

# Corpus des Deutschen Bundesrechts (C-DBR)

COMPILATION REPORT

Version 2021-07-30

License MIT-0

DOI: [10.5281/zenodo.5133962](https://doi.org/10.5281/zenodo.5133962)

<b>Titel</b>	Source Code des »Corpus des Deutschen Bundesrechts«
<b>Abkürzung</b>	C-DBR-Source
<b>Autor</b>	Seán Fobbe
<b>Version</b>	2021-07-30
<b>Download</b>	<a href="https://doi.org/10.5281/zenodo.5133962">https://doi.org/10.5281/zenodo.5133962</a>
<b>Lizenz</b>	MIT No Attribution (MIT-0)

### Zitiervorschlag

*Seán Fobbe* (2021). Source Code des »Corpus des Deutschen Bundesrechts« (C-DBR-Source). Version 2021-07-30. Zenodo. DOI: 10.5281/zenodo.5133962.

### Digital Object Identifier (DOI): Concept DOI und Version DOI

Soweit nicht anders angegeben ist die DOI immer eine »Version DOI« und bezieht sich nur auf eine bestimmte Version der Software. Sie verlinkt daher nur Version 2021-07-30. Für das Gesamtkonzept der Software steht eine »Concept DOI« zur Verfügung, die auf der Zenodo-Seite jeder Version unter »Cite all versions?« zu finden ist. Sie lautet 10.5281/zenodo.4072934. Die »Concept DOI« verlinkt immer die aktuellste Version.

### Lizenz: MIT No Attribution (MIT-0)

Copyright — 2021— Seán Fobbe

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the »Software«), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so.

THE SOFTWARE IS PROVIDED »AS IS«, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

### Disclaimer

Dieser Datensatz ist eine private wissenschaftliche Initiative und steht in keiner Verbindung zu Behörden, Gerichten oder anderen amtlichen Stellen der Bundesrepublik Deutschland.

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>9</b>
1.1	Überblick . . . . .	9
1.2	Endprodukte . . . . .	9
1.3	Kompilierung . . . . .	10
1.3.1	Datensatz . . . . .	10
1.3.2	Codebook . . . . .	10
1.4	Systemanforderungen . . . . .	11
<b>2</b>	<b>Parameter</b>	<b>12</b>
2.1	Name des Datensatzes . . . . .	12
2.2	DOI des Datensatz-Konzeptes . . . . .	12
2.3	DOI der konkreten Version . . . . .	12
2.4	Lizenz . . . . .	12
2.5	Verzeichnis für Analyse-Ergebnisse . . . . .	12
2.6	Optionen: Quanteda . . . . .	12
2.7	Optionen: Knitr . . . . .	12
2.7.1	Ausgabe-Formate . . . . .	12
2.7.2	DPI für Raster-Grafiken . . . . .	13
2.7.3	Ausrichtung von Grafiken im Compilation Report . . . . .	13
2.8	Frequenztabellen: Liste zu prüfender Variablen . . . . .	13
<b>3</b>	<b>Vorbereitung</b>	<b>14</b>
3.1	Datumsstempel . . . . .	14
3.2	Datum und Uhrzeit (Beginn) . . . . .	14
3.3	Ordner für Analyse-Ergebnisse erstellen . . . . .	14
3.4	Packages Laden . . . . .	14
3.5	Zusätzliche Funktionen einlesen . . . . .	16
3.6	Quanteda-Optionen setzen . . . . .	16
3.7	Knitr Optionen setzen . . . . .	16
3.8	Vollzitate statistischer Software . . . . .	17
3.9	Parallelisierung aktivieren . . . . .	17
3.9.1	Anzahl logischer Kerne bestimmen . . . . .	17
3.9.2	Quanteda . . . . .	17
3.9.3	Data.table . . . . .	17
<b>4</b>	<b>Funktionen definieren</b>	<b>18</b>
4.1	Transformation von Gliederungs-Metadaten . . . . .	18
4.2	NA in leere Listen-Elemente einsetzen . . . . .	18
4.3	Erstellen von Titel- und Bezeichnungshierarchien . . . . .	18
<b>5</b>	<b>Download vorbereiten</b>	<b>20</b>
5.1	XML-Inhaltsverzeichnis einlesen . . . . .	20
5.2	Links zu XML-Dateien aus XML-Inhaltsverzeichnis extrahieren . . . . .	20
5.3	Links zu HTML Landing Pages generieren . . . . .	20
5.4	Funktion anzeigen: f.linkextract . . . . .	20
5.5	Links aus HTML Landing Pages extrahieren . . . . .	21
5.6	Dateinamen von PDF und EPUB-Dateien in separate Vektoren sortieren . .	21
5.7	Vektor der Langtitel erstellen . . . . .	21

5.7.1	Namen bereinigen und kürzen . . . . .	21
5.7.2	Indizes der AEG bestimmen . . . . .	21
5.7.3	AEGs umbenennen . . . . .	21
5.8	Vektor der Kurztitel erstellen . . . . .	22
5.9	Vektoren der Titel vereinigen . . . . .	22
5.10	Prüfung auf Namens-Kollisionen . . . . .	22
5.11	Bereinigung von Namens-Kollisionen . . . . .	22
5.12	Dateierweiterungen hinzufügen . . . . .	23
5.13	Links zu EPUB-Dateien erstellen . . . . .	23
5.14	Links zu PDF-Dateien erstellen . . . . .	23
5.15	Data Table für Download vorbereiten . . . . .	23
5.16	Abkürzungsverzeichnis erstellen . . . . .	23
5.17	Download Table als CSV speichern . . . . .	24
5.18	Verzeichnis aller Rechtsakte als CSV speichern . . . . .	24
5.19	Anzahl herunterzuladender Dateien . . . . .	24
5.19.1	Pro Format . . . . .	24
5.19.2	Insgesamt . . . . .	24
<b>6</b>	<b>Verarbeitung der DTD und XML-Dateien mit Anlagen</b>	<b>25</b>
6.1	Document Type Definition herunterladen . . . . .	25
6.2	Download der XML-Dateien . . . . .	25
6.3	Download-Ergebnis . . . . .	25
6.3.1	Anzahl herunterzuladender Dateien . . . . .	25
6.3.2	Anzahl heruntergeladener Dateien . . . . .	25
6.3.3	Fehlbetrag . . . . .	25
6.3.4	Fehlende Dateien . . . . .	26
6.4	Extrahieren der XML-Dateien und ihrer Anlagen . . . . .	26
6.5	XML Dateien auflisten und Dateigrößen speichern . . . . .	26
6.6	Zu extrahierende Metadaten-Variablen definieren . . . . .	26
6.7	Korpus erstellen: Einzelnormen . . . . .	27
6.7.1	Beginn XML Parsing . . . . .	27
6.7.2	Fork Cluster starten . . . . .	27
6.7.3	XML Parsen . . . . .	27
6.7.4	Cluster beenden . . . . .	30
6.7.5	Liste in Data Table umwandeln . . . . .	30
6.7.6	Ende XML Parsing . . . . .	31
6.7.7	Dauer XML Parsing . . . . .	31
6.7.8	Variable “doc_id” erstellen . . . . .	31
6.7.9	Variablen-Name für Ausfertigungsdatum anpassen . . . . .	31
6.7.10	Variable “fundstellentyp” anpassen . . . . .	31
6.7.11	Variable “builddate_iso” erstellen . . . . .	31
6.7.12	Variable “aenderung_datum” erstellen . . . . .	32
6.7.13	Variable “aufhebung_verkuendung_datum” erstellen . . . . .	32
6.7.14	Variable “aufhebung_wirkung_datum” erstellen . . . . .	32
6.7.15	Variable “neufassung_datum” erstellen . . . . .	32
6.7.16	Variable “ausfertigung_jahr” hinzufügen . . . . .	32
6.7.17	Variable “doi_concept” hinzufügen . . . . .	33
6.7.18	Variable “doi_version” hinzufügen . . . . .	33
6.7.19	Variable “version” hinzufügen . . . . .	33

6.7.20	Variable "lizenz" hinzufügen . . . . .	33
6.8	Stichprobe für Qualitätsprüfung ziehen . . . . .	33
6.9	Korpus erstellen: Rechtsakte . . . . .	34
6.9.1	Variablen definieren . . . . .	34
6.9.2	Vollständiger Satz an Variablen . . . . .	34
6.9.3	Einzelnormen zu Rechtsakten vereinigen . . . . .	34
6.9.4	Variable "dateiname" in "doc_id" umbenennen . . . . .	35
6.10	Datensatz erstellen: XML-Metadaten . . . . .	36
6.10.1	Beginn XML Parsing . . . . .	36
6.10.2	Fork Cluster starten . . . . .	36
6.10.3	XML Parsen . . . . .	36
6.10.4	Fork Cluster beenden . . . . .	37
6.10.5	Liste in Data Table umwandeln . . . . .	37
6.10.6	Ende XML Parsing . . . . .	38
6.10.7	Dauer XML Parsing . . . . .	38
6.10.8	Variablen-Name für Ausfertigungsdatum anpassen . . . . .	38
6.10.9	Variable "fundstellentyp" anpassen . . . . .	38
6.10.10	Variable "builddate_iso" erstellen . . . . .	38
6.10.11	Variable "aenderung_datum" erstellen . . . . .	38
6.10.12	Variable "aufhebung_verkuendung_datum" erstellen . . . . .	39
6.10.13	Variable "aufhebung_wirkung_datum" erstellen . . . . .	39
6.10.14	Variable "neufassung_datum" erstellen . . . . .	39
6.10.15	Variable "ausfertigung_jahr" hinzufügen . . . . .	39
6.10.16	Variable "doi_concept" hinzufügen . . . . .	39
6.10.17	Variable "doi_version" hinzufügen . . . . .	40
6.10.18	Variable "version" hinzufügen . . . . .	40
6.10.19	Variable "lizenz" hinzufügen . . . . .	40
6.11	Netzwerk-Analyse (experimentell) . . . . .	40
6.11.1	Funktion definieren: f.kennzahlen.search . . . . .	40
6.11.2	Funktion definieren: f.kennzahlen.collapse . . . . .	40
6.11.3	Funktion definieren: f.kennzahlen.edgelist . . . . .	41
6.11.4	Funktion definieren: f.network.analysis . . . . .	42
6.11.5	Netzwerk-Analyse durchführen . . . . .	44
6.11.6	XML-Dateien bei denen Fehler auftreten . . . . .	45
6.12	Wiederverpacken der XML-Dateien . . . . .	45
6.12.1	XML-Dateien definieren . . . . .	45
6.12.2	XML-Dateien verpacken . . . . .	45
6.12.3	Anhänge zu XML-Dateien verpacken . . . . .	45
<b>7</b>	<b>Frequenztabellen erstellen: Einzelnormen</b>	<b>46</b>
7.1	Funktion anzeigen: f.fast.freqtable . . . . .	46
7.2	Liste zu prüfender Variablen . . . . .	47
7.3	Frequenztabellen erstellen . . . . .	47
<b>8</b>	<b>Frequenztabellen erstellen: Rechtsakte</b>	<b>107</b>
8.1	Variablen ignorieren . . . . .	107
8.2	Liste zu prüfender Variablen . . . . .	107
8.3	Frequenztabellen erstellen . . . . .	107
<b>9</b>	<b>Frequenztabellen erstellen: XML-Metadaten</b>	<b>116</b>

9.1	Liste zu prüfender Variablen . . . . .	116
9.2	Frequenztabellen erstellen . . . . .	116
<b>10</b>	<b>Frequenztabellen visualisieren</b>	<b>126</b>
10.1	Präfixe erstellen . . . . .	126
10.2	Tabellen für Einzelnormen einlesen . . . . .	126
10.3	Tabellen für Rechtsakte einlesen . . . . .	126
10.4	Tabellen für XML-Metadaten einlesen . . . . .	126
10.5	Periodikum . . . . .	127
10.5.1	Einzelnormen . . . . .	127
10.5.2	Rechtsakte . . . . .	129
10.5.3	XML-Metadaten . . . . .	131
10.6	Ausfertigungsjahr . . . . .	133
10.6.1	Einzelnormen . . . . .	133
10.6.2	Rechtsakte . . . . .	135
10.6.3	XML-Metadaten . . . . .	136
<b>11</b>	<b>Korpus-Analytik</b>	<b>138</b>
11.1	Berechnung linguistischer Kennwerte . . . . .	138
11.1.1	Funktion anzeigen: f.summarize.iterator . . . . .	138
11.1.2	Berechnung durchführen . . . . .	140
11.2	Variablen-Namen anpassen . . . . .	140
11.2.1	Einzelnormen . . . . .	140
11.2.2	Rechtsakte . . . . .	141
11.3	Kennwerte den Korpora hinzufügen . . . . .	141
11.3.1	Einzelnormen . . . . .	141
11.3.2	Rechtsakte . . . . .	141
11.4	Varianten mit Metadaten erstellen . . . . .	141
11.4.1	Einzelnormen . . . . .	141
11.4.2	Rechtsakte . . . . .	141
11.5	Linguistische Kennwerte: Einzelnormen . . . . .	142
11.5.1	Zusammenfassungen berechnen . . . . .	142
11.5.2	Zusammenfassungen anzeigen . . . . .	144
11.5.3	Zusammenfassungen speichern . . . . .	144
11.6	Linguistische Kennwerte: Rechtsakte . . . . .	145
11.6.1	Zusammenfassungen berechnen . . . . .	145
11.6.2	Zusammenfassungen anzeigen . . . . .	147
11.6.3	Zusammenfassungen speichern . . . . .	147
11.7	Verteilungen . . . . .	148
11.7.1	Density (Zeichen) . . . . .	148
11.7.2	Density (Tokens) . . . . .	151
11.7.3	Density (Typen) . . . . .	153
11.7.4	Density (Sätze) . . . . .	155
11.8	Quantitative Variablen . . . . .	157
11.8.1	Ausfertigungsdatum . . . . .	157
11.8.2	Ausfertigungsjahr . . . . .	157
<b>12</b>	<b>Kontrolle der Variablen</b>	<b>159</b>
12.1	Semantische Sortierung der Variablen . . . . .	159
12.1.1	Variablen sortieren: Einzelnormen . . . . .	159

12.1.2 Variablen sortieren: Rechtsakte . . . . .	161
12.1.3 Variablen sortieren: XML-Metadaten . . . . .	163
12.2 Anzahl Variablen der Datensätze . . . . .	163
12.3 Alle Variablen-Namen der Datensätze . . . . .	164
<b>13 CSV-Dateien erstellen</b>	<b>167</b>
13.1 Einzelnormen (Korpus) . . . . .	167
13.1.1 Name für CSV definieren . . . . .	167
13.1.2 Datensatz speichern . . . . .	167
13.2 Einzelnormen (Metadaten) . . . . .	167
13.2.1 Name für CSV definieren . . . . .	167
13.2.2 Datensatz speichern . . . . .	167
13.3 Rechtsakte (Korpus) . . . . .	167
13.3.1 Name für CSV definieren . . . . .	167
13.3.2 Datensatz speichern . . . . .	168
13.4 Rechtsakte (Metadaten) . . . . .	168
13.4.1 Name für CSV definieren . . . . .	168
13.4.2 Datensatz speichern . . . . .	168
13.5 XML-Metadaten . . . . .	168
13.5.1 Name für CSV definieren . . . . .	168
13.5.2 Datensatz speichern . . . . .	168
<b>14 Download der PDF-Dateien</b>	<b>169</b>
14.1 Download durchführen . . . . .	169
14.2 Download-Ergebnis . . . . .	169
14.2.1 Anzahl herunterzuladender Dateien . . . . .	169
14.2.2 Anzahl heruntergeladener Dateien . . . . .	169
14.2.3 Fehlbetrag . . . . .	169
14.2.4 Fehlende Dateien . . . . .	169
<b>15 TXT-Dateien erstellen</b>	<b>170</b>
15.1 Anzahl zu extrahierender Dateien . . . . .	170
15.2 Funktion anzeigen: f.dopar.pagenums . . . . .	170
15.3 Anzahl zu extrahierender Seiten . . . . .	171
15.4 Funktion anzeigen: f.dopar.pdfextract . . . . .	171
15.5 Text Extrahieren . . . . .	172
<b>16 Download der EPUB-Dateien</b>	<b>173</b>
16.1 Download durchführen . . . . .	173
16.2 Download-Ergebnis . . . . .	173
16.2.1 Anzahl herunterzuladender Dateien . . . . .	173
16.2.2 Anzahl heruntergeladener Dateien . . . . .	173
16.2.3 Fehlbetrag . . . . .	173
16.2.4 Fehlende Dateien . . . . .	173
<b>17 Dateigrößen analysieren</b>	<b>174</b>
17.1 Gesamtgröße . . . . .	174
17.1.1 PDF-Dateien (MB) . . . . .	174
17.1.2 EPUB-Dateien (MB) . . . . .	174
17.1.3 XML-Dateien (MB) . . . . .	174

17.1.4	TXT-Dateien (MB)	174
17.1.5	Korpus-Objekte in RAM (MB)	175
17.2	Verteilung der Dateigrößen (PDF)	176
17.3	Verteilung der Dateigrößen (EPUB)	178
17.4	Verteilung der Dateigrößen (XML)	180
17.5	Verteilung der Dateigrößen (TXT)	182
<b>18</b>	<b>ZIP-Archive erstellen</b>	<b>184</b>
18.1	Verpacken der CSV-Dateien	184
18.2	Verpacken der PDF-Dateien	184
18.3	Verpacken der TXT-Dateien	184
18.4	Verpacken der EPUB-Dateien	185
18.5	Verpacken der Netzwerk-Dateien	185
18.6	Verpacken der Analyse-Dateien	185
18.7	Verpacken der Source-Dateien	185
<b>19</b>	<b>Kryptographische Hashes</b>	<b>187</b>
19.1	Liste der ZIP-Archive erstellen	187
19.2	Funktion anzeigen: f.dopar.multihashes	187
19.3	Hashes berechnen	188
19.4	In Data Table umwandeln	188
19.5	Index hinzufügen	188
19.6	Hashes in CSV-Datei speichern	188
19.7	Leerzeichen hinzufügen um Zeilenumbruch zu ermöglichen	189
19.8	In Bericht anzeigen	190
<b>20</b>	<b>Abschluss</b>	<b>193</b>
20.1	Datumsstempel	193
20.2	Datum und Uhrzeit (Anfang)	193
20.3	Datum und Uhrzeit (Ende)	193
20.4	Laufzeit des gesamten Skripts	193
20.5	Warnungen	193
<b>21</b>	<b>Parameter für strenge Replikationen</b>	<b>194</b>
	<b>Literaturverzeichnis</b>	<b>196</b>



# 1 Einleitung

## 1.1 Überblick

Dieses Skript wertet das amtliche Internetangebot “Gesetze im Internet” (<https://www.gesetze-im-internet.de>) der Bundesrepublik Deutschland vollständig aus und kompiliert es in einen reichhaltigen menschen- und maschinenlesbaren Korpus. Es ist die Grundlage des **Corpus des Deutschen Bundesrechts (C-DBR)**.

Alle mit diesem Skript erstellten Datensätze werden dauerhaft kostenlos und urheberrechtsfrei auf Zenodo, dem wissenschaftlichen Archiv des CERN, veröffentlicht. Alle Versionen sind mit einem persistenten Digital Object Identifier (DOI) versehen. Die neueste Version des Datensatzes ist immer über den Link der Concept DOI erreichbar: <https://doi.org/10.5281/zenodo.3832111>

## 1.2 Endprodukte

Primäre Endprodukte des Skripts sind folgende ZIP-Archive:

1. Der volle Datensatz im CSV-Format, unterteilt in Einzelnormen; nur Rechtsakte mit veröffentlichtem Normtext sind erfasst
2. Die Metadaten aller Einzelnormen im CSV-Format (wie 1, nur ohne Normtexte)
3. Der volle Datensatz im CSV-Format, unterteilt in Rechtsakte; nur Rechtsakte mit veröffentlichtem Normtext sind erfasst
4. Die Metadaten aller Rechtsakte im CSV-Format (wie 3, nur ohne Normtexte)
5. Die Metadaten aller auf »Gesetze im Internet« als XML veröffentlichten Rechtsakte, im CSV-Format, unabhängig davon ob sie Normtext enthalten oder nicht
6. Der volle Datensatz im XML-Format, unterteilt in Rechtsakte; Grundlage für die CSV-Varianten
7. Alle Anlagen zu den XML-Dateien im jeweiligen Original-Format
8. Alle Rechtstexte im TXT-Format, unterteilt in Rechtsakte (deutlich reduzierter Umfang an Metadaten)
9. Alle Rechtstexte im PDF-Format, unterteilt in Rechtsakte (deutlich reduzierter Umfang an Metadaten)
10. Alle Rechtstexte im EPUB-Format, unterteilt in Rechtsakte (deutlich reduzierter Umfang an Metadaten)
11. Alle Analyse-Ergebnisse (Tabellen als CSV, Grafiken als PDF und PNG)
12. Netzwerk-Strukturen (Adjazenzmatrizen, Edgelists, GraphML, und Netzwerk-Diagramme) für alle Rechtsakte (experimentell!)

Zusätzlich werden für alle ZIP-Archive kryptographische Signaturen (SHA2-256 und SHA3-512) berechnet und in einer CSV-Datei hinterlegt. Die Analyse-Ergebnisse werden zum Ende hin nicht gelöscht, damit sie für die Codebook-Erstellung verwendet werden können. Weiterhin kann optional ein PDF-Bericht erstellt werden (siehe unter “Kompilierung”).

## 1.3 Kompilierung

Mit der Funktion `render()` von **rmarkdown** können der **vollständige Datensatz** und das **Codebook** kompiliert und die Skripte mitsamt ihrer Rechenergebnisse in ein gut lesbares PDF-Format überführt werden.

Alle Kommentare sind im roxygen2-Stil gehalten. Die beiden Skripte können daher auch **ohne** `render()` regulär als R-Skripte ausgeführt werden. Es wird in diesem Fall kein PDF-Bericht erstellt und Diagramme werden nicht abgespeichert.

### 1.3.1 Datensatz

Um den **vollständigen Datensatz** zu kompilieren und einen PDF-Bericht zu erstellen, kopieren Sie bitte alle im Source-Archiv bereitgestellten Dateien in einen leeren Ordner und führen mit R diesen Befehl aus:

```
rmarkdown::render(input = "C-DBR_Source_CorpusCreation.R",
                  output_file = paste0("C-DBR_",
                                       Sys.Date(),
                                       "_CompilationReport.pdf"),
                  envir = new.env())
```

### 1.3.2 Codebook

Um das **Codebook** zu kompilieren und einen PDF-Bericht zu erstellen, führen Sie bitte im Anschluss an die Kompilierung des Datensatzes (!) untenstehenden Befehl mit R aus.

Bei der Prüfung der GPG-Signatur wird ein Fehler auftreten und im Codebook dokumentiert, weil die Daten nicht mit meiner Original-Signatur versehen sind. Dieser Fehler hat jedoch keine Auswirkungen auf die Funktionalität und hindert die Kompilierung nicht.

```
rmarkdown::render(input = "C-DBR_Source_CodebookCreation.R",
                  output_file = paste0("C-DBR_",
                                       Sys.Date(),
                                       "_Codebook.pdf"),
                  envir = new.env())
```

## 1.4 Systemanforderungen

Das Skript in seiner veröffentlichten Form kann nur unter Linux ausgeführt werden, da es Linux-spezifische Optimierungen (z.B. Fork Cluster) und Shell-Kommandos (z.B. OpenSSL) nutzt. Das Skript wurde unter Fedora Linux entwickelt und getestet. Die zur Kompilierung benutzte Version entnehmen Sie bitte dem **sessionInfo()**-Ausdruck am Ende dieses Berichts.

In der Standard-Einstellung wird das Skript vollautomatisch die maximale Anzahl an Rechenkernen/Threads auf dem System zu nutzen. Wenn die Anzahl Threads (Variable “fullCores”) auf 1 gesetzt wird, ist die Parallelisierung deaktiviert.

Auf der Festplatte sollten 8 GB Speicherplatz vorhanden sein.

Um die PDF-Berichte kompilieren zu können benötigen Sie das R package **rmarkdown**, eine vollständige Installation von  $\text{\LaTeX}$  und alle in der Präambel-TEX-Datei angegebenen  $\text{\LaTeX}$  Packages.

## 2 Parameter

### 2.1 Name des Datensatzes

```
datasetname <- "C-DBR"
```

### 2.2 DOI des Datensatz-Konzeptes

```
doi.concept <- "10.5281/zenodo.3832111" # checked
```

### 2.3 DOI der konkreten Version

```
doi.version <- "10.5281/zenodo.5133932"
```

### 2.4 Lizenz

```
license <- "Creative Commons Zero 1.0 Universal"
```

### 2.5 Verzeichnis für Analyse-Ergebnisse

Muss mit einem Schrägstrich enden!

```
outputdir <- paste0(getwd(),  
                    "/ANALYSE/")
```

### 2.6 Optionen: Quanteda

```
tokens_locale <- "de_DE"
```

### 2.7 Optionen: Knitr

#### 2.7.1 Ausgabe-Formate

```
dev <- c("pdf",  
        "png")
```

### 2.7.2 DPI für Raster-Grafiken

```
dpi <- 300
```

### 2.7.3 Ausrichtung von Grafiken im Compilation Report

```
fig.align <- "center"
```

## 2.8 Frequenztabellen: Liste zu prüfender Variablen

**Hinweis:** Nur diese Variablen werden bei der Erstellung der Frequenztabellen berücksichtigt.

```
vars.freqtable <- c("periodikum",  
                    "fundstellentyp",  
                    "check_neuf",  
                    "check_aufh",  
                    "check_sonst",  
                    "check_hinweis",  
                    "check_stand",  
                    "gliederungskennzahl",  
                    "ausfertigung_jahr",  
                    "doi_concept",  
                    "doi_version",  
                    "version",  
                    "lizenz")
```

## 3 Vorbereitung

### 3.1 Datumsstempel

Dieser Datumsstempel wird in alle Dateinamen eingefügt. Er wird am Anfang des Skripts gesetzt, für den den Fall, dass die Laufzeit die Datumsbarriere durchbricht.

```
datestamp <- Sys.Date()
print(datestamp)
```

```
## [1] "2021-07-30"
```

### 3.2 Datum und Uhrzeit (Beginn)

```
begin.script <- Sys.time()
print(begin.script)
```

```
## [1] "2021-07-30 18:45:34 CEST"
```

### 3.3 Ordner für Analyse-Ergebnisse erstellen

```
dir.create(outputdir)

dir.create("Netzwerke")
dir.create("Netzwerke/Edgelisten")
dir.create("Netzwerke/Adjazenzmatrizen")
dir.create("Netzwerke/Netzwerkdiagramme")
dir.create("Netzwerke/GraphML")
```

### 3.4 Packages Laden

```
library(rvest)      # HTML/XML-Extraktion
library(xml2)       # Arbeit mit XML-Format
library(knitr)       # Professionelles Reporting
library(kableExtra)  # Verbesserte Kable Tabellen
library(pdftools)    # Extrahieren von PDF-Dateien
```

```
## Using poppler version 21.01.0
```

```
library(doParallel)    # Parallelisierung
```

```
## Loading required package: foreach
```

```
## Loading required package: iterators
```

```
## Loading required package: parallel
```

```
library(ggplot2)       # Fortgeschrittene Datenvisualisierung  
library(data.table)    # Fortgeschrittene Datenverarbeitung
```

```
## data.table 1.14.0 using 8 threads (see ?getDTthreads). Latest news: r-  
datatable.com
```

```
library(quanteda)      # Fortgeschrittene Computerlinguistik
```

```
## Package version: 3.0.0  
## Unicode version: 13.0  
## ICU version: 67.1
```

```
## Parallel computing: 16 of 16 threads used.
```

```
## See https://quanteda.io for tutorials and examples.
```

```
library(scales)        # Skalierung von Diagrammen  
library(openssl)       # Kryptographische Signaturen
```

```
## Linking to: OpenSSL 1.1.1k FIPS 25 Mar 2021 (FIPS)
```

```
library(igraph)        # Analyse von Graphen
```

```
##  
## Attaching package: 'igraph'
```

```
## The following objects are masked from 'package:stats':  
##  
##   decompose, spectrum
```

```
## The following object is masked from 'package:base':  
##  
##   union
```

```
library(ggraph)      # Analyse von Graphen  
library(qgraph)      # Analyse von Graphen
```

```
##  
## Attaching package: 'qgraph'
```

```
## The following object is masked from 'package:ggraph':  
##  
##   qgraph
```

### 3.5 Zusätzliche Funktionen einlesen

**Hinweis:** Die hieraus verwendeten Funktionen werden jeweils vor der ersten Benutzung in vollem Umfang angezeigt um den Lesefluss zu verbessern.

```
source("General_Source_Functions.R")
```

### 3.6 Quanteda-Optionen setzen

```
quanteda_options(tokens_locale = tokens_locale)
```

### 3.7 Knitr Optionen setzen

```
knitr::opts_chunk$set(fig.path = outputdir,  
                        dev = dev,  
                        dpi = dpi,  
                        fig.align = fig.align)
```



## 3.8 Vollzitate statistischer Software

```
knitr::write_bib(c(.packages()),  
                "packages.bib")
```

```
## tweaking foreach
```

## 3.9 Parallelisierung aktivieren

Parallelisierung wird zur Beschleunigung des XML-Parsings, der Konvertierung von PDF zu TXT und der Datenanalyse mittels **quanteda** und **data.table** verwendet. Die Anzahl threads wird automatisch auf das verfügbare Maximum des Systems gesetzt, kann aber auch nach Belieben auf das eigene System angepasst werden. Die Parallelisierung kann deaktiviert werden, indem die Variable **fullCores** auf 1 gesetzt wird.

Die hier verwendete Funktion **makeForkCluster()** ist viel schneller, funktioniert aber nur auf Unix-basierten Systemen (Linux, MacOS). Bei einer Ausführung unter Windows sollten Sie **makecluster()** verwenden.

### 3.9.1 Anzahl logischer Kerne bestimmen

```
fullCores <- detectCores()  
print(fullCores)
```

```
## [1] 16
```

### 3.9.2 Quanteda

```
quanteda_options(threads = fullCores)
```

### 3.9.3 Data.table

```
setDTthreads(threads = fullCores)
```

## 4 Funktionen definieren

### 4.1 Transformation von Gliederungs-Metadaten

Wird bei der Umwandlung der Metadaten aus dem XML-Format benötigt. Konkret werden hierdurch Werte die nur einmal pro Abschnitt (z.B. Gliederungsüberschriften) hochgerechnet, damit jede Norm die ihr zugehörigen Abschnitts-Metadaten zugewiesen erhält.

```
f.heading.transform <- function(inputvec){  
  
  rep.text <- c("NA", inputvec[is.na(inputvec) == FALSE])  
  
  which <- c(1, which(is.na(inputvec) == FALSE), length(inputvec) + 1)  
  
  rep.count <- diff(which)  
  
  rep <- data.table(rep.text,  
                   rep.count)  
  
  replist <- vector("list",  
                   rep[,.N])  
  
  for (i in 1:rep[,.N]){  
    replist[[i]]<- rep(rep.text[i],  
                      rep.count[i])  
  }  
  
  outvec <- unlist(replist)  
  return(outvec)  
}
```

### 4.2 NA in leere Listen-Elemente einsetzen

```
f.zero.NA <- function(x) if (length(x) == 0){  
  NA_character_  
}else{  
  paste(x, collapse = " ")}
```

### 4.3 Erstellen von Titel- und Bezeichnungshierarchien

Diese Funktion nimmt die für jedes Gesetz bereitgestellten Gliederungskennzahlen, bricht diese in ihre Bestandteile herunter und definiert für jede Gliederungskennzahl die volle Hierarchie an Titeln bzw. Gliederungsbezeichnungen.

Beispiel Titelhierarchie: Recht der Schuldverhältnisse | Einzelne Schuldverhältnisse | Mietvertrag, Pachtvertrag | Mietverhältnisse über Wohnraum | Beendigung des Mietverhältnisses | Werkwohnungen

```
f.namechain <- function(kennzahl,
                        titel,
                        bez){

  out.list <- vector("list", length(kennzahl))

  for (i in seq_along(kennzahl)){

    einzelzahl <- kennzahl[i]

    breaks <- seq_len(nchar(einzelzahl) / 3 ) * 3

    chain <- unname(mapply(substr, einzelzahl, 1, breaks))

    titelchain <- paste(titel[match(chain, kennzahl)], collapse = " | ")

    bezchain <- paste(bez[match(chain, kennzahl)], collapse = " | ")

    out.list[[i]] <- data.table(einzelzahl,
                                titelchain,
                                bezchain)

  }

  out.vec <- rbindlist(out.list)

  return(out.vec)

}
```

## 5 Download vorbereiten

### 5.1 XML-Inhaltsverzeichnis einlesen

```
URL <- "https://www.gesetze-im-internet.de/gii-toc.xml"

XML <- read_xml(URL)
```

### 5.2 Links zu XML-Dateien aus XML-Inhaltsverzeichnis extrahieren

```
links <- xml_nodes(XML,
                    "link")
```

```
## Warning: `xml_nodes()` was deprecated in rvest 1.0.0.
## Please use `html_elements()` instead.
```

```
links.xml <- xml_text(links)
```

### 5.3 Links zu HTML Landing Pages generieren

```
links.html <- gsub("/xml.zip",
                  "/index.html",
                  links.xml)
```

### 5.4 Funktion anzeigen: f.linkextract

```
print(f.linkextract)
```

```
## function(URL){
##   tryCatch({
##     read_html(URL) %>%
##       html_nodes("a")%>%
##       html_attr('href'),
##     error=function(cond) {
##       return(NA)}
##   )
## }
```

## 5.5 Links aus HTML Landing Pages extrahieren

```
links.list <- lapply(links.html,  
                     f.linkextract)  
  
links.raw <- unlist(links.list)
```

## 5.6 Dateinamen von PDF und EPUB-Dateien in separate Vektoren sortieren

```
filenames.pdf <- grep (".pdf$",  
                      links.raw,  
                      ignore.case = TRUE,  
                      value = TRUE)  
  
filenames.epub <- grep (".epub$",  
                       links.raw,  
                       ignore.case = TRUE,  
                       value = TRUE)
```

## 5.7 Vektor der Langtitel erstellen

**Hinweis:** Es gibt zwei Rechtsakte mit dem Namen “Allgemeine Eisenbahngesetz”, obwohl es sich um zwei unterschiedliche Rechtsakte handelt. Die beiden Rechtsakte werden daher um ihr jeweiliges Ausfertigungsjahr ergänzt um die Dateinamen einzigartig zu machen.

```
longtitle.raw <- xml_nodes(XML, "title") %>% xml_text()
```

### 5.7.1 Namen bereinigen und kürzen

```
longtitle <- gsub(" ", "", longtitle.raw)  
longtitle <- gsub("[:punct:]", "", longtitle)
```

### 5.7.2 Indizes der AEG bestimmen

```
AEGindex <- grep("AllgemeinesEisenbahngesetz", longtitle)
```

### 5.7.3 AEGs umbenennen

```
longtitle[AEGindex] <- c("AllgemeinesEisenbahngesetz1993",  
                        "AllgemeinesEisenbahngesetz1951")
```

## 5.8 Vektor der Kurztitel erstellen

```
shorttitle <- filenames.pdf

shorttitle <- gsub(".pdf",
                  "",
                  shorttitle)

shorttitle <- gsub("_",
                  "",
                  shorttitle)
```

## 5.9 Vektoren der Titel vereinigen

Die Kurz- und Langtitel werden zu einem Vektor zusammengefügt. Dieser wird dann auf maximal 200 Zeichen gekürzt, damit keine Probleme für Windows-User entstehen.

```
title <- paste(shorttitle,
               longtitle,
               sep="_")

title <- strtrim(title,
                 200)
```

## 5.10 Prüfung auf Namens-Kollisionen

Kollidierende Namen anzeigen. Wenn Namens-Kollisionen bestehen (wie oben beim AEG) müssen diese unbedingt bereinigt werden, weil ansonsten beim Herunterladen eine Datei alle anderen mit dem gleichen Namen überschreibt.

```
title[duplicated(title)]
```

```
## character(0)
```

## 5.11 Bereinigung von Namens-Kollisionen

Eine manuelle Bereinigung von Kollisionen ist bevorzugt. Falls keine manuelle Bereinigung stattgefunden hat wird in diesem Schritt eine automatische Bereinigung durchgeführt.

```
title <- make.unique(title,
                    sep = "-")
```

## 5.12 Dateierweiterungen hinzufügen

```
title.xml <- paste0(title, ".zip")
title.epub <- paste0(title, ".epub")
title.pdf <- paste0(title, ".pdf")
```

## 5.13 Links zu EPUB-Dateien erstellen

```
prelinks.epub <- gsub("xml.zip",
                     "",
                     links.xml)

links.epub <- paste0(prelinks.epub,
                    filenames.epub)
```

## 5.14 Links zu PDF-Dateien erstellen

```
prelinks.pdf <- gsub("xml.zip",
                    "",
                    links.xml)

links.pdf <- paste0(prelinks.pdf,
                   filenames.pdf)
```

## 5.15 Data Table für Download vorbereiten

```
download <- data.table(title.xml,
                      links.xml,
                      title.epub,
                      links.epub,
                      title.pdf,
                      links.pdf)
```

## 5.16 Abkürzungsverzeichnis erstellen

```
conctable <- data.table(shorttitle,
                       longtitle.raw)

colnames(conctable) <- c("Kurztitel",
                       "Langtitel")
```

## 5.17 Download Table als CSV speichern

```
fwrite(download,
       paste0(outputdir,
              datasetname,
              "_02_Links.csv"),
       na = "NA")
```

## 5.18 Verzeichnis aller Rechtsakte als CSV speichern

```
fwrite(conctable,
       paste0(datasetname,
              "_",
              datestamp,
              "_DE_AlleRechtsakteVerzeichnis.csv"),
       na = "NA")
```

## 5.19 Anzahl herunterzuladender Dateien

### 5.19.1 Pro Format

```
download[, .N]
```

```
## [1] 6630
```

### 5.19.2 Insgesamt

```
download[, .N] * 3
```

```
## [1] 19890
```



## 6 Verarbeitung der DTD und XML-Dateien mit Anlagen

### 6.1 Document Type Definition herunterladen

Die Document Type Definition (DTD) “definiert den Aufbau des XML-Formats zur Veröffentlichung der aktuellen Bundesgesetze und Rechtsverordnungen ueber [www.gesetze-im-internet.de](http://www.gesetze-im-internet.de)” (Zitat aus dem Inhalt der Datei).

```
download.file("http://www.gesetze-im-internet.de/dtd/1.01/gii-norm.dtd",
              paste0(datasetname,
                    "_",
                    datestamp,
                    "_DE_XML_DocumentTypeDefinition_v1-01.dtd"))
```

### 6.2 Download der XML-Dateien

```
mcmapply(download.file,
          download$links.xml,
          download$title.xml)
```

### 6.3 Download-Ergebnis

#### 6.3.1 Anzahl herunterzuladender Dateien

```
download[, .N]
```

```
## [1] 6630
```

#### 6.3.2 Anzahl heruntergeladener Dateien

```
files.zip <- list.files(pattern = "\\\\.zip")
length(files.zip)
```

```
## [1] 6630
```

#### 6.3.3 Fehlbetrag

```
N.missing <- download[, .N] - length(files.zip)
print(N.missing)
```

```
## [1] 0
```

### 6.3.4 Fehlende Dateien

```
missing <- setdiff(download$title.xml,  
                   files.zip)  
print(missing)
```

```
## character(0)
```

## 6.4 Extrahieren der XML-Dateien und ihrer Anlagen

XML-Dateien und ihre Anlagen sind einzeln nach Rechtsakten in ZIP-Archiven verpackt. Diese werden nun extrahiert und die ZIP-Archive im Anschluss gelöscht.

```
files.zip <- list.files(pattern = "\\\\.zip",  
                       ignore.case = TRUE)  
  
for (file in files.zip){  
  unzip(file)  
}  
  
unlink(files.zip)
```

## 6.5 XML Dateien auflisten und Dateigrößen speichern

```
files.xml <- list.files(pattern = "\\\\.xml",  
                      ignore.case = TRUE)  
  
xml.MB <- file.size(files.xml) / 10^6
```

## 6.6 Zu extrahierende Metadaten-Variablen definieren

```
varlist <- c("jurabk",  
            "amtabk",  
            "ausfertigung-datum",  
            "periodikum",  
            "zitstelle",  
            "langue",  
            "kurzue")
```

## 6.7 Korpus erstellen: Einzelnormen

**Wichtiger Hinweis:** Es werden für diese Variante nur Rechtsakte ausgewertet, bei denen mindestens eine Einzelnorm mit Text-Inhalt vorhanden ist!

