

# Statistische Methoden der Datenanalyse

Markus Schumacher, Stan Lai, Florian Kiss

## Übung VII

07.12.2012, 11.12.2012

### Anwesenheitsaufgaben

#### Aufgabe 43 'Binned' Maximum Likelihood Fit zu $e^+e^-$ mit Monte Carlo Statistik

Der differentielle Wirkungsquerschnitt für  $e^+e^- \rightarrow \mu^+\mu^-$  folgt der theoretischen Verteilung

$$\frac{d\sigma}{d(\cos\theta)d\phi} \propto 1 + \alpha \cos\theta + \beta \cos^2\theta, \quad (1)$$

wobei  $\theta$  und  $\phi$  für die Winkel des  $\mu^+$  Teilchens in Kugelkoordinaten stehen. Der Winkel  $\theta$  wird relativ zur  $z$ -Achse und der Winkel  $\phi$  relativ zur  $x$ -Achse gemessen. In der Übung sollen Sie einen 'Binned' Maximum Likelihood Fit durchführen, um die Parameter zu finden, welche zur Erstellung der Monte Carlo Ereignisse benutzt wurden. Ein solcher simulierter Datensatz befindet sich in `/home/slai/StatisticsCourse/PS7/MuMuEvGen.root`.

Um eine 'Binned' Maximum Likelihood Statistik zu erstellen betrachte man ein Histogramm mit  $N$  Bins, wobei jedes  $n_i$  Einträge  $\vec{n} = (n_1, \dots, n_N)$  enthält. Für eine Wahrscheinlichkeitsdichtefunktion (WDF)  $f(x; \vec{\theta})$  ist der Erwartungswert gegeben durch  $\vec{\nu} = (\nu_1, \dots, \nu_N)$ , wobei  $\nu_i(\vec{\theta})$  gegeben ist durch das Integral der WDF über die Breite des Bins

$$\nu_i(\vec{\theta}) = n_{tot} \int_{x_i^{min}}^{x_i^{max}} f(x; \vec{\theta}) dx \quad (2)$$

wobei  $x_i^{min}$  und  $x_i^{max}$  für die Grenzen jedes Bins  $i$  stehen.

Daher ist die gemeinsame WDF,  $f_g(n; \vec{\nu})$ , gegeben durch die Multinomialverteilung

$$f_g(n; \vec{\nu}) = \frac{n_{tot}!}{n_1! \dots n_N!} \left(\frac{\nu_1}{n_{tot}}\right)^{n_1} \dots \left(\frac{\nu_N}{n_{tot}}\right)^{n_N} \quad (3)$$

und demzufolge ist die log-Likelihoodfunktion für die gemeinsame WDF gegeben durch

$$\ln \mathcal{L}(\vec{\theta}) = \sum_{i=1}^N n_i \ln \nu_i(\vec{\theta}) \quad (4)$$

Dies ist eine Funktion, welche maximiert werden muss, um die zur Erstellung des Monte Carlo Datensatzes verwendeten Parameter  $\vec{\theta}$  zu finden. Beachten Sie diesbezüglich, dass  $n_{tot}$  nicht in der Berechnung der log-Likelihoodfunktion benötigt wird, da es sich hierbei um eine additive Konstante handelt.

Um die Parameter Ihrer Monte Carlo Verteilung zu berechnen gehen Sie folgende Schritte durch:

- (i) Zum Berechnen der Variablen  $\cos\theta$  findet sich in `/home/slai/StatisticsCourse/PS3/aufgabe43_i.C` ein ROOT-Skript. Kopieren Sie sich dieses Skript in ein Verzeichnis Ihrer Wahl. Öffnen Sie es in einem Texteditor, lassen es laufen, und versuchen Sie die Funktionsweise zu verstehen.

