Praktische Übung: Computergestützte Datenauswertung

Institut für Experimentelle Teilchenphysik

Prof. Dr. G. Quast
Dr. Andreas Poenicke
`http://comp.physik.kit.edu`

SS16 – Blatt 05
zu bearbeiten am 27.6 (Gr. a) bzw. 4.7 (Gr. b)

Aufgabe 5.1 Volumen von Hyper-Kugeln mit der Monte Carlo-Methode — Testat

Mit Hilfe der Monte Carlo-Methode können recht einfach hochdimensionale, bestimmte Integrale näherungsweise numerisch berechnet werden. In dieser Aufgabe soll das Volumen V_d von (Hyper-)Kugeln in Abhängigkeit von der Anzahl der Dimensionen d berechnet werden.

Als Basis schauen Sie sich das Script `pi.py` an: darin wird ein einfaches Verfahren zur Bestimmung des Verhältnisses der Flächen eines Viertelkreis und des Einheitsquadrats benutzt. Dazu werden zufällig Punktepaare in der Fläche verteilt, und der Anteil der Punkte innerhalb des Kreises wird gezählt.

a) Erweitern Sie diesen Code, so dass Sie mit dem Verfahren das d-dimensionale Kugelvolumen für $d \in [1, 10]$ berechnen können. Arbeiten Sie zunächst mit 10'000 gleichmäßig im Hyperkubus mit Kantenlänge 1 verteilten Punkten. Berechnen Sie aus dem so bestimmten Volumen des Kugelsegments im Einheits-Hyperkubus das Volumen der vollen Hyperkugel.

b) Bestimmen Sie jeweils die statistische Unsicherheit σ_{V_d} der Ergebnisse. Nutzen Sie dazu aus, dass bei N Versuchen die Anzahl N_{in} der innerhalb der Kugel liegenden Punkte einer Binomialverteilung $B(N_{in}; N, p)$ folgt. Die Unsicherheit ergibt sich aus der bekannten Varianz der Binomialverteilung mit der Näherung $p \simeq \hat{p} = N_{in}/N$.

c) Tragen Sie Ihre Ergebnisse für V_d und die Unsicherheit σ_{V_d} in ein Diagramm ein. Die Vorlage `VKugel.py` enthält die Funktion `VolShere(d, r=1.)`, die das mit mathematischen Integrationsmethoden berechnete Volumen von Hyperkugeln angibt. Tragen Sie auch dieses „theoretische“ Ergebnis in Ihre Grafik ein.

Aufgabe 5.2: Korrelation — Testat

In dieser Aufgabe soll die Verteilung der Bin-Inhalte eines Histogramms sowie die Korrelation zwischen ihnen näher untersucht werden. Dazu gibt es eine Vorlagen, die Sie ergänzen können (`Covariance.py`).

a) Untersuchung der Häufigkeitsverteilung von Bin-Inhalten

Füllen Sie als Experiment $N = 100$ in $[0, 1]$ gleichverteilten Zufallszahlen in ein Histogramm mit $n_b = 5$ Bins zwischen $[0, 1]$. Die Bins enthalten nun im Mittel je 100/5 Einträge, deren Verteilung weiter untersucht werden soll. Wiederholen Sie dazu dieses Experiment 10'000 mal und erzeugen Sie für jedes Bin i ein Histogramm der dabei gefundenen Zahlen von Einträgen n_i in diesem Bin.

Welche Verteilung der Bin-Inhalte erwarten Sie? Vergleichen Sie, indem Sie die erwartete Verteilung einzeichnen.

b) Untersuchung der Korrelation von Bin-Inhalten

Berechnen Sie die Kovarianz und den Korrelationskoeffizienten der Anzahl von Einträgen in zwei Bins: Verallgemeinern Sie dazu den Programmcode aus Aufg. 3.2 von Blatt 3 so, dass Kovarianz und Korrelationskoeffizient von zwei Arrays berechnet werden.

Bestimmen Sie den Korrelationskoeffizenten der in **a)** gefundenen Inhalte n_i und n_j zweier Bins, also z.B. o.B.d.A. Bin 2 und Bin 4.

Eine anschauliche Methode, Korrelationen zwischen den Bin-Inhalten zu untersuchen, besteht darin, zwei-dimensionale Histogramme zu füllen. Wenn Sie sich die aus dem zwei-dimensionalen Histogramm bestimmten Korrelationskoeffizienten ausgeben lassen, können Sie Ihren eigenen Code zur Berechnung validieren. Siehe dazu die `Python`-Funktion `hist2dstat` in der `Python`-Vorlage `Histogram.py`.

Stellen Sie die Einträge in den Bins 2 und 4 als zweidimensionales Histogramm dar.

