



Universität Stuttgart

Institut für Parallele und Verteilte Systeme

Abt. Scientific Computing

Universität Stuttgart

Universitätsstraße 38

70569 Stuttgart

Bachelorarbeit

Was haben Schlafmusik-Stücke gemein? Eine empirische Analyse

Niclas Böttle

Studiengang: B.Sc. Informatik

1. Prüfer: Prof. Pflüger

2. Prüfer:

Betreuer: Prof. Pflüger, Jun.-Prof. Akkermann

begonnen am: 11.04.2022

beendet am: 11.10.2022

Inhaltsverzeichnis

1	Einleitung	2
1.1	Related Work	3
1.2	Vorgehen	4
1.3	Physikalische und musikalische Grundlagen	5
2	Datensatz und Features	10
2.1	Datensatz	10
2.2	Benutzte Features	15
2.2.1	Vorgefertigte Features	15
2.2.2	Nachimplementierte Features	17
2.2.3	Eigene Features	19
3	Analyse	36
3.1	Diskriminierende Features	36
3.1.1	Allgemeine Vergleiche	38
3.1.2	Genreinterne Vergleiche	43
3.2	Featureevaluation	48
3.3	Variationsarme Features	60
3.4	Diskussion	63
4	Fazit	65
A	Spotifyfeatures	67
B	CDs Schlaflieder aus aller Welt	68
C	Tabellen für Analyse	69

Zusammenfassung

In dieser Arbeit wurde Musik, welche zum Einschlafen gehört wird, mittels Features aus dem Bereich des Music Information Retrieval analysiert. Hierfür wurden teilweise neue Features entwickelt. Außerdem wurden zusätzlich Referenzmusik verschiedener Genres, therapeutische Schlafmusik und Schlaflieder verwendet, um die Ergebnisse einzuordnen. Dabei zeigt die untersuchte Schlafmusik ähnliche, wenn auch schwächer ausgeprägte Charakteristiken wie therapeutische Schlafmusik und Schlaflieder. Hauptsächlich Klangfarben konnten herausgefunden werden: Ein wenig voller, dunkler, sanfter, wenig rauschender Klang, welcher sich über das Stück vergleichsweise wenig verändert. Außerdem konnte herausgefunden werden, dass wenige Tonansätze charakteristisch sind. Andere musikalische Charakteristiken wie etwa ein niedriges Tempo konnten aufgrund von noch zu ungenauen, teilweise genreabhängigen Algorithmen, wenn überhaupt, nur in schwachen Ausprägungen gefunden werden.

1 Einleitung

Schlafprobleme nehmen in den westlichen Gesellschaften zu (18; 28; 51), obwohl Schlaf wichtig für unsere Gesundheit und damit auch für unsere Lebensqualität ist: So gibt es kausale Zusammenhänge zwischen Schlafproblemen und Fettleibigkeit (19) und Diabetes (20). Dennoch konsultieren viele Menschen mit Schlafproblemen keinen Arzt, sondern versuchen, sich mit Musik selbst zu helfen (13; 42; 47), wobei Musik tatsächlich den Schlaf von Erwachsenen verbessern kann (37; 67).

Es gibt bereits einige Studien, welche die beruhigende und schlafverbessernde Wirkung von Musik belegen. So kann mit Musik etwa der Cortisolpegel gesenkt (45) und der Oxytocinpegel gesteigert werden (50). Zudem kann Musik als subjektiv entspannend wahrgenommen werden (50) und körperliche Erregung vermindern (53). Außerdem wurden in Studien musikalische Parameter herausgefunden, welche beruhigend und schlaffördernd wirken: langsames Tempo (15; 34; 68), Dur, einfacher Rhythmus und legato Artikulation (24).

Der deutlichsten Zusammenhang zwischen Musik und Schlaf ist bei der Pflege von Babies und Kindern erkennbar: Seit Urzeiten singen Eltern ihren Kindern Schlaflieder vor, um diese zum Einschlafen zu bringen (46). Dabei sind Lullabies sehr repetitiv, haben ein geringes Tempo und einen einfachen Rhythmus (64; 46).

Dennoch ist noch nicht gut erforscht, welche Musik tatsächlich zum Einschlafen verwendet wird. Im folgenden Abschnitt werden erste Versuche dessen beschrieben und aufgegriffen.

1.1 Related Work

Die bislang größte Studie, welche die Musik, die Menschen zum Schlafen hören, adressiert, ist „*The music that people use to sleep: universal and subgroup characteristics*” (57). In ihr wurde mittels des Streaming Dienstes Spotify¹ ein großer Schlafmusik-Datensatz aus Playlisten, welche das Keyword *Sleep* im Titel oder in der Beschreibung hatten, aufgebaut und analysiert. Diese Arbeit von Scarratt wird als Grundlage für diese Arbeit gelten, wobei der Datensatz und eine Analysemethode verwendet werden und in der Auswertung die Ergebnisse verglichen werden. Anders als in (57) werden jedoch nicht die Spotifyfeatures, welche in Anhang A „Spotifyfeatures” vorgestellt werden, verwendet. Der Grund hierfür ist, dass die zugrundeliegenden Algorithmen nicht öffentlich sind und somit nicht klar ist, was die Features genau messen (32). Deshalb beschäftigt sich diese Arbeit mit anderen Features, welche teilweise neu entwickelt und vorgestellt werden.

Eine andere Arbeit (69) setzte sich zum Ziel, ein System zu entwickeln, welches Musik beruhigender und besser zum Einschlafen macht, und analysierte dafür ebenfalls Schlafmusik. Die Schlaflieder wurden zum einen analog zu (57) via Spotify ermittelt zum anderen Studien entnommen. Insgesamt werden allerdings deutlich weniger Lieder verwendet als in (57). Dabei wurden verschiedene Features verwendet, welche jedoch nicht zwangsweise direkt musikalisch und klanglich interpretierbar sind. Aus dieser Arbeit wurden teilweise Features übernommen und zur guten Ver-

¹<https://www.spotify.com/>

gleichbarkeit wurde dieselbe Analysemethode verwendet. Desweiteren wurden Songs aus den gleichen Studien verwendet.

Das nächste größere Forschungsfeld, welches mit dieser Analyse zusammenhängt, ist das der Genre Recognition. Dabei geht es darum, einen Song, welcher als Audio Datei oder Stream gegeben ist, einem aus vorher definierten Genres zuzuordnen. Hierfür werden meist low-level Features extrahiert und diese werden als Input für den Klassifizierungsalgorithmus verwendet (44). Hierbei sind besonders Features hilfreich, welche für jedes Genre unterschiedliche Ausprägungen haben, da mittels diesen Songs verschiedener Genres gut unterschieden werden können. Ähnlich zur Genre Recognition werden auch hier Referenzsongs verwendet, welche einerseits dazu dienen, diskriminierende Features zwischen Schlafmusik und anderer Musik herauszufinden und andererseits dazu genutzt werden, den normalen Wertebereich und die übliche Streuung von Features herauszuarbeiten. Einige der hier benutzten bereits existierenden Features, welche in Kapitel 2.2.1 „Vorgefertigte Features“ aufgelistet sind, wurden schon bereits in der Genre Recognition verwendet (66).

1.2 Vorgehen

Zunächst wurden die Songs ausgewählt, welche analysiert werden sollen. Dafür wurden repräsentative Songs aus dem Datensatz aus (57) verwendet. Um die Ergebnisse bewerten zu können, wurden wie in (69) Songs benutzt, welche in Studien therapeutische Wirkung bewiesen. Zusätzlich wurden zu einer weiteren Bewertung Schlaflieder verwendet: Schlaflieder aus der gesamten Welt und einige westliche Schlaflieder in verschiedenen Versionen. Die verschiedenen Versionen wurden verwendet, um einen Bias zu verhindern, da es sehr viele Versionen dieser Schlaflieder gibt. Für zehn Genres wurden außerdem Referenzsongs ausgewählt, um eine Abgrenzung möglich zu machen, sowie als Referenz für die Streuung der Features.

Diese Songs wurden auf verschiedene Features analysiert, wobei jedes Feature direkt klanglich oder musikalisch interpretierbar ist. Dafür wurden vorgefertigte Features verwendet, es wurden Features nachimplementiert und einige Features wurden eigens für diese Arbeit neu entwickelt.

Die Analyse wird von zwei Fragen bestimmt:

1. Was unterscheidet Schlafmusik von anderer Musik?
2. Welche Gemeinsamkeiten haben Schlafmusiklieder?

Die erste Frage liegt ebenfalls den beiden Arbeiten (57) und (69) zugrunde, weshalb hier jeweils eine Metrik wie in der jeweiligen Arbeit zu Vergleichszwecken verwendet wurde. In Folge dessen werden die Ergebnisse diskutiert und die benutzten Features bewertet.

1.3 Physikalische und musikalische Grundlagen

Im Folgenden werden einige Grundlagen gelegt, welche für das Verständnis der Funktionsweise und Aussagekraft der Features benötigt werden.

Musik und Klang liegen in der physikalischen Welt als Luftdruckwellen vor. Diese können erfasst werden, etwa von unseren Ohren oder Mikrofonen, oder erzeugt werden z.B. von unserer Stimme, Instrumenten oder Lautsprechern. Vereinfacht gilt, dass ein Ton als sinusförmige Funktion moduliert werden kann, wobei die Tonhöhe von der Frequenz angegeben wird: Je höher die Frequenz, desto höher die wahrgenommene Tonhöhe. Die Lautstärke wird durch die Amplitude der Funktion ausgedrückt: Je größer die Amplitude, desto lauter der Ton. Nach dem Interferenzprinzip können sich Wellen überlagern, ohne Informationen der anderen Funktionen zu zerstören. So werden mehrere Töne zusammen als Addition der einzelnen sinusförmigen Funktionen dargestellt. Der Mensch ist hierbei in der Lage, die verschiedenen Töne gleichzeitig zu hören.

Außerdem nimmt der Mensch Musik und Klänge auf eine spezielle Art und Weise wahr. Dies wird in Abbildung 1 illustriert: Phon ist eine psychoakustische Einheit und Töne mit gleichem Phonwert werden als gleich laut wahrgenommen. Dabei stellt drei phon die Schwellenlautstärke dar: die geringste Lautstärke, welche Menschen wahrnehmen können. Gut zu sehen ist, dass nur Frequenzen im Bereich 20Hz bis

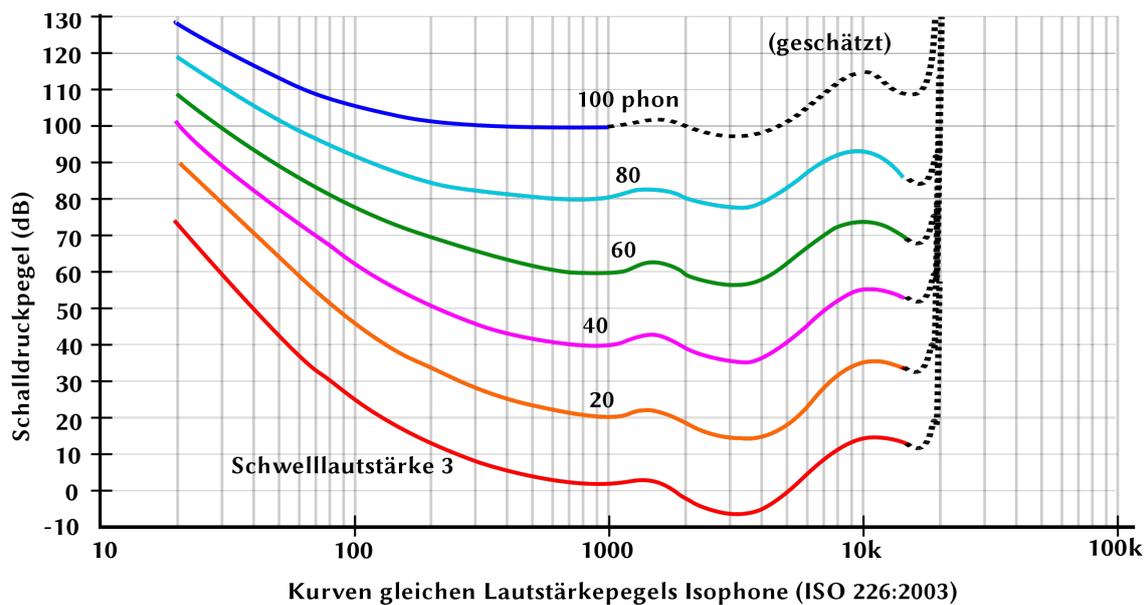


Abbildung 1: Kurven gleicher Lautstärkepegel nach (36)

https://commons.wikimedia.org/wiki/File:Kurve_gleicher_lautstärkte.svg

20kHz wahrgenommen werden können. Außerdem wird ersichtlich, dass die Randfrequenzen deutlich lauter sein müssen, um wahrgenommen zu werden als die mittleren Frequenzen. Die Randfrequenzen werden also schlechter wahrgenommen.

Insbesondere mit höherem Alter wird ein immer kleineres Spektrum immer schwächer wahrgenommen. Außerdem nehmen Kinder, insbesondere Säuglinge, Klänge und Lautstärken anders wahr als erwachsene Menschen.

Wie auch bei anderen Sinnen nimmt der Mensch Klänge auf logarithmische Weise wahr - So werden etwa zwei Töne mit Frequenzen, welche sich um den Faktor 2 unterscheiden, als gleichwertig wahrgenommen: Eine Oktave². Des Weiteren werden absolute Frequenzabweichungen je nach Startwert vollkommen anders wahrgenommen, während relative immer gleich wahrgenommen werden: Wird etwa zu einer Frequenz von 100 Hz 100 Hz addiert, so entspricht das der Wahrnehmung einer Oktave. Dagegen entspricht eine Addition von 100Hz auf die Grundfrequenz 200Hz einer Quinte. Man erhält immer eine Quinte, wenn man die ursprüngliche Frequenz mit dem Faktor 1,5 multipliziert.

²Oktaven und Quinten sind Intervalle, auf welche im Folgenden noch genauer eingegangen wird

Außerdem wird die Lautstärke logarithmisch wahrgenommen. So ist beispielsweise die bekannteste Lautstärkeneinheit Dezibel eine logarithmische Skala. 10 Dezibel entsprechen etwa einer wahrgenommenen Verdopplung der Lautstärke.

In dieser Arbeit wird angenommen, dass die untersuchte Musik den westlichen musikalischen Grundlagen entspricht, da die meisten Analysemethoden des Music Information Retrieval (MIR) auf dem westlichen System aufbauen. Mit wenigen Ausnahmen (wie etwa einer Rede) trifft dies für die analysierten Songs zu.

In der westlichen Musik gibt es Schläge und Takte. Diese stellen das rhythmische Gerüst der Musik dar: Takte bestehen aus einer festgelegten Anzahl von Schlägen, meist Vielfache von zwei oder drei. Die Dauer zwischen zwei Schlägen bestimmt die wahrgenommene Geschwindigkeit des Stückes und ist meist recht konstant. Dabei sind eine Beschleunigung (*accelerando*) oder eine Verlangsamung (*ritardando*) musikalische Mittel³ und oft als Ausdrucksweise und Interpretation eines Musikers zu verstehen, um den Ausdruck eines Stückes verstärken zu können. Sehr abrupte Tempowechsel sind selten und das Tempo bleibt meist für eine längere Zeit konstant. Takte bilden das Gerüst eines Musikstückes und häufig dauern Phrasen und Melodien ein, zwei oder vier Takte.

Als Ton wird hierbei ein Schallereignis mit definierter Tondauer und Tonhöhe betrachtet. Die Tondauer nennt man Notenwert: Beispielsweise entspricht eine Viertelnote einem Schlag und eine Achtelnote einem halben Schlag. Die westliche Musik bedient sich der oben beschriebenen Wahrnehmung und teilt eine Oktave in zwölf Tonhöhen auf, wobei diese auf einer logarithmischen Frequenzskala mit gleichem Abstand verteilt sind. Diese Tonhöhen spiegeln mit geringen Abweichungen die Obertonreihe wieder. Dabei sind Obertöne Töne mit Frequenzen, welche ganzzahlige Vielfache der Grundfrequenz sind. Diese sind bei jedem Ton eines Instruments vorhanden und bestimmen maßgeblich dessen Klang.

Die westliche Harmonielehre beinhaltet unter anderem Intervalle und Akkorde. Intervalle sind Tonunterschiede. Als Basis, um diese zu beschreiben, werden

³Accelerando und Ritardando sind nicht die einzigen Stilmittel, welche das Tempo verändern (*rubato*, *doppio movimento*, ...)

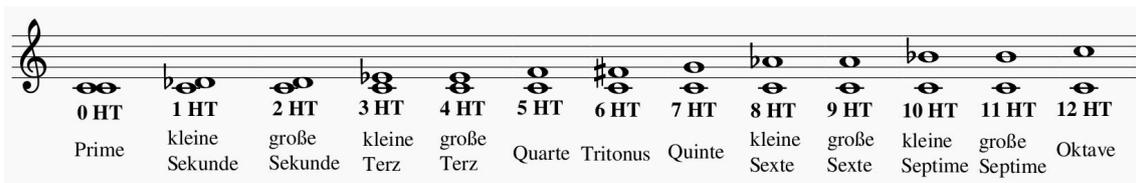


Abbildung 2: Die gängigsten Intervalle.

Halbtonschritte (HT) verwendet. Ein Halbtonschritt entspricht einem Schritt in der Zwölftonskala: Etwa von c zu cis oder e zu f. Die wichtigsten Intervalle sind in Abbildung 2 dargestellt.

Akkorde sind Schichtungen von Tönen. Die am meisten verwendeten Akkorde sind die Dur- und Moll-Akkorde, welche aus drei Tönen aufgebaut werden: Bei einem Dur-Dreiklang wird eine kleine Terz auf eine große geschichtet. Insgesamt bilden diese beiden Intervalle somit eine Quinte. Ein Moll-Dreiklang besteht dagegen aus einer Schichtung von einer großen Terz auf eine kleine Terz. Auch diese Intervalle bilden eine Quinte ⁴.

Bezeichnungen von Dur- und Moll-Akkorden wie etwa C-Dur können gleichermaßen eine Tonart angeben, wobei Tonarten das ganze Stück über oder abschnittsweise angegeben werden. Diese bestehen aus einer dazugehörigen Tonleiter, welche sich, bei Dur oder moll aus sieben verschiedenen Tönen zusammensetzt. Der erste und der darauffolgende achte Ton (die Grundtöne) bilden eine Oktave. Diese Tonleiter stellt eine Art Tonvorrat dar, welche im Stück oder im Abschnitt verwendet werden. Um diese Skalen zu erzeugen, werden Vorzeichen benötigt, welche einzelne Töne um einen Halbton erhöhen oder erniedrigen.

Im Quintenzirkel (zu sehen in Abbildung 3) werden zunächst Paare von Dur- und Moll-Tonarten gruppiert, welche die gleichen Vorzeichen haben: Diese werden parallele Dur- und Moll-Tonarten genannt. Analog dazu heißen die dementsprechenden Akkorde „parallele Akkorde“. Daraufhin wird neben einer Tonart die Tonart gestellt, deren Tonleiter erhalten wird, wenn ein Vorzeichen hinzugefügt wird: Nach

⁴Der hier beschriebene Aufbau nennt sich Grundstellung und ist nicht die einzige Stellung, welche den Akkord erzeugen kann

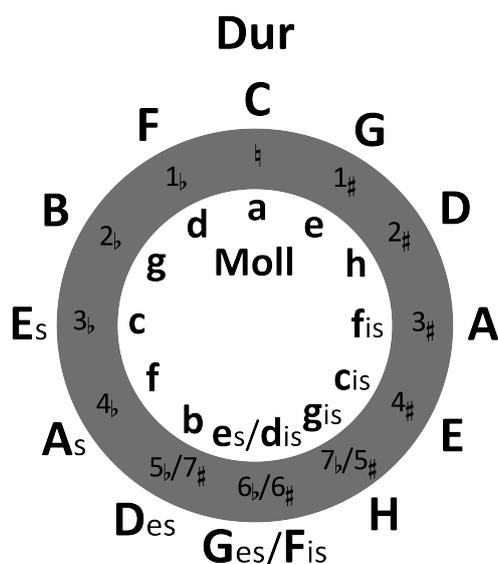


Abbildung 3: Quintenzirkel

rechts wird ein zusätzlicher Ton erhöht, wodurch das Intervall benachbarter Tonarten einer Quinte entspricht. Diese benachbarte Tonarten heißen „quintverwandt“. Je näher zwei Tonarten im Quintenzirkel beieinanderliegen, desto stärker sind sie miteinander verwandt. Entsprechendes gilt auch für Akkorde. Nach links wird ein Ton erniedrigt, wodurch ein Intervall einer Quarte zwischen benachbarten Tonarten entsteht. Nach zwölf Schritten in dieselbe Richtung endet man bei der Tonart, bei welcher begonnen wurde.

Dies ist eine extrem vereinfachte Sicht auf das westliche Musiksystem: Es existieren noch viele andere Arten von Akkorden, sowie verschiedene Skalen, Taktarten etc. Da die meisten Stücke jedoch zu großen Teilen auf dem oben beschriebenen System basieren und die Programme für diese Systeme bessere Ergebnisse erzielen (48), werden im MIR wie auch hier meist diese Vereinfachungen verwendet. Dabei treten aber beispielsweise in der Akkorderkennung Probleme auf: Zum Beispiel wenn Akkorde, die aus vier Tönen bestehen, einem der 24 Dur- oder Moll-Akkorde zugeordnet werden sollen. Eine ausführlichere Einführung in die Gesetzmäßigkeiten westlicher Musik für MIR kann in (48) gefunden werden, zusätzlich zu Verweisen zu genaueren Darstellungen.

2 Datensatz und Features

Um Schlafmusik analysieren zu können, werden zunächst Musik und Features ausgewählt und vorgestellt. Die daraus resultierenden Featurevektoren für jedes Lied werden in Kapitel 3 „Analyse“ betrachtet. Außerdem werden die Ergebnisse sowie die Features diskutiert.

2.1 Datensatz

Der Datensatz der untersuchten Musikstücke teilt sich in vier Teile auf:

Den ersten Teil stellt dabei der Datensatz und die Vorarbeit von Scaratt dar (57). Im Folgenden werden die benutzten Repräsentanten beschrieben, da das gesamte Datensatz zu groß wäre: Im Gegensatz zu (57), in welchem die Features via Spotify-API Aufruf generiert werden konnten, muss hier jeweils der gesamte Song analysiert werden. Bei einer Gesamtlänge aller Songs von etwa 500 Tagen ist somit eine komplette Einbeziehung des Datensatzes in dieser Arbeit nicht möglich.

Es wurden jeweils die 100 Songs verwendet, welche am nächsten an den Clustermittelpunkten und dem Mittelpunkt des Datensatzes liegen. Für diese Berechnung wurden die beim Clustering verwendeten normalisierten Spotify Features verwendet: Zunächst wurde der Mittelwert des Features abgezogen, um das Feature bei 0 zu zentrieren, anschließend wurde durch die Standardabweichung geteilt, um Abstände richtig darzustellen und kein Feature mit kleinen Standardabweichungen zu benachteiligen.

$$\hat{x}_i = \frac{x_i - \bar{x}_i}{s_i}$$

mit x_i i-tes Spotifyfeature, \bar{x}_i dessen Mittelwert und s_i dessen Standardabweichung.

Der Abstand, welcher überprüft wurde, ist dabei der summierte quadrierte Fehler.

$$\sum_{i=1}^9 (\hat{x}_i - \hat{c}_i)^2$$

mit dem normalisierten Clustermittelpunkt \hat{c} .

Es traten trotz Deduplikation in Bezug auf die Song IDs Dopplungen auf und es waren nicht mehr alle Songs auf Spotify verfügbar. Deshalb wurden hierbei insgesamt 654 Songs verwendet:

- Cluster 1: Live Tracks (C1): 94 Songs
- Cluster 2.1: Ambient Tracks Slow (C2.1): 91 Songs
- Cluster 2.2: Ambient Tracks Fast (C2.2): 97 Songs
- Cluster 3: Radio Tracks (C3): 91 Songs
- Cluster 4: Acoustic Radio Tracks (C4): 92 Songs
- Cluster 5: Speechy Tracks (C5): 89 Songs
- Cluster 6: Instrumental Tracks (C6): 100 Songs

Außerdem wurden auf dieselbe Art und Weise die 100 Songs, welche am nächsten zum Featurevektor mit elementweisen Mittelwerten des gesamten Datensatzes liegen, extrahiert (Center Tracks Mean (CTMean): 92 Songs). Ebenso wie diese, welche am nächsten zum Featurevektor mit elementweisen Medianen liegen (Center Tracks Median (CTMed): 91Songs). Es wurden Mittelwert und Median verwendet, da sich diese teilweise stark unterscheiden: Bis zu etwa 0.6 Standardabweichungen pro Feature.

