PHY153 (virtual class)
Today: 04/16 (data analysis)
J. Kiryluk

Topics:
- Straight line fitting

This class assignments:
0416Ex1.py (team work, 2 students per team, assigned in today's class)
If you missed today's class, Ex1 is an individual assignment. Email by 04/21 8am.

0416Ex2Ex3Ex4.py (1 python file, 3 exercises: Ex2, Ex3 and Ex4) individual assignments. Email by 04/21 8am.

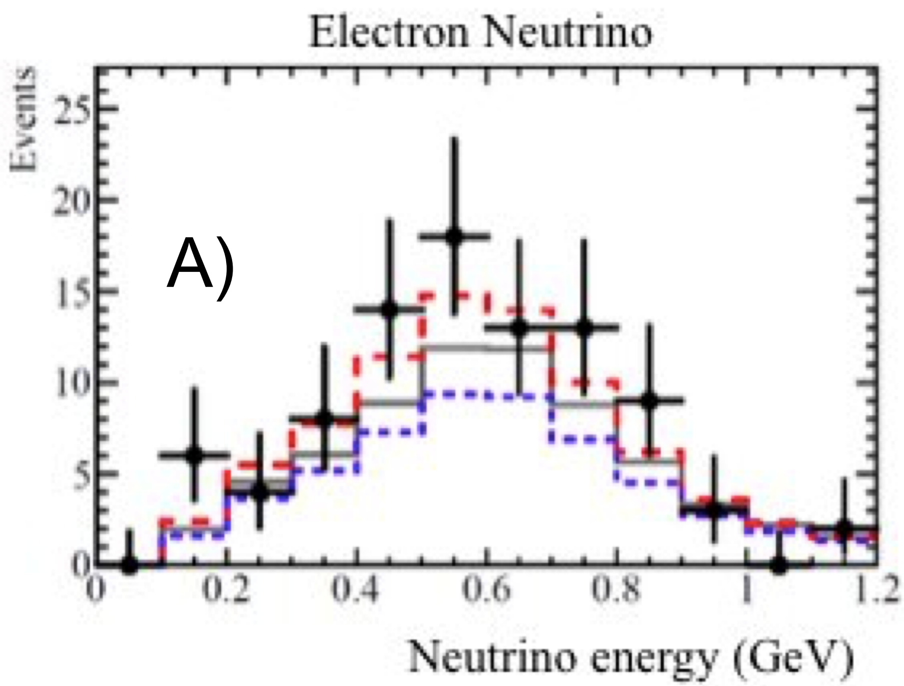
Teams should send one email (email sent to me by student1 with cc to student2)

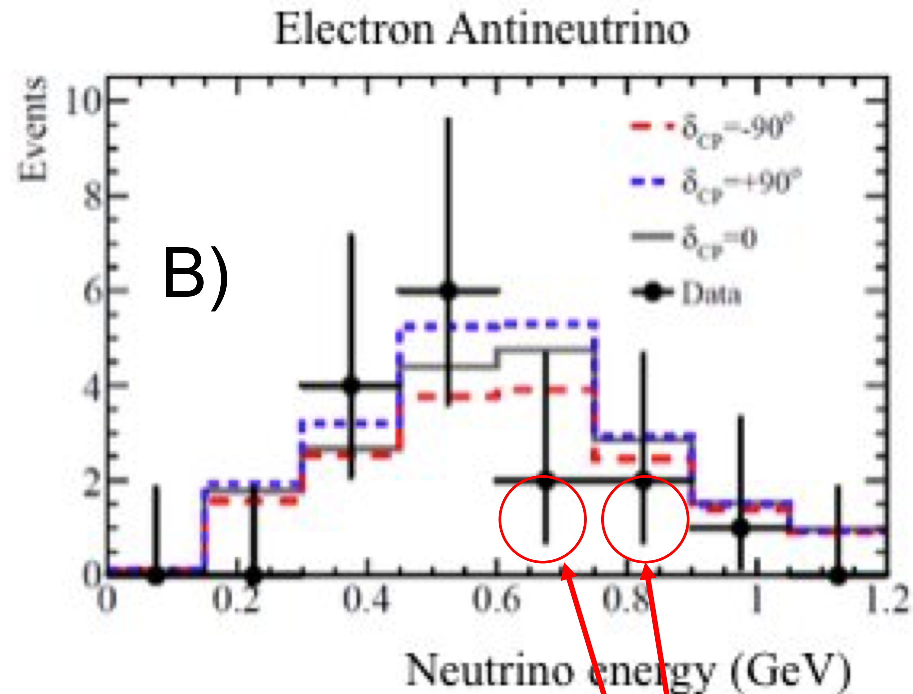
Reminder: office hours FRI 12-1pm and TU 9-10am
zoom id 535-288-6713

