Sprecherklassifizierung anhand des vokalen Dreiecks

Bachelorarbeit

im Studiengang Technomathematik

vorgelegt von:	Lea Charlotte Sophie Kowsky
Matrikelnummer:	52 11 56
Geburtsdatum:	10.01.1992
Geburtsort:	Marl
Erstgutachter:	Prof. Dr. Babette Dellen
Zweitgutachter:	Prof. Dr. Markus Neuhäuser
externer Betreuer:	Dr. Rolf Bardeli

Remagen, den 14. Dezember 2015

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Listings

1 Einleitung

Für eine Vielzahl von Anwendungen, wie beispielsweise der Sprecheridentifikation oder zur Diagnose von neurodegenerativen Erkrankungen, wird die quantitative Analyse von Sprache immer wichtiger. So können Veränderungen in der menschlichen Sprache in Frühstadien verschiedener Krankheiten wie Parkinson, Demenz und Depressionen beobachtet werden [McRae et al., 2002]. Es wurde bereits belegt, dass durch Depressionen die Sprechmotorik beeinflusst wird und der Frequenzbereich der gesprochenen Sprache deutlich begrenzt wird [Darby et al., 1984].
Zur Auswertung der akustischen Signale wird von verschiedenen Methoden der digitalen Signalverarbeitung Gebrauch gemacht. Hierbei liegt der Fokus auf der Auswertung der im Vokalraum am meisten verstärkten Frequenzen. Diese nennen sich die ersten beiden Formanten. Die Bestimmung dieser ersten beiden Formanten erfolgte hierbei über die Auswertung des Linear Predictive Polynomial [Snell and Milinazzo, 1993].
Hierbei werden die komplexen Nullstellen über numerische Methoden bestimmt und mit Hilfe der in [Snell and Milinazzo, 1993] beschriebenen Methode die Formanten bestimmt. Dieser Algorithmus wurde bereits im vorherigen Praxisprojekt implementiert. Werden die ersten beiden Formanten gegeneinander aufgetragen, spannen sie das vokale Dreieck auf, wie in [McRae et al., 2002] und [Scherer et al., 2015] beschrieben. Hierbei beschreiben die ersten beiden Formanten der Phoneme a, u und i die Eckpunkte des vokalen Dreiecks. Die aufgespannte Fläche kann genutzt werden um verschiedene Sprecher mittels unterschiedlicher Eigenschaften des aufgespannten Dreiecks zu charakterisieren.

In der vorliegenden Arbeit soll unter Zuhilfenahme der verschiedenen Charakteristika des vokalen Dreiecks ein statistisches Modell zur Beschreibung von unterschiedlichen Sprechern entwickelt werden. Mit diesem Modell soll es ermöglicht werden, einen neuen Datensatz einer Klasse zuzuordnen. Zur Unterscheidung verschiedener Sprecher wird hierbei von den Methoden des

maschinellen Lernens Gebrauch gemacht. Dabei werden Sprecher unterteilt nach unterschiedlichen Features klassifiziert und in verschiedene Klassen geordnet. Bei der Klassifikation von Daten werden diese in einen Lerndatensatz, sowie in einen Testdatensatz unterteilt. Hierbei sollen anhand des Lerndatensatzes vorhandene Strukturen und Muster erlernt werden. Anschließend wird das Modell mit noch nicht bekannten Daten getestet. Zum Lernen und Klassifizieren von Daten stehen verschiedene Algorithmen und Methoden bereit. Dabei werden zunächst gemischte Gauß-Modelle (GMM) betrachtet. In dieser Arbeit wird zunächst versucht eine Aussage über die Genauigkeit des maschinellen Lernens mit Gaußmischverteilungen im Bezug auf die Sprechererkennung zu machen. Anschließend wird versucht durch weitere Verfahren wie den nächste-Nachbar Klassifikator (nNK) eine größtmögliche Genauigkeit zu erreichen. Die in der Arbeit benutzen Audio-Dateien sind Mitschnitte aus Fernsehsendungen. Hierbei sollen Unterschiede zwischen weiblichen und männlichen Sprechern, sowie die Art des Sprechens erschlossen werden. So sollen beispielsweise Personengruppen wie Diskussionsteilnehmer, Außenreporter und Moderator auseinandergehalten werden.

