# Computergestütztes wissenschaftliches Rechnen SoSe 2004

Alexander K. Hartmann, Universität Göttingen

5. Mai 2004

## 3.2 Inversions Methode

Gegeben: `drand48()` generiert gleichverteilte Zahlen in $[0,1)$, bezeichnet mit $U$.

Ziel: Zufallszahlen $Z$ gemäß W.-Dichte $p(z)$ mit Verteilung

$$P(z) \equiv \mathrm{Prob}(Z \leq z) \equiv \int_{-\infty}^{z} dz' p(z') \tag{1}$$

Idee: suche Funktion $g()$ mit $Z = g(U)$. Annahme: $g$ is streng monoton steigend, d.h. kann invertiert werden $\rightarrow$

$$P(z) = \mathrm{Prob}(Z \leq z) = \mathrm{Prob}(g(U) \leq z) = \mathrm{Prob}(U \leq g^{-1}(z)) \tag{2}$$

Mit
1) $\mathrm{Prob}(U \leq u) = F(u) = u$ wenn $U$ gleichverteilt in $[0,1)$
2) Identifizierung $u$ mit $g^{-1}(z)$
$\Rightarrow P(z) = g^{-1}(z) \Rightarrow g(z) = P^{-1}(z)$.

Funktioniert, wenn $P$ (evtl. numerisch) berechnet und invertiert werden kann.

Beispiel: Exponentialverteilung: $p(z) = \lambda \exp(-\lambda z)$, also $P(z) = 1 - \exp(-\lambda z)$.
Damit: Erzeuge gleichverteilte Zahl $U$ und wähle $Z = -\ln(1-U)/\lambda$.

Programm `exponential.c`

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>

#define NUM_BINS 100

int main(int argc, char *argv[])
{
  int histo[NUM_BINS];                                      /* histogram */
  int bin;
  double start_histo, end_histo;                   /* range of histogram */
  double delta;                                        /* width of bin */
  int t;                                               /* loop counter */
  int num_runs;                       /* number of generated random numbers */
```

```
  double lambda;                                /* parameter of distribution */
  double number;                                         /* generated number */

  num_runs = atoi(argv[1]);                               /* read parameters */
  sscanf(argv[2], "%lf", &lambda);

  for(t=0; t<NUM_BINS; t++)                          /* initialise histogram */
      histo[t] = 0;
  start_histo = 0.0; end_histo = 10.0/lambda;
  delta = (end_histo - start_histo)/NUM_BINS;

  for(t=0; t<num_runs; t++)                                     /* main loop */
  {
    number = -log(drand48())/lambda;     /* generate exp-distr. number */
    bin = (int) floor((number-start_histo)/delta);
    if( (bin >= 0)&&(bin < NUM_BINS))                      /* inside range ? */
       histo[bin]++;                                          /* count event */
  }

  for(t=0; t<NUM_BINS; t++)                    /* print normalized histogram */
      printf("%f %f\n", start_histo + (t+0.5)*delta,
             histo[t]/(delta*num_runs));