Um einen möglichst guten Einblick in den gesamten Datensatz zu erhalten, wurden zusätzlich die Songs ausgewählt, welche sich am weitesten von dem Mittelwert und Median Mittelpunkten entfernt befinden. Es wurde hier analog wie bereits zuvor gearbeitet. Da die resultierenden Ergebnisse jedoch sehr ähnlich waren, wurden sie in einen Datensatz zusammengefasst: Alle Songs beider Playlisten wurden aneinandergesetzt und daraufhin wurde Deduplikation ausgeführt (Distant Tracks (DT): 91 Songs).

Als zweiter Teil wurden wie in (69) Songs untersucht, welche sich in bisherigen Studien als gut zum Einschlafen erwiesen haben (24; 68). Diese dienen als Referenz für gute Schlafmusik: Therapeutical Sleep Music (TSM): 27 Songs

Der dritte Teil besteht aus Schlafliedern, welche wie TSM dazu dienen sollen, die Songs aus dem Datensatz zu bewerten. Dafür wurde eine CD Sammlung des Autors von Schlafliedern aus der ganzen Welt, bestehend aus 7 Alben des Labels Ellipsis Arts, verwendet. Genaue Informationen zu den CDs sind in Anhang B „CDs Schlaflieder aus aller Welt“ zu finden: World Lullaby (WL): 114 Songs

Um populäre und klassische Schlaflieder zu berücksichtigen, wurden die Songs „*Der Mond ist aufgegangen*“, „*Twinkle, Twinkle, Little Star*“, „*Schlaf, Kindlein, schlaf*“ und „*Slaap, kindje slaap*“ ausgewählt. Um einem Bias vorzubeugen, wurden verschiedene Versionen des gleichen Songs verwendet. Da für jeden dieser Songs sehr viele Versionen auf Spotify existieren, wurde der Name des Songs in die Spotify Desktop Songsuche eingegeben. Daraufhin wurden die ersten 50 Treffer mittels deren Namen darauf überprüft, ob der gewünschte Song enthalten ist ⁵.

- Der Mond ist aufgegangen (Mond): 42 Songs
- Twinkle, Twinkle, Little Star (Twinkle): 45 Songs
- Schlaf, Kindlein, schlaf (Schlaf): 40 Songs
- Slaap, kindje slaap (Slaap): 38 Songs

Der letzte Teil stellt Referenzmusik dar. Diese dient zur Bestimmung der üblichen Wertbereiche, in welchen die Features Ausprägungen besitzen können, und zur Abgrenzung. Da die häufig verwendeten Datensätze, wie etwa in (43; 60) genannt, für diesen Zweck zum großen Teil unbrauchbar sind, wurde die Referenzmusik eigens herausgearbeitet. GTZAN (66) welches in (69) verwendet wird, hat viele Probleme wie Song- und Künstler-Duplikate (59). Viele andere Datenbanken liefern nur

⁵Dabei ist zu erwähnen, dass die meisten dieser Songs für Erwachsene aufgenommen wurden und nicht für Babies

Features, was hier ungenügend ist, da u.a. selbsterzeugte Features verwendet werden. Viele sind dann entweder deutlich zu groß, sodass vor Benutzung eine Auswahl stattfinden müsste, oder liefern nur sehr kurze Abschnitte von Songs.

Es wurden für die zehn Genres, welche in GTZAN aufgeführt werden, jeweils 50 Songs auf folgende Art und Weise herausgesucht: Zunächst wurde wie in (57) mit der Spotify Desktop Playlisten-Suche und dem Genrenamen als Suchwort Playlisten gesucht. Es wurden, um einen Bias zu verhindern, die 10 Playlisten mit den meisten „Gefällt mir“-Angaben verwendet. Nur Playlisten, welche einen klaren Bezug zum Genre im Titel oder Beschreibung haben, wurden verwendet: So wurde beim Genre Reggae beispielsweise Reggaeton Playlisten ausgeschlossen. Insgesamt haben alle Playlisten mindestens 90000 „Gefällt mir“-Angaben, wobei die meisten deutlich mehr haben.

- Reference Blues (RB): 50 Songs
- Reference Classical (RCI): 50 Songs
- Reference Country (RCo): 50 Songs
- Reference Disco (RD): 50 Songs
- Reference Hip-Hop (RHH): 50 Songs
- Reference Jazz (RJ): 50 Songs
- Reference Metal (RM): 50 Songs
- Reference Pop (RP): 50 Songs
- Reference Reggae (RRe): 50 Songs
- Reference Rock (RRo): 50 Songs

Hierbei ist anzumerken, dass 94 % der Playlisten von Spotify selbst erstellt wurden. Inwieweit Spotify hiermit den globalen Musikgeschmack prägt, wäre eine interessante Fragestellung für zukünftige Studien. Nun wurden mit dem Spotify Feature

„popularity“ jeweils die 50 Songs aus den Playlisten ausgewählt, welche aktuell am häufigsten gehört werden. Dabei ist zu beachten, dass aktuelle Trends stark gewertet werden. Dadurch spiegeln die Resultate jedoch gut wider, was zu dem Zeitpunkt wirklich am meisten gehört wird. Beispielsweise sind 26 der 50 Tracks aus dem Genre Pop aus 2022 (dem Jahr des Entstehens dieser Arbeit). Der stärkste zeitliche Ausreißer „*Running Up That Hill (A Deal With God) – 2018 Remaster*“ von *Kate Bush* aus dem Jahr 1985 erfreut sich zum Zeitpunkt dieser Arbeit aufgrund seines prominenten Auftritts in der populären neu erschienenen vierten Staffel der Netflix ⁶ Serie „*Stranger Things*“ großer Beliebtheit. Die Verteilung der Erscheinungsjahre ist in Abbildung 4 illustriert.

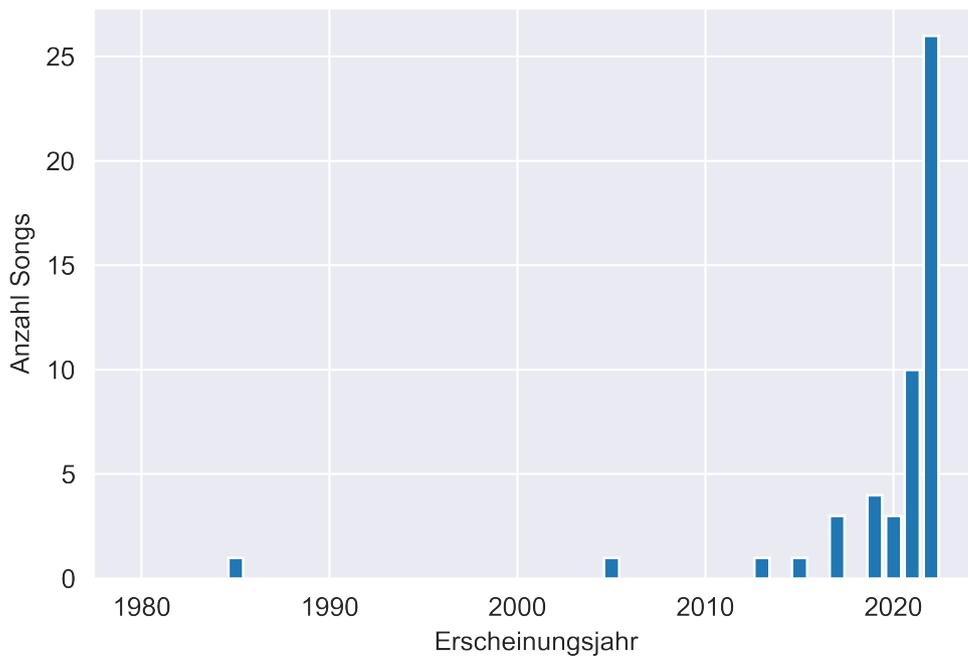


Abbildung 4: Verteilung der Erscheinungsjahre der Songs aus Reference Pop

⁶<https://www.netflix.com>

2.2 Benutzte Features

Im Folgenden werden nur Features betrachtet, welche eine direkte Interpretation bezüglich Klang oder Musik zulassen. Aus diesem Grund werden etwa MFCCs, welche etwa bei Genre Recognition sehr beliebt sind (44) und die als Indikator für Klangfarbe und Instrumentation verwendet werden (48; 49), im Folgenden nur indirekt verwendet, da einzelne MFCC Bänder keine direkte Interpretation ermöglichen.

2.2.1 Vorgefertigte Features

Mittels des Commandline-Toolkits „Music extractor“ der Library Essentia (16) wurden die folgenden Features extrahiert, welche meist low-level features sind. Für Features, welche für jedes Sample ⁷ einzeln berechnet werden, wird pro Song der Mittelwert (mean) und die Standardabweichung (std) analysiert.

- **average_loudness** (3; 10; 58) ⁸ gibt an, wie stark sich die Lautstärke innerhalb des Stückes verändert (0 entspricht großem dynamischen Umfang, 1 geringem). Die tatsächliche Lautstärke zu messen hätte keinen großen Effekt, da Spotify all seine Songs mittels einer loudness-normalization auf -14dB LUFS (Einheit für Lautstärke, die sich am menschlichen Gehör orientiert) normt (5).
- **barkbands_spread** (1; 7; 2) dient als Indikator für einen vollen Klang (1 ist voller Klang, 0 nicht) (6) ⁹. (Mean und std)
- **spectral_centroid** (4; 8; 52; 40) beschreibt den Schwerpunkt des Frequenzspektrums. Dies lässt sich als Helligkeit eines Songs interpretieren. Je höher diese Frequenz, desto heller die Klangfarbe (66; 6). (Mean und std)
- **spectral_flux** (65; 25; 9) lässt sich als Indikator für die Rauheit eines Klangs interpretieren, wobei hohe Werte für starke Rauheit stehen (6). (Mean und std)

⁷Kurzer Abschnitt, welcher einzeln betrachtet wird (wie etwa bei STFT)

⁸Die Quellen direkt nach den Featurenamen sind eine Vorschrift von Essentia und werden in deren Dokumentation zu diesen Features zitiert

⁹Die Quellen nach den Interpretationen sind die Quellen, welchen diese Interpretationen entnommen wurden

- **spectral_spread** (7; 2) gibt an, ob ein Klang rauschend ist oder eine definierte Tonhöhe besitzt. Ein hoher Wert steht hierfür für einen hohen Rauschanteil (6). (Mean und std)
- **spectral_rolloff** (52) gibt die Frequenz an, unter welcher 85 % der Energie des Signals liegt. So kann hiermit die Verteilung von Bass und Höhen überprüft werden (69) (Mean und std)
- **zerocrossingrate** (11; 52) ist ein Indikator für rauschende Klänge (66). Dabei bedeutet ein hoher Wert einen hohen Rauschanteil. (Mean und std)
- **onset_rate**: Als Onset wird der Beginn eines neuen Tones bezeichnet. Mittels Onset Detection (14; 17; 26; 41) werden diese Beginne extrahiert und daraus die **onset_rate** bestimmt: Die durchschnittliche Anzahl der Onsets pro Sekunde. Dies kann musikalisch etwa mit Notenwerten und Tempo zusammenhängen: Kleine Notenwerte und hohes Tempo erzeugen eine hohe **onset_rate**.
- **bpm (Beats per Minute)** (23; 70) beschreibt das Tempo der Musik. Ein hoher Wert entspricht einem hohen Tempo. Der hier verwendete Algorithmus erzeugt jedoch zu hohe **bpm** Werte, da die **bpm** algorithmisch leicht verdoppelt oder verdreifacht werden kann, wie in Kapitel 3.2 „bpm“ gezeigt wird.
- **chords_number_rate** gibt das Verhältnis von unterschiedlichen Akkorden im Stück an. In Kapitel 2.2.3 „Features basierend auf dem Chords Histogramm“ wird auf den Chord-Detector eingegangen, welcher für **chords_number_rate** benutzt wurde.
- **chords_change_rate** gibt die Rate an, mit welcher sich die Akkorde im Stück verändern. In Kapitel 2.2.3 „Features basierend auf dem Chords Histogramm“ wird auf den Chord-Detector eingegangen, der für **chords_change_rate** verwendet wurde.

2.2.2 Nachimplementierte Features

Artikulation beschreibt die Art und Weise wie Töne gespielt, gestaltet und miteinander verbunden werden. Die beiden prominentesten Formen sind hierbei staccato (abgesetzt) und legato (gebunden). Artikulation ist ein wichtiger Bestandteil von Musik, welcher bestimmt, wie wir Musik wahrnehmen: So können etwa die Stimmungen Fröhlichkeit und Ärger sowie Traurigkeit und Angst durch die Unterscheidung zwischen Staccato und Legato unterschieden werden (38). Auch im Bezug auf Musik zum Schlafen wurde bereits festgestellt, dass sich Songs mit Legato-Artikulation besonders gut zum Schlafen eignen (24), weshalb das folgende Feature betrachtet wird.

average_silence_rate gibt die Prozentzahl der als still empfundenen Samples pro Sekunde an. Dies ist ein Indikator, ob staccato oder legato vorliegt: Unter 0.15 legato, über 0.15 staccato (27). Dafür wurde die folgende Formel aus (27) mittels Librosa¹⁰ implementiert:

$$\frac{1}{2N} \sum_{n=0}^{N-1} 1 - \text{sgn}(\text{STE}(n) - p * \overline{\text{STE}})$$

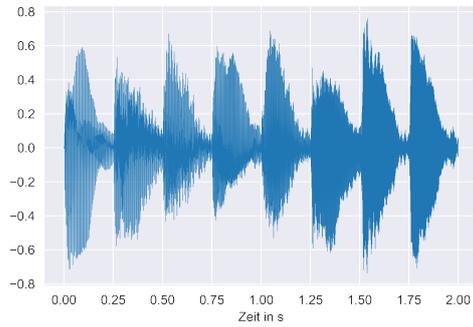
mit $p=0.25$, $\text{STE}(n)$ Short Time Energy an Frame n ¹¹, N entspricht Anzahl an Samples in einer Sekunde ($N=43$, da sampling rate 22050 Hz und 512 pro Frame, analog zu (66)), $\overline{\text{STE}}$ arithmetisches Mittel der Short Time Energy in der analysierten Sekunde

Da die **average_silence_rate** für jede Sekunde eines Stückes einzeln berechnet wird, wird das arithmetische Mittel (mean) und die Standardabweichung (std) verwendet. Die **average_silence_rate** wurde schon in (69) verwendet und ein bis auf kleine Änderungen ($p=1$ und RNS Energy anstatt STE) identisches Feature wurde in (66) als „Low-Energy Feature“ vorgestellt.

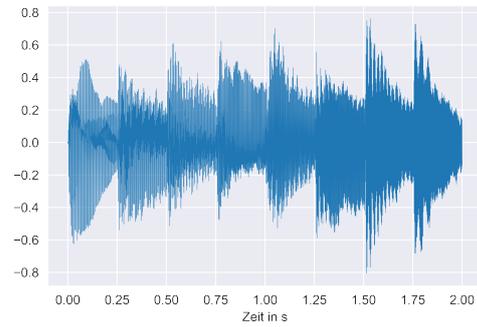
In Abbildung 5 sind die Wavefunktionen für kurze einstimmige Staccato- und Legatopassagen verschiedener Instrumente zu sehen (Klavier, Klarinette und Violine,

¹⁰<https://librosa.org/>

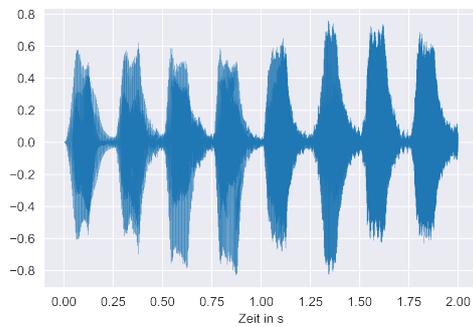
¹¹Short Time Energy entspricht der Summe aller quadrierten Werten der Wavefunktion eines Frames



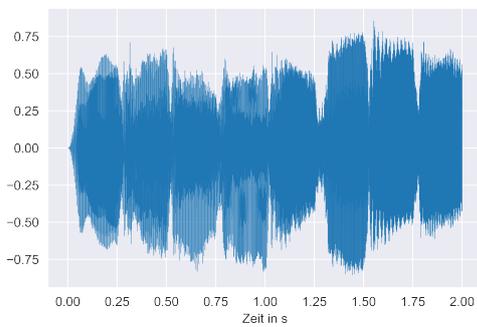
(a) Staccato Klavier
 mean = 0.2907, std = 0.0349



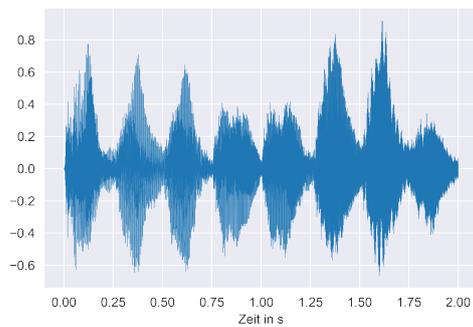
(b) Legato Klavier
 mean = 0.0117, std = 0.0117



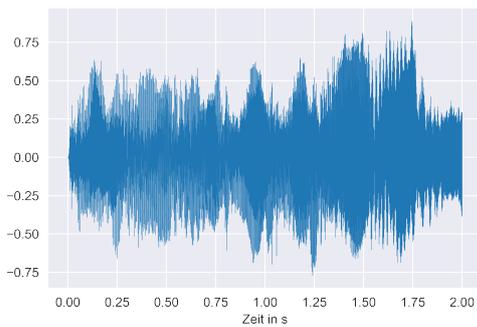
(c) Staccato Klarinette
 mean = 0.4070, std = 0.0117



(d) Legato Klarinette
 mean = 0.0465, std = 0.0



(e) Staccato Violine
 mean = 0.2558, std = 0.0233



(f) Legato Violine
 mean = 0.0233, std = 0.0233

Abbildung 5: Beispiele für average_silence_rate

jeweils 2 Sekunden). Außerdem die dazugehörige mittlere `average_silence_rate`. Je nach Instrument unterscheiden sich die Werte zwar deutlich, jedoch ist die in (27) vorgestellte Trennung an 0.15 deutlich zu erkennen. Die Klangbeispiele wurden mit Musescore3 ¹² erzeugt.

2.2.3 Eigene Features

Es gibt noch nicht zu jedem Bereich der Musik, welche zum Einschlafen relevant sein könnte, passende MIR Features. Deshalb werden im Folgenden einige neue Features entworfen, welche auf bereits bestehenden aufbauen, und werden mit Beispielen illustriert.

Features basierend auf dem Chords Histogramm Harmonik ist einer der Grundbausteine westlicher Musik. Daher liegt die Vermutung nahe, dass Harmonik bei Schlafmusik eine Rolle spielen kann. Außerdem wurde in (24) herausgefunden, dass Songs in Dur gut zum Einschlafen geeignet sind. Deshalb wird die Harmonik im Folgenden mit zwei Features beschrieben.

Essentia liefert ein sogenanntes Chords Histogramm. In diesem sind Häufigkeiten der Dur- und Moll-Akkorde im Quintenzirkel in Prozent angegeben. Ausgehend von diesem wurden die beiden Features `major_minor_ratio` und `harmonic_complexity` erzeugt.

In dem Chords Histogramm treten einige Fehler auf, welche die Genauigkeit der Features beeinträchtigen. Deswegen wird nun auf den Chords Detection Algorithmus eingegangen, welchen Essentia nutzt und welcher in deren Dokumentation schon als qualitativ „experimentell“ gekennzeichnet wurde.

Die beiden in der Essentiadokumentation hinterlegten Paper (31; 62) basieren beide auf dem Krumhansl-Schmuckler Algorithmus, welcher Tonarten für MIDI Dateien bestimmen kann. Das Paper von Gomez (31) zeigt eine Möglichkeit auf, wie dieser Algorithmus für Audiodateien zur Chord-Detection angewandt werden kann,

¹²<https://musescore.org/de>

während Temperly (62) den Algorithmus an sich verbessert. Ob für die Implementierung in Essentia alle Verbesserungen von Temperly mittels der von Gomez vorgestellten Implementierung für Audiodateien umgesetzt wurden, ist nicht ersichtlich.

Dies ist insbesondere interessant bezüglich der Frage, ob die von Temperly berücksichtigte Historie verwendet wurde: Akkorde hängen meist stark zusammen und sind kontextabhängig, weshalb in state-of-the-art-Systemen Techniken wie hidden Markov Models verwendet werden (48). Die Implementierung von Gomez (und auch der Krumhansl-Schmuckler Algorithmus) sind lediglich Pattern-Matching-Ansätze ohne die Historie zu verwenden:

Zunächst wird ein Chromagramm generiert: Dieses wird mittels der Kurzzeit-Fourier-Transformation (STFT) erzeugt und entspricht einer Funktion, welche Tonhöhen auf eine gewisse Zeit betrachtet jeweils einen Wert zuweist, der beschreibt wie viel diese Tonhöhe in dem Zeitabschnitt (Frame) im Audiosignal vorhanden ist. Dafür wird eine sogenannte Windowfunktion¹³ mit der eigentlichen Wavefunktion multipliziert und mit einer diskrete Fast Fouriertransformation verarbeitet. Das Ergebnis hieraus sind dann die Koeffizienten für die verschiedenen Tonhöhen für einen einzelnen Zeitabschnitt. Dieser ist durch die Länge der Windowfunktion und die Abtastrate gegeben. Daraufhin wird die Windowfunktion um einen konstanten Wert (die Hopsiz) verschoben¹⁴.

Im Folgenden wird das Chromagramm in Chromafeatures umgewandelt. Dazu wird jede Frequenz einem der zwölf Töne zugeordnet und so die Verteilung der Vorkommen der zwölf Töne aufsummiert. Für jeden Frame wird nun ein Vergleich zwischen dem dazugehörigen Chromafeaturevektor und dem davor definierten Referenz Chromafeaturevektoren für die 24 Dur- und Moll-Akkorde angestellt. Jedem Frame wird der Akkord zugeordnet, welcher in diesem Vergleich den besten Wert erzielte.

Die erkannten Fehler sind etwa, dass sehr viele Akkorde im Chords Histogramm kleine Häufigkeiten über 0 % haben, was auf falsche Chordzuweisung hindeutet.

¹³Eine Funktion, welche nur in einem kleinen Bereich ungleich 0 ist. Weitere Anforderungen und Beispiele für Windowfunktionen sind in (48) zu finden

¹⁴Die benutzten Parameter sind die Standardparameter in https://essentia.upf.edu/streaming_extractor_music.html

Dies könnte auf einen benutzten Algorithmus mit fehlender oder unzureichender Historienbetrachtung hinweisen. Um damit umzugehen, wurden alle Häufigkeiten kleiner als 1 % auf 0 % gesetzt und daraufhin wieder auf 100 % skaliert. Damit werden allerdings auch Akkorde, welche tatsächlich nur sehr selten vorkommen, unterschlagen. Dennoch überwiegen die Vorteile. Insbesondere mit der Annahme, dass die meisten Songs aus einer kleinen Anzahl an Akkorden aufgebaut sind.

Jedoch sind noch weitere Fehler erkennbar, wie eine fälschliche Durerkennung von Moll-Akkorden. Dies wird beispielsweise in dem Song „*Shape of You*“ von *Ed Sheeran* deutlich, welcher zu großen Teilen aus den Akkorden fis-Moll und cis-Moll besteht. Jedoch werden zu etwa 9 % die Akkorde Cis-Dur und Fis-Dur erkannt, welche in diesem Stück nicht vorkommen. Dieses Problem rührt wahrscheinlich aus dem vierten Oberton des Grundtons. Dieser entspricht einer großen Terz zum Grundton, welche charakteristisch für Dur-Akkorde ist (48). Dieses Problem wurde jedoch nicht behoben, da nicht ausgeschlossen werden kann, dass die Moll Akkorde richtig erkannt wurden und nicht das Ergebnis einer Moll-Dur-Verwechslung sind.