2 Grundlagen

2.1 Erzeugung von Lauten im menschlichen Stimmtrakt

In diesem Kapitel soll ein grundlegender Einblick in die Funktionsweise des menschlichen Vokaltraktes gegeben werden. Die menschliche Stimme kann als die Modulation von Schall durch Zusammenwirken verschiedener Organe beschrieben werden, welche in Abbildung 2.1 benannt sind. Zur Erzeugung von Tönen wird ein Luftstrom erzeugt, welcher durch die Luftröhre gelangt und danach über die Stimmlippen streift. Hierbei versetzt er diese in Schwingungen und erzeugt den Grundton. Es kann zwischen stimmhaften und stimmlosen Lauten unterschieden werden. Wenn sich die Stimmlippen, während sie von dem Luftstrom gestreift werden, schnell öffnen und wieder schließen, spricht man von stimmhaften Lauten. Wenn der Luftstrom die Stimmlippen ungehindert passieren kann, spricht man von stimmlosen Lauten. Die Größe der Stimmlippen variiert von Mensch zu Mensch und beeinflusst die Tonhöhe maßgeblich. Ein kleinerer Kehlkopf besitzt kürzere und schmalere Stimmlippen und erzeugt verhältensmäßig hohe Grundtöne. Hingegen sind in einem größeren Kehlkopf längere und breitere Stimmlippen vorhanden, wodurch Grundtöne in tieferen Frequenzen erzeugt werden. Sind die Stimmlippen entspannter, so schwingen diese langsamer und erzeugen tiefere Töne, sind die Stimmlippen angespannt, so erzeugen sie Töne mit einer höheren Grundfrequenz. Die Frequenz des Grundtons ist vor allem von der Größe des Kehlkopfes und von der Länge der Stimmbänder abhängig. Nach dem Passieren des Kehlkopfes gelangt die Luft in den Vokaltrakt, welcher aus Mund, Nasen- und Rachenraum besteht. Hier werden die vorher erzeugten Töne weiter geformt. Dies geschieht durch Bewegungen mit Lippen, Zunge und Zähnen mit denen der Vokaltrakt vergrößert oder verkleinert wird. Durch die Veränderung des Vokaltraktes wird der Luftstrom in unterschiedlichen Frequenzen reflektiert, was in unterschiedlichen Tonhöhen resultiert. Somit können Töne anhand ihres Entstehungsortes

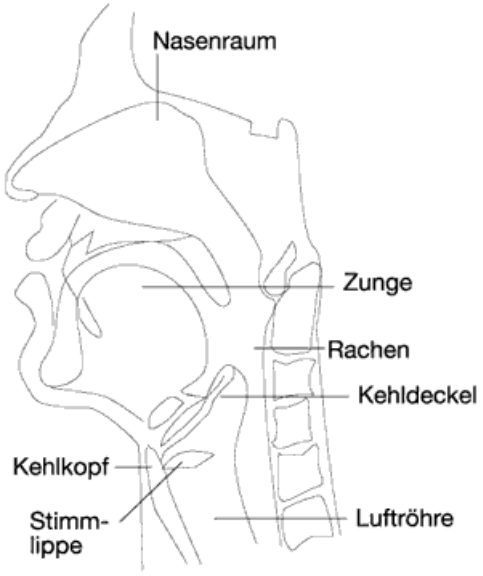


Abbildung 2.1: Abbildung des menschlichen Stimmtrakts [Schnaitter, 2015]

in unterschiedliche Kategorien wie nasal, frikativ oder plosiv eingeordnet werden. Bei nasalen Tönen entweicht die vorher eingeatmete Luft durch die Nase, bei frikativen Lauten, die auch als Reibelaute bezeichnet werden, entweicht die Luft durch eine Engstelle und wird verwirbelt, bei plosiven Lauten entsteht der Klang durch Freisetzen eines zuvor gestauten Luftstromes.

2.2 Parameter zur Charakterisierung der menschlichen Stimme

Zur Charakterisierung der menschlichen Stimme wird von unterschiedlichen Methoden der Signalverarbeitung Gebrauch gemacht. Zur Analyse eines akustischen Signals wird die Fouriertransformation genutzt, welche ein Signal vom Zeit- in den Frequenzbereich transformiert. Die Grundidee der Fouriertransformation beruht darauf, dass sich ein periodisches Signal als Überlagerung von Sinus- und Kosinusfunktionen schreiben lässt. Die komplexe Fourierreihe eines Signals $s(t)$ mit Periode T hat die folgende Form:

$$s(t) = \sum_{k=-\infty}^{\infty} F_k e^{-ik\omega t} \tag{2.1}$$

Mit den Fourierkoeffizienten F_k:

$$F_k = \frac{1}{T} \int_0^T s(t) e^{ik\omega t} dt \tag{2.2}$$

wobei $\omega = \frac{2\pi}{T}$.
Bei der Betrachtung eines nicht periodischen Signals wird von einer unendlichen Periodendauer ausgegangen ($T \to \infty$), somit nähert sich die Grundfrequenz $\omega = \frac{2\pi}{T}$ null und wird durch das infinitesimale Frequenzelement $d\omega$ ersetzt. Die Summe aus Gleichung 2.1 geht somit in ein Integral über [von Grünigen, 2001]:

$$s(t) = \int_{-\infty}^{\infty} S(\omega) e^{-i\omega t} d\omega \tag{2.3}$$

$$S(\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} s(t) e^{i\omega t} dt \tag{2.4}$$

