

Diplomarbeit

**Zur Bedeutung verlaufspezifischer Merkmale
bei Klassifikationsproblemen auf
Musiksignalen**

Ettiboa Daniel Alexis Adouakou
19. April 2016

Gutachter:

Dr. Igor Vatulkin

Prof. Dr. Günter Rudolph

Technische Universität Dortmund

Fakultät für Informatik

Algorithm Engineering (LS-11)

<http://ls11-www.cs.tu-dortmund.de>

Danksagung

An dieser Stelle möchte ich all jenen danken, die mich im Rahmen dieser Diplomarbeit begleitet haben. Ich möchte meinen Betreuern Herrn Prof. Dr. Rudolf und Herrn Dr. Vatolkin danken für die fachliche Unterstützung meiner Arbeit. Darüber hinaus möchte ich mich bei meinen Freunden bedanken, mit denen ich mich fachlich austauschen konnte und gute Anregungen für meine Arbeit erhalten habe, mit dessen Hilfe die Arbeit verbessert werden konnte.

Ein besonderer Dank gilt meiner Familie, die mich moralisch und mit viel Geduld in der Zeit des Schreibens unterstützt haben. Insbesondere zu erwähnen ist mein kleiner Sohn, der noch vor der Abgabe das Licht der Welt erblickt hat und der mich, trotz schlafloser Nächte, immer motiviert hat, weiterzumachen und dran zu bleiben.

Inhaltsverzeichnis

1	Einführung	1
1.1	Motivation und Hintergründe	1
1.2	Aufbau der Arbeit	4
2	Theoretische Grundlagen	5
2.1	Audiomerkmale	5
2.1.1	Überblick	5
2.1.2	Merkmalsgruppen	6
2.1.2.1	Zeitbereich	6
2.1.2.2	Spektralbereich	7
2.1.2.3	Cepstralbereich	7
2.1.2.4	Chroma	8
2.2	Temporale Merkmale	8
2.2.1	Überblick	8
2.2.2	Statistiken	9
2.2.3	Stacking	9
2.2.4	Autoregressive Modelle	10
2.2.4.1	Multivariates autoregressives Modell	10
2.2.4.2	Diagonal autoregressives Modell	11
2.2.4.3	Kombination der autoregressiven Modelle	11
2.3	Klassifikation	11
2.3.1	Überblick	11
2.3.2	Überwachte Lernverfahren	12
2.3.3	Vom Mehrklassen- zum Zweiklassen-Problem	13
2.3.4	Ausgewählte Methoden von überwachten Lernverfahren	13
2.3.4.1	k -Nächste-Nachbarn	13
2.3.4.2	Gaussian Mixture Model	15
2.3.4.3	Random Forest	15
2.3.4.4	Support Vector Machine	16
2.3.5	Bewertungskriterien für Klassifikationsmodelle	18

2.3.5.1	Konfusionsmatrix	18
2.3.5.2	Standard-Maße	19
3	Implementierung	21
3.1	Angewandte Tools	21
3.1.1	AMUSE	21
3.1.2	Intellij IDEA	21
3.1.3	ARFIT-Toolbox	22
3.1.4	WEKA	22
3.2	Struktur	23
3.3	Temporale Merkmalsberechnung	24
3.4	Vorgehensweise	28
4	Experimente und Auswertung	29
4.1	Daten-Korpus	29
4.1.1	Daten für die Klassifikation von Genres	29
4.1.2	Daten für die Instrumentenerkennung	30
4.2	Binäre Klassifikationsexperimente	30
4.2.1	Klassifikation von Musikgenres	31
4.2.2	Instrumentenerkennung	34
4.3	Statistische Tests	38
4.3.1	Wilcoxon-Vorzeichen-Rang-Test	39
4.3.2	Testen der Experimentergebnisse	40
4.4	Dimensionsreduktion	60
4.4.1	Reduktion anhand des multivariaten autoregressiven Modells	62
4.4.2	Reduktion anhand des diagonal autoregressiven Modells	67
5	Zusammenfassung und Ausblick	75
A	Klassifikation-Ergebnisse	77
	Abbildungsverzeichnis	91
	Tabellenverzeichnis	94
	Literaturverzeichnis	98
	Erklärung	99

Kapitel 1

Einführung

1.1 Motivation und Hintergründe

Mit zunehmendem Wachstum des Internets gibt es heute einen erhöhten Zugriff auf eine unüberschaubare Menge von Musikdaten. Dienste wie Amazon-Prime-Musik, Apple-Itunes, Spotify bieten beispielsweise Millionen von Musiktiteln an. Diese Musikdaten werden sowohl in der Cloud als auch lokal in der Musiksammlung der Benutzer gespeichert. Bei einer so großen Datenmenge ist es für den einzelnen Datennutzer oft schwierig, den Überblick über die Musikdaten zu behalten.

Seitens der Benutzer gibt es sehr viele „Abfrage-Wünsche“. Der Benutzer möchte z.B. neue Musik empfohlen bekommen, Cover-Lieder identifizieren, Plagiate erkennen, Musikstücke zu bestimmten Stimmungen finden. Um den Benutzern alle diese Möglichkeiten anbieten zu können, ist die automatische Indexierung von Musikdaten entstanden.

Die automatische Indexierung von Musikdaten versucht dem Anwender ein Werkzeug anzubieten, das ihm unterschiedliche Suchanfragen ermöglicht. Dafür werden aus den Musikdaten Audiomerkmale extrahiert. Diese repräsentieren Informationen über die Eigenschaften der originalen Daten. Anhand der Audiomerkmale können unterschiedliche Anfragen, wie zum Beispiel die Kategorisierung der Musikdaten durchgeführt werden. Die Kategorisierung kann nach Genres, Instrumenten, Künstlern, etc. erfolgen.

Audiosignaldaten können in verschiedene Kategorien (siehe Abbildung 1.1) eingeordnet werden. Die Kategorien können objektiv oder subjektiv sein. Als objektive Kategorien werden typischerweise die Stimme, die Instrumente oder der Künstler angesehen. Diese Informationen können vom Künstler als Metadaten angegeben werden. Die subjektiven Kategorien wie Emotionen, Genres können auch angegeben werden. Sie hängen aber sehr stark von der Interpretation der Künstler ab. In dieser Arbeit wird auf die Klassifizierung nach Genres und Instrumenten eingegangen. Somit wird je ein Element aus jeder Kategorie betrachtet.

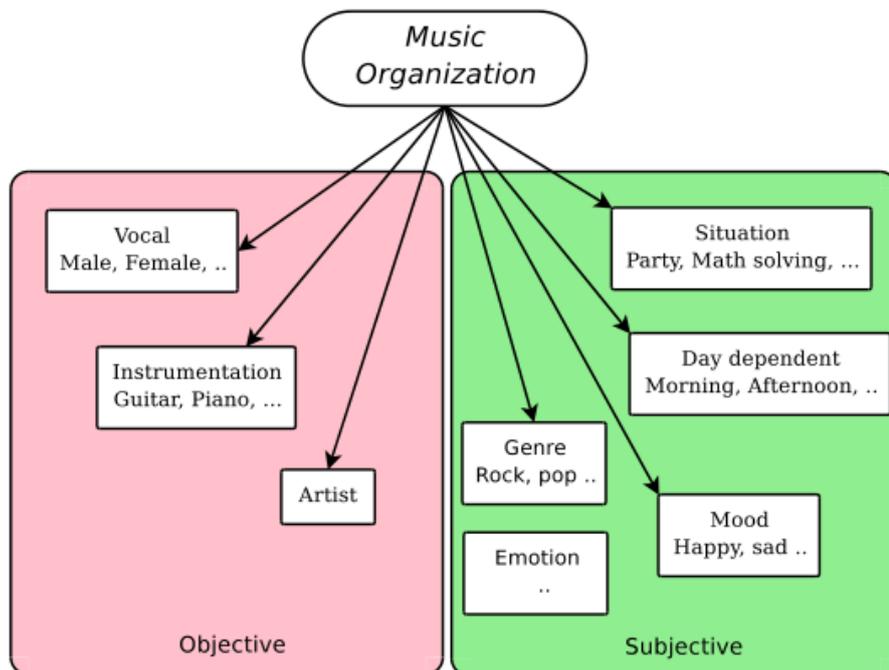


Abbildung 1.1: Ansätze der Organisation von Musik [21]

Automatische Klassifizierung

Ausgehend von Musiksignaldateien werden zunächst Audiomerkmale extrahiert. Diese Merkmale stellen verschiedene Eigenschaften der Musik dar und werden meist über eine bestimmte Zeitspanne berechnet. Basierend auf diesen Merkmalen kann dann eine automatische Klassifizierung erfolgen. Die automatische Klassifizierung versucht die richtige Kategorie oder Klasse einer Musiksignaldatei zuzuordnen. Ein typisches Klassifikationssystem besteht aus diesen zwei Schritten:

- *Die Trainingsphase:* In dieser Phase lernt das System aus einer Trainingsmenge. Es werden die Merkmale mit den passenden Klassen assoziiert.
- *Die Testphase:* Aus dem erlernten Wissen werden Vorhersagen über die Klassen-Zugehörigkeit von einer Testmenge gemacht. Es wird die Performanz des Systems evaluiert.

Diese beiden Schritte müssen mehrmals durchgeführt werden, bis man ein performantes System erhält. Die Trainingsphase beinhaltet folgende Schritte:

- Aus der Musik wird ein Musiksignal berechnet.
- Ausgehend vom Audiosignal werden die Merkmale extrahiert.
- Danach wird die Klassifikation durchgeführt.
- Zum Schluss werden die Ergebnisse ausgewertet.

Temporale Merkmale

Bei dem Ansatz der Audiomerkmale für die Klassifikation von Musiksignalen wird die Dynamik des Musikstücks meist nicht berücksichtigt. In der Studie von Ntalampiras [27] wird aber angenommen, dass die Verwendung von temporalen Merkmalen eine Verbesserung der Systemperformanz bringt.

Ein temporales Merkmal ist die Aggregation von mehreren Audiomerkmale, die über ein gleiches Zeitfenster berechnet wurden. Aus Merkmalen, die über eine kurze Zeitspanne berechnet wurden, wird ein Vektor, der diese über eine größere Zeitspanne kombiniert, berechnet.

Die Abbildung 1.2 zeigt ein Klassifikationssystem mit der Benutzung der temporalen Merkmale. Aus der Musiksignaldatei werden zuerst wie oben schon erwähnt die Audiomerkmale berechnet. Die berechneten Merkmale sind in Form von Vektoren gespeichert. Danach werden diese Merkmal-Vektoren zu einem temporalen Merkmal-Vektor kombiniert. Im nächsten Schritt wird eine Klassifikation durchgeführt.



Abbildung 1.2: Musikgenre-Klassifikationssystem

Verwandte Arbeit

In den letzten Jahren haben sich mehrere Studien mit der temporalen Aggregation von Merkmalen befasst [21, 23, 14, 27]. Die vorliegende Arbeit stützt sich auf die Studien von Meng [23] und Ntalampiras [27]. Zudem haben sich auch diese beiden Studien mit den temporalen Merkmalen befasst, und kommen auf zwei unterschiedliche Ergebnisse.

In der Studie von Meng [23] werden zwei neue Methoden für die temporale Merkmalsaggregation vorgestellt. Die Methoden basieren auf autoregressiven Modellen und versuchen die Klassifikationsprobleme von Musikgenres zu lösen. Die neuen Aggregationsmethoden sind das Multivariate Autoregressive Model (MAR) und das Diagonal Autoregressive Model (DAR). Beide Modelle werden mit anderen temporalen Methoden wie „Einfache Statistiken“ (Durchschnitt und Varianz (MeanVar), Durchschnitt und Kovarianz (MeanCov)) und „Filter Bank Coefficients (FC)“ verglichen. Die Ergebnisse von Meng zeigen, dass MAR besser als die anderen Aggregationsmethoden funktioniert. Auch DAR funktioniert besser als FC und MeanVar auf größeren Daten und hat eine kleinere Dimension als MAR.

In der Studie von Ntalampiras [27] wird eine allgemeine Methodologie zur Verwendung von temporalen Merkmalsaggregationen bei automatischen Audioerkennungproble-

men dargestellt. Es werden drei Aggregationsmethoden („Statistiken“, MAR, DAR) auf drei unterschiedliche Merkmalsgruppen („Mel filterbank“, „MPEG-7 audio spectrum projection“, „PWP integration analysis“) angewandt. Die Ergebnisse von Ntalampiras zeigen für alle Merkmalsgruppen eine bessere Erkennungsrate bei der Nutzung von Statistiken.

Die Ergebnisse dieser beiden Studien zeigen, dass die unterschiedlichen Aggregationsmethoden sich als vorteilhafter herausstellen. Aus diesem Grund wird in dieser Arbeit ein System für die Untersuchung der Bedeutung der temporalen Merkmale bei Klassifikationsproblemen entwickelt. Das System soll helfen, folgende Fragestellung zu beantworten:

- Welche temporalen Merkmale sind am besten geeignet für welches Problem?
- Wie ist die Auswirkung der Audiomerkmale auf die Performanz der temporalen Merkmale?
- Bringen die temporalen Merkmale Verbesserungen in der Performanz der Klassifikationssysteme?

1.2 Aufbau der Arbeit

Diese Arbeit wird wie folgt gegliedert:

Kapitel 1 beinhaltet die Einleitung und benennt eine erste Gliederung.

Kapitel 2 beschäftigt sich mit den theoretischen Grundlagen. Diese Grundlagen umfassen die Merkmalsextraktion, die Aggregationsmethoden und die Klassifikationsverfahren. Zudem wird auch auf die Qualitätskriterien der Klassifikationen eingegangen.

Kapitel 3 beschreibt die Implementierung. Es werden die verwendeten Tools und die konkrete Implementierung der temporalen Merkmalsberechnungen behandelt.

Kapitel 4 widmet sich der Untersuchung der temporalen Merkmale. Dort werden die aufgesetzten Experimente behandelt. Die Ergebnisse dieser Experimente werden analysiert und Hypothesen werden erstellt. Zudem werden diese Hypothesen anhand von statistischen Tests überprüft.

Kapitel 5 enthält die Zusammenfassung dieser Arbeit. Es wird auch auf Ausblicke auf mögliche Weiterentwicklungen eingegangen.

Anhang A stellt die Ergebnisse der durchgeführten Experimente zusammen.

Kapitel 2

Theoretische Grundlagen

Dieses Kapitel widmet sich den theoretischen Grundlagen. Die hier eingeführten Begriffe und Definitionen sind für den weiteren Teil dieser Arbeit von Relevanz.

2.1 Audiomerkmale

Die Merkmalsextraktion ist der erste Schritt zu jeder Klassifikation von Musiksignalen. In dieser Sektion werden alle Merkmale, die in dieser Arbeit benutzt wurden, beschrieben.

2.1.1 Überblick

Das Ziel der Merkmalsextraktion ist es, Merkmale aus den Datensätzen zu extrahieren. Diese Merkmale repräsentieren Informationen über die Eigenschaften der originalen Daten. In dieser Arbeit werden Musiksignalen betrachtet. Die Berechnung der Merkmale erfolgt über Zeitspannen von kleiner Größe. Diese Merkmale bilden wichtige Aspekte der Musik, wie zum Beispiel die Harmonizität, die Tonhöhe, die Instrumentierung oder den Rhythmus, nach.

Eine Musiksignaldatei an sich ist sehr groß und meistens ungeeignet für Analyse-Aufgaben. Ein typisches Musikstück ist 4 Minuten lang [21]. Die meisten Musikdaten werden mit einer Frequenz von 44100 Hz gesampelt, was etwa 21 Millionen Samples pro Lied beträgt (für ein Stereo-Signal). Die Merkmalsextraktion bereitet Daten vor, die einfacher zu analysieren sind. Daher kann sie als Reduktionsverfahren angesehen werden.

In den meisten Audioanalysen wird das Signal in kleinere Fenster unterteilt. Für jedes Fenster wird dann eine Gruppe von Merkmalen berechnet. Daraus resultiert dann eine Sequenz von Merkmals-Vektoren für jedes Audiosignal. Hierbei hängt die Dimension der Merkmale sehr stark von ihrem Typ ab.

Gewöhnlich gibt es viele Merkmale der Dimension 1, wie zum Beispiel die Energie des Signals. Bei komplexeren Merkmalen werden mehrere Merkmale extrahiert und zu einem Vektor von einer größeren Dimension kombiniert.

2.1.2 Merkmalsgruppen

In der vorliegenden Arbeit finden Merkmale aus vier Gruppen Anwendung. Diese Merkmale stammen aus den Merkmalsgruppen, die in der Studie von Vatolkin [34] verwendet wurden. Die Merkmalsgruppen sind in der Tabelle 2.1 aufgelistet. Alle diese Merkmale werden über ein Extraktionsfenster von 23 *ms* berechnet.

Tabelle 2.1: Merkmale nach Merkmalsgruppen

Merkmalsgruppe	Merkmale
ZEITBEREICH	Zero-Crossing Rate, Low Energy, Root Mean Square
SPECTRALBEREICH	Tristimulus, Spectral Kurtosis, Spectral Discrepancy
CEPSTRALBEREICH	Mel Frequency Cepstral Coefficient (MFCC)
CHROMA	Chroma

Im Folgenden werden diese Gruppen definiert und kurz beschrieben, sowie die Definitionen der gewählten Merkmale aus den jeweiligen Gruppen gegeben. Eine detaillierte Beschreibung der Merkmale ist in der Studie [33] zu finden.

2.1.2.1 Zeitbereich

Die zeitbasierten Merkmale charakterisieren die Zeitreihen des Audiosignals, zum Beispiel durch eine Annäherung mit der „Linear Prediction Coefficient“ oder Energieverteilung [34]. Beispielsweise vergleicht das Merkmal „Low Energy“ die Energie eines Fensters mit der Energie des vorherigen größeren Extraktionsfensters.

Zero-Crossing Rate

Die Zero-Crossing Rate ist ein Maß für den Hochfrequenzanteil eines Signals. Es beschreibt die Häufigkeit der Vorzeichenwechsel eines Signals. Die Zero-Crossing Rate korreliert mit der Stärke des Signals, die wiederum seine Klangfarbe beschreibt [11, S. 73].

Low Energy

Das Merkmal „Low Energy“ ist der Anteil der Extraktionsfenster in einem Texturfenster, die eine kleinere Signalenergie als die durchschnittliche Signalenergie enthalten. Es beschreibt also den Anteil an Extraktionsfenstern mit niedrigem Energiegehalt. Das Texturfenster besteht aus mehreren Extraktionsfenstern mit verschiedenen Kurzzeitspektren, die ein Klangmuster (genannt „Musik-Textur“) bilden.

Root mean square

Das Merkmal „Root mean square“ (RMS) stellt ein normiertes Maß für die Signalenergie in einem Zeitfenster dar. Dieses beschreibt die Lautstärke eines Musikstücks anhand der Amplituden.

2.1.2.2 Spektralbereich

Die spektralbasierten Eigenschaften werden auf Basis des Spektrums des Signals berechnet. Diese Merkmale entsprechen zahlreichen Statistiken über das Spektrum: Schwerpunktwellenlänge, Crest-Faktor, Neigung, Symmetrie, Flussmittel, Schiefe, Abstände zwischen spektralen Spitzen, usw.

Tristimulus

Das Merkmal „Tristimulus“ besteht aus drei Werten und wurde als Klang-Äquivalent für die Farbattribute im Licht eingeführt. Es bildet drei Arten von Energieverhältnissen ab und ermöglicht eine Beschreibung der ersten harmonischen Eigenschaften eines Signals. Der „Tristimulus“ wurde erstmals in der Studie von Pollard [29] vorgestellt.

Spectral Kurtosis

Das Merkmal „Spectral Kurtosis“ gibt ein Maß der Ebenheit einer Verteilung um seinen Mittelwert. Es ist aus dem vierten Ordnungsmoment berechnet.

Spectral Discrepancy

Das Merkmal „Spectral Discrepancy“ ist die normalisierte Summe aller spektralen Amplitudenabweichungen von einer linearen Regressionslinie.

2.1.2.3 Cepstralbereich

Die cepstralbasierten Merkmale bestehen aus den verschiedenen Implementierungen der „Mel Frequency Cepstral Coefficient“-Merkmale (MFCCs) und der „Cepstral Modulation Ratio Regression“-Merkmale (CMRARE). Die CMRARE-Merkmale beschreiben die zeitlichen Cepstrum-Fortschritte mit einer polynomialen Annäherung [20].

Mel Frequency Cepstral Coefficient

Dieses Merkmal wurde ursprünglich für die automatische Spracherkennung entwickelt zur Entkopplung des Stimmanregungssignals aus der Vokaltrakt-Form [7]. Es findet aber Anwendung in anderen Bereichen wie dem auditiven Lernen und dem „Music Information Retrieval“.



Abbildung 2.1: Das MFCC Extraktions-Schema [21]. Die Zahlen über jedem Schritt stellen die Dimension der Kompression in dem jeweiligen Schritt dar.

Das ursprüngliche Verfahren für die Extraktion von MFCCs ist in Abbildung 2.1 dargestellt, wobei die Zahlen über den verschiedenen Schritten eine intuitive Idee der Dimension geben. Die Leistungen (oder Amplituden) aller Frequenzkomponenten in kritischen Bändern des menschlichen auditiven Systems werden unter Verwendung der Mel-Skala summiert. Anschließend wird der Ton im Frequenzbereich um eine Kurzzeit-Fourier-Transformation verändert. Der Ausgang der Filterbank ist logarithmisch gewichtet und wendet schließlich ein „Discrete Cosine Transform“ (DCT) zu dekorrelieren.

2.1.2.4 Chroma

Chroma-Vektor

Der Chroma-Vektor fasst alle spektralen Komponenten der gleichen Tonhöhe-Klasse zusammen [11, S. 91]. Seine 12 Werte representieren die zwölf Töne einer Oktave. er wird beispielsweise in der Studie von Fujishima [10] verwendet.

2.2 Temporale Merkmale

2.2.1 Überblick

Meng definiert in [21] die temporale Merkmalsaggregation als einen Prozess, der alle Merkmalsvektoren aus einem Texturfenster zu einem einzigen Merkmalsvektor kombiniert. Dieser Aggregationsvektor erfasst die impliziten zeitlichen Informationen des Texturfensters, die für einen späteren Lernalgorithmus nützlich sind. Allerdings erfasst diese Aggregation nicht unbedingt Wahrnehmungsaspekte wie Tempo oder Stimmung der Musik.

In diesem Abschnitt werden die in dieser Arbeit verwendeten Aggregationsmethoden vorgestellt. Zudem werden zwei Typen von Fenstern verwendet: ein Extraktionsfenster und ein Texturfenster. Als Extraktionsfenster wird die kurze Zeitspanne, meist zwischen 20 und 30 ms [24], wo ein Wert für die Merkmale berechnet wird, bezeichnet. Das Texturfenster besteht aus mehreren Extraktionsfenstern und stellt die Zeitspanne für die Aggregation der Merkmale dar.

Sei d die Anzahl der Audiomerkmale, $\{x_i[k]\}_{i=1\dots d}$ ihre Werte für das k -te Extraktionsfenster, und der Vektor aus diesen Werten $x[k] = [x_1[k], \dots, x_d[k]]$. Weiterhin seien

$k_t, \dots, k_t + l$ die Indizes der Extraktionsfenster für ein Texturfenster t von Länge l . Der aggregierte Vektor für dieses Texturfenster wird X_t genannt. Jede Aggregationsmethode kann dann als Funktion gesehen werden [27], die versucht eine vordefinierte Anzahl an Extraktionsfenstern zu aggregieren.

2.2.2 Statistiken

Die Aggregationsmethode „Statistiken“ ist eine relativ einfache Methode, um Merkmale zu aggregieren. Mit dieser Methode werden einige statistische Werte für diese Merkmale über das Texturfenster berechnet. Diese Methode wurde in anderen Arbeiten wie [27] und [26] eingesetzt.

Für je eine Sequenz von Werten eines Audiomerkmals werden folgende statistische Werte berechnet: der Durchschnittswert, die Varianz, das erste Quartil, der Median und das dritte Quartil. Diese Werte bilden dann eine neue Sequenz, die in den aggregierten Vektor aufgenommen wird. Für ein Merkmal x_i wird der Durchschnittswert $\mu_{i,t}$ für ein Texturfenster t wie folgt berechnet:

$$\mu_{i,t} = \frac{1}{l} \sum_{k=k_t}^{k_t+l} x_i[k] \quad (2.1)$$

Die Varianz $\sigma_{i,t}^2$ eines Merkmals x_i für ein Texturfenster t wird wie folgt berechnet:

$$\sigma_{i,t}^2 = \frac{1}{l} \sum_{k=k_t}^{k_t+l} (x_i[k] - \mu_{i,t})^2 \quad (2.2)$$

Seien $q1_{i,t}$, $q2_{i,t}$, $q3_{i,t}$ respektive das erste Quartil, der Median und das dritte Quartil eines Merkmals x_i für ein Texturfenster t . Die Funktion für die Aggregationsmethode „Statistiken“ kann wie folgt definiert werden:

$$F_{stats}(x_i[k_t], \dots, x_i[k_t + p]) = [\mu_{i,t}, \sigma_{i,t}^2, q1_{i,t}, q2_{i,t}, q3_{i,t}] \quad (2.3)$$

Der aggregierte Vektor für die Methode „Statistiken“ über dieses Texturfenster ist dann gegeben durch:

$$X_t = [F_{stats}^{1,t}, \dots, F_{stats}^{d,t}] \quad (2.4)$$

2.2.3 Stacking

Diese Methode wurde in anderen Arbeiten wie in [21] und [32] eingesetzt. Hier werden die Merkmalsvektoren zu einem einzigen Vektor gestapelt. Der neue Vektor erfasst die temporalen Informationen in diesem Fenster. Die Funktion für „Stacking“ ist folgende:

$$F_{stack}(x_i[k_t], \dots, x_i[k_t + p]) = [x_i[k_t], \dots, x_i[k_t + p]] \quad (2.5)$$

Der aggregierte Vektor für die Methode „Stacking“ über dieses Texturfenster ist dann gegeben durch:

$$X_t = [F_{stack}^{1,t}, \dots, F_{stack}^{d,t}] \quad (2.6)$$

Die Daten werden hier nicht komprimiert. Es ergeben sich hier also oft sehr große Dimensionen. Deshalb benötigt diese Aggregationsmethode einen sehr robusten Klassifikationsalgorithmus.

2.2.4 Autoregressive Modelle

Die autoregressiven Modelle sind sehr verbreitet im ökonomischen Bereich und werden meist für die Analyse von Zeitreihen verwendet. Die Grundannahme der Zeitreihenanalyse besagt, daß es sich bei einer Zeitreihe um die Realisierung eines stochastischen Prozesses handelt, dessen Zufallsvariablen stochastisch voneinander abhängen [28, S. 193]. In der Arbeit von Lütkepohl [19] wird ein guter Überblick über diese Modelle gegeben. Sie wurden für die Klassifikation von Musikstücken erstmals in der Studie von Meng [22] angewandt.

2.2.4.1 Multivariates autoregressives Modell

Diese Modelle können den zeitlichen Verlauf gut einfangen. Die Parameter der Modelle werden durch den Algorithmus der kleinsten Quadrate berechnet. Die berechneten Koeffizienten werden zu einem aggregierten Vektor zusammengefasst.

Für die Berechnung der Koeffizienten eines Modells von Rang O wird die folgende Formel verwendet:

$$x[t] = w + \sum_{n=1}^O x[t-n]A_n + e_t \quad (2.7)$$

wobei:

- w der Abfangvektor ist,
- A_n $D \times D$ Koeffizienten-Matrizen der autoregressiven Modelle sind: Sie kodieren wie viele von den vorherigen Informationen im aktuellen Vektor enthalten sind.
- e_t ein weißes Rauschen von Dimension D ist.

Die Koeffizienten-Matrizen werden hier komplett verwendet. Sei d die Anzahl der Audiomerkmale. Der aggregierte Vektor wird wie folgt zusammengesetzt:

$$X_t = [w, A_1(1, 1), \dots, A_1(d, 1), A_O(1, 1), \dots, A_O(d, d)] \quad (2.8)$$

MAR vergrößert den Merkmalsvektor sehr stark. Die Dimension des MAR-Modells ist $D(O \times D + 1)$, also beispielsweise 182 für den Rang 1 mit 13 Merkmalen.

2.2.4.2 Diagonal autoregressives Modell

Hier wird angenommen, dass die Merkmale voneinander unabhängig sind. Deshalb wird nur die Diagonale der Koeffizienten-Matrizen verwendet. Für die Berechnung der Koeffizienten eines Modells von Rang O wird für ein Merkmal x_i die folgende Formel verwendet:

$$x_i[k] = w_i + \sum_{n=1}^O x_i[k-n]a_n^i + e_k \quad (2.9)$$

Der aggregierte Vektor wird wie folgt zusammengesetzt:

$$X_t = [w_1, a_1^1, \dots, a_O^1, \dots, w_d, a_1^d, a_O^d] \quad (2.10)$$

Eine komplette Abhandlung der autoregressiven Modelle ist in [25] und [18] gegeben.

2.2.4.3 Kombination der autoregressiven Modelle

Bei dieser Methode werden die berechneten Vektoren für MAR und DAR zu einem neuen Vektor gestapelt. Der aggregierte Vektor wird wie folgt zusammengesetzt:

$$X_t = [w, A_1(1, 1), \dots, A_1(d, 1), A_O(1, 1), \dots, A_O(d, d), w_1, a_1^1, \dots, a_O^1, \dots, w_d, a_1^d, a_O^d] \quad (2.11)$$

Das neue Modell hat eine größere Dimension als die beiden anderen Modelle und könnte eventuell eine Verbesserung bringen. Mit der Kombination der beiden Modelle wird untersucht, ob man dadurch bessere Klassifikationsergebnisse erzielen kann.

2.3 Klassifikation

2.3.1 Überblick

Klassifikationsverfahren sind Verfahren aus den Bereichen der Mathematik, Statistik und Informatik, die dazu dienen, Kategorien von Dateninstanzen zu entdecken oder die Funktion der Zuordnung in vorgegebene Kategorien zu erlernen. Im ersten Fall spricht man von unüberwachtem, im zweiten von überwachtem Lernen [8, S. 16-17].

Beim unüberwachten Lernen wird hauptsächlich die Ähnlichkeit der Dateninstanzen untersucht und darauf basierend werden Kategorien gebildet. Unter anderem werden Methoden wie K-Means, EM (Expectation Maximisation) Clustering und „Divisive Analysis Clustering“ (DIANA) eingesetzt.

Beim überwachten Lernen werden vielmehr die Abhängigkeiten zwischen den vorgegebenen Kategorien und den Dateninstanzen untersucht, um darauf basierend eine Funktion für die Zuordnung in die jeweiligen Kategorien abzuleiten. Dies wird mit Verfahren wie Regressionsanalysen (Statistik), „Linear Programming“ (Mathematik) und künstlichen neuronalen Netzen (Informatik) durchgeführt.

Ein umfassender Überblick über die am häufigsten verwendeten Klassifikationsmethoden kann in den Studien [8], [1] und [3] nachgelesen werden.

Da in dieser Studie die Kategorien für die Klassifikation von Musikstücken vorgegeben sind, ist hier das überwachte Lernen relevant. Zudem wird im Abschnitt 2.3.2 näher auf überwachte Lernverfahren eingegangen. Weiterhin werden im Abschnitt 2.3.4 die für diese Studie ausgewählten Lernverfahren dargestellt.

2.3.2 Überwachte Lernverfahren

Sei $x_i = (x_{i1}, x_{i2}, \dots, x_{iP})$ die Dateninstanz i und $y_i \in (1, 2, \dots, M)$ die dazugehörige Kategorie, P die Gesamtanzahl der Variablen und M die Gesamtanzahl der Kategorien. Das Ziel des überwachten Lernens ist eine Abbildung f zu finden, die zu jeder x_i ein $f(x_i)$ assoziiert, sodass die Differenz zwischen $f(x_i)$ und y_i minimal ist. Je nach Herkunft des Lernverfahrens wird die Abbildung f unterschiedlich ermittelt.

