

**Corpus der
Amtlichen Entscheidungssammlung
des
Bundesverfassungsgerichts
(C-BVerfGE-Source)**

COMPILATION REPORT

Version 2022-06-20

License MIT-0

DOI: 10.5281/zenodo.6569897

Titel	Source Code des »Corpus der amtlichen Entscheidungssammlung des Bundesverfassungsgerichts«
Abkürzung	C-BVerfGE-Source
Autor	Seán Fobbe
Version	2022-06-20
Download	https://doi.org/10.5281/zenodo.6569897
Lizenz	MIT No Attribution (MIT-0)

Zitiervorschlag

Seán Fobbe (2022). Source Code des »Corpus der amtlichen Entscheidungssammlung des Bundesverfassungsgerichts« (C-BVerfGE-Source). Version 2022-06-20. Zenodo. DOI: 10.5281/zenodo.6569897.

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 2022-06-20. 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. 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

1	Corpus der amtlichen Entscheidungssammlung des Bundesverfassungsgerichts (C-BVerfGE)	9
1.1	Überblick	9
1.2	Funktionsweise	9
1.3	Kompilierung	9
1.4	Systemanforderungen	10
1.4.1	Betriebssystem	10
1.4.2	Software	10
1.4.3	Parallelisierung	10
1.4.4	Speicherplatz	10
1.5	Weitere Open Access Veröffentlichungen (Fobbe)	10
1.6	Kontakt	10
2	Vorbereitung	11
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	13
2.6	Weitere Verzeichnisse definieren	13
2.7	Dateien aus vorherigen Runs bereinigen	13
2.8	Verzeichnisse anlegen	13
2.9	Vollzitate statistischer Software schreiben	14
2.10	Allgemeine Konfiguration	14
2.10.1	Konfiguration einlesen	14
2.10.2	Konfiguration anzeigen	14
2.10.3	Knitr Optionen setzen	15
2.10.4	Download Timeout setzen	15
2.10.5	Quellenangabe für Diagramme definieren	15
2.10.6	Präfix für Dateien definieren	15
2.10.7	Präfix für Diagramme definieren	16
2.10.8	Quanteda-Optionen setzen	16
2.11	LaTeX Konfiguration	16
2.11.1	LaTeX Parameter definieren	16
2.11.2	LaTeX Parameter schreiben	17
2.12	Parallelisierung aktivieren	17
2.12.1	Anzahl logischer Kerne festlegen	17
2.12.2	Quanteda	18
2.12.3	Data.table	18
3	Download: Entscheidungen der BVerfGE	19
3.1	Zeitstempel: Linksammlung Beginn	19
3.2	Download vorbereiten	19
3.2.1	Funktion zeigen	19
3.2.2	Link zur Entscheidungsliste der amtlichen Sammlung definieren	19
3.2.3	Links zu HTML-Übersichten extrahieren	19
3.2.4	PDF-Links zu Entscheidungen extrahieren	20

3.3	Einzelkorrektur für AfD-Beitritt zu Mietendeckeln (PDF)	21
3.4	HTML-Links definieren	22
3.5	Einzelkorrektur für AfD-Beitritt zu Mietendeckeln (HTML)	22
3.6	Zeitstempel: Linksammlung Ende	22
3.7	Dauer: Linksammlung	22
3.8	Dateinamen erstellen	22
3.8.1	Extrahieren relevanter Metadaten	22
3.8.2	Formatierung von Registerzeichen anpassen	23
3.8.3	Formatierung von Spruchkoerper-Typ anpassen	23
3.8.4	Erste strenge REGEX-Validierung der Dateinamen	23
3.8.5	Ergebnis der ersten REGEX-Validierung	24
3.8.6	Skript stoppen falls erste REGEX-Validierung gescheitert	24
3.8.7	Zusätzliche Variablen einfügen	24
3.8.8	Nicht benannte Entscheidungen anzeigen	25
3.8.9	NAs einfügen für nicht benannte Entscheidungen	25
3.8.10	Zweite strenge REGEX-Validierung der Dateinamen	26
3.8.11	Ergebnis der zweiten REGEX-Validierung	26
3.8.12	Skript stoppen falls zweite REGEX-Validierung gescheitert	26
4	PDF-Download	28
4.1	Data Table für PDF-Download erstellen	28
4.2	Zeitstempel: PDF-Download Beginn	28
4.3	PDF-Download durchführen	28
4.4	Zeitstempel: PDF-Download Ende	29
4.5	Dauer: PDF-Download	29
4.6	PDF-Download: Ergebnis	29
4.6.1	Anzahl herunterzuladender Dateien	29
4.6.2	Anzahl heruntergeladener Dateien	29
4.6.3	Fehlbetrag	29
4.6.4	Fehlende Dateien	30
4.7	PDF-Wiederholungsversuch	30
4.8	PDF-Download: Gesamtergebnis	31
4.8.1	Anzahl herunterzuladender Dateien	31
4.8.2	Anzahl heruntergeladener Dateien	31
4.8.3	Fehlbetrag	31
4.8.4	Fehlende Dateien	31
4.8.5	Abschließende Hinweise	31
5	HTML-Download	32
5.1	Data Table für HTML-Download erstellen	32
5.2	Zeitstempel: HTML-Download Beginn	32
5.3	HTML-Download durchführen	32
5.4	Zeitstempel: HTML-Download Ende	32
5.5	Dauer: HTML-Download	33
5.6	HTML-Download: Ergebnis	33
5.6.1	Anzahl herunterzuladender Dateien	33
5.6.2	Anzahl heruntergeladener Dateien	33
5.6.3	Fehlbetrag	33
5.6.4	Fehlende Dateien	33
5.7	HTML-Wiederholungsversuch	34

5.8	HTML-Download: Gesamtergebnis	34
5.8.1	Anzahl herunterzuladender Dateien	34
5.8.2	Anzahl heruntergeladener Dateien	34
5.8.3	Fehlbetrag	34
5.8.4	Fehlende Dateien	35
6	HTML verarbeiten	36
6.1	Funktion anzeigen: f.bverfg.extract.meta	36
6.2	Funktion anzeigen: f.bverfg.extract.content	37
6.3	HTML-Dateien definieren	41
6.4	HTML-Dateien einlesen	41
6.5	HTML-Dateien parsen	41
6.6	Data Table mit allen Metadaten (inkl. ECLI)	42
6.7	Data Table mit vollständiger segmentierter Variante	42
6.8	Special Character entfernen	42
6.9	Stichprobe Metadaten	42
6.10	Stichprobe Segmentierte Variante	42
7	Text-Extraktion aus PDF	43
7.1	Vektor der zu extrahierenden Dateien erstellen	43
7.2	Anzahl zu extrahierender Dateien	43
7.3	PDF extrahieren: Funktion anzeigen	43
7.4	Text Extrahieren	44
8	Korpus Erstellen	45
8.1	TXT-Dateien Einlesen	45
8.2	In Data Table umwandeln	45
8.3	Durch Zeilenumbruch getrennte Wörter zusammenfügen	45
8.3.1	Funktion anzeigen	45
8.3.2	Funktion ausführen	46
8.4	Variable "datum" als Datentyp "IDate" kennzeichnen	46
8.5	Variable "entscheidungsjahr" hinzufügen	46
8.6	Variable "eingangsjahr_iso" hinzufügen	46
8.7	Datensatz nach Datum sortieren	46
8.8	Variable "praesi" hinzufügen	47
8.8.1	Lebensdaten einlesen	47
8.8.2	Personaldaten anzeigen	47
8.8.3	Hypothetisches Amtsende für PräsidentIn	48
8.8.4	Schleife vorbereiten	48
8.8.5	Vektor erstellen	48
8.8.6	Vektor einfügen	48
8.9	Variable "v_praesi" hinzufügen	48
8.9.1	Personaldaten einlesen	48
8.9.2	Personaldaten anzeigen	49
8.9.3	Hypothetisches Amtsende für Vize-PräsidentIn	49
8.9.4	Schleife vorbereiten	49
8.9.5	Vektor erstellen	50
8.9.6	Vektor einfügen	50
8.10	Variable "verfahrensart" hinzufügen	50
8.10.1	Datensatz einlesen	50

8.10.2	Datensatz auf relevante Daten reduzieren	50
8.10.3	Indizes bestimmen	50
8.10.4	Vektor der Verfahrensarten erstellen und einfügen	50
8.11	Variable “aktenzeichen” hinzufügen	51
8.12	Variable “doi_concept” hinzufügen	51
8.13	Variable “doi_version” hinzufügen	51
8.14	Variable “version” hinzufügen	51
8.15	Variable “lizenz” hinzufügen	51
8.16	Variable “ecli” hinzufügen	52
8.16.1	Fehlerhafte ECLI korrigieren (HTML Meta)	52
8.16.2	Fehlerhafte ECLI korrigieren (HTML Full)	53
8.16.3	ECLI-Test 1: ECLI die in PDF, aber nicht in HTML vorhanden sind	54
8.16.4	ECLI-Test 2: ECLI die in HTML, aber nicht in PDF vorhanden sind	54
8.16.5	Zum Vergleich: Beim Download fehlende PDF-Dateien	54
8.16.6	ECLI Merge: Metadaten aus Hauptdatensatz in segmentierte Vari- ante mergen	54
8.16.7	ECLI Merge: Metadaten aus HTML-Extraktion in Hauptdatensatz mergen	54
8.17	Variable “entscheidung_typ” hinzufügen	55
8.17.1	Zitiervorschläge parsen	55
8.17.2	Kürzen	55
8.17.3	Vektor in Datensatz einfügen	55
9	Frequenztabellen erstellen	56
9.1	Funktion anzeigen	56
9.2	Ignorierte Variablen	57
9.3	Liste zu prüfender Variablen	57
9.4	Frequenztabellen erstellen	58
10	Frequenztabellen visualisieren	82
10.1	Präfix erstellen	82
10.2	Tabellen einlesen	82
10.3	Diagramm: Typ der Entscheidung	83
10.4	Diagramm: Typ des Spruchkörpers	84
10.5	Diagramm: Spruchkörper nach Aktenzeichen	85
10.6	Diagramm: Registerzeichen	86
10.7	Diagramm: Präsident:in	88
10.8	Diagramm: Vize-Präsident:in	89
10.9	Diagramm: Entscheidungsjahr	90
10.10	Diagramm: Eingangsjahr (ISO)	91
11	Korpus-Analytik	92
11.1	Berechnung linguistischer Kennwerte	92
11.1.1	Funktion anzeigen	92
11.1.2	Berechnung durchführen	93
11.2	Variablen-Namen anpassen	94
11.3	Kennwerte dem Korpus hinzufügen	94
11.4	Variante mit Metadaten erstellen	94
11.5	Linguistische Kennwerte	94
11.5.1	Zusammenfassungen berechnen	94

11.5.2 Zusammenfassungen anzeigen	95
11.5.3 Zusammenfassungen speichern	95
11.6 Quantitative Variablen	97
11.6.1 Entscheidungsdatum	97
11.6.2 Zusammenfassungen berechnen	97
11.6.3 Zusammenfassungen anzeigen	98
11.6.4 Zusammenfassungen speichern	98
11.7 Verteilungen linguistischer Kennwerte	99
11.7.1 Diagramm: Verteilung Zeichen	99
11.7.2 Diagramm: Verteilung Tokens	99
11.7.3 Diagramm: Verteilung Typen	101
11.7.4 Diagramm: Verteilung Sätze	102
12 Linguistische Annotationen berechnen	103
12.1 Funktion anzeigen: f.future_spacyparse	103
12.2 Berechnungen durchführen	104
13 Kontrolle der Variablen	105
13.1 Semantische Sortierung der Variablen	105
13.1.1 Variablen sortieren: Hauptdatensatz	105
13.1.2 Variablen sortieren: Metadaten	106
13.1.3 Variablen sortieren: Segmentiert	107
13.2 Anzahl Variablen der Datensätze	108
13.3 Alle Variablen-Namen der Datensätze	108
14 CSV-Dateien erstellen	110
14.1 CSV mit vollem Datensatz speichern	110
14.2 CSV mit Metadaten speichern	110
14.3 CSV mit Segmenten speichern	110
14.4 CSV mit Annotationen speichern	110
15 Dateigrößen analysieren	111
15.1 Gesamtgröße	111
15.1.1 Korpus-Objekt in RAM (MB)	111
15.1.2 CSV Korpus (MB)	111
15.1.3 CSV Metadaten (MB)	111
15.1.4 CSV Annotiert (MB)	111
15.1.5 CSV Segmentiert (MB)	111
15.1.6 PDF-Dateien (MB)	112
15.1.7 TXT-Dateien (MB)	112
15.2 Diagramm: Verteilung der Dateigrößen (PDF)	113
15.3 Diagramm: Verteilung der Dateigrößen (TXT)	114
16 Erstellen der ZIP-Archive	115
16.1 Verpacken der CSV-Dateien	115
16.1.1 Vollständiger Datensatz	115
16.1.2 Metadaten	115
16.1.3 Segmentiert	115
16.1.4 Annotiert	115
16.2 Verpacken der PDF-Dateien	116

16.3 Verpacken der HTML-Dateien	116
16.4 Verpacken der TXT-Dateien	116
16.5 Verpacken der Analyse-Dateien	117
16.6 Verpacken der Source-Dateien	117
17 Kryptographische Hashes	118
17.1 Liste der ZIP-Archive erstellen	118
17.2 Funktion anzeigen: future_multihashes	118
17.3 Hashes berechnen	119
17.4 In Data Table umwandeln	119
17.5 Index hinzufügen	119
17.6 In Datei schreiben	120
17.7 Leerzeichen hinzufügen um Zeilenumbruch zu ermöglichen	120
17.8 In Bericht anzeigen	120
18 Aufräumen	123
19 Abschluss	124
19.1 Datumsstempel	124
19.2 Datum und Uhrzeit (Anfang)	124
19.3 Datum und Uhrzeit (Ende)	124
19.4 Laufzeit des gesamten Skriptes	124
19.5 Warnungen	124
20 Parameter für strenge Replikationen	125
Literaturverzeichnis	126

1 Corpus der amtlichen Entscheidungssammlung des Bundesverfassungsgerichts (C-BVerfGE)

1.1 Überblick

Dieser Code lädt alle auf www.bundesverfassungsgericht.de veröffentlichten Entscheidungen der amtlichen Entscheidungssammlung des Bundesverfassungsgerichts (BVerfG) herunter und kompiliert sie in einen reichhaltigen menschen- und maschinenlesbaren Korpus. Es ist die Grundlage für den **Corpus der amtlichen Entscheidungssammlung des Bundesverfassungsgerichts (C-BVerfGE)**.

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.3831111>

1.2 Funktionsweise

Primäre Endprodukte des Skripts sind folgende ZIP-Archive (im Ordner ‘output’):

- Der volle Datensatz im CSV-Format
- Die reinen Metadaten im CSV-Format (wie unter 1, nur ohne Entscheidungstexte)
- (Optional) Tokenisierte Form aller Texte mit linguistischen Annotationen im CSV-Format
- Der volle Datensatz im TXT-Format (reduzierter Umfang an Metadaten)
- Der volle Datensatz im PDF-Format (reduzierter Umfang an Metadaten)
- Alle Analyse-Ergebnisse (Tabellen als CSV, Grafiken als PDF und PNG)
- Der Source Code und alle weiteren Quelldaten

Zusätzlich werden für alle ZIP-Archive kryptographische Signaturen (SHA2-256 und SHA3-512) berechnet und in einer CSV-Datei hinterlegt. Weiterhin kann optional ein PDF-Bericht erstellt werden (siehe unter “Kompilierung”).

1.3 Kompilierung

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-BVerfGE_FullCompile.R")
```

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

1.4 Systemanforderungen

1.4.1 Betriebssystem

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 des jeweiligen Compilation Reports.

1.4.2 Software

Sie müssen die Programmiersprache R installiert haben. Starten Sie danach eine Session im Ordner des Projekts, Sie sollten automatisch zur Installation aller packages in der empfohlenen Version aufgefordert werden. Andernfalls führen Sie bitte folgenden Befehl aus:

```
renv::restore()
```

Um die PDF Reports zu kompilieren benötigen Sie eine LaTeX-Installation. Sie können diese auf Fedora wie folgt installieren:

```
sudo dnf install texlive-scheme-full
```

Alternativ können sie das R package **tinytex** installieren.

1.4.3 Parallelisierung

In der Standard-Einstellung wird das Skript vollautomatisch die maximale Anzahl an Rechenkernen/Threads auf dem System zu nutzen. Die Anzahl der verwendeten Kerne kann in der Konfigurationsdatei angepasst werden. Wenn die Anzahl Threads auf 1 gesetzt wird, ist die Parallelisierung deaktiviert.

1.4.4 Speicherplatz

Auf der Festplatte sollten 8 GB Speicherplatz vorhanden sein.

1.5 Weitere Open Access Veröffentlichungen (Fobbe)

Website — <https://www.seanfobbe.de>

Open Data — <https://zenodo.org/communities/sean-fobbe-data/>

Source Code — <https://zenodo.org/communities/sean-fobbe-code/>

Volltexte regulärer Publikationen — <https://zenodo.org/communities/sean-fobbe-publications/>

1.6 Kontakt

Fehler gefunden? Anregungen? Kommentieren Sie gerne im Issue Tracker auf GitHub oder schreiben Sie mir eine E-Mail an fobbe-data@posteo.de

2 Vorbereitung

2.1 Datumsstempel

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

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

```
## [1] "2022-06-20"
```

2.2 Datum und Uhrzeit (Beginn)

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

```
## [1] "2022-06-20 22:53:32 CEST"
```

2.3 Packages Laden

```
library(magick)      # Cropping von PNG-Dateien
```

```
## Linking to ImageMagick 6.9.12.32
## Enabled features: cairo, fontconfig, freetype, ghostscript, lcms, pango, raw,
    rsvg, webp, x11
## Disabled features: fftw, heic
```

```
## Using 16 threads
```

```
library(RcppTOML)    # Verarbeitung von TOML-Format
library(mgsub)       # Mehrfache simultane String-Substitutions
library(httr)        # HTTP-Werkzeuge
library(rvest)       # HTML/XML-Extraktion
library(knitr)       # Professionelles Reporting
library(kableExtra)  # Verbesserte Kable Tabellen
library(pdftools)    # Verarbeitung von PDF-Dateien
```

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

```
library(ggplot2)      # Fortgeschrittene Datenvisualisierung  
library(scales)       # Skalierung von Diagrammen  
library(data.table)   # Fortgeschrittene Datenverarbeitung
```

```
## data.table 1.14.2 using 8 threads (see ?getDTthreads). Latest news: r-  
datatable.com
```

```
library(readtext)     # TXT-Dateien einlesen  
library(quanteda)     # Fortgeschrittene Computerlinguistik
```

```
## Package version: 3.2.0  
## Unicode version: 13.0  
## ICU version: 67.1
```

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

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

```
library(spacyr)        # Linguistische Annotationen  
library(future)        # Parallelisierung mit Futures  
library(future.apply)  # Apply-Funktionen für Futures
```

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("functions/f.bverfg.extract.content.R")  
source("functions/f.bverfg.extract.meta.R")  
  
source("R-fobbe-proto-package/f.remove.specialunderline.R")  
source("R-fobbe-proto-package/f.linkextract.R")  
  
source("R-fobbe-proto-package/f.hyphen.remove.R")  
source("R-fobbe-proto-package/f.year.iso.R")  
source("R-fobbe-proto-package/f.fast.freqtable.R")
```

```
source("R-fobbe-proto-package/f.future_lingsummarize.R")
source("R-fobbe-proto-package/f.future_multihashes.R")
source("R-fobbe-proto-package/f.future_pdf_to_txt.R")
source("R-fobbe-proto-package/f.future_spacyparse.R")
```

2.5 Verzeichnis für Analyse-Ergebnisse und Diagramme definieren

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

2.6 Weitere Verzeichnisse definieren

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

2.7 Dateien aus vorherigen Runs bereinigen

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

unlink(dirs,
       recursive = TRUE)

files.delete <- list.files(pattern = "\\\\.zip|\\.pdf|\\.txt|\\.html",
                           ignore.case = TRUE)

unlink(files.delete)
```

2.8 Verzeichnisse anlegen

```
dir.create(dir.analysis)

lapply(dirs, dir.create)
```

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

2.9 Vollzitate statistischer Software schreiben

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

```
## Warning in as.vector(x): input string ' author = {Stéphane Dray and Anne-
## Béatrice Dufour and Jean Thioulouse and Thibaut Jombart and Sandrine Pavoine
## and
## Jean R. Lobry and Sébastien Ollier and Aurélie Siberchicot and Daniel Chessel
## },'
## cannot be translated to UTF-8, is it valid in 'ANSI_X3.4-1968'?
```

2.10 Allgemeine Konfiguration

2.10.1 Konfiguration einlesen

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

2.10.2 Konfiguration anzeigen

```
print(config)
```

```
## List of 10
## $ annotate :List of 1
##   ..$ toggle: logi TRUE
## $ cores    :List of 2
##   ..$ max   : logi TRUE
##   ..$ number: int 8
## $ doi      :List of 4
##   ..$ aktenzeichen : chr "10.5281/zenodo.4569564"
##   ..$ data          :List of 2
##     .. ..$ concept: chr "10.5281/zenodo.3831111"
##     .. ..$ version: chr "10.5281/zenodo.6569896"
##   ..$ personendaten: chr "10.5281/zenodo.4568682"
##   ..$ software      :List of 2
##     .. ..$ concept: chr "10.5281/zenodo.4265933"
##     .. ..$ version: chr "10.5281/zenodo.6569897"
## $ 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 [1:11] "text" "eingangsnummer" "datum" "doc_id" ...
## $ license   :List of 2
```

```
## ..$ code: chr "MIT-0"
## ..$ data: chr "Creative Commons Zero 1.0 Universal"
## $ parallel :List of 4
## ..$ extractPDF : logi TRUE
## ..$ lingsummarize: logi TRUE
## ..$ multihashes : logi TRUE
## ..$ spacyparse : logi FALSE
## $ project :List of 3
## ..$ author : chr "Seán Fobbe"
## ..$ fullname : chr "Corpus der amtlichen Entscheidungssammlung des
Bundesverfassungsgerichts"
## ..$ shortname: chr "C-BVerfGE"
## $ quanteda :List of 1
## ..$ tokens_locale: chr "de_DE"
```

2.10.3 Knitr Optionen setzen

```
knitr::opts_chunk$set(fig.path = paste0(dir.analysis, "/"),
  dev = config$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.6569896"
```

2.10.6 Präfix für Dateien definieren

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

```
## [1] "C-BVerfGE_2022-06-20"
```

2.10.7 Präfix für Diagramme definieren

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

```
## [1] "C-BVerfGE | Version 2022-06-20"
```

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",  
              "\\n% NOTE: Diese Datei wurde während des Kompilierungs-Prozesses  
automatisch erstellt.\\n",  
              "\\n%-----Autor-----",  
              paste0("\\\\newcommand{\\projectauthor}{",  
                     config$project$author,  
                     "}") ,  
              "\\n%-----Version-----",  
              paste0("\\\\newcommand{\\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$version,
        "}"),
    "\\n%-----Additional DOIs-----",
    paste0("\\newcommand{\\aktenzeichenurldoi}{https://doi.org/",
        config$doi$aktenzeichen,
        "}"),
    paste0("\\newcommand{\\personendatenurldoi}{https://doi.org/",
        config$doi$personendaten,
        "}"))

```

2.11.2 LaTeX Parameter schreiben

```

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

```

2.12 Parallelisierung aktivieren

Parallelisierung wird zur Beschleunigung 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.