Die Gleichung 2.3 beschreibt hierbei die Fouriertransformation und durch Gleichung 2.4 wird die inverse Fouriertransformation beschrieben. Reelle Signale haben jedoch eine endliche Dauer und liegen in diskreter (abgetasteter) Form vor. Zur Diskretisierung muss das Abtasttheorem eingehalten werden. Durch das von Nyquist und Shannon formulierte Abtasttheorem [Shannon, 2001], wird sicher gestellt, dass das ursprüngliche Signal fehlerfrei aus dem abgetasteten Signal rekonstruiert werden kann. Das Abtasttheorem besagt, dass die Abtastfrequenz f_a größer als die doppelte im Signal vorkommende maximale Frequenz f_{max} sein soll:

$$f_a > 2 \cdot f_{max} \tag{2.5}$$

Zur Herleitung der diskreten Fouriertransformation aus 2.3 wird die kontinuierliche Funktion $s(t)$ durch die diskreten Abtastwerte $s[nT]$ ersetzt. Das Integral kann aufgrund der Diskretisierung nur numerisch ausgewertet werden und wird durch eine Summe approximiert. Das Differential dt beschreibt anschaulich die Breite der Rechteckflächen unter der Funktion und wird durch den Faktor T genähert. Zudem wird die Kreisfrequenz ω durch $2\pi f$ ersetzt:

$$S(f) = \sum_{-\infty}^{\infty} s(nT) e^{i2\pi f nT} T \tag{2.6}$$

Da die Abtastfrequenz konstant ist, stellt die Multiplikation mit dem Faktor T eine simple Multiplikation mit einem Skalar dar, die sowohl in der Hin- als auch in der Rücktransformation auftaucht. Diese kann aus Gründen der Einfachheit weggelassen werden [von Grünigen, 2001]. Aus der unendlichen Anzahl an Messwerten im Signal wird eine endliche Anzahl an Abtastwerten herausgeschnitten. Zudem kann das Abtastintervall T im Nenner durch den Kehrwert der Abtastfrequenz $\frac{1}{f_a}$ ersetzt werden:

$$S(f) = \sum_{0}^{N-1} s(nT) e^{i2\pi f n \frac{f}{f_a}} \tag{2.7}$$

Die Funktion $S(f)$ ist periodisch zur Abtastfrequenz f_a und hat an den Frequenzstellen $f_m = 0, \frac{f_a}{N}, 2\frac{f_a}{N}, \ldots, m\frac{f_a}{N}$ unabhängige Funktionswerte. Einsetzen von $f_m = m\frac{f_a}{N}$ liefert:

$$S(m\frac{f_a}{N}) = \sum_{0}^{N-1} s(nT) e^{i2\pi n \frac{m\frac{f_a}{N} f}{f_a}} \tag{2.8}$$

Aus Gründen der Bequemlichkeit wird in der Literatur meist auf die Faktoren $\frac{f_s}{N}$ und T verzichtet. Dann lautet die Fouriertransformation:

$$S(m) = \sum_{0}^{N-1} s(n) e^{i2\pi n \frac{2\pi}{N}} \tag{2.9}$$

Analog zur Fouriertransformation kann die Rücktransformation definiert werden:

$$s[n] = \frac{1}{N} \sum_{n=0}^{N-1} S[m] e^{-\frac{i2\pi nm}{N}} \tag{2.10}$$

In digitalen Systemen findet zur Verringerung des Rechenaufwandes die Fast Fourier Transformation Anwendung. Hierbei reduziert sich der Rechenaufwand von $O(N^2)$ zu $O(N \log_2 N)$ Berechnungen. Hierzu muss allerdings die Anzahl an Abtastwerten einer Zweierpotenz entsprechen. Gegebenenfalls muss deswegen das abgetastete Signal durch sogenanntes Zero-Padding künstlich verlängert werden, damit dieser Formalismus Anwendung finden kann. Mit Hilfe der in Gleichung 2.2 berechneten Fourierkoeffizienten lassen sich das Amplituden- und Phasenspektrum bestimmen. Der Informationsgehalt des Signals im Frequenzbereich ist identisch mit dem Informationsgehalt des Signals im Zeitbereich. Somit lassen sich Signale ohne Informationsverlust von der einen in die andere Darstellung überführen. Abbildung 2.2 zeigt das akustische Signal eines