Die meisten Lernverfahren aus der Statistik (parametrische Verfahren) betrachten die Kategorien y_i als Realisierung einer Zufallsvariablen Y_i und nehmen entsprechend eine Verteilung für die Dateninstanzen in den jeweiligen Kategorien auf. Die Abbildung f wird dann so bestimmt, dass die Auftrittswahrscheinlichkeit für jede einzelne Dateninstanz maximal ist. Letzteres ist auch als Maximum-Likelihood-Prinzip bekannt [8, S. 85].

Die Verfahren aus der Mathematik nutzen deterministische Ansätze um die Abbildung f zu ermitteln. Dabei wird eine Zielfunktion definiert unter Berücksichtigung von topologischen Gesichtspunkten und es werden Nebenbedingungen aufgestellt. Zudem wird die Zielfunktion unter Berücksichtigung der Nebenbedingungen optimiert.

Bei den Verfahren aus der Informatik wird eine Zielfunktion definiert, zumeist eine Verlustfunktion. Algorithmen werden dann umgesetzt um die Zielfunktion zu optimieren. Meistens bestehen die Algorithmen aus Kombinationen von weiteren Algorithmen, statistischen und/oder mathematischen Verfahren.

Um sicherzustellen, dass die auf Basis der vorliegenden Dateninstanzen ermittelte Funktion allgemein gilt, wird der Entwicklungsprozess in zwei Schritten durchgeführt. Im ersten Schritt werden die Parameter der Abbildung f ermittelt. Dieser Schritt wird Trainingsphase genannt. Im zweiten Schritt wird geprüft, ob die ermittelte Abbildung verallgemeinerungsfähig ist. Dieser Schritt wird Testphase genannt.

Die Gestaltung der Trainings- und Testphase hängt von der Datenlage ab. Bei ausreichender Datenlage wird der Ausgangssatz in zwei Datensätze geteilt. Bei nicht ausreichender Datenlage wird der ganze Ausgangssatz in der Trainingsphase verwendet. Die Testphase wird dann mit Methoden wie der Kreuzvalidierung durchgeführt. Bei der Kreuzvalidierung, beispielsweise, wird eine Dateninstanz unberücksichtigt gelassen und mit den restlichen Dateninstanzen die Trainingsphase durchgeführt. Dann wird die

ermittelte Abbildung an der unberücksichtigt gelassenen Dateninstanz getestet. Dies kann der Reihe nach für alle Dateninstanzen durchgeführt werden [1, S. 351].

In dieser Studie stehen ausreichend Daten zur Verfügung, sodass wir unseren Ausgangsdatensatz in Training- und Testdatensatz unterteilen können. Welche Kriterien für die Überprüfung der Verallgemeinerungsfähigkeit gewählt wurden, wird in Abschnitt aufgeführt. Zuvor wird im Abschnitt 2.3.3 dargestellt, wie das Mehrklassen-Problem in ein Zweiklassen-Problem reduziert werden kann.

2.3.3 Vom Mehrklassen- zum Zweiklassen-Problem

Im Vergleich zu Mehrklassen-Problemen sind Zweiklassen-Probleme einfacher zu behandeln und entsprechend stehen mehr Methoden zur Behandlung von Zweiklassen-Problemen, auch binäre Klassen-Probleme genannt, zur Verfügung [36, S. 338]. Daher wurde in dieser Studie das Mehrklassen-Problem in ein binäres Klassen-Problem umgewandelt. Hierfür gibt es zwei Hauptansätze: „One-vs-All“ und „One-vs-One“ [3, S. 182-183].

- **One-vs-All (OVA):** Bei OVA, zu Deutsch „eins-gegen-alle“, werden zwei Klassen gebildet. Die erste Klasse enthält lediglich eine Kategorie und die zweite Klasse die restlichen Kategorien.
- **One-vs-One (OVO):** Bei OVO, zu Deutsch „eins-gegen-eins“, werden die Kategorien paarweise klassifiziert. D.h. es werden Modelle für jedes Paar von Kategorien ermittelt und die Ergebnisse für die jeweiligen Kategorien am Ende aggregiert (z. B. Modus über alle betroffenen Dateninstanzen).

2.3.4 Ausgewählte Methoden von überwachten Lernverfahren

Diese Studie wurde breit aufgestellt und entsprechend wurden Methoden aus drei unterschiedlichen Disziplinen (Statistik, Mathematik und Informatik) ausgewählt. Als Einstieg wurde ein einfacher Modellansatz aus der Informatik verwendet, um einen schnellen Einblick in die Daten verschaffen zu können, und zwar der „ k -Nächste-Nachbarn“ (Abschnitt 2.3.4.1). Einen etwas komplizierteren Modellansatz aus derselben Disziplinen stellt das Random forest (Abschnitt 2.3.4.3) dar. Aus der Statistik wurde der parametrische Modellansatz „Gaussian Mixture Model“ gewählt, und aus der Mathematik der „Support Vector Machine“.

2.3.4.1 k -Nächste-Nachbarn

Beim „ k -Nächste-Nachbarn“-Klassifikator geht es darum, ähnliche Dateninstanzen wie die zu klassifizierende Dateninstanz zu finden und eine entsprechende Klassen-Zugehörigkeit zu übernehmen. Die Hauptaufgabe beim „ k -Nächste-Nachbarn“ liegt daher darin, zu be-

stimmen, wie viele Dateninstanzen, auch „nächste Nachbarn“ genannt, herangezogen werden sollen und welches Ähnlichkeitsmaß benutzt wird.

Die Anzahl der „nächsten Nachbarn“ soll so bestimmt werden, dass das Modell spezifisch genug ist, um eine genaue Vorhersage machen zu können. Gleichzeitig muss eine Überanpassung des Modells an die Daten durch zu spezifische Modellvorgaben vermieden werden. Die Abbildung 2.2 zeigt ein Beispiel für die Wichtigkeit der Anzahl der nächsten Nachbarn.

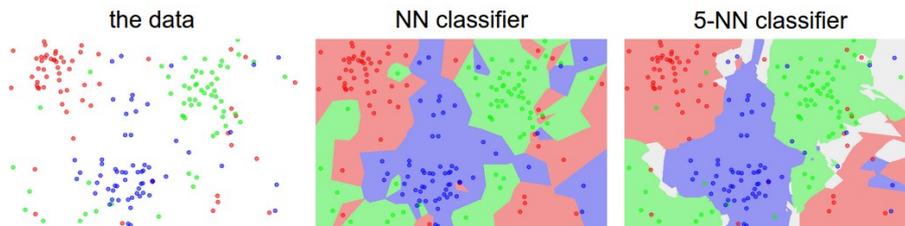


Abbildung 2.2: Ein Beispiel für den Unterschied zwischen einem Nächster-Nachbar- und einem 5-Nächste-Nachbarn-Klassifikator unter Verwendung von 2-dimensionalen Punkten und 3 Klassen (rot, blau, grün) [15]

Wenn alle Elemente der Dateninstanzen numerische Merkmale sind, lässt sich eine Distanzmatrix als Ähnlichkeitsmaß nutzen. Die Methode zur Bestimmung der Distanzen zwischen den Dateninstanzen hängt von der Datenlage ab. Bei einer geringen Korrelation zwischen den Merkmalen lässt sich der Euklidische Abstand gut verwenden. Sei $d(x_1, x_2)$ der Euklidische Abstand zwischen den D -dimensionalen Dateninstanzen x_1 und x_2 ,

$$d(x_1, x_2) = \sqrt{\sum_{j=1}^D (x_1(j) - x_2(j))^2}. \quad (2.12)$$

Diese Art der Distanzberechnung geht von einem D -dimensionalen Raum aus, gespannt mit D orthogonalen Achsenbasen, sodass die Distanz $d(x_1, x_2)$ als Hypotenuse eines entsprechenden Dreiecks mit Anwendung des Pythagoras-Theorems berechnet werden kann. Wenn aber die Merkmale stark korrelieren, gilt die orthogonale Achsenbasis nicht mehr, sodass das Pythagoras-Theorem nicht verwendet werden kann. In diesem Fall kann alternativ der Mahalanobis-Abstand verwendet werden. Letzterer berücksichtigt die Kovarianzmatrix der Merkmale bei der Bestimmung der Distanz.

Sobald die „nächsten Nachbarn“ bestimmt sind, lässt sich die Klasse der zu klassifizierenden Dateninstanz durch Übernahme der Klasse, die in dieser direkten Nachbarschaft am meisten vertreten ist, bestimmen. Weitere Details zum k -Nächste-Nachbarn-Klassifikator ist in [9, S. 463-475] zu finden.

Der k -Nächste-Nachbarn-Klassifikator ist einfach und trotzdem gut geeignet sowohl für binäre als auch für mehrklassige Klassifikationsprobleme. Einer seiner großen Vorteile ist, dass keine Trainingsphase benötigt wird. Die Trainingsdaten werden direkt bei der

Klassifikation verwendet. Zudem ist er intuitiv und relativ einfach zu berechnen [1, S. 167].

2.3.4.2 Gaussian Mixture Model

Das „Gaussian Mixture Model“ (GMM) ist ein statistisches Modell. Statistische Modelle sind sehr verbreitet im Bereich der Signalverarbeitung. Der GMM basiert auf der Annahme, dass die zu klassifizierenden Dateninstanzen aus einer Grundgesamtheit stammen. Diese besitzt mehrere Teilmengen, die einzeln modelliert werden. Die Mischung der Modelle der Teilmengen repräsentiert die ursprüngliche Grundgesamtheit, und wird zu einem einzelnen Modell zusammengefasst. Dieses Modell wird GMM genannt.

Nach Reynolds [30] ist ein GMM eine parametrische Wahrscheinlichkeitsdichtefunktion, die als eine gewichtete Summe von gaußschen Komponentendichten dargestellt wird. Sie besteht aus drei Parametern: der Variablen, dem Mittelwert und der Kovarianzmatrix. Diese drei Parameter müssen abgeschätzt werden. Es gibt verschiedene Ansätze, um die Parameter einer GMM zu berechnen. Die am häufigsten verwendete Methode ist der EM-Algorithmus.

EM-Algorithmus Das Ziel ist hier die Modell-Parameter zu finden, die die Zuordnung der Daten maximieren.

Die Funktionsweise des EM-Algorithmus ist folgende:

- Mit dem Initial-Modell λ wird das neue Modell $\bar{\lambda}$ berechnet, sodass $p(X|\bar{\lambda}) \geq p(X|\lambda)$.
- Das neue Modell wird zum Initial-Modell in der nächsten Iteration. Die Wiederholung wird gemacht, bis es keine wesentliche Veränderung mehr gibt.

2.3.4.3 Random Forest

„Random Forest“ (RF) ist ein Klassifikationsverfahren, das sowohl für die Klassifikation als auch für die Regression von Daten angewandt wird. Es wurde zum ersten Mal 2001 von Breimans in [5] eingeführt. Ein „Random Forest“ besteht aus vielen Entscheidungsbäumen, daher ein „Forest“. Da alle Entscheidungsbäume zufällig generiert werden, ist es ein „Random Forest“. Meistens wird der Algorithmus „Classification And Regression Trees“ (CART) als Methode für die Generierung der Entscheidungsbäume für „Random Forest“ benutzt.

Das Prinzip von CART besteht darin, die Dateninstanzen zunächst rekursiv binär zu teilen, um dann eine optimale Teilmenge für die Entscheidung zu ermitteln. Zu jedem Schritt der Partitionierung werden die Dateninstanzen in zwei Teilmengen unterteilt. Ein binärer Baum wird aus dieser Teilung abgeleitet. Die Knoten des Baums werden zu den Elemente der Teilung assoziiert. Beispielsweise ist die Wurzel des Baums mit den gesamten

Dateninstanzen assoziiert. Die nächsten zwei Knoten sind dann mit den zwei Teilmengen nach der ersten Spaltung assoziiert, und so weiter. Eine Darstellung der Teilung und des dazugehörigen Baums wird in der Abbildung 2.3 gezeigt. Eine detaillierte Beschreibung von CART kann in [37, S. 179-197] nachgelesen werden.

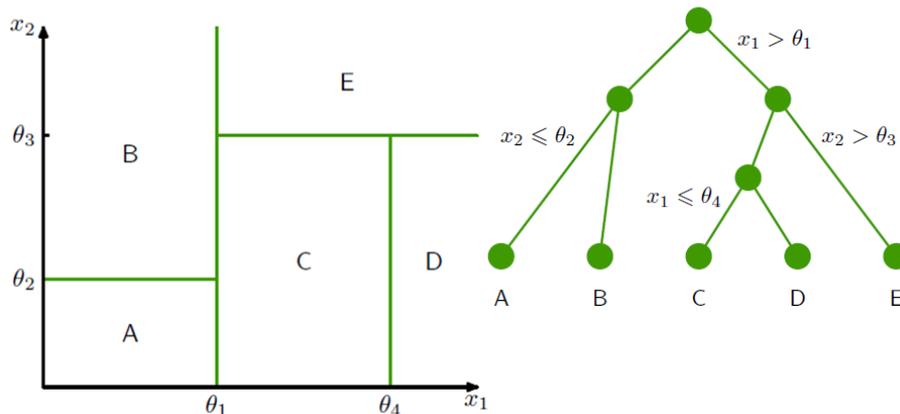


Abbildung 2.3: Die Teilung eines Quadrats und der abgeleitete CART-Baum [3]

„Random Forest“ benötigt nicht die ganzen Dateninstanzen für die Generierung der Bäume. Stattdessen werden Bootstrap-Lernstichproben verwendet. Danach wird für je eine Bootstrap-Stichprobe ein CART-Baum konstruiert. Zur Klassifikation von neuen Daten werden die Entscheidungen aller konstruierten Bäume aggregiert. Die Klassifikation des „Random Forest“ ist die Kategorie, welche am häufigsten gewählt wurde. Eine detaillierte Erläuterung von „Random Forest“ ist in [17] zu finden.

2.3.4.4 Support Vector Machine

Eine „Support Vector Machine“ ist ein Klassifikator, der aus dem Bereich der Mathematik kommt. Er gehört zu der Familie der Kernel-Methoden und ist der meist verwendete Klassifikator aus dieser Familie. Zudem kann er meist nur zwischen zwei Klassen unterscheiden [3].

Lineare SVM Das Prinzip der „Support Vector Machine“ ist es, zwei Klassen durch eine Hyperebene optimal zu unterteilen [6]. Sei $\{x_i\}_{i=1..l}$ eine Menge in \mathbb{R} von l Vektoren und d Merkmalen, die zu zwei unterschiedlichen Kategorien Ω_1 und Ω_{-1} gehören. Wir nennen $\{y_i\}_{i=1..l}$ in $\{-1, +1\}$ das Label der Kategorie.

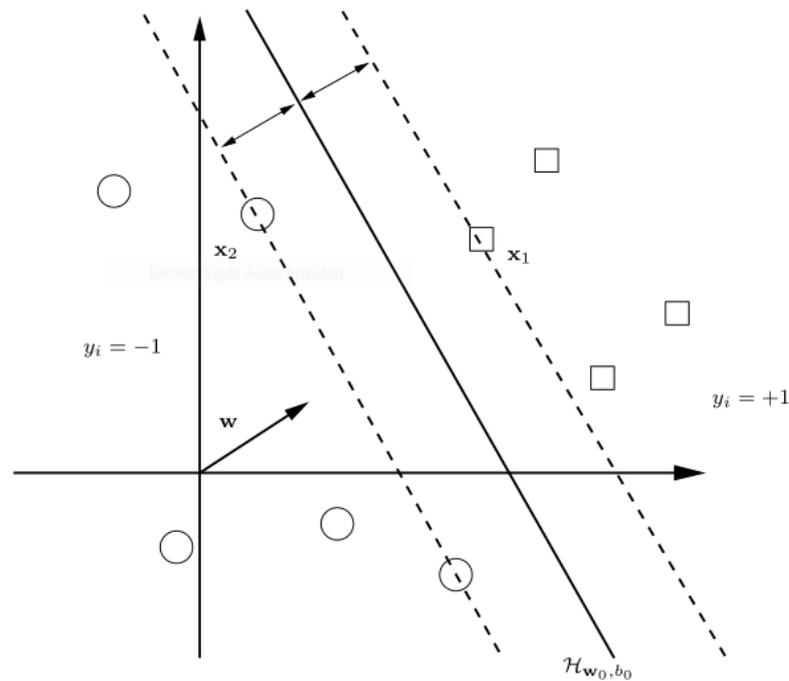


Abbildung 2.4: Optimale Hyperebene und Margin eines SVM Klassifikators

Wenn die Daten linear trennbar sind, können wir eine Hyperebene H definieren. Die Definition von H ist folgende:

$$H : w \cdot x + b = 0, w \in \mathbb{R}^d, b \in \mathbb{R} \quad (2.13)$$

so dass gilt:

$$w \cdot x_i + b \geq +1 \text{ für } y_i = +1,$$

$$w \cdot x_i + b \leq -1 \text{ für } y_i = -1$$

Die Gleichung kann zu

$$y_i(w \cdot x_i + b) \geq +1 \quad (2.14)$$

umformuliert werden.

Der Abstand von der Trennebene zu den Instanzen, die zu ihr auf jeder Seite am nächsten liegen, bezeichnet man auch als „Margin“ [1]. Die gesuchte optimale Hyperebene ist die, die diesen Abstand maximiert. Die Lösung dieses Optimierungsproblems ist gegeben durch:

$$\min \frac{1}{2} \|w\|^2 \text{ unter } y_i(w \cdot x_i + b) \geq 1 \quad (2.15)$$

SVM mit Soft-Margin-Trennebenen Wenn die Daten nicht linear trennbar sind, hat das oben genannte Problem keine Lösung. Ein Ansatz, um dies zu umgehen, ist die

Einführung von Schlupfvariablen. Diese Variablen speichern die Abweichung vom definierten „Margin“. Nähere Details können der Studie [16] entnommen werden.

2.3.5 Bewertungskriterien für Klassifikationsmodelle

Nach der Klassifikation von Daten durch verschiedene Klassifikatoren werden die unterschiedlichen Ergebnisse verglichen. Dieser Vergleich wird anhand eines Gütemaßes durchgeführt. Das Gütemaß ermöglicht es, die Klassifikatoren zu vergleichen. Da in dieser Arbeit nur mit binären Klassifikationen gearbeitet wird, wird hier das Gütemaß für diese Aufgabe vorgestellt.

2.3.5.1 Konfusionsmatrix

Die Konfusionsmatrix ist eines der meist benutzten Gütemaße. Diese stellt die Vorhersagefähigkeit des Modells dar und wird durch eine 2x2 Matrix repräsentiert (bei binärer Klassifikation). Während die Zeilen der Instanzen eine Klasse repräsentieren, stellen die Spalten die vorhergesagte Klasse dar. Einer der größten Vorteile der Konfusionsmatrix ist, dass schnell ersichtlich ist, ob das Modell die Klassen falsch zuordnet [36, S. 164].

Tabelle 2.2: Konfusionsmatrix

		Vorhergesagte Klasse		Total
		Positiv	Negativ	
Aktuelle Klasse	Positiv	r_p	f_n	$r_p + f_p$
	Negativ	f_p	r_n	$f_n + r_n$
Total		$r_p + f_n$	$f_p + r_n$	N

Die Konfusionsmatrix wird in der Tabelle 2.2 dargestellt. Die möglichen Werte sind richtig positiv, falsch positiv, falsch negativ, richtig negativ. Diese Werte sind wie folgt definiert:

- **Richtig positiv** (r_p): entspricht der Anzahl der positiven Instanzen, die als positiv vorausgesagt wurden.
- **Falsch positiv** (f_p): entspricht der Anzahl von negativen Instanzen, die als positiv vorhergesagt wurden.
- **Falsch negativ** (f_n): entspricht der Anzahl von positiven Instanzen, die als negativ vorhergesagt wurden.
- **Richtig negativ** (r_n): entspricht der Anzahl der negativen Instanzen, die als negativ vorhergesagt wurden.

2.3.5.2 Standard-Maße

Aus der Konfusionsmatrix können folgende Gütemaße abgeleitet werden [36, S. 175]:

Genauigkeit (engl. „precision“) ist ein Maß, welches aussagt, wieviel der Information, die ein System geliefert hat, wirklich korrekt ist.

$$precision = \frac{r_p}{r_p + f_p} \quad (2.16)$$

Trefferquote (engl. „recall“) ist der Anteil der korrekt als positiv klassifizierten Objekte an der Gesamtheit der tatsächlich positiven Objekte.

$$recall = \frac{r_p}{r_p + f_n} \quad (2.17)$$

Treffergenauigkeit (engl. „accuracy“) entspricht dem Durchschnittskurs der korrekt vorhergesagten Instanzen.

$$precision = \frac{r_p + r_n}{r_p + f_p + r_n + f_n} \quad (2.18)$$

F-Maß ist das gewichtete harmonische Mittel aus der Genauigkeit (precision) und der Trefferquote (recall).

$$precision = 2 \frac{precision \cdot recall}{precision + recall} \quad (2.19)$$

Kapitel 3

Implementierung

In diesem Kapitel wird auf die Implementierung der temporalen Merkmalsaggregation eingegangen. Zunächst werden verwendete Tools vorgestellt. Danach wird die Struktur des Systems und die konkrete Implementierung erläutert. In der Implementierung geht es um die Vorverarbeitung der Daten für die späteren Experimente.

3.1 Angewandte Tools

Für die Realisierung des Ziels dieser Arbeit wurden verschiedene Tools eingesetzt. Hier wird eine kurze Beschreibung der jeweiligen Tools gegeben.

3.1.1 AMUSE

AMUSE steht für „Advanced MUSic Explorer“ und wird vom LS11 an der Technischen Universität Dortmund entwickelt. Es ist ein Multitool-Framework für die Analyse von Musikdaten. Eine komplette Beschreibung ist in [35] zu finden.

Die Audiomerkmale, die diese Arbeit benutzt, wurden mit AMUSE vorberechnet. Die Merkmalsvektoren sind im ARFF-Format gespeichert. Die Abbildung 3.1 zeigt ein Beispiel einer ARFF-Datei. Hier wird ein erweitertes ARFF-Format verwendet: AMUSE-Attribute werden als Kommentare unter der Beziehungsbeschreibung platziert. Diese Attribute speichern Informationen über die Datensatzgröße, Sample-Rate und Zeitrahmen-Größe. Da für jedes Merkmal der entsprechende Zeitrahmen im Attribut „*WindowNumber*“ gespeichert wird, ist es einfach, die Zeitintervalle, aus denen die Merkmale extrahiert wurden, zu erkennen.

3.1.2 IntelliJ IDEA

IntelliJ IDEA ist eine integrierte Entwicklungsumgebung (IDE) für die Programmiersprache Java. Sie wird durch das Softwareunternehmen JetBrains entwickelt. Das Hauptprogramm wurde mit dieser IDE entwickelt. Deren Vorteil besteht darin, dass alle benötigten

```

@RELATION 'Music feature'
%rows=13
%columns=10427
%sample_rate=22050
>window_size=512

@ATTRIBUTE 'Mel frequency cepstral coefficients - MIR Toolbox' NUMERIC
@ATTRIBUTE 'Mel frequency cepstral coefficients - MIR Toolbox' NUMERIC
@ATTRIBUTE 'Mel frequency cepstral coefficients - MIR Toolbox' NUMERIC
@ATTRIBUTE 'Mel frequency cepstral coefficients - MIR Toolbox' NUMERIC
@ATTRIBUTE 'Mel frequency cepstral coefficients - MIR Toolbox' NUMERIC
@ATTRIBUTE 'Mel frequency cepstral coefficients - MIR Toolbox' NUMERIC
@ATTRIBUTE 'Mel frequency cepstral coefficients - MIR Toolbox' NUMERIC
@ATTRIBUTE 'Mel frequency cepstral coefficients - MIR Toolbox' NUMERIC
@ATTRIBUTE 'Mel frequency cepstral coefficients - MIR Toolbox' NUMERIC
@ATTRIBUTE 'Mel frequency cepstral coefficients - MIR Toolbox' NUMERIC
@ATTRIBUTE 'Mel frequency cepstral coefficients - MIR Toolbox' NUMERIC
@ATTRIBUTE 'Mel frequency cepstral coefficients - MIR Toolbox' NUMERIC
@ATTRIBUTE WindowNumber NUMERIC

@DATA
-1.99840144433e-14,1.7763568394e-15,-7.9936057773e-15,5.3290705182e-15,-1.37667655054e-14,7.5495165
-1.99840144433e-14,-2.26485497024e-14,-2.88657986403e-14,-1.33226762955e-15,1
-1.99840144433e-14,1.7763568394e-15,-7.9936057773e-15,5.3290705182e-15,-1.37667655054e-14,7.5495165
-1.99840144433e-14,-2.26485497024e-14,-2.88657986403e-14,-1.33226762955e-15,2
0.269100689124,-0.00920434246428,-0.243960065528,-0.108891186091,-0.0265755503633,-0.203239464397,0
.198166026018,0.312136113626,-0.114667176423,3
2.81505966652,1.03382708973,-0.1081474671,0.136642308296,0.235363768105,0.0928950027467,-0.21017622
00761201005403,0.184237102568,-0.217594813841,4
2.97281626977,0.628709481559,0.310820482213,-0.255729279339,0.183495266708,0.14560337977,0.03995030
0265413477297,-0.147290526158,-0.460812055706,5
3.62737806344,0.390235656236,0.511880293515,0.222066824197,0.196303264017,-0.058132351151,0.1906897
108420650708,0.0523921710708,0.0117008783618,6

```

Abbildung 3.1: Beispiel einer ARFF-Datei

Plugins in der Ultimate-Version vorinstalliert sind. Als Student kann eine kostenlose Lizenz zum Zwecke einer Studie erworben werden.

3.1.3 ARFIT-Toolbox

„ARFIT-Toolbox“ ist ein Paket von MATLAB-Funktionen zur Analyse von multivariaten Zeitreihen mit autoregressiven Verfahren. Eine komplette Beschreibung kann in [31] eingesehen werden. Das Paket wird verwendet für die Berechnung der MAR- und DAR-Koeffizienten.

3.1.4 WEKA

WEKA ist die Abkürzung für „Waikato Environment for Knowledge Analysis“. Es ist ein Softwaretool, das an der University of Waikato entwickelt wurde und ist in Java geschrieben. Es stellt verschiedene Klassifikationsverfahren wie Entscheidungsbäume und künstliche neuronale Netze bereit. In der Studie von Hall [12] wird eine komplette Beschreibung von WEKA gegeben.

In den Experimenten der vorliegenden Arbeit werden die Klassifikatoren: k -NN, SVM und RF aus WEKA verwendet. Diese werden in MATLAB durchgeführt. Die Klassifikatoren werden aus MATLAB aufgerufen. Die Klassifikation wird in der Sektion 4.2 behandelt.

3.2 Struktur

Die Audiomerkmale wurden schon berechnet und liegen als ARFF-Dateien vor. Die Daten für die Klassifikation von Genres und für die Erkennung von Instrumenten liegen in zwei unterschiedliche Ordnern. Diese Ordner besitzen eine eigene Struktur. Bei der Klassifikation von Genres gibt es einen Ordner pro Künstler, der als Unterordner alle Alben des Künstlers beinhaltet. Für jedes Album gibt es je Musikstück einen Ordner mit allen Merkmalen, die berechnet wurden.

Für die Instrumentenerkennung gibt es pro Musikstück einen Ordner. In diesem sind alle berechneten Merkmale für diesen Titel gespeichert.

Es liegt je eine ARFF-Datei für ein Merkmal vor. Auf alle diese ARFF-Dateien müssen nun die Aggregationsmethoden angewandt werden, um die temporalen Merkmale zu berechnen. Diese Vorverarbeitung ist wichtig für die spätere Untersuchung der temporalen Merkmale.

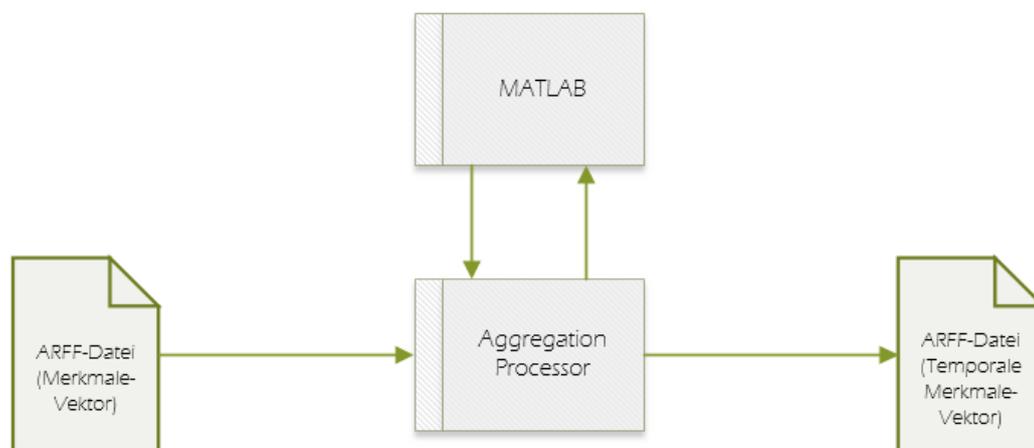


Abbildung 3.2: Die Struktur der Vorverarbeitung.

Die Abbildung 3.2 zeigt die Struktur der Vorverarbeitung. Die ARFF-Dateien werden zunächst eingelesen. Dazu wird der Pfad eingegeben und die Musikstücke werden sequenziell abgearbeitet. Für das Lesen des ARFF-Formats wird der „ARFF-Loader“ aus dem AMUSE Framework verwendet. Damit können die AMUSE-Attribute auch mitgelesen werden. Die Daten aus den Dateien werden an den „Aggregation Processor“ weitergegeben. Der letztere ist für die konkrete Berechnung der temporalen Merkmale zuständig. Für jede

Aggregationsmethode gibt es einen „Aggregation Processor“. Die ARFF-Dateien aus einer Merkmalsgruppe werden zunächst zusammengeführt. Nun kann die konkrete Berechnung der temporalen Merkmale erfolgen:

- „Statistiken“: hier werden die Werte direkt in der Java Klasse berechnet.
- „Stacking“: die Werte werden auch in Java berechnet.
- MAR, DAR, MARDAR: die Daten aus den zusammengeführten ARFF-Dateien werden an MATLAB weitergeleitet. Dort werden die Koeffizienten der autoregressiven Modelle berechnet. Die Berechnung findet mit Hilfe des Pakets ARFIT-Toolbox statt. Nach der Berechnung der Koeffizienten werden sie an Java weitergeleitet. Die Kommunikation zwischen Java und Matlab wird durch das Paket MatlabControl ermöglicht.

3.3 Temporale Merkmalsberechnung

Das Klassendiagramm in Abbildung 3.3 zeigt die benötigten Klassen für die Merkmalsberechnung. Die Klassen sind im Folgenden aufgelistet:

[FEATURE]

Die Feature-Klasse ist eine Klasse, in der die Informationen eines Audiomerkmals gespeichert werden. Eine Feature-Klasse hat eine Dimension, die im Attribut „*dimension*“ gespeichert wird. Sie besitzt auch eine Liste von Werten (das Attribut „*values*“) und eine Liste von Extraktionsfenstern (das Attribut „*windows*“). Weiterhin speichert das Attribut „*samplerate*“ die Anzahl der Samples pro Sekunde, und die Beschreibung des Merkmals ist in dem Attribut „*description*“ enthalten. Das Attribut „*colValues*“ beinhaltet den Vektor aller Werte über das Texturfenster. Dieses Attribut wird für die Eliminierung der NaN-Werte verwendet.

Außer den Methoden *columnsValues()* und *eliminateNaN()* sind alle anderen Methoden der Feature-Klasse Getter- und Setter-Methoden. Die Methode *columnsValues()* berechnet den Wert von „*colValues*“. Bei der Methode *eliminateNaN()* werden die NaN-Werte durch den Durchschnittswert des Vektors ersetzt.

Die Feature-Klasse wird von der FeaturesLoader-Klasse für die Speicherung der Werte aus einem Merkmalsvektor benutzt.

[ARFFFEATURELOADER]

Die ArffFeatureLoader-Klasse stellt eine Methode zum Laden eines Merkmalsvektors aus einer ARFF-Datei bereit. Die Methode heißt *loadFeature()* und benötigt als Parameter den Pfad der Datei. Die ARFF-Datei wird eingelesen und jede Zeile abgearbeitet. Die Daten werden dann in einem Objekt der Feature-Klasse gespeichert und zurückgegeben.