  return(0);
}
```

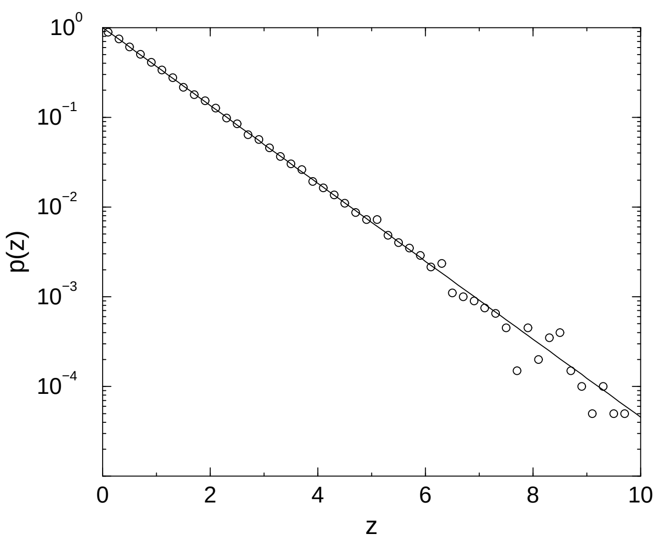


Abbildung 1: Histogramm von Zufallszahlen, generiert für eine Exponential Verteilung ($\lambda = 1$) verglichen mit W.-Dichte in logarithmischer Auftragung.

### 3.3 Zurückweisungsmethode

Für nicht-integrable W.-Dichten, oder nicht-invertierbare Verteilungen.

Bedingung: W-Dichte $p(z)$ passt in Kasten $[x_0, x_1) \times [0, p_{\max}]$, d.h. $p(z) = 0$ für $z \notin [x_0, x_1]$ und $p(z) \leq p_{\max}$.

Grundidee: Erzeuge zufällige Paare $(x, y)$, gleichverteilt in $[x_0, x_1) \times [0, p_{\max})$. Akzeptiere nur die $x$ mit $y \leq p(x)$, d.h. die Paare unterhalb $p(x)$, siehe Abb. 2.

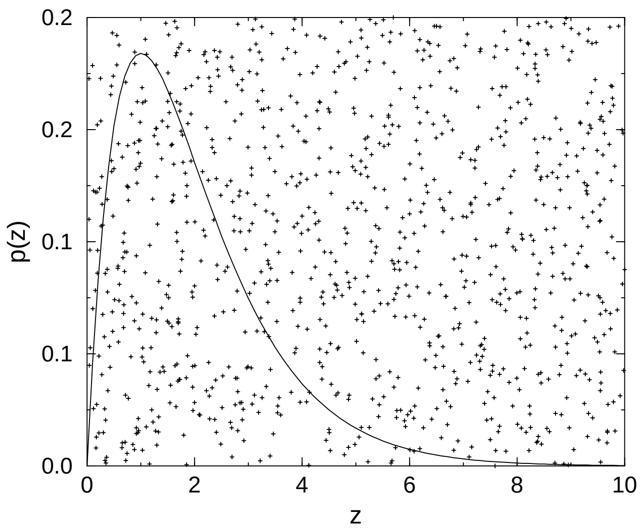


Abbildung 2: Zurückweisungsmethode: Punkte $(x, y)$ sind gleichmäßig in dem Rechteck verteilt. Die Wahrscheinlichkeit, dass $y \leq p(x)$ ist proportional zu $p(x)$.

Implementierung als Funktion:

```
/** gerenates random number for 'pdf' in the range **/
/** ['x0', 'x1'). condition: pdf(x)<='p_max' in    **/
//* the range ['x0', 'x1')                          **/
double reject(double p_max, double x0, double x1,
              double (* pdf)(double))
{
  int found;                 /* flag if valid number has been found */
  double x,y;                 /* random points in [x0,x1]x[0,p_max] */
  found = 0;
  while(!found)                   /* loop until number is generated */
  {
    x = x0 + (x1-x0)*drand48();             /* uniformly on [x0,x1] */
    y = p_max *drand48();                 /* uniformly in [0,p_max] */
    if(y <= pdf(x))                                     /* accept ? */
        found = 1;
  }
  return(x);
}
```

Beispiel:

