2 Alle Standardabweichungen σ_i sind bekannt, bzw. die Kovarianzmatrix der Daten ist bekannt: Minimieren der χ^2-Funktion.

2.1 Allgemeine Behandlung

Definition der χ^2-Funktion. Hier definieren wir die $\chi^2 - Funktion$

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

für statistisch unabhängige ϵ_i, bzw.

$$\chi^2(\theta) := \sum_{i=1}^{N} \sum_{j=1}^{N} (y_i - f(x_i;\theta)) W_{ij} (y_j - f(x_j;\theta))$$

im statistisch abhängigen Fall.

Bemerkung 17. *Bei bekannten Wertepaaren (x_i, y_i) und bekannten σ_i ist die χ^2-Funktion eine Funktion der M unbekannten Parameter θ.*

Beispiel 4. *Ist das Modell der Messung $y_i = \mu + \epsilon_i$, dann ist die χ^2-Funktion quadratisch im Parameter μ, nämlich*

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

Graphisch beschreibt diese Funktion eine nach oben geöffnete Parabel. Damit ist klar, dass ein eindeutiges Minimum von χ^2 existiert.

Beispiel 5. *Ist das Modell der Messung $y_i = ax_i + b + \epsilon_i$, dann ist die χ^2-Funktion eine quadratische Form der beiden Parameter a und b. Die Konturlinien dieser Funktion sind Ellipsen und die Funktion besitzt ein eindeutiges Minimum.*

Bemerkung 18. *Ist das Modell der Messung nichtlinear in den Parametern θ, dann besitzt die χ^2-Funktion nicht notwendigerweise ein eindeutiges Minimum.*

2.1.1 Priorverteilung der Parameter θ.

Im Rahmen der Gaussschen Fehlerrechnung verwendet man meist Priorverteilungen für die Parameter, die jedem möglichen Parameterwert dieselbe Wahrscheinlichkeit(sdichte) zuordnen (sog. flache Priorverteilungen). Damit ist

$$\mathrm{prob}(\theta) = \mathrm{pdf}(\theta) d\theta = \mathrm{const.} \times d\theta.$$

Bemerkung 19. *Diese Wahl folgt dem 'Prinzip des unzureichenden Grundes' (Indifferenzprinzip) von Laplace. Wenn keine Gründe dafür bekannt sind, einen Parameterwert gegenüber anderen zu begünstigen, dann sind alle Parameterwerte als gleich wahrscheinlich anzusehen. (Wie beim Würfel: jedem der sechs möglichen Ergebnisse wird dieselbe Wahrscheinlichkeit 1/6 zugeordnet.)*

Bemerkung 20. *Wenn ein Modell Parameter besitzt, die nicht beliebige reelle Werte annehmen können, dann kann der Wertebereich über die Priorwahrscheinlichkeit eingeschränkt werden. Ist ein Parameter θ zum Beispiel mit Sicherheit positiv (z.B. eine Masse, eine Zerfallskonstante, etc.), dann wählt man am einfachsten die Priorverteilung*

$$\mathrm{prob}(\theta) = \begin{cases} \text{const.} & \textit{for } \theta \geq 0 \\ 0 & \textit{elsewhere.} \end{cases}$$

Auf diese Weise kann man den möglichen Parameterwerten einfach Bedingungen auferlegen.

Bemerkung 21. *Es gibt weiterführende Prinzipien und Methoden Priorverteilungen festzulegen. Besonders zu erwähnen ist dabei die Methode der maximalen Entropie (maximum entropy method, maxEnt), die wir hier jedoch nicht weiter besprechen.*

2.1.2 Anwendung des Bayesschen Theorems: von der Priorverteilung zur Posteriorverteilung.

Das Bayessche Theorem ist zentral für die Parameterabschätzung in der Datenanalyse. Bezeichnen wir mit D die gemessenen Daten, M das Modell für die Messung, mit P die Parameter des Modells, dann können wir das Bayessche Theorem in der Form

$$\boxed{\mathrm{prob}(P|D,M) = \frac{\mathrm{prob}(P|M)\mathrm{prob}(D|P,M)}{\mathrm{prob}(D|M)} \quad \text{(Bayessches Theorem)}}$$

schreiben.

