

Schriftliche Ausarbeitung des Themas

Metropolis-Algorithmus

zum Seminar „Moderne Anwendungen der
Theorie der Markovketten“
Sommersemester 2016

Sabine Weingarten
Bachelor Mathematik

Prof. Dr. Wolfgang König
Priv. Doz. Dr. Konstantin Fackeldey
Technische Universität Berlin

Inhaltsverzeichnis

1	Einführung	2
1.1	Markovketten-Monte-Carlo-Verfahren (MCMC)	2
1.2	Metropolis-Algorithmus	2
2	Algorithmus	3
2.1	Symmetrische Referenzkette	4
2.2	Anwendung auf das Bergsteigerproblem	5
2.3	Allgemeine Referenzkette	6
2.4	Eine kleine Aussage für irreduzible Q Matrizen	7
2.5	Vorteile & Nachteile des Algorithmus	7
3	Eine konkrete Anwendung: Packproblem	8
4	Übersicht über weitere Anwendungsmöglichkeiten	10
5	Kurze Vorstellung verschiedener Variationen und interessanter Ideen	10
5.1	Burn-In	11
5.2	Thinning-Idee	11
5.3	Akzeptanzrate	11
5.4	Accept-Reject	11
5.5	Single Component vs. Block-at-a-time	11
6	Quellen	12

1 Einführung

1.1 Markovketten-Monte-Carlo-Verfahren (MCMC)

Ausgehend von einer irreduziblen¹ Übergangsmatrix P haben wir in Wahrscheinlichkeitstheorie 1 eine stationäre Verteilung² π konstruiert, d.h. also eine Wahrscheinlichkeitsverteilung, die die Bedingung $\pi = \pi P$ erfüllt.

Wir wollen uns nun mit dem umgekehrten Fall befassen. Das Ziel ist es, auf möglichst effiziente Weise aus der gegebenen, endlichen Menge Ω eine Stichprobe mit gegebener Verteilung π auf Ω zu ziehen.

Intuitiv naheliegende, naive Methoden, wie z.B. Simulation von Verteilung oder die Verwerfungsmethode, findet man in [1, Kapitel 1].

Bei deterministischen Lösungsverfahren, die sämtliche mögliche Fälle ausprobieren („try & error“), haben wir jedoch das Problem, dass die Laufzeit vom Umfang der Datenmenge abhängt, d.h. für sehr großes Ω können wir aufgrund der Laufzeit und des erforderlichen Speicherplatzbedarfs diese Methoden nicht nutzen. Ein einfacher Ansatz ist es, zufallsbasierte Algorithmen zu verwenden, die nur „typische“ Fälle behandeln.

Wir benötigen also ein Verfahren, das ausgehend von einer auf Ω gegebenen Verteilung π eine geeignete Markovkette konstruiert, die bei genügend langer Laufzeit die gewünschte Verteilung π genügend gut approximiert.

Eine solche Methode nennt sich Markovketten-Monte-Carlo-Verfahren (kurz MCMC³) und erzielt in der Praxis seit vielen Jahren trotz pessimistischer theoretischer Schranken⁴ gute Ergebnisse. Zwei bekannte Vertreter sind der Metropolis-Algorithmus sowie der Gibbs-Sampler, den wir im nachfolgenden Vortrag kennen lernen werden.

1.2 Metropolis-Algorithmus

Der Metropolis-Algorithmus wurde 1953 von Metropolis et al. [4] veröffentlicht und 1970 von Hastings [5] erstmals in einer weiteren Variation vorgestellt.

Durch den Algorithmus wird eine Markovkette auf dem Zustandsraum Ω so verändert, dass sie - bei beliebiger Startverteilung - aufgrund der sogenannten

¹ Irreduzibel: Jeder Zustand ist von jedem anderen aus mit positiver Wahrscheinlichkeit erreichbar.

² Oft auch als invariante Verteilung bekannt.