0416Ex1.py (HW) Team work (2 or 3 students per group)

T2K Collaboration

https://phys.org/news/2020-04-matter-antimatter-asymmetry-t2k-results-restrict.html





Use methods from 04/14 class and apply them for histograms (x-axis = Energy, y-axis = number of events). Calculate S_m and corresponding p-values (9 S_m values and 9 p-values in total) by comparing the data with 3 theoretical predictions (given by red, blue and gray curves) for results shown in plot:

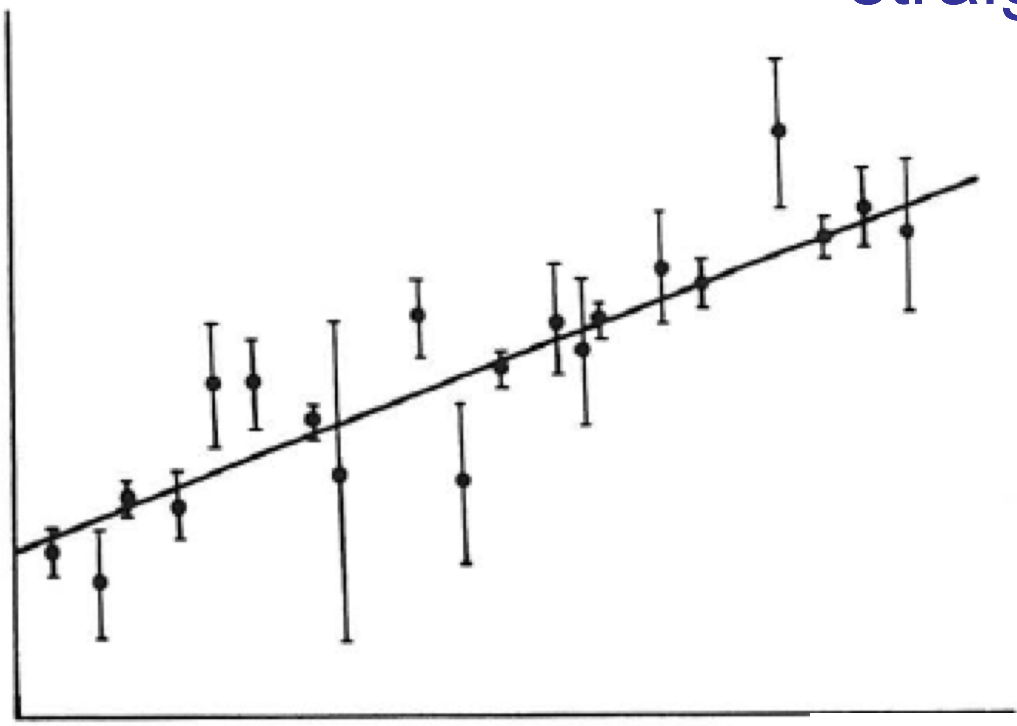
1) A (N=12)
2) B (N=8)
3) combined data sets A and B (N = 20) if it can be done. Justify your answer.

Assume that all data points have uncertainties that are symmetric and take smaller uncertainty values.

Ex1 Example code structure for 1) (plot A, 12 points)

```
#Read data and theory values from plots and enter them (hard coded) in your code as arrays
(or lists). All arrays should have an equal length.

#Experimental data
data_A=np.array([0.,    ….. ])
data_A_err = np.array([2, …. ])  # one uncertainty value per data point
                                  # (choose a smaller value, if uncertainties are not symmetric)


#Theoretical model 1 (red)
Theory_1 = np.array([ ….. ] )
#Theoretical model 2 (blue)
Theory_2 = np.array([ ….. ] )
#theoretical model3 (grey)
Theory_3 = np.array([…..])

# compare data_A with Theory1:  Sm_1, p_value_1
# compare data_A with Theory2:  Sm_2, p_value_2
# compare data_A with Theory3   Sm_3, p_value_3
```

The least squares fitting method: straight line



Straight line:

