

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

COMPILATION REPORT

Version 2023-02-20

License MIT-0

DOI: 10.5281/zenodo.7655164

<b>Titel</b>	Source Code des »Corpus der amtlichen Entscheidungssammlung des Bundesverfassungsgerichts«
<b>Abkürzung</b>	C-BVerfGE-Source
<b>Autor</b>	Seán Fobbe
<b>Version</b>	2023-02-20
<b>Download</b>	<a href="https://doi.org/10.5281/zenodo.7655164">https://doi.org/10.5281/zenodo.7655164</a>
<b>Lizenz</b>	MIT No Attribution (MIT-0)

### Zitiervorschlag

*Seán Fobbe* (2023). Source Code des »Corpus der amtlichen Entscheidungssammlung des Bundesverfassungsgerichts« (C-BVerfGE-Source). Version 2023-02-20. Zenodo. DOI: 10.5281/zenodo.7655164.

### 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 2023-02-20. 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 — 2023— Seán Fobbe

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

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

### Disclaimer

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

# Inhaltsverzeichnis

<b>1</b>	<b>Corpus der amtlichen Entscheidungssammlung des Bundesverfassungsgerichts (C-BVerfGE)</b>	<b>9</b>
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
<b>2</b>	<b>Vorbereitung</b>	<b>11</b>
2.1	Datumsstempel . . . . .	11
2.2	Datum und Uhrzeit (Beginn) . . . . .	11
2.3	Packages Laden . . . . .	11
2.4	Zusätzliche Funktionen einlesen . . . . .	12
2.5	Verzeichnis für Analyse-Ergebnisse und Diagramme definieren . . . . .	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
<b>3</b>	<b>Download: Entscheidungen der BVerfGE</b>	<b>19</b>
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
<b>4</b>	<b>PDF-Download</b>	<b>28</b>
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
<b>5</b>	<b>HTML-Download</b>	<b>32</b>
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
<b>6</b>	<b>HTML verarbeiten</b>	<b>36</b>
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
<b>7</b>	<b>Text-Extraktion aus PDF</b>	<b>43</b>
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
<b>8</b>	<b>Korpus Erstellen</b>	<b>45</b>
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
<b>9</b>	<b>Frequenztabellen erstellen</b>	<b>56</b>
9.1	Funktion anzeigen . . . . .	56
9.2	Ignorierte Variablen . . . . .	57
9.3	Liste zu prüfender Variablen . . . . .	57
9.4	Frequenztabellen erstellen . . . . .	58
<b>10</b>	<b>Frequenztabellen visualisieren</b>	<b>85</b>
10.1	Präfix erstellen . . . . .	85
10.2	Tabellen einlesen . . . . .	85
10.3	Diagramm: Typ der Entscheidung . . . . .	86
10.4	Diagramm: Typ des Spruchkörpers . . . . .	87
10.5	Diagramm: Spruchkörper nach Aktenzeichen . . . . .	88
10.6	Diagramm: Registerzeichen . . . . .	89
10.7	Diagramm: Präsident:in . . . . .	91
10.8	Diagramm: Vize-Präsident:in . . . . .	92
10.9	Diagramm: Entscheidungsjahr . . . . .	93
10.10	Diagramm: Eingangsjahr (ISO) . . . . .	94
<b>11</b>	<b>Korpus-Analytik</b>	<b>95</b>
11.1	Berechnung linguistischer Kennwerte . . . . .	95
11.1.1	Funktion anzeigen . . . . .	95
11.1.2	Berechnung durchführen . . . . .	96
11.2	Variablen-Namen anpassen . . . . .	97
11.3	Kennwerte dem Korpus hinzufügen . . . . .	97
11.4	Variante mit Metadaten erstellen . . . . .	97
11.5	Linguistische Kennwerte . . . . .	97
11.5.1	Zusammenfassungen berechnen . . . . .	97

11.5.2 Zusammenfassungen anzeigen . . . . .	98
11.5.3 Zusammenfassungen speichern . . . . .	98
11.6 Quantitative Variablen . . . . .	100
11.6.1 Entscheidungsdatum . . . . .	100
11.6.2 Zusammenfassungen berechnen . . . . .	100
11.6.3 Zusammenfassungen anzeigen . . . . .	101
11.6.4 Zusammenfassungen speichern . . . . .	101
11.7 Verteilungen linguistischer Kennwerte . . . . .	102
11.7.1 Diagramm: Verteilung Zeichen . . . . .	102
11.7.2 Diagramm: Verteilung Tokens . . . . .	102
11.7.3 Diagramm: Verteilung Typen . . . . .	104
11.7.4 Diagramm: Verteilung Sätze . . . . .	105
<b>12 Linguistische Annotationen berechnen</b>	<b>106</b>
12.1 Funktion anzeigen: f.future_spacyparse . . . . .	106
12.2 Berechnungen durchführen . . . . .	107
<b>13 Kontrolle der Variablen</b>	<b>108</b>
13.1 Semantische Sortierung der Variablen . . . . .	108
13.1.1 Variablen sortieren: Hauptdatensatz . . . . .	108
13.1.2 Variablen sortieren: Metadaten . . . . .	109
13.1.3 Variablen sortieren: Segmentiert . . . . .	110
13.2 Anzahl Variablen der Datensätze . . . . .	111
13.3 Alle Variablen-Namen der Datensätze . . . . .	111
<b>14 CSV-Dateien erstellen</b>	<b>113</b>
14.1 CSV mit vollem Datensatz speichern . . . . .	113
14.2 CSV mit Metadaten speichern . . . . .	113
14.3 CSV mit Segmenten speichern . . . . .	113
14.4 CSV mit Annotationen speichern . . . . .	113
<b>15 Dateigrößen analysieren</b>	<b>115</b>
15.1 Gesamtgröße . . . . .	115
15.1.1 Korpus-Objekt in RAM (MB) . . . . .	115
15.1.2 CSV Korpus (MB) . . . . .	115
15.1.3 CSV Metadaten (MB) . . . . .	115
15.1.4 CSV Annotiert (MB) . . . . .	115
15.1.5 CSV Segmentiert (MB) . . . . .	115
15.1.6 PDF-Dateien (MB) . . . . .	116
15.1.7 TXT-Dateien (MB) . . . . .	116
15.2 Diagramm: Verteilung der Dateigrößen (PDF) . . . . .	117
15.3 Diagramm: Verteilung der Dateigrößen (TXT) . . . . .	118
<b>16 Erstellen der ZIP-Archive</b>	<b>119</b>
16.1 Verpacken der CSV-Dateien . . . . .	119
16.1.1 Vollständiger Datensatz . . . . .	119
16.1.2 Metadaten . . . . .	119
16.1.3 Segmentiert . . . . .	119
16.1.4 Annotiert . . . . .	119
16.2 Verpacken der PDF-Dateien . . . . .	120

16.3 Verpacken der HTML-Dateien . . . . .	120
16.4 Verpacken der TXT-Dateien . . . . .	120
16.5 Verpacken der Analyse-Dateien . . . . .	121
16.6 Verpacken der Source-Dateien . . . . .	121
<b>17 Kryptographische Hashes</b>	<b>122</b>
17.1 Liste der ZIP-Archive erstellen . . . . .	122
17.2 Funktion anzeigen: future_multihashes . . . . .	122
17.3 Hashes berechnen . . . . .	123
17.4 In Data Table umwandeln . . . . .	123
17.5 Index hinzufügen . . . . .	123
17.6 In Datei schreiben . . . . .	124
17.7 Leerzeichen hinzufügen um Zeilenumbruch zu ermöglichen . . . . .	124
17.8 In Bericht anzeigen . . . . .	124
<b>18 Aufräumen</b>	<b>127</b>
<b>19 Abschluss</b>	<b>128</b>
19.1 Datumsstempel . . . . .	128
19.2 Datum und Uhrzeit (Anfang) . . . . .	128
19.3 Datum und Uhrzeit (Ende) . . . . .	128
19.4 Laufzeit des gesamten Skriptes . . . . .	128
19.5 Warnungen . . . . .	128
<b>20 Parameter für strenge Replikationen</b>	<b>129</b>
<b>Literaturverzeichnis</b>	<b>130</b>



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

## 1.1 Überblick

Dieser Code lädt alle auf [www.bundesverfassungsgericht.de](http://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](mailto: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] "2023-02-20"
```

### 2.2 Datum und Uhrzeit (Beginn)

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

```
## [1] "2023-02-20 00:09:30 CET"
```

### 2.3 Packages Laden

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

```
## Linking to ImageMagick 6.9.12.52
## 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 22.01.0
```

```
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: 69.1
```

```
## 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.7655163"  
## ..$ personendaten: chr "10.5281/zenodo.4568682"  
## ..$ software      :List of 2  
## .. ..$ concept: chr "10.5281/zenodo.4265933"  
## .. ..$ version: chr "10.5281/zenodo.7655164"  
## $ 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.7655163"
```

### 2.10.6 Präfix für Dateien definieren

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

```
## [1] "C-BVerfGE_2023-02-20"
```

## 2.10.7 Präfix für Diagramme definieren

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

```
## [1] "C-BVerfGE | Version 2023-02-20"
```

## 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] "2023-02-20 00:09:31 CET"
```

### 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_2bvff000120_1",
                  "fs20210325_2bvff000120",
                  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] "2023-02-20 00:09:40 CET"
```

### 3.7 Dauer: Linksammlung

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

```
## Time difference of 8.659401 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.<sup>1</sup>

<sup>1</sup> [https://e-justice.europa.eu/content\\_european\\_case\\_law\\_identifier\\_ecli-175-de-de.do?member=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
  "_",
  "[0-9ßÄÜÖäüöA-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] "2023-02-20 00:09:40 CET"
```

### 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] "2023-02-20 00:38:04 CET"
```

## 4.5 Dauer: PDF-Download

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

```
## Time difference of 28.39305 mins
```

## 4.6 PDF-Download: Ergebnis

### 4.6.1 Anzahl herunterzuladender Dateien

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

```
## [1] 842
```

### 4.6.2 Anzahl heruntergeladener Dateien

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

```
## [1] 841
```

### 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] 842
```

### 4.8.2 Anzahl heruntergeladener Dateien

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

```
## [1] 841
```

### 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] "2023-02-20 00:38:06 CET"
```

### 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] "2023-02-20 00:54:02 CET"
```



## 5.5 Dauer: HTML-Download

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

```
## Time difference of 15.93503 mins
```

## 5.6 HTML-Download: Ergebnis

### 5.6.1 Anzahl herunterzuladender Dateien

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

```
## [1] 842
```

### 5.6.2 Anzahl heruntergeladener Dateien

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

```
## [1] 842
```

### 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] 842
```

### 5.8.2 Anzahl heruntergeladener Dateien

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

```
## [1] 842
```

### 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] 841
```

### 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 841 files. Begin at: 2023-02-20 00:54:40

```

```

## Successfully processed 841 files. 0 files failed. Runtime was 4.34 secs. Ended
  at: 2023-02-20 00:54:44

```

## 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

```
vpPraesi <- fread(file.path("data",
  "PVP-FCG_2021-04-08_GermanFederalCourts_VicePresidents.
  csv"))
vpPraesi <- vpPraesi[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,  
                                   "/" ,  
                                   formatC(txt.bverfg$eingangsjahr_az, flag = "0",  
                                             width = 2))
```

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$data$conzept,  
                               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.<sup>2</sup>

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)

<sup>2</sup> [https://e-justice.europa.eu/content\\_european\\_case\\_law\\_identifier\\_ecli-175-de-de.do?member=1](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.<sup>3</sup>

```
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.<sup>4</sup>

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

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

<sup>4</sup> <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$zitativvorschlag,  
                             ignore.case = TRUE)  
  
entscheidung_typ.segmented <- gsub(".*(Beschluss|Urteil|Verfügung).*",  
                                   "\\1",  
                                   dt.segmented.full$zitativvorschlag,  
                                   ignore.case = TRUE)
```