Die XML-Daten enthalten keine Leerzeichen zwischen den XML-Tags, sowie zwischen den XML-Tags und ihrem Inhalt. Damit beim Entfernen der XML-Tags keine Inhalte zusammengefügt werden, wird die XML-Datei zunächst als Character-Vektor eingelesen, Leerzeichen hinzugefügt und im Anschluss erst die XML-Struktur eingelesen. Zwischen dem Anfang des Dokuments und dem ersten XML-Tag darf kein Leerzeichen sein, dieses wird einzeln nachkorrigiert. Zusätzlicher whitespace ist bei späterer Text-Verarbeitung unschädlich und wird im Rahmen der Tokenisierung praktisch immer entfernt.

Ohne diesen Schritt können Ergebnisse so aussehen: “Zollkodex,d)alle Verfahren”

### 6.7.1 Beginn XML Parsing

```
begin.parse <- Sys.time()
```

### 6.7.2 Fork Cluster starten

```
cl <- makeForkCluster(fullCores)
registerDoParallel(cl)
```

### 6.7.3 XML Parsen

```
limit <- length(files.xml)

out <- foreach(z = 1:limit, .errorhandling = 'pass') %dopar% {

  ## XML als Character-Vektor einlesen
  xml.char <- readChar(files.xml[z],
                        file.info(files.xml[z])$size)

  ## Leerzeichen einfügen
  xml.char <- gsub(">", "> ", xml.char)
  xml.char <- gsub("<", " <", xml.char)
  xml.char <- sub(" <", "<", xml.char)

  ## XML-Struktur lesen
  XML <- read_xml(xml.char)

  ## Schleife vorbereiten
  nodes <- xml_nodes(XML, xpath = "//norm")
  scope <- seq_along(nodes)

  ## Inhaltsdaten extrahieren
  text.temp <- vector("list", max(scope))
  enbez.temp <- vector("list", max(scope))
}
```

```

g.kennzahl.temp <- vector("list", max(scope))
g.bez.temp <- vector("list", max(scope))
g.titel.temp <- vector("list", max(scope))

for (i in scope){

  text.temp[[i]] <- xml_nodes(nodes[i],
                              xpath = "textdaten//text//Content") %>% xml_
text(trim = TRUE)

  enbez.temp[[i]] <- xml_nodes(nodes[i],
                              xpath = "metadaten//enbez") %>% xml_text(
trim = TRUE)

  g.kennzahl.temp[[i]] <- xml_nodes(nodes[i],
                              xpath = "metadaten//gliederungseinheit/
/gliederungskennzahl") %>% xml_text(trim = TRUE)

  g.bez.temp[[i]] <- xml_nodes(nodes[i],
                              xpath = "metadaten//gliederungseinheit//
gliederungsbez") %>% xml_text(trim = TRUE)

  g.titel.temp[[i]] <- xml_nodes(nodes[i],
                              xpath = "metadaten//gliederungseinheit//
gliederungstitel") %>% xml_text(trim = TRUE)

}

## Leere Elemente mit NA kennzeichnen
enbez <- sapply(enbez.temp, f.zero.NA)
text <- sapply(text.temp, f.zero.NA)
g.kennzahl.pos <- sapply(g.kennzahl.temp, f.zero.NA)
g.bez.pos <- sapply(g.bez.temp, f.zero.NA)
g.titel.pos <- sapply(g.titel.temp, f.zero.NA)

## Gliederungsinformationen transformieren
gliederungskennzahl <- f.heading.transform(g.kennzahl.pos)
gliederungsbez <- f.heading.transform(g.bez.pos)
gliederungstitel <- f.heading.transform(g.titel.pos)

## Grundlage für Ketten extrahieren
g.kennzahl.vec <- xml_nodes(XML, xpath = "//norm//gliederungskennzahl") %>%
xml_text(trim = TRUE)
g.bez.vec <- xml_nodes(XML, xpath = "//norm//gliederungsbez") %>% xml_text(
trim = TRUE)
g.titel.vec <- xml_nodes(XML, xpath = "//norm//gliederungstitel") %>% xml_
text(trim = TRUE)

## Ketten anhand von Gliederungskennzahlen erstellen
chain.dt <- f.namechain(g.kennzahl.vec,
                        g.titel.vec,
                        g.bez.vec)

## Ketten einfügen
titelkette <- chain.dt$titelchain[match(gliederungskennzahl,
chain.dt$einzelzahl)]

```

```

bezketten <- chain.dt$bezchain[match(gliederungskennzahl,
                                     chain.dt$einzelzahl)]

## Build Date extrahieren
builddate_original <- xml_attr(nodes, attr = "builddate")

## Content Data Table erstellen
content.out <- data.table(builddate_original,
                          gliederungskennzahl,
                          gliederungsbez,
                          bezkette,
                          gliederungstitel,
                          titelkette,
                          enbez,
                          text)

content.out <- content.out[text != ""]

## Allgemeine Metadaten extrahieren
meta <- vector("list", length(varlist))

for (i in 1:length(varlist)){
  temp <- xml_node(XML, varlist[i]) %>% xml_text(trim = TRUE)
  meta[[i]] <- rep(temp,
                  content.out[, .N])
}

setDT(meta)
setnames(meta, new = varlist)

meta$fundstellentyp <- rep(xml_node(XML, "fundstelle") %>% xml_attr(attr = "
typ"),
                          content.out[, .N])

meta$dateiname <- rep(files.xml[z],
                      content.out[, .N])

## Standangaben extrahieren
standtyp <- xml_nodes(XML, "standtyp") %>% xml_text(trim = TRUE)
standkommentar <- xml_nodes(XML, "standkommentar") %>% xml_text(trim = TRUE)
standcheck <- xml_nodes(XML, "standangabe") %>% xml_attr(attr = "checked")

dt.stand <- data.table(standtyp,
                      standkommentar,
                      standcheck)

if (dt.stand[, .N] > 0){
  ## Standkommentar
  dt.typ <- dt.stand[,
                    lapply(list(standkommentar),

```

```

        function(x)paste(x, collapse = " | "),
        keyby = c("standtyp")]

setnames(dt.typ,
         "V1",
         "standkommentar")

dt.typ <- transpose(dt.typ,
                   make.names = "standtyp")

setnames(dt.typ,
         names(dt.typ),
         tolower(names(dt.typ)))

## Standcheck
dt.check <- dt.stand[,lapply(.SD, as.factor)][, .(standtyp, standcheck)]
dt.check <- dt.check[, lapply(list(standtyp), unique), keyby = "
standcheck"]
setnames(dt.check,
         "V1",
         "standtyp")

dt.check <- transpose(dt.check, make.names = "standtyp")

setnames(dt.check,
         names(dt.check),
         paste0("check_",
               tolower(names(dt.check))))

dt.stand.all <- cbind(dt.typ, dt.check)

dt.stand.all.rep <- dt.stand.all[rep(dt.stand.all[, .I],
                                   content.out[, .N])]

out.dt <- cbind(meta,
               dt.stand.all.rep,
               content.out)
}else{
  out.dt <- cbind(meta,
                 content.out)
}

return(out.dt)
}

```

#### 6.7.4 Cluster beenden

```
stopCluster(cl)
```

#### 6.7.5 Liste in Data Table umwandeln

```
dt.normen <- rbindlist(out,
                        use.names = TRUE,
                        fill = TRUE)
```

### 6.7.6 Ende XML Parsing

```
end.parse <- Sys.time()
```

### 6.7.7 Dauer XML Parsing

```
end.parse - begin.parse
```

```
## Time difference of 3.815433 mins
```

### 6.7.8 Variable “doc\_id” erstellen

Eine einzigartige doc\_id wird benötigt um z.B. einen Quanteda-Korpus erstellen zu können. Diese wird aus dem Dateinamen zusammen mit einer Kollisionsnummer gebildet.

```
dt.normen$doc_id <- make.unique(dt.normen$dateiname)
```

### 6.7.9 Variablen-Name für Ausfertigungsdatum anpassen

```
setnames(dt.normen,
         "ausfertigung-datum",
         "ausfertigung_datum")
```

### 6.7.10 Variable “fundstellentyp” anpassen

```
dt.normen[grep("amtlich",
               dt.normen$fundstellentyp,
               invert = TRUE)]$fundstellentyp <- "nichtamtlich"
```

### 6.7.11 Variable “builddate\_iso” erstellen

```
dt.normen$builddate_iso <- as.POSIXct(dt.normen$builddate_original,
                                       format = "%Y%m%d%H%M%S")
```

### 6.7.12 Variable “aenderung\_datum” erstellen

```
dt.normen$aenderung_datum <- as.Date(gsub(".*([0-9]{1,2}\\.[0-9]{1,2}\\.[0-9]{4})
.*",
                                         "\\1",
                                         dt.normen$stand),
                                   format = "%d.%m.%Y")
```

### 6.7.13 Variable “aufhebung\_verkuendung\_datum” erstellen

Das Textfeld mit Informationen zur Aufhebung enthält zwei Daten. Das erste ist das der Verkündung des aufhebenden Rechtsaktes, das zweite das der Wirkung des aufhebenden Rechtsaktes. Für diese Variable wird das erste Datum verwendet.

```
dt.normen$aufhebung_verkuendung_datum <- as.Date(sub(".*
([0-9]{1,2}\\.[0-9]{1,2}\\.[0-9]{4}).*([0-9]{1,2}\\.[0-9]{1,2}\\.[0-9]{4}).*"
,
                                         "\\1",
                                         dt.normen$aufh),
                                   format = "%d.%m.%Y")
```

### 6.7.14 Variable “aufhebung\_wirkung\_datum” erstellen

Das Textfeld mit Informationen zur Aufhebung enthält zwei Daten. Das erste ist das der Verkündung des aufhebenden Rechtsaktes, das zweite das der Wirkung des aufhebenden Rechtsaktes. Für diese Variable wird das zweite Datum verwendet.

```
dt.normen$aufhebung_wirkung_datum <- as.Date(sub(".*
([0-9]{1,2}\\.[0-9]{1,2}\\.[0-9]{4}).*([0-9]{1,2}\\.[0-9]{1,2}\\.[0-9]{4}).*"
,
                                         "\\2",
                                         dt.normen$aufh),
                                   format = "%d.%m.%Y")
```

### 6.7.15 Variable “neufassung\_datum” erstellen

```
dt.normen$neufassung_datum <- as.Date(gsub(".*
([0-9]{1,2}\\.[0-9]{1,2}\\.[0-9]{4}).*",
                                         "\\1",
                                         dt.normen$neuf),
                                   format = "%d.%m.%Y")
```

### 6.7.16 Variable “ausfertigung\_jahr” hinzufügen

```
dt.normen$ausfertigung_jahr <- year(dt.normen$ausfertigung_datum)
```



#### 6.7.17 Variable “doi\_concept” hinzufügen

```
dt.normen$doi_concept <- rep(doi.concept,  
                             dt.normen[, .N])
```

#### 6.7.18 Variable “doi\_version” hinzufügen

```
dt.normen$doi_version <- rep(doi.version,  
                             dt.normen[, .N])
```

#### 6.7.19 Variable “version” hinzufügen

```
dt.normen$version <- as.character(rep(datestamp,  
                                       dt.normen[, .N]))
```

#### 6.7.20 Variable “lizenz” hinzufügen

```
dt.normen$lizenz <- as.character(rep(license,  
                                     dt.normen[, .N]))
```

### 6.8 Stichprobe für Qualitätsprüfung ziehen

```
idx <- sample(dt.normen[, .N], 300)  
  
check <- dt.normen[idx]  
  
fwrite(check,  
       paste0(outputdir,  
               datasetname,  
               "_Stichprobe_Normen.csv"),  
       na = "NA")
```

## 6.9 Korpus erstellen: Rechtsakte

### 6.9.1 Variablen definieren

Zunächst der vordefinierte Satz an Metadaten.

```
varlist.r1 <- gsub("ausfertigung-datum",  
                  "ausfertigung_datum",  
                  varlist)
```

Die Stand-Variablen haben immer auch ein Pendant das mit “check\_” beginnt.

```
standvars <- c("stand",  
               "aufh",  
               "neuf",  
               "hinweis",  
               "sonst")  
  
standvars <- c(standvars,  
               paste0("check_",  
                      standvars))
```

### 6.9.2 Vollständiger Satz an Variablen

```
varlist.r2 <- c(varlist.r1,  
                standvars,  
                "fundstellentyp",  
                "ausfertigung_jahr",  
                "aenderung_datum",  
                "aufhebung_verkuendung_datum",  
                "aufhebung_wirkung_datum",  
                "neufassung_datum",  
                "doi_concept",  
                "doi_version",  
                "version",  
                "lizenz")
```

### 6.9.3 Einzelnormen zu Rechtsakten vereinigen

```
text.rechtsakte <- dt.normen[,  
                             lapply(list(text),  
                                     function(x)paste(x, collapse = " ")),  
                             keyby = dateiname]  
  
setnames(text.rechtsakte,  
          "V1",  
          "text")
```

```
meta.rechtsakte <- dt.normen[,  
  lapply(.SD, unique),  
  .SDcols = varlist.r2,  
  keyby = dateiname]  
  
dt.rechtsakte <- text.rechtsakte[meta.rechtsakte,  
  on = "dateiname"]
```

#### 6.9.4 Variable “dateiname” in “doc\_id” umbenennen

```
setnames(dt.rechtsakte,  
  "dateiname",  
  "doc_id")
```

## 6.10 Datensatz erstellen: XML-Metadaten

An dieser Stelle werden Metadaten für alle Rechtsakte von “Gesetze im Internet” erhoben, unabhängig davon ob die Rechtsakte Text enthalten oder nur mit Überschrift nachgewiesen sind.

### 6.10.1 Beginn XML Parsing

```
begin.parse <- Sys.time()
```

### 6.10.2 Fork Cluster starten

```
cl <- makeForkCluster(fullCores)
registerDoParallel(cl)
```

### 6.10.3 XML Parsen

```
limit <- length(files.xml)

out <- foreach(z = 1:limit, .errorhandling = 'pass') %dopar% {

  XML <- read_xml(files.xml[z])
  nodes <- xml_nodes(XML, xpath = "//norm//metadaten")
  scope <- 1:length(nodes)

  meta <- vector("list", length(varlist))

  for (i in 1:length(varlist)){
    meta[[i]] <- xml_node(XML, varlist[i]) %>% xml_text()
  }

  setDT(meta)
  setnames(meta, new = varlist)

  meta$fundstellentyp <- xml_node(XML, "fundstelle") %>% xml_attr(attr = "typ")

  meta$doc_id <- files.xml[z]

  meta$builddate_original <- xml_attr(XML, attr = "builddate")

  ## Standangaben extrahieren
  standtyp <- xml_nodes(XML, "standtyp") %>% xml_text(trim = TRUE)
  standkommentar <- xml_nodes(XML, "standkommentar") %>% xml_text(trim = TRUE)
  standcheck <- xml_nodes(XML, "standangabe") %>% xml_attr(attr = "checked")

  dt.stand <- data.table(standtyp,
                        standkommentar,
                        standcheck)
```

```

if (dt.stand[,.N] > 0){

  ## Standkommentar
  dt.typ <- dt.stand[,
                    lapply(list(standkommentar),
                           function(x)paste(x, collapse = "  ")),
                    keyby = c("standtyp")]

  setnames(dt.typ,
            "V1",
            "standkommentar")

  dt.typ <- transpose(dt.typ,
                     make.names = "standtyp")

  setnames(dt.typ,
            names(dt.typ),
            tolower(names(dt.typ)))

  ## Standcheck
  dt.check <- dt.stand[,lapply(.SD, as.factor)][, .(standtyp, standcheck)]
  dt.check <- dt.check[, lapply(list(standtyp), unique), keyby = "
standcheck"]
  setnames(dt.check,
            "V1",
            "standtyp")

  dt.check <- transpose(dt.check, make.names = "standtyp")

  setnames(dt.check,
            names(dt.check),
            paste0("check_",
                  tolower(names(dt.check))))

  dt.stand.all <- cbind(dt.typ, dt.check)

  meta <- cbind(meta,
                dt.stand.all)
}

return(meta)
}

```

#### 6.10.4 Fork Cluster beenden

```
stopCluster(cl)
```

#### 6.10.5 Liste in Data Table umwandeln

```
dt.meta <- rbindlist(out,
                     use.names = TRUE,
                     fill = TRUE)
```

#### 6.10.6 Ende XML Parsing

```
end.parse <- Sys.time()
```

#### 6.10.7 Dauer XML Parsing

```
end.parse - begin.parse
```

```
## Time difference of 31.96959 secs
```

#### 6.10.8 Variablen-Name für Ausfertigungsdatum anpassen

```
setnames(dt.meta,
         "ausfertigung-datum",
         "ausfertigung_datum")
```

#### 6.10.9 Variable “fundstellentyp” anpassen

```
dt.meta[grepl("amtlich", dt.meta$fundstellentyp, invert = TRUE)]$fundstellentyp <-
  "nichtamtlich"
```

#### 6.10.10 Variable “builddate\_iso” erstellen

```
dt.meta$builddate_iso <- as.POSIXct(dt.meta$builddate_original,
                                   format = "%Y%m%d%H%M%S")
```

#### 6.10.11 Variable “aenderung\_datum” erstellen

```
dt.meta$aenderung_datum <- as.Date(gsub(".*([0-9]{1,2}\\.[0-9]{1,2}\\.[0-9]{4}).*",
    "\1",
    dt.meta$stand),
    format = "%d.%m.%Y")
```

#### 6.10.12 Variable “aufhebung\_verkuendung\_datum” erstellen

Das Textfeld mit Informationen zur Aufhebung enthält zwei Daten. Das erste ist das der Verkündung des aufhebenden Rechtsaktes, das zweite das der Wirkung des aufhebenden Rechtsaktes. Für diese Variable wird das erste Datum verwendet.

```
dt.meta$aufhebung_verkuendung_datum <- as.Date(sub(".*
([0-9]{1,2}\\. [0-9]{1,2}\\. [0-9]{4}) .* ([0-9]{1,2}\\. [0-9]{1,2}\\. [0-9]{4}) .*"
,
                                                    "\\1",
                                                    dt.meta$aufh),
format = "%d.%m.%Y")
```

#### 6.10.13 Variable “aufhebung\_wirkung\_datum” erstellen

Das Textfeld mit Informationen zur Aufhebung enthält zwei Daten. Das erste ist das der Verkündung des aufhebenden Rechtsaktes, das zweite das der Wirkung des aufhebenden Rechtsaktes. Für diese Variable wird das zweite Datum verwendet.

```
dt.meta$aufhebung_wirkung_datum <- as.Date(sub(".*
([0-9]{1,2}\\. [0-9]{1,2}\\. [0-9]{4}) .* ([0-9]{1,2}\\. [0-9]{1,2}\\. [0-9]{4}) .*"
,
                                                    "\\2",
                                                    dt.meta$aufh),
format = "%d.%m.%Y")
```

#### 6.10.14 Variable “neufassung\_datum” erstellen

```
dt.meta$neufassung_datum <- as.Date(gsub(".*([0-9]{1,2}\\. [0-9]{1,2}\\. [0-9]{4}) .
*",
                                                    "\\1",
                                                    dt.meta$neuf),
format = "%d.%m.%Y")
```

#### 6.10.15 Variable “ausfertigung\_jahr” hinzufügen

```
dt.meta$ausfertigung_jahr <- year(as.IDate(dt.meta$ausfertigung_datum))
```

#### 6.10.16 Variable “doi\_concept” hinzufügen

```
dt.meta$doi_concept <- rep(doi.concept, dt.meta[,.N])
```

### 6.10.17 Variable “doi\_version” hinzufügen

```
dt.meta$doi_version <- rep(doi.version, dt.meta[,.N])
```

### 6.10.18 Variable “version” hinzufügen

```
dt.meta$version <- as.character(rep(datestamp, dt.meta[,.N]))
```

### 6.10.19 Variable “lizenz” hinzufügen

```
dt.meta$lizenz <- as.character(rep(license,  
                                dt.meta[,.N]))
```

## 6.11 Netzwerk-Analyse (experimentell)

### 6.11.1 Funktion definieren: f.kennzahlen.search

```
f.kennzahlen.search <- function(pattern, targetvec){  
  
  pattern.N <- nchar(pattern)  
  target <- substr(targetvec, 1, pattern.N)  
  targetvec[grepl(pattern, target, fixed = TRUE)]  
  
}
```

### 6.11.2 Funktion definieren: f.kennzahlen.collapse

```
f.kennzahlen.collapse <- function(lev.begin, targets.list){  
  
  out.list <- vector("list", length(targets.list))  
  
  for (i in 1:length(targets.list)){  
  
    targets.vector <- targets.list[[i]]  
  
    out.list[[i]] <- data.table(rep(lev.begin[i],  
                                length(targets.vector)),  
                              targets.vector)  
  
  }  
  
  out.vec <- rbindlist(out.list)  
  return(out.vec)  
  
}
```



### 6.11.3 Funktion definieren: f.kennzahlen.edgelist

f.kennzahlen.edgelist: erstellt aus einem vektor an Gliederungskennzahlen und dem Gesetzesnamen ein Netzwerk-Diagramm der Inhaltsstruktur. Basiert auf f.kennzahlen.search und f.kennzahlen.collapse.

```
f.kennzahlen.edgelist <- function(kennzahl, name){  
  
  level <- nchar(kennzahl) / 3  
  
  level.unique <- sort(unique(level))  
  
  depth.begin <- head(seq_along(level.unique), -1)  
  depth.end <- depth.begin + 1  
  
  out.list <- vector("list", length(depth.begin))  
  
  for (i in seq_along(depth.begin)){  
  
    lev.begin <- kennzahl[level == depth.begin[i]]  
    lev.end <- kennzahl[level == depth.end[i]]  
  
    targets.list <- lapply(lev.begin, f.kennzahlen.search, lev.end)  
    out.list[[i]] <- f.kennzahlen.collapse(lev.begin, targets.list)  
  
  }  
  
  out.dt <- rbindlist(out.list)  
  
  ## Add zero level  
  
  if (length(depth.begin) != 0){  
    lev1 <- kennzahl[level == depth.begin[1]]  
  
    zerolinks <- data.table(rep(name, length(lev1)),  
                           lev1)  
  
    out.dt <- rbind(zerolinks,  
                   out.dt,  
                   use.names = FALSE)  
  }else{  
    lev1 <- kennzahl  
    out.dt <- data.table(rep(name, length(lev1)),  
                        lev1)  
  
  }  
  
  setnames(out.dt,  
           new = c("from",  
                  "to"))  
  
  return(out.dt)  
}
```

#### 6.11.4 Funktion definieren: f.network.analysis

f.network.analysis benötigt f.kennzahlen.search, f.kennzahlen.collapse und f.kennzahlen.edgelist.

```
f.network.analysis <- function(xml.name){

  XML <- read_xml(xml.name)

  kennzahl <- xml_nodes(XML, xpath = "//norm//gliederungskennzahl") %>% xml_text()
  kennzahl <- make.unique(kennzahl)
  bez <- xml_nodes(XML, xpath = "//norm//gliederungsbez") %>% xml_text()
  titel <- xml_nodes(XML, xpath = "//norm//gliederungstitel") %>% xml_text()

  bez <- gsub(" +",
             "\n",
             bez)

  titel <- gsub(" +",
               "\n",
               titel)

  jurabk <- xml_node(XML, xpath = "//norm//jurabk") %>% xml_text()

  node.labels0 <- ifelse(titel != "",
                        titel,
                        bez)

  node.labels <- c(jurabk,
                  node.labels0)

  edgelist <- f.kennzahlen.edgelist(kennzahl,
                                   jurabk)

  nodes.df <- data.table(kennzahl,
                        titel)

  addname <- data.table(jurabk,
                       jurabk)

  setnames(addname, new = c("kennzahl",
                           "titel"))

  nodes.df <- rbind(addname,
                   nodes.df)

  setnames(nodes.df, new = c("kennzahl",
                           "label"))

  g <- graph.data.frame(edgelist,
                       directed = TRUE,
                       vertices = nodes.df)

  M.adjacency <- as.matrix(get.adjacency(g,
```

```

edges = F))

filename <- paste0(gsub("\\\\.xml",
                        "",
                        xml.name))

fwrite(edgelist,
       paste0("Netzwerke/Edgelisten/",
              filename,
              "_Edgelist.csv"))

fwrite(M.adjacency,
       paste0("Netzwerke/Adjazenzmatrizen/",
              filename,
              "_AdjazenzMatrix.csv"))

write_graph(g,
            file = paste0("Netzwerke/GraphML/",
                           filename,
                           ".graphml"),
            format = "graphml")

if (length(V(g)) > 1){

  networkplot <- ggraph(g,
                        'dendrogram',
                        circular = TRUE) +
    geom_edge_elbow(colour = "grey") +
    geom_node_text(aes(label = label),
                  size = 2,
                  repel = TRUE)+
    theme_void()+
    labs(
      title = paste(datasetname,
                    "| Version",
                    datestamp,
                    "| Struktur des",
                    jurabk),
      caption = paste("DOI:",
                      doi.version)
    )+
    theme(
      plot.title = element_text(size = 50,
                                face = "bold"),
      legend.position = "none",
      plot.margin = margin(10, 20, 10, 10)
    )

  ## may conflict with markdown save
  ggsave(
    filename = paste0("Netzwerke/Netzwerkdiagramme/",
                      filename,
                      "_NetzwerkDiagramm.pdf"),
    plot = networkplot,
    device = "pdf",

```

```

        scale = 1,
        width = 50,
        height = 50,
        units = "in",
        dpi = 300,
        limitsize = FALSE
    )
}
}

```

### 6.11.5 Netzwerk-Analyse durchführen

```

files.xml <- list.files(pattern = "\\\\.xml$")

errorfiles <- c("BJNR008810961.xml",
               "BJNR010599989.xml",
               "BJNR043410015.xml",
               "BJNR093000015.xml",
               "BJNR135410017.xml",
               "BJNR158720007.xml",
               "BJNR203210978.xml",
               "BJNR203220978.xml",
               "BJNR277700013.xml",
               "BJNR284600017.xml",
               "BJNR364800009.xml")

files.xml <- setdiff(files.xml, errorfiles)

length(files.xml)

```

```
## [1] 6619
```

```

### Parallele Berechnung funktioniert nicht mit errorfiles
cl <- makeForkCluster(fullCores)
registerDoParallel(cl)

### Sequentielle Berechnung funktioniert auch mit errorfiles
#registerDoSEQ(cl)

out <- foreach(file = files.xml,
               .errorhandling = 'pass') %dopar% {

  f.network.analysis(file)

}

stopCluster(cl)

```

### 6.11.6 XML-Dateien bei denen Fehler auftreten

```
files.xml[grep("error",  
              out)]
```

```
## character(0)
```

## 6.12 Wiederverpacken der XML-Dateien

Wiederverpacken der gesammelten XML-Dateien in ein einziges Archiv. Wiederverpacken der Anlagen in ein separates Archiv. Die Roh-Daten werden im Anschluss jeweils gelöscht.

### 6.12.1 XML-Dateien definieren

```
files.xml <- list.files(pattern = "\\\\.xml")
```

### 6.12.2 XML-Dateien verpacken

```
zip(paste(datasetname,  
          datestamp,  
          "DE_XML_Datensatz.zip",  
          sep = "_"),  
    files.xml)  
  
unlink(files.xml)
```

### 6.12.3 Anhänge zu XML-Dateien verpacken

```
attachments <- list.files(pattern="(\\.jpg)|\\.gif)|\\.pdf)|\\.png)",  
                          ignore.case = TRUE)  
  
zip(paste(datasetname,  
          datestamp,  
          "DE_XML_Anlagen.zip",  
          sep = "_"),  
    attachments)  
  
unlink(attachments)
```

## 7 Frequenztabellen erstellen: Einzelnormen

### 7.1 Funktion anzeigen: f.fast.freqtable

```
print(f.fast.freqtable)
```

```
function(x, varlist = names(x), sumrow = TRUE, output.list = TRUE, output.kable = FALSE, output.csv = FALSE, outputdir = "./", prefix = „“, align = "r"){
```

```
## Begin List
freqtable.list <- vector("list", length(varlist))

## Calculate Frequency Table
for (i in seq_along(varlist)){

  varname <- varlist[i]

  freqtable <- x[, .N, keyby=c(paste0(varname))]

  freqtable[, c("exactpercent",
               "roundedpercent",
               "cumulpercent") := {
    exactpercent <- N/sum(N)*100
    roundedpercent <- round(exactpercent, 2)
    cumulpercent <- round(cumsum(exactpercent), 2)
    list(exactpercent,
         roundedpercent,
         cumulpercent)}]

  ## Calculate Summary Row
  if (sumrow == TRUE){
    colsums <- cbind("Total",
                    freqtable[, lapply(.SD, function(x){round(sum(x))}),
                      .SDcols = c("N",
                                   "exactpercent",
                                   "roundedpercent")
                    ], round(max(freqtable$cumulpercent)))

    colnames(colsums)[c(1,5)] <- c(varname, "cumulpercent")
    freqtable <- rbind(freqtable, colsums)
  }

  ## Add Frequency Table to List
  freqtable.list[[i]] <- freqtable

  ## Write CSV
  if (output.csv == TRUE){

    fwrite(freqtable,
           paste0(outputdir,
                  prefix,
                  varname,
```

```

        ".csv"),
        na = "NA")

}

## Output Kable
if (output.kable == TRUE){

  cat("\n-----\n")
  cat(paste0("Frequency Table for Variable:  ", varname, "\n"))
  cat("-----\n")
  cat(paste0("\n ",
             x[, .N, keyby=c(paste0(varname))][, .N],
             " unique value(s) detected.\n\n"))

  print(kable(freqtable,
             format = "latex",
             align = align,
             booktabs = TRUE,
             longtable = TRUE) %>% kable_styling(latex_options = "repeat_
header"))
}

}

## Return List of Frequency Tables
if (output.list == TRUE){
  return(freqtable.list)
}

}

```

## 7.2 Liste zu prüfender Variablen

```
print(vars.freqtable)
```

```
## [1] "periodikum"      "fundstellentyp"  "check_neuf"
## [4] "check_aufh"      "check_sonst"     "check_hinweis"
## [7] "check_stand"     "gliederungskennzahl" "ausfertigung_jahr"
## [10] "doi_concept"     "doi_version"     "version"
## [13] "lizenz"
```

## 7.3 Frequenztabellen erstellen

```
prefix <- paste0(datasetname,
                  "_01_Einzelnormen_Frequenztafel_var-")
```

```
f.fast.freqtable(dt.normen,
                 varlist = vars.freqtable,
                 sumrow = TRUE,
                 output.list = FALSE,
                 output.kable = TRUE,
                 output.csv = TRUE,
                 outputdir = outputdir,
                 prefix = prefix)
```

---

Frequency Table for Variable: periodikum

---

21 unique value(s) detected.

periodikum	N	exactpercent	roundedpercent	cumulpercent
BAGVBl	1	0.0010023	0.00	0.00
BAnz	1237	1.2399014	1.24	1.24
BGBI	2292	2.2973759	2.30	3.54
BGBI I	83769	83.9654792	83.97	87.50
BGBI II	3254	3.2616322	3.26	90.77
GBl DDR	233	0.2335465	0.23	91.00
GBl DDR I	616	0.6174448	0.62	91.62
GBl DDR II	43	0.0431009	0.04	91.66
GVBl BE	9	0.0090211	0.01	91.67
NV	15	0.0150352	0.02	91.68
RAnz	38	0.0380891	0.04	91.72
RGBI	5649	5.6622497	5.66	97.38
RGBI I	1630	1.6338231	1.63	99.02
RGBI II	347	0.3478139	0.35	99.37
RMBI	230	0.2305395	0.23	99.60
VOBl BrZ	19	0.0190446	0.02	99.62
VkBl	106	0.1062486	0.11	99.72
WiGBI	122	0.1222861	0.12	99.84
ZBl	63	0.0631478	0.06	99.91
eBAnz	52	0.0521220	0.05	99.96



(continued)

periodikum	N	exactpercent	roundedpercent	cumulpercent
Öff Anz	41	0.0410962	0.04	100.00
Total	99766	100.0000000	100.00	100.00

Frequency Table for Variable: fundstellentyp

1 unique value(s) detected.

fundstellentyp	N	exactpercent	roundedpercent	cumulpercent
amtlich	99766	100	100	100
Total	99766	100	100	100

Frequency Table for Variable: check\_neuf

2 unique value(s) detected.

check_neuf	N	exactpercent	roundedpercent	cumulpercent
NA	75748	75.92567	75.93	75.93
ja	24018	24.07433	24.07	100.00
Total	99766	100.00000	100.00	100.00

Frequency Table for Variable: check\_aufh

2 unique value(s) detected.

check_aufh	N	exactpercent	roundedpercent	cumulpercent
NA	97579	97.80787	97.81	97.81
ja	2187	2.19213	2.19	100.00
Total	99766	100.00000	100.00	100.00

---

Frequency Table for Variable: check\_sonst

---

2 unique value(s) detected.

check_sonst	N	exactpercent	roundedpercent	cumulpercent
NA	82698	82.89197	82.89	82.89
ja	17068	17.10803	17.11	100.00
Total	99766	100.00000	100.00	100.00

---

Frequency Table for Variable: check\_hinweis

---

2 unique value(s) detected.

check_hinweis	N	exactpercent	roundedpercent	cumulpercent
NA	66500	66.65597	66.66	66.66
ja	33266	33.34403	33.34	100.00
Total	99766	100.00000	100.00	100.00

---

Frequency Table for Variable: check\_stand

---

2 unique value(s) detected.

check_stand	N	exactpercent	roundedpercent	cumulpercent
NA	22951	23.00483	23	23
ja	76815	76.99517	77	100
Total	99766	100.00000	100	100

---

Frequency Table for Variable: gliederungskennzahl

---

1456 unique value(s) detected.

