Project 2 – Least Squares Method (LSM)

Reference:
G. Cowan, *Statistical Data Analysis*, Chapter 7

This notebook presents the estimation of model parameters from experimental data affected by uncertainties using the Least Squares Method (LSM).
Three different functional models are tested and compared using the Pearson χ² goodness-of-fit test.

1. Problem Statement

We are given a set of independent experimental measurements x_i, y_i, where:

- x_i is known with negligible uncertainty,
- y_i represents the mean of repeated measurements,
- $sigma_i$ is the standard deviation of the mean.

The goal is to estimate the parameters of three different models using the Least Squares method and determine which model best describes the data using the Pearson χ² test.

2. Models Considered

The following functional forms are investigated:

Model 1 – Power Law

$$y = t_0 x^{t_1}$$

Model 2 – Polynomial

$$y = t_0 + t_1 x + t_2 x^2$$

Model 3 – Exponential-Based Model

$$y = t_0 + t_1 x + t_2 e^x$$

Models 2 and 3 are linear in the parameters and can be solved analytically, while Model 1 is non-linear and requires numerical minimization.

3. Least Squares Method and χ² Test

Assuming Gaussian uncertainties on the measurements, the goodness of fit is quantified by the χ² function:

$$\chi^2 = \sum_{i=1}^{N} \frac{\left[y_i - f(x_i)\right)^2}{\sigma_i^2}$$

The weights used in the Least Squares method are defined as:

$$w_i = \frac{1}{\sigma_i^2}$$

The number of degrees of freedom is:

$$\nu = N - p$$

where ¿) is the number of fitted parameters.

The quality of the fit is assessed using:

- the reduced chi-square ¿),
- the p-value from the χ^2 distribution.

4. Experimental Data

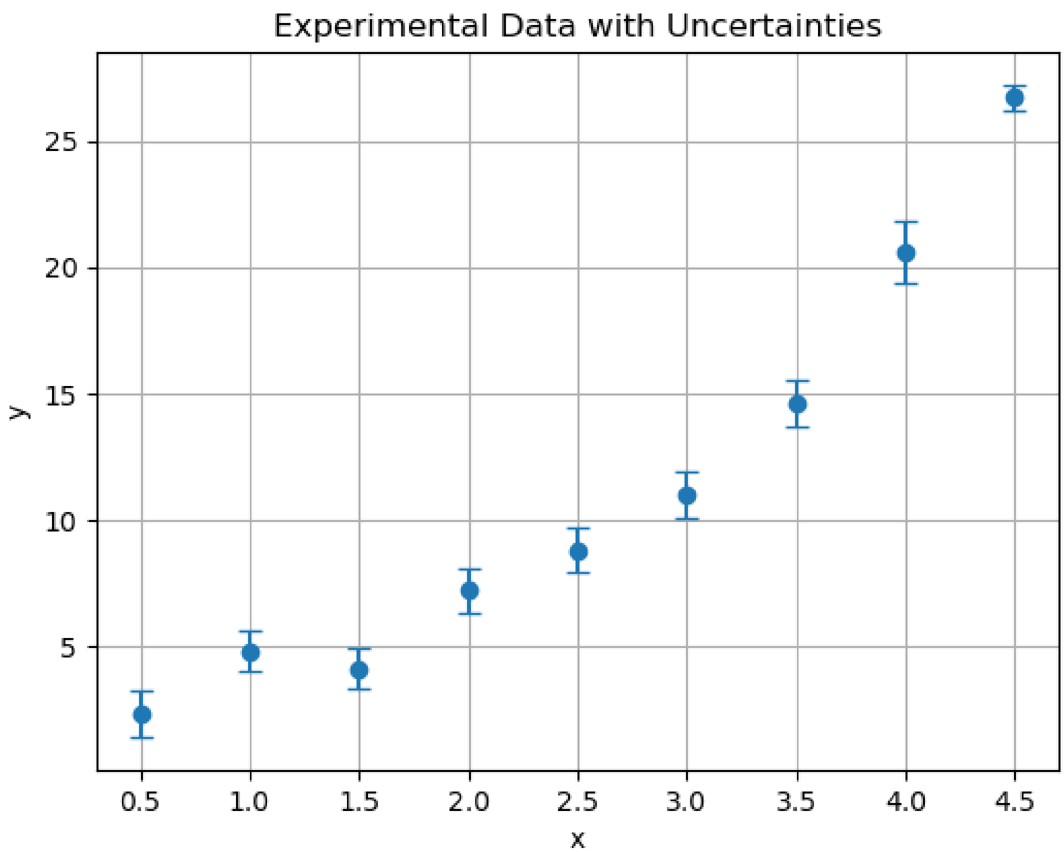
```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from scipy.optimize import curve_fit
from scipy.stats import chi2

data = pd.read_csv("data_points.csv")
data
```

	x	y_mean	std
0	0.5	2.3	0.9
1	1.0	4.8	0.8
2	1.5	4.1	0.8
3	2.0	7.2	0.9
4	2.5	8.8	0.9
5	3.0	11.0	0.9
6	3.5	14.6	0.9
7	4.0	20.6	1.2
8	4.5	26.7	0.5

The experimental data with their uncertainties are shown below.

```python
plt.errorbar(
    data["x"], data["y_mean"],
    yerr=data["std"],
    fmt="o", capsize=4
)
plt.xlabel("x")
plt.ylabel("y")
```

```
plt.title("Experimental Data with Uncertainties")
plt.grid(True)
plt.show()
```



