The University of Franche-Comté

Master CompuPhys - Fortran practical work 1

Smoothing spectra by least-squares fits

Chiari Even
M1 Compuphys 2022

Supervisor : Kathleen Stehlin

December 16, 2022

Contents

I Introduction

In this practical work we have to fit a could of points by the least-squares method, we have five could of points called $spectre_i.txt$ this files contains the intensity for interval of wave-number.
We want to fit them to get a completely equation for each $spectre_i.txt$ with their parameter and we want to smoothing their curves to "erase" the background of errors.
After we will do a linear regression for two of their parameter in order to get their 0 values.

I.1 Least-squares method

I.2 Definition

This method is called "least-squares" because the purpose is to minimize the sum of squared deviation :

$$E(\{a_k\}) = \sum_{i=1}^{N} W_i(f(\{a_k\}, x_i) - y_i)^2 \tag{1}$$

With W_i is the weight, it can be different for each function that we want to fit.In our case we will take:

$$f(\{a_k\}, x) = \sum_k a_k f_k(x) \tag{2}$$

This equation give us a system of linear equations for $\{a_k\}$.

I.3 Our case

We know that the shape of spectral lines is a Gaussian or a Lorentzian, in our case we take a Lorentzian equation to fit our data and we will smoothing it by the least-squares methods.

$$F(\omega) = \frac{S}{\pi} \frac{\gamma}{(\omega - \omega_m)^2 + \gamma^2} \tag{3}$$

with :

- $S = \int_{-\infty}^{\infty} d\omega F(\omega)$ is the integral density of the spectral line
- ω_m is the wave number of the maximum $F(\omega)$
- γ is the half-width at half-height

We reduced our Lorentzian to a system of linear equation by taking $F^{-1}(\omega)$:

$$F^{-1}(\omega) = \frac{\pi}{S\gamma}\omega^2 - \frac{2\pi\omega_m}{S\gamma}\omega + \frac{\pi(\omega_m^2 + \gamma^2)}{S\gamma} \tag{4}$$

Now we use reduced and centered variable $X = \frac{\omega - \overline{\omega}}{\sigma}$ and we injected it in the previous equation:

$$F^{-1}(\omega) = \frac{2\pi\sigma^2}{S\gamma}X^2 - \frac{2\pi\sigma}{S\gamma}(\overline{\omega} - \omega_m)X + \frac{\pi}{S\gamma}\left[\gamma^2 + (\overline{\omega} - \omega_m)^2\right]^2 \tag{5}$$

with :

- $a_0 = \frac{\pi}{S\gamma}\left[\gamma^2 + (\overline{\omega} - \omega_m)^2\right]^2$
- $a_1 = \frac{2\pi\sigma}{S\gamma}(\overline{\omega} - \omega_m)$
- $a_2 = \frac{2\pi\sigma^2}{S\gamma}$

Now we can use the minimization formula and we get a system of linear equations:

$$\begin{cases} a_2 \sum W_i X_i^2 + a_1 \sum W_i X_i + a_0 \sum W_i & = \sum W_i y_i \\ a_2 \sum W_i X_i^3 + a_1 \sum W_i X_i^2 + a_0 \sum W_i X_i & = \sum W_i y_i X_i \\ a_2 \sum W_i X_i^4 + a_1 \sum W_i X_i^3 + a_0 \sum W_i X_i^2 & = \sum W_i y_i X_i^2 \end{cases} \tag{6}$$

- W_i the weight 1 for the non weigthed case and $(F^i_{exp})^2$ for the weighted case
- $y_i ==> F^{-1}(\omega)$

We dividing by N (the number of data for each spectre) to get the average and putting in matrix form:

$$\begin{pmatrix} \langle 1 \rangle & \langle X \rangle & \langle X^2 \rangle \\ \langle X \rangle & \langle X^2 \rangle & \langle X^3 \rangle \\ \langle X^2 \rangle & \langle X^3 \rangle & \langle X^4 \rangle \end{pmatrix} = \begin{pmatrix} \langle y \rangle \\ \langle yX \rangle \\ \langle yX^2 \rangle \end{pmatrix} \tag{7}$$