- (ii) Modifizieren Sie dieses Skript nun so, dass Sie die Werte von  $\cos\theta$  in ein Histogramm mit 200 Bins zwischen -1 und +1 einfüllen. Stellen Sie das Histogramm am Ende des Skripts graphisch auf dem Bildschirm dar.
- (iii) Definieren Sie eine TF1 Funktion der Form

$$f(x; \alpha, \beta) = \frac{1 + \alpha x + \beta x^2}{2 + \frac{2\beta}{3}}, \quad (5)$$

welche in dem Maximum Likelihood Fit benutzt werden soll. Dies können Sie mit folgendem Befehl bewerkstelligen:

```
TF1 fitFunk = TF1("fitFunk",
    "( 1.0 + [0]*TMath::Power(x,2) + [1]*x)/(2.0+2.0*[0]/3.0)" , -1., 1.);
```

Um die Parameter auf bestimmte Werte zu setzen, können Sie die Funktion `SetParameter` wie folgt benutzen:

```
fitFunk.SetParameter(0, 1.0);
fitFunk.SetParameter(1, 0.1);
```

- (iv) Definieren Sie - analog zu dem Maximum Likelihood Fit aus Aufgabe 39 - eine Schleife, um über mögliche Werte von  $\alpha$  zu iterieren. Definieren Sie innerhalb dieser Schleife eine weitere Schleife über möglichen Werte von  $\beta$ . Sinnvolle Werte für die Variation von  $\alpha$  sind zwischen 0 und 0.25 und für  $\beta$  zwischen 0.8 und 1.3. Nehmen Sie sowohl für  $\alpha$  als auch für  $\beta$  50 äquidistante Werte an (somit wird die Likelihood für 2500 Kombinationen ausgerechnet).
- (v) Erstellen Sie für jeden Wert von  $\alpha_i$  und  $\beta_i$  eine Schleife über die Bins in Ihrem  $\cos\theta$  Histogramm und berechnen Sie den log-Likelihood Wert. Gehen Sie wie folgt vor:

- Setzen Sie zunächst die Parameter in der bereitgestellten TF1-Funktion auf die gerade aktuellen Werte von  $\alpha_i$  und  $\beta_i$ .
- Erstellen Sie eine Schleife von  $i=1$  (das niedrigste Bin des Histogramms - Bin 0 - ist das Underflow Bin) bis  $i=\text{hist.GetNbinsX}()$ . Dies ist bereits die dritte und gleichzeitig innerste `for`-Schleife.
- Die Anzahl an Einträgen im  $i$ ten Bin erhalten Sie mit der TH1 Funktion `GetBinContent(int i)`
- Um Ihre Funktion über die Binbreite zu integrieren benutzen Sie die TF1 Funktion `Integral(float a, float b)`. `a` steht für die untere, `b` für die obere Integrationsgrenze. Um diese Werte aus dem  $\cos\theta$  Histogramm zu erhalten, benutzen Sie die TH1 Funktion `GetBinLowEdge(int i)`, welche die untere Grenze des  $i$ -ten Bins ausgibt.
- Summieren Sie die log-Likelihoodfunktion für alle Bins des Histogramms auf.

- (vi) Füllen Sie für jedes  $\alpha_i$  und  $\beta_i$  den log-Likelihood Wert in einen zweidimensionalen Graph. Dies funktioniert analog zu einem eindimensionalen Graphen:

```
TGraph2D likeGraph = TGraph2D(likepoints);
```

wobei `likepoints` die Gesamtanzahl an Kombinationen von  $\alpha$  und  $\beta$  ist, für die die Likelihood ausgewertet werden soll.

Jeder Punkt des Graphen kann gesetzt werden durch die `TGraph2D` Funktion:

```
SetPoint(int pointNumber, float alpha, float beta, float loglikelihood)
```

Denken Sie daran, sich `pointNumber` auszurechnen (dies ist nicht die Zählvariable in den Schleifen über die Werte von  $\alpha$  und  $\beta$ !).

- (vii) Zeichnen Sie den Graphen (log-Likelihood gegen  $\alpha$  und  $\beta$ ) unter Benutzung von

```
likeGraph.Draw("colz");
```

wobei die Option `"colz"` ROOT die Anweisung gibt, einen Surface-Plot zu erstellen. Weitere Optionen des Draw-Befehls können im 'ROOT User's Guide' in der `TGraph2D` Sektion gefunden werden.

- (viii) Ermitteln Sie aus den erzeugten Graph eine Abschätzung für  $\hat{\alpha}$  und  $\hat{\beta}$  und deren Standardabweichungen  $\sigma_{\hat{\alpha}}$  und  $\sigma_{\hat{\beta}}$ . Dazu könnten folgende Befehle nützlich sein:

```
float maximum=likeGraph.GetHistogram().GetMaximum();
```

gibt das Maximum des Graphen zurück.

```
likeGraph.GetHistogram().SetMaximum(maximum-0.5);
```

setzt für die Darstellung die minimale Zeichenebene auf das Maximum des Graphen minus 0.5.

