

Blatt03

November 20, 2023

1 Rechnernutzung in der Physik

Institut für Experimentelle Teilchenphysik Institut für Theoretische Teilchenphysik
Prof. G. Quast, Prof. M. Steinhauser Dr. A. Mildenerger, Dr. Th. Chwalek [Ilias Seite zum Kurs](#)
WS 2023/24 – Blatt 03 Abgabe: Montag 27.11.2023 bzw. Dienstag 28.11.2023

[]: Auf dem dritten Übungsblatt beschäftigen Sie sich mit einem der wichtigsten
↳ Grundbegriffe der modernen Parameteranpassung - der
↳ Maximum-Likelihood-Methode. Sie haben mit großer Wahrscheinlichkeit im Laufe
↳ Ihres Studiums oder Privatlebens bereits die Maximum-Likelihood-Methode
↳ (oder ihre Abwandlungen) verwendet, um die Steigung einer Geraden oder
↳ andere Parameter einer Messung zu bestimmen. Mit diesem Übungsblatt
↳ beschäftigen Sie sich noch mal ganz genau mit der Methodik und nehmen
↳ hoffentlich mit, auf welchem Prinzip die unterschiedlichen eingebauten
↳ Fitfunktionen in *SciPy*, *kafé2*, *probfit* und vielen mehr (ja, auch
↳ Excel) aufgebaut sind. Sie werden merken, dass sich hinter all den
↳ angesprochenen Methoden keine schwarze Magie befindet, sondern die Umsetzung
↳ und Überführung auf eigene Programme ganz einfach ist. Mit einem guten
↳ Verständnis der Maximum-Likelihood-Methode können Sie dann nicht nur die
↳ Methode der kleinsten Quadrate verstehen, sondern auch schon bald
↳ Anpassungen wie diese selbst durchführen.

In der ersten Aufgabe beschäftigen Sie sich mit einem vermutlich bereits bekanntem Beispiel aus dem Praktikum: Sie haben eine radioaktive Quelle erhalten und messen elektronisch die Klicks eines Geiger-Müller-Zählrohrs in aufeinanderfolgenden, gleich langen Zeitintervallen. Die Messdaten haben Sie in der Datei *Geiger_Zaehler.txt* abgespeichert und wollen nun für die weitere Auswertung die Anzahl der radioaktiven Zerfälle pro Zeitintervall bestimmen.

Bei der Auswertung der Daten erinnern Sie sich an die Vorlesung *Rechnernutzung* und die Parameterschätzung mithilfe der Maximum Likelihood Methode und beschließen, statt *kafé2* zu nutzen, die gesuchte Zahl selbst abzuschätzen. Sie schauen zurück in die Folien der Vorlesung und entdecken die Funktion

$$L(\{x_i\} | a) = \prod_{i \leq n} f(x_i, a).$$

Sie erinnern sich daran, dass $\{x_i\}$ ihre Messungen sind, $f(x, a)$ die zugrunde liegende Wahrscheinlichkeitsdichte und a der gesuchte Parameter ist.

1.1 a) Likelihoodfunktion

Im ersten Aufgabenteil überprüfen Sie graphisch die Form der Likelihoodfunktion. > Frage: Warum ist es gerechtfertigt die Poissonverteilung als zugrunde liegende Wahrscheinlichkeitsdichte für den Ausgang des Zählexperimentes zu wählen? Welche Abweichungen von der Annahme gibt es beim Aufbau und der Messung?

Antwort:

Implementieren Sie die Likelihoodfunktion $L(\{x_i\} | a)$. Es bietet sich an, die Wahrscheinlichkeitsdichte als Argument der Funktion zu übergeben. Denken Sie daran, dass Sie einen gewissen Parameterbereich abdecken müssen. Es bietet sich also an nicht nur x , sondern auch a als Array zu behandeln.

```
[ ]: import numpy as np
import matplotlib.pyplot as plt
import scipy.special as sp
```

```
[ ]: # Poisson distribution
def fPoisson(x, nu_P):
    k=np.around(x)
    return (nu_P**k) / np.exp(nu_P) / sp.gamma(k+1.)

# Likelihood function
```

Betrachten Sie nun die Likelihoodfunktion für eine steigende Zahl an Messergebnissen $L(\{x_i\} | a)$. Stellen Sie dafür die Likelihoodfunktion für a im Intervall $[2, 9]$ dar für * nur den ersten Messwert * den ersten und den zweiten Messwert * die ersten 10 Messwerte * die ersten 100 Messwerte.

