/usr/bin/python

```
import matplotlib
3
   #import pyreport
   import numpy as np
   from pylab import *
6
   #from pylab import show
7
   from matplotlib import pyplot as pl
8
   x_t,y_t_unsc = np.loadtxt('data/CG10e2t.csv', delimiter=',', unpack=True, usecols
10
      = [0,1]
   x_x, y_x_unsc = np.loadtxt('data/CG10e2x.csv', delimiter=',', unpack=True, usecols
11
      = [0,1]
12
   x_y,y_unsc = np.loadtxt('data/CG10e2y.csv', delimiter=',',unpack=True, usecols
      = [0,1]
   x_z,y_z_unsc = np.loadtxt('data/CG10e2z.csv', delimiter=',',unpack=True, usecols
13
      = [0,1]
                 = np.loadtxt('data/LO_water.csv', delimiter=',', unpack=True, usecols
14
   x_w, y_w
       = [0,1]
15
   #x_t_eV ,y_t_unsc = np.loadtxt('data/CG10e2t.csv',delimiter=',',unpack=True,
16
      usecols = [0,1])
   #x_x_eV ,y_x_unsc = np.loadtxt('data/CG10e2x.csv',delimiter=',',unpack=True,
17
      usecols = [0,1])
   #x_y_eV ,y_y_unsc = np.loadtxt('data/CG10e2y.csv',delimiter=',',unpack=True,
18
      usecols = [0,1])
   #x_z_eV ,y_z_unsc = np.loadtxt('data/CG10e2z.csv',delimiter=',',unpack=True,
19
      usecols = [0,1])
20
   \#x\_w\_eV , y\_w
                     = np.loadtxt('data/LO_water.csv', delimiter=',', unpack=True,
      usecols = [0,1])
2.1
22
   \#x_t = x_t_eV/6.582e-16 \# convert eV to inverse sec by eV/hbar = ev/(eV*s)=eV/6.6
   \#x_x = x_x_eV/6.582e-16 \# convert eV to inverse sec by eV/hbar = ev/(eV*s)=eV/6.6
23
      e - 16
   \#x_y = x_y = 0/6.582e - 16 \# convert eV to inverse sec by eV/hbar = ev/(eV*s) = eV/6.6
24
      e - 16
   \#x_z = x_z = V/6.582e-16 \# convert eV to inverse sec by eV/hbar = ev/(eV*s)=eV/6.6
2.5
      e - 16
26
   \#x\_w = x\_w\_eV/6.582e-16 \# convert eV to inverse sec by eV/hbar = ev/(eV*s)=eV/6.6
      e-16
27
28
   y_t = y_t_unsc*4.949
   y_x = y_x_unsc * 4.949
29
   y_y = y_y_unsc * 4.949
30
   y_z = y_z_unsc*4.949
31
32
   ## DEFINE FUNCTIONS FOR CALCULATING e(iz)
33
   # Matsubara frequencies: z_n at room temp is (2pikbT/hbar)*n (ie coeff*n)
34
   coeff = 0.159 \# in eV \# (2.41 * 1e14) \# in rad/s
35
36
   \#coeff = 2.41e14 \# in (1 rad)*(1/s)=inverse seconds
37
   T = 300.0
38
   \#kb_J = 1.3806488e-23 \# in J/K
39
40
   \#hbar = 6.625e-34 \# in J/s
41
   \#coeff_J = 2.0*np.pi*kb_J*T/hbar\#1.602e-19*0.159e15 \# in eV \#(2.41*1e14) \# in
      rad/s
   n = arange(0,500)
42
43
   z = n * coeff
   \#coeff_J = 1.602e-19*0.159e15 \# in eV \#(2.41*1e14) \# in rad/s
44
45
```

```
#z = n * coeff
46
47
    \#z = n * coeff_J
48
49
    eiz_x = empty(len(z))
    eiz_y = empty(len(z))
50
51
    eiz_z = empty(len(z))
    eiz_w = empty(len(z))
52
53
54
    eiz \times arg = empty(len(x x))
    eiz_y_arg=empty(len(x_y))
55
56
    eiz_z_arg = empty (len (x_z))
57
    eiz_w_arg=empty(len(x_w))
58
59
    for j in range(len(z)):
        for i in range (len (x_x)):
60
61
            eiz_x_arg[i]=x_x[i]*y_x[i] / (x_x[i]**2 + z[j]**2)
        eiz_x[j] = 1 + (2./pi) * trapz(eiz_x_arg,x_x)
62
63
        for k in range (len (x_y)):
64
65
            eiz_y_arg[k]=x_y[k]*y_y[k] / (x_y[k]**2 + z[j]**2)
66
        eiz_y[j] = 1 + (2./pi) * trapz(eiz_y_arg,x_y)
67
        for m in range(len(x_z)):
68
            eiz_z_arg[m]=x_z[m]*y_z[m] / (x_z[m]**2 + z[j]**2)
69
70
        eiz_z[j] = 1 + (2./pi) * trapz(eiz_z_arg,x_z)
71
72
        for p in range(len(x_w)):
73
            eiz w arg[p]=x w[p]*y w[p] / (x w[p]**2 + z[j]**2)
74
        eiz_w[j] = 1 + (2./pi) * trapz(eiz_w_arg,x_w)
