Markow-Ketten: Wie man Praktikanten aus der Schule mit stochastischen Inhalten beschäftigen kann

Albrecht Gebhardt und Manfred Borovcnik, Klagenfurt

Zusammenfassung: *Dieser Beitrag gibt einen Überblick über ein Ferialpraktikum, das im Sommer 2011 am Institut für Statistik der Universität Klagenfurt durchgeführt wurde. Die beiden Praktikanten implementierten einen Textsimulator, der das Prinzip Markowscher Ketten höherer Ordnung zur Anwendung bringt. Wir zeigen auch Anwendungen von Markow-Ketten auf, die – im Hintergrund – die Motivation der Praktikanten mitbestimmt haben.*

1 Die Ausgangslage

Markow-Ketten vereinen zwei vordergründig konträre Sichtweisen, nämlich Muster und Zufall. Diese stochastischen Modelle werden in den Anwendungen immer wichtiger. Wenn man junge Praktikanten an die Universität bringt, so kann man sie auch dadurch begeistern, dass man ihnen etwas beibringt, was später im Berufsleben als Mathematiker unabdingbar wird, nämlich Programmieren oder Programme anpassen. Wenn das zusätzlich noch einen motivierenden Hintergrund hat, umso besser.

Muster und Zufall

Der reine Zufall wird vielfach durch folgende Eigenschaften beschrieben: Zufall ist ein Sammelbecken für Phänomene, welche geprägt sind durch ein Fehlen von Mustern, eine fehlende Kontrolle über die Ausgänge und Fairness (siehe u. a. Borovcnik 2011). Wahrscheinlichkeit ist nur eines von vielen Konzepten, mit Zufall umzugehen; als solches ist es mit konkurrierenden Vorstellungen konfrontiert.

Unter den Fehlvorstellungen taucht immer wieder die Suche nach Mustern in den vorhandenen Daten auf. Es ist ein mühsamer Weg, Lernende davon zu überzeugen, dass der Zufall jedes Muster „erzeugen“ kann, dass es aber keine Präferenzen gibt. Erst unter den so genannten iid-Bedingungen (independent, identically distributed), das sind Versuche, die unabhängig unter denselben Bedingungen realisiert werden, ist es sinnvoll, Häufigkeiten auszuzählen und damit eine unbekannte Wahrscheinlichkeit zu schätzen. Wenn die Daten durch davon abweichende Bedingungen entstehen, ist eine solche Deutung von Wahrscheinlichkeit und eine Schätzung unzulässig.

Das ist das Problem in vielen Anwendungen, weil ja die Zuschreibung des Charakters einer Zufallsstichprobe für die vorhandenen Daten oft mehr einem Wunschdenken entspringt als dann tatsächlich erfüllt ist. In der Praxis werden verstärkt Prozesse untersucht, die keineswegs als unabhängige Realisierungen desselben Experiments gelten; Abhängigkeiten zwischen den Daten sind vorhanden. Wie soll man dann stochastische Modelle anwenden?

Markow-Ketten bilden irgendwie eine Spange zwischen zwei widersprüchlichen Konzepten – nämlich Muster und Zufall. Der Zufall wird durch vom erreichten Zustand abhängige bedingte Wahrscheinlichkeiten modelliert. Insofern ein reizvoller Ansatz, Muster und Zufall zu „versöhnen“.

Praktikanten an der Universität

Vermehrt bemühen sich die Universitäten, interessierte Jugendliche schon zu Schulzeiten an die Universität heranzuführen. Dazu gibt es Projekte im Sommer während der Schulferien, in denen die Jugendlichen am Institut „eingebunden“ werden. Es erhebt sich natürlich die Frage, womit man sie „beschäftigt“. Soll man sie die Grundbegriffe studieren lassen, gibt man ihnen Aufgaben wie in der Schule, die sie lösen können und bespricht man dann die Lösungen, oder bringt man ihnen Programmieren bei, was in der späteren Praxis enorm wichtig ist?

Um das Interesse der jungen Leute zu wecken, wird man vielleicht Kopieren des Schulstoffs vermeiden, und sie gleich an wichtige praktische Fragestellungen heranführen. Die Motivation ist größer, wenn man überraschende Begriffsbildungen verwendet und wenn sie ein Ergebnis herzeigen können.

