

Übung zur Vorlesung “Statistische Methoden der Datenanalyse” H. Kolanoski, A. Schälicke – SS 2008

Übung 7

7.1 ML-Bestimmung einer Zerfallswinkelverteilung

Bei dem Zerfall eines Teilchens trete ein Zerfallsprodukt unter dem Polarwinkel θ relativ zu der Spin-Richtung des zerfallenden Teilchens auf. Ein theoretisches Modell sagt folgende Abhängigkeit der Wahrscheinlichkeitsdichte von $z = \cos \theta$ voraus:

$$f(z|a) = \frac{1}{2}(1 + az) \quad (1)$$

Die Beobachtung von 20 Zerfällen ergab folgende Werte für z :

z[1] = -0.387	z[11] = 0.534
z[2] = 0.330	z[12] = -0.030
z[3] = 0.851	z[13] = 0.111
z[4] = -0.361	z[14] = -0.676
z[5] = 0.503	z[15] = 0.531
z[6] = 0.697	z[16] = -0.767
z[7] = 0.565	z[17] = 0.064
z[8] = 0.758	z[18] = 0.065
z[9] = 0.207	z[19] = 0.909
z[10] = -0.679	z[20] = 0.246

Bestimmen Sie einen Schätzwert für a mit der Maximum-Likelihood-Methode. (Daten sind auch als ASCII-Datei auf der Webseite der Übung verfügbar.)

7.2 Winkelsumme

Es werden die drei Winkel eines Dreieckes gemessen. Die Messwerte sind $\alpha = 82^\circ$, $\beta = 40^\circ$ und $\gamma = 61^\circ$ und die Messungenauigkeiten seien alle gleich (z.B. $\sigma = 1^\circ$).

Bestimmen Sie verbesserte Schätzungen für die drei Winkel unter Verwendung der Nebenbedingung $\alpha + \beta + \gamma = 180^\circ$ und geben Sie die Kovarianzmatrix für diese Schätzwerte an.

b.w.

Hinweise

- Ein Funktion kann in Python über einen `def`-Ausdruck definiert werden

```
def f(x):  
    # do some calculation  
    w = sin(x)  
    return w
```

einfache Funktionen schreibt man kürzer durch einen `lambda`-Befehl:

```
f = lambda x : sin(x)
```

- Im `scipy.optimize` Packet werden eine Reihe von Minimierungsfunktionen bereitgestellt. Soll ein Maximum bestimmt werden, muss das Vorzeichen der Funktion erst invertiert werden, z.B.:

```
import scipy  
f = lambda x: -special.jv(1.,x)  
x_max = optimize.fminbound(f,0,6)
```

findet das lokale Maximum der Besselfunktion $J_{\nu=1}(x)$.

- Einlesen einer ASCII-Datendatei geschieht am einfachsten durch:

```
import pylab  
x = pylab.load('sample.dat',unpack=True)
```

In modernen Pythoninstallationen könnte auch `numpy.loadtxt` verwendet werden.