

Guide

November 19, 2021

1

-

,

.

-

,

.

1.0.1

```
[1]: import matplotlib
from matplotlib import pyplot as plt
import numpy as np
import pandas as pd
from scipy.optimize import minimize

# 2 1.pgf.
# , ,
plt.savefig('data/1.pgf')
plt.show()

matplotlib.use("pgf")
matplotlib.rcParams.update({
    "pgf.texsystem": "pdflatex",
    'font.family': 'serif',
    'text.usetex': True,
    'pgf.rcfonts': False,
})
```

<Figure size 432x288 with 0 Axes>

tex pgfplots

\usepackage{pgfplots}

1.0.2

csv

.

data,

.

csv

excel,

.

```
[2]: # , ( )
def read_files(files):
    ret = []
    for f in files:
        ret += [pd.read_csv(f)]
    return ret
```

```
[3]: geig, spectrum = read_files(['data/tab1.csv', 'data/spectrum.csv'])
```

1.0.3

5.4.1,

$$N(x) = \frac{A}{1 + e^{\alpha(x-x_0)}}$$

3 : A, α, x_0 .

```
[4]: # par[0] = A
# par[1] = alpha
# par[2] = x0
def sigmoid(par, x):
    return par[0]/(1 + np.exp(par[1]*(x-par[2])))
```

(), ,

```
[5]: def fit(f, params, x, y):
    """
    f - , .
    params - , ,

    x, y - ,
    """
    if len(x) != len(y):
        raise " , "
    def err(par, x_, y_):
        y1 = f(par, x_)
        return np.sum((y1-y_)**2)

    return minimize(err, params, args=(x, y)).x
```

, , ()

```
[6]: geig.head()
```

```
[6]:      1      N      t
      0 10    14   72
      1   5   439   30
      2   1   411   30
      3   8   242   30
      4   9    42   30
```

$(I; N/t).$, N/t

```
[7]: geig['N1'] = geig['N'] / geig['t']
```

```
[8]: par = np.zeros((3,))
      par = fit(sigmoid, par, geig['l'], geig['N1'])

      print(f'A = {par[0]}, alpha = {par[1]}, x0 = {par[2]}')
```

A = 14.20390254820025, alpha = 2.707506281046367, x0 = 8.115884956009317

```
[9]: # 2 - , cov
      poly, cov = np.polyfit(spectrum['T'], spectrum['A'], 2, cov=True)
```

1.0.4

```
[10]: #
      fig = plt.figure(figsize=(7, 4))

      #
      x = np.linspace(0, 50, num=100)
      #
      plt.plot(x, sigmoid(par, x), 'k', linewidth=1, label='N approximation')

      #
      delta_N = geig['N'] / geig['t']**2

      #
      plt.errorbar(geig['l'], geig['N1'], fmt='k.', yerr=delta_N, xerr=0,
                  ↪linewidth=1, label='measurments')

      #
      #plt.plot(geig['l'], geig['N1'], 'k.', linewidth=1, label='measurments')

      #
```

```
plt.grid(linestyle='--')
plt.xlabel('$l, \text{mm}$', fontsize=15)
plt.ylabel('$N$', fontsize=15)
# fig.legend()

#
plt.savefig('data/geig.pgf')
plt.show()
```

/tmp/ipykernel_35240/953711285.py:26: UserWarning: Matplotlib is currently using pgf, which is a non-GUI backend, so cannot show the figure.

```
plt.show()
```

```
[11]: plt.errorbar(spectrum['T'], spectrum['A'], xerr=10, yerr=0, fmt='.',
↳ linewidth=1)
x = np.linspace(1850, 2650, num=100)
plt.plot(x, np.polyval(poly, x), linewidth=1)

plt.grid(linestyle='--')
plt.xlabel('$\theta, ^\circ$')
plt.ylabel('$\lambda, \text{\AA}$')

plt.savefig('data/spectrum.pgf')
plt.show()
```

```
/tmp/ipykernel_35240/1272136587.py:10: UserWarning: Matplotlib is currently
using pgf, which is a non-GUI backend, so cannot show the figure.
plt.show()
```

```
tex
:
\begin{figure}[h!]
  \centering
  \begin{floatrow}
    \input{data/geig.pgf}
    \caption{$\alpha$-}
  \end{floatrow}
\end{figure}
```

1.0.5

CSV , tex

```
[12]: # float_format
print(geig.to_latex(index=False, float_format='%.2f'))
```

```
\begin{tabular}{rrrr}
\toprule
l & N & t & N1 \\
\midrule
10 & 14 & 72 & 0.19 \\
5 & 439 & 30 & 14.63 \end{tabular}
```

```

1 & 411 & 30 & 13.70 \\
8 & 242 & 30 & 8.07 \\
9 & 42 & 30 & 1.40 \\
40 & 12 & 113 & 0.11 \\
3 & 419 & 30 & 13.97 \\
7 & 418 & 30 & 13.93 \\
15 & 18 & 107 & 0.17 \\
20 & 18 & 107 & 0.17 \\
25 & 22 & 93 & 0.24 \\
\bottomrule
\end{tabular}

```

,

-

```

[13]: def latex_tab(df):
        tab = df.to_latex(index=False, float_format='%.2f')
        tab = tab.replace('\\\\\\n', '\\\\ \\hline\\n')
        tab = tab.replace('\\toprule', '\\hline')
        tab = tab.replace('\\midrule\\n', '')
        tab = tab.replace('\\bottomrule\\n', '')
        return tab

```

```

[14]: print(latex_tab(geig))

```

```

\begin{tabular}{rrrr}
\hline
1 & N & t & N1 \\ \hline
10 & 14 & 72 & 0.19 \\ \hline
5 & 439 & 30 & 14.63 \\ \hline
1 & 411 & 30 & 13.70 \\ \hline
8 & 242 & 30 & 8.07 \\ \hline
9 & 42 & 30 & 1.40 \\ \hline
40 & 12 & 113 & 0.11 \\ \hline
3 & 419 & 30 & 13.97 \\ \hline
7 & 418 & 30 & 13.93 \\ \hline
15 & 18 & 107 & 0.17 \\ \hline
20 & 18 & 107 & 0.17 \\ \hline
25 & 22 & 93 & 0.24 \\ \hline
\end{tabular}

```

;)

[]: