Experimente mit MATLAB

Von Cleve Moler

Experiments with MATLAB® ist ein kostenloses Online-Buch für Lehrer und Schüler höherer Klassenstufen, die über den Schulstoff hinaus gehende Materialien suchen. Die vorgestellten Übungen und Materialien sind aber auch für Studienanfänger sicher eine Bereicherung.

MATLAB ist heute eine umfassend ausgestattete Sprache für technische Berechnungen. In den späten 70ern hat es jedoch als ziemlich einfaches „Matrix Laboratory" begonnen. *Experiments with MATLAB* knüpft an diese Labor-Tradition des Tüftelns an und stellt Experimente in angewandter Mathematik sowie mit technischen Berechnungen vor. Kurze Code-Abschnitte und kleine Programme, von denen viele interaktive Grafiken enthalten, geben dem Leser eine kleine Einführung in die Programmierung mit MATLAB.

Um die im Buch gezeigten Experimente vollständig durchführenzu können, sollte man in etwa über Oberstufen-Kenntnisse in Geometrie, Algebra und Trigonometrie verfügen. Daneben beschäftigt sich *Experiments with MATLAB* mit Themen, für die man ein wenig über Analysis, Matrizen und Differentialgleichungen wissen sollte – es setzt aber keine abgeschlossenen Kurse in diesen Fächern voraus.

In diesem *Cleve's Corner* stelle ich einige bearbeitete Auszüge aus *Experiments with MATLAB* vor, die Ihnen eine Vorstellung vom Niveau und Tonfall des Buches vermitteln sollen. Ich möchte Sie ermutigen, die Experimente nicht nur durchzulesen, sondern sie zu verändern und zu verbessern.

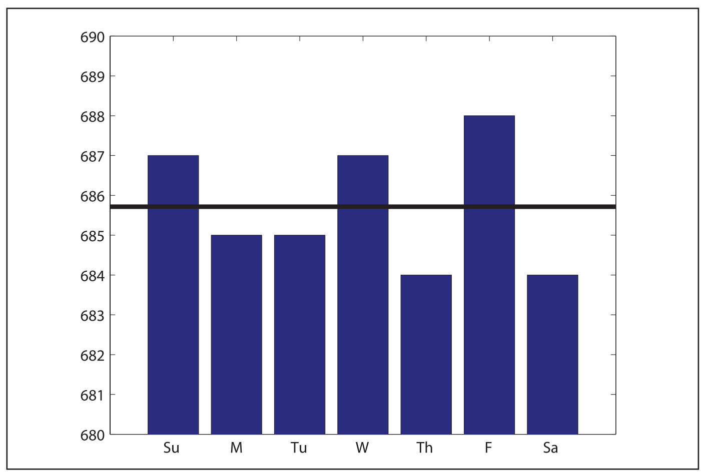

Abb. 1. *Dieses MATLAB-Diagramm zeigt, dass der 13. Tag eines Monats mit größerer Wahrscheinlichkeit ein Freitag ist als ein anderer Wochentag.*

Iterationen

Im ersten Kapitel werden Sie aufgefordert, sich eine Zahl auszusuchen – irgendeine Zahl – und diese in in MATLAB einzugeben:

```
x = Ihre Zahl
```

Geben Sie nun die Anweisung

```
x = sqrt(1+x)
```

ein und führen sie mithilfe der Pfeil-nach-oben- und Enter-Taste immer wieder aus, bis sich die angezeigte Zahl nicht mehr verändert.

Hier die ersten und letzten Zeilen für einen Anfangswert von x = 3:

```
3.000000000000000
2.000000000000000
1.732050807568877
1.652891650281070
1.628769980777233
      . . .
1.618033988749915
1.618033988749901
1.618033988749897
1.618033988749895
1.618033988749895
```

Ganz gleich, mit welchem Wert man beginnt: Die Reihe konvergiert immer zum selben Endwert von 1.6180339... . Erkennen Sie diese Zahl?

Dieses Experiment regt außerdem dazu an, über die beiden Bedeutungen des Gleichheitszeichens nachzudenken: Als Zuweisungsoperator in Programmiersprachen und als Symbol für Gleichheit in Gleichungen.

