

Corpus des Deutschen Bundesrechts (C-DBR)

COMPILATION REPORT

Version 2021-09-16

License MIT-0

DOI: [10.5281/zenodo.5510687](https://doi.org/10.5281/zenodo.5510687)

Titel	Source Code des »Corpus des Deutschen Bundesrechts«
Abkürzung	C-DBR-Source
Autor	Seán Fobbe
Version	2021-09-16
Download	https://doi.org/10.5281/zenodo.5510687
Lizenz	MIT No Attribution (MIT-0)

Zitiervorschlag

Seán Fobbe (2021). Source Code des »Corpus des Deutschen Bundesrechts« (C-DBR-Source). Version 2021-09-16. Zenodo. DOI: 10.5281/zenodo.5510687.

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 2021-09-16. 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

1	Einleitung	9
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
2	Parameter	12
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
3	Vorbereitung	14
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
4	Funktionen definieren	18
4.1	Transformation von Gliederungs-Metadaten	18
4.2	NA in leere Listen-Elemente einsetzen	18
4.3	Erstellen von Titel- und Bezeichnungshierarchien	18
5	Download vorbereiten	20
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
6	Verarbeitung der DTD und XML-Dateien mit Anlagen	25
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
7	Frequenztabellen erstellen: Einzelnormen	46
7.1	Funktion anzeigen: f.fast.freqtable	46
7.2	Liste zu prüfender Variablen	47
7.3	Frequenztabellen erstellen	47
8	Frequenztabellen erstellen: Rechtsakte	107
8.1	Variablen ignorieren	107
8.2	Liste zu prüfender Variablen	107
8.3	Frequenztabellen erstellen	107
9	Frequenztabellen erstellen: XML-Metadaten	116

9.1	Liste zu prüfender Variablen	116
9.2	Frequenztabellen erstellen	116
10	Frequenztabellen visualisieren	126
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
11	Korpus-Analytik	138
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
12	Kontrolle der Variablen	159
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
13	CSV-Dateien erstellen	167
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
14	Download der PDF-Dateien	169
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
15	TXT-Dateien erstellen	170
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
16	Download der EPUB-Dateien	173
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
17	Dateigrößen analysieren	174
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
18	ZIP-Archive erstellen	184
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
19	Kryptographische Hashes	187
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
20	Abschluss	193
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
21	Parameter für strenge Replikationen	194
	Literaturverzeichnis	196

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 \LaTeX und alle in der Präambel-TEX-Datei angegebenen \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.5510458" # checked
```

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] "2021-09-16"
```

3.2 Datum und Uhrzeit (Beginn)

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

```
## [1] "2021-09-16 00:08:07 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: 3.1.0  
## 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 1.1.1l FIPS 24 Aug 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

Beispiel Bezeichnungshierarchie: Buch 2 | Abschnitt 8 | Titel 5 | Untertitel 2 | Kapitel 5 |
Unterkapitel 4

```
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.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_warnings()` to see where this warning was generated.
```

```
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

```
ID <- gsub("\\.epub",
          "",
          filenames.epub)

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

```
colnames(conctable) <- c("ID",  
                        "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] 6638
```

5.19.2 Insgesamt

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

```
## [1] 19914
```


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” (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] 6638
```

6.3.2 Anzahl heruntergeladener Dateien

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

```
## [1] 6638
```

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) / 106
```

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 3.82471 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 28.27967 secs
```