Sie können mithilfe von `np.loadtxt("filename")` die Messdaten aus dem Verzeichnis laden, indem sie das Notebook ausführen. Sollten Sie also das Notebook von ihrem eigenen PC in jupytertermachine.etp.kit.edu laden oder lokal auf ihrem Rechner arbeiten, denken Sie daran, zuerst die Datei `Geiger_Zaehler.txt` herunterzuladen.

```
[ ]: # load observations

# create parameter intervall

# list for desired observation numbers

#build plots
```

Frage: Erläutern Sie das Verhalten des Maximums der Likelihoodfunktion. Schätzen Sie den Wert des gesuchten Parameters ab.

Antwort:

1.2 b) Negative Log Likelihood (NLL)

Beachten Sie die y -Skala bzw. den Wertebereich der Likelihoodfunktion für die vier verschiedenen Anzahlen an Messpunkten. Für eine große Anzahl an Messpunkten nimmt die Funktion immer kleinere Werte an. Eine Logarithmierung des Ausdruck vereinfacht nicht nur die mathematische Analyse, sondern hilft auch numerisch, da das Produkt einer großen Anzahl kleiner Wahrscheinlichkeiten die numerische Genauigkeit des Computers für Fließkommaoperationen leicht unterschreiten kann, und dies wird gelöst, indem stattdessen die effizientere Summe der natürlichen Logarithmen der Wahrscheinlichkeiten ("Log Likelihood") berechnet wird. Statt nach dem Maximum der Likelihoodfunktion zu suchen, wird häufig nach dem Minimum der negativen Log Likelihood (NLL) gesucht.

$$-\ln(L(\{x_i\}|a)) = -\sum_{i \leq n} \ln(f(x_i, a)).$$

Wiederholen Sie den ersten Aufgabenteil mit der NLL Methode. Implementieren Sie dazu eine neue Funktion, die die NLL ausgibt. Betrachten Sie den gleichen Parameterbereich aber diesmal für * nur den ersten Messwert * den ersten und den zweiten Messwert * die ersten 10 Messwerte * **alle Messwerte** statt nur 100 Messwerte

Stellen Sie Ihr Ergebnis geeignet dar.

```
[ ]: # Negative Log Likelihood function

#build plots
```

1.3 c) Parameterschätzung

Bestimmen Sie nun den gesuchten Parameter und schätzen Sie dessen 1σ Konfidenzintervall ab, indem Sie die *asymmetrischen* Grenzen für die Unsicherheit angeben.

Für den ersten Schritt kann die *NumPy*-Methode `np.argmin()` hilfreich sein. Sie gibt den Index des Arrays am Minimum aus. Den zweiten Schritt erhalten Sie, indem Sie diejenigen Punkte finden, deren NLL Wert um $1/2$ größer sind als der des Maximums. Das 1σ Konfidenzintervall entspricht dann dem Wertebereich $[\hat{a} - \Delta^-, \hat{a} + \Delta^+]$. Die üblich Angabe für den besten Schätzwert erfolgt dann über folgende Darstellung:

$$\hat{a}_{-\Delta^-}^{+\Delta^+}.$$

Hier eine schnelle Illustration:

```
[ ]: x = np.linspace(0, 10, 1000)
y = (x - 3)**2 + 0.2

dots_x = [3 - 1/np.sqrt(2), 3, 3 + 1/np.sqrt(2)]
dots_y = [0.7, 0.2, 0.7]

plt.plot(x, y, 'b-', label='NLL')
```

```

plt.hlines(0.7, 1, 5, color='green', label=r'NLL $\hat{a}$ +0.5')
plt.plot(dots_x, dots_y, 'ro')

colors = ['gold', 'orange', 'tomato']
labels = [r'Unterschranke  $\hat{a}-\Delta^-$ ', r'Minimum  $\hat{a}$ ',
↪r'Oberschranke  $\hat{a}+\Delta^+$ ]
for i in range(3):
    plt.vlines(dots_x[i], 0, dots_y[i], linestyle='dashed', color=colors[i],
↪label=labels[i])

plt.ylim(-0.1, 1)
plt.xlim(1, 5)
plt.xlabel(r' $a$ ')
plt.ylabel(r'NLL  $-\ln L(\left(x_i\right)|a)$ ')
plt.legend(bbox_to_anchor=(1.05, 1), loc='upper left')
plt.show()

```

Es bietet sich an die Zwischenschritte als Funktion zu implementieren, damit sie erneut verwendet werden können.