Allgemein gilt, dass eine wie im Chords Histogramm genutzte Zuweisung zu Dur- und Moll-Akkorden eine starke Vereinfachung ist und hieraus Fehler entstehen können: Etwa da nur Dur- und Moll-Akkorde erkannt werden können und Akkorde bestehend aus mehr Tönen wie etwa Sept-Akkorde nicht. Ebenfalls können nicht alle aus drei Tönen bestehende Akkorde erkannt werden wie etwa verminderte Akkorde. Wenn solche Akkorde auftreten, müssen sie dennoch einem Dur- oder Moll-Akkord zugewiesen werden, was einerseits zu verschiedenen Möglichkeiten pro Akkord führt (48) andererseits musikalisch nicht immer sinnvoll ist.

major_minor_ratio Die hiermit von mir vorgestellte `major_minor_ratio` gibt an, zu wieviel Prozent ein Song aus Dur-Akkorden im Vergleich zu Moll-Akkorden besteht. Weil $p_{Dur} + p_{Moll} = 1$ gilt, da der Chorddetectionalgorithmus nur die 24 Dur- und Moll-Akkorde erkennt, ist die `major_minor_ratio = pDur`. Die Berechnung erfolgt auf dem bereinigten Chords Histogramm.

Dies ist ein Indikator dafür, welche Stimmung ein Song hat. So wird Dur mit Fröhlichkeit und Moll mit Traurigkeit assoziiert (33; 55; 54).

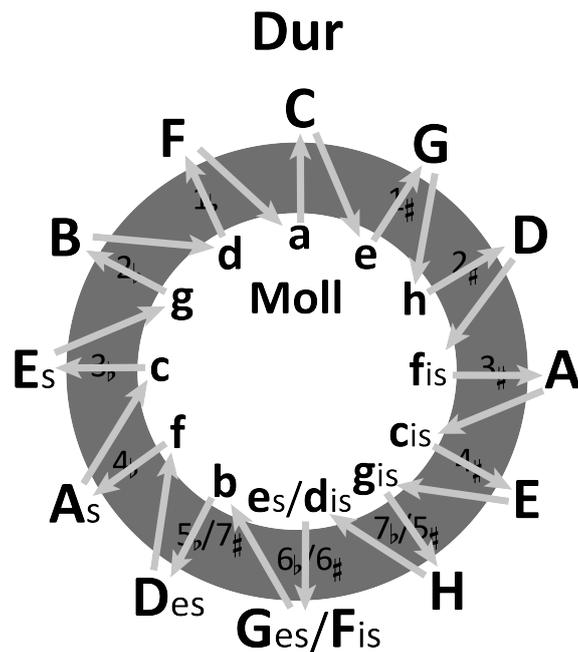


Abbildung 6: Reihenfolge im Quintenzirkel

harmonic_complexity Es werden vereinfachte Annahmen getroffen:

1. Das Stück steht in **einer** Tonart.
2. Ein Stück gilt als harmonisch komplex, wenn viele Akkorde auftreten, welche wenig verwandt sind mit dem Akkord, welcher der Tonart entspricht. Verwandtschaft bedeutet hier Quintverwandtschaft sowie Parallelverwandtschaft.

Das Chords Histogramm liefert die Akkorde in der Reihenfolge des Quintenzirkels. Der erste Eintrag stellt dabei die Tonart des Stückes dar. Diese Reihenfolge ist in Abbildung 6 illustriert. Um aus dem Chordshistogramm die harmonische Komplexität herauszuarbeiten, wird eine gewichtete Summe gebildet. Die Gewichte korrespondieren mit dem Verwandtschaftsgrad ¹⁵ und sind in Tabelle 1 aufgelistet.

¹⁵Diese Gewichte sind stark vereinfacht und folgen der Einschätzung des Autors. Dabei wurden zwei Annahmen getroffen: 1) Quintverwandtschaft ist enger als Parallelverwandtschaft 2) Quintverwandtschaft im Uhrzeigersinn im Quintenzirkel ist enger als gegen den Uhrzeigersinn. Die Gewichtung ist insofern musikalisch nicht sinnvoll, als dass je nach Genre verschiedenen Akkorden, verschiedene Wichtigkeiten beigemessen wird.

Index	Gewichtung Dur	Gewichtung Moll
0	0	0
1	5	3
2	1	1
3	7	5
4	3	3
5	9	7
6	5	5
7	11	9
8	7	7
9	13	11
10	9	9
11	15	13
12	12	12
13	13	15
14	10	10
15	11	13
16	8	8
17	9	11
18	6	6
19	7	9
20	4	4
21	5	7
22	2	2
23	3	5

Tabelle 1: Gewichte in Abhängigkeit zur Grundtonart in Dur bzw Moll

$$\text{harmonic_complexity} = \frac{1}{175} * \sum_{a \in \text{Akkorde}} w(g, a) * p(a)$$

mit $w(g, a)$ dem Gewicht des Akkordes a in Abhängigkeit zur Grundtonart g und $p(a)$ dem Vorkommen des Akkordes a

Der Faktor $\frac{1}{175}$ ergibt sich aus

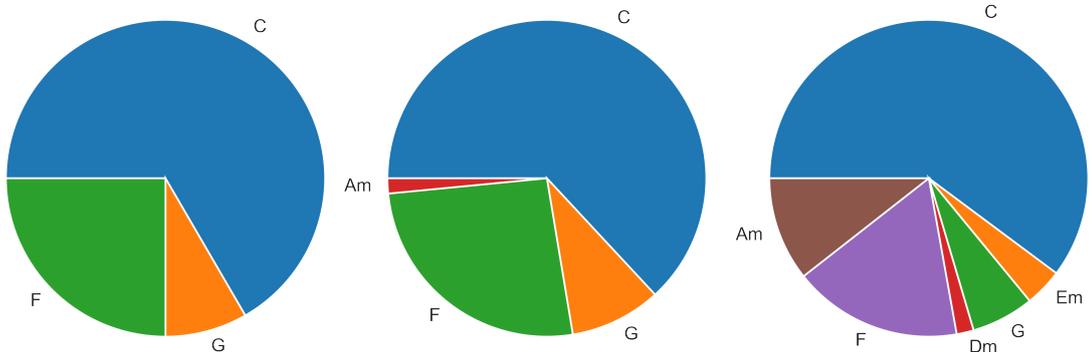
$$\sum_{a \in \text{Akkorde}} w(g, a) = 175$$

Ein großer Wert steht für hohe harmonische Komplexität, ein kleiner für niedrige.

Um die Features zu demonstrieren wurden folgende Beispiele mittels Musescore3¹⁶ erzeugt. Dabei liegt jeder Akkord einen Takt lang an. Das Tempo ist 120 bpm:

- In Abbildung 7a ist die optimale Akkordverteilung eines Blueschemas in C zu sehen und die dazugehörigen Werte für `major_minor_ratio` und `harmonic_complexity`. Da im Blueschema keine Moll-Akkorde vorkommen, gilt `major_minor_ratio` = 1 und da F-Dur und G-Dur zu C-Dur benachbart sind, ist `harmonic_complexity` sehr niedrig.
- In Abbildung 7b ist die Akkordverteilung eines Blueschemas in C zu sehen, welche durch Essentia erzeugt wurde, und die dazugehörigen Werte für `major_minor_ratio` und `harmonic_complexity`. Vergleichend zum optimalen Fall ist zu sehen, dass A-Moll fälschlicherweise erkannt wurde und dadurch `major_minor_ratio` geringer wurde. `harmonic_complexity` wurde größer.
- In Abbildung 7c ist die Akkordverteilung eines Blueschemas in C, bestehend aus gebrochenen Dreiklängen, zu sehen, welche durch Essentia erzeugt wurde. Außerdem die dazugehörigen Werte für `major_minor_ratio` und `harmonic_complexity`. Vergleichend zum optimalen Fall und dem vorherigen wurden mehr Akkorde falsch erkannt, wodurch sowohl die `major_minor_ratio` als

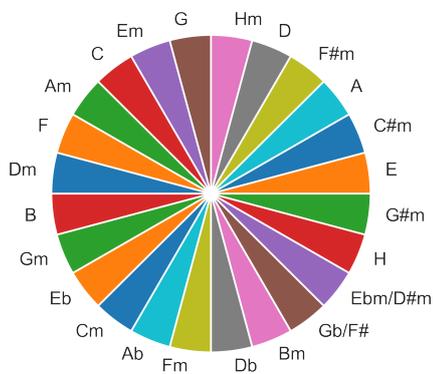
¹⁶<https://musescore.org/de>



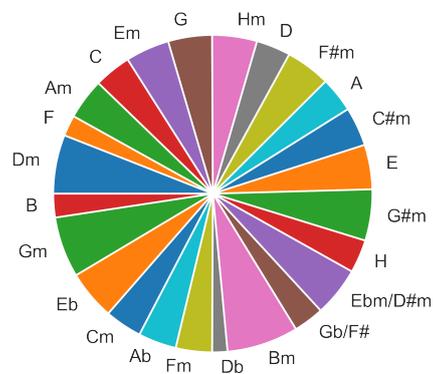
(a) Bluesschema optimal
 major_minor_ratio = 1.0
 harmonic_complexity = 0.0033

(b) Bluesschema
 major_minor_ratio = 0.9846
 harmonic_complexity = 0.0038

(c) Bluesschema mit gebrochenen Dreiklängen
 major_minor_ratio = 0.8378
 harmonic_complexity = 0.0057



(d) Quintenzirkelschema optimal
 major_minor_ratio = 0.5
 harmonic_complexity = 0.0417



(e) Quintenzirkelschema
 major_minor_ratio = 0.4126
 harmonic_complexity = 0.0402

Abbildung 7: Beispiele Features basierend auf dem Chords Histogramm

auch `harmonic_complexity` stark verfälscht wurden. Dass gebrochene Akkorde schwieriger zu erkennen sind, ist keine Überraschung, da hierbei die Töne in unterschiedlichen Samples liegen können.

- In Abbildung 7d ist die optimale Akkordverteilung eines Schemas, in welchem jeder Akkord des Quintenzirkels genau gleich oft vorkommt, zu sehen und die dazugehörigen Werte für `major_minor_ratio` und `harmonic_complexity`. Da Dur und Moll-Akkorde gleichviel vorkommen, gilt `major_minor_ratio` = 0.5 und die `harmonic_complexity` ist um eine Größenordnung größer als in den vorherigen Beispielen.
- In Abbildung 7e ist die Akkordverteilung eines Schemas, in welchem jeder Akkord des Quintenzirkels genau gleich oft vorkommt, zu sehen, welches mittels Essentia erzeugt wurde, und die dazugehörigen Werte für `major_minor_ratio` und `harmonic_complexity`. Insbesondere ist zu sehen, dass fälschlicherweise mehr Moll-Akkorde zu beobachten sind.

Features basierend auf der Melodie Kontur Die Melodie ist ein wichtiger Baustein eines Stückes. Dementsprechend nimmt sie eine relevante Rolle in der Wahrnehmung eines Songs ein. So sind beispielsweise Schlaffieder durch Melodiewiederholungen und eine einfache Melodiekontur geprägt (64). Wenn Menschen mit Babies reden, um diese zu beruhigen, so fällt die Kontur meist (64). Um dies in Musik zu untersuchen, wird im Folgenden die Melodiekontur herausgearbeitet und auf Linearität untersucht, sowie die Richtung der Kontur: Nach oben oder nach unten.

Zunächst wurde mittels Melodia (56) die Melodie des Stückes extrahiert, welche als .csv ausgegeben wird: Die erste Spalte listet Zeitsignaturen und die zweite die Tonhöhe in Hz. Dabei werden Hz Werte kleiner gleich 0 als keine Melodie interpretiert.

Melodieerkennung ist ein Thema, welches noch immer erforscht wird, unter anderem, weil die Definition einer Melodie recht vage ist (48). Deshalb ist es nicht überraschend, dass das System nicht jeden Melodieton erkennt bzw teilweise zu viele Töne anzeigt. Laut den Autoren liegt die Accuracy zwischen 76% und 78% (56).

Im Folgenden wurden einige Annahmen, bezüglich der erkannten Melodien getroffen. Die Melodielinie wird in mehrere Segmente zerteilt: Ein neues Melodiesegment beginnt erst nach mindestens einer Achtelnote Pause, ansonsten werden die folgenden Töne dem alten Melodiesegment zugeschrieben. Außerdem werden Melodiesegmente verworfen, welche kürzer als eine Viertelnote dauern.

$$t_{\text{Viertelnote}} = 60/bpm$$

$$t_{\text{Achtelnote}} = 30/bpm$$

Zunächst wird aufgrund der logarithmischen Wahrnehmung des Menschen der Logarithmus zur Basis 2 der Tonhöhen gebildet und mit 12 multipliziert, wodurch ein Halbtonsprung einer Addition von 1 entspricht. Daraufhin wird für jedes Melodiesegment nun eine Lineare Regression gestartet. Im Anschluss wird die resultierende Steigung mit 60/bpm multipliziert, wodurch die Steigung nun die intuitive Einheit Halbton/Schlag besitzt. Aus dieser Linearen Regression, welche mittels der Library `scipy`¹⁷ durchgeführt wird, werden nun drei Features gebildet:

contour_slope_average gibt an, inwieweit die Melodien fallen oder steigen: positive Werte geben an, dass Melodien steigen, negative dass sie fallen. Dies wird mittels einer gewichteten Summe und der Steigungsparameter der Linearen Regressionen errechnet, mit den Zeitdauern der Melodiesegmente als Gewichten:

$$\text{contour_slope_abs} = \frac{1}{\sum_i t_i} * \sum_i^n t_i * slope_i$$

mit n Anzahl Melodiesegmenten, t_i Dauer des i ten Melodiesegments und $slope_i$ Steigung des i ten Melodiesegments.

contour_slope_abs gibt an, wie stark Melodien steigen bzw. fallen: Werte nahe Null geben an, dass die Melodiesegmente kaum steigen oder fallen, während

¹⁷<https://scipy.org/>

hohe Werte große Flanken in Melodiesegmenten indizieren. Dies wird mittels einer Gewichteten Summe und den Beträgen der Steigungsparameter der Linearen Regressionen errechnet:

$$\text{contour_slope_abs} = \frac{1}{\sum_i t_i} * \sum_i^n t_i * |\text{slope}_i|$$

mit n Anzahl Melodiesegmenten, t_i Dauer des i ten Melodiesegments und slope_i Steigung des i ten Melodiesegments.

contour_linearity_score gibt an, ob die Melodiesegmente gut durch eine Lineare Regression abgebildet werden können: Also wie sehr die Kontur einer Geraden entspricht. Dies wird mittels des Betrags von Pearson's r (12) gemessen¹⁸. Daraufhin wird eine Gewichteten Summe über alle Melodiesegmente gebildet. Dabei stehen Werte nahe eins für sehr lineare Melodiesegmente, während kleinere Werte weniger Linearität bedeuten:

$$\text{contour_slope_abs} = \frac{1}{\sum_i t_i} * \sum_i^n t_i * |\text{score}_i|$$

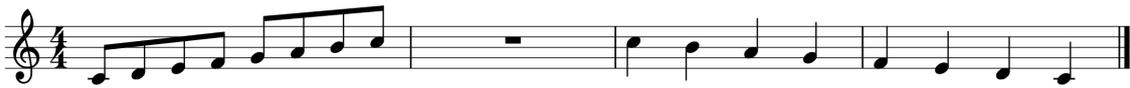
mit n Anzahl Melodiesegmenten, t_i Dauer des i ten Melodiesegments und score_i Pearsons r des i ten Melodiesegments.

Zur Illustration werden im Folgenden zwei Beispiele diskutiert. Diese wurden mit Musescore3¹⁹ erzeugt.

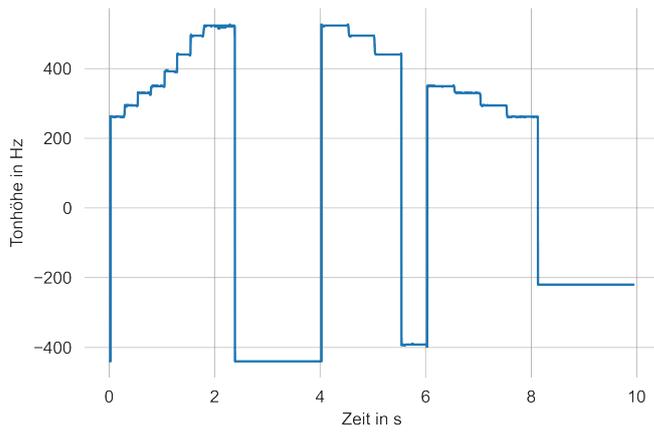
- In Abbildung 8a ist das Notenbild des ersten Beispiels zu sehen, in Abbildung 8b der mittels Melodia (56) erzeugte Melodiezug. Deutlich zu erkennen ist das Fehlen eines Melodietons bei Sekunde 5,5. In Abbildung 8c sind die Längen, die Steigungen und die Scores der erkannten Melodiesegmente

¹⁸Pearson's r misst den linearen Zusammenhang zwischen zwei Merkmalen. Dabei kann Pearson's r Werte im Intervall [-1, 1] annehmen. 1 (bzw. -1) stehen hierbei für einen perfekten positiven (bzw. negativen) Zusammenhang, während 0 bedeutet, dass die Merkmale nicht linear voneinander abhängen.

¹⁹<https://musescore.org/de>



(a) Notenbild



(b) Ergebnis der Melodiererkennung

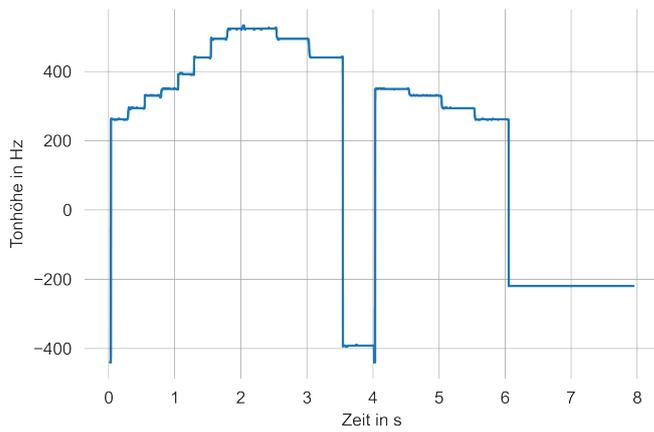
	length	slope	score
0	2.35	3.04	0.98
1	1.51	-1.32	0.93
2	2.09	-1.57	0.96

(c) Tabelle zur Illustration
slope_average = 0.3123
slope_abs = 2.0861
score = 0.9595

Abbildung 8: Beispiele 1 Kontur Features



(a) Notenbild



(b) Ergebnis der Melodierekennung

	length	slope	score
0	3.61	1.47	0.78
1	2.02	-1.60	0.96

(c) Tabelle zur Illustration

slope_average = 0.3472

slope_abs = 1.5150

score = 0.8428

Abbildung 9: Beispiele 2 Kontur Features

zu sehen. Das Fehlen des Melodietons erzeugt drei Abschnitte anstatt zwei. Die Steigungen können leicht überprüft werden: Beim ersten Melodianstieg wird eine Oktave innerhalb von vier Schlägen bewältigt. Da eine Oktave aus zwölf Halbtonschritten besteht, resultiert daraus eine Steigung von $12/4 = 3$, was hier zu sehen ist. Die resultierenden Werte für `contour_slope_average`, `contour_slope_abs` und `contour_linearity_score` sind ebenfalls in Abbildung 8c zu sehen: Aufgrund des nicht erkannten Tons ist `contour_slope_average` leicht positiv und nicht genau null. Anhand von `contour_slope_abs` ist deutlich eine starke Steigung zu sehen und `contour_linearity_score` ist nahe eins, was die vorliegende Linearität bestätigt.

- In Abbildung 9a ist das Notenbild des zweiten Beispiels zu sehen, in Abbildung 9b der mittels Melodia (56) erzeugte Melodiezug. Deutlich zu erkennen ist das Fehlen eines Melodietons bei Sekunde 3,5. In Abbildung 9c sind die Längen, die Steigungen und die Scores der erkannten Melodiesegmente zu sehen. Das Fehlen des Melodietons erzeugt zwei Abschnitte anstatt einem. Die resultierenden Werte für `contour_slope_average`, `contour_slope_abs` und `contour_linearity_score` sind auch in Abbildung 9c zu sehen: Aufgrund des nicht erkannten Tons ist `contour_slope_average` leicht positiv und nicht genau null. Da Auf- und Abstieg innerhalb eines (bzw. zweier) Melodiesegmente geschieht, sind `contour_slope_abs` und `contour_linearity_score` geringer als im vorherigen Beispiel.

Ähnlichkeits-Features basierend auf Self Similarity Matrices Eine wichtige Eigenschaft von Schlafliedern sind Repetitionen: Sowohl rhythmisch als auch melodisch und harmonisch (64). Um dies zu untersuchen, wurden im Folgenden Features entworfen, welche Ähnlichkeiten mittels einer Self Similarity Matrix (SSM) herausarbeiten und quantifizieren:

Eine SSM vergleicht eine Sequenz von Featurevektoren mit sich selbst. Jeder Eintrag gibt an, wie ähnlich sich zwei Featurevektoren sind. Die hierbei verwendete Funktion ist das Skalarprodukt über normalisierten Featurevektoren, was gleichbedeutend mit dem Cosinus des Winkels ist, welcher von den beiden Featurevektoren

aufgespannt wird. Die SSM wird normalerweise zur Structure Analysis verwendet (48). Sie eignet sich aber ebenfalls gut, um Repetitionen herauszuarbeiten, da sie Ähnlichkeiten angibt. Um diese als numerisches Feature zu nutzen, wird über alle Einträge der SSM die Summe gebildet. Dabei wird die mittlere Diagonale ausgenommen, auf welcher Featurevektoren mit sich selbst verglichen werden, also somit die Einträge immer eins sind. Anschließend wird durch die Anzahl an Einträgen, die berücksichtigt wurden, geteilt. Der Featurewert liegt also im Intervall $[0, 1]$ für rein positive Features und ansonsten im Intervall $[-1, 1]$:

$$\text{similarity_featuretype} = \frac{1}{(n^2 - n)} * \sum_{i \neq j} SSM_{\text{featuretype}}(i, j)$$

mit n Anzahl Featurevektoren und $SSM_{\text{featuretype}}(i, j)$ Wert der $SSM_{\text{featuretype}}$ an Position (i, j) ($i, j \in N, 1 \leq i, j \leq n$).

Hier wurden drei verschiedene Featuretypen gewählt, um verschiedene Aspekte der Musik zu betrachten:

- **similarity_chroma:** Chromafeatures geben an, welche der zwölf Töne sich in welchem Maße in dem Stück analysierter Musik befinden. Deshalb können aus einer SSM mit Chromafeatures als Basis Melodie und Harmonikähnlichkeiten abgelesen werden. Chromafeatures sind immer positiv.
- **similarity_timbre:** MFCC Features geben Auskunft über Instrumentation und Klangfarbe (48; 49). Eine SSM mit MFCC Features gibt also Aufschluss über die Veränderung der Instrumentation und Klangfarbe. MFCC Features können positiv und negativ sein. Es wurden die Koeffizienten 4-14 verwendet, da andere Features sonst stark dominieren und mit dieser Auswahl Strukturen deutlicher sichtbar werden (48; 49).
- **similarity_tempo:** Tempogram-Features schätzen das Tempo von Musikabschnitten ab, folglich gibt eine SSM also an, wie stark sich das Tempo im Stück verändert. Tempogram-Features sind immer positiv.

Genauere Informationen zu den Features und SSMs können (48) entnommen werden. Die Implementierung von (49) wurde für die Featureextraktion und die Erzeugung der SSM genutzt.

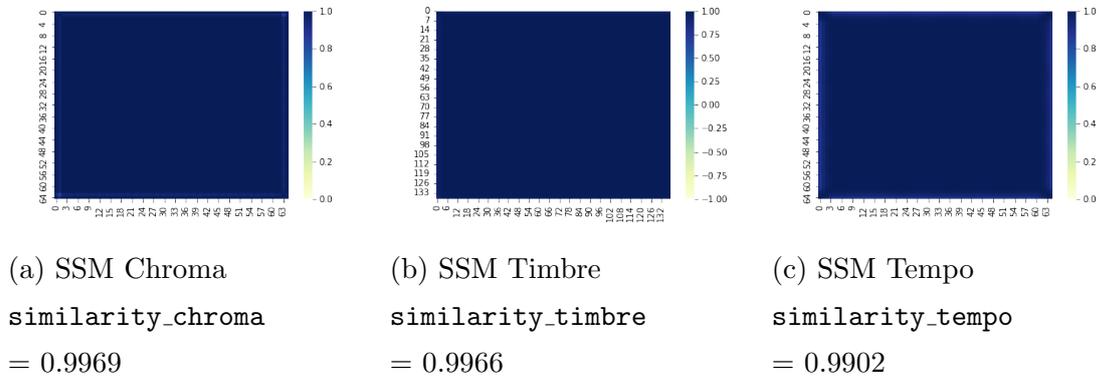


Abbildung 10: Beispiele 1 Similarity Features

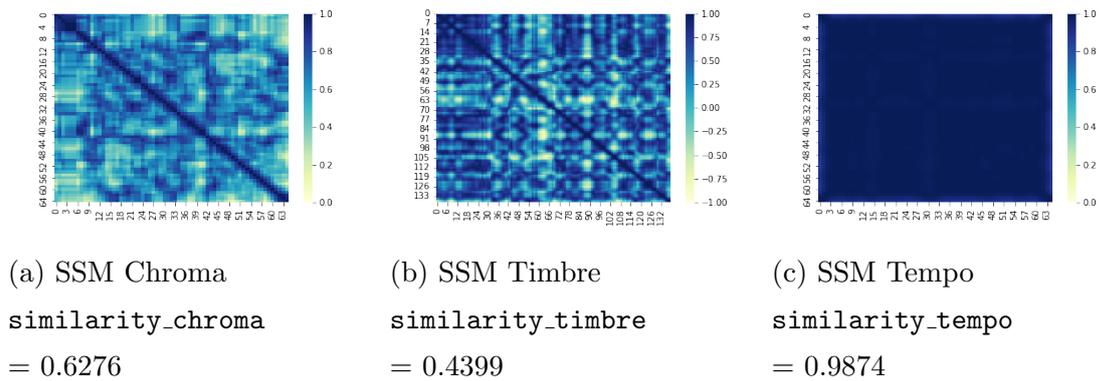
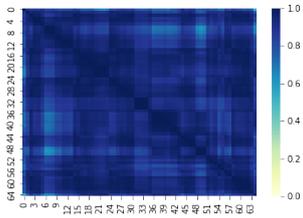


Abbildung 11: Beispiele 2 Similarity Features

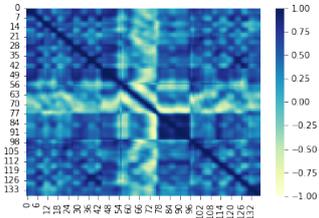
Für diese Features sind im Folgenden vier Beispiele mit Musescore3 ²⁰ erzeugt worden, um ihre Bildung und Aussage zu illustrieren. Diese dauern jeweils 32 Takte:

- Im ersten Beispiel wird mit Vierteln zwischen c und d gewechselt. Das Tempo (120bpm) und die Instrumentation (solo Klavier) bleiben gleich. In Abbil-

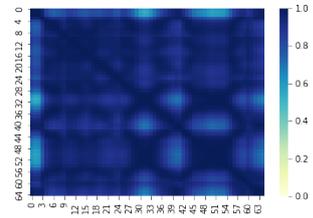
²⁰<https://musescore.org/de>



(a) SSM Chroma
`similarity_chroma`
 $= 0.9144$

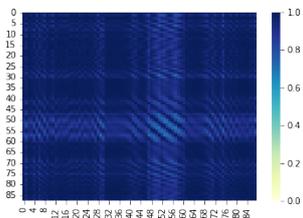


(b) SSM Timbre
`similarity_timbre`
 $= 0.3511$

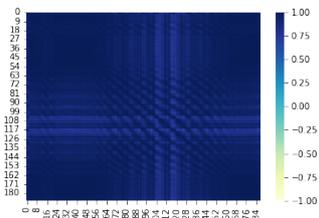


(c) SSM Tempo
`similarity_tempo`
 $= 0.9215$

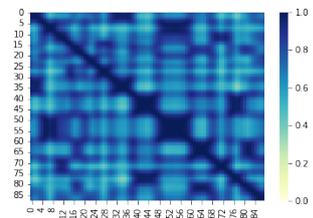
Abbildung 12: Beispiele 3 Similarity Features



(a) SSM Chroma
`similarity_chroma`
 $= 0.9534$



(b) SSM Timbre
`similarity_timbre`
 $= 0.9338$



(c) SSM Tempo
`similarity_tempo`
 $= 0.7327$

Abbildung 13: Beispiele 4 Similarity Features

dung 10 sind die SSM für die verschiedenen Features zu sehen und die einzelnen Features, welche alle sehr nahe an 1 liegen.

- Im zweiten Beispiel werden Viertel mit zufälligen Tonwerten gespielt. Das Tempo (120bpm) und die Instrumentation (solo Klavier) bleiben dabei gleich. In Abbildung 11 sind die SSM für die verschiedenen Features zu sehen und die einzelnen Features. Deutlich erkennbar ist die geringere Ähnlichkeit im Chroma SSM und `similarity_chroma`. Interessanterweise wird offensichtlich die Klangfarbe ebenfalls stark verändert. Zu beachten ist hierbei, dass MFCC-Features negativ sein können und damit auch `similarity_timbre`, weshalb `similarity_chroma` als geringer zu interpretieren ist als `similarity_timbre`.
- Im dritten Beispiel wird eine C-Dur-Tonleiter in Vierteln von 16 verschiedenen Instrumenten gespielt. Dabei bleibt das Tempo (120bpm) gleich. In Abbildung 12 sind die SSM für die verschiedenen Features zu sehen und die einzelnen Features. Deutlich erkennbar der geringe Wert im Timbre sowie die hohen Werte in Chroma und Tempo.
- Im letzten Beispiel werden die ersten vier Töne der C-Dur-Tonleiter aufsteigend in Vierteln gespielt. Dabei wechselt das Tempo alle zwei Takte: Insgesamt werden 16 verschiedene Tempi zwischen 40 bpm und 220 bpm benutzt. In Abbildung 13 sind die SSM für die verschiedenen Features zu sehen und die einzelnen Features. Die Tempowechsel sind im Tempo SSM deutlich zu sehen. Ebenso ist ersichtlich, dass die anderen beiden Features sehr hohe Werte aufweisen.

3 Analyse

In der folgenden Analyse werden die vorher erzeugten Daten betrachtet. Dazu werden zwei Forschungsfragen behandelt:

1. Was unterscheidet Schlafmusik von anderer Musik?
2. Welche Gemeinsamkeiten haben Schlafmusiklieder?

Dabei wird Musik, welche zum Schlafen verwendet wird, untersucht. Um diese Ergebnisse möglichst gut interpretieren zu können, werden auch Schlaflieder, sowie Musik, welche in Studien als gut zum Einschlafen erachtet wurde, bewertet. Um einen Vergleich zu (69) anstellen zu können, wird zusätzlich Schlafmusik allgemein (also eine Zusammenführung aller vorherig genannten Gruppen) untersucht. Unerwartete Ergebnisse werden überprüft und insbesondere die eigenen Features werden in der Praxis evaluiert. Alle Tabellen und genauen Daten finden sich in Anhang C „Tabellen für Analyse“.

Zunächst wird die erste Frage behandelt. Dezu werden diskriminierende Features herausgearbeiten. Im Anschluss werden Features evaluiert und danach wird auf die zweite Frage eingegangen, bei welcher Features mit geringer Standardabweichung bestimmt werden.

3.1 Diskriminierende Features

Es werden drei verschiedene Methoden verwendet, um diskriminierende Features herauszufinden:

Mittels **Cohen's d** (22) wird die Effektstärke bestimmt. Dazu wird die Differenz der Mittelwerte $mean_i$ mit den Standardabweichungen std_i und Samplegrößen n_i normalisiert.

$$cohen's_d = \frac{mean_1 - mean_2}{s}$$
$$s = \sqrt{\frac{(n_1 - 1) * std_1^2 + (n_2 - 1) * std_2^2}{n_1 + n_2 - 2}}$$

Diese Metrik wurde in (57) am meisten verwendet und dient neben der guten Interpretierbarkeit unter anderem dazu, einen Vergleich der Ergebnisse möglich zu machen.

Als zweite Metrik wird wie in (69) mittels des k-means Algorithmus der **adjusted Rand Index (ARI)** (35) berechnet. Dieser gibt an, wie gut ein Clustering ist. Dieser Index liegt im Intervall $[-1, 1]$, wobei 1 ein perfektes Clustering darstellt und 0 eine zufällige Zuordnung. Der k-means Algorithmus arbeitet auf normalisierten Daten (gleiche Normalisierung wie schon zuvor in dieser Arbeit) und startet auf den featureweisen Mittelpunkten der beiden zu vergleichenden Gruppen, um den Zufallsfaktor der Startpunkte zu eliminieren und die Ergebnisse vergleichbar zu machen. Hierbei werden nur Gruppen von Features bewertet und diese Metrik dient zur Vergleichbarkeit mit (69), sowie als Indikator für die Relevanz dieser Feature-Gruppen und für die räumliche Separierbarkeit der Daten.

Als dritte Metrik wird der **Weight Vector eines Perceptron** verwendet. Da das Perceptron auf normalisierten Daten deutlich besser arbeitet, wurden die Daten wie bereits in Kapitel 2.1 „Datensatz“ normalisiert. Dadurch kann der Weight-Vector direkt interpretiert werden, was einen weiteren Vorteil von normalisierte Daten darstellt. Um gute Werte zu erreichen, wurden jeweils 1000 Epochen gerechnet. Es wird davon ausgegangen, dass das Verfahren bis dahin konvergiert ist, falls eine Konvergenz möglich ist. Diese Methode dient einerseits dazu zu sehen, wie gut die Daten mittels einer Ebene trennbar sind, andererseits können die Weight-Vector Einträge als Wichtigkeit des dazugehörigen Features interpretiert werden. Jedoch sind diese weniger aussagekräftig als die Cohen’s d Werte, da diese rein auf den Featurewerten berechnet werden, während das Perceptron den ganzen Featurevektor verwendet.

Im Folgenden werden verschiedene Song-Gruppierungen miteinander verglichen. Die Gruppierungen sind in Tabelle 2 erklärt. Zunächst finden wie in (57; 69) allgemeine Vergleiche zwischen Schlafmusik und Referenzmusik statt, danach genreinterne Vergleiche. Dabei haben genreinterne Vergleiche den großen Vorteil, dass innerhalb eines Genres Features musikalisch sinnvoller miteinander verglichen werden können.

Western Lullabies (WeL)	Alle westlichen Lullabies zusammengefasst
Lullabies (L)	Western Lullabies und World Lullabies
Dataset (D)	Alle verwendeten Songs aus dem Datensatz aus (57)
Sleep Music (S)	Dataset mit Lullabies und TSM Dabei wurden, um einen Bias durch die vielen Versionen in Western Lullabies des jeweils selben Songs zu verhindern, anstatt den verschiedenen Versionen jeweils ein Datenpunkt pro Lied verwendet, mit den arithmetischen Mitteln der Features aller Songversionen
Reference (R)	Alle Referenzmusik

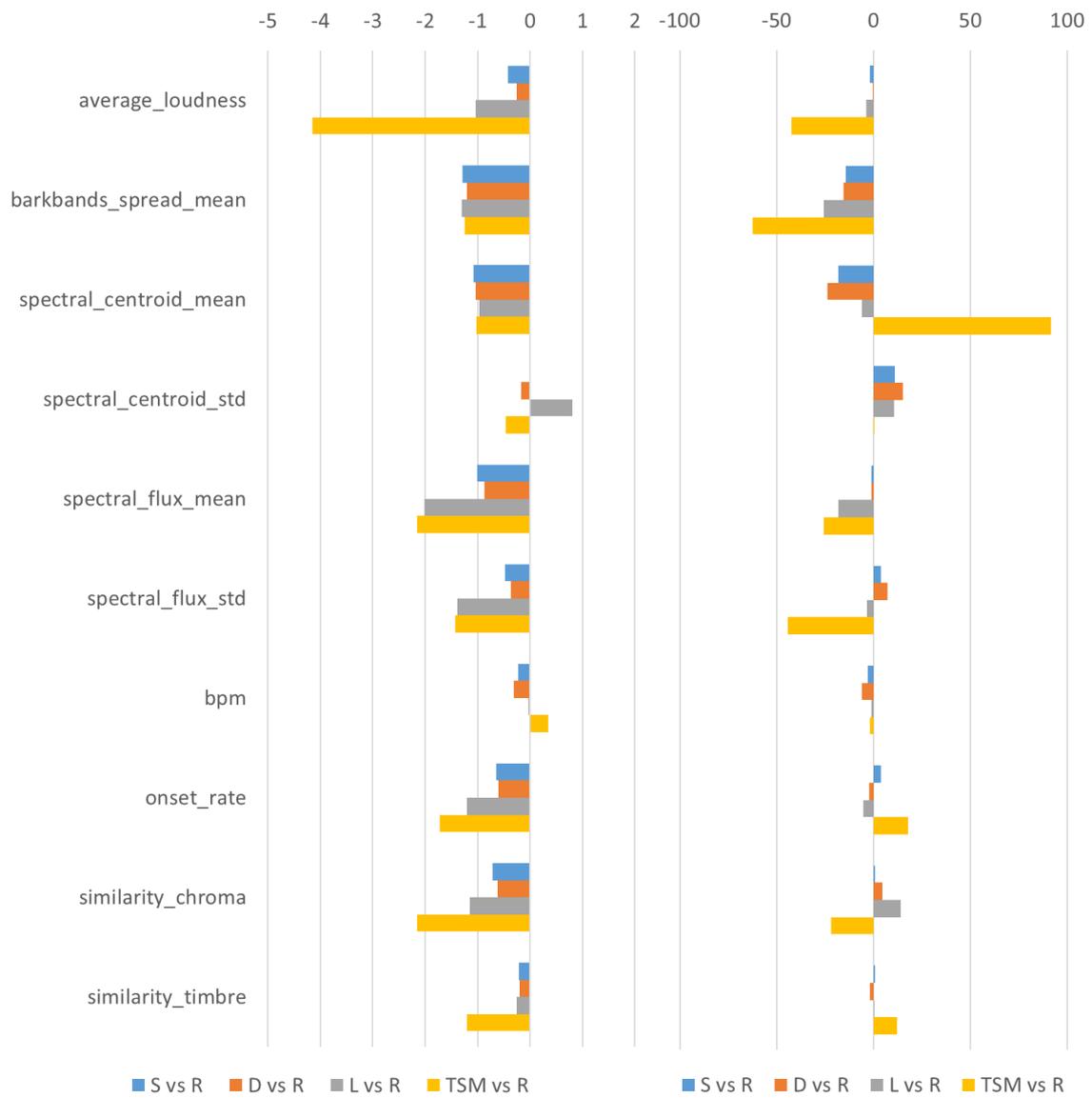
Tabelle 2: Erklärung der Gruppierungen

3.1.1 Allgemeine Vergleiche

Die allgemeinen Vergleiche haben das Ziel, diskriminierende Features herauszuarbeiten, indem große Gruppen von Schlafmusikliedern mit Referenzmusik verglichen werden. Dabei ist insbesondere der Datensatz sehr vielseitig, weshalb keine perfekte Trennung möglich ist. Jedoch können Tendenzen herausgearbeitet werden. Eine Auswahl von Features ist in Abbildung 14 und Abbildung 15 illustriert.

Sleep Music und Reference Dieser Vergleich vergleicht alle verfügbare Schlafmusik mit aller verfügbaren Referenzmusik und ist damit analog zu dem in (69) mit Ausnahme der Schlaflieder. Deshalb werden die Ergebnisse verglichen.

Die wichtigsten Features sind hierbei nach Cohen’s d `barksband_spread_mean`, `spectral_centroid_mean` und `spectral_flux_mean`. Diese Tendenzen sind im Perceptron ebenfalls sichtbar, wobei das hier dominante `spectral_rolloff_mean` die entgegengesetzte Richtung hat, wie bei der Analyse mittels Cohen’s d. Deshalb wird es nicht berücksichtigt. Somit zeichnet sich Schlafmusik gegenüber Referenzmusik durch einen weniger vollen Klang, dunklerer Klangfarbe und weniger rauhen Klang



(a) Cohens d

(b) Perceptron

Abbildung 14: Allgemeine Vergleiche

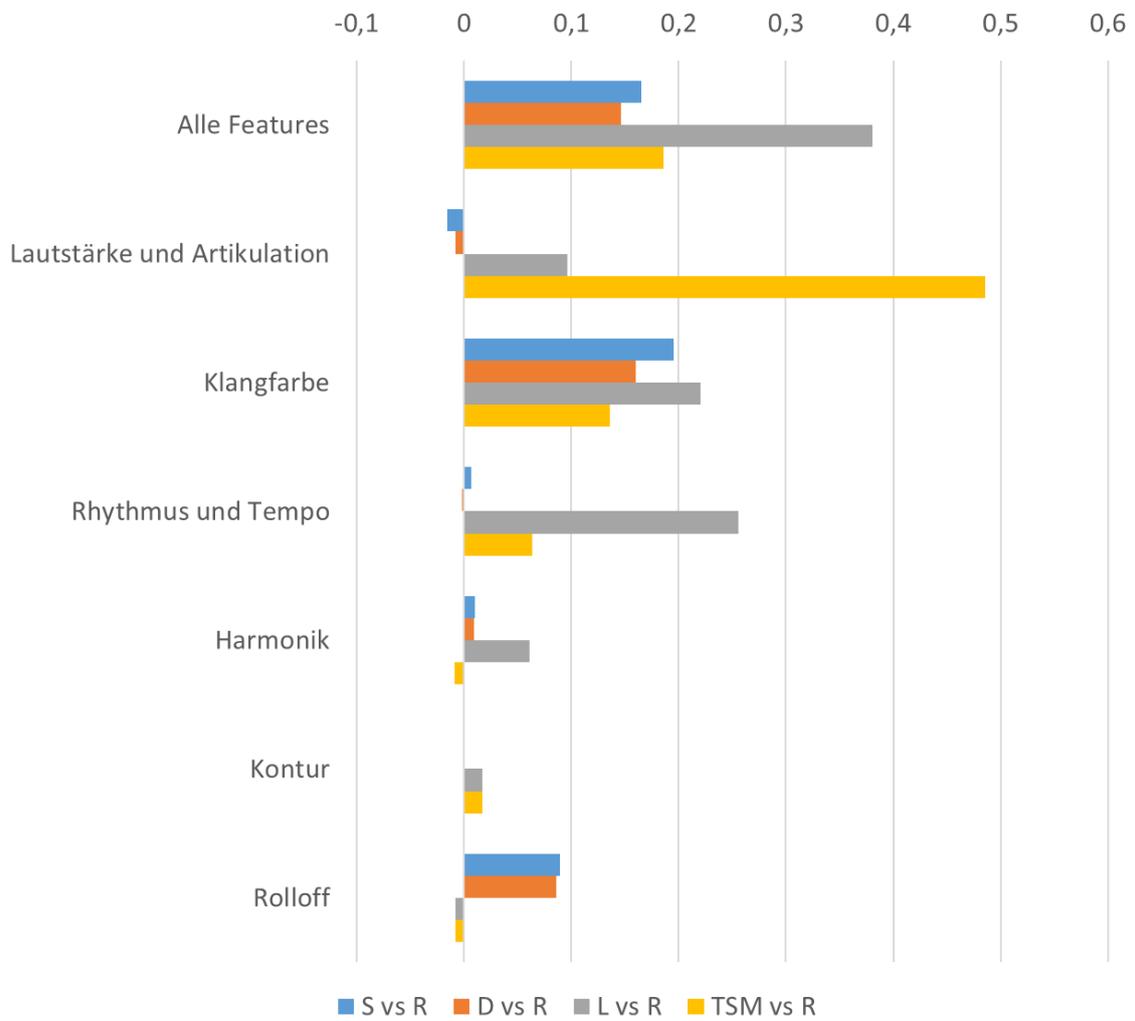


Abbildung 15: Allgemeine Vergleiche mittels ARI

aus. Allerdings ist der Vergleich weniger deutlich als später folgende: Die Fehlerquote des Perceptrons liegt bei 24.38%, wodurch zwar deutlich wird, dass die Datenpunkte nicht vollständig aufeinanderliegen, eine gute Trennung dennoch nicht möglich ist.

Dies wird auch in der Analyse mittels ARI deutlich: Alle Werte sind recht klein, die beste Trennung kann mittels der Klangfarbenfeatures ermittelt werden, was zu der vorigen Analyse passt. Die Werte sind ähnlich zu denen, welche in (69) ermittelt wurden, mit Ausnahme der Featuregruppe, welche dort am besten diskriminieren konnte: „*Rolloff*“ konnte anders als in (69) in diesem Vergleich keine gute Trennung erzeugen.

Dataset und Reference In diesem Vergleich wird der Datensatz mit der Referenzmusik verglichen. Somit werden hier die realen Hörgewohnheiten von Menschen verglichen, welche mittels Spotify ermittelt wurden. Dieser Vergleich ist analog zu dem aus (57), weshalb die Ergebnisse miteinander verglichen werden.

Dieser Vergleich ist etwas weniger deutlich als der vorherige, was anhand der betragslich kleineren Cohen's d Werten, der höheren Fehlerwahrscheinlichkeit des Perceptrons und der kleineren ARI gesehen werden kann. Davon abgesehen liefert der Vergleich dieselben Ergebnisse wie der vorherige.

Die Cohen's d Werte sind hierbei in einem ähnlichen Bereich wie die in (57) ermittelten Werte. Einige Features scheinen jedoch deutlich weniger aussagekräftig als die Features in (57) zu sein: Der geringste Wert lag dabei bei -0.34, während in diesem Vergleich der Wert von 0.00 erreicht wurde. Ein featurebasierter Vergleich ist aufgrund der verschiedenen Features und der Intransparenz der Spotifyfeatures nicht gut möglich. Eine Ausnahme bildet dabei das Tempo: Das Tempo hat in beiden Vergleichen einen überraschend geringen Einfluss. Deswegen werden in Kapitel 3.2 „bpm“ die Ergebnisse beider Algorithmen, welche die bpm bestimmen, eingehender beleuchtet.

Lullabies und Reference Bei diesem Vergleich ist erneut anzumerken, dass die meisten der Schlaflieder für Erwachsene aufgenommen und produziert wurden, was erheblichen Einfluss auf den verwendeten Frequenzbereich hat.

Insgesamt ist dieser Vergleich deutlicher als die vorherigen: So ist die Fehlerwahrscheinlichkeit des Perceptrons mit 12.97% merklich geringer und die ARI Werte sind deutlich größer. Die dominierenden Features sind hierbei `spectral_flux_mean` und `std` sowie `spectral_spread_mean` und die `onset_rate`. Das Perceptron liefert recht geringe Werte, wobei die Orientierungen der gerade genannten Features übereinstimmt. Somit trennt Schlaflieder von Referenzmusik ein weniger rauher Klang, sowie eine dahingehende geringe Änderung innerhalb des Stückes. Außerdem sind Schlaflieder weniger rauschend und es gibt weniger Notenansätze. Diese Wichtigkeit der Klangfarbe sowie des Rhythmus zeichnet sich auch in der Analyse mittels ARI wieder: So können Features aus der „*Rhythmus und Tempo*“-Gruppierung einen deutlich besseren Wert erzielen als in den vorherigen Vergleichen. Dennoch hat eine Analyse mit allen Features den besten Wert.

Therapeutical Sleep Music und Reference In diesem Vergleich werden Musikstücke, welche in klinischen Studien als therapeutisch bestimmt wurden, welche also einen tatsächlichen positiven Effekt auf den Schlaf haben, mit den Referenzsongs verglichen. Dabei fällt der Vergleich sehr deutlich aus. Besonders interessant ist, dass fast sämtliche Features Ausprägung in dieselbe Richtung haben, wie schon in den vorherigen Vergleichen, jedoch meist deutlicher ausgeprägt.

Mit einem Extremwert von -4.15 wird der Vergleich von `average_loudness` dominiert. Dabei ist vor allem die Orientierung dieses Features überraschend. Deshalb wird das Feature in Kapitel 3.2 „`average_loudness`“ genauer betrachtet.

Weitere in diesem Vergleich gut diskriminierende Features sind `barksband_spread_mean`, `spectral_flux_mean`, `spectral_spread_mean`, die `onset_rate` und `similarity_chroma`, sowie `similarity_timbre`. Die meisten der Orientierungen konnten durch das Perceptron bestätigt werden. Dass `similarity_chroma` und `similarity_timbre` bei TSM einen geringeren Wert annimmt als bei Referenz Musik, ist überraschend. Deshalb werden diese Features in Kapitel 3.2 „Similarity Features“ genauer betrachtet.

Insgesamt ist durch das Perceptron eine perfekte Trennung möglich, was die Deutlichkeit dieses Vergleiches unterstreicht. Die ARI-Werte sind dagegen recht

niedrig, mit Ausnahme von „*Lautstärke und Artikulation*“, welches aufgrund von `average_loudness` einen sehr hohen Wert hat.

Somit ist es möglich, Therapeutical Sleep Music gänzlich von Reference zu trennen. Dabei sind viele Klangfarben wichtig: Ein weniger voller, rauher und rauschender Klang und eine dunkle Klangfarbe scheinen wichtig zu sein.

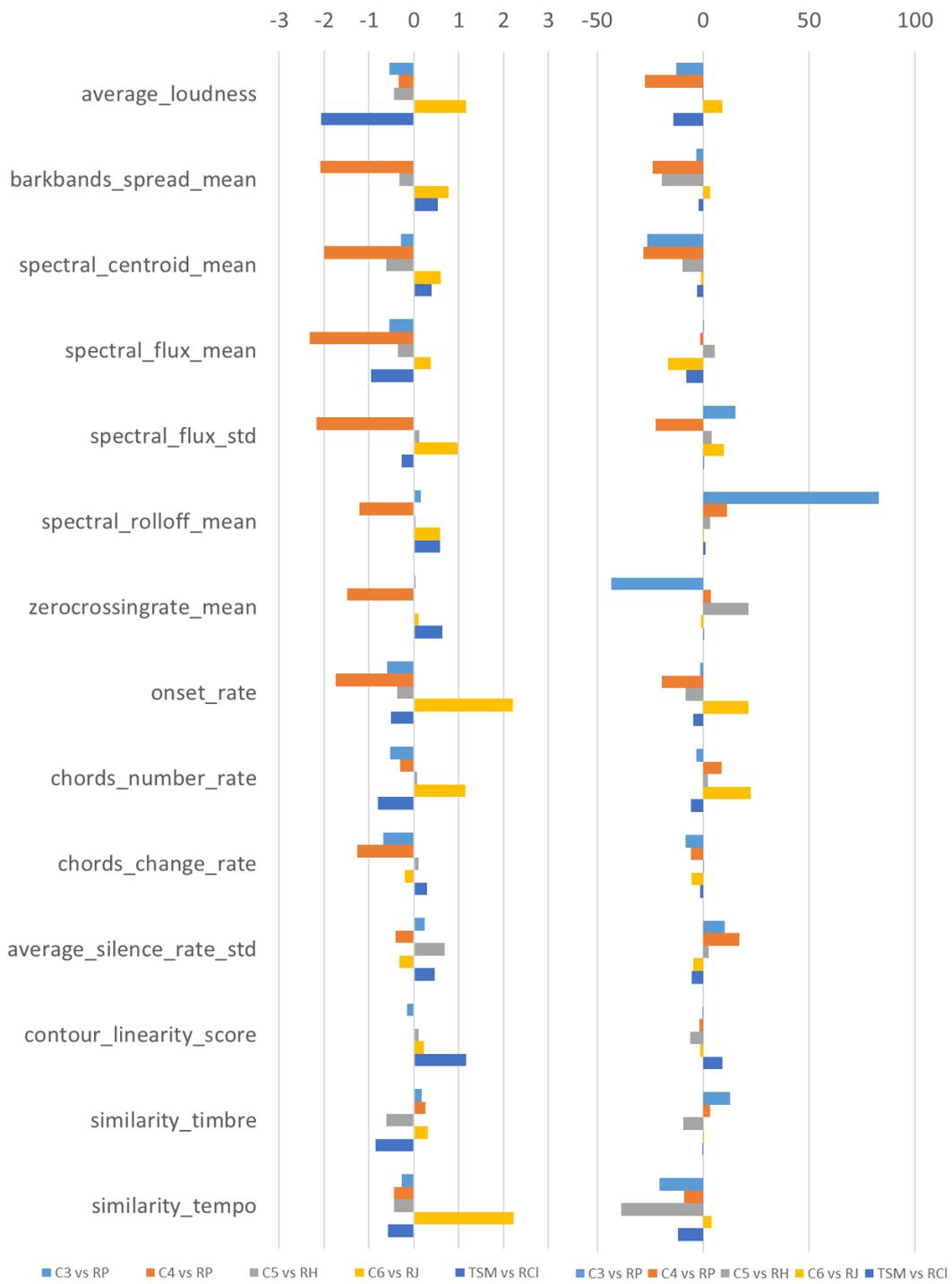
3.1.2 Genreinterne Vergleiche

Da einige Features auf unterschiedlichen Genres unterschiedlich gut funktionieren (siehe Kapitel 3.2 „Featureevaluation“) und überdies eine unterschiedliche Wichtigkeit genießen, sind vor allem musikalisch genreinterne Vergleiche wichtig. Dafür wurden die Untergruppen des Datensatzes auf Genres untersucht und für die folgenden fünf Vergleiche wurde alle Songs eines speziellen Genres manuell ausgewählt. Dabei ist die getroffene Auswahl subjektiv, da viele Genres ineinander übergehen und es keine klaren Trennungen gibt: Beispielsweise wird Instrumental Hip Hop nicht als Hip Hop gewertet, da der Autor Rap für einen essentiellen Teil des Hip Hop hält.

Die Vergleiche werden mit ausgewählten Features in Abbildung 16 und Abbildung 17 illustriert. Genaue Daten aller Features können den Tabellen 7, 8 und 9 entnommen werden.

Radio Tracks und Pop In diesem Vergleich werden alle Popsongs aus Cluster 3: Radiotracks, welches das gleichnamige Cluster in (57) repräsentieren soll, mit den Pop-Referenzsongs verglichen.

Der Vergleich zeigt auf, dass keine deutlichen Unterschiede bestehen: Die Fehlerquote des Perceptron liegt bei 17.78% und die ARI Werte sind sehr gering. Nur „*Rhythmus und Tempo*“ erlangt einen Wert über 0.1. Ebenso sind die Cohen’s d Werte recht gering: Kein Wert hat einen Betrag über 1.0. Die stärksten Diskriminatoren sind hierbei `chords_change_rate` und `onset_rate`. Die Orientierung wird von dem Perceptron bestätigt und dieses wertet `spectral_roll_off` sehr hoch. Somit liegt der Unterschied zwischen Cluster 3 und Pop Referenz in einem langsameren Akkordwechsel und weniger Tonansätzen. Anders als in den vorigen Vergleichen



(a) Cohens d

(b) Perceptron

Abbildung 16: Genreinterne Vergleiche

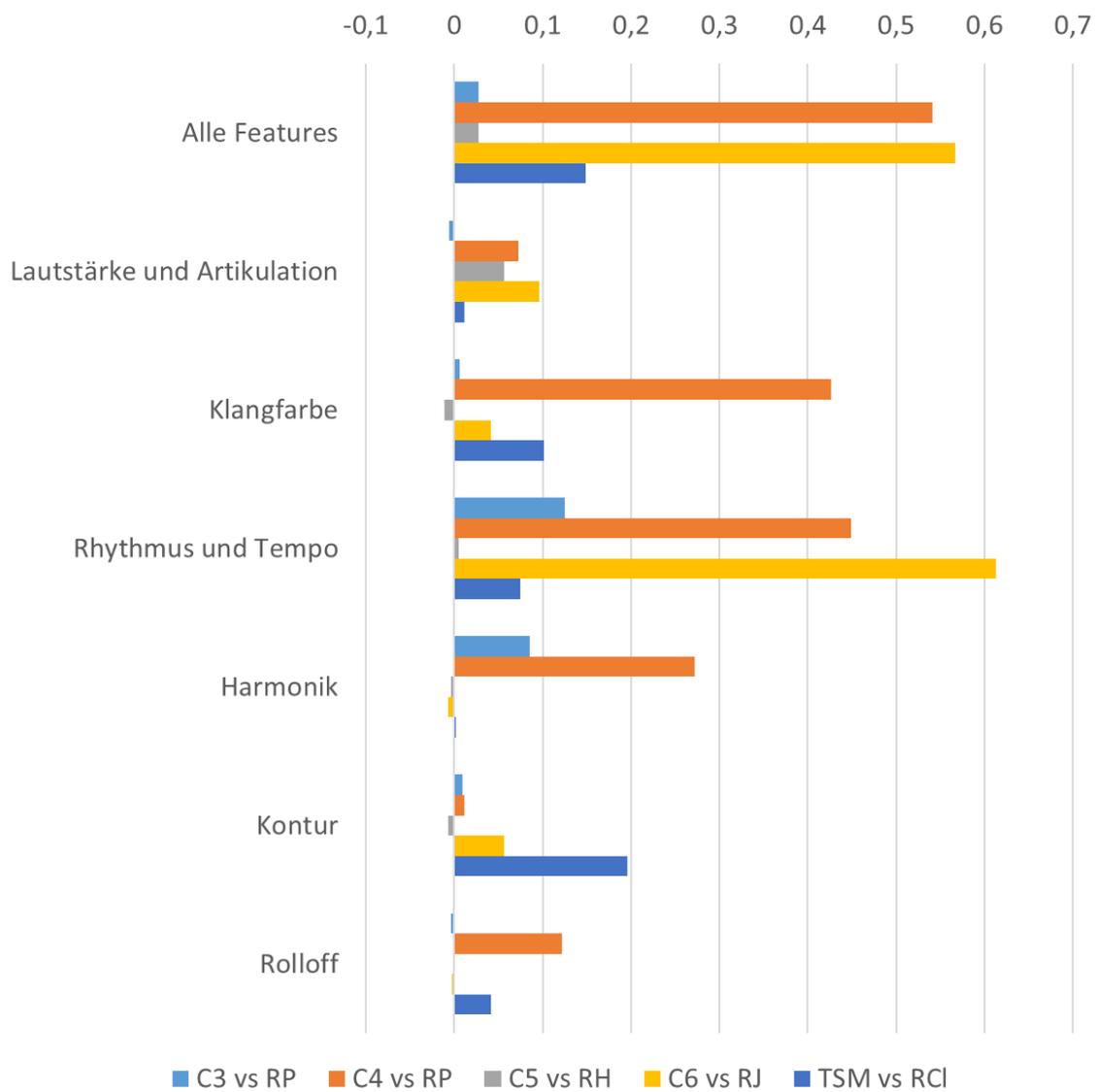


Abbildung 17: Genreinterne Vergleiche mittels ARI

spielt die Klangfarbe eine deutlich untergeordnete Rolle, wahrscheinlich aufgrund des genreinternen Vergleichs.

Acoustic Radio Tracks und Pop Dieser Vergleich behandelt Popsongs aus Cluster 4: Acoustic Radio Tracks und die Pop Referenzsongs. Dabei sind die Songs in Cluster 4, wie der Name schon aussagt, akustischer angehaucht als die in Cluster 3. Die meisten können am besten als poppige Singer/Songwriter-Songs beschrieben werden.

Dementsprechend fällt der Vergleich deutlicher aus als der vorherige. Dabei liefert Cohen's d viele große Werte: Besonders `sprectral_flux_mean` und `std`, sowie `barksbands_spread_mean` und `std` sowie `sprectral_centroid_mean` fallen auf. Das Perceptron schafft eine perfekte Trennung mit recht kleinen Weight Vektor Werten, bei welchen `sprectral_centroid_mean` betont wird. Dementsprechend sind bei diesem Vergleich ein wenig voller und rauher Klang wichtig sowie ein geringe Änderung bei diesen Attributen. Ebenfalls spielt eine dunklere Klangfarbe eine große Rolle.

Die Analyse mittels ARI ergibt eine sehr gute Trennung mittels aller Features, Klangfarbenfeatures und „*Rhythmus und Tempo*“-Features. Außerdem schneiden die „*Harmonik*“ Features deutlich besser ab als in jedem anderen Vergleich, was mittels der hohen Cohens d Werten bei `chords_change_rate` sowie `similarity_chroma` weiter illustriert wird.

Speechy Tracks und Hip Hop Der Vergleich innerhalb des Genres Hip-Hop fällt wenig deutlich aus. So sind die Cohen's d Werte sehr gering, genauso wie die ARI Werte. Das Perceptron liefert mit 26.27% eine hohe Fehlerquote, welche die schwierige Trennung unterstreicht.

Die ausschlaggebensten Features sind `average_silence_rate_std`, `sprectral_centroid_mean` und `similarity_timbre`. Das Perceptron wertet außerdem `similarity_tempo` und `zerocrossingrate_mean` sehr hoch. Somit sind eine stärkere Artikulationsvariation, eine düsterere Klangfarbe, mehr Klangfarbenvariation, mehr Tempoveränderungen und ein rauschenderer Klang wichtig.

Instrumental Tracks und Jazz Der Vergleich zwischen Cluster 6: Instrumental Tracks und der Jazz Referenz liefert deutliche Unterschiede. Dabei fiel bei der manuellen Genreüberprüfung auf, dass sich die meisten Songs aus Cluster 6 sehr ähnlich anhören.

Auffällig ist unter anderem, dass `average_loudness` erstmals einen deutlichen positiven Wert bekommt. Da `average_loudness` im genreinternen Vergleich potenziell Aussagekraft besitzt (siehe Kapitel 3.2 „`average_loudness`“) lässt sich sagen, dass bei den Songs aus dem Datensatz eine kleinere Dynamikänderung vorhanden ist.

Die am besten diskriminierenden Features sind `similarity_tempo`, `onset_rate`, `spectral_spread_mean`, `chords_number_rate` und `chords_number_rate`. Das Perceptron erzielt eine perfekte Trennung mit recht geringen Gewichtungswerten. Dabei werden abermals `chords_number_rate` und `onset_rate` betont. Damit ist ein konstantes Tempo und ein weniger rauschender Klang wichtig.

Interessanterweise liegt die `onset_rate` bei der Schlafmusik höher als bei der Referenz, was für mehr Tonbeginne spricht. Ebenfalls liegen mehr Akkordwechsel vor. Ein Erklärungsansatz hierfür ist das spezielle Jazzsubgenre, welches in Cluster 6 vorliegt. Dieses ist zwar recht monoton jedoch nicht zwingend langsam mit wenig Tonanschlägen und wenigen Akkordwechseln.

Gleicherweise konnten mittels ARI gute Ergebnisse erzielt werden. Mit allen Features und vor allem mit „*Rhythmus und Tempo*“ konnten gute ARI Werte erreicht werden, welche erstmals annähernd an den Wert von „*Rolloff*“ aus (69) heranreichen.

Therapeutical Sleep Music und Classical Der letzte Vergleich arbeitet im klassischen Genre. Dabei werden die Songs aus TSM mit den klassischen Referenzsongs verglichen.

Dabei sind die Cohen's d Werte zwar größten Teils recht gering und auch mittels ARI ist keine gute Trennung möglich, jedoch erzielt das Perceptron eine perfekte Trennung.

Die diskriminierensten Features sind hierbei `average_loudness`, `contour_linearity_score` und `spectral_flux_mean`. Von dem Perceptron wird zusätzlich `similarity_tempo` hervorgehoben. Somit ist für klassische Songs aus TSM im Vergleich zu klassischen Referenzsongs eine höhere dynamische Reichweite, linearere Konturen und mehr Tempoänderungen charakteristisch.

3.2 Featureevaluation

Im Folgenden werden besonders auffällige Features sowie alle selbsterstellten Features in Bezug auf ihre Aussagekraft und Richtigkeit der Ergebnisse genauer untersucht.

average_loudness überrascht in den vorherigen Vergleichen immer wieder mit der Aussage, dass Schlafmusik eine größere dynamische Reichweite als die Referenzsongs besitze. Dies liegt vermutlich am Mastering ²¹, welches für unterschiedliche Genres unterschiedlich ausfällt, um verschiedene Effekte zu erzielen: So ist es im Genre Pop und anderen radiobasierten Genres wichtig, dass mit jedem Abspielgerät möglichst viele Details zu hören sind, weshalb Pop häufig sehr laut gemastert wird. Dadurch wird als Nebeneffekt häufig die dynamische Reichweite deutlich eingeschränkt (39).

Anders hingegen werden etwa Jazz Songs gemastert, da Jazz Songs meist für sehr hochwertige Anlagen gemastert werden, welche auch leise Details deutlich wahrnehmbar übertragen. Somit wird Jazz leiser gemastert mit einer deutlich größeren dynamischen Reichweite (39).

Zusammenfassend lässt sich sagen, dass das Master, welches extrem genreabhängig ist, auf `average_loudness` wahrscheinlich einen deutlich größeren Effekt hat als die zugrunde liegende Musik. Deshalb ist die musikalische Aussagekraft nur sehr marginal gegeben, obwohl `average_loudness` ein gut diskriminierendes Feature ist.

²¹Mastering ist die Endbearbeitung von Songs, bevor diese auf Tonträgern oder im Streaming veröffentlicht werden.

bpm Das Tempo sollte eigentlich ein deutlicher Indikator für Schlafmusik sein. Dies wird jedoch in den Vergleichen nicht deutlich. In (57) ist der Unterschied ebenso deutlich geringer als erwartet. Ein Blick auf dem Boxplots des Features in Abbildung 18 lässt erkennen, dass der verwendete Algorithmus oft falsche Ergebnisse liefert.

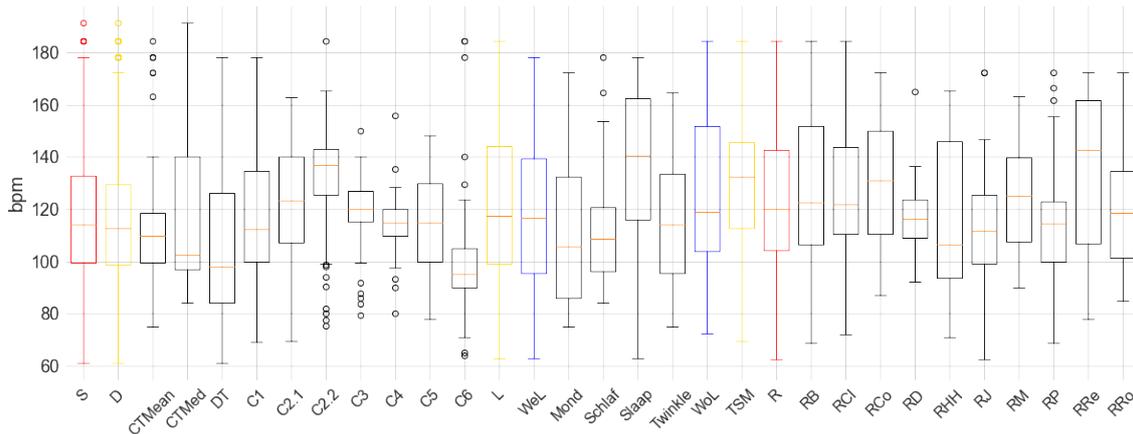
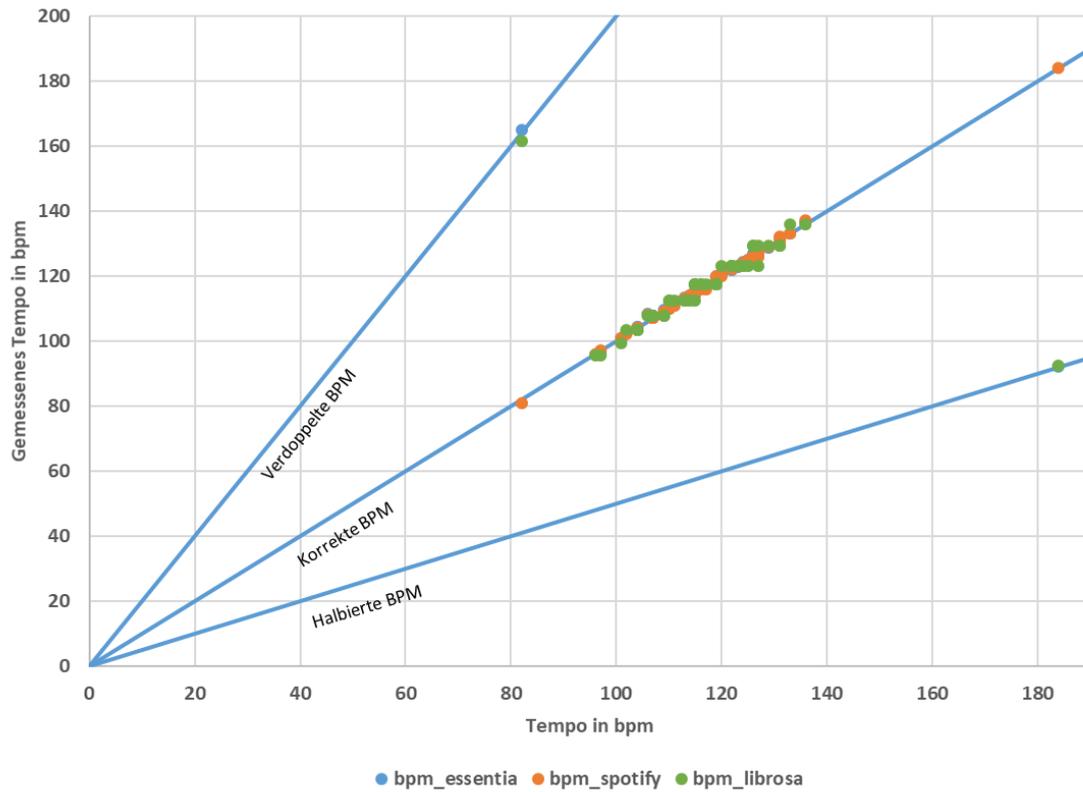


Abbildung 18: Boxplots des Features bpm

So sind in vielen Gruppierungen riesige Streuungen zu bemerken, insbesondere sehr unrealistische hohe bpm Werte etwa bei Schlafliedern. Eine der wenigen Gruppierungen mit geringer Streuung ist Reference Disco. Es liegt also die Vermutung nahe, dass die Korrektheit der Ergebnisse des verwendeten Algorithmus genreabhängig ist. Um dies zu überprüfen wurde die bpm der Songs aus TSM und Reference Disco manuell bestimmt ²² und dann mit verschiedenen Algorithmen verglichen: einmal mit dem in dieser Arbeit verwendeten Algorithmus von Essentia, dem in (57) verwendeten Spotify Algorithmus und die in (69) verwendeten Librosa Variante. Die Ergebnisse sind in Abbildung 19 und Abbildung 20 visualisiert.

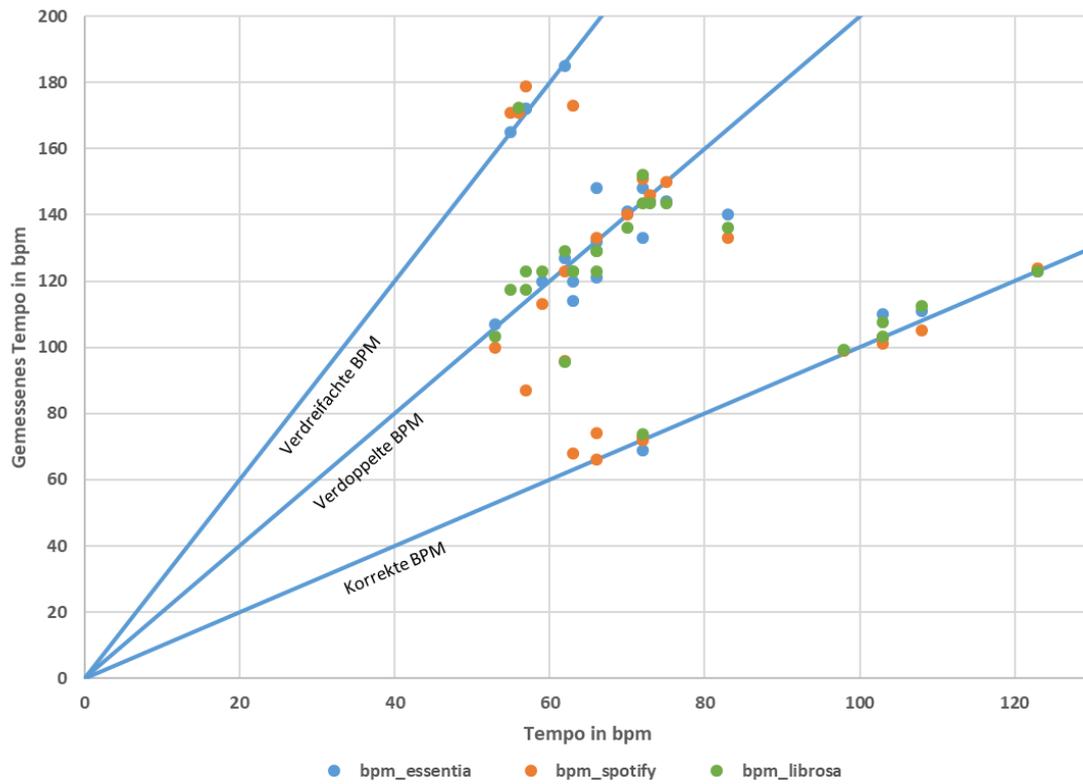
Die Ergebnisse bestätigen die Hypothese: Bei Referenz Disco wird die BPM aller

²²Dabei ist insbesondere bei TSM die manuelle Tempobestimmung schwierig, da eine hohe Varianz im Tempo im Stück vorliegt



	bpm	bpm_essentia	bpm_spotify	bpm_librosa
Median	116	116.35	116	117.45
Mittelwert	116.84	116.87	116.9	116.74

Abbildung 19: BPM Analyse Referenz Disco



	bpm	bpm_essentia	bpm_spotify	bpm_librosa
Median	66	132	113	123.05
Mittelwert	73.43	132.8	118.04	123.33

Abbildung 20: bpm Analyse Therapeutical Sleep Music

Songs mit Ausnahme von zwei von allen Algorithmen richtig bestimmt. Dagegen sind bei TSM viele Werte deutlich unterschiedlich, was gleichermaßen an den Medianen und Mittelwerten deutlich wird: Teilweise ist der Wert sogar verdoppelt.



Abbildung 21: Anfangstakte der „Mondscheinsonate“

Creator: Craig Stuart Sapp, Title: First 4 measures of the first movement of Ludwig van Beethoven's Piano Sonata in C-sharp minor "Moonlight", Op. 27, No. 2., Link: [https://de.wikipedia.org/wiki/Klaviersonate_Nr._14_\(Beethoven\)](https://de.wikipedia.org/wiki/Klaviersonate_Nr._14_(Beethoven)),
Lizenz: <https://creativecommons.org/licenses/by/3.0/#>

Um den Effekt der Halbierung, Verdopplung oder sogar Verdreifachung des Tempos zu demonstrieren, sind in den Abbildungen die jeweiligen Geraden eingezeichnet. Dadurch wird deutlich, dass die meisten Ergebnisse der Algorithmen sich an diesen Geraden orientieren und dadurch häufig hohe Werte erzeugen. Ein gutes Beispiel ist hierbei die „Mondscheinsonate“ von *Ludwig van Beethoven*, deren ersten vier Takte in Abbildung 21 zu sehen sind: Ihr wurden von allen Algorithmen das dreifache Tempo zugeschrieben, was vermutlich an der Begleitung liegt: Achteltriolen, die fälschlicherweise als Schläge interpretiert werden.

Es lässt sich also sagen, dass alle überprüften Algorithmen stark genreabhängig sind, wobei der Spotify Algorithmus die besten Ergebnisse liefert, und insbesondere Tanzmusik mit deutlichen Schlägen gut erkannt wird. Andere Genres werden dahingegen durch Verdopplungen und Verdreifachungen recht schlecht erkannt und insbesondere Mittelwerte und Mediane werden dadurch so stark verfälscht, dass sie kaum Aussagekraft haben. Somit wäre für zukünftige Arbeiten die Entwicklung eines genrestabilen Algorithmuses sehr wichtig bzw. die Entwicklungen verschiedener genrespezifischer Algorithmen in Kombination mit guter Genre Recognition.

chords_number_rate sagt in den meisten allgemeinen Vergleichen aus, dass Schlaflieder eine höhere Anzahl verschiedener Akkorde pro insgesamt auftretenden Ak-

korden hat, was überraschend ist. Bei der genauen Betrachtung des Features fällt allerdings auf, dass die überprüfte Metrik musikalisch wenig sinnvoll ist:

$$\text{chords_number_rate} = \frac{\sum_{a \in \text{Akkorde}} \text{ind}(a, f)}{\sum_{a \in \text{Akkorde}} f(a)}$$

$$\text{ind}(a, f) = \begin{cases} 1 & f(a) \geq 0 \\ 0 & \text{sonst} \end{cases}$$

Mit f der Akkordfolge des gesamten Stückes, $f(a)$ Anzahl des Akkordes a in der Folge f

So ist dieses Feature sowohl vom Tempo eines Stückes als auch von dessen Länge abhängig. Dabei hat wohl vor allem die Länge einen höheren Einfluss als die tatsächliche harmonische Grundlage des Stückes. Hinzukommt die schlechte Akkorderkennung, welche in vielen Stücken deutlich mehr verschiedene Akkorde erkennt, als eigentlich vorhanden sind (bereits in Kapitel 2.2.3 „Features basierend auf dem Chords Histogramm“ diskutiert).

average_silence_rate Die `average_silence_rate` liefert in den allgemeinen Vergleichen überraschende Ergebnisse: Bei Schlafmusik und dem Datensatz können keine großen Unterschiede festgestellt werden, während Lullabies einen etwas größeren Wert erreichen. Auch in den Genres Hip Hop, Jazz und Klassik hat Schlafmusik etwas höhere Werte als die Referenzmusik, allerdings niedrigere Werte im Pop. Beim Blick auf den Boxplot in Abbildung 22 zeigt sich, dass ein Großteil der Songs einen `average_silence_rate_mean` Wert unter 0.15 haben, somit also legato sind.

Es ist möglich, dass wieder das unterschiedliche Mastering Einfluss auf das Feature genommen hat. Eine genaue Untersuchung des Zusammenhangs zwischen Mastering und `average_silence_rate` und möglichen Methoden, um dem entgegenzuwirken, wäre eine interessante zukünftige Forschungsfragestellung.

Histogrammfeatures Wie bereits in Kapitel 2.2.3 „Features basierend auf dem Chords Histogramm“ beschrieben, hängt die Genauigkeit der Features `major_minor_ratio` und `harmonic_complexity` zum großen Teil von dem zugrunde liegenden

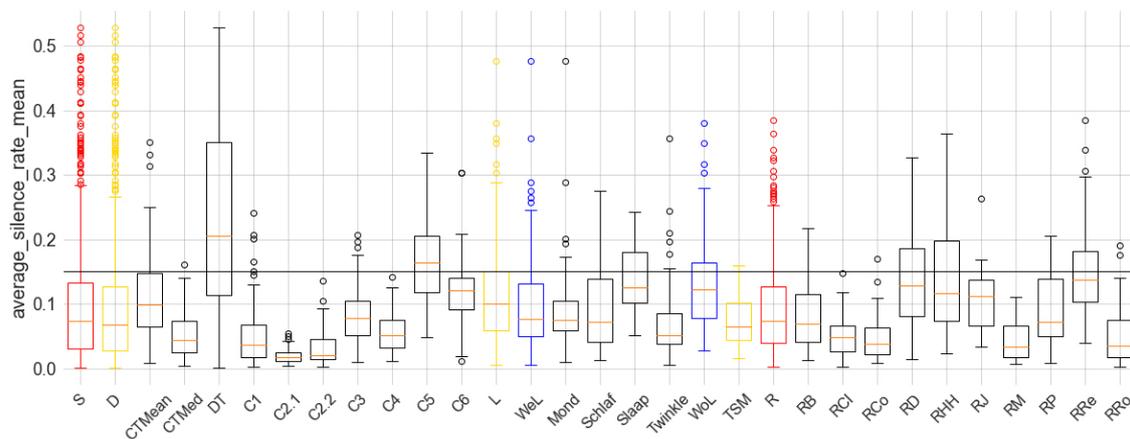


Abbildung 22: Boxplots des Features `average_silence_rate_mean`

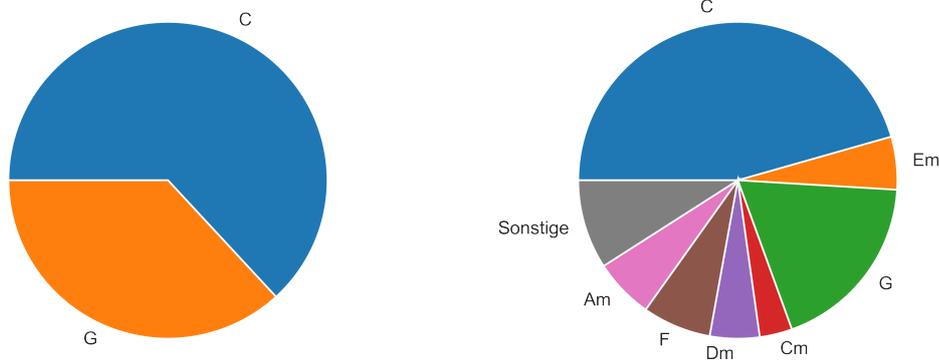
Algorithmus zur Akkorderkennung ab. Insofern ist es nicht verwunderlich, dass bei beiden Features nur leichte Tendenzen erkannt werden konnten:

Es konnte sowohl in den allgemeinen Vergleichen als auch in einigen genreinternen Vergleichen mehr Dur als in den Vergleichsongs gefunden werden. Dabei waren die Effekte bei Lullabies und in den Genres Pop und Klassik am stärksten.

Sehr ähnlich dazu verhält sich die harmonische Komplexität: Es konnte in fast jedem Vergleich eine kleinere harmonische Komplexität gefunden werden. Am deutlichsten bei den Schlafliedern und innerhalb der Genres Pop und Klassik.

Um das Ganze zu demonstrieren, wurde über alle chords-histogramme des Schlafliedes „*Schlaf, Kindchen schlaf*“ gemittelt und mit dem eigentlichen harmonischen Gut des Songs verglichen (wobei C-Dur für das tonale Zentrum des Songs steht). Die beiden Diagramme sowie die resultierenden Werte sind in Abbildung 23 zu sehen. Dabei unterscheiden sich die Werte deutlicher, als sie durch künstlerische Freiheit eigentlich sollten. Zwar ist noch immer eine Tendenz zu Dur und einer niedrigen Komplexität erkenntlich, jedoch deutlich weniger schwach ausgeprägt.

Zusammenfassend lässt sich sagen, dass die beiden Features wertvolle Ergebnisse liefern, welche jedoch aufgrund der schwachen Akkorderkennung weniger deutlich



(a) Nach Notentext

`major_minor_ratio = 1.0`

`harmonic_complexity = 0.0021`

(b) In realen Aufnahmen

`major_minor_ratio = 0.7550`

`harmonic_complexity = 0.0111`

Abbildung 23: „Schlaf, Kindchen schlaf“ Chords Histogramm Analyse

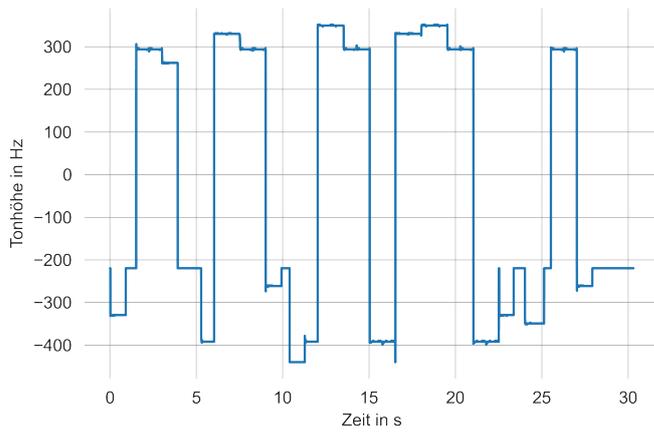
sichtbar sind als wahrscheinlich vorhanden. Eine tiefgreifende harmonische Analyse mit besserer zugrunde liegender Akkorderkennung wäre eine sehr interessante zukünftige Forschungsfrage.

Konturfeatures Da die Konturfeatures auf Melodieerkennung basieren, ist diese der limitierende Faktor für die Qualität der Konturfeatures. Insofern überrascht es nicht, dass die Features keine allzu deutlichen Ergebnisse liefern. So konnten in den allgemeinen Vergleichen im Datensatz keinerlei Unterschiede zu Referenzmusik gefunden werden, jedoch sowohl bei Schlafliedern als auch bei TSM: Die Steigung ist positiver, weniger steil und linearer. Dies ist überraschend, da in Kapitel 2.2.3 „Features basierend auf der Melodie Kontur“ mit fallenden Konturen gerechnet wurde.

Die genreinternen Vergleiche sind differenzierter: Hier kann eine Tendenz zur Linearität entdeckt werden, die anderen Metriken sind jedoch genreabhängig: So ist die Kontur bei Schlafmusik im Pop etwas steigender und weniger steil, im Hip-Hop etwas fallender, im Jazz deutlich fallender und steiler und in der Klassik steiler.



(a) Notenbild



(b) Ergebnis der Melodieerkennung

	length	slope	score
0	2.39	-0.89	0.84
1	2.99	-0.75	0.87
2	3.00	-1.12	0.87
3	4.50	-0.44	0.62
4	1.49	-0.01	0.18

(c) Tabelle zur Illustration

slope_average = -0.6793

slope_abs = 0.6793

score = 0.7142

Abbildung 24: Konturanalyse zu „Schlaf, Kindlein schlaf“

Um die Ergebnisse zu verifizieren und um insbesondere die unerwartete Steigung bei Schlafliedern zu untersuchen wurde „*Schlaf, Kindchen schlaf*“ analysiert. Das Notenbild, das Ergebnis der Melodieerkennung sowie die resultierenden Featurewerte sind in Abbildung 24 dargestellt. Dabei fällt vor allem die schlechte Melodieerkennung auf, welche die Aussagekraft der Featurewerte deutlich relativiert. Dabei ist jedoch aufgrund der Präzisionsangaben der Autoren der Melodieerkennung (56) zu vermuten, dass dies ein Beispiel für eine besonders schlechte Erkennung ist.

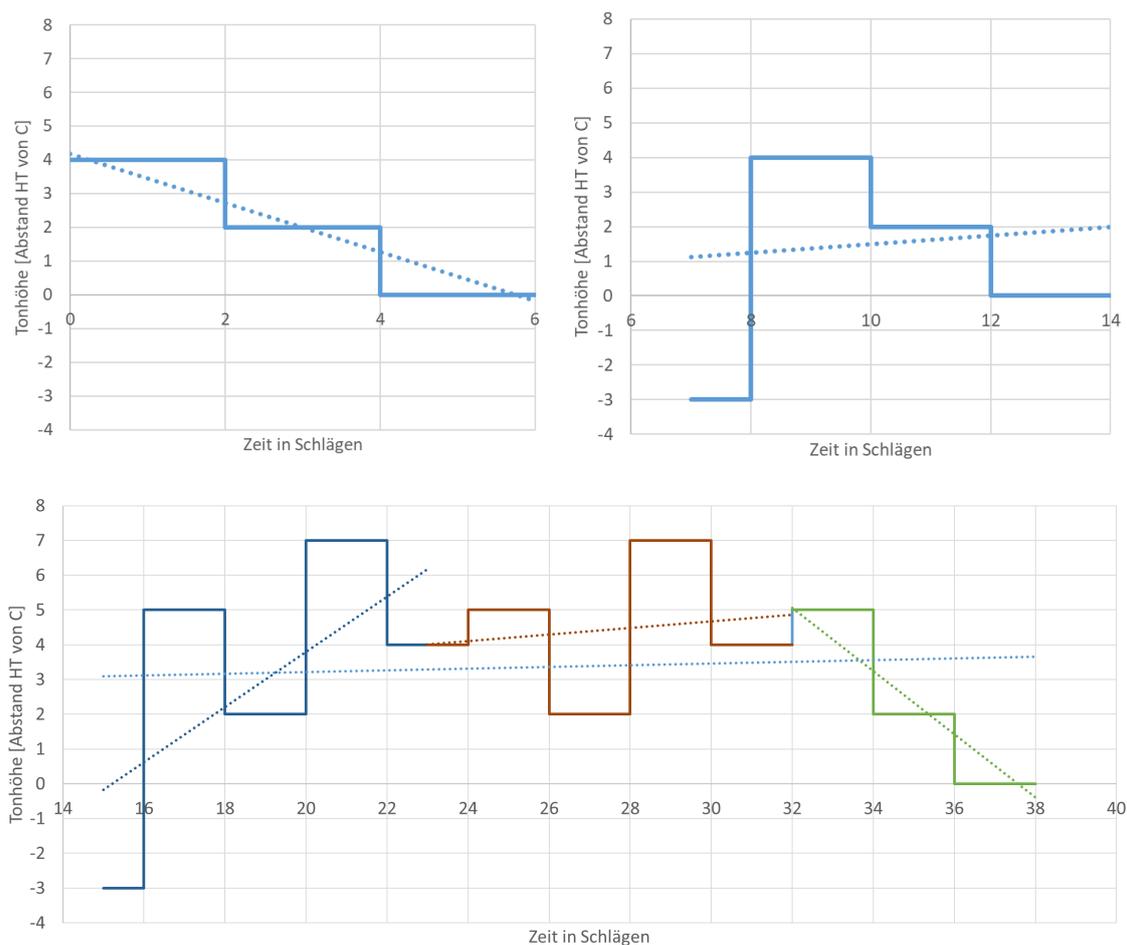


Abbildung 25: Optimale Konturanalyse zu „*Schlaf, Kindchen schlaf*“

In Abbildung 25 ist „*Schlaf, Kindchen schlaf*“ in optimaler Darstellung zu sehen. Insbesondere sind verschiedene Regressionsgeraden eingezeichnet. Jeweils in blau die

durch das in Kapitel 2.2.3 „Features basierend auf der Melodie Kontur“ vorgestellte Melodiesegmentierung. Dabei wird klar, dass sich die Steigungen zum großen Teil ausgleichen. In anderen Farben die Regressionsgeraden der einzelnen Phrasen ²³, wobei sich diese größtenteils ausgleichen. Es fällt jedoch auf, dass das letzte Intervall einer Phrase immer nach unten gerichtet ist, in drei von fünf Fällen sogar mehrere Intervalle. Diese Auffälligkeit findet sich auch in den anderen analysierten westlichen Schlafliedern, wie eine manuelle Analyse ergab.

Somit wäre es eine sehr interessante zukünftige Aufgabe ein System zu entwickeln, welches die Richtung der letzten Intervalle bestimmt. Herausforderungen sind hierbei einerseits die Melodieerkennung, andererseits die automatische Bestimmung von Phrasen, welche nicht immer durch deutliche Pausen voneinander getrennt sind.

Abschließend lässt sich sagen, dass die Features interessante Ergebnisse liefern, welche jedoch aufgrund der Qualität der Melodieerkennung deutlicher sein könnten. Die Eigenheit von Schlafliedern fallende Konturen aufzuweisen, konnte nicht bestätigt werden, was wohl am dafür unpassenden Design der Features liegt.

Similarity Features wirken zwar robust, liefern jedoch sehr überraschende Ergebnisse:

similarity_chroma liefert überraschende Ergebnisse: In allen allgemeinen Vergleichen, insbesondere Schlaflieder und TSM, sind die Schlafmusikstücke merklich weniger monoton in Bezug auf Harmonik und Melodie. Derselbe Trend kann in den genreinternen Vergleichen beobachtet werden, mit Ausnahme von Jazz. Es wird davon ausgegangen, dass sowohl die Ermittlung der Chromafeatures als auch die Erstellung und Auswertung der dazugehörigen SSM korrekt sind. Diese These wird durch den Fakt untermauert, der im Boxplot zu **similarity_chroma** in Abbildung 26 zu sehen ist: Die beiden Referenzgenres mit den geringsten Werten, also den größten Unterschieden, sind Klassik und Jazz, was den Erwartungen entspricht.

²³Eine Phrase ist ein Melodieabschnitt, welcher in einem Atemzug gesungen werden kann

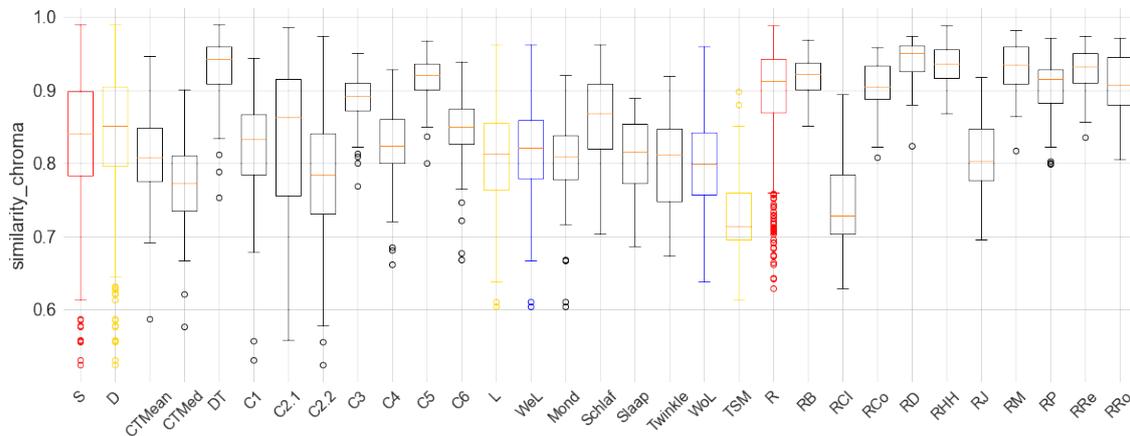


Abbildung 26: Boxplots des Features `similarity_chroma`

Die größte Varianz in diesem Feature ist in C2.1 und C2.2 zu finden, was wenig überraschend ist, da die Songs in diesen Clustern oft eher Klangteppiche als Musikstücke sind. Damit kann jedoch nicht der gesamte Unterschied erklärt werden, insbesondere nicht der von TSM und den Schlafliedern.

`similarity_timbre` liefert dieselben überraschenden Ergebnisse wie `similarity_chroma`, jedoch mit schwächerer Ausprägung. In den allgemeinen Vergleichen deutet es schwach auf eine niedrigere klangliche Monotonität mit Ausnahme des TSM Vergleichs, in welchem derselbe Vergleich deutlich stärker zum Vorschein kommt. Die genreinternen Vergleiche sind jedoch unterschiedlich, wobei die beiden stärksten Indikatoren in HipHop und Klassik in dieselbe Richtung deuten. Eventuell ist die Auswahl der MFCC Features nicht optimal, ansonsten lässt sich kein Störfaktor ausmachen. Besonders überraschend ist der Widerspruch zu den niedrigen Standardverteilungswerten der verschiedenen Klangfarbenfeatures.

`similarity_tempo` liefert einen recht neutralen Wert. In den allgemeinen Vergleichen konnten nur in TSM wesentliche Unterschiede gefunden werden. Der Aussage, dass die Songs aus TSM ein recht ungleichmäßiges Tempo aufweisen, kann

nach der manuellen Tempoanalyse, welche in Kapitel 3.2 „bpm“ durchgeführt wurde, zugestimmt werden. In den genreinternen Vergleichen deuten alle Werte auf eine schwache größere Variationsfreude im Tempo der Schlafmusik hin, bis auf den Bereich des Jazz, in welchem Schlafmusikstücke ein deutlich monotoneres Tempo haben.

Zusammenfassend lässt sich sagen, dass die Similarity-Features unerwartete Ergebnisse liefern, welche sich hauptsächlich durch das tatsächliche Vorhandensein in den Songs erklären lassen. Eventuell entsprechen die Analysemethoden nicht der Wahrnehmung, oder eine größere Variabilität innerhalb des Songs drückt etwas entspannendes aus. Eine tiefergreifende Analyse dessen wäre eine interessante Forschungsfrage für künftige Arbeiten.

3.3 Variationsarme Features

Um Features herauszufinden, welche bei Schlafmusik ähnlich sind, wurde die Standardabweichung der Gruppierung durch die mittlere Standardabweichung der Referenzgenres geteilt. Dabei bedeuten kleine Werte eine große Ähnlichkeit:

$$metric_{\text{Gruppe}} = \frac{std_{\text{Gruppe}}}{std_{ref}}$$

$$std_{ref} = \frac{1}{|\text{Genre}|} * \sum_{g \in \text{Genre}} std_g$$

Da einige Features relativ wahrgenommen werden, wie etwa Tonhöhen, wurden diese Features zuvor logarithmiert. Denn ansonsten würde dieselbe Standardabweichung um verschiedene Mittelwerte unterschiedliche Dinge beschreiben und so ist sie symmetrisch: Im Tonspektrum, welches mit einem \log_2 behandelt wurde, sagt eine Standardabweichung von 1 aus, dass sich die meisten Datenpunkte in den zwei Oktaven um den Mittelwert befinden, unabhängig davon, wo der Mittelpunkt angesiedelt ist. Dabei wurden alle spektralen Features und bpm als solche Features behandelt.

