

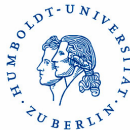
Stochastik–Praktikum

Zufallszahlen und deskriptive Statistik

Thorsten Dickhaus

Humboldt-Universität zu Berlin

04.10.2010



Übersicht

- 1 Erzeugung von Zufallszahlen
- 2 Monte Carlo-Integration
- 3 Deskriptive Statistik

Übersicht

- 1 Erzeugung von Zufallszahlen
- 2 Monte Carlo-Integration
- 3 Deskriptive Statistik

Johann von Neumann:

„Anyone who considers arithmetical methods of producing random digits is, of course, in a state of sin.“

Definition

Ein **Generator** (uniformer) **Pseudozufallszahlen** ist ein Algorithmus, der von einem Startwert u_0 (seed) und einer Transformation T ausgehend, eine rekursive deterministische Zahlenfolge $u_i = T^i u_0$ ($[0,1]$ -wertiger) Folgeglieder erzeugt, die sich wie eine zufällige i. i. d. Folge von echten (uniformen) Zufallszahlen verhalten soll.

Monte Carlo Methoden basieren auf der häufigen Wiederholung eines Zufallsexperimentes bzw. Erzeugung von Pseudozufallszahlen um (zumeist komplexe) analytische Probleme näherungsweise zu lösen, wobei das Gesetz der großen Zahlen die Grundlage hierfür bildet.

Pseudo- und Quasi-Zufallszahlen

Pseudo-Zufallszahlen:

Eine Zahlenfolge, deren Bildungsgesetz möglichst schwer zu „erraten“ ist.

Quasi-Zufallszahlen:

Eine Zahlenfolge, deren Häufigkeitsverteilung gemäß eines vorgegebenen Abstandsbegriffs einer vorgegebenen Wahrscheinlichkeitsverteilung (i. d. R. $\text{UNI}[0, 1]$) möglichst nahe kommt.

Pseudozufall: Mersenne Twister

Moderner Zufallszahlengenerator, der auch in R zur Erzeugung gleichverteilter Zufallszahlen Verwendung findet.

Es sei $\mathbb{F}_2$ der Körper der Charakteristik 2, also $\mathbb{F}_2 = \{0, 1\}$ mit der Addition $\oplus$ und der Multiplikation $\odot$.

Vorgegeben seien Parameter $\omega \in \mathbb{N}$, $n \in \mathbb{N}$, $m \in \{1, \dots, n\}$ und $r \in \{0, \dots, \omega - 1\}$.

Notation:

$$\begin{aligned} y^l &= (y_1, \dots, y_r, 0, \dots, 0), \\ z^u &= (0, \dots, 0, z_{r+1}, \dots, z_\omega), \\ (y^l | z^u) &= (y_1, \dots, y_r, z_{r+1}, \dots, z_\omega). \end{aligned}$$

Pseudozufall: Mersenne Twister

Notation:

$$y^l = (y_1, \dots, y_r, 0, \dots, 0),$$

$$z^u = (0, \dots, 0, z_{r+1}, \dots, z_\omega),$$

$$(y^l | z^u) = (y_1, \dots, y_r, z_{r+1}, \dots, z_\omega).$$

Mit der $\omega \times \omega$ Matrix A und $(n - 1)$ Startwerten $x_0, \dots, x_{n-1} \in \mathbb{F}_2^\omega$ ist der Mersenne-Twister der folgende rekursive Algorithmus:

$$x_{k+n} = x_{k+m} \oplus^\omega \left(x_k^l | x_{k+1}^u \right) \odot^\omega A, k \in \mathbb{N}_0.$$

$\oplus^\omega$ und $\odot^\omega$ bezeichnen die Addition und Multiplikation in $\mathbb{F}_2^\omega$.

In R: $(\omega, n, m, r) = (32, 624, 397, 31)$.

