FH AACHEN
UNIVERSITY OF APPLIED SCIENCES

Fachhochschule Aachen, Campus Jülich

Fachbereich 09 - Medizintechnik und Technomathematik
Studiengang Angewandte Mathematik und Informatik

Seminararbeit

Analyse von Access Point Daten zur Bestimmung der Auslastung von Lernräumen der RWTH

Autor:
Sebastian Menne, 3582710

Betreuer:
Prof. Dr. rer. nat. Alexander Voß
Marc Steffens, M. Eng.

Aachen, 13. Dezember 2024

Eidesstattliche Erklärung

Hiermit versichere ich, dass ich die Seminararbeit mit dem Thema

Analyse von Access Point Daten zur Bestimmung der Auslastung von Lernräumen der RWTH

selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe, alle Ausführungen, die anderen Schriften wörtlich oder sinngemäß entnommen wurden, kenntlich gemacht sind und die Arbeit in gleicher oder ähnlicher Fassung noch nicht Bestandteil einer Studien- oder Prüfungsleistung war.

Ich verpflichte mich, ein Exemplar der Seminararbeit fünf Jahre aufzubewahren und auf Verlangen dem Prüfungsamt des Fachbereiches Medizintechnik und Technomathematik auszuhändigen.

Name: Sebastian Menne

Aachen, den 03.12.24

Unterschrift der Studentin / des Studenten

Sebastian Menne

Zusammenfassung

Um wenig Zeit bei der Suche nach freien Räumen zum Lernen zu verbrauchen, stellt die RWTHapp den Studierenden einen Lernraumfinder zur Verfügung, in dem die Auslastung verschiedener Räume angegeben ist. Die Auslastung wird dabei auf Basis von Informationen von Access Points geschätzt und in einer Art Ampelschema angegeben. Da dieser Ansatz aufgrund von Störfaktoren wie mehreren internetfähigen Geräten pro Studierenden, Einwählen in einen Access Point, der in einem anderen Raum stationiert ist, Vorlesungsbetrieb, etc. fehleranfällig ist, wurde in dieser Arbeit versucht, ein Verfahren zu konzipieren, das auf Basis von Methoden der Felder Data Science und Machine Learning eine möglichst genaue Schätzung erlaubt.

Zunächst wurden die vorliegenden Daten aufbereitet, sodass für jeden Raum zu jedem Zeitpunkt die durchschnittliche Anzahl der Geräte, die mit dem Access Point verbunden waren, vorlag. Mittels eines hierarchischen Clusterings mit Dynamic Time Warping als Metrik konnten die Räume dann in Gruppen von ähnlichen Auslastungsentwicklungen eingeteilt werden. Eine Betrachtung der Silhouettenkoeffizienten ergab, dass sechs Gruppen optimal waren. Lernräume in einer Gruppe konnten daraufhin als äquivalent angenommen werden.

Als nächster Schritt wurde sich mit den Störfaktoren auseinandergesetzt und es konnten vier Gruppen identifiziert werden, bei denen die Anzahl der Geräte pro Access Point deutlich zu hoch war, als dass diese mit der Anzahl der Studierenden in diesen Räumen gleichgesetzt werden konnte. Damit ein Ausgleich geschaffen werden konnte, wurden Räume der vier Gruppen besucht, die tatsächliche Anzahl der Personen im Raum gemessen und mit der gemessenen Anzahl der Geräte in dem Raum verbunden. Über eine logistische Regression und Gradient Descent Verfahren konnte dann für jede Gruppe eine Ausgleichsfunktion berechnet werden. Auch wurden drei Klassen entwickelt, die angeben, wie voll ein Raum ist.

Eine Evaluation ergab, dass die gefundenen Funktionen teilweise sehr präzise und in anderen Fällen ungenau sind.

Inhaltsverzeichnis

1 Einleitung **1**
- 1.1 Motivation . . . 1
- 1.2 Zielsetzung . . . 1
- 1.3 Vorgehen . . . 2

2 Clusteranalyse **3**
- 2.1 Die vorliegenden Daten . . . 3
  - 2.1.1 Datenakquise . . . 3
  - 2.1.2 Datenaufbereitung . . . 4
- 2.2 Grundlagen der Clusteranalyse . . . 5
  - 2.2.1 Dynamic Time Warping . . . 6
  - 2.2.2 Silhouettenkoeffizient . . . 7
  - 2.2.3 Agglomeratives Hierarchisches Clustering . . . 8
- 2.3 Clusterauswahl . . . 8

3 Fehleranalyse **12**
- 3.1 Einfluss der Störfaktoren . . . 12
- 3.2 Klassifikation . . . 13
- 3.3 Ausgleich der Störung . . . 13
  - 3.3.1 Lineare Regression . . . 15
  - 3.3.2 Logistische Regression . . . 17
  - 3.3.3 Leave-One-Out Kreuzvalidierung . . . 19

4 Evaluation **22**
- 4.1 Bewertung . . . 22
- 4.2 Diskussion . . . 23

5 Fazit **24**
- 5.1 Zusammenfassung . . . 24
- 5.2 Ausblick . . . 24

A Literaturverzeichnis **26**

B Tabellenverzeichnis **28**

C Abbildungsverzeichnis **29**

D Abkürzungsverzeichnis **30**

1 Einleitung

1.1 Motivation

Die Klausurenphasen sind für Studierende häufig mit erheblichem Zeitaufwand und Stress verbunden. Für ein effektives Lernen ist ein ablenkungsfreies Umfeld von entscheidender Bedeutung, welches in vielen Fällen durch die eigenen Wohnverhältnisse nicht gewährleistet werden kann. Um diesem Bedürfnis entgegenzukommen, bietet die RWTH Aachen Lernräume für Studierende an. Aufgrund der begrenzten Anzahl an verfügbaren Plätzen und der hohen Nachfrage gestaltet es sich jedoch oft schwierig, ausreichend freie Plätze für Lerngruppen zu finden.

Um diese Herausforderung zu adressieren, stellt die RWTHapp, eine App, die den Studierenden der RWTH eine Vielzahl an Funktionalitäten zu Erleichterung des Hochschulalltags bietet [1], bereits einen Lernraumfinder zur Verfügung, der es ermöglicht, freie Räume zu identifizieren. Die Auslastung wird dabei in einer Art Ampelschema angegeben. Es gibt drei Klassen:

- wenig los
- mäßig viel los
- viel los

Die Einteilung in diese Klassen basiert auf der Anzahl der Geräte, die mit dem Access Point, das ist eine kabellose Schnittstelle, die Endgeräte wie Laptops oder Handys mit einem Datennetz verbindet und so beispielsweise Internetzugang ermöglicht [2], des jeweiligen Raumes verbunden sind. Zudem wird auch die Anzahl an Sitzplätzen miteinbezogen. Diese Methode ist jedoch ungenau, da sie die Auslastung tendenziell zu hoch einschätzt. Beispielsweise kann die Anzahl der verbundenen Geräte nicht mit der Anzahl der anwesenden Personen gleichgesetzt werden. Es ist davon auszugehen, dass Studierende in der Regel mehrere Geräte wie Smartphone, Laptop und Tablet mit sich führen. Zudem besteht die Möglichkeit, dass Geräte außerhalb des Raumes mit dem Access Point des Lernraums verbunden sind, was die Berechnung verfälschen kann. Umgekehrt kann auch der Fall auftreten, dass Geräte innerhalb eines Raumes mit einem Access Point eines anderen Raumes verbunden sind, was ebenfalls berücksichtigt werden muss.

Um solche Störfaktoren zu identifizieren und in die Berechnung einzubeziehen, ist eine entsprechende Analyse und Auswertung der gesammelten Daten erforderlich. Nur durch diese präzise Herangehensweise kann eine zuverlässige Auslastungsbestimmung erfolgen und eine für die Studierenden effektive Lernraumampel entwickelt werden.

1.2 Zielsetzung

Was ist nun eine effektive Lernraumampel? Effektiv meint in diesem Zusammenhang nicht, dass aus den durch Störungen “verrauschten” Daten die wahre Anzahl der Personen im Raum entnommen wird und diese den Nutzern der RWTHapp angezeigt wird. Vielmehr soll eine Klassenstruktur entwickelt werden. Diese könnte die momentane “Ampelstruktur” sein, möglicherweise aber auch eine andere. Um diese Verbesserung vorzunehmen, müssen die noch offenen Fragen

- Wie soll die Klassifizierung geschehen?

- Wie groß ist der Einfluss der Störfaktoren auf eine Messung und kann man diese ausgleichen?
- Verhalten sich alle Räume äquivalent oder muss nach Gruppen unterschieden werden?

geklärt werden, mit dem Ziel, ein Verfahren zu konzipieren, dass für jeden Lernraum auf Basis der Anzahl der verbundenen Geräte eine möglichst genaue Klassifizierung der Raumauslastung vornimmt.

1.3 Vorgehen

Da nicht jedes Gebäude und jeder Raum für die Messung mithilfe der Access Points geeignet ist, wird in der RWTHapp nur für bestimmte, und nach Schätzungen die meistgenutzten, Lernräume die Auslastung angezeigt. Eine Auflistung dieser Räume kann in Tabelle 1.1 gefunden werden. Im Zuge dieser Seminararbeit werden deswegen auch nur die genannten Räume betrachtet und nicht alle.

Raum	Öffnungszeiten	Lernplätze	Access-Point
SE 001	Mo - Fr: 20:00 Uhr - 24:00 Uhr	30	ap-1580-001
SE 002	Mo - Fr: 20:00 Uhr - 24:00 Uhr	18	ap-1580-002
SE 003	Mo - Fr: 20:00 Uhr - 24:00 Uhr	18	ap-1580-003
SE 101	Mo - Fr: 20:00 Uhr - 24:00 Uhr	30	ap-1580-101
SE 102	Mo - Fr: 20:00 Uhr - 24:00 Uhr	18	ap-1580-102
SE 103	Mo - Fr: 20:00 Uhr - 24:00 Uhr	18	ap-1580-103
SE 108	Mo - Fr: 20:00 Uhr - 24:00 Uhr	30	ap-1580-108
SE 209	Mo - Fr: 20:00 Uhr - 24:00 Uhr	30	ap-1580-209
B 002	Mo - Fr: 08:00 Uhr - 17:00 Uhr	90	ap-3011-002
B 061	Mo - Fr: 08:00 Uhr - 17:30 Uhr	8	ap-3011-061
Großer Lernraum Audimax	Mo - Fr: 07:30 Uhr - 21:00 Uhr	144	ap-1420-u215
Kleiner Lernraum Audimax	Mo - Fr: 07:30 Uhr - 21:00 Uhr	44	ap-1420-u132
Foyer Audimax	Mo - Fr: 07:30 Uhr - 21:00 Uhr	88	ap-1420-302
Flur Zwischengeschoss Audimax	Mo - Fr: 07:30 Uhr - 21:00 Uhr	64	ap-1420-101o ap-1420-102w ap-1420-001w p-1420-002o
514 Geo	Mo - Fr: 07:00 Uhr - 20:00 Uhr	24	ap-1140-504
SG 202	In der vorlesungsfreien Zeit gemäß Aushang	24	ap-1810-202 ap-1810-303
SG 422	In der vorlesungsfreien Zeit gemäß Aushang	48	ap-1810-422
Flur Anglistik	Mo - Fr: 07:00 Uhr - 20:30 Uhr	24	ap-1070-114
PPS Foyer im Erdgeschoss	Mo - Fr: 07:00 Uhr - 20:00 Uhr	48	ap-2315-eg_foyerh ap-2315-eg_foyerv ap-2315-hs_2vr
Foyer AH VI	Mo - Fr: 08:30 Uhr - 17:30 Uhr	13	ap-2356-5057
Garderobe AH V	Mo - Fr: 08:30 Uhr - 17:30 Uhr	4	ap-2356-5058
Bibliothek 4006	Mo - Do: 09:00 Uhr - 19:00 Uhr Fr: 09:00 Uhr - 14:00 Uhr	12	ap-2353-4006
Foyer Informatik	Mo - Fr: 08:30 Uhr - 17:30 Uhr	44	ap-2352-006 ap-2352-011

Tabelle 1.1: Liste der Lernräume, deren Auslastung in der RWTHapp angezeigt wird [3]

Kapitel 2 beschäftigt sich zuerst mit der Frage: *"Verhalten sich alle Räume äquivalent oder muss nach Gruppen unterschieden werden?"*.