Aufgabe 5.3: Anpassung von Modellen an Daten mit `kafe` — Testat

Obwohl sich viele Anpassungsprobleme und insbesondere die linearen auch analytisch lösen lassen, setzt man in der Praxis Software-Werkzeuge ein, die die Verwaltung von Daten, Modellen und das Durchführen der eigentlichen Anpassung mit numerischen Methoden bewerkstelligen.

Ein solches Werkzeug ist `kafe`, s. `http://www.ekp.kit.edu/~quast/kafe` . Verschaffen Sie sich zunächst mit Hilfe der online-Dokumentation einen Überblick über die Funktionalität. Als einfaches Beispiel und Grundlage für diese Aufgabe dient die Vorlage `fitexample_kafe.py`, die Sie leicht für die Aufgabenstellung unten anpassen können (s. Abb. 1. Nach jedem Ausführen einer Anpassung erzeugt `kafe` eine Anzahl von Protokoll-Dateien im Unterverzeichnis `.kafe`, deren Namen aus dem Namen des verwendeten Datensatzes (`basename` und einem evtl. spezifizierten Namen für die jeweils durchgeführte Anpassung (`fitname`) mit der Endung `.log` abgeleitet werden.

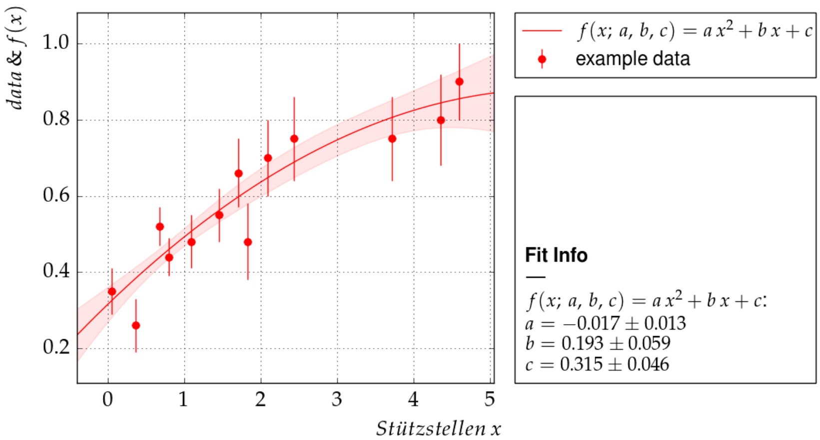

Abbildung 1: Grafische Ausgabe von `fitexample_kafe.py`: Datenpunkte mit Unsicherheit und angepasster Parabel.

Hinweis: `kafe` läuft nicht unter `idle`, sondern muss auf der Kommandozeile gestartet werden !

a) Ergänzen Sie die Vorlage um das Einlesen der Daten aus der Datei `FittingExercise.dat` und erzeugen Sie aus diesen eine Instanz eines `kafe`-Datensatzes (`kafe.Dataset()`). Bei den Daten handelt es sich um eine mit gaußförmigen Unsicherheiten behaftete, simulierte Messung an festen Stützstellen. Passen Sie zunächst eine Gerade in der üblichen Parametrisierung y=ax+b an. Erstellen Sie dazu die entsprechende Fit-Funktion, analog zu der Parametrisierung einer Parabel (`poly2(x, a, b, c)`) in der Vorlage, und führen Sie die Anpassung durch. Studieren Sie die `log`-Datei bzw. die Bildschirmausgabe. Sie werden feststellen, dass die χ^2-Wahrscheinlichkeit akzeptabel ist, die beiden Parameter a und b der Anpassung aber eine große Korrelation zeigen. Um dies näher zu studieren, schauen Sie sich als nächstes ein grafische Darstellung der Korrelationen in Form von Kovarianz-Ellipsen an. Nutzen Sie dazu die Methode `kafe.Fit.plot_correlations()`.

b) Ändern Sie nun die Parametrisierung, so dass die Gerade um den Schwerpunkt der x-Werte dargestellt wird, also gemäß $y = s(x - \bar{x}) + c$. Führen sie wie in **a)** die Anpassung durch und vergleichen Sie. Welche Korrelation der Parameter s und c beobachten Sie? Sehen Sie sich dazu auch die Darstellung der Kovarianz-Ellipsen an.

Vergleichen Sie in die grafische Darstellung des angepassten Modells und die Lage des Bands der Unsicherheiten[1]. Überlegen Sie, welche Parametrisierung bevorzugen Sie für die Angabe Ihres Ergebnisses?

Hinweis: Mit dem Rechnernamen `fphctssh.physik.uni-karlsruhe.de` können Sie von überall aus mittels ssh/scp Programm per Netzwerk auf einen Poolrechner zugreifen.

---

[1] In `kafe` wird die vollständige Kovarianzmatrix der angepassten Parameter $\vec{p}$ benutzt, um mittels Fehlerfortpflanzung für jeden x-Wert die Unsicherheit auf die Modellvorhersage $y(x; \vec{p})$ zu berechnen.