### 8.17.2 Kürzen

```
lang.etyp <- c("Urteil",  
              "Beschluss",  
              "Verfügung")  
  
kurz.etyp <- c("U",  
              "B",  
              "V")  
  
entscheidung_typ.main <- mgsub(entscheidung_typ.main,  
                               lang.etyp,  
                               kurz.etyp,  
                               ignore.case = TRUE)  
  
entscheidung_typ.segmented <- mgsub(entscheidung_typ.segmented,  
                                    lang.etyp,  
                                    kurz.etyp,  
                                    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	841	100	100	100
Total	841	100	100	100

Frequency Table for Variable: spruchkoerper\_typ

3 unique value(s) detected.

spruchkoerper_typ	N	exactpercent	roundedpercent	cumulpercent
K	3	0.3567182	0.36	0.36
P	2	0.2378121	0.24	0.59

(continued)

spruchkoerper_typ	N	exactpercent	roundedpercent	cumulpercent
S	836	99.4054697	99.41	100.00
Total	841	100.0000000	100.00	100.00

Frequency Table for Variable: spruchkoerper\_az

2 unique value(s) detected.

spruchkoerper_az	N	exactpercent	roundedpercent	cumulpercent
1	386	45.89774	45.9	45.9
2	455	54.10226	54.1	100.0
Total	841	100.00000	100.0	100.0

Frequency Table for Variable: registerzeichen

14 unique value(s) detected.

registerzeichen	N	exactpercent	roundedpercent	cumulpercent
BvB	17	2.0214031	2.02	2.02
BvC	28	3.3293698	3.33	5.35
BvE	90	10.7015458	10.70	16.05
BvF	53	6.3020214	6.30	22.35
BvG	8	0.9512485	0.95	23.31
BvH	5	0.5945303	0.59	23.90
BvK	9	1.0701546	1.07	24.97
BvL	160	19.0249703	19.02	44.00
BvM	4	0.4756243	0.48	44.47
BvN	1	0.1189061	0.12	44.59
BvP	1	0.1189061	0.12	44.71
BvQ	26	3.0915577	3.09	47.80

(continued)

registerzeichen	N	exactpercent	roundedpercent	cumulpercent
BvR	437	51.9619501	51.96	99.76
PBvU	2	0.2378121	0.24	100.00
Total	841	100.0000000	100.00	100.00

Frequency Table for Variable: eingangsjahr\_az

45 unique value(s) detected.

eingangsjahr_az	N	exactpercent	roundedpercent	cumulpercent
0	27	3.2104637	3.21	3.21
1	36	4.2806183	4.28	7.49
2	33	3.9239001	3.92	11.41
3	31	3.6860880	3.69	15.10
4	28	3.3293698	3.33	18.43
5	29	3.4482759	3.45	21.88
6	19	2.2592152	2.26	24.14
7	39	4.6373365	4.64	28.78
8	23	2.7348395	2.73	31.51
9	29	3.4482759	3.45	34.96
10	27	3.2104637	3.21	38.17
11	40	4.7562426	4.76	42.93
12	38	4.5184304	4.52	47.44
13	37	4.3995244	4.40	51.84
14	24	2.8537455	2.85	54.70
15	32	3.8049941	3.80	58.50
16	16	1.9024970	1.90	60.40
17	17	2.0214031	2.02	62.43
18	16	1.9024970	1.90	64.33
19	16	1.9024970	1.90	66.23

(continued)

eingangsjahr_az	N	exactpercent	roundedpercent	cumulpercent
20	15	1.7835910	1.78	68.01
21	15	1.7835910	1.78	69.80
22	2	0.2378121	0.24	70.04
51	2	0.2378121	0.24	70.27
52	2	0.2378121	0.24	70.51
56	1	0.1189061	0.12	70.63
58	1	0.1189061	0.12	70.75
77	1	0.1189061	0.12	70.87
83	3	0.3567182	0.36	71.22
84	2	0.2378121	0.24	71.46
85	2	0.2378121	0.24	71.70
86	2	0.2378121	0.24	71.94
87	3	0.3567182	0.36	72.29
88	6	0.7134364	0.71	73.01
89	5	0.5945303	0.59	73.60
90	10	1.1890606	1.19	74.79
91	22	2.6159334	2.62	77.41
92	15	1.7835910	1.78	79.19
93	22	2.6159334	2.62	81.81
94	22	2.6159334	2.62	84.42
95	27	3.2104637	3.21	87.63
96	28	3.3293698	3.33	90.96
97	21	2.4970273	2.50	93.46
98	31	3.6860880	3.69	97.15
99	24	2.8537455	2.85	100.00
Total	841	100.0000000	100.00	100.00

---

Frequency Table for Variable: kollision

---

4 unique value(s) detected.

kollision	N	exactpercent	roundedpercent	cumulpercent
NA	831	98.8109394	98.81	98.81
a	7	0.8323424	0.83	99.64
b	2	0.2378121	0.24	99.88
c	1	0.1189061	0.12	100.00
Total	841	100.0000000	100.00	100.00

Frequency Table for Variable: band

81 unique value(s) detected.

band	N	exactpercent	roundedpercent	cumulpercent
1	3	0.3567182	0.36	0.36
6	1	0.1189061	0.12	0.48
7	1	0.1189061	0.12	0.59
10	1	0.1189061	0.12	0.71
46	1	0.1189061	0.12	0.83
62	1	0.1189061	0.12	0.95
65	1	0.1189061	0.12	1.07
85	7	0.8323424	0.83	1.90
86	3	0.3567182	0.36	2.26
87	11	1.3079667	1.31	3.57
88	10	1.1890606	1.19	4.76
89	2	0.2378121	0.24	4.99
90	2	0.2378121	0.24	5.23
91	1	0.1189061	0.12	5.35
92	1	0.1189061	0.12	5.47
93	13	1.5457788	1.55	7.02
96	5	0.5945303	0.59	7.61
97	18	2.1403092	2.14	9.75

(continued)

band	N	exactpercent	roundedpercent	cumulpercent
98	14	1.6646849	1.66	11.41
99	26	3.0915577	3.09	14.51
100	14	1.6646849	1.66	16.17
101	15	1.7835910	1.78	17.95
102	17	2.0214031	2.02	19.98
103	19	2.2592152	2.26	22.24
104	26	3.0915577	3.09	25.33
105	14	1.6646849	1.66	26.99
106	17	2.0214031	2.02	29.01
107	14	1.6646849	1.66	30.68
108	18	2.1403092	2.14	32.82
109	12	1.4268728	1.43	34.24
110	14	1.6646849	1.66	35.91
111	15	1.7835910	1.78	37.69
112	18	2.1403092	2.14	39.83
113	10	1.1890606	1.19	41.02
114	13	1.5457788	1.55	42.57
115	12	1.4268728	1.43	44.00
116	11	1.3079667	1.31	45.30
117	14	1.6646849	1.66	46.97
118	9	1.0701546	1.07	48.04
119	10	1.1890606	1.19	49.23
120	10	1.1890606	1.19	50.42
121	10	1.1890606	1.19	51.61
122	13	1.5457788	1.55	53.15
123	9	1.0701546	1.07	54.22
124	12	1.4268728	1.43	55.65
125	7	0.8323424	0.83	56.48
126	13	1.5457788	1.55	58.03

(continued)

band	N	exactpercent	roundedpercent	cumulpercent
127	10	1.1890606	1.19	59.22
128	13	1.5457788	1.55	60.76
129	12	1.4268728	1.43	62.19
130	11	1.3079667	1.31	63.50
131	11	1.3079667	1.31	64.80
132	13	1.5457788	1.55	66.35
133	14	1.6646849	1.66	68.01
134	14	1.6646849	1.66	69.68
135	11	1.3079667	1.31	70.99
136	11	1.3079667	1.31	72.29
137	9	1.0701546	1.07	73.37
138	12	1.4268728	1.43	74.79
139	10	1.1890606	1.19	75.98
140	12	1.4268728	1.43	77.41
141	8	0.9512485	0.95	78.36
142	13	1.5457788	1.55	79.90
143	8	0.9512485	0.95	80.86
144	4	0.4756243	0.48	81.33
145	9	1.0701546	1.07	82.40
146	7	0.8323424	0.83	83.23
147	9	1.0701546	1.07	84.30
148	10	1.1890606	1.19	85.49
149	13	1.5457788	1.55	87.04
150	8	0.9512485	0.95	87.99
151	9	1.0701546	1.07	89.06
152	12	1.4268728	1.43	90.49
153	6	0.7134364	0.71	91.20
154	7	0.8323424	0.83	92.03
155	8	0.9512485	0.95	92.98



(continued)

band	N	exactpercent	roundedpercent	cumulpercent
156	10	1.1890606	1.19	94.17
157	8	0.9512485	0.95	95.12
158	14	1.6646849	1.66	96.79
159	14	1.6646849	1.66	98.45
160	13	1.5457788	1.55	100.00
Total	841	100.0000000	100.00	100.00

---

Frequency Table for Variable: entscheidungsjahr

---

39 unique value(s) detected.

entscheidungsjahr	N	exactpercent	roundedpercent	cumulpercent
1951	1	0.1189061	0.12	0.12
1952	2	0.2378121	0.24	0.36
1957	1	0.1189061	0.12	0.48
1958	1	0.1189061	0.12	0.59
1959	1	0.1189061	0.12	0.71
1977	1	0.1189061	0.12	0.83
1983	2	0.2378121	0.24	1.07
1991	1	0.1189061	0.12	1.19
1992	22	2.6159334	2.62	3.80
1993	10	1.1890606	1.19	4.99
1994	3	0.3567182	0.36	5.35
1995	11	1.3079667	1.31	6.66
1996	3	0.3567182	0.36	7.02
1997	9	1.0701546	1.07	8.09
1998	51	6.0642093	6.06	14.15
1999	31	3.6860880	3.69	17.84
2000	20	2.3781213	2.38	20.21

(continued)

entscheidungsjahr	N	exactpercent	roundedpercent	cumulpercent
2001	37	4.3995244	4.40	24.61
2002	37	4.3995244	4.40	29.01
2003	38	4.5184304	4.52	33.53
2004	41	4.8751486	4.88	38.41
2005	38	4.5184304	4.52	42.93
2006	25	2.9726516	2.97	45.90
2007	27	3.2104637	3.21	49.11
2008	29	3.4482759	3.45	52.56
2009	29	3.4482759	3.45	56.00
2010	29	3.4482759	3.45	59.45
2011	24	2.8537455	2.85	62.31
2012	35	4.1617122	4.16	66.47
2013	28	3.3293698	3.33	69.80
2014	37	4.3995244	4.40	74.20
2015	29	3.4482759	3.45	77.65
2016	29	3.4482759	3.45	81.09
2017	27	3.2104637	3.21	84.30
2018	30	3.5671819	3.57	87.87
2019	21	2.4970273	2.50	90.37
2020	30	3.5671819	3.57	93.94
2021	41	4.8751486	4.88	98.81
2022	10	1.1890606	1.19	100.00
Total	841	100.0000000	100.00	100.00

---

Frequency Table for Variable: eingangsjahr\_iso

---

45 unique value(s) detected.

eingangsjahr_iso	N	exactpercent	roundedpercent	cumulpercent
1951	2	0.2378121	0.24	0.24
1952	2	0.2378121	0.24	0.48
1956	1	0.1189061	0.12	0.59
1958	1	0.1189061	0.12	0.71
1977	1	0.1189061	0.12	0.83
1983	3	0.3567182	0.36	1.19
1984	2	0.2378121	0.24	1.43
1985	2	0.2378121	0.24	1.66
1986	2	0.2378121	0.24	1.90
1987	3	0.3567182	0.36	2.26
1988	6	0.7134364	0.71	2.97
1989	5	0.5945303	0.59	3.57
1990	10	1.1890606	1.19	4.76
1991	22	2.6159334	2.62	7.37
1992	15	1.7835910	1.78	9.16
1993	22	2.6159334	2.62	11.77
1994	22	2.6159334	2.62	14.39
1995	27	3.2104637	3.21	17.60
1996	28	3.3293698	3.33	20.93
1997	21	2.4970273	2.50	23.42
1998	31	3.6860880	3.69	27.11
1999	24	2.8537455	2.85	29.96
2000	27	3.2104637	3.21	33.17
2001	36	4.2806183	4.28	37.46
2002	33	3.9239001	3.92	41.38
2003	31	3.6860880	3.69	45.07
2004	28	3.3293698	3.33	48.39
2005	29	3.4482759	3.45	51.84
2006	19	2.2592152	2.26	54.10
2007	39	4.6373365	4.64	58.74

(continued)

eingangsjahr_iso	N	exactpercent	roundedpercent	cumulpercent
2008	23	2.7348395	2.73	61.47
2009	29	3.4482759	3.45	64.92
2010	27	3.2104637	3.21	68.13
2011	40	4.7562426	4.76	72.89
2012	38	4.5184304	4.52	77.41
2013	37	4.3995244	4.40	81.81
2014	24	2.8537455	2.85	84.66
2015	32	3.8049941	3.80	88.47
2016	16	1.9024970	1.90	90.37
2017	17	2.0214031	2.02	92.39
2018	16	1.9024970	1.90	94.29
2019	16	1.9024970	1.90	96.20
2020	15	1.7835910	1.78	97.98
2021	15	1.7835910	1.78	99.76
2022	2	0.2378121	0.24	100.00
Total	841	100.0000000	100.00	100.00

Frequency Table for Variable: praesi

9 unique value(s) detected.

praesi	N	exactpercent	roundedpercent	cumulpercent
Benda	3	0.3567182	0.36	0.36
Harbarth	64	7.6099881	7.61	7.97
Herzog	35	4.1617122	4.16	12.13
Höpker-Aschoff	3	0.3567182	0.36	12.49
Limbach	177	21.0463734	21.05	33.53
Müller	1	0.1189061	0.12	33.65
Papier	253	30.0832342	30.08	63.73

(continued)

praesi	N	exactpercent	roundedpercent	cumulpercent
Voßkuhle	303	36.0285375	36.03	99.76
Wintrich	2	0.2378121	0.24	100.00
Total	841	100.0000000	100.00	100.00

---

Frequency Table for Variable: v\_praesi

---

12 unique value(s) detected.

v_praesi	N	exactpercent	roundedpercent	cumulpercent
Harbarth	42	4.9940547	4.99	4.99
Hassemer	207	24.6135553	24.61	29.61
Henschel	9	1.0701546	1.07	30.68
Katz	6	0.7134364	0.71	31.39
Kirchhof	261	31.0344828	31.03	62.43
König	64	7.6099881	7.61	70.04
Limbach	1	0.1189061	0.12	70.15
Mahrenholz	34	4.0428062	4.04	74.20
Papier	144	17.1224732	17.12	91.32
Seidl	24	2.8537455	2.85	94.17
Voßkuhle	46	5.4696790	5.47	99.64
Zeidler	3	0.3567182	0.36	100.00
Total	841	100.0000000	100.00	100.00

---

Frequency Table for Variable: verfahrensart

---

14 unique value(s) detected.

verfahrensart	N	exactpercent	roundedpercent	cumulpercent
Abstrakte Normenkontrolle	53	6.3020214	6.30	6.30

(continued)

verfahrensart	N	exactpercent	roundedpercent	cumulpercent
Andere Streitigkeiten zwischen Bund und Ländern	5	0.5945303	0.59	6.90
Bund-Länder-Streitigkeiten	8	0.9512485	0.95	7.85
Divergenzvorlagen eines Landesverfassungsgerichts zur Auslegung des Grundgesetzes	1	0.1189061	0.12	7.97
Einstweilige Anordnungen	26	3.0915577	3.09	11.06
Feststellung der Anwendbarkeit einer Regel des Völkerge- wohnheitsrechts	4	0.4756243	0.48	11.53
Konkrete Normenkontrolle	160	19.0249703	19.02	30.56
Landesverfassungsstreitigkeiten	9	1.0701546	1.07	31.63
Organstreitverfahren	90	10.7015458	10.70	42.33
Plenarentscheidungen	2	0.2378121	0.24	42.57
Sonstige durch Bundesrecht zugewiesene Verfahren	1	0.1189061	0.12	42.69
Verfassungsbeschwerden; Kommunalverfassungsbe- schwerden	437	51.9619501	51.96	94.65
Verfassungswidrigkeit von Par- teien	17	2.0214031	2.02	96.67
Wahlprüfungsverfahren	28	3.3293698	3.33	100.00
Total	841	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	841	100	100	100
Total	841	100	100	100

---

Frequency Table for Variable: doi\_version

---

1 unique value(s) detected.

doi_version	N	exactpercent	roundedpercent	cumulpercent
10.5281/zenodo.7655163	841	100	100	100
Total	841	100	100	100

---

Frequency Table for Variable: version

---

1 unique value(s) detected.

version	N	exactpercent	roundedpercent	cumulpercent
2023-02-20	841	100	100	100
Total	841	100	100	100

---

Frequency Table for Variable: lizenz

---

1 unique value(s) detected.

lizenz	N	exactpercent	roundedpercent	cumulpercent
Creative Commons Zero 1.0 Universal	841	100	100	100
Total	841	100	100	100

---

Frequency Table for Variable: richter

---

162 unique value(s) detected.

richter	N	exactpercent	roundedpercent	cumulpercent
	12	1.4268728	1.43	1.43
Benda Simon Hesse Katzenstein Niemeyer Helsmer Henschel Niedermier	1	0.1190476	0.12	1.55

(continued)

richter	N	exactpercent	roundedpercent	cumulpercent
Broß Osterloh Di Fabio Mellinghoff Lübbe- Wolff Gerhardt Landau	1	0.1189061	0.12	1.66
Bundesverfassungsrichter  Lim- bach Kirchhof Sommer Jentsch Hassemer Broß Osterloh  Winter ist an der Unter-   schrift verhindert.	1	0.1189061	0.12	1.78
Böckenförde Klein Graßhof Kruis Kirchhof Witzel Sommer	2	0.2378125	0.24	2.02
Dr. Benda Dr. Böhmer Dr. Si- mon Dr. Faller Dr. Hesse Dr. Katzenstein	1	0.1189061	0.12	2.14
Dr. Dr. Höpker- Aschoff Ellinghaus Dr. Heiland Dr. Drath Dr. Stein Wessel Ritterspach Lehmann Dr. Zweigert	1	0.1189061	0.12	2.26
Dr. Dr. Höpker- Aschoff Ellinghaus Dr. Scheffler Dr. Hei- land Dr. Scholtissek Dr. Drath Wessel Ritterspach Lehmann Dr. Zweigert	1	0.1189061	0.12	2.38
Dr. Katz Dr. Fröh- lich Wolff Dr. Roediger Dr. Klaas Henneka Dr. Friesen- hahn Dr. Rupp Dr. Gei- ger Leuser Dr. Federer	1	0.1189061	0.12	2.50
Dr. Müller Dr. Scheffler Dr. Heiland Dr. Heck Dr. Ber- ger Dr. Scholtissek Dr. Drath Dr. Stein Ritterspach	1	0.1189061	0.12	2.62
Dr. Wintrich Dr. Scheff- ler Dr. Heiland Dr. Heck Dr. Scholtissek Dr. Stein Wessel Ritterspach Lehmann	1	0.1189061	0.12	2.73
Eichberger Masing Paulus Baer Britz Ott Chiesi	1	0.1189061	0.12	2.85
Gaier Eichberger Schluckebier Masing Paulus Baer Britz	2	0.2378125	0.24	3.09
Gaier Eichberger Schluckebier Masing Paulus Baer Britz	1	0.1189061	0.12	3.21



(continued)

richter	N	exactpercent	roundedpercent	cumulpercent
Gaier Masing Baer	1	0.1189061	0.12	3.33
Grimm Kühling Jaeger Haas Hömig Stein	1	0.1189061	0.12	3.45
Grimm Kühling Jaeger Haas Hömig Stein	4	0.4756243	0.48	3.92
Grimm Kühling Seibert Jaeger Hömig Stein	1	0.1189061	0.12	4.04
Harbarth Masing Paulus Baer Britz Ott Christ Radtke	9	0.5701546	1.07	5.11
Harbarth Paulus Baer Britz Ott Christ Radtke	1	0.1189061	0.12	5.23
Harbarth Paulus Baer Britz Ott Christ Radtke	16	1.021976	1.90	7.13
Hassemer Broß Di Fabio Lübbe-Wolff Gerhardt	1	0.1189061	0.12	7.25
Hassemer Broß Osterloh Di Fabio Mellinghoff Lübbe-Wolff	1	0.1189061	0.12	7.37
Hassemer Broß Osterloh Di Fabio Mellinghoff Lübbe-Wolff Gerhardt Landau	32	3.8049941	3.80	11.18