Wie würden Sie folgende Gleichung lösen?

$$x = \sqrt{1 + x}$$

Finger weg vom Computer! Lösen Sie diese im Kopf oder schriftlich. Quadrieren Sie beide Seiten, bringen Sie alles auf eine Seite des Gleichheitszeichens. Sie erhalten die quadratische Gleichung:

$$x^2 - x - 1 = 0$$

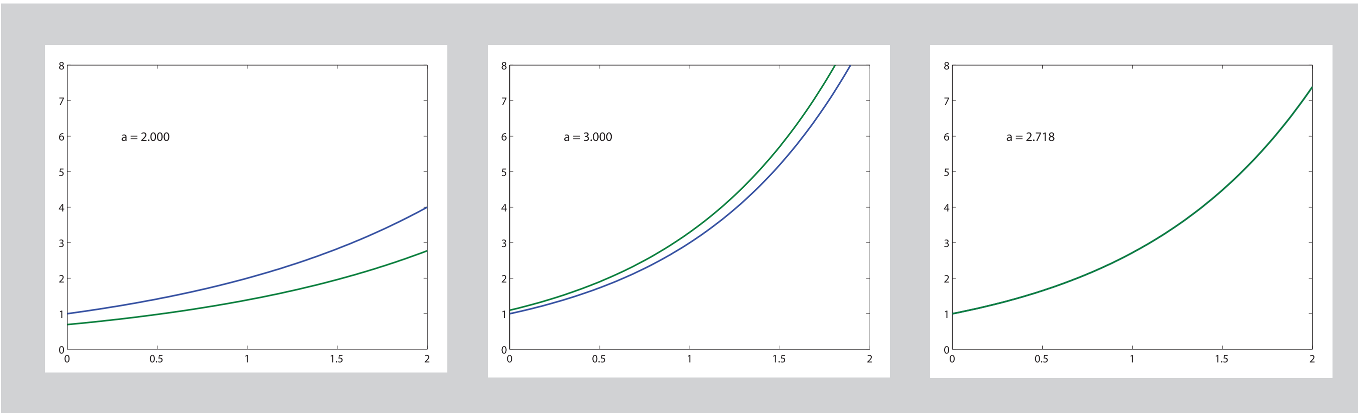

Abb. 2. *Durch Gegenüberstellung der Funktion $y = a^x$ und deren genäherter Ableitung kann man bei sukzessiver Variation der Basis die Euler-Zahl e entdecken.*

Mithilfe der quadratischen Ergänzung finden Sie deren positive Wurzel:

$$\varphi = (1 + \sqrt{5})/2 = 1.6180339 \ldots$$

Die Lösung ist wieder unser alter Freund, der Goldene Schnitt.

Kalender und Uhren

Viele Menschen halten Freitag den 13. für einen Unglückstag – aber wie oft kommt er eigentlich vor? Wie hoch ist die Wahrscheinlichkeit, dass der 13. Tag eines Monats auf einen Freitag fällt? Die spontane Antwort ist 1/7, doch das stimmt nicht ganz. Durch die Schaltregeln des Gregorianischen Kalenders wiederholen sich nämlich exakt alle 400 Jahre oder 4800 Monate die gleichen Wochenabläufe.

Ein einfaches Experiment mit den `datenum`- und `weekday`-Funktionen aus MATLAB kann abzählen, wie oft der 13. Tag eines Monats auf einen Freitag fällt (Abb. 1). Dabei zeigt sich, dass der 13. mit höherer Wahrscheinlichkeit auf einen Freitag fällt als auf irgendeinen anderen Wochentag. Die Wahrscheinlichkeit ist 688/4800 = 0,143333 und damit etwas größer als 1/7 = 0,142857.