Now we can compute a_0, a_1, a_2 :

$$a_0 = \frac{\begin{vmatrix} \langle y \rangle & \langle X \rangle & \langle X^2 \rangle \\ \langle yX \rangle & \langle X^2 \rangle & \langle X^3 \rangle \\ \langle yX^2 \rangle & \langle X^3 \rangle & \langle X^4 \rangle \end{vmatrix}}{\begin{vmatrix} \langle 1 \rangle & \langle X \rangle & \langle X^2 \rangle \\ \langle X \rangle & \langle X^2 \rangle & \langle X^3 \rangle \\ \langle X^2 \rangle & \langle X^3 \rangle & \langle X^4 \rangle \end{vmatrix}} \tag{8}$$

$$a_1 = \frac{\begin{vmatrix} \langle 1 \rangle & \langle y \rangle & \langle X^2 \rangle \\ \langle X \rangle & \langle yX \rangle & \langle X^3 \rangle \\ \langle X^2 \rangle & \langle yX^2 \rangle & \langle X^4 \rangle \end{vmatrix}}{\begin{vmatrix} \langle 1 \rangle & \langle X \rangle & \langle X^2 \rangle \\ \langle X \rangle & \langle X^2 \rangle & \langle X^3 \rangle \\ \langle X^2 \rangle & \langle X^3 \rangle & \langle X^4 \rangle \end{vmatrix}} \tag{9}$$

$$a_2 = \frac{\begin{vmatrix} \langle 1 \rangle & \langle X \rangle & \langle y \rangle \\ \langle X \rangle & \langle X^2 \rangle & \langle yX \rangle \\ \langle X^2 \rangle & \langle X^3 \rangle & \langle yX^2 \rangle \end{vmatrix}}{\begin{vmatrix} \langle 1 \rangle & \langle X \rangle & \langle X^2 \rangle \\ \langle X \rangle & \langle X^2 \rangle & \langle X^3 \rangle \\ \langle X^2 \rangle & \langle X^3 \rangle & \langle X^4 \rangle \end{vmatrix}} \tag{10}$$

For all the coefficients we compute the determinant of the previous matrix precise with $|matrix|$, now we can compute S, γ, ω_m :

$$\omega_m = \overline{\omega} - \sigma \frac{a_1}{2a_2}$$

$$\gamma = \sigma \sqrt{\frac{a_0}{a_2} - \frac{a_1^2}{4a_2^2}}$$

$$S = \frac{\pi \sigma}{\sqrt{a_0 a_2 - \frac{a_1^2}{4}}}$$

II Program

We have implemented that in FORTRAN the code is already commented and well explained but i will summarize the main steps:

In order to fit all our data we will use two FORTRAN files and one PYTHON file.
The first FORTRAN file called "TP1.f90" is the main file the contains two subroutines to perform the non-linear fit and the linear one, he call the first subroutine one times for each files and the second two times for γ, ω_m .

The first subroutine do all the steps for the non-linear fit:

- 1. he read the file $spectre_i.txt$
- 2. he compute the axis y and x
- 3. he call the compute function that calculate the coefficient a_0, a_1, a_2 and the S, γ, ω_m
- 4. he write a file $result_i.txt$

The second do the same thing but for a linear case.

- 1. he read the two vector or γ, ω_m
- 2. he compute the coefficient a and b

The second Fortran file "tp1.f90" is a module to compute several function in the main like :

- $mean$ average function
- std standard deviation function
- $deter_3$ determinant of matrix 3x3
- $deter_2$ determinant of matrix 2x2
- $compute$ compute coefficient a_0, a_1, a_2 and the S, γ, ω_m
- $Lorentz$ Lorentzian function
- str function to switch a integer into a character

And the PYTHON file plot all the graph that we need

III Results

We can run the computation with this line in a terminal "$gfortran\, TP1 - mod.f90\, TP1.f90$

III.1 Non-linear fit

And after the python file and we get for the first spectre:

