[25]

```
import numpy as np
import matplotlib.pyplot as plt
import scipy.optimize as opt
import os
import glob
```

[26]

```
files = glob.glob('*.csv')
```

[27]

```
files
```

```
['S35_300ml_big_wave.csv',
 'S35_300ml_small_wave.csv',
 'S135_test1_big_wave.csv',
 'S35_350ml_small_wave.csv',
 'S35_150ml_big_wave.csv',
 'S35_600ml_big_wave.csv',
 'S135_test2_small_wave.csv',
 'S35_150ml_small_wave.csv',
 'S135_test2_big_wave.csv',
 'S35_350ml_big_wave.csv',
 'S135_test1_small_wave.csv']
```

[28]

```
files.sort()
```

[29]

```
def wavefit(name):
    filename = name
    x, y = np.loadtxt(filename,skiprows=1, delimiter=',', unpack=True)
    name = os.path.splitext(filename)[0]
    def fitfunc(x, A, λ, x0, y0):
        return A * 1/np.cosh((x-x0)/λ)**2 + y0
    Aguess = max(y)
    λguess = x[2*len(x)//3] - x[len(x)//3]
    x0guess = np.mean(x)
    y0guess = min(y)
    p0 = [Aguess,λguess,x0guess,y0guess]
    p, pcov = opt.curve_fit(fitfunc, x, y, p0=p0)
    plt.plot(x, y, 'o',label=name+' Collected Data')
    plt.plot(x, fitfunc(x, *p),label=name + ' Fitted Data')
    plt.xlabel('distance (cm)')
    plt.ylabel('distance (cm)')
    plt.legend(loc='best')
    plt.savefig(name +'.png', bbox_inches = 'tight')
    plt.close()
    dp = np.sqrt(np.diag(pcov))
    plt.errorbar(x,fitfunc(x,*p),fitfunc(x,*dp),capsize=2)
    plt.savefig(name + 'error.png', bbox_inches='tight')
    plt.close()
    #data = open(name +'output.csv','w')
    #data.writelines('p , dp \n')
    #np.savetxt(name+'output.csv', np.transpose([p,dp]),delimiter=",")
    #data.close()
    return
```

[30]

```
for x in files:
    wavefit(x)
```

```
/anaconda3/lib/python3.6/site-packages/ipykernel_launcher.py:6:
RuntimeWarning: overflow encountered in square
```

[ ]