

Kapitel 2

Erzeugung von Zufallsvariablen

2

2	Erzeugung von Zufallsvariablen	
2.1	Zufallszahlen	21
2.2	Die Inversionsmethode	34
2.3	Die Verwerfungsmethode	44
2.4	Die Faltungsmethode	50
2.5	Die Alias-Methode	54
2.6	Die Kompositionsmethode	57
2.7	Berücksichtigung weiterer Verteilungszusammenhänge...	62
2.8	Erzeugung mehrdimensionaler Zufallsvariablen.....	66
2.9	Aufgaben	70

Erzeugung von Zufallsvariablen

Der Umgang mit dem Zufall in einer Simulationsstudie basiert auf der Erzeugung von Zufallszahlen, d.h. auf der zufälligen Auswahl von Zahlen zwischen 0 und 1.

Will man den Wurf mit einem fairen Würfel simulieren, so unterteilt man das Intervall $[0, 1]$ in sechs gleichlange Teilintervalle, z.B. $[0, 1/6)$, $[1/6, 2/6)$, \dots , $[5/6, 1]$, ordnet jeder Augenzahl ein Teilintervall zu, erzeugt eine Zufallszahl und interpretiert die Augenzahl, die dem Teilintervall zugeordnet ist, in das die Zufallszahl fällt, als Ergebnis des Wurfes mit dem fairen Würfel.

Ist man an der Anzahl der Würfe bis zum ersten Auftreten der Augenzahl 6 interessiert, so kann man das Experiment solange wiederholen, bis das Ereignis eintritt; man kommt aber auch, wie wir noch sehen werden, mit einer einzigen Zufallszahl aus, wenn man berücksichtigt, dass die Anzahl der Würfe geometrisch verteilt ist.

Dieses einfache Experiment zeigt bereits die prinzipielle Vorgehensweise bei der Modellierung der stochastischen Objekte innerhalb der Simulation: Ausgehend von einer Zufallszahl, d.h. Realisation einer $U[0, 1]$ -verteilten Zufallsvariablen, erzeugen wir eine Realisation einer beliebig verteilten Zufallsvariablen. Damit kommen auf uns zwei Aufgaben zu: (a) die Erzeugung von unabhängigen, $U[0, 1]$ -verteilten Zufallsvariablen und (b) deren Transformation im Hinblick auf die Erzeugung einer beliebig verteilten Zufallsvariablen.

Zufallszahlen

2.1

Bei der technischen Umsetzung der Zufälligkeit verwendet man (im Gegensatz zu den Anfängen der Simulation) keine echten Zufallszahlen, sondern erzeugt *deterministisch* eine Zahlenfolge z_0, z_1, \dots mit Werten in $[0, 1]$, die man als Realisationen einer Folge unabhängiger, $U[0, 1]$ -verteilter Zufallsvariablen interpretiert. Dabei handelt es sich streng genommen nur um **Pseudo-Zufallszahlen**. Der Einfachheit halber sprechen wir aber von **Zufallszahlen**. Gegenüber echten Zufallszahlen haben Pseudo-Zufallszahlen den Vorteil, dass sie einfach erzeugt werden können und reproduzierbar sind. Um die Zufälligkeit der erzeugten Zahlen sicherzustellen, muss der zugrunde liegende Algorithmus (Generator) jedoch sorgfältig ausgewählt werden.

Lineare Kongruenzgeneratoren gehören zu den bekanntesten Generatoren und waren lange sehr populär. Sie erzeugen eine Folge von nichtnegativen

ganzen Zahlen x_0, x_1, x_2, \dots gemäß der Vorschrift

$$x_{n+1} = (ax_n + c) \text{ modulo } m. \quad (2.1)$$

Weniger formal lautet die Rekursionsgleichung: Der Wert x_{n+1} ergibt sich als Rest bei der Ganzzahl-Division von $(a \cdot x_n + c)$ geteilt durch m und nimmt so einen Wert in der Menge $\{0, 1, \dots, m-1\}$ an.

Der Anfangswert $x_0 \in \{0, \dots, m-1\}$ wird gewöhnlich als **Quelle** (Seed) bezeichnet, die Konstanten $a \in \mathbb{N}$ als **Multiplikator**, $c \in \mathbb{N}_0$ als **additive Konstante** und $m \in \mathbb{N}$ als **Modulus**.

Das folgende Beispiel veranschaulicht die Vorgehensweise.

2.1 Beispiel

Für den linearen Kongruenzgenerator mit $x_0 = 1$, $a = 5$, $c = 3$ und $m = 16$ erhalten wir die Zahlen $0, 1, \dots, 15$ in der Reihenfolge

1, 8, 11, 10, 5, 12, 15, 14, 9, 0, 3, 2, 13, 4, 7, 6, 1, 8, ...

$x_0 = 1$ ist der Anfangswert. Teilt man $ax_0 + c = 5 \cdot 1 + 3 = 8$ durch 16, so erhält man 0 Rest 8 und damit $x_1 = 8$. Teilt man $ax_1 + c = 5 \cdot 8 + 3 = 43$ durch 16, so erhält man 2 Rest 11 und damit $x_2 = 11$. Fährt man auf diese Weise fort, so ergeben sich die restlichen Zahlen. Mit $x_{16}, x_{17}, \dots, x_{32}, x_{33}, \dots$ wiederholen sich diese Zahlen. \diamond

Werden wie in Beispiel 2.1 alle m Zahlen ausgeschöpft, so sprechen wir von einem linearen Kongruenzgenerator mit **voller Periodenlänge**.

2.2 Beispiel

Betrachten wir den linearen Kongruenzgenerator aus Beispiel 2.1 mit $x_0 = 1$, $a = 5$ und $m = 16$, wählen jedoch $c = 2$, so erhalten wir die Zahlenfolge

1, 7, 5, 11, 9, 15, 13, 3, 1, 7, ...

Hier werden lediglich 8 der 16 möglichen Werte angenommen, bevor die Zahlenfolge sich wiederholt. Noch extremer verhält sich der lineare Kongruenzgenerator bei Wahl von $a = 14$ und $c = 3$. Wir erhalten die konstante Zahlenfolge

1, 1, 1, 1, 1, 1, 1, 1, ... ;

der Generator liefert also nur einen einzigen Wert. \diamond

Beispiel 2.2 zeigt, dass wir nicht generell von einer vollen Periodenlänge ausgehen können und daher die Parameter des Generators sorgfältig auswählen müssen. Mit Hilfe zahlentheoretischer Überlegungen lässt sich jedoch zeigen, dass ein linearer Kongruenzgenerator genau dann die volle Periodenlänge hat, wenn die folgenden drei Bedingungen (z.B. Law (2007), Theorem 7.1) erfüllt sind:

- (a) Die 1 ist die einzige positive Zahl, die sowohl Teiler von m als auch von c ist.
- (b) Ist q Teiler von m und ist q eine Primzahl, so ist q auch Teiler von $a - 1$.
- (c) Ist 4 Teiler von m , so ist 4 auch Teiler von $a - 1$.

Dabei ist wie üblich $q \in \mathbb{N}$ ein Teiler von $m \in \mathbb{N}$, wenn die Ganzzahl-Division den Rest 0 ergibt. Eine Primzahl ist nur durch sich und 1 teilbar.

Zur Veranschaulichung der Bedingungen kommen wir noch einmal auf die Beispiele 2.1 und 2.2 zurück.

Beispiel

2.3

Bei dem Generator aus Beispiel 2.1 sind die Bedingungen (a)-(c) erfüllt: (a) Die 1 ist die einzige positive Zahl, die sowohl Teiler von $m = 16$ als auch von $c = 3$ ist. (b) $m = 16$ hat die Teiler 1, 2, 4, 8, 16, von denen 1 und 2 Primzahlen sind. 1 und 2 sind auch Teiler von $a - 1 = 5 - 1 = 4$. (c) Die Zahl 4 ist Teiler von $m = 16$ und auch von $a - 1 = 5 - 1 = 4$. Demzufolge hat der Generator die volle Periodenlänge.

Dies trifft auf die Generatoren aus Beispiel 2.2 nicht zu. Bei dem ersten ist die Bedingung (a) nicht erfüllt: Neben der 1 ist die 2 sowohl Teiler von $m = 16$ als auch von $c = 2$; bei dem zweiten sind die Bedingungen (b) und (c) nicht erfüllt: (b) $m = 16$ hat die Teiler 1, 2, 4, 8, 16, von denen 1 und 2 Primzahlen sind. 2 ist aber kein Teiler von $a - 1 = 14 - 1 = 13$. (c) Die Zahl 4 ist Teiler von $m = 16$, aber nicht von $a - 1 = 14 - 1 = 13$. \diamond

Abschließend betrachten wir noch den linearen Kongruenzgenerator mit $x_0 = 1$, $a = 1$, $c = 1$ und $m = 16$. Er liefert die Zahlenfolge

1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 0, 1, 2, ...

Die volle Periodenlänge wird zwar erreicht, doch die angestrebte „Zufälligkeit“ der Zahlen darf zumindest in Frage gestellt werden. Daher müssen wir neben der (nahezu) vollen Periodenlänge auch auf die „Zufälligkeit“ der generierten Zahlen achten.

Unter einem **multiplikativen Kongruenzgenerator** versteht man einen linearen Kongruenzgenerator, dessen additive Konstante c den Wert Null annimmt. Dadurch reduziert sich die Rekursionsgleichung (2.1) auf

$$x_{n+1} = ax_n \quad \text{modulo } m \quad (2.2)$$

mit $x_0 \in \{1, \dots, m-1\}$ und $a \in \{2, \dots, m-1\}$.

Durch den Wegfall der additiven Konstante ist der multiplikative Kongruenzgenerator dem linearen Kongruenzgenerator im Hinblick auf eine effiziente Berechnung überlegen. Die maximale Periodenlänge ist $m-1$. Sie wird erreicht, wenn $m > 2$ eine Primzahl ist und $\ell = m-1$ die kleinste ganze Zahl ℓ , für die $a^\ell - 1$ durch m teilbar ist (Knuth (1998), p. 20).

Bei einem Rechner mit einer 32 Bit Arithmetik galt $m = 2^{31} - 1$, $a = 16807$ lange Zeit als „minimaler Standard“ für einen „guten“ Generator (vgl. Park, Miller (1988)). Neueren Untersuchungen zufolge eignen sich auch $a = 48271$ oder $a = 69621$ gut (für die Klasse dieser einfachen Generatoren). Bei einer intensiveren Suche erwies sich $a = 62089911$ als besser, aber auch deutlich schwieriger portabel zu implementieren.

Eine Verallgemeinerung des linearen Kongruenzgenerators ist das Rekursionsschema

$$x_n = (a_1 x_{n-1} + \dots + a_k x_{n-k}) \quad \text{modulo } m. \quad (2.3)$$

Ist m eine Primzahl, so ist die maximale Periodenlänge $m^k - 1$ erreichbar (siehe Knuth (1998), p. 29). Die typische Wahl von m ist eine Primzahl, die nahe an der größten auf dem Rechner darstellbaren Integer-Zahl ist. Dies könnte z.B. $m = 2^{31} - 1$ auf einem 32 Bit Rechner sein. Die Implementierung erfordert jedoch eine zusätzliche Programmieretechnik, falls $a_1 x_{n-1} + \dots + a_k x_{n-k}$ nicht direkt darstellbar ist.

Da die Zahlenfolge x_0, x_1, \dots , die wir mit den linearen Kongruenzgeneratoren (2.1) - (2.3) erhalten, nur Werte in der Menge $\{0, \dots, m-1\}$ annimmt, müssen wir noch eine Normierung vornehmen. Die normierte Zahlenfolge z_0, z_1, \dots mit $z_n := x_n/m$ nimmt dann Werte in $[0, 1]$ an.

Der gravierende praktische Nachteil dieser Generatoren besteht darin, dass heute auf gängigen PCs eine Periodenlänge von $2.1 \cdot 10^9$ sehr schnell ausgeschöpft wird und durch Wiederholungen negative Effekte in die Simulati-

on eingeschleppt werden. Andererseits benötigen bestimmte, numerisch sehr aufwendige Simulationen einen Vorrat an Zufallszahlen, der durch die bisher betrachteten Kongruenzgeneratoren nicht bereit gestellt werden kann. Abhilfe schafft eine Mischung von Kongruenzgeneratoren. Als Beispiel betrachten wir das folgende vierstufige Verfahren, das in namhaften Software-Paketen implementiert ist:

1. $x_n = (a_1 x_{n-2} - a_2 x_{n-3}) \text{ modulo } m$.
2. $x'_n = (a'_1 x'_{n-1} - a'_2 x'_{n-3}) \text{ modulo } m'$.
3. $x''_n = (x_n - x'_n) \text{ modulo } m$.
4. Ist $x''_n > 0$, setze $z_n = \frac{x''_n}{m+1}$. Andernfalls setze $z_n = \frac{m}{m+1}$.

Der Algorithmus geht auf L'Ecuyer (1999) zurück zusammen mit der Empfehlung $m = 4294967087$, $m' = 4294944443$, $a_1 = 1403580$, $a_2 = 810728$, $a'_1 = 527612$ und $a'_2 = 1370589$, die auf umfangreichen numerischen Vergleichen basiert.

Durch die Mischung in Schritt 3 kommt man auf eine Periodenlänge von etwa 2^{191} . Da $x''_n = 0$ möglich ist, nimmt man in Schritt 4 eine Verschiebung vor, die sicherstellt, dass die resultierende Zufallszahl $z_n \in (0, 1)$ ist.

Abschließend betrachten wir noch eine Klasse von Zufallszahlengeneratoren, die auf der Dualdarstellung der Zahlen basieren. Alle Operationen sind bzgl. der modulo-2-Arithmetik zu verstehen.

Der **Tausworthe Generator** mit den Parametern $p, q, \ell \in \mathbb{N}$ und $q < p$ erzeugt eine Folge x_0, x_1, \dots natürlicher Zahlen,

$$x_n = b_{n\ell}2^0 + b_{n\ell+1}2^1 + \dots + b_{n\ell+\ell-1}2^{\ell-1},$$

die sich aus der Dualdarstellung

$$x_n = (b_{n\ell+\ell-1}, \dots, b_{n\ell+1}, b_{n\ell})_2$$

(mit der Wortlänge ℓ) rekursiv berechnen lassen, wobei die Folge (b_m) von Bits mit den Startwerten $b_0, \dots, b_{p-1} \in \{0, 1\}$ durch die Operation

$$b_n = b_{n-p} + b_{n-(p-q)} \text{ modulo } 2$$

erzeugt wird. Bspw. angewandt auf $p = 5, q = 2, \ell = 3$ und $(b_0, b_1, b_2, b_3, b_4) = (1, 1, 1, 1, 1)$ ergeben sich die in Tab. 2.1 angegebenen Werte.

b_0	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	b_9	b_{10}	b_{11}	b_{12}	b_{13}	b_{14}
1	1	1	1	1	0	0	0	1	1	0	1	1	1	0
x_0			x_1			x_2			x_3			x_4		
7			3			4			5			3		

Tabelle 2.1. Veranschaulichung des Tausworthe Generators

Die Auswahl des Abschnitts von Bits, die eine Zahl x_n bilden, kann auch mit Lücken oder überlappend erfolgen. Allgemein gilt dann mit $t \in \mathbb{N}$:

$$x_n = (b_{nt+\ell-1}, \dots, b_{nt+1}, b_{nt})_2.$$

Für $b_n = b_{n-102} + b_{n-249}$ modulo 2 erhält man (vgl. Asmussen, Glynn (2007), p. 34) ein sehr schnelles Verfahren mit der Periodenlänge $2^{500} - 1$.

Ein Tausworthe Verfahren erzeugt demnach primär zufällige Bit-Folgen nahezu beliebiger Länge. Daher eignen sich manche Varianten auch zum Einsatz in der Kryptographie. Zur Erzeugung von letzten Endes $U[0, 1]$ -verteilten Zufallszahlen gibt es mehrere Wege. In beiden Fällen greift man Abschnitte meist gleicher Länge aus der Gesamtfolge heraus. Dann kann man diese Abschnitte mit nicht-negativen Zweierpotenzen versehen als Integer-Zahlen auffassen, die man direkt als solche verwenden könnte, oder die man durch jeweils eine Division zu Zufallszahlen z aus $[0, 1]$ macht. Dies ist der oben beschriebene Weg, der sehr ausführlich bei Kolonko (2008) dargestellt ist (und dem auch das Beispiel entnommen wurde), einschließlich der üblichen Schieberegister-Implementierungen. Ein anderer Weg geht über die Verwendung der Abschnitte der Bit-Folgen als duale Nachkommastellen (verwendet also negative Zweierpotenzen) und führt nach Dezimalkonversion direkt zu Zufallszahlen z aus $[0, 1]$. Bezeichnet man die gemeinsame Länge der selektierten Bit-Abschnitte mit r , so kann man aber auf einem r -Bit Computer direkt nicht über die Periodenlänge von 2^r hinauskommen. Dies wird allgemein, neben gewissen Schwachpunkten bei den statistischen Eigenschaften, als ein Minuspunkt bei Tausworthe Generatoren angesehen.

Ein sehr effizienter Zufallszahlengenerator ist der **Mersenne Twister** von Matsumoto, Nishimura (1998), der über eine Periodenlänge von $2^{19937} - 1$ ($\approx 4.3 \cdot 10^{6001}$) und eine exzellente Anpassung an die Gleichverteilung verfügt. Dabei handelt es sich um ein rekursives Verfahren, das auf der Dualdarstellung der Zahlen basiert und von einer Reihe von z.T. unkonventionellen Operationen Gebrauch macht. Letztendlich gehen 624 Speicherworte der Länge 32 als „Gedächtnis“ ein. Eine besonders elegante Implementierung findet sich in der zitierten Originalarbeit bzw. auf der Webseite Mersenne Twister Home Pa-

ge, auf die wir bzgl. weiterer Details verweisen (<http://www.math.sci.hiroshima-u.ac.jp/~m-mat/MT/emt.html>).