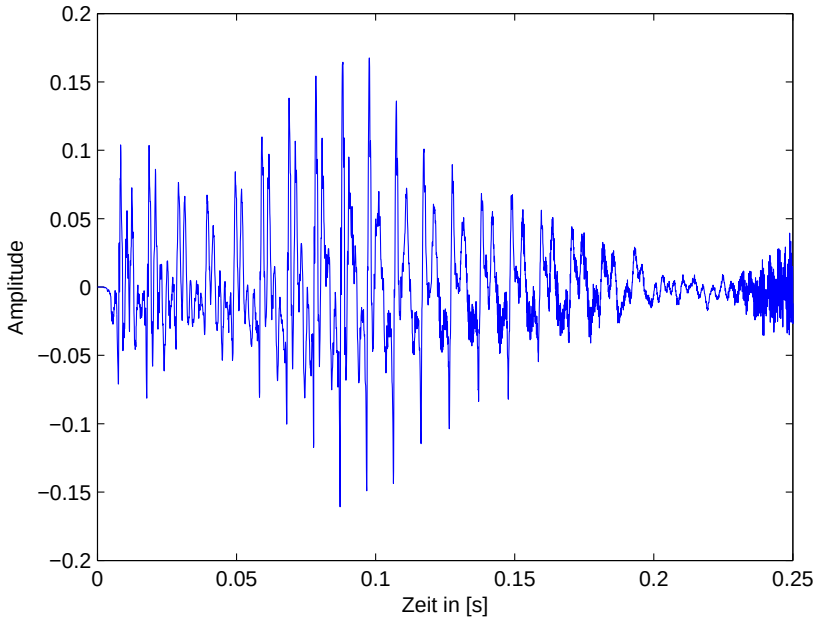


Abbildung 2.2: Zeitlicher Verlauf der Stimme eines männlichen Sprechers

männlichen Sprechers, der unter anderem die Vokale i und a sagt. Trägt man den Betrag der komplexen Fourierkoeffizienten dieses Signals gegen die jeweilige Frequenz auf, so lässt sich am Spektrum das Vorkommen der einzelnen Frequenzen im Signal erkennen (siehe Abbildung 2.3). Aus Abbildung 2.3 ist erkennbar, dass das Spektrum eines abgetasteten Signals periodisch ist. Dies beruht auf der Definition der diskreten Fouriertransformation, das Spektrum wiederholt sich mit der Abtastfrequenz f_a [Unbehauen, 2002].

In Abbildung 2.3 ist ein zentrales Maximum bei 97,9755 Hz erkennbar. Weiterhin sind einige deutliche Nebenmaxima ersichtlich. In einem Spektrum wird die Konzentration der Energie in Abhängigkeit der Frequenz dargestellt. Auch wenn ein Signal ohne Informationsverlust vom Zeitbereich in den Frequenzbereich und wieder zurück transformiert werden kann, bedarf es weiterhin einer Darstellung der Energiekonzentration in Abhängigkeit der Zeit und der Frequenz. Dazu wird Gebrauch von einem Spektrogramm gemacht. Zur Erzeugung des Spektrogramms wird das Signal in viele äquidistante kurze Abschnitte unterteilt, welche überlappen. Auf jeden dieser Abschnitte wird wieder einzeln die Fouriertransformation angewendet. Werden diese einzelnen Spektren jeweils gegen die Zeit aufgetragen erhält man ein Spektrogramm. Aus diesem wird ersichtlich, zu welchen Zeitpunkten die Energiekonzentration im Signal besonders hoch war. So zeigt Abbildung 2.4 eine hohe Konzentration von

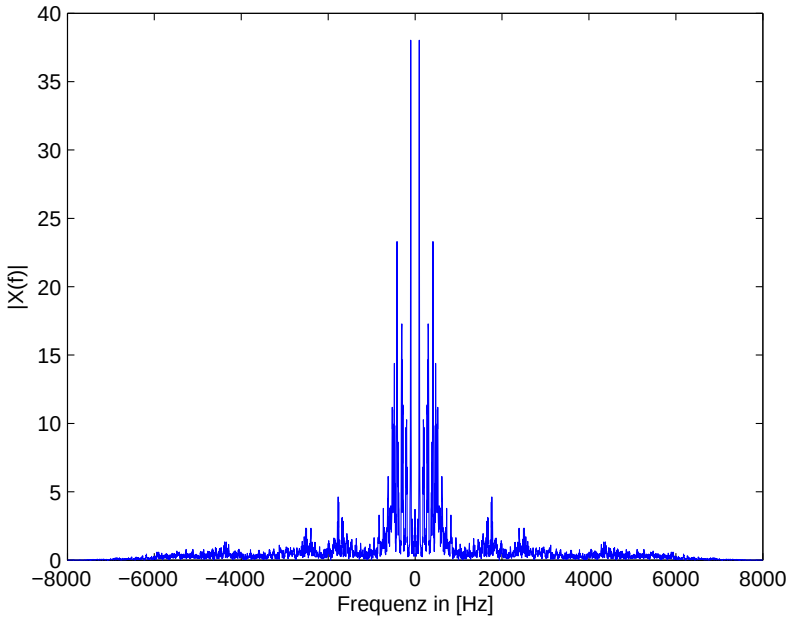


Abbildung 2.3: Frequenzspektrum des Signals aus Abbildung 2.2. Hier wird der Betrag der komplexen Fourierkoeffizienten des Signals gegen die entsprechende Frequenz aufgetragen.

