Statistische Methoden der Datenanalyse

Markus Schumacher

Übung VI

Matthew Beckingham und Markus Warsinsky

10.6.2009

Computerübung

Aufgabe 31 *'Binned' Maximum Likelihood Fit zu $e+e-$ mit Monte Carlo Statistik*

In Aufgabe 24 benutzten Sie die Monte Carlo Methode, um einen Satz von $e^+e^- \to \mu^+\mu^-$ Ereignissen zu erstellen, welcher gemäß des differentiellen Wirkungsquerschnitts

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

verteilt war, wobei θ und ϕ für die Winkel des μ^+ Teilchens in Kugelkoordinaten stehen. Der Winkel θ wird relativ zur z-Achse und der Winkel ϕ 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.

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

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

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

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

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

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

Dies ist eine Funktion, welche maximiert werden muss, um die zur Erstellung des Monte Carlo Datensatzes verwendeten Parameter $\vec{\theta}$ zu finden.

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

(i) Öffnen Sie die RooT-Datei, welche Sie in Aufgabe 24 angefertigt haben. Erstellen Sie eine Schleife über alle Ereignisse innerhalb des `TTree`, welcher sich in der Datei befindet, und füllen Sie ein Histogramm mit den Werten von $\cos\theta$ für jedes Ereignis. Ein Beispiel, wie dies zu tun ist, finden Sie in dem Beispielmakro `2DBinLike_i.C`.

(ii) Definieren Sie eine `TF1` Funktion der Form

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

welche in dem Maximum Likelihood Fit benutzt werden soll. Sehen Sie sich Ihre Lösung aus Aufgabe 24 an, falls Sie hierfür Hilfestellung benötigen.

(iii) Definieren Sie - analog zu dem Maximum Likelihood Fit aus Aufgabe 25 - eine Schleife, um über mögliche Werte von α zu iterieren. Definieren Sie innerhalb dieser Schleife eine weitere Schleife über möglichen Werte von β.

(iv) Erstellen Sie für jeden Wert von α_i und β_i eine Schleife über die Bins in Ihrem $\cos\theta$ Histogramm und berechnen Sie den log-Likelihood Wert. Gehen Sie wie folgt vor:

- Erstellen Sie eine Schleife von i=1 (das niedrigste Bin des Histogramms - Bin 0 - ist das Underflow Bin) bis i=`Hist->GetNbinsX()`
- Die Anzahl an Einträgen im $i$ten Bin erhalten Sie mit der `TH1` Funktion `TH1::GetBinContent(Int i)`
- Um Ihre Funktion über die Binbreite zu integrieren benutzen Sie die `TF1` Funktion `TF1::Integral(Double_t a, Double_t 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 `TH1::GetBinLowEdge(Int_t i)`, welche die untere Grenze des $i$ten Bins ausgibt.
- Summieren Sie die log-Likelihoodfunktion für alle Bins des Histogramms auf.

(v) Füllen Sie für jedes α_i und β_i den log-Likelihood Wert in einen zweidimensionalen Graph. Dies funktioniert analog zu einem eindimensionalen Graphen:

```
TGraph2D* likeGraph = new TGraph2D();
```

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

```
TGraph2D::SetPoint(Int_t pointNumber, Double_t x, Double_t y, Double_t z)
```

(vi) Zeichnen Sie den Graphen (log-Likelihood gegen α und β) unter Benutzung von

```
likeGraph->Draw("surf1");
```

wobei die Option `"surf1"` 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.

(vii) Ermitteln Sie aus den erzeugten Graph eine Abschätzung für $\hat{\alpha}$ und $\hat{\beta}$ und deren Standardabweichungen $\sigma_{\hat{\alpha}}$ und $\sigma_{\hat{\alpha}}$.

(viii) Vergleichen Sie die Werte Ihres Fits mit denen des RooT Fitting Paketes, indem Sie die Histogramm Fit Funktion

```
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 32 *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:

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

wobei entsprechend μ_s und μ_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:

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

Daher ist die log-Likelihoodfunktion gegeben durch

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

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 geschieht in folgender Weise (Ein Beispiel dafür findet sich im ROOT-Makro `GausExp_i.C`.):

(i) Definieren Sie eine `TF1` Funktion, welche die Summe einer Gauss- und einer Exponentialfunktion darstellt:

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

in dem Intervall $0.0 < x < 100.0$. Wählen Sie für die Parameter $\mu = 5.0$, $\sigma = 0.5$ und $\tau = 10.0$.

(ii) Benutzen Sie die `TF1` Funktion `GetRandom`, um einen Satz von 1000 Werten der obigen Verteilungsfunktion zu erhalten und füllen Sie diese in ein Array.

Als nächstes führen Sie einen erweiterten Maximum Likelihood Fit auf dem Datensatz durch, welchen Sie erstellt haben. Hierzu sind folgende Schritte erforderlich:

(iii) 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 $\mu = 2.0$, $\sigma = 0.5$ und $\tau = 10.0$ zu setzen (man könnte diese Parameter variabel lassen, jedoch müsste man dann einen fünfdimensionalen Fit durchführen!).

(iv) Erstellen Sie, analog zu Aufgabe 31, eine Schleife über mögliche Werte des Parameters μ_s. Definieren Sie innerhalb dieser Schleife eine weitere Schleife über mögliche Werte von μ_b.

(v) Erstellen Sie für jeden Wert von μ_s und μ_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 `TF1::Eval(Double_t x)` den Wert der Funktion am Punkt x zu berechnen.

(vi) Befüllen Sie ein `TGraph2D` Objekt mit den aufsummierten log-Likelihood Werten für jeden Wert von μ_s und μ_b.

(vii) 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.

(viii) Vergleichen Sie die Werte Ihres Fits mit denen des RooT fitting Paketes.