

# Rechnernutzung in der Physik

Institut für Experimentelle Teilchenphysik  
Institut für Theoretische Teilchenphysik

Prof. Dr. M. Steinhauser, Dr. M. Giffels, Dr. R. Wolf  
Dr. A. Mildenerger

WS2015/16 – Blatt 10

<http://comp.physik.kit.edu>

Prog.: Di., 19.01.2015 / Ausarb.: Di., 15.01.2016

---

## Bitte beachten Sie:

Dieses Übungsblatt hat besondere Abgabezeiten für Ausarbeitung und Programmtestat.

## Bitte beachten Sie weiterhin:

Melden Sie sich unter <https://campus.studium.kit.edu/> zu diesem Kurs an. Dies ist zur Leistungsverbuchung erforderlich.

---

## Korrelationen und Variablentransformation

Die Aufgaben dieses Blattes können Sie wahlweise in Python oder ROOT lösen.

### Aufgabe 19: Korrelationen

### Ausarbeitung

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

#### a) Untersuchung der Häufigkeitsverteilung von Bin-Inhalten

Füllen Sie ein Histogramm mit  $n_b = 5$  Bins mit  $N = 100$  in  $[0, 1]$  gleichverteilten Zufallszahlen (analog Aufgabe 17 vom letzten Blatt). Die Bins enthalten nun im Mittel je  $100/5$  Einträge, deren Verteilung untersucht werden soll. Erzeugen Sie dazu je ein weiteres Histogramm, in das Sie in jedem Experiment den Inhalt  $n_i$  des Bins  $i$  füllen und wiederholen Sie die Prozedur 10'000 mal.

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. 6 von Blatt 3 so, dass Kovarianz und Korrelationskoeffizient von zwei Arrays berechnet werden. Fügen Sie Ihrer Ausarbeitung den (kommentierten) Programmcode bei. Geben Sie den Korrelationskoeffizienten der Inhalte zweier Bins  $n_i$  und  $n_j$ , also z.B. o.B.d.A. Bin 2 und Bin 4, an.

Eine anschauliche Methode, Korrelationen zwischen den Bin-Inhalten zu untersuchen, besteht darin, zweidimensionale Histogramme zu füllen<sup>1</sup>. Wenn Sie sich die aus dem zweidimensionalen Histogramm bestimmten Korrelationskoeffizienten ausgeben lassen, können Sie Ihren eigenen Code zur Berechnung validieren. Siehe dazu die Python-Funktion `hist2dstat` in der Python-Vorlage bzw. die Methode `GetCorrelationFactor()` der ROOT-Klasse `TH2F`.

---

<sup>1</sup>In ROOT verwendet man dazu die Histogramm-Klasse `TH2F` (siehe auch ANHANG). Wenn Sie Python vorziehen, finden Sie eine Implementierung eines zweidimensionalen Histogramms in der Vorlage.

Stellen Sie die Einträge in den Bins 2 und 4 als zweidimensionales Histogramm dar und fügen Sie es der schriftlichen Ausarbeitung bei. Geben Sie eine (qualitative) Erklärung des Ergebnisses.

### c) Verteilung von Bin-Inhalten (2. Teil)

Modifizieren Sie Ihren Code so, dass die gleichverteilten Zufallszahlen in ein Histogramm mit  $n_b = 100$  Bins eintragen werden. Füllen Sie in einer Schleife 10'000 mal je 2000 Zufallszahlen. Der Erwartungswert für jedes Bin ist nun wieder 20, genau wie oben. Vergleichen Sie die Verteilung der Bin-Inhalte in Bin 2 durch Überlagerung von geeignet normierten Binomial- und Poisson-Verteilungen. Fügen Sie Ihrer Ausarbeitung einen Ausdruck bei.

### d) Untersuchung der Korrelation von Bin-Inhalten (2. Teil)

Kehren Sie zurück zur ursprünglichen Version des Programms mit  $n_b = 5$ . Statt einer festen Anzahl von Einträgen verwenden Sie aber nun jeweils eine Poisson-verteilte Anzahl von Einträgen mit Erwartungswert  $\mu = 100$ , `Poisson(N;mu)` statt der festen Anzahl  $N = 100$  (in Python mittels `numpy.random.poisson(mu, isize)`, in ROOT mit der Methode `Poisson(mu)` der Klasse `TRandom`).

Welcher Verteilung folgen die beobachteten Anzahlen im zweiten Bin? Bestimmen Sie wieder den Korrelationskoeffizienten. *Anm.: Diese Variante entspricht einem typischen Experiment, in dem Ereignisse über eine feste Zeitspanne registriert werden.*

## Aufgabe 20: Transformierte Zufallszahlen

## Programmtestat

Gehen Sie von gleichverteilten Zufallszahlen im Intervall  $r \in [0, 1[$  aus. Wenden Sie auf diese Zufallszahlen die unten angegebenen Funktionen  $t_i(r)$ ,  $i = 1, \dots, 4$  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)  $t_4(r) = \log(1. + r)$

---

## ANHANG: Hilfen zu ROOT

Ein Histogramm mit Variablennamen `id` kann folgendermaßen zurückgesetzt werden:

```
id->Reset();
```

Der Bin-Inhalt eines Histogramms der Klasse `TH1F` kann mit der Methode `GetArray()` ausgelesen werden, die einen Zeiger auf ein Feld zurück liefert:

```
Float_t *content = id->GetArray();
```

Der Inhalt des Feldes ist wie folgt:

```
content[0]          number of underflows
content[1,...,n]    number of entries in bins 1,...,n
content[n+1]        number of overflows
```

Zahlreiche mathematische Funktionen finden sich in der ROOT-Klasse `TMath`.

Das Erzeugen einer Instanz eines zweidimensionalen Histogramms geht so:

```
TH2F *hscat = new TH2F("hscat","some title"
```

```
NumberBinsX,XMin,XMax,NumberBinsY,YMin,YMax);
```

Gefüllt wird TH2F analog zum eindimensionalen Fall mit der Methode `Fill`: `hscat->Fill(x,y)`;

Die Korrelation bzw. Kovarianz lässt sich in ROOT ausgeben mit

`id->GetCorrelationFactor()` bzw. `id->GetCovariance()`.

Zusätzlich zu den zweidimensionalen Histogrammen gibt es in ROOT das sogenannte *profile histogram*, Klasse `TProfile`. Damit lässt sich die Korrelation deutlicher darstellen:

```
TProfile *hprof=new TProfile("hprof","title",NBinsX,XMin,XMax,YMin,YMax);
```

Das Füllen funktioniert genau wie im Fall eines zweidimensionalen Histogramms.