Hassemer Jaeger Broß Osterloh Di Fabio Mellinghoff Lübbe-Wolff	1	0.1189061	0.12	11.30
Hassemer Jentsch Broß Di Fabio Mellinghoff Lübbe-Wolff	1	0.1189061	0.12	11.41
Hassemer Jentsch Broß Osterloh Di Fabio Mellinghoff Lübbe-Wolff Gerhard	1	0.1189061	0.12	11.53
Hassemer Jentsch Broß Osterloh Di Fabio Mellinghoff Lübbe-Wolff Gerhardt	37	4.3995244	4.40	15.93
Hassemer Jentsch Broß Osterloh Mellinghoff Lübbe-Wolff Gerhardt	1	0.1189061	0.12	16.05
Hassemer Jentsch Richter Broß ist an der Osterloh Di Fabio Mellinghoff Lübbe-Wolff Gerhardt  Unterschrift verhindert.	1	0.1189061	0.12	16.17

(continued)

richter	N	exactpercent	roundedpercent	cumulpercent
Hassemer Jentsch Richter Broß ist an  Oster- loh Richter Di Fabio ist Mellinghoff Lübbe- Wolff Gerhardt  der Un- terschrift verhindert.   an	1	0.1189061	0.12	16.29
Hassemer Sommer Broß Osterloh Di Fabio Mellinghoff Lübbe- Wolff	1	0.1189061	0.12	16.41
Hassemer Sommer Jentsch Broß Osterloh Di Fabio Mellinghoff	2	0.2378121	0.24	16.65
Hassemer Sommer Jentsch Broß Osterloh Di Fabio Mellinghoff Lübbe- Wolff	3	0.3592152	2.26	18.91
Hassemer Sommer Jentsch Broß Osterloh Di Fabio Mellinghoff Lübbe- Wolff Di Fabio	4	0.4781271	0.12	19.02
Hassemer Sommer Jentsch Osterloh Di Fabio Mellinghoff Lübbe- Wolff	1	0.1189061	0.12	19.14
Hassemer Sommer Jentsch Richter Broß ist  Osterloh Di Fabio Mellinghoff Lübbe- Wolff  an der Unterschrift gehindert.   Hassemer	1	0.1189061	0.12	19.26
Henschel Seidl Grimm Kühling Seibert Jaeger Henschel Seidl Grimm Söllner Kühling Seibert Henschel Seidl Grimm Söllner Kühling Seibert Legner Haas	1 1 2	0.1189061 0.1189061 0.2378121	0.12 0.12 0.24	19.38 19.50 19.74
Henschel Seidl Grimm Söllner Kühling Seibert Legner Haas	3	0.3592152	0.12	19.86
Herzog Henschel Böckenförde Seidl Grimm Dieterich Kühling Seibert Herzog Henschel Der Richter Seidl ist durch Krankheit an der Unter- schrift gehindert. Her- zog Grimm Söllner Dieterich Kühling Seibert	1 1	0.1189061 0.1189061	0.12 0.12	19.98 20.10
Herzog Henschel Seidl Grimm Dieterich Kühling Seibert	1	0.1189061	0.12	20.21
Herzog Henschel Seidl Grimm Söllner Dieterich Kühling Seibert	2	0.2378121	2.73	22.95

(continued)

richter	N	exactpercent	roundedpercent	cumulpercent
Herzog Henschel Seidl Grimm Söllner Dietrich Seibert	2	0.2378121	0.24	23.19
Herzog Henschel Seidl Grimm Söllner Kühnig Seibert	1	0.1189061	0.12	23.31
Herzog Henschel Seidl Grimm Söllner Kühnig Seibert Jaeger	1	0.1189061	0.12	23.42
Jaeger Frau Haas ist gehindert, ihre Hömig Steiner Hohmann-Dennhardt Hoffmann-Riem Bryde Lübbe-Wolff Unterschrift beizufügen.  Jaeger	1	0.1189061	0.12	23.54
Jaeger Haas Hömig Steiner Hohmann-Dennhardt Hoffmann-Riem Bryde	1	0.1189061	0.12	23.66
Jentsch Broß Osterloh Di Fabio Mellinghoff Lübbe-Wolff Gerhardt	1	0.1189061	0.12	23.78
Kirchhof Eichberger Masing Paulus Baer Britz	2	0.2378121	0.24	24.02
Kirchhof Eichberger Masing Paulus Baer Britz Christ	3	0.3567182	0.36	24.97
Kirchhof Eichberger Schluckebier Masing Paulus Baer Britz	3	0.3567182	0.36	25.33
Kirchhof Eichberger Schluckebier Masing Paulus Baer Britz Ott	5	0.5945333	0.59	25.92
Kirchhof Gaier Eichberger Schluckebier Masing Paulus Baer Britz	6	0.7253769	0.72	33.17
Kirchhof Hohmann-Dennhardt Bryde Gaier Eichberger Schluckebier Masing	2	0.2378121	0.24	33.41
Kirchhof Hohmann-Dennhardt Bryde Gaier Eichberger Schluckebier Masing Paulus	16	1.9024970	1.90	35.32
Kirchhof Masing Paulus Baer Britz Ott Christ Schucke	1	0.1189061	0.12	35.43
Kruis	1	0.1189061	0.12	35.55
König Hermanns Müller Kessal-Wulf Maidowski Langenfeld Wallrabenstein	3	0.3567182	0.36	35.91
König Huber Hermanns Kessal-Wulf Maidowski Langenfeld Wallrabenstein	2	0.2378121	0.24	36.15
König Huber Hermanns Müller Kessal-Wulf Maidowski Langenfeld	4	0.4756243	0.48	36.62
König Huber Hermanns Müller Kessal-Wulf Maidowski Langenfeld Wallrabenstein	29	3.4482759	3.45	40.07

(continued)

richter	N	exactpercent	roundedpercent	cumulpercent
König Huber Hermanns Müller Maidowski Langenfeld Härtel	1	0.1189061	0.12	40.19
König Huber Hermanns Müller Maidowski Langenfeld Wallrabenstein	1	0.1756243	0.18	40.67
König Huber Müller Kessal-Wulf Maidowski Langenfeld	2	0.2378121	0.24	40.90
Kühling Jaeger Haas Hömig Steiner Hohmann Dennhardt Hoffmann-Riem	1	0.1189061	0.12	41.02
Kühling Jaeger Haas Hömig Steiner Hohmann Dennhardt Hoffmann-Riem Di Fabio	1	0.1189061	0.12	41.14
Landau Huber Hermanns Müller Kessal-Wulf König Maidowski	1	0.1189061	0.12	41.26
Limbach Böckenförde Graßhof Kruis Kirchhof Sommer	1	0.1189061	0.12	41.38
Limbach Böckenförde Klein Graßhof Kruis Winter Sommer	3	0.3567182	0.36	41.74
Limbach Böckenförde Klein Graßhof Kruis Winter Seibert Sommer	1	0.1189061	0.12	41.85
Limbach Graßhof Kruis Kirchhof Winter Sommer Hassemer	1	0.1189061	0.12	41.97
Limbach Graßhof Kruis Kirchhof Winter Sommer Jentsch	1	0.1189061	0.12	42.09
Limbach Graßhof Kruis Kirchhof Winter Sommer Jentsch Hassemer	4	0.6646849	1.66	43.76
Limbach Hassemer Broß	1	0.1189061	0.12	43.88
Limbach Kirchhof Graßhof	1	0.1189061	0.12	44.00
Limbach Kirchhof Jentsch Hassemer Broß	1	0.1189061	0.12	44.11
Limbach Kirchhof Sommer Jentsch Hassemer Broß Osterloh	3	0.3567182	0.36	44.47
Limbach Kirchhof Winter Sommer Jentsch Broß Osterloh	1	0.1189061	0.12	44.59
Limbach Kirchhof Winter Sommer Jentsch Hassemer Broß Osterloh	1	0.1189061	0.12	44.71
Limbach Kirchhof Winter Sommer Jentsch Hassemer Broß	1	0.1756243	0.48	45.18
Limbach Kirchhof Winter Sommer Jentsch Hassemer Broß Osterloh	1	0.13079667	1.31	46.49
Limbach Klein Graßhof Kruis Kirchhof Winter	2	0.2378121	0.24	46.73
Limbach Sommer Hassemer Broß Osterloh Di Fabio	1	0.1189061	0.12	46.85
Limbach Sommer Jentsch Hassemer Broß Osterloh	1	0.1189061	0.12	46.97
Limbach Sommer Jentsch Hassemer Broß Osterloh Di Fabio	4	0.5457783	1.55	48.51

(continued)

richter	N	exactpercent	roundedpercent	cumulpercent
Limbach Sommer Jentsch Hassemer Broß Osterloh Di Fabio Mellinghoff	20	0.3781213	2.38	50.89
Limbach Sommer Jentsch Hassemer Herr Broß ist an der  Osterloh Di Fabio  Unterschrift gehindert.   Limbach	1	0.1189061	0.12	51.01
Limbach Winter Sommer Jentsch Hassemer Brühl Osterloh	9	0.189061	0.12	51.13
Lübbe- Wolff Gerhardt Landau Huber Hermanns Müller Kessal- Wulf	4	0.4756243	0.48	51.61
Mahrenholz Böckenförde Klein Graf Kunze Stroh Winter Sommer	2	0.2378121	0.24	51.84
Masing Paulus Baer Britz Ott Chris Radtke	3	0.3567182	0.36	52.20
Papier  Grimm Kühling Seibert Jaeger Haas Hömig Steiner  Frau BVRin Seibert   ist aus dem Amt   ausgeschieden.	1	0.1189061	0.12	52.32
Papier Der Richter Grimm ist aus dem Kühling Jaeger Haas Hömig Steiner Hohmann- Dennhardt  Amt ausgeschie- den und deshalb an der Unterschrift  gehindert.   Papier	2	0.2378121	0.24	52.56
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	0.1189061	0.12	52.68
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	0.1189061	0.12	52.79

(continued)

richter	N	exactpercent	roundedpercent	cumulpercent
Papier Der Richter Steine ist Hohmann- Dennhardt Hoffmann- Riem Bryde Gaier Eichberger Schluckebier  ausgeschieden und daher der Unterschrift gehindert.   Papier	1	0.1189061	0.12	52.91
Papier Die Richterin Haas ist aus dem Steiner Hohmann- Dennhardt Hoffmann- Riem Bryde Gaier Eichberger  Amt ausgeschieden und des- halb an der Unterschrift  gehindert.   Papier	3	0.3567182	0.36	53.27
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	0.1189061	0.12	53.39
Papier Die Richterin Jaeger ist aus Haas Hömig Steiner Hohmann- Dennhardt Hoffmann-Riem  dem Amt ausgeschieden und daher an der Unterschrift  gehindert   Papier	1	0.1189061	0.12	53.51
Papier Die Richterin Jaeger ist aus Haas Hömig Steiner Hohmann- Dennhardt Hoffmann-Riem  dem Amt ausgeschieden und daher an der Unterschrift  gehindert.   Papier	1	0.1189061	0.12	53.63
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	0.1189061	0.12	53.75

(continued)

richter	N	exactpercent	roundedpercent	cumulpercent
Papier Die Richter Haas Hömig Steiner Hohmann- Dennhardt Hoffmann- Riem Bryde  Jaeger ist aus dem Amt ausgeschieden und deshalb an der  Unterschrift gehindert.   Papier	1	0.1189061	0.12	53.86
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	0.1189061	0.12	53.98
Papier Graßhof Grimm Kühling Frah- BVR in Seibert ist aus dem Amt ausgeschieden. Pa- pier Jaeger Haas Hömig	1	0.1189061	0.12	54.10
Papier Grimm Kühling Jaeger Haas Hömig Steiner Hohmann- Dennhardt	1	0.1189061	1.19	55.29
Papier Grimm Kühling Jaeger Haas Hömig Steiner Hohmann- Dennhardt	1	0.1189061	1.19	56.48
Papier Grimm Kühling Jaeger Hömig Steiner Hohmann- Dennhardt	1	0.1189061	0.12	56.60
Papier Grimm Kühling Seibert Jaeger Haas Hömig Steiner	6	0.713436	0.71	57.31
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	0.1189061	0.12	57.43
Papier Haas Hömig Steiner Hohmann- Dennhardt Hoffmann- Riem Bryde Gaier	2	2.6159334	2.62	60.05
Papier Haas Steiner Hohmann- Dennhardt Hoffmann- Riem Bryde Gaier Eichberger	3	0.3567182	0.36	60.40

(continued)

richter	N	exactpercent	roundedpercent	cumulpercent
Papier Hassemer Sommer Jaeger Haas Hömig Steiner Jentsch Broß Oslerloh Hohn Dennhardt Hoffmann-Riem Di Fabio Bryde Mellinghoff Lübbe-Wolff		0.1189061	0.12	60.52
Papier Hohmann-Dennhardt Bryde Gaier Eichberger Schluckebier Kirchhof Masing	21	2.4970273	2.50	63.02
Papier Hohmann-Dennhardt Der Richter Hoffmann-Riemist Bryde Gaier Eichberger Schluckebier Kirchhof aus dem Amt ausgeschieden und daher an der Unterschrift gehindert.   Papier	1	0.1189061	0.12	63.14
Papier Hohmann-Dennhardt Hoffmann-Riem Bryde Gaier Eichberger Schluckebier	1	0.1189061	0.12	63.26
Papier Hohmann-Dennhardt Hoffmann-Riem Bryde Gaier Eichberger Schluckebier Kirchhof	7	0.8323424	0.83	64.09
Papier Jaeger Haas Hömig Steiner Hohmann-Dennhardt Hoffmann-Riem	1	0.1189061	0.12	64.21
Papier Jaeger Haas Hömig Steiner Hohmann-Dennhardt Hoffmann-Riem Bryde	16	6.6587396	6.66	70.87
Papier Jaeger Hömig Steiner Hohmann-Dennhardt Hoffmann-Riem		0.1189061	0.12	70.99
Papier Jaeger Hömig Steiner Hohmann-Dennhardt Hoffmann-Riem Bryde		0.1189061	0.12	71.11
Papier Kühling Jaeger Haas Hömig Steiner Hohmann-Dennhardt Hoffmann-Riem	7	0.8323424	0.83	71.94
Papier Kühling Jaeger Hömig Steinder Hohmann-Dennhardt	1	0.1189061	0.12	72.06
Papier Kühling Jaeger Hömig Steinder Hohmann-Dennhardt Hoffmann-Riem	1	0.1189061	0.12	72.18



(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 zu  unterschreiben.   Papier	1	0.1189061	0.12	72.29
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	0.1189061	0.12	72.41
Papier Steiner Hohmann-Dennhardt Hoffmann-Riem Bryde Gaier Eichberger Schluckebier	13	1.5457788	1.55	73.96
Paulus Britz Ott Christ Radtke Härtel	1	0.1189061	0.12	74.08
Präsidentin Limbach ist aus Sommer Jentsch Richter Broß ist an der Hassemer Osterloh Di Fabio Mellinghoff  dem Amt ausgeschieden.   Unterschrift gehindert.   Hassemer	1	0.1189061	0.12	74.20
Seidl Grimm Kühling Jaeger Haas Hömig Steiner	1	0.1189061	0.12	74.32
Seidl Grimm Kühling Seibert Jaeger Haas Hömig Steiner	7	0.1823494	0.83	75.15
Vizepräsident Henschel und Richter Söllner sind aus dem Amt ausgeschieden und deshalb an der Unterschrift gehindert. Seidl Seidl Grimm Kühling Seibert Jaeger Haas	1	0.1189061	0.12	75.27
Vizepräsident Seidl  Grimm Kühling Seibert Jaeger Haas Hömig Steiner  ist aus dem Amt   ausgeschieden und deshalb   an der Unterschrift   gehindert.   Grimm	2	0.2378121	0.24	75.51
Voßkuhle Broß Osterloh Di Fabio Mellinghoff Lübbe-Wolff Gerhardt Landau	35	4.1617122	4.16	79.67

(continued)

richter	N	exactpercent	roundedpercent	cumulpercent
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	0.1189061	0.12	79.79
Voßkuhle Di Fabio Lübbe-Wolff Gerhardt Landau Huber Hermanns	3	0.3567182	0.36	80.14
Voßkuhle Di Fabio Mellinghoff Lübbe-Wolff Gerhardt Landau	1	0.1189061	0.12	80.26
Voßkuhle Di Fabio Mellinghoff Lübbe-Wolff Gerhardt Landau Huber Hermanns	12	1.4268728	1.43	81.69
Voßkuhle Die Richterin Lübbe- Wolff ist aus dem Amt  Der Richter Gerhardt ist aus dem Amt  Landau Huber Hermanns Müller Kessal-Wulf  ausgeschieden und deshalb an der Unter- schrift gehindert   Voßkuhle	1	0.1189061	0.12	81.81
Voßkuhle Gerhardt Landau Huber Hermanns Müller Kessal-Wulf König	1	0.1189061	0.12	81.93
Voßkuhle Huber Hermanns Kessal-Wulf König Maidowski	1	0.1189061	0.12	82.05
Voßkuhle Huber Hermanns Kessal-Wulf König Maidowski Langenfeld	2	0.2378121	0.24	82.28
Voßkuhle Huber Hermanns Müller Kessal-Wulf König	1	0.1189061	0.12	82.40
Voßkuhle Huber Hermanns Müller Kessal-Wulf König Maidowski	1	0.1189061	1.19	83.59
Voßkuhle Huber Hermanns Müller Kessal-Wulf König Maidowski Langenfeld	1	0.1189061	7.02	90.61
Voßkuhle Huber Müller	1	0.1189061	0.12	90.73

(continued)

richter	N	exactpercent	roundedpercent	cumulpercent
Voßkuhle Kirchhof Lübbe- Wolff Gerhardt Gaier Eichberger Schluckebier Masing Paulus Huber Hermanns Baer Britz Müller Kessal- Wulf	1	0.1189061	0.12	90.84
Voßkuhle Landau Hermanns Müller Kessal- Wulf König Maidowski	1	0.1189061	0.12	90.96
Voßkuhle Landau Huber Hermanns Kessal- Wulf König Maidowski	2	0.2378121	0.24	91.20
Voßkuhle Landau Huber Hermanns Müller Kessal- Wulf	2	0.2378121	0.24	91.44
Voßkuhle Landau Huber Hermanns Müller Kessal- Wulf König	4	0.4756243	0.48	91.91
Voßkuhle Landau Huber Hermanns Müller Kessal- Wulf König Maidowski	8	0.9512485	2.85	94.77
Voßkuhle Landau Huber Müller Kessal- Wulf Maidowski	1	0.1189061	0.12	94.89
Voßkuhle Lübbe- Wolff Gerhardt Huber Hermanns Müller Kessal- Wulf	1	0.1189061	0.12	95.01
Voßkuhle Lübbe- Wolff Gerhardt Landau Huber Hermanns	4	0.4756243	0.48	95.48
Voßkuhle Lübbe- Wolff Gerhardt Landau Huber Hermanns Kessal- Wulf	1	0.1189061	0.12	95.60
Voßkuhle Lübbe- Wolff Gerhardt Landau Huber Hermanns Müller Kessal- Wulf	33	3.9239001	3.92	99.52
Voßkuhle Lübbe- Wolff Gerhardt Landau Huber Hermanns Müller Kessal- Wulff	2	0.2378121	0.24	99.76
Voßkuhle Masing Huber Hermanns Kessal- Wulf König Maidowski Langenfeld	1	0.1189061	0.12	99.88
Zeidler Rinck Wand Dr. Rottmann Dr. Dr. h. c. Niebler Steinberger Träger Mahrenholz H.- J. Rinck	1	0.1189061	0.12	100.00
Total	841	100.0000000	100.00	100.00

---

Frequency Table for Variable: entscheidung\_typ

---

2 unique value(s) detected.

entscheidung_typ	N	exactpercent	roundedpercent	cumulpercent
B	650	77.28894	77.29	77.29
U	191	22.71106	22.71	100.00
Total	841	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-"))
```