$g(x) = ax + b$

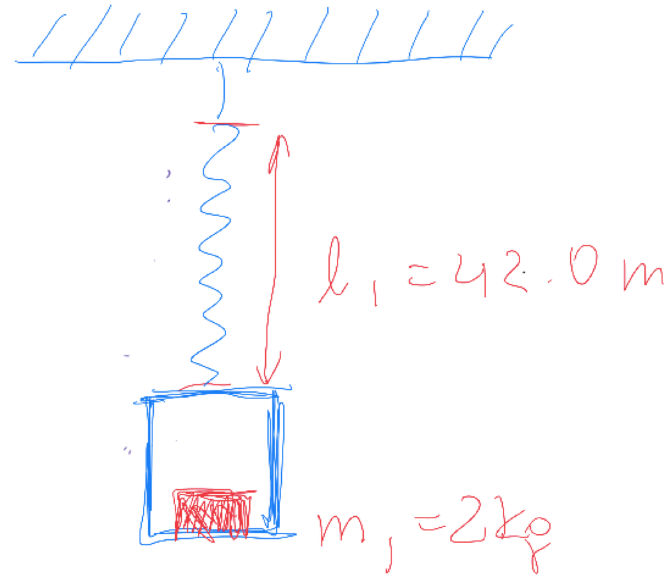
a= slope
b = intercept

- We want a weighted fit, which takes into account uncertainties
- Careful: some plotting tools don't use uncertainties of individual data points (not a weighted fit!)

Hypothesis: data can be described by a straight line. If this is correct, **from the data we want to find best values of parameters a (slope) and b (intercept)**

The least squares fitting method: straight line

Application example: Spring scale (Intro physics)



Used to measure weight (old days ...)
Calibration is done by taking a series of measurements (based on the Hook's law: force is proportional to the extension
$mg = k\Delta l = l - l_0$
l_0 = length of the unloaded spring

$l\,(m) = l_0 + m*(g/k)$

where $g=9.81\ m/s^2$ is a known constant. k and l_0 are both unknown.
l depends on m linearly (from the Hook's law)

How to calibrate this scale? i.e. find k and l_0?
From a series of measurements (l1, m1), (l2, m2), …. (lN,mN)
where m1 < m2 < m3 < … < mN and l1 < l2< l3 < … < mN
we want to find the best value of l_0 and (g/)k based on data from the fit.

The least squares fitting method: straight line

Suppose we measured n points at x_i and got results: $f_i \pm \sigma_{i,f}$
We want to fit a function g to these data g(x;a,b)=ax+b , where a and b are unknown parameters to be determined from the data

The method of least squares (also called as chi-square χ^2 minima-lization) states that the best values of a and b parameters are those for which S_m:

$$S_m = \sum_{i=1}^{n} \left[\frac{f_i - g(x_i; a, b)}{\sigma_{i,f}}\right]^2 \longrightarrow \chi^2$$

If f_i is Gaussian distributed with mean $g(x_{i;a,b})$ and variance $(\sigma_{\iota,f})^2$

reaches a minimum.

Note: this method is general (works beyond straight line)

The least squares fitting method: straight line

Suppose we measured n points at x_i and got results: $f_i \pm \sigma_{i,f}$
We want to fit a function g to these data g(x;a,b)=ax+b , where
a and b are unknown parameters to be determined from the data

The method of least squares (also called as chi-square χ^2 minima-lization) states that the best values of a and b parameters are those for which S_m :

$$S_m = \sum_{i=1}^{n} \left[\frac{f_i - g(x_i; a, b)}{\sigma_{i,f}}\right]^2 \longrightarrow \chi^2$$

If f_i is Gaussian distributed with mear $g(x_{i;a,b})$ and variance $(\sigma_{\iota,f})^2$

is a minimum.

To find a and b one must solve the system of equations

$$\boxed{\begin{aligned} \frac{\partial S_m}{\partial a} &= 0 \\ \frac{\partial S_m}{\partial b} &= 0 \end{aligned}}$$

In general, numerical methods are used to *minimize* S_m.