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
000	92	0.0922158	0.09	0.09
000010	12	0.0120281	0.01	0.10
000020	18	0.0180422	0.02	0.12
000030	17	0.0170399	0.02	0.14
000040	17	0.0170399	0.02	0.16
000050	18	0.0180422	0.02	0.17
000060	10	0.0100235	0.01	0.18
000070	3	0.0030070	0.00	0.19
000080	2	0.0020047	0.00	0.19
000090	1	0.0010023	0.00	0.19
000100	2	0.0020047	0.00	0.19
000110	3	0.0030070	0.00	0.20
000120	5	0.0050117	0.01	0.20
001000	2	0.0020047	0.00	0.20
001010	1	0.0010023	0.00	0.20
001020	3	0.0030070	0.00	0.21
001030	1	0.0010023	0.00	0.21
001040	1	0.0010023	0.00	0.21
001051	1	0.0010023	0.00	0.21
001060	2	0.0020047	0.00	0.21
001080	1	0.0010023	0.00	0.21
001090	1	0.0010023	0.00	0.21
001100	1	0.0010023	0.00	0.21
001101	1	0.0010023	0.00	0.22
001110	1	0.0010023	0.00	0.22
001120	5	0.0050117	0.01	0.22
010	6040	6.0541668	6.05	6.28
010010	905	0.9071227	0.91	7.18
010010000010	1	0.0010023	0.00	7.18
010010000020	1	0.0010023	0.00	7.18

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
010010010	118	0.1182768	0.12	7.30
010010020	117	0.1172744	0.12	7.42
010010020010	3	0.0030070	0.00	7.42
010010020010010	37	0.0370868	0.04	7.46
010010020010020	24	0.0240563	0.02	7.48
010010020020	12	0.0120281	0.01	7.50
010010020030	1	0.0010023	0.00	7.50
010010030	28	0.0280657	0.03	7.53
010010030010	7	0.0070164	0.01	7.53
010010030020	16	0.0160375	0.02	7.55
010010030030	12	0.0120281	0.01	7.56
010010030040	12	0.0120281	0.01	7.57
010010040	35	0.0350821	0.04	7.61
010010050	9	0.0090211	0.01	7.62
010010060	6	0.0060141	0.01	7.62
010010070010	10	0.0100235	0.01	7.63
010010070020	5	0.0050117	0.01	7.64
010020	788	0.7898482	0.79	8.43
010020010	135	0.1353166	0.14	8.56
010020010000030	1	0.0010023	0.00	8.56
010020010000040	1	0.0010023	0.00	8.57
010020010000050	1	0.0010023	0.00	8.57
010020010000060	1	0.0010023	0.00	8.57
010020020	131	0.1313073	0.13	8.70
010020020000070	1	0.0010023	0.00	8.70
010020020000080	1	0.0010023	0.00	8.70
010020020000090	1	0.0010023	0.00	8.70
010020020000100	1	0.0010023	0.00	8.70
010020020000110	1	0.0010023	0.00	8.70

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
010020020000120	1	0.0010023	0.00	8.70
010020020010	16	0.0160375	0.02	8.72
010020020020	19	0.0190446	0.02	8.74
010020020030	9	0.0090211	0.01	8.75
010020020040	14	0.0140328	0.01	8.76
010020020050	18	0.0180422	0.02	8.78
010020030	103	0.1032416	0.10	8.88
010020030000130	1	0.0010023	0.00	8.88
010020030000140	1	0.0010023	0.00	8.89
010020030000170	1	0.0010023	0.00	8.89
010020030000171	1	0.0010023	0.00	8.89
010020030000172	1	0.0010023	0.00	8.89
010020030000190	1	0.0010023	0.00	8.89
010020030000200	1	0.0010023	0.00	8.89
010020030000210	1	0.0010023	0.00	8.89
010020030000220	1	0.0010023	0.00	8.89
010020030000230	1	0.0010023	0.00	8.89
010020030000240	1	0.0010023	0.00	8.89
010020030010	3	0.0030070	0.00	8.90
010020030020	2	0.0020047	0.00	8.90
010020030030	2	0.0020047	0.00	8.90
010020040	70	0.0701642	0.07	8.97
010020040000250	1	0.0010023	0.00	8.97
010020040000260	1	0.0010023	0.00	8.97
010020050	53	0.0531243	0.05	9.03
010020050020000380	1	0.0010023	0.00	9.03
010020050020000390	1	0.0010023	0.00	9.03
010020050020000400	1	0.0010023	0.00	9.03
010020050020000410	1	0.0010023	0.00	9.03

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
010020050020000420	1	0.0010023	0.00	9.03
010020060	44	0.0441032	0.04	9.08
010020060000430	1	0.0010023	0.00	9.08
010020060000440	1	0.0010023	0.00	9.08
010020060000450	1	0.0010023	0.00	9.08
010020060000460	1	0.0010023	0.00	9.08
010020070	31	0.0310727	0.03	9.11
010020070010000461	1	0.0010023	0.00	9.11
010020070020000462	1	0.0010023	0.00	9.11
010020070020000463	1	0.0010023	0.00	9.11
010020070030000464	1	0.0010023	0.00	9.12
010020070040000465	1	0.0010023	0.00	9.12
010020080	41	0.0410962	0.04	9.16
010020090	1	0.0010023	0.00	9.16
010030	526	0.5272337	0.53	9.69
010030000470	1	0.0010023	0.00	9.69
010030000480	1	0.0010023	0.00	9.69
010030010	133	0.1333119	0.13	9.82
010030010010	8	0.0080188	0.01	9.83
010030010020	14	0.0140328	0.01	9.84
010030010030	20	0.0200469	0.02	9.86
010030010040	12	0.0120281	0.01	9.88
010030010050	4	0.0040094	0.00	9.88
010030010060	8	0.0080188	0.01	9.89
010030010070	8	0.0080188	0.01	9.90
010030010080	4	0.0040094	0.00	9.90
010030010380	2	0.0020047	0.00	9.90
010030010400	4	0.0040094	0.00	9.91
010030010440	1	0.0010023	0.00	9.91

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
010030010450	3	0.0030070	0.00	9.91
010030020	84	0.0841970	0.08	9.99
010030020010	86	0.0862017	0.09	10.08
010030020020	40	0.0400938	0.04	10.12
010030020030	12	0.0120281	0.01	10.13
010030020040	12	0.0120281	0.01	10.14
010030020050	9	0.0090211	0.01	10.15
010030020060	18	0.0180422	0.02	10.17
010030030	70	0.0701642	0.07	10.24
010030030010	9	0.0090211	0.01	10.25
010030030020	9	0.0090211	0.01	10.26
010030030030	6	0.0060141	0.01	10.27
010030030040	1	0.0010023	0.00	10.27
010030040	39	0.0390915	0.04	10.31
010030050	39	0.0390915	0.04	10.34
010030060	6	0.0060141	0.01	10.35
010030060010	16	0.0160375	0.02	10.37
010030060020	8	0.0080188	0.01	10.37
010030060030	3	0.0030070	0.00	10.38
010030060040	3	0.0030070	0.00	10.38
010030060050	2	0.0020047	0.00	10.38
010030070	17	0.0170399	0.02	10.40
010031	2	0.0020047	0.00	10.40
010040	341	0.3417998	0.34	10.74
010040010	83	0.0831947	0.08	10.83
010040020	69	0.0691618	0.07	10.90
010040020010	3	0.0030070	0.00	10.90
010040020020	5	0.0050117	0.01	10.90
010040020030	2	0.0020047	0.00	10.91

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
010040030	42	0.0420985	0.04	10.95
010040030010	8	0.0080188	0.01	10.96
010040030020	3	0.0030070	0.00	10.96
010040040	21	0.0210493	0.02	10.98
010040040010	7	0.0070164	0.01	10.99
010040040020	11	0.0110258	0.01	11.00
010040040030	4	0.0040094	0.00	11.00
010040040040	9	0.0090211	0.01	11.01
010040040050	1	0.0010023	0.00	11.01
010040040060	3	0.0030070	0.00	11.01
010040040070	8	0.0080188	0.01	11.02
010040050	2	0.0020047	0.00	11.02
010040120010	1	0.0010023	0.00	11.03
010040120020	1	0.0010023	0.00	11.03
010040120030	1	0.0010023	0.00	11.03
010040120040	1	0.0010023	0.00	11.03
010040120050	1	0.0010023	0.00	11.03
010040120060	1	0.0010023	0.00	11.03
010041	2	0.0020047	0.00	11.03
010050	163	0.1633823	0.16	11.20
010050010	61	0.0611431	0.06	11.26
010050020	86	0.0862017	0.09	11.34
010050020020	2	0.0020047	0.00	11.35
010050030	42	0.0420985	0.04	11.39
010050030010	2	0.0020047	0.00	11.39
010050030020	1	0.0010023	0.00	11.39
010050030030	2	0.0020047	0.00	11.39
010050040	2	0.0020047	0.00	11.39
010060	181	0.1814245	0.18	11.58



(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
010060010	48	0.0481126	0.05	11.62
010060020	73	0.0731712	0.07	11.70
010060020010	11	0.0110258	0.01	11.71
010060020020	10	0.0100235	0.01	11.72
010060020030	5	0.0050117	0.01	11.72
010060020040	14	0.0140328	0.01	11.74
010060020050	1	0.0010023	0.00	11.74
010060030	57	0.0571337	0.06	11.80
010060030810	7	0.0070164	0.01	11.80
010060030820	8	0.0080188	0.01	11.81
010060030830	3	0.0030070	0.00	11.81
010060030840	1	0.0010023	0.00	11.81
010060040	14	0.0140328	0.01	11.83
010060050	33	0.0330774	0.03	11.86
010060051	4	0.0040094	0.00	11.87
010060060	23	0.0230539	0.02	11.89
010060070	5	0.0050117	0.01	11.89
010070	220	0.2205160	0.22	12.11
010070010	8	0.0080188	0.01	12.12
010070010010	11	0.0110258	0.01	12.13
010070010020	6	0.0060141	0.01	12.14
010070020	14	0.0140328	0.01	12.15
010070030	14	0.0140328	0.01	12.17
010080	131	0.1313073	0.13	12.30
010080010	9	0.0090211	0.01	12.31
010080010010	2	0.0020047	0.00	12.31
010080010020	11	0.0110258	0.01	12.32
010080020	17	0.0170399	0.02	12.34
010080030	8	0.0080188	0.01	12.35

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
010090	61	0.0611431	0.06	12.41
010090010	23	0.0230539	0.02	12.43
010090020	19	0.0190446	0.02	12.45
010090030	6	0.0060141	0.01	12.46
010091	5	0.0050117	0.01	12.46
010092	1	0.0010023	0.00	12.46
010100	25	0.0250586	0.03	12.49
010100010	2	0.0020047	0.00	12.49
010100020	1	0.0010023	0.00	12.49
010110	37	0.0370868	0.04	12.53
010110010	1	0.0010023	0.00	12.53
010110020	1	0.0010023	0.00	12.53
010110030	1	0.0010023	0.00	12.53
010120	4	0.0040094	0.00	12.53
010130	9	0.0090211	0.01	12.54
010140	5	0.0050117	0.01	12.55
011	11	0.0110258	0.01	12.56
012	2	0.0020047	0.00	12.56
014	2	0.0020047	0.00	12.56
020	6196	6.2105326	6.21	18.77
020000500	1	0.0010023	0.00	18.77
020000510	1	0.0010023	0.00	18.77
020000520	1	0.0010023	0.00	18.78
020000530	1	0.0010023	0.00	18.78
020000531	1	0.0010023	0.00	18.78
020010	1606	1.6097669	1.61	20.39
020010010	293	0.2936872	0.29	20.68
020010020	230	0.2305395	0.23	20.91
020010020010	25	0.0250586	0.03	20.94

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020010020020	8	0.0080188	0.01	20.95
020010020020010	4	0.0040094	0.00	20.95
020010020020020	3	0.0030070	0.00	20.95
020010020020030	3	0.0030070	0.00	20.96
020010020020040	3	0.0030070	0.00	20.96
020010020020050	1	0.0010023	0.00	20.96
020010020030	12	0.0120281	0.01	20.97
020010020040	6	0.0060141	0.01	20.98
020010020050	5	0.0050117	0.01	20.98
020010020060	4	0.0040094	0.00	20.99
020010030	144	0.1443378	0.14	21.13
020010030010	11	0.0110258	0.01	21.14
020010030020	7	0.0070164	0.01	21.15
020010030030010	4	0.0040094	0.00	21.15
020010030030020010	5	0.0050117	0.01	21.16
020010030030020020	8	0.0080188	0.01	21.17
020010030040	4	0.0040094	0.00	21.17
020010040	91	0.0912134	0.09	21.26
020010040010	5	0.0050117	0.01	21.27
020010040020	3	0.0030070	0.00	21.27
020010040030	6	0.0060141	0.01	21.27
020010041	5	0.0050117	0.01	21.28
020010050	92	0.0922158	0.09	21.37
020010060	32	0.0320751	0.03	21.40
020010070	38	0.0380891	0.04	21.44
020010070010	4	0.0040094	0.00	21.45
020010070020	7	0.0070164	0.01	21.45
020010070030	7	0.0070164	0.01	21.46
020010080	19	0.0190446	0.02	21.48

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020010090	31	0.0310727	0.03	21.51
020010100	11	0.0110258	0.01	21.52
020010110	7	0.0070164	0.01	21.53
020010120	9	0.0090211	0.01	21.54
020011	3	0.0030070	0.00	21.54
020011042	1	0.0010023	0.00	21.54
020020	1521	1.5245675	1.52	23.07
020020010	147	0.1473448	0.15	23.21
020020010010	31	0.0310727	0.03	23.24
020020010010010	2	0.0020047	0.00	23.25
020020010010020	1	0.0010023	0.00	23.25
020020010010030	3	0.0030070	0.00	23.25
020020010010040	1	0.0010023	0.00	23.25
020020010010050	3	0.0030070	0.00	23.25
020020010020	11	0.0110258	0.01	23.27
020020010030	6	0.0060141	0.01	23.27
020020010040	10	0.0100235	0.01	23.28
020020010050	2	0.0020047	0.00	23.28
020020010060	6	0.0060141	0.01	23.29
020020010070	3	0.0030070	0.00	23.29
020020010080	2	0.0020047	0.00	23.29
020020020	220	0.2205160	0.22	23.52
020020020010	33	0.0330774	0.03	23.55
020020020020	29	0.0290680	0.03	23.58
020020020030	25	0.0250586	0.03	23.60
020020020040	9	0.0090211	0.01	23.61
020020020050	19	0.0190446	0.02	23.63
020020020060	1	0.0010023	0.00	23.63
020020020070010	3	0.0030070	0.00	23.63

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020020020070020	7	0.0070164	0.01	23.64
020020020070030	2	0.0020047	0.00	23.64
020020030	93	0.0932181	0.09	23.74
020020030010	19	0.0190446	0.02	23.76
020020030020	23	0.0230539	0.02	23.78
020020030030	19	0.0190446	0.02	23.80
020020030040	17	0.0170399	0.02	23.81
020020030050	5	0.0050117	0.01	23.82
020020030060	2	0.0020047	0.00	23.82
020020030070	16	0.0160375	0.02	23.84
020020030080	4	0.0040094	0.00	23.84
020020030090	3	0.0030070	0.00	23.84
020020030100	1	0.0010023	0.00	23.85
020020040	64	0.0641501	0.06	23.91
020020043	1	0.0010023	0.00	23.91
020020050	19	0.0190446	0.02	23.93
020020050010	24	0.0240563	0.02	23.95
020020050020	9	0.0090211	0.01	23.96
020020060	44	0.0441032	0.04	24.01
020020060010	1	0.0010023	0.00	24.01
020020060020	6	0.0060141	0.01	24.01
020020060030	5	0.0050117	0.01	24.02
020020070	12	0.0120281	0.01	24.03
020020080	2	0.0020047	0.00	24.03
020020080010	6	0.0060141	0.01	24.04
020020080020	11	0.0110258	0.01	24.05
020020080030	4	0.0040094	0.00	24.05
020020080040	2	0.0020047	0.00	24.06
020020090	4	0.0040094	0.00	24.06

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020020100	13	0.0130305	0.01	24.07
020021	23	0.0230539	0.02	24.10
020030	1084	1.0865425	1.09	25.18
020030010	148	0.1483471	0.15	25.33
020030010010	11	0.0110258	0.01	25.34
020030010020	5	0.0050117	0.01	25.35
020030010020010	2	0.0020047	0.00	25.35
020030010020020	7	0.0070164	0.01	25.36
020030010020030	2	0.0020047	0.00	25.36
020030010020040	1	0.0010023	0.00	25.36
020030010030	8	0.0080188	0.01	25.37
020030010040	6	0.0060141	0.01	25.37
020030020	222	0.2225207	0.22	25.60
020030020010	19	0.0190446	0.02	25.61
020030020020	2	0.0020047	0.00	25.62
020030030	106	0.1062486	0.11	25.72
020030040	48	0.0481126	0.05	25.77
020030040010	6	0.0060141	0.01	25.78
020030040020	2	0.0020047	0.00	25.78
020030040030	5	0.0050117	0.01	25.78
020030040040	1	0.0010023	0.00	25.79
020030050	39	0.0390915	0.04	25.82
020030050010	9	0.0090211	0.01	25.83
020030050020	16	0.0160375	0.02	25.85
020030060	6	0.0060141	0.01	25.86
020030070	7	0.0070164	0.01	25.86
020030080	9	0.0090211	0.01	25.87
020030090	3	0.0030070	0.00	25.87
020030100	1	0.0010023	0.00	25.88

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020030100010	1	0.0010023	0.00	25.88
020030100020	11	0.0110258	0.01	25.89
020030100030	1	0.0010023	0.00	25.89
020030110	3	0.0030070	0.00	25.89
020030120	2	0.0020047	0.00	25.89
020030130	7	0.0070164	0.01	25.90
020030140	2	0.0020047	0.00	25.90
020030150	8	0.0080188	0.01	25.91
020031	18	0.0180422	0.02	25.93
020040	745	0.7467474	0.75	26.68
020040010	54	0.0541267	0.05	26.73
020040020	56	0.0561313	0.06	26.79
020040030	38	0.0380891	0.04	26.82
020040030010	1	0.0010023	0.00	26.82
020040030020	6	0.0060141	0.01	26.83
020040040	22	0.0220516	0.02	26.85
020040050	5	0.0050117	0.01	26.86
020040060	8	0.0080188	0.01	26.87
020040070	1	0.0010023	0.00	26.87
020040160	1	0.0010023	0.00	26.87
020041	6	0.0060141	0.01	26.87
020050	428	0.4290039	0.43	27.30
020050010	40	0.0400938	0.04	27.34
020050010010	5	0.0050117	0.01	27.35
020050010020	4	0.0040094	0.00	27.35
020050010030	3	0.0030070	0.00	27.36
020050020	39	0.0390915	0.04	27.39
020050030	5	0.0050117	0.01	27.40
020050030010	7	0.0070164	0.01	27.41

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020050030020	14	0.0140328	0.01	27.42
020050030030	6	0.0060141	0.01	27.43
020050030040	2	0.0020047	0.00	27.43
020050030050	12	0.0120281	0.01	27.44
020050040	7	0.0070164	0.01	27.45
020050050	1	0.0010023	0.00	27.45
020050180	1	0.0010023	0.00	27.45
020050190	1	0.0010023	0.00	27.45
020050200	1	0.0010023	0.00	27.45
020050210	1	0.0010023	0.00	27.45
020051	7	0.0070164	0.01	27.46
020051211	1	0.0010023	0.00	27.46
020051212	1	0.0010023	0.00	27.46
020051213	1	0.0010023	0.00	27.46
020052	1	0.0010023	0.00	27.46
020053	1	0.0010023	0.00	27.46
020060	297	0.2976966	0.30	27.76
020060010	31	0.0310727	0.03	27.79
020060020	18	0.0180422	0.02	27.81
020060020010	11	0.0110258	0.01	27.82
020060020020010	2	0.0020047	0.00	27.82
020060020020020	2	0.0020047	0.00	27.83
020060020020030	1	0.0010023	0.00	27.83
020060020020040	1	0.0010023	0.00	27.83
020060020020050	1	0.0010023	0.00	27.83
020060030	10	0.0100235	0.01	27.84
020060030010	2	0.0020047	0.00	27.84
020060030020	2	0.0020047	0.00	27.84
020060030030	12	0.0120281	0.01	27.86



(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020060030040	6	0.0060141	0.01	27.86
020060030050	1	0.0010023	0.00	27.86
020060040	9	0.0090211	0.01	27.87
020060050	4	0.0040094	0.00	27.88
020060050010	2	0.0020047	0.00	27.88
020060050020	14	0.0140328	0.01	27.89
020060060	6	0.0060141	0.01	27.90
020060070	9	0.0090211	0.01	27.91
020060080010	5	0.0050117	0.01	27.91
020060080020	14	0.0140328	0.01	27.93
020060090	11	0.0110258	0.01	27.94
020060100	1	0.0010023	0.00	27.94
020060110	3	0.0030070	0.00	27.94
020060220	1	0.0010023	0.00	27.94
020060230	2	0.0020047	0.00	27.94
020061	3	0.0030070	0.00	27.95
020070	184	0.1844316	0.18	28.13
020070010	17	0.0170399	0.02	28.15
020070020	5	0.0050117	0.01	28.15
020070020010	10	0.0100235	0.01	28.16
020070020020	23	0.0230539	0.02	28.19
020070020020010	1	0.0010023	0.00	28.19
020070020020020	1	0.0010023	0.00	28.19
020070020030010	15	0.0150352	0.02	28.20
020070020030020010	3	0.0030070	0.00	28.21
020070020030020020	6	0.0060141	0.01	28.21
020070020030020030	1	0.0010023	0.00	28.21
020070020030020040	2	0.0020047	0.00	28.22
020070020030020050	1	0.0010023	0.00	28.22

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020070020030030	1	0.0010023	0.00	28.22
020070020040	1	0.0010023	0.00	28.22
020070020050	2	0.0020047	0.00	28.22
020070020060	5	0.0050117	0.01	28.23
020070020069	1	0.0010023	0.00	28.23
020070020070	3	0.0030070	0.00	28.23
020070020080	1	0.0010023	0.00	28.23
020070020090	1	0.0010023	0.00	28.23
020070030	4	0.0040094	0.00	28.24
020070030010	10	0.0100235	0.01	28.25
020070030020	6	0.0060141	0.01	28.25
020070030030	14	0.0140328	0.01	28.27
020070030040	3	0.0030070	0.00	28.27
020070030050	1	0.0010023	0.00	28.27
020070040	5	0.0050117	0.01	28.27
020070240	1	0.0010023	0.00	28.28
020080	136	0.1363190	0.14	28.41
020080010	5	0.0050117	0.01	28.42
020080010010	23	0.0230539	0.02	28.44
020080010020010	2	0.0020047	0.00	28.44
020080010020020	7	0.0070164	0.01	28.45
020080010020030	11	0.0110258	0.01	28.46
020080010030	6	0.0060141	0.01	28.47
020080010040	1	0.0010023	0.00	28.47
020080020	18	0.0180422	0.02	28.48
020080030	3	0.0030070	0.00	28.49
020080030010010	3	0.0030070	0.00	28.49
020080030010020	24	0.0240563	0.02	28.51
020080030020	3	0.0030070	0.00	28.52

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020080030030	1	0.0010023	0.00	28.52
020080030040	1	0.0010023	0.00	28.52
020080030050	2	0.0020047	0.00	28.52
020080030060	2	0.0020047	0.00	28.52
020080040	21	0.0210493	0.02	28.54
020080050	1	0.0010023	0.00	28.55
020080050010	19	0.0190446	0.02	28.56
020080050020010	7	0.0070164	0.01	28.57
020080050020011	6	0.0060141	0.01	28.58
020080050020020010	4	0.0040094	0.00	28.58
020080050020020011	4	0.0040094	0.00	28.59
020080050020020020	16	0.0160375	0.02	28.60
020080050020030	5	0.0050117	0.01	28.61
020080050020040	14	0.0140328	0.01	28.62
020080050020050010	5	0.0050117	0.01	28.63
020080050020050020	9	0.0090211	0.01	28.64
020080050020050030	2	0.0020047	0.00	28.64
020080050020050040	3	0.0030070	0.00	28.64
020080050020060	2	0.0020047	0.00	28.64
020080050030	5	0.0050117	0.01	28.65
020080050040	8	0.0080188	0.01	28.66
020080050050	31	0.0310727	0.03	28.69
020080060	10	0.0100235	0.01	28.70
020080070	7	0.0070164	0.01	28.70
020080080	3	0.0030070	0.00	28.71
020080080010	24	0.0240563	0.02	28.73
020080080020	8	0.0080188	0.01	28.74
020080090010010	25	0.0250586	0.03	28.76
020080090010020	8	0.0080188	0.01	28.77

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020080090010030	6	0.0060141	0.01	28.78
020080090010040	1	0.0010023	0.00	28.78
020080090020	5	0.0050117	0.01	28.78
020080090030	2	0.0020047	0.00	28.79
020080090040	25	0.0250586	0.03	28.81
020080100010	4	0.0040094	0.00	28.81
020080100020	5	0.0050117	0.01	28.82
020080100030	1	0.0010023	0.00	28.82
020080100040	4	0.0040094	0.00	28.82
020080110	6	0.0060141	0.01	28.83
020080120010	13	0.0130305	0.01	28.84
020080120020	3	0.0030070	0.00	28.85
020080120030010	3	0.0030070	0.00	28.85
020080120030020	4	0.0040094	0.00	28.85
020080120030030010	4	0.0040094	0.00	28.86
020080120030030020	7	0.0070164	0.01	28.86
020080120030030030	10	0.0100235	0.01	28.87
020080130	11	0.0110258	0.01	28.89
020080140	13	0.0130305	0.01	28.90
020080150	5	0.0050117	0.01	28.90
020080160	36	0.0360844	0.04	28.94
020080170	18	0.0180422	0.02	28.96
020080180	3	0.0030070	0.00	28.96
020080190	3	0.0030070	0.00	28.96
020080200	14	0.0140328	0.01	28.98
020080210	1	0.0010023	0.00	28.98
020080220	3	0.0030070	0.00	28.98
020080230	10	0.0100235	0.01	28.99
020080240	16	0.0160375	0.02	29.01

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020080250	4	0.0040094	0.00	29.01
020080260	11	0.0110258	0.01	29.02
020080270	32	0.0320751	0.03	29.05
020090	106	0.1062486	0.11	29.16
020090010	15	0.0150352	0.02	29.18
020090020	3	0.0030070	0.00	29.18
020090030	12	0.0120281	0.01	29.19
020100	80	0.0801876	0.08	29.27
020100010	8	0.0080188	0.01	29.28
020100020	7	0.0070164	0.01	29.29
020100030	3	0.0030070	0.00	29.29
020100040	2	0.0020047	0.00	29.29
020110	105	0.1052463	0.11	29.40
020110010	1	0.0010023	0.00	29.40
020110020	2	0.0020047	0.00	29.40
020110030	2	0.0020047	0.00	29.40
020110040	2	0.0020047	0.00	29.40
020110050	3	0.0030070	0.00	29.41
020120	51	0.0511196	0.05	29.46
020120010	1	0.0010023	0.00	29.46
020120020	3	0.0030070	0.00	29.46
020120030	3	0.0030070	0.00	29.46
020130	72	0.0721689	0.07	29.54
020130010	1	0.0010023	0.00	29.54
020130020	1	0.0010023	0.00	29.54
020130030	7	0.0070164	0.01	29.55
020130040	4	0.0040094	0.00	29.55
020140	94	0.0942205	0.09	29.64
020150	42	0.0420985	0.04	29.69

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020160	53	0.0531243	0.05	29.74
020170	39	0.0390915	0.04	29.78
020180	48	0.0481126	0.05	29.83
020190	40	0.0400938	0.04	29.87
020200	37	0.0370868	0.04	29.90
020210	37	0.0370868	0.04	29.94
020220	42	0.0420985	0.04	29.98
020230	46	0.0461079	0.05	30.03
020240	34	0.0340797	0.03	30.06
020250	43	0.0431009	0.04	30.11
020260	35	0.0350821	0.04	30.14
020270	36	0.0360844	0.04	30.18
020280	37	0.0370868	0.04	30.21
020290	13	0.0130305	0.01	30.23
020300	25	0.0250586	0.03	30.25
021	32	0.0320751	0.03	30.28
021010	5	0.0050117	0.01	30.29
021020	2	0.0020047	0.00	30.29
021030	4	0.0040094	0.00	30.30
030	5322	5.3344827	5.33	35.63
030000550	1	0.0010023	0.00	35.63
030000560	1	0.0010023	0.00	35.63
030000590	1	0.0010023	0.00	35.63
030000600	1	0.0010023	0.00	35.63
030000610	1	0.0010023	0.00	35.64
030000620	1	0.0010023	0.00	35.64
030000630	1	0.0010023	0.00	35.64
030000640	1	0.0010023	0.00	35.64
030000660	1	0.0010023	0.00	35.64

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030000670	1	0.0010023	0.00	35.64
030000680	1	0.0010023	0.00	35.64
030000690	1	0.0010023	0.00	35.64
030000730	1	0.0010023	0.00	35.64
030000740	1	0.0010023	0.00	35.64
030000760	1	0.0010023	0.00	35.65
030000770	1	0.0010023	0.00	35.65
030000780	1	0.0010023	0.00	35.65
030000790	1	0.0010023	0.00	35.65
030000800	1	0.0010023	0.00	35.65
030000810	1	0.0010023	0.00	35.65
030000820	1	0.0010023	0.00	35.65
030000830	1	0.0010023	0.00	35.65
030000840	1	0.0010023	0.00	35.65
030000850	1	0.0010023	0.00	35.65
030000860	1	0.0010023	0.00	35.66
030000890	1	0.0010023	0.00	35.66
030000900	1	0.0010023	0.00	35.66
030000910	1	0.0010023	0.00	35.66
030000930	1	0.0010023	0.00	35.66
030000940	1	0.0010023	0.00	35.66
030000960	1	0.0010023	0.00	35.66
030000970	1	0.0010023	0.00	35.66
030000980	1	0.0010023	0.00	35.66
030000990	1	0.0010023	0.00	35.66
030001000	1	0.0010023	0.00	35.67
030001010	1	0.0010023	0.00	35.67
030001020	1	0.0010023	0.00	35.67
030001040	1	0.0010023	0.00	35.67

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030001050	1	0.0010023	0.00	35.67
030001060	1	0.0010023	0.00	35.67
030001070	1	0.0010023	0.00	35.67
030001080	1	0.0010023	0.00	35.67
030001090	1	0.0010023	0.00	35.67
030001100	1	0.0010023	0.00	35.67
030001110	1	0.0010023	0.00	35.68
030001120	1	0.0010023	0.00	35.68
030001130	1	0.0010023	0.00	35.68
030001140	1	0.0010023	0.00	35.68
030001150	1	0.0010023	0.00	35.68
030001160	1	0.0010023	0.00	35.68
030001170	1	0.0010023	0.00	35.68
030001180	1	0.0010023	0.00	35.68
030001190	1	0.0010023	0.00	35.68
030001200	1	0.0010023	0.00	35.68
030001210	1	0.0010023	0.00	35.69
030001220	1	0.0010023	0.00	35.69
030001230	1	0.0010023	0.00	35.69
030001240	1	0.0010023	0.00	35.69
030001250	1	0.0010023	0.00	35.69
030001260	1	0.0010023	0.00	35.69
030001270	1	0.0010023	0.00	35.69
030001280	1	0.0010023	0.00	35.69
030001290	1	0.0010023	0.00	35.69
030001300	1	0.0010023	0.00	35.69
030001310	1	0.0010023	0.00	35.70
030001320	1	0.0010023	0.00	35.70
030001330	1	0.0010023	0.00	35.70



(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030001370	1	0.0010023	0.00	35.70
030001380	1	0.0010023	0.00	35.70
030001390	1	0.0010023	0.00	35.70
030001400	1	0.0010023	0.00	35.70
030001430	1	0.0010023	0.00	35.70
030001440	1	0.0010023	0.00	35.70
030001470	1	0.0010023	0.00	35.70
030001520	1	0.0010023	0.00	35.71
030010	1090	1.0925566	1.09	36.80
030010010	120	0.1202815	0.12	36.92
030010020	117	0.1172744	0.12	37.04
030010020010	4	0.0040094	0.00	37.04
030010020020	6	0.0060141	0.01	37.05
030010020030	6	0.0060141	0.01	37.05
030010030	73	0.0731712	0.07	37.12
030010030010	17	0.0170399	0.02	37.14
030010030020	8	0.0080188	0.01	37.15
030010030030	4	0.0040094	0.00	37.15
030010030040	6	0.0060141	0.01	37.16
030010030050	1	0.0010023	0.00	37.16
030010031	5	0.0050117	0.01	37.17
030010040	35	0.0350821	0.04	37.20
030010040010	16	0.0160375	0.02	37.22
030010040020	5	0.0050117	0.01	37.22
030010050	19	0.0190446	0.02	37.24
030010060	11	0.0110258	0.01	37.25
030010070	3	0.0030070	0.00	37.26
030010080	3	0.0030070	0.00	37.26
030010150	3	0.0030070	0.00	37.26

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030020	1021	1.0233947	1.02	38.28
030020010	97	0.0972275	0.10	38.38
030020010010	11	0.0110258	0.01	38.39
030020010020	13	0.0130305	0.01	38.41
030020010030	6	0.0060141	0.01	38.41
030020010050	4	0.0040094	0.00	38.42
030020010060	7	0.0070164	0.01	38.42
030020020	118	0.1182768	0.12	38.54
030020020010	4	0.0040094	0.00	38.55
030020020020	2	0.0020047	0.00	38.55
030020020030	3	0.0030070	0.00	38.55
030020020040	7	0.0070164	0.01	38.56
030020020050	3	0.0030070	0.00	38.56
030020020060	1	0.0010023	0.00	38.56
030020020070	2	0.0020047	0.00	38.56
030020020080	2	0.0020047	0.00	38.57
030020020090	5	0.0050117	0.01	38.57
030020020100	1	0.0010023	0.00	38.57
030020030	83	0.0831947	0.08	38.65
030020040	30	0.0300704	0.03	38.68
030020050	21	0.0210493	0.02	38.71
030020060	10	0.0100235	0.01	38.72
030020060010	6	0.0060141	0.01	38.72
030020060020	4	0.0040094	0.00	38.73
030020070	4	0.0040094	0.00	38.73
030020070010	1	0.0010023	0.00	38.73
030020070020	5	0.0050117	0.01	38.74
030020070030	3	0.0030070	0.00	38.74
030020080	9	0.0090211	0.01	38.75

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030020090	6	0.0060141	0.01	38.75
030020170	2	0.0020047	0.00	38.76
030020180	2	0.0020047	0.00	38.76
030020190	12	0.0120281	0.01	38.77
030021	2	0.0020047	0.00	38.77
030030	590	0.5913838	0.59	39.36
030030010	88	0.0882064	0.09	39.45
030030020	82	0.0821923	0.08	39.53
030030030	38	0.0380891	0.04	39.57
030030030010	10	0.0100235	0.01	39.58
030030030020	9	0.0090211	0.01	39.59
030030030030	7	0.0070164	0.01	39.60
030030030040	5	0.0050117	0.01	39.60
030030030050	7	0.0070164	0.01	39.61
030030030060	20	0.0200469	0.02	39.63
030030040	38	0.0380891	0.04	39.67
030030050	17	0.0170399	0.02	39.68
030030060	4	0.0040094	0.00	39.69
030030070	1	0.0010023	0.00	39.69
030030270	2	0.0020047	0.00	39.69
030030280	2	0.0020047	0.00	39.69
030030290	5	0.0050117	0.01	39.70
030030330	4	0.0040094	0.00	39.70
030031	11	0.0110258	0.01	39.71
030032	8	0.0080188	0.01	39.72
030040	359	0.3598420	0.36	40.08
030040010	32	0.0320751	0.03	40.11
030040010010	1	0.0010023	0.00	40.11
030040010020	4	0.0040094	0.00	40.12