Genau genommen ist Mersenne-Twister die Sammelbezeichnung für eine von Matsumoto und Nishimura begründete Familie von Generatoren, die heute einen „Quasi-Standard“ darstellen und von vielen Software Systemen als Hauptgenerator übernommen wurden (vgl. https://en.wikipedia.org/wiki/Mersenne_Twister für eine aktuelle Liste). Dabei wurden typischerweise Implementierungen ausgewählt, die wie im Falle „minimaler Standard“ durch eine Integer Zahl (seed) gesteuert werden: Ausgehend von dieser einen Zahl werden intern die 624 benötigten Zahlen erzeugt, die für den Start des Mersenne Twister Algorithmus erforderlich sind. Dies ist bequem, gibt dem Analysten aber nur dann die volle Kontrolle und ermöglicht den systemübergreifenden portablen Einsatz, wenn auch die Regel, wie aus einer Seed jeweils die 624 Startwerte generiert werden, offen zugänglich ist.

Der klassische Mersenne Twister (MT19937) hat einen Schwachpunkt: Befindet sich die Iterationsfolge in einem Bereich von Dualzahlen, die sehr viele Nullen enthalten, so dauert es relativ lange bis dieser Bereich verlassen werden kann. Weiterentwicklungen, die diesen Schwachpunkt (slow zero excess recovery) vermeiden und teilweise auch schneller sind, laufen unter Namen wie SFMT (SIMD-oriented Fast Mersenne Twister), MTGP, TinyMT oder WELL (siehe Panneton et al.(2006) und <http://www.math.sci.hiroshima-u.ac.jp/~m-mat/MT/emt.html>). Diese Generatoren markieren den derzeitigen Standard an Zufallszahlengeneratoren für den Einsatz bei Simulationen vom Monte Carlo Typ. Sie sind aber nicht kryptographisch sicher; m.a.W. in der Kryptographie (Kryptologie) müssen für sichere Verschlüsselungen andere Generatoren Verwendung finden.

Bei den bisherigen Überlegungen stand die effiziente Berechnung der Zahlenfolge z_0, z_1, \dots , ihre Reproduzierbarkeit und ein hinreichend großer Zahlenvorrat im Vordergrund. Diese Zahlenfolge wird zu einer Folge von **Zufallszahlen**, wenn sie als Realisation einer Folge Z_0, Z_1, \dots von unabhängigen, auf dem Intervall $[0, 1]$ gleichverteilten Zufallsvariablen aufgefasst werden kann. Demzufolge müssen wir die erzeugten Zahlen z_0, z_1, \dots noch auf ihre „Zufälligkeit“ hin überprüfen. Hierzu stehen uns eine Reihe von statistischen Tests zur Verfügung, die unterschiedliche Aspekte, die wir mit dem Begriff Zufälligkeit in Verbindung bringen, beleuchten.

Mit Hilfe eines Anpassungstests (vgl. Anhang B.9) können wir zunächst überprüfen, ob die erzeugten Zahlen z_0, \dots, z_n als Realisationen einer $U[0, 1]$ -verteilten Zufallsvariablen Z aufgefasst werden können.

Führen wir einen χ^2 -**Anpassungstest** durch, so unterteilen wir das Intervall $[0, 1]$ in k gleichlange Teilintervalle. Handelt es sich tatsächlich um Realisationen bzgl. der Gleichverteilung, so können wir erwarten, dass in jedem Teilintervall annähernd dieselbe Anzahl von Zahlen liegt. Trifft dies nicht zu, sind also die Unterschiede zu „groß“, so spricht dies gegen eine Gleichverteilung und damit eine zentrale Voraussetzung von Zufallszahlen.

Als Alternative bietet sich der **Kolmogorov-Smirnov Test** an, der die maximale Abweichung der auf der Zahlenfolge basierenden empirischen Verteilungsfunktion von der Verteilungsfunktion der unterstellten Gleichverteilung (hier speziell der Winkelhalbierenden) misst. Ist dieser Abstand zu „groß“, so spricht dies auch hier gegen eine Gleichverteilung und damit eine zentrale Voraussetzung von Zufallszahlen.

Der **Anderson-Darling Test** basiert im Gegensatz zum Kolmogorov-Smirnov Test auf einer gewichteten Abweichung der auf der Zahlenfolge basierenden empirischen Verteilungsfunktion von der Verteilungsfunktion der unterstellten Gleichverteilung. Ist auch hier dieser Abstand zu „groß“, so spricht dies gegen eine Gleichverteilung und führt zur Ablehnung dieser Hypothese.

Schwieriger gestaltet sich die Überprüfung der Unabhängigkeit der erzeugten Zahlen. Wir beginnen mit dem **Runs Test**. Hierzu betrachten wir exemplarisch die Zahlen

0.87 0.15 0.23 0.45 0.69 0.32 0.30 0.19 0.24 0.65 0.18 .

Ausgehend von 0.87 erhalten wir mit 0.15 eine kleinere Zahl, die wir mit $-$ markieren, gefolgt von drei Zahlen 0.23, 0.45, 0.69 aufsteigender Größe, die wir mit $+$ markieren, wiederum gefolgt von drei Zahlen 0.32, 0.30, 0.19 absteigender Größe ($-$), zwei Zahlen 0.24, 0.65 aufsteigender Größe ($+$), einer kleineren Zahl 0.18 ($-$). Fassen wir die Vorzeichen zusammen, so ergibt sich das Schema (Alternativfolge)

$- + + + - - - + + -$

mit 5 Runs (Teilfolgen mit demselben Vorzeichen). Sind die Vorzeichenwechsel zu „selten“ oder zu „häufig“, wie in den Zahlenfolgen

0.08 0.18 0.23 0.36 0.42 0.55 0.63 0.72 0.89 0.91

bzw.

0.08 0.93 0.15 0.96 0.29 0.84 0.28 0.79 0.36 0.57 ,

so wird man die Unabhängigkeit der erzeugten Zahlen verwerfen.

Ein möglicher Test basiert auf der Länge der Runs und führt auf einen χ^2 -Anpassungstest, der die beobachtete Anzahl mit der bei Unabhängigkeit erwarteten Anzahl Runs der Länge i vergleicht.

Zieht man anstelle der Längen der einzelnen Runs lediglich die *Anzahl* A der Runs zur Beurteilung der Unabhängigkeit der Zahlen heran, so kann man für hinreichend großes n ($n > 25$) ausnutzen, dass A in guter Näherung normalverteilt ist und erhält mit der zugehörigen Verteilungsfunktion als Prüfgröße die Möglichkeit, bei zu „kleiner“ oder zu „großer“ Anzahl an Runs die Hypothese der Unabhängigkeit der Zahlen zu verwerfen.

Alternativ zu auf- und absteigenden Teilfolgen kann man auch die Teilfolgen oberhalb und unterhalb von 0.5, dem Erwartungswert bei unterstellter Gleichverteilung, betrachten. Für weitere Einzelheiten zu den Runs Tests siehe Anhang B.10.

Anstelle einzelner Zahlen wie bei der Überprüfung auf Gleichverteilung kann man auch d aufeinanderfolgende Zahlen (z_0, \dots, z_{d-1}) , (z_d, \dots, z_{2d-1}) , \dots , $(z_{nd-d}, \dots, z_{nd-1})$ zu Zahlentupeln zusammenfassen und mit Hilfe eines χ^2 -Anpassungstests auf Gleichverteilung auf $[0, 1]^d$ testen. Auf diese Weise (**Serial Test**) ist es möglich, Abhängigkeiten bei Teilfolgen der Länge d aufzudecken. Unterteilt man in jeder Dimension das Intervall $[0, 1]$ in m Teilintervalle der Länge $1/m$, so erhält man insgesamt $k = m^d$ Teilbereiche, für die man die Anzahl der beobachteten Zahlentupel mit der bei unterstellter Unabhängigkeit erwarteten Anzahl an Zahlentupel vergleicht. Mit zunehmender Dimension d stößt das Verfahren jedoch schnell an seine Grenzen, da die bei Anwendung des χ^2 -Anpassungstests geforderte Mindestanzahl an Beobachtungen in den einzelnen Teilbereichen unterschritten wird. Für weitere Einzelheiten siehe Anhang B.10.

Der **Gap Test** berücksichtigt stärker als der Serial Test die zeitliche Entwicklung der Zufallszahlen. Hierzu wird ein Intervall $[a, b) \subset [0, 1]$ festgelegt. Ein Wert $z_n \in [a, b)$ gilt als Erfolg; eine Lücke (Gap) ist die Anzahl von Zufallszahlen zwischen zwei Erfolgen. Sind Z_0, Z_1, \dots unabhängige, $U[0, 1]$ -verteilte Zufallsvariable, so ist die Anzahl X zwischen zwei Erfolgen geometrisch verteilt mit Werten in \mathbb{N}_0 und Parameter $b - a$. Mit Hilfe eines χ^2 -Anpassungstests kann man dann überprüfen, ob die beobachtete Anzahl von der erwarteten Anzahl von Lücken der Länge i , $i \in \mathbb{N}_0$, „zu stark“ abweicht oder nicht. Bei „zu starker“ Abweichung wird man schließlich die Hypothese der Unabhängigkeit der Zahlen verwerfen. Siehe Anhang B.10.

Lange galt der multiplikative Kongruenzgenerator (2.2) mit $a = 16807$ und $m = 2147483647$ als minimaler Standard für die Anforderungen an einen „guten“ Zufallszahlengenerator. Dies nehmen wir zum Anlass, die „Güte“ dieses Generators anhand der vorgestellten statistischen Verfahren zu überprüfen.

Hierzu wählen wir $x_0 = 1$ als Quelle (Seed) und betrachten die ersten 100 erzeugten Zufallszahlen. Im Hinblick auf eine übersichtliche Darstellung runden wir die Zufallszahlen auf 6 Dezimalstellen und geben die gerundeten Zufallszahlen multipliziert mit 10^6 in Tab. 2.2 an. Den Berechnungen liegen jedoch die exakten Werte zugrunde. Dies trifft insbesondere auf die erste Zufallszahl zu, deren exakter Wert $(2^{31} - 1)^{-1}$ ist (und selbst multipliziert mit 10^6 in Tab. 2.2 noch den Wert 0 annimmt).

000000	000008	131538	755605	458650	532767	218959	047045	678865	679296
934693	383502	519416	830965	034572	053462	529700	671149	007698	383416
066842	417486	686773	588977	930436	846167	526929	091965	653919	415999
701191	910321	762198	262453	047465	736082	328234	632639	756410	991037
365339	247039	982550	722660	753356	651519	072686	631635	884707	272710
436411	766495	477732	237774	274907	359265	166507	486517	897656	909208
060564	904653	504523	516292	319033	986642	493977	266145	090733	947764
073749	500707	384142	277082	913817	529747	464446	940980	050084	761514
770205	827817	125365	015868	688455	868247	629543	736225	725412	999458
888572	233195	306322	351015	513274	591114	845982	412081	841511	269317

Tabelle 2.2. 100 Zufallszahlen (multipliziert mit 10^6)

Die Implementierung des Generators auf einem Rechner mit 64 Bit Arithmetik ist unproblematisch. Beispielsweise reicht es aus, in Excel 2007 den Befehl `REST(x_n *16807;2147483647)` einzugeben, um den Wert x_{n+1} zu berechnen. Erhält man (beginnend mit $x_0 = 1$) den Wert $x_{10000} = 1043618065$, so ist der Generator richtig implementiert. Bei einem Rechner mit einer 32 Bit Arithmetik bedarf es jedoch einer sorgfältigen Implementierung, um einen Zahlenüberlauf zu verhindern. Hinweise findet der interessierte Leser in Park, Miller (1988) und Marsaglia et al. (1993).

Für die Anwendung des χ^2 -Anpassungstests unterteilen wir das Einheitsintervall in 10 Teilintervalle (Klassen) der Länge 0.1. Die beobachteten Klassenhäufigkeiten sind den erwarteten Klassenhäufigkeiten in Abb. 2.1 (links) gegenübergestellt. Die zugehörige Prüfgröße T (vgl. Beispiel B.14) nimmt den Wert $t = 8.60$ an. Dieser Wert von T führt bei einer Irrtumswahrscheinlichkeit $\alpha = 0.05$ zur Annahme der Hypothese (Realisationen der $U[0, 1]$ -Verteilung). Der Kolmogorov-Smirnov Test ($t = 0.0595$) kommt zu demselben Ergebnis. Ebenso der Anderson-Darling Test ($t = 0.95$). Der beiden Prüfgrößen (siehe (B.26) und (B.28)) zugrunde liegende Abstand der empirischen Verteilungsfunktion von der Verteilungsfunktion der $U[0, 1]$ -Verteilung geht aus Abb. 2.1 (rechts) hervor. Die formale Berechnung der drei Prüfgrößen und zugehörigen Ablehnungsbereiche nehmen wir in den Beispielen B.14, B.17 und B.18 vor.

Bei Anwendung des Runs Tests erhält man 62 Teilfolgen aufsteigender (+) oder absteigender (-) Größe. Diese sind in Tab. 2.3 aufgeführt. Die Länge der Runs variiert zwischen 1 und 5. Der χ^2 -Anpassungstest (vgl. Beispiel B.21) mit den Längen 1, 2, 3 und > 3 als Klassen kommt bei einer Irrtumswahr-

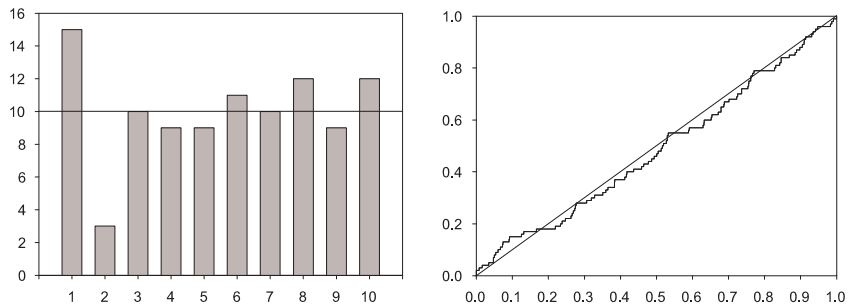


Abb. 2.1. Klassenhäufigkeiten (links) und empirische Verteilungsfunktion (rechts)

scheinlichkeit von 5% zur Annahme der Hypothese der Unabhängigkeit. Bei alleiniger Betrachtung der Anzahl der Runs (vgl. Beispiel B.21) kommt man zu demselben Ergebnis.

+	+	+	-	+	-	-	+	+	+	-	+	+	-	+	+	+	-	+	-	+	+	-	+	-
-	-	+	-	+	+	-	-	-	+	-	+	+	+	-	-	+	-	+	-	-	+	+	-	+
+	-	-	+	+	-	+	+	+	-	+	-	+	-	+	-	-	-	+	-	+	-	-	+	-
-	+	-	+	+	+	-	-	+	+	-	+	-	+	-	-	+	+	+	+	+	+	-	+	-

Tabelle 2.3. Runs up and down

Betrachtet man die Runs oberhalb und unterhalb von 0.5, so kommt man auch hier zur Annahme der Hypothese der Unabhängigkeit. Die beobachteten Runs sind in Tab. 2.4 aufgeführt; auf alle weiteren Einzelheiten zum Test gehen wir in Beispiel B.22 ein.

-	-	-	+	-	+	-	-	+	+	+	-	+	+	-	-	+	+	-	-	-	-	+	+	+
+	+	-	+	-	+	+	+	-	-	+	-	+	+	+	-	-	+	+	+	+	-	+	+	-
-	+	-	-	-	-	-	-	+	+	-	+	+	+	-	+	-	-	-	+	-	+	-	-	+
+	-	+	-	+	+	+	-	-	+	+	+	+	+	+	+	+	-	-	-	+	+	+	-	+

Tabelle 2.4. Runs oberhalb und unterhalb von 0.5

Bei Anwendung des Serial Tests können wir aufgrund der sehr kleinen Anzahl von Zufallszahlen nur wenige Teilintervalle in jeder Dimension zulassen. Daher betrachten wir anstelle der 100 Zufallszahlen aus Tab. 2.2 die ersten 100 Zahlentupel, die sich aus den ersten $100 \cdot d$ Zufallszahlen ergeben. Mit den Abb. 2.2 und 2.3 vermitteln wir zunächst einen optischen Eindruck über die Verteilung der Zahlentupel im Einheitsquadrat ($d = 2$) und Einheitswürfel ($d = 3$).

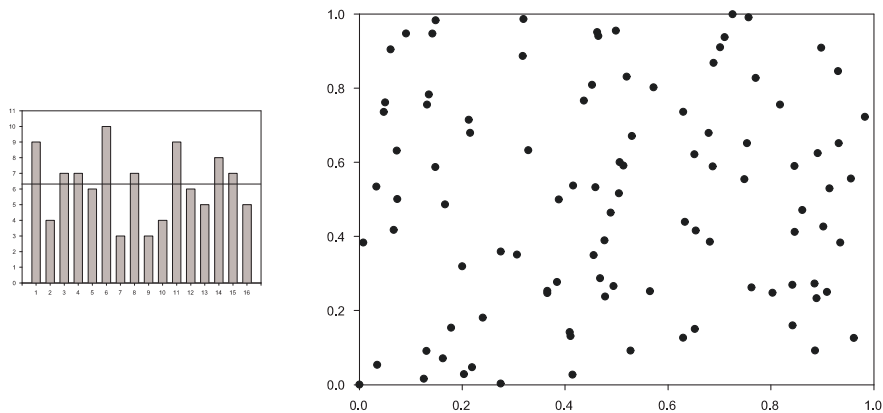


Abb. 2.2. Serial Test ($d = 2$): Klassenhäufigkeiten und Anordnung der Zahlentupel

Der χ^2 -Anpassungstest auf Gleichverteilung über $[0, 1]^d$ muss sich auf wenige Klassen in jeder Dimension beschränken, um die Faustregeln zur Anwendbarkeit des Tests einzuhalten. Die sich für $k = 16$ im Falle $d = 2$ und $k = 8$ im Falle $d = 3$ ergebenden Klassenhäufigkeiten sind den erwarteten Klassenhäufigkeiten in den Abb. 2.2 und 2.3 gegenübergestellt. Der Test führt in beiden Fällen zur Annahme der Hypothese der Unabhängigkeit.