Quasi-Zufallszahlen

Ziel bei der Generierung einer Folge von **Quasi-Zufallszahlen** $x_1, \dots, x_N$ ist die Minimierung der Diskrepanz

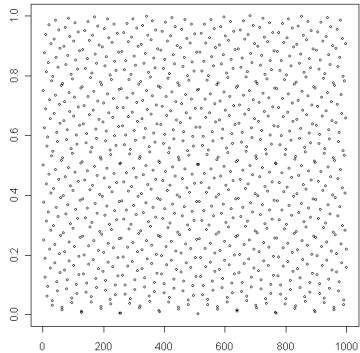
$$D_N(x_1, \dots, x_N) = \sup_{u \in [0,1]} \left| \frac{|\{x_i : i = 1, \dots, N, x_i \in [0, u]\}|}{N} - u \right|.$$

Quasi-Zufallszahlen (Halton-, Sobol-Folgen): $D_N \leq C(\log N/N)$

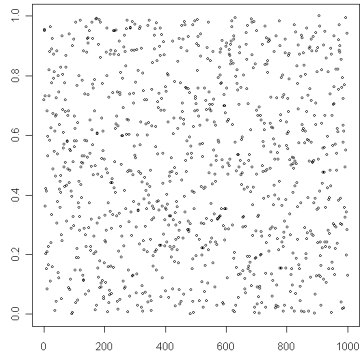
Für Pseudozufallszahlen indes $D_N \simeq N^{-1/2}$ nach ZGWS

⇒ kleinerer Fehler bei **Quasi**-Monte Carlo Integration

Sobol-Quasi-Zufallszahlen



Pseudozufallszahlen



Quasi-Zufall: Halton-Folge

Man wähle eine Primzahl p als Basis und einen Startwert $m \neq 0$ und stelle m zur Basis p dar:

$$m = \sum_{j=0}^k a_j p^j .$$

Die Haltonzahlen sind dann gegeben durch

$$h = \sum_{j=0}^k a_j p^{-j-1} .$$

Man verfährt analog mit $(m+1), \dots$

Quasi-Zufall: Halton-Folge

Die folgende Tabelle enthält die ersten drei Haltonzahlen zum Startwert $m = 3$ für $p = 2$.

m	binäre Darstellung	h
3	11	$3/4$
4	100	$1/8$
5	101	$5/8$
$\vdots$	$\vdots$	$\vdots$

Erzeugung Bernoulli-verteilter Zufallszahlen

Gegeben eine Zufallsvariable $U \sim U[0, 1]$ folgt

$$X := T(U) = \begin{cases} 1 & \text{für } U \leq p \\ 0 & \text{für } U > p \end{cases}$$

ist Bernoulli-verteilt mit Parameter $p \in (0, 1)$.

Erzeugt man iid. $U_1, \dots, U_n \sim U[0, 1]$ und addiert die resultierenden X_i , so ist die Summe binomialverteilt mit Parametern n und p .

Ähnliche Diskretisierungsmethoden sind für andere diskrete Verteilungen anwendbar!

Erzeugung normalverteilter Zufallszahlen

Eine reellwertige Zufallsvariable heißt $\mathbf{N}(\mu, \sigma^2)$ -verteilt, wenn sie die stetige Dichte

$$f_{\mu, \sigma^2}(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp(-(x - \mu)^2 / (2\sigma^2))$$

bezüglich des Lebesguemaßes besitzt.

Ist X normalverteilt mit Parametern μ und σ^2 , so gilt

$$X = \mu + \sigma Z,$$

wobei Z standardnormalverteilt ist.

Erzeugung normalverteilter Zufallszahlen

Box–Muller–Methode: $U, V \stackrel{i.i.d.}{\sim} \mathbf{U}[0, 1] \Rightarrow$

$$(X, Y) = \sqrt{-2 \log U} (\cos 2\pi V, \sin 2\pi V) \stackrel{i.i.d.}{\sim} \mathbf{N}(0, 1).$$

Polarkoordinaten: Es seien (U, V) uniform auf dem Einheitskreis verteilt (erhält man durch die Verwerfungsmethode), so folgt

$$(X, Y) = \sqrt{\frac{-2 \log (U^2 + V^2)}{U^2 + V^2}} (U, V) \stackrel{i.i.d.}{\sim} \mathbf{N}(0, 1).$$

Die Inversionsmethode für stetige Verteilungen

Definition

Es sei F eine streng monotone Verteilungsfunktion. Die Funktion

$$F^{-1}(u) := \begin{cases} \inf \{x | F(x) \geq u\} & , u \in (0, 1] \\ \sup \{x \in \mathbb{R} | F(x) = 0\} & , u = 0 \end{cases}$$

heißt Quantilfunktion der zugehörigen Verteilung.