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030040010030	3	0.0030070	0.00	40.12
030040010040	1	0.0010023	0.00	40.12
030040010050	2	0.0020047	0.00	40.12
030040010060	1	0.0010023	0.00	40.13
030040010070	1	0.0010023	0.00	40.13
030040010080	3	0.0030070	0.00	40.13
030040020	26	0.0260610	0.03	40.16
030040020010	44	0.0441032	0.04	40.20
030040020020	18	0.0180422	0.02	40.22
030040020030	8	0.0080188	0.01	40.23
030040020040	4	0.0040094	0.00	40.23
030040020050	2	0.0020047	0.00	40.23
030040020060	1	0.0010023	0.00	40.23
030040020070	1	0.0010023	0.00	40.23
030040020080	4	0.0040094	0.00	40.24
030040030	6	0.0060141	0.01	40.24
030040030010	2	0.0020047	0.00	40.25
030040030020	5	0.0050117	0.01	40.25
030040030030	2	0.0020047	0.00	40.25
030040040	12	0.0120281	0.01	40.27
030040100	2	0.0020047	0.00	40.27
030041	2	0.0020047	0.00	40.27
030050	186	0.1864363	0.19	40.46
030050010	59	0.0591384	0.06	40.51
030050020	44	0.0441032	0.04	40.56
030050030	13	0.0130305	0.01	40.57
030050040	11	0.0110258	0.01	40.58
030050050	3	0.0030070	0.00	40.59
030060	137	0.1373213	0.14	40.72

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030060010010	1	0.0010023	0.00	40.72
030060010020	3	0.0030070	0.00	40.73
030060010030	4	0.0040094	0.00	40.73
030060010040	1	0.0010023	0.00	40.73
030060010050	1	0.0010023	0.00	40.73
030060010060	1	0.0010023	0.00	40.73
030060020	3	0.0030070	0.00	40.74
030070	75	0.0751759	0.08	40.81
030070010	83	0.0831947	0.08	40.90
030070010010	4	0.0040094	0.00	40.90
030070010020	3	0.0030070	0.00	40.90
030070020	11	0.0110258	0.01	40.91
030070020010	8	0.0080188	0.01	40.92
030070020020	5	0.0050117	0.01	40.93
030070030	3	0.0030070	0.00	40.93
030070030010	2	0.0020047	0.00	40.93
030070030020	8	0.0080188	0.01	40.94
030070030030	2	0.0020047	0.00	40.94
030070030040	1	0.0010023	0.00	40.94
030070040	6	0.0060141	0.01	40.95
030080	38	0.0380891	0.04	40.99
030080010	56	0.0561313	0.06	41.04
030080020	24	0.0240563	0.02	41.07
030081	2	0.0020047	0.00	41.07
030090	57	0.0571337	0.06	41.13
030100	46	0.0461079	0.05	41.17
030100010	6	0.0060141	0.01	41.18
030100020	4	0.0040094	0.00	41.18
030100030	4	0.0040094	0.00	41.19

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030110	29	0.0290680	0.03	41.22
030120	36	0.0360844	0.04	41.25
030130	31	0.0310727	0.03	41.28
030130010	13	0.0130305	0.01	41.30
030130020	12	0.0120281	0.01	41.31
030130030	2	0.0020047	0.00	41.31
030130040	2	0.0020047	0.00	41.31
030140	26	0.0260610	0.03	41.34
030150	48	0.0481126	0.05	41.39
030160	1	0.0010023	0.00	41.39
030170	4	0.0040094	0.00	41.39
030220	1	0.0010023	0.00	41.39
030260	1	0.0010023	0.00	41.39
030280	14	0.0140328	0.01	41.41
030970	57	0.0571337	0.06	41.46
030971	5	0.0050117	0.01	41.47
030980	1	0.0010023	0.00	41.47
030990	1	0.0010023	0.00	41.47
030991	1	0.0010023	0.00	41.47
030992	1	0.0010023	0.00	41.47
030993	1	0.0010023	0.00	41.47
031	9	0.0090211	0.01	41.48
040	4232	4.2419261	4.24	45.72
040001570	1	0.0010023	0.00	45.73
040001630	1	0.0010023	0.00	45.73
040001640	1	0.0010023	0.00	45.73
040001650	1	0.0010023	0.00	45.73
040001660	1	0.0010023	0.00	45.73
040001670	1	0.0010023	0.00	45.73

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
040001680	1	0.0010023	0.00	45.73
040001690	1	0.0010023	0.00	45.73
040001700	1	0.0010023	0.00	45.73
040001710	1	0.0010023	0.00	45.74
040001720	1	0.0010023	0.00	45.74
040001730	1	0.0010023	0.00	45.74
040001740	1	0.0010023	0.00	45.74
040001750	1	0.0010023	0.00	45.74
040001760	1	0.0010023	0.00	45.74
040001770	1	0.0010023	0.00	45.74
040001780	1	0.0010023	0.00	45.74
040001790	1	0.0010023	0.00	45.74
040001800	1	0.0010023	0.00	45.74
040001810	1	0.0010023	0.00	45.75
040001820	1	0.0010023	0.00	45.75
040001830	1	0.0010023	0.00	45.75
040001840	1	0.0010023	0.00	45.75
040001850	1	0.0010023	0.00	45.75
040001860	1	0.0010023	0.00	45.75
040001870	1	0.0010023	0.00	45.75
040001880	1	0.0010023	0.00	45.75
040001890	1	0.0010023	0.00	45.75
040001900	1	0.0010023	0.00	45.75
040001910	1	0.0010023	0.00	45.76
040001920	1	0.0010023	0.00	45.76
040001930	1	0.0010023	0.00	45.76
040001940	1	0.0010023	0.00	45.76
040001950	1	0.0010023	0.00	45.76
040001960	1	0.0010023	0.00	45.76

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
040001970	1	0.0010023	0.00	45.76
040001980	1	0.0010023	0.00	45.76
040001990	1	0.0010023	0.00	45.76
040002000	1	0.0010023	0.00	45.76
040002010	1	0.0010023	0.00	45.77
040002020	1	0.0010023	0.00	45.77
040002030	1	0.0010023	0.00	45.77
040002070	1	0.0010023	0.00	45.77
040002080	1	0.0010023	0.00	45.77
040002090	1	0.0010023	0.00	45.77
040002100	1	0.0010023	0.00	45.77
040002120	1	0.0010023	0.00	45.77
040002130	1	0.0010023	0.00	45.77
040002140	1	0.0010023	0.00	45.77
040002150	1	0.0010023	0.00	45.78
040002170	1	0.0010023	0.00	45.78
040002180	1	0.0010023	0.00	45.78
040010	475	0.4761141	0.48	46.25
040010010	55	0.0551290	0.06	46.31
040010020	75	0.0751759	0.08	46.38
040010020010	7	0.0070164	0.01	46.39
040010020020	20	0.0200469	0.02	46.41
040010020030	1	0.0010023	0.00	46.41
040010020040	3	0.0030070	0.00	46.41
040010030	29	0.0290680	0.03	46.44
040010030010	8	0.0080188	0.01	46.45
040010030020	4	0.0040094	0.00	46.46
040010030030	7	0.0070164	0.01	46.46
040010040	19	0.0190446	0.02	46.48



(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
040010050	27	0.0270633	0.03	46.51
040010060	1	0.0010023	0.00	46.51
040010060010	28	0.0280657	0.03	46.54
040010060020010	6	0.0060141	0.01	46.54
040010060020020	1	0.0010023	0.00	46.54
040010060020030010	7	0.0070164	0.01	46.55
040010060020030020	28	0.0280657	0.03	46.58
040010060020030030	20	0.0200469	0.02	46.60
040010060020030040	12	0.0120281	0.01	46.61
040010060020030050	36	0.0360844	0.04	46.65
040010060020040	1	0.0010023	0.00	46.65
040010060030	6	0.0060141	0.01	46.66
040010070	1	0.0010023	0.00	46.66
040010070010	5	0.0050117	0.01	46.66
040010070011	2	0.0020047	0.00	46.66
040010070020010	1	0.0010023	0.00	46.66
040010070020020	13	0.0130305	0.01	46.68
040010070020030	4	0.0040094	0.00	46.68
040010070020040	4	0.0040094	0.00	46.69
040010070020050	3	0.0030070	0.00	46.69
040010070030	1	0.0010023	0.00	46.69
040010080	3	0.0030070	0.00	46.69
040020	415	0.4159734	0.42	47.11
040020010	139	0.1393260	0.14	47.25
040020010010	5	0.0050117	0.01	47.25
040020010020	13	0.0130305	0.01	47.27
040020010030	8	0.0080188	0.01	47.27
040020010040	11	0.0110258	0.01	47.28
040020010050	6	0.0060141	0.01	47.29

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
040020010060	7	0.0070164	0.01	47.30
040020010070	4	0.0040094	0.00	47.30
040020010080	1	0.0010023	0.00	47.30
040020020	111	0.1112603	0.11	47.41
040020020010	29	0.0290680	0.03	47.44
040020020020	8	0.0080188	0.01	47.45
040020020030	12	0.0120281	0.01	47.46
040020020040	15	0.0150352	0.02	47.48
040020020050	7	0.0070164	0.01	47.49
040020030	62	0.0621454	0.06	47.55
040020030010	28	0.0280657	0.03	47.58
040020030020	5	0.0050117	0.01	47.58
040020030030	3	0.0030070	0.00	47.58
040020030040	1	0.0010023	0.00	47.58
040020040	37	0.0370868	0.04	47.62
040020050	82	0.0821923	0.08	47.70
040020060	16	0.0160375	0.02	47.72
040020070	11	0.0110258	0.01	47.73
040020070010	27	0.0270633	0.03	47.76
040020070020	6	0.0060141	0.01	47.76
040020080	7	0.0070164	0.01	47.77
040020090	5	0.0050117	0.01	47.78
040030	381	0.3818936	0.38	48.16
040030010	28	0.0280657	0.03	48.19
040030010010	38	0.0380891	0.04	48.22
040030010020	52	0.0521220	0.05	48.28
040030010030	21	0.0210493	0.02	48.30
040030010040	2	0.0020047	0.00	48.30
040030010050	7	0.0070164	0.01	48.31

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
040030010060	14	0.0140328	0.01	48.32
040030020	61	0.0611431	0.06	48.38
040030020010	9	0.0090211	0.01	48.39
040030020020	2	0.0020047	0.00	48.39
040030030	48	0.0481126	0.05	48.44
040030040	11	0.0110258	0.01	48.45
040030040010	1	0.0010023	0.00	48.45
040030040020	1	0.0010023	0.00	48.45
040030050	2	0.0020047	0.00	48.46
040040	182	0.1824269	0.18	48.64
040040010	71	0.0711665	0.07	48.71
040040020	14	0.0140328	0.01	48.72
040040030	5	0.0050117	0.01	48.73
040050	75	0.0751759	0.08	48.80
040050010	5	0.0050117	0.01	48.81
040050020	3	0.0030070	0.00	48.81
040050030	7	0.0070164	0.01	48.82
040060	43	0.0431009	0.04	48.86
040060010	6	0.0060141	0.01	48.87
040060020	4	0.0040094	0.00	48.87
040060050	5	0.0050117	0.01	48.88
040070	10	0.0100235	0.01	48.89
040080	17	0.0170399	0.02	48.90
040090	33	0.0330774	0.03	48.94
040100	16	0.0160375	0.02	48.95
040110	5	0.0050117	0.01	48.96
040120	6	0.0060141	0.01	48.96
040130	5	0.0050117	0.01	48.97
040140	2	0.0020047	0.00	48.97

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
040160	14	0.0140328	0.01	48.98
040170	6	0.0060141	0.01	48.99
040190	1	0.0010023	0.00	48.99
040200	1	0.0010023	0.00	48.99
040390	1	0.0010023	0.00	48.99
040400	8	0.0080188	0.01	49.00
040410	1	0.0010023	0.00	49.00
040420	1	0.0010023	0.00	49.00
040430	1	0.0010023	0.00	49.00
040440	1	0.0010023	0.00	49.01
040450	1	0.0010023	0.00	49.01
040460	1	0.0010023	0.00	49.01
040470	1	0.0010023	0.00	49.01
040480	1	0.0010023	0.00	49.01
040500	1	0.0010023	0.00	49.01
040510	1	0.0010023	0.00	49.01
040520	1	0.0010023	0.00	49.01
041	31	0.0310727	0.03	49.04
041010	9	0.0090211	0.01	49.05
041020	5	0.0050117	0.01	49.06
041030	2	0.0020047	0.00	49.06
041040	10	0.0100235	0.01	49.07
041451	3	0.0030070	0.00	49.07
041452	3	0.0030070	0.00	49.08
041454	14	0.0140328	0.01	49.09
042	2	0.0020047	0.00	49.09
042010	1	0.0010023	0.00	49.09
042020	9	0.0090211	0.01	49.10
042030	1	0.0010023	0.00	49.10

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
042040	11	0.0110258	0.01	49.11
050	3090	3.0972476	3.10	52.21
050002190	1	0.0010023	0.00	52.21
050002200	1	0.0010023	0.00	52.21
050002210	1	0.0010023	0.00	52.21
050002220	1	0.0010023	0.00	52.22
050002230	1	0.0010023	0.00	52.22
050002231	1	0.0010023	0.00	52.22
050002240	3	0.0030070	0.00	52.22
050002250	1	0.0010023	0.00	52.22
050002260	1	0.0010023	0.00	52.22
050002270	1	0.0010023	0.00	52.22
050002280	1	0.0010023	0.00	52.22
050002290	55	0.0551290	0.06	52.28
050010	327	0.3277670	0.33	52.61
050010010	34	0.0340797	0.03	52.64
050010020	39	0.0390915	0.04	52.68
050010030	16	0.0160375	0.02	52.70
050010040	42	0.0420985	0.04	52.74
050010050	34	0.0340797	0.03	52.77
050010050010	1	0.0010023	0.00	52.77
050010050020	1	0.0010023	0.00	52.77
050010050030	2	0.0020047	0.00	52.78
050010050040	2	0.0020047	0.00	52.78
050010060	5	0.0050117	0.01	52.78
050010070	3	0.0030070	0.00	52.79
050010080	4	0.0040094	0.00	52.79
050010090	5	0.0050117	0.01	52.80
050010100010	3	0.0030070	0.00	52.80

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
050010100020	3	0.0030070	0.00	52.80
050010100030	1	0.0010023	0.00	52.80
050010110010	1	0.0010023	0.00	52.80
050010110020	19	0.0190446	0.02	52.82
050010110030	13	0.0130305	0.01	52.84
050010110040	8	0.0080188	0.01	52.84
050010110050	10	0.0100235	0.01	52.85
050010110060	1	0.0010023	0.00	52.85
050010120	9	0.0090211	0.01	52.86
050011	5	0.0050117	0.01	52.87
050020	306	0.3067177	0.31	53.18
050020010	72	0.0721689	0.07	53.25
050020010010	12	0.0120281	0.01	53.26
050020010010010	17	0.0170399	0.02	53.28
050020010010020	15	0.0150352	0.02	53.29
050020010010030	14	0.0140328	0.01	53.31
050020010020	12	0.0120281	0.01	53.32
050020020	38	0.0380891	0.04	53.36
050020020010	5	0.0050117	0.01	53.36
050020020020	15	0.0150352	0.02	53.38
050020020030	31	0.0310727	0.03	53.41
050020020040	28	0.0280657	0.03	53.44
050020020050	4	0.0040094	0.00	53.44
050020030	54	0.0541267	0.05	53.49
050020040	53	0.0531243	0.05	53.55
050020040010	28	0.0280657	0.03	53.57
050020040020	16	0.0160375	0.02	53.59
050020040030	8	0.0080188	0.01	53.60
050020050	17	0.0170399	0.02	53.62

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
050020060	11	0.0110258	0.01	53.63
050020070	4	0.0040094	0.00	53.63
050020080	2	0.0020047	0.00	53.63
050020090	2	0.0020047	0.00	53.63
050020100	2	0.0020047	0.00	53.64
050020110	9	0.0090211	0.01	53.65
050030	206	0.2064832	0.21	53.85
050030010	47	0.0471102	0.05	53.90
050030020	49	0.0491149	0.05	53.95
050030030	54	0.0541267	0.05	54.00
050030040	48	0.0481126	0.05	54.05
050030050	5	0.0050117	0.01	54.06
050030060	32	0.0320751	0.03	54.09
050030070	23	0.0230539	0.02	54.11
050030080	9	0.0090211	0.01	54.12
050040	110	0.1102580	0.11	54.23
050040010	9	0.0090211	0.01	54.24
050040020	17	0.0170399	0.02	54.26
050040030	8	0.0080188	0.01	54.26
050050	83	0.0831947	0.08	54.35
050060	20	0.0200469	0.02	54.37
050070	23	0.0230539	0.02	54.39
050080	22	0.0220516	0.02	54.41
050090	16	0.0160375	0.02	54.43
050460	6	0.0060141	0.01	54.43
050660	1	0.0010023	0.00	54.44
051	44	0.0441032	0.04	54.48
052	7	0.0070164	0.01	54.49
053	3	0.0030070	0.00	54.49

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
060	2329	2.3344626	2.33	56.82
060002300	1	0.0010023	0.00	56.82
060002310	10	0.0100235	0.01	56.83
060002320	12	0.0120281	0.01	56.85
060002330010	13	0.0130305	0.01	56.86
060002330020	7	0.0070164	0.01	56.87
060002340	12	0.0120281	0.01	56.88
060002350	2	0.0020047	0.00	56.88
060002360	3	0.0030070	0.00	56.88
060002370	2	0.0020047	0.00	56.89
060010	194	0.1944550	0.19	57.08
060010010	9	0.0090211	0.01	57.09
060010020	23	0.0230539	0.02	57.11
060010030	11	0.0110258	0.01	57.12
060010040	12	0.0120281	0.01	57.14
060010050	5	0.0050117	0.01	57.14
060010060	3	0.0030070	0.00	57.14
060010070	6	0.0060141	0.01	57.15
060020	189	0.1894433	0.19	57.34
060020010	25	0.0250586	0.03	57.36
060020020	23	0.0230539	0.02	57.39
060020030	2	0.0020047	0.00	57.39
060020030010	4	0.0040094	0.00	57.39
060020030020	24	0.0240563	0.02	57.42
060020030030	13	0.0130305	0.01	57.43
060020040	2	0.0020047	0.00	57.43
060020050	3	0.0030070	0.00	57.44
060020060	1	0.0010023	0.00	57.44
060021	4	0.0040094	0.00	57.44



(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
060030	118	0.1182768	0.12	57.56
060030010	23	0.0230539	0.02	57.58
060030020	20	0.0200469	0.02	57.60
060030030	6	0.0060141	0.01	57.61
060030040	2	0.0020047	0.00	57.61
060030050	12	0.0120281	0.01	57.62
060030060	1	0.0010023	0.00	57.62
060040	58	0.0581360	0.06	57.68
060040010	7	0.0070164	0.01	57.69
060040020	6	0.0060141	0.01	57.69
060050	3	0.0030070	0.00	57.70
060101	6	0.0060141	0.01	57.70
060102	5	0.0050117	0.01	57.71
060110	3	0.0030070	0.00	57.71
060670	1	0.0010023	0.00	57.71
061	17	0.0170399	0.02	57.73
070	1515	1.5185534	1.52	59.25
070002390	1	0.0010023	0.00	59.25
070002400	7	0.0070164	0.01	59.26
070002420	2	0.0020047	0.00	59.26
070002430	1	0.0010023	0.00	59.26
070002440	1	0.0010023	0.00	59.26
070002460	1	0.0010023	0.00	59.26
070002461	4	0.0040094	0.00	59.26
070002462	2	0.0020047	0.00	59.27
070002463	1	0.0010023	0.00	59.27
070002470	19	0.0190446	0.02	59.29
070002471	2	0.0020047	0.00	59.29
070002480010	2	0.0020047	0.00	59.29

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
070002480020	9	0.0090211	0.01	59.30
070002480030	6	0.0060141	0.01	59.31
070002480040	4	0.0040094	0.00	59.31
070002490	3	0.0030070	0.00	59.31
070002500	10	0.0100235	0.01	59.32
070002510	2	0.0020047	0.00	59.32
070002520	1	0.0010023	0.00	59.33
070002530	3	0.0030070	0.00	59.33
070010	196	0.1964597	0.20	59.53
070010010	10	0.0100235	0.01	59.54
070010020	10	0.0100235	0.01	59.55
070010030	7	0.0070164	0.01	59.55
070010040	7	0.0070164	0.01	59.56
070010050	5	0.0050117	0.01	59.56
070010060	1	0.0010023	0.00	59.57
070010070	1	0.0010023	0.00	59.57
070011	1	0.0010023	0.00	59.57
070020	175	0.1754105	0.18	59.74
070020010	13	0.0130305	0.01	59.76
070020020	20	0.0200469	0.02	59.78
070020020010	3	0.0030070	0.00	59.78
070020020020	6	0.0060141	0.01	59.78
070020020030	1	0.0010023	0.00	59.79
070020030	13	0.0130305	0.01	59.80
070020040	1	0.0010023	0.00	59.80
070030	86	0.0862017	0.09	59.89
070030010	7	0.0070164	0.01	59.89
070030020	3	0.0030070	0.00	59.90
070030020010	8	0.0080188	0.01	59.90

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
070030020020	2	0.0020047	0.00	59.91
070030030	1	0.0010023	0.00	59.91
070030040	1	0.0010023	0.00	59.91
070040	34	0.0340797	0.03	59.94
070050	29	0.0290680	0.03	59.97
070060	6	0.0060141	0.01	59.98
070070	2	0.0020047	0.00	59.98
070080	5	0.0050117	0.01	59.98
070090	6	0.0060141	0.01	59.99
070680	1	0.0010023	0.00	59.99
070690	1	0.0010023	0.00	59.99
071	3	0.0030070	0.00	60.00
080	1013	1.0153760	1.02	61.01
080010	233	0.2335465	0.23	61.24
080010010	8	0.0080188	0.01	61.25
080010020	24	0.0240563	0.02	61.28
080010030	14	0.0140328	0.01	61.29
080010040	7	0.0070164	0.01	61.30
080010050	6	0.0060141	0.01	61.30
080020	146	0.1463424	0.15	61.45
080020010	15	0.0150352	0.02	61.46
080020020	8	0.0080188	0.01	61.47
080020020010	6	0.0060141	0.01	61.48
080020020020	25	0.0250586	0.03	61.50
080020020030	55	0.0551290	0.06	61.56
080020030	14	0.0140328	0.01	61.57
080020040	12	0.0120281	0.01	61.59
080020050	7	0.0070164	0.01	61.59
080020060	8	0.0080188	0.01	61.60

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
080030	69	0.0691618	0.07	61.67
080030010	18	0.0180422	0.02	61.69
080030020	4	0.0040094	0.00	61.69
080030020010	3	0.0030070	0.00	61.69
080030020020	3	0.0030070	0.00	61.70
080030020030	2	0.0020047	0.00	61.70
080030020040	1	0.0010023	0.00	61.70
080030020050	1	0.0010023	0.00	61.70
080030030	4	0.0040094	0.00	61.71
080030040	1	0.0010023	0.00	61.71
080040	54	0.0541267	0.05	61.76
080050	52	0.0521220	0.05	61.81
080060	26	0.0260610	0.03	61.84
080060010	4	0.0040094	0.00	61.84
080060020	3	0.0030070	0.00	61.85
080060030	5	0.0050117	0.01	61.85
080060040	2	0.0020047	0.00	61.85
080070	3	0.0030070	0.00	61.86
080080	7	0.0070164	0.01	61.86
081	5	0.0050117	0.01	61.87
090	1086	1.0885472	1.09	62.96
090010	43	0.0431009	0.04	63.00
090010010	25	0.0250586	0.03	63.02
090010020	12	0.0120281	0.01	63.04
090020	49	0.0491149	0.05	63.09
090020010	9	0.0090211	0.01	63.09
090020020	20	0.0200469	0.02	63.11
090020030	7	0.0070164	0.01	63.12
090030	69	0.0691618	0.07	63.19

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
090040	32	0.0320751	0.03	63.22
090040010	11	0.0110258	0.01	63.23
090040020	3	0.0030070	0.00	63.24
090050	20	0.0200469	0.02	63.26
090060	1	0.0010023	0.00	63.26
091	11	0.0110258	0.01	63.27
100	583	0.5843674	0.58	63.85
100010	13	0.0130305	0.01	63.87
100010010	5	0.0050117	0.01	63.87
100010020	10	0.0100235	0.01	63.88
100020	25	0.0250586	0.03	63.91
100020010	20	0.0200469	0.02	63.93
100020020	13	0.0130305	0.01	63.94
100020030	3	0.0030070	0.00	63.94
100030	27	0.0270633	0.03	63.97
100030010	4	0.0040094	0.00	63.97
100030020	4	0.0040094	0.00	63.98
100040	6	0.0060141	0.01	63.98
100050	14	0.0140328	0.01	64.00
100051	1	0.0010023	0.00	64.00
100060	8	0.0080188	0.01	64.01
100070	1	0.0010023	0.00	64.01
100080	2	0.0020047	0.00	64.01
100090	4	0.0040094	0.00	64.01
100100	1	0.0010023	0.00	64.01
101	11	0.0110258	0.01	64.03
102	11	0.0110258	0.01	64.04
102010	1	0.0010023	0.00	64.04
102020	2	0.0020047	0.00	64.04

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
102030010	10	0.0100235	0.01	64.05
102030020010	4	0.0040094	0.00	64.05
102030020020	6	0.0060141	0.01	64.06
102030020030	1	0.0010023	0.00	64.06
102030030	5	0.0050117	0.01	64.07
103	1	0.0010023	0.00	64.07
103010	1	0.0010023	0.00	64.07
103020	1	0.0010023	0.00	64.07
103030	1	0.0010023	0.00	64.07
103040	1	0.0010023	0.00	64.07
103050	1	0.0010023	0.00	64.07
103060	1	0.0010023	0.00	64.07
103070	1	0.0010023	0.00	64.07
103080	1	0.0010023	0.00	64.07
103090	1	0.0010023	0.00	64.08
103100	1	0.0010023	0.00	64.08
103110	1	0.0010023	0.00	64.08
103120	1	0.0010023	0.00	64.08
103130	1	0.0010023	0.00	64.08
104	1	0.0010023	0.00	64.08
105	1	0.0010023	0.00	64.08
105010	1	0.0010023	0.00	64.08
106	1	0.0010023	0.00	64.08
107	1	0.0010023	0.00	64.08
107010	1	0.0010023	0.00	64.09
108	1	0.0010023	0.00	64.09
109	1	0.0010023	0.00	64.09
110	427	0.4280015	0.43	64.52
110010	61	0.0611431	0.06	64.58

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
110010010	5	0.0050117	0.01	64.58
110010020	3	0.0030070	0.00	64.59
110010030	5	0.0050117	0.01	64.59
110020	19	0.0190446	0.02	64.61
110020010	14	0.0140328	0.01	64.62
110020020	5	0.0050117	0.01	64.63
110020030	5	0.0050117	0.01	64.63
110030	25	0.0250586	0.03	64.66
110040	7	0.0070164	0.01	64.67
110040010	3	0.0030070	0.00	64.67
110040020	5	0.0050117	0.01	64.67
110050	8	0.0080188	0.01	64.68
110050010	10	0.0100235	0.01	64.69
110050020	4	0.0040094	0.00	64.70
110050020010	4	0.0040094	0.00	64.70
110050020020	6	0.0060141	0.01	64.71
110050020030	2	0.0020047	0.00	64.71
110050020040	2	0.0020047	0.00	64.71
110050030	2	0.0020047	0.00	64.71
110050040	5	0.0050117	0.01	64.72
110050050	2	0.0020047	0.00	64.72
110050060	2	0.0020047	0.00	64.72
110050070	2	0.0020047	0.00	64.72
110050080	1	0.0010023	0.00	64.72
110060	10	0.0100235	0.01	64.73
110060010	9	0.0090211	0.01	64.74
110060020	5	0.0050117	0.01	64.75
110070	6	0.0060141	0.01	64.75
110070010	2	0.0020047	0.00	64.76

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
110070020	6	0.0060141	0.01	64.76
110080	13	0.0130305	0.01	64.77
111	1	0.0010023	0.00	64.78
112	2	0.0020047	0.00	64.78
120	183	0.1834292	0.18	64.96
120010	43	0.0431009	0.04	65.00
120020	24	0.0240563	0.02	65.03
120030	24	0.0240563	0.02	65.05
130	139	0.1393260	0.14	65.19
130010	18	0.0180422	0.02	65.21
130020	24	0.0240563	0.02	65.23
130030	10	0.0100235	0.01	65.24
130040	9	0.0090211	0.01	65.25
130050	26	0.0260610	0.03	65.28
130060	2	0.0020047	0.00	65.28
130070	3	0.0030070	0.00	65.28
130080	5	0.0050117	0.01	65.29
140	108	0.1082533	0.11	65.40
150	94	0.0942205	0.09	65.49
150010	8	0.0080188	0.01	65.50
150020	8	0.0080188	0.01	65.51
150030	1	0.0010023	0.00	65.51
160	109	0.1092557	0.11	65.62
160010	12	0.0120281	0.01	65.63
160020	8	0.0080188	0.01	65.64
160030	11	0.0110258	0.01	65.65
170	26	0.0260610	0.03	65.67
171	5	0.0050117	0.01	65.68
172	2	0.0020047	0.00	65.68



(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
174	2	0.0020047	0.00	65.68
180	28	0.0280657	0.03	65.71
180010	35	0.0350821	0.04	65.75
180020	4	0.0040094	0.00	65.75
180030	5	0.0050117	0.01	65.76
180040	1	0.0010023	0.00	65.76
180050	1	0.0010023	0.00	65.76
180060	1	0.0010023	0.00	65.76
180070	1	0.0010023	0.00	65.76
180080	1	0.0010023	0.00	65.76
180090	1	0.0010023	0.00	65.76
180100	1	0.0010023	0.00	65.76
180110	1	0.0010023	0.00	65.76
180120	1	0.0010023	0.00	65.76
180130	1	0.0010023	0.00	65.77
180140	3	0.0030070	0.00	65.77
180160	1	0.0010023	0.00	65.77
180170	1	0.0010023	0.00	65.77
180180	1	0.0010023	0.00	65.77
180190	2	0.0020047	0.00	65.77
190	21	0.0210493	0.02	65.79
200	13	0.0130305	0.01	65.81
210	21	0.0210493	0.02	65.83
220	22	0.0220516	0.02	65.85
230	25	0.0250586	0.03	65.88
231	1	0.0010023	0.00	65.88
240	14	0.0140328	0.01	65.89
250	13	0.0130305	0.01	65.90
260	5	0.0050117	0.01	65.91

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
270	3	0.0030070	0.00	65.91
280	4	0.0040094	0.00	65.92
281	16	0.0160375	0.02	65.93
290	5	0.0050117	0.01	65.94
300	3	0.0030070	0.00	65.94
301	1	0.0010023	0.00	65.94
310	5	0.0050117	0.01	65.95
320	2	0.0020047	0.00	65.95
330	16	0.0160375	0.02	65.96
340	6	0.0060141	0.01	65.97
350	6	0.0060141	0.01	65.98
360	2	0.0020047	0.00	65.98
370	2	0.0020047	0.00	65.98
371	1	0.0010023	0.00	65.98
380	4	0.0040094	0.00	65.99
390	2	0.0020047	0.00	65.99
400	2	0.0020047	0.00	65.99
410	3	0.0030070	0.00	65.99
420	4	0.0040094	0.00	66.00
421	1	0.0010023	0.00	66.00
430	2	0.0020047	0.00	66.00
440	5	0.0050117	0.01	66.00
450	1	0.0010023	0.00	66.01
460	5	0.0050117	0.01	66.01
460460	3	0.0030070	0.00	66.01
460490	2	0.0020047	0.00	66.02
460510	8	0.0080188	0.01	66.02
460590	4	0.0040094	0.00	66.03
460620	1	0.0010023	0.00	66.03

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
470	3	0.0030070	0.00	66.03
480	3	0.0030070	0.00	66.03
491	3	0.0030070	0.00	66.04
510	2	0.0020047	0.00	66.04
520	6	0.0060141	0.01	66.05
530	2	0.0020047	0.00	66.05
550	1	0.0010023	0.00	66.05
560	2	0.0020047	0.00	66.05
580	1	0.0010023	0.00	66.05
590	1	0.0010023	0.00	66.05
600	2	0.0020047	0.00	66.05
610	2	0.0020047	0.00	66.06
620	8	0.0080188	0.01	66.06
630	7	0.0070164	0.01	66.07
640	1	0.0010023	0.00	66.07
650	1	0.0010023	0.00	66.07
680	1	0.0010023	0.00	66.07
681	2	0.0020047	0.00	66.08
690	2	0.0020047	0.00	66.08
700	2	0.0020047	0.00	66.08
700010	1	0.0010023	0.00	66.08
700020	1	0.0010023	0.00	66.08
700030	1	0.0010023	0.00	66.08
700031	1	0.0010023	0.00	66.08
700032	1	0.0010023	0.00	66.09
700040	1	0.0010023	0.00	66.09
700050	1	0.0010023	0.00	66.09
700060	1	0.0010023	0.00	66.09
700070	1	0.0010023	0.00	66.09

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
700080	1	0.0010023	0.00	66.09
700090	1	0.0010023	0.00	66.09
700100	1	0.0010023	0.00	66.09
700110	1	0.0010023	0.00	66.09
700120	1	0.0010023	0.00	66.09
700130	1	0.0010023	0.00	66.10
700140	1	0.0010023	0.00	66.10
700150	1	0.0010023	0.00	66.10
700160	1	0.0010023	0.00	66.10
700170	1	0.0010023	0.00	66.10
700180	1	0.0010023	0.00	66.10
710	2	0.0020047	0.00	66.10
720	7	0.0070164	0.01	66.11
730	1	0.0010023	0.00	66.11
730740	5	0.0050117	0.01	66.12
731	6	0.0060141	0.01	66.12
740	6	0.0060141	0.01	66.13
780	2	0.0020047	0.00	66.13
800	1	0.0010023	0.00	66.13
810	1	0.0010023	0.00	66.13
820	1	0.0010023	0.00	66.13
830	5	0.0050117	0.01	66.14
840	4	0.0040094	0.00	66.14
870	1	0.0010023	0.00	66.14
880	5	0.0050117	0.01	66.15
911	1	0.0010023	0.00	66.15
920	1	0.0010023	0.00	66.15
930	1	0.0010023	0.00	66.15
950	1	0.0010023	0.00	66.15

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
960	2	0.0020047	0.00	66.15
980	1	0.0010023	0.00	66.15
990	1	0.0010023	0.00	66.16
991	3	0.0030070	0.00	66.16
NA	33762	33.8411884	33.84	100.00
Total	99766	100.0000000	100.00	100.00

---

Frequency Table for Variable: ausfertigung\_jahr

---

123 unique value(s) detected.