³ Engl. „Markov chain Monte Carlo methods“

⁴ D.h. Schranken die sehr hoch gewählt wurden und dadurch zu vielen unnötigen Schritten führen, aber dies kann nicht bewiesen werden.

Detailed-Balance-Bedingung⁵ gegen die gewünschte stationäre Verteilung π konvergiert.

Eine kurze Erinnerung:

Eine irreduzible, positiv rekurrente⁶ Markovkette konvergiert genau dann gegen eine stationäre Verteilung⁷, wenn sie aperiodisch⁸ ist. (siehe hierfür z.B. [2, Satz 2.6.15])

2 Algorithmus

Der Algorithmus wurde mittlerweile in verschiedenen Variationen veröffentlicht. Wir konzentrieren uns zuerst auf zwei in [3] publizierte Varianten, die den beiden ursprünglichen Veröffentlichungen entnommen wurden.

Grob gesprochen besteht der Algorithmus immer aus den folgenden Schritten:

- Definiere eine Referenz-Übergangsmatrix $Q = (q_{x,y})_{x,y \in \Omega}$ aperiodisch, irreduzibel, dünn besetzt
- Definiere mit Hilfe von Q eine Metropolis-Matrix $P = (p_{x,y})_{x,y \in \Omega}$
- Realisiere eine Markovkette mit Übergangsmatrix P , auch Metropolis-kette genannt, die aufgrund der Detailed-Balance-Bedingung konvergiert

⁵ Eine Markovkette mit Übergangsmatrix $P = (p_{i,j})_{i,j \in \Omega}$ heißt reversibel bzgl. der Verteilung π , wenn für alle $i, j \in \Omega$ gilt $\pi(i)p_{i,j} = \pi(j)p_{j,i}$.

⁶ Für jeden Zustand ist die erwartete Rückkehrzeit endlich.

⁷ Jede reversible Verteilung, also jede Verteilung, die die Detailed-Balance-Bedingung erfüllt, ist eine stationäre Verteilung.

⁸ D.h. alle Zustände der Markovkette haben Periode 1.

Wir wollen nun konkret analysieren wie die Matrizen zu wählen sind.

2.1 Symmetrische Referenzkette

Sei Q eine symmetrische Übergangsmatrix, dann ist Q reversibel bzgl. uniformer Verteilung auf Ω . Wir konstruieren nun eine neue Markovkette aus Q , um eine Kette mit stationärer Verteilung π zu erhalten.

Die neue Kette wird wie folgt entwickelt:

Die Referenzkette schlägt zunächst, gemäß der Verteilung von Q , einen Übergang vom aktuellen Zustand x zu einem Zustand y vor. Um zu vermeiden, dass immer wieder dieselben Übergänge angenommen werden, führt man eine Akzeptanzwahrscheinlichkeit $a_{x,y}$ ein. Bei Erfolg wird der von der Referenzkette vorgeschlagene Übergang nach y akzeptiert, ansonsten verharret man im momentanen Zustand x . Das heißt mit Wahrscheinlichkeit $1 - a_{x,y}$ wird der Vorschlag abgelehnt. Jede Ablehnung eines Übergangs verlangsamt die Kette und reduziert die Effizienz des Rechengangs. Ablehnungen sind aber ggf. nötig, um die gewünschte stationäre Verteilung zu erlangen.

Die Übergangsmatrix P der neuen Markovkette ist gegeben durch:

$$p_{x,y} = \begin{cases} q_{x,y}a_{x,y}, & \text{falls } y \neq x \\ 1 - \sum_{z:z \neq x} q_{x,z}a_{x,z}, & \text{falls } y = x \end{cases}$$

Wie wählt man nun die Akzeptanzwahrscheinlichkeit $a_{x,y}$?