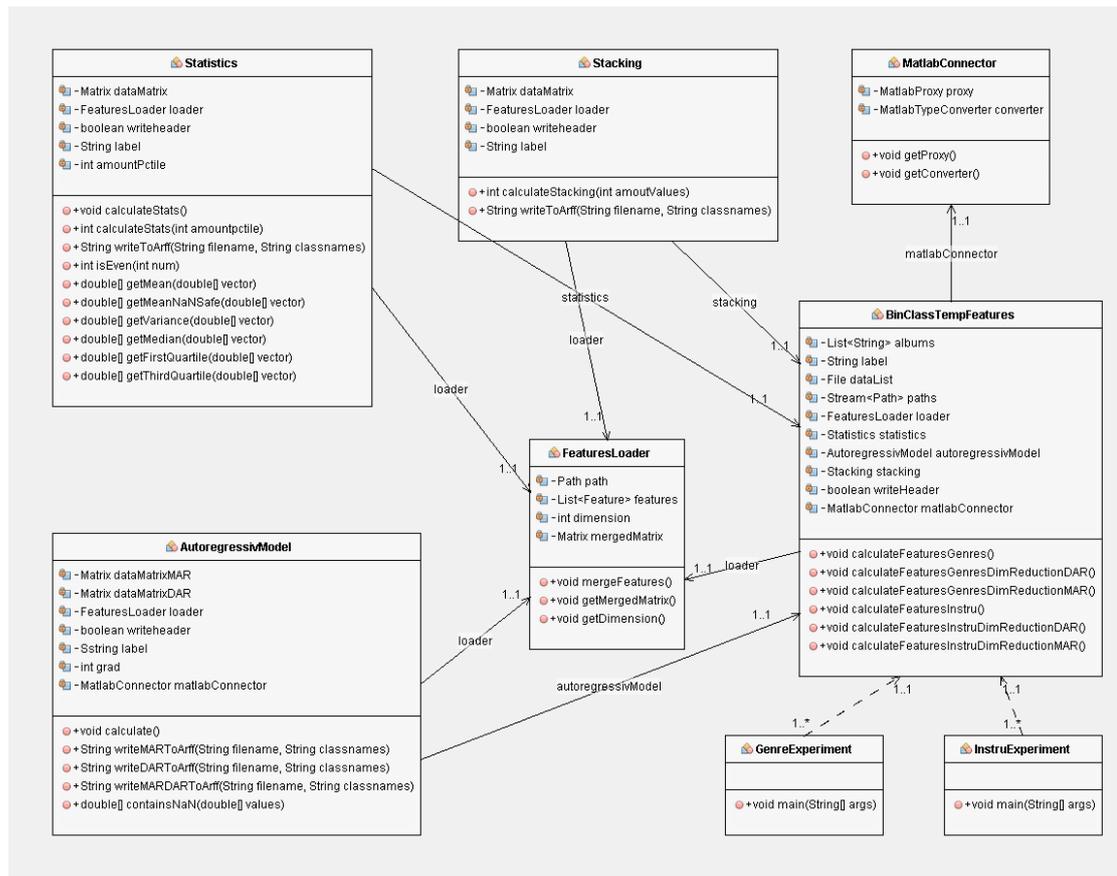


Abbildung 3.3: Das Klassendiagramm für die temporale Merkmalsberechnung

[FEATURESLOADER]

Die FeaturesLoader-Klasse ist ein Loader für mehrere Merkmale. Diese Merkmale können mit der Methode *mergeFeatures()* zusammengeführt werden. Die geladenen Merkmalsvektoren werden in einer Matrix gespeichert. Die Attribute der FeaturesLoader-Klasse sind ein Pfad *path*, eine Liste von Merkmalen *features*, eine Dimension *dimension* und eine Matrix *mergedMatrix*.

[MATLABCONNECTOR]

Die MatlabConnector-Klasse repräsentiert die Schnittstelle zu Matlab. Darüber kann das Programm Matlab steuern. Sie verwendet das Paket Matlabcontrol. Die Kommunikation mit Matlab läuft über das Attribut *proxy* vom Typ MatlabProxy. Außerdem besitzt die Klasse das Attribut *converter* vom Typ MatlabTypeConverter. Das Attribut *converter* ist für die Typumwandlung zwischen Java-Objekten und Matlab-Objekten zuständig. Die Klasse beinhaltet nur die notwendigen Gettermethoden.

Die `MatlabConnector`-Klasse wird von der `AutoregressivModel`-Klasse für die Berechnung der Koeffizienten auf Basis der ARFIT-Toolbox verwendet.

[AUTOREGRESSIVMODEL]

Die `AutoregressivModel`-Klasse ist für die Berechnung der MAR-, DAR- und MARDAR-Vektoren zuständig. Die Attribute `dataMatrixMAR` und `dataMatrixDAR` speichern die Werte der autoregressiven Modelle MAR und DAR. Das Attribut `writeheader` ist ein boolescher Wert, der merkt, ob der Header der ARFF-Datei schon geschrieben wurde. Die wichtigsten Methoden sind:

- `calculate()`: berechnet die Koeffizienten der autoregressiven Modelle. Die Koeffizienten für MAR und DAR werden jeweils in der Matrix `dataMatrixMAR` bzw. `dataMatrixDAR` gespeichert.
- `writeMARToArff()`: schreibt die MAR-Koeffizienten in eine ARFF-Datei. Alle Werte in der Matrix `dataMatrixMAR` werden sequenziell in die ARFF-Datei geschrieben.
- `writeDARToArff()`: schreibt die DAR-Koeffizienten in eine ARFF-Datei. Alle Werte in der Matrix `dataMatrixDAR` werden sequenziell in die ARFF-Datei geschrieben.
- `writeMARDARToArff()`: schreibt die MARDAR-Koeffizienten in eine ARFF-Datei. Die Werte aus den Matrizen `dataMatrixMAR` und `dataMatrixDAR` werden kombiniert und auch sequenziell in die ARFF-Datei geschrieben.

[STACKING]

Die `Stacking`-Klasse repräsentiert die Aggregationsmethode „Stacking“. Sie besitzt ein Attribut `dataMatrix` zum Speichern der berechneten Werte. Außerdem beinhaltet die Klasse das Attribut `loader` zum Laden der Merkmale und das Attribut `label` für das Label der benutzten Musikstücke. Die wichtigsten Methoden sind:

- `calculateStacking()`: berechnet den Vektor der Aggregationsmethode „Stacking“. Der Vektor verwendet die geladenen Merkmale vom „FeaturesLoader“.
- `writeToArff()`: schreibt den berechneten Vektor in eine ARFF-Datei.

[STATISTICS]

Die `Statistics`-Klasse bezeichnet die Aggregationsmethode „Statistiken“. Sie beinhaltet ein Attribut `dataMatrix` als Zwischenspeicher für die berechneten Werte. Das Attribut `loader` vom Typ `FeaturesLoader` lädt die Merkmale aus der ARFF-Datei. Die wichtigsten Methoden sind:

- `calculateStatistics()`: dient zur Berechnung der Werte. Alle Werte werden in der `dataMatrix` gespeichert. Für jeden Merkmalsvektor werden fünf statistische Werte berechnet.

- *writeToArff()*: schreibt die berechneten Werte der Aggregationsmethode „Statistiken“ in eine ARFF-Datei.

[BINCLASSTEMPFEATURES]

Die `BinClassTempFeatures`-Klasse berechnet die Daten für die binären Klassifikationsexperimente. Der Konstruktor *„BinClassTempFeatures(File data, String label)“* initialisiert ein binäres Klassifikationsexperiment. Als Eingabe-Parameter wird eine Liste aller Alben für ein Genre oder eine Liste aller Musikstücke für ein Instrument benötigt. Zusätzlich wird ein Label für jedes Genre und für jedes Instrument benötigt.

Weiterhin beinhaltet die Klasse folgende Methoden:

- *calculateFeaturesGenres()*: Berechnet für ein Genre alle temporalen Merkmalsvektoren („Statistiken“, „Stacking“, MAR(1 bis 3), DAR(1 bis 3), MARDAR(1 bis 3)).
- *calculateFeaturesGenresDimReductionMAR()*: Berechnet für ein Genre die dimensionsgleichen temporalen Merkmalsvektoren mit den Methoden „Statistiken“, „Stacking“ und MAR.
- *calculateFeaturesGenresDimReductionDAR()*: Berechnet für ein Genre die dimensionsgleichen temporalen Merkmalsvektoren mit den Methoden „Statistiken“, „Stacking“ und DAR.
- *calculateFeaturesInstru*: Berechnet für ein Instrument alle temporalen Merkmalsvektoren.
- *calculateFeaturesInstruDimReductionMAR()*: Berechnet für ein Instrument die dimensionsgleichen temporalen Merkmalsvektoren mit den Methoden „Statistiken“, „Stacking“ und MAR.
- *calculateFeaturesInstruDimReductionDAR()*: Berechnet für ein Instrument die dimensionsgleichen temporalen Merkmalsvektoren mit den Methoden „Statistiken“, „Stacking“ und DAR.

[GENREEXPERIMENT]

Die `GenreExperiment`-Klasse initiiert die Berechnung der Daten für die binären Klassifikationen von Genres. Es werden vier Objekte vom Typ `BinClassTempFeatures` für die Klassifikation von Genres erzeugt. Es gibt ein Objekt je Genre.

[INSTRUEXPERIMENT]

Die `InstruExperiment`-Klasse startet die Berechnung der Daten für die Instrumentenerkennung. Es werden vier Objekte vom Typ `BinClassTempFeatures` für die Erkennung von Instrumenten erzeugt. Es gibt ein Objekt je Instrument.

3.4 Vorgehensweise

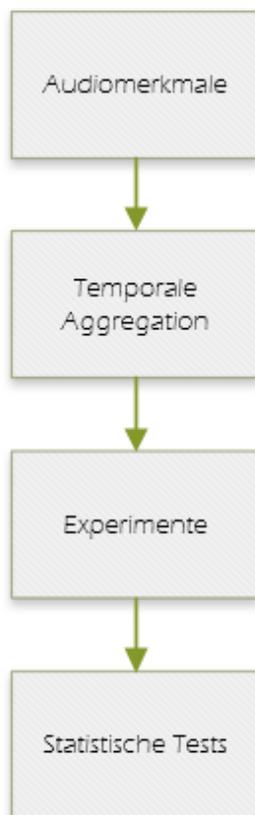


Abbildung 3.4: Vorgehensweise für die Untersuchung der Aggregationsmethoden

Die Abbildung 3.4 zeigt die Vorgehensweise für die Untersuchung der Aggregationsmethoden. Die Audiomerkmale liegen schon in Form von ARFF-Dateien für alle Musikstücke vor. Die Beschreibung des Datenkorpus wird im nächsten Kapitel erbracht. Mit dem entwickelten Programm können aus den Audiomerkmalen temporale Merkmale berechnet werden. Die temporalen Merkmalsvektoren werden für die Experimente verwendet. Bei den Experimenten werden Klassifikationen mit verschiedenen Parametern ausgeführt. Danach werden die Ergebnisse dieser Klassifikation analysiert. Zuletzt werden die Erkenntnisse aus den Ergebnissen statistisch getestet.

Im nächsten Kapitel wird auf die Experimente und ihre Auswertungen eingegangen.

Kapitel 4

Experimente und Auswertung

4.1 Daten-Korpus

Das Daten-Korpus stellt ein fundamentales Element für den Aufbau eines Klassifikationssystems dar. Musikdaten sind im Allgemeinen einfach zu bekommen. Wegen der Urheberrechte kann allerdings nicht jedes beliebige Musikstück benutzt werden. Daher wurden im Rahmen dieser Arbeit zwei Datensätze aus dem Lehrstuhl 11 der TU Dortmund für die Experimente verwendet. Während der erste Datensatz Musikstücke für die Klassifikation von Genres beinhaltet, enthält der zweite Datensatz Stücke für die Instrumentenerkennung. Im Folgenden werden die beiden Datensätze beschrieben.

4.1.1 Daten für die Klassifikation von Genres

Wie unter Kapitel 4.1 erwähnt, wurde dieser Datensatz am Lehrstuhl 11 der Fakultät Informatik der TU Dortmund zusammengestellt. Dieser beinhaltet 120 kommerzielle Alben, die zu Forschungszwecken erworben wurden. Die komplette Alben-Liste ist der Webseite des Lehrstuhls zu entnehmen. Die 120 Alben sind in sechs Kategorien unterteilt: Klassik, Jazz, Pop/Rock, RnB, Electro und Rap. Für die Experimente wurden nur vier Kategorien ausgewählt. Insgesamt sind 610 Titel vorhanden, die aus den Alben stammen. Die Distribution der Titel und Alben nach Musikgenres ist in der Tabelle 4.1 dargestellt.

Tabelle 4.1: Verteilung der Alben nach Genres

Genres	Anzahl Alben	Anzahl Musikstücke
Klassik	15	57
Jazz	15	62
Electro	15	75
Rap	15	82

Diese Sammlung wurde schon in anderen Arbeiten verwendet, unter anderem in der Arbeit von Vatulkin [34]. Die Audiomerkmale wurden für alle Musikstücke bei einer Sample-Rate von 22050 Hz extrahiert. Die Extraktion wurde mit dem Framework AMUSE durchgeführt. Die Daten werden hier nicht in eine Trainings- und Testmenge unterteilt. Stattdessen wird eine 10-fache Kreuzvalidierung benutzt. Außerdem werden binäre Klassifikationen mit diesem Datensatz durchgeführt, wobei ein Genre gegen alle anderen klassifiziert wird („One-Vs-All“). Dieses Experiment wird im nächsten Abschnitt erläutert.

4.1.2 Daten für die Instrumentenerkennung

Dieser Datensatz besteht aus 3000 Titeln, die mehrere Instrumententöne beinhalten. Ein Musikstück aus dieser Sammlung kann beispielsweise Töne von den Instrumenten Piano, Cello, Gitarre, etc. enthalten. Für diese Arbeit wurden vier Standard-Instrumente ausgewählt: Flöte, Piano, Gitarre und Trompete. Für alle Musikstücke wurden die Merkmale bei einer Sample-Rate von 22050 Hz berechnet. Diese Merkmale wurden wie oben mit dem Framework AMUSE berechnet. Die Tabelle 4.2 zeigt die Verteilung der Titel nach Instrumenten.

Hier werden auch die binären Klassifikationen unter Verwendung einer 10-fachen Kreuzvalidierung durchgeführt. Dieser Datensatz wurde auch durch den Lehrstuhl 11 bereitgestellt.

Tabelle 4.2: Verteilung der Titel nach Instrumenten

Instrument	Anzahl Musikstücke
Piano	1104
Gitarre	1684
Flöte	935
Trompete	1118

4.2 Binäre Klassifikationsexperimente

Mehrere binäre Klassifikationsexperimente wurden durchgeführt. Das Ziel dieser Experimente ist es, eine detaillierte Untersuchung der temporalen Merkmale durchzuführen. Im Folgenden wird auf die Experimente eingegangen. Hier werden die Probleme der Klassifikation von Genres und der Erkennung von Instrumenten untersucht.

Für jedes dieser Experimente wird eine Tabelle mit der Zusammenfassung der Parameter und eine Visualisierung gegeben. Danach wird eine Analyse der Ergebnisse erarbeitet.

4.2.1 Klassifikation von Musikgenres

Im ersten Experiment wird eine Klassifikation von Musikgenres durchgeführt. Die Parameter für dieses Experiment sind in der Tabelle 4.3 aufgelistet. Die verschiedenen Parameter werden im nächsten Absatz erläutert:

- **KLASSIFIKATIONSAUFGABE:** Hier werden binäre Klassifikationen aus vier Genres (Klassik, Electro, Jazz, Rap) durchgeführt. Die binäre Klassifikation ist vom Typ „One-Vs-All“, also ein Genre wird gegenüber den anderen klassifiziert. Die verwendeten Daten sind aus der ersten Sammlung. Es werden alle 610 Titel benutzt.
- **MERKMALE:** Es wurden insgesamt sieben Merkmale für diese Experimente ausgewählt. Diese sieben Merkmale stammen aus vier Merkmalsgruppen. Für jede Merkmalsgruppe werden alle Aggregationsmethoden angewandt. Die autoregressiven Modelle wurden für den Rang eins bis drei berechnet.
- **KLASSIFIKATIONSMETHODEN:** Vier Klassifikatoren (k -NN, RF, GMM und SVM) wurden benutzt. Für drei der Klassifikatoren (k -NN, RF, SVM) wurden die Standardwerte aus der Software WEKA verwendet. Beim k -NN ist aber der Wert von $k = 10$ benutzt worden. k ist die Anzahl an „Nächsten-Nachbarn“. Diese Werte wurden aus der Evaluation der WEKA-Software ermittelt. Für GMM wurde eine Implementierung aus MATLAB verwendet. Die Ergebnisse wurden mit dem F-Maß berechnet. Da die Daten nicht balanciert sind, wäre die Genauigkeit („*accuracy*“) kein geeignetes Maß für die Qualität der Klassifikation.

Tabelle 4.3: Parameter für die Klassifikation von Genres

Parametername	Werte	Anzahl
KLASSIFIKATIONSAUFGABE		
Klassifikationsaufgabe	Genres: Klassik, Electro, Jazz, Rap	4
Datenset	610 Titel aus 120 Alben	1
MERKMALE		
Merkmale	ZCR, RMS, „Spectral kurtosis“, „Spectral Discrepancy“, „Spectral skewness“, MFCC, Chroma	7
Aggregationsmethode	„Statistiken“, „Stacking“, DAR(r), MAR(r), DARMAR(r) mit $r \in \{1, 2, 3\}$	11
KLASSIFIKATIONSMETHODEN		
Algorithmen	k -NN, SVM, GMM, RF	1
Evaluation	10-fache Kreuzvalidierung auf dem Datensatz	1
Gütemaß	F-Maß	1

Alle Ergebnisse der Klassifikation von Genres sind im Anhang A in den Tabellen A.1, A.2, A.3 und A.4 aufgelistet. Diese Ergebnisse werden nun analysiert.

Die Abbildung 4.1 stellt eine Visualisierung der Ergebnisse der Klassifikation „Klassik-Vs-Alle“ dar. Es werden die F-Maß-Werte der Klassifikationsergebnisse dargestellt. Auf der Y-Achse sind die Aggregationsmethoden aufgelistet und auf der X-Achse die Klassifikationsmethoden mit den Merkmalsgruppen.

In der Merkmalsgruppe „Zeitbereich“ (ZB) wird das genaueste Ergebnis mit der Aggregationsmethode „Statistiken“ und dem Klassifikator RF erzielt, nämlich 71,52%. Bei der Merkmalsgruppe „Spektralbereich“ (SB) wird das beste Ergebnis von 77,77% mit der Methode „Stacking“ und dem Klassifikator RF ermittelt. Die Methode „Stacking“ liefert auch das beste Ergebnis von 63,6% bei der Merkmalsgruppe „Cepstral-Bereich“ (CB) mit dem Klassifikator SVM. Ebenso für die Merkmalsgruppe „Chroma“ (Ch) wird das beste Ergebnis von 62,78% mit der Aggregationsmethode „Stacking“ ermittelt, allerdings hier mit dem Klassifikator GMM. Insgesamt liefert die Aggregationsmethode „Stacking“ bei der Klassifikation „Klassik-Vs-Alle“ die besten Ergebnisse. Diese besten Ergebnisse werden mit unterschiedlichen Klassifikatoren erzielt.

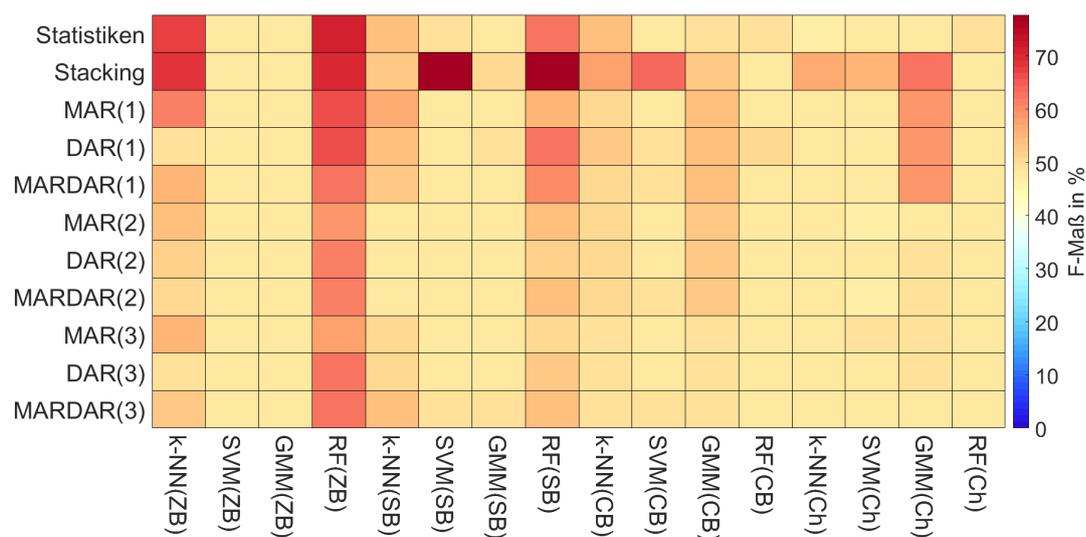


Abbildung 4.1: Ergebnisse der Klassifikation für „Klassik-Vs-Alle“

Die Abbildung 4.2 stellt eine Visualisierung der Ergebnisse der Klassifikation „Electro-Vs-Alle“ dar. Für die Merkmalsgruppe „Zeitbereich“ liefert die Aggregationsmethode „Stacking“ das beste Ergebnis von 54,82% mit dem Klassifikator k -NN. In der Merkmalsgruppe „Spektralbereich“ wird auch hier das beste Ergebnis mit der Methode „Stacking“ erreicht. Hier sind es 52,15% mit dem Klassifikator SVM. Bezüglich der Merkmalsgruppe „Cepstralbereich“ wurde das beste Ergebnis von 64,32% mit der Aggregationsmethode MAR von Rang drei, unter Verwendung des Klassifikators GMM erhalten. Letztlich wurde das beste

Ergebnis bei der Merkmalsgruppe „Chroma“ mit der Methode MARDAR vom Rang eins geliefert. Hier erzielt die Aggregationsmethode MAR von Rang drei das insgesamt beste Ergebnis. Allerdings ist die Methode „Stacking“ in zwei von vier Merkmalsgruppen besser.

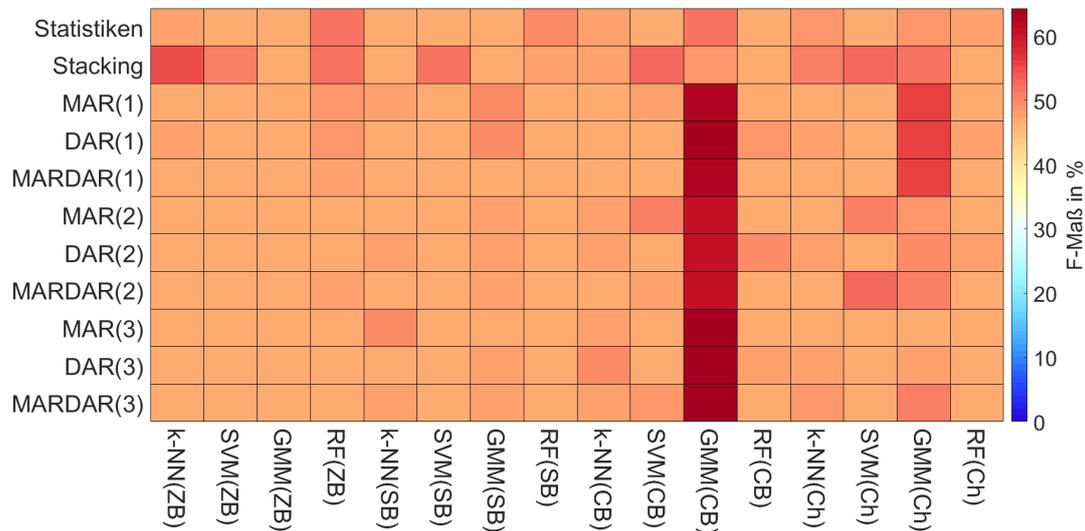


Abbildung 4.2: Ergebnisse der Klassifikation für „Electro-Vs-Alle“

Bei der Klassifikation „Jazz-Vs-Alle“ (siehe Abbildung 4.3) liefert „Stacking“ das beste Ergebnis von 53,28% für die Merkmalsgruppe „Zeitbereich“ mit dem Klassifikator RF. Bei der Merkmalsgruppe „Spektralbereich“ wird das beste Ergebnis von 54,64% auch mit der Aggregationsmethode „Stacking“ und dem Klassifikator SVM erzielt. Auch für die Merkmalsgruppen „Cepstralbereich“ und „Chroma“ werden die besten Ergebnisse von 54,91% bzw. 59,6% mit der Methode „Stacking“ und dem Klassifikator SVM ermittelt. Insgesamt liefert die Methode „Stacking“ für alle vier Merkmalsgruppen die besten Ergebnisse.

Die Abbildung 4.4 stellt eine Visualisierung der Ergebnisse der Klassifikation „Rap-Vs-Alle“ dar. Für die Merkmalsgruppe „Zeitbereich“ liefert die Aggregationsmethode MAR vom Rang zwei das beste Ergebnis von 56,19% mit dem Klassifikator RF. Bei der Merkmalsgruppe „Spektralbereich“ erhält man das beste Ergebnis von 55,44% mit der Methode „Statistiken“. Der hier verwendete Klassifikator ist k -NN. Für die Merkmalsgruppe „Cepstralbereich“ wird das beste Ergebnis von 57,96% mit der Aggregationsmethode „Stacking“ und dem Klassifikator SVM errechnet. Ebenso wird bei der Merkmalsgruppe „Chroma“ das beste Ergebnis von 57,8% mit der Methode „Stacking“ und dem Klassifikator SVM geliefert. Insgesamt erzielt die Aggregationsmethode „Stacking“ die besten Ergebnisse für zwei von vier Merkmalsgruppen.

Es ist also eine Tendenz zu erkennen. Die Aggregationsmethode „Stacking“ scheint die besten Ergebnisse zu erzielen, gefolgt von der Methode „Statistiken“. Diese Tendenz wurde

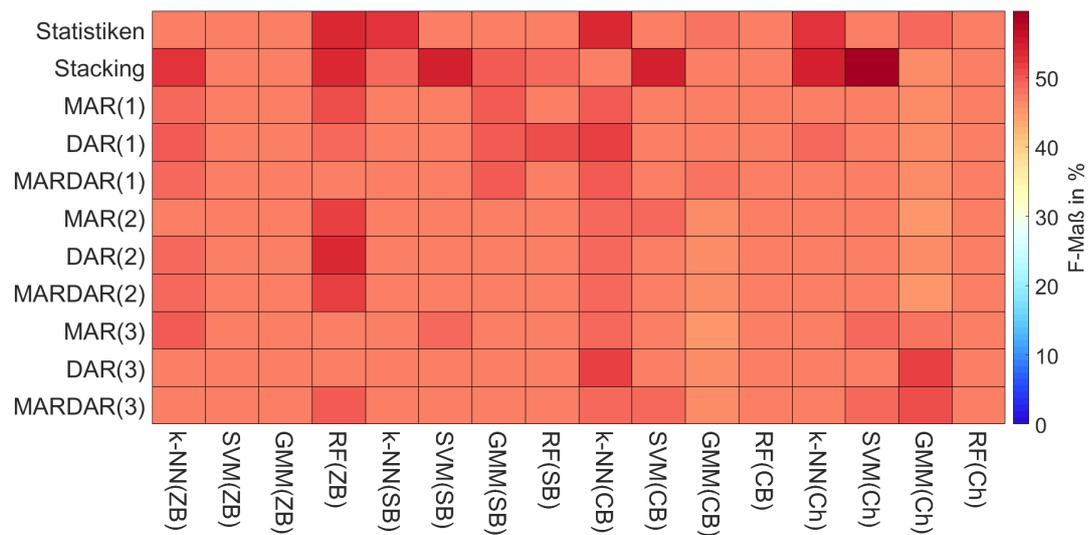


Abbildung 4.3: Ergebnisse der Klassifikation für „Jazz-Vs-Alle“

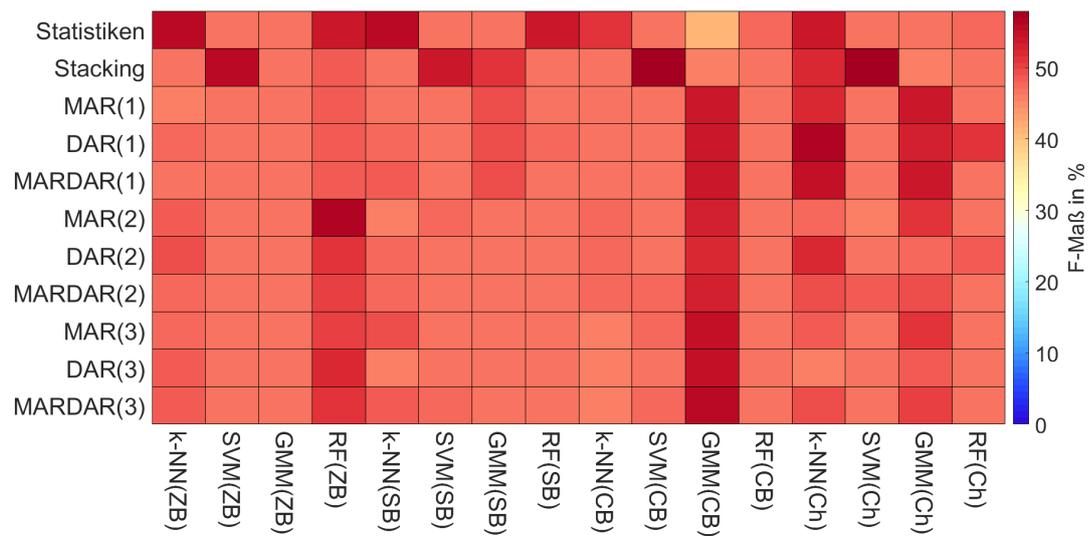


Abbildung 4.4: Ergebnisse der Klassifikation für „Rap-Vs-Alle“

für die Klassifikation von Genres festgestellt. Im nächsten Schritt wird das Problem der Instrumentenerkennung untersucht.

4.2.2 Instrumentenerkennung

In diesem Experiment wird eine Erkennung von Instrumenten durchgeführt. Die Parameter für dieses Experiment sind in der Tabelle 4.4 aufgelistet. Die Parameter der Klassifikation sind im Folgenden erklärt:

- **KLASSIFIKATIONSAUFGABE:** Auch hier werden binäre Klassifikationen für vier Instrumente (Piano, Gitarre, Flöte, Trompete) durchgeführt. In diesem Fall wird auch die Methode „One-Vs-All“ eingesetzt. Die verwendeten Daten sind aus der zweiten Sammlung. Es werden insgesamt 3000 Musikstücke verwendet.
- **MERKMALE:** Wie im vorherigen Experiment wurden insgesamt sieben Merkmale ausgewählt. Damit können auch die Ergebnisse der beiden Experimente untereinander verglichen werden. Diese sieben Merkmale stammen aus vier Merkmalsgruppen. Für jede Merkmalsgruppe werden alle Aggregationsmethoden angewandt.
- **KLASSIFIKATIONSMETHODEN:** Vier Klassifikatoren (k -NN, RF, GMM und SVM) wurden benutzt. Die Standard-Werte aus der Software WEKA wurden für drei der Klassifikatoren (k -NN, RF, SVM) verwendet. Bei k -NN wird wieder der Wert $k = 10$ benutzt. Für GMM wurde eine Implementierung aus MATLAB verwendet. Die Ergebnisse wurden mit dem F-Maß berechnet.