Bemerkung 22. *Das Bayessche Theorem ist eine modifizierte Form der Produktregel der Wahrscheinlichkeitsrechnung. Letztere lautet, in unserer Notation,*

$$\mathrm{prob}(D,P|M) = \mathrm{prob}(D|M)\mathrm{prob}(P|D,M) = \mathrm{prob}(P|M)\mathrm{prob}(D|P,M).$$

Dabei ist $\mathrm{prob}(D,P|M)$ *die Verbundwahrscheinlichkeit für D und P.*

Bemerkung 23. *Die vier Wahrscheinlichkeiten, die im Bayesschen Theorem auftreten, haben im Rahmen der Parameterabschätzung eigene Namen:* $\mathrm{prob}(\theta|M)$ *ist die oben bereits eingeführte Priorwahrscheinlichkeit der Parameter bei gegebenem Modell.* $\mathrm{prob}(D|P,M)$ *wird als Stichprobenverteilung (sampling distribution) der Daten bei gegebenen Parametern* *bezeichnet, wenn wir die Abhängigkeit von den Daten D in den Vordergrund stellen. Interessiert uns hingegen die Abhängigkeit dieser Funktion von den Parameterwerten P, dann nennen wir sie die 'Likelihood' der Parameter.* $\mathrm{prob}(P|D,M)$ *heisst Posteriorverteilung der Parameter bei gegebenen Daten.* $\mathrm{prob}(D|M)$ *nennt man 'prior predictive probability for the data', oder auch Evidenz für das Modell.*

Bemerkung 24. *Alle Wahrscheinlichkeiten in unserer Form des Bayesschen Theorems setzen voraus, dass das Modell M für die Messung bekannt und für die Analyse zutreffend ist. Es ist die Aufgabe des Experimentalphysikers, ein geeignetes Modell zu verwenden. Im Rahmen des Modellvergleichs, der später zu besprechen sein wird, lassen sich verschiedene geeignete Modelle bezüglich ihrer Kompatibilität mit den Daten vergleichen.*

Bemerkung 25. *Alle Wahrscheinlichkeiten auf der rechten Seite des Bayesschen Theorems sind bereits vor der Messung bekannt, die Wahrscheinlichkeit auf der linken Seite hingegen kann erst nach der Messung (bei bekannten Daten) berechnet werden. Auf diese Weise beschreibt das Bayessche Theorem den Prozess des 'Lernens aus den Daten'. Schematisch:*

$$\mathrm{prob}(P|M) \overset{Daten}{\longmapsto} \mathrm{prob}(P|D,M)$$

Bemerkung 26. *Wir interpretieren Wahrscheinlichkeiten als eine Beschreibung unseres Kenntnisstandes.* $\mathrm{prob}(P|M)$ *beschreibt, was wir über die Parameter wissen, bevor wir messen. Entsprechend beschreibt* $\mathrm{prob}(P|D,M)$*, was wir über die Parameter nach der Messung wissen. Die Veränderung von der Priorverteilung zur Posteriorverteilung der Parameter beschreibt, was wir aus der Messung über die Parameter gelernt haben.*

2.1.3 Posteriorverteilung für die Parameter.

Bei Annahme flacher Priorverteilungen für die Parameter θ ist die Posteriorverteilung proportional zur Likelihood, also

$$\mathrm{pdf}(\theta|\{(x_i,y_i)\},\{\sigma_i\}) \propto \exp\left(-\frac{1}{2}\chi^2(\theta)\right).$$

Bemerkung 27. *Besitzt die* $\chi^2(\theta)$*-Funktion ein eindeutiges Minimum für gewisse Parameterwerte* $\hat{\theta}$*, so besitzt die Posteriorverteilung an genau dieser Stelle ein Maximum.*