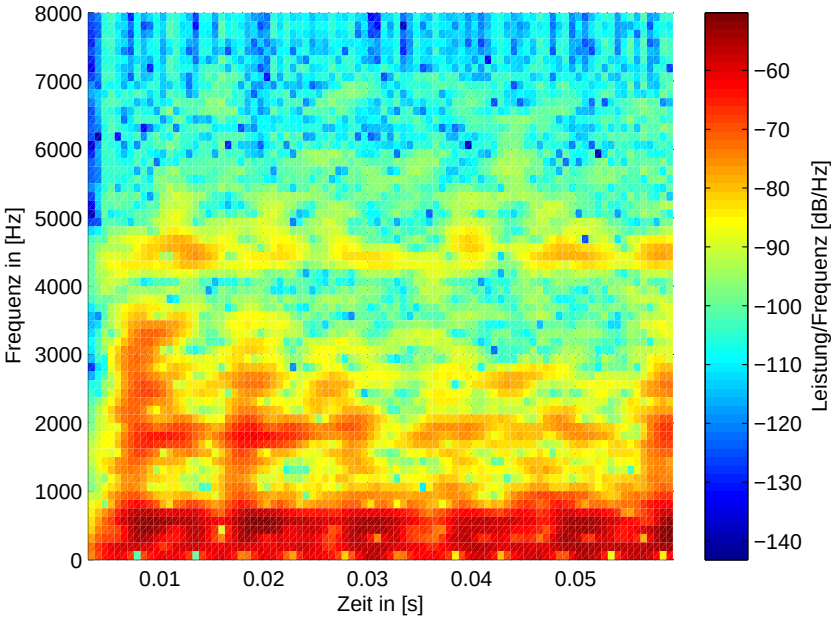


Abbildung 2.4: Spektrogramm des in Abbildung 2.2 und Abbildung 2.3 gezeigten Signals. In dieser Darstellung sind Konzentration der Energie in Abhängigkeit der Zeit und Frequenz erkennbar. Somit können die ersten beiden Formanten, anhand einer Konzentration von Energie, lokalisiert werden.

Energie in Rot an. Die Bereiche in denen die Energie konzentriert ist, nennen sich Formanten. Im weiteren Verlauf dieser Arbeit sind der erste und der zweite Formant von besonderer Bedeutung. Zur Bestimmung der Formanten stehen unterschiedliche Algorithmen zur Verfügung, wie das Peak-Picking oder das Filtern der Formanten mittels des Linear Predictive Polynomial, wie in [Snell and Milinazzo, 1993] beschrieben. Auf die Algorithmen wird in Kapitel 3 näher eingegangen. Zur Bestimmung des vokalen Dreiecks werden die ersten beiden Formanten der Eckpunkte des vokalen Dreiecks ermittelt. Die Eckpunkte entsprechen den Vokalen *a, i* und *u*. Danach wird für den ersten und zweiten Formanten jedes Vokals der Mittelwert bestimmt. Diese drei Punkte spannen im Koordinatensystem das vokale Dreieck auf, wie in Abbildung 2.5 gezeigt. Hierbei werden der unterschiedliche Flächeninhalt des roten Dreiecks, welches durch die Formanten eines weiblichen Sprechers erzeugt wurde, und des blauen Dreiecks, welches durch die ersten beiden Formanten eines männlichen Sprechers erzeugt wurde deutlich. In Abbildung 2.5 wird deutlich, dass sich die Lage und der Flächeninhalt der vokalen Dreiecke in Abhängigkeit der Sprecher ändern, was bereits in [Scherer et al., 2015] beschrieben wurde.

2.3 Die multivariate Gaußverteilung und ihre Kenngrößen

Die spätere Modellierung der Audio-Daten erfolgt anhand einer Gaußmischverteilung. Dazu werden in diesem Kapitel einige wichtige Kenngrößend der Gaußverteilung angesprochen. Zur Beschreibung von Daten in statistischen Zusammenhängen wird vom Begriff der Zufallsvariable Gebrauch gemacht. Einer Zufallsvariablen werden Realisationen eines Zufallsexperiments auf einem entsprechenden Messraum zugeordnet. Die zugehörige Verteilungsfunktion wird beschrieben durch eine reellwertige Funktion. Diese gibt an, wie sich die Werte der Zufallsvariablen auf die unterschiedlichen Wahrscheinlichkeiten verteilen [Bosch, 2006]. Zufallsvariablen können sowohl stetige als auch diskrete Werte annehmen. Im stetigen Fall spricht man von der Wahrscheinlichkeitsdichtefunktion. Hierbei ist es nur möglich zu berechnen, mit welcher Wahrscheinlichkeit sich eine Realisation der Zufallsvariable in einem bestimmten Intervall befindet. Die in der vorliegenden Arbeit zur Modellierung genutzte Funktion ist die Gauß- oder Normalverteilung. Die Wahrscheinlichkeitsdichte einer stetigen Zufallsvariable heißt normal verteilt mit $X \sim N(\mu, \sigma^2)$, wenn für die

