Albert-Ludwigs-Universität Freiburg | Sommersemester 2009

Statistische Methoden der Datenanalyse

Markus Schumacher

Übung II

Matthew Beckingham und Markus Warsinsky

6.5.2009

Computerübung

Aufgabe 11 *Erste Schritte in ROOT*

Beachten Sie die ausgehändigte Einführung in ROOT und machen Sie sich mit ROOT vertraut.

Aufgabe 12 *Erzeugung gleichverteilter Zufallszahlen*

Die Monte-Carlo-Methode, die in der kommenden Vorlesung besprochen wird, ist hochgradig abhängig von der Erzeugung von Zufallszahlen. Im Prinzip liessen sich wahre Zufallszahlen mittels physikalischer Prozesse, wie zum Beispiel radioaktive Zerfälle erzeugen. In der Praxis werden allerdings deterministisch erzeugte *Pseudozufallszahlen* benutzt, da diese von einem Rechner erzeugt werden können.

Ein möglicher Algorithmus ist der *linear kongruente Generator*, der gleichverteilte Zahlen u_j zwischen 0 und 1 erzeugt, und nach folgender Rekursionsformel arbeitet:

$$i_j = (a \cdot i_{j-1} + c) \mod m, \qquad u_j = i_j/m,$$

wobei a,c und m positive ganze Zahlen sind.

(i) In `/home/warsinsk/stat/ueb2/` befindet sich ein ROOT-Makro `aufgabe12_i.C`, das diesen Zufallsgenerator implementiert. Kopieren Sie das Makro in ein von Ihnen erstelltes Verzeichnis, öffnen Sie es in einem beliebigen Editor (z. B. `emacs`) und versuchen Sie, die Funktionsweise zu verstehen.

(ii) Lassen Sie das Makro laufen, welches ein Histogramm der erzeugten Zahlen ausgibt. Stimmen die angezeigten Werte für Mittelwert und Varianz[1] mit Ihren Erwartungen überein? Sehen die Zahlen wirklich zufallsverteilt aus?

(iii) Nun soll mit einem einfachen Test die Güte dieses Zufallsgenerators überprüft werden. Dazu stellt man aufeinander folgende Zufallszahlen, z.B. gerade und ungerade in der Abfolge, durch einen Punkt in der xy-Ebene dar, dessen Koordinaten gegeben sind durch $(x,y)_i = (u_{2i+1}, u_{2i})$, $i \in \{0,1,2,3,\ldots\}$. Modifizieren Sie das Makro so, dass es einem solchen sogenannten „Scatterplot“ darstellt. Benutzen Sie dazu die Klasse `TGraph`, um `ngen/2` Punkte darzustellen und benutzen Sie jede Zufallszahl nur einmal.

Ist der verwendete linear kongruente Generator mit den im Beispielskript verwendeten Werten ein guter Zufallszahlengenerator? Benutzen Sie stattdessen die Vorgabewerte der Funktion `LinKongr`. Ist dies eine bessere Einstellung?

(iv) Vergleichen Sie jetzt diesen einfachen linear kongruenten Generator mit dem in ROOT implementierten Zufallsgenerator `gRandom->Rndm();`. Ist dieser besser oder schlechter als der selbst programmierte Generator? In den folgenden Aufgaben werden wir `gRandom->Rndm();` verwenden.

Aufgabe 13 *Produkt zweier gleichverteilter Zufallszahlen*

In Aufgabe 8 haben Sie gezeigt, dass die Wahrscheinlichkeitsdichtefunktion des Produktes zweier gleichverteilter unabhängiger Größen zwischen 0 und 1 gegeben ist durch $-\ln z$, $z = x_1 x_2$.

(i) Schreiben Sie ein ROOT-Makro, bzw. modifizieren Sie das in der vorherigen Aufgabe benutzte ROOT-Makro, um zwei Zufallszahlen zu generieren, ihr Produkt zu bilden in ein Histogramm zu füllen und es schliesslich auszugeben.