2.12.1 Anzahl logischer Kerne festlegen

```

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

```

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

```
## system  
## 16
```

2.12.2 Quanteda

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

2.12.3 Data.table

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

3 Download: Entscheidungen der BVerfGE

3.1 Zeitstempel: Linksammlung Beginn

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

```
## [1] "2022-06-20 22:53:33 CEST"
```

3.2 Download vorbereiten

3.2.1 Funktion zeigen

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

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

3.2.2 Link zur Entscheidungsliste der amtlichen Sammlung definieren

```
URL <- "https://www.bundesverfassungsgericht.de/DE/Entscheidungen/Entscheidungen/
Amtliche%20Sammlung%20BVerfGE.html"
```

3.2.3 Links zu HTML-Übersichten extrahieren

```
links1 <- f.linkextract(URL)
links2 <- grep ("Entscheidungen/Liste",
               links1,
               ignore.case = TRUE,
               value = TRUE)

links2 <- paste0("https://www.bundesverfassungsgericht.de/",
               links2)
```

3.2.4 PDF-Links zu Entscheidungen extrahieren

Es gibt zwei verschiedene URL-Varianten mit denen Entscheidungen verlinkt sind. Diese werden als Variante A und B separat ausgewertet und danach zusammengefügt.

```
links3 <- lapply(links2,
                  f.linkextract)

links4 <- unlist(links3)
```

Variante A

```
links5a <- grep ("SharedDocs/Entscheidungen",
                 links4,
                 ignore.case = TRUE,
                 value = TRUE)

links5a <- paste0("https://www.bundesverfassungsgericht.de/",
                  links5a)

links6a <- gsub("Entscheidungen",
               "Downloads",
               links5a)

links.pdf.a <- gsub("\\\\.html.*",
                  "\\..pdf\\?__blob=publicationFile\\&v\\=1",
                  links6a)
```

Variante B

```
links5b <- grep ("https://www.bverfg.de/e/",
                 links4,
                 ignore.case = TRUE,
                 value = TRUE)

links6b <- gsub("https://www.bverfg.de/e",
               "",
               links5b)

links6b <- gsub("(/[a-z]{2})([0-9]{4})([0-9]{2})(.*)",
               "\\2/\\3\\1\\2\\3\\4",
               links6b)

links.pdf.b <- paste0("https://www.bundesverfassungsgericht.de/SharedDocs/
                      Downloads/DE/",
                      links6b,
                      ".pdf?__blob=publicationFile&v=1")
```

Links manuell hinzufügen

Diese Entscheidungen sind in der offiziellen Liste nicht verlinkt und müssen daher manuell der Liste hinzugefügt werden.

```
links.add <- c("https://www.bundesverfassungsgericht.de/SharedDocs/Downloads/DE/
2003/04/up20030430_1pbvu000102.pdf?__blob=publicationFile&v=1",
              "https://www.bundesverfassungsgericht.de/SharedDocs/Downloads/DE/
2004/03/ks20040330_2bvk000101.pdf?__blob=publicationFile&v=1",
              "https://www.bundesverfassungsgericht.de/SharedDocs/Downloads/DE/
2007/03/es20070329_2bve000207.pdf?__blob=publicationFile&v=1",
              "https://www.bundesverfassungsgericht.de/SharedDocs/Downloads/DE/
2015/07/qk20150720_1bvq002515.pdf?__blob=publicationFile&v=2",
              "https://www.bundesverfassungsgericht.de/SharedDocs/Downloads/DE/
2015/12/rs20151216_2bvr195813.pdf?__blob=publicationFile&v=5")
```

Links manuell entfernen

Diese Entscheidungen sind in der offiziellen Liste irrtümlicherweise verlinkt obwohl nicht in der amtlichen Sammlung enthalten und müssen entfernt werden.

```
links.remove <- c("https://www.bundesverfassungsgericht.de/SharedDocs/Downloads/
DE/2004/03/ks20040323_2bvk000101.pdf?__blob=publicationFile&v=1",
                  "https://www.bundesverfassungsgericht.de/SharedDocs/Downloads/
DE/2015/07/qk20150718_1bvq002515.pdf?__blob=publicationFile&v=1",
                  "https://www.bundesverfassungsgericht.de/SharedDocs/Downloads/
DE/2013/12/rk20131216_2bvr195813.pdf?__blob=publicationFile&v=1")
```

Varianten Zusammenfügen

Hinweis: In der Auflistung der Entscheidungen der amtlichen Sammlung sind abweichende Meinungen separat aufgeführt. Diese sind aber zusammen mit dem ursprünglichen Urteil in derselben PDF-Datei dokumentiert. Daher führen für manche Urteile mehrere Links zur selben PDF-Datei. Durch `setdiff()` werden nicht nur die oben definierten Urteile entfernt, sondern auch alle Duplikate.

```
links.pdf <- c(links.pdf.a,
               links.pdf.b,
               links.add)

links.pdf <- setdiff(links.pdf,
                    links.remove)
```

3.3 Einzelkorrektur für AfD-Beitritt zu Mietendeckeln (PDF)

Hier weicht die Struktur des PDF-Links von der des HTML-Links ab.

```
links.pdf <- gsub("fs20210325_2bvf000120_1",
                 "fs20210325_2bvf000120",
                 links.pdf)
```

3.4 HTML-Links definieren

```
links.html <- gsub("Downloads",  
                  "Entscheidungen",  
                  links.pdf)  
  
links.html <- gsub("pdf\\(?:_|\\.)*",  
                  "html",  
                  links.html)
```

3.5 Einzelkorrektur für AfD-Beitritt zu Mietendeckeln (HTML)

Hier weicht die Struktur des PDF-Links von der des HTML-Links ab.

```
links.html <- gsub("fs20210325_2bvf000120",  
                  "fs20210325_2bvf000120_1",  
                  links.html)
```

3.6 Zeitstempel: Linksammlung Ende

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

```
## [1] "2022-06-20 22:53:41 CEST"
```

3.7 Dauer: Linksammlung

```
end.links-begin.links
```

```
## Time difference of 7.439526 secs
```

3.8 Dateinamen erstellen

3.8.1 Extrahieren relevanter Metadaten

Die Links zu jeder Entscheidung enthalten das Ordinalzahl-Element ihres jeweiligen ECLI-Codes. Struktur und Inhalt der ECLI für deutsche Gerichte sind auf dem Europäischen Justizportal näher erläutert.¹

¹ https://e-justice.europa.eu/content_european_case_law_identifier_ecli-175-de-de.do?member=1

```
filenames <- basename(links.pdf)

filenames <- gsub("[?].*",
  "",
  filenames)
```

Normale Struktur

```
filenames1 <- gsub("[a-z]([a-z])([0-9]{4})([0-9]{2})([0-9]{2})_([0-9])([a-z]*)
  ([0-9]{4})([0-9]{2}).*",
  "BVerfG_\\2-\\3-\\4_\\1_\\5_\\6_\\7_\\8_NA",
  filenames)
```

Struktur von Entscheidungen mit Kollisions-Variable

```
filenames1 <- gsub("[a-z]([a-z])([0-9]{4})([0-9]{2})([0-9]{2})([a-z])_([0-9])([a-
  z]*) ([0-9]{4})([0-9]{2}).*",
  "BVerfG_\\2-\\3-\\4_\\1_\\6_\\7_\\8_\\9_\\5",
  filenames1)
```

3.8.2 Formatierung von Registerzeichen anpassen

```
filenames1 <- gsub("_bv([a-z])_",
  "_Bv\\U\\1_",
  perl = TRUE,
  filenames1)

filenames1 <- gsub("pbvu",
  "PBvU",
  filenames1)
```

3.8.3 Formatierung von Spruchkoerper-Typ anpassen

```
filenames1 <- gsub("_[kps])_",
  "_\\U\\1_",
  perl = TRUE,
  filenames1)
```

3.8.4 Erste strenge REGEX-Validierung der Dateinamen

```
#regex.test1 <- grep("BVerfG_[0-9]{4}-[0-9]{2}-[0-9]{2}_[A-Z]_[0-9NA]+_[A-Za-z]+_
  [0-9]{4}_[0-9]{2}_[0-9a-zA-ZNA]+$",
#
  filenames1,
```

```
#             invert = TRUE,
#             value = TRUE)

regex.test1 <- grep(paste0("BVerfG", # gericht
                           "_",
                           "[0-9]{4}-[0-9]{2}-[0-9]{2}", # datum
                           "_",
                           "[A-Z]", # spruchkoerper_typ
                           "_",
                           "[0-9NA]+", # spruchkoerper_az
                           "_",
                           "[A-Za-z]+", # registerzeichen
                           "_",
                           "[0-9]{4}", # eingangsnummer
                           "_",
                           "[0-9]{2}", # eingangsjahr
                           "_",
                           "[0-9a-zA-Z]+$"), # kollision
                    filenames1,
                    invert = TRUE,
                    value = TRUE)
```

3.8.5 Ergebnis der ersten REGEX-Validierung

Das Ergebnis sollte ein leerer Vektor sein!

```
print(regex.test1)
```

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

3.8.6 Skript stoppen falls erste REGEX-Validierung gescheitert

```
if (length(regex.test1) != 0){
  stop("REGEX VALIDIERUNG 1 GESCHEITERT: AKTENZEICHEN ENTSPRECHEN NICHT DEM
CODEBOOK-SCHEMA!")
}
```

3.8.7 Zusätzliche Variablen einfügen

```
extravariablen <- fread("data/BVerfGE_Variablen_NameBandSeite.csv")

extravariablen$newname <- paste(extravariablen$oldname,
                                extravariablen$name,
                                extravariablen$band,
                                extravariablen$seite,
                                sep = "_")
```



```

extravariablen$newname <- paste0(extravariablen$newname,
                                ".pdf")

filenames2 <- filenames1

targetindices <- match(extravariablen$oldname,
                      filenames2)

newname <- extravariablen$newname

dt <- data.table(targetindices, newname)[complete.cases(targetindices)]

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

filenames2 <- replace(filenames2,
                      dt$targetindices,
                      dt$newname)

}

```

3.8.8 Nicht benannte Entscheidungen anzeigen

Für alle Entscheidungen im C-BVerGE sollten per Hand ein Name vergeben werden sein. Ist dies nicht der Fall, werden noch zu benennende Entscheidungen hier angezeigt.

```

values <- grep(".pdf",
               filenames2,
               invert = TRUE,
               value = TRUE)

indices <- grep(".pdf",
                filenames2,
                invert = TRUE)

print(values)

```

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

3.8.9 NAs einfügen für nicht benannte Entscheidungen

```

filenames2[indices] <- paste0(values,
                              "_NA_NA_NA.pdf")

```

3.8.10 Zweite strenge REGEX-Validierung der Dateinamen

```
#regex.test2 <- grep("^BVerfG_[0-9]{4}-[0-9]{2}-[0-9]{2}_[SPKB]_[0-9NA]+_[A-Za-z]
  ]+_[0-9]{4}_[0-9]{2}_[0-9a-zA]+_[0-9ÄÜÖäüöA-Za-z\\-]+_[NAO-9]+_[NAO-9]+\\.
  pdf$",
#
#                               filenames2,
#                               value = TRUE,
#                               invert = TRUE)

regex.test2 <- grep(paste0("^BVerfG", # gericht
  "_",
  "[0-9]{4}-[0-9]{2}-[0-9]{2}", # datum
  "_",
  "[SPKB]", # spruchkoerper_typ
  "_",
  "[0-9NA]+", # spruchkoerper_az
  "_",
  "[A-Za-z]+", # registerzeichen
  "_",
  "[0-9]{4}", # eingangsnummer
  "_",
  "[0-9]{2}", # eingangsjahr_az
  "_",
  "[0-9a-zA]+", # kollision
  "_",
  "[0-9ÄÜÖäüöA-Za-z\\-]+", # name
  "_",
  "[NAO-9]+", # band
  "_",
  "[NAO-9]+", # seite
  "\\..pdf$"), # Dateiendung
  filenames2,
  value = TRUE,
  invert = TRUE)
```

3.8.11 Ergebnis der zweiten REGEX-Validierung

Das Ergebnis sollte ein leerer Vektor sein!

```
print(regex.test2)
```

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

3.8.12 Skript stoppen falls zweite REGEX-Validierung gescheitert

```
if (length(regex.test2) != 0){
  stop("REGEX VALIDIERUNG 2 GESCHEITERT: DATEINAMEN ENTSPRECHEN NICHT DEM
  CODEBOOK-SCHEMA!")
}
```

}

4 PDF-Download

4.1 Data Table für PDF-Download erstellen

```
dt <- data.table(links.pdf,  
                 filenames2)
```

4.2 Zeitstempel: PDF-Download Beginn

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

```
## [1] "2022-06-20 22:53:41 CEST"
```

4.3 PDF-Download durchführen

Hinweis: Es ist nötig jeden Link auf das Vorhandensein einer PDF-Datei zu prüfen, weil für manche Entscheidungen zwar HTML-Seiten vorhanden sind, aber keine korrespondierende PDF-Datei.

```
for (i in seq_len(dt[,.N])){  
  
  response <- GET(dt$links.pdf[i])  
  
  Sys.sleep(runif(1, 0.25, 0.75))  
  
  if (response$headers$"content-type" == "application/pdf;charset=UTF-8" &  
      response$status_code == 200){  
    tryCatch({download.file(url = dt$links.pdf[i],  
                           destfile = dt$filenames2[i])  
    },  
    error=function(cond) {  
      return(NA)}  
    )  
  }else{  
    print(paste0(dt$filenames2[i],  
                 " : kein PDF vorhanden"))  
  }  
  Sys.sleep(runif(1, 0.5, 1.5))  
}
```

```
## [1] "BVerfG_2001-10-01_S_2_BvB_0001_01_NA_NPD-Verbot-EA-RückgabeEDV-2_104_61.  
pdf : kein PDF vorhanden"
```

4.4 Zeitstempel: PDF-Download Ende

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

```
## [1] "2022-06-20 23:18:30 CEST"
```

4.5 Dauer: PDF-Download

```
end.download - begin.download
```

```
## Time difference of 24.81089 mins
```

4.6 PDF-Download: Ergebnis

4.6.1 Anzahl herunterzuladender Dateien

```
dt[, .N]
```

```
## [1] 764
```

4.6.2 Anzahl heruntergeladener Dateien

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

```
## [1] 763
```

4.6.3 Fehlbetrag

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

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

4.6.4 Fehlende Dateien

```
missing.pdf <- setdiff(dt$filenames2,  
                      files.pdf)  
  
print(missing.pdf)
```

```
## [1] "BVerfG_2001-10-01_S_2_BvB_0001_01_NA_NPD-Verbot-EA-RückgabeEDV-2_104_61.  
pdf"
```

4.7 PDF-Wiederholungsversuch

Download für fehlende Dokumente wiederholen.

```
if(N.missing.pdf > 0){  
  
  dt.retry <- dt[filenames2 %in% missing.pdf]  
  
  for (i in seq_len(dt.retry[,.N])){  
  
    response <- GET(dt.retry$links.pdf[i])  
  
    Sys.sleep(runif(1, 0.25, 0.75))  
  
    if (response$headers$"content-type" == "application/pdf;charset=UTF-8" &  
        response$status_code == 200){  
  
      tryCatch({download.file(url = dt.retry$links.pdf[i],  
                             destfile = dt.retry$filenames2[i])  
  
      },  
      error = function(cond) {  
        return(NA)}  
      )  
    }else{  
      print(paste0(dt.retry$filenames2[i],  
                   " : kein PDF vorhanden"))  
    }  
    Sys.sleep(runif(1, 0.5, 1.5))  
  }  
}
```

```
## [1] "BVerfG_2001-10-01_S_2_BvB_0001_01_NA_NPD-Verbot-EA-RückgabeEDV-2_104_61.  
pdf : kein PDF vorhanden"
```

4.8 PDF-Download: Gesamtergebnis

4.8.1 Anzahl herunterzuladender Dateien

```
dt[, .N]
```

```
## [1] 764
```

4.8.2 Anzahl heruntergeladener Dateien

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

```
## [1] 763
```

4.8.3 Fehlbetrag

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

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

4.8.4 Fehlende Dateien

```
missing.pdf <- setdiff(dt$filenames2, files.pdf)  
print(missing.pdf)
```

```
## [1] "BVerfG_2001-10-01_S_2_BvB_0001_01_NA_NPD-Verbot-EA-RückgabeEDV-2_104_61.  
pdf"
```

4.8.5 Abschließende Hinweise

Hinweis: Für die Entscheidung vom 1.10.2001 zur Rückgabe von EDV-Anlagen im Rahmen des NPD-Verfahrens war auch nach manueller Suche keine PDF-Datei auffindbar.

5 HTML-Download

5.1 Data Table für HTML-Download erstellen

```
names.html <- basename(links.html)

dt.download.html <- data.table(links.html,
                                names.html)
```

5.2 Zeitstempel: HTML-Download Beginn

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

```
## [1] "2022-06-20 23:18:31 CEST"
```

5.3 HTML-Download durchführen

```
for (i in sample(dt.download.html[, .N])){

  tryCatch({download.file(dt.download.html$links.html[i],
                          dt.download.html$names.html[i])
  },
  error = function(cond) {
    return(NA)}
  )

  Sys.sleep(runif(1, 0.3, 1))

}
```

5.4 Zeitstempel: HTML-Download Ende

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

```
## [1] "2022-06-20 23:32:28 CEST"
```


5.5 Dauer: HTML-Download

```
end.download - begin.download
```

```
## Time difference of 13.9493 mins
```

5.6 HTML-Download: Ergebnis

5.6.1 Anzahl herunterzuladender Dateien

```
dt.download.html[,.N]
```

```
## [1] 764
```

5.6.2 Anzahl heruntergeladener Dateien

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

```
## [1] 764
```

5.6.3 Fehlbetrag

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

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

5.6.4 Fehlende Dateien

```
missing <- setdiff(dt.download.html$names.html,  
                  files.html)  
  
print(missing)
```

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

5.7 HTML-Wiederholungsversuch

Download für fehlende Dokumente wiederholen.

```
if(N.missing > 0){  
  dt.retry <- dt.download.html[names.html %in% missing]  
  for (i in seq_len(dt.retry[,.N])){  
    tryCatch({download.file(dt.retry$links.html[i],  
                           dt.retry$names.html[i])  
    },  
    error = function(cond) {  
      return(NA)}  
    )  
    Sys.sleep(runif(1, 0.5, 1.5))  
  }  
}
```

5.8 HTML-Download: Gesamtergebnis

5.8.1 Anzahl herunterzuladender Dateien

```
dt.download.html[,.N]
```

```
## [1] 764
```

5.8.2 Anzahl heruntergeladener Dateien

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

```
## [1] 764
```

5.8.3 Fehlbetrag

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

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

5.8.4 Fehlende Dateien

```
missing <- setdiff(dt.download.html$names.html,  
                  files.html)  
  
print(missing)
```

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

6 HTML verarbeiten

6.1 Funktion anzeigen: f.bverfg.extract.meta

```
print(f.bverfg.extract.meta)
```

```
## function(HTML){
##
##   ## ECLI
##   eclli <- html_elements(HTML, "[class='ecli']") %>% html_text(trim = TRUE)
##
##   ## Alle Aktenzeichen
##   aktenzeichen_alle <- html_elements(HTML, "[class='az2']") %>% html_text(
trim = TRUE)
##   aktenzeichen_alle <- gsub("-", "|",
##                               "",
##                               aktenzeichen_alle)
##   aktenzeichen_alle <- gsub(" +",
##                               " ",
##                               aktenzeichen_alle)
##   aktenzeichen_alle <- tstrsplit(aktenzeichen_alle,
##                                   split = "\n")
##   aktenzeichen_alle <- unlist(aktenzeichen_alle)
##   aktenzeichen_alle <- na.omit(aktenzeichen_alle)
##   aktenzeichen_alle <- trimws(aktenzeichen_alle)
##   aktenzeichen_alle <- unique(aktenzeichen_alle)
##   aktenzeichen_alle <- paste(aktenzeichen_alle, collapse = "|")
##
##   ## Pressemitteilungen
##   pressemitteilung <- html_elements(HTML, "[class='pm']") %>% html_text(trim
= TRUE)
##
##   if(length(pressemitteilung) == 0){
##     pressemitteilung <- "NA"
##   }else{
##     pressemitteilung <- pressemitteilung[1]
##   }
##
##   ## Kurzbeschreibung
##   kurzbeschreibung <- HTML %>% html_elements("title") %>% html_text(trim =
TRUE)
##   kurzbeschreibung <- gsub(".*Bundesverfassungsgericht.*.*Entscheidungen.*
-(.*)",
##                               "\\1",
##                               kurzbeschreibung)
##   kurzbeschreibung <- trimws(kurzbeschreibung)
##
##
##   ## Verkündung ## hier treten noch Probleme auf
##   ##verkuendung <- html_elements(HTML, "[class='vvm2']") %>% html_text(trim
= TRUE)
##}
```



```

##      ## Leitsätze
##      leitsaetze <- unique(html_elements(HTML, ".ls") %>% html_text(trim = TRUE)
##    )
##
##      if (length(leitsaetze) != 0){
##        segment <- paste0("leitsatz-",
##                          1:length(leitsaetze))
##
##        dt.leitsaetze <- data.table(leitsaetze,
##                                   segment)
##
##      }else{
##        dt.leitsaetze <- data.table("NA",
##                                   "NA")
##
##      }
##
##      ## Gegenstand
##      gegenstand <- html_elements(HTML, ".vgt3") %>% html_text(trim = TRUE)
##
##      if (length(gegenstand) == 0){
##        gegenstand <- html_elements(HTML, ".vg1") %>% html_text(trim = TRUE)
##      }
##
##      if (length(gegenstand) != 0){
##
##        segment <- paste0("gegenstand-",
##                          1:length(gegenstand))
##
##        dt.gegenstand <- data.table(gegenstand,
##                                   segment)
##
##      }else{
##        dt.gegenstand <- data.table("NA",
##                                   "NA")
##
##      }
##
##      ## Formel
##      formel <- paste(html_elements(HTML, "[class='bf']") %>% html_text(trim =
##    TRUE),
##                      collapse = " ")
##
##      if (length(formel) != 0){
##
##        segment <- "formel"
##
##        dt.formel <- data.table(formel,
##                                segment)
##
##      }else{
##        dt.formel <- data.table("NA",
##                                "NA")
##