Jede Verteilung π , die die Detailed-Balance-Bedingung erfüllt, ist eine stationäre Verteilung für P . Hieraus folgt Bedingung (1)

$$\pi(x)q_{x,y}a_{x,y} = \pi(y)q_{y,x}a_{y,x} \text{ für alle } x \neq y \quad (1)$$

Aus der Symmetrie von Q folgt, (1) gilt g.d.w.

$$b_{x,y} = b_{y,x} \text{ mit } b_{x,y} = \pi(x)a_{x,y} \quad (2)$$

Da $a_{x,y}$ eine Wahrscheinlichkeit ist, muss gelten $a_{x,y} \leq 1$. Hieraus folgt für b

$$b_{x,y} \leq \pi(x) \quad (3)$$

$$b_{x,y} = b_{y,x} \leq \pi(y) \quad (4)$$

Das Ablehnen von Übergängen führt zu Laufzeitverlusten, daher muss die Lösung b für (2)-(4) möglichst maximal gewählt werden. Offensichtlich sind alle Lösungen nach oben beschränkt durch

$$b_{x,y} = \pi(x) \wedge \pi(y) := \min\{\pi(x), \pi(y)\} \quad (5)$$

Somit ist die Akzeptanzwahrscheinlichkeit $a_{x,y} = \left(\frac{\pi(y)}{\pi(x)}\right) \wedge 1$

Insgesamt folgt also, dass die Metropolis-Matrix $P = (p_{x,y})_{x,y \in \Omega}$ der neuen Kette definiert ist als:

$$p_{x,y} = \begin{cases} q_{x,y} \min\{1, \frac{\pi(y)}{\pi(x)}\}, & \text{falls } y \neq x \\ 1 - \sum_{z:z \neq x} q_{x,z} \min\{1, \frac{\pi(z)}{\pi(x)}\}, & \text{falls } y = x \end{cases}$$

Offensichtlich ist π aufgrund der gewählten Nebenbedingungen stationäre Verteilung für die Metropolisübergangsmatrix P .

Die Metropolkette hängt nur vom Verhältnis $\frac{\pi(x)}{\pi(y)}$ ab. Oft ist $\pi(x)$ von der Form $\frac{h(x)}{Z}$, mit bekannter Funktion $h : \Omega \rightarrow [0, \infty)$ und der Normierungskonstante $Z = \sum_{x \in \Omega} h(x)$. Besonders bei gigantisch großem Ω kann Z extrem

schwer bis gar nicht berechenbar sein. Da $\frac{\pi(x)}{\pi(y)} = \frac{h(x)}{h(y)}$ ist, können wir die Kette simulieren, ohne Z explizit zu berechnen. Dies ist besonders bei Optimierungsproblemen häufig der Fall. Hierfür betrachten wir nun eine Anwendung.

2.2 Anwendung auf das Bergsteigerproblem

Sei f eine reelle Funktion⁹, die auf der Knotenmenge V eines Graphen definiert ist. In vielen Anwendungen wird ein Knoten x gesucht, für den $f(x)$ maximal ist. Wenn der Definitionsbereich sehr groß ist, kann eine vollständige Suche zu teuer sein. Ein „hill climb“ (Bergsteigeralgorithmus) versucht Maxima von f wie folgt zu bestimmen:

Wenn am Punkt x , und für einen Nachbarn y von x gilt $f(y) > f(x)$, dann gehe zu y . Hat f lokale Maxima, die keine globalen Maxima sind, kann es passieren, dass der Bergsteiger stecken bleibt, bevor ein globales Maximum gefunden wird. Eine Lösung hierfür ist es, Übergänge zu randomisieren, sodass von einem lokalen Maximum aus mit einer gewissen Wahrscheinlichkeit auch zu einem niedrigeren Punkt übergegangen wird.

Wir nehmen der Einfachheit halber an, dass Ω ein regulärer Graph¹⁰ ist, sodass die einfache Irrfahrt auf Ω eine symmetrische Übergangsmatrix hat.

⁹ Häufig findet sich hier bei thermodynamischen Modellen der Boltzmann-Faktor $\lambda^{f(x)} = e^{-E \cdot \beta}$ mit $\beta = \frac{1}{k \cdot T}$, wobei E eine Energie, T die Temperatur und k die Boltzmann-Konstante beschreibt. Siehe hierfür z.B. den Vortrag zum Thema „Simulated Annealing“ oder auch das Ising-Modell [z.B. in 3, 3.3.5].