Kapitel 3 beantwortet auf Grundlage der Erkenntnisse aus dem vorherigen Kapitel die Frage: *"Wie groß ist der Einfluss der Störfaktoren auf eine Messung und kann man diese ausgleichen?"* und *"Wie soll die Klassifizierung geschehen?"*.

In Kapitel 4 wird abschließend eine Bewertung des erarbeiteten Modells durchgeführt.

2 Clusteranalyse

In diesem Kapitel wird sich mit der Gruppierung von äquivalenten Lernräumen auseinandergesetzt. Räume sind hierbei äquivalent, wenn diese ähnlich besucht werden, also zu gleichen Zeiten gleich gefüllt sind. Mithilfe von Visualisierungen wird das Verständnis der textuellen Beschreibung der zugrundeliegenden Datenanalyse unterstützt.

2.1 Die vorliegenden Daten

Um eine Gruppierung vorzunehmen, bedarf es eines Datensatzes, der regelmäßige Messungen der Anzahl der verbundenen Geräte eines jeden Raumes enthält. Ohne diesen kann das allgemeine zeitliche Verhalten der Lernräume nicht analysiert und ausgewertet werden. Jedoch existiert ein solcher Datensatz noch nicht, weshalb eine Akquise während dieser Seminararbeit stattfand.

Nachfolgend wird erläutert, wie die Daten gesammelt und aufbereitet wurden.

2.1.1 Datenakquise

Die RWTHapp nutzt in der aktuellen Anzeige der Raumauslastung einen Endpunkt, der auf eine HTTP-Anfrage die Anzahl der aktuell verbundenen Geräte für jeden Access-Point als JSON zurückgibt. Dabei werden das 2.4 GHz-Band und 5 GHz-Band separat gemessen. Das JSON sieht wie folgt aus:

```
[
    {
        "name": "ap-2500-haus4-bar",
        "user_2_4": 1,
        "user_5": 0
    },
    {
        "name": "ap-6401-0001_1",
        "user_2_4": 0,
        "user_5": 2
    },
    {
        "name": "ap-5385-001",
        "user_2_4": 0,
        "user_5": 0
    },
    ...
]
```

Durch diesen Endpunkt konnten Daten regelmäßig gesammelt werden. Dazu wurde ein Python-Skript geschrieben, dass alle zehn Minuten eine Anfrage an den Endpunkt sendet, die Antwort mithilfe der Abbildungstabelle 1.1 nach den gesuchten Räumen filtert und die Daten mit Zeitstempel

in einer CSV-Datei abspeichert. Eine Konsequenz davon, dass der Datensatz im Zuge dieser Arbeit erstellt wurde, ist, dass keine Daten während einer Klausurenphase, also genau der Zeit, in der die Lernräume am meisten verwendet werden, vorliegen.

Ein Ausschnitt des Datensatzes ist in Tabelle 2.1 zu sehen. Dabei handelt es sich um eine Zusammenstellung mehrerer sogenannter multivariater Zeitreihen. Das sind Ansammlungen von reellwertigen Datenvektoren, in diesem Fall die beiden Frequenzbänder, welche durch einen Zeitstempel in einer temporalen Reihenfolge geordnet werden können [4, 1.1. Time-series clustering] [5]. Jeder Raum bildet dabei eine eigene Zeitreihe.

time	name	2.4 GHz	5 GHz
2024-10-07 15:00:00	SG 202 - 1	2	8
2024-10-07 15:00:00	SG 422	3	4
2024-10-07 15:00:00	514 Geo - 2	1	5
2024-10-07 15:00:00	PPS Foyer im Erdgeschoss - 2	1	11
2024-10-07 15:00:00	PPS Foyer im Erdgeschoss - 1	0	7
2024-10-07 15:00:00	PPS Foyer im Erdgeschoss - 3	0	1
2024-10-07 15:00:00	Foyer Audimax - 4	5	67
2024-10-07 15:00:00	Großer Lernraum Audimax	3	47
2024-10-07 15:00:00	SG 202 - 2	2	31
2024-10-07 15:00:00	Foyer Informatik - 2	2	19
2024-10-07 15:00:00	Foyer Informatik - 1	1	2
2024-10-07 15:00:00	Kleiner Lernraum Audimax	3	11
2024-10-07 15:00:00	SE 001	1	9
2024-10-07 15:00:00	SE 003	0	8
2024-10-07 15:00:00	SE 101	0	2
2024-10-07 15:00:00	SE 102	0	2
2024-10-07 15:00:00	SE 103	0	1
2024-10-07 15:00:00	SE 108	2	2
2024-10-07 15:00:00	SE 209	0	0
2024-10-07 15:00:00	SE 002	0	14
2024-10-07 15:00:00	Foyer AH VI	0	0
...	...	...	...

Tabelle 2.1: Ausschnitt aus den Daten der verbundenen Geräte

2.1.2 Datenaufbereitung

Für die Analyse der Daten und damit diese aufbereitet werden können, wird die erstellte CSV-Datei in Python importiert und mithilfe der Bibliothek *pandas* [6] verfügbar gemacht. *pandas* ist ein flexibles und häufig genutztes Werkzeug in der Datenanalyse [7] und erlaubt einfache Datenmanipulation und -visualisierung.

Der Liste der Lernräume (s. Tabelle 1.1) kann man entnehmen, dass manche Räume von mehreren Access Points abgedeckt werden, wie beispielsweise das *PPS Foyer im Erdgeschoss*. Der erste Schritt der Datenaufbereitung ist es, die Messung aller dieser Access Points für jeden Zeitpunkt zu summieren. Des Weiteren muss beachtet werden, dass die Daten ununterbrochen akquiriert werden, also auch zu Zeiten, wenn verschiedene Lernräume nicht geöffnet haben. Eine Auflistung der Öffnungszeiten ist in Tabelle 1.1 zu finden. Der nächste Schritt ist, den Datensatz nach Öffnungszeiten zu filtern, damit Statistiken wie Mittelwert und Standardabweichung nicht verfälscht werden. Das hat als Nebeneffekt, dass die Räume *SG 202* und *SG 422* aus der Analyse dieser Seminararbeit herausgenommen werden müssen, da sie nur während der vorlesungsfreien Zeit als Lernraum zu Verfügung stehen und sonst anders genutzt werden. Zudem muss beachtet werden,

dass die Räume verschiedene Kapazitäten haben (s. Tabelle 1.1) und dadurch zu einem Zeitpunkt unterschiedlich hohe Messungen vorliegen können, die Räume nicht aber zwingend unterschiedlich stark gefüllt sein müssen. Damit ein Vergleich der verschiedenen Zeitreihen sinnvoll getätigt werden kann, wird deswegen die Auslastung als

$$\text{Auslastung} = \frac{\text{Gemessene Anzahl von Personen}}{\text{Kapazität des Raums}} \tag{2.1}$$

definiert und mit dieser gearbeitet. Da wie bereits erwähnt von dem Endpunkt die Anzahl der Geräte pro Frequenzband zurückgegeben wird, sollten diese noch zusammengefasst werden. Für eine spätere ausführliche Betrachtung könnten die Bänder separat betrachtet werden, um Informationen über die Verbindung zu erhalten, bei der dem aktuellen Vorhaben ist aber nur die Gesamtanzahl relevant. Für die Analyse ist es zudem sinnvoll, die Daten eines jeden Raums auf einen "ideellen" Tag zu mitteln, weil die Messungen zu einem willkürlichen Zeitpunkt gestartet wurden und nicht jeder Raum eine volle Woche geöffnet ist. Der "ideelle" Tag eines Lernraums hat an jedem gemessenen Zeitpunkt, der in der Öffnungszeit liegt, den durchschnittlichen Wert der Auslastung aller Messungen zu diesem Zeitpunkt. Abbildung 2.1 zeigt die durchschnittliche Auslastung aller Räume über den Tag verteilt.

Alle Graphen wurden mit der Python Bibliothek *Seaborn* [8] erstellt.

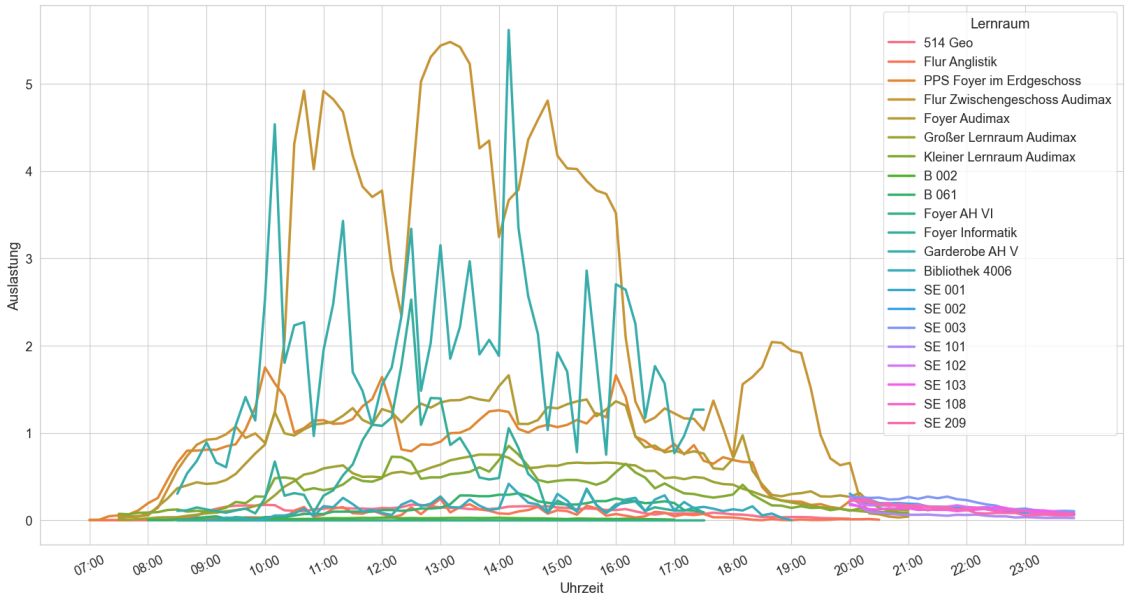

Abbildung 2.1: Auslastung der Lernräume

2.2 Grundlagen der Clusteranalyse

Das Finden äquivalenter Räume kann mit der Aufgabe des Clustering gleichgesetzt werden. Clustering ist eine Technik, bei der Daten mit großer Ähnlichkeit einer gleichen und Daten mit geringer Ähnlichkeit einer unterschiedlichen Gruppe zugeordnet werden [4, 1. Introduction]. Genau das ist das Ziel der Suche von äquivalenten Räumen.

Im Folgenden wird eine Methode des Clustering erarbeitet. Dazu wird zuerst eine Metrik, um die Unterschiedlichkeit der Zeitreihen zu messen, definiert, eine Möglichkeit zur Bewertung der Clusterqualität gewählt und ein Algorithmus vorgestellt.

2.2.1 Dynamic Time Warping

Ein Vergleich der rohen Zeitreihen ist in den meisten Fällen nicht effizient, aus dem Grund, dass die Reihen extrem lang werden und enormen Speicherplatz einnehmen können. Deswegen werden Dimensionsreduktionsverfahren verwendet und ein Clustering auf den niedrig-dimensionalen Räumen angewandt [4, 2. Representation methods for time series clustering]. Da der zu untersuchende Datensatz aber klein genug ist, kann auf den rohen Daten gearbeitet werden. Die Clusteranalyse dieser basiert zu einem hohen Maße auf der Auswahl einer geeigneten Metrik, um die Unterschiedlichkeit zu definieren [4, 3. Similarity/dissimilarity measures in time-series clustering]. Eine Metrik, die sich für Zeitreihen unterschiedlicher Länge anbietet und deshalb hier verwendet wird, ist das Dynamic Time Warping (DTW).