```

```

##
##   }
##
##   ## Tenor
##   tenor <- html_elements(HTML, "ol .bs, .hr") %>% html_text(trim = TRUE) #
##   Mehrere Tenorpunkte
##
##   if (length(tenor) == 0){
##
##       tenor <- html_elements(HTML, ".bs, .hr") %>% html_text(trim = TRUE) #
##       Ein Tenorpunkt
##
##   }
##
##
##   if (length(tenor) != 0){
##       segment <- paste0("tenor-",
##                          1:length(tenor))
##
##       dt.tenor <- data.table(tenor,
##                              segment)
##
##   }else{
##       dt.tenor <- data.table("NA",
##                              "NA")
##
##   }
##
##
##
##   ## Gründe der Entscheidung
##   gruende.nodes <- html_elements(HTML, ".std, .lszb")
##
##   if (length(gruende.nodes) != 0){
##
##       if(html_attr(gruende.nodes, "class")[1] == "lszb"){
##
##           ## Entferne Leitsatzüberschriften
##           gruende.classes <- html_attr(gruende.nodes, "class")
##           ueberschrift.indizes <- grep("lszb", gruende.classes[1:5])
##
##           gruende.nodes <- gruende.nodes[-ueberschrift.indizes]
##       }
##
##       inhaltsverzeichnis.indizes <- grep("inhvz", html_attr(gruende.nodes, "
class"))
##
##       if (length(inhaltsverzeichnis.indizes) > 0){
##           gruende.nodes <- gruende.nodes[-inhaltsverzeichnis.indizes] #
##           Entferne Inhaltsverzeichnis
##       }
##
##
##       etstd.indizes <- grep("et std", html_attr(gruende.nodes, "class"))
##
##       if (length(etstd.indizes) > 0){

```

```

##
##           gruende.nodes <- gruende.nodes[-etzstd.indizes] # Entferne
Langzitate
##
##           }
##
##           etzstd.indizes <- grep("(etz std)|(std etz)", html_attr(gruende.nodes,
"class"))
##
##           if (length(etzstd.indizes) > 0){
##
##           gruende.nodes <- gruende.nodes[-etzstd.indizes] # Entferne
Langzitatüberschriften
##
##           }
##
##
##
##           abw.vorhanden <- "lszb" %in% html_attr(gruende.nodes, "class")
##
##           if (abw.vorhanden == TRUE){
##
##           gruende.classes <- html_attr(gruende.nodes, "class")
##
##           abw.logical <- gruende.classes %in% "lszb"
##
##           abw.pos.all <- which(abw.logical)
##           abw.pos.first <- abw.pos.all[1]
##
##           gruende.nodes <- gruende.nodes[-abw.pos.all] # Entferne Abw-
Meinung-Überschriften
##
##           }
##
##           gruende <- html_text(gruende.nodes,
##                               trim = TRUE)
##
##           segment <- paste0("gruende-",
##                             1:length(gruende))
##
##           if (abw.vorhanden == TRUE){
##
##           abw.indizes <- abw.pos.first:length(segment)
##           segment[abw.indizes] <- paste0(segment[abw.indizes],
##                                           "-sondervotum")
##           }
##
##
##           dt.gruende <- data.table(gruende,
##                                     segment)
##
##       }else{
##           dt.gruende <- data.table("NA",
##                                     "NA")
##       }
##
##

```



```
##
##   ## Unterschriften
##   unterschriften <- paste(html_elements(HTML, "[class='st']") %>% html_
##     elements("td") %>% html_text(trim = TRUE),
##                               collapse = " ")
##
##   segment <- "unterschriften"
##   dt.unterschriften <- data.table(unterschriften,
##                                   segment)
##
##   ## Segmentierter Inhalt
##
##   inhalt <- rbind(dt.leitsaetze,
##                   dt.gegenstand,
##                   dt.formel,
##                   dt.tenor,
##                   dt.gruende,
##                   dt.unterschriften,
##                   use.names = FALSE)
##
##   setnames(inhalt,
##             new = c("text",
##                     "segment"))
##
##   inhalt <- inhalt[grep("NA", inhalt$segment, invert = TRUE)]
##
##   inhalt[, text := lapply(.(text), f.remove.specialunderline)]
##
##   return(inhalt)
## }

```

6.3 HTML-Dateien definieren

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

```

6.4 HTML-Dateien einlesen

```
html.list <- lapply(files.html,
                    read_html)

```

6.5 HTML-Dateien parsen

```
meta.list <- lapply(html.list,
                    f.bverfg.extract.meta)

```

```

content.list <- lapply(html.list,
                      f.bverfg.extract.content)

segmented.full.list <- vector("list",
                             length(meta.list))

for (i in 1:length(meta.list)){
  content.rows <- content.list[[i]][,.N]
  meta.replicate <- meta.list[[i]][rep(1, content.rows)]
  segmented.full.list[[i]] <- cbind(content.list[[i]],
                                   meta.replicate)
}

```

6.6 Data Table mit allen Metadaten (inkl. ECLI)

```
dt.meta.html <- rbindlist(meta.list)
```

6.7 Data Table mit vollständiger segmentierter Variante

```
dt.segmented.full <- rbindlist(segmented.full.list)
```

6.8 Special Character entfernen

An dieser Stelle wird ein mysteriöser Unterstrich entfernt, vermutlich ein non-breaking space. Es ist allerdings unklar wieso dieser in den Daten auftaucht. Der Code wird nicht im Compilation Report angezeigt, weil sich dieses Zeichen bei dem listings package zu Fehlern führt.

6.9 Stichprobe Metadaten

```

fwrite(dt.meta.html[sample(.N, 50)],
       file.path(dir.analysis,
                 "QA_Stichprobe_HTML-Metadaten.csv"))

```

6.10 Stichprobe Segmentierte Variante

```

fwrite(dt.segmented.full[sample(.N, 50)],
       file.path(dir.analysis,
                 "QA_Stichprobe_HTML-SegmentierterVolltext.csv"))

```

7 Text-Extraktion aus PDF

7.1 Vektor der zu extrahierenden Dateien erstellen

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

7.2 Anzahl zu extrahierender Dateien

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

```
## [1] 763
```

7.3 PDF extrahieren: Funktion anzeigen

```
print(f.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,  
                                       f.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))

```

```

}

```

7.4 Text Extrahieren

```

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

f.future_pdf_to_txt(files.pdf)

```

```

## Processing 763 files. Begin at: 2022-06-20 23:33:02

```

```

## Successfully processed 763 files. 0 files failed. Runtime was 4.03 secs. Ended
  at: 2022-06-20 23:33:06

```

8 Korpus Erstellen

8.1 TXT-Dateien Einlesen

```
txt.bverfg <- readtext("./*.txt",
  docvarsfrom = "filenames",
  docvarnames = c("gericht",
    "datum",
    "spruchkoerper_typ",
    "spruchkoerper_az",
    "registerzeichen",
    "eingangsnummer",
    "eingangsjahr_az",
    "kollision",
    "name",
    "band",
    "seite"),
  dvsep = "_",
  encoding = "UTF-8")
```

8.2 In Data Table umwandeln

```
setDT(txt.bverfg)
```

8.3 Durch Zeilenumbruch getrennte Wörter zusammenfügen

Durch Zeilenumbrüche getrennte Wörter stellen bei aus PDF-Dateien gewonnene Text-Korpora ein erhebliches Problem dar. Wörter werden dadurch in zwei sinnentleerte Tokens getrennt, statt ein einzelnes und sinnvolles Token zu bilden. Dieser Schritt entfernt die Bindestriche, den Zeilenumbruch und ggf. dazwischenliegende Leerzeichen.

8.3.1 Funktion anzeigen

```
print(f.hyphen.remove)
```

```
## function(text){
##   ## Examples: Ham-\nburg, Mei-\n  nungsäußerung
##   text.out <- gsub("([a-zöäüß])-[:blank:]*\n[:blank:]*([a-zöäüß])",
##     "\\1\\2",
##     text)
##   ## Examples: SARS-CoV-\n2
##   text.out <- gsub("([a-zA-ZöäüÖÄÜß])-[:blank:]*\n[:blank:]*([A-Z0-9ÖÄÜß
##   ])",
##     "\\1-\\2",
##     text.out)
##   ## Example: hat- 2\nte, Unsterb- 6\nliche
```

```
##      text.out <- gsub("([a-zöäüß])-[:blank:]*[0-9]+[:blank:]*\\n[:blank:]*
##      ([a-zöäüß])",
##                      "\\1\\2",
##                      text.out)
##
##      ## Example: hat-  \\n  2 te, Unsterb-  \\n  6 liche
##      text.out <- gsub("([a-zöäüß])-[:space:]*[0-9]+[:blank:]*([a-zöäüß])",
##                      "\\1\\2",
##                      text.out)
##
##      return(text.out)
## }
```

8.3.2 Funktion ausführen

```
txt.bverfg[, text := lapply(.text), f.hyphen.remove]]
dt.segmented.full[, text := lapply(.text), f.hyphen.remove]]
```

8.4 Variable “datum” als Datentyp “IDate” kennzeichnen

```
txt.bverfg$datum <- as.IDate(txt.bverfg$datum)
```

8.5 Variable “entscheidungsjahr” hinzufügen

```
txt.bverfg$entscheidungsjahr <- year(txt.bverfg$datum)
```

8.6 Variable “eingangsjahr_iso” hinzufügen

```
txt.bverfg$eingangsjahr_iso <- f.year.iso(txt.bverfg$eingangsjahr_az)
```

8.7 Datensatz nach Datum sortieren

Aufgrund der Position der Datums-Variable ist der Datensatz vermutlich schon von Linux nach Datum sortiert worden. Die Erstellung der Variablen für Präsidenten und Vize-Präsidenten trifft allerdings die starke Annahme, dass eine aufsteigende Sortierung nach Datum besteht. Wäre das nicht der Fall, würden dort Fehler auftreten. Diese Sortierung ist als fail-safe gedacht.

```
setorder(txt.bverfg,
          datum)
```

8.8 Variable “praesi” hinzufügen

Diese Variable dokumentiert für jede Entscheidung welche/r Präsident:in am Tag der Entscheidung im Amt war.

8.8.1 Lebensdaten einlesen

```
praesi <- fread(file.path("data",
                           "PVP-FCG_2021-04-08_GermanFederalCourts_Presidents.csv"
                           ))
praesi <- praesi[court == "BVerfG", c(1:3, 5:6)]
```

8.8.2 Personaldaten anzeigen

```
kable(praesi,
      format = "latex",
      align = "r",
      booktabs = TRUE,
      longtable = TRUE) %>% kable_styling(latex_options = "repeat_header")
```

court	name_last	name_first	term_begin_date	term_end_date
BVerfG	Höpker-Aschoff	Hermann	1951-09-07	1954-01-15
BVerfG	VACANCY-1	VACANCY-1	1954-01-16	1954-03-22
BVerfG	Wintrich	Josef	1954-03-23	1958-10-19
BVerfG	VACANCY-2	VACANCY-2	1958-10-20	1959-01-07
BVerfG	Müller	Gebhard	1959-01-08	1971-12-07
BVerfG	Benda	Ernst	1971-12-08	1983-12-19
BVerfG	Zeidler	Wolfgang	1983-12-20	1987-11-15
BVerfG	Herzog	Roman	1987-11-16	1994-06-30
BVerfG	VACANCY-3	VACANCY-3	1994-07-01	1994-09-13
BVerfG	Limbach	Jutta	1994-09-14	2002-04-09
BVerfG	Papier	Hans-Jürgen	2002-04-10	2010-03-15
BVerfG	Voßkuhle	Andreas	2010-03-16	2020-06-21
BVerfG	Harbarth	Stephan	2020-06-22	NA

8.8.3 Hypothetisches Amtsende für PräsidentIn

Weil der/die aktuelle PräsidentIn noch im Amt ist, ist der Wert für das Amtsende “NA”. Dieser ist aber für die verwendete Logik nicht greifbar, weshalb an dieser Stelle ein hypothetisches Amtsende in einem Jahr ab dem Tag der Datensatzerstellung fingiert wird. Es wird nur an dieser Stelle verwendet und danach verworfen.

```
praesi[is.na(term_end_date)]$term_end_date <- Sys.Date() + 365
```

8.8.4 Schleife vorbereiten

```
N <- praesi[, .N]
praesi.list <- vector("list", N)
```

8.8.5 Vektor erstellen

```
for (i in seq_len(N)){
  praesi.N <- txt.bverfg[datum >= praesi$term_begin_date[i] & datum <= praesi$
    term_end_date[i], .N]
  praesi.list[[i]] <- rep(praesi$name_last[i],
    praesi.N)
}
```

8.8.6 Vektor einfügen

```
txt.bverfg$praesi <- unlist(praesi.list)
```

8.9 Variable “v_praesi” hinzufügen

Diese Variable dokumentiert für jede Entscheidung welche/r Vize-PräsidentIn am Tag der Entscheidung im Amt war.

8.9.1 Personaldaten einlesen

```
vpraesi <- fread(file.path("data",
  "PVP-FCG_2021-04-08_GermanFederalCourts_VicePresidents.
  csv"))
vpraesi <- vpraesi[court == "BVerfG", c(1:3, 5:6)]
```


8.9.2 Personaldaten anzeigen

```
kable(vpraesi,  
      format = "latex",  
      align = "r",  
      booktabs = TRUE,  
      longtable = TRUE) %>% kable_styling(latex_options = "repeat_header")
```

court	name_last	name_first	term_begin_date	term_end_date
BVerfG	Katz	Rudolf	1951-09-07	1961-07-23
BVerfG	Wagner	Friedrich Wilhelm	1961-12-19	1967-10-18
BVerfG	Seuffert	Walter	1967-10-18	1975-11-07
BVerfG	Zeidler	Wolfgang	1975-11-07	1983-12-20
BVerfG	Herzog	Roman	1983-12-20	1987-11-16
BVerfG	Mahrenholz	Ernst Gottfried	1987-11-16	1994-03-24
BVerfG	Limbach	Jutta	1994-03-24	1994-09-14
BVerfG	Henschel	Johann Friedrich	1994-09-29	1995-10-13
BVerfG	Seidl	Otto	1995-10-13	1998-02-27
BVerfG	Papier	Hans-Jürgen	1998-02-27	2002-04-10
BVerfG	Hassemer	Winfried	2002-04-10	2008-05-07
BVerfG	Voßkuhle	Andreas	2008-05-07	2010-03-16
BVerfG	Kirchhof	Ferdinand	2010-03-16	2018-11-30
BVerfG	Harbarth	Stephan	2018-11-30	2020-06-22
BVerfG	König	Doris	2020-06-22	NA

8.9.3 Hypothetisches Amtsende für Vize-PräsidentIn

Weil der/die aktuelle Vize-PräsidentIn noch im Amt ist, ist der Wert für das Amtsende "NA". Dieser ist aber für die verwendete Logik nicht greifbar, weshalb an dieser Stelle ein hypothetisches Amtsende in einem Jahr ab dem Tag der Datensatzerstellung fingiert wird. Es wird nur an dieser Stelle verwendet und danach verworfen.

```
vpraesi[is.na(term_end_date)]$term_end_date <- Sys.Date() + 365
```

8.9.4 Schleife vorbereiten

```
N <- vpraesi[,.N]

vpraesi.list <- vector("list", N)
```

8.9.5 Vektor erstellen

```
for (i in seq_len(N)){
  vpraesi.N <- txt.bverfg[datum >= vpraesi$term_begin_date[i] & datum < vpraesi
    $term_end_date[i], .N]
  vpraesi.list[[i]] <- rep(vpraesi$name_last[i],
    vpraesi.N)
}
```

8.9.6 Vektor einfügen

```
txt.bverfg$v_praesi <- unlist(vpraesi.list)
```

8.10 Variable “verfahrensart” hinzufügen

Die Registerzeichen werden an dieser Stelle mit ihren detaillierten Bedeutungen aus dem folgenden Datensatz abgeglichen: “Seán Fobbe (2021). Aktenzeichen der Bundesrepublik Deutschland (AZ-BRD). Version 1.0.1. Zenodo. DOI: 10.5281/zenodo.4569564.” Das Ergebnis des Abgleichs wird in der Variable “verfahrensart” in den Datensatz eingefügt.

8.10.1 Datensatz einlesen

```
az.source <- fread(file.path("data",
  "AZ-BRD_1-0-1_DE_Registerzeichen_Datensatz.csv"))
```

8.10.2 Datensatz auf relevante Daten reduzieren

```
az.bverfg <- az.source[stelle == "BVerfG" & position == "hauptzeichen"]
```

8.10.3 Indizes bestimmen

```
targetindices <- match(txt.bverfg$registerzeichen,
  az.bverfg$zeichen_code)
```

8.10.4 Vektor der Verfahrensarten erstellen und einfügen

```
txt.bverfg$verfahrensart <- az.bverfg$bedeutung[targetindices]
```

8.11 Variable “aktenzeichen” hinzufügen

```
txt.bverfg$aktenzeichen <- paste0(txt.bverfg$spruchkoerper_az,  
                                   " ",  
                                   txt.bverfg$registerzeichen,  
                                   " ",  
                                   txt.bverfg$eingangsnummer,  
                                   "/" ,  
                                   txt.bverfg$eingangsjahr_az)
```

Bei Entscheidungen der Verzögerungskammer fehlt das Spruchkörper-Element des Aktenzeichens. Diese Zeile entfernt die “NA”-Angabe um ein korrektes Aktenzeichen herzustellen.

```
txt.bverfg$aktenzeichen <- gsub("NA ",  
                                "",  
                                txt.bverfg$aktenzeichen)
```

8.12 Variable “doi_concept” hinzufügen

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

8.13 Variable “doi_version” hinzufügen

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

8.14 Variable “version” hinzufügen

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

8.15 Variable “lizenz” hinzufügen

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

8.16 Variable “ecli” hinzufügen

Struktur und Inhalt der ECLI für deutsche Gerichte sind auf dem Europäischen Justizportal näher erläutert.²

Sofern die Variablen korrekt extrahiert wurden lässt sich die ECLI vollständig rekonstruieren.

```
ecli.ordinalzahl <- paste0(gsub("Bv([A-Z])",
                              "\\1",
                              txt.bverfg$registerzeichen),
                          txt.bverfg$spruchkoerper_typ,
                          txt.bverfg$datum,
                          txt.bverfg$kollision,
                          ".",
                          txt.bverfg$spruchkoerper_az,
                          txt.bverfg$registerzeichen,
                          formatC(txt.bverfg$eingangsnummer,
                                   width = 4,
                                   flag = "0"),
                          formatC(txt.bverfg$eingangsjahr_az,
                                   width = 2,
                                   flag = "0"))

ecli.ordinalzahl <- gsub("NA",
                      "",
                      ecli.ordinalzahl)

ecli.ordinalzahl <- gsub("-",
                      "",
                      ecli.ordinalzahl)

ecli.ordinalzahl <- tolower(ecli.ordinalzahl)

ecli.ordinalzahl <- gsub("vzb",
                      "vb",
                      ecli.ordinalzahl)

ecli.ordinalzahl <- gsub("pup",
                      "up",
                      ecli.ordinalzahl)

txt.bverfg$ecli <- paste0("ECLI:DE:BVerfG:",
                        txt.bverfg$entscheidungsjahr,
                        ":",
                        ecli.ordinalzahl)
```

8.16.1 Fehlerhafte ECLI korrigieren (HTML Meta)

² https://e-justice.europa.eu/content_european_case_law_identifier_ecli-175-de-de.do?member=1

```
dt.meta.html$ecli <- gsub("ECLI:DE:BVerfG:1951:rs19580115.1bvr040051",
                        "ECLI:DE:BVerfG:1958:rs19580115.1bvr040051",
                        dt.meta.html$ecli) ## Lüth

dt.meta.html$ecli <- gsub("ECLI:DE:BVerfG:2003:rk20030407.2bvr212902",
                        "ECLI:DE:BVerfG:2003:rk20030407.1bvr212902",
                        dt.meta.html$ecli)

dt.meta.html$ecli <- gsub("ECLI:DE:BVerfG:2007:rk20060529.1bvr043003",
                        "ECLI:DE:BVerfG:2006:rk20060529.1bvr043003",
                        dt.meta.html$ecli)
```

Die folgende ECLI ist auf der Homepage des BVerfG fehlerhaft. Sie betrifft das Vorverfahren statt die Verzögerungsbeschwerde. Auf [rechtsprechung-im-internet.de](https://www.rechtsprechung-im-internet.de) ist sie korrekt nachgewiesen.³

```
dt.meta.html$ecli <- gsub("ECLI:DE:BVerfG:2015:rs20151208a.1bvr009911",
                        "ECLI:DE:BVerfG:2015:vb20151208.vz000115",
                        dt.meta.html$ecli)
```

8.16.2 Fehlerhafte ECLI korrigieren (HTML Full)

```
dt.segmented.full$ecli <- gsub("ECLI:DE:BVerfG:1951:rs19580115.1bvr040051",
                             "ECLI:DE:BVerfG:1958:rs19580115.1bvr040051",
                             dt.segmented.full$ecli) ## Lüth

dt.segmented.full$ecli <- gsub("ECLI:DE:BVerfG:2003:rk20030407.2bvr212902",
                             "ECLI:DE:BVerfG:2003:rk20030407.1bvr212902",
                             dt.segmented.full$ecli)

dt.segmented.full$ecli <- gsub("ECLI:DE:BVerfG:2007:rk20060529.1bvr043003",
                             "ECLI:DE:BVerfG:2006:rk20060529.1bvr043003",
                             dt.segmented.full$ecli)
```

Die folgende ECLI ist auf der Homepage des BVerfG fehlerhaft. Sie betrifft das Vorverfahren statt die Verzögerungsbeschwerde. Auf [rechtsprechung-im-internet.de](https://www.rechtsprechung-im-internet.de) ist sie korrekt nachgewiesen.⁴

```
dt.segmented.full$ecli <- gsub("ECLI:DE:BVerfG:2015:rs20151208a.1bvr009911",
                             "ECLI:DE:BVerfG:2015:vb20151208.vz000115",
                             dt.segmented.full$ecli)
```

³ <https://www.rechtsprechung-im-internet.de/jportal/?quelle=jlink&docid=KVRE412291501&psml=bsjrsprod.psml&max=true>

⁴ <https://www.rechtsprechung-im-internet.de/jportal/?quelle=jlink&docid=KVRE412291501&psml=bsjrsprod.psml&max=true>

8.16.3 ECLI-Test 1: ECLI die in PDF, aber nicht in HTML vorhanden sind

```
sort(setdiff(txt.bverfg$ecli, dt.segmented.full$ecli))
```

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

8.16.4 ECLI-Test 2: ECLI die in HTML, aber nicht in PDF vorhanden sind

```
sort(setdiff(dt.segmented.full$ecli, txt.bverfg$ecli))
```

```
## [1] "ECLI:DE:BVerfG:2001:bs20011001.2bvb000101"
```

8.16.5 Zum Vergleich: Beim Download fehlende PDF-Dateien

```
print(sort(missing.pdf))
```

```
## [1] "BVerfG_2001-10-01_S_2_BvB_0001_01_NA_NPD-Verbot-EA-RückgabeEDV-2_104_61.pdf"
```

8.16.6 ECLI Merge: Metadaten aus Hauptdatensatz in segmentierte Variante mergen

```
meta.bverfg <- txt.bverfg[,!"text"]

dt.segmented.full <- merge(dt.segmented.full,
                           meta.bverfg,
                           by = "ecli",
                           all.x = TRUE,
                           sort = FALSE)
```

8.16.7 ECLI Merge: Metadaten aus HTML-Extraktion in Hauptdatensatz mergen

```
txt.bverfg <- merge(txt.bverfg,
                   dt.meta.html,
                   by = "ecli",
                   all.x = TRUE,
                   sort = FALSE)
```

8.17 Variable “entscheidung_typ” hinzufügen

8.17.1 Zitiervorschläge parsen