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1869	146	0.1463424	0.15	0.15
1871	562	0.5633182	0.56	0.71
1877	99	0.0992322	0.10	0.81
1884	10	0.0100235	0.01	0.82
1886	3	0.0030070	0.00	0.82
1887	4	0.0040094	0.00	0.83
1889	215	0.2155043	0.22	1.04
1892	115	0.1152697	0.12	1.16
1894	4	0.0040094	0.00	1.16
1895	74	0.0741736	0.07	1.23
1896	2868	2.8747269	2.87	4.11
1897	1132	1.1346551	1.13	5.24
1898	3	0.0030070	0.00	5.25
1899	2	0.0020047	0.00	5.25
1901	51	0.0511196	0.05	5.30
1903	1	0.0010023	0.00	5.30
1906	1	0.0010023	0.00	5.30

(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1907	26	0.0260610	0.03	5.33
1908	7	0.0070164	0.01	5.34
1909	120	0.1202815	0.12	5.46
1910	11	0.0110258	0.01	5.47
1911	47	0.0471102	0.05	5.51
1913	45	0.0451055	0.05	5.56
1919	99	0.0992322	0.10	5.66
1920	7	0.0070164	0.01	5.67
1921	40	0.0400938	0.04	5.71
1922	206	0.2064832	0.21	5.91
1923	15	0.0150352	0.02	5.93
1924	2	0.0020047	0.00	5.93
1925	2	0.0020047	0.00	5.93
1926	31	0.0310727	0.03	5.96
1927	32	0.0320751	0.03	5.99
1928	8	0.0080188	0.01	6.00
1929	8	0.0080188	0.01	6.01
1930	14	0.0140328	0.01	6.02
1931	46	0.0461079	0.05	6.07
1933	196	0.1964597	0.20	6.27
1934	515	0.5162079	0.52	6.78
1935	157	0.1573682	0.16	6.94
1936	284	0.2846661	0.28	7.22
1937	321	0.3217529	0.32	7.55
1938	42	0.0420985	0.04	7.59
1939	90	0.0902111	0.09	7.68
1940	215	0.2155043	0.22	7.89
1941	25	0.0250586	0.03	7.92
1942	20	0.0200469	0.02	7.94

(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1943	24	0.0240563	0.02	7.96
1944	12	0.0120281	0.01	7.98
1947	19	0.0190446	0.02	7.99
1948	73	0.0731712	0.07	8.07
1949	314	0.3147365	0.31	8.38
1950	2093	2.0979091	2.10	10.48
1951	585	0.5863721	0.59	11.07
1952	445	0.4460437	0.45	11.51
1953	1816	1.8202594	1.82	13.33
1954	210	0.2104926	0.21	13.54
1955	303	0.3037107	0.30	13.85
1956	700	0.7016418	0.70	14.55
1957	991	0.9933244	0.99	15.54
1958	150	0.1503518	0.15	15.69
1959	749	0.7507568	0.75	16.44
1960	1103	1.1055871	1.11	17.55
1961	1283	1.2860093	1.29	18.84
1962	247	0.2475793	0.25	19.08
1963	337	0.3377904	0.34	19.42
1964	332	0.3327787	0.33	19.75
1965	1481	1.4844737	1.48	21.24
1966	570	0.5713369	0.57	21.81
1967	330	0.3307740	0.33	22.14
1968	478	0.4791211	0.48	22.62
1969	722	0.7236934	0.72	23.34
1970	439	0.4400297	0.44	23.78
1971	776	0.7778201	0.78	24.56
1972	825	0.8269350	0.83	25.39
1973	239	0.2395606	0.24	25.63

(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1974	775	0.7768178	0.78	26.40
1975	911	0.9131367	0.91	27.32
1976	2501	2.5068661	2.51	29.82
1977	742	0.7437404	0.74	30.57
1978	429	0.4300062	0.43	31.00
1979	530	0.5312431	0.53	31.53
1980	1116	1.1186176	1.12	32.65
1981	581	0.5823627	0.58	33.23
1982	542	0.5432713	0.54	33.77
1983	244	0.2445723	0.24	34.02
1984	357	0.3578373	0.36	34.38
1985	601	0.6024096	0.60	34.98
1986	563	0.5643205	0.56	35.54
1987	455	0.4560672	0.46	36.00
1988	1276	1.2789928	1.28	37.28
1989	1142	1.1446785	1.14	38.42
1990	2426	2.4316902	2.43	40.85
1991	780	0.7818295	0.78	41.64
1992	1011	1.0133713	1.01	42.65
1993	992	0.9943267	0.99	43.64
1994	3791	3.7998917	3.80	47.44
1995	749	0.7507568	0.75	48.19
1996	957	0.9592446	0.96	49.15
1997	1864	1.8683720	1.87	51.02
1998	1472	1.4754526	1.48	52.50
1999	801	0.8028787	0.80	53.30
2000	703	0.7046489	0.70	54.00
2001	1674	1.6779263	1.68	55.68
2002	2168	2.1730850	2.17	57.86



(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
2003	1232	1.2348896	1.23	59.09
2004	2732	2.7384079	2.74	61.83
2005	2063	2.0678387	2.07	63.90
2006	2213	2.2181906	2.22	66.11
2007	1670	1.6739170	1.67	67.79
2008	2220	2.2252070	2.23	70.01
2009	2821	2.8276166	2.83	72.84
2010	1448	1.4513963	1.45	74.29
2011	2402	2.4076339	2.41	76.70
2012	1508	1.5115370	1.51	78.21
2013	3447	3.4550849	3.46	81.67
2014	1635	1.6388349	1.64	83.31
2015	2042	2.0467895	2.05	85.35
2016	3122	3.1293226	3.13	88.48
2017	3523	3.5312632	3.53	92.01
2018	1638	1.6418419	1.64	93.66
2019	1731	1.7350600	1.74	95.39
2020	2857	2.8637011	2.86	98.25
2021	1742	1.7460858	1.75	100.00
Total	99766	100.0000000	100.00	100.00

Frequency Table for Variable: doi\_concept

1 unique value(s) detected.

doi_concept	N	exactpercent	roundedpercent	cumulpercent
10.5281/zenodo.3832111	99766	100	100	100
Total	99766	100	100	100

---

Frequency Table for Variable: doi\_version

---

1 unique value(s) detected.

doi_version	N	exactpercent	roundedpercent	cumulpercent
10.5281/zenodo.5133932	99766	100	100	100
Total	99766	100	100	100

---

Frequency Table for Variable: version

---

1 unique value(s) detected.

version	N	exactpercent	roundedpercent	cumulpercent
2021-07-30	99766	100	100	100
Total	99766	100	100	100

---

Frequency Table for Variable: lizenz

---

1 unique value(s) detected.

	lizenz	N	exactpercent	roundedpercent	cumulpercent
Creative Commons Zero 1.0 Universal		99766	100	100	100
	Total	99766	100	100	100

## 8 Frequenztabellen erstellen: Rechtsakte

### 8.1 Variablen ignorieren

Folgende Variablen sind wegen der geringeren Auflösung der Metadaten (nur Rechtsaktebene, nicht Normebene) nicht mehr nutzbar:

```
varremove <- c("gliederungskennzahl")

vars.freqtable.rechtsakte <- grep(paste(varremove,
                                         collapse = "|"),
                                vars.freqtable,
                                invert = TRUE,
                                value = TRUE)
```

### 8.2 Liste zu prüfender Variablen

```
print(vars.freqtable.rechtsakte)
```

```
## [1] "periodikum"      "fundstellentyp"  "check_neuf"
## [4] "check_aufh"      "check_sonst"     "check_hinweis"
## [7] "check_stand"     "ausfertigung_jahr" "doi_concept"
## [10] "doi_version"     "version"         "lizenz"
```

### 8.3 Frequenztabellen erstellen

```
prefix <- paste0(datasetname,
                  "_01_Rechtsakte_Frequenztafel_var-")
```

```
f.fast.freqtable(dt.rechtsakte,
                 varlist = vars.freqtable.rechtsakte,
                 sumrow = TRUE,
                 output.list = FALSE,
                 output.kable = TRUE,
                 output.csv = TRUE,
                 outputdir = outputdir,
                 prefix = prefix)
```

---

Frequency Table for Variable: periodikum

---

21 unique value(s) detected.

periodikum	N	exactpercent	roundedpercent	cumulpercent
BAGVBl	1	0.0177431	0.02	0.02
BAnz	144	2.5550035	2.56	2.57
BGBI	21	0.3726047	0.37	2.95
BGBI I	4853	86.1071682	86.11	89.05
BGBI II	375	6.6536551	6.65	95.71
GBl DDR	15	0.2661462	0.27	95.97
GBl DDR I	43	0.7629524	0.76	96.74
GBl DDR II	3	0.0532292	0.05	96.79
GVBl BE	1	0.0177431	0.02	96.81
NV	1	0.0177431	0.02	96.82
RAnz	2	0.0354862	0.04	96.86
RGBI	51	0.9048971	0.90	97.76
RGBI I	47	0.8339248	0.83	98.60
RGBI II	45	0.7984386	0.80	99.40
RMBI	4	0.0709723	0.07	99.47
VOBl BrZ	1	0.0177431	0.02	99.49
VkBl	10	0.1774308	0.18	99.66
WiGBI	7	0.1242016	0.12	99.79
ZBl	2	0.0354862	0.04	99.82
eBAnz	6	0.1064585	0.11	99.93
Öff Anz	4	0.0709723	0.07	100.00
Total	5636	100.0000000	100.00	100.00

Frequency Table for Variable: fundstellentyp

1 unique value(s) detected.

fundstellentyp	N	exactpercent	roundedpercent	cumulpercent
amtlich	5636	100	100	100
Total	5636	100	100	100

---

Frequency Table for Variable: check\_neuf

---

2 unique value(s) detected.

check_neuf	N	exactpercent	roundedpercent	cumulpercent
NA	5220	92.618879	92.62	92.62
ja	416	7.381121	7.38	100.00
Total	5636	100.000000	100.00	100.00

---

Frequency Table for Variable: check\_aufh

---

2 unique value(s) detected.

check_aufh	N	exactpercent	roundedpercent	cumulpercent
NA	5445	96.611072	96.61	96.61
ja	191	3.388928	3.39	100.00
Total	5636	100.000000	100.00	100.00

---

Frequency Table for Variable: check\_sonst

---

2 unique value(s) detected.

check_sonst	N	exactpercent	roundedpercent	cumulpercent
NA	5151	91.394606	91.39	91.39
ja	485	8.605394	8.61	100.00
Total	5636	100.000000	100.00	100.00

---

Frequency Table for Variable: check\_hinweis

---

2 unique value(s) detected.

check_hinweis	N	exactpercent	roundedpercent	cumulpercent
NA	5156	91.483322	91.48	91.48
ja	480	8.516678	8.52	100.00
Total	5636	100.000000	100.00	100.00

Frequency Table for Variable: check\_stand

2 unique value(s) detected.

check_stand	N	exactpercent	roundedpercent	cumulpercent
NA	2861	50.76295	50.76	50.76
ja	2775	49.23705	49.24	100.00
Total	5636	100.00000	100.00	100.00

Frequency Table for Variable: ausfertigung\_jahr

123 unique value(s) detected.

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1869	1	0.0177431	0.02	0.02
1871	3	0.0532292	0.05	0.07
1877	3	0.0532292	0.05	0.12
1884	1	0.0177431	0.02	0.14
1886	2	0.0354862	0.04	0.18
1887	1	0.0177431	0.02	0.20
1889	2	0.0354862	0.04	0.23
1892	1	0.0177431	0.02	0.25
1894	2	0.0354862	0.04	0.28
1895	2	0.0354862	0.04	0.32
1896	2	0.0354862	0.04	0.35
1897	6	0.1064585	0.11	0.46

(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1898	1	0.0177431	0.02	0.48
1899	2	0.0354862	0.04	0.51
1901	2	0.0354862	0.04	0.55
1903	1	0.0177431	0.02	0.57
1906	1	0.0177431	0.02	0.59
1907	1	0.0177431	0.02	0.60
1908	1	0.0177431	0.02	0.62
1909	3	0.0532292	0.05	0.67
1910	2	0.0354862	0.04	0.71
1911	2	0.0354862	0.04	0.75
1913	2	0.0354862	0.04	0.78
1919	4	0.0709723	0.07	0.85
1920	1	0.0177431	0.02	0.87
1921	4	0.0709723	0.07	0.94
1922	6	0.1064585	0.11	1.05
1923	2	0.0354862	0.04	1.08
1924	2	0.0354862	0.04	1.12
1925	2	0.0354862	0.04	1.15
1926	4	0.0709723	0.07	1.22
1927	5	0.0887154	0.09	1.31
1928	3	0.0532292	0.05	1.37
1929	3	0.0532292	0.05	1.42
1930	3	0.0532292	0.05	1.47
1931	5	0.0887154	0.09	1.56
1933	8	0.1419446	0.14	1.70
1934	7	0.1242016	0.12	1.83
1935	3	0.0532292	0.05	1.88
1936	7	0.1242016	0.12	2.00
1937	9	0.1596877	0.16	2.16

(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1938	10	0.1774308	0.18	2.34
1939	6	0.1064585	0.11	2.45
1940	7	0.1242016	0.12	2.57
1941	2	0.0354862	0.04	2.61
1942	2	0.0354862	0.04	2.64
1943	1	0.0177431	0.02	2.66
1944	1	0.0177431	0.02	2.68
1947	1	0.0177431	0.02	2.70
1948	5	0.0887154	0.09	2.79
1949	12	0.2129170	0.21	3.00
1950	21	0.3726047	0.37	3.37
1951	32	0.5677786	0.57	3.94
1952	14	0.2484031	0.25	4.19
1953	41	0.7274663	0.73	4.91
1954	17	0.3016324	0.30	5.22
1955	26	0.4613201	0.46	5.68
1956	33	0.5855216	0.59	6.26
1957	46	0.8161817	0.82	7.08
1958	20	0.3548616	0.35	7.43
1959	27	0.4790632	0.48	7.91
1960	33	0.5855216	0.59	8.50
1961	52	0.9226402	0.92	9.42
1962	31	0.5500355	0.55	9.97
1963	26	0.4613201	0.46	10.43
1964	28	0.4968062	0.50	10.93
1965	45	0.7984386	0.80	11.73
1966	25	0.4435770	0.44	12.17
1967	39	0.6919801	0.69	12.86
1968	41	0.7274663	0.73	13.59



(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1969	54	0.9581263	0.96	14.55
1970	45	0.7984386	0.80	15.35
1971	54	0.9581263	0.96	16.31
1972	53	0.9403833	0.94	17.25
1973	33	0.5855216	0.59	17.83
1974	53	0.9403833	0.94	18.77
1975	64	1.1355571	1.14	19.91
1976	73	1.2952449	1.30	21.20
1977	58	1.0290987	1.03	22.23
1978	45	0.7984386	0.80	23.03
1979	40	0.7097232	0.71	23.74
1980	54	0.9581263	0.96	24.70
1981	41	0.7274663	0.73	25.43
1982	42	0.7452094	0.75	26.17
1983	29	0.5145493	0.51	26.69
1984	32	0.5677786	0.57	27.25
1985	40	0.7097232	0.71	27.96
1986	51	0.9048971	0.90	28.87
1987	38	0.6742370	0.67	29.54
1988	48	0.8516678	0.85	30.39
1989	50	0.8871540	0.89	31.28
1990	157	2.7856636	2.79	34.07
1991	77	1.3662172	1.37	35.43
1992	77	1.3662172	1.37	36.80
1993	82	1.4549326	1.45	38.25
1994	163	2.8921221	2.89	41.15
1995	70	1.2420156	1.24	42.39
1996	88	1.5613911	1.56	43.95
1997	108	1.9162527	1.92	45.87

(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1998	122	2.1646558	2.16	48.03
1999	79	1.4017033	1.40	49.43
2000	60	1.0645848	1.06	50.50
2001	114	2.0227111	2.02	52.52
2002	148	2.6259759	2.63	55.15
2003	101	1.7920511	1.79	56.94
2004	143	2.5372605	2.54	59.47
2005	135	2.3953158	2.40	61.87
2006	129	2.2888573	2.29	64.16
2007	105	1.8630234	1.86	66.02
2008	129	2.2888573	2.29	68.31
2009	162	2.8743790	2.87	71.19
2010	122	2.1646558	2.16	73.35
2011	122	2.1646558	2.16	75.51
2012	109	1.9339957	1.93	77.45
2013	186	3.3002129	3.30	80.75
2014	103	1.8275373	1.83	82.58
2015	132	2.3420866	2.34	84.92
2016	147	2.6082328	2.61	87.53
2017	178	3.1582683	3.16	90.68
2018	97	1.7210788	1.72	92.41
2019	121	2.1469127	2.15	94.55
2020	184	3.2647268	3.26	97.82
2021	123	2.1823989	2.18	100.00
Total	5636	100.0000000	100.00	100.00

Frequency Table for Variable: doi\_concept

1 unique value(s) detected.

doi_concept	N	exactpercent	roundedpercent	cumulpercent
10.5281/zenodo.3832111	5636	100	100	100
Total	5636	100	100	100

---

Frequency Table for Variable: doi\_version

---

1 unique value(s) detected.

doi_version	N	exactpercent	roundedpercent	cumulpercent
10.5281/zenodo.5133932	5636	100	100	100
Total	5636	100	100	100

---

Frequency Table for Variable: version

---

1 unique value(s) detected.

version	N	exactpercent	roundedpercent	cumulpercent
2021-07-30	5636	100	100	100
Total	5636	100	100	100

---

Frequency Table for Variable: lizenz

---

1 unique value(s) detected.

	lizenz	N	exactpercent	roundedpercent	cumulpercent
Creative Commons Zero 1.0 Universal		5636	100	100	100
	Total	5636	100	100	100

## 9 Frequenztabellen erstellen: XML-Metadaten

### 9.1 Liste zu prüfender Variablen

```
print(vars.freqtable.rechtsakte)
```

```
## [1] "periodikum"      "fundstellentyp"  "check_neuf"
## [4] "check_aufh"      "check_sonst"     "check_hinweis"
## [7] "check_stand"     "ausfertigung_jahr" "doi_concept"
## [10] "doi_version"     "version"         "lizenz"
```

### 9.2 Frequenztabellen erstellen

```
prefix <- paste0(datasetname,
                  "_01_Meta_Frequenztabelle_var-")
```

```
f.fast.freqtable(dt.meta,
                 varlist = vars.freqtable.rechtsakte,
                 sumrow = TRUE,
                 output.list = FALSE,
                 output.kable = TRUE,
                 output.csv = TRUE,
                 outputdir = outputdir,
                 prefix = prefix)
```

---

Frequency Table for Variable: periodikum

---

36 unique value(s) detected.

periodikum	N	exactpercent	roundedpercent	cumulpercent
NA	4	0.0603318	0.06	0.06
ABl EU	1	0.0150830	0.02	0.08
BAGVBl	2	0.0301659	0.03	0.11
BAnz	255	3.8461538	3.85	3.95
BGBI	23	0.3469080	0.35	4.30
BGBI I	4870	73.4539970	73.45	77.75
BGBI II	1171	17.6621418	17.66	95.41

(continued)

periodikum	N	exactpercent	roundedpercent	cumulpercent
BGBI III	1	0.0150830	0.02	95.43
BremGBI	4	0.0603318	0.06	95.49
GBI BW	1	0.0150830	0.02	95.51
GBI DDR	15	0.2262443	0.23	95.73
GBI DDR I	44	0.6636501	0.66	96.40
GBI DDR II	3	0.0452489	0.05	96.44
GV NW	2	0.0301659	0.03	96.47
GVBI BB	1	0.0150830	0.02	96.49
GVBI BE	1	0.0150830	0.02	96.50
GVBI BY	4	0.0603318	0.06	96.56
GVBI HE	5	0.0754148	0.08	96.64
GVBI RP I	2	0.0301659	0.03	96.67
Mtbl BAA	1	0.0150830	0.02	96.68
NV	4	0.0603318	0.06	96.74
Nds GVBI	1	0.0150830	0.02	96.76
RAnz	2	0.0301659	0.03	96.79
RBesBI	1	0.0150830	0.02	96.80
RGBI	56	0.8446456	0.84	97.65
RGBI I	54	0.8144796	0.81	98.46
RGBI II	48	0.7239819	0.72	99.19
RMBI	4	0.0603318	0.06	99.25
RegBI WB	4	0.0603318	0.06	99.31
RegBI WH	1	0.0150830	0.02	99.32
VOBI BrZ	3	0.0452489	0.05	99.37
VkBI	23	0.3469080	0.35	99.71
WiGBI	7	0.1055807	0.11	99.82
ZBI	2	0.0301659	0.03	99.85
eBAnz	6	0.0904977	0.09	99.94
Öff Anz	4	0.0603318	0.06	100.00

(continued)

periodikum	N	exactpercent	roundedpercent	cumulpercent
Total	6630	100.0000000	100.00	100.00

Frequency Table for Variable: fundstellentyp

2 unique value(s) detected.

fundstellentyp	N	exactpercent	roundedpercent	cumulpercent
amtlich	6626	99.9396682	99.94	99.94
nichtamtlich	4	0.0603318	0.06	100.00
Total	6630	100.0000000	100.00	100.00

Frequency Table for Variable: check\_neuf

2 unique value(s) detected.

check_neuf	N	exactpercent	roundedpercent	cumulpercent
NA	6214	93.72549	93.73	93.73
ja	416	6.27451	6.27	100.00
Total	6630	100.00000	100.00	100.00

Frequency Table for Variable: check\_aufh

2 unique value(s) detected.

check_aufh	N	exactpercent	roundedpercent	cumulpercent
NA	6427	96.93816	96.94	96.94
ja	203	3.06184	3.06	100.00
Total	6630	100.00000	100.00	100.00

---

Frequency Table for Variable: check\_sonst

---

2 unique value(s) detected.

check_sonst	N	exactpercent	roundedpercent	cumulpercent
NA	6136	92.54902	92.55	92.55
ja	494	7.45098	7.45	100.00
Total	6630	100.00000	100.00	100.00

---

Frequency Table for Variable: check\_hinweis

---

2 unique value(s) detected.

check_hinweis	N	exactpercent	roundedpercent	cumulpercent
NA	6150	92.760181	92.76	92.76
ja	480	7.239819	7.24	100.00
Total	6630	100.000000	100.00	100.00

---

Frequency Table for Variable: check\_stand

---

2 unique value(s) detected.

check_stand	N	exactpercent	roundedpercent	cumulpercent
NA	3853	58.11463	58.11	58.11
ja	2777	41.88537	41.89	100.00
Total	6630	100.00000	100.00	100.00

---

Frequency Table for Variable: ausfertigung\_jahr

---

126 unique value(s) detected.

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
NA	1	0.0150830	0.02	0.02
1869	1	0.0150830	0.02	0.03
1871	3	0.0452489	0.05	0.08
1877	3	0.0452489	0.05	0.12
1884	1	0.0150830	0.02	0.14
1886	2	0.0301659	0.03	0.17
1887	1	0.0150830	0.02	0.18
1889	2	0.0301659	0.03	0.21
1892	1	0.0150830	0.02	0.23
1894	2	0.0301659	0.03	0.26
1895	2	0.0301659	0.03	0.29
1896	2	0.0301659	0.03	0.32
1897	6	0.0904977	0.09	0.41
1898	1	0.0150830	0.02	0.42
1899	2	0.0301659	0.03	0.45
1901	2	0.0301659	0.03	0.48
1903	1	0.0150830	0.02	0.50
1906	1	0.0150830	0.02	0.51
1907	1	0.0150830	0.02	0.53
1908	1	0.0150830	0.02	0.54
1909	4	0.0603318	0.06	0.60
1910	2	0.0301659	0.03	0.63
1911	3	0.0452489	0.05	0.68
1913	2	0.0301659	0.03	0.71
1919	7	0.1055807	0.11	0.81
1920	1	0.0150830	0.02	0.83
1921	4	0.0603318	0.06	0.89
1922	6	0.0904977	0.09	0.98
1923	2	0.0301659	0.03	1.01
1924	2	0.0301659	0.03	1.04



(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1925	2	0.0301659	0.03	1.07
1926	4	0.0603318	0.06	1.13
1927	5	0.0754148	0.08	1.21
1928	4	0.0603318	0.06	1.27
1929	4	0.0603318	0.06	1.33
1930	3	0.0452489	0.05	1.37
1931	5	0.0754148	0.08	1.45
1933	9	0.1357466	0.14	1.58
1934	7	0.1055807	0.11	1.69
1935	3	0.0452489	0.05	1.73
1936	9	0.1357466	0.14	1.87
1937	9	0.1357466	0.14	2.01
1938	11	0.1659125	0.17	2.17
1939	11	0.1659125	0.17	2.34
1940	8	0.1206637	0.12	2.46
1941	2	0.0301659	0.03	2.49
1942	2	0.0301659	0.03	2.52
1943	1	0.0150830	0.02	2.53
1944	4	0.0603318	0.06	2.59
1945	1	0.0150830	0.02	2.61
1946	12	0.1809955	0.18	2.79
1947	11	0.1659125	0.17	2.96
1948	13	0.1960784	0.20	3.15
1949	18	0.2714932	0.27	3.42
1950	23	0.3469080	0.35	3.77
1951	41	0.6184012	0.62	4.39
1952	24	0.3619910	0.36	4.75
1953	47	0.7088989	0.71	5.46
1954	26	0.3921569	0.39	5.85

(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1955	34	0.5128205	0.51	6.37
1956	54	0.8144796	0.81	7.18
1957	56	0.8446456	0.84	8.02
1958	36	0.5429864	0.54	8.57
1959	48	0.7239819	0.72	9.29
1960	45	0.6787330	0.68	9.97
1961	69	1.0407240	1.04	11.01
1962	41	0.6184012	0.62	11.63
1963	36	0.5429864	0.54	12.17
1964	42	0.6334842	0.63	12.81
1965	61	0.9200603	0.92	13.73
1966	38	0.5731523	0.57	14.30
1967	50	0.7541478	0.75	15.05
1968	50	0.7541478	0.75	15.81
1969	71	1.0708899	1.07	16.88
1970	57	0.8597285	0.86	17.74
1971	67	1.0105581	1.01	18.75
1972	66	0.9954751	1.00	19.74
1973	69	1.0407240	1.04	20.78
1974	61	0.9200603	0.92	21.70
1975	74	1.1161388	1.12	22.82
1976	82	1.2368024	1.24	24.06
1977	71	1.0708899	1.07	25.13
1978	58	0.8748115	0.87	26.00
1979	56	0.8446456	0.84	26.85
1980	69	1.0407240	1.04	27.89
1981	48	0.7239819	0.72	28.61
1982	47	0.7088989	0.71	29.32
1983	40	0.6033183	0.60	29.92

(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1984	52	0.7843137	0.78	30.71
1985	52	0.7843137	0.78	31.49
1986	74	1.1161388	1.12	32.61
1987	46	0.6938160	0.69	33.30
1988	61	0.9200603	0.92	34.22
1989	59	0.8898944	0.89	35.11
1990	171	2.5791855	2.58	37.69
1991	85	1.2820513	1.28	38.97
1992	92	1.3876320	1.39	40.36
1993	99	1.4932127	1.49	41.86
1994	189	2.8506787	2.85	44.71
1995	113	1.7043741	1.70	46.41
1996	108	1.6289593	1.63	48.04
1997	139	2.0965309	2.10	50.14
1998	137	2.0663650	2.07	52.20
1999	95	1.4328808	1.43	53.63
2000	72	1.0859729	1.09	54.72
2001	129	1.9457014	1.95	56.67
2002	155	2.3378582	2.34	59.00
2003	120	1.8099548	1.81	60.81
2004	155	2.3378582	2.34	63.15
2005	157	2.3680241	2.37	65.52
2006	138	2.0814480	2.08	67.60
2007	119	1.7948718	1.79	69.40
2008	152	2.2926094	2.29	71.69
2009	175	2.6395173	2.64	74.33
2010	139	2.0965309	2.10	76.43
2011	135	2.0361991	2.04	78.46
2012	123	1.8552036	1.86	80.32

(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
2013	190	2.8657617	2.87	83.18
2014	104	1.5686275	1.57	84.75
2015	134	2.0211161	2.02	86.77
2016	152	2.2926094	2.29	89.06
2017	181	2.7300151	2.73	91.79
2018	99	1.4932127	1.49	93.29
2019	125	1.8853695	1.89	95.17
2020	189	2.8506787	2.85	98.02
2021	131	1.9758673	1.98	100.00
Total	6630	100.0000000	100.00	100.00

Frequency Table for Variable: doi\_concept

1 unique value(s) detected.

doi_concept	N	exactpercent	roundedpercent	cumulpercent
10.5281/zenodo.3832111	6630	100	100	100
Total	6630	100	100	100

Frequency Table for Variable: doi\_version

1 unique value(s) detected.

doi_version	N	exactpercent	roundedpercent	cumulpercent
10.5281/zenodo.5133932	6630	100	100	100
Total	6630	100	100	100

Frequency Table for Variable: version

1 unique value(s) detected.

version	N	exactpercent	roundedpercent	cumulpercent
2021-07-30	6630	100	100	100
Total	6630	100	100	100

Frequency Table for Variable: lizenz

1 unique value(s) detected.

	lizenz	N	exactpercent	roundedpercent	cumulpercent
Creative Commons Zero 1.0 Universal		6630	100	100	100
	Total	6630	100	100	100

## 10 Frequenztabellen visualisieren

### 10.1 Präfixe erstellen

```
prefix.normen <- paste0("ANALYSE/",  
                        datasetname,  
                        "_01_Einzelnormen_Frequenztafel_var-")  
  
prefix.rechtsakte <- paste0("ANALYSE/",  
                            datasetname,  
                            "_01_Rechtsakte_Frequenztafel_var-")  
  
prefix.meta <- paste0("ANALYSE/",  
                     datasetname,  
                     "_01_Meta_Frequenztafel_var-")
```

### 10.2 Tabellen für Einzelnormen einlesen

```
table.normen.periodikum <- fread(paste0(prefix.normen,  
                                         "periodikum.csv"))  
  
table.normen.ausjahr <- fread(paste0(prefix.normen,  
                                       "ausfertigung_jahr.csv"))
```

### 10.3 Tabellen für Rechtsakte einlesen

```
table.rechtsakte.periodikum <- fread(paste0(prefix.rechtsakte,  
                                             "periodikum.csv"))  
  
table.rechtsakte.ausjahr <- fread(paste0(prefix.rechtsakte,  
                                           "ausfertigung_jahr.csv"))
```

### 10.4 Tabellen für XML-Metadaten einlesen

```
table.meta.periodikum <- fread(paste0(prefix.meta,  
                                       "periodikum.csv"))  
  
table.meta.ausjahr <- fread(paste0(prefix.meta,  
                                    "ausfertigung_jahr.csv"))
```

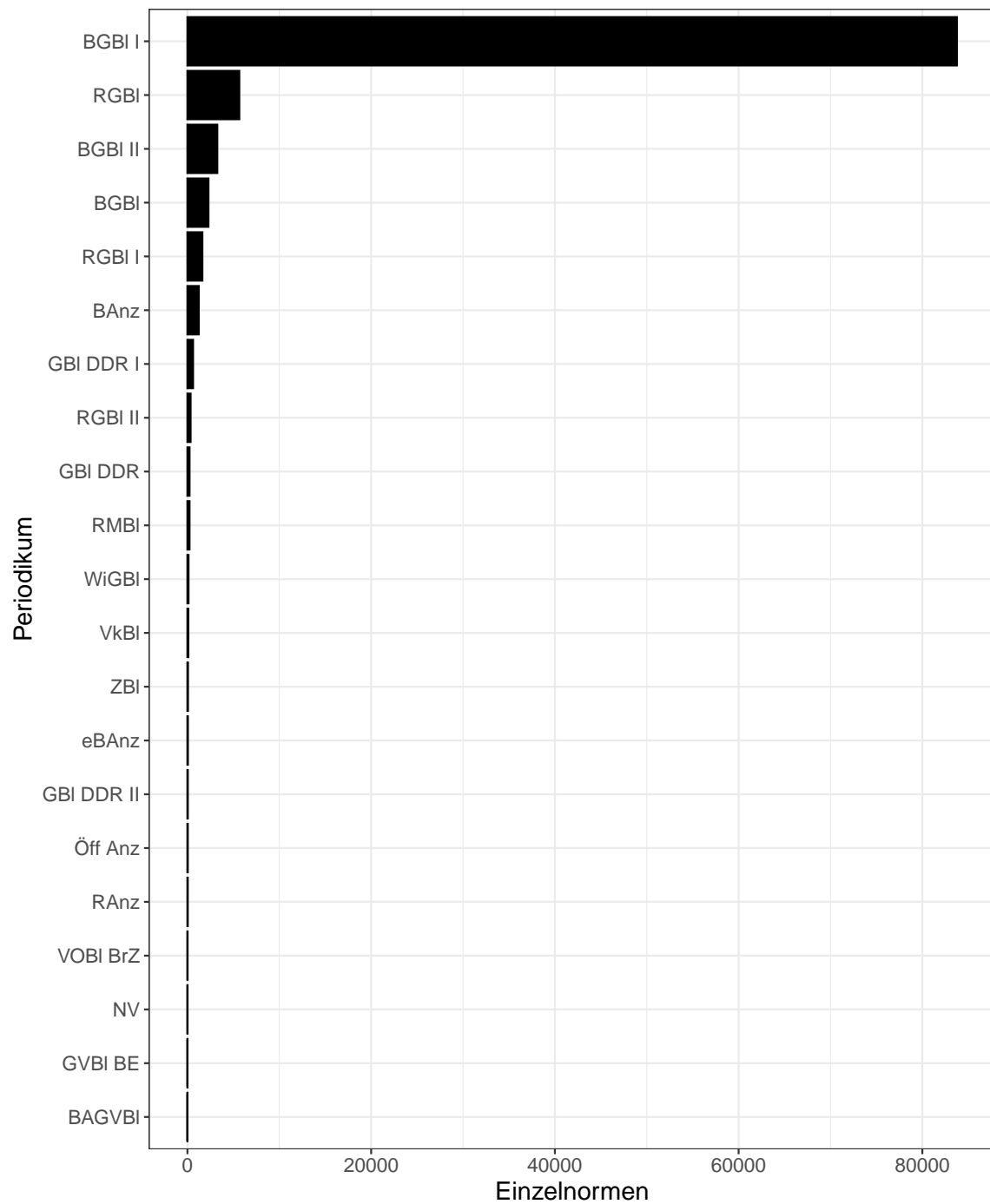
## 10.5 Periodikum

### 10.5.1 Einzelnormen

```
freqtable <- table.normen.periodikum[-.N]
```

```
ggplot(data = freqtable)+  
  geom_bar(aes(x = reorder(periodikum,  
                           N),  
              y = N),  
          stat = "identity",  
          fill = "black",  
          color = "black")+  
  coord_flip()+  
  theme_bw()+  
  labs(  
    title = paste(datasetname,  
                  "| Version",  
                  datestamp,  
                  "| Einzelnormen je Periodikum"),  
    caption = paste("DOI:",  
                    doi.version,  
                    "| S. Fobbe"),  
    x = "Periodikum",  
    y = "Einzelnormen"  
  )+  
  theme(  
    text = element_text(size = 14),  
    plot.title = element_text(size = 14,  
                               face = "bold"),  
    legend.position = "none",  
    plot.margin = margin(10, 20, 10, 10)  
  )
```

# C-DBR | Version 2021-07-30 | Einzelnormen je Periodikum



DOI: 10.5281/zenodo.5133932 | S. Fobbe

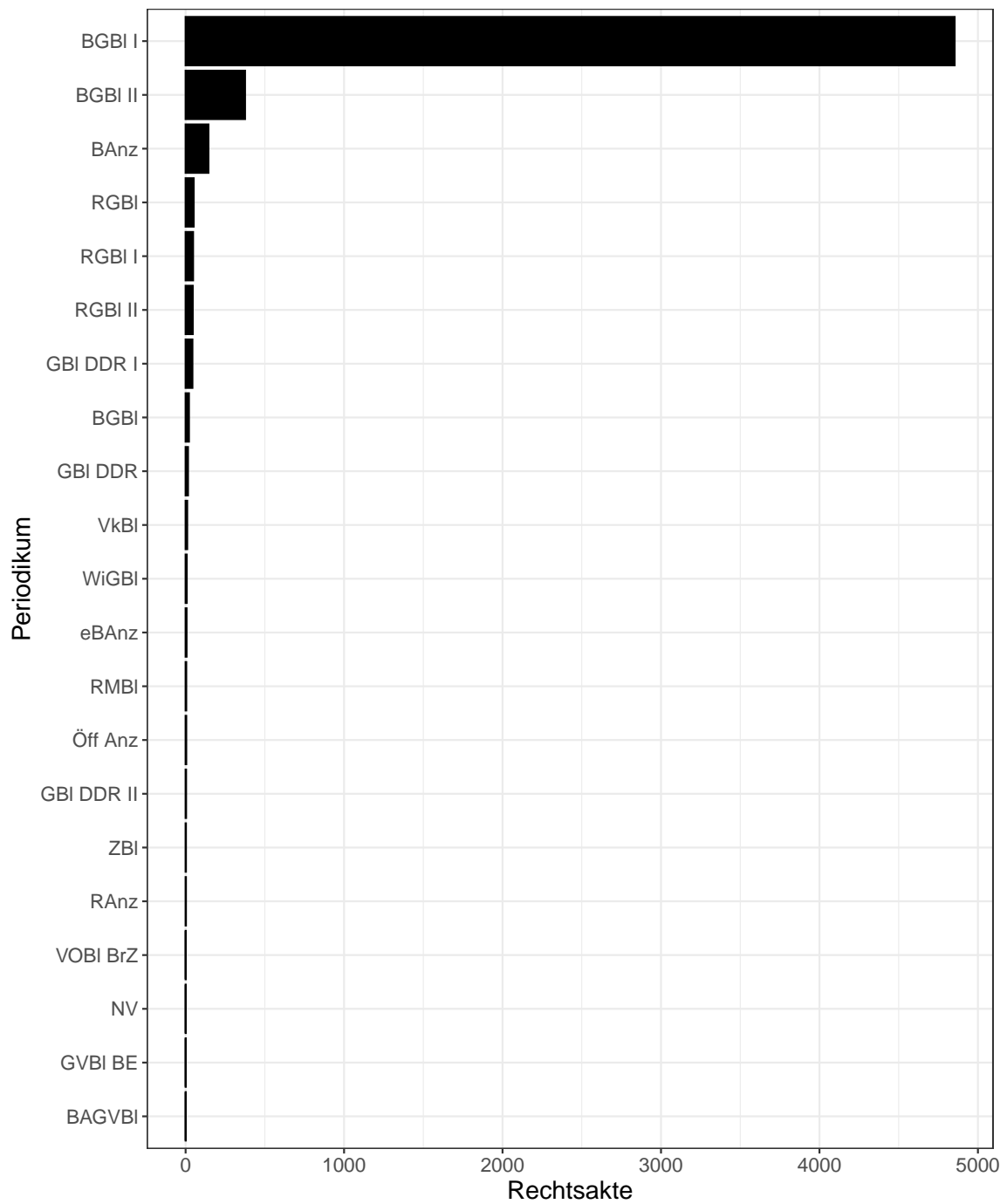


## 10.5.2 Rechtsakte

```
freqtable <- table.rechtsakte.periodikum[-.N]
```

```
ggplot(data = freqtable) +  
  geom_bar(aes(x = reorder(periodikum,  
                           N),  
              y = N),  
          stat = "identity",  
          fill = "black",  
          color = "black") +  
  coord_flip() +  
  theme_bw() +  
  labs(  
    title = paste(datasetname,  
                  "| Version",  
                  datestamp,  
                  "| Rechtsakte mit Inhalt je Periodikum"),  
    caption = paste("DOI:",  
                   doi.version,  
                   "| S. Fobbe"),  
    x = "Periodikum",  
    y = "Rechtsakte"  
  ) +  
  theme(  
    text = element_text(size = 14),  
    plot.title = element_text(size = 14,  
                              face = "bold"),  
    legend.position = "none",  
    plot.margin = margin(10, 20, 10, 10)  
  )
```

