Ein „Deep Embedded Clustering Framework" für Soundscapes in Wohnräumen und dessen empirische Auswertung der affektiven Beurteilung

---

Studiengang: Ton und Bild (B. Eng.)

Kurs: Audio Data Science

Betreuer: Prof. Dr. Jochen Steffens, Prof. Dr. Florian Huber

Studenten:	Fietje Schlegelmilch, Tim Müller
Matrikelnummer:	*869797, 875462*
Semester:	9./7. Semester
E-Mail:	fietje.schlegelmilch@study.hs-duesseldorf.de
	t.mueller@study.hs-duesseldorf.de
Telefon:	+49 174 9494004/+49 176 47265746
Datum:	25.09.2024

Inhaltsverzeichnis

I. Abkürzungsverzeichnis

IEEE ..... Institute of Electrical and Electronics Engineers
ISO ..... International Organization for Standardization
L3-Net ..... *Look, Listen and Learn Net*
OpenL3 ..... *Look, Listen and Learn*
RPCA ..... Robust principal component analysis
t-SNE ..... T-distributed Stochastic Neighbor Embedding

II. Abbildungsverzeichnis

1 Einführung

Unüberwachtes maschinelles Lernen ist ein Ansatz, große und „ungelabelte“ Datensätze in Cluster zu unterteilen und so eventuelle Zusammenhänge zu erkennen. Aufgrund der komplexen Daten stellt diese Herangehensweise im Audiobereich immer noch diverse Herausforderungen dar (Jason Cramer, 2019). In dieser Arbeit soll ein unüberwachtes Clustering eines Datensatzes vorgenommen werden, um die Grundlagen und Grenzen dieser Methode im Audiobereich zu untersuchen. Den Ausgangspunkt hierzu bildet ein Datensatz, der im Rahmen einer Studie an der Hochschule Düsseldorf erhoben wurde, welcher aus 6594 kurzen Tonaufnahmen aus dem Zuhause verschiedener Probanden besteht.

Die Untersuchungen basieren unter anderem auf Aspekten der Soundscape Forschung: ,,Soundscape ist eine akustische Umgebung, die durch eine Person oder eine Gruppe von Menschen im Kontext wahrgenommen, erfahren und/ oder begriffen wird” (ISO 12913-1, 2014). Ziel dieser Arbeit ist es, den zugrunde liegenden Datensatz zu clustern und die Clusterergebnisse semantisch zu bewerten und in den Kontext der von den Probanden angegebenen Soundscape Parameter zu setzen.

Für das Clustering orientieren wir uns in Teilen an dem Vorgehen des IEEE Papers *„Multiview Embeddings for Soundscape Classification“ (*Dhanunjaya Varma Devalraju und Padmanabhan Rajan, 2022). Hier werden durch das vortrainierte Neurale Netz „OpenL3“ die Embeddings aus den Clusterdateien extrahiert und dann mithilfe des k-means Algorithmus geclustert. Es stellt sich die Frage, wie gut das Clustering auf Grundlage eines vortrainierten Netzwerks funktioniert. Außerdem soll untersucht werden, ob eine Trennung der Signale in „Vordergrund“ und „Hintergrund“ das Clustering verbessert und die semantische Analyse erleichtert. Am Ende werden die Ergebnisse empirisch ausgewertet, um mögliche Zusammenhänge zwischen der affektiven Beurteilung der Probanden und der Clusteranalyse zu betrachten.

2 Der Datensatz

Der zugrunde liegende Datensatz stammt aus einer Studie mit dem Titel „*Extensive crowdsourced dataset of in-situ evaluated binaural soundscapes of private dwellings containing subjective sound-related and situational ratings along with person factors to study tim-varying influences on sound perseption*“, die an der Hochschule Düsseldorf in Kooperation mit der Technischen Universität Berlin durchgeführt wurde. Ziel dieser Studie war es, auf Grundlage der Soundscape Forschung, potenziell relevante Klänge im natürlichen Umfeld der Probanden aufzuzeichnen und von diesen bewerten zu lassen (Siegbert Versümer, 2023, S. 2).

Grundlage dieser Bewertung ist die standardisierte Soundscape Bewertung, die die Probanden in Form eines Fragebogens ausfüllen mussten. Außerdem wurden Fragen über die wahrgenommene Lautheit sowie das von ihnen als salient angesehene Geräusch gestellt (Siegbert Versümer, 2023, S. 2). Die Art der Datenerhebung basiert auf der sogenannten *„Expirience Sample Method“* (Siegbert Versümer, 2023), bei der es darum geht, *„die Gefühle, Gedanken, Handlungen, den Kontext und/oder die Aktivitäten der Teilnehmer wiederholt* [zu messen]*, während sie ihrem täglichen Leben nachgehen“* (Sabrina Zirkel, 2015, S. 1). Insgesamt haben 105 Probanden teilgenommen, welche in Summe 6594 Situationen in ihrem gewohnten Zuhause aufgezeichnet und bewertet haben (Siegbert Versümer, 2023, S. 2)

3 Clustering mit k-means und t-SNE Plot

Als Clustering bezeichnet man das Auffinden von Untergruppen bzw. Clustern in einem Datensatz. Man versucht also, die verschiedenen Observationen eines Datensatzes so zu sortieren, dass sich die Observationen, die sich im selben Cluster befinden, sehr ähnlich sind, während sich die Observationen in unterschiedlichen Clustern deutlich voneinander unterscheiden (Gareth James, 2023, S. 520).

Hierbei handelt es sich um ein unüberwachtes Lernen („unsupervised learning“), da versucht wird, Daten ohne Label in unspezifizierte Kategorien zu sortieren (IBM , 2024). Das hier verwendete Clustering-Verfahren ist der sog. k-means Algorithmus.

3.1 K-means Clustering

Bei dem k-means Clustering handelt es sich um ein sog. nicht-überlappendes bzw. exklusives Clustering. Das bedeutet, dass jeder Datenpunkt nur einem bestimmten Cluster zugeordnet wird (IBM , 2024). Die Grundidee des k-means Clustering besteht darin, das Ausmaß, in dem sich die verschiedenen Datenpunkte in ein und demselben Cluster unterscheiden, so gering wie möglich zu halten.

3.1.1 Quadratische Euklidische Distanz

Der Wert bzw. die Operation, die hierbei verwendet wird, ist die sog. quadratische euklidische Distanz aller Clusterpunkt zum Cluster Mittelpunkt.

$$W(C_k) = \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2$$

Die eigentliche quadratische Distanz steckt in dem Term,

$$(x_{ij} - x_{i'j})^2$$

in dem zwei Clusterpunkte x_{ij} und $x_{i'j}$ voneinander abgezogen werden. „j“ steht dabei für die zu aufsummierende Dimension des zugrunde liegenden Datenpunktes, „i“ für die verschiedenen Punkte im “k-ten“ Cluster.

$$\sum_{j=1}^{p} (x_{ij} - x_{i'j})^2$$

Es wird also zuerst über alle Dimensionen der beiden Punkte aufsummiert, um deren tatsächlichen Abstand zu erhalten.

$$\sum_{i,i' \in C_k} \sum_{j=1}^{p} \left(x_{ij} - x_{i'j}\right)^2$$

Diese Operation wird dann für alle möglichen Punktpaare durchgeführt und aufsummiert.

$$W(C_k) = \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} \left(x_{ij} - x_{i'j}\right)^2$$

Dieses Ergebnis wird dann durch die Summe aller Punkte im Cluster geteilt.

Die Herausforderung besteht nun darin, diese Summe aller quadratischen euklidischen Abstände durch die Anpassung der Clusteranzahl so zu optimieren, dass jedes Cluster die kleinstmöglichen quadratischen euklidischen Abstände innerhalb der dem Cluster zugehörigen Punkte aufweist. Diese Operation wird im Pythoncode (siehe 9.3) durch die Funktion „Inertia“ ausgedrückt und später zur Bewertung der Clusterergebnisse genutzt (siehe Kapitel 5).

3.1.2 Silhouette Score

Bei der Optimierung der Clusteranzahl in dieser Arbeit ergänzt der Silhouette Score $s(x_i)$ die quadratische euklidische Distanz.

$$s(x_i) = \frac{b(x_i) - a(x_i)}{\max\{a(x_i), b(x_i)\}}$$

In der Formel steht $a(x_i)$ für den durchschnittlichen Abstand des Datenpunkts x_i zu den restlichen Datenpunkten im Cluster, zu dem x_i zugeordnet wurde. Dagegen steht $b(x_i)$ für den durchschnittlichen Abstand von x_i zu den Punkten im nächstgelegenen Cluster, zu dem x_i aber nicht gehört. Der Term $\max\{a(x_i), b(x_i)\}$ verwendet den größeren der beiden Werte im Zähler und ist dazu gedacht, den Wert von $s(x_i)$ auf einer Skala von -1 bis 1 zu skalieren (Jian Chen, 2019, S. 9).

SILHOUETTE SCORE	**1**	**0**	**-1**
INTERPRETATION	Punkt ist näher an seinem eigenen Cluster als an dem nächstgelegenen	Punkt ist gleichweit von eigenem Cluster wie von nächstgelegen Cluster entfernt.	Punkt ist näher an anderem Cluster als an seinem eigenen

Abbildung 1- Interpretation Silhouette Score

3.2 t-distributed Stochastic Neighbor Embedding (t-SNE)

Um die hochdimensionierten Datenpunkte bzw. deren „Embeddings“ grafisch darstellen zu können, müssen wir die Dimensionen der zugrunde liegenden Daten auf zwei Dimensionen reduzieren, mit dem Ziel, die signifikante Struktur der hochdimensionalen Daten so weit wie

möglich in der niederdimensionalen Visualisierung zu erhalten (Laurens van der Maaten, 2008, S. 2). Um dies zu erreichen, wird in dieser Arbeit das t-distributed Stochastic Neighbor Embedding (kurz und im Folgenden t-SNE) – Verfahren angewendet.

Die Grundidee des t-SNE Verfahrens ist es, die Abstände der Datenpunkte im hochdimensionalen Raum in eine bedingte Wahrscheinlichkeit $p_{j|i}$ umzuwandeln. Das heißt, je näher die Datenpunkte im hochdimensionalen Raum zusammenliegen oder anders ausgedrückt, je ähnlicher sich die beiden Datenpunkte sind, desto höher ist die Wahrscheinlichkeit, dass diese beiden Punkte auch in einer niedrigeren Dimension Nachbarn sind (Laurens van der Maaten, 2008, S. 2).

$$p_{j|i} = \frac{\exp\left(-\frac{|x_i - x_j|^2}{2\sigma_i^2}\right)}{\sum_{k \neq i} \exp\left(-\frac{|x_i - x_k|^2}{2\sigma_i^2}\right)}$$

Grundlage für die Wahrscheinlichkeit ist die quadratische euklidische Distanz (4.1.1) $|x_i - x_j|^2$, welche durch die Varianz $2\sigma_i^2$, die den Einfluss weiter entfernter Punkte auf $p_{j|i}$ steuert, geteilt wird.

4 Feature-Extraktion und Signaltrennung

Um nun den Datensatz, der aus Audiodateien besteht, mit den in Kapitel 3 beschriebenen Methoden zu clustern, müssen sog. High-Level Features aus den zu komplexen und großen Daten extrahiert werden. Bei dieser Feature Extraktion geht es darum, eine sehr kompakte und informative Gruppe von Merkmalen zu finden, um die Effizienz der Datenspeicherung und -verarbeitung zu verbessern (Kacprzyk, 2006, S. 8). Diese Daten werden in Form eines Vektors dargestellt, um die oben beschrieben mathematischen Operationen anwenden zu können (Kacprzyk, 2006, S. 8). Einen solchen Vektor nennt man Embedding.

Das „Look, Listen, and Learn“ (L3-Net) ist ein vortrainiertes Neuronales Netzwerk. Für die Feature-Extraktion in der Bearbeitungskette dieser Arbeit wird die OpenL3- Python Bibliothek verwendet. Dies ist eine Open-Source-Implementierung des L3-Netzwerks, das für die Erstellung tiefer Audio- und Bild-Embeddings entwickelt wurde (Dhanunjaya Varma Devalraju, 2022, S. 5). Aus jedem Abtastwert einer Audiodatei des Datensatzes wird ein gemitteltes Embedding extrahiert, das durch einen Spaltenvektor der Dimension 6144 x 1 beschrieben wird (Dhanunjaya Varma Devalraju, 2022, S. 5).

Vorder- und Hintergrundtrennung

Zunächst werden dabei die 6549 binauralen Aufnahmen des Datensatzes monoisiert und normalisiert. Diese Dateien werden im fortlaufenden als „Summe“ bezeichnet. Durch Verarbeitung dieser Dateien in Matlab wird die von (Dhanunjaya Varma Devalraju, 2022, S. 6) vorgeschlagene Vordergrund- und Hintergrundtrennung durch eine RPCA errechnet. Diese Dateien werden im fortlaufenden als ,,Vordergrund“ und ,,Hintergrund“ bezeichnet. Abschließend erfahren alle drei Audiogruppen ein Downsampling auf 22 kHz. Für alle Dateien der verschiedenen Gruppen werden nun die Embeddings errechnet. Dabei wird darauf verzichtet, die Vordergrund- und Hintergrund Embeddings in einem weiteren neuronalen Netzwerk zu fusionieren. Die Ergebnisse des Clustering für die verschiedenen Perspektiven (Summe, Vordergrund, Hintergrund) werden fortlaufend einzeln betrachtet.