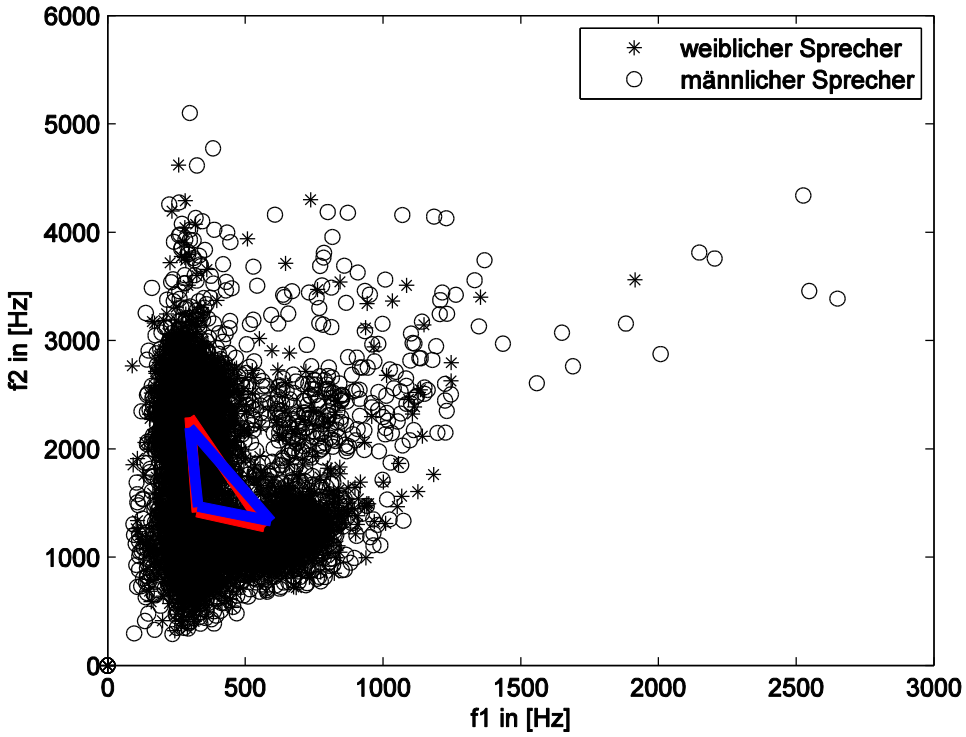


Abbildung 2.5: Die Audio-Daten, welche Mitschnitte verschiedener Fernsehsendungen sind, wurden getrennt nach weiblicher und männlicher Rede analysiert. Hierbei wurden die ersten beiden Formanten bestimmt. Die Mittelwerte der Formanten a, i und u spannen das vokale Dreieck auf. Hier werden das vokale Dreieck von männlicher Rede (blau) und weiblicher Rede (rot) gezeigt.

Dichtefunktion $f(x)$ gilt:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2} \tag{2.11}$$

Zudem muss die Normierungseigenschaft erfüllt sein:

$$\int_{-\infty}^{\infty} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2} dx = 1 \tag{2.12}$$

Die Wahrscheinlichkeitsdichtefunktion ist von dem Erwartungswert μ und der Varianz σ^2 abhängig. Für eine diskrete Zufallsvariable X, welche den endlichen Wertevorrat $W = \{x_1, \ldots, x_m\}$ besitzt, definiert sich der Erwartungswert $E(X)$ wie folgt [Bosch, 2006]:

$$E(X) = \sum_{i=1}^{m} x_i P(X = x_i) \tag{2.13}$$

Für stetige Zufallsvariablen wird der Erwartungswert durch Gleichung 2.14 beschrieben:

$$E(X) = \int_{-\infty}^{\infty} x \cdot f(x) dx \tag{2.14}$$

In Gleichung 2.15 und 2.16 wird die Varianz für diskrete als auch für stetige Zufallsvariabeln definiert:

$$\sigma^2 = \sum (x - \mu)^2 P(X = x) \tag{2.15}$$

$$\sigma^2 = \int_{-\infty}^{\infty} (x - \mu)^2 \cdot f(x) dx \tag{2.16}$$

Die in dieser Arbeit analysierten Daten liegen in Tupeln vor. Somit werden zur hinreichenden Darstellung eines mehrdimensionalen Datensatzes weitere Dimensionen benötigt. Analog zur Gaußverteilung kann durch Gleichung 2.17 die multivariate Gaußverteilung definiert werden:

$$f(x) = \frac{1}{\sqrt{(2\pi)^p |\Sigma|}} \exp(-\frac{1}{2}(\vec{x} - \vec{\mu})^\top \Sigma^{-1} (\vec{x} - \vec{\mu})) \tag{2.17}$$

Hierbei gibt p die Anzahl der Dimensionen der Gaußverteilung an. Der Faktor $\frac{1}{\sqrt{(2\pi)^p|\Sigma|}}$ wird als Normalisierungsfaktor bezeichnet. Dieser gewährleistet die Normierung der Fläche unter dem Integral. Weiterhin hängt die multivariate Gaußverteilung von der Kovarianzmatrix Σ und dem Vektor der Erwartungswerte $\vec{\mu}$ ab. Die Kovarianzmatrix ist eine symmetrische positiv-definite Matrix, in welcher die Korrelationen zwischen den einzelnen Elementen eines Zufallsvektors gegeben ist. Die Kovarianz eines Vektors $\vec{x}$ definiert sich wie in Gleichung 2.18 beschrieben:

$$Cov(\vec{x}) = \begin{pmatrix} Cov(x_1, x_1) & \dots & Cov(x_1, x_n) \\ \vdots & \ddots & \vdots \\ Cov(x_n, x_1) & \dots & Cov(x_n, x_n) \end{pmatrix} \tag{2.18}$$