DTW wird oft genutzt im Zusammenhang mit der Auswertung von Zeitreihen, hat seinen Ursprung in der automatischen Spracherkennung und wurde dort unabhängig von Sakobe & Chiba [9] und Itakura [10] entwickelt [11].

Seien

$$X = \{X_0, \ldots, X_{T_X-1}\}, \quad Y = \{Y_0, \ldots, Y_{T_Y-1}\}$$

Zeitreihen der Länge T_X und T_Y und

$$A(X, Y)$$

die Menge aller Index-Tupel $(i, j), \quad i \in [T_X - 1], j \in [T_Y - 1]$. Dann gibt die Funktion

$$\mathrm{DTW}(X, Y) = \min_{\pi \subset A(X,Y)} \sqrt{\sum_{(i,j)\in\pi} \|X_i - Y_j\|^2} \tag{2.2}$$

mit Pfad π der Länge K die durch Verschiebung und Verzerrung erzeugte minimale euklidische Distanz zwischen zwei Reihen. Damit ein Pfad gültig ist, muss er folgende Bedingungen erfüllen (s. [11]):

1. Anfang und Ende der zwei Zeitreihen werden zusammen gruppiert

$$\pi_0 = (0, 0), \quad \pi_{K-1} = (T_X - 1, T_Y - 1)$$

2. Die Indizes der Zeitreihen steigen monoton an und jeder Index taucht mindestens einmal auf

Bei dieser Berechnung werden die beiden Zeitreihen verschoben und gestreckt, bis die euklidische Distanz zwischen diesen minimal ist. So ergibt sich ein Verschiedenheitsmaß, das für unterschiedliche lange Reihen genutzt und für multivariate erweitert werden kann. Aus diesen Gründen bietet es sich für die vorliegenden Daten perfekt an. Mithilfe der Python Implementierung *dtaidistance* [12] wird das DTW berechnet und visualisiert. Abbildung 2.2 zeigt beispielhaft wie das DTW angewandt auf den *großen Lernraum Audimax* und *kleinen Lernraum Audimax* dargestellt werden kann.

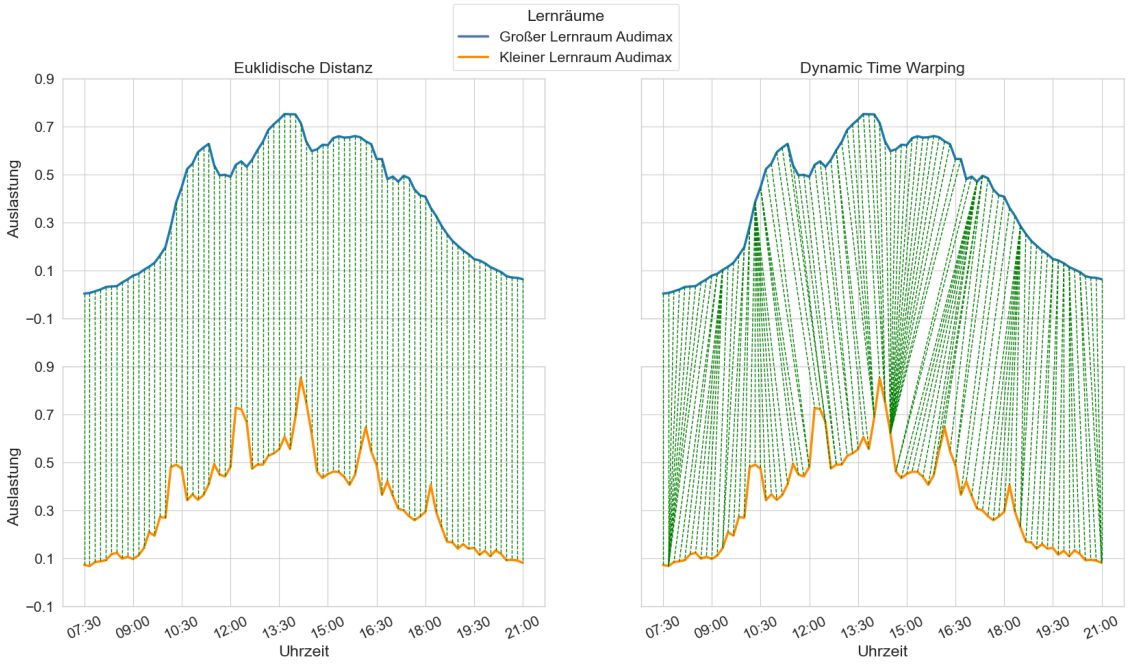

Abbildung 2.2: Beispielhafte Visualisierung der Abstände zwischen zwei Zeitreihen (Punkte, die miteinander verglichen werden, sind verbunden)

Da die Metrik nun festgelegt ist, kann eine sogenannte Distanzmatrix erstellt werden. Diese gibt für jede Zeitreihe den Abstand zu allen anderen Reihen und ist Grundlage der benutzten Clustering-Algorithmen.

2.2.2 Silhouettenkoeffizient

Was ist die optimale Anzahl an Clustern im Datensatz? Für eine sichere Beantwortung braucht man externe Kenntnisse über die vorliegenden Daten, die einem ermöglichen, die Anzahl im Vorhinein zu bestimmen. Bei dem hier zu betrachtenden Datensatz könnte die Anzahl der Gebäude, in denen die einzelnen Räume liegen, Aufschluss über die korrekte Anzahl geben. Dann wäre ein zu erwartendes Ergebnis, dass Räume im gleichen Gebäude einen gemeinsamen Cluster bilden. Eine weitere valide Annahme ist aber, dass die verschiedenen Raumtypen einen Cluster bilden, denn es gibt offene und geschlossene Räume. Ein offener Lernraum ist in diesem Kontext ein großflächiger Raum, der auch als Durchgang dient, wie beispielsweise der *Flur Zwischengeschoss Audimax*. Da nicht gesagt werden kann, ob Gebäude oder Raumtypen mehr Einfluss auf die Clusterbildung hat und welche weiteren unbekannten Faktoren es noch gibt, ist das Einbeziehen externer Informationen nicht sinnvoll und könnte das Ergebnis verfälschen. Deswegen wird nun die optimale Anzahl der Cluster als unbekannt angenommen und über interne Bewertungsmethoden geschätzt.

Die Bewertungsmethode, die für diese Analyse genommen wird, ist Rousseeuws Silhouettenkoeffizient [13]. Dieser bietet sich an, weil er mit verschiedenen Distanzmaßen verwendet werden kann (hier also mit dem DTW) und leicht zu interpretieren ist [14]. Des Weiteren benötigen verschiedene andere Bewertungsmetriken sogenannte Prototypen, das bedeutet den Durchschnittspunkt eines Clusters, um beispielsweise die Clusterintravarianz zu berechnen. Dieser Prototyp ist aber für Zeitreihen unterschiedlicher Länge weder leicht noch immer sinnvoll zu definieren [4, 4. Time-series cluster prototypes]. Dass der Silhouettenkoeffizient keine Prototypen verwendet, ist es weiterer Grund für die Auswahl.

Um den Koeffizienten zu definieren, muss erst festgelegt werden, was eine Silhouette ist. Die Silhouette $S(p)$ eines Datenpunktes p ist:

$$S(p) = \frac{b(p) - a(p)}{max\{a(p), b(p)\}} \tag{2.3}$$

wobei $a(p)$ der durchschnittliche Abstand zu allen Punkten im gleichen Cluster und $b(p)$ der durchschnittliche Abstand zu allen Datenpunkten im nächstgelegenen Cluster ist [13]. Es ist zu sehen, dass eine Silhouette immer im Bereich $[-1, 1]$ liegt. Der Koeffizient s_C entsteht dann aus der Mittelung der Silhouetten aller Punkte. Eine Bewertung des Koeffizienten kann Tabelle 2.2 entnommen werden.

Strukturierung	Wertebereich von s_C
stark	$0.75 < s_C \leq 1$
brauchbar	$0.5 < s_C \leq 0.75$
schwach	$0.25 < s_C \leq 0.5$
keine Struktur	$0 < s_C \leq 0.25$

Tabelle 2.2: Bewertung der Strukturierung der Cluster durch den Silhouettenkoeffizienten [14]

2.2.3 Agglomeratives Hierarchisches Clustering

Das agglomerative hierarchische Clusterverfahren ist ein Verfahren, bei dem zunächst jeder Datenpunkt als eigenständiger Cluster betrachtet wird und dann schrittweise nach gewählter Metrik die beiden am nächsten liegenden Cluster zusammengefasst werden. Dieser Algorithmus bietet sich an, weil Zeitreihen unterschiedlicher Länge und mit mehrdimensionalen Datenpunkten durch die freie Wahl der Distanzmetrik analysiert werden können. Auch brauchen andere Verfahren wie beispielsweise K-Means einen Prototyp. Die Problematik, die dieser für Zeitreihen verursacht, wurde bereits in Unterabschnitt 2.2.2 erwähnt. Nachteile des hierarchischen Verfahren umfassen aber, dass einmal gebildete Cluster nicht noch einmal verändert werden können, worunter die Qualität des Clustering leidet [4, 5.1. Hierarchical clustering of time-series]. Für das Clustering und den Silhouettenkoeffizienten wird die Implementierung von Scikit-Learn [15] verwendet.

2.3 Clusterauswahl

Nachfolgend wird auf Basis der vorgestellten Methoden die Clusteranalyse durchgeführt. Dazu werden die Daten mit dem hierarchischen Verfahren in unterschiedlich viele Cluster eingeteilt, für jede Menge wird der Silhouettenkoeffizient berechnet und die Anzahl mit dem besten Wert genauer betrachtet.

Abbildung 2.3 zeigt die Silhouettenkoeffizienten in Abhängigkeit der Clusteranzahl, die sich für die vorliegenden Daten ergeben. Es ist zu sehen, dass zwei Cluster von dem Koeffizienten bevorzugt werden. Eine Visualisierung der Zeitreihen nach der sich ergebenden Clusterzugehörigkeit ist in Abbildung 2.4 zu sehen.

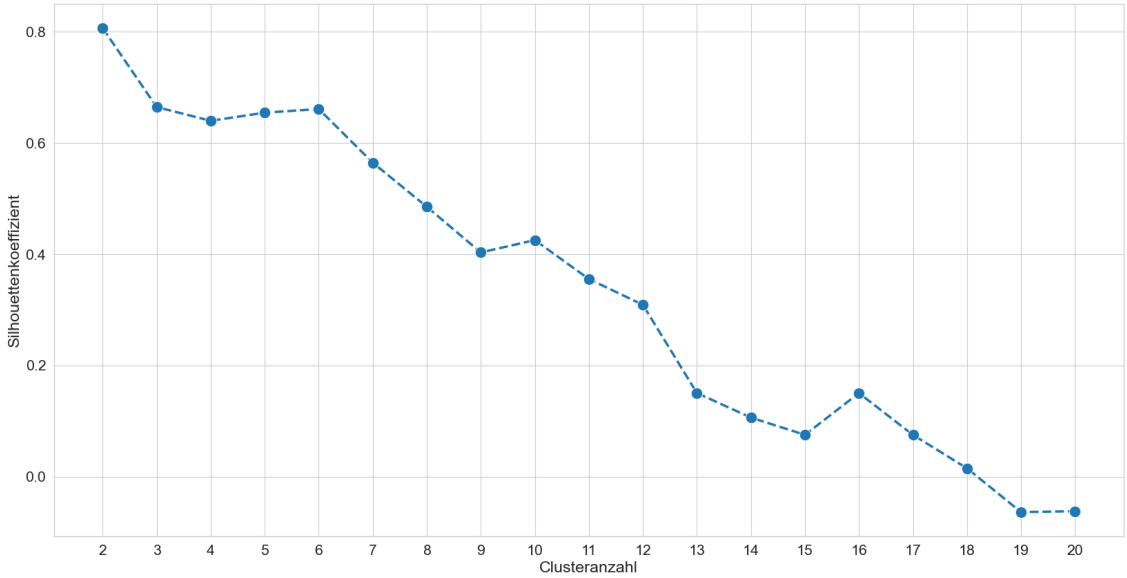

Abbildung 2.3: Grafische Darstellung der Silhouettenkoeffizienten beim hierarchischen Clustering