The least squares fitting method: straight line

$$S_m = \sum_{i=1}^{n} \left[\frac{f_i - ax_i - b}{\sigma_{i,f}}\right]^2$$

Taking partial derivatives:

$$\frac{\partial S_m}{\partial a} = -2\sum \frac{(f_i - ax_i - b)x_i}{\sigma_{i,f}^2} = 0$$

$$\frac{\partial S_m}{\partial b} = -2\sum \frac{(f_i - ax_i - b)}{\sigma_{i,f}^2} = 0$$

2 equations, 2 unknown (a and b)

$$A \equiv \sum \frac{x_i}{\sigma_{i,f}^2} \qquad B \equiv \sum \frac{1}{\sigma_{i,f}^2}$$

$$C \equiv \sum \frac{f_i}{\sigma_{i,f}^2} \qquad D \equiv \sum \frac{x_i^2}{\sigma_{i,f}^2}$$

$$E \equiv \sum \frac{x_i f_i}{\sigma_{i,f}^2} \qquad F \equiv \sum \frac{f_i^2}{\sigma_{i,f}^2}$$

[notation follows: L. Lyons]

Where A through F are determined from the data

The least squares fitting method: straight line

Central values:

$$\boxed{a = \frac{EB - CA}{DB - A^2} \qquad b = \frac{DC - EA}{DB - A^2}}$$

$$A \equiv \sum \frac{x_i}{\sigma_{i,f}^2} \qquad C \equiv \sum \frac{f_i}{\sigma_{i,f}^2} \qquad D \equiv \sum \frac{x_i^2}{\sigma_{i,f}^2}$$

$$B \equiv \sum \frac{1}{\sigma_{i,f}^2} \qquad E \equiv \sum \frac{x_i f_i}{\sigma_{i,f}^2} \qquad F \equiv \sum \frac{f_i^2}{\sigma_{i,f}^2}$$

Note: a, b parameters are correlated.
Correlation and covariance cov(a,b) will be discussed next class

$g(x;a,b)=ax+b$

Covariance: Extra material

What are the errors on a and b?

$$V^{-1} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \qquad A_{11} = \frac{1}{2}\frac{\partial^2 S}{\partial^2 a} = D \qquad A_{22} = \frac{1}{2}\frac{\partial^2 S}{\partial^2 b} = B$$

$$A_{12} = A_{21} = \frac{1}{2}\frac{\partial^2 S}{\partial a \partial b} = A$$

$$V = \frac{1}{A_{11}A_{22} - A_{12}^2}\begin{pmatrix} A_{22} & -A_{12} \\ -A_{12} & A_{11} \end{pmatrix} = \frac{1}{DB - A^2}\begin{pmatrix} B & -A \\ -A & D \end{pmatrix}$$

V = error matrix for correlated parameters
Diagonal elements = variances for a and b
Off-diagonal elements = covariances between a and b

The least squares fitting method: straight line

Central values:

$$a = \frac{EB - CA}{DB - A^2}$$

$$b = \frac{DC - EA}{DB - A^2}$$

Variances:

$$\sigma_a^2 = \frac{B}{DB - A^2}$$

$$\sigma_b^2 = \frac{D}{DB - A^2}$$

$$\mathrm{cov}(a,b) = \frac{-A}{DB - A^2}$$

$$A \equiv \sum \frac{x_i}{\sigma_{i,f}^2} \qquad C \equiv \sum \frac{f_i}{\sigma_{i,f}^2} \qquad D \equiv \sum \frac{x_i^2}{\sigma_{i,f}^2}$$

$$B \equiv \sum \frac{1}{\sigma_{i,f}^2} \qquad E \equiv \sum \frac{x_i f_i}{\sigma_{i,f}^2} \qquad F \equiv \sum \frac{f_i^2}{\sigma_{i,f}^2}$$

Note: a, b parameters are correlated (i.e. not independent)
Correlation and covariance cov(a,b) will be discussed next class

Quality of the straight line fit

For a good "fit", S_m/ndf should be ~ 1 (for a large values of ndf)

- **ndf (number of degree of freedom) = k= N-m,**

m=number of parameters, n= number of data points

for m=2 (straight line fit, 2 parameters), ndf=k=N-2

- **S_m:**

Best fit parameters a and b (output of your program to calculate a and b) are used to calculate S_m (aka Chi2) for the straight line fit:

Eq.1 $$\boxed{S_m = \sum \frac{(f_i - a x_i - b)^2}{\sigma_{i,f}^2}}$$

[here Sm is the minimum value obtained for the best fit parameters a and b uncertainties on a and b are not used in S_m calculation]

For a good **straight line fit,** one should expect **S_m ~ N-2**

Ex2 (in class or HW):

Write python code which:

1) defines function(s) to calculate A, …, F using data points x_i , $f_i \pm \sigma_{i,f}$

2) defines function(s) a and b , their uncertainties and covariance factor using A, …, F.