```
entscheidung_typ.main <- gsub(".*(Beschluss|Urteil|Verfügung).*",  
                             "\\1",  
                             txt.bverfg$zitiervorschlag,  
                             ignore.case = TRUE)  
  
entscheidung_typ.segmented <- gsub(".*(Beschluss|Urteil|Verfügung).*",  
                                   "\\1",  
                                   dt.segmented.full$zitiervorschlag,  
                                   ignore.case = TRUE)
```

8.17.2 Kürzen

```
lang.etyt <- c("Urteil",  
              "Beschluss",  
              "Verfügung")  
  
kurz.etyt <- c("U",  
              "B",  
              "V")  
  
entscheidung_typ.main <- mgsub(entscheidung_typ.main,  
                               lang.etyt,  
                               kurz.etyt,  
                               ignore.case = TRUE)  
  
entscheidung_typ.segmented <- mgsub(entscheidung_typ.segmented,  
                                    lang.etyt,  
                                    kurz.etyt,  
                                    ignore.case = TRUE)
```

8.17.3 Vektor in Datensatz einfügen

```
txt.bverfg$entscheidung_typ <- entscheidung_typ.main  
  
dt.segmented.full$entscheidung_typ <- entscheidung_typ.segmented
```

9 Frequenztabellen erstellen

9.1 Funktion anzeigen

```
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,
           file.path(outputdir,
                     paste0(prefix,
                             varname,
```



```

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

}

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

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

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

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

}

```

9.2 Ignorierte Variablen

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

```
## [1] "text"          "eingangsnummer" "datum"          "doc_id"
## [5] "seite"         "name"           "ecli"           "aktenzeichen"
## [9] "pressemitteilung" "zitiervorschlag" "kurzbeschreibung"
```

9.3 Liste zu prüfender Variablen

```
varlist <- names(txt.bverfg)
varlist <- grep(paste(config$freqtable$ignore,
                      collapse = "|"),
               varlist,
               invert = TRUE,
               value = TRUE)
print(varlist)
```

```
## [1] "gericht"      "spruchkoerper_typ" "spruchkoerper_az"
## [4] "registerzeichen" "eingangsjahr_az"   "kollision"
## [7] "band"         "entscheidungsjahr" "eingangsjahr_iso"
## [10] "praesi"       "v_praesi"          "verfahrensart"
## [13] "doi_concept"  "doi_version"        "version"
## [16] "lizenz"       "richter"            "entscheidung_typ"
```

9.4 Frequenztabellen erstellen

```
prefix <- paste0(config$project$shortname,
  "_01_Frequenztafel_var-")
```

```
f.fast.freqtable(txt.bverfg,
  varlist = varlist,
  sumrow = TRUE,
  output.list = FALSE,
  output.kable = TRUE,
  output.csv = TRUE,
  outputdir = dir.analysis,
  prefix = prefix,
  align = c("p{5cm}",
    rep("r", 4)))
```

Frequency Table for Variable: gericht

1 unique value(s) detected.

gericht	N	exactpercent	roundedpercent	cumulpercent
BVerfG	763	100	100	100
Total	763	100	100	100

Frequency Table for Variable: spruchkoerper_typ

3 unique value(s) detected.

spruchkoerper_typ	N	exactpercent	roundedpercent	cumulpercent
K	3	0.3931848	0.39	0.39
P	2	0.2621232	0.26	0.66

(continued)

spruchkoerper_typ	N	exactpercent	roundedpercent	cumulpercent
S	758	99.3446920	99.34	100.00
Total	763	100.0000000	100.00	100.00

Frequency Table for Variable: spruchkoerper_az

2 unique value(s) detected.

spruchkoerper_az	N	exactpercent	roundedpercent	cumulpercent
1	338	44.29882	44.3	44.3
2	425	55.70118	55.7	100.0
Total	763	100.00000	100.0	100.0

Frequency Table for Variable: registerzeichen

14 unique value(s) detected.

registerzeichen	N	exactpercent	roundedpercent	cumulpercent
BvB	17	2.2280472	2.23	2.23
BvC	24	3.1454784	3.15	5.37
BvE	81	10.6159895	10.62	15.99
BvF	51	6.6841415	6.68	22.67
BvG	8	1.0484928	1.05	23.72
BvH	4	0.5242464	0.52	24.25
BvK	9	1.1795544	1.18	25.43
BvL	150	19.6592398	19.66	45.09
BvM	4	0.5242464	0.52	45.61
BvN	1	0.1310616	0.13	45.74
BvP	1	0.1310616	0.13	45.87
BvQ	22	2.8833552	2.88	48.75

(continued)

registerzeichen	N	exactpercent	roundedpercent	cumulpercent
BvR	389	50.9829620	50.98	99.74
PBvU	2	0.2621232	0.26	100.00
Total	763	100.0000000	100.00	100.00

Frequency Table for Variable: eingangsjahr_az

39 unique value(s) detected.

eingangsjahr_az	N	exactpercent	roundedpercent	cumulpercent
0	27	3.5386632	3.54	3.54
1	36	4.7182176	4.72	8.26
2	33	4.3250328	4.33	12.58
3	31	4.0629096	4.06	16.64
4	28	3.6697248	3.67	20.31
5	29	3.8007864	3.80	24.12
6	19	2.4901704	2.49	26.61
7	39	5.1114024	5.11	31.72
8	23	3.0144168	3.01	34.73
9	29	3.8007864	3.80	38.53
10	27	3.5386632	3.54	42.07
11	39	5.1114024	5.11	47.18
12	38	4.9803408	4.98	52.16
13	36	4.7182176	4.72	56.88
14	24	3.1454784	3.15	60.03
15	31	4.0629096	4.06	64.09
16	15	1.9659240	1.97	66.06
17	17	2.2280472	2.23	68.28
18	15	1.9659240	1.97	70.25
19	15	1.9659240	1.97	72.21

(continued)

eingangsjahr_az	N	exactpercent	roundedpercent	cumulpercent
20	7	0.9174312	0.92	73.13
21	4	0.5242464	0.52	73.66
51	1	0.1310616	0.13	73.79
52	2	0.2621232	0.26	74.05
83	2	0.2621232	0.26	74.31
86	1	0.1310616	0.13	74.44
87	1	0.1310616	0.13	74.57
88	1	0.1310616	0.13	74.71
89	3	0.3931848	0.39	75.10
90	7	0.9174312	0.92	76.02
91	13	1.7038008	1.70	77.72
92	6	0.7863696	0.79	78.51
93	17	2.2280472	2.23	80.73
94	20	2.6212320	2.62	83.36
95	23	3.0144168	3.01	86.37
96	28	3.6697248	3.67	90.04
97	21	2.7522936	2.75	92.79
98	31	4.0629096	4.06	96.85
99	24	3.1454784	3.15	100.00
Total	763	100.0000000	100.00	100.00

Frequency Table for Variable: kollision

4 unique value(s) detected.

kollision	N	exactpercent	roundedpercent	cumulpercent
NA	753	98.6893840	98.69	98.69
a	7	0.9174312	0.92	99.61
b	2	0.2621232	0.26	99.87

(continued)

kollision	N	exactpercent	roundedpercent	cumulpercent
c	1	0.1310616	0.13	100.00
Total	763	100.0000000	100.00	100.00

Frequency Table for Variable: band

70 unique value(s) detected.

band	N	exactpercent	roundedpercent	cumulpercent
1	2	0.2621232	0.26	0.26
7	1	0.1310616	0.13	0.39
62	1	0.1310616	0.13	0.52
65	1	0.1310616	0.13	0.66
90	1	0.1310616	0.13	0.79
91	1	0.1310616	0.13	0.92
93	1	0.1310616	0.13	1.05
96	5	0.6553080	0.66	1.70
97	18	2.3591088	2.36	4.06
98	14	1.8348624	1.83	5.90
99	26	3.4076016	3.41	9.31
100	14	1.8348624	1.83	11.14
101	15	1.9659240	1.97	13.11
102	17	2.2280472	2.23	15.33
103	19	2.4901704	2.49	17.82
104	26	3.4076016	3.41	21.23
105	14	1.8348624	1.83	23.07
106	17	2.2280472	2.23	25.29
107	14	1.8348624	1.83	27.13
108	18	2.3591088	2.36	29.49
109	12	1.5727392	1.57	31.06

(continued)

band	N	exactpercent	roundedpercent	cumulpercent
110	14	1.8348624	1.83	32.90
111	15	1.9659240	1.97	34.86
112	18	2.3591088	2.36	37.22
113	10	1.3106160	1.31	38.53
114	13	1.7038008	1.70	40.24
115	12	1.5727392	1.57	41.81
116	11	1.4416776	1.44	43.25
117	14	1.8348624	1.83	45.09
118	9	1.1795544	1.18	46.26
119	10	1.3106160	1.31	47.58
120	10	1.3106160	1.31	48.89
121	10	1.3106160	1.31	50.20
122	13	1.7038008	1.70	51.90
123	9	1.1795544	1.18	53.08
124	12	1.5727392	1.57	54.65
125	7	0.9174312	0.92	55.57
126	13	1.7038008	1.70	57.27
127	10	1.3106160	1.31	58.58
128	13	1.7038008	1.70	60.29
129	12	1.5727392	1.57	61.86
130	11	1.4416776	1.44	63.30
131	11	1.4416776	1.44	64.74
132	13	1.7038008	1.70	66.45
133	14	1.8348624	1.83	68.28
134	14	1.8348624	1.83	70.12
135	11	1.4416776	1.44	71.56
136	11	1.4416776	1.44	73.00
137	9	1.1795544	1.18	74.18
138	12	1.5727392	1.57	75.75

(continued)

band	N	exactpercent	roundedpercent	cumulpercent
139	10	1.3106160	1.31	77.06
140	12	1.5727392	1.57	78.64
141	8	1.0484928	1.05	79.69
142	13	1.7038008	1.70	81.39
143	8	1.0484928	1.05	82.44
144	4	0.5242464	0.52	82.96
145	9	1.1795544	1.18	84.14
146	7	0.9174312	0.92	85.06
147	9	1.1795544	1.18	86.24
148	10	1.3106160	1.31	87.55
149	13	1.7038008	1.70	89.25
150	8	1.0484928	1.05	90.30
151	9	1.1795544	1.18	91.48
152	12	1.5727392	1.57	93.05
153	6	0.7863696	0.79	93.84
154	7	0.9174312	0.92	94.76
155	8	1.0484928	1.05	95.81
156	10	1.3106160	1.31	97.12
157	8	1.0484928	1.05	98.17
158	14	1.8348624	1.83	100.00
Total	763	100.0000000	100.00	100.00

Frequency Table for Variable: entscheidungsjahr

30 unique value(s) detected.

entscheidungsjahr	N	exactpercent	roundedpercent	cumulpercent
1952	2	0.2621232	0.26	0.26
1958	1	0.1310616	0.13	0.39

(continued)

entscheidungsjahr	N	exactpercent	roundedpercent	cumulpercent
1983	2	0.2621232	0.26	0.66
1994	2	0.2621232	0.26	0.92
1996	1	0.1310616	0.13	1.05
1997	9	1.1795544	1.18	2.23
1998	51	6.6841415	6.68	8.91
1999	31	4.0629096	4.06	12.98
2000	20	2.6212320	2.62	15.60
2001	37	4.8492792	4.85	20.45
2002	37	4.8492792	4.85	25.29
2003	38	4.9803408	4.98	30.28
2004	41	5.3735256	5.37	35.65
2005	38	4.9803408	4.98	40.63
2006	25	3.2765400	3.28	43.91
2007	27	3.5386632	3.54	47.44
2008	29	3.8007864	3.80	51.25
2009	29	3.8007864	3.80	55.05
2010	29	3.8007864	3.80	58.85
2011	24	3.1454784	3.15	61.99
2012	35	4.5871560	4.59	66.58
2013	28	3.6697248	3.67	70.25
2014	37	4.8492792	4.85	75.10
2015	29	3.8007864	3.80	78.90
2016	29	3.8007864	3.80	82.70
2017	27	3.5386632	3.54	86.24
2018	30	3.9318480	3.93	90.17
2019	21	2.7522936	2.75	92.92
2020	30	3.9318480	3.93	96.85
2021	24	3.1454784	3.15	100.00
Total	763	100.0000000	100.00	100.00

Frequency Table for Variable: eingangsjahr_iso

39 unique value(s) detected.

eingangsjahr_iso	N	exactpercent	roundedpercent	cumulpercent
1951	1	0.1310616	0.13	0.13
1952	2	0.2621232	0.26	0.39
1983	2	0.2621232	0.26	0.66
1986	1	0.1310616	0.13	0.79
1987	1	0.1310616	0.13	0.92
1988	1	0.1310616	0.13	1.05
1989	3	0.3931848	0.39	1.44
1990	7	0.9174312	0.92	2.36
1991	13	1.7038008	1.70	4.06
1992	6	0.7863696	0.79	4.85
1993	17	2.2280472	2.23	7.08
1994	20	2.6212320	2.62	9.70
1995	23	3.0144168	3.01	12.71
1996	28	3.6697248	3.67	16.38
1997	21	2.7522936	2.75	19.13
1998	31	4.0629096	4.06	23.20
1999	24	3.1454784	3.15	26.34
2000	27	3.5386632	3.54	29.88
2001	36	4.7182176	4.72	34.60
2002	33	4.3250328	4.33	38.93
2003	31	4.0629096	4.06	42.99
2004	28	3.6697248	3.67	46.66
2005	29	3.8007864	3.80	50.46
2006	19	2.4901704	2.49	52.95
2007	39	5.1114024	5.11	58.06
2008	23	3.0144168	3.01	61.07

(continued)

eingangsjahr_iso	N	exactpercent	roundedpercent	cumulpercent
2009	29	3.8007864	3.80	64.88
2010	27	3.5386632	3.54	68.41
2011	39	5.1114024	5.11	73.53
2012	38	4.9803408	4.98	78.51
2013	36	4.7182176	4.72	83.22
2014	24	3.1454784	3.15	86.37
2015	31	4.0629096	4.06	90.43
2016	15	1.9659240	1.97	92.40
2017	17	2.2280472	2.23	94.63
2018	15	1.9659240	1.97	96.59
2019	15	1.9659240	1.97	98.56
2020	7	0.9174312	0.92	99.48
2021	4	0.5242464	0.52	100.00
Total	763	100.0000000	100.00	100.00

Frequency Table for Variable: praesi

8 unique value(s) detected.

praesi	N	exactpercent	roundedpercent	cumulpercent
Benda	2	0.2621232	0.26	0.26
Harbarth	37	4.8492792	4.85	5.11
Herzog	1	0.1310616	0.13	5.24
Höpker-Aschoff	2	0.2621232	0.26	5.50
Limbach	164	21.4941022	21.49	27.00
Papier	253	33.1585845	33.16	60.16
Voßkuhle	303	39.7116645	39.71	99.87
Wintrich	1	0.1310616	0.13	100.00
Total	763	100.0000000	100.00	100.00

Frequency Table for Variable: v_praesi

11 unique value(s) detected.

v_praesi	N	exactpercent	roundedpercent	cumulpercent
Harbarth	42	5.5045872	5.50	5.50
Hassemer	207	27.1297510	27.13	32.63
Henschel	1	0.1310616	0.13	32.77
Katz	3	0.3931848	0.39	33.16
Kirchhof	261	34.2070773	34.21	67.37
König	37	4.8492792	4.85	72.21
Mahrenholz	1	0.1310616	0.13	72.35
Papier	144	18.8728702	18.87	91.22
Seidl	19	2.4901704	2.49	93.71
Voßkuhle	46	6.0288336	6.03	99.74
Zeidler	2	0.2621232	0.26	100.00
Total	763	100.0000000	100.00	100.00

Frequency Table for Variable: verfahrensart

14 unique value(s) detected.

verfahrensart	N	exactpercent	roundedpercent	cumulpercent
Abstrakte Normenkontrolle	51	6.6841415	6.68	6.68
Andere Streitigkeiten zwischen Bund und Ländern	4	0.5242464	0.52	7.21
Bund-Länder-Streitigkeiten	8	1.0484928	1.05	8.26
Divergenzvorlagen eines Landesverfassungsgerichts zur Auslegung des Grundgesetzes	1	0.1310616	0.13	8.39
Einstweilige Anordnungen	22	2.8833552	2.88	11.27

(continued)

verfahrensart	N	exactpercent	roundedpercent	cumulpercent
Feststellung der Anwendbarkeit einer Regel des Völkerge- wohnheitsrechts	4	0.5242464	0.52	11.80
Konkrete Normenkontrolle	150	19.6592398	19.66	31.45
Landesverfassungsstreitigkeiten	9	1.1795544	1.18	32.63
Organstreitverfahren	81	10.6159895	10.62	43.25
Plenarentscheidungen	2	0.2621232	0.26	43.51
Sonstige durch Bundesrecht zugewiesene Verfahren	1	0.1310616	0.13	43.64
Verfassungsbeschwerden; Kommunalverfassungsbe- schwerden	389	50.9829620	50.98	94.63
Verfassungswidrigkeit von Par- teien	17	2.2280472	2.23	96.85
Wahlprüfungsverfahren	24	3.1454784	3.15	100.00
Total	763	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.3831111	763	100	100	100
Total	763	100	100	100

Frequency Table for Variable: doi_version

1 unique value(s) detected.

doi_version	N	exactpercent	roundedpercent	cumulpercent
10.5281/zenodo.6569896	763	100	100	100

(continued)

doi_version	N	exactpercent	roundedpercent	cumulpercent
Total	763	100	100	100

Frequency Table for Variable: version

1 unique value(s) detected.

version	N	exactpercent	roundedpercent	cumulpercent
2022-06-20	763	100	100	100
Total	763	100	100	100

Frequency Table for Variable: lizenz

1 unique value(s) detected.

lizenz	N	exactpercent	roundedpercent	cumulpercent
Creative Commons Zero 1.0 Universal	763	100	100	100
Total	763	100	100	100

Frequency Table for Variable: richter

141 unique value(s) detected.

richter	N	exactpercent	roundedpercent	cumulpercent
	12	1.5727392	1.57	1.57
Benda Simon Hesse Katzenstein Niemeyer Hess Henschel Niedner	1	0.1310616	0.13	1.70
Broß Osterloh Di Fabio Mellinghoff Lübbe-Wolff Gerhardt Landau	1	0.1310616	0.13	1.83

(continued)

richter	N	exactpercent	roundedpercent	cumulpercent
Bundesverfassungsrichter Lim- bach Kirchhof Sommer Jentsch Hassemer Broß Osterloh Winter ist an der Unter- schrift verhindert.	1	0.1310616	0.13	1.97
Dr. Dr. Höpker- Aschoff Ellinghaus Dr. Heiland Dr. Drath Dr. Stein Wessel Ritterspach Lehmann Dr. Zweigert	1	0.1310616	0.13	2.10
Dr. Dr. Höpker- Aschoff Ellinghaus Dr. Scheffler Dr. Hei- land Dr. Scholtissek Dr. Drath Wessel Ritterspach Lehmann Dr. Zweigert	1	0.1310616	0.13	2.23
Eichberger Masing Paulus Baer Britz Ott Christ Gaier Eichberger Schluckebier Masing Paulus Britz	2	0.2621232	0.26	2.62
Gaier Eichberger Schluckebier Masing Paulus Hofmanns Baer Britz	1	0.1310616	0.13	2.75
Gaier Masing Baer	1	0.1310616	0.13	2.88
Grimm Kühling Jaeger Haas Hömig Steine Grimm Kühling Jaeger Haas Hömig Steine	1	0.1310616	0.13	3.01
Grimm Kühling Jaeger Haas Hömig Steine	1	0.1310616	0.52	3.54
Grimm Kühling Seibert Jaeger Hömig Steine	1	0.1310616	0.13	3.67
Harbarth Masing Paulus Baer Britz Ott Christ Harbarth Paulus Baer Britz Ott Christ	1	0.1310616	1.18	4.85
Harbarth Paulus Baer Britz Ott Christ	1	0.1310616	0.13	4.98
Harbarth Paulus Baer Britz Ott Christ	1	0.1310616	1.05	6.03
Hassemer Broß Di Fabio Lübbe-Wolff Gerhardt	1	0.1310616	0.13	6.16
Hassemer Broß Osterloh Di Fabio Mellinghoff Lübbe- Wolff	1	0.1310616	0.13	6.29
Hassemer Broß Osterloh Di Fabio Mellinghoff Lübbe- Wolff Gerhardt Landau	32	4.1939712	4.19	10.48

(continued)

richter	N	exactpercent	roundedpercent	cumulpercent
Hassemer Jaeger Broß Osterloh Di Fabio Mellinghoff Lübbe-Wolff	1	0.1310616	0.13	10.62
Hassemer Jentsch Broß Di Fabio Mellinghoff Lübbe-Wolff	1	0.1310616	0.13	10.75
Hassemer Jentsch Broß Osterloh Di Fabio Mellinghoff Lübbe-Wolff Gerhard	1	0.1310616	0.13	10.88
Hassemer Jentsch Broß Osterloh Di Fabio Mellinghoff Lübbe-Wolff Gerhardt	37	4.8492792	4.85	15.73
Hassemer Jentsch Broß Osterloh Mellinghoff Lübbe-Wolff Gerhardt	1	0.1310616	0.13	15.86
Hassemer Jentsch Richter Broß ist an der Osterloh Di Fabio Mellinghoff Lübbe-Wolff Gerhardt Unterschrift verhindert.	1	0.1310616	0.13	15.99
Hassemer Jentsch Richter Broß ist an Osterloh Richter Di Fabio ist Mellinghoff Lübbe-Wolff Gerhardt der Unterschrift verhindert. an	1	0.1310616	0.13	16.12
Hassemer Sommer Broß Osterloh Di Fabio Mellinghoff Lübbe-Wolff	1	0.1310616	0.13	16.25
Hassemer Sommer Jentsch Broß Osterloh Di Fabio Mellinghoff	2621232	0.26	0.26	16.51
Hassemer Sommer Jentsch Broß Osterloh Di Fabio Mellinghoff Lübbe-Wolff	4901704	2.49	2.49	19.00
Hassemer Sommer Jentsch Broß Osterloh Di Fabio Mellinghoff Lübbe-Wolff Di Fabio	1310616	0.13	0.13	19.13
Hassemer Sommer Jentsch Osterloh Di Fabio Mellinghoff Lübbe-Wolff	1	0.1310616	0.13	19.27

(continued)

richter	N	exactpercent	roundedpercent	cumulpercent
Hassemer Sommer Jentsch Richter Broß ist Osterloh Di Fabio Mellinghoff Lübbe- Wolff an der Unterschrift gehindert. Hassemer	1	0.1310616	0.13	19.40
Hentschel Seidl Grimm Söllner Kühling Seibert Haas	1	0.1310616	0.13	19.53
Herzog Henschel Seidl Grimm Söllner Dietrich Kühling Seibert	1	0.1310616	0.13	19.66
Jaeger Frau Haas ist gehindert, ihre Hömig Steiner Hohmann- Dennhardt Hoffmann- Riem Bryde Lübbe-Wolff Unterschrift beizufügen. Jaeger	1	0.1310616	0.13	19.79
Jaeger Haas Hömig Steiner Hohmann- Dennhardt Hoffmann- Riem Bryde	1	0.1310616	0.13	19.92
Jentsch Broß Osterloh Di Fabio Mellinghoff Lübbe- Wolff Gerhardt	1	0.1310616	0.13	20.05
Kirchhof Eichberger Masing Paulus Baer Christ	1	0.2621232	0.26	20.31
Kirchhof Eichberger Masing Paulus Baer Christ	1	0.2621232	1.05	21.36
Kirchhof Eichberger Schluckebier Masing Paulus Baer Britz	1	0.3931818	0.39	21.76
Kirchhof Eichberger Schluckebier Masing Paulus Baer Britz Ott	1	0.6653030	0.66	22.41
Kirchhof Gaier Eichberger Schluckebier Masing Paulus Baer Britz	1	7.9947571	7.99	30.41
Kirchhof Hohmann- Dennhardt Bryde Gaier Eichberger Schluckebier Masing	2	0.2621232	0.26	30.67
Kirchhof Hohmann- Dennhardt Bryde Gaier Eichberger Schluckebier Masing Paulus	16	2.0969856	2.10	32.77
Kirchhof Masing Paulus Baer Britz Ott Christ Schluckebier	1	0.1310616	0.13	32.90
Kruis	1	0.1310616	0.13	33.03
König Hermanns Müller Kessal- Wulf Maidowski Langenfeld Wallrabenstein	2	0.2621232	0.26	33.29
König Huber Hermanns Kessal- Wulf Maidowski Langenfeld Wallrabenstein	2	0.2621232	0.26	33.55
König Huber Hermanns Müller Kessal- Wulf Maidowski Langenfeld	4	0.5242464	0.52	34.08

(continued)

richter	N	exactpercent	roundedpercent	cumulpercent
König Huber Hermanns Müller Kessal- Wulf Maidowski Langenfeld Wallrabenstein	141	1.8348624	1.83	35.91
König Huber Hermanns Müller Maidowski Langenfeld Härtel	131	0.1310616	0.13	36.04
König Huber Hermanns Müller Maidowski Langenfeld Wallrabenstein	126	0.2621232	0.26	36.30
König Huber Müller Kessal- Wulf Maidowski Langenfeld	2	0.2621232	0.26	36.57
Kühling Jaeger Haas Hömig Steiner Hohmann- Dennhardt Hoffmann-Riem	113	0.1310616	0.13	36.70
Kühling Jaeger Haas Hömig Steiner Hohmann- Dennhardt Hoffmann- Riem Di Fabio	113	0.1310616	0.13	36.83
Landau Huber Hermanns Müller Kessal- Wulf König Maidowski	113	0.1310616	0.13	36.96
Limbach Graßhof Kruis Kirchhof Winter Sommer Hassemer	113	0.1310616	0.13	37.09
Limbach Graßhof Kruis Kirchhof Winter Sommer Jentsch	113	0.1310616	0.13	37.22
Limbach Graßhof Kruis Kirchhof Winter Sommer Jentsch Hassemer	113	0.348624	1.83	39.06
Limbach Hassemer Broß	1	0.1310616	0.13	39.19
Limbach Kirchhof Graßhof	1	0.1310616	0.13	39.32
Limbach Kirchhof Jentsch Hassemer Broß	113	0.1310616	0.13	39.45
Limbach Kirchhof Sommer Jentsch Hassemer Broß Osterloh	113	0.348624	0.39	39.84
Limbach Kirchhof Winter Sommer Jentsch Broß Osterloh	113	0.1310616	0.13	39.97
Limbach Kirchhof Winter Sommer Jentsch Hassemer Broß Osterloh	113	0.1310616	0.13	40.10
Limbach Kirchhof Winter Sommer Jentsch Hassemer Broß	113	0.5242464	0.52	40.63
Limbach Kirchhof Winter Sommer Jentsch Hassemer Broß Osterloh	113	0.446776	1.44	42.07
Limbach Sommer Hassemer Broß Osterloh Di Fabio	113	0.1310616	0.13	42.20
Limbach Sommer Jentsch Hassemer Broß Osterloh Di Fabio	113	0.1310616	0.13	42.33
Limbach Sommer Jentsch Hassemer Broß Osterloh Di Fabio	113	0.7038008	1.70	44.04
Limbach Sommer Jentsch Hassemer Broß Osterloh Di Fabio Mellinghoff	210	2.6212320	2.62	46.66

(continued)

richter	N	exactpercent	roundedpercent	cumulpercent
Limbach Sommer Jentsch Hassemer Herr Broß ist an der Osterloh Di Fabio Unterschrift gehindert. Limbach	1	0.1310616	0.13	46.79
Limbach Winter Sommer Jentsch Hassemer Briß Osterloh	1	0.1310616	0.13	46.92
Lübbe- Wolff Gerhardt Landau Huber Hermanns Müller Kessal- Wulf	4	0.5242464	0.52	47.44
Masing Paulus Baer Britz Ott Chris Radtke	3	0.3931848	0.39	47.84
Papier Grimm Kühling Seibert Jaeger Haas Hömig Steiner Frau BVRin Seibert ist aus dem Amt ausgeschieden.	1	0.1310616	0.13	47.97
Papier Der Richter ter Grimm ist aus dem Kühling Jaeger Haas Hömig Steiner Hohmann- Dennhardt Amt ausgeschie- den und deshalb an der Unterschrift gehindert. Papier	2	0.2621232	0.26	48.23
Papier Der Richter Kh- ling ist aus dem Amt geschieden und daher ge- hindert zu unterschreiben Papier Jaeger Haas Hmig Steiner Hohmann- Dennhardt Hoffmann-Riem	1	0.1310616	0.13	48.36
Papier Der Richter Kühling ist aus Jaeger Haas Hömig Steiner Hohmann- Dennhardt Hoffmann-Riem dem Amt geschieden und daher gehindert zu unter- schreiben. Papier	1	0.1310616	0.13	48.49
Papier Der Richter Steine ist Hohmann- Dennhardt Hoffmann- Riem Bryde Gaier Eichberger Schluckebier ausgeschieden und daher der Unterschrift gehindert. Papier	1	0.1310616	0.13	48.62

(continued)

richter	N	exactpercent	roundedpercent	cumulpercent
Papier Die Richterin Haas ist aus dem Steiner Hohmann-Dennhardt Hoffmann-Riem Bryde Gaier Eichberger Amt ausgeschieden und deshalb an der Unterschrift gehindert. Papier	3	0.3931848	0.39	49.02
Papier Die Richterin Haas Steiner Hohmann-Dennhardt Hoffmann-Riem Bryde Gaier Eichberger ist aus dem Amt ausge- schieden und deshalb an der Unterschrift gehindert. Papier	1	0.1310616	0.13	49.15
Papier Die Richterin Jaeger ist aus Haas Hömig Steiner Hohmann-Dennhardt Hoffmann-Riem dem Amt ausgeschieden und daher an der Unterschrift gehindert Papier	1	0.1310616	0.13	49.28
Papier Die Richterin Jaeger ist aus Haas Hömig Steiner Hohmann-Dennhardt Hoffmann-Riem dem Amt ausgeschieden und daher an der Unterschrift gehindert. Papier	1	0.1310616	0.13	49.41
Papier Die Richterin Jaeger ist aus Haas Hömig Steiner Hohmann-Dennhardt Hoffmann-Riem Bryde dem Amt ausgeschieden und deshalb an der Unterschrift gehindert. Papier	1	0.1310616	0.13	49.54
Papier Die Richterin Haas Hömig Steiner Hohmann-Dennhardt Hoffmann-Riem Bryde Jaeger ist aus dem Amt ausgeschieden und deshalb an der Unterschrift gehindert. Papier	1	0.1310616	0.13	49.67

(continued)

richter	N	exactpercent	roundedpercent	cumulpercent
Papier Die Haas Hömig Steiner Hohmann- Dennhardt Hoffmann- Riem Bryde Richter Jaeger ist aus dem Amt ausgeschieden und deshalb an der Unterschrift gehindert. Papier	1	0.1310616	0.13	49.80
Papier Graßhof Grimm Kühling Frah BVRin Seibert ist aus dem Amt ausgeschieden. Pa- pier Jaeger Haas Hömig	1	0.1310616	0.13	49.93
Papier Grimm Kühling Jaeger Haas Hömig Steiner Hohmann- Dennhardt	10	1.3106160	1.31	51.25
Papier Grimm Kühling Jaeger Haas Hömig Steiner Hohmann- Dennhardt	10	1.3106160	1.31	52.56
Papier Grimm Kühling Jaeger Hömig Steiner Hohmann- Dennhardt	1	0.1310616	0.13	52.69
Papier Grimm Kühling Seibert Jaeger Haas Hömig Steiner	6	0.7536960	0.79	53.47
Papier Haas Der Richter Hömig ist aus dem Steiner Hohmann- Dennhardt Hoffmann- Riem Bryde Gaier Amt ausgeschieden und deshalb an der Unterschrift gehindert Papier	1	0.1310616	0.13	53.60
Papier Haas Hömig Steiner Hohmann- Dennhardt Hoffmann- Riem Bryde Gaier	21	2.8833552	2.88	56.49
Papier Haas Steiner Hohmann- Dennhardt Hoffmann- Riem Bryde Gaier Eichberger	3	0.3931848	0.39	56.88
Papier Hassemer Sommer Jaeger Haas Hömig Steiner Jentsch Broß Oslerloh Hohmann- Dennhardt Hoffmann- Riem Di Fabio Bryde Mellinghoff Lübbe- Wolff	11	1.3906160	1.39	57.01
Papier Hohmann- Dennhardt Bryde Gaier Eichberger Schluckebier Kirchhof Masing	21	2.7522936	2.75	59.76

(continued)

richter	N	exactpercent	roundedpercent	cumulpercent
Papier Hohmann-Dennhardt Der Richter Hoffmann-Riem ist Bryde Gaier Eichberger Schluckebier Kirchhof aus dem Amt ausgeschieden und daher an der Unterschrift gehindert. Papier	1	0.1310616	0.13	59.90
Papier Hohmann-Dennhardt Hoffmann-Riem Bryde Gaier Eichberger Schluckebier	1	0.1310616	0.13	60.03
Papier Hohmann-Dennhardt Hoffmann-Riem Bryde Gaier Eichberger Schluckebier Kirchhof	7	0.9174312	0.92	60.94
Papier Jaeger Haas Hömig Steiner Hohmann-Dennhardt Hoffmann-Riem	1	0.1310616	0.13	61.07
Papier Jaeger Haas Hömig Steiner Hohmann-Dennhardt Hoffmann-Riem Bryde	16	0.73394495	7.34	68.41
Papier Jaeger Hömig Steiner Hohmann-Dennhardt Hoffmann-Riem	1	0.1310616	0.13	68.55
Papier Jaeger Hömig Steiner Hohmann-Riem Bryde	1	0.1310616	0.13	68.68
Papier Kühling Jaeger Haas Hömig Steiner Hohmann-Dennhardt Hoffmann-Riem	7	0.9174312	0.92	69.59
Papier Kühling Jaeger Hömig Steiner Hohmann-Dennhardt	1	0.1310616	0.13	69.72
Papier Kühling Jaeger Hömig Steiner Hohmann-Dennhardt Hoffmann-Riem	1	0.1310616	0.13	69.86
Papier Richter Kühling ist aus dem Jaeger Haas Hömig Steiner Hohmann-Dennhardt Hoffmann-Riem Amt geschieden und daher gehindert zu unterschreiben. Papier	1	0.1310616	0.13	69.99

(continued)

richter	N	exactpercent	roundedpercent	cumulpercent
Papier Richter Kühling ist aus dem Jaeger Haas Hömig Steiner Hohmann-Dennhardt Hoffmann-Riem Amt geschieden und daher gehindert, die Unterschrift zu leisten. Papier	1	0.1310616	0.13	70.12
Papier Steiner Hohmann-Dennhardt Hoffmann-Riem Bryde Gaier Eichberger Schluckebier	13	1.7038008	1.70	71.82
Präsidentin Limbach ist aus Sommer Jentsch Richter Broß ist an der Hassemer Osterloh Di Fabio Mellinghoff dem Amt ausgeschieden. Unterschrift gehindert. Hassemer	1	0.1310616	0.13	71.95
Seidl Grimm Kühling Jaeger Haas Hömig Steiner	1	0.1310616	0.13	72.08
Seidl Grimm Kühling Seibert Jaeger Haas Hömig Steiner	5	0.6553080	0.66	72.74
Vizepräsident Seidl Grimm Kühling Seibert Jaeger Haas Hömig Steiner ist aus dem Amt ausgeschieden und deshalb an der Unterschrift gehindert. Grimm	2	0.2621232	0.26	73.00
Voßkuhle Broß Osterloh Di Fabio Mellinghoff Lübbe-Wolff Gerhardt Landau	35	4.5871560	4.59	77.59
Voßkuhle Di Fabio Der Richter Mellinghoff Lübbe-Wolff Gerhardt Landau Huber Hermanns ist aus dem Amt aus- geschieden und deshalb an der Unterschrift gehindert. Voßkuhle	1	0.1310616	0.13	77.72
Voßkuhle Di Fabio Lübbe-Wolff Gerhardt Landau Huber Hermanns	3	0.3931848	0.39	78.11
Voßkuhle Di Fabio Mellinghoff Lübbe-Wolff Gerhardt Landau	1	0.1310616	0.13	78.24

(continued)

richter	N	exactpercent	roundedpercent	cumulpercent
Voßkuhle Di Fabio Mellinghoff Lübbe-Wolff Gerhardt Landau Huber Hermanns	12	1.5727392	1.57	79.82
Voßkuhle Die Richterinnen Lübbe-Wolff ist aus dem Amt Der Richter Gerhard ist aus dem Amt Landau Huber Hermanns Müller Kessal-Wulf ausgeschieden und deshalb an der Unterschrift gehindert Voßkuhle	1	0.1310616	0.13	79.95
Voßkuhle Gerhardt Landau Huber Hermanns Müller Kessal-Wulf König	1	0.1310616	0.13	80.08
Voßkuhle Huber Hermanns Kessal-Wulf König Maidowski	1	0.1310616	0.13	80.21
Voßkuhle Huber Hermanns Kessal-Wulf König Maidowski Langenfeld	2	0.2621232	0.26	80.47
Voßkuhle Huber Hermanns Müller Kessal-Wulf König	1	0.1310616	0.13	80.60
Voßkuhle Huber Hermanns Müller Kessal-Wulf König Maidowski	1	0.1310616	1.31	81.91
Voßkuhle Huber Hermanns Müller Kessal-Wulf König Maidowski Langenfeld	1	7.7326343	7.73	89.65
Voßkuhle Huber Müller	1	0.1310616	0.13	89.78
Voßkuhle Kirchhof Lübbe-Wolff Gerhardt Gaier Eichberger Schluckebier Masing Paulus Huber Hermanns Baer Britz Müller Kessal-Wulf	1	0.1310616	0.13	89.91
Voßkuhle Landau Hermanns Müller Kessal-Wulf König Maidowski	1	0.1310616	0.13	90.04
Voßkuhle Landau Huber Hermanns Kessal-Wulf König Maidowski	2	0.2621232	0.26	90.30
Voßkuhle Landau Huber Hermanns Müller Kessal-Wulf	2	0.2621232	0.26	90.56
Voßkuhle Landau Huber Hermanns Müller Kessal-Wulf König	4	0.5242464	0.52	91.09
Voßkuhle Landau Huber Hermanns Müller Kessal-Wulf König Maidowski	3	0.3931848	3.15	94.23

(continued)

richter	N	exactpercent	roundedpercent	cumulpercent
Voßkuhle Landau Huber Müller Kessal-Wulf Maidowski		0.1310616	0.13	94.36
Voßkuhle Lübbe-Wolff Gerhardt Huber Hermanns Müller Kessal-Wulf	1	0.1310616	0.13	94.50
Voßkuhle Lübbe-Wolff Gerhardt Landau Huber Hermanns	4	0.5242464	0.52	95.02
Voßkuhle Lübbe-Wolff Gerhardt Landau Huber Hermanns Kessal-Wulf	1	0.1310616	0.13	95.15
Voßkuhle Lübbe-Wolff Gerhardt Landau Huber Hermanns Müller Kessal-Wulf	33	4.3250328	4.33	99.48
Voßkuhle Lübbe-Wolff Gerhardt Landau Huber Hermanns Müller Kessal-Wulff	2	0.2621232	0.26	99.74
Voßkuhle Masing Huber Hermanns Kessal-Wulf König Maidowski Langenfeld	0.1310616	0.13	99.87	
Zeidler Rinck Wand Dr. Rottmann Dr. Dr. h. c. Niebler Steinberger Träger Mahrenholz H.-J. Rinck	1	0.1310616	0.13	100.00
Total	763	100.0000000	100.00	100.00

Frequency Table for Variable: entscheidung_typ

2 unique value(s) detected.

entscheidung_typ	N	exactpercent	roundedpercent	cumulpercent
B	580	76.01573	76.02	76.02
U	183	23.98427	23.98	100.00
Total	763	100.00000	100.00	100.00

10 Frequenztabellen visualisieren

10.1 Präfix erstellen