# **C-DBR | Version 2021-07-30 | Rechtsakte mit Inhalt je Periodikum**



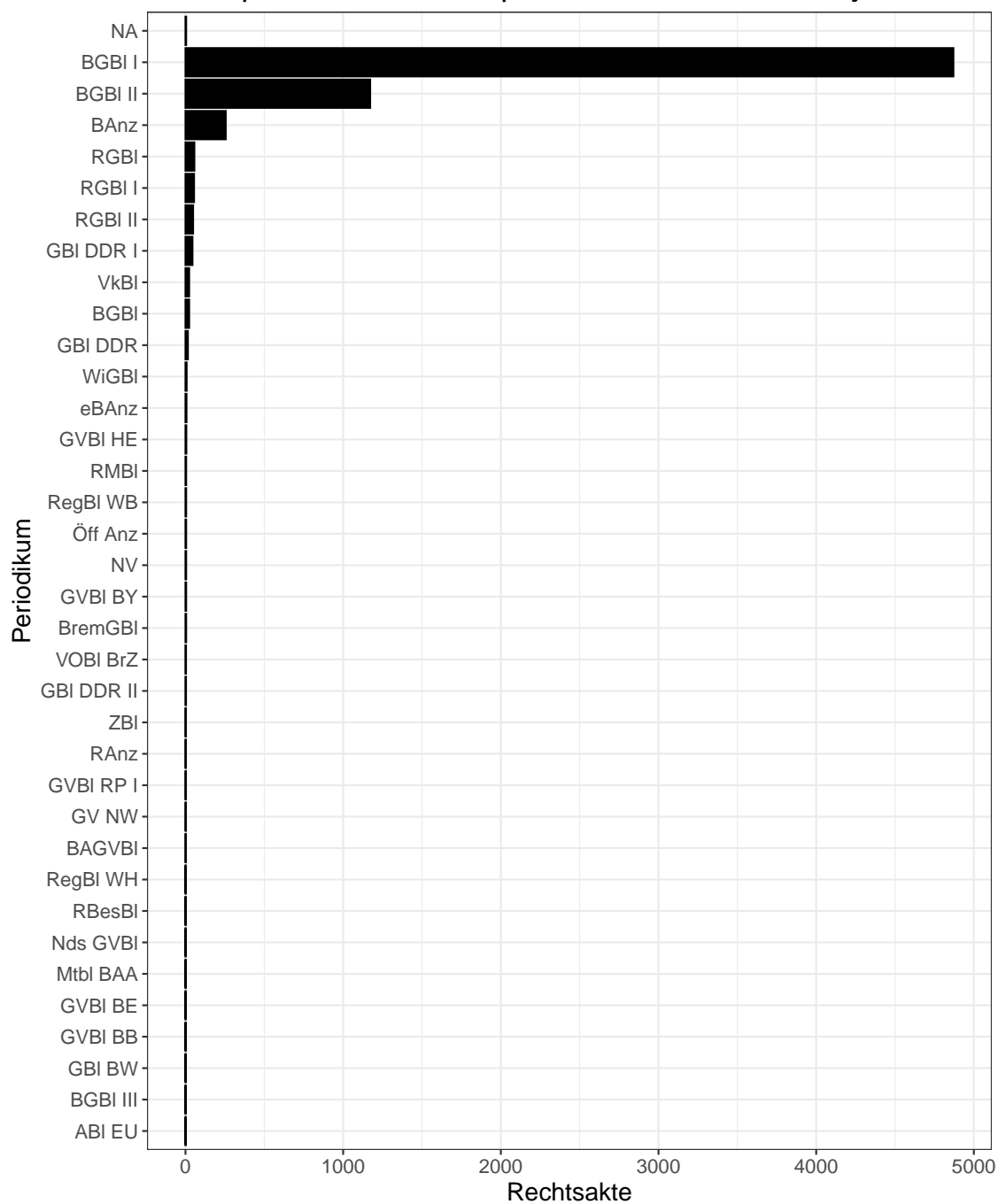
DOI: 10.5281/zenodo.5133932 | S. Fobbe

### 10.5.3 XML-Metadaten

```
freqtable <- table.meta.periodikum[-.N]
```

```
ggplot(data = freqtable) +  
  geom_bar(aes(x = reorder(periodikum,  
                           N),  
              y = N),  
          stat = "identity",  
          fill = "black",  
          color = "black") +  
  coord_flip() +  
  theme_bw() +  
  labs(  
    title = paste(datasetname,  
                  "| Version",  
                  datestamp,  
                  "| Rechtsakte nach Metadaten je Periodikum"),  
    caption = paste("DOI:",  
                    doi.version,  
                    "| S. Fobbe"),  
    x = "Periodikum",  
    y = "Rechtsakte"  
  ) +  
  theme(  
    text = element_text(size = 14),  
    plot.title = element_text(size = 14,  
                               face = "bold"),  
    legend.position = "none",  
    plot.margin = margin(10, 20, 10, 10)  
  )
```

# C-DBR | Version 2021-07-30 | Rechtsakte nach Metadaten je Periodikum



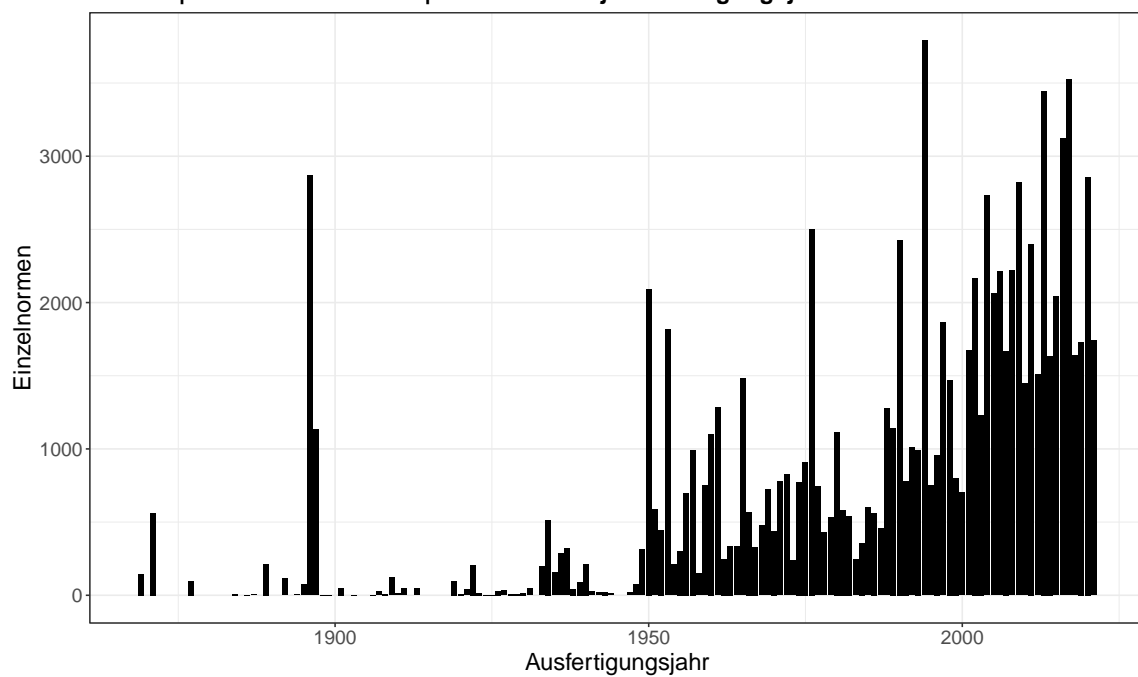
DOI: 10.5281/zenodo.5133932 | S. Fobbe

## 10.6 Ausfertigungsjahr

### 10.6.1 Einzelnormen

```
freqtable <- table.normen.ausjahr[-.N][,lapply(.SD, as.numeric)]
```

```
ggplot(data = freqtable) +  
  geom_bar(aes(x = ausfertigung_jahr,  
               y = N),  
           stat = "identity",  
           fill = "black")+  
  theme_bw()+  
  labs(  
    title = paste(datasetname,  
                  "| Version",  
                  datestamp,  
                  "| Einzelnormen je Ausfertigungsjahr"),  
    caption = paste("DOI:",  
                    doi.version,  
                    "| S. Fobbe"),  
    x = "Ausfertigungsjahr",  
    y = "Einzelnormen"  
  )+  
  theme(  
    text = element_text(size = 16),  
    plot.title = element_text(size = 16,  
                               face = "bold"),  
    legend.position = "none",  
    plot.margin = margin(10, 20, 10, 10)  
  )
```

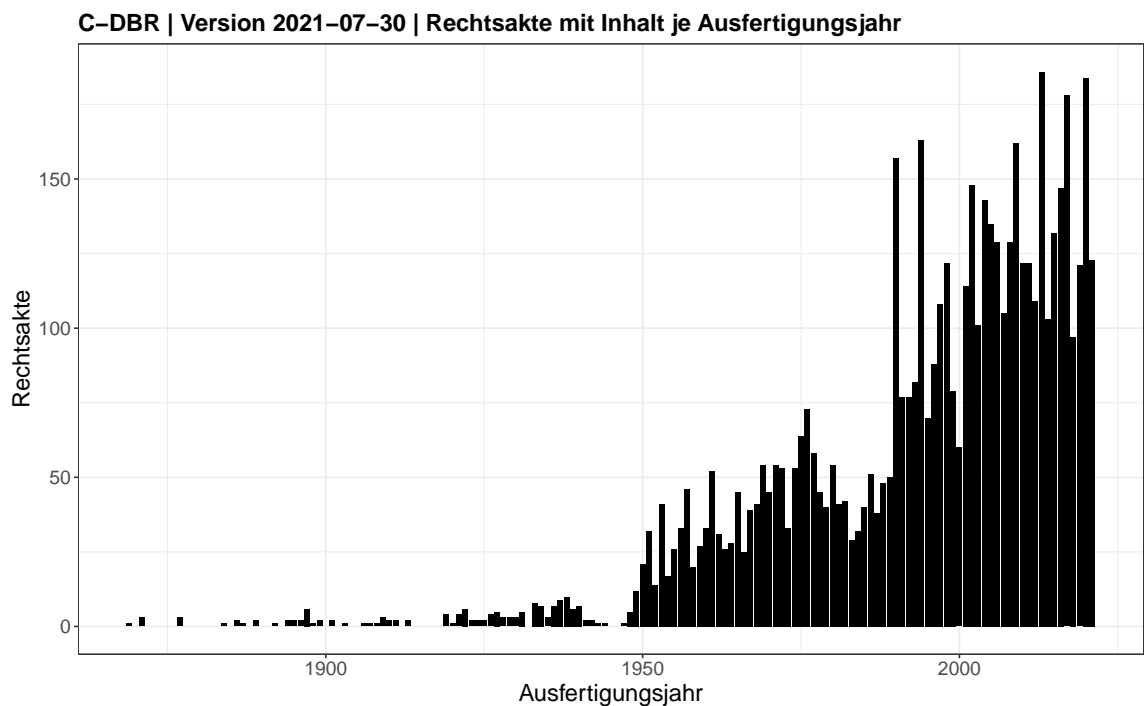


DOI: 10.5281/zenodo.5133932 | S. Fobbe

## 10.6.2 Rechtsakte

```
freqtable <- table.rechtsakte.ausjahr[-.N][,lapply(.SD, as.numeric)]
```

```
ggplot(data = freqtable) +  
  geom_bar(aes(x = ausfertigung_jahr,  
               y = N),  
           stat = "identity",  
           fill = "black") +  
  theme_bw() +  
  labs(  
    title = paste(datasetname,  
                  "| Version",  
                  datestamp,  
                  "| Rechtsakte mit Inhalt je Ausfertigungsjahr"),  
    caption = paste("DOI:",  
                    doi.version,  
                    "| S. Fobbe"),  
    x = "Ausfertigungsjahr",  
    y = "Rechtsakte"  
  ) +  
  theme(  
    text = element_text(size = 16),  
    plot.title = element_text(size = 16,  
                               face = "bold"),  
    legend.position = "none",  
    plot.margin = margin(10, 20, 10, 10)  
  )
```



### 10.6.3 XML-Metadaten

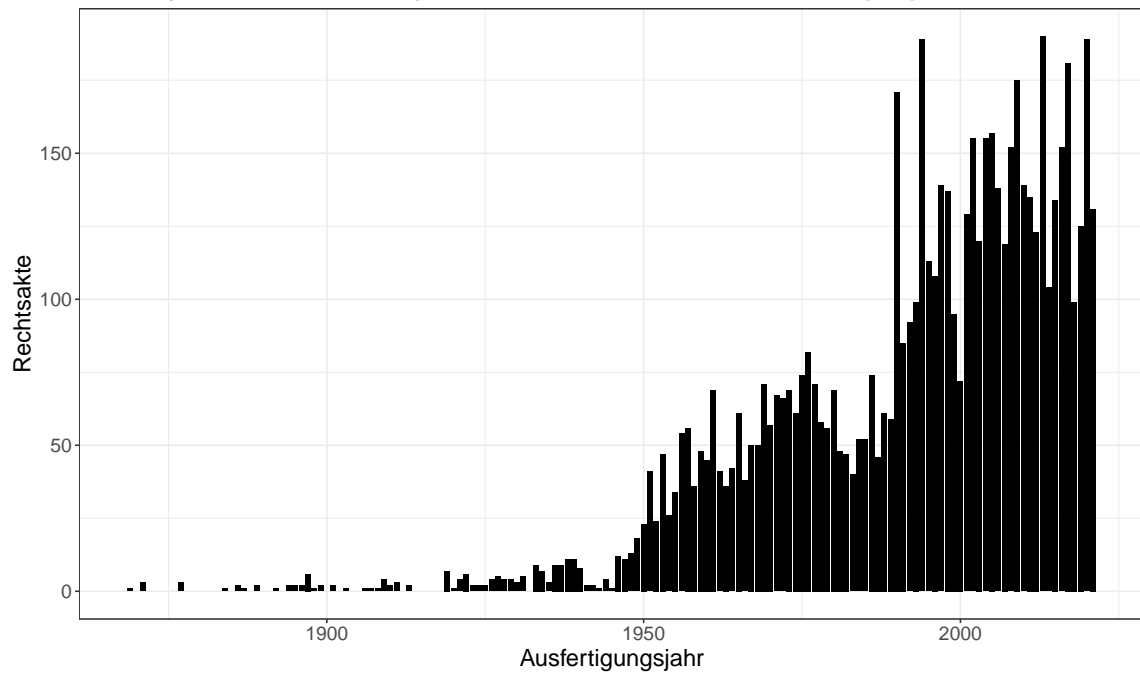
```
freqtable <- table.meta.ausjahr[-.N][,lapply(.SD, as.numeric)]
```

```
ggplot(data = freqtable) +  
  geom_bar(aes(x = ausfertigung_jahr,  
               y = N),  
           stat = "identity",  
           fill = "black") +  
  theme_bw() +  
  labs(  
    title = paste(datasetname,  
                  "| Version",  
                  datestamp,  
                  "| Rechtsakte nach Metadaten je Ausfertigungsjahr"),  
    caption = paste("DOI:",  
                    doi.version,  
                    "| S. Fobbe"),  
    x = "Ausfertigungsjahr",  
    y = "Rechtsakte"  
  ) +  
  theme(  
    text = element_text(size = 16),  
    plot.title = element_text(size = 16,  
                               face = "bold"),  
    legend.position = "none",  
    plot.margin = margin(10, 20, 10, 10)  
  )
```

```
## Warning: Removed 1 rows containing missing values (position_stack).
```



C-DBR | Version 2021-07-30 | Rechtsakte nach Metadaten je Ausfertigungsjahr



DOI: 10.5281/zenodo.5133932 | S. Fobbe

## 11 Korpus-Analytik

### 11.1 Berechnung linguistischer Kennwerte

An dieser Stelle werden für jedes Dokument die Anzahl Zeichen, Tokens, Typen und Sätze berechnet und mit den jeweiligen Metadaten verknüpft. Das Ergebnis ist grundsätzlich identisch mit dem eigentlichen Datensatz, nur ohne den Text der Entscheidungen.

#### 11.1.1 Funktion anzeigen: `f.summarize.iterator`

```
print(f.summarize.iterator)
```

```
## function(dt,
##                                     threads = detectCores(),
##                                     chunksize = 1){
##
##
##   begin.dopar <- Sys.time()
##   dt <- dt[,.(doc_id, text)]
##   nchars <- dt[, lapply(.(text), nchar)]
##   print(paste0("Parallel processing using ",
##               threads,
##               " threads. Begin at ",
##               begin.dopar,
##               ". Processing ",
##               dt[,.N],
##               " documents with a total length of ",
##               sum(nchars),
##               " characters."))
##
##   ord <- order(-nchars)
##   dt <- dt[ord]
##
##   cl <- makeForkCluster(threads)
##   registerDoParallel(cl)
##
##   itx <- iter(dt["nchars" > 0],
##              by = "row",
##              chunksize = chunksize)
##
##   result.list <- foreach(i = itx,
##                          .errorhandling = 'pass') %dopar% {
##
##               corpus <- corpus(i)
##
##               tokens <- tokens(corpus,
##                                what = "word",
```

```

##                                     remove_punct = FALSE,
##                                     remove_symbols = FALSE,
##                                     remove_numbers = FALSE,
##                                     remove_url = FALSE,
##                                     remove_separators = TRUE,
##                                     split_hyphens = FALSE,
##                                     include_docvars = FALSE,
##                                     padding = FALSE
##                                     )
##
##                                     ntokens <- unname(ntoken(tokens))
##                                     ntypes <- unname(ntype(tokens))
##                                     nsentences <- unname(nsentence(corpus))
##
##                                     temp <- data.table(ntokens,
##                                                         ntypes,
##                                                         nsentences)
##
##                                     return(temp)
##                                     }
##
## stopCluster(cl)
##
##
## end.dopar <- Sys.time()
## duration.dopar <- end.dopar - begin.dopar
##
## result.dt <- rbindlist(result.list)
##
## summary.corpus <- cbind(nchars[ord],
##                         result.dt)
##
## setnames(summary.corpus,
##          "V1",
##          "nchars")
##
##
## if(dt["nchars" == 0, .N] > 0){
##
##     dt.charnull <- dt["nchars" == 0]
##     dt.charnull$text <- NULL
##     dt.charnull$ntokens <- rep(0, dt.charnull[,.N])
##     dt.charnull$ntypes <- rep(0, dt.charnull[,.N])
##     dt.charnull$nsentences <- rep(0, dt.charnull[,.N])
##
##     summary.corpus <- rbind(summary.corpus,
##                             dt.charnull)
## }
##
##
## summary.corpus <- summary.corpus[order(ord)]
##
##
## print(paste0("Runtime was ",
##              round(duration.dopar,
##                    digits = 2),

```

```
##           " ",
##           attributes(duration.dopar)$units,
##           ". Ended at ",
##           end.dopar, ".")
##
##     return(summary.corpus)
##
## }
```

## 11.1.2 Berechnung durchführen

```
lingstats.normen.raw <- f.summarize.iterator(dt.normen,
                                             threads = fullCores,
                                             chunksize = 1)
```

```
## [1] "Parallel processing using 16 threads. Begin at 2021-07-30 19:01:38.
      Processing 99766 documents with a total length of 139377349 characters."
## [1] "Runtime was 5.08 mins. Ended at 2021-07-30 19:06:43."
```

```
lingstats.rechtsakte.raw <- f.summarize.iterator(dt.rechtsakte,
                                                  threads = fullCores,
                                                  chunksize = 1)
```

```
## [1] "Parallel processing using 16 threads. Begin at 2021-07-30 19:06:43.
      Processing 5636 documents with a total length of 139471479 characters."
## [1] "Runtime was 28.48 secs. Ended at 2021-07-30 19:07:12."
```

## 11.2 Variablen-Namen anpassen

### 11.2.1 Einzelnormen

```
setnames(lingstats.normen.raw,
         old = c("nchars",
                 "ntokens",
                 "ntypes",
                 "nsentences"),
         new = c("zeichen",
                 "tokens",
                 "typen",
                 "saetze"))
```

### 11.2.2 Rechtsakte

```
setnames(lingstats.rechtsakte.raw,  
  old = c("nchars",  
          "ntokens",  
          "ntypes",  
          "nsentences"),  
  new = c("zeichen",  
          "tokens",  
          "typen",  
          "saetze"))
```

## 11.3 Kennwerte den Korpora hinzufügen

### 11.3.1 Einzelnormen

```
dt.normen <- cbind(dt.normen,  
  lingstats.normen.raw)
```

### 11.3.2 Rechtsakte

```
dt.rechtsakte <- cbind(dt.rechtsakte,  
  lingstats.rechtsakte.raw)
```

## 11.4 Varianten mit Metadaten erstellen

### 11.4.1 Einzelnormen

```
meta.normen <- dt.normen[, !"text"]
```

### 11.4.2 Rechtsakte

```
meta.rechtsakte <- dt.rechtsakte[, !"text"]
```

## 11.5 Linguistische Kennwerte: Einzelnormen

**Hinweis:** Typen sind definiert als einzigartige Tokens und werden hier noch einmal bezogen auf den Gesamtkorpus berechnet, statt wie vorher bezogen auf jedes Dokument.

### 11.5.1 Zusammenfassungen berechnen

```
dt.summary.ling <- lingstats.normen.raw[, lapply(.SD,
                                              function(x) unclass(summary(x))),
                                          .SDcols = c("zeichen",
                                                    "tokens",
                                                    "typen",
                                                    "saetze")]

dt.sums.ling <- lingstats.normen.raw[,
                                      lapply(.SD, sum),
                                      .SDcols = c("zeichen",
                                                  "tokens",
                                                  "typen",
                                                  "saetze")]

tokens.normen <- tokens(corpus(dt.normen),
                        what = "word",
                        remove_punct = FALSE,
                        remove_symbols = FALSE,
                        remove_numbers = FALSE,
                        remove_url = FALSE,
                        remove_separators = TRUE,
                        split_hyphens = FALSE,
                        include_docvars = FALSE,
                        padding = FALSE
                        )

dt.sums.ling$typen <- nfeat(dfm(tokens.normen))

dt.stats.ling <- rbind(dt.sums.ling,
                      dt.summary.ling)

dt.stats.ling <- transpose(dt.stats.ling,
                          keep.names = "names")

setnames(dt.stats.ling, c("Variable",
                          "Sum",
                          "Min",
                          "Quart1",
                          "Median",
                          "Mean",
                          "Quart3",
```

```
"Max"))
```

### 11.5.2 Zusammenfassungen anzeigen

```
kable(dt.stats.ling,  
      format.args = list(big.mark = ","),  
      format = "latex",  
      booktabs = TRUE,  
      longtable = TRUE)
```

Variable	Sum	Min	Quart1	Median	Mean	Quart3	Max
zeichen	139,377,349	1	291	682	1,397.042570	1,472	398,079
tokens	21,838,310	1	46	108	218.895315	231	56,584
typen	401,213	1	35	66	92.391235	114	11,096
saetze	734,476	1	2	4	7.361987	8	2,381

### 11.5.3 Zusammenfassungen speichern

```
fwrite(dt.stats.ling,  
       paste0(outputdir,  
               datasetname,  
               "_00_Einzelnormen_KorpusStatistik_ZusammenfassungLinguistisch.csv"),  
       ,  
       na = "NA")
```



## 11.6 Linguistische Kennwerte: Rechtsakte

**Hinweis:** Typen sind definiert als einzigartige Tokens und werden hier noch einmal bezogen auf den Gesamtkorpus berechnet, statt wie vorher bezogen auf jedes Dokument.

### 11.6.1 Zusammenfassungen berechnen

```
dt.summary.ling <- lingstats.rechtsakte.raw[, lapply(.SD,
                                                    function(x) unclass(summary(x))),
                                                    .SDcols = c("zeichen",
                                                                "tokens",
                                                                "typen",
                                                                "saetze")]

dt.sums.ling <- lingstats.rechtsakte.raw[,
                                           lapply(.SD, sum),
                                           .SDcols = c("zeichen",
                                                         "tokens",
                                                         "typen",
                                                         "saetze")]

tokens.rechtsakte <- tokens(corpus(dt.rechtsakte),
                           what = "word",
                           remove_punct = FALSE,
                           remove_symbols = FALSE,
                           remove_numbers = FALSE,
                           remove_url = FALSE,
                           remove_separators = TRUE,
                           split_hyphens = FALSE,
                           include_docvars = FALSE,
                           padding = FALSE
                           )

dt.sums.ling$typen <- nfeat(dfm(tokens.rechtsakte))

dt.stats.ling <- rbind(dt.sums.ling,
                      dt.summary.ling)

dt.stats.ling <- transpose(dt.stats.ling,
                          keep.names = "names")

setnames(dt.stats.ling, c("Variable",
                          "Sum",
                          "Min",
                          "Quart1",
                          "Median",
                          "Mean",
                          "Quart3",
```

```
"Max"))
```

### 11.6.2 Zusammenfassungen anzeigen

```
kable(dt.stats.ling,  
      format.args = list(big.mark = ","),  
      format = "latex",  
      booktabs = TRUE,  
      longtable = TRUE)
```

Variable	Sum	Min	Quart1	Median	Mean	Quart3	Max
zeichen	139,471,479	94	1,570.00	5,669.5	24,746.5364	22,203.25	1,992,774
tokens	21,838,310	17	265.75	913.0	3,874.7889	3,380.50	293,235
typen	401,213	15	130.00	301.0	634.0546	801.25	14,657
saetze	723,013	1	13.00	36.0	128.2848	114.00	14,306

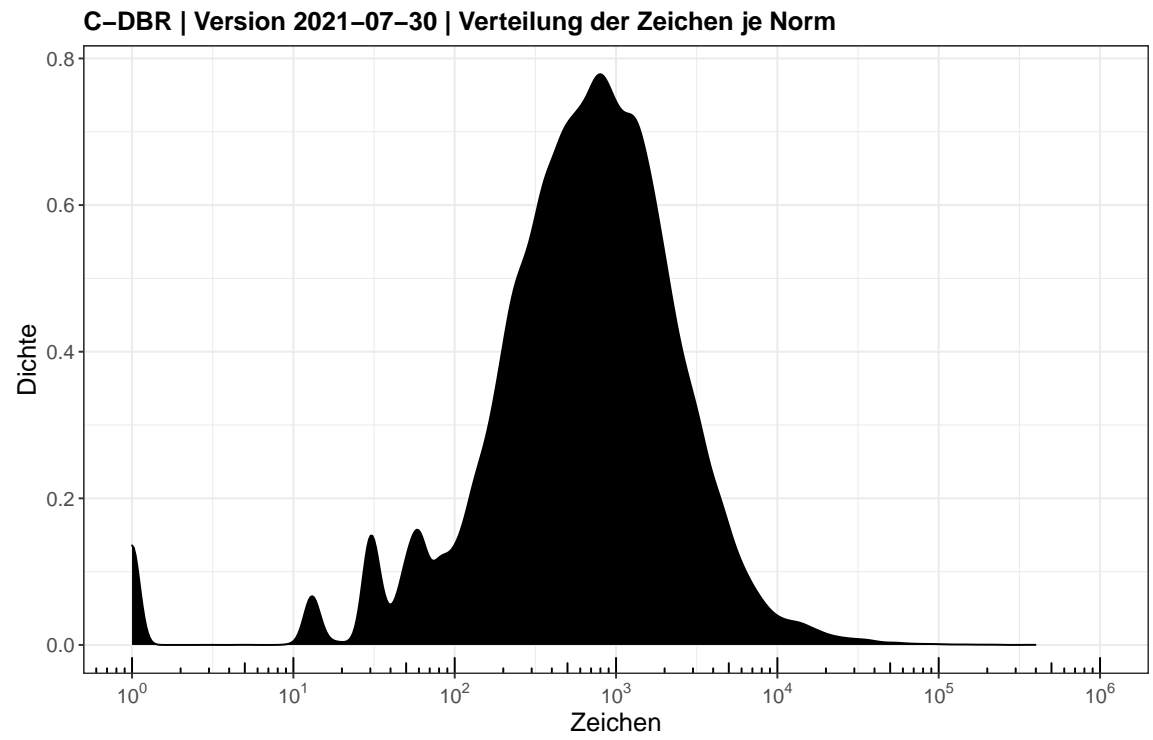
### 11.6.3 Zusammenfassungen speichern

```
fwrite(dt.stats.ling,  
       paste0(outputdir,  
               datasetname,  
               "_00_Rechtsakte_KorpusStatistik_ZusammenfassungLinguistisch.csv"),  
       na = "NA")
```

## 11.7 Verteilungen

### 11.7.1 Density (Zeichen)

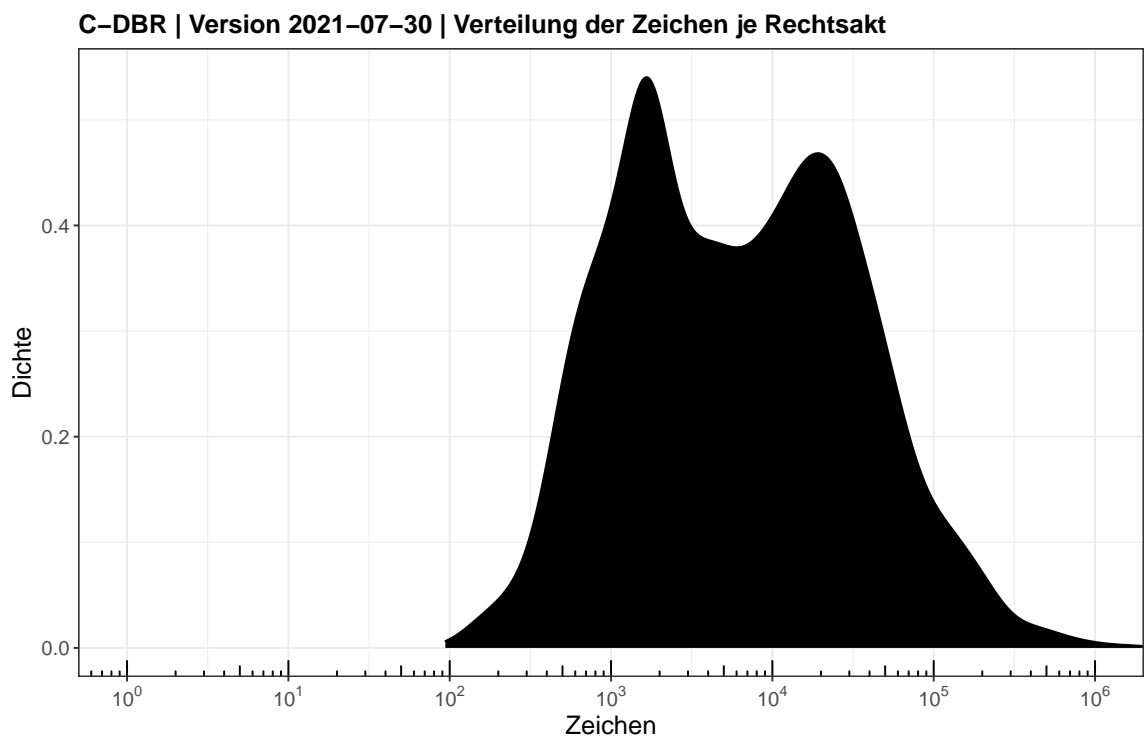
```
ggplot(data = meta.normen)+
  geom_density(aes(x = zeichen),
               fill = "black") +
  scale_x_log10(breaks = trans_breaks("log10", function(x) 10^x),
               labels = trans_format("log10", math_format(10^.x)))+
  annotation_logticks(sides = "b")+
  coord_cartesian(xlim = c(1, 10^6))+
  theme_bw()+
  labs(
    title = paste(datasetname,
                  "| Version",
                  datestamp,
                  "| Verteilung der Zeichen je Norm"),
    caption = paste("DOI:",
                    doi.version,
                    "| S. Fobbe"),
    x = "Zeichen",
    y = "Dichte"
  )+
  theme(
    text = element_text(size = 14),
    plot.title = element_text(size = 14,
                               face = "bold"),
    legend.position = "none",
    plot.margin = margin(10, 20, 10, 10)
  )
```



```

ggplot(data = meta.rechtsakte)+
  geom_density(aes(x = zeichen),
    fill = "black") +
  scale_x_log10(breaks = trans_breaks("log10", function(x) 10^x),
    labels = trans_format("log10", math_format(10^.x)))+
  annotation_logticks(sides = "b")+
  coord_cartesian(xlim = c(1, 10^6))+
  theme_bw()+
  labs(
    title = paste(datasetname,
      "| Version",
      datestamp,
      "| Verteilung der Zeichen je Rechtsakt"),
    caption = paste("DOI:",
      doi.version,
      "| S. Fobbe"),
    x = "Zeichen",
    y = "Dichte"
  )+
  theme(
    text = element_text(size = 14),
    plot.title = element_text(size = 14,
      face = "bold"),
    legend.position = "none",
    plot.margin = margin(10, 20, 10, 10)
  )

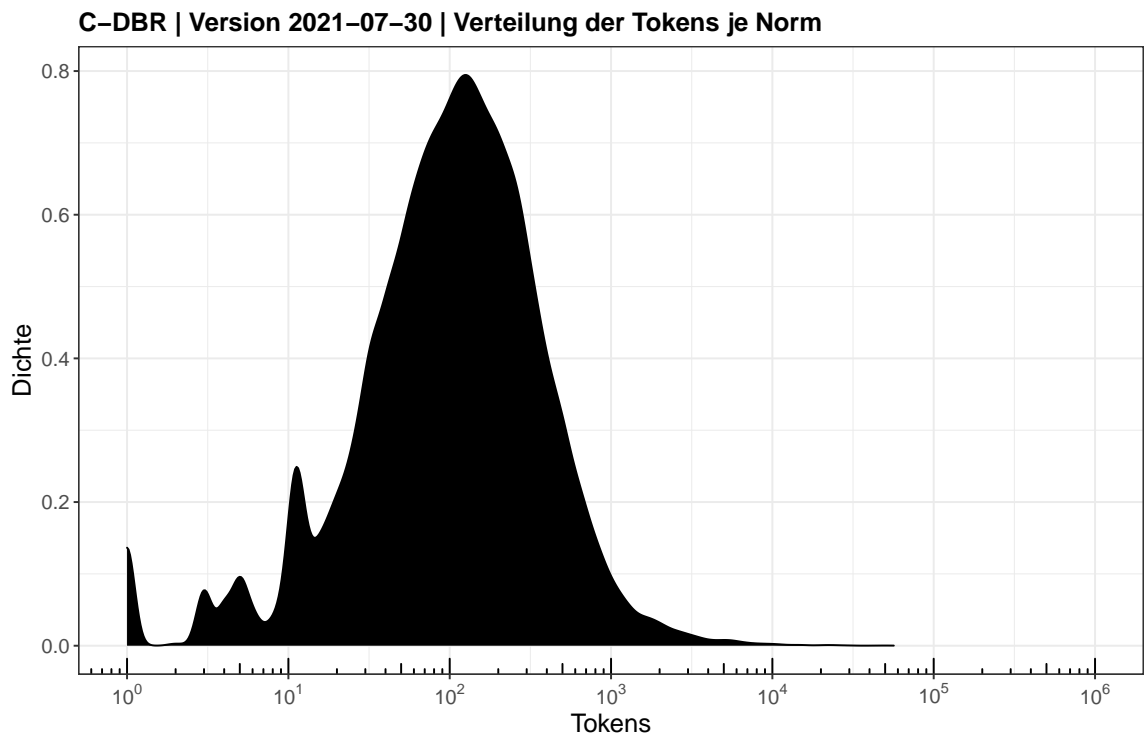
```