Tabelle 4.4: Parameter für das Instrumentenerkennungsexperiment

Parametername	Werte	Anzahl
KLASSIFIKATIONSAUFGABE		
Klassifikationsaufgabe	Piano, Gitarre, Flöte, Trompete	4
Datenset	3000 Musikstücke	1
MERKMALE		
Merkmale	ZCR, RMS, „Spectral Kurtosis“, „Spectral Discrepancy“, „Spectral Skweness“, MFCC, Chroma	7
Aggregationsmethode	„Statistiken“, „Stacking“, DAR(r), MAR(r), DARMAR(r) mit $r \in \{1, 2, 3\}$	11
KLASSIFIKATIONSMETHODEN		
Algorithmen	k -NN, SVM, GMM, RF	1
Evaluation	10-fache Kreuzvalidierung auf dem Datensatz	1
Gütemaß	F-Maß	1

Alle Ergebnisse der binären Klassifikationen sind in den Tabellen A.5, A.6, A.7 und A.8 im Anhang A aufgelistet.

In der Abbildung 4.5 ist eine Visualisierung der Ergebnisse für die Erkennung von „Piano-Vs-Alle“ dargestellt. Bei der Merkmalsgruppe „Zeitbereich“ wird das beste Ergebnis von 69,96% mit der Aggregationsmethode „Stacking“ und dem Klassifikator RF errechnet. Auch für die Merkmalsgruppe „Spektralbereich“ liefert die Methode „Stacking“ das beste Ergebnis von 72,16% mit dem Klassifikator RF. In der Aggregationsmethode

„Stacking“ ergeben sich auch die besten Ergebnisse von jeweils 75,69% und 68,25% für die Merkmalsgruppen „Cepstralbereich“ und „Chroma“. Allerdings werden diese Ergebnisse mit dem Klassifikator SVM erzielt. Insgesamt werden die besten Ergebnisse bei allen vier Merkmalsgruppen mit der Aggregationsmethode „Stacking“ ermittelt.

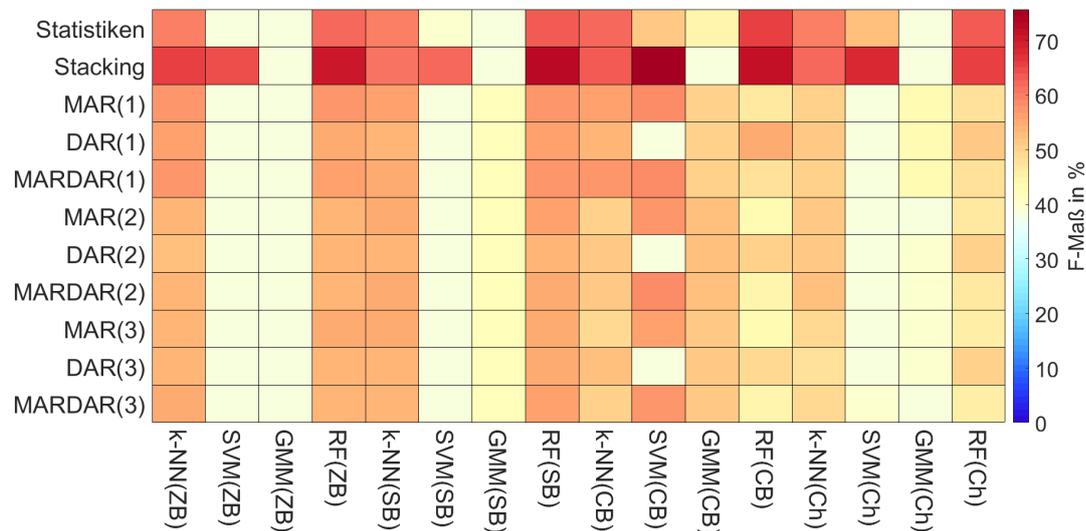


Abbildung 4.5: Ergebnisse der Klassifikation für „Piano-Vs-Alle“

Auch bei der Erkennung von „Flöte-Vs-Alle“ (siehe Abbildung 4.6) werden die besten Ergebnisse bei allen vier Merkmalsgruppen mit der Aggregationsmethode „Stacking“ errechnet. Für die Merkmalsgruppe „Zeitbereich“ wird das beste Ergebnis von 61,27% mit dem Klassifikator k -NN ermittelt. Bei der Merkmalsgruppe „Spektralbereich“ wird das beste Ergebnis von 64,35% mit dem Klassifikator RF errechnet. Der Klassifikator SVM liefert für die beiden Merkmalsgruppen „Cepstralbereich“ und „Chroma“ die besten Ergebnisse von 71,61% bzw. 63,8%.

Die Abbildung 4.7 stellt eine Visualisierung der Ergebnisse für die Erkennung „Gitarre-Vs-Alle“ dar. Bei der Merkmalsgruppe „Zeitbereich“ wird das beste Ergebnis von 79,25% mit der Aggregationsmethode „Stacking“ und dem Klassifikator SVM errechnet. Für die Merkmalsgruppe „Spektralbereich“ liefert auch die Methode „Stacking“ das beste Ergebnis von 81,23%. Aber in diesem Fall wird das Ergebnis mit dem Klassifikator RF errechnet. Bei der Merkmalsgruppe „Cepstralbereich“ wird das beste Ergebnis von 82,37% mit der Aggregationsmethode „Stacking“ und dem Klassifikator SVM ermittelt. Für die letzte Merkmalsgruppe „Chroma“ wird das beste Ergebnis von 66,94% mit der Aggregationsmethode „Stacking“ und dem Klassifikator RF berechnet. Insgesamt sind auch hier die besten Ergebnisse mit der Aggregationsmethode „Stacking“ erzielt worden.

Die Abbildung 4.8 stellt eine Visualisierung der Ergebnisse der Erkennung „Trompete-Vs-Alle“ dar. Bei den Merkmalsgruppen „Zeitbereich“ und „Spektralbereich“ werden die

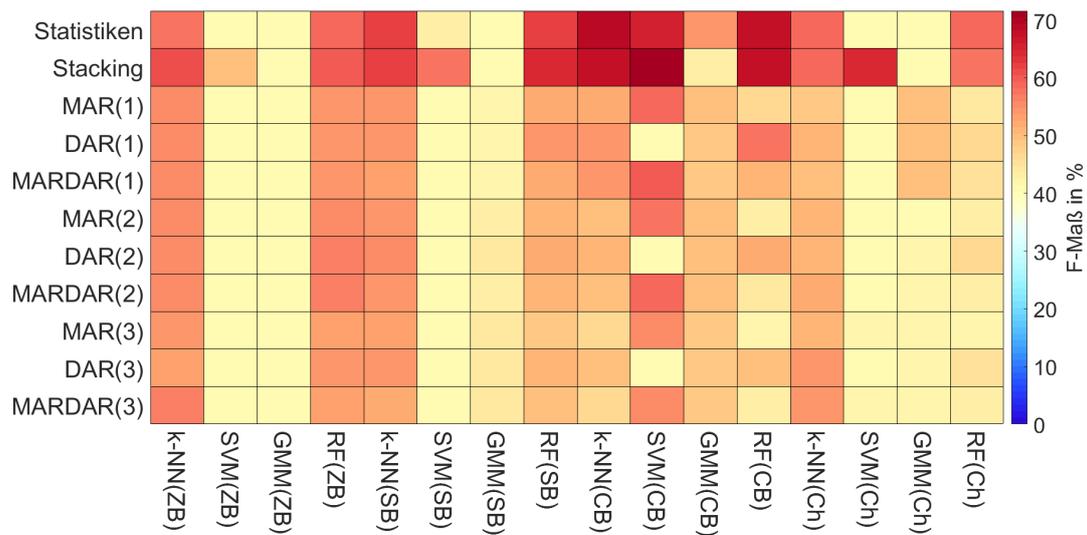


Abbildung 4.6: Ergebnisse der Klassifikation für „Flöte-Vs-Alle“

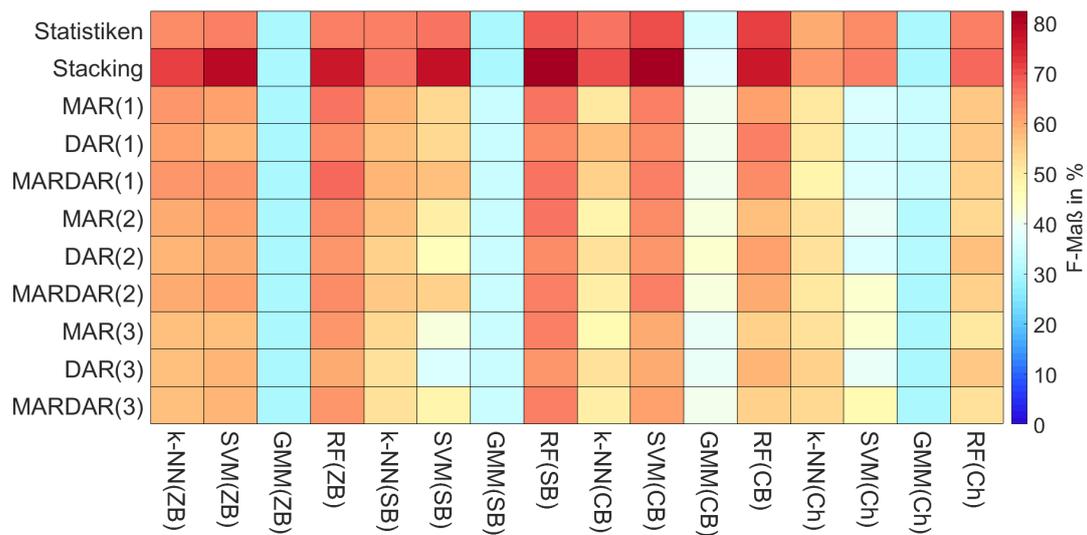


Abbildung 4.7: Ergebnisse der Klassifikation für „Gitarre-Vs-Alle“

besten Ergebnisse von 79,53% und 80,85% mit der Aggregationsmethode „Stacking“ und dem Klassifikator RF errechnet. Für die Merkmalsgruppe „Cepstralbereich“ wird das beste Ergebnis von 81,48% mit der Aggregationsmethode „Stacking“ ermittelt. Bei der Merkmalsgruppe „Chroma“ wird das beste Ergebnis von 69,79% mit der Aggregationsmethode „Statistiken“ und dem Klassifikator RF errechnet. Insgesamt wird das beste Ergebnis mit den Aggregationsmethoden „Stacking“ und „Statistiken“ geliefert.

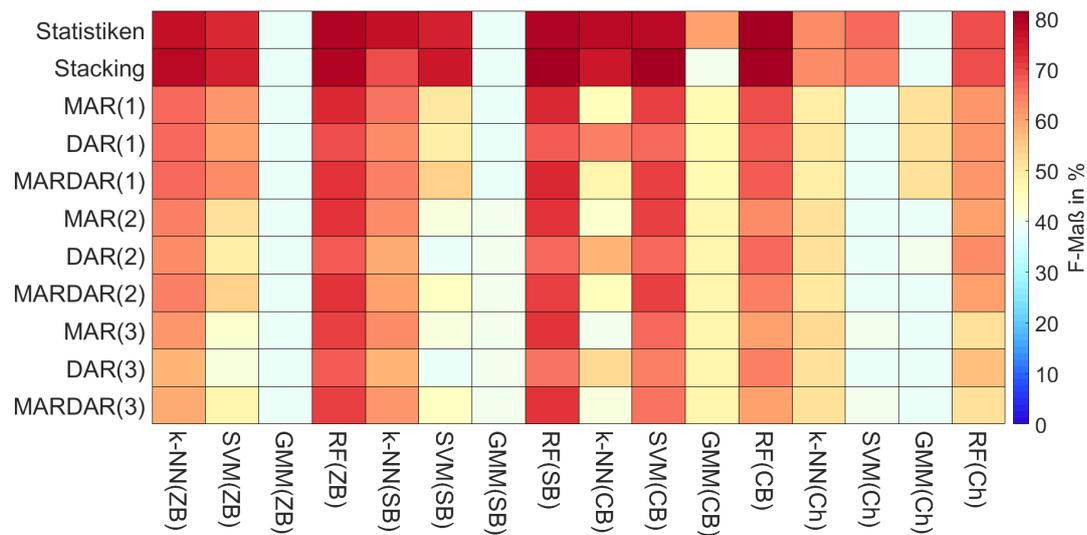


Abbildung 4.8: Ergebnisse der Klassifikation für „Trompete-Vs-Alle“

Auch hier zeigt sich die Tendenz bei der Erkennung von Instrumenten, dass bessere Ergebnisse mit der Aggregationsmethode „Stacking“ geliefert werden. Die Hypothese, dass die Methode „Stacking“ die beste bei der Klassifikation von Genres und der Erkennung von Instrumenten sei, kann aufgestellt werden. Diese Hypothese wird im nächsten Abschnitt statistisch überprüft.

4.3 Statistische Tests

Nach den Ergebnissen der Klassifikationen stellt sich die Frage, ob die Ergebnisse signifikant sind oder durch Zufall zustande kommen. Deswegen werden hier statistische Tests eingesetzt. Statistische Tests helfen, die Signifikanz der Ergebnisse zu berechnen. Damit können Eigenschaften der Klassifikationen festgestellt werden.

Ein statistischer Test ist eine Regel, die es erlaubt, für jedes Stichprobenergebnis eine Entscheidung zwischen der Null- und der Alternativhypothese zu treffen [4, S. 99]. Das statistische Testen wird in folgenden Schritten realisiert:

- Zunächst wird eine Hypothese aufgestellt. Diese Hypothese wird Null-Hypothese genannt. Meistens wird das Gegenteil, von dem was getestet wird, als Null-Hypothese genommen (z.B. 'Es gibt keine Unterschiede zwischen „Stacking“ und MAR').
- Dann wird eine geeignete statistische Test-Methode gewählt, um die Null-Hypothese zu testen.

- Es wird ein kritischer Bereich gewählt (p-Wert). Der p-Wert wird auch als „beobachtetes Signifikanzniveau“ bezeichnet. Er entspricht dem kleinsten Wert von α für den das Testergebnis gerade noch Signifikanz erreicht [2, S. 221].
- Wenn die berechnete Statistik für die Null-Hypothese im kritischen Bereich liegt, wird diese abgelehnt. Wenn die Null-Hypothese nicht abgelehnt wird, bedeutet dies jedoch nicht, dass sie bestätigt wird.
- Das Ablehnen der Null-Hypothese gibt die Sicherheit, dass die Ergebnisse nicht durch Zufall entstanden sind.

Das Hypothesen-Testen stellt aber keinen Beweis dar, dass die Beobachtungen gültig sind. Diese geben nur eine Hilfestellung, die nicht 100%ig sicher ist. Es gibt zwei statistische Tests:

- **Parametrische Tests** werden eingesetzt, wenn die Verteilung der Daten bekannt ist.
- **Nicht-Parametrische Tests** werden eingesetzt, wenn Informationen über die Verteilung der Daten nicht vorhanden sind.

Die Wahl geeigneter statistischer Test-Methoden hängt sehr stark vom Typ des untersuchten Problems ab. Für die hier untersuchten Probleme kann der Wilcoxon-Vorzeichen-Rang-Test gut angewandt werden, da die Aggregationsmethoden untereinander verglichen werden. Im nächsten Abschnitt wird auf diese Methode eingegangen und die Tests werden für die vorherigen Ergebnisse durchgeführt.

4.3.1 Wilcoxon-Vorzeichen-Rang-Test

Der Wilcoxon-Vorzeichen-Rang-Test ist ein nichtparametrischer statistischer Test. Er untersucht zwei gepaarte Stichproben auf die Gleichheit der Mittelwerte ihrer Verteilung. Er ergänzt den Vorzeichentest. Zudem berücksichtigt er nicht nur die Richtung der Differenzen, sondern auch die Höhe der Differenzen zwischen zwei gepaarten Stichproben [13, S. 233-235].

Der Wilcoxon-Test funktioniert wie folgt:

- Für jede Stichprobe wird die Performanz-Differenz der beiden Klassifikatoren berechnet.
- Die absoluten Werte der Differenzen werden geordnet und der Rang mit Vorzeichen markiert.
- Die Summen der positiven bzw. negativen Ränge werden berechnet. Diese Summen sind als W_{S1} bzw. W_{S2} gekennzeichnet.

- $T_{wilcox} = \min(W_{S1}, W_{S2})$
- T_{wilcox} wird mit dem kritischen Wert V_α . Wenn $V_\alpha \geq T_{wilcox}$, wird die Null-Hypothese abgelehnt.

4.3.2 Testen der Experimentergebnisse

Mit Hilfe des Wilcoxon-Tests soll nun untersucht werden, welche Aggregationsmethode gegenüber den anderen am besten ist. Dafür wird das Problem sukzessiv auf die Genres bzw. Instrumente fixiert. Dabei sind die Merkmalsgruppen und die Klassifikationsmethoden nicht von Bedeutung. Folgende Paarungen werden untersucht:

- „Statistiken“ vs. MAR(1)
- „Statistiken“ vs. DAR(1)
- „Stacking“ vs. MAR(1)
- „Stacking“ vs. DAR(1)
- MAR(1) vs. DAR(1)
- MARDAR(1) vs. „Statistiken“
- MARDAR(1) vs. „Statistiken“
- „Statistiken“ vs. Stacking

Die Aggregationsmethoden werden miteinander verglichen. Das Problem ist immer auf eine binäre Klassifikation aus den vorherigen Experimenten fixiert. Die Klassifikationsmethoden und Merkmalsgruppen werden dagegen variiert. Wenn zwei Aggregationsmethoden a und b verglichen werden, stellt u den Vektor mit allen Ergebnissen der Klassifikation für die Methode a dar und v den Vektor für die Methode b . Die Tests werden mit dem Wilcoxon-Vorzeichen-Rang-Test durchgeführt.

Seien \bar{u} und \bar{v} jeweils die Mittelwerte von u und v . Die Hypothesen sind dann wie folgt formuliert:

1. H_0 : \bar{u} und \bar{v} sind im Mittel gleich.
2. H_1 : \bar{u} und \bar{v} sind unterschiedlich.

Die folgenden vier Fälle werden dann unterschieden:

- H_0 wurden abgelehnt und $\bar{u} < \bar{v}$
- H_0 wurden abgelehnt und $\bar{u} > \bar{v}$
- H_0 kann nicht abgelehnt werden und $\bar{u} < \bar{v}$

- H_0 kann nicht abgelehnt werden und $\bar{u} > \bar{v}$

Alle statistischen Tests wurden im MATLAB durchgeführt. Die angewandte Funktion ist „*signrank*“. Diese Funktion berechnet den Wilcoxon-Vorzeichen-Rang-Test. Als Signifikanz-Wert wird der Default-Wert von $\alpha = 0,05$ benutzt.

„Statistiken“ vs. MAR(1)

Dieser Test vergleicht die Aggregationsmethode „Statistiken“ und MAR(1). Zudem ist u der Vektor mit allen Ergebnissen für die Aggregationsmethode „Statistiken“ und v der Vektor für MAR(1). Die Abbildungen 4.9 und 4.10 zeigen jeweils die statistischen Werte der beiden Aggregationsmethoden bei der Klassifikation von Genres und Instrumentenerkennung.

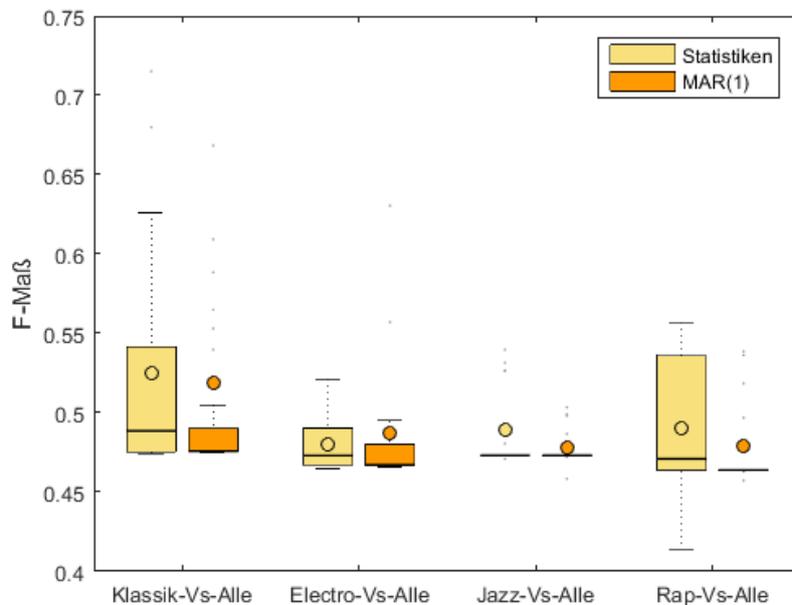


Abbildung 4.9: Boxplot der statistischen Verteilungen von „Statistiken“ und MAR(1) bei der Klassifikation von Genres

Die Tabelle 4.5 zeigt die Ergebnisse des Wilcoxon-Vorzeichen-Rang-Tests für die Aggregationsmethoden „Statistiken“ und MAR(1). Anhand dieser Ergebnisse können die folgenden Beobachtungen gemacht werden:

- In drei von vier Fällen (75%) bei der Klassifikation von Genres ist „Statistiken“ besser als MAR(1). Nur bei der Klassifikation „Jazz-Vs-All“ (25%) ist die Methode „Statistiken“ signifikant besser.

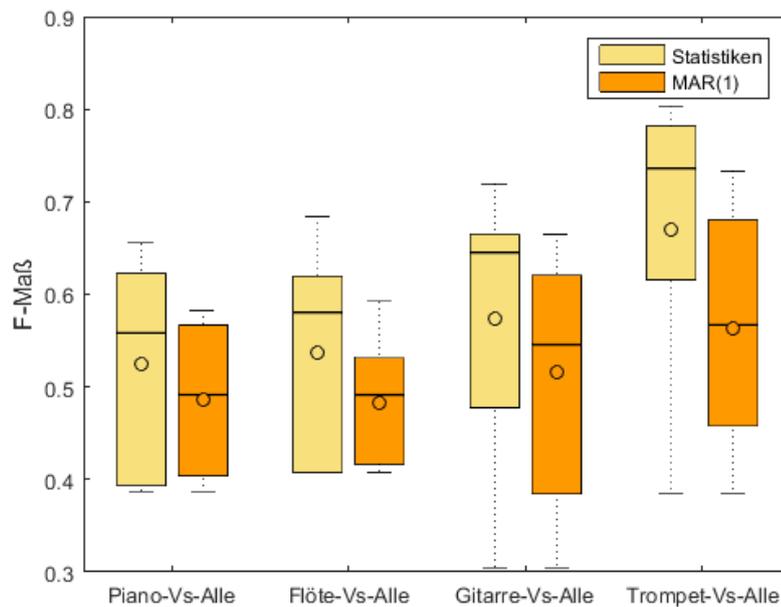


Abbildung 4.10: Boxplot der statistischen Verteilungen von „Statistiken“ und MAR(1) bei der Instrumentenerkennung

Tabelle 4.5: Statistischer Test zum Vergleich des F-Maßes bei „Statistiken“ und MAR(1)

	H_1 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} > \bar{v}$	H_1 und $\bar{u} > \bar{v}$
	↓ MAR(1) ist besser	↓ Beide sind vergleichbar	↓	↓ „Statistiken“ ist besser
Klassik	0	0	1	0
Electro	0	1	0	0
Jazz	0	0	0	1
Rap	0	0	1	0
Σ	0	1	2	1
Piano	0	0	1	0
Flöte	0	0	0	1
Gitarre	0	0	0	1
Trompet	0	0	0	1
Σ	0	0	1	3

- Bei der Erkennung von Instrumenten ist „Statistiken“ in allen vier Fällen (100%) besser als MAR(1). In 75% der Fälle ist „Statistiken“ signifikant besser als MAR(1).
- MAR(1) ist nur bei der Klassifikation „Electro-Vs-All“ besser als „Statistiken“. Aber dieses Ergebnis ist nicht signifikant.

- Insgesamt ist „Statistiken“ in vier von acht Fällen (50%) signifikant besser als MAR(1).

„Statistiken“ vs. DAR(1)

Dieser Test vergleicht die Aggregationsmethoden „Statistiken“ und DAR(1). Sei u der Vektor mit allen Ergebnissen für die Aggregationsmethode „Statistiken“ und v der Vektor für DAR(1). Die Abbildungen 4.11 und 4.12 zeigen jeweils die statistischen Werte der beiden Aggregationsmethoden bei der Klassifikation von Genres und Instrumentenerkennung.

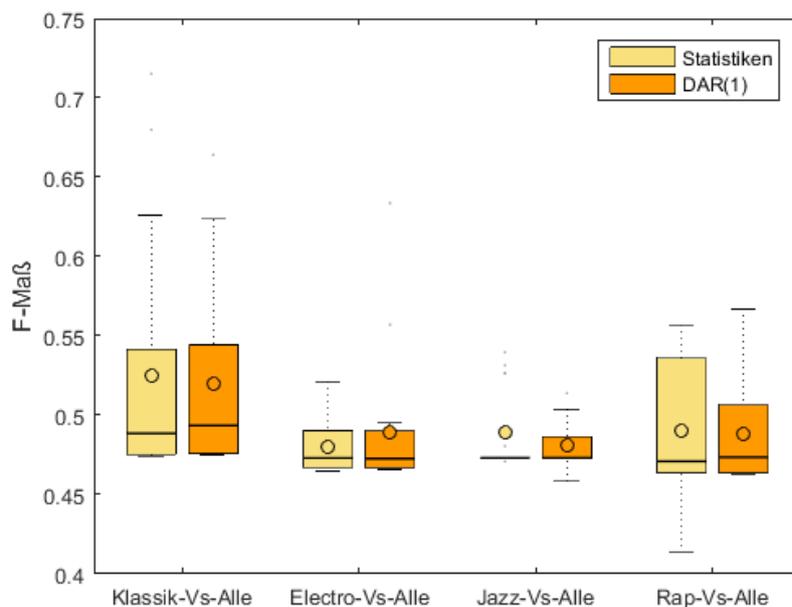


Abbildung 4.11: Boxplot der statistischen Verteilungen von „Statistiken“ und DAR(1) bei der Klassifikation von Genres

Die Tabelle 4.6 liefert die Ergebnisse des Wilcoxon-Vorzeichen-Rang-Tests für die Aggregationsmethoden „Statistiken“ und DAR(1). Aus diesen Ergebnissen können die folgenden Erkenntnisse gewonnen werden:

- In drei von vier Fällen (75%) bei der Klassifikation von Genres ist „Statistiken“ besser als DAR(1). Das ist allerdings in keinem Fall signifikant.
- Bei der Erkennung von Instrumenten ist „Statistiken“ in allen vier Fällen (100%) besser als DAR(1). In 100% der Fälle ist „Statistiken“ signifikant besser als DAR(1).
- DAR(1) ist nur bei der Klassifikation „Electro-Vs-All“ besser als „Statistiken“. Aber das Ergebnis ist nicht signifikant.

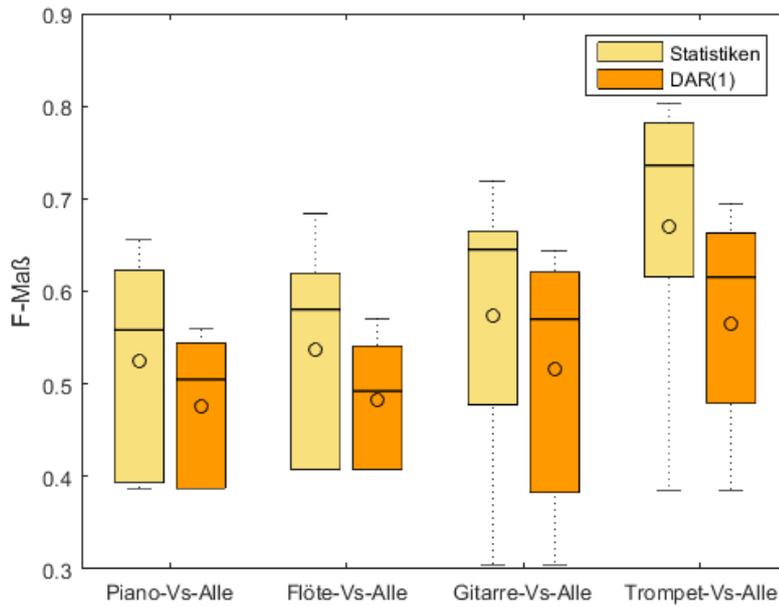


Abbildung 4.12: Boxplot der statistischen Verteilungen von „Statistiken“ und DAR(1) bei der Instrumentenerkennung

Tabelle 4.6: Statistischer Test zum Vergleich des F-Maßes bei „Statistiken“ und DAR(1)

	H_1 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} > \bar{v}$	H_1 und $\bar{u} > \bar{v}$
	↓ DAR(1) ist besser	↓ Beide sind vergleichbar	↓	↓ „Statistiken“ ist besser
Klassik	0	0	1	0
Electro	0	1	0	0
Jazz	0	0	1	0
Rap	0	0	1	0
Σ	0	1	3	0
Piano	0	0	0	1
Flöte	0	0	0	1
Gitarre	0	0	0	1
Trompet	0	0	0	1
Σ	0	0	0	4

„Stacking“ vs. MAR(1)

Dieser Test vergleicht die Aggregationsmethoden „Stacking“ und MAR(1). Sei u der Vektor mit allen Ergebnissen für die Aggregationsmethode „Stacking“ und v der Vektor für

MAR(1). Die Abbildungen 4.13 und 4.14 zeigen jeweils die statistischen Werte der beiden Aggregationsmethoden bei der Klassifikation von Genres und Instrumentenerkennung.

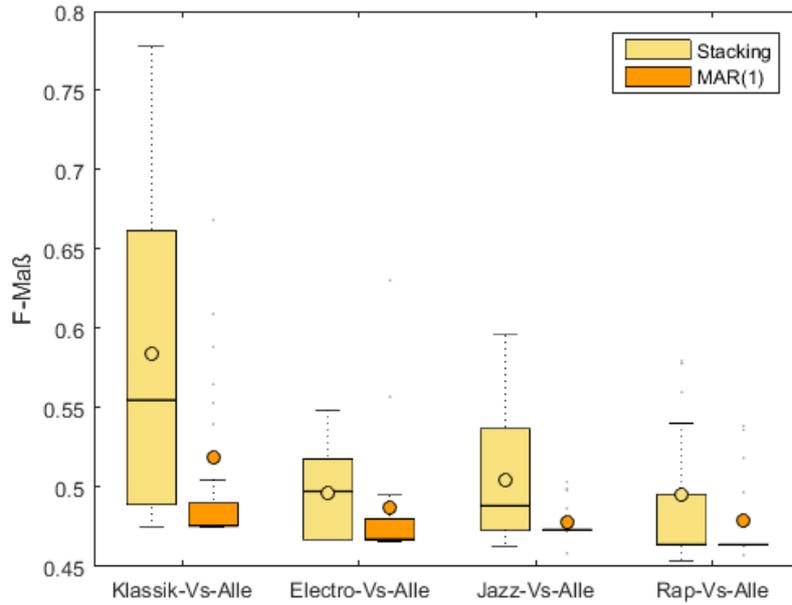


Abbildung 4.13: Boxplot der statistischen Verteilungen von „Stacking“ und MAR(1) bei der Klassifikation von Genres

Tabelle 4.7: Statistischer Test zum Vergleich des F-Maßes bei „Stacking“ und MAR(1)

	H_1 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} > \bar{v}$	H_1 und $\bar{u} > \bar{v}$
	↓	↓	↓	↓
	MAR(1) ist besser	Beide sind vergleichbar		„Stacking“ ist besser
Klassik	0	0	0	1
Electro	0	0	1	0
Jazz	0	0	0	1
Rap	0	0	1	0
Σ	0	0	2	2
Piano	0	0	0	1
Flöte	0	0	0	1
Gitarre	0	0	0	1
Trompet	0	0	0	1
Σ	0	0	0	4

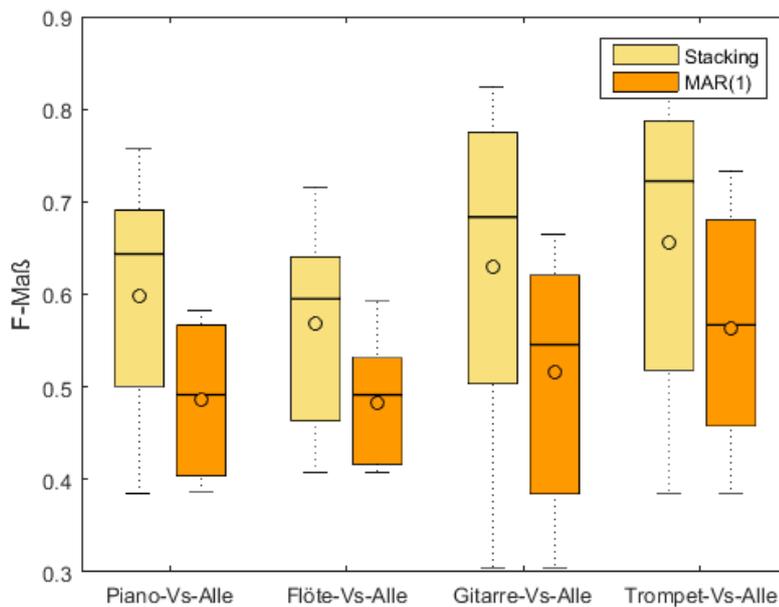


Abbildung 4.14: Boxplot der statistischen Verteilungen von „Stacking“ und MAR(1) bei der Instrumentenerkennung

Die Tabelle 4.7 zeigt die Ergebnisse des Wilcoxon-Vorzeichen-Rang-Tests für die Aggregationsmethoden „Stacking“ und MAR(1). Zu diesen Ergebnissen können die folgenden Beobachtungen gemacht werden:

- In vier von vier Fällen (100%) bei der Klassifikation von Genres ist „Stacking“ besser als MAR(1). Das ist allerdings nur in 50% der Fälle signifikant.
- Bei der Erkennung von Instrumenten ist „Stacking“ in allen vier Fällen (100%) besser als MAR(1). In 100% der Fälle ist „Stacking“ signifikant besser als MAR(1).
- MAR(1) ist in keinem der Fälle besser als „Stacking“.
- Es zeigt sich, dass „Stacking“ in sechs von acht Fällen (75%) signifikant besser als MAR(1) ist.