6.10.8 Variablen-Name für Ausfertigungsdatum anpassen

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

6.10.9 Variable “fundstellentyp” anpassen

```
dt.meta[grep("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("+",
                      "-",
                      jurabk),
                  "_",
                  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] 6627
```

```

### 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	0.0009997	0.00	0.00
BAnz	1236	1.2355923	1.24	1.24
BGBl	2292	2.2912439	2.29	3.53
BGBl I	84014	83.9862845	83.99	87.51
BGBl II	3254	3.2529265	3.25	90.77
GBl DDR	233	0.2329231	0.23	91.00
GBl DDR I	616	0.6157968	0.62	91.62
GBl DDR II	43	0.0429858	0.04	91.66
GVBl BE	9	0.0089970	0.01	91.67
NV	15	0.0149951	0.01	91.68
RAnz	38	0.0379875	0.04	91.72
RGBl	5662	5.6601322	5.66	97.38
RGBl I	1640	1.6394590	1.64	99.02
RGBl II	347	0.3468855	0.35	99.37
RMBl	230	0.2299241	0.23	99.60
VOBl BrZ	19	0.0189937	0.02	99.62
VkBl	106	0.1059650	0.11	99.72
WiGBl	122	0.1219598	0.12	99.84
ZBl	63	0.0629792	0.06	99.91
eBAnz	52	0.0519828	0.05	99.96

(continued)

periodikum	N	exactpercent	roundedpercent	cumulpercent
Öff Anz	41	0.0409865	0.04	100.00
Total	100033	100.0000000	100.00	100.00

Frequency Table for Variable: fundstellentyp

1 unique value(s) detected.

fundstellentyp	N	exactpercent	roundedpercent	cumulpercent
amtlich	100033	100	100	100
Total	100033	100	100	100

Frequency Table for Variable: check_neuf

2 unique value(s) detected.

check_neuf	N	exactpercent	roundedpercent	cumulpercent
NA	76008	75.98293	75.98	75.98
ja	24025	24.01707	24.02	100.00
Total	100033	100.00000	100.00	100.00

Frequency Table for Variable: check_aufh

2 unique value(s) detected.

check_aufh	N	exactpercent	roundedpercent	cumulpercent
NA	97770	97.737747	97.74	97.74
ja	2263	2.262253	2.26	100.00
Total	100033	100.000000	100.00	100.00

Frequency Table for Variable: check_sonst

2 unique value(s) detected.

check_sonst	N	exactpercent	roundedpercent	cumulpercent
NA	82478	82.45079	82.45	82.45
ja	17555	17.54921	17.55	100.00
Total	100033	100.00000	100.00	100.00

Frequency Table for Variable: check_hinweis

2 unique value(s) detected.

check_hinweis	N	exactpercent	roundedpercent	cumulpercent
NA	64139	64.11784	64.12	64.12
ja	35894	35.88216	35.88	100.00
Total	100033	100.00000	100.00	100.00

Frequency Table for Variable: check_stand

2 unique value(s) detected.

check_stand	N	exactpercent	roundedpercent	cumulpercent
NA	22695	22.68751	22.69	22.69
ja	77338	77.31249	77.31	100.00
Total	100033	100.00000	100.00	100.00

Frequency Table for Variable: gliederungskennzahl

1465 unique value(s) detected.

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
000	77	0.0769746	0.08	0.08
000010	12	0.0119960	0.01	0.09
000020	18	0.0179941	0.02	0.11
000030	17	0.0169944	0.02	0.12
000040	17	0.0169944	0.02	0.14
000050	18	0.0179941	0.02	0.16
000060	10	0.0099967	0.01	0.17
000070	3	0.0029990	0.00	0.17
000080	2	0.0019993	0.00	0.17
000090	1	0.0009997	0.00	0.17
000100	2	0.0019993	0.00	0.18
000110	3	0.0029990	0.00	0.18
000120	5	0.0049984	0.00	0.18
001000	2	0.0019993	0.00	0.19
001010	1	0.0009997	0.00	0.19
001020	3	0.0029990	0.00	0.19
001030	1	0.0009997	0.00	0.19
001040	1	0.0009997	0.00	0.19
001051	1	0.0009997	0.00	0.19
001060	2	0.0019993	0.00	0.20
001080	1	0.0009997	0.00	0.20
001090	1	0.0009997	0.00	0.20
001100	1	0.0009997	0.00	0.20
001101	1	0.0009997	0.00	0.20
001110	1	0.0009997	0.00	0.20
001120	5	0.0049984	0.00	0.21
010	6128	6.1259784	6.13	6.33
010010	908	0.9077005	0.91	7.24
010010000010	1	0.0009997	0.00	7.24
010010000020	1	0.0009997	0.00	7.24

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
010010010	118	0.1179611	0.12	7.36
010010020	117	0.1169614	0.12	7.48
010010020010	3	0.0029990	0.00	7.48
010010020010010	37	0.0369878	0.04	7.52
010010020010020	24	0.0239921	0.02	7.54
010010020020	12	0.0119960	0.01	7.55
010010020030	1	0.0009997	0.00	7.55
010010030	28	0.0279908	0.03	7.58
010010030010	7	0.0069977	0.01	7.59
010010030020	16	0.0159947	0.02	7.60
010010030030	12	0.0119960	0.01	7.62
010010030040	12	0.0119960	0.01	7.63
010010040	35	0.0349885	0.03	7.66
010010050	11	0.0109964	0.01	7.67
010010060	6	0.0059980	0.01	7.68
010010070010	10	0.0099967	0.01	7.69
010010070020	5	0.0049984	0.00	7.70
010020	792	0.7917387	0.79	8.49
010020010	135	0.1349555	0.13	8.62
010020010000030	1	0.0009997	0.00	8.62
010020010000040	1	0.0009997	0.00	8.62
010020010000050	1	0.0009997	0.00	8.63
010020010000060	1	0.0009997	0.00	8.63
010020020	132	0.1319565	0.13	8.76
010020020000070	1	0.0009997	0.00	8.76
010020020000080	1	0.0009997	0.00	8.76
010020020000090	1	0.0009997	0.00	8.76
010020020000100	1	0.0009997	0.00	8.76
010020020000110	1	0.0009997	0.00	8.76

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
010020020000120	1	0.0009997	0.00	8.76
010020020010	16	0.0159947	0.02	8.78
010020020020	19	0.0189937	0.02	8.80
010020020030	9	0.0089970	0.01	8.81
010020020040	14	0.0139954	0.01	8.82
010020020050	18	0.0179941	0.02	8.84
010020030	103	0.1029660	0.10	8.94
010020030000130	1	0.0009997	0.00	8.94
010020030000140	1	0.0009997	0.00	8.95
010020030000170	1	0.0009997	0.00	8.95
010020030000171	1	0.0009997	0.00	8.95
010020030000172	1	0.0009997	0.00	8.95
010020030000190	1	0.0009997	0.00	8.95
010020030000200	1	0.0009997	0.00	8.95
010020030000210	1	0.0009997	0.00	8.95
010020030000220	1	0.0009997	0.00	8.95
010020030000230	1	0.0009997	0.00	8.95
010020030000240	1	0.0009997	0.00	8.95
010020030010	3	0.0029990	0.00	8.96
010020030020	2	0.0019993	0.00	8.96
010020030030	2	0.0019993	0.00	8.96
010020040	70	0.0699769	0.07	9.03
010020040000250	1	0.0009997	0.00	9.03
010020040000260	1	0.0009997	0.00	9.03
010020050	53	0.0529825	0.05	9.09
010020050020000380	1	0.0009997	0.00	9.09
010020050020000390	1	0.0009997	0.00	9.09
010020050020000400	1	0.0009997	0.00	9.09
010020050020000410	1	0.0009997	0.00	9.09

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
010020050020000420	1	0.0009997	0.00	9.09
010020060	44	0.0439855	0.04	9.13
010020060000430	1	0.0009997	0.00	9.14
010020060000440	1	0.0009997	0.00	9.14
010020060000450	1	0.0009997	0.00	9.14
010020060000460	1	0.0009997	0.00	9.14
010020070	31	0.0309898	0.03	9.17
010020070010000461	1	0.0009997	0.00	9.17
010020070020000462	1	0.0009997	0.00	9.17
010020070020000463	1	0.0009997	0.00	9.17
010020070030000464	1	0.0009997	0.00	9.17
010020070040000465	1	0.0009997	0.00	9.17
010020080	41	0.0409865	0.04	9.22
010020090	1	0.0009997	0.00	9.22
010030	526	0.5258265	0.53	9.74
010030000470	1	0.0009997	0.00	9.74
010030000480	1	0.0009997	0.00	9.74
010030010	132	0.1319565	0.13	9.88
010030010010	8	0.0079974	0.01	9.88
010030010020	14	0.0139954	0.01	9.90
010030010030	20	0.0199934	0.02	9.92
010030010040	12	0.0119960	0.01	9.93
010030010050	4	0.0039987	0.00	9.93
010030010060	8	0.0079974	0.01	9.94
010030010070	8	0.0079974	0.01	9.95
010030010080	4	0.0039987	0.00	9.95
010030010380	2	0.0019993	0.00	9.96
010030010400	4	0.0039987	0.00	9.96
010030010440	1	0.0009997	0.00	9.96

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
010030010450	3	0.0029990	0.00	9.96
010030020	84	0.0839723	0.08	10.05
010030020010	86	0.0859716	0.09	10.13
010030020020	40	0.0399868	0.04	10.17
010030020030	12	0.0119960	0.01	10.19
010030020040	12	0.0119960	0.01	10.20
010030020050	9	0.0089970	0.01	10.21
010030020060	18	0.0179941	0.02	10.23
010030030	70	0.0699769	0.07	10.30
010030030010	9	0.0089970	0.01	10.30
010030030020	9	0.0089970	0.01	10.31
010030030030	6	0.0059980	0.01	10.32
010030030040	1	0.0009997	0.00	10.32
010030040	39	0.0389871	0.04	10.36
010030050	39	0.0389871	0.04	10.40
010030060	6	0.0059980	0.01	10.40
010030060010	16	0.0159947	0.02	10.42
010030060020	8	0.0079974	0.01	10.43
010030060030	3	0.0029990	0.00	10.43
010030060040	3	0.0029990	0.00	10.43
010030060050	2	0.0019993	0.00	10.44
010030070	17	0.0169944	0.02	10.45
010031	2	0.0019993	0.00	10.46
010040	341	0.3408875	0.34	10.80
010040010	83	0.0829726	0.08	10.88
010040020	69	0.0689772	0.07	10.95
010040020010	3	0.0029990	0.00	10.95
010040020020	5	0.0049984	0.00	10.96
010040020030	2	0.0019993	0.00	10.96

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
010040030	42	0.0419861	0.04	11.00
010040030010	8	0.0079974	0.01	11.01
010040030020	3	0.0029990	0.00	11.01
010040040	21	0.0209931	0.02	11.03
010040040010	7	0.0069977	0.01	11.04
010040040020	11	0.0109964	0.01	11.05
010040040030	4	0.0039987	0.00	11.05
010040040040	9	0.0089970	0.01	11.06
010040040050	1	0.0009997	0.00	11.06
010040040060	3	0.0029990	0.00	11.07
010040040070	8	0.0079974	0.01	11.08
010040050	2	0.0019993	0.00	11.08
010040120010	1	0.0009997	0.00	11.08
010040120020	1	0.0009997	0.00	11.08
010040120030	1	0.0009997	0.00	11.08
010040120040	1	0.0009997	0.00	11.08
010040120050	1	0.0009997	0.00	11.08
010040120060	1	0.0009997	0.00	11.08
010041	2	0.0019993	0.00	11.09
010050	163	0.1629462	0.16	11.25
010050010	61	0.0609799	0.06	11.31
010050020	86	0.0859716	0.09	11.40
010050020020	2	0.0019993	0.00	11.40
010050030	43	0.0429858	0.04	11.44
010050030010	2	0.0019993	0.00	11.44
010050030020	1	0.0009997	0.00	11.44
010050030030	2	0.0019993	0.00	11.45
010050040	2	0.0019993	0.00	11.45
010060	181	0.1809403	0.18	11.63

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
010060010	48	0.0479842	0.05	11.68
010060020	73	0.0729759	0.07	11.75
010060020010	11	0.0109964	0.01	11.76
010060020020	10	0.0099967	0.01	11.77
010060020030	5	0.0049984	0.00	11.78
010060020040	14	0.0139954	0.01	11.79
010060020050	1	0.0009997	0.00	11.79
010060030	57	0.0569812	0.06	11.85
010060030810	7	0.0069977	0.01	11.85
010060030820	8	0.0079974	0.01	11.86
010060030830	3	0.0029990	0.00	11.87
010060030840	1	0.0009997	0.00	11.87
010060040	14	0.0139954	0.01	11.88
010060050	33	0.0329891	0.03	11.91
010060051	4	0.0039987	0.00	11.92
010060060	23	0.0229924	0.02	11.94
010060070	5	0.0049984	0.00	11.95
010070	222	0.2219268	0.22	12.17
010070010	8	0.0079974	0.01	12.17
010070010010	11	0.0109964	0.01	12.19
010070010020	6	0.0059980	0.01	12.19
010070020	14	0.0139954	0.01	12.21
010070030	14	0.0139954	0.01	12.22
010080	131	0.1309568	0.13	12.35
010080010	9	0.0089970	0.01	12.36
010080010010	2	0.0019993	0.00	12.36
010080010020	11	0.0109964	0.01	12.37
010080020	17	0.0169944	0.02	12.39
010080030	8	0.0079974	0.01	12.40

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
010090	61	0.0609799	0.06	12.46
010090010	23	0.0229924	0.02	12.48
010090020	19	0.0189937	0.02	12.50
010090030	6	0.0059980	0.01	12.51
010091	5	0.0049984	0.00	12.51
010092	1	0.0009997	0.00	12.51
010100	25	0.0249918	0.02	12.54
010100010	2	0.0019993	0.00	12.54
010100020	1	0.0009997	0.00	12.54
010110	37	0.0369878	0.04	12.58
010110010	1	0.0009997	0.00	12.58
010110020	1	0.0009997	0.00	12.58
010110030	1	0.0009997	0.00	12.58
010120	4	0.0039987	0.00	12.58
010130	9	0.0089970	0.01	12.59
010140	5	0.0049984	0.00	12.60
011	11	0.0109964	0.01	12.61
012	2	0.0019993	0.00	12.61
014	2	0.0019993	0.00	12.61
020	6304	6.3019204	6.30	18.92
020000	1	0.0009997	0.00	18.92
020000010	2	0.0019993	0.00	18.92
020000020	8	0.0079974	0.01	18.93
020000030	8	0.0079974	0.01	18.93
020000040	8	0.0079974	0.01	18.94
020000050	8	0.0079974	0.01	18.95
020000060	8	0.0079974	0.01	18.96
020000500	1	0.0009997	0.00	18.96
020000510	1	0.0009997	0.00	18.96

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020000520	1	0.0009997	0.00	18.96
020000530	1	0.0009997	0.00	18.96
020000531	1	0.0009997	0.00	18.96
020010	1607	1.6064699	1.61	20.57
020010010	293	0.2929033	0.29	20.86
020010020	230	0.2299241	0.23	21.09
020010020010	25	0.0249918	0.02	21.12
020010020020	8	0.0079974	0.01	21.13
020010020020010	4	0.0039987	0.00	21.13
020010020020020	3	0.0029990	0.00	21.13
020010020020030	3	0.0029990	0.00	21.14
020010020020040	3	0.0029990	0.00	21.14
020010020020050	1	0.0009997	0.00	21.14
020010020030	12	0.0119960	0.01	21.15
020010020040	6	0.0059980	0.01	21.16
020010020050	5	0.0049984	0.00	21.16
020010020060	4	0.0039987	0.00	21.17
020010030	144	0.1439525	0.14	21.31
020010030010	11	0.0109964	0.01	21.32
020010030020	7	0.0069977	0.01	21.33
020010030030010	4	0.0039987	0.00	21.33
020010030030020010	5	0.0049984	0.00	21.34
020010030030020020	8	0.0079974	0.01	21.35
020010030040	4	0.0039987	0.00	21.35
020010040	91	0.0909700	0.09	21.44
020010040010	5	0.0049984	0.00	21.45
020010040020	3	0.0029990	0.00	21.45
020010040030	6	0.0059980	0.01	21.45
020010041	5	0.0049984	0.00	21.46

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020010050	92	0.0919697	0.09	21.55
020010060	32	0.0319894	0.03	21.58
020010070	38	0.0379875	0.04	21.62
020010070010	4	0.0039987	0.00	21.63
020010070020	7	0.0069977	0.01	21.63
020010070030	7	0.0069977	0.01	21.64
020010080	19	0.0189937	0.02	21.66
020010090	31	0.0309898	0.03	21.69
020010100	11	0.0109964	0.01	21.70
020010110	7	0.0069977	0.01	21.71
020010120	9	0.0089970	0.01	21.72
020011	3	0.0029990	0.00	21.72
020011042	1	0.0009997	0.00	21.72
020020	1524	1.5234972	1.52	23.24
020020010	147	0.1469515	0.15	23.39
020020010010	31	0.0309898	0.03	23.42
020020010010010	2	0.0019993	0.00	23.42
020020010010020	1	0.0009997	0.00	23.43
020020010010030	3	0.0029990	0.00	23.43
020020010010040	1	0.0009997	0.00	23.43
020020010010050	3	0.0029990	0.00	23.43
020020010020	11	0.0109964	0.01	23.44
020020010030	6	0.0059980	0.01	23.45
020020010040	10	0.0099967	0.01	23.46
020020010050	2	0.0019993	0.00	23.46
020020010060	6	0.0059980	0.01	23.47
020020010070	3	0.0029990	0.00	23.47
020020010080	2	0.0019993	0.00	23.47
020020020	221	0.2209271	0.22	23.69

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020020020010	33	0.0329891	0.03	23.73
020020020020	29	0.0289904	0.03	23.76
020020020030	25	0.0249918	0.02	23.78
020020020040	9	0.0089970	0.01	23.79
020020020050	19	0.0189937	0.02	23.81
020020020060	1	0.0009997	0.00	23.81
020020020070010	3	0.0029990	0.00	23.81
020020020070020	7	0.0069977	0.01	23.82
020020020070030	2	0.0019993	0.00	23.82
020020030	93	0.0929693	0.09	23.91
020020030010	19	0.0189937	0.02	23.93
020020030020	23	0.0229924	0.02	23.96
020020030030	19	0.0189937	0.02	23.98
020020030040	17	0.0169944	0.02	23.99
020020030050	5	0.0049984	0.00	24.00
020020030060	2	0.0019993	0.00	24.00
020020030070	16	0.0159947	0.02	24.02
020020030080	4	0.0039987	0.00	24.02
020020030090	3	0.0029990	0.00	24.02
020020030100	1	0.0009997	0.00	24.02
020020040	64	0.0639789	0.06	24.09
020020043	1	0.0009997	0.00	24.09
020020050	19	0.0189937	0.02	24.11
020020050010	24	0.0239921	0.02	24.13
020020050020	9	0.0089970	0.01	24.14
020020060	44	0.0439855	0.04	24.18
020020060010	1	0.0009997	0.00	24.19
020020060020	6	0.0059980	0.01	24.19
020020060030	5	0.0049984	0.00	24.20

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020020070	12	0.0119960	0.01	24.21
020020080	2	0.0019993	0.00	24.21
020020080010	6	0.0059980	0.01	24.22
020020080020	11	0.0109964	0.01	24.23
020020080030	4	0.0039987	0.00	24.23
020020080040	2	0.0019993	0.00	24.23
020020090	4	0.0039987	0.00	24.24
020020100	13	0.0129957	0.01	24.25
020021	23	0.0229924	0.02	24.27
020030	1087	1.0866414	1.09	25.36
020030010	148	0.1479512	0.15	25.51
020030010010	11	0.0109964	0.01	25.52
020030010020	5	0.0049984	0.00	25.52
020030010020010	2	0.0019993	0.00	25.53
020030010020020	7	0.0069977	0.01	25.53
020030010020030	2	0.0019993	0.00	25.53
020030010020040	1	0.0009997	0.00	25.54
020030010030	8	0.0079974	0.01	25.54
020030010040	6	0.0059980	0.01	25.55
020030020	222	0.2219268	0.22	25.77
020030020010	19	0.0189937	0.02	25.79
020030020020	2	0.0019993	0.00	25.79
020030030	106	0.1059650	0.11	25.90
020030040	48	0.0479842	0.05	25.95
020030040010	6	0.0059980	0.01	25.95
020030040020	2	0.0019993	0.00	25.95
020030040030	5	0.0049984	0.00	25.96
020030040040	1	0.0009997	0.00	25.96
020030050	39	0.0389871	0.04	26.00

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020030050010	9	0.0089970	0.01	26.01
020030050020	16	0.0159947	0.02	26.02
020030060	10	0.0099967	0.01	26.03
020030070	7	0.0069977	0.01	26.04
020030080	9	0.0089970	0.01	26.05
020030090	3	0.0029990	0.00	26.05
020030100	1	0.0009997	0.00	26.05
020030100010	1	0.0009997	0.00	26.06
020030100020	11	0.0109964	0.01	26.07
020030100030	1	0.0009997	0.00	26.07
020030110	3	0.0029990	0.00	26.07
020030120	2	0.0019993	0.00	26.07
020030130	7	0.0069977	0.01	26.08
020030140	2	0.0019993	0.00	26.08
020030150	8	0.0079974	0.01	26.09
020031	18	0.0179941	0.02	26.11
020040	735	0.7347575	0.73	26.84
020040010	66	0.0659782	0.07	26.91
020040020	64	0.0639789	0.06	26.97
020040030	38	0.0379875	0.04	27.01
020040030010	1	0.0009997	0.00	27.01
020040030020	6	0.0059980	0.01	27.02
020040040	22	0.0219927	0.02	27.04
020040050	5	0.0049984	0.00	27.04
020040060	8	0.0079974	0.01	27.05
020040070	1	0.0009997	0.00	27.05
020040160	1	0.0009997	0.00	27.05
020041	6	0.0059980	0.01	27.06
020050	428	0.4278588	0.43	27.49

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020050010	40	0.0399868	0.04	27.53
020050010010	5	0.0049984	0.00	27.53
020050010020	4	0.0039987	0.00	27.54
020050010030	3	0.0029990	0.00	27.54
020050020	39	0.0389871	0.04	27.58
020050030	5	0.0049984	0.00	27.58
020050030010	7	0.0069977	0.01	27.59
020050030020	14	0.0139954	0.01	27.60
020050030030	6	0.0059980	0.01	27.61
020050030040	2	0.0019993	0.00	27.61
020050030050	12	0.0119960	0.01	27.62
020050040	7	0.0069977	0.01	27.63
020050050	1	0.0009997	0.00	27.63
020050180	1	0.0009997	0.00	27.63
020050190	1	0.0009997	0.00	27.63
020050200	1	0.0009997	0.00	27.64
020050210	1	0.0009997	0.00	27.64
020051	7	0.0069977	0.01	27.64
020051211	1	0.0009997	0.00	27.64
020051212	1	0.0009997	0.00	27.65
020051213	1	0.0009997	0.00	27.65
020052	1	0.0009997	0.00	27.65
020053	1	0.0009997	0.00	27.65
020060	297	0.2969020	0.30	27.95
020060010	31	0.0309898	0.03	27.98
020060020	18	0.0179941	0.02	27.99
020060020010	11	0.0109964	0.01	28.01
020060020020010	2	0.0019993	0.00	28.01
020060020020020	2	0.0019993	0.00	28.01

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020060020020030	1	0.0009997	0.00	28.01
020060020020040	1	0.0009997	0.00	28.01
020060020020050	1	0.0009997	0.00	28.01
020060030	10	0.0099967	0.01	28.02
020060030010	2	0.0019993	0.00	28.02
020060030020	2	0.0019993	0.00	28.03
020060030030	12	0.0119960	0.01	28.04
020060030040	6	0.0059980	0.01	28.04
020060030050	1	0.0009997	0.00	28.05
020060040	9	0.0089970	0.01	28.05
020060050	4	0.0039987	0.00	28.06
020060050010	2	0.0019993	0.00	28.06
020060050020	14	0.0139954	0.01	28.07
020060060	6	0.0059980	0.01	28.08
020060070	9	0.0089970	0.01	28.09
020060080010	5	0.0049984	0.00	28.09
020060080020	14	0.0139954	0.01	28.11
020060090	11	0.0109964	0.01	28.12
020060100	1	0.0009997	0.00	28.12
020060110	3	0.0029990	0.00	28.12
020060220	1	0.0009997	0.00	28.12
020060230	2	0.0019993	0.00	28.13
020061	3	0.0029990	0.00	28.13
020070	184	0.1839393	0.18	28.31
020070010	17	0.0169944	0.02	28.33
020070020	5	0.0049984	0.00	28.34
020070020010	10	0.0099967	0.01	28.35
020070020020	23	0.0229924	0.02	28.37
020070020020010	1	0.0009997	0.00	28.37

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020070020020020	1	0.0009997	0.00	28.37
020070020030010	15	0.0149951	0.01	28.39
020070020030020010	3	0.0029990	0.00	28.39
020070020030020020	6	0.0059980	0.01	28.39
020070020030020030	1	0.0009997	0.00	28.40
020070020030020040	2	0.0019993	0.00	28.40
020070020030020050	1	0.0009997	0.00	28.40
020070020030030	1	0.0009997	0.00	28.40
020070020040	1	0.0009997	0.00	28.40
020070020050	2	0.0019993	0.00	28.40
020070020060	5	0.0049984	0.00	28.41
020070020069	1	0.0009997	0.00	28.41
020070020070	3	0.0029990	0.00	28.41
020070020080	1	0.0009997	0.00	28.41
020070020090	1	0.0009997	0.00	28.41
020070030	4	0.0039987	0.00	28.42
020070030010	10	0.0099967	0.01	28.43
020070030020	6	0.0059980	0.01	28.43
020070030030	14	0.0139954	0.01	28.45
020070030040	3	0.0029990	0.00	28.45
020070030050	1	0.0009997	0.00	28.45
020070040	5	0.0049984	0.00	28.46
020070240	1	0.0009997	0.00	28.46
020080	136	0.1359551	0.14	28.59
020080010	5	0.0049984	0.00	28.60
020080010010	23	0.0229924	0.02	28.62
020080010020010	2	0.0019993	0.00	28.62
020080010020020	7	0.0069977	0.01	28.63
020080010020030	11	0.0109964	0.01	28.64

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020080010030	6	0.0059980	0.01	28.65
020080010040	1	0.0009997	0.00	28.65
020080020	18	0.0179941	0.02	28.67
020080030	3	0.0029990	0.00	28.67
020080030010010	3	0.0029990	0.00	28.67
020080030010020	24	0.0239921	0.02	28.70
020080030020	3	0.0029990	0.00	28.70
020080030030	1	0.0009997	0.00	28.70
020080030040	1	0.0009997	0.00	28.70
020080030050	2	0.0019993	0.00	28.70
020080030060	2	0.0019993	0.00	28.71
020080040	21	0.0209931	0.02	28.73
020080050	1	0.0009997	0.00	28.73
020080050010	19	0.0189937	0.02	28.75
020080050020010	7	0.0069977	0.01	28.75
020080050020011	6	0.0059980	0.01	28.76
020080050020020010	4	0.0039987	0.00	28.76
020080050020020011	4	0.0039987	0.00	28.77
020080050020020020	16	0.0159947	0.02	28.78
020080050020030	5	0.0049984	0.00	28.79
020080050020040	14	0.0139954	0.01	28.80
020080050020050010	5	0.0049984	0.00	28.81
020080050020050020	9	0.0089970	0.01	28.82
020080050020050030	2	0.0019993	0.00	28.82
020080050020050040	3	0.0029990	0.00	28.82
020080050020060	2	0.0019993	0.00	28.82
020080050030	5	0.0049984	0.00	28.83
020080050040	8	0.0079974	0.01	28.84
020080050050	31	0.0309898	0.03	28.87

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020080060	10	0.0099967	0.01	28.88
020080070	7	0.0069977	0.01	28.88
020080080	3	0.0029990	0.00	28.89
020080080010	24	0.0239921	0.02	28.91
020080080020	8	0.0079974	0.01	28.92
020080090010010	25	0.0249918	0.02	28.94
020080090010020	8	0.0079974	0.01	28.95
020080090010030	6	0.0059980	0.01	28.96
020080090010040	1	0.0009997	0.00	28.96
020080090020	5	0.0049984	0.00	28.96
020080090030	2	0.0019993	0.00	28.97
020080090040	25	0.0249918	0.02	28.99
020080100010	4	0.0039987	0.00	29.00
020080100020	5	0.0049984	0.00	29.00
020080100030	1	0.0009997	0.00	29.00
020080100040	4	0.0039987	0.00	29.01
020080110	6	0.0059980	0.01	29.01
020080120010	13	0.0129957	0.01	29.02
020080120020	3	0.0029990	0.00	29.03
020080120030010	3	0.0029990	0.00	29.03
020080120030020	4	0.0039987	0.00	29.03
020080120030030010	4	0.0039987	0.00	29.04
020080120030030020	7	0.0069977	0.01	29.05
020080120030030030	10	0.0099967	0.01	29.06
020080130	11	0.0109964	0.01	29.07
020080140	13	0.0129957	0.01	29.08
020080150	5	0.0049984	0.00	29.08
020080160	36	0.0359881	0.04	29.12
020080170	18	0.0179941	0.02	29.14

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020080180	3	0.0029990	0.00	29.14
020080190	3	0.0029990	0.00	29.14
020080200	14	0.0139954	0.01	29.16
020080210	1	0.0009997	0.00	29.16
020080220	3	0.0029990	0.00	29.16
020080230	10	0.0099967	0.01	29.17
020080240	16	0.0159947	0.02	29.19
020080250	4	0.0039987	0.00	29.19
020080260	11	0.0109964	0.01	29.20
020080270	32	0.0319894	0.03	29.24
020090	106	0.1059650	0.11	29.34
020090010	15	0.0149951	0.01	29.36
020090020	3	0.0029990	0.00	29.36
020090030	12	0.0119960	0.01	29.37
020100	80	0.0799736	0.08	29.45
020100010	8	0.0079974	0.01	29.46
020100020	7	0.0069977	0.01	29.47
020100030	3	0.0029990	0.00	29.47
020100040	2	0.0019993	0.00	29.47
020110	105	0.1049654	0.10	29.58
020110010	1	0.0009997	0.00	29.58
020110020	2	0.0019993	0.00	29.58
020110030	2	0.0019993	0.00	29.58
020110040	2	0.0019993	0.00	29.58
020110050	3	0.0029990	0.00	29.59
020120	51	0.0509832	0.05	29.64
020120010	1	0.0009997	0.00	29.64
020120020	3	0.0029990	0.00	29.64
020120030	3	0.0029990	0.00	29.64

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020130	72	0.0719762	0.07	29.72
020130010	1	0.0009997	0.00	29.72
020130020	1	0.0009997	0.00	29.72
020130030	7	0.0069977	0.01	29.73
020130040	4	0.0039987	0.00	29.73
020140	94	0.0939690	0.09	29.82
020150	42	0.0419861	0.04	29.87
020160	53	0.0529825	0.05	29.92
020170	39	0.0389871	0.04	29.96
020180	48	0.0479842	0.05	30.01
020190	40	0.0399868	0.04	30.05
020200	37	0.0369878	0.04	30.08
020210	37	0.0369878	0.04	30.12
020220	42	0.0419861	0.04	30.16
020230	46	0.0459848	0.05	30.21
020240	34	0.0339888	0.03	30.24
020250	43	0.0429858	0.04	30.28
020260	35	0.0349885	0.03	30.32
020270	36	0.0359881	0.04	30.35
020280	37	0.0369878	0.04	30.39
020290	13	0.0129957	0.01	30.40
020300	25	0.0249918	0.02	30.43
021	32	0.0319894	0.03	30.46
021010	5	0.0049984	0.00	30.47
021020	2	0.0019993	0.00	30.47
021030	4	0.0039987	0.00	30.47
022	3	0.0029990	0.00	30.48
030	5341	5.3392381	5.34	35.82
030000550	1	0.0009997	0.00	35.82

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030000560	1	0.0009997	0.00	35.82
030000590	1	0.0009997	0.00	35.82
030000600	1	0.0009997	0.00	35.82
030000610	1	0.0009997	0.00	35.82
030000620	1	0.0009997	0.00	35.82
030000630	1	0.0009997	0.00	35.82
030000640	1	0.0009997	0.00	35.82
