In [1]:

```
###################################################################
######################
#    Filename : Proj_Resonance_Lorentzian_GitHub_Run_Me.ipynb
#    Date : Aug 25, 2015
#    What : Runs the main file, which is def_resonance_lorentzian.p
y
#           You must give it four parameters
#
###################################################################
######################
import scipy
import numpy as np
import def_resonance_lorentzian_for_github as drl
import matplotlib.pyplot as plt
%matplotlib inline
plt.rcParams['figure.figsize']=(18,6)

''' Tell me the file which you wanna look at'''
dset = np.load('/Users/TzeGoh/Desktop/2015-07-01-starcryo-nevins-in
itial-sweep.npz')
dset.keys()
print "Keys:",dset.keys()
```

```
Keys: ['lo', 'freq', 'data']
```

In [2]:

```
''' This cell tells you the main variables --> x-axis(freq) and y-a
xis (data)'''
input_x = raw_input ("What is the frequency ? usually 'freq', don't
put parathensis: ")
input_y = raw_input ("What is the data ? usually 'data', do not put
parathensis: ")
freq = dset[input_x]
raw_data = dset[input_y]
data = raw_data.copy()
print 'Length of data:', len(data)
```

```
What is the frequency ? usually 'freq', don't put parathensis: fre
q
What is the data ? usually 'data', do not put parathensis: data
Length of data: 32768
```

In [3]:

```
''' Just toggle these 4 parameters'''
flat = 1.2              # flat is the gradient of the points around
centre // flat = 1.2
move = 5                # move is the number of points around of ce
ntre    // move = 5
noise = 5               # noise is how many points away from 'move'
// noise = 5
tol = 1                 # tolerance determine spread of lorentz fun
ction   // tol = 1
```

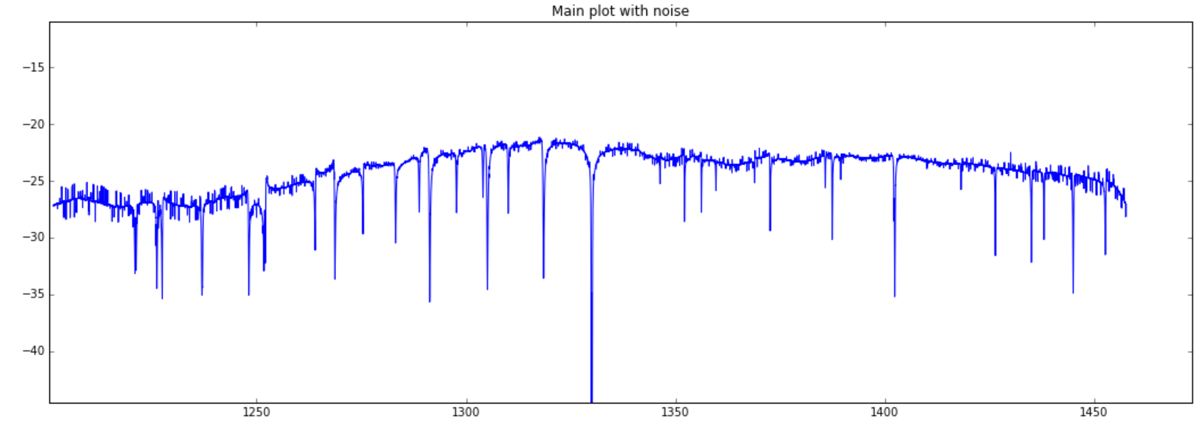
In [4]:

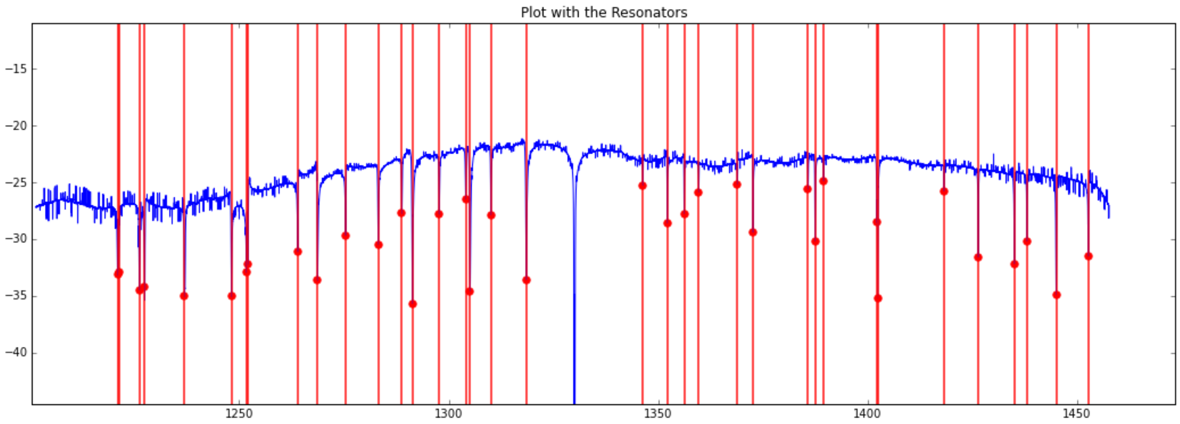
```
''' This cell fixes x and y for the rest of the programme, as well as where to look for the minimums  '''
a = freq
b = data
x = a
y = 20 * np.log10(np.abs(b))
locmins,order_of_mins = drl.resonators(flat,move,noise,tol,x,y)
ranges_to_look, freq_to_look, data_to_look = drl.where_do_i_look(locmins,x,y)
xstart, xend, ystart, yend = drl.box_in_the_plot(ranges_to_look, freq_to_look, data_to_look)
```

In [5]:

```
'''This shows you the main plot'''
drl.main_plot_with_noise(x,y,xstart,xend,ystart,yend)
plt.figure()

'''This shows you where the resonators are, and prints out those points for you'''
drl.show_me_resonators(order_of_mins,data,x,y,xstart,xend,ystart,yend)
plt.figure()
```

```
         length of data 32768
         There are 36 resonators, and they occur at ...'
         Resonator        Frequency        data numbers
         1                1220.9765625     -33.1219951956
         2                1221.25          -32.8532786777
         3                1226.171875      -34.4539471071
         4                1227.46875       -34.2255308117
         5                1236.9609375     -34.9767043493
         6                1248.1484375     -35.0230008177
         7                1251.75          -32.9079550993
         8                1252.0859375     -32.1999891693
         9                1263.9765625     -31.0803250273
         10               1268.703125      -33.6111148132
         11               1275.3984375     -29.6476439313
         12               1283.171875      -30.4361533103
         13               1288.8203125     -27.6973588984
         14               1291.328125      -35.6302226431
         15               1297.7265625     -27.7788344157
         16               1304.0390625     -26.4336752332
         17               1305.1015625     -34.5468471503
         18               1310.0625        -27.8254459921
         19               1318.4921875     -33.5482191418
         20               1346.328125      -25.2203384846
         21               1352.125         -28.5604651254
         22               1356.1640625     -27.7355898259
         23               1359.65625       -25.8192480918
         24               1368.8828125     -25.1404558801
         25               1372.59375       -29.3762206551
         26               1385.734375      -25.5779921556
         27               1387.3984375     -30.1442885274
         28               1389.4453125     -24.8547588908
         29               1402.1328125     -28.4974608824
         30               1402.328125      -35.1776781685
         31               1418.140625      -25.7348940454
         32               1426.3515625     -31.5536835678
         33               1434.9453125     -32.1421898909
         34               1437.9453125     -30.1347538849
         35               1444.921875      -34.8617774542
         36               1452.6015625     -31.4758264322

Out[5]:  <matplotlib.figure.Figure at 0x10884ff10>
```