[1] Das angezeigte RMS ist in ROOT tatsächlich die Wurzel aus der Varianz.

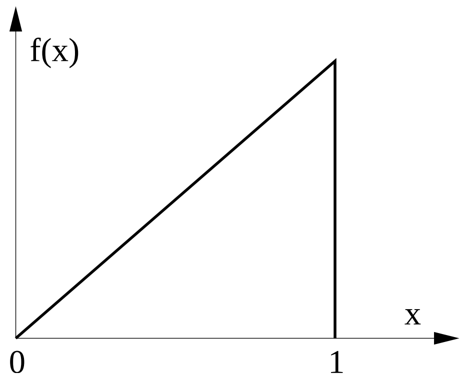
(ii) Normieren Sie das Histogramm auf die Gesamtanzahl der Einträge.

(iii) Zeichnen Sie die erwartete Wahrscheinlichkeitsdichtefunktion ein, indem Sie die ROOT-Klasse `TF1` benutzen. Warum ist hierbei die Binbreite des Histogramms zu beachten?

Aufgabe 14 *Gleichverteilung im Dreieck*

Wie in der nächsten Vorlesung besprochen wird, können Zufallszahlen benutzt werden, um Funtionen numerisch zu integrieren. Zumeist wird dabei zunächst versucht, im Definitionsbereich einer Funktion gleichmäßig verteilte Zufallszahlen zu erzeugen.

Nehmen wir an, ein solcher Bereich sei durch ein Dreieck mit Steigung 1 gegeben.



Erzeugen Sie mit einem ROOT-Makro Zufallspunkte in diesem Wertebereich.

(i) Würfeln Sie dazu zunächst die x-Koordinate gleichverteilt zwischen 0 und 1, würfeln eine andere Zufallszahl zwischen 0 und 1 und transformieren diese so, dass beim Bilden eines (x,y)-Wertepaares die y-Koordinate innerhalb des Dreiecks liegt. Prüfen Sie nach, inwiefern die so gebildeten Punkte wirklich gleichmäßig innerhalb des Dreiecks verteilt sind. Benutzen Sie dazu

a) eine Punktwolke wie in Aufgabe 12 mittels `TGraph`,

b) ein zweidimensionales Histogramm mittels `TH2D`.

(ii) Gibt es einen Trick, wie man ohne grossen Aufwand im Dreeieck gleichverteilte Zufallspunkte erzeugen kann?

Aufgabe 15 *Zentraler Grenzwertsatz*

Mit dem ROOT-Makro `aufgabe_15_i_bis_v.C` kann der Zentrale Grenzwertsatz demonstriert werden. Das Makro befindet sich in `/home/warsinsk/stat/ueb2/`.

Es werden eine oder mehrere gleichförmig verteilte Zufallszahlen erzeugt und ihre Summe als Histogramm dargestellt. Zusätzlich kann eine Gaußfunktion mit vorgegebenem Mittelwert und vorgegebener Breite eingezeichnet werden.

(i) Laden Sie das Makro und versuchen Sie, die Funktionsweise zu verstehen.

(ii) Berechnen Sie die Verteilung für eine Zufallsvariable. Stimmen die angezeigten Werte für Mittelwert und Varianz[2] mit Ihren Erwartungen überein?

(iii) Zeichnen Sie eine Gaußverteilung mit den erwarteten Werten für Mittelwert und Breite ein (setzen Sie dazu `bool DrawGauss = true;`).

(iv) Lassen Sie zwei gleichförmig verteilte Zufallszahlen addieren (`int NVar = 2;`). Beobachten Sie wieder Mittelwert und Varianz.

(v) Erhöhen Sie `NVar` weiter. Wieviele gleichförmig verteilte Zufallsvariablen sollte man mindestens addieren, um eine gute Annäherung an die Gaußfunktion zu erhalten?

[2]Das angezeigte RMS ist in ROOT tatsächlich die Wurzel aus der Varianz.