IN DIESEM KAPITEL

- Die Vorteile von R entdecken
- Einige Programmierkonzepte kennenlernen, die für R charakteristisch sind

Kapitel 1
R im Überblick

Mit geschätzt weltweit mehr als zwei Millionen Anwendern hat sich die Sprache R seit ihren Ursprüngen als Lehr- und Übungssprache in den 1990er Jahren schnell verbreitet.

Manche Leute würden behaupten, dass R weit mehr ist als eine statistische Programmiersprache – das sehen wir genauso. R ist auch

- ✔ ein sehr leistungsstarkes Werkzeug für alle Arten von Datenverarbeitung und -bearbeitung.
- ✔ eine Gemeinde von Programmierern, akademischen Anwendern sowie Anwendern aus der Praxis.
- ✔ ein Werkzeug, das alle möglichen Grafiken und Visualisierungen von Daten in Publikationsqualität erzeugt.
- ✔ eine riesige Sammlung von zusätzlichen Paketen.
- ✔ ein ganzer Werkzeugkoffer mit unglaublicher Vielseitigkeit.

In diesem Kapitel informieren wir Sie über die Vorteile von R sowie seine einzigartigen Eigenschaften und Eigenarten.

Sie können R von der Site `www.r-project.org` herunterladen. Auf der Website erhalten Sie auch weitere Information über R sowie Links zu Onlinehandbüchern, Mailinglisten, Konferenzen und weiteren Publikationen.

Auf den Spuren der Geschichte von R

Ross Ihaka und Robert Gentleman entwickelten R als freie Software für ihre Lehrveranstaltungen während ihrer Zeit als Kollegen an der Universität von Auckland in Neuseeland. Sie beide kannten S, eine kommerzielle Sprache für Statistik, und so verwendeten sie ähnliche Syntax für ihr Projekt. Nachdem Ihaka und Gentleman ihre Software auf der S-News-Mailingliste angekündigt hatten, interessierten sich zunehmend Leute für das Projekt und arbeiteten mit den Initiatoren zusammen, insbesondere Martin Mächler.

Aktuell haben 21 Personen das Recht, das zentrale Archiv des Quellcodes von R zu verändern. Diese bilden das *R Development Core Team*. Darüber hinaus haben viele weitere Personen neuen Code sowie Fehlerbehebungen zum Projekt beigesteuert.

Hier ein paar Meilensteine in der Entwicklung von R:

- ✔ **Frühe 1990er:** Die Entwicklung von R begann.
- ✔ **August 1993:** Die Software wurde in der S-Mailingliste angekündigt. Seitdem wurden einige aktive R-Mailinglisten ins Leben gerufen. Die Webseite `www.r-project.org/mail.html` enthält Beschreibungen der einzelnen Listen und Anleitungen, wie man sie abonniert. (Mehr Informationen hierzu finden Sie im Abschnitt »Eine engagierte Nutzergemeinde« weiter hinten in diesem Kapitel.)
- ✔ **Juni 1995:** Mithilfe einiger überzeugender Argumente von Martin Mächler und einigen anderen gelang es, den Code als »Freie Software« unter der *GNU General Public License (GPL) Version 2* der *Free Software Foundation* zur Verfügung zu stellen.
- ✔ **Mitte 1997:** Das erste *R Development Core Team* wurde gebildet (damals noch unter dem Namen *core group*).
- ✔ **Februar 2000:** Die erste Version von R, Version 1.0.0, wurde veröffentlicht.
- ✔ **Oktober 2004:** Veröffentlichung der Version 2.0.0 von R.
- ✔ **April 2013:** Veröffentlichung der Version 3.0.0 von R.
- ✔ **Juni 2017:** Veröffentlichung der Version 3.4.1 von R (die in diesem Buch verwendete Version).

Ross Ihaka hat einen umfassenden Überblick über die Entwicklung von R verfasst. Dieser kann unter `http://cran.r-project.org/doc/html/interface98-paper/paper.html` eingesehen werden.

Die Vorteile der Anwendung von R erkennen

Von den vielen attraktiven Vorteilen von R sind einige besonders erwähnenswert: Es wird aktiv weiterentwickelt, es hat gute Schnittstellen zu den verschiedensten Datenformaten und anderen Systemen und es ist äußerst flexibel, sodass es in sehr unterschiedlichen Bereichen eingesetzt werden kann. Und das Allerbeste: Es ist kostenlos – mit allen Vorteilen.

Kostenloser, frei zugänglicher Quellcode

R ist unter einer Open-Source-Lizenz zugänglich, das heißt, jeder kann den Quellcode herunterladen und verändern. Häufig wird das als »frei wie Sprache« bezeichnet (man kann damit machen, was man will). R ist zudem kostenlos erhältlich, also »frei wie Freibier« (zwar kostenlos, aber möglicherweise mit geheimem Rezept und verbunden mit gewissen Einschränkungen hinsichtlich der Verwendung). Kurz, Sie können R kostenlos herunterladen und verwenden.

