For_Git_FM318_Fitting

1 Fitting

1.1 Metallic Region

1.1.1 Resitivity

[ ]:
```
"10.1126/sciadv.1500797"

import sys
np.set_printoptions(threshold=1000) # can change to sys.maxsize, default is 1000


fig = plt.figure(figsize=(12,12))
ax = fig.add_subplot(111)

CCLXXX = int(closest_element_index(temperature,280)[0])
CC = int(closest_element_index(temperature,165)[0])

#print(CCLXXX)
#print(CC)


for i,data in enumerate(pathlist_RT_film_cleaned):



    if i==0:

        dataextracted = dataextractorRT(data)
        temperature = dataextracted[0]
        resistance2pt = dataextracted[1]
        resitivity2pt = resistance2pt *22.86E-9*(np.pi/np.log(2))*1E2*1E6
        resistance4pt = dataextracted[3]
        resitivity4pt = resistance4pt *22.86E-9*(np.pi/np.log(2))*1E2*1E6

        temperature_metallic_region = temperature[CCLXXX:CC]
        resitivity4pt_metallic_region = resitivity4pt[CCLXXX:CC]
```

```
    plt.plot(temperature_metallic_region,resitivity4pt_metallic_region,␣
↪color = 'black')

    ##### Numpy Polynomial Fit 1st Order #####

    a, b = np.polynomial.polynomial.
↪polyfit(temperature_metallic_region,resitivity4pt_metallic_region, 1)
    print('a',a)
    print('b',b)
    fit1 = a + b*temperature_metallic_region
    print('Polynomial Fit 1st Order', np.polynomial.polynomial.
↪Polynomial([a,b]))
    plt.plot(temperature_metallic_region, fit1, linestyle = "--", linewidth␣
↪= 2, color = 'orange', alpha = 1)

    #pd.DataFrame({'a':[a],'b':[b]}).to_csv(r'C:\Users\pblah\Data\Navy␣
↪Beach\FM318\Data\Film\RT\Fitting Parameters\Linear Fit\ ' +
    #'Linear_Fitting_Params' + '.csv')

    ################################

    ##### Numpy Polynomial Fit 2nd Order #####

    c, d, e = np.polynomial.polynomial.
↪polyfit(temperature_metallic_region,resitivity4pt_metallic_region, 2)
    print('c',c)
    print('d',d)
    print('e',e)
    fit2 = c + d*temperature_metallic_region␣
↪+e*(temperature_metallic_region)**2
    print('Polynomial Fit 2nd Order', np.polynomial.polynomial.
↪Polynomial([c,d,e]))
    plt.plot(temperature_metallic_region, fit2, linestyle = "--", linewidth␣
↪= 2, color = 'red', alpha = 1)

    ################################


    #plt.title("FM318 Film RT",fontsize = 30)
    #plt.ylabel(r'$Ln(d\rho_{xx}/dT)$',fontsize =30, labelpad = 20)
    #plt.xlabel("$Ln(T)$(K)",fontsize =30,labelpad = 20)
    #plt.yticks(fontsize=20)
    #plt.xticks(fontsize=20)
    #plt.ylim(-5,5)
```

```
        #slope = np.gradient(np.log(gradient),np.
↪log(temperature_metallic_region))

        #n = slope + 1

#plt.plot(np.log(temperature_metallic_region),n)

#plt.ylim(-0.25E6,0.25E6)

#plt.title("n")
plt.show()

#temperature_insulating_region = temperature[CC::]
```