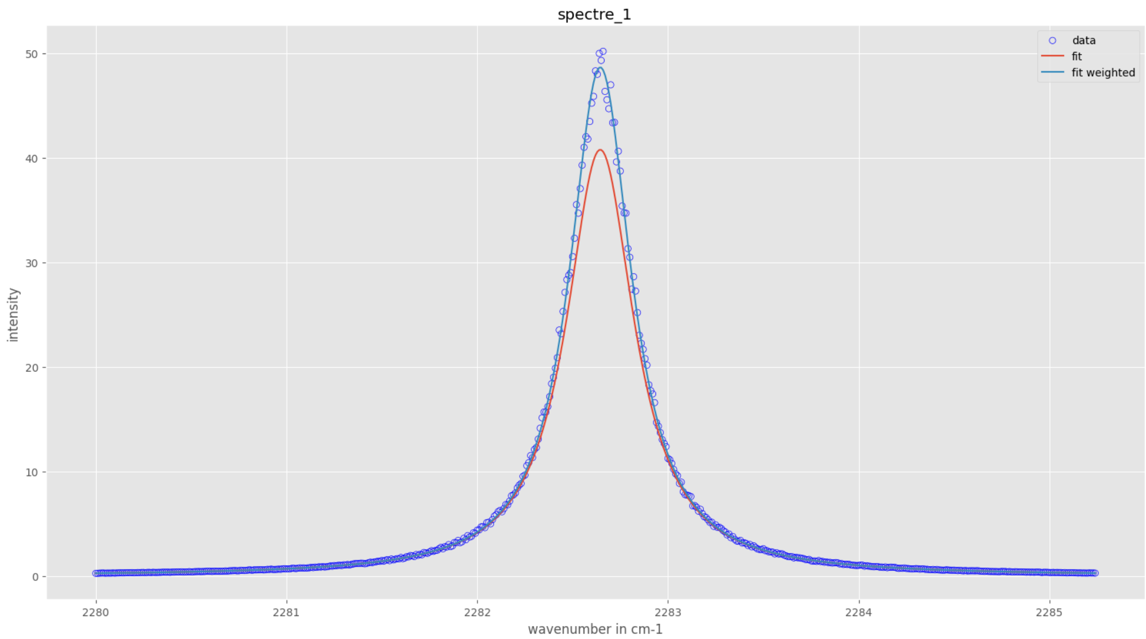

Figure 1: fit of spectre 1

On this graph the blue circle are the could of points, the red line is the non-weighted fit and the blue one is the weighted.
We can see the the non weighted case is less accurate than the weighted one.
The weighted one fit almost perfectly the data.
This gap between the two method is weaker for the other graph because the pressure increase (1,3,6,10,15) atm :

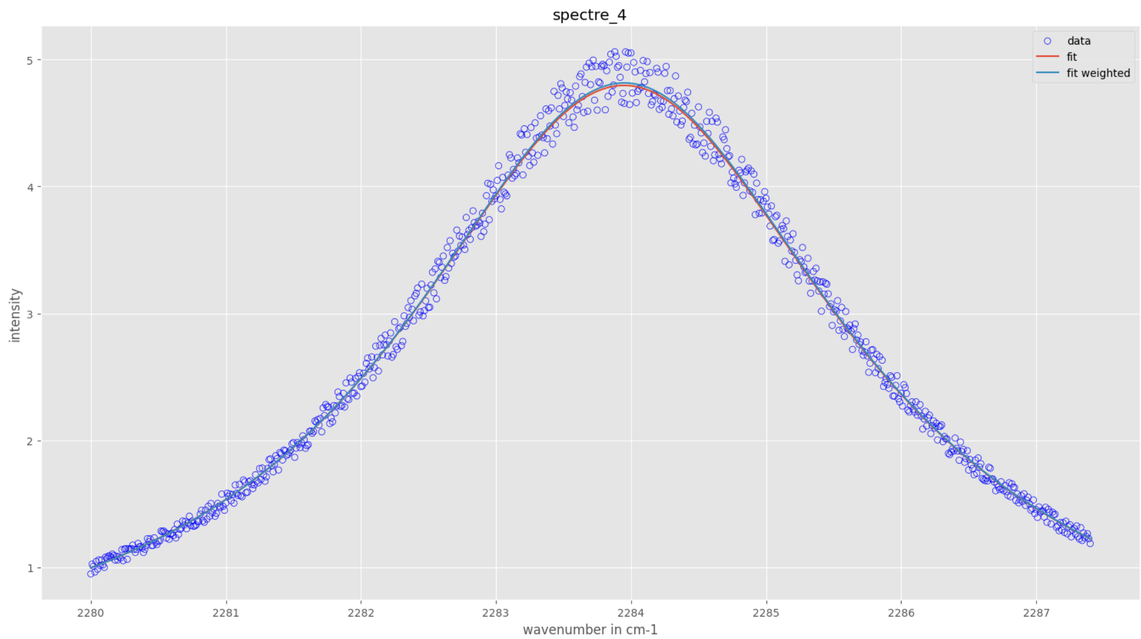

Figure 2: fit of spectre 4

We can see the gap between the weighted and non weighted method is very small and they seems be the same.

III.2 Linear-fit

Now we want to see the linear dependence of ω_m, γ with the respect of :

