
Inhaltsverzeichnis

1	Einleitung	1
1.1	Die Geschichte	3
1.2	Warum R?	5
1.3	Überblick	6
2	Grundlagen	9
2.1	R als Taschenrechner	9
2.2	Zuweisungen	11
2.3	Objekte	13
2.4	Hilfe	14
2.4.1	Das integrierte Hilfesystem	15
2.4.2	Handbücher und weiterführende Literatur	16
2.4.3	Mailinglisten	19
2.5	Eine Beispielsitzung	21
2.6	<i>Workspace</i> – der Arbeitsplatz	25
2.7	Logik und fehlende Werte	26
2.8	Datentypen	31
2.9	Datenstrukturen und deren Behandlung	33
2.9.1	Vektoren	34
2.9.2	Matrizen	38
2.9.3	Arrays	42
2.9.4	Listen	42
2.9.5	Datensätze – <i>data frames</i>	44
2.9.6	Objekte für formale S4 Klassen	48
2.10	Konstrukte	49
2.10.1	Bedingte Anweisungen	49
2.10.2	Schleifen	52
2.11	Zeichenketten	55
2.12	Datum und Zeit	58

3	Ein- und Ausgabe von Daten	61
3.1	ASCII – Dateien	61
3.2	Binärdateien	63
3.3	R Objekte lesen und schreiben	64
3.4	Spezielle Datenformate	66
3.5	Zugriff auf Datenbanken	67
3.6	Zugriff auf Excel-Daten	69
4	Die Sprache im Detail	71
4.1	Funktionen	71
4.1.1	Funktionsaufruf	72
4.1.2	Eigene Funktionen definieren	73
4.2	Verzögerte Auswertung – <i>Lazy Evaluation</i>	76
4.3	Umgebungen und deren Regeln – <i>Environments</i> und <i>Scoping Rules</i>	78
4.4	Umgang mit Fehlern	86
4.4.1	Finden und Beseitigen von Fehlern – <i>Debugging</i>	86
4.4.2	Fehlerbehandlung	90
4.5	Rekursion	92
4.6	Umgang mit Sprachobjekten	93
4.7	Vergleich von Objekten	96
5	Effizientes Programmieren	99
5.1	Programmierstil	101
5.2	Vektorwertiges Programmieren und Schleifen	104
5.2.1	Sinnvolles Benutzen von Schleifen	104
5.2.2	Vektorwertiges Programmieren – mit <code>apply()</code> und <code>Co</code>	106
5.3	Hilfsmittel zur Effizienzanalyse	111
5.3.1	Laufzeitanalyse – <i>Profiling</i>	115
6	Objektorientiertes Programmieren	117
6.1	OOP mit S3-Methoden und -Klassen	118
6.2	OOP mit S4-Methoden und -Klassen	122
6.2.1	Beispiel: Eine Klasse <i>Wave</i> und Methoden	124
7	Statistik mit R	131
7.1	Grundlegende Funktionen	132
7.2	Zufallszahlen	135
7.3	Verteilungen und Stichproben	136
7.4	Modelle und Formelnotation	138
7.5	Lineare Modelle	139
7.6	Überblick: Weitere spezielle Verfahren	147

8	Grafik	153
8.1	Konventionelle Grafik	154
8.1.1	Ausgabe von Grafik – <i>Devices</i>	154
8.1.2	<i>High-level</i> Grafik	156
8.1.3	Konfigurierbarkeit – <code>par()</code>	161
8.1.4	<i>Low-level</i> Grafik	167
8.1.5	Mathematische Beschriftung	169
8.1.6	Eigene Grafikfunktionen definieren	171
8.2	Trellis Grafiken mit lattice	173
8.2.1	Unterschiede zu konventioneller Grafik	173
8.2.2	Das Paket grid – mehr als nur Grundlage für lattice ..	175
8.2.3	Ausgabe von Trellis Grafiken – <code>trellis.device()</code>	176
8.2.4	Formelinterface	178
8.2.5	Konfiguration und Erweiterbarkeit	180
8.3	Dynamische und interaktive Grafik	182
9	Erweiterungen	185
9.1	Einbinden von Quellcode: C, C++, Fortran	185
9.2	Integration	188
9.3	Der Batch Betrieb	190
9.4	Aufruf des Betriebssystems	190
10	Pakete	193
10.1	Warum Pakete?	194
10.2	Paketübersicht	194
10.3	Verwaltung und Installation von Paketen	196
10.3.1	Libraries	197
10.3.2	Source- und Binärpakete	198
10.4	Struktur von Paketen	203
10.5	Funktionen und Daten in Paketen	205
10.6	Namespaces	206
10.7	Dokumentation	207
10.7.1	Das Rd Format	208
10.7.2	SWeave	209
 Anhang		
A	R installieren, konfigurieren und benutzen	211
A.1	R herunterladen und installieren	211
A.2	R konfigurieren	214

B Editoren für R	219
B.1 Der Emacs mit ESS	219
B.2 Tinn-R	220
B.3 WinEdt mit RWinEdt	220
C Grafische Benutzeroberflächen (GUI) für R	223
C.1 Der R Commander	224
C.2 Windows GUI	224
D Tabelle englischer und deutscher Begriffe	227
Literaturverzeichnis	229
Tabellenverzeichnis	237
Index	239