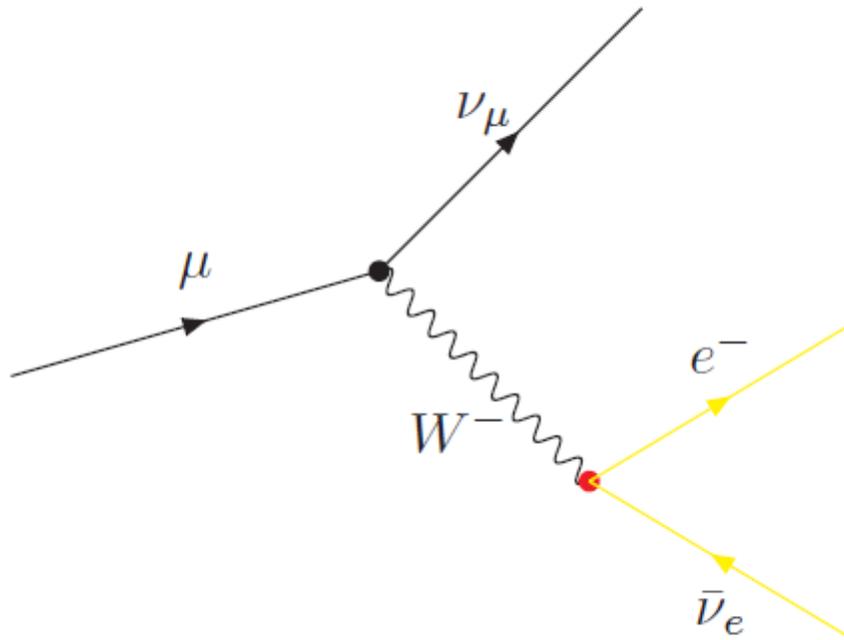
```

[ ]: # define function for estimator and CL calculation

# print results

```

In der zweiten Aufgabe schauen Sie sich ein vielleicht bereits bekanntes Beispiel aus dem Praktikum an: Der Myon-Zerfall. Das Myon hat dieselben Eigenschaften wie das Elektron, nur ist es ca. 200 Mal schwerer. Myonen entstehen beispielsweise als Zerfallsprodukte kosmischer Strahlung und können dann am Boden z. B. durch ein praktisch zeitgleiches Signal (“Koinzidenz”) in mehreren Lagen von Szintillationszählern (Detektoren, die bei Teilchendurchgang Licht emittieren, das mit Photondetektoren nachgewiesen werden kann) nachgewiesen werden. Dabei zerfallen sie größtenteils gemäß dem folgenden Prozess, illustriert als “Feynman-Diagramm”, bei dem das Myon in ein Elektron, ein Elektron-Antineutrino und ein Myon-Neutrino zerfällt und ein virtuelles W-Boson als “Kraftteilchen” fungiert:



Bei physikalischen Experimenten treten häufig Störparameter auf, die für das physikalische Ergebnis irrelevant sind, aber dessen Unsicherheit beeinflussen. Die Berechnung und Analyse der Profile-Likelihood für den jeweils interessierenden physikalischen Parameter ist eine einfache Möglichkeit, die durch Störparameter verursachten Unsicherheiten zu berücksichtigen.

In der Datei *tau_mu.dat* finden Sie 150 Werte gemessener Lebensdauern von Myonen, die im oder in der Nähe eines Myondetektors gestoppt wurden. Die Zeitdifferenz zwischen dem Nachweis des einfallenden Myons und dem Eintreffen des Elektrons aus dem Zerfall ist die Messgröße. Bedingt durch das Messverfahren sind nur Zeiten zwischen $\Delta t_{min} = 1.0 \mu s$ und $\Delta t_{max} = 11.5 \mu s$ erfasst. Als Störparameter bei den Messungen tritt ein Untergrund aus Zufallskoinzidenzen von Signalen auf, die im Bereich von $[\Delta t_{min}, \Delta t_{max}]$ als flach verteilt angenommen werden können.