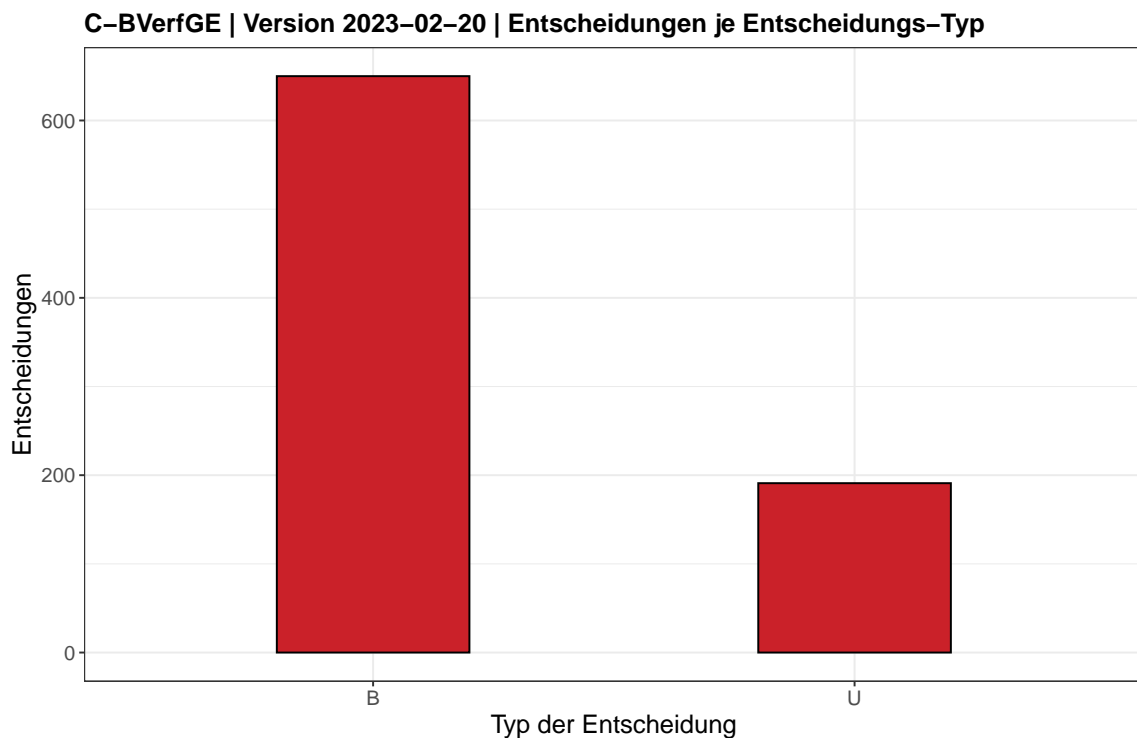
### 10.2 Tabellen einlesen

```
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)  
  )
```