Mit Markow-Ketten kann man Muster auf probabilistischem Weg erzeugen, das trifft zunächst auf Erstaunen. Man kann dabei literarische Texte oder bestimmte Musik zufällig erzeugen wollen. Wir haben uns für Goethe entschieden. Als technisches Hilfsmittel setzen wir R ein, das die Praktikanten im Verlauf der Arbeit nach und nach erlernen.

Zwei Anwendungen von Markow-Ketten

Am einfachsten führt man in das Prinzip anhand der klassischen Anwendungen ein.

Die Simulation von Texten ist eine der klassischen Anwendungen der von Andrej Andrejevich Markow

eingeführten und nach ihm benannten Zufallsfolgen. Hierbei wird ein Text in seine Bausteine (Buchstaben oder Wörter) zerlegt und dann die Wahrscheinlichkeiten des Aufeinanderfolgens dieser Bausteine als relative Häufigkeiten ermittelt. Dieses Aufeinanderfolgen passiert nicht unabhängig vom jeweils davorstehenden Kontext, da sich naturgemäß ein sinnvoller Text von einer rein zufälligen Buchstabenfolge deutlich unterscheidet. Anschließend lassen sich ausgehend von einem oder mehreren Startbausteinen Texte simulieren, indem neue Bausteine gemäß den vorher geschätzten bedingten Verteilungen ausgewürfelt werden. In diesem Beitrag werden Buchstaben als Textbausteine betrachtet. Der technische Aufwand steigt, wenn man zu Worten übergeht, das Prinzip bleibt aber das gleiche.

Christmann (o. J.) beschreibt vielfältige Verbindungen zwischen Mathematik und Musik und zwischen Zufall und Musik. Eine betrifft die Modellierung von Musikstücken mittels Markow-Ketten. Er schränkt dabei die Komplexität des musikalischen Geschehens ein und modelliert nur die Tonhöhe. Wie charakteristisch sind bestimmte Komponisten? Wie kann man deren Melodieführung „simulieren“?

Bei Vorgabe des Rhythmus kann man oft schon bereits nach 3 oder 4 Tönen den Komponisten erkennen. Es gibt also nach n aufeinander folgenden Tönen bereits eine enge Wahl für den folgenden Ton. Diese Übergänge schätzt man aus den Werken eines Komponisten und simuliert mit diesen Übergangswahrscheinlichkeiten die weitere Abfolge. Orientiert man sich nur am vorhergehenden Ton, so wird das Ergebnis eine schlechte Übereinstimmung mit dem Komponisten bringen. Allerdings mit „Tiefen“ von 3 oder 4 erhält man – mit einer entsprechenden händischen Nachbearbeitung (Arrangement) – durchaus akzeptable Ergebnisse, wie man auch auf der Internet-Seite von Christmann hören kann (etwa im Beispiel eines Gospel-Songs).

Das Ergebnis kann noch stark verbessert werden, denn aus der Vielzahl der musikalischen Parameter (Tonhöhen, Tondauern, Tempo, Klangfarbe, …), wurde ja nur die Tonhöhenfolge als ein durchaus wesentlicher Charakterzug modelliert. Christmann selbst zieht aber einer solchen Verfeinerung zur Simulation bestimmter Komponisten einen systemischeren Ansatz vor, der nicht mehr auf Markow-Ketten setzt, siehe die angegebene Internet-Seite.

Manches Mal sind Markow-Ketten an einen anderen Prozess von Zufallsexperimenten (mit unabhängigen identischen Wiederholungen des Experiments) angeschlossen, wie etwa im Beitrag von Riehl (2011) in diesem Heft. Wirft man einen Würfel, so kann man nach der Zahl der Runs (der Abfolgen gleicher Ergebnisse) bestimmter Länge (etwa der Länge 4) fragen. Der Zustandsraum besteht also nicht aus den Ergebnissen des Würfelns selbst, sondern aus dem „Zustand“, wie viele Runs der vorgegebenen Länge man schon hat und wie viele Ergebnisse im letzten Run schon vorhanden sind. Darauf kann man dann die neuen bedingten Wahrscheinlichkeiten ansetzen, weitere Zustände zu erreichen.