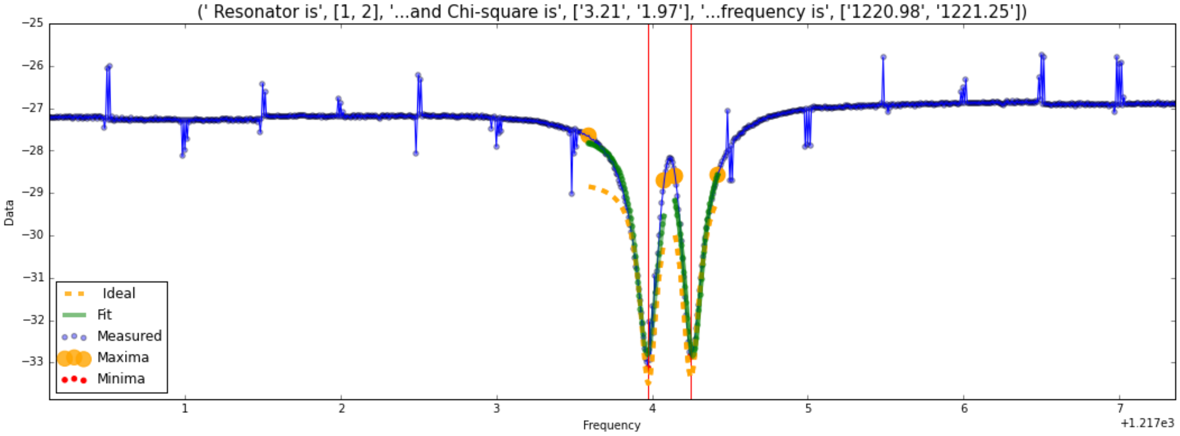

Plot with the Resonators
−15
−20
−25
−30
−35
−40
1250
1300
1350
1400
1450

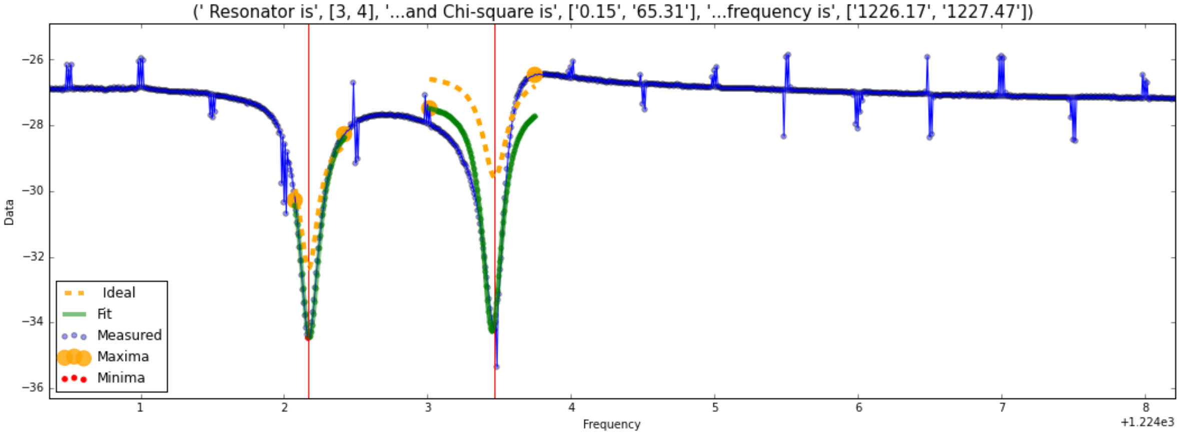
```
<matplotlib.figure.Figure at 0x10884ff10>
```

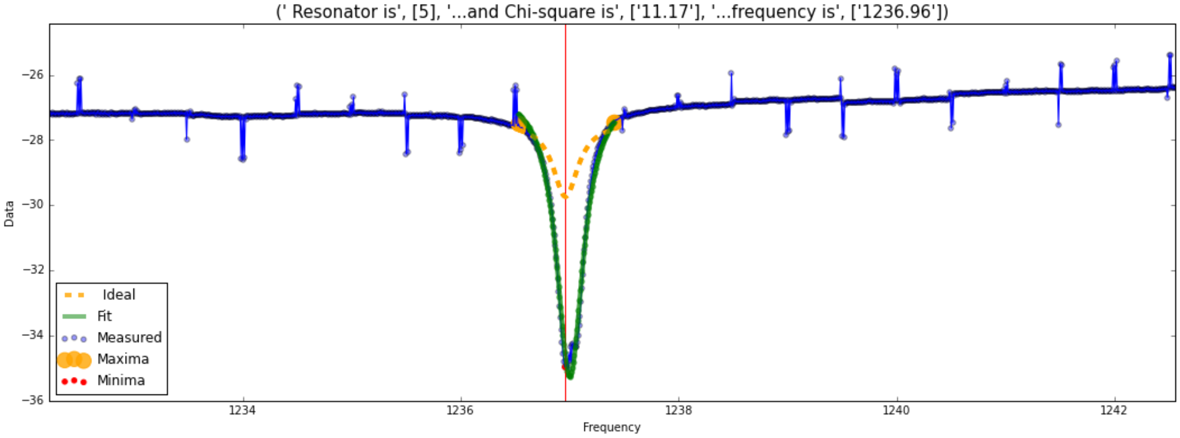
In [6]:

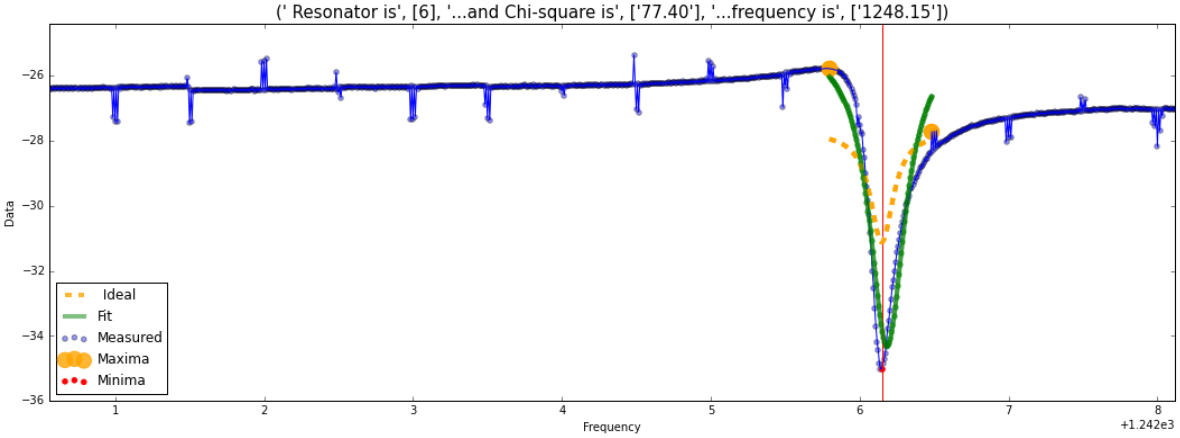
```
'''And now, you get the option to see the resonators close up, and
their chi-squared to the Lorentz function'''
input_a = raw_input ( "Enter : \
1)See Resonance close up \
2)See Resonance close-up w/ Lorentzian only ( This is usually enoug
h ) \
3)See Resonance close-up, and then see Resonance close-up Lorentzia
n "
                    )
if input_a == '1':
    drl.show_me_resonators_in_close_range(ranges_to_look,freq_to_lo
ok,data_to_look,\
                                          x,y,order_of_mins)

elif input_a == '2':
    count_chi, count_chi_less_than_1, chi_squared_total,freq_point
s_total, \
    length_of_chi_squared_total = drl.show_me_resonators_in_close_r
ange_with_lorentzian\
    (
        ranges_to_look,freq_to_look,data_to_look,x,y,move,noise,to
l,flat,\
     order_of_mins,locmins\
    )
    drl.print_me_those_chi_square\
    (count_chi,count_chi_less_than_1,chi_squared_total,freq_point
s_total)

elif input_a == '3':
    drl.show_me_resonators_in_close_range(ranges_to_look,freq_to_lo
ok,data_to_look,\
                                          x,y,order_of_mins)

    count_chi, count_chi_less_than_1, chi_squared_total,freq_point
s_total, \
    length_of_chi_squared_total = drl.show_me_resonators_in_close_r
ange_with_lorentzian\
    (

        ranges_to_look,freq_to_look,data_to_look,x,y,move,noise,to
l,flat,\
     order_of_mins,locmins\
    )
    drl.print_me_those_chi_square(count_chi,count_chi_less_than_1,c
hi_squared_total,\
                                  freq_points_total)

else :
    print "I'm sorry, I don't recognise that input. Please try agai
n"
```