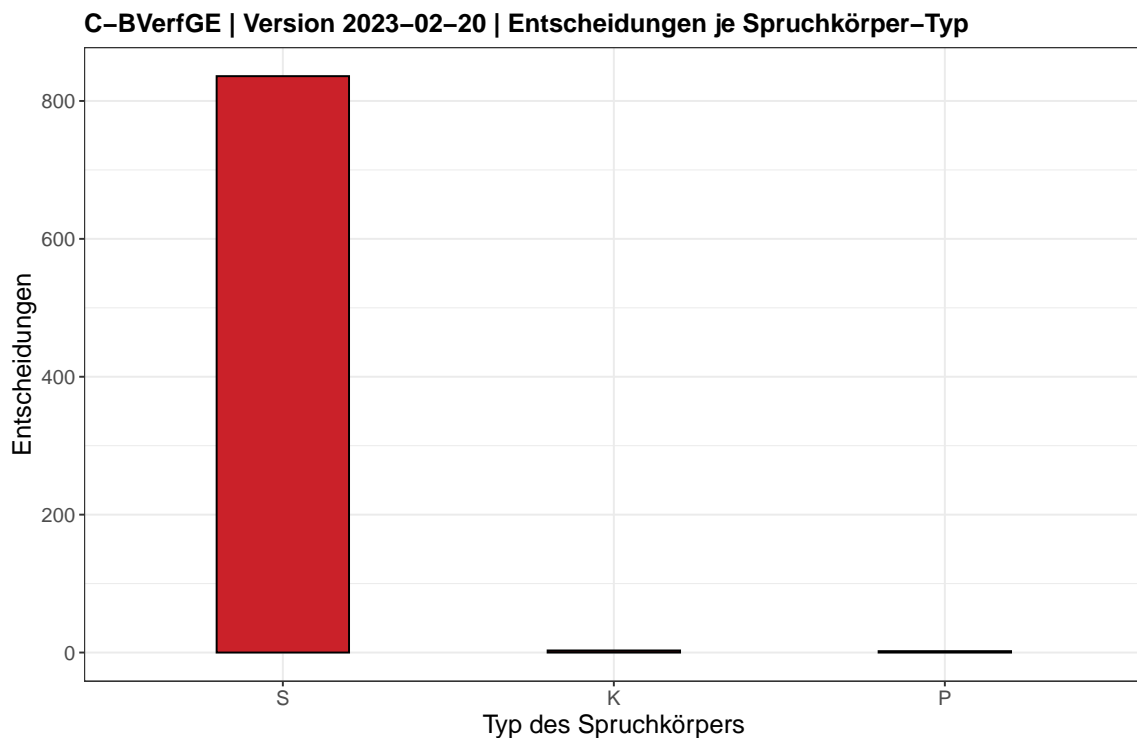


Fobbe | DOI: 10.5281/zenodo.7655163

## 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)  
  )
```

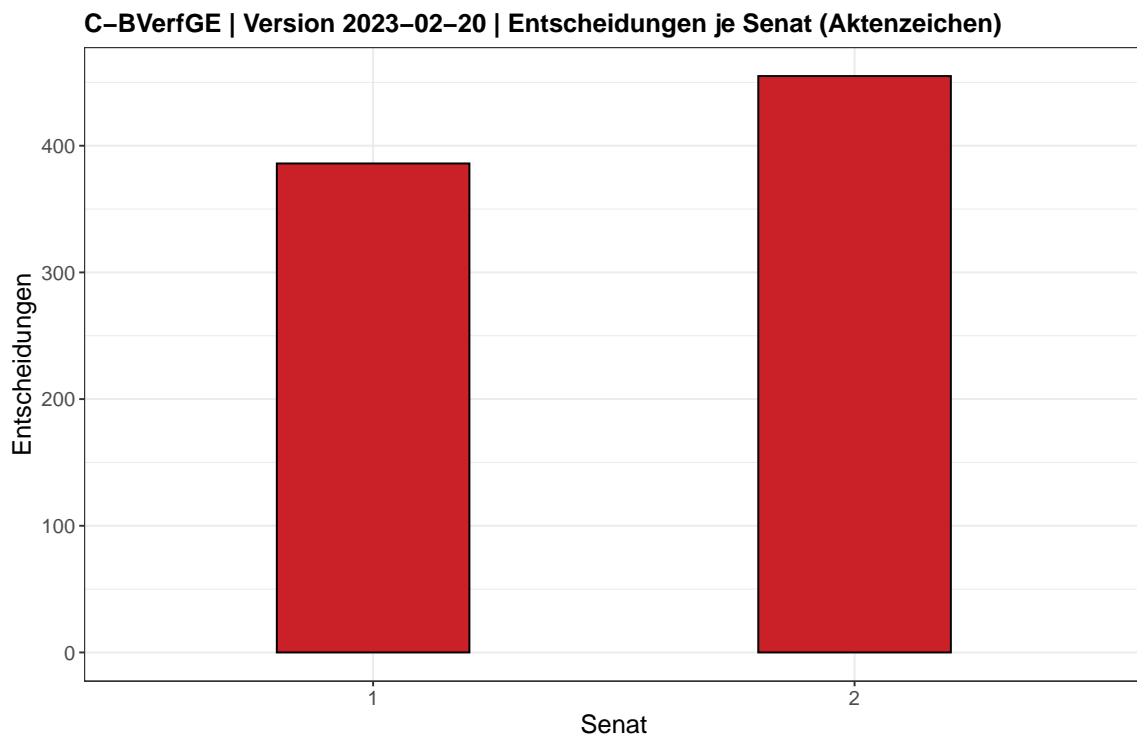


Fobbe | DOI: 10.5281/zenodo.7655163

## 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.7655163

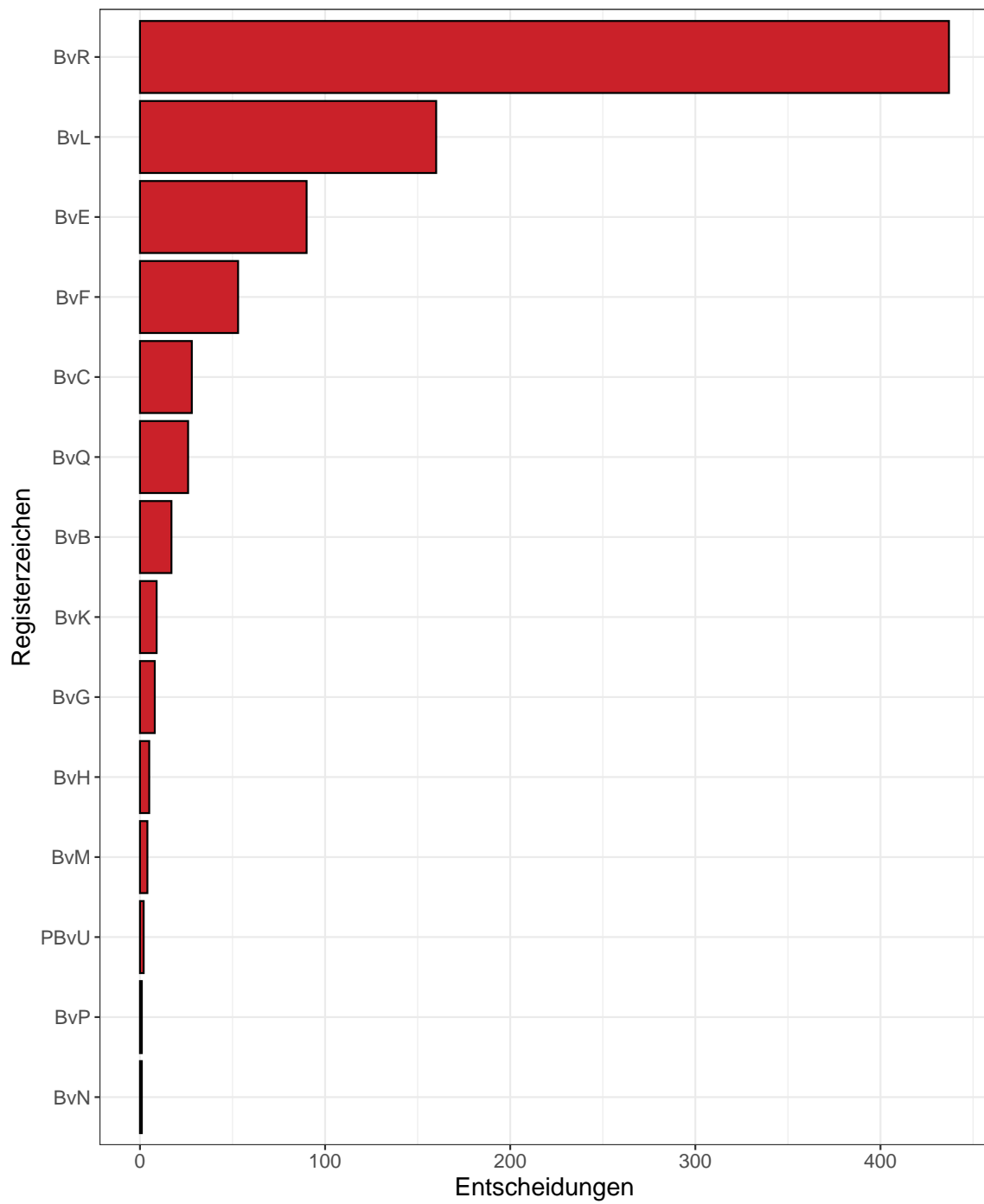


## 10.6 Diagramm: Registerzeichen

```
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 2023-02-20 | Entscheidungen je Registerzeichen

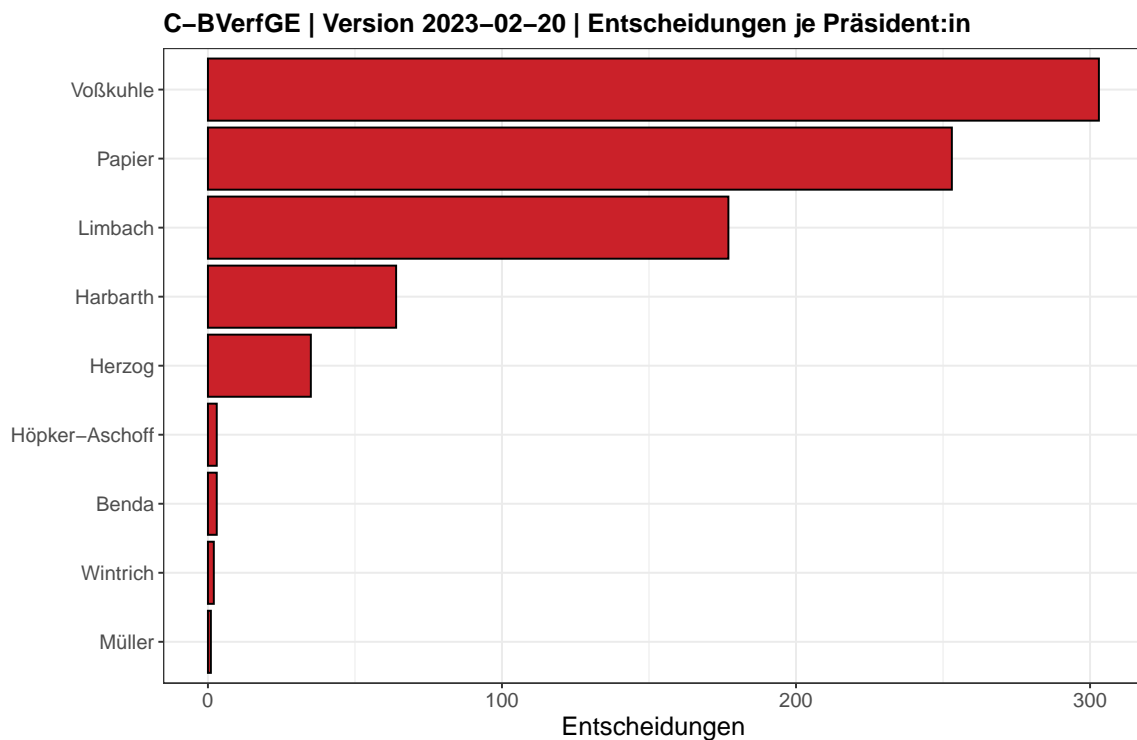


Fobbe | DOI: 10.5281/zenodo.7655163

## 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)  
  )
```

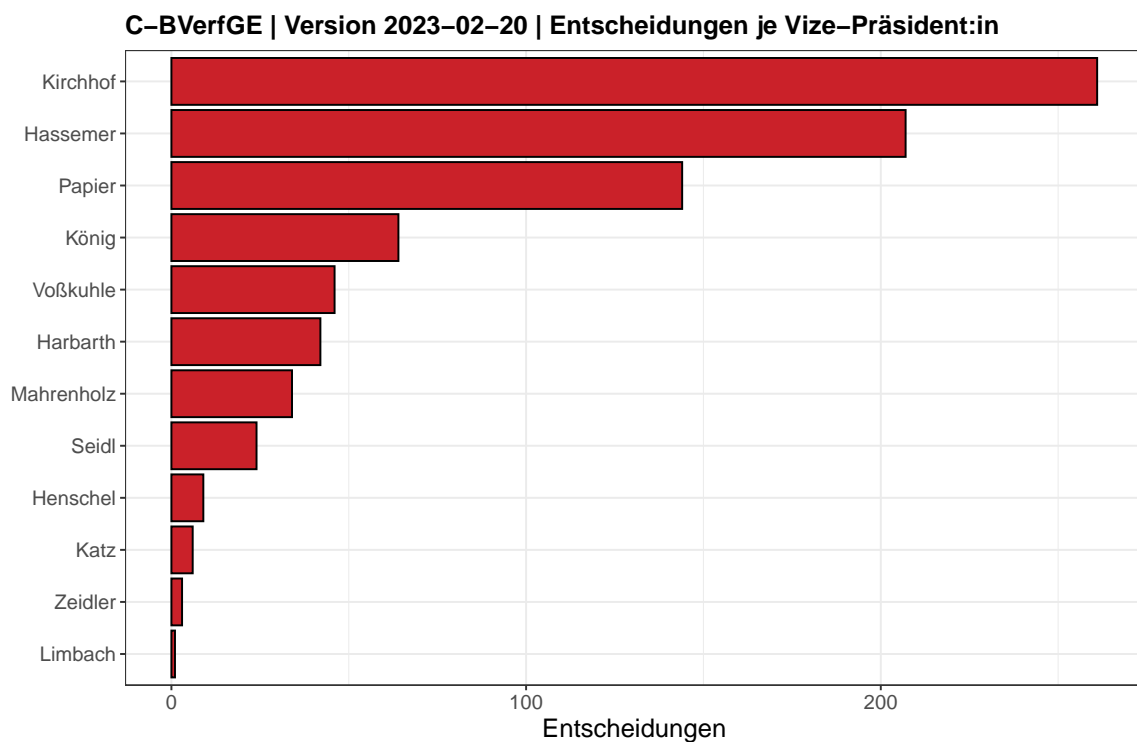


Fobbe | DOI: 10.5281/zenodo.7655163

## 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)
  )
```

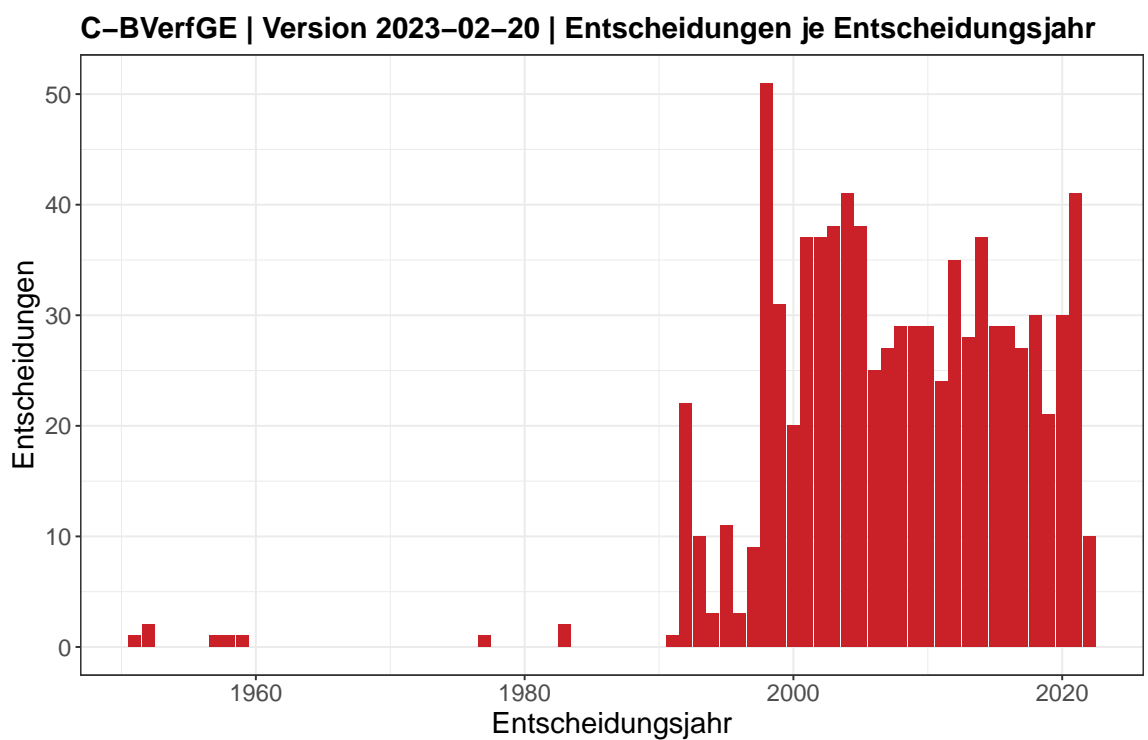


Fobbe | DOI: 10.5281/zenodo.7655163

## 10.9 Diagramm: Entscheidungsjahr

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

```
ggplot(data = frequetable) +  
  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)  
  )
```

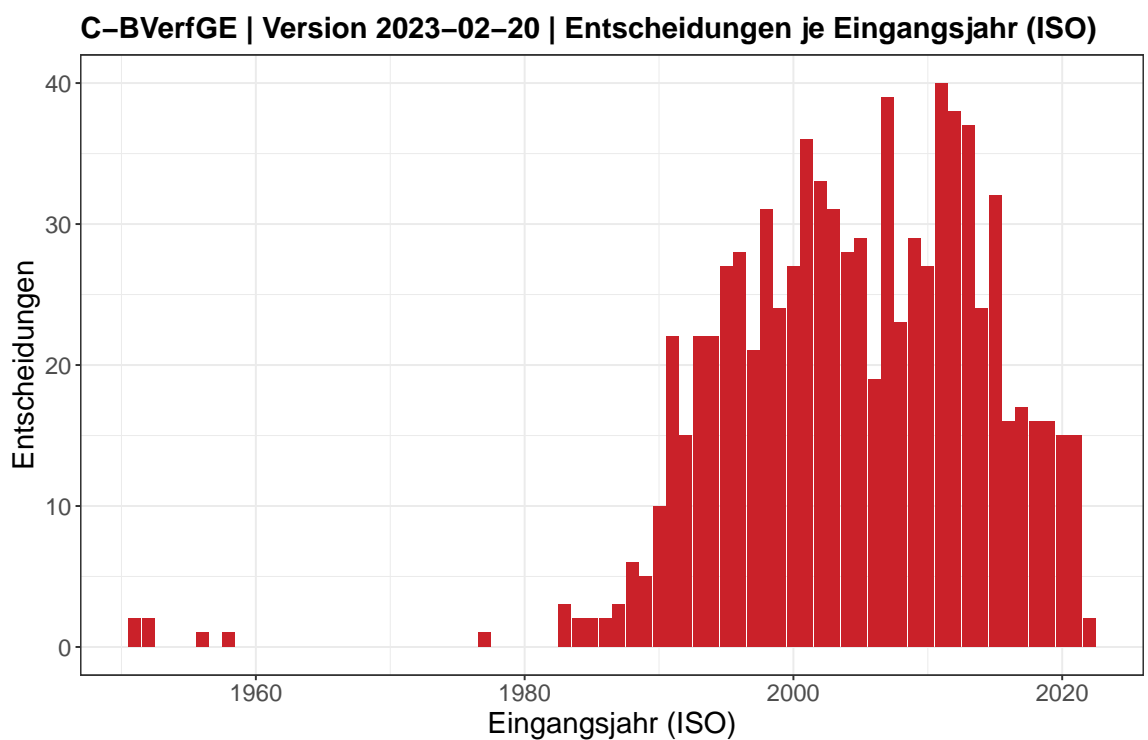


Fobbe | DOI: 10.5281/zenodo.7655163

## 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.7655163

## 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	60,401,611	1,827	29,293	57,421	71,821.1784	92,231	781,226
tokens	9,326,284	178	4,392	8,748	11,089.5172	14,378	115,540
typen	167,330	92	1,190	1,927	2,069.3115	2,654	13,491
saetze	503,364	8	251	478	598.5303	776	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.
## "1951-09-09" "2002-01-22" "2008-02-26" "2008-02-10" "2015-04-21" "2022-03-22"
```

### 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	39	1951	2002	2008	2007.6302	2015	2022
eingangsjahr_iso	45	1951	1998	2005	2004.5577	2012	2022
band	81	1	104	120	121.3876	139	160
eingangsnummer	420	1	4	142	698.8193	1327	3588

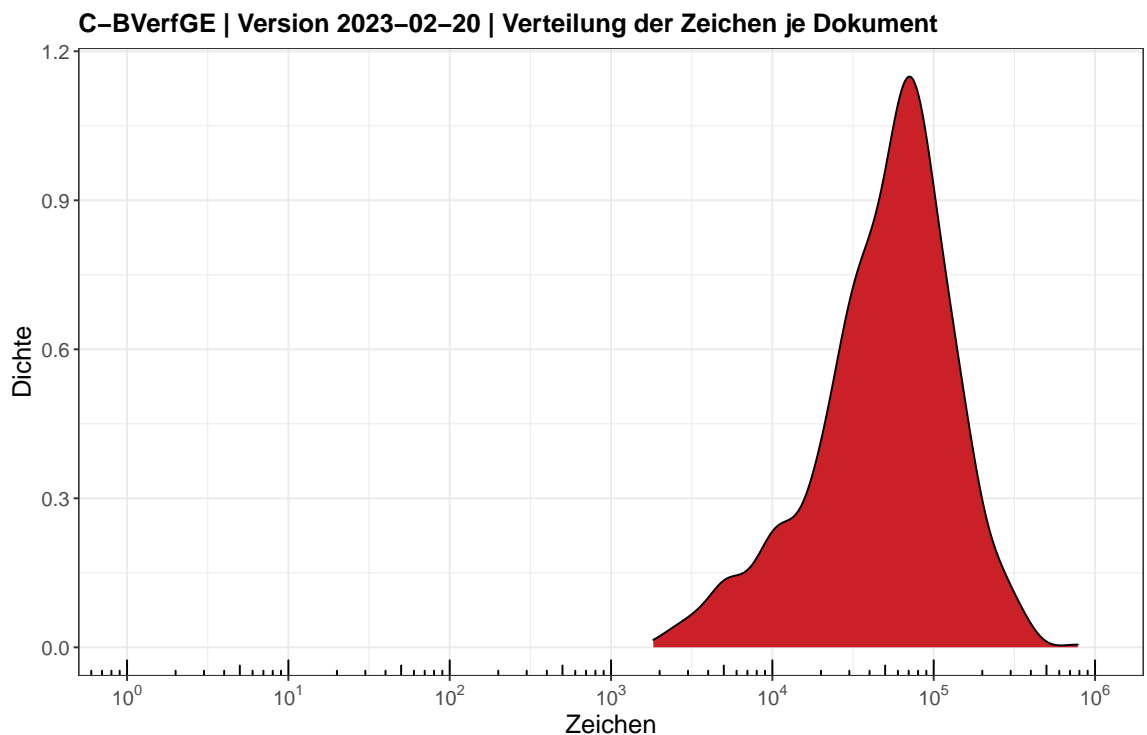
### 11.6.4 Zusammenfassungen speichern

```
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.7655163

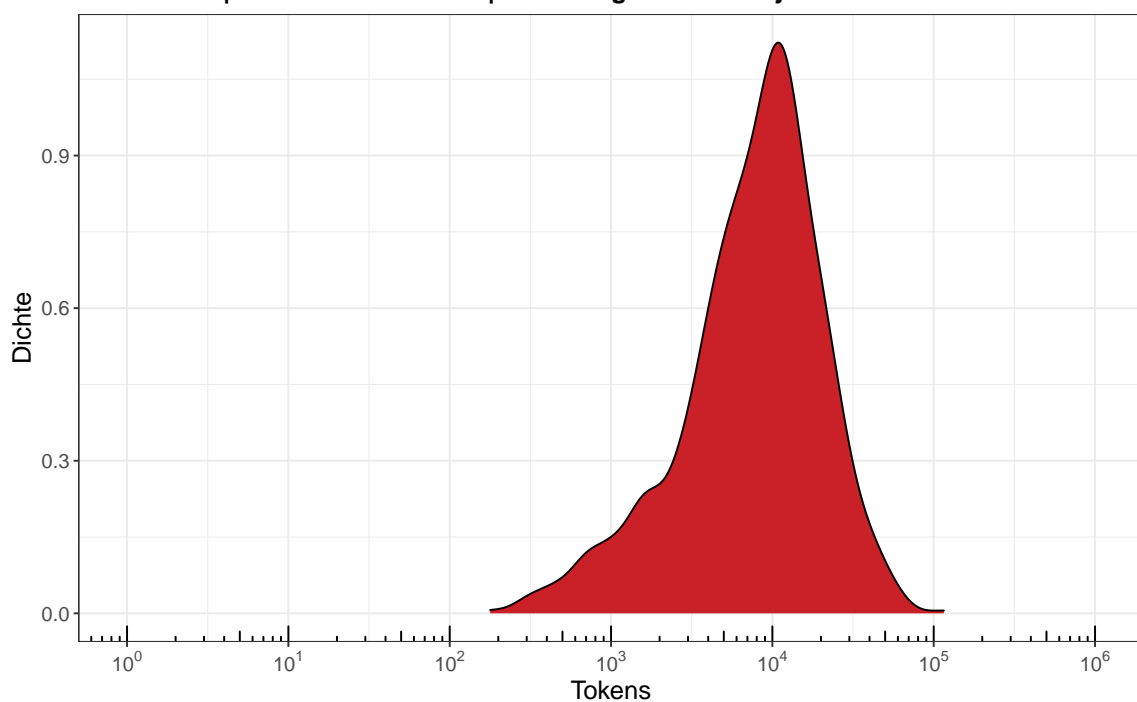
### 11.7.2 Diagramm: Verteilung Tokens

```

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)
  )

```

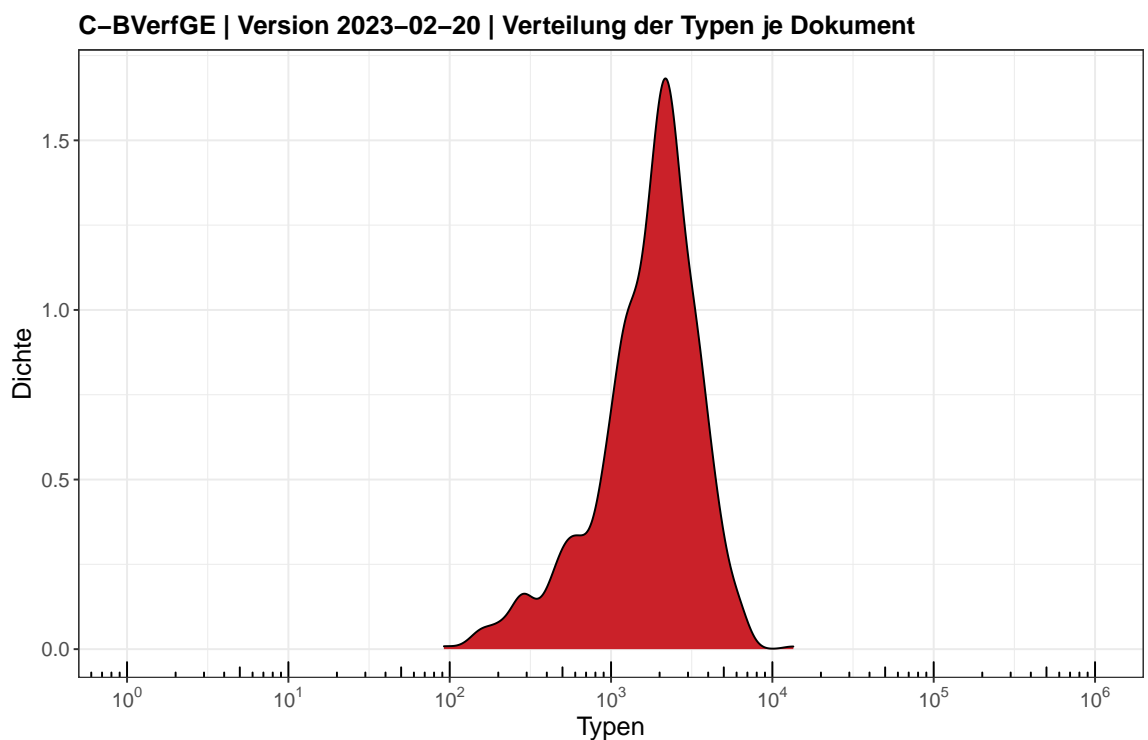
**C-BVerfGE | Version 2023-02-20 | Verteilung der Tokens je Dokument**



Fobbe | DOI: 10.5281/zenodo.7655163

### 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)
  )
```

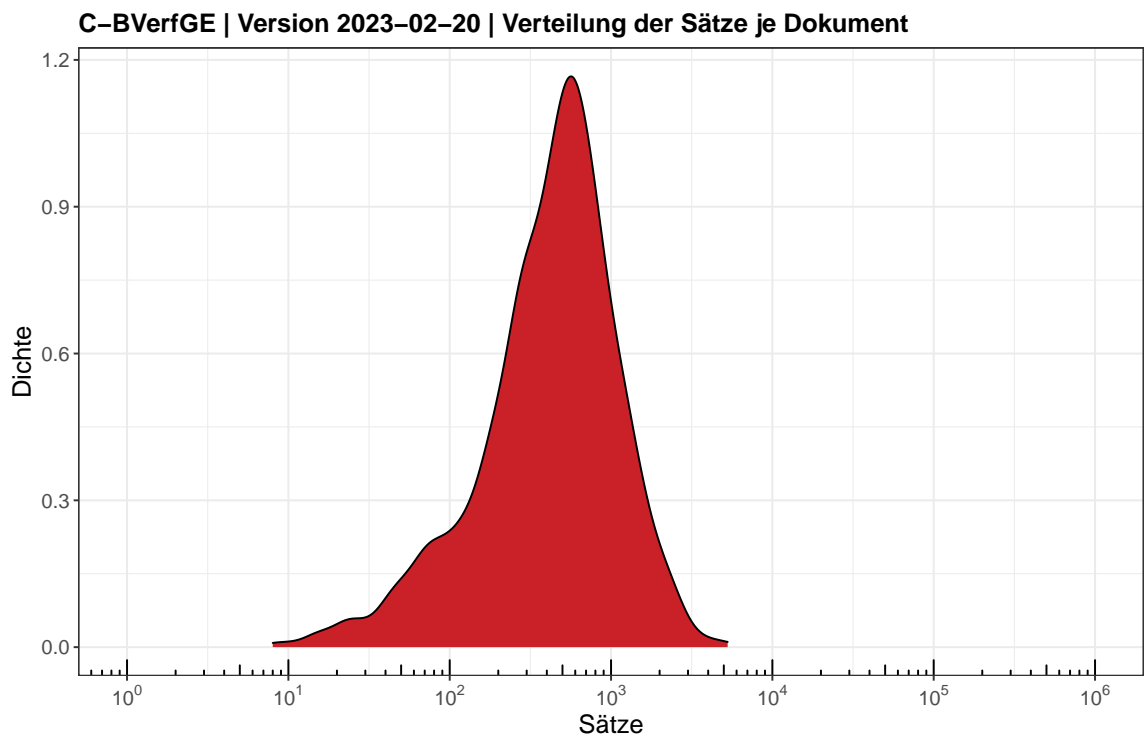


Fobbe | DOI: 10.5281/zenodo.7655163



### 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.7655163

## 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",
    "zitativvorschlag",
    "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")
```

```
## 62.4 MB
```

#### 15.1.2 CSV Korpus (MB)

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

```
## [1] 62.02327
```

#### 15.1.3 CSV Metadaten (MB)

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

```
## [1] 0.672536
```

#### 15.1.4 CSV Annotiert (MB)

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

```
## [1] 1321.641
```

#### 15.1.5 CSV Segmentiert (MB)

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

```
## [1] 120.2708
```

### 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] 145.1497
```

### 15.1.7 TXT-Dateien (MB)

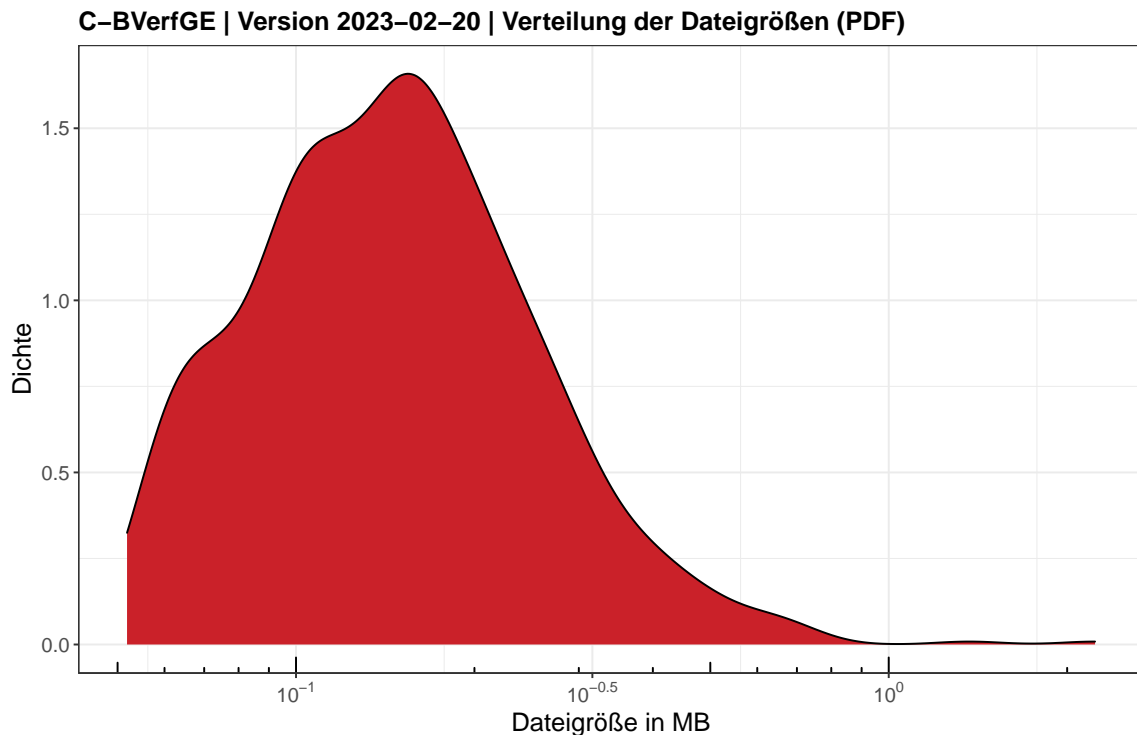
```
files.txt <- list.files(pattern = "\\..txt$",  
                        ignore.case = TRUE)  
  
txt.MB <- file.size(files.txt) / 10^6  
sum(txt.MB)
```

```
## [1] 62.29284
```

## 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)
  )
```

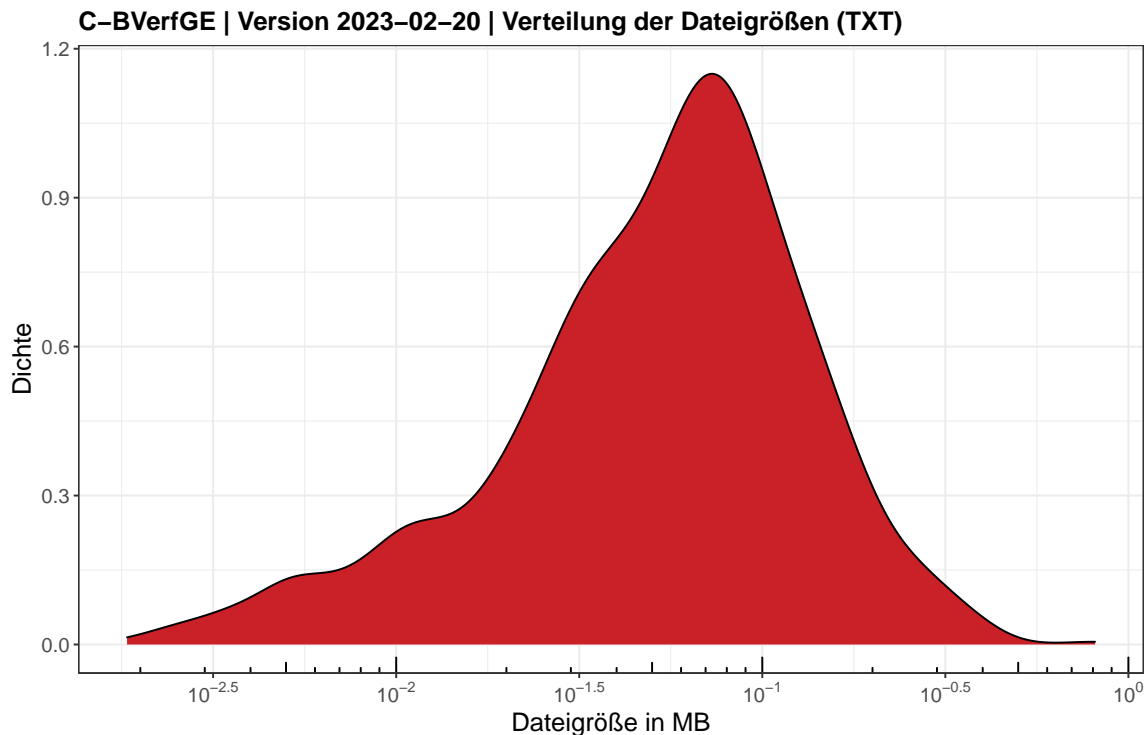


Fobbe | DOI: 10.5281/zenodo.7655163

### 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.7655163

## 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: 2023-02-20 01:14:39
```

```
## Processed 9 files. Runtime was 0.94 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_2023-02-20_DE_ANALYSE.zip
2	C-BVerfGE_2023-02-20_DE_CSV_Annotiert.zip
3	C-BVerfGE_2023-02-20_DE_CSV_Datensatz.zip
4	C-BVerfGE_2023-02-20_DE_CSV_Metadaten.zip
5	C-BVerfGE_2023-02-20_DE_CSV_Segmentiert.zip
6	C-BVerfGE_2023-02-20_DE_HTML_Datensatz.zip
7	C-BVerfGE_2023-02-20_DE_PDF_Datensatz.zip
8	C-BVerfGE_2023-02-20_DE_TXT_Datensatz.zip
9	C-BVerfGE_2023-02-20_Source_Files.zip

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

index	sha2.256
1	3958069570c5d3251a6b91daee85a5a5707180a7ca5ab06ed101e89b3d5b624c
2	475cf7f2a5e6b619b7dbfec2207ff23b5c80eabc523162be0f90774655f8d1e4
3	c8acfbcd8f3582fbc16703a9d01eeeb3f8475550f326a23994e3092747f05b2
4	3027800eddd9365a20cd1406bc02d55c732e38f8be69eb5cc58fb423d18b22c4
5	9f4b890d469c3cb446eca7538b98c12643ee46dd7aedf56c80b6a7f2a7eecfb8
6	3b5c6da5cea563865b97756585162aeecbeeb5f4e8f688c3d2f40616da4340543
7	2968d27a508f7ac6b3a09cece10484935e4712066e4729c5515e48d9499351f7
8	08b550b8dc1516e338b638683f457a7a42f1415788bd4c20c4ae74eda9151da0
9	4de82f6aa0445a5955b5d54a565b67265c43be96fce31f9c75104c5744dc3724

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

index	sha3.512
1	c20247a63dae63fafc17f82929014d062e4a146e428051bff7727de3c306fce6 1a0aab90faf3b8d1582850be46a48761d978fc9c23e4dcd81637cbc1b2ba652c
2	661c60b863b710214689ad439b6d6797f9f2b31de11a729339ff25c7a2baed2b 7dc38ee55f5ae088cb60d33f92290db66b4cd87575aac33f592d62d70e2bd121
3	384e5603fcc9b5c97888067864760e98cad2d2d27610b7958fe451b2b31593da d5fa6fa1154d7b2936b2757ecc1b858ffb3c627554b6d8967708a17211210b28
4	386b270fd76997ee7201e1ec0faf090e8b6e90d8ca0641f51541cd2751f4e00d e7ae2b65097358ff4034edd2a5663464b539c8f112219bef183e59a2dd0be947
5	67dba90ceda463381ec689b4b60a0b32497499d810ad7d851ebddf48e937f700 91afc69f4d617ba267d72b24342c7b644ead5540fccceea360f590705354a1aa

6	52c2c54c520b2047a68ba274563eb8af61b417493072689ada0a7d85fcb6dcb13c29e5c67a5f543c3cf06b05fc85c33424efe46f0c97a384db4e575d8bb85782
7	bf2c209f9c7c7ecd3e05e2c3555c19bd479ac18003baf97eb07ef0c697db7bdbf90fb915fe32153c9c968bce22ac4c8be9de0cdbcf175bba87a9a59f72a1729d
8	399c72d7a9a36a40486ab3be55d32a6a14ea0c4feb3560c5421b033189357d8573db7ed3adb7077c7518a052dffe5bd626255c62d184ac18d8a641a4c9a0f647
9	c7dcd7f8a12f73819f2856d1f3f9d14a7544af8ce5bfc54010661897e2b23f6e85d937ec910695677cf3555a33446a6a5554a8eefe6f1eb63c6edd2a1acb9017

---

## 18 Aufräumen

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

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

print(files.output)
```

```
## [1] "C-BVerfGE_2023-02-20_DE_ANALYSE.zip"
## [2] "C-BVerfGE_2023-02-20_DE_CSV_Annotiert.zip"
## [3] "C-BVerfGE_2023-02-20_DE_CSV_Datensatz.zip"
## [4] "C-BVerfGE_2023-02-20_DE_CSV_Metadaten.zip"
## [5] "C-BVerfGE_2023-02-20_DE_CSV_Segmentiert.zip"
## [6] "C-BVerfGE_2023-02-20_DE_HTML_Datensatz.zip"
## [7] "C-BVerfGE_2023-02-20_DE_PDF_Datensatz.zip"
## [8] "C-BVerfGE_2023-02-20_DE_TXT_Datensatz.zip"
## [9] "C-BVerfGE_2023-02-20_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] "2023-02-20"
```

### 19.2 Datum und Uhrzeit (Anfang)

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

```
## [1] "2023-02-20 00:09:30 CET"
```

### 19.3 Datum und Uhrzeit (Ende)

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

```
## [1] "2023-02-20 01:14:40 CET"
```

### 19.4 Laufzeit des gesamten Skriptes

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

```
## Time difference of 1.086202 hours
```

### 19.5 Warnungen

```
warnings()
```



## 20 Parameter für strenge Replikationen

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

```
## [1] "OpenSSL 1.1.1q  5 Jul 2022"
```

```
sessionInfo()
```

```
## R version 4.1.3 (2022-03-10)
## Platform: x86_64-redhat-linux-gnu (64-bit)
## Running under: Fedora Linux 36 (Workstation Edition)
##
## Matrix products: default
## BLAS/LAPACK: /usr/lib64/libflexiblas.so.3.3
##
## 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_4.1.3  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_4.1.3          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_0.16.0         xml2_1.3.3      ellipsis_0.3.2  stopwords_2.3
## [53] vctrs_0.3.8         fastmatch_1.1-3 tools_4.1.3     glue_1.6.0
## [57] parallel_4.1.3      fastmap_1.1.0   yaml_2.2.1      colorspace_2.0-2
```

## Literaturverzeichnis

- Bengtsson, Henrik. 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>.
- . 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>.
- . 2021c. *Future: Unified Parallel and Distributed Processing in r for Everyone*. <https://CRAN.R-project.org/package=future>.
- . 2021d. *Future.apply: Apply Function to Elements in Parallel Using Futures*. <https://CRAN.R-project.org/package=future.apply>.
- 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=pdfutils>.
- R Core Team. 2022. *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>.