¹⁰ Ein Graph heißt regulär, falls alle seine Knoten gleich viele Nachbarn haben, d.h. gleichen Knotengrad besitzen.

Sei $\lambda \geq 1$ und $\pi_\lambda(x) = \frac{\lambda^{f(x)}}{Z(\lambda)}$, wobei $Z(\lambda) := \sum_{x \in \Omega} \lambda^{f(x)}$ (Normierungsfaktor).

Solange $\pi_\lambda(x)$ durch den Anstieg von $f(x)$ wächst, bevorzugt die Verteilung π_λ Knoten x für die $f(x)$ groß ist. Wenn $f(y) < f(x)$ ist, wird der Übergang von x zu y mit Wahrscheinlichkeit $\lambda^{-[f(x)-f(y)]}$ akzeptiert. Definiere

$$\Omega^* := \{x \in \Omega : f(x) = f^* := \max_{y \in \Omega} f(y)\}$$

Dann gilt

$$\lim_{\lambda \rightarrow \infty} \pi_\lambda(x) = \lim_{\lambda \rightarrow \infty} \frac{\frac{\lambda^{f(x)}}{\lambda^{f^*}}}{|\Omega^*| + \sum_{x \in \Omega \setminus \Omega^*} \frac{\lambda^{f(x)}}{\lambda^{f^*}}} = \frac{\mathbb{1}_{\{x \in \Omega^*\}}}{|\Omega^*|}$$

Für $\lambda \rightarrow \infty$ konvergiert die invariante Verteilung π_λ dieser Metropolis-Kette gegen die Gleichverteilung auf der Menge der globalen Maxima von f .

2.3 Allgemeine Referenzkette

Wir betrachten nun noch kurz den Fall der Konstruktion einer Metropolis-Kette mit nicht symmetrischer Referenz-Übergangsmatrix Q , wie er von Hastings eingeführt wurde:

Sei Q eine irreduzible stochastische Übergangsmatrix und π eine beliebige Verteilung auf Ω . Wir definieren $P = (p_{x,y})_{x,y \in \Omega}$ durch:

$$p_{x,y} = \begin{cases} q_{x,y} \min\{1, \frac{\pi(y)q_{y,x}}{\pi(x)q_{x,y}}\}, & \text{falls } y \neq x \text{ und } q_{x,y} > 0 \\ 0, & \text{falls } x \neq y \text{ und } q_{x,y} = 0 \\ 1 - \sum_{z:z \neq x} p_{x,z}, & \text{falls } y = x \end{cases}$$

P ist per Definition zeilenstochastisch und für alle $x, y \in \Omega$ ist $p_{x,y} \in [0, 1]$, d.h. P ist eine stochastische Matrix und π ist invariante Verteilung für P , was man wie folgt sieht:

Für $x \neq y$ gilt:

Wenn $q_{x,y} = 0$ oder $q_{y,x} = 0 \Rightarrow p_{x,y} = p_{y,x} = 0$

Wenn $q_{x,y} > 0$ und $q_{y,x} > 0$:

$$\begin{aligned} \pi(x)p_{x,y} &= \pi(x)q_{x,y} \min\{1, \frac{\pi(y)q_{y,x}}{\pi(x)q_{x,y}}\} \\ &= \min\{\pi(x)q_{x,y}, \pi(y)q_{y,x}\} \\ &= \pi(y)q_{y,x} \min\{1, \frac{\pi(x)q_{x,y}}{\pi(y)q_{y,x}}\} \\ &= \pi(y)p_{y,x} \end{aligned}$$

d.h. π ist reversibel für P und somit invariante Verteilung.