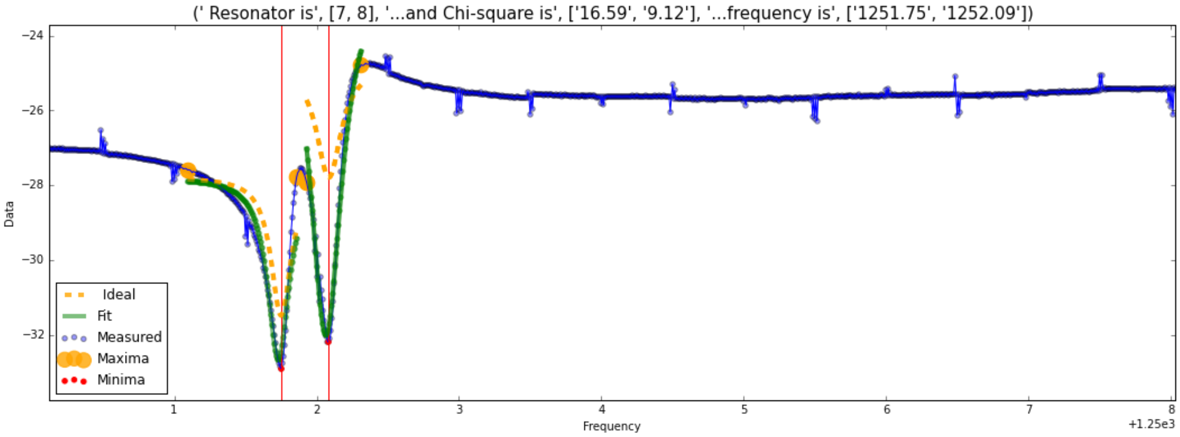
```
Enter : 1)See Resonance close up 2)See Resonance close-up w/ Loren
tzian only ( This is usually enough ) 3)See Resonance close-up, an
d then see Resonance close-up Lorentzian 2
The number of fits with Chi_square more than 100 is 2
The number of fits with Chi_square less than 1 is 7
Resonator       Chi_square      at these Frequencies
1               3.20781440324   1220.9765625
2               1.97168993574   1221.25
3               0.14749572427   1226.171875
4               65.3081690897   1227.46875
5               11.1658294688   1236.9609375
6               77.3980218632   1248.1484375
7               16.5930376968   1251.75
8               9.11888516869   1252.0859375
9               60.4072291613   1263.9765625
10              173.512355778   1268.703125
11              37.326653699    1275.3984375
12              21.5959235042   1283.171875
13              0.586312173621  1288.8203125
14              178.122720952   1291.328125
15              0.475124614128  1297.7265625
16              2.07146184792   1304.0390625
17              62.4713465476   1305.1015625
18              8.09197532611   1310.0625
19              96.6829821473   1318.4921875
20              1.98255774937   1346.328125
21              13.3925015828   1352.125
22              6.17114624302   1356.1640625
23              0.821559112864  1359.65625
24              1.69045310215   1368.8828125
25              23.8088802121   1372.59375
26              0.221762536531  1385.734375
27              17.799291693    1387.3984375
28              1.92953767497   1389.4453125
29              0.364718901825  1402.1328125
30              5.3092631849    1402.328125
31              0.180412784934  1418.140625
32              2.44229591983   1426.3515625
33              11.9452269703   1434.9453125
34              1.66069380506   1437.9453125
35              16.8936605692   1444.921875
36              26.1684723843   1452.6015625

//anaconda/lib/python2.7/site-packages/matplotlib/pyplot.py:424: R
untimeWarning: More than 20 figures have been opened. Figures crea
ted through the pyplot interface (`matplotlib.pyplot.figure`) are
retained until explicitly closed and may consume too much memory.
(To control this warning, see the rcParam `figure.max_open_warnin
g`).
  max_open_warning, RuntimeWarning)
```

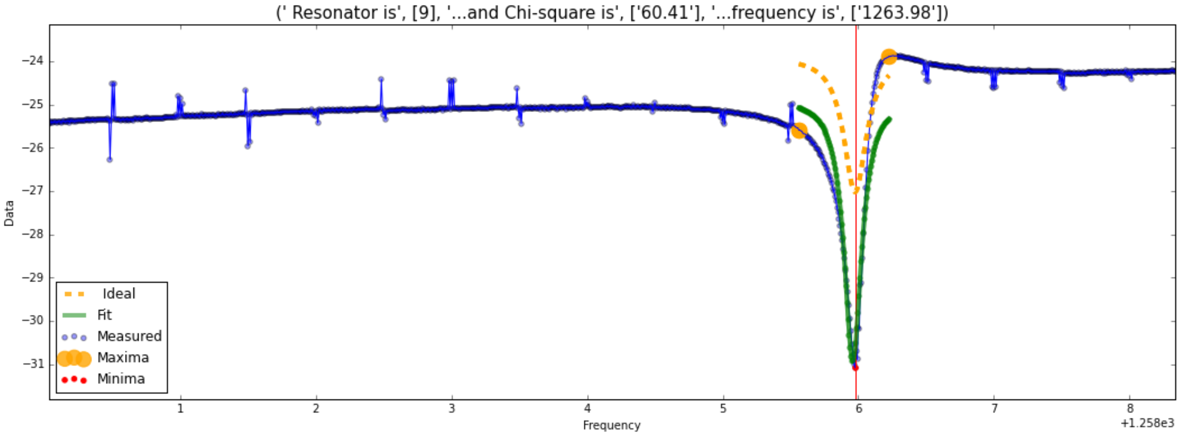

(' Resonator is', [1, 2], '...and Chi-square is', ['3.21', '1.97'], '...frequency is', ['1220.98', '1221.25'])

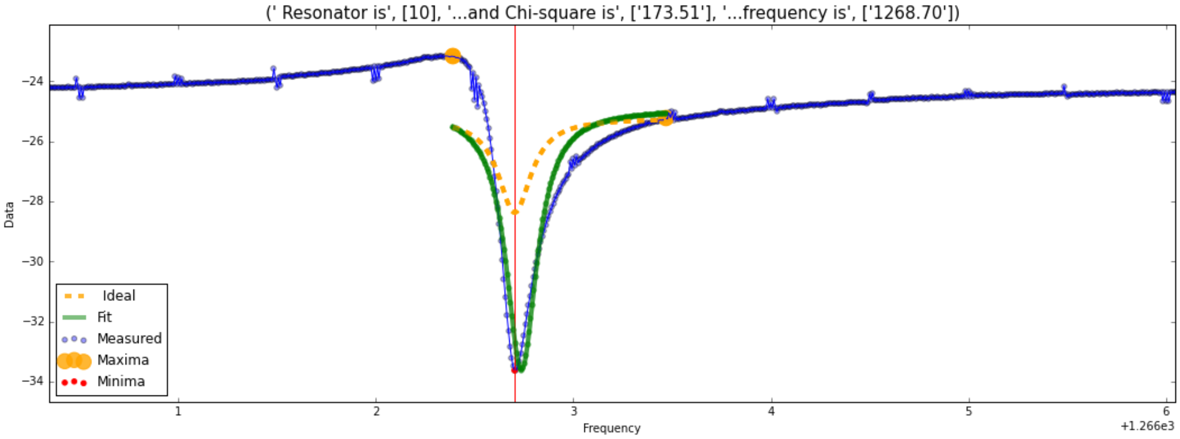

(' Resonator is', [3, 4], '...and Chi-square is', ['0.15', '65.31'], '...frequency is', ['1226.17', '1227.47'])

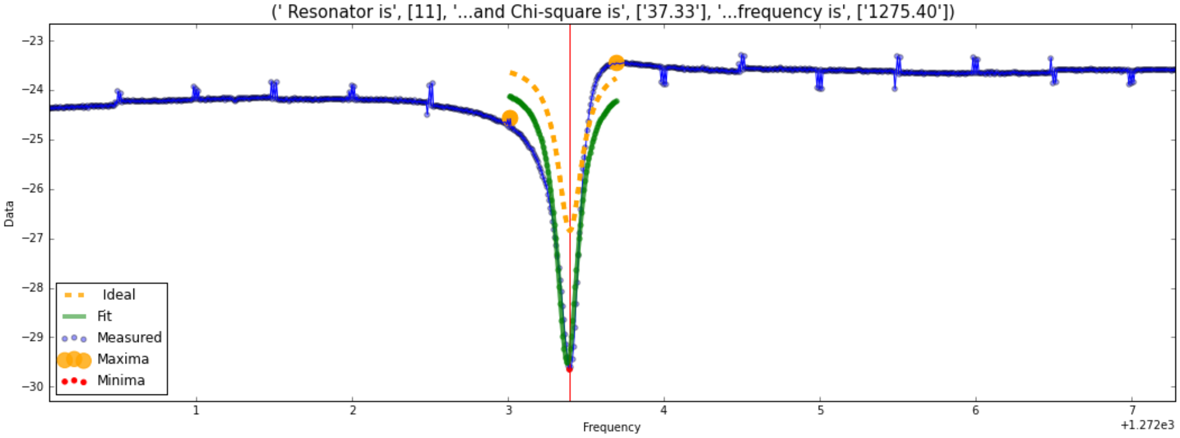

(' Resonator is', [5], '...and Chi-square is', ['11.17'], '...frequency is', ['1236.96'])