2 Markow-Ketten

Bei der Textsimulation kommen diskrete Markow-Ketten mit endlichem Zustandsraum zur Anwendung. Der Zustandsraum wird durch das entsprechende Alphabet $L = \{l_i | i = 1, \ldots, n\}$ gebildet, der vorgegebene Text sei eine endliche Folge $T = \{c_1, c_2, \ldots, c_N\}$ von N Zeichen c_t aus dem Alphabet L. Wenn man die Folge T als die Realisierung einer Markow-Kette erster Ordnung auffassen will, muss diese die sogenannte Markow-Eigenschaft besitzen, siehe z. B. (Viertl 1997):

$$P(C_t = c_t | C_{t-1} = c_{t-1}, C_{t-2} = c_{t-2}, \ldots, C_1 = c_1) = P(C_t = c_t | C_{t-1} = c_{t-1}) \quad (1)$$

D. h., die stochastische Abhängigkeit darf sich nur auf den unmittelbaren Vorgänger des aktuellen Zustands erstrecken. Bei der Simulation eines Textes aus einzelnen Buchstaben ist die Beschränkung der Abhängigkeit auf den unmittelbaren Vorgängerbuchstaben aber nicht ausreichend. Hier kommt nun die Markow-Kette k-ter Ordnung ins Spiel, deren grundlegende Eigenschaft lautet:

$$P(C_t = c_t | C_{t-1} = c_{t-1}, C_{t-2} = c_{t-2}, \ldots, C_1 = c_1) = P(C_t = c_t | C_{t-1} = c_{t-1}, \ldots, C_{t-k} = c_{t-k}) \quad (2)$$

Somit ist der aktuelle Zustand nun von seinen k Vorgängern abhängig. Für diskrete Markow-Prozesse mit endlichem Zustandsraum lassen sich die ein- und mehrstufigen Übergangswahrscheinlichkeiten in einer Matrix oder einem Array höherer Ordnung zusammenfassen,

$$P_1 = ((p_{ij}))_{i,j=1,\ldots,n}$$
$$p_{ij} = P(C_t = j | C_{t-1} = i)$$
$$P_2 = ((p_{ijk}))_{i,j,k=1,\ldots,n}$$
$$P_{ijk} = P(C_t = j | C_{t-1} = i, C_{t-2} = k)$$

usw. Im Falle einer Markow-Kette erster Ordnung gilt speziell $P_2 = P_1 \cdot P_1$; das trifft aber für Ordnungen größer 1 nicht mehr zu.

3 Simulation von Texten

Für die Details der Umsetzung siehe die Beschreibung des Programms in R, welches unter wwwu.uni-klu.ac.at/agebhard/mcksim/ abrufbar ist. Die Studierenden hatten sich unter anderem mit den Problemen auseinanderzusetzen, die sich daraus ergeben, dass die Arrays für die Übergangswahrscheinlichkeiten einerseits eine sehr große Dimension, andererseits aber fast nur Nullen als Einträge haben. Viele willkürliche Buchstabenfolgen haben in einem sinnvollen Text keinen Platz. Nach dem Laden der Library `mksim` werden Übergangsmatrizen der Ordnung 1 durch folgenden Befehl geschätzt:

```
> P1 <- estletter("ababcabcdadb")
> P1
          a    b           c     d
a 0.0000000 0.75   0.0000000  0.25
b 0.3333333 0.00   0.6666667  0.00
c 0.5000000 0.00   0.0000000  0.50
d 0.5000000 0.50   0.0000000  0.00
```

Bei Übergangsmatrizen höherer Ordnung wird die Ordnung als weiterer Parameter angefügt:

```
> P2 <- estletter ("ababcabcdadb", 2).
```

Die Simulation eines Texts von 30 Zeichen wird durch den Befehl `simulletter(30, P1, P2)` ausgeführt, wenn zuvor auf P1 und P2 die geschätzten Übergangswahrscheinlichkeiten erster und zweiter Ordnung schon gespeichert sind.

Folgende Beispiele illustrieren die Methode und ihren relativen Erfolg. Als Beispiel der Textsimulation dient Goethes Faust, als Textdatei ist er z. B. wie andere Klassiker über das Projekt Gutenberg verfügbar und umfasst ca. 10kB an Zeichen. Zunächst ein paar Zeilen simulierter Fausttext basierend auf einer Markow-Kette erster Ordnung:

> dennn nd imedaunder iläur desch ndild gehre we ufach t s in laber weneickeirtagerkas h be zwer datibenendih eran maler wen zürickl

Mit einer Kette der Ordnung 3 kommt das Resultat der deutschen Sprache schon etwas näher:

> lauf so gewonnen such bedarung es in scholde nüten irreibt sich in und segen kann mitwelt zur das ele gage person mir hin der jugen was an

Erhöht man die Ordnung auf 5, könnte man das Ergebnis fast schon in einer Lesung präsentieren?