Da die in Gleichung 2.18 beschriebene Kovarianzmatrix symmetrisch positiv-definit ist, wird durch $(\vec{x} - \vec{\mu})^T \Sigma^{-1} (\vec{x} - \vec{\mu})$ ein Skalarprodukt definiert, somit wird hiermit zwei Vektoren ein Skalar zugeordnet, wie in [Jänich, 2013] beschrieben. Das Skalarprodukt induziert eine Norm auf dem jeweiligen Vektorraum. Somit wird mit $(\vec{x}-\vec{\mu})^T \Sigma^{-1} (\vec{x}-\vec{\mu})$ der Abstand zwischen den Vektoren $\vec{x}$ und $\vec{\mu}$ berechnet. Also ist die Wahrscheinlichkeit eines Merkmalsvektors durch den Abstand von $\vec{\mu}$ bezüglich der durch Σ^{-1} gegebenen Metrik gegeben. Die Wahrscheinlichkeit verringert sich exponentiell mit dem Abstand von $\vec{\mu}$. Die zugehörige Kovarianzmatrix kann somit als Drehung und Streckung eines Kreises interpretiert werden. Die zugehörigen Eigenvektoren geben die Richtung der Hauptachsen der resultierenden Ellipse an.

2.4 Der Maximum-Likelihood-Schätzer

Beim Modellieren von aufgenommenen Messdaten müssen die unbekannten Parameter der Verteilungsfunktion mit Hilfe der schon vorhandenen Daten geschätzt werden. Im weiteren Verlauf wird zur Herleitung des Expectation-Maximisation-Algorithmus (EM-Algorithmus) (siehe Kapitel 3.3.1)
der Maximum-Likelihood-Schätzer (ML-Schätzer) benötigt. Die Grundidee des ML-Schätzers besteht darin, die unbekannten Parameter der Wahrscheinlichkeitsdichtefunktion so zu schätzen, dass sich die Verteilungsfunktion möglichst gut an die Realisationen der Zufallsvariable X anpasst. Zur Anwendung des ML-Schätzers muss die Wahrscheinlichkeitsdichtefunktion der Zufallsvariabeln

bekannt sein. Bei gegebener Zufallsstichprobe $X = \{x_1, \dots x_n\}$, die aus einer bekannten Verteilung f gezogen wurden, beschreibt $f(\vec{x}|\theta)$ eine Familie von Verteilungen, die abhängig von dem zu schätzenden Parameter θ variiert. Somit hängt die Verbundwahrscheinlichkeit und folglich auch die Wahrscheinlichkeit für diese konkrete Stichprobe von den Realisierungen der Stichprobe und dem zu schätzenden Parameter θ ab:

$$f(x_1 \dots x_n|\theta) = \prod_{i=1}^{n} f(x_i|\theta) =: L(\theta|x_1 ... x_n) \tag{2.19}$$

Diese Funktion wird als Likelihoodfunktion L bezeichnet. L wird als Funktion von θ betrachtet. Ziel ist es nun, bei fester Zufallsstichprobe X den unbekannten Parameter so zu schätzen, dass die größtmögliche Wahrscheinlichkeit gefunden wird, bei der f gerade diese Zufallsstichprobe produziert hat. Um nun θ zu schätzen, muss der Wert gefunden werden, für den L ihr Maximum annimmt. Um das Maximum von L zu finden, wird die Funktion nach dem unbekannten Parameter abgeleitet. Zur Vereinfachung der Ableitung, wird statt der Likelihoodfunktion der Logarithmus der Likelihoodfunktion $\mathcal{L} := \log(L)$ verwendet. Dieser hat aufgrund der strengen Monotonie sein Maximum genau an der gleichen Stelle. Zudem kann unter Einhalten der Maximumsbedingungen das Maximum entsprechend berechnet werden:

$$\frac{dL}{d\theta} = 0 \tag{2.20}$$

$$\frac{d^2L}{d^2\theta} < 0 \tag{2.21}$$

Für den Fall einer multivariaten normalverteilten Zufallsvariable wird der Maximum-Likelihood-Schätzer für μ folgendermaßen hergeleitet:
Sei $X \sim N(\mu, \Sigma)$ und sei $(x_1, x_2, ..., x_N)$ eine Zufallsstichprobe. So lautet die Wahrscheinlichkeitsdichtefunktion der multivariaten Normalverteilung

$$f(x) = \frac{1}{\sqrt{(2\pi)^p det(\Sigma)}} \exp(-\frac{1}{2}(\vec{x} - \vec{\mu})^\top \Sigma^{-1}(\vec{x} - \vec{\mu})) \tag{2.22}$$

mit $\mu \in \Re^p$ und $\Sigma \in \Re^{p \times p}$.
Mit Gleichung 2.22 wird nun die Ableitung der Loglikelihoodfunktion $\mathcal{L}$ berechnet:

$$\frac{\partial \mathcal{L}}{\partial \mu} = \frac{\partial}{\partial \mu} \log \prod_{i=1}^{N} \mathcal{N}(x_i, \mu, \Sigma) \tag{2.23}$$