030000660	1	0.0009997	0.00	35.82
030000670	1	0.0009997	0.00	35.83
030000680	1	0.0009997	0.00	35.83
030000690	1	0.0009997	0.00	35.83
030000730	1	0.0009997	0.00	35.83
030000740	1	0.0009997	0.00	35.83
030000760	1	0.0009997	0.00	35.83
030000770	1	0.0009997	0.00	35.83
030000780	1	0.0009997	0.00	35.83
030000790	1	0.0009997	0.00	35.83
030000800	1	0.0009997	0.00	35.83
030000810	1	0.0009997	0.00	35.84
030000820	1	0.0009997	0.00	35.84
030000830	1	0.0009997	0.00	35.84
030000840	1	0.0009997	0.00	35.84
030000850	1	0.0009997	0.00	35.84
030000860	1	0.0009997	0.00	35.84
030000890	1	0.0009997	0.00	35.84
030000900	1	0.0009997	0.00	35.84
030000910	1	0.0009997	0.00	35.84
030000930	1	0.0009997	0.00	35.84
030000940	1	0.0009997	0.00	35.85

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030000960	1	0.0009997	0.00	35.85
030000970	1	0.0009997	0.00	35.85
030000980	1	0.0009997	0.00	35.85
030000990	1	0.0009997	0.00	35.85
030001000	1	0.0009997	0.00	35.85
030001010	1	0.0009997	0.00	35.85
030001020	1	0.0009997	0.00	35.85
030001040	1	0.0009997	0.00	35.85
030001050	1	0.0009997	0.00	35.85
030001060	1	0.0009997	0.00	35.86
030001070	1	0.0009997	0.00	35.86
030001080	1	0.0009997	0.00	35.86
030001090	1	0.0009997	0.00	35.86
030001100	1	0.0009997	0.00	35.86
030001110	1	0.0009997	0.00	35.86
030001120	1	0.0009997	0.00	35.86
030001130	1	0.0009997	0.00	35.86
030001140	1	0.0009997	0.00	35.86
030001150	1	0.0009997	0.00	35.86
030001160	1	0.0009997	0.00	35.87
030001170	1	0.0009997	0.00	35.87
030001180	1	0.0009997	0.00	35.87
030001190	1	0.0009997	0.00	35.87
030001200	1	0.0009997	0.00	35.87
030001210	1	0.0009997	0.00	35.87
030001220	1	0.0009997	0.00	35.87
030001230	1	0.0009997	0.00	35.87
030001240	1	0.0009997	0.00	35.87
030001250	1	0.0009997	0.00	35.87

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030001260	1	0.0009997	0.00	35.88
030001270	1	0.0009997	0.00	35.88
030001280	1	0.0009997	0.00	35.88
030001290	1	0.0009997	0.00	35.88
030001300	1	0.0009997	0.00	35.88
030001310	1	0.0009997	0.00	35.88
030001320	1	0.0009997	0.00	35.88
030001330	1	0.0009997	0.00	35.88
030001370	1	0.0009997	0.00	35.88
030001380	1	0.0009997	0.00	35.88
030001390	1	0.0009997	0.00	35.89
030001400	1	0.0009997	0.00	35.89
030001430	1	0.0009997	0.00	35.89
030001440	1	0.0009997	0.00	35.89
030001470	1	0.0009997	0.00	35.89
030001520	1	0.0009997	0.00	35.89
030010	1094	1.0936391	1.09	36.98
030010010	120	0.1199604	0.12	37.10
030010020	117	0.1169614	0.12	37.22
030010020010	4	0.0039987	0.00	37.22
030010020020	6	0.0059980	0.01	37.23
030010020030	6	0.0059980	0.01	37.24
030010030	73	0.0729759	0.07	37.31
030010030010	17	0.0169944	0.02	37.33
030010030020	8	0.0079974	0.01	37.33
030010030030	4	0.0039987	0.00	37.34
030010030040	6	0.0059980	0.01	37.34
030010030050	1	0.0009997	0.00	37.35
030010031	5	0.0049984	0.00	37.35

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030010040	35	0.0349885	0.03	37.39
030010040010	16	0.0159947	0.02	37.40
030010040020	5	0.0049984	0.00	37.41
030010050	19	0.0189937	0.02	37.43
030010060	11	0.0109964	0.01	37.44
030010070	3	0.0029990	0.00	37.44
030010080	3	0.0029990	0.00	37.44
030010150	3	0.0029990	0.00	37.45
030020	1023	1.0226625	1.02	38.47
030020010	96	0.0959683	0.10	38.56
030020010010	11	0.0109964	0.01	38.58
030020010020	13	0.0129957	0.01	38.59
030020010030	6	0.0059980	0.01	38.59
030020010050	4	0.0039987	0.00	38.60
030020010060	7	0.0069977	0.01	38.61
030020020	118	0.1179611	0.12	38.72
030020020010	4	0.0039987	0.00	38.73
030020020020	2	0.0019993	0.00	38.73
030020020030	3	0.0029990	0.00	38.73
030020020040	7	0.0069977	0.01	38.74
030020020050	3	0.0029990	0.00	38.74
030020020060	1	0.0009997	0.00	38.74
030020020070	2	0.0019993	0.00	38.75
030020020080	2	0.0019993	0.00	38.75
030020020090	5	0.0049984	0.00	38.75
030020020100	1	0.0009997	0.00	38.75
030020030	83	0.0829726	0.08	38.84
030020040	30	0.0299901	0.03	38.87
030020050	21	0.0209931	0.02	38.89

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030020060	10	0.0099967	0.01	38.90
030020060010	6	0.0059980	0.01	38.90
030020060020	4	0.0039987	0.00	38.91
030020070	4	0.0039987	0.00	38.91
030020070010	1	0.0009997	0.00	38.91
030020070020	5	0.0049984	0.00	38.92
030020070030	3	0.0029990	0.00	38.92
030020080	9	0.0089970	0.01	38.93
030020090	6	0.0059980	0.01	38.94
030020170	2	0.0019993	0.00	38.94
030020180	2	0.0019993	0.00	38.94
030020190	12	0.0119960	0.01	38.95
030021	2	0.0019993	0.00	38.95
030030	600	0.5998021	0.60	39.55
030030010	88	0.0879710	0.09	39.64
030030020	82	0.0819729	0.08	39.72
030030030	38	0.0379875	0.04	39.76
030030030010	10	0.0099967	0.01	39.77
030030030020	9	0.0089970	0.01	39.78
030030030030	7	0.0069977	0.01	39.79
030030030040	5	0.0049984	0.00	39.79
030030030050	7	0.0069977	0.01	39.80
030030030060	20	0.0199934	0.02	39.82
030030040	38	0.0379875	0.04	39.86
030030050	17	0.0169944	0.02	39.87
030030060	4	0.0039987	0.00	39.88
030030070	1	0.0009997	0.00	39.88
030030270	2	0.0019993	0.00	39.88
030030280	2	0.0019993	0.00	39.88

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030030290	5	0.0049984	0.00	39.89
030030330	4	0.0039987	0.00	39.89
030031	11	0.0109964	0.01	39.90
030032	8	0.0079974	0.01	39.91
030040	362	0.3618806	0.36	40.27
030040010	32	0.0319894	0.03	40.30
030040010010	1	0.0009997	0.00	40.31
030040010020	4	0.0039987	0.00	40.31
030040010030	3	0.0029990	0.00	40.31
030040010040	1	0.0009997	0.00	40.31
030040010050	2	0.0019993	0.00	40.32
030040010060	1	0.0009997	0.00	40.32
030040010070	1	0.0009997	0.00	40.32
030040010080	3	0.0029990	0.00	40.32
030040020	26	0.0259914	0.03	40.35
030040020010	44	0.0439855	0.04	40.39
030040020020	18	0.0179941	0.02	40.41
030040020030	8	0.0079974	0.01	40.42
030040020040	4	0.0039987	0.00	40.42
030040020050	2	0.0019993	0.00	40.42
030040020060	1	0.0009997	0.00	40.42
030040020070	1	0.0009997	0.00	40.42
030040020080	4	0.0039987	0.00	40.43
030040030	6	0.0059980	0.01	40.43
030040030010	2	0.0019993	0.00	40.44
030040030020	5	0.0049984	0.00	40.44
030040030030	2	0.0019993	0.00	40.44
030040040	12	0.0119960	0.01	40.46
030040100	2	0.0019993	0.00	40.46

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030041	2	0.0019993	0.00	40.46
030050	186	0.1859386	0.19	40.65
030050010	61	0.0609799	0.06	40.71
030050020	44	0.0439855	0.04	40.75
030050030	13	0.0129957	0.01	40.76
030050040	11	0.0109964	0.01	40.77
030050050	3	0.0029990	0.00	40.78
030060	137	0.1369548	0.14	40.91
030060010010	1	0.0009997	0.00	40.92
030060010020	3	0.0029990	0.00	40.92
030060010030	4	0.0039987	0.00	40.92
030060010040	1	0.0009997	0.00	40.92
030060010050	1	0.0009997	0.00	40.92
030060010060	1	0.0009997	0.00	40.93
030060020	3	0.0029990	0.00	40.93
030070	75	0.0749753	0.07	41.00
030070010	83	0.0829726	0.08	41.09
030070010010	4	0.0039987	0.00	41.09
030070010020	3	0.0029990	0.00	41.09
030070020	11	0.0109964	0.01	41.10
030070020010	8	0.0079974	0.01	41.11
030070020020	5	0.0049984	0.00	41.12
030070030	3	0.0029990	0.00	41.12
030070030010	2	0.0019993	0.00	41.12
030070030020	8	0.0079974	0.01	41.13
030070030030	2	0.0019993	0.00	41.13
030070030040	1	0.0009997	0.00	41.13
030070040	6	0.0059980	0.01	41.14
030080	38	0.0379875	0.04	41.18

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030080010	56	0.0559815	0.06	41.23
030080020	24	0.0239921	0.02	41.26
030081	2	0.0019993	0.00	41.26
030090	57	0.0569812	0.06	41.32
030100	48	0.0479842	0.05	41.36
030100010	6	0.0059980	0.01	41.37
030100020	4	0.0039987	0.00	41.37
030100030	4	0.0039987	0.00	41.38
030110	29	0.0289904	0.03	41.41
030120	36	0.0359881	0.04	41.44
030130	31	0.0309898	0.03	41.47
030130010	13	0.0129957	0.01	41.49
030130020	12	0.0119960	0.01	41.50
030130030	2	0.0019993	0.00	41.50
030130040	2	0.0019993	0.00	41.50
030140	26	0.0259914	0.03	41.53
030150	48	0.0479842	0.05	41.58
030160	1	0.0009997	0.00	41.58
030170	4	0.0039987	0.00	41.58
030220	1	0.0009997	0.00	41.58
030260	1	0.0009997	0.00	41.58
030280	14	0.0139954	0.01	41.60
030970	57	0.0569812	0.06	41.66
030971	5	0.0049984	0.00	41.66
030980	1	0.0009997	0.00	41.66
030990	1	0.0009997	0.00	41.66
030991	1	0.0009997	0.00	41.66
030992	1	0.0009997	0.00	41.66
030993	1	0.0009997	0.00	41.67

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
031	9	0.0089970	0.01	41.67
040	4230	4.2286046	4.23	45.90
040001570	1	0.0009997	0.00	45.90
040001630	1	0.0009997	0.00	45.90
040001640	1	0.0009997	0.00	45.91
040001650	1	0.0009997	0.00	45.91
040001660	1	0.0009997	0.00	45.91
040001670	1	0.0009997	0.00	45.91
040001680	1	0.0009997	0.00	45.91
040001690	1	0.0009997	0.00	45.91
040001700	1	0.0009997	0.00	45.91
040001710	1	0.0009997	0.00	45.91
040001720	1	0.0009997	0.00	45.91
040001730	1	0.0009997	0.00	45.91
040001740	1	0.0009997	0.00	45.92
040001750	1	0.0009997	0.00	45.92
040001760	1	0.0009997	0.00	45.92
040001770	1	0.0009997	0.00	45.92
040001780	1	0.0009997	0.00	45.92
040001790	1	0.0009997	0.00	45.92
040001800	1	0.0009997	0.00	45.92
040001810	1	0.0009997	0.00	45.92
040001820	1	0.0009997	0.00	45.92
040001830	1	0.0009997	0.00	45.92
040001840	1	0.0009997	0.00	45.93
040001850	1	0.0009997	0.00	45.93
040001860	1	0.0009997	0.00	45.93
040001870	1	0.0009997	0.00	45.93
040001880	1	0.0009997	0.00	45.93

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
040001890	1	0.0009997	0.00	45.93
040001900	1	0.0009997	0.00	45.93
040001910	1	0.0009997	0.00	45.93
040001920	1	0.0009997	0.00	45.93
040001930	1	0.0009997	0.00	45.93
040001940	1	0.0009997	0.00	45.94
040001950	1	0.0009997	0.00	45.94
040001960	1	0.0009997	0.00	45.94
040001970	1	0.0009997	0.00	45.94
040001980	1	0.0009997	0.00	45.94
040001990	1	0.0009997	0.00	45.94
040002000	1	0.0009997	0.00	45.94
040002010	1	0.0009997	0.00	45.94
040002020	1	0.0009997	0.00	45.94
040002030	1	0.0009997	0.00	45.94
040002070	1	0.0009997	0.00	45.95
040002080	1	0.0009997	0.00	45.95
040002090	1	0.0009997	0.00	45.95
040002100	1	0.0009997	0.00	45.95
040002120	1	0.0009997	0.00	45.95
040002130	1	0.0009997	0.00	45.95
040002140	1	0.0009997	0.00	45.95
040002150	1	0.0009997	0.00	45.95
040002170	1	0.0009997	0.00	45.95
040002180	1	0.0009997	0.00	45.95
040010	477	0.4768426	0.48	46.43
040010010	57	0.0569812	0.06	46.49
040010020	76	0.0759749	0.08	46.56
040010020010	7	0.0069977	0.01	46.57

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
040010020020	20	0.0199934	0.02	46.59
040010020030	1	0.0009997	0.00	46.59
040010020040	3	0.0029990	0.00	46.60
040010030	30	0.0299901	0.03	46.63
040010030010	8	0.0079974	0.01	46.63
040010030020	4	0.0039987	0.00	46.64
040010030030	7	0.0069977	0.01	46.64
040010040	19	0.0189937	0.02	46.66
040010050	27	0.0269911	0.03	46.69
040010060	1	0.0009997	0.00	46.69
040010060010	28	0.0279908	0.03	46.72
040010060020010	6	0.0059980	0.01	46.73
040010060020020	1	0.0009997	0.00	46.73
040010060020030010	7	0.0069977	0.01	46.73
040010060020030020	28	0.0279908	0.03	46.76
040010060020030030	20	0.0199934	0.02	46.78
040010060020030040	12	0.0119960	0.01	46.79
040010060020030050	36	0.0359881	0.04	46.83
040010060020040	1	0.0009997	0.00	46.83
040010060030	6	0.0059980	0.01	46.84
040010070	1	0.0009997	0.00	46.84
040010070010	5	0.0049984	0.00	46.84
040010070011	2	0.0019993	0.00	46.84
040010070020010	1	0.0009997	0.00	46.85
040010070020020	13	0.0129957	0.01	46.86
040010070020030	4	0.0039987	0.00	46.86
040010070020040	4	0.0039987	0.00	46.87
040010070020050	3	0.0029990	0.00	46.87
040010070030	1	0.0009997	0.00	46.87

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
040010080	3	0.0029990	0.00	46.87
040020	419	0.4188618	0.42	47.29
040020010	139	0.1389541	0.14	47.43
040020010010	5	0.0049984	0.00	47.44
040020010020	13	0.0129957	0.01	47.45
040020010030	8	0.0079974	0.01	47.46
040020010040	11	0.0109964	0.01	47.47
040020010050	6	0.0059980	0.01	47.47
040020010060	7	0.0069977	0.01	47.48
040020010070	4	0.0039987	0.00	47.49
040020010080	1	0.0009997	0.00	47.49
040020020	112	0.1119631	0.11	47.60
040020020010	29	0.0289904	0.03	47.63
040020020020	8	0.0079974	0.01	47.64
040020020030	12	0.0119960	0.01	47.65
040020020040	15	0.0149951	0.01	47.66
040020020050	7	0.0069977	0.01	47.67
040020030	62	0.0619795	0.06	47.73
040020030010	28	0.0279908	0.03	47.76
040020030020	5	0.0049984	0.00	47.76
040020030030	3	0.0029990	0.00	47.77
040020030040	1	0.0009997	0.00	47.77
040020040	37	0.0369878	0.04	47.81
040020050	82	0.0819729	0.08	47.89
040020060	16	0.0159947	0.02	47.90
040020070	12	0.0119960	0.01	47.92
040020070010	27	0.0269911	0.03	47.94
040020070020	6	0.0059980	0.01	47.95
040020080	7	0.0069977	0.01	47.96

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
040020090	5	0.0049984	0.00	47.96
040030	380	0.3798746	0.38	48.34
040030010	28	0.0279908	0.03	48.37
040030010010	38	0.0379875	0.04	48.41
040030010020	52	0.0519828	0.05	48.46
040030010030	21	0.0209931	0.02	48.48
040030010040	2	0.0019993	0.00	48.48
040030010050	7	0.0069977	0.01	48.49
040030010060	14	0.0139954	0.01	48.50
040030020	61	0.0609799	0.06	48.56
040030020010	9	0.0089970	0.01	48.57
040030020020	2	0.0019993	0.00	48.57
040030030	49	0.0489838	0.05	48.62
040030040	11	0.0109964	0.01	48.63
040030040010	1	0.0009997	0.00	48.63
040030040020	1	0.0009997	0.00	48.64
040030050	2	0.0019993	0.00	48.64
040040	182	0.1819400	0.18	48.82
040040010	71	0.0709766	0.07	48.89
040040020	14	0.0139954	0.01	48.90
040040030	5	0.0049984	0.00	48.91
040050	75	0.0749753	0.07	48.98
040050010	5	0.0049984	0.00	48.99
040050020	3	0.0029990	0.00	48.99
040050030	7	0.0069977	0.01	49.00
040060	43	0.0429858	0.04	49.04
040060010	6	0.0059980	0.01	49.05
040060020	4	0.0039987	0.00	49.05
040060050	5	0.0049984	0.00	49.06

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
040070	10	0.0099967	0.01	49.07
040080	18	0.0179941	0.02	49.09
040090	34	0.0339888	0.03	49.12
040100	16	0.0159947	0.02	49.14
040110	5	0.0049984	0.00	49.14
040120	6	0.0059980	0.01	49.15
040130	5	0.0049984	0.00	49.15
040140	2	0.0019993	0.00	49.15
040160	14	0.0139954	0.01	49.17
040170	6	0.0059980	0.01	49.17
040190	1	0.0009997	0.00	49.17
040200	1	0.0009997	0.00	49.18
040390	1	0.0009997	0.00	49.18
040400	8	0.0079974	0.01	49.18
040410	1	0.0009997	0.00	49.19
040420	1	0.0009997	0.00	49.19
040430	1	0.0009997	0.00	49.19
040440	1	0.0009997	0.00	49.19
040450	1	0.0009997	0.00	49.19
040460	1	0.0009997	0.00	49.19
040470	1	0.0009997	0.00	49.19
040480	1	0.0009997	0.00	49.19
040500	1	0.0009997	0.00	49.19
040510	1	0.0009997	0.00	49.19
040520	1	0.0009997	0.00	49.20
041	31	0.0309898	0.03	49.23
041010	9	0.0089970	0.01	49.24
041020	5	0.0049984	0.00	49.24
041030	2	0.0019993	0.00	49.24

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
041040	10	0.0099967	0.01	49.25
041451	3	0.0029990	0.00	49.26
041452	3	0.0029990	0.00	49.26
041454	14	0.0139954	0.01	49.27
042	2	0.0019993	0.00	49.27
042010	1	0.0009997	0.00	49.28
042020	9	0.0089970	0.01	49.28
042030	1	0.0009997	0.00	49.29
042040	11	0.0109964	0.01	49.30
050	3110	3.1089740	3.11	52.41
050002190	1	0.0009997	0.00	52.41
050002200	1	0.0009997	0.00	52.41
050002210	1	0.0009997	0.00	52.41
050002220	1	0.0009997	0.00	52.41
050002230	1	0.0009997	0.00	52.41
050002231	1	0.0009997	0.00	52.41
050002240	3	0.0029990	0.00	52.41
050002250	1	0.0009997	0.00	52.42
050002260	1	0.0009997	0.00	52.42
050002270	1	0.0009997	0.00	52.42
050002280	1	0.0009997	0.00	52.42
050002290	55	0.0549819	0.05	52.47
050010	327	0.3268921	0.33	52.80
050010010	34	0.0339888	0.03	52.83
050010020	39	0.0389871	0.04	52.87
050010030	16	0.0159947	0.02	52.89
050010040	42	0.0419861	0.04	52.93
050010050	34	0.0339888	0.03	52.97
050010050010	1	0.0009997	0.00	52.97

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
050010050020	1	0.0009997	0.00	52.97
050010050030	2	0.0019993	0.00	52.97
050010050040	2	0.0019993	0.00	52.97
050010060	5	0.0049984	0.00	52.98
050010070	3	0.0029990	0.00	52.98
050010080	4	0.0039987	0.00	52.98
050010090	5	0.0049984	0.00	52.99
050010100010	3	0.0029990	0.00	52.99
050010100020	3	0.0029990	0.00	52.99
050010100030	1	0.0009997	0.00	53.00
050010110010	1	0.0009997	0.00	53.00
050010110020	19	0.0189937	0.02	53.02
050010110030	13	0.0129957	0.01	53.03
050010110040	8	0.0079974	0.01	53.04
050010110050	10	0.0099967	0.01	53.05
050010110060	1	0.0009997	0.00	53.05
050010120	9	0.0089970	0.01	53.06
050011	5	0.0049984	0.00	53.06
050020	307	0.3068987	0.31	53.37
050020010	72	0.0719762	0.07	53.44
050020010010	12	0.0119960	0.01	53.45
050020010010010	17	0.0169944	0.02	53.47
050020010010020	15	0.0149951	0.01	53.48
050020010010030	14	0.0139954	0.01	53.50
050020010020	12	0.0119960	0.01	53.51
050020020	38	0.0379875	0.04	53.55
050020020010	5	0.0049984	0.00	53.55
050020020020	15	0.0149951	0.01	53.57
050020020030	31	0.0309898	0.03	53.60

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
050020020040	28	0.0279908	0.03	53.63
050020020050	4	0.0039987	0.00	53.63
050020030	54	0.0539822	0.05	53.69
050020040	53	0.0529825	0.05	53.74
050020040010	28	0.0279908	0.03	53.77
050020040020	16	0.0159947	0.02	53.78
050020040030	8	0.0079974	0.01	53.79
050020050	17	0.0169944	0.02	53.81
050020060	11	0.0109964	0.01	53.82
050020070	4	0.0039987	0.00	53.82
050020080	2	0.0019993	0.00	53.82
050020090	2	0.0019993	0.00	53.83
050020100	2	0.0019993	0.00	53.83
050020110	9	0.0089970	0.01	53.84
050030	206	0.2059320	0.21	54.04
050030010	47	0.0469845	0.05	54.09
050030020	49	0.0489838	0.05	54.14
050030030	54	0.0539822	0.05	54.19
050030040	48	0.0479842	0.05	54.24
050030050	5	0.0049984	0.00	54.25
050030060	32	0.0319894	0.03	54.28
050030070	23	0.0229924	0.02	54.30
050030080	9	0.0089970	0.01	54.31
050040	110	0.1099637	0.11	54.42
050040010	9	0.0089970	0.01	54.43
050040020	17	0.0169944	0.02	54.45
050040030	8	0.0079974	0.01	54.45
050050	83	0.0829726	0.08	54.54
050060	20	0.0199934	0.02	54.56

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
050070	23	0.0229924	0.02	54.58
050080	22	0.0219927	0.02	54.60
050090	16	0.0159947	0.02	54.62
050460	6	0.0059980	0.01	54.62
050660	1	0.0009997	0.00	54.62
051	44	0.0439855	0.04	54.67
052	7	0.0069977	0.01	54.68
053	3	0.0029990	0.00	54.68
060	2331	2.3302310	2.33	57.01
060002300	1	0.0009997	0.00	57.01
060002310	10	0.0099967	0.01	57.02
060002320	12	0.0119960	0.01	57.03
060002330010	13	0.0129957	0.01	57.05
060002330020	7	0.0069977	0.01	57.05
060002340	12	0.0119960	0.01	57.06
060002350	2	0.0019993	0.00	57.07
060002360	3	0.0029990	0.00	57.07
060002370	2	0.0019993	0.00	57.07
060010	194	0.1939360	0.19	57.27
060010010	9	0.0089970	0.01	57.27
060010020	23	0.0229924	0.02	57.30
060010030	11	0.0109964	0.01	57.31
060010040	12	0.0119960	0.01	57.32
060010050	5	0.0049984	0.00	57.33
060010060	3	0.0029990	0.00	57.33
060010070	6	0.0059980	0.01	57.33
060020	188	0.1879380	0.19	57.52
060020010	25	0.0249918	0.02	57.55
060020020	23	0.0229924	0.02	57.57

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
060020030	2	0.0019993	0.00	57.57
060020030010	4	0.0039987	0.00	57.58
060020030020	24	0.0239921	0.02	57.60
060020030030	13	0.0129957	0.01	57.61
060020040	2	0.0019993	0.00	57.61
060020050	3	0.0029990	0.00	57.62
060020060	1	0.0009997	0.00	57.62
060021	4	0.0039987	0.00	57.62
060030	119	0.1189607	0.12	57.74
060030010	23	0.0229924	0.02	57.76
060030020	20	0.0199934	0.02	57.78
060030030	6	0.0059980	0.01	57.79
060030040	2	0.0019993	0.00	57.79
060030050	12	0.0119960	0.01	57.80
060030060	1	0.0009997	0.00	57.81
060040	59	0.0589805	0.06	57.86
060040010	7	0.0069977	0.01	57.87
060040020	6	0.0059980	0.01	57.88
060050	3	0.0029990	0.00	57.88
060101	6	0.0059980	0.01	57.89
060102	5	0.0049984	0.00	57.89
060110	3	0.0029990	0.00	57.89
060670	1	0.0009997	0.00	57.90
061	17	0.0169944	0.02	57.91
070	1527	1.5264963	1.53	59.44
070002390	1	0.0009997	0.00	59.44
070002400	7	0.0069977	0.01	59.45
070002420	2	0.0019993	0.00	59.45
070002430	1	0.0009997	0.00	59.45

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
070002440	1	0.0009997	0.00	59.45
070002460	1	0.0009997	0.00	59.45
070002461	4	0.0039987	0.00	59.46
070002462	2	0.0019993	0.00	59.46
070002463	1	0.0009997	0.00	59.46
070002470	19	0.0189937	0.02	59.48
070002471	2	0.0019993	0.00	59.48
070002480010	2	0.0019993	0.00	59.48
070002480020	9	0.0089970	0.01	59.49
070002480030	6	0.0059980	0.01	59.50
070002480040	4	0.0039987	0.00	59.50
070002490	3	0.0029990	0.00	59.50
070002500	10	0.0099967	0.01	59.51
070002510	2	0.0019993	0.00	59.52
070002520	1	0.0009997	0.00	59.52
070002530	3	0.0029990	0.00	59.52
070010	196	0.1959353	0.20	59.72
070010010	10	0.0099967	0.01	59.73
070010020	10	0.0099967	0.01	59.74
070010030	7	0.0069977	0.01	59.74
070010040	7	0.0069977	0.01	59.75
070010050	5	0.0049984	0.00	59.76
070010060	1	0.0009997	0.00	59.76
070010070	1	0.0009997	0.00	59.76
070011	1	0.0009997	0.00	59.76
070020	174	0.1739426	0.17	59.93
070020010	13	0.0129957	0.01	59.95
070020020	20	0.0199934	0.02	59.97
070020020010	3	0.0029990	0.00	59.97

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
070020020020	6	0.0059980	0.01	59.97
070020020030	1	0.0009997	0.00	59.98
070020030	13	0.0129957	0.01	59.99
070020040	1	0.0009997	0.00	59.99
070030	86	0.0859716	0.09	60.08
070030010	7	0.0069977	0.01	60.08
070030020	3	0.0029990	0.00	60.09
070030020010	8	0.0079974	0.01	60.09
070030020020	2	0.0019993	0.00	60.10
070030030	1	0.0009997	0.00	60.10
070030040	1	0.0009997	0.00	60.10
070040	34	0.0339888	0.03	60.13
070050	29	0.0289904	0.03	60.16
070060	6	0.0059980	0.01	60.17
070070	2	0.0019993	0.00	60.17
070080	5	0.0049984	0.00	60.17
070090	6	0.0059980	0.01	60.18
070680	1	0.0009997	0.00	60.18
070690	1	0.0009997	0.00	60.18
071	3	0.0029990	0.00	60.18
080	1013	1.0126658	1.01	61.20
080010	234	0.2339228	0.23	61.43
080010010	8	0.0079974	0.01	61.44
080010020	24	0.0239921	0.02	61.46
080010030	14	0.0139954	0.01	61.48
080010040	7	0.0069977	0.01	61.48
080010050	6	0.0059980	0.01	61.49
080020	147	0.1469515	0.15	61.64
080020010	15	0.0149951	0.01	61.65

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
080020020	8	0.0079974	0.01	61.66
080020020010	6	0.0059980	0.01	61.67
080020020020	25	0.0249918	0.02	61.69
080020020030	55	0.0549819	0.05	61.75
080020030	14	0.0139954	0.01	61.76
080020040	12	0.0119960	0.01	61.77
080020050	8	0.0079974	0.01	61.78
080020060	8	0.0079974	0.01	61.79
080030	69	0.0689772	0.07	61.86
080030010	18	0.0179941	0.02	61.87
080030020	4	0.0039987	0.00	61.88
080030020010	3	0.0029990	0.00	61.88
080030020020	3	0.0029990	0.00	61.88
080030020030	2	0.0019993	0.00	61.89
080030020040	1	0.0009997	0.00	61.89
080030020050	1	0.0009997	0.00	61.89
080030030	4	0.0039987	0.00	61.89
080030040	1	0.0009997	0.00	61.89
080040	55	0.0549819	0.05	61.95
080050	52	0.0519828	0.05	62.00
080060	26	0.0259914	0.03	62.03
080060010	4	0.0039987	0.00	62.03
080060020	3	0.0029990	0.00	62.03
080060030	5	0.0049984	0.00	62.04
080060040	2	0.0019993	0.00	62.04
080070	3	0.0029990	0.00	62.04
080080	7	0.0069977	0.01	62.05
081	5	0.0049984	0.00	62.06
090	1082	1.0816431	1.08	63.14

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
090010	43	0.0429858	0.04	63.18
090010010	25	0.0249918	0.02	63.21
090010020	12	0.0119960	0.01	63.22
090020	49	0.0489838	0.05	63.27
090020010	9	0.0089970	0.01	63.28
090020020	20	0.0199934	0.02	63.30
090020030	7	0.0069977	0.01	63.30
090030	69	0.0689772	0.07	63.37
090040	32	0.0319894	0.03	63.40
090040010	11	0.0109964	0.01	63.41
090040020	3	0.0029990	0.00	63.42
090050	20	0.0199934	0.02	63.44
090060	1	0.0009997	0.00	63.44
091	11	0.0109964	0.01	63.45
100	581	0.5808083	0.58	64.03
100010	13	0.0129957	0.01	64.04
100010010	5	0.0049984	0.00	64.05
100010020	10	0.0099967	0.01	64.06
100020	25	0.0249918	0.02	64.08
100020010	20	0.0199934	0.02	64.10
100020020	13	0.0129957	0.01	64.12
100020030	3	0.0029990	0.00	64.12
100030	27	0.0269911	0.03	64.15
100030010	4	0.0039987	0.00	64.15
100030020	4	0.0039987	0.00	64.15
100040	6	0.0059980	0.01	64.16
100050	14	0.0139954	0.01	64.17
100051	1	0.0009997	0.00	64.17
100060	8	0.0079974	0.01	64.18

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
100070	1	0.0009997	0.00	64.18
100080	2	0.0019993	0.00	64.19
100090	4	0.0039987	0.00	64.19
100100	1	0.0009997	0.00	64.19
101	11	0.0109964	0.01	64.20
102	11	0.0109964	0.01	64.21
102010	1	0.0009997	0.00	64.21
102020	2	0.0019993	0.00	64.22
102030010	10	0.0099967	0.01	64.23
102030020010	4	0.0039987	0.00	64.23
102030020020	6	0.0059980	0.01	64.24
102030020030	1	0.0009997	0.00	64.24
102030030	5	0.0049984	0.00	64.24
103	1	0.0009997	0.00	64.24
103010	1	0.0009997	0.00	64.24
103020	1	0.0009997	0.00	64.24
103030	1	0.0009997	0.00	64.25
103040	1	0.0009997	0.00	64.25
103050	1	0.0009997	0.00	64.25
103060	1	0.0009997	0.00	64.25
103070	1	0.0009997	0.00	64.25
103080	1	0.0009997	0.00	64.25
103090	1	0.0009997	0.00	64.25
103100	1	0.0009997	0.00	64.25
103110	1	0.0009997	0.00	64.25
103120	1	0.0009997	0.00	64.25
103130	1	0.0009997	0.00	64.26
104	1	0.0009997	0.00	64.26
105	1	0.0009997	0.00	64.26