> wer läßt das beste nicht minder wirklichkeiten gib ungebändigt jener sauberhauch so ein soll der offenbart vom lesen was plagt ihr einen wehen was in tiefe schmerz herauf geh hin und man sprich mild und jedes element nach jener satt vom lesen da und eh man sich dem schüttern und immer dankbar seid ein freventliche reihe beigestalt muß euch glühn.

4 Anwendungen

Texte und Literatur

Der amerikanische Poet Jeff Harrison experimentiert mit Markow-Algorithmen, um seine Texte zufällig zu erzeugen. Ausschnitte seiner Werke kann man unter Harrison (o. J.) nachlesen. Solche Sprachexperimente gehen bis auf den Dadaismus im frühen 20. Jh. zurück. Ernst Jandl ist ein jüngeres Beispiel aus der Region; er war allerdings stolz darauf, dass er die Texte seiner inneren Kreativität zu verdanken hat.

Eine weitere Anwendung von Markow-Ketten liegt im Erkennen der Autorschaft anhand von Übergangsmatrizen bestimmter Ordnung. Dazu muss man auch noch um Standards bei der Definition von Übergängen ringen. So werden etwa Namen weggelassen. Entsprechend muss die Datenspeicherung normiert werden (nicht alles ist in ASCII gespeichert).

Internet

In der virtuellen Welt der Internets gibt es viele Varianten, die Identität zu „wahren“. Oder auch, eine andere vorzuspielen und damit Reaktionen anderer zu provozieren: Etwa in Chatrooms automatischen Text erzeugen und warten, was passiert. Dazu werden auch Markow-Algorithmen verwendet, siehe etwa Hutchens (1997), der allerdings das Ziel weiter steckt und intelligente Konversation erzeugen will.

Markow-Algorithmen werden auch verwendet, um Platzhalter- oder Spam-Webseiten zu betreiben. Spezielle Algorithmen, die solche Internetseiten erkennen sollen, versagen zum Teil; ja selbst für das flüchtige Auge eines menschlichen Beobachters ist eine solche Fälschung nicht immer auf Anhieb zu erkennen. Auf der anderen Seite trainieren Programmierer ihre Spam-Abwehr u. a. mit Markow-Algorithmen, damit diese sich interaktiv verbessert.

Im einfachsten Fall ist Ranking von Internetseiten das Ergebnis der Simulation eines Nutzers, der sich zufällig entsprechend der vorhandenen Links weiter„hantelt“. Statt mit Verzweigungswahrscheinlichkeiten symbolisiert man die Graphen oft mit Besuchswahrscheinlichkeiten. Die Bewertung einer Seite durch PageRank kann man unter Gaijin (o. J.) selbst sehen. Mehr zur Verbindung des Ranking mit Markow-Ketten findet man in Demleitner (o. J.)

Monte-Carlo Integration

Ein unerwartetes Anwendungsgebiet des Zufalls liegt in der Bestimmung nicht geschlossen lösbarer bestimmter Integrale, und auch dort spielen Markow-Ketten letztendlich wieder eine Rolle. Die Bestim-

mung solcher Integrale wird zunächst quasi in Umkehr des Prinzips der geometrischen Wahrscheinlichkeit als "hit-and-miss"-Verfahren eingeführt: Man werfe zufällig gleichverteilt über einen begrenzten Ausschnitt des Graphen der zu integrierenden Funktion Punkte. Der Anteil der Punkte, die „unterhalb" des Graphen liegen, bestimmt das gesuchte Integral als eben diesen Teil der (üblicherweise rechteckigen) Fläche, die mit gleichverteilten Punkten „bestreut" wurde, vlg. Jones, et al. (2009). Während dieses Verfahren im eindimensionalen nicht wirklich mit numerischer Integration konkurrieren kann, wendet sich dies zugunsten dieser sogenannten Monte-Carlo Integration in höheren Dimensionen.