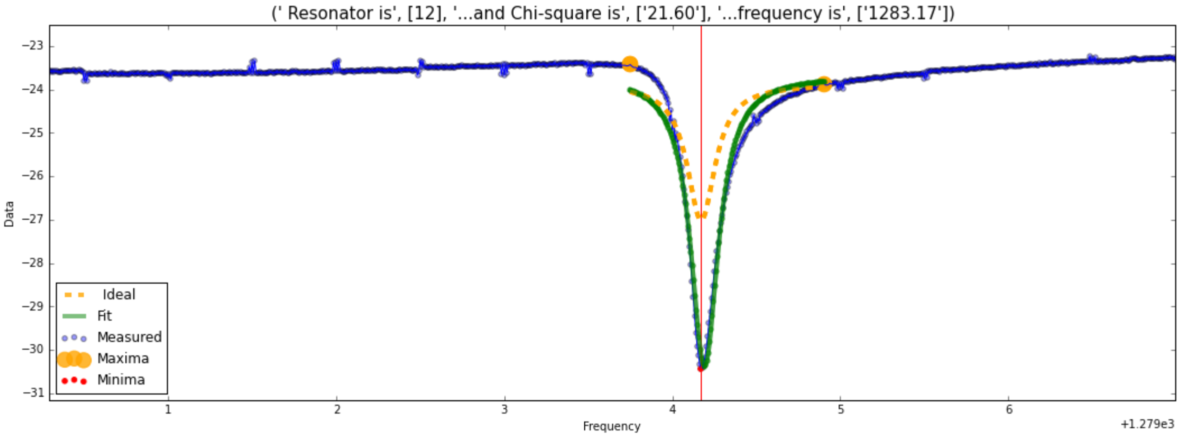

(' Resonator is', [6], '...and Chi-square is', ['77.40'], '...frequency is', ['1248.15'])

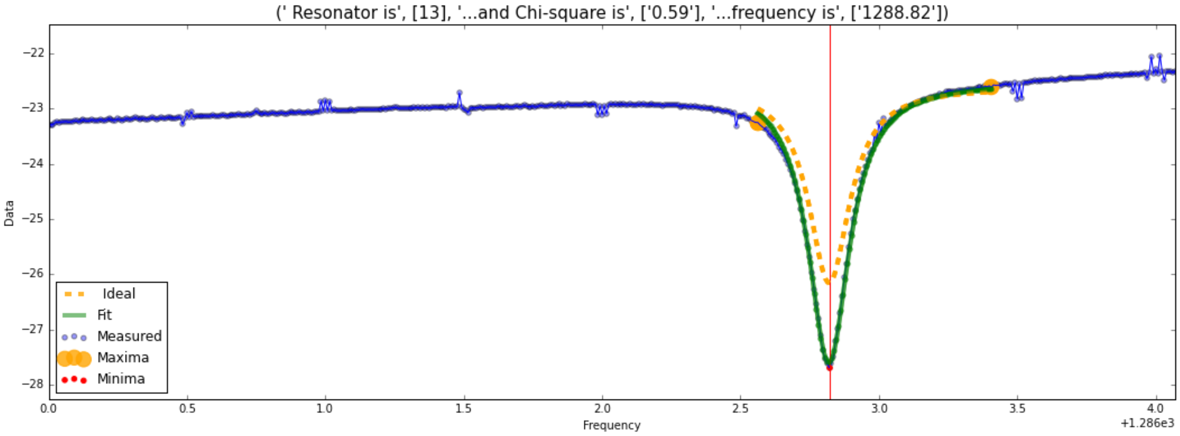

(' Resonator is', [7, 8], '...and Chi-square is', ['16.59', '9.12'], '...frequency is', ['1251.75', '1252.09'])
Data
−24
−26
−28
−30
−32
Frequency
1
2
3
4
5
6
7
8
+1.25e3
Ideal
Fit
Measured
Maxima
Minima

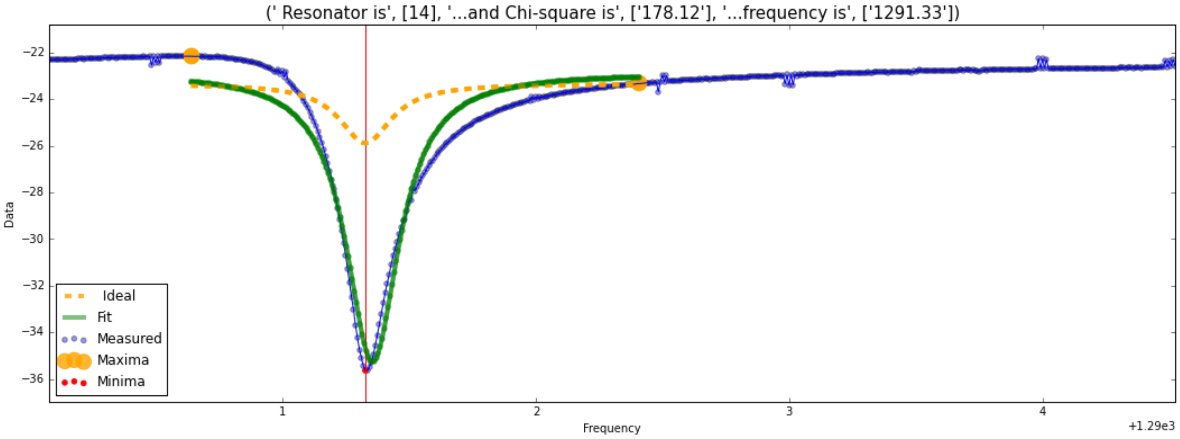

(' Resonator is', [9], '...and Chi-square is', ['60.41'], '...frequency is', ['1263.98'])
Data
−24
−25
−26
−27
−28
−29
−30
−31
Frequency
1
2
3
4
5
6
7
8
+1.258e3
Ideal
Fit
Measured
Maxima
Minima

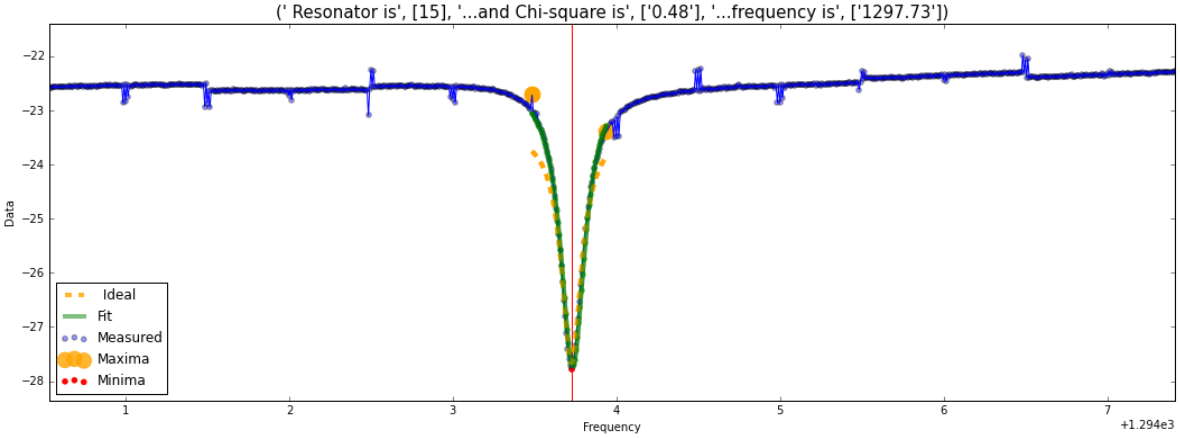

(' Resonator is', [10], '...and Chi-square is', ['173.51'], '...frequency is', ['1268.70'])
Data
−24
−26
−28
−30
−32
−34
Frequency
1
2
3
4
5
6
+1.266e3
Ideal
Fit
Measured
Maxima
Minima