- (ix) Vergleichen Sie die Werte Ihres Fits mit denen des Root Fitting Paketes, indem Sie die Histogramm-Anpassungsfunktion

```
hist.Fit("functionName", "LI");
```

benutzen, wobei "functionName" der Name der WDF Funktion ist und die Option "LI" einen Likelihood Fit ausführt, indem die gegebene Funktion über jedes Bin des Histogramms integriert wird.

#### Aufgabe 44 *Erweiterter Maximum Likelihood Fit für eine Signal- and Untergrundverteilung*

In dieser Übung werden wir einen Beispieldatensatz betrachten, welcher aus zwei verschiedenen Arten von Ereignissen besteht: Signalereignisse, welche in einer Gaussverteilung  $f_s(x)$  vorliegen und Untergrundereignisse, welche nach einer Exponentialfunktion  $f_b(x)$  verteilt sind. Aufgabe ist es, die erweiterte Maximum Likelihood Methode zu benutzen, um die Anzahl von Signal- und Untergrundereignissen aus dem Beispieldatensatz zu errechnen.

Betrachten Sie die WDF  $f(x; \mu_s, \mu_b)$  für die Signal- und Untergrundverteilung:

$$f(x; \mu_s, \mu_b) = \frac{\mu_s}{\mu_s + \mu_b} f_s(x) + \frac{\mu_b}{\mu_s + \mu_b} f_b(x) \quad (6)$$

wobei entsprechend  $\mu_s$  und  $\mu_b$  für die Anzahl von Signal- und Untergrundereignissen stehen. Um das Beispiel zu vereinfachen wird angenommen, dass die Parameter der Signal- und Untergrund-WDF bekannt sind. Daher sind  $f_s$  und  $f_b$  Funktionen nur einer Variablen  $x$ .

Falls die Gesamtzahl an Ereignissen poissonverteilt ist, ergibt sich für die WDF:

$$P(n; \mu_s, \mu_b) = \frac{(\mu_s + \mu_b)^n}{n!} \exp(-(\mu_s + \mu_b)) \quad (7)$$

Daher ist die log-Likelihoodfunktion gegeben durch

$$\ln \mathcal{L} = -(\mu_s + \mu_b) + \sum_{i=1}^n \ln[(\mu_s + \mu_b) f(x_i; \mu_s, \mu_b)] \quad (8)$$

Diese Funktion sollen Sie minimieren, um die Parameter der Signal- und Untergrundnormalisierung zu finden.

Zuerst muss ein Datensatz mit der Signal- und der Untergrund-WDF erzeugt werden. Dies wurde bereits in `/home/slai/StatisticsCourse/PS7/aufgabe44_i.c` vorbereitet. Kopieren Sie sich dieses Skript in ein beliebiges Verzeichnis und lassen es laufen. Es erledigt die folgenden Aufgaben:

- (i) Eine TF1 Funktion, welche die Summe einer Gauss- und einer Exponentialfunktion darstellt wird bereitgestellt:

$$f(x; \mu, \sigma, \tau, \mu_s, \mu_b) = \frac{\mu_s}{\mu_s + \mu_b} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(\frac{-(x - \mu)^2}{2\sigma^2}\right) + \frac{\mu_b}{\mu_s + \mu_b} \frac{1}{\tau} \exp\left(-\frac{x}{\tau}\right) \quad (9)$$

in dem Intervall  $0.0 < x < 100.0$ . Die Parameter werden auf  $\mu = 5.0$ ,  $\sigma = 0.5$  und  $\tau = 10.0$  gesetzt.

- (ii) Die TF1-Funktion `GetRandom` wird benutzt, um einen Satz von 1000 Werten der obigen Verteilungsfunktion zu erhalten und diese in ein Array gefüllt. Ein Array ist eine feste Aneinanderreihung von (in diesem Fall) Fließkommazahlen. Um auf das  $i$ -te Element zuzugreifen, wird einfach `data[i]` benutzt.
- (iii) Die Daten werden weiterhin in ein Histogramm gefüllt und graphisch dargestellt.