Abschließend betrachten wir noch den Gap Test. Bei Wahl des Intervalls $[0.9, 1.0)$ treten 12 Erfolge auf (Tab. 2.5) verbunden mit den Wartezeiten (Lücken) 10, 13, 6, 7, 2, 16, 1, 3, 3, 4, 2, 11. Fassen wir die Wartezeiten 0 – 6 zu einer Klasse und die restlichen Wartezeiten zu einer zweiten Klasse zusammen, so führt der resultierende χ^2 -Anpassungstest (siehe Beispiel B.24) zur Annahme der Hypothese der Unabhängigkeit.

0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	1	0	0
0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

Tabelle 2.5. Gap Test: Zahlen im Intervall $[0.9, 1.0)$

Zusammenfassend können wir festhalten, dass die überzeugenden statistischen Eigenschaften des multiplen Kongruenzgenerators (2.2) mit $a = 16807$ und $m = 2147483647$ für dessen Einsatz in Simulationsprojekten sprechen. Andererseits könnte bei größeren Projekten oder dem unprofessionellen Umgang mit Zufallszahlen der Vorrat von $2^{31} - 1$ Zahlen sehr schnell aufgebraucht

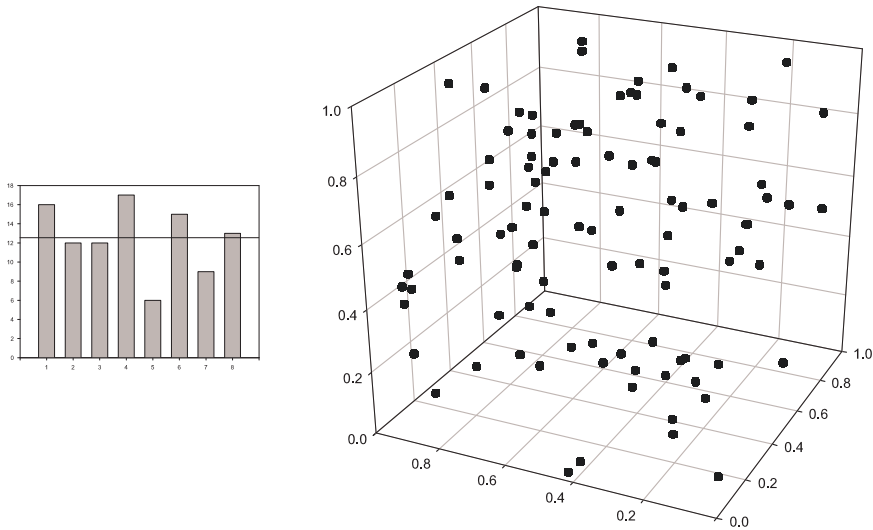


Abb. 2.3. Serial Test ($d = 3$): Klassenhäufigkeiten und Anordnung der Zahlentupel

sein, was zu einem Eintritt in einen neuen Zyklus und damit zu einer möglicherweise nicht zu erkennenden Verfälschung der Ergebnisse führen würde.

Die folgende Überslagsrechnung verdeutlicht diesen Aspekt: Wir erzeugen eine Folge x_0, x_1, \dots von unabhängigen Realisationen der Standardnormalverteilung. Dabei interessieren wir uns für die mittlere Wartezeit auf die erste Realisation, die einen Wert größer als 3 annimmt. Diese mittlere Wartezeit wollen wir bis auf einen Fehler ± 0.1 schätzen. Sei also $p = 1 - \Phi(3) = 0.00135$ die Wahrscheinlichkeit, eine Realisation größer als 3 zu erzeugen. Dann ist die Wartezeit W auf das Eintreten dieses Ereignisses auf \mathbb{N}_0 geometrisch verteilt (vgl. Anhang A.3) mit Erwartungswert $\mu = (1 - p)/p = 739.7$ und Standardabweichung $\sigma = \sqrt{1 - p}/p = 740.2$.

μ ist also genau der Wert, den wir bis auf ± 0.1 schätzen wollen. Hierzu führen wir n Simulationsläufe durch, realisieren also n Wartezeiten w_1, \dots, w_n und betrachten das arithmetische Mittel $\bar{w} = (w_1 + \dots + w_n)/n$ der Wartezeiten als Schätzwert für μ . Für hinreichend großes n (was auf unsere Situation zutrifft) können wir \bar{W} als normalverteilt mit Erwartungswert μ und Standardabweichung σ/\sqrt{n} annehmen (siehe Satz A.11). Um den vorgegebenen Fehler bei einer Irrtumswahrscheinlichkeit von 1% einzuhalten, müssen wir

$$c_{1-\alpha}\sigma/\sqrt{n} = 2.56 \cdot 740.2/\sqrt{n} \leq 0.1$$

einhalten (vgl. (B.17)) und damit $n \geq 358623716$ wählen. Zu bedenken haben wir noch, dass für jeden Simulationslauf im Mittel 739.7 Realisationen der Normalverteilung erforderlich sind. Verhalten wir uns wenig professionell und erzeugen eine Realisation der Normalverteilung mit Hilfe von 12 Zufallszahlen (vgl. Beispiel A.5), so benötigen wir insgesamt

$$12 \cdot 739.7 \cdot n \geq 12 \cdot 739.7 \cdot 358623716 = 3.2 \cdot 10^{12}$$

Zufallszahlen. Das sprengt den Wertevorrat von $2147483646 = 2.1 \cdot 10^9$ Zufallszahlen des Generators.

Das Beispiel mag künstlich erscheinen. Eine vergleichbare Situation (mit unbekannten Parametern) tritt bei der Analyse von Qualitätsregelkarten mit Gedächtnis im Rahmen der statistischen Fertigungsüberwachung auf. Als Ausweg bieten sich z.B. der Mersenne-Twister oder, wenn man an dem bisherigen Generator festhalten will, varianzreduzierende Verfahren an, auf die wir in Kapitel 7 näher eingehen werden.

Bei den Zufallszahlengeneratoren, die in der *neueren* Software implementiert sind, können wir davon ausgehen, dass sie umfangreichen statistischen Tests unterzogen wurden und diese zu zufriedenstellenden Ergebnissen geführt haben. Daher unterstellen wir im Folgenden, dass der (zu verwendende) Zufallszahlengenerator eine Folge z_0, z_1, \dots von Zahlen erzeugt, die als unabhängige Realisationen einer $U[0, 1]$ -verteilten Zufallsvariablen Z aufgefasst werden können.

Wir schließen uns einer Reihe von führenden Experten an, die alle von der Verwendung von selbst gestrickten Generatoren abraten. Wenn dies in Ausnahmefällen doch sinnvoll erscheint, so verweisen wir auf die klassische Testbatterie Diehard von Marsaglia (d.h. eine unter bestimmten Gesichtspunkten zusammengestellte Kombination unterschiedlicher Tests), die Weiterentwicklung Dieharder von R.G. Brown sowie auf die Empfehlungen und Testvorschläge des National Institute of Standards and Technology (NIST) für die kryptologische Verwendung von Generatoren.

2.2 Die Inversionsmethode

Wir kommen noch einmal auf die Simulation eines Wurfes mit einem fairen Würfel zurück. Hierzu bezeichne X die Augenzahl, die sich bei dem Wurf ergibt. X ist eine Zufallsvariable mit Werten in der Menge $\{1, \dots, 6\}$ und der Zähldichte

$$P(X = i) = 1/6 \quad \text{für } i = 1, \dots, 6.$$

Die Zufallsvariable X können wir durch eine $U[0, 1]$ -verteilte Zufallsvariable Z ausdrücken. Insbesondere gilt

$$X = \begin{cases} 1 & \text{für } 0 \leq Z \leq 1/6 \\ 2 & \text{für } 1/6 < Z \leq 2/6 \\ 3 & \text{für } 2/6 < Z \leq 3/6 \\ 4 & \text{für } 3/6 < Z \leq 4/6 \\ 5 & \text{für } 4/6 < Z \leq 5/6 \\ 6 & \text{für } 5/6 < Z \leq 1 \end{cases} \quad (2.4)$$

Etwas kompakter können wir auch

$$X = \lceil 6Z \rceil$$

schreiben, wobei $\lceil x \rceil$ (ceiling von x) für $x \in \mathbb{R}$ die kleinste ganze Zahl bezeichnet, die größer oder gleich x ist, z.B. $\lceil 3.6 \rceil = 4$. Später werden wir noch Gebrauch machen von $\lfloor x \rfloor$ (floor von x), der größten ganzen Zahl, die kleiner oder gleich x ist.

Die Darstellung von X als Funktion von Z führt auf die Grundidee der Inversionsmethode: Wir ordnen einer Zufallszahl z eine Realisation x einer nichtgleichverteilten Zufallsvariablen, hier einer diskreten Gleichverteilung auf der Menge $\{1, \dots, 6\}$ zu. Liegt z.B. z im Intervall $(1/6, 2/6]$, so ordnen wir z die Realisation $x = 2$ zu. Im Allgemeinen ist die Transformation jedoch nicht so einfach wie bei unserem Würfelexperiment. Sie hängt wesentlich von den Eigenschaften der Verteilungsfunktion der zu erzeugenden Zufallsvariablen ab.

Die Verteilungsfunktion F einer Zufallsvariablen X ist eine monoton wachsende Funktion. Sie ist aber nicht notwendigerweise streng monoton wachsend und stetig. Daher müssen wir den Begriff der Umkehrfunktion (Inversen) allgemeiner fassen und verstehen im Folgenden unter

$$F^{-1}(z) := \min\{x \in \mathbb{R} \mid F(x) \geq z\}, \quad 0 < z < 1,$$

die kleinste reelle Zahl x , für die $F(x) \geq z$ gilt.

Ist Z eine auf dem Intervall $(0, 1)$ gleichverteilte Zufallsvariable, so ist auch $F^{-1}(Z)$ als Funktion einer Zufallsvariablen wieder eine Zufallsvariable. Darüber hinaus gilt der folgende zentrale Satz, der den Ausgangspunkt für die Erzeugung *beliebig* verteilter Zufallsvariablen bildet.

Satz

2.4

Sei F eine Verteilungsfunktion und sei Z eine auf dem Intervall $(0, 1)$ gleich-

verteilte Zufallsvariable. Dann besitzt die Zufallsvariable

$$X = F^{-1}(Z)$$

die Verteilungsfunktion F .

Beweis: Da F (als Verteilungsfunktion) rechtsstetig ist, gilt zunächst $z \leq F(x)$ genau dann wenn $F^{-1}(z) \leq x$. Die Behauptung folgt nun unmittelbar aus

$$P(F^{-1}(Z) \leq x) = P(Z \leq F(x)) = F(x), \quad x \in \mathbb{R}. \quad \square$$

Somit können wir die Transformation $x = F^{-1}(z)$ einer Zufallszahl z als Realisation einer Zufallsvariablen X mit der Verteilungsfunktion F auffassen. Diese Vorgehensweise der Erzeugung einer Realisation einer Zufallsvariablen X bezeichnet man als **Inversionsmethode**.

Zusammenfassend fallen die folgenden Rechenschritte zur Erzeugung einer Realisation x einer Zufallsvariablen X mit der Verteilungsfunktion F an:

1. Erzeuge Zufallszahl $z \in (0, 1)$.
2. Setze $x = F^{-1}(z)$.

Hauptsächlich treten zwei Spezialfälle auf:

- (a) *X ist eine stetige Zufallsvariable.* Ist F auf der Menge $\{x' \in \mathbb{R} \mid 0 < F(x') < 1\}$ streng monoton wachsend, so ist x die Lösung der Gleichung $F(x) = z$.
- (b) *X ist eine diskrete Zufallsvariable.* Nimmt X Werte in der Menge $\{x_0, x_1, \dots\}$ an und ist x_k der kleinste Wert von X , für den $F(x_k) \geq z$ gilt, so ist $x = x_k$.

Abb. 2.4 veranschaulicht beide Spezialfälle.

Neben diesen beiden wichtigen Spezialfällen treten gelegentlich Mischformen auf, die sich bspw. aus der Mischung von stetigen und/oder diskreten Verteilungsfunktionen ergeben. Abb. 2.5 veranschaulicht die Inversionsmethode angewandt auf die Mischung zweier stetiger Verteilungsfunktionen (links) mit einem Bereich, der die Wahrscheinlichkeit Null hat, und die Mischung einer stetigen und einer diskreten Verteilungsfunktion (rechts).

Gewöhnlich haben wir nicht nur eine, sondern eine Vielzahl von Realisationen einer Zufallsvariablen X zu erzeugen. Daher ist es angebracht, die folgenden

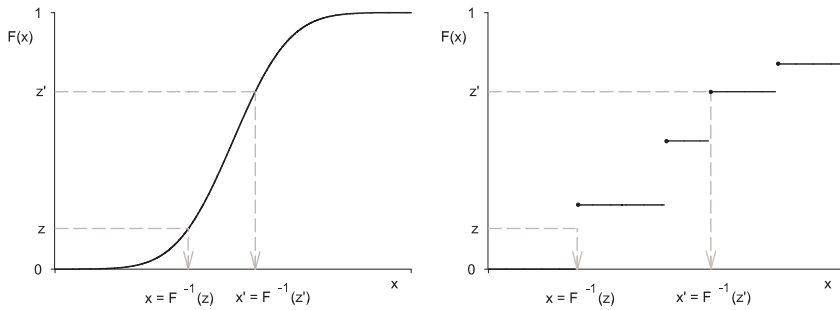


Abb. 2.4. Inversionsmethode bei stetiger (links) und diskreter (rechts) Zufallsvariable

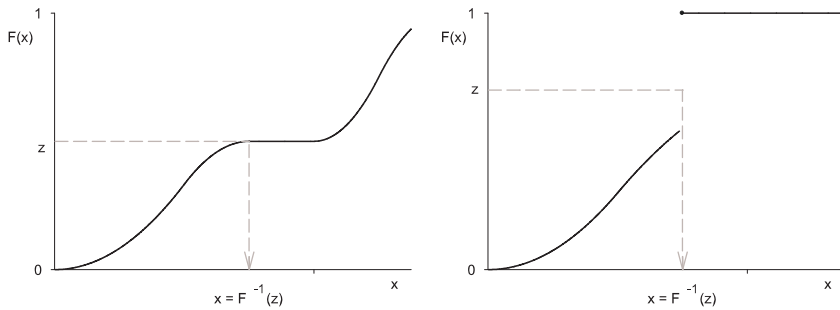


Abb. 2.5. Inversionsmethode bei Mischung von Verteilungsfunktionen

Modifikationen der Inversionsmethode im Hinblick auf eine Zeitersparnis in Betracht zu ziehen:

Bemerkung

2.5

- (a) Mit Z ist auch $1 - Z$ auf dem Intervall $[0, 1]$ gleichverteilt (siehe (A.4)). Somit ist neben z auch $1 - z$ eine Zufallszahl. Dieser Zusammenhang bietet uns die Möglichkeit, eine Zufallszahl z zu erzeugen, aber die Inversionsmethode nicht auf z , sondern auf $1 - z$ anzuwenden, d.h. x nicht als Realisation von $F^{-1}(z)$ zu bestimmen, sondern als Realisation von $F^{-1}(1 - z)$. U.a. bei der Exponentialverteilung entfällt dann die Berechnung von $1 - z$. Siehe (2.5) und (2.6).
- (b) Ist $g(z)$ eine stetige, streng monoton wachsende Transformation einer Zufallszahl z , so stimmen $\lceil g(z) \rceil$ und $\lfloor g(z) \rfloor + 1$ bis auf alle $g(z) \in \mathbb{Z}$ überein. Diesen Unterschied können wir jedoch vernachlässigen, da eine

Zufallszahl Realisation einer $U[0, 1]$ -verteilten und damit stetigen Zufallsvariable ist.

Um die Notation zu vereinfachen, bezeichnen wir im Folgenden die erzeugte Realisation x einer Zufallsvariablen X mit der Dichte/Zähldichte f häufig auch als Zufallszahl bzgl. f oder, wenn es sich dabei um eine Standardverteilung wie die Exponentialverteilung mit Parameter λ handelt, als

$\text{Expo}(\lambda)$ -verteilte Zufallszahl. (Die wichtigsten Verteilungen, ihre grundlegenden Eigenschaften sowie die von uns verwendete abkürzende Schreibweise sind Gegenstand von Anhang A.3.)

Die Inversionsmethode kann grundsätzlich auf jede Verteilung angewendet werden; der mit der Inversion verbundene Rechenaufwand ist jedoch sehr unterschiedlich.