5 Durchführung

Ein wichtiger Schritt für ein erfolgreiches Clustering mit dem k-means Verfahren ist die Verwendung einer geeigneten Clusteranzahl für den k-means Algorithmus. Dafür werden zunächst die Kenngrößen Inertia und Silhouette Score herangezogen. Die Abbildungen 2 und 3 zeigen die Graphen für diese Werte in einzelnen Diagrammen für die Dateien der Summe über eine Clusteranzahl von 2-20. Dabei wird der sogenannte “Kneepoint” bestimmt, also die Stelle, bei der die Abnahme der euklidischen Distanz pro Clusterzunahme stark abflacht. In dem Inertiaplot der Summe ist dies nicht sehr eindeutig abzulesen, aber zwischen der Clustzeranzahl 5 und 6 anzunehmen.

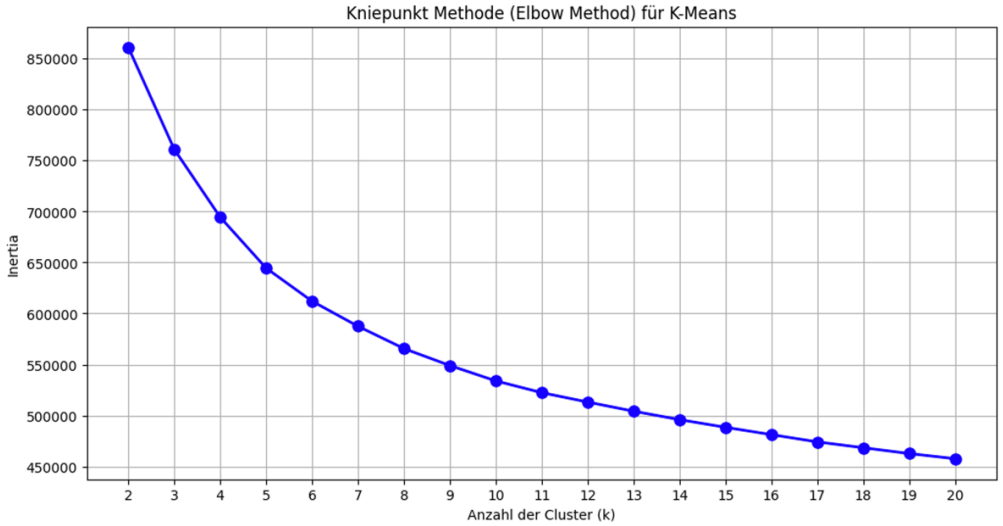


Abbildung 2 - Inertia der Cluster von k = 2 bis 20

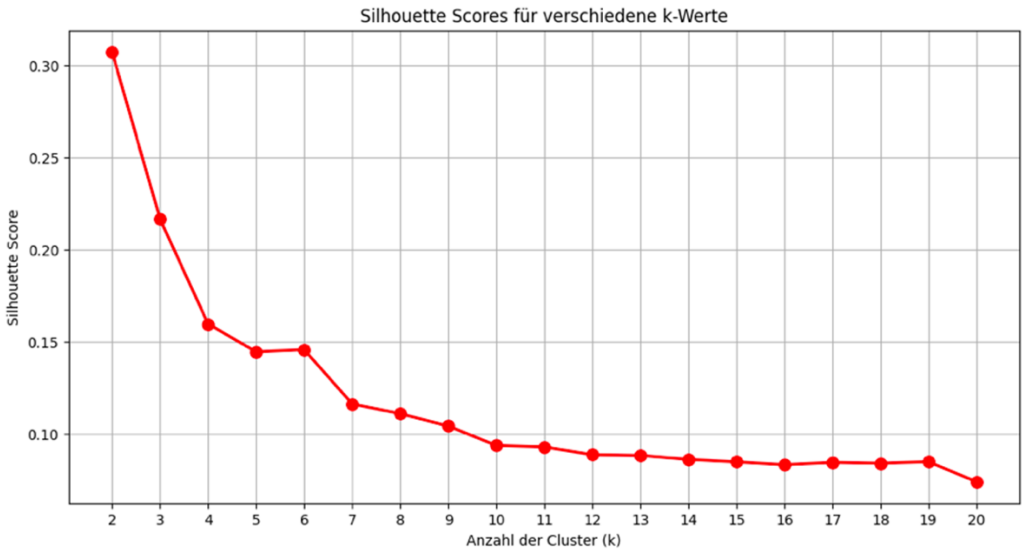


Abbildung 3 - Silhouette Score Werte der Cluster von k = 2 bis 20

Zunächst wird bei der Clusterbestimmung lediglich mit der Summe gearbeitet, da diese eine zuverlässige Referenz darstellt. Wie in Abbildung 3 zu sehen ist, verspricht der Anstieg im Silhouette Score bei der Clusteranzahl 6 ein gutes Ergebnis, wobei dieses auch in etwa mit dem Kneepoint der Inertiakurve übereinstimmt.

Der Silhouette Score von ca. 0.15 gilt allgemein nicht als optimaler Wert für ein Clustering. Im Kontext unseres Datensatzes gehen wir trotzdem von einem ausreichend hohen Wert, für die weitere Analyse aus.

Die bestehende Probe wird im nächsten Schritt über Hörtests der 5 nächsten Datenpunkte pro Clusterzentrum geprüft. Im Anhang befindet sich eine Dokumentation der Auswertungsergebnisse sowie einer subjektiven Bewertung. Wir kommen zu dem Ergebnis, das die Clusteranzahl 6 (Label 0-5) für die Summe am geeigneten ist und die besten Ergebnisse verspricht. Im Unterschied zum Clustering mit k=5, gibt es hier ein Cluster für die Kategorie Musik. Die Tabelle (Abbildung 4) beschreibt die semantische Interpretation und benennt die jeweiligen Clusterlabel.

CLUSTER	0	1	2	3	4	5
BENENNUNG	**Sprache**	**Sprache**	**Haushalt**	**Background**	**Straßenverkehr**	**Musik**
SPEZIFIZIERUNG	im Hintergrund	im Vordergrund	Geräte, auffällige Geräusche	Umgebungsgeräusche, Normale Hintergrundkulisse, vereinzelt Impulse		Musik im Vordergrund als auch Hintergrund,
BEISPIELE	Menschen, Fernseher, Umgebungsgeräusche	Menschen, Fernseher, kaum Umgebungsgeräusche	Kaffeemaschine, Wasser etc.	Rauschen, Wind Entfernte Geräusche (Piepen, Vögel, Hämmern)	Vorbeifahrende Autos	Aktives Musikhören, Musik im Kontext von anderen Medien

Abbildung 4 – Bezeichnungen für das Clustering von k = 6 der Summe

6 Auswertung

In diesem Kapitel sollen die Ergebnisse der Untersuchung im Hinblick auf die gestellten Forschungsfragen ausgewertet werden: Welchen Erfolg verspricht ein vortrainiertes neuronales Netzwerk in Hinblick auf das Clustering eines großen Datensatzes? Verspricht eine Quellentrennung der Summe eine Verbesserung dieses Ergebnisses? Und im Weiteren: Können mittels empirischer Auswertung Affekttendenzen der Probanden zwischen den Clustern gefunden werden?

6.1 Clusteranalyse

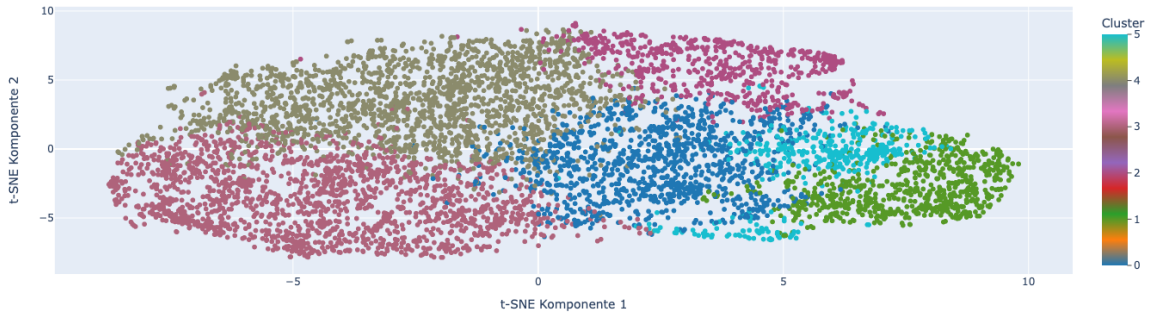


Abbildung 5 - t-SNE Plot des Clustering für k = 6 der Summe

Das Clustering der Summe wird als t-SNE Plot (Abbildung 5) dargestellt. Betrachtet man zunächst lediglich die quantitative Zuordnung der 6594 Audioszenen auf die 6 Cluster (Abbildung 6), erkennt man die höchste Zuordnung bei dem Cluster 4 – Verkehr (n = 1907) und bei dem Cluster 3 – Umgebung (n = 1793). Die niedrigsten Zuordnungen haben Cluster 5 – Musik (n = 426) und Cluster 2 - Haushalt (n = 611). Dies lässt den Schluss zu, dass der überwiegende Anteil der Aufnahmen eine normale Alltagsszene innerhalb der Wohnung abbildet und speziellere Geräusche wie Musik und Haushaltsgeräusche im kleineren Teil vorkommen.

Cluster 0 - Sprache (n = 1111) und Cluster 1 - Sprache (n = 746) bilden ähnliche Szenen ab, wobei Cluster 1 ca. 350-mal weniger vorkommt. Nach einem Hörtest der 5 Datenpunkte, die am nächsten zum Clusterzentrum liegen, kommen wir zu der Annahme, dass Cluster 0 eher Sprachanteile im Hintergrund und Cluster 1 eher vordergründige Sprache zuordnet.

Deskriptivstatistik

	Label	ID
N	0	1111
	1	746
	2	611
	3	1793
	4	1907
	5	426

Abbildung 6 - Anzahl der Daten pro Cluster / "Label"

Eine Herausforderung ist die Unterscheidung zwischen menschlicher Interaktion und Sprache, die von anderen Medien wie Fernseher und Radio stammt. Für jede Aufnahme haben die Probanden jeweils das für sie auffälligste Geräusch mit angegeben. Filtert man in Cluster 0 nach den Schlagwörtern „Fernseher“ und „TV“ erhält man 104 Fälle, in denen dies zutrifft. Relativ entspricht das etwa 10,26 %. Bei analogem Vorgehen für Cluster 1 findet man 253 Fälle, in dem der Fernseher das saliente Geräusch war. Dies entspricht relativ betrachtet 33,91 %. An diesem Beispiel werden Unterschiede in der Salienzbewertung der beiden Cluster deutlich. Es kann also nicht von einer komplett erfolgreichen Trennung von menschlicher - und medialer Interaktion ausgegangen werden. Ob man Cluster 0 und 1 daher sinnvoll zusammenfassen könnte, bleibt offen.

Grafische Analyse

Bei der grafischen Analyse des t-SNE Plots (Abbildung 5) ist zunächst eine farbliche Zuordnung der Summendateien für ihre Cluster zu sehen. Es gibt kaum Freiraum zwischen den Clusterbereichen und daher keine klare Abgrenzung. Dies bestätigt den moderaten Silhouette Score von ca. 0,15. Ein anderes Ergebnis wäre für solch einen realistischen Datensatz zunächst auch nicht zu erwarten, da viele Audioszenen am gleichen Ort und im gleichen Rahmen aufgenommen wurden. Innerhalb der 15s Aufnahmezeit können demnach auch mehrere saliente Geräusche auftreten und besonders im Randbereich zu Überschneidungen und nicht eindeutigen Zuordnungen in Kategorien führen. Im Rahmen dieser Einschränkung erwarten wir jedoch eine aussagekräftige Clusteranalyse.

Man erkennt eine Nähe der Cluster 0 und 1 in den Farben dunkelblau und grün. In diesem Übergangsbereich liegt das Cluster 5 – Musik (hellblau). Dass Musik ebenfalls Sprachanteile beinhalten kann und das Medium (z.B. Radio, Fernseher) anscheinend in allen 3 Kategorien vorkommt, könnte eine Erklärung für diese Überschneidungen sein.

Die beiden größten Cluster Hintergrund (rot) und Verkehr (braun) liegen ebenfalls aneinander. Womöglich ähneln sich diese Soundkulissen in den Randbereichen zunehmend. Aufgrund dessen wollen wir im fortlaufenden mit einer qualitativeren Stichprobe arbeiten. Für die Hörtests begrenzen wir die Auswahl auf die 5 nächsten Audioszenen zu jedem Clusterzentrum. Diese werden über eine Minimumsfunktion des euklidischen Abstands bestimmt.

Auditive Analyse

Durch das Erstellen eines Hörtestprotokolls beobachten wir mindestens 4 von 5 semantisch sinnvollen Zuordnungen für jedes Cluster. Dabei schneiden die Cluster mit Verkehr und Musik besonders gut ab. Das Cluster 3 – Background ist dabei schwieriger zu beurteilen, da die

Grundstimmung gleich ist, jedoch unterschiedliche saliente Geräusche auftreten können (siehe 9.1.7).
Nach einer Erweiterung des Hörtests auf die 10 nächsten Clusterdateien und erfolgreicher Überprüfung schätzen wir den Zuordnungserfolg des verwendeten Algorithmus im begrenzten Umfeld zum jeweiligen Clusterzentrum auf etwa 80 %.