Bemerkung 28. *Daten zu fitten heisst, die Parameter* $\hat{\theta}$ *zu finden, welche die Posteriorwahrscheinlichkeit maximieren. Das kommt im Rahmen der Gaussschen Fehlerrechnung mit bekannten Fehlerbalken der Messpunkte einer Minimierung von* $\chi^2(\theta)$ *gleich. Allgemeiner können wir sagen, wir minimieren den negativen Logarithmus der Posteriorverteilung.*

Der negative Logarithmus der Posteriorverteilung ist

$$L(\theta) = \mathrm{const.} + \frac{1}{2}\chi^2(\theta).$$

Zum Minimieren dieser Funktion wird häufig der Levenberg-Marquardt-Algorithmus verwendet.[1] Auf diese Weise finden wir $\hat{\theta}$ und $\chi^2_{\min} = \chi^2(\hat{\theta})$, sowie $L_{\min} = L(\hat{\theta})$.

Python: die Funktion `scipy.optimize.curve_fit`. In Python kommt der Levenberg-Marquardt Algorithmus bei der Funktion `scipy.optimize.curve_fit` zum Einsatz, welche die Minimierung von $\chi^2(\theta)$ durchführen kann. Sie bekommt als Eingabeparameter die Fitfunktion $f(x_i;\theta)$, die Datenpunkte (x_i, y_i), sowie die Fehlerbalken σ_i. Zusätzlich werden noch Startwerte für die Parameter angegeben. Wir geben hier ein Beispiel für die Anwendung dieser Funktion:

```
import numpy as np
from scipy.optimize import curve_fit

# Fitfunktion definieren
def fit_func(x,param0,param1):
    return param0*exp(-param1*x)

# Vektor mit den Fehlerbalken (Standardabweichungen der Datenpunkte)
# definieren
sig = 0.1*np.ones(shape=xdata.shape)

# Startwerte f. Parameter festlegen
startparam = (1,1)

# fit the data with scipy.optimize.curve_fit
result = curve_fit(fit_func,xdata,ydata,startparam,sigma=sig,
    absolute_sigma=True)

p0hat = result[0][0]
p1hat = result[0][1]
```

2.2 Spezialfall 1: Konstante Fitfunktion und unkorrelierte Fehler

Nehmen wir

$$f(x_i, \mu) = \mu,$$

[1] K. Levenberg, *A method for the solution of certain non-linear problems in least squares*, Quart. Appl. Math. **2**, 164 (1944); D.W. Marquardt, *An Algorithm for Least-Squares Estimation of Nonlinear Parameters*, J. Soc. Industrial and Applied Mathematics **11**, 431 (1963).

dann ist

$$\chi^2(\mu) = \sum_{i=1}^{N} \frac{(y_i - \mu)^2}{\sigma_i^2} = \sum_{i=1}^{N} \frac{y_i^2 - 2y_i\mu + \mu^2}{\sigma_i^2}.$$

Bezeichnen wir

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

als Gewichte,[2] und

$$w = \sum_{i=1}^{N} w_i$$

dann haben wir

$$\chi^2(\mu) = w \sum_{i=1}^{N} \frac{w_i}{w}(y_i^2 - 2y_i\mu + \mu^2) = w(\langle y_i^2\rangle - 2\langle y_i\rangle\mu + \mu^2) = w(\mu - \langle y_i\rangle)^2 + w\mathrm{Var}(y_i),$$

mit den gewichteten Mittelwerten

$$\langle y_i\rangle = \sum_{i=1}^{N} \frac{w_i}{w} y_i \quad \text{und} \quad \langle y_i^2\rangle = \sum_{i=1}^{N} \frac{w_i}{w} y_i^2$$

und der gewichteten Varianz

$$\mathrm{Var}(y_i) = \sum_{i=1}^{N} \frac{w_i}{w}(y_i - \langle y_i\rangle)^2.$$

Damit finden wir den Schätzwert

$$\hat{\mu} = \langle y_i\rangle.$$

Bemerkung 29. *Intuitiv macht die Gewichtung einzelner Werte mit $1/\sigma_i^2$ Sinn. Werte mit grossen Fehlerbalken erhalten so ein kleines Gewicht für den Mittelwert, Werte mit kleinen Fehlerbalken ein grösseres.*