2.6 **Beispiel** (Gleichverteilung auf dem Intervall $[a, b]$)

Eine $U[a, b]$ -verteilte Zufallsvariable X hat die Verteilungsfunktion

$$F(x) = \frac{x - a}{b - a}, \quad a \leq x \leq b.$$

Zu gegebener Zufallszahl $z \in (0, 1)$ erfüllt die kleinste reelle Zahl x , für die $F(x) \geq z$ gilt, die Bedingung $F(x) = z$. Diese ist äquivalent zu $x - a = (b - a)z$ und wir erhalten

$$x = a + (b - a)z$$

als Realisation von X . \diamond

2.7 **Beispiel** (Exponentialverteilung)

Eine $\text{Expo}(\alpha)$ -verteilte Zufallsvariable X hat die Verteilungsfunktion

$$F(x) = 1 - e^{-\alpha x}, \quad x \geq 0.$$

Zu gegebener Zufallszahl $z \in (0, 1)$ erfüllt die kleinste reelle Zahl x , für die $F(x) \geq z$ gilt, die Bedingung $F(x) = z$. Diese ist äquivalent zu $e^{-\alpha x} = 1 - z$ und $-\alpha x = \ln(1 - z)$. Damit erhalten wir

$$x = -\frac{1}{\alpha} \ln(1 - z) \tag{2.5}$$

oder unter Einbeziehung von Bemerkung 2.5(a)

$$x = -\frac{1}{\alpha} \ln z \quad (2.6)$$

als Realisation von X . \diamond

Beispiel (Pareto-Verteilung)

2.8

Eine $Pareto(\lambda, \kappa)$ -verteilte Zufallsvariable X hat die Verteilungsfunktion

$$F(x) = 1 - \left(\frac{\kappa}{\kappa + x} \right)^\lambda, \quad x \geq 0.$$

Zu gegebener Zufallszahl $z \in (0, 1)$ erfüllt die kleinste reelle Zahl x , für die $F(x) \geq z$ gilt, die Bedingung $F(x) = z$. Diese ist äquivalent zu $\left(\frac{\kappa}{\kappa + x} \right)^\lambda = 1 - z$ und wir erhalten

$$x = \kappa(1 - z)^{-1/\lambda} - \kappa$$

oder unter Einbeziehung von Bemerkung 2.5(a) $x = \kappa z^{-1/\lambda} - \kappa$ als Realisation von X . \diamond

Beispiel (Weibull-Verteilung)

2.9

Wendet man die Inversionsmethode auf eine $Weibull(\alpha, \beta)$ -verteilte Zufallsvariable X mit der Verteilungsfunktion

$$F(x) = 1 - e^{-\alpha x^\beta}, \quad x \geq 0,$$

an, so erhält man zu gegebener Zufallszahl $z \in (0, 1)$ unter Einbeziehung von Bemerkung 2.5(a)

$$x = \left(-\frac{1}{\alpha} \ln z \right)^{1/\beta} \quad (2.7)$$

als Realisation von X . Für den Spezialfall ($\beta = 1$) einer $Expo(\alpha)$ -Verteilung stimmt (2.7) mit (2.6) überein. \diamond

Beispiel (Dreieck-Verteilung)

2.10

Für eine $Dreieck(a, b, m)$ -verteilte Zufallsvariable X , die die Verteilungsfunktion

$$F(x) = \begin{cases} \frac{(x-a)^2}{(b-a)(m-a)} & \text{für } a < x \leq m \\ 1 - \frac{(b-x)^2}{(b-a)(b-m)} & \text{für } m < x \leq b \end{cases}$$

besitzt, erhält man als Realisation x zu gegebener Zufallszahl $z \in (0, 1)$ den Wert

$$x = \begin{cases} a + \sqrt{z(b-a)(m-a)} & \text{für } z \leq (m-a)/(b-a) \\ b - \sqrt{(1-z)(b-a)(b-m)} & \text{für } z > (m-a)/(b-a) \end{cases} \quad \diamond$$

2.11 **Beispiel** (Diskrete Gleichverteilung auf der Menge $\{1, \dots, m\}$)

Eine $UD\{1, \dots, m\}$ -verteilte Zufallsvariable X hat die Verteilungsfunktion

$$F(x) = i/m \quad \text{für } i \leq x < i+1, i \in \{1, \dots, m\}.$$

Zu gegebener Zufallszahl $z \in (0, 1)$ ist die kleinste reelle Zahl x , für die $F(x) \geq z$ gilt, einer der Werte, den die Zufallsvariable X annehmen kann. Sei k dieser Wert. Als *kleinster* Wert erfüllt k die Bedingung $k/m \geq z > (k-1)/m$. Diese ist äquivalent zu $k \geq mz > k-1$. Damit ist

$$k = \lceil m \cdot z \rceil$$

die gesuchte Realisation von X . Zusammen mit Bemerkung 2.5(b) kann man alternativ auch $k = \lfloor m \cdot z \rfloor + 1$ wählen. \diamond

2.12 **Beispiel** (Geometrische Verteilung)

Eine $Geo_{\mathbb{N}}(p)$ -verteilte Zufallsvariable X hat die Verteilungsfunktion

$$F(x) = 1 - (1-p)^i \quad \text{für } i \leq x < i+1, i \in \mathbb{N}.$$

Zu gegebener Zufallszahl $z \in (0, 1)$ ist die kleinste reelle Zahl x , für die $F(x) \geq z$ gilt, ein Wert von X . Sei $k \in \mathbb{N}$ dieser Wert. Dann erfüllt k die Bedingung

$$1 - (1-p)^k \geq z > 1 - (1-p)^{k-1}.$$

Diese ist äquivalent zu $(1-p)^k \leq 1-z < (1-p)^{k-1}$ und $k \geq \ln(1-z)/\ln(1-p) > k-1$. Damit ist

$$k = \left\lceil \frac{\ln(1-z)}{\ln(1-p)} \right\rceil$$

die gesuchte Realisation von X . Unter Einbeziehung von Bemerkung 2.5 kann man auch $k = \left\lfloor \frac{\ln z}{\ln(1-p)} \right\rfloor + 1$ als Realisation von X festlegen.

Ist X' eine $\text{Geo}_{N_0}(p)$ -verteilte Zufallsvariable (vgl. Anhang A.3), so ist $X' = X - 1$ und wir erhalten mit $k' = k - 1$ die angestrebte Realisation von X' . \diamond

Beispiel (Diskrete Verteilung)

2.13

Eine diskrete Zufallsvariable X , die die Werte $x_1 < x_2 < \dots < x_N$ mit den Wahrscheinlichkeiten p_1, p_2, \dots, p_N annimmt, hat die Verteilungsfunktion

$$F(x) = \sum_{j=1}^i p_j \quad \text{für} \quad x_i \leq x < x_{i+1}, \quad i \in \{1, \dots, N\}.$$

Zu gegebener Zufallszahl $z \in (0, 1)$ ist x_1 die gesuchte Realisation von X , falls $F(x_1) = p_1 \geq z$ gilt und x_i ($i > 1$) die gesuchte Realisation, falls

$$F(x_i) = p_1 + \dots + p_i \geq z > p_1 + \dots + p_{i-1} = F(x_{i-1}).$$

Auf diese Weise zerfällt das Intervall $(0, 1)$ in disjunkte Teilintervalle $I_1 = (0, p_1]$, $I_2 = (p_1, p_1 + p_2]$, \dots , $I_N = (p_1 + \dots + p_{N-1}, 1)$ und der Wert x von X , der dem Teilintervall zugeordnet ist, in das die Zufallszahl z fällt, wird als Realisation von X festgelegt.

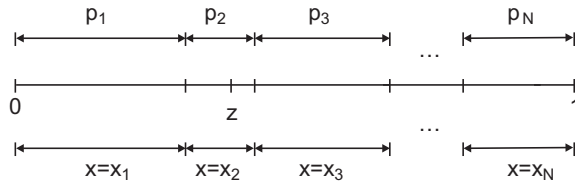


Abb. 2.6. Inversionsmethode bei diskreter Zufallsvariable

Abb. 2.6 veranschaulicht die Situation: Zur Bestimmung einer Realisation x von X wird das Intervall $[0, 1]$ in N disjunkte Teilintervalle der Längen p_1, \dots, p_N aufgeteilt und der Wert von X , der dem Teilintervall zugeordnet ist, in das die Zufallszahl z fällt, ergibt sich als Realisation, hier $x = x_2$. \diamond

Das Verfahren lässt sich unmittelbar auf eine diskrete Zufallsvariable mit abzählbar vielen Werten übertragen. Es ist jedoch nur dann effizient, wenn es gelingt, die Suche nach dem Teilintervall, in das die Zufallszahl fällt, effizient zu gestalten.

2.14 Beispiel (Binomialverteilung; Inversionsmethode)

Sei X eine $\text{Bin}(n, p)$ -verteilte Zufallsvariable. Die zugehörige Zähldichte

$$P(X = i) = \binom{n}{i} p^i (1-p)^{n-i}, \quad i = 0, \dots, n, \quad (2.8)$$

lässt sich, ausgehend von $p_0 = (1-p)^n$, rekursiv berechnen gemäß

$$p_{i+1} = \frac{n-i}{i+1} \frac{p}{1-p} p_i, \quad i = 0, \dots, n-1.$$

Unter Berücksichtigung dieser zusätzlichen Struktur ergibt die Inversionsmethode das folgende Verfahren zur Erzeugung einer Realisation x von X .

1. Erzeuge eine Zufallszahl z .
2. Setze $k = 0$, $a = \frac{p}{1-p}$, $b = (1-p)^n$, $F = b$.
3. Ist $z \leq F$, setze $x = k$ and stoppe. Andernfalls fahre mit Schritt 4 fort.
4. Setze $b = b \cdot a \frac{n-k}{k+1}$, $F = F + b$, $k = k + 1$ und fahre mit Schritt 3 fort.

Für den Spezialfall $n = 1$ (Bernoulli-Verteilung) erhalten wir zu gegebener Zufallszahl z die Realisation $x = 0$ im Falle $z \leq 1-p$ und die Realisation $x = 1$ im Falle $z > 1-p$. Häufiger angewandt wird das folgende, zur Inversionsmethode äquivalente Verfahren zur Erzeugung einer Realisation x einer $\text{Ber}(p)$ -Verteilung:

1. Erzeuge eine Zufallszahl z .
2. Ist $z \leq p$, setze $x = 1$. Andernfalls setze $x = 0$. \diamond

Beispiel (Poisson-Verteilung; Inversionsmethode)**2.15**

Sei X eine $Poi(\lambda)$ -verteilte Zufallsvariable. Die zugehörige Zähldichte

$$P(X = i) = \frac{\lambda^i}{i!} e^{-\lambda}, \quad i \in \mathbb{N}_0,$$

lässt sich, ausgehend von $p_0 = e^{-\lambda}$, rekursiv berechnen gemäß

$$p_{i+1} = \frac{\lambda}{i+1} p_i, \quad i \in \mathbb{N}_0.$$

Unter Berücksichtigung dieser zusätzlichen Struktur ergibt die Inversionsmethode das folgende Verfahren zur Erzeugung einer Realisation x von X .

1. Erzeuge eine Zufallszahl z .
2. Setze $k = 0$, $p = e^{-\lambda}$, $F = p$.
3. Ist $z \leq F$, setze $x = k$ und stoppe. Andernfalls fahre mit 4. fort.
4. Setze $p = p \cdot \frac{\lambda}{k+1}$, $F = F + p$, $k = k + 1$ und fahre mit 3. fort. \diamond

Beispiel (negative Binomialverteilung; Inversionsmethode)**2.16**

Sei X eine $NegBin(\alpha, p)$ -verteilte Zufallsvariable. Die zugehörige Zähldichte

$$P(X = i) = \frac{\Gamma(\alpha + i)}{\Gamma(\alpha) i!} p^\alpha (1 - p)^i, \quad i \in \mathbb{N}_0,$$

lässt sich, ausgehend von $p_0 = p^\alpha$, rekursiv berechnen gemäß

$$p_{i+1} = \frac{\alpha + i}{i + 1} (1 - p) p_i, \quad i \in \mathbb{N}_0.$$

Unter Berücksichtigung dieser zusätzlichen Struktur ergibt die Inversionsmethode das folgende Verfahren zur Erzeugung einer Realisation x von X .

1. Erzeuge eine Zufallszahl z .
2. Setze $k = 0$, $q = p^\alpha$, $F = q$.
3. Ist $z \leq F$, setze $x = k$ und stoppe. Andernfalls fahre mit 4. fort.
4. Setze $q = q \cdot \frac{\alpha + k}{k + 1} (1 - p)$, $F = F + q$, $k = k + 1$ und fahre mit 3. fort. \diamond

Die Inversionsmethode stößt an ihre Grenzen, wenn x nur durch *numerische* Lösung der Gleichung $F(x) = z$ (z.B. mit Hilfe der Bisektionsmethode) berechnet werden kann oder die Verteilungsfunktion (wie im Falle der Normalverteilung) nur in Tabellenform vorliegt. In solchen Fällen bieten sich als Alternative die Verwerfungsmethode und eine Reihe weiterer, spezielle Eigenschaften der Verteilung oder den Zusammenhang bestimmter Verteilungen ausnutzende Verfahren an.

2.3 Die Verwerfungsmethode

Sie möchten eine der Zahlen $1, \dots, 5$ zufällig (d.h. mit Wahrscheinlichkeit $1/5$) auswählen. Hierzu steht Ihnen ein fairer Würfel zur Verfügung. Sie werfen den Würfel. Ergibt der Wurf eine der Augenzahlen 1, 2, 3, 4 oder 5, so akzeptieren Sie das Ergebnis als die gesuchte Zahl. Ergibt der Wurf jedoch die Augenzahl 6, so lehnen Sie das Ergebnis ab, werfen den Würfel ein zweites Mal und entscheiden erneut. Sie fahren schließlich fort, indem Sie solange den Würfel werfen, bis Sie erstmals eine Augenzahl kleiner als 6 erhalten und akzeptieren dann diese Augenzahl als die gesuchte Zahl. Simulieren Sie noch den Wurf mit dem fairen Würfel auf einem Computer, so haben Sie ein besonders anschauliches Beispiel der Vorgehensweise bei der Verwerfungsmethode. Wir werden später noch sehen, dass im Mittel 1.2 Würfe benötigt werden, um die gesuchte Zahl zu erzeugen.

Das Experiment zeigt, dass wir *keine* Realisation der uns eigentlich interessierenden Zufallsvariablen X erzeugen, sondern eine Realisation einer Zufallsvariablen Y . Über diese von Y erzeugte Realisation wird dann in einem zweiten Schritt entschieden, sie als Realisation von X zu akzeptieren oder zu verwerfen. Diese Vorgehensweise ist natürlich nur dann sinnvoll, wenn eine Realisation von Y „einfach“ zu erhalten ist und die Verteilung von Y „nahe“ bei der von X liegt.

Kommen wir noch einmal auf die Exponentialverteilung zurück. Wir wissen bereits (Beispiel 2.7), dass sich eine exponentialverteilte Zufallszahl mit Hilfe der Inversionsmethode effizient erzeugen lässt. Daher wird man an der Inversionsmethode festhalten. Darüber hinaus bietet sich die Exponentialverteilung als „einfache“ Verteilung im Rahmen der Verwerfungsmethode an.

Für die $Beta(2, 4)$ -Verteilung, um ein einfaches Beispiel zu nennen, ist die Inversionsmethode nicht ohne Weiteres anwendbar. Hier bietet sich die Verwerfungsmethode an. Siehe Beispiel 2.18.

Formalisieren wir den Ansatz, so liegt der Verwerfungsmethode die folgende Idee zugrunde: Man erzeugt zunächst eine Realisation x einer Zufallsvara-

blen Y mit der „einfachen“ Dichte $g(x)$, z.B. der Gleichverteilung im Falle der Beta-Verteilung, und entscheidet anschließend, x auch als Realisation von X mit der eigentlichen Dichte f zu akzeptieren oder nicht zu akzeptieren und das Verfahren mit einer neuen Realisation von Y zu wiederholen. Die Annahmeentscheidung hängt (zumindest tendenziell) davon ab, wie stark sich beide Dichten an der Stelle x unterscheiden. Hierzu bestimmt man eine Konstante $c \geq 1$ mit $f(x') \leq cg(x')$ für alle $x' \in \mathbb{R}$. Die Funktion $cg(x')$ wird damit zu einer oberen Schranke für die Funktion $f(x')$. Je kleiner die Konstante c gewählt werden kann, umso stärker rückt die obere Schranke $cg(x')$ an die Funktion $f(x')$ heran. Im Idealfall berührt sie diese in mindestens einem Punkt.

Die endgültige Entscheidung über die Annahme/Ablehnung von x als Realisation von X hängt dann noch vom Ausgang eines Zufallsexperiments ab: Man erzeugt eine Zufallszahl z und akzeptiert x als Realisation von X , falls $z \leq f(x)/cg(x)$ gilt.

Zusammenfassend ergeben sich die folgenden Schritte der **Verwerfungsmethode**:

1. Erzeuge x als Realisation von Y mit der Dichte g .
2. Erzeuge Zufallszahl z (unabhängig von Y).
3. Ist $z \leq \frac{f(x)}{c \cdot g(x)}$, so akzeptiere x als Realisation von X mit der Dichte f und stoppe. Ansonsten fahre mit Schritt 1 fort.

Die theoretischen Eigenschaften der Verwerfungsmethode sind Gegenstand von Satz 2.17. Zunächst wird überprüft, dass das Verfahren tatsächlich eine Realisation bzgl. f liefert. Weiter wird gezeigt, dass die Anzahl der Iterationen bis zur Annahme einer Realisation von Y geometrisch verteilt ist mit Erwartungswert c . Die Konstante c gibt somit die mittlere Anzahl von Iterationen an, die nötig ist, um eine Realisation bzgl. f zu erhalten und ist schließlich mitentscheidend für die Effizienz des Verfahrens.

Satz
2.17

- (i) Die Zufallsvariable X , die mit Hilfe der Verwerfungsmethode erzeugt wird, hat die Dichte f .
- (ii) Die Anzahl N der Iterationen, die nötig sind, um eine Realisation von X zu erzeugen, ist geometrisch verteilt mit Erwartungswert $E(N) = c$.