```
/** artifical pdf **/
double pdf(double x)
{
  if( (x<0)||
      ((x>=0.5)&&(x<1))||
      (x>1.5) )
      return(0);
  else if((x>=0)&&(x<0.5))
      return(1);
  else
      return(4*(x-1));
}
```

ergibt bei 100000 Zufallszahlen:

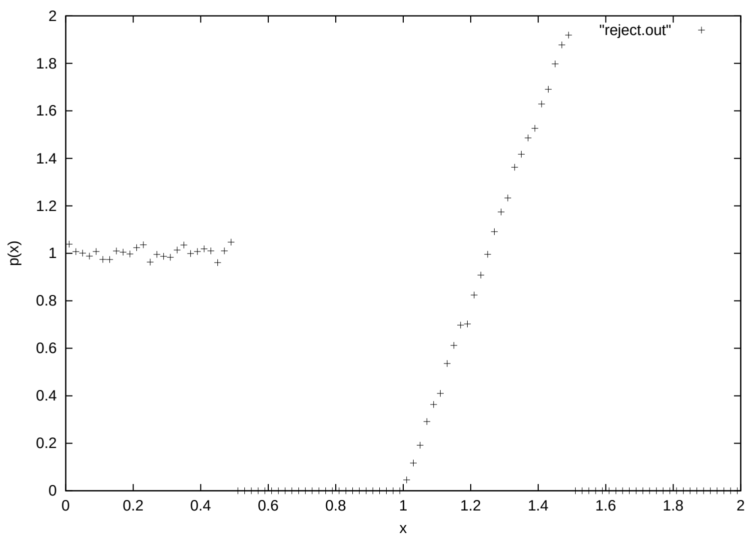


Abbildung 3: Zurückweisungsmethode: Histogramm für die angegebene W. Dichte.

Nachteil: Es müssen unter Umständen viele Zufallszahlen weggeworfen werden. Effektivität $1/2p_{\max}(x_1 - x_0)$.

## 3.4 Gaußverteilung

Gaußverteilung mit Mittelwert $m$ und Breite $\sigma$, die am meistverbreitete Verteilung in Physiksimulationen. W.-Dichte (siehe auch Abb. 4):

$$p_G(z) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(z-m)^2}{2\sigma^2}\right) \tag{3}$$

Im Folgenden: $z$ sei normalverteilt ($m = 0,\ \sigma = 1$). Allgemeiner Fall: Benutze $\sigma z + m$.
Weder Inversions- noch Zurückweisungsmethode geht. 3 Möglichkeiten:

- Mit künstlicher Beschränkung, z.B. in $[-3, 3] \rightarrow$ keine großen Zahlen.

- Verwende zentralen Grenzwertsatz:
  Summe von $N$ unabhängig verteilten Zufallszahlen $u_i$ (mit gleichem Mittelwert $m$ und Varianz $v$) konvergiert gegen Gaußverteilung mit Mittelwert $Nm$ und Varianz $Nv$.

  $\rightarrow$ z.B. Benutze $u_i$ gleichverteilt in $[0,1)$ ($m = 0.5$,$v = 1/12$), $N = 12$. Dann ist $Z = \sum_{i=1}^{12} u_i - 6$ wie gewünscht.

  Nachteile: 12 Zahlen nötig für eine Zufallszahl, nur Bereich $[-6,6]$.

- *Box-Müller Methode*: Nehme zwei $[0,1)$ gleichverteilte Zahlen $u_1, u_2$ und setzte:

$$\begin{aligned} n_1 &= \sqrt{-2\log(1-u_1)}\cos(2\pi u_2) \\ n_2 &= \sqrt{-2\log(1-u_1)}\sin(2\pi u_2) \end{aligned}$$

  dann sind $n_1, n_2$ normalverteilt.

  Beweis [1, 2]: Wir schreiben $n_1, n_2$ in Polarkoordinaten $(r,\theta)$, d.h. $(r,\theta) = f(n_1, n_2)$, das Inverse ist:

$$\begin{aligned} n_1 &= r\cos(\theta) \\ n_2 &= r\sin(\theta) \end{aligned}$$

  Gesucht: W.-Dichte für $(r,\theta)$.