```
prefix <- file.path(dir.analysis,  
                    paste0(config$project$shortname,  
                          "_01_Frequenztafel_var-"))
```

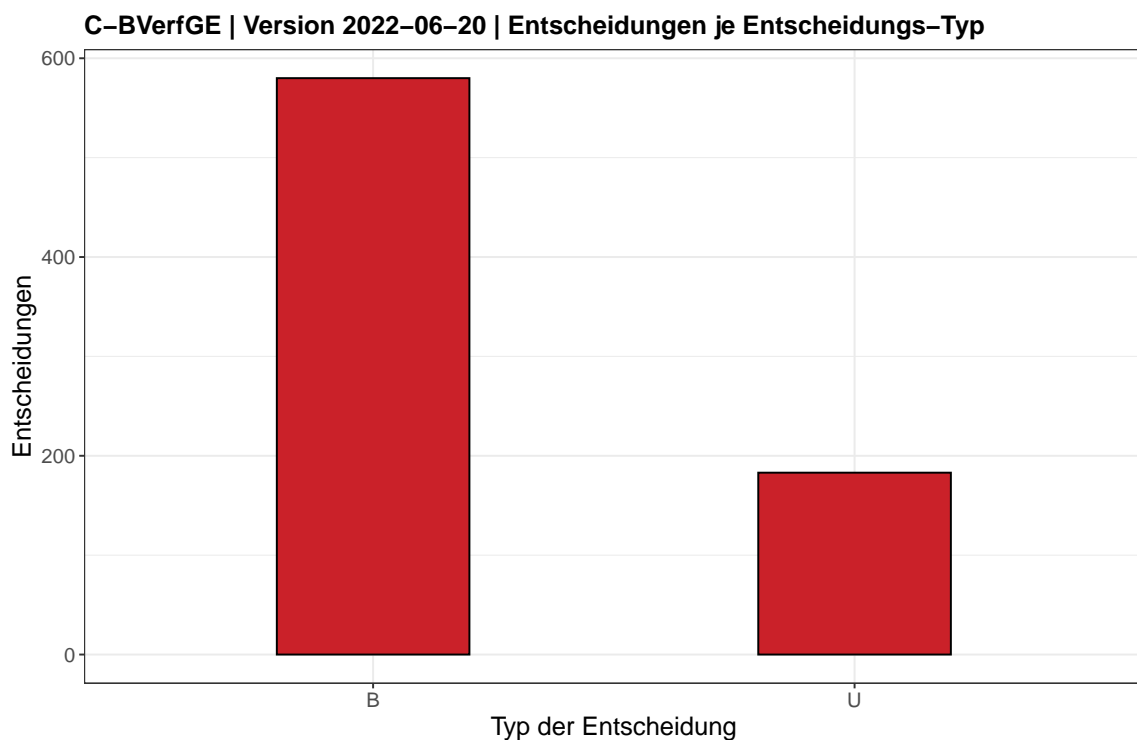
10.2 Tabellen einlesen

```
table.entsch.typ <- fread(paste0(prefix,  
                                "entscheidung_typ.csv"))  
  
table.spruch.typ <- fread(paste0(prefix,  
                                "spruchkoerper_typ.csv"))  
  
table.spruch.az <- fread(paste0(prefix,  
                                "spruchkoerper_az.csv"))  
  
table.regz <- fread(paste0(prefix,  
                           "registerzeichen.csv"))  
  
table.jahr.eingangISO <- fread(paste0(prefix,  
                                    "eingangsjahr_iso.csv"))  
  
table.jahr.entscheid <- fread(paste0(prefix,  
                                    "entscheidungsjahr.csv"))  
  
table.output.praesi <- fread(paste0(prefix,  
                                   "praesi.csv"))  
  
table.output.vpraesi <- fread(paste0(prefix,  
                                   "v_praesi.csv"))
```

10.3 Diagramm: Typ der Entscheidung

```
freqtable <- table.entsch.typ[-.N]
```

```
ggplot(data = freqtable) +  
  geom_bar(aes(x = reorder(entscheidung_typ,  
                           -N),  
              y = N),  
          stat = "identity",  
          fill = "#ca2129",  
          color = "black",  
          width = 0.4) +  
  theme_bw() +  
  labs(  
    title = paste(prefix.figuretitle,  
                  "| Entscheidungen je Entscheidungs-Typ"),  
    caption = caption,  
    x = "Typ der Entscheidung",  
    y = "Entscheidungen"  
  ) +  
  theme(  
    text = element_text(size = 14),  
    plot.title = element_text(size = 14,  
                              face = "bold"),  
    legend.position = "none",  
    plot.margin = margin(10, 20, 10, 10)  
  )
```

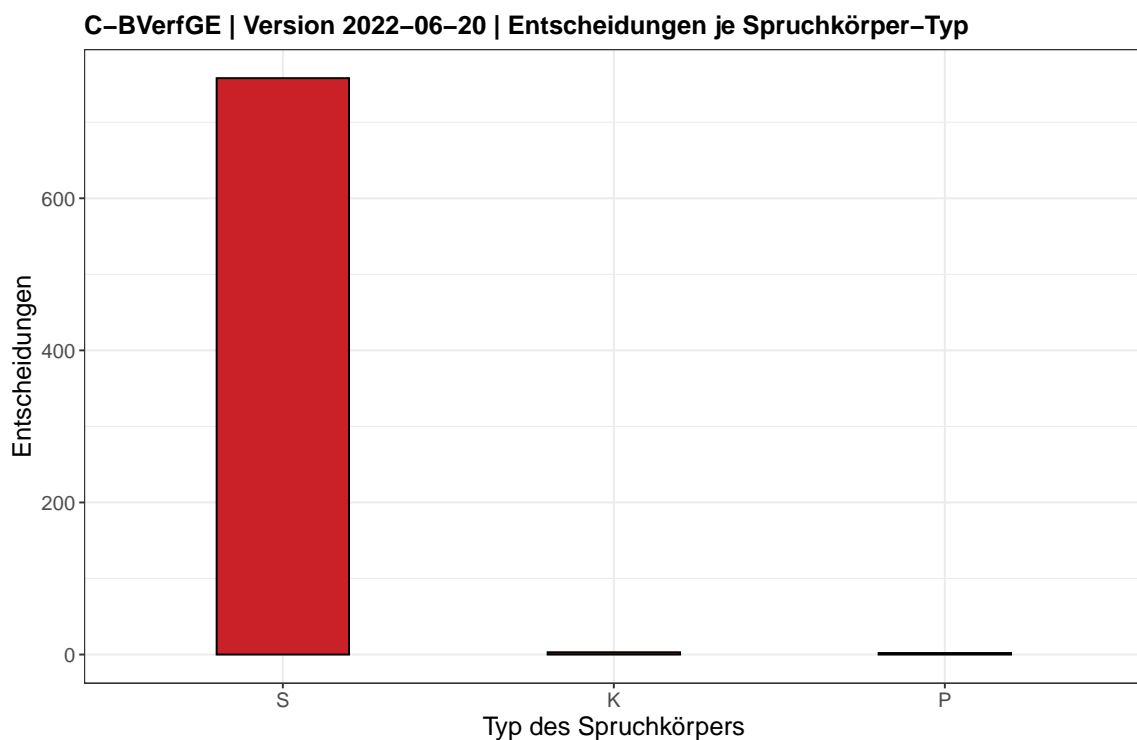


Fobbe | DOI: 10.5281/zenodo.6569896

10.4 Diagramm: Typ des Spruchkörpers