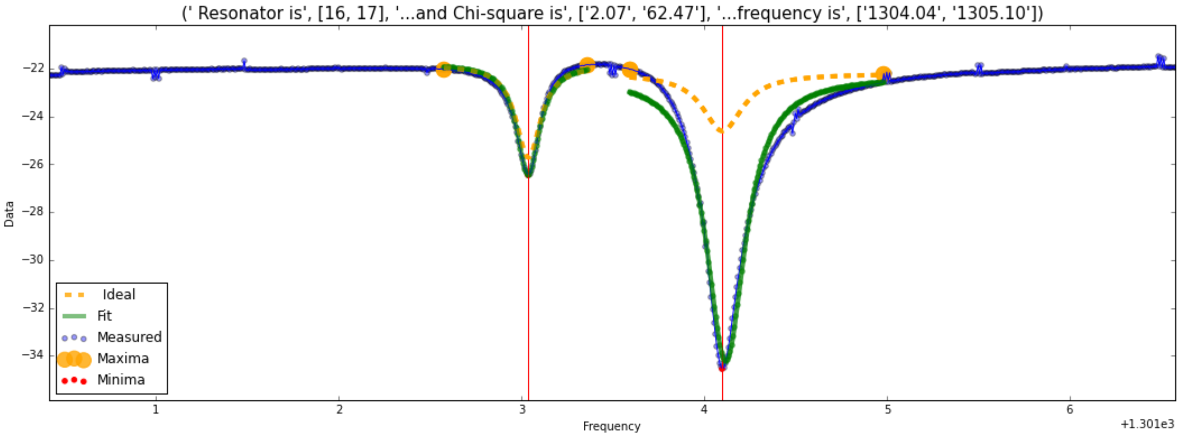

(' Resonator is', [11], '...and Chi-square is', ['37.33'], '...frequency is', ['1275.40'])
Data
−23
−24
−25
−26
−27
−28
−29
−30
Frequency
1
2
3
4
5
6
7
+1.272e3
Ideal
Fit
Measured
Maxima
Minima

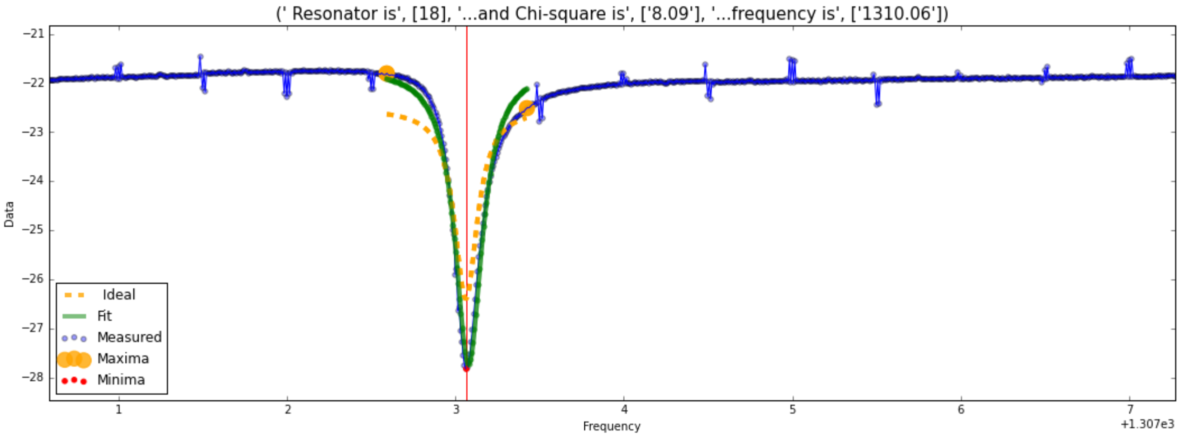

(' Resonator is', [12], '...and Chi-square is', ['21.60'], '...frequency is', ['1283.17'])
Data
−23
−24
−25
−26
−27
−28
−29
−30
−31
Frequency
1
2
3
4
5
6
+1.279e3
Ideal
Fit
Measured
Maxima
Minima

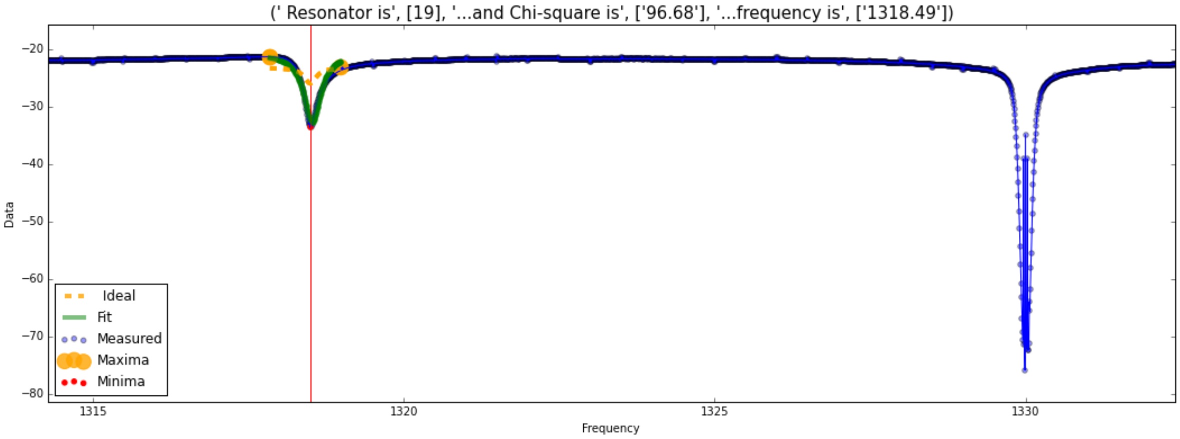

(' Resonator is', [13], '...and Chi-square is', ['0.59'], '...frequency is', ['1288.82'])
Data
−22
−23
−24
−25
−26
−27
−28
Frequency
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
+1.286e3
Ideal
Fit
Measured
Maxima
Minima

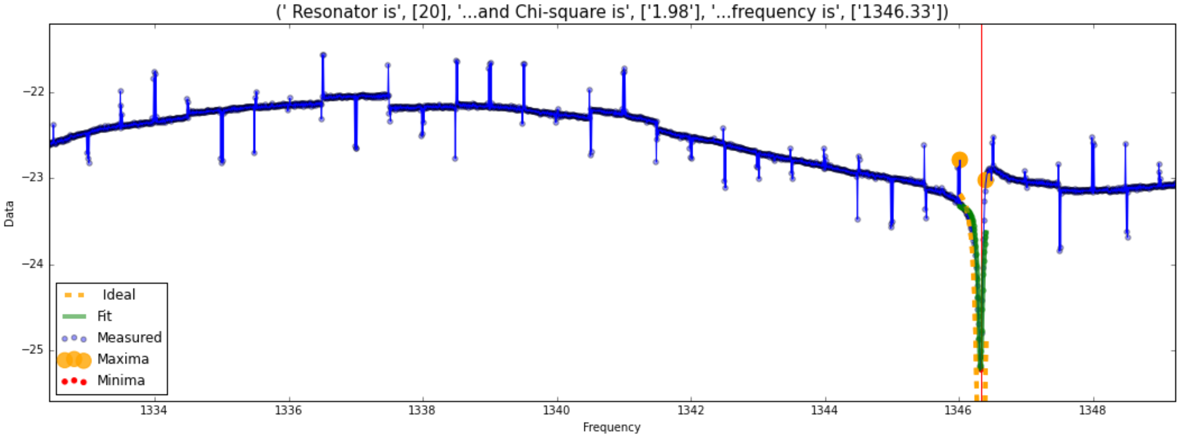

(' Resonator is', [14], '...and Chi-square is', ['178.12'], '...frequency is', ['1291.33'])
Data
−22
−24
−26
−28
−30
−32
−34
−36
Frequency
1
2
3
4
+1.29e3
Ideal
Fit
Measured
Maxima
Minima