Bei der Durchführung von den Summenhörtests über eine variable Clusteranzahl konnte eine starke Schwankung des Ergebnisses festgestellt werden. Im Vergleich der Clusteranzahlen 3 - 7 haben die niedrigeren Clusteranzahlen bereits semantisch sinnvolle Ergebnisse gezeigt z.B. Im Cluster 3 (*siehe 9.1.1*) mit den Kategorien Außengeräusche, Background und Sprache. Jedoch schienen diese noch nicht die Komplexität des Datensatzes in Gänze abbilden zu können. Vergleicht man die Clusteranzahl 7 (*siehe 9.1.10)* mit den in der Arbeit verwendeten 6 Clustern (*siehe 9.1.7)*, so bleiben zwar die Grundkategorien präsentiert, haben in Clusteranzahl 7 jedoch eine deutlich höhere Vermischung und scheinbar nicht so eine hohe Erfolgsquote. Dies lässt vermuten, dass die Anzahl der Cluster eine sehr hohe Auswirkung auf das Ergebnis hat und diese Art von Hörtests einen geeigneten Ansatz für die Wahl der Clusteranzahl darstellen können.

6.2 Quellentrennung

Bei der Untersuchung der quellengetrennten Audiosignale wurden die vorverarbeiteten Vordergrund- und Hintergrunddateien dem Algorithmus zugeführt und anschließend das Clustering über die variable Clusteranzahl verglichen. Die zu untersuchende Annahme bestand darin, dass die vordergründigen Signale womöglich mehr saliente Geräusche beinhalten, die für die Hörwahrnehmung am bedeutendsten sein könnten. Dadurch könnte man sich ein noch präzisere Clustering versprechen.

Im Hörtest wurden die 5 zum Clusterzentrum nächsten Dateien des jeweiligen Clustering (Vordergrund / Hintergrund) verglichen, jedoch wurden dabei die Audiodateien der Summe abgespielt. Dies ermöglicht die Zuordnung des Kontexts der Szene bei der Beurteilung und hilft, die Szenen zu vergleichen. Zudem verhindert dieses Vorgehen eine verzerrte Beurteilung, die nicht der Realität entspricht und erleichtert die akustische Bewertung, da mögliche Artefakte der Quellentrennung ausgeschlossen sind. Wir analysieren vorrangig die Clusteranzahl 6 (siehe 9.1.7), da diese bei den Summendateien die besten Ergebnisse lieferte.

6.2.1 Vordergrund

Die Kategorien des Vordergrunds ergaben eine Aufteilung auf Haushalt, Fernseher/Sprache, Verkehr, Background und zwei Cluster, bei denen kein klares Muster erkennbar war. Die Trefferquote der Zuordnung scheint zudem wesentlich schlechter als bei der Summe zu sein. Die Kategorie Fernseher wurde jedoch sehr gut erkannt und zugeordnet, obwohl die entsprechenden Audioanteile im Original

meist spektral und dynamisch verdeckt auftreten. Eine ähnliche Beobachtung wurde bei der Clusteranzahl 3 gemacht (siehe 9.1.2), der sehr leise Sprache zugeordnet werden konnte. Bei dieser geringen Clusteranzahl war ebenfalls auffällig, dass zudem sehr spezielle Klänge wie hochfrequente Liegetöne und andernfalls eher tieffrequente Geräusche geclustert wurden.

Für diese speziellen Fälle (verdeckte Sprache, besondere Frequenzen) könnte eine Vordergrundtrennung für das Clustering eventuell nützlich sein und bietet Anlass für weitere Forschung. Für die Analyse des vorliegenden Datensatzes verspricht das Verfahren aufgrund der niedrigen semantischen Trefferquote keine guten Ergebnisse.

Grund dafür könnte ein nicht optimiertes Trennungsverfahren, zu drastische Verzerrungen oder eine nicht optimierte Clusteranzahl sein. Außerdem sind oft nur sehr kurze, sporadische Audioimpulse hörbar, die teilweise nicht vom Menschen deutbar sind und dementsprechend möglicherweise auch dem neuronalen Netz zu wenig Information geben.

6.2.2 Hintergrund

Diese Beobachtungen werden ebenfalls durch die akustische Bewertung des Hintergrunds mit 6 Clustern unterstützt (siehe 9.1.9). Der überwiegende Teil der Audiodateien entspricht im Höreindruck stark den Summendateien. Manchmal sind saliente Geräusche leiser vorhanden und seltener komplett aus der Hintergrunddatei entfernt. Die naheliegende Annahme, dass das Clustering des Hintergrunds ähnliche Ergebnisse wie das der Summe liefert, wird durch Hörtests bestätigt: Die Kategorien Haushalt, Sprache, Musik/Sprache, Verkehr decken sich. Statt zwei Sprachclustern gibt es hier aber zwei Backgroundcluster, von denen eines der Summe ähnelt und das andere vorrangig aus Rauschen besteht. Die Annahme, dass Musik aus „Sprache mit Musik“ herausgerechnet wird oder der Vordergrund aus der Rauschkategorie herausgerechnet wurde, kann nicht bestätigt werden. Die Trefferquote ist im Allgemeinen ähnlich gut wie die der Summe. Das Verfahren verspricht im Rahmen dieser Auswertung keinen merklichen Mehrwert und birgt eher die Gefahr des Informationsverlusts.

6.3 Empirische Auswertung

Für die Untersuchung der affektiven Beurteilung der geclusterten Soundscapes beziehen wir uns auf das von (Östen Axelsson, 2020) vorgeschlagene zweidimensionale Modell mit den Dimensionen *eventfulness* und *pleasantness*. Dieses gilt nach (Davies, 2012) als Standardmodell für die wahrgenommenen Dimensionen von Soundscapes.

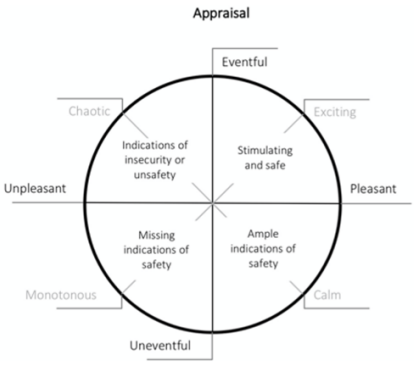


Abbildung 7 - Beurteilung einer Umgebung in Form eines Circumplex Modells (Axelsson, 2012)

In der von uns verwendeten Dokumentation werden die beiden Dimensionen auf einen Wertebereich von -10 bis 10 skaliert.

Bei der empirischen Auswertung fokussieren wir uns auf die 100 nächsten Dateien zu den Clusterzentren der Summenbetrachtung. Dadurch wird eine aussagekräftige Stichprobe von 6 mal 100 erreicht, die somit 600 Daten beinhaltet und etwa 10 % der Grundgesamtheit abbildet. Diese Vorgehensweise und das damit einhergehende Vermeiden von Ausreißern verspricht allgemeine Aussagen über den Erfolg des Clustering und den Vergleich der affektiven Beurteilung treffen zu können.

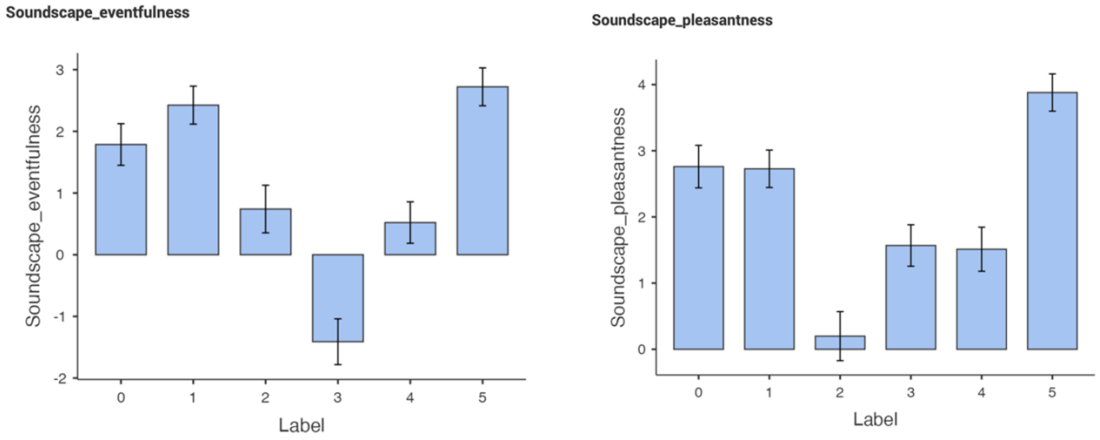


Abbildung 8 - Mittelwerte von eventfulness (links) und pleasantness (rechts) nach Cluster - Balkendiagramm

Deskriptivstatistik

	Label	Soundscape_eventfulness	Soundscape_pleasantness
N	0	100	100
	1	100	100
	2	100	100
	3	100	100
	4	100	100
	5	100	100
Mittelwert	0	1.79	2.76
	1	2.43	2.73
	2	0.741	0.198
	3	-1.41	1.57
	4	0.521	1.51
	5	2.72	3.88

Abbildung 9 - Mittelwerte von eventfulness und pleasantness nach Cluster - Tabelle

6.3.1 Signifikanztest und Effektstärke

Für die Analyse werden die Mittelwerte der Dimensionen *pleasantness* und *eventfulness* pro Cluster (genannt „Label“) berechnet und verglichen. In der Tabelle (Abbildung 8) werden die dabei entstandenen Werte aufgezeigt und durch das Balkendiagramm (Abbildung 9) visualisiert. Auf den ersten Blick können einige Tendenzen vermutet werden. Bei der *eventfulness* scheint es große Unterschiede zum Minimum des Cluster ,,3 – Background“ (-1,41) zu geben. Im Bereich der *pleasantness* besteht die größte Differenz zwischen dem Cluster 2 – Haushalt (0,198) und dem Cluster – Musik (3,88). Um die statistische Signifikanz des Einflusses des Clusters auf die Mittelwerte der beiden Soundscape Dimensionen zu prüfen, wird eine Varianzanalyse herangezogen. Für *pleasantness* sowie *eventfulness* ergibt sich jeweils ein p - Wert <0.01, weshalb ein Zufallsbefund ausgeschlossen werden kann. Um die Effektstärke dieser Abhängigkeit zu berechnen, wird ein lineares Modell verwendet, das den Einfluss der persönlichen Bewertungstendenzen der insgesamt 105 Probanden aus dem Ergebnis herausrechnet. Für *pleasantness* wird eine reine Effektstärke R^2 von 12% berechnet, die sich durch das Ausgleichen der Benutzer-ID auf $R^2_{conditional}$ = 41,9% erhöht. Für *eventfulness* ergeben diese Werte zunächst R^2 = 12 % und unter Herausrechnung der Person $R^2_{conditional}$ = 34,4%.

Somit scheint es einen mittel starken Einfluss des Clustering auf die affektive Beurteilung der Soundscapes zu geben. Diesen wird im Folgenden genauer untersucht.

ANOVA

ANOVA - Soundscape_pleasantness

	Quadratsumme	df	Mittlere quad. Abw.	F	p	η²
Label	824	5	164.8	16.2	< .001	0.120
Residuen	6033	594	10.2			

[4]

ANOVA

ANOVA - Soundscape_eventfulness

	Quadratsumme	df	Mittlere quad. Abw.	F	p	η²
Label	1162	5	232.4	19.9	< .001	0.143
Residuen	6951	594	11.7			

Abbildung 10 - Anova der Werte pleasantness und eventfulness

Model Info

Info	
Estimate	Linear mixed model fit by REML
Call	Soundscape_pleasantness ~ 1 + Label+(1 | Benutzer_ID)
AIC	3008.606
BIC	3046.132
LogLikel.	−1497.478
R-squared Marginal	0.120
R-squared Conditional	0.419
Converged	yes
Optimizer	bobyqa

Info	
Estimate	Linear mixed model fit by REML
Call	Soundscape_eventfulness ~ 1 + Label+(1 | Benutzer_ID)
AIC	3128.976
BIC	3165.459
LogLikel.	−1557.142
R-squared Marginal	0.122
R-squared Conditional	0.344
Converged	yes
Optimizer	bobyqa

Abbildung 11 - Gemischt-Lineare Regression für pleasantness (links) eventfulness (rechts)

6.3.2 Post Hoc Test und Analyse

Mithilfe des Post Hoc Tests wird der Einfluss der verschiedenen Cluster aufeinander für beide Dimensionen untersucht. Dabei wird bei einem paarweisen Vergleich der Mittelwerte der Dimensionen *eventfulness* und *pleasantness* geprüft, ob der p-Wert unterhalb des Signifikanzniveaus α = 0,05 liegt.