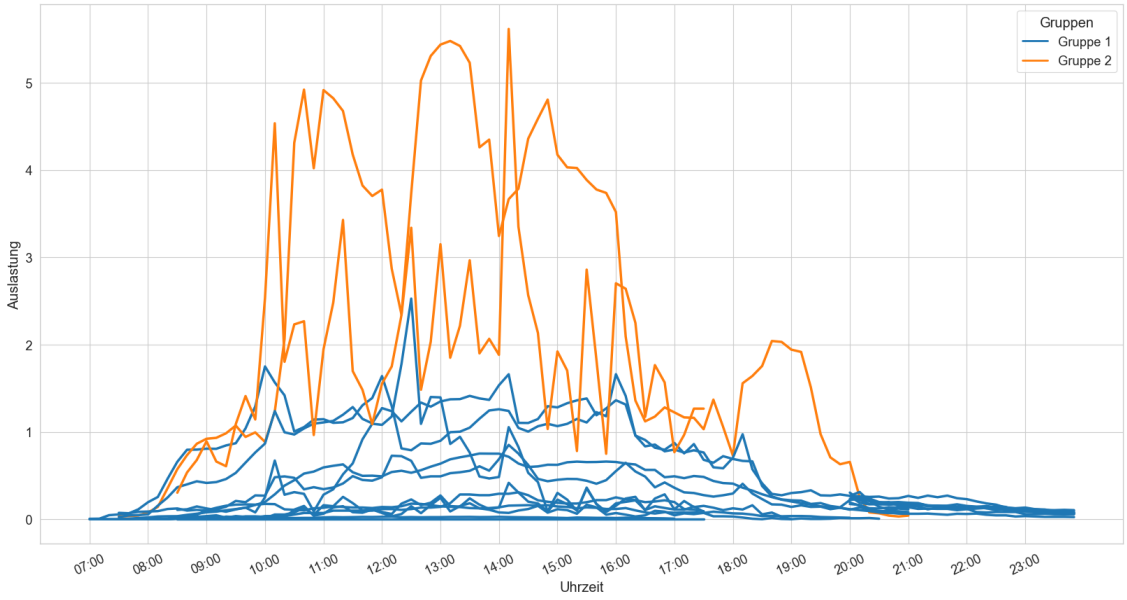

Abbildung 2.4: Einteilung der Zeitreihen in zwei Gruppen

Diese Zuteilung ist nicht intuitiv, denn einer der Cluster ist deutlich größer als der andere. Auch existieren erhebliche Unterschiede innerhalb der Gruppen. Damit Räume wirklich als äquivalent betrachtet werden können, sollten die Unterschiede in einem Cluster möglichst gering sein. Andernfalls kann ein Verfahren zu Auslastungsklassifizierung der einzelnen Gruppen fehleranfällig sein. Es sprechen also Argumente dafür, dass diese Zuteilung für den Anwendungsfall nicht optimal ist.

Um zu verstehen, warum wenige Cluster von dem Silhouettenkoeffizienten bevorzugt werden, ist es lohnenswert, sich die Daten in einem zweidimensionalen Raum anzuschauen. Für diese Dimensionsreduktion wird Isomap verwendet. Isomap ist ein Dimensionsreduktionsverfahren, das

auf Grundlage einer Distanzmatrix die zugrundeliegende Geometrie eines Datensatzes lernen und im Gegensatz zu anderen Verfahren wie der Principal Component Analysis und Multidimensional Scaling sogar nichtlineare Zusammenhänge erkennen kann [16]. Als Beispiel, um das Verhalten zu verdeutlichen: Besitzt man eine Distanzmatrix der Autobahnkilometer zwischen den Deutschen Städten Frankfurt, Hamburg, Berlin und Köln, könnte man mit Isomap diese (4×4)-Matrix auf eine (4×2)-Matrix reduzieren. Da dadurch jeder Datenpunkt nur noch 2-Dimensionen hat, ist eine visuelle Darstellung dieser problemlos möglich und der entstehende Graph würde die tatsächliche Lage der Städte zueinander fast fehlerfrei wiedergeben. Abbildung 2.5 zeigt die räumliche Lage der Zeitreihen im Zweidimensionalen.

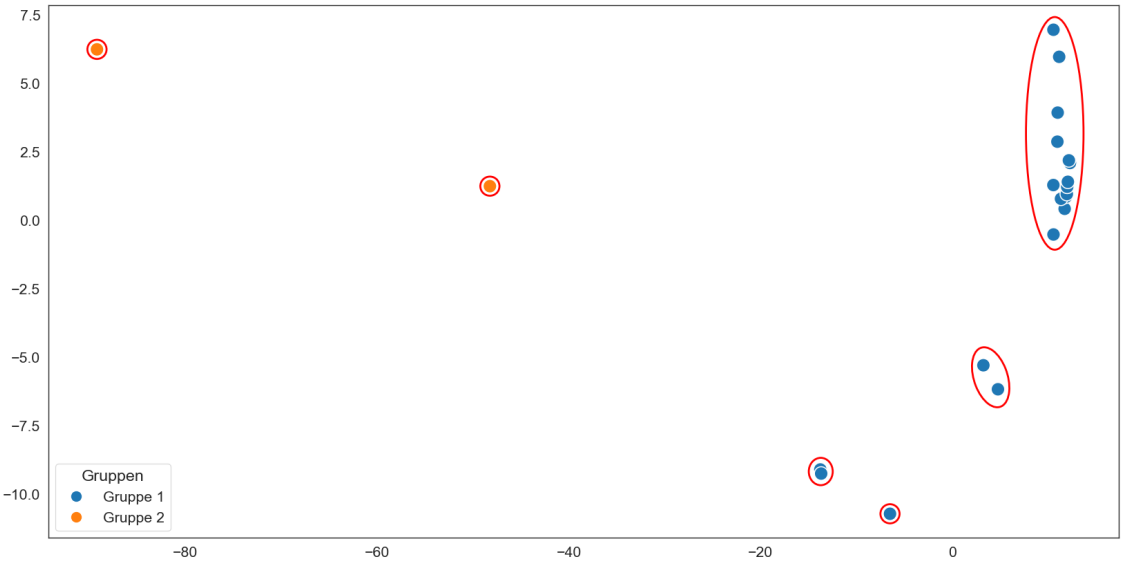

Abbildung 2.5: Visualisierung der Zeitreihen durch Isomap

Es ist zu erkennen, dass ein großer Abstand zwischen beiden Clustern (farblich markierte Punkte) existiert. Aus der Formel des Silhouettenkoeffizienten (s. Gleichung 2.3) geht hervor, dass dieser hohe Werte hat, wenn die Abstände zwischen den Clustern im Vergleich zu den Abständen innerhalb der Cluster groß sind. Deshalb werden hier wenige Gruppen vom Koeffizienten bevorzugt, weil so die Abstände zu der nächstgelegenen Gruppe maximal sind. Durch Standardisierung der Daten könnte ein solcher Effekt möglicherweise ausgeglichen werden, ist bei diesen Daten aber nicht sinnvoll, da dann durch das DTW lediglich die Form der Zeitreihen, also wann Anstiege und Abfälle sind, und nicht auch die Mächtigkeit dieser verglichen werden würde.

Auch zeigt die Visualisierung, dass potenziell sechs Gruppen gefunden werden können (s. rote Umrandungen). Der Koeffizient für sechs Cluster liegt bei ungefähr 0.65 und hat damit einen brauchbaren Wert (s. Tabelle 2.2). Eine Einteilung der Zeitreihen in sechs Gruppen kann Abbildung 2.6 entnommen werden.

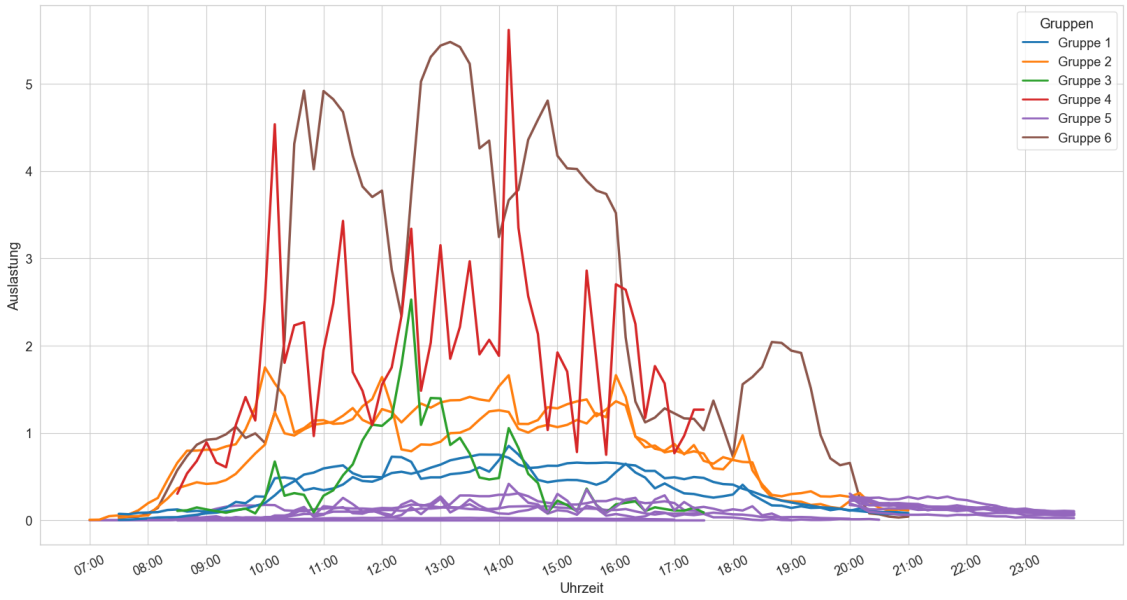

Abbildung 2.6: Einteilung der Zeitreihen in sechs Gruppen

Dieses Clustering ist nun intuitiv. Vielmehr sind, bis auf wenige Ausnahmen, Räume aus dem gleichen Gebäude in einer Gruppe. Die anderen Gruppen setzen sich aus offenen Räumen, wie beispielsweise die Gruppe bestehend aus dem *Foyer Audimax* und *PPS Foyer im Erdgeschoss*, zusammen. Aus dem Grund, dass diese beiden Zusammenhänge, Gebäudezugehörigkeit und Raumtypen, in Unterabschnitt 2.2.2 als eine Möglichkeit der externen Clustervalidierung genannt wurden, ist davon auszugehen, dass es sich nun um eine optimale Einteilung handelt und die Gruppierung in zwei Cluster fehlerhaft ist.

Diese Gruppen enthalten nun jeweils Räume, die während der Öffnungszeiten ähnlich besucht werden, in ihrem Verhalten also äquivalent sind. Da die Gruppen unterschiedlich zueinander sind, müssen sie in den weiteren Analysen eigenständig betrachtet werden. Tabelle 2.3 fasst die Gruppeneinteilung zusammen.

Gruppe	Räume
Gruppe 1	Großer Lernraum Audimax
	Kleiner Lernraum Audimax
Gruppe 2	Foyer Audimax
	PPS Foyer im Erdgeschoss
Gruppe 3	Foyer Informatik
Gruppe 4	Garderobe AH V
Gruppe 5	514 Geo
	B 002
	B 061
	Bibliothek 4006
	Flur Anglistik
	Foyer AH VI
	SE 001
	SE 002
	SE 003
	SE 101
	SE 102
	SE 103
	SE 108
	SE 209
Gruppe 6	Flur Zwischengeschoss Audimax

Tabelle 2.3: Gruppierung der Lernräume nach gleichem Verhalten

3 Fehleranalyse

Da nun Räume gleichen Verhaltens gefunden wurden, können die Einflüsse der Störfaktoren für jede dieser Gruppen betrachtet werden. In der Einleitung wurde bereits erwähnt, dass sowohl mehrere Geräte pro Person als auch das Einwählen im falschen Access Point zu Störungen bei der korrekten Schätzung der Personenanzahl führen können. In diesem Kapitel wird sich mit der Frage *"Wie groß ist der Einfluss der Störfaktoren auf eine Messung und kann man diese ausgleichen?"* auseinandergesetzt. Die Frage wird dabei in

- Einfluss der Störfaktoren
- Ausgleich der Störung

gespalten, wobei der Einfluss der Störfaktoren zuerst abgearbeitet wird. Danach wird ein Einschub vorgenommen und die Frage *"Wie soll die Klassifizierung geschehen?"* geklärt, damit anschließend auf Basis der gewonnenen Erkenntnisse der Ausgleich der Störung behandelt werden kann.