Ein weiterer Vorteil, obgleich etwas weniger offensichtlich, liegt in der Tatsache, dass jeder den Quellcode einsehen, verändern und verbessern kann. Auf diese Weise haben viele exzellente Programmierer Verbesserungen und Fehlerbehebungen am Quellcode vorgenommen. Aus diesem Grund läuft R sehr stabil und zuverlässig.

Jede Freiheit ist mit Einschränkungen verbunden. Im Fall von R wird dies in der *GNU General Public License (GPL), Version 2* geregelt. Der vollständige Wortlaut der Lizenzbestimmungen kann unter `www.r-project.org/COPYING` eingesehen werden. Beachten Sie, dass die Bestimmungen nur die Weitergabe von Code betreffen. Die Nutzung ist davon unberührt. In Kurzform sagt die Lizenz: Wenn Sie Code verändern oder weitergeben, müssen Sie diesen für alle (frei) zugänglich machen.

Läuft überall

Das *R Development Core Team* hat einigen Aufwand betrieben, um die Software auf verschiedenen Typen von Hardware und Software lauffähig zu machen. Dies bedeutet, R läuft auf Windows, Unix (auch Linux) und dem Mac.

Unterstützt Erweiterungen

R selbst ist eine leistungsstarke Sprache, die viele verschiedene Funktionen wie Datenbearbeitung, statistische Modellierung und Grafik umfasst. Ein wirklich großer Vorteil ist jedoch seine Erweiterbarkeit. Entwickler können problemlos eigene Software schreiben und als Erweiterungspaket veröffentlichen. Da es vergleichsweise einfach ist, solche Erweiterungen zu schreiben, existieren wirklich Tausende von Paketen. In der Tat werden heute viele neue (und nicht so neue) statistische Methoden zusammen mit einem R-Paket veröffentlicht.

Eine engagierte Nutzergemeinde

Die Anzahl der R-Anwender wächst beständig. Viele Anwender unterstützen Neulinge bei den ersten Schritten oder setzen sich für die Verwendung von R in ihrem Arbeitsbereich und Kollegenkreis ein. Manchmal werden sie auch aktiv

- ✔ in R-Mailinglisten (`www.r-project.org/mail.html`).
- ✔ Foren, wie
  - *Stack Overflow*, einer Seite für R-Programmierer (`www.stackoverflow.com/questions/tagged/r`)
  - *CrossValidated*, einer Seite für Statistiker (`http://stats.stackexchange.com/questions/tagged/r`).

Zusätzlich zu diesen Mailinglisten und Foren gibt es R-Anwender, die

- ✔ aktive Blogger sind (`www.r-bloggers.com`),
- ✔ sich in sozialen Netzwerken wie Twitter (`www.twitter.com/search/rstats`) engagieren,
- ✔ und auf regionalen und internationalen Konferenzen zu finden sind.

Für weitere Informationen siehe auch Kapitel 11.

Schnittstellen zu anderen Sprachen

Nachdem mehr und mehr Menschen begannen, für ihre Analysen auf R umzusteigen, versuchten sie, R mit ihren alten Prozessen zu kombinieren. Dies führte zu einer riesigen Auswahl von Paketen, die R mit Dateisystemen, Datenbanken und anderen Anwendungen verbinden. Viele dieser Pakete sind mit der Zeit in die Basisinstallation von R aufgenommen worden und stehen nach dem Download gleich zur Verfügung.

Das Paket `foreign` (`https://cran.r-project.org/web/packages/foreign/index.html`) ermöglicht zum Beispiel den lesenden Zugriff auf Dateien, die von Statistikpaketen wie SPSS, SAS, Stata und anderen stammen (siehe Kapitel 12).

Für die Anbindung an Datenbanken stehen mehrere Pakete zur Verfügung, beispielsweise

- ✔ das `RODBC`-Paket für Datenbanken, die das *Open Database Connectivity Protocol* (ODBC) verwenden (`https://cran.r-project.org/web/packages/RODBC/index.html`), oder
- ✔ das `ROracle`-Paket für Oracle-Datenbanken (`https://cran.r-project.org/web/packages/ROracle/index.html`).

Zu Beginn wurde R im Wesentlichen in Fortran und C geschrieben. Daher konnte Code in diesen beiden Sprachen problemlos aus R heraus aufgerufen werden. Mit der Zeit kamen immer mehr Sprachen wie C++, Java, Python und weitere hinzu, die auf einfache Weise aus R heraus aufgerufen werden können.

Da es immer mehr R-Anwender gab, konnten die Entwickler kommerzieller Softwarelösungen R nicht mehr einfach so ignorieren. Deshalb haben heute viele der großen kommerziellen Softwarepakete Add-ons zur Anbindung an R. Dies betrifft besonders die SPSS-Software (IBM) wie auch SAS (SAS Institute). In beiden Fällen gibt es Schnittstellen, um Daten und Grafiken zwischen R und der jeweiligen Statistiksoftware hin- und herzubewegen.
Auch andere Entwickler haben zur besseren Verknüpfbarkeit unterschiedlicher Datenanalyse- und Statistiksoftware beigetragen. Beispielsweise hat Statconn `RExcel` entwickelt, eine Excel-Schnittstelle, die es Anwendern erlaubt, mit R innerhalb von Excel zu arbeiten (`http://www.statconn.com/products.html`).