„Stacking“ vs. DAR(1)

Dieser Test vergleicht die Aggregationsmethoden „Stacking“ und DAR(1). Sei u der Vektor mit allen Ergebnissen für die Aggregationsmethode „Stacking“ und v der Vektor für DAR(1). Die Abbildungen 4.15 und 4.16 zeigen jeweils die statistischen Werte der beiden Aggregationsmethoden bei der Klassifikation von Genres und Instrumentenerkennung.

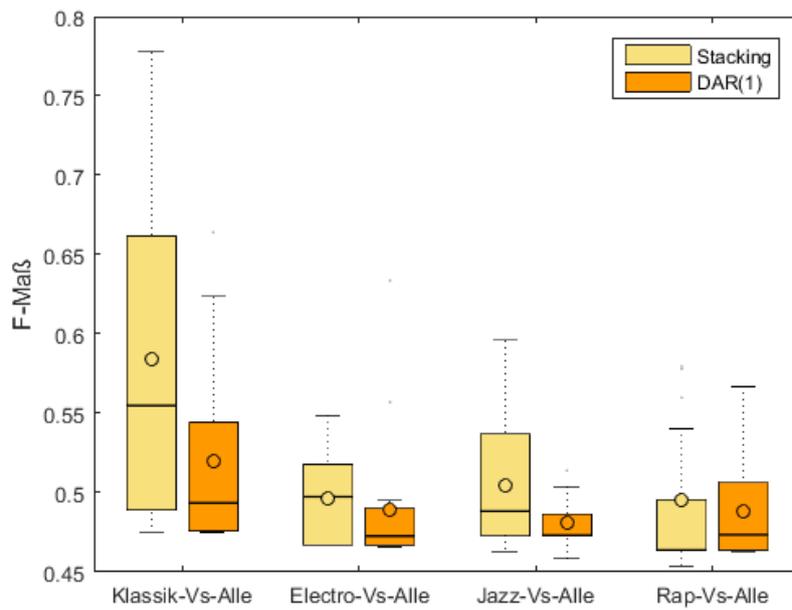


Abbildung 4.15: Boxplot der statistischen Verteilungen von „Stacking“ und DAR(1) bei der Klassifikation von Genres

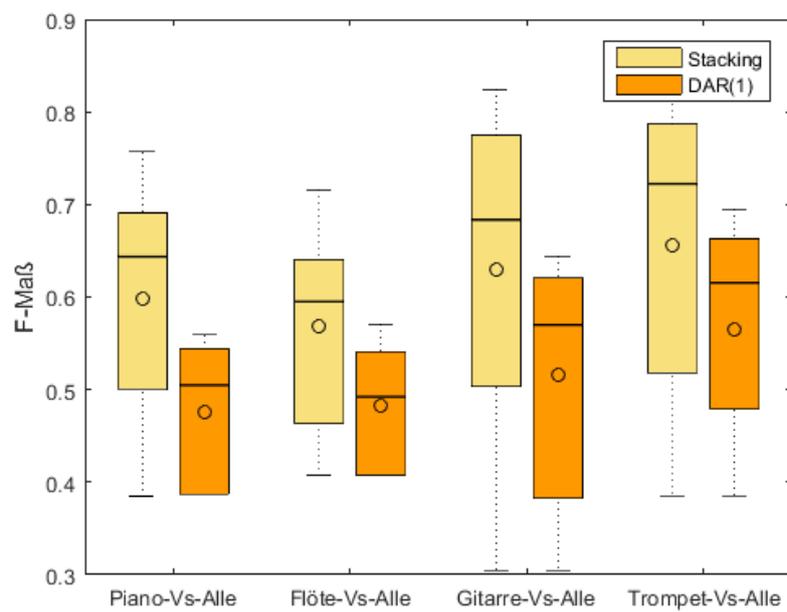


Abbildung 4.16: Boxplot der statistischen Verteilungen von „Stacking“ und DAR(1) bei der Instrumentenerkennung

Tabelle 4.8: Statistischer Test zum Vergleich des F-Maßes bei „Stacking“ und DAR(1)

	H_1 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} > \bar{v}$	H_1 und $\bar{u} > \bar{v}$
	↓ DAR(1) ist besser	↓ Beide sind vergleichbar	↓	↓ „Stacking“ ist besser
Klassik	0	0	0	1
Electro	0	0	1	0
Jazz	0	0	0	1
Rap	0	0	1	0
Σ	0	0	2	2
Piano	0	0	0	1
Flöte	0	0	0	1
Gitarre	0	0	0	1
Trompet	0	0	0	1
Σ	0	0	0	4

Die Tabelle 4.8 liefert die Ergebnisse des Wilcoxon-Vorzeichen-Rang-Tests für die Aggregationsmethoden „Stacking“ und DAR(1). Folgende Beobachtungen können festgestellt werden:

- In vier von vier Fällen (100%) bei der Klassifikation von Genres ist „Stacking“ besser als DAR(1). Das Ergebnis ist in 50% der Fälle signifikant.
- Bei der Erkennung von Instrumenten ist „Stacking“ in allen vier Fällen (100%) besser als DAR(1). In 100% der Fälle ist „Stacking“ signifikant besser als DAR(1).
- „Stacking“ ist insgesamt in sechs von acht Fällen (75%) signifikant besser als DAR(1).
- DAR(1) ist in keinem der Fälle besser als „Stacking“.

MAR(1) vs. DAR(1)

Dieser Test vergleicht die Aggregationsmethoden MAR(1) und DAR(1). Sei u der Vektor mit allen Ergebnissen für die Aggregationsmethode MAR(1) und v der Vektor für DAR(1). Die Abbildungen 4.17 und 4.18 zeigen jeweils die statistischen Werte der beiden Aggregationsmethoden bei der Klassifikation von Genres und Instrumentenerkennung.

Die Tabelle 4.9 zeigt die Ergebnisse des Wilcoxon-Vorzeichen-Rang-Tests für die Aggregationsmethoden MAR(1) und DAR(1). Zu diesen Ergebnissen können die folgenden Beobachtungen festgestellt werden:

- In vier von vier Fällen (100%) bei der Klassifikation von Genres ist DAR(1) besser als MAR(1). Das Ergebnis ist aber nur in einem Fall (25%) signifikant.

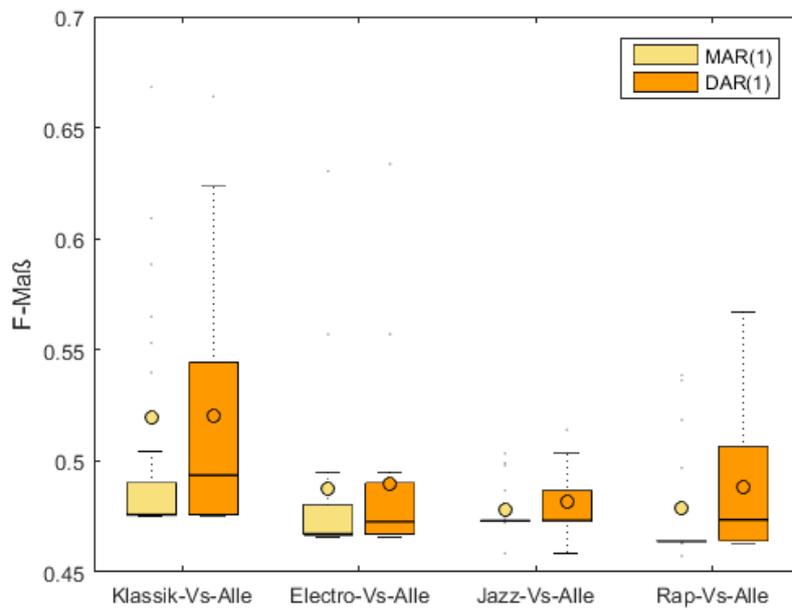


Abbildung 4.17: Boxplot der statistischen Verteilungen von MAR(1) und DAR(1) bei der Klassifikation von Genres

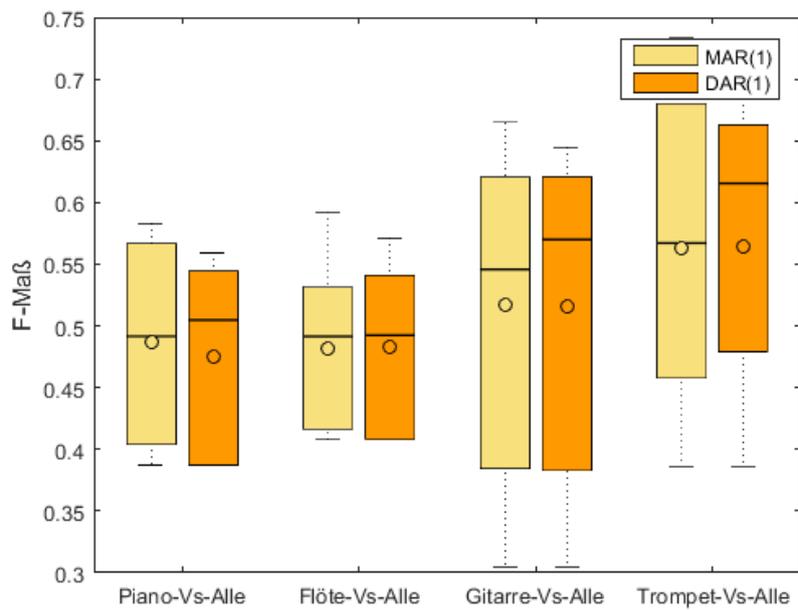


Abbildung 4.18: Boxplot der statistischen Verteilungen von MAR(1) und DAR(1) bei der Instrumentenerkennung

Tabelle 4.9: Statistischer Test zum Vergleich des F-Maßes bei MAR(1) und DAR(1)

	H_1 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} > \bar{v}$	H_1 und $\bar{u} > \bar{v}$
	↓ DAR(1) ist besser	↓ Beide sind vergleichbar	↓	↓ MAR(1) ist besser
Klassik	0	1	0	0
Electro	0	1	0	0
Jazz	0	1	0	0
Rap	1	0	0	0
Σ	1	3	0	0
Piano	0	0	1	0
Flöte	0	1	0	0
Gitarre	0	0	1	0
Trompet	0	1	0	0
Σ	0	2	2	0

- Bei der Erkennung von Instrumenten ist DAR(1) in zwei von vier Fällen (50%) besser als MAR(1). In keinem der Fälle ist das Ergebnis signifikant.
- In 50% der Fälle ist DAR(1) besser als MAR(1), aber auch hier in keinem der Fälle signifikant besser. Die beiden Aggregationsmethoden sind bei Erkennung von Instrumenten vergleichbar.
- Insgesamt ist DAR(1) in einem von acht Fällen (12,5%) signifikant besser als MAR(1).

MAR(1) vs. MARDAR(1)

Dieser Test vergleicht die Aggregationsmethoden MAR(1) und MARDAR(1). Sei u der Vektor mit allen Ergebnissen für die Aggregationsmethode MAR(1) und v der Vektor für MARDAR(1). Die Abbildungen 4.19 und 4.20 zeigen jeweils die statistischen Werte der beiden Aggregationsmethoden bei der Klassifikation von Genres und Instrumentenerkennung.

Die Tabelle 4.10 zeigt die Ergebnisse des Wilcoxon-Vorzeichen-Rang-Tests für die Aggregationsmethoden MAR(1) und MARDAR(1). Aus diesen Ergebnissen können die folgenden Beobachtungen festgestellt werden:

- In drei von vier Fällen (75%) bei der Klassifikation von Genres ist MAR(1) besser als MARDAR(1). Das Ergebnis ist aber in keinem Fall signifikant.
- Bei der Erkennung von Instrumenten ist MARDAR(1) in drei von vier Fällen (75%) besser als MAR(1). Das Ergebnis ist aber in keinem Fall signifikant.
- Die beiden Aggregationsmethoden sind vergleichbar.

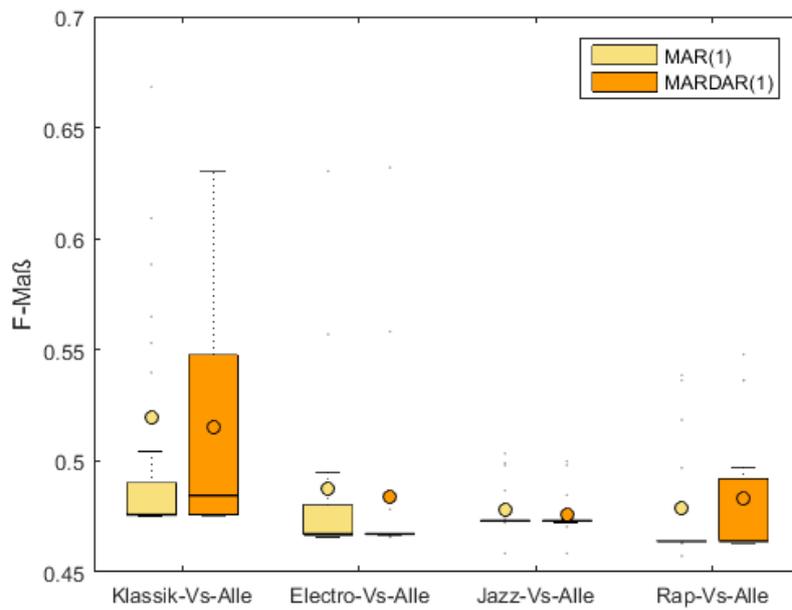


Abbildung 4.19: Boxplot der statistischen Verteilungen von MAR(1) und MARDAR(1) bei der Klassifikation von Genres

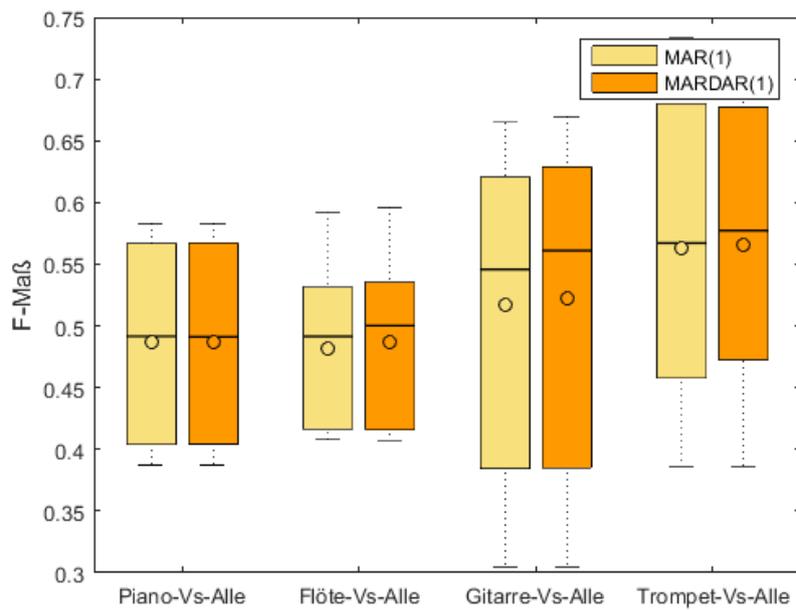


Abbildung 4.20: Boxplot der statistischen Verteilungen von MAR(1) und MARDAR(1) bei der Instrumentenerkennung

Tabelle 4.10: Statistischer Test zum Vergleich des F-Maßes bei MAR(1) und MARDAR(1)

	H_1 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} > \bar{v}$	H_1 und $\bar{u} > \bar{v}$
	↓ MARDAR ist besser	↓ Beide sind vergleichbar	↓	↓ MAR(1) ist besser
Klassik	0	0	1	0
Electro	0	0	1	0
Jazz	0	0	1	0
Rap	0	1	0	0
Σ	0	1	3	0
Piano	0	0	1	0
Flöte	0	1	0	0
Gitarre	0	1	0	0
Trompet	0	1	0	0
Σ	0	3	1	0

DAR(1) vs. MARDAR(1)

Dieser Test vergleicht die Aggregationsmethoden DAR(1) und MARDAR(1). Sei u der Vektor mit allen Ergebnissen für die Aggregationsmethode DAR(1) und v der Vektor für MARDAR(1). Die Abbildungen 4.21 und 4.22 zeigen jeweils die statistischen Werte der beiden Aggregationsmethoden bei der Klassifikation von Genres und Instrumentenerkennung.

Tabelle 4.11: Statistischer Test zum Vergleich des F-Maßes bei DAR(1) und MARDAR(1)

	H_1 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} > \bar{v}$	H_1 und $\bar{u} > \bar{v}$
	↓ MARDAR ist besser	↓ Beide sind vergleichbar	↓	↓ DAR(1) ist besser
Klassik	0	0	1	0
Electro	0	0	0	1
Jazz	0	0	0	1
Rap	0	0	1	0
Σ	0	0	2	2
Piano	0	1	0	0
Flöte	0	1	0	0
Gitarre	0	1	0	0
Trompet	0	1	0	0
Σ	0	4	0	0

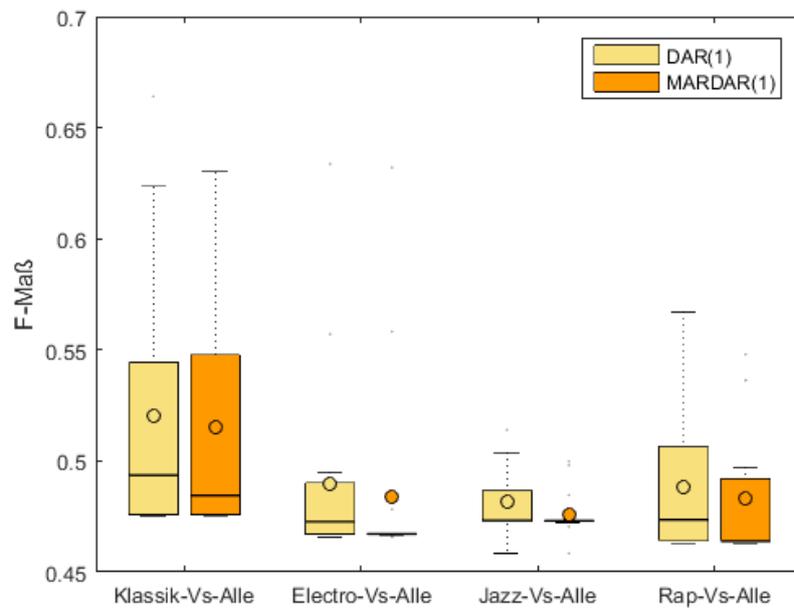


Abbildung 4.21: Boxplot der statistischen Verteilungen von DAR(1) und MARDAR(1) bei der Klassifikation von Genres

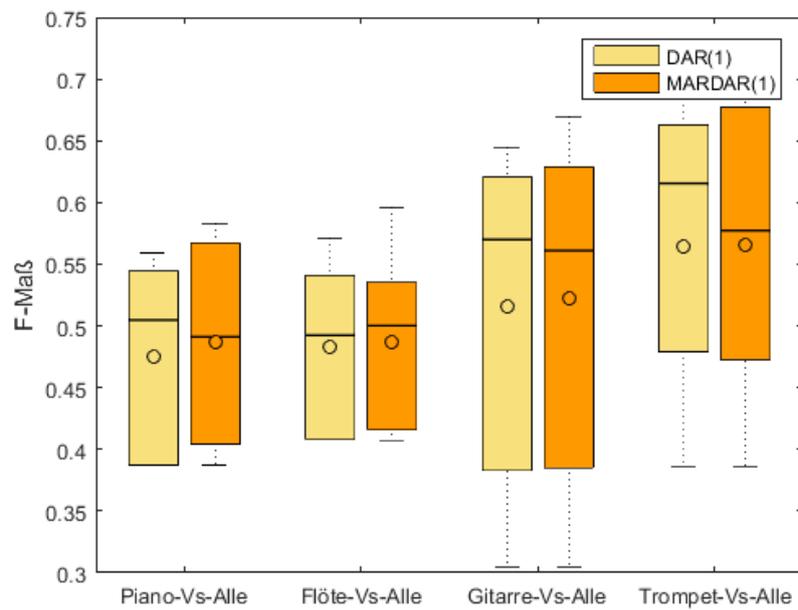


Abbildung 4.22: Boxplot der statistischen Verteilungen von DAR(1) und MARDAR(1) bei der Instrumentenerkennung

Die Tabelle 4.11 zeigt die Ergebnisse des Wilcoxon-Vorzeichen-Rang-Tests für die Aggregationsmethoden DAR(1) und MARDAR(1). Aus diesen Ergebnissen können die folgenden Erkenntnisse gewonnen werden:

- In vier von vier Fällen (100%) bei der Klassifikation von Genres ist DAR(1) besser als MARDAR(1). Das Ergebnis ist aber nur in 50% der Fälle signifikant.
- Bei der Erkennung von Instrumenten ist MARDAR(1) in allen vier Fällen (100%) besser als DAR(1). Das Ergebnis ist aber in keinem Fall signifikant.
- DAR(1) ist in zwei von acht Fällen (25%) signifikant besser als MARDAR(1).

MARDAR(1) vs. „Statistiken“

Dieser Test vergleicht die Aggregationsmethoden MARDAR(1) und „Statistiken“. Sei u der Vektor mit allen Ergebnissen für die Aggregationsmethode MARDAR(1) und v der Vektor für „Statistiken“. Die Abbildungen 4.23 und 4.24 zeigen jeweils die statistischen Werte der beiden Aggregationsmethoden bei der Klassifikation von Genres und Instrumentenerkennung.

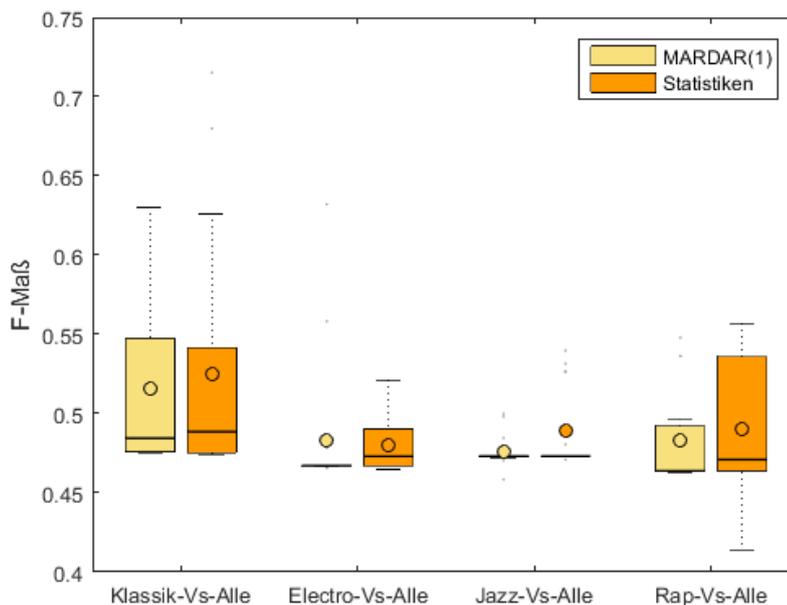


Abbildung 4.23: Boxplot der statistischen Verteilungen von MARDAR(1) und „Statistiken“ bei der Klassifikation von Genres

Die Tabelle 4.12 zeigt die Ergebnisse des Wilcoxon-Vorzeichen-Rang-Tests für die Aggregationsmethoden MARDAR(1) und „Statistiken“. Bezüglich dieser Ergebnisse können die folgenden Beobachtungen festgestellt werden:

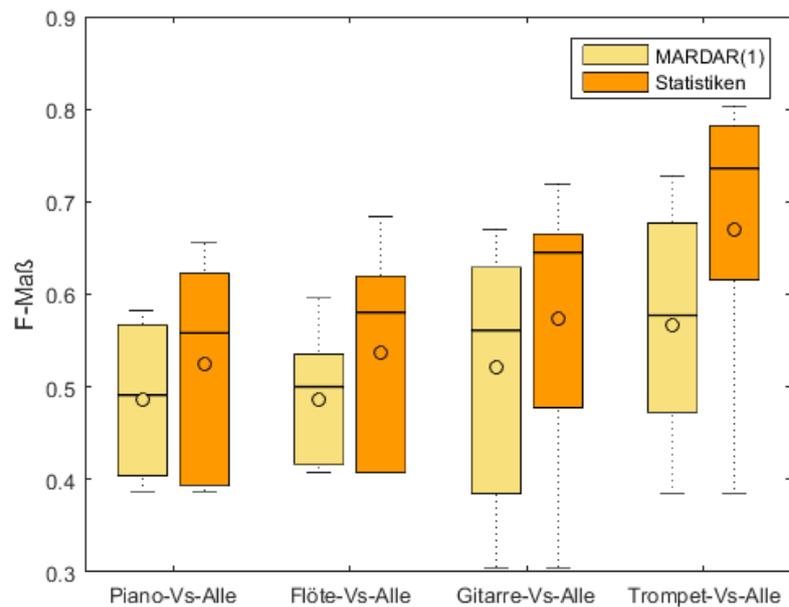


Abbildung 4.24: Boxplot der statistischen Verteilungen von MARDAR(1) und „Statistiken“ bei der Instrumentenerkennung

Tabelle 4.12: Statistischer Test zum Vergleich des F-Maßes bei MARDAR(1) und „Statistiken“

	H_1 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} > \bar{v}$	H_1 und $\bar{u} > \bar{v}$
	↓	↓	↓	↓
	Statistiken ist besser	Beide sind vergleichbar		MARDAR ist besser
Klassik	0	1	0	0
Electro	0	0	1	0
Jazz	1	0	0	0
Rap	0	1	0	0
Σ	1	2	1	0
Piano	0	1	0	0
Flöte	1	0	0	0
Gitarre	1	0	0	0
Trompet	1	0	0	0
Σ	3	1	0	0

- In drei von vier Fällen (75%) bei der Klassifikation von Genres ist „Statistiken“ besser als MARDAR(1). Das Ergebnis ist aber nur in 25% der Fälle signifikant.
- Bei der Erkennung von Instrumenten ist „Statistiken“ in allen vier Fällen (100%) besser als MARDAR(1). Das Ergebnis ist in 75% der Fälle signifikant.

- MARDAR(1) ist in einem von vier Fällen bei der Klassifikation von Genres besser als „Statistiken“. Aber in keinem der Fälle signifikant besser.
- „Statistiken“ ist insgesamt in vier von acht Fällen (50%) signifikant besser als MARDAR(1).

MARDAR(1) vs. „Stacking“

Dieser Test vergleicht die Aggregationsmethoden MARDAR(1) und „Stacking“. Sei u der Vektor mit allen Ergebnissen für die Aggregationsmethode MARDAR(1) und v der Vektor für „Stacking“. Die Abbildungen 4.25 und 4.26 zeigen jeweils die statistischen Werte der beiden Aggregationsmethoden bei der Klassifikation von Genres und Instrumentenerkennung.

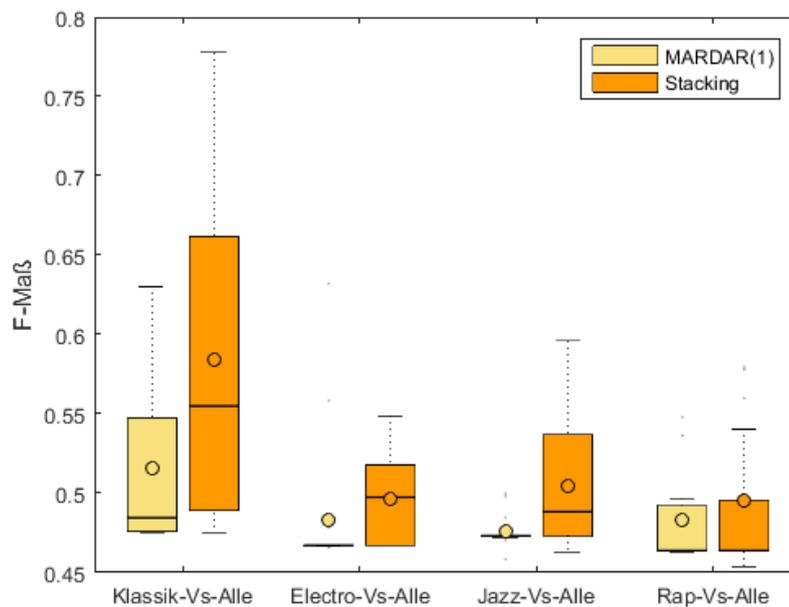


Abbildung 4.25: Boxplot der statistischen Verteilungen von MARDAR(1) und „Stacking“ bei der Klassifikation von Genres

Die Tabelle 4.13 zeigt die Ergebnisse des Wilcoxon-Vorzeichen-Rang-Tests für die Aggregationsmethoden MARDAR(1) und „Stacking“. Bezüglich dieser Ergebnisse können die folgenden Beobachtungen festgestellt werden:

- In vier von vier Fällen (100%) bei der Klassifikation von Genres ist „Stacking“ besser als MARDAR(1). Das Ergebnis ist in 75% der Fälle signifikant.
- Bei der Erkennung von Instrumenten ist „Stacking“ in allen vier Fällen (100%) besser als MARDAR(1). Das Ergebnis ist in 100% der Fälle signifikant.

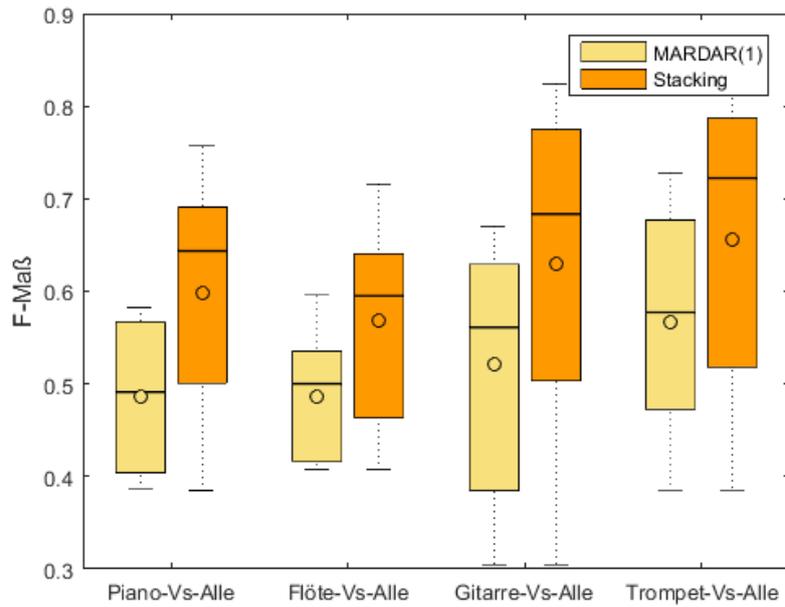


Abbildung 4.26: Boxplot der statistischen Verteilungen von MARDAR(1) und „Stacking“ bei der Instrumentenerkennung

Tabelle 4.13: Statistischer Test zum Vergleich des F-Maßes bei MARDAR(1) und „Stacking“

	H_1 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} > \bar{v}$	H_1 und $\bar{u} > \bar{v}$
	↓	↓	↓	↓
	Stacking ist besser	Beide sind vergleichbar		MARDAR ist besser
Klassik	1	0	0	0
Electro	1	0	0	0
Jazz	1	0	0	0
Rap	0	1	0	0
Σ	3	1	0	0
Piano	1	0	0	0
Flöte	1	0	0	0
Gitarre	1	0	0	0
Trompet	1	0	0	0
Σ	4	0	0	0

- MARDAR(1) ist in keinem der Fälle besser als „Stacking“.
- „Stacking“ ist insgesamt in sieben von acht Fällen (87,5%) signifikant besser als MARDAR(1).

„Statistiken“ vs. „Stacking“

Dieser Test vergleicht die Aggregationsmethoden „Statistiken“ und „Stacking“. Sei u der Vektor mit allen Ergebnissen für die Aggregationsmethode „Statistiken“ und v der Vektor für „Stacking“. Die Abbildungen 4.27 und 4.28 zeigen jeweils die statistischen Werte der beiden Aggregationsmethoden bei der Klassifikation von Genres und Instrumentenerkennung.

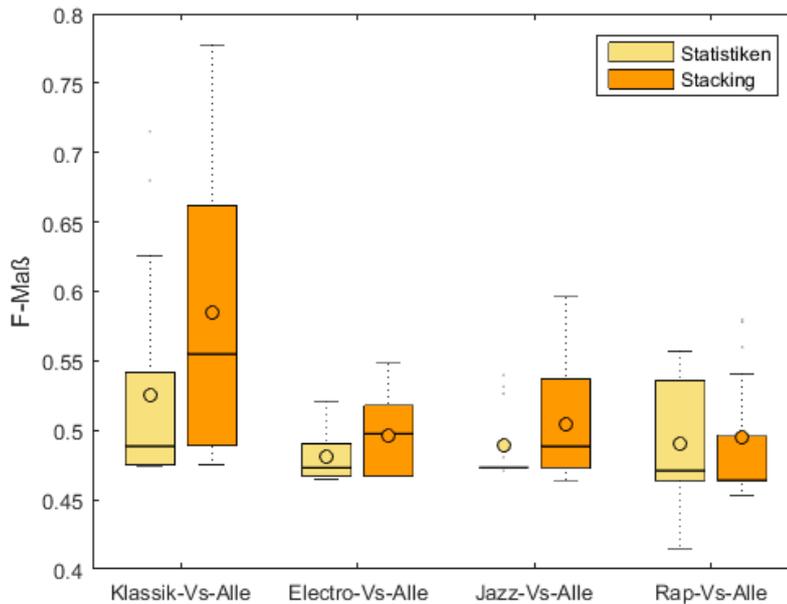


Abbildung 4.27: Boxplot der statistischen Verteilungen von „Statistiken“ und „Stacking“ bei der Klassifikation von Genres

Die Tabelle 4.14 zeigt die Ergebnisse des Wilcoxon-Vorzeichen-Rang-Tests für die Aggregationsmethoden „Statistiken“ und „Stacking“. Aus diesen Ergebnissen können die folgenden Beobachtungen festgestellt werden:

- In vier von vier Fällen (100%) bei der Klassifikation von Genres ist „Stacking“ besser als „Statistiken“. Das Ergebnis ist in 25% der Fälle signifikant.
- Bei der Erkennung von Instrumenten ist „Stacking“ in drei von vier Fällen (75%) besser als „Statistiken“. Das Ergebnis ist in 75% der Fälle signifikant.
- „Statistiken“ ist in keinem der Fälle besser als „Stacking“.
- Insgesamt ist „Stacking“ in vier von acht Fällen signifikant besser als „Statistiken“.

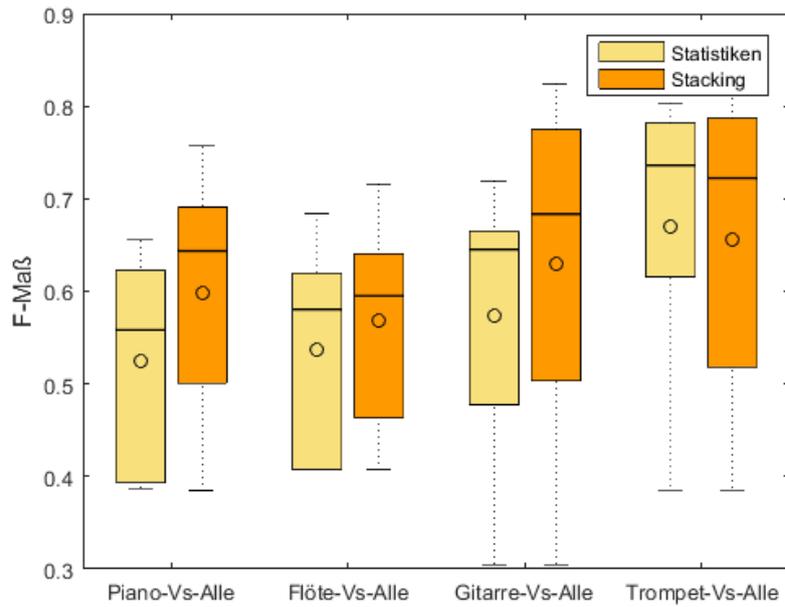


Abbildung 4.28: Boxplot der statistischen Verteilungen von „Statistiken“ und „Stacking“ bei der Instrumentenerkennung

Tabelle 4.14: Statistischer Test zum Vergleich des F-Maßes bei „Statistiken“ und „Stacking“

	H_1 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} > \bar{v}$	H_1 und $\bar{u} > \bar{v}$
	↓	↓	↓	↓
	Stacking ist besser	Beide sind vergleichbar		Statistiken ist besser
Klassik	1	0	0	0
Electro	0	1	0	0
Jazz	0	1	0	0
Rap	0	1	0	0
Σ	1	3	0	0
Piano	1	0	0	0
Flöte	1	0	0	0
Gitarre	1	0	0	0
Trompet	0	0	1	0
Σ	3	0	1	0

Zusammenfassung der statistischen Tests

Es zeigt sich, dass die Aggregationsmethode „Stacking“ insgesamt besser als alle anderen Methoden ist. Zudem zeichnet sich die Aggregationsmethode „Statistiken“ als zweitbeste Methode aus. Zusätzlich wird bei den autoregressiven Modellen keine klare Tendenz

festgestellt. DAR(1) ist in einem der acht Fällen (12,5%) signifikant besser als MAR(1) und in zwei der acht Fälle (25%) signifikant besser als MARDAR(1). MAR(1) und die kombinierte Methode MARDAR(1) sind vergleichbar.

Alle diese Aggregationsmethoden ergeben unterschiedliche Dimensionen. Im nächsten Abschnitt wird geklärt, ob die Dimension einen Einfluss auf die Performanz der jeweiligen Methode hat.

4.4 Dimensionsreduktion

Nach den ersten Ergebnissen und der Überprüfung durch statistische Tests ist es interessant zu untersuchen, ob sich die Hypothesen auch bestätigen, wenn die temporalen Aggregationsmethoden auf die gleiche Dimension reduziert sind. Es geht hierbei darum, den Einfluss der Dimension auf die Ergebnisse zu untersuchen.

Tabelle 4.15: Dimension nach Merkmalsgruppen

Merkmalsgruppen	Merkmale	Dimension
Zeitbereich	Zero-Crossing-Rate, Root Mean Square	2
Spektralbereich	Tristimulus, Spektral Kurtosis, Spectral Discrepancy	4
Cepstralbereich	MFCC	13
Chroma	Chroma	12

Die Tabelle 4.15 stellt die Dimension der verwendeten Merkmalsgruppen dar. Sei D die Dimension der Aggregationsmethoden. Basierend auf der Dimension der Merkmalsgruppe, ergibt sich die Dimension der Aggregationsmethoden wie folgt:

1. „Statistiken“

$$D = d(q + m) \quad (4.1)$$

2. „Stacking“

$$D = td \quad (4.2)$$

3. MAR(i)

$$D = d(id + 1) \quad (4.3)$$

4. DAR(i)

$$D = d(i + 1) \quad (4.4)$$

5. MARDAR(i)

$$D = d(id + i + 2) \quad (4.5)$$

wobei:

- d = Anzahl der Merkmale
- t = Texturfenster
- q = Anzahl der Quantils
- $m = 2$ min/max Werte aus (min,max,mean,var)

Tabelle 4.16: Dimension nach Aggregationsmethoden

Aggregationsmethode	Dimension für ein Texturfenster(t) von 88 Intervallen			
	Zeitbereich	Spektralbereich	Cepstralbereich	Chroma
„Statistiken“	10	20	65	60
„Stacking“	176	352	1144	1056
MAR(1)	6	20	182	156
DAR(1)	4	8	26	24
MARDAR(1)	10	28	208	180

Die Tabelle 4.16 zeigt die Dimensionen der Aggregationsmethoden für die vorherigen Experimente. Es können folgende Beobachtungen aufgelistet werden:

- Die Dimension von „Stacking“ ist verhältnismäßig groß im Vergleich zu den anderen Methoden.
- Die Dimensionen von „Statistiken“ und „Stacking“ lassen sich gut anpassen, wenn man die Anzahl der Quantile entsprechend der Dimension eines autoregressiven Modells anpasst.
- Eine autoregressive Methode kann gegen „Stacking“ und „Statistiken“ verglichen werden, wobei die Dimension von „Stacking“ und „Statistiken“ an das autoregressive Modell angepasst wird.

Für den Vergleich von „Stacking“ und „Statistiken“ gegen ein autoregressives Modell werden die Dimensionen wie folgt reduziert:

- Die Ziel-Dimension wird auf die Dimension des autoregressiven Modells fixiert.
- Für die Methode „Statistiken“ wird die Anzahl der Quantile variiert.
- Bei der Aggregationsmethode „Stacking“ wird die Dimension anhand der Methode „stratified uniform Stacking“ reduziert. Die Daten werden demzufolge in n möglichst gleich große Intervalle aufgeteilt. Aus jedem Intervall wird genau ein Wert gleich verteilt gezogen.

4.4.1 Reduktion anhand des multivariaten autoregressiven Modells

„Statistiken“ vs. MAR(1)

Dieser Test vergleicht die Aggregationsmethode „Statistiken“ und MAR(1). Sei u der Vektor mit allen Ergebnissen für die Aggregationsmethode „Statistiken“ und v der Vektor für MAR(1). Die Abbildungen 4.29 und 4.30 zeigen jeweils die statistischen Werte der beiden Aggregationsmethoden bei der Klassifikation von Genres und Instrumentenerkennung.

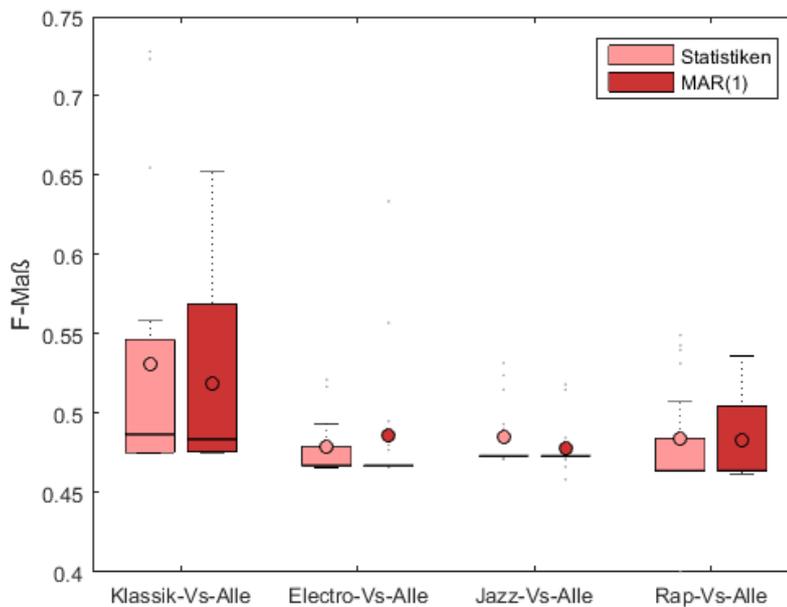


Abbildung 4.29: Boxplot der statistischen Verteilungen von „Statistiken“ und MAR(1) bei der Klassifikation von Genres

Die Tabelle 4.17 zeigt die Ergebnisse des Wilcoxon-Vorzeichen-Rang-Tests für die Aggregationsmethoden „Statistiken“ und MAR(1). Aus diesen Ergebnissen können die folgenden Beobachtungen festgestellt werden:

- In drei von vier Fällen (75%) bei der Klassifikation von Genres ist „Statistiken“ besser als MAR(1). Aber das Ergebnis ist in keinem der Fälle signifikant.
- Bei der Erkennung von Instrumenten ist „Statistiken“ in allen vier Fällen (100%) besser als MAR(1). In 100% der Fälle ist „Statistiken“ signifikant besser als MAR(1).

„Stacking“ vs. MAR(1)

Dieser Test vergleicht die Aggregationsmethoden „Stacking“ und MAR(1). Sei u der Vektor mit allen Ergebnissen für die Aggregationsmethode „Stacking“ und v der Vektor für

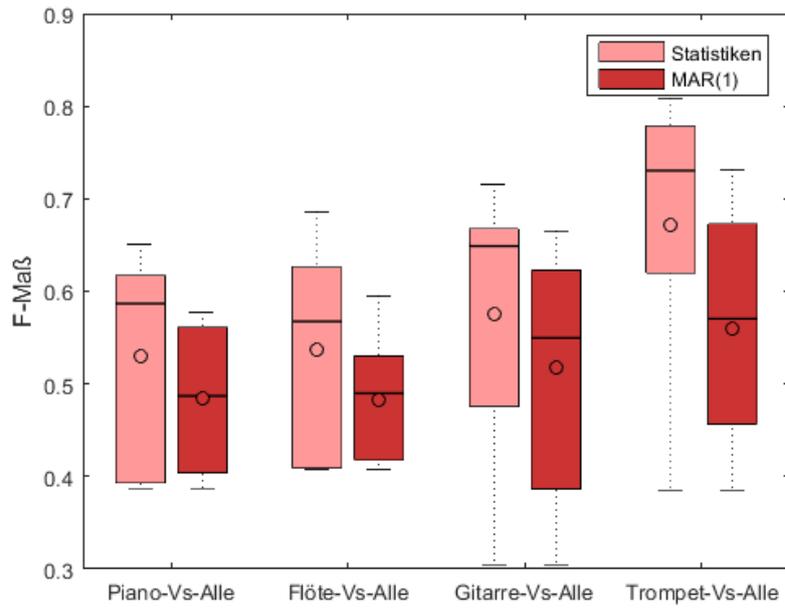


Abbildung 4.30: Boxplot der statistischen Verteilungen von „Statistiken“ und MAR(1) bei der Instrumentenerkennung

Tabelle 4.17: Statistischer Test zum Vergleich des F-Maßes bei „Statistiken“ und MAR(1)

	H_1 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} > \bar{v}$	H_1 und $\bar{u} > \bar{v}$
	↓ MAR(1) ist besser	↓ Beide sind vergleichbar	↓	↓ „Statistiken“ ist besser
Klassik	0	0	1	0
Electro	0	1	0	0
Jazz	0	0	1	0
Rap	0	0	1	0
Σ	0	1	3	0
Piano	0	0	0	1
Flöte	0	0	0	1
Gitarre	0	0	0	1
Trompet	0	0	0	1
Σ	0	0	0	4

MAR(1). Die Abbildungen 4.31 und 4.32 zeigen jeweils die statistischen Werte der beiden Aggregationsmethoden bei der Klassifikation von Genres und Instrumentenerkennung.

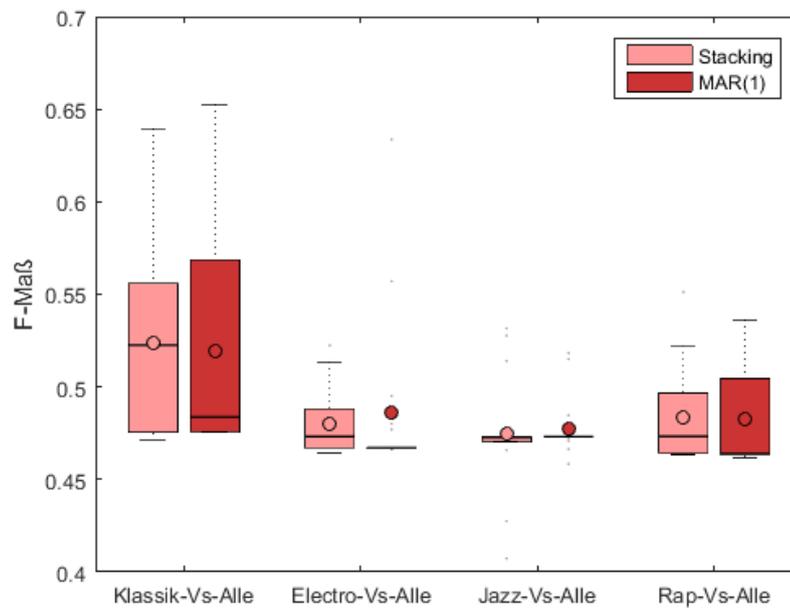


Abbildung 4.31: Boxplot der statistischen Verteilungen von „Stacking“ und MAR(1) bei der Klassifikation von Genres

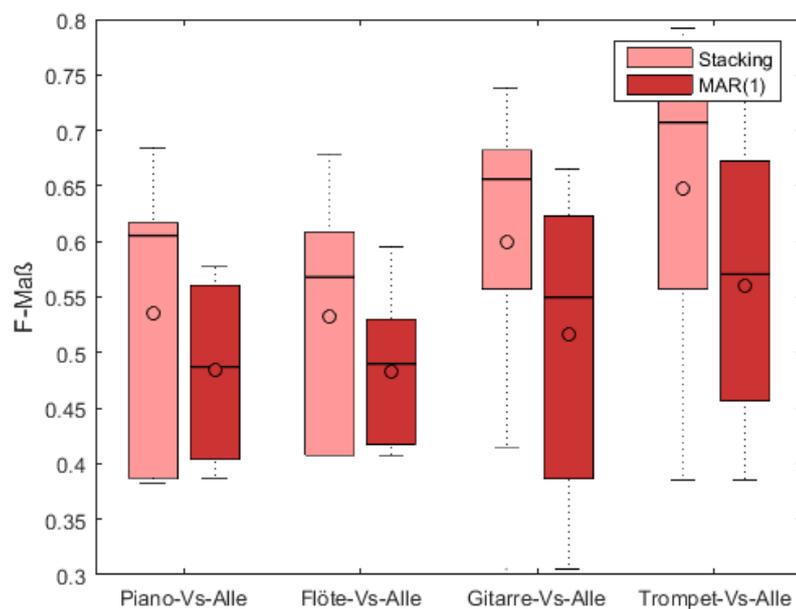


Abbildung 4.32: Boxplot der statistischen Verteilungen von „Stacking“ und MAR(1) bei der Instrumentenerkennung

Tabelle 4.18: Statistischer Test zum Vergleich des F-Maßes bei „Stacking“ und MAR(1)

	H_1 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} > \bar{v}$	H_1 und $\bar{u} > \bar{v}$
	↓ MAR(1) ist besser	↓ Beide sind vergleichbar	↓	↓ „Stacking“ ist besser
Klassik	0	0	1	0
Electro	0	1	0	0
Jazz	0	1	0	0
Rap	0	0	1	0
Σ	0	2	2	0
Piano	0	0	0	1
Flöte	0	0	0	1
Gitarre	0	0	0	1
Trompet	0	0	0	1
Σ	0	0	0	4

Die Tabelle 4.18 zeigt die Ergebnisse des Wilcoxon-Vorzeichen-Rang-Tests für die Aggregationsmethoden „Stacking“ und MAR(1). Aus diesen Ergebnissen können die folgenden Beobachtungen festgestellt werden:

- In zwei von vier Fällen (50%) bei der Klassifikation von Genres ist „Stacking“ besser als MAR(1). MAR(1) ist auch in 50% der Fälle besser als „Statistiken“. In keinem der Fälle ist das Ergebnis signifikant.
- Bei der Erkennung von Instrumenten ist „Stacking“ in allen vier Fällen (100%) besser als MAR(1). In 100% der Fälle ist „Statistiken“ signifikant besser als MAR(1).

„Statistiken“ vs. „Stacking“

Dieser Test vergleicht die Aggregationsmethoden „Statistiken“ und „Stacking“. Sei u der Vektor mit allen Ergebnissen für die Aggregationsmethode „Statistiken“ und v der Vektor für „Stacking“. Die Abbildungen 4.33 und 4.34 zeigen jeweils die statistischen Werte der beiden Aggregationsmethoden bei der Klassifikation von Genres und Instrumentenerkennung.

Die Tabelle 4.19 zeigt die Ergebnisse des Wilcoxon-Vorzeichen-Rang-Tests für die Aggregationsmethoden „Statistiken“ und „Stacking“. Aus diesen Ergebnissen können die folgenden Beobachtungen festgestellt werden:

- In drei von vier Fällen (75%) bei der Klassifikation von Genres ist „Statistiken“ besser als „Stacking“. Die Methode „Stacking“ ist nur in 25% der Fälle besser. Das Ergebnis ist in keinem der Fälle signifikant.

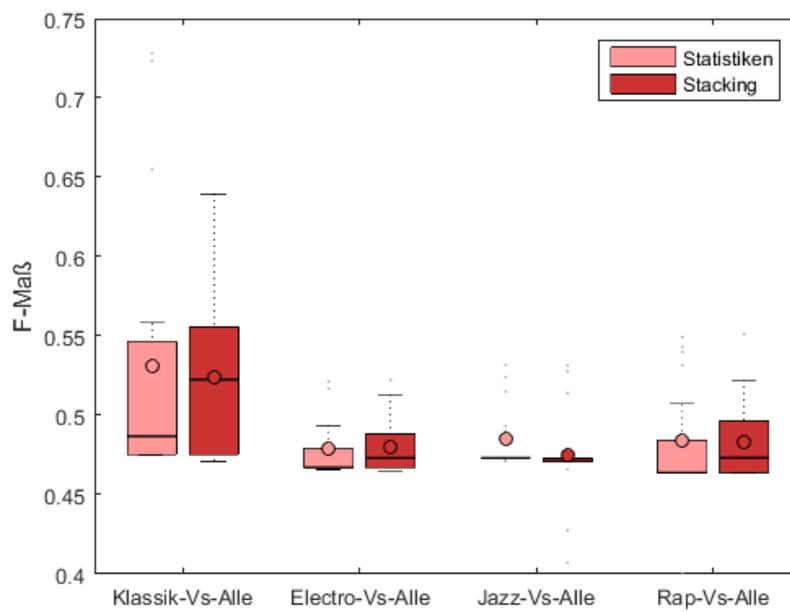


Abbildung 4.33: Boxplot der statistischen Verteilungen von „Statistiken“ und „Stacking“ bei der Klassifikation von Genres

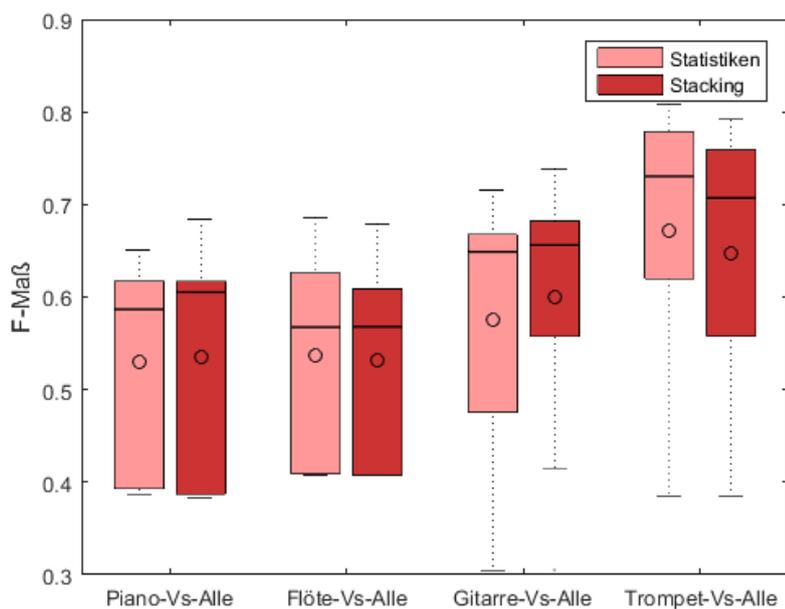


Abbildung 4.34: Boxplot der statistischen Verteilungen von „Statistiken“ und „Stacking“ bei der Instrumentenerkennung

Tabelle 4.19: Statistischer Test zum Vergleich des F-Maßes bei „Statistiken“ und „Stacking“

	H_1 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} > \bar{v}$	H_1 und $\bar{u} > \bar{v}$
	↓	↓	↓	↓
	Stacking ist besser	Beide sind vergleichbar		Statistiken ist besser
Klassik	0	0	1	0
Electro	0	1	0	0
Jazz	0	0	1	0
Rap	0	0	1	0
Σ	0	1	3	0
Piano	0	1	0	0
Flöte	0	0	1	0
Gitarre	1	0	0	0
Trompet	0	0	0	1
Σ	1	1	1	1

- Bei der Erkennung von Instrumenten ist „Statistiken“ in zwei von vier Fällen (100%) besser als „Stacking“. Das Ergebnis ist in 25% der Fälle signifikant. Die Methode „Stacking“ ist nur in 25% der Fälle signifikant besser als „Statistiken“.

4.4.2 Reduktion anhand des diagonal autoregressiven Modells

„Statistiken“ vs. DAR(1)

Dieser Test vergleicht die Aggregationsmethoden „Statistiken“ und DAR(1). Sei u der Vektor mit allen Ergebnissen für die Aggregationsmethode „Statistiken“ und v der Vektor für DAR(1). Die Abbildungen 4.35 und 4.36 zeigen jeweils die statistischen Werte der beiden Aggregationsmethoden bei der Klassifikation von Genres und Instrumentenerkennung.

Die Tabelle 4.20 zeigt die Ergebnisse des Wilcoxon-Vorzeichen-Rang-Tests für die Aggregationsmethoden „Statistiken“ und DAR(1). Anhand dieser Ergebnisse können die folgenden Beobachtungen gemacht werden:

- In drei von vier Fällen (75%) bei der Klassifikation von Genres ist „Statistiken“ besser als DAR(1). In keinem der Fälle ist das Ergebnis signifikant.
- Bei der Erkennung von Instrumenten ist „Statistiken“ in allen vier Fällen (100%) besser als DAR(1). In 100% der Fälle ist „Statistiken“ signifikant besser als DAR(1).

„Stacking“ vs. DAR(1)

Dieser Test vergleicht die Aggregationsmethoden „Stacking“ und DAR(1). Sei u der Vektor mit allen Ergebnissen für die Aggregationsmethode „Stacking“ und v der Vektor für

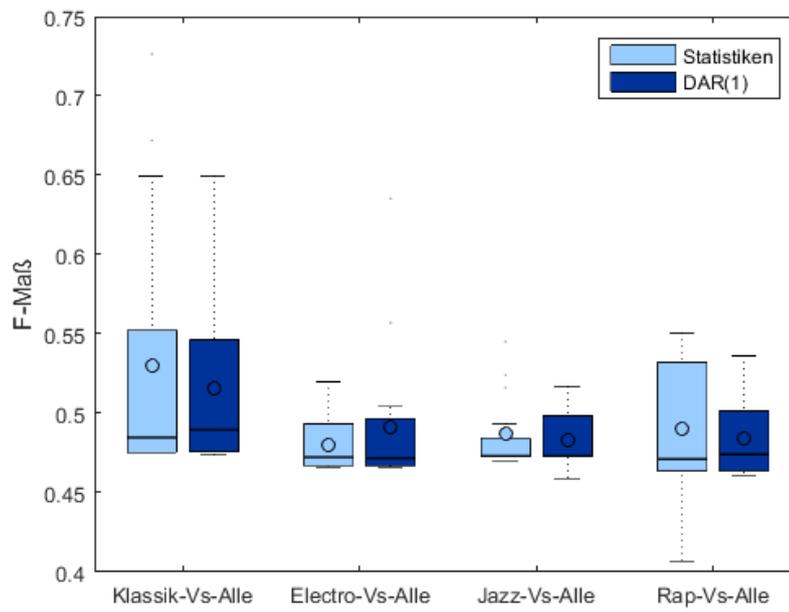


Abbildung 4.35: Boxplot der statistischen Verteilungen von „Statistiken“ und DAR(1) bei der Klassifikation von Genres

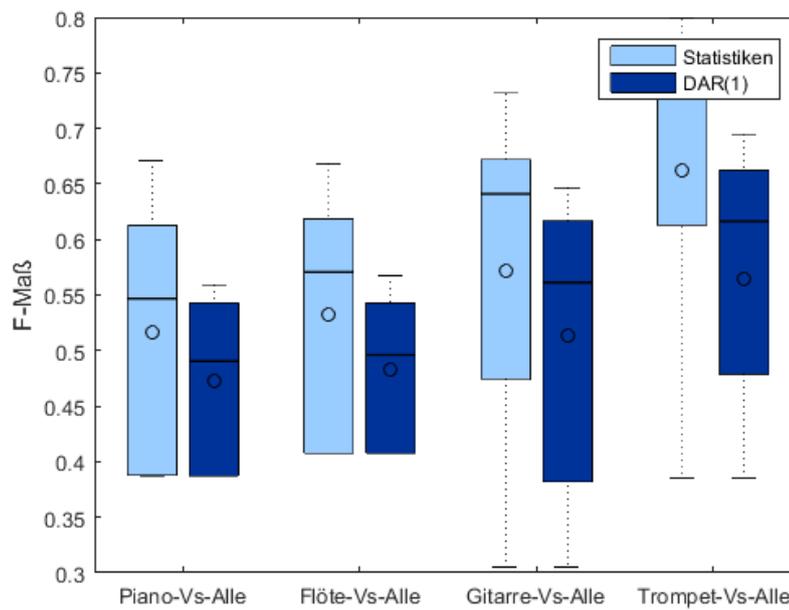
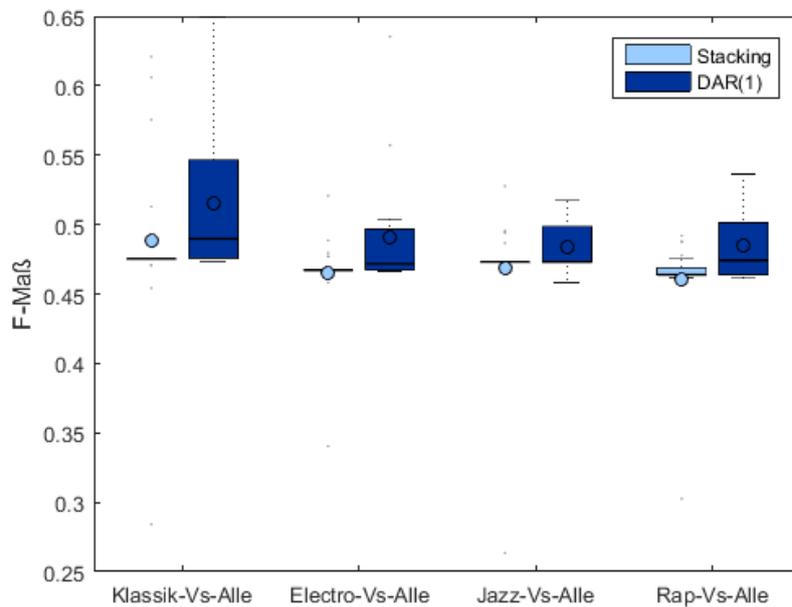


Abbildung 4.36: Boxplot der statistischen Verteilungen von „Statistiken“ und DAR(1) bei der Instrumentenerkennung

Tabelle 4.20: Statistischer Test zum Vergleich des F-Maßes bei „Statistiken“ und DAR(1)

	H_1 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} > \bar{v}$	H_1 und $\bar{u} > \bar{v}$
	↓ DAR(1) ist besser	↓ Beide sind vergleichbar	↓ Beide sind vergleichbar	↓ Statistiken ist besser
Klassik	0	0	1	0
Electro	0	1	0	0
Jazz	0	0	1	0
Rap	0	0	1	0
Σ	0	1	3	0
Piano	0	0	0	1
Flöte	0	0	0	1
Gitarre	0	0	0	1
Trompet	0	0	0	1
Σ	0	0	0	4

DAR(1). Die Abbildungen 4.37 und 4.38 zeigen jeweils die statistischen Werte der beiden Aggregationsmethoden bei der Klassifikation von Genres und Instrumentenerkennung.