Beweis: Sei A das Ereignis, dass eine Realisation von Y als Realisation von X akzeptiert wird. Dann gilt

$$P(A) = \int_{-\infty}^{+\infty} P\left(Z \leq \frac{f(x)}{c \cdot g(x)}\right) g(x) dx = \int_{-\infty}^{+\infty} \frac{f(x)}{c \cdot g(x)} g(x) dx = 1/c.$$

$P(A)$ können wir interpretieren als Erfolgswahrscheinlichkeit bei unabhängigen Wiederholungen. Damit ist N , die Anzahl der Iterationen bis zur Akzeptanz, $Geo_{\mathbb{N}}(1/c)$ -verteilt und wir erhalten für den zugehörigen Erwartungswert

$$E(N) = \sum_{n=0}^{\infty} P(N > n) = \sum_{n=0}^{\infty} (1 - 1/c)^n = c.$$

Wir müssen noch zeigen, dass die so erzeugte Zufallsvariable X die Dichte f hat.

$$\begin{aligned} P(X \leq x) &= P(Y \leq x \mid A) \\ &= \frac{P(\{Y \leq x\} \cap A)}{P(A)} \\ &= c \int_{-\infty}^{+\infty} P(\{Y \leq x\} \cap A \mid Y = y) g(y) dy \\ &= c \int_{-\infty}^x P(A \mid Y = y) g(y) dy \\ &= c \int_{-\infty}^x P\left(Z \leq \frac{f(y)}{c \cdot g(y)}\right) g(y) dy \\ &= c \int_{-\infty}^x \frac{f(y)}{c} dy \\ &= \int_{-\infty}^x f(y) dy. \end{aligned}$$

Damit ist der Satz bewiesen. \square

2.18

Beispiel

Sei X eine $Beta(2, 4)$ -verteilte Zufallsvariable. Die zugehörige Dichte

$$f(x) = 20x(1-x)^3, \quad 0 < x < 1,$$

nimmt positive Werte lediglich im Intervall $(0, 1)$ an. Daher wählen wir als „einfache“ Dichte eine Gleichverteilung auf $(0, 1)$, also $g(x) = 1$ für $x \in (0, 1)$. Auf dem Intervall $(0, 1)$ ist $h(x) := f(x)/g(x)$ wohldefiniert und wir erhalten durch Nullsetzen der ersten Ableitung von h deren Maximum. Insbesondere ist $h(x) \leq h(1/4) = 135/64$ und wir können $c = 135/64$ wählen. Zusammenfassend ergibt sich:

1. Erzeuge Zufallszahlen z_1 und z_2 .
2. Ist $z_2 \leq \frac{256}{27} z_1(1 - z_1)^3$, setze $x = z_1$. Andernfalls fahre mit Schritt 1 fort.

Im Mittel wird etwa jede zweite Zufallszahl z_1 als Realisation von X akzeptiert.

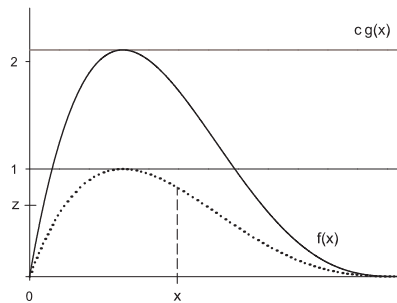


Abb. 2.7. Verwerfungsmethode angewandt auf die $Beta(2, 4)$ -Verteilung

Abb. 2.7 veranschaulicht das Verfahren. Die punktierte Kurve stellt die obere Schranke $f(x)/cg(x)$ für z dar, die zur Annahme von x als Realisation von X führt. In der konkreten Situation führt dies zur Ablehnung von x als Realisation von X und damit zur Fortsetzung des Verfahrens. \diamond

Beispiel

2.19

Der Absolutwert $|X|$ einer $N(0, 1)$ -verteilten Zufallsvariablen X hat die Dichte

$$f(x) = \sqrt{\frac{2}{\pi}} e^{-x^2/2}, \quad x \geq 0.$$

Wählen wir als „einfache“ Dichte eine Exponentialverteilung mit Parameter $\lambda = 1$, also $g(x) = e^{-x}$ für $x \geq 0$, so ist $h(x) := f(x)/g(x) = \sqrt{\frac{2}{\pi}} e^{x-x^2/2}$, $x \geq 0$, wohldefiniert und nimmt das Maximum an der Stelle $x = 1$ an. Daher können wir $c = h(1) = \sqrt{\frac{2e}{\pi}}$ wählen. Zusammenfassend ergibt sich:

1. Erzeuge Zufallszahlen z_1 und z_2 .
2. Setze $x = -\ln z_1$ (als $Expo(1)$ -verteilte Zufallszahl)
3. Ist $z_2 \leq \frac{1}{\sqrt{e}} e^{x-x^2/2}$, akzeptiere x als Realisation von $|X|$. Andernfalls fahre mit Schritt 1 fort.

Die Akzeptanzwahrscheinlichkeit ist $1/c \approx 0.76$.

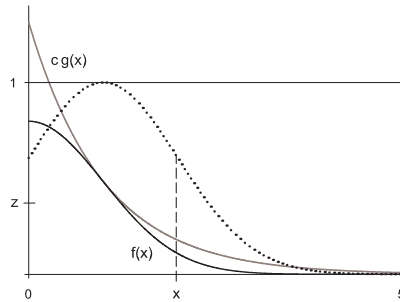


Abb. 2.8. Verwerfungsmethode angewandt auf $|X|$ der $N(0, 1)$ -Verteilung

Abb. 2.8 veranschaulicht die Vorgehensweise. Wie schon in Abb. 2.7 stellt die punktierte Kurve die Obergrenze von z für die Annahme von x als Realisation von X dar.

Die Erzeugung einer Realisation von $|X|$ kann auch als Zwischenschritt zur Erzeugung einer Realisation von X herangezogen werden. Hierzu hat man lediglich noch das Vorzeichen mit Hilfe einer Zufallszahl festzulegen.

1. Erzeuge eine Realisation x von $|X|$.
2. Erzeuge eine Zufallszahl z .
3. Ist $z \leq 0.5$, lege x als Realisation von X fest. Andernfalls (d.h. $z > 0.5$), lege $-x$ als Realisation von X fest.

Hierbei handelt es sich natürlich nicht mehr um eine reine Anwendung der Verwerfungsmethode. Andererseits gibt die Vorgehensweise einen ersten Eindruck, wie trickreich der Zusammenhang bestimmter Verteilungen ausgenutzt werden kann. \diamond

Die betrachteten Beispiele lassen bereits die Bedeutung der Verwerfungsmethode als Alternative zur Inversionsmethode erkennen. Die konkrete Auswahl einer „einfachen“ Dichte g und/oder Festlegung der Konstanten c ist jedoch nicht immer so einfach. Daher ist es nicht überraschend, dass für bestimmte Verteilungen, z.B. die Gamma-Verteilung, in der Literatur unterschiedliche Umsetzungen bis hin zu Modifikationen der Verwerfungsmethode vorgeschlagen werden.

Beispiel (Gamma-Verteilung)
2.20

Sei X $Gamma(\alpha, \beta)$ -verteilt. Wir schließen zunächst den Fall $\alpha \in \mathbb{N}$ (Spezialfall Erlang-Verteilung, siehe Beispiel 2.22) aus und betrachten die Fälle $\alpha < 1$ sowie $\alpha > 1$ separat. Hierzu reicht es aus, sich auf $\beta = 1$ zu konzentrieren, da wir eine Realisation x' einer $Gamma(\alpha, \beta)$ -verteilten Zufallsvariablen X' durch Transformation $x' = \beta x$ einer $Gamma(\alpha, 1)$ -verteilten Zufallsvariablen X erhalten.

Für $\alpha < 1$ erhält man das folgende, auf Ahrens, Dieter (1974) zurückgehende Verfahren zur Erzeugung einer Realisation x einer $Gamma(\alpha, 1)$ -verteilten Zufallsvariablen X :

1. Setze $b = (e + \alpha)/e$.
2. Erzeuge Zufallszahl z und setze $p = bz$. Ist $p > 1$, fahre mit 4. fort; andernfalls mit 3..
3. Setze $y = p^{1/\alpha}$ und erzeuge Zufallszahl z' . Ist $z' \leq e^{-y}$, setze $x = y$ und stoppe. Andernfalls fahre mit 2. fort.
4. Setze $y = -\ln[(b - p)/\alpha]$ und erzeuge Zufallszahl z' . Ist $z' \leq y^{\alpha-1}$, setze $x = y$ und stoppe. Andernfalls fahre mit 2. fort.

Ist $\alpha > 1$ so kann man das folgende, auf Cheng (1977) zurückgehende Verfahren zur Erzeugung einer $Gamma(\alpha, 1)$ -verteilten Zufallszahl x heranziehen:

1. Setze $a = 1/\sqrt{2\alpha - 1}$, $b = \alpha - \ln 4$, $q = \alpha + 1/a$, $\theta = 4.5$, $d = 1 + \ln \theta$.
2. Erzeuge Zufallszahlen z und z' .
3. Setze $v = a \ln[z/(1 - z)]$, $y = \alpha e^v$, $u = z^2 z'$, $w = b + qv - y$.
4. Ist $w + d - \theta u \geq 0$, setze $x = y$ und stoppe. Andernfalls fahre mit 5. fort.
5. Ist $w \geq \ln u$, setze $x = y$ und stoppe. Andernfalls fahre mit 2. fort.

Marsaglia, Tsang (2000) schlagen das folgende, ebenfalls auf der Verwerfungsmethode basierende Verfahren zur Erzeugung einer Realisation x einer $Gamma(\alpha, 1)$ -verteilten Zufallsvariablen X für $\alpha > 1$ vor:

1. Setze $a = \alpha - 1/3$, $b = 1/\sqrt{9a}$.
2. Erzeuge Zufallszahl z .
3. Erzeuge Realisation y bzgl. der Standard-Normalverteilung.
4. Ist $y > -1/b$ und gilt

$$\ln z < (1 - 3\alpha) \ln(1 + by) - a(1 + by)^3 + a + y^2/2,$$

setze $x = a(1 + by)^3$ und stoppe. Andernfalls fahre mit Schritt 2 fort.

Für $\alpha < 1$ bietet sich eine Modifikation des Verfahrens für $\alpha > 1$ an, das auf dem folgenden Zusammenhang basiert: Ist X eine $Gamma(1 + \alpha, 1)$ -verteilte Zufallsvariable und Z eine von X unabhängige, auf $(0, 1)$ gleichverteilte Zufallsvariable, so ist $XZ^{1/\alpha}$ $Gamma(\alpha, 1)$ -verteilt. \diamond

Neben der Inversions- und Verwerfungsmethode bieten sich eine Reihe spezieller, mehr oder weniger trickreich den Zusammenhang bestimmter Verteilungen ausnutzende Verfahren an.

2.4 Die Faltungsmethode

Werfen Sie einen fairen Würfel, so erhalten Sie die Augenzahl 3 mit Wahrscheinlichkeit $1/6$. Werfen Sie den Würfel zehnmal, so können Sie zählen, wie häufig die Augenzahl 3 auftritt. Aus der Statistik wissen wir, dass die Häufigkeit, mit der die Augenzahl 3 auftritt, binomialverteilt ist mit den Parametern

$n = 10$ und $p = 1/6$. Sie haben damit zwei Möglichkeiten, die Häufigkeit der Augenzahl 3 zu realisieren: (a) Sie erzeugen eine Realisation der Binomialverteilung wie in Beispiel 2.14 beschrieben oder (b) Sie simulieren das Werfen mit dem Würfel, indem Sie zehn Zufallszahlen z_1, \dots, z_{10} erzeugen, $z_j \leq p$ als Erfolg (Augenzahl 3) interpretieren, und die Erfolge zählen.

Das Experiment mit dem Würfel lässt folgende Verallgemeinerung zu: Ein Ereignis trete mit Wahrscheinlichkeit $p \in (0, 1)$ ein. Wir wissen, dass die Binomialverteilung „zählt“, wie häufig das Ereignis bei n unabhängigen Beobachtungen eintritt. Diese Struktur können wir bei der Simulation ausnutzen, indem wir anstelle der Zähldichte der Binomialverteilung (vgl. (2.8)) das der Definition zugrunde liegende Experiment simulieren. Beschreiben wir Erfolg/Misserfolg durch eine $Ber(p)$ -verteilte Zufallsvariable Y (mit den Werten $Y = 1$ für Erfolg und $Y = 0$ für Misserfolg sowie der Erfolgswahrscheinlichkeit p), so können wir X , die Anzahl der Erfolge, als Summe $X = Y_1 + \dots + Y_n$ von n unabhängigen, $Ber(p)$ -verteilten Zufallsvariablen Y_1, \dots, Y_n darstellen.

Beispiel (Binomialverteilung; Faltungsmethode)

2.21

Stellt man eine $Bin(n, p)$ -verteilte Zufallsvariable X als Summe

$$X = Y_1 + \dots + Y_n$$

von n unabhängigen, $Ber(p)$ -verteilten Zufallsvariablen Y_1, \dots, Y_n dar (siehe Beispiel A.5(a)), so erhält man die folgende Alternative zur Inversionsmethode:

1. Erzeuge Zufallszahlen z_1, \dots, z_n .
2. Für $j = 1, \dots, n$ setze $y_j = 1$, falls $z_j \leq p$ und $y_j = 0$ sonst.
3. Setze $x = y_1 + \dots + y_n$. \diamond

Die Vorgehensweise ist nicht beschränkt auf die Binomialverteilung als Summe von Bernoulli-Verteilungen.

Lässt sich eine (beliebige) Zufallsvariable X als Summe $X = Y_1 + \dots + Y_n$ von n unabhängigen Zufallsvariablen Y_1, \dots, Y_n darstellen, so kann man anstelle einer Realisation von X Realisationen von Y_1, \dots, Y_n erzeugen und diese aufsummieren. In diesem Fall sprechen wir von der **Faltungsmethode**.

2.22 Beispiel (Erlang-Verteilung)

Die Verteilungsfunktion einer $Erlang(n, \alpha)$ -verteilten Zufallsvariable X

$$F(x) = 1 - \sum_{k=0}^{n-1} \frac{(\alpha x)^k}{k!} e^{-\alpha x}, \quad x \geq 0,$$

liegt nicht in expliziter Form vor. Daher scheidet die Inversionsmethode aus. Andererseits kann man einen Zusammenhang zwischen der Erlang-Verteilung und der Exponentialverteilung ausnutzen (vgl. Beispiel A.5(d)): Eine $Erlang(n, \alpha)$ -verteilte Zufallsvariable X ist die Summe $X = Y_1 + \dots + Y_n$ von n unabhängigen, $Expo(\alpha)$ -verteilten Zufallsvariablen Y_1, \dots, Y_n . Darüber hinaus wissen wir aus Beispiel 2.7, dass die Inversionsmethode zu gegebener Zufallszahl z mit $y = -\frac{1}{\alpha} \ln z$ eine Realisation einer Exponentialverteilung liefert. Hieraus ergibt sich folgendes Verfahren zur Erzeugung einer Realisation von X :

1. Erzeuge n Zufallszahlen z_1, \dots, z_n .
2. Setze $x = -\frac{1}{\alpha} \ln(z_1 z_2 \dots z_n)$.

(unter Berücksichtigung von $\ln z_1 + \dots + \ln z_n = \ln(z_1 z_2 \dots z_n)$). \diamond

2.23 Beispiel (Negative Binomialverteilung; Faltungsmethode)

Eine $NegBin(n, p)$ -verteilte Zufallsvariable X mit $n \in \mathbb{N}$ ist nach Beispiel A.5(b) darstellbar als Summe $X = Y_1 + \dots + Y_n$ von n unabhängigen, $Geo_{\mathbb{N}_0}(p)$ -verteilten Zufallsvariablen. Zusammen mit Beispiel 2.12 liefert dann die Faltungsmethode:

1. Erzeuge n Zufallszahlen z_1, \dots, z_n .
2. Setze $x = \sum_{i=1}^n \left\lfloor \frac{\ln z_i}{\ln(1-p)} \right\rfloor$. \diamond

Bisher haben wir unterstellt, dass X *exakt* in der Form $X = Y_1 + \dots + Y_n$ darstellbar ist. Auf der Grundlage des zentralen Grenzwertsatzes (Satz A.11) ergibt sich eine weitere interessante Anwendung.

Seien Y_1, \dots, Y_n unabhängige, identisch verteilte Zufallsvariable mit Erwartungswert μ und Varianz σ^2 . Nach dem zentralen Grenzwertsatz ist

$$X = \frac{Y_1 + \dots + Y_n - n\mu}{\sigma\sqrt{n}}$$

näherungsweise standardnormalverteilt.

Beispiel (Normalverteilung; Faltungsmethode)

2.24

Sind Z_1, \dots, Z_n unabhängige, $U[0, 1]$ -verteilte Zufallsvariable, so gilt $E(Z) = 1/2$ und $Var(Z) = 1/12$. Unter Berücksichtigung des zentralen Grenzwertsatzes ist dann

$$X = \frac{Z_1 + \dots + Z_n - 0.5n}{\sqrt{n/12}}$$

näherungsweise $N(0, 1)$ -verteilt. Speziell für $n = 12$ lassen sich beide Dichten optisch nicht mehr unterscheiden und man erhält das folgende einfache Verfahren zur Erzeugung einer Realisation einer $N(0, 1)$ -verteilten Zufallsvariablen X :

1. Erzeuge zwölf Zufallszahlen z_1, \dots, z_{12} .
2. Setze $x = z_1 + \dots + z_{12} - 6$.

Der einfachen Realisation steht ein hoher „Verbrauch“ an Zufallszahlen gegenüber, der es erforderlich macht, bei komplexen Problemen über Alternativen (Box-Müller Methode, Polarmethode; siehe Beispiele 2.30, 2.31) nachzudenken. \diamond

Machen wir Gebrauch von der Möglichkeit, die Binomialverteilung durch eine Normalverteilung zu approximieren, so erhalten wir die folgende Alternative zur Erzeugung $Bin(n, p)$ -verteilter Zufallszahlen.