Mit den Logarithmusgesetzen wird daraus:

$$\frac{\partial \mathcal{L}}{\partial \mu} = \frac{\partial}{\partial \mu} \sum_{i=1}^{N} \log \mathcal{N}(x_i, \mu, \Sigma) \tag{2.24}$$

$$= \sum_{i=1}^{N} \frac{\partial}{\partial \mu} \log(c \cdot \exp(-\frac{1}{2}(x_n - \mu)^\top \Sigma^{-1}(x_n - \mu))) \tag{2.25}$$

Mit $c := \frac{1}{\sqrt{(2\pi)^p |\Sigma|}}$

$$\frac{\partial \mathcal{L}}{\partial \mu} = \sum_{n=1}^{N} \underbrace{\frac{\partial}{\partial \mu} \log c}_{=0} + (-\frac{1}{2} \frac{\partial}{\partial \mu}(x_n - \mu)^\top \Sigma^{-1}(x_n - \mu)) \tag{2.26}$$

$$= -\frac{1}{2} \sum_{n=1}^{N} \frac{\partial}{\partial \mu}(x_n - \mu)^\top \Sigma^{-1}(x_n - \mu) \tag{2.27}$$

Mit $\frac{\partial}{\partial x} x^\top A x = 2x^\top A$ wird daraus:

$$\frac{\partial \mathcal{L}}{\partial \mu} = -\sum_{n=1}^{N} (x_n - \mu)^\top \Sigma^{-1} \tag{2.28}$$

$$\tag{2.29}$$

Nun wird das Ergebnis gleich null gesetzt und nach μ aufgelöst. Somit ergibt sich als Schätzer $\hat{\mu}$:

$$\hat{\mu} = \frac{1}{N} \sum_{n=1}^{N} x_n \tag{2.30}$$

Zur Schätzung der inversen Kovarianzmatrix Σ^{-1} wird analog vorgegangen. Zunächst wird die Loglikelihoodfunktion nach Σ^{-1} abgeleitet:

$$\frac{\partial \mathcal{L}}{\partial \Sigma^{-1}}(\mu, \Sigma) = \sum_{n=1}^{N} \frac{\partial}{\partial \Sigma^{-1}} \log(\frac{1}{\sqrt{(2\pi)^d |\Sigma|}}) - \frac{1}{2} \frac{\partial}{\partial \Sigma^{-1}} (x_n - \mu)^\top \Sigma^{-1} (x_n - \mu) \tag{2.31}$$

$$= \sum_{n=1}^{N} \frac{\partial}{\partial \Sigma^{-1}} (\log \frac{1}{\sqrt{(2\pi)^d}} + \frac{1}{2} \log |\Sigma^{-1}|) \tag{2.32}$$

$$- \frac{1}{2} \frac{\partial}{\partial \Sigma^{-1}} (x_n - \mu)^\top \Sigma^{-1} (x_n - \mu) \tag{2.33}$$

$$= \frac{1}{2} \sum_{n=1}^{N} \frac{\partial}{\partial \Sigma^{-1}} (\log |\Sigma^{-1}|) - \frac{1}{2} \sum_{n=1}^{N} \frac{\partial}{\partial \Sigma^{-1}} (x_n - \mu)^\top \Sigma^{-1} (x_n - \mu) \tag{2.34}$$

Das Ergebnis der zweiten Summe ist ein Skalar. Da die Spur einer 1×1 Matrix die Matrix selber ist, kann Gleichung 2.34 geschrieben werden als:

$$\frac{\partial \mathcal{L}}{\partial \Sigma^{-1}} = \frac{1}{2} \sum_{n=1}^{N} \frac{\partial}{\partial \Sigma^{-1}} (\log |\Sigma^{-1}|) - \frac{1}{2} \sum_{n=1}^{N} \frac{\partial L}{\partial \Sigma^{-1}} \text{spur}((x_n - \mu)^\top \Sigma^{-1} (x_n - \mu)) \tag{2.35}$$

Mit $\frac{\partial}{\partial A} \log |A| = A^{-\top}$,$(AB)^\top = B^\top A^\top$ und $\frac{\partial}{\partial A} \text{spur}(BA) = B^\top$ wird Gleichung 2.35 zu:

$$\frac{\partial \mathcal{L}}{\partial \Sigma^{-1}}(\mu, \Sigma) = \frac{1}{2} \sum_{n=1}^{N} \Sigma^\top - \frac{1}{2} \sum_{n=1}^{N} (x_n - \mu)(x_n - \mu)^\top \tag{2.36}$$

Somit kann Σ geschätzt werden als:

$$\hat{\Sigma} = \frac{1}{N} \sum_{n=1}^{N} (x_n - \mu)(x_n - \mu)^\top \tag{2.37}$$

3 Methoden

3.1 Vorbereitung der Daten

Um ein statistisches Modell zur Datenmodellierung zu entwickeln, wurden die vorhandenen Daten in einen Lern- und einen Testdatensatz geteilt. Der Lerndatensatz, der aus 70% der Dateien besteht, wird zum Anpassen des Modells benutzt. Der