3.1 Einfluss der Störfaktoren

Liegen überhaupt Störungen in der Messung vor? Bisher wurde die Existenz von Störfaktoren einfach angenommen. Auch wenn die genannten Annahmen logisch erscheinen, muss die Existenz jener dennoch belegt werden.

Betrachtet man die Definition der Auslastung (s. Gleichung 2.1), so ist leicht zu sehen, dass, wenn keine Störfaktoren vorliegen, für die Auslastung U gelten muss: $U \in [0, 1]$. Aus dem vorherigen Kapitel geht aber hervor, dass für vier gefundene Gruppen die gemessene Auslastung nicht in diesem Bereich liegt (s. beispielsweise Abbildung 2.1). Für die Gruppen

- Gruppe 2
- Gruppe 3
- Gruppe 4
- Gruppe 6

kann also mit Sicherheit gesagt werden, dass die gemessenen Auslastungen nicht gleich mit der tatsächlichen sind. Für die anderen Gruppen können die Störungen mit dieser Methode nicht nachgewiesen werden, weswegen sie nicht als störungsfrei angenommen werden, aber davon auszugehen ist, dass keine "großen" Störungen vorliegen, weshalb diese Gruppen nicht weiter betrachtet werden müssen.

Auch ist zu sehen, dass die Auslastung jeder Gruppe in einem anderen Intervall liegt, also verschiedene Obergrenzen u_i existieren. Folglich kann der Einfluss der Störungen auch nichts als allgemeingültig angenommen werden und jede Gruppe wird anders beeinflusst und muss eigenständig betrachtet werden. Eine mögliche Erklärung für diese Unterschiede ist, dass die Räume verschieden nah an Hörsälen oder anderen stark besuchten Räumen liegen und sich so das falsche Einwählen in Access Points mehr oder weniger stark auf die Messung auswirkt. Die genaue Herkunft der Störung hat eine hohe Relevanz und wird auch weiter in Abschnitt 4.2 beachtet.

3.2 Klassifikation

Bevor eine Möglichkeit zum Ausgleichen der Störungen vorgestellt wird, sollte sich mit der Klassifikation der Auslastung auseinandergesetzt werden. Dafür besteht nun die Annahme, dass bereits eine Möglichkeit zum Entfernen der Störfaktoren gefunden wurde und jegliche Auslastungen im Intervall $[0, 1]$ liegen. Um ein Verständnis für die typische Nutzung der Lernräume zu erhalten, werden verschiedene Räume besucht, Zählungen der Auslastung vorgenommen und die Belegung von Sitzplätzen beobachtet. Aufgrund des zeitlichen Aufwands können dabei nur subjektive Erfahrungen gewonnen werden, die zusammen mit dem aktuellen Vorgehen in der RWTHapp genutzt werden, um ein Klassifizierungsmodell zu erstellen.

Wie in Kapitel 1 angesprochen, wird in der momentanen Variante die Auslastung über eine Art Ampelschema mit drei Klassen angegeben. Dieser Ansatz ist sinnvoll, denn er erlaubt es, kleinere Fehler beim Ausgleichen der Störungen zu machen. Durch eine Einteilung in Klassen werden $\varepsilon - Intervalle$ geschaffen, in denen die vorhergesagte Auslastung richtig interpretiert wird, auch wenn sie von der tatsächlichen abweicht.

Die aktuell verwendeten Klassen mit der Schwelle der Auslastung, ab der sie gelten, sind:

- wenig los $\geq 0\%$
- mäßig viel los $\geq 50\%$
- viel los $\geq 95\%$

Die Wahl von drei Klassen ist vernünftig und wird auch hier verwendet. Zu viele Klassen wären zu kompliziert zu interpretieren, zu wenige zu unpräzise. Die Grenzen basieren aber auf der Annahme, dass die Sitzplätze "vernünftig"genutzt werden, also beispielsweise ein Tisch, der für vier Personen ausgelegt ist, auch von vier Personen genutzt wird. Das ist aber in der Regel nicht der Fall. Oftmals belegen weniger Personen einen Tisch, wodurch ein Raum bereits voll sein kann beziehungsweise als voll wahrgenommen wird, auch wenn die Auslastung dies nicht hergibt.

Bei dem Besuchen der Lernräume, wurde festgestellt, dass oft und zu unterschiedlichen Uhrzeiten eine Auslastung um den Wert 0.4 vorhanden war. Das liegt nicht daran, dass die Lernräume wenig besucht waren, sondern dass kaum noch Platz in diesen war. Es wurde bereits erklärt, dass Räume nicht ideell benutzt werden, weswegen sie bereits bei mittelmäßiger Auslastung voll sein können. Dieser Effekt greift hier. Aus diesem Grund sollte die Schranke der Klasse viel los von 95% nach unten gesetzt werden. Ein valider Wert ist 60%. Das ist höher als die vorher beschrieben Grenze von 40%, denn dabei wird mit einbezogen, dass in der vorlesungsfreien Zeit, wenn für Klausuren gelernt werden muss, sich eher an einen Tisch gesetzt wird, an dem schon andere Personen sitzen. Diese Annahme basiert auch auf subjektiven Erfahrungen. Die übrigen beiden Klassen werden gleichmäßig auf das Intervall $[0, 0.6]$ aufgeteilt. Tabelle 3.1 fasst die Klasseneinteilung zusammen.

Klasse	Bereich
wenig los	$[0, 0.3)$
mäßig viel los	$[0.3, 0.6)$
viel los	$[0.6, 1]$

Tabelle 3.1: Bereiche der verschiedenen Klassen

Nachfolgend wird ein Versuch zum Ausgleich der Störfaktoren unternommen.

3.3 Ausgleich der Störung

Sei im weiteren Verlauf

$$U_i \in [0, u_i] \tag{3.1}$$

die gemessene Auslastung der Gruppe i mit der Obergrenze u_i.

Wie kann die Störung ausgeglichen werden? Ein Ausgleichen der Störung heißt in diesem Fall, dass für alle Gruppen eine neue Obergrenze $u_i' \leq 1$ gefunden werden muss. Es wird demnach nach stetigen Abbildungen

$$f_i : [0, u_i] \to [0, u_i'] \subseteq [0, 1] \tag{3.2}$$

mit $f_i(0) = 0$ gesucht. Warum diese Bedingung? Wenn $f_i(0) \neq 0$ gelten würde, wäre eine Grundlast vorhanden.

Der Verständlichkeit halber wird im weiteren Verlauf von der allgemeinen Abbildungsfunktion f und nicht den f_i für jede Gruppe gesprochen. Diese Differenzierung wird dann bei der Berechnung wieder aufgenommen.

Eine simple Möglichkeit einer solchen Abbildung ist die Min-Max Normalisierung. Dabei wird eine lineare Transformation der Daten in ein neues Intervall vorgenommen [17, 3.5.2. Data Transformation by Normalization]. Seien min_A und max_A das Minimum und Maximum des Datensatzes und new_min_A, new_max_A die Grenzen des gewünschten Intervalls. Dann wird jeder Datenpunkt v_k über folgende Formel zu v_k':

$$v_k' = \frac{v_k - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A$$

$v_k \in [min_A, max_A]$ und $v_i' \in [new\_min_A, new\_max_A]$ [17, 3.5.2. Data Transformation by Normalization].

Für den hier betrachteten Anwendungsfall würde sich die Formel zu

$$f : [0, u] \to [0, 1], \quad f(x) = \frac{x}{u}$$

vereinfachen. Dieser Ansatz liefert eine gute Abbildung, basiert aber auf Annahmen, die in diesem Fall für Probleme sorgen könnten:

- Es wird $u_i' = 1$ gesetzt. Es besteht also die Annahme, dass ein Raum bei einer gemessenen Auslastung von u_i komplett gefüllt ist.
- f ist eine streng monoton steigende Gerade, also wird angenommen, dass eine höhere Messung bedeutet, dass auch tatsächlich mehr Personen im Raum sind.

Beide Annahmen können und sollten aber nicht als allgemeingültig angenommen werden und könnten dazu führen, dass auch nach der Abbildung Auslastungen > 1 zu messen sind.

Deswegen wird ein anderer Ansatz zum Finden einer Abbildung verfolgt: Es wird nachfolgend für jede von Störungen befallene Gruppe ein Repräsentant ausgewählt, die ausgewählten Räume werden besucht und die tatsächliche Auslastung wird gemessen. So kann der Zusammenhang zwischen durch den Access Point gemessener und tatsächlicher Auslastung approximiert und die Störung ausgeglichen werden.

3.3.1 Lineare Regression

Die Auswahl eines Repräsentanten ist bei den störanfälligen Gruppen simpel, denn es gibt nur eine Gruppe (Gruppe 2), die mehr als einen Raum enthält (s. Tabelle 2.3). Bei dieser wird das *Foyer Audimax* repräsentativ für die Gruppe genommen, da es in einem Gebäude (dem Audimax) liegt, das sowieso für Messungen anderer Gruppen besucht werden muss. Tabelle 3.2 gibt eine Übersicht der ausgewählten Repräsentanten. Es sei angemerkt, dass eine Auswahl solcher repräsentativen Lernräume unproblematisch ist, da in einer Gruppe nur Räume sind, die über längeren Zeitraum im Durchschnitt gleich besucht wurden, demnach als äquivalent angenommen werden können.

Gruppe	Repräsentant
Gruppe 2	Foyer Audimax
Gruppe 3	Foyer Informatik
Gruppe 4	Garderobe AH V
Gruppe 6	Flur Zwischengeschoss Audimax

Tabelle 3.2: Repräsentanten der Gruppen

Da es verschiedene Notationen für Matrizen, Vektoren, Skalare, etc. gibt, wird die in den nachfolgenden Abschnitten verwendete Notation hier kurz zusammengefasst:

$$
\begin{aligned}
x \in \mathbb{R}: &\quad \text{Ein Skalar} \\
\mathbf{x} \in \mathbb{R}^n: &\quad \text{Ein Vektor} \\
\mathbf{X} \in \mathbb{R}^{m \times n}: &\quad \text{Eine Matrix}
\end{aligned}
$$

Dieses Konzept wird auch auf Funktionen angewandt, beispielsweise:

$$
\mathbf{f}(\cdot) : \cdot \to \mathbb{R}^n: \quad \text{Eine Funktion, die auf einen Vektor abbildet}
$$

Dem Umstand entsprechend, dass Messungen der tatsächlichen Anzahl der Personen in einem Raum nicht automatisch durchgeführt werden können, müssen die ausgewählten Räume wie auch in Abschnitt 3.2 zu verschiedenen Zeiten besucht und die Personen in dem Raum gezählt werden. Dann wird anhand von Gleichung 2.1 die Auslastung bestimmt. Es ergibt sich somit für jede Gruppe für jede Messung ein Datentupel $(\mathbf{x}, t)$, wobei $\mathbf{x}$ die gemessene Auslastung anhand des Access Points und t die tatsächliche Auslastung, das "Target", ist. Alle gemessenen Auslastungen werden durch die Matrix $\mathbf{X}$ und alle "Targets" durch den Vektor $\mathbf{t}$ angegeben. Eine grafische Darstellung der Messungen ist in Abbildung 3.1 zu sehen.