5. Model 2 – Polynomial Fit

$$y = t_0 + t_1 x + t_2 x^2$$

This model is linear in the parameters and is solved using analytic weighted Least Squares.

```
x = data["x"].values
y = data["y_mean"].values
sigma = data["std"].values
N = len(x)

Phi2 = np.column_stack([np.ones_like(x), x, x**2])
W = np.diag(1 / sigma**2)

M2 = Phi2.T @ W @ Phi2
b2 = Phi2.T @ W @ y

t2 = np.linalg.solve(M2, b2)
```

```
cov2 = np.linalg.inv(M2)
err2 = np.sqrt(np.diag(cov2))

y2_fit = Phi2 @ t2
chi2_2 = np.sum(((y - y2_fit) / sigma)**2)
nu2 = N - 3
pval2 = 1 - chi2.cdf(chi2_2, nu2)
```

Model 2 – Results

```
pd.DataFrame({
    "Parameter": ["t0", "t1", "t2"],
    "Estimate": t2,
    "Uncertainty": err2
})
```

	Parameter	Estimate	Uncertainty
0	t0	4.357350	1.086546
1	t1	-2.389180	0.965616
2	t2	1.621382	0.176181

```
pd.DataFrame({
    "chi2": [chi2_2],
    "nu": [nu2],
    "chi2/nu": [chi2_2 / nu2],
    "p-value": [pval2]
})
```

	chi2	nu	chi2/nu	p-value
0	9.119416	6	1.519903	0.166973

The polynomial model provides an acceptable description of the data.
The reduced χ^2 is moderately close to unity and the p-value indicates that the observed discrepancies are compatible with statistical fluctuations.

6. Model 3 – Exponential-Based Fit

$$y = t_0 + t_1 x + t_2 e^x$$

This model is also linear in the parameters and is solved analytically.

```
Phi3 = np.column_stack([np.ones_like(x), x, np.exp(x)])

M3 = Phi3.T @ W @ Phi3
b3 = Phi3.T @ W @ y

t3 = np.linalg.solve(M3, b3)
cov3 = np.linalg.inv(M3)
err3 = np.sqrt(np.diag(cov3))
```

```
y3_fit = Phi3 @ t3
chi2_3 = np.sum(((y - y3_fit) / sigma)**2)
nu3 = N - 3
pval3 = 1 - chi2.cdf(chi2_3, nu3)
```

Model 3 – Results

```
pd.DataFrame({
    "Parameter": ["t0", "t1", "t2"],
    "Estimate": t3,
    "Uncertainty": err3
})
```

	Parameter	Estimate	Uncertainty
0	t0	1.139982	0.804583
1	t1	2.201918	0.475663
2	t2	0.174900	0.018526

```
pd.DataFrame({
    "chi2": [chi2_3],
    "nu": [nu3],
    "chi2/nu": [chi2_3 / nu3],
    "p-value": [pval3]
})
```

	chi2	nu	chi2/nu	p-value
0	4.685278	6	0.78088	0.584764

The exponential-based model provides an excellent fit to the data.
The reduced χ^2 is close to unity and the p-value is comfortably large, indicating strong agreement between the model and the experimental measurements.

7. Model 1 – Power-Law Fit

$$y = t_0 x^{t_1}$$

This model is non-linear in the parameters and requires numerical minimization of χ^2.

```
def model1(x, t0, t1):
    return t0 * x**t1

t1_par, cov1 = curve_fit(
    model1,
    x, y,
    sigma=sigma,
    absolute_sigma=True,
    p0=[1, 1],
    maxfev=10000
)
```

```
err1 = np.sqrt(np.diag(cov1))

y1_fit = model1(x, *t1_par)
chi2_1 = np.sum(((y - y1_fit) / sigma)**2)
nu1 = N - 2
pval1 = 1 - chi2.cdf(chi2_1, nu1)
```

Model 1 – Results

```
pd.DataFrame({
    "Parameter": ["t0", "t1"],
    "Estimate": t1_par,
    "Uncertainty": err1
})
```

	Parameter	Estimate	Uncertainty
0	t0	1.776567	0.219753
1	t1	1.787192	0.086615

```
pd.DataFrame({
    "chi2": [chi2_1],
    "nu": [nu1],
    "chi2/nu": [chi2_1 / nu1],
    "p-value": [pval1]
})
```

	chi2	nu	chi2/nu	p-value
0	30.293363	7	4.327623	0.000084

The power-law model yields a very large χ^2 and a very small p-value.
This indicates that the discrepancies between the model predictions and the data are too large to be attributed to statistical fluctuations.
The model is therefore statistically incompatible with the data and must be rejected.

8. Model Comparison and Conclusions

```
pd.DataFrame({
    "Model": ["Power Law", "Polynomial", "Exponential-Based"],
    "chi2/nu": [chi2_1/nu1, chi2_2/nu2, chi2_3/nu3],
    "p-value": [pval1, pval2, pval3],
    "Conclusion": ["Rejected", "Acceptable", "Best"]
})
```

	Model	chi2/nu	p-value	Conclusion
0	Power Law	4.327623	0.000084	Rejected
1	Polynomial	1.519903	0.166973	Acceptable
2	Exponential-Based	0.780880	0.584764	Best

Among the three models considered, the exponential-based model provides the best overall description of the experimental data, as confirmed by both the reduced χ^2 and the p-value.

The polynomial model yields an acceptable but less optimal fit, while the power-law model is clearly rejected due to poor statistical agreement with the data.