Post Hoc Tests

Post Hoc Comparisons - Label

Comparison							
Label		Label	Difference	SE	t	df	$p_{bonferroni}$
0	-	1	-0.8218	0.465	-1.7678	581	1.000
0	-	2	0.9476	0.451	2.1001	568	0.542
0	-	3	2.8531	0.470	6.0665	589	< .001
0	-	4	1.2366	0.467	2.6465	584	0.125
0	-	5	-0.8030	0.459	-1.7486	579	1.000
1	-	2	1.7694	0.482	3.6734	592	0.004
1	-	3	3.6748	0.480	7.6579	591	< .001
1	-	4	2.0584	0.483	4.2646	593	< .001
1	-	5	0.0188	0.463	0.0406	578	1.000
2	-	3	1.9055	0.477	3.9927	592	0.001
2	-	4	0.2890	0.465	0.6213	579	1.000
2	-	5	-1.7506	0.468	-3.7436	587	0.003
3	-	4	-1.6165	0.465	-3.4794	583	0.008
3	-	5	-3.6561	0.468	-7.8074	586	< .001
4	-	5	-2.0396	0.478	-4.2670	591	< .001

Abbildung 12 - Post Hoc Test – eventfulness / Cluster

<u>Eventfulness</u>

Jedes Cluster bildet einen signifikanten Unterschied zu dem Cluster 3 – Background, das ebenso den niedrigsten *eventfulness* Wert hat. Dies bestätigt, dass das Clustering der gleichbleibenden, wenig salienten Soundscapes erfolgreich war. Cluster 2 - Haushalt und Cluster 4 - Verkehr haben beide einen signifikanten Unterschied zum Cluster 5 - Musik, das mit 2.7 den höchsten Mittelwert hat. Ebenso werden die Cluster für Haushalt und Verkehr beide sehr ähnlich bewertet (0.7; 0.5) und es gibt keine relevanten Unterschiede zwischen ihnen. Womöglich bewerten die Probanden diese Situationen als gewöhnlich innerhalb der Wohnung. Demnach werden beide Sprachkategorien höher in *eventfulness* bewertet, als die Kategorie Verkehr, was vielleicht auf eine persönliche Betroffenheit mit der Situation zurückzuführen ist.

Zwischen den beiden Sprachclustern 0 und 1 sind keine statistisch signifikanten Unterschiede festzustellen. Beide Cluster für Sprache 0 und 1 haben mit einem p-Wert von 1.0 ebenfalls keine signifikanten Unterschiede zum Cluster 5. Nur minimale Bewertungsunterschiede gibt es zwischen der vordergründigen Sprachkategorie in Cluster 1 und dem Cluster 2 – Haushalt. Das

hintergründige Sprachbild in Cluster 0 liegt im Vergleich zu Cluster 2 mit einem p-Wert von 0,542 deutlich über dem nötigen Signifikanzniveau von 0,05.

Post Hoc Tests

Post Hoc Comparisons - Label

Comparison							
Label		Label	Difference	SE	t	df	$p_{bonferroni}$
0	-	1	-0.6249	0.416	-1.5015	568	1.000
0	-	2	2.2626	0.403	5.6163	556	< .001
0	-	3	1.1127	0.422	2.6355	578	0.129
0	-	4	1.1405	0.419	2.7234	572	0.100
0	-	5	-1.3041	0.411	-3.1731	567	0.024
1	-	2	2.8876	0.433	6.6670	583	< .001
1	-	3	1.7376	0.431	4.0309	580	< .001
1	-	4	1.7655	0.434	4.0661	584	< .001
1	-	5	-0.6792	0.414	-1.6407	565	1.000
2	-	3	-1.1500	0.429	-2.6789	584	0.114
2	-	4	-1.1221	0.416	-2.6956	566	0.109
2	-	5	-3.5668	0.419	-8.5032	575	< .001
3	-	4	0.0279	0.416	0.0670	571	1.000
3	-	5	-2.4168	0.420	-5.7546	575	< .001
4	-	5	-2.4447	0.429	-5.6928	580	< .001

Abbildung 13 - Post Hoc Test – pleasantness / Cluster

Pleasantness

Die auffälligste signifikante Differenz in *pleasantness* besteht zwischen dem Cluster 2 - Haushalt mit dem geringsten Wert von 0.198 und dem Cluster 5 - Musik mit dem höchsten Wert von 3.88. Auch alle anderen Cluster sind im Vergleich zu Cluster 2 signifikant höher in *pleasantness* bewertet, was auf besonders scharfe oder nervige tonale Anteile innerhalb der Haushaltscluster zurückzuführen sein könnte, oder auch mit der Einstellung zu Hausarbeit an sich.

An zweit höchster Stelle stehen die Sprachcluster 0 und 1 mit einem Wert von 2,7 in *pleasantness*, zwischen denen kein Unterschied in der Bewertung festgestellt werden kann. Auffälligerweise werden Cluster 3 - Background und 4 - Verkehr ähnlich bewertet mit ca. 1.5 in *pleasantness*. Verkehr wird in *eventfulness* höher bewertet als Background jedoch scheinen die Soundscapes keine maßgeblichen Auswirkungen auf die *pleasantness* zu haben. Dies kann auch an einem ungenauen Clustering liegen: Ist eine Aufnahme durchgehend monoton und wird in Cluster 3 eingeordnet, während auf einer anderen bei gleichem Stimmungsbild ein Auto vorbeifährt und diese in Cluster 4 eingeordnet wird, muss das anscheinend noch nicht entscheidend für die Stimmung der jeweiligen Versuchsperson sein. Um diesen Unterschied zu erforschen, bräuchte es womöglich eine genaue Filterung auf das auffälligste wahrgenommene Geräusch, worauf im nächsten Teil (6.3.3) genauer eingegangen wird. Cluster 3 und 4 werden weniger „pleasant“ eingeschätzt als die Sprachcluster 0 und 1 und Cluster 5 - Musik, was darauf schließen lässt, dass menschliche Interaktion und mediale Unterhaltung den *pleasantness* - Wert im Gegensatz zur normalen Soundkulisse in der Wohnung anheben.

Interessanterweise ist das Cluster 0 - Hintergründige Sprache in der geringeren Einschätzung zu Cluster 5 - Musik mit p = 2,4% statistisch relevant, währenddessen Cluster 1 - vordergründige Sprache bei fast gleichem *pleasantness* - Wert einen Zufallsbefund im Vergleich zu Cluster 5 feststellt (p = 1). Diese Beobachtung könnte mit einer erhöhten Involvierung der Person in Cluster 1 zu tun haben oder auch mit dem höheren Vorkommen von Fernseherszenen in Cluster 1 (wie in Kapitel 6.1 beschrieben 23% häufiger).

In der Gesamtheit werden von den Mittelwerten der Cluster keine negativen Werte abgebildet. Das lässt vermuten, dass die generelle Grundstimmung eher neutral und positiv ist und für negative Ausreißer eine feinere Filterung notwendig ist, z.B. über saliente Geräusche.

Abschließend werden beide beschriebenen Dimensionen auf die Achsen des Diagramms in Abbildung 14 aufgetragen und ermöglichen eine einheitliche Betrachtung.

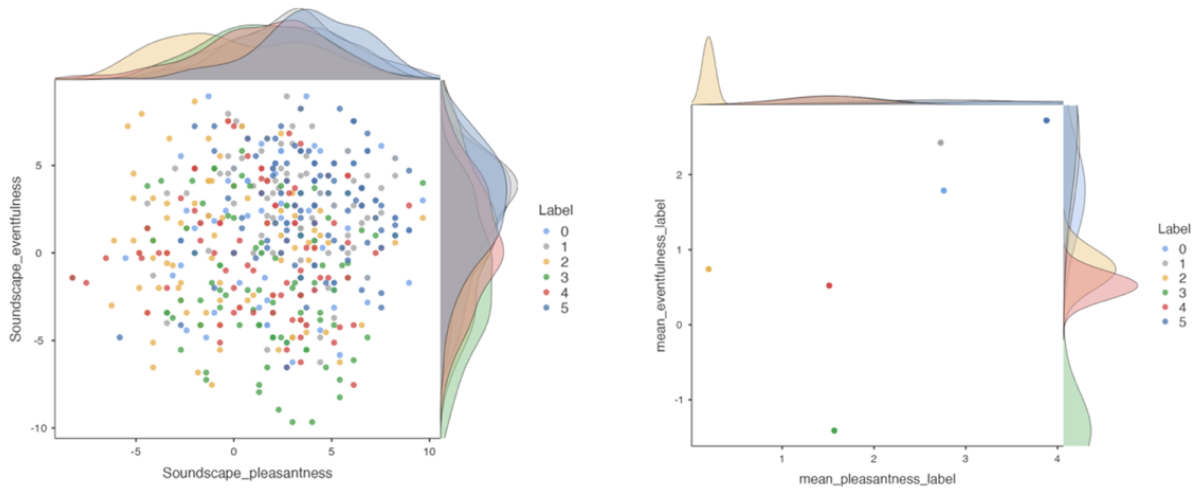


Abbildung 14 - Streudiagramm der gesamten Stichprobe (links) und deren Mittelwerte (rechts)

6.3.3 Einfluss der Salienz

„Messverfahren, Bewertungen oder die Evaluation von Soundscapes sind in Übereinstimmung mit dem Standard durch die Wahrnehmung der akustischen Umgebung geprägt“ (ISO 12913-2, 2018). Ein entscheidender Aspekt in der Bewertung der empirischen Ergebnisse ist die Interaktion zwischen der physikalischen Geräuschkulisse und der persönlichen Wahrnehmung dieser. Ein Versuch, Erkenntnisse über die Wahrnehmung der Teilnehmer zu gewinnen, besteht darin, das Clustering mit dem von den Teilnehmern angegebenen, auffälligsten Geräusch der Szene abzugleichen. Abbildung 15 zeigt die Häufigkeitsverteilungen für die Einordnung in die Kategorien (Speech, Traffic, Music, Domestic Installation etc.) für jedes ermittelte Cluster /„Label“. Die grafische Analyse bestätigt, dass sich die häufigsten salienten Geräusche mit der Clusterbenennung decken.

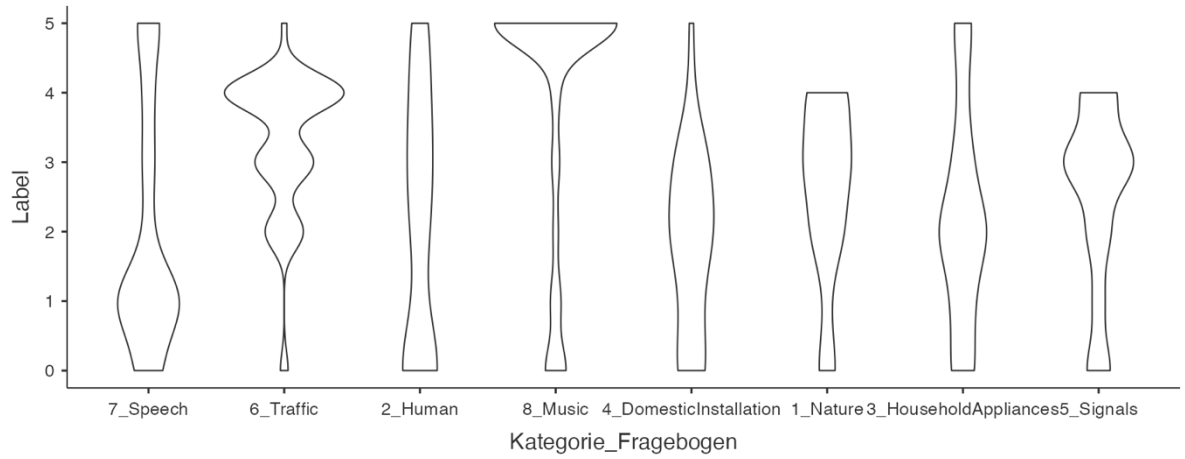


Abbildung 15 - Vorkomme der von Probanden als salient angegebene Kategorie in den Clustern von k = 5

Wird die Auswahl von 100 Daten pro Cluster auf die jeweilige Fragebogenkategorie gefiltert, sieht man die absolute Häufigkeit der Übereinstimmung. Diese kann ebenso als die Wahrscheinlichkeit interpretiert werden, mit der das als "am auffälligsten bewertete Geräusch", mit der Clusterbenennung der Datei übereinstimmt (Abbildung 16).

Die höchste Übereinstimmung gibt es in Cluster 1 - Sprache im Vordergrund mit 71%. In Cluster 0 - Sprache haben lediglich 26 aus 100 Teilnehmern Sprache als salientes Geräusch wahrgenommen. Dies bestätigt die These, dass in diesem Cluster Sprache eher hintergründig vorkommt. Weitere 29 von 100 haben die Szenen der Kategorie „Human" zugeordnet. Da dieser Begriff Raum für Interpretation lässt und teilweise mit anderen Kategorien wie z.B. Haushalt zusammenhängt, sehen wir ihn nur als Ergänzung der Clusterbenennung an.

LABEL	KATEGORIE FILTERUNG	ÜBEREINSTIMMUNG
0	„Speech"	26 %
1	„Speech"	71 %
2	„Household Aplliance", „Domestic Installation", „Human"	59 %
3	„Traffic", „Nature", „Signals"	47 %
4	„Traffic"	47 %
5	„Music"	48 %

Abbildung 16 - Vorkommen der von Probanden als salient angegebenen Kategorie in den Clustern 0 bis 5 und deren prozentuale Übereinstimmung zum Clusterlabel

Das Cluster 3 - Background weist eine ausgeglichenere Verteilung in allen Kategorien auf, wobei die Kategorien „Traffic", „Nature" und im kleinen Teil ,,Signals" die meisten Übereinstimmungen haben.