Die Inversionsmethode für stetige Verteilungen

Lemma

Ist $U \sim \mathcal{U}[0, 1] \Rightarrow X = F^{-1}(U) \sim F$ für jede streng monotone Verteilungsfunktion F . Die letzte Aussage bedeutet, dass die Zufallsvariable $F^{-1}(U)$ verteilt ist nach der zu F gehörenden Verteilung.

Beweis:

$$\mathbb{P}(X \leq x) = \mathbb{P}(F^{-1}(U) \leq x) = \mathbb{P}(U \leq F(x)) = F(x).$$

Die Inversionsmethode für stetige Verteilungen

Lemma

Ist $U \sim \mathcal{U}[0, 1] \Rightarrow X = F^{-1}(U) \sim F$ für jede streng monotone Verteilungsfunktion F . Die letzte Aussage bedeutet, dass die Zufallsvariable $F^{-1}(U)$ verteilt ist nach der zu F gehörenden Verteilung.

Beispielsweise ist

$$-\frac{1}{\lambda} \log U \sim \text{Exp}(\lambda) .$$

Die Inversionsmethode für diskrete Verteilungen

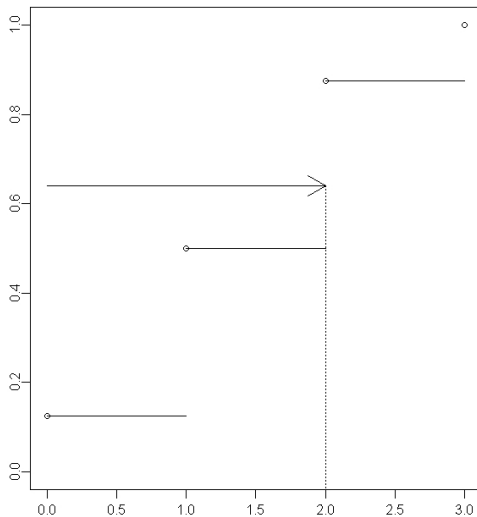
Es seien $p_i, i \in J$ die diskreten Gewichte der zur Rede stehenden Verteilung zu den Werten $m_i, i \in J$, wobei $m_1 < m_2 < \dots < m_{|J|}$. Die rechtsstetige Treppenfunktion

$$P(x) = \sum_{i: m_i \leq x} p_i$$

ist die zugehörige Verteilungsfunktion. Wenn nun u eine auf dem Einheitsintervall gleichverteilte Zufallszahl ist, kann durch

$$x := \min \{t, u \leq P(t)\}$$

eine Zufallszahl der diskreten Verteilung erzeugt werden.



Übersicht

- 1 Erzeugung von Zufallszahlen
- 2 Monte Carlo-Integration
- 3 Deskriptive Statistik

Monte Carlo-Integration

Näherungsweise Berechnung von

$$\begin{aligned}\int_a^b g(x) dx &= \int_a^b h(x) f(x) dx \\ &= \int h(x) d\mathbb{P}_f(x) = \mathbb{E}_f(h(X))\end{aligned}$$

durch den Mittelwert einer endlichen Folge von iid. erzeugten Zufallszahlen x_i mit Verteilung $\mathbb{P}_f$:

$$S = \frac{1}{n} \sum_{i=1}^n h(x_i) .$$

Crude Monte Carlo Integration

Speziell mit $X_i \stackrel{i.i.d.}{\sim} U[a, b]$ ergibt sich der Schätzer

$$S = \frac{1}{n} \sum_{i=1}^n g(x_i)(b-a).$$

Der Schätzer ist erwartungstreu, da

$$\mathbb{E}(S) = \frac{(b-a)}{n} \sum_1^n \mathbb{E}[g(X_i)] = \int_a^b g(x) dx$$

und nach dem Gesetz der großen Zahlen konsistent mit Varianz

$$\frac{(b-a)^2}{n^2} \sum_{i=1}^n \text{Var}(g(X_i)) = \frac{(b-a)}{n} \int_a^b \left(g(x) - \int_a^b g(t) dt \right)^2 dx.$$