Beispiel (Binomialverteilung; Approximation durch Normalverteilung)

2.25

Sei X eine $Bin(n, p)$ -verteilte Zufallsvariable mit $np \geq 5$. Nach dem zentralen Grenzwertsatz ist X in guter Näherung normalverteilt mit Erwartungswert $\mu = np$ und Varianz $\sigma^2 = np(1 - p)$ und wir erhalten unter Berücksichtigung der (als Stetigkeitskorrektur bekannten) Verschiebung des Erwartungswertes um $1/2$:

1. Erzeuge $N(0, 1)$ -verteilte Zufallszahl y .
2. Setze $x = \left\lceil np - 0.5 + \sqrt{np(1-p)} y \right\rceil$.

Weniger bekannt ist die arcsin-Transformation, die besagt, dass $\arcsin \sqrt{\frac{X}{n}}$ näherungsweise $N(\arcsin \sqrt{p}, \frac{1}{4n})$ -verteilt ist. Hieraus folgt:

1. Erzeuge $N(0, 1)$ -verteilte Zufallszahl y .
2. Setze $x = \left\lceil n \sin^2 \left(\arcsin \sqrt{p} + \frac{1}{2\sqrt{n}} y \right) \right\rceil$. \diamond

Die Vorgehensweise lässt sich in natürlicher Weise auf jede auf dem zentralen Grenzwertsatz basierende Approximation einer Verteilung übertragen.

2.5 Die Alias-Methode

Sie haben drei faire Würfel; einen mit den Zahlen 1 und 2, einen mit den Zahlen 1 und 3 und einen mit den Zahlen 1 und 4. Sie wählen einen der drei Würfel zufällig aus und werfen ihn. Dann erhalten Sie die Zahl 1 mit Wahrscheinlichkeit $1/2$ und jede der übrigen Zahlen mit Wahrscheinlichkeit $1/6$.

Stellen Sie sich umgekehrt vor, sie wollen eine Realisation einer Zufallsvariablen X mit der Zähldichte $P(X = 1) = 1/2$ und $P(X = i) = 1/6$ für $i = 2, 3, 4$ erzeugen. Dann können Sie die Inversionsmethode anwenden. Sie können aber auch das beschriebene Würfelexperiment heranziehen: Hierzu wählen Sie einen der drei Würfel zufällig aus (diskrete Gleichverteilung auf $\{1, 2, 3\}$), werfen ihn und interpretieren die erhaltene Zahl (nur zwei Möglichkeiten) als Realisation von X . Beide Schritte lassen sich einfach und schnell realisieren.

Sei daher X eine diskrete Zufallsvariable, die die Werte x_1, \dots, x_m annehmen kann mit den Wahrscheinlichkeiten $P(X = x) = p_m(x)$ für $x \in \{x_1, \dots, x_m\}$.

Die Idee der Alias-Methode besteht nun darin, die Zähldichte p_m als arithmetisches Mittel

$$p_m(x) = \frac{1}{m-1} \sum_{j=1}^{m-1} q_j(x), \quad x \in \{x_1, \dots, x_m\}, \quad (2.9)$$

von $m - 1$ Zähl-dichten q_1, \dots, q_{m-1} darzustellen, wobei jede Zähl-dichte q_j höchstens zwei von Null verschiedene Wahrscheinlichkeiten α_j und $\beta_j = 1 - \alpha_j$ hat. Seien y_j und \bar{y}_j die zugehörigen x -Werte.

Weniger offensichtlich ist die folgende Zerlegung

$$\begin{pmatrix} 1/10 \\ 2/10 \\ 4/10 \\ 3/10 \end{pmatrix} = \frac{1}{3} \begin{pmatrix} 3/10 \\ 0 \\ 7/10 \\ 0 \end{pmatrix} + \frac{1}{3} \begin{pmatrix} 0 \\ 6/10 \\ 4/10 \\ 0 \end{pmatrix} + \frac{1}{3} \begin{pmatrix} 0 \\ 0 \\ 1/10 \\ 9/10 \end{pmatrix}, \quad (2.10)$$

die wir zur Veranschaulichung der Rechenschritte heranziehen wollen.

Gelingt es in einer Initialisierungsphase eine solche Zerlegung effizient vorzunehmen, so ergeben sich die Realisationen von X auf einfache und schnelle Weise: Man wählt q_j zufällig aus und erzeugt dann eine Realisation bzgl. $q_j(x)$ unter Berücksichtigung, dass höchstens zwei der m Wahrscheinlichkeiten von $q_j(x)$ von Null verschieden sind. Hierzu sind zwei Zufallszahlen z_1 und z_2 nötig; die erste, um $j = \lceil (m - 1)z_1 \rceil$ und damit q_j auszuwählen und die zweite, um $x = y_j$ im Falle $z_2 \leq \alpha_j$ und $x = \bar{y}_j$ im Falle $z_2 > \alpha_j$ festzulegen. Angewandt auf Zerlegung (2.10) erhält man z.B. bei Vorliegen der Zufallszahlen $z_1 = 0.2$ und $z_2 = 0.6$ zunächst $j = \lceil (m - 1)z_1 \rceil = \lceil 0.6 \rceil = 1$ und dann mit $\alpha_1 = 3/10$ die Realisation $x = 3$, da $z_2 = 0.6 > 0.3 = \alpha_1$.

Zusammenfassend ergeben sich die folgenden Rechenschritte:

1. Erzeuge Zufallszahlen z_1 und z_2 .
2. Setze $j = \lceil (m - 1)z_1 \rceil$.
3. Ist $z_2 \leq \alpha_j$, setze $x = y_j$. Andernfalls setze $x = \bar{y}_j$.

Wir kommen nun zur Durchführung der Zerlegung der Zähl-dichte $p_m(x)$. Hierzu wählen wir ein $x_i \in \{x_1, \dots, x_m\}$ aus mit der Eigenschaft $p_m(x_i) \leq 1/(m - 1)$. Dies ist immer möglich; denn wäre es nicht möglich, so wäre $\sum_{\nu=1}^m p_m(x_\nu) > m/(m - 1) > 1$ im Widerspruch zu den Eigenschaften einer Zähl-dichte. Ferner wählen wir ein $x_k \in \{x_1, \dots, x_m\}$ aus mit $p_m(x_i) + p_m(x_k) \geq 1/(m - 1)$. Auch dies ist immer möglich, wie sich der Leser leicht selbst überlegt. Mit Hilfe von x_i und x_k legen wir dann q_1 fest,

$$\begin{aligned} q_1(x_i) &= (m - 1)p_m(x_i) = \alpha_1 \\ q_1(x_k) &= 1 - q_1(x_i) = 1 - \alpha_1 = \beta_1 \\ q_1(x_\nu) &= 0 \quad \text{für } \nu \neq i, k \end{aligned}$$

und erhalten zunächst

$$p_m(x_\nu) = \frac{1}{m-1} q_1(x_\nu) + \frac{m-2}{m-1} p_{m-1}(x_\nu), \quad (2.11)$$

wobei

$$\begin{aligned} p_{m-1}(x_i) &= 0 \\ p_{m-1}(x_k) &= \frac{m-1}{m-2} \left(p_m(x_k) - \frac{1}{m-1} q_1(x_k) \right) \\ &= \frac{m-1}{m-2} \left(p_m(x_k) + p_m(x_i) - \frac{1}{m-1} \right) \\ p_{m-1}(x_\nu) &= \frac{m-1}{m-2} p_m(x_\nu) \quad \text{für } \nu \neq i, k. \end{aligned}$$

Vereinfacht kann man auch $p_{m-1}(x_i)$ gleich Null setzen, $p_{m-1}(x_\nu)$ für $\nu \neq i, k$ gemäß $p_{m-1}(x_\nu) = \frac{m-1}{m-2} p_m(x_\nu)$ berechnen und $p_{m-1}(x_k)$ unter Ausnutzung der Normierungsbedingung $p_{m-1}(x_k) = 1 - \sum_{\nu \neq i, k} p_{m-1}(x_\nu)$ festlegen.

Fasst man p_{m-1} als Zähldichte bzgl. $\{x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_m\}$ auf, so kann man die Vorgehensweise mit p_{m-1} anstelle von p_m für die verbleibenden $m-1$ Komponenten (also ohne x_i) wiederholen und erhält

$$p_{m-1}(x_\nu) = \frac{1}{m-2} q_2(x_\nu) + \frac{m-3}{m-2} p_{m-2}(x_\nu)$$

und eingesetzt in (2.11)

$$p_m(x_\nu) = \frac{1}{m-1} q_1(x_\nu) + \frac{1}{m-1} q_2(x_\nu) + \frac{m-3}{m-1} p_{m-2}(x_\nu).$$

Fährt man auf diese Weise fort, so erhält man schließlich die angestrebte Zerlegung (2.9).

Tabelle 2.6 enthält die Rechenschritte, die der Zerlegung (2.10) zugrunde liegen. Den Indizes i und k kann man die gewählten Variablen x_i und x_k auf den einzelnen Stufen entnehmen. So wurde beispielsweise $i = 1$ und $k = 3$ im Falle p_4 zur Festlegung von q_1 gewählt.

p_4	q_1	p_3	q_2	p_2	q_3
$1/10_i$	$3/10$	0	0	0	0
$2/10$	0	$3/10_i$	$6/10$	0	0
$4/10_k$	$7/10$	$5/20_k$	$4/10$	$1/10_i$	$1/10$
$3/10$	0	$9/20$	0	$9/10_k$	$9/10$

Tabelle 2.6. Zerlegung der Zähldichte (2.10)

Zusammenfassend können wir festhalten: Der erhöhte Rechenaufwand in der Initialisierungsphase (Zerlegung der Zähldichte) zahlt sich in der Regel sehr schnell aus, da man gewöhnlich nicht nur eine Realisation, sondern eine Vielzahl von Realisationen zu erzeugen hat.

Liegt eine Zufallsvariable X mit Werten in einer abzählbaren Menge I vor, so kann man von der folgenden Verallgemeinerung der Alias-Methode Gebrauch machen: Man wählt eine endliche Menge $M \subset I$ mit $c_M := P(X \in M)$ „nahe“ 1 im Hinblick auf eine Anwendung der Alias-Methode aus. Hierzu seien Y_1 eine Zufallsvariable mit Werten in M und Zähldichte $P(Y_1 = x) = c_M^{-1}P(X = x)$, $x \in M$, und Y_2 eine weitere Zufallsvariable mit Werten in $I \setminus M$ und Zähldichte $P(Y_2 = x) = (1 - c_M)^{-1}P(X = x)$, $x \notin M$. Dann lässt sich die Zähldichte $P(X = x)$, $x \in I$, mit Hilfe der Formel von der totalen Wahrscheinlichkeit als Mischung

$$\begin{aligned} P(X = x) &= c_M P(X = x \mid X \in M) + (1 - c_M) P(X = x \mid X \notin M) \\ &= c_M \frac{P(X = x, X \in M)}{P(X \in M)} + (1 - c_M) \frac{P(X = x, X \notin M)}{P(X \notin M)} \\ &= c_M P(Y_1 = x) + (1 - c_M) P(Y_2 = x) \end{aligned}$$

der Zähldichten von Y_1 und Y_2 darstellen. Hieraus ergibt sich unmittelbar die verallgemeinerte Vorgehensweise:

1. Erzeuge Zufallszahl z .
2. Ist $z \leq c_M$, erzeuge Realisation y von Y_1 (mit Hilfe der Alias-Methode). Andernfalls erzeuge Realisation y von Y_2 .
3. Setze $x = y$.

Die Kompositionsmethode

An einem Urlaubsort scheint an 80 von 100 Tagen die Sonne und an 20 von 100 Tagen regnet es. Bei Sonnenschein ist die (Höchst-)Temperatur gleichverteilt zwischen 20 und 30 Grad; bei Regen ist sie gleichverteilt zwischen 15 und 25 Grad.

Wollen wir die Temperatur an diesem Urlaubsort simulieren, so bietet sich folgende zweistufige Vorgehensweise an: Mit Hilfe einer Zufallszahl wählen wir zunächst einen Sonnen- oder Regentag aus. Ergibt die Zufallszahl einen Sonnentag (Regentag), so erzeugen wir eine Realisation bzgl. eines Sonnentages

(Regentages) und interpretieren das Ergebnis als Realisation der Temperatur X in dem Urlaubsort. Diese Vorgehensweise ist möglich, da wir $F(x)$ als Mischung zweier Verteilungen $F_{So}(x)$ und $F_{Re}(x)$ darstellen können;

$$\begin{aligned} F(x) &= P(X \leq x) \\ &= P(\text{„Sonne“})P(X \leq x | \text{„Sonne“}) + P(\text{„Regen“})P(X \leq x | \text{„Regen“}) \\ &= 0.8 F_{So}(x) + 0.2 F_{Re}(x). \end{aligned}$$

Bei der Temperatur unseres Urlaubsortes ergibt sich diese Mischung in natürlicher Weise. Bei der Alias-Methode haben wir mit der Darstellung der Verteilung als arithmetisches Mittel von Zweipunktverteilungen eine solche Mischung algorithmisch erzeugt. Auch die Alias-Methode ist nur ein Beispiel für eine rechentechnisch motivierte Zerlegung.

Betrachtet man z.B. eine Zufallsvariable mit der Verteilungsfunktion

$$F(x) = \frac{1}{6}x + \frac{1}{2}x^2 + \frac{1}{3}x^3, \quad 0 \leq x \leq 1,$$

so würde die Inversionsmethode zu gegebener Zufallszahl z die Lösung der nichtlinearen Gleichung $x + 3x^2 + 2x^3 = 6z$ erfordern. Wesentlich bequemer ist die Darstellung von $F(x)$ als Mischung

$$F(x) = \frac{1}{6}F_1(x) + \frac{1}{2}F_2(x) + \frac{1}{3}F_3(x), \quad 0 \leq x \leq 1,$$

mit $F_i(x) = x^i$ für $i = 1, 2, 3$ und $0 \leq x \leq 1$. Angewandt auf $F_i(x)$ liefert die Inversionsmethode $x = z^{1/i}$. Damit erhält man die angestrebte Realisation von $F(x)$, indem man zunächst eine Zufallszahl z_1 erzeugt, mit deren Hilfe man i festlegt, um dann mit Hilfe einer zweiten Zufallszahl z_2 die Realisation x gemäß $x = z_2^{1/i}$ zu bestimmen.

Eine Verteilungsfunktion F bezeichnet man als (endliche) **Mischung** der Verteilungsfunktionen F_1, \dots, F_m , wenn zu vorgegebenen Konstanten $a_1, \dots, a_m > 0$ mit $a_1 + \dots + a_m = 1$ gilt:

$$F(x) = a_1 F_1(x) + \dots + a_m F_m(x), \quad x \in \mathbb{R}. \quad (2.12)$$

Seien daher X und X_i ($i = 1, \dots, m$) die zugehörigen Zufallsvariablen und Y eine davon unabhängige Zufallsvariable mit $P(Y = i) = a_i$, $1 \leq i \leq m$. Dann erhält man mit Hilfe des Satzes von der totalen Wahrscheinlichkeit

$$\begin{aligned}
P\left(\sum_{k=1}^m X_k 1_{\{Y=k\}} \leq x\right) &= \sum_{i=1}^m a_i P\left(\sum_{k=1}^m X_k 1_{\{Y=k\}} \leq x \mid Y=i\right) \\
&= \sum_{i=1}^m a_i P(X_i \leq x) \\
&= \sum_{i=1}^m a_i F_i(x).
\end{aligned}$$

Somit stimmen die Verteilungen von X und $\sum_{k=1}^m X_k 1_{\{Y=k\}}$ überein und wir können anstelle einer Realisation von X eine Realisation von $\sum_{k=1}^m X_k 1_{\{Y=k\}}$ mit dem folgenden zweistufigen Verfahren bestimmen (**Kompositionsmethode**).

1. Erzeuge Realisation k von Y .
2. Erzeuge Realisation x_k von X_k und setze $x = x_k$.

Die Vorgehensweise lässt sich in natürlicher Weise auf Mischungen übertragen, die in Form von Dichten oder Zähldichten vorliegen. Hierzu hat man in (2.12) lediglich die Verteilungsfunktionen durch die entsprechenden Dichten/Zähldichten zu ersetzen.

Bei einer Klausur kommt es gelegentlich vor, dass neben gut vorbereiteten Studierenden auch schlecht oder nicht vorbereitete Studierende mitschreiben. Dies führt zu einer *zweigipfligen* Zähldichte des Klausurergebnisses, die sich häufig durch eine Mischung von zwei symmetrischen Zähldichten beschreiben lässt.

Zur Veranschaulichung der Vorgehensweise bei der Kompositionsmethode betrachten wir das folgende Beispiel.

Beispiel (Mischung zweier Erlang-Verteilungen)

2.26

Abb. 2.9 zeigt die Dichte

$$f(x) = a_1 \frac{\alpha^{n_1} x^{n_1-1}}{(n_1-1)!} e^{-\alpha_1 x} + a_2 \frac{\alpha^{n_2} x^{n_2-1}}{(n_2-1)!} e^{-\alpha_2 x}, \quad x \geq 0,$$

der Mischung zweier $Erlang(n_i, \alpha_i)$ -Verteilungen mit $a_1 = a_2 = 1/2$, $n_1 = 4$, $n_2 = 15$ und $\alpha_1 = \alpha_2 = 1/7$.

Mit Hilfe von Beispiel 2.22 können wir unmittelbar eine Realisation der Erlang-Verteilung angeben. Wir müssen also lediglich eine der beiden Erlang-

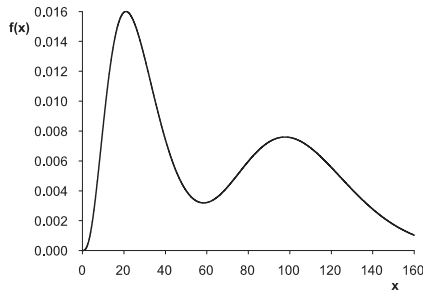


Abb. 2.9. Mischung zweier Erlang-Verteilungen

Verteilungen zufällig auswählen und dann bzgl. dieser ausgewählten Erlang-Verteilung eine Realisation erzeugen. Dies führt auf die folgenden Rechenschritte:

1. Erzeuge eine Zufallszahl z .
2. Setze $k = \lceil 2z \rceil$, $\ell = n_k$.
3. Erzeuge ℓ Zufallszahlen z_1, \dots, z_ℓ .
4. Setze $x = -\frac{1}{\alpha} \ln(z_1 z_2 \dots z_\ell)$. \diamond

Eine Mischung ist nicht auf endlich (oder abzählbar unendlich) viele Verteilungsfunktionen beschränkt. So ist z.B. die Verteilungsfunktion

$$F(x) = \int_0^\infty F_\lambda(x) g(\lambda) d\lambda, \quad x \in \mathbb{R}, \quad (2.13)$$

eine Mischung von überabzählbar vielen Verteilungsfunktionen F_λ , wobei die **Mischungsfunktion** $g(\lambda)$, die auch als **Strukturfunktion** bezeichnet wird, die Eigenschaften einer Dichte besitzt ($g(\lambda) \geq 0$ für alle $\lambda > 0$ und $\int_0^\infty g(\lambda) d\lambda = 1$).