Als nächstes führen Sie nun selbst einen erweiterten Maximum Likelihood Fit auf dem erstellten Datensatz durch. Ergänzen Sie hierzu das Skript um folgende Komponenten (fügen Sie die Komponenten am Ende des Skriptes hinzu):

- (iv) Definieren sie eine weitere TF1 Funktion, wie oben, der Form  $f(x; \mu, \sigma, \tau, \mu_s, \mu_b)$ . Benutzen Sie die TF1 Funktion `TF1::FixParameter(Int_t parNum, Double_t value)`, um die Funktionsparameter auf die Werte  $\mu = 5.0$ ,  $\sigma = 0.5$  und  $\tau = 10.0$  zu setzen (man könnte diese Parameter variabel lassen, jedoch müsste man dann einen fünfdimensionalen Fit durchführen!).
- (v) Erstellen Sie, analog zu Aufgabe 43, eine Schleife über mögliche Werte des Parameters  $\mu_s$ . Definieren Sie innerhalb dieser Schleife eine weitere Schleife über mögliche Werte von  $\mu_b$ . Sinnvolle Bereiche für  $\mu_s$  und  $\mu_b$  sind zwischen 450 und 550 mit 50 äquidistanten Schritten.
- (vi) Setzen Sie innerhalb der Schleife die Parameter der neuen TF1-Funktion auf die gerade aktuellen Werte von  $\mu_s$  und  $\mu_b$  mittels der `SetParameter(int parnumber, float parwert)`-Funktion.
- (vii) Erstellen Sie für jeden Wert von  $\mu_s$  und  $\mu_b$  eine Schleife über das Array mit den generierten Daten und berechnen Sie den aufsummierten log-Likelihood Wert. Hier kann es nützlich sein, mit der TF1 Funktion `Eval(Double_t x)` den Wert der Funktion am Punkt  $x$  zu berechnen.
- (viii) Befüllen Sie ein `TGraph2D` Objekt mit den aufsummierten log-Likelihood Werten für jeden Wert von  $\mu_s$  und  $\mu_b$ .
- (ix) Zeichnen Sie den Graphen und schätzen Sie damit die Parameter  $\hat{\mu}_s$  und  $\hat{\mu}_b$  und deren Standardabweichungen  $\sigma_{\hat{\mu}_s}$  und  $\sigma_{\hat{\mu}_b}$  ab.
- (x) Vergleichen Sie die Werte Ihres Fits mit denen des ROOT fitting Paketes (benutzen Sie `dataHist.Fit("fitFunk2", "LI");`). Definieren Sie sich zuvor `fitFunk2` als separate Funktion und fixieren Sie wieder  $\mu = 5.0$ ,  $\sigma = 0.5$  und  $\tau = 10.0$ . Stellen Sie diese Funktion wie folgt bereit:

```
TF1 fitFunk2 = TF1("fitFunk2",
  "([0]*1.0/TMath::Sqrt(2*TMath::Pi()*[2]*[2])
   *exp(-0.5*((x-[1])/[2])**2)
   + [3]*(1.0/[4])*exp(-x/[4]))", 0.0, 100.0);
fitFunk2.FixParameter(1, 5.0);
fitFunk2.FixParameter(2, 0.5);
fitFunk2.FixParameter(4, 10.0);
fitFunk2.SetParameter(0, 1000.0);
fitFunk2.SetParameter(3, 1000.0);
```

Beachten Sie, dass die Fitparameter 0 und 3 mit einem Faktor 10 zu multiplizieren sind, da die Binbreite nicht 1 ist.

# Hausaufgaben

## Aufgabe 45 *Messung der Polarisationsasymmetrie*

7 Punkte

Eine wichtige Messung der Paritätsverletzung in der elektroschwachen Wechselwirkung des Standardmodells wurde durchgeführt, indem die inelastische Streuung von polarisierten Elektronen an einem Deuteriumtarget vermessen wurde. Die Paritätsverletzung führt dabei zu einer Differenz im Streuwirkungsquerschnitt zwischen links- und rechts-händig polarisierten Elektronen. Die Meßgröße war dabei die Polarisationsasymmetrie  $\alpha$ , die gegeben ist durch

$$\alpha = \frac{\sigma_R - \sigma_L}{\sigma_R + \sigma_L}, \quad (10)$$

wobei  $\sigma_L$  ( $\sigma_R$ ) die Wirkungsquerschnitte für links- (rechts)-händige Elektronen bei der inelastischen Streuung an einem Deuteriumtarget sind.