3) defines a function to calculate S_m (Eq.1) using a,b and data points: $f_i \pm \sigma_{i,f}$ and p-value. Assume data points f_i[], sigma_if[] are known.

Test your code by doing next Ex3 and Ex4.

functions.py

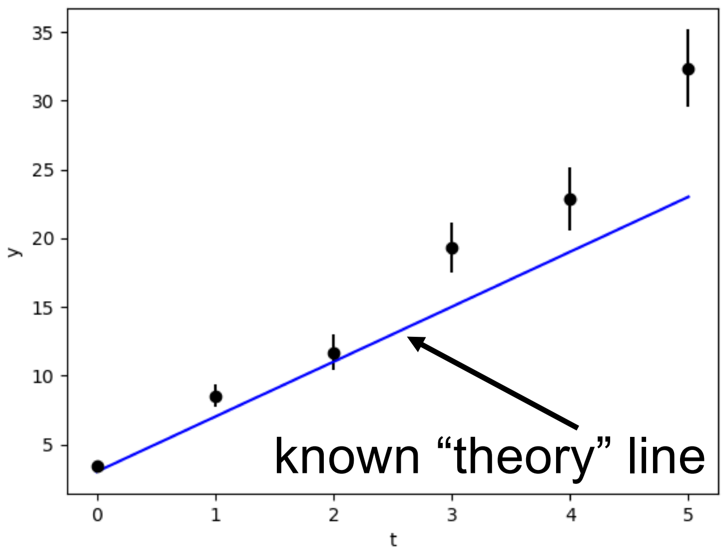
```
from scipy.integrate import quad
#Chi2 function from homework 04/14
 def coeff(x, f, sigma_f)
     #…. Your code
     return al,bl,cl,dl,el,fl


 def fit(A,B,C,D,E,F):
     #your code
     return slope,slope_error,intercept,intercept_error



 def get_sm(x,f,sigma_f,slope,intercept):
     # your code
     return sm

 def get_pvalue(k,sm):
     pval=quad(lambda x,k: myChi2(x,k), sm, np.inf,args=(k))
     return pval[0]
```

Example how to use functions.py in a different code, e.g. 0416Ex3.py

```
import functions
p=functions.get_pvalue(7,20)
```

Example: Past HW problem

We measure two values of y as a function of t:




```
Y1= 3.4 +/- 0.3
Y2= 8.5 +/- 0.8
Y3= 11.7 +/- 1.2
Y4= 19.3 +/- 1.9
Y5= 22.9 +/- 2.3
Y6= 32.4 +/- 3.2
```

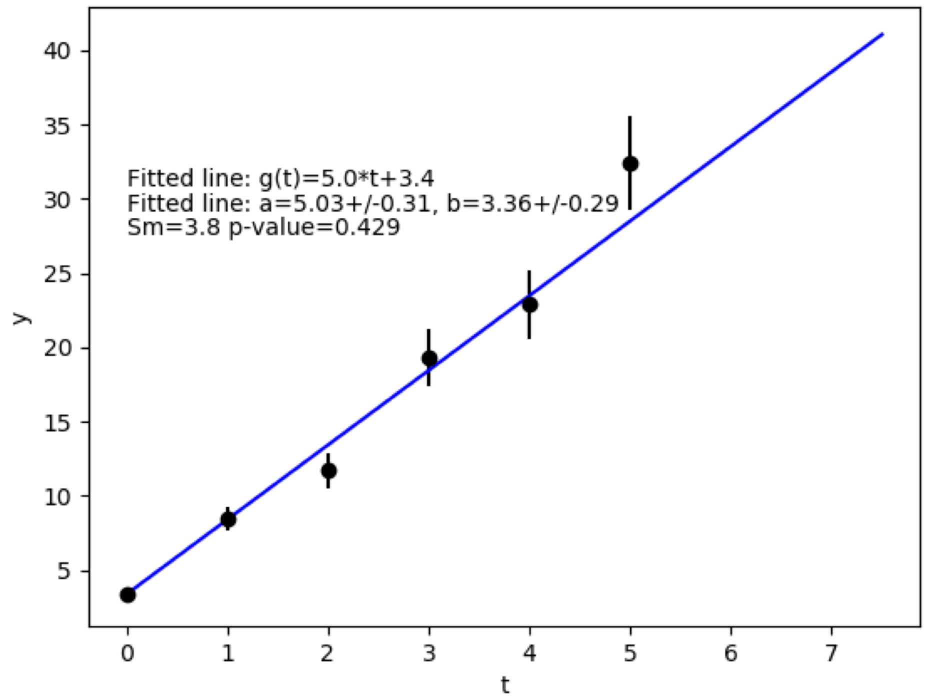
Are data (black points) consistent with (or in other words described by)
g(t) = 3 + 4 *t (shown as a blue line)