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
105010	1	0.0009997	0.00	64.26
106	1	0.0009997	0.00	64.26
107	1	0.0009997	0.00	64.26
107010	1	0.0009997	0.00	64.26
108	1	0.0009997	0.00	64.26
109	1	0.0009997	0.00	64.26
110	423	0.4228605	0.42	64.69
110010	61	0.0609799	0.06	64.75
110010010	5	0.0049984	0.00	64.75
110010020	3	0.0029990	0.00	64.76
110010030	5	0.0049984	0.00	64.76
110020	19	0.0189937	0.02	64.78
110020010	14	0.0139954	0.01	64.79
110020020	5	0.0049984	0.00	64.80
110020030	5	0.0049984	0.00	64.80
110030	25	0.0249918	0.02	64.83
110040	7	0.0069977	0.01	64.84
110040010	3	0.0029990	0.00	64.84
110040020	5	0.0049984	0.00	64.84
110050	8	0.0079974	0.01	64.85
110050010	10	0.0099967	0.01	64.86
110050020	4	0.0039987	0.00	64.87
110050020010	4	0.0039987	0.00	64.87
110050020020	6	0.0059980	0.01	64.88
110050020030	2	0.0019993	0.00	64.88
110050020040	2	0.0019993	0.00	64.88
110050030	2	0.0019993	0.00	64.88
110050040	5	0.0049984	0.00	64.89
110050050	2	0.0019993	0.00	64.89

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
110050060	2	0.0019993	0.00	64.89
110050070	2	0.0019993	0.00	64.89
110050080	1	0.0009997	0.00	64.89
110060	10	0.0099967	0.01	64.90
110060010	9	0.0089970	0.01	64.91
110060020	5	0.0049984	0.00	64.92
110070	6	0.0059980	0.01	64.92
110070010	2	0.0019993	0.00	64.93
110070020	6	0.0059980	0.01	64.93
110080	13	0.0129957	0.01	64.94
111	1	0.0009997	0.00	64.95
112	2	0.0019993	0.00	64.95
120	183	0.1829396	0.18	65.13
120010	43	0.0429858	0.04	65.17
120020	24	0.0239921	0.02	65.20
120030	24	0.0239921	0.02	65.22
130	139	0.1389541	0.14	65.36
130010	18	0.0179941	0.02	65.38
130020	24	0.0239921	0.02	65.40
130030	10	0.0099967	0.01	65.41
130040	9	0.0089970	0.01	65.42
130050	26	0.0259914	0.03	65.45
130060	2	0.0019993	0.00	65.45
130070	3	0.0029990	0.00	65.45
130080	5	0.0049984	0.00	65.46
140	106	0.1059650	0.11	65.56
150	93	0.0929693	0.09	65.66
150010	8	0.0079974	0.01	65.66
150020	8	0.0079974	0.01	65.67

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
150030	1	0.0009997	0.00	65.67
160	109	0.1089640	0.11	65.78
160010	12	0.0119960	0.01	65.79
160020	8	0.0079974	0.01	65.80
160030	11	0.0109964	0.01	65.81
170	26	0.0259914	0.03	65.84
171	5	0.0049984	0.00	65.84
172	2	0.0019993	0.00	65.85
174	2	0.0019993	0.00	65.85
180	28	0.0279908	0.03	65.88
180010	35	0.0349885	0.03	65.91
180020	4	0.0039987	0.00	65.92
180030	5	0.0049984	0.00	65.92
180040	1	0.0009997	0.00	65.92
180050	1	0.0009997	0.00	65.92
180060	1	0.0009997	0.00	65.92
180070	1	0.0009997	0.00	65.92
180080	1	0.0009997	0.00	65.93
180090	1	0.0009997	0.00	65.93
180100	1	0.0009997	0.00	65.93
180110	1	0.0009997	0.00	65.93
180120	1	0.0009997	0.00	65.93
180130	1	0.0009997	0.00	65.93
180140	3	0.0029990	0.00	65.93
180160	1	0.0009997	0.00	65.93
180170	1	0.0009997	0.00	65.94
180180	1	0.0009997	0.00	65.94
180190	2	0.0019993	0.00	65.94
190	21	0.0209931	0.02	65.96

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
200	13	0.0129957	0.01	65.97
210	21	0.0209931	0.02	65.99
220	22	0.0219927	0.02	66.02
230	25	0.0249918	0.02	66.04
231	1	0.0009997	0.00	66.04
232	1	0.0009997	0.00	66.04
240	14	0.0139954	0.01	66.06
250	13	0.0129957	0.01	66.07
260	5	0.0049984	0.00	66.07
270	3	0.0029990	0.00	66.08
280	4	0.0039987	0.00	66.08
281	16	0.0159947	0.02	66.10
290	5	0.0049984	0.00	66.10
300	3	0.0029990	0.00	66.11
301	1	0.0009997	0.00	66.11
310	5	0.0049984	0.00	66.11
320	2	0.0019993	0.00	66.11
330	16	0.0159947	0.02	66.13
340	6	0.0059980	0.01	66.14
350	6	0.0059980	0.01	66.14
360	2	0.0019993	0.00	66.14
370	2	0.0019993	0.00	66.15
371	1	0.0009997	0.00	66.15
380	4	0.0039987	0.00	66.15
390	2	0.0019993	0.00	66.15
400	2	0.0019993	0.00	66.15
410	3	0.0029990	0.00	66.16
420	4	0.0039987	0.00	66.16
421	1	0.0009997	0.00	66.16

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
430	2	0.0019993	0.00	66.16
440	4	0.0039987	0.00	66.17
450	1	0.0009997	0.00	66.17
460	6	0.0059980	0.01	66.18
460460	3	0.0029990	0.00	66.18
460490	2	0.0019993	0.00	66.18
460510	8	0.0079974	0.01	66.19
460590	4	0.0039987	0.00	66.19
460620	1	0.0009997	0.00	66.19
470	3	0.0029990	0.00	66.20
480	3	0.0029990	0.00	66.20
491	4	0.0039987	0.00	66.20
510	2	0.0019993	0.00	66.21
520	6	0.0059980	0.01	66.21
530	2	0.0019993	0.00	66.21
550	1	0.0009997	0.00	66.21
560	2	0.0019993	0.00	66.22
580	1	0.0009997	0.00	66.22
590	1	0.0009997	0.00	66.22
600	2	0.0019993	0.00	66.22
610	2	0.0019993	0.00	66.22
620	8	0.0079974	0.01	66.23
630	7	0.0069977	0.01	66.24
640	1	0.0009997	0.00	66.24
650	1	0.0009997	0.00	66.24
680	1	0.0009997	0.00	66.24
681	2	0.0019993	0.00	66.24
690	2	0.0019993	0.00	66.24
700	2	0.0019993	0.00	66.25

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
700010	1	0.0009997	0.00	66.25
700020	1	0.0009997	0.00	66.25
700030	1	0.0009997	0.00	66.25
700031	1	0.0009997	0.00	66.25
700032	1	0.0009997	0.00	66.25
700040	1	0.0009997	0.00	66.25
700050	1	0.0009997	0.00	66.25
700060	1	0.0009997	0.00	66.25
700070	1	0.0009997	0.00	66.26
700080	1	0.0009997	0.00	66.26
700090	1	0.0009997	0.00	66.26
700100	1	0.0009997	0.00	66.26
700110	1	0.0009997	0.00	66.26
700120	1	0.0009997	0.00	66.26
700130	1	0.0009997	0.00	66.26
700140	1	0.0009997	0.00	66.26
700150	1	0.0009997	0.00	66.26
700160	1	0.0009997	0.00	66.26
700170	1	0.0009997	0.00	66.27
700180	1	0.0009997	0.00	66.27
710	2	0.0019993	0.00	66.27
720	7	0.0069977	0.01	66.28
730	1	0.0009997	0.00	66.28
730740	5	0.0049984	0.00	66.28
731	6	0.0059980	0.01	66.29
740	6	0.0059980	0.01	66.29
780	2	0.0019993	0.00	66.30
800	1	0.0009997	0.00	66.30
810	1	0.0009997	0.00	66.30

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
820	1	0.0009997	0.00	66.30
830	5	0.0049984	0.00	66.30
840	4	0.0039987	0.00	66.31
870	1	0.0009997	0.00	66.31
880	5	0.0049984	0.00	66.31
911	1	0.0009997	0.00	66.31
920	1	0.0009997	0.00	66.32
930	1	0.0009997	0.00	66.32
950	1	0.0009997	0.00	66.32
960	2	0.0019993	0.00	66.32
980	1	0.0009997	0.00	66.32
990	1	0.0009997	0.00	66.32
991	3	0.0029990	0.00	66.32
NA	33687	33.6758870	33.68	100.00
Total	100033	100.0000000	99.00	100.00

Frequency Table for Variable: ausfertigung_jahr

123 unique value(s) detected.

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1869	146	0.1459518	0.15	0.15
1871	562	0.5618146	0.56	0.71
1877	100	0.0999670	0.10	0.81
1884	10	0.0099967	0.01	0.82
1886	3	0.0029990	0.00	0.82
1887	4	0.0039987	0.00	0.82
1889	215	0.2149291	0.21	1.04
1892	115	0.1149621	0.11	1.15

(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1894	4	0.0039987	0.00	1.16
1895	74	0.0739756	0.07	1.23
1896	2868	2.8670539	2.87	4.10
1897	1133	1.1326262	1.13	5.23
1898	3	0.0029990	0.00	5.24
1899	2	0.0019993	0.00	5.24
1901	51	0.0509832	0.05	5.29
1903	1	0.0009997	0.00	5.29
1906	1	0.0009997	0.00	5.29
1907	26	0.0259914	0.03	5.32
1908	8	0.0079974	0.01	5.32
1909	129	0.1289574	0.13	5.45
1910	11	0.0109964	0.01	5.46
1911	47	0.0469845	0.05	5.51
1913	46	0.0459848	0.05	5.56
1919	99	0.0989673	0.10	5.66
1920	7	0.0069977	0.01	5.66
1921	40	0.0399868	0.04	5.70
1922	206	0.2059320	0.21	5.91
1923	15	0.0149951	0.01	5.92
1924	2	0.0019993	0.00	5.93
1925	2	0.0019993	0.00	5.93
1926	31	0.0309898	0.03	5.96
1927	32	0.0319894	0.03	5.99
1928	8	0.0079974	0.01	6.00
1929	8	0.0079974	0.01	6.01
1930	14	0.0139954	0.01	6.02
1931	46	0.0459848	0.05	6.07
1933	196	0.1959353	0.20	6.26

(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1934	515	0.5148301	0.51	6.78
1935	157	0.1569482	0.16	6.93
1936	284	0.2839063	0.28	7.22
1937	331	0.3308908	0.33	7.55
1938	42	0.0419861	0.04	7.59
1939	90	0.0899703	0.09	7.68
1940	215	0.2149291	0.21	7.90
1941	25	0.0249918	0.02	7.92
1942	20	0.0199934	0.02	7.94
1943	24	0.0239921	0.02	7.97
1944	12	0.0119960	0.01	7.98
1947	19	0.0189937	0.02	8.00
1948	73	0.0729759	0.07	8.07
1949	314	0.3138964	0.31	8.38
1950	2093	2.0923095	2.09	10.48
1951	585	0.5848070	0.58	11.06
1952	445	0.4448532	0.44	11.51
1953	1821	1.8203993	1.82	13.33
1954	210	0.2099307	0.21	13.54
1955	303	0.3029000	0.30	13.84
1956	706	0.7057671	0.71	14.54
1957	992	0.9916727	0.99	15.54
1958	150	0.1499505	0.15	15.69
1959	751	0.7507523	0.75	16.44
1960	1103	1.1026361	1.10	17.54
1961	1282	1.2815771	1.28	18.82
1962	247	0.2469185	0.25	19.07
1963	336	0.3358892	0.34	19.40
1964	332	0.3318905	0.33	19.74

(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1965	1481	1.4805114	1.48	21.22
1966	570	0.5698120	0.57	21.79
1967	330	0.3298911	0.33	22.12
1968	478	0.4778423	0.48	22.59
1969	721	0.7207621	0.72	23.31
1970	439	0.4388552	0.44	23.75
1971	774	0.7737447	0.77	24.53
1972	825	0.8247278	0.82	25.35
1973	239	0.2389212	0.24	25.59
1974	764	0.7637480	0.76	26.35
1975	913	0.9126988	0.91	27.27
1976	2503	2.5021743	2.50	29.77
1977	742	0.7417552	0.74	30.51
1978	429	0.4288585	0.43	30.94
1979	530	0.5298252	0.53	31.47
1980	1117	1.1166315	1.12	32.59
1981	581	0.5808083	0.58	33.17
1982	542	0.5418212	0.54	33.71
1983	244	0.2439195	0.24	33.95
1984	357	0.3568822	0.36	34.31
1985	601	0.6008017	0.60	34.91
1986	564	0.5638139	0.56	35.47
1987	455	0.4548499	0.45	35.93
1988	1281	1.2805774	1.28	37.21
1989	1142	1.1416233	1.14	38.35
1990	2426	2.4251997	2.43	40.78
1991	780	0.7797427	0.78	41.56
1992	1012	1.0116662	1.01	42.57
1993	992	0.9916727	0.99	43.56

(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1994	3794	3.7927484	3.79	47.35
1995	749	0.7487529	0.75	48.10
1996	948	0.9476873	0.95	49.05
1997	1864	1.8633851	1.86	50.91
1998	1472	1.4715144	1.47	52.38
1999	788	0.7877400	0.79	53.17
2000	703	0.7027681	0.70	53.87
2001	1674	1.6734478	1.67	55.55
2002	2165	2.1642858	2.16	57.71
2003	1197	1.1966051	1.20	58.91
2004	2727	2.7261004	2.73	61.63
2005	2061	2.0603201	2.06	63.69
2006	2214	2.2132696	2.21	65.91
2007	1663	1.6624514	1.66	67.57
2008	2164	2.1632861	2.16	69.73
2009	2823	2.8220687	2.82	72.56
2010	1450	1.4495217	1.45	74.01
2011	2361	2.3602211	2.36	76.37
2012	1511	1.5105015	1.51	77.88
2013	3471	3.4698549	3.47	81.35
2014	1636	1.6354603	1.64	82.98
2015	2039	2.0383274	2.04	85.02
2016	3117	3.1159717	3.12	88.14
2017	3535	3.5338338	3.53	91.67
2018	1635	1.6344606	1.63	93.30
2019	1733	1.7324283	1.73	95.04
2020	2884	2.8830486	2.88	97.92
2021	2081	2.0803135	2.08	100.00
Total	100033	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	100033	100	100	100
Total	100033	100	100	100

Frequency Table for Variable: doi_version

1 unique value(s) detected.

doi_version	N	exactpercent	roundedpercent	cumulpercent
10.5281/zenodo.5510458	100033	100	100	100
Total	100033	100	100	100

Frequency Table for Variable: version

1 unique value(s) detected.

version	N	exactpercent	roundedpercent	cumulpercent
2021-09-16	100033	100	100	100
Total	100033	100	100	100

Frequency Table for Variable: lizenz

1 unique value(s) detected.

lizenz	N	exactpercent	roundedpercent	cumulpercent
Creative Commons Zero 1.0 Universal	100033	100	100	100
Total	100033	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_Frequenztabelle_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	0.0177179	0.02	0.02
BAAnz	143	2.5336641	2.53	2.55
BGBI	21	0.3720765	0.37	2.92
BGBI I	4862	86.1445783	86.14	89.07
BGBI II	375	6.6442240	6.64	95.71
GBl DDR	15	0.2657690	0.27	95.98
GBl DDR I	43	0.7618710	0.76	96.74
GBl DDR II	3	0.0531538	0.05	96.79
GVBl BE	1	0.0177179	0.02	96.81
NV	1	0.0177179	0.02	96.83
RAAnz	2	0.0354359	0.04	96.86
RGBI	51	0.9036145	0.90	97.77
RGBI I	47	0.8327427	0.83	98.60
RGBI II	45	0.7973069	0.80	99.40
RMBI	4	0.0708717	0.07	99.47
VOBl BrZ	1	0.0177179	0.02	99.49
VkBl	10	0.1771793	0.18	99.66
WiGBl	7	0.1240255	0.12	99.79
ZBl	2	0.0354359	0.04	99.82
eBAAnz	6	0.1063076	0.11	99.93
Öff Anz	4	0.0708717	0.07	100.00
Total	5644	100.0000000	100.00	100.00

Frequency Table for Variable: fundstellentyp

1 unique value(s) detected.

fundstellentyp	N	exactpercent	roundedpercent	cumulpercent
amtlich	5644	100	100	100
Total	5644	100	100	100

Frequency Table for Variable: check_neuf

2 unique value(s) detected.

check_neuf	N	exactpercent	roundedpercent	cumulpercent
NA	5229	92.647059	92.65	92.65
ja	415	7.352941	7.35	100.00
Total	5644	100.000000	100.00	100.00

Frequency Table for Variable: check_aufh

2 unique value(s) detected.

check_aufh	N	exactpercent	roundedpercent	cumulpercent
NA	5452	96.598157	96.6	96.6
ja	192	3.401843	3.4	100.0
Total	5644	100.000000	100.0	100.0

Frequency Table for Variable: check_sonst

2 unique value(s) detected.

check_sonst	N	exactpercent	roundedpercent	cumulpercent
NA	5149	91.229624	91.23	91.23
ja	495	8.770376	8.77	100.00
Total	5644	100.000000	100.00	100.00

Frequency Table for Variable: check_hinweis

2 unique value(s) detected.

check_hinweis	N	exactpercent	roundedpercent	cumulpercent
NA	5064	89.7236	89.72	89.72
ja	580	10.2764	10.28	100.00
Total	5644	100.0000	100.00	100.00

Frequency Table for Variable: check_stand

2 unique value(s) detected.

check_stand	N	exactpercent	roundedpercent	cumulpercent
NA	2858	50.63785	50.64	50.64
ja	2786	49.36215	49.36	100.00
Total	5644	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	0.0177179	0.02	0.02
1871	3	0.0531538	0.05	0.07
1877	3	0.0531538	0.05	0.12
1884	1	0.0177179	0.02	0.14
1886	2	0.0354359	0.04	0.18
1887	1	0.0177179	0.02	0.19
1889	2	0.0354359	0.04	0.23
1892	1	0.0177179	0.02	0.25
1894	2	0.0354359	0.04	0.28
1895	2	0.0354359	0.04	0.32
1896	2	0.0354359	0.04	0.35
1897	6	0.1063076	0.11	0.46

(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1898	1	0.0177179	0.02	0.48
1899	2	0.0354359	0.04	0.51
1901	2	0.0354359	0.04	0.55
1903	1	0.0177179	0.02	0.57
1906	1	0.0177179	0.02	0.58
1907	1	0.0177179	0.02	0.60
1908	1	0.0177179	0.02	0.62
1909	3	0.0531538	0.05	0.67
1910	2	0.0354359	0.04	0.71
1911	2	0.0354359	0.04	0.74
1913	2	0.0354359	0.04	0.78
1919	4	0.0708717	0.07	0.85
1920	1	0.0177179	0.02	0.87
1921	4	0.0708717	0.07	0.94
1922	6	0.1063076	0.11	1.05
1923	2	0.0354359	0.04	1.08
1924	2	0.0354359	0.04	1.12
1925	2	0.0354359	0.04	1.15
1926	4	0.0708717	0.07	1.22
1927	5	0.0885897	0.09	1.31
1928	3	0.0531538	0.05	1.36
1929	3	0.0531538	0.05	1.42
1930	3	0.0531538	0.05	1.47
1931	5	0.0885897	0.09	1.56
1933	8	0.1417434	0.14	1.70
1934	7	0.1240255	0.12	1.82
1935	3	0.0531538	0.05	1.88
1936	7	0.1240255	0.12	2.00
1937	9	0.1594614	0.16	2.16

(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1938	10	0.1771793	0.18	2.34
1939	6	0.1063076	0.11	2.45
1940	7	0.1240255	0.12	2.57
1941	2	0.0354359	0.04	2.60
1942	2	0.0354359	0.04	2.64
1943	1	0.0177179	0.02	2.66
1944	1	0.0177179	0.02	2.68
1947	1	0.0177179	0.02	2.69
1948	5	0.0885897	0.09	2.78
1949	12	0.2126152	0.21	2.99
1950	21	0.3720765	0.37	3.37
1951	32	0.5669738	0.57	3.93
1952	14	0.2480510	0.25	4.18
1953	41	0.7264352	0.73	4.91
1954	17	0.3012048	0.30	5.21
1955	26	0.4606662	0.46	5.67
1956	33	0.5846917	0.58	6.25
1957	46	0.8150248	0.82	7.07
1958	20	0.3543586	0.35	7.42
1959	27	0.4783841	0.48	7.90
1960	33	0.5846917	0.58	8.49
1961	52	0.9213324	0.92	9.41
1962	31	0.5492558	0.55	9.96
1963	26	0.4606662	0.46	10.42
1964	28	0.4961021	0.50	10.91
1965	45	0.7973069	0.80	11.71
1966	25	0.4429483	0.44	12.15
1967	39	0.6909993	0.69	12.85
1968	41	0.7264352	0.73	13.57

(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1969	54	0.9567682	0.96	14.53
1970	45	0.7973069	0.80	15.33
1971	54	0.9567682	0.96	16.28
1972	53	0.9390503	0.94	17.22
1973	33	0.5846917	0.58	17.81
1974	52	0.9213324	0.92	18.73
1975	64	1.1339476	1.13	19.86
1976	73	1.2934089	1.29	21.16
1977	58	1.0276400	1.03	22.18
1978	45	0.7973069	0.80	22.98
1979	40	0.7087172	0.71	23.69
1980	54	0.9567682	0.96	24.65
1981	41	0.7264352	0.73	25.37
1982	42	0.7441531	0.74	26.12
1983	29	0.5138200	0.51	26.63
1984	32	0.5669738	0.57	27.20
1985	40	0.7087172	0.71	27.91
1986	51	0.9036145	0.90	28.81
1987	38	0.6732814	0.67	29.48
1988	48	0.8504607	0.85	30.33
1989	50	0.8858965	0.89	31.22
1990	157	2.7817151	2.78	34.00
1991	77	1.3642807	1.36	35.36
1992	77	1.3642807	1.36	36.73
1993	82	1.4528703	1.45	38.18
1994	163	2.8880227	2.89	41.07
1995	70	1.2402551	1.24	42.31
1996	87	1.5414600	1.54	43.85
1997	108	1.9135365	1.91	45.77

(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1998	122	2.1615875	2.16	47.93
1999	78	1.3819986	1.38	49.31
2000	60	1.0630758	1.06	50.37
2001	114	2.0198441	2.02	52.39
2002	147	2.6045358	2.60	55.00
2003	99	1.7540751	1.75	56.75
2004	142	2.5159461	2.52	59.27
2005	135	2.3919206	2.39	61.66
2006	129	2.2856130	2.29	63.94
2007	104	1.8426648	1.84	65.79
2008	125	2.2147413	2.21	68.00
2009	159	2.8171510	2.82	70.82
2010	122	2.1615875	2.16	72.98
2011	121	2.1438696	2.14	75.12
2012	109	1.9312544	1.93	77.06
2013	186	3.2955351	3.30	80.35
2014	103	1.8249468	1.82	82.18
2015	131	2.3210489	2.32	84.50
2016	147	2.6045358	2.60	87.10
2017	178	3.1537916	3.15	90.26
2018	95	1.6832034	1.68	91.94
2019	121	2.1438696	2.14	94.08
2020	185	3.2778172	3.28	97.36
2021	149	2.6399717	2.64	100.00
Total	5644	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	5644	100	100	100
Total	5644	100	100	100

Frequency Table for Variable: doi_version

1 unique value(s) detected.

doi_version	N	exactpercent	roundedpercent	cumulpercent
10.5281/zenodo.5510458	5644	100	100	100
Total	5644	100	100	100

Frequency Table for Variable: version

1 unique value(s) detected.

version	N	exactpercent	roundedpercent	cumulpercent
2021-09-16	5644	100	100	100
Total	5644	100	100	100

Frequency Table for Variable: lizenz

1 unique value(s) detected.

	lizenz	N	exactpercent	roundedpercent	cumulpercent
Creative Commons Zero 1.0 Universal		5644	100	100	100
	Total	5644	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	0.0602591	0.06	0.06
ABl EU	1	0.0150648	0.02	0.08
BAGVBl	2	0.0301296	0.03	0.11
BAnz	254	3.8264538	3.83	3.93
BGBI	23	0.3464899	0.35	4.28
BGBI I	4879	73.5010545	73.50	77.78
BGBI II	1171	17.6408557	17.64	95.42

(continued)

periodikum	N	exactpercent	roundedpercent	cumulpercent
BGBI III	1	0.0150648	0.02	95.44
BremGBI	4	0.0602591	0.06	95.50
GBI BW	1	0.0150648	0.02	95.51
GBI DDR	15	0.2259717	0.23	95.74
GBI DDR I	44	0.6628503	0.66	96.40
GBI DDR II	3	0.0451943	0.05	96.44
GV NW	2	0.0301296	0.03	96.47
GVB I BB	1	0.0150648	0.02	96.49
GVB I BE	1	0.0150648	0.02	96.50
GVB I BY	4	0.0602591	0.06	96.57
GVB I HE	5	0.0753239	0.08	96.64
GVB I RP I	2	0.0301296	0.03	96.67
Mtbl BAA	1	0.0150648	0.02	96.69
NV	4	0.0602591	0.06	96.75
Nds GVB I	1	0.0150648	0.02	96.76
RAnz	2	0.0301296	0.03	96.79
RBesBI	1	0.0150648	0.02	96.81
RGBI	56	0.8436276	0.84	97.65
RGBI I	54	0.8134980	0.81	98.46
RGBI II	48	0.7231094	0.72	99.19
RMBI	4	0.0602591	0.06	99.25
RegBI WB	4	0.0602591	0.06	99.31
RegBI WH	1	0.0150648	0.02	99.32
VOBI BrZ	3	0.0451943	0.05	99.37
VkBI	23	0.3464899	0.35	99.71
WiGBI	7	0.1054534	0.11	99.82
ZBI	2	0.0301296	0.03	99.85
eBAnz	6	0.0903887	0.09	99.94
Öff Anz	4	0.0602591	0.06	100.00

(continued)

periodikum	N	exactpercent	roundedpercent	cumulpercent
Total	6638	100.0000000	100.00	100.00

Frequency Table for Variable: fundstellentyp

2 unique value(s) detected.

fundstellentyp	N	exactpercent	roundedpercent	cumulpercent
amtlich	6634	99.9397409	99.94	99.94
nichtamtlich	4	0.0602591	0.06	100.00
Total	6638	100.0000000	100.00	100.00

Frequency Table for Variable: check_neuf

2 unique value(s) detected.

check_neuf	N	exactpercent	roundedpercent	cumulpercent
NA	6223	93.748117	93.75	93.75
ja	415	6.251883	6.25	100.00
Total	6638	100.000000	100.00	100.00

Frequency Table for Variable: check_aufh

2 unique value(s) detected.

check_aufh	N	exactpercent	roundedpercent	cumulpercent
NA	6435	96.94185	96.94	96.94
ja	203	3.05815	3.06	100.00
Total	6638	100.00000	100.00	100.00

Frequency Table for Variable: check_sonst

2 unique value(s) detected.

check_sonst	N	exactpercent	roundedpercent	cumulpercent
NA	6134	92.407352	92.41	92.41
ja	504	7.592648	7.59	100.00
Total	6638	100.000000	100.00	100.00

Frequency Table for Variable: check_hinweis

2 unique value(s) detected.

check_hinweis	N	exactpercent	roundedpercent	cumulpercent
NA	6058	91.262428	91.26	91.26
ja	580	8.737572	8.74	100.00
Total	6638	100.000000	100.00	100.00

Frequency Table for Variable: check_stand

2 unique value(s) detected.

check_stand	N	exactpercent	roundedpercent	cumulpercent
NA	3850	57.9994	58	58
ja	2788	42.0006	42	100
Total	6638	100.0000	100	100

Frequency Table for Variable: ausfertigung_jahr

126 unique value(s) detected.

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
NA	1	0.0150648	0.02	0.02
1869	1	0.0150648	0.02	0.03
1871	3	0.0451943	0.05	0.08
1877	3	0.0451943	0.05	0.12
1884	1	0.0150648	0.02	0.14
1886	2	0.0301296	0.03	0.17
1887	1	0.0150648	0.02	0.18
1889	2	0.0301296	0.03	0.21
1892	1	0.0150648	0.02	0.23
1894	2	0.0301296	0.03	0.26
1895	2	0.0301296	0.03	0.29
1896	2	0.0301296	0.03	0.32
1897	6	0.0903887	0.09	0.41
1898	1	0.0150648	0.02	0.42
1899	2	0.0301296	0.03	0.45
1901	2	0.0301296	0.03	0.48
1903	1	0.0150648	0.02	0.50
1906	1	0.0150648	0.02	0.51
1907	1	0.0150648	0.02	0.53
1908	1	0.0150648	0.02	0.54
1909	4	0.0602591	0.06	0.60
1910	2	0.0301296	0.03	0.63
1911	3	0.0451943	0.05	0.68
1913	2	0.0301296	0.03	0.71
1919	7	0.1054534	0.11	0.81
1920	1	0.0150648	0.02	0.83
1921	4	0.0602591	0.06	0.89
1922	6	0.0903887	0.09	0.98
1923	2	0.0301296	0.03	1.01
1924	2	0.0301296	0.03	1.04

(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1925	2	0.0301296	0.03	1.07
1926	4	0.0602591	0.06	1.13
1927	5	0.0753239	0.08	1.21
1928	4	0.0602591	0.06	1.27
1929	4	0.0602591	0.06	1.33
1930	3	0.0451943	0.05	1.37
1931	5	0.0753239	0.08	1.45
1933	9	0.1355830	0.14	1.58
1934	7	0.1054534	0.11	1.69
1935	3	0.0451943	0.05	1.73
1936	9	0.1355830	0.14	1.87
1937	9	0.1355830	0.14	2.00
1938	11	0.1657126	0.17	2.17
1939	11	0.1657126	0.17	2.34
1940	8	0.1205182	0.12	2.46
1941	2	0.0301296	0.03	2.49
1942	2	0.0301296	0.03	2.52
1943	1	0.0150648	0.02	2.53
1944	4	0.0602591	0.06	2.59
1945	1	0.0150648	0.02	2.61
1946	12	0.1807773	0.18	2.79
1947	11	0.1657126	0.17	2.95
1948	13	0.1958421	0.20	3.15
1949	18	0.2711660	0.27	3.42
1950	23	0.3464899	0.35	3.77
1951	41	0.6176559	0.62	4.38
1952	24	0.3615547	0.36	4.75
1953	47	0.7080446	0.71	5.45
1954	26	0.3916842	0.39	5.85

(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1955	34	0.5122025	0.51	6.36
1956	54	0.8134980	0.81	7.17
1957	56	0.8436276	0.84	8.01
1958	36	0.5423320	0.54	8.56
1959	48	0.7231094	0.72	9.28
1960	45	0.6779150	0.68	9.96
1961	69	1.0394697	1.04	11.00
1962	41	0.6176559	0.62	11.61
1963	36	0.5423320	0.54	12.16
1964	42	0.6327207	0.63	12.79
1965	61	0.9189515	0.92	13.71
1966	38	0.5724616	0.57	14.28
1967	50	0.7532389	0.75	15.03
1968	50	0.7532389	0.75	15.79
1969	71	1.0695993	1.07	16.86
1970	57	0.8586924	0.86	17.72
1971	67	1.0093402	1.01	18.73
1972	66	0.9942754	0.99	19.72
1973	69	1.0394697	1.04	20.76
1974	60	0.9038867	0.90	21.66
1975	74	1.1147936	1.11	22.78
1976	82	1.2353118	1.24	24.01
1977	71	1.0695993	1.07	25.08
1978	58	0.8737572	0.87	25.96
1979	56	0.8436276	0.84	26.80
1980	69	1.0394697	1.04	27.84
1981	48	0.7231094	0.72	28.56
1982	47	0.7080446	0.71	29.27
1983	40	0.6025911	0.60	29.87

(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1984	52	0.7833685	0.78	30.66
1985	52	0.7833685	0.78	31.44
1986	74	1.1147936	1.11	32.55
1987	46	0.6929798	0.69	33.25
1988	61	0.9189515	0.92	34.17
1989	59	0.8888219	0.89	35.06
1990	171	2.5760771	2.58	37.63
1991	84	1.2654414	1.27	38.90
1992	92	1.3859596	1.39	40.28
1993	99	1.4914131	1.49	41.77
1994	189	2.8472431	2.85	44.62
1995	113	1.7023200	1.70	46.32
1996	107	1.6119313	1.61	47.94
1997	139	2.0940042	2.09	50.03
1998	137	2.0638747	2.06	52.09
1999	94	1.4160892	1.42	53.51
2000	72	1.0846641	1.08	54.59
2001	129	1.9433564	1.94	56.54
2002	154	2.3199759	2.32	58.86
2003	118	1.7776439	1.78	60.64
2004	154	2.3199759	2.32	62.96
2005	157	2.3651702	2.37	65.32
2006	138	2.0789394	2.08	67.40
2007	118	1.7776439	1.78	69.18
2008	148	2.2295872	2.23	71.41
2009	172	2.5911419	2.59	74.00
2010	139	2.0940042	2.09	76.09
2011	134	2.0186803	2.02	78.11
2012	123	1.8529678	1.85	79.96

(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
2013	190	2.8623079	2.86	82.83
2014	104	1.5667370	1.57	84.39
2015	133	2.0036155	2.00	86.40
2016	152	2.2898463	2.29	88.69
2017	181	2.7267249	2.73	91.41
2018	97	1.4612835	1.46	92.87
2019	125	1.8830973	1.88	94.76
2020	190	2.8623079	2.86	97.62
2021	158	2.3802350	2.38	100.00
Total	6638	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	6638	100	100	100
Total	6638	100	100	100

Frequency Table for Variable: doi_version

1 unique value(s) detected.

doi_version	N	exactpercent	roundedpercent	cumulpercent
10.5281/zenodo.5510458	6638	100	100	100
Total	6638	100	100	100

Frequency Table for Variable: version

1 unique value(s) detected.

version	N	exactpercent	roundedpercent	cumulpercent
2021-09-16	6638	100	100	100
Total	6638	100	100	100

Frequency Table for Variable: lizenz

1 unique value(s) detected.

	lizenz	N	exactpercent	roundedpercent	cumulpercent
Creative Commons Zero 1.0 Universal		6638	100	100	100
	Total	6638	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"))
```

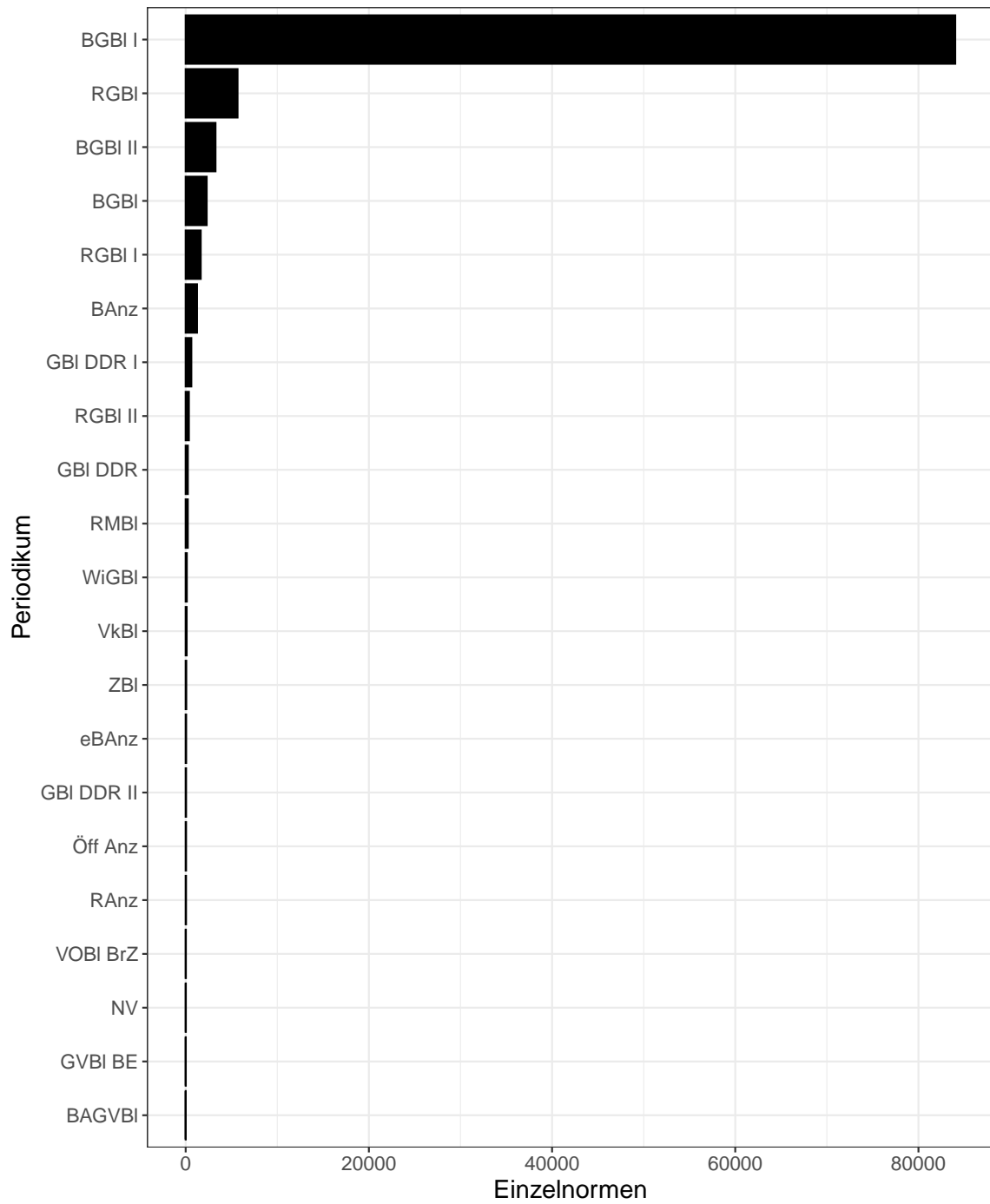
10.5 Periodikum

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 2021-09-16 | Einzelnormen je Periodikum



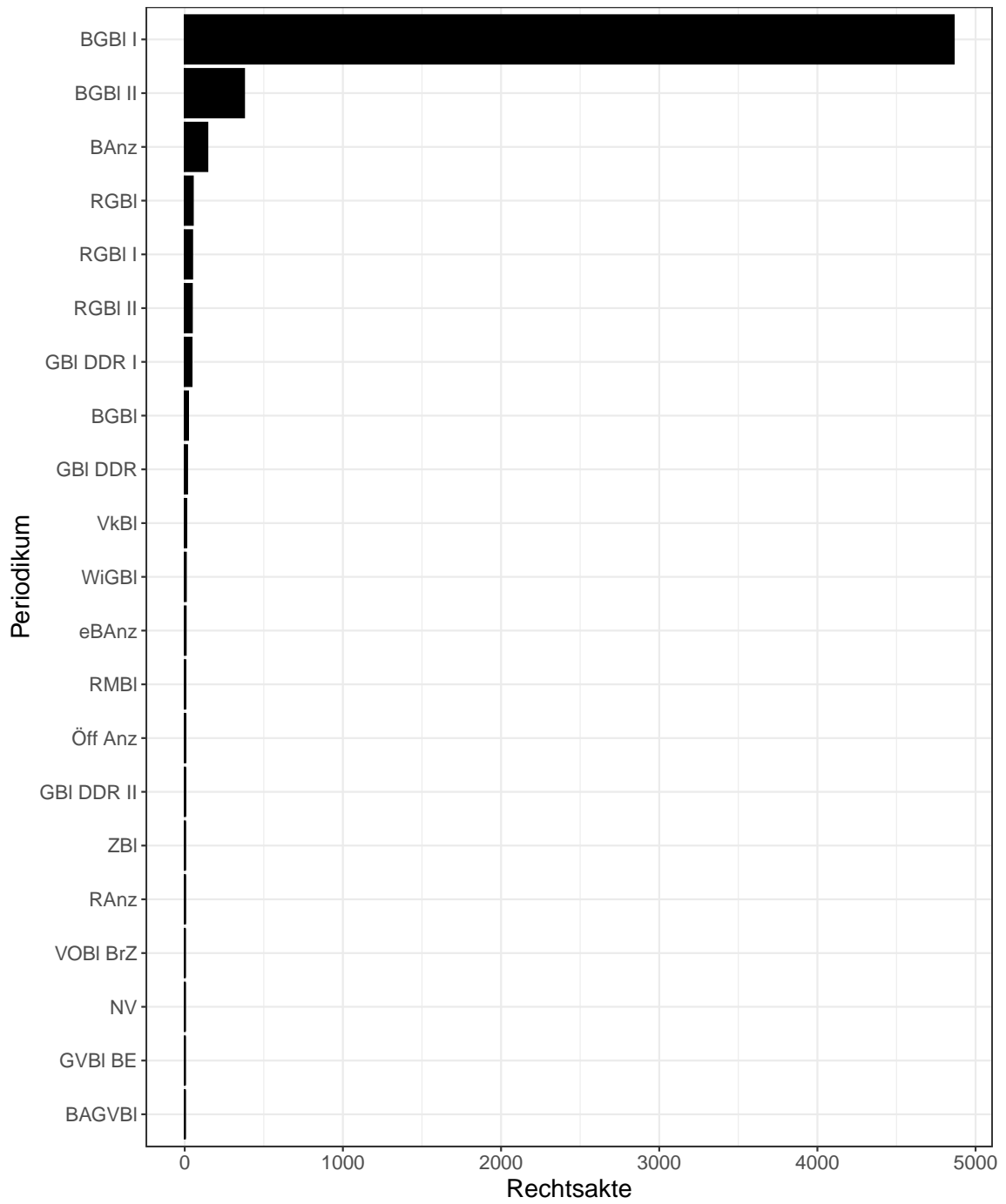
DOI: 10.5281/zenodo.5510458 | S. Fobbe

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 2021-09-16 | Rechtsakte mit Inhalt je Periodikum



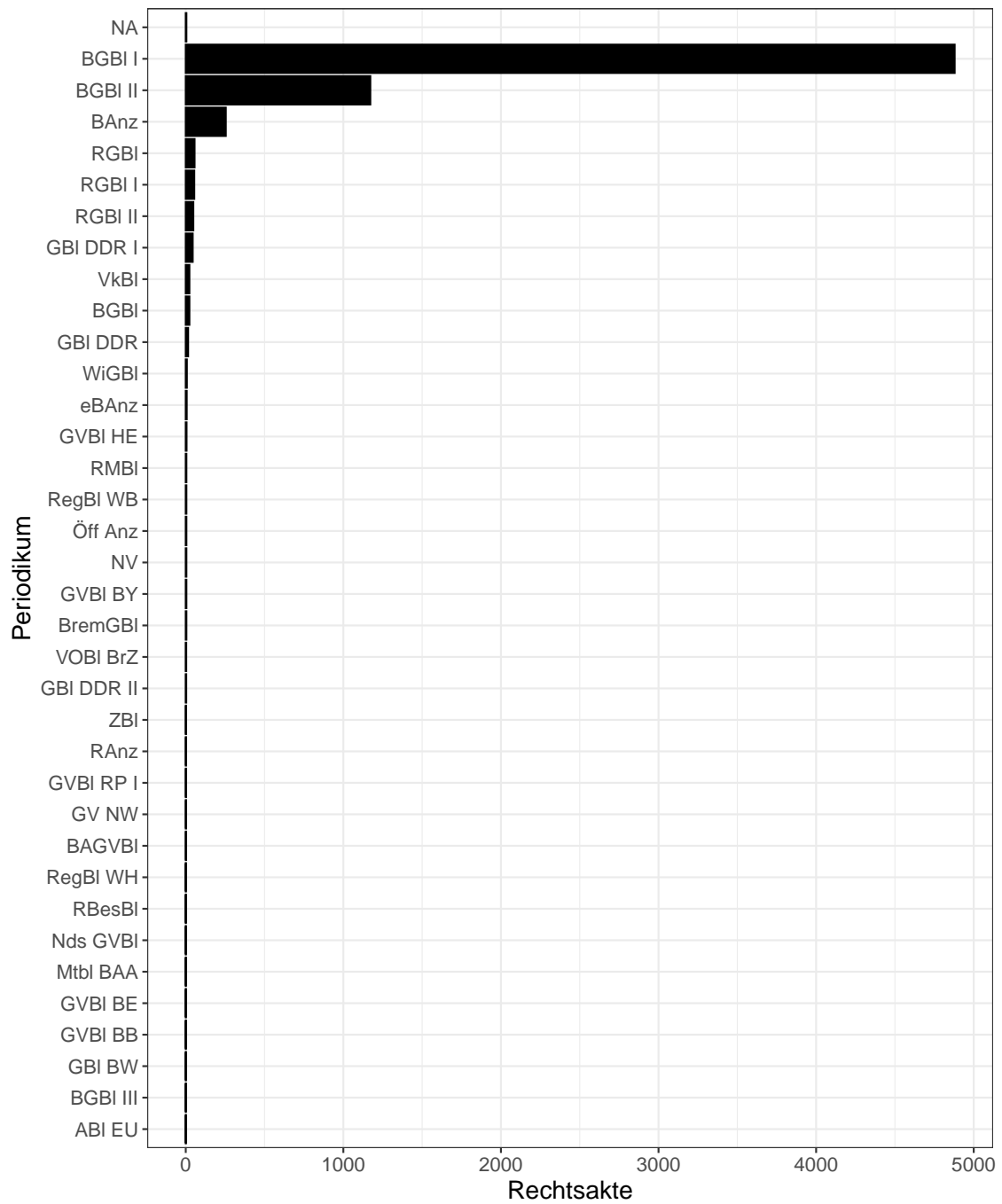
DOI: 10.5281/zenodo.5510458 | S. Fobbe

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 2021-09-16 | Rechtsakte nach Metadaten je Periodikum



DOI: 10.5281/zenodo.5510458 | S. Fobbe

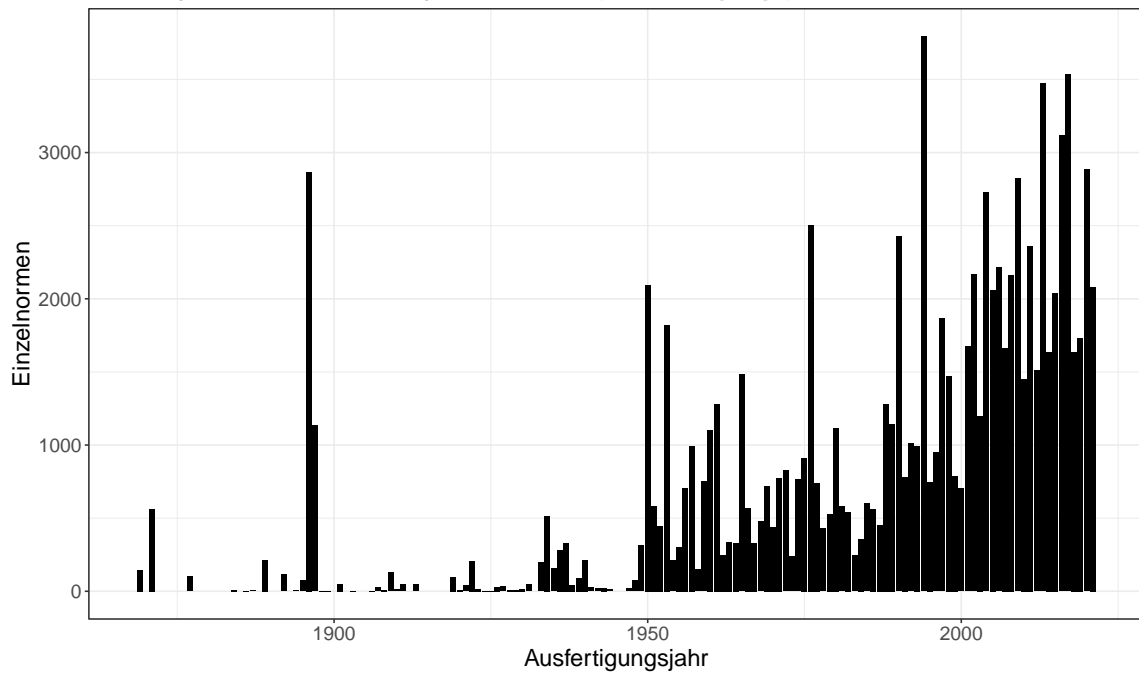
10.6 Ausfertigungsjahr

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)  
  )
```

C-DBR | Version 2021-09-16 | Einzelnormen je Ausfertigungsjahr

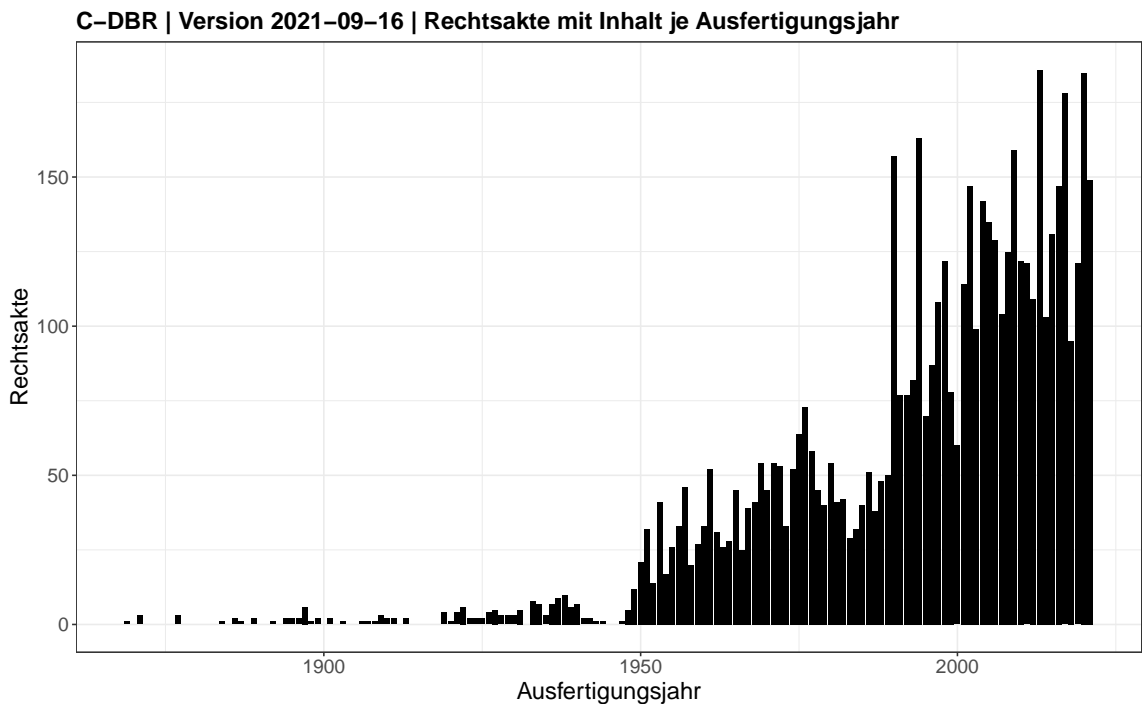


DOI: 10.5281/zenodo.5510458 | S. Fobbe

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)  
  )
```



DOI: 10.5281/zenodo.5510458 | S. Fobbe

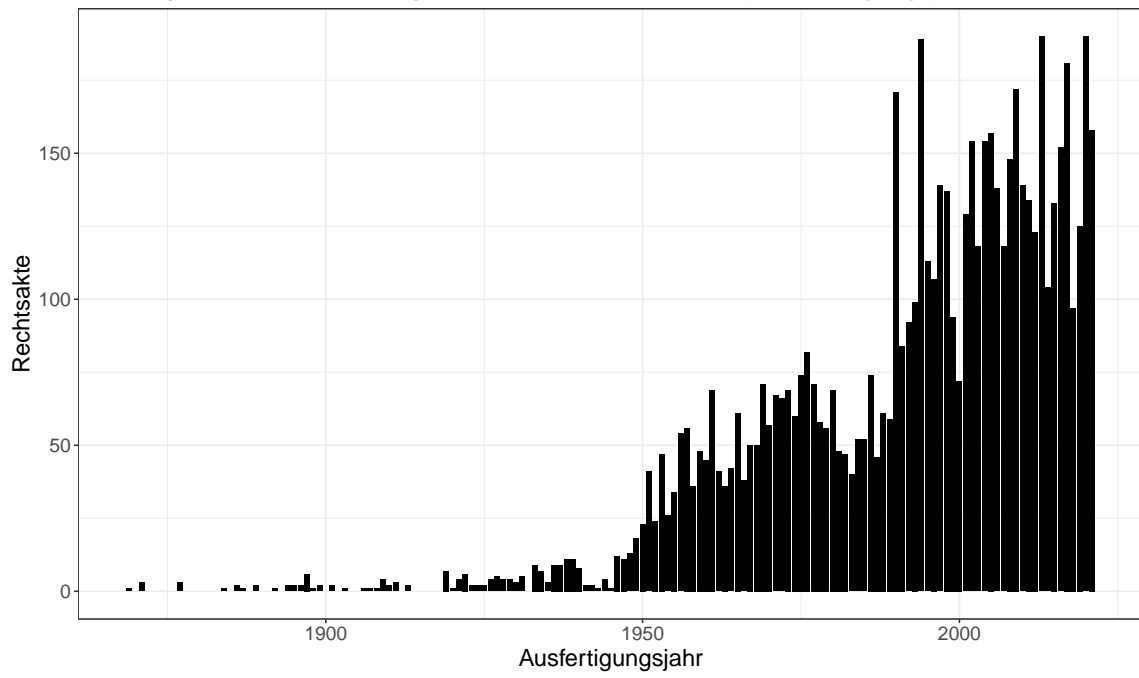
10.6.3 XML-Metadaten

```
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 2021-09-16 | Rechtsakte nach Metadaten je Ausfertigungsjahr



DOI: 10.5281/zenodo.5510458 | 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 2021-09-16 00:21:09.
##      Processing 100033 documents with a total length of 140025629 characters."
## [1] "Runtime was 5.08 mins. Ended at 2021-09-16 00:26:14."
```

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

```
## [1] "Parallel processing using 16 threads. Begin at 2021-09-16 00:26:14.
##      Processing 5644 documents with a total length of 140120018 characters."
## [1] "Runtime was 28.47 secs. Ended at 2021-09-16 00:26:42."
```

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	140,025,629	1	292	683	1,399.794358	1,477	398,079
tokens	21,926,646	1	46	108	219.194126	232	56,584
typen	415,598	1	36	66	92.684644	114	23,559
saetze	734,191	1	2	4	7.339488	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	140,120,018	94	1,570	5,669.5	24,826.3675	22,256.25	2,077,592
tokens	21,926,646	17	265	912.5	3,884.9479	3,385.00	305,340
typen	415,598	15	130	301.0	637.9848	802.25	23,606
saetze	722,734	1	13	35.0	128.0535	113.25	14,306

11.6.3 Zusammenfassungen speichern

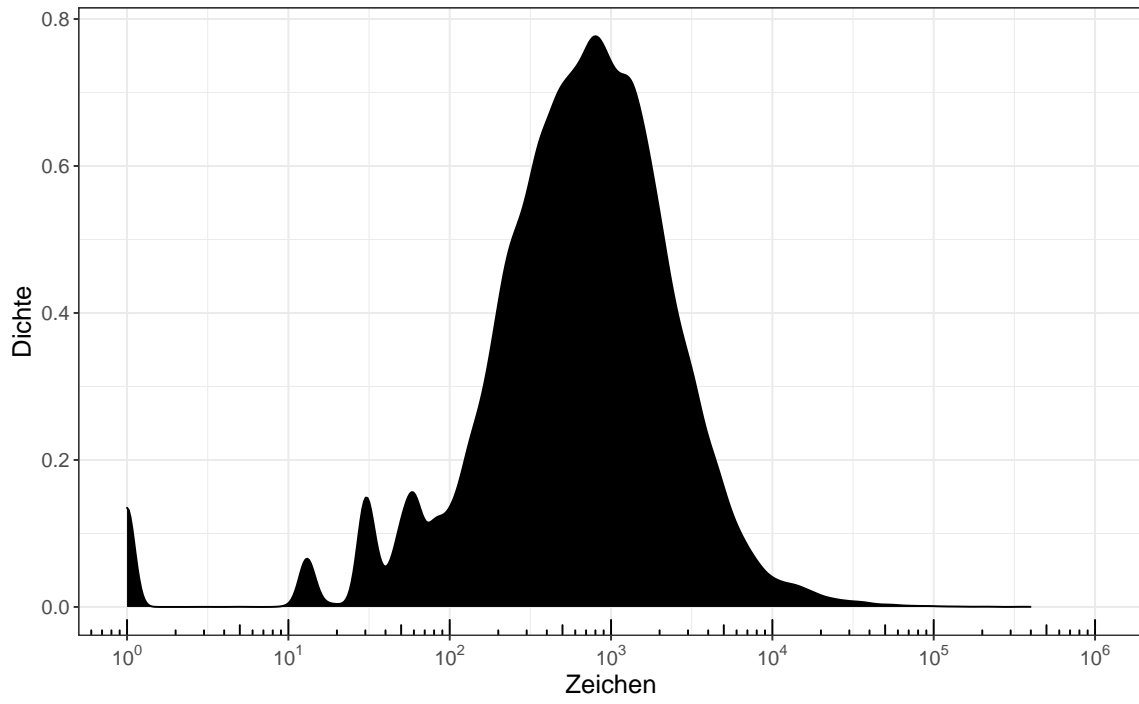
```
fwrite(dt.stats.ling,  
       paste0(outputdir,  
              datasetname,  
              "_00_Rechtsakte_KorpusStatistik_ZusammenfassungLinguistisch.csv"),  
       na = "NA")
```

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)
  )
```

C-DBR | Version 2021-09-16 | Verteilung der Zeichen je Norm



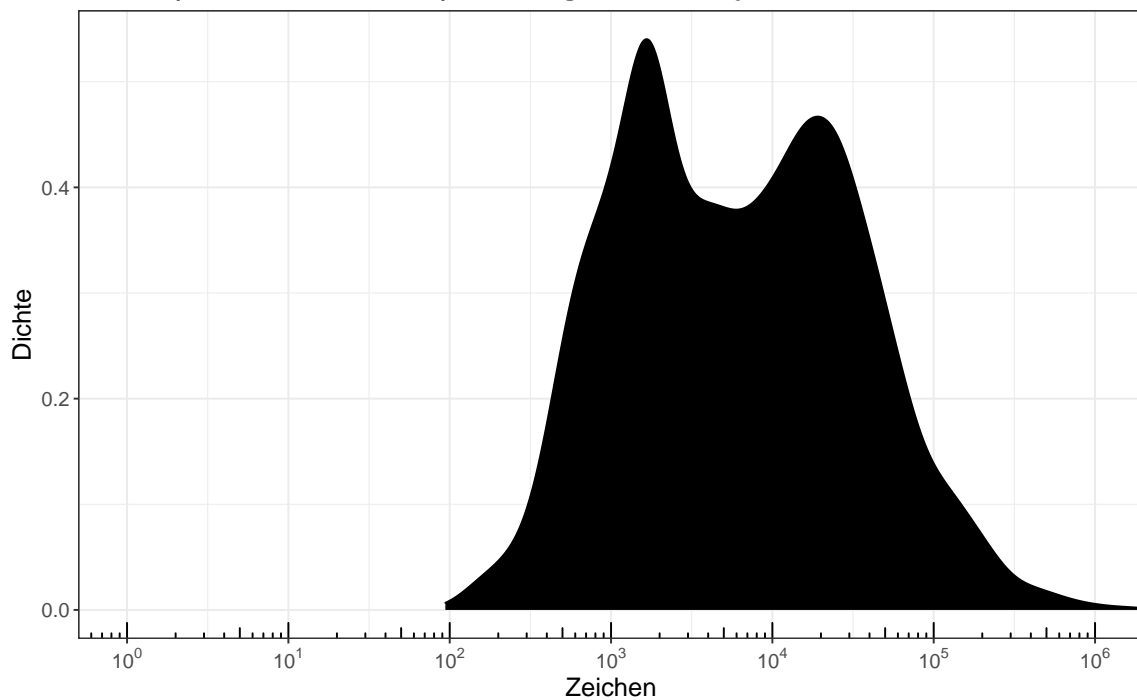
DOI: 10.5281/zenodo.5510458 | S. Fobbe

```

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)
  )

```

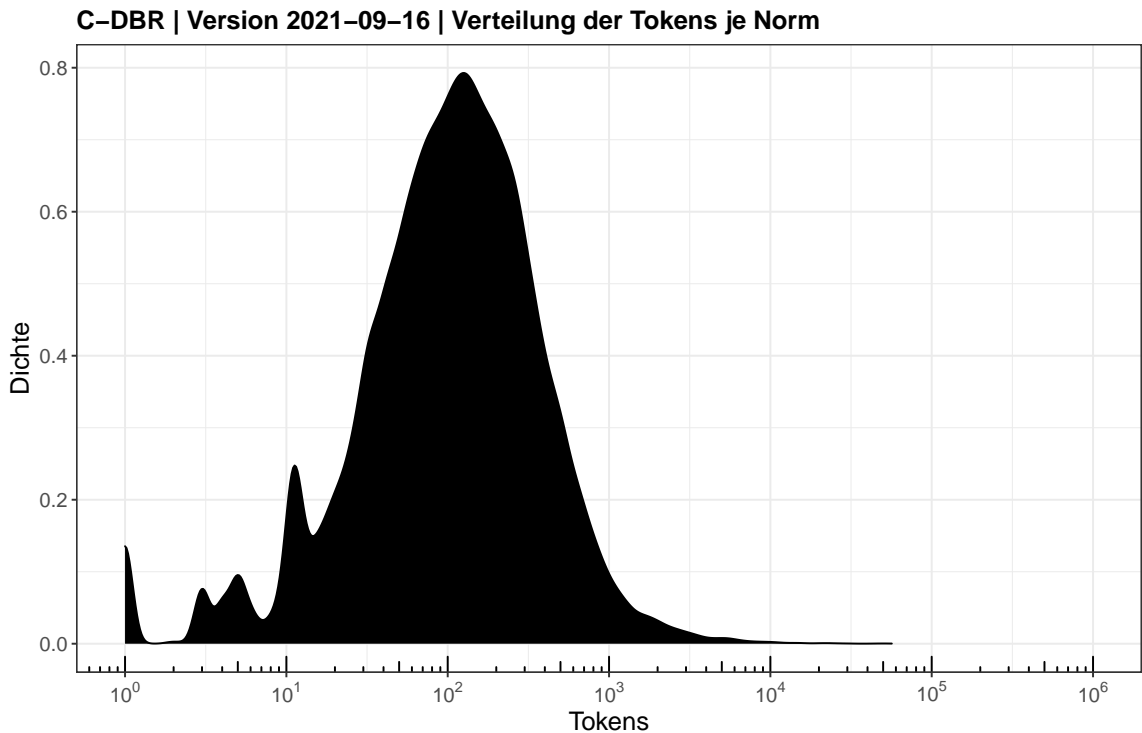
C-DBR | Version 2021-09-16 | Verteilung der Zeichen je Rechtsakt



DOI: 10.5281/zenodo.5510458 | 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)
  )
)
```



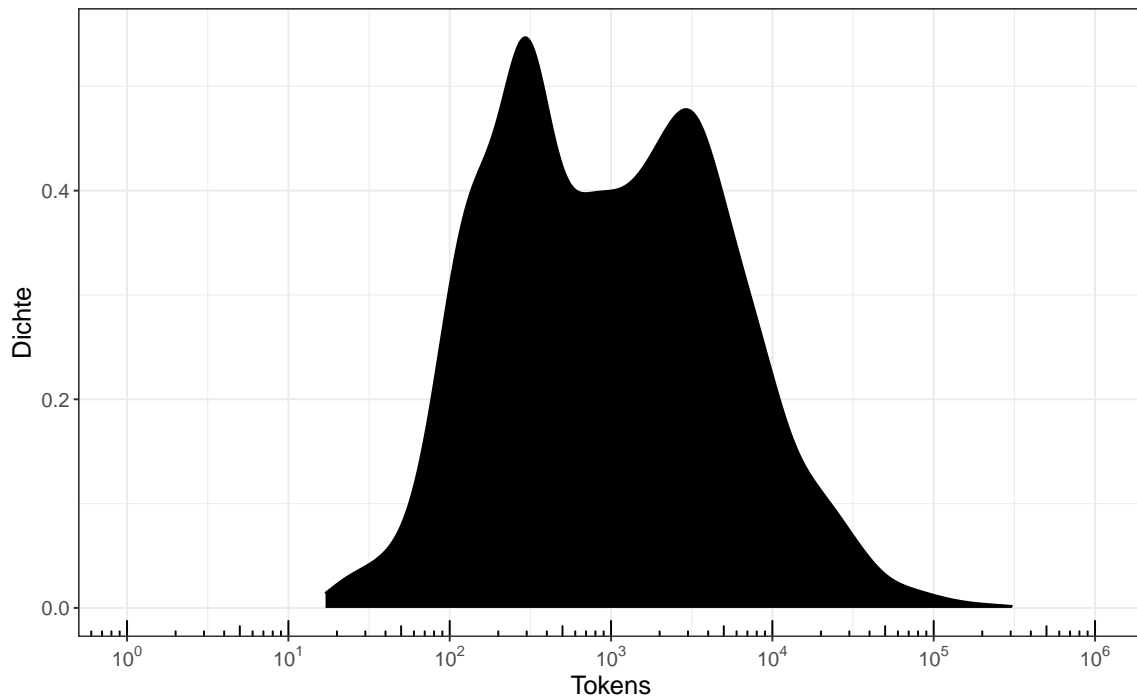
DOI: 10.5281/zenodo.5510458 | S. Fobbe

```

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)
  )

```

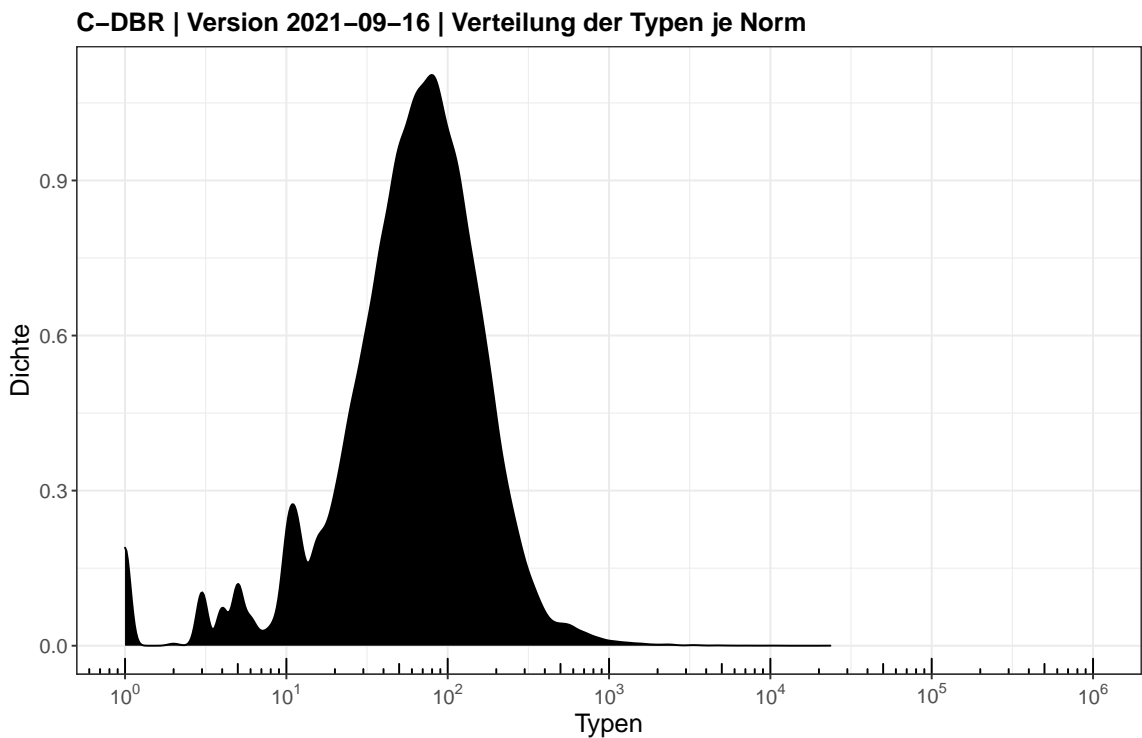
C-DBR | Version 2021-09-16 | Verteilung der Tokens je Rechtsakt



DOI: 10.5281/zenodo.5510458 | 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)
  )
)
```

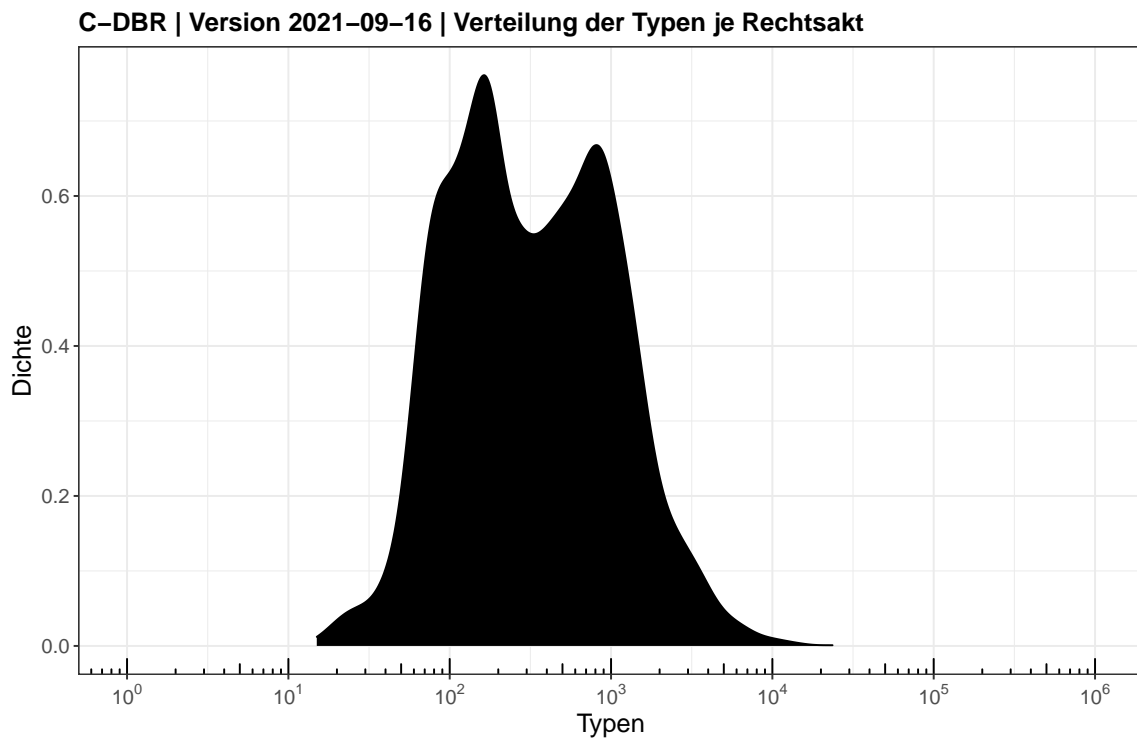


DOI: 10.5281/zenodo.5510458 | S. Fobbe

```

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)
  )

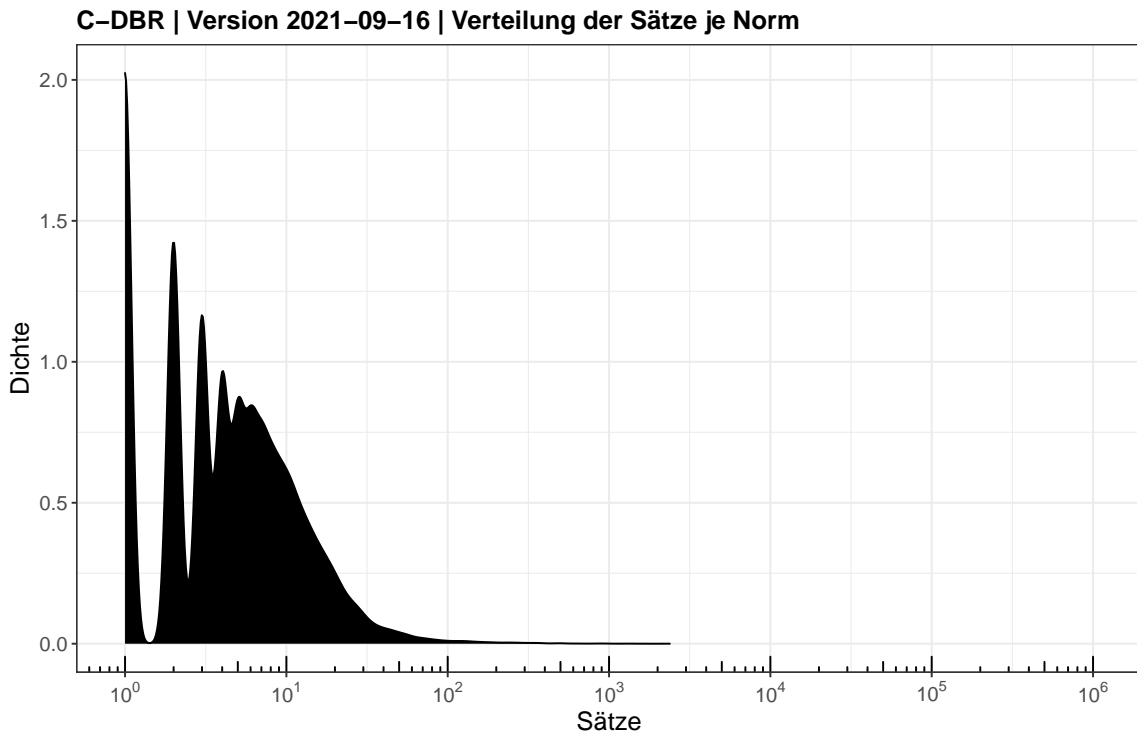
```



DOI: 10.5281/zenodo.5510458 | 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)
  )
)
```

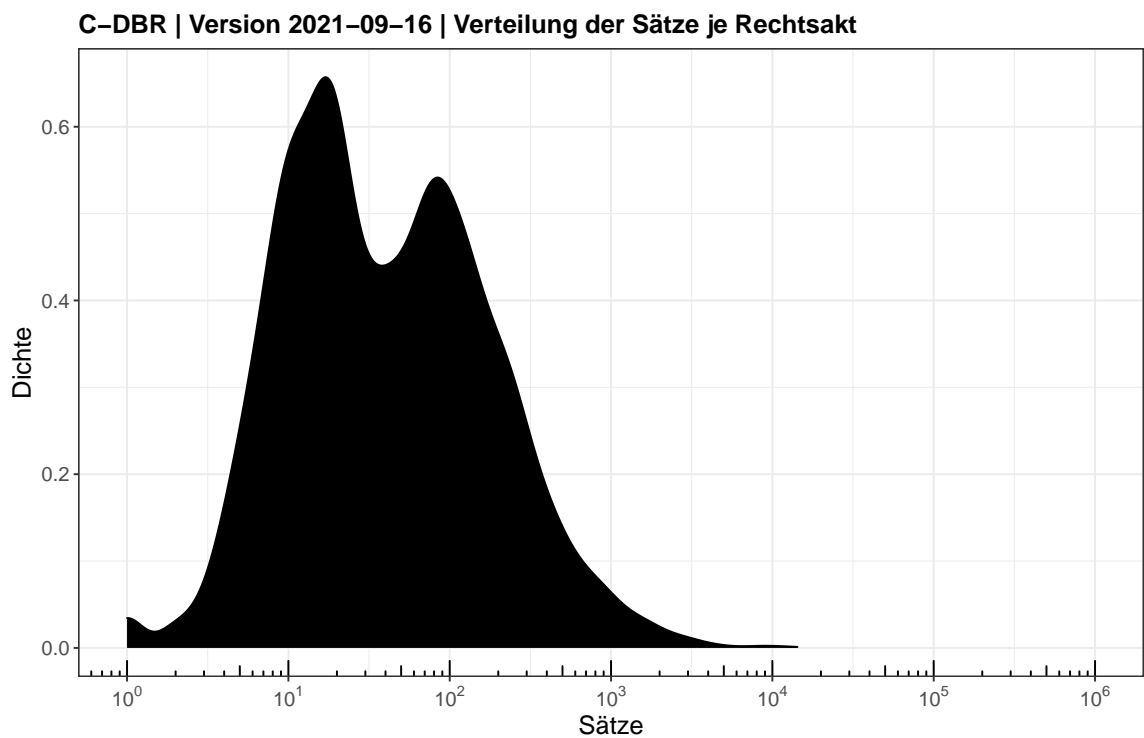


DOI: 10.5281/zenodo.5510458 | S. Fobbe

```

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.5510458 | 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" "1972-07-07" "1997-07-28" "1987-11-28" "2011-07-28" "2021-09-10"
```