DOI: 10.5281/zenodo.5133932 | S. Fobbe

### 11.7.2 Density (Tokens)

```
ggplot(data = meta.normen)+
  geom_density(aes(x = tokens),
    fill = "black") +
  scale_x_log10(breaks = trans_breaks("log10", function(x) 10^x),
    labels = trans_format("log10", math_format(10^.x)))+
  annotation_logticks(sides = "b")+
  coord_cartesian(xlim = c(1, 10^6))+
  theme_bw()+
  labs(
    title = paste(datasetname,
      "| Version",
      datestamp,
      "| Verteilung der Tokens je Norm"),
    caption = paste("DOI:",
      doi.version,
      "| S. Fobbe"),
    x = "Tokens",
    y = "Dichte"
  )+
  theme(
    text = element_text(size = 14),
    plot.title = element_text(size = 14,
      face = "bold"),
    legend.position = "none",
    plot.margin = margin(10, 20, 10, 10)
  )
)
```

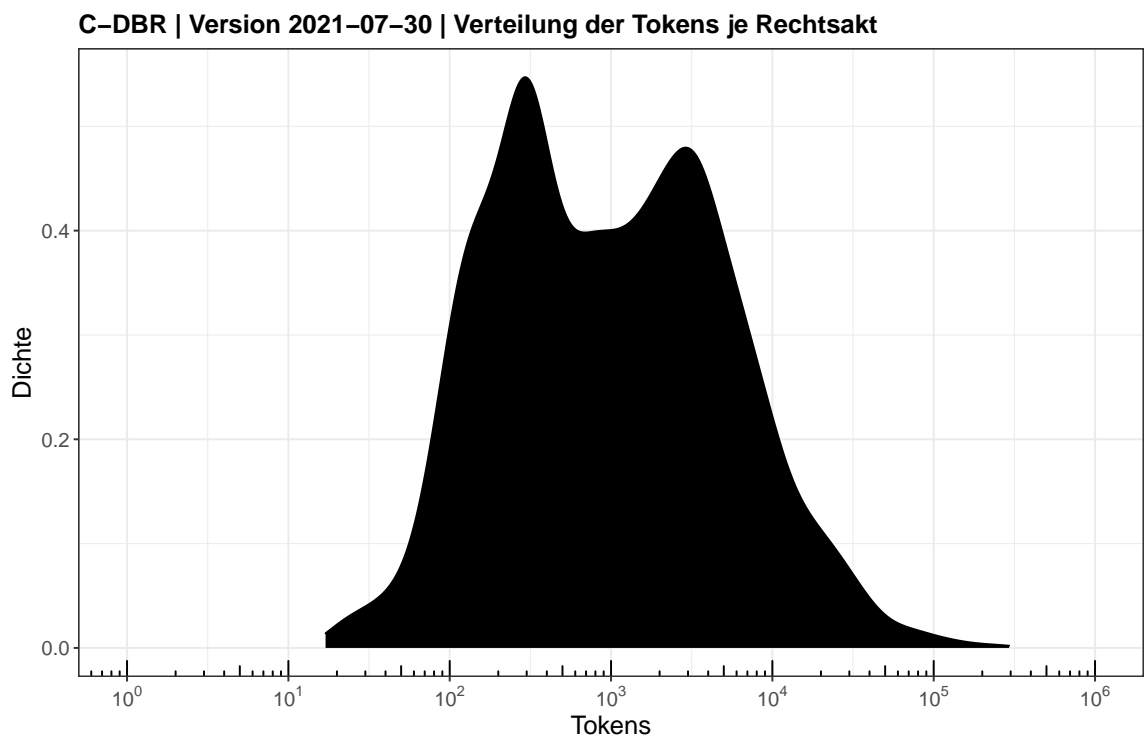


DOI: 10.5281/zenodo.5133932 | S. Fobbe

```

ggplot(data = meta.rechtsakte)+
  geom_density(aes(x = tokens),
    fill = "black")+
  scale_x_log10(breaks = trans_breaks("log10", function(x) 10^x),
    labels = trans_format("log10", math_format(10^.x)))+
  annotation_logticks(sides = "b")+
  coord_cartesian(xlim = c(1, 10^6))+
  theme_bw() +
  labs(
    title = paste(datasetname,
      "| Version",
      datestamp,
      "| Verteilung der Tokens je Rechtsakt"),
    caption = paste("DOI:",
      doi.version,
      "| S. Fobbe"),
    x = "Tokens",
    y = "Dichte"
  )+
  theme(
    text = element_text(size = 14),
    plot.title = element_text(size = 14,
      face = "bold"),
    legend.position = "none",
    plot.margin = margin(10, 20, 10, 10)
  )

```

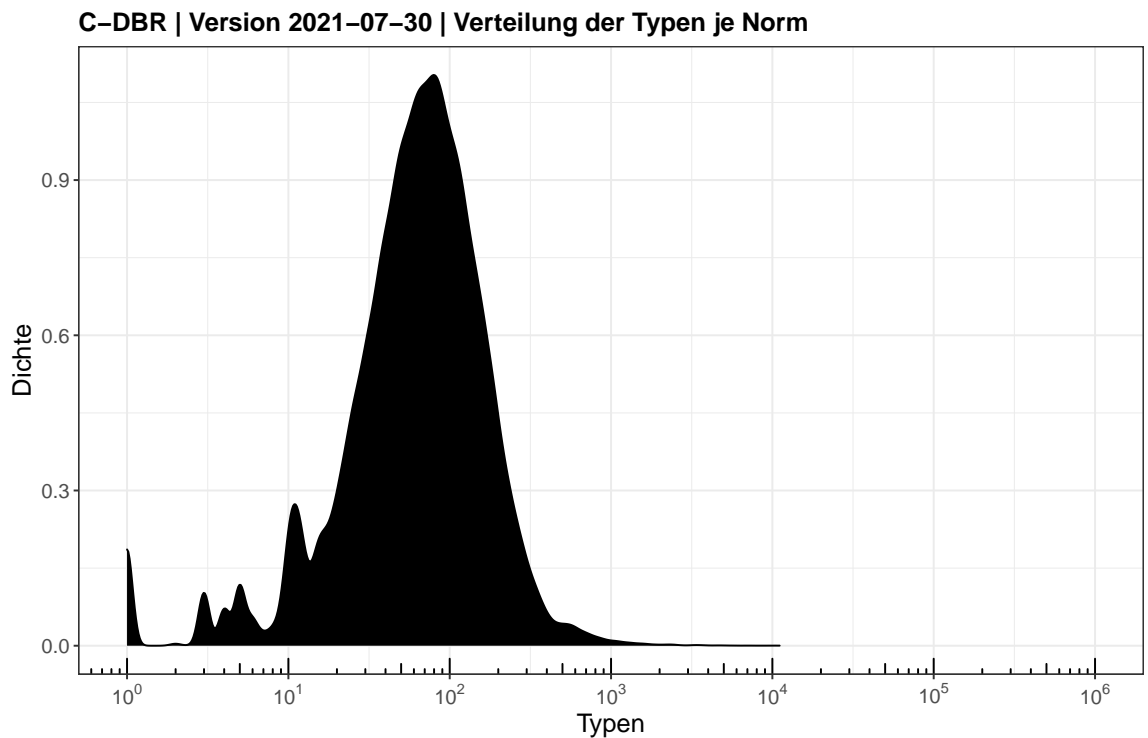


DOI: 10.5281/zenodo.5133932 | S. Fobbe



### 11.7.3 Density (Typen)

```
ggplot(data = meta.normen)+
  geom_density(aes(x = typen),
               fill = "black")+
  scale_x_log10(breaks = trans_breaks("log10", function(x) 10^x),
               labels = trans_format("log10", math_format(10^.x)))+
  annotation_logticks(sides = "b")+
  coord_cartesian(xlim = c(1, 10^6))+
  theme_bw()+
  labs(
    title = paste(datasetname,
                  "| Version",
                  datestamp,
                  "| Verteilung der Typen je Norm"),
    caption = paste("DOI:",
                    doi.version,
                    "| S. Fobbe"),
    x = "Typen",
    y = "Dichte"
  )+
  theme(
    text = element_text(size = 14),
    plot.title = element_text(size = 14,
                              face = "bold"),
    legend.position = "none",
    plot.margin = margin(10, 20, 10, 10)
  )
```

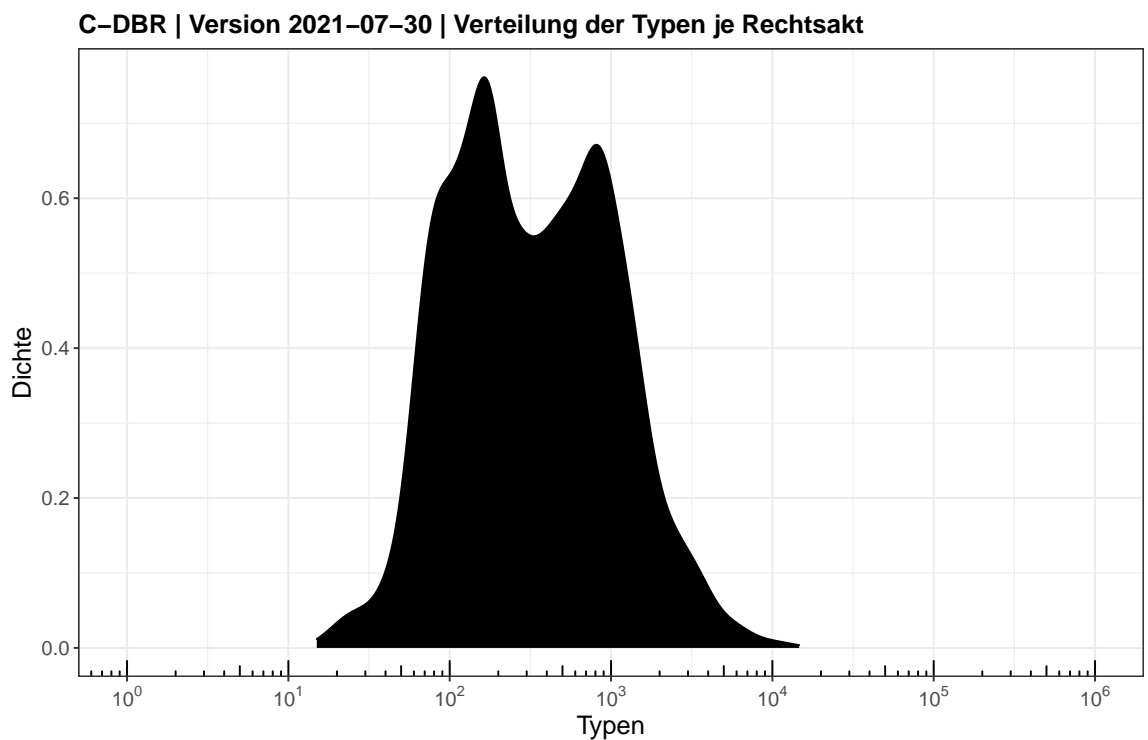


DOI: 10.5281/zenodo.5133932 | S. Fobbe

```

ggplot(data = meta.rechtsakte)+
  geom_density(aes(x = typen),
    fill = "black")+
  scale_x_log10(breaks = trans_breaks("log10", function(x) 10^x),
    labels = trans_format("log10", math_format(10^.x)))+
  annotation_logticks(sides = "b")+
  coord_cartesian(xlim = c(1, 10^6))+
  theme_bw()+
  labs(
    title = paste(datasetname,
      "| Version",
      datestamp,
      "| Verteilung der Typen je Rechtsakt"),
    caption = paste("DOI:",
      doi.version,
      "| S. Fobbe"),
    x = "Typen",
    y = "Dichte"
  )+
  theme(
    text = element_text(size = 14),
    plot.title = element_text(size = 14,
      face = "bold"),
    legend.position = "none",
    plot.margin = margin(10, 20, 10, 10)
  )

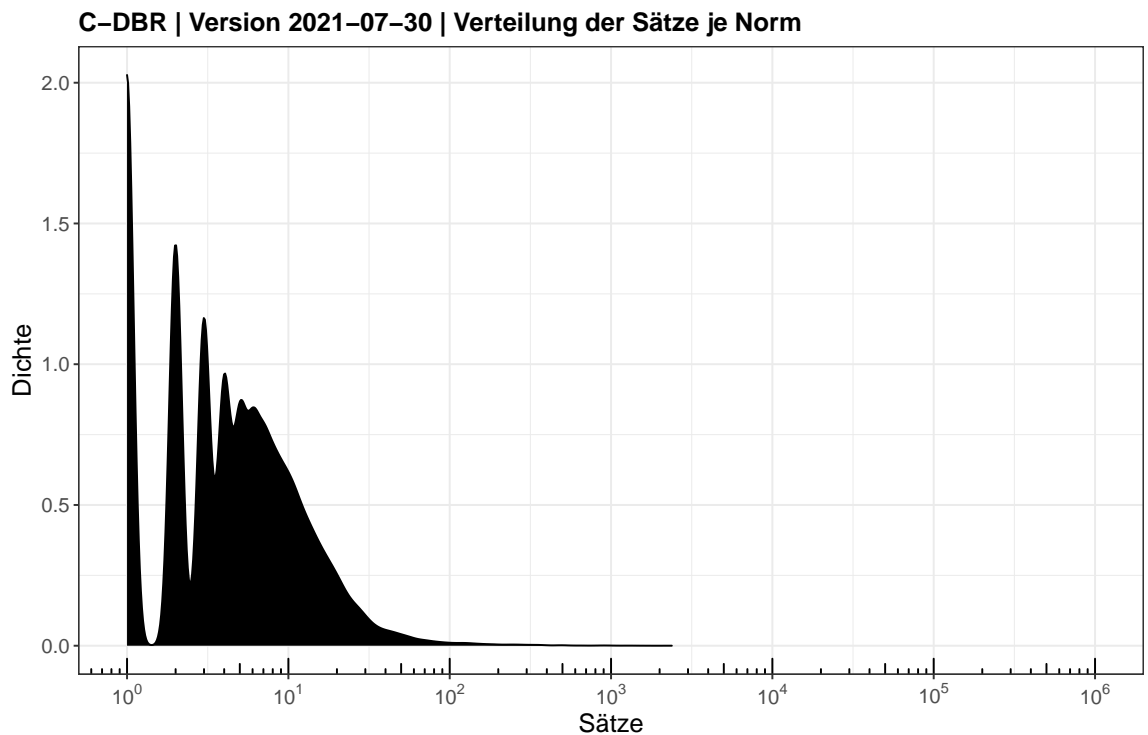
```



DOI: 10.5281/zenodo.5133932 | S. Fobbe

### 11.7.4 Density (Sätze)

```
ggplot(data = meta.normen)+
  geom_density(aes(x = saetze),
    fill = "black")+
  scale_x_log10(breaks = trans_breaks("log10", function(x) 10^x),
    labels = trans_format("log10", math_format(10^.x)))+
  annotation_logticks(sides = "b")+
  coord_cartesian(xlim = c(1, 10^6))+
  theme_bw()+
  labs(
    title = paste(datasetname,
      "| Version",
      datestamp,
      "| Verteilung der Sätze je Norm"),
    caption = paste("DOI:",
      doi.version,
      "| S. Fobbe"),
    x = "Sätze",
    y = "Dichte"
  )+
  theme(
    text = element_text(size = 14),
    plot.title = element_text(size = 14,
      face = "bold"),
    legend.position = "none",
    plot.margin = margin(10, 20, 10, 10)
  )
)
```

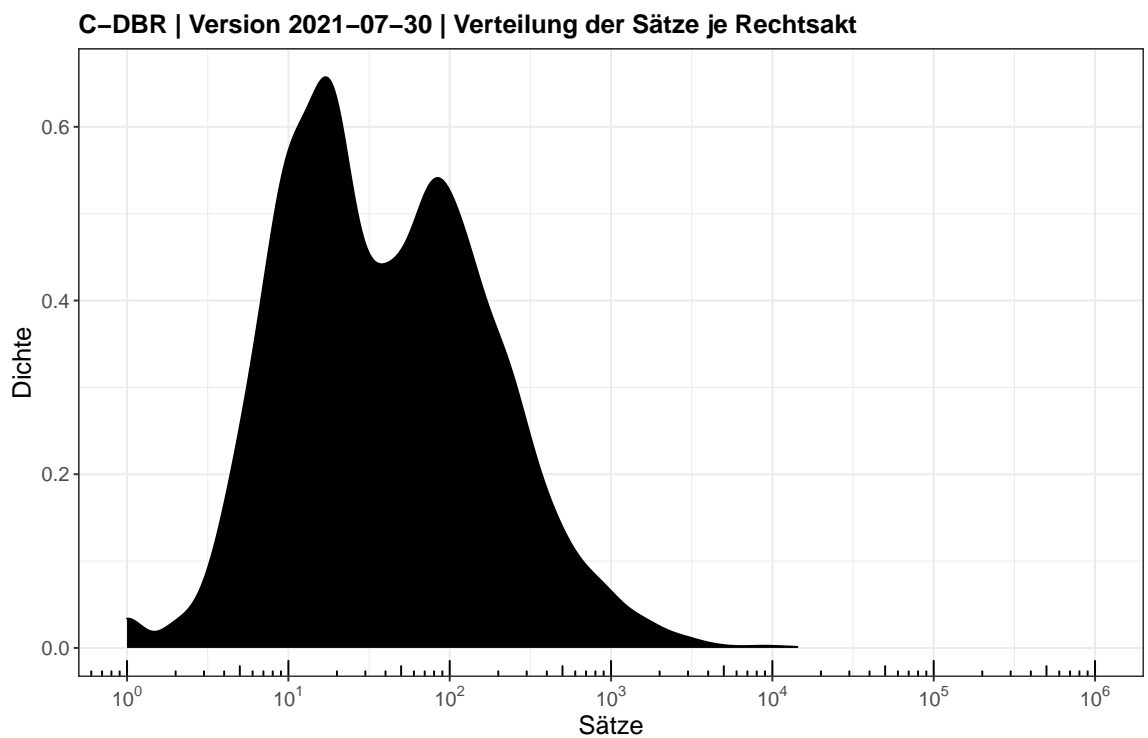


DOI: 10.5281/zenodo.5133932 | S. Fobbe

```

ggplot(data = meta.rechtsakte)+
  geom_density(aes(x = saetze),
    fill = "black")+
  scale_x_log10(breaks = trans_breaks("log10", function(x) 10^x),
    labels = trans_format("log10", math_format(10^.x)))+
  annotation_logticks(sides = "b")+
  coord_cartesian(xlim = c(1, 10^6))+
  theme_bw()+
  labs(
    title = paste(datasetname,
      "| Version",
      datestamp,
      "| Verteilung der Sätze je Rechtsakt"),
    caption = paste("DOI:",
      doi.version,
      "| S. Fobbe"),
    x = "Sätze",
    y = "Dichte"
  )+
  theme(
    text = element_text(size = 14),
    plot.title = element_text(size = 14,
      face = "bold"),
    legend.position = "none",
    plot.margin = margin(10, 20, 10, 10)
  )

```



DOI: 10.5281/zenodo.5133932 | S. Fobbe

## 11.8 Quantitative Variablen

### 11.8.1 Ausfertigungsdatum

#### *Einzelnormen*

```
summary(as.IDate(dt.normen$ausfertigung_datum))
```

```
##           Min.         1st Qu.         Median         Mean         3rd Qu.         Max.
## "1869-06-21" "1972-06-29" "1997-06-18" "1987-10-29" "2011-06-17" "2021-07-21"
```

#### *Rechtsakte*

```
summary(as.IDate(dt.rechtsakte$ausfertigung_datum))
```

```
##           Min.         1st Qu.         Median         Mean         3rd Qu.         Max.
## "1869-06-21" "1981-07-28" "2000-07-14" "1994-09-27" "2011-12-06" "2021-07-21"
```

#### *XML-Metadaten*

```
summary(as.IDate(dt.meta$ausfertigung_datum))
```

```
##           Min.         1st Qu.         Median         Mean         3rd Qu.         Max.
## "1869-06-21" "1977-12-15" "1997-12-19" "1992-12-10" "2010-05-22" "2021-07-21"
##           NA's
##           "1"
```

### 11.8.2 Ausfertigungsjahr

#### *Einzelnormen*

```
summary(dt.normen$ausfertigung_jahr)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1869   1972   1997   1987   2011   2021
```

#### *Rechtsakte*

```
summary(dt.rechtsakte$ausfertigung_jahr)
```

##	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
##	1869	1981	2000	1994	2011	2021

### *XML-Metadaten*

```
summary(dt.meta$ausfertigung_jahr)
```

##	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	NA's
##	1869	1977	1997	1992	2010	2021	1

## 12 Kontrolle der Variablen

### 12.1 Semantische Sortierung der Variablen

#### 12.1.1 Variablen sortieren: Einzelnormen

```
setcolorder(dt.normen,  
  c("doc_id",  
    "dateiname",  
    "text",  
    "jurabk",  
    "amtabk",  
    "langue",  
    "kurzue",  
    "gliederungskennzahl",  
    "gliederungsbez",  
    "gliederungstitel",  
    "enbez",  
    "bezketten",  
    "titelketten",  
    "ausfertigung_datum",  
    "ausfertigung_jahr",  
    "aenderung_datum",  
    "aufhebung_verkuendung_datum",  
    "aufhebung_wirkung_datum",  
    "neufassung_datum",  
    "fundstellentyp",  
    "periodikum",  
    "zitstelle",  
    "stand",  
    "aufh",  
    "neuf",  
    "hinweis",  
    "sonst",  
    "check_stand",  
    "check_aufh",  
    "check_neuf",  
    "check_hinweis",  
    "check_sonst",  
    "bulldate_original",  
    "bulldate_iso",  
    "zeichen",  
    "tokens",  
    "typen",  
    "saetze",  
    "doi_concept",  
    "doi_version",  
    "version",  
    "lizenz"))
```

```

setcolorder(meta.normen,
  c("doc_id",
    "dateiname",
    "jurabk",
    "amtabk",
    "langue",
    "kurzue",
    "gliederungskennzahl",
    "gliederungsbez",
    "gliederungstitel",
    "enbez",
    "bezketten",
    "titelkette",
    "ausfertigung_datum",
    "ausfertigung_jahr",
    "aenderung_datum",
    "aufhebung_verkuendung_datum",
    "aufhebung_wirkung_datum",
    "neufassung_datum",
    "fundstellentyp",
    "periodikum",
    "zitstelle",
    "stand",
    "aufh",
    "neuf",
    "hinweis",
    "sonst",
    "check_stand",
    "check_aufh",
    "check_neuf",
    "check_hinweis",
    "check_sonst",
    "bulldate_original",
    "bulldate_iso",
    "zeichen",
    "tokens",
    "typen",
    "saetze",
    "doi_concept",
    "doi_version",
    "version",
    "lizenz"))

```



### 12.1.2 Variablen sortieren: Rechtsakte

```
setcolorder(dt.rechtsakte,  
  c("doc_id",  
    "text",  
    "jurabk",  
    "amtabk",  
    "langue",  
    "kurzue",  
    "ausfertigung_datum",  
    "ausfertigung_jahr",  
    "aenderung_datum",  
    "aufhebung_verkuendung_datum",  
    "aufhebung_wirkung_datum",  
    "neufassung_datum",  
    "fundstellentyp",  
    "periodikum",  
    "zitstelle",  
    "stand",  
    "aufh",  
    "neuf",  
    "hinweis",  
    "sonst",  
    "check_stand",  
    "check_aufh",  
    "check_neuf",  
    "check_hinweis",  
    "check_sonst",  
    "zeichen",  
    "tokens",  
    "typen",  
    "saetze",  
    "doi_concept",  
    "doi_version",  
    "version",  
    "lizenz"))
```

```
setcolororder(meta.rechtsakte,  
  c("doc_id",  
    "jurabk",  
    "amtabk",  
    "langue",  
    "kurzue",  
    "ausfertigung_datum",  
    "ausfertigung_jahr",  
    "aenderung_datum",  
    "aufhebung_verkuendung_datum",  
    "aufhebung_wirkung_datum",  
    "neufassung_datum",  
    "fundstellentyp",  
    "periodikum",  
    "zitstelle",  
    "stand",  
    "aufh",  
    "neuf",  
    "hinweis",  
    "sonst",  
    "check_stand",  
    "check_aufh",  
    "check_neuf",  
    "check_hinweis",  
    "check_sonst",  
    "zeichen",  
    "tokens",  
    "typen",  
    "saetze",  
    "doi_concept",  
    "doi_version",  
    "version",  
    "lizenz"))
```

### 12.1.3 Variablen sortieren: XML-Metadaten

```
setcolorder(dt.meta,  
            c("doc_id",  
              "jurabk",  
              "amtabk",  
              "langue",  
              "kurzue",  
              "ausfertigung_datum",  
              "ausfertigung_jahr",  
              "aenderung_datum",  
              "aufhebung_verkuendung_datum",  
              "aufhebung_wirkung_datum",  
              "neufassung_datum",  
              "fundstellentyp",  
              "periodikum",  
              "zitstelle",  
              "stand",  
              "aufh",  
              "neuf",  
              "hinweis",  
              "sonst",  
              "check_stand",  
              "check_aufh",  
              "check_neuf",  
              "check_hinweis",  
              "check_sonst",  
              "bulddate_original",  
              "bulddate_iso",  
              "doi_concept",  
              "doi_version",  
              "version",  
              "lizenz"))
```

## 12.2 Anzahl Variablen der Datensätze

```
length(dt.normen)
```

```
## [1] 42
```

```
length(meta.normen)
```

```
## [1] 41
```

```
length(dt.rechtsakte)
```

```
## [1] 33
```

```
length(meta.rechtsakte)
```

```
## [1] 32
```

```
length(dt.meta)
```

```
## [1] 30
```

### 12.3 Alle Variablen-Namen der Datensätze

```
names(dt.normen)
```

```
## [1] "doc_id" "dateiname"
## [3] "text" "jurabk"
## [5] "amtabk" "langue"
## [7] "kurzue" "gliederungskennzahl"
## [9] "gliederungsbez" "gliederungstitel"
## [11] "enbez" "bezkette"
## [13] "titelkette" "ausfertigung_datum"
## [15] "ausfertigung_jahr" "aenderung_datum"
## [17] "aufhebung_verkuendung_datum" "aufhebung_wirkung_datum"
## [19] "neufassung_datum" "fundstellentyp"
## [21] "periodikum" "zitstelle"
## [23] "stand" "aufh"
## [25] "neuf" "hinweis"
## [27] "sonst" "check_stand"
## [29] "check_aufh" "check_neuf"
## [31] "check_hinweis" "check_sonst"
## [33] "builddate_original" "builddate_iso"
## [35] "zeichen" "tokens"
## [37] "typen" "saetze"
## [39] "doi_concept" "doi_version"
## [41] "version" "lizenz"
```

```
names(meta.normen)
```

```
## [1] "doc_id" "dateiname"
## [3] "jurabk" "amtabk"
## [5] "langue" "kurzue"
## [7] "gliederungskennzahl" "gliederungsbez"
## [9] "gliederungstitel" "enbez"
## [11] "bezkette" "titelkette"
## [13] "ausfertigung_datum" "ausfertigung_jahr"
## [15] "aenderung_datum" "aufhebung_verkuendung_datum"
## [17] "aufhebung_wirkung_datum" "neufassung_datum"
## [19] "fundstellentyp" "periodikum"
## [21] "zitstelle" "stand"
## [23] "aufh" "neuf"
## [25] "hinweis" "sonst"
## [27] "check_stand" "check_aufh"
## [29] "check_neuf" "check_hinweis"
## [31] "check_sonst" "builddate_original"
## [33] "builddate_iso" "zeichen"
## [35] "tokens" "typen"
## [37] "saetze" "doi_concept"
## [39] "doi_version" "version"
## [41] "lizenz"
```

```
names(dt.rechtsakte)
```

```
## [1] "doc_id" "text"
## [3] "jurabk" "amtabk"
## [5] "langue" "kurzue"
## [7] "ausfertigung_datum" "ausfertigung_jahr"
## [9] "aenderung_datum" "aufhebung_verkuendung_datum"
## [11] "aufhebung_wirkung_datum" "neufassung_datum"
## [13] "fundstellentyp" "periodikum"
## [15] "zitstelle" "stand"
## [17] "aufh" "neuf"
## [19] "hinweis" "sonst"
## [21] "check_stand" "check_aufh"
## [23] "check_neuf" "check_hinweis"
## [25] "check_sonst" "zeichen"
## [27] "tokens" "typen"
## [29] "saetze" "doi_concept"
## [31] "doi_version" "version"
## [33] "lizenz"
```

```
names(meta.rechtsakte)
```

```
## [1] "doc_id" "jurabk"
## [3] "amtabk" "langue"
## [5] "kurzue" "ausfertigung_datum"
```

```
## [7] "ausfertigung_jahr"      "aenderung_datum"
## [9] "aufhebung_verkuendung_datum" "aufhebung_wirkung_datum"
## [11] "neufassung_datum"      "fundstellentyp"
## [13] "periodikum"            "zitstelle"
## [15] "stand"                 "aufh"
## [17] "neuf"                  "hinweis"
## [19] "sonst"                 "check_stand"
## [21] "check_aufh"            "check_neuf"
## [23] "check_hinweis"         "check_sonst"
## [25] "zeichen"               "tokens"
## [27] "typen"                 "saetze"
## [29] "doi_concept"           "doi_version"
## [31] "version"               "lizenz"
```

```
names(dt.meta)
```

```
## [1] "doc_id"                "jurabk"
## [3] "amtabk"                "langue"
## [5] "kurzue"                "ausfertigung_datum"
## [7] "ausfertigung_jahr"     "aenderung_datum"
## [9] "aufhebung_verkuendung_datum" "aufhebung_wirkung_datum"
## [11] "neufassung_datum"     "fundstellentyp"
## [13] "periodikum"           "zitstelle"
## [15] "stand"                "aufh"
## [17] "neuf"                 "hinweis"
## [19] "sonst"                "check_stand"
## [21] "check_aufh"           "check_neuf"
## [23] "check_hinweis"        "check_sonst"
## [25] "bulldate_original"    "bulldate_iso"
## [27] "doi_concept"          "doi_version"
## [29] "version"              "lizenz"
```

## 13 CSV-Dateien erstellen

### 13.1 Einzelnormen (Korpus)

#### 13.1.1 Name für CSV definieren

```
csvname.normen.gesamt <- paste(datasetname,  
                                datestamp,  
                                "DE_CSV_Einzelnormen_Datensatz.csv",  
                                sep = "_")
```

#### 13.1.2 Datensatz speichern

```
fwrite(dt.normen,  
       csvname.normen.gesamt,  
       na = "NA")
```

### 13.2 Einzelnormen (Metadaten)

#### 13.2.1 Name für CSV definieren

```
csvname.normen.meta <- paste(datasetname,  
                              datestamp,  
                              "DE_CSV_Einzelnormen_Metadaten.csv",  
                              sep = "_")
```

#### 13.2.2 Datensatz speichern

```
fwrite(meta.normen,  
       csvname.normen.meta,  
       na = "NA")
```

### 13.3 Rechtsakte (Korpus)

#### 13.3.1 Name für CSV definieren

```
csvname.rechtsakte.gesamt <- paste(datasetname,  
                                    datestamp,  
                                    "DE_CSV_Rechtsakte_Datensatz.csv",  
                                    sep = "_")
```

### 13.3.2 Datensatz speichern

```
fwrite(dt.rechtsakte,  
       csvname.rechtsakte.gesamt,  
       na = "NA")
```

## 13.4 Rechtsakte (Metadaten)

### 13.4.1 Name für CSV definieren

```
csvname.rechtsakte.meta <- paste(datasetname,  
                                  datestamp,  
                                  "DE_CSV_Rechtsakte_Metadaten.csv",  
                                  sep = "_")
```

### 13.4.2 Datensatz speichern

```
fwrite(meta.rechtsakte,  
       csvname.rechtsakte.meta,  
       na = "NA")
```

## 13.5 XML-Metadaten

Diese Datei unterscheidet sich von der Variante “DE\_CSV\_Rechtsakte\_Metadaten”, weil sie auch Rechtsakte enthält, die ohne Text veröffentlicht wurden. Die Differenz betrifft etwa 1000 Rechtsakte, ist also erheblich.