**Abbildung 4.37:** Boxplot der statistischen Verteilungen von „Stacking“ und DAR(1) bei der Klassifikation von Genres

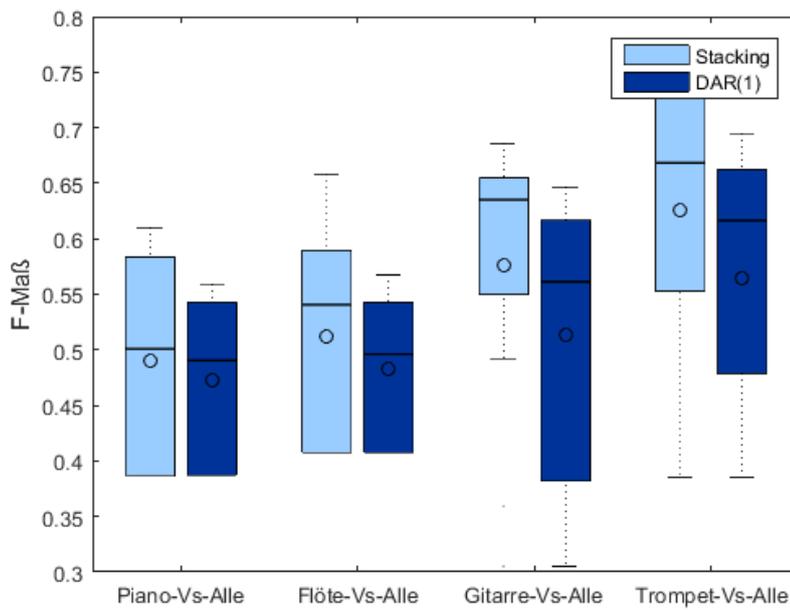


Abbildung 4.38: Boxplot der statistischen Verteilungen von „Stacking“ und DAR(1) bei der Instrumentenerkennung

Tabelle 4.21: Statistischer Test zum Vergleich des F-Maßes bei „Stacking“ und DAR(1)

	H_1 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} > \bar{v}$	H_1 und $\bar{u} > \bar{v}$
	↓ DAR(1) ist besser	↓ Beide sind vergleichbar	↓	↓ Stacking ist besser
Klassik	0	1	0	0
Electro	0	1	0	0
Jazz	0	1	0	0
Rap	0	1	0	0
Σ	0	4	0	0
Piano	0	0	1	0
Flöte	0	0	1	0
Gitarre	0	0	0	1
Trompet	0	0	0	1
Σ	0	0	2	2

Die Tabelle 4.21 zeigt die Ergebnisse des Wilcoxon-Vorzeichen-Rang-Tests für die Aggregationsmethoden „Stacking“ und DAR(1). Anhand dieser Ergebnisse können die folgenden Beobachtungen gemacht werden:

- In vier von vier Fällen (100%) bei der Klassifikation von Genres ist DAR(1) besser als „Statistiken“. Das Ergebnis ist allerdings in keinem Fall signifikant.
- Bei der Erkennung von Instrumenten ist „Stacking“ in allen vier Fällen (100%) besser als DAR(1). In 50% der Fälle ist „Stacking“ signifikant besser als DAR(1).

„Statistiken“ vs. „Stacking“

Dieser Test vergleicht die Aggregationsmethoden „Statistiken“ und „Stacking“. Sei u der Vektor mit allen Ergebnissen für die Aggregationsmethode „Statistiken“ und v der Vektor für „Stacking“. Die Abbildungen 4.39 und 4.40 zeigen jeweils die statistischen Werte der beiden Aggregationsmethoden bei der Klassifikation von Genres und Instrumentenerkennung.

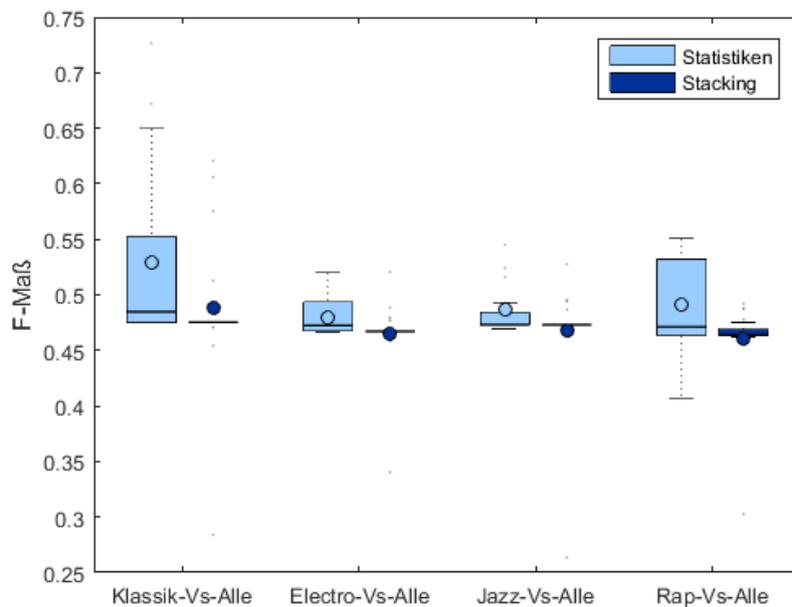


Abbildung 4.39: Boxplot der statistischen Verteilungen von „Statistiken“ und „Stacking“ bei der Klassifikation von Genres

Die Tabelle 4.22 zeigt die Ergebnisse des Wilcoxon-Vorzeichen-Rang-Tests für die Aggregationsmethoden „Statistiken“ und „Stacking“. Aus diesen Ergebnissen können die folgenden Beobachtungen festgestellt werden:

- In drei von vier Fällen (75%) bei der Klassifikation von Genres ist „Stacking“ besser als „Statistiken“. Das Ergebnis ist in 25% der Fälle signifikant.
- Bei der Erkennung von Instrumenten ist „Stacking“ in drei von vier Fällen (75%) besser als „Statistiken“. Das Ergebnis ist in 75% der Fälle signifikant.

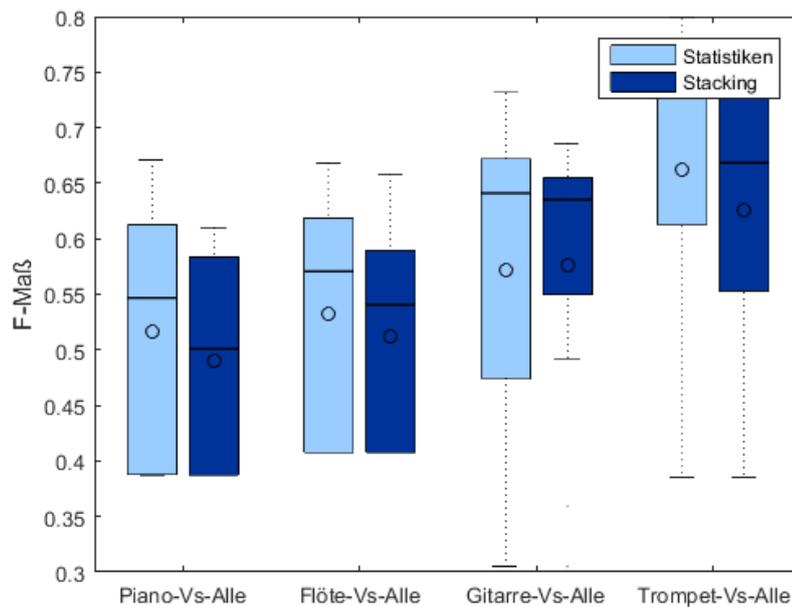


Abbildung 4.40: Boxplot der statistischen Verteilungen von „Statistiken“ und „Stacking“ bei der Instrumentenerkennung

Tabelle 4.22: Statistischer Test zum Vergleich des F-Maßes bei Statistiken und Stacking

	H_1 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} < \bar{v}$	H_0 und $\bar{u} > \bar{v}$	H_1 und $\bar{u} > \bar{v}$
	↓	↓	↓	↓
	Stacking ist besser	Beide sind vergleichbar		Statistiken ist besser
Klassik	0	0	0	1
Electro	0	0	1	0
Jazz	0	0	1	0
Rap	0	0	0	1
Σ	0	0	2	2
Piano	0	0	0	1
Flöte	0	0	1	0
Gitarre	0	1	0	0
Trompet	0	0	0	1
Σ	0	1	1	2

Zusammenfassung der statistischen Tests nach der Dimensionsreduktion

Es zeigt sich nun, dass die Aggregationsmethode „Statistiken“ insgesamt besser als alle anderen Methoden ist. „Statistiken“ ist in fünf der acht Fälle (62,5%) besser als „Stacking“ nach der Kompression anhand MAR(1), was eine deutliche Veränderung darstellt. Aber

das Ergebnis ist nur in einem der acht Fälle (12,5%) signifikant. Vor der Reduktion war das Ergebnis nur in einem der Fälle (12,5%) besser als „Stacking“ und in keinem der Fälle signifikant besser. Bei der Kompression nach DAR(1) ist „Statistiken“ sogar in sieben von acht Fällen (87,5%) besser als „Stacking“. Das Ergebnis ist in vier von acht Fällen (50%) signifikant. Zudem bleibt „Statistiken“, wie vor der Dimensionsreduktion, in sieben von acht Fällen (87,5%) besser als MAR(1) und DAR(1). Das Ergebnis bleibt auch in vier von acht Fällen (50%) signifikant.

Weiterhin zeichnet sich die Aggregationsmethode „Stacking“ als zweitbeste Methode aus. „Stacking“ ist nach der Dimensionsreduktion nur in sechs der acht Fälle (75%) besser als MAR(1). Das Ergebnis ist nur in vier der acht Fälle (50%) signifikant. Vor der Dimensionsreduktion war „Stacking“ in allen Fällen (100%) besser als MAR(1), und in 75% der Fälle signifikant besser. Zudem ist „Stacking“ nach der Dimensionsreduktion nur in vier von acht Fällen (50%) besser als DAR(1) und in zwei der acht Fälle signifikant besser. Vor der Dimensionsreduktion war „Stacking“ in allen Fällen (100%) besser als DAR(1) und in 75% der Fälle signifikant besser. Es wurde festgestellt, dass „Stacking“ nach der Dimensionsreduktion besser als MAR(1) und DAR(1) bleibt. Allerdings ist es nicht mehr sehr deutlich.

Kapitel 5

Zusammenfassung und Ausblick

Für die Klärung der Bedeutung verlaufspezifischer Merkmale bei Klassifikationsproblemen auf Musiksignalen wurden in dieser Arbeit Experimente durchgeführt und ausgewertet. Hierzu wurden als Klassifikationsprobleme die Klassifikation von Genres und die Erkennung von Instrumenten festgelegt. Als Aggregationsmethoden wurden „Statistiken“, „Stacking“, MAR, DAR und das kombinierte MARDAR untersucht. Zudem wurden vier Klassifikatoren GMM, SVM, k -NN und RF aus den Bereichen Statistik, Mathematik und Informatik benutzt.

Vor der Durchführung der Experimente mussten die Audiomerkmale berechnet werden. Für die Klassifikation von Genres wurden 610 Titel bearbeitet. Bei der Erkennung von Instrumenten wurden 3000 Titel benutzt. Zudem wurden insgesamt jeweils sieben Merkmale pro Titel und 25270 Dateien bearbeitet. Die Ergebnisse der Vorverarbeitung waren aggregierte Vektoren nach den Methoden „Statistiken“, „Stacking“, MAR, DAR, und das kombinierte MARDAR.

Bereits in den zwei ersten Experimenten hat sich gezeigt, dass die Klassifikationen mit „Stacking“ am genauesten waren. Als zweitbeste Aggregationsmethode zeichnete sich die Methode „Statistiken“ aus. Bei den restlichen autoregressiven Modellen war keine Tendenz erkennbar. Diese Ergebnisse wurden dann anhand von statistischen Tests überprüft. Hierbei wurde der Wilcoxon-Rang-Test benutzt. Die Aggregationsmethode „Stacking“ war meistens besser als die anderen Methoden. Im Vergleich zur Methode „Statistiken“ war sie in vier von acht Fällen (50%) signifikant besser. Sie war sogar in sechs von acht Fällen (75%) signifikant besser als MAR und DAR. Weiterhin war „Stacking“ in sieben von acht Fällen (87,5%) besser als MARDAR. Die Methode „Statistiken“ war in vier von acht Fällen (50%) signifikant besser als MAR und DAR. Zudem war sie in vier von acht Fällen (50%) signifikant besser als MARDAR. Die Ergebnisse der statistischen Tests bestätigen die ersten Erkenntnisse.

Da die Aggregationsmethoden sehr unterschiedliche Dimensionen haben, ging es im nächsten Schritt darum, die Dimension zu reduzieren und neue Experimente durchzuführen.

Damit sollte der Einfluss der Dimension auf die Ergebnisse untersucht und überprüft werden, ob sich nicht andere Methoden nach einer Dimensionsreduktion als genauer erweisen. Eine Analyse der Dimensionen ergab, dass die Dimensionen von „Statistiken“ und „Stacking“ gut anpassbar sind. Diese wurden auf die Dimensionen der autoregressiven Modelle reduziert. Anschließend konnte ein erneuter Vergleich von autoregressiven Modellen (MAR, DAR oder das kombinierte MARDAR) mit „Statistiken“ und „Stacking“ vorgenommen werden. Nach der Reduktion der Dimension war nun „Statistiken“ besser als die anderen Aggregationsmethoden. Die Methode „Stacking“ war die zweitbeste. Allerdings war die Methode „Stacking“ nicht mehr deutlich besser als MAR(1) und DAR(1).

Ausblicke

Durch diese Studie konnte ein direkter Vergleich von Aggregationsmethoden bei Klassifikationsproblemen auf Musiksignalen durchgeführt werden. Es gibt aber verschiedene Aspekte bei diesen Vergleichen, die noch berücksichtigt werden können. Diese Aspekte stellen weiterführende Ideen zu Verbesserungen oder für Erweiterungen der Ansätze dieser Arbeit.

Zum Beispiel können diese Untersuchungen auf andere Klassifikationsprobleme, wie die Klassifikation von Subgenres, die Erkennung von Emotionen oder die Erkennung von Umgebungsgeräuschen erweitert werden. Dort können die hier erzielten Ergebnisse noch einmal überprüft werden. Ein weiterer Aspekt wäre die Auswahl der Audiomerkmale. Wie gut funktionieren die Aggregationsmethoden mit anderen Merkmalen? Zudem können auch eventuell die Auswahl der Merkmale mit einer automatischen Merkmalsauswahl-Algorithmus erfolgen. Und daraufhin kann die Untersuchung der Methoden anhand der besten Merkmale durchgeführt werden.

Weiterhin könnte die gleiche Studie auf Basis anderer Datensätze durchgeführt werden. Für die Nutzung von anderen Datensätzen müssten aber im Vorfeld die Nutzungsrechte geklärt werden, da es nicht möglich ist, mit beliebigen kommerziellen Musikstücken zu experimentieren. Damit könnten die Erkenntnisse aus dieser Studie noch überprüft werden. Zudem waren die Berechnungen der temporalen Merkmalsvektoren sehr zeitaufwendig. An dieser Stelle könnte versucht werden, diese noch zu optimieren. Ein anderer Aspekt ist die Entwicklung einer GUI-Anwendung für die Berechnung der temporalen Merkmale. Auch eine Integration in das AMUSE-Framework wäre auch sicherlich von Nutzen. Damit würde ein komplettes Tool für unterschiedliche Kombinationen aus Problemen und Merkmalsgruppen erstellt werden können für weitere Vergleiche der Aggregationsmethoden.

Darüber hinaus könnte auch untersucht werden, ob bestimmte Aggregationsmethoden besser in Kombination mit bestimmten Klassifikationsmethoden oder Merkmalsgruppen funktionieren. Dieser Punkt wäre sicherlich interessant für weitere Untersuchungen der Aggregationsmethoden.

Anhang A

Klassifikation-Ergebnisse

Folgende Abkürzungen wurden verwendet: Merkmalskategorie (MK), Aggregationsmethode (AM), Texturfenster (TF).

Tabelle A.1: Klassifikationsergebnisse für „Klassik-Vs-Alle“

MK	AM (Rang)	TF	F-Maß(%)			
			<i>k</i> -NN	SVM	GMM	RF
ZB	„Statistiken“	88	67,99%	47,55%	47,55%	71,52%
	„Stacking“	88	68,78%	47,55%	47,55%	69,62%
	MAR(1)	88	60,93%	47,59%	47,59%	66,84%
	DAR(1)	88	48,74%	47,59%	47,59%	66,41%
	MARDAR(1)	88	55,56%	47,59%	47,59%	63,02%
	MAR(2)	88	54,1%	47,59%	47,59%	58,87%
	DAR(2)	88	51,83%	47,59%	47,59%	61,38%
	MARDAR(2)	88	50,25%	47,59%	47,59%	61,38%
	MAR(3)	88	55,72%	47,59%	47,59%	58,15%
	DAR(3)	88	49,39%	47,59%	47,59%	62,84%
SB	MARDAR(3)	88	53,41%	47,59%	47,59%	62,84%
	„Statistiken“	88	53,97%	49,31%	47,55%	62,6%
	„Stacking“	88	53,02%	77,53%	50,28%	77,77%
	MAR(1)	88	56,5%	47,59%	47,59%	55,31%
	DAR(1)	88	54,24%	47,59%	49,41%	62,35%
	MARDAR(1)	88	52,52%	47,55%	47,59%	60,37%
	MAR(2)	88	47,5%	47,59%	47,5%	53,68%
	DAR(2)	88	47,55%	47,59%	47,5%	52,17%
	MARDAR(2)	88	47,59%	47,59%	47,5%	53,97%
MAR(3)	88	50,92%	47,55%	47,5%	51,02%	

Fortsetzung auf der nächsten Seite...

Tabelle A.1: Klassifikationsergebnisse für „Klassik-Vs-Alle“ (Fortsetzung)

MK	AM (Rang)	TF	F-Maß(%)			
			<i>k</i> -NN	SVM	GMM	RF
CB	DAR(3)	88	51,02%	47,59%	47,5%	52,42%
	MARDAR(3)	88	54,42%	49,31%	49,24%	53,97%
	„Statistiken“	88	54,39%	47,55%	48,78%	49,24%
	„Stacking“	88	58,16%	63,6%	52,3%	47,55%
	MAR(1)	88	50,44%	47,5%	53,98%	47,55%
	DAR(1)	88	52,3%	49,31%	54,63%	50,82%
	MARDAR(1)	88	50,62%	49,31%	53,98%	47,55%
	MAR(2)	88	50,72%	47,41%	53,24%	47,55%
	DAR(2)	88	50,62%	47,55%	53,02%	47,5%
	MARDAR(2)	88	50,92%	49,24%	52,32%	47,55%
	MAR(3)	88	49,17%	47,41%	49,28%	47,55%
	DAR(3)	88	49,17%	47,55%	49,65%	47,55%
	MARDAR(3)	88	49,17%	49,02%	49,55%	47,55%
	Ch	„Statistiken“	88	47,37%	47,55%	48,21%
„Stacking“		88	56,05%	54,94%	62,78%	47,55%
MAR(1)		88	47,59%	47,59%	58,85%	47,59%
DAR(1)		88	47,59%	47,59%	58,43%	47,5%
MARDAR(1)		88	47,59%	47,59%	58,85%	47,59%
MAR(2)		88	47,59%	47,14%	48,01%	47,59%
DAR(2)		88	47,59%	47,59%	49,35%	47,55%
MARDAR(2)		88	47,59%	47,23%	49,35%	47,59%
MAR(3)		88	47,59%	49,24%	48,85%	47,59%
DAR(3)		88	47,59%	47,59%	49,24%	47,59%
MARDAR(3)	88	47,59%	47,45%	48,01%	47,59%	

Tabelle A.2: Klassifikationsergebnisse für „Electro-Vs-Alle“

MK	AM (Rang)	TF	F-Maß(%)			
			<i>k</i> -NN	SVM	GMM	RF
ZB	„Statistiken“	88	47,95%	46,72%	46,72%	52,13%
	„Stacking“	88	54,82%	50,41%	46,72%	51,42%
	MAR(1)	88	46,58%	46,72%	46,72%	49,08%
	DAR(1)	88	47,95%	46,72%	46,72%	48,93%
	MARDAR(1)	88	46,67%	46,72%	46,72%	47,82%

Fortsetzung auf der nächsten Seite...

Tabelle A.2: Klassifikationsergebnisse für „Electro-Vs-Alle“ (Fortsetzung)

MK	AM (Rang)	TF	F-Maß(%)				
			<i>k</i> -NN	SVM	GMM	RF	
	MAR(2)	88	46,67%	46,72%	46,72%	46,72%	
	DAR(2)	88	46,72%	46,72%	46,72%	46,53%	
	MARDAR(2)	88	46,67%	46,72%	46,72%	47,95%	
	MAR(3)	88	46,63%	46,72%	46,72%	46,63%	
	DAR(3)	88	46,58%	46,72%	46,72%	46,53%	
	MARDAR(3)	88	46,58%	46,72%	46,72%	46,72%	
	SB	„Statistiken“	88	46,63%	46,68%	46,72%	49,96%
		„Stacking“	88	46,72%	52,15%	46,72%	47,95%
		MAR(1)	88	48,08%	46,67%	49,51%	46,67%
		DAR(1)	88	46,58%	46,72%	49,51%	46,58%
		MARDAR(1)	88	46,58%	46,67%	46,72%	46,58%
		MAR(2)	88	46,67%	46,72%	48,01%	46,67%
DAR(2)		88	48,01%	46,72%	48,01%	46,63%	
MARDAR(2)		88	46,58%	46,72%	48,01%	46,67%	
MAR(3)		88	49,4%	46,72%	46,72%	46,72%	
DAR(3)		88	46,72%	46,72%	48,01%	46,72%	
MARDAR(3)	88	48,08%	46,67%	48,01%	46,72%		
CB	„Statistiken“	88	47,95%	46,72%	51,38%	46,49%	
	„Stacking“	88	47,95%	52,32%	49,09%	46,72%	
	MAR(1)	88	46,58%	47,95%	63,04%	46,68%	
	DAR(1)	88	46,63%	46,72%	63,37%	49,09%	
	MARDAR(1)	88	46,63%	46,68%	63,21%	46,72%	
	MAR(2)	88	47,95%	50,32%	60,9%	46,72%	
	DAR(2)	88	47,89%	46,72%	61,18%	49,25%	
	MARDAR(2)	88	46,63%	47,83%	60,77%	46,72%	
	MAR(3)	88	48,08%	46,54%	64,32%	46,72%	
	DAR(3)	88	49,25%	46,72%	63,98%	48,02%	
MARDAR(3)	88	47,89%	48,94%	63,5%	46,72%		
Ch	„Statistiken“	88	48,86%	46,72%	49,2%	47,89%	
	„Stacking“	88	50,5%	52,85%	51,28%	46,72%	
	MAR(1)	88	46,72%	46,72%	55,7%	46,67%	
	DAR(1)	88	47,82%	46,72%	55,7%	47,82%	
	MARDAR(1)	88	46,72%	46,72%	55,83%	46,72%	
	MAR(2)	88	46,72%	50,91%	48,91%	46,72%	

Fortsetzung auf der nächsten Seite...

Tabelle A.2: Klassifikationsergebnisse für „Electro-Vs-Alle“ (Fortsetzung)

MK	AM (Rang)	TF	F-Maß(%)			
			<i>k</i> -NN	SVM	GMM	RF
	DAR(2)	88	47,76%	46,72%	49,26%	48,08%
	MARDAR(2)	88	46,72%	52,62%	50,35%	46,72%
	MAR(3)	88	46,58%	46,44%	46,85%	46,72%
	DAR(3)	88	48,01%	46,72%	48,19%	46,72%
	MARDAR(3)	88	49,16%	46,53%	50,49%	46,72%

Tabelle A.3: Klassifikationsergebnisse für „Jazz-Vs-Alle“

MK	AM (Rang)	TF	F-Maß(%)			
			<i>k</i> -NN	SVM	GMM	RF
ZB	„Statistiken“	88	47,09%	47,32%	47,32%	53,15%
	„Stacking“	88	52,89%	47,32%	47,32%	53,28%
	MAR(1)	88	48,67%	47,32%	47,32%	50,34%
	DAR(1)	88	50,16%	47,32%	47,32%	48,47%
	MARDAR(1)	88	48,47%	47,32%	47,32%	47,23%
	MAR(2)	88	47,14%	47,32%	47,32%	51,5%
	DAR(2)	88	48,67%	47,32%	47,32%	53,14%
	MARDAR(2)	88	48,74%	47,32%	47,32%	51,39%
	MAR(3)	88	50,07%	47,32%	47,32%	47,23%
	DAR(3)	88	47,18%	47,32%	47,32%	47,18%
SB	MARDAR(3)	88	47,09%	47,32%	47,32%	50,07%
	„Statistiken“	88	52,65%	47,32%	47,32%	47,32%
	„Stacking“	88	48,81%	54,64%	49,82%	48,88%
	MAR(1)	88	47,23%	47,23%	49,81%	47,27%
	DAR(1)	88	47,14%	47,32%	49,9%	50,34%
	MARDAR(1)	88	47,04%	47,27%	49,81%	47,27%
	MAR(2)	88	47,04%	47,27%	47,23%	47,32%
	DAR(2)	88	47,18%	47,32%	47,27%	47,27%
	MARDAR(2)	88	47,23%	47,27%	47,23%	47,32%
	MAR(3)	88	47,18%	48,88%	47,23%	47,32%
CB	DAR(3)	88	47,14%	47,32%	47,23%	47,32%
	MARDAR(3)	88	47,23%	47,32%	47,23%	47,32%
	„Statistiken“	88	53,97%	47,32%	48,06%	47,32%
	„Stacking“	88	47,32%	54,91%	47,09%	47,32%

Fortsetzung auf der nächsten Seite...

Tabelle A.3: Klassifikationsergebnisse für „Jazz-Vs-Alle“ (Fortsetzung)

MK	AM (Rang)	TF	F-Maß(%)				
			<i>k</i> -NN	SVM	GMM	RF	
	MAR(1)	88	49,9%	47,28%	47,39%	47,32%	
	DAR(1)	88	51,4%	47,32%	47,47%	47,32%	
	MARDAR(1)	88	49,99%	47,23%	47,55%	47,32%	
	MAR(2)	88	48,81%	48,88%	46,35%	47,32%	
	DAR(2)	88	48,88%	47,28%	46,35%	47,32%	
	MARDAR(2)	88	48,81%	47,23%	46,01%	47,32%	
	MAR(3)	88	48,47%	47,19%	45,54%	47,32%	
	DAR(3)	88	51,4%	47,32%	45,68%	47,32%	
	MARDAR(3)	88	48,61%	48,74%	45,68%	47,32%	
	Ch	„Statistiken“	88	52,65%	47,32%	49,17%	47,32%
		„Stacking“	88	54,11%	59,6%	46,3%	47,32%
		MAR(1)	88	47,32%	47,32%	45,84%	47,32%
		DAR(1)	88	48,67%	47,32%	45,84%	47,23%
		MARDAR(1)	88	47,27%	47,32%	45,84%	47,32%
MAR(2)		88	47,32%	46,9%	45,26%	47,32%	
DAR(2)		88	47,23%	47,32%	46,2%	47,32%	
MARDAR(2)		88	47,32%	47,14%	45,45%	47,32%	
MAR(3)		88	47,32%	48,88%	47,7%	47,32%	
DAR(3)		88	47,32%	47,32%	51,48%	47,32%	
MARDAR(3)	88	47,32%	48,81%	51,01%	47,32%		

Tabelle A.4: Klassifikationsergebnisse für „Rap-Vs-Alle“

MK	AM (Rang)	TF	F-Maß(%)			
			<i>k</i> -NN	SVM	GMM	RF
ZB	„Statistiken“	88	55,71%	46,4%	46,4%	53,52%
	„Stacking“	88	46,35%	55,99%	46,4%	48,42%
	MAR(1)	88	45,72%	46,39%	46,39%	48,12%
	DAR(1)	88	47,98%	46,39%	46,39%	48,64%
	MARDAR(1)	88	47,03%	46,39%	46,39%	48,72%
	MAR(2)	88	48,19%	46,39%	46,39%	56,19%
	DAR(2)	88	49,44%	46,39%	46,39%	51,17%
	MARDAR(2)	88	47,21%	46,39%	46,39%	50,62%
	MAR(3)	88	47,15%	46,39%	46,39%	50,04%

Fortsetzung auf der nächsten Seite...

Tabelle A.4: Klassifikationsergebnisse für „Rap-Vs-Alle“ (Fortsetzung)

MK	AM (Rang)	TF	F-Maß(%)			
			<i>k</i> -NN	SVM	GMM	RF
SB	DAR(3)	88	48,41%	46,39%	46,39%	51,69%
	MARDAR(3)	88	48,27%	46,39%	46,39%	50,91%
	„Statistiken“	88	55,44%	46,35%	46,4%	53,64%
	„Stacking“	88	46,4%	54,04%	50,72%	46,35%
	MAR(1)	88	46,39%	46,34%	49,69%	46,39%
	DAR(1)	88	47,21%	46,39%	49,69%	47,51%
	MARDAR(1)	88	48,63%	46,34%	49,69%	46,39%
	MAR(2)	88	46,11%	47,64%	46,3%	46,34%
	DAR(2)	88	47,98%	46,39%	46,3%	46,25%
	MARDAR(2)	88	47,21%	46,39%	46,3%	46,39%
	MAR(3)	88	49,53%	46,34%	46,3%	46,39%
	DAR(3)	88	46,11%	46,39%	46,3%	46,34%
MARDAR(3)	88	48,48%	47,64%	46,3%	46,39%	
CB	„Statistiken“	88	51,18%	46,4%	41,43%	47,52%
	„Stacking“	88	46,4%	57,96%	46,16%	46,4%
	MAR(1)	88	46,3%	46,3%	53,86%	46,4%
	DAR(1)	88	46,26%	46,4%	53,86%	46,4%
	MARDAR(1)	88	46,26%	46,35%	53,64%	46,4%
	MAR(2)	88	47,16%	46,3%	52,53%	46,4%
	DAR(2)	88	47,22%	46,35%	52,42%	46,4%
	MARDAR(2)	88	47,34%	47,46%	53,18%	46,4%
	MAR(3)	88	46,11%	47,46%	54,96%	46,4%
	DAR(3)	88	45,92%	46,35%	54,84%	46,4%
	MARDAR(3)	88	46,07%	47,52%	55,44%	46,4%
	Ch	„Statistiken“	88	53,87%	46,4%	46,68%
„Stacking“		88	51,69%	57,8%	45,34%	46,4%
MAR(1)		88	51,85%	46,39%	53,63%	46,39%
DAR(1)		88	56,69%	46,39%	53,12%	51,58%
MARDAR(1)		88	54,79%	46,39%	53,63%	46,39%
MAR(2)		88	47,96%	46,11%	50,99%	46,39%
DAR(2)		88	51,74%	46,39%	47,78%	48,63%
MARDAR(2)		88	48,99%	48,19%	49,16%	46,39%
MAR(3)		88	48,12%	46,3%	51,13%	46,39%
DAR(3)		88	46,01%	46,39%	48,18%	46,39%

Fortsetzung auf der nächsten Seite...

Tabelle A.4: Klassifikationsergebnisse für „Rap-Vs-Alle“ (Fortsetzung)

MK	AM (Rang)	TF	F-Maß(%)			
			<i>k</i> -NN	SVM	GMM	RF
	MARDAR(3)	88	49,28%	46,39%	50,59%	46,39%

Tabelle A.5: Klassifikationsergebnisse für „Piano-Vs-Alle“

MK	AM (Rang)	TF	F-Maß(%)			
			<i>k</i> -NN	SVM	GMM	RF
ZB	„Statistiken“	88	59,57%	38,74%	38,74%	62,33%
	„Stacking“	88	65,59%	65,01%	38,74%	69,96%
	MAR(1)	88	57,15%	38,74%	38,74%	57,8%
	DAR(1)	88	55,7%	38,74%	38,74%	55,48%
	MARDAR(1)	88	57,48%	38,74%	38,74%	56,24%
	MAR(2)	88	53,84%	38,74%	38,74%	54,22%
	DAR(2)	88	53,19%	38,74%	38,74%	53,24%
	MARDAR(2)	88	54,36%	38,74%	38,74%	54,12%
	MAR(3)	88	54,25%	38,74%	38,74%	54,93%
	DAR(3)	88	53,44%	38,74%	38,74%	53,97%
SB	MARDAR(3)	88	54,75%	38,74%	38,74%	53,55%
	„Statistiken“	88	60,07%	40,09%	38,74%	63,29%
	„Stacking“	88	61,48%	62,38%	38,74%	72,16%
	MAR(1)	88	56,19%	38,74%	42,05%	57,66%
	DAR(1)	88	53,81%	38,74%	41,97%	55,91%
	MARDAR(1)	88	55,37%	38,74%	42,05%	57,62%
	MAR(2)	88	54,46%	38,74%	42,49%	55,63%
	DAR(2)	88	53,6%	38,74%	42,39%	53,68%
	MARDAR(2)	88	54,51%	38,74%	42,52%	54,65%
	MAR(3)	88	55,11%	38,74%	42,29%	54,62%
CB	DAR(3)	88	53,98%	38,74%	42,17%	55,15%
	MARDAR(3)	88	54,22%	38,74%	42,25%	55,79%
	„Statistiken“	88	62,18%	51,07%	44,74%	65,55%
	„Stacking“	88	63,73%	75,69%	38,49%	71,06%
	MAR(1)	88	55,85%	58,25%	50,17%	46,82%
	DAR(1)	88	54,08%	38,74%	50,08%	54,82%
MARDAR(1)	88	57,24%	58,24%	49,77%	48,48%	
MAR(2)	88	50,56%	57,21%	52,36%	43,61%	

Fortsetzung auf der nächsten Seite...

Tabelle A.5: Klassifikationsergebnisse für „Piano-Vs-Alle“ (Fortsetzung)

MK	AM (Rang)	TF	F-Maß(%)			
			<i>k</i> -NN	SVM	GMM	RF
Ch	DAR(2)	88	51,77%	38,74%	52,31%	50,59%
	MARDAR(2)	88	51,31%	58,47%	52,38%	44,16%
	MAR(3)	88	49,29%	56,58%	51,29%	43,07%
	DAR(3)	88	52,87%	38,74%	50,93%	49,28%
	MARDAR(3)	88	50,05%	56,83%	51,26%	43,79%
	„Statistiken“	88	59,17%	52,5%	38,74%	63,19%
	„Stacking“	88	61,73%	68,25%	38,74%	65,68%
	MAR(1)	88	50,04%	38,73%	43,03%	48,29%
	DAR(1)	88	50,88%	38,74%	42,84%	51,53%
	MARDAR(1)	88	50,1%	38,81%	42,85%	47,77%
	MAR(2)	88	51,71%	38,75%	38,69%	46,19%
	DAR(2)	88	51,78%	38,71%	39,16%	50,31%
	MARDAR(2)	88	52,72%	38,67%	39,19%	46,32%
	MAR(3)	88	49,2%	39%	39,11%	45,48%
	DAR(3)	88	48,25%	38,7%	39,1%	50,38%
	MARDAR(3)	88	48,56%	39,08%	38,99%	46,1%

Tabelle A.6: Klassifikationsergebnisse für „Flöte-Vs-Alle“

MK	AM (Rang)	TF	F-Maß(%)			
			<i>k</i> -NN	SVM	GMM	RF
ZB	„Statistiken“	88	57,73%	40,78%	40,78%	58,92%
	„Stacking“	88	61,27%	50,11%	40,78%	60,17%
	MAR(1)	88	55,72%	40,78%	40,78%	53,79%
	DAR(1)	88	55,93%	40,78%	40,78%	53,95%
	MARDAR(1)	88	55,04%	40,78%	40,78%	53,95%
	MAR(2)	88	55,5%	40,78%	40,78%	55,83%
	DAR(2)	88	55,71%	40,78%	40,78%	56,09%
	MARDAR(2)	88	55,51%	40,78%	40,78%	56,18%
	MAR(3)	88	54,77%	40,78%	40,78%	52,98%
	DAR(3)	88	53,47%	40,78%	40,78%	53,78%
	MARDAR(3)	88	56,33%	40,78%	40,78%	53,68%
SB	„Statistiken“	88	61,73%	42,83%	40,78%	62,21%
	„Stacking“	88	62,54%	57,57%	40,78%	64,35%

Fortsetzung auf der nächsten Seite...

Tabelle A.6: Klassifikationsergebnisse für „Flöte-Vs-Alle“ (Fortsetzung)

MK	AM (Rang)	TF	F-Maß(%)			
			<i>k</i> -NN	SVM	GMM	RF
	MAR(1)	88	53,79%	40,78%	42,36%	52,43%
	DAR(1)	88	54,09%	40,78%	42,46%	54,71%
	MARDAR(1)	88	53,1%	40,78%	42,36%	52,58%
	MAR(2)	88	54,07%	40,78%	43,63%	50,5%
	DAR(2)	88	55,01%	40,78%	43,76%	52,26%
	MARDAR(2)	88	53,82%	40,78%	43,62%	50,95%
	MAR(3)	88	53,7%	40,78%	43,72%	49,15%
	DAR(3)	88	53,76%	40,78%	43,67%	51,11%
	MARDAR(3)	88	52,58%	40,78%	43,77%	49,91%
CB	„Statistiken“	88	68,4%	65,06%	53,87%	68,21%
	„Stacking“	88	68,1%	71,61%	42,53%	68,23%
	MAR(1)	88	52,4%	59,23%	49,26%	46,08%
	DAR(1)	88	54,14%	40,78%	49,02%	57,08%
	MARDAR(1)	88	54,17%	59,57%	49,22%	50,83%
	MAR(2)	88	50,23%	57,86%	49,89%	42,6%
	DAR(2)	88	50,84%	40,78%	50,13%	51,66%
	MARDAR(2)	88	49,68%	58,52%	50,2%	44,01%
	MAR(3)	88	46,45%	55,57%	48,67%	41,51%
Ch	„Statistiken“	88	58,51%	40,78%	40,78%	58,35%
	„Stacking“	88	58,89%	63,8%	40,78%	57,66%
	MAR(1)	88	49,06%	40,98%	50,19%	43,87%
	DAR(1)	88	51,19%	40,78%	49,48%	46,6%
	MARDAR(1)	88	49,84%	40,75%	50,21%	44,94%
	MAR(2)	88	50,41%	41,16%	40,82%	42,69%
	DAR(2)	88	51,15%	40,82%	42,15%	46,37%
	MARDAR(2)	88	51,55%	41,38%	42,15%	42,7%
	MAR(3)	88	50,81%	41,44%	41,91%	42,02%
DAR(3)	88	54,72%	40,82%	41,46%	45,28%	
MARDAR(3)	88	54,15%	41,42%	41,8%	42,69%	

Tabelle A.7: Klassifikation-Ergebnisse für „Gitarre-Vs-Alle“

MK	AM (Rang)	TF	F-Maß(%)			
			<i>k</i> -NN	SVM	GMM	RF
ZB	„Statistiken“	88	63,48%	64,41%	30,51%	65,39%
	„Stacking“	88	71,37%	79,25%	30,51%	76,3%
	MAR(1)	88	62,89%	60,65%	30,51%	66,42%
	DAR(1)	88	60,82%	59,16%	30,51%	63,32%
	MARDAR(1)	88	62,72%	61,8%	30,51%	66,98%
	MAR(2)	88	59,58%	61,11%	30,51%	64,01%
	DAR(2)	88	58,01%	60,08%	30,51%	61,95%
	MARDAR(2)	88	59,48%	61,57%	30,51%	64,05%
	MAR(3)	88	57,25%	57,79%	30,51%	61,92%
	DAR(3)	88	57,06%	58,06%	30,51%	59,92%
SB	MARDAR(3)	88	57,4%	58,71%	30,51%	62,11%
	„Statistiken“	88	64,63%	66,29%	30,51%	68,99%
	„Stacking“	88	66,79%	78,29%	30,51%	81,23%
	MAR(1)	88	58,08%	53,5%	33,98%	66,47%
	DAR(1)	88	57,63%	53,52%	33,84%	63,31%
	MARDAR(1)	88	58,65%	57,62%	34%	65,69%
	MAR(2)	88	56,73%	50,1%	34,01%	65,79%
	DAR(2)	88	54,48%	46,32%	33,97%	63,43%
	MARDAR(2)	88	56,39%	54,43%	34,05%	65,32%
	MAR(3)	88	53,67%	41,98%	34,33%	65,08%
CB	DAR(3)	88	51,94%	36,82%	34,47%	62,13%
	MARDAR(3)	88	52,48%	48,61%	34,25%	64,97%
	„Statistiken“	88	66,76%	70,77%	35,23%	71,87%
	„Stacking“	88	69,77%	82,37%	38,23%	76,65%
	MAR(1)	88	51,4%	64,48%	40,39%	61,2%
	DAR(1)	88	57,71%	63,36%	40,54%	64,42%
	MARDAR(1)	88	54,46%	64,74%	40,46%	63,08%
	MAR(2)	88	47,92%	64,18%	41,96%	57,22%
	DAR(2)	88	52,31%	62,73%	42,48%	61,69%
	MARDAR(2)	88	49,41%	64,74%	42,11%	59,34%
MAR(3)	88	47,38%	59,44%	39,88%	54,2%	
DAR(3)	88	51,51%	59,65%	39,83%	58,89%	
MARDAR(3)	88	49,76%	61,13%	40,1%	54,93%	

Fortsetzung auf der nächsten Seite...

Tabelle A.7: Klassifikation-Ergebnisse für „Gitarre-Vs-Alle“ (Fortsetzung)

MK	AM (Rang)	TF	F-Maß(%)			
			<i>k</i> -NN	SVM	GMM	RF
Ch	„Statistiken“	88	60,27%	64,16%	30,51%	65,58%
	„Stacking“	88	62,48%	65,61%	30,51%	66,94%
	MAR(1)	88	50,6%	36,48%	34,01%	55,65%
	DAR(1)	88	50,66%	36,02%	34,07%	56,39%
	MARDAR(1)	88	48,87%	36,59%	34,11%	54,59%
	MAR(2)	88	51,73%	39,74%	31,32%	53,2%
	DAR(2)	88	51,95%	36,53%	30,92%	57,61%
	MARDAR(2)	88	50,95%	42,93%	30,45%	54,18%
	MAR(3)	88	52,64%	42,47%	30,44%	50,77%
	DAR(3)	88	54,74%	39,32%	30,44%	55,39%
	MARDAR(3)	88	53,23%	46,6%	30,44%	52,12%

Tabelle A.8: Klassifikationsergebnisse für „Trompete-Vs-Alle“

MK	AM (Rang)	TF	F-Maß(%)			
			<i>k</i> -NN	SVM	GMM	RF
ZB	„Statistiken“	88	77,02%	72,57%	38,56%	79,08%
	„Stacking“	88	77,87%	74,61%	38,56%	79,53%
	MAR(1)	88	67,27%	61,6%	38,56%	72,6%
	DAR(1)	88	66,35%	60,8%	38,56%	69,5%
	MARDAR(1)	88	67,44%	63,1%	38,56%	72,27%
	MAR(2)	88	64,7%	51,04%	38,56%	72,24%
	DAR(2)	88	62,41%	49,27%	38,56%	68,4%
	MARDAR(2)	88	64,22%	54,54%	38,56%	71,75%
	MAR(3)	88	61,22%	42,09%	38,56%	71,01%
	DAR(3)	88	57,66%	41,45%	38,56%	67,67%
	MARDAR(3)	88	59,06%	47,57%	38,56%	70,39%
SB	„Statistiken“	88	76,66%	74,68%	38,56%	80,11%
	„Stacking“	88	69,88%	75,71%	38,56%	80,85%
	MAR(1)	88	65,6%	50,27%	38,8%	73,32%
	DAR(1)	88	63,3%	49,09%	38,71%	67,97%
	MARDAR(1)	88	64,16%	53,63%	38,83%	72,82%
	MAR(2)	88	62,81%	41,25%	40,19%	71,31%
	DAR(2)	88	59,39%	39,1%	40,4%	66,8%

Fortsetzung auf der nächsten Seite...

Tabelle A.8: Klassifikationsergebnisse für „Trompete-Vs-Alle“ (Fortsetzung)

MK	AM (Rang)	TF	F-Maß(%)			
			<i>k</i> -NN	SVM	GMM	RF
	MARDAR(2)	88	60,9%	44,05%	40,31%	70,49%
	MAR(3)	88	62,75%	41,42%	39,82%	71,72%
	DAR(3)	88	58,55%	38,8%	39,86%	65,57%
	MARDAR(3)	88	62,13%	44,41%	39,81%	72,04%
CB	„Statistiken“	88	78,56%	77,79%	60,44%	80,36%
	„Stacking“	88	75,89%	81,48%	40,32%	80,67%
	MAR(1)	88	44,9%	70,15%	46,6%	68,79%
	DAR(1)	88	64,26%	66,26%	46,71%	67,96%
	MARDAR(1)	88	48,08%	71,24%	46,48%	67,98%
	MAR(2)	88	42,91%	70,9%	47,88%	63,49%
	DAR(2)	88	58,51%	66,87%	47,96%	66,88%
	MARDAR(2)	88	45,29%	70,41%	47,83%	64,43%
	MAR(3)	88	39,98%	66,35%	47,77%	60,3%
	DAR(3)	88	53,24%	64,26%	47,74%	63,66%
Ch	MARDAR(3)	88	41,51%	66,2%	47,44%	60,82%
	„Statistiken“	88	62,74%	66,67%	38,56%	69,79%
	„Stacking“	88	63,36%	64,57%	38,56%	68,77%
	MAR(1)	88	49,24%	39,01%	51,82%	62,32%
	DAR(1)	88	50,64%	38,56%	51,99%	62,3%
	MARDAR(1)	88	48,71%	38,98%	51,77%	61,85%
	MAR(2)	88	52,05%	39,02%	38,57%	59,96%
	DAR(2)	88	51,48%	38,57%	40,07%	62,65%
	MARDAR(2)	88	50,8%	39,09%	38,57%	60,41%
	MAR(3)	88	52,97%	39,49%	39,08%	51,29%
	DAR(3)	88	51,69%	38,69%	39,31%	56,86%
	MARDAR(3)	88	51,85%	39,75%	39,08%	51,68%

Abbildungsverzeichnis

1.1	Ansätze der Organisation von Musik [21]	2
1.2	Musikgenre-Klassifikationssystem	3
2.1	Das MFCC Extraktions-Schema [21]. Die Zahlen über jedem Schritt stellen die Dimension der Kompression in dem jeweiligen Schritt dar.	8
2.2	Ein Beispiel für den Unterschied zwischen einem Nächster-Nachbar- und einem 5-Nächste-Nachbarn- Klassifikator unter Verwendung von 2-dimensionalen Punkten und 3 Klassen (rot, blau, grün) [15]	14
2.3	Die Teilung eines Quadrats und der abgeleitete CART-Baum [3]	16
2.4	Optimale Hyperebene und Margin eines SVM Klassifikators	17
3.1	Beispiel einer ARFF-Datei	22
3.2	Die Struktur der Vorverarbeitung.	23
3.3	Das Klassendiagramm für die temporale Merkmalsberechnung	25
3.4	Vorgehensweise für die Untersuchung der Aggregationsmethoden	28
4.1	Ergebnisse der Klassifikation für „Klassik-Vs-Alle“	32
4.2	Ergebnisse der Klassifikation für „Electro-Vs-Alle“	33
4.3	Ergebnisse der Klassifikation für „Jazz-Vs-Alle“	34
4.4	Ergebnisse der Klassifikation für „Rap-Vs-Alle“	34
4.5	Ergebnisse der Klassifikation für „Piano-Vs-Alle“	36
4.6	Ergebnisse der Klassifikation für „Flöte-Vs-Alle“	37
4.7	Ergebnisse der Klassifikation für „Gitarre-Vs-Alle“	37
4.8	Ergebnisse der Klassifikation für „Trompete-Vs-Alle“	38
4.9	Boxplot der statistischen Verteilungen von „Statistiken“ und MAR(1) bei der Klassifikation von Genres	41
4.10	Boxplot der statistischen Verteilungen von „Statistiken“ und MAR(1) bei der Instrumentenerkennung	42
4.11	Boxplot der statistischen Verteilungen von „Statistiken“ und DAR(1) bei der Klassifikation von Genres	43

4.12	Boxplot der statistischen Verteilungen von „Statistiken“ und DAR(1) bei der Instrumentenerkennung	44
4.13	Boxplot der statistischen Verteilungen von „Stacking“ und MAR(1) bei der Klassifikation von Genres	45
4.14	Boxplot der statistischen Verteilungen von „Stacking“ und MAR(1) bei der Instrumentenerkennung	46
4.15	Boxplot der statistischen Verteilungen von „Stacking“ und DAR(1) bei der Klassifikation von Genres	47
4.16	Boxplot der statistischen Verteilungen von „Stacking“ und DAR(1) bei der Instrumentenerkennung	47
4.17	Boxplot der statistischen Verteilungen von MAR(1) und DAR(1) bei der Klassifikation von Genres	49
4.18	Boxplot der statistischen Verteilungen von MAR(1) und DAR(1) bei der Instrumentenerkennung	49
4.19	Boxplot der statistischen Verteilungen von MAR(1) und MARDAR(1) bei der Klassifikation von Genres	51
4.20	Boxplot der statistischen Verteilungen von MAR(1) und MARDAR(1) bei der Instrumentenerkennung	51
4.21	Boxplot der statistischen Verteilungen von DAR(1) und MARDAR(1) bei der Klassifikation von Genres	53
4.22	Boxplot der statistischen Verteilungen von DAR(1) und MARDAR(1) bei der Instrumentenerkennung	53
4.23	Boxplot der statistischen Verteilungen von MARDAR(1) und „Statistiken“ bei der Klassifikation von Genres	54
4.24	Boxplot der statistischen Verteilungen von MARDAR(1) und „Statistiken“ bei der Instrumentenerkennung	55
4.25	Boxplot der statistischen Verteilungen von MARDAR(1) und „Stacking“ bei der Klassifikation von Genres	56
4.26	Boxplot der statistischen Verteilungen von MARDAR(1) und „Stacking“ bei der Instrumentenerkennung	57
4.27	Boxplot der statistischen Verteilungen von „Statistiken“ und „Stacking“ bei der Klassifikation von Genres	58
4.28	Boxplot der statistischen Verteilungen von „Statistiken“ und „Stacking“ bei der Instrumentenerkennung	59
4.29	Boxplot der statistischen Verteilungen von „Statistiken“ und MAR(1) bei der Klassifikation von Genres	62
4.30	Boxplot der statistischen Verteilungen von „Statistiken“ und MAR(1) bei der Instrumentenerkennung	63

4.31	Boxplot der statistischen Verteilungen von „Stacking“ und MAR(1) bei der Klassifikation von Genres	64
4.32	Boxplot der statistischen Verteilungen von „Stacking“ und MAR(1) bei der Instrumentenerkennung	64
4.33	Boxplot der statistischen Verteilungen von „Statistiken“ und „Stacking“ bei der Klassifikation von Genres	66
4.34	Boxplot der statistischen Verteilungen von „Statistiken“ und „Stacking“ bei der Instrumentenerkennung	66
4.35	Boxplot der statistischen Verteilungen von „Statistiken“ und DAR(1) bei der Klassifikation von Genres	68
4.36	Boxplot der statistischen Verteilungen von „Statistiken“ und DAR(1) bei der Instrumentenerkennung	68
4.37	Boxplot der statistischen Verteilungen von „Stacking“ und DAR(1) bei der Klassifikation von Genres	69
4.38	Boxplot der statistischen Verteilungen von „Stacking“ und DAR(1) bei der Instrumentenerkennung	70
4.39	Boxplot der statistischen Verteilungen von „Statistiken“ und „Stacking“ bei der Klassifikation von Genres	71
4.40	Boxplot der statistischen Verteilungen von „Statistiken“ und „Stacking“ bei der Instrumentenerkennung	72

Tabellenverzeichnis

2.1	Merkmale nach Merkmalsgruppen	6
2.2	Konfusionsmatrix	18
4.1	Verteilung der Alben nach Genres	29
4.2	Verteilung der Titel nach Instrumenten	30
4.3	Parameter für die Klassifikation von Genres	31
4.4	Parameter für das Instrumentenerkennungsexperiment	35
4.5	Statistischer Test zum Vergleich des F-Maßes bei „Statistiken“ und MAR(1)	42
4.6	Statistischer Test zum Vergleich des F-Maßes bei „Statistiken“ und DAR(1)	44
4.7	Statistischer Test zum Vergleich des F-Maßes bei „Stacking“ und MAR(1)	45
4.8	Statistischer Test zum Vergleich des F-Maßes bei „Stacking“ und DAR(1)	48
4.9	Statistischer Test zum Vergleich des F-Maßes bei MAR(1) und DAR(1)	50
4.10	Statistischer Test zum Vergleich des F-Maßes bei MAR(1) und MARDAR(1)	52
4.11	Statistischer Test zum Vergleich des F-Maßes bei DAR(1) und MARDAR(1)	52
4.12	Statistischer Test zum Vergleich des F-Maßes bei MARDAR(1) und „Statistiken“	55
4.13	Statistischer Test zum Vergleich des F-Maßes bei MARDAR(1) und „Stacking“	57
4.14	Statistischer Test zum Vergleich des F-Maßes bei „Statistiken“ und „Stacking“	59
4.15	Dimension nach Merkmalsgruppen	60
4.16	Dimension nach Aggregationsmethoden	61
4.17	Statistischer Test zum Vergleich des F-Maßes bei „Statistiken“ und MAR(1)	63
4.18	Statistischer Test zum Vergleich des F-Maßes bei „Stacking“ und MAR(1)	65
4.19	Statistischer Test zum Vergleich des F-Maßes bei „Statistiken“ und „Stacking“	67
4.20	Statistischer Test zum Vergleich des F-Maßes bei „Statistiken“ und DAR(1)	69
4.21	Statistischer Test zum Vergleich des F-Maßes bei „Stacking“ und DAR(1)	70
4.22	Statistischer Test zum Vergleich des F-Maßes bei Statistiken und Stacking	72
A.1	Klassifikationsergebnisse für „Klassik-Vs-Alle“	77
A.1	Klassifikationsergebnisse für „Klassik-Vs-Alle“ (Fortsetzung)	78
A.2	Klassifikationsergebnisse für „Electro-Vs-Alle“	78

A.2	Klassifikationsergebnisse für „Electro-Vs-Alle“ (Fortsetzung)	79
A.2	Klassifikationsergebnisse für „Electro-Vs-Alle“ (Fortsetzung)	80
A.3	Klassifikationsergebnisse für „Jazz-Vs-Alle“	80
A.3	Klassifikationsergebnisse für „Jazz-Vs-Alle“ (Fortsetzung)	81
A.4	Klassifikationsergebnisse für „Rap-Vs-Alle“	81
A.4	Klassifikationsergebnisse für „Rap-Vs-Alle“ (Fortsetzung)	82
A.4	Klassifikationsergebnisse für „Rap-Vs-Alle“ (Fortsetzung)	83
A.5	Klassifikationsergebnisse für „Piano-Vs-Alle“	83
A.5	Klassifikationsergebnisse für „Piano-Vs-Alle“ (Fortsetzung)	84
A.6	Klassifikationsergebnisse für „Flöte-Vs-Alle“	84
A.6	Klassifikationsergebnisse für „Flöte-Vs-Alle“ (Fortsetzung)	85
A.7	Klassifikation-Ergebnisse für „Gitarre-Vs-Alle“	86
A.7	Klassifikation-Ergebnisse für „Gitarre-Vs-Alle“ (Fortsetzung)	87
A.8	Klassifikationsergebnisse für „Trompete-Vs-Alle“	87
A.8	Klassifikationsergebnisse für „Trompete-Vs-Alle“ (Fortsetzung)	88

Literaturverzeichnis

- [1] ALPAYDIN, ETHEM: *Maschinelles Lernen*. Oldenbourg Verlag, 2008.
- [2] BICKEL, PETER J und KJELL A DOKSUM: *Mathematical Statistics: Basic Ideas and Selected Topics, volume I*, Band 117. CRC Press, 2015.
- [3] BISHOP, C.M.: *Pattern Recognition and Machine Learning*. Information Science and Statistics. Springer, 2006.
- [4] BORTZ, J. und C. SCHUSTER: *Statistik für Human- und Sozialwissenschaftler: 7. Auflage*. Springer-Lehrbuch. Springer Berlin Heidelberg, 2011.
- [5] BREIMAN, LEO: *Random forests*. Machine learning, 45(1):5–32, 2001.
- [6] BURGESS, CHRISTOPHER JC: *A tutorial on support vector machines for pattern recognition*. Data mining and knowledge discovery, 2(2):121–167, 1998.
- [7] DAVIS, STEVEN B und PAUL MERMELSTEIN: *Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences*. Acoustics, Speech and Signal Processing, IEEE Transactions on, 28(4):357–366, 1980.
- [8] DUDA, RICHARD O, PETER E HART und DAVID G STORK: *Pattern classification*. John Wiley & Sons, 2012.
- [9] FRIEDMAN, JEROME, TREVOR HASTIE und ROBERT TIBSHIRANI: *The elements of statistical learning*, Band 1. Springer series in statistics Springer, Berlin, 2001.
- [10] FUJISHIMA, TAKUYA: *Realtime chord recognition of musical sound: A system using common lisp music*. In: *Proc. ICMC*, Band 1999, Seiten 464–467, 1999.
- [11] GIANNAKOPOULOS, THEODOROS und AGGELOS PIKRAKIS: *Introduction to Audio Analysis: A MATLAB® Approach*. Academic Press, 2014.
- [12] HALL, MARK, EIBE FRANK, GEOFFREY HOLMES, BERNHARD PFAHRINGER, PETER REUTEMANN und IAN H WITTEN: *The WEKA data mining software: an update*. ACM SIGKDD explorations newsletter, 11(1):10–18, 2009.

- [13] JAPKOWICZ, NATHALIE und MOHAK SHAH: *Evaluating learning algorithms: a classification perspective*. Cambridge University Press, 2011.
- [14] JODER, CYRIL, SLIM ESSID und GAËL RICHARD: *Temporal integration for audio classification with application to musical instrument classification*. Audio, Speech, and Language Processing, IEEE Transactions on, 17(1):174–186, 2009.
- [15] KARPATHY, ANDREJ: *Image Classification: Data-driven Approach, k-Nearest Neighbor, train/val/test splits*. CS231n Convolutional Neural Networks for Visual Recognition, Computer Science Department, Stanford University, URL: <http://cs231n.github.io/classification/>, letzter Zugriff am 19. April 2016.
- [16] LAW, MARTIN: *A simple introduction to support vector machines*. Lecture for CSE, 802, Department of Computer Science and Engineering, Michigan State University, 2006, URL: http://www.cise.ufl.edu/class/cis4930sp11dtm/notes/intro_svm_new.pdf, letzter Zugriff am 19. April 2016.
- [17] LIAW, ANDY und MATTHEW WIENER: *Classification and Regression by Random Forest*. R news, 2(3):18–22, 2002.
- [18] LÜTKEPOHL, HELMUT: *New introduction to multiple time series analysis*. Springer Science & Business Media, 2005.
- [19] LÜTKEPOHL, HELMUT: *Vector autoregressive models*. Springer, 2011.
- [20] MARTIN, RAINER und ANIL NAGATHIL: *Cepstral modulation ratio regression (CM-RARE) parameters for audio signal analysis and classification*. In: *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, Seiten 321–324, 2009.
- [21] MENG, ANDERS: *Temporal feature integration for music organisation*. Doktorarbeit, Technical University of Denmark, Department of Informatics and Mathematical Modeling, 2006.
- [22] MENG, ANDERS, PETER AHRENDT und JAN LARSEN: *Improving music genre classification by short time feature integration*. In: *Acoustics, Speech, and Signal Processing, 2005. Proceedings.(ICASSP'05). IEEE International Conference on*, Band 5, Seiten v–497, 2005.
- [23] MENG, ANDERS, PETER AHRENDT, JAN LARSEN und LARS KAI HANSEN: *Temporal feature integration for music genre classification*. Audio, Speech, and Language Processing, IEEE Transactions on, 15(5):1654–1664, 2007.

- [24] NAGATHIL, ANIL, IGOR VATOLKIN und WOLFGANG THEIMER: *Comparison of partition-based audio features for music classification*. ITG-Fachbericht-Sprachkommunikation 2010, 2010.
- [25] NEUMAIER, ARNOLD und TAPIO SCHNEIDER: *Estimation of parameters and eigenmodes of multivariate autoregressive models*. ACM Transactions on Mathematical Software (TOMS), 27(1):27–57, 2001.
- [26] NTALAMPIRAS, STAVROS und NIKOS FAKOTAKIS: *Modeling the temporal evolution of acoustic parameters for speech emotion recognition*. Affective Computing, IEEE Transactions on, 3(1):116–125, 2012.
- [27] NTALAMPIRAS, STAVROS, ILYAS POTAMITIS und NIKOS FAKOTAKIS: *Exploiting temporal feature integration for generalized sound recognition*. EURASIP Journal on Advances in Signal Processing, 2009(1):1–12, 2009.
- [28] PETERSOHN, H.: *Data Mining: Verfahren, Prozesse, Anwendungsarchitektur*. Oldenbourg, 2005.
- [29] POLLARD, HOWARD F und ERIK V JANSSON: *A tristimulus method for the specification of musical timbre*. Acta Acustica united with Acustica, 51(3):162–171, 1982.
- [30] REYNOLDS, DOUGLAS: *Gaussian mixture models*. Encyclopedia of Biometrics, Seiten 827–832, Springer, 2015.
- [31] SCHNEIDER, TAPIO und ARNOLD NEUMAIER: *Algorithm 808: ARfit—A Matlab package for the estimation of parameters and eigenmodes of multivariate autoregressive models*. ACM Transactions on Mathematical Software (TOMS), 27(1):58–65, 2001.
- [32] SOLTAU, HAGEN, TANJA SCHULTZ, MARTIN WESTPHAL und ALEX WAIBEL: *Recognition of music types*. In: *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*, Band 2, Seiten 1137–1140, 1998.
- [33] THEIMER, WOLFGANG, IGOR VATOLKIN und ANTTI ERONEN: *Definitions of audio features for music content description*. Technical University of Dortmund, Faculty of Computer Science, 2008.
- [34] VATOLKIN, IGOR: *Improving supervised music classification by means of multi-objective evolutionary feature selection*. Doktorarbeit, Technische Universität Dortmund, Fakultät für Informatik, 2013.
- [35] VATOLKIN, IGOR, WOLFGANG M THEIMER und MARTIN BOTTECK: *AMUSE (Advanced MUSic Explorer)—A Multitool Framework for Music Data Analysis*. In: *Procee-*

dings of the 11th International Society for Music Information Retrieval Conference (ISMIR), Seiten 33–38, 2010.

- [36] WITTEN, IAN H und EIBE FRANK: *Data Mining: Practical machine learning tools and techniques, 3rd edition*. Elsevier, 2011.
- [37] WU, X. und V. KUMAR: *The Top Ten Algorithms in Data Mining*. Chapman & Hall/CRC Data Mining and Knowledge Discovery Series. CRC Press, 2009.

Hiermit versichere ich, dass ich die vorliegende Arbeit selbstständig verfasst habe und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet sowie Zitate kenntlich gemacht habe.

Dortmund, den 19. April 2016

Ettiboa Daniel Alexis Adouakou