Für eine gegebene Luminosität  $L$  ist die Anzahl der beobachteten Streuereignisse bei links- bzw. rechtshändigen Elektronen eine poissonverteilte Zufallsvariable,  $n_L$  bzw.  $n_R$ , mit Mittelwerten  $\nu_L$  bzw.  $\nu_R$ . Die Mittelwerte stehen mit den Wirkungsquerschnitten über  $\nu_L = \sigma_L L$  bzw.  $\nu_R = \sigma_R L$  in Beziehung, wobei die Luminosität  $L$  für die beiden Datensätze identisch ist.

- (i) Benutzen Sie die Tatsache, dass die beobachtete Anzahl von Streuereignissen poissonverteilt ist, um einen "Maximum Likelihood"-Schätzer  $\hat{\alpha}$  für die Asymmetrie  $\alpha$  aufzustellen.  
*Hinweis:* Wie kann man die Wahrscheinlichkeit, dass man  $n_R$  und  $n_L$  Ereignisse beobachtet, abhängig von  $\alpha$  ausdrücken?
- (ii) Benutzen Sie Fehlerfortpflanzung (und nehmen Sie dabei an, dass die Messungen der Streuereignisse für links- bzw. rechtshändige Elektronen unabhängig voneinander sind), um zu zeigen dass die Standardabweichung  $\sigma_{\hat{\alpha}}$  auf diesen Schätzer  $\sigma_{\hat{\alpha}} = \sqrt{\frac{1-\hat{\alpha}^2}{n_{tot}}}$  geschrieben werden kann (wobei  $n_{tot} = n_R + n_L$ ).
- (iii) Bevor das Experiment aufgebaut wurde, wurde erwartet, dass die Asymmetrie ungefähr  $10^{-4}$  betragen würde. Welche Gesamtzahl von Streuereignissen musste man dafür beobachten, um die Asymmetrie mit einer relativen Genauigkeit von 10% zu messen?

Die Anzahl  $k$  der innerhalb einer Minute das Höllental passierenden LKW sei poissonverteilt, d.h. es gilt:

$$f(k; \nu) = \frac{\nu^k}{k!} e^{-\nu}, \quad k = 0, 1, 2, \dots, \quad \nu > 0.$$

Eine Stichprobe mit Umfang  $n = 100$  liefert folgendes Ergebnis für  $k$ :

$k$	absolute Häufigkeit
0	10
1	20
2	30
3	20
4	10
5	5
6	5
7+	0

- (i) Drücken Sie die Likelihoodfunktion für diese Daten aus, abhängig vom wahren Parameter  $\nu$ .
- (ii) Berechnen Sie den ML-Schätzer  $\hat{\nu}$  für eine allgemeine Stichprobe vom Umfang  $n$ , indem Sie  $\ln L$  maximieren. Die Messwerte seien dabei poissonverteilt.
- (iii) Ist der Schätzer erwartungstreu?
- (iv) Berechnen Sie die Varianz  $V(\hat{\nu})$  für den Fall  $n \rightarrow \infty$  unter der Annahme, dass der ML-Schätzer effizient ist. Verwenden Sie die MVB für erwartungstreue Schätzer

$$V[\hat{\nu}] = E \left[ -\frac{\partial^2 \ln L}{\partial \nu^2} \right]^{-1}$$

und dass für einen effizienten Schätzer gilt

$$E \left[ \frac{\partial^2 \ln L}{\partial \nu^2} \right] = \left( \frac{\partial^2 \ln L}{\partial \nu^2} \right)_{\nu=\hat{\nu}}$$

Gegeben seien zwei unabhängige Messungen  $s \pm \sigma$  und  $c \pm \sigma$  von  $\sin \theta$  beziehungsweise  $\cos \theta$ . Nehmen Sie an, dass die gauss'schen Fehler der Messungen gleich sind. Was ist der Maximum-Likelihood Schätzer  $\hat{\theta}$ ?