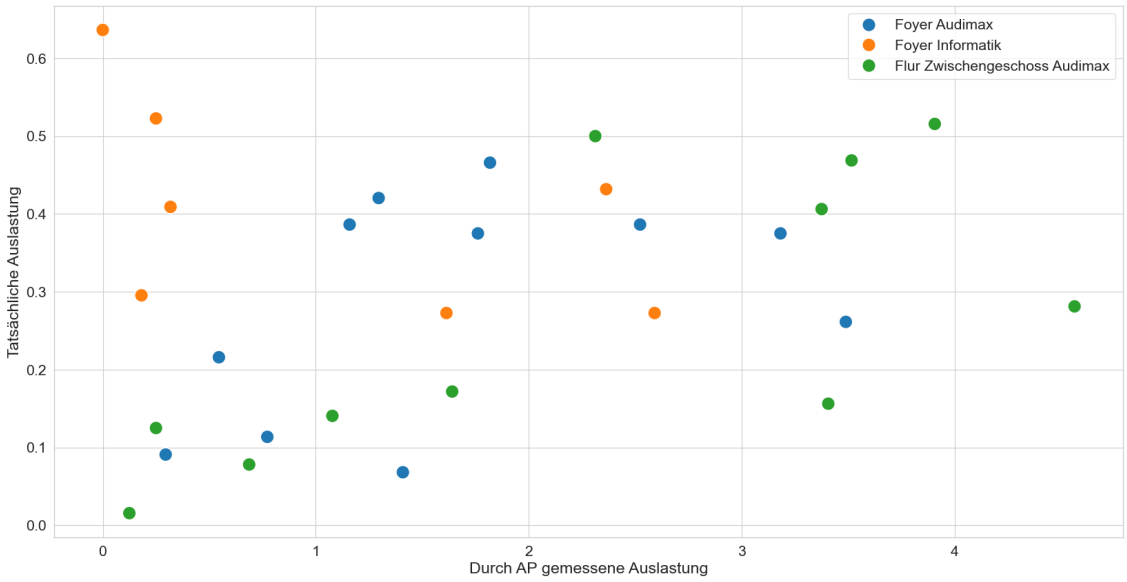

Abbildung 3.1: Durch Access Points gemessene Auslastung im Vergleich mit der tatsächlichen

Auffällig ist, dass zu Beginn von Messungen von vier Gruppen die Rede war, nun aber nur drei zu sehen sind. Das liegt daran, dass während des Durchführens der Messungen aufgefallen ist, dass der Lernraum *Garderobe AH V* so nicht existiert. Normalerweise sollte vor dem Hörsaal AH V ein einzelner Tisch stehen. Das ist aber nicht der Fall. Dieser Umstand wird in Abschnitt 4.2 diskutiert. Für den weiteren Verlauf wird dieser Lernraum deswegen nicht weiter betrachtet.

Wie kann man aus diesen Stichproben allgemeine Funktionen f_i finden? Dazu wird der Ansatz der linearen Regression verwendet. Das Ziel dieser ist es, einen Prädiktor

$$y(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) + w_0$$

zu berechnen, der den quadratischen Fehler zwischen Vorhersage und tatsächlichem Wert minimiert [18, 3.1. Linear Basis Function Models]. $\mathbf{w}$ ist dabei ein Vektor von Gewichten und ϕ eine nicht-lineare Transformation. Es ist zu erkennen, dass $\mathbf{w}^T\mathbf{x}$ die Definition einer Hyperebene, also die Verallgemeinerung einer Geraden in einem mehrdimensionalen Raum ist. $\mathbf{w}^T\phi(\mathbf{x})$ ist dann eine Hyperebene in einem nicht-linearen Dimensionsraum. Diese Funktion mag zwar wie eine Verkomplizierung des betrachteten Problems wirken, nichtsdestotrotz kann aber durch diese Darstellung, mithilfe von Differenzierung, eine geschlossene Lösung für $\mathbf{w}$ gefunden werden und dadurch eine Funktion y von beliebiger Komplexität berechnet werden.

Nachfolgend wird der Prädiktor mit der gesuchten Abbildungsfunktion verbunden, um ein theoretisches Modell herzuleiten. Sei $f(x) = y(\mathbf{x})$. Es folgt aus der Bedingung $f(0) = 0$, dass $w_0 = 0$ gelten muss. Im Allgemeinen würde das w_0 mit in den Vektor $\mathbf{w}$ gezogen, indem die Bedingung $\phi_0(\mathbf{x}) = 1$ eingeführt wird. Da hier kein w_0 erwünscht ist, wird die Bedingung zu $\phi_0(\mathbf{x}) \neq 1$ geändert. Die Regression wird nun durch ein Approximationspolynom durchgeführt, das bedeutet $\phi(x) = [x^1, x^2, \ldots, x^Q]^T$. Ein Polynom wird verwendet, da es schnell zu berechnen ist und mit steigendem Grad auch komplizierte Funktionen lokal darstellen kann.

Ein gutes Approximationspolynom kann durch Minimierung des quadratischen Fehlers (s. Gleichung 3.3) gefunden werden.

$$L(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}(\mathbf{w}^T\phi(\mathbf{x}_n) - t_n)^2 \tag{3.3}$$

$$\begin{aligned} &\frac{\partial L(\mathbf{w})}{\partial \mathbf{w}} \stackrel{!}{=} 0 \\ \Leftrightarrow \quad &\mathbf{w} = (\mathbf{\Phi}^T\mathbf{\Phi})^{-1}\mathbf{\Phi}^T\mathbf{t} \end{aligned}$$

mit $\mathbf{\Phi} = \phi(\mathbf{X}) = [\phi(\mathbf{x}_1), \ldots, \phi(\mathbf{x}_N)]^T$ [18, 3.1.4. Regularized least squares]. Es liegt nun also eine Möglichkeit vor, schnell ein Ausgleichspolynom zu berechnen.

Zwei wichtige Umstände müssen dabei aber beachtet werden

1. Polynome bilden auf ein Intervall $[-\infty, \infty]$ ab, wodurch auch nach Abbildung noch Auslastungen > 1 und < 0 auftauchen könnten.
2. Die Polynomfunktion muss nicht monoton steigend sein, was dazu führen kann, dass ein Abfall zu sehen ist und bei hoher gemessener Auslastung eine geringe tatsächliche Auslastung prognostiziert wird. Es ist aber davon auszugehen (wenn nicht starke Störungen vorliegen), dass die gemessene Auslastung annähernd proportional zur tatsächlichen ist.

Damit dieses Problem gelöst werden kann, wird die Regression modifiziert.

3.3.2 Logistische Regression

Um zu verhindern, dass Werte des Polynoms unendlich groß werden können, wird dieses wie gewünscht auf das Intervall $[0, 1]$ beschränkt. Das wird durch das Anwenden der Sigmoid- oder Schwanenhalsfunktion $\sigma(x) = \frac{1}{1+e^{-x}} \in [0, 1]$ erreicht. Es wird nun nicht mehr $y(\mathbf{x})$, sondern $\hat{y}(\mathbf{x}) = \sigma(y(\mathbf{x}))$ betrachtet. Dieses Modell ist bekannt als logistische Regression [18, 4.3.2 Logistic regression]. Ein kleineres Problem, das sich dadurch ergibt, ist, dass $\sigma(0) = 0.5$ gilt. Somit ist die Bedingung $f(0) = 0$ nicht erfüllt. Generell kann diese Bedingung durch die Sigmoidfunktion nicht erfüllt werden, da diese Funktion nur für $-\infty$ gegen 0 konvergiert. Eine Lösung ist aber, die Funktion nach rechts zu verschieben und die Bedingung aufzulockern, sodass gelten soll $f(0) \approx 0$. Das ist logisch vertretbar, da zwar eine Grundlast existieren würde, die Klasse aber trotzdem wenig los wäre. Die verwendete Sigmoidfunktion sei deshalb

$$\sigma(x) = \frac{1}{1 + e^{-(x-3)}}$$

Um zu verhindern, dass die Regressionsfunktion abfällt, wird ein Term eingeführt, der steigende Funktionen belohnt und fallende Funktionen bestraft. Bei gegebenen Parametern $\mathbf{w} \in \mathbb{R}^D$ ist dieser Term

$$\Omega(\mathbf{w}) = -\sum_{n=1}^{N}\frac{\partial}{\partial \mathbf{x}}\Big(\sigma(\mathbf{w}^T\phi(\mathbf{x}_n))\Big) = -\sum_{n=1}^{N}\mathbf{w}^T\phi'(\mathbf{x}_n)\hat{y}_n(1-\hat{y}_n) \tag{3.4}$$

$$\begin{aligned} \hat{y}_n &= \hat{y}(\mathbf{x}_n) \\ \phi'(x) &= [1, 2x, \ldots, Qx^{Q-1}]^T \end{aligned}$$

$\Omega(\mathbf{w})$ ist die Summe des Differenzials der Approximationsfunktion an den Stellen $\mathbf{x}_i$. Die Summe ist groß positiv, wenn die Funktion fallend ist und negativ, wenn sie steigend ist. Durch einen Faktor λ kann die Stärke des Einflusses gesteuert werden. Dadurch wird forciert, dass die Funktion im Allgemeinen steigend ist. Es wird zudem wie auch in der linearen Regression der quadratische Fehler verwendet (s. Gleichung 3.3)

Es folgt die Fehlerfunktion

$$E(\mathbf{w}) = L(\mathbf{w}) + \lambda\Omega(\mathbf{w}) \tag{3.5}$$

Das Minimum dieser Funktion kann nicht mehr geschlossen gefunden werden, weswegen das Gradientenabstiegsverfahren benutzt wird. Bei gegebenen Startgewichten $\mathbf{w}^{(0)}$ und Lernrate η folgt die Aktualisierungsregel [18, 3.1.3 Sequential learning]

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} - \eta\nabla E(\mathbf{w}^{(\tau)})$$

Nachfolgend wird nun stückweise $\nabla E(\mathbf{w}) = \nabla L(\mathbf{w}) + \lambda\nabla\Omega(\mathbf{w})$ berechnet.

$$\begin{aligned}
\nabla L(\mathbf{w}) &= \frac{\partial}{\partial \mathbf{w}}\left(\frac{1}{2}\sum_{n=1}^{N}(\hat{y}_n - t_n)^2\right) \\
&= \frac{1}{2}\sum_{n=1}^{N}\frac{\partial}{\partial \mathbf{w}}\left((\hat{y}_n - t_n)^2\right) \\
&= \frac{1}{2}\sum_{n=1}^{N}2\frac{\partial}{\partial \mathbf{w}}\left(\hat{y}_n - t_n\right)(\hat{y}_n - t_n) \\
&= \sum_{n=1}^{N}\phi_n\hat{y}_n(1-\hat{y}_n)(\hat{y}_n - t_n) \\
&= \mathbf{\Phi}^T\hat{\mathbf{S}}(\hat{\mathbf{y}} - \mathbf{t})
\end{aligned}$$

$\hat{\mathbf{y}} = [\hat{y}_1, \ldots, \hat{y}_N]^T$
$\phi_n = \phi(\mathbf{x}_n)$
$\hat{\mathbf{S}}$ Diagonalmatrix mit $\hat{S}_{nn} = \hat{y}_n(1-\hat{y}_n)$

$$
\begin{aligned}
\nabla\Omega(\mathbf{w}) &= \frac{\partial}{\partial \mathbf{w}}\left(-\sum_{n=1}^{N} \mathbf{w}^T \phi'(\mathbf{x}_n)\hat{y}_n(1-\hat{y}_n)\right) \\
&= -\sum_{n=1}^{N} \frac{\partial}{\partial \mathbf{w}}\left(\mathbf{w}^T \phi'(\mathbf{x}_n)\hat{y}_n(1-\hat{y}_n)\right) \\
&= -\sum_{n=1}^{N} \frac{\partial}{\partial \mathbf{w}}\left(\mathbf{w}^T \phi'_n\right)\hat{y}_n(1-\hat{y}_n) + \mathbf{w}^T\phi'_n \frac{\partial}{\partial \mathbf{w}}\left(\hat{y}_n(1-\hat{y}_n)\right) \\
&= -\sum_{n=1}^{N} \phi'_n\hat{y}_n(1-\hat{y}_n) + \mathbf{w}^T\phi'_n(\phi_n\hat{y}_n(1-\hat{y}_n) - \phi_n\hat{y}_n(1-\hat{y}_n)2\hat{y}_n) \\
&= -\sum_{n=1}^{N} \phi'_n\hat{y}_n(1-\hat{y}_n) + \mathbf{w}^T\phi'_n\phi_n\hat{y}_n(1-\hat{y}_n)(1-2\hat{y}_n) \\
&= -(\mathbf{\Phi'}^T\hat{\mathbf{S}}\mathbf{1} + \mathbf{\Phi}^T\hat{\mathbf{S}}\tilde{\mathbf{S}}\mathbf{\Phi'}\mathbf{w})
\end{aligned}
$$

$\mathbf{1} = [1, \ldots, 1]^T \in \mathbb{R}^N$
$\tilde{\mathbf{S}}$ Diagonalmatrix mit $\tilde{S}_{nn} = (1 - 2\hat{y}_n)$

Daraus folgt die Aktualisierungsregel

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} - \eta(\mathbf{\Phi}^T\hat{\mathbf{S}}(\hat{\mathbf{y}} - \mathbf{t}) - \lambda(\mathbf{\Phi'}^T\hat{\mathbf{S}}\mathbf{1} + \mathbf{\Phi}^T\hat{\mathbf{S}}\tilde{\mathbf{S}}\mathbf{\Phi'}\mathbf{w}^{(\tau)}))$$

Mit dieser kann iterativ eine Regressionsfunktion gefunden werden. Für das Regressionsmodell müssen nun noch zwei Hyperparameter gewählt werden

1. der Polynomgrad Q
2. die Stärke des Strafterms λ

Um für diese passende Werte zu finden, wird die Leave-One-Out Kreuzvalidierung (LOOCV) verwendet.