Das Cluster 2 - Haushalt benötigt ebenfalls eine erweiterte Filterung mit den Kategorien „Household Alliance", „Domestic Installation", und in kleineren Teilen „Human", da sich dort oft auf Haushaltsaktivitäten bezogen wird.

Auffällig ist, dass die Kategorie „Traffic“ zwar zum größten Teil dem Cluster 4 - Verkehr zugeordnet wird, es aber auch, wie vermutet, Überschneidungen in das Cluster 3 - Background gibt. Außerdem kommt ein kleinerer Teil der Kategorie „Traffic“ auch in dem Cluster 2 - Haushalt vor. Vermutlich könnte es klangspezifische Ähnlichkeiten (Spektrum, Pegel) zwischen den Quellen (Haushaltsgeräte, Kraftfahrzeuge) geben, die zu Schwierigkeiten bei der Unterscheidung führen und somit Optimierungsbedarf für das verwendete neuronale Netzwerk darstellen.

Errechnet man das arithmetische Mittel der Übereinstimmung aller Cluster so erhält man eine 50% Wahrscheinlichkeit, dass die Wahrnehmung der Personen mit der Clusterbenennung übereinstimmt.

Dieses Vorgehen bringt neue Möglichkeiten der empirischen Auswertung der Dimensionen *eventfulness* und *pleasantness* auf. Ein Ansatz besteht darin, die ursprünglichen Mittelwerte dieser Dimensionen pro Cluster, mit den auf Salienz gefilterten Stichproben zu vergleichen. In den meisten Fällen erhält man dabei zunächst sehr ähnliche Mittelwerte im Bereich von ±0.5.

Etwas aussagekräftigere Ergebnisse erhält man bei der im vorigen Kapitel erwähnten Fragestellung auf die Auswirkungen der bewussten Wahrnehmung von Verkehrsgeräuschen.

Untersucht man das Cluster 4 - Verkehr, erhält man in der gefilterten Stichprobe „Traffic“ (n = 47) einen verringerten ,,Pleasantwert” um -1,2. Filtert man im selben Cluster statt der Kategorie „Traffic“ auf die Kategorie „Nature“ (n = 16), wird *pleasantness* um +2,81 erhöht. Die entstehende Differenz liegt für dieses Cluster mit 4,1 über der Standardabweichung von 3,3 und lässt einen Einfluss der Salienz vermuten, der genauer untersucht werden könnte.

Bei analogem Vorgehen für das Cluster 3 - Background erhält man für „Traffic“ (n = 23) -1,22 und für ,,Nature“ (n = 18) + 1,75 Abweichung vom ursprünglichen Mittelwert. Diese Werte zeigen die Schwierigkeiten der Unterscheidung von Verkehr und Background auf. Zwar haben etwa doppelt so viele Teilnehmer „Traffic“ in Cluster 4 - Verkehr als auffällig wahrgenommen jedoch ist die Bewertung der *pleasantness* in beiden Fällen fast gleich (1,5 - 1,2 = 0,3). Allerdings wird „Traffic“ in Cluster 3 - Background mit -1,37 und im Cluster 4 - Verkehr mit 0.378 um -1.0 geringer in *eventfulness* bewertet. Dieser Unterschied rechtfertigt das Clustering in diesem Punkt eventuell, zeigt aber auch dessen eventuelle Ungenauigkeiten auf.

Betrachtet man die Fälle, in denen im Cluster 5 - Musik die Kategorie „Music“ angegeben wurde, so sieht man eine Erhöhung des Mittelwertes von *pleasantness* um +0,98 und der *eventfulness* um +0,79.

Die Beobachtungen zeigen, dass das Clustering im überwiegenden Teil erfolgreich war. Die Auswertung der empirischen Dimensionen ist signifikant und gibt Tendenzen für die jeweiligen

Kategorien vor. Diese können durch den Einfluss von Salienz in Maßen beeinflusst werden. Das auffälligste Geräusch und die Clusterbezeichnung decken sich in jedem zweiten Fall. Dies zeigt die Diskrepanz, die einerseits in der Genauigkeit des neuronalen Netzwerkes und anderseits in der individuellen Wahrnehmung der Personen begründet ist.

7 Fazit

Das Clustering der Summe hat im Kontext des komplexen Datensatzes gut funktioniert. Die Embeddings, die mithilfe des OpenL3 Netzes extrahiert werden, scheinen die grundlegenden Informationen der Dateien in großen Teilen widerspiegeln zu können.

Da die Cluster teilweise überlappen und generell sehr nah beieinanderliegen, kann man hier nicht von einem sehr guten Clustering sprechen, was zum einen mit dem angewendeten Clustering-Algorithmus und zum anderen mit der Komplexität der vorliegenden Daten zusammenhängt. Im Kontext dieser Arbeit konnte für eine Clusteranzahl von 6 die beste Clusteranalyse festgestellt werden.

Die Trennung in Vordergrund und Hintergrund hat das Ergebnis dabei nicht nennenswert verbessert. Grund dafür könnte zunächst die Qualität der Quellentrennung sein, gerade die Vordergrundtrennung ist sehr artefakt-behaftet und enthält oft nur sporadische Geräuschimpulse. Der Hintergrund ist dabei der Summe sehr ähnlich, was ebenfalls keinen Vorteil bringt. Das ist auch der Grund, weshalb sich die empirische Auswertung dieser Arbeit nur mit der Summe beschäftigt.

Prinzipiell ist die Idee der Trennung in Vordergrund und Hintergrund jedoch interessant. Um diesen Ansatz weiterzuverfolgen, müsste man zuallererst den Trennungsalgorithmus verbessern und z.B. durch eine geeignete Vorfilterung besser an den vorliegenden Datensatz anpassen. Außerdem kann es sinnvoll sein, den semantischen Zusammenhang von Vordergrund und Hintergrund nicht durch eine getrennte Betrachtung bzw. ein getrenntes Clustering aufzulösen, sondern vielmehr beide Ansichten zur Extraktion eines gemeinsamen Embeddings zu nutzen, wie es z.B. in dem Paper *„Multiview Embeddings for Soundscape Classification“* (Dhanunjaya Varma Devalraju, 2022) beschrieben wird.

Die Einblicke in die empirische Auswertung und vor allem die Post Hoc Testanalyse hat hier interessante Aufschlüsse über die signifikanten Unterschiede zwischen den Clustern gegeben. Die Zusammenhänge zwischen den Embeddings und der subjektiven Wahrnehmung der Probanden versprechen interessante Aussichten für weitere Untersuchungen.

8 Literaturverzeichnis

Östen Axelsson, C. G. (2020). *Soundscape Assessment.* Stockholm: Frontiers in Psychology.

Axelsson, Ö. N. (2012). *A principal components model of soundscape perception*.

Davies, W. J. (2012). *Reproducibility of soundscape dimensions. Paper Presented at the 41st International Congress and Exposition on Noise Control Engineering.* New York.

Dhanunjaya Varma Devalraju, P. R. (2022). *Multiview Embeddings for Soundscape Classification*. IEEE/ACM TRANSACTIONS ON AUDIO, SPEECH, AND LANGUAGE PROCESSING, VOL. 30.

Gareth James, D. W. (2023). *An Introduction to Statistical Learning*. Stanford University: Springer Texts in Statistics.

IBM . (9. September 2024). *Common unsupervised learning approaches*. Von https://www.ibm.com/topics/unsupervised-learning abgerufen

ISO 12913-2. (2018). *Acoustics -Soundscape Part 2: Data collection and reporting requirements*.

ISO 12931-1. (2015). *Acoustics-Soundscape- Part 1: Definition and Conceptual Framework,*.

Jason Cramer, H.-H. W. (2019). *LOOK, LISTEN, AND LEARN MORE: DESIGN CHOICES FOR DEEP AUDIO EMBEDDINGS*.

Jian Chen, V. N.-N. (2019). *Soundscapes ans System Science*. Da Nang, Vietnam: Springer.

Kacprzyk, P. J. (2006). *Feature Extraction - Foundations and Applications*. Polish Academy of Sciences, Warsaw: Springer.

Laurens van der Maaten, G. H. (2008). *Visualizing Data using t-SNE*. Niederlande, Kanada: Journal of Machine Learning Research.

Rajan, D. V. (2022). *Multiview Embeddings for Soundscape Classification*. IEEE.

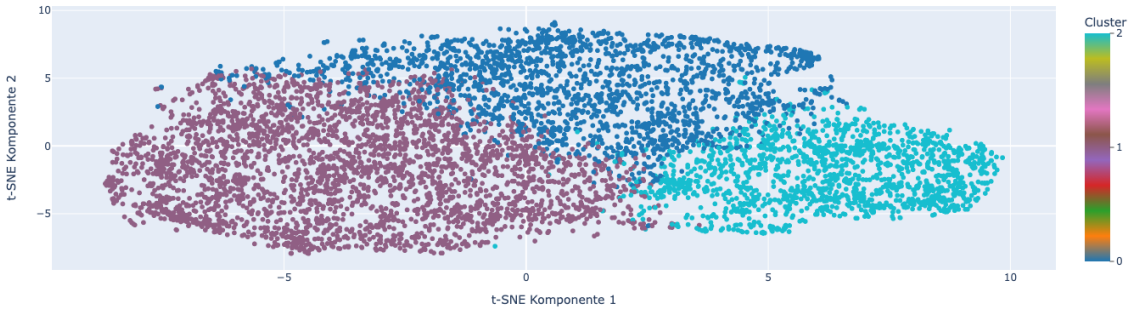
Sabrina Zirkel, J. A. (2015). *Experience-Sampling Research Methods and Their Potential for Education Research*. San Francisco: AERA.

Siegbert Versümer, J. S. (2023). *Extensive crowdsourced dataset of in-situ evaluated binaural soundscapes of private dwellings containing subjective sound-related and situational ratings along with person factors to study tim-varying influences on sound perseption*. Düsseldorf.

9 Anhang

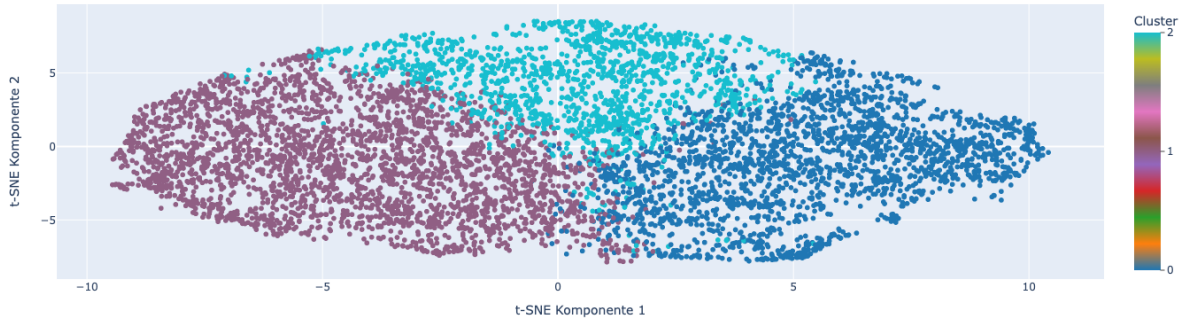
9.1 Hörtest Protokolle mit Clustergrafiken

9.1.1 Clustering mit 3 Clustern - Summe



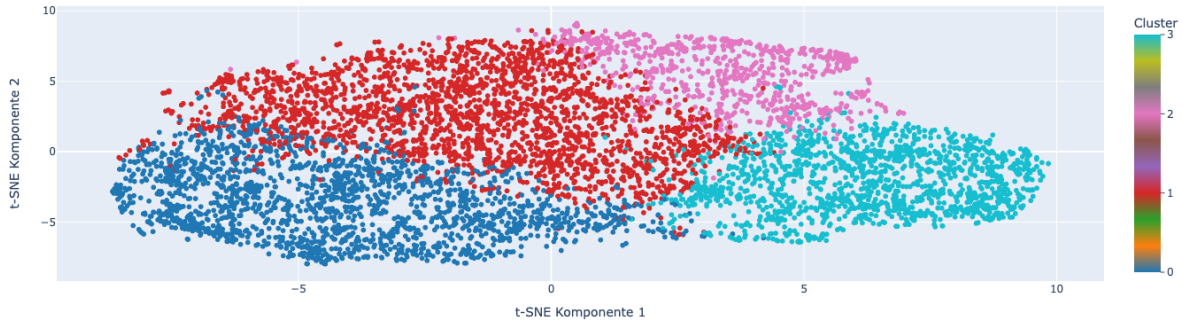
CLUSTER	HÖRPROBE KATEGORIE	EINSCHÄTZUNG
0	Außengeräusche (z.B. Autos, Flugzeug) Haushalt (z.B. Kaffeemaschine, menschliche Stimme	Schlecht - mittel,
1	Rauschen, keine sehr lauten salienten Impulse, Vögel, entfernte Musik	Mittel
2	Fernseher, Sprache und Musik Entfernte, Saliente Impulse	Gut