In Phython berechnet man gewichtete Mittelwerte, wie im nachfolgenden Beispiel gezeigt.

```
from numpy import average
from numpy import square
```

[2] Die Python-Funktion `numpy.polyfit()`, die für Polynomfits mit bekannten Standardabweichungen verwendet werden kann, nimmt als Eingabe-Parameter stattdessen die Gewichte $w_i = 1/\sigma_i$, minimiert aber ebenfalls die χ^2-Funktion.

```
# Weighted average of x-values
Meanx = average(xdata[:,0],weights=xdata[:,1])

# Weighted average of x**2
Meanx2 = average(square(xdata[:,0]),weights=xdata[:,1])

# Weighted variance
Varx = Meanx2 - Meanx*Meanx
```

Spezialfall, wenn alle Standardabweichungen σ_i gleich σ sind. In diesem Fall ist

$$w = \frac{N}{\sigma^2}, \quad \text{und} \quad \langle y_i \rangle = \frac{1}{N} \sum_{i=1}^{N} y_i$$

so dass

$$\mu = \langle y_i \rangle$$

der ungewichtete Mittelwert ist. In Python berechnet man den ungewichteten Mittelwert wie im nachfolgenden Beispiel gezeigt.

```
from numpy import mean

Meanx = mean(data[:,0])
```

2.3 Spezialfall 2: lineare Fitfunktion und unkorrelierte Fehler

Nehmen wir den Fall der linearen Regression

$$f(x_i, \alpha, \beta) = \alpha(x_i - \langle x_i \rangle) + \beta,$$

wobei

$$\langle x_i \rangle = \sum_{i=1}^{N} \frac{w_i}{w} x_i,$$

dann ist

$$\begin{aligned}\chi^2(\alpha, \beta) = w \sum_{i=1}^{N} \frac{w_i}{w} (y_i - \alpha(x_i - \langle x_i \rangle) - \beta)^2 = \\ w(\langle y_i^2 \rangle + \alpha^2 \langle (x_i - \langle x_i \rangle)^2 \rangle + \beta^2 - 2\alpha \langle (y_i - \langle y_i \rangle)(x_i - \langle x_i \rangle) \rangle - 2\beta \langle y_i \rangle) = \\ w\mathrm{Var}(x_i) \left[\alpha - \frac{\mathrm{Cov}(x_i, y_i)}{\mathrm{Var}(x_i)} \right]^2 + w(\beta - \langle y_i \rangle)^2 + w\mathrm{Var}(y_i)(1 - \rho^2),\end{aligned}$$

wobei alle statistischen Grössen als gewichtete Mittelwerte zu berechnen sind. Insbesondere ist die empirische Kovarianz gegeben durch

$$\mathrm{Cov}(x_i, y_i) = \sum_{i=1}^{N} \frac{w_i}{w}(x_i - \langle x_i \rangle)(y_i - \langle y_i \rangle),$$

und der empirische lineare Korrelationskoeffizient ist

$$\rho = \frac{\mathrm{Cov}(x_i, y_i)}{\sqrt{\mathrm{Var}(x_i)\mathrm{Var}(y_i)}}.$$

Damit sind unsere Schätzwerte, welche die χ^2-Funktion minimieren gegeben durch

$$\hat{\alpha} = \frac{\mathrm{Cov}(x_i, y_i)}{\mathrm{Var}(x_i)}, \quad \text{und} \quad \hat{\beta} = \langle y_i \rangle.$$

In Python berechnet man die gewichtete Kovarianz wie im nachfolgenden Beispiel gezeigt.

```
from numpy import average

Meanx = average(xdata[:,0],weights=ydata[:,1])
Meany = average(ydata[:,0],weights=ydata[:,1])
xy = xdata[:,0]*ydata[:,0];
Meanxy = average(xy,weights=ydata[:,1])
Covxy = Meanxy - Meanx*Meany
```