3.3.3 Leave-One-Out Kreuzvalidierung

Im Folgenden wird die Kreuzvalidierung vorgestellt. Danach werden für die drei noch bestehenden Gruppen die besten Parameter Q und λ gesucht.

Die LOOCV ist ein Validierungsverfahren für Modelle des maschinellen Lernens, bei dem aus einem gegebenen Datensatz iterativ ein Punkt entfernt wird und auf den verbleibenden das Modell trainiert wird. Der entfernte Punkt wird dann zum Validieren des Modells verwendet, indem dieser mit der Vorhersage verglichen wird. Für den Vergleich wird hier ein Missklassifikationsfehler verwendet. Dieser ist 1, wenn eine unterschiedliche Klasse zwischen Prediction und Target vorliegt und 0 sonst. Dieses Verfahren wird für jeden Punkt des Modells durchgeführt und der allgemeine Fehler ergibt sich als Mittelung des Fehlers aller Punkte. Die LOOCV eignet sich besonders für Datensätze mit wenigen Datenpunkten [19]. Die Parameter Q und λ werden auf dieser Basis durch eine Gittersuche gefunden. Das heißt, es wird für jede mögliche Kombination von Q und λ der Modellfehler berechnet und anschließend das Paar mit dem kleinsten Fehler als optimal angenommen [19]. Mit diesen Parametern wird das Modell dann auf allen Datenpunkten trainiert.

Aus den Regressionsfunktionen wird daraufhin mittels der in Tabelle 3.1 gegebenen Intervalle jeweils ein Klassifikator erzeugt. Eine berechtigte Frage an dieser Stelle ist, warum nicht ein Diskriminator verwendet wurde. Ein Diskriminator ist eine Funktion, die gegebene Eingaben in eine Klasse einteilt. Gleiches wurde hier durch den Umweg über die Regression erzeugt. Die Vorgehensweise, um einen Diskriminator zu berechnen, ist nahezu identisch zur Regression und es würde keine falschen Klassifizierungen geben [18, 4. Linear Modeles for Classification]. Der wichtige Unterschied ist aber, dass ein Diskriminator nur Klassen vorhersagen kann, für die schon Messwerte existieren. Bei allen durchgeführten tatsächlichen Messungen gab es aber keine Auslastung, die der Klasse viel los zuzuordnen ist. Dementsprechend könnte ein Diskriminator diese Klasse auch nicht vorhersagen. Bei einem Regressor mit nachheriger Klassifikation ist das durch Hinzufügen des gewählten Strafterms trotzdem möglich.

Für die Gittersuche sei nun $Q \in \{1, \ldots, 5\}$ und $\lambda \in \{0, 0.2, \ldots, 2\}$. In Tabelle 3.3 sind die sich unter Berücksichtigung von potenziellen Ausreißern ergebenden Regressionsfunktionen aller Gruppen angegeben. Für die Gruppen, für die die Messungen bereits im Intervall $[0, 1]$ waren, ist die Identitätsfunktion $id(x) = x$ die gewählte Abbildungsfunktion. Da Gruppe 4 die Gruppe ist, die aus der *Garderobe AH V* besteht, wurde für diese keine Funktion gewählt.

Gruppe	$f(x) =$
Gruppe 1	x
Gruppe 2	$\sigma(0.23x^3 - 1.54x^2 + 3.39x)$
Gruppe 3	$\sigma(1.08x)$
Gruppe 4	/
Gruppe 5	x
Gruppe 6	$\sigma(-0.04x^2 + 0.91x)$

Tabelle 3.3: Abbildungsfunktionen der Gruppen

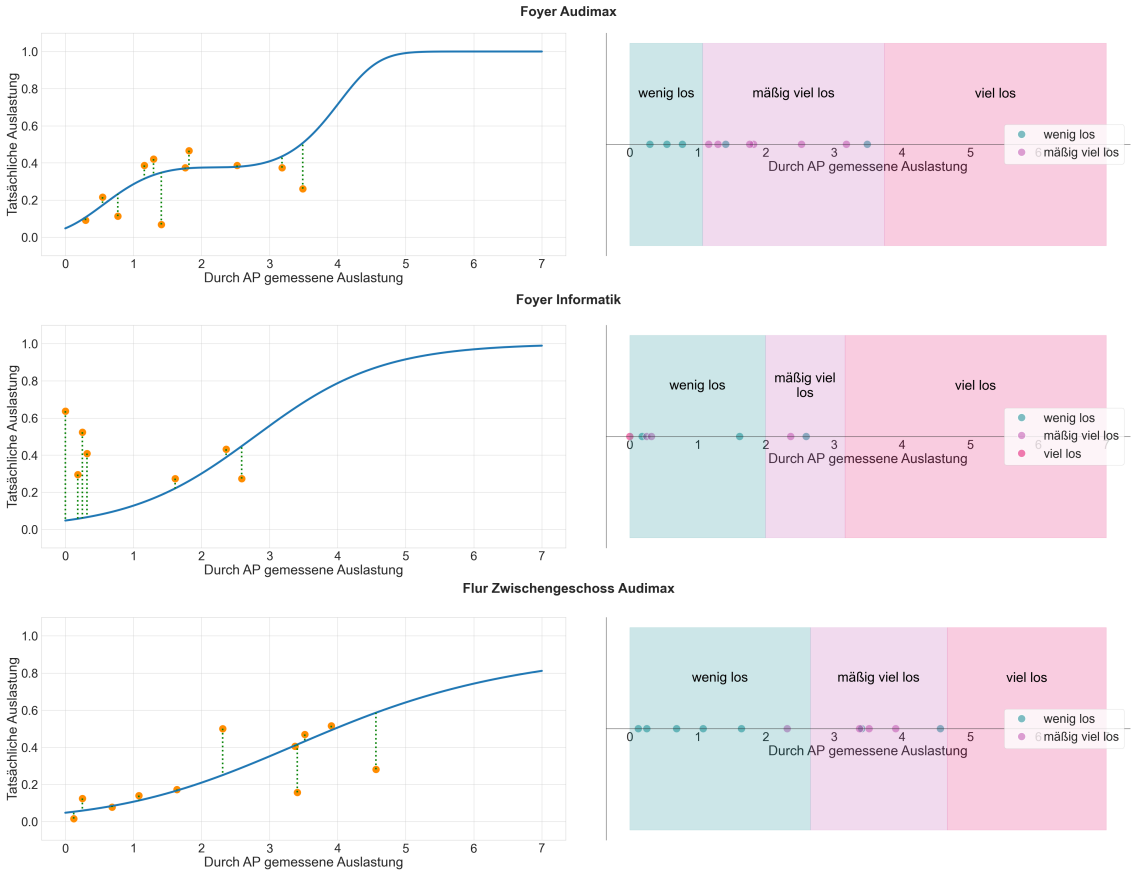

Abbildung 3.2: Lineare Regression und Klassifikatoren der Messungen

Abbildung 3.2 stellt die durch die LOOCV und Gittersuche berechneten Funktionen und die daraus erzeugten Klassifikatoren dar. Es ist zu sehen, dass nur wenige Punkte falsch klassifiziert wurden, welche aber bereits vorher als Ausreißer betrachtet worden sind. Den unterschiedlichen Einfluss von Störungen kann man eindeutig daran erkennen, dass die Klassifikationsbereiche verschieden groß sind.

4 Evaluation

In diesem Kapitel werden die erzeugten Klassifikatoren bewertet. Danach werden Probleme des gewählten Ansatzes diskutiert.

4.1 Bewertung

Um eine Bewertung durchzuführen, werden erneut die ausgewählten Repräsentanten (s. Tabelle 3.2) mehrmals besucht, Messungen der tatsächlichen Auslastung durchgeführt und die wahre Klasse der Auslastung mit der prognostizierten verglichen. Eine Auflistung aller Vergleiche ist in Tabelle 4.1 zu sehen.

Gruppe / Repräsentant	Gemessene Auslastung	Vom Regressor bestimmte Auslastung	Vom Regressor bestimmte Klasse	Tatsächliche Auslastung	Tatsächliche Klasse
Gruppe 2 / *Foyer Audimax*	1.41	0.35	mäßig viel los	0.38	mäßig viel los
	2.26	0.38	mäßig viel los	0.33	mäßig viel los
	0.59	0.18	wenig los	0.41	mäßig viel los
	3.01	0.41	mäßig viel los	0.26	wenig los
Gruppe 3 / *Foyer Informatik*	0.23	0.06	wenig los	0.20	wenig los
	1.45	0.19	wenig los	0.02	wenig los
	0.36	0.07	wenig los	0.14	wenig los
	2.09	0.32	mäßig viel los	0.09	wenig los
Gruppe 6 / *Flur Zwischengeschoss Audimax*	5.88	0.73	viel los	0.47	mäßig viel los
	0.44	0.07	wenig los	0.38	mäßig viel los
	1.41	0.14	wenig los	0.17	wenig los
	0.56	0.08	wenig los	0.14	wenig los

Tabelle 4.1: Test der Modelle

Es ist teilweise eine geringe Abweichung von Vorhersage und der tatsächlichen Auslastung zu erkennen. Bei anderen Messungen ist der Unterschied so groß, dass eine falsche Klassifizierung vorgenommen wird. Eine genaue Auswertung der Fehler kann Tabelle 4.2 entnommen werden.

Gruppe	Durchschnittlicher absoluter Fehler	Durchschnittlicher relativer Fehler	Durchschnittlicher Missklassifikationsfehler
Gruppe 2	0.11	33%	50%
Gruppe 3	0.15	280%	25%
Gruppe 6	0.17	50%	50%

Tabelle 4.2: Fehler der Modelle

Bei der Fehlerbetrachtung ist der Missklassifikationsfehler am wichtigsten, da die Klassen gebildet wurden, um große absolute und relative Fehler auszugleichen und nur die Klasse der Auslastung den Nutzern mitgeteilt werden würde. Es ist zu sehen, dass die Missklassifikationsfehler sehr groß sind. Das könnte daran liegen, dass die wenigen Messungen, die für die Tests der Modelle verwendet wurden, Ausreißer waren oder dass die Modelle generell fehlerbehaftet sind. Nachfolgend werden mögliche Ursachen für die Fehler und Probleme bei der Modellerstellung kurz diskutiert.