9.1.2 Clustering mit 3 Clustern - Vordergrund



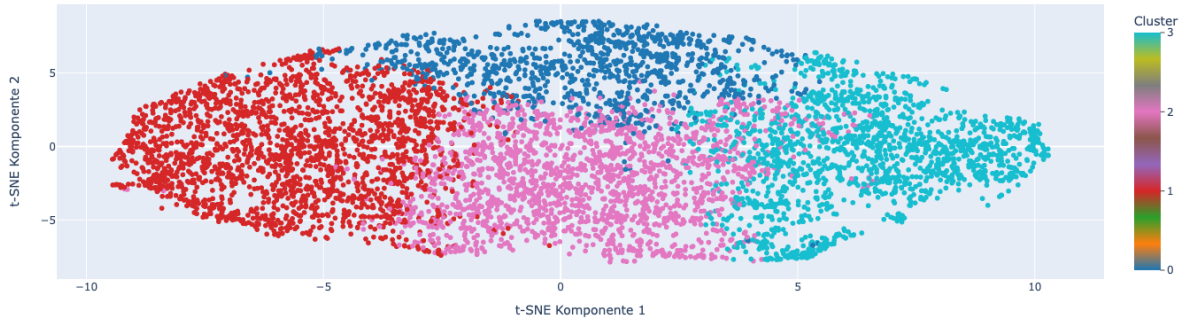
CLUSTER	HÖRPROBE KATEGORIE	EINSCHÄTZUNG
0	Fernseher, Youtube, Sprache	Gut Da Hintergrund oft sehr laut, könnte Vordergrundtrennung Verbesserung bedeuten.
1	Hochfrequente Töne	Mittel - Gut
2	Tieffrequente Geräusche	Schlecht - Mittel Kein semantischer Zusammenhang

9.1.3 Clustering mit 4 Clustern - Summe



CLUSTER	HÖRPROBE KATEGORIE	EINSCHÄTZUNG
0	Viel rauschen mit kurzen teilweise Salienten Geräuschen (Piepen, Tür öffnen)	Mittel bis Schlecht – wenig Zusammenhang zwischen Geräuschen
1	Ähnlich zu Cluster 0 Verkehrslärm, Küchengeräusch und Radio	Mittel bis Schlecht – wenig Zusammenhang zwischen Geräuschen
2	Küche, Kaffeemaschine, Toilette, Wasserplätschern	Mittel - gut
3	Menschliche Stimme - Echt und Fernseher, Musik	Gut

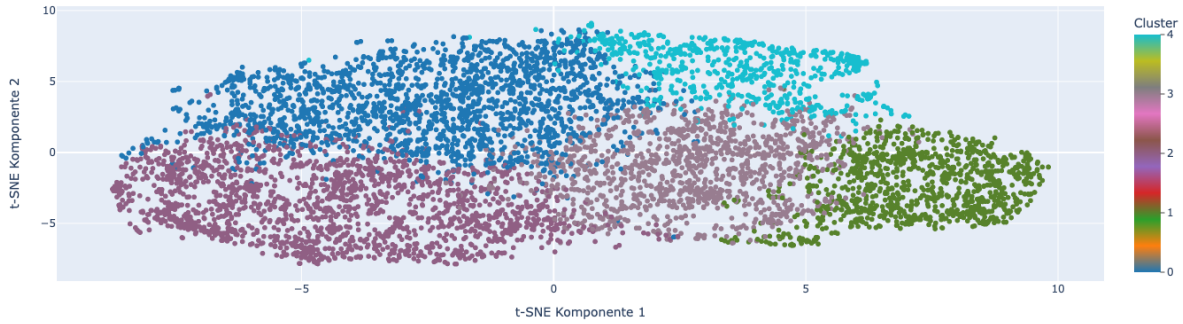
9.1.4 Clustering mit 4 Clustern - Vordergrund



CLUSTER	HÖRPROBE KATEGORIE	EINSCHÄTZUNG
0	Uhr, Bohrmaschine, Stimme, Fernseher Ähnliche wie bei Cluster 3 in k=3	Mittel bis schlecht
1	Hohes sirren, Wasser/Duschen, Straßenlärm, Uhrticken, Gespräch und Rauschen	Mittel bis Schlecht, Einzige Gemeinsamkeit: überall ein salientes Geräusch (ohne Zusammenhang
2	Viel Rauschen, Gespräch, Tellergeräusche bzw. Essen, Maschinen Geräusche	Schlecht

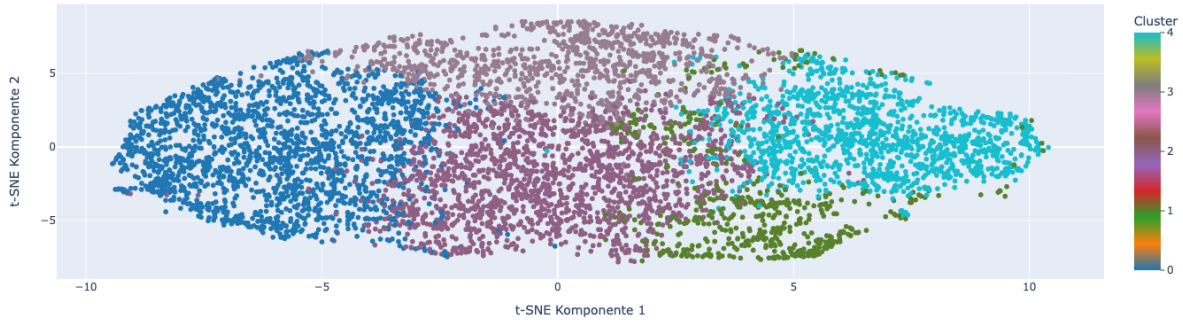
3	Menschliche Sprache (leise und eigentlich verdeckt vom Hintergrund, z.B. Fernseher / Musik / Rauschen	Gut – Sehr gut Auch hier könnte Vordergrundtrennung geholfen haben

9.1.5 Clustering mit 5 Clustern - Summe



CLUSTER	HÖRPROBE KATEGORIE	EINSCHÄTZUNG
0	Motoren, Auto, Straßenlärm, teilweise gemischt mit Wohnungsgeräuschen	Gut
1	Fernseher, Sprache	Gut - Sehr gut
2	Saliente Geräusche innerhalb normaler Soundszenen	Mittel kein semantischer Zusammenhang zwischen Soundereignissen
3	Laute Impulse, Vögel, Uhren, Presslufthammer, Sprache, viel Abwechslung innerhalb in 15s	Mittel - Schlecht
4	Haushaltsgeräusche (Kaffemaschine, Bad und Spülung)	gut

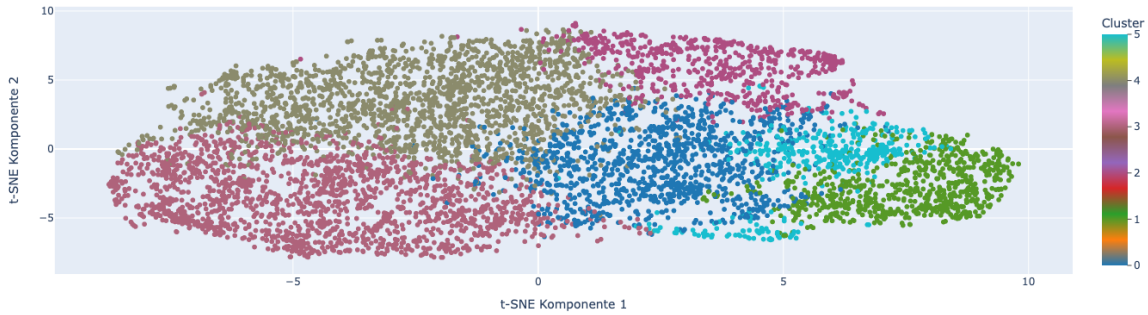
9.1.6 Clustering mit 5 Clustern - Vordergrund



CLUSTER	HÖRPROBE KATEGORIE	ERSTE EINSCHÄTZUNG
0	Straßenlärm	Mittel Keine Verbesserung zu Summe
1	Straße, Bad, Fernseher Generell alles hell und scharf	Mittel

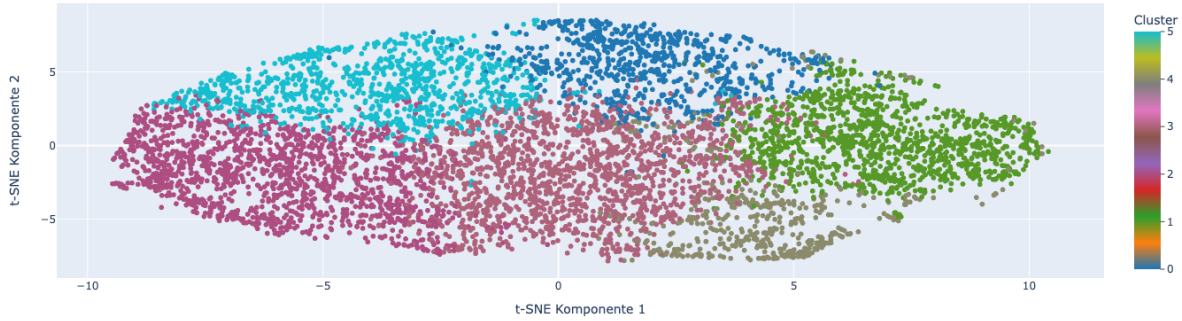
		Verschlechterung im Vergleich zur Summe
2	Kein klares Muster, viel monotone Soundscapes	Mittel Ähnlich zu Summe
3	Kauen, Uhr, Vögel, Handywecker und Fernseher	Mittel Relativ klare saliente Geräusche aber kein semantischer Zusammenhang
4	Fernseher	Sehr Gut Besser als Summe

9.1.7 Clustering mit 6 Clustern - Summe



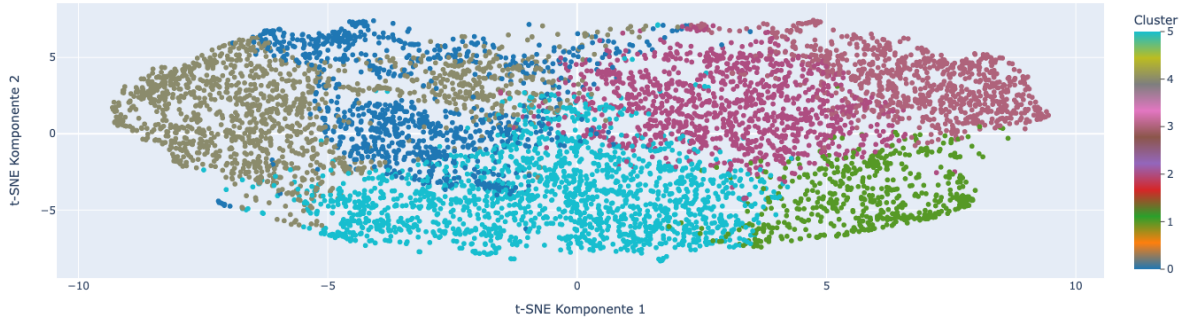
CLUSTER	HÖRPROBE KATEGORIE	EINSCHÄTZUNG
0	Sprache (Menschen, Fernseher, Umgebungsgeräusche) Im Hintergrund	Gut
1	Sprache(Menschen, Fernseher) Im Vordergrund	Gut
2	Haushalt (Kaffeemaschine, Wasser, etc.)	Gut
3	Rauschen, Wind Entfernte Geräusche (Piepen, Vögel, Hämmern)	Mittel – Gut
4	Straßenverkehr	Gut – Sehr gut
5	Musik	Sehr gut

9.1.8 Clustering mit 6 Clustern - Vordergrund



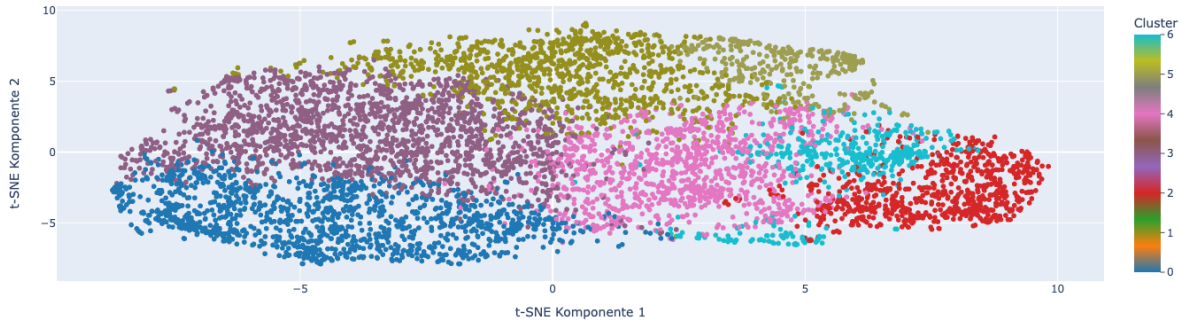
CLUSTER	HÖRPROBE KATEGORIE	EINSCHÄTZUNG
0	Klare Impulse im Haushalt (Abspülen, Schniefen aber auch Fernseher)	Mittel – schlecht Kein wirklich klares Muster
1	Fernseher	Sehr gut
2	Vorbeifahrende Autos, Straße	Gut
3	Background mit Impulsen aller Art	Mittel – Gut Nicht so klare Beispiele
4	Haushalt, Auto, Fernseher, Rauschen, Hintergrundgeräusche	Schlecht – Mittel Nicht so klar
5	Verschiedene Szenen	Schlecht Nicht so viel Gemeinsamkeiten

