pyplot as plt

      #N=50
      #x=np.random.rand(N)
      #y=np.random.rand(N)
      #colors=np.random.rand(N)
      #plt.scatter(x,y)
      #plt.show()
```

```
[52]: #import pandas as pd

      #df=pd.DataFrame(np.random.randn(10,5))
      #df.head
```

```
[ ]:
```