```
freqtable <- table.spruch.typ[-.N]
```

```
ggplot(data = freqtable) +  
  geom_bar(aes(x = reorder(spruchkoerper_typ,  
                           -N),  
               y = N),  
           stat = "identity",  
           fill = "#ca2129",  
           color = "black",  
           width = 0.4) +  
  theme_bw() +  
  labs(  
    title = paste(prefix.figuretitle,  
                  "| Entscheidungen je Spruchkörper-Typ"),  
    caption = caption,  
    x = "Typ des Spruchkörpers",  
    y = "Entscheidungen"  
  ) +  
  theme(  
    text = element_text(size = 14),  
    plot.title = element_text(size = 14,  
                               face = "bold"),  
    legend.position = "none",  
    plot.margin = margin(10, 20, 10, 10)  
  )
```

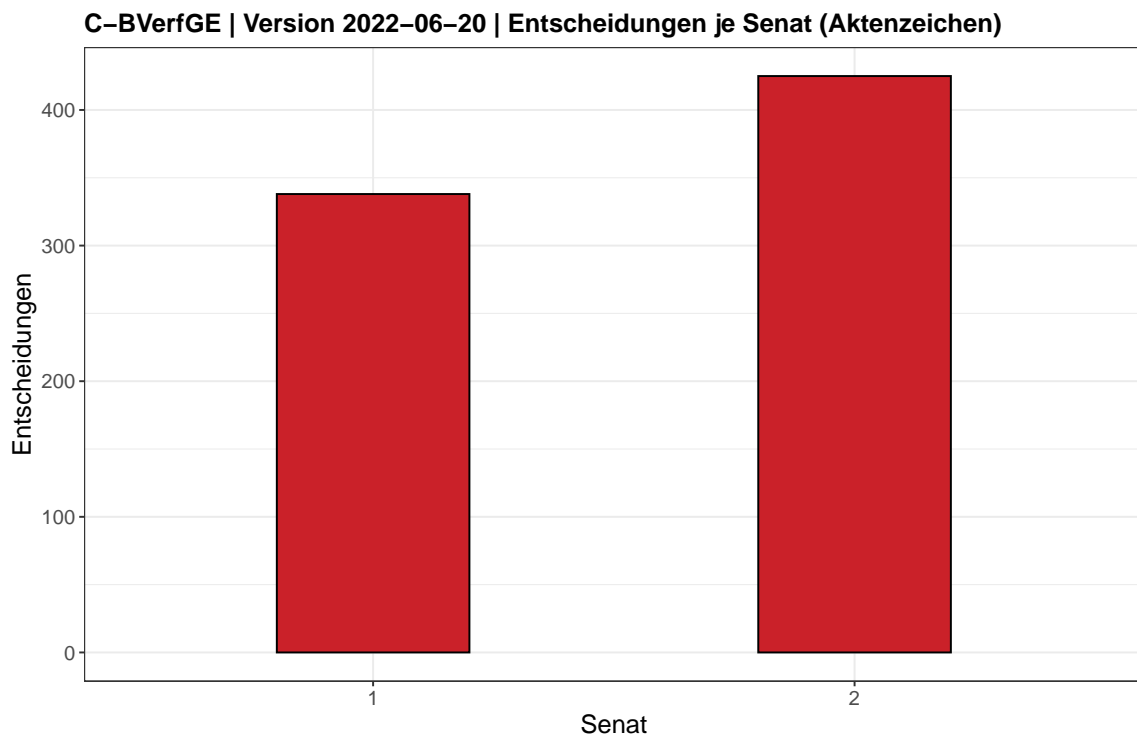


Fobbe | DOI: 10.5281/zenodo.6569896

10.5 Diagramm: Spruchkörper nach Aktenzeichen

```
freqtable <- table.spruch.az[-.N]
```

```
ggplot(data = freqtable) +  
  geom_bar(aes(x = spruchkoerper_az,  
               y = N),  
           stat = "identity",  
           fill = "#ca2129",  
           color = "black",  
           width = 0.4) +  
  theme_bw() +  
  labs(  
    title = paste(prefix.figuretitle,  
                  "| Entscheidungen je Senat (Aktenzeichen)",  
    caption = caption,  
    x = "Senat",  
    y = "Entscheidungen"  
  ) +  
  theme(  
    text = element_text(size = 14),  
    plot.title = element_text(size = 14,  
                              face = "bold"),  
    legend.position = "none",  
    plot.margin = margin(10, 20, 10, 10)  
  )
```



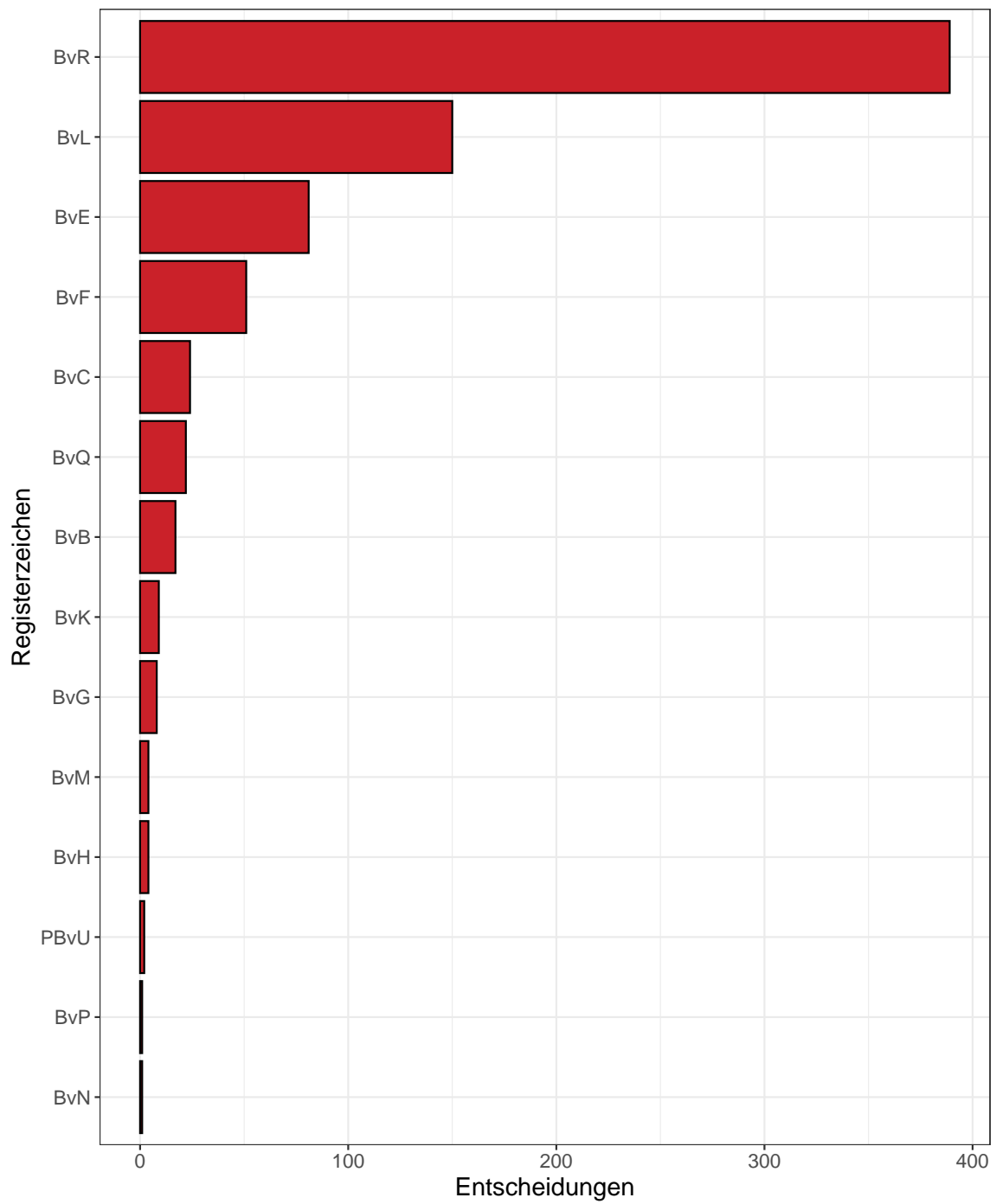
Fobbe | DOI: 10.5281/zenodo.6569896

10.6 Diagramm: Registerzeichen

```
freqtable <- table.regz[-.N]
```

```
ggplot(data = freqtable) +  
  geom_bar(aes(x = reorder(registerzeichen,  
                           N),  
              y = N),  
          stat = "identity",  
          fill = "#ca2129",  
          color = "black") +  
  coord_flip() +  
  theme_bw() +  
  labs(  
    title = paste(prefix.figuretitle,  
                  "| Entscheidungen je Registerzeichen"),  
    caption = caption,  
    x = "Registerzeichen",  
    y = "Entscheidungen"  
  ) +  
  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-BVerfGE | Version 2022-06-20 | Entscheidungen je Registerzeichen

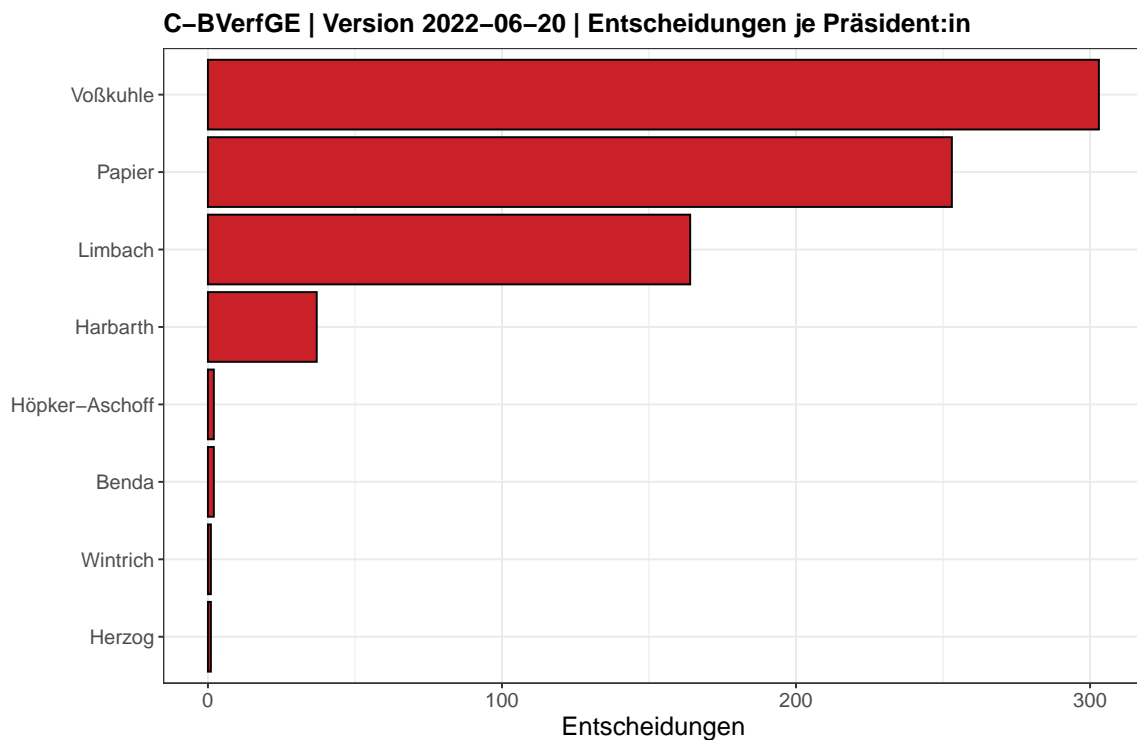


Fobbe | DOI: 10.5281/zenodo.6569896

10.7 Diagramm: Präsident:in

```
freqtable <- table.output.praesi[-.N]
```

```
ggplot(data = freqtable) +  
  geom_bar(aes(x = reorder(praesi,  
                           N),  
               y = N),  
           stat = "identity",  
           fill = "#ca2129",  
           color = "black") +  
  coord_flip() +  
  theme_bw() +  
  labs(  
    title = paste(prefix.figuretitle,  
                  "| Entscheidungen je Präsident:in"),  
    caption = caption,  
    x = "Präsident:in",  
    y = "Entscheidungen"  
  ) +  
  theme(  
    axis.title.y = element_blank(),  
    text = element_text(size = 14),  
    plot.title = element_text(size = 14,  
                              face = "bold"),  
    legend.position = "none",  
    plot.margin = margin(10, 20, 10, 10)  
  )
```

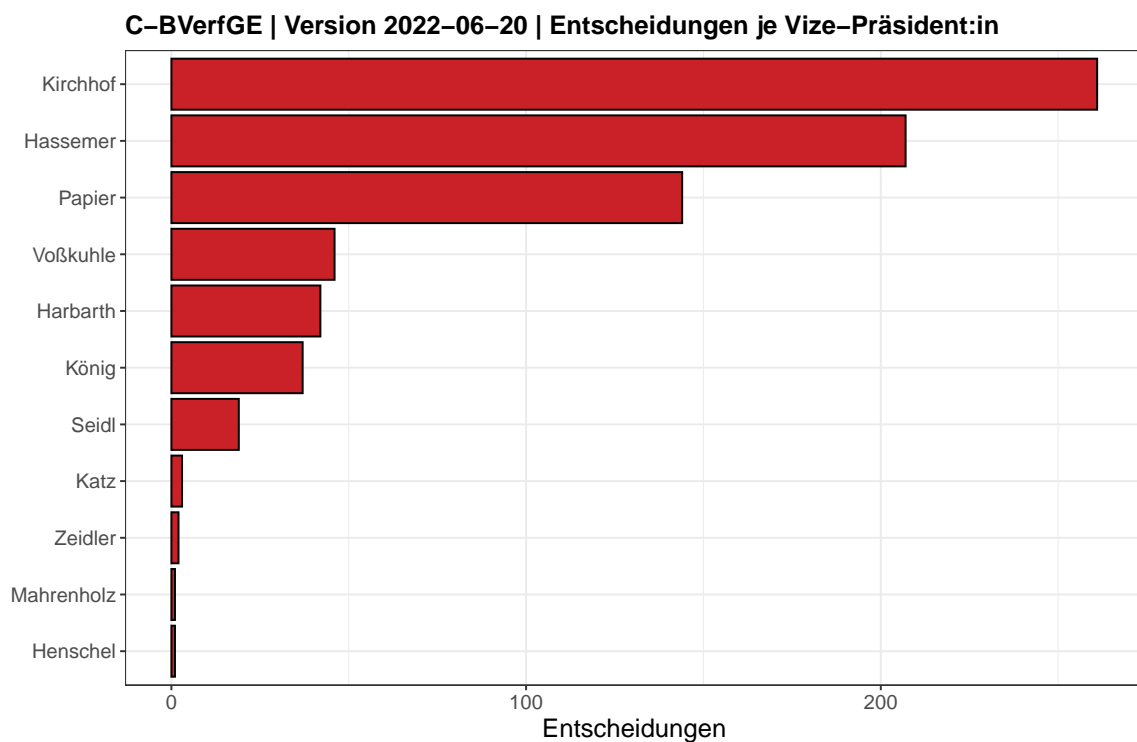


Fobbe | DOI: 10.5281/zenodo.6569896

10.8 Diagramm: Vize-Präsident:in

```
freqtable <- table.output.vpraesi[-.N]
```

```
ggplot(data = freqtable) +  
  geom_bar(aes(x = reorder(v_praesi,  
                           N),  
               y = N),  
           stat = "identity",  
           fill = "#ca2129",  
           color = "black") +  
  coord_flip() +  
  theme_bw() +  
  labs(  
    title = paste(prefix.figuretitle,  
                  "| Entscheidungen je Vize-Präsident:in"),  
    caption = caption,  
    x = "Vize-Präsident:in",  
    y = "Entscheidungen"  
  ) +  
  theme(  
    axis.title.y = element_blank(),  
    text = element_text(size = 14),  
    plot.title = element_text(size = 14,  
                              face = "bold"),  
    legend.position = "none",  
    plot.margin = margin(10, 20, 10, 10)  
  )
```

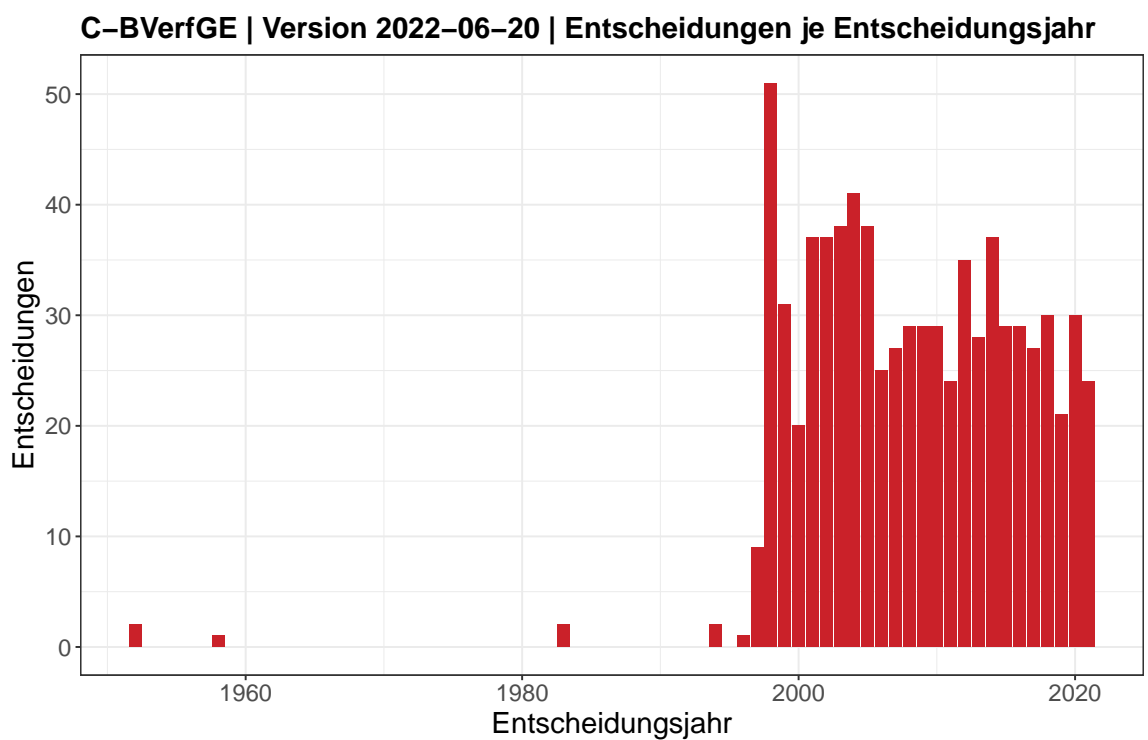


Fobbe | DOI: 10.5281/zenodo.6569896

10.9 Diagramm: Entscheidungsjahr

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

```
ggplot(data = freqtable) +  
  geom_bar(aes(x = entscheidungsjahr,  
               y = N),  
           stat = "identity",  
           fill = "#ca2129") +  
  theme_bw() +  
  labs(  
    title = paste(prefix.figuretitle,  
                  "| Entscheidungen je Entscheidungsjahr"),  
    caption = caption,  
    x = "Entscheidungsjahr",  
    y = "Entscheidungen"  
  ) +  
  theme(  
    text = element_text(size = 16),  
    plot.title = element_text(size = 16,  
                               face = "bold"),  
    legend.position = "none",  
    plot.margin = margin(10, 20, 10, 10)  
  )
```

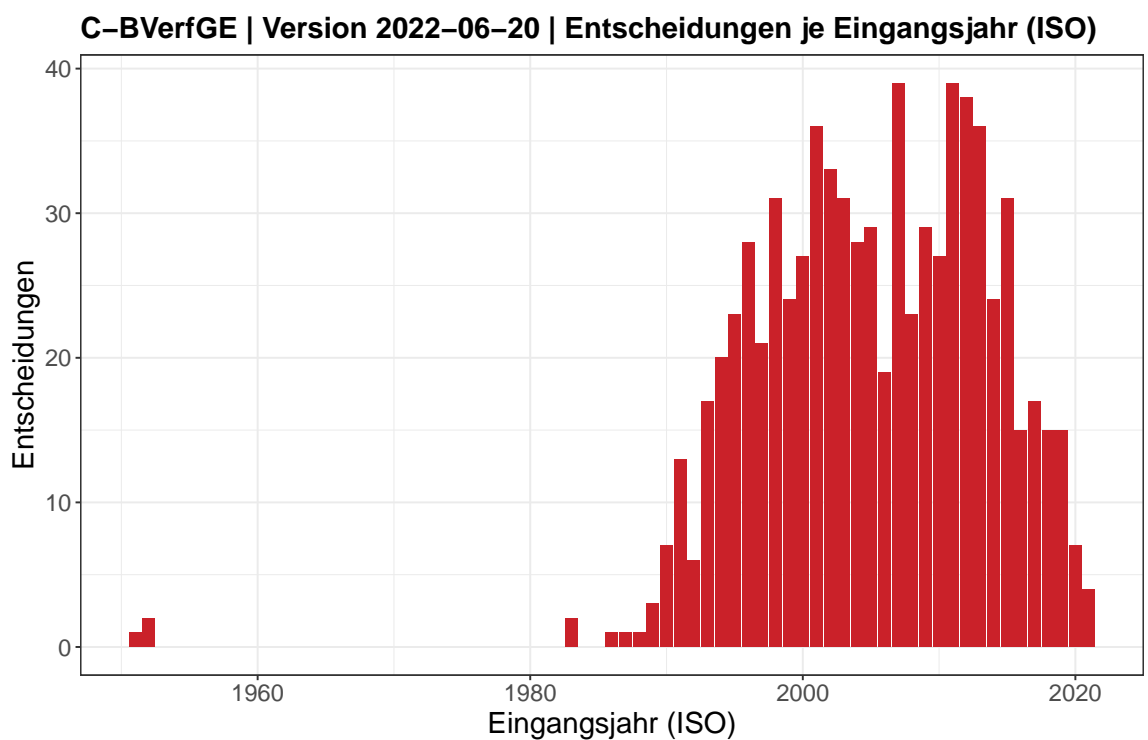


Fobbe | DOI: 10.5281/zenodo.6569896

10.10 Diagramm: Eingangsjahr (ISO)

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

```
ggplot(data = freqtable) +  
  geom_bar(aes(x = eingangsjahr_iso,  
               y = N),  
           stat = "identity",  
           fill = "#ca2129") +  
  theme_bw() +  
  labs(  
    title = paste(prefix.figuretitle,  
                  "| Entscheidungen je Eingangsjahr (ISO)"),  
    caption = caption,  
    x = "Eingangsjahr (ISO)",  
    y = "Entscheidungen"  
  ) +  
  theme(  
    text = element_text(size = 16),  
    plot.title = element_text(size = 16,  
                               face = "bold"),  
    legend.position = "none",  
    plot.margin = margin(10, 20, 10, 10)  
  )
```