2.4 Eine kleine Aussage für irreduzible Q Matrizen

Sei Q irreduzibel. Unter der Voraussetzung, dass $q_{x,y} > 0 \Leftrightarrow q_{y,x} > 0$ können wir leicht sehen, dass P ebenfalls irreduzibel ist:

Sind $x, y \in \Omega$, dann existiert ein Pfad von x nach y mit nur jeweils positiven Übergängen. P erlaubt dieselben Übergänge wie Q , d.h.

$$p_{x,y}, p_{y,x} > 0 \Leftrightarrow q_{x,y} > 0$$

2.5 Vorteile & Nachteile des Algorithmus

Vorteil:

- Es ist nur der Quotient $\frac{\pi(y)}{\pi(x)}$ notwendig um P zu berechnen (siehe 2.2)
- Oft einfach zu implementieren
- Vorgeschlagene Schritte müssen nicht gespeichert werden

Nachteil:

- Häufig werden Nachbarschaftsstrukturen genutzt, was bei Knoten mit hohem Knotengrad, also mit vielen Nachbarn, problematisch sein kann.

3 Eine konkrete Anwendung: Packproblem

Der Metropolis-Algorithmus wird häufig für Optimierungsprobleme genutzt. Wir betrachten folgende Problemstellung:

Gegeben sind Objekte $1, \dots, m$ mit Größen a_1, \dots, a_m und ein Rucksack mit Kapazität K , wobei gilt $a_1 + \dots + a_m > K$. Wir definieren einen Mitnahmevektor $y = (y_1, \dots, y_m) \in \{0, 1\}^m$, wobei 0 das Weglassen und 1 das Mitnehmen des Objekts bezeichnet.

Wir beginnen mit der Irrfahrt auf dem Hyperwürfel $H := \{0, 1\}^m$. Es gilt:

$$y \sim y' \Leftrightarrow y, y' \text{ unterscheiden sich in höchstens einer Komponente}$$

Sei $a = (a_1, \dots, a_m) \in \mathbb{N}^m$ und $a_1 + \dots + a_m > K$. Sei Ω die Menge der Tupel $y \in H$ mit der Eigenschaft

$$\langle a, y \rangle := \sum_{k=1}^m a_k y_k \leq K$$

Wir suchen einen Metropolis-Algorithmus, der eine Markovkette simuliert, für die die Gleichverteilung auf $\Omega = \{y \in H : \langle a, y \rangle \leq K\}$ eine invariante Verteilung ist.

1. Definiere Q , sodass

$$q_{y,y'} = \begin{cases} \frac{1}{2m}, & \text{falls } y \sim y' \\ \frac{1}{2}, & \text{falls } y = y' \\ 0, & \text{sonst} \end{cases} \quad {}^{11}$$

für alle Übergänge von $y \in H$ nach $y' \in H$ gilt.

2. Definiere P , sodass

$$p_{y,y'} = \begin{cases} \frac{1}{2m} \min\left\{1, \frac{\pi(y')q_{y',y}}{\pi(y)q_{y,y'}}\right\}, & \text{falls } y \sim y' \\ 1 - \sum_{k \neq y} p_{y,k}, & \text{falls } y = y' \\ 0, & \text{sonst} \end{cases}$$

für alle Übergänge von $y \in H$ nach $y' \in H$ gilt, wobei π die Gleichverteilung auf Ω ist, d.h. $\pi(y) = 0$, falls $y \notin \Omega$.

¹¹ Der Faktor $\frac{1}{m}$ ergibt sich, wie man leicht sieht, durch Gleichverteilung auf der Menge aller y' , für die für ein bestimmtes y gilt $y \sim y'$.

Wir haben also eine Markovkette mit Übergangsmatrix P und Gleichverteilung auf $\Omega = \{y \in H : \langle a, y \rangle \leq b\}$ als reversible Verteilung. Für jeden Vektor y , der eine gültige Lösung auf Ω liefert, sieht man leicht, dass man mit positiver Wahrscheinlichkeit wieder zum Nullvektor zurückkehren kann, indem man schrittweise die Nichtnulleinträge des Vektors nacheinander auf Null setzt (d.h. zum jeweiligen Nachbarn wechselt). Ebenso kann man vom Nullvektor aus jeden gültigen Lösungsvektor y erreichen. Die Markovkette ist also irreduzibel.