Die Exponentialfunktion

Viele Schüler, die gerade einen Anfängerkurs in Analysis absolviert haben, denken leider, dass die Ableitung von e^x gleich xe^{x-1} ist. Wie kann man also die Zahl *e* und die Funktion e^x besser verstehen? MATLAB kann Potenzen berechnen, etwa $y = a^x$ für einen Skalar *a* und einen Vektor *x*.

```
a = 2
x = 0:0.01:2
y = a.^x
```

Man kann damit außerdem ohne jede formale Regel oder Differenzierung ungefähre Steigungen und damit genäherte Ableitungen berechnen, etwa mit:

```
h = 0.0001
yp = (a.^(x+h) - a.^h)/h
```

Die Anweisung

```
plot(x,[y; yp])
```

erzeugt das erste in Abbildung 2 gezeigte Diagramm. Die blaue Kurve ist der Graph von 2^x. Die grüne Kurve ist die genäherte Ableitung. Der Graph der genäherten Ableitung hat die gleiche Form wie der Graph der Ursprungsfunktion, liegt aber darunter. Tatsächlich ist das Verhältnis `yp./y` eine von *x* unabhängige Konstante.

Mit dem M-File `expqui` kann man nun die blaue Linie mit der Maus verschieben und so die Basis von a^x variieren. Das zweite Diagramm zeigt den Graphen für 3^x und seine genäherte Ableitung. Die grüne Kurve liegt nun oberhalb der blauen. Auf dem Weg von $a = 2$ nach $a = 3$ durchläuft man die im dritten Diagramm gezeigte Situation, in der die grüne und blaue Kurve deckungsgleich sind. Der hierbei für *a* angezeigte Wert ist gleich *e* und die Funktion ist e^x, eine der wichtigsten mathematischen Funktionen.

Eine wichtige aber doch subtile mathematische und rechentechnische Frage haben wir

Experiments with MATLAB

Inhalt

- Iterationen
- Fibonacci-Zahlen
- Kalender und Uhren
- T-Puzzle
- Matrizen
- Fraktal-Farn
- Magische Quadrate
- Magie mit TicTacToe
- Spiel des Lebens
- Mandelbrot-Menge
- Lineare Gleichungen
- Google PageRank™
- Gewöhnliche Differentialgleichungen
- Exponential-Funktion
- Räuber und Beute
- Flachwasser-Gleichungen

Experiments with MATLAB steht zum kostenlosen Download unter *www.mathworks.de/nn8/moler*

entwickelt sich in diskreten, als Generationen bezeichneten Zeitschritten weiter. Das Schicksal jeder einzelnen Zelle wird bei jedem Schritt durch die Vitalität ihrer acht nächsten Nachbarn eindeutig bestimmt. Die Regel dazu lautet: Eine lebendige Zelle mit zwei lebendigen Nachbarn oder aber jede beliebige Zelle mit drei lebendigen Nachbarn ist im nächsten Schritt lebendig.

Diese trügerisch einfache Regel erzeugt eine unglaubliche Vielfalt an Mustern, Rätseln und ungelösten mathematischen Problemen.

Das in *Experiments with MATLAB* vorgestellte MATLAB-Programm Game of Life ist ein sehr schönes Beispiel für die Verwendung *dünnbesetzter Matrizen*-Strukturen. Sein Universum ist nämlich eine dünnbesetzte Matrix X mit einer endlichen Zahl von Einsen, die die lebendigen Zellen markieren. Die Größe dieser Matrix passt sich durch Erweiterung jedem beliebigen Bevölkerungswachstum an.

Die Anweisungen

```
n = size(X,1);
p = [n 1:n-1];
q = [2:n 1];
Y = X(:,p)+X(:,q)+X(p,:)+X(q,:)+...
    X(p,p)+X(q,q)+X(p,q)+X(q,p);
```

erzeugen eine zweite dünnbesetzte Matrix Y mit Elementen der Werte 0 bis 8, die die lebendigen Nachbarn zählen. Die Grundregel des Spiels des Lebens lässt sich nun in Form einer einzigen weiteren MATLAB-Anweisung festhalten:

```
X = (X & (Y == 2)) | (Y == 3);
```