Fobbe | DOI: 10.5281/zenodo.6569896

11 Korpus-Analytik

11.1 Berechnung linguistischer Kennwerte

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

11.1.1 Funktion anzeigen

```
print(f.future_lingsummarize)
```

```
## function(dt,
##                                     chunkspersworker = 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,
##                                f.lingsummarize,
##                                future.seed = TRUE,
##                                future.scheduling = chunkspersworker,
##                                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)
##
## }

```

11.1.2 Berechnung durchführen

```

if(config$parallel$lingssummarize == TRUE){

  plan("multicore",
        workers = fullCores)

}else{

  plan("sequential")

}

```

```

summary.corpus <- f.future_lingsummarize(txt.bverfg)

#deprecated; Parallelisierung jetzt mit futures
#summary.corpus <- f.lingsummarize.iterator(txt.bverfg,
#                                           threads = fullCores,
#                                           chunksize = 1)

```

11.2 Variablen-Namen anpassen

```
setnames(summary.corpus,  
  old = c("nchars",  
          "ntokens",  
          "ntypes",  
          "nsentences"),  
  new = c("zeichen",  
          "tokens",  
          "typen",  
          "saetze"))
```

11.3 Kennwerte dem Korpus hinzufügen

```
txt.bverfg <- cbind(txt.bverfg,  
  summary.corpus)
```

11.4 Variante mit Metadaten erstellen

```
meta.bverfg <- txt.bverfg[, !"text"]
```

11.5 Linguistische Kennwerte

11.5.1 Zusammenfassungen berechnen

```
dt.summary.ling <- meta.bverfg[, lapply(.SD,  
  function(x)unclass(summary(x))),  
  .SDcols = c("zeichen",  
              "tokens",  
              "typen",  
              "saetze")]  
  
dt.sums.ling <- meta.bverfg[,  
  lapply(.SD, sum),  
  .SDcols = c("zeichen",  
              "tokens",  
              "typen",  
              "saetze")]  
  
tokens.temp <- tokens(corpus(txt.bverfg),  
  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.temp))

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

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

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

```

11.5.2 Zusammenfassungen anzeigen

```

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

```

Variable	Sum	Min	Quart1	Median	Mean	Quart3	Max
zeichen	56,663,202	2,339	30,395.0	60,026	74,263.6986	96,662.0	781,226
tokens	8,747,474	266	4,667.0	9,156	11,464.5793	14,762.5	115,540
typen	160,446	126	1,221.5	1,981	2,120.1245	2,723.5	13,491
saetze	470,438	10	261.5	497	616.5636	789.5	5,282

11.5.3 Zusammenfassungen speichern

```
fwrite(dt.stats.ling,  
       file.path(dir.analysis,  
                 paste0(config$project$shortname,  
                        "_00_KorpusStatistik_ZusammenfassungLinguistisch.csv")),  
       na = "NA")
```


11.6 Quantitative Variablen

11.6.1 Entscheidungsdatum

```
summary(as.IDate(meta.bverfg$datum))
```

```
##           Min.         1st Qu.         Median         Mean         3rd Qu.         Max.
## "1952-09-10" "2002-12-11" "2008-07-03" "2008-10-05" "2014-12-16" "2021-07-20"
```

11.6.2 Zusammenfassungen berechnen

```
dt.summary.docvars <- meta.bverfg[,
                                lapply(.SD, function(x)unclass(summary(na.omit(
                                  x))))),
                                .SDcols = c("entscheidungsjahr",
                                              "eingangsjahr_iso",
                                              "band",
                                              "eingangsnummer")]

dt.unique.docvars <- meta.bverfg[,
                                lapply(.SD, function(x)length(unique(na.omit(x))
                                  )),
                                .SDcols = c("entscheidungsjahr",
                                              "eingangsjahr_iso",
                                              "band",
                                              "eingangsnummer")]

dt.stats.docvars <- rbind(dt.unique.docvars,
                          dt.summary.docvars)

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

setnames(dt.stats.docvars, c("Variable",
                              "Anzahl",
                              "Min",
                              "Quart1",
                              "Median",
                              "Mean",
                              "Quart3",
                              "Max"))
```

11.6.3 Zusammenfassungen anzeigen

```
kable(dt.stats.docvars,  
      format = "latex",  
      booktabs = TRUE,  
      longtable = TRUE)
```

Variable	Anzahl	Min	Quart1	Median	Mean	Quart3	Max
entscheidungsjahr	30	1952	2002	2008	2008.2831	2014.0	2021
eingangsjahr_iso	39	1951	1999	2005	2005.1311	2012.0	2021
band	70	1	106	121	122.6107	138.0	158
eingangsnummer	381	1	4	97	705.5688	1350.5	3588

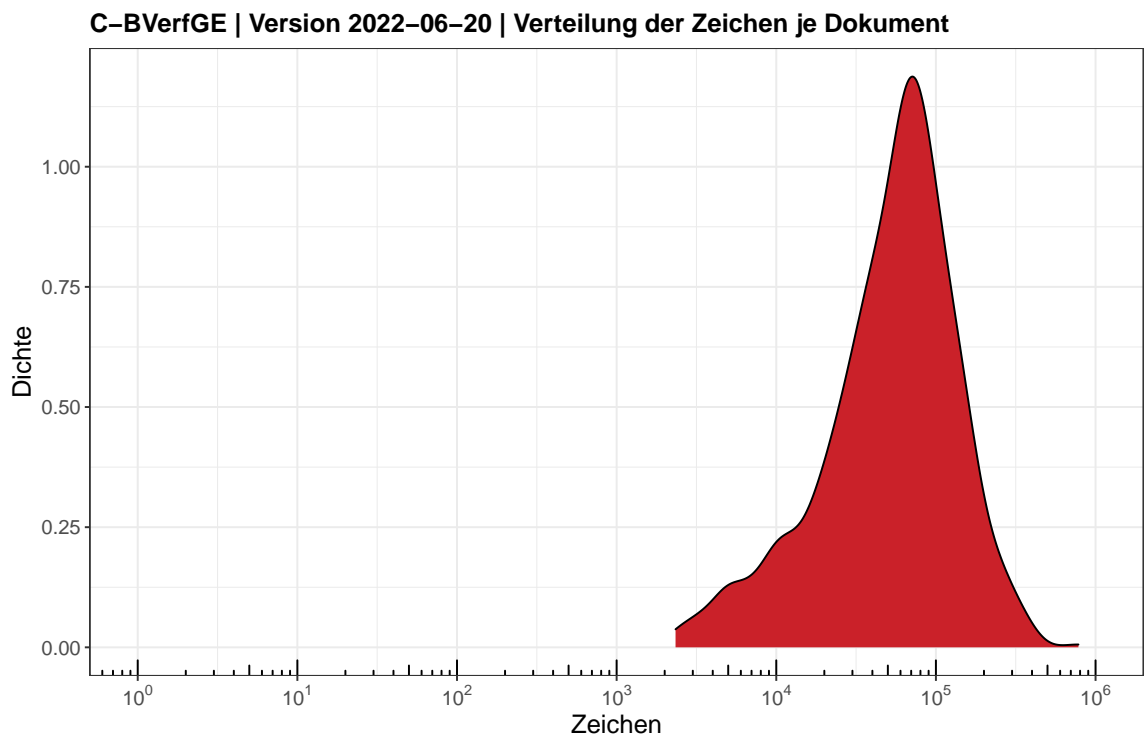
11.6.4 Zusammenfassungen speichern

```
fwrite(dt.stats.docvars,  
       file.path(dir.analysis,  
                 paste0(config$project$shortname,  
                        "_00_KorpusStatistik_ZusammenfassungDocvarsQuantitativ.  
csv")),  
       na = "NA")
```

11.7 Verteilungen linguistischer Kennwerte

11.7.1 Diagramm: Verteilung Zeichen

```
ggplot(data = meta.bverfg)+
  geom_density(aes(x = zeichen),
    fill = "#ca2129")+
  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 Dokument"),
    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)
  )
```



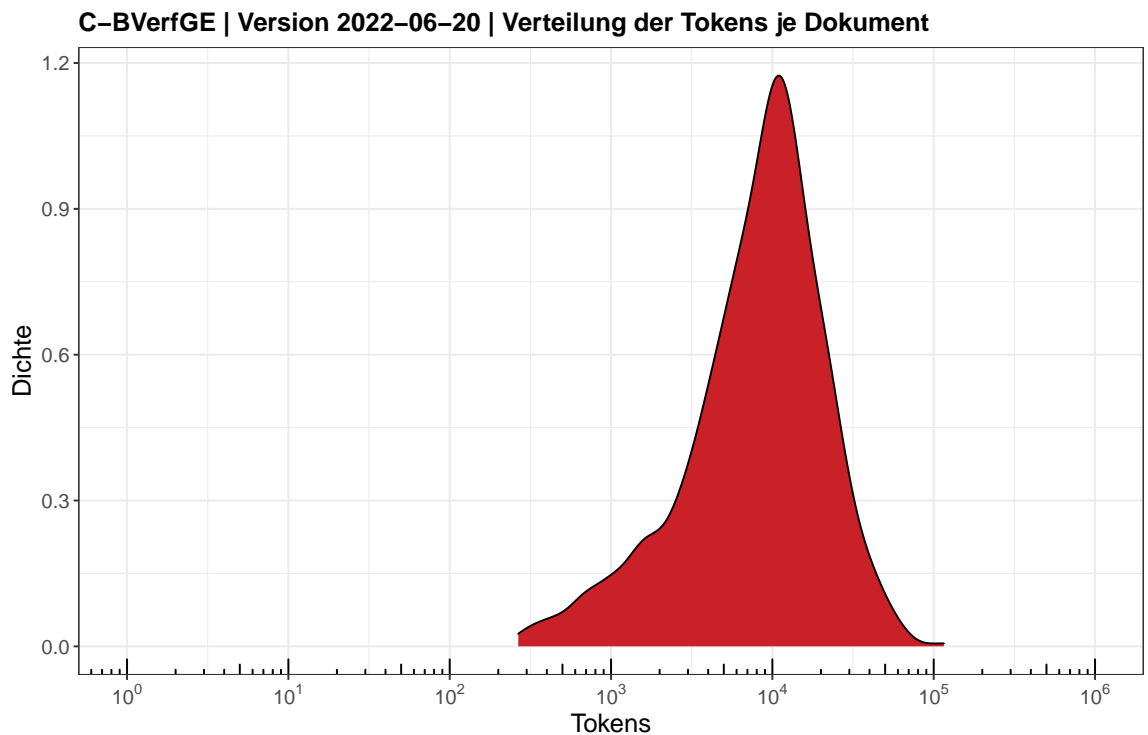
Fobbe | DOI: 10.5281/zenodo.6569896

11.7.2 Diagramm: Verteilung Tokens

```

ggplot(data = meta.bverfg)+
  geom_density(aes(x = tokens),
    fill = "#ca2129")+
  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 Dokument"),
    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)
  )

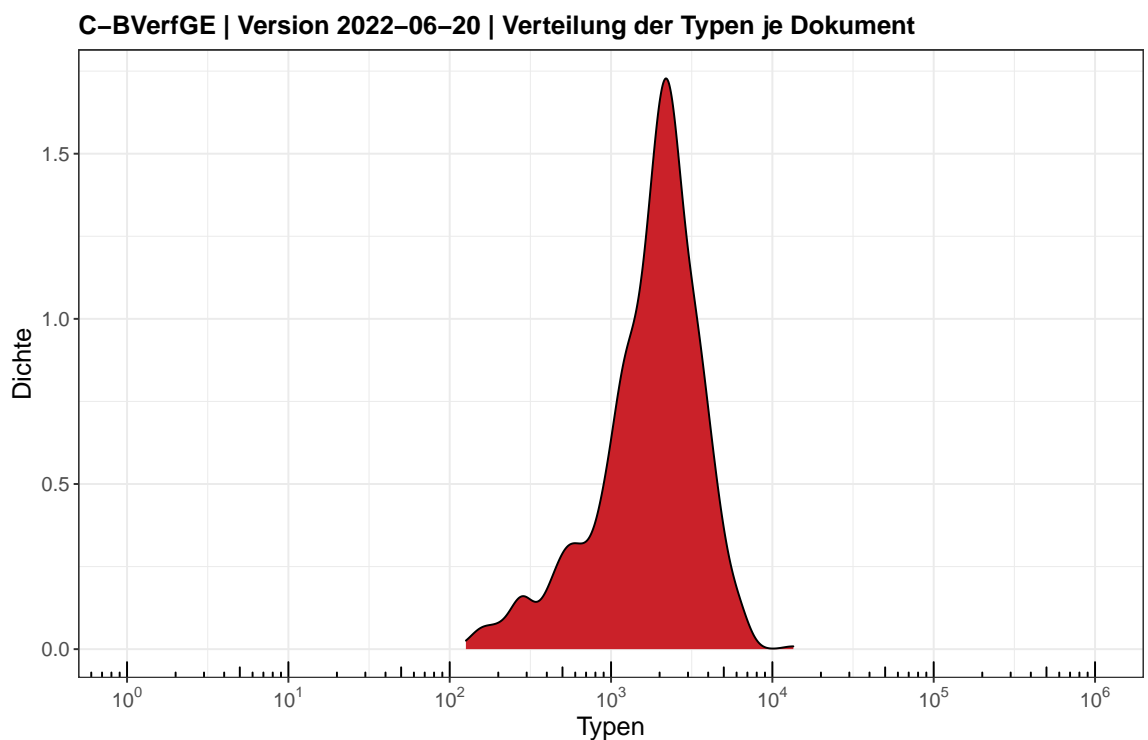
```



Fobbe | DOI: 10.5281/zenodo.6569896

11.7.3 Diagramm: Verteilung Typen

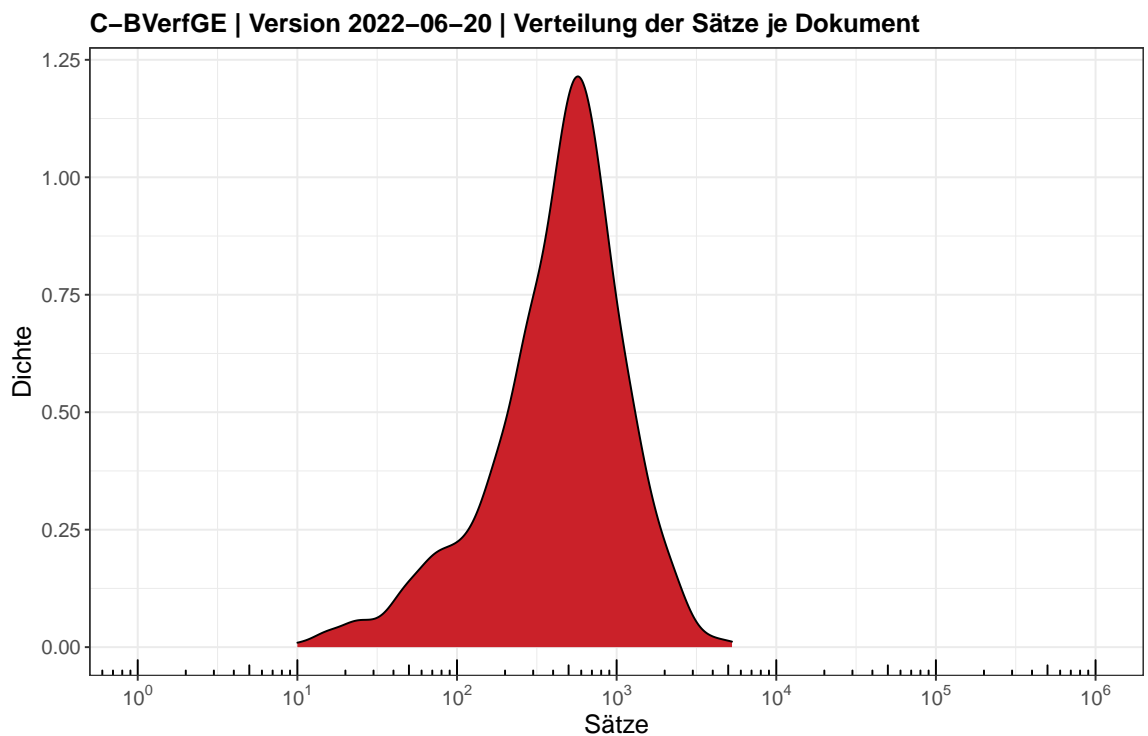
```
ggplot(data = meta.bverfg)+
  geom_density(aes(x = typen),
    fill = "#ca2129")+
  scale_x_log10(breaks = trans_breaks("log10", function(x) 10^x),
    labels = trans_format("log10", math_format(10^.x)))+
  annotation_logticks(sides = "b")+
  coord_cartesian(xlim = c(1, 10^6))+
  theme_bw()+
  labs(
    title = paste(prefix.figuretitle,
      "| Verteilung der Typen je Dokument"),
    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.6569896

11.7.4 Diagramm: Verteilung Sätze

```
ggplot(data = meta.bverfg)+
  geom_density(aes(x = saetze),
    fill = "#ca2129")+
  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 Dokument"),
    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.6569896

12 Linguistische Annotationen berechnen

12.1 Funktion anzeigen: f.future_spacyparse

```
print(f.future_spacyparse)
```

```
## function(x,
##
##             chunkspersworker = 1,
##             chunksize = NULL,
##             model = "en_core_web_sm",
##             pos = TRUE,
##             tag = FALSE,
##             lemma = FALSE,
##             entity = FALSE,
##             dependency = FALSE,
##             nounphrase = FALSE){
##
##   begin <- Sys.time()
##
##   spacy_initialize(model = model)
##
##   print(paste0("Begin at ",
##               begin,
##               ". Processing ",
##               x[,.N],
##               " documents"))
##
##
##   raw.list <- split(x, seq(nrow(x)))
##
##   result.list <- future_lapply(raw.list,
##                               spacy_parse,
##                               future.seed = TRUE,
##                               future.scheduling = chunkspersworker,
##                               future.chunk.size = chunksize,
##                               pos = pos,
##                               tag = tag,
##                               lemma = lemma,
##                               entity = entity,
##                               dependency = dependency,
##                               nounphrase = nounphrase,
##                               multithread = FALSE)
##
##   result.dt <- rbindlist(result.list)
##
##
##   end <- Sys.time()
##   duration <- end - begin
```

```
##
##   print(paste0("Runtime was ",
##               round(duration,
##                   digits = 2),
##               " ",
##               attributes(duration)$units,
##               ". Ended at ",
##               end, "."))
##
##   spacy_finalize()
##
##   return(result.dt)
##
## }
```

12.2 Berechnungen durchführen

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

  if(config$parallel$spacyparse == TRUE){

    plan("multicore",
          workers = fullCores)

  }else{

    plan("sequential")

  }

  txt.annotated <- f.future_spacyparse(txt.bverfg,
                                       chunksperworker = 1,
                                       chunksize = NULL,
                                       model = "de_core_news_sm",
                                       pos = TRUE,
                                       tag = TRUE,
                                       lemma = TRUE,
                                       entity = TRUE,
                                       dependency = TRUE,
                                       nounphrase = TRUE)

}
```


13 Kontrolle der Variablen

13.1 Semantische Sortierung der Variablen

13.1.1 Variablen sortieren: Hauptdatensatz

```
setcolorder(txt.bverfg,  
  c("doc_id",  
    "text",  
    "gericht",  
    "datum",  
    "entscheidung_typ",  
    "spruchkoerper_typ",  
    "spruchkoerper_az",  
    "registerzeichen",  
    "verfahrensart",  
    "eingangsnummer",  
    "eingangsjahr_az",  
    "eingangsjahr_iso",  
    "entscheidungsjahr",  
    "kollision",  
    "name",  
    "band",  
    "seite",  
    "aktenzeichen",  
    "aktenzeichen_alle",  
    "ecli",  
    "zitativvorschlag",  
    "kurzbeschreibung",  
    "pressemitteilung",  
    "praesi",  
    "v_praesi",  
    "richter",  
    "zeichen",  
    "tokens",  
    "typen",  
    "saetze",  
    "version",  
    "doi_concept",  
    "doi_version",  
    "lizenz"))
```

13.1.2 Variablen sortieren: Metadaten

```
setcolorder(meta.bverfg,  
  c("doc_id",  
    "gericht",  
    "datum",  
    "entscheidung_typ",  
    "spruchkoerper_typ",  
    "spruchkoerper_az",  
    "registerzeichen",  
    "verfahrensart",  
    "eingangsnummer",  
    "eingangsjahr_az",  
    "eingangsjahr_iso",  
    "entscheidungsjahr",  
    "kollision",  
    "name",  
    "band",  
    "seite",  
    "aktenzeichen",  
    "aktenzeichen_alle",  
    "ecli",  
    "zitativorschlag",  
    "kurzbeschreibung",  
    "pressemitteilung",  
    "praesi",  
    "v_praesi",  
    "richter",  
    "zeichen",  
    "tokens",  
    "typen",  
    "saetze",  
    "version",  
    "doi_concept",  
    "doi_version",  
    "lizenz"))
```

13.1.3 Variablen sortieren: Segmentiert

```
setcolorder(dt.segmented.full,  
  c("doc_id",  
    "text",  
    "segment",  
    "gericht",  
    "datum",  
    "entscheidung_typ",  
    "spruchkoerper_typ",  
    "spruchkoerper_az",  
    "registerzeichen",  
    "verfahrensart",  
    "eingangsnummer",  
    "eingangsjahr_az",  
    "eingangsjahr_iso",  
    "entscheidungsjahr",  
    "kollision",  
    "name",  
    "band",  
    "seite",  
    "aktenzeichen",  
    "aktenzeichen_alle",  
    "ecli",  
    "zitativorschlag",  
    "kurzbeschreibung",  
    "pressemitteilung",  
    "praesi",  
    "v_praesi",  
    "richter",  
    "version",  
    "doi_concept",  
    "doi_version",  
    "lizenz"))
```

13.2 Anzahl Variablen der Datensätze

```
length(txt.bverfg)
```

```
## [1] 34
```

```
length(meta.bverfg)
```

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

```
length(txt.annotated)
```

```
## [1] 12
```

```
length(dt.segmented.full)
```

```
## [1] 31
```

13.3 Alle Variablen-Namen der Datensätze

```
names(txt.bverfg)
```

```
## [1] "doc_id"      "text"        "gericht"
## [4] "datum"      "entscheidung_typ" "spruchkoerper_typ"
## [7] "spruchkoerper_az" "registerzeichen" "verfahrensart"
## [10] "eingangsnummer" "eingangsjahr_az" "eingangsjahr_iso"
## [13] "entscheidungsjahr" "kollision"      "name"
## [16] "band"        "seite"         "aktenzeichen"
## [19] "aktenzeichen_alle" "ecli"           "zitativorschlag"
## [22] "kurzbeschreibung" "pressemitteilung" "praesi"
## [25] "v_praesi"     "richter"        "zeichen"
## [28] "tokens"       "typen"           "saetze"
## [31] "version"      "doi_concept"     "doi_version"
## [34] "lizenz"
```

```
names(meta.bverfg)
```

```
## [1] "doc_id"      "gericht"      "datum"
## [4] "entscheidung_typ" "spruchkoerper_typ" "spruchkoerper_az"
## [7] "registerzeichen" "verfahrensart"   "eingangsnummer"
## [10] "eingangsjahr_az" "eingangsjahr_iso" "entscheidungsjahr"
## [13] "kollision"      "name"          "band"
## [16] "seite"          "aktenzeichen"  "aktenzeichen_alle"
## [19] "ecli"           "zitiervorschlag" "kurzbeschreibung"
## [22] "pressemitteilung" "praesi"         "v_praesi"
## [25] "richter"        "zeichen"        "tokens"
## [28] "typen"          "saetze"         "version"
## [31] "doi_concept"    "doi_version"    "lizenz"
```

```
names(txt.annotated)
```

```
## [1] "doc_id"      "sentence_id"   "token_id"      "token"
## [5] "lemma"       "pos"           "tag"           "head_token_id"
## [9] "dep_rel"     "entity"        "nounphrase"    "whitespace"
```

```
names(dt.segmented.full)
```

```
## [1] "doc_id"      "text"          "segment"
## [4] "gericht"     "datum"         "entscheidung_typ"
## [7] "spruchkoerper_typ" "spruchkoerper_az" "registerzeichen"
## [10] "verfahrensart" "eingangsnummer" "eingangsjahr_az"
## [13] "eingangsjahr_iso" "entscheidungsjahr" "kollision"
## [16] "name"        "band"          "seite"
## [19] "aktenzeichen" "aktenzeichen_alle" "ecli"
## [22] "zitiervorschlag" "kurzbeschreibung" "pressemitteilung"
## [25] "praesi"       "v_praesi"      "richter"
## [28] "version"     "doi_concept"   "doi_version"
## [31] "lizenz"
```

14 CSV-Dateien erstellen

14.1 CSV mit vollem Datensatz speichern

```
csvname.full <- paste(prefix.files,  
                      "DE_CSV_Datensatz.csv",  
                      sep = "_")  
  
fwrite(txt.bverfg,  
       csvname.full,  
       na = "NA")
```

14.2 CSV mit Metadaten speichern

Diese Datei ist grundsätzlich identisch mit dem eigentlichen Datensatz, nur ohne den Text der Entscheidungen.