In den folgenden Teilaufgaben soll die Myonenlebensdauer τ_μ bestimmt werden. Sie werden dabei bemerken, wie einfach es ist, mit dem Wissen aus der Vorlesung und der ersten Aufgabe eine reelle Größe abzuschätzen.

1.4 a) Verteilungsdichte der Zeitdifferenzen

Zunächst soll die zugrunde liegende Wahrscheinlichkeitsdichte der Messdaten erstellt werden. Aus dem Aufbau der Messung ist zu entnehmen, dass die Verteilung der gemessenen Zeitdifferenzen aus der Summe einer Gleichverteilung und einer Exponentialverteilung besteht. * Die **Gleichverteilung** für den Untergrund aus Zufallskoinzidenzen ist durch den relativen Anteil f_b an der Gesamtanzahl gemessener Zeitdifferenz als Störparameter skaliert. * Die **Exponentialverteilung** für den Myonenzerfall ist charakterisiert durch die Myonenlebensdauer τ_μ und beinhaltet die noch

zu bestimmende Normierungskonstante C_n . Außerdem wird der Störparameter $f_b \in [0, 1]$ berücksichtigt durch den Faktor $(1 - f_b)$, so dass die relativen Anteile in der Summe Eins ergeben.

$$\text{PDF}(t) = C_n(1 - f_b) \exp\left\{-\frac{t}{\tau_\mu}\right\} + \frac{f_b}{\Delta t_{max} - \Delta t_{min}}$$

Zur Vorbereitung der Auswertung führen Sie folgende Schritte aus: 1. Implementieren Sie eine Funktion, die die Wahrscheinlichkeitsdichte ausgibt. Beachten Sie, dass die PDF auf Eins normiert sein muss. Bestimmen Sie dazu die Normierungskonstante C_n . 2. Implementieren Sie eine Funktion, die die NLL der PDF ausgibt, falls diejenige, die Sie in der ersten Aufgabe geschrieben haben, nicht ausreichen sollte. 3. Lesen Sie die Daten aus der gegebenen Datei ein und histogrammieren Sie diese, damit Sie eine grobe Vorstellung des Problems erhalten.

```
[ ]: # pdf for myuon decay

# NLL if not implemented

# load observations

# plot obsevration
```

1.5 b) Bestimmung der Profile Likelihood

Zur Bestimmung der Profile Likelihood wählen Sie 1000 Werte τ_i für die Lebensdauer τ_μ im Intervall zwischen $1.5 \mu\text{s}$ und $2.9 \mu\text{s}$. Minimieren Sie für jeden Wert die NLL $-\ln L(\Delta t; \tau_i, f_b)$ bezüglich des Parameters f_b . Die Untergrundrate f_b können Sie durch 200 Werte im Intervall $[0, 0.3]$ wählen – dies entspricht einem kleinen Untergrundanteil von ca. 0-30%. Stellen Sie die Profile Likelihood geeignet dar. Zeichnen Sie ebenfalls die Linie ein, die die Abschätzung des 1σ Konfidenzintervalls zeigt.

```
[ ]: # create array for tau and f_b

# creat profile likelihood

# plot profile likelihood
```

1.6 c) Parameterschätzung

Bestimmen Sie als letzten Schritt * den bestmöglichen Schätzer für die Myonenlebenszeit $\hat{\tau}_\mu$, * dessen asymmetrischen Grenzen für die Unsicherheiten $+\Delta^+, -\Delta^-$ * und den bestmöglichen Schätzer für die Untergrundrate \hat{f}_b unter der Annahme $\hat{\tau}_\mu$.

Stellen Sie die histogrammierten Zeitdifferenzen, die Verteilung des Signals unter der Annahme der bestmöglichen Schätzer und alle Ergebnisse in einem Plot dar.

```
[ ]: # calculate best estimate for tau and f_b  
  
# plot final result
```