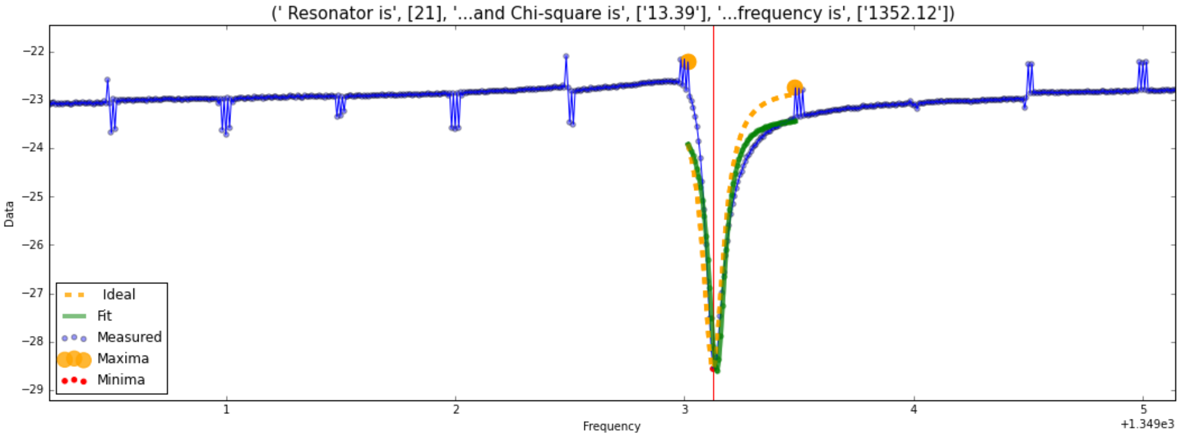

(' Resonator is', [15], '...and Chi-square is', ['0.48'], '...frequency is', ['1297.73'])
Data
−22
−23
−24
−25
−26
−27
−28
Frequency
1
2
3
4
5
6
7
+1.294e3
Ideal
Fit
Measured
Maxima
Minima

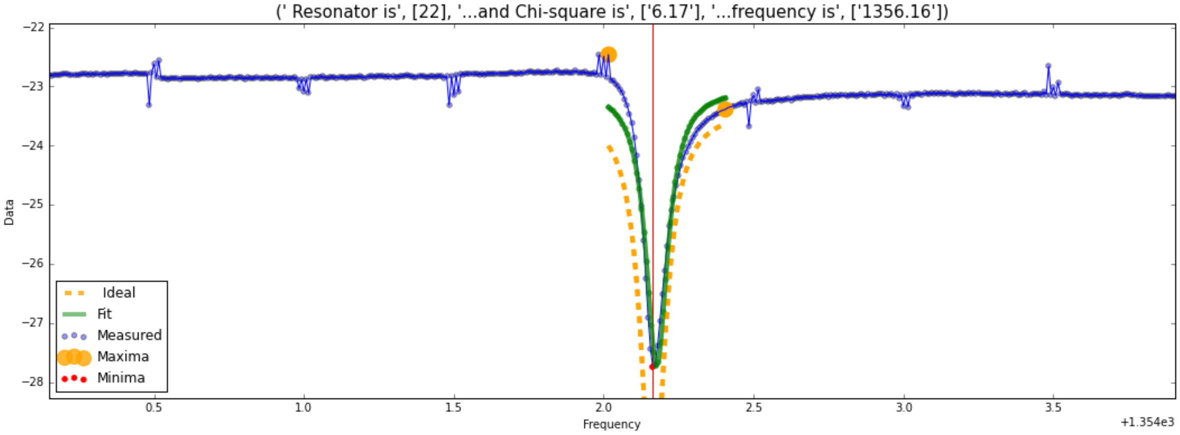

(' Resonator is', [16, 17], '...and Chi-square is', ['2.07', '62.47'], '...frequency is', ['1304.04', '1305.10'])
Data
−22
−24
−26
−28
−30
−32
−34
Frequency
1
2
3
4
5
6
+1.301e3
Ideal
Fit
Measured
Maxima
Minima

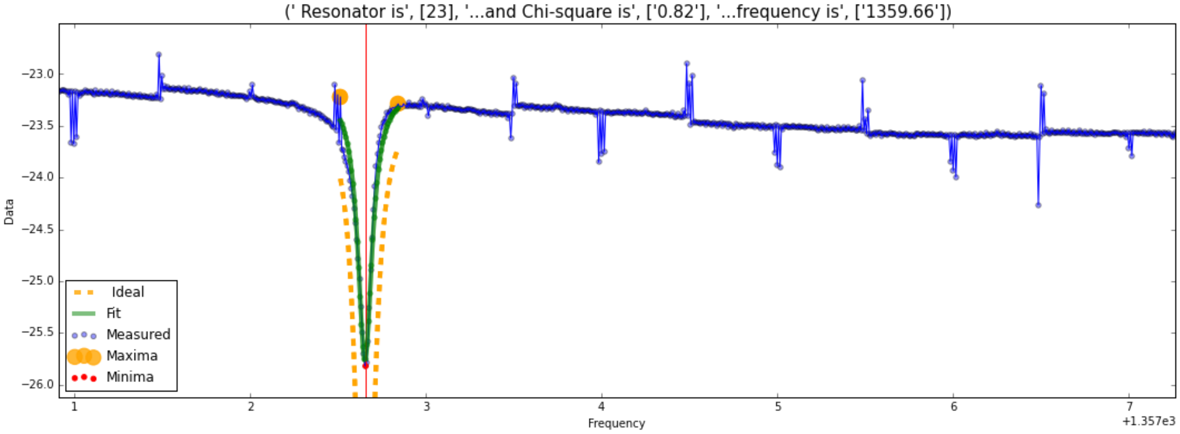

(' Resonator is', [18], '...and Chi-square is', ['8.09'], '...frequency is', ['1310.06'])
Data
−21
−22
−23
−24
−25
−26
−27
−28
Frequency
1
2
3
4
5
6
7
+1.307e3
Ideal
Fit
Measured
Maxima
Minima

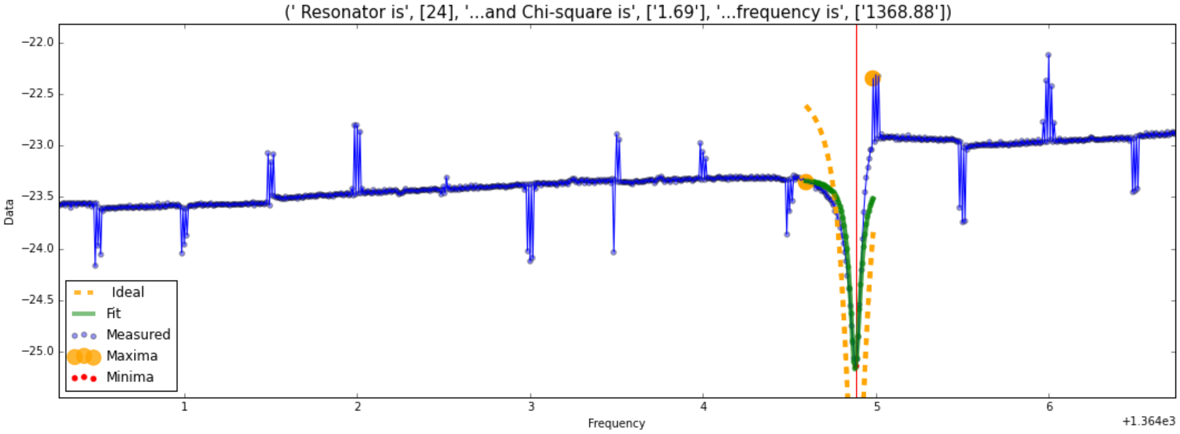

(' Resonator is', [19], '...and Chi-square is', ['96.68'], '...frequency is', ['1318.49'])
Data
−20
−30
−40
−50
−60
−70
−80
Frequency
1315
1320
1325
1330
Ideal
Fit
Measured
Maxima
Minima