In wichtigen Spezialfällen führt die gemischte Verteilung wieder auf eine Standardverteilung und ergibt so neben einem zusätzlichen Einblick in den Zusammenhang von Verteilungen eine weitere Möglichkeit Realisationen zu erzeugen.

Beispiel

2.27

Sei F_λ die Verteilungsfunktion einer Exponentialverteilung mit Parameter $\lambda > 0$. Als Strukturfunktion $g(\lambda)$ wählen wir eine Gamma-Verteilung mit den Parametern $\alpha > 0$ und $\beta > 0$. Dann folgt aus Beispiel A.7(a) für $x \geq 0$

$$F(x) = \int_0^\infty F_\lambda(x)g(\lambda)d\lambda = \int_0^\infty (1 - e^{-\lambda x}) \frac{\beta^\alpha}{\Gamma(\alpha)} \lambda^{\alpha-1} e^{-\beta\lambda} d\lambda = 1 - \left(\frac{\beta}{\beta + x}\right)^\alpha.$$

Die resultierende Verteilungsfunktion F stimmt mit der Verteilungsfunktion einer Pareto-Verteilung mit den Parametern α und β überein. Somit kann die Pareto-Verteilung auch als Mischung von Exponentialverteilungen aufgefasst werden.

Prinzipiell könnte man auf diese Weise zur Erzeugung einer Realisation einer Pareto-Verteilung zunächst eine Realisation der Gamma-Verteilung erzeugen und dann mit dem ermittelten Parameter λ eine Realisation der Exponentialverteilung. Verglichen mit der Inversionsmethode (siehe Beispiel 2.8) ist diese Vorgehensweise jedoch zu aufwendig. \diamond

Eine Reihe weiterer Beispiele von Mischungen, die wieder auf eine Standardverteilung führen, ergeben sich im Rahmen der Bayes-Schätzung einer Input-Verteilung. Siehe Abschnitt 6.5. Diese Beispiele sollten jedoch nicht den Eindruck erwecken, dass die Mischung von Verteilungen häufig auf eine Standardverteilung führt. Das Gegenteil ist der Fall.

Nicht zuletzt begegnen uns Mischungen von Verteilungen bei zusammengesetzten Verteilungen. Siehe Anhang A.10. Hier treffen wir nur in den seltensten Fällen auf eine Standardverteilung.

Abschließend heben wir noch einmal hervor, dass sich die Kompositionsmethode in natürlicher Weise auf überabzählbare Mischungen überträgt:

1. Erzeuge Realisation bzgl. $g(\lambda)$.
2. Erzeuge Realisation x_λ bzgl. F_λ .
3. Setze $x = x_\lambda$.

Beispiel

2.28

Experten rechnen damit, dass im nächsten Jahr N Familien einen Antrag auf Asyl stellen werden. Dabei unterstellen sie, dass N eine Zufallsvariable mit Werten in \mathbb{N}_0 und Zähldichte $g(i)$, $i \in \mathbb{N}_0$, ist. Weiter gehen sie davon aus,

dass eine Familie aus einer zufälligen Anzahl Y von Personen besteht, wobei die Zufallsvariablen Y_1, Y_2, \dots unabhängig und identisch verteilt sind mit der Zähldichte $f(y)$, $y \in \mathbb{N}$.

Die Anzahl X der Personen, die aufgrund dieser Annahmen einen Antrag auf Asyl stellen werden, ist somit eine *zufällige* Summe von Zufallsvariablen, m.a.W.

$$X = Y_1 + Y_2 + \dots + Y_N$$

(mit $X = 0$ für $N = 0$). Um die Verteilung von X oder ausgewählte Kenngrößen (z.B. Erwartungswert) durch Simulation zu ermitteln, bietet sich zur Erzeugung einer Realisation x von X die Kompositionsmethode an:

1. Erzeuge eine Realisation k von N .
2. Erzeuge k (unabhängige) Realisationen y_1, \dots, y_k von Y .
3. Setze $x = y_1 + \dots + y_k$.

Ist N Poisson-verteilt, so liegt ein zusammengesetzter Poisson-Prozess vor und die Ergebnisse des Abschnitts 9 können ebenfalls herangezogen werden.
 \diamond

2.7 Berücksichtigung weiterer Verteilungszusammenhänge

Mit der Faltungs- und Kompositionsmethode haben wir bereits Verfahren kennen gelernt, die es ermöglichen, Zufallszahlen über einen Verteilungszusammenhang zu erzeugen. Im folgenden Beispiel nutzen wir einen Zusammenhang zwischen der Poisson- und der Exponentialverteilung aus.

2.29 Beispiel (Poisson-Verteilung; Zusammenhang mit Exponentialverteilung)

Tritt ein zufälliges Ereignis in unabhängigen, $\text{Expo}(\alpha)$ -verteilten Zeitabständen D_1, D_2, \dots ein, so ist nach Satz 9.2 die Häufigkeit X , mit der das Ereignis im Intervall $[0, 1]$ eintritt, $\text{Poi}(\alpha)$ -verteilt. Insbesondere ist $X = k$, falls

$$D_1 + \dots + D_k \leq 1 < D_1 + \dots + D_{k+1}$$

gilt (wobei $X = 0$ im Falle $D_1 > 1$).

Mit $D_j = -\frac{1}{\alpha} \ln Z_j$ (siehe Beispiel 2.7) und

$$\sum_{j=1}^k D_j \leq 1 \Leftrightarrow \frac{-1}{\alpha} \sum_{j=1}^k \ln Z_j \leq 1 \Leftrightarrow \ln(Z_1 Z_2 \dots Z_k) \geq -\alpha \Leftrightarrow Z_1 Z_2 \dots Z_k \geq e^{-\alpha}$$

erhalten wir schließlich die folgende Alternative zur Inversionsmethode:

1. Setze $a = e^{-\alpha}$, $b = 1$, $k = 0$.
2. Erzeuge eine Zufallszahl z_{k+1} und setze $b = bz_{k+1}$.
3. Ist $b < a$, setze $x = k$ und stoppe. Andernfalls setze $k = k + 1$ und fahre mit Schritt 2 fort. \diamond

Die Normalverteilung gehört zu den bedeutendsten Verteilungen der Statistik. Ihre wichtigsten Eigenschaften sind in Abschnitt A.3 zusammengefasst.

Um eine $N(\mu, \sigma^2)$ -verteilte Zufallszahl zu erzeugen, reicht es aus, eine $N(0, 1)$ -verteilte Zufallszahl x zu erzeugen und diese gemäß der Transformation $x' = \mu + \sigma x$ in eine $N(\mu, \sigma^2)$ -verteilte Zufallszahl x' zu überführen.

Da die Verteilungsfunktion $\Phi(x)$ der Standard-Normalverteilung nur in Tabellenform vorliegt, ist der mit der Inversionsmethode verbundene Rechenaufwand zu groß. Als Nachteil der Faltungsmethode (siehe Beispiel 2.24) erweist sich der „hohe“ Verbrauch an Zufallszahlen. Daher kommt der Box-Müller Methode und der Polar-Methode, die wir nun vorstellen werden, eine zentrale Bedeutung bei der Erzeugung einer $N(0, 1)$ -verteilten Zufallszahl x zu.

Beispiel (Normalverteilung; Box-Müller Methode)

2.30

Zu gegebenen Zufallszahlen z_1 und z_2 erzeugt die **Box-Müller Methode** zwei unabhängige Realisationen x_1 und x_2 einer $N(0, 1)$ -verteilten Zufallsvariablen gemäß folgender Vorschrift:

1. Erzeuge Zufallszahlen z_1 und z_2 .
2. Setze $x_1 = \sqrt{-2 \ln z_1} \cos(2\pi z_2)$ und $x_2 = \sqrt{-2 \ln z_1} \sin(2\pi z_2)$.

Zur formalen Überprüfung seien X und Y unabhängige, $N(0, 1)$ -verteilte Zufallsvariable und R und Θ die Polarkoordinaten des Vektors (X, Y) :

$$\begin{aligned} X &= R \cos \Theta, & R^2 &= X^2 + Y^2 \\ Y &= R \sin \Theta, & \tan \Theta &= Y/X. \end{aligned}$$

Da X und Y unabhängig sind, gilt für die Dichte $f(x, y)$ von (X, Y) :

$$f(x, y) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2} \frac{1}{\sqrt{2\pi}} e^{-y^2/2} = \frac{1}{2\pi} e^{-(x^2+y^2)/2}.$$

Mit Hilfe der Transformationsregel für Dichten folgt dann für die Dichte $g_1(r, \theta)$ von (R, Θ) :

$$g_1(r, \theta) = f(r \cos \theta, r \sin \theta) \cdot \left| \begin{array}{cc} \frac{\partial x}{\partial r} & \frac{\partial y}{\partial r} \\ \frac{\partial x}{\partial \theta} & \frac{\partial y}{\partial \theta} \end{array} \right| = \frac{1}{2\pi} e^{-r^2/2} r.$$

Führt man eine weitere Substitution mit $R' = R^2/2$ und $\Theta' = \Theta$ durch, so folgt schließlich für die Dichte $g_2(r', \theta')$ von (R', Θ') :

$$g_2(r', \theta') = g_1(\sqrt{2r'}, \theta') \cdot \left| \begin{array}{cc} \frac{\partial r}{\partial r'} & \frac{\partial \theta}{\partial r'} \\ \frac{\partial r}{\partial \theta'} & \frac{\partial \theta}{\partial \theta'} \end{array} \right| = \frac{1}{2\pi} e^{-r'}.$$

Somit sind die Zufallsvariablen R' und Θ' unabhängig; R' ist *Expo*(1)-verteilt und Θ' ist $U(0, 2\pi)$ -verteilt. Beide Verteilungen lassen sich einfach simulieren (Beispiele 2.6 und 2.7). Greift man hierauf zurück, so erhält man durch Rücksubstitution die Box-Müller Transformation. \diamond

2.31 Beispiel (Normalverteilung; Polar-Methode)

Als Nachteil der Box-Müller Transformation erweist sich der Rechenaufwand zur Bestimmung der Funktionswerte $\cos(z_2)$ und $\sin(z_2)$. Diese Schwierigkeit wird bei der folgenden Modifikation des Verfahrens (sog. **Polar-Methode**) umgangen:

1. Erzeuge Zufallszahlen z_1 und z_2 .
2. Setze $v_1 = 2z_1 - 1, v_2 = 2z_2 - 1, s = v_1^2 + v_2^2$.
3. Ist $s \leq 1$, setze $x_1 = \sqrt{\frac{-2 \ln s}{s}} v_1, x_2 = \sqrt{\frac{-2 \ln s}{s}} v_2$ und stoppe.
Andernfalls fahre mit Schritt 1 fort.

Die Polar-Methode kommt im Durchschnitt mit $4/\pi = 1.273$ Iterationen aus. Weitere Einzelheiten und eine Herleitung findet man z.B. in Ross (2013), Section 5.3. \diamond

Für die Erzeugung einer Beta-verteilten Zufallszahl bietet sich neben der Verwerfungsmethode (siehe Beispiel 2.18) die folgende Alternative an.

Beispiel (Beta-Verteilung)

2.32

Zur Erzeugung einer $Beta(\alpha, \beta)$ -verteilten Zufallszahl x betrachten wir die folgenden Spezialfälle:

- (a) Ist $\alpha = \beta = 1$, so stimmt die Beta-Verteilung mit der Gleichverteilung auf $[0, 1]$ überein.
- (b) Ist $\beta = 1$, so reduziert sich die Dichte auf

$$f(x) = \alpha x^{\alpha-1}, \quad 0 \leq x \leq 1,$$

und die Verteilungsfunktion geht über in $F(x) = x^\alpha$, $0 \leq x \leq 1$. Daher bietet sich die Inversionsmethode an und wir erhalten $x = F^{-1}(z) = z^{1/\alpha}$ zu gegebener Zufallszahl z .

- (c) Ist $\alpha = 1$, so folgt $x = F^{-1}(1 - z) = 1 - z^{1/\beta}$ in Analogie zu (b).
- (d) Für $\alpha > 0$ und $\beta > 0$ kann man den Zusammenhang mit der Gamma-Verteilung ausnutzen: Ist Y_1 eine $Gamma(\alpha, 1)$ -verteilte Zufallsvariable und Y_2 eine davon unabhängige, $Gamma(\beta, 1)$ -verteilte Zufallsvariable, so ist

$$X = \frac{Y_1}{Y_1 + Y_2}$$

$Beta(\alpha, \beta)$ -verteilt. Unter Ausnutzung dieses Zusammenhangs ergibt sich dann das folgende Verfahren zur Erzeugung einer Realisation einer $Beta(\alpha, \beta)$ -verteilten Zufallsvariable:

1. Erzeuge Realisation y_1 einer $Gamma(\alpha, 1)$ -verteilten Zufallsvariablen.
2. Erzeuge Realisation y_2 einer $Gamma(\beta, 1)$ -verteilten Zufallsvariablen.
2. Setze $x = \frac{y_1}{y_1 + y_2}$. \diamond

Beispiel (Ordnungsstatistiken)

2.33

Seien X_1, \dots, X_n unabhängige, identisch verteilte Zufallsvariable mit der Verteilungsfunktion F . Um eine Realisation von

$$X_{(n)} := \max\{X_1, \dots, X_n\} \quad \text{oder} \quad X_{(1)} := \min\{X_1, \dots, X_n\}$$

zu erzeugen, kann man n Realisationen bzgl. F erzeugen und den größten (kleinsten) Wert als Realisation von $X_{(n)}$ (bzw. $X_{(1)}$) wählen. Man kann

aber auch die Zusammenhänge

$$P(X_{(n)} \leq x) = P(X_1 \leq x, \dots, X_n \leq x) = F(x)^n$$

und

$$P(X_{(1)} > x) = P(X_1 > x, \dots, X_n > x) = [1 - F(x)]^n$$

ausnutzen und mit Hilfe der Inversionsmethode zu gegebener Zufallszahl z

$$x_{(n)} = F^{-1}(z^{1/n})$$

als Realisation von $X_{(n)}$ bzw. zu gegebener Zufallszahl $1 - z$

$$x_{(1)} = F^{-1}(1 - z^{1/n})$$

als Realisation von $X_{(1)}$ festlegen.

Speziell für $U[0, 1]$ -verteilte Zufallsvariable X_1, \dots, X_n erhält man dann zu gegebener Zufallszahl z die Realisationen $x_{(n)} = z^{1/n}$ bzw. $x_{(1)} = 1 - z^{1/n}$.

◇

2.8 Erzeugung mehrdimensionaler Zufallsvariablen

Sind X_1, \dots, X_n diskrete Zufallsvariable mit Werten $x_1 \in I_1, \dots, x_n \in I_n$, so lässt sich die gemeinsame Wahrscheinlichkeitsfunktion mit Hilfe bedingter Wahrscheinlichkeiten in der Form

$$\begin{aligned} P(X_1 = x_1, \dots, X_n = x_n) &= P(X_1 = x_1) \cdot \\ &\cdot P(X_2 = x_2 | X_1 = x_1) \dots P(X_n = x_n | X_1 = x_1, \dots, X_{n-1} = x_{n-1}) \end{aligned}$$

oder in der etwas kompakteren Schreibweise

$$\begin{aligned} p(x_1, \dots, x_n) \\ = p_{X_1}(x_1) p_{X_2|X_1}(x_2|x_1) \dots p_{X_n|X_1, \dots, X_{n-1}}(x_n|x_1, \dots, x_{n-1}) \end{aligned} \quad (2.14)$$

darstellen. Somit können wir eine Realisation \mathbf{x} des Zufallsvektors $\mathbf{X} = (X_1, \dots, X_n)^T$ rekursiv berechnen:

1. Erzeuge Zufallszahl x_1 bzgl. p_{X_1} .
 Erzeuge Zufallszahl x_2 bzgl. $p_{X_2|X_1}(\cdot|x_1)$.
 \vdots
 Erzeuge Zufallszahl x_n bzgl. $p_{X_n|X_1, \dots, X_{n-1}}(\cdot|x_1, \dots, x_{n-1})$.
2. Setze $\mathbf{x} = (x_1, \dots, x_n)^T$.

Natürlich hängt die Anwendbarkeit des Verfahrens von der Kenntnis der bedingten Verteilungen ab. Bei einer Markov-Kette (siehe Kapitel 8) sind diese bedingten Verteilungen nicht nur bekannt, sondern auch sehr einfach zu bestimmen.

Sind X_1, \dots, X_n stetige Zufallsvariable mit der Dichte $f(x_1, \dots, x_n)$, so erhält man eine (2.14) entsprechende Produktform

$$f(x_1, \dots, x_n) = f_{X_1}(x_1)f_{X_2|X_1}(x_2|x_1) \dots f_{X_n|X_1, \dots, X_{n-1}}(x_n|x_1, \dots, x_{n-1})$$

auf der Basis der bedingten Dichten (siehe Abschnitt A.9) und die Vorgehensweise lässt sich unmittelbar auf stetige Zufallsvariable übertragen. Auch hier hängt die Anwendbarkeit von der Kenntnis oder effizienten Bestimmung der bedingten Dichten ab.

