

# Corpus der Entscheidungen des Bundesgerichtshofs (CE-BGH-Source)

COMPILATION REPORT

Version 2022-02-12

License MIT-0

DOI: 10.5281/zenodo.5910138

<b>Titel</b>	Source Code des »Corpus der Entscheidungen des Bundesgerichtshofs«
<b>Abkürzung</b>	CE-BGH-Source
<b>Autor</b>	Seán Fobbe
<b>Version</b>	2022-02-12
<b>Download</b>	<a href="https://doi.org/10.5281/zenodo.5910138">https://doi.org/10.5281/zenodo.5910138</a>
<b>Lizenz</b>	MIT No Attribution (MIT-0)

### Zitiervorschlag

*Seán Fobbe* (2022). Source Code des »Corpus der Entscheidungen des Bundesgerichtshofs« (CE-BGH-Source). Version 2022-02-12. Zenodo. DOI: 10.5281/zenodo.5910138.

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

Soweit nicht anders angegeben ist die DOI immer eine »Version DOI« und bezieht sich nur auf eine bestimmte Version der Software. Sie verlinkt daher nur Version 2022-02-12. Für das Gesamtkonzept der Software steht eine »Concept DOI« zur Verfügung, die auf der Zenodo-Seite jeder Version unter »Cite all versions?« zu finden ist. Die »Concept DOI« verlinkt immer die aktuellste Version.

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

Copyright — 2022 — Seán Fobbe

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

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

### Disclaimer

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

# Inhaltsverzeichnis

<b>1</b>	<b>Corpus der Entscheidungen des Bundesgerichtshofs (CE-BGH)</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 . . . . .	12
2.6	Weitere Verzeichnisse definieren . . . . .	12
2.7	Dateien aus vorherigen Runs bereinigen . . . . .	13
2.8	Verzeichnisse anlegen . . . . .	13
2.9	Vollzitate statistischer Software schreiben . . . . .	13
2.10	Allgemeine Konfiguration . . . . .	13
2.10.1	Konfiguration einlesen . . . . .	13
2.10.2	Konfiguration anzeigen . . . . .	13
2.10.3	Knitr Optionen setzen . . . . .	14
2.10.4	Download Timeout setzen . . . . .	14
2.10.5	Quellenangabe für Diagramme definieren . . . . .	15
2.10.6	Präfix für Dateien definieren . . . . .	15
2.10.7	Präfix für Diagramme definieren . . . . .	15
2.10.8	Quanteda-Optionen setzen . . . . .	15
2.11	LaTeX Konfiguration . . . . .	15
2.11.1	LaTeX Parameter definieren . . . . .	15
2.11.2	LaTeX Parameter schreiben . . . . .	16
2.12	Parallelisierung aktivieren . . . . .	17
2.12.1	Anzahl logischer Kerne festlegen . . . . .	17
2.12.2	Quanteda . . . . .	17
2.12.3	Data.table . . . . .	17
<b>3</b>	<b>Download: Weitere Datensätze</b>	<b>18</b>
3.1	Registerzeichen und Verfahrensarten . . . . .	18
3.2	Personendaten zu Präsident:innen . . . . .	18
3.3	Personendaten zu Vize-Präsident:innen . . . . .	18
<b>4</b>	<b>Links suchen</b>	<b>19</b>
4.1	Maximalen Such-Umfang einlesen . . . . .	19
4.2	Maximalen Such-Umfang anzeigen . . . . .	19
4.3	Funktion definieren . . . . .	19

4.4	Genauen Such-Umfang berechnen . . . . .	20
4.5	Locator einfügen . . . . .	20
4.6	[Debugging Modus] Reduzierung des Such-Umfangs . . . . .	20
4.7	Geschätzte Such-Dauer in Minuten . . . . .	20
4.8	Zeitstempel: Linksammlung Beginn . . . . .	21
4.9	Metadaten extrahieren . . . . .	21
4.10	Zeitstempel: Linksammlung Ende . . . . .	24
4.11	Dauer Linksammlung . . . . .	24
4.12	Zusammenfügen . . . . .	24
<b>5</b>	<b>Test-Reihe: Vollständigkeit der Auswertung</b>	<b>25</b>
5.1	Locator einfügen . . . . .	25
5.2	Theoretischer Fehlbetrag . . . . .	25
5.3	Seiten mit weniger als 30 Entscheidungen anzeigen . . . . .	25
5.4	Fehlbetrag durch Seiten mit weniger als 30 Entscheidungen . . . . .	26
5.5	Tatsächlicher Fehlbetrag . . . . .	26
5.5.1	Fehlbetrag der NICHT durch Seiten mit weniger als 30 Entscheidungen erklärbar ist . . . . .	26
5.5.2	Gegenüberstellung: Anzahl Jahre und Anzahl Seiten mit weniger als 30 Entscheidungen . . . . .	26
5.6	Vorhandensein aller Jahr/Seiten-Kombinationen . . . . .	27
<b>6</b>	<b>Bereinigung der Metadaten</b>	<b>28</b>
6.1	Datum bereinigen . . . . .	28
6.2	Aktenzeichen bereinigen . . . . .	28
6.2.1	Reguläre Substitutionen . . . . .	28
6.2.2	Einzelne Fehler bereinigen . . . . .	29
6.2.3	Variable “zusatz_az” einfügen . . . . .	30
6.2.4	Finale Korrekturen . . . . .	30
6.2.5	Strenge REGEX-Validierung des Aktenzeichens . . . . .	30
6.2.6	Ergebnis der REGEX-Validierung . . . . .	30
6.2.7	Skript stoppen falls REGEX-Validierung gescheitert . . . . .	31
6.2.8	Aktenzeichen-Vektor in Download Table einfügen . . . . .	31
6.3	Spruchkörper bereinigen . . . . .	31
6.3.1	Reguläre Substitutionen . . . . .	31
6.3.2	Alle Spruchkörper anzeigen . . . . .	32
6.3.3	Spruchkörper-Vektor in Download Table einfügen . . . . .	33
6.4	Bemerkungen bereinigen . . . . .	33
6.5	Variable “leitsatz” erstellen . . . . .	33
6.6	Variable “name” erstellen . . . . .	33
6.7	Variable “name_datei” erstellen . . . . .	34
6.8	Dateinamen erstellen . . . . .	34
6.9	Einzelkorrektur vornehmen . . . . .	34
6.10	KollisionsID einfügen . . . . .	34
6.10.1	Anzahl Duplikate . . . . .	34
6.10.2	Kollisions-IDs vergeben . . . . .	35
6.11	Zufällige Auswahl zur Prüfung anzeigen . . . . .	35
6.12	PDF-Endung anfügen . . . . .	36
6.13	Strenge REGEX-Validierung: Gesamter Dateiname . . . . .	36
6.14	Ergebnis der REGEX-Validierung . . . . .	37

6.15	Skript stoppen falls REGEX-Validierung gescheitert . . . . .	37
6.16	Vollen Dateinamen in Download Table einfügen . . . . .	37
<b>7</b>	<b>Download der Entscheidungen im PDF-Format</b>	<b>38</b>
7.1	[Debugging Modus] Reduzierung des Download-Umfangs . . . . .	38
7.2	Zeitstempel: Download Beginn . . . . .	38
7.3	Download durchführen . . . . .	38
7.4	Zeitstempel: Download Ende . . . . .	38
7.5	Dauer: Download . . . . .	39
7.6	[Debugging Modus] Löschen zufälliger Dateien . . . . .	39
7.7	Download: Zwischenergebnis . . . . .	39
7.7.1	Anzahl herunterzuladender Dateien . . . . .	39
7.7.2	Anzahl heruntergeladener Dateien . . . . .	39
7.7.3	Fehlbetrag . . . . .	39
7.7.4	Fehlende Dateien . . . . .	40
7.8	Wiederholungsversuch: Download . . . . .	40
7.9	Download: Gesamtergebnis . . . . .	40
7.9.1	Anzahl herunterzuladender Dateien . . . . .	40
7.9.2	Anzahl heruntergeladener Dateien . . . . .	41
7.9.3	Fehlbetrag . . . . .	41
7.9.4	Fehlende Dateien . . . . .	41
<b>8</b>	<b>Text-Extraktion aus PDF</b>	<b>42</b>
8.1	Vektor der zu extrahierenden Dateien erstellen . . . . .	42
8.2	Anzahl zu extrahierender Dateien . . . . .	42
8.3	PDF extrahieren: Funktion anzeigen . . . . .	42
8.4	Text Extrahieren . . . . .	43
8.5	TXT-Dateien Einlesen . . . . .	43
8.6	In Data Table umwandeln . . . . .	44
8.7	Durch Zeilenumbruch getrennte Wörter zusammenfügen . . . . .	44
8.7.1	Funktion anzeigen . . . . .	44
8.7.2	Funktion ausführen . . . . .	45
8.8	Variable "datum" als Datentyp "IDate" kennzeichnen . . . . .	45
8.9	Variable "entscheidungsjahr" hinzufügen . . . . .	45
8.10	Variable "eingangsjahr_iso" hinzufügen . . . . .	45
8.11	Datensatz nach Datum sortieren . . . . .	45
8.12	Variable "praesi" hinzufügen . . . . .	45
8.12.1	Personaldaten einlesen . . . . .	45
8.12.2	Personaldaten anzeigen . . . . .	46
8.12.3	Hypothetisches Amtsende für Präsident:in . . . . .	46
8.12.4	Schleife vorbereiten . . . . .	46
8.12.5	Vektor erstellen . . . . .	46
8.12.6	Vektor einfügen . . . . .	47
8.13	Variable "v_praesi" hinzufügen . . . . .	47
8.13.1	Personaldaten einlesen . . . . .	47
8.13.2	Personaldaten anzeigen . . . . .	47
8.13.3	Hypothetisches Amtsende für Vize-Präsident:in . . . . .	48
8.13.4	Schleife vorbereiten . . . . .	48
8.13.5	Vektor erstellen . . . . .	48
8.13.6	Vektor einfügen . . . . .	49

8.14	Variable “verfahrensart” hinzufügen . . . . .	49
8.14.1	Datensatz einlesen . . . . .	49
8.14.2	Datensatz auf relevante Daten reduzieren . . . . .	49
8.14.3	Indizes bestimmen . . . . .	49
8.14.4	Vektor der Verfahrensarten erstellen und einfügen . . . . .	49
8.15	Variable “aktenzeichen” hinzufügen . . . . .	49
8.16	Variable “entscheidung_typ” hinzufügen . . . . .	50
8.16.1	Entscheidungen Parsen . . . . .	50
8.16.2	Indizes bestimmen . . . . .	50
8.16.3	Leeren Vektor erstellen . . . . .	50
8.16.4	Typen bei Indizes platzieren . . . . .	50
8.16.5	Typen auf Kurzform reduzieren . . . . .	50
8.16.6	Vektor in Datensatz einfügen . . . . .	51
8.17	Variable “ecli” hinzufügen . . . . .	51
8.17.1	Formatieren der Registerzeichen für ECLI . . . . .	51
8.17.2	Erstellen der ECLI-Ordinalzahl . . . . .	51
8.17.3	Vollständige ECLI erstellen . . . . .	52
8.17.4	Zufällige ECLI-Beispiele zur manuellen Nachprüfung . . . . .	52
8.18	Variable “bemerkung” hinzufügen . . . . .	52
8.19	Variable “berichtigung” hinzufügen . . . . .	53
8.20	Variable “leitsatz” hinzufügen . . . . .	53
8.21	Variable “name” hinzufügen . . . . .	53
8.22	Variable “doi_concept” hinzufügen . . . . .	53
8.23	Variable “doi_version” hinzufügen . . . . .	53
8.24	Variable “version” hinzufügen . . . . .	53
8.25	Variable “lizenz” hinzufügen . . . . .	54
8.26	Entfernen von Dokumenten ohne Typ/Name/Berichtigung . . . . .	54
8.26.1	Platzhalter-Dokumente definieren . . . . .	54
8.26.2	Falsch-positive Dokumente behalten . . . . .	54
8.26.3	Einzelkorrektur . . . . .	54
8.26.4	Dokumente ohne Typ, Name und Berichtigung anzeigen . . . . .	55
8.26.5	PDF-Namen definieren . . . . .	60
8.26.6	Platzhalter PDF/TXT speichern . . . . .	60
8.26.7	Platzhalter aus Datensatz entfernen . . . . .	60
<b>9</b>	<b>Frequenztabellen erstellen</b>	<b>62</b>
9.1	Funktion anzeigen . . . . .	62
9.2	Ignorierte Variablen . . . . .	63
9.3	Liste zu prüfender Variablen . . . . .	63
9.4	Präfix definieren . . . . .	64
9.5	Frequenztabellen berechnen . . . . .	64
<b>10</b>	<b>Frequenztabellen visualisieren</b>	<b>83</b>
10.1	Präfix erstellen . . . . .	83
10.2	Tabellen einlesen . . . . .	83
10.3	Diagramm: Typ der Entscheidung . . . . .	84
10.4	Diagramm: Spruchkörper nach Datenbank . . . . .	85
10.5	Diagramm: Spruchkörper nach Aktenzeichen . . . . .	87
10.6	Diagramm: Registerzeichen . . . . .	88
10.7	Diagramm: Präsident:in . . . . .	90

10.8	Diagramm: Vize-Präsident:in . . . . .	91
10.9	Diagramm: Entscheidungsjahr . . . . .	92
10.10	Diagramm: Eingangsjahr (ISO) . . . . .	93
<b>11</b>	<b>Korpus-Analytik</b>	<b>94</b>
11.1	Berechnung linguistischer Kennwerte . . . . .	94
11.1.1	Funktion anzeigen . . . . .	94
11.1.2	Berechnung durchführen . . . . .	95
11.1.3	Variablen-Namen anpassen . . . . .	95
11.2	Kennwerte dem Korpus hinzufügen . . . . .	96
11.3	Variante mit Metadaten erstellen . . . . .	96
11.4	Linguistische Kennwerte . . . . .	96
11.4.1	Zusammenfassungen berechnen . . . . .	96
11.4.2	Zusammenfassungen anzeigen . . . . .	97
11.4.3	Zusammenfassungen speichern . . . . .	97
11.5	Quantitative Variablen . . . . .	99
11.5.1	Entscheidungsdatum . . . . .	99
11.5.2	Zusammenfassungen berechnen . . . . .	99
11.5.3	Zusammenfassungen anzeigen . . . . .	100
11.5.4	Zusammenfassungen speichern . . . . .	100
11.6	Verteilungen linguistischer Kennwerte . . . . .	101
11.6.1	Diagramm: Verteilung Zeichen . . . . .	101
11.6.2	Diagramm: Verteilung Tokens . . . . .	102
11.6.3	Diagramm: Verteilung Typen . . . . .	104
11.6.4	Diagramm: Verteilung Sätze . . . . .	106
<b>12</b>	<b>Kontrolle der Variablen</b>	<b>108</b>
12.1	Semantische Sortierung der Variablen . . . . .	108
12.1.1	Variablen sortieren: Hauptdatensatz . . . . .	108
12.1.2	Variablen sortieren: Metadaten . . . . .	109
12.2	Anzahl Variablen der Datensätze . . . . .	110
12.3	Alle Variablen-Namen der Datensätze . . . . .	110
<b>13</b>	<b>CSV-Dateien erstellen</b>	<b>111</b>
13.1	CSV mit vollem Datensatz speichern . . . . .	111
13.2	CSV mit Metadaten speichern . . . . .	111
<b>14</b>	<b>Dateigrößen analysieren</b>	<b>112</b>
14.1	Gesamtgröße . . . . .	112
14.1.1	Korpus-Objekt in RAM (MB) . . . . .	112
14.1.2	CSV Korpus (MB) . . . . .	112
14.1.3	CSV Metadaten (MB) . . . . .	112
14.1.4	PDF-Dateien (MB) . . . . .	112
14.1.5	TXT-Dateien (MB) . . . . .	113
14.2	Diagramm: Verteilung der Dateigrößen (PDF) . . . . .	114
14.3	Diagramm: Verteilung der Dateigrößen (TXT) . . . . .	115
<b>15</b>	<b>Erstellen der ZIP-Archive</b>	<b>116</b>
15.1	Verpacken der CSV-Dateien . . . . .	116
15.1.1	Vollständiger Datensatz . . . . .	116

15.1.2	Metadaten . . . . .	116
15.2	Verpacken der PDF-Dateien . . . . .	116
15.2.1	Nur Leitsatz-Entscheidungen . . . . .	116
15.2.2	Nur Benannte Entscheidungen . . . . .	116
15.2.3	Alle Entscheidungen . . . . .	117
15.3	Verpacken der TXT-Dateien . . . . .	117
15.4	Verpacken der Analyse-Dateien . . . . .	117
15.5	Verpacken der Source-Dateien . . . . .	117
<b>16</b>	<b>Kryptographische Hashes</b>	<b>119</b>
16.1	Liste der ZIP-Archive erstellen . . . . .	119
16.2	Funktion anzeigen: future_multihashes . . . . .	119
16.3	Hashes berechnen . . . . .	120
16.4	In Data Table umwandeln . . . . .	120
16.5	Index hinzufügen . . . . .	120
16.6	In Datei schreiben . . . . .	121
16.7	Leerzeichen hinzufügen um Zeilenumbruch zu ermöglichen . . . . .	121
16.8	In Bericht anzeigen . . . . .	121
<b>17</b>	<b>Aufräumen</b>	<b>124</b>
<b>18</b>	<b>Abschluss</b>	<b>125</b>
18.1	Datumsstempel . . . . .	125
18.2	Datum und Uhrzeit (Anfang) . . . . .	125
18.3	Datum und Uhrzeit (Ende) . . . . .	125
18.4	Laufzeit des gesamten Skriptes . . . . .	125
18.5	Warnungen . . . . .	125
<b>19</b>	<b>Parameter für strenge Replikationen</b>	<b>126</b>
	<b>Literaturverzeichnis</b>	<b>127</b>



# 1 Corpus der Entscheidungen des Bundesgerichtshofs (CE-BGH)

## 1.1 Überblick

Dieser Code lädt alle in der amtlichen Datenbank des Bundesgerichtshofs ([www.bundesgerichtshof.de](http://www.bundesgerichtshof.de)) veröffentlichten Entscheidungen des Bundesgerichtshofs (BGH) herunter und kompiliert sie in einen reichhaltigen menschen- und maschinenlesbaren Korpus. Es ist die Basis für den **Corpus der Entscheidungen des Bundesgerichtshofs**.

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 diesen Link erreichbar: <https://doi.org/10.5281/zenodo.3942742>

## 1.2 Funktionsweise

Primäre Endprodukte des Skripts sind folgende ZIP-Archive:

- Der volle Datensatz im CSV-Format (mit zusätzlichen Metadaten)
- Die reinen Metadaten im CSV-Format (wie unter 1, nur ohne Entscheidungsinhalte)
- Alle Entscheidungen im TXT-Format
- Alle Entscheidungen im PDF-Format
- Nur Leitsatz-Entscheidungen im PDF-Format
- Nur benannte Entscheidungen im PDF-Format
- Alle Analyse-Ergebnisse (Tabellen als CSV, Grafiken als PDF und PNG)

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 auf Zenodo im ZIP-Archiv “Source\_Files” bereitgestellten Dateien in einen leeren Ordner (!) und führen mit R diesen Befehl aus:

```
source("00_CE-BGH_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 20 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 den Fall, dass die Laufzeit die Datumsbarriere durchbricht.

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

```
## [1] "2022-02-12"
```

### 2.2 Datum und Uhrzeit (Beginn)

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

```
## [1] "2022-02-12 00:14:54 CET"
```

### 2.3 Packages Laden

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

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

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

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

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

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

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

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

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

## 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")
```

## 2.10 Allgemeine Konfiguration

### 2.10.1 Konfiguration einlesen

```
config <- parseTOML("CE-BGH_Config.toml")
```

### 2.10.2 Konfiguration anzeigen

```
print(config)
```

```
## List of 10
## $ cores      :List of 2
##   ..$ max     : logi TRUE
##   ..$ number: int 8
## $ debug      :List of 3
##   ..$ pages  : int 50
##   ..$ sample: int 500
##   ..$ toggle: logi FALSE
## $ doi        :List of 4
##   ..$ aktenzeichen : chr "10.5281/zenodo.4569564"
##   ..$ data         :List of 2
##   .. ..$ concept: chr "10.5281/zenodo.3942742"
##   .. ..$ version: chr "10.5281/zenodo.5910135"
##   ..$ personendaten: chr "10.5281/zenodo.4568682"
##   ..$ software     :List of 2
##   .. ..$ concept: chr "10.5281/zenodo.4705864"
##   .. ..$ version: chr "10.5281/zenodo.5910138"
## $ 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 [1:8] "text" "eingangsnummer" "datum" "doc_id" ...
## $ license   :List of 2
##   ..$ code  : chr "MIT-0"
##   ..$ data  : chr "Creative Commons Zero 1.0 Universal"
## $ parallel  :List of 3
##   ..$ extractPDF : logi TRUE
##   ..$ lingsummarize: logi TRUE
##   ..$ multihashes : logi TRUE
## $ project   :List of 3
##   ..$ author    : chr "Seán Fobbe"
##   ..$ fullname  : chr "Corpus der Entscheidungen des Bundesgerichtshofs"
##   ..$ shortname : chr "CE-BGH"
## $ 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.5910135"
```

## 2.10.6 Präfix für Dateien definieren

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

```
## [1] "CE-BGH_2022-02-12"
```

## 2.10.7 Präfix für Diagramme definieren

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

## 2.10.8 Quanteda-Optionen setzen

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

## 2.11 LaTeX Konfiguration

### 2.11.1 LaTeX Parameter definieren

```
latexdefs <- c("%=====\\n% Definitionen\\n  
%=====\\n",  
              "\\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: Weitere Datensätze

### 3.1 Registerzeichen und Verfahrensarten

Die Registerzeichen werden im Laufe des Skripts 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.

```
if (file.exists("data/AZ-BRD_1-0-1_DE_Registerzeichen_Datensatz.csv") == FALSE){  
  download.file("https://zenodo.org/record/4569564/files/AZ-BRD_1-0-1_DE_  
    Registerzeichen_Datensatz.csv?download=1",  
    "data/AZ-BRD_1-0-1_DE_Registerzeichen_Datensatz.csv")  
}
```

### 3.2 Personendaten zu Präsident:innen

Die Personendaten stammen aus folgendem Datensatz: »Seán Fobbe and Tilko Swalve (2021). Presidents and Vice-Presidents of the Federal Courts of Germany (PVP-FCG). Version 2021-04-08. Zenodo. DOI: 10.5281/zenodo.4568682«.

```
if (file.exists("data/PVP-FCG_2021-04-08_GermanFederalCourts_Presidents.csv") ==  
  FALSE){  
  download.file("https://zenodo.org/record/4568682/files/PVP-FCG_2021-04-08_  
    GermanFederalCourts_Presidents.csv?download=1",  
    "data/PVP-FCG_2021-04-08_GermanFederalCourts_Presidents.csv")  
}
```

### 3.3 Personendaten zu Vize-Präsident:innen

Die Personendaten stammen aus folgendem Datensatz: »Seán Fobbe and Tilko Swalve (2021). Presidents and Vice-Presidents of the Federal Courts of Germany (PVP-FCG). Version 2021-04-08. Zenodo. DOI: 10.5281/zenodo.4568682«.

```
if (file.exists("data/PVP-FCG_2021-04-08_GermanFederalCourts_VicePresidents.csv")  
  == FALSE){  
  download.file("https://zenodo.org/record/4568682/files/PVP-FCG_2021-04-08_  
    GermanFederalCourts_VicePresidents.csv?download=1",  
    "data/PVP-FCG_2021-04-08_GermanFederalCourts_VicePresidents.csv"  
  )  
}
```

## 4 Links suchen

### 4.1 Maximalen Such-Umfang einlesen

```
scope.source <- fread("data/CE-BGH_Scope.csv")
```

### 4.2 Maximalen Such-Umfang anzeigen

Die Variable “pagemax1” ist die maximale Anzahl Seiten wenn der Index mit 1 beginnt.  
Die Variable “pagemax0” ist die maximale Anzahl Seiten wenn der Index mit 0 beginnt.  
Die URL beginnt mit dem Index 0.

```
print(scope.source)
```

```
##      year pagemax1 pagemax0
## 1: 2000         74         73
## 2: 2001         81         80
## 3: 2002         92         91
## 4: 2003         97         96
## 5: 2004        103        102
## 6: 2005        104        103
## 7: 2006        104        103
## 8: 2007        111        110
## 9: 2008        122        121
## 10: 2009        114        113
## 11: 2010        122        121
## 12: 2011        124        123
## 13: 2012        118        117
## 14: 2013        107        106
## 15: 2014        103        102
## 16: 2015        103        102
## 17: 2016        110        109
## 18: 2017        105        104
## 19: 2018        100         99
## 20: 2019         99         98
## 21: 2020        109        108
## 22: 2021        113        112
## 23: 2022         3          2
##      year pagemax1 pagemax0
```

### 4.3 Funktion definieren

Diese Funktion nimmt eine ganzzahlige y-Variable als Maximum einer Sequenz von 1 bis y und weist ihr in einer data.table jeweils immer die gleiche x-Variable zu.

```
f.extend <- function(x, y, begin = 0){
  y.ext <- begin:y
}
```

```

x.ext <- rep(x, length(y.ext))
dt.out <- list(data.table(x.ext, y.ext))
return(dt.out)
}

f.extend <- Vectorize(f.extend)

```

#### 4.4 Genauen Such-Umfang berechnen

```

scope <- f.extend(scope.source$year,
                  scope.source$pagemax0)

scope <- rbindlist(scope)

setnames(scope,
          c("year",
            "page"))

```

#### 4.5 Locator einfügen

```

scope[, loc := {
  loc <- paste0(year, "-", page)
  list(loc)
}]

```

#### 4.6 [Debugging Modus] Reduzierung des Such-Umfangs

```

if (config$debug$toggle == TRUE){
  scope <- scope[sample(scope[,.N], config$debug$pages)][order(year, page)]
}

```

#### 4.7 Geschätzte Such-Dauer in Minuten

```

(scope[,.N] * 2.5) / 60

```

```
## [1] 96.58333
```

## 4.8 Zeitstempel: Linksammlung Beginn

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

```
## [1] "2022-02-12 00:14:56 CET"
```

## 4.9 Metadaten extrahieren

```
meta.all.list <- vector("list",
                        scope[,.N])

scope.random <- sample(scope[,.N])

for (i in seq_along(scope.random)){

  year <- scope$year[scope.random[i]]
  page <- scope$page[scope.random[i]]

  URL <- paste0("https://juris.bundesgerichtshof.de/cgi-bin/rechtsprechung/
list.py?Gericht=bgh&Art=en&Datum=",
               year,
               "&Seite=",
               page)

  html <- read_html(URL)

  link <- html_nodes(html, "a") %>% html_attr('href')

  link <- grep ("Blank=1.pdf",
               link,
               ignore.case = TRUE,
               value = TRUE)

  link <- sprintf("https://juris.bundesgerichtshof.de/cgi-bin/rechtsprechung/%s",
                 link)

  datum <- html_nodes(html, "[class='EDatum']") %>% html_text(trim = TRUE)
  spruch <- html_nodes(html, "[class='ESpruchk']") %>% html_text(trim = TRUE)
  az <- html_nodes(html, "[class='doklink']") %>% html_text(trim = TRUE)
  comment <- html_nodes(html, "[class='ETitel']") %>% html_text(trim = TRUE)

  meta.all.list[[scope.random[i]]] <- data.table(year,
                                                  page,
                                                  link,
```

```

        datum,
        spruch,
        az,
        comment)

remaining <- length(scope.random) - i

if ((remaining %% 10^2) == 0){
  message(paste(Sys.time(), "| Noch", remaining, "verbleibend."))
}

if((i %% 100) == 0){
  Sys.sleep(runif(1, 5, 15))
}else{
  Sys.sleep(runif(1, 1.5, 2.5))
}
}

```

```
## 2022-02-12 00:15:47 | Noch 2300 verbleibend.
```

```
## 2022-02-12 00:19:52 | Noch 2200 verbleibend.
```

```
## 2022-02-12 00:24:04 | Noch 2100 verbleibend.
```

```
## 2022-02-12 00:28:34 | Noch 2000 verbleibend.
```

```
## 2022-02-12 00:32:53 | Noch 1900 verbleibend.
```

```
## 2022-02-12 00:37:20 | Noch 1800 verbleibend.
```

```
## 2022-02-12 00:41:54 | Noch 1700 verbleibend.
```

```
## 2022-02-12 00:46:07 | Noch 1600 verbleibend.
```

```
## 2022-02-12 00:50:24 | Noch 1500 verbleibend.
```

```
## 2022-02-12 00:54:33 | Noch 1400 verbleibend.
```

## 2022-02-12 00:58:41 | Noch 1300 verbleibend.

## 2022-02-12 01:03:05 | Noch 1200 verbleibend.

## 2022-02-12 01:07:15 | Noch 1100 verbleibend.

## 2022-02-12 01:11:27 | Noch 1000 verbleibend.

## 2022-02-12 01:15:43 | Noch 900 verbleibend.

## 2022-02-12 01:19:51 | Noch 800 verbleibend.

## 2022-02-12 01:24:02 | Noch 700 verbleibend.

## 2022-02-12 01:28:24 | Noch 600 verbleibend.

## 2022-02-12 01:32:58 | Noch 500 verbleibend.

## 2022-02-12 01:37:25 | Noch 400 verbleibend.

## 2022-02-12 01:44:20 | Noch 300 verbleibend.

## 2022-02-12 01:49:54 | Noch 200 verbleibend.

## 2022-02-12 01:54:13 | Noch 100 verbleibend.

## 2022-02-12 02:01:00 | Noch 0 verbleibend.

## 4.10 Zeitstempel: Linksammlung Ende

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

```
## [1] "2022-02-12 02:01:02 CET"
```

## 4.11 Dauer Linksammlung

```
end.linkcollect - begin.linkcollect
```

```
## Time difference of 1.768227 hours
```

## 4.12 Zusammenfügen

```
dt.download <- rbindlist(meta.all.list)
```



## 5 Test-Reihe: Vollständigkeit der Auswertung

### 5.1 Locator einfügen

```
dt.download[, loc := {  
  loc <- paste0(year,  
                "-",  
                page)  
  list(loc)  
}]
```

### 5.2 Theoretischer Fehlbetrag

```
SOLL <- scope[, .N] * 30  
IST <- dt.download[, .N]  
  
missing.N <- SOLL - IST  
  
print(missing.N)
```

```
## [1] 246
```

### 5.3 Seiten mit weniger als 30 Entscheidungen anzeigen

```
less30 <- dt.download[, .N, keyby = "loc"][N < 30]  
  
print(less30)
```

```
##      loc  N  
## 1: 2000-73 16  
## 2: 2001-80  8  
## 3: 2002-91 23  
## 4: 2003-96 12  
## 5: 2004-102 27  
## 6: 2005-103 13  
## 7: 2006-103 23  
## 8: 2007-110 18  
## 9: 2008-121  4  
## 10: 2009-113 28  
## 11: 2010-121 18  
## 12: 2011-123 22  
## 13: 2012-117 21  
## 14: 2013-106 24  
## 15: 2014-102 20  
## 16: 2015-102 19
```

```
## 17: 2016-109 15
## 18: 2017-104 7
## 19: 2018-99 24
## 20: 2019-98 17
## 21: 2022-2 25
##      loc N
```

## 5.4 Fehlbetrag durch Seiten mit weniger als 30 Entscheidungen

```
less30.N <- (length(less30$N) * 30) - sum(less30$N)
print(less30.N)
```

```
## [1] 246
```

## 5.5 Tatsächlicher Fehlbetrag

**Test:** Ist der Fehlbetrag vollständig durch Seiten mit weniger als 30 Entscheidungen zu erklären? Falls ja, weisen beide sub-Tests maximal ein Ergebnis von 0 auf.

### 5.5.1 Fehlbetrag der NICHT durch Seiten mit weniger als 30 Entscheidungen erklärbar ist

```
print(missing.N - less30.N)
```

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

### 5.5.2 Gegenüberstellung: Anzahl Jahre und Anzahl Seiten mit weniger als 30 Entscheidungen

Für jedes Jahr sollte es eine letzte Seite mit weniger als 30 Entscheidungen geben. Falls zufällig die letzte Seite exakt 30 Entscheidungen hat, wäre das Ergebnis negativ. Ein Ergebnis von 0 oder kleiner bedeutet, dass der Test bestanden wurde. Der Test ist nur aussagekräftig wenn der gesamte Such-Umfang abgefragt wurde.

```
if (config$debug$toggle == FALSE){
  less30[,.N] - uniqueN(scope$year)
}
```

```
## [1] -2
```

## 5.6 Vorhandensein aller Jahr/Seiten-Kombinationen

Dieser Test zeigt an, ob alle Jahr/Seiten-Kombinationen auch in den Daten vorhanden sind. Falls nicht, zeigt er die fehlenden Kombinationen an.

```
setdiff(scope$loc,  
        dt.download$loc)
```

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

## 6 Bereinigung der Metadaten

### 6.1 Datum bereinigen

```
dt.download[, datum := {  
  datum <- as.character(datum)  
  datum <- as.IDate(datum, "%d.%m.%Y")  
  list(datum)}]
```

### 6.2 Aktenzeichen bereinigen

#### 6.2.1 Reguläre Substitutionen

```
az.out <- dt.download$az
```

**Hinweis:** An dieser Stelle wird eine mysteriöse Unterstrich-Variante mit einem regulären Unterstrich ersetzt. Es ist mir aktuell unklar um was für eine Art von Zeichen es sich handelt und wieso es in den Daten des Bundesgerichtshofs auftaucht. Weil die Code-Zeile zu einem Anzeigefehler in der LaTeX-Kompilierung führt, ist sie im Compilation Report nicht abgedruckt. Sie finden die Zeile im eigentlichen Source Code. Ich arbeite daran, die Anzeige zu vervollständigen.

```
az.out1 <- gsub("AK_34_und_35/21", "AK_34/21", az.out1)  
az.out1 <- gsub("StB_25_und_26/21", "StB_25/21", az.out1)  
  
az.out1 <- gsub("\\\\", "_", az.out1)  
  
az.out1 <- gsub("_und.*$", "", az.out1)  
  
az.out1 <- gsub("StB_", "NA_StB_", az.out1)  
az.out1 <- gsub("StbSt_\\(R\\)", "NA_StbStR", az.out1)  
az.out1 <- gsub("StbSt_\\(B\\)", "NA_StbStB", az.out1)  
  
az.out1 <- gsub("PatAnwZ_", "NA_PatAnwZ_", az.out1)  
  
az.out1 <- gsub("AnwZ_?\\(Brf[gG]\\)", "NA_AnwZBrfg", az.out1)  
  
az.out1 <- gsub("AK", "NA_AK", az.out1)  
az.out1 <- gsub("ARAnw_", "NA_ARAnw_", az.out1)  
az.out1 <- gsub("ARs_\\(Voll[zZ]\\)", "ARsVollz_", az.out1)  
az.out1 <- gsub("AR_\\(VZ\\)", "ARVZ", az.out1)  
az.out1 <- gsub("AR_\\(VS\\)", "ARVS", az.out1)  
az.out1 <- gsub("AR_\\(Ri\\)", "NA_ARRi", az.out1)  
az.out1 <- gsub("^AnwSt_\\(B\\)", "NA_AnwStB", az.out1)  
az.out1 <- gsub("^AnwSt_\\(R\\)", "NA_AnwStR", az.out1)  
az.out1 <- gsub("^PatAnwSt_\\(B\\)", "NA_PatAnwStB", az.out1)  
az.out1 <- gsub("^PatAnwSt_\\(R\\)", "NA_PatAnwStR", az.out1)  
az.out1 <- gsub("AnwZ_\\(B\\)", "NA_AnwZB", az.out1)
```

```

az.out1 <- gsub("AnwZ_\\(P\\)_", "NA_AnwZP_", az.out1)
az.out1 <- gsub("^AnwZ_", "NA_AnwZ_", az.out1)
az.out1 <- gsub("ARNot_", "NA_ARNot_", az.out1)

az.out1 <- gsub("BLw", "NA_BLw", az.out1)

az.out1 <- gsub("En[vV]R_", "NA_EnVR_", az.out1)
az.out1 <- gsub("EnZR_", "NA_EnZR_", az.out1)
az.out1 <- gsub("EnVZ_", "NA_EnVZ_", az.out1)
az.out1 <- gsub("EnZB_", "NA_EnZB_", az.out1)

az.out1 <- gsub("GmS-OGB_", "NA_GMSOGB_", az.out1)
az.out1 <- gsub("GSSt_", "NA_GSSt_", az.out1)
az.out1 <- gsub("GSZ_", "NA_GSZ_", az.out1)

az.out1 <- gsub("KZR_", "NA_KZR_", az.out1)
az.out1 <- gsub("KVZ_", "NA_KVZ_", az.out1)
az.out1 <- gsub("KRB_", "NA_KRB_", az.out1)
az.out1 <- gsub("KZB_", "NA_KZB_", az.out1)
az.out1 <- gsub("KVR_", "NA_KVR_", az.out1)

az.out1 <- gsub("LwZA_", "NA_LwZA_", az.out1)
az.out1 <- gsub("LwZR_", "NA_LwZR_", az.out1)
az.out1 <- gsub("LwZB", "NA_LwZB", az.out1)

az.out1 <- gsub("NotSt_\\(B\\)", "NA_NotStB", az.out1)
az.out1 <- gsub("NotSt_\\(Brfg\\)", "NA_NotStBrfg", az.out1)
az.out1 <- gsub("NotZ_\\(Brfg\\)", "NA_NotZBrfg", az.out1)
az.out1 <- gsub("NotZ_", "NA_NotZ_", az.out1)

az.out1 <- gsub("RiZ_?\\(R\\)", "NA_RiZR", az.out1)
az.out1 <- gsub("RiZ_?\\(B\\)", "NA_RiZB", az.out1)
az.out1 <- gsub("RiZ_", "NA_RiZ_", az.out1)
az.out1 <- gsub("RiSt_\\(R\\)", "NA_RiStR", az.out1)
az.out1 <- gsub("RiSt_\\(B\\)", "NA_RiStB", az.out1)

az.out1 <- gsub("WpSt_\\(R\\)", "NA_WpStR", az.out1)
az.out1 <- gsub("WpSt_\\(B\\)", "NA_WpStB", az.out1)

az.out1 <- gsub("VGS_", "NA_VGS_", az.out1)

az.out1 <- gsub("V_I_", "VI_", az.out1)
az.out1 <- gsub("XA", "Xa", az.out1)

```

## 6.2.2 Einzelne Fehler bereinigen

```

az.out1 <- gsub("_ZR_\\(Ü\\)", "_ZRÜ_", az.out1)
az.out1 <- gsub("_\\+_48", "", az.out1)
az.out1 <- gsub("u\\. _25_", "", az.out1)
az.out1 <- gsub("_- _28_07", "", az.out1)

az.out1 <- gsub("-[0-9]*_", "_", az.out1)

```

```
az.out1 <- gsub("_\\(a\\)", "_a", az.out1)
```

### 6.2.3 Variable “zusatz\_az” einfügen

```
indices <- grep("[0-9]*_[A-Za-zÜ]*_[0-9]*_[0-9]*_[a-z]*",  
               az.out1,  
               invert = TRUE)  
  
values <- grep("[0-9]*_[A-Za-zÜ]*_[0-9]*_[0-9]*_[a-z]*",  
              az.out1,  
              invert = TRUE,  
              value = TRUE)  
  
az.out1[indices] <- paste0(values,  
                           "_NA")
```

### 6.2.4 Finale Korrekturen

Bei der Entscheidung 1 BGs 29/2009 ist als einzige unter allen Entscheidungen das Eingangsjahr vierstellig und nicht zweistellig, auch im Text der Entscheidung selber. Ich gehe dennoch davon aus, das es sich hier um einen Schreibfehler handelt und nehme eine Korrektur vor.

```
az.out1 <- gsub("1_BGs_29_2009_NA",  
               "1_BGs_29_09_NA",  
               az.out1)
```

### 6.2.5 Strenge REGEX-Validierung des Aktenzeichens

```
regex.test1 <- grep(paste0("[0-9A-Za]+", # Senatsnummer  
                           "_",  
                           "[A-Za-zÜ]+", # Registerzeichen  
                           "_",  
                           "[0-9]+", # Eingangsnummer  
                           "_",  
                           "[0-9]{2}", # Eingangsjahr  
                           "_",  
                           "[A-Za-z]+"), # Kollision  
                   az.out1,  
                   invert = TRUE,  
                   value = TRUE)
```

### 6.2.6 Ergebnis der REGEX-Validierung

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

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

### 6.2.7 Skript stoppen falls REGEX-Validierung gescheitert

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

### 6.2.8 Aktenzeichen-Vektor in Download Table einfügen

```
dt.download$az <- az.out1
```

## 6.3 Spruchkörper bereinigen

### 6.3.1 Reguläre Substitutionen

**Hinweis:** An dieser Stelle wird eine mysteriöse Unterstrich-Variante mit einem regulären Unterstrich ersetzt. Es ist mir aktuell unklar um was für eine Art von Zeichen es sich handelt und wieso es in den Daten des Bundesgerichtshofs auftaucht. Weil die Code-Zeile zu einem Anzeigefehler in der LaTeX-Kompilierung führt, ist sie im Compilation Report nicht abgedruckt. Sie finden die Zeile im eigentlichen Source Code. Ich arbeite daran, die Anzeige zu vervollständigen.

```
spruch1 <- gsub("(.)\\._Zivilsenat",  
               "Zivilsenat-\\1",  
               spruch1)  
  
spruch1 <- gsub("(I?Xa)",  
               "Zivilsenat-\\1",  
               spruch1)  
  
spruch1 <- gsub("(.)\\._Strafsenat",  
               "Strafsenat-\\1",  
               spruch1)  
  
spruch1 <- gsub("Senat_für Anwaltssachen",  
               "Anwaltsenat",  
               spruch1)  
  
spruch1 <- gsub("Senat_für Landwirtschaftssachen",  
               "Landwirtschaftsenat",  
               spruch1)  
  
spruch1 <- gsub("Senat_für Notarsachen",  
               "Notarsenat",  
               spruch1)
```

```

spruch1 <- gsub("Dienstgericht des_Bundes",
               "DienstgerichtBund",
               spruch1)

spruch1 <- gsub("Senat_für Wirtschaftsprüfersachen",
               "Wirtschaftsprüfersenat",
               spruch1)

spruch1 <- gsub("Senat_für Steuerberater-_und Steuerbevollmächtigtensachen",
               "Steuerberatersenat",
               spruch1)

spruch1 <- gsub("Gemeinsamer Senat der obersten Gerichtshöfe des Bundes",
               "GemSenatOGBund",
               spruch1)

spruch1 <- gsub("Senat_für Patentanwaltssachen",
               "Patentanwaltsenat",
               spruch1)

spruch1 <- gsub("Großer_Senat für_Strafsachen",
               "GrosserStrafsenat",
               spruch1)

spruch1 <- gsub("Großer_Senat für_Zivilsachen",
               "GrosserZivilsenat",
               spruch1)

spruch1 <- gsub("-_Zivilsenat",
               "",
               spruch1)

spruch1 <- gsub("Vereinigte Große_Senate",
               "VereinigteGrosseSenate",
               spruch1)

```

### 6.3.2 Alle Spruchkörper anzeigen

```
print(unique(spruch1))
```

```

## [1] "Strafsenat-5"      "Strafsenat-2"      "Strafsenat-3"
## [4] "Zivilsenat-VII"    "Strafsenat-4"      "Zivilsenat-III"
## [7] "Zivilsenat-V"      "Zivilsenat-VIII"   "Zivilsenat-XII"
## [10] "Zivilsenat-VI"     "Zivilsenat-X"      "Zivilsenat-XI"
## [13] "Zivilsenat-II"     "Zivilsenat-I"      "Zivilsenat-IX"
## [16] "Strafsenat-1"      "Zivilsenat-IV"     "Anwaltsenat"
## [19] "Landwirtschaftsenat" "Kartellsenat"      "Notarsenat"
## [22] "DienstgerichtBund" "Wirtschaftsprüfersenat" "Steuerberatersenat"
## [25] "GemSenatOGBund"    "Patentanwaltsenat"  "Ermittlungsrichter"
## [28] "GrosserStrafsenat" "Zivilsenat-IXa"     "GrosserZivilsenat"
## [31] "Zivilsenat-Xa"     "VereinigteGrosseSenate" "Zivilsenat-XIII"

```



```
## [34] "Strafsenat-6"          "Zivilsenat-VIa"
```

### 6.3.3 Spruchkörper-Vektor in Download Table einfügen

```
dt.download$spruch <- spruch1
```

## 6.4 Bemerkungen bereinigen

```
dt.download$comment <- gsub("Leitsaetz",  
                           "Leitsatz",  
                           dt.download$comment)  
  
dt.download$comment <- gsub("Leitsaz",  
                           "Leitsatz",  
                           dt.download$comment)  
  
dt.download$comment <- gsub("Leitsazt",  
                           "Leitsatz",  
                           dt.download$comment)
```

## 6.5 Variable “leitsatz” erstellen

```
dt.download$leitsatz <- ifelse(grepl("Leitsatz",  
                                   dt.download$comment),  
                              "LE",  
                              "NA")
```

## 6.6 Variable “name” erstellen

```
name <- sub("(.*\\r\\n.*)",  
           "\\1",  
           dt.download$comment)  
  
name[grepl("Leitsatz|Pressemitteilung|Berichtigung",  
           name,  
           ignore.case = TRUE)] <- NA  
  
name[grepl("^$", name)] <- NA  
  
dt.download$name <- name
```

## 6.7 Variable “name\_datei” erstellen

```
name_datei <- name

name_datei <- trimws(name_datei)

name_datei <- path_sanitize(name_datei,
                             "")

name_datei <- gsub(" +",
                  "_",
                  name_datei)

name_datei <- gsub("'",
                  "'",
                  name_datei)

name_datei <- gsub(",",
                  ",",
                  name_datei)

dt.download$name_datei <- name_datei
```

## 6.8 Dateinamen erstellen

```
filename <- paste("BGH",
                  dt.download$spruch,
                  dt.download$leitsatz,
                  dt.download$datum,
                  dt.download$az,
                  dt.download$name_datei,
                  sep="_")
```

## 6.9 Einzelkorrektur vornehmen

```
filename <- gsub("BGH_Strafsenat-3_([NALE]{2})_NA_NA_AK_13_19_NA",
                "BGH_Strafsenat-3_\\1_2019-05-07_NA_AK_13_19_NA",
                filename)
```

## 6.10 KollisionsID einfügen

### 6.10.1 Anzahl Duplikate

```
length(filename[duplicated(filename)])
```

```
## [1] 824
```

### 6.10.2 Kollisions-IDs vergeben

```
filenames1 <- make.unique(filename, sep = "-----")

indices <- grep("-----",
               filenames1,
               invert = TRUE)

values <- grep("-----",
              filenames1,
              invert = TRUE,
              value = TRUE)

filenames1[indices] <- paste0(values,
                              "_0")

filenames1 <- gsub("-----",
                  "_",
                  filenames1)
```

### 6.11 Zufällige Auswahl zur Prüfung anzeigen

```
filenames1[sample(length(filenames1), 50)]
```

```
## [1] "BGH_Strafsenat-2_NA_2014-08-06_2_StR_255_14_NA_NA_0"
## [2] "BGH_Zivilsenat-IX_NA_2002-07-18_IX_ZB_77_02_NA_NA_0"
## [3] "BGH_Zivilsenat-XII_LE_2016-03-09_XII_ZB_693_14_NA_NA_0"
## [4] "BGH_Strafsenat-3_NA_2002-01-15_3_StR_461_01_NA_NA_0"
## [5] "BGH_Zivilsenat-X_NA_2002-04-23_X_ZB_5_01_NA_NA_0"
## [6] "BGH_Strafsenat-4_NA_2019-07-18_4_StR_43_19_NA_NA_0"
## [7] "BGH_Zivilsenat-VIII_LE_2008-10-15_VIII_ZR_321_07_NA_NA_0"
## [8] "BGH_Strafsenat-4_NA_2006-12-05_4_StR_484_06_NA_NA_0"
## [9] "BGH_Strafsenat-5_NA_2021-02-02_5_StR_508_20_NA_NA_0"
## [10] "BGH_Strafsenat-3_NA_2009-03-17_3_StR_74_09_NA_NA_0"
## [11] "BGH_Anwaltsenat_NA_2000-02-14_NA_AnwZB_13_99_NA_NA_0"
## [12] "BGH_Strafsenat-4_NA_2009-09-22_4_StR_657_08_NA_NA_0"
## [13] "BGH_Strafsenat-5_NA_2008-02-06_5_StR_590_07_NA_NA_0"
## [14] "BGH_Zivilsenat-III_NA_2013-02-28_III_ZR_242_12_NA_NA_0"
## [15] "BGH_Strafsenat-3_NA_2006-10-17_3_StR_386_06_NA_NA_0"
## [16] "BGH_Notarsenat_LE_2020-07-20_NA_NotZBrfg_2_19_NA_NA_0"
## [17] "BGH_Zivilsenat-IX_NA_2007-11-08_IX_ZR_100_05_NA_NA_0"
## [18] "BGH_Strafsenat-3_NA_2007-05-02_3_StR_145_07_NA_NA_0"
## [19] "BGH_Strafsenat-4_NA_2020-12-01_4_StR_213_20_NA_NA_0"
## [20] "BGH_Zivilsenat-VI_LE_2004-07-13_VI_ZR_136_03_NA_NA_0"
## [21] "BGH_Zivilsenat-IX_NA_2010-10-21_IX_ZB_278_09_NA_NA_0"
## [22] "BGH_Strafsenat-3_NA_2011-03-01_3_StR_450_10_NA_NA_0"
## [23] "BGH_Zivilsenat-I_NA_2002-02-21_I_ZR_158_00_NA_NA_0"
## [24] "BGH_Zivilsenat-II_LE_2012-07-17_II_ZR_55_11_NA_NA_0"
## [25] "BGH_Zivilsenat-V_LE_2013-07-04_V_ZB_197_12_NA_NA_0"
```

```
## [26] "BGH_Strafsenat-5_NA_2020-07-08_5_StR_80_20_NA_NA_0"
## [27] "BGH_Strafsenat-1_NA_2017-08-24_1_StR_625_16_NA_NA_0"
## [28] "BGH_Zivilsenat-IV_NA_2019-03-06_IV_ZR_108_18_NA_NA_0"
## [29] "BGH_Zivilsenat-I_LE_2017-02-09_I_ZR_91_15_NA_Flughafen-Lübeck_0"
## [30] "BGH_Zivilsenat-IV_NA_2000-09-27_IV_ZB_6_00_NA_NA_0"
## [31] "BGH_Steuerberatersenat_LE_2007-07-19_NA_StbStR_3_06_NA_NA_0"
## [32] "BGH_Strafsenat-1_NA_2006-03-08_1_StR_60_06_NA_NA_0"
## [33] "BGH_Zivilsenat-IV_NA_2015-05-13_IV_ZR_501_14_NA_NA_0"
## [34] "BGH_Zivilsenat-XII_NA_2010-08-11_XII_ZB_78_10_NA_NA_0"
## [35] "BGH_Zivilsenat-VII_LE_2014-02-13_VII_ZB_39_13_NA_NA_0"
## [36] "BGH_Zivilsenat-V_NA_2013-01-23_V_ZB_4_12_NA_NA_0"
## [37] "BGH_Strafsenat-3_NA_2009-04-23_3_StR_83_09_NA_NA_0"
## [38] "BGH_Zivilsenat-VI_NA_2010-10-19_VI_ZR_304_08_NA_NA_0"
## [39] "BGH_Zivilsenat-II_NA_2015-02-03_II_ZR_54_14_NA_NA_0"
## [40] "BGH_Zivilsenat-X_LE_2005-03-30_X_ZR_126_01_NA_Blasfolienherstellung_0"
## [41] "BGH_Zivilsenat-II_NA_2011-06-06_II_ZB_7_07_NA_NA_0"
## [42] "BGH_Zivilsenat-VI_NA_2014-08-05_VI_ZR_544_13_NA_NA_0"
## [43] "BGH_Strafsenat-5_NA_2008-01-23_5_StR_553_07_NA_NA_0"
## [44] "BGH_Zivilsenat-IX_LE_2004-12-02_IX_ZR_200_03_NA_NA_0"
## [45] "BGH_Strafsenat-5_NA_2011-02-10_5_StR_588_10_NA_NA_0"
## [46] "BGH_Zivilsenat-VII_NA_2021-11-24_VII_ZR_591_21_NA_NA_0"
## [47] "BGH_Zivilsenat-IV_LE_2011-07-20_IV_ZR_76_09_NA_NA_0"
## [48] "BGH_Zivilsenat-III_NA_2012-02-22_III_ZR_301_11_NA_NA_0"
## [49] "BGH_Zivilsenat-III_LE_2006-07-06_III_ZR_257_05_NA_NA_0"
## [50] "BGH_Strafsenat-5_NA_2012-11-07_5_StR_511_12_NA_NA_0"
```

## 6.12 PDF-Endung anfügen

```
filenames2 <- paste0(filenames1,
                      ".pdf")
```

## 6.13 Strenge REGEX-Validierung: Gesamter Dateiname

```
regex.test2 <- grep(paste0("BGH", # gericht
                           "_",
                           ".*", # spruchkoerper_db
                           "_",
                           "[NALE]{2}", # leitsatz
                           "_",
                           "[0-9]{4}-[0-9]{2}-[0-9]{2}", # datum
                           "_",
                           "[A-Za-z0-9]*", # spruchkoerper_az
                           "_",
                           "[A-Za-zÜ]*", # registerzeichen
                           "_",
                           "[0-9-]*", # eingangsnummer
                           "_",
                           "[0-9]{2}", # eingangsjahr_az
                           "_",
                           "[A-Za-z]*", # zusatz_az
```

```

        "_",
        ".*", # name
        "_",
        "[NAO-9]*", # kollision
        "\\..pdf"), # Dateiendung
filenames2,
value = TRUE,
invert = TRUE)

```

## 6.14 Ergebnis der REGEX-Validierung

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

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

## 6.15 Skript stoppen falls REGEX-Validierung gescheitert

```

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

```

## 6.16 Vollen Dateinamen in Download Table einfügen

```
dt.download$filenames.final <- filenames2
```

## 7 Download der Entscheidungen im PDF-Format

### 7.1 [Debugging Modus] Reduzierung des Download-Umfangs

```
if (config$debug$toggle == TRUE){  
  dt.download <- dt.download[sample(.N,  
                                     config$debug$sample)]  
}
```

### 7.2 Zeitstempel: Download Beginn

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

```
## [1] "2022-02-12 02:01:09 CET"
```

### 7.3 Download durchführen

```
for (i in sample(dt.download[,.N])){  
  tryCatch({download.file(url = dt.download$link[i],  
                          destfile = dt.download$filenames.final[i])  
    },  
    error = function(cond) {  
      return(NA)}  
    )  
  Sys.sleep(runif(1, 0, 0.1))  
}
```

```
## Warning in download.file(url = dt.download$link[i],  
## destfile = dt.download$filenames.final[i]): URL 'https://  
## juris.bundesgerichtshof.de/cgi-bin/rechtsprechung/document.py?  
## Gericht=bgh&Art=en&Datum=2016&Seite=50&nr=75652&pos=1515&anz=3285&Blank=1.pdf  
## ':  
## status was 'Couldn't resolve host name'
```

### 7.4 Zeitstempel: Download Ende

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

```
## [1] "2022-02-12 07:00:10 CET"
```

## 7.5 Dauer: Download

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

```
## Time difference of 4.983732 hours
```

## 7.6 [Debugging Modus] Löschen zufälliger Dateien

Dient dazu den Wiederholungsversuch zu testen.

```
if (config$debug$toggle == TRUE){  
  files.pdf <- list.files(pattern = "\\\\.pdf")  
  unlink(sample(files.pdf, 5))  
}
```

## 7.7 Download: Zwischenergebnis

### 7.7.1 Anzahl herunterzuladender Dateien

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

```
## [1] 69294
```

### 7.7.2 Anzahl heruntergeladener Dateien

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

```
## [1] 69293
```

### 7.7.3 Fehlbetrag

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

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

### 7.7.4 Fehlende Dateien

```
missing <- setdiff(dt.download$filenames.final, files.pdf)
print(missing)
```

```
## [1] "BGH_Zivilsenat-VI_NA_2016-07-05_VI_ZR_340_15_NA_NA_0.pdf"
```

## 7.8 Wiederholungsversuch: Download

Download für fehlende Dokumente wiederholen.

```
if(N.missing > 0){

  dt.retry <- dt.download[filenames.final %in% missing]

  for (i in 1:dt.retry[,.N]){
    response <- GET(dt.retry$link[i])
    Sys.sleep(runif(1, 0.25, 0.75))
    if (response$headers$"content-type" == "application/pdf" & response$
status_code == 200){
      tryCatch({download.file(url = dt.retry$link[i],
                             destfile = dt.retry$filenames.final[i])
      },
      error=function(cond) {
        return(NA)}
      )
    }else{
      print(paste0(dt.retry$filenames.final[i], " : kein PDF vorhanden"))
    }
    Sys.sleep(runif(1, 2, 5))
  }
}
```

## 7.9 Download: Gesamtergebnis

### 7.9.1 Anzahl herunterzuladender Dateien

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

```
## [1] 69294
```



### 7.9.2 Anzahl heruntergeladener Dateien

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

```
## [1] 69294
```

### 7.9.3 Fehlbetrag

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

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

### 7.9.4 Fehlende Dateien

```
missing <- setdiff(dt.download$filenames.final, files.pdf)  
print(missing)
```

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

## 8 Text-Extraktion aus PDF

### 8.1 Vektor der zu extrahierenden Dateien erstellen

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

### 8.2 Anzahl zu extrahierender Dateien

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

```
## [1] 69294
```

### 8.3 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))
}

```

## 8.4 Text Extrahieren

```

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

  plan("multicore",
        workers = fullCores)

}else{

  plan("sequential")

}

```

```
f.future_pdf_to_txt(files.pdf)
```

Für fast alle Dateien taucht der Fehler “PDF Error: Expected the default config, but wasn’t able to find it or it isn’t a Dictionary” auf. Dieser scheint die Konvertierung zu TXT aber nicht zu hindern und in den Dateien finden sich bei Stichproben keine Fehler. # Korpus Erstellen

## 8.5 TXT-Dateien Einlesen

```

txt.bgh <- readtext("./*.txt",
                   docvarsfrom = "filenames",
                   docvarnames = c("gericht",
                                   "spruchkoerper_db",
                                   "leitsatz",

```

```

        "datum",
        "spruchkoerper_az",
        "registerzeichen",
        "eingangsnummer",
        "eingangsjahr_az",
        "zusatz_az",
        "name",
        "kollision"),
    dvsep = "_",
    encoding = "UTF-8")

```

## 8.6 In Data Table umwandeln

```
setDT(txt.bgh)
```

## 8.7 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.

### 8.7.1 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)
## }

```

### 8.7.2 Funktion ausführen

```
txt.bgh[, text := lapply(.(text), f.hyphen.remove)]
```

### 8.8 Variable “datum” als Datentyp “IDate” kennzeichnen

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

### 8.9 Variable “entscheidungsjahr” hinzufügen

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

### 8.10 Variable “eingangsjahr\_iso” hinzufügen

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

### 8.11 Datensatz nach Datum sortieren

Die Erstellung der Variablen für Präsident:innen und Vize-Präsident:innen trifft die starke Annahme, dass eine aufsteigende Sortierung nach Datum besteht. Wäre das nicht der Fall, würden dort Fehler auftreten.

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

### 8.12 Variable “praesi” hinzufügen

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

#### 8.12.1 Personaldaten einlesen

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

### 8.12.2 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
BGH	Weinkauff	Hermann	1950-10-01	1960-03-31
BGH	Heusinger	Bruno	1960-04-01	1968-03-31
BGH	Fischer	Robert	1968-04-01	1977-09-30
BGH	Pfeiffer	Gerd	1977-10-01	1987-12-31
BGH	Odersky	Walter	1988-01-01	1996-07-31
BGH	Geiss	Karlmann	1996-08-01	2000-05-31
BGH	VACANCY-1	VACANCY-1	2000-06-01	2000-07-14
BGH	Hirsch	Günter	2000-07-15	2008-01-31
BGH	Tolksdorf	Klaus	2008-02-01	2014-01-31
BGH	VACANCY-2	VACANCY-2	2014-02-01	2014-06-30
BGH	Limberg	Bettina	2014-07-01	NA

### 8.12.3 Hypothetisches Amtsende für Präsident:in

Weil der/die aktuelle Präsident:in 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
```

### 8.12.4 Schleife vorbereiten

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

### 8.12.5 Vektor erstellen

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

### 8.12.6 Vektor einfügen

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

## 8.13 Variable “v\_praesi” hinzufügen

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

### 8.13.1 Personaldaten einlesen

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

### 8.13.2 Personaldaten anzeigen

```
kable(vpPraesi,
      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
BGH	Glanzmann	Roderich	1965-05-17	1972-04-30
BGH	VACANCY-1	VACANCY-1	1972-05-01	1972-05-22
BGH	Hauß	Fritz	1972-05-23	1976-10-31
BGH	VACANCY-2	VACANCY-2	1976-11-01	1976-11-02
BGH	Pfeiffer	Gerd	1976-11-03	1977-09-30
BGH	Stimpel	Walter	1977-10-01	1985-11-30
BGH	VACANCY-3	VACANCY-3	1985-12-01	1985-12-01

(continued)

court	name_last	name_first	term_begin_date	term_end_date
BGH	Thumm	Ludwig	1985-12-02	1988-04-30
BGH	Salger	Hannskarl	1988-05-01	1994-11-30
BGH	Hagen	Horst	1994-12-01	1999-02-28
BGH	Jähnke	Burkhard	1999-03-01	2002-05-31
BGH	Wenzel	Joachim	2002-06-01	2005-06-30
BGH	Müller	Gerda	2005-07-01	2009-06-30
BGH	Schlick	Wolfgang	2009-07-01	2015-07-31
BGH	VACANCY-4	VACANCY-4	2015-08-01	2016-12-01
BGH	Ellenberger	Jürgen	2016-12-02	NA

### 8.13.3 Hypothetisches Amtsende für Vize-Präsident:in

Weil der/die aktuelle Vize-Präsident:in 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
```

### 8.13.4 Schleife vorbereiten

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

### 8.13.5 Vektor erstellen

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



### 8.13.6 Vektor einfügen

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

## 8.14 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.

### 8.14.1 Datensatz einlesen

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

### 8.14.2 Datensatz auf relevante Daten reduzieren

```
az.bgh <- az.source[stelle == "BGH" & position == "hauptzeichen"]
```

### 8.14.3 Indizes bestimmen

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

### 8.14.4 Vektor der Verfahrensarten erstellen und einfügen

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

## 8.15 Variable “aktenzeichen” hinzufügen

```
txt.bgh$aktenzeichen <- paste0(txt.bgh$spruchkoerper_az,  
                               " ",  
                               mgsub(txt.bgh$registerzeichen,  
                                     az.bgh$zeichen_code,  
                                     az.bgh$zeichen_original),  
                               " ",  
                               txt.bgh$eingangsnummer,  
                               "/",  
                               txt.bgh$eingangsjahr_az)
```

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

## 8.16 Variable “entscheidung\_typ” hinzufügen

### 8.16.1 Entscheidungen Parsen

```
matches <- regexpr("BESCHLUSS|URTEIL|VERFÜGUNG",
                  txt.bgh$text,
                  ignore.case = TRUE)
```

### 8.16.2 Indizes bestimmen

```
matches.logical <- ifelse(matches > 0,
                          TRUE,
                          FALSE)

matches.indices <- which(matches.logical)
```

### 8.16.3 Leeren Vektor erstellen

```
entscheidung_typ <- rep(NA,
                      txt.bgh[,.N])
```

### 8.16.4 Typen bei Indizes platzieren

```
entscheidung_typ[matches.indices] <- regmatches(txt.bgh$text,
                                                matches)
```

### 8.16.5 Typen auf Kurzform reduzieren

```
entscheidung_typ <- gsub("URTEIL",
                       "U",
                       entscheidung_typ,
                       ignore.case = TRUE)

entscheidung_typ <- gsub("BESCHLUSS",
                       "B",
                       entscheidung_typ,
                       ignore.case = TRUE)

entscheidung_typ <- gsub("VERFÜGUNG",
```

```
"V",
entscheidung_typ,
ignore.case = TRUE)
```

### 8.16.6 Vektor in Datensatz einfügen

```
txt.bgh$entscheidung_typ <- entscheidung_typ
```

## 8.17 Variable “ecli” hinzufügen

Struktur und Inhalt der ECLI für deutsche Gerichte sind auf dem Europäischen Justizportal näher erläutert.<sup>1</sup>

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

**ACHTUNG:** diese ECLIs sind experimentell. Der BGH vergibt offiziell ECLI-Identifikatoren nur für Entscheidungen ab 2016. Ausserdem steht die originale Kollisions-ID nicht zur Verfügung. Alle Entscheidungen mit einer Kollisions-ID größer 0 und ihre korrespondierenden Entscheidungen mit der ID 0 haben potenziell syntaktisch korrekte, aber semantisch fehlerhafte ECLIs.

### 8.17.1 Formatieren der Registerzeichen für ECLI

```
ecli.registerzeichen <- az.bgh$zeichen_original[targetindices]

ecli.registerzeichen <- gsub("\\\\(",
                           "\\\\.",
                           ecli.registerzeichen)

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

ecli.registerzeichen <- toupper(ecli.registerzeichen)
```

### 8.17.2 Erstellen der ECLI-Ordinalzahl

```
ecli.ordinalzahl <- paste0(format(txt.bgh$datum,
                                "%d%m%y"),
                           txt.bgh$entscheidung_typ,
                           ifelse(is.na(txt.bgh$spruchkoerper_az),
                                "",
                                txt.bgh$spruchkoerper_az),
                           ecli.registerzeichen,
                           txt.bgh$eingangsnummer,
```

<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)

```

      ".",
      formatC(txt.bgh$eingangsjahr_az,
              width = 2,
              flag = "0"),
      ".",
      txt.bgh$kollision)

```

### 8.17.3 Vollständige ECLI erstellen

```

txt.bgh$ecli <- paste0("ECLI:DE:BGH:",
                      txt.bgh$entscheidungsjahr,
                      ":",
                      eccli.ordinalzahl)

```

### 8.17.4 Zufällige ECLI-Beispiele zur manuellen Nachprüfung

```

sample(txt.bgh$ecli,
       20)

```

```

## [1] "ECLI:DE:BGH:2020:101120BXIZR216.20.0"
## [2] "ECLI:DE:BGH:2002:250602B4STR186.02.0"
## [3] "ECLI:DE:BGH:2006:120106BIIIZR407.04.0"
## [4] "ECLI:DE:BGH:2007:281107BIIIZR90.06.0"
## [5] "ECLI:DE:BGH:2014:130314BIXZB48.13.0"
## [6] "ECLI:DE:BGH:2015:160615BENVR66.13.0"
## [7] "ECLI:DE:BGH:2004:181104UIXZR299.00.0"
## [8] "ECLI:DE:BGH:2012:311012UXIIZR129.10.0"
## [9] "ECLI:DE:BGH:2012:220312B4STR651.11.0"
## [10] "ECLI:DE:BGH:2012:070212BIIIZR13.10.0"
## [11] "ECLI:DE:BGH:2007:180107BIXZB145.06.0"
## [12] "ECLI:DE:BGH:2005:211105BVIZA17.05.0"
## [13] "ECLI:DE:BGH:2014:030414BIXZR236.13.0"
## [14] "ECLI:DE:BGH:2007:250907BKVR25.06.0"
## [15] "ECLI:DE:BGH:2015:090615BVIZA32.14.0"
## [16] "ECLI:DE:BGH:2017:111017B4STR572.16.0"
## [17] "ECLI:DE:BGH:2004:290904B5STR339.04.0"
## [18] "ECLI:DE:BGH:2020:220620BXZB7.19.0"
## [19] "ECLI:DE:BGH:2002:060802B5STR322.02.0"
## [20] "ECLI:DE:BGH:2014:120214B1STR601.13.1"

```

## 8.18 Variable “bemerkung” hinzufügen

```

dt.names.txt <- gsub("\\.pdf",
                    "\\ .txt",
                    dt.download$filenames.final)

```

```
targetindices <- match(txt.bgh$doc_id,
                      dt.names.txt)

txt.bgh$bemerkung <- dt.download[targetindices]$comment
```

## 8.19 Variable “berichtigung” hinzufügen

```
txt.bgh$berichtigung <- ifelse(grepl("Berichtigung",
                                   txt.bgh$bemerkung,
                                   ignore.case = TRUE),
                              "Berichtigung",
                              NA)
```

## 8.20 Variable “leitsatz” hinzufügen

```
#txt.bgh$leitsatz <- dt.download[targetindices]$leitsatz
```

## 8.21 Variable “name” hinzufügen

```
#txt.bgh$name <- dt.download[targetindices]$name
```

## 8.22 Variable “doi\_concept” hinzufügen

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

## 8.23 Variable “doi\_version” hinzufügen

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

## 8.24 Variable “version” hinzufügen

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

## 8.25 Variable “lizenz” hinzufügen

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

## 8.26 Entfernen von Dokumenten ohne Typ/Name/Berichtigung

Dokumente ohne Typ, Name oder Berichtigung sind fast immer Platzhalter-Dokumente, die auf den Push-Service der Universität des Saarlandes hinweisen. Diese werden am Ende des Variablen-Abschnitts aus der CSV-Datei entfernt und die PDF- und TXT-Dateien zur weiteren Prüfung separat gespeichert.

### 8.26.1 Platzhalter-Dokumente definieren

Dokumente ohne Typ, Name und Berichtigung sind fast immer Platzhalter-Dokumente, die keine Begründung enthalten und/oder auf den Push-Service der Universität des Saarlandes hinweisen. Diese werden am Ende dieses Abschnitts entfernt.

```
placeholder.txt <- txt.bgh[is.na(entscheidung_typ) == TRUE & is.na(name) == TRUE  
    & is.na(berichtigung) == TRUE]$doc_id
```

### 8.26.2 Falsch-positive Dokumente behalten

```
falsepositives.names <- c("BGH_IX_NA_2002-03-21_IX_ZB_57_02_NA_NA_0",  
                          "BGH_X_LE_2006-11-23_X_ZR_16_05_NA_NA_0",  
                          "BGH_X_LE_2008-04-22_X_ZR_76_07_NA_NA_0")  
  
falsepositives.index <- grep(paste(falsepositives.names,  
                                collapse = "|"),  
                             falsepositives.names)  
  
if (length(falsepositives.index) != 0){  
  
placeholder.txt <- placeholder.txt[-falsepositives.index]  
  
}
```

### 8.26.3 Einzelkorrektur

Das folgende Dokument ist nach Extraktion ein leeres Text-Dokument, im originalen PDF aber ein funktionaler Scan. Es wird temporär vom Datensatz ausgeschlossen damit keine Fehler in der Zählung linguistischer Kennzahlen auftreten. In Zukunft wird ein OCR-Modul hierfür eingerichtet.

```
if (file.exists("BGH_I_LE_2006-07-13_I_ZR_241_03_NA_Kontaktanzeigen_0.txt") ==  
    TRUE){
```

```
placeholder.txt <- c(placeholder.txt,
                      "BGH_I_LE_2006-07-13_I_ZR_241_03_NA_Kontaktanzeigen_0.txt")
}
```

#### 8.26.4 Dokumente ohne Typ, Name und Berichtigung anzeigen

```
print(placeholder.txt)
```

```
## [1] "BGH_Strafsenat-4_NA_2000-08-03_4_StR_280_00_NA_NA_0.txt"
## [2] "BGH_Strafsenat-4_NA_2000-08-03_4_StR_302_00_NA_NA_0.txt"
## [3] "BGH_Strafsenat-4_NA_2000-08-10_4_StR_188_00_NA_NA_0.txt"
## [4] "BGH_Strafsenat-2_NA_2000-08-25_2_StR_295_00_NA_NA_0.txt"
## [5] "BGH_Strafsenat-4_NA_2000-12-21_4_StR_431_00_NA_NA_0.txt"
## [6] "BGH_Strafsenat-5_NA_2000-12-23_5_StR_377_00_NA_NA_0.txt"
## [7] "BGH_Strafsenat-3_NA_2001-02-22_3_StR_587_00_NA_NA_0.txt"
## [8] "BGH_Strafsenat-1_NA_2001-05-10_1_StR_505_00_NA_NA_0.txt"
## [9] "BGH_Strafsenat-4_NA_2001-07-10_4_StR_175_01_NA_NA_0.txt"
## [10] "BGH_Strafsenat-5_NA_2001-12-11_5_StR_372_01_NA_NA_0.txt"
## [11] "BGH_Strafsenat-1_NA_2002-02-05_1_StR_403_01_NA_NA_0.txt"
## [12] "BGH_Strafsenat-1_NA_2002-02-05_1_StR_570_01_NA_NA_0.txt"
## [13] "BGH_Zivilsenat-IX_NA_2002-03-21_IX_ZB_57_02_NA_NA_0.txt"
## [14] "BGH_Strafsenat-2_NA_2002-06-26_2_StR_175_02_NA_NA_0.txt"
## [15] "BGH_Strafsenat-1_NA_2002-09-10_1_StR_318_02_NA_NA_0.txt"
## [16] "BGH_Strafsenat-4_NA_2002-11-05_4_StR_312_02_NA_NA_0.txt"
## [17] "BGH_Strafsenat-5_NA_2002-11-06_5_StR_421_02_NA_NA_0.txt"
## [18] "BGH_Strafsenat-2_NA_2002-12-11_2_StR_400_02_NA_NA_0.txt"
## [19] "BGH_Strafsenat-1_NA_2003-01-14_1_StR_502_02_NA_NA_0.txt"
## [20] "BGH_Strafsenat-1_NA_2003-01-29_1_StR_494_02_NA_NA_0.txt"
## [21] "BGH_Strafsenat-1_NA_2003-01-29_1_StR_519_02_NA_NA_0.txt"
## [22] "BGH_Strafsenat-3_NA_2003-01-30_3_StR_428_02_NA_NA_0.txt"
## [23] "BGH_Strafsenat-1_NA_2003-03-02_1_StR_25_04_NA_NA_0.txt"
## [24] "BGH_Zivilsenat-I_LE_2003-05-08_I_ZR_287_02_NA_NA_0.txt"
## [25] "BGH_Strafsenat-5_NA_2003-05-20_5_StR_66_03_NA_NA_0.txt"
## [26] "BGH_Zivilsenat-IV_NA_2003-06-25_IV_ZB_32_02_NA_NA_0.txt"
## [27] "BGH_Strafsenat-4_NA_2003-06-26_4_StR_168_03_NA_NA_0.txt"
## [28] "BGH_Zivilsenat-I_LE_2003-06-26_I_ZR_269_00_NA_NA_0.txt"
## [29] "BGH_Strafsenat-1_NA_2003-09-09_1_StR_278_03_NA_NA_0.txt"
## [30] "BGH_Strafsenat-3_NA_2003-10-08_3_StR_342_03_NA_NA_0.txt"
## [31] "BGH_Strafsenat-4_NA_2003-12-16_4_StR_482_03_NA_NA_0.txt"
## [32] "BGH_Strafsenat-1_NA_2004-02-04_1_StR_545_03_NA_NA_0.txt"
## [33] "BGH_Strafsenat-2_NA_2004-02-13_2_StR_489_03_NA_NA_0.txt"
## [34] "BGH_Strafsenat-5_NA_2004-03-31_5_StR_498_03_NA_NA_0.txt"
## [35] "BGH_Zivilsenat-XII_LE_2004-04-07_XII_ZB_79_04_NA_NA_0.txt"
## [36] "BGH_Strafsenat-2_NA_2004-05-21_2_StR_35_04_NA_NA_0.txt"
## [37] "BGH_Strafsenat-4_NA_2004-05-27_4_StR_564_03_NA_NA_0.txt"
## [38] "BGH_Strafsenat-2_NA_2004-07-02_2_StR_32_04_NA_NA_0.txt"
## [39] "BGH_Zivilsenat-X_LE_2004-07-06_X_ZR_171_02_NA_NA_1.txt"
## [40] "BGH_Strafsenat-5_NA_2004-09-30_5_StR_266_04_NA_NA_0.txt"
## [41] "BGH_Strafsenat-3_NA_2005-01-25_3_StR_486_04_NA_NA_0.txt"
## [42] "BGH_Strafsenat-2_NA_2005-03-01_2_StR_18_05_NA_NA_0.txt"
```

```

## [43] "BGH_Strafsenat-1_NA_2005-03-30_1_StR_537_04_NA_NA_0.txt"
## [44] "BGH_Strafsenat-1_NA_2005-09-08_1_StR_323_05_NA_NA_0.txt"
## [45] "BGH_Strafsenat-1_NA_2005-10-26_1_StR_412_05_NA_NA_0.txt"
## [46] "BGH_Strafsenat-1_NA_2006-02-07_1_StR_555_05_NA_NA_0.txt"
## [47] "BGH_Strafsenat-1_NA_2006-05-09_1_StR_142_06_NA_NA_0.txt"
## [48] "BGH_Strafsenat-4_NA_2006-05-11_4_StR_10_06_NA_NA_0.txt"
## [49] "BGH_Strafsenat-4_NA_2006-05-11_4_StR_90_06_NA_NA_0.txt"
## [50] "BGH_Strafsenat-5_NA_2006-10-10_5_StR_212_06_NA_NA_0.txt"
## [51] "BGH_Strafsenat-4_NA_2006-10-19_4_StR_359_06_NA_NA_0.txt"
## [52] "BGH_Zivilsenat-X_LE_2006-11-23_X_ZR_16_05_NA_NA_0.txt"
## [53] "BGH_Strafsenat-2_NA_2006-12-13_2_StR_403_06_NA_NA_0.txt"
## [54] "BGH_Strafsenat-1_NA_2007-01-17_1_StR_539_06_NA_NA_0.txt"
## [55] "BGH_Strafsenat-3_NA_2007-04-03_3_StR_107_07_NA_NA_0.txt"
## [56] "BGH_Strafsenat-2_NA_2007-04-13_2_StR_519_06_NA_NA_0.txt"
## [57] "BGH_Strafsenat-1_NA_2007-04-25_1_StR_130_07_NA_NA_0.txt"
## [58] "BGH_Strafsenat-1_NA_2007-04-25_1_StR_152_07_NA_NA_0.txt"
## [59] "BGH_Strafsenat-3_NA_2007-05-02_3_StR_145_07_NA_NA_0.txt"
## [60] "BGH_Strafsenat-3_NA_2007-10-01_3_StR_379_07_NA_NA_0.txt"
## [61] "BGH_Strafsenat-2_NA_2007-10-24_2_StR_421_07_NA_NA_0.txt"
## [62] "BGH_Strafsenat-2_NA_2008-01-09_2_StR_504_07_NA_NA_0.txt"
## [63] "BGH_Strafsenat-2_NA_2008-01-16_2_StR_542_07_NA_NA_0.txt"
## [64] "BGH_Strafsenat-5_NA_2008-01-23_5_StR_606_07_NA_NA_0.txt"
## [65] "BGH_Strafsenat-2_NA_2008-02-01_2_StR_539_07_NA_NA_0.txt"
## [66] "BGH_Strafsenat-2_NA_2008-02-27_2_StR_520_07_NA_NA_0.txt"
## [67] "BGH_Strafsenat-5_NA_2008-03-03_5_StR_9_08_NA_NA_0.txt"
## [68] "BGH_Strafsenat-4_NA_2008-03-04_4_StR_589_07_NA_NA_0.txt"
## [69] "BGH_Zivilsenat-X_LE_2008-04-22_X_ZR_76_07_NA_NA_0.txt"
## [70] "BGH_Strafsenat-2_NA_2008-06-11_2_StR_30_08_NA_NA_0.txt"
## [71] "BGH_Strafsenat-1_NA_2008-07-09_1_StR_316_08_NA_NA_0.txt"
## [72] "BGH_Strafsenat-1_NA_2008-07-16_1_StR_259_08_NA_NA_0.txt"
## [73] "BGH_Strafsenat-3_NA_2008-08-12_3_StR_110_08_NA_NA_0.txt"
## [74] "BGH_Strafsenat-1_NA_2008-08-26_1_StR_391_08_NA_NA_0.txt"
## [75] "BGH_Strafsenat-1_NA_2008-08-26_1_StR_398_08_NA_NA_0.txt"
## [76] "BGH_Strafsenat-1_NA_2008-08-27_1_StR_433_08_NA_NA_0.txt"
## [77] "BGH_Strafsenat-1_NA_2008-09-09_1_StR_459_08_NA_NA_0.txt"
## [78] "BGH_Strafsenat-2_NA_2008-09-10_2_StR_320_08_NA_NA_0.txt"
## [79] "BGH_Strafsenat-2_NA_2008-10-15_2_StR_430_08_NA_NA_0.txt"
## [80] "BGH_Strafsenat-2_NA_2008-10-31_2_StR_378_08_NA_NA_0.txt"
## [81] "BGH_Strafsenat-3_NA_2008-11-25_3_StR_444_08_NA_NA_0.txt"
## [82] "BGH_Strafsenat-2_NA_2008-11-28_2_StR_471_08_NA_NA_0.txt"
## [83] "BGH_Strafsenat-5_NA_2009-01-07_5_StR_539_08_NA_NA_0.txt"
## [84] "BGH_Strafsenat-3_NA_2009-02-10_3_StR_546_08_NA_NA_0.txt"
## [85] "BGH_Strafsenat-2_NA_2009-02-11_2_StR_558_08_NA_NA_0.txt"
## [86] "BGH_Strafsenat-3_NA_2009-02-17_3_StR_37_09_NA_NA_0.txt"
## [87] "BGH_Strafsenat-3_NA_2009-03-17_3_StR_18_09_NA_NA_0.txt"
## [88] "BGH_Strafsenat-1_NA_2009-04-28_1_StR_148_09_NA_NA_0.txt"
## [89] "BGH_Strafsenat-1_NA_2009-04-28_1_StR_176_09_NA_NA_0.txt"
## [90] "BGH_Strafsenat-5_NA_2009-05-05_5_StR_118_09_NA_NA_0.txt"
## [91] "BGH_Strafsenat-5_NA_2009-05-06_5_StR_131_09_NA_NA_0.txt"
## [92] "BGH_Strafsenat-2_NA_2009-05-13_2_StR_142_09_NA_NA_0.txt"
## [93] "BGH_Strafsenat-5_NA_2009-05-26_5_StR_139_09_NA_NA_0.txt"
## [94] "BGH_Strafsenat-5_NA_2009-05-26_5_StR_180_09_NA_NA_0.txt"
## [95] "BGH_Strafsenat-2_NA_2009-06-03_2_StR_162_09_NA_NA_0.txt"
## [96] "BGH_Strafsenat-5_NA_2009-06-09_5_StR_190_09_NA_NA_0.txt"
## [97] "BGH_Strafsenat-2_NA_2009-06-10_2_StR_145_09_NA_NA_0.txt"
## [98] "BGH_Strafsenat-2_NA_2009-06-17_2_StR_190_09_NA_NA_0.txt"

```



```

## [99] "BGH_Strafsenat-4_NA_2009-07-28_4_StR_57_09_NA_NA_0.txt"
## [100] "BGH_Strafsenat-1_NA_2009-08-04_1_StR_349_09_NA_NA_0.txt"
## [101] "BGH_Strafsenat-1_NA_2009-08-05_1_StR_366_09_NA_NA_0.txt"
## [102] "BGH_Strafsenat-5_NA_2009-09-01_5_StR_309_09_NA_NA_0.txt"
## [103] "BGH_Strafsenat-3_NA_2009-09-08_3_StR_356_09_NA_NA_0.txt"
## [104] "BGH_Strafsenat-3_NA_2009-09-22_3_StR_203_09_NA_NA_0.txt"
## [105] "BGH_Strafsenat-3_NA_2009-09-22_3_StR_262_09_NA_NA_0.txt"
## [106] "BGH_Strafsenat-4_NA_2009-09-22_4_StR_657_08_NA_NA_0.txt"
## [107] "BGH_Strafsenat-2_NA_2009-09-23_2_StR_293_09_NA_NA_0.txt"
## [108] "BGH_Strafsenat-3_NA_2009-10-06_3_StR_375_09_NA_NA_0.txt"
## [109] "BGH_Strafsenat-2_NA_2009-10-21_2_StR_287_09_NA_NA_0.txt"
## [110] "BGH_Strafsenat-2_NA_2009-10-23_2_StR_317_09_NA_NA_0.txt"
## [111] "BGH_Strafsenat-5_NA_2009-12-08_5_StR_441_09_NA_NA_0.txt"
## [112] "BGH_Strafsenat-2_NA_2010-01-27_2_StR_555_09_NA_NA_0.txt"
## [113] "BGH_GemSenatOGBund_NA_2010-03-08_NA_GMSOGB_2_07_NA_NA_0.txt"
## [114] "BGH_Strafsenat-2_NA_2010-03-24_2_StR_579_09_NA_NA_0.txt"
## [115] "BGH_Strafsenat-5_NA_2010-03-24_5_StR_29_10_NA_NA_0.txt"
## [116] "BGH_Strafsenat-5_NA_2010-03-24_5_StR_31_10_NA_NA_0.txt"
## [117] "BGH_Strafsenat-1_NA_2010-05-11_1_StR_103_10_NA_NA_0.txt"
## [118] "BGH_Strafsenat-1_NA_2010-05-11_1_StR_188_10_NA_NA_0.txt"
## [119] "BGH_Strafsenat-3_NA_2010-05-18_3_StR_149_10_NA_NA_0.txt"
## [120] "BGH_Strafsenat-5_NA_2010-06-14_5_StR_207_10_NA_NA_0.txt"
## [121] "BGH_Zivilsenat-II_NA_2010-06-21_II_ZR_166_09_NA_NA_0.txt"
## [122] "BGH_Strafsenat-2_NA_2010-07-14_2_StR_240_10_NA_NA_0.txt"
## [123] "BGH_Strafsenat-1_NA_2010-08-24_1_StR_414_10_NA_NA_0.txt"
## [124] "BGH_Strafsenat-5_NA_2010-08-31_5_StR_321_10_NA_NA_0.txt"
## [125] "BGH_Strafsenat-2_NA_2010-09-01_2_StR_347_10_NA_NA_0.txt"
## [126] "BGH_Strafsenat-2_NA_2010-09-08_2_StR_316_10_NA_NA_0.txt"
## [127] "BGH_Strafsenat-2_NA_2010-09-08_2_StR_389_10_NA_NA_0.txt"
## [128] "BGH_Strafsenat-1_NA_2010-09-09_1_StR_376_10_NA_NA_0.txt"
## [129] "BGH_GemSenatOGBund_LE_2010-09-27_NA_GMSOGB_1_09_NA_NA_0.txt"
## [130] "BGH_Strafsenat-5_NA_2010-09-29_5_StR_280_10_NA_NA_0.txt"
## [131] "BGH_Strafsenat-3_NA_2010-10-12_3_StR_330_10_NA_NA_0.txt"
## [132] "BGH_Strafsenat-1_NA_2010-11-03_1_StR_432_10_NA_NA_0.txt"
## [133] "BGH_Strafsenat-5_NA_2010-11-23_5_StR_380_10_NA_NA_0.txt"
## [134] "BGH_Strafsenat-2_NA_2010-12-15_2_StR_196_10_NA_NA_0.txt"
## [135] "BGH_Strafsenat-5_NA_2011-01-12_5_StR_538_10_NA_NA_0.txt"
## [136] "BGH_Strafsenat-1_NA_2011-01-19_1_StR_569_10_NA_NA_0.txt"
## [137] "BGH_Strafsenat-5_NA_2011-02-21_5_StR_27_11_NA_NA_0.txt"
## [138] "BGH_Strafsenat-5_NA_2011-02-22_5_StR_11_11_NA_NA_0.txt"
## [139] "BGH_Strafsenat-5_NA_2011-02-24_5_StR_534_10_NA_NA_0.txt"
## [140] "BGH_Strafsenat-1_NA_2011-03-17_1_StR_11_11_NA_NA_0.txt"
## [141] "BGH_Strafsenat-1_NA_2011-05-03_1_StR_140_11_NA_NA_0.txt"
## [142] "BGH_Strafsenat-5_NA_2011-06-08_5_StR_181_11_NA_NA_0.txt"
## [143] "BGH_Strafsenat-1_NA_2011-07-13_1_StR_692_10_NA_NA_0.txt"
## [144] "BGH_Strafsenat-2_NA_2011-09-22_2_StR_391_11_NA_NA_0.txt"
## [145] "BGH_Strafsenat-5_NA_2011-10-11_5_StR_390_11_NA_NA_0.txt"
## [146] "BGH_Strafsenat-2_NA_2011-11-23_2_StR_292_11_NA_NA_0.txt"
## [147] "BGH_Strafsenat-5_NA_2011-11-30_5_StR_470_11_NA_NA_0.txt"
## [148] "BGH_Strafsenat-2_NA_2011-12-13_2_StR_521_11_NA_NA_0.txt"
## [149] "BGH_Strafsenat-1_NA_2011-12-21_1_StR_400_11_NA_NA_0.txt"
## [150] "BGH_Strafsenat-5_NA_2012-01-10_5_StR_490_11_NA_NA_0.txt"
## [151] "BGH_Strafsenat-5_NA_2012-01-11_5_StR_491_11_NA_NA_0.txt"
## [152] "BGH_Strafsenat-2_NA_2012-02-09_2_StR_534_11_NA_NA_0.txt"
## [153] "BGH_Zivilsenat-II_NA_2012-03-06_II_ZR_89_11_NA_NA_0.txt"
## [154] "BGH_Zivilsenat-XII_NA_2012-03-07_XII_ZR_23_11_NA_NA_0.txt"

```

```

## [155] "BGH_Strafsenat-2_NA_2012-03-15_2_StR_436_11_NA_NA_0.txt"
## [156] "BGH_Zivilsenat-IV_NA_2012-03-21_IV_ZR_50_10_NA_NA_0.txt"
## [157] "BGH_Strafsenat-3_NA_2012-03-27_3_StR_38_12_NA_NA_0.txt"
## [158] "BGH_Strafsenat-5_NA_2012-03-28_5_StR_81_12_NA_NA_0.txt"
## [159] "BGH_Strafsenat-5_NA_2012-05-07_5_StR_164_12_NA_NA_0.txt"
## [160] "BGH_Strafsenat-2_NA_2012-05-16_2_StR_107_12_NA_NA_0.txt"
## [161] "BGH_Strafsenat-5_NA_2012-06-05_5_StR_235_12_NA_NA_0.txt"
## [162] "BGH_Strafsenat-5_NA_2012-06-19_5_StR_257_12_NA_NA_0.txt"
## [163] "BGH_Strafsenat-2_NA_2012-07-10_2_StR_26_12_NA_NA_0.txt"
## [164] "BGH_Strafsenat-1_NA_2012-07-24_1_StR_221_12_NA_NA_0.txt"
## [165] "BGH_Strafsenat-5_NA_2012-08-16_5_StR_321_12_NA_NA_0.txt"
## [166] "BGH_GemSenatOGBund_LE_2012-08-22_NA_GMSOGB_1_10_NA_NA_0.txt"
## [167] "BGH_Strafsenat-1_NA_2012-08-23_1_StR_356_12_NA_NA_0.txt"
## [168] "BGH_Strafsenat-5_NA_2012-09-13_5_StR_244_12_NA_NA_0.txt"
## [169] "BGH_Strafsenat-1_NA_2012-09-25_1_StR_412_12_NA_NA_0.txt"
## [170] "BGH_Strafsenat-3_NA_2012-10-02_3_StR_202_12_NA_NA_0.txt"
## [171] "BGH_Strafsenat-1_NA_2012-10-24_1_StR_483_12_NA_NA_0.txt"
## [172] "BGH_Strafsenat-5_NA_2012-11-06_5_StR_473_12_NA_NA_0.txt"
## [173] "BGH_Strafsenat-5_NA_2012-11-27_5_StR_559_12_NA_NA_0.txt"
## [174] "BGH_Strafsenat-1_NA_2012-12-05_1_StR_546_12_NA_NA_0.txt"
## [175] "BGH_Strafsenat-1_NA_2013-01-22_1_StR_232_12_NA_NA_0.txt"
## [176] "BGH_Strafsenat-1_NA_2013-01-23_1_StR_596_12_NA_NA_0.txt"
## [177] "BGH_Strafsenat-1_NA_2013-02-19_1_StR_275_12_NA_NA_0.txt"
## [178] "BGH_Strafsenat-1_NA_2013-03-05_1_StR_37_13_NA_NA_0.txt"
## [179] "BGH_Strafsenat-5_NA_2013-03-19_5_StR_80_13_NA_NA_0.txt"
## [180] "BGH_Zivilsenat-VI_NA_2013-03-19_VI_ZR_106_12_NA_NA_0.txt"
## [181] "BGH_Zivilsenat-VI_NA_2013-03-19_VI_ZR_107_12_NA_NA_0.txt"
## [182] "BGH_Zivilsenat-VI_NA_2013-03-19_VI_ZR_108_12_NA_NA_0.txt"
## [183] "BGH_Strafsenat-5_NA_2013-03-20_5_StR_28_13_NA_NA_0.txt"
## [184] "BGH_Strafsenat-1_NA_2013-05-02_1_StR_96_13_NA_NA_0.txt"
## [185] "BGH_Strafsenat-4_NA_2013-07-16_4_StR_66_13_NA_NA_0.txt"
## [186] "BGH_Strafsenat-5_NA_2013-09-03_5_StR_187_13_NA_NA_0.txt"
## [187] "BGH_Strafsenat-3_NA_2013-09-17_3_StR_227_13_NA_NA_0.txt"
## [188] "BGH_Strafsenat-3_NA_2013-11-14_3_StR_92_13_NA_NA_0.txt"
## [189] "BGH_Strafsenat-5_NA_2014-01-22_5_StR_561_13_NA_NA_0.txt"
## [190] "BGH_Strafsenat-1_NA_2014-01-30_1_StR_616_13_NA_NA_0.txt"
## [191] "BGH_Strafsenat-5_NA_2014-03-10_5_StR_51_14_NA_NA_0.txt"
## [192] "BGH_Strafsenat-5_NA_2014-04-08_5_StR_97_14_NA_NA_0.txt"
## [193] "BGH_Strafsenat-2_NA_2014-07-08_2_StR_195_14_NA_NA_0.txt"
## [194] "BGH_Strafsenat-2_NA_2014-08-05_2_StR_172_14_NA_NA_0.txt"
## [195] "BGH_Strafsenat-3_NA_2014-10-01_3_StR_150_14_NA_NA_0.txt"
## [196] "BGH_Strafsenat-2_NA_2014-10-14_2_StR_44_14_NA_NA_0.txt"
## [197] "BGH_Strafsenat-2_NA_2014-10-22_2_StR_62_14_NA_NA_0.txt"
## [198] "BGH_Strafsenat-3_NA_2015-03-31_3_StR_527_14_NA_NA_0.txt"
## [199] "BGH_Strafsenat-4_NA_2015-05-06_4_StR_87_15_NA_NA_0.txt"
## [200] "BGH_Strafsenat-5_NA_2015-06-16_5_StR_184_15_NA_NA_0.txt"
## [201] "BGH_Strafsenat-5_NA_2015-09-30_5_StR_347_15_NA_NA_0.txt"
## [202] "BGH_Strafsenat-4_NA_2015-12-16_4_StR_226_15_NA_NA_0.txt"
## [203] "BGH_Strafsenat-5_NA_2016-01-13_5_StR_460_15_NA_NA_0.txt"
## [204] "BGH_Strafsenat-2_NA_2016-01-14_2_StR_449_15_NA_NA_0.txt"
## [205] "BGH_Strafsenat-1_NA_2016-01-19_1_StR_603_15_NA_NA_0.txt"
## [206] "BGH_Strafsenat-2_NA_2016-02-04_2_StR_527_15_NA_NA_0.txt"
## [207] "BGH_Strafsenat-1_NA_2016-02-17_1_StR_209_15_NA_NA_0.txt"
## [208] "BGH_Strafsenat-2_NA_2016-02-24_2_StR_533_15_NA_NA_0.txt"
## [209] "BGH_Strafsenat-2_NA_2016-03-15_2_StR_157_15_NA_NA_0.txt"
## [210] "BGH_Strafsenat-5_NA_2016-04-05_5_StR_18_16_NA_NA_0.txt"

```

```

## [211] "BGH_Strafsenat-4_NA_2016-04-28_4_StR_88_16_NA_NA_0.txt"
## [212] "BGH_Strafsenat-2_NA_2016-05-19_2_StR_482_15_NA_NA_0.txt"
## [213] "BGH_Strafsenat-1_NA_2016-06-29_1_StR_110_16_NA_NA_0.txt"
## [214] "BGH_Strafsenat-4_NA_2016-09-13_4_StR_371_16_NA_NA_0.txt"
## [215] "BGH_Strafsenat-1_NA_2016-09-20_1_StR_349_16_NA_NA_0.txt"
## [216] "BGH_Strafsenat-3_NA_2016-12-21_3_StR_454_16_NA_NA_0.txt"
## [217] "BGH_Strafsenat-4_NA_2017-07-04_4_StR_149_17_NA_NA_0.txt"
## [218] "BGH_Strafsenat-2_NA_2017-08-03_2_StR_265_17_NA_NA_0.txt"
## [219] "BGH_Strafsenat-2_NA_2017-08-15_2_StR_222_17_NA_NA_0.txt"
## [220] "BGH_Strafsenat-2_NA_2017-08-22_2_StR_97_17_NA_NA_0.txt"
## [221] "BGH_Strafsenat-1_NA_2017-10-12_1_StR_324_17_NA_NA_0.txt"
## [222] "BGH_Strafsenat-2_NA_2017-11-15_2_StR_261_17_NA_NA_0.txt"
## [223] "BGH_Strafsenat-1_NA_2017-12-18_1_StR_547_17_NA_NA_0.txt"
## [224] "BGH_Strafsenat-2_NA_2018-02-21_2_StR_511_17_NA_NA_0.txt"
## [225] "BGH_Strafsenat-1_NA_2018-03-20_1_StR_401_17_NA_NA_0.txt"
## [226] "BGH_Strafsenat-1_NA_2018-03-22_1_StR_412_17_NA_NA_0.txt"
## [227] "BGH_Strafsenat-2_NA_2018-09-19_2_StR_455_17_NA_NA_0.txt"
## [228] "BGH_Strafsenat-5_NA_2018-09-24_5_StR_471_18_NA_NA_0.txt"
## [229] "BGH_Strafsenat-2_NA_2018-11-20_2_StR_325_18_NA_NA_0.txt"
## [230] "BGH_Strafsenat-5_NA_2019-01-22_5_StR_583_18_NA_NA_0.txt"
## [231] "BGH_Strafsenat-2_NA_2019-03-12_2_StR_22_19_NA_NA_0.txt"
## [232] "BGH_Strafsenat-5_NA_2019-04-16_5_StR_558_18_NA_NA_0.txt"
## [233] "BGH_Strafsenat-5_NA_2019-05-08_5_StR_182_19_NA_NA_0.txt"
## [234] "BGH_Strafsenat-1_NA_2019-08-06_1_StR_188_19_NA_NA_0.txt"
## [235] "BGH_Strafsenat-2_NA_2019-09-24_2_StR_222_19_NA_NA_0.txt"
## [236] "BGH_Strafsenat-3_NA_2019-10-02_3_StR_200_19_NA_NA_0.txt"
## [237] "BGH_Zivilsenat-V_NA_2019-12-19_V_ZR_85_19_NA_NA_0.txt"
## [238] "BGH_Strafsenat-3_NA_2020-01-08_3_StR_288_19_NA_NA_0.txt"
## [239] "BGH_Strafsenat-5_NA_2020-04-14_5_StR_473_19_NA_NA_0.txt"
## [240] "BGH_Strafsenat-6_NA_2020-04-21_6_StR_42_20_NA_NA_0.txt"
## [241] "BGH_Strafsenat-4_NA_2020-04-22_4_StR_492_19_NA_NA_0.txt"
## [242] "BGH_Strafsenat-5_NA_2020-04-27_5_StR_74_20_NA_NA_0.txt"
## [243] "BGH_Strafsenat-2_NA_2020-04-28_2_StR_25_20_NA_NA_0.txt"
## [244] "BGH_Strafsenat-4_NA_2020-05-07_4_StR_633_19_NA_NA_0.txt"
## [245] "BGH_Strafsenat-5_NA_2020-07-23_5_StR_251_20_NA_NA_0.txt"
## [246] "BGH_Strafsenat-4_NA_2020-07-28_4_StR_97_20_NA_NA_0.txt"
## [247] "BGH_Strafsenat-4_NA_2020-07-29_4_StR_598_19_NA_NA_0.txt"
## [248] "BGH_Strafsenat-6_NA_2020-07-29_6_StR_215_20_NA_NA_0.txt"
## [249] "BGH_Strafsenat-6_NA_2020-07-30_6_StR_182_20_NA_NA_0.txt"
## [250] "BGH_Strafsenat-3_NA_2020-09-01_3_StR_624_19_NA_NA_0.txt"
## [251] "BGH_Strafsenat-3_NA_2020-10-01_3_StR_265_20_NA_NA_0.txt"
## [252] "BGH_Strafsenat-2_NA_2020-10-14_2_StR_270_20_NA_NA_0.txt"
## [253] "BGH_Strafsenat-3_NA_2020-10-27_3_StR_260_20_NA_NA_0.txt"
## [254] "BGH_Strafsenat-2_NA_2020-11-04_2_StR_130_20_NA_NA_0.txt"
## [255] "BGH_Strafsenat-3_NA_2020-11-10_3_StR_311_20_NA_NA_0.txt"
## [256] "BGH_Strafsenat-6_NA_2020-11-16_6_StR_332_20_NA_NA_0.txt"
## [257] "BGH_Strafsenat-5_NA_2020-12-08_5_StR_437_20_NA_NA_0.txt"
## [258] "BGH_Strafsenat-6_NA_2020-12-16_6_StR_314_20_NA_NA_0.txt"
## [259] "BGH_Strafsenat-5_NA_2021-01-19_5_StR_471_20_NA_NA_0.txt"
## [260] "BGH_Strafsenat-5_NA_2021-01-19_5_StR_492_20_NA_NA_0.txt"
## [261] "BGH_Strafsenat-6_NA_2021-02-23_6_StR_11_21_NA_NA_0.txt"
## [262] "BGH_Strafsenat-3_NA_2021-03-09_3_StR_26_21_NA_NA_0.txt"
## [263] "BGH_Strafsenat-2_NA_2021-03-16_2_StR_36_21_NA_NA_0.txt"
## [264] "BGH_Strafsenat-6_NA_2021-03-23_6_StR_100_21_NA_NA_0.txt"
## [265] "BGH_Strafsenat-1_NA_2021-04-08_1_StR_69_21_NA_NA_0.txt"
## [266] "BGH_Strafsenat-5_NA_2021-04-13_5_StR_47_21_NA_NA_0.txt"

```

```
## [267] "BGH_Strafsenat-3_NA_2021-05-05_3_StR_157_20_NA_NA_0.txt"
## [268] "BGH_Strafsenat-3_NA_2021-05-26_3_StR_140_21_NA_NA_0.txt"
## [269] "BGH_Strafsenat-2_NA_2021-05-27_2_StR_405_20_NA_NA_0.txt"
## [270] "BGH_Strafsenat-6_NA_2021-06-02_6_StR_172_21_NA_NA_0.txt"
## [271] "BGH_Strafsenat-2_NA_2021-06-10_2_StR_44_21_NA_NA_0.txt"
## [272] "BGH_Strafsenat-1_NA_2021-06-16_1_StR_81_21_NA_NA_0.txt"
## [273] "BGH_Strafsenat-5_NA_2021-07-06_5_StR_185_21_NA_NA_0.txt"
## [274] "BGH_Strafsenat-1_NA_2021-07-29_1_StR_194_21_NA_NA_0.txt"
## [275] "BGH_Strafsenat-3_NA_2021-08-10_3_StR_212_21_NA_NA_0.txt"
## [276] "BGH_Strafsenat-4_NA_2021-08-19_4_StR_198_21_NA_NA_0.txt"
## [277] "BGH_Strafsenat-5_NA_2021-09-28_5_StR_290_21_NA_NA_0.txt"
## [278] "BGH_Strafsenat-1_NA_2021-10-06_1_StR_330_21_NA_NA_0.txt"
## [279] "BGH_Strafsenat-1_NA_2021-10-06_1_StR_335_21_NA_NA_0.txt"
## [280] "BGH_Strafsenat-6_NA_2021-10-06_6_StR_432_21_NA_NA_0.txt"
## [281] "BGH_Strafsenat-2_NA_2021-10-13_2_StR_418_19_NA_NA_0.txt"
## [282] "BGH_Strafsenat-2_NA_2021-12-08_2_StR_447_21_NA_NA_0.txt"
## [283] "BGH_Zivilsenat-I_NA_2021-12-09_I_ZR_146_20_NA_NA_0.txt"
## [284] "BGH_Strafsenat-3_NA_2022-01-12_3_StR_273_21_NA_NA_0.txt"
## [285] "BGH_Strafsenat-3_NA_2022-01-12_3_StR_443_21_NA_NA_0.txt"
## [286] "BGH_Strafsenat-4_NA_2022-01-19_4_StR_319_21_NA_NA_0.txt"
## [287] "BGH_Zivilsenat-XII_NA_2022-01-19_XII_ZB_183_21_NA_NA_0.txt"
## [288] "BGH_Zivilsenat-VIa_NA_2022-01-24_VIa_ZR_100_21_NA_NA_0.txt"
## [289] "BGH_Zivilsenat-III_NA_2022-01-27_III_ZR_3_21_NA_NA_0.txt"
## [290] "BGH_Zivilsenat-III_NA_2022-01-27_III_ZR_4_21_NA_NA_0.txt"
## [291] "BGH_Zivilsenat-VII_NA_2022-01-27_VII_ZR_303_20_NA_NA_0.txt"
## [292] "BGH_Strafsenat-5_NA_2022-02-07_5_StR_542_20_NA_NA_0.txt"
## [293] "BGH_Zivilsenat-VII_NA_2022-02-10_VII_ZR_365_21_NA_NA_0.txt"
## [294] "BGH_Zivilsenat-VII_NA_2022-02-10_VII_ZR_396_21_NA_NA_0.txt"
## [295] "BGH_Zivilsenat-VII_NA_2022-02-10_VII_ZR_679_21_NA_NA_0.txt"
## [296] "BGH_Zivilsenat-VII_NA_2022-02-10_VII_ZR_692_21_NA_NA_0.txt"
## [297] "BGH_Zivilsenat-VII_NA_2022-02-10_VII_ZR_717_21_NA_NA_0.txt"
```

### 8.26.5 PDF-Namen definieren

```
placeholder.pdf <- gsub("\\.txt",
                        "\\pdf",
                        placeholder.txt)
```

### 8.26.6 Platzhalter PDF/TXT speichern

```
dir.create("output/PlatzhalterDokumente")

file_move(placeholder.txt,
          "output/PlatzhalterDokumente")

file_move(placeholder.pdf,
          "output/PlatzhalterDokumente")
```

### 8.26.7 Platzhalter aus Datensatz entfernen

```
txt.bgh <- txt.bgh[!(doc_id %in% placeholder.txt)]
```

## 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] "ekli"          "aktenzeichen"   "name"           "bemerkung"
```

## 9.3 Liste zu prüfender Variablen

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

```
## [1] "gericht"      "spruchkoerper_db" "leitsatz"
## [4] "spruchkoerper_az" "registerzeichen"  "eingangsjahr_az"
## [7] "zusatz_az"      "kollision"        "entscheidungsjahr"
## [10] "eingangsjahr_iso" "praesi"           "v_praesi"
## [13] "verfahrensart"   "entscheidung_typ" "berichtigung"
## [16] "doi_concept"    "doi_version"      "version"
## [19] "lizenz"
```

## 9.4 Präfix definieren

```
prefix <- paste0(config$project$shortname,
                 "_01_Frequenztabelle_var-")
```

## 9.5 Frequenztabellen berechnen

```
f.fast.freqtable(txt.bgh,
                 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
BGH	68997	100	100	100
Total	68997	100	100	100



---

Frequency Table for Variable: spruchkoerper\_db

---

34 unique value(s) detected.

spruchkoerper_db	N	exactpercent	roundedpercent	cumulpercent
Anwaltsenat	2083	3.0189718	3.02	3.02
DienstgerichtBund	119	0.1724713	0.17	3.19
Ermittlungsrichter	46	0.0666696	0.07	3.26
GemSenatOGBund	1	0.0014493	0.00	3.26
GrosserStrafsenat	21	0.0304361	0.03	3.29
GrosserZivilsenat	6	0.0086960	0.01	3.30
Kartellsenat	948	1.3739728	1.37	4.67
Landwirtschaftsenat	396	0.5739380	0.57	5.25
Notarsenat	581	0.8420656	0.84	6.09
Patentanwaltsenat	29	0.0420308	0.04	6.13
Steuerberatersenat	16	0.0231894	0.02	6.15
Strafsenat-1	4023	5.8306883	5.83	11.98
Strafsenat-2	5743	8.3235503	8.32	20.31
Strafsenat-3	5399	7.8249779	7.82	28.13
Strafsenat-4	4806	6.9655202	6.97	35.10
Strafsenat-5	4790	6.9423308	6.94	42.04
Strafsenat-6	409	0.5927794	0.59	42.63
VereinigteGrosseSenate	1	0.0014493	0.00	42.64
Wirtschaftsprüfersenat	5	0.0072467	0.01	42.64
Zivilsenat-I	3786	5.4871951	5.49	48.13
Zivilsenat-II	2550	3.6958129	3.70	51.83
Zivilsenat-III	3142	4.5538212	4.55	56.38
Zivilsenat-IV	2712	3.9306057	3.93	60.31
Zivilsenat-IX	6510	9.4351928	9.44	69.75
Zivilsenat-IXa	200	0.2898677	0.29	70.03
Zivilsenat-V	4019	5.8248909	5.82	75.86

(continued)

spruchkoerper_db	N	exactpercent	roundedpercent	cumulpercent
Zivilsenat-VI	2816	4.0813369	4.08	79.94
Zivilsenat-VII	2081	3.0160732	3.02	82.96
Zivilsenat-VIII	2912	4.2204734	4.22	87.18
Zivilsenat-X	2173	3.1494123	3.15	90.33
Zivilsenat-XI	2769	4.0132180	4.01	94.34
Zivilsenat-XII	3564	5.1654420	5.17	99.51
Zivilsenat-XIII	179	0.2594316	0.26	99.77
Zivilsenat-Xa	162	0.2347928	0.23	100.00
Total	68997	100.0000000	100.00	100.00

Frequency Table for Variable: leitsatz

2 unique value(s) detected.

leitsatz	N	exactpercent	roundedpercent	cumulpercent
NA	49422	71.6292	71.63	71.63
LE	19575	28.3708	28.37	100.00
Total	68997	100.0000	100.00	100.00

Frequency Table for Variable: spruchkoerper\_az

22 unique value(s) detected.

spruchkoerper_az	N	exactpercent	roundedpercent	cumulpercent
NA	4861	7.0452339	7.05	7.05
1	4042	5.8582257	5.86	12.90
2	5755	8.3409424	8.34	21.24
3	4752	6.8872560	6.89	28.13
4	4807	6.9669696	6.97	35.10

(continued)

spruchkoerper_az	N	exactpercent	roundedpercent	cumulpercent
5	4791	6.9437802	6.94	42.04
6	413	0.5985767	0.60	42.64
I	3787	5.4886444	5.49	48.13
II	2550	3.6958129	3.70	51.83
III	3141	4.5523718	4.55	56.38
IV	2712	3.9306057	3.93	60.31
IX	6511	9.4366422	9.44	69.75
IXa	201	0.2913170	0.29	70.04
V	4019	5.8248909	5.82	75.86
VI	2815	4.0798875	4.08	79.94
VII	2082	3.0175225	3.02	82.96
VIII	2912	4.2204734	4.22	87.18
X	2173	3.1494123	3.15	90.33
XI	2768	4.0117686	4.01	94.34
XII	3563	5.1639926	5.16	99.50
XIII	180	0.2608809	0.26	99.77
Xa	162	0.2347928	0.23	100.00
Total	68997	100.0000000	100.00	100.00

Frequency Table for Variable: registerzeichen

55 unique value(s) detected.

registerzeichen	N	exactpercent	roundedpercent	cumulpercent
AK	337	0.4884270	0.49	0.49
ARAnw	6	0.0086960	0.01	0.50
ARNot	2	0.0028987	0.00	0.50
ARRi	7	0.0101454	0.01	0.51
ARVS	42	0.0608722	0.06	0.57

(continued)

registerzeichen	N	exactpercent	roundedpercent	cumulpercent
ARVZ	44	0.0637709	0.06	0.63
ARZ	138	0.2000087	0.20	0.83
ARs	982	1.4232503	1.42	2.26
ARsVollz	2	0.0028987	0.00	2.26
AnwStB	70	0.1014537	0.10	2.36
AnwStR	41	0.0594229	0.06	2.42
AnwZ	29	0.0420308	0.04	2.46
AnwZB	1124	1.6290563	1.63	4.09
AnwZBrfg	812	1.1768628	1.18	5.27
AnwZP	1	0.0014493	0.00	5.27
BGs	46	0.0666696	0.07	5.34
BLw	290	0.4203081	0.42	5.76
EnVR	346	0.5014711	0.50	6.26
EnVZ	41	0.0594229	0.06	6.32
EnZB	2	0.0028987	0.00	6.32
EnZR	41	0.0594229	0.06	6.38
GMSOGB	1	0.0014493	0.00	6.38
GSSt	21	0.0304361	0.03	6.41
GSZ	5	0.0072467	0.01	6.42
KRB	40	0.0579735	0.06	6.48
KVR	110	0.1594272	0.16	6.64
KVZ	57	0.0826123	0.08	6.72
KZB	13	0.0188414	0.02	6.74
KZR	295	0.4275548	0.43	7.17
LwZA	2	0.0028987	0.00	7.17
LwZB	17	0.0246388	0.02	7.19
LwZR	85	0.1231938	0.12	7.32
NotStB	19	0.0275374	0.03	7.35
NotStBrfg	59	0.0855110	0.09	7.43

(continued)

registerzeichen	N	exactpercent	roundedpercent	cumulpercent
NotZ	370	0.5362552	0.54	7.97
NotZBrfg	130	0.1884140	0.19	8.16
PatAnwStB	4	0.0057974	0.01	8.16
PatAnwStR	3	0.0043480	0.00	8.17
PatAnwZ	22	0.0318854	0.03	8.20
RiStB	1	0.0014493	0.00	8.20
RiStR	5	0.0072467	0.01	8.21
RiZ	15	0.0217401	0.02	8.23
RiZB	9	0.0130440	0.01	8.24
RiZR	88	0.1275418	0.13	8.37
StB	319	0.4623389	0.46	8.83
StR	23477	34.0261171	34.03	42.86
StbStB	6	0.0086960	0.01	42.87
StbStR	10	0.0144934	0.01	42.88
VGS	1	0.0014493	0.00	42.88
WpStB	2	0.0028987	0.00	42.88
WpStR	3	0.0043480	0.00	42.89
ZA	1308	1.8957346	1.90	44.78
ZB	11761	17.0456687	17.05	61.83
ZR	26333	38.1654275	38.17	100.00
ZRÜ	3	0.0043480	0.00	100.00
Total	68997	100.0000000	100.00	100.00

---

Frequency Table for Variable: eingangsjahr\_az

---

34 unique value(s) detected.

eingangsjahr_az	N	exactpercent	roundedpercent	cumulpercent
0	2368	3.4320333	3.43	3.43
1	2530	3.6668261	3.67	7.10
2	3231	4.6828123	4.68	11.78
3	3280	4.7538299	4.75	16.54
4	3037	4.4016407	4.40	20.94
5	3054	4.4262794	4.43	25.36
6	3161	4.5813586	4.58	29.94
7	3446	4.9944200	4.99	34.94
8	3464	5.0205081	5.02	39.96
9	3384	4.9045611	4.90	44.86
10	3466	5.0234068	5.02	49.89
11	3563	5.1639926	5.16	55.05
12	3257	4.7204951	4.72	59.77
13	3086	4.4726582	4.47	64.24
14	3144	4.5567199	4.56	68.80
15	3277	4.7494819	4.75	73.55
16	2962	4.2929403	4.29	77.84
17	2916	4.2262707	4.23	82.07
18	2834	4.1074250	4.11	86.18
19	3185	4.6161427	4.62	90.79
20	2943	4.2654028	4.27	95.06
21	1426	2.0667565	2.07	97.13
22	1	0.0014493	0.00	97.13
80	1	0.0014493	0.00	97.13
86	1	0.0014493	0.00	97.13
88	1	0.0014493	0.00	97.13
91	1	0.0014493	0.00	97.13
93	2	0.0028987	0.00	97.14
94	2	0.0028987	0.00	97.14
95	4	0.0057974	0.01	97.14

(continued)

eingangsjahr_az	N	exactpercent	roundedpercent	cumulpercent
96	9	0.0130440	0.01	97.16
97	100	0.1449338	0.14	97.30
98	516	0.7478586	0.75	98.05
99	1345	1.9493601	1.95	100.00
Total	68997	100.0000000	100.00	100.00

Frequency Table for Variable: zusatz\_az

2 unique value(s) detected.

zusatz_az	N	exactpercent	roundedpercent	cumulpercent
NA	68996	99.9985507	100	100
a	1	0.0014493	0	100
Total	68997	100.0000000	100	100

Frequency Table for Variable: kollision

6 unique value(s) detected.

kollision	N	exactpercent	roundedpercent	cumulpercent
0	68174	98.8071945	98.81	98.81
1	752	1.0899025	1.09	99.90
2	53	0.0768149	0.08	99.97
3	11	0.0159427	0.02	99.99
4	5	0.0072467	0.01	100.00
5	2	0.0028987	0.00	100.00
Total	68997	100.0000000	100.00	100.00

---

Frequency Table for Variable: entscheidungsjahr

---

23 unique value(s) detected.

entscheidungsjahr	N	exactpercent	roundedpercent	cumulpercent
2000	2200	3.188544	3.19	3.19
2001	2404	3.484209	3.48	6.67
2002	2745	3.978434	3.98	10.65
2003	2879	4.172645	4.17	14.82
2004	3078	4.461063	4.46	19.28
2005	3098	4.490050	4.49	23.77
2006	3105	4.500196	4.50	28.28
2007	3310	4.797310	4.80	33.07
2008	3613	5.236460	5.24	38.31
2009	3389	4.911808	4.91	43.22
2010	3625	5.253852	5.25	48.47
2011	3697	5.358204	5.36	53.83
2012	3506	5.081380	5.08	58.91
2013	3190	4.623389	4.62	63.54
2014	3071	4.450918	4.45	67.99
2015	3074	4.455266	4.46	72.44
2016	3271	4.740786	4.74	77.18
2017	3120	4.521936	4.52	81.71
2018	2988	4.330623	4.33	86.04
2019	2949	4.274099	4.27	90.31
2020	3249	4.708900	4.71	95.02
2021	3365	4.877024	4.88	99.90
2022	71	0.102903	0.10	100.00
Total	68997	100.000000	100.00	100.00

---



---

Frequency Table for Variable: eingangsjahr\_iso

---

34 unique value(s) detected.

eingangsjahr_iso	N	exactpercent	roundedpercent	cumulpercent
1980	1	0.0014493	0.00	0.00
1986	1	0.0014493	0.00	0.00
1988	1	0.0014493	0.00	0.00
1991	1	0.0014493	0.00	0.01
1993	2	0.0028987	0.00	0.01
1994	2	0.0028987	0.00	0.01
1995	4	0.0057974	0.01	0.02
1996	9	0.0130440	0.01	0.03
1997	100	0.1449338	0.14	0.18
1998	516	0.7478586	0.75	0.92
1999	1345	1.9493601	1.95	2.87
2000	2368	3.4320333	3.43	6.30
2001	2530	3.6668261	3.67	9.97
2002	3231	4.6828123	4.68	14.65
2003	3280	4.7538299	4.75	19.41
2004	3037	4.4016407	4.40	23.81
2005	3054	4.4262794	4.43	28.24
2006	3161	4.5813586	4.58	32.82
2007	3446	4.9944200	4.99	37.81
2008	3464	5.0205081	5.02	42.83
2009	3384	4.9045611	4.90	47.74
2010	3466	5.0234068	5.02	52.76
2011	3563	5.1639926	5.16	57.92
2012	3257	4.7204951	4.72	62.64
2013	3086	4.4726582	4.47	67.12
2014	3144	4.5567199	4.56	71.67

(continued)

eingangsjahr_iso	N	exactpercent	roundedpercent	cumulpercent
2015	3277	4.7494819	4.75	76.42
2016	2962	4.2929403	4.29	80.72
2017	2916	4.2262707	4.23	84.94
2018	2834	4.1074250	4.11	89.05
2019	3185	4.6161427	4.62	93.67
2020	2943	4.2654028	4.27	97.93
2021	1426	2.0667565	2.07	100.00
2022	1	0.0014493	0.00	100.00
Total	68997	100.0000000	100.00	100.00

Frequency Table for Variable: praesi

6 unique value(s) detected.

praesi	N	exactpercent	roundedpercent	cumulpercent
Geiss	938	1.3594794	1.36	1.36
Hirsch	21888	31.7231184	31.72	33.08
Limberg	23581	34.1768483	34.18	67.26
Tolksdorf	21013	30.4549473	30.45	97.71
VACANCY-1	288	0.4174095	0.42	98.13
VACANCY-2	1289	1.8681972	1.87	100.00
Total	68997	100.0000000	100.00	100.00

Frequency Table for Variable: v\_praesi

6 unique value(s) detected.

v_praesi	N	exactpercent	roundedpercent	cumulpercent
Ellenberger	15974	23.151731	23.15	23.15

(continued)

v_praesi	N	exactpercent	roundedpercent	cumulpercent
Jähnke	5719	8.288766	8.29	31.44
Müller	13228	19.171848	19.17	50.61
Schlick	20623	29.889705	29.89	80.50
VACANCY-4	4284	6.208966	6.21	86.71
Wenzel	9169	13.288984	13.29	100.00
Total	68997	100.000000	100.00	100.00

---

Frequency Table for Variable: verfahrensart

---

54 unique value(s) detected.

verfahrensart	N	exactpercent	roundedpercent	cumulpercent
NA	10	0.0144934	0.01	0.01
Aktenkontrolle für Haftprüfungsverfahren	337	0.4884270	0.49	0.50
Allgemeines Register (Anwaltssachen)	6	0.0086960	0.01	0.51
Allgemeines Register (Dienstgericht des Bundes)	7	0.0101454	0.01	0.52
Allgemeines Register (Notarsachen)	2	0.0028987	0.00	0.52
Allgemeines Register und Gerichtsstandsbestimmungen (Strafsachen)	982	1.4232503	1.42	1.95
Allgemeines Register und Gerichtsstandsbestimmungen (Zivilsachen)	138	0.2000087	0.20	2.15
Anträge außerhalb eines in der Rechtsmittelinstanz anhängigen Verfahrens (Zivilsachen)	1308	1.8957346	1.90	4.04
Anträge außerhalb eines in der Revisionsinstanz für Landwirtschaftssachen anhängigen Verfahrens	2	0.0028987	0.00	4.05

(continued)

verfahrensart	N	exactpercent	roundedpercent	cumulpercent
Anträge betreffend Richter im Bundesdienst und Mitglieder des Bundesrechnungshofes auf gerichtliche Entscheidung im Versetzungs- und Prüfungsverfahren sowie auf vorläufige Untersagung der Amtsgeschäfte	15	0.0217401	0.02	4.07
Berufungen und Anträge auf Zulassung der Berufung gegen Entscheidungen der Oberlandesgerichte in Notarsachen	130	0.1884140	0.19	4.26
Berufungen und Anträge auf Zulassung der Berufung gegen Entscheidungen eines Anwaltsgerichtshofes	812	1.1768628	1.18	5.43
Berufungen und Anträge auf Zulassung der Berufung gegen Urteile der Oberlandesgerichte in Disziplinarsachen gegen Notare	59	0.0855110	0.09	5.52
Beschwerden (Strafsachen)	319	0.4623389	0.46	5.98
Beschwerden gegen Beschlüsse der Oberlandesgerichte in Disziplinarsachen gegen Notare	19	0.0275374	0.03	6.01
Beschwerden gegen Entscheidungen eines Anwaltsgerichtshofes	1124	1.6290563	1.63	7.64
Beschwerden gegen die Nichtzulassung der Revision und Beschwerden der Richter, Staatsanwälte und Mitglieder des Bundesrechnungshofes gegen Disziplinarverfügungen	1	0.0014493	0.00	7.64
Beschwerden gegen die Nichtzulassung der Revision und Beschwerden gegen Entscheidungen eines Anwaltsgerichtshofes	70	0.1014537	0.10	7.74

(continued)

verfahrensart	N	exactpercent	roundedpercent	cumulpercent
Beschwerden gegen die Nichtzulassung der Revision und Beschwerden in berufsgerichtlichen Verfahren (Steuerberater- und Steuerbevollmächtigtensachen)	6	0.0086960	0.01	7.75
Beschwerden gegen die Nichtzulassung der Revision und Beschwerden in berufsgerichtlichen Verfahren (Wirtschaftsprüfersachen)	2	0.0028987	0.00	7.75
Beschwerden gegen die Nichtzulassung der Revision und Beschwerden nach der PatAO	4	0.0057974	0.01	7.76
Beschwerden und Rechtsbeschwerden in Landwirtschaftssachen der streitigen bürgerlichen Gerichtsbarkeit	17	0.0246388	0.02	7.78
Beschwerden, Rechtsbeschwerden, weitere Beschwerden, Beschwerden gegen die Nichtzulassung der Revision nach dem BEG (Zivilsachen)	11761	17.0456687	17.05	24.83
Einzelne richterliche Anordnungen des Ermittlungsrichters (Strafsachen)	46	0.0666696	0.07	24.90
Entscheidungen über Justizverwaltungsakte (Strafsachen)	42	0.0608722	0.06	24.96
Entscheidungen über Justizverwaltungsakte (Zivilsachen)	44	0.0637709	0.06	25.02
Erstinstanzliche Klagen auf Entschädigung wegen überlanger Gerichtsverfahren und strafrechtlicher Ermittlungsverfahren (Zivilsachen)	3	0.0043480	0.00	25.02
Großer Senat (Strafsachen)	21	0.0304361	0.03	25.05
Großer Senat (Zivilsachen)	5	0.0072467	0.01	25.06

(continued)

verfahrensart	N	exactpercent	roundedpercent	cumulpercent
Klagen über Entscheidungen in Zulassungssachen oder gegen sonstige Verwaltungsakte betreffend Rechtsanwälte beim Bundesgerichtshof	29	0.0420308	0.04	25.10
Klagen über die Anfechtung von Wahlen und Beschlüssen der Rechtsanwaltskammer beim BGH und der Bundesrechtsanwaltskammer	1	0.0014493	0.00	25.11
Nichtzulassungsbeschwerden (Kartellverwaltungssachen)	57	0.0826123	0.08	25.19
Nichtzulassungsbeschwerden in energiewirtschaftsrechtlichen Verwaltungssachen nach dem EnWG	41	0.0594229	0.06	25.25
Rechtsbeschwerden (Kartellverwaltungssachen)	110	0.1594272	0.16	25.41
Rechtsbeschwerden in Kartellbußgeldverfahren	40	0.0579735	0.06	25.46
Rechtsbeschwerden in Landwirtschaftssachen der freiwilligen Gerichtsbarkeit	290	0.4203081	0.42	25.89
Rechtsbeschwerden in energiewirtschaftsrechtlichen Verwaltungssachen nach dem EnWG	346	0.5014711	0.50	26.39
Rechtsbeschwerden und Beschwerden in bürgerlichen Rechtsstreitigkeiten (Kartellsachen)	13	0.0188414	0.02	26.41
Rechtsbeschwerden und Beschwerden in bürgerlichen Rechtsstreitigkeiten nach dem EnWG	2	0.0028987	0.00	26.41
Revisionen gegen Urteile eines Anwaltsgerichtshofes	41	0.0594229	0.06	26.47
Revisionen in Disziplinarsachen nach dem Deutschen Richtergesetz	5	0.0072467	0.01	26.48

(continued)

verfahrensart	N	exactpercent	roundedpercent	cumulpercent
Revisionen in Versetzungs- und Prüfungsverfahren nach dem Deutschen Richtergesetz	88	0.1275418	0.13	26.60
Revisionen in berufsgerichtlichen Verfahren (Steuerberater- und Steuerbevollmächtigensachen)	10	0.0144934	0.01	26.62
Revisionen in berufsgerichtlichen Verfahren (Wirtschaftsprüfersachen)	3	0.0043480	0.00	26.62
Revisionen nach der PatAO	3	0.0043480	0.00	26.63
Revisionen und Vorlegungssachen nach § 121 Abs.1 Nr.1, Abs. 2 GVG, § 79 Abs. 3 OWiG, §§ 13 Abs. 4, 25 StrRehaG (Strafsachen)	23477	34.0261171	34.03	60.65
Revisionen, Beschwerden gegen die Nichtzulassung der Revision und Anträge auf Zulassung der Sprungrevision (Landwirtschaftssachen)	85	0.1231938	0.12	60.78
Revisionen, Beschwerden gegen die Nichtzulassung der Revision und Anträge auf Zulassung der Sprungrevision in bürgerlichen Rechtsstreitigkeiten (Kartellsachen)	295	0.4275548	0.43	61.20
Revisionen, Beschwerden gegen die Nichtzulassung der Revision und Anträge auf Zulassung der Sprungrevision in bürgerlichen Rechtsstreitigkeiten nach dem EnWG	41	0.0594229	0.06	61.26
Revisionen, Beschwerden gegen die Nichtzulassung der Revision, Anträge auf Zulassung der Sprungrevision, Berufungen in Patentsachen (Zivilsachen)	26333	38.1654275	38.17	99.43
Vereinigte Große Senate	1	0.0014493	0.00	99.43

(continued)

verfahrensart	N	exactpercent	roundedpercent	cumulpercent
Verwaltungsstreitverfahren in Notarsachen und Beschwerden gegen Entscheidungen der Oberlandesgerichte	370	0.5362552	0.54	99.97
Verwaltungsstreitverfahren in Patentanwaltssachen; Berufungen und Anträge auf Zulassung der Berufung und Beschwerden gegen Entscheidungen der Oberlandesgerichte	22	0.0318854	0.03	100.00
Vorlegungssachen (Strafvollzugssachen)	2	0.0028987	0.00	100.00
Total	68997	100.0000000	100.00	100.00

Frequency Table for Variable: entscheidung\_typ

4 unique value(s) detected.

entscheidung_typ	N	exactpercent	roundedpercent	cumulpercent
NA	26	0.0376828	0.04	0.04
B	49766	72.1277737	72.13	72.17
U	19194	27.8186008	27.82	99.98
V	11	0.0159427	0.02	100.00
Total	68997	100.0000000	100.00	100.00

Frequency Table for Variable: berichtigung

2 unique value(s) detected.

berichtigung	N	exactpercent	roundedpercent	cumulpercent
NA	68209	98.857921	98.86	98.86
Berichtigung	788	1.142079	1.14	100.00
Total	68997	100.000000	100.00	100.00



(continued)

berichtigung	N	exactpercent	roundedpercent	cumulpercent
--------------	---	--------------	----------------	--------------

---

Frequency Table for Variable: doi\_concept

---

1 unique value(s) detected.

doi_concept	N	exactpercent	roundedpercent	cumulpercent
10.5281/zenodo.3942742	68997	100	100	100
Total	68997	100	100	100

---

Frequency Table for Variable: doi\_version

---

1 unique value(s) detected.

doi_version	N	exactpercent	roundedpercent	cumulpercent
10.5281/zenodo.5910135	68997	100	100	100
Total	68997	100	100	100

---

Frequency Table for Variable: version

---

1 unique value(s) detected.

version	N	exactpercent	roundedpercent	cumulpercent
2022-02-12	68997	100	100	100
Total	68997	100	100	100

---

Frequency Table for Variable: lizenz

---

1 unique value(s) detected.

lizenz	N	exactpercent	roundedpercent	cumulpercent
Creative Commons Zero 1.0 Universal	68997	100	100	100
Total	68997	100	100	100

## 10 Frequenztabellen visualisieren

### 10.1 Präfix erstellen

```
prefix <- file.path(dir.analysis,  
                    paste0(config$project$shortname,  
                          "_01_Frequenztabelle_var-"))
```

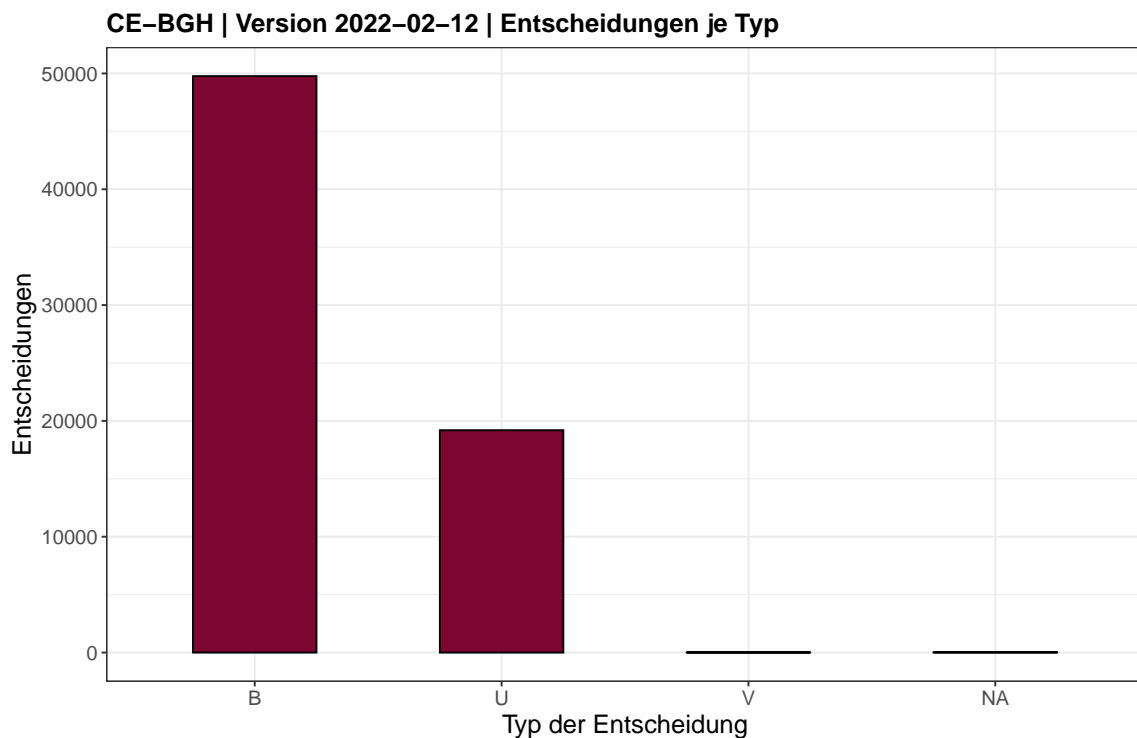
### 10.2 Tabellen einlesen

```
table.entsch.typ <- fread(paste0(prefix,  
                                "entscheidung_typ.csv"))  
  
table.spruch.db <- fread(paste0(prefix,  
                                "spruchkoerper_db.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 = "#7e0731",  
          color = "black",  
          width = 0.5) +  
  theme_bw() +  
  labs(  
    title = paste(prefix.figuretitle,  
                  "| Entscheidungen je 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)  
  )
```



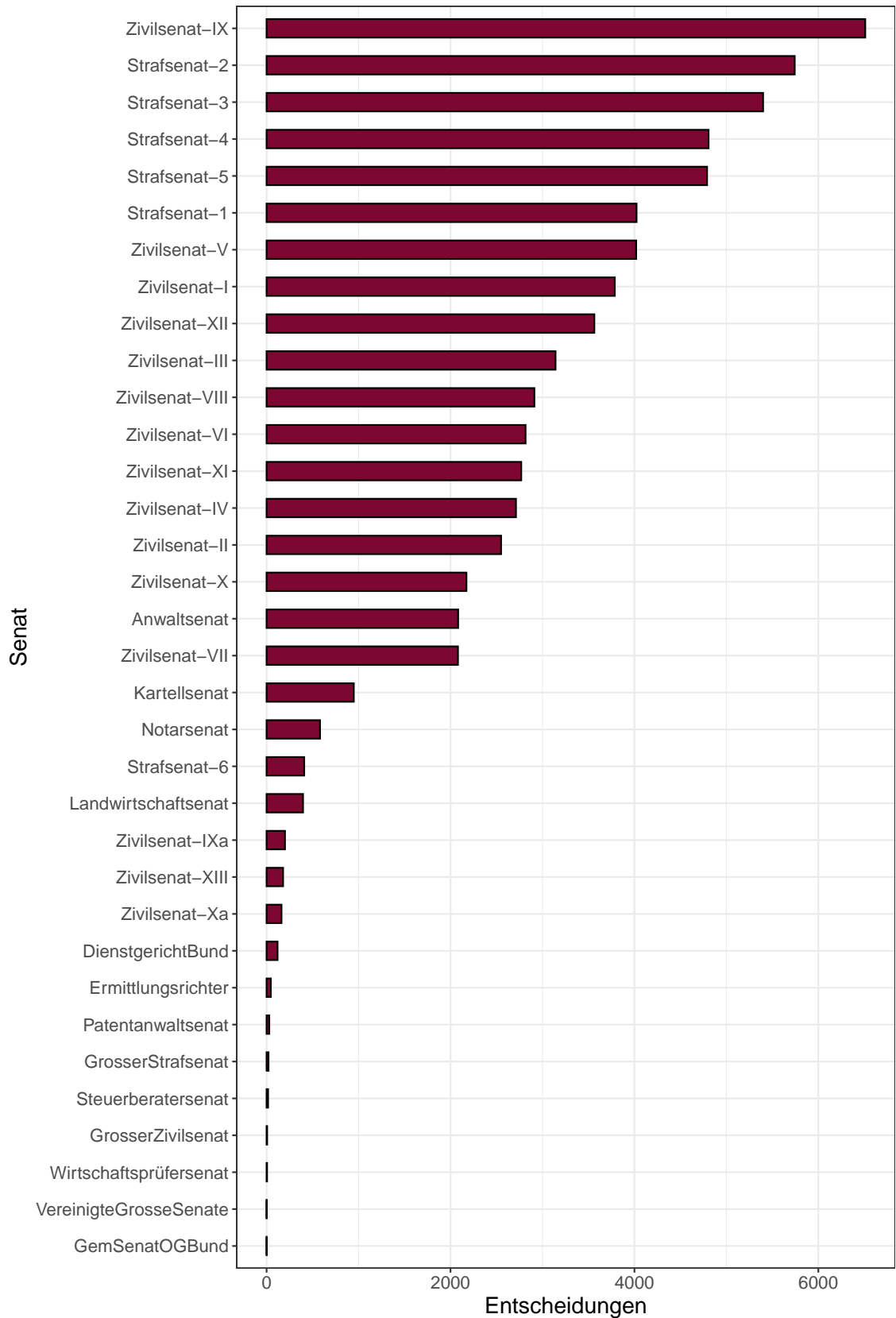
Fobbe | DOI: 10.5281/zenodo.5910135

## 10.4 Diagramm: Spruchkörper nach Datenbank

```
frequetable <- table.spruch.db[-.N]
```

```
ggplot(data = frequetable) +  
  geom_bar(aes(x = reorder(spruchkoerper_db,  
                           N),  
               y = N),  
           stat = "identity",  
           fill = "#7e0731",  
           color = "black",  
           width = 0.5) +  
  coord_flip() +  
  theme_bw() +  
  labs(  
    title = paste(prefix.figuretitle,  
                  "| Entscheidungen je Senat (DB)"),  
    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)  
  )
```

CE-BGH | Version 2022-02-12 | Entscheidungen je Senat (DB)

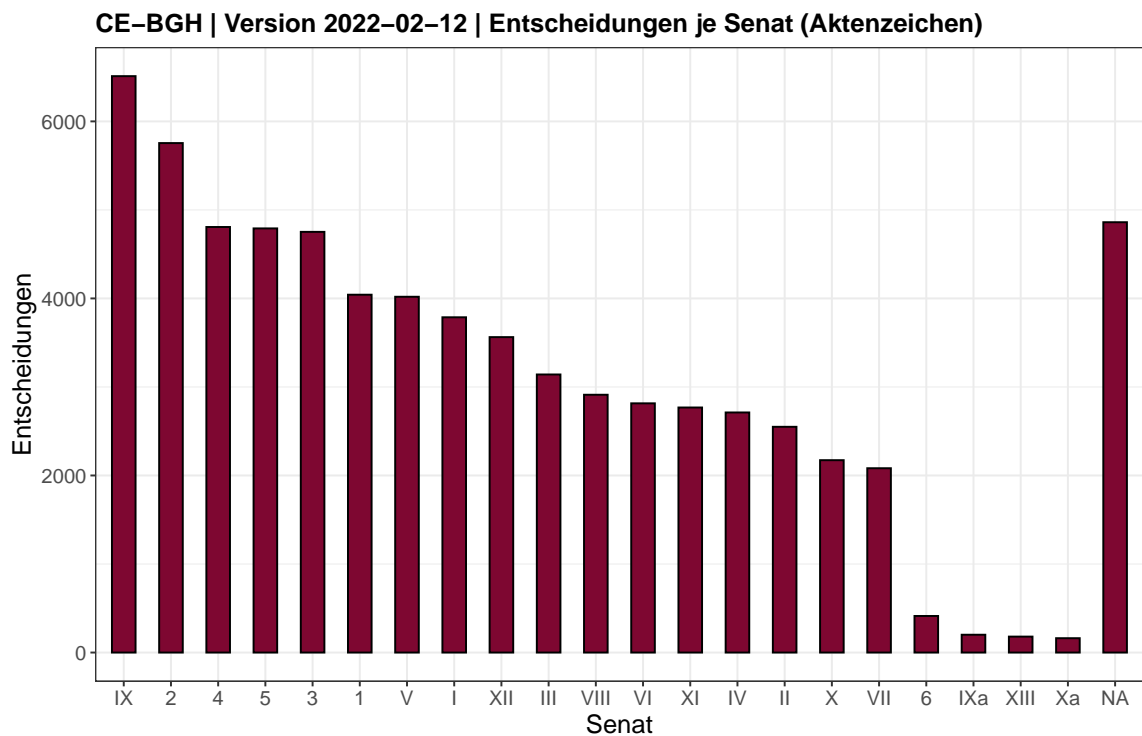


Fobbe | DOI: 10.5281/zenodo.5910135

## 10.5 Diagramm: Spruchkörper nach Aktenzeichen

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

```
ggplot(data = freqtable) +  
  geom_bar(aes(x = reorder(spruchkoerper_az,  
                           -N),  
               y = N),  
           stat = "identity",  
           fill = "#7e0731",  
           color = "black",  
           width = 0.5) +  
  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.5910135

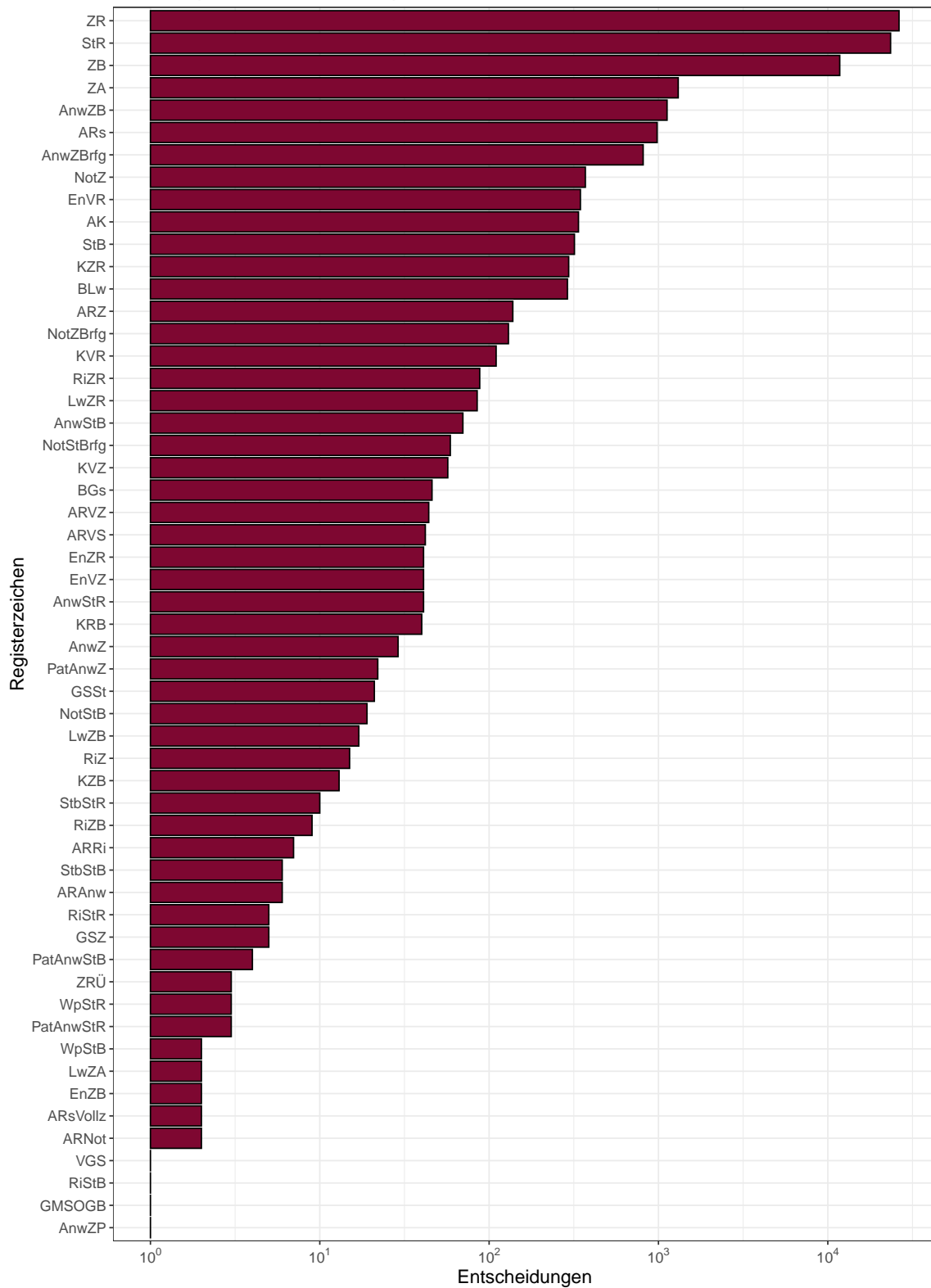
## 10.6 Diagramm: Registerzeichen

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

```
ggplot(data = freqtable) +  
  geom_bar(aes(x = reorder(registerzeichen,  
                           N),  
              y = N),  
          stat = "identity",  
          fill = "#7e0731",  
          color = "black") +  
  scale_y_log10(breaks = trans_breaks("log10", function(x) 10^x),  
               labels = trans_format("log10", math_format(10^.x))) +  
  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)  
  )
```



CE-BGH | Version 2022-02-12 | Entscheidungen je Registerzeichen

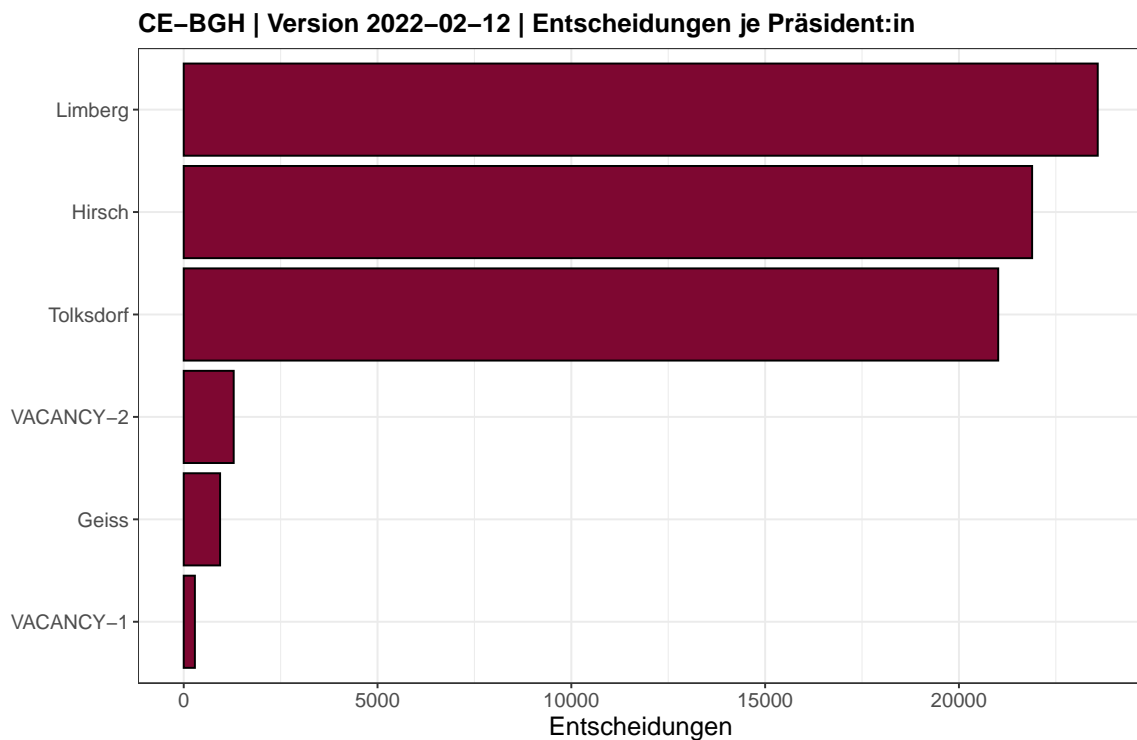


Fobbe | DOI: 10.5281/zenodo.5910135

## 10.7 Diagramm: Präsident:in

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

```
ggplot(data = frequetable) +  
  geom_bar(aes(x = reorder(praesi,  
                           N),  
               y = N),  
           stat = "identity",  
           fill = "#7e0731",  
           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)  
  )
```

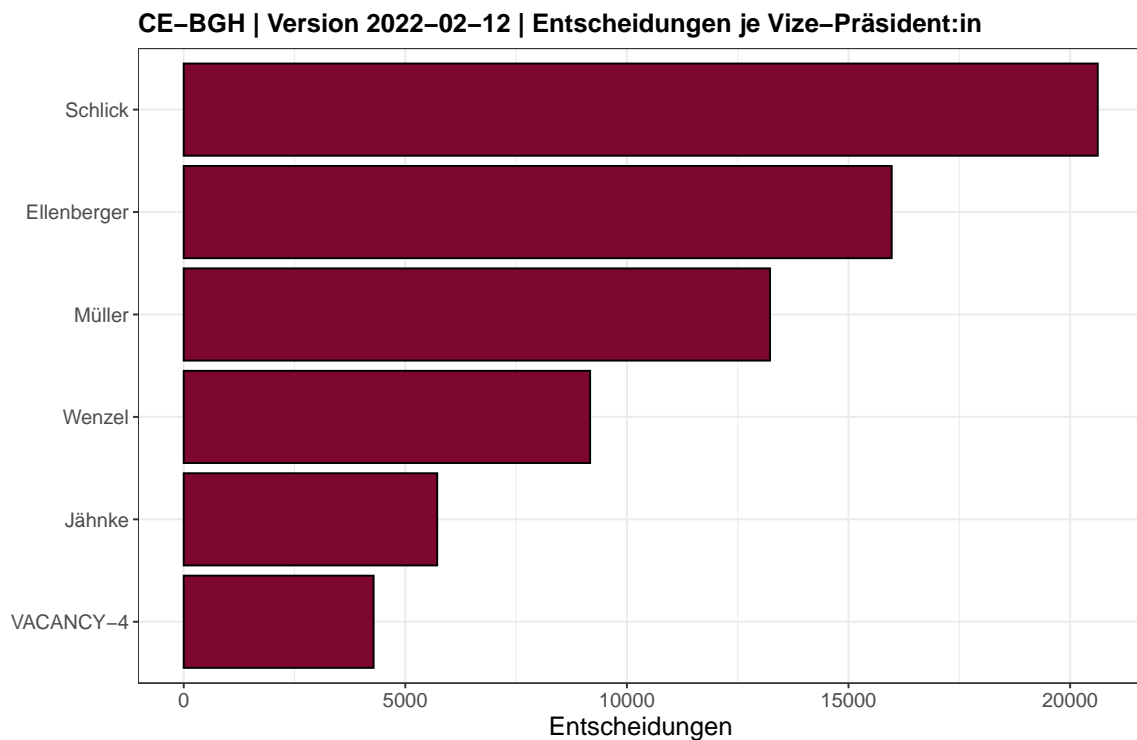


Fobbe | DOI: 10.5281/zenodo.5910135

## 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 = "#7e0731",
           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)
  )
```

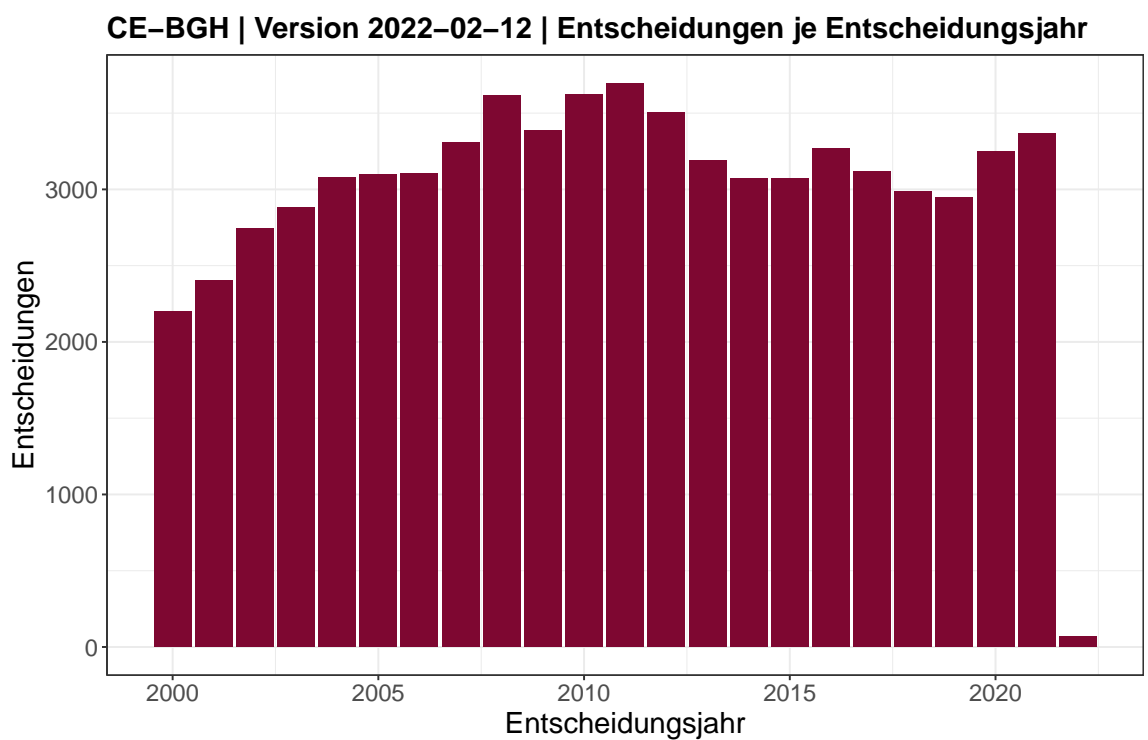


Fobbe | DOI: 10.5281/zenodo.5910135

## 10.9 Diagramm: Entscheidungsjahr

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

```
ggplot(data = freqtable) +  
  geom_bar(aes(x = entscheidungsjahr,  
              y = N),  
          stat = "identity",  
          fill = "#7e0731") +  
  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)  
  )
```

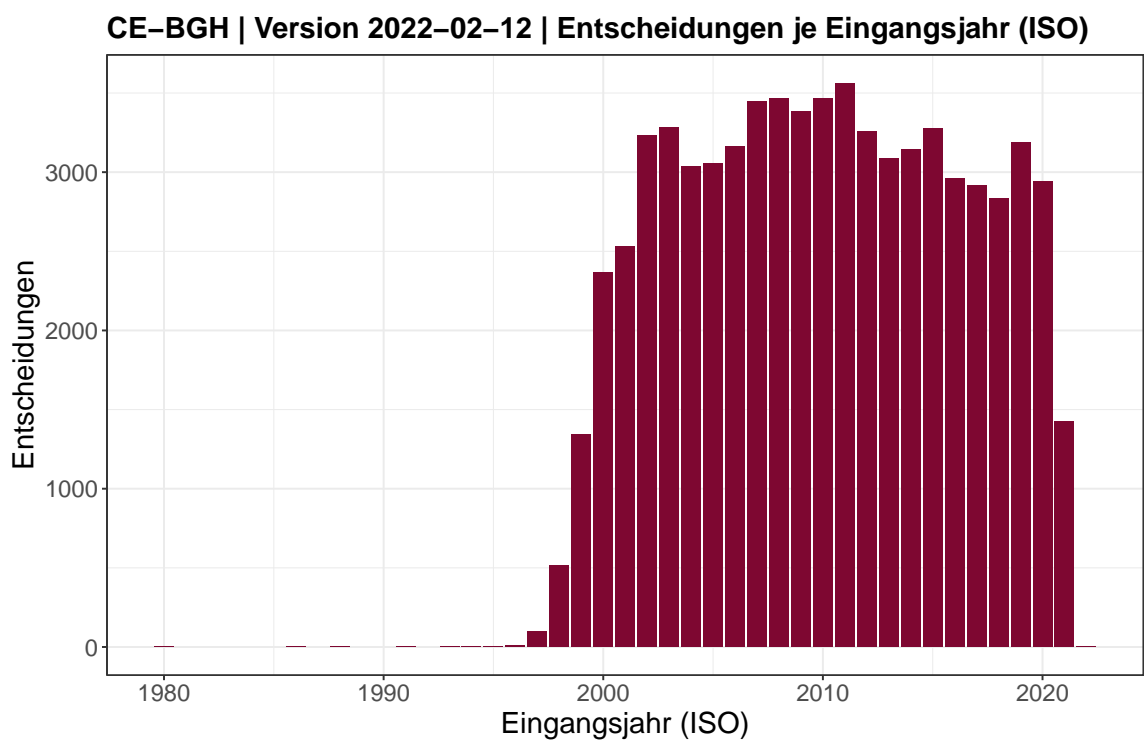


Fobbe | DOI: 10.5281/zenodo.5910135

## 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 = "#7e0731") +  
  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.5910135

## 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,
##                                     chunksperworker = 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 = chunksperworker,
##                                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.bgh)
```

### 11.1.3 Variablen-Namen anpassen

```

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

```

## 11.2 Kennwerte dem Korpus hinzufügen

```

txt.bgh <- cbind(txt.bgh,
  summary.corpus)

```

## 11.3 Variante mit Metadaten erstellen

```

meta.bgh <- txt.bgh[, !"text"]

```

## 11.4 Linguistische Kennwerte

### 11.4.1 Zusammenfassungen berechnen

```

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

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

tokens.temp <- tokens(corpus(txt.bgh),
  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"))

```

#### 11.4.2 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	790,731,070	12	3,029	7,424	11,460.36886	15,985	348,611
tokens	118,931,726	0	428	1,096	1,723.72315	2,406	53,371
typen	695,936	0	214	437	542.47641	771	5,608
saetze	6,315,914	0	29	63	91.53897	126	2,700

#### 11.4.3 Zusammenfassungen speichern

```

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

```

---

## 11.5 Quantitative Variablen

### 11.5.1 Entscheidungsdatum

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

##	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
##	"2000-01-04"	"2006-03-30"	"2011-04-08"	"2011-05-02"	"2016-07-05"	"2022-02-01"

### 11.5.2 Zusammenfassungen berechnen

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

### 11.5.3 Zusammenfassungen anzeigen

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

Variable	Anzahl	Min	Quart1	Median	Mean	Quart3	Max
entscheidungsjahr	23	2000	2006	2011	2010.8338	2016	2022
eingangsjahr_iso	34	1980	2005	2010	2009.9749	2015	2022
eingangsnummer	779	1	55	156	198.2252	302	1304

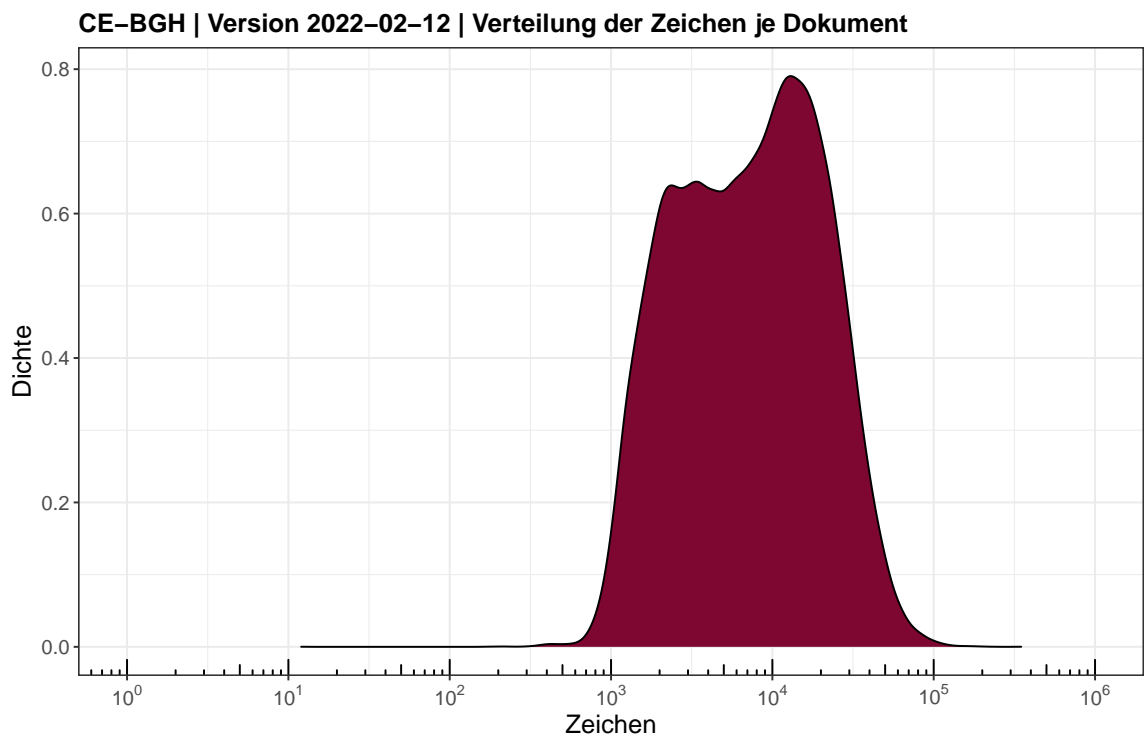
### 11.5.4 Zusammenfassungen speichern

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

## 11.6 Verteilungen linguistischer Kennwerte

### 11.6.1 Diagramm: Verteilung Zeichen

```
ggplot(data = meta.bgh)+
  geom_density(aes(x = zeichen),
    fill = "#7e0731")+
  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)
  )
```



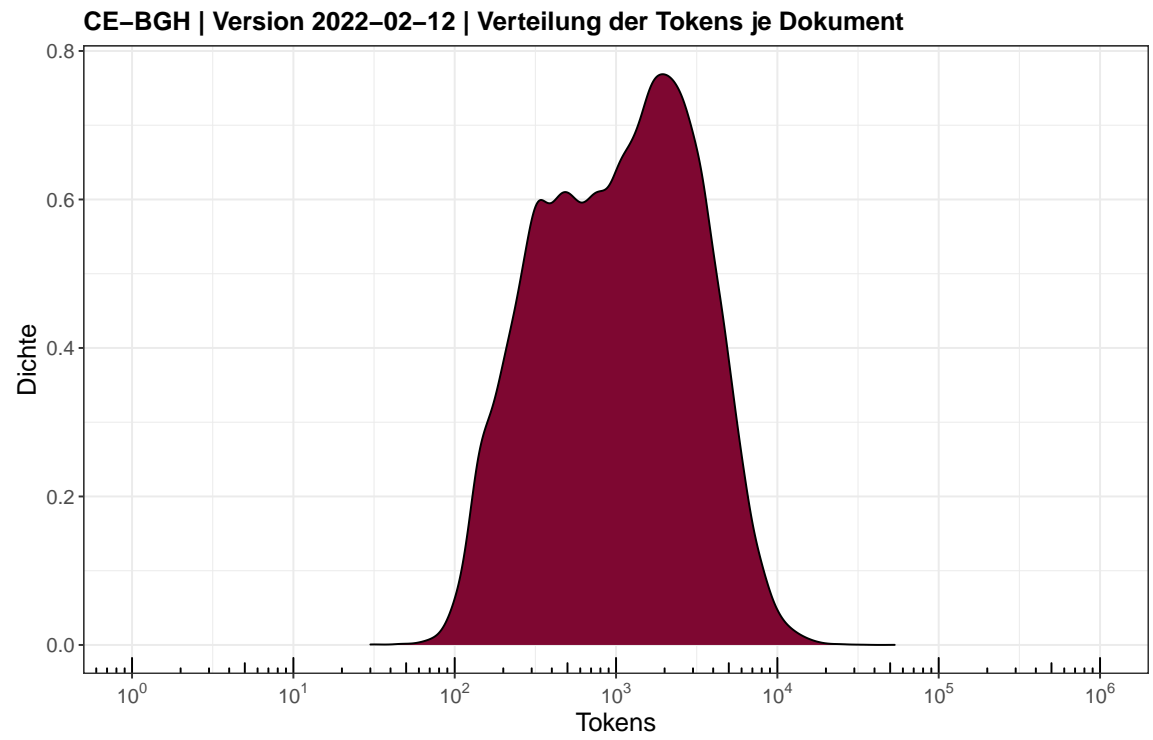
Fobbe | DOI: 10.5281/zenodo.5910135

## 11.6.2 Diagramm: Verteilung Tokens

```
ggplot(data = meta.bgh)+
  geom_density(aes(x = tokens),
               fill = "#7e0731")+
  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)
  )
```

```
## Warning: Transformation introduced infinite values in continuous x-axis
```

```
## Warning: Removed 1 rows containing non-finite values (stat_density).
```



Fobbe | DOI: 10.5281/zenodo.5910135

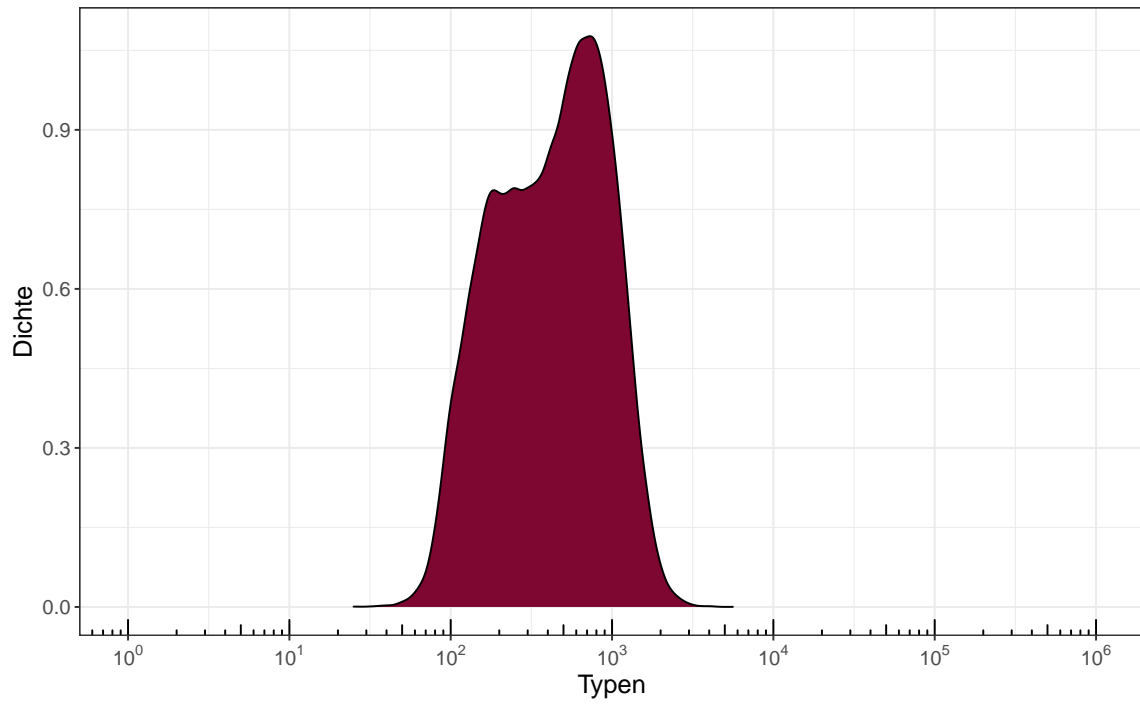
### 11.6.3 Diagramm: Verteilung Typen

```
ggplot(data = meta.bgh)+
  geom_density(aes(x = typen),
    fill = "#7e0731")+
  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)
  )
```

```
## Warning: Transformation introduced infinite values in continuous x-axis
```

```
## Warning: Removed 1 rows containing non-finite values (stat_density).
```





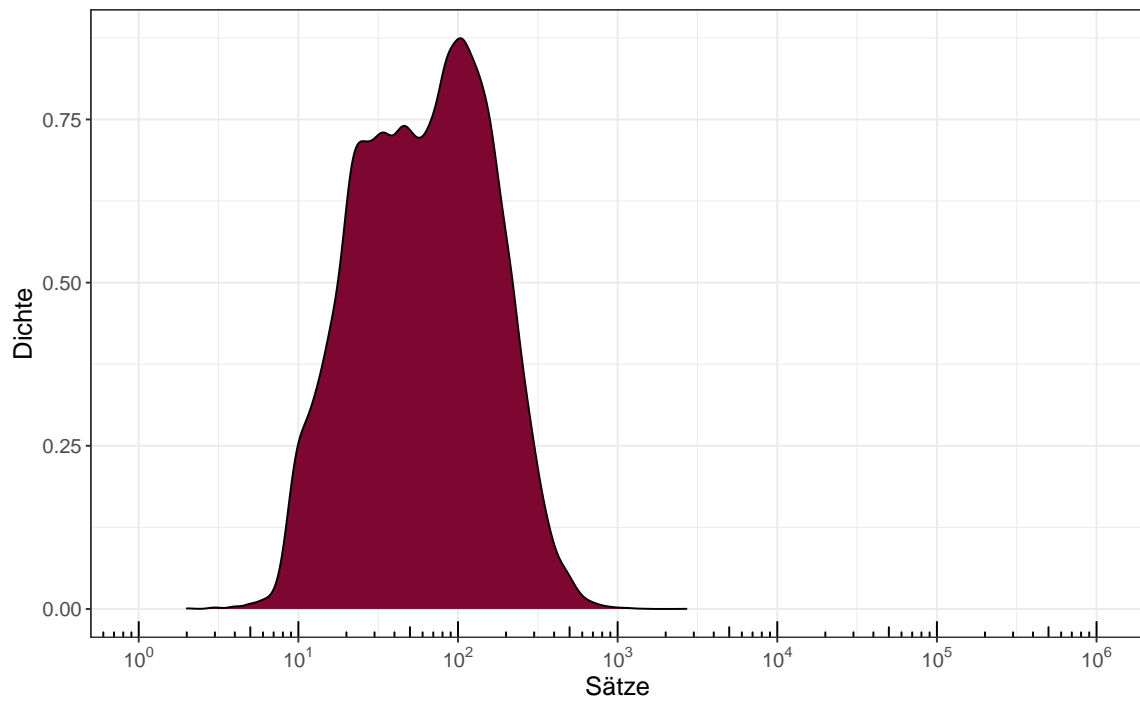
Fobbe | DOI: 10.5281/zenodo.5910135

### 11.6.4 Diagramm: Verteilung Sätze

```
ggplot(data = meta.bgh)+
  geom_density(aes(x = saetze),
               fill = "#7e0731")+
  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)
  )
```

```
## Warning: Transformation introduced infinite values in continuous x-axis
```

```
## Warning: Removed 1 rows containing non-finite values (stat_density).
```



Fobbe | DOI: 10.5281/zenodo.5910135

## 12 Kontrolle der Variablen

### 12.1 Semantische Sortierung der Variablen

#### 12.1.1 Variablen sortieren: Hauptdatensatz

```
setcolorder(txt.bgh,  
  c("doc_id",  
    "text",  
    "gericht",  
    "datum",  
    "entscheidung_typ",  
    "leitsatz",  
    "spruchkoerper_db",  
    "spruchkoerper_az",  
    "registerzeichen",  
    "verfahrensart",  
    "eingangsnummer",  
    "eingangsjahr_az",  
    "eingangsjahr_iso",  
    "entscheidungsjahr",  
    "zusatz_az",  
    "name",  
    "kollision",  
    "praesi",  
    "v_praesi",  
    "aktenzeichen",  
    "bemerkung",  
    "berichtigung",  
    "ecli",  
    "zeichen",  
    "tokens",  
    "typen",  
    "saetze",  
    "version",  
    "doi_concept",  
    "doi_version",  
    "lizenz"))
```

### 12.1.2 Variablen sortieren: Metadaten

```
setcolorder(txt.bgh,  
  c("doc_id",  
    "gericht",  
    "datum",  
    "entscheidung_typ",  
    "leitsatz",  
    "spruchkoerper_db",  
    "spruchkoerper_az",  
    "registerzeichen",  
    "verfahrensart",  
    "eingangsnummer",  
    "eingangsjahr_az",  
    "eingangsjahr_iso",  
    "entscheidungsjahr",  
    "zusatz_az",  
    "name",  
    "kollision",  
    "praesi",  
    "v_praesi",  
    "aktenzeichen",  
    "bemerkung",  
    "berichtigung",  
    "ecli",  
    "zeichen",  
    "tokens",  
    "typen",  
    "saetze",  
    "version",  
    "doi_concept",  
    "doi_version",  
    "lizenz"))
```

## 12.2 Anzahl Variablen der Datensätze

```
length(txt.bgh)
```

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

```
length(meta.bgh)
```

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

## 12.3 Alle Variablen-Namen der Datensätze

```
names(txt.bgh)
```

```
## [1] "doc_id"      "gericht"      "datum"
## [4] "entscheidung_typ" "leitsatz"     "spruchkoerper_db"
## [7] "spruchkoerper_az" "registerzeichen" "verfahrensart"
## [10] "eingangsnummer" "eingangsjahr_az" "eingangsjahr_iso"
## [13] "entscheidungsjahr" "zusatz_az"     "name"
## [16] "kollision"      "praesi"       "v_praesi"
## [19] "aktenzeichen"   "bemerkung"     "berichtigung"
## [22] "ecli"           "zeichen"      "tokens"
## [25] "typen"          "saetze"       "version"
## [28] "doi_concept"    "doi_version"   "lizenz"
## [31] "text"
```

```
names(meta.bgh)
```

```
## [1] "doc_id"      "gericht"      "spruchkoerper_db"
## [4] "leitsatz"     "datum"        "spruchkoerper_az"
## [7] "registerzeichen" "eingangsnummer" "eingangsjahr_az"
## [10] "zusatz_az"     "name"         "kollision"
## [13] "entscheidungsjahr" "eingangsjahr_iso" "praesi"
## [16] "v_praesi"      "verfahrensart" "aktenzeichen"
## [19] "entscheidung_typ" "ecli"         "bemerkung"
## [22] "berichtigung"   "doi_concept"   "doi_version"
## [25] "version"        "lizenz"       "zeichen"
## [28] "tokens"        "typen"        "saetze"
```

## 13 CSV-Dateien erstellen

### 13.1 CSV mit vollem Datensatz speichern

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

### 13.2 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.bgh,  
       csvname.meta,  
       na = "NA")
```

## 14 Dateigrößen analysieren

### 14.1 Gesamtgröße

#### 14.1.1 Korpus-Objekt in RAM (MB)

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

```
## 835.8 MB
```

#### 14.1.2 CSV Korpus (MB)

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

```
## [1] 835.7774
```

#### 14.1.3 CSV Metadaten (MB)

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

```
## [1] 32.94665
```

#### 14.1.4 PDF-Dateien (MB)

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

```
## [1] 6462.305
```



### 14.1.5 TXT-Dateien (MB)

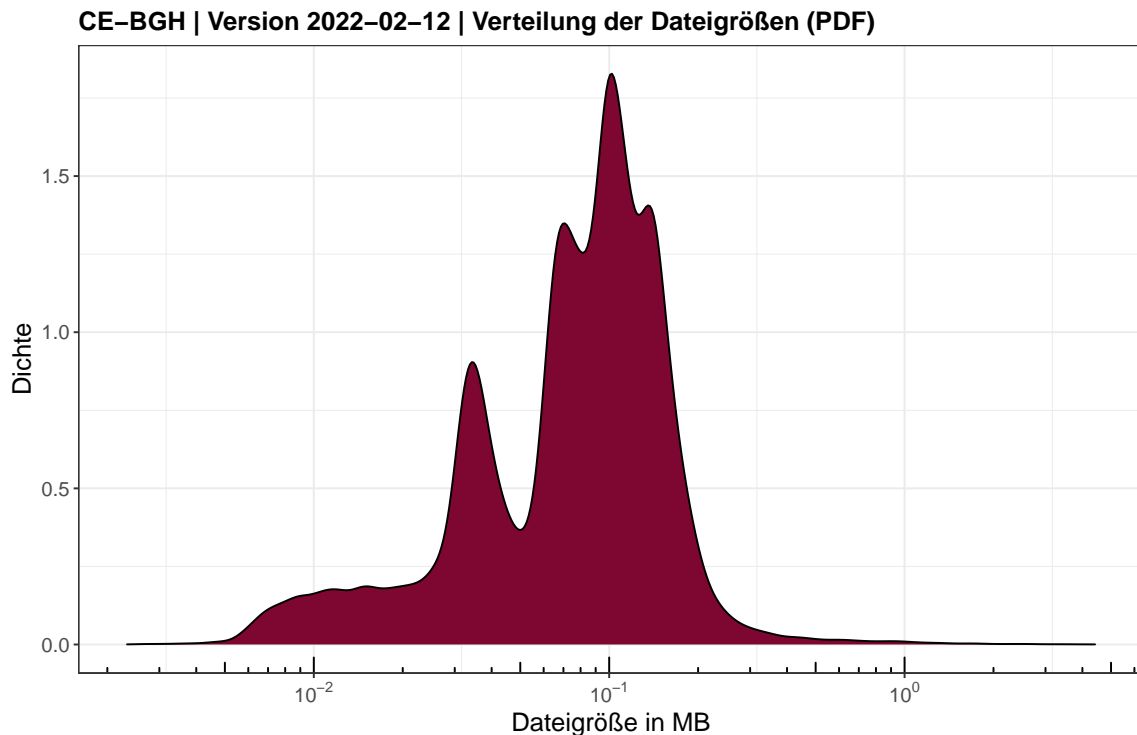
```
files.txt <- list.files(pattern = "\\..txt$",  
                        ignore.case = TRUE)  
  
txt.MB <- file.size(files.txt) / 10^6  
sum(txt.MB)
```

```
## [1] 820.612
```

## 14.2 Diagramm: Verteilung der Dateigrößen (PDF)

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

```
ggplot(data = dt.plot,
  aes(x = pdf.MB)) +
  geom_density(fill = "#7e0731") +
  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)
  )
```

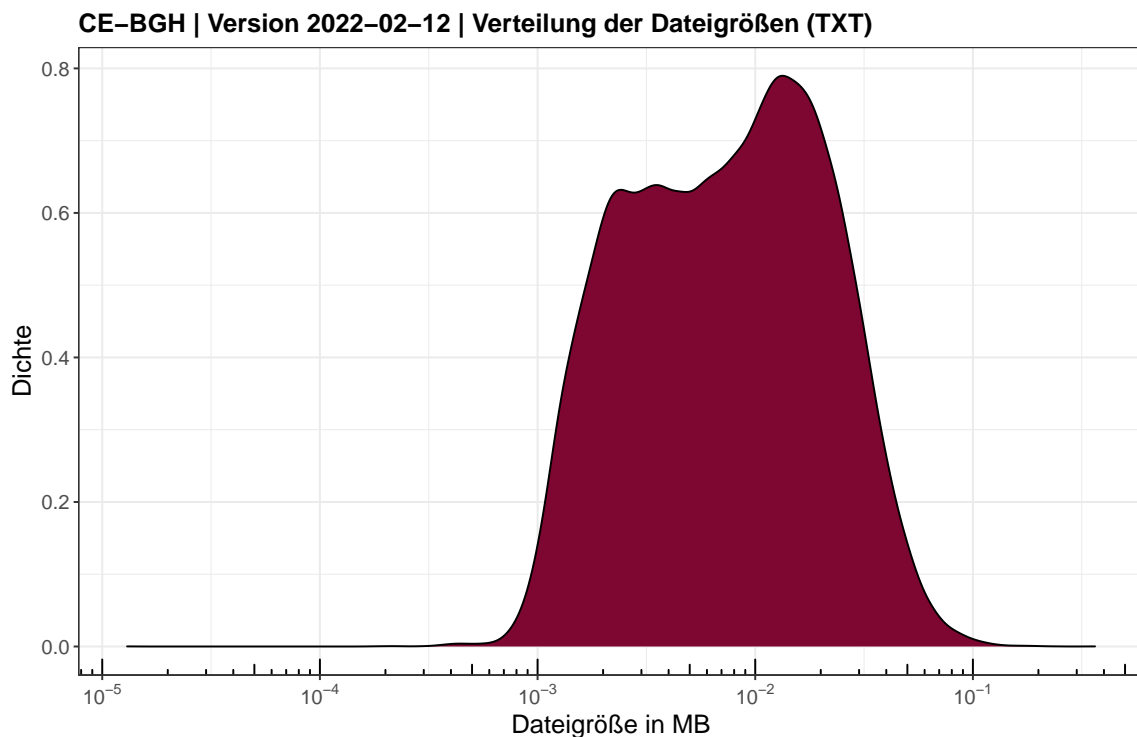


Fobbe | DOI: 10.5281/zenodo.5910135

### 14.3 Diagramm: Verteilung der Dateigrößen (TXT)

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

```
ggplot(data = dt.plot,
  aes(x = txt.MB)) +
  geom_density(fill = "#7e0731") +
  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.5910135

## 15 Erstellen der ZIP-Archive

### 15.1 Verpacken der CSV-Dateien

#### 15.1.1 Vollständiger Datensatz

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

#### 15.1.2 Metadaten

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

### 15.2 Verpacken der PDF-Dateien

#### 15.2.1 Nur Leitsatz-Entscheidungen

```
files.leitsatz <- gsub("\\.txt",  
                    "\\pdf",  
                    txt.bgh[leitsatz == "LE"]$doc_id)
```

```
zip(paste(prefix.files,  
          "DE_PDF_Leitsatz-Entscheidungen.zip",  
          sep = "_"),  
    files.leitsatz)
```

#### 15.2.2 Nur Benannte Entscheidungen

```
files.benannt <- gsub("\\.txt",  
                    "\\pdf",  
                    txt.bgh[is.na(name) == FALSE]$doc_id)
```

```
zip(paste(prefix.files,  
          "DE_PDF_Entscheidungen-mit-Namen.zip",  
          sep = "_"),  
    files.benannt)
```

### 15.2.3 Alle Entscheidungen

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

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

## 15.3 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)
```

## 15.4 Verpacken der Analyse-Dateien

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

## 15.5 Verpacken der Source-Dateien

```
files.source <- c(list.files(pattern = "\\..R$|\\..toml$"),  
                  "CHANGELOG.md",  
                  "README.md",  
                  "R-fobbe-proto-package",  
                  "buttons",  
                  "data",
```

```

        "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)

```

## 16 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.

### 16.1 Liste der ZIP-Archive erstellen

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

### 16.2 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(hashses.table)
##
## }
```

### 16.3 Hashes berechnen

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

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

```
## Processing 8 files. Begin at: 2022-02-12 07:18:54
```

```
## Processed 8 files. Runtime was 25.97 secs.
```

### 16.4 In Data Table umwandeln

```
setDT(multihashes)

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

### 16.5 Index hinzufügen

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



## 16.6 In Datei schreiben

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

## 16.7 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))
```

## 16.8 In Bericht anzeigen

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

index	filename
1	CE-BGH_2022-02-12_DE_ANALYSE.zip
2	CE-BGH_2022-02-12_DE_CSV_Datensatz.zip
3	CE-BGH_2022-02-12_DE_CSV_Metadaten.zip
4	CE-BGH_2022-02-12_DE_PDF_Datensatz.zip
5	CE-BGH_2022-02-12_DE_PDF_Entscheidungen-mit-Namen.zip
6	CE-BGH_2022-02-12_DE_PDF_Leitsatz-Entscheidungen.zip
7	CE-BGH_2022-02-12_DE_TXT_Datensatz.zip
8	CE-BGH_2022-02-12_Source_Files.zip

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

index	sha2.256
1	f9521b574311b7d057f545369f67b4f4981bebec12cde4ff44bd75bd0cf53ec0
2	e694738e9ade60e08aed877e391f4c402ebb9785ce8eb40c747a01b898ad741f
3	9d4513d8a81fe79ff44977fb68e5077aab3b8e11b7e74d481eb03e929ed8ab94
4	8558ef74f333882b588dcf32aac069f25729acaeb23ddb9ffbb1506220553f7c
5	5f8b36ff65fa565f2b54c4bfd3da916c28ac33c7e9d4b0f39749ab1f0e704f21
6	f0a6df242ae0e923286b72dbbb4390894ba7648df6bce5a2c0e71dffcecaf0ea
7	90c1e833896e9d739bfbdb77c48ff80dbba85ad1048b375bc3783015245affbb
8	098d86ec25d89e3065c0011fef942c75ebeaa70bc5ecef043f64cdce7a28d66

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

index	sha3.512
1	754d8208cb8297790ab1d370a56ecd0e554f1cf63c35689b7b3233e87ac62057 30053ceca07698fc6480fcc710189ee4ba2a4de58ac98eb83819841af20dd7dc
2	caecb31ff38a09d59053504f6820a012787d806bb60e2b208f03d1bdc07b1bf1 af1cb40a78f9e8cb2eb44e36b7764960049b1d18e2099be2173d7d8a24b289df
3	c3409d9337749ad5b8810f8b7122f8acd98665d9c1684d3745599318c9a26f6c 9acb360b95fe9ae3c0c10d2eed75cd50cc9b90e445032f8763b04538fef848f1
4	1055a107c3c7daac0a66519dc639b92fff4956d71378638ceb1796c8f8db9e06 ff4520e91c57e2f57144c8ffa03eab49c17c98f34672a03aa1e3284a2417d638
5	51fe82533687359c529f76c6868a23654cf18d3e81cf2b42dc0e5e17543a7094 feefd1830b73938a2d41a9dc24ad3fba69ba67c36810d13e5ee9a68fe6cd5e60
6	3ece2d43b62d6c4592243b4e52dd7500efb15c3d0d2d890340a7cbdf4e449e12 bf9b2f6b4da9e80a3aa7071fa4b5124f398b0f11d13833a563dcc51c0f1682d3

- 7      6f976db5f8ac7b0efacb9aff155945739f47d31158e7d220cf7cfc87eb23287f  
8e4755fdabc6f5802ccd7a453f7d30f294079c91413e4f55974019cd7279b16a
  - 8      2f11a53bdf5a1f622d4e964da8a17f5245ab89ce61898884e073032642ae117  
1eab7eb81dd4c52342246d51b8bc48e148e0aca05af484ae4515e3ffa400b764
-

## 17 Aufräumen

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

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

print(files.output)
```

```
## [1] "CE-BGH_2022-02-12_DE_ANALYSE.zip"
## [2] "CE-BGH_2022-02-12_DE_CSV_Datensatz.zip"
## [3] "CE-BGH_2022-02-12_DE_CSV_Metadaten.zip"
## [4] "CE-BGH_2022-02-12_DE_PDF_Datensatz.zip"
## [5] "CE-BGH_2022-02-12_DE_PDF_Entscheidungen-mit-Namen.zip"
## [6] "CE-BGH_2022-02-12_DE_PDF_Leitsatz-Entscheidungen.zip"
## [7] "CE-BGH_2022-02-12_DE_TXT_Datensatz.zip"
## [8] "CE-BGH_2022-02-12_Source_Files.zip"
```

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

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

## 18 Abschluss

### 18.1 Datumsstempel

```
print(datestamp)
```

```
## [1] "2022-02-12"
```

### 18.2 Datum und Uhrzeit (Anfang)

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

```
## [1] "2022-02-12 00:14:54 CET"
```

### 18.3 Datum und Uhrzeit (Ende)

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

```
## [1] "2022-02-12 07:19:20 CET"
```

### 18.4 Laufzeit des gesamten Skriptes

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

```
## Time difference of 7.073835 hours
```

### 18.5 Warnungen

```
warnings()
```

## 19 Parameter für strenge Replikationen

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

```
## [1] "OpenSSL 1.1.1l FIPS 24 Aug 2021"
```

```
sessionInfo()
```

```
## R version 4.0.5 (2021-03-31)
## Platform: x86_64-redhat-linux-gnu (64-bit)
## Running under: Fedora 34 (Workstation Edition)
##
## Matrix products: default
## BLAS/LAPACK: /usr/lib64/libflexiblas.so.3.1
##
## locale:
##  [1] LC_CTYPE=en_US.utf8      LC_NUMERIC=C
##  [3] LC_TIME=en_US.utf8      LC_COLLATE=en_US.utf8
##  [5] LC_MONETARY=en_US.utf8  LC_MESSAGES=en_US.utf8
##  [7] LC_PAPER=en_US.utf8     LC_NAME=C
##  [9] LC_ADDRESS=C            LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_US.utf8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] stats      graphics  grDevices  utils      datasets  methods   base
##
## other attached packages:
##  [1] future.apply_1.8.1 future_1.23.0   spacyr_1.2.1   quanteda_3.2.0
##  [5] readtext_0.81      data.table_1.14.2 scales_1.1.1   ggplot2_3.3.5
##  [9] pdftools_3.0.1     kableExtra_1.3.4 knitr_1.37     rvest_1.0.2
## [13] httr_1.4.2         mgsub_1.7.3    RcppTOML_0.1.7 fs_1.5.2
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_1.0.8          svglite_2.0.0    lattice_0.20-45 listenr_0.8.0
##  [5] digest_0.6.29      utf8_1.2.2       parallelly_1.30.0 R6_2.5.1
##  [9] evaluate_0.14       highr_0.9        pillar_1.6.5    rlang_0.4.12
## [13] curl_4.3.2          rstudioapi_0.13  magick_2.7.3    Matrix_1.4-0
## [17] rmarkdown_2.11      qpdf_1.1         labeling_0.4.2  webshot_0.5.2
## [21] stringr_1.4.0       selectr_0.4-2    munsell_0.5.0   compiler_4.0.5
## [25] xfun_0.29           pkgconfig_2.0.3  askpass_1.1     systemfonts
## [29] globals_0.14.0     htmltools_0.5.2  tibble_3.1.6    codetools_0.2-18
## [33] fansi_1.0.2         viridisLite_0.4.0 crayon_1.4.2    withr_2.4.3
## [37] grid_4.0.5          gtable_0.3.0     lifecycle_1.0.1 magrittr_2.0.1
## [41] RcppParallel_5.1.5 stringi_1.7.6     farver_2.1.0    renv_0.15.2
## [45] xml2_1.3.3          ellipsis_0.3.2   stopwords_2.3    vctrs_0.3.8
## [49] fastmatch_1.1-3     tools_4.0.5      glue_1.6.1      parallel_4.0.5
## [53] fastmap_1.1.0       yaml_2.2.2       colorspace_2.0-2
```

## Literaturverzeichnis

- Bengtsson, Henrik. 2021a. “A Unifying Framework for Parallel and Distributed Processing in R Using Futures.” <https://journal.r-project.org/archive/2021/RJ-2021-048/index.html>.
- . 2021b. “A Unifying Framework for Parallel and Distributed Processing in R Using Futures.” <https://journal.r-project.org/archive/2021/RJ-2021-048/index.html>.
- . 2021c. *Future.apply: Apply Function to Elements in Parallel Using Futures*. <https://CRAN.R-project.org/package=future.apply>.
- . 2021d. *Future: Unified Parallel and Distributed Processing in R for Everyone*. <https://CRAN.R-project.org/package=future>.
- Benoit, Kenneth, and Akitaka Matsuo. 2020. *Spacyr: Wrapper to the spaCy 'Nlp' Library*. <https://spacyr.quanteda.io>.
- Benoit, Kenneth, and Adam Obeng. 2021. *Readtext: Import and Handling for Plain and Formatted Text Files*. <https://github.com/quanteda/readtext>.
- Benoit, Kenneth, Kohei Watanabe, Haiyan Wang, Paul Nulty, Adam Obeng, Stefan Müller, and Akitaka Matsuo. 2018. “Quanteda: An R Package for the Quantitative Analysis of Textual Data.” *Journal of Open Source Software* 3 (30): 774. <https://doi.org/10.21105/joss.00774>.
- Benoit, Kenneth, Kohei Watanabe, Haiyan Wang, Paul Nulty, Adam Obeng, Stefan Müller, Akitaka Matsuo, and William Lowe. 2021. *Quanteda: Quantitative Analysis of Textual Data*. <https://quanteda.io>.
- Dowle, Matt, and Arun Srinivasan. 2021. *Data.table: Extension of 'Data.frame'*. <https://CRAN.R-project.org/package=data.table>.
- Eddelbuettel, Dirk. 2020. *RcppTOML: Rcpp Bindings to Parser for Tom's Obvious Markup Language*. <http://dirk.eddelbuettel.com/code/rcpp.toml.html>.
- Ewing, Mark. 2021. *Mgsub: Safe, Multiple, Simultaneous String Substitution*. <https://CRAN.R-project.org/package=mgsub>.
- Hester, Jim, Hadley Wickham, and Gábor Csárdi. 2021. *Fs: Cross-Platform File System Operations Based on Libuv*. <https://CRAN.R-project.org/package=fs>.
- Ooms, Jeroen. 2021. *Pdftools: Text Extraction, Rendering and Converting of Pdf Documents*. <https://CRAN.R-project.org/package=pdftools>.
- R Core Team. 2021. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. <https://www.R-project.org/>.
- Wickham, Hadley. 2016. *Ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. <https://ggplot2.tidyverse.org>.
- . 2020. *Httr: Tools for Working with Urls and Http*. <https://CRAN.R-project.org/package=httr>.
- . 2021. *Rvest: Easily Harvest (Scrape) Web Pages*. <https://CRAN.R-project.org/package=rvest>.
- Wickham, Hadley, Winston Chang, Lionel Henry, Thomas Lin Pedersen, Kohske Takahashi, Claus Wilke, Kara Woo, Hiroaki Yutani, and Dewey Dunnington. 2021. *Ggplot2:*

- Create Elegant Data Visualisations Using the Grammar of Graphics*. <https://CRAN.R-project.org/package=ggplot2>.
- Wickham, Hadley, and Dana Seidel. 2020. *Scales: Scale Functions for Visualization*. <https://CRAN.R-project.org/package=scales>.
- Xie, Yihui. 2014. “Knitr: A Comprehensive Tool for Reproducible Research in R.” In *Implementing Reproducible Computational Research*, edited by Victoria Stodden, Friedrich Leisch, and Roger D. Peng. Chapman; Hall/CRC. <http://www.crcpress.com/product/isbn/9781466561595>.
- . 2015. *Dynamic Documents with R and Knitr*. 2nd ed. Boca Raton, Florida: Chapman; Hall/CRC. <https://yihui.org/knitr/>.
- . 2021. *Knitr: A General-Purpose Package for Dynamic Report Generation in R*. <https://yihui.org/knitr/>.
- Zhu, Hao. 2021. *KableExtra: Construct Complex Table with Kable and Pipe Syntax*. <https://CRAN.R-project.org/package=kableExtra>.