$$\omega_m = \omega_0 + p\Delta\omega \tag{11}$$

$$\gamma = \gamma_1 p \tag{12}$$

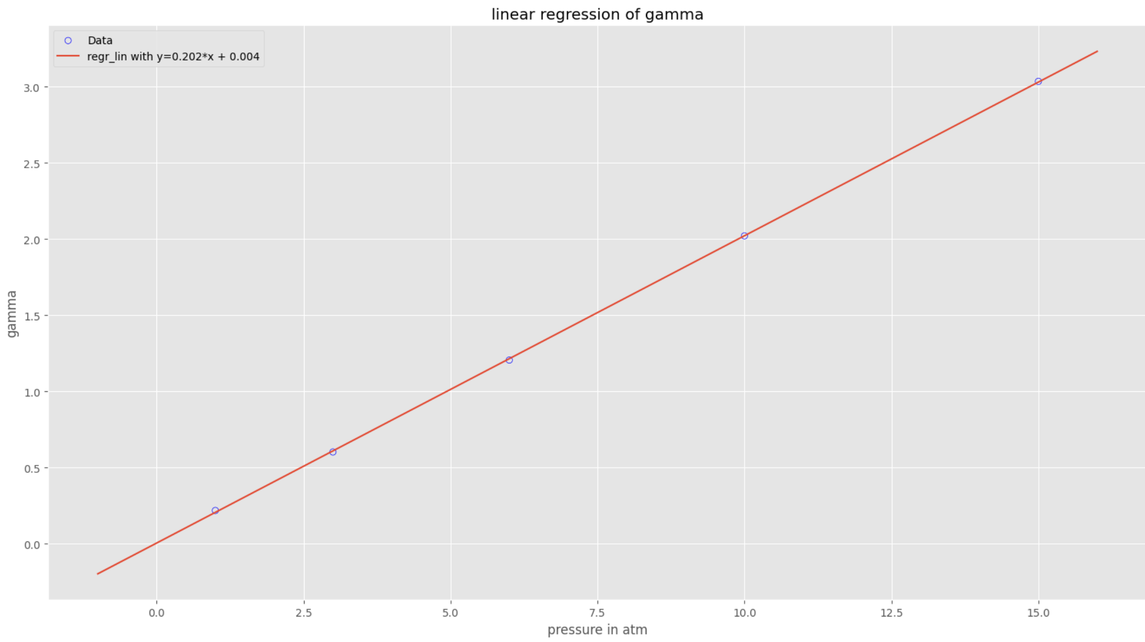

Figure 3: linear regression of γ against the pressure

On this first graph we can see that the data follow a linear line against the pressures and we can read the coefficient b of the linear regression, it is almost 0 like the theory.

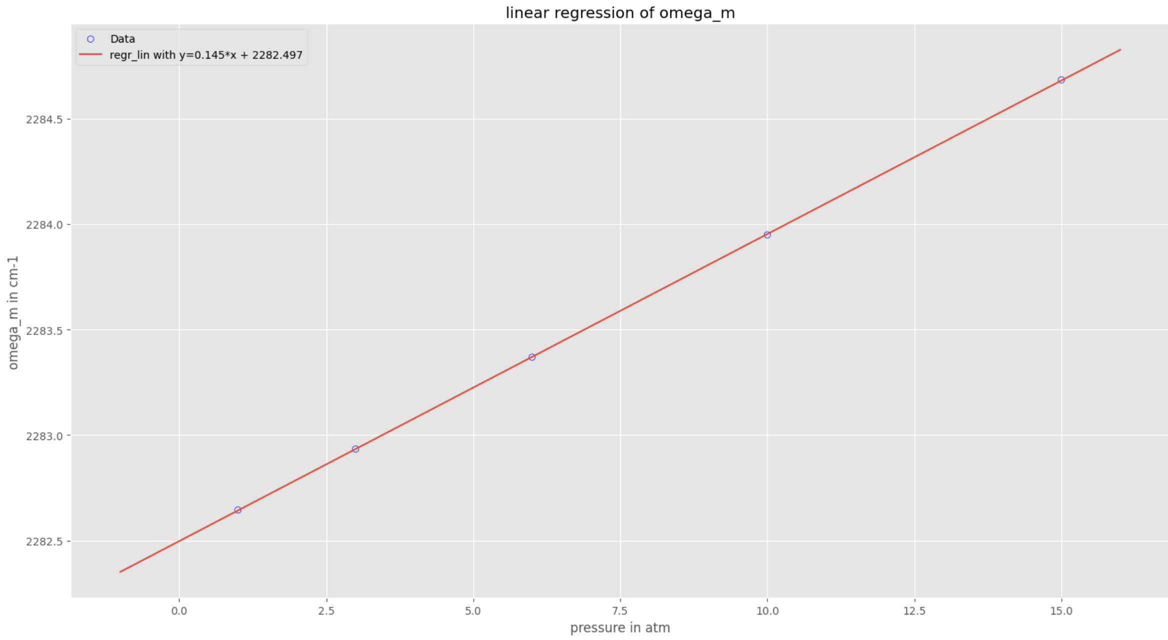

Figure 4: linear regression of ω_m against the pressure

Now we see the same thing on this graph the data follow the linear regression and we can have the $\omega_0 = 2282.47 cm^{-1}$.

IV Conclusion

We can conclude on the consistence on ours fit is very good because we have see how are close the data and the fit.The relation for the linear regression of γ, ω_m is very good we refind the equation in third part and the value of ω_m.