  Allgemeiner Fall $(W,Z) = f(X,Y)$, $p_{X,Y}$ sei die (gemeinsame) W. Dichte für $(X,Y)$. Dann gilt: $p_{W,Z}(w,z) = p_{X,Y}(f^{-1}(w,z))|\mathbf{J^{-1}}|$ mit $|\mathbf{J^{-1}}|$ ist die Jacobi Determinante der inversen Transformation.

  Hier ist:

$$|\mathbf{J^{-1}}| = \begin{vmatrix} \frac{\partial n_1}{\partial r} & \frac{\partial n_1}{\partial \theta} \\ \frac{\partial n_2}{\partial r} & \frac{\partial n_2}{\partial \theta} \end{vmatrix} = \begin{vmatrix} \cos(\theta) & -r\sin(\theta) \\ \sin(\theta) & r\cos(\theta) \end{vmatrix} = r\cos^2(\theta) + r\sin^2(\theta) = r \quad (4)$$

  Da $n_1, n_2$ Gauß-verteilt sein sollen:

$$p_{R,\Theta}(r,\theta) = \frac{r}{2\pi} e^{-n_1^2/2 - n_2^2/2} = \frac{r}{2\pi} e^{-r^2/2} \quad (5)$$

  Faktorisiert! Wir nehmen $\theta$ gleichförmig in $[0, 2\pi)$ verteilt (d.h. Erzeugung $\theta = 2\pi u_2$). Es bleibt $p_R(r) = re^{-r^2/2}$ (*) übrig. Wie erzeugt man Zufallszahlen gemäß $p_R$?

  Sei nun $X$ exponentiell verteilt mit Parameter $\lambda$, d.h. $p_X(x) = \lambda e^{-\lambda x}$. Sei $Y = \sqrt{X}$. Gesucht: $p_Y(y)$.

  Allgemein gilt für $Y = H(X)$ ($H$ streng monoton):
  $p_Y(y) = p_X(H^{-1}(y))\frac{1}{|H'(H^{-1}(y))|}$.

  Hier ist $H(x) = \sqrt{x}$, d.h.. $H^{-1}(y) = y^2$ und $H'(x) = -1/2\sqrt{x}$. $\Rightarrow p_Y(y) = 2\lambda y e^{-\lambda y^2}$.

  Vergleich mit (*): Wir erzeugen somit $X$ exponential verteilt mit $\lambda = 0.5$ (d.h. $x = -2\log(1-u_1)$ mit Inversionsmethode), dann $r = \sqrt{x} \rightarrow$ gwünschte Verteilung für $r$. QED.

- Physikalischer Ansatz: Simulation von Gasteilchen in einem Kasten, siehe [3]

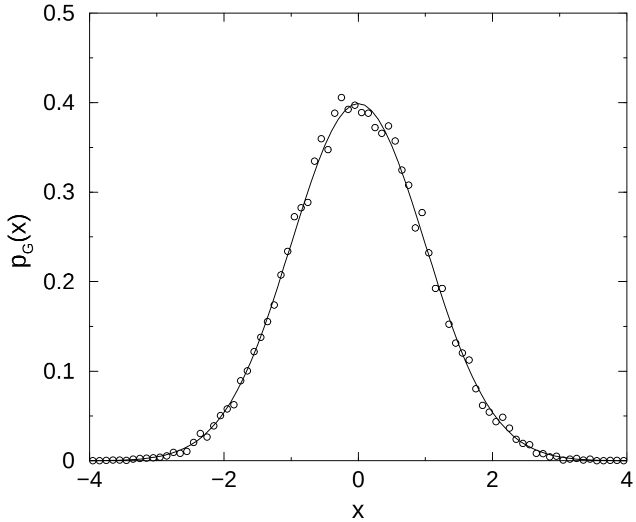


Abbildung 4: Gaußverteilung mit Mittelwert 0 und Varianz 1. Die Kreise entsprechen einem Histogramm, das aus $10^4$ Box-Müller erzeugten Zufallszahlen erstellt wurde.

# Literatur


[1] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery, *Numerical Recipes in C* (Cambridge University Press, Cambridge 1995)

[2] B.J.T. Morgan, *Elements of Simulation*, (Cambridge University Press, Cambridge 1984)

[3] J.F. Fernandez and C. Criado, *Phys. Rev. E* **60**, 3361 (1999)