Übersicht

- 1 Erzeugung von Zufallszahlen
- 2 Monte Carlo-Integration
- 3 Deskriptive Statistik**

Univariate Daten: Michelson's Lichtgeschwindigkeits-Daten

1	850	1	1
2	740	2	1
3	900	3	1
4	1070	4	1
5	930	5	1
6	850	6	1
7	950	7	1
8	980	8	1
9	980	9	1
10	880	10	1
⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮

Interpretation der Daten

1	850	1	1
2	740	2	1
3	900	3	1
4	1070	4	1
⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮

- Erste Spalte : Fortlaufende Nummer der Messungen (1-100)
Zweite Spalte : (Gemessene Geschwindigkeit - 299.000) in km/s
Dritte Spalte : Fortlaufende Nummer in der Messreihe (1-20)
Vierte Spalte : Nummer der Messreihe (1-5)

Einlesen der Daten

```
> l<-read.table("lightspeed.dat")
> str(l)
'data.frame': 100 obs. of 4 variables:
 $ V1: int 1 2 3 4 5 6 7 8 9 10 ...
 $ V2: int 850 740 900 1070 930 850 950 980 980 880 ...
 $ V3: int 1 2 3 4 5 6 7 8 9 10 ...
 $ V4: int 1 1 1 1 1 1 1 1 1 1 ...
> attributes(l)
> dim(l)
[1] 100 4
> is.matrix(l)
[1] FALSE
> is.list(l)
[1] TRUE
> mode(l)
[1] "list"
> speed<-l$V2
```

Variablen

```
> names(l)
[1] "V1" "V2" "V3" "V4"
> names(l) <- c("No", "Speed", "ExNo", "Ex")
> attach(l)
```

The following object(s) are masked from l (position 3) :

Ex ExNo No Speed

```
> ex1 <- subset(l, Ex == 1)
> s <- ex1$Speed
```

Statistische Kenngrößen

(Arithmetischer) Mittelwert $\bar{x} = \sum_{i=1}^n x_i$:

```
> mean(s) [1] 909
```

Standardabweichung $\sqrt{(1/(N-1)) \sum_i (x_i - \bar{x})^2}$:

```
> sd(s) [1] 104.9260
```

Median $med = x_{((n+1)/2)}$:

```
> median(s) [1] 940
```

Median absoluter Abweichungen $MAD = n^{-1} \sum_i |x_i - med(\mathbf{x})|$:

```
> mad(s) [1] 88.956
```

Statistische Kenngrößen

Schiefe (skewness):

$$v(X) = \frac{\mathbb{E}[(X - \mathbb{E}X)^3]}{\text{Var}(X)^{3/2}}.$$

Die Schiefe einer empirischen Verteilung:

$$v_e(\mathbf{x}) = \frac{n^{-1} \sum_i (x_i - \bar{x})^3}{(n^{-1} \sum_i (x_i - \bar{x})^2)^{3/2}}$$

```
> skew<-function(x){  
+ skewness<-  
+ ((sqrt(length(x))*sum((x-mean(x))^3))/(sum((x-mean(x))^2))^(3/2))  
+ return(skewness)}  
> skew(s)  
[1] -0.890699
```

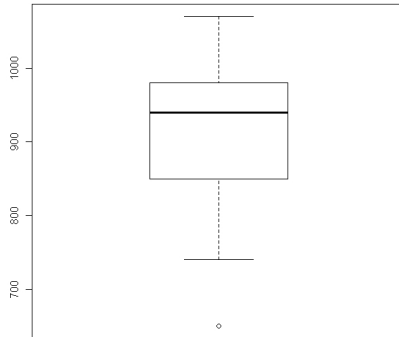
Die **summary**-Funktion

```
> summary(ex1)
```

No	Speed	ExNo	Ex
Min. : 1.00	Min. : 650	Min. : 1.00	Min. :1
1st Qu.: 5.75	1st Qu.: 850	1st Qu.: 5.75	1st Qu.:1
Median :10.50	Median : 940	Median :10.50	Median :1
Mean :10.50	Mean : 909	Mean :10.50	Mean :1
3rd Qu.:15.25	3rd Qu.: 980	3rd Qu.:15.25	3rd Qu.:1
Max. :20.00	Max. :1070	Max. :20.00	Max. :1