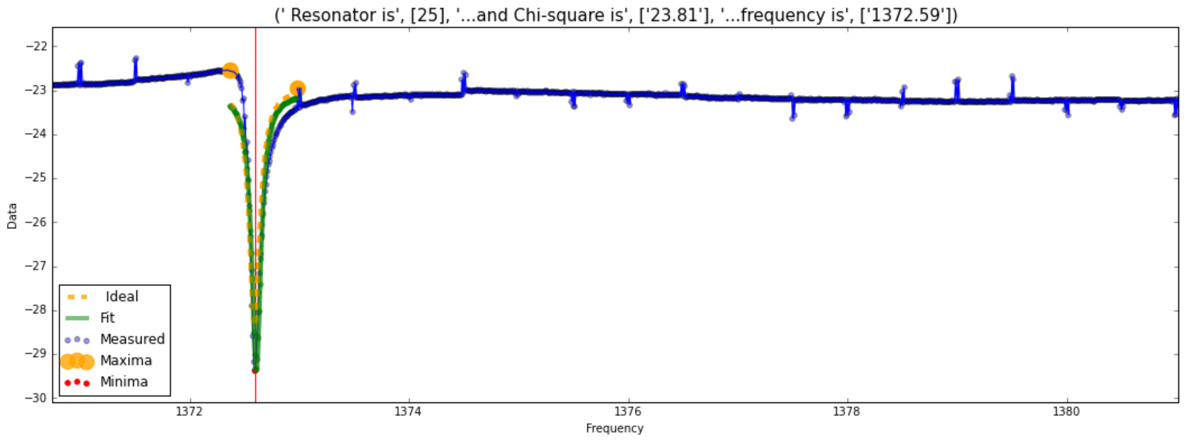

(' Resonator is', [20], '...and Chi-square is', ['1.98'], '...frequency is', ['1346.33'])
Data
−22
−23
−24
−25
Frequency
1334
1336
1338
1340
1342
1344
1346
1348
Ideal
Fit
Measured
Maxima
Minima

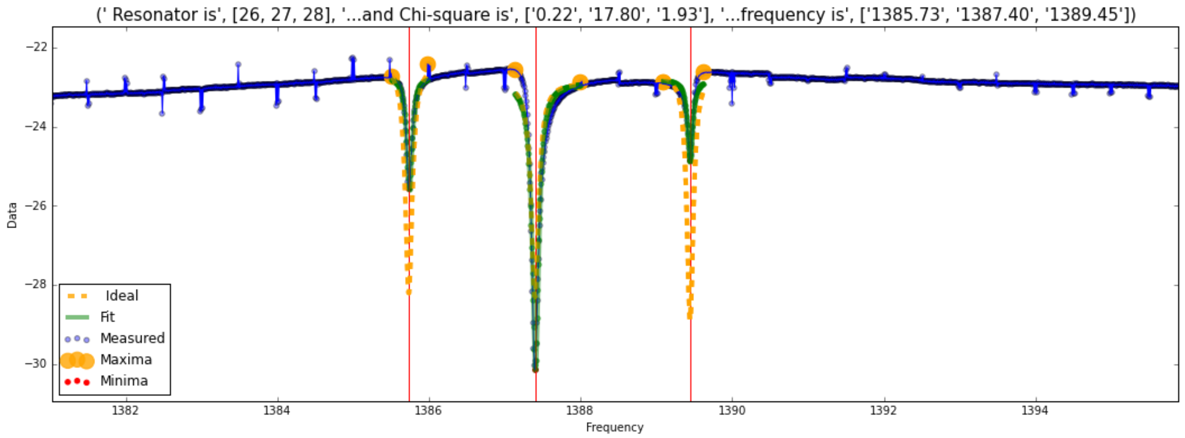

(' Resonator is', [21], '...and Chi-square is', ['13.39'], '...frequency is', ['1352.12'])

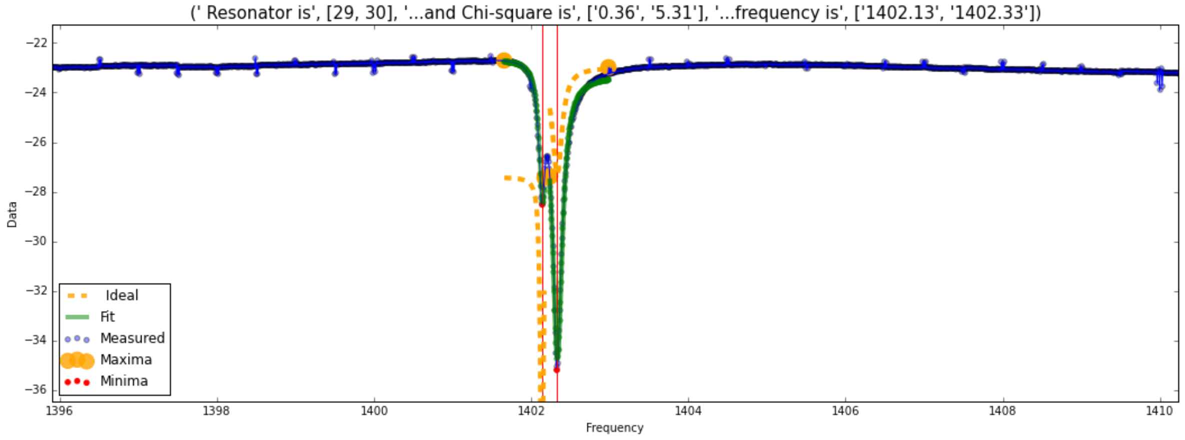

(' Resonator is', [22], '...and Chi-square is', ['6.17'], '...frequency is', ['1356.16'])

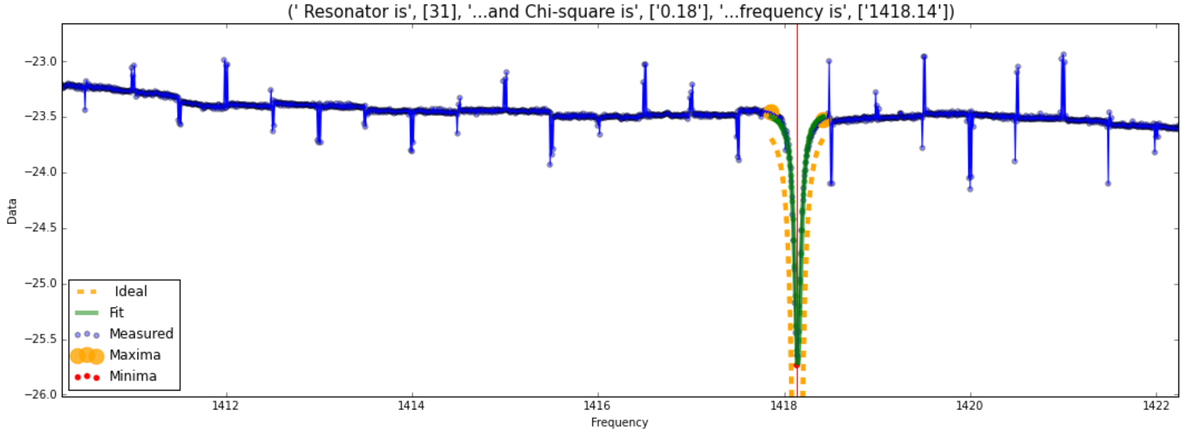

(' Resonator is', [23], '...and Chi-square is', ['0.82'], '...frequency is', ['1359.66'])

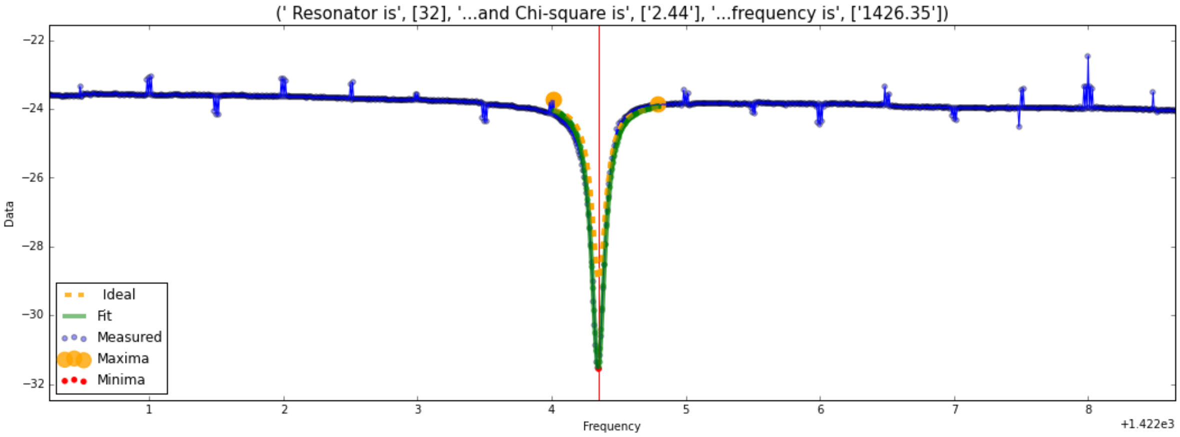

(' Resonator is', [24], '...and Chi-square is', ['1.69'], '...frequency is', ['1368.88'])

