

# Corpus des Deutschen Bundesrechts (C-DBR)

COMPILATION REPORT

Version 2022-01-12

License MIT-0

DOI: [10.5281/zenodo.5832095](https://doi.org/10.5281/zenodo.5832095)

<b>Titel</b>	Source Code des »Corpus des Deutschen Bundesrechts«
<b>Abkürzung</b>	C-DBR-Source
<b>Autor</b>	Seán Fobbe
<b>Version</b>	2022-01-12
<b>Download</b>	<a href="https://doi.org/10.5281/zenodo.5832095">https://doi.org/10.5281/zenodo.5832095</a>
<b>Lizenz</b>	MIT No Attribution (MIT-0)

### Zitiervorschlag

*Seán Fobbe* (2022). Source Code des »Corpus des Deutschen Bundesrechts« (C-DBR-Source). Version 2022-01-12. Zenodo. DOI: 10.5281/zenodo.5832095.

### 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 2022-01-12. 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 — 2022 — Seán Fobbe

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

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

### Disclaimer

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

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>9</b>
1.1	Überblick . . . . .	9
1.2	Endprodukte . . . . .	9
1.3	Kompilierung . . . . .	10
1.4	Systemanforderungen . . . . .	10
<b>2</b>	<b>Vorbereitung</b>	<b>11</b>
2.1	Datumsstempel . . . . .	11
2.2	Datum und Uhrzeit (Beginn) . . . . .	11
2.3	Packages Laden . . . . .	11
2.4	Zusätzliche Funktionen einlesen . . . . .	12
2.5	Verzeichnis für Analyse-Ergebnisse und Diagramme definieren . . . . .	12
2.6	Weitere Verzeichnisse definieren . . . . .	12
2.7	Dateien aus vorherigen Runs bereinigen . . . . .	12
2.8	Verzeichnisse anlegen . . . . .	13
2.9	Vollzitate statistischer Software schreiben . . . . .	13
2.10	Allgemeine Konfiguration . . . . .	13
2.10.1	Konfiguration einlesen . . . . .	13
2.10.2	Konfiguration anzeigen . . . . .	13
2.10.3	Knitr Optionen setzen . . . . .	14
2.10.4	Download Timeout setzen . . . . .	14
2.10.5	Quellenangabe für Diagramme definieren . . . . .	15
2.10.6	Präfix für Dateien definieren . . . . .	15
2.10.7	Präfix für Diagramme definieren . . . . .	15
2.10.8	Quanteda-Optionen setzen . . . . .	15
2.11	LaTeX Konfiguration . . . . .	15
2.11.1	LaTeX Parameter definieren . . . . .	15
2.11.2	LaTeX Parameter schreiben . . . . .	16
2.12	Parallelisierung aktivieren . . . . .	17
2.12.1	Anzahl logischer Kerne festlegen . . . . .	17
2.12.2	Quanteda . . . . .	17
2.12.3	Data.table . . . . .	17
<b>3</b>	<b>Download vorbereiten</b>	<b>18</b>
3.1	XML-Inhaltsverzeichnis einlesen . . . . .	18
3.2	Links zu XML-Dateien aus XML-Inhaltsverzeichnis extrahieren . . . . .	18
3.3	Debugging-Modus: Ausgewertete Links reduzieren . . . . .	18
3.4	Links zu HTML Landing Pages generieren . . . . .	18
3.5	Funktion anzeigen: f.linkextract . . . . .	18
3.6	Links aus HTML Landing Pages extrahieren . . . . .	19
3.7	Dateinamen von PDF und EPUB-Dateien in separate Vektoren sortieren . . . . .	19
3.8	Vektor der Langtitel erstellen . . . . .	19
3.8.1	Namen bereinigen und kürzen . . . . .	19
3.8.2	Indizes der AEG bestimmen . . . . .	20
3.8.3	AEGs umbenennen . . . . .	20
3.9	Vektor der Kurztitel erstellen . . . . .	20
3.10	Vektoren der Titel vereinigen . . . . .	20

3.11	Prüfung auf Namens-Kollisionen . . . . .	20
3.12	Bereinigung von Namens-Kollisionen . . . . .	21
3.13	Dateierweiterungen hinzufügen . . . . .	21
3.14	Links zu EPUB-Dateien erstellen . . . . .	21
3.15	Links zu PDF-Dateien erstellen . . . . .	21
3.16	Data Table für Download vorbereiten . . . . .	21
3.17	Abkürzungsverzeichnis erstellen . . . . .	22
3.18	Download Table als CSV speichern . . . . .	22
3.19	Verzeichnis aller Rechtsakte als CSV speichern . . . . .	22
3.20	Debugging-Modus: Anzahl der heruntergeladenen Dateien reduzieren . . . . .	22
3.21	Anzahl herunterzuladender Dateien . . . . .	22
3.21.1	Pro Format . . . . .	22
3.21.2	Insgesamt . . . . .	23
<b>4</b>	<b>Verarbeitung der DTD und XML-Dateien mit Anlagen</b>	<b>24</b>
4.1	Document Type Definition herunterladen . . . . .	24
4.2	Download der XML-Dateien . . . . .	24
4.3	Download-Ergebnis . . . . .	24
4.3.1	Anzahl herunterzuladender Dateien . . . . .	24
4.3.2	Anzahl heruntergeladener Dateien . . . . .	24
4.3.3	Fehlbetrag . . . . .	25
4.3.4	Fehlende Dateien . . . . .	25
4.4	Extrahieren der XML-Dateien und ihrer Anlagen . . . . .	25
4.5	XML Dateien auflisten und Dateigrößen speichern . . . . .	25
4.6	Korpus erstellen: Einzelnormen . . . . .	26
4.6.1	Beginn XML Parsing . . . . .	29
4.6.2	Parallelisierung definieren . . . . .	30
4.6.3	XML Parsen . . . . .	30
4.6.4	Liste in Data Table umwandeln . . . . .	30
4.6.5	Ende XML Parsing . . . . .	30
4.6.6	Dauer XML Parsing . . . . .	30
4.6.7	Variable “doc_id” erstellen . . . . .	30
4.6.8	Variablen-Name für Ausfertigungsdatum anpassen . . . . .	31
4.6.9	Variable “fundstellentyp” anpassen . . . . .	31
4.6.10	Variable “builddate_iso” erstellen . . . . .	31
4.6.11	Variable “aenderung_datum” erstellen . . . . .	31
4.6.12	Variable “aufhebung_verkuendung_datum” erstellen . . . . .	31
4.6.13	Variable “aufhebung_wirkung_datum” erstellen . . . . .	31
4.6.14	Variable “neufassung_datum” erstellen . . . . .	32
4.6.15	Variable “ausfertigung_jahr” hinzufügen . . . . .	32
4.6.16	Variable “doi_concept” hinzufügen . . . . .	32
4.6.17	Variable “doi_version” hinzufügen . . . . .	32
4.6.18	Variable “version” hinzufügen . . . . .	32
4.6.19	Variable “lizenz” hinzufügen . . . . .	32
4.7	Stichprobe für Qualitätsprüfung ziehen . . . . .	33
4.8	Korpus erstellen: Rechtsakte . . . . .	34
4.8.1	Variablen definieren . . . . .	34
4.8.2	Vollständiger Satz an Variablen . . . . .	34
4.8.3	Einzelnormen zu Rechtsakten vereinigen . . . . .	34

4.8.4	Variable “dateiname” in “doc_id” umbenennen . . . . .	35
4.9	Datensatz erstellen: XML-Metadaten . . . . .	36
4.9.1	Funktion für XML-Parsing definieren . . . . .	36
4.9.2	Beginn XML Parsing . . . . .	38
4.9.3	Parallelisierung definieren . . . . .	38
4.9.4	XML Parsen . . . . .	38
4.9.5	Liste in Data Table umwandeln . . . . .	38
4.9.6	Ende XML Parsing . . . . .	38
4.9.7	Dauer XML Parsing . . . . .	38
4.9.8	Variablen-Name für Ausfertigungsdatum anpassen . . . . .	39
4.9.9	Variable “fundstellentyp” anpassen . . . . .	39
4.9.10	Variable “builddate_iso” erstellen . . . . .	39
4.9.11	Variable “aenderung_datum” erstellen . . . . .	39
4.9.12	Variable “aufhebung_verkuendung_datum” erstellen . . . . .	39
4.9.13	Variable “aufhebung_wirkung_datum” erstellen . . . . .	39
4.9.14	Variable “neufassung_datum” erstellen . . . . .	40
4.9.15	Variable “ausfertigung_jahr” hinzufügen . . . . .	40
4.9.16	Variable “doi_concept” hinzufügen . . . . .	40
4.9.17	Variable “doi_version” hinzufügen . . . . .	40
4.9.18	Variable “version” hinzufügen . . . . .	40
4.9.19	Variable “lizenz” hinzufügen . . . . .	40
4.10	Netzwerk-Analyse (experimentell!) . . . . .	41
4.10.1	Funktion definieren: f.kennzahlen.search . . . . .	41
4.10.2	Funktion definieren: f.kennzahlen.collapse . . . . .	41
4.10.3	Funktion definieren: f.kennzahlen.edgelist . . . . .	41
4.10.4	Funktion definieren: f.network.analysis . . . . .	43
4.10.5	Netzwerk-Analyse durchführen . . . . .	46
4.10.6	Beginn Network Analysis . . . . .	47
4.10.7	Parallelisierung definieren . . . . .	47
4.10.8	XML Parsen . . . . .	47
4.10.9	XML-Dateien bei denen Fehler auftreten . . . . .	47
4.10.10	Ende XML Parsing . . . . .	48
4.10.11	Dauer XML Parsing . . . . .	48
4.11	Wiederverpacken der XML-Dateien . . . . .	48
4.11.1	XML-Dateien definieren . . . . .	48
4.11.2	XML-Dateien verpacken . . . . .	48
4.11.3	Anhänge zu XML-Dateien verpacken . . . . .	48
<b>5</b>	<b>Frequenztabellen erstellen: Einzelnormen</b>	<b>50</b>
5.1	Funktion anzeigen: f.fast.freqtable . . . . .	50
5.2	Liste zu prüfender Variablen . . . . .	51
5.3	Frequenztabellen erstellen . . . . .	52
<b>6</b>	<b>Frequenztabellen erstellen: Rechtsakte</b>	<b>112</b>
6.1	Variablen ignorieren . . . . .	112
6.2	Liste zu prüfender Variablen . . . . .	112
6.3	Frequenztabellen erstellen . . . . .	112
<b>7</b>	<b>Frequenztabellen erstellen: XML-Metadaten</b>	<b>121</b>
7.1	Liste zu prüfender Variablen . . . . .	121

7.2	Frequenztabellen erstellen . . . . .	121
<b>8</b>	<b>Frequenztabellen visualisieren</b>	<b>131</b>
8.1	Präfixe erstellen . . . . .	131
8.2	Tabellen für Einzelnormen einlesen . . . . .	131
8.3	Tabellen für Rechtsakte einlesen . . . . .	131
8.4	Tabellen für XML-Metadaten einlesen . . . . .	131
8.5	Periodikum . . . . .	132
8.5.1	Einzelnormen . . . . .	132
8.5.2	Rechtsakte . . . . .	134
8.5.3	XML-Metadaten . . . . .	136
8.6	Ausfertigungsjahr . . . . .	138
8.6.1	Einzelnormen . . . . .	138
8.6.2	Rechtsakte . . . . .	139
8.6.3	XML-Metadaten . . . . .	140
<b>9</b>	<b>Korpus-Analytik</b>	<b>141</b>
9.1	Berechnung linguistischer Kennwerte . . . . .	141
9.1.1	Funktion anzeigen: future_lingsummarize . . . . .	141
9.1.2	Berechnung durchführen . . . . .	142
9.2	Variablen-Namen anpassen . . . . .	143
9.2.1	Einzelnormen . . . . .	143
9.2.2	Rechtsakte . . . . .	143
9.3	Kennwerte den Korpora hinzufügen . . . . .	143
9.3.1	Einzelnormen . . . . .	143
9.3.2	Rechtsakte . . . . .	143
9.4	Varianten mit Metadaten erstellen . . . . .	144
9.4.1	Einzelnormen . . . . .	144
9.4.2	Rechtsakte . . . . .	144
9.5	Linguistische Kennwerte: Einzelnormen . . . . .	145
9.5.1	Zusammenfassungen berechnen . . . . .	145
9.5.2	Zusammenfassungen anzeigen . . . . .	147
9.5.3	Zusammenfassungen speichern . . . . .	147
9.6	Linguistische Kennwerte: Rechtsakte . . . . .	148
9.6.1	Zusammenfassungen berechnen . . . . .	148
9.6.2	Zusammenfassungen anzeigen . . . . .	150
9.6.3	Zusammenfassungen speichern . . . . .	150
9.7	Verteilungen . . . . .	151
9.7.1	Density (Zeichen) . . . . .	151
9.7.2	Density (Tokens) . . . . .	153
9.7.3	Density (Typen) . . . . .	155
9.7.4	Density (Sätze) . . . . .	157
9.8	Quantitative Variablen . . . . .	158
9.8.1	Ausfertigungsdatum . . . . .	158
9.8.2	Ausfertigungsjahr . . . . .	159
<b>10</b>	<b>Strenge Kontrolle der Variablen-Namen</b>	<b>161</b>
10.1	Semantische Sortierung der Variablen . . . . .	161
10.1.1	Variablen sortieren: Einzelnormen . . . . .	161
10.1.2	Variablen sortieren: Rechtsakte . . . . .	163

10.1.3 Variablen sortieren: XML-Metadaten . . . . .	165
10.2 Anzahl Variablen der Datensätze . . . . .	165
10.3 Alle Variablen-Namen der Datensätze . . . . .	166
<b>11 CSV-Dateien erstellen</b>	<b>169</b>
11.1 Einzelnormen (Korpus) . . . . .	169
11.1.1 Name für CSV definieren . . . . .	169
11.1.2 Datensatz speichern . . . . .	169
11.2 Einzelnormen (Metadaten) . . . . .	169
11.2.1 Name für CSV definieren . . . . .	169
11.2.2 Datensatz speichern . . . . .	169
11.3 Rechtsakte (Korpus) . . . . .	169
11.3.1 Name für CSV definieren . . . . .	169
11.3.2 Datensatz speichern . . . . .	169
11.4 Rechtsakte (Metadaten) . . . . .	170
11.4.1 Name für CSV definieren . . . . .	170
11.4.2 Datensatz speichern . . . . .	170
11.5 XML-Metadaten . . . . .	170
11.5.1 Name für CSV definieren . . . . .	170
11.5.2 Datensatz speichern . . . . .	170
<b>12 Download der PDF-Dateien</b>	<b>171</b>
12.1 Download durchführen . . . . .	171
12.2 Download-Ergebnis . . . . .	171
12.2.1 Anzahl herunterzuladender Dateien . . . . .	171
12.2.2 Anzahl heruntergeladener Dateien . . . . .	171
12.2.3 Fehlbetrag . . . . .	171
12.2.4 Fehlende Dateien . . . . .	172
<b>13 TXT-Dateien erstellen</b>	<b>173</b>
13.1 Anzahl zu extrahierender Dateien . . . . .	173
13.2 Funktion anzeigen: <code>future_pdf_to_txt</code> . . . . .	173
13.3 Text Extrahieren . . . . .	174
<b>14 Download der EPUB-Dateien</b>	<b>175</b>
14.1 Download durchführen . . . . .	175
14.2 Download-Ergebnis . . . . .	175
14.2.1 Anzahl herunterzuladender Dateien . . . . .	175
14.2.2 Anzahl heruntergeladener Dateien . . . . .	175
14.2.3 Fehlbetrag . . . . .	175
14.2.4 Fehlende Dateien . . . . .	176
<b>15 Dateigrößen analysieren</b>	<b>177</b>
15.1 Gesamtgröße . . . . .	177
15.1.1 PDF-Dateien (MB) . . . . .	177
15.1.2 EPUB-Dateien (MB) . . . . .	177
15.1.3 XML-Dateien (MB) . . . . .	177
15.1.4 TXT-Dateien (MB) . . . . .	178
15.1.5 Objekte in RAM (MB) . . . . .	178
15.2 Verteilung der Dateigrößen (PDF) . . . . .	179

15.3	Verteilung der Dateigrößen (EPUB)	180
15.4	Verteilung der Dateigrößen (XML)	181
15.5	Verteilung der Dateigrößen (TXT)	182
<b>16</b>	<b>ZIP-Archive erstellen</b>	<b>183</b>
16.1	Verpacken der CSV-Dateien	183
16.2	Verpacken der PDF-Dateien	183
16.3	Verpacken der TXT-Dateien	183
16.4	Verpacken der EPUB-Dateien	183
16.5	Verpacken der Netzwerk-Dateien	184
16.6	Verpacken der Analyse-Dateien	184
16.7	Verpacken der Source-Dateien	184
<b>17</b>	<b>Roh-Dateien löschen</b>	<b>186</b>
<b>18</b>	<b>Kryptographische Hashes</b>	<b>187</b>
18.1	Liste der ZIP-Archive erstellen	187
18.2	Funktion anzeigen: future_multihashes	187
18.3	Hashes berechnen	188
18.4	In Data Table umwandeln	188
18.5	Index hinzufügen	188
18.6	Hashes in CSV-Datei speichern	189
18.7	Leerzeichen hinzufügen um Zeilenumbruch zu ermöglichen	189
18.8	In Bericht anzeigen	190
<b>19</b>	<b>Aufräumen</b>	<b>193</b>
<b>20</b>	<b>Abschluss</b>	<b>194</b>
20.1	Datumsstempel	194
20.2	Datum und Uhrzeit (Anfang)	194
20.3	Datum und Uhrzeit (Ende)	194
20.4	Laufzeit des gesamten Skripts	194
20.5	Warnungen	194
<b>21</b>	<b>Parameter für strenge Replikationen</b>	<b>197</b>
	<b>Literaturverzeichnis</b>	<b>199</b>



# 1 Einleitung

## 1.1 Überblick

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

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

## 1.2 Endprodukte

Primäre Endprodukte des Skripts (im Ordner ‘output’) 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.

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

```
source("00_C-DBR_FullCompile.R")
```

### 1.4 Systemanforderungen

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

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

Auf der Festplatte sollten 8 GB Speicherplatz vorhanden sein.

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

## 2 Vorbereitung

### 2.1 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] "2022-01-12"
```

### 2.2 Datum und Uhrzeit (Beginn)

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

```
## [1] "2022-01-12 13:50:58 CET"
```

### 2.3 Packages Laden

```
library("zip")           # ZIP Files
library("rvest")         # HTML/XML-Extraktion
library("xml2")          # Verarbeitung von XML-Format
library("RcppTOML")      # Verarbeitung von TOML-Format
library("knitr")         # Professionelles Reporting
library("kableExtra")    # Verbesserte Kable Tabellen
library("magick")        # Verarbeitung von Bild-Dateien
library("pdftools")      # Extrahieren von PDF-Dateien
library("parallel")      # Parallelisierung
##library("doParallel")  # Parallelisierung
library("ggplot2")       # Fortgeschrittene Datenvisualisierung
library("data.table")    # Fortgeschrittene Datenverarbeitung
library("quanteda")      # Fortgeschrittene Computerlinguistik
library("scales")        # Skalierung von Diagrammen
library("openssl")       # Kryptographische Signaturen
library("igraph")        # Analyse von Graphen
library("ggraph")        # Analyse von Graphen
##library("qgraph")      # Analyse von Graphen
library("future")        # Parallelisierung
library("future.apply")  # Parallelisierung von base-r-Funktionen
```

## 2.4 Zusätzliche Funktionen einlesen

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

```
source("R-fobbe-proto-package/f.linkextract.R")
source("R-fobbe-proto-package/f.fast.freqtable.R")
#source("R-fobbe-proto-package/f.lingsummarize.iterator.R") # deprecated
#source("R-fobbe-proto-package/f.dopar.pagenums.R") # deprecated
#source("R-fobbe-proto-package/f.dopar.pdfextract.R") # deprecated
#source("R-fobbe-proto-package/f.dopar.multihashes.R") # deprecated

source("functions/f.heading.transform.R")
source("functions/f.namechain.R")
source("functions/f.zero.NA.R")
source("functions/f.multihashes.R")
source("functions/f.future_multihashes.R")
source("functions/f.pdf_to_txt.R")
source("functions/f.future_pdf_to_txt.R")
source("functions/f.future_lingsummarize.R")
```

## 2.5 Verzeichnis für Analyse-Ergebnisse und Diagramme definieren

Muss mit einem Schrägstrich enden!

```
dir.analysis <- paste0(getwd(),
                      "/analyse/")
```

## 2.6 Weitere Verzeichnisse definieren

```
dirs <- c("output",
         "temp",
         "netzwerke")
```

## 2.7 Dateien aus vorherigen Runs bereinigen

```
unlink(dir.analysis, recursive = TRUE)

unlink(dirs, recursive = TRUE)

files.delete <- list.files(pattern = "\\\\.zip|\\.xml|\\.jpe?g|\\.png|\\.gif|\\.pdf|
  |\\.epub",
                          ignore.case = TRUE)

unlink(files.delete)
```

## 2.8 Verzeichnisse anlegen

```
dir.create(dir.analysis)

lapply(dirs, dir.create)
```

```
## [[1]]
## [1] TRUE
##
## [[2]]
## [1] TRUE
##
## [[3]]
## [1] TRUE
```

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

## 2.9 Vollzitate statistischer Software schreiben

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

## 2.10 Allgemeine Konfiguration

### 2.10.1 Konfiguration einlesen

```
config <- parseTOML("C-DBR_Config.toml")
```

### 2.10.2 Konfiguration anzeigen

```
print(config)
```

```
## List of 11
## $ cores      :List of 2
## ..$ max      : logi TRUE
## ..$ number: int 8
## $ debug      :List of 2
```

```

## ..$ sample: int 500
## ..$ toggle: logi FALSE
## $ doi      :List of 2
## ..$ data   :List of 2
## .. ..$ concept: chr "10.5281/zenodo.3832111"
## .. ..$ version: chr "10.5281/zenodo.5832094"
## ..$ software:List of 2
## .. ..$ concept: chr "10.5281/zenodo.4072934"
## .. ..$ version: chr "10.5281/zenodo.5832095"
## $ download :List of 1
## ..$ timeout: int 600
## $ fig      :List of 3
## ..$ align  : chr "center"
## ..$ dpi    : int 300
## ..$ format: chr [1:2] "pdf" "png"
## $ freqtable:List of 1
## ..$ ignore: chr [1:13] "periodikum" "fundstellentyp" "check_neuf" "check_
aufh" ...
## $ license  :List of 2
## ..$ code: chr "MIT-0"
## ..$ data: chr "Creative Commons Zero 1.0 Universal"
## $ parallel :List of 10
## ..$ downloadEPUB      : logi TRUE
## ..$ downloadPDF       : logi TRUE
## ..$ downloadXML       : logi TRUE
## ..$ extractPDF        : logi TRUE
## ..$ htmlLandingPages  : logi TRUE
## ..$ lingsummarize     : logi TRUE
## ..$ multihashes       : logi TRUE
## ..$ parseEinzelnormen: logi FALSE
## ..$ parseMeta         : logi FALSE
## ..$ parseNetworks     : logi FALSE
## $ project  :List of 3
## ..$ author   : chr "Seán Fobbe"
## ..$ fullname : chr "Corpus des Deutschen Bundesrechts"
## ..$ shortname: chr "C-DBR"
## $ qa       :List of 1
## ..$ sample: int 50
## $ quanteda :List of 1
## ..$ tokens_locale: chr "de_DE"

```

### 2.10.3 Knitr Optionen setzen

```

knitr::opts_chunk$set(fig.path = dir.analysis,
                      dev = config$fig$format,
                      dpi = config$fig$dpi,
                      fig.align = config$fig$align)

```

### 2.10.4 Download Timeout setzen

```

options(timeout = config$download$timeout)

```

## 2.10.5 Quellenangabe für Diagramme definieren

```
caption <- paste("Fobbe | DOI:",
                 config$doi$data$version)
print(caption)
```

```
## [1] "Fobbe | DOI: 10.5281/zenodo.5832094"
```

## 2.10.6 Präfix für Dateien definieren

```
prefix.files <- paste0(config$project$shortname,
                       "_",
                       datestamp)
print(prefix.files)
```

```
## [1] "C-DBR_2022-01-12"
```

## 2.10.7 Präfix für Diagramme definieren

```
prefix.figuretitle <- paste(config$project$shortname,
                             "| Version",
                             datestamp)
```

## 2.10.8 Quanteda-Optionen setzen

```
quanteda_options(tokens_locale = config$quanteda$tokens_locale)
```

## 2.11 LaTeX Konfiguration

### 2.11.1 LaTeX Parameter definieren

```
latexdefs <- c("%===== \n% Definitionen \n
               %=====",
               "\n% NOTE: Diese Datei wurde während des Kompilierungs-Prozesses
               automatisch erstellt.\n",
               "\n%-----Autor-----",
               paste0("\n\\newcommand{\n\\projectauthor}{",
                       config$project$author,
                       "}"),
               "\n%-----Version-----",
               paste0("\n\\newcommand{\n\\version}{",
```

```

        datestamp,
        "}"),
    "\n%-----Titles-----",
    paste0("\newcommand{\datatitle}{",
           config$project$fullname,
           "}"),
    paste0("\newcommand{\datashort}{",
           config$project$shortname,
           "}"),
    paste0("\newcommand{\softwaretitle}{Source Code des \enquote{",
           config$project$fullname,
           "}}"),
    paste0("\newcommand{\softwareshort}{",
           config$project$shortname,
           "-Source}"),
    "\n%-----Data DOIs-----",
    paste0("\newcommand{\dataconceptdoi}{",
           config$doi$data$concept,
           "}"),
    paste0("\newcommand{\dataversiondoi}{",
           config$doi$data$version,
           "}"),
    paste0("\newcommand{\dataconcepturldoi}{https://doi.org/",
           config$doi$data$concept,
           "}"),
    paste0("\newcommand{\dataversionurldoi}{https://doi.org/",
           config$doi$data$version,
           "}"),
    "\n%-----Software DOIs-----",
    paste0("\newcommand{\softwareconceptdoi}{",
           config$doi$software$concept,
           "}"),
    paste0("\newcommand{\softwareversiondoi}{",
           config$doi$software$version,
           "}"),
    paste0("\newcommand{\softwareconcepturldoi}{https://doi.org/",
           config$doi$software$concept,
           "}"),
    paste0("\newcommand{\softwareversionurldoi}{https://doi.org/",
           config$doi$software$version,
           "}")

```

## 2.11.2 LaTeX Parameter schreiben

```

writeLines(latexdefs,
           paste0("temp/",
                  config$project$shortname,
                  "_Definitions.tex"))

```



## 2.12 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.

### 2.12.1 Anzahl logischer Kerne festlegen

```
if (config$cores$max == TRUE){
  fullCores <- detectCores()
}

if (config$cores$max == FALSE){
  fullCores <- as.integer(config$cores$number)
}

print(fullCores)
```

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

### 2.12.2 Quanteda

```
quanteda_options(threads = 1)
```

### 2.12.3 Data.table

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

## 3 Download vorbereiten

### 3.1 XML-Inhaltsverzeichnis einlesen

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

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

```
links <- html_elements(XML,
                        "link")
links.xml <- xml_text(links)
```

### 3.3 Debugging-Modus: Ausgewertete Links reduzieren

```
#if (config$debug$toggle == TRUE){
#   links.xml <- links.xml[sample(length(links.xml),
#                                 config$debug$sample)]
#}
```

### 3.4 Links zu HTML Landing Pages generieren

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

### 3.5 Funktion anzeigen: f.linkextract

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

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

### 3.6 Links aus HTML Landing Pages extrahieren

```
if(config$parallel$htmlLandingPages == TRUE){  
  plan("multicore",  
       workers = fullCores)  
}else{  
  plan("sequential")  
}
```

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

### 3.7 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)
```

### 3.8 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 <- html_elements(XML, "title") %>% xml_text()
```

#### 3.8.1 Namen bereinigen und kürzen

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

### 3.8.2 Indizes der AEG bestimmen

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

### 3.8.3 AEGs umbenennen

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

## 3.9 Vektor der Kurztitel erstellen

```
shorttitle <- filenames.pdf  
  
shorttitle <- gsub(".pdf",  
                  "",  
                  shorttitle)  
  
shorttitle <- gsub("_",  
                  "",  
                  shorttitle)
```

## 3.10 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)
```

## 3.11 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)
```

### 3.12 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 = "-")
```

### 3.13 Dateierweiterungen hinzufügen

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

### 3.14 Links zu EPUB-Dateien erstellen

```
prelinks.epub <- gsub("xml.zip",  
                     "",  
                     links.xml)  
  
links.epub <- paste0(prelinks.epub,  
                     filenames.epub)
```

### 3.15 Links zu PDF-Dateien erstellen

```
prelinks.pdf <- gsub("xml.zip",  
                    "",  
                    links.xml)  
  
links.pdf <- paste0(prelinks.pdf,  
                    filenames.pdf)
```

### 3.16 Data Table für Download vorbereiten

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

### 3.17 Abkürzungsverzeichnis erstellen

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

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

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

### 3.18 Download Table als CSV speichern

```
fwrite(download,
        paste0(dir.analysis,
               config$project$shortname,
               "_02_Links.csv"),
        na = "NA")
```

### 3.19 Verzeichnis aller Rechtsakte als CSV speichern

```
fwrite(conctable,
        paste0("output/",
               prefix.files,
               "_DE_AlleRechtsakteVerzeichnis.csv"),
        na = "NA")
```

### 3.20 Debugging-Modus: Anzahl der heruntergeladenen Dateien reduzieren

```
if (config$debug$toggle == TRUE){

  download <- download[sample(download[, .N],
                              config$debug$sample)]

}
```

### 3.21 Anzahl herunterzuladender Dateien

#### 3.21.1 Pro Format

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

```
## [1] 6640
```

### 3.21.2 Insgesamt

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

```
## [1] 19920
```

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

### 4.1 Document Type Definition herunterladen

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

```
download.file("https://www.gesetze-im-internet.de/dtd/1.01/gii-norm.dtd",
             paste0("output/",
                   prefix.files,
                   "_DE_XML_DocumentTypeDefinition_v1-01.dtd"))
```

### 4.2 Download der XML-Dateien

```
if(config$parallel$downloadXML == TRUE){
  plan("multicore",
       workers = fullCores)
}else{
  plan("sequential")
}
```

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

### 4.3 Download-Ergebnis

#### 4.3.1 Anzahl herunterzuladender Dateien

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

```
## [1] 6640
```

#### 4.3.2 Anzahl heruntergeladener Dateien

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



```
## [1] 6640
```

### 4.3.3 Fehlbetrag

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

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

### 4.3.4 Fehlende Dateien

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

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

## 4.4 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(zipfile = file)  
}  
  
unlink(files.zip)
```

## 4.5 XML Dateien auflisten und Dateigrößen speichern

```
files.xml <- list.files(pattern = "\\\\.xml",  
                      ignore.case = TRUE)  
  
xml.MB <- file.size(files.xml) / 106
```

## 4.6 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” ###  
Funktion für XML-Parsing definieren

```
xmlparse.einzelnormen <- function(file.xml){

  ## XML als Character-Vektor einlesen
  xml.char <- readChar(file.xml,
                      file.info(file.xml)$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 <- html_elements(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]] <- html_elements(nodes[i],
                                   xpath = "textdaten//text//Content") %>% xml_
text(trim = TRUE)

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

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

```

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

g.titel.temp[[i]] <- html_elements(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 <- html_elements(XML, xpath = "//norm//gliederungskennzahl")
%>% xml_text(trim = TRUE)
g.bez.vec <- html_elements(XML, xpath = "//norm//gliederungsbez") %>% xml_
text(trim = TRUE)
g.titel.vec <- html_elements(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

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

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

for (i in 1:length(varlist)){

  temp <- html_element(XML, varlist[i]) %>% xml_text(trim = TRUE)
  meta[[i]] <- rep(temp,
                  content.out[,.N])

}

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

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

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

## Standangaben extrahieren
standtyp <- html_elements(XML, "standtyp") %>% xml_text(trim = TRUE)
standkommentar <- html_elements(XML, "standkommentar") %>% xml_text(trim =
TRUE)
standcheck <- html_elements(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)
}

xmlparse.einzelnormen.robust <- function(file.xml){
  tryCatch({xmlparse.einzelnormen(file.xml)},
           error = function(cond) {
             return(NA)}
           )
}

```

#### 4.6.1 Beginn XML Parsing

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

#### 4.6.2 Parallelisierung definieren

```
if(config$parallel$parseEinzelnormen == TRUE){  
  plan("multicore",  
        workers = fullCores)  
}else{  
  plan("sequential")  
}
```

#### 4.6.3 XML Parsen

```
out.einzelnormen <- future_lapply(files.xml,  
                                  xmlparse.einzelnormen.robust)
```

#### 4.6.4 Liste in Data Table umwandeln

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

#### 4.6.5 Ende XML Parsing

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

#### 4.6.6 Dauer XML Parsing

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

```
## Time difference of 4.746102 mins
```

#### 4.6.7 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)
```

#### 4.6.8 Variablen-Name für Ausfertigungsdatum anpassen

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

#### 4.6.9 Variable “fundstellentyp” anpassen

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

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

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

#### 4.6.11 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")
```

#### 4.6.12 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.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")
```

#### 4.6.13 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.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")
```

#### 4.6.14 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")
```

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

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

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

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

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

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

#### 4.6.18 Variable “version” hinzufügen

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

#### 4.6.19 Variable “lizenz” hinzufügen

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



## 4.7 Stichprobe für Qualitätsprüfung ziehen

```
print(config$qa$sample)
```

```
## [1] 50
```

```
idx <- sample(dt.normen[,.N],
              config$qa$sample)

check <- dt.normen[idx]

fwrite(check,
        paste0(dir.analysis,
               prefix.files,
               "_Stichprobe_Normen.csv"),
        na = "NA")
```

## 4.8 Korpus erstellen: Rechtsakte

### 4.8.1 Variablen definieren

Zunächst der vordefinierte Satz an Metadaten.

```
varlist.r1 <- c("jurabk",
               "amtabk",
               "ausfertigung_datum",
               "periodikum",
               "zitstelle",
               "langue",
               "kurzue")
```

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))
```

### 4.8.2 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")
```

### 4.8.3 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"]
```

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

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

## 4.9 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.

### 4.9.1 Funktion für XML-Parsing definieren

```
xmlparse.meta <- function(file.xml){

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

  ## Schleife vorbereiten
  nodes <- html_elements(XML, xpath = "//norm//metadaten")
  scope <- 1:length(nodes)

  ## Metadaten extrahieren

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

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

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

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

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

  meta$doc_id <- file.xml

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

  ## Standangaben extrahieren
  standtyp <- html_elements(XML, "standtyp") %>% xml_text(trim = TRUE)
  standkommentar <- html_elements(XML, "standkommentar") %>% xml_text(trim =
  TRUE)
  standcheck <- html_elements(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)
}

xmlparse.meta.robust <- function(file.xml){
  tryCatch({xmlparse.meta(file.xml)},
          error = function(cond) {
            return(NA)}
          )
}

```

## 4.9.2 Beginn XML Parsing

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

## 4.9.3 Parallelisierung definieren

```
if(config$parallel$parseMeta == TRUE){  
  plan("multicore",  
       workers = fullCores)  
}else{  
  plan("sequential")  
}
```

## 4.9.4 XML Parsen

```
out.meta <- future_lapply(files.xml,  
                          xmlparse.meta.robust)
```

## 4.9.5 Liste in Data Table umwandeln

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

## 4.9.6 Ende XML Parsing

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

## 4.9.7 Dauer XML Parsing

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

```
## Time difference of 1.148411 mins
```

#### 4.9.8 Variablen-Name für Ausfertigungsdatum anpassen

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

#### 4.9.9 Variable “fundstellentyp” anpassen

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

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

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

#### 4.9.11 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")
```

#### 4.9.12 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")
```

#### 4.9.13 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")
```

#### 4.9.14 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")
```

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

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

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

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

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

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

#### 4.9.18 Variable “version” hinzufügen

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

#### 4.9.19 Variable “lizenz” hinzufügen

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



## 4.10 Netzwerk-Analyse (experimentell!)

### 4.10.1 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)]  
  
}
```

### 4.10.2 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)  
  
}
```

### 4.10.3 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)
}

f.split.gliederungseinheit <- function(gliederungseinheit){

  kennzahl <- html_elements(gliederungseinheit, xpath = "gliederungskennzahl")
  %>% xml_text()

  bez <- html_elements(gliederungseinheit, xpath = "gliederungsbez") %>% xml_
  text()

  # Newlines, damit Umbrüche in Diagrammen
  funktionieren
  bez <- gsub(" +",
            "\n",
            bez)

  titel <- html_elements(gliederungseinheit, xpath = "gliederungstitel") %>%
  xml_text()

```

```

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

if(length(titel) == 0){
  titel <- NA
}

dt <- data.table(kennzahl,
                 bez,
                 titel)

return(dt)
}

#xml.name <- "XML/BJNR001950896.xml" # BGB

#xml.name <- "XML/BJNR335610017.xml" # problem
#f.network.analysis(xml.name)

```

#### 4.10.4 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,
                              prefix.figuretitle,
                              caption){
#   message(xml.name) # remove when debugging done
  XML <- read_xml(xml.name)

  ## Gliederungseinheiten extrahieren
  gliederungseinheit <- html_elements(XML, xpath = "//norm//gliederungseinheit"
)

  ## Gliederungseinheit splitten
  gliederungseinheit.split <- lapply(gliederungseinheit,
                                     f.split.gliederungseinheit)
  gliederungseinheit.split <- rbindlist(gliederungseinheit.split)

  gliederungseinheit.split <- unique(gliederungseinheit.split, by = "kennzahl")

  if (gliederungseinheit.split[,.N] > 0){

    ## Abkürzung extrahieren
    jurabk <- html_element(XML, xpath = "//norm//jurabk") %>% xml_text()

    if (length(jurabk) == 0){
      jurabk <- "NA"
    }
  }
}

```

```

## Titel als Label priorieren, sonst Bezeichnung einsetzen
node.labels0 <- ifelse(gliederungseinheit.split$titel != "",
                      gliederungseinheit.split$titel,
                      gliederungseinheit.split$bez)

## Rechtsakt als Quelle des Netzwerks einfügen
node.labels <- c(jurabk,
                 node.labels0)

## Edgelist erstellen
edgelist <- tryCatch({f.kennzahlen.edgelist(kennzahl = gliederungseinheit
.split$kennzahl,
                                           name = jurabk)},
                    error = function(cond) {
                      return(0)}
                    )

# to do: print errorfilename to disk

if (edgelist != 0){

## Node Labels definieren
nodes.df <-gliederungseinheit.split[,.(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"))

## Graph aus Edgelist erstellen
g <- graph.data.frame(edgelist,
                      directed = TRUE,
                      vertices = nodes.df)

## Adjazenz-Matrix erstellen
M.adjacency <- as.matrix(get.adjacency(g,
                                       edges = F))

## Dateiname definieren
filename <- paste0(gsub("( +)|(/)",
                      "-",
                      jurabk),
                  "-",
                  gsub("\\.xml",
                      "",
                      xml.name))

```

```

## Gliederungstabelle speichern
fwrite(gliederungseinheit.split,
       paste0("netzwerke/Gliederungstabellen/",
             filename,
             "_Gliederungstabelle.csv"))

## Edgelist speichern
fwrite(edgelist,
       paste0("netzwerke/Edgelisten/",
             filename,
             "_Edgelist.csv"))

## Adjazenz-Matrix speichern
fwrite(M.adjacency,
       paste0("netzwerke/Adjazenzmatrizen/",
             filename,
             "_AdjazenzMatrix.csv"))

## GraphML speichern
write_graph(g,
            file = paste0("netzwerke/GraphML/",
                          filename,
                          ".graphml"),
            format = "graphml")

## Diagramm erstellen und speichern
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(prefix.figuretitle,
                  "| Struktur des",
                  jurabk),
    caption = caption
  )+
  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
    )
}

}

}

}

f.network.analysis.robust <- function(xml.name,
                                     prefix.figuretitle,
                                     caption){

    tryCatch({f.network.analysis(xml.name,
                                prefix.figuretitle,
                                caption)},
            error = function(cond) {
                return(NA)}
    )

}

```

#### 4.10.5 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",
               "BJNR000939960.xml")

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

length(files.xml)

```

```
## [1] 6628
```

```
#https://www.gesetze-im-internet.de/bgb/BJNR001950896.epub  
#xml.name <- files.xml[205]  
#xml.name <- "XML/BJNR002089971.xml" # problem  
#xml.name <- "XML/BJNR001950896.xml" # BGB
```

#### 4.10.6 Beginn Network Analysis

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

#### 4.10.7 Parallelisierung definieren

Parallele Berechnung funktioniert nicht mit errorfiles; sequentielle Berechnung schon

```
if(config$parallel$parseNetworks == TRUE){  
  plan("multicore",  
       workers = fullCores)  
}else{  
  plan("sequential")  
}  
  
#f.network.analysis.robust(files.xml[35],  
#                          prefix.figuretitle = prefix.figuretitle,  
#                          caption = caption)
```

#### 4.10.8 XML Parsen

```
out.netanalysis <- future_lapply(files.xml,  
                                f.network.analysis.robust,  
                                prefix.figuretitle = prefix.figuretitle,  
                                caption = caption,  
                                future.seed = TRUE)
```

#### 4.10.9 XML-Dateien bei denen Fehler auftreten

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

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

#### 4.10.10 Ende XML Parsing

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

#### 4.10.11 Dauer XML Parsing

```
end.netanalysis - begin.netanalysis
```

```
## Time difference of 3.545953 mins
```

### 4.11 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.

#### 4.11.1 XML-Dateien definieren

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

#### 4.11.2 XML-Dateien verpacken

```
zip(paste0(prefix.files,  
          "_DE_XML_Datensatz.zip"),  
    files.xml,  
    mode = "cherry-pick")  
  
unlink(files.xml)
```

#### 4.11.3 Anhänge zu XML-Dateien verpacken

```
attachments <- list.files(pattern = "(\\.jpg)|\\.gif|\\.pdf|\\.png",  
                          ignore.case = TRUE)  
  
if (length(attachments) > 0){  
  
zip(paste0(prefix.files,
```



```
        "_DE_XML_Anlagen.zip"),
    attachments,
    mode = "cherry-pick")
}

unlink(attachments)
```

## 5 Frequenztabellen erstellen: Einzelnormen

### 5.1 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],
##                 "\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)
##   }
## }

```

## 5.2 Liste zu prüfender Variablen

```
print(config$freqtable$ignore)
```

```

## [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"

```

### 5.3 Frequenztabellen erstellen

```
prefix.freqtable.einzelnormen <- paste0(config$project$shortname,  
                                         "_01_Einzelnormen_Frequenztafel_var-")
```

```
f.fast.freqtable(dt.normen,  
                 varlist = config$freqtable$ignore,  
                 sumrow = TRUE,  
                 output.list = FALSE,  
                 output.kable = TRUE,  
                 output.csv = TRUE,  
                 outputdir = dir.analysis,  
                 prefix = prefix.freqtable.einzelnormen)
```

---

Frequency Table for Variable: periodikum

---

21 unique value(s) detected.

periodikum	N	exactpercent	roundedpercent	cumulpercent
BAGVBl	1	0.0009931	0.00	0.00
BAnz	1243	1.2344208	1.23	1.24
BGBI	2305	2.2890908	2.29	3.52
BGBI I	84613	84.0289985	84.03	87.55
BGBI II	3251	3.2285615	3.23	90.78
GBI DDR	233	0.2313918	0.23	91.01
GBI DDR I	616	0.6117483	0.61	91.63
GBI DDR II	43	0.0427032	0.04	91.67
GVBl BE	9	0.0089379	0.01	91.68
NV	15	0.0148965	0.01	91.69
RAnz	38	0.0377377	0.04	91.73
RGBI	5700	5.6606584	5.66	97.39
RGBI I	1641	1.6296738	1.63	99.02
RGBI II	350	0.3475843	0.35	99.37
RMBI	229	0.2274194	0.23	99.59
VOBl BrZ	19	0.0188689	0.02	99.61

(continued)

periodikum	N	exactpercent	roundedpercent	cumulpercent
VkBl	111	0.1102339	0.11	99.72
WiGBI	122	0.1211580	0.12	99.85
ZBl	63	0.0625652	0.06	99.91
eBAanz	52	0.0516411	0.05	99.96
Öff Anz	41	0.0407170	0.04	100.00
Total	100695	100.0000000	100.00	100.00

Frequency Table for Variable: fundstellentyp

1 unique value(s) detected.

fundstellentyp	N	exactpercent	roundedpercent	cumulpercent
amtlich	100695	100	100	100
Total	100695	100	100	100

Frequency Table for Variable: check\_neuf

2 unique value(s) detected.

check_neuf	N	exactpercent	roundedpercent	cumulpercent
NA	76646	76.11699	76.12	76.12
ja	24049	23.88301	23.88	100.00
Total	100695	100.00000	100.00	100.00

Frequency Table for Variable: check\_aufh

2 unique value(s) detected.

check_aufh	N	exactpercent	roundedpercent	cumulpercent
NA	99027	98.343513	98.34	98.34
ja	1668	1.656487	1.66	100.00
Total	100695	100.000000	100.00	100.00

Frequency Table for Variable: check\_sonst

2 unique value(s) detected.

check_sonst	N	exactpercent	roundedpercent	cumulpercent
NA	83180	82.60589	82.61	82.61
ja	17515	17.39411	17.39	100.00
Total	100695	100.00000	100.00	100.00

Frequency Table for Variable: check\_hinweis

2 unique value(s) detected.

check_hinweis	N	exactpercent	roundedpercent	cumulpercent
NA	69603	69.1226	69.12	69.12
ja	31092	30.8774	30.88	100.00
Total	100695	100.0000	100.00	100.00

Frequency Table for Variable: check\_stand

2 unique value(s) detected.

check_stand	N	exactpercent	roundedpercent	cumulpercent
NA	23045	22.88594	22.89	22.89
ja	77650	77.11406	77.11	100.00
Total	100695	100.00000	100.00	100.00

---

Frequency Table for Variable: gliederungskennzahl

---

1473 unique value(s) detected.

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
000	77	0.0764685	0.08	0.08
000010	12	0.0119172	0.01	0.09
000020	18	0.0178758	0.02	0.11
000030	17	0.0168827	0.02	0.12
000040	17	0.0168827	0.02	0.14
000050	18	0.0178758	0.02	0.16
000060	10	0.0099310	0.01	0.17
000070	3	0.0029793	0.00	0.17
000080	2	0.0019862	0.00	0.17
000090	1	0.0009931	0.00	0.17
000100	2	0.0019862	0.00	0.18
000110	3	0.0029793	0.00	0.18
000120	5	0.0049655	0.00	0.18
001000	2	0.0019862	0.00	0.19
001010	1	0.0009931	0.00	0.19
001020	3	0.0029793	0.00	0.19
001030	1	0.0009931	0.00	0.19
001040	1	0.0009931	0.00	0.19
001051	1	0.0009931	0.00	0.19
001060	2	0.0019862	0.00	0.19
001080	1	0.0009931	0.00	0.20
001090	1	0.0009931	0.00	0.20
001100	1	0.0009931	0.00	0.20
001101	1	0.0009931	0.00	0.20
001110	1	0.0009931	0.00	0.20
001120	5	0.0049655	0.00	0.20

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
010	6151	6.1085456	6.11	6.31
010010	929	0.9225880	0.92	7.24
010010000010	1	0.0009931	0.00	7.24
010010000020	1	0.0009931	0.00	7.24
010010010	118	0.1171856	0.12	7.35
010010020	117	0.1161925	0.12	7.47
010010020010	3	0.0029793	0.00	7.47
010010020010010	37	0.0367446	0.04	7.51
010010020010020	24	0.0238344	0.02	7.53
010010020020	12	0.0119172	0.01	7.55
010010020030	1	0.0009931	0.00	7.55
010010030	28	0.0278067	0.03	7.58
010010030010	7	0.0069517	0.01	7.58
010010030020	16	0.0158896	0.02	7.60
010010030030	12	0.0119172	0.01	7.61
010010030040	12	0.0119172	0.01	7.62
010010040	35	0.0347584	0.03	7.66
010010050	11	0.0109241	0.01	7.67
010010060	6	0.0059586	0.01	7.67
010010070010	10	0.0099310	0.01	7.68
010010070020	5	0.0049655	0.00	7.69
010020	800	0.7944784	0.79	8.48
010020010	151	0.1499578	0.15	8.63
010020010000030	1	0.0009931	0.00	8.63
010020010000040	1	0.0009931	0.00	8.63
010020010000050	1	0.0009931	0.00	8.64
010020010000060	1	0.0009931	0.00	8.64
010020020	138	0.1370475	0.14	8.77
010020020000070	1	0.0009931	0.00	8.78



*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
010020020000080	1	0.0009931	0.00	8.78
010020020000090	1	0.0009931	0.00	8.78
010020020000100	1	0.0009931	0.00	8.78
010020020000110	1	0.0009931	0.00	8.78
010020020000120	1	0.0009931	0.00	8.78
010020020010	16	0.0158896	0.02	8.80
010020020020	19	0.0188689	0.02	8.81
010020020030	9	0.0089379	0.01	8.82
010020020040	14	0.0139034	0.01	8.84
010020020050	18	0.0178758	0.02	8.86
010020030	108	0.1072546	0.11	8.96
010020030000130	1	0.0009931	0.00	8.96
010020030000140	1	0.0009931	0.00	8.96
010020030000170	1	0.0009931	0.00	8.97
010020030000171	1	0.0009931	0.00	8.97
010020030000172	1	0.0009931	0.00	8.97
010020030000190	1	0.0009931	0.00	8.97
010020030000200	1	0.0009931	0.00	8.97
010020030000210	1	0.0009931	0.00	8.97
010020030000220	1	0.0009931	0.00	8.97
010020030000230	1	0.0009931	0.00	8.97
010020030000240	1	0.0009931	0.00	8.97
010020030010	3	0.0029793	0.00	8.98
010020030020	2	0.0019862	0.00	8.98
010020030030	2	0.0019862	0.00	8.98
010020040	76	0.0754754	0.08	9.06
010020040000250	1	0.0009931	0.00	9.06
010020040000260	1	0.0009931	0.00	9.06
010020050	58	0.0575997	0.06	9.12

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
010020050020000380	1	0.0009931	0.00	9.12
010020050020000390	1	0.0009931	0.00	9.12
010020050020000400	1	0.0009931	0.00	9.12
010020050020000410	1	0.0009931	0.00	9.12
010020050020000420	1	0.0009931	0.00	9.12
010020060	44	0.0436963	0.04	9.16
010020060000430	1	0.0009931	0.00	9.17
010020060000440	1	0.0009931	0.00	9.17
010020060000450	1	0.0009931	0.00	9.17
010020060000460	1	0.0009931	0.00	9.17
010020070	31	0.0307860	0.03	9.20
010020070010000461	1	0.0009931	0.00	9.20
010020070020000462	1	0.0009931	0.00	9.20
010020070020000463	1	0.0009931	0.00	9.20
010020070030000464	1	0.0009931	0.00	9.20
010020070040000465	1	0.0009931	0.00	9.20
010020080	41	0.0407170	0.04	9.24
010020090	1	0.0009931	0.00	9.25
010030	534	0.5303143	0.53	9.78
010030000470	1	0.0009931	0.00	9.78
010030000480	1	0.0009931	0.00	9.78
010030010	133	0.1320820	0.13	9.91
010030010010	8	0.0079448	0.01	9.92
010030010020	14	0.0139034	0.01	9.93
010030010030	20	0.0198620	0.02	9.95
010030010040	12	0.0119172	0.01	9.96
010030010050	4	0.0039724	0.00	9.97
010030010060	8	0.0079448	0.01	9.98
010030010070	8	0.0079448	0.01	9.98

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
010030010080	4	0.0039724	0.00	9.99
010030010380	2	0.0019862	0.00	9.99
010030010400	4	0.0039724	0.00	9.99
010030010440	1	0.0009931	0.00	9.99
010030010450	3	0.0029793	0.00	10.00
010030020	84	0.0834202	0.08	10.08
010030020010	86	0.0854064	0.09	10.17
010030020020	41	0.0407170	0.04	10.21
010030020030	12	0.0119172	0.01	10.22
010030020040	12	0.0119172	0.01	10.23
010030020050	9	0.0089379	0.01	10.24
010030020060	18	0.0178758	0.02	10.26
010030030	70	0.0695169	0.07	10.33
010030030010	9	0.0089379	0.01	10.34
010030030020	9	0.0089379	0.01	10.35
010030030030	6	0.0059586	0.01	10.35
010030030040	1	0.0009931	0.00	10.35
010030040	39	0.0387308	0.04	10.39
010030050	39	0.0387308	0.04	10.43
010030060	6	0.0059586	0.01	10.44
010030060010	16	0.0158896	0.02	10.45
010030060020	8	0.0079448	0.01	10.46
010030060030	3	0.0029793	0.00	10.46
010030060040	3	0.0029793	0.00	10.47
010030060050	2	0.0019862	0.00	10.47
010030070	17	0.0168827	0.02	10.48
010031	2	0.0019862	0.00	10.49
010040	345	0.3426188	0.34	10.83
010040010	83	0.0824271	0.08	10.91

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
010040020	69	0.0685238	0.07	10.98
010040020010	3	0.0029793	0.00	10.98
010040020020	5	0.0049655	0.00	10.99
010040020030	2	0.0019862	0.00	10.99
010040030	42	0.0417101	0.04	11.03
010040030010	8	0.0079448	0.01	11.04
010040030020	3	0.0029793	0.00	11.04
010040040	21	0.0208551	0.02	11.06
010040040010	7	0.0069517	0.01	11.07
010040040020	11	0.0109241	0.01	11.08
010040040030	4	0.0039724	0.00	11.08
010040040040	9	0.0089379	0.01	11.09
010040040050	1	0.0009931	0.00	11.09
010040040060	3	0.0029793	0.00	11.10
010040040070	8	0.0079448	0.01	11.11
010040050	2	0.0019862	0.00	11.11
010040120010	1	0.0009931	0.00	11.11
010040120020	1	0.0009931	0.00	11.11
010040120030	1	0.0009931	0.00	11.11
010040120040	1	0.0009931	0.00	11.11
010040120050	1	0.0009931	0.00	11.11
010040120060	1	0.0009931	0.00	11.11
010041	2	0.0019862	0.00	11.12
010050	164	0.1628681	0.16	11.28
010050010	61	0.0605790	0.06	11.34
010050020	86	0.0854064	0.09	11.42
010050020020	2	0.0019862	0.00	11.43
010050030	43	0.0427032	0.04	11.47
010050030010	2	0.0019862	0.00	11.47

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
010050030020	1	0.0009931	0.00	11.47
010050030030	2	0.0019862	0.00	11.47
010050040	2	0.0019862	0.00	11.48
010060	184	0.1827300	0.18	11.66
010060010	48	0.0476687	0.05	11.71
010060020	73	0.0724962	0.07	11.78
010060020010	11	0.0109241	0.01	11.79
010060020020	10	0.0099310	0.01	11.80
010060020030	5	0.0049655	0.00	11.80
010060020040	14	0.0139034	0.01	11.82
010060020050	1	0.0009931	0.00	11.82
010060030	57	0.0566066	0.06	11.88
010060030810	7	0.0069517	0.01	11.88
010060030820	8	0.0079448	0.01	11.89
010060030830	3	0.0029793	0.00	11.89
010060030840	1	0.0009931	0.00	11.90
010060040	14	0.0139034	0.01	11.91
010060050	33	0.0327722	0.03	11.94
010060051	4	0.0039724	0.00	11.95
010060060	23	0.0228413	0.02	11.97
010060070	5	0.0049655	0.00	11.97
010070	223	0.2214608	0.22	12.20
010070010	8	0.0079448	0.01	12.20
010070010010	11	0.0109241	0.01	12.21
010070010020	6	0.0059586	0.01	12.22
010070020	14	0.0139034	0.01	12.23
010070030	14	0.0139034	0.01	12.25
010080	134	0.1330751	0.13	12.38
010080010	9	0.0089379	0.01	12.39

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
010080010010	2	0.0019862	0.00	12.39
010080010020	11	0.0109241	0.01	12.40
010080020	17	0.0168827	0.02	12.42
010080030	8	0.0079448	0.01	12.43
010090	62	0.0615721	0.06	12.49
010090010	23	0.0228413	0.02	12.51
010090020	19	0.0188689	0.02	12.53
010090030	6	0.0059586	0.01	12.54
010091	5	0.0049655	0.00	12.54
010092	1	0.0009931	0.00	12.54
010100	29	0.0287998	0.03	12.57
010100010	2	0.0019862	0.00	12.57
010100020	1	0.0009931	0.00	12.57
010110	37	0.0367446	0.04	12.61
010110010	1	0.0009931	0.00	12.61
010110020	1	0.0009931	0.00	12.61
010110030	1	0.0009931	0.00	12.61
010120	4	0.0039724	0.00	12.62
010130	9	0.0089379	0.01	12.63
010140	5	0.0049655	0.00	12.63
011	11	0.0109241	0.01	12.64
012	2	0.0019862	0.00	12.65
014	2	0.0019862	0.00	12.65
020	6347	6.3031928	6.30	18.95
020000	1	0.0009931	0.00	18.95
020000010	2	0.0019862	0.00	18.95
020000020	8	0.0079448	0.01	18.96
020000030	8	0.0079448	0.01	18.97
020000040	8	0.0079448	0.01	18.98

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020000050	8	0.0079448	0.01	18.99
020000060	8	0.0079448	0.01	18.99
020000500	1	0.0009931	0.00	18.99
020000510	1	0.0009931	0.00	18.99
020000520	1	0.0009931	0.00	19.00
020000530	1	0.0009931	0.00	19.00
020000531	1	0.0009931	0.00	19.00
020010	1630	1.6187497	1.62	20.62
020010010	294	0.2919708	0.29	20.91
020010020	230	0.2284125	0.23	21.14
020010020010	25	0.0248274	0.02	21.16
020010020020	8	0.0079448	0.01	21.17
020010020020010	4	0.0039724	0.00	21.17
020010020020020	3	0.0029793	0.00	21.18
020010020020030	3	0.0029793	0.00	21.18
020010020020040	3	0.0029793	0.00	21.18
020010020020050	1	0.0009931	0.00	21.18
020010020030	12	0.0119172	0.01	21.20
020010020040	6	0.0059586	0.01	21.20
020010020050	5	0.0049655	0.00	21.21
020010020060	4	0.0039724	0.00	21.21
020010030	144	0.1430061	0.14	21.35
020010030010	11	0.0109241	0.01	21.36
020010030020	7	0.0069517	0.01	21.37
020010030030010	4	0.0039724	0.00	21.38
020010030030020010	5	0.0049655	0.00	21.38
020010030030020020	8	0.0079448	0.01	21.39
020010030040	4	0.0039724	0.00	21.39
020010040	91	0.0903719	0.09	21.48

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020010040010	5	0.0049655	0.00	21.49
020010040020	3	0.0029793	0.00	21.49
020010040030	6	0.0059586	0.01	21.50
020010041	5	0.0049655	0.00	21.50
020010050	92	0.0913650	0.09	21.59
020010060	32	0.0317791	0.03	21.62
020010070	38	0.0377377	0.04	21.66
020010070010	4	0.0039724	0.00	21.67
020010070020	7	0.0069517	0.01	21.67
020010070030	7	0.0069517	0.01	21.68
020010080	19	0.0188689	0.02	21.70
020010090	31	0.0307860	0.03	21.73
020010100	11	0.0109241	0.01	21.74
020010110	7	0.0069517	0.01	21.75
020010120	9	0.0089379	0.01	21.76
020011	3	0.0029793	0.00	21.76
020011042	1	0.0009931	0.00	21.76
020020	1549	1.5383088	1.54	23.30
020020010	160	0.1588957	0.16	23.46
020020010010	31	0.0307860	0.03	23.49
020020010010010	2	0.0019862	0.00	23.49
020020010010020	1	0.0009931	0.00	23.49
020020010010030	3	0.0029793	0.00	23.49
020020010010040	1	0.0009931	0.00	23.50
020020010010050	3	0.0029793	0.00	23.50
020020010020	11	0.0109241	0.01	23.51
020020010030	6	0.0059586	0.01	23.52
020020010040	10	0.0099310	0.01	23.53
020020010050	2	0.0019862	0.00	23.53



*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020020010060	6	0.0059586	0.01	23.53
020020010070	3	0.0029793	0.00	23.54
020020010080	2	0.0019862	0.00	23.54
020020020	236	0.2343711	0.23	23.77
020020020010	33	0.0327722	0.03	23.81
020020020020	29	0.0287998	0.03	23.83
020020020030	25	0.0248274	0.02	23.86
020020020040	9	0.0089379	0.01	23.87
020020020050	19	0.0188689	0.02	23.89
020020020060	1	0.0009931	0.00	23.89
020020020070010	3	0.0029793	0.00	23.89
020020020070020	7	0.0069517	0.01	23.90
020020020070030	2	0.0019862	0.00	23.90
020020030	102	0.1012960	0.10	24.00
020020030010	19	0.0188689	0.02	24.02
020020030020	23	0.0228413	0.02	24.04
020020030030	19	0.0188689	0.02	24.06
020020030040	17	0.0168827	0.02	24.08
020020030050	5	0.0049655	0.00	24.08
020020030060	2	0.0019862	0.00	24.09
020020030070	16	0.0158896	0.02	24.10
020020030080	4	0.0039724	0.00	24.11
020020030090	3	0.0029793	0.00	24.11
020020030100	1	0.0009931	0.00	24.11
020020040	71	0.0705100	0.07	24.18
020020043	1	0.0009931	0.00	24.18
020020050	26	0.0258205	0.03	24.21
020020050010	24	0.0238344	0.02	24.23
020020050020	9	0.0089379	0.01	24.24

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020020060	50	0.0496549	0.05	24.29
020020060010	1	0.0009931	0.00	24.29
020020060020	6	0.0059586	0.01	24.30
020020060030	5	0.0049655	0.00	24.30
020020070	12	0.0119172	0.01	24.31
020020080	2	0.0019862	0.00	24.32
020020080010	6	0.0059586	0.01	24.32
020020080020	11	0.0109241	0.01	24.33
020020080030	4	0.0039724	0.00	24.34
020020080040	2	0.0019862	0.00	24.34
020020090	4	0.0039724	0.00	24.34
020020100	13	0.0129103	0.01	24.35
020021	23	0.0228413	0.02	24.38
020030	1108	1.1003525	1.10	25.48
020030010	161	0.1598888	0.16	25.64
020030010010	11	0.0109241	0.01	25.65
020030010020	5	0.0049655	0.00	25.65
020030010020010	2	0.0019862	0.00	25.66
020030010020020	7	0.0069517	0.01	25.66
020030010020030	2	0.0019862	0.00	25.66
020030010020040	1	0.0009931	0.00	25.67
020030010030	8	0.0079448	0.01	25.67
020030010040	6	0.0059586	0.01	25.68
020030020	214	0.2125230	0.21	25.89
020030020010	32	0.0317791	0.03	25.92
020030020020	9	0.0089379	0.01	25.93
020030021010	20	0.0198620	0.02	25.95
020030021020	2	0.0019862	0.00	25.95
020030030	109	0.1082477	0.11	26.06

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020030040	50	0.0496549	0.05	26.11
020030040010	6	0.0059586	0.01	26.12
020030040020	2	0.0019862	0.00	26.12
020030040030	5	0.0049655	0.00	26.13
020030040040	1	0.0009931	0.00	26.13
020030050	39	0.0387308	0.04	26.17
020030050010	9	0.0089379	0.01	26.17
020030050020	16	0.0158896	0.02	26.19
020030060	10	0.0099310	0.01	26.20
020030070	7	0.0069517	0.01	26.21
020030080	9	0.0089379	0.01	26.22
020030090	3	0.0029793	0.00	26.22
020030100	1	0.0009931	0.00	26.22
020030100010	1	0.0009931	0.00	26.22
020030100020	11	0.0109241	0.01	26.23
020030100030	1	0.0009931	0.00	26.23
020030110	3	0.0029793	0.00	26.24
020030120	2	0.0019862	0.00	26.24
020030130	7	0.0069517	0.01	26.24
020030140	2	0.0019862	0.00	26.25
020030150	8	0.0079448	0.01	26.25
020031	18	0.0178758	0.02	26.27
020040	739	0.7338994	0.73	27.01
020040010	66	0.0655445	0.07	27.07
020040020	64	0.0635583	0.06	27.14
020040030	38	0.0377377	0.04	27.17
020040030010	1	0.0009931	0.00	27.17
020040030020	6	0.0059586	0.01	27.18
020040040	22	0.0218482	0.02	27.20

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020040050	5	0.0049655	0.00	27.21
020040060	8	0.0079448	0.01	27.21
020040070	1	0.0009931	0.00	27.22
020040160	1	0.0009931	0.00	27.22
020041	6	0.0059586	0.01	27.22
020050	427	0.4240528	0.42	27.65
020050010	40	0.0397239	0.04	27.69
020050010010	5	0.0049655	0.00	27.69
020050010020	4	0.0039724	0.00	27.70
020050010030	3	0.0029793	0.00	27.70
020050020	39	0.0387308	0.04	27.74
020050030	5	0.0049655	0.00	27.74
020050030010	7	0.0069517	0.01	27.75
020050030020	14	0.0139034	0.01	27.76
020050030030	6	0.0059586	0.01	27.77
020050030040	2	0.0019862	0.00	27.77
020050030050	12	0.0119172	0.01	27.78
020050040	7	0.0069517	0.01	27.79
020050050	1	0.0009931	0.00	27.79
020050180	1	0.0009931	0.00	27.79
020050190	1	0.0009931	0.00	27.79
020050200	1	0.0009931	0.00	27.79
020050210	1	0.0009931	0.00	27.79
020051	7	0.0069517	0.01	27.80
020051211	1	0.0009931	0.00	27.80
020051212	1	0.0009931	0.00	27.80
020051213	1	0.0009931	0.00	27.80
020052	1	0.0009931	0.00	27.81
020053	1	0.0009931	0.00	27.81

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020060	297	0.2949501	0.29	28.10
020060010	31	0.0307860	0.03	28.13
020060020	18	0.0178758	0.02	28.15
020060020010	11	0.0109241	0.01	28.16
020060020020010	2	0.0019862	0.00	28.16
020060020020020	2	0.0019862	0.00	28.17
020060020020030	1	0.0009931	0.00	28.17
020060020020040	1	0.0009931	0.00	28.17
020060020020050	1	0.0009931	0.00	28.17
020060030	10	0.0099310	0.01	28.18
020060030010	2	0.0019862	0.00	28.18
020060030020	2	0.0019862	0.00	28.18
020060030030	12	0.0119172	0.01	28.19
020060030040	6	0.0059586	0.01	28.20
020060030050	1	0.0009931	0.00	28.20
020060040	9	0.0089379	0.01	28.21
020060050	4	0.0039724	0.00	28.21
020060050010	2	0.0019862	0.00	28.22
020060050020	14	0.0139034	0.01	28.23
020060060	6	0.0059586	0.01	28.24
020060070	9	0.0089379	0.01	28.24
020060080010	5	0.0049655	0.00	28.25
020060080020	14	0.0139034	0.01	28.26
020060090	11	0.0109241	0.01	28.27
020060100	1	0.0009931	0.00	28.28
020060110	3	0.0029793	0.00	28.28
020060220	1	0.0009931	0.00	28.28
020060230	2	0.0019862	0.00	28.28
020061	3	0.0029793	0.00	28.28

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020070	185	0.1837231	0.18	28.47
020070010	17	0.0168827	0.02	28.49
020070020	5	0.0049655	0.00	28.49
020070020010	10	0.0099310	0.01	28.50
020070020020	23	0.0228413	0.02	28.52
020070020020010	1	0.0009931	0.00	28.52
020070020020020	1	0.0009931	0.00	28.52
020070020030010	15	0.0148965	0.01	28.54
020070020030020010	3	0.0029793	0.00	28.54
020070020030020020	6	0.0059586	0.01	28.55
020070020030020030	1	0.0009931	0.00	28.55
020070020030020040	2	0.0019862	0.00	28.55
020070020030020050	1	0.0009931	0.00	28.55
020070020030030	1	0.0009931	0.00	28.55
020070020040	1	0.0009931	0.00	28.55
020070020050	2	0.0019862	0.00	28.56
020070020060	5	0.0049655	0.00	28.56
020070020069	1	0.0009931	0.00	28.56
020070020070	3	0.0029793	0.00	28.57
020070020080	1	0.0009931	0.00	28.57
020070020090	1	0.0009931	0.00	28.57
020070030	4	0.0039724	0.00	28.57
020070030010	10	0.0099310	0.01	28.58
020070030020	6	0.0059586	0.01	28.59
020070030030	14	0.0139034	0.01	28.60
020070030040	3	0.0029793	0.00	28.60
020070030050	1	0.0009931	0.00	28.61
020070040	5	0.0049655	0.00	28.61
020070240	1	0.0009931	0.00	28.61

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020080	136	0.1350613	0.14	28.75
020080010	5	0.0049655	0.00	28.75
020080010010	24	0.0238344	0.02	28.78
020080010020010	2	0.0019862	0.00	28.78
020080010020020	7	0.0069517	0.01	28.78
020080010020030	11	0.0109241	0.01	28.79
020080010030	11	0.0109241	0.01	28.81
020080010040	1	0.0009931	0.00	28.81
020080020	18	0.0178758	0.02	28.82
020080030	3	0.0029793	0.00	28.83
020080030010010	3	0.0029793	0.00	28.83
020080030010020	24	0.0238344	0.02	28.85
020080030020	3	0.0029793	0.00	28.86
020080030030	1	0.0009931	0.00	28.86
020080030040	1	0.0009931	0.00	28.86
020080030050	2	0.0019862	0.00	28.86
020080030060	2	0.0019862	0.00	28.86
020080040	22	0.0218482	0.02	28.89
020080050	1	0.0009931	0.00	28.89
020080050010	20	0.0198620	0.02	28.91
020080050020010	7	0.0069517	0.01	28.91
020080050020011	6	0.0059586	0.01	28.92
020080050020020010	4	0.0039724	0.00	28.92
020080050020020011	4	0.0039724	0.00	28.93
020080050020020020	16	0.0158896	0.02	28.94
020080050020030	5	0.0049655	0.00	28.95
020080050020040	14	0.0139034	0.01	28.96
020080050020050010	5	0.0049655	0.00	28.97
020080050020050020	9	0.0089379	0.01	28.98

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020080050020050030	2	0.0019862	0.00	28.98
020080050020050040	3	0.0029793	0.00	28.98
020080050020060	2	0.0019862	0.00	28.98
020080050030	6	0.0059586	0.01	28.99
020080050040	8	0.0079448	0.01	29.00
020080050050	31	0.0307860	0.03	29.03
020080060	10	0.0099310	0.01	29.04
020080070	7	0.0069517	0.01	29.04
020080080	3	0.0029793	0.00	29.05
020080080010	24	0.0238344	0.02	29.07
020080080020	8	0.0079448	0.01	29.08
020080090010010	25	0.0248274	0.02	29.10
020080090010020	8	0.0079448	0.01	29.11
020080090010030	6	0.0059586	0.01	29.12
020080090010040	1	0.0009931	0.00	29.12
020080090020	5	0.0049655	0.00	29.12
020080090030	2	0.0019862	0.00	29.13
020080090040	25	0.0248274	0.02	29.15
020080100010	4	0.0039724	0.00	29.15
020080100020	5	0.0049655	0.00	29.16
020080100030	1	0.0009931	0.00	29.16
020080100040	4	0.0039724	0.00	29.16
020080110	6	0.0059586	0.01	29.17
020080120010	13	0.0129103	0.01	29.18
020080120020	3	0.0029793	0.00	29.19
020080120030010	3	0.0029793	0.00	29.19
020080120030020	4	0.0039724	0.00	29.19
020080120030030010	4	0.0039724	0.00	29.20
020080120030030020	7	0.0069517	0.01	29.20



(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020080120030030030	10	0.0099310	0.01	29.21
020080130	11	0.0109241	0.01	29.22
020080140	13	0.0129103	0.01	29.24
020080150	5	0.0049655	0.00	29.24
020080160	37	0.0367446	0.04	29.28
020080170	18	0.0178758	0.02	29.30
020080180	3	0.0029793	0.00	29.30
020080190	3	0.0029793	0.00	29.30
020080200	14	0.0139034	0.01	29.32
020080210	1	0.0009931	0.00	29.32
020080220	3	0.0029793	0.00	29.32
020080230	10	0.0099310	0.01	29.33
020080240	16	0.0158896	0.02	29.35
020080250	4	0.0039724	0.00	29.35
020080260	11	0.0109241	0.01	29.36
020080270	32	0.0317791	0.03	29.39
020090	106	0.1052684	0.11	29.50
020090010	15	0.0148965	0.01	29.51
020090020	3	0.0029793	0.00	29.52
020090030	12	0.0119172	0.01	29.53
020100	80	0.0794478	0.08	29.61
020100010	8	0.0079448	0.01	29.62
020100020	7	0.0069517	0.01	29.62
020100030	3	0.0029793	0.00	29.63
020100040	2	0.0019862	0.00	29.63
020110	105	0.1042753	0.10	29.73
020110010	1	0.0009931	0.00	29.73
020110020	2	0.0019862	0.00	29.74
020110030	2	0.0019862	0.00	29.74

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
020110040	2	0.0019862	0.00	29.74
020110050	3	0.0029793	0.00	29.74
020120	51	0.0506480	0.05	29.79
020120010	1	0.0009931	0.00	29.79
020120020	3	0.0029793	0.00	29.80
020120030	3	0.0029793	0.00	29.80
020130	73	0.0724962	0.07	29.87
020130010	1	0.0009931	0.00	29.87
020130020	1	0.0009931	0.00	29.87
020130030	7	0.0069517	0.01	29.88
020130040	4	0.0039724	0.00	29.89
020140	95	0.0943443	0.09	29.98
020150	42	0.0417101	0.04	30.02
020160	53	0.0526342	0.05	30.07
020170	39	0.0387308	0.04	30.11
020180	48	0.0476687	0.05	30.16
020190	40	0.0397239	0.04	30.20
020200	37	0.0367446	0.04	30.24
020210	37	0.0367446	0.04	30.27
020220	42	0.0417101	0.04	30.32
020230	46	0.0456825	0.05	30.36
020240	34	0.0337653	0.03	30.39
020250	43	0.0427032	0.04	30.44
020260	35	0.0347584	0.03	30.47
020270	36	0.0357515	0.04	30.51
020280	37	0.0367446	0.04	30.54
020290	13	0.0129103	0.01	30.56
020300	25	0.0248274	0.02	30.58
021	32	0.0317791	0.03	30.61

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
021010	5	0.0049655	0.00	30.62
021020	2	0.0019862	0.00	30.62
021030	4	0.0039724	0.00	30.63
022	3	0.0029793	0.00	30.63
030	5401	5.3637221	5.36	35.99
030000550	1	0.0009931	0.00	35.99
030000560	1	0.0009931	0.00	35.99
030000590	1	0.0009931	0.00	35.99
030000600	1	0.0009931	0.00	36.00
030000610	1	0.0009931	0.00	36.00
030000620	1	0.0009931	0.00	36.00
030000630	1	0.0009931	0.00	36.00
030000640	1	0.0009931	0.00	36.00
030000660	1	0.0009931	0.00	36.00
030000670	1	0.0009931	0.00	36.00
030000680	1	0.0009931	0.00	36.00
030000690	1	0.0009931	0.00	36.00
030000730	1	0.0009931	0.00	36.00
030000740	1	0.0009931	0.00	36.01
030000760	1	0.0009931	0.00	36.01
030000770	1	0.0009931	0.00	36.01
030000780	1	0.0009931	0.00	36.01
030000790	1	0.0009931	0.00	36.01
030000800	1	0.0009931	0.00	36.01
030000810	1	0.0009931	0.00	36.01
030000820	1	0.0009931	0.00	36.01
030000830	1	0.0009931	0.00	36.01
030000840	1	0.0009931	0.00	36.01
030000850	1	0.0009931	0.00	36.02

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030000860	1	0.0009931	0.00	36.02
030000890	1	0.0009931	0.00	36.02
030000900	1	0.0009931	0.00	36.02
030000910	1	0.0009931	0.00	36.02
030000930	1	0.0009931	0.00	36.02
030000940	1	0.0009931	0.00	36.02
030000960	1	0.0009931	0.00	36.02
030000970	1	0.0009931	0.00	36.02
030000980	1	0.0009931	0.00	36.02
030000990	1	0.0009931	0.00	36.03
030001000	1	0.0009931	0.00	36.03
030001010	1	0.0009931	0.00	36.03
030001020	1	0.0009931	0.00	36.03
030001040	1	0.0009931	0.00	36.03
030001050	1	0.0009931	0.00	36.03
030001060	1	0.0009931	0.00	36.03
030001070	1	0.0009931	0.00	36.03
030001080	1	0.0009931	0.00	36.03
030001090	1	0.0009931	0.00	36.03
030001100	1	0.0009931	0.00	36.04
030001110	1	0.0009931	0.00	36.04
030001120	1	0.0009931	0.00	36.04
030001130	1	0.0009931	0.00	36.04
030001140	1	0.0009931	0.00	36.04
030001150	1	0.0009931	0.00	36.04
030001160	1	0.0009931	0.00	36.04
030001170	1	0.0009931	0.00	36.04
030001180	1	0.0009931	0.00	36.04
030001190	1	0.0009931	0.00	36.04

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030001200	1	0.0009931	0.00	36.05
030001210	1	0.0009931	0.00	36.05
030001220	1	0.0009931	0.00	36.05
030001230	1	0.0009931	0.00	36.05
030001240	1	0.0009931	0.00	36.05
030001250	1	0.0009931	0.00	36.05
030001260	1	0.0009931	0.00	36.05
030001270	1	0.0009931	0.00	36.05
030001280	1	0.0009931	0.00	36.05
030001290	1	0.0009931	0.00	36.05
030001300	1	0.0009931	0.00	36.06
030001310	1	0.0009931	0.00	36.06
030001320	1	0.0009931	0.00	36.06
030001330	1	0.0009931	0.00	36.06
030001370	1	0.0009931	0.00	36.06
030001380	1	0.0009931	0.00	36.06
030001390	1	0.0009931	0.00	36.06
030001400	1	0.0009931	0.00	36.06
030001430	1	0.0009931	0.00	36.06
030001440	1	0.0009931	0.00	36.06
030001470	1	0.0009931	0.00	36.07
030001520	1	0.0009931	0.00	36.07
030010	1106	1.0983664	1.10	37.16
030010010	121	0.1201649	0.12	37.28
030010020	116	0.1151994	0.12	37.40
030010020010	4	0.0039724	0.00	37.40
030010020020	6	0.0059586	0.01	37.41
030010020030	6	0.0059586	0.01	37.42
030010030	73	0.0724962	0.07	37.49

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030010030010	17	0.0168827	0.02	37.51
030010030020	8	0.0079448	0.01	37.51
030010030030	4	0.0039724	0.00	37.52
030010030040	6	0.0059586	0.01	37.52
030010030050	1	0.0009931	0.00	37.52
030010031	5	0.0049655	0.00	37.53
030010040	35	0.0347584	0.03	37.56
030010040010	16	0.0158896	0.02	37.58
030010040020	5	0.0049655	0.00	37.58
030010050	19	0.0188689	0.02	37.60
030010060	10	0.0099310	0.01	37.61
030010070	3	0.0029793	0.00	37.62
030010080	3	0.0029793	0.00	37.62
030010150	3	0.0029793	0.00	37.62
030020	1021	1.0139530	1.01	38.64
030020010	106	0.1052684	0.11	38.74
030020010010	11	0.0109241	0.01	38.75
030020010020	13	0.0129103	0.01	38.77
030020010030	6	0.0059586	0.01	38.77
030020010050	4	0.0039724	0.00	38.78
030020010060	7	0.0069517	0.01	38.78
030020020	124	0.1231441	0.12	38.91
030020020010	4	0.0039724	0.00	38.91
030020020020	2	0.0019862	0.00	38.91
030020020030	3	0.0029793	0.00	38.91
030020020040	7	0.0069517	0.01	38.92
030020020050	3	0.0029793	0.00	38.92
030020020060	1	0.0009931	0.00	38.93
030020020070	2	0.0019862	0.00	38.93

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030020020080	2	0.0019862	0.00	38.93
030020020090	5	0.0049655	0.00	38.93
030020020100	1	0.0009931	0.00	38.94
030020030	87	0.0863995	0.09	39.02
030020040	30	0.0297929	0.03	39.05
030020050	21	0.0208551	0.02	39.07
030020060	10	0.0099310	0.01	39.08
030020060010	6	0.0059586	0.01	39.09
030020060020	4	0.0039724	0.00	39.09
030020070	4	0.0039724	0.00	39.10
030020070010	1	0.0009931	0.00	39.10
030020070020	5	0.0049655	0.00	39.10
030020070030	3	0.0029793	0.00	39.11
030020080	9	0.0089379	0.01	39.11
030020090	6	0.0059586	0.01	39.12
030020170	2	0.0019862	0.00	39.12
030020180	2	0.0019862	0.00	39.12
030020190	12	0.0119172	0.01	39.14
030021	2	0.0019862	0.00	39.14
030030	603	0.5988381	0.60	39.74
030030010	103	0.1022891	0.10	39.84
030030020	87	0.0863995	0.09	39.93
030030030	38	0.0377377	0.04	39.96
030030030010	10	0.0099310	0.01	39.97
030030030020	9	0.0089379	0.01	39.98
030030030030	7	0.0069517	0.01	39.99
030030030040	5	0.0049655	0.00	39.99
030030030050	7	0.0069517	0.01	40.00
030030030060	20	0.0198620	0.02	40.02

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030030040	35	0.0347584	0.03	40.06
030030050	17	0.0168827	0.02	40.07
030030060	4	0.0039724	0.00	40.08
030030070	1	0.0009931	0.00	40.08
030030270	2	0.0019862	0.00	40.08
030030280	2	0.0019862	0.00	40.08
030030290	5	0.0049655	0.00	40.09
030030330	4	0.0039724	0.00	40.09
030031	11	0.0109241	0.01	40.10
030032	8	0.0079448	0.01	40.11
030040	345	0.3426188	0.34	40.45
030040010	41	0.0407170	0.04	40.49
030040010010	1	0.0009931	0.00	40.49
030040010020	4	0.0039724	0.00	40.50
030040010030	3	0.0029793	0.00	40.50
030040010040	1	0.0009931	0.00	40.50
030040010050	2	0.0019862	0.00	40.50
030040010060	1	0.0009931	0.00	40.50
030040010070	1	0.0009931	0.00	40.51
030040010080	3	0.0029793	0.00	40.51
030040020	40	0.0397239	0.04	40.55
030040020010	44	0.0436963	0.04	40.59
030040020020	18	0.0178758	0.02	40.61
030040020030	8	0.0079448	0.01	40.62
030040020040	4	0.0039724	0.00	40.62
030040020050	2	0.0019862	0.00	40.62
030040020060	1	0.0009931	0.00	40.62
030040020070	1	0.0009931	0.00	40.63
030040020080	4	0.0039724	0.00	40.63



*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030040030	8	0.0079448	0.01	40.64
030040030010	2	0.0019862	0.00	40.64
030040030020	5	0.0049655	0.00	40.64
030040030030	2	0.0019862	0.00	40.65
030040040	16	0.0158896	0.02	40.66
030040050	2	0.0019862	0.00	40.66
030040100	2	0.0019862	0.00	40.67
030041	2	0.0019862	0.00	40.67
030050	186	0.1847162	0.18	40.85
030050010	49	0.0486618	0.05	40.90
030050020	31	0.0307860	0.03	40.93
030050030	11	0.0109241	0.01	40.94
030050040	7	0.0069517	0.01	40.95
030050050	3	0.0029793	0.00	40.95
030060	132	0.1310889	0.13	41.08
030060010010	1	0.0009931	0.00	41.09
030060010020	3	0.0029793	0.00	41.09
030060010030	4	0.0039724	0.00	41.09
030060010040	1	0.0009931	0.00	41.09
030060010050	1	0.0009931	0.00	41.09
030060010060	1	0.0009931	0.00	41.10
030060020	3	0.0029793	0.00	41.10
030070	75	0.0744823	0.07	41.17
030070010	83	0.0824271	0.08	41.26
030070010010	4	0.0039724	0.00	41.26
030070010020	3	0.0029793	0.00	41.26
030070020	11	0.0109241	0.01	41.27
030070020010	8	0.0079448	0.01	41.28
030070020020	5	0.0049655	0.00	41.29

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030070030	3	0.0029793	0.00	41.29
030070030010	2	0.0019862	0.00	41.29
030070030020	8	0.0079448	0.01	41.30
030070030030	2	0.0019862	0.00	41.30
030070030040	1	0.0009931	0.00	41.30
030070040	6	0.0059586	0.01	41.31
030080	38	0.0377377	0.04	41.35
030080010	56	0.0556135	0.06	41.40
030080020	24	0.0238344	0.02	41.43
030081	2	0.0019862	0.00	41.43
030090	57	0.0566066	0.06	41.48
030100	48	0.0476687	0.05	41.53
030100010	6	0.0059586	0.01	41.54
030100020	4	0.0039724	0.00	41.54
030100030	4	0.0039724	0.00	41.55
030110	29	0.0287998	0.03	41.57
030120	36	0.0357515	0.04	41.61
030130	31	0.0307860	0.03	41.64
030130010	13	0.0129103	0.01	41.65
030130020	12	0.0119172	0.01	41.67
030130030	2	0.0019862	0.00	41.67
030130040	2	0.0019862	0.00	41.67
030140	26	0.0258205	0.03	41.70
030150	48	0.0476687	0.05	41.74
030160	1	0.0009931	0.00	41.74
030170	4	0.0039724	0.00	41.75
030220	1	0.0009931	0.00	41.75
030260	1	0.0009931	0.00	41.75
030280	15	0.0148965	0.01	41.76

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
030970	57	0.0566066	0.06	41.82
030971	5	0.0049655	0.00	41.83
030980	1	0.0009931	0.00	41.83
030990	1	0.0009931	0.00	41.83
030991	1	0.0009931	0.00	41.83
030992	1	0.0009931	0.00	41.83
030993	1	0.0009931	0.00	41.83
031	13	0.0129103	0.01	41.84
032	5	0.0049655	0.00	41.85
040	4219	4.1898803	4.19	46.04
040001570	1	0.0009931	0.00	46.04
040001630	1	0.0009931	0.00	46.04
040001640	1	0.0009931	0.00	46.04
040001650	1	0.0009931	0.00	46.04
040001660	1	0.0009931	0.00	46.04
040001670	1	0.0009931	0.00	46.04
040001680	1	0.0009931	0.00	46.05
040001690	1	0.0009931	0.00	46.05
040001700	1	0.0009931	0.00	46.05
040001710	1	0.0009931	0.00	46.05
040001720	1	0.0009931	0.00	46.05
040001730	1	0.0009931	0.00	46.05
040001740	1	0.0009931	0.00	46.05
040001750	1	0.0009931	0.00	46.05
040001760	1	0.0009931	0.00	46.05
040001770	1	0.0009931	0.00	46.05
040001780	1	0.0009931	0.00	46.06
040001790	1	0.0009931	0.00	46.06
040001800	1	0.0009931	0.00	46.06

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
040001810	1	0.0009931	0.00	46.06
040001820	1	0.0009931	0.00	46.06
040001830	1	0.0009931	0.00	46.06
040001840	1	0.0009931	0.00	46.06
040001850	1	0.0009931	0.00	46.06
040001860	1	0.0009931	0.00	46.06
040001870	1	0.0009931	0.00	46.06
040001880	1	0.0009931	0.00	46.07
040001890	1	0.0009931	0.00	46.07
040001900	1	0.0009931	0.00	46.07
040001910	1	0.0009931	0.00	46.07
040001920	1	0.0009931	0.00	46.07
040001930	1	0.0009931	0.00	46.07
040001940	1	0.0009931	0.00	46.07
040001950	1	0.0009931	0.00	46.07
040001960	1	0.0009931	0.00	46.07
040001970	1	0.0009931	0.00	46.07
040001980	1	0.0009931	0.00	46.08
040001990	1	0.0009931	0.00	46.08
040002000	1	0.0009931	0.00	46.08
040002010	1	0.0009931	0.00	46.08
040002020	1	0.0009931	0.00	46.08
040002030	1	0.0009931	0.00	46.08
040002070	1	0.0009931	0.00	46.08
040002080	1	0.0009931	0.00	46.08
040002090	1	0.0009931	0.00	46.08
040002100	1	0.0009931	0.00	46.08
040002120	1	0.0009931	0.00	46.09
040002130	1	0.0009931	0.00	46.09

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
040002140	1	0.0009931	0.00	46.09
040002150	1	0.0009931	0.00	46.09
040002170	1	0.0009931	0.00	46.09
040002180	1	0.0009931	0.00	46.09
040010	505	0.5015145	0.50	46.59
040010010	57	0.0566066	0.06	46.65
040010020	76	0.0754754	0.08	46.72
040010020010	7	0.0069517	0.01	46.73
040010020020	20	0.0198620	0.02	46.75
040010020030	1	0.0009931	0.00	46.75
040010020040	3	0.0029793	0.00	46.76
040010030	30	0.0297929	0.03	46.78
040010030010	8	0.0079448	0.01	46.79
040010030020	4	0.0039724	0.00	46.80
040010030030	7	0.0069517	0.01	46.80
040010040	19	0.0188689	0.02	46.82
040010050	27	0.0268136	0.03	46.85
040010060	1	0.0009931	0.00	46.85
040010060010	28	0.0278067	0.03	46.88
040010060020010	6	0.0059586	0.01	46.88
040010060020020	1	0.0009931	0.00	46.89
040010060020030010	7	0.0069517	0.01	46.89
040010060020030020	28	0.0278067	0.03	46.92
040010060020030030	20	0.0198620	0.02	46.94
040010060020030040	12	0.0119172	0.01	46.95
040010060020030050	36	0.0357515	0.04	46.99
040010060020040	1	0.0009931	0.00	46.99
040010060030	6	0.0059586	0.01	46.99
040010070	1	0.0009931	0.00	47.00

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
040010070010	5	0.0049655	0.00	47.00
040010070011	2	0.0019862	0.00	47.00
040010070020010	1	0.0009931	0.00	47.00
040010070020020	13	0.0129103	0.01	47.02
040010070020030	4	0.0039724	0.00	47.02
040010070020040	4	0.0039724	0.00	47.02
040010070020050	3	0.0029793	0.00	47.03
040010070030	1	0.0009931	0.00	47.03
040010080	3	0.0029793	0.00	47.03
040020	428	0.4250459	0.43	47.46
040020010	141	0.1400268	0.14	47.60
040020010010	5	0.0049655	0.00	47.60
040020010020	13	0.0129103	0.01	47.61
040020010030	8	0.0079448	0.01	47.62
040020010040	11	0.0109241	0.01	47.63
040020010050	6	0.0059586	0.01	47.64
040020010060	7	0.0069517	0.01	47.65
040020010070	4	0.0039724	0.00	47.65
040020010080	1	0.0009931	0.00	47.65
040020020	117	0.1161925	0.12	47.77
040020020010	29	0.0287998	0.03	47.80
040020020020	8	0.0079448	0.01	47.80
040020020030	12	0.0119172	0.01	47.82
040020020040	15	0.0148965	0.01	47.83
040020020050	7	0.0069517	0.01	47.84
040020030	62	0.0615721	0.06	47.90
040020030010	28	0.0278067	0.03	47.93
040020030020	5	0.0049655	0.00	47.93
040020030030	3	0.0029793	0.00	47.93

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
040020030040	1	0.0009931	0.00	47.94
040020040	37	0.0367446	0.04	47.97
040020050	82	0.0814340	0.08	48.05
040020060	16	0.0158896	0.02	48.07
040020070	12	0.0119172	0.01	48.08
040020070010	27	0.0268136	0.03	48.11
040020070020	6	0.0059586	0.01	48.11
040020080	7	0.0069517	0.01	48.12
040020090	5	0.0049655	0.00	48.13
040030	387	0.3843289	0.38	48.51
040030010	28	0.0278067	0.03	48.54
040030010010	38	0.0377377	0.04	48.58
040030010020	52	0.0516411	0.05	48.63
040030010030	21	0.0208551	0.02	48.65
040030010040	2	0.0019862	0.00	48.65
040030010050	7	0.0069517	0.01	48.66
040030010060	14	0.0139034	0.01	48.67
040030020	61	0.0605790	0.06	48.73
040030020010	9	0.0089379	0.01	48.74
040030020020	2	0.0019862	0.00	48.74
040030030	49	0.0486618	0.05	48.79
040030040	11	0.0109241	0.01	48.80
040030040010	1	0.0009931	0.00	48.80
040030040020	1	0.0009931	0.00	48.80
040030050	2	0.0019862	0.00	48.81
040030060	1	0.0009931	0.00	48.81
040040	204	0.2025920	0.20	49.01
040040010	71	0.0705100	0.07	49.08
040040020	14	0.0139034	0.01	49.09

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
040040030	5	0.0049655	0.00	49.10
040050	94	0.0933512	0.09	49.19
040050010	5	0.0049655	0.00	49.20
040050020	3	0.0029793	0.00	49.20
040050030	7	0.0069517	0.01	49.21
040060	47	0.0466756	0.05	49.25
040060010	6	0.0059586	0.01	49.26
040060020	4	0.0039724	0.00	49.26
040060050	5	0.0049655	0.00	49.27
040070	10	0.0099310	0.01	49.28
040080	18	0.0178758	0.02	49.30
040090	34	0.0337653	0.03	49.33
040100	16	0.0158896	0.02	49.35
040110	5	0.0049655	0.00	49.35
040120	6	0.0059586	0.01	49.36
040130	5	0.0049655	0.00	49.36
040140	2	0.0019862	0.00	49.36
040160	14	0.0139034	0.01	49.38
040170	6	0.0059586	0.01	49.38
040190	1	0.0009931	0.00	49.39
040200	1	0.0009931	0.00	49.39
040390	1	0.0009931	0.00	49.39
040400	8	0.0079448	0.01	49.40
040410	1	0.0009931	0.00	49.40
040420	1	0.0009931	0.00	49.40
040430	1	0.0009931	0.00	49.40
040440	1	0.0009931	0.00	49.40
040450	1	0.0009931	0.00	49.40
040460	1	0.0009931	0.00	49.40



*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
040470	1	0.0009931	0.00	49.40
040480	1	0.0009931	0.00	49.40
040500	1	0.0009931	0.00	49.40
040510	1	0.0009931	0.00	49.41
040520	1	0.0009931	0.00	49.41
041	38	0.0377377	0.04	49.44
041010	9	0.0089379	0.01	49.45
041020	5	0.0049655	0.00	49.46
041030	2	0.0019862	0.00	49.46
041040	10	0.0099310	0.01	49.47
041451	3	0.0029793	0.00	49.47
041452	3	0.0029793	0.00	49.48
041454	14	0.0139034	0.01	49.49
042	2	0.0019862	0.00	49.49
042010	1	0.0009931	0.00	49.49
042020	9	0.0089379	0.01	49.50
042030	1	0.0009931	0.00	49.50
042040	11	0.0109241	0.01	49.51
050	3162	3.1401758	3.14	52.65
050002190	1	0.0009931	0.00	52.66
050002200	1	0.0009931	0.00	52.66
050002210	1	0.0009931	0.00	52.66
050002220	1	0.0009931	0.00	52.66
050002230	1	0.0009931	0.00	52.66
050002231	1	0.0009931	0.00	52.66
050002240	3	0.0029793	0.00	52.66
050002250	1	0.0009931	0.00	52.66
050002260	1	0.0009931	0.00	52.66
050002270	1	0.0009931	0.00	52.67

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
050002280	1	0.0009931	0.00	52.67
050002290	59	0.0585928	0.06	52.73
050010	316	0.3138190	0.31	53.04
050010010	34	0.0337653	0.03	53.07
050010020	39	0.0387308	0.04	53.11
050010030	16	0.0158896	0.02	53.13
050010040	42	0.0417101	0.04	53.17
050010050	34	0.0337653	0.03	53.20
050010050010	1	0.0009931	0.00	53.20
050010050020	1	0.0009931	0.00	53.21
050010050030	2	0.0019862	0.00	53.21
050010050040	2	0.0019862	0.00	53.21
050010060	5	0.0049655	0.00	53.21
050010070	3	0.0029793	0.00	53.22
050010080	4	0.0039724	0.00	53.22
050010090	5	0.0049655	0.00	53.23
050010100010	3	0.0029793	0.00	53.23
050010100020	3	0.0029793	0.00	53.23
050010100030	1	0.0009931	0.00	53.23
050010110010	1	0.0009931	0.00	53.23
050010110020	19	0.0188689	0.02	53.25
050010110030	13	0.0129103	0.01	53.27
050010110040	8	0.0079448	0.01	53.27
050010110050	10	0.0099310	0.01	53.28
050010110060	1	0.0009931	0.00	53.28
050010120	9	0.0089379	0.01	53.29
050011	5	0.0049655	0.00	53.30
050020	297	0.2949501	0.29	53.59
050020010	72	0.0715031	0.07	53.67

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
050020010010	12	0.0119172	0.01	53.68
050020010010010	17	0.0168827	0.02	53.69
050020010010020	15	0.0148965	0.01	53.71
050020010010030	14	0.0139034	0.01	53.72
050020010020	12	0.0119172	0.01	53.73
050020020	38	0.0377377	0.04	53.77
050020020010	5	0.0049655	0.00	53.78
050020020020	15	0.0148965	0.01	53.79
050020020030	31	0.0307860	0.03	53.82
050020020040	28	0.0278067	0.03	53.85
050020020050	4	0.0039724	0.00	53.85
050020030	54	0.0536273	0.05	53.91
050020040	53	0.0526342	0.05	53.96
050020040010	28	0.0278067	0.03	53.99
050020040020	16	0.0158896	0.02	54.00
050020040030	8	0.0079448	0.01	54.01
050020050	17	0.0168827	0.02	54.03
050020060	11	0.0109241	0.01	54.04
050020070	4	0.0039724	0.00	54.04
050020080	2	0.0019862	0.00	54.05
050020090	2	0.0019862	0.00	54.05
050020100	2	0.0019862	0.00	54.05
050020110	9	0.0089379	0.01	54.06
050030	198	0.1966334	0.20	54.26
050030010	37	0.0367446	0.04	54.29
050030020	31	0.0307860	0.03	54.32
050030030	54	0.0536273	0.05	54.38
050030040	48	0.0476687	0.05	54.42
050030050	5	0.0049655	0.00	54.43

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
050030060	32	0.0317791	0.03	54.46
050030070	23	0.0228413	0.02	54.48
050030080	9	0.0089379	0.01	54.49
050040	110	0.1092408	0.11	54.60
050040010	9	0.0089379	0.01	54.61
050040020	17	0.0168827	0.02	54.63
050040030	8	0.0079448	0.01	54.64
050050	82	0.0814340	0.08	54.72
050060	20	0.0198620	0.02	54.74
050070	23	0.0228413	0.02	54.76
050080	22	0.0218482	0.02	54.78
050090	16	0.0158896	0.02	54.80
050460	6	0.0059586	0.01	54.80
050660	1	0.0009931	0.00	54.81
051	45	0.0446894	0.04	54.85
052	12	0.0119172	0.01	54.86
053	3	0.0029793	0.00	54.86
060	2335	2.3188838	2.32	57.18
060002300	1	0.0009931	0.00	57.18
060002310	10	0.0099310	0.01	57.19
060002320	12	0.0119172	0.01	57.21
060002330010	13	0.0129103	0.01	57.22
060002330020	7	0.0069517	0.01	57.23
060002340	12	0.0119172	0.01	57.24
060002350	2	0.0019862	0.00	57.24
060002360	3	0.0029793	0.00	57.24
060002370	2	0.0019862	0.00	57.25
060010	194	0.1926610	0.19	57.44
060010010	9	0.0089379	0.01	57.45

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
060010020	23	0.0228413	0.02	57.47
060010030	11	0.0109241	0.01	57.48
060010040	12	0.0119172	0.01	57.49
060010050	5	0.0049655	0.00	57.50
060010060	3	0.0029793	0.00	57.50
060010070	6	0.0059586	0.01	57.51
060020	188	0.1867024	0.19	57.69
060020010	25	0.0248274	0.02	57.72
060020020	23	0.0228413	0.02	57.74
060020030	2	0.0019862	0.00	57.74
060020030010	4	0.0039724	0.00	57.75
060020030020	24	0.0238344	0.02	57.77
060020030030	13	0.0129103	0.01	57.78
060020040	2	0.0019862	0.00	57.79
060020050	3	0.0029793	0.00	57.79
060020060	1	0.0009931	0.00	57.79
060021	4	0.0039724	0.00	57.79
060030	128	0.1271165	0.13	57.92
060030010	23	0.0228413	0.02	57.94
060030020	20	0.0198620	0.02	57.96
060030030	6	0.0059586	0.01	57.97
060030040	2	0.0019862	0.00	57.97
060030050	12	0.0119172	0.01	57.98
060030060	1	0.0009931	0.00	57.98
060040	58	0.0575997	0.06	58.04
060040010	7	0.0069517	0.01	58.05
060040020	6	0.0059586	0.01	58.05
060050	7	0.0069517	0.01	58.06
060101	6	0.0059586	0.01	58.07

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
060102	5	0.0049655	0.00	58.07
060110	3	0.0029793	0.00	58.08
060670	1	0.0009931	0.00	58.08
061	17	0.0168827	0.02	58.09
070	1579	1.5681017	1.57	59.66
070002390	1	0.0009931	0.00	59.66
070002400	7	0.0069517	0.01	59.67
070002420	2	0.0019862	0.00	59.67
070002430	1	0.0009931	0.00	59.67
070002440	1	0.0009931	0.00	59.67
070002460	1	0.0009931	0.00	59.67
070002461	4	0.0039724	0.00	59.68
070002462	2	0.0019862	0.00	59.68
070002463	1	0.0009931	0.00	59.68
070002470	19	0.0188689	0.02	59.70
070002471	2	0.0019862	0.00	59.70
070002480010	2	0.0019862	0.00	59.70
070002480020	9	0.0089379	0.01	59.71
070002480030	6	0.0059586	0.01	59.72
070002480040	4	0.0039724	0.00	59.72
070002490	3	0.0029793	0.00	59.73
070002500	10	0.0099310	0.01	59.74
070002510	2	0.0019862	0.00	59.74
070002520	1	0.0009931	0.00	59.74
070002530	3	0.0029793	0.00	59.74
070010	193	0.1916679	0.19	59.93
070010010	10	0.0099310	0.01	59.94
070010020	10	0.0099310	0.01	59.95
070010030	7	0.0069517	0.01	59.96

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
070010040	7	0.0069517	0.01	59.97
070010050	5	0.0049655	0.00	59.97
070010060	1	0.0009931	0.00	59.97
070010070	1	0.0009931	0.00	59.97
070011	1	0.0009931	0.00	59.98
070020	159	0.1579026	0.16	60.13
070020010	13	0.0129103	0.01	60.15
070020020	20	0.0198620	0.02	60.17
070020020010	3	0.0029793	0.00	60.17
070020020020	6	0.0059586	0.01	60.17
070020020030	1	0.0009931	0.00	60.18
070020030	13	0.0129103	0.01	60.19
070020040	1	0.0009931	0.00	60.19
070030	68	0.0675307	0.07	60.26
070030010	7	0.0069517	0.01	60.26
070030020	3	0.0029793	0.00	60.27
070030020010	8	0.0079448	0.01	60.28
070030020020	2	0.0019862	0.00	60.28
070030030	1	0.0009931	0.00	60.28
070030040	1	0.0009931	0.00	60.28
070040	34	0.0337653	0.03	60.31
070050	29	0.0287998	0.03	60.34
070060	6	0.0059586	0.01	60.35
070070	2	0.0019862	0.00	60.35
070080	5	0.0049655	0.00	60.35
070090	6	0.0059586	0.01	60.36
070680	1	0.0009931	0.00	60.36
070690	1	0.0009931	0.00	60.36
071	3	0.0029793	0.00	60.37

(continued)

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
080	1027	1.0199116	1.02	61.39
080010	234	0.2323849	0.23	61.62
080010010	8	0.0079448	0.01	61.63
080010020	24	0.0238344	0.02	61.65
080010030	14	0.0139034	0.01	61.66
080010040	7	0.0069517	0.01	61.67
080010050	6	0.0059586	0.01	61.68
080020	157	0.1559164	0.16	61.83
080020010	15	0.0148965	0.01	61.85
080020020	8	0.0079448	0.01	61.86
080020020010	6	0.0059586	0.01	61.86
080020020020	23	0.0228413	0.02	61.88
080020020030	55	0.0546204	0.05	61.94
080020030	14	0.0139034	0.01	61.95
080020040	12	0.0119172	0.01	61.96
080020050	8	0.0079448	0.01	61.97
080020060	8	0.0079448	0.01	61.98
080030	73	0.0724962	0.07	62.05
080030010	12	0.0119172	0.01	62.06
080030020010	3	0.0029793	0.00	62.07
080030020020	3	0.0029793	0.00	62.07
080030020030	2	0.0019862	0.00	62.07
080030020040	1	0.0009931	0.00	62.07
080030020050	1	0.0009931	0.00	62.07
080030030	2	0.0019862	0.00	62.08
080030040	1	0.0009931	0.00	62.08
080040	67	0.0665376	0.07	62.14
080050	52	0.0516411	0.05	62.20
080060	26	0.0258205	0.03	62.22



*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
080060010	4	0.0039724	0.00	62.23
080060020	3	0.0029793	0.00	62.23
080060030	5	0.0049655	0.00	62.23
080060040	2	0.0019862	0.00	62.24
080070	3	0.0029793	0.00	62.24
080080	7	0.0069517	0.01	62.25
081	5	0.0049655	0.00	62.25
090	1085	1.0775113	1.08	63.33
090010	43	0.0427032	0.04	63.37
090010010	25	0.0248274	0.02	63.40
090010020	12	0.0119172	0.01	63.41
090020	49	0.0486618	0.05	63.46
090020010	9	0.0089379	0.01	63.46
090020020	20	0.0198620	0.02	63.48
090020030	7	0.0069517	0.01	63.49
090030	69	0.0685238	0.07	63.56
090040	32	0.0317791	0.03	63.59
090040010	11	0.0109241	0.01	63.60
090040020	3	0.0029793	0.00	63.61
090050	20	0.0198620	0.02	63.63
090060	1	0.0009931	0.00	63.63
091	11	0.0109241	0.01	63.64
100	576	0.5720244	0.57	64.21
100010	34	0.0337653	0.03	64.24
100010010	5	0.0049655	0.00	64.25
100010020	10	0.0099310	0.01	64.26
100020	32	0.0317791	0.03	64.29
100020010	19	0.0188689	0.02	64.31
100020020	13	0.0129103	0.01	64.32

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
100020030	3	0.0029793	0.00	64.32
100030	27	0.0268136	0.03	64.35
100030010	4	0.0039724	0.00	64.36
100030020	4	0.0039724	0.00	64.36
100040	6	0.0059586	0.01	64.37
100050	14	0.0139034	0.01	64.38
100051	1	0.0009931	0.00	64.38
100060	8	0.0079448	0.01	64.39
100070	1	0.0009931	0.00	64.39
100080	2	0.0019862	0.00	64.39
100090	4	0.0039724	0.00	64.40
100100	1	0.0009931	0.00	64.40
101	11	0.0109241	0.01	64.41
102	11	0.0109241	0.01	64.42
102010	1	0.0009931	0.00	64.42
102020	2	0.0019862	0.00	64.42
102030010	10	0.0099310	0.01	64.43
102030020010	4	0.0039724	0.00	64.44
102030020020	6	0.0059586	0.01	64.44
102030020030	1	0.0009931	0.00	64.44
102030030	5	0.0049655	0.00	64.45
103	1	0.0009931	0.00	64.45
103010	1	0.0009931	0.00	64.45
103020	1	0.0009931	0.00	64.45
103030	1	0.0009931	0.00	64.45
103040	1	0.0009931	0.00	64.45
103050	1	0.0009931	0.00	64.45
103060	1	0.0009931	0.00	64.45
103070	1	0.0009931	0.00	64.46

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
103080	1	0.0009931	0.00	64.46
103090	1	0.0009931	0.00	64.46
103100	1	0.0009931	0.00	64.46
103110	1	0.0009931	0.00	64.46
103120	1	0.0009931	0.00	64.46
103130	1	0.0009931	0.00	64.46
104	1	0.0009931	0.00	64.46
105	1	0.0009931	0.00	64.46
105010	1	0.0009931	0.00	64.46
106	1	0.0009931	0.00	64.46
107	1	0.0009931	0.00	64.47
107010	1	0.0009931	0.00	64.47
108	1	0.0009931	0.00	64.47
109	1	0.0009931	0.00	64.47
110	424	0.4210735	0.42	64.89
110010	76	0.0754754	0.08	64.97
110010010	6	0.0059586	0.01	64.97
110010020	3	0.0029793	0.00	64.97
110010030	6	0.0059586	0.01	64.98
110020	32	0.0317791	0.03	65.01
110020010	13	0.0129103	0.01	65.03
110020020	5	0.0049655	0.00	65.03
110020030	5	0.0049655	0.00	65.04
110030	29	0.0287998	0.03	65.06
110030010	2	0.0019862	0.00	65.07
110030020	6	0.0059586	0.01	65.07
110030030	4	0.0039724	0.00	65.08
110030040	2	0.0019862	0.00	65.08
110040	7	0.0069517	0.01	65.08

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
110040010	3	0.0029793	0.00	65.09
110040020	5	0.0049655	0.00	65.09
110050	8	0.0079448	0.01	65.10
110050010	10	0.0099310	0.01	65.11
110050020	4	0.0039724	0.00	65.11
110050020010	4	0.0039724	0.00	65.12
110050020020	6	0.0059586	0.01	65.12
110050020030	2	0.0019862	0.00	65.13
110050020040	2	0.0019862	0.00	65.13
110050030	2	0.0019862	0.00	65.13
110050040	5	0.0049655	0.00	65.14
110050050	2	0.0019862	0.00	65.14
110050060	2	0.0019862	0.00	65.14
110050070	2	0.0019862	0.00	65.14
110050080	1	0.0009931	0.00	65.14
110060	10	0.0099310	0.01	65.15
110060010	9	0.0089379	0.01	65.16
110060020	5	0.0049655	0.00	65.17
110070	6	0.0059586	0.01	65.17
110070010	2	0.0019862	0.00	65.17
110070020	6	0.0059586	0.01	65.18
110080	12	0.0119172	0.01	65.19
111	1	0.0009931	0.00	65.19
112	2	0.0019862	0.00	65.19
120	197	0.1956403	0.20	65.39
120010	43	0.0427032	0.04	65.43
120020	24	0.0238344	0.02	65.46
120030	24	0.0238344	0.02	65.48
130	140	0.1390337	0.14	65.62

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
130010	13	0.0129103	0.01	65.63
130020	24	0.0238344	0.02	65.66
130030	10	0.0099310	0.01	65.67
130040	9	0.0089379	0.01	65.68
130050	26	0.0258205	0.03	65.70
130060	2	0.0019862	0.00	65.70
130070	3	0.0029793	0.00	65.71
130080	5	0.0049655	0.00	65.71
140	108	0.1072546	0.11	65.82
150	92	0.0913650	0.09	65.91
150010	8	0.0079448	0.01	65.92
150020	8	0.0079448	0.01	65.93
150030	1	0.0009931	0.00	65.93
160	109	0.1082477	0.11	66.04
160010	13	0.0129103	0.01	66.05
160020	8	0.0079448	0.01	66.06
160030	11	0.0109241	0.01	66.07
170	29	0.0287998	0.03	66.10
171	5	0.0049655	0.00	66.10
172	2	0.0019862	0.00	66.10
174	2	0.0019862	0.00	66.10
180	29	0.0287998	0.03	66.13
180010	35	0.0347584	0.03	66.17
180020	4	0.0039724	0.00	66.17
180030	5	0.0049655	0.00	66.18
180040	1	0.0009931	0.00	66.18
180050	1	0.0009931	0.00	66.18
180060	1	0.0009931	0.00	66.18
180070	1	0.0009931	0.00	66.18

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
180080	1	0.0009931	0.00	66.18
180090	1	0.0009931	0.00	66.18
180100	1	0.0009931	0.00	66.18
180110	1	0.0009931	0.00	66.19
180120	1	0.0009931	0.00	66.19
180130	1	0.0009931	0.00	66.19
180140	3	0.0029793	0.00	66.19
180160	1	0.0009931	0.00	66.19
180170	1	0.0009931	0.00	66.19
180180	1	0.0009931	0.00	66.19
180190	4	0.0039724	0.00	66.20
190	21	0.0208551	0.02	66.22
200	13	0.0129103	0.01	66.23
210	21	0.0208551	0.02	66.25
220	22	0.0218482	0.02	66.27
230	25	0.0248274	0.02	66.30
231	1	0.0009931	0.00	66.30
232	1	0.0009931	0.00	66.30
240	14	0.0139034	0.01	66.31
250	13	0.0129103	0.01	66.33
260	5	0.0049655	0.00	66.33
270	3	0.0029793	0.00	66.33
280	4	0.0039724	0.00	66.34
281	16	0.0158896	0.02	66.35
290	5	0.0049655	0.00	66.36
300	3	0.0029793	0.00	66.36
301	1	0.0009931	0.00	66.36
310	5	0.0049655	0.00	66.37
320	2	0.0019862	0.00	66.37

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
330	16	0.0158896	0.02	66.39
340	6	0.0059586	0.01	66.39
350	6	0.0059586	0.01	66.40
360	2	0.0019862	0.00	66.40
370	2	0.0019862	0.00	66.40
371	1	0.0009931	0.00	66.40
380	4	0.0039724	0.00	66.41
390	2	0.0019862	0.00	66.41
400	2	0.0019862	0.00	66.41
410	3	0.0029793	0.00	66.41
420	4	0.0039724	0.00	66.42
421	1	0.0009931	0.00	66.42
430	2	0.0019862	0.00	66.42
440	5	0.0049655	0.00	66.43
450	1	0.0009931	0.00	66.43
460	6	0.0059586	0.01	66.43
460460	3	0.0029793	0.00	66.44
460490	2	0.0019862	0.00	66.44
460510	8	0.0079448	0.01	66.45
460590	4	0.0039724	0.00	66.45
460620	1	0.0009931	0.00	66.45
470	3	0.0029793	0.00	66.45
480	4	0.0039724	0.00	66.46
491	4	0.0039724	0.00	66.46
510	2	0.0019862	0.00	66.46
520	6	0.0059586	0.01	66.47
530	2	0.0019862	0.00	66.47
550	1	0.0009931	0.00	66.47
560	2	0.0019862	0.00	66.47

*(continued)*

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
580	1	0.0009931	0.00	66.48
590	1	0.0009931	0.00	66.48
600	2	0.0019862	0.00	66.48
610	2	0.0019862	0.00	66.48
620	8	0.0079448	0.01	66.49
630	7	0.0069517	0.01	66.50
640	1	0.0009931	0.00	66.50
650	1	0.0009931	0.00	66.50
680	1	0.0009931	0.00	66.50
681	2	0.0019862	0.00	66.50
690	2	0.0019862	0.00	66.50
700	2	0.0019862	0.00	66.50
700010	1	0.0009931	0.00	66.51
700020	1	0.0009931	0.00	66.51
700030	1	0.0009931	0.00	66.51
700031	1	0.0009931	0.00	66.51
700032	1	0.0009931	0.00	66.51
700040	1	0.0009931	0.00	66.51
700050	1	0.0009931	0.00	66.51
700060	1	0.0009931	0.00	66.51
700070	1	0.0009931	0.00	66.51
700080	1	0.0009931	0.00	66.51
700090	1	0.0009931	0.00	66.52
700100	1	0.0009931	0.00	66.52
700110	1	0.0009931	0.00	66.52
700120	1	0.0009931	0.00	66.52
700130	1	0.0009931	0.00	66.52
700140	1	0.0009931	0.00	66.52
700150	1	0.0009931	0.00	66.52



*(continued)*

---

gliederungskennzahl	N	exactpercent	roundedpercent	cumulpercent
700160	1	0.0009931	0.00	66.52
700170	1	0.0009931	0.00	66.52
700180	1	0.0009931	0.00	66.52
710	2	0.0019862	0.00	66.53
720	7	0.0069517	0.01	66.53
730	1	0.0009931	0.00	66.53
730740	5	0.0049655	0.00	66.54
731	6	0.0059586	0.01	66.55
740	6	0.0059586	0.01	66.55
780	2	0.0019862	0.00	66.55
800	1	0.0009931	0.00	66.55
810	1	0.0009931	0.00	66.56
820	1	0.0009931	0.00	66.56
830	5	0.0049655	0.00	66.56
840	4	0.0039724	0.00	66.57
870	1	0.0009931	0.00	66.57
880	5	0.0049655	0.00	66.57
911	1	0.0009931	0.00	66.57
920	1	0.0009931	0.00	66.57
930	1	0.0009931	0.00	66.57
950	1	0.0009931	0.00	66.58
960	2	0.0019862	0.00	66.58
980	1	0.0009931	0.00	66.58
990	1	0.0009931	0.00	66.58
991	3	0.0029793	0.00	66.58
NA	33650	33.4177467	33.42	100.00
Total	100695	100.0000000	99.00	100.00

---

---

Frequency Table for Variable: ausfertigung\_jahr

---

123 unique value(s) detected.

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1869	148	0.1469785	0.15	0.15
1871	565	0.5611004	0.56	0.71
1877	99	0.0983167	0.10	0.81
1884	10	0.0099310	0.01	0.82
1886	3	0.0029793	0.00	0.82
1887	4	0.0039724	0.00	0.82
1889	216	0.2145092	0.21	1.04
1892	116	0.1151994	0.12	1.15
1894	4	0.0039724	0.00	1.16
1895	74	0.0734892	0.07	1.23
1896	2903	2.8829634	2.88	4.11
1897	1130	1.1222007	1.12	5.24
1898	3	0.0029793	0.00	5.24
1899	2	0.0019862	0.00	5.24
1901	51	0.0506480	0.05	5.29
1903	1	0.0009931	0.00	5.29
1906	1	0.0009931	0.00	5.29
1907	26	0.0258205	0.03	5.32
1908	8	0.0079448	0.01	5.33
1909	129	0.1281096	0.13	5.46
1910	11	0.0109241	0.01	5.47
1911	47	0.0466756	0.05	5.51
1913	46	0.0456825	0.05	5.56
1919	99	0.0983167	0.10	5.66
1920	7	0.0069517	0.01	5.66
1921	40	0.0397239	0.04	5.70

*(continued)*

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1922	206	0.2045782	0.20	5.91
1923	15	0.0148965	0.01	5.92
1924	2	0.0019862	0.00	5.92
1925	2	0.0019862	0.00	5.93
1926	31	0.0307860	0.03	5.96
1927	32	0.0317791	0.03	5.99
1928	8	0.0079448	0.01	6.00
1929	8	0.0079448	0.01	6.01
1930	14	0.0139034	0.01	6.02
1931	46	0.0456825	0.05	6.06
1933	196	0.1946472	0.19	6.26
1934	515	0.5114455	0.51	6.77
1935	156	0.1549233	0.15	6.93
1936	287	0.2850191	0.29	7.21
1937	333	0.3307016	0.33	7.54
1938	41	0.0407170	0.04	7.58
1939	90	0.0893788	0.09	7.67
1940	215	0.2135161	0.21	7.89
1941	25	0.0248274	0.02	7.91
1942	20	0.0198620	0.02	7.93
1943	24	0.0238344	0.02	7.95
1944	12	0.0119172	0.01	7.97
1947	19	0.0188689	0.02	7.98
1948	73	0.0724962	0.07	8.06
1949	314	0.3118328	0.31	8.37
1950	2106	2.0914643	2.09	10.46
1951	584	0.5799692	0.58	11.04
1952	445	0.4419286	0.44	11.48
1953	1823	1.8104176	1.81	13.29

*(continued)*

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1954	209	0.2075575	0.21	13.50
1955	303	0.3009087	0.30	13.80
1956	704	0.6991410	0.70	14.50
1957	975	0.9682705	0.97	15.47
1958	150	0.1489647	0.15	15.62
1959	754	0.7487959	0.75	16.37
1960	1105	1.0973733	1.10	17.46
1961	1282	1.2731516	1.27	18.74
1962	210	0.2085506	0.21	18.95
1963	336	0.3336809	0.33	19.28
1964	332	0.3297085	0.33	19.61
1965	1483	1.4727643	1.47	21.08
1966	570	0.5660658	0.57	21.65
1967	330	0.3277223	0.33	21.98
1968	477	0.4737077	0.47	22.45
1969	723	0.7180098	0.72	23.17
1970	425	0.4220666	0.42	23.59
1971	776	0.7706440	0.77	24.36
1972	829	0.8232782	0.82	25.18
1973	239	0.2373504	0.24	25.42
1974	771	0.7656785	0.77	26.19
1975	913	0.9066984	0.91	27.09
1976	2505	2.4877104	2.49	29.58
1977	750	0.7448235	0.74	30.33
1978	404	0.4012116	0.40	30.73
1979	531	0.5273350	0.53	31.25
1980	1106	1.0983664	1.10	32.35
1981	570	0.5660658	0.57	32.92
1982	535	0.5313074	0.53	33.45

*(continued)*

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1983	244	0.2423159	0.24	33.69
1984	357	0.3545360	0.35	34.05
1985	596	0.5918864	0.59	34.64
1986	564	0.5601073	0.56	35.20
1987	455	0.4518596	0.45	35.65
1988	1283	1.2741447	1.27	36.92
1989	1142	1.1341179	1.13	38.06
1990	2447	2.4301107	2.43	40.49
1991	756	0.7507821	0.75	41.24
1992	993	0.9861463	0.99	42.23
1993	983	0.9762153	0.98	43.20
1994	3780	3.7539103	3.75	46.96
1995	748	0.7428373	0.74	47.70
1996	913	0.9066984	0.91	48.61
1997	1842	1.8292865	1.83	50.43
1998	1459	1.4489299	1.45	51.88
1999	781	0.7756095	0.78	52.66
2000	705	0.7001341	0.70	53.36
2001	1645	1.6336462	1.63	54.99
2002	2152	2.1371468	2.14	57.13
2003	1177	1.1688763	1.17	58.30
2004	2497	2.4797656	2.48	60.78
2005	2020	2.0060579	2.01	62.78
2006	2218	2.2026913	2.20	64.99
2007	1659	1.6475495	1.65	66.63
2008	2152	2.1371468	2.14	68.77
2009	2660	2.6416406	2.64	71.41
2010	1426	1.4161577	1.42	72.83
2011	2358	2.3417250	2.34	75.17

(continued)

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
2012	1491	1.4807091	1.48	76.65
2013	3389	3.3656090	3.37	80.02
2014	1634	1.6227221	1.62	81.64
2015	2050	2.0358508	2.04	83.68
2016	3123	3.1014450	3.10	86.78
2017	3523	3.4986841	3.50	90.28
2018	1611	1.5998808	1.60	91.88
2019	1771	1.7587765	1.76	93.64
2020	3041	3.0200109	3.02	96.66
2021	3368	3.3447540	3.34	100.00
Total	100695	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	100695	100	100	100
Total	100695	100	100	100

Frequency Table for Variable: doi\_version

1 unique value(s) detected.

doi_version	N	exactpercent	roundedpercent	cumulpercent
10.5281/zenodo.5832094	100695	100	100	100
Total	100695	100	100	100

---

Frequency Table for Variable: version

---

1 unique value(s) detected.

---

version	N	exactpercent	roundedpercent	cumulpercent
2022-01-12	100695	100	100	100
Total	100695	100	100	100

---

---

Frequency Table for Variable: lizenz

---

1 unique value(s) detected.

---

	lizenz	N	exactpercent	roundedpercent	cumulpercent
Creative Commons Zero 1.0 Universal		100695	100	100	100
	Total	100695	100	100	100

---

## 6 Frequenztabellen erstellen: Rechtsakte

### 6.1 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 = "|"),
                                 config$freqtable$ignore,
                                 invert = TRUE,
                                 value = TRUE)
```

### 6.2 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"
```

### 6.3 Frequenztabellen erstellen

```
prefix.freqtable.rechtsakte <- paste0(config$project$shortname,
                                       "_01_Rechtsakte_Frequenztafel_var-")
```

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

---

Frequency Table for Variable: periodikum

---

21 unique value(s) detected.



periodikum	N	exactpercent	roundedpercent	cumulpercent
BAGVBl	1	0.0177117	0.02	0.02
BAnz	145	2.5681899	2.57	2.59
BGBI	21	0.3719447	0.37	2.96
BGBI I	4861	86.0963514	86.10	89.05
BGBI II	375	6.6418704	6.64	95.70
GBl DDR	15	0.2656748	0.27	95.96
GBl DDR I	43	0.7616011	0.76	96.72
GBl DDR II	3	0.0531350	0.05	96.78
GVBl BE	1	0.0177117	0.02	96.79
NV	1	0.0177117	0.02	96.81
RAnz	2	0.0354233	0.04	96.85
RGBI	51	0.9032944	0.90	97.75
RGBI I	47	0.8324478	0.83	98.58
RGBI II	45	0.7970244	0.80	99.38
RMBI	4	0.0708466	0.07	99.45
VOBl BrZ	1	0.0177117	0.02	99.47
VkBl	11	0.1948282	0.19	99.66
WiGBl	7	0.1239816	0.12	99.79
ZBl	2	0.0354233	0.04	99.82
eBAnz	6	0.1062699	0.11	99.93
Öff Anz	4	0.0708466	0.07	100.00
Total	5646	100.0000000	100.00	100.00

Frequency Table for Variable: fundstellentyp

1 unique value(s) detected.

fundstellentyp	N	exactpercent	roundedpercent	cumulpercent
amtlich	5646	100	100	100
Total	5646	100	100	100

---

---

Frequency Table for Variable: check\_neuf

---

---

2 unique value(s) detected.

---

check_neuf	N	exactpercent	roundedpercent	cumulpercent
NA	5234	92.702798	92.7	92.7
ja	412	7.297202	7.3	100.0
Total	5646	100.000000	100.0	100.0

---

---

---

Frequency Table for Variable: check\_aufh

---

---

2 unique value(s) detected.

---

check_aufh	N	exactpercent	roundedpercent	cumulpercent
NA	5510	97.591215	97.59	97.59
ja	136	2.408785	2.41	100.00
Total	5646	100.000000	100.00	100.00

---

---

---

Frequency Table for Variable: check\_sonst

---

---

2 unique value(s) detected.

---

check_sonst	N	exactpercent	roundedpercent	cumulpercent
NA	5184	91.817216	91.82	91.82
ja	462	8.182784	8.18	100.00
Total	5646	100.000000	100.00	100.00

---

---

---

Frequency Table for Variable: check\_hinweis

---

---

2 unique value(s) detected.

check_hinweis	N	exactpercent	roundedpercent	cumulpercent
NA	5117	90.630535	90.63	90.63
ja	529	9.369465	9.37	100.00
Total	5646	100.000000	100.00	100.00

Frequency Table for Variable: check\_stand

2 unique value(s) detected.

check_stand	N	exactpercent	roundedpercent	cumulpercent
NA	2896	51.29295	51.29	51.29
ja	2750	48.70705	48.71	100.00
Total	5646	100.000000	100.00	100.00

Frequency Table for Variable: ausfertigung\_jahr

123 unique value(s) detected.

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1869	1	0.0177117	0.02	0.02
1871	3	0.0531350	0.05	0.07
1877	3	0.0531350	0.05	0.12
1884	1	0.0177117	0.02	0.14
1886	2	0.0354233	0.04	0.18
1887	1	0.0177117	0.02	0.19
1889	2	0.0354233	0.04	0.23
1892	1	0.0177117	0.02	0.25
1894	2	0.0354233	0.04	0.28
1895	2	0.0354233	0.04	0.32
1896	2	0.0354233	0.04	0.35
1897	6	0.1062699	0.11	0.46

*(continued)*

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1898	1	0.0177117	0.02	0.48
1899	2	0.0354233	0.04	0.51
1901	2	0.0354233	0.04	0.55
1903	1	0.0177117	0.02	0.57
1906	1	0.0177117	0.02	0.58
1907	1	0.0177117	0.02	0.60
1908	1	0.0177117	0.02	0.62
1909	3	0.0531350	0.05	0.67
1910	2	0.0354233	0.04	0.71
1911	2	0.0354233	0.04	0.74
1913	2	0.0354233	0.04	0.78
1919	4	0.0708466	0.07	0.85
1920	1	0.0177117	0.02	0.87
1921	4	0.0708466	0.07	0.94
1922	6	0.1062699	0.11	1.04
1923	2	0.0354233	0.04	1.08
1924	2	0.0354233	0.04	1.12
1925	2	0.0354233	0.04	1.15
1926	4	0.0708466	0.07	1.22
1927	5	0.0885583	0.09	1.31
1928	3	0.0531350	0.05	1.36
1929	3	0.0531350	0.05	1.42
1930	3	0.0531350	0.05	1.47
1931	5	0.0885583	0.09	1.56
1933	8	0.1416932	0.14	1.70
1934	7	0.1239816	0.12	1.82
1935	3	0.0531350	0.05	1.88
1936	7	0.1239816	0.12	2.00
1937	9	0.1594049	0.16	2.16

*(continued)*

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1938	10	0.1771165	0.18	2.34
1939	6	0.1062699	0.11	2.44
1940	7	0.1239816	0.12	2.57
1941	2	0.0354233	0.04	2.60
1942	2	0.0354233	0.04	2.64
1943	1	0.0177117	0.02	2.66
1944	1	0.0177117	0.02	2.67
1947	1	0.0177117	0.02	2.69
1948	5	0.0885583	0.09	2.78
1949	12	0.2125399	0.21	2.99
1950	21	0.3719447	0.37	3.37
1951	32	0.5667729	0.57	3.93
1952	14	0.2479632	0.25	4.18
1953	41	0.7261778	0.73	4.91
1954	17	0.3010981	0.30	5.21
1955	26	0.4605030	0.46	5.67
1956	33	0.5844846	0.58	6.25
1957	45	0.7970244	0.80	7.05
1958	20	0.3542331	0.35	7.40
1959	27	0.4782147	0.48	7.88
1960	33	0.5844846	0.58	8.47
1961	52	0.9210060	0.92	9.39
1962	30	0.5313496	0.53	9.92
1963	26	0.4605030	0.46	10.38
1964	28	0.4959263	0.50	10.87
1965	45	0.7970244	0.80	11.67
1966	25	0.4427914	0.44	12.11
1967	39	0.6907545	0.69	12.81
1968	41	0.7261778	0.73	13.53

*(continued)*

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1969	54	0.9564293	0.96	14.49
1970	44	0.7793128	0.78	15.27
1971	54	0.9564293	0.96	16.22
1972	53	0.9387177	0.94	17.16
1973	33	0.5844846	0.58	17.75
1974	52	0.9210060	0.92	18.67
1975	64	1.1335459	1.13	19.80
1976	73	1.2929508	1.29	21.09
1977	58	1.0272759	1.03	22.12
1978	44	0.7793128	0.78	22.90
1979	40	0.7084662	0.71	23.61
1980	53	0.9387177	0.94	24.55
1981	40	0.7084662	0.71	25.26
1982	41	0.7261778	0.73	25.98
1983	29	0.5136380	0.51	26.50
1984	32	0.5667729	0.57	27.06
1985	40	0.7084662	0.71	27.77
1986	51	0.9032944	0.90	28.68
1987	38	0.6730429	0.67	29.35
1988	48	0.8501594	0.85	30.20
1989	50	0.8855827	0.89	31.08
1990	157	2.7807297	2.78	33.86
1991	75	1.3283741	1.33	35.19
1992	75	1.3283741	1.33	36.52
1993	81	1.4346440	1.43	37.96
1994	160	2.8338647	2.83	40.79
1995	70	1.2398158	1.24	42.03
1996	84	1.4877790	1.49	43.52
1997	106	1.8774354	1.88	45.39

(continued)

---

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1998	120	2.1253985	2.13	47.52
1999	77	1.3637974	1.36	48.88
2000	60	1.0626993	1.06	49.95
2001	110	1.9482820	1.95	51.90
2002	145	2.5681899	2.57	54.46
2003	96	1.7003188	1.70	56.16
2004	139	2.4619199	2.46	58.63
2005	129	2.2848034	2.28	60.91
2006	129	2.2848034	2.28	63.20
2007	102	1.8065887	1.81	65.00
2008	124	2.1962451	2.20	67.20
2009	153	2.7098831	2.71	69.91
2010	118	2.0899752	2.09	72.00
2011	120	2.1253985	2.13	74.12
2012	106	1.8774354	1.88	76.00
2013	176	3.1172512	3.12	79.12
2014	102	1.8065887	1.81	80.92
2015	130	2.3025151	2.30	83.23
2016	147	2.6036132	2.60	85.83
2017	177	3.1349628	3.13	88.97
2018	94	1.6648955	1.66	90.63
2019	116	2.0545519	2.05	92.69
2020	178	3.1526745	3.15	95.84
2021	235	4.1622388	4.16	100.00
Total	5646	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	5646	100	100	100
Total	5646	100	100	100

Frequency Table for Variable: doi\_version

1 unique value(s) detected.

doi_version	N	exactpercent	roundedpercent	cumulpercent
10.5281/zenodo.5832094	5646	100	100	100
Total	5646	100	100	100

Frequency Table for Variable: version

1 unique value(s) detected.

version	N	exactpercent	roundedpercent	cumulpercent
2022-01-12	5646	100	100	100
Total	5646	100	100	100

Frequency Table for Variable: lizenz

1 unique value(s) detected.

lizenz	N	exactpercent	roundedpercent	cumulpercent
Creative Commons Zero 1.0 Universal	5646	100	100	100
Total	5646	100	100	100



## 7 Frequenztabellen erstellen: XML-Metadaten

### 7.1 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"
```

### 7.2 Frequenztabellen erstellen

```
prefix.freqtable.meta <- paste0(config$project$shortname,  
                                "_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 = dir.analysis,  
                 prefix = prefix.freqtable.meta)
```

---

Frequency Table for Variable: periodikum

---

36 unique value(s) detected.

periodikum	N	exactpercent	roundedpercent	cumulpercent
NA	4	0.0602410	0.06	0.06
ABl EU	1	0.0150602	0.02	0.08
BAGVBl	2	0.0301205	0.03	0.11
BAnz	256	3.8554217	3.86	3.96
BGBI	23	0.3463855	0.35	4.31
BGBI I	4880	73.4939759	73.49	77.80
BGBI II	1171	17.6355422	17.64	95.44

(continued)

periodikum	N	exactpercent	roundedpercent	cumulpercent
BGBI III	1	0.0150602	0.02	95.45
BremGBI	4	0.0602410	0.06	95.51
GBI BW	1	0.0150602	0.02	95.53
GBI DDR	15	0.2259036	0.23	95.75
GBI DDR I	44	0.6626506	0.66	96.42
GBI DDR II	3	0.0451807	0.05	96.46
GV NW	2	0.0301205	0.03	96.49
GVB I BB	1	0.0150602	0.02	96.51
GVB I BE	1	0.0150602	0.02	96.52
GVB I BY	4	0.0602410	0.06	96.58
GVB I HE	5	0.0753012	0.08	96.66
GVB I RP I	2	0.0301205	0.03	96.69
Mtbl BAA	1	0.0150602	0.02	96.70
NV	4	0.0602410	0.06	96.76
Nds GVB I	1	0.0150602	0.02	96.78
RAnz	2	0.0301205	0.03	96.81
RBesBI	1	0.0150602	0.02	96.82
RGBI	56	0.8433735	0.84	97.67
RGBI I	54	0.8132530	0.81	98.48
RGBI II	48	0.7228916	0.72	99.20
RMBI	4	0.0602410	0.06	99.26
RegBI WB	4	0.0602410	0.06	99.32
RegBI WH	1	0.0150602	0.02	99.34
VOBI BrZ	3	0.0451807	0.05	99.38
VkBI	22	0.3313253	0.33	99.71
WiGBI	7	0.1054217	0.11	99.82
ZBI	2	0.0301205	0.03	99.85
eBAnz	6	0.0903614	0.09	99.94
Öff Anz	4	0.0602410	0.06	100.00

(continued)

---

periodikum	N	exactpercent	roundedpercent	cumulpercent
Total	6640	100.0000000	100.00	100.00

---

---

Frequency Table for Variable: fundstellentyp

---

2 unique value(s) detected.

---

fundstellentyp	N	exactpercent	roundedpercent	cumulpercent
amtlich	6636	99.939759	99.94	99.94
nichtamtlich	4	0.060241	0.06	100.00
Total	6640	100.000000	100.00	100.00

---

---

Frequency Table for Variable: check\_neuf

---

2 unique value(s) detected.

---

check_neuf	N	exactpercent	roundedpercent	cumulpercent
NA	6228	93.795181	93.8	93.8
ja	412	6.204819	6.2	100.0
Total	6640	100.000000	100.0	100.0

---

---

Frequency Table for Variable: check\_aufh

---

2 unique value(s) detected.

---

check_aufh	N	exactpercent	roundedpercent	cumulpercent
NA	6496	97.831325	97.83	97.83
ja	144	2.168675	2.17	100.00
Total	6640	100.000000	100.00	100.00

---

---

Frequency Table for Variable: check\_sonst

---

2 unique value(s) detected.

check_sonst	N	exactpercent	roundedpercent	cumulpercent
NA	6168	92.891566	92.89	92.89
ja	472	7.108434	7.11	100.00
Total	6640	100.000000	100.00	100.00

---

Frequency Table for Variable: check\_hinweis

---

2 unique value(s) detected.

check_hinweis	N	exactpercent	roundedpercent	cumulpercent
NA	6111	92.033132	92.03	92.03
ja	529	7.966868	7.97	100.00
Total	6640	100.000000	100.00	100.00

---

Frequency Table for Variable: check\_stand

---

2 unique value(s) detected.

check_stand	N	exactpercent	roundedpercent	cumulpercent
NA	3888	58.55422	58.55	58.55
ja	2752	41.44578	41.45	100.00
Total	6640	100.00000	100.00	100.00

---

Frequency Table for Variable: ausfertigung\_jahr

---

126 unique value(s) detected.

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
NA	1	0.0150602	0.02	0.02
1869	1	0.0150602	0.02	0.03
1871	3	0.0451807	0.05	0.08
1877	3	0.0451807	0.05	0.12
1884	1	0.0150602	0.02	0.14
1886	2	0.0301205	0.03	0.17
1887	1	0.0150602	0.02	0.18
1889	2	0.0301205	0.03	0.21
1892	1	0.0150602	0.02	0.23
1894	2	0.0301205	0.03	0.26
1895	2	0.0301205	0.03	0.29
1896	2	0.0301205	0.03	0.32
1897	6	0.0903614	0.09	0.41
1898	1	0.0150602	0.02	0.42
1899	2	0.0301205	0.03	0.45
1901	2	0.0301205	0.03	0.48
1903	1	0.0150602	0.02	0.50
1906	1	0.0150602	0.02	0.51
1907	1	0.0150602	0.02	0.53
1908	1	0.0150602	0.02	0.54
1909	4	0.0602410	0.06	0.60
1910	2	0.0301205	0.03	0.63
1911	3	0.0451807	0.05	0.68
1913	2	0.0301205	0.03	0.71
1919	7	0.1054217	0.11	0.81
1920	1	0.0150602	0.02	0.83
1921	4	0.0602410	0.06	0.89
1922	6	0.0903614	0.09	0.98
1923	2	0.0301205	0.03	1.01
1924	2	0.0301205	0.03	1.04

*(continued)*

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1925	2	0.0301205	0.03	1.07
1926	4	0.0602410	0.06	1.13
1927	5	0.0753012	0.08	1.20
1928	4	0.0602410	0.06	1.27
1929	4	0.0602410	0.06	1.33
1930	3	0.0451807	0.05	1.37
1931	5	0.0753012	0.08	1.45
1933	9	0.1355422	0.14	1.58
1934	7	0.1054217	0.11	1.69
1935	3	0.0451807	0.05	1.73
1936	9	0.1355422	0.14	1.87
1937	9	0.1355422	0.14	2.00
1938	11	0.1656627	0.17	2.17
1939	11	0.1656627	0.17	2.33
1940	8	0.1204819	0.12	2.45
1941	2	0.0301205	0.03	2.48
1942	2	0.0301205	0.03	2.52
1943	1	0.0150602	0.02	2.53
1944	4	0.0602410	0.06	2.59
1945	1	0.0150602	0.02	2.61
1946	12	0.1807229	0.18	2.79
1947	11	0.1656627	0.17	2.95
1948	13	0.1957831	0.20	3.15
1949	18	0.2710843	0.27	3.42
1950	23	0.3463855	0.35	3.77
1951	41	0.6174699	0.62	4.38
1952	24	0.3614458	0.36	4.74
1953	47	0.7078313	0.71	5.45
1954	26	0.3915663	0.39	5.84

*(continued)*

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1955	34	0.5120482	0.51	6.36
1956	54	0.8132530	0.81	7.17
1957	55	0.8283133	0.83	8.00
1958	36	0.5421687	0.54	8.54
1959	48	0.7228916	0.72	9.26
1960	45	0.6777108	0.68	9.94
1961	69	1.0391566	1.04	10.98
1962	40	0.6024096	0.60	11.58
1963	36	0.5421687	0.54	12.12
1964	42	0.6325301	0.63	12.76
1965	61	0.9186747	0.92	13.67
1966	38	0.5722892	0.57	14.25
1967	50	0.7530120	0.75	15.00
1968	50	0.7530120	0.75	15.75
1969	71	1.0692771	1.07	16.82
1970	56	0.8433735	0.84	17.67
1971	67	1.0090361	1.01	18.67
1972	66	0.9939759	0.99	19.67
1973	69	1.0391566	1.04	20.71
1974	60	0.9036145	0.90	21.61
1975	74	1.1144578	1.11	22.73
1976	82	1.2349398	1.23	23.96
1977	71	1.0692771	1.07	25.03
1978	57	0.8584337	0.86	25.89
1979	56	0.8433735	0.84	26.73
1980	68	1.0240964	1.02	27.76
1981	47	0.7078313	0.71	28.46
1982	46	0.6927711	0.69	29.16
1983	40	0.6024096	0.60	29.76

*(continued)*

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
1984	52	0.7831325	0.78	30.54
1985	52	0.7831325	0.78	31.33
1986	74	1.1144578	1.11	32.44
1987	46	0.6927711	0.69	33.13
1988	61	0.9186747	0.92	34.05
1989	59	0.8885542	0.89	34.94
1990	171	2.5753012	2.58	37.52
1991	82	1.2349398	1.23	38.75
1992	90	1.3554217	1.36	40.11
1993	98	1.4759036	1.48	41.58
1994	186	2.8012048	2.80	44.38
1995	113	1.7018072	1.70	46.08
1996	104	1.5662651	1.57	47.65
1997	137	2.0632530	2.06	49.71
1998	135	2.0331325	2.03	51.75
1999	93	1.4006024	1.40	53.15
2000	72	1.0843373	1.08	54.23
2001	125	1.8825301	1.88	56.11
2002	152	2.2891566	2.29	58.40
2003	115	1.7319277	1.73	60.14
2004	151	2.2740964	2.27	62.41
2005	151	2.2740964	2.27	64.68
2006	138	2.0783133	2.08	66.76
2007	115	1.7319277	1.73	68.49
2008	147	2.2138554	2.21	70.71
2009	166	2.5000000	2.50	73.21
2010	134	2.0180723	2.02	75.23
2011	133	2.0030120	2.00	77.23
2012	120	1.8072289	1.81	79.04



(continued)

---

ausfertigung_jahr	N	exactpercent	roundedpercent	cumulpercent
2013	180	2.7108434	2.71	81.75
2014	103	1.5512048	1.55	83.30
2015	132	1.9879518	1.99	85.29
2016	152	2.2891566	2.29	87.58
2017	180	2.7108434	2.71	90.29
2018	95	1.4307229	1.43	91.72
2019	119	1.7921687	1.79	93.51
2020	183	2.7560241	2.76	96.27
2021	248	3.7349398	3.73	100.00
Total	6640	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	6640	100	100	100
Total	6640	100	100	100

---

---

Frequency Table for Variable: doi\_version

---

1 unique value(s) detected.

---

doi_version	N	exactpercent	roundedpercent	cumulpercent
10.5281/zenodo.5832094	6640	100	100	100
Total	6640	100	100	100

---

---

Frequency Table for Variable: version

---

1 unique value(s) detected.

---

version	N	exactpercent	roundedpercent	cumulpercent
2022-01-12	6640	100	100	100
Total	6640	100	100	100

---

---

Frequency Table for Variable: lizenz

---

1 unique value(s) detected.

---

	lizenz	N	exactpercent	roundedpercent	cumulpercent
Creative Commons Zero 1.0 Universal		6640	100	100	100
	Total	6640	100	100	100

---

## 8 Frequenztabellen visualisieren

### 8.1 Präfixe erstellen

```
prefix.normen <- paste0(basename(dir.analysis),
                        "/",
                        config$project$shortname,
                        "_01_Einzelnormen_Frequenztafel_var-")

prefix.rechtsakte <- paste0(basename(dir.analysis),
                            "/",
                            config$project$shortname,
                            "_01_Rechtsakte_Frequenztafel_var-")

prefix.meta <- paste0(basename(dir.analysis),
                     "/",
                     config$project$shortname,
                     "_01_Meta_Frequenztafel_var-")
```

### 8.2 Tabellen für Einzelnormen einlesen

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

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

### 8.3 Tabellen für Rechtsakte einlesen

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

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

### 8.4 Tabellen für XML-Metadaten einlesen

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

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

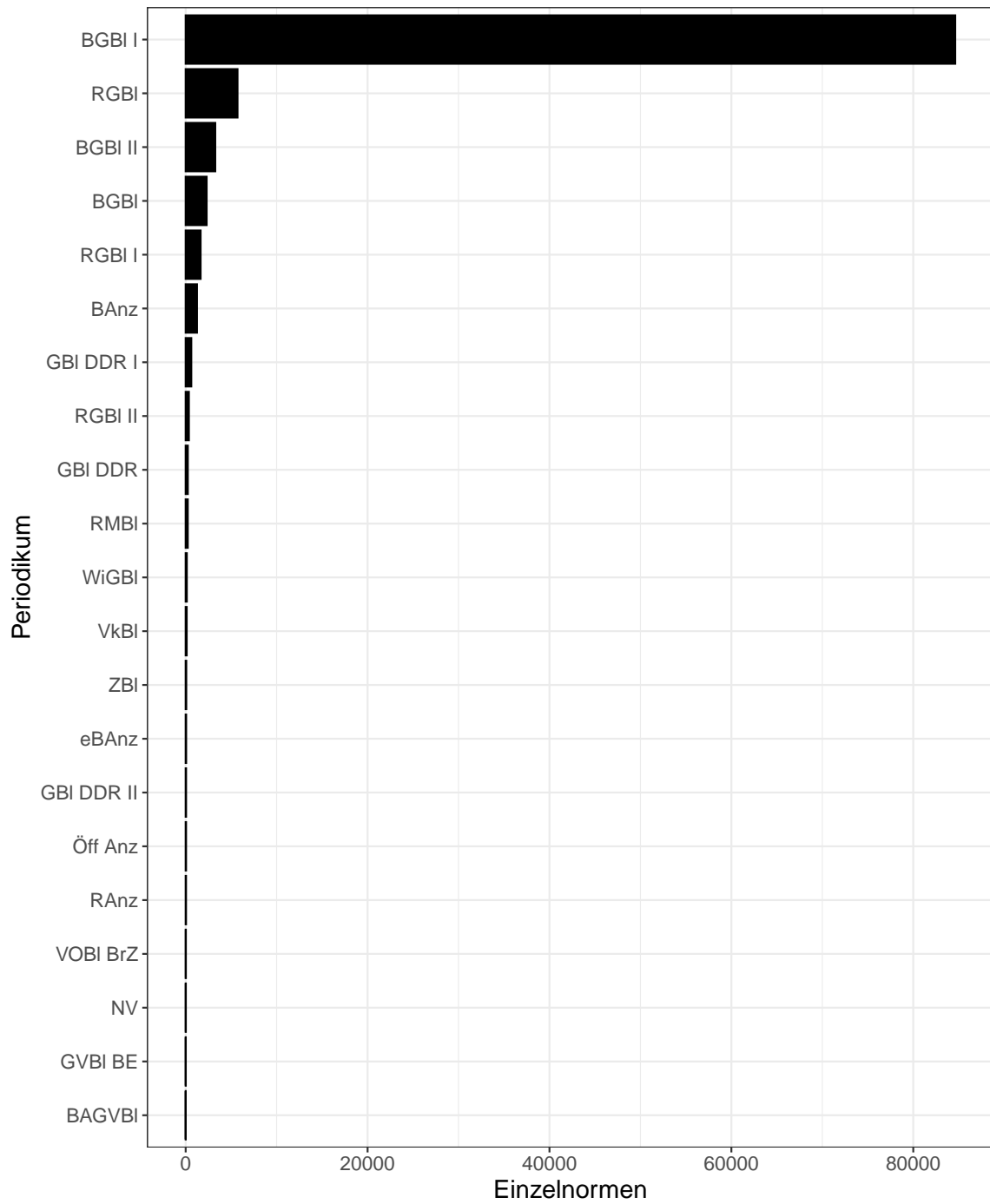
## 8.5 Periodikum

### 8.5.1 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(prefix.figuretitle,  
                  "| Einzelnormen je Periodikum"),  
    caption = caption,  
    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 2022-01-12 | Einzelnormen je Periodikum



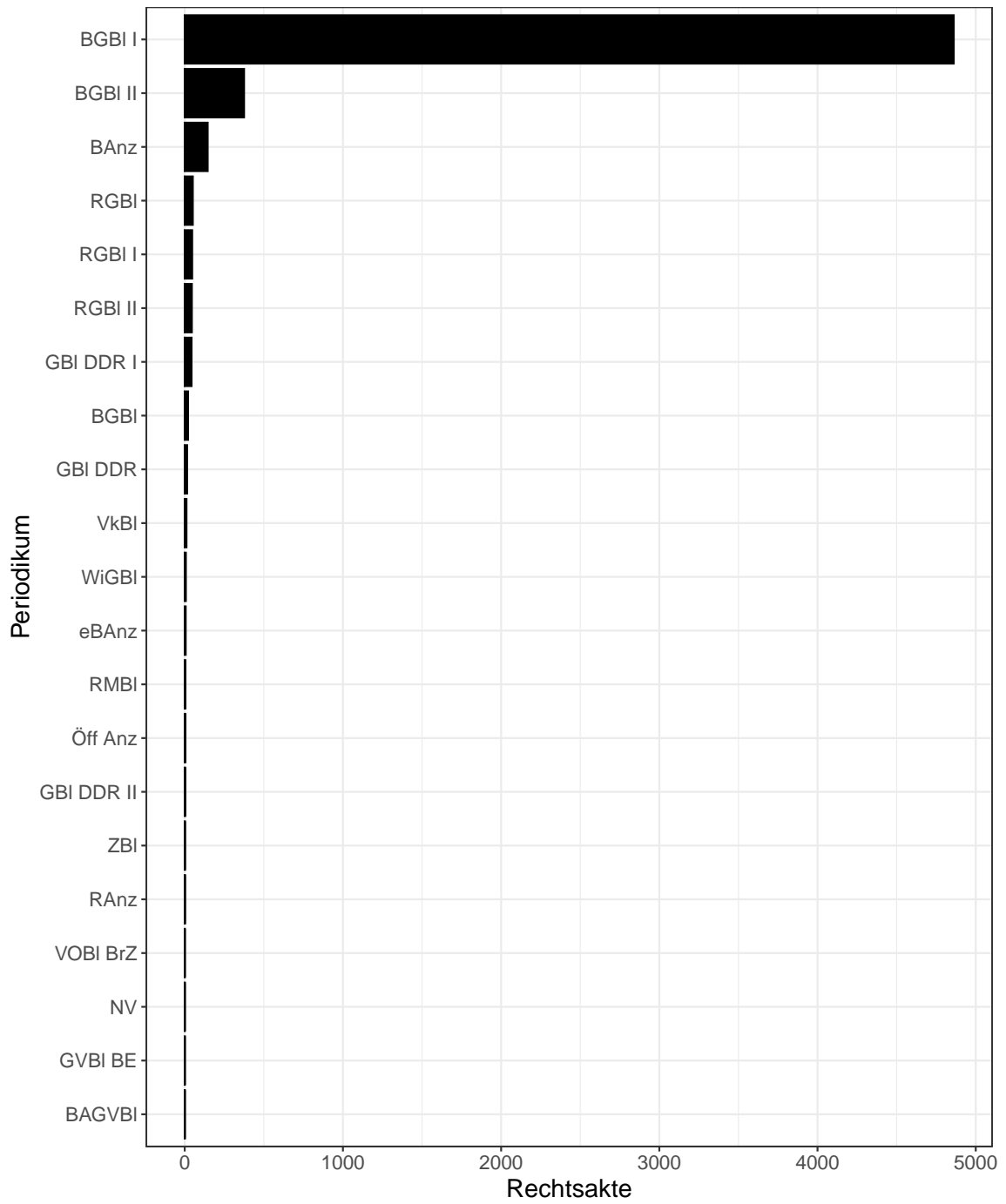
Fobbe | DOI: 10.5281/zenodo.5832094

## 8.5.2 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(prefix.figuretitle,  
                  "| Rechtsakte mit Inhalt je Periodikum"),  
    caption = caption,  
    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 2022-01-12 | Rechtsakte mit Inhalt je Periodikum



Fobbe | DOI: 10.5281/zenodo.5832094

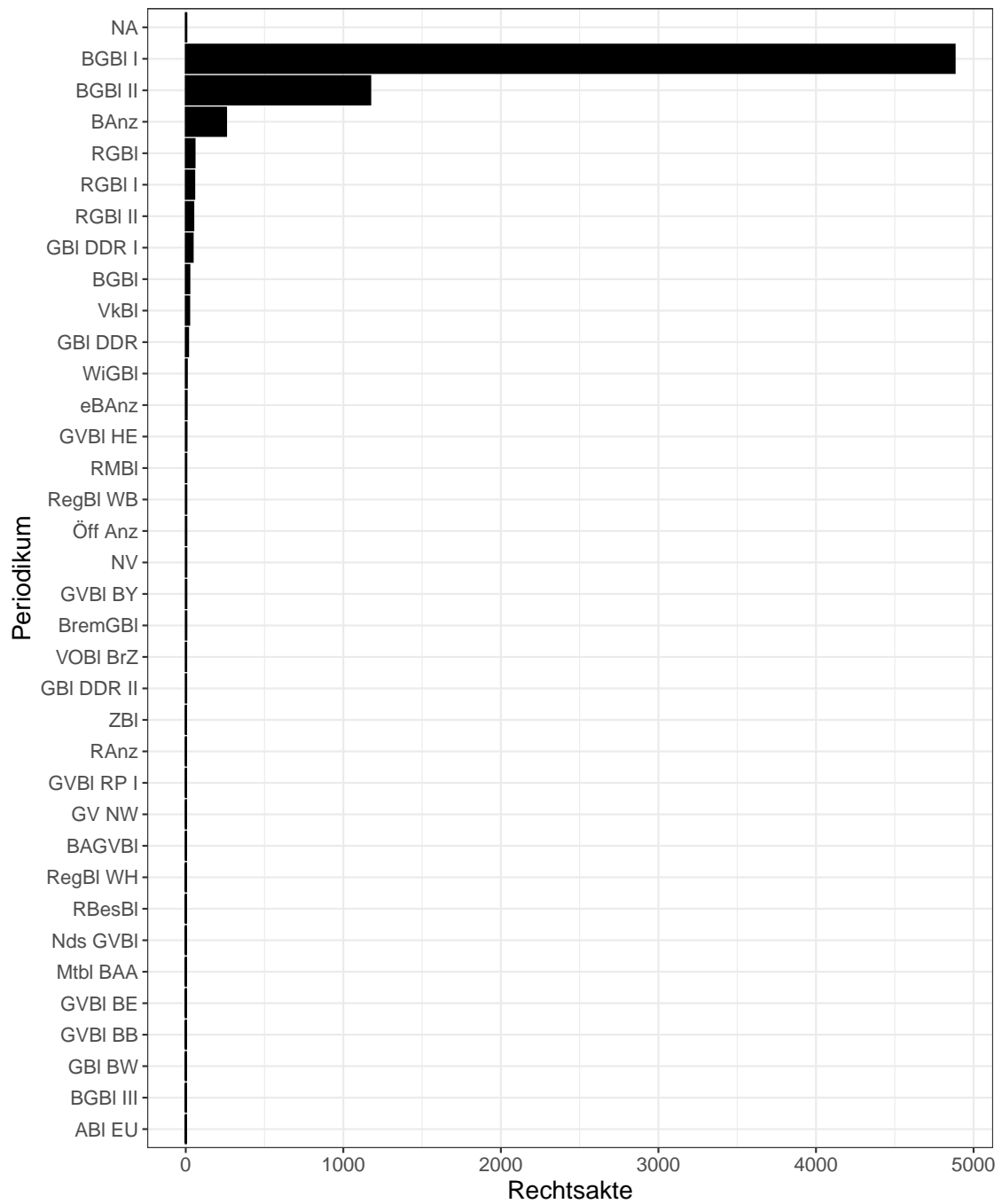
### 8.5.3 XML-Metadaten

```
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(prefix.figuretitle,  
                  "| Rechtsakte nach Metadaten je Periodikum"),  
    caption = caption,  
    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 2022-01-12 | Rechtsakte nach Metadaten je Periodikum



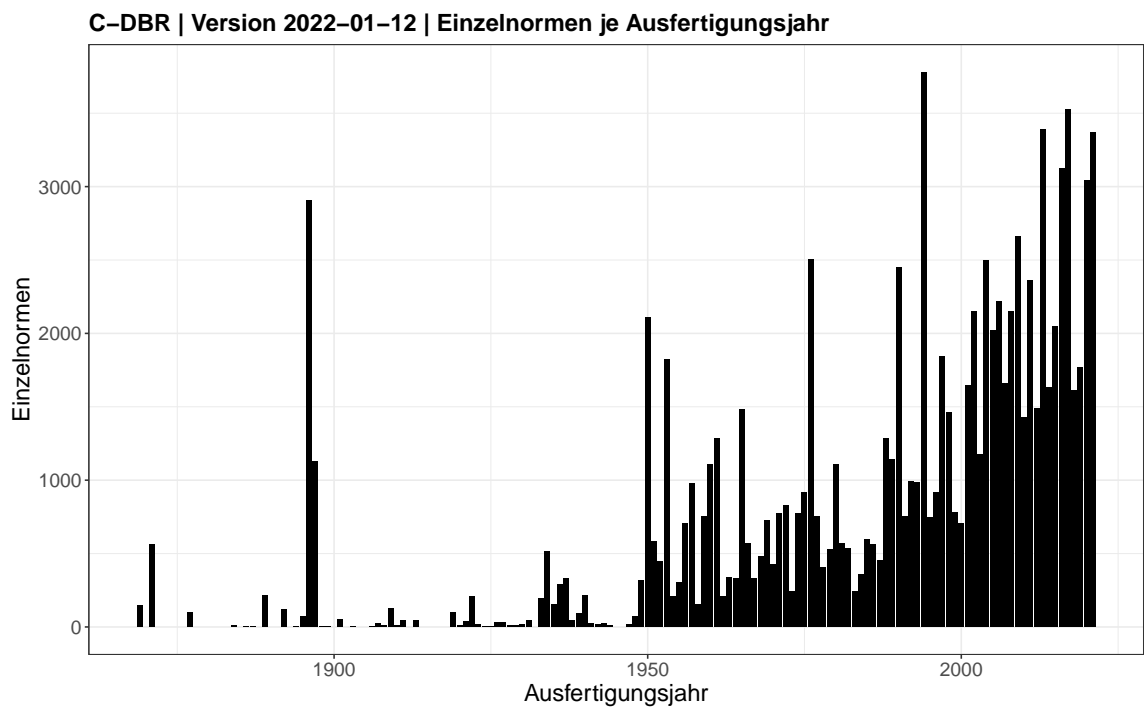
Fobbe | DOI: 10.5281/zenodo.5832094

## 8.6 Ausfertigungsjahr

### 8.6.1 Einzelnormen

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

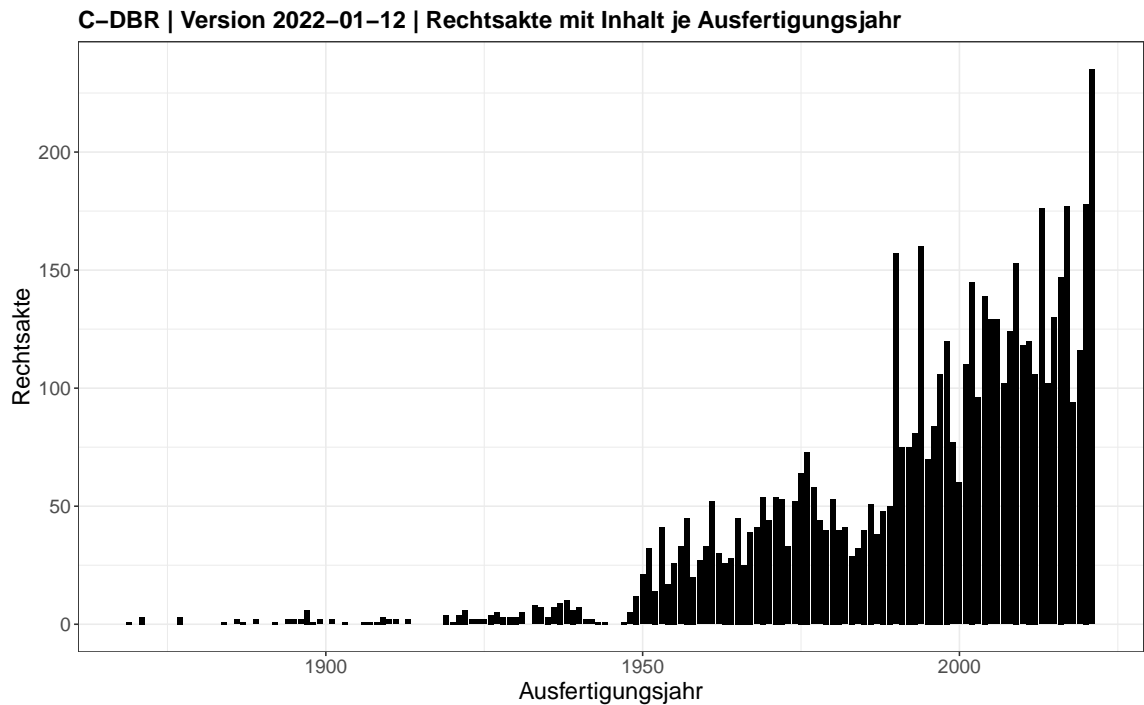
```
ggplot(data = freqtable) +  
  geom_bar(aes(x = ausfertigung_jahr,  
              y = N),  
          stat = "identity",  
          fill = "black")+  
  theme_bw()+  
  labs(  
    title = paste(prefix.figuretitle,  
                  "| Einzelnormen je Ausfertigungsjahr"),  
    caption = caption,  
    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)  
  )
```



## 8.6.2 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(prefix.figuretitle,  
                  "| Rechtsakte mit Inhalt je Ausfertigungsjahr"),  
    caption = caption,  
    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)  
  )
```

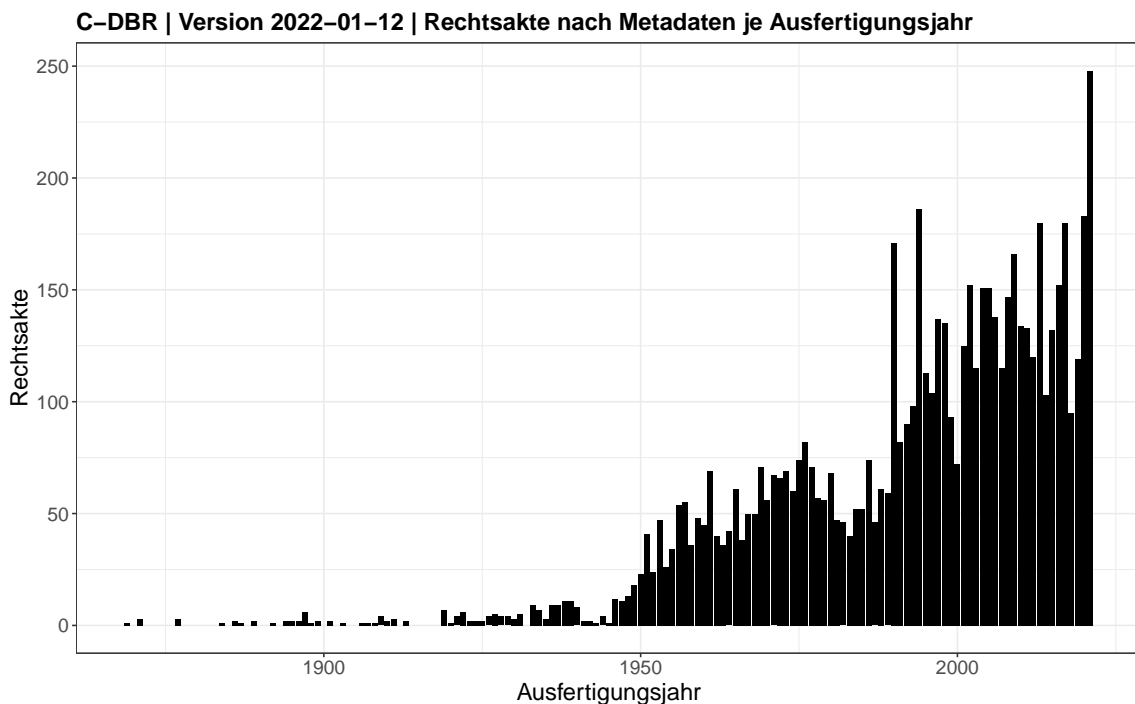


### 8.6.3 XML-Metadaten

```
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(prefix.figuretitle,  
                  "| Rechtsakte nach Metadaten je Ausfertigungsjahr"),  
    caption = caption,  
    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).
```



Fobbe | DOI: 10.5281/zenodo.5832094

## 9 Korpus-Analytik

### 9.1 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.

#### 9.1.1 Funktion anzeigen: `future_lingsummarize`

```
print(future_lingsummarize)
```

```
## function(dt,
##           chunkswerker = 1,
##           chunksize = NULL){
##
##   begin.dopar <- Sys.time()
##
##   dt <- dt[,.(doc_id, text)]
##
##   nchars <- dt[, lapply(.text), nchar)]
##
##   print(paste0("Processing ",
##               dt[,.N],
##               " documents with a total length of ",
##               sum(nchars),
##               " characters."))
##
##   ord <- order(-nchars)
##   dt <- dt[ord]
##
##   raw.list <- split(dt, seq(nrow(dt)))
##
##   result.list <- future_lapply(raw.list,
##                                lingsummarize,
##                                future.seed = TRUE,
##                                future.scheduling = chunkswerker,
##                                future.chunk.size = chunksize)
##
##   result.dt <- rbindlist(result.list)
##
##
##   end.dopar <- Sys.time()
##   duration.dopar <- end.dopar - begin.dopar
##
##
##   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)
## }

```

```

#deprecated; Parallelisierung jetzt mit futures
#lingstats.normen.raw <- f.lingsummarize.iterator(dt.normen,
#                                               threads = fullCores,
#                                               chunksize = 1)#

#deprecated; Parallelisierung jetzt mit futures
#lingstats.rechtsakte.raw <- f.lingsummarize.iterator(dt.rechtsakte,
#                                                    threads = fullCores,
#                                                    chunksize = 1)

```

### 9.1.2 Berechnung durchführen

```

if(config$parallel$lingsummarize == TRUE){
  plan("multicore",
       workers = fullCores)
}else{
  plan("sequential")
}

```

```
}
```

```
lingstats.normen.raw <- future_lingsummarize(dt.normen)
```

```
lingstats.rechtsakte.raw <- future_lingsummarize(dt.rechtsakte)
```

## 9.2 Variablen-Namen anpassen

### 9.2.1 Einzelnormen

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

### 9.2.2 Rechtsakte

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

## 9.3 Kennwerte den Korpora hinzufügen

### 9.3.1 Einzelnormen

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

### 9.3.2 Rechtsakte

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

## 9.4 Varianten mit Metadaten erstellen

### 9.4.1 Einzelnormen

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

### 9.4.2 Rechtsakte

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



## 9.5 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.

### 9.5.1 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"))
```

### 9.5.2 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	141,840,859	1	293	687	1,408.618690	1,483	398,079
tokens	22,194,452	1	46	109	220.412652	233	56,584
typen	418,839	1	36	66	92.922936	114	23,559
saetze	740,749	1	2	4	7.356363	8	2,381

### 9.5.3 Zusammenfassungen speichern

```
fwrite(dt.stats.ling,  
       paste0(dir.analysis,  
              config$project$shortname,  
              "_00_Einzelnormen_KorpusStatistik_ZusammenfassungLinguistisch.csv"),  
       ,  
       na = "NA")
```

## 9.6 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.

### 9.6.1 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"))
```

### 9.6.2 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	141,935,908	98	1,563.25	5,681.5	25,139.1973	22,517.75	2,084,129
tokens	22,194,452	17	264.25	917.0	3,931.0046	3,414.50	306,315
typen	418,839	15	129.00	302.5	641.4155	807.75	23,606
saetze	729,314	1	13.00	36.0	129.1736	114.00	14,306

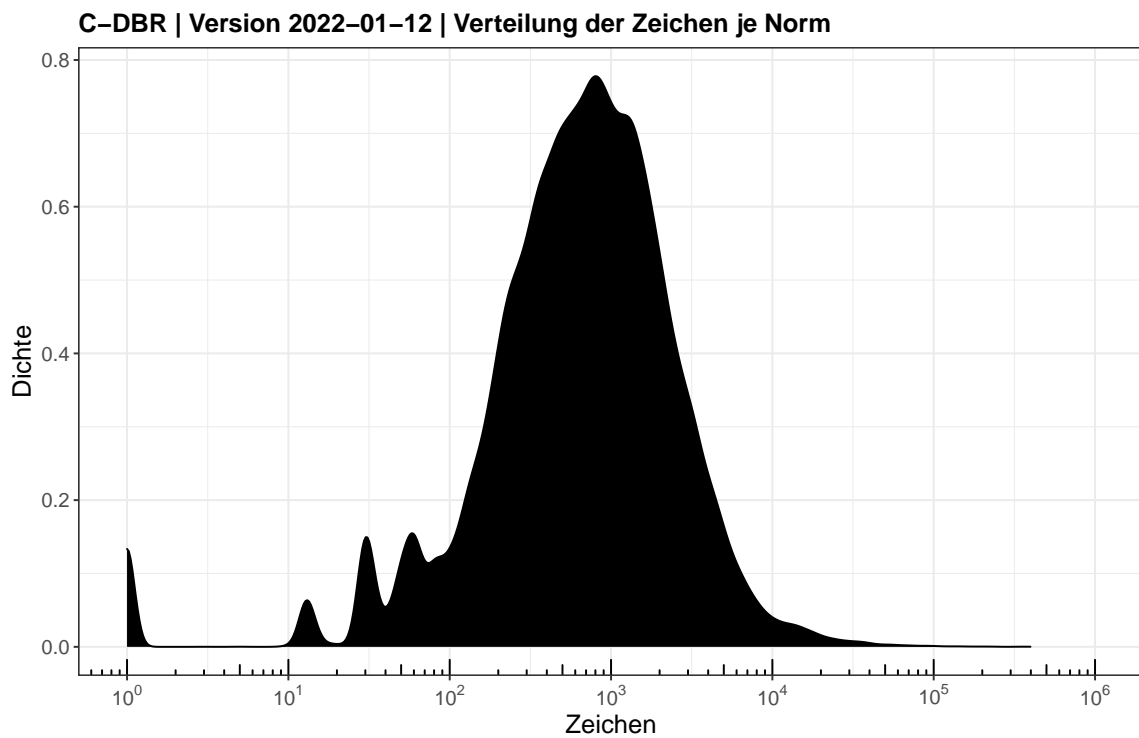
### 9.6.3 Zusammenfassungen speichern

```
fwrite(dt.stats.ling,  
       paste0(dir.analysis,  
              config$project$shortname,  
              "_00_Rechtsakte_KorpusStatistik_ZusammenfassungLinguistisch.csv"),  
       na = "NA")
```

## 9.7 Verteilungen

### 9.7.1 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(prefix.figuretitle,
                  "| Verteilung der Zeichen je Norm"),
    caption = caption,
    x = "Zeichen",
    y = "Dichte"
  )+
  theme(
    text = element_text(size = 14),
    plot.title = element_text(size = 14,
                               face = "bold"),
    legend.position = "none",
    plot.margin = margin(10, 20, 10, 10)
  )
)
```

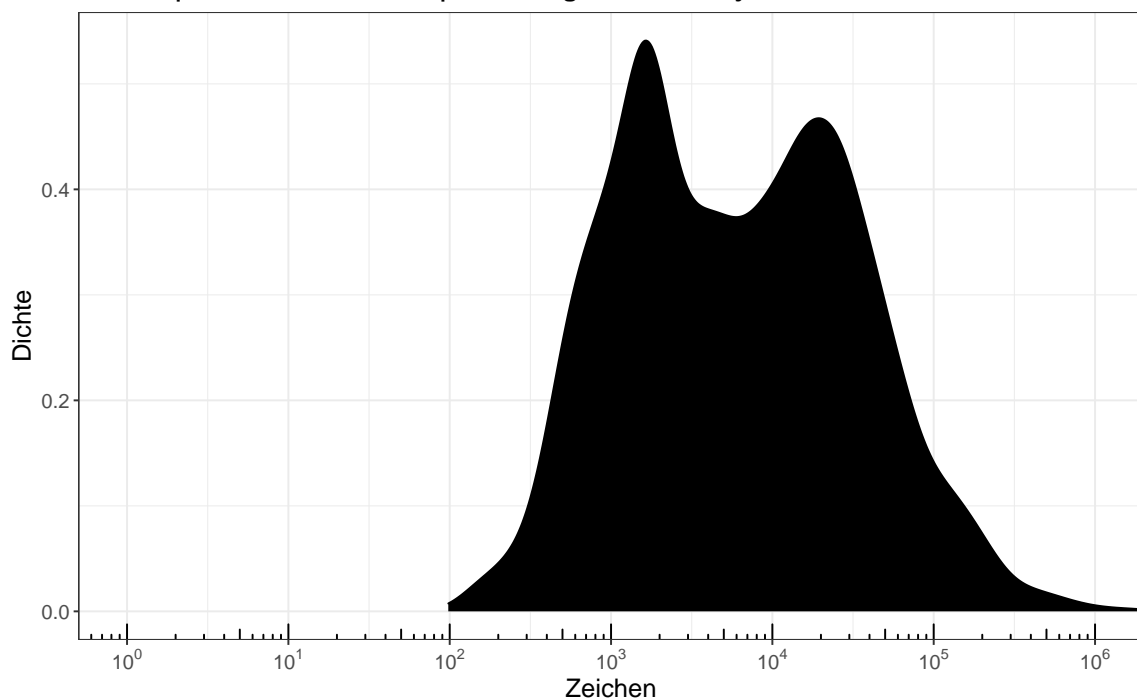


```

ggplot(data = meta.rechtsakte)+
  geom_density(aes(x = zeichen),
              fill = "black") +
  scale_x_log10(breaks = trans_breaks("log10", function(x) 10^x),
              labels = trans_format("log10", math_format(10^.x)))+
  annotation_logticks(sides = "b")+
  coord_cartesian(xlim = c(1, 10^6))+
  theme_bw()+
  labs(
    title = paste(prefix.figuretitle,
                  "| Verteilung der Zeichen je Rechtsakt"),
    caption = caption,
    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 2022-01-12 | Verteilung der Zeichen je Rechtsakt

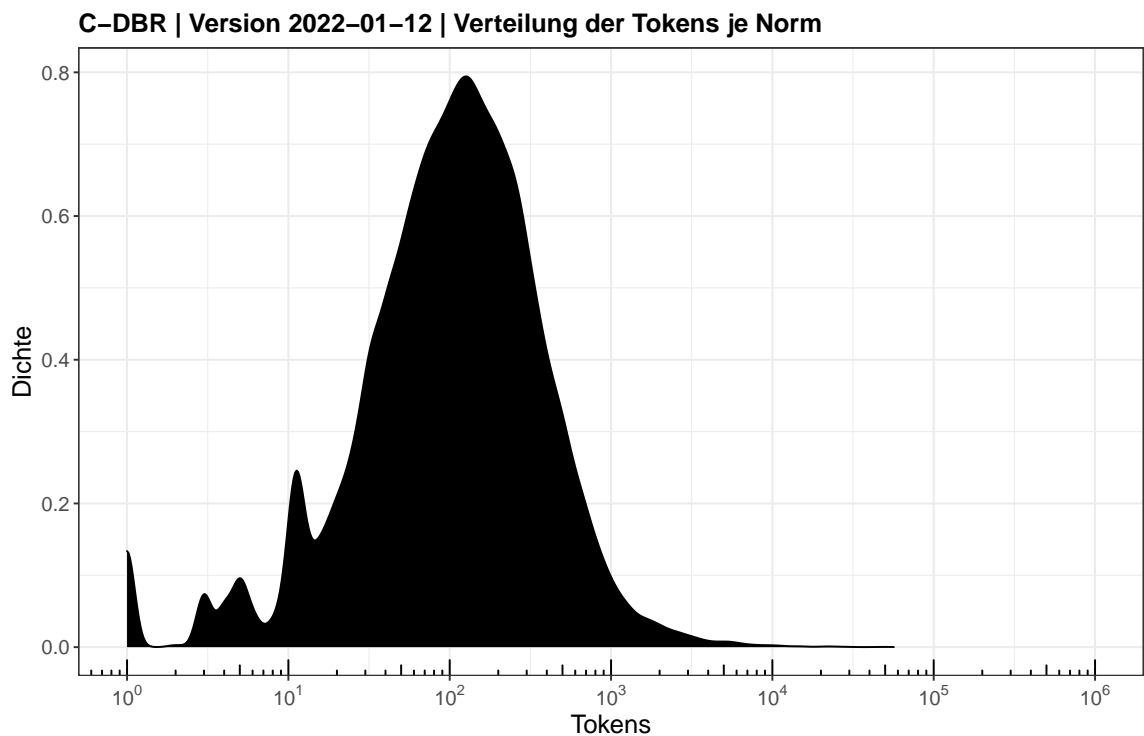


Fobbe | DOI: 10.5281/zenodo.5832094



## 9.7.2 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(prefix.figuretitle,
                  "| Verteilung der Tokens je Norm"),
    caption = caption,
    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)
  )
```



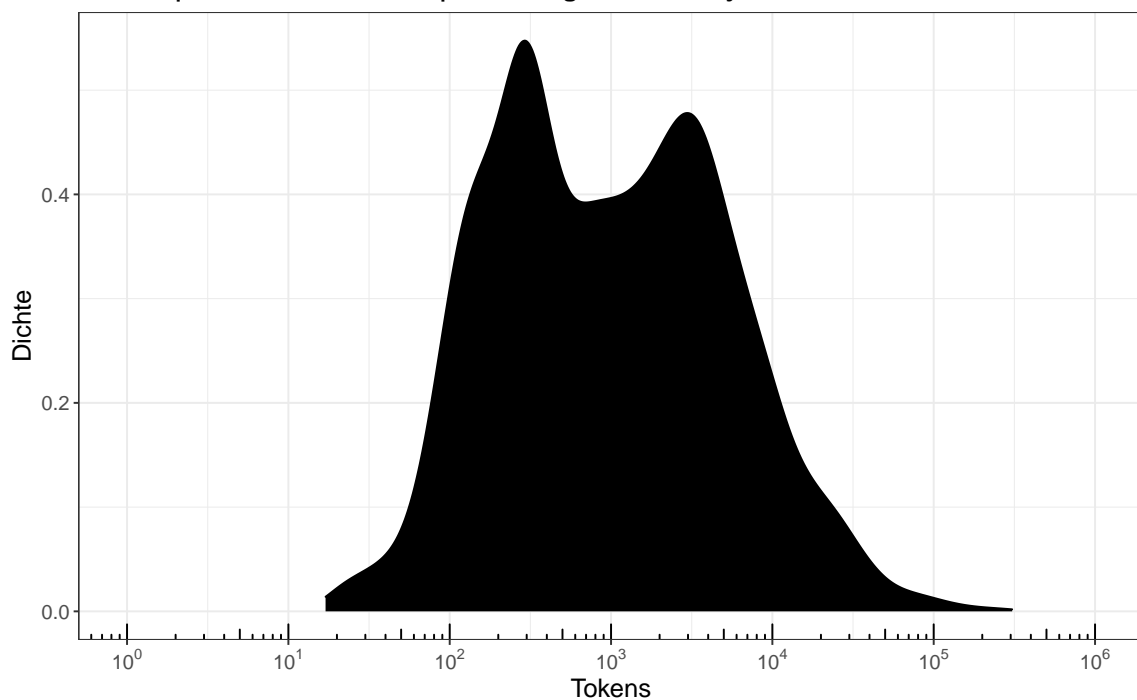
Fobbe | DOI: 10.5281/zenodo.5832094

```

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(prefix.figuretitle,
                  "| Verteilung der Tokens je Rechtsakt"),
    caption = caption,
    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)
  )

```

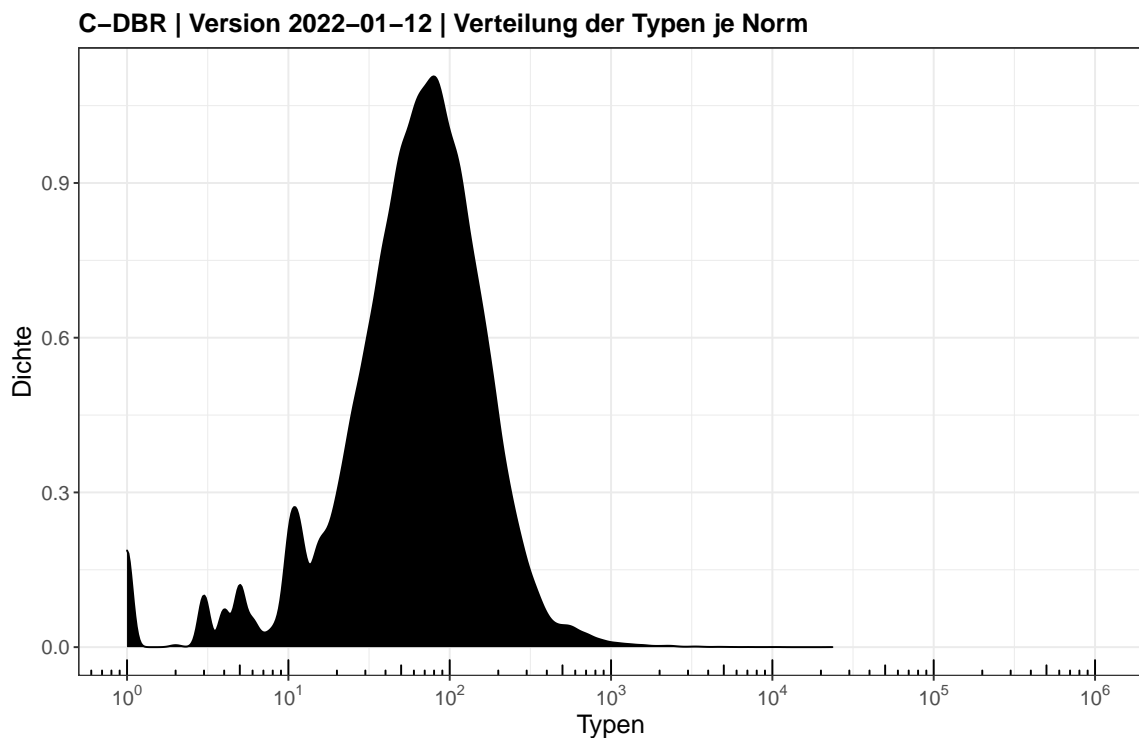
#### C-DBR | Version 2022-01-12 | Verteilung der Tokens je Rechtsakt



Fobbe | DOI: 10.5281/zenodo.5832094

### 9.7.3 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(prefix.figuretitle,
                  "| Verteilung der Typen je Norm"),
    caption = caption,
    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)
  )
)
```



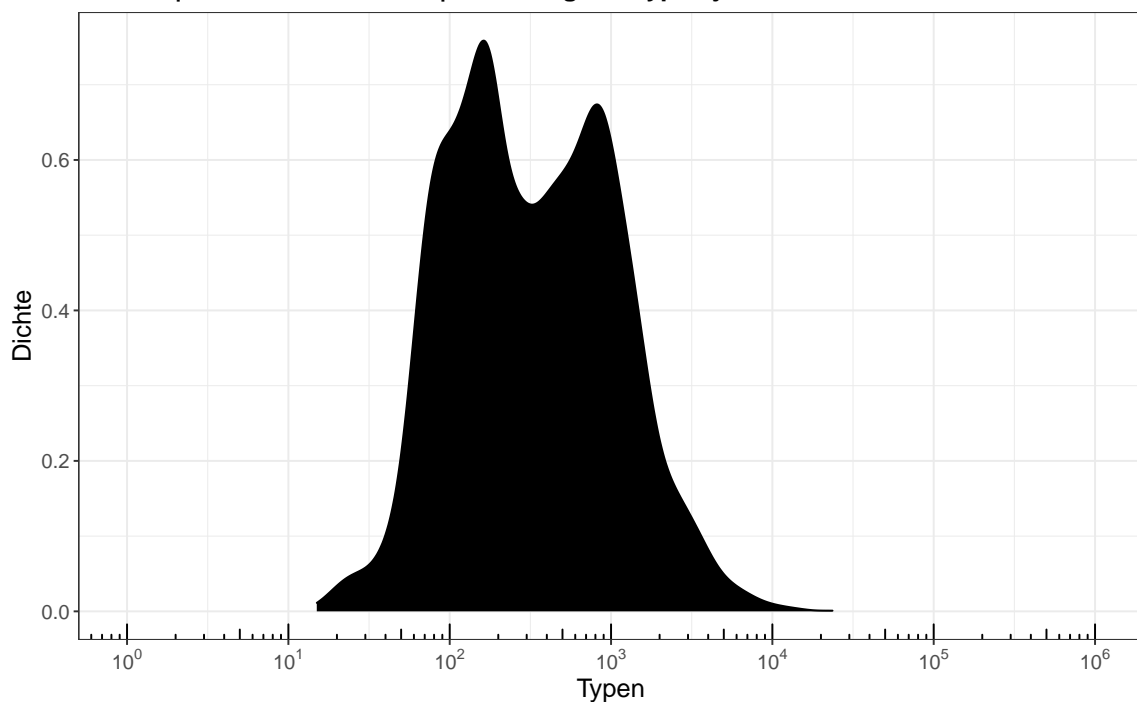
Fobbe | DOI: 10.5281/zenodo.5832094

```

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(prefix.figuretitle,
                  "| Verteilung der Typen je Rechtsakt"),
    caption = caption,
    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)
  )

```

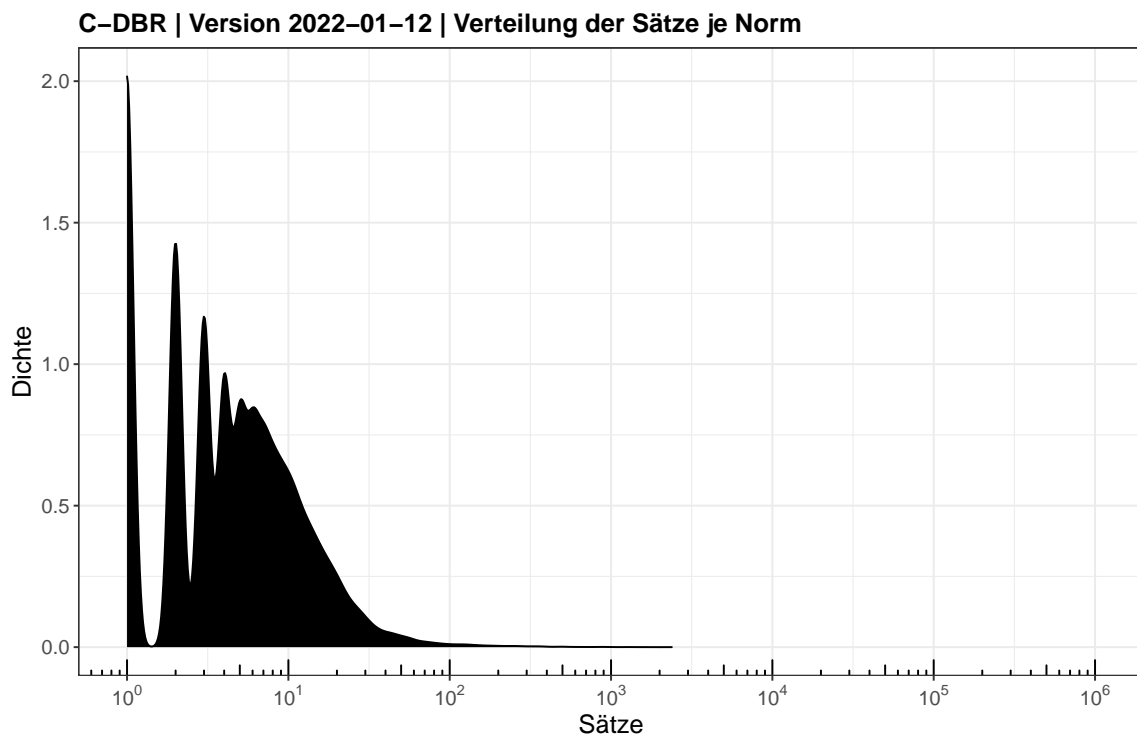
C-DBR | Version 2022-01-12 | Verteilung der Typen je Rechtsakt



Fobbe | DOI: 10.5281/zenodo.5832094

## 9.7.4 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(prefix.figuretitle,
                  "| Verteilung der Sätze je Norm"),
    caption = caption,
    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)
  )
```

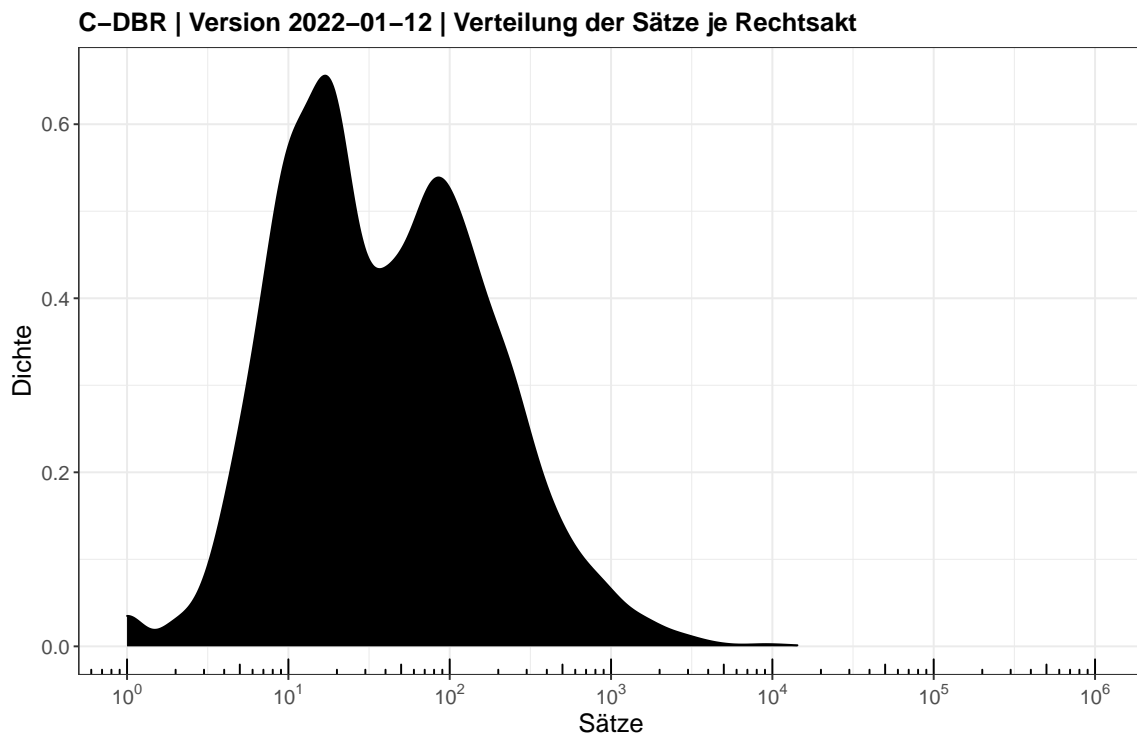


Fobbe | DOI: 10.5281/zenodo.5832094

```

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(prefix.figuretitle,
                  "| Verteilung der Sätze je Rechtsakt"),
    caption = caption,
    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)
  )

```



Fobbe | DOI: 10.5281/zenodo.5832094

## 9.8 Quantitative Variablen

### 9.8.1 Ausfertigungsdatum

#### *Einzelnormen*

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

```
##           Min.          1st Qu.          Median          Mean          3rd Qu.          Max.
## "1869-06-21" "1972-08-31" "1997-10-31" "1988-03-28" "2011-12-16" "2021-12-29"
```

### *Rechtsakte*

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

```
##           Min.          1st Qu.          Median          Mean          3rd Qu.          Max.
## "1869-06-21" "1981-10-16" "2001-02-04" "1995-02-01" "2012-07-21" "2021-12-29"
```

### *XML-Metadaten*

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

```
##           Min.          1st Qu.          Median          Mean          3rd Qu.          Max.
## "1869-06-21" "1977-12-25" "1998-03-03" "1993-04-01" "2010-12-15" "2021-12-29"
##           NA's
##           "1"
```

## 9.8.2 Ausfertigungsjahr

### *Einzelnormen*

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

```
##           Min. 1st Qu.  Median  Mean 3rd Qu.  Max.
##           1869   1972   1997   1988   2011   2021
```

### *Rechtsakte*

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

```
##           Min. 1st Qu.  Median  Mean 3rd Qu.  Max.
##           1869   1981   2001   1995   2012   2021
```

### *XML-Metadaten*

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

##	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	NA's
##	1869	1977	1998	1993	2010	2021	1



## 10 Strenge Kontrolle der Variablen-Namen

### 10.1 Semantische Sortierung der Variablen

#### 10.1.1 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"))
```

## 10.1.2 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"))
```

### 10.1.3 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"))
```

## 10.2 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
```

### 10.3 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"

```



## 11 CSV-Dateien erstellen

### 11.1 Einzelnormen (Korpus)

#### 11.1.1 Name für CSV definieren

```
csvname.normen.gesamt <- paste0(prefix.files,  
                                "_DE_CSV_Einzelnormen_Datensatz.csv")
```

#### 11.1.2 Datensatz speichern

```
fwrite(dt.normen,  
       paste0(csvname.normen.gesamt),  
       na = "NA")
```

### 11.2 Einzelnormen (Metadaten)

#### 11.2.1 Name für CSV definieren

```
csvname.normen.meta <- paste0(prefix.files,  
                               "_DE_CSV_Einzelnormen_Metadaten.csv")
```

#### 11.2.2 Datensatz speichern

```
fwrite(meta.normen,  
       paste0(csvname.normen.meta),  
       na = "NA")
```

### 11.3 Rechtsakte (Korpus)

#### 11.3.1 Name für CSV definieren

```
csvname.rechtsakte.gesamt <- paste0(prefix.files,  
                                     "_DE_CSV_Rechtsakte_Datensatz.csv")
```

#### 11.3.2 Datensatz speichern

```
fwrite(dt.rechtsakte,  
       paste0(csvname.rechtsakte.gesamt),  
       na = "NA")
```

## 11.4 Rechtsakte (Metadaten)

### 11.4.1 Name für CSV definieren

```
csvname.rechtsakte.meta <- paste0(prefix.files,  
                                   "_DE_CSV_Rechtsakte_Metadaten.csv")
```

### 11.4.2 Datensatz speichern

```
fwrite(meta.rechtsakte,  
       paste0(csvname.rechtsakte.meta),  
       na = "NA")
```

## 11.5 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.

### 11.5.1 Name für CSV definieren

```
csvname.meta <- paste0(prefix.files,  
                       "_DE_CSV_MetadatenXML.csv")
```

### 11.5.2 Datensatz speichern

```
fwrite(dt.meta,  
       paste0(csvname.meta),  
       na = "NA")
```

## 12 Download der PDF-Dateien

### 12.1 Download durchführen

```
if(config$parallel$downloadPDF == TRUE){  
  plan("multicore",  
        workers = fullCores)  
}else{  
  plan("sequential")  
}
```

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

### 12.2 Download-Ergebnis

#### 12.2.1 Anzahl herunterzuladender Dateien

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

```
## [1] 6640
```

#### 12.2.2 Anzahl heruntergeladener Dateien

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

```
## [1] 6640
```

#### 12.2.3 Fehlbetrag

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

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

## 12.2.4 Fehlende Dateien

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

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

## 13 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)
```

### 13.1 Anzahl zu extrahierender Dateien

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

```
## [1] 6640
```

### 13.2 Funktion anzeigen: future\_pdf\_to\_txt

```
print(future_pdf_to_txt)
```

```
## function(x){
##
##   ## Timestamp: Begin
##   begin.extract <- Sys.time()
##
##   ## Intro message
##   message(paste("Processing",
##                 length(x),
##                 "files. Begin at:",
##                 begin.extract))
##
##   ## Perform conversion from PDF to TXT
##   invisible(future.apply::future_lapply(x,
##                                         pdf_to_txt,
##                                         future.seed = TRUE))
##
##   ## Construct full list of TXT names
##   txt.names <- gsub("\\*.pdf",
##                    "\\*.txt",
##                    x,
##                    ignore.case = TRUE)
##
##   ## Check list of TXT files in folder
##   txt.results <- list.files(pattern = "\\*.txt")
```

```

##
##   ## Compare full list to files in folder
##   txt.missing <- setdiff(txt.names,
##                           txt.results)
##
##   ## Timestamp: End
##   end.extract <- Sys.time()
##
##   ## Duration
##   duration.extract <- end.extract - begin.extract
##
##   ## Outro message
##   message(paste0("Successfully processed ",
##                  length(x) - length(txt.missing),
##                  " files. ",
##                  length(txt.missing),
##                  " files failed. Runtime was ",
##                  round(duration.extract,
##                          digits = 2),
##                  " ",
##                  attributes(duration.extract)$units,
##                  ". Ended at: ",
##                  end.extract))
##
##
## }

```

### 13.3 Text Extrahieren

```

if(config$parallel$extractPDF == TRUE){
  plan("multicore",
        workers = fullCores)
}else{
  plan("sequential")
}

future_pdf_to_txt(files.pdf)

```

```
## Processing 6640 files. Begin at: 2022-01-12 14:25:57
```

```
## Successfully processed 6640 files. 0 files failed. Runtime was 20.54 secs.
## Ended at: 2022-01-12 14:26:18
```

## 14 Download der EPUB-Dateien

### 14.1 Download durchführen

```
if(config$parallel$downloadEPUB == TRUE){  
  plan("multicore",  
        workers = fullCores)  
}else{  
  plan("sequential")  
}
```

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

### 14.2 Download-Ergebnis

#### 14.2.1 Anzahl herunterzuladender Dateien

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

```
## [1] 6640
```

#### 14.2.2 Anzahl heruntergeladener Dateien

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

```
## [1] 6640
```

#### 14.2.3 Fehlbetrag

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

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

#### 14.2.4 Fehlende Dateien

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

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



## 15 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
```

### 15.1 Gesamtgröße

#### 15.1.1 PDF-Dateien (MB)

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

```
## [1] 633.544
```

#### 15.1.2 EPUB-Dateien (MB)

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

```
## [1] 448.8979
```

#### 15.1.3 XML-Dateien (MB)

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

```
## [1] 266.0645
```

#### 15.1.4 TXT-Dateien (MB)

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

```
## [1] 199.7343
```

#### 15.1.5 Objekte in RAM (MB)

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

```
## 194.5 MB
```

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

```
## 149.7 MB
```

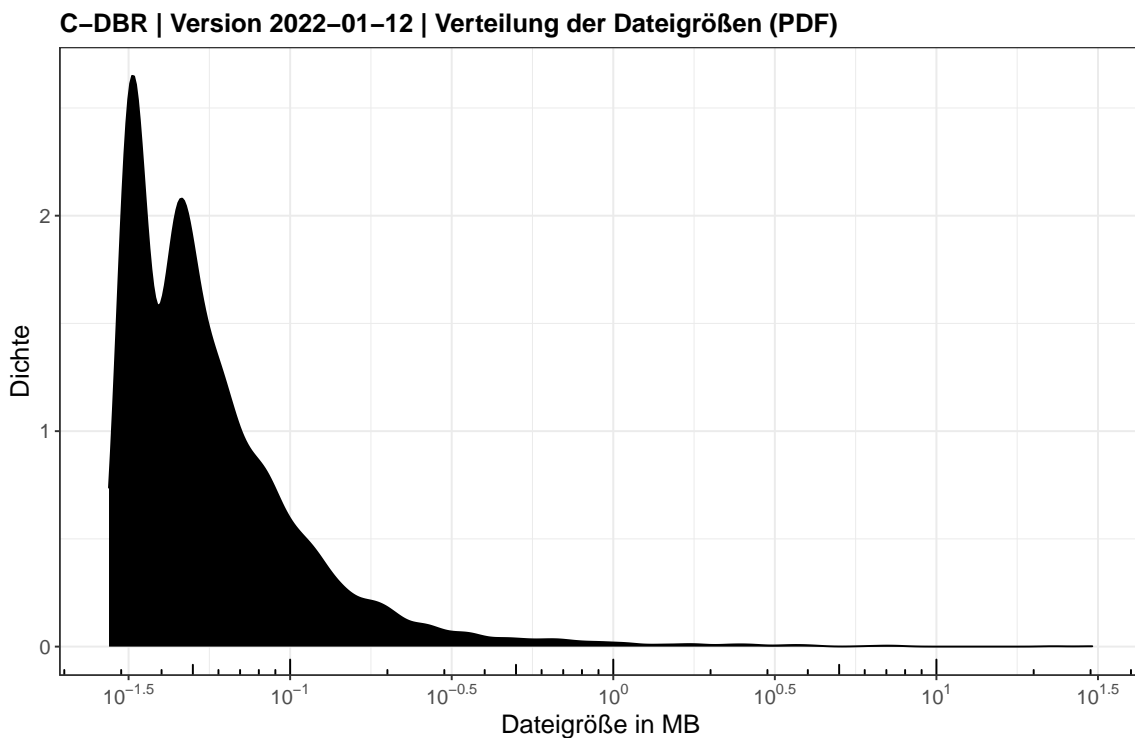
```
print(object.size(dt.meta),  
      standard = "SI",  
      humanReadable = TRUE,  
      units = "MB")
```

```
## 5.7 MB
```

## 15.2 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(prefix.figuretitle,  
      "| Verteilung der Dateigrößen (PDF)",  
    caption = caption,  
    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)  
  )
```

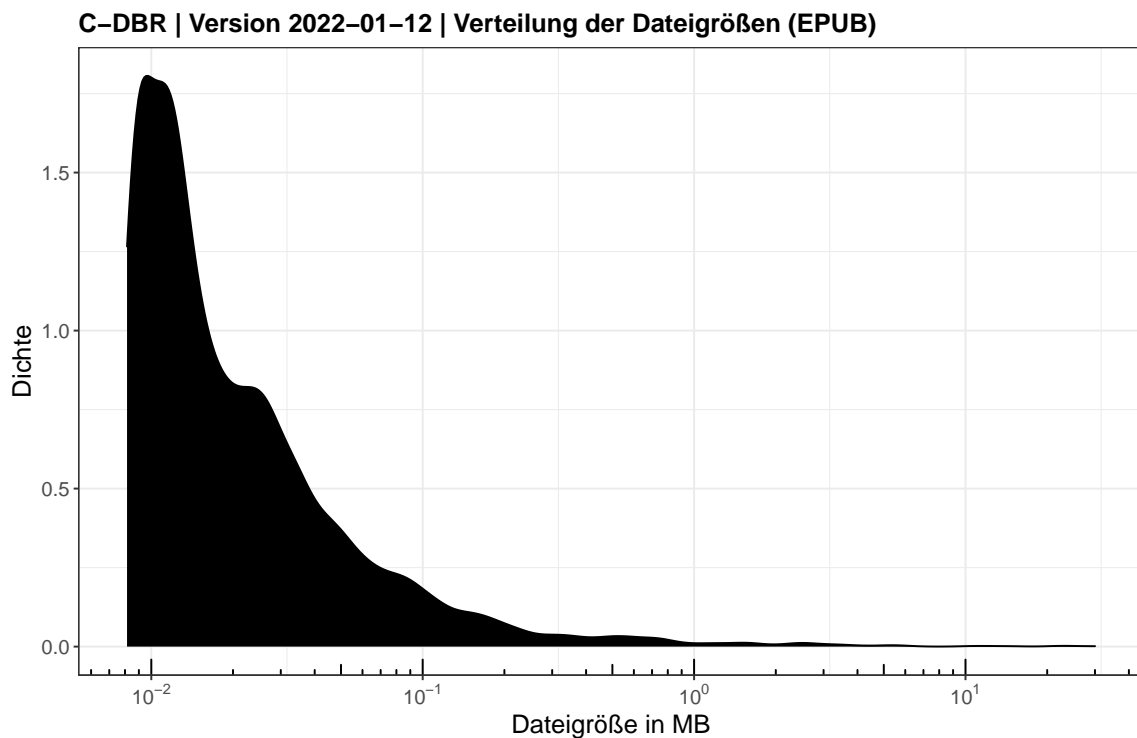


Fobbe | DOI: 10.5281/zenodo.5832094

## 15.3 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(prefix.figuretitle,  
      "| Verteilung der Dateigrößen (EPUB)"),  
    caption = caption,  
    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)  
  )
```

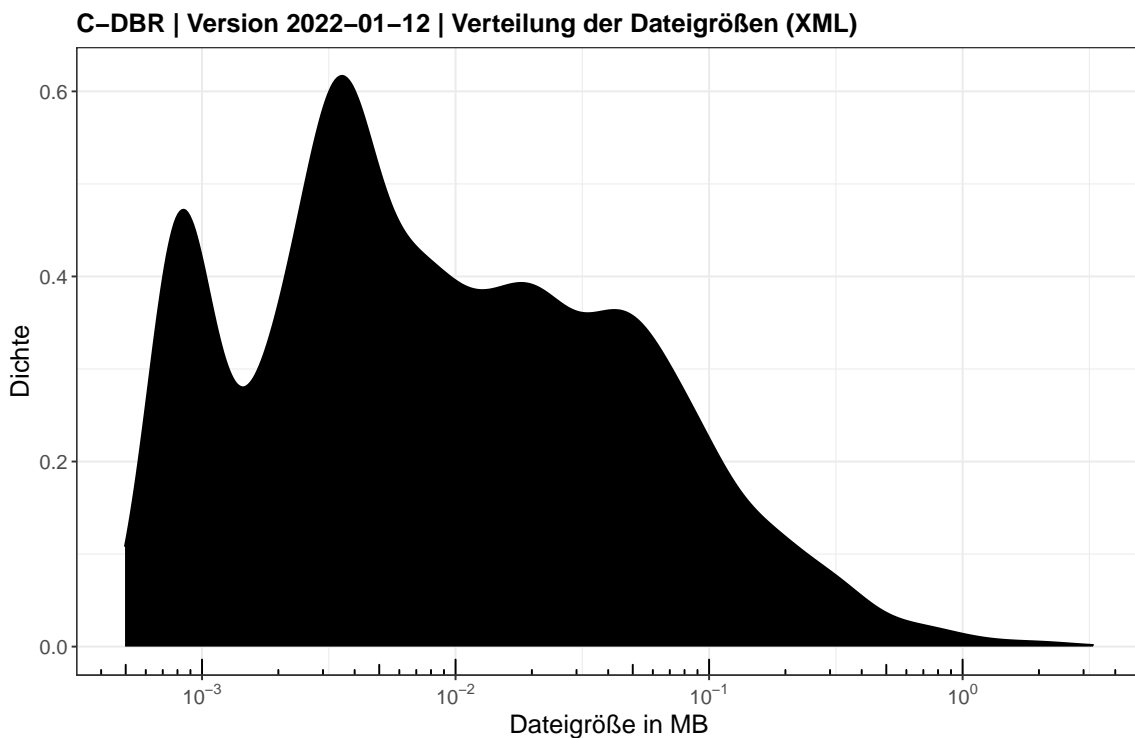


Fobbe | DOI: 10.5281/zenodo.5832094

## 15.4 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(prefix.figuretitle,  
      "| Verteilung der Dateigrößen (XML)",  
    caption = caption,  
    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)  
  )
```

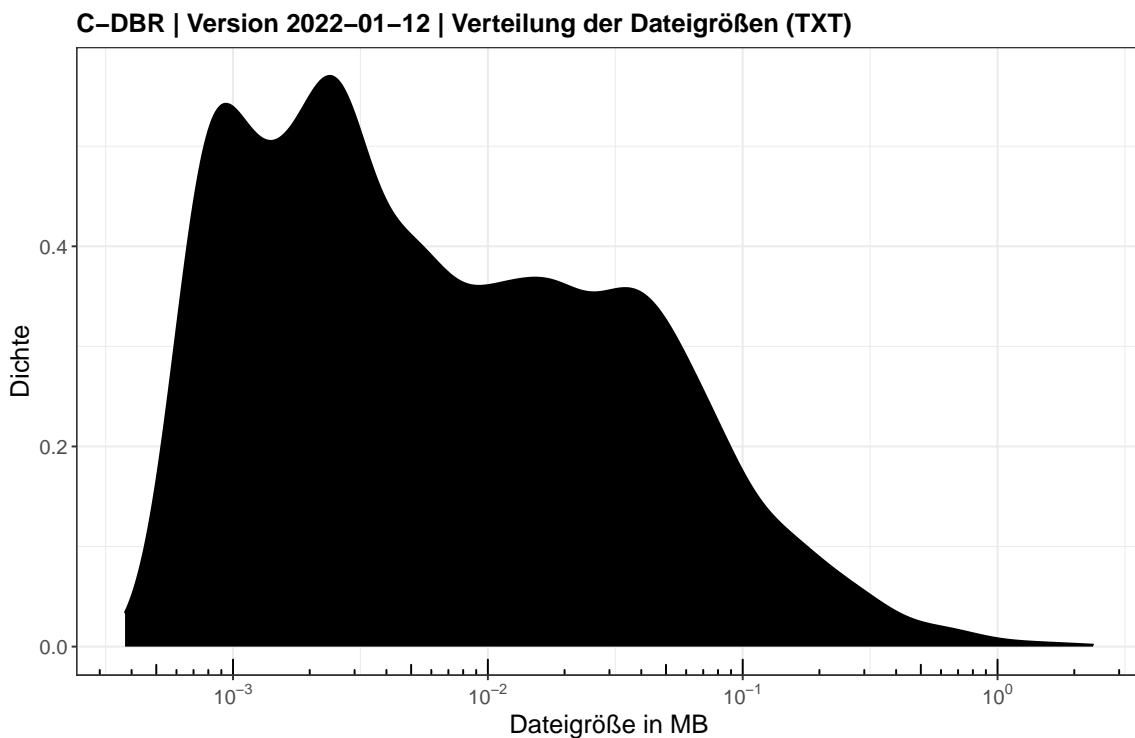


Fobbe | DOI: 10.5281/zenodo.5832094

## 15.5 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(prefix.figuretitle,  
      "| Verteilung der Dateigrößen (TXT)",  
    caption = caption,  
    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)  
  )  
)
```



Fobbe | DOI: 10.5281/zenodo.5832094

## 16 ZIP-Archive erstellen

### 16.1 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],  
      mode = "cherry-pick")  
}  
  
unlink(files.csv)
```

### 16.2 Verpacken der PDF-Dateien

```
files.pdf <- list.files(pattern = "\\..pdf$",  
                       ignore.case = TRUE)  
  
zip(paste0(prefix.files,  
           "_DE_PDF_Datensatz.zip"),  
    files.pdf,  
    mode = "cherry-pick")
```

### 16.3 Verpacken der TXT-Dateien

```
files.txt <- list.files(pattern = "\\..txt$",  
                       ignore.case = TRUE)  
  
zip(paste0(prefix.files,  
           "_DE_TXT_Datensatz.zip"),  
    files.txt,  
    mode = "cherry-pick")
```

### 16.4 Verpacken der EPUB-Dateien

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

zip(paste0(prefix.files,
           "_DE_EPUB_Datensatz.zip"),
    files.epub,
    mode = "cherry-pick")
```

## 16.5 Verpacken der Netzwerk-Dateien

```
zip(paste0(prefix.files,
           "_DE_Netzwerke.zip"),
    "netzwerke",
    mode = "cherry-pick")
```

## 16.6 Verpacken der Analyse-Dateien

```
zip(paste0(prefix.files,
           "_DE_",
           toupper(basename(dir.analysis)),
           ".zip"),
    basename(dir.analysis),
    mode = "cherry-pick")
```

## 16.7 Verpacken der Source-Dateien

```
files.source <- c(list.files(pattern = "\\\\.R$|\\.toml$",
                            "R-fobbe-proto-package",
                            "functions",
                            "tex",
                            "gpg",
                            "buttons",
                            list.files(pattern = "renv\\.lock|\\.Rprofile",
                                        all.files = TRUE),
                            list.files("renv",
                                        pattern = "activate\\.R",
                                        full.names = TRUE)))

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

zip(paste0(prefix.files,
```



```
        "_Source_Code.zip"),  
files.source,  
mode = "cherry-pick")
```

## 17 Roh-Dateien löschen

```
files.delete <- list.files(pattern = "\\..xml|\\.jpe?g|\\.png|\\.gif|\\.pdf|\\.txt  
|\\.epub",  
                           ignore.case = TRUE)  
  
unlink(files.delete)  
unlink("netzwerke", recursive = TRUE)  
unlink("Rplots.pdf", recursive = TRUE)
```

## 18 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.

### 18.1 Liste der ZIP-Archive erstellen

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

### 18.2 Funktion anzeigen: future\_multihashes

```
print(future_multihashes)
```

```
## function(x){  
##  
##   ## Timestamp: Begin  
##   begin <- Sys.time()  
##  
##   ## Intro Message  
##   message(paste("Processing",  
##                 length(x),  
##                 "files. Begin at:",  
##                 begin))  
##  
##   ## Compute Hashes  
##   hashes.list <- future.apply::future_lapply(x,  
##                                             multihashes)  
##  
##   ## Coerce List to data.table  
##   hashes.table <- data.table::rbindlist(hashes.list)  
##  
##   ## Coerce data.table to data.frame  
##   data.table::setDF(hashes.table)  
##  
##   ## Timestamp: End  
##   end <- Sys.time()  
##  
##   ## Duration  
##   duration <- end - begin  
##  
##   ## Result Message  
##   message(paste0("Processed ",
```

```

##           length(x),
##           " files. Runtime was ",
##           round(duration,
##             digits = 2),
##           " ",
##           attributes(duration)$units,
##           ".")
##
##   return(hashses.table)
##
## }

```

### 18.3 Hashes berechnen

```

if(config$parallel$multihashes == TRUE){
  plan("multicore",
       workers = fullCores)
}else{
  plan("sequential")
}

multihashes <- future_multihashes(files.zip)

```

```
## Processing 13 files. Begin at: 2022-01-12 14:37:07
```

```
## Processed 13 files. Runtime was 3.22 secs.
```

### 18.4 In Data Table umwandeln

```

setDT(multihashes)

setnames(multihashes,
        old = "x",
        new = "filename")

```

### 18.5 Index hinzufügen

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

## 18.6 Hashes in CSV-Datei speichern

```
fwrite(multihashes,  
       paste0("output/",  
              prefix.files,  
              "_KryptographischeHashes.csv"),  
       na = "NA")
```

## 18.7 Leerzeichen hinzufügen um Zeilenumbruch zu ermöglichen

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

## 18.8 In Bericht anzeigen

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

---

index	filename
1	C-DBR_2022-01-12_DE_ANALYSE.zip
2	C-DBR_2022-01-12_DE_CSV_Einzelnormen_Datensatz.zip
3	C-DBR_2022-01-12_DE_CSV_Einzelnormen_Metadaten.zip
4	C-DBR_2022-01-12_DE_CSV_MetadatenXML.zip
5	C-DBR_2022-01-12_DE_CSV_Rechtsakte_Datensatz.zip
6	C-DBR_2022-01-12_DE_CSV_Rechtsakte_Metadaten.zip
7	C-DBR_2022-01-12_DE_EPUB_Datensatz.zip
8	C-DBR_2022-01-12_DE_Netzwerke.zip
9	C-DBR_2022-01-12_DE_PDF_Datensatz.zip
10	C-DBR_2022-01-12_DE_TXT_Datensatz.zip
11	C-DBR_2022-01-12_DE_XML_Anlagen.zip
12	C-DBR_2022-01-12_DE_XML_Datensatz.zip
13	C-DBR_2022-01-12_Source_Code.zip

---

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

---

index	sha2.256
1	73b3e7ef5ba08199b158671939132c5329d413836fd73c2c888392a1e8ea055b
2	04ba8e44ff3cbfad3e5ab586629c43e3b8c10dee3b3b6951d8140154f404b59a
3	5657c2f68ef766a5c3d3800bb5a1a1bfb19fce8c6ebfb213621d6306f8498ac7
4	27b1853bdad86df58075ebeba24c71bf21cf3ccda99c2a5e86565e55ba099b0
5	84cee08a019ecb6873971a59022edde3d5c0cc675e2489424d6244f3e617c5a6
6	4eb770204d4a70e6eca7fe428603cf77fd0cfe459ef94deff3b35d01977bedae
7	5a5bdc673b8930565d96f443a592929ae6f0b74d0d760c8a3e437ff8bb595f6e
8	dfecdc0a3213d8a155ebcc63e7bc99f59a2e822659ef0289dd0f281ca2c6e7b1
9	91e93eb292d3231b13036ad7e995a6aa1e4980bcf141a8bd2f8865c1273bd40b
10	d3277386a7de8a16164ba68fea9a76e605237d966a1b2054e058d48d4edd0fc3
11	338b200f4a1c96f97cb7a36edcaca5e14f8be3caedc84edbc4e34d96d1447fec
12	d8d2b0b105e9317663d37e9189f29e41b8fed56d450a299f765453e2c44855f7
13	02fd00346896d4a3ff03e3d108db961ba97bf78b3bb8e1064dfd623c865ecb69

---

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

---

index	sha3.512
1	0fa3c98c86df141ec31926c9d041890a701bdad26a8fe0d9eadcf1153a7e88bd51c4bb9dae84ef5cf8fd07969b8c213eedb8a1ee0e29640d49aca3315777d1a7
2	4cf2805e255f90fa5c2762edf09e8a3d7406eb63e9cf16042a2e7af3933ca5fd023c18d440a73e70b9913cfb19c2e4b605655965d27f4ad44bcff850540ef335
3	0e63580e89a882666cb9299eb7de6d14e9a08e3ad25014aecf1d0839be36a80ecd5df7c594f4d1de761a10a71b3769ed5524bc9394841635358d082600155f2e
4	2368c3cc3afa9f9b52fa2225baa64055839ab15ec4c627a937f5f57e4f6a25cc3890cf002ed92ccff83c729560047e9979a5b189360d3cd61a31406964a10968
5	06e319459adeb10d894302a0f259b4c9784496576838dae69fff78dc6a6b380fd4f42e00b4de44d3dbbcef44c1f74f5b575bd4060ca516e96c7a669175569888
6	cbd87fed9976af07f49b78597a348c2a07e10a42b94beb66d880b229a0a8f303b0e6878f87af010a08eaf66c40fa1d29d7a3978713ce1501a9828739ade46b60
7	32869395d9265b0005da487ea8e024c00f3deda2f038984dc8363bb93713c75fa265b78c581c35f2b41557911340bf9f026a3ce8ff85ca4871802b9e8262c9f3
8	bb49ccf308e0f8041544619247d5619dbf11ecafc08528ec9da739e0d85ace2ba4041d553772fbdef3645f39cee5acb287826e49edc14c314fef9d59e7d8ded6
9	bcfa615174b0aa0f440b8969f0e902fd05c6f7646f1a48f5bb45b75d4005497a44c3df1c4ddd9bd992fa26e6b4fe2aee27c946e9059523470e6e68f3f837dcc
10	010e0b2073a331c3f0be948beb93048b607d929701e9bbd20ff36e8472cb8d0d03a95b3c86be59161406151994b33ea7395f51fbd19846b407659fbd34a06f7b
11	480ecfd746c8417e924689c698078a8b6d5a5450479d9027c5cd08a051c367a6af155b1b2e76ec1cf484e3ccc784807fe043c6b5b8f5d29f08bbb6210e876968
12	b87125568b3f70820b167ca258b9c23346eafd35fc387e78cb7939b0e8a5a941c4bf2ecb3f4ab544af67a980f32c48fc7815bf8919d76ab7f8780aab41a87863
13	5107f51dc985d1604863bb61d19d27620fd8db41b3506fb4746f2239cfc24dc25970eb669e26517c7b06863f1de32edd5a143dad117feeb437dc06961635e185

---



## 19 Aufräumen

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

cleanup.destination <- file.path("output",
                                cleanup)

print(cleanup)
```

```
## [1] "C-DBR_2022-01-12_DE_ANALYSE.zip"
## [2] "C-DBR_2022-01-12_DE_CSV_Einzelnormen_Datensatz.zip"
## [3] "C-DBR_2022-01-12_DE_CSV_Einzelnormen_Metadaten.zip"
## [4] "C-DBR_2022-01-12_DE_CSV_MetadatenXML.zip"
## [5] "C-DBR_2022-01-12_DE_CSV_Rechtsakte_Datensatz.zip"
## [6] "C-DBR_2022-01-12_DE_CSV_Rechtsakte_Metadaten.zip"
## [7] "C-DBR_2022-01-12_DE_EPUB_Datensatz.zip"
## [8] "C-DBR_2022-01-12_DE_Netzwerke.zip"
## [9] "C-DBR_2022-01-12_DE_PDF_Datensatz.zip"
## [10] "C-DBR_2022-01-12_DE_TXT_Datensatz.zip"
## [11] "C-DBR_2022-01-12_DE_XML_Anlagen.zip"
## [12] "C-DBR_2022-01-12_DE_XML_Datensatz.zip"
## [13] "C-DBR_2022-01-12_Source_Code.zip"
```

```
file.rename(cleanup,
            cleanup.destination)
```

```
## [1] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
```

## 20 Abschluss

### 20.1 Datumsstempel

```
print(datestamp)
```

```
## [1] "2022-01-12"
```

### 20.2 Datum und Uhrzeit (Anfang)

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

```
## [1] "2022-01-12 13:50:58 CET"
```

### 20.3 Datum und Uhrzeit (Ende)

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

```
## [1] "2022-01-12 14:37:10 CET"
```

### 20.4 Laufzeit des gesamten Skripts

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

```
## Time difference of 46.20468 mins
```

### 20.5 Warnungen

```
warnings()
```

```
## Warning messages:
## 1: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 2: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 3: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 4: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 5: Removed 1 rows containing missing values (geom_text_repel).
## 6: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 7: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 8: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 9: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 10: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 11: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 12: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 13: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 14: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 15: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 16: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 17: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 18: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 19: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 20: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 21: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 22: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 23: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 24: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 25: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 26: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 27: In if (edgelist != 0) { ... :
##   the condition has length > 1 and only the first element will be used
## 28: In if (edgelist != 0) { ... :
```

```

## the condition has length > 1 and only the first element will be used
## 29: In if (edgelist != 0) { ... :
## the condition has length > 1 and only the first element will be used
## 30: In if (edgelist != 0) { ... :
## the condition has length > 1 and only the first element will be used
## 31: In if (edgelist != 0) { ... :
## the condition has length > 1 and only the first element will be used
## 32: In if (edgelist != 0) { ... :
## the condition has length > 1 and only the first element will be used
## 33: In if (edgelist != 0) { ... :
## the condition has length > 1 and only the first element will be used
## 34: In if (edgelist != 0) { ... :
## the condition has length > 1 and only the first element will be used
## 35: In if (edgelist != 0) { ... :
## the condition has length > 1 and only the first element will be used
## 36: In if (edgelist != 0) { ... :
## the condition has length > 1 and only the first element will be used
## 37: In if (edgelist != 0) { ... :
## the condition has length > 1 and only the first element will be used
## 38: In if (edgelist != 0) { ... :
## the condition has length > 1 and only the first element will be used
## 39: In if (edgelist != 0) { ... :
## the condition has length > 1 and only the first element will be used
## 40: In if (edgelist != 0) { ... :
## the condition has length > 1 and only the first element will be used
## 41: In if (edgelist != 0) { ... :
## the condition has length > 1 and only the first element will be used
## 42: In if (edgelist != 0) { ... :
## the condition has length > 1 and only the first element will be used
## 43: In if (edgelist != 0) { ... :
## the condition has length > 1 and only the first element will be used
## 44: In if (edgelist != 0) { ... :
## the condition has length > 1 and only the first element will be used
## 45: In if (edgelist != 0) { ... :
## the condition has length > 1 and only the first element will be used
## 46: In if (edgelist != 0) { ... :
## the condition has length > 1 and only the first element will be used
## 47: In if (edgelist != 0) { ... :
## the condition has length > 1 and only the first element will be used
## 48: Removed 1 rows containing missing values (geom_text_repel).
## 49: In if (edgelist != 0) { ... :
## the condition has length > 1 and only the first element will be used
## 50: In if (edgelist != 0) { ... :
## the condition has length > 1 and only the first element will be used

```

## 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] future.apply_1.8.1 future_1.23.0 ggraph_2.0.5 igraph_1.2.11
##  [5] openssl_1.4.6 scales_1.1.1 quanteda_3.2.0 data.table
##  [9] ggplot2_3.3.5 pdftools_3.0.1 magick_2.7.3 kableExtra_1.3.4
## [13] knitr_1.37 RcppTOML_0.1.7 xml2_1.3.3 rvest_1.0.2
## [17] zip_2.2.0
##
## loaded via a namespace (and not attached):
##  [1] viridis_0.6.2 httr_1.4.2 tidyr_1.1.4 tidygraph_1.2.0
##  [5] viridisLite_0.4.0 RcppParallel_5.1.5 askpass_1.1 highr_0.9
##  [9] selectr_0.4-2 renv_0.15.0 yaml_2.2.1 ggrepel_0.9.1
## [13] globals_0.14.0 qpdf_1.1 pillar_1.6.4 lattice_0.20-45
## [17] glue_1.6.0 digest_0.6.29 polyclip_1.10-0 colorspace_2.0-2
## [21] htmltools_0.5.2 Matrix_1.4-0 pkgconfig_2.0.3 listenv_0.8.0
## [25] purrr_0.3.4 webshot_0.5.2 svglite_2.0.0 tweenr_1.0.2
## [29] ggforce_0.3.3 tibble_3.1.6 generics_0.1.1 farver_2.1.0
## [33] ellipsis_0.3.2 withr_2.4.3 magrittr_2.0.1 crayon_1.4.2
## [37] evaluate_0.14 stopwords_2.3 fansi_1.0.0 parallelly
##  [41] MASS_7.3-54 tools_4.0.5 lifecycle_1.0.1 stringr_1.4.0
## [45] munsell_0.5.0 compiler_4.0.5 systemfonts_1.0.3 rlang_0.4.12
```

```
## [49] grid_4.0.5      rstudioapi_0.13  labeling_0.4.2   rmarkdown_2.11
## [53] gtable_0.3.0    codetools_0.2-18 curl_4.3.2       graphlayouts
## [57] R6_2.5.1        gridExtra_2.3    dplyr_1.0.7      fastmap_1.1.0
## [61] utf8_1.2.2      fastmatch_1.1-3 stringi_1.7.6     Rcpp_1.0.7
## [65] vctrs_0.3.8     tidyselect_1.1.1 xfun_0.29
```

## Literaturverzeichnis

- Bengtsson, Henrik. 2021a. “A Unifying Framework for Parallel and Distributed Processing in R Using Futures.” <https://journal.r-project.org/archive/2021/RJ-2021-048/index.html>.
- . 2021b. “A Unifying Framework for Parallel and Distributed Processing in R Using Futures.” <https://journal.r-project.org/archive/2021/RJ-2021-048/index.html>.
- . 2021c. *Future.apply: Apply Function to Elements in Parallel Using Futures*. <https://CRAN.R-project.org/package=future.apply>.
- . 2021d. *Future: Unified Parallel and Distributed Processing in R for Everyone*. <https://CRAN.R-project.org/package=future>.
- 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>.
- Csardi, Gabor, and Tamas Nepusz. 2006. “The Igraph Software Package for Complex Network Research.” *InterJournal Complex Systems*: 1695. <https://igraph.org>.
- Csárdi, Gábor, Kuba Podgórski, and Rich Geldreich. 2021. *Zip: Cross-Platform Zip Compression*. <https://github.com/r-lib/zip#readme>.
- Dowle, Matt, and Arun Srinivasan. 2021. *Data.table: Extension of ‘Data.frame’*. <https://CRAN.R-project.org/package=data.table>.
- Eddelbuettel, Dirk. 2020. *RcppTOML: Rcpp Bindings to Parser for Tom’s Obvious Markup Language*. <http://dirk.eddelbuettel.com/code/rcpp.toml.html>.
- file., See AUTHORS. 2022. *Igraph: Network Analysis and Visualization*. <https://igraph.org>.
- Ooms, Jeroen. 2021a. *Magick: Advanced Graphics and Image-Processing in R*. <https://CRAN.R-project.org/package=magick>.
- . 2021b. *Openssl: Toolkit for Encryption, Signatures and Certificates Based on Openssl*. <https://github.com/jeroen/openssl>.
- . 2021c. *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/>.
- 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. 2021. *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>.