### 13.5.1 Name für CSV definieren

```
csvname.meta <- paste(datasetname,  
                      datestamp,  
                      "DE_CSV_MetadatenXML.csv",  
                      sep = "_")
```

### 13.5.2 Datensatz speichern

```
fwrite(dt.meta,  
       csvname.meta,  
       na = "NA")
```



## 14 Download der PDF-Dateien

### 14.1 Download durchführen

```
mcmapply(download.file,  
          download$links.pdf,  
          download$title.pdf)
```

### 14.2 Download-Ergebnis

#### 14.2.1 Anzahl herunterzuladender Dateien

```
download[, .N]
```

```
## [1] 6630
```

#### 14.2.2 Anzahl heruntergeladener Dateien

```
files.pdf <- list.files(pattern = "\\..pdf")  
length(files.pdf)
```

```
## [1] 6630
```

#### 14.2.3 Fehlbetrag

```
N.missing <- download[, .N] - length(files.pdf)  
print(N.missing)
```

```
## [1] 0
```

#### 14.2.4 Fehlende Dateien

```
missing <- setdiff(download$title.pdf,  
                   files.pdf)  
print(missing)
```

```
## character(0)
```

## 15 TXT-Dateien erstellen

An dieser Stelle wird der reine Text aus den PDF-Dateien extrahiert und ein zusätzliches Datei-Format (TXT) generiert. TXT-Dateien sind besonders für quantitative Analysten ohne XML-Kenntnisse ein lohnenswerter Einstieg und verringern die Hürde für die Arbeit mit dem Korpus.

```
files.pdf <- list.files(pattern = "\\..pdf",
                        ignore.case = TRUE)
```

### 15.1 Anzahl zu extrahierender Dateien

```
length(files.pdf)
```

```
## [1] 6630
```

### 15.2 Funktion anzeigen: f.dopar.pagenums

```
print(f.dopar.pagenums)
```

```
function(x, sum = FALSE, threads = detectCores()){
```

```
  print(paste("Parallel processing using", threads, "threads."))
```

```
  cl <- makeForkCluster(threads)
  registerDoParallel(cl)
```

```
  pagenums <- foreach(filename = x,
                      .combine = 'c',
                      .errorhandling = 'remove',
                      .inorder = TRUE) %dopar% {
    pdf_length(filename)
  }
```

```
  stopCluster(cl)
```

```
  if (sum == TRUE){
    sum.out <- sum(pagenums)
    print(paste("Total number of pages:", sum.out))
    return(sum.out)
  }else{
    return(pagenums)
  }
}
```

```
}
```

### 15.3 Anzahl zu extrahierender Seiten

```
sum(f.dopar.pagenums(files.pdf))
```

```
## [1] "Parallel processing using 16 threads."
```

```
## [1] 54682
```

### 15.4 Funktion anzeigen: f.dopar.pdfextract

```
print(f.dopar.pdfextract)
```

```
function(x, threads = detectCores()){
```

```
begin.extract <- Sys.time()

print(paste("Parallel processing using", threads, "threads. Begin at", begin.
  extract))

cl <- makeForkCluster(threads)
registerDoParallel(cl)

newnames <- gsub("\\.pdf",
  "\\ .txt",
  x)

result <- foreach(i = seq_along(x),
  .errorhandling = 'pass') %dopar% {

  ## Extract text layer from PDF
  pdf.extracted <- pdf_text(x[i])

  ## Write TXT to Disk
  write.table(pdf.extracted,
    newnames[i],
    quote = FALSE,
    row.names = FALSE,
    col.names = FALSE)

}

stopCluster(cl)

end.extract <- Sys.time()
duration.extract <- end.extract - begin.extract

print(paste0("Processed ",
```

```
length(result),  
" files. Runtime was ",  
round(duration.extract,  
      digits = 2),  
" ",  
attributes(duration.extract)$units,  
". Ended at ",  
end.extract, ".")  
  
return(result)
```

```
}
```

## 15.5 Text Extrahieren

```
f.dopar.pdfextract(files.pdf)
```

## 16 Download der EPUB-Dateien

### 16.1 Download durchführen

```
mcmapply(download.file,  
          download$links.epub,  
          download$title.epub)
```

### 16.2 Download-Ergebnis

#### 16.2.1 Anzahl herunterzuladender Dateien

```
download[, .N]
```

```
## [1] 6630
```

#### 16.2.2 Anzahl heruntergeladener Dateien

```
files.epub <- list.files(pattern = "\\\\.epub")  
length(files.epub)
```

```
## [1] 6630
```

#### 16.2.3 Fehlbetrag

```
N.missing <- download[, .N] - length(files.epub)  
print(N.missing)
```

```
## [1] 0
```

#### 16.2.4 Fehlende Dateien

```
missing <- setdiff(download$title.epub, files.epub)  
print(missing)
```

```
## character(0)
```

## 17 Dateigrößen analysieren

```
files.txt <- list.files(pattern = "\\..txt$", ignore.case = TRUE)
files.pdf <- list.files(pattern = "\\..pdf$", ignore.case = TRUE)
files.epub <- list.files(pattern = "\\..epub$", ignore.case = TRUE)

txt.MB <- file.size(files.txt) / 10^6
pdf.MB <- file.size(files.pdf) / 10^6
epub.MB <- file.size(files.epub) / 10^6
```

### 17.1 Gesamtgröße

#### 17.1.1 PDF-Dateien (MB)

```
sum(pdf.MB)
```

```
## [1] 627.1262
```

#### 17.1.2 EPUB-Dateien (MB)

```
sum(epub.MB)
```

```
## [1] 439.4726
```

#### 17.1.3 XML-Dateien (MB)

```
sum(xml.MB)
```

```
## [1] 261.7479
```

#### 17.1.4 TXT-Dateien (MB)

```
sum(txt.MB)
```

```
## [1] 195.3027
```

### 17.1.5 Korpus-Objekte in RAM (MB)

```
print(object.size(dt.normen),  
      standard = "SI",  
      humanReadable = TRUE,  
      units = "MB")
```

```
## 191.5 MB
```

```
print(object.size(dt.rechtsakte),  
      standard = "SI",  
      humanReadable = TRUE,  
      units = "MB")
```

```
## 147.2 MB
```

```
print(object.size(dt.meta),  
      standard = "SI",  
      humanReadable = TRUE,  
      units = "MB")
```

```
## 5.7 MB
```

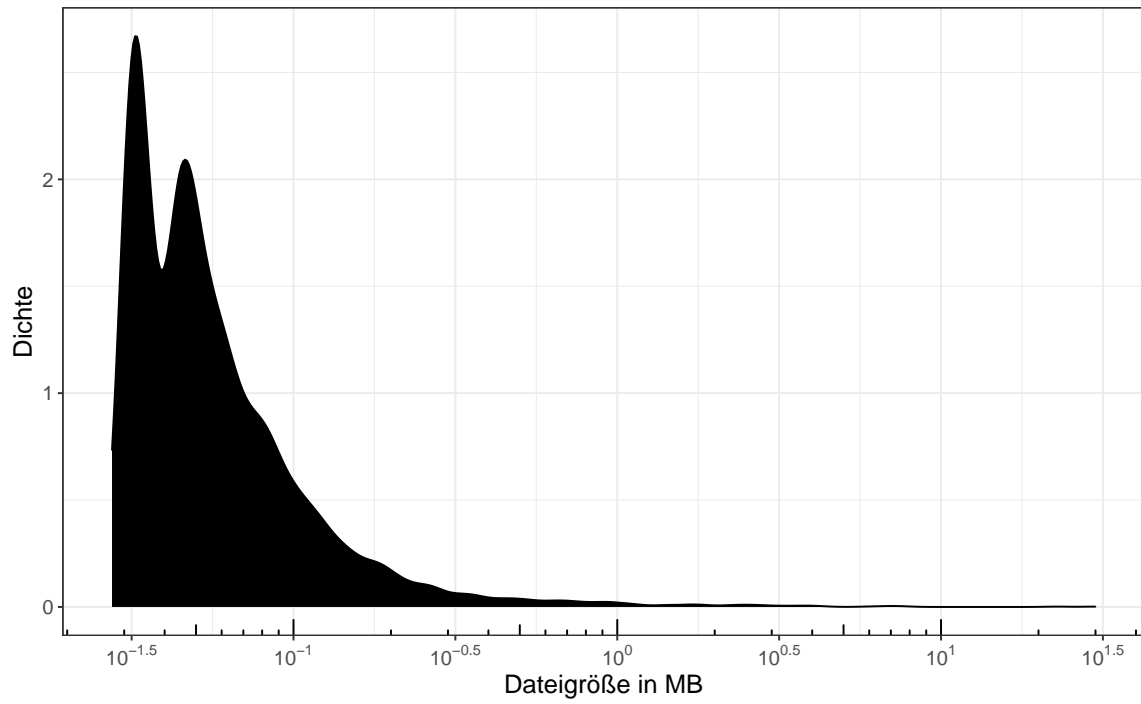
## 17.2 Verteilung der Dateigrößen (PDF)

```
dt.plot <- data.table(pdf.MB)
```

```
ggplot(data = dt.plot,
  aes(x = pdf.MB))+
  geom_density(fill = "black")+
  scale_x_log10(breaks = trans_breaks("log10", function(x) 10^x),
    labels = trans_format("log10", math_format(10^.x)))+
  annotation_logticks(sides = "b")+
  theme_bw()+
  labs(
    title = paste(datasetname,
      "| Version",
      datestamp,
      "| Verteilung der Dateigrößen (PDF)",
    caption = paste("DOI:",
      doi.version,
      "| S. Fobbe"),
    x = "Dateigröße in MB",
    y = "Dichte"
  )+
  theme(
    text = element_text(size = 14),
    plot.title = element_text(size = 14,
      face = "bold"),
    legend.position = "none",
    plot.margin = margin(10, 20, 10, 10)
  )
```



C-DBR | Version 2021-07-30 | Verteilung der Dateigrößen (PDF)



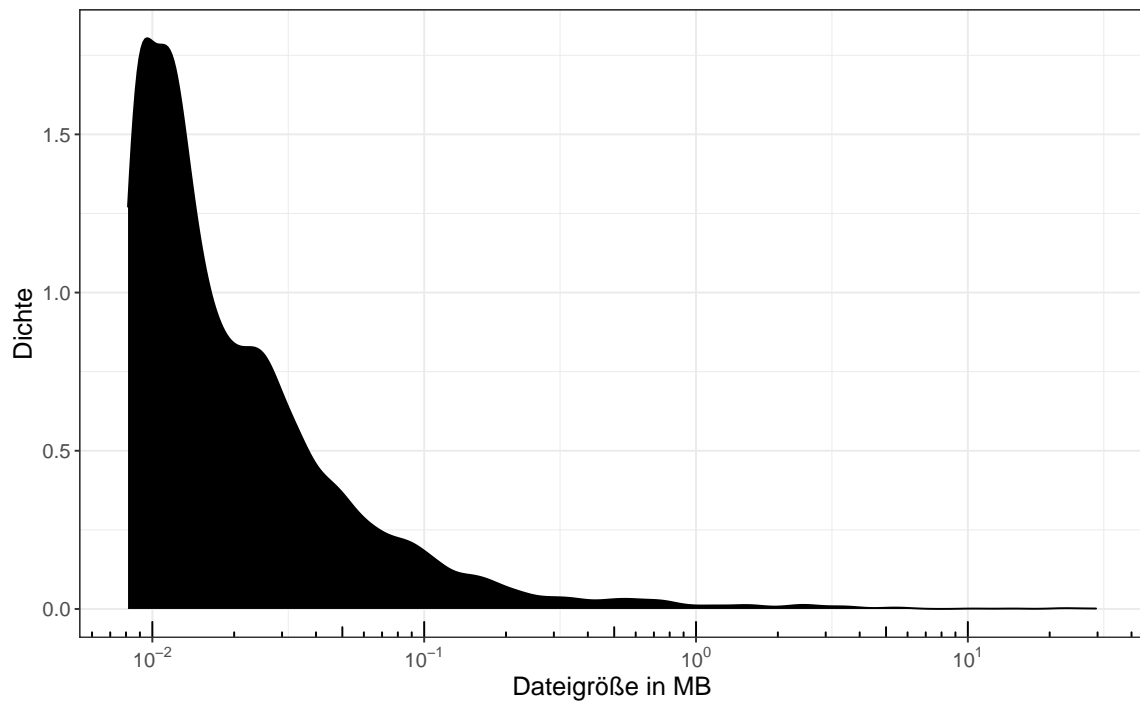
DOI: 10.5281/zenodo.5133932 | S. Fobbe

## 17.3 Verteilung der Dateigrößen (EPUB)

```
dt.plot <- data.table(epub.MB)
```

```
ggplot(data = dt.plot,
       aes(x = epub.MB))+
  geom_density(fill = "black")+
  scale_x_log10(breaks = trans_breaks("log10", function(x) 10^x),
               labels = trans_format("log10", math_format(10^.x)))+
  annotation_logticks(sides = "b")+
  theme_bw()+
  labs(
    title = paste(datasetsname,
                  "| Version",
                  datestamp,
                  "| Verteilung der Dateigrößen (EPUB)" ),
    caption = paste("DOI:",
                    doi.version,
                    "| S. Fobbe" ),
    x = "Dateigröße in MB",
    y = "Dichte"
  )+
  theme(
    text = element_text(size = 14),
    plot.title = element_text(size = 14,
                              face = "bold"),
    legend.position = "none",
    plot.margin = margin(10, 20, 10, 10)
  )
```

C-DBR | Version 2021-07-30 | Verteilung der Dateigrößen (EPUB)



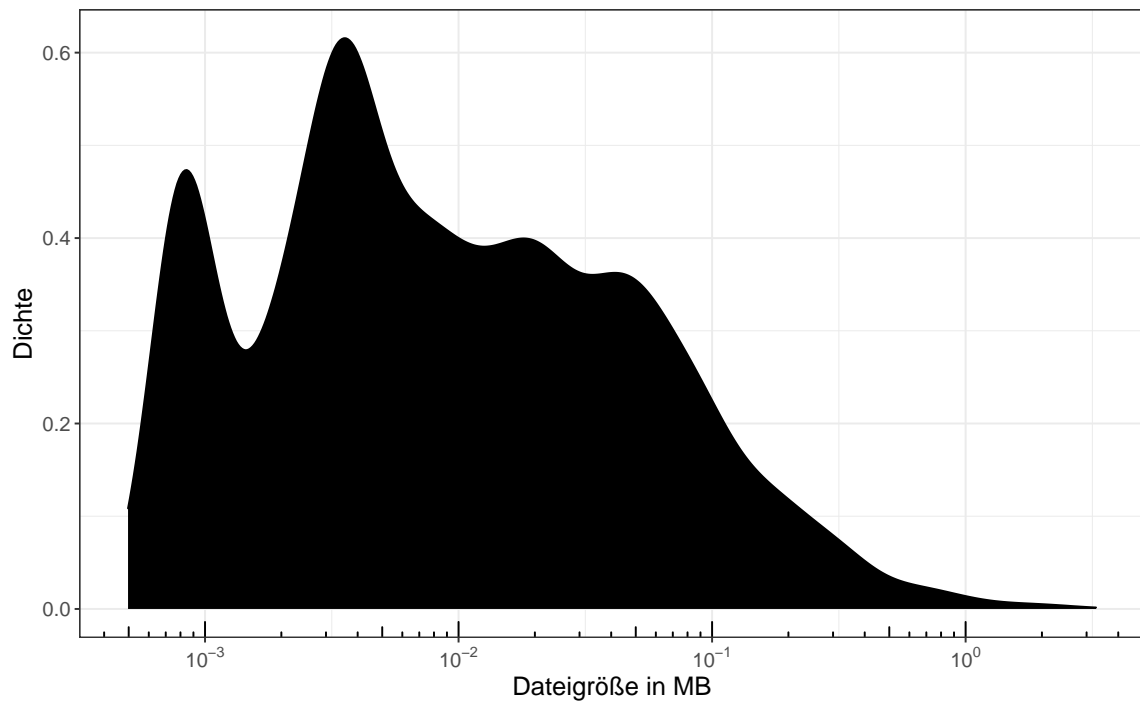
DOI: 10.5281/zenodo.5133932 | S. Fobbe

## 17.4 Verteilung der Dateigrößen (XML)

```
dt.plot <- data.table(xml.MB)
```

```
ggplot(data = dt.plot,
       aes(x = xml.MB))+
  geom_density(fill = "black")+
  scale_x_log10(breaks = trans_breaks("log10", function(x) 10^x),
               labels = trans_format("log10", math_format(10^.x)))+
  annotation_logticks(sides = "b")+
  theme_bw()+
  labs(
    title = paste(datasetname,
                  "| Version",
                  datestamp,
                  "| Verteilung der Dateigrößen (XML)" ),
    caption = paste("DOI:",
                    doi.version,
                    "| S. Fobbe"),
    x = "Dateigröße in MB",
    y = "Dichte"
  )+
  theme(
    text = element_text(size = 14),
    plot.title = element_text(size = 14,
                              face = "bold"),
    legend.position = "none",
    plot.margin = margin(10, 20, 10, 10)
  )
```

**C-DBR | Version 2021-07-30 | Verteilung der Dateigrößen (XML)**



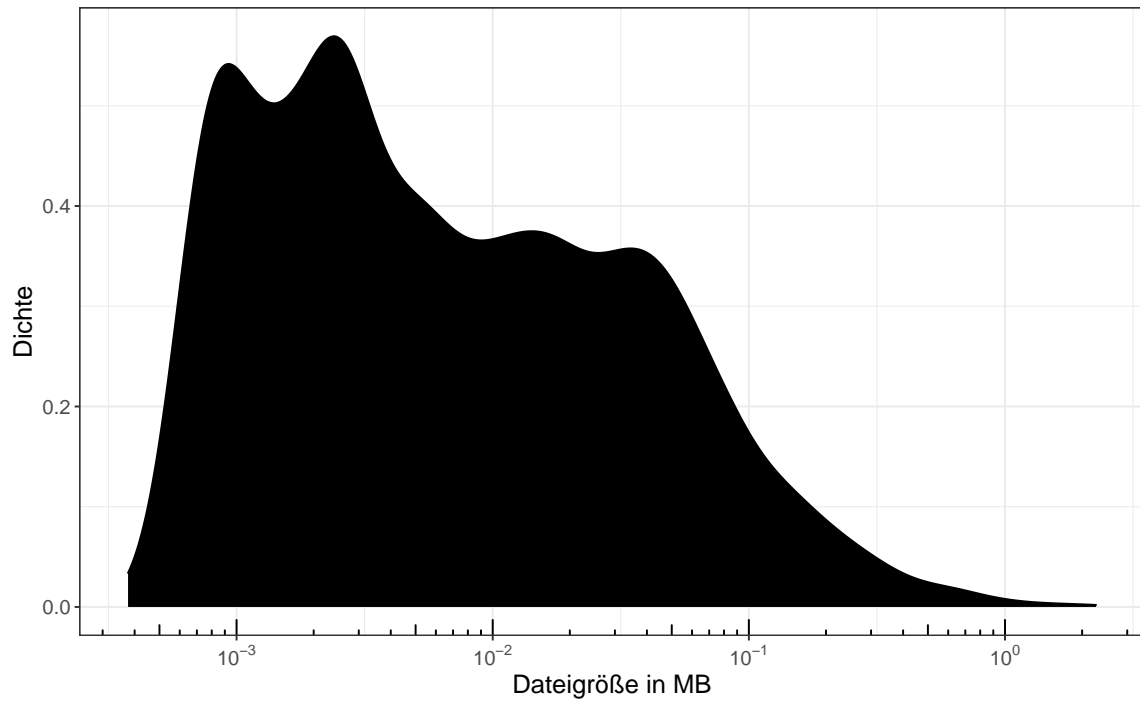
DOI: 10.5281/zenodo.5133932 | S. Fobbe

## 17.5 Verteilung der Dateigrößen (TXT)

```
dt.plot <- data.table(txt.MB)
```

```
ggplot(data = dt.plot,
       aes(x = txt.MB))+
  geom_density(fill = "black")+
  scale_x_log10(breaks = trans_breaks("log10", function(x) 10^x),
               labels = trans_format("log10", math_format(10^.x)))+
  annotation_logticks(sides = "b")+
  theme_bw()+
  labs(
    title = paste(datasetname,
                  "| Version",
                  datestamp,
                  "| Verteilung der Dateigrößen (TXT)",
    caption = paste("DOI:",
                    doi.version,
                    "| S. Fobbe"),
    x = "Dateigröße in MB",
    y = "Dichte"
  )+
  theme(
    text = element_text(size = 14),
    plot.title = element_text(size = 14,
                              face = "bold"),
    legend.position = "none",
    plot.margin = margin(10, 20, 10, 10)
  )
```

C-DBR | Version 2021-07-30 | Verteilung der Dateigrößen (TXT)



DOI: 10.5281/zenodo.5133932 | S. Fobbe

## 18 ZIP-Archive erstellen

### 18.1 Verpacken der CSV-Dateien

```
files.csv <- c(csvname.normen.gesamt,  
              csvname.normen.meta,  
              csvname.rechtsakte.gesamt,  
              csvname.rechtsakte.meta,  
              csvname.meta)  
  
csvnames.zip <- gsub(".csv",  
                    ".zip",  
                    files.csv)  
  
for (i in seq_along(files.csv)){  
  zip(csvnames.zip[i],  
      files.csv[i])  
}  
  
unlink(files.csv)
```

### 18.2 Verpacken der PDF-Dateien

```
files.pdf <- list.files(pattern = "\\..pdf$",  
                       ignore.case = TRUE)  
  
zip(paste(datasetname,  
          datestamp,  
          "DE_PDF_Datensatz.zip",  
          sep = "_"),  
    files.pdf)  
  
unlink(files.pdf)
```

### 18.3 Verpacken der TXT-Dateien

```
files.txt <- list.files(pattern = "\\..txt$",  
                       ignore.case = TRUE)  
  
zip(paste(datasetname,  
          datestamp,  
          "DE_TXT_Datensatz.zip",  
          sep = "_"),  
    files.txt)  
  
unlink(files.txt)
```



## 18.4 Verpacken der EPUB-Dateien

```
files.epub <- list.files(pattern = "\\..epub$",
                        ignore.case = TRUE)

zip(paste(datasetname,
          datestamp,
          "DE_EPUB_Datensatz.zip",
          sep = "_"),
    files.epub)

unlink(files.epub)
```

## 18.5 Verpacken der Netzwerk-Dateien

```
zip(paste0(datasetname,
          "_",
          datestamp,
          "_DE_Netzwerke.zip"),
    "Netzwerke")

unlink("Netzwerke",
      recursive = TRUE)
```

## 18.6 Verpacken der Analyse-Dateien

```
zip(paste0(datasetname,
          "_",
          datestamp,
          "_DE_",
          basename(outputdir),
          ".zip"),
    basename(outputdir))
```

## 18.7 Verpacken der Source-Dateien

```
files.source <- c(list.files(pattern = "Source"),
                  "buttons")

files.source <- grep("spin",
                    files.source,
                    value = TRUE,
                    ignore.case = TRUE,
                    invert = TRUE)
```

```
zip(paste(datasetname,  
          datestamp,  
          "Source_Code.zip",  
          sep = "_"),  
    files.source)
```

## 19 Kryptographische Hashes

Dieses Modul berechnet für jedes ZIP-Archiv zwei Arten von Hashes: SHA2-256 und SHA3-512. Mit diesen kann die Authentizität der Dateien geprüft werden und es wird dokumentiert, dass sie aus diesem Source Code hervorgegangen sind. Die SHA-2 und SHA-3 Algorithmen gelten derzeit als sicher und ein SHA3-Hash mit 512 bit Länge ist nach derzeitigem Wissen auch gegenüber quantenkryptoanalytischen Verfahren hinreichend resistent.

### 19.1 Liste der ZIP-Archive erstellen

```
files.zip <- list.files(pattern = "\\\\.zip$",  
                        ignore.case = TRUE)
```

### 19.2 Funktion anzeigen: f.dopar.multihashes

```
print(f.dopar.multihashes)
```

```
function(x, threads = detectCores()){
```

```
  print(paste("Parallel processing using", threads, "threads."))  
  
  begin <- Sys.time()  
  
  cl <- makeForkCluster(threads)  
  registerDoParallel(cl)  
  
  multihashes <- foreach(filename = x,  
                        .errorhandling = 'pass',  
                        .combine = 'rbind') %dopar% {  
  
    sha2.256 <- system2("openssl",  
                      paste("sha256",  
                            filename),  
                      stdout = TRUE)  
  
    sha2.256 <- gsub("^.*\\\\" = ",  
                  "",  
                  sha2.256)  
  
    sha3.512 <- system2("openssl",  
                      paste("sha3-512",  
                            filename),  
                      stdout = TRUE)  
  
    sha3.512 <- gsub("^.*\\\\" = ",  
                  "",  
                  sha3.512)
```

```

        out <- data.frame(filename,
                           sha2.256,
                           sha3.512)
        return(out)
    }
stopCluster(cl)

end <- Sys.time()
duration <- end - begin

print(paste0("Processed ",
             length(x),
             " files. Runtime was ",
             round(duration,
                   digits = 2),
             " ",
             attributes(duration)$units,
             "."))

return(multihashes)
}

```

### 19.3 Hashes berechnen

```
multihashes <- f.dopar.multihashes(files.zip)
```

```
## [1] "Parallel processing using 16 threads."
## [1] "Processed 13 files. Runtime was 3.16 secs."
```

### 19.4 In Data Table umwandeln

```
setDT(multihashes)
```

### 19.5 Index hinzufügen

```
multihashes$index <- seq_len(multihashes[,.N])
```

### 19.6 Hashes in CSV-Datei speichern

```
fwrite(multihashes,
       paste(datasetname,
             datestamp,
```

```
      "KryptographischeHashes.csv",  
      sep = "_"),  
na = "NA")
```

## 19.7 Leerzeichen hinzufügen um Zeilenumbruch zu ermöglichen

```
multihashes$sha3.512 <- paste(substr(multihashes$sha3.512, 1, 64),  
                              substr(multihashes$sha3.512, 65, 128))
```

## 19.8 In Bericht anzeigen

```
kable(multihashes[,.(index,filename)],  
      format = "latex",  
      align = c("p{1cm}", "p{13cm}"),  
      booktabs = TRUE,  
      longtable = TRUE)
```

index	filename
1	C-DBR_2021-07-30_DE_ANALYSE.zip
2	C-DBR_2021-07-30_DE_CSV_Einzelnormen_Datensatz.zip
3	C-DBR_2021-07-30_DE_CSV_Einzelnormen_Metadaten.zip
4	C-DBR_2021-07-30_DE_CSV_MetadatenXML.zip
5	C-DBR_2021-07-30_DE_CSV_Rechtsakte_Datensatz.zip
6	C-DBR_2021-07-30_DE_CSV_Rechtsakte_Metadaten.zip
7	C-DBR_2021-07-30_DE_EPUB_Datensatz.zip
8	C-DBR_2021-07-30_DE_Netzwerke.zip
9	C-DBR_2021-07-30_DE_PDF_Datensatz.zip
10	C-DBR_2021-07-30_DE_TXT_Datensatz.zip
11	C-DBR_2021-07-30_DE_XML_Anlagen.zip
12	C-DBR_2021-07-30_DE_XML_Datensatz.zip
13	C-DBR_2021-07-30_Source_Code.zip

```
kable(multihashes[,.(index,sha2.256)],
      format = "latex",
      align = c("c", "p{13cm}"),
      booktabs = TRUE,
      longtable = TRUE)
```

index	sha2.256
1	90c15afc17d0220449182a07c28239ccfa925394779bf02def63c887f095fe22
2	9c935acba487913c0711909f40ce8cdc370f63fa078adf4a61f342777394b0b6
3	8cd0471eae2f2c9639a18f5215c5ca0935d40f8bf5edfee19f9abd2f63ea94e
4	462335d58a8723f4734a582eabc85279205c4e542f87225b51431028c419fc4d
5	ef5c69d3ffbb7c664687c4ae2c0db27802d01e80d5431700c6311f5e264975c0
6	aed6fba6abca97fdbb361f2a518909524a3f59c7d628bd220a1f7cc8f04b9897
7	c26f1bcfec5c111fbf0a66d3abbedf8491aab4b762aada421e5b74e781622b1c
8	0be12685f3ec875fb1fe79ec00ede563ccde17537dff539fd40c002642484e3a
9	aa09e49d9b9091658f573a3805a7fcd9543997542a1bfc6672add0e1c7d22e94
10	beb6127609237bb39df69b4206afaaf8cd56836fe7bf771301dc9d529b42bbb1
11	721a4d9e0e81e3341510741ab9a7834847ac88d3508bd9a008056c9d343b9478
12	37e6447ade1956eae274b248dc117477a5ada15e20b4d9b2c94077f96ba80d1c
13	43fc072f93214d4e71a098e890281b25587c76562b8fb181445c8e905b8a6810

```
kable(multihashes[,.(index,sha3.512)],
      format = "latex",
      align = c("c", "p{13cm}"),
      booktabs = TRUE,
      longtable = TRUE)
```

index	sha3.512
1	d676999c425d4def117d1cf0a9d22981c5293e15d3dd4c68ea90051218a83fd78045faff64bb45a0efe71ea2330793c19595621e899d458cd129ad0e4d5592e0
2	2ec6ceaf3213f86863f324537f898ec6b05ab260cb06631f481ee79eb2c635f2e49b3bb3dc8358ef07f1ff020c00af13e01776c9184a2bb0d132bc16080cff6b
3	844019f50aa9c52f0a26a202e8f13a97d5440a393dc97c6a40d2205d5843b74574e472403ed5793b3f51880ffc962aee4285dc65d487fd0e5f9f52238c738172
4	9bd7548e6c9d5fbf54deca4800a17ac867305aab864755e6a9e96a47fd43773395eaac098b878afc46c2be9f79a6af516d37cbd15d39e7c382d8a40c66615855
5	8c9a0e38712999e098dc5a67453a4dfefb6ba1190dece896fe28c35afd1dea2488a57988093dcfcd1a61a763b1575ffc78f59135726d1d4ef5a867a6be900c71
6	fc9d30df0daa6aa9fced2e64056edeb1486d3712cba438e35a02766971f6341e3dee6c020885aa63ca3ada01eaf36c32804299998f6e06af406a60ba8a67763
7	47c6bfe26e7236584e3647a758845ca1c181b9948a96cd0979d617cd3eee4c04fa743f97b862d7d403227ef512315827dccc120dc0e235137d036a6f2e7d4612
8	c435a71b6088a0a151b355d7120f924ddfa6b3aaa02d315a3b3d802561e3cc1c4a1459a31592e1d49863e375db9b6d0d59e6731830647054ca01dcd5c7afb50f
9	5249e9bdc9c2c69a0b5451cabe24581cbd8ad69f7b8cd63ba7db32e7b8b494f3c77c0ff698d9a5af162b493b7101d2100056e3917bd921405366c1b7f48f6054
10	13f15956db7d49718af1217ff328d0d156d3b1482056f2ac335d5d22821327d6e72eec33131d25338411acb6f68028d2bf3cc8514fa33d5fe50d888f28f39351
11	7caae75de6c204129f0a7ff177ce474e65a97769262e820070c445f66bb1ce8c4dc288432e8f7f27b46e26c6a7b31762e312057b24c7201bb1a0f3b2e21c0934
12	bfd40ffe1016277cafeefceb449c086da52f5ad127c6d250f1a4542ada25f40aa72bc7a6e93e29d8ac1ec98ac958df614c82e05fef0b63759dfea52d4268a18d
13	109cf8535c9aafcd325360381a70712bf506105cdf956eddb5bee5fdb3e11a98790005bf0c6b4b5d91e7a31eb7787465f2a48371af6d59e1ef9f06834cbb8269



## 20 Abschluss

### 20.1 Datumsstempel

```
print(datestamp)
```

```
## [1] "2021-07-30"
```

### 20.2 Datum und Uhrzeit (Anfang)

```
print(begin.script)
```

```
## [1] "2021-07-30 18:45:34 CEST"
```

### 20.3 Datum und Uhrzeit (Ende)

```
end.script <- Sys.time()  
print(end.script)
```

```
## [1] "2021-07-30 19:18:42 CEST"
```

### 20.4 Laufzeit des gesamten Skripts

```
print(end.script - begin.script)
```

```
## Time difference of 33.12727 mins
```

### 20.5 Warnungen

```
warnings()
```

## 21 Parameter für strenge Replikationen

```
system2("openssl", "version", stdout = TRUE)
```

```
## [1] "OpenSSL 1.1.1k FIPS 25 Mar 2021"
```

```
sessionInfo()
```

```
## R version 4.0.5 (2021-03-31)
## Platform: x86_64-redhat-linux-gnu (64-bit)
## Running under: Fedora 34 (Workstation Edition)
##
## Matrix products: default
## BLAS/LAPACK: /usr/lib64/libflexiblas.so.3.0
##
## locale:
##  [1] LC_CTYPE=en_US.utf8      LC_NUMERIC=C
##  [3] LC_TIME=en_US.utf8      LC_COLLATE=en_US.utf8
##  [5] LC_MONETARY=en_US.utf8  LC_MESSAGES=en_US.utf8
##  [7] LC_PAPER=en_US.utf8     LC_NAME=C
##  [9] LC_ADDRESS=C            LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_US.utf8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] parallel stats      graphics grDevices utils      datasets methods
## [8] base
##
## other attached packages:
##  [1] qgraph_1.6.9      ggraph_2.0.5      igraph_1.2.6      openssl_1.4.4
##  [5] scales_1.1.1      quanteda_3.0.0     data.table_1.14.0 ggplot2_3.3.3
##  [9] doParallel_1.0.16 iterators_1.0.13 foreach_1.5.1     pdftools_3.0.1
## [13] kableExtra_1.3.4  knitr_1.33        xml2_1.3.2        rvest_1.0.0
##
## loaded via a namespace (and not attached):
##  [1] nlme_3.1-152      webshot_0.5.2      RColorBrewer_1.1-2
##  [4] httr_1.4.2        tools_4.0.5        backports_1.2.1
##  [7] utf8_1.2.1        R6_2.5.0           rpart_4.1-15
## [10] Hmisc_4.5-0       colorspace_2.0-0   nnet_7.3-15
## [13] withr_2.4.2       mnormt_2.0.2       tidyselect_1.1.1
## [16] gridExtra_2.3     curl_4.3.1         compiler_4.0.5
## [19] fdrtool_1.2.16    htmlTable_2.1.0    labeling_0.4.2
## [22] checkmate_2.0.0   psych_2.1.3        pbapply_1.4-3
## [25] askpass_1.1       pbivnorm_0.6.0     systemfonts_1.0.2
## [28] stringr_1.4.0     digest_0.6.27      foreign_0.8-81
## [31] rmarkdown_2.8     svglite_2.0.0      base64enc_0.1-3
## [34] jpeg_0.1-8.1      pkgconfig_2.0.3    htmltools_0.5.1.1
## [37] highr_0.9         htmlwidgets_1.5.3  rlang_0.4.11
## [40] rstudioapi_0.13   farver_2.1.0       generics_0.1.0
## [43] gtools_3.8.2      dplyr_1.0.6        magrittr_2.0.1
```

## [46]	Formula_1.2-4	Matrix_1.3-3	Rcpp_1.0.6
## [49]	munSELL_0.5.0	fansi_0.4.2	abind_1.4-5
## [52]	viridis_0.6.1	lifecycle_1.0.0	stringi_1.5.3
## [55]	yaml_2.2.1	MASS_7.3-53.1	plyr_1.8.6
## [58]	lavaan_0.6-8	grid_4.0.5	ggrepel_0.9.1
## [61]	crayon_1.4.1	lattice_0.20-41	graphlayouts_0.7.1
## [64]	splines_4.0.5	magick_2.7.2	tmvnsim_1.0-2
## [67]	pillar_1.6.1	corpcor_1.6.9	stats4_4.0.5
## [70]	reshape2_1.4.4	codetools_0.2-18	stopwords_2.2
## [73]	fastmatch_1.1-0	glue_1.4.2	evaluate_0.14
## [76]	qpdf_1.1	latticeExtra_0.6-29	RcppParallel_5.1.4
## [79]	selectr_0.4-2	png_0.1-7	vctrs_0.3.8
## [82]	tweenr_1.0.2	gtable_0.3.0	purrr_0.3.4
## [85]	polycIip_1.10-0	tidyr_1.1.3	xfun_0.22
## [88]	ggforce_0.3.3	tidygraph_1.2.0	glasso_1.11
## [91]	survival_3.2-10	viridisLite_0.4.0	tibble_3.0.4
## [94]	cluster_2.1.1	ellipsis_0.3.2	

## Literaturverzeichnis

- Analytics, Revolution, and Steve Weston. 2020. *Iterators: Provides Iterator Construct*. <https://github.com/RevolutionAnalytics/iterators>.
- Benoit, Kenneth, Kohei Watanabe, Haiyan Wang, Paul Nulty, Adam Obeng, Stefan Müller, and Akitaka Matsuo. 2018. “Quanteda: An R Package for the Quantitative Analysis of Textual Data.” *Journal of Open Source Software* 3 (30): 774. <https://doi.org/10.21105/joss.00774>.
- Benoit, Kenneth, Kohei Watanabe, Haiyan Wang, Paul Nulty, Adam Obeng, Stefan Müller, Akitaka Matsuo, and William Lowe. 2021. *Quanteda: Quantitative Analysis of Textual Data*. <https://quanteda.io>.
- Corporation, Microsoft, and Steve Weston. 2020. *DoParallel: Foreach Parallel Adaptor for the Parallel Package*. <https://CRAN.R-project.org/package=doParallel>.
- Csardi, Gabor, and Tamas Nepusz. 2006. “The Igraph Software Package for Complex Network Research.” *InterJournal Complex Systems*: 1695. <https://igraph.org>.
- Dowle, Matt, and Arun Srinivasan. 2021. *Data.table: Extension of ‘Data.frame’*. <https://CRAN.R-project.org/package=data.table>.
- Epskamp, Sacha, Giulio Costantini, Jonas Haslbeck, and Adela Isvoranu. 2021. *Qgraph: Graph Plotting Methods, Psychometric Data Visualization and Graphical Model Estimation*. <https://CRAN.R-project.org/package=qgraph>.
- Epskamp, Sacha, Angélique O. J. Cramer, Lourens J. Waldorp, Verena D. Schmittmann, and Denny Borsboom. 2012. “qgraph: Network Visualizations of Relationships in Psychometric Data.” *Journal of Statistical Software* 48 (4): 1–18.
- file., See AUTHORS. 2020. *Igraph: Network Analysis and Visualization*. <https://igraph.org>.
- Ooms, Jeroen. 2021a. *Openssl: Toolkit for Encryption, Signatures and Certificates Based on Openssl*. <https://github.com/jeroen/openssl>.
- . 2021b. *Pdftools: Text Extraction, Rendering and Converting of Pdf Documents*. <https://CRAN.R-project.org/package=pdfutils>.
- Pedersen, Thomas Lin. 2021. *Ggraph: An Implementation of Grammar of Graphics for Graphs and Networks*. <https://CRAN.R-project.org/package=ggraph>.
- R Core Team. 2021. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. <https://www.R-project.org/>.
- Revolution Analytics, and Steve Weston. n.d. *Foreach: Provides Foreach Looping Construct*.
- Wickham, Hadley. 2016. *Ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. <https://ggplot2.tidyverse.org>.
- . 2021. *Rvest: Easily Harvest (Scrape) Web Pages*. <https://CRAN.R-project.org/package=rvest>.
- Wickham, Hadley, Winston Chang, Lionel Henry, Thomas Lin Pedersen, Kohske Takahashi, Claus Wilke, Kara Woo, Hiroaki Yutani, and Dewey Dunnington. 2020. *Ggplot2: Create Elegant Data Visualisations Using the Grammar of Graphics*. <https://CRAN.R-project.org/package=ggplot2>.

- Wickham, Hadley, Jim Hester, and Jeroen Ooms. 2020. *Xml2: Parse Xml*. <https://CRAN.R-project.org/package=xml2>.
- Wickham, Hadley, and Dana Seidel. 2020. *Scales: Scale Functions for Visualization*. <https://CRAN.R-project.org/package=scales>.
- Xie, Yihui. 2014. “Knitr: A Comprehensive Tool for Reproducible Research in R.” In *Implementing Reproducible Computational Research*, edited by Victoria Stodden, Friedrich Leisch, and Roger D. Peng. Chapman; Hall/CRC. <http://www.crcpress.com/product/isbn/9781466561595>.
- . 2015. *Dynamic Documents with R and Knitr*. 2nd ed. Boca Raton, Florida: Chapman; Hall/CRC. <https://yihui.org/knitr/>.
- . 2021. *Knitr: A General-Purpose Package for Dynamic Report Generation in R*. <https://yihui.org/knitr/>.
- Zhu, Hao. 2021. *KableExtra: Construct Complex Table with Kable and Pipe Syntax*. <https://CRAN.R-project.org/package=kableExtra>.