4.2 Diskussion

Das *Foyer Audimax* und der *Flur Zwischengeschoss Audimax* liegen im gleichen Gebäude direkt übereinander und sind beide offene Räume. Über spezielle Software [20] wurde die Netztopologie des Audimax untersucht, indem überprüft wurde, mit welchem Access Point ein Gerät verbunden war. Es konnte zwar nicht die interne Namensgebung des Access Point angezeigt werden, weswegen es nicht möglich war, festzustellen, ob der verbundene Access Point auch derjenige des Lernraums war, aber über die BSSID war es möglich Verbindungswechsel nachzuvollziehen. Das Ergebnis: Es kann nicht davon ausgegangen werden, dass der Aufenthalt in einem Raum auch bedeutet, dass man mit den Access Points eben dieses Raumes verbunden ist. Teilweise wurde die Verbindung alle paar Meter gewechselt, wieder in anderen Situationen wurde eine Verbindung lange gehalten, auch nachdem sich in einen anderen Raum bewegt wurde. Das Verhalten der Verbindung lässt sich dementsprechend nur als chaotisch beschreiben. Eine mögliche Folge dieser unberechenbaren Verbindung ist, dass die Räume des Audimax stark in dem Verhältnis von gemessener und tatsächlicher Auslastung variieren. Eine Regression ist nur möglich, wenn die Störung eine normalverteilte Größe ist, was hier nicht der Fall zu sein scheint.

Der gewählte Ansatz hat aber noch weitere Nachteile. In Unterabschnitt 3.3.1 wurde erwähnt, dass der Lernraum *Garderobe AH V* nicht mehr existiert, da der Tisch, der dort stehen sollte, entfernt wurde. So eine Änderung, die nicht auf den Seiten der RWTH dokumentiert wird, kann nicht erwartet werden. Trotzdem wurden für diesen Lernraum Messungen und folglich Schätzungen vorgenommen, die alle auf einer falschen Grundlage basieren.

Des Weiteren wurden die Messungen der tatsächlichen Auslastung der Lernräume des Audimax teilweise zu Zeiten durchgeführt, die bewusst spät am Abend waren und während denen eine große Vorlesung in den Hörsälen im gleichen Gebäude stattgefunden hat. Das wurde einerseits gemacht, damit solche Messungen in der Approximation mit einbezogen werden konnten, aber auch andererseits, um zu verifizieren, dass der Vorlesungsbetrieb einen erheblichen Störfaktor in der Messung darstellt. Diese Messungen sind als die falsch klassifizierten Punkte wiederzufinden (s. Abbildung 3.2). Das *Foyer Informatik* weist dadurch, dass dieser Lernraum direkt neben der Mensa Ahornstraße und neben einem Hörsaal liegt, ein ähnliches Problem auf.

In Anbetracht dieser Faktoren muss die Frage gestellt werden, ob der gewählte Ansatz, also, ob die Auslastung eines Raumes mithilfe der Geräte, die mit einem Access Point verbunden sind, approximiert werden sollte, für die RWTHapp sinnvoll ist. Dass ein Klassifizierungsmodell möglich ist, wurde gezeigt, ob dieses aber im Allgemeinen präzise ist, ist stark anzuzweifeln.

5 Fazit

5.1 Zusammenfassung

In der RWTHapp wird auf Basis der Anzahl von Geräten, die mit einem Access Point verbunden sind, versucht, die Auslastung von Räumen zu schätzen. Besonders in Klausurenphasen ist diese sogenannte Lernraumampel ein viel genutztes Werkzeug von Studierenden. Da die Messungen der Access Points aber anfällig für verschiedene Arten von Störungen sind, wurde in dieser Arbeit versucht, mit Methoden der Felder Data Science und Machine Learning eine genaue Schätzung vorzunehmen.

Dazu wurde zuerst die Gruppe der betrachteten Lernräume über ein hierarchisches Clustering in sechs kleinere Gruppen von äquivalenten Räumen aufgeteilt. Diese Gruppen konnten dann eigenständig betrachtet werden. Danach wurde der Einfluss von Störfaktoren auf diese Gruppen untersucht und vier Gruppen als definitiv störanfällig identifiziert. Mit dieser Erkenntnis konnte ein Verfahren zum Ausgleich dieser Störungen erarbeitet werden, wobei sich für ein Regressionsverfahren entschieden wurde. Damit dieses durchgeführt werden konnte, wurden die Anzahl der Geräte in einem Access Point mit der durch Zählungen herausgefundenen tatsächlichen Auslastung in Verbindung gesetzt. Als Regressionsmodell wurde sich dann für eine logistische Regression mit angepasster Fehlerfunktion entschieden.

Es wurde so ein Verfahren vorgestellt, dass im Sinne der aufgestellten Thesen und Definitionen eine teilweise valide Schätzung der Auslastung liefert, sich generell aber als unpräzise herausgestellt hat.

5.2 Ausblick

Im vorherigen Kapitel wurde erwähnt, dass die größten Störungsquellen das chaotische Verbindungsverhalten der Access Points und der Vorlesungsbetrieb sind. Mehrere Geräte pro Person sind ein Störungsfaktor, der durch Schätzungen und Regression leicht auszugleichen ist. Ein neuer Ansatz müsste also so aufgebaut werden, dass er nicht mehr zu stark an die Access Points gebunden ist.

Eine Möglichkeit dafür ist, nicht die Auslastung eines Raumes, sondern eines Gebäudes anzugeben und das nur während der vorlesungsfreien Zeit. Warum reduziert dieser Ansatz Fehler? Wenn die Auslastung so angegeben wird, ist es egal, ob Geräte in einem falschen Access Point eingewählt sind, da sie sich trotzdem immer noch im gleichen Gebäude befinden. Dadurch, dass keine Vorlesungen stattfinden, können diese die Messungen nicht stören. Beide Störungen würden also eliminiert werden. Warum ist dieser Ansatz sinnvoll? Der Anwendungsfall der Lernraumampel ist, dass geprüft werden kann, ob ein Raum zu voll ist, die eigene Lerngruppe aufzunehmen und so nicht unnötige Wege aufgenommen werden müssen. Wenn nun die Auslastung eines Gebäudes angegeben wird, so können zwei Fälle unterschieden werden

1. das Gebäude ist in der Klasse viel los
2. das Gebäude ist in einer anderen Klasse

Im ersten Fall wird das Gebäude gemieden, da jeder Lernraum höchstwahrscheinlich voll ist. In der jetzigen Variante hätte jeder Raum des Gebäudes diese Klasse gehabt, wodurch das Gebäude auch gemieden worden wäre. Das Ergebnis ist also das gleiche. Im zweiten Fall gibt es innerhalb des Gebäudes, einzelne Räume, die noch Platz bieten. Es muss also nur einer dieser Räume gefunden werden. Innerhalb des Gebäudes einen freien Raum zu suchen ist aber kein Zeitaufwand, der beachtet werden muss. Die Wege zwischen den Gebäuden sind die, die Zeit kosten. Es kann natürlich der Fall auftreten, dass in einem Gebäude noch Platz ist, aber keiner der Räume genug Kapazität für die ganze Lerngruppe bietet. Nichtsdestotrotz sollte das in vielen Fällen kein Problem darstellen, da in öffentlichen Räumen zumeist nicht während des Lernens diskutiert wird, da das die Konzentration der anderen Personen in dem Raum stört, oder sogar von der RWTH selbst untersagt ist (sogenannte Leiselernräume). Aus Erfahrung kann berichtet werden, dass die Lerngruppe eher dafür da ist, dass man die Pausen gemeinsam verbringen kann. Ein solcher Ansatz könnte somit eine Verbesserung bieten.

Wenn am bestehenden Ansatz festgehalten werden soll, ist eine Möglichkeit der Verbesserung, mehr Daten zu Verfügung zu stellen. Aktuell beruht die vorgestellte Regression lediglich auf Messungen der Geräte und der tatsächlichen Auslastung eines Lernraumes. Ein Modell, das aber auch Informationen wie die Uhrzeit der Messung, den aktuellen Vorlesungsbetrieb, eine Schätzung, wie voll verschiedene Vorlesungen sind, die Wetterverhältnisse, etc. zur Verfügung hätte, könnte in der Theorie deutlich präzisere Schätzungen abgeben. Die Komplexität eines solchen Modells sei an dieser Stelle außen vor gelassen. Durch Hinzunehmen von Netzplänen, die die topologische Struktur der Gebäude angeben, könnten auch die Verbindungseigenschaften der Access Points genauer untersucht werden.

Insgesamt zeigt sich, dass Bedarf an weiteren Untersuchungen besteht.

A Literaturverzeichnis

[1] IT Center Help. Rwthapp. https://help.itc.rwth-aachen.de/service/rhtxrjubsj8m/. [Online; Stand 04.11.2024].

[2] Wikipedia. Wireless access point — wikipedia, die freie enzyklopädie, 2024. [Online; Stand 6. Dezember 2024].

[3] Lernräume - rwth aachen university. https://www.rwth-aachen.de/go/id/brok. [Online; Stand 04.11.2024].

[4] Saeed Aghabozorgi, Ali Seyed Shirkhorshidi, and Teh Ying Wah. Time-series clustering – a decade review. *Information Systems*, 53:16–38, 2015.

[5] Cláudia Antunes and Arlindo Oliveira. Temporal Data Mining: an overview. In *Workshop on Temporal Data Mining in the ACM International Conference on Knowledge Discovery and Data Mining (TDM@KDD 2001)*, pages 1–1, August 2001.

[6] The pandas development team. pandas-dev/pandas: Pandas, September 2024.

[7] pandas - python data analysis library. https://pandas.pydata.org/. [Online; Stand 04.11.2024].

[8] Michael L. Waskom. seaborn: statistical data visualization. *Journal of Open Source Software*, 6(60):3021, 2021.

[9] H. Sakoe and S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 26(1):43–49, 1978.

[10] F. Itakura. Minimum prediction residual principle applied to speech recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 23(1):67–72, 1975.

[11] Romain Tavenard. An introduction to dynamic time warping. https://rtavenar.github.io/blog/dtw.html, 2021. [Online; Stand 04.11.2024].

[12] wannesm, khendrickx, Aras Yurtman, Pieter Robberechts, Dany Vohl, Eric Ma, Gust Verbruggen, Marco Rossi, Mazhar Shaikh, Muhammad Yasirroni, Todd, Wojciech Zieliński, Toon Van Craenendonck, and Sai Wu. wannesm/dtaidistance: v2.3.5, January 2022.

[13] Peter J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

[14] Wikipedia. Silhouettenkoeffizient — wikipedia, die freie enzyklopädie, 2023. [Online; Stand 29. Oktober 2024].

[15] Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122, 2013.

[16] J B Tenenbaum, V de Silva, and J C Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, December 2000.

[17] Jiawei Han, Micheline Kamber, and Jian Pei. 3 - data preprocessing. In Jiawei Han, Micheline Kamber, and Jian Pei, editors, *Data Mining (Third Edition)*, The Morgan Kaufmann Series in Data Management Systems, pages 83–124. Morgan Kaufmann, Boston, third edition edition, 2012.

[18] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006.

[19] Prof. Dr. rer. nat. Stephan Bialonski. Machine learning, 2024.

[20] Jiri Trechet. Network analyzer, February 2016.

B Tabellenverzeichnis

1.1 Liste der Lernräume, deren Auslastung in der RWTHapp angezeigt wird [3] 2

2.1 Ausschnitt aus den Daten der verbundenen Geräte 4
2.2 Bewertung der Strukturierung der Cluster durch den Silhouettenkoeffizienten [14] . 8
2.3 Gruppierung der Lernräume nach gleichem Verhalten 11

3.1 Bereiche der verschiedenen Klassen 13
3.2 Repräsentanten der Gruppen 15
3.3 Abbildungsfunktionen der Gruppen 20

4.1 Test der Modelle 22
4.2 Fehler der Modelle 22

C Abbildungsverzeichnis

2.1 Auslastung der Lernräume 5
2.2 Beispielhafte Visualisierung der Abstände zwischen zwei Zeitreihen (Punkte, die miteinander verglichen werden, sind verbunden) 7
2.3 Grafische Darstellung der Silhouettenkoeffizienten beim hierarchischen Clustering . 9
2.4 Einteilung der Zeitreihen in zwei Gruppen 9
2.5 Visualisierung der Zeitreihen durch Isomap 10
2.6 Einteilung der Zeitreihen in sechs Gruppen 11

3.1 Durch Access Points gemessene Auslastung im Vergleich mit der tatsächlichen . . . 16
3.2 Lineare Regression und Klassifikatoren der Messungen 21

D Abkürzungsverzeichnis

DTW Dynamic Time Warping

LOOCV Leave-One-Out Kreuzvalidierung