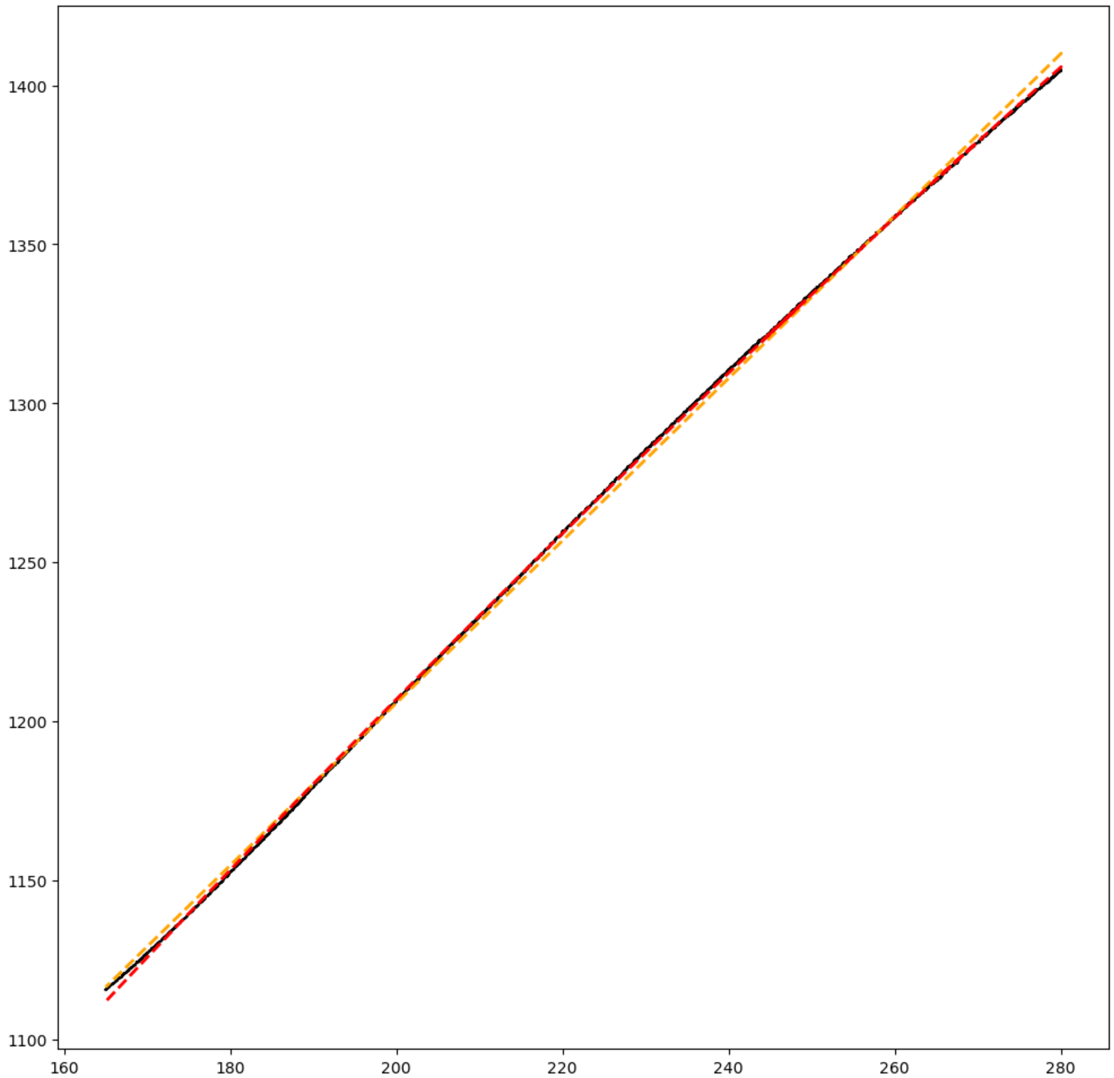
[10]:

```
"10.1126/sciadv.1500797"

import sys
np.set_printoptions(threshold=1000) # can change to sys.maxsize, default is 1000


fig = plt.figure(figsize=(12,12))
ax = fig.add_subplot(111)

CCLXXX = int(closest_element_index(temperature,280)[0])
CC = int(closest_element_index(temperature,165)[0])

#print(CCLXXX)
#print(CC)


for i,data in enumerate(pathlist_RT_film_cleaned):



    if i==0:

        dataextracted = dataextractorRT(data)
        temperature = dataextracted[0]
        resistance2pt = dataextracted[1]
        resitivity2pt = resistance2pt *22.86E-9*(np.pi/np.log(2))*1E2*1E6
        resistance4pt = dataextracted[3]
        resitivity4pt = resistance4pt *22.86E-9*(np.pi/np.log(2))*1E2*1E6

        temperature_metallic_region = temperature[CCLXXX:CC]
        resitivity4pt_metallic_region = resitivity4pt[CCLXXX:CC]
```

```
    plt.plot(temperature_metallic_region,resitivity4pt_metallic_region,␣
↪color = 'black')

    ##### Numpy Polynomial Fit 1st Order #####

    a, b = np.polynomial.polynomial.
↪polyfit(temperature_metallic_region,resitivity4pt_metallic_region, 1)
    print('a',a)
    print('b',b)
    fit1 = a + b*temperature_metallic_region
    print('Polynomial Fit 1st Order', np.polynomial.polynomial.
↪Polynomial([a,b]))
    plt.plot(temperature_metallic_region, fit1, linestyle = "--", linewidth␣
↪= 2, color = 'orange', alpha = 1)

    #pd.DataFrame({'a':[a],'b':[b]}).to_csv(r'C:\Users\pblah\Data\Navy␣
↪Beach\FM318\Data\Film\RT\Fitting Parameters\Linear Fit\ ' +
    #'Linear_Fitting_Params' + '.csv')

    ################################

    ##### Numpy Polynomial Fit 2nd Order #####

    c, d, e = np.polynomial.polynomial.
↪polyfit(temperature_metallic_region,resitivity4pt_metallic_region, 2)
    print('c',c)
    print('d',d)
    print('e',e)
    fit2 = c + d*temperature_metallic_region␣
↪+e*(temperature_metallic_region)**2
    print('Polynomial Fit 2nd Order', np.polynomial.polynomial.
↪Polynomial([c,d,e]))
    plt.plot(temperature_metallic_region, fit2, linestyle = "--", linewidth␣
↪= 2, color = 'red', alpha = 1)

    ################################


    #plt.title("FM318 Film RT",fontsize = 30)
    #plt.ylabel(r'$Ln(d\rho_{xx}/dT)$',fontsize =30, labelpad = 20)
    #plt.xlabel("$Ln(T)$(K)",fontsize =30,labelpad = 20)
    #plt.yticks(fontsize=20)
    #plt.xticks(fontsize=20)
    #plt.ylim(-5,5)
```

```
        #slope = np.gradient(np.log(gradient),np.
↪log(temperature_metallic_region))

        #n = slope + 1

#plt.plot(np.log(temperature_metallic_region),n)

#plt.ylim(-0.25E6,0.25E6)

#plt.title("n")
plt.show()

#temperature_insulating_region = temperature[CC::]
```

```
a 694.8247220834903
b 2.5543103090996135
Polynomial Fit 1st Order 694.82472208 + 2.55431031 x
c 596.6698058411239
d 3.4545215631937554
e -0.0020171478450257355
Polynomial Fit 2nd Order 596.66980584 + 3.45452156 x - 0.00201715 x**2
```



