Rechnernutzung in der Physik

Institut für Experimentelle Teilchenphysik
Institut für Theoretische Teilchenphysik

Prof. Dr. G. Quast, Prof. Dr. M. Steinhauser
Dr. A. Mildenberger, Dr. Th. Chwalek
http://comp.physik.kit.edu

WS2018/19 – Blatt 07
Prog. bis Di., 8.01.2019

Einführung in die Anwendung von ROOT

ROOT (http://root.cern.ch) ist ein Programmpaket zur grafischen Auswertung insbesondere großer Datenmengen. Die von ROOT zur Verfügung gestellten Klassen können entweder in eigene Programme eingebunden oder interaktiv in der „ROOT shell" aufgerufen werden. Zusätzlich gibt es die `Python`-Anbindung `pyroot`, die die Nutzung einzelner ROOT-Klassen aus `Python` erlaubt.

ROOT-Tutorial — für Bachelorstudiengang

Um sich mit ROOT vertraut zu machen, laden Sie die Datei `divingROOT.zip` in Ihr Arbeitsverzeichnis und entpacken Sie sie. Im Kurs arbeiten wir mit der Version 5.34 von ROOT, die auch als Standard-Paket unter Ubuntu installiert werden kann. Arbeiten Sie Kapitel 1 – 3 und 8 des Tutorials `diving_into_ROOT.pdf` durch. Alle Beispiele finden Sie nach Entpacken der `zip`-Datei im Unterverzeichnis `macros`. Bitte machen Sie sich auch mit den im Tutorial angegebenen Informationsquellen vertraut, insbesondere mit den Klassendefinitionen im `Reference-Guide` und dem `ROOT User's Guide`.

Aufgabe 13: Gauß-Funktion in ROOT * Bachelor

Nachdem Sie das Tutorial durchgearbeitet haben, sollten Sie nun in der Lage sein, eine normierte Gauß-Funktion mit vorgegebenem Mittelwert von $\mu = 5.0$ und Standardabweichung $\sigma = 1.5$ in ROOT darzustellen. Als Vorlage zu dieser Aufgabe dienen die Dateien `PlotGauss.C` bzw. `PlotGauss-pyroot.py`, die Sie erweitern können.
Als Beispiel sollten Sie auch die Dateien `PlotFermi.C` bzw. `PlotFermi-pyroot.py` anschauen, die die Darstellung einer Kurvenschar als ROOT C++-Makro bzw. als `pyroot`-Skript illustrieren (s. auch Anhang).
Ob Sie Code in `C++/C` oder `Python` verwenden wollen, bleibt Ihnen überlassen. Wie schon in der Vorlesung angemerkt, ist für große, zeitkritische Projekte die Erstellung von C-Code unbedingt vorzuziehen, für die Übungsaufgaben dieses Kurses und manche spätere praktische Anwendung ist aber `Python` ausreichend.
Zeichnen Sie auch die Ableitung und das Integral der von Ihnen implementierten Funktion im vorgegebenen Intervall. Sehen Sie dazu unter http://root.cern.ch/root/html/TF1.html nach, welche `Draw()`-Funktionen es zur Klasse `TF1` gibt.

Aufgabe 13L: Statistische Auswertung von Prüfungsnoten * für Lehramtsstudierende

a: Einlesen der Daten in sinnvolle Datenstruktur

Die auszuwertenden Daten liegen in Form der `csv`-Datei `studierende.csv` vor.[1] Die erste Zeile erklärt den Inhalt jeder „Spalte": Geschlecht, ECTS-Punkte und Note (mit Zehntelwerten und multipliziert mit 100). Lesen Sie diese Daten in eine sinnvolle Datenstruktur ein.

---

[1] „`csv`" steht für „comma separated value".

b: Darstellung der Daten

Stellen Sie die Häufigkeitsverteilung der Noten und ECTS-Punkte sowie den Zusammenhang zwischen beiden Größen graphisch dar.

c: Berechnung von Mittelwerten und Quantilen

Errechnen Sie mithilfe selbstgeschriebener Routinen zu dem vorliegenden Datensatz den Mittelwert, den Median, d.h. das 50%-Quantil, und das 10%-Quantil jeweils für alle Studierende sowie getrennt für weibliche und männliche Studierende.
Hinweis: Zur Berechnung der Quantile ist es hilfreich, die Sortierfunktion `numpy.sort` zu verwenden.

d: Berechnung der Stichprobenvarianzen

Errechnen Sie wie im vorherigen Aufgabenteil mithilfe selbstgeschriebener Routinen jeweils die Stichprobenvarianzen mit und ohne Bessel-Korrektur.

e: Klasseneinteilungen und Signifikanz

Beurteilen Sie die Signifikanz der Abweichungen in den mittleren Noten zwischen weiblichen und männlichen Studierenden. Wie stark (in kombinierten Standardabweichungen $\sqrt{\sigma_w^2 + \sigma_m^2}$) unterscheiden sich die beiden Mittelwerte? Benutzen Sie hierzu die Stichprobenvarianzen mit Bessel-Korrektur als Schätzwerte für die Varianzen.

Aufgabe 14: Transformierte Zufallszahlen *

Gehen Sie von gleichverteilten Zufallszahlen im Intervall $r \in [0, 1[$ aus. Wenden Sie auf diese Zufallszahlen die unten angegebenen Funktionen $t_i(r)$, $i = 1, \ldots, 4$ an und histogrammieren Sie die Verteilungen für 100'000 Zufallszahlen. Welchen Verteilungsdichten g_i folgen jeweils die t_i? Zeichnen Sie die passend normierten – d.h. an die Binbreite und Anzahl der Zufallszahlen angepassten – Verteilungsdichten in die Histogramme ein.
1) $t_1(r) = r^2$
2) $t_2(r) = \exp(r)$
3) $t_3(r) = \tan(r)$
4) $t_4(r) = \log(1. + r)$

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