Es bleibt zu zeigen, dass die Markovkette zudem aperiodisch ist.

Angenommen $\sum_{k=1}^m a_k > K$, dann gibt es ein $j \in [1, m]$ und einen Vektor $y = (y_1, \dots, y_j, \dots, y_m) \in H$, mit $y_j \neq 0$, sodass $\sum_{k=1}^m a_k y_k \leq K$, aber es gilt $\sum_{k=1}^m a_k y_k + a_j > K$, d.h. $p_{y,y} > 0$. Mit der Irreduzibilität folgt die Behauptung für alle $y \in H$ mit $y_j = 0$ - die Markovkette ist also aperiodisch.

Nun betrachten wir noch den Fall, dass die m Objekte zusätzlich einen bestimmten Wert w haben (z.B. der Preis des Objekts). Ziel ist es, für gegebene Werte w_1, \dots, w_m die Funktion $U(y) = \sum_{k=1}^m w_k y_k$ zu maximieren. Da sich der Metropolis-Algorithmus in der bisherigen Form mit Minimierungsproblemen befasst, betrachten wir einfach die negativen Werte und minimieren die Funktion $U'(y) = -\sum_{k=1}^m w_k y_k$. Sei $\tilde{\pi}(y) = \frac{1}{Y_\beta} e^{-\beta U'(y)} = \frac{1}{Y_\beta} e^{\beta U(y)}$ mit $Y_\beta = \sum_{y \in \Omega} e^{-\beta U'(y)}$ für $\beta > 0$. Ausgehend von der Übergangsmatrix P gemäß 2. suchen wir eine Übergangsmatrix \tilde{P} mit invarianter Verteilung $\tilde{\pi}$. Wir benötigen hier nun also

$$\begin{aligned} \min\left\{1, \frac{\tilde{\pi}(y') p_{y',y}}{\tilde{\pi}(y) q_{y,y'}}\right\} &= \min\left\{1, \underbrace{\frac{\frac{1}{Y_\beta} e^{\beta U(y')}}{\frac{1}{Y_\beta} e^{\beta U(y)}}}_{=1} \frac{p_{y',y}}{p_{y,y'}}\right\} \\ &= \min\left\{1, e^{\beta(U(y')-U(y))} \frac{p_{y',y}}{p_{y,y'}}\right\}, \end{aligned}$$

wobei $(U(y') - U(y))$ gerade den Wertzuwachs beschreibt. Für $y \sim y'$ ergibt sich daher

$$\tilde{p}_{y,y'} = p_{y,y'} \min\left\{1, e^{\beta(U(y')-U(y))} \frac{p_{y',y}}{p_{y,y'}}\right\} = \min\{p_{y,y'}, e^{\beta(U(y')-U(y))} p_{y',y}\}.$$

Die gesuchte Übergangsmatrix ist somit gegeben durch

$$\tilde{p}_{y,y'} = \begin{cases} \min\{p_{y,y'}, e^{\beta(U(y')-U(y))}p_{y',y}\}, & \text{falls } y \sim y' \\ 1 - \sum_{k \neq y} \tilde{p}_{y,k}, & \text{falls } y = y' \\ 0, & \text{sonst} \end{cases}$$

4 Übersicht über weitere Anwendungsmöglichkeiten

Weitere typische Anwendungsszenarien sind z.B.