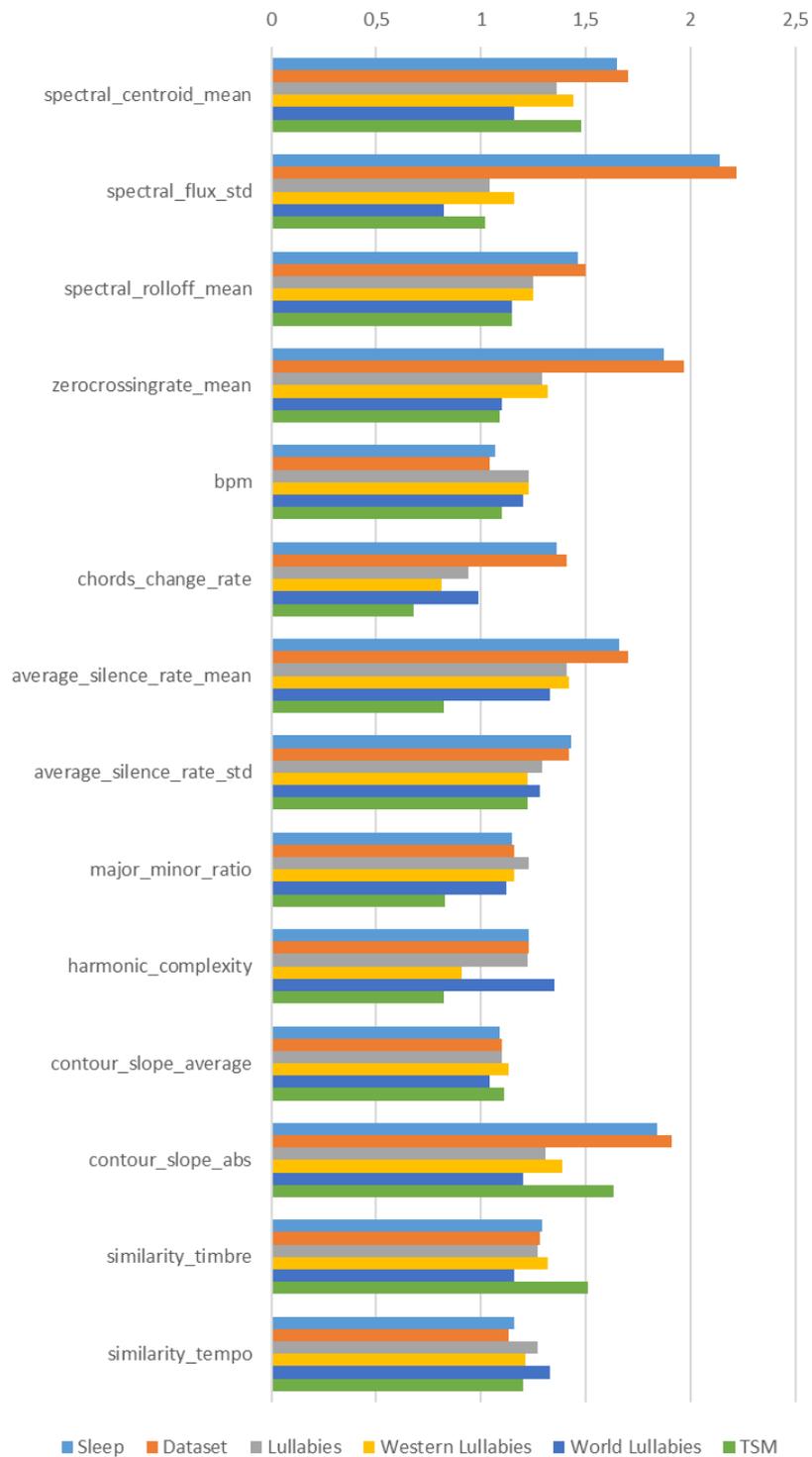


Abbildung 27: Variationsarme Features

Ausgewählte Features für die großen Gruppen sind in Abbildung 27 zu sehen. In den Tabellen 10, 11 und 12 sind alle Werte aufgelistet.

In der gesamten Schlafmusik kommen nur wenige Features an den Genremittelwert der Standardabweichungen heran: `bpm`, `major_minor_ratio` und `similarity_tempo` sind die nächsten. Aufgrund der aber eher zufälligen Streuung von `bpm`, welche in Kapitel 3.2 „bpm“ herausgefunden wurde, ist dieses Feature nicht aussagekräftig. Ebenso ist die `major_minor_ratio` nicht so genau, wie sie sein könnte, wie in Kapitel 3.2 „Histogrammfeatures“ herausgearbeitet wurde, weshalb auch sie mit etwas Vorsicht interpretiert werden muss. Dennoch wird angenommen, dass das hier gefundene Ergebnis nicht einzig aus der Ungenauigkeit der Akkorderkennung entstanden ist.

Im gesamten Datensatz sind es ebenfalls dieselben Features und insgesamt fällt auf, dass die Songs im Datensatz sich nicht so sehr ähneln, wie die Songs in den einzelnen Genre-Referenzen. Betrachtet man jedoch die einzelnen Gruppierungen im Datensatz (Tabelle 11), fällt ein sehr hoher Zusammenhalt auf. Insbesondere sind hier `spectral_centroid_mean`, `zerocrossingrate_mean`, `chords_change_rate` und `average_silence_rate_mean` auffällig. Dies sagt jedoch mehr über die Qualität der Spotify-Features aus als über Schlafmusik im Generellen. Es lässt sich also sagen, dass Songs, welche innerhalb des Spotify-Feature-Koordinatensystem nahe beieinanderliegen, auch in diesem System wahrscheinlich nahe beieinanderliegen.

Innerhalb der Lullabies stechen dreierlei andere Features hervor: `chords_change_rate`, `spectral_flux_std` und `contor_slope_abs`. Diese finden sich sowohl in westlichen Schlafliedern wieder als auch in Schlafliedern aus der ganzen Welt. Überraschenderweise streuen die Schlaflieder der ganzen Welt insgesamt weniger als die westlichen Schlaflieder. Die Werte der einzelnen westlichen Schlaflieder (Tabelle 12) überraschen: Obwohl jeweils die selben Songs zugrunde liegen, sind die Ähnlichkeiten doch nur verhalten zu sehen. Insbesondere die Aufnahmen zu „*Der Mond ist aufgegangen*“ und „*Schlaf, Kindlein schlaf*“ sind unterschiedlicher als gedacht. Dass allerdings in den Aufnahmen zu „*Schlaf, Kindlein schlaf*“ Metal- und Rockversionen zu finden sind, erklärt dies. Insgesamt sind die Features, welche den größten Zusammenhalt in den westlichen Lullabies genießen, `spectral_flux_std`,

`chords_change_rate`, `major_minor_ratio`, `harmonic_complexity` und `contour_slope_abs`.

Therapeutical Sleep Music streut recht wenig, weshalb einige Features hier geringe Werte liefern: Besonders `chords_change_rate`, `average_silence_rate_mean`, `harmonic_complexity` und `major_minor_ratio`.

Insgesamt lässt sich sagen, dass die meisten Features in der untersuchten Schlafmusik deutlich mehr streuen als in den Referenzgenres. Ausnahmen bilden `major_minor_ratio`, `similarity_tempo` und `harmonic_complexity`. Innerhalb von Lullabies und TSM gibt es jedoch einige Features mit geringer Streuung und damit hoher Ähnlichkeit.

3.4 Diskussion

Insgesamt fällt auf, dass der Datensatz verglichen mit TSM deutlich mehr streut. Dennoch haben Lieder des Datensatzes in nahezu allen Features im Vergleich zur Referenzmusik Ausprägungen in dieselbe Richtung, wenn auch nicht so stark wie TSM. Dies deutet auf eine allgemeine beruhigende Charakteristik hin.

Ähnliches fällt im Vergleich mit Lullabies auf, wobei diese Schlaflieder mehr streuen als TSM und andere Schwerpunkte setzen: So scheinen eine hohe `major_minor_ratio` und geringe `harmonic_complexity` für die Schlaflieder wichtiger zu sein als für TSM.

Bei den genreinternen Vergleichen konnten nicht alle Tendenzen in allen Genres wiedergefunden werden, was betont, dass Features innerhalb der Genres unterschiedliche Ausprägungen und Aussagekraft besitzen.

Insgesamt konnten vor allem Klangfarben herausgearbeitet werden, welche für Schlafmusik wichtig sind: Ein wenig voller, dunkler, sanfter, wenig rauschender Klang, welcher sich über das Stück vergleichsweise wenig verändert. Musikalische Features sind aufgrund verschiedener algorithmischer Probleme schwieriger nachzuweisen, jedoch sind wenige Tonanschläge wichtig und überraschenderweise erhöhte

Veränderungen im Stück in Bezug auf Chromafeatures und Klangfarbe, was im Gegensatz zu den Ergebnissen der `_std` Features und `harmonic_complexity` steht.

Dass die Songs im Datensatz im Mittel in dieselbe Richtung tendieren wie TSM und eine große Streuung haben, legt die Vermutung nahe, dass Menschen insgesamt wissen, was gut zum Einschlafen ist. Dies wird jedoch in der Praxis nicht konsequent ausgeführt. Verschiedene mögliche Erklärungen wurden bereits in (57) genannt:

So könnte mittels der Theorie des Predictive Codings erklärt werden, dass auch aufregende Musik auf Menschen beruhigend wirken könnte, wenn diese sie bereits häufig gehört haben (57). Dies wird durch eine Studie gestützt, in welcher herausgefunden wurde, dass Musik-Präferenz und Vertrautheit der Songs sehr wichtige Rollen für den Entspannungseffekt spielen (61).

Ebenfalls wurde in einer Studie herausgefunden, dass Musik zum Schlafen aus verschiedenen Gründen gehört wird: So kann Musik Schlaf stimulieren, ist Teil einer Schlafroutine, kann einen mentalen oder physischen Status erzeugen oder von anderen Einflüssen ablenken (63). Dies erklärt gleichfalls eine größere Streuung, genauso wie die Erkenntnis, dass Menschen nicht immer die Musik mit dem besten gewünschten Effekt auswählen (21; 30; 29).

4 Fazit

In dieser Arbeit wurden Repräsentanten des Datensatzes, welcher in (57) erzeugt wurde, mittels verschiedener Music Information Retrieval Features untersucht. Dabei wurden neue Features entwickelt, welche auf bereits bestehenden Features aufbauen. Um die Ergebnisse einzuordnen und von anderer Musik abzugrenzen, wurden Referenzsongs aus zehn Genres mittels Spotify herausgearbeitet. Um die Ergebnisse hinsichtlich der Eignung zum Schlafen zu bewerten wurden Schlaflieder und Lieder, welche in Studien als gut zum Einschlafen bewertet wurden, auf dieselbe Art und Weise wie der Datensatz analysiert und die Ergebnisse verglichen.

Dabei ergab sich, dass musikalische Features zum Teil noch zu undeutliche Ergebnisse liefern, als dass diese eine große Aussagekraft haben. So ist insbesondere die Tempoerkennung noch stark genreabhängig und liefert für viele Genres schlechte Ergebnisse, welche eine Aussage in dieser Analyse nicht möglich machen. Auch andere Algorithmen, welche als Grundlage für neue Features benutzt wurden, arbeiten noch nicht so genau wie erhofft, etwa die benutzte Akkord- und Melodieerkennung. Dennoch sind hier leichte Tendenzen zu sehen. Eine ähnliche Analyse mittels genauerer Algorithmen in der Zukunft wäre sinnvoll und könnte diese bisher leichten Tendenzen möglicherweise bestätigen.

Die Ergebnisse deuten insofern auf einen allgemeinen Charakter von Schlafmusik hin, als dass die meisten Features der Songs im Datensatz im Mittel ähnliche Ausprägungen aufweisen wie Schlaflieder und Lieder, welche in Studien als gut zum Einschlafen bewertet wurden. Dennoch sind diese Ausprägungen deutlich schwächer und die Streuung ist merklich stärker. Die eben beschriebenen Charakteristiken sind zum Großteil klanglicher Natur: Ein wenig voller, dunkler, sanfter, wenig rauschender Klang, welcher sich über das Stück vergleichsweise wenig verändert. Außerdem konnte ein musikalisches Feature herausgearbeitet werden: Wenige Tonansätze scheinen wichtig zu sein. Dies könnte mit einem niedrigen Tempo zusammenhängen.

Schlafprobleme sind ein wachsendes gesellschaftliches Problem und Erkenntnisse im Bereich der Schlafmusik können Menschen helfen, Musik auszuwählen, welche es ihnen ermöglicht besser einzuschlafen. Außerdem wurden neue Features für eine

automatische musikalische Analyse von Aufnahmen von Liedern vorgestellt. Diese Arbeit zeigt, dass eine solche Analyse bereits für einige Features sehr gut möglich ist, während dies für andere vorraussichtlich in Zukunft der Fall sein wird. Dadurch könnten größere Datensätze von Aufnahmen analysiert werden als manuell möglich wäre.

A Spotifyfeatures

Im Folgenden werden die Spotify Features vorgestellt mittels ihrer Beschreibung in der Spotify API Dokumentation ²⁴.

acousticness A confidence measure from 0.0 to 1.0 of whether the track is acoustic. 1.0 represents high confidence the track is acoustic.

danceability Danceability describes how suitable a track is for dancing based on a combination of musical elements including tempo, rhythm stability, beat strength, and overall regularity. A value of 0.0 is least danceable and 1.0 is most danceable.

energy Energy is a measure from 0.0 to 1.0 and represents a perceptual measure of intensity and activity. Typically, energetic tracks feel fast, loud, and noisy. For example, death metal has high energy, while a Bach prelude scores low on the scale. Perceptual features contributing to this attribute include dynamic range, perceived loudness, timbre, onset rate, and general entropy.

instrumentalness Predicts whether a track contains no vocals. „Ooh” and „aah” sounds are treated as instrumental in this context. Rap or spoken word tracks are clearly ”vocal”. The closer the instrumentalness value is to 1.0, the greater likelihood the track contains no vocal content. Values above 0.5 are intended to represent instrumental tracks, but confidence is higher as the value approaches 1.0.

liveness Detects the presence of an audience in the recording. Higher liveness values represent an increased probability that the track was performed live. A value above 0.8 provides strong likelihood that the track is live.

²⁴<https://developer.spotify.com/documentation/web-api/reference/>
Accessed: 2022-09-05

loudness The overall loudness of a track in decibels (dB). Loudness values are averaged across the entire track and are useful for comparing relative loudness of tracks. Loudness is the quality of a sound that is the primary psychological correlate of physical strength (amplitude). Values typically range between -60 and 0 db.

speechiness Speechiness detects the presence of spoken words in a track. The more exclusively speech-like the recording (e.g. talk show, audio book, poetry), the closer to 1.0 the attribute value. Values above 0.66 describe tracks that are probably made entirely of spoken words. Values between 0.33 and 0.66 describe tracks that may contain both music and speech, either in sections or layered, including such cases as rap music. Values below 0.33 most likely represent music and other non-speech-like tracks.

tempo The overall estimated tempo of a track in beats per minute (BPM). In musical terminology, tempo is the speed or pace of a given piece and derives directly from the average beat duration.

valence A measure from 0.0 to 1.0 describing the musical positiveness conveyed by a track. Tracks with high valence sound more positive (e.g. happy, cheerful, euphoric), while tracks with low valence sound more negative (e.g. sad, depressed, angry).

B CDs Schlaflieder aus aller Welt

In Tabelle 3 sind die benutzten CDs von ellipsis arts mitsamt der dazugehörigen ISBN aufgelistet.

Name	ISBN
Guitar Lullaby	1-55961-621-0
Mediterranean Lullaby	1-55961-625-3
Latin Lullaby	1-55961-475-7
Celtic Lullaby	1-55961-493-5
Cuban Lullaby	1-55961-623-7
Brazilian Lullaby	1-55961-558-3
African Lullaby	1-55961-523-0

Tabelle 3: CDs Schlaflieder aus aller Welt

C Tabellen für Analyse

Auf den folgenden Seiten finden sich die Tabellen mit den genauen Daten zu den Vergleichen aus Kapitel 3 „Analyse“.

Feature	S vs R	D vs R	L vs R	TSM vs R
average_loudness	-0.42	-0.25	-1.04	-4.15
barkbands_spread_mean	-1.29	-1.21	-1.30	-1.25
barkbands_spread_std	-0.81	-0.78	-0.88	-1.37
spectral_centroid_mean	-1.08	-1.04	-0.96	-1.02
spectral_centroid_std	-0.02	-0.17	0.81	-0.46
spectral_flux_mean	-1.01	-0.87	-2.02	-2.16
spectral_flux_std	-0.48	-0.37	-1.39	-1.43
spectral_spread_mean	-0.80	-0.78	-0.88	-1.75
spectral_spread_std	0.07	-0.09	1.04	-0.00
spectral_rolloff_mean	-0.46	-0.44	-0.45	-0.71
spectral_rolloff_std	-0.20	-0.23	0.31	-0.89
zerocrossingrate_mean	-0.69	-0.65	-0.69	-0.94
zerocrossingrate_std	-0.07	-0.02	0.19	-1.17
onset_rate	-0.65	-0.61	-1.21	-1.72
bpm	-0.23	-0.31	-0.03	0.34
chords_number_rate	0.34	0.34	0.66	-0.15
chords_change_rate	-0.42	-0.41	-0.69	-0.92
average_silence_rate_mean	0.08	0.03	0.35	-0.22
average_silence_rate_std	0.24	0.16	0.68	0.23
major_minor_ratio	0.07	0.05	0.65	0.33
harmonic_complexity	-0.01	-0.02	-0.46	-0.15
contour_slope_average	0.00	-0.02	0.18	0.50
contour_slope_abs	-0.09	-0.05	-0.34	-0.14
contour_linearity_score	0.10	0.03	0.62	0.88
similarity_chroma	-0.71	-0.62	-1.15	-2.15
similarity_timbre	-0.21	-0.20	-0.25	-1.21
similarity_tempo	-0.00	0.04	-0.04	-0.73

Tabelle 4: Allgemeine Vergleiche mittels Cohen’s d. Werte mit Betrag über 1 wurden hervorgehoben.

Feature	S vs R	D vs R	L vs R	TSM vs R
average_loudness	-1.68	-0.37	-3.57	-42.44
barkbands_spread_mean	-14.35	-15.43	-25.52	-62.45
barkbands_spread_std	-0.22	0.33	-6.03	-37.32
spectral_centroid_mean	-18.18	-23.78	-6.03	91.71
spectral_centroid_std	11.19	15.31	10.75	0.17
spectral_flux_mean	-1.16	-1.15	-17.96	-25.62
spectral_flux_std	3.99	7.32	-3.35	-44.07
spectral_spread_mean	0.77	2.63	-0.97	31.31
spectral_spread_std	-13.07	-14.77	-15.95	-7.05
spectral_rolloff_mean	36.89	42.66	4.45	9.27
spectral_rolloff_std	-3.76	-4.17	29.06	116.95
zerocrossingrate_mean	-3.74	-4.56	2.90	-34.28
zerocrossingrate_std	-3.67	2.01	-23.37	-96.45
onset_rate	3.88	-2.33	-5.29	18.02
bpm	-2.79	-6.10	-1.00	-1.85
chords_number_rate	2.33	4.52	2.64	-4.39
chords_change_rate	-2.03	-4.36	-4.12	-1.47
average_silence_rate_mean	-1.05	-0.19	-0.71	44.87
average_silence_rate_std	-3.57	2.44	-1.85	-20.56
major_minor_ratio	0.83	-0.98	5.10	11.99
harmonic_complexity	2.16	3.59	0.16	-44.20
contour_slope_average	-7.86	-1.39	3.97	7.67
contour_slope_abs	-1.02	-0.77	-7.27	-29.87
contour_linearity_score	-1.12	-7.85	2.97	45.67
similarity_chroma	0.75	4.77	14.25	-22.01
similarity_timbre	0.84	-1.76	0.33	12.18
similarity_tempo	9.43	10.34	31.65	-133.52
Bias	1	-3	-41	-95
Fehlerwahrscheinlichkeit	0.2438	0.2948	0.1297	0.0

Tabelle 5: Allgemeine Vergleiche mittels Perceptron: Weight Vector Einträge über 30 sind hervorgehoben

Featuregruppe	S vs R	D vs R	L vs R	TSM vs R
Alle Feature	0.165	0.146	0.381	0.186
Lautstärke und Artikulation	-0.016	-0.008	0.096	0.486
Klangfarbe	0.195	0.160	0.220	0.136
Rhythmus und Tempo	0.007	-0.002	0.256	0.064
Harmonik	0.010	0.009	0.061	-0.009
Kontur	0.000	-0.001	0.017	0.017
Rolloff	0.089	0.086	-0.008	-0.008

Tabelle 6: Allgemeiner Vergleich mittels Adjusted Rand Index. Werte über 0.3 sind hervorgehoben

Feature	C3 vs RP	C4 vs RP	C5 vs RH	C6 vs RJ	TSM vs RCI
average_loudness	-0.55	-0.34	-0.45	1.16	-2.06
barkbands_spread_mean	0.01	-2.08	-0.33	0.77	0.54
barkbands_spread_std	0.13	-1.10	0.21	-0.19	0.44
spectral_centroid_mean	-0.29	-2.00	-0.62	0.60	0.39
spectral_centroid_std	-0.08	-0.50	0.21	-0.27	0.19
spectral_flux_mean	-0.54	-2.32	-0.35	0.38	-0.96
spectral_flux_std	-0.02	-2.17	0.12	0.98	-0.28
spectral_spread_mean	-0.14	-1.67	-0.19	-1.48	0.55
spectral_spread_std	0.20	-0.03	0.32	-0.52	0.02
spectral_rolloff_mean	0.16	-1.22	0.04	0.59	0.59
spectral_rolloff_std	-0.04	-0.41	0.21	-0.53	0.15
zerocrossingrate_mean	0.04	-1.48	-0.02	0.10	0.64
zerocrossingrate_std	-0.21	-0.13	0.12	-0.09	-0.26
onset_rate	-0.59	-1.74	-0.38	2.21	-0.51
bpm	0.23	-0.07	-0.03	-0.83	0.05
chords_number_rate	-0.52	-0.31	0.07	1.14	-0.80
chords_change_rate	-0.69	-1.26	0.10	-0.21	0.29
average_silence_rate_mean	-0.18	-0.97	0.40	0.42	0.63
average_silence_rate_std	0.24	-0.41	0.68	-0.32	0.46
major_minor_ratio	0.19	0.59	-0.20	-0.11	0.52
harmonic_complexity	-0.32	-0.68	0.19	-0.15	-0.47
contour_slope_average	0.16	0.34	-0.19	-0.76	0.02
contour_slope_abs	-0.50	-0.60	-0.09	0.50	0.46
contour_linearity_score	-0.16	0.00	0.10	0.22	1.16
similarity_chroma	-0.36	-1.65	-0.49	0.91	0.08
similarity_timbre	0.17	0.25	-0.62	0.31	-0.86
similarity_tempo	-0.27	-0.44	-0.45	2.22	-0.58

Tabelle 7: Genre interne Vergleiche mittels Cohen's d. Werte mit Betrag über 1 sind hervorgehoben

Feature	C3 vs RP	C4 vs RP	C5 vs RH	C6 vs RJ	TSM vs RCI
average_loudness	-12.79	-27.56	-0.24	9.20	-14.08
barkbands_spread_mean	-3.09	-23.68	-19.37	3.30	-2.12
barkbands_spread_std	-8.84	-6.29	13.50	-3.73	1.44
spectral_centroid_mean	-26.48	-28.36	-9.89	-1.00	-2.87
spectral_centroid_std	5.11	-7.64	5.45	0.90	1.07
spectral_flux_mean	0.52	-1.30	5.53	-16.76	-7.86
spectral_flux_std	15.42	-22.26	4.16	9.74	0.32
spectral_spread_mean	10.79	-10.54	-1.42	-8.54	3.63
spectral_spread_std	-4.68	1.31	-0.32	-3.55	-1.37
spectral_rolloff_mean	82.82	11.35	3.20	0.11	1.29
spectral_rolloff_std	-14.53	24.99	-6.36	0.53	3.44
zerocrossingrate_mean	-43.24	3.54	21.28	-1.20	0.14
zerocrossingrate_std	-2.38	-17.20	-12.02	11.46	-2.64
onset_rate	-1.24	-19.44	-8.11	21.43	-4.76
bpm	2.27	-7.44	5.01	0.76	-1.97
chords_number_rate	-3.30	8.58	2.38	22.59	-5.92
chords_change_rate	-8.16	-5.76	0.09	-5.50	-1.45
average_silence_rate_mean	-17.77	-21.48	1.11	1.23	-0.17
average_silence_rate_std	10.18	17.06	2.65	-4.69	-5.40
major_minor_ratio	-1.39	9.50	-5.29	4.25	2.36
harmonic_complexity	10.63	-1.63	0.04	-9.29	-5.88
contour_slope_average	0.22	12.06	0.01	-8.11	-3.41
contour_slope_abs	-7.25	8.37	9.05	7.78	-0.46
contour_linearity_score	-0.49	-1.80	-5.97	-1.40	9.06
similarity_chroma	10.50	8.70	4.06	8.76	-0.60
similarity_timbre	12.76	3.37	-9.42	0.54	-0.45
similarity_tempo	-20.60	-9.18	-38.81	3.85	-12.06
Bias	28	-10	38	3	-13
Fehlerwahrscheinlichkeit	0.1778	0.0	0.2627	0.0	0.0

Tabelle 8: Genre interne Vergleiche mittels Perceptron. Werte mit Betrag über 30 sind hervorgehoben

Vergleich	C3 vs RP	C4 vs RP	C5 vs RH	C6 vs RJ	TSM vs RCI
Alle Feature	0.027	0.541	0.027	0.567	0.149
Lautstärke und Artikulation	-0.006	0.073	0.057	0.096	0.011
Klangfarbe	0.006	0.427	-0.011	0.041	0.102
Rhythmus und Tempo	0.125	0.449	0.005	0.613	0.075
Harmonik	0.085	0.272	-0.004	-0.007	0.002
Kontur	0.009	0.011	-0.007	0.056	0.196
Rolloff	-0.004	0.122	0.001	-0.003	0.041

Tabelle 9: Genre interner Vergleich mittels Adjusted Rand Index. Werte über 0.3 sind hervorgehoben

Feature	S	D	L	WeL	WoL	TSM
average_loudness	2.27	1.81	2.80	2.51	2.75	2.57
barkbands_spread_mean	1.92	1.92	1.84	1.84	1.76	1.70
barkbands_spread_std	2.22	2.18	2.24	2.07	2.36	2.65
spectral_centroid_mean	1.65	1.70	1.36	1.44	1.16	1.48
spectral_centroid_std	1.65	1.60	1.52	1.63	1.23	1.72
spectral_flux_mean	2.11	1.87	2.31	2.70	1.60	1.95
spectral_flux_std	2.14	2.22	1.04	1.16	0.82	1.02
spectral_spread_mean	3.05	3.10	2.80	2.96	2.54	2.69
spectral_spread_std	1.55	1.49	1.40	1.51	1.16	1.33
spectral_rolloff_mean	1.46	1.50	1.25	1.25	1.15	1.15
spectral_rolloff_std	1.48	1.44	1.52	1.51	1.51	2.04
zerocrossingrate_mean	1.87	1.97	1.29	1.32	1.10	1.09
zerocrossingrate_std	1.32	1.29	1.68	1.75	1.42	1.48
onset_rate	1.93	2.00	1.35	1.31	1.12	1.22
bpm	1.07	1.04	1.23	1.23	1.20	1.10
chords_number_rate	2.20	2.19	2.39	2.38	2.42	1.16
chords_change_rate	1.36	1.41	0.94	0.81	0.99	0.68
average_silence_rate_mean	1.66	1.70	1.41	1.42	1.33	0.82
average_silence_rate_std	1.43	1.42	1.29	1.22	1.28	1.22
major_minor_ratio	1.15	1.16	1.23	1.16	1.12	0.83
harmonic_complexity	1.23	1.23	1.22	0.91	1.35	0.82
contour_slope_average	1.58	1.62	1.47	1.56	1.33	1.07
contour_slope_abs	1.42	1.46	1.07	1.10	1.03	1.33
contour_linearity_score	1.49	1.49	1.56	1.71	1.33	1.23
similarity_chroma	2.15	2.12	1.80	1.79	1.78	1.87
similarity_timbre	1.29	1.28	1.27	1.32	1.16	1.51
similarity_tempo	1.16	1.13	1.27	1.21	1.33	1.20

Tabelle 10: Variationsarme Features. Werte kleiner gleich 1.2 sind hervorgehoben.

Feature	CTMean	CTMed	DT	C1	C2.1	C2.2	C3	C4	C5	C6
average_loudness	2.68	0.92	2.20	1.40	0.85	1.66	1.20	1.14	1.46	0.47
barkbands_spread_mean	1.29	1.02	1.69	1.17	0.95	1.18	0.79	0.73	0.92	1.38
barkbands_spread_std	1.70	1.48	1.72	1.26	1.20	1.76	0.90	1.10	1.01	1.09
spectral_centroid_mean	0.95	0.74	1.80	1.17	0.77	0.97	0.76	0.77	0.80	1.19
spectral_centroid_std	1.16	1.52	1.39	1.17	1.97	2.07	0.59	1.06	0.76	1.25
spectral_flux_mean	1.93	1.13	2.02	1.52	1.40	1.82	0.98	0.92	0.94	0.89
spectral_flux_std	2.08	1.03	2.19	2.17	1.25	1.15	1.35	0.76	0.88	0.86
spectral_spread_mean	2.03	2.52	2.25	1.84	2.45	2.36	0.93	0.93	1.22	1.17
spectral_spread_std	1.04	1.29	1.59	1.60	1.78	1.90	0.66	0.98	0.79	1.13
spectral_rolloff_mean	0.94	0.69	2.29	1.19	0.72	0.91	0.87	0.83	1.10	1.08
spectral_rolloff_std	1.22	1.23	1.73	1.40	1.27	1.62	0.75	0.96	1.21	1.09
zerocrossingrate_mean	0.74	0.39	4.66	1.33	0.36	0.54	0.97	0.68	1.03	0.66
zerocrossingrate_std	1.10	1.15	2.01	1.26	0.77	1.28	0.93	1.22	1.25	0.83
onset_rate	1.11	1.23	1.57	2.22	1.10	1.40	0.84	0.76	0.79	0.60
bpm	0.96	1.25	1.34	1.09	1.04	0.86	0.54	0.42	0.90	0.85
chords_number_rate	1.65	1.25	3.62	2.43	1.39	1.31	0.81	1.14	1.28	2.27
chords_change_rate	0.69	0.73	2.04	0.99	0.63	0.66	0.75	0.71	1.08	0.67
average_silence_rate_mean	1.32	0.70	2.81	0.95	0.20	0.48	0.83	0.52	1.22	0.89
average_silence_rate_std	1.71	1.04	1.62	1.40	0.70	1.04	1.04	0.90	0.95	1.13
major_minor_ratio	0.99	1.02	1.16	1.16	1.57	1.27	0.95	0.99	0.97	0.87
harmonic_complexity	1.04	1.00	1.41	1.09	1.65	1.42	0.75	0.80	1.11	0.89
contour_slope_average	1.83	1.59	2.61	1.36	0.73	1.17	1.02	0.96	1.40	2.01
contour_slope_abs	1.38	1.09	2.21	1.07	0.76	0.99	0.75	0.67	1.46	1.43
contour_linearity_score	1.54	1.17	1.64	1.70	1.78	1.61	1.16	0.91	1.03	1.22
similarity_chroma	1.57	1.58	1.18	1.89	2.74	2.46	0.93	1.40	0.80	1.15
similarity_timbre	1.39	1.18	1.21	1.46	1.21	1.39	0.86	0.91	1.09	1.00
similarity_tempo	1.44	1.17	0.70	1.48	0.53	1.18	0.82	0.91	0.90	0.54

Tabelle 11: Variationsarme Features innerhalb des Datensatzes. Werte kleiner gleich 1.0 sind hervorgehoben.

Feature	Mond	Schlaf	Slaap	Twinkle
average_loudness	3.18	1.45	2.60	1.44
barkbands_spread_mean	1.53	1.94	1.61	2.14
barkbands_spread_std	2.06	2.04	2.03	2.13
spectral_centroid_mean	1.16	1.61	1.17	1.66
spectral_centroid_std	1.23	1.77	1.12	1.94
spectral_flux_mean	2.69	3.02	2.37	2.27
spectral_flux_std	1.49	1.29	0.88	0.84
spectral_spread_mean	2.17	2.46	2.85	3.86
spectral_spread_std	1.06	1.66	0.92	1.76
spectral_rolloff_mean	1.07	1.33	1.11	1.40
spectral_rolloff_std	1.40	1.63	1.28	1.51
zerocrossingrate_mean	0.97	1.37	1.27	1.56
zerocrossingrate_std	1.47	1.66	2.04	1.61
onset_rate	1.81	1.07	0.62	1.39
bpm	1.27	0.93	1.32	1.15
chords_number_rate	1.87	1.57	2.64	1.92
chords_change_rate	0.77	1.03	0.62	0.65
average_silence_rate_mean	1.56	1.51	1.05	1.31
average_silence_rate_std	1.03	1.29	1.05	1.29
major_minor_ratio	1.12	1.37	0.71	0.95
harmonic_complexity	1.07	1.19	0.52	0.56
contour_slope_average	1.01	1.56	1.86	1.59
contour_slope_abs	1.11	0.99	1.21	0.99
contour_linearity_score	1.41	1.71	1.62	1.80
similarity_chroma	1.80	1.65	1.38	1.86
similarity_timbre	1.28	1.24	1.09	1.47
similarity_tempo	1.39	1.48	0.97	0.93

Tabelle 12: Variationsarme Features Westliche Lullabies. Werte kleiner gleich 1.0 sind hervorgehoben.

Literatur

- [1] The bark frequency scale. http://ccrma.stanford.edu/~jos/bbt/Bark_Frequency_Scale.html. Accessed: 2022-07-25.
- [2] Central moment. http://en.wikipedia.org/wiki/Central_moment. Accessed: 2022-07-25.
- [3] Energy (signal processing). [https://en.wikipedia.org/wiki/Energy_\(signal_processing\)](https://en.wikipedia.org/wiki/Energy_(signal_processing)). Accessed: 2022-07-25.
- [4] Function centroid. <http://mathworld.wolfram.com/FunctionCentroid.html>. Accessed: 2022-07-25.
- [5] Loudness normalization. <https://artists.spotify.com/help/article/loudness-normalization>. Accessed: 2022-07-25.
- [6] Meyda audio features. <https://meyda.js.org/audio-features>. Accessed: 2022-07-25.
- [7] Sample central moment. <http://mathworld.wolfram.com/SampleCentralMoment.html>. Accessed: 2022-07-25.
- [8] Spectral centroid. https://en.wikipedia.org/wiki/Spectral_centroid. Accessed: 2022-07-25.
- [9] Spectral flux. http://en.wikipedia.org/wiki/Spectral_flux. Accessed: 2022-07-25.
- [10] Stevens's power law. https://en.wikipedia.org/wiki/Stevens's_power_law. Accessed: 2022-07-25.
- [11] Zero crossing. http://en.wikipedia.org/wiki/Zero-crossing_rate. Accessed: 2022-07-25.
- [12] Pearson's correlation coefficient. In *Encyclopedia of Public Health*, W. Kirch, Ed. Springer Netherlands, Dordrecht, 2008, pp. 1090–1091.

- [13] ARITAKE, S., KANEITA, Y., UCHIYAMA, M., MISHIMA, K., AND OHIDA, T. Non-pharmacological self-management of sleep among the japanese general population. *Journal of clinical sleep medicine : JCSM : official publication of the American Academy of Sleep Medicine* 5 (10 2009), 464–9.
- [14] BELLO, J., DUXBURY, C., DAVIES, M., AND SANDLER, M. On the use of phase and energy for musical onset detection in the complex domain. *Signal Processing Letters, IEEE* 11 (07 2004), 553 – 556.
- [15] BERNARDI, L., PORTA, C., AND SLEIGHT, P. Cardiovascular, cerebrovascular, and respiratory changes induced by different types of music in musicians and non-musicians: the importance of silence. *Heart* 92, 4 (2006), 445–452.
- [16] BOGDANOV, D., WACK, N., GÓMEZ, E., SANKALP, G., HERRERA, P., MAYOR, O., ROMA, G., SALAMON, J., ZAPATA, J., AND SERRA, X. Essentia: an audio analysis library for music information retrieval.
- [17] BROSSIER, P., BELLO, J., AND PLUMBLEY, M. Fast labelling of notes in music signals., 2004.
- [18] CALEM, M., BISLA, J., BEGUM, A., DEWEY, M., BEBBINGTON, P., BRUGHA, T., COOPER, C., JENKINS, R., LINDESAY, J., MCMANUS, S., MELTZER, H., SPIERS, N., WEICH, S., AND STEWART, R. Increased prevalence of insomnia and changes in hypnotics use in england over 15 years: Analysis of the 1993, 2000, and 2007 national psychiatric morbidity surveys. *Sleep* 35, 3 (Mar. 2012), 377 – 384.
- [19] CAPPUCCIO, F., TAGGART, F., KANDALA, N.-B., CURRIE, A., PEILE, E., STRANGES, S., AND MILLER, M. Meta-analysis of short sleep duration and obesity in children and adults. *Sleep* 31 (05 2008), 619–26.
- [20] CAPPUCCIO, F. P., D’ELIA, L., STRAZZULLO, P., AND MILLER, M. A. Quantity and Quality of Sleep and Incidence of Type 2 Diabetes: A systematic review and meta-analysis. *Diabetes Care* 33, 2 (11 2009), 414–420.

- [21] CARLSON, E., SAARIKALLIO, S., TOIVIAINEN, P., BOGERT, B., KLIUCHKO, M., AND BRATTICO, E. Maladaptive and adaptive emotion regulation through music: a behavioral and neuroimaging study of males and females. *FRONTIERS IN HUMAN NEUROSCIENCE* 9 (Aug. 2015), 1–13.
- [22] COHEN, J. *Statistical Power Analysis for the Behavioral Sciences*, 2nd ed. Erlbaum, 1988.
- [23] DEGARA, N., RUA, E. A., PENA, A., TORRES-GUIJARRO, S., DAVIES, M. E., AND PLUMBLEY, M. D. Reliability-informed beat tracking of musical signals. *IEEE Transactions on Audio, Speech, and Language Processing* 20, 1 (2012), 290–301.
- [24] DICKSON, G. T., AND SCHUBERT, E. Musical features that aid sleep. *Musicae Scientiae* (2020).
- [25] DIXON, S. Onset detection revisited. In *In Proceedings of the 9th international conference on digital audio effects* (2006), pp. 133–137.
- [26] ELLIS, D. Beat tracking by dynamic programming. *Journal of New Music Research* 36 (03 2007), 51–60.
- [27] FENG, Y., ZHUANG, Y., AND PAN, Y. Popular music retrieval by detecting mood. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval* (New York, NY, USA, 2003), SIGIR '03, Association for Computing Machinery, p. 375–376.
- [28] GARLAND, S., ROWE, H., REPA, L., FOWLER, K., ZHOU, E., AND GRANDNER, M. A decade’s difference: 10-year change in insomnia symptom prevalence in canada depends on sociodemographics and health status. *Sleep Health* 4, 2 (2018), 160–165.
- [29] GARRIDO, S., AND SCHUBERT, E. Adaptive and maladaptive attraction to negative emotions in music. *Musicae Scientiae* 17, 2 (2013), 147–166.

- [30] GARRIDO, S., AND SCHUBERT, E. Music and people with tendencies to depression. *Music Perception: An Interdisciplinary Journal* 32, 4 (2015), 313–321.
- [31] GÓMEZ, E. Tonal description of polyphonic audio for music content processing. *INFORMS Journal on Computing* 18 (08 2006), 294–304.
- [32] HEGGLI, O. A., STUPACHER, J., AND VUUST, P. Diurnal fluctuations in musical preference. *Royal Society Open Science* 8, 11 (2021), 210885.
- [33] HEVNER, K. The affective character of the major and minor modes in music. *The American Journal of Psychology* 47, 1 (1935), 103–118.
- [34] HOLBROOK, M. B., AND ANAND, P. Effects of tempo and situational arousal on the listener’s perceptual and affective responses to music. *Psychology of Music* 18 (1990), 150 – 162.
- [35] HUBERT, L. J., AND ARABIE, P. Comparing partitions. *Journal of Classification* 2 (1985), 193–218.
- [36] ISO226:2003. Acoustics – Normal equal-loudness-level contours. Standard, International Organization for Standardization, Geneva, CH, 2003.
- [37] JESPERSEN, K., KOENIG, J., JENNUM, P., AND VUUST, P. Music for insomnia in adults. *Cochrane Database of Systematic Reviews* 8 (Aug. 2015), CD010459.
- [38] JUSLIN, P. N. Cue utilization in communication of emotion in music performance: Relating performance to perception. *Journal of Experimental Psychology: Human Perception and Performance* (2000), 1797–1813.
- [39] KIRCHBERGER, M., AND RUSSO, F. A. Dynamic range across music genres and the perception of dynamic compression in hearing-impaired listeners. *Trends in Hearing* 20 (2016), 2331216516630549. PMID: 26868955.
- [40] KLAPURI, A., AND DAVY, M. *Signal Processing Methods for Music Transcription*. Springer New York, NY, 2006.

- [41] LAROCHE, J. Efficient tempo and beat tracking in audio recordings. *Journal of the Audio Engineering Society (JAES)* 51 (04 2003), 226–.
- [42] LEGER, D., POURSAIN, B., NEUBAUER, D., AND UCHIYAMA, M. An international survey of sleeping problems in the general population. *Current medical research and opinion* 24 (01 2008), 307–17.
- [43] LERCH, A. Datasets. <https://www.ismir.net/resources/datasets/>. Accessed: 2022-07-25.
- [44] LI, T., CHAN, A., AND CHUN, A. Automatic musical pattern feature extraction using convolutional neural network. *Lecture Notes in Engineering and Computer Science* 2180 (03 2010).
- [45] LINNEMANN, A., DITZEN, B., STRAHLER, J., DOERR, J. M., AND NATER, U. M. Music listening as a means of stress reduction in daily life. *Psychoneuroendocrinology* 60 (2015), 82–90.
- [46] MEHR, S. A., SINGH, M., KNOX, D., KETTER, D. M., PICKENS-JONES, D., ATWOOD, S., LUCAS, C., JACOBY, N., EGNER, A. A., HOPKINS, E. J., HOWARD, R. M., HARTSHORNE, J. K., JENNINGS, M. V., SIMSON, J., BAINBRIDGE, C. M., PINKER, S., O'DONNELL, T. J., KRASNOW, M. M., AND GLOWACKI, L. Universality and diversity in human song. *Science* 366, 6468 (2019), eaax0868.
- [47] MORIN, C., LEBLANC, M., DALEY, M., GREGOIRE, J., AND MÉRETTE, C. Epidemiology of insomnia: Prevalence, self-help treatments, consultations, and determinants of help-seeking behaviors. *Sleep Medicine* 7, 2 (2006), 123–130.
- [48] MÜLLER, M. *Fundamentals of Music Processing*, 2nd ed. Springer Cham, 2021.
- [49] MÜLLER, M., AND ZALKOW, F. Fmp notebooks: Educational material for teaching and learning fundamentals of music processing.

- [50] NILSSON, U. Soothing music can increase oxytocin levels during bed rest after open-heart surgery: a randomised control trial. *Journal of Clinical Nursing* 18, 15 (2009), 2153–2161.
- [51] PALLESEN, S., SIVERTSEN, B., NORDHUS, I. H., AND BJORVATN, B. A 10-year trend of insomnia prevalence in the adult norwegian population. *Sleep Medicine* 15, 2 (2014), 173–179.
- [52] PEETERS, G. A large set of audio features for sound description (similarity and classification) in the cuidado project. Tech. rep., 2004.
- [53] PELLETIER, C. L. The Effect of Music on Decreasing Arousal Due to Stress: A Meta-Analysis. *Journal of Music Therapy* 41, 3 (10 2004), 192–214.
- [54] PERETZ, I., GAGNON, L., AND BOUCHARD, B. Music and emotion: perceptual determinants, immediacy, and isolation after brain damage. *Cognition* 68 (1998), 111–141.
- [55] RIGG, M. An experiment to determine how accurately college students can interpret the intended meanings of musical compositions. *Journal of Experimental Psychology* 21, 2 (1937), 223–229.
- [56] SALAMON, J., AND GOMEZ, E. Melody extraction from polyphonic music signals using pitch contour characteristics. *IEEE Transactions on Audio, Speech, and Language Processing* 20, 6 (2012), 1759–1770.
- [57] SCARRATT, R., HEGGLI, O., VUUST, P., AND JESPERSEN, K. The music that people use to sleep: universal and subgroup characteristics.
- [58] STEVENS, S. S. *Psychophysics: Introduction to its perceptual, neural, and social prospects*. Transaction Publishers, 1975.
- [59] STURM, B. L. An analysis of the gtzan music genre dataset. In *Proceedings of the Second International ACM Workshop on Music Information Retrieval with User-Centered and Multimodal Strategies* (New York, NY, USA, 2012), MIRUM '12, Association for Computing Machinery, p. 7–12.

- [60] STURM, B. L. A survey of evaluation in music genre recognition. In *Adaptive Multimedia Retrieval: Semantics, Context, and Adaptation* (Cham, 2014), A. Nürnberger, S. Stober, B. Larsen, and M. Detyniecki, Eds., Springer International Publishing, pp. 29–66.
- [61] TAN, X., YOWLER, C. J., SUPER, D. M., AND FRATIANNE, R. B. The Interplay of Preference, Familiarity and Psychophysical Properties in Defining Relaxation Music. *Journal of Music Therapy* 49, 2 (07 2012), 150–179.
- [62] TEMPERLEY, D. What’s key for key? the krumhansl-schmuckler key-finding algorithm reconsidered. *Music Perception: An Interdisciplinary Journal* 17, 1 (1999), 65–100.
- [63] TRAHAN, T., DURRANT, S. J., MÜLLENSIEFEN, D., AND WILLIAMSON, V. J. The music that helps people sleep and the reasons they believe it works: A mixed methods analysis of online survey reports. *PLOS ONE* 13, 11 (11 2018), 1–19.
- [64] TREHUB, S. E., UNYK, A. M., AND TRAINOR, L. J. Adults identify infant-directed music across cultures. *Infant Behavior and Development* 16, 2 (1993), 193–211.
- [65] TZANETAKIS, G., AND COOK, P. Multifeature audio segmentation for browsing and annotation. In *Proceedings of the 1999 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics. WASPAA’99 (Cat. No.99TH8452)* (1999), pp. 103–106.
- [66] TZANETAKIS, G., AND COOK, P. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing* 10 (08 2002), 293 – 302.
- [67] WANG, C.-F., SUN, Y., AND ZANG, H. Music therapy improves sleep quality in acute and chronic sleep disorders: a meta-analysis of 10 randomized studies. *International journal of nursing studies* 51 1 (2014), 51–62.

- [68] YAMASATO, A., KONDO, M., HOSHINO, S., KIKUCHI, J., OKINO, S., AND YAMAMOTO, K. Characteristics of music to improve the quality of sleep. *Music and Medicine* 11, 3 (2019).
- [69] YANG, J., MIN, C., MATHUR, A., AND KAWSAR, F. Sleepgan: Towards personalized sleep therapy music. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2022), pp. 966–970.
- [70] ZAPATA, J. R., DAVIES, M. E. P., AND GÓMEZ, E. Multi-feature beat tracking. 816–825.

Erklärung

Ich versichere, diese Arbeit selbstständig verfasst zu haben. Ich habe keine anderen als die angegebenen Quellen benutzt und alle wörtlich oder sinngemäß aus anderen Werken übernommene Aussagen als solche gekennzeichnet. Weder diese Arbeit noch wesentliche Teile daraus waren bisher Gegenstand eines anderen Prüfungsverfahrens. Ich habe diese Arbeit bisher weder teilweise noch vollständig veröffentlicht. Das elektronische Exemplar stimmt mit allen eingereichten Druck-Exemplaren überein.

Datum und Unterschrift:

09.10.2022 

Declaration

I hereby declare that the work presented in this thesis is entirely my own. I did not use any other sources and references than the listed ones. I have marked all direct or indirect statements from other sources contained therein as quotations. Neither this work nor significant parts of it were part of another examination procedure. I have not published this work in whole or in part before. The electronic copy is consistent with all submitted hard copies.

Date and Signature:

09.10.2022 