9.1.9 Clustering mit 6 Clustern - Hintergrund



CLUSTER	HÖRPROBE KATEGORIE	EINSCHÄTZUNG
0	Backgroundszenen: Rauschen, Kinder, Vögel, tiefe Frequenz Anteile	Gut
1	Haushalt, Wasser	Gut - Sehr gut
2	Sprache, Fernseher, Musik,	Gut
3	Musik? Sprache? --> Musik besser getrennt?	Gut
4	Rauschen, Vordergrund	Gut
5	Straßenverkehr	Gut

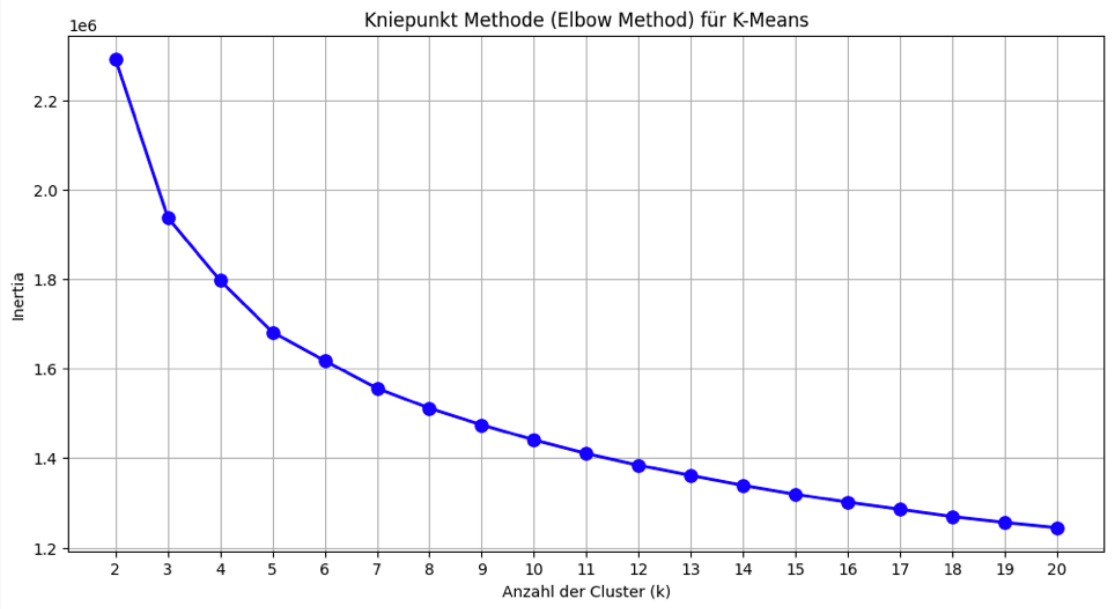
9.1.10 Clustering mit 7 Clustern - Summe

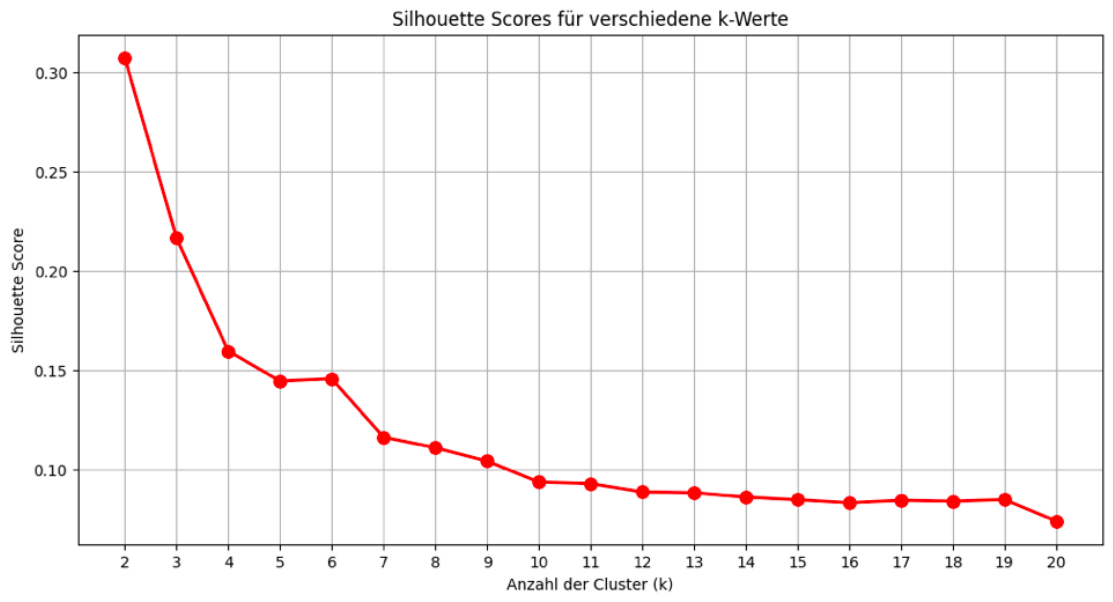


CLUSTER	HÖRPROBE KATEGORIE	EINSCHÄTZUNG
0	Verkehr und andere Hintergrundgeräusche	Mittel - Gut
1	Lärm von draußen, Auto, Vögel	Mittel - Gut
2	Fernseher und Menschen	Mittel - gut
3	Auto, Haushalt, Schnarchen	Schlecht Im Kontext der anderen Cluster
4	Hintergrundgeräusche, Musik, Sprache, Baugeräusche	Schlecht
5	Lärm, Musik, Haushalt	Schlecht
6	Musik und Fernseher	Mittel – gut

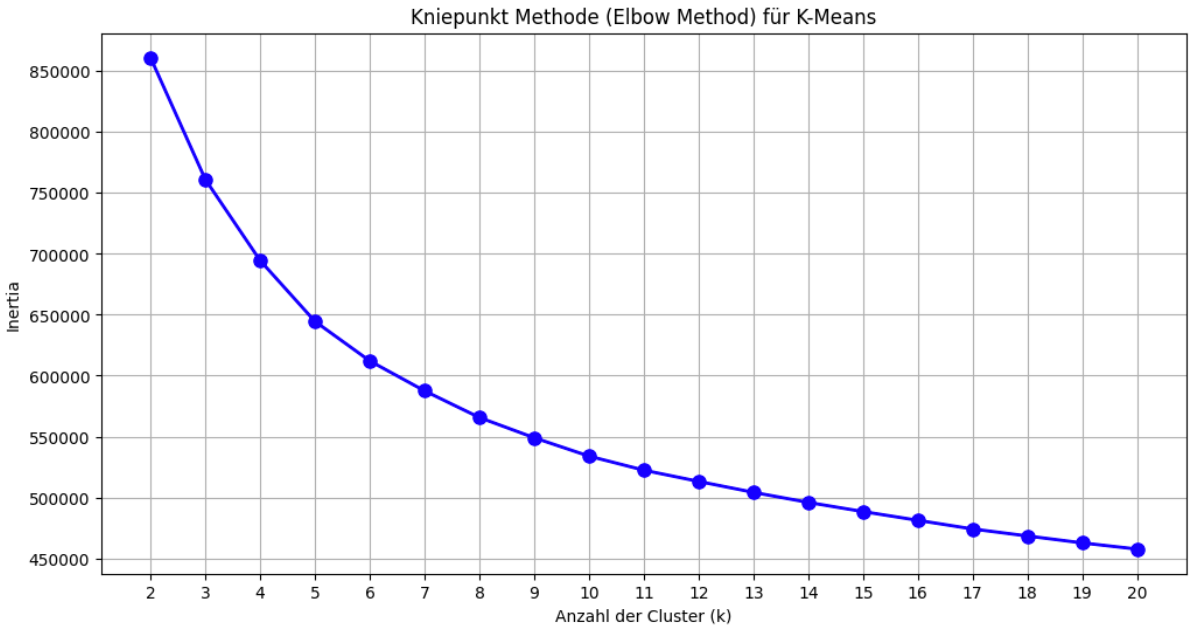
9.2 Inertia und Silhouette Score Plots (Teils evtl. Im Fließtext)

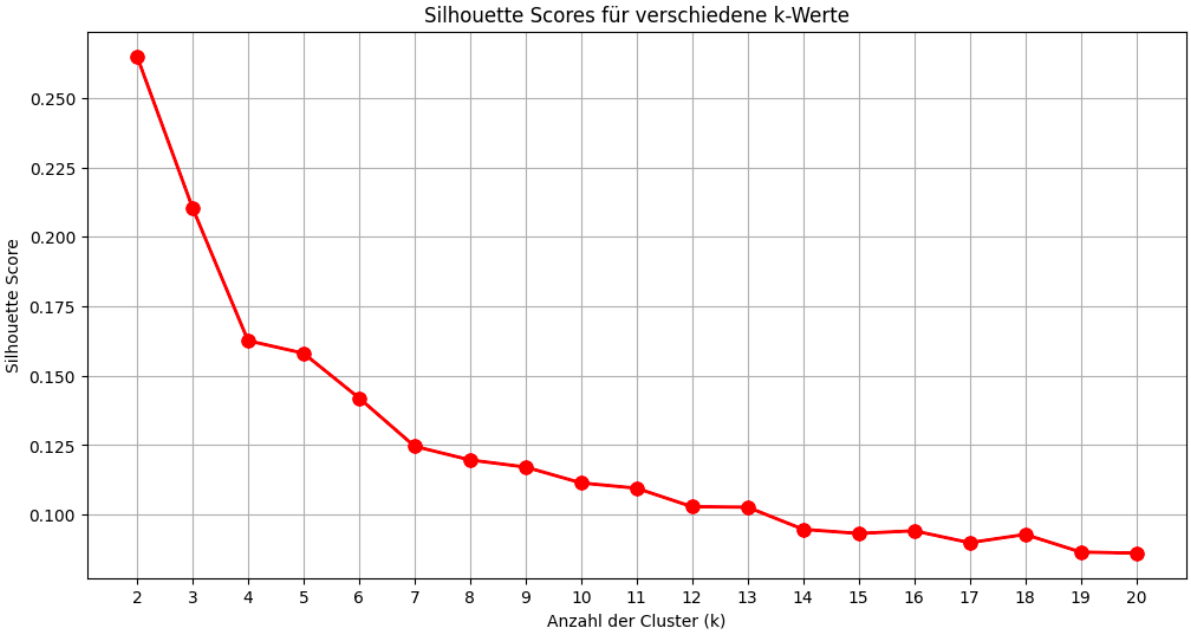
9.2.1 Plots für Inertia und Silhouette Score der Summe über 2-20 Cluster