Einige bemerkenswerte Eigenschaften von R

R ist mehr als eine Programmiersprache für den Statistikbereich. Es hat einige einzigartige Eigenschaften, die es sehr leistungsstark machen. Dazu gehört das vektorwertige Konzept, das Berechnungen mit vielen Werten auf einmal ermöglicht.

Berechnungen mit Vektoren durchführen

R ist eine vektorbasierte Sprache. Stellen Sie sich einen Vektor als Zeile oder Spalte mit Zahlen oder Text vor. Die Liste der Zahlen `{1,2,3,4,5}` könnte beispielsweise einen Vektor darstellen. Im Gegensatz zu vielen anderen Programmen ermöglicht Ihnen R, Funktionen auf den ganzen Vektor gleichzeitig anzuwenden, ohne dass Sie eine Schleife programmieren müssen.

Lassen Sie uns das mit richtigem R-Code illustrieren. Zunächst weisen wir die Werte `1:5` einem Vektor zu, den wir `x` nennen:

```
> x <- 1:5
> x
[1] 1 2 3 4 5
```

Anschließend addieren wir zu jedem Element des Vektors `x` den Wert `2` und geben das Ergebnis aus:

```
> x + 2
[1] 3 4 5 6 7
```

Sie können auch zwei Vektoren addieren. Um die Werte `6:10` elementweise zu `x` zu addieren, geben Sie ein:

```
> x + 6:10
[1]  7  9 11 13 15
```

In den meisten anderen Programmiersprachen würden diese Operationen eine explizite Schleife erfordern, die die Addition elementweise durchführt.

Diese Eigenschaft ist äußerst hilfreich, da sie Ihnen ermöglicht, viele Operationen in einem einzigen Schritt auszuführen. In anderen Sprachen, die nicht vektororientiert sind, müssten Sie Schleifen programmieren, um dasselbe zu erreichen.

Wir stellen das Konzept der Vektoren in Kapitel 2 vor und vertiefen Vektoren und Vektorisierung gründlich in Kapitel 4.

Mehr als nur statistische Berechnungen

R wurde von Statistikern entwickelt, um statische Berechnungen zu vereinfachen. Dieses Erbe besteht fort: R ist immer noch ein sehr leistungsstarkes Werkzeug, um praktisch jede statistische Berechnung durchzuführen.

Je mehr sich R über seine Ursprünge in Statistik hinaus entwickelte, zog es immer mehr Programmierer als reine Statistiker an. Aus diesem Grund ist R sehr geeignet für eine Reihe nicht statistischer Aufgaben. Dazu gehören Datenbearbeitung, grafische Visualisierung und Analysen aller Art. Aktuell wird R in den Bereichen Finanzmathematik, Sprachverarbeitung, Genetik, Biologie und Marktforschung verwendet, um nur einige zu nennen.

R ist *Turing-vollständig*. Dies bedeutet, Sie können damit alles programmieren, was Sie wollen. (Das wird allerdings nicht immer einfach sein.)
Für dieses Buch nehmen wir an, dass Sie die Programmierung mit R erlernen wollen, und nicht Statistik. Dennoch enthält Teil IV eine Einführung in Statistik.

Code ohne Compiler ausführen

R ist eine interpretierte Sprache, Sie benötigen also – im Gegensatz zu kompilierten Sprachen wie Java oder C – keinen Compiler, der aus Ihrem Code erst ein ausführbares Programm erstellt, bevor Sie es verwenden können. R interpretiert den von Ihnen vorgegebenen Code und wandelt ihn in Aufrufe vorkompilierter Funktionen um.

In der Praxis bedeutet dies, dass Sie einfach Ihren Code schreiben und an R senden, wo er direkt ausgeführt wird. Dies vereinfacht den Entwicklungszyklus enorm. Diese Bequemlichkeit gibt es jedoch nicht ganz umsonst. Die Ausführung des Codes dauert etwas länger. Interpretierte Sprachen sind meist langsamer als kompilierte.

Wenn Sie bereits Erfahrung mit anderen Sprachen haben, führen Sie sich bitte immer wieder vor Augen, dass R *nicht* C oder Java ist. Obwohl Sie R wie eine prozedurale Sprache – wie C – oder wie eine objektorientierte Sprache – wie Java – verwenden können, entspricht R dem Paradigma der funktionalen Programmierung. Wie Sie später in diesem Buch, insbesondere in Teil III, sehen werden, erfordert dieses Paradigma eine veränderte Sichtweise. Vergessen Sie, was Sie über andere Sprachen wissen, und machen Sie sich auf etwas komplett Neues gefasst!