Quantify the agreement/disagreement, by calculating S_m (python program), ndf and finding corresponding p-value from "χ^2 Test: p-Value Reference plot".

Large Sm, small p-value.

Let's fit the data (2 parameter fit) instead.

Ex3 (HW): Reproduce fit results.
Write python code to find :
3.1) the straight line fit parameters a and b (and their uncertainties) , S_m , the p-value
3.2) Make a plot, which superimposes
data and the fitted line
3.3) Interpret the results: how good is the fit and why? Give probabilistic interpretation of obtained S_m value

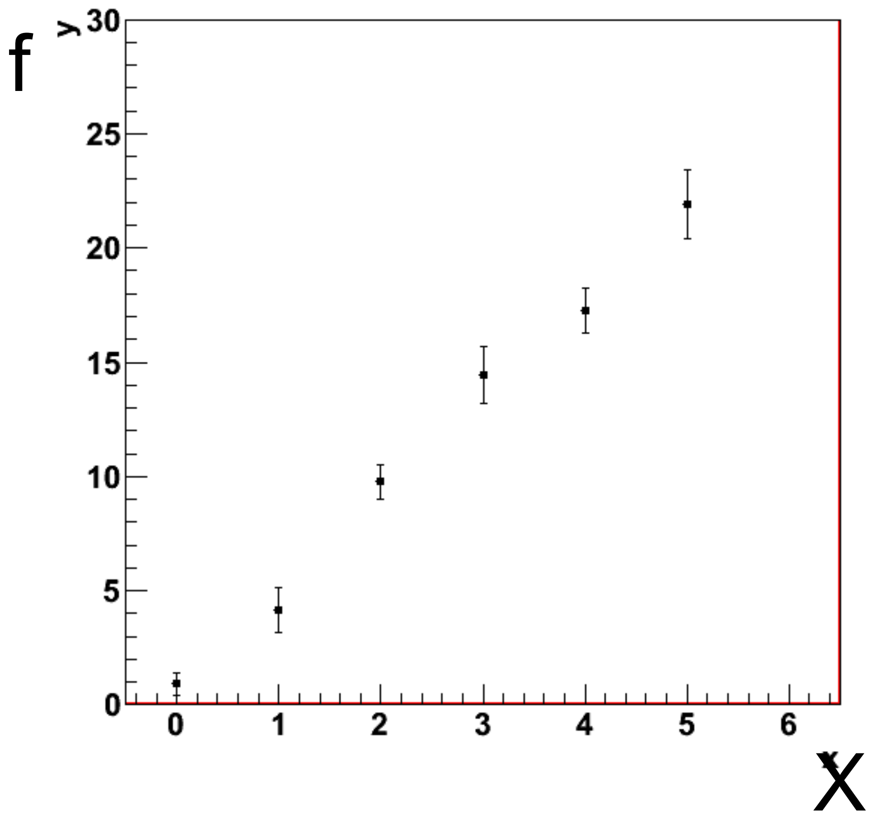


g(t) = 3 + 4 *t does not reproduce data (previous page), but
g(t) = 3 + 5 *t works better.

Ex4 (HW) : 4.1, 4.2 and 4.3

4.1) Write python code to find the straight line fit parameters a and b (and their uncertainties) , S_m, the p-value for the data given in the table below:

x	0	1	2	3	4	5
f	0.92	4.15	9.78	14.46	17.26	21.9
σ	0.5	1.0	0.75	1.25	1.0	1.5



4.2) Make a plot, which superimposes data and the fitted line g(x)=ax+b

4.3) Interpret the results: how good is the fit and why? Give probabilistic interpretation of obtained S_m value.

Answers:

$$a = 4.23 \pm 0.21$$

$$b = 0.88 \pm 0.45$$

S_m=2.078, p-value = 0.72 (72%)