1.1.2 Conductance

[19]:
```
"10.1126/sciadv.1500797"

import sys
np.set_printoptions(threshold=1000) # can change to sys.maxsize, default is 1000


fig = plt.figure(figsize=(12,12))
ax = fig.add_subplot(111)
```

```
for i,data in enumerate(pathlist_RT_film_cleaned):



    if i==0:



        dataextracted = dataextractorRT(data)
        temperature = dataextracted[0]
        resistance2pt = dataextracted[1]
        resitivity2pt = resistance2pt *22.86E-9*(np.pi/np.log(2))*1E2*1E6
        resistance4pt = dataextracted[3]
        resitivity4pt = resistance4pt *22.86E-9*(np.pi/np.log(2))*1E2*1E6

        CCLXXX = int(closest_element_index(temperature,280)[0])
        CC = int(closest_element_index(temperature,165)[0])

        print(CCLXXX)
        print(CC)

        temperature_metallic_region = temperature[CCLXXX:CC]
        resitivity4pt_metallic_region = resitivity4pt[CCLXXX:CC]
        print(temperature_metallic_region)
        print(resitivity4pt_metallic_region)

        conductivity4pt_metallic_region = 1/(resitivity4pt[CCLXXX:CC])



        plt.plot(temperature_metallic_region,conductivity4pt_metallic_region,␣
↪color = 'black')

        ##### Numpy Polynomial Fit 1st Order #####

        a, b = np.polynomial.polynomial.
↪polyfit(temperature_metallic_region,conductivity4pt_metallic_region, 1)
        print('a',a)
        print('b',b)
        fit1 = a + b*temperature_metallic_region
        print('fit1',fit1)
        print('Polynomial Fit 1st Order', np.polynomial.polynomial.
↪Polynomial([a,b]))
        plt.plot(temperature_metallic_region, fit1, linestyle = "--", linewidth␣
↪= 2, color = 'orange', alpha = 1)
```

```
        pd.DataFrame({'a':[a],'b':[b]}).to_csv(r'C:\Users\pblah\Data\Navy␣
↪Beach\FM318\Data\Film\RT\Fitting Parameters\Linear Fit\ ' +
        'Linear_Fitting_Params' + '.csv')

        ################################

        ##### Numpy Polynomial Fit 2nd Order #####

        c, d, e = np.polynomial.polynomial.
↪polyfit(temperature_metallic_region,conductivity4pt_metallic_region, 2)
        print('c',c)
        print('d',d)
        print('e',e)
        fit2 = c + d*temperature_metallic_region␣
↪+e*(temperature_metallic_region)**2
        print('Polynomial Fit 2nd Order', np.polynomial.polynomial.
↪Polynomial([c,d,e]))
        plt.plot(temperature_metallic_region, fit2, linestyle = "--", linewidth␣
↪= 2, color = 'red', alpha = 1)

        ################################


        plt.title("FM318 Film Conductance",fontsize = 30)
        plt.ylabel(r'$\frac{1}{\rho}$',fontsize =30, labelpad = 20)
        plt.xlabel("T(K)",fontsize =30,labelpad = 20)
        #plt.yticks(fontsize=20)
        #plt.xticks(fontsize=20)
        #plt.ylim(-5,5)



        #slope = np.gradient(np.log(gradient),np.
↪log(temperature_metallic_region))

        #n = slope + 1

#plt.plot(np.log(temperature_metallic_region),n)

#plt.ylim(-0.25E6,0.25E6)

#plt.title("n")
plt.show()

#temperature_insulating_region = temperature[CC::]
```

```
683
6147
[280.   280.02 279.97 ... 165.03 165.06 165.02]
[1404.75871954 1404.67462063 1404.77740522 ... 1115.58966108 1115.4243912
 1115.49717228]
a 0.0011545801534937569
b -1.6143926292792337e-06
fit1 [0.00070255 0.00070252 0.0007026  ... 0.00088816 0.00088811 0.00088817]
Polynomial Fit 1st Order 0.00115458 - (1.61439263e-06) x
c 0.0013739214185099656
d -3.626044035178049e-06
e 4.5076067199749296e-09
Polynomial Fit 2nd Order 0.00137392 - (3.62604404e-06) x + (4.50760672e-09) x**2
```