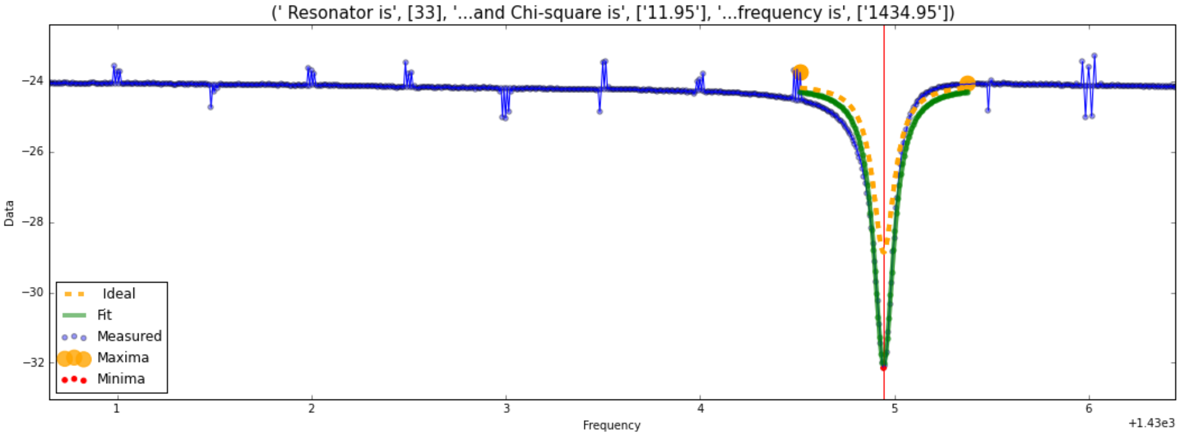

(' Resonator is', [25], '...and Chi-square is', ['23.81'], '...frequency is', ['1372.59'])

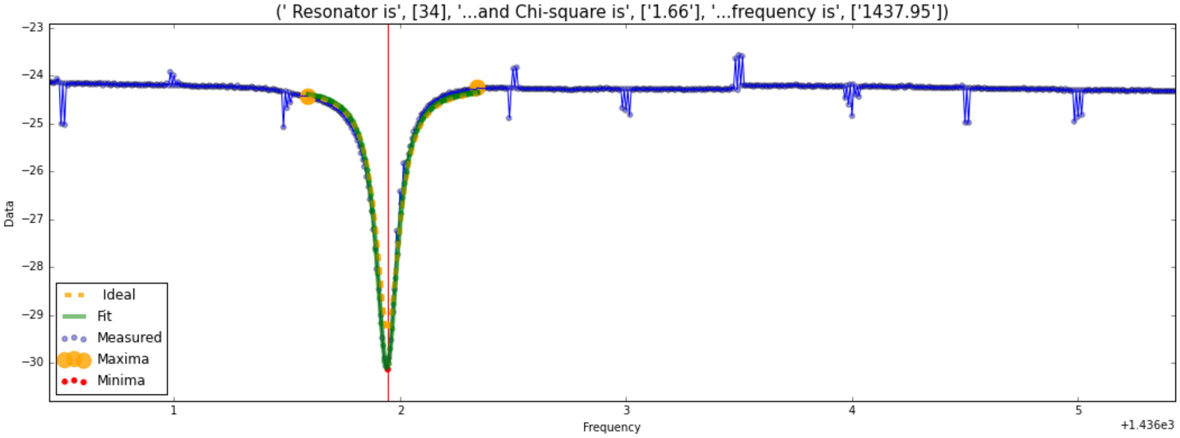

(' Resonator is', [26, 27, 28], '...and Chi-square is', ['0.22', '17.80', '1.93'], '...frequency is', ['1385.73', '1387.40', '1389.45'])

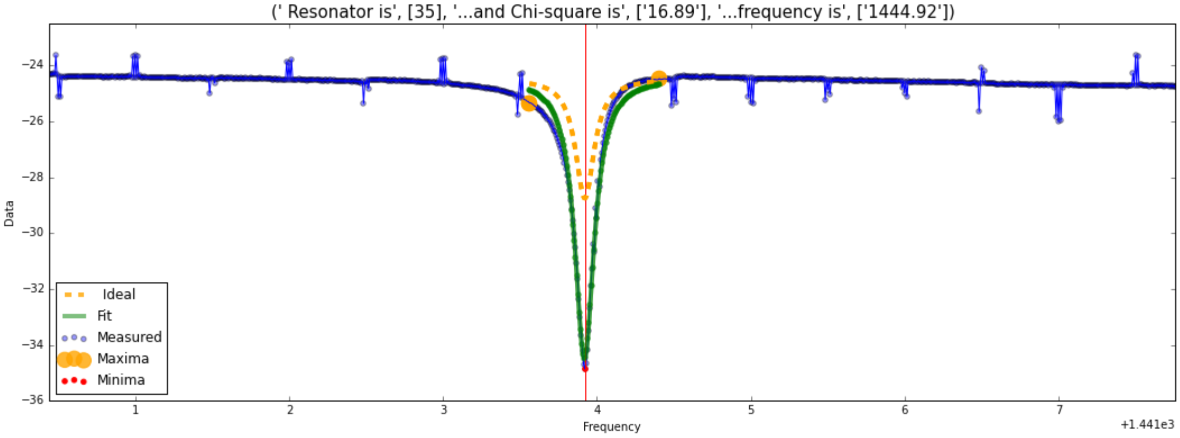

(' Resonator is', [29, 30], '...and Chi-square is', ['0.36', '5.31'], '...frequency is', ['1402.13', '1402.33'])

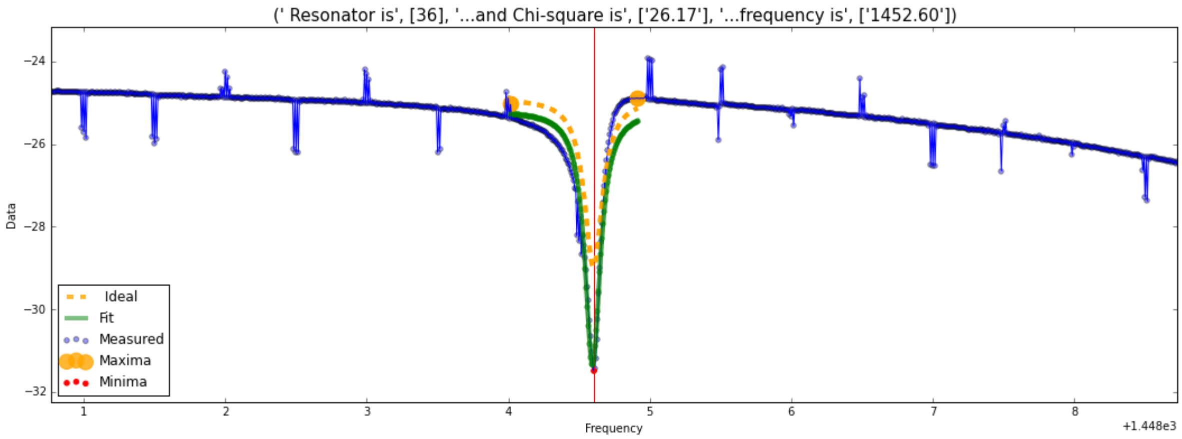

(' Resonator is', [31], '...and Chi-square is', ['0.18'], '...frequency is', ['1418.14'])


(' Resonator is', [32], '...and Chi-square is', ['2.44'], '...frequency is', ['1426.35'])


(' Resonator is', [33], '...and Chi-square is', ['11.95'], '...frequency is', ['1434.95'])


(' Resonator is', [34], '...and Chi-square is', ['1.66'], '...frequency is', ['1437.95'])


(' Resonator is', [35], '...and Chi-square is', ['16.89'], '...frequency is', ['1444.92'])


(' Resonator is', [36], '...and Chi-square is', ['26.17'], '...frequency is', ['1452.60'])
−24
−26
−28
−30
−32
Data
1
2
3
4
5
6
7
8
+1.448e3
Frequency
Ideal
Fit
Measured
Maxima
Minima

```
<matplotlib.figure.Figure at 0x111bab250>
```

In [ ]: