



# **Identifikation typischen Benutzerverhaltens in digitalen Studienformaten**

Bachelorarbeit zur Erlangung des akademischen Grades Bachelor of Science  
Berliner Hochschule für Technik · Fachbereich VI · Informatik und Medien

**AUTOR**  
Werner Breitenstein  
Matrikelnr.: 866059

**BETREUER**  
Prof. Dr. Petra Sauer

**GUTACHTER**  
Prof. Dr. Heike Ripphausen-Lipa

**ABGABE**  
08.08.2022

Zur besseren Lesbarkeit und Verständlichkeit wird in dieser Arbeit bewusst auf die Verwendung geschlechtsneutraler Formulierungen verzichtet. Sämtliche Schreibweisen in maskuliner Form beziehen sich stets gleichermaßen auf alle Geschlechter.

# **Inhaltsverzeichnis**

<b>1. Einleitung</b>	<b>11</b>
<b>2. Grundlagen</b>	<b>12</b>
2.1. Theorie . . . . .	12
2.1.1. Standardisierte Vorgehensmodelle der Datenanalyse . . . . .	13
2.1.2. Angepasstes Vorgehensmodell für diese Arbeit . . . . .	16
2.1.3. Explorative Datenanalyse . . . . .	19
2.1.4. Formen der Datenvisualisierung . . . . .	19
2.2. Technik . . . . .	22
2.3. Datenbasis . . . . .	23
2.3.1. Beschreibung der Daten . . . . .	24
2.3.2. Visualisierung der Daten . . . . .	30
<b>3. Analysen</b>	<b>34</b>
3.1. Identifikation von Studenten . . . . .	34
3.1.1. Ermittlung des Benutzerstatus . . . . .	35
3.1.2. Kennzeichnung des Benutzerstatus . . . . .	43
3.1.3. Zusammenfassung . . . . .	45
3.2. Konkretisierung der zu untersuchenden Datenbasis . . . . .	45
3.2.1. Betrachtung von viewed-Action und viewed-Events . . . . .	45
3.2.2. Entscheidung für viewed-Events als Grundlage . . . . .	47
3.3. Lokalität des Lern- und Kommunikationsverhaltens . . . . .	48
3.3.1. Betrachtung des studentischen Verhaltens auf Wochenbasis .	49
3.3.2. Kategorisierung nach Tagestypen . . . . .	51
3.3.3. Betrachtung des studentischen Verhaltens auf Tagesbasis .	53
3.3.4. Kategorisierung nach Tageszeiten . . . . .	55
3.3.5. Vergleich des Lern- und Kommunikationsverhaltens . . . .	58
3.3.6. Zusammenfassung . . . . .	60
3.4. Kontinuität des Lern- und Kommunikationsverhaltens . . . . .	61
3.4.1. Betrachtung des studentischen Verhaltens im Gesamtzeitraum	61
3.4.2. Ermittlung der Vergleichsgöße . . . . .	64
3.4.3. Kategorisierung nach IKK . . . . .	68
3.4.4. Vergleich des Lern- und Kommunikationsverhaltens . . . .	70
3.4.5. Zusammenfassung . . . . .	71
3.5. Dynamik des Lern- und Kommunikationsverhaltens . . . . .	72
3.5.1. Betrachtung des studentischen Verhaltens im Gesamtzeitraum	72
3.5.2. Ermittlung der Vergleichsgöße . . . . .	73
3.5.3. Kategorisierung nach IDK . . . . .	75
3.5.4. Vergleich des Lern- und Kommunikationsverhaltens . . . .	77
3.5.5. Zusammenfassung . . . . .	78
<b>4. Ergebnisse</b>	<b>80</b>
<b>5. Ausblick</b>	<b>84</b>
<b>Literaturverzeichnis</b>	<b>86</b>

<b>A. Anhang</b>	<b>88</b>
A.2. Grundlagen . . . . .	88
A.3. Analysen . . . . .	92
<b>Erklärung zur Urheberschaft</b>	<b>131</b>
<b>Inhalt des beigefügten Datenträgers</b>	<b>132</b>

## Abbildungsverzeichnis

1.	Phasen des KDD-Prozesses. Original von Fayyad et al. (1996). . . . .	14
2.	Phasen des CRISP-DM. Original von Shearer (2000). . . . .	15
3.	KDD, SEMMA, CRISP-DM. Original von Azevedo & Santos (2008). .	16
4.	Phasen des verwendeten Vorgehensmodells . . . . .	18
5.	Beispiel für ein Histogramm mit 16 Klassen . . . . .	20
6.	Beispiel für ein Säulendiagramm . . . . .	21
7.	Beispiel für einen Boxplot . . . . .	21
8.	Systematischer Aufbau der Bibliotheken . . . . .	23
9.	Struktur und Art der importierten Originaldaten . . . . .	25
10.	Menge aller Benutzer . . . . .	27
11.	Menge der Log-Einträge pro Benutzer . . . . .	27
12.	Menge der Benutzer pro Studiengang . . . . .	27
13.	Menge der Kurse pro Benutzer ( <a href="#">s. Anhang</a> ) . . . . .	28
14.	Benutzer mit überdurchschnittlich vielen Kursen . . . . .	29
15.	Menge der Studiengänge 1 bis 4 pro Benutzer . . . . .	29
16.	Menge der Log-Einträge pro Benutzer ( <a href="#">s. Anhang</a> ) . . . . .	31
17.	Menge der Benutzer pro Studiengang . . . . .	32
18.	Menge der Kurse pro Benutzer ( <a href="#">s. Anhang</a> ) . . . . .	33
19.	Mengen aller Actions in der Gesamtbetrachtung ( <a href="#">s. Anhang</a> ) . . . . .	36
20.	Menge der viewed-Actions pro Benutzer ( <a href="#">s. Anhang</a> ) . . . . .	37
21.	Anteil der viewed-Actions an der Gesamtaktivität ( <a href="#">s. Anhang</a> ) . . .	38
22.	Kombiniertes Datenset für Studenten und Andere ( <a href="#">s. Anhang</a> ) . . .	40
23.	Menge der Log-Einträge pro Aktivität und Benutzergruppe . . . . .	41
24.	Identifikation von Studenten ( <a href="#">s. Anhang</a> ) . . . . .	43
25.	Überprüfung der Änderungen auf Vollständigkeit . . . . .	44
26.	Überprüfung der Änderungen auf Richtigkeit . . . . .	44
27.	Ermittlung korrespondierender viewed-Events . . . . .	47
28.	Ermittlung korrespondierender sent-Events . . . . .	48
29.	Verteilung der Log-Einträge pro Wochentag . . . . .	50
30.	Verteilung der Log-Einträge über die Wochentage . . . . .	50
31.	Erstellung des neuen Datensets <i>loggings_daytype</i> ( <a href="#">s. Anhang</a> ) . . . .	52
32.	Anteilige Mengen an Log-Einträgen pro Tagestyp ( <a href="#">s. Anhang</a> ) . . .	52
33.	Typisierung der Studenten nach Tagestyp ( <a href="#">s. Anhang</a> ) . . . . .	53
34.	Verteilung der Log-Einträge pro Tagesstunde ( <a href="#">s. Anhang</a> ) . . . .	54
35.	Verteilung der Log-Einträge über die Tagesstunden ( <a href="#">s. Anhang</a> ) . .	55
36.	Erstellung des neuen Datensets <i>loggings_daytime_1</i> ( <a href="#">s. Anhang</a> ) . .	56
37.	Anteilige Mengen an Log-Einträgen pro Tageszeit ( <a href="#">s. Anhang</a> ) . . .	56
38.	Typisierung der Studenten nach Tageszeit ( <a href="#">s. Anhang</a> ) . . . . .	57
39.	Typisierung nach Tagestyp, Lernverhalten ( <a href="#">s. Anhang</a> ) . . . . .	59
40.	Typisierung nach Tagestyp, Kommunikationsverhalten ( <a href="#">s. Anhang</a> ) .	59
41.	Typisierung nach Tageszeit, Lernverhalten ( <a href="#">s. Anhang</a> ) . . . . .	59
42.	Typisierung nach Tageszeit, Kommunikationsverhalten ( <a href="#">s. Anhang</a> )	60
43.	Verteilung der Log-Einträge pro Tag ( <a href="#">s. Anhang</a> ) . . . . .	61
44.	Verteilung der Log-Einträge pro Tag (Ausschnitt) . . . . .	62
45.	Menge der Arbeitswochen pro Student ( <a href="#">s. Anhang</a> ) . . . . .	63

46.	Menge der Arbeitstage pro Student ( <a href="#">s. Anhang</a> ) . . . . .	64
47.	Datenset <i>time_rel_con</i> zur Kontinuitätsanalyse ( <a href="#">s. Anhang</a> ) . . . . .	66
48.	Individueller Kontinuitätskoeffizient, IKK ( <a href="#">s. Anhang</a> ) . . . . .	67
49.	Typisierung der Studenten nach IKK ( <a href="#">s. Anhang</a> ) . . . . .	69
50.	Menge der Log-Einträge für einzelne Studenten ( <a href="#">s. Anhang</a> ) . . . . .	69
51.	Typisierung nach Kontinuität, Lernverhalten ( <a href="#">s. Anhang</a> ) . . . . .	70
52.	Typisierung nach Kontinuität, Kommunikationsverh. ( <a href="#">s. Anhang</a> ) . . . . .	70
53.	Menge der Log-Einträge für einzelne Studenten ( <a href="#">s. Anhang</a> ) . . . . .	72
54.	Datenset <i>time_rel_dyn</i> zur Dynamikanalyse ( <a href="#">s. Anhang</a> ) . . . . .	74
55.	Individueller Dynamikkoeffizient, IDK ( <a href="#">s. Anhang</a> ) . . . . .	75
56.	Typisierung der Studenten nach IDK ( <a href="#">s. Anhang</a> ) . . . . .	77
57.	Menge der Log-Einträge für einzelne Studenten ( <a href="#">s. Anhang</a> ) . . . . .	77
58.	Typisierung nach Dynamik, Lernverhalten ( <a href="#">s. Anhang</a> ) . . . . .	78
59.	Typisierung nach Dynamik, Kommunikationsverh. ( <a href="#">s. Anhang</a> ) . . . . .	78
60.	Individueller Kontinuitätskoeffizient, IKK ( <a href="#">s. Anhang</a> ) . . . . .	81
61.	Individueller Dynamikkoeffizient, IDK ( <a href="#">s. Anhang</a> ) . . . . .	82
62.	Menge der Kurse pro Benutzer . . . . .	88
63.	Menge der Log-Einträge pro Benutzer . . . . .	90
64.	Menge der Kurse pro Benutzer . . . . .	91
65.	Mengenverteilung aller Actions in der Gesamtbetrachtung . . . . .	93
66.	Menge der viewed-Actions pro Benutzer . . . . .	94
67.	Anteil der viewed-Actions an der Gesamtaktivität . . . . .	95
68.	Kombiniertes Datenset für Studenten und Andere . . . . .	95
69.	Identifikation von Studenten . . . . .	96
70.	Erstellung des neuen Datensets <i>loggings_daytype</i> . . . . .	98
71.	Anteilige Mengen an Log-Einträgen pro Student und Tagestyp . . . . .	99
72.	Typisierung nach Tagestyp . . . . .	100
73.	Verteilung der Log-Einträge pro Tagesstunde . . . . .	101
74.	Verteilung der Log-Einträge über die Tagesstunden . . . . .	102
75.	Erstellung des neuen Datensets <i>loggings_daytime_1</i> . . . . .	103
76.	Anteilige Mengen an Log-Einträgen pro Tageszeit . . . . .	104
77.	Typisierung nach Tageszeit . . . . .	105
78.	Typisierung nach Tagestyp und Lernverhalten . . . . .	106
79.	Typisierung nach Tagestyp und Kommunikationsverhalten . . . . .	107
80.	Typisierung nach Tageszeit und Lernverhalten . . . . .	108
81.	Typisierung nach Tageszeit und Kommunikationsverhalten . . . . .	109
82.	Verteilung der Log-Einträge im Gesamtzeitraum pro Tag . . . . .	111
83.	Menge der Log-Einträge im Gesamtzeitraum pro Student . . . . .	112
84.	Menge der Arbeitswochen im Gesamtzeitraum pro Student . . . . .	113
85.	Menge der Arbeitstage im Gesamtzeitraum pro Student . . . . .	114
86.	Datenset <i>time_rel_con</i> zur Kontinuitätsanalyse . . . . .	115
87.	Individueller Kontinuitätskoeffizient, IKK . . . . .	116
88.	Typisierung nach IKK mit Bezug auf IKK . . . . .	117
89.	Typisierung nach IKK mit Bezug auf Log-Einträge . . . . .	118
90.	Menge der Log-Einträge für einzelne Studenten . . . . .	119
91.	Menge der Log-Einträge pro Student im Gesamtzeitraum nach Tagen	120
92.	Typisierung nach Kontinuität, Lernverhalten . . . . .	121
93.	Typisierung nach Kontinuität, Kommunikationsverhalten . . . . .	122
94.	Menge der Log-Einträge für einzelne Studenten . . . . .	124

95.	Datenset <i>time_rel_dyn</i> zur Dynamikanalyse . . . . .	125
96.	Individueller Dynamikkoeffizient, IDK . . . . .	126
97.	Typisierung der Studenten nach IDK . . . . .	127
98.	Menge der Log-Einträge für einzelne Studenten . . . . .	128
99.	Typisierung nach Dynamik, Lernverhalten . . . . .	129
100.	Typisierung nach Dynamik, Kommunikationsverhalten . . . . .	130

## Quellcodeverzeichnis

1.	Abfrage zu Struktur und Art der importierten Originaldaten . . . . .	25
2.	Abfrage zur Menge aller Benutzer . . . . .	25
3.	Abfrage zur Menge der Log-Einträge pro Benutzer . . . . .	27
4.	Abfrage zur Menge der Benutzer pro Studiengang . . . . .	27
5.	Abfrage zur Menge der Kurse pro Benutzer . . . . .	28
6.	Abfrage zu Benutzern mit überdurchschnittlich vielen Kursen . . . . .	28
7.	Abfrage zur Menge der Studiengänge 1 bis 4 pro Benutzer . . . . .	29
8.	Auswahl der Arbeitsdaten . . . . .	30
9.	Menge der Log-Einträge pro Benutzer . . . . .	30
10.	Menge der Benutzer pro Studiengang . . . . .	32
11.	Menge der Kurse pro Benutzer . . . . .	33
12.	Mengen aller Actions in der Gesamtbetrachtung . . . . .	36
13.	Menge der viewed-Actions pro Benutzer . . . . .	37
14.	Anteil der viewed-Actions an der Gesamtaktivität . . . . .	38
15.	Auswahl der Log-Einträge der Studenten . . . . .	39
16.	Auswahl der Log-Einträge der Anderen . . . . .	40
17.	Konkatenation der Datensets von Studenten und Anderen . . . . .	40
18.	Menge der Log-Einträge pro Aktivität und Benutzergruppe . . . . .	40
19.	Identifikation von Studenten . . . . .	42
20.	Erstellen der neuen Tabelle moodle_data_students . . . . .	43
21.	Kennzeichnung von Studenten . . . . .	44
22.	Überprüfung der Änderungen auf Vollständigkeit . . . . .	44
23.	Überprüfung der Änderungen auf Richtigkeit . . . . .	44
24.	Ermittlung korrespondierender viewed-Events . . . . .	46
25.	Ermittlung korrespondierender sent-Events . . . . .	48
26.	Ergänzung des Merkmals <i>behaviour</i> . . . . .	49
27.	Definition der Arbeitsdaten . . . . .	49
28.	Wochentag pro Log-Eintrag . . . . .	50
29.	Verteilung der Log-Einträge pro Wochentag . . . . .	50
30.	Verteilung der Log-Einträge über die Wochentage . . . . .	50
31.	Erstellung des neuen Datensets <i>loggings_daytype</i> . . . . .	51
32.	Anteilige Mengen an Log-Einträgen pro Tagestyp . . . . .	52
33.	Typisierung der Studenten nach Tagestyp . . . . .	53
34.	Typisierung der Studenten nach Tagestyp . . . . .	53
35.	Tagesstunde pro Log-Eintrag . . . . .	54
36.	Verteilung der Log-Einträge pro Tagesstunde . . . . .	54
37.	Verteilung der Log-Einträge über die Tagesstunden . . . . .	54
38.	Erstellung des neuen Datensets <i>loggings_daytime_1</i> . . . . .	55
39.	Anteilige Mengen an Log-Einträgen pro Tageszeit . . . . .	56
40.	Typisierung der Studenten nach Tageszeit . . . . .	57
41.	Typisierung der Studenten nach Tageszeit . . . . .	57
42.	Definition der Arbeitsdaten, Lernverhalten . . . . .	58
43.	Definition der Arbeitsdaten, Kommunikationsverhalten . . . . .	58
44.	Verteilung der Log-Einträge im Gesamtzeitraum pro Tag . . . . .	61

45.	Menge der Arbeitswochen pro Student . . . . .	62
46.	Menge der Arbeitswochen pro Student . . . . .	63
47.	Menge der Arbeitstage pro Student . . . . .	63
48.	Menge der Arbeitstage pro Student . . . . .	64
49.	Erstellung des neuen Datensets zur Kontinuitätsanalyse . . . . .	65
50.	Ermittlung der Arbeitswochen im Toleranzbereich . . . . .	66
51.	Ermittlung der Arbeitstage im Toleranzbereich . . . . .	67
52.	Ergänzung des IKK im Datenset <i>time_rel_con</i> . . . . .	68
53.	Typisierung der Studenten nach der Kontinuität der Aktivitäten . . . . .	68
54.	Typisierung der Studenten nach IKK . . . . .	69
55.	Vorbereitung der Arbeitsdaten . . . . .	73
56.	Erstellung des Datensets zur Aufnahme individueller Kennziffern . . . . .	74
57.	Ermittlung des individuellen Dynamikkoeffizienten, IDK . . . . .	75
58.	Typisierung der Studenten nach der Dynamik der Aktivitäten . . . . .	76
59.	Typisierung der Studenten nach IDK . . . . .	76
60.	Import von Bibliotheken und anderen Erweiterungen . . . . .	89
61.	Definitionen zur Darstellung der Visualisierungen . . . . .	89
62.	Herstellung der Verbindung zur MySQL-Datenbank . . . . .	89
63.	Import der Arbeitsdaten aus der MySQL-Datenbank . . . . .	89

## **Zusammenfassung**

...

## **Abstract**

...

## 1. Einleitung

*Ziel- und Endpunkt der Arbeit ist die detaillierte Analyse und Dokumentation des IST-Zustands. Es werden weder Prognosen abgeleitet noch Empfehlungen gegeben.*

...

## 2. Grundlagen

In diesem Kapitel werden die theoretischen Grundlagen dieser Arbeit beleuchtet und mithin wichtige Informationen zur angewandten Methodik, zu technischen Mitteln und zu dem zu untersuchenden Gegenstand bereitgestellt.

Ausgehend von in der Wissenschaft und in der Industrie seit langer Zeit anerkannten standardisierten Vorgehensmodellen wie dem *KDD – Knowledge Discovery in Databases Process* – (Fayyad, Piatetsky-Shapiro & Smyth, 1996) bzw. dem etwas jüngeren *CRISP-DM – Cross Industry Standard Process for Data Mining* – (Shearer, 2000) wird zunächst das im Rahmen dieser Arbeit praktizierte Analyseverfahren skizziert sowie die wesentlichen Grundlagen der explorativen Datenanalyse und der Visualisierung von Daten beschrieben.

Im folgenden zweiten Abschnitt werden die im Zuge der zahlreichen praktischen Untersuchungen eingesetzten Werkzeuge und Technologien vorgestellt.

Unter verschiedenen Aspekten wird abschließend die Datenbasis betrachtet und präsentiert. So werden hier die Daten u. a. durch Angaben zu ihrer Herkunft, ihrer Zusammensetzung und ihrer Qualität zum einen formal beschrieben. Statistische Abfragen sowie erste Visualisierungen z.B. zu bestehenden Mengengerüsten geben hier aber auch bereits interessante Einblicke in Struktur und Inhalt der Daten.

### 2.1. Theorie

Der Wunsch, Wissen aus Daten zu extrahieren, ist nicht nur sinnstiftend für diese Arbeit. Vielmehr ist er in der heutigen Informationsgesellschaft, in der viele erfolgreiche Geschäftsmodelle wie die der Big Five<sup>1</sup> gerade auf einer intelligenten wirtschaftlichen Verwertung dieser Ressource beruhen, nahezu allgegenwärtig.

Aber nicht nur Google, Apple und andere haben früh erkannt, dass Daten gerade auch mit Blick auf ihr expansives Wachstum eine sehr ergiebige Quelle wertvoller Informationen<sup>2</sup> darstellen, sondern auch die Wissenschaften.

Diese letzteren waren es, die schon in den 1980er Jahren damit begonnen haben, Daten nicht nur sporadisch auf interessante Muster hin zu untersuchen, sondern unter dem Begriff *Data Mining* und später auch *Data Analytics* strategisch sinnvolle und allgemeingültige Prozesse zu etablieren (Runkler, 2020).

---

<sup>1</sup> Die Bezeichnungen *The Big Five* oder auch *GAFAM* gelten den fünf größten globalen Technologieunternehmen: Google, Apple, Facebook, Amazon und Microsoft: [Statista, 01/2020](#)

<sup>2</sup> Siehe hierzu die geschätzten Mengen der E-Mails, WhatsApp-Nachrichten oder YouTube-Uploads, die jede Minute allein im Internet entstehen bzw. verarbeitet werden: [Statista, 06/2021](#)

### 2.1.1. Standardisierte Vorgehensmodelle der Datenanalyse

Neben organisatorischen und wirtschaftlichen Erwägungen waren und sind es auch einfach faktische Gegebenheiten, die die Notwendigkeit der Standardisierung und Automatisierung von Analyseprozessen früh verdeutlichte und über die Jahre viele Experten zu entsprechenden Lösungsansätzen motivierte.

Denn wie Runkler (2020) und andere schreiben, ist die Datenanalyse ein stark interdisziplinärer Prozess, bei dem je nach Kontext oft mehrere Personen aus ganz unterschiedlichen Fachbereichen zusammenkommen. Damit liegt es auf der Hand, dass hier in einem äußerst heterogenen Umfeld von Experten, u. a. für Statistik, für maschinelles Lernen oder für Datenbanksysteme, die Orientierung an einem klar strukturierten Verfahren die Zusammenarbeit erheblich vereinfacht.

Konkrete wirtschaftliche Vorteile durch Zeit- und Kosteneinsparungen und die größere Objektivität bei der Durchführung der Analyse werden von Fayyad et al. (1996) als wichtige weitere Motive genannt. Schon im Jahr 1996 erkannten sie aber auch das Problem des *Data Overload* in manchen Bereichen der Forschung und sie wiesen darauf hin, dass ein organisierter Prozess unbedingt erforderlich ist, um die faktische Durchführbarkeit einer Datenanalyse überhaupt zu gewährleisten.

### KDD – Knowledge Discovery in Databases Process

Der *Knowledge Discovery in Databases Process* (KDD), wie er von Fayyad et al. (1996) geprägt wurde, beschreibt einen umfassenden Datenanalyseprozess, in dessen Kern Verfahren des Data Mining zur Anwendung kommen.<sup>3</sup>

Die folgende Übersicht veranschaulicht die fünf verschiedenen Phasen des KDD – *Selektion, Vorverarbeitung, Transformation, Data Mining, Interpretation/Evaluierung* –, die, wie durch die gestrichelten Pfeile angedeutet, bei einer Analyse in vielen Fällen auch wiederholt durchlaufen werden müssen, bis tatsächlich ein aussagekräftiges Ergebnis vorliegt.

Über die genaue Zuordnung und Differenzierung von Arbeitsschritten innerhalb der oben dargestellten Hauptphasen des KDD, gibt es in der Literatur verschiedene Meinungen. Azevedo & Santos (2008) ordnen diese wie folgt ein:

1. *Selektion*: Auswahl des relevanten Teils des Datenbestands, der als Gegenstand der Untersuchung geeignet erscheint.
2. *Vorverarbeitung*: Zusammenführung und Bereinigung der selektierten Daten, bei der u. a. falsche und inkonsistente Daten entfernt werden sollten.
3. *Transformation*: Überführung der Daten u. a. mittels Konvertierung von Datentypen, wodurch z. B. verschiedene Datumsformate vereinheitlicht werden.

---

<sup>3</sup> Unter dem folgenden Link findet sich dauerhaft die zitierfähige Version der im Text erwähnten Definition: [Gabler Wirtschaftslexikon, Springer Gabler, 04/2022](#)

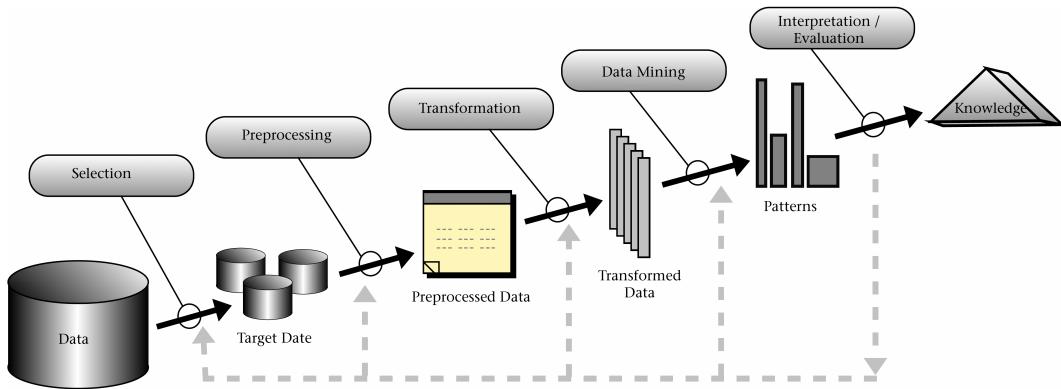


Abbildung 1.: Phasen des KDD-Prozesses. Original von Fayyad et al. (1996).

4. *Data Mining*: Anwendung von Methoden und Algorithmen mit deren Unterstützung möglichst automatisch empirische Zusammenhänge aus der bereitgestellten Datenbasis extrahiert werden sollen.<sup>4</sup>
5. *Interpretation/Evaluierung*: Auslegung und Prüfung der gewonnenen Erkenntnisse, ggf. unterstützt durch Visualisierung extrahierter Muster.

### CRISP-DM – Cross Industry Standard Process for Data Mining

Der *Cross Industry Standard Process for Data Mining* (CRISP-DM) ist ein auf Basis eines ehemals durch die EU geförderten Projekts entstandenes anwendungs- und branchenunabhängiges Vorgehensmodell für das Data Mining.

Konzipiert und entwickelt wurde das Vorhaben in den Jahren 1996 bis 2000 durch ein Konsortium namhafter Industrieunternehmen, der CRISP-DM Special Interest Group, der damals u. a. Daimler-Benz, NCR und ISL angehörten. Ihr Ziel war es, für Data Mining-Projekte ein nicht-propriätes Standard-Prozessmodell zu etablieren, das konkret als Blaupause dienen kann, um Datenbestände z. B. nach interessanten Mustern und Trends zu durchsuchen (Shearer, 2000).

Wie in der obigen Abbildung ersichtlich, umfasst der CRISP-DM insgesamt sechs Phasen, die hiernach in einem normalen Data Mining-Projekt zu durchlaufen sind. Ähnlich wie beim KDD können sich verschiedene Phasen dabei wiederholen oder es wird auch ein Springen zwischen den einzelnen Phasen erforderlich.

Die Ziele und Aufgaben der einzelnen Phasen des CRISP-DM lassen sich nach Shearer (2000) folgendermaßen kurz zusammenfassen:

1. *Geschäftsverständnis*: Beschreibung übergeordneter Ziele, Anforderungen und Beschränkungen; Definition von Strategien, Aufgaben und Methoden.

<sup>4</sup> Unter dem folgenden Link findet sich dauerhaft die zitierfähige Version der im Text erwähnten Definition: [Gabler Wirtschaftslexikon, Springer Gabler, 04/2022](#)

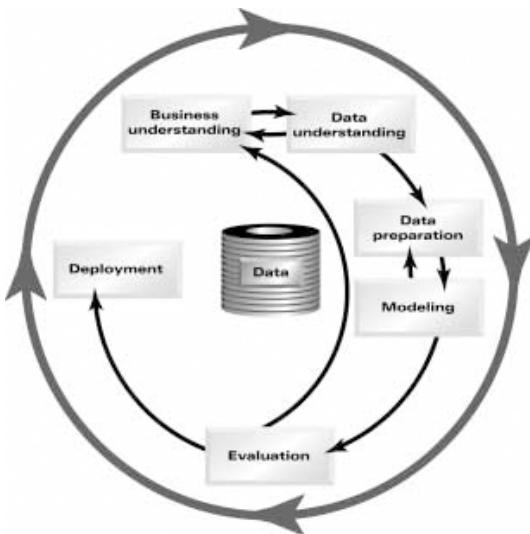


Abbildung 2.: Phasen des CRISP-DM. Original von Shearer (2000).

2. *Datenverständnis*: Sammlung und Beschreibung der Rohdaten; Prüfung und Bewertung der Datenqualität; Feststellung von Datenmängeln.
3. *Datenaufbereitung*: Auswahl, Zusammenführung, Bereinigung und Transformation der Daten zur Erstellung des zu untersuchenden Datenbestands.
4. *Modellierung*: Auswahl und Anwendung geeigneter Modellierungstechniken; Erstellung von Tests; Bewertung und Optimierung von Modellen.
5. *Evaluierung*: Bewertung der Analyseergebnisse und der genutzten Modelle; Prüfung des Gesamtprozesses; Ableitung nachfolgender Verfahrensschritte.
6. *Einsatz*: Aufbereitung und Vorstellung der gewonnenen Erkenntnisse; Ausarbeitung von Strategien und Maßnahmen zur Einführung und dauerhaften Verwendung;

### Vergleich der standardisierten Vorgehensmodelle

Zum Abschluss dieses Kapitels über die standardisierten Vorgehensmodelle in der Datenanalyse soll hier noch einmal auf die Arbeit von Azevedo & Santos (2008) hingewiesen werden, die zum Ziel hatte die Gemeinsamkeiten und Unterschiede von KDD, CRISP-DM und SEMMA<sup>5</sup> miteinander zu vergleichen.

Im Ergebnis bestätigt diese Vergleichsstudie die vollkommene Übereinstimmung von KDD und SEMMA, bzw. definiert SEMMA als praktische Implementation des älteren KDD-Prozesses, weshalb auch in dieser Arbeit auf eine Darstellung dieses Standardprozesses verzichtet wurde.

Im Vergleich von KDD und CRISP-DM gibt es dagegen erkennbare Unterschiede, die sich darin zeigen, dass der CRISP-DM die im KDD implizit enthaltenen vor- und

<sup>5</sup> Unter dem folgenden Link findet sich eine kurze Einführung zu SEMMA, das den übergeordneten Prozess für den SAS® Enterprise Miner™ darstellt: [Introduction to SEMMA, SAS, 04/2022](#)

nachgelagerten Stufen explizit als separate Teil des Prozesses ausführlich beschreibt. Weitere Abweichungen lassen sich feststellen bei der Zuordnung von Teilschritten innerhalb des *Data Understanding* und *Data Preparation*. Interessanterweise wird dies in dieser Studie nicht konsistent behandelt, und stimmt daher auch nur bedingt mit dem ursprünglich von Shearer (2000) skizzierten Prozess überein.

KDD	SEMMA	CRISP-DM
Pre KDD	-----	Business understanding
Selection	Sample	Data Understanding
Pre processing	Explore	
Transformation	Modify	Data preparation
Data mining	Model	Modeling
Interpretation/Evaluation	Assessment	Evaluation
Post KDD	-----	Deployment

Abbildung 3.: KDD, SEMMA, CRISP-DM. Original von Azevedo & Santos (2008).

### 2.1.2. Angepasstes Vorgehensmodell für diese Arbeit

Die im vorausgegangenen Abschnitt präsentierten Vorgehensmodelle haben alleamt dasselbe Ziel: Sie wollen den äußerst vielfältigen Prozess einer Datenanalyse möglichst vollständig und genau in einem Standardverfahren abbilden und für den Anwender sinnvolle Handlungsempfehlungen formulieren.

Diese Verfahren sind also keineswegs verpflichtend. Sie sollen zur Orientierung dienen, aber es obliegt demnach stets dem Anwender je nach Anwendungskontext die standardisierten Verfahrensschritte auf die im konkreten Fall vorliegenden Anforderungen anzupassen (Shearer, 2000).

### Grundzüge des verwendeten Vorgehensmodells

Im Hinblick auf die anstehenden Untersuchungen im Rahmen dieser Arbeit, wird das im weiteren Verlauf verwendete Vorgehensmodell – auf Basis des von Shearer (2000) beschriebenen CRISP-DM – wie folgt skizziert:

1. *Geschäftsverständnis*: Das Thema dieser Arbeit definiert gleichzeitig auch das übergeordnete Ziel: die *Identifikation typischen Benutzerverhaltens in digitalen Studienformaten*. Untergeordnete Ziele lassen sich mit Blick auf die Methodik und den Gegenstand der Untersuchung beschreiben. So gilt es, wie in der Einleitung zu dieser Arbeit beschrieben, mit Mitteln der explorativen Datenanalyse den Ist-Zustand studentischen Lern- und Kommunikationsverhaltens möglichst detailliert zu skizzieren und das jeweilige Vorgehen dabei verständlich und nachvollziehbar zu dokumentieren. Dazu bedarf es im Rahmen der eigentlichen Analyse neben der bestimmten Auswahl von Daten gerade auch

der gezielten Entwicklung von Fragen, die geeignet sein könnten, das in den Daten verborgene Benutzerverhalten zu offenbaren und davon ausgehende neue Annahmen zu formulieren.

2. *Datenverständnis*: Ein fundiertes Verständnis über die Herkunft der zu untersuchenden Daten, deren Bedeutung und Qualität ist essentiell, um mögliche Zusammenhänge zu verstehen oder neues Wissen aus den Daten extrahieren zu können. Das nachfolgende Kapitel [Datenbasis](#) trägt diesem grundlegenden Erfordernis Rechnung und gibt detailliert Aufschluss über den Gegenstand der Untersuchung.
3. *Datenaufbereitung*: Im Fokus dieser Phase steht der konkrete Untersuchungsgegenstand. Dessen Bereitstellung vollzieht sich entsprechend der gegebenen Zielsetzung in mehreren Schritten. Zu nennen sind hier in erster Linie:
  - Datenauswahl: Die für die Untersuchung relevanten Daten sind nach Art und Umfang aus den Spalten und Zeilen der initial vorbereiteten Daten zu selektieren. Warum gewisse Daten relevant sind bzw. diese nicht in der Auswahl berücksichtigt werden, sollte begründet werden können.
  - Datenbereinigung: Da die Daten initial keine falschen Werte aufweisen, entfällt naturgemäß eine entsprechende Korrektur. Gegebenfalls müssen aber fehlende Werte ergänzt werden, um bestimmte Abfragen sinnvoll durchführen zu können.
  - Datentransformation: Für eine Untersuchung kann es erforderlich sein, zuvor aus den Daten ein neues Attribut abzuleiten, den Datentypen eines Attributs zu konvertieren oder auch weitere Datensätze zu ergänzen. Die Gründe hierfür sollten ebenfalls klar ersichtlich dokumentiert werden.
4. *Datenanalyse*:<sup>6</sup> Das Verfahren, das bei den eigentlichen Untersuchungen zur Anwendung kommen soll, orientiert sich an der Methodik der explorativen Statistik bzw. der [explorativen Datenanalyse](#). Insbesondere durch geeignete visuelle Darstellungen<sup>7</sup> sollen in den Daten bemerkenswerte Strukturen und Zusammenhänge aufgezeigt werden, die zur Formulierung von Hypothesen anregen. Mögliche Darstellungsformen sind beispielsweise:
  - Balkendiagramm
  - Streudiagramm
  - Liniendiagramm

Aufgrund komplexer Fragestellungen und Zwischenbewertungen sind bei der Analyse oft mehrere Anläufe nötig, um schließlich interessante Hypothesen

---

<sup>6</sup> Im weiteren Verlauf der Arbeit soll diese Phase vorzugsweise *Datenanalyse* genannt werden, da der Begriff Modellierung häufig die Anwendung komplexer Machine Learning Modelle impliziert.

<sup>7</sup> s. das nachfolgende Kapitel [Formen der Datenvisualisierung](#)

generieren zu können. Gegebenenfalls muss auch die Frage selbst angepasst werden bzw. sind auch die Daten erneut aufzubereiten.

5. *Evaluierung:* Die Interpretation und die Bewertung von Analyseergebnissen vollzieht sich typischerweise im stetigen Wechsel mit der Optimierung der Methoden in der vorhergehenden Analysephase. Das Ziel ist dabei nur die Entwicklung einer Hypothese auf den erkannten Mustern oder Verbindungen in den Daten, nicht aber die Evaluierung der Hypothese selbst oder die Ableitung weiterer Verfahrensschritte aus einer gewonnenen Hypothese.
6. *Dokumentation:*<sup>8</sup> Erkenntnisse aus den Untersuchungen sind letztlich noch verständlich aufzubereiten und umfassend zu dokumentieren, so dass diese z. B. auch in einer neuen Studie zur Entwicklung von Kursempfehlungen genutzt werden könnten. Im Kapitel [Ergebnisse](#) werden dazu wichtige Erfahrungen aus dieser Arbeit zusammengefasst sowie bemerkenswerte Untersuchungsansätze und deren Resultate betrachtet bzw. miteinander verglichen.

Dieses Modell wird später bei der tatsächlichen Durchführung von Analysen (siehe das Kapitel [Analysen](#)) erneut als Vorlage dienen und wie erwähnt in den Phasen *Datenaufbereitung*, *Datenanalyse* und *Evaluierung* je nach Anforderung auch mehrmals spezifisch angepasst werden müssen.

Die nachfolgende Grafik zeigt das in dieser Arbeit verwendete Vorgehensmodell mit den oben beschriebenen Phasen. Die nur im Rahmen der konkreten Analyse zu durchlaufenden Phasen sind dabei farblich hervorgehoben.

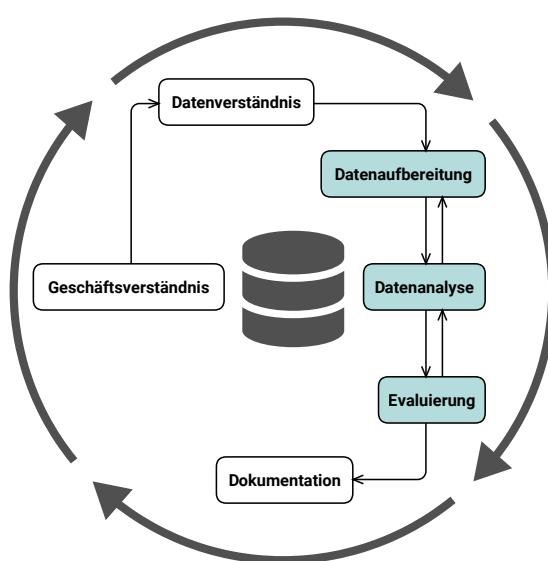


Abbildung 4.: Phasen des verwendeten Vorgehensmodells

---

<sup>8</sup> In dieser Arbeit soll diese Phase bevorzugt mit *Dokumentation* bezeichnet werden, da der Begriff *Einsatz* zu sehr auf die praktische Anwendung konkreter Untersuchungsergebnisse abzielt.

### 2.1.3. Explorative Datenanalyse

Unter anderem Polasek (1994) betrachtet die Explorative Datenanalyse (EDA) als ein Teilgebiet der deskriptiven Statistik, deren grundlegende Methodik maßgeblich von Tukey et al. (1977) entwickelt wurde.

Letzterer führte in seiner Arbeit aus, dass eine statistische Betrachtung sich nicht damit begnügen sollte, Vermutungen zu prüfen, um z. B. ein bestimmtes Ergebnis zu bestätigen oder zu verwerfen, sondern eher darauf ausgelegt sein sollte, in den Daten unerwartete Trends, Muster oder Zusammenhänge zu erkennen, um darauf basierend schließlich neues Wissen generieren zu können.

Das Hauptziel einer Datenanalyse ist nach Tukey also weniger darin zu sehen, Hypothesen anhand statistischer Methoden zu testen und auszuwerten, als vielmehr aus der Betrachtung der Daten heraus neue Hypothesen abzuleiten.

Polasek (1994) verbindet diese beiden Zielsetzungen mit zwei unterschiedlichen Fragestellungen, die bei einer Datenanalyse grundsätzlich gestellt werden können. Dem Ansatz einer deskriptiven Betrachtung folgend ist stets zu fragen: *Wie kann man die Verteilung eines Merkmals beschreiben?*, während die explorative Analyse eine Antwort sucht auf die Frage: *Was ist an der Verteilung eines Merkmals bemerkenswert?*

Antworten auf Fragen, die im Kontext einer explorativen Datenanalyse gestellt werden, werden dabei meistens in anschaulicher visualisierter Form gegeben und dienen, wie Kusnierz (2020) mit Verweis auf Bubenhofer & Kupietz (2018) schreibt, mit Blick auf nachfolgende Untersuchungen dann also eher als Einstieg oder als Zwischenschritt auf dem Wege der Erkenntnisgewinnung.

Allgemein eignen sich gerade Visualisierungen besonders gut, um explorative Analysen zügig voranzubringen, denn aufgrund der menschlichen Wahrnehmung und der Art und Weise wie bildhafte Informationen im menschlichen Gehirn verarbeitet werden, können Bilder komplexe Zusammenhänge schneller und besser verdeutlichen als andere Informationsquellen (Kusnierz, 2020).

Darüberhinaus können Visualisierungen auch über Sprachbarrieren hinweg als universelles Mittel zur Präsentation von Fakten und Informationen genutzt werden (Schumann & Müller, 2000) und damit sowohl das Verständnis als auch den Austausch über die aus einer Analyse abzuleitenden Erkenntnisse erleichtern.

### 2.1.4. Formen der Datenvisualisierung

Um die wesentlichen Aussagen der zahlreichen Abbildungen im Hauptteil dieser Arbeit besser nachvollziehen zu können, ist es von Vorteil, den Einsatzzweck und die spezifischen Besonderheiten verschiedener Visualisierungsformen zu kennen.

Aus diesem Grunde sollen nun zum Abschluss dieses theoretischen Grundlagen-teils die in dieser Arbeit häufig verwendeten Darstellungsarten auch anhand ent-

sprechender beispielhafter Anwendungen vorgestellt und kurz erläutert werden.

Ein **Histogramm** ist eine graphische Darstellungsform der Häufigkeitsverteilung metrisch skalierter Merkmale, die in sog. Klassen zusammengefasst werden. Eine Klasse (oder engl. bin) bezeichnet dabei einen Teil einer statistischen Grundgesamtheit innerhalb eines bestimmten Intervalls (Rönz, Strohe & Eckstein, 1994).

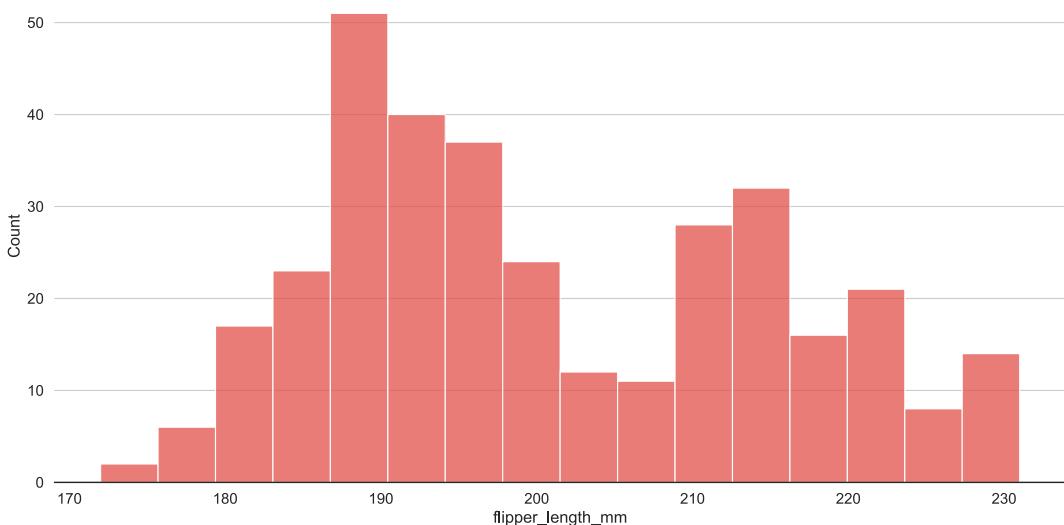


Abbildung 5.: Beispiel für ein Histogramm mit 16 Klassen

Wie im Diagramm zu erkennen, werden bei Histogrammen auf der x-Achse stets die Klassen angegeben und auf der y-Achse die entsprechenden Häufigkeitsdichten, so z. B. in absoluten Zahlen oder auch prozentualen Anteilen.<sup>9</sup>

Die Rechtecke, die über den jeweiligen Klassen gezeichnet werden entsprechen in ihrer Höhe den relativen Häufigkeitsdichten, wenn alle Klassenbreiten gleich 1 sind. Sind die Klassenbreiten ggf. unterschiedlich ergibt sich die relative Häufigkeit dagegen aus dem Produkt von Klassenbreite und Höhe des Rechtecks.

Dient ein Histogramm als sog. univariate Darstellung in der Regel der Betrachtung nur eines Merkmals, kann ein **Balken- bzw. Säulendiagramm** als bivariate Visualisierungsform den Zusammenhang zwischen zwei Merkmalen wiedergeben (Weins, 2010).

Hierbei kann, abweichend zu Histogrammen, die Häufigkeitsverteilung nicht nur für metrisch skalierte, sondern auch für nominal- und ordinalskalierte bzw. diskrete (nicht klassierte) Merkmale dargestellt werden (Rönz, Strohe & Eckstein, 1994).

Das obige Säulendiagramm zeigt beispielhaft wie auf der horizontalen Achse die Tage als Werte des ordinalskalierten Merkmals *day* abgetragen wurden, während

<sup>9</sup> s. seaborn Documentation (Waskom, 2021): [seaborn.histplot, 07/2022](#)

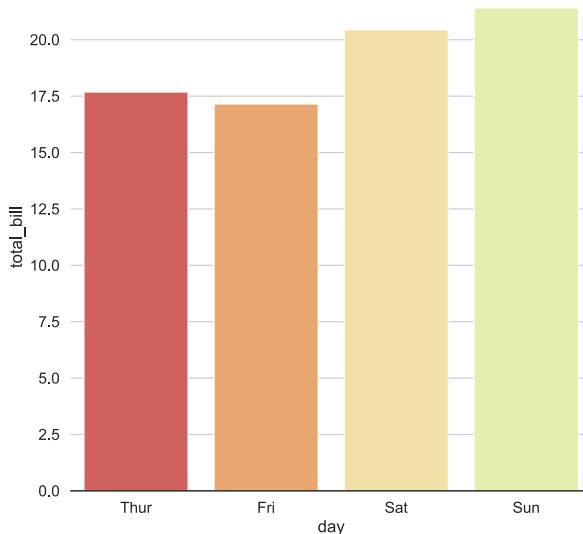


Abbildung 6.: Beispiel für ein Säulendiagramm

auf der vertikalen Achse die Höhe der Rechnungen als Ausprägungen des metrisch skalierten Merkmals *total bill* angegeben sind. Die absoluten bzw. relativen Häufigkeiten werden im Diagramm durch die Höhe der Rechtecke (Säulen) angezeigt.<sup>10</sup>

Ein **Boxplot** ist eine spezielle graphische Darstellung wichtiger Kenngrößen einer Größe nach geordneten Beobachtungsreihe oder auch einer Häufigkeitsverteilung eines kardinalskalierten Merkmals (Rönz, Strohe & Eckstein, 1994).

Er veranschaulicht auf verständliche Weise die Verteilung der Daten und gibt Aufschluss über deren Struktur, die nachfolgend genauer beschrieben wird.

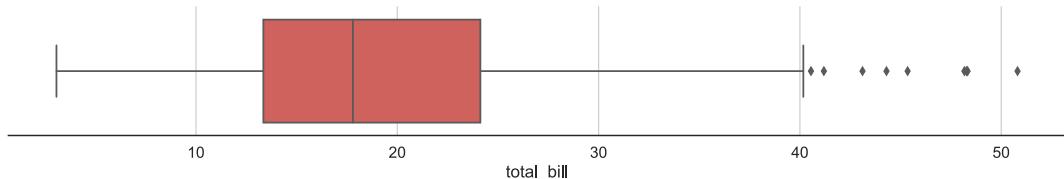


Abbildung 7.: Beispiel für einen Boxplot

Die in der Abbildung rot dargestellte Box beschreibt den Interquartilsabstand, der durch das untere Quartil und das obere Quartil definiert wird und genau die Hälfte aller Werte umfasst. Innerhalb der Box zeigt ein senkrechter Strich den Median an, der den mittleren Wert der Beobachtungsreihe repräsentiert.

Unterhalb des unteren Quartils befinden sich genau ein Viertel aller Werte, oberhalb des oberen Quartils ebenfalls. Ihre Lage wird durch die sog. Whiskers (waagerechte Linien) beschrieben und durch senkrechte Striche an deren Enden limitiert.

Liegen Werte außerhalb dieser Begrenzungen liegen, so gelten diese als Ausreißer,

<sup>10</sup> s. seaborn Documentation (Waskom, 2021): [seaborn.barplot, 07/2022](#)

die gerade im Rahmen einer explorativen Datenanalyse von besonderem Interesse sein können. Befinden sich alle Werte dagegen innerhalb der Quartile kennzeichnen die äußereren Enden der Whiskers das Minimum bzw. Maximum der Wertereihe.<sup>11</sup>

## 2.2. Technik

Verschiedene Technologien, Anwendungen und Bibliotheken sind im Bereich der Datenanalyse heute Standard und bieten eine Vielfalt an Einsatzmöglichkeiten, so dass die Auswahl der geeigneten technischen Mittel mit Bedacht erfolgen sollte.

Welche Ressourcen nun im Rahmen dieser Arbeit verwendet wurden und auch weshalb, soll in den folgenden Abschnitten in kurzer Form vorgestellt werden.

### Einrichtung der Arbeitsumgebung

Grundlegende vorbereitende Maßnahmen zur Gestaltung der Arbeitsumgebung waren die Installation eines geeigneten relationalen Datenbankmanagementsystems (DBMS) und einer Programmiersprache, die im weiteren Verlauf der Arbeit eine leicht nachvollziehbare Implementierung der Analysen ermöglichen sollte.

Aufgrund seiner Bekanntheit und Zuverlässigkeit fiel die Wahl bezüglich des DBMS dann auf MySQL.<sup>12</sup> Neben der persistenten Speicherung der Daten gewährleistet MySQL einen sehr schnellen und flexiblen Zugriff hierauf, da die MySQL-Schnittstelle zu vielen weiteren Anwendungen kompatibel ist und so die Nutzung anderer Ressourcen offen ließ.

Anwendungen und Bibliotheken, die in der Datenanalyse heute gewissermaßen als Standard zu betrachten sind, und nachfolgend auch noch besprochen werden sollen, basieren zu einem großen Teil auf der Programmiersprache Python.<sup>13</sup> Daher und weil Python außerdem eine große Developer- und User-Community gerade auch im Bereich Datenanalyse besitzt, wurde dieser Programmiersprache für diese Arbeit der Vorzug gegeben.

### Durchführung von Analysen

Um in späteren Projekten nahtlos an die Erfahrungen und Erkenntnisse aus dieser Arbeit anknüpfen zu können, wurden in den verschiedenen Phasen der praktischen Untersuchungen ebenfalls nur quelloffene Standards berücksichtigt.

Wichtigste Anwendung war hierbei das vom Project Jupyter entwickelte *Jupyter Notebook*<sup>14</sup>. Als Laufzeitumgebung zur Organisation und Ausführung von Datenanalysen ist dieses in der Lage, je nach Anforderung verschiedene Bibliotheken zu importieren und damit die Anwendung flexibel zu erweitern.

---

<sup>11</sup> s. seaborn Documentation (Waskom, 2021): [seaborn.boxplot, 07/2022](#)

<sup>12</sup> s. MySQL Documentation (Axmark & Widenius, 2022): [MySQL Reference Manual, 07/2022](#)

<sup>13</sup> s. Python Documentation (Van Rossum & Drake Jr, 1995): [Python Documentation, 07/2022](#)

<sup>14</sup> s. Jupyter Project Documentation (Kluyver et al., 2016): [Jupyter Project Documentation, 07/2022](#)

Um auch die fortgeschrittenen Funktionalitäten einer Entwicklungsumgebung (z. B. Syntax-Highlighting, Auto-Completion oder Debugging) nutzen zu können, wurde entschieden Jupyter Notebooks innerhalb von *PyCharm*<sup>15</sup> auszuführen und nicht wie oft üblich mit *Anaconda*<sup>16</sup> nur als Frontend-Applikation zu installieren.



Abbildung 8.: Systematischer Aufbau der Bibliotheken

Die grundlegende Funktionalität auf der andere Bibliotheken zur Datenanalyse aufbauen, wird, wie in der obigen Abbildung dargestellt, in Python- bzw. Jupyter-Umgebungen in der Regel von *NumPy*<sup>17</sup> bereitgestellt. Seine effizient implementierten Methoden ermöglichen die Ausführung komplexer numerischer Berechnungen auf einfache Weise und machen diese Programmabibliothek damit zur ersten Wahl, wenn z. B. die Verarbeitung großer mehrdimensionaler Arrays erforderlich wird.

*Hier werden noch weitere Abschnitte zu anderen Bibliotheken ergänzt.*

### 2.3. Datenbasis

Gegenstand der Untersuchungen zu dieser Arbeit ist ein vom *Projektteam DiSEA* zur Verfügung gestellter Datenbestand aus dem Wintersemester 2020/2021<sup>18</sup>. In diesem

<sup>15</sup> s. PyCharm Documentation: [PyCharm Documentation, 07/2022](#)

<sup>16</sup> s. Anaconda Documentation: [Anaconda Distribution Documentation, 07/2022](#)

<sup>17</sup> s. NumPy Reference (Harris et al., 2020): [NumPy Reference, 07/2022](#)

<sup>18</sup> Das gesamte Semester musste nach der SARS-CoV-2-Infektionsschutzmaßnahmenverordnung des Berliner Senates unter erhöhten Sicherheitsbedingungen stattfinden. Die Regelungen für das Lehr- und Prüfungsgeschehen wurden an der BHT infolgedessen wie folgt angepasst:

- keine Lehrveranstaltungen und Prüfungen in Präsenz
- keine Zählung des Semesters als Fachsemester
- keine Zählung von Prüfungsfehlversuchen

enthalten sind die anonymisierten Moodle-Daten von Studenten, Dozenten sowie anderem Personal (in der weiteren Arbeit ‚Andere‘ genannt) der *Berliner Hochschule für Technik (BHT)* und der *Alice Salomon Hochschule Berlin (ASH)* aus den folgenden Studiengängen:

- Master-Studiengang Medieninformatik Online (MMIO)
- Bachelor-Studiengang Wirtschaftsingenieurwesen Online (BWIO)
- Bachelor-Studiengang Wirtschaftsinformatik Online (BWINF)
- Bachelor-Studiengang Soziale Arbeit Online (BSAO)

### 2.3.1. Beschreibung der Daten

Um den Zugriff auf die Daten und deren praktische Untersuchung zu erleichtern, wurden diese zunächst vom Projektteam aus der Datenbank des Moodle-Systems (Green, 2022) extrahiert und in einem ersten Arbeitsschritt in nur einer Relation zusammengeführt.

Hierbei wurden Merkmale, die für diese Arbeit erwartungsgemäß keinen Mehrwert besitzen bereits eliminiert, während z. B. das Attribut *Studiengang* als neue Spalte in die Tabelle aufgenommen wurde, um die Zuordnung der Datensätze zu den jeweiligen Studiengängen<sup>19</sup> unmittelbar erkennen zu können.

Des weiteren wurden vorab die Merkmale *course\_module\_type* und *instanceid* eingefügt, um bei der Datenanalyse auch deren Informationsgehalt zur Identifikation typischen Benutzerverhaltens sinnvoll nutzen zu können.

Damit die Daten in einem beliebigen IT-Umfeld einfach weiterverarbeitet werden können, wurden sie im Anschluss an ihre Vorbereitung in einem für diesen Zweck typischen CSV-Format exportiert. Übergeben wurden die CSV-Daten schließlich als offene und komprimierte Textdateien in ASCII-Kodierung, in der die Daten entgegen der üblichen Praxis jedoch nicht durch Kommata, sondern durch Semikola strukturiert waren.

Der bereitgestellte Datenbestand umfasst insgesamt 969032 Datensätze. Dabei handelt es sich um eine Teilmenge von Log-Einträgen auf dem Moodle-Server, mit denen client- und serverseitige Aktionen fortlaufend protokolliert werden. Typische Aktionen, die so u. a. aufgezeichnet werden sind das Aufrufen eines Kursmoduls, das Starten eines Uploads, das Senden einer Nachricht oder auch die Bewertung einer Aufgabe.

---

<sup>19</sup> Ergänzend zu den genannten offiziellen Studiengängen, sind in den Daten ferner auch Datensätze zu einem Studiengang 0 enthalten. Hierbei handelt es sich jedoch um eine besondere Entität, die sich nur auf Aktivitäten bezieht, die außerhalb des eigentlichen Kursgeschehens stattfanden, z. B. Logins, Chats oder Aufrufe des Kalenders bzw. Dashboards.

## Formale Angaben über die Daten

Ein erster informativer Einblick in die Struktur und die Art der zu untersuchenden Daten ergibt sich nach deren Import in eine MySQL-Datenbank mithilfe der folgenden einfachen SQL-Abfrage:

---

```
1 DESCRIBE moodle_data;
```

---

Listing 1: Abfrage zu Struktur und Art der importierten Originaldaten

Field	Type	Null	Key	Default	Extra
courseid	int (11)	YES		NULL	
Studiengang	varchar(11)	YES		NULL	
userid	int (11)	YES	MUL	NULL	
relateduserid	int (11)	YES		NULL	
action	varchar(10)	YES		NULL	
eventname	varchar(57)	YES		NULL	
objecttable	varchar(27)	YES		NULL	
objectid	int (11)	YES		NULL	
timecreated	int (11)	YES		NULL	
course_module_type	varchar(18)	YES		NULL	
instanceid	int (11)	YES		NULL	

Abbildung 9.: Struktur und Art der importierten Originaldaten

Die obige Ausgabe beschreibt das Schema der importierten Daten. Von Interesse für diese Arbeit sind hier aber nur die Werte zu *Field* und *Type*, die die Spaltennamen der Tabelle und die Datentypen der darin enthaltenen Werte angeben.

## Informationen und deren Beziehungen

Die nachfolgende tabellarische Übersicht zeigt nun, welche Informationen in den Feldern der verschiedenen Merkmale des Datenbestandes tatsächlich enthalten sind und in welchen Beziehungen diese innerhalb der aktuell betriebenen relationalen Datenbank des VFH-Moodle stehen.<sup>20</sup>

## Erste Erkenntnisse über die Daten

Um die Beschreibung der Daten zu vervollständigen, soll im Folgenden anhand einiger statistischer Abfragen der Gegenstand der Untersuchung, die sogenannten Arbeitsdaten, inhaltlich genauer betrachtet und mithin erste Erkenntnisse daraus gewonnen werden.

---

```
1 SELECT COUNT(DISTINCT userid) AS "total_number_users"
2 FROM moodle_data;
```

---

Listing 2: Abfrage zur Menge aller Benutzer

<sup>20</sup> s. Moodle Entity Relationship Documentation (Green, 2022): [Moodle ERD, 05/2022](#)

Merkmal	Information / Beziehung innerhalb des VFH-Moodle
courseid	Studienmodul, das im WS 2020/2021 belegt wurde. <i>Fremdschlüssel zur Identifikation eines bestimmten Studienmoduls in der Relation course.</i>
Studiengang	Studiengang, in dem aktuell studiert wird. <i>Frei gewählte Kennziffer zur eindeutigen Unterscheidung der Studiengänge; bedeutet keine Referenz auf eine andere Entität.</i>
userid	Kennzahl zur Identifikation des Benutzers. <i>Aus Datenschutzgründen verschlüsselte ID zur Identifikation eines bestimmten Benutzers (z. B. der Sender einer Nachricht).</i>
relateduserid	Kennzahl zur Identifikation eines weiteren Benutzers. <i>Verschlüsselte ID des interagierenden Benutzers, der z. B. bei einem Chat den Empfänger einer Nachricht repräsentiert.</i>
action	Interaktion, die im Moodle-System ausgeführt wurde. <i>Allgemeinere Form des eventtype, der auch im eventname als notwendiger Bestandteil redundant enthalten ist.</i>
eventname	Mehrteiliger Bezeichner für das ausgelöste Event. <i>Ausgelöst durch eine Interaktion wird ein Bezeichner durch die drei Werte modulename, instance und eventtype der Relation event generiert und eingetragen.</i>
objecttable	Relation zur Verwaltung von Objekttabellen. <i>Abhängig von der Art des Kursmoduls und der Interaktion werden die durch Verwendung bestimmter Objekte tangierten Tabellen dokumentiert, z. B. assign_grades, course_modules oder forum_discussions</i>
objectid	Kennzahl zur Identifikation des verwendeten Objekts. <i>Fremdschlüssel zur Identifikation des durch die Interaktion tangierten Objekts in der zugehörigen Relation objecttable.</i>
timecreated	Zeitpunkt der ausgeführten Interaktion. <i>10-stelliger Unix Epoch Timestamp, der seit Donnerstag, dem 01.01.1970, 00:00 Uhr UTC die vergangenen Sekunden zählt.</i>
course_module_type	Typ des verwendeten Kursmoduls. <i>Zur Anreicherung des Informationsgehalts aus der Relation course_modules entnommener Bezeichner des Modultyps, z. B. assign, forum, label oder resource</i>
instanceid	Kennzahl zur Identifikation des Kursmodultyps. <i>Fremdschlüssel zur Identifikation des Kursmodultyps in der zugehörigen Relation course_modules.</i>

Tabelle 1.: Schema des Datenbestandes mit Erläuterungen

Im Ergebnis inkludiert sind neben Einzelpersonen auch zwei Benutzergruppen, die einer Beobachtung ihres Verhaltens nicht zugestimmt haben (userid = -2) oder die

```
+-----+
| total_number_users |
+-----+
| 144 |
+-----+
1 row in set (0,00 sec)
```

Abbildung 10.: Menge aller Benutzer

im Bachelor-Studiengang Medieninformatik aktiv waren (`userid = -3`)<sup>21</sup>. Abzüglich dieser beiden Gruppen erhielte man im Ergebnis somit 142 Einzelpersonen.

---

```
1 SELECT userid, COUNT(userid) AS "total_number_records"
2 FROM moodle_data
3 GROUP BY userid;
```

---

Listing 3: Abfrage zur Menge der Log-Einträge pro Benutzer

```
+-----+-----+
| userid | total_number_records |
+-----+-----+
| ...   | ...           |
| 1     | 3865          |
| 2     | 4706          |
| 3     | 3373          |
| ...   | ...           |
| 26    | 92242         |
| 142   | 10             |
| 143   | 1387          |
| 144   | 240           |
+-----+-----+
144 rows in set (0,27 sec)
```

Abbildung 11.: Menge der Log-Einträge pro Benutzer

Aus Platzgründen werden in der obigen Ergebnistabelle nur wenige der insgesamt 144 Zeilen des Abfrageergebnisses angezeigt. Es wird aber auch bereits in diesem kleinen Ausschnitt deutlich, wie unterschiedlich die Benutzeraktivitäten über das Semester hinweg in ihrem Umfang waren.

---

```
1 SELECT Studiengang, COUNT(DISTINCT userid) AS "total_number_users"
2 FROM moodle_data
3 GROUP BY Studiengang;
```

---

Listing 4: Abfrage zur Menge der Benutzer pro Studiengang

```
+-----+-----+
| Studiengang | total_number_users |
+-----+-----+
| 0          | 144            |
| 1          | 54              |
| 2          | 40              |
| 3          | 33              |
| 4          | 25              |
+-----+-----+
5 rows in set (0,46 sec)
```

Abbildung 12.: Menge der Benutzer pro Studiengang

<sup>21</sup> Um die Privatsphäre meiner Komilitonen zu schützen und meine Unvoreingenommenheit bei den Untersuchungen zu wahren, wurde vom Projektteam entschieden, alle Studenten im Bachelor-Studiengang Medieninformatik in einer Gruppe zusammenzufassen und diese nur bei Kontextbetrachtungen zu berücksichtigen.

Bemerkenswert am Ergebnis ist, dass dem allgemeinen Studiengang 0 alle zuvor ermittelten Benutzer zugeordnet sind, deren Summe in den Studiengängen 1 bis 4 dagegen höher liegt. Insofern lässt sich an dieser Stelle bereits folgern, dass es auch Benutzer geben muss, die in mehreren Studiengängen aktiv waren, insbesondere auch deshalb, da manche Benutzer wie z. B. Angehörige der Hochschulverwaltung nicht am Geschehen in den offiziellen Studiengängen teilnehmen.

---

```

1 SELECT userid, COUNT(DISTINCT courseid) AS "total_number_courses"
2 FROM moodle_data
3 GROUP BY userid
4 ORDER BY total_number_courses;
```

---

Listing 5: Abfrage zur Menge der Kurse pro Benutzer

userid	total_number_courses
144	2
...	...
130	3
...	...
42	4
...	...
32	39
26	168
-2	195

144 rows in set (1,96 sec)

Abbildung 13.: Menge der Kurse pro Benutzer ([s. Anhang](#))

Auch wenn die Tabelle die Ergebnisse aus Platzgründen wiederum nur teilweise darstellt, ist sofort zu erkennen, dass die Menge an Kursen pro Benutzer mitunter weit über der empfohlenen Menge von sechs Kursmodulen für ein Vollzeitstudium in Regelstudienzeit lag. Dies könnte in manchen Fällen wie z. B. beim Benutzer mit der userid 26 mit einer Dozententätigkeit zu begründen sein oder auf eine andere Rolle hindeuten, was aber erst im Hauptteil dieser Arbeit untersucht werden soll.

Den beiden Benutzergruppen mit der userid -2 und -3 sind erwartungsgemäß ebenfalls große Kursmengen zugeordnet, da diese Gruppen eine unbekannte Zahl an Einzelpersonen umfassen. Infolgedessen nehmen sie hier eine Sonderrolle ein und werden nur der Vollständigkeit halber ebenfalls angezeigt. Bei den weiteren Untersuchungen wird je nach Anforderung stets abzuwegen sein, inwiefern diese beiden Personengruppen bei der Interpretation der Ergebnisse tatsächlich berücksichtigt werden dürfen.

Mit Blick auf die unerwartet hohen Mengen an Kursen pro Benutzer soll zum Schluss dieses Kapitels die Anzahl an Benutzern mit überdurchschnittlich vielen Kursen und die Zuordnung von Benutzern und Studiengängen betrachtet werden.

---

```

1 SELECT userid, COUNT(DISTINCT courseid) AS "total_number_courses"
2 FROM moodle_data
3 WHERE userid > 0
4 GROUP BY userid
5 HAVING total_number_courses >= 12
```

---

```
6 ORDER BY total_number_courses;
```

Listing 6: Abfrage zu Benutzern mit überdurchschnittlich vielen Kursen

userid	total_number_courses
68	12
114	30
78	31
53	33
133	34
32	39
26	168

84 rows in set (1,71 sec)

Abbildung 14.: Benutzer mit überdurchschnittlich vielen Kursen

```
1 SELECT userid, COUNT(DISTINCT Studiengang) AS "total_number_studies"
2 FROM moodle_data
3 WHERE Studiengang > 0 AND userid > 0
4 GROUP BY userid
5 HAVING total_number_studies > 1
6 ORDER BY total_number_studies;
```

Listing 7: Abfrage zur Menge der Studiengänge 1 bis 4 pro Benutzer

userid	total_number_studies
44	2
6	2
81	2
27	2
28	2
50	2
29	2
30	2
31	2
32	2
55	2
88	2
21	3
26	4

14 rows in set (1,71 sec)

Abbildung 15.: Menge der Studiengänge 1 bis 4 pro Benutzer

Auch die letzten zwei Abfragen, bei denen nur Einzelbenutzer (s. WHERE-Klausel) betrachtet wurden, können mit ihren Ergebnissen überraschen. So waren 84 von 142 Benutzern und damit wohl auch eine höhere Zahl an Studenten über das Semester hinweg in mindestens doppelt so vielen Kursen aktiv, wie es von den Hochschulen für ein Vollzeitstudium in der Regel empfohlen wird.

Der Gedanke, dass es dann auch Benutzer gegeben haben könnte, die außer dem unspezifischen Studiengang 0 (s. WHERE-Klausel) vielleicht mehrere der eingangs genannten Studiengänge besucht haben, wird durch die Abfrage zur Anzahl der Studiengänge pro Benutzer eindrucksvoll bestätigt: Insgesamt 14 Benutzer waren in mehr als einem der **Studiengänge 1 bis 4** tätig. Dieser Umstand könnte ebenfalls für eine Dozententätigkeit der im Ergebnis enthaltenen Benutzer sprechen und soll im weiteren Verlauf der Arbeit noch genauer untersucht werden.

### 2.3.2. Visualisierung der Daten

Ergänzend zur vorhergehenden Beschreibung der Daten mittels allgemeiner Ausführungen zum Untersuchungsgegenstand und verschiedener SQL-Abfragen über dessen Struktur und Inhalt, soll nun in diesem Abschnitt die Datenbasis anhand graphischer Untersuchungsmethoden anschaulich dargestellt werden.

Dabei soll es aber nicht nur darum gehen, die Abfrageergebnisse des vorherigen Kapitels ansprechend zu visualisieren. Vielmehr soll hier bereits mit Blick auf den nachfolgenden Hauptteil praktisch gezeigt werden, wie bei Analysen methodisch vorzugehen ist. Die Analysen selbst sind dabei in ihrem Umfang kurz gehalten.

#### Beispiele mit Hinweisen zur Durchführung von Analysen

Der Ablauf von Analysen orientiert sich an dem zuvor im Kapitel *Grundzüge des verwendeten Vorgehensmodells* vorgestellten **Vorgehensmodell** für Datenanalysen und ist demnach unterteilt in Datenaufbereitung, Datenanalyse und Evaluierung.

Anhand einer beispielhaften ersten Untersuchung soll nun dieser Ablauf in ein Schema konkreter Verfahrensschritte übersetzt werden, das wiederum i. S. einer Vorlage referenziert werden kann.<sup>22</sup>

Um das Vorgehen vollständig aufzuzeigen, den Text hier jedoch nicht mit Nebeninformationen zu überladen, werden die für dieses Analysebeispiel notwendigen Vorbereitungen im [Anhang](#) im einleitenden Prolog exemplarisch vorgestellt. Die untenstehende Datenaufbereitung schließt sich hieran nahtlos an.

#### Datenaufbereitung

Gegenstand der Untersuchung sind an dieser Stelle nur Datensätze mit einer userid größer als 0. Damit werden jene Benutzer bei der Analyse nicht beachtet, die einer Beobachtung ihres Verhaltens nicht zugestimmt haben (userid = -2) oder die im Bachelor-Studiengang Medieninformatik Online studierten (userid = -3).

---

```

1 # Konvertierung des Datentyps des Tabellenmerkmals timecreated
2 moodle_data['timecreated'] =
3     pd.to_datetime(moodle_data['timecreated'], unit='s')
4 moodle_data = moodle_data[moodle_data.userid > 0]
5 moodle_data

```

---

Listing 8: Auswahl der Arbeitsdaten

#### Datenanalyse: Menge der Log-Einträge pro Benutzer

---

```

1 # Spezifische Definitionen zur Darstellung der Visualisierung
2 plt.figure(figsize=(64, 36)) # Größe der Visualisierung (in inch)
3
4 # Visualisierung der Menge der Log-Einträge pro Benutzer
5 chart = sns.countplot(x=moodle_data.userid)

```

---

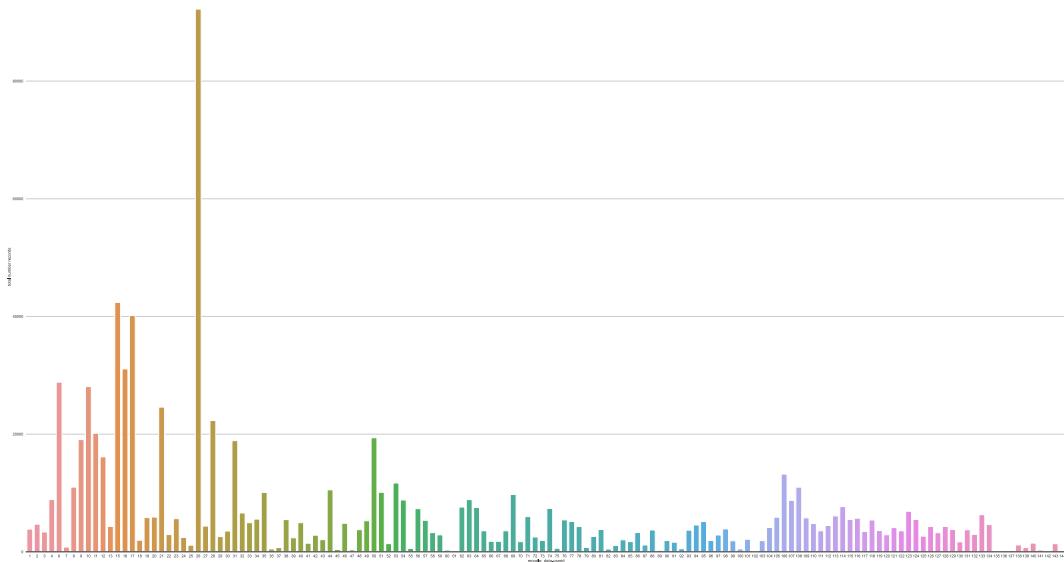
<sup>22</sup> s. zu dieser Arbeit beigefügten Jupyter Notebook Dokumente

```

6
7 # weitere Anweisungen zur Darstellung der Visualisierung
8 chart.grid(axis='y')
9 chart.set_axisbelow(True)
10 chart.set_xlabel('moodle_data.userid')
11 chart.set_ylabel('total number records')
12 chart.tick_params(left=False, bottom=False)
13 sns.despine(left=True)
14 plt.show()

```

Listing 9: Menge der Log-Einträge pro Benutzer

Abbildung 16.: Menge der Log-Einträge pro Benutzer ([s. Anhang](#))

Um in dieser Arbeit auch größere Visualisierungen leicht verständlich abbilden und evaluieren zu können, sollen diese im Hauptteil nach Möglichkeit nur in relevanten Ausschnitten inklusive einem Verweis auf den Anhang präsentiert werden. In Fällen wie oben, wo dieses dagegen wenig sinnvoll erscheint, weil z. B. eine Gesamtbetrachtung erfolgen soll, sind die Abbildungen insgesamt einfach zu verkleinern und mit einem Link auf das großformatige Original zu versehen. Ergänzend sei hier auch noch einmal auf die Plots in den beigefügten Jupyter Notebooks verwiesen.

### Evaluierung

Die obige Abbildung lässt erahnen, warum Visualisierungen für die Datenanalyse bestens geeignet sind: In der kompakten Darstellung zeigen sich z. B. die Benutzer mit minimalen oder maximalen Werten, wie auch die Häufung höherer Werte bei Benutzern mit einer niedrigen userid deutlich schneller als in jeder Ergebnistabelle.

Als Basis der folgenden Analyse diente erneut die oben im Listing [Auswahl der Arbeitsdaten](#) definierte Datenaufbereitung, d. h. die Benutzer, die der Beobachtung ihres Verhaltens nicht zugestimmt haben oder jene die im Bachelor-Studiengang Medieninformatik studierten, wurden bei der Untersuchung nicht berücksichtigt.

Aus Gründen der Vergleichbarkeit mit dem Ergebnis der korrespondierenden SQL-Abfrage zur [Menge der Benutzer pro Studiengang](#) und um die Größenunterschiede der Benutzermengen noch einmal besser verständlich aufzuzeigen, wird auch bei dieser Analyse der übergeordnete Studiengang 0 mitberücksichtigt.

Aus Gründen der Übersichtlichkeit werden im weiteren Verlauf der Arbeit die Anweisungen zur Darstellung von Visualisierungen nur noch in begründeten Fällen explizit angegeben. Bei Interesse können gerne die detaillierten Jupyter Notebook Dokumente eingesehen werden, die dieser Arbeit beiliegen.

### **Datenanalyse: Menge der Benutzer pro Studiengang**

---

```

1 # Ermittlung der Menge der Benutzer pro Studiengang
2 result = moodle_data.userid.groupby(moodle_data.Studiengang).nunique()
3 # Visualisierung der Menge der Benutzer pro Studiengang
4 chart = sns.barplot(x=result.index, y=result)

```

---

Listing 10: Menge der Benutzer pro Studiengang

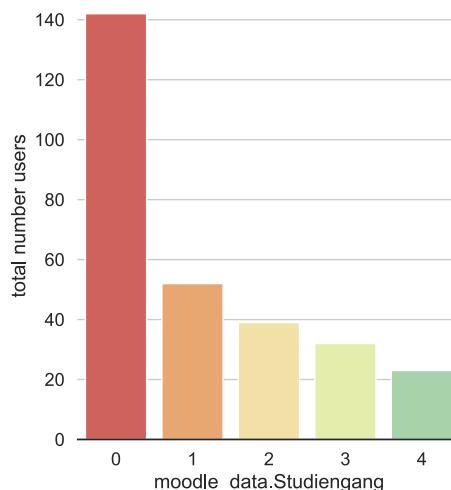


Abbildung 17.: Menge der Benutzer pro Studiengang

### **Evaluierung**

Die Abbildung zur Menge der Benutzer pro Studiengang präsentiert nicht nur die reinen Zahlen, die auch die entsprechende Ergebnistabelle im vorheigen Abschnitt bereits auflistete. Sie verdeutlicht darüberhinaus auch sehr schnell die Größenverhältnisse zwischen den einzelnen Werten des Diagramms. Dies ist ein weiterer großer Vorteil gegenüber Ergebnistabellen, deren Aussagen sich durch analytische Überlegungen manchmal erst recht langsam erschließen.

Wie eingangs erwähnt, sind die hier gezeigten ersten Untersuchungen einfach und nur wenig umfangreich. Bei komplexeren Aufgabenstellungen wie sie im folgenden Kapitel zu lösen sind, sind die Phasen der Datenaufbereitung bzw. Datenanalyse und Evaluierung dagegen häufig in mehreren Schritten wiederholt zu durchlaufen.

**Datenanalyse: Menge der Kurse pro Benutzer**


---

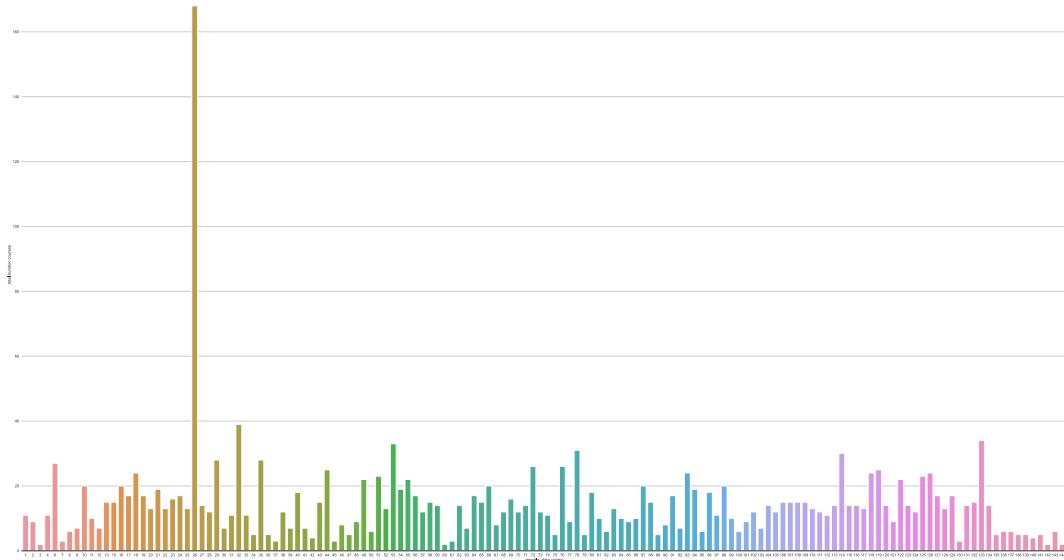
```

1 # Ermittlung der Menge der Kurse pro Benutzer
2 result = moodle_data.courseid.groupby(moodle_data.userid).nunique()
3 # Visualisierung der Menge der Kurse pro Benutzer
4 chart = sns.barplot(x=result.index, y=result)

```

---

Listing 11: Menge der Kurse pro Benutzer

Abbildung 18.: Menge der Kurse pro Benutzer ([s. Anhang](#))**Evaluierung**

Betrachtet man das folgende Diagramm, so fällt erneut der Benutzer mit der userid 26 auf. Wie schon im Plot zur [Menge der Log-Einträge pro Benutzer](#) überragt sein Wert den der anderen bei weitem und man könnte hier bereits vermuten, dass es sich dabei nicht um einen Studenten, sondern um einen Angehörigen des Hochschulpersonals handelt.

## 3. Analysen

Dieses Kapitel repräsentiert den eigentlichen Kern dieser Arbeit. Es beinhaltet ausführliche Beschreibungen zu allen Untersuchungen, die im Rahmen dieser Arbeit über die Identifikation von Studenten und deren typisches Verhalten angestrengt werden, und lässt dabei Schritt für Schritt die Gedanken sichtbar werden, die das jeweilige Vorgehen motivieren und begleiten

Aktivitätsbezogene Analysen zur Differenzierung des benutzerspezifischen Verhaltens machen den Anfang. Sie zeigen auf, wie anhand bestimmter Merkmalsausprägungen die Unterscheidung von Benutzergruppen möglich ist und bilden damit das Fundament auf dem danach die zeitbezogenen Untersuchungen studentischen Verhaltens jeweils aufbauen.

In Anbetracht eines sinnvollen Beitrags, den diese Arbeit zur Entwicklung des DiSEA-Projekts leisten möchte, sind die zeitlichen Betrachtungen dann so gewählt und in Fragen ausformuliert, dass sie im Hinblick auf eine spätere Bestimmung von Kriterien zur Messung von Studienerfolgen von Nutzen sein können:

1. Wie lassen sich Studenten nach der zeitlichen Lokalität der gezeigten Lern- und Kommunikationsaktivitäten einordnen?
2. Auf welche Weise lassen sich Studenten nach der Kontinuität ihres Handelns in verschiedene Gruppen unterscheiden?
3. Inwiefern kann man Studenten nach der Dynamik ihres Verhaltens beurteilen und in unterschiedliche Kategorien einteilen?

Mit Beantwortung dieser grundlegenden Fragen in jeweils eigenen Kapiteln werden schließlich wesentliche zeitliche Aspekte des studentischen Verhaltens sichtbar, die im Anschluss auch eine sinnvolle Kategorisierung der Studenten erlauben.

Abschließende getrennte Untersuchungen des Lern- und Kommunikationsverhaltens beurteilen deren Unterschiede und geben Auskunft über deren jeweiligen Einfluss auf das Gesamtverhalten.

### 3.1. Identifikation von Studenten

Im Grundlagenkapitel zur [Datenbasis](#) ist bereits mehrfach angeklungen, dass die im Rahmen dieser Arbeit zu betrachtenden Benutzer durchaus ganz verschiedenen Personengruppen angehören können.

Neben den Studenten, deren Lern- und Kommunikationsverhalten ganz allein den Untersuchungsgegenstand darstellt, gibt es im Umfeld der Hochschule viele weitere Personen, deren Verhalten zwar möglicherweise im Kontext studentischer Aktivitäten eine gewisse Bedeutung zukommt, dieses für sich betrachtet in dieser Arbeit aber nicht weiter von Interesse sein sollte.

Dass die Identifikation von Studenten demnach eine notwendige Voraussetzung für die weiteren Untersuchungen darstellen würde, war also früh ersichtlich und so stellte sich damit auch unmittelbar die Frage, ob und wie sich Studenten mithilfe analytischer Untersuchungen des Datenbestands tatsächlich als eine ganz eigene Benutzergruppe identifizieren ließen.

Eine erste Überlegung war, die Benutzergruppen über die in Moodle definierten Rollen zu unterscheiden. Nach Informationen der Hochschule, wird in Moodle die Rolle eines Benutzers jedoch nur auf Kursebene festgelegt. Dies bedeutet, dass ein Benutzer, unabhängig von seinem offiziellen Status, in mehreren Kursen auch verschiedene Rollen einnehmen kann. Somit war schnell offensichtlich, dass sich diese Rollenzuweisung nicht als zuverlässiges Unterscheidungskriterium eignete.

Gesichert war hingegen der Umstand, dass in der Gesamtmenge der Benutzer insgesamt 75 *einzelne Studenten* enthalten sind.<sup>23</sup> Diese vom Projektteam bestätigte Auskunft war zur Identifikation der Studenten wiederum nützlich, da sich daran schließlich die Qualität der Analyseergebnisse jederzeit messen lassen konnte.

### 3.1.1. Ermittlung des Benutzerstatus

Aber nicht nur die Qualität der Ergebnisse, sondern auch die des Datenbestands besitzt bei der Datenanalyse eine enorme Bedeutung. Daten müssen zwingend in einer entsprechend hohen Qualität vorliegen, damit im Nachhinein die gewonnenen Analyseergebnisse als fundiert gelten dürfen.

Wichtige Kriterien der Datenqualität sind u. a. die Vollständigkeit, die Richtigkeit sowie die Eindeutigkeit der Daten (Wang & Strong, 1996). Daneben ist aber auch die eigentliche Relevanz von grundlegendem Interesse, da die Einbeziehung nicht relevanter Daten in eine Untersuchung die daraus resultierenden Ergebnisse stark negativ beeinflussen kann.

Mit Blick auf den Untersuchungsgegenstand dieser Arbeit – *das studentische Lern- und Kommunikationsverhalten* – wurde folglich mit dem Betreuerteam entschieden, nach einer Unterscheidung von Studenten und anderen Benutzern, jene Datensätze die sich nicht sicher auf studentische Aktivitäten beziehen, zu kennzeichnen und bei den anschließenden Untersuchungen gesondert zu behandeln.

Bedeutete die Identifikation der Studenten also die Grundlage für alle weiteren

---

<sup>23</sup> Die genannte Menge an Studenten wurde im Rahmen einer Umfrage festgestellt, bei der Benutzer um ihr Einverständnis zur Nutzung ihrer Daten im Rahmen des DiSEA-Projekts gebeten wurden.

Analysen, so musste diese demnach zwingend ein hinreichend gesichertes Ergebnis erbringen. Die praktischen Schritte bei den Untersuchungen orientierten sich dabei erneut an dem in den Grundlagen vorgestellten [Vorgehensmodell](#).

## *Datenaufbereitung*

Gegenstand der Untersuchung waren initial nur Datensätze mit einer userid > 0, d.h. es wurden nur Einzelbenutzer betrachtet ([s. Listing im Grundlagenkapitel](#)).

## Datenanalyse: Untersuchungen verschiedener Tabellenmerkmale

Mehrere Überlegungen wie auch die Reflektion des eigenen Benutzerverhaltens als Student orientierten sich zunächst an der Frage, wie sich ein typisch studentisches Verhalten tatsächlich darstellen könnte und führten so zu einigen Untersuchungen über die Merkmale des Datenbestands, u. a. auch zum Merkmal *action*:

```
1 # Visualisierung der Mengen aller Actions in der Gesamtbetrachtung
2 chart = sns.histplot(data=moodle_data.action.sort_values(),
3                      stat='percent', color='#6DAEE2', alpha=1)
```

Listing 12: Mengen aller Actions in der Gesamtbetrachtung

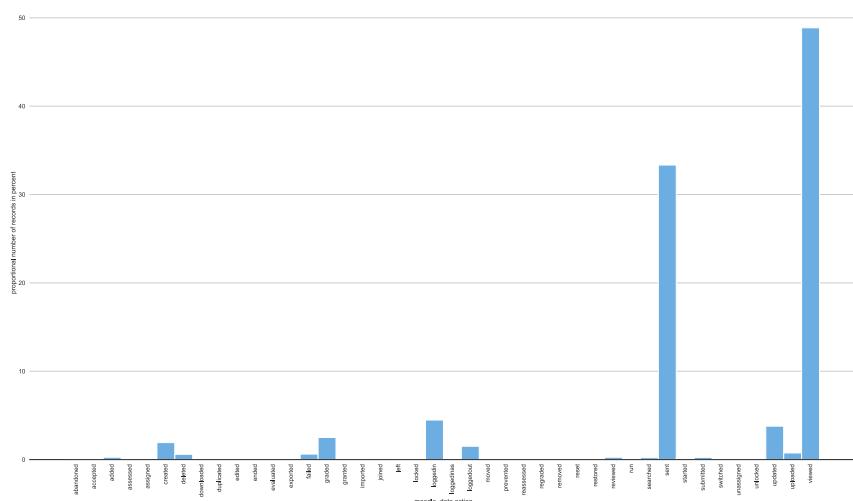


Abbildung 19.: Mengen aller Actions in der Gesamtbetrachtung (s. Anhang)

## *Evaluierung*

Während manche Betrachtungen gerade in zeitlicher Hinsicht auf den ersten Blick wenig aufschlussreiche Ergebnisse lieferten, fiel bei Untersuchung des Merkmals *action* sofort auf, dass Benutzer neben einem hohen Anteil an sent-Actions einen noch höheren Anteil an Werten vom Typ *viewed* aufwiesen. Mit einem Anteil von insgesamt über 80% bestimmten diese beiden Aktivitäten die Gesamtaktivität aller Benutzer im Untersuchungszeitraum.

Aus diesem Ergebnis nun schon ein typisch studentisches Verhalten abzuleiten war zwar nicht möglich, es widerlegte jedoch auch nicht direkt meine Vermutung, dass Studenten oft als Leser z. B. von Lehrmaterialien, Forumsdiskussionen oder

Mitteilungen auftreten und ganz nebenbei deckte es sich ebenfalls weitgehend mit meinem eigenen Verhalten als Studenten.

Auch inspirierte das Ergebnis zu der Frage, wie sich gerade die Menge der viewed-Actions tatsächlich über das Semester hinweg auf die Benutzer verteilte.

### Datenaufbereitung

Die Datenauswahl umfasste erneut alle Datensätze mit einer userid > 0, d.h. es wurden nur Einzelbenutzer betrachtet ([s. Listing im Grundlagenkapitel](#)).

### Datenanalyse: Betrachtung der Menge an viewed-Actions pro Benutzer

---

```
1 md = moodle_data # Umbenennung zur kompakteren Darstellung des Codes
2 # Visualisierung der Menge der viewed-Actions pro Benutzer
3 chart = sns.countplot(x=md.userid[md.action == 'viewed'], alpha=1)
```

---

Listing 13: Menge der viewed-Actions pro Benutzer

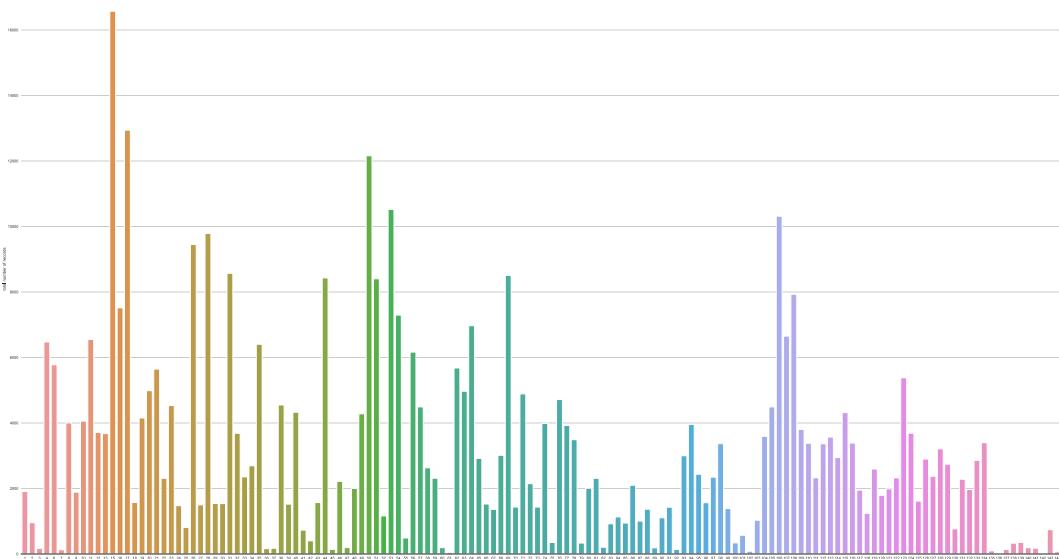


Abbildung 20.: Menge der viewed-Actions pro Benutzer ([s. Anhang](#))

### Evaluierung

Wie im obigen Diagramm zu erkennen ist, gibt es einige Benutzer denen relativ hohe Mengen an viewed-Actions zuzuordnen sind. Gleichzeitig finden sich aber auch Personen, die nur eine geringe Anzahl aufweisen. Ob und wie sich aus dieser einfachen Differenzierung vielleicht schon ein Hinweis ableiten lassen könnte auf ein echtes benutzertypisches Verhalten, war nun die spannende Frage.

Um diese Frage für die Gesamtheit aller Benutzer sicher beantworten zu können, war zum einen zu klären, welchen Anteil die viewed-Actions an der Gesamtaktivität der jeweiligen Benutzer tatsächlich hatte. Zum anderen war es aber auch notwendig, eine variable Vergleichsgröße zu definieren anhand derer es möglich war Benutzer beliebig ein- oder auszuschließen.

Auf diese Weise ließe sich dann auch konkret bestätigen oder widerlegen, ob die oben ermittelten Benutzer mit den hohen Mengen an Werten vom Typ viewed wirklich auch diejenigen waren, deren Verhalten maßgeblich durch die höheren viewed-Actions bestimmt war.

Aufschluss über all die Fragen gab schließlich die folgende SQL-Anweisung, die auf der gleichen Datenauswahl wie die vorausgehende Analyse basierte (s. WHERE-Klausel unten). Die Vergleichsgröße wurde dabei anfänglich mit einem viewed-Anteil von 50% an der Gesamtaktivität (s. HAVING-Klausel unten) definiert, da dies ziemlich genau dem zuvor ermittelten **Gesamtdurchschnitt** entsprach. Danach wurde sie in einem iterativen Prozess, begleitet von Einzelbenutzerbetrachtungen, in mehreren Schritten angepasst.<sup>24</sup>

#### Datenanalyse: Anteile der viewed-Actions an der Gesamtaktivität

```

1  SELECT mdl1.userid,
2    COUNT(mdl1.action) AS 'all_actions',
3    (SELECT COUNT(md2.action) FROM moodle_data md2
4      WHERE mdl1.userid = md2.userid AND md2.action = 'viewed')
5      AS 'viewed_action',
6    (SELECT COUNT(md2.action) FROM moodle_data md2
7      WHERE mdl1.userid = md2.userid AND md2.action != 'viewed')
8      AS 'other_actions',
9    (SELECT COUNT(md2.action) FROM moodle_data md2
10     WHERE mdl1.userid = md2.userid AND md2.action = 'viewed') /
11     COUNT(action) AS 'percentage'
12   FROM moodle_data mdl1
13   WHERE userid > 0
14   GROUP BY mdl1.userid
15   HAVING percentage > 0.5
16   ORDER BY percentage DESC;
```

Listing 14: Anteil der viewed-Actions an der Gesamtaktivität

userid	all_actions	viewed_action	other_actions	percentage
64	7544	6970	574	0.9239
53	11699	10520	1179	0.8992
40	4953	4328	625	0.8738
69	9756	8507	1249	0.8720
91	1641	1430	211	0.8714
...	...	...	...	...

99 rows in set (6,49 sec)

Abbildung 21.: Anteil der viewed-Actions an der Gesamtaktivität (s. Anhang)

#### Evaluierung

Im Ergebnis der obigen SQL-Abfrage zeigten sich 99 Benutzer (ca. 70% der Gesamtbenutzeranzahl), bei denen die Menge der viewed-Actions mehr als die Hälfte der Gesamtaktivität ausmachte und die nun anhand von Stichproben exemplarisch zu prüfen waren. In einem stetigen Wechsel folgten dann weitere Abfragen immer mit angepasster Vergleichsgröße und weiteren Betrachtungen einzelner Benutzer.

Im Zuge dieses Vorgehens zeigten die zahlreichen Einzelanalysen, die ferner Art und Umfang weiterer Merkmale wie auch den zeitlichen Kontext betrachteten, dass

<sup>24</sup> Siehe auch die zu dieser Arbeit beigefügten Jupyter Notebook Dokumente zu Einzelanalysen.

sich die Trefferquote der SQL-Abfrage durch paralleles Anheben des Grenzwerts in der HAVING-Klausel sukzessive verbessern ließ. Es wurde dabei aber auch offensichtlich, dass hierdurch zunehmend mehr mutmaßliche Studenten ausgeschlossen wurden. Bei einem Grenzwert von 0.8 wurde schließlich der iterative Prozess des stetigen Testens und Optimierens beendet: Mit 35 mutmaßlichen Studenten lag zwar ein sorgfältig getestetes und damit gesichertes Ergebnis vor, rein zahlenmäßig betrachtet war es aber unzureichend.

Daneben sind bei den Einzelanalysen noch weitere Phänomene sichtbar geworden: Manche Benutzer unterschieden sich in ihren Kursprofilen, d. h. in der Art und der Menge ihrer Kurse deutlich, verhielten sich in Bezug auf andere aber wiederum recht ähnlich. Genau diese Besonderheit war interessanterweise aber auch bei anderen Benutzeraktivitäten zu beobachten. Auch hier schien es solche Unterschiede und Gemeinsamkeiten gleichzeitig zu geben, was zu dem Gedanken führte, dass gerade die genauere Betrachtung weiterer spezifischer Aktivitäten eine verbesserte Typisierung und folglich auch eine Unterscheidung von Studenten und Anderen ermöglichen könnte.

Die thematische Ausrichtung für die weiteren Schritte war damit klar. Fraglich war nur, ob an dieser Stelle wieder eine Gesamtbetrachtung aller Benutzer ratsam war oder ob nicht mittlerweile eine bessere Option bestünde. Dies war auch ein passender Moment, die praktischen Untersuchungen einmal zu pausieren und zu reflektieren, wie bei einer Datenexploration methodisch sinnvoll vorzugehen ist.

*Hier noch ein oder zwei Sätze zur explorativen Datenanalyse mit Literaturangaben bzw. Verweis auf das entsprechende Grundlagenkapitel ergänzen ...*

Nach kurzer Überlegung stand fest, dass eine Betrachtung des gesamten Datenbestands und damit verbunden eine Rückkehr zum Startpunkt der Analysen zwar möglich wäre, die effiziente Nutzung bereits gewonnener Erkenntnisse als sicherer Ausgangspunkt für neue Schritte aber zu bevorzugen ist.

### Datenaufbereitung

Gemäß den bei den durchgeführten Einzelanalysen erlangten Einsichten, wurden nun also für die weiteren Untersuchungen gezielt bestimmte Benutzer ausgewählt und deren Log-Einträge in neuen Datensets für Studenten und Andere (Others) zusammengefasst.

---

```

1 md = moodle_data # Umbenennung der Variable, um den Code zu verkürzen
2 records_students = [md[md.userid == 1], md[md.userid == 13],
3                      md[md.userid == 18], md[md.userid == 19],
4                      md[md.userid == 20], md[md.userid == 22],
5                      md[md.userid == 23], md[md.userid == 24],
6                      md[md.userid == 25], md[md.userid == 38]]
7 md_students = pd.concat(records_students)

```

---

Listing 15: Auswahl der Log-Einträge der Studenten

---

```

1 md = moodle_data # Umbenennung der Variable, um den Code zu verkürzen
2 records_others = [md[md.userid == 2], md[md.userid == 4],
3                     md[md.userid == 6], md[md.userid == 9],
4                     md[md.userid == 10], md[md.userid == 11],
5                     md[md.userid == 27], md[md.userid == 28],
6                     md[md.userid == 29], md[md.userid == 32]]
7 md_others = pd.concat(records_others)

```

---

Listing 16: Auswahl der Log-Einträge der Anderen

Anschließend wurden die spezifischen Aktivitäten der einzelnen Benutzergruppen ermittelt und ausgewertet sowie in einem gemeinsamen Datenset kombiniert.

---

```

1 # Ermittlung der Menge der Log-Einträge pro Action
2 students_actions = md_students.action.groupby(md.action).count()
3 others_actions = md_others.action.groupby(md.action).count()
4
5 # Erstellung eines kombinierten Datensets für Studenten und Andere
6 users_actions = pd.concat([students_actions, others_actions], axis=1,
7                           keys=['students', 'others']).sort_index()
8
9 # Ersetzung von NaN-Werten durch den Wert 0
10 users_actions = users_actions.fillna(0)
11
12 # Ausgabe des kombinierten Datensets
13 display(users_actions)

```

---

Listing 17: Konkatenation der Datensets von Studenten und Anderen

Die Tabelle unten zeigt in einem Ausschnitt das fertig aufbereitete Datenset, das in der nachfolgenden Analyse dann noch einmal visualisiert und interpretiert wurde.

action	students	others
abandoned	0.0	2.0
accepted	28.0	3.0
added	21.0	403.0
created	392.0	2248.0
deleted	46.0	303.0
...	...	...
submitted	443.0	3.0
switched	0.0	16.0
updated	88.0	5106.0
uploaded	344.0	743.0
viewed	22718.0	26185.0

Abbildung 22.: Kombiniertes Datenset für Studenten und Andere ([s. Anhang](#))

#### Datenanalyse: Menge der Log-Einträge pro Aktivität und Benutzergruppe

---

```

1 # Visualisierung der Menge der Log-Einträge pro Action
2 result = users_actions.stack().reset_index().set_index('action').
3         rename(columns={'level_1': 'students', 0: 'others'})
4 chart = sns.barplot(x=result.index, y='others',
                      data=result, hue='students')

```

---

Listing 18: Menge der Log-Einträge pro Aktivität und Benutzergruppe

Die Visualisierung unten veranschaulicht noch einmal deutlich die bereits in der Ergebnistabelle sichtbaren Wertdifferenzen. Zu beachten ist, dass die Balken für die viewed-Actions beim Wert 5500 oben abgeschnitten wurde, um die Differenzen der anderen Werte in ihren Proportionen besser erkennen zu können.

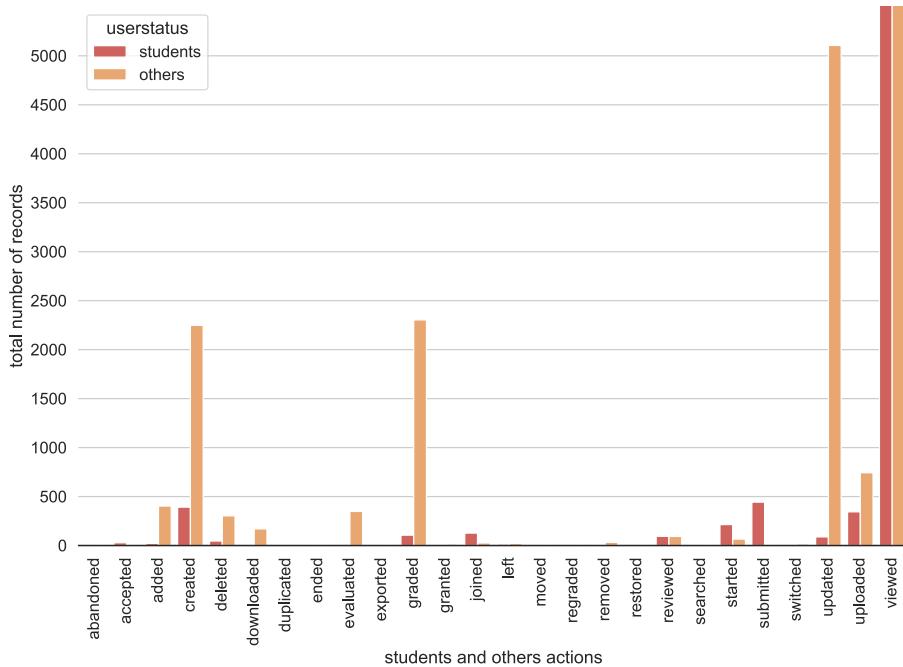


Abbildung 23.: Menge der Log-Einträge pro Aktivität und Benutzergruppe

### Evaluierung

Wie das oben aufbereitete Datenset und die Grafik auf den ersten Blick zeigen, überragen z. B. bei den Werten *created*, *graded* und *updated* die Log-Einträge der Anderen die der Studenten um ein Vielfaches, während man erst bei genauerem Hinsehen erkennt, dass sich das Verhältnis beim Wert *submitted* andersherum darstellt.

Damit bestätigte also die Untersuchung mittels vordefinierter Benutzergruppen die zuvor formulierte Vermutung, dass sich Studenten und Andere durch die Art und den Umfang ihrer Aktivitäten unterscheiden.

Was nun zwangsläufig folgen musste, war die Beantwortung der Frage, ob und wie sich mit dieser Erkenntnis die Studenten im Gesamtkontext auch auf direktem Wege identifizieren ließen. Hierzu erschien es ratsam, die Mengen der Log-Einträge zu den einzelnen Aktivitäten erneut zu prüfen. Dabei kamen rasch auch noch die Werte *added*, *deleted* und *evaluated* in den Fokus, weil sie ebenfalls selbst eine gewisse Anzahl an Log-Einträgen besaßen, andererseits jedoch auch eine mindestens genauso beachtliche Mengendifferenz aufwiesen.

In dieser Phase waren noch einige weitere Untersuchungen notwendig. Insbesondere zum besseren Verständnis der Benutzeraktivitäten wurden wiederholt Einzelanalysen durchgeführt, bis schließlich deutlich wurde, dass die mögliche Lösung des Problems vielleicht schon vorlag: Die beabsichtigte direkte Identifikation von Studenten müsste, ähnlich dem Vorgehen bei der Betrachtung der *viewed*-Actions, über eine dem Gesamtkontext angemessene Gewichtung ausgewählter Aktivitäten herzustellen sein.

Einzelnen oder in Gruppen zusammengefasst müssten die verschiedenen Mengen

an Log-Einträgen zu den Aktivitäten wie Stellschrauben justiert werden können, um die Studenten aus der Gesamtmenge der Benutzer herauszufiltern. Dabei sollte es von Vorteil sein, dass mit der Summe der anteiligen Mengen an added-, created-, deleted-, evaluated-, graded- und updated-Actions einerseits sowie der anteiligen Menge an submitted-Actions andererseits zwei Größen zur Verfügung standen, die grundsätzlich gegenläufig waren.

Für die praktische Umsetzung dieser Idee schien es am einfachsten, das vormals verwendete SQL-Statement zur Selektion von Benutzern mit einem hohen Anteil an viewed-Actions entsprechend zu modifizieren.

### Datenaufbereitung

Die Datenauswahl umfasste erneut alle Datensätze mit einer userid > 0, d.h. es wurden nur Einzelbenutzer betrachtet (s.u. die WHERE-Klausel im SQL-Listing).

Die obige [SQL-Anweisung zur Betrachtung der viewed-Actions](#) wurde einst西路s hinsichtlich der Unterabfragen geändert. Vielmehr wurden aber auch in dem oben beschriebenen iterativen Prozess die für die Selektion von Studenten relevanten Größen in der HAVING-Klausel angepasst.

### Datenanalyse: Identifikation von Studenten

---

```

1  SELECT userid,
2      COUNT(action) AS "all_actions",
3      (SELECT COUNT(action) FROM moodle_data md2
4          WHERE md2.userid = md1.userid AND md2.action = "added")
5          AS "added",
6      (SELECT COUNT(action) FROM moodle_data md2
7          WHERE md2.userid = md1.userid AND md2.action = "created")
8          AS "created",
9      (SELECT COUNT(action) FROM moodle_data md2
10         WHERE md2.userid = md1.userid AND md2.action = "deleted")
11         AS "deleted",
12     (SELECT COUNT(action) FROM moodle_data md2
13         WHERE md2.userid = md1.userid AND md2.action = "evaluated")
14         AS "evaluated",
15     (SELECT COUNT(action) FROM moodle_data md2
16         WHERE md2.userid = md1.userid AND md2.action = "graded")
17         AS "graded",
18     (SELECT COUNT(action) FROM moodle_data md2
19         WHERE md2.userid = md1.userid AND md2.action = "submitted")
20         AS "submitted",
21     (SELECT COUNT(action) FROM moodle_data md2
22         WHERE md2.userid = md1.userid AND md2.action = "updated")
23         AS "updated"
24 FROM moodle_data md1
25 WHERE userid > 0
26 GROUP BY userid
27 HAVING ((added + created + deleted + evaluated +
28           graded + updated) < (0.25 * all_actions))
29     AND (submitted > (0.001 * all_actions));

```

---

Listing 19: Identifikation von Studenten

### Evaluierung

Durch sorgfältiges Testen mittels Einzelanalysen und Optimieren der Vergleichsgrößen im stetigen Wechsel, konnten schließlich die in der obigen Ergebnistabelle

userid	all_actions	added	created	deleted	evaluated	graded	submitted	updated
1	3865	0	43	0	0	0	12	20
13	4330	2	40	2	0	15	51	11
18	1978	2	17	1	0	0	24	14
19	5823	2	77	10	0	24	75	11
20	5909	3	58	3	0	19	55	10
132	2973	2	112	19	0	0	6	279
134	4629	2	146	22	0	0	12	304
136	33	0	0	0	0	0	2	0
142	10	0	0	0	0	0	1	0
143	1387	0	11	0	0	0	4	2

72 rows in set (10,24 sec)

Abbildung 24.: Identifikation von Studenten (s. Anhang)

angezeigten 72 Benutzer ausreichend sicher als Studenten identifiziert werden.

Ob es sich bei diesen 72 Studenten nun um eine exakte Teilmenge der bei einer Umfrage sicher ermittelten 75 Studenten handelt oder im Ergebnis eventuell auch Studenten erfasst wurden, die eventuell erst später auf Anfrage<sup>25</sup> der Beobachtung ihres Verhalten zugestimmt haben, wäre bei Bedarf noch zu prüfen, soll hier für die weiteren Untersuchungen in dieser Arbeit jedoch nicht von Interesse sein.

Weitere Überprüfungen in Form neuer Einzelbetrachtungen könnten allerdings notwendig werden, sollte die beschriebene Methodik z. B. bei einer Untersuchung auf einem ganz neuen Datenbestand doch Fehler aufweisen. Ein grundsätzliches *Overfitting*, d. h. eine zu genaue Anpassung eines statistischen Modells an gewisse Besonderheiten eines Datensets aus der eine mangelhafte Übertragbarkeit resultiert (Dietterich, 1995), müsste bei der Einfachheit der gewählten Vergleichsgrößen aber auszuschließen sein.

### 3.1.2. Kennzeichnung des Benutzerstatus

Um im weiteren Verlauf der Analysen die Auswahl der identifizierten Studenten zu vereinfachen und damit auch den gesamten Arbeitsprozess zu beschleunigen, wurde entschieden, die identifizierten Studenten durch ein neues Tabellenmerkmal dauerhaft zu kennzeichnen.

Hierzu wurden zunächst die Ergebnisse aus der Abfrage zur Identifikation von Studenten in eine neue Tabelle *moodle\_data\_students* übernommen. Das vorherige SQL-Statement war hierfür nur um zwei Zeilen Code zu ergänzen:

#### Erstellen der neuen Tabelle moodle\_data\_students

---

```

1 CREATE TABLE moodle_data_students
2 AS
3 /*
4 SQL-Statement zur Identifikation von Studenten
5 */

```

---

Listing 20: Erstellen der neuen Tabelle moodle\_data\_students

<sup>25</sup> Um die im Rahmen der Umfrage erfassten Daten um einen größeren Kontext zu ergänzen, wurden Benutzer auch direkt kontaktiert, ohne jedoch deren offiziellen Status zu dokumentieren.

Hiernach wurden in der Relation *moodle\_data* die neuen Merkmale *userstatus* und *relateduserstatus* mit dem Default-Wert *other* eingefügt, und dieser in einem letzten Schritt entsprechend den folgenden Anweisungen angepasst (die letzte Anweisung dient einer einfacheren Selektion von Aktivitäten ohne Personenbezug):

### *Kennzeichnung von Studenten*

---

```

1 UPDATE moodle_data SET userstatus = 'student'
2 WHERE userid IN (SELECT userid FROM moodle_data_students);
3
4 UPDATE moodle_data SET relateduserstatus = 'student'
5 WHERE relateduserid IN (SELECT userid FROM moodle_data_students);
6
7 UPDATE moodle_data SET relateduserstatus = 'none'
8 WHERE relateduserid = 0;

```

---

Listing 21: Kennzeichnung von Studenten

Abschließende Prüfungen der durchgeführten Änderungen ergaben das erwartete Resultat: Alle Datensätze mit einer userid eines zuvor erkannten Studenten wurden vollständig und richtig aktualisiert.

### *Überprüfung der Änderungen auf Vollständigkeit*

---

```

1 SELECT DISTINCT userid FROM moodle_data
2 WHERE userstatus = 'student';

```

---

Listing 22: Überprüfung der Änderungen auf Vollständigkeit

userid
1
13
18
..
136
142
143

72 rows in set (2,39 sec)

Abbildung 25.: Überprüfung der Änderungen auf Vollständigkeit

### *Überprüfung der Änderungen auf Richtigkeit*

---

```

1 SELECT * FROM moodle_data
2 WHERE (userstatus != 'student')
3 AND (userid IN (SELECT userid FROM moodle_data_students));

```

---

Listing 23: Überprüfung der Änderungen auf Richtigkeit

Empty set (0,40 sec)

Abbildung 26.: Überprüfung der Änderungen auf Richtigkeit

### 3.1.3. Zusammenfassung

In diesem Teil der Arbeit wurde gezeigt, wie rein datenorientiert eine hinreichend gesicherte Identifikation von Studenten auf Basis der bereitgestellten Moodle-Daten möglich war. Anhand der beschriebenen Überlegungen, der durchgeführten Betrachtungen und der Auswertungen wurde detailliert der Weg beschrieben, der schließlich zur Lösung der Problems geführt hat. Schritt für Schritt wurden dabei folgende Auswahlkriterien ermittelt:

1. Studenten wurden nur als Einzelbenutzer betrachtet, d. h. sie mussten zuvor der Beobachtung ihres Verhaltens zugestimmt haben und durften außerdem nicht im Bachelor-Studiengang Medieninformatik Online aktiv gewesen sein.
2. Studenten verfügten im Vergleich zu Anderen über einen relativ hohen Anteil an viewed- und submitted-Actions.
3. Studenten besaßen im Vergleich zu Anderen einen relativ geringen Anteil an added-, created-, deleted-, evaluated-, graded- und updated-Actions.

In einem iterativen Prozess wurden die anteiligen Mengen der genannten Actions mithilfe bestimmter Faktoren wiederholt gewichtet und die daraus resultierenden Ergebnisse anhand von Einzelanalysen<sup>26</sup> exemplarisch geprüft bis schließlich eine Menge von insgesamt 72 Studenten bestätigt werden konnte.

Abschließend wurden die identifizierten Studenten mit ihren charakteristischen Aktivitätsprofilen in einer eigenen Relation zusammengefasst und im originären Datenbestand gekennzeichnet, so dass sie im weiteren Verlauf der Arbeit unmittelbar selektiert werden konnten.

## 3.2. Konkretisierung der zu untersuchenden Datenbasis

Nachdem zuvor die Identität von Studenten anhand ihrer Aktivitäten festgestellt werden konnte, sollte im weiteren Verlauf der Arbeit deren Verhalten in zeitlicher Hinsicht untersucht und ausgewertet werden.

Als Datenbasis für die anstehenden zeitbezogenen Analysen sollte erneut die *viewed*-Action dienen können, die im vorhergehenden Kapitel zur [Identifikation von Studenten](#) wegen ihres hohen Anteils schon als relevante Größe zur Bestimmung studentischen Verhaltens in Erwägung gezogen wurde.

### 3.2.1. Betrachtung von viewed-Action und viewed-Events

Problematisch war nur, dass die *viewed*-Action allein die für diese Arbeit geforderte getrennte Betrachtung des Lern- und Kommunikationsverhaltens nicht zuließ.

---

<sup>26</sup> Siehe auch die zu dieser Arbeit beigefügten Jupyter Notebook Dokumente zu Einzelanalysen.

Es stellte sich an dieser Stelle also die Frage, ob es vielleicht ein anderes Merkmal gab, das mit der *viewed*-Action in Beziehung stand, jedoch für eine differenziertere Analyse besser geeignet war.

Eine Antwort auf diese Frage gab das nachfolgende SQL-Statement, mit dessen Hilfe sich die zur *viewed*-Action korrespondierenden *viewed*-Events ermitteln ließen.

### Datenaufbereitung

Bei der Auswahl der Daten berücksichtigt wurden sowohl die Studenten, die selbst eine *viewed*-Action initiiert haben, als auch solche die mit einer *viewed*-Action einer anderen Person in Beziehung standen (s. u. die WHERE-Klausel im SQL-Listing).

### Datenanalyse: Ermittlung korrespondierender *viewed*-Events

---

```

1 SELECT eventname, COUNT(eventname) AS "total_number_records"
2 FROM moodle_data
3 WHERE (userstatus = 'student' OR relateduserstatus = 'student')
4     AND action = 'viewed'
5 GROUP BY eventname
6 ORDER BY total_number_records DESC;

```

---

Listing 24: Ermittlung korrespondierender viewed-Events

Die Entscheidung, für eine genauere Betrachtung des studentischen Verhaltens die Werte des Merkmals *eventname* heranzuziehen, ergab sich u. a. aus Beobachtungen in vorbereitenden Tests und zahlreichen Einzelanalysen.

Insbesondere durch exemplarische Überprüfung wurde also ersichtlich, dass die Eventnames nicht nur sprechender und präziser waren als andere mit der *viewed*-Action korrespondierenden Werte der Merkmale *objecttable* und *course\_module\_type*. Sie gaben vielmehr auch einen direkten Hinweis auf die im webbasierten Moodle-System verwendeten URLs und ließen so erahnen, welche Webseiten des Systems mit den bezeichneten events sehr wahrscheinlich in Verbindung stehen.

### Evaluierung

Die Ergebnistabelle zeigt mit 19 Eventnames nur einen Ausschnitt von etwa einem Drittel aller Werte, die mit dem Wert *viewed* des Merkmals *action* korrespondierten. Nicht angezeigt und damit auch bei weiteren Untersuchungen nicht berücksichtigt wurden dagegen beispielsweise Eventnames wie *\core\event\dashboard\_viewed*, *\mod\_assign\event\submission\_status\_viewed* oder *\core\event\user\_profile\_viewed*, die mithilfe der Moodle Event API<sup>27</sup> und des eigenen Moodle-Benutzerkontos geprüft wurden und demnach einem Lern- oder Kommunikationsverhalten nicht konkret zugeordnet werden konnten.

Dementgegen hat die systematische Prüfung der Eventnames in den mit *L* und *K* markierten Ergebnisseilen einen ausreichenden Hinweis erbracht, der mit hoher

---

<sup>27</sup> Siehe auch die Moodle Documentation (Moodle, 2022): [Moodle Event API, 07/2022](#)

eventname	total_number_records	
\core\event\course_viewed	69507	L
...	...	
\mod_resource\event\course_module_viewed	20113	L
\mod_assign\event\course_module_viewed	15088	L
\mod_forum\event\discussion_viewed	14237	K
\mod_quiz\event\attempt_viewed	13605	L
\mod_forum\event\course_module_viewed	12146	K
...	...	
\mod_url\event\course_module_viewed	6636	L
...	3198	L
\mod_quiz\event\course_module_viewed	...	
\mod_page\event\course_module_viewed	2878	L
\core\event\message_viewed	2036	K
\mod_wiki\event\course_module_viewed	1790	L
\mod_wiki\event\page_viewed	1568	L
...	...	
\mod_choice\event\course_module_viewed	1336	L
...	...	
\mod_folder\event\course_module_viewed	1186	L
...	...	
\mod_glossary\event\course_module_viewed	920	L
...	...	
\mod_workshop\event\course_module_viewed	518	L
...	...	
\mod_bigbluebuttonbn\event\recording_viewed	388	L
...	...	
\mod_chat\event\course_module_viewed	119	K
...	...	
\mod_chat\event\sessions_viewed	66	K
...	...	

62 rows in set (0,39 sec)

Abbildung 27.: Ermittlung korrespondierender viewed-Events

Wahrscheinlichkeit auf eine *Lernaktivität* (L) bzw. eine *Kommunikationsaktivität* (K) schließen ließ.

In Summe ergaben die Mengen in den mit L markierten Ergebnisseilen so einen Anteil von 59,67% an der Gesamtmenge aller Datensätze, die mit viewed-Events in Beziehung standen. Dies sollte als solide Basis für weitere Betrachtungen des Lernverhaltens in dieser Arbeit ausreichend sein.

### 3.2.2. Entscheidung für viewed-Events als Grundlage

Da sent-Actions in der Regel ebenfalls in hohem Maße zum kommunikativen Austausch beitragen und im vorherigen Kapitel bereits große Mengen an sent-Actions festgestellt wurden, war an dieser Stelle jedoch noch unklar, wie es sich mit den in der Tabelle mit K gekennzeichneten viewed-Events verhielt. Aufschluss hierüber sollte die nachfolgende SQL-Abfrage geben.

#### Datenaufbereitung

Die Datenauswahl wurde analog zu der vorhergehenden Analyse getroffen, nur wurden dieses mal Studenten betrachtet, die selbst eine *sent*-Action initiiert haben oder mit einer *sent*-Action einer anderen Person in Beziehung standen (s. u. die WHERE-Klausel im SQL-Listing).

#### Datenanalyse: Ermittlung korrespondierender sent-Events

---

```

1 SELECT eventname, COUNT(eventname) AS "total_number_records"
2 FROM moodle_data
3 WHERE (userstatus = 'student' OR relateduserstatus = 'student')
4     AND action = 'sent'
5 GROUP BY eventname
6 ORDER BY total_number_records DESC;

```

---

Listing 25: Ermittlung korrespondierender sent-Events

eventname	total_number_records
\core\event\notification_sent	38762
\core\event\message_sent	1255
\core\event\group_message_sent	176
\mod_chat\event\message_sent	20

4 rows in set (0,34 sec)

Abbildung 28.: Ermittlung korrespondierender sent-Events

Im Ergebnis zeigte sich, dass neben den hohen Zahlen an *notification*-Events, die sich aber nur auf automatisch generierte Nachrichten des Moodle-Systems<sup>28</sup> bezogen, nur relativ wenige andere *sent*-Events protokolliert wurden.

Somit stand fest, dass die vorab ermittelten viewed-Events den weitaus größten Teil des studentischen Kommunikationsverhaltens repräsentierten und die hiermit in Beziehung stehenden Datensätze fortan als Grundlage weiterer Untersuchungen dienen konnten.

### 3.3. Lokalität des Lern- und Kommunikationsverhaltens

Unter Bezugnahme auf die im vorausgegangenen Abschnitt ermittelte Datenbasis sollte es nun darum gehen, die erste der drei eingangs formulierten Grundfragen zu beantworten:

Wie lassen sich Studenten nach der zeitlichen Lokalität der gezeigten Lern- und Kommunikationsaktivitäten einordnen?

Im Fokus der Betrachtung stand hier also das zeitliche Auftreten der studentischen Aktivitäten, das im Ergebnis auch belegen sollte, zu welchen Zeiten die Studenten über das Semester hinweg bevorzugt arbeiteten.

Wie Janneck et al. (2021) in ihrem Paper schreiben, unterscheiden sich Studenten in digitalen Studienformaten von solchen in Präsenzstudiengängen insbesondere durch ihre Berufstätigkeit und familiäre Verpflichtungen. Insofern ließ sich für den konkreten Fall vermuten, dass die studentische Aktivität überwiegend in der verbleibenden freien Zeit stattfinden musste, also u. a. an Wochenenden und in den Abendstunden.

---

<sup>28</sup> Siehe auch die Moodle Documentation (Moodle, 2022): [Moodle Messaging Notifications, 06/2022](#)

### 3.3.1. Betrachtung des studentischen Verhaltens auf Wochenbasis

Mit Blick auf den häufig durch die Lehrveranstaltungen bedingten wöchentlichen Arbeitsrhythmus wurde mit Analysen auf Wochenbasis begonnen.

#### Datenaufbereitung

Bei der Auswahl der Daten berücksichtigt wurden sowohl die Studenten, die selbst ein *viewed*-Event initiiert haben, als auch solche die mit einem *viewed*-Event einer anderen Person in Beziehung standen (s. das nachfolgende Listing zur Ergänzung des Merkmals *behaviour* und die Auswahl spezifischer Werte dieses Merkmals im anschließenden Listing zur *Definition der Arbeitsdaten*).

---

```

1 md['behaviour'] = 'other'
2 md.loc[(md.eventname == r'\core\event\course_viewed'),
3         ['behaviour']] = 'learning'
4 md.loc[(md.eventname == r'\mod_resource\event\course_module_viewed'),
5         ['behaviour']] = 'learning'
6 md.loc[(md.eventname == r'\mod_assign\event\course_module_viewed'),
7         ['behaviour']] = 'learning'
8 md.loc[(md.eventname == r'\mod_quiz\event\attempt_viewed'),
9         ['behaviour']] = 'learning'
10 md.loc[(md.eventname == r'\mod_url\event\course_module_viewed'),
11        ['behaviour']] = 'learning'
12 md.loc[(md.eventname == r'\mod_quiz\event\course_module_viewed'),
13        ['behaviour']] = 'learning'
14 md.loc[(md.eventname == r'\mod_page\event\course_module_viewed'),
15        ['behaviour']] = 'learning'
16 md.loc[(md.eventname == r'\mod_wiki\event\course_module_viewed'),
17        ['behaviour']] = 'learning'
18 md.loc[(md.eventname == r'\mod_wiki\event\page_viewed'),
19        ['behaviour']] = 'learning'
20 md.loc[(md.eventname == r'\mod_choice\event\course_module_viewed'),
21        ['behaviour']] = 'learning'
22 md.loc[(md.eventname == r'\mod_folder\event\course_module_viewed'),
23        ['behaviour']] = 'learning'
24 md.loc[(md.eventname == r'\mod_glossary\event\course_module_viewed'),
25        ['behaviour']] = 'learning'
26 md.loc[(md.eventname == r'\mod_workshop\event\course_module_viewed'),
27        ['behaviour']] = 'learning'
28 md.loc[(md.eventname == r'\mod_bigbluebuttonbn\event\recording_viewed'),
29        ['behaviour']] = 'learning'
30 md.loc[(md.eventname == r'\mod_forum\event\course_module_viewed'),
31        ['behaviour']] = 'communication'
32 md.loc[(md.eventname == r'\mod_forum\event\discussion_viewed'),
33        ['behaviour']] = 'communication'
34 md.loc[(md.eventname == r'\core\event\message_viewed'),
35        ['behaviour']] = 'communication'
36 md.loc[(md.eventname == r'\mod_chat\event\course_module_viewed'),
37        ['behaviour']] = 'communication'
38 md.loc[(md.eventname == r'\mod_chat\event\sessions_viewed'),
39        ['behaviour']] = 'communication'
```

---

Listing 26: Ergänzung des Merkmals *behaviour*

Durchgeführt wurde die Analyse hier also mit Bezug auf das gesamte studentische Lern- und Kommunikationsverhalten in Form einer Gesamtbetrachtung.

---

```

1 # Definition der Arbeitsdaten
2 md = md[(md['behaviour'] == 'learning') |
3          (md['behaviour'] == 'communication')]
```

---

Listing 27: Definition der Arbeitsdaten

Zum Abschluss der Datenaufbereitung wurden entsprechend den Log-Einträgen die Wochentage ermittelt und zur weiteren Verwendung gespeichert.

---

```
1 # Ermittlung der Wochentage der protokollierten Log-Einträge
2 days_per_week = md.timecreated.dt.dayofweek.sort_values()
```

---

Listing 28: Wochentag pro Log-Eintrag

#### Datenanalyse: Verteilung der Log-Einträge pro Wochentag

---

```
1 # Visualisierung der Menge der Log-Einträge pro Wochentag
2 chart = sns.histplot(days_per_week, bins=7, discrete=True,
3                      color=colors_set3[4], alpha=1)
```

---

Listing 29: Verteilung der Log-Einträge pro Wochentag

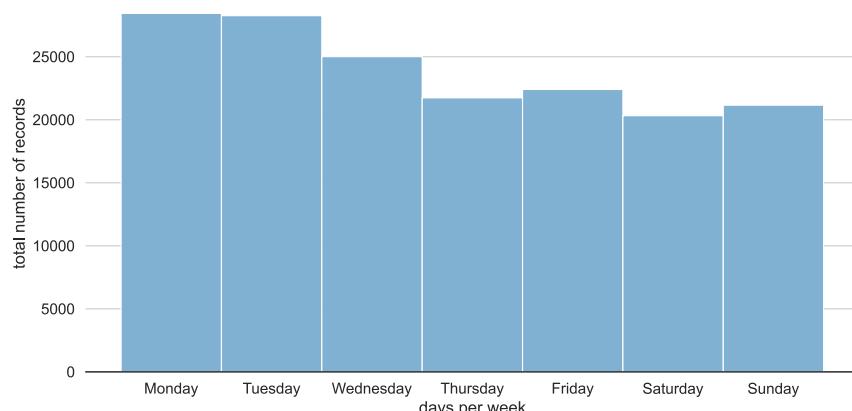


Abbildung 29.: Verteilung der Log-Einträge pro Wochentag

Wie die obige Visualisierung verdeutlicht, waren die Studenten an den Wochenenden weniger aktiv, als an den normalen Arbeitstagen unter der Woche. Da dieser Befund so nicht erwartet worden war, sollte eine weitere Analyse dieses Ergebnis noch einmal bestätigen.

---

```
1 # Visualisierung der Verteilung der Log-Einträge über die Wochentage
2 chart = sns.boxplot(x=days_per_week, orient='h', color=colors_set3[4])
```

---

Listing 30: Verteilung der Log-Einträge über die Wochentage

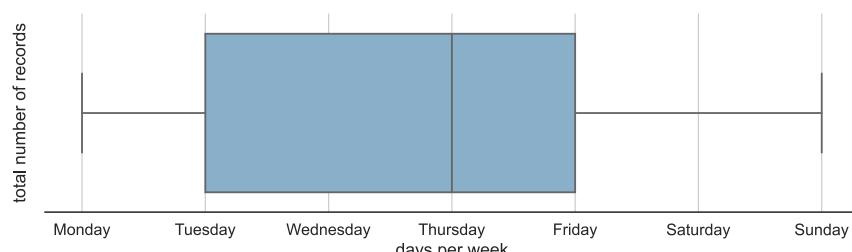


Abbildung 30.: Verteilung der Log-Einträge über die Wochentage

## Evaluierung

Der Boxplot macht noch etwas besser sichtbar, dass schon im Laufe des Dienstags ein Viertel und am Donnerstag bereits die Hälfte der wöchentlichen Aktivitäten stattgefunden hatte.

Damit bekräftigte und konkretisierte diese zweite Analyse die zuvor getroffene Aussage, und mithin stand fest, dass die Annahme, Studenten in digitalen Studienformaten seien überwiegend an Wochenenden aktiv, nicht zutreffend war.

### 3.3.2. Kategorisierung nach Tagestypen

Die im vorigen Abschnitt gewonnene Erkenntnis machte zwar deutlich, dass die Studenten an den gewöhnlichen Werktagen aktiver waren als an den Wochenenden.

Fraglich war jedoch, ob diese Beobachtung so auch auf die einzelnen Studenten übertragbar war oder ob es nicht individuelle Unterschiede gab, die in der Gesamt-betrachtung nur nicht zu erkennen waren.

Um diese Frage beantworten zu können und mögliche Unterschiede sichtbar zu machen, musste hier eine Betrachtung auf Basis des individuellen studentischen Verhaltens erfolgen.

Hierzu sollte auf Basis des gegebenen Datenbestands zunächst ein neues Datenset *loggings\_daytype* erstellt werden, das für jeden Studenten die anteiligen Mengen an Log-Einträgen für den Tagestyp *workingday* und *weekend* aufnehmen konnte.

## Datenaufbereitung

Ausgewählt wurden die Daten, die das Lern- und Kommunikationsverhalten für alle Studiengänge umfassen (s. Listing zur [Definition der Arbeitsdaten](#)).

---

```

1 # Erstellung eines neuen Datensets, bestehend aus den anteiligen Mengen
2 # an Log-Einträgen pro Student und Tagestyp
3 loggings_daytype = pd.DataFrame()
4 loggings_daytype['loggings'] =
5     md.timecreated[md.userstatus == 'student'].groupby(md.userid).count()
6 loggings_daytype['total'] =
7     md.daytype[md.userstatus == 'student'].groupby(md.userid).count()
8
9 # Ermittlung der absoluten Mengen an Log-Einträgen
10 # pro Student und Tagestyp
11 loggings_daytype['workingday'] =
12     md.daytype[(md.userstatus == 'student') &
13                 (md.daytype == 'workingday')].groupby(md.userid).count()
14 loggings_daytype['weekend'] =
15     md.daytype[(md.userstatus == 'student') &
16                 (md.daytype == 'weekend')].groupby(md.userid).count()
17
18 loggings_daytype.fillna(value=0, inplace=True)
19
20 # Ermittlung der anteiligen Mengen an Log-Einträgen
21 # pro Student und Tagestyp
22 loggings_daytype['workingday'] =
23     [i / j * 100 for i, j in zip(loggings_daytype['workingday'],
24                                    loggings_daytype['total'])]
24 loggings_daytype['weekend'] =
25     [i / j * 100 for i, j in zip(loggings_daytype['weekend'],
26                                    loggings_daytype['total'])]
```

```

26
27 # Ermittlung der anteiligen Mengen an Log-Einträgen
28 # pro Student (hier jeweils 100%)
29 loggings_daytype['total'] =
30     [i / j * 100 for i,j in zip(loggings_daytype['total'],
31                                     loggings_daytype['total'])]

```

Listing 31: Erstellung des neuen Datensets *loggings\_daytype*

userid	loggings	total	workingday	weekend
1	1324	100.0	90.785498	9.214502
13	2759	100.0	77.129395	22.870605
18	1128	100.0	83.244681	16.755319
19	3004	100.0	73.901465	26.098535
20	3733	100.0	81.275114	18.724886
...	...	...	...	...
132	1555	100.0	83.665595	16.334405
134	3063	100.0	70.421156	29.578844
136	18	100.0	61.111111	38.888889
142	3	100.0	0.000000	100.000000
143	538	100.0	79.925651	20.074349

Abbildung 31.: Erstellung des neuen Datensets *loggings\_daytype* (s. Anhang)

Um einen ersten visuellen Eindruck von den Wertigkeiten und den Unterschieden der individuellen Mengen zu erhalten, sollten diese zunächst in einer Gesamtübersicht dargestellt werden.

#### Datenanalyse: Anteilige Mengen an Log-Einträgen pro Student und Tagestyp

```

1 # Visualisierung der Gesamtmengen an Log-Einträgen
2 # pro Student und Tagestyp (in 100%)
3 bar_total = sns.barplot(x=loggings_daytype.index, y='total',
4                         data=loggings_daytype, color=colors_set2[1], alpha=1)
5 # Visualisierung der anteiligen Mengen an Log-Einträgen
6 # pro Student und Tagestyp (in x%)
7 bar_workingday = sns.barplot(x=loggings_daytype.index, y="workingday",
8                               data=loggings_daytype, color=colors_set2[0], alpha=1)

```

Listing 32: Anteilige Mengen an Log-Einträgen pro Tagestyp

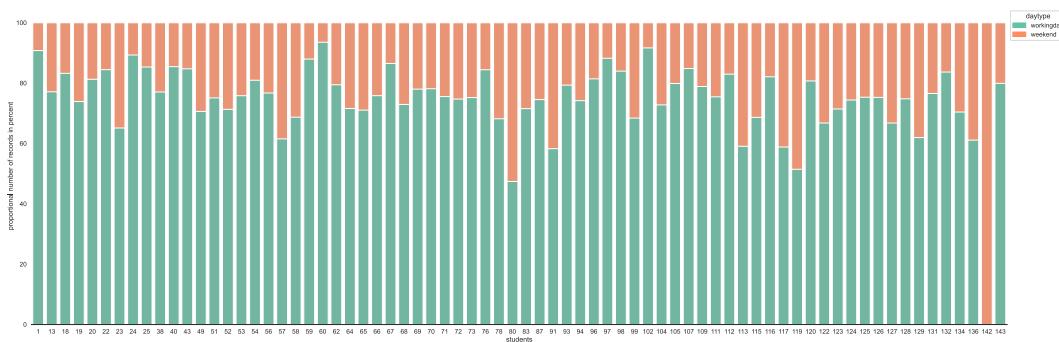


Abbildung 32.: Anteilige Mengen an Log-Einträgen pro Tagestyp (s. Anhang)

#### Evaluierung

Wie man der Darstellung entnehmen kann, gab es außer dem Studenten 142, der überhaupt nur drei Log-Einträge zu verzeichnen hatte (s. Abbildung darüber) und daher als Ausreißer nicht gewertet wurde, nur den Studenten 80, der überwiegend

am Wochenende gearbeitet hatte. Daneben gab es aber auch 29 Studenten, die mehr als 25% ihrer Leistungen an den Wochenenden erbrachten.

Für die abschließende Kategorisierung der Studenten nach Tagestyp wurde dann zwar auf einen mehrheitlichen Wert größer als 50% abgestellt, es wären aber auch andere Werte zur Einordnung möglich gewesen.

### Datenaufbereitung

Ausgewählt wurden die Daten, die das Lern- und Kommunikationsverhalten für alle Studiengänge umfassen (s. Listing zur [Definition der Arbeitsdaten](#)).

---

```

1 # Erstellung einer neuen Spalte zur Typisierung
2 loggings_daytype['location'] = 'workingday'
3 # Einordnung der Studenten nach der Lokalität ihrer Aktivitäten
4 loggings_daytype.loc[(loggings_daytype['workingday'] <= 50.0),
5                      ['location']] = 'weekend'
```

---

Listing 33: Typisierung der Studenten nach Tagestyp

### Datenanalyse: Typisierung der Studenten nach Tagestyp

---

```

1 # Visualisierung der Typisierung der Studenten nach Tagestyp
2 chart = sns.barplot(x=loggings_daytype.index,
3                      y=loggings_daytype.loggings,
4                      hue=loggings_daytype.location,
5                      hue_order=['workingday', 'weekend'],
6                      dodge=False, palette='Set2')
```

---

Listing 34: Typisierung der Studenten nach Tagestyp

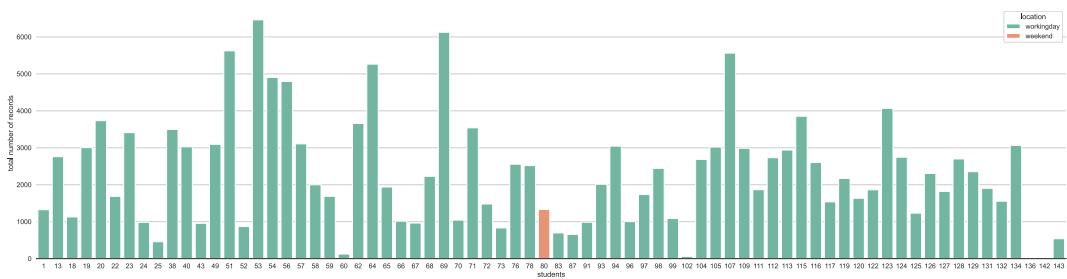


Abbildung 33.: Typisierung der Studenten nach Tagestyp ([s. Anhang](#))

### Evaluierung

In der obigen Visualisierung gut sichtbar ist der Student 80, der als einziger über das gesamte Semester hinweg größtenteils an den Wochenenden aktiv war.

Damit konnte die individuelle Analyse des studentischen Verhaltens schließlich auch das Ergebnis der eingangs durchgeföhrten Gesamtbetrachtung bestätigen.

### 3.3.3. Betrachtung des studentischen Verhaltens auf Tagesbasis

Nachdem die Betrachtungen auf Wochenbasis bereits die anfängliche Vermutung widerlegt hatten, dass Studenten überwiegend an Wochenenden aktiv seien, sollte

deren Verhalten des weiteren auch auf Tagessicht untersucht werden.

### Datenaufbereitung

Ausgewählt wurden die Daten, die das Lern- und Kommunikationsverhalten für alle Studiengänge beschreiben (s. Listing zur [Definition der Arbeitsdaten](#)), und es wurden gemäß den Log-Einträgen die Tagesstunden bestimmt.

---

```
1 # Ermittlung der Tagesstunden der protokollierten Log-Einträge
2 hours_per_day = md.timecreated.dt.hour.sort_values()
```

---

Listing 35: Tagesstunde pro Log-Eintrag

### Datenanalyse: Verteilung der Log-Einträge pro Tagesstunde

---

```
1 # Visualisierung der Menge der Log-Einträge pro Tagesstunde
2 chart = sns.histplot(hours_per_day, bins=24, discrete=True,
3                      color=colors_set3[6], alpha=1)
```

---

Listing 36: Verteilung der Log-Einträge pro Tagesstunde

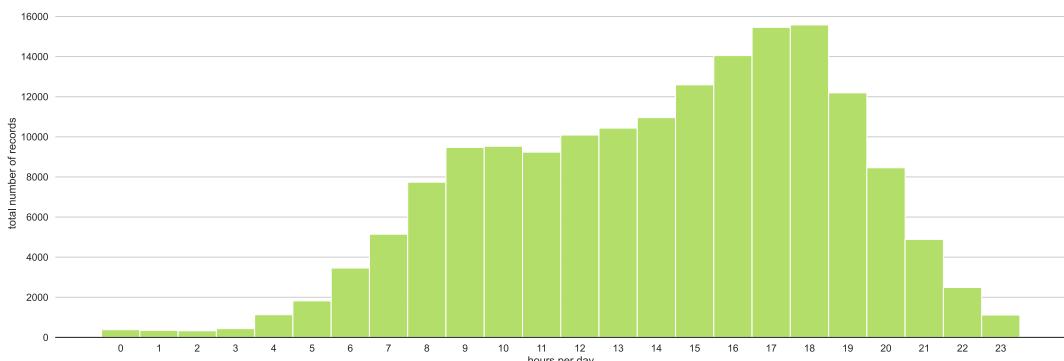


Abbildung 34.: Verteilung der Log-Einträge pro Tagesstunde ([s. Anhang](#))

Wie schon das Untersuchungsergebnis der Wochentage, zeigte die Verteilung der Log-Einträge pro Tagesstunde ein unerwartetes Bild: Die Studenten waren sowohl morgens als auch abends deutlich weniger aktiv als zu den sonst üblichen täglichen Arbeitszeiten.

---

```
1 # Visualisierung der Verteilung der Log-Einträge über die Tagesstunden
2 chart = sns.boxplot(x=hours_per_day, orient='h', color=colors_set3[6])
```

---

Listing 37: Verteilung der Log-Einträge über die Tagesstunden

### Evaluierung

Der obige Befund, dass Studenten überwiegend während der normalen Bürozeiten aktiv waren, wurde durch den Boxplot bestätigt: Nur ein Viertel aller Aktivitäten fand vor 11 Uhr bzw. nach 19 Uhr statt, während die Hälfte aller Log-Einträge bereits bis ca. 15 Uhr verzeichnet werden konnte.

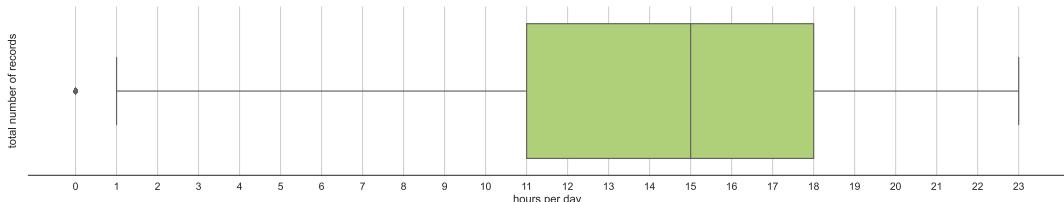


Abbildung 35.: Verteilung der Log-Einträge über die Tagesstunden (s. Anhang)

Mit Blick auf die beiden Analyseergebnisse konnte also festgestellt werden, dass die These, Studenten in Online-Studiengängen seien meist außerhalb der normalen Arbeitszeiten aktiv, nicht mit den wirklichen Gegebenheiten übereinstimmte.

### 3.3.4. Kategorisierung nach Tageszeiten

Wie bei der Analyse der Wochentage, so stellte sich nun auch hier die Frage, ob das Ergebnis aus der Gesamtbetrachtung der Tagesstunden für alle Studenten gleichermaßen galt, oder ob es nicht doch signifikante Unterschiede gab.

Eine Antwort darauf konnte hier ebenfalls nur eine Analyse des individuellen studentischen Verhaltens geben.

Entsprechend wurde auch hier zunächst ein neues Datenset *loggings\_daytime\_1* erstellt, in dem für die Studenten die anteiligen Mengen an Log-Einträgen für die Tageszeit *workingtime* und *freetime* gespeichert wurden.

#### Datenaufbereitung

Ausgewählt wurden die Daten, die das Lern- und Kommunikationsverhalten für alle Studiengänge umfassen (s. Listing zur [Definition der Arbeitsdaten](#)), und es wurde Bezug genommen auf die bereits vordefinierten Werte für die Tageszeit: Als *workingtime* wurden die Stunden von 9 bis 18 Uhr definiert, als *freetime* wurden die verbleibenden Stunden bestimmt.

```

1 # Erstellung eines neuen Datensets, bestehend aus den anteiligen Mengen
2 # an Log-Einträgen pro Student und Tageszeit
3 loggings_daytime_1 = pd.DataFrame()
4 loggings_daytime_1['loggings'] =
5     md.timecreated[md.userstatus == 'student'].groupby(md.userid).count()
6 loggings_daytime_1['total'] =
7     md.daytime_1[md.userstatus == 'student'].groupby(md.userid).count()
8
9 # Ermittlung der absoluten Mengen an Log-Einträgen
10 # pro Student und Tageszeit
11 loggings_daytime_1['workingtime'] =
12     md.daytime_1[(md.userstatus == 'student') &
13     (md.daytime_1 == 'workingtime')].groupby(md.userid).count()
14 loggings_daytime_1['freetime'] =
15     md.daytime_1[(md.userstatus == 'student') &
16     (md.daytime_1 == 'freetime')].groupby(md.userid).count()
17
18 loggings_daytime_1.fillna(value=0, inplace=True)
19
20 # Ermittlung der anteiligen Mengen an Log-Einträgen
21 # pro Student und Tageszeit (in x%)
22 loggings_daytime_1['workingtime'] =

```

```

23     [i / j * 100 for i, j in zip(loggings_daytime_1['workingtime'],
24         loggings_daytime_1['total'])]
25     loggings_daytime_1['freetime'] =
26     [i / j * 100 for i, j in zip(loggings_daytime_1['freetime'],
27         loggings_daytime_1['total'])]
28     # Ermittlung der anteiligen Mengen an Log-Einträgen
29     # pro Student (hier jeweils 100%)
30     loggings_daytime_1['total'] =
31     [i / j * 100 for i, j in zip(loggings_daytime_1['total'],
32         loggings_daytime_1['total'])]

```

Listing 38: Erstellung des neuen Datensets *loggings\_daytime\_1*

userid	loggings	total	workingtime	freetime
1	1324	100.0	61.706949	38.293051
13	2759	100.0	68.901776	31.098224
18	1128	100.0	72.340426	27.659574
19	3004	100.0	48.934754	51.065246
20	3733	100.0	81.837664	18.162336
...	...	...	...	...
132	1555	100.0	68.745981	31.254019
134	3063	100.0	71.237349	28.762651
136	18	100.0	61.111111	38.888889
142	3	100.0	33.333333	66.666667
143	538	100.0	51.672862	48.327138

Abbildung 36.: Erstellung des neuen Datensets *loggings\_daytime\_1* (s. Anhang)

Eine Übersicht über die individuellen anteiligen Mengen sollte auch hier ersten Aufschluss geben über mögliche Unterschiede des studentischen Verhaltens.

#### Datenanalyse: Anteilige Mengen an Log-Einträgen pro Student und Tageszeit

```

1  # Visualisierung der Gesamtmengen an Log-Einträgen
2  # pro Student und Tageszeit (in 100%)
3  bar_total = sns.barplot(x=loggings_daytime_1.index, y='total',
4                          data=loggings_daytime_1, color=colors_set2[1], alpha=1)
5  # Visualisierung der anteiligen Mengen an Log-Einträgen
6  # pro Student und Tageszeit (in %)
7  bar_workingtime = sns.barplot(x=loggings_daytime_1.index,
8                                y='workingtime', data=loggings_daytime_1,
9                                color=colors_set2[0], alpha=1)

```

Listing 39: Anteilige Mengen an Log-Einträgen pro Tageszeit

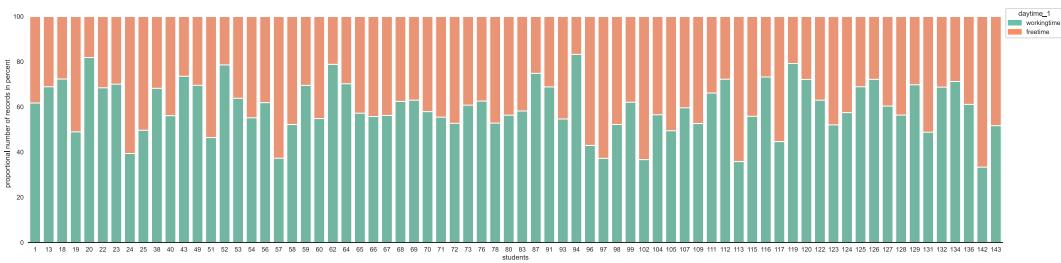


Abbildung 37.: Anteilige Mengen an Log-Einträgen pro Tageszeit (s. Anhang)

#### Evaluierung

Das obige Diagramm zeigte auf, dass es doch einige Studenten gab, die zu einem Großteil in den Stunden außerhalb der üblichen Geschäftszeiten tätig war.

Zu beachten war bei diesem Ergebnis jedoch, dass darunter erneut Student 142 enthalten war, der bereits als Ausreißer festgestellt wurde und es durchaus noch weitere Studenten mit nicht nennenswerten absoluten Mengen an Log-Einträgen gab (s. Student 136 in der Abbildung zum Datenset loggings\_daytime\_1).

Da diese Studenten aber ohnehin bei der abschließenden Typisierung ersichtlich werden sollten, konnte hier auf eine weitere Analyse anhand der absoluten Mengen verzichtet werden.

Die folgende Typisierung ordnete die Studenten nach den genannten Tageszeiten schließlich in zwei Gruppen und stellte dabei auf einen mehrheitlichen Wert größer als 50% ab. Wie in den beigefügten Jupyter Notebook Dokumenten gezeigt, wären auch andere Einordnungen z. B. anhand weiterer Tageszeiten möglich gewesen.<sup>29</sup>

### Datenaufbereitung

Ausgewählt wurden die Daten, die das Lern- und Kommunikationsverhalten für alle Studiengänge umfassen (s. Listing zur [Definition der Arbeitsdaten](#)).

---

```

1 # Erstellung einer neuen Spalte zur Typisierung
2 loggings_daytime_1['location'] = 'workingtime'
3 # Einordnung der Studenten nach der Lokalität ihrer Aktivitäten
4 loggings_daytime_1.loc[(loggings_daytime_1['workingtime'] <= 50.0),
5                         ['location']] = 'freetime'
```

---

Listing 40: Typisierung der Studenten nach Tageszeit

### Datenanalyse: Typisierung der Studenten nach Tageszeit

---

```

1 # Visualisierung der Typisierung der Studenten nach Tageszeit
2 chart = sns.barplot(x=loggings_daytime_1.index,
3                      y=loggings_daytime_1.loggings,
4                      hue=loggings_daytime_1.location,
5                      hue_order=['workingtime', 'freetime'],
6                      dodge=False, palette='Set2')
```

---

Listing 41: Typisierung der Studenten nach Tageszeit

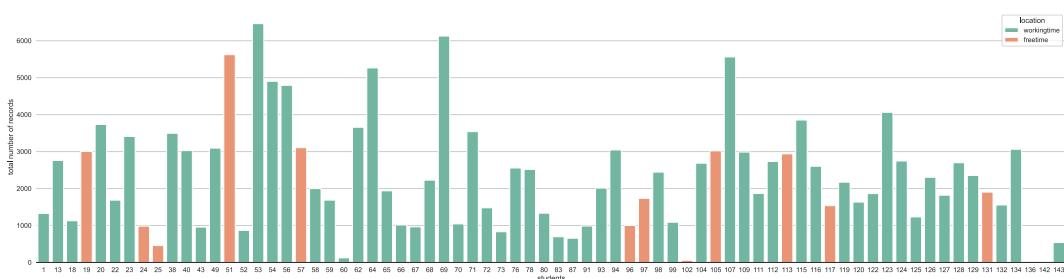


Abbildung 38.: Typisierung der Studenten nach Tageszeit ([s. Anhang](#))

### Evaluierung

Wie die Visualisierung zur Typisierung der Studenten zeigte, gab es etwas mehr als

<sup>29</sup> Siehe auch die Jupyter Notebook Dokumente zur Lokalitätsanalyse.

10 Studenten, die, wenn auch nur knapp, den größten Teil ihrer Aktivitäten außerhalb der normalen Arbeitszeiten ausführte. Alle anderen Studenten waren dagegen überwiegend in der Zeit von 9 bis 18 Uhr aktiv.

Damit konnte die individuelle Analyse des studentischen Verhaltens schließlich auch das Ergebnis der zuvor durchgeführten Gesamtbetrachtung bestätigen und die eingangs formulierte Annahme widerlegen, dass Studenten primär außerhalb der üblichen Geschäftszeiten aktiv seien.

### 3.3.5. Vergleich des Lern- und Kommunikationsverhaltens

Wie in der Einleitung zu dieser Arbeit schon erwähnt, sollte neben der allgemeinen Betrachtung des studentischen Verhaltens ebenfalls eine spezifische Untersuchung des Lernverhaltens sowie des Kommunikationsverhaltens erfolgen, um hierdurch eventuelle Abweichungen zur Gesamtbetrachtung zu erkennen bzw. im direkten Vergleich von Lern- und Kommunikationsverhalten mögliche signifikante Unterschiede sichtbar zu machen.

Entsprechend dem Aufbau der Jupyter Notebook Dokumente bedurfte es hierzu lediglich kleiner Anpassungen bei der Definition der Arbeitsdaten.

#### *Datenaufbereitung: Lern- und Kommunikationsverhalten*

Ausgewählt wurden jeweils nur die Daten, die das Lern- bzw. Kommunikationsverhalten für alle Studiengänge beschreiben (s. nachfolgende Listings).

---

```
1 # Definition der Arbeitsdaten
2 md = md[ (md['behaviour'] == 'learning') ]
```

---

Listing 42: Definition der Arbeitsdaten, Lernverhalten

---

```
1 # Definition der Arbeitsdaten
2 md = md[ (md['behaviour'] == 'communication') ]
```

---

Listing 43: Definition der Arbeitsdaten, Kommunikationsverhalten

Alle weiteren Schritte bis hin zur Typisierung erfolgten anschließend vollkommen analog dem zuvor bei der Gesamtbetrachtung beschriebenen Vorgehen, so dass sich im Ergebnis schließlich folgende Darstellungen zeigten:

#### *Evaluierung*

Auf den ersten Blick zeigten die beiden obigen Diagramme zur Kategorisierung nach Tagestyp ein ähnliches Bild wie das der vorherigen Gesamtbetrachtung von Lern- und Kommunikationsverhalten: Insgesamt nur sehr wenige der Studenten waren überwiegend an den Wochenenden aktiv.

### 3. Analysen

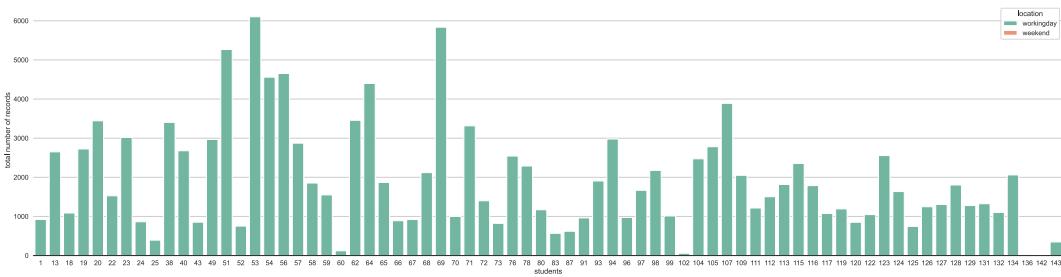


Abbildung 39.: Typisierung nach Tagestyp, Lernverhalten (s. Anhang)

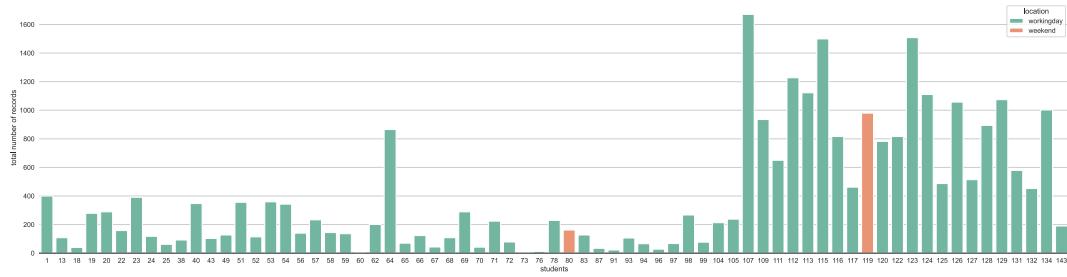


Abbildung 40.: Typisierung nach Tagestyp, Kommunikationsverhalten (s. Anhang)

Was dagegen bei der unteren Abbildung zu den Kommunikationsaktivitäten auffiel, waren die vergleichsweise hohen Mengen an Log-Einträgen bei den Studenten mit userids im dreistelligen Bereich. Dies deutete auf ein gänzlich anderes Nutzerverhalten hin.

Hier lag die Vermutung zwar nahe, dass es sich dabei um Studenten handelte, die einem bestimmten Studiengang angehörten, eine weitergehende Analyse des Lern- und Kommunikationsverhaltens auf Studiengangsbasis erschien aber in Anbetracht des Aufwands und des zu erwartenden Erkenntnisgewinns nicht lohnenswert.

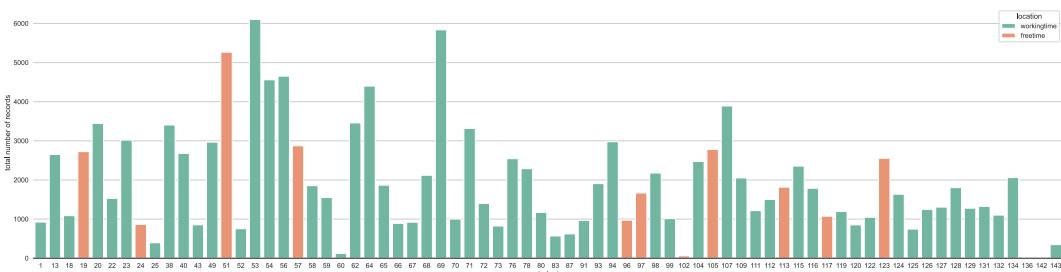


Abbildung 41.: Typisierung nach Tageszeit, Lernverhalten (s. Anhang)

#### Evaluierung

Auch bei der Typisierung nach Tageszeit wiesen die Visualisierungen keine offensichtlichen Abweichungen zum Ergebnis der Gesamtbetrachtung auf.

Im wesentlichen waren es bei der Betrachtung des Lern- und Kommunikationsverhaltens wie auch bei der Gesamtbetrachtung die selben Studenten, die zumeist außerhalb der normalen Arbeitszeiten aktiv waren.

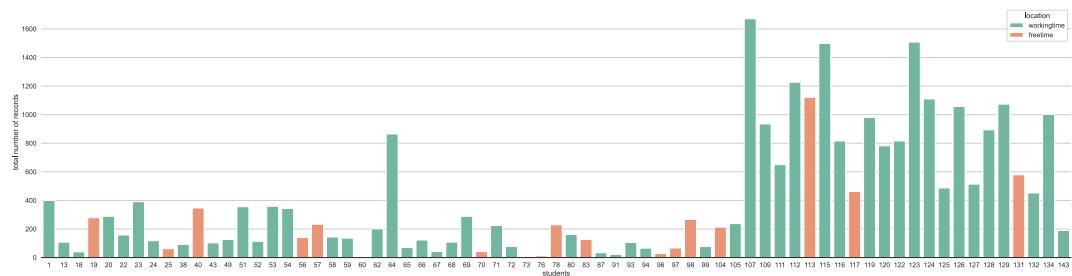


Abbildung 42.: Typisierung nach Tageszeit, Kommunikationsverhalten ([s. Anhang](#))

### 3.3.6. Zusammenfassung

In diesem Kapitel wurde dargelegt, wie mittels der zuvor konkretisierten Datenbasis das zeitliche Auftreten studentischer Aktivitäten untersucht wurde und die Studenten selbst hiernach kategorisiert werden konnten.

Ausgehend von der Tatsache, dass Studenten in digitalen Studienformaten mehrheitlich berufstätig sind oder auch familiären Verpflichtungen nachkommen, wurde die Annahme abgeleitet, dass diese infolgedessen überwiegend außerhalb normaler Arbeitszeiten aktiv seien.

Im Folgenden konnte in verschiedenen Untersuchungen gezeigt werden, dass diese Annahme nicht mit den realen Gegebenheiten übereinstimmte. Anhand der Verteilungen der Log-Einträge konnte bewiesen werden, dass die Studenten sowohl auf Wochensicht als auch auf Tagessicht überwiegend zu üblichen Geschäftszeiten aktiv waren.

Um mögliche, in der Gesamtbetrachtung nicht sichtbare, Unterschiede im studentischen Verhalten feststellen zu können, wurden im Anschluss Untersuchungen auf Basis invidividueller anteiliger Mengen an Log-Einträgen ergänzt. Diese erbrachten u. a. die folgenden Resultate:

- Nur ein einziger Student war über das gesamte Semester hinweg zumeist an den Wochenenden zu Studienzwecken aktiv.
- Fast 30 Studenten haben im Untersuchungszeitraum einen Anteil von mehr als 25% ihrer Leistungen am Wochenende erbracht.
- Etwas mehr als 10 Studenten waren überwiegend außerhalb gewöhnlicher täglicher Geschäftszeiten (9 bis 18 Uhr) für ihr Studium aktiv.

Nach Kategorisierung der Studenten entsprechend ihres individuellen Verhaltens wurde abschließend das Lernverhalten und das Kommunikationsverhalten getrennt betrachtet, um auch hier mögliche Unterschiede erkennen zu können.

Bezüglich der Typisierung konnten hier keine nennenswerten Abweichungen zur Gesamtbetrachtung oder Unterschiede zwischen Lern- und Kommunikationsverhalten festgestellt werden. Bemerkenswert waren hier nur die im Umfang deutlich erhöhten Kommunikationsaktivitäten mancher Studenten.

### 3.4. Kontinuität des Lern- und Kommunikationsverhaltens

Auf Grundlage der im Kapitel zur [Konkretisierung der Datenbasis](#) festgestellten viewed-Events sollten nun weitere Untersuchungen sich anschließen. Insbesondere ging es dabei um die Beantwortung der folgenden Frage:

Auf welche Weise lassen sich Studenten nach der Kontinuität ihres Handelns in verschiedene Gruppen unterscheiden?

Wesentlicher Aspekt der Analysen sollte also die Regelmäßigkeit der studentischen Aktivitäten sein, die schließlich einen Hinweis darauf liefern konnte wie beständig die Studenten ihren Verpflichtungen nachkamen.

#### 3.4.1. Betrachtung des studentischen Verhaltens im Gesamtzeitraum

Betrachtungen auf verschiedenen zeitlichen Ebenen, insbesondere auf Wochen- und Tagessicht wie in der folgenden Analyse, haben gezeigt, dass sich das studentische Verhalten über das Semester hinweg keineswegs konstant präsentierte.

##### Datenaufbereitung

Ausgewählt wurden die Daten, die das Lern- und Kommunikationsverhalten für alle Studiengänge umfassen (s. Listing zur [Definition der Arbeitsdaten](#)).

##### Datenanalyse: Verteilung der Log-Einträge im Gesamtzeitraum pro Tag

```
1 # Visualisierung der Menge der Log-Einträge über den Gesamtzeitraum
2 chart = sns.histplot(data=md.timecreated, bins=235,
3                      color=colors_set3[4], alpha=1)
```

Listing 44: Verteilung der Log-Einträge im Gesamtzeitraum pro Tag

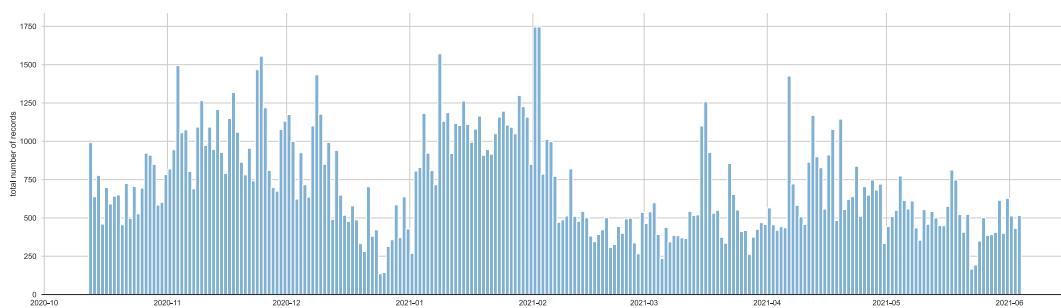


Abbildung 43.: Verteilung der Log-Einträge pro Tag ([s. Anhang](#))

##### Evaluierung

Auf den ersten Blick ließ das Histogramm in der Gesamtübersicht keine Kontinuität i. S. eines konstanten Verlaufs erkennen. Aufallend waren primär die zu erwartenden starken Rückgänge an den Feiertagen oder zwischen den Prüfungszeiträumen sowie die höheren Mengen an Log-Einträgen vor und während der Prüfungen. Des

weiteren deutete das stetige Auf und Ab der Aktivitäten eher auf ein sprunghaftes Verhalten hin.

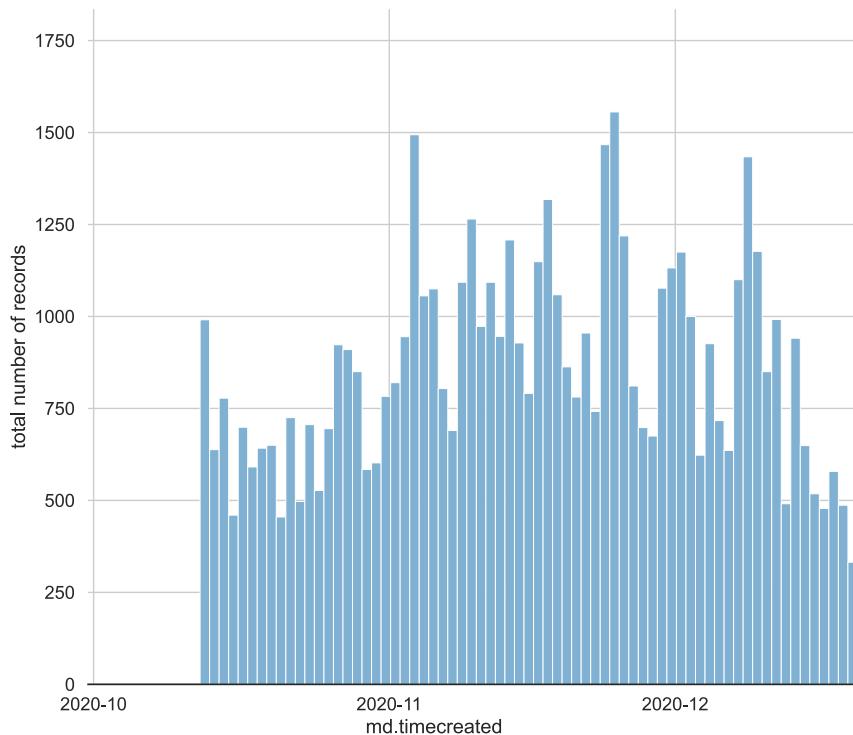


Abbildung 44.: Verteilung der Log-Einträge pro Tag (Ausschnitt)

Bei genauerer Betrachtung des Zeitraums bis Mitte Dezember wurde aber ersichtlich, dass die Änderungen in der Verteilung doch nicht willkürlich waren. Vielmehr folgten sie einem etwas unregelmäßigen Muster, wonach sich steile Anstiege und mehrere Tage hoher Aktivität mit einem starken Rückgang stetig abwechselten. Der Gedanke lag nahe, dass sich darin ein periodischer Ablauf von Aktivitäten zeigte, der auf Wochensicht durchaus eine gewisse Kontinuität besaß.

Dieser Beobachtung folgend erschien es lohnenswert, hier weitere Analysen auf wöchentlicher Basis anzuschließen. Dabei war zu vermuten, dass die Studenten, die bei der [Typisierung nach Lokalität](#) mit nur geringen Mengen an Log-Einträgen bereits auffällig wurden, wahrscheinlich auch nur an wenigen Wochen überhaupt aktiv waren. Aufschluss hierüber sollte die folgende Analyse geben.

### Datenaufbereitung

Ausgewählt wurden die Daten, die das Lern- und Kommunikationsverhalten für alle Studiengänge umfassen (s. Listing zur [Definition der Arbeitsdaten](#)), und es wurden bezüglich der Arbeitswochen die jeweiligen Zahlen pro Student aus dem Datenbestand ermittelt.

---

1 # Ermittlung der Menge der Arbeitswochen pro Student  
2 weeks\_user = pd.Series(md.year\_week[

```
3     md.userstatus == 'student'].groupby(md.userid).nunique(),
4     name='weeks')
```

Listing 45: Menge der Arbeitswochen pro Student

Die Grafik unten zeigt die Menge der Arbeitswochen im Gesamtzeitraum für jeden einzelnen Studenten.

#### *Datenanalyse: Menge der Arbeitswochen pro Student*

```
1 # Visualisierung der absoluten Menge der Arbeitswochen pro Student
2 chart = sns.barplot(x=weeks_user.index.astype(int),
3                      y=weeks_user, color=colors_general[3])
```

Listing 46: Menge der Arbeitswochen pro Student

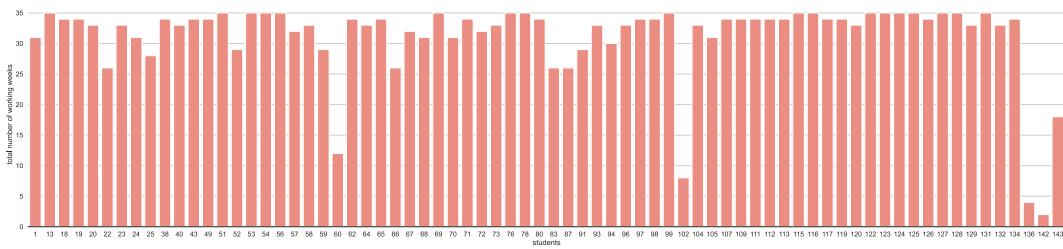


Abbildung 45.: Menge der Arbeitswochen pro Student ([s. Anhang](#))

#### *Evaluierung*

Die obige Visualisierung bestätigte die vorherige Vermutung und zeigte bei den Studenten mit allgemein wenigen Log-Einträgen auch nur wenige aktive Wochen. Insofern war in diesen Fällen von einem sporadischen Lernverhalten auszugehen.

Fraglich waren an dieser Stelle jedoch die durchgängig hohen Wochenzahlen der anderen Studenten. Ließen diese unmittelbar auf eine ebenfalls hohe Kontinuität des Studierens schließen?

Unter Berücksichtigung dessen, dass die Untersuchung nur darauf abstellte, die Anzahl der Wochen zu ermitteln an denen ein Student überhaupt aktiv war, nicht aber zum Ziel hatte auch den Umfang der studentischen Aktivitäten innerhalb der Wochen aufzuzeigen, war diese Frage klar zu verneinen.

Folglich war es geboten, neben den Wochen auch die nächstkleinere Zeiteinheit also die Tage an denen ein Student aktiv war, genauer zu untersuchen.

#### *Datenaufbereitung*

Ausgewählt wurden die Daten, die das Lern- und Kommunikationsverhalten für alle Studiengänge beschreiben (s. Listing zur [Definition der Arbeitsdaten](#)), und es wurden pro Student die Menge der aktiven Arbeitstage bestimmt.

```
1 # Ermittlung der Menge der Arbeitstage pro Student
2 days_user = pd.Series(md.year_day[
3     md.userstatus == 'student'].groupby(md.userid).nunique(),
```

```
4     name='days')
```

Listing 47: Menge der Arbeitstage pro Student

Die nachfolgende Visualisierung präsentiert in einer Gesamtübersicht die Menge der Arbeitstage im gesamten Zeitraum pro Student.

#### *Datenanalyse: Menge der Arbeitstage pro Student*

```
1 # Visualisierung der absoluten Menge der Arbeitstage pro Student
2 chart = sns.barplot(x=days_user.index.astype(int),
3                      y=days_user, color=colors_general[6])
```

Listing 48: Menge der Arbeitstage pro Student

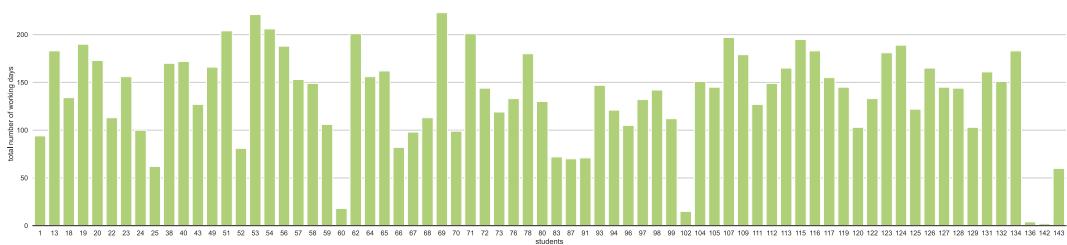


Abbildung 46.: Menge der Arbeitstage pro Student (s. Anhang)

#### *Evaluierung*

Betrachtete man das obige Diagramm zu den Arbeitstagen im direkten Vergleich mit dem vorherigen zu den Mengen der Arbeitswochen, so wurde also deutlich, dass einige Studenten auf Tagessicht viel weniger aktiv waren als auf Wochensicht.

Dies bedeutete, dass eine nur wochenbasierte Sicht für die Untersuchungen auf Kontinuität nicht ausreichte, sondern diese vielmehr auch die Tagesaktivitäten berücksichtigen mussten.

#### **3.4.2. Ermittlung der Vergleichsgöße**

Waren die zeitlichen Bezüge nun klar definiert, ging es um die Frage: Gab es eine für Studenten allgemein gültige Größe, an der man die Kontinuität des Verhaltens objektiv messen konnte?

Insbesondere nach Betrachtung von Einzelfällen war diese Frage zu verneinen, denn man musste stets auch jene Studenten im Blick haben, die z. B. aufgrund einer erhöhten beruflichen Belastung nur wenige Kursmodule belegt hatten, diese aber konsequent bearbeiteten. Hätte man nun auf Basis der Gesamtaktivitäten, die wie bereits gezeigt sehr unterschiedlich war, und des Gesamtzeitraums eine fixe Größe zur Unterscheidung definiert, wäre man diesen Fällen nicht gerecht geworden.

Daher war es für eine präzise und zweckgerichtete Analyse erforderlich, stets die individuelle Situation eines Studenten zu betrachten, nur daran konnte man die Kontinuität von Aktivitäten fair beurteilen.

Für das weitere Vorgehen war es zunächst notwendig, das originäre Datenset u. a. um Informationen zur Kalenderwoche und zum Kalendertag zu erweitern, um bei Bedarf unmittelbar darauf zugreifen zu können.

Von besonderer Bedeutung war aber vielmehr das neue Datenset *time\_rel\_con*, das nun im Laufe der Analysen wichtige individuelle Kennziffern für die Studenten aufnehmen sollte. Bei Erstellung wurden diesem bereits die vorab ermittelten Ergebnisse zu den individuellen Log-Einträgen, Arbeitswochen und Arbeitstagen hinzugefügt und in einem weiteren Schritt wurden die Wochen- und Tagesdurchschnitte ergänzt.<sup>30</sup>

Letztere sollten jedoch nicht selbst als Größe in die Untersuchungen eingehen. Vielmehr sollten die Durchschnitte als Anhaltspunkte dienen, von denen aus für die Studenten eigene untere bzw. obere Aktivitätslimits berechnet werden konnten.

Mithilfe der durch diese Limits begrenzten Toleranzbereiche, die sich in der Ausdehnung auch an der individuellen Gesamtaktivität ausrichteten, sollte es möglich sein, auch natürliche moderate Aktivitätsschwankungen angemessener zu berücksichtigen, als dieses durch die Verwendung eines einfachen Schwellwerts möglich gewesen wäre.

### Datenaufbereitung

Ausgewählt wurden die Daten, die das Lern- und Kommunikationsverhalten für alle Studiengänge umfassen (s. Listing zur [Definition der Arbeitsdaten](#)).

---

```

1 # Erstellung des neuen Datensets
2 time_rel_con =
3     pd.concat([loggings_user, weeks_user, days_user], axis=1)
4
5 # Erstellung neuer Spalten für die individuellen Kennziffern pro Woche
6 time_rel_con['avg_count_per_week'] = 0
7 time_rel_con.loc[(time_rel_con['avg_count_per_week'] == 0),
8     ['avg_count_per_week']] =
9         (loggings_user / weeks_user).astype(int)
10 time_rel_con['lower_count_per_week'] = 0
11 time_rel_con.loc[(time_rel_con['lower_count_per_week'] == 0),
12     ['lower_count_per_week']] =
13         ((loggings_user / weeks_user) * 0.5).astype(int)
14 time_rel_con['upper_count_per_week'] = 0
15 time_rel_con.loc[(time_rel_con['upper_count_per_week'] == 0),
16     ['upper_count_per_week']] =
17         ((loggings_user / weeks_user) * 1.5).astype(int)
18
19 # Erstellung neuer Spalten für die individuellen Kennziffern pro Tag
20 time_rel_con['avg_count_per_day'] = 0
21 time_rel_con.loc[(time_rel_con['avg_count_per_day'] == 0),
22     ['avg_count_per_day']] =
23         (loggings_user / days_user).astype(int)
24 time_rel_con['lower_count_per_day'] = 0
25 time_rel_con.loc[(time_rel_con['lower_count_per_day'] == 0),
26     ['lower_count_per_day']] =
27         ((loggings_user / days_user) * 0.5).astype(int)
28 time_rel_con['upper_count_per_day'] = 0
29 time_rel_con.loc[(time_rel_con['upper_count_per_day'] == 0),
30     ['upper_count_per_day']] =
31         ((loggings_user / days_user) * 1.5).astype(int)

```

---

<sup>30</sup> Siehe auch die beigefügten Jupyter Notebook Dokumente zur Kontinuitätsanalyse.

---

Listing 49: Erstellung des neuen Datensets zur Kontinuitätsanalyse

Unten dargestellt ist das neue Datenset, bei dem aus Platzgründen die Spalten für die Log-Einträge, die Arbeitswochen und die Arbeitstage pro Student ebenso wie verschiedene Zeilen nur durch Punkte (...) angezeigt werden können.

	userid	...	avg_count_per_week	lower_count_per_week	upper_count_per_week	avg_count_per_day	lower_count_per_day	upper_count_per_day
0	1	...	42	21	64	14	7	21
1	13	...	78	39	118	15	7	22
2	18	...	33	16	49	8	4	12
3	19	...	88	44	132	15	7	23
4	20	...	113	56	169	21	10	32
5	...	...	...	...	...	...	...	...
67	132	...	47	23	70	10	5	15
68	134	...	90	45	135	16	8	25
69	136	...	4	2	6	4	2	6
70	142	...	1	0	2	1	0	2
71	143	...	29	14	44	8	4	13

Abbildung 47.: Datenset *time\_rel\_con* zur Kontinuitätsanalyse (s. Anhang)

Mit diesem neuen Datenset waren die Voraussetzungen geschaffen, um auch die Mengen der Wochen und Tage zu ermitteln, an denen ein Student innerhalb seines persönlichen Toleranzbereichs agiert hat. Hierzu wurden nun ebenfalls die Werte zu den Kalenderwochen und Kalendertagen benötigt, die zuvor dem ursprünglichen Datenbestand hinzugefügt wurden.

### Datenaufbereitung

Die Datenaufbereitung erfolgte anschließend in zwei Phasen: Zunächst waren die persönlichen Toleranzwerte mit den Kalenderwochen und -tagen abzugleichen, um die entsprechenden Mengen zu ermitteln. Danach sollten diese in Listen gespeicherten Mengen als neue Kennziffern zu dem neuen Datenset ergänzt werden.

---

```

1 list_weeks = list()
2
3 # Funktion zur Ermittlung der Menge der Arbeitswochen
4 def get_weeks_in_range(i, list_weeks):
5     count = 0
6     for row in md.year_week[
7         (md.userid == time_rel_con.iloc[i]['userid']) &
8         ((md['behaviour'] == 'learning') |
9          (md['behaviour'] == 'communication'))]
10        ].groupby(md.year_week).count():
11         if row > time_rel_con.iloc[i]['lower_count_per_week'] &
12             row < time_rel_con.iloc[i]['upper_count_per_week']:
13                 count += 1
14             list_weeks.append(count)
15
16 # Schleife zur Steuerung der Ermittlungsfunktion
17 for i in time_rel_con.index:
18     get_weeks_in_range(i, list_weeks)
19
20 weeks_in_range =
21     pd.Series(list_weeks, name='weeks_in_range', dtype='int32')
22
23 # Ergänzung des Datensets mit den Arbeitswochen im Toleranzbereich
24 time_rel_con = pd.concat([time_rel_con, weeks_in_range], axis=1)

```

---

Listing 50: Ermittlung der Arbeitswochen im Toleranzbereich

Die Ermittlung der Mengen der Arbeitstage, an denen die Menge der Log-Einträge pro Arbeitstag innerhalb des Toleranzbereichs lag, verlief analog:

---

```

1 list_days = list()
2
3 # Funktion zur Ermittlung der Menge der Arbeitstage
4 def get_days_in_range(i, list_days):
5     count = 0
6     for row in md.year_day[
7         (md.userid == time_rel_con.iloc[i]['userid']) &
8         ((md['behaviour'] == 'learning') |
9          (md['behaviour'] == 'communication'))]
10    ].groupby(md.year_day).count():
11        if row > time_rel_con.iloc[i]['lower_count_per_day'] &
12            row < time_rel_con.iloc[i]['upper_count_per_day']:
13                count += 1
14    list_days.append(count)
15
16 # Schleife zur Steuerung der Ermittlungsfunktion
17 for i in time_rel_con.index:
18     get_days_in_range(i, list_days)
19
20 days_in_range =
21     pd.Series(list_days, name='days_in_range', dtype='int32')
22
23 # Ergänzung des Datensets mit den Arbeitstagen im Toleranzbereich
24 time_rel_con = pd.concat([time_rel_con, days_in_range], axis=1)

```

---

Listing 51: Ermittlung der Arbeitstage im Toleranzbereich

An dieser Stelle waren nun alle erforderlichen Werte ermittelt, um in einem weiteren Schritt die Größe zu bestimmen, nach der die Kontinuität studentischer Aktivitäten bestimmt werden konnte. Folgende Bedingungen sollte die gesuchte Größe erfüllen:

1. Gemäß den bisherigen Ergebnissen sollten die ermittelten Mengen an Wochen und Tagen im individuellen Toleranzbereich berücksichtigt werden.
2. Als feste Bezugsgröße war der Zeitraum miteinzubeziehen, innerhalb dessen die Kontinuität festgestellt werden sollte.
3. War ein Student nie aktiv, musste der mathematische Ausdruck im Ergebnis den Wert 0 ergeben.
4. War ein Student an jedem Tag und somit dann auch in jeder Woche aktiv, sollte der mathematische Ausdruck den maximalen Wert liefern.

Entsprechend dieser Vorgaben konnte schließlich für das Verhältnis der genannten Mengen die nachfolgende Gesetzmäßigkeit definiert werden:

$$\begin{array}{ccc}
 \frac{\text{Menge der Arbeitswochen im Toleranzbereich} \longrightarrow 0}{\text{Gesamtmenge der Wochen im Untersuchungszeitraum}} & \times & \frac{\text{Menge der Arbeitstage im Toleranzbereich} \longrightarrow 0}{\text{Gesamtmenge der Tage im Untersuchungszeitraum}} \longrightarrow 0 \\
 \longrightarrow 0 & & \longrightarrow 0
 \end{array}$$
  

$$\begin{array}{ccc}
 \frac{\text{Menge der Arbeitswochen im Toleranzbereich} \longrightarrow \max}{\text{Gesamtmenge der Wochen im Untersuchungszeitraum}} & \times & \frac{\text{Menge der Arbeitstage im Toleranzbereich} \longrightarrow \max}{\text{Gesamtmenge der Tage im Untersuchungszeitraum}} \longrightarrow 1 \\
 \longrightarrow 1 & & \longrightarrow 1
 \end{array}$$

Abbildung 48.: Individueller Kontinuitätskoeffizient, IKK ([s. Anhang](#))

Dargestellt in der obigen Abbildung sind nur die maximalen Ausprägungen des IKK (s. o. Überlegung 3 und 4). In allen anderen Situationen, repräsentiert durch beliebige Werte zwischen 0 und den maximalen Werten für die Arbeitswochen bzw. Arbeitstage im Toleranzbereich eines Studenten könnte das Ergebnis des Gesamtausdrucks dann immer nur in Richtung des Werts 0 oder in Richtung des Werts 1 tendieren, dieses Intervall aber nicht verlassen.

Mit dem individuellen Kontinuitätskoeffizienten (IKK) war also die gesuchte Größe gefunden, mit der sich die Kontinuität studentischen Verhaltens messen ließ.

### 3.4.3. Kategorisierung nach IKK

Der IKK wurde nun noch als Kennziffer berechnet und ebenfalls in das Datenset *time\_rel\_con* aufgenommen, um für die abschließende Typisierung der Studenten zur Verfügung zu stehen.

#### Datenaufbereitung

---

```

1 # Definition der Menge an Wochen im Untersuchungszeitraum
2 weeks_in_period_count =
3     md.timecreated.sort_values().dt.strftime('%Y-%U').unique().size
4 # Definition der Menge an Tagen im Untersuchungszeitraum
5 days_in_period_count =
6     md.timecreated.sort_values().dt.dayofyear.unique().size
7
8 # Ergänzung des individuellen Kontinuitätskoeffizienten (IKK)
9 time_rel_con['ikk'] = 0
10 time_rel_con.loc[(time_rel_con['ikk'] == 0), ['ikk']] =
11     ((time_rel_con.weeks_in_range / weeks_in_period_count) *
12      (time_rel_con.days_in_range / days_in_period_count))

```

---

Listing 52: Ergänzung des IKK im Datenset *time\_rel\_con*

Danach konnte in einem letzten vorbereitenden Schritt das Datenset *time\_rel\_con* durch das Merkmal *continuity* und Kategorien auf Basis der IKK-Schwellwerte 0.3 und 0.6 komplettiert werden. Letztere wurden im konkreten Fall zwar so gewählt, dass es einen mittleren Bereich gab, der je nach Schwellwert schmäler oder breiter bestimmt werden konnte, es wären aber auch beliebige andere Einteilungen des IKK-Intervalls möglich gewesen.

---

```

1 # Erstellung einer neuen Spalte zur Typisierung
2 time_rel_con['continuity'] = 'orange'
3
4 # Einordnung der Studenten nach der Kontinuität ihrer Aktivitäten
5 time_rel_con.loc[(time_rel_con['ikk'] > 0.6), ['continuity']] = 'green'
6 time_rel_con.loc[(time_rel_con['ikk'] < 0.3), ['continuity']] = 'blue'

```

---

Listing 53: Typisierung der Studenten nach der Kontinuität der Aktivitäten

Die folgende Analyse sollte schließlich Aufschluss geben über die einleitende Frage, wie sich die Gesamtmenge der identifizierten Studenten aufgrund einer gewissen Kontinuität der Lern- und Kommunikationsaktivitäten in Gruppen einteilen lassen.

### Datenanalyse: Typisierung der Studenten nach IKK

```

1 # Visualisierung der Typisierung der Studenten nach IKK
2 chart = sns.barplot(x=time_rel_con.userid.astype(int),
3                      y=time_rel_con.ikk, hue=time_rel_con.continuity,
4                      hue_order=['green', 'orange', 'blue'],
5                      dodge=False, palette='Set2')

```

Listing 54: Typisierung der Studenten nach IKK

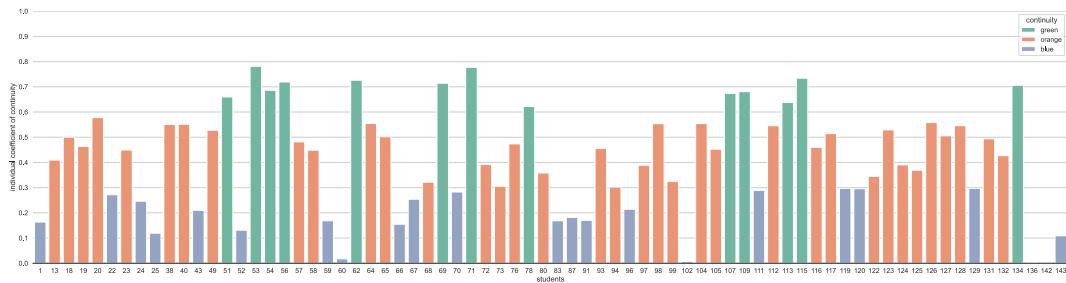


Abbildung 49.: Typisierung der Studenten nach IKK (s. Anhang)

### Evaluierung

Wie die obige Grafik zur Typisierung zeigte, ließen sich die Studenten nach den bei der Datenaufbereitung gewählten IKK-Schwellwerten in die drei Gruppen Green, Orange und Blue einteilen, wobei die mittlere Gruppe Orange genau die Hälfte der Studenten umfasste.

Dass die Einordnung nach dem IKK zweckmäßig funktionierte, war mithilfe des nachfolgenden Ausschnitts aus der [Gesamtübersicht](#) zu den Verteilungen der Log-Einträge gut zu erkennen. So waren bei den Studenten 1 und 59 über den Untersuchungszeitraum hinweg relativ große Lücken zu beobachten, was entsprechend in einer niedrigen Kontinuität resultierte. Die Studenten 18 und 80, die eine eher mittlere Kontinuität besaßen, präsentierten bei den Log-Einträgen hingegen ein geschlosseneres Bild.

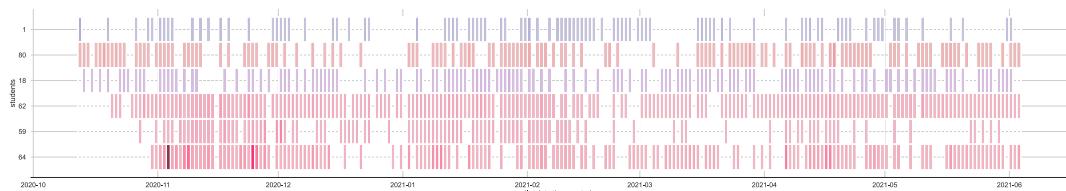


Abbildung 50.: Menge der Log-Einträge für einzelne Studenten (s. Anhang)

Auch die Beobachtungen im direkten Vergleich von Student 62 und 64 belegten, dass die Kriterien zur Kategorisierung sinnvoll gewählt waren. Student 62 hatte zwar weniger Log-Einträge zu verzeichnen als Student 64 ([s. Datenset time\\_rel\\_con zur Kontinuitätsanalyse](#)). Er war aber, wie auch im obigen Ausschnitt sichtbar, an mehr Wochen und Tagen aktiv als sein Komilitone. Dies musste notwendigerweise bedeuten, dass die Log-Einträge von Student 62 über den Untersuchungszeitraum

kontinuierlicher verteilt waren und entsprechend war seine Kategorisierung auch besser als die von Student 64.

#### 3.4.4. Vergleich des Lern- und Kommunikationsverhaltens

Mit Blick auf die schon bei der Analyse der zeitlichen Lokalität festgestellten Unterschiede, insbesondere bei den Kommunikationsaktivitäten, sollte hier im Anschluss an die Gesamtbetrachtung ebenfalls eine spezifische Untersuchung des Lern- und Kommunikationsverhaltens erfolgen.

Die Datenaufbereitung erfolgte vollkommen identisch zu den Untersuchungen zur Lokalität. Alle weiteren Schritte bis hin zur Kategorisierung waren wiederum analog zu denjenigen bei der vorausgegangenen Kontinuitätsanalyse.

Im Ergebnis zeigten sich bezüglich der Lern- und Kommunikationsaktivitäten schließlich die folgenden Darstellungen:

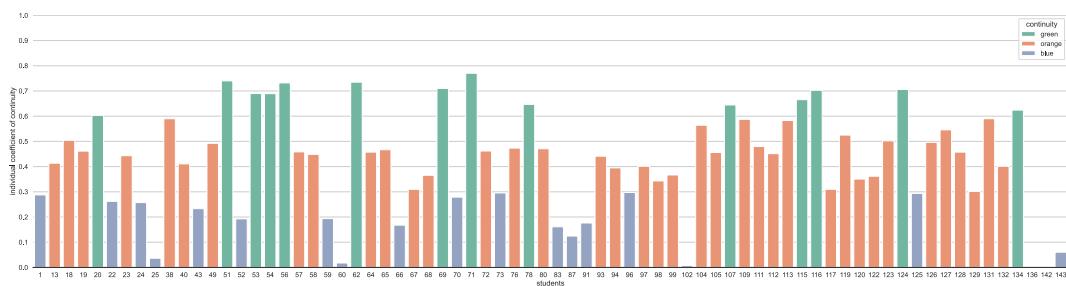


Abbildung 51.: Typisierung nach Kontinuität, Lernverhalten (s. Anhang)

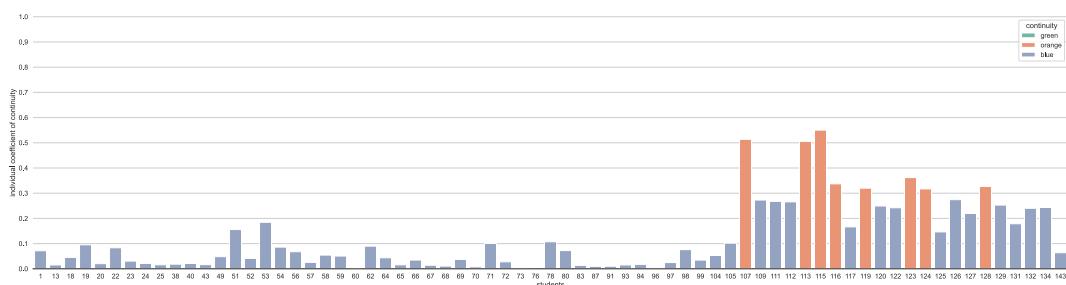


Abbildung 52.: Typisierung nach Kontinuität, Kommunikationsverh. (s. Anhang)

#### Evaluierung

Beide Abbildungen präsentierten ein ähnliches Resultat, wie das bei der Analyse der Lokalität: Während das Untersuchungsergebnis der Lernaktivitäten bis auf wenige Unterschiede dem der Gesamtbetrachtung glich, zeigte das Kommunikationsverhalten erneut ein andersartiges Aussehen.

Auffallend war bei letzterem aber nicht die aufgrund größerer Mengen an Log-Einträgen zu erwartende höhere Kontinuität mancher Studenten, sondern vielmehr die allgemein stark abweichende Kategorisierung.

So war zwar zu beobachten, dass Studenten, die eine höhere Kommunikationskontinuität zeigten, meist auch jene waren, die eine insgesamt hohe Kontinuität besaßen. Viele andere Studenten, die nur wenige Log-Einträge zu Kommunikationsaktivitäten verzeichneten und mithin nur eine geringe Kommunikationskontinuität aufwiesen, bewiesen aber ebenfalls eine hohe Gesamtkontinuität.

Fasste man die Punkte zusammen, so ließ sich daraus also schließen, dass die Kommunikationsaktivität und die dabei gezeigte Kontinuität nur in beschränktem Umfang Einfluss auf das Gesamtergebnis hatte und dieses vielmehr durch die Lernaktivitäten der Studenten bestimmt wurde.

#### 3.4.5. Zusammenfassung

Im Verlauf dieses Kapitels wurde ausgeführt, nach welchen Kriterien die Kontinuität des studentischen Verhaltens bemessen werden konnte und wie die Studenten diesen zufolge in verschiedene Gruppen eingeteilt wurden.

Beginnend mit allgemeinen Beobachtungen wurden zeitliche Maßstäbe in Form von Wochen und Tagen betrachtet und die Bedeutung der individuellen Mengen an Arbeitswochen und Arbeitstagen pro Student erörtert. Es wurde festgestellt, dass die Kontinuität studentischen Verhaltens nur in der gemeinsamen Betrachtung von Wochen- und Tagesaktivitäten zu ermitteln sei.

Im Anschluss daran wurden pro Student sukzessive verschiedene individuelle Kennziffern ermittelt und gespeichert, die wiederum als Grundlage dienen sollten um weitere Werte zu bestimmen, nach welcher schließlich die Größe zur Bestimmung der Kontinuität studentischer Aktivitäten definiert werden konnte.

In einem entsprechenden mathematischen Ausdruck wurden diese für die Ermittlung der Kontinuität relevanten Werte dann so in Beziehung gebracht, dass dieser in seiner maximalen Ausprägung nur den Wert 0 lieferte, wenn ein Student keinerlei Aktivität zeigte bzw. den Wert 1, wenn ein Student dagegen an jedem Tag des betrachteten Zeitraums aktiv war.

Mithilfe dieses mathematischen Ausdrucks, der so für alle Fälle studentischer Aktivitäten ausgelegt war, konnte schließlich die Größe ermittelt werden, wonach Studenten kategorisiert werden konnten. Ihrem Zweck entsprechend wurde diese Größe als *Individueller Kontinuitätskoeffizient (IKK)* bezeichnet.

Die abschließende differenzierte Betrachtung des Lern- und Kommunikationsverhaltens erbrachte im Vergleich zur vorausgegangenen Kategorisierung, dass die Gesamtkontinuität im wesentlichen nur von der Kontinuität des Lernverhaltens bestimmt wurde, nicht aber von den Kommunikationsaktivitäten auch wenn dabei deutliche Unterschiede zwischen den einzelnen Studenten sichtbar wurden.

### 3.5. Dynamik des Lern- und Kommunikationsverhaltens

Die letzte der genannten [Grundfragen zur Analyse des studentischen Verhaltens](#) bildete den Kern der Untersuchungen in diesem Kapitel:

Inwiefern kann man Studenten nach der Dynamik ihres Verhaltens beurteilen und in unterschiedliche Kategorien einteilen?

Von hoher Relevanz bei den Analysen waren also die Änderungen im Umfang der studentischen Aktivitäten, die im Ergebnis Aufschluss darüber geben sollten, mit welchem Einsatz die Studenten ihre Interessen und Ziele verfolgten.

#### 3.5.1. Betrachtung des studentischen Verhaltens im Gesamtzeitraum

Wie zuvor schon bei der Kontinuitätsanalyse wurden zunächst verschiedene zeitliche Beobachtungen vorangestellt, um einen ersten Eindruck von den Ausmaßen der Aktivitätsänderungen zu erhalten.

Dabei wurde insbesondere mit Blick auf die große Übersicht zur [Evaluierung der Kontinuität](#) ersichtlich, dass manche Studenten punktuell oder auch phasenweise eine deutlich erhöhte Aktivität zeigten (erkennbar an der stärkeren Färbung der Tage), andere dagegen zu diesen Zeiten u. U. sogar Lücken aufwiesen (s. a. die nachfolgende Abbildung).

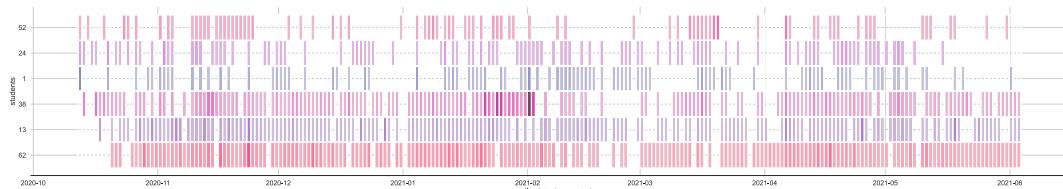


Abbildung 53.: Menge der Log-Einträge für einzelne Studenten ([s. Anhang](#))

Bei genauerer Betrachtung des Bildes konnte man dieses Phänomen dann gerade auch vor und während der Prüfungszeiträume erahnen. Dies motivierte schließlich die Frage, ob es sinnvoll sei, die Dynamik der studentischen Aktivität ähnlich wie deren Kontinuität gleichbleibend über den gesamten Untersuchungszeitraum zu analysieren oder ob das Ergebnis hinsichtlich einer späteren Entwicklung von Kriterien zur Messung von Studienerfolgen nicht mehr Aussagekraft hätte, nähme man die Aktivitätsänderungen im Umfeld der Prüfungen gezielter in den Blick.

In Anbetracht dessen, dass eine angemessene Vorbereitung und ein damit verbundener höherer Einsatz für einen Prüfungserfolg unabdingbar ist, eine erhöhte Aktivität zu Semesterbeginn daran aber wohl nur wenig Anteil hat, war offensichtlich, dass bei der Betrachtung der Dynamik der pauschale Ansatz der vorherigen Kontinuitätsanalyse nicht verfing. Vielmehr sollte hier die Analyse miteinbeziehen, dass die Prüfungen in der Regel die Schlussphase eines Semesters darstellen und

die Leistungskurve der Studenten zu diesen Wochen hin ansteigen sollte, um das Semester erfolgreich beenden zu können.

Im Anschluss an diese allgemeinen Überlegungen stellte sich sogleich konkret die Frage, wie es sich dann mit den zwei Prüfungszeiträumen (PZ) verhielt. Wollte man beide PZ betrachten, so ergab sich das Problem, dass Studenten die im PZ1 bereits alle Prüfungen absolviert hatten, zwangsläufig unter ihrem Niveau bewertet würden, weil ihre Leistungskurve zum PZ2 hin abfiel. Untersuchte man nur einen PZ, nahm man ebenfalls in Kauf, dass manche Studenten falsch evaluiert würden.

Die Lösung war schließlich, nur die Tage bis zum Beginn des PZ1 zu betrachten. Bis dahin mussten die Studenten in den Fachkursen in der Regel die Prüfungsvorleistungen erbringen, unabhängig davon in welchem PZ sie anschließend an den Prüfungen teilnehmen wollten. Entsprechend musste bis zu diesem Zeitpunkt dann auch eine erhöhte Aktivität festgestellt werden können.

Damit war nur noch zu klären, wie mit den Aktivitätsrückgängen in der Weihnachtszeit umzugehen war. Diese nahmen natürlich Einfluss auf das Ergebnis der Dynamikbetrachtung, waren aber durchgängig bei fast allen Studenten zu beobachten und hätten daher auch ignoriert werden können. Dennoch wurde entschieden, die Woche vor und nach Weihnachten bei den folgenden Analysen nicht zu beachten, um eine allgemeine Absenkung der absoluten Dynamik zu vermeiden.

### **Datenaufbereitung**

Nachdem die Datensätze entsprechend den vorausgegangenen Überlegungen für den relevanten Zeitraum markiert wurden, wurden erneut nur die Daten ausgewählt, die das Lern- und Kommunikationsverhalten für alle Studiengänge umfassen (s. a. Listing zur [Definition der Arbeitsdaten](#)).

---

```

1 # Ergänzung von Merkmalen zur Unterscheidung von Semesterabschnitten
2 md['semesterperiod'] = 'other'
3 md.loc[(md.timecreated < pd.to_datetime('2020-12-18')) |
4         ((md.timecreated > pd.to_datetime('2021-01-01')) &
5          (md.timecreated < pd.to_datetime('2021-01-22'))),
6         ['semesterperiod']] = 'before_exams1'
7
8 # Definition der Arbeitsdaten
9 md = md[(md['behaviour'] == 'learning') |
10          (md['behaviour'] == 'communication')]

```

---

Listing 55: Vorbereitung der Arbeitsdaten

#### **3.5.2. Ermittlung der Vergleichsgöße**

Im Folgenden ging es nun darum, ähnlich zu dem Vorgehen bei der Kontinuitätsanalyse, die Voraussetzungen zu erarbeiten, auf deren Grundlage anschließend eine entsprechende Größe zur Kategorisierung der Studenten definiert werden konnte.

Für eine Analyse auf Tagesbasis, wie sie sich nach den einleitenden Überlegungen

empfahl, wurde zunächst das bisherige Datenset noch um ein weiteres Merkmal *year\_day* zur Identifikation der Kalendertage ergänzt.

Anschließend sollte auch hier ein neues Datenset *time\_rel\_dyn* erstellt werden, um für jeden Studenten verschiedene individuelle Werte, wie die Menge der Log-Einträge und die der aktiven Tage aufnehmen zu können. Diese Werte waren erforderlich, weil sich nur so ein individueller Durchschnitt ermitteln ließ, der danach als kontante Bezugsgröße zur Berechnung der Aktivitätsänderungen dienen konnte.

### Datenaufbereitung

---

```

1 # Ermittlung der Menge der Log-Einträge pro Student
2 loggings_user = pd.Series(md.userid[
3     md.userstatus == 'student'].groupby(md.userid).count(),
4     name='loggings')
5
6 # Ermittlung der Menge der Arbeitstage pro Student
7 days_user = pd.Series(md.year_day[
8     md.userstatus == 'student'].groupby(md.userid).nunique(),
9     name='days')
10
11 # Erstellung des neuen Datensets
12 time_rel_dyn = pd.concat([loggings_user, days_user], axis=1)
13
14 # Erstellung einer neuen Spalte für die durchschnittlichen
15 # Mengen der Log-Einträge pro Arbeitstag
16 time_rel_dyn['avg_count_per_day'] = 0
17 time_rel_dyn.loc[(time_rel_dyn['avg_count_per_day'] == 0),
18     ['avg_count_per_day']] =
19         (loggings_user / days_user).astype(int)

```

---

Listing 56: Erstellung des Datensets zur Aufnahme individueller Kennziffern

userid	loggings	days	avg_count_per_day
1	1324	94	14
13	2759	183	15
18	1128	134	8
19	3004	190	15
20	3733	173	21
...	...	...	...
132	1555	151	10
134	3063	183	16
136	18	4	4
142	3	2	1
143	538	60	8

Abbildung 54.: Datenset *time\_rel\_dyn* zur Dynamikanalyse ([s. Anhang](#))

Mit diesem neuen Datenset waren bereits alle notwendigen Kennziffern ermittelt, um im nächsten Schritt auch die Größe zu bestimmen, nach der die Dynamik der studentischen Aktivitäten ermittelt werden konnte. Folgende Bedingungen sollte die gesuchte Größe dabei erfüllen:

1. Der individuelle Tagesschnitt war so zu berücksichtigen, dass sich im Vergleich mit der Menge an Log-Einträgen eines einzelnen Tages ein positiver oder negativer Wert als Abweichung ergab.
2. War ein Student an einem Tag nicht aktiv, sollte der negative Tagesdurchschnitt als Abweichung in die Größe einfließen.

3. Die Abweichung war gemäß den vorausgegangen Ausführungen mit einem Faktor zu gewichten, der zum Ende des Untersuchungszeitraums hin bis zum Wert 1 ansteigen sollte und tägliche Abweichungen auf diese Weise immer größeren Einfluss nehmen konnten.
4. Die Summe aller positiven und negativen Abweichungen sollte schließlich die gesuchte Größe ergeben.

Unter Berücksichtigung der vorgenannten Bedingungen konnte im Anschluss der folgende mathematische Ausdruck formuliert werden:

$$\begin{aligned}
 & \text{Menge der Log-Einträge} \\
 & \text{des betrachteten Tages} - \frac{\text{Menge der Log-Einträge}}{\text{im Untersuchungszeitraum}} \times \frac{\text{Ordinalwert des}}{\text{betrachteten Tages}} \longrightarrow \max \\
 & \qquad\qquad\qquad \longrightarrow \pm \infty \\
 & \qquad\qquad\qquad \longrightarrow 1 \\
 & \qquad\qquad\qquad = \text{Tagesdurchschnitt } (\emptyset) \\
 & \qquad\qquad\qquad \longrightarrow -\emptyset
 \end{aligned}$$

Abbildung 55.: Individueller Dynamikkoeffizient, IDK ([s. Anhang](#))

Entsprechend der obigen Formulierung ergibt sich bei Addition aller gewichteter Tagesabweichungen ein beliebiger positiver oder negativer Wert. Ein Ergebnis nahe dem Wert 0 impliziert dabei ein wenig dynamisches Verhalten, während eine große positive oder negative Summe auf eine hohe Dynamik schließen lässt.

### 3.5.3. Kategorisierung nach IDK

War mit dem individuellen Dynamikkoeffizienten (IDK) also die gesuchte Größe ermittelt worden, wonach die Dynamik des studentischen Verhaltens zu messen war, sollte diese Kennziffer nun für jeden Studenten individuell berechnet und in das Datenset *time\_rel\_dyn* aufgenommen werden, so dass danach eine Typisierung vorgenommen werden konnte.

#### Datenaufbereitung

```

1 list_idk = list() # leeres Listen-Objekt zur Aufnahme der Ergebnisse
2
3 # Definition eines Listen-Objekts mit allen
4 # Kalendertagen im Untersuchungszeitraum
5 days_in_period =
6     md.timecreated[(md.semesterperiod ==
7         'before_exams1')].sort_values().dt.dayofyear.unique()
8
9 # Definition der Menge an Tagen im Untersuchungszeitraum
10 days_in_period_count =
11     md.timecreated[(md.semesterperiod ==
12         'before_exams1')].groupby(md.year_day).unique().size
13
14 # Funktion zur Ermittlung des individuellen Dynamikkoeffizienten (IDK)
15 def get_idk(i, list_idk, days_in_period, days_in_period_count):
16     # Array mit individuellen Arbeitstagen
17     days_user = md.timecreated[

```

```

18     md.userid == time_rel_dyn.iloc[i]['userid']
19     ].sort_values().dt.dayofyear.unique()
20 x = 1
21 idk = 0
22 for d in days_in_period:
23     if d in days_user:
24         for row in md.year_day[
25             (md.semesterperiod == 'before_examsl') &
26             (md.userid == time_rel_dyn.iloc[i]['userid']) &
27             (md.year_day == d) &
28             ((md['behaviour'] == 'learning') |
29              (md['behaviour'] == 'communication'))]
29             ].groupby(md.year_day).count():
30                 idk +=
31                     (row - time_rel_dyn.iloc[i]['avg_count_per_day']) *
32                     (x / days_in_period_count)
33
34     else:
35         idk +=
36             (0 - time_rel_dyn.iloc[i]['avg_count_per_day']) *
37             (x / days_in_period_count)
38     x += 1
39 list_idk.append(idk)
40
41 # Schleife zur Steuerung der Ermittlungsfunktion
42 for i in time_rel_dyn.index:
43     get_idk(i, list_idk, days_in_period, days_in_period_count)
44
45 # Erstellung eines DataFrame-kompatiblen Series-Objekts
46 idk = pd.Series(list_idk, name='idk')
47
48 # Ergänzung des IDK im Datenset time_rel_dyn
49 time_rel_dyn = pd.concat([time_rel_dyn, idk], axis=1)

```

Listing 57: Ermittlung des individuellen Dynamikkoeffizienten, IDK

Nachdem das Datenset für jeden einzelnen Studenten mit entsprechendem IDK ergänzt wurde, stand dieser für die abschließende Kategorisierung zur Verfügung. Auch hierfür wurde im Datenset eine neue Spalte hinzugefügt und dieses mit Angaben zur Dynamik vervollständigt.

### Datenaufbereitung

```

1 # Erstellung einer neuen Spalte zur Typisierung
2 time_rel_dyn['dynamic'] = ''
3
4 # Einordnung der Studenten nach der Dynamik ihrer Aktivitäten
5 time_rel_dyn.loc[(time_rel_dyn['idk'] > 0), ['dynamic']] = 'green'
6 time_rel_dyn.loc[(time_rel_dyn['idk'] < 0), ['dynamic']] = 'orange'

```

Listing 58: Typisierung der Studenten nach der Dynamik der Aktivitäten

Die abschließende Untersuchung sollte nun die zu Beginn formulierte Frage beantworten und veranschaulichen, inwiefern man Studenten nach der Dynamik ihres Verhaltens in unterschiedliche Kategorien einteilen konnte.

### Datenanalyse: Typisierung der Studenten nach IDK

```

1 # Visualisierung der Typisierung der Studenten nach IDK
2 chart = sns.barplot(x=time_rel_dyn.userid.astype(int),
3                      y=time_rel_dyn.idk, hue=time_rel_dyn.dynamic,
4                      hue_order=['green', 'orange'],
5                      dodge=False, palette='Set2')

```

Listing 59: Typisierung der Studenten nach IDK

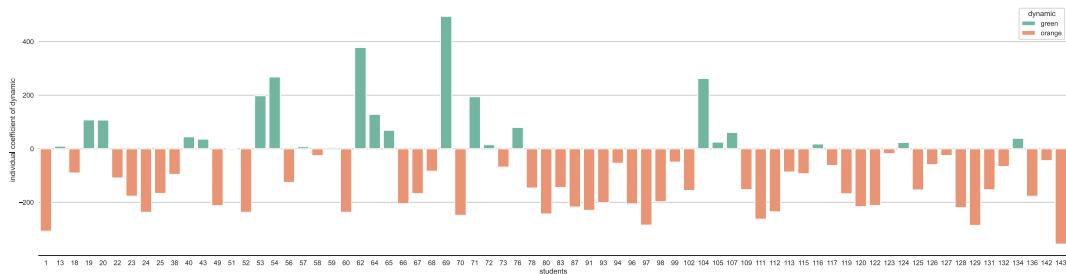


Abbildung 56.: Typisierung der Studenten nach IDK (s. Anhang)

### Evaluierung

Wie man im Diagramm sehen konnte, zeigte nur knapp ein Drittel aller Studenten eine insgesamt positive Dynamik, die sich erkennbar in der Übersicht zu den Arbeitstagen pro Student (s. Evaluierung der Kontinuität) oder im folgenden Ausschnitt auch in einer vermuteten erhöhten Aktivität in den Wochen vor dem PZ1 ausdrückte.

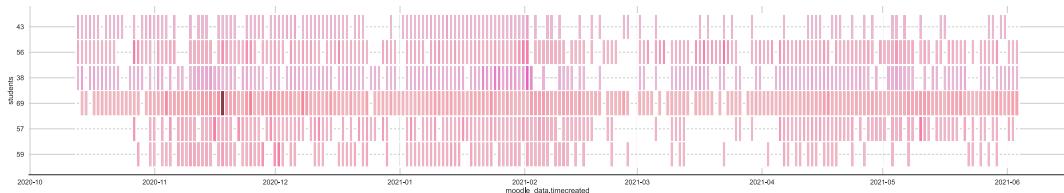


Abbildung 57.: Menge der Log-Einträge für einzelne Studenten (s. Anhang)

Insofern schien die Kategorisierung grundsätzlich sinnvoll und angemessen. Aufgrund von Beobachtungen wie z. B. bei den Studenten 38 und 56, die im Januar ebenfalls sehr aktiv waren, jedoch insgesamt nur eine negative Dynamik aufwiesen, war zu vermuten, dass die linear ansteigende Gewichtung der Arbeitstage die Ergebnisse insgesamt noch etwas zu sehr egalisiert belief.

Hätte man hier z. B. eine quadratisch ansteigende Gewichtung gewählt, wären die Aktivitäten in den letzten Wochen vor den Prüfungen sicher noch einmal stärker bewertet worden und hätten vermutlich in dem einen oder anderen Fall zu einer anderen Kategorisierung geführt.

#### 3.5.4. Vergleich des Lern- und Kommunikationsverhaltens

Zum Abschluss der Betrachtungen zur Dynamik sollte ebenfalls noch das Lern- und Kommunikationsverhalten auf Besonderheiten hin getrennt untersucht werden.

Die Datenaufbereitung erfolgte hier erneut wie zuvor bei den Untersuchungen zur Lokalität bzw. Kontinuität. Sämtliche weitere Schritte bis hin zur Typisierung der Studenten waren dann analog zu denen bei der vorherigen Dynamikanalyse.

Im Ergebnis zeigten sich hinsichtlich des Lern- und Kommunikationsverhaltens schließlich die folgenden Darstellungen:

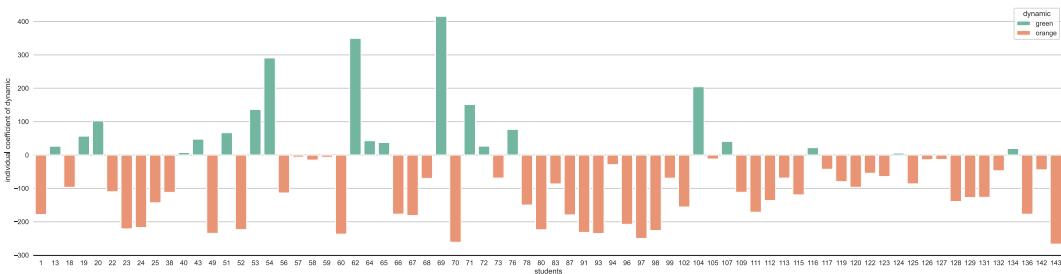


Abbildung 58.: Typisierung nach Dynamik, Lernverhalten ([s. Anhang](#))

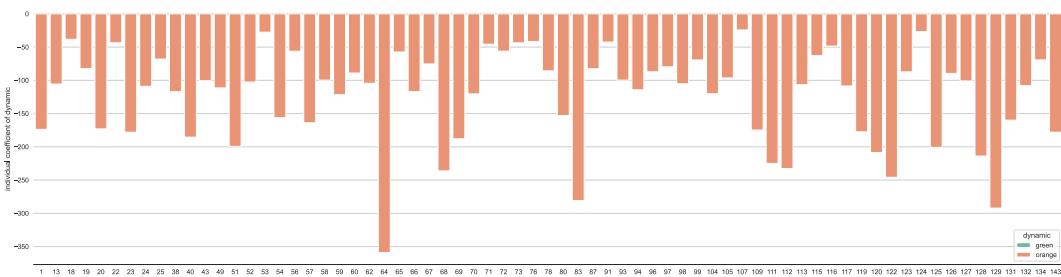


Abbildung 59.: Typisierung nach Dynamik, Kommunikationsverh. ([s. Anhang](#))

### Evaluierung

Mit Kenntnis der Analyseergebnisse zur Lokalität und zur Kontinuität zeigten die Resultate zur getrennten Dynamikbetrachtung des Lern- und Kommunikationsverhaltens das bereits erwartete Bild: Während das Untersuchungsergebnis zum Lernverhalten dem der vorausgegangenen Gesamtbetrachtung recht ähnlich war, stellte sich die Kommunikationsdynamik ganz anders dar.

Auch hier konnte man also feststellen, dass die Kommunikationsaktivität und die dabei gezeigte Dynamik das Gesamtergebnis kaum beeinflusste, sondern dieses fast ausschließlich durch die Lernaktivitäten der Studenten bestimmt wurde.

### 3.5.5. Zusammenfassung

In diesem Kapitel wurde vorgestellt, auf welche Weise die Studenten entsprechend der Dynamik ihres Verhaltens beurteilt und kategorisiert wurden.

Zu Beginn wurden zunächst verschiedene zeitliche Zusammenhänge erhöhter studentischer Aktivität betrachtet und diskutiert, wie die Dynamik auch im Hinblick auf eine spätere Entwicklung von Kriterien zur Messung von Studienerfolgen zu untersuchen sei. Es wurde festgestellt, dass die Dynamik nicht gleichbleibend untersucht werden konnte wie die Kontinuität, sondern mit Blick auf eine zu den Prüfungen hin idealerweise ansteigende Leistungskurve sinnvoll gewichtet werden musste. Hinsichtlich der Problematik mit den zwei Prüfungszeiträumen und den Aktivitätsrückgängen um Weihnachten wurde entschieden, diese aus der Betrachtung herauszunehmen.

Anschließend wurden für die Studenten mehrere individuelle Werte ermittelt

und nach und nach in ein neues Datenset aufgenommen. Auf Basis dieser Werte konnte schließlich die Größe definiert werden, die die Dynamik der Aktivitäten beschrieb und hiernach die Kategorisierung der Studenten ermöglichte.

Wie bereits zur Messung der individuellen Kontinuität, wurde zur Bewertung der individuellen Dynamik ebenfalls ein mathematischer Ausdruck entwickelt, der auf Basis vorab definierter Bedingungen die bereitgestellten Werte so in Beziehung brachte, dass ein Ergebnis nahe 0 ein wenig dynamisches Verhalten implizierte, ein betragsmäßig großer Wert dagegen ein Verhalten mit hoher Dynamik.

Nach Übersetzung des mathematischen Ausdrucks in einen lauffähigen Algorithmus konnte schließlich für jeden Studenten eine individueller Wert ermittelt werden, der als Bezugsgöße für die abschließende Kategorisierung dienen konnte. Ihrem Zweck entsprechend wurde diese Größe als *Individueller Dynamikkoeffizient (IDK)* bezeichnet.

Die nachfolgende getrennte Analyse des Lern- und Kommunikationsverhaltens zeigte wie schon die differenzierte Kontinuitätsanalyse, dass die Gesamtkontinuität im wesentlichen nur von der Kontinuität des Lernverhaltens bestimmt wurde, das Kommunikationsverhalten in diesem Kontext mithin nicht relevant war.

## 4. Ergebnisse

In dieser Arbeit wurde gezeigt, wie auf Basis des vom *Projektteam DiSEA* zur Verfügung gestellten Datenbestandes Analysen durchgeführt wurden, die dazu geeignet waren, grundlegende Erkenntnisse über das studentische Verhalten zu gewinnen und hiernach sinnvolle Kategorisierungen vornehmen zu können.

Da der Datenbestand selbst keinerlei Informationen über den offiziellen Status eines Benutzers enthielt, mithin die Identität eines Studenten nicht unmittelbar bestimmt werden konnte, waren zu Beginn verschiedene *aktivitätsbezogene Analysen* durchzuführen anhand derer schließlich eine Gruppe von 72 Benutzern mit hoher Wahrscheinlichkeit als Studenten identifiziert wurden. Erst nach diesen einleitenden Analysen zu benutzerspezifischen Aktivitäten, konnten die eigentlichen Untersuchungen des studentischen Lern- und Kommunikationsverhaltens beginnen.

Dabei standen nun *zeitbezogene Analysen* im Fokus, deren Ergebnisse im Hinblick auf eine spätere Nutzung im Rahmen des *DiSEA-Projekts* von Interesse sein konnten. Konkret ging es um die Beantwortung der folgenden grundlegenden Fragen:

- Wie lassen sich Studenten nach der zeitlichen *Lokalität* der Lern- und Kommunikationsaktivitäten einordnen?
- Auf welche Weise lassen sich Studenten nach der *Kontinuität* ihres Handelns in verschiedene Gruppen unterscheiden?
- Inwiefern kann man Studenten nach der *Dynamik* ihres Verhaltens beurteilen und in unterschiedliche Kategorien einteilen?

### *Lokalität des Lern- und Kommunikationsverhaltens*

Hinsichtlich des zeitlichen Auftretens studentischer Aktivitäten wurde festgestellt, dass die Vermutung, Studenten in digitalen Studienformaten seien aufgrund von Berufstätigkeit und familiären Verpflichtungen überwiegend außerhalb normaler Arbeitszeiten aktiv, nicht mit den realen Gegebenheiten übereinstimmte.

Es konnte vielmehr belegt werden, dass die Studenten in der Gesamtheit sowohl auf Wochensicht wie auch auf Tagessicht überwiegend zu den üblichen Geschäftzeiten aktiv waren. Genauere Untersuchungen des individuellen studentischen Verhaltens zeigten nur wenige Abweichungen davon.

Entsprechend waren die anschließenden Kategorisierungen sehr unausgewogen: Die Einordnung nach Tagestypen erbrachte ein fast einheitliches Ergebnis, während die Typisierung nach Tageszeiten nur etwas mehr als 10 Studenten aufzeigte, die den größten Teil ihrer Aktivitäten außerhalb normaler Arbeitszeiten ausführte.

### Kontinuität des Lern- und Kommunikationsverhaltens

Verschiedene Betrachtungen zu zeitlichen Bezügen in Form von Wochen und Tagen zeigten, dass die Kontinuität mit der Studenten ihre Aktivitäten ausführten nicht allein auf Wochenbasis, sondern nur in einer gemeinsamen Analyse individueller Wochen- und Tagesaktivitäten bestimmt werden konnte.

Diese Erkenntnisse und weitere Überlegungen waren dann die Grundlage, um im Folgenden für die Studenten individuelle Kennziffern zu ermitteln und darauf aufbauend weitere Werte, nach denen die Größe zur Bestimmung der Kontinuität definiert werden konnte. Folgende Bedingungen sollte die gesuchte Größe erfüllen:

1. Gemäß den bisherigen Ergebnissen sollten die ermittelten Mengen an Wochen und Tagen im individuellen Toleranzbereich berücksichtigt werden.
2. Als feste Bezugsgröße war der Zeitraum miteinzubeziehen, innerhalb dessen die Kontinuität festgestellt werden sollte.
3. War ein Student nie aktiv, musste der mathematische Ausdruck im Ergebnis den Wert 0 ergeben.
4. War ein Student an jedem Tag und somit dann auch in jeder Woche aktiv, sollte der mathematische Ausdruck den maximalen Wert liefern.

Die Formulierung des folgenden mathematischen Ausdrucks brachte im Anschluss alle Bedingungen und Größen in einen sinnvollen Zusammenhang:

$$\frac{\text{Menge der Arbeitswochen im Toleranzbereich} \longrightarrow 0}{\text{Gesamtmenge der Wochen im Untersuchungszeitraum}} \times \frac{\text{Menge der Arbeitstage im Toleranzbereich} \longrightarrow 0}{\text{Gesamtmenge der Tage im Untersuchungszeitraum}} \longrightarrow 0$$

$$\rightarrow 0 \qquad \qquad \qquad \rightarrow 0$$
  

$$\frac{\text{Menge der Arbeitswochen im Toleranzbereich} \longrightarrow \max}{\text{Gesamtmenge der Wochen im Untersuchungszeitraum}} \times \frac{\text{Menge der Arbeitstage im Toleranzbereich} \longrightarrow \max}{\text{Gesamtmenge der Tage im Untersuchungszeitraum}} \longrightarrow 1$$

$$\rightarrow 1 \qquad \qquad \qquad \rightarrow 1$$

Abbildung 60.: Individueller Kontinuitätskoeffizient, IKK ([s. Anhang](#))

Der mathematische Ausdruck war hier so definiert, dass er in seiner maximalen Ausprägung nur den Wert 0 lieferte, wenn ein Student keine Aktivität zeigte bzw. den Wert 1, wenn ein Student an jedem Tag des betrachteten Zeitraums aktiv war. In den anderen Fällen ergab sich immer ein Wert innerhalb des Intervalls zwischen 0 und 1. Ein Wert außerhalb des Intervalls war dagegen nicht möglich.

Mithilfe dieses mathematischen Ausdrucks, der so für alle Fälle studentischer Aktivitäten ausgelegt war, konnte schließlich die Größe ermittelt werden, wonach Studenten kategorisiert werden konnten. Ihrem Zweck entsprechend wurde diese Größe als *Individueller Kontinuitätskoeffizient (IKK)* bezeichnet.

### Dynamik des Lern- und Kommunikationsverhaltens

Hinsichtlich gegebener zeitlicher Zusammenhänge erhöhter studentischer Aktivität und der Frage, wie die Dynamik auch im Hinblick auf eine spätere Entwicklung von Kriterien zur Messung von Studienerfolgen zu untersuchen sei, wurde erkannt, dass die Dynamik nicht gleichbleibend untersucht werden konnte wie die Kontinuität.

Mit Blick auf eine idealerweise zu den Prüfungen hin ansteigende Leistungskurve wurde daher entschieden, die studentischen Aktivitäten bei dieser Untersuchung der Dynamik zu gewichten, so dass den Tagen und Wochen unmittelbar vor den Prüfungen eine größere Bedeutung zukam. Die Prüfungszeiträume selbst und auch die Wochen dazwischen sollten aber aus Gründen einer fairen Bewertung nicht in die Betrachtung miteinfließen. Gleichermassen wurde beschlossen, die Wochen um Weihnachten herum aus der Betrachtung zu entfernen, dies jedoch nur um eine generelle Absenkung der Dynamik zu vermeiden.

Nach diesen Vorüberlegungen wurden ähnlich wie bei der Kontinuitätsanalyse verschiedene Werte für die einzelnen Studenten ermittelt, auf deren Grundlage die Größe definiert werden konnte, die die Dynamik der Aktivitäten beschrieb. Die nachfolgenden Bedingungen musste die gesuchte Größe erfüllen:

1. Der individuelle Tagesschnitt war so zu berücksichtigen, dass sich im Vergleich mit der Menge an Log-Einträgen eines einzelnen Tages ein positiver oder negativer Wert als Abweichung ergab.
2. War ein Student an einem Tag nicht aktiv, sollte der negative Tagesdurchschnitt als Abweichung in die Größe einfließen.
3. Die Abweichung war gemäß den vorausgegangen Ausführungen mit einem Faktor zu gewichten, der zum Ende des Untersuchungszeitraums hin bis zum Wert 1 ansteigen sollte und tägliche Abweichungen auf diese Weise immer größeren Einfluss nehmen konnten.
4. Die Summe aller positiven und negativen Abweichungen sollte schließlich die gesuchte Größe ergeben.

Der folgende mathematische Ausdruck wurde so entwickelt, dass er schließlich alle notwendigen Bedingungen und Größen sinnvoll in Beziehung brachte:

$$\begin{aligned}
 & \text{Menge der Log-Einträge} && \text{Menge der Log-Einträge} && \text{Ordinalwert des} \\
 & \text{des betrachteten Tages} &-& \text{im Untersuchungszeitraum} &\longrightarrow \max & \longrightarrow +\infty \\
 & \longrightarrow 0 && \text{Menge der Arbeitstage} && \text{Gesamtmenge der Tage} \\
 & && \text{im Untersuchungszeitraum} && \text{im Untersuchungszeitraum} \\
 & && = \text{Tagesdurchschnitt } (\emptyset) && \longrightarrow 1 \\
 & && \swarrow && \downarrow \\
 & && \longrightarrow -\emptyset &&
 \end{aligned}$$

Abbildung 61.: Individueller Dynamikkoeffizient, IDK ([s. Anhang](#))

## 4. Ergebnisse

Mit dem obigen mathematischen Ausdruck, ergab sich bei Addition aller gewichteter Tagesabweichungen ein beliebiger positiver oder negativer Wert. Ein Ergebnis nahe dem Wert 0 implizierte dabei ein wenig dynamisches Verhalten, während eine großer positiver oder negativer Wert auf eine hohe Dynamik schließen ließ.

Nach Übersetzung des mathematischen Ausdrucks in einen lauffähigen Algorithmus konnte schließlich für jeden Studenten eine individueller Wert ermittelt werden, der als Bezugsgösse für die abschließende Kategorisierung dienen konnte. Ihrem Zweck entsprechend wurde diese Größe als *Individueller Dynamikkoeffizient (IDK)* bezeichnet.

### **Vergleich des Lern- und Kommunikationsverhaltens**

Die nach Gesamtanalysen jeweils durchgeführten getrennten Betrachtungen des Lern- und Kommunikationsverhaltens erbrachten im wesentlichen stets das selbe Resultat: Das studentische Gesamtverhalten wurde zum Großteil durch das Lernverhalten bestimmt und nur marginal vom Kommunikationsverhalten beeinflusst, auch wenn bei letzterem deutliche Unterschiede zwischen den einzelnen Studenten festgestellt werden konnten.

## 5. Ausblick

Manche Überlegung oder Frage, die im Laufe dieser Arbeit auftrat, hätte sicher eine eingehendere Betrachtung verdient, sie hätte aber leider auch die Überschreitung des zeitlichen oder thematischen Rahmens dieser Arbeit bedeutet.

Daher soll nun im Folgenden zumindest näherungsweise skizziert und aufgezeigt werden, um welche Aspekte es sich dabei handelte und in welcher Richtung eine sinnvolle Fortsetzung dieser Arbeit möglich wäre.

Bei den differenzierten Analysen zur Lokalität des Lern- und Kommunikationsverhaltens war offensichtlich geworden, dass die Kommunikationsaktivitäten der Studenten insgesamt sehr unterschiedlich ausfielen und es wurde vermutet, dass dies auf eine studiengangsspezifische Besonderheit hindeuten könnte. Insofern es also für das DiSEA-Projekt zweckmäßig sein könnte, wäre es empfehlenswert, die zeitbezogenen Analysen in dieser Arbeit auch auf *Studiengangsbasis* auszuführen, um noch einmal etwas genaueres Erkenntnisse gewinnen zu können.

Auch könnte man zeitliche Bezüge des studentischen Verhaltens im Kontext mit dem Verhalten Anderer untersuchen und so die Frage angehen, wie sich Studenten nach ihrem *Reaktionsverhalten* einordnen lassen? Eine Antwort auf diese Frage gäbe z. B. Aufschluss darüber, wie schnell Studenten auf das Verhalten Anderer reagieren und einen neu eingestellten Selbstlerntest oder eine neue Aufgabe bearbeiten.

Ferner könnte man den Fokus der Betrachtung auch auf die Fach- bzw. Modulkurse richten. Gerade in aktivitätsbezogener Hinsicht ergäben sich hier Ansätze zu Analysen, so z. B. zur Frage: Wie lassen sich Studenten nach *Kursaktivitäten* unterscheiden? Hier ginge es um eine Klassifizierung des Lern- und Kommunikationsverhaltens bezüglich konkreter Aktivitäten wie das Aufrufen von Lehrmaterialien, die Teilnahme an Forumsdiskussionen oder die Bearbeitung von Aufgaben. Auch könnte es von Interesse sein, im Kurskontext das individuelle Verhalten dahingehend zu analysieren, in welchen Kursen ein Student aktiv war oder ob er z. B. auch abgeschlossene Kurse in seine Lernaktivitäten einbezogen hat.

Ergänzend wäre ebenso die Untersuchung des individuellen *Vernetzungsgrades*, also der Art und des Umfangs des Austauschs mit anderen Personen, sicher gut für eine Kategorisierung geeignet und könnte insbesondere auch bei der Bestimmung von Kriterien zur Messung von Studienerfolgen im Rahmen des DiSEA-Projekts von Nutzen sein.

### ***Kategorisierung anhand von Musterstudenten***

Einem anderen Gedanken folgend, der nicht konkret eine neue spezifische Frage formuliert, sondern eher eine andere Untersuchungsmethodik impliziert, könnten vergleichende Analysen anhand sogenannter *Musterstudenten* ebenfalls interessante Einblicke in das studentische Verhalten ermöglichen.

Hierzu könnten, ähnlich dem Vorgehen bei der Identifikation von Studenten, auf Basis des zur Verfügung gestellten Datenbestands Studenten ermittelt werden, die in aktivitäts- oder zeitbezogener Hinsicht ein typisches Verhalten zeigen.

So wäre es u. a. denkbar, zu Beginn mehrere Musterstudenten zu bestimmen, die

- die größten Mengen an Log-Einträgen zu bestimmten Actions zeigen,
- eingestellte Aufgaben überdurchschnittlich schnell abgearbeitet haben,
- über das Semester hinweg in keinem der aktuellen Kurse aktiv waren,
- sich weder aktiv noch passiv an Forumsdiskussionen beteiligt haben.

Je nachdem wie sich ein Musterverhalten dann präsentiert, können infolgedessen andere Studenten ermittelt werden, die ein ähnliches Verhalten aufweisen und sich so in einer Kategorie zusammenfassen lassen. Ausreißer, die ein gänzlich anderes Verhalten zeigen, können dann u. U. selbst erneut als Musterstudenten definiert werden, eine neue Kategorie begründen und neue vergleichende Untersuchungen motivieren.

## Literaturverzeichnis

- Axmark, D. & Widenius, M. (2022). *MySQL 5.7 Reference Manual*. Zugriff am 2022-07-23 auf <https://dev.mysql.com/doc/refman/5.7/en/>
- Azevedo, A. & Santos, M. (2008, 01). KDD, SEMMA and CRISP-DM: A parallel overview. In (S. 182-185).
- Bubenhofer, N. & Kupietz, M. (Hrsg.). (2018). *Visualisierung sprachlicher Daten. Visual Linguistics – Praxis – Tools*. Heidelberg: Heidelberg University Publishing. doi: 10.17885/heiup.345.474
- Dietterich, T. (1995). Overfitting and undercomputing in machine learning. *ACM computing surveys (CSUR)*, 27 (3), 326–327.
- Fayyad, U., Piatetsky-Shapiro, G. & Smyth, P. (1996, Mar.). From data mining to knowledge discovery in databases. *AI Magazine*, 17 (3), 37. Zugriff auf <https://ojs.aaai.org/index.php/aimagazine/article/view/1230> doi: 10.1609/aimag.v17i3.1230
- Green, M. (2022). *The Moodle Database. Table and relationship documentation generated from moodle source code*. Zugriff am 2022-04-08 auf <https://www.examulator.com/er/>
- Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., ... Oliphant, T. E. (2020, September). Array programming with NumPy. *Nature*, 585 (7825), 357–362. Zugriff auf <https://doi.org/10.1038/s41586-020-2649-2> doi: 10.1038/s41586-020-2649-2
- Janneck, M., Merceron, A. & Sauer, P. (2021, March). Disease: Analysing success and dropout in online-degrees.. Zugriff auf <https://drive.google.com/drive/folders/1xXAKWkTlCeMb3PsmkiPEjNRq8G2pXelq>
- Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B., Bussonnier, M., Frederic, J., ... Willing, C. (2016). Jupyter notebooks – a publishing format for reproducible computational workflows. In F. Loizides & B. Schmidt (Hrsg.), *Positioning and power in academic publishing: Players, agents and agendas* (S. 87 - 90).
- Kusnierz, A. (2020). *Studienperformanz - Interaktive Visualisierung der Daten. Bachelorarbeit Medieninformatik, Beuth Hochschule für Technik*.
- Moodle. (2022). *The Moodle Documentation. Version 4.0*. Zugriff am 2022-06-30 auf [https://docs.moodle.org/400/en/Main\\_page](https://docs.moodle.org/400/en/Main_page)
- Polasek, W. (1994). Explorative und Deskriptive Statistik. In *EDA Explorative Datenanalyse: Einführung in die deskriptive Statistik* (S. 3–15). Berlin, Heidelberg: Springer Berlin Heidelberg. Zugriff auf [https://doi.org/10.1007/978-3-642-57889-2\\_2](https://doi.org/10.1007/978-3-642-57889-2_2) doi: 10.1007/978-3-642-57889-2\_2
- Rönz, B., Strohe, H. G. & Eckstein, P. (1994). *Lexikon Statistik*. Gabler.

- Runkler, T. A. (2020). Introduction. In *Data analytics: Models and algorithms for intelligent data analysis* (S. 1–4). Wiesbaden: Springer Fachmedien. Zugriff auf [https://doi.org/10.1007/978-3-658-29779-4\\_1](https://doi.org/10.1007/978-3-658-29779-4_1) doi: 10.1007/978-3-658-29779-4\_1
- Schumann, H. & Müller, W. (2000). Einleitung. In *Visualisierung: Grundlagen und allgemeine Methoden* (S. 1–3). Berlin, Heidelberg: Springer Berlin Heidelberg. Zugriff auf [https://doi.org/10.1007/978-3-642-57193-0\\_1](https://doi.org/10.1007/978-3-642-57193-0_1) doi: 10.1007/978-3-642-57193-0\_1
- Shearer, C. (2000). The CRISP-DM Model: The new blueprint for data mining. *Journal of Data Warehousing*, 5 (4).
- Tukey, J. W. et al. (1977). *Exploratory data analysis* (Bd. 2). Reading, MA.
- Van Rossum, G. & Drake Jr, F. L. (1995). *Python tutorial* (Bd. 620). Centrum voor Wiskunde en Informatica Amsterdam, The Netherlands.
- Wang, R. Y. & Strong, D. M. (1996). Beyond accuracy: What data quality means to data consumers. *J. Manag. Inf. Syst.*, 12, 5-33.
- Waskom, M. L. (2021). seaborn: statistical data visualization. *Journal of Open Source Software*, 6 (60), 3021. Zugriff auf <https://doi.org/10.21105/joss.03021> doi: 10.21105/joss.03021
- Weins, C. (2010). Uni-und bivariate deskriptive Statistik. In C. Wolf & H. Best (Hrsg.), *Handbuch der sozialwissenschaftlichen Datenanalyse* (S. 65–89). Wiesbaden: VS Verlag für Sozialwissenschaften. Zugriff auf [https://doi.org/10.1007/978-3-531-92038-2\\_4](https://doi.org/10.1007/978-3-531-92038-2_4) doi: 10.1007/978-3-531-92038-2\_4

## **A. Anhang**

### **A.2. Grundlagen**

#### **Datenbasis / Beschreibung der Daten**

userid	total_number_courses
144	2
...	...
130	3
...	...
42	4
...	...
47	5
...	...
95	6
...	...
63	7
...	...
67	8
...	...
48	9
...	...
81	10
...	...
111	12
...	...
69	16
...	...
16	20
...	...
18	24
...	...
35	28
...	...
114	30
...	...
-3	34
...	...
32	39
26	168
-2	195

Abbildung 62.: Menge der Kurse pro Benutzer

## Datenbasis / Visualisierung der Daten

Die folgenden Listings zeigen u. a. die erforderlichen Anweisungen zur Einrichtung der Arbeitsumgebung oder dem Import der Arbeitsdaten. Bei den Untersuchungen in dieser Arbeit wurden diese stets vorausgesetzt bzw. in besonderen Fällen entsprechend angepasst.

### *Prolog*

---

```

1 from sqlalchemy import create_engine
2 import numpy as np
3 import pandas as pd
4 from matplotlib import pyplot as plt
5 import seaborn as sns
6 from IPython.core.display_functions import display

```

---

Listing 60: Import von Bibliotheken und anderen Erweiterungen

---

```

1 sns.set_theme(style='white', font_scale=1.2, palette='Spectral')

```

---

Listing 61: Definitionen zur Darstellung der Visualisierungen

---

```

1 user = "*****"
2 password = "*****"
3 host = "localhost"
4 database = "vhf_moodle_ws20"
5 port = 3306
6
7 engine = create_engine(f'mysql+pymysql://{{user}}:{password}@{{host}}/{{database}}',
8                         pool_recycle=port)
10 connection = engine.connect()

```

---

Listing 62: Herstellung der Verbindung zur MySQL-Datenbank

---

```

1 query = """SELECT * FROM moodle_data"""
2 # Definition der Arbeitsdaten
3 moodle_data = pd.read_sql(query, connection)

```

---

Listing 63: Import der Arbeitsdaten aus der MySQL-Datenbank

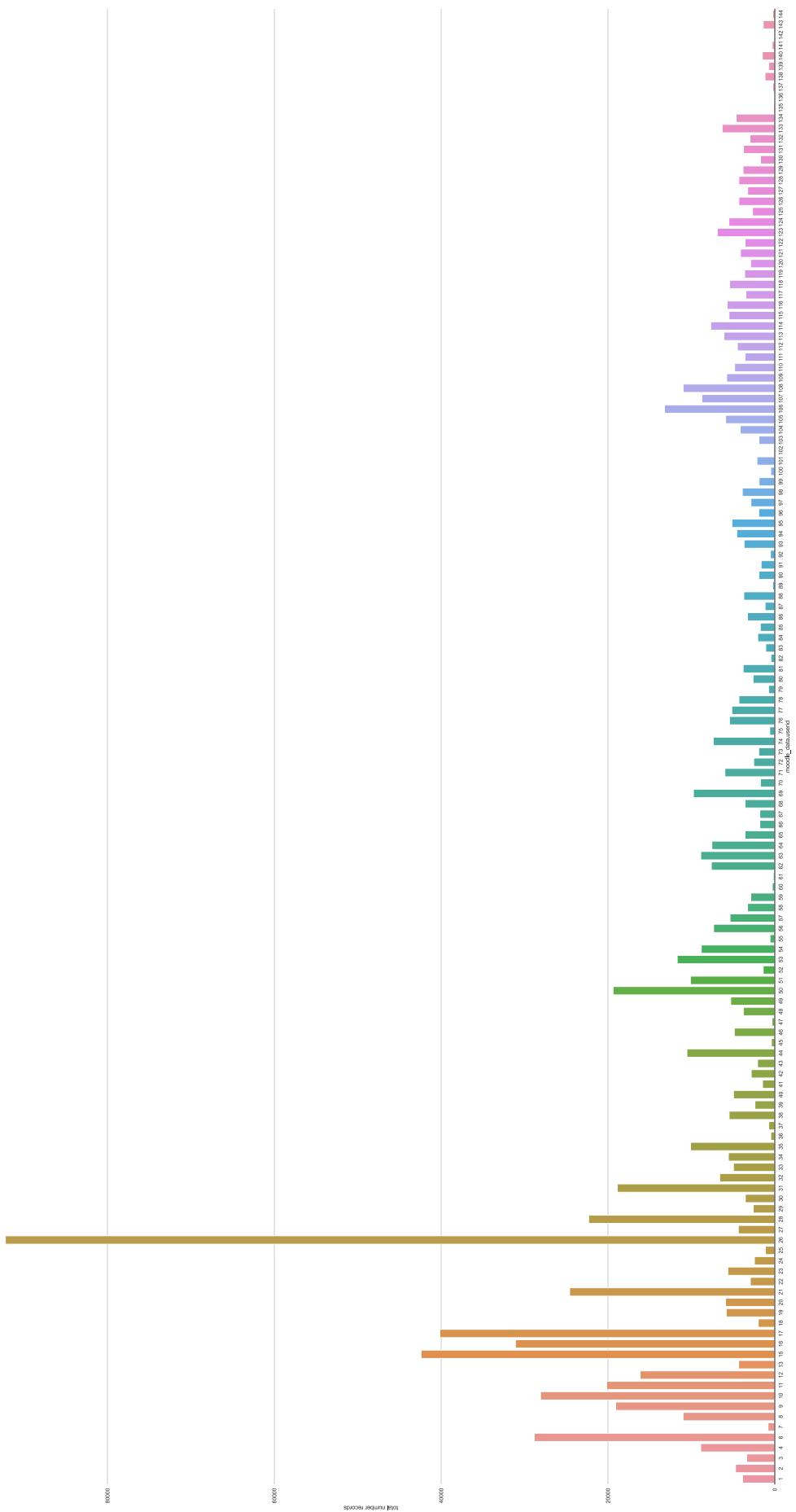
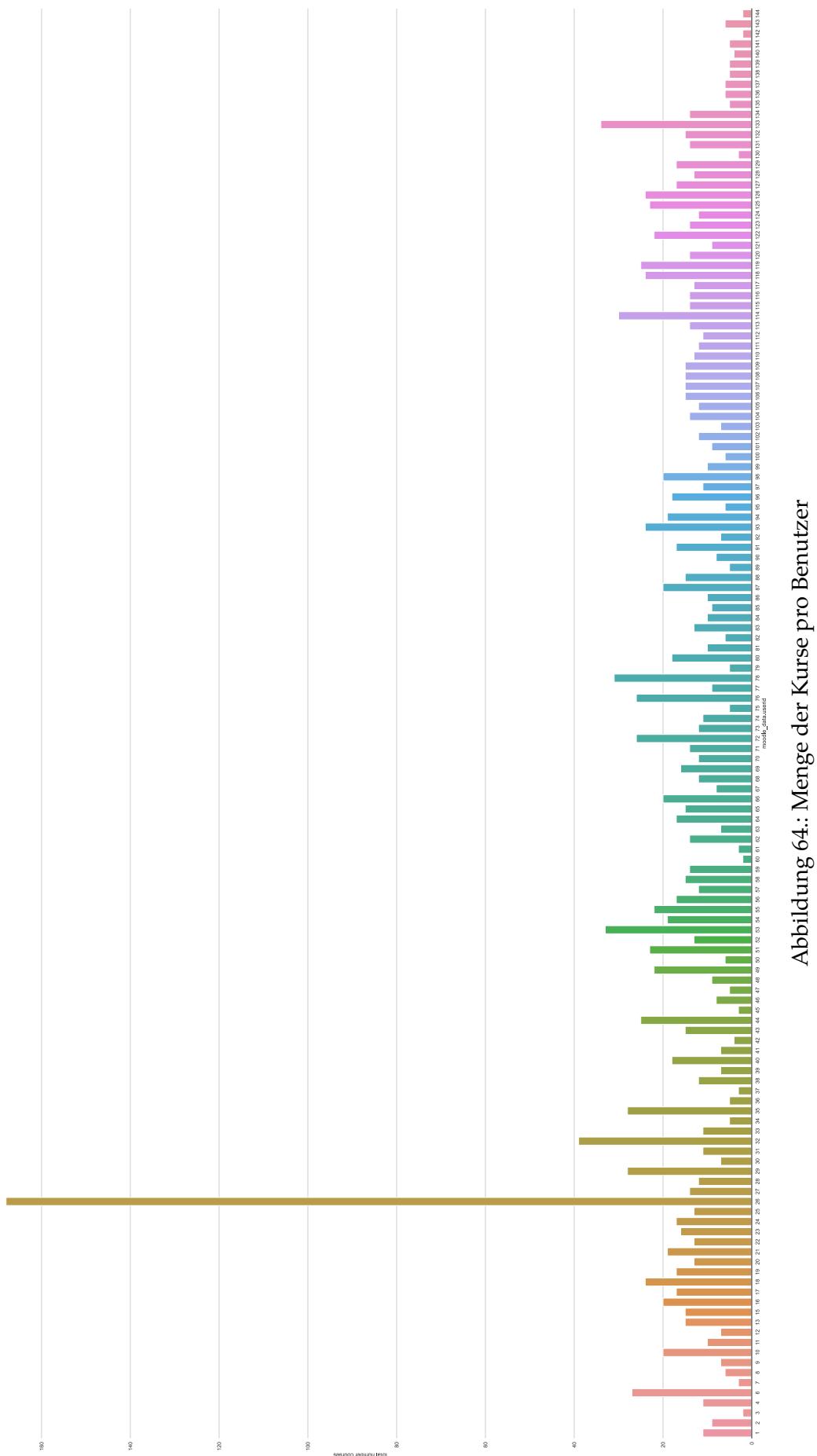


Abbildung 63.: Menge der Log-Einträge pro Benutzer



### **A.3. Analysen**

#### **Identifikation von Studenten**

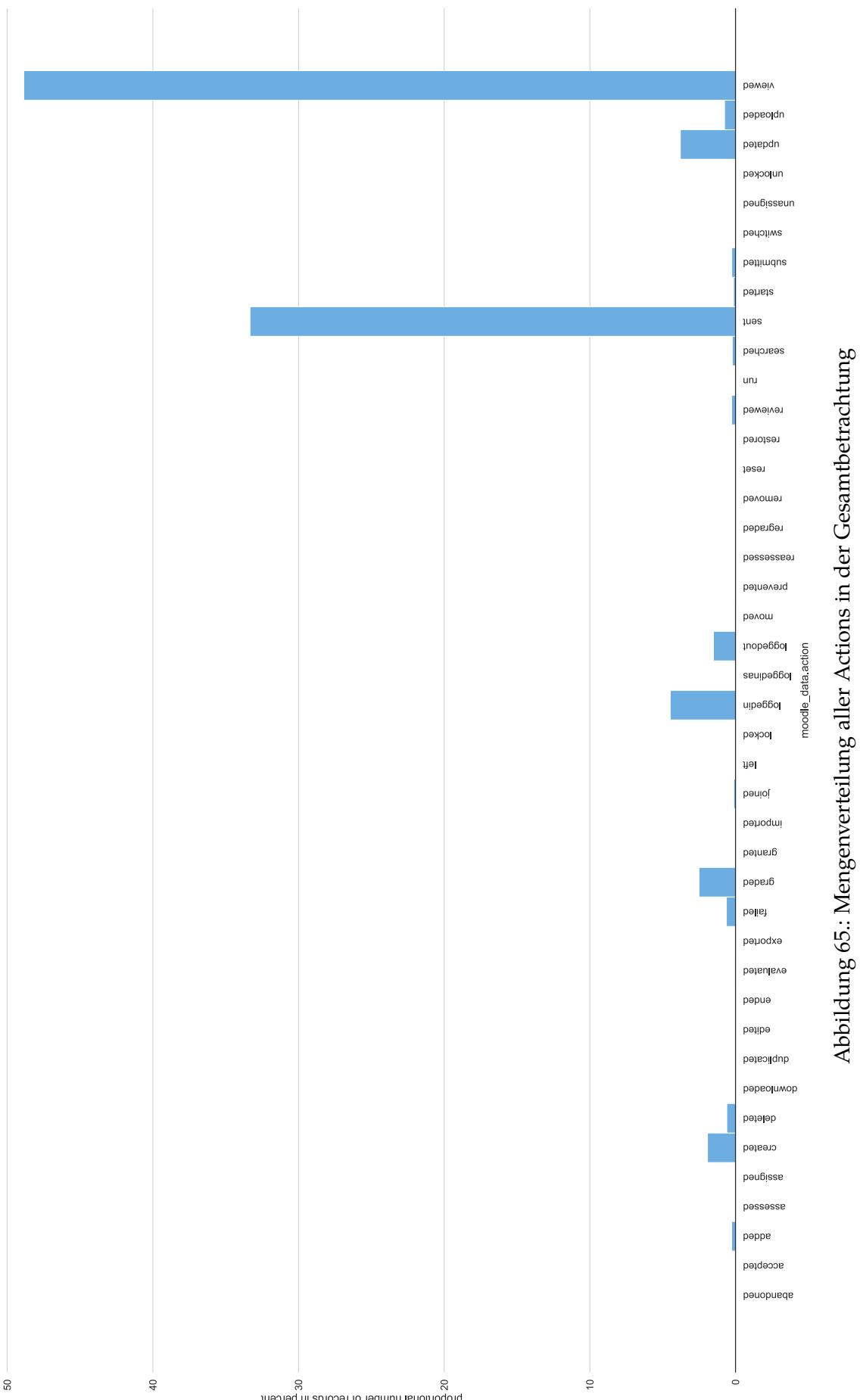
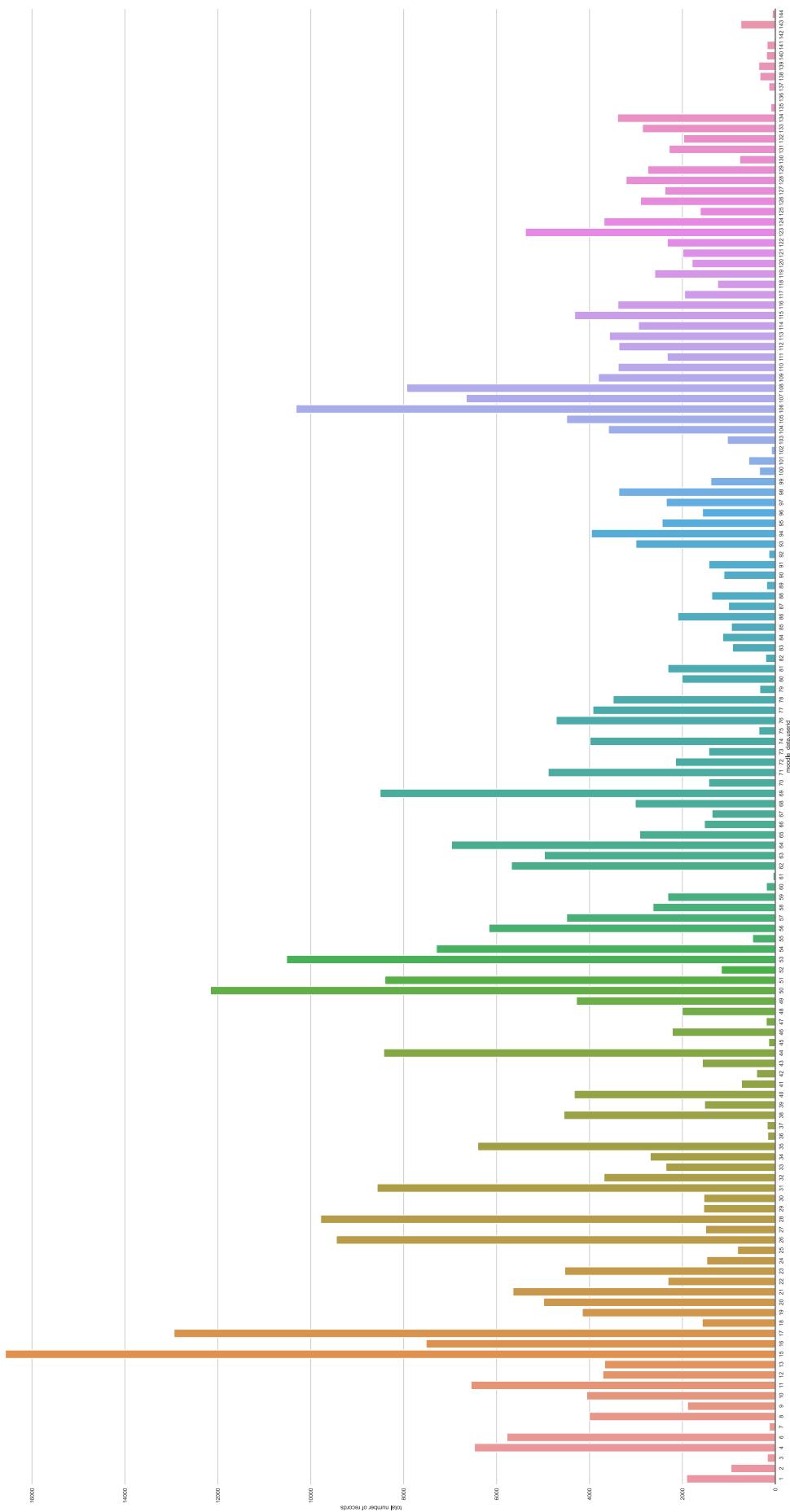


Abbildung 65.: Mengenverteilung aller Actions in der Gesamtbeobachtung



userid	all_actions	viewed_action	other_actions	percentage
64	7544	6970	574	0.9239
53	11699	10520	1179	0.8992
40	4953	4328	625	0.8738
69	9756	8507	1249	0.8720
91	1641	1430	211	0.8714
104	4136	3592	544	0.8685
76	5434	4716	718	0.8679
94	4561	3958	603	0.8678
98	3894	3368	526	0.8649
87	1165	1006	159	0.8635
83	1084	922	162	0.8506
55	575	489	86	0.8504
66	1795	1526	269	0.8501
72	2526	2147	379	0.8500
13	4330	3675	655	0.8487
20	5909	4986	923	0.8438
68	3579	3015	564	0.8424
56	7335	6165	1170	0.8405
57	5361	4491	870	0.8377
52	1390	1162	228	0.8360
38	5478	4551	927	0.8308
51	10118	8404	1714	0.8306
70	1727	1434	293	0.8303
54	8813	7295	1518	0.8278
97	2861	2347	514	0.8203
71	5985	4889	1096	0.8169
65	3576	2918	658	0.8160
93	3685	3000	685	0.8141
96	1928	1566	362	0.8122
78	4300	3490	810	0.8116
49	5286	4280	1006	0.8097
58	3268	2632	636	0.8054
23	5634	4531	1103	0.8042
59	2885	2314	571	0.8021
44	10536	8430	2106	0.8001
...	...	...	...	...

99 rows in set (6,49 sec)

Abbildung 67.: Anteil der viewed-Actions an der Gesamtaktivität

action	students	others
abandoned	0.0	2.0
accepted	28.0	3.0
added	21.0	403.0
created	392.0	2248.0
deleted	46.0	303.0
downloaded	2.0	170.0
duplicated	1.0	0.0
ended	4.0	6.0
evaluated	0.0	348.0
exported	0.0	4.0
graded	106.0	2304.0
granted	0.0	15.0
joined	127.0	26.0
left	15.0	20.0
moved	0.0	2.0
regraded	0.0	3.0
removed	2.0	32.0
restored	0.0	2.0
reviewed	94.0	93.0
searched	4.0	12.0
started	214.0	66.0
submitted	443.0	3.0
switched	0.0	16.0
updated	88.0	5106.0
uploaded	344.0	743.0
viewed	22718.0	26185.0

Abbildung 68.: Kombiniertes Datenset für Studenten und Andere

## A. Anhang

userid	all_actions	added	created	deleted	evaluated	graded	submitted	updated
1	3865	0	43	0	0	0	12	20
13	4330	2	40	2	0	15	51	11
18	1978	2	17	1	0	0	24	14
19	5823	2	77	10	0	24	75	11
20	5909	3	58	3	0	19	55	10
22	2932	1	26	0	0	0	22	5
23	5634	5	76	3	0	35	106	12
24	2444	0	17	36	0	0	13	3
25	1133	6	21	0	0	0	16	2
38	5478	0	46	5	0	13	94	10
40	4953	0	43	9	0	9	44	21
43	2068	1	5	0	0	2	8	4
49	5286	6	51	5	0	57	97	77
51	10118	1	49	2	0	17	58	157
52	1390	2	23	0	0	0	18	13
53	11699	2	35	2	0	10	34	46
54	8813	1	57	16	0	22	63	192
56	7335	3	51	2	0	63	164	71
57	5361	2	74	0	0	26	89	31
58	3268	0	27	0	0	8	27	104
59	2885	1	50	4	0	11	50	18
60	298	0	0	0	0	0	4	0
62	7606	4	61	0	0	21	72	10
64	7544	2	42	0	0	8	35	12
65	3576	1	49	1	0	8	47	13
66	1795	0	25	13	0	1	17	5
67	1788	4	29	2	0	7	29	31
68	3579	2	41	0	0	26	72	6
69	9756	1	63	0	0	16	78	15
70	1727	0	13	6	0	9	18	6
71	5985	1	68	0	0	16	72	21
72	2526	1	5	0	0	3	7	2
73	1929	0	3	0	0	0	3	0
76	5434	1	12	0	0	2	12	0
78	4300	1	35	2	0	3	23	17
80	2611	2	47	0	0	2	24	48
83	1084	1	11	0	0	0	10	1
87	1165	0	6	0	0	1	8	0
91	1641	4	19	0	0	0	23	2
93	3685	7	42	3	0	26	41	102
94	4561	3	27	2	0	49	78	25
96	1928	2	16	0	0	9	24	4
97	2861	0	25	0	0	36	60	89
98	3894	6	37	3	0	12	36	14
99	1883	1	22	0	0	11	26	10
102	112	0	0	0	0	0	1	0
104	4136	1	67	0	0	11	57	20
105	5887	2	85	2	0	16	70	31
107	8751	2	167	33	0	0	11	358
109	5774	4	193	144	0	0	10	387
111	3577	2	141	32	0	1	10	292
112	4486	4	145	31	0	1	11	351
113	6108	2	254	67	0	0	9	377
115	5488	4	166	29	0	1	11	300
116	5717	4	191	16	0	0	10	328
117	3466	3	162	37	0	0	9	329
119	3616	1	87	2	0	0	9	244
120	2902	1	175	10	0	0	5	284
122	3564	1	150	15	0	0	8	288
123	6896	3	134	56	0	0	9	358
124	5505	2	162	22	0	0	9	296
125	2669	1	108	8	0	0	6	219
126	4311	1	171	80	0	0	5	297
127	3250	2	78	17	0	0	5	264
128	4309	3	134	41	0	0	8	311
129	3803	0	114	57	0	0	8	303
131	3748	33	162	17	0	1	7	276
132	2973	2	112	19	0	0	6	279
134	4629	2	146	22	0	0	12	304
136	33	0	0	0	0	0	2	0
142	10	0	0	0	0	0	1	0
143	1387	0	11	0	0	0	4	2

72 rows in set (10,24 sec)

Abbildung 69.: Identifikation von Studenten

**Lokalität des Lern- und Kommunikationsverhaltens**

userid	loggings	total	workingday	weekend
1	1324	100.0	90.785498	9.214502
13	2759	100.0	77.129395	22.870605
18	1128	100.0	83.244681	16.755319
19	3004	100.0	73.901465	26.098535
20	3733	100.0	81.275114	18.724886
22	1686	100.0	84.460261	15.539739
23	3410	100.0	65.131965	34.868035
24	983	100.0	89.318413	10.681587
25	457	100.0	85.339168	14.660832
38	3497	100.0	77.066057	22.933943
40	3026	100.0	85.459352	14.540648
43	957	100.0	84.743992	15.256008
49	3094	100.0	70.620556	29.379444
51	5623	100.0	75.102259	24.897741
52	868	100.0	71.313364	28.686636
53	6462	100.0	75.827917	24.172083
54	4903	100.0	80.950439	19.049561
56	4794	100.0	76.720901	23.279099
57	3106	100.0	61.526079	38.473921
58	1998	100.0	68.718719	31.281281
59	1687	100.0	87.966805	12.033195
60	124	100.0	93.548387	6.451613
62	3659	100.0	79.447937	20.552063
64	5264	100.0	71.599544	28.400456
65	1938	100.0	71.052632	28.947368
66	1014	100.0	75.838264	24.161736
67	964	100.0	86.514523	13.485477
68	2228	100.0	72.935368	27.064632
69	6126	100.0	78.011753	21.988247
70	1043	100.0	78.139981	21.860019
71	3540	100.0	75.536723	24.463277
72	1479	100.0	74.712644	25.287356
73	831	100.0	75.210590	24.789410
76	2556	100.0	84.428795	15.571205
78	2518	100.0	68.189039	31.810961
80	1333	100.0	47.411853	52.588147
83	696	100.0	71.551724	28.448276
87	656	100.0	74.542683	25.457317
91	985	100.0	58.274112	41.725888
93	2013	100.0	79.334327	20.665673
94	3045	100.0	74.187192	25.812808
96	1001	100.0	81.418581	18.581419
97	1735	100.0	88.242075	11.757925
98	2445	100.0	83.967280	16.032720
99	1089	100.0	68.411387	31.588613
102	60	100.0	91.666667	8.333333
104	2686	100.0	72.784810	27.215190
105	3020	100.0	79.867550	20.132450
107	5562	100.0	84.843581	15.156419
109	2987	100.0	78.875126	21.124874
111	1866	100.0	75.455520	24.544480
112	2731	100.0	82.973270	17.026730
113	2939	100.0	59.067710	40.932290
115	3854	100.0	68.655942	31.344058
116	2604	100.0	82.104455	17.895545
117	1537	100.0	58.815875	41.184125
119	2173	100.0	51.449609	48.550391
120	1634	100.0	80.722154	19.277846
122	1863	100.0	66.774020	33.225980
123	4064	100.0	71.432087	28.567913
124	2746	100.0	74.399126	25.600874
125	1232	100.0	75.324675	24.675325
126	2305	100.0	75.271150	24.728850
127	1821	100.0	66.776496	33.223504
128	2698	100.0	74.796145	25.203855
129	2353	100.0	61.963451	38.036549
131	1904	100.0	76.575630	23.424370
132	1555	100.0	83.665595	16.334405
134	3063	100.0	70.421156	29.578844
136	18	100.0	61.111111	38.888889
142	3	100.0	0.000000	100.000000
143	538	100.0	79.925651	20.074349

Abbildung 70.: Erstellung des neuen Datensets *loggings\_daytype*

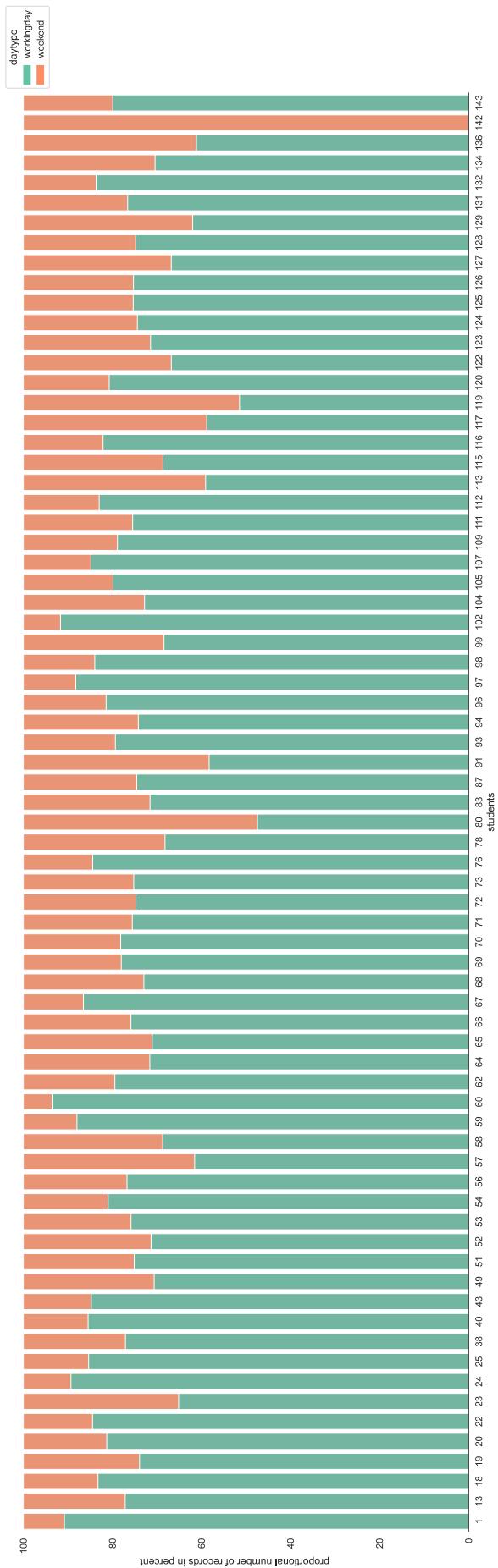
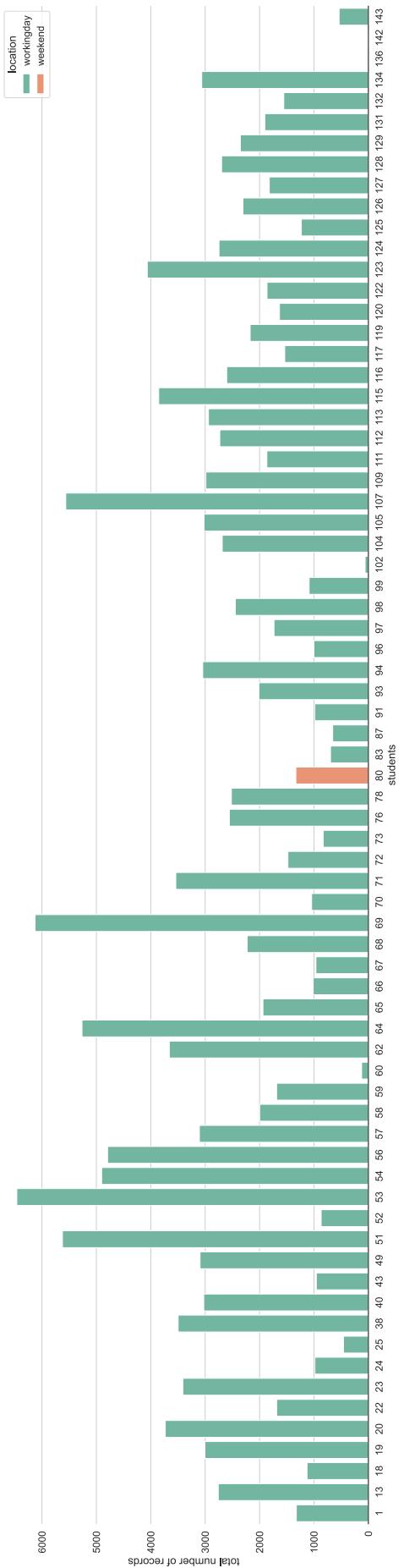
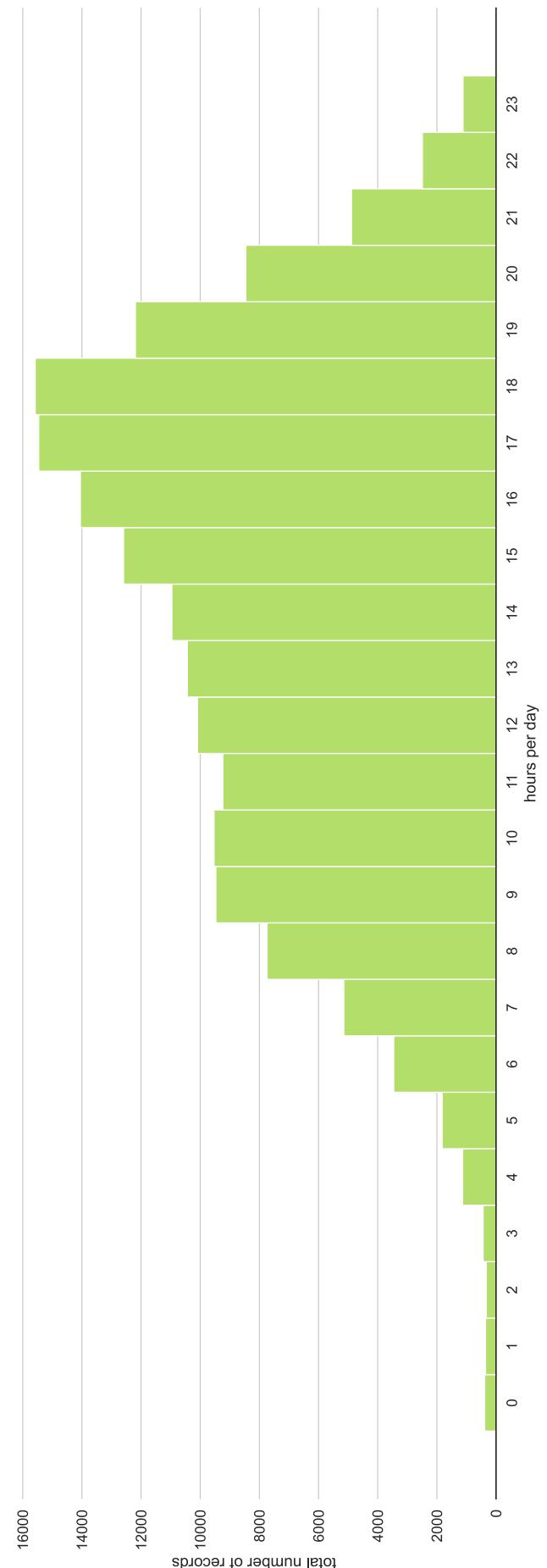


Abbildung 71.: Anteilige Mengen an Log-Einträgen pro Student und Tagestyp





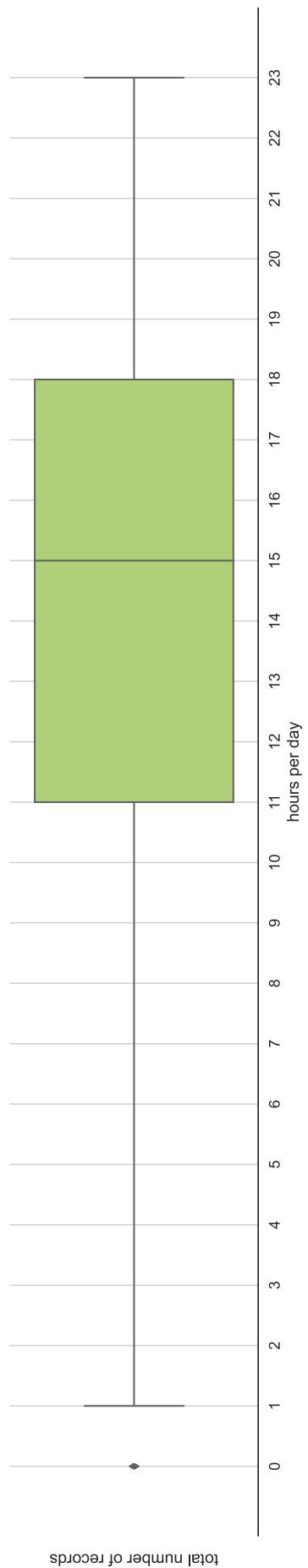


Abbildung 74.: Verteilung der Log-Einträge über die Tagesstunden

userid	loggings	total	workingtime	freetime
1	1324	100.0	61.706949	38.293051
13	2759	100.0	68.901776	31.098224
18	1128	100.0	72.340426	27.659574
19	3004	100.0	48.934754	51.065246
20	3733	100.0	81.837664	18.162336
22	1686	100.0	68.446026	31.553974
23	3410	100.0	70.058651	29.941349
24	983	100.0	39.369278	60.630722
25	457	100.0	49.671772	50.328228
38	3497	100.0	68.229911	31.770089
40	3026	100.0	56.113681	43.886319
43	957	100.0	73.563218	26.436782
49	3094	100.0	69.553975	30.446025
51	5623	100.0	46.416504	53.583496
52	868	100.0	78.571429	21.428571
53	6462	100.0	63.819251	36.180749
54	4903	100.0	55.170304	44.829696
56	4794	100.0	61.869003	38.130997
57	3106	100.0	37.282679	62.717321
58	1998	100.0	52.202202	47.797798
59	1687	100.0	69.531713	30.468287
60	124	100.0	54.838710	45.161290
62	3659	100.0	78.819350	21.180650
64	5264	100.0	70.231763	29.768237
65	1938	100.0	57.223942	42.776058
66	1014	100.0	55.719921	44.280079
67	964	100.0	56.224066	43.775934
68	2228	100.0	62.432675	37.567325
69	6126	100.0	62.961149	37.038851
70	1043	100.0	57.909875	42.090125
71	3540	100.0	55.451977	44.548023
72	1479	100.0	52.738337	47.261663
73	831	100.0	60.770156	39.229844
76	2556	100.0	62.558685	37.441315
78	2518	100.0	52.819698	47.180302
80	1333	100.0	56.339085	43.660915
83	696	100.0	58.189655	41.810345
87	656	100.0	74.847561	25.152439
91	985	100.0	68.832487	31.167513
93	2013	100.0	54.644809	45.355191
94	3045	100.0	83.251232	16.748768
96	1001	100.0	42.957043	57.042957
97	1735	100.0	37.233429	62.766571
98	2445	100.0	52.229039	47.770961
99	1089	100.0	62.075298	37.924702
102	60	100.0	36.666667	63.333333
104	2686	100.0	56.515264	43.484736
105	3020	100.0	49.437086	50.562914
107	5562	100.0	59.600863	40.399137
109	2987	100.0	52.628055	47.371945
111	1866	100.0	66.130761	33.869239
112	2731	100.0	72.244599	27.755401
113	2939	100.0	35.794488	64.205512
115	3854	100.0	55.838090	44.161910
116	2604	100.0	73.233487	26.766513
117	1537	100.0	44.632401	55.367599
119	2173	100.0	79.199264	20.800736
120	1634	100.0	72.154223	27.845777
122	1863	100.0	62.962963	37.037037
123	4064	100.0	52.042323	47.957677
124	2746	100.0	57.465404	42.534596
125	1232	100.0	68.912338	31.087662
126	2305	100.0	72.190889	27.809111
127	1821	100.0	60.351455	39.648545
128	2698	100.0	56.375093	43.624907
129	2353	100.0	69.783255	30.216745
131	1904	100.0	48.792017	51.207983
132	1555	100.0	68.745981	31.254019
134	3063	100.0	71.237349	28.762651
136	18	100.0	61.111111	38.888889
142	3	100.0	33.333333	66.666667
143	538	100.0	51.672862	48.327138

Abbildung 75.: Erstellung des neuen Datensets *loggings\_daytime\_1*

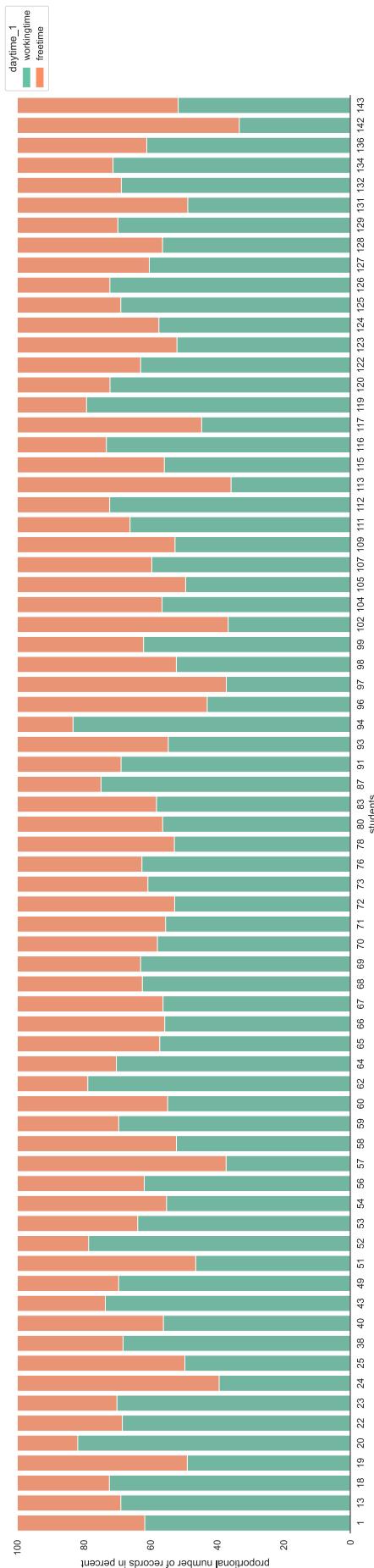


Abbildung 76.: Anteilige Mengen an Log-Einträgen pro Tageszeit

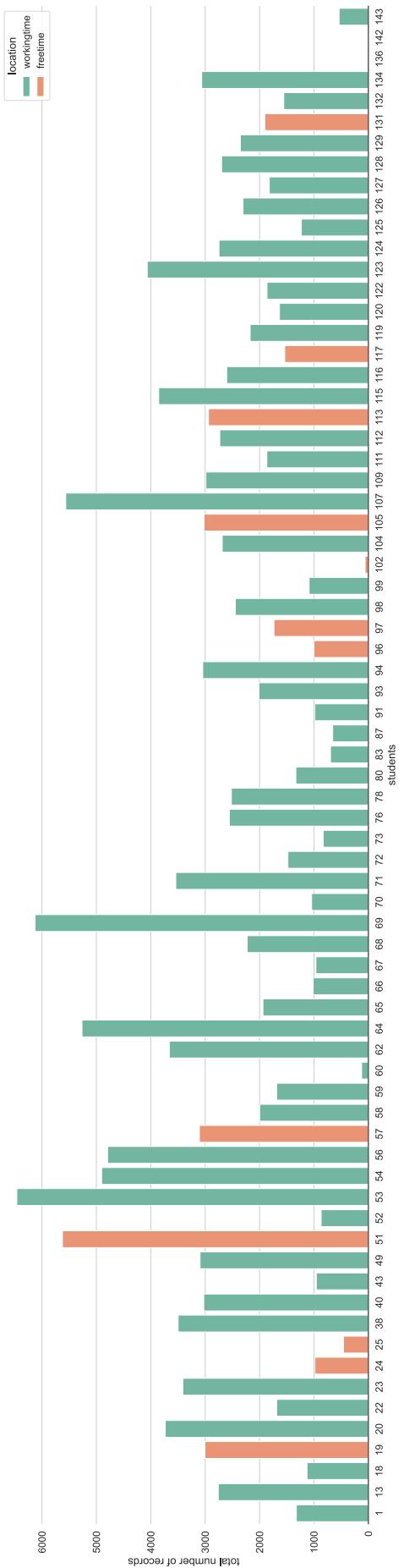


Abbildung 77.: Typisierung nach Tageszeit

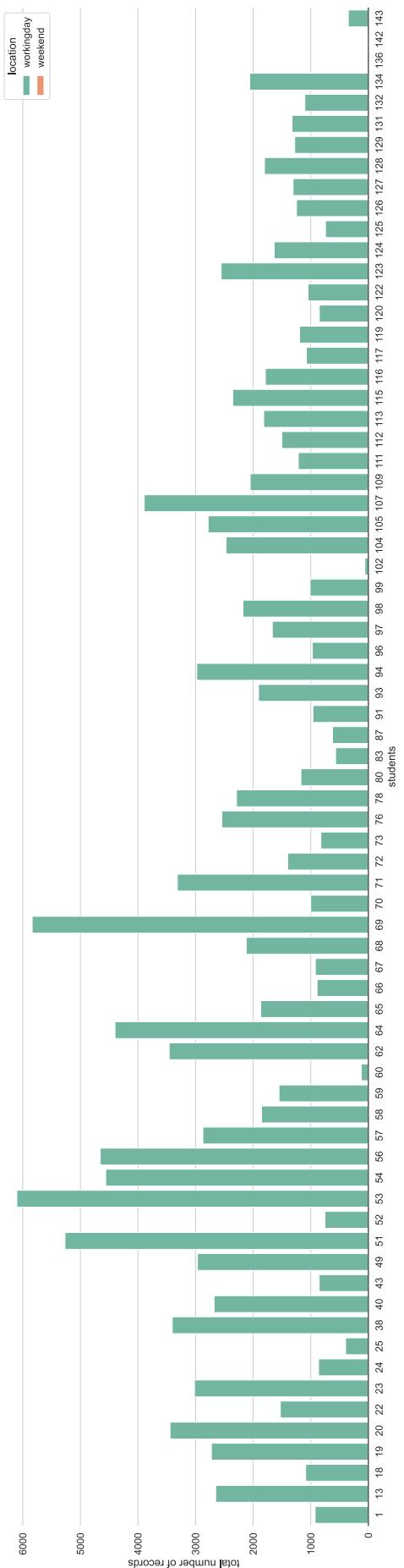


Abbildung 78.: Typisierung nach Tagestyp und Lernverhalten

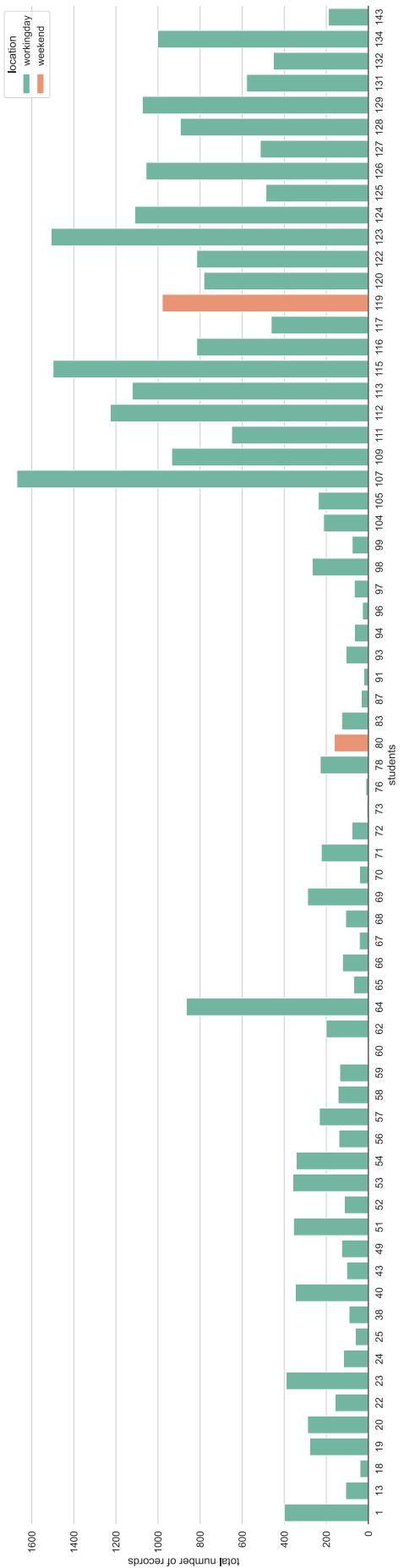


Abbildung 79.: Typisierung nach Tagestyp und Kommunikationsverhalten

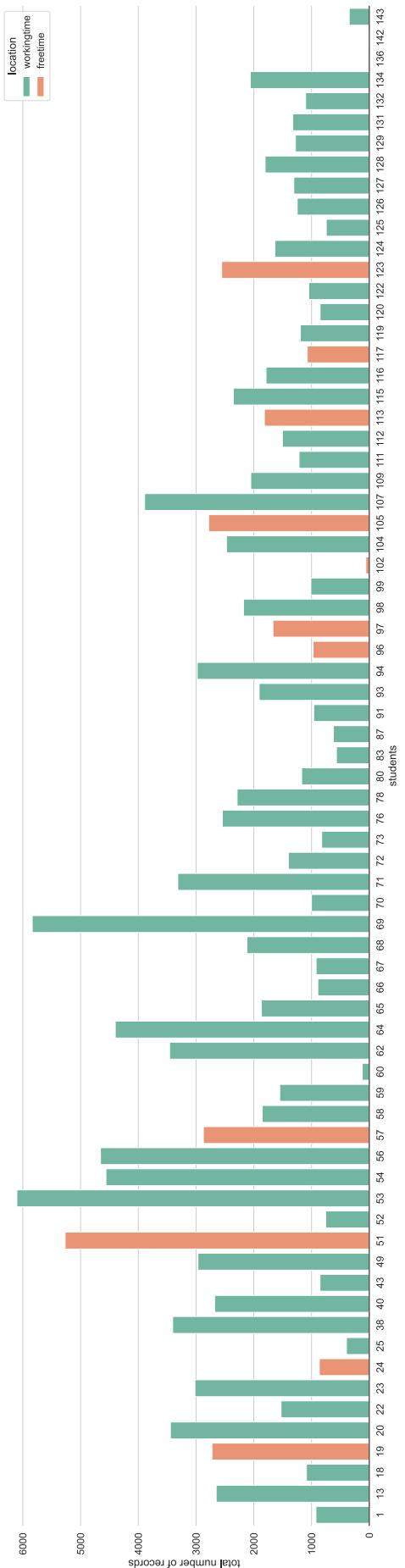


Abbildung 80.: Typisierung nach Tageszeit und Lernverhalten

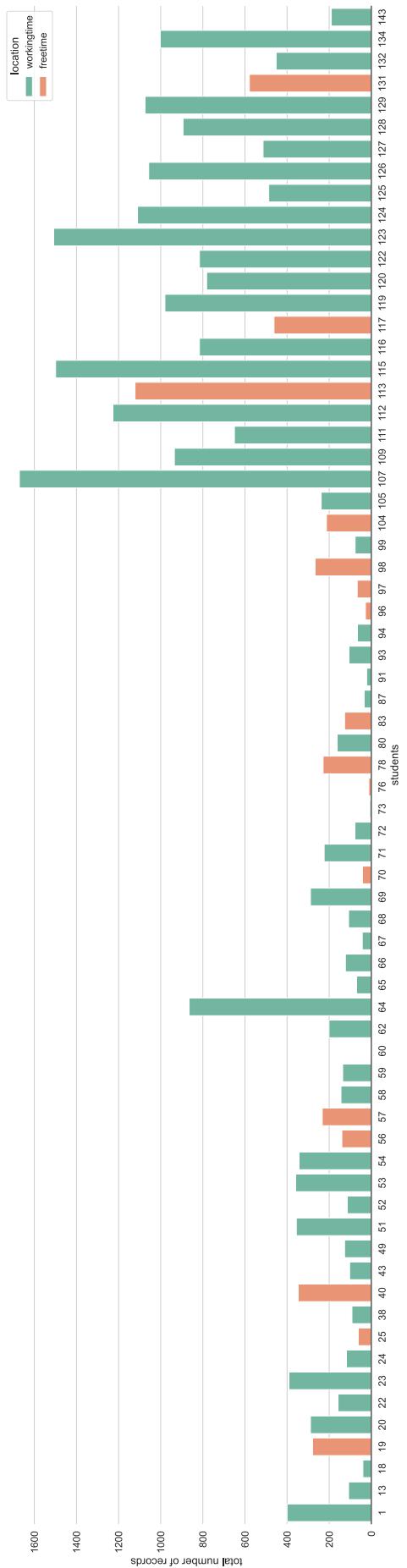


Abbildung 81.: Typisierung nach Tageszeit und Kommunikationsverhalten

**Kontinuität des Lern- und Kommunikationsverhaltens**

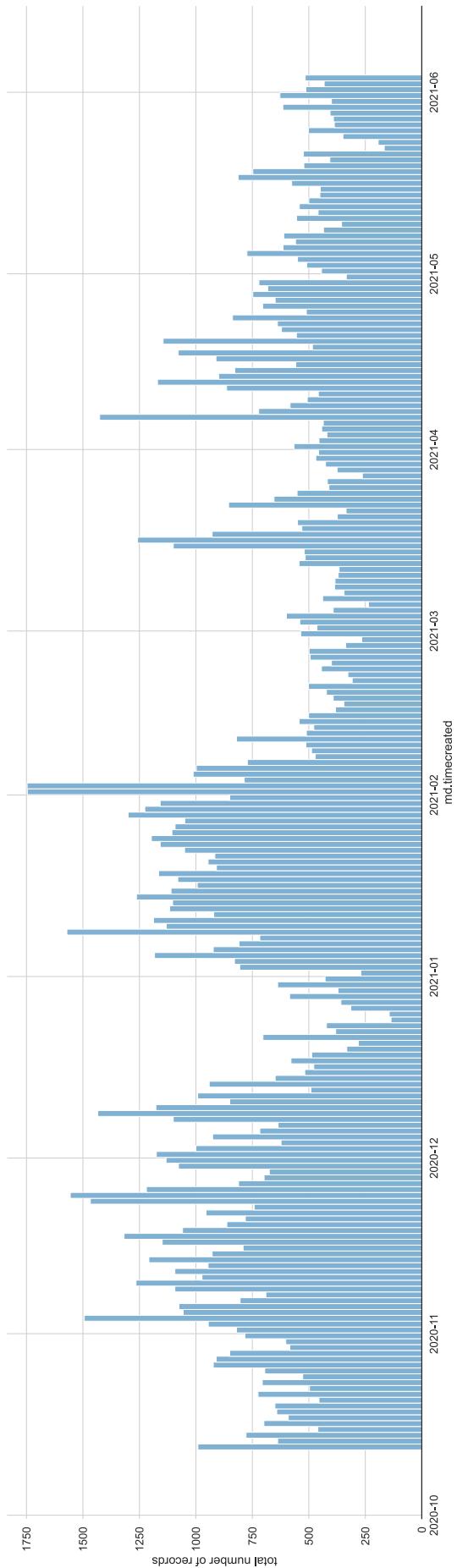


Abbildung 82.: Verteilung der Log-Einträge im Gesamtzeitraum pro Tag

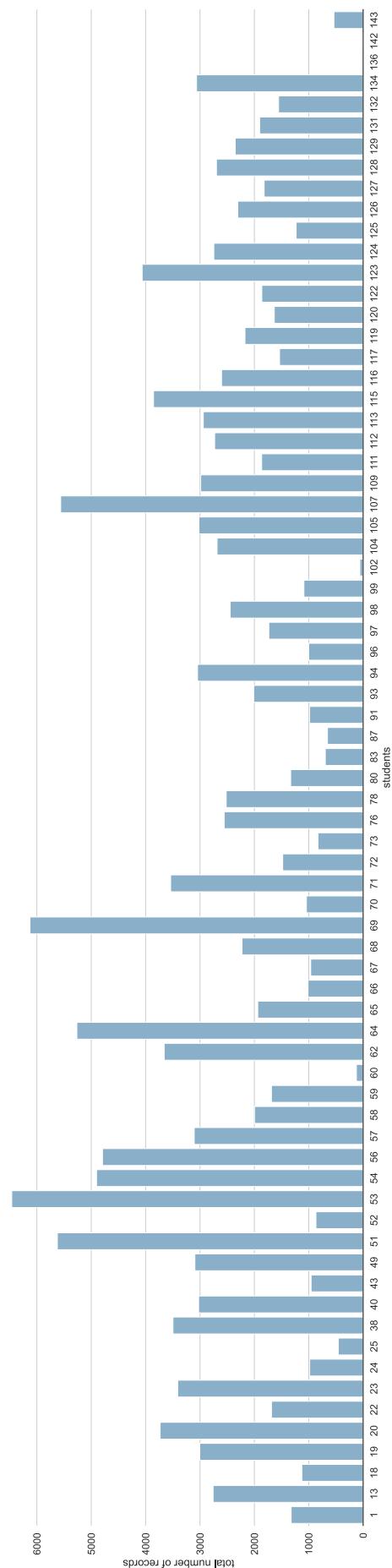


Abbildung 83.: Menge der Log-Einträge im Gesamtzeitraum pro Student

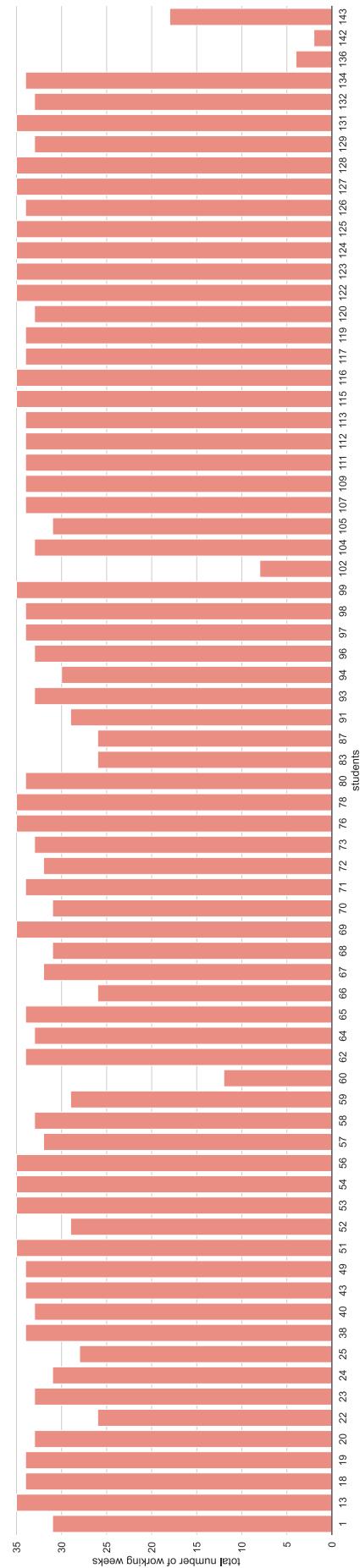


Abbildung 84.: Menge der Arbeitswochen im Gesamtzeitraum pro Student

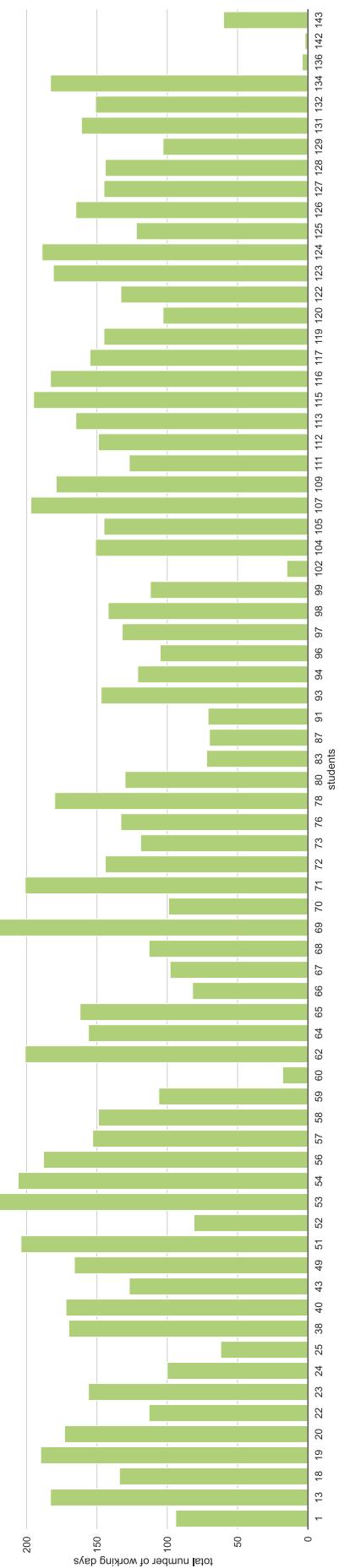


Abbildung 85.: Menge der Arbeitstage im Gesamtzeitraum pro Student

userid	loggings	days		weeks		per_week		avg_count_per_week		upper_count_per_week		lower_count_per_week		avg_count_per_day		upper_count_per_day		lower_count_per_day		upper_count_per_day			
		days	days	weeks	weeks	days	days	days	days	days	days	days	days	days	days	days	days	days	days	days	days		
0	1	134	31	94	21	42	21	183	78	39	158	118	15	118	15	154	276	15	21	22	21		
1	18	1128	35	134	33	16	49	190	88	44	132	132	8	49	8	210	21	8	12	12	12		
2	19	3004	34	190	88	44	132	173	113	56	169	169	15	7	7	205	21	7	23	23	23		
3	20	3733	33	173	113	56	142	113	64	32	97	97	14	10	10	136	12	10	32	32	32		
4	22	1666	26	127	48	142	42	113	64	51	155	155	21	7	7	136	12	7	22	22	22		
5	24	3410	31	100	31	15	47	156	103	51	155	155	47	9	9	144	14	9	14	14	14		
6	23	983	31	100	31	15	47	156	103	51	155	155	47	7	7	144	14	7	10	10	10		
7	24	457	25	45	28	16	8	457	45	16	8	457	47	7	7	205	20	10	30	30	30		
8	9	38	34	45	28	16	8	3497	34	170	102	51	137	17	17	137	17	8	26	26	26		
10	11	40	43	43	33	172	91	172	127	28	14	42	25	7	3	137	12	3	11	11	11		
11	12	49	49	49	34	166	91	166	166	80	145	145	20	10	10	136	13	6	20	20	20		
12	13	5623	51	51	29	14	5623	51	204	160	80	140	140	10	5	136	13	6	23	23	23		
14	14	52	53	53	29	14	5623	51	204	160	80	140	140	10	5	136	13	6	23	23	23		
15	15	53	53	53	29	14	5623	51	204	160	80	140	140	10	5	136	13	6	23	23	23		
16	16	54	4903	35	206	140	70	140	188	82	136	136	8	58	58	210	23	11	35	35	35		
17	17	56	3106	35	32	153	97	153	153	97	48	48	145	45	45	205	25	12	18	18	18		
18	18	57	1998	33	149	60	30	149	149	60	30	30	90	90	145	145	12	4	4	14	14		
19	19	58	1687	29	106	58	10	1687	29	106	58	10	58	29	87	87	15	15	15	41	41		
21	20	59	124	12	18	10	5	124	12	18	10	5	107	53	157	157	13	10	10	27	27		
22	22	60	62	62	34	156	99	221	201	156	159	159	159	159	157	157	13	18	18	50	50		
23	23	64	5264	33	156	156	79	5264	33	156	156	156	156	156	156	156	156	15	15	15	26	26	
24	24	65	1938	34	162	162	57	1938	34	162	162	162	162	162	162	162	162	15	15	15	18	18	
25	25	66	1014	34	144	144	46	1014	34	144	144	144	144	144	144	144	144	14	14	14	14	14	
26	26	67	964	32	126	82	46	964	32	126	82	82	82	82	82	82	82	12	12	12	14	14	
27	27	68	2228	31	113	98	30	2228	31	113	98	98	98	30	30	30	109	109	19	19	19	29	29
28	28	69	6116	35	223	175	175	6116	35	223	175	175	175	175	175	175	175	107	107	107	13	13	
29	29	70	1033	31	99	104	104	1033	31	99	104	104	104	104	104	104	104	50	50	50	15	15	
30	30	71	3510	34	201	144	144	3510	34	201	144	144	144	144	144	144	144	52	52	52	15	15	
31	31	72	831	32	119	119	23	831	32	119	119	119	119	119	119	119	119	69	69	69	10	10	
32	32	73	2556	33	133	133	36	2556	33	133	133	133	133	133	133	133	133	37	37	37	10	10	
33	33	76	2558	34	130	130	39	2558	34	130	130	130	130	130	130	130	130	50	50	50	20	20	
34	34	78	1333	34	134	134	39	1333	34	134	134	134	134	134	134	134	134	50	50	50	13	13	
35	35	80	6956	34	26	72	72	6956	34	26	72	72	72	72	72	72	72	12	12	12	14	14	
36	36	83	6956	34	26	72	72	6956	34	26	72	72	72	72	72	72	72	40	40	40	14	14	
37	37	87	985	29	71	71	33	985	29	71	71	71	71	71	71	71	33	33	33	14	14		
38	38	91	2013	33	147	147	61	2013	33	147	147	147	147	147	147	147	147	61	61	61	20	20	
39	39	93	1005	30	121	101	30	1005	30	121	101	101	101	30	30	30	152	152	152	15	15		
40	40	94	1001	33	132	132	51	1001	33	132	132	132	132	51	51	51	152	152	152	15	15		
41	41	96	1735	34	134	142	34	1735	34	134	142	142	142	34	34	34	152	152	152	15	15		
42	42	97	2445	34	142	142	34	2445	34	142	142	142	142	34	34	34	107	107	107	17	17		
43	43	98	1083	35	112	112	35	1083	35	112	112	112	112	35	35	35	107	107	107	17	17		
44	44	99	60	102	102	8	60	102	102	8	151	151	151	151	151	151	151	46	46	46	4	4	
45	45	1004	46	105	105	33	2686	33	105	105	105	105	33	33	33	40	40	40	146	146	146		
46	46	105	3020	31	145	145	81	3020	31	145	145	145	145	81	81	81	48	48	48	122	122		
47	47	105	5562	34	145	145	81	5562	34	145	145	145	145	81	81	81	48	48	48	146	146		
48	48	107	2983	34	179	179	87	2983	34	179	179	179	179	87	87	87	43	43	43	131	131		
49	49	109	1866	34	127	127	54	1866	34	127	127	127	127	54	54	54	27	27	27	82	82		
50	50	111	2731	34	149	149	80	2731	34	149	149	149	149	80	80	80	40	40	40	120	120		
51	51	112	2939	34	165	165	86	2939	34	165	165	165	165	86	86	86	43	43	43	129	129		
52	52	113	3895	34	195	195	81	3895	34	195	195	195	195	81	81	81	55	55	55	165	165		
53	53	115	2604	35	183	183	78	2604	35	183	183	183	183	78	78	78	37	37	37	117	117		
54	54	116	1537	34	145	145	63	1537	34	145	145	145	145	63	63	63	22	22	22	71	71		
55	55	117	2173	34	145	145	63	2173	34	145	145	145	145	63	63	63	31	31	31	95	95		
56	56	119	1634	33	120	120	53	1634	33	120	120	120	120	53	53	53	24	24	24	74	74		
57	57	120	1823	33	133	133	53	1823	33	133	133	133	133	53	53	53	24	24	24	74	74		
58	58	122	4043	35	181	181	58	4043	35	181	181	181	181	58	58	58	26	26	26	129	129		
59	59	123	2746	35	189	189	78	2746	35	189	189	189	189	78	78	78	39	39	39	174	174		
60	60	125	1202	35	122	122	35	1202	35	122	122	122	122	35	35	35	17	17	17	117	117		
61	61	125	2305	34	165	165	67	2305	34	165	165	165	165	67	67	67	33	33	33	101	101		
62	62	126	1216	34	145	145	63	1216	34	145	145	145	145	63	63	63	26	26	26	78	78		
63	63	127	1821	35	144	144	77</td																

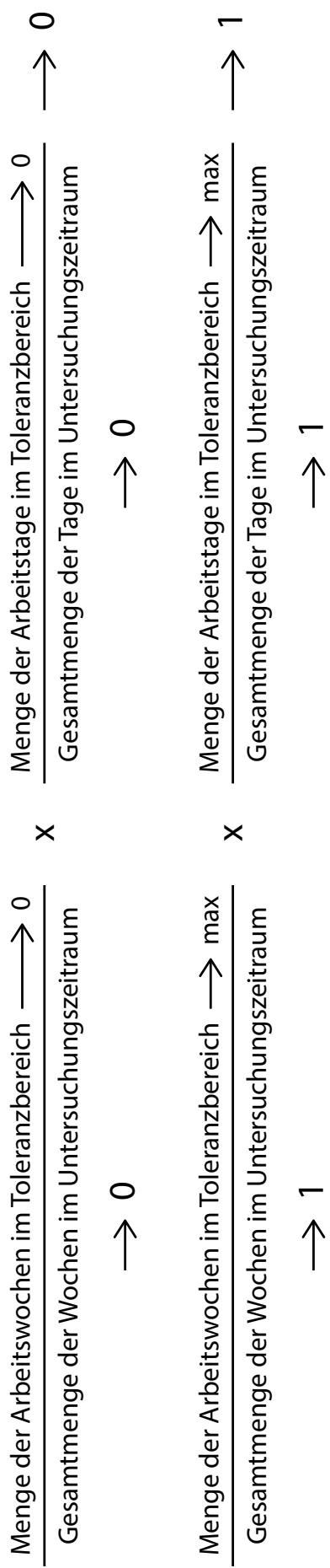


Abbildung 87.: Individueller Kontinuitätskoeffizient, IKK

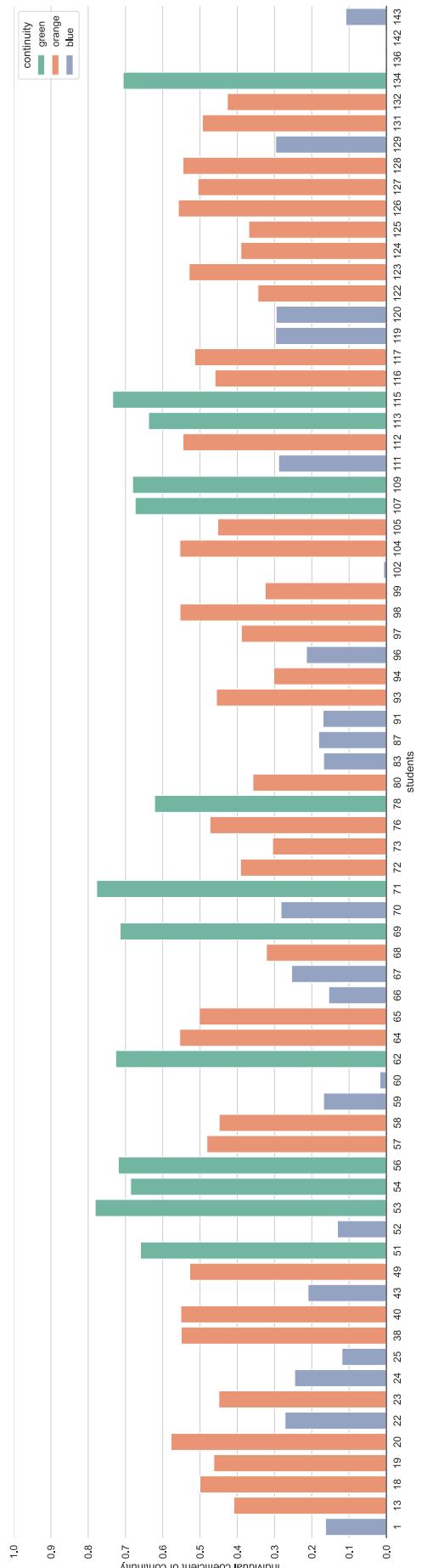


Abbildung 88: Typisierung nach IKK mit Bezug auf IKK

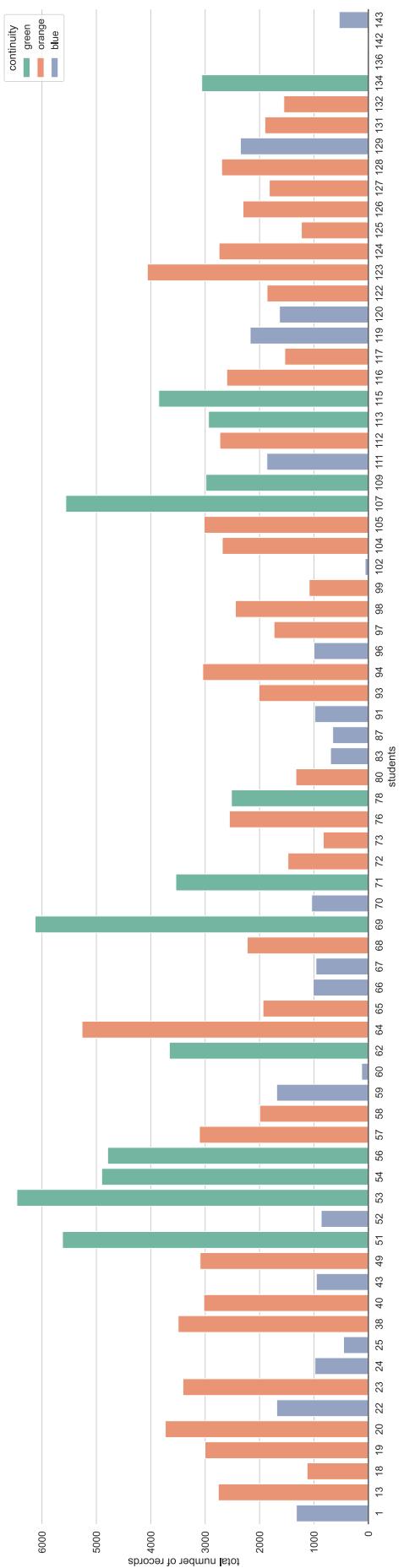


Abbildung 89.: Typisierung nach IKK mit Bezug auf Log-Einträge

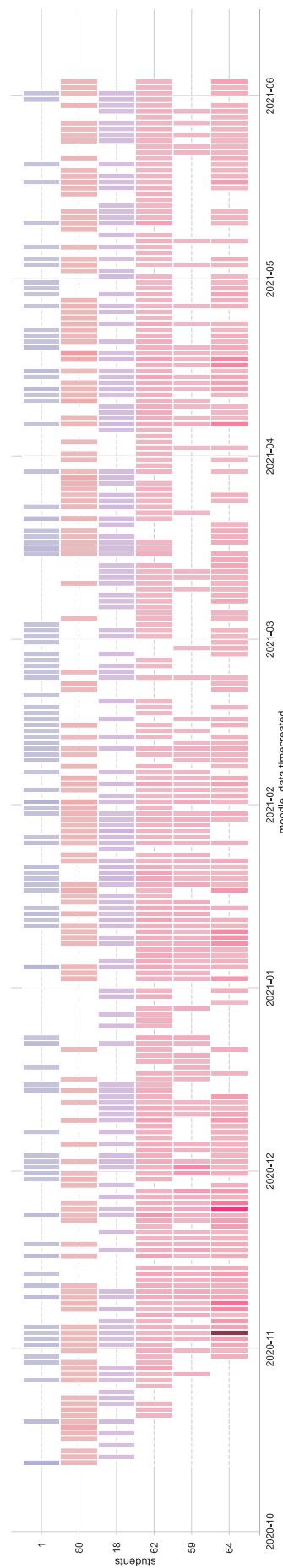


Abbildung 90.: Menge der Log-Einträge für einzelne Studenten

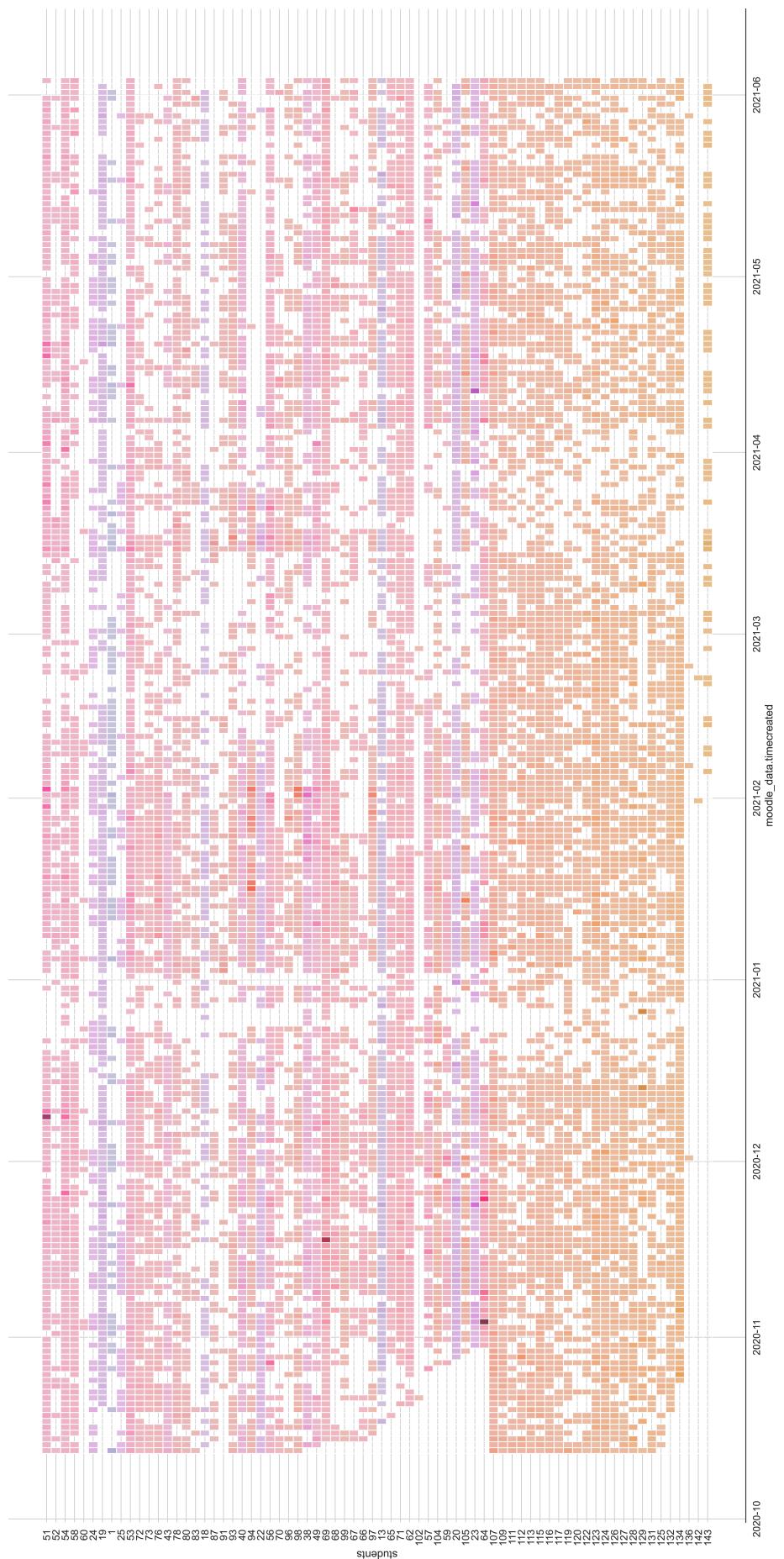


Abbildung 91.: Menge der Log-Einträge pro Student im Gesamtzeitraum nach Tagen

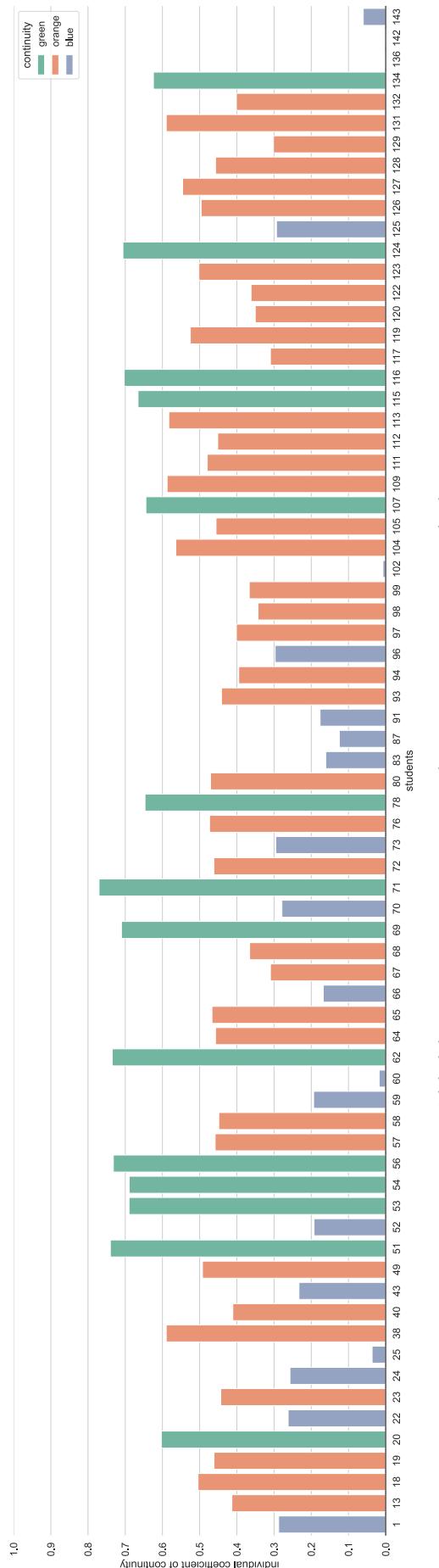


Abbildung 92.: Typisierung nach Kontinuität, Lernverhalten

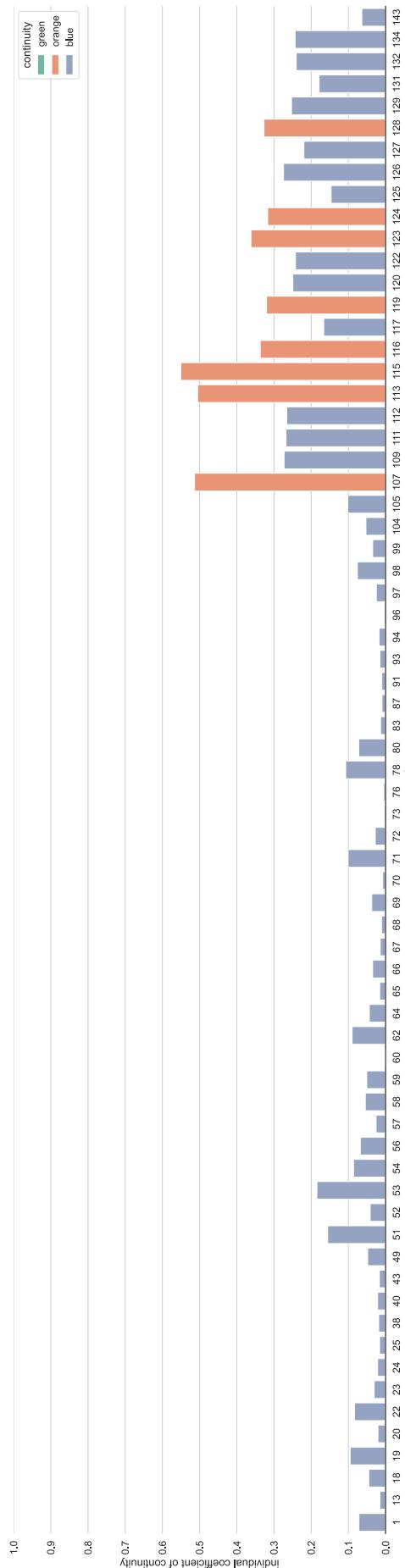


Abbildung 93.: Typisierung nach Kontinuität, Kommunikationsverhalten

**Dynamik des Lern- und Kommunikationsverhaltens**

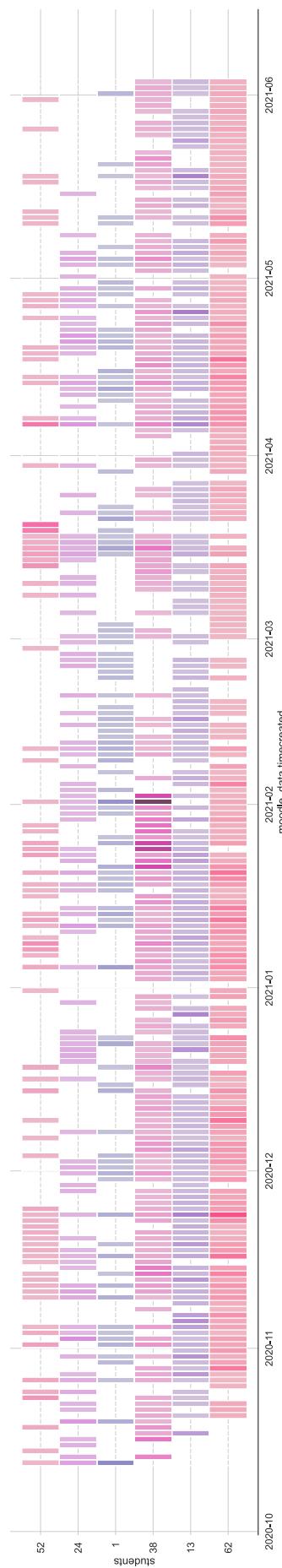


Abbildung 94.: Menge der Log-Einträge für einzelne Studenten

userid	loggings	days	avg_count_per_day
1	1324	94	14
13	2759	183	15
18	1128	134	8
19	3004	190	15
20	3733	173	21
22	1686	113	14
23	3410	156	21
24	983	100	9
25	457	62	7
38	3497	170	20
40	3026	172	17
43	957	127	7
49	3094	166	18
51	5623	204	27
52	868	81	10
53	6462	221	29
54	4903	206	23
56	4794	188	25
57	3106	153	20
58	1998	149	13
59	1687	106	15
60	124	18	6
62	3659	201	18
64	5264	156	33
65	1938	162	11
66	1014	82	12
67	964	98	9
68	2228	113	19
69	6126	223	27
70	1043	99	10
71	3540	201	17
72	1479	144	10
73	831	119	6
76	2556	133	19
78	2518	180	13
80	1333	130	10
83	696	72	9
87	656	70	9
91	985	71	13
93	2013	147	13
94	3045	121	25
96	1001	105	9
97	1735	132	13
98	2445	142	17
99	1089	112	9
102	60	15	4
104	2686	151	17
105	3020	145	20
107	5562	197	28
109	2987	179	16
111	1866	127	14
112	2731	149	18
113	2939	165	17
115	3854	195	19
116	2604	183	14
117	1537	155	9
119	2173	145	14
120	1634	103	15
122	1863	133	14
123	4064	181	22
124	2746	189	14
125	1232	122	10
126	2305	165	13
127	1821	145	12
128	2698	144	18
129	2353	103	22
131	1904	161	11
132	1555	151	10
134	3063	183	16
136	18	4	4
142	3	2	1
143	538	60	8

Abbildung 95.: Datenset *time\_rel\_dyn* zur Dynamikanalyse

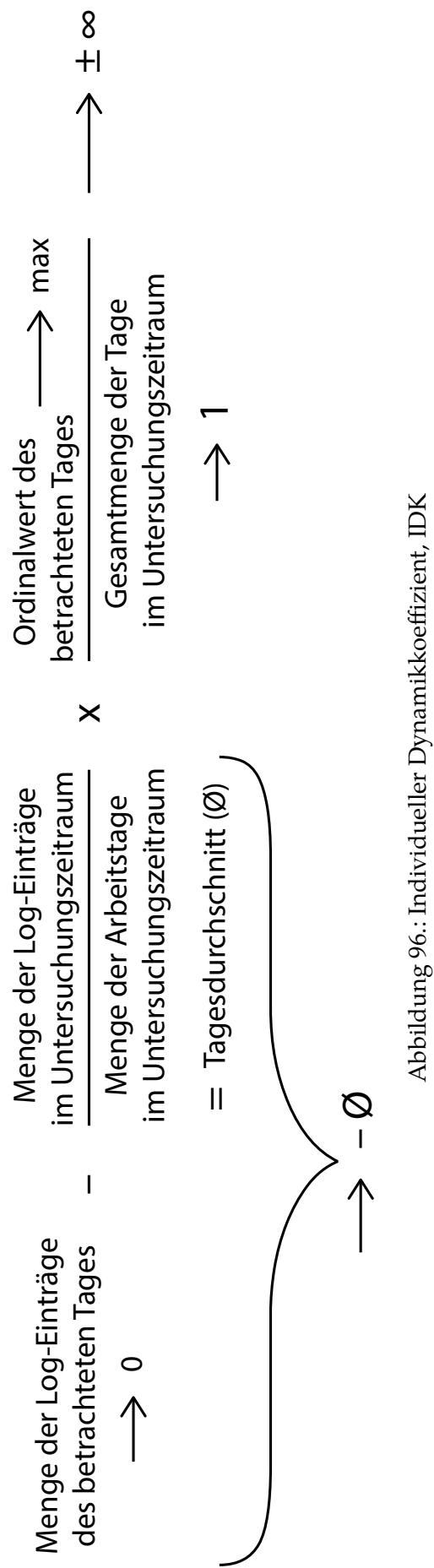


Abbildung 96.: Individueller Dynamikoeffizient, IDK

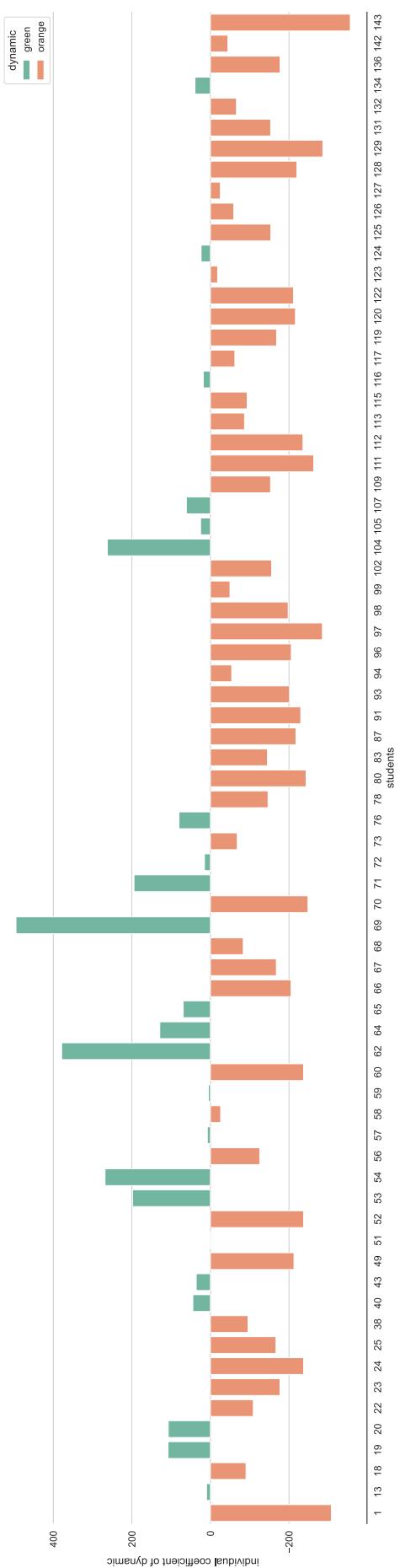


Abbildung 97.: Typisierung der Studenten nach IDK



Abbildung 98.: Menge der Log-Einträge für einzelne Studenten

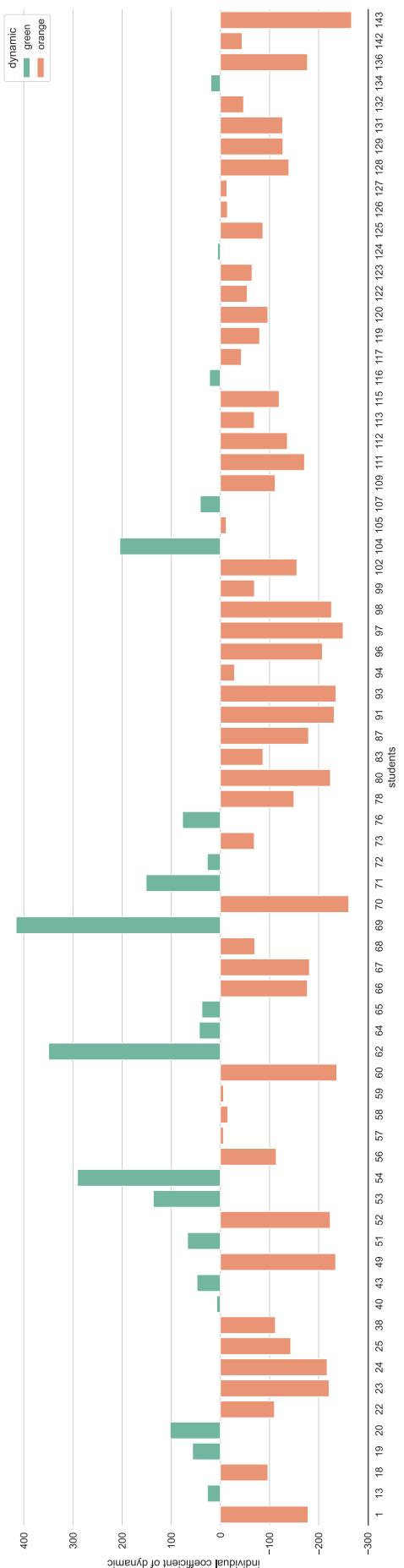


Abbildung 99.: Typisierung nach Dynamik, Lernverhalten

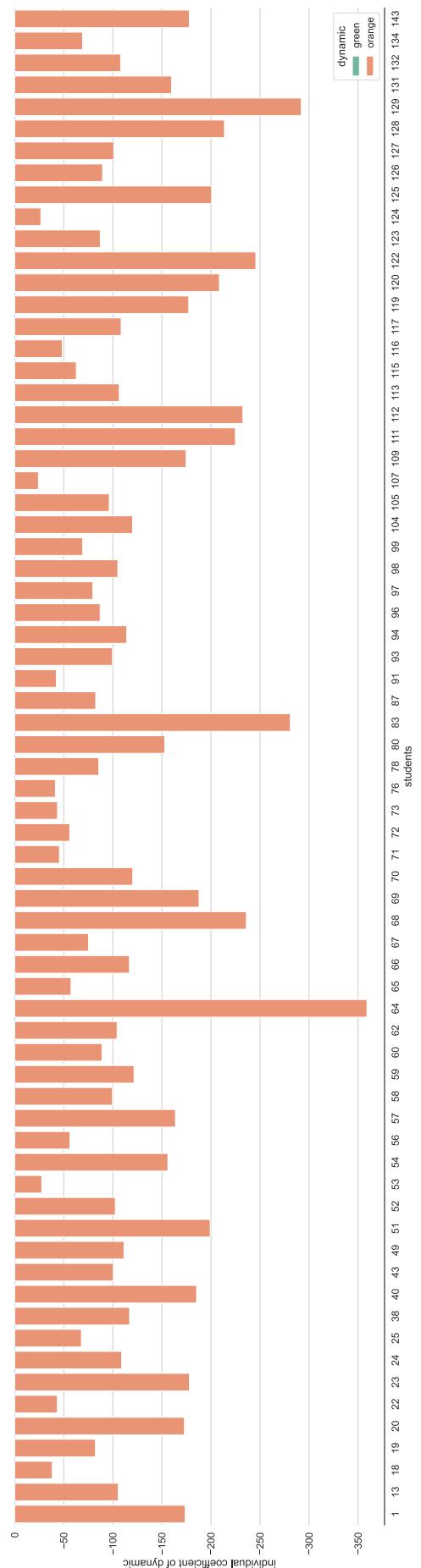


Abbildung 100.: Typisierung nach Dynamik, Kommunikationsverhalten

## **Erklärung zur Urheberschaft**

Ich habe die Arbeit selbständig verfasst, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt, sowie alle Zitate und Übernahmen von fremden Aussagen kenntlich gemacht.

Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt.

Die vorgelegten Druckexemplare und die vorgelegte digitale Version dieser Arbeit sind vollkommen identisch.

Heidelberg, 08.08.2022

---

Unterschrift

## **Inhalt des beigefügten Datenträgers**

Verzeichnis / Beschreibung

---

/1\_ ...

---

/2\_ ...

---

/3\_ ...

---