Der Box-Whisker-Plot

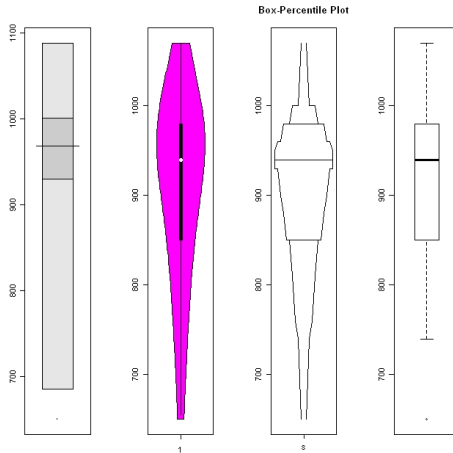
>**boxplot(s)**



R Code: Darstellung von Daten

```
> require(hdrcde)
> require(vioplot)
> require(Hmisc)
> par(mfrow=c(1,4))
> hdr.boxplot(s)
> vioplot(s)
> bpplot(s)
> boxplot(s)
> dev.off()
null device
      1
```

Box-Plot & Co



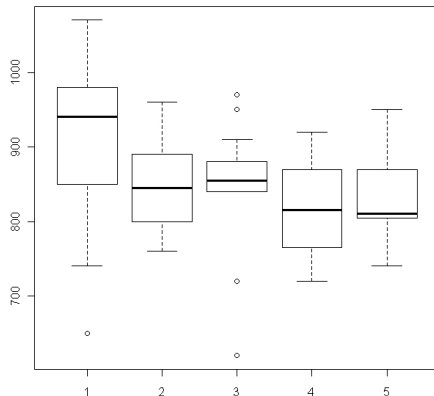
R Code: Herausnehmen von Ausreißern

```
> strim<-s[which(s>700) ]  
> summary( strim )
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
740.0	865.0	950.0	922.6	980.0	1070.0

Vergleich der Messreihen

> **boxplot**(I\$Speed~I\$Ex)



Multivariate Daten: Mietspiegel-Daten

```
> miete<-read.table(file="miete03.asc",header=TRUE)
> str(miete)
'data.frame':   2053 obs. of  16 variables:
 $ GKM      : num  741 716 528 554 698 ...
 $ QMKM     : num  10.9 11.01 8.38 8.52 6.98 ...
 $ QM       : int   68 65 63 65 100 81 55 79 52 77 ...
 $ Zi       : int    2 2 3 3 4 4 2 3 1 3 ...
 $ BJ       : num  1918 1995 1918 1983 1995 ...
 $ B        : int    2 2 2 16 16 16 6 6 6 6 ...
 $ L        : int    1 1 1 0 1 0 0 0 0 0 ...
 $ best     : int    0 0 0 0 0 0 0 0 0 0 ...
 $ VW       : int    0 0 0 0 0 0 0 0 0 0 ...
 $ ZH       : int    0 0 0 0 0 0 0 0 0 0 ...
 $ BK       : int    0 0 0 0 0 0 0 0 0 0 ...
 $ BA       : int    0 0 0 1 1 0 1 0 0 0 ...
 $ KUE      : int    0 0 0 0 1 0 0 0 0 0 ...
```

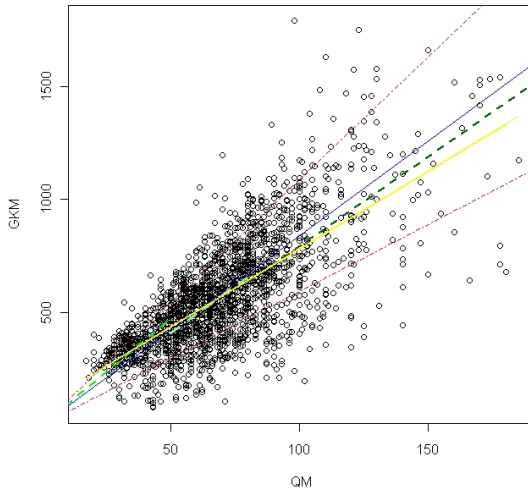
Abgeleitete Variablen

Hier: Klassierung von Baujahr und Quadratmeterzahl