Rechtsakte

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

```
##           Min.          1st Qu.          Median          Mean          3rd Qu.          Max.
## "1869-06-21" "1981-08-07" "2000-09-09" "1994-10-29" "2011-12-21" "2021-09-10"
```

XML-Metadaten

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

```
##           Min.          1st Qu.          Median          Mean          3rd Qu.          Max.
## "1869-06-21" "1977-12-20" "1997-12-29" "1993-01-09" "2010-07-21" "2021-09-10"
##           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",  
              "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",  
              "builddate_original",  
              "builddate_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",  
    "builddate_original",  
    "builddate_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"))
```

```
setcolorder(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",  
    "bulldate_original",  
    "bulldate_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" "bezketten"  
## [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] "builddate_original"    "builddate_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] 6638
```

14.2.2 Anzahl heruntergeladener Dateien

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

```
## [1] 6638
```

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] 6638
```

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] 55087
```

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] 6638
```

16.2.2 Anzahl heruntergeladener Dateien

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

```
## [1] 6638
```

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] 629.5923
```

17.1.2 EPUB-Dateien (MB)

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

```
## [1] 435.0296
```

17.1.3 XML-Dateien (MB)

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

```
## [1] 263.9377
```

17.1.4 TXT-Dateien (MB)

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

```
## [1] 197.2045
```

17.1.5 Korpus-Objekte in RAM (MB)

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

```
## 192.3 MB
```

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

```
## 147.9 MB
```

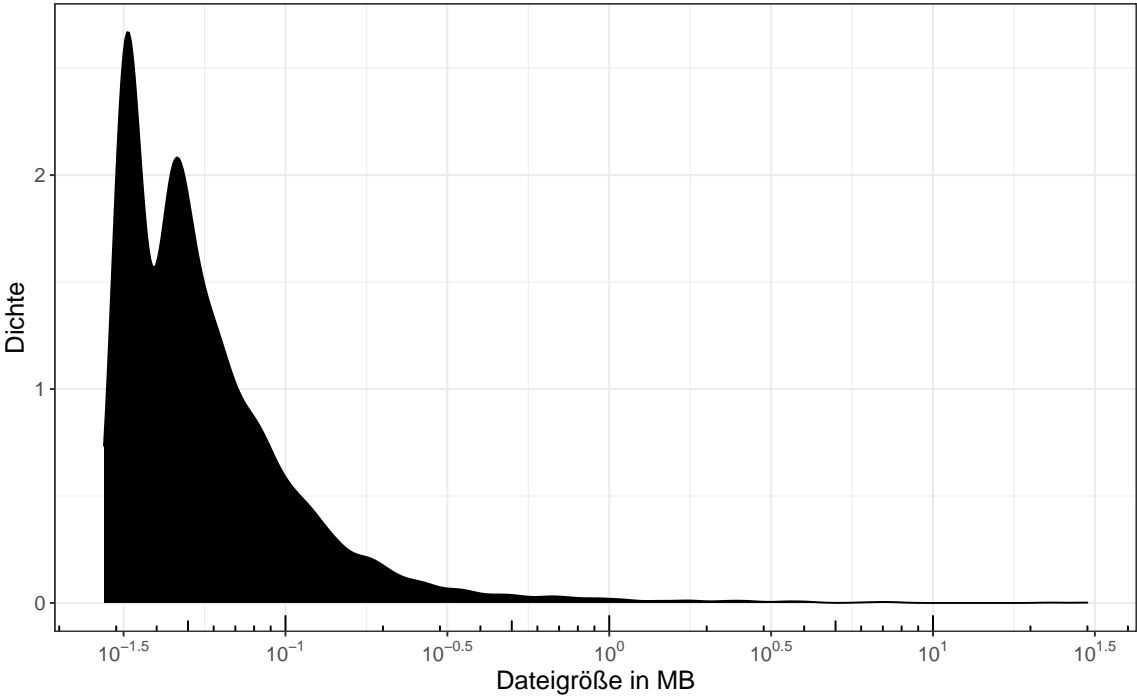
```
print(object.size(dt.meta),  
      standard = "SI",  
      humanReadable = TRUE,  
      units = "MB")
```

```
## 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)
  )
```

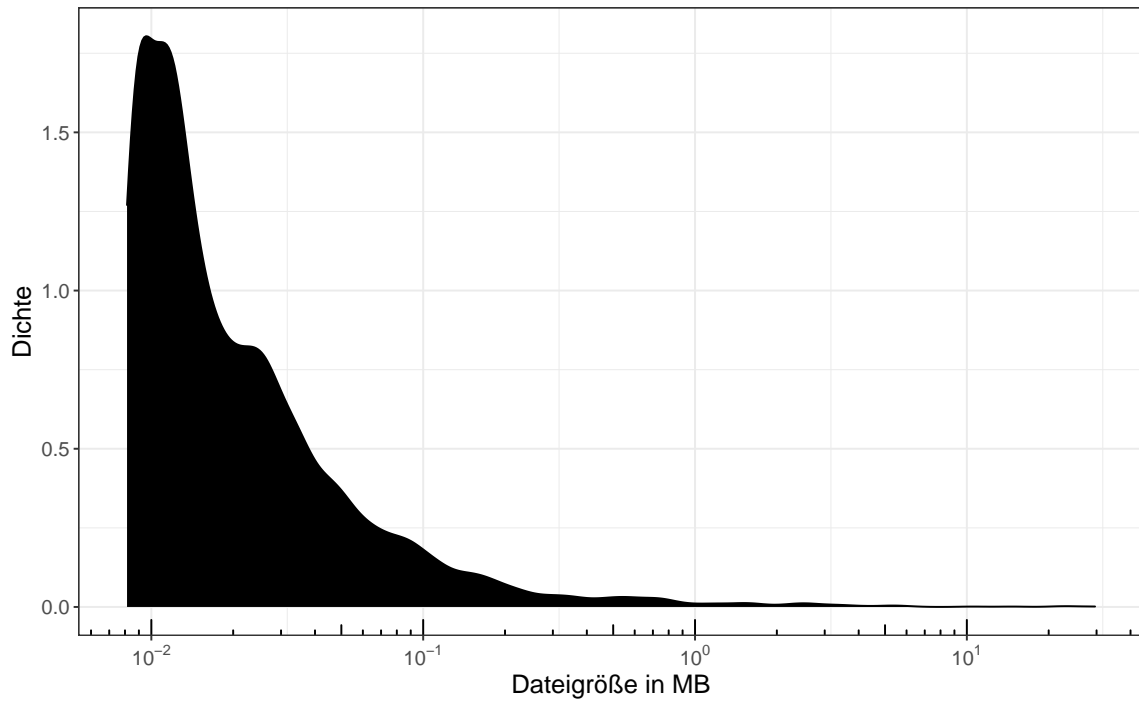
DOI: 10.5281/zenodo.5510458 | S. Fobbe

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(datasetname,
      "| 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 2021-09-16 | Verteilung der Dateigrößen (EPUB)



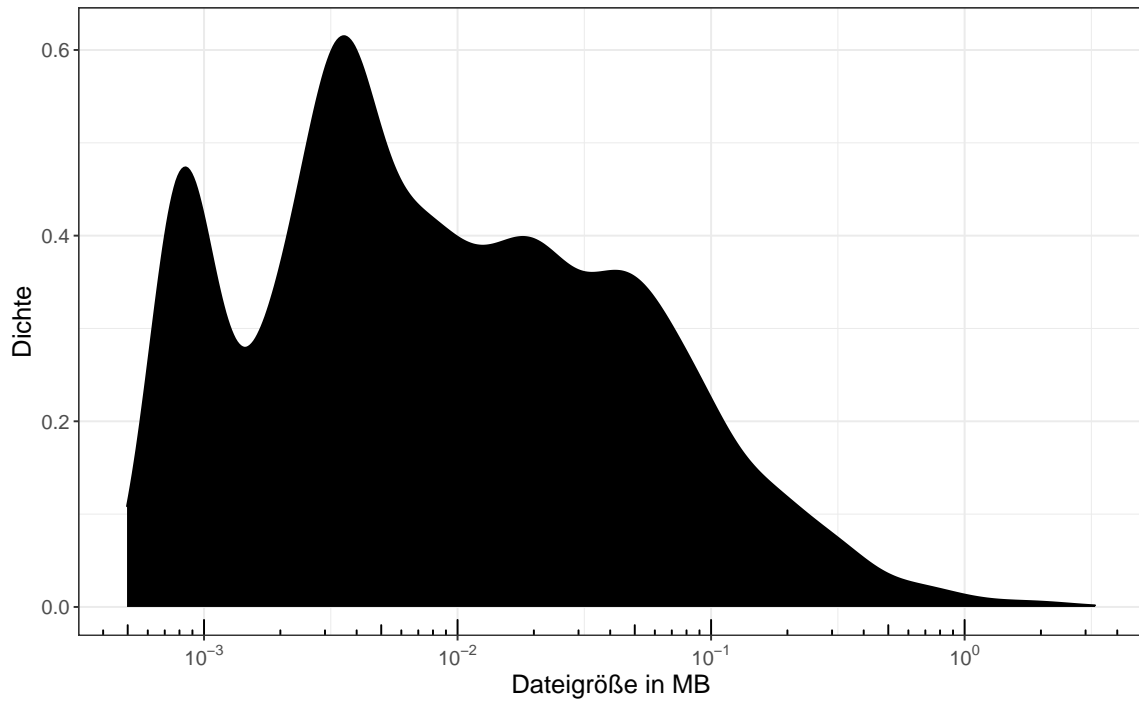
DOI: 10.5281/zenodo.5510458 | S. Fobbe

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 2021-09-16 | Verteilung der Dateigrößen (XML)



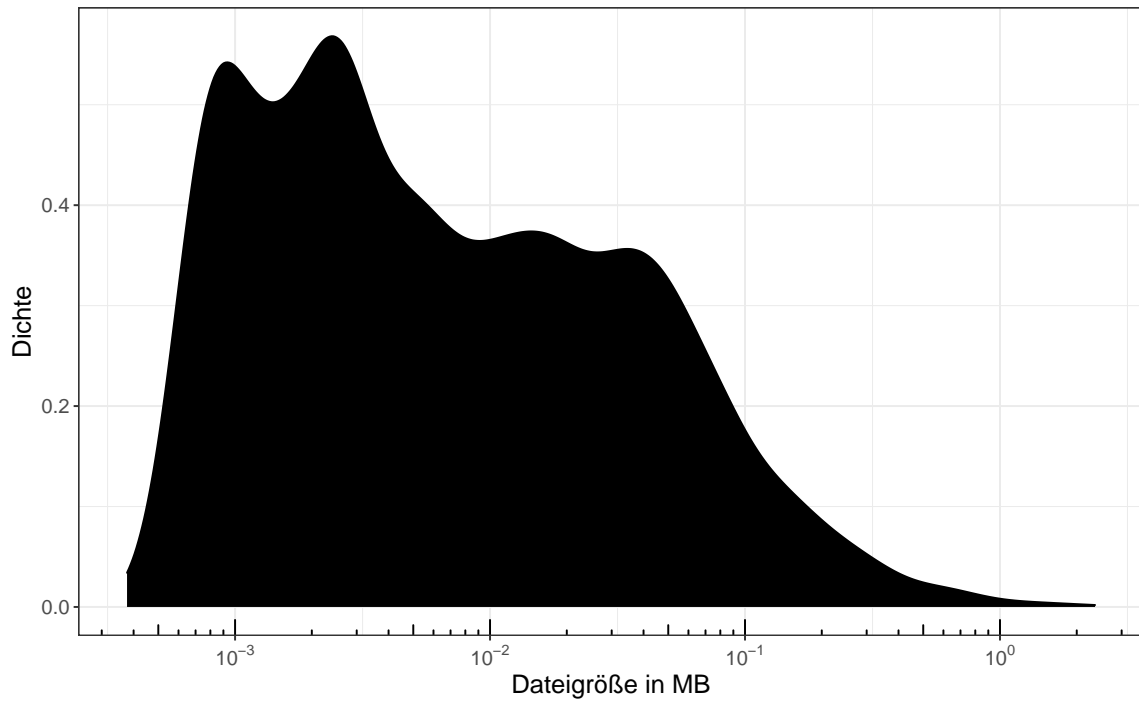
DOI: 10.5281/zenodo.5510458 | S. Fobbe

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 2021-09-16 | Verteilung der Dateigrößen (TXT)



DOI: 10.5281/zenodo.5510458 | 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 3.42 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_2021-09-16_DE_ANALYSE.zip
2	C-DBR_2021-09-16_DE_CSV_Einzelnormen_Datensatz.zip
3	C-DBR_2021-09-16_DE_CSV_Einzelnormen_Metadaten.zip
4	C-DBR_2021-09-16_DE_CSV_MetadatenXML.zip
5	C-DBR_2021-09-16_DE_CSV_Rechtsakte_Datensatz.zip
6	C-DBR_2021-09-16_DE_CSV_Rechtsakte_Metadaten.zip
7	C-DBR_2021-09-16_DE_EPUB_Datensatz.zip
8	C-DBR_2021-09-16_DE_Netzwerke.zip
9	C-DBR_2021-09-16_DE_PDF_Datensatz.zip
10	C-DBR_2021-09-16_DE_TXT_Datensatz.zip
11	C-DBR_2021-09-16_DE_XML_Anlagen.zip
12	C-DBR_2021-09-16_DE_XML_Datensatz.zip
13	C-DBR_2021-09-16_Source_Code.zip

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

index	sha2.256
1	ae87c561e650bb612617e5c0c63149e7931b6862426b0bfd5670a5079e705ab
2	43e1104dd11d61711ba29db3d882e15bd7360f49b573086f9d19b03040ce12a9
3	598b6e05ee32994050dbb0966f24f95d780e5e89bdd2ac65f1412491cc2c2290
4	8fb7200a87d1d96086f93cdb5aa17fd01fc377ff2591b99b68f10f17459e9af4
5	fa97ff3e9bd5d23fa4a31b46f846c375c4f160a144db5a83fe773ab395ce30ac
6	74b9edb35b42403be2e4aa44d25c78674a6113ff450100e62317ac7c79a7cb38
7	12aa4d985ddbba767ae3652a9890f819d094aacdc46c622a2a37a0c6356f50157
8	e2fc50f9da9daf2d37955b5b2d9f352238fa6214a4586f71973b2ed073bb7270
9	a94b6769f8bd5bd13e85f86e4cbd1bf206339b2c8ac672b33ac71903e4f3d931
10	ad79d287dd8066fb8ecf70279592a3eb267c0c30c46c24db14169cc7417df978
11	b2c836ac1359f39c454399600225f333fbb4d00219a14f0bbd3736e107eb91c4
12	bc5d051d45841ac08ed7b7731f12420fb68732df6a076936c9279791a9e7d1de
13	35e22e23840cb04c2fc7252aa6042536250b77a957c109f14398ed1891c6433b

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

index	sha3.512
1	fd916a60fcf5e59364359bfaea49672c50faf3fd45b1ac08442db8ebd4fc30fd9aaa904f7e7185e7e033ab2067ceee28ed8966bb6ef3ca836c76b48e6b0c1f
2	e70763e4664f6d6f2fa530050372e4017394f80da83027dbc8ff16c6dd2eb3e0061999224e66ac7480db2d0b632baf1bee1b4a148f3f7310e4a809f2e83d195c
3	36f542c12e0573cd72750eb1a49f6e5a9531b9a286040e476e1defba8559791fb3425a310b5266c4345718034345877b715bf3a13a3cebe2332887b57afd5f1d
4	d1f9b27d3287c1ff640c20828a62f22b8ba25c36ef52371a99a334f705a212501705f52c9f7f5e46290f1fa4e24cfe8f22c2d54c0d3017cd49f7b801205b813e
5	a2322861b8191cc52dc8e8c7571aacbd36c345379ba303b7f5bc8e4a0164ae50f5bd959e4e03d568cfe997e60c8dbc4e02e4a1e020565659302ae9bd57b709ba
6	9d30bf9482dd9e13a376f98b4ee2d5ce2b695f43e6fa5f0e9632c5ef5a84ad186172290abbfd7621eeffb3218c5b77de275043cfa6252e8dfe01b442012112fc
7	0a0906cb5ba74210c28794a672e7602908ce8c89d625918c44eea460dee6817dbb7d8240f6382488a950c089560101dd4065c1f8b501868fdc61177e4c1316b6
8	a4b0bd36d1bb5eb4ccfd7ebcf300cec598787c5b82115c6dd124da8c2fde886c78ba6f28baaed8c3556ad32677ae0067e1535b57651185cff5cc750cfe9ba87
9	d6c5c4c89d4a48423d4276081fba267f2203218a22ee8fa8c854afe54615cbea7aeab8c1be35c24dd90ce23486e54132572719fecdd2f879979deddbd1a0b1e
10	4203a30b690eb757ea5efbecf276bbec32ad560ed4f788395d7b55682025ccca f75f9a4df67a7c06d089f81feb4bfd5622d60270f04614b6b5612de973f0404f
11	04f694b32fcadd80ee86d4aaaf74105a7795b1c353923c98c0b6737ae0d77e5c53007d7b32caaaef5e325e2e1230007bb60e6c7503a0e39266745bf3e77c112c
12	eb74f982a2b9a9c8417ee298b34102e19713d4c451447236a83210a550cc13c4a0e3e7a92b7cc6a37f595a1681d5cc4c135f2867dd3911688111980eab8a8928
13	6a28991e5d544b8d8cb6fcaff908c29b962404bce84d13b97733361637201be76738fd62693cba90da0c37bd9a9632f46f33828199beb86967e92cdde6c52c5a

20 Abschluss

20.1 Datumsstempel

```
print(datestamp)
```

```
## [1] "2021-09-16"
```

20.2 Datum und Uhrzeit (Anfang)

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

```
## [1] "2021-09-16 00:08:07 CEST"
```

20.3 Datum und Uhrzeit (Ende)

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

```
## [1] "2021-09-16 00:37:16 CEST"
```

20.4 Laufzeit des gesamten Skripts

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

```
## Time difference of 29.1566 mins
```

20.5 Warnungen

```
warnings()
```

21 Parameter für strenge Replikationen

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

```
## [1] "OpenSSL 1.1.1l FIPS 24 Aug 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_1.4.5
##  [5] scales_1.1.1      quanteda_3.1.0    data.table_1.14.0 ggplot2_3.3.5
##  [9] doParallel_1.0.16 iterators_1.0.13 foreach_1.5.1     pdftools_3.0.1
## [13] kableExtra_1.3.4 knitr_1.34         xml2_1.3.2        rvest_1.0.1
##
## loaded via a namespace (and not attached):
##  [1] nlme_3.1-153      webshot_0.5.2      RColorBrewer_1.1-2
##  [4] httr_1.4.2        tools_4.0.5        backports_1.2.1
##  [7] utf8_1.2.2        R6_2.5.1           rpart_4.1-15
## [10] Hmisc_4.5-0       colorspace_2.0-2   nnet_7.3-16
## [13] withr_2.4.2       mnormt_2.0.2       tidyselect_1.1.1
## [16] gridExtra_2.3     curl_4.3.2         compiler_4.0.5
## [19] fdrtool_1.2.16    htmlTable_2.2.1    labeling_0.4.2
## [22] checkmate_2.0.0   psych_2.1.6        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_2.10    svglite_2.0.0      base64enc_0.1-3
## [34] jpeg_0.1-9        pkgconfig_2.0.3    htmltools_0.5.2
## [37] fastmap_1.1.0     highr_0.9          htmlwidgets_1.5.4
## [40] rlang_0.4.11      rstudioapi_0.13    farver_2.1.0
## [43] generics_0.1.0    gtools_3.9.2       dplyr_1.0.7
```

```

## [46] magrittr_2.0.1      Formula_1.2-4      Matrix_1.3-4
## [49] Rcpp_1.0.7          munsell_0.5.0      fansi_0.5.0
## [52] abind_1.4-5         viridis_0.6.1      lifecycle_1.0.0
## [55] stringi_1.7.4       yaml_2.2.1         MASS_7.3-54
## [58] plyr_1.8.6          lavaan_0.6-9       grid_4.0.5
## [61] ggrepel_0.9.1       crayon_1.4.1       lattice_0.20-44
## [64] graphlayouts_0.7.1 splines_4.0.5      magick_2.7.3
## [67] tmvnsim_1.0-2       pillar_1.6.2       corpcor_1.6.9
## [70] stats4_4.0.5        reshape2_1.4.4     codetools_0.2-18
## [73] stopwords_2.2        fastmatch_1.1-3    glue_1.4.2
## [76] evaluate_0.14       qpdf_1.1           latticeExtra_0.6-29
## [79] RcppParallel_5.1.4 selectr_0.4-2       png_0.1-7
## [82] vctrs_0.3.8         tweenr_1.0.2       gtable_0.3.0
## [85] purrr_0.3.4         polyclip_1.10-0    tidyr_1.1.3
## [88] xfun_0.25           ggforce_0.3.3      tidygraph_1.2.0
## [91] glasso_1.11         survival_3.2-13    viridisLite_0.4.0
## [94] tibble_3.1.4        cluster_2.1.2      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. 2021. *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>.