Im Rahmen der Markov Chain Monte Carlo Verfahren (MCMC-Verfahren) stellen wir in Abschnitt 8.8 eine weitere Möglichkeit vor, Realisationen eines Zufallsvektors \mathbf{X} zu erzeugen. Die Idee besteht darin, die zugehörige Dichte oder Zähldichte $f(\mathbf{x})$ als stationäre Verteilung einer Markov-Kette (oder deren Verallgemeinerung mit überabzählbarem Zustandsraum) darzustellen und durch Simulation dieses Markov-Prozesses eine Folge $\mathbf{x}_0, \mathbf{x}_1, \dots$ von Realisationen zu erzeugen, die wiederum als Realisationen bzgl. (der in diesem Zusammenhang Grenzverteilung) $f(\mathbf{x})$ aufgefasst werden. Dabei bleiben die „ersten“ Realisationen unberücksichtigt, da es sich lediglich um ein asymptotisches Verfahren handelt. Siehe Beispiel 8.12 für weitere Einzelheiten.

Wir kommen nun zu einem wichtigen Spezialfall, der Erzeugung von Realisationen der multivariaten Normalverteilung.

Beispiel (Multivariate Normalverteilung)

2.34

Sei \mathbf{Y} eine n -dimensionale, normalverteilte Zufallsvariable mit Erwartungswert $\boldsymbol{\mu} = (\mu_1, \dots, \mu_n)^T \in \mathbb{R}^n$ und positiv definiter Kovarianzmatrix $\boldsymbol{\Sigma} = (\sigma_{ij}) \in \mathbb{R}^{(n,n)}$ (vgl. Anhang A.6).

Dann lässt sich \mathbf{Y} als affine Transformation

$$\mathbf{Y} = \boldsymbol{\mu} + B\mathbf{X}$$

unabhängiger, $N(0, 1)$ -verteilter Zufallsvariablen X_1, \dots, X_n darstellen, wobei $\mathbf{X} = (X_1, \dots, X_n)^T$ und $\boldsymbol{\Sigma} = BB^T$. Die Matrix B kann als untere Dreiecksmatrix gewählt und mit Hilfe der **Cholesky Zerlegung**

$$\begin{aligned} b_{11} &= \sqrt{\sigma_{11}}, & b_{j1} &= \frac{\sigma_{j1}}{b_{11}} \quad \text{für } 1 < j \leq n \\ b_{ii} &= \sqrt{\sigma_{ii} - \sum_{k=1}^{i-1} b_{ik}^2}, & b_{ji} &= \frac{1}{b_{ii}} \left(\sigma_{ji} - \sum_{k=1}^{i-1} b_{jk} b_{ik} \right) \quad \text{für } 2 \leq i < j \leq n \end{aligned}$$

effizient berechnet werden. Weitere Einzelheiten findet der interessierte Leser in Abschnitt A.6. Hieraus ergibt sich das folgende Verfahren zur Erzeugung einer Realisation \mathbf{y} von \mathbf{Y} :

1. Erzeuge n $N(0, 1)$ -verteilte Zufallszahlen x_1, \dots, x_n .
2. Führe die Cholesky Zerlegung $\boldsymbol{\Sigma} = BB^T$ durch.
3. Setze $\mathbf{x} = (x_1, \dots, x_n)^T$ und lege $\mathbf{y} = \boldsymbol{\mu} + B\mathbf{x}$ als Realisation von \mathbf{Y} fest.

Speziell für $n = 2$ erhalten wir (mit $\sigma_{ii} = \sigma_i^2$):

1. Erzeuge zwei $N(0, 1)$ -verteilte Zufallszahlen x_1, x_2 .
2. Setze $y_1 = \mu_1 + \sqrt{\sigma_1^2} x_1$ und $y_2 = \mu_2 + \frac{\sigma_{21}}{\sqrt{\sigma_1^2}} x_1 + \sqrt{\sigma_2^2 - \frac{\sigma_{21}^2}{\sigma_1^2}} x_2$.
3. Lege $\mathbf{y} = (y_1, y_2)^T$ als Realisation von \mathbf{Y} fest. \diamond

Abschließend gehen wir noch kurz auf das Konzept der Copula zur Erzeugung von Realisationen eines Zufallsvektors \mathbf{X} mit abhängigen Komponenten X_1, \dots, X_n ein.

Die theoretische Grundlage liefert der Satz von Sklar. Er besagt: Ist F die Verteilungsfunktion eines Zufallsvektors $\mathbf{X} = (X_1, \dots, X_n)^T$ mit den Randverteilungsfunktionen F_1, \dots, F_n , so existiert eine Abbildung $C : [0, 1]^n \rightarrow [0, 1]$, mit deren Hilfe F in der Form

$$F(x_1, \dots, x_n) = C(F_1(x_1), \dots, F_n(x_n)) \quad (2.15)$$

dargestellt werden kann. Sind die F_i stetig, so ist C eindeutig.

Abbildung C ist Verteilungsfunktion eines Zufallsvektors $\mathbf{Z} = (Z_1, \dots, Z_n)^T$ mit (i.d.R. abhängigen) $U[0, 1]$ -verteilten Komponenten Z_1, \dots, Z_n . Sie wird als **Copula** bezeichnet.

Umgekehrt lässt sich zu gegebener Copula C (als Verteilungsfunktion des Zufallsvektors $\mathbf{Z} = (Z_1, \dots, Z_n)^T$) und univariaten Verteilungsfunktionen F_1, \dots, F_n ein Zufallsvektor $\mathbf{X} = (F_1^{-1}(Z_1), \dots, F_n^{-1}(Z_n))^T$ mit der Verteilungsfunktion F und der Eigenschaft (2.15) konstruieren.

Sind die F_i stetig, so erhalten wir mit $x_i = F_i^{-1}(z_i)$ aus (2.15) eine explizite Darstellung

$$C(z_1, \dots, z_n) = F(F_1^{-1}(z_1), \dots, F_n^{-1}(z_n)) \quad (2.16)$$

von C in Abhängigkeit von F und den zugehörigen Randverteilungsfunktionen.

Die Gleichungen (2.15) und (2.16) sind fundamental für den Umgang mit Copulas. Die erste Gleichung zeigt, wie die gemeinsame Verteilungsfunktion F aus den Randverteilungen und einer Abhängigkeitsstruktur in Form einer Copula gewonnen werden kann. Die zweite Gleichung zeigt, wie diese Abhängigkeitsstruktur (Copula) aus der gemeinsamen Verteilung und den (stetigen) Randverteilungen extrahiert werden kann.

Darüber hinaus gilt die folgende Invarianz: Sei $\mathbf{X} = (X_1, \dots, X_n)^T$ ein Zufallsvektor mit stetigen Randverteilungen und Copula C und seien T_1, \dots, T_n streng monoton wachsende Funktionen. Dann ist C auch Copula von $\mathbf{Y} = (T_1(X_1), \dots, T_n(X_n))^T$.

Einfache Beispiele einer Copula sind die Unabhängigkeits-, die Komonotonie- und die Kontramonotonie-Copula. Zur Veranschaulichung sei $\mathbf{X} = (X_1, X_2)^T$ ein Zufallsvektor mit Verteilungsfunktion $F(x_1, x_2)$ und stetigen Randverteilungsfunktionen $F_1(x_1)$ und $F_2(x_2)$.

Unabhängigkeits-Copula. Sind X_1 und X_2 unabhängig, so ist bekanntlich $F(x_1, x_2) = F_1(x_1)F_2(x_2)$ und es folgt

$$C(z_1, z_2) = F(F_1^{-1}(z_1), F_2^{-1}(z_2)) = F_1(F_1^{-1}(z_1)) F_2(F_2^{-1}(z_2)) = z_1 z_2.$$

Komonotonie-Copula. Gilt $X_1 = Z$ und $X_2 = Z$, wobei $Z \sim U[0, 1]$, so liegt eine perfekte positive Abhängigkeit vor und wir erhalten

$$C(z_1, z_2) = \min\{z_1, z_2\}.$$

Kontramonetie-Copula. Gilt $X_1 = Z$ und $X_2 = 1 - Z$, wobei $Z \sim U[0, 1]$, so liegt eine perfekte negative Abhängigkeit vor und es folgt

$$C(z_1, z_2) = \max\{z_1 + z_2 - 1, 0\}.$$

Zu den klassischen Beispielen einer Copula gehören die Gauß- und die t -Copula. Die Gauß-Copula repräsentiert die Abhängigkeitsstruktur der multivariaten Normalverteilung. Ist $\mathbf{X} \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, so bleibt die zugehörige Copula bei Übergang zu $\mathbf{Y} \sim N(\mathbf{0}, \boldsymbol{\Sigma}_\rho)$ erhalten, wobei $\boldsymbol{\Sigma}_\rho$ die zur Kovarianzmatrix $\boldsymbol{\Sigma}$ zugehörige Korrelationsmatrix bezeichnet. Insbesondere gilt

$$C(z_1, \dots, z_n) = \Phi_{\boldsymbol{\Sigma}_\rho}(\Phi^{-1}(z_1), \dots, \Phi^{-1}(z_n)),$$

wobei Φ die Verteilungsfunktion der Standardnormalverteilung bezeichnet und $\Phi_{\boldsymbol{\Sigma}_\rho}$ die Verteilungsfunktion von \mathbf{Y} .

Sowohl die Gauß-, als auch die (auf der multivariaten t -Verteilung basierende) t -Copula können nur implizit angegeben werden. Dennoch sind sie von hoher Praxisrelevanz. Will man z.B. einen Zufallsvektor $\mathbf{X} = (X_1, \dots, X_n)^T$ mit abhängigen *Expo*(1)-verteilten Komponenten erzeugen, wobei die Abhängigkeitsstruktur durch die Gauß-Copula gegeben ist, so kann man wie folgt vorgehen:

1. Erzeuge $N(\mathbf{0}, \boldsymbol{\Sigma}_\rho)$ -verteilten Zufallsvektor $\mathbf{y} = (y_1, \dots, y_n)^T$ (wie in Beispiel 2.34).
2. Bestimme $z_i = \Phi(y_i)$ für $i = 1, \dots, n$.
3. Setze $\mathbf{x} = (-\ln(1 - z_1), \dots, -\ln(1 - z_n))^T$.

Das Beispiel der exponentialverteilten Zufallsvariablen mit einer auf der Gauß-Copula basierenden Abhängigkeitsstruktur zeigt bereits die prinzipielle Vorgehensweise bei der Erzeugung eines Zufallsvektors $\mathbf{X} = (X_1, \dots, X_n)^T$ mit stetigen Randverteilungsfunktionen F_1, \dots, F_n : Zunächst erzeugt man eine Realisation $\mathbf{z} = (z_1, \dots, z_n)^T \in [0, 1]^n$ der gewählten Copula. Hierzu stehen zahlreiche, größtenteils explizit darstellbare Copulas zur Verfügung. Anschließend bestimmt man für jede der Komponenten z_i die Inverse $x_i = F_i^{-1}(z_i)$ und erhält mit $\mathbf{x} = (x_1, \dots, x_n)^T$ die angestrebte Realisation des Zufallsvektors \mathbf{X} .

Nelsen (2007) ist ein Standardwerk über Copulas. Dort werden auch Simulationsaspekte behandelt. Siehe auch Joe (1997), Johnson (1982) und McNeil et al. (2005).

Aufgaben**2.9****Aufgabe****2.35**

Bestimmen Sie einen linearen Kongruenzgenerator, der die Zahlenfolge $0, 2, 4, 6, 8, 0, \dots$ erzeugt.

Aufgabe**2.36**

Betrachten Sie den linearen Kongruenzgenerator

$$x_{n+1} = (41x_n + c) \text{ modulo } 1000$$

mit $x_0 = 3$ und $c = 2$.

- (a) Wie müssen Sie den Parameter c anpassen, um die volle Periodenlänge zu erreichen?
- (b) Gibt es mehrere Möglichkeiten, den Parameter c anzupassen? Wenn ja, worauf sollte man bei der endgültigen Auswahl achten?

Aufgabe**2.37**

Zur Erzeugung einer Realisation x einer Zufallsvariablen X mit Hilfe der Inversionsmethode seien die folgenden Rechenschritte durchzuführen:

1. Erzeuge Zufallszahl $z \in (0, 1)$.
2. Setze $x = 3 + 2\sqrt{z}$.

Welche Verteilungsfunktion hat diese Zufallsvariable?

Aufgabe**2.38**

Betrachten Sie eine diskrete Zufallsvariable X mit Werten in der Menge $\{0, 1, \dots, 5\}$ und der Zähldichte

$$P(X = i) = p_i, \quad i = 1, 2, \dots, 5,$$

wobei $p_0 = 0.1$, $p_1 = 0.05$, $p_2 = 0.05$, $p_3 = 0.5$, $p_4 = 0.2$ und $p_5 = 0.1$.

Erzeugen Sie eine Realisation von X . Verwenden Sie hierzu

- (a) die (herkömmliche) Inversionsmethode
- (b) die folgende Modifikation: Sortieren Sie zunächst die p_i in absteigender Reihenfolge und wenden Sie anschließend die Inversionsmethode auf die neu angeordneten Werte an.

Vergleichen Sie beide Vorgehensweisen im Hinblick auf die Anzahl an Schritten, die im Mittel zur Erzeugung einer Realisation von X erforderlich sind.

2.39 Aufgabe

Erzeugen Sie eine Realisation einer $\text{Bin}(n, p)$ -verteilten Zufallsvariablen X

- (a) mit Hilfe der Inversionsmethode
- (b) mit Hilfe der Faltungsmethode
- (c) unter Ausnutzung einer Beziehung zwischen der $\text{Ber}(p)$ - und der $\text{Geo}_{\mathbb{N}}(p)$ -Verteilung.

2.40 Aufgabe

Verifizieren Sie die in Beispiel 2.10 angegebenen Rechenschritte zur Erzeugung einer Realisation einer Dreieck-Verteilung.

2.41 Aufgabe

Betrachten Sie eine Zufallsvariable X mit stückweise konstanter Dichte

$$f(x) = \begin{cases} c_i & \text{für } x_{i-1} \leq x \leq x_i, \\ 0 & \text{sonst} \end{cases}$$

wobei $c_i \geq 0$ für $i = 1, \dots, m$ und $x_0 < x_1 < \dots < x_m$.

Geben Sie die Rechenschritte an, die zur Erzeugung einer Realisation von X mit Hilfe der Inversionsmethode erforderlich sind.

2.42 Aufgabe

Verifizieren Sie: Ist z eine Zufallszahl, so ist auch $1 - z$ eine Zufallszahl.

Aufgabe**2.43**

Eine diskrete Zufallsvariable habe die Zähldichte

$P(X = 1)$	$P(X = 2)$	$P(X = 3)$	$P(X = 4)$	$P(X = 5)$
$1/24$	$3/24$	$5/24$	$7/24$	$8/24$

Nehmen Sie eine Zerlegung der Zähldichte vor, um die Alias-Methode anwenden zu können. Welche x_i und x_k kommen für die Festlegung von q_1 in Frage?

Aufgabe**2.44**

Betrachten Sie eine *Erlang*($n, 1$)-verteilte Zufallsvariable X mit der Dichte

$$f(x) = \frac{x^{n-1}}{(n-1)!} e^{-x}, \quad x \geq 0.$$

Erzeugen Sie eine Realisation von X mit Hilfe der Verwerfungsmethode. Wählen Sie hierzu die *Expo*($1/n$)-Verteilung.

- (a) Wie viele Realisationen von g sind im Mittel erforderlich, um eine Realisation von f zu erhalten? (Hinweis: Die notwendige Bedingung für c liefert das Maximum.)
- (b) Als Alternative bietet sich die Faltungsmethode an. Vergleichen Sie den Aufwand.

Aufgabe**2.45**

Eine Zufallsvariable X habe die Dichte $f(x)$. Ihr Ziel sei es, eine Realisation von X zu erzeugen, die größer als a ist. Hierzu verwenden Sie die Verwerfungsmethode mit f als alternativer Dichte.

Wie viele Realisation von X sind dann im Mittel notwendig, um eine Realisation zu erhalten, die größer als a ist?

Aufgabe**2.46**

Sie werfen zwei faire Würfel solange, bis jede der möglichen Augensummen $2, 3, \dots, 12$ aufgetreten ist. Dann stoppen Sie.

Welche Rechenschritte fallen an, um die erwartete Anzahl an Würfeln mittels Simulation zu berechnen.

2.47 Aufgabe

Erzeugen Sie mit Hilfe der Kompositionsmethode eine Realisation der nicht-negativen Zufallsvariablen X mit der Dichte

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2} + \frac{1}{2} e^{-x}, \quad x \geq 0.$$

2.48 Aufgabe

Erzeugen Sie eine Realisation einer multivariaten Normalverteilung mit Erwartungswert $\boldsymbol{\mu} = (1, 1, 0, 2)^T$ und Kovarianzmatrix $\Sigma = BB^T$, wobei

$$B = \begin{pmatrix} 1.0000 & 0 & 0 & 0 \\ 0.3000 & 0.9539 & 0 & 0 \\ -0.2000 & -0.2516 & 0.9469 & 0 \\ 0.4000 & -0.0210 & 0.6069 & 0.6864 \end{pmatrix}.$$

2.49 Aufgabe

Geben Sie eine Rechenvorschrift an zur Erzeugung einer Realisation einer nichtnegativen Zufallsvariablen X mit der Dichte

$$f(x) = 2.4 e^{-3x} + e^{-5x}, \quad x \geq 0.$$

2.50 Aufgabe

Seien X_1, X_2, \dots unabhängige, $\text{Expo}(\alpha)$ -verteilte Zufallsvariable und N eine von den X_i unabhängige, $\text{Geo}_{\mathbb{N}}(p)$ -verteilte Zufallsvariable. Erzeugen Sie eine Realisation der *zufälligen* Summe $X = X_1 + X_2 + \dots + X_N$.

Hinweis: Beispiel A.8.

Simulation stochastischer Systeme

Eine anwendungsorientierte Einführung

Waldmann, K.-H.; Helm, W.E.

2016, XI, 412 S. 88 Abb., Softcover

ISBN: 978-3-662-49757-9