```
csvname.meta <- paste(prefix.files,  
                     "DE_CSV_Metadaten.csv",  
                     sep = "_")  
  
fwrite(meta.bverfg,  
       csvname.meta,  
       na = "NA")
```

14.3 CSV mit Segmenten speichern

```
csvname.segmented <- paste(prefix.files,  
                          "DE_CSV_Segmentiert.csv",  
                          sep = "_")  
  
fwrite(dt.segmented.full,  
       csvname.segmented,  
       na = "NA")
```

14.4 CSV mit Annotationen speichern

```
if (config$annotate$toggle == TRUE){  
  csvname.annotated <- paste(prefix.files,  
                            "DE_CSV_Annotiert.csv",  
                            sep = "_")  
  
  fwrite(txt.annotated,  
        csvname.annotated,  
        na = "NA")  
}
```

15 Dateigrößen analysieren

15.1 Gesamtgröße

15.1.1 Korpus-Objekt in RAM (MB)

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

```
## 58.5 MB
```

15.1.2 CSV Korpus (MB)

```
file.size(csvname.full) / 10 ^ 6
```

```
## [1] 58.15826
```

15.1.3 CSV Metadaten (MB)

```
file.size(csvname.meta) / 10 ^ 6
```

```
## [1] 0.607879
```

15.1.4 CSV Annotiert (MB)

```
file.size(csvname.annotated) / 10 ^ 6
```

```
## [1] 1225.549
```

15.1.5 CSV Segmentiert (MB)

```
file.size(csvname.segmented) / 10 ^ 6
```

```
## [1] 112.2885
```

15.1.6 PDF-Dateien (MB)

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

```
## [1] 134.7723
```

15.1.7 TXT-Dateien (MB)

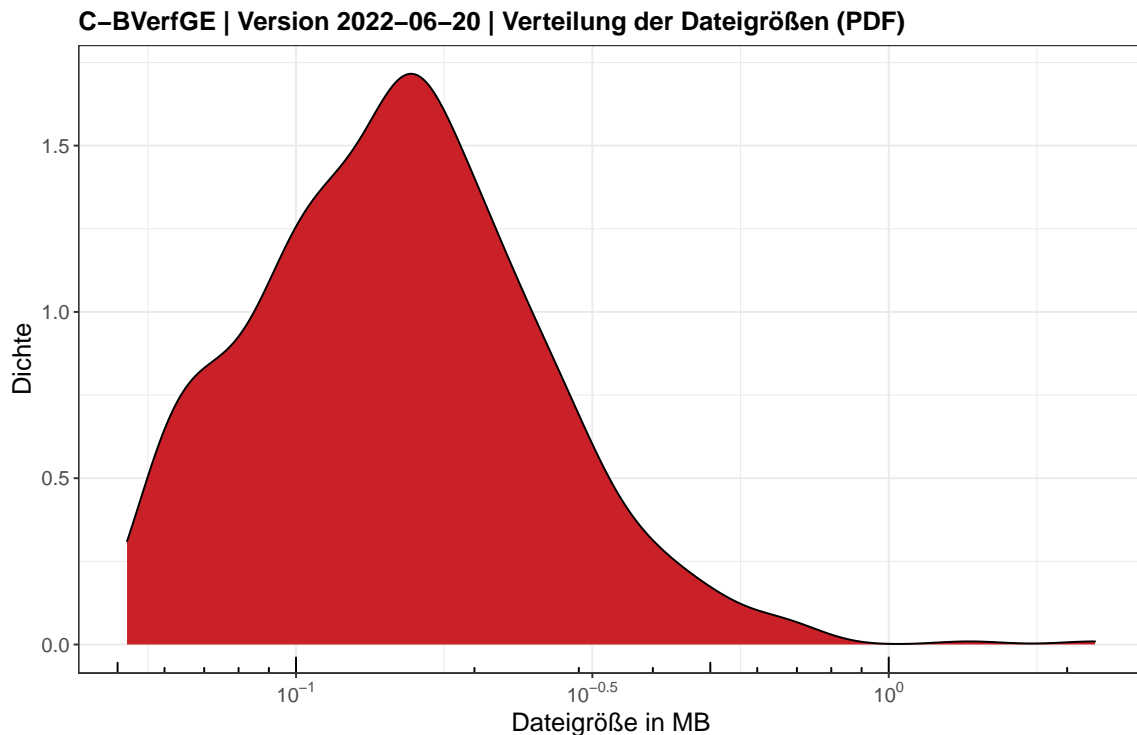
```
files.txt <- list.files(pattern = "\\..txt$",  
                        ignore.case = TRUE)  
  
txt.MB <- file.size(files.txt) / 10^6  
sum(txt.MB)
```

```
## [1] 58.43008
```


15.2 Diagramm: Verteilung der Dateigrößen (PDF)

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

```
ggplot(data = dt.plot,
  aes(x = pdf.MB)) +
  geom_density(fill = "#ca2129") +
  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",
    panel.spacing = unit(0.1, "lines"),
    plot.margin = margin(10, 20, 10, 10)
  )
```

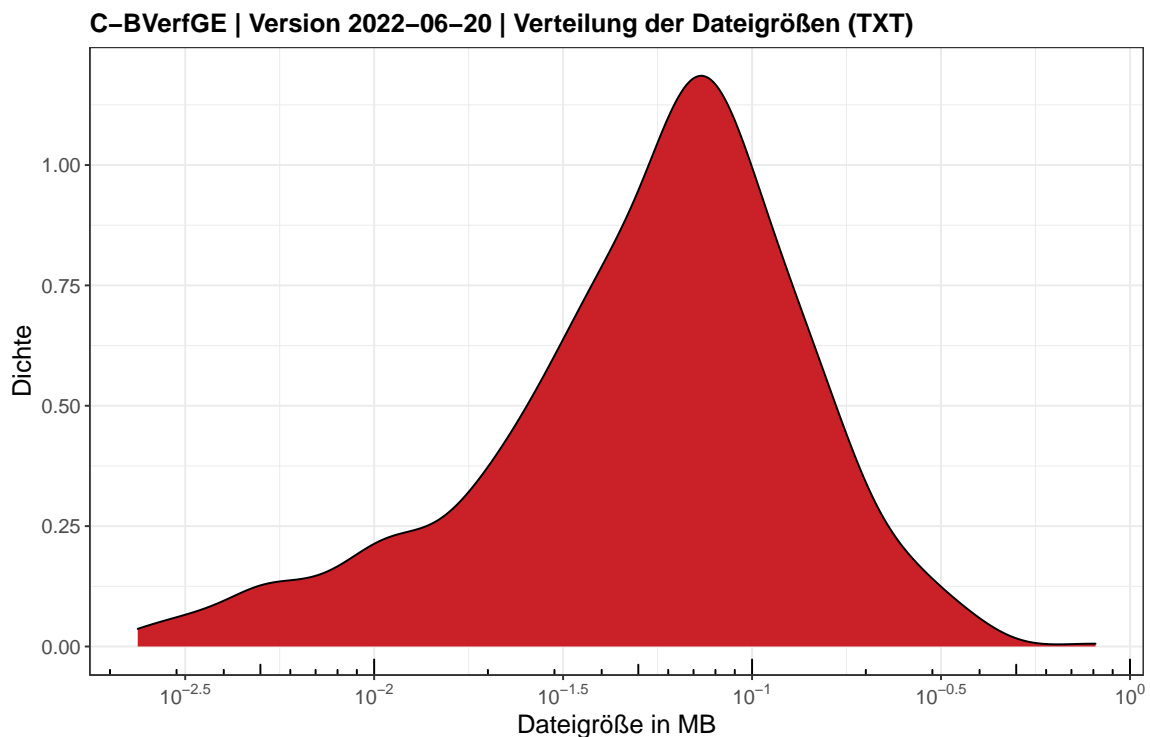


Fobbe | DOI: 10.5281/zenodo.6569896

15.3 Diagramm: Verteilung der Dateigrößen (TXT)

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

```
ggplot(data = dt.plot,
  aes(x = txt.MB)) +
  geom_density(fill = "#ca2129") +
  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",
    panel.spacing = unit(0.1, "lines"),
    plot.margin = margin(10, 20, 10, 10)
  )
```



Fobbe | DOI: 10.5281/zenodo.6569896

16 Erstellen der ZIP-Archive

16.1 Verpacken der CSV-Dateien

16.1.1 Vollständiger Datensatz

```
csvname.full.zip <- gsub(".csv",  
                        ".zip",  
                        csvname.full)  
  
zip(csvname.full.zip,  
    csvname.full)  
  
unlink(csvname.full)
```

16.1.2 Metadaten

```
csvname.meta.zip <- gsub(".csv",  
                        ".zip",  
                        csvname.meta)  
  
zip(csvname.meta.zip,  
    csvname.meta)  
  
unlink(csvname.meta)
```

16.1.3 Segmentiert

```
csvname.segmented.zip <- gsub(".csv",  
                             ".zip",  
                             csvname.segmented)  
  
zip(csvname.segmented.zip,  
    csvname.segmented)  
  
unlink(csvname.segmented)
```

16.1.4 Annotiert

```
if (config$annotate$toggle == TRUE){  
  
    csvname.annotated.zip <- gsub(".csv",  
                                ".zip",  
                                csvname.annotated)  
  
    zip(csvname.annotated.zip,  
        csvname.annotated)
```

```
    unlink(csvname.annotated)
}
```

16.2 Verpacken der PDF-Dateien

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

```
zip(paste(prefix.files,
           "DE_PDF_Datensatz.zip",
           sep = "_"),
    files.pdf)

unlink(files.pdf)
```

16.3 Verpacken der HTML-Dateien

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

```
zip(paste(prefix.files,
           "DE_HTML_Datensatz.zip",
           sep = "_"),
    files.html)

unlink(files.html)
```

16.4 Verpacken der TXT-Dateien

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

```
zip(paste(prefix.files,
           "DE_TXT_Datensatz.zip",
           sep = "_"),
    files.txt)

unlink(files.txt)
```

16.5 Verpacken der Analyse-Dateien

```
zip(paste0(prefix.files,  
           "_DE_ANALYSE.zip"),  
    basename(dir.analysis))
```

16.6 Verpacken der Source-Dateien

```
files.source <- c(list.files(pattern = "\\R$|\\.toml$"),  
                  "CHANGELOG.md",  
                  "README.md",  
                  "R-fobbe-proto-package",  
                  "buttons",  
                  "data",  
                  "functions",  
                  "tex",  
                  "gpg",  
                  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(paste(prefix.files,  
          "Source_Files.zip",  
          sep = "_"),  
    files.source)
```

17 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 sind äußerst resistent gegenüber *collision* und *pre-imaging* Angriffen, sie gelten derzeit als kryptographisch sicher. Ein SHA3-Hash mit 512 bit Länge ist nach Stand von Wissenschaft und Technik auch gegenüber quantenkryptoanalytischen Verfahren unter Einsatz des *Grover-Algorithmus* hinreichend resistent.

17.1 Liste der ZIP-Archive erstellen

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

17.2 Funktion anzeigen: future_multihashes

```
print(f.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,  
##                                             f.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(hashes.table)
##
## }
```

17.3 Hashes berechnen

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

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

```
## Processing 9 files. Begin at: 2022-06-20 23:51:56
```

```
## Processed 9 files. Runtime was 0.82 secs.
```

17.4 In Data Table umwandeln

```
setDT(multihashes)

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

17.5 Index hinzufügen

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

17.6 In Datei schreiben

```
fwrite(multihashes,
      file.path("output",
                paste(prefix.files,
                      "KryptographischeHashes.csv",
                      sep = "_")),
      na = "NA")
```

17.7 Leerzeichen hinzufügen um Zeilenumbruch zu ermöglichen

Hierbei handelt es sich lediglich um eine optische Notwendigkeit. Die normale 128 Zeichen lange Zeichenfolge wird ansonsten nicht umgebrochen und verschwindet über die Seiten-
grenze. Das Leerzeichen erlaubt den automatischen Zeilenumbruch und damit einen für
Menschen sinnvoll lesbaren Abdruck im Codebook. Diese Variante wird nur zur Anzeige
verwendet und danach verworfen.

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

17.8 In Bericht anzeigen

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

index	filename
1	C-BVerfGE_2022-06-20_DE_ANALYSE.zip
2	C-BVerfGE_2022-06-20_DE_CSV_Annotiert.zip
3	C-BVerfGE_2022-06-20_DE_CSV_Datensatz.zip
4	C-BVerfGE_2022-06-20_DE_CSV_Metadaten.zip
5	C-BVerfGE_2022-06-20_DE_CSV_Segmentiert.zip
6	C-BVerfGE_2022-06-20_DE_HTML_Datensatz.zip
7	C-BVerfGE_2022-06-20_DE_PDF_Datensatz.zip
8	C-BVerfGE_2022-06-20_DE_TXT_Datensatz.zip
9	C-BVerfGE_2022-06-20_Source_Files.zip


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

index	sha2.256
1	70634563e7ac3326813756b6a3022bd3987c626e27ca8e61ec9575f6970b5b46
2	04dd910b6c7f17b459c7f18d4ca7ffcb9b0b14a26f211733df92c0a80073f114
3	fbea277294753216ca8515cd9331b1edad06c91086d53d0bacbc5f6c95ba40f2
4	ba2bd8d68cc4f110b7c6f92c9281511f36d89f192ff2583bac11e67da1e610f9
5	c44cd0f77d497b2eee470e0574fa9ef309841820027e1e376c248777686aed64
6	31468a4482063fddc065f88cbb9fde86d381a1673ca1e1e5c0ccd50902069e5d
7	a44ffad5d12486fa0eebd029ebfcf95369269f8e47eb65d61d2e7af1892eee27
8	2461dac170732589a7a05d7979fadf10f94f7c09d858a9b6197963aa2e1fdbd1
9	a6d7e3c9cfbf40eb89b9d0c6713563ff4ca43352c0c4216683732b3c80deb474

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

index	sha3.512
1	2e363654957fd612961449b498f28e6df23ff2f43d0f268c1cfc455c8005549e 423bc5a4958dbb0f173cf3548c5712ef795348ae5728b460fdfaa0ff7c272bd8
2	f7057185eab8f20b805b62f1daa0d58af870699acd92c6a862b1df217fb39bcb e8b4113866f7347baf27d510e171ec61119e31fd98db3cb70a6b6a73ceaeaa74
3	1fa6a9af84c1cec17d290ef28b29cc419a934046d5dc4f41274cb82112026819 ec8e990b44c9b2254470eafb352cc37cb0c7988c2bd69533d15da052b98776c7
4	55a15c3780c56901eea8652a422b2119a76a55e7cce4bcd52168aed4529276e a8110c068d76183b28e06ba715f49056f9b0510527c9877d40af21d832948c24
5	d895fee989d4e05676449d8e17918f32cc5ce2fa38f488a6fa91a92e8b275a18 483998107800838575f8ca7c79c1fff0504c71aef7b17486ab651f7a206b8cd0

6	7a85979dab65c6fdc32a291ca2e250d1235d14db3298f239f0cebb453cc12229fec5539b92944b29eb56e10f6a47565958c5ee4a25acc667cdbc315e25a72bdd
7	269111142255aa078cf95ad4fd7bf9c063dbd6d09fbf3e0e3478f68dce04b98f16306e5b60791e55e2e22d38c6c43ff1682008f2fef3d6ace7263899aa163fe6
8	49f2c16dde78e422dfb6cb3d57104a8e6cdabbbae348bde3354d9c5f8370be1e63bc3ec7af389e4e4a9a3b7b292311688d5016dd158a27af1aa62f029658f39be
9	52a98bf43fecbe3bd1a2743bb1f04270773efceffb5932f0d0cf5b86984b2190d0897e028c17a6ef3ab7af2617026f206a4bd11f0b1d0461fe4b6e22a036e186

18 Aufräumen

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

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

print(files.output)
```

```
## [1] "C-BVerfGE_2022-06-20_DE_ANALYSE.zip"
## [2] "C-BVerfGE_2022-06-20_DE_CSV_Annotiert.zip"
## [3] "C-BVerfGE_2022-06-20_DE_CSV_Datensatz.zip"
## [4] "C-BVerfGE_2022-06-20_DE_CSV_Metadaten.zip"
## [5] "C-BVerfGE_2022-06-20_DE_CSV_Segmentiert.zip"
## [6] "C-BVerfGE_2022-06-20_DE_HTML_Datensatz.zip"
## [7] "C-BVerfGE_2022-06-20_DE_PDF_Datensatz.zip"
## [8] "C-BVerfGE_2022-06-20_DE_TXT_Datensatz.zip"
## [9] "C-BVerfGE_2022-06-20_Source_Files.zip"
```

```
file.rename(files.output,
            output.destination)
```

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

19 Abschluss

19.1 Datumsstempel

```
print(datestamp)
```

```
## [1] "2022-06-20"
```

19.2 Datum und Uhrzeit (Anfang)

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

```
## [1] "2022-06-20 22:53:32 CEST"
```

19.3 Datum und Uhrzeit (Ende)

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

```
## [1] "2022-06-20 23:51:57 CEST"
```

19.4 Laufzeit des gesamten Skriptes

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

```
## Time difference of 58.41247 mins
```

19.5 Warnungen

```
warnings()
```

20 Parameter für strenge Replikationen

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

```
## [1] "OpenSSL 1.1.1o 3 May 2022"
```

```
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.3.1
##
## 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] stats      graphics  grDevices  utils      datasets  methods   base
##
## other attached packages:
##  [1] future.apply_1.8.1 future_1.23.0   spacyr_1.2.1   quanteda_3.2.0
##  [5] readtext_0.81      data.table_1.14.2 scales_1.1.1   ggplot2_3.3.5
##  [9] pdftools_3.0.1     kableExtra_1.3.4 knitr_1.33     rvest_1.0.2
## [13] httr_1.4.2         mgsub_1.7.3    RcppTOML_0.1.7 magick_2.7.3
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_1.0.7          here_1.0.1      svglite_2.0.0   lattice_0.20-45
##  [5] listenv_0.8.0       png_0.1-7       rprojroot_2.0.2 digest_0.6.29
##  [9] utf8_1.2.2          parallelly_1.30.0 R6_2.5.1        evaluate_0.14
## [13] highr_0.9           pillar_1.6.4    rlang_0.4.12    curl_4.3.2
## [17] rstudioapi_0.13     Matrix_1.4-0    reticulate_1.22 rmarkdown_2.11
## [21] qpdf_1.1            labeling_0.4.2  webshot_0.5.2   stringr_1.4.0
## [25] selectr_0.4-2       munsell_0.5.0   compiler_4.0.5  xfun_0.29
## [29] pkgconfig_2.0.3     askpass_1.1     systemfonts_1.0.3 globals_0.14.0
## [33] htmltools_0.5.2     tibble_3.1.6    codetools_0.2-18 fansi_1.0.0
## [37] viridisLite_0.4.0   crayon_1.4.2    withr_2.4.3     rappdirs_0.3.3
## [41] grid_4.0.5          jsonlite_1.7.2  gtable_0.3.0    lifecycle_1.0.1
## [45] magrittr_2.0.1      RcppParallel_5.1.4 stringi_1.7.6    farver_2.1.0
## [49] renv_0.15.4         xml2_1.3.3      ellipsis_0.3.2  stopwords_2.3
## [53] vctrs_0.3.8         fastmatch_1.1-3 tools_4.0.5     glue_1.6.0
## [57] parallel_4.0.5      fastmap_1.1.0   yaml_2.2.1      colorspace_2.0-2
```

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, and Akitaka Matsuo. 2020. *Spacyr: Wrapper to the spaCy 'Nlp' Library*. <https://spacyr.quanteda.io>.
- Benoit, Kenneth, and Adam Obeng. 2021. *Readtext: Import and Handling for Plain and Formatted Text Files*. <https://github.com/quanteda/readtext>.
- 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>.
- 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>.
- Ewing, Mark. 2021. *Mgsub: Safe, Multiple, Simultaneous String Substitution*. <https://CRAN.R-project.org/package=mgsub>.
- Ooms, Jeroen. 2021a. *Magick: Advanced Graphics and Image-Processing in R*. <https://CRAN.R-project.org/package=magick>.
- . 2021b. *Pdftools: Text Extraction, Rendering and Converting of Pdf Documents*. <https://CRAN.R-project.org/package=pdftools>.
- 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>.
- . 2020. *Httr: Tools for Working with Urls and Http*. <https://CRAN.R-project.org/package=httr>.
- . 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, 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>.