Nachdem also auf diese Weise Integrale mit Zufallszahlen bestimmbar sind, geht man nun weiter mit der Idee spezielle Integrale, nämlich Erwartungswerte, ebenfalls durch das einfache Mitteln der dazu passenden Zufallszahlen zu bestimmen. Man benötigt also einen Zufallsgenerator für eine bestimmte Dichtefunktion $f(x)$, so dass man mit den danach ausgewürfelten Zufallszahlen das Integral

$$E(X) = \int x \cdot f(x) dx$$

einfach als Mittelwert bestimmen kann. Hier kommen Markow-Ketten erster Ordnung ins Spiel. Man nutzt eine ihrer speziellen Eigenschaften aus, nämlich die Existenz einer Grenzverteilung, die gerade dem gewünschten $f(x)$ entsprechen muss. Die Grenzverteilung lässt sich aus den Potenzen der Übergangsmatrix bestimmen, vgl. Suess und Trumbo (2010). Diese Verfahren werden unter dem Begriff Markov-Chain-Monte-Carlo (MCMC) zusammengefasst. In der Bayesschen Statistik werden von den so bestimmten Zufallszahlen nicht nur das Mittel, sondern auch Streumaße oder mittels Kernschätzern die gesamte Verteilung weiter genutzt.

5 Unsere Praktikanten

Abb. 1: Unsere Praktikanten bei der Arbeit

Wir zitieren gerne aus dem Resümee unserer Praktikanten Michael Eichholzer und Johannes Steinbacher (2. bzw. 3. Klasse Oberstufe):

> „Da man sich das Themengebiet nach eigenem Interesse aussuchen konnte, war, anders als im Schulunterricht, für mehr Motivation und Laune beim Arbeiten gesorgt. […] Obwohl die Anforderungen höher waren als in der Schule, beschäftigte man sich doch gerne mit der Materie, da man genau das lernte, was einen auch wirklich interessiert."

Literatur

Borovcnik, M. (2001): Strengthening the Role of Probability Within Statistics Curricula. In: C. Batanero, G. Burrill, C. Reading (Hsg.), *Teaching Statistics in School. Mathematics-Challenges for Teaching and Teacher Education.* New York und Berlin: Springer, S. 71–83.

Christmann, N. (o. J.): Tonhöhenanalyse und Komponieren mittels Markov-Ketten. *Zufall und Musik.* http://optimierung.mathematik.uni-kl.de/~nchrist/MAMUSI/4Zufall_und_musik.htm (Zugriff: 1.7.2011).

Demleitner, M. (o. J.): *Einführung in die Statistische Sprachverarbeitung.* www.cl.uni-heidelberg.de/kurs/skripte/stat/html/page027.html (Zugriff: 1.7.2011).

Gaijin (o. J.): *Google Page Rank.* www.gaijin.at/olsgprank.php (Zugriff: 1.7.2011)

Harrison, J. (o. J.): *Postmortem Series und Accuracy.* www.moriapoetry.com/harrison.html (Zugriff: 1.7.2011).

Hutchens, J. L. (1997): *How to Pass the Turing Test by Cheating.* www.agent.ai/doc/upload/200403/hutc97_1.pdf (Zugriff: 1.7.2011).

Jones, O., Maillardet, R., Robinson, A. (2009): *Introduction to Scientific Programming and Simulation Using R.* London: Chapman & Hall.

R Development Core Team (2011): *R: A Language and Environment for Statistical Computing.* www.R-project.org (Zugriff: 1.7.2011).

Riehl, G. (2011): Warten auf einen Run – und was kommt dann? In: *Stochastik in der Schule* 31(3), S. 16–21.

Spiegel online (o. J.): *Projekt Gutenberg.* http://gutenberg.spiegel.de (Zugriff: 1.7.2011).

Suess, E. A., Trumbo, B. E. (2010): *Introduction to Probability Simulation and Gibbs Sampling with R.* New York und Berlin: Springer.

Viertl, R. (1997): *Einführung in die Stochastik.* New York und Berlin: Springer.

Anschrift der Verfasser

Manfred Borovcnik und Albrecht Gebhardt
Institut für Statistik
Universität Klagenfurt
9020 Klagenfurt
manfred.borovcnik@uni-klu.ac.at
albrecht.gebhardt@uni-klu.ac.at