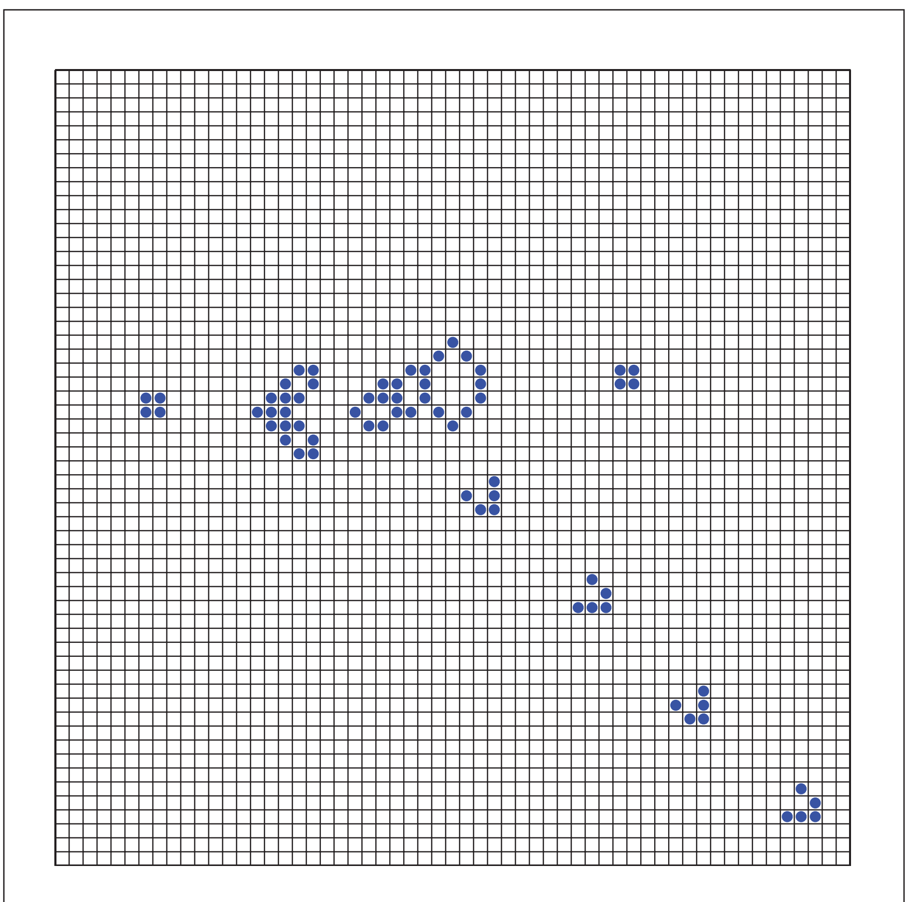

Abb. 6. *Bill Gospers Gleiter-Kanone.*

Da jede beliebige Population dieser Regel gehorcht, wird der Ausgang des Spiels eindeutig durch die Gestalt seiner Anfangspopulation bestimmt. In Bill Gospers als *Gleiterkanone* bekannter Anfangspopulation (Abb. 6) oszilliert der Zentralteil der Kanone und stößt einen unendlichen Strom von Gleitern aus, die das Sichtfeld verlassen und sich im leeren Raum verlieren. Gospers Konfiguration war die erste, die eine unbegrenzte Population erzeugen konnte.

Weitere Experimente

Sich die Übungen in *Experiments with MATLAB* nur durchzulesen macht nicht halb so viel Spaß – und ist nicht halb so lehrreich – wie sie selbst durchzuspielen. Wenn Sie das Gezeigte also interessant finden, oder wenn Sie Schüler oder Studenten kennen, die daran interessiert sein könnten, werfen Sie einen Blick auf *Experiments with MATLAB*. Und dann machen Sie die Experimente selber, diskutieren Sie darüber oder verbessern Sie diese. Programme, und besonders MATLAB-Programme, sind Vehikel zur Kommunikation mit anderen Menschen und nicht nur dazu da, um Befehle an eine Maschine zu übermitteln. ▪

Quellen

EXPERIMENTS WITH MATLAB
www.mathworks.de/nn8/moler

CLEVE'S CORNER COLLECTION
www.mathworks.de/nn8/clevescorner

91621v00 10/08