   #savetxt("data/eiz_x_output.txt", eiz_x)
75
    #savetxt("data/eiz_y_output.txt", eiz_y)
76
77
    #savetxt("data/eiz_z_output.txt", eiz_z)
78
    #savetxt("data/eiz_w_output.txt", eiz_w)
79
    savetxt("data/eiz_x_output_eV.txt", eiz_x)
80
    savetxt("data/eiz_y_output_eV.txt", eiz_y)
81
    savetxt("data/eiz_z_output_eV.txt", eiz_z)
82
    savetxt("data/eiz_w_output_eV.txt", eiz_w)
83
84
    #pl.figure()
85
                          color = 'k', label = 'total')
86
   #pl.plot(x_t,y_t,
   \#pl.plot(x_x+10,y_x, color = 'b', label = r' \uparrow \land \{x\} \uparrow')
87
   \#pl.plot(x_y+20,y_y, color = 'g', label = r'$\hat{y}$')
88
   \#pl.plot(x_z+30,y_z, color = 'r', label = r' \$ \setminus hat \{z\} \$')
89
   \#pl.xlabel(r'\$\hbar\omega\,\,\,\[eV]\,\,\,!shifted\,\,10\,eV\,for\,visualization\$',
90
        size = 24)
    #pl.ylabel(r'$\varepsilon^{\prime\prime}(\omega)$', size = 24)
91
92
   #pl.legend()
93
   #pl.title(r'CG-10 DNA eps2... shifted for visualization')
    \#\#pl.savefig('plots/131010\_Hopkins\_CG10\_eps2\_x\_z.png', dpi = 300)
94
    #pl.savefig('plots/131010_Hopkins_CG10_eps2_all_directions.pdf')
95
96
   ##pl.show()
97
98
    pl.figure()
    pl.plot(x_t,y_t, color = 'k', label = r'$\varepsilon^{\prime\prime}_{total}(\
99
       omega)$')
    pl.plot(x_x, y_x, color = 'b', label = r'\alpha^{\star}\varepsilon^{\prime\prime}_\hat{x}(\
100
       omega)$')
    pl.plot(x_y, y_y, color = 'g', label = r'varepsilon^{\prime}_hat\{y\}(\
101
       omega)$')
```

```
102
        pl.plot(x_z, y_z, color = 'r', label = r'v_z) varepsilon v_z varepsilon v_z varepsilon v_z
              omega)$')
        pl.plot(x_w, y_w, color = 'c', label = r'v_v | varepsilon v_v | prime v_v | v_v | prime v_v | v_
103
              omega)$')
        pl.xlabel(r'\$\hbar\omega\,\,\,\,[eV]\$', size = 24)
104
        pl.ylabel(r'\\varepsilon^{\prime\prime}(\omega)$', size = 24)
105
        pl.legend()
106
        pl. title (r'CG-10 DNA and water eps2')
107
        pl.show()
108
109
        #pl.close()
       #imshow(1)
110
       #pl.savefig('plots/131010_Hopkins_CG10_eps2_x_z.png', dpi = 300 )
111
       #pl.savefig('plots/131010_Hopkins_CG10_eps2_x_z.pdf')
112
113
       ##pl.figure()
114
       ###pl.plot(x_t,y_t, label = 'total')
115
       \#\#pl.plot(x_x,y_x, label = r'\$\hat{x}
116
       \#\#pl.plot(x_y,y_y, label = r'\$\hat{y}$')
117
       \#\#pl.plot(x_z,y_z, label = r'\$\hat{z}$')
118
       \#\#pl.xlabel(r'\$\hbar\omega\,\,\,\[eV]\$', size = 24)
119
       ##pl.ylabel(r'$\varepsilon^{\prime\prime}(\omega)$', size = 24)
120
121
       ##pl.legend()
        ##pl.show()
122
123
        ###pl.close()
124
        ###imshow(1)
       ##pl.savefig('plots/DNA_spectra_x_y.png', dpi = 300 )
125
126
       #
127
        pl.figure()
        pl.plot(n, eiz_x, color = 'b', label = r' varepsilon_{\hat{x}}(i zeta_{N})')
128
        pl.plot(n,eiz_y, color = 'g', label = r'\varepsilon_{\hat{y}}(i\zeta_{N})\$')
129
        pl.plot(n,eiz\_z, color = 'r', label = r'\$\varepsilon\_\{\hat\{z\}\}(i\zeta\_\{n\})\$')
130
        pl.plot(n,eiz_w, color = 'c', label = r'\varepsilon_{\hat{w}}(i\zeta_{n})$')
131
132
        pl.xlabel(r'$N$', size = 24)
        pl.ylabel(r'$\varepsilon(i\zeta)$', size = 24)
133
        pl.legend()
134
        pl. title (r'CG-10 DNA eiz')
135
        #pl.savefig('plots/DNA_eiz_x_z.png', dpi = 300 )
136
        #pl.savefig('plots/131010_Hopkins_CG10_eiz.pdf')
137
        pl.show()
138
139
        #pl.close()
```