```
> miete$BJKL<-1*(BJ<=1918)+2*(BJ<=1948)*(BJ>1919)+3*(BJ<=1965)  
      *(BJ>1948)+4*(BJ<=1977)*(BJ>1965)+5*(BJ<=1983)  
      *(BJ>1977)+6*(BJ>1983)  
  
> miete$QMKL<-1*(QM<=50)+2*(QM>50)*(QM<=80)+3*(QM>80)
```

Zusammenhänge zwischen Variablen

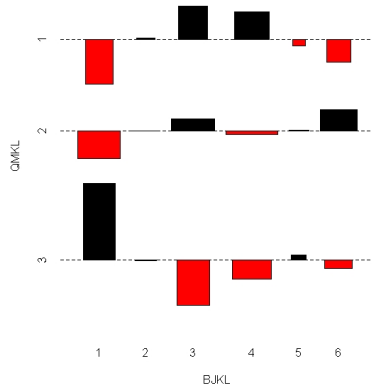
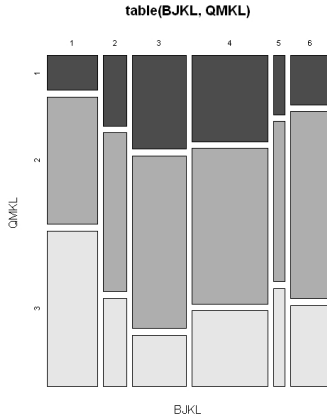
```
> plot(QM,GKM)
> abline(0,mean(QMKM),col="blue")
> abline(0,mean(QMKM)+sd(QMKM),col="red",lty=4)
> abline(0,mean(QMKM)-sd(QMKM),col="red",lty=4)
> z<-tapply(QMKM,QMKL,mean)
> segments(0,0,50,z[1]*50,col="green",lwd=2,lty=2)
> segments(50,50*z[2],80,z[2]*80,col="lightgreen",lwd=3,lty=2)
> segments(80,80*z[3],200,z[3]*200,col="darkgreen",lwd=2,lty=2)
> lines(QM,fitted(lm(GKM~QM)),col="yellow")
```



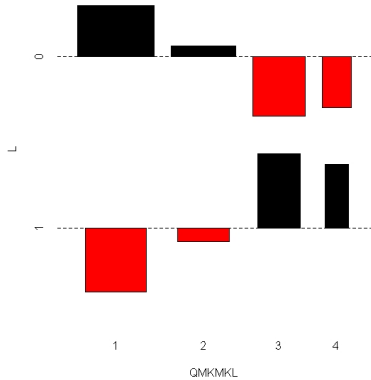
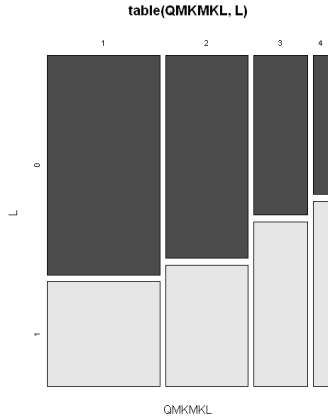
R Code: assocplot und mosaicplot

```
> par(mfrow=c(1,2))  
> mosaicplot(table(BJKL,QMKL),col=TRUE)  
> assocplot(table(BJKL,QMKL))  
  
> miete$QMKMKL<-1*(QMKM<=8)+2*(QMKM>8)*(QMKM<=10)  
+3*(QMKM>10)*(QMKM<=12)+4*(QMKM>12)  
> mosaicplot(table(QMKMKL,L),col=TRUE)  
> assocplot(table(QMKMKL,L))
```

Baujahr $\leftrightarrow$ Wohnungsgröße



Miete $\leftrightarrow$ Wohnlage



R Code: Häufigkeitsdarstellungen

```
> h<-numeric(6)
> for(i in 1:6){
+ h[i]<-length(which(BJKL==i))}
> names(h)<-c("vor_1918", "1919-1948", "1948-1965", "1966-1977",
+ "1978-1983", "Neubau")
> pie(h, col=rainbow(6))
> barplot(h, col=heat.colors(6), density=100)
```