1. Physikalische Modelle, wie z.B. das Ising-Modell oder
2. Optimierungsmodelle mit zeitabhängiger Temperaturabkühlung (Simulated Annealing)
3. q-Färbung auf Graphen
4. Problem des Handelsreisenden
5. Kartenmischen
6. Hard-Core-Modell
7. Self-avoiding walk
8. Anwendungen in der Bayesschen Statistik, z.B. im Logit-Modell mit A-posteriori-Verteilung als Zielverteilung

5 Kurze Vorstellung verschiedener Variationen und interessanter Ideen

Der Metropolis-Algorithmus wird häufig als einer der Top 10 Algorithmen des 20. Jahrhunderts bezeichnet. Dementsprechend umfangreich ist die Vielzahl an Artikeln zum Thema. Hierunter finden sich immer wieder weitere Ideen zur Umsetzung und Anwendung des Algorithmus, die hier kurz erwähnt werden sollen, aber selbstverständlich keinem Anspruch auf Vollständigkeit gerecht werden.

5.1 Burn-In

Die Markovkette konvergiert unabhängig vom gewählten Startpunkt, jedoch kann von einer geeigneten Wahl die Zeit bis zur Konvergenz ins Gleichgewicht abhängen. Unter Burn-In versteht man die ersten k Stichproben, die meist wenig oder gar nicht brauchbar sind. Hierbei ist k abhängig vom konkreten Problem.

5.2 Thinning-Idee

Da die Stichproben per Konstruktion alle korreliert sind, scheint es eine weit verbreitete Idee zu sein, nur jeden k -ten Schritt der Kette zu behalten, um näher an eine u.i.v. Stichprobe zu kommen.

5.3 Akzeptanzrate

Es gilt als sinnvoll die Rate zu überwachen mit der Übergänge akzeptiert werden, um zu vermeiden, dass die Kette bei zu hoher Akzeptanzrate nicht gut genug gemischt wird, oder das bei zu niedriger Rate, kaum Vorschläge angenommen werden und die Kette im selben Zustand verharrt. Einige interessante Empfehlungen finden sich in [6]. So sollte die Akzeptanzrate hiernach für univariate Probleme um die 44% liegen und fällt für multivariate Probleme bei zunehmender Dimension n , für $\lim_{n \rightarrow \infty}$ bis auf 23%.

5.4 Accept-Reject

Rejecting-Algorithmen oder auch Accept-Reject-Algorithmen sind weitere Algorithmen, um Stichproben mit gewünschter Verteilung zu erhalten. Ausführliche Informationen zur Kombination mit dem Metropolis-Algorithmus finden sich z.B. in [6].

5.5 Single Component vs. Block-at-a-time

Für multivariate Zufallsvariablen wird häufig komponentenweise (Single Component) vorgegangen. D.h. erst führt man den Metropolis-Algorithmus für die 1. Komponente durch und hierauf basierenden dann für die jeweils nachfolgende Komponente. Block-at-a-time ist eine Variante in der gleichzeitig mehrere Komponenten betrachtet werden.

6 Quellen

1. „Stochastische Algorithmen“, W. König
<http://www.wias-berlin.de/people/koenig/www/AlgStoch.pdf>
2. „Stochastische Modelle“, M. Scheutzow
<http://www3.math.tu-berlin.de/Vorlesungen/WS15/StoMo/stochmod2015.pdf>
3. „Markov Chains and Mixing Times“, David A. Levin, Yuval Peres and Elizabeth L. Wilmer
<http://research.microsoft.com/en-us/um/people/peres/markovmixing.pdf>
4. „Equation of State Calculations by Fast Computing Machines“, N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller und E. Teller, in: Journal of Chemical Physics. Band 21, 1953, S. 1087–1092
5. „Monte Carlo Sampling Methods Using Markov Chains and Their Applications“, W.K. Hastings, in Biometrika. Band 57, 1970, S. 97–109
6. „Efficient Metropolis Jumping Rules“, A. Gelman, G.O. Roberts, W.R. Gilks, in: Bayesian Statistics. 1996, S. 599-607

Weitere interessante Quellen

- „A Short History of Markov Chain Monte Carlo: Subjective Recollections from Incomplete Data“, Christian Robert, George Casella, in: Statistical Science. 2011, Vol. 26, No. 1, 102–115
- „Understanding the Metropolis-Hastings-Algorithm“, S. Chib, E. Greenberg, in: The American Statistician. 1995, Vol. 49, No. 4