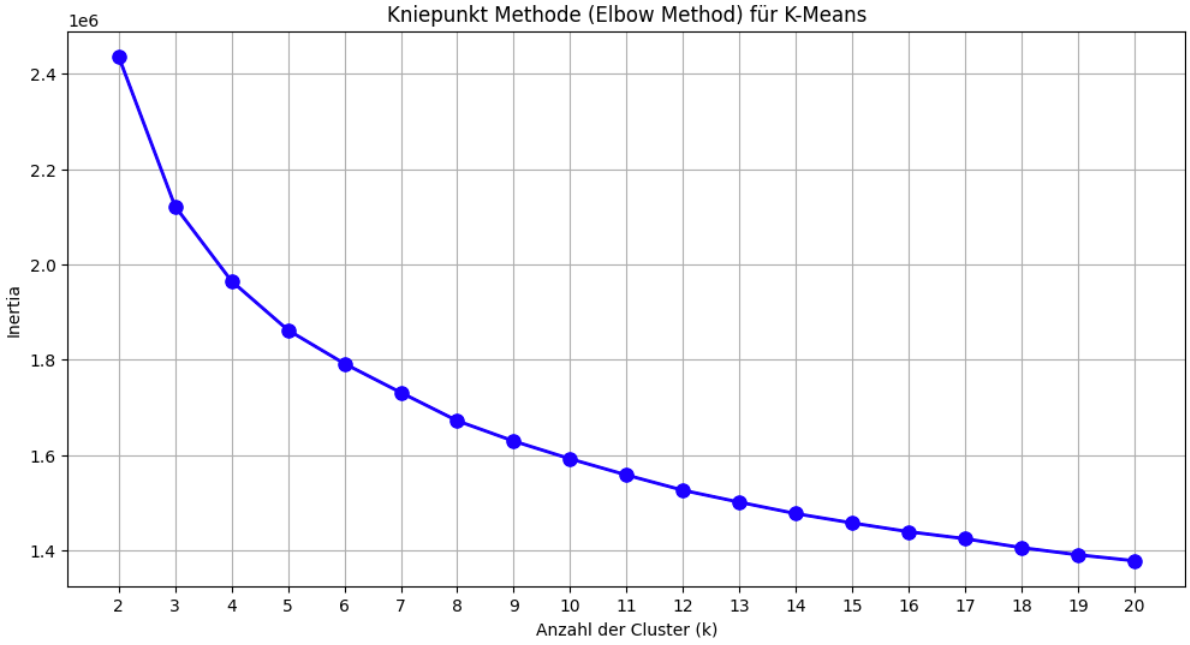


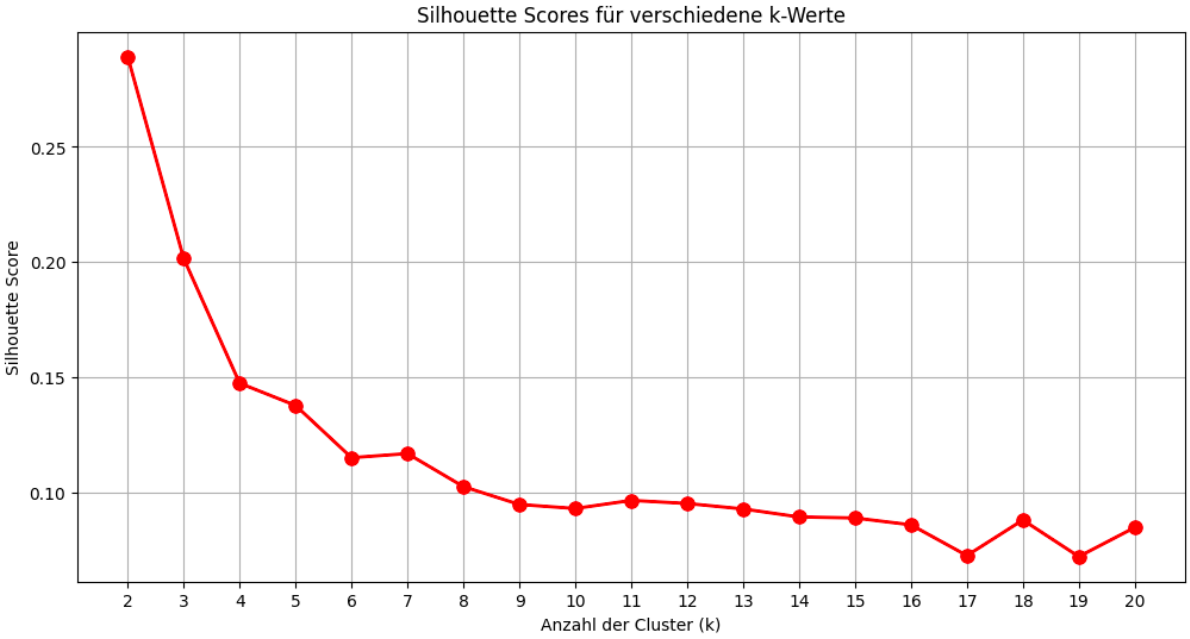
9.2.2 Inertia und Silhouette Score des Vordergrunds über 2-20 Cluster





9.2.3 Inertia und Silhouette Score des Hintergrunds über 2-20 Cluster





9.3 Python Code

Folgend ein Screenshot unseres Python Codes. Das Jupiter Notebook wird der Abgabe als Datei beigefügt.

```
# %% [markdown]
# Bibliotheken importieren

# %%
#emb
import os
import numpy as np
import openl3
import soundfile as sf

#Cluster
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_score

#TSNE
from sklearn.manifold import TSNE

#Plotten
import matplotlib.pyplot as plt
from sklearn.decomposition import PCA

#interaktives PLotly Plotten

import plotly.express as px
import pandas as pd


# %% [markdown]
# Audiodaten einlesen

# %%
input_path =''
output=''

# %% [markdown]
# Embeddings mit dem OpenL3 Netz erstellen und speichern

# %%
files = os.listdir(input_path)
for file in files:
    filePath = input_path + '/' + file
    audio, sr = sf.read(filePath)
    #emb, ts = openl3.get_embedding(audio, sr)
    emb, ts = openl3.get_audio_embedding(audio, sr)
    outFileName = output + '/' + file.split('.')[0]
    np.save(outFileName, emb)


# %% [markdown]
# Sammelmatrix der Embeddings erstellen und speichern

# %%
input_path = ''
output_path= ''

data = []
datanames = []

files = os.listdir(input_path)
for file in files:
    filePath = input_path + '/' + file
    data.append(np.mean(np.load(filePath), axis=0))
    datanames.append(file)

print(np.array(data).shape)
np.save(output_path, np.array(data))

data_matrix = np.array(data)

#Ausgabe: (x = Anzahl der Dateien, y = Feature Exktraktion)

# %% [markdown]
# Funktion für Clustering mit kMeans erstellen und wichtige Größen zurückgeben

# %%
# Anzahl der Cluster definieren mit Übergabeparameter k und Clustering auf die Sammelmatrix "data_matrix" anwenden
def makeKmeans(k, data_matrix):

    # KMeans Modell initialisieren
    kmeans = KMeans(n_clusters=k, random_state=0)

    # Modell auf die Sammelmatrix anwenden
    kmeans.fit(data_matrix)

    # Clusterlabels erhalten
    labels = kmeans.labels_

    # Clusterzentren erhalten
    centers = kmeans.cluster_centers_

    # Inertia (Summe der quadrierten Abstände zum nächsten Clusterzentrum)
    inertia = kmeans.inertia_

    # Silhouette Score berechnen
    score = silhouette_score(data_matrix, labels)
    # Daten für die Analyse übergeben
    return(data_matrix, labels, centers, inertia, score)


# %% [markdown]
# Optional: Optimale Clusteranzahl bestimmen über Kneepoint Methode der zu erstellenden Inertia und Silhouette Score Diagramme

# %%
def plot_inertia_and_find_elbow(data_matrix):
    inertias = []
    silhouettes = []
```

```
    k_values = range(2, 21)

    for k in k_values:
        _, _, _, inertia, silhouette = makeKmeans(k, data_matrix)
        inertias.append(inertia)
        silhouettes.append(silhouette)

    # Inertia-Werte plotten
    plt.figure(figsize=(12, 6))
    plt.plot(k_values, inertias, 'bo-', linewidth=2, markersize=8)
    plt.xlabel('Anzahl der Cluster (k)')
    plt.ylabel('Inertia')
    plt.title('Kniepunkt Methode (Elbow Method) für K-Means')
    plt.grid(True)
    plt.xticks(k_values)
    plt.show()

    # Silhouette Scores plotten
    plt.figure(figsize=(12, 6))
    plt.plot(k_values, silhouettes, 'ro-', linewidth=2, markersize=8)
    plt.xlabel('Anzahl der Cluster (k)')
    plt.ylabel('Silhouette Score')
    plt.title('Silhouette Scores für verschiedene k-Werte')
    plt.grid(True)
    plt.xticks(k_values)
    plt.show()

    return inertias, silhouettes

inertias, silhouettes = plot_inertia_and_find_elbow(data_matrix)


# %% [markdown]
# Das Clustering wird durchgeführt mit variablen k für k_Means, interaktiver t-SNE Plot wird ausgegeben, Dataframes für den Export des Gesamten Clusterings und

# %% [markdown]
# Neues Plotten mit Ausgabe der nähsten Punkte

# %%
Inertiawerte = []
silhouette_scores = []
#Optional: closest Points dataframe erstellen
closest_points_df = []


# Schleife für unterschiedliche k-Werte
for k in range(6, 7):
    # Aufruf der k-means Funktion mit variablem k für die Clusteranzahl
    tempdata_matrix, labels, centers, inertia, score = makeKmeans(k, data_matrix)

    # Listen erstellen für gerundeten Inertiawert und den Silhouette Score
    round_inertia = round(inertia, 2)
    Inertiawerte.append(round_inertia)
    silhouette_scores.append(score)

    # t-SNE Reduktion auf Datenpunkte und Clusterzentren anwenden
    combined_data = np.vstack([tempdata_matrix, centers])
    perplexity = 50
    tsne = TSNE(n_components=2, perplexity=perplexity, n_iter=300, random_state=42)
    combined_2d_tsne = tsne.fit_transform(combined_data)

    data_matrix_2d_tsne = combined_2d_tsne[:tempdata_matrix.shape[0], :]
    centers_2d_tsne = combined_2d_tsne[tempdata_matrix.shape[0]:, :]

    # Abstände der Datenpunkte zu ihrem jeweiligen Clusterzentrum berechnen
    distances = np.linalg.norm(tempdata_matrix[:, np.newaxis] - centers, axis=2)
    min_distances = np.min(distances, axis=1)

    # DataFrame für den Plot der gesamten Dateien erstellen
    df = pd.DataFrame({
        't-SNE Komponente 1': data_matrix_2d_tsne[:, 0],
        't-SNE Komponente 2': data_matrix_2d_tsne[:, 1],
        'Label': labels,
        'Dateiname': datanames,
        'Abstand zum Zentrum': min_distances
    })

    # DataFrame für die fünf nächsten Punkte erstellen
    closest_points = []
    for i in range(k):
        cluster_mask = (labels == i)
        cluster_distances = distances[cluster_mask, i]
        sorted_indices = np.argsort(cluster_distances)[:5]

        for idx in sorted_indices:
            closest_points.append({
                'Cluster': i,
                'Dateiname': datanames[np.where(cluster_mask)[0][idx]],
                't-SNE Komponente 1': data_matrix_2d_tsne[np.where(cluster_mask)[0][idx], 0],
                't-SNE Komponente 2': data_matrix_2d_tsne[np.where(cluster_mask)[0][idx], 1],
                'Abstand zum Zentrum': cluster_distances[idx]
            })

    # Hinzufügen der nächsten Punkte zum DataFrame
    closest_points_df.append(pd.DataFrame(closest_points))

    # Scatter Plot erstellen
    fig = px.scatter(
        df, x='t-SNE Komponente 1', y='t-SNE Komponente 2', color='Label',
        color_continuous_scale=px.colors.qualitative.D3,
        hover_data={
            'Dateiname': True,
            't-SNE Komponente 1': True,
            't-SNE Komponente 2': True,
            'Label': True,
            'Abstand zum Zentrum': True
```

```
        }
    )

    # Clusterzentren hinzufügen
    center_df = pd.DataFrame({
        't-SNE Komponente 1': centers_2d_tsne[:, 0],
        't-SNE Komponente 2': centers_2d_tsne[:, 1],
        'Label': [f'Center_{i}' for i in range(k)]
    })
    fig.add_scatter(x=center_df['t-SNE Komponente 1'], y=center_df['t-SNE Komponente 2'],
                    mode='markers', marker=dict(color='red', size=20, symbol='x'), name='Cluster Centers')

    # Identifizieren des Datenpunkts mit dem geringsten Abstand für jedes Cluster
    for i in range(k):
        cluster_mask = (labels == i)
        cluster_distances = distances[cluster_mask, i]
        min_idx = np.argmin(cluster_distances)

        closest_point_idx = np.where(cluster_mask)[0][min_idx]
        closest_point = df.iloc[closest_point_idx]

        # Datenpunkt mit dem geringsten Abstand hervorheben
        fig.add_scatter(
            x=[closest_point['t-SNE Komponente 1']],
            y=[closest_point['t-SNE Komponente 2']],
            mode='markers',
            marker=dict(color='blue', size=12, symbol='circle'),
            name=f'Closest Point {i}',
            text=[f"Dateiname: {closest_point['Dateiname']}, "
                  f"t-SNE Komponente 1: {closest_point['t-SNE Komponente 1']}, "
                  f"t-SNE Komponente 2: {closest_point['t-SNE Komponente 2']}, "
                  f"Label: {closest_point['Label']}, "
                  f"Abstand zum Zentrum: {closest_point['Abstand zum Zentrum']}"]
        )

    # Layout anpassen
    fig.update_layout(
        xaxis_title='t-SNE Komponente 1',
        yaxis_title='t-SNE Komponente 2',
        coloraxis_colorbar=dict(
            title='Cluster',
            tickvals=list(range(k)),
            ticks="outside"
        )
    )

    # Interaktives Diagramm anzeigen
    fig.show()

#Liste der Inertiawerte und Silhouette Scorewerte ausgeben, falls die Kneepoint
print(Inertiawerte, silhouette_scores)

# %% [markdown]
# Alle Closest Points in eine CSV Datei schreiben und exportieren

# %%
# Alle DataFrames für die nächsten 5 Punkte jedes Clusters zusammenführen
all_closest_points_df = pd.concat(closest_points_df, ignore_index=True)

# Die DataFrames in CSV-Dateien speichern
all_closest_points_df.to_csv('.csv',  index=False)

# %% [markdown]
# Gesamten Dataframe in CSV Datei schreiben und exportieren

# %%
# Output_Path angeben
file_path = os.path.join('', 'output.csv')

# DataFrame als CSV exportieren
df.to_csv(file_path, index=False)

# %% [markdown]
# Zusammenführen der Tabellen, CSV Output des Clusterings und Fragebogen über eine gemeinsame Spalte "BenutzerID_Audioaufnahme" (muss vorher in den Tabellen an

# %%
import pandas as pd

# Pfade zu den CSV-Dateien
file1 = ''
file2 = ''

# Gemeinsame Spalte zum Zusammenführen definieren
common_column = 'ID_Trigger'

# CSV-Dateien einlesen
table1 = pd.read_csv(file1)
table2 = pd.read_csv(file2)

 #Tabellen basierend auf der gemeinsamen Spalte zusammenführen
merged_table = pd.merge(table1, table2, on=common_column)

# Ergebnis als neue CSV-Datei speichern
merged_table.to_csv('.csv', index=False)

print("Die Tabellen wurden erfolgreich zusammengeführt")
```

9.4 Vollständiges Messprotokoll mit empirischer Auswertung

Das Messprotokoll wird der Abgabe als PDF-Datei beigefügt.

9.5 Fragebogen und Cluster-Ergebnisse in kombinierter Tabelle

Die zusammengeführte Tabelle des gesamten Datensatzes für die empirische Auswertung wird der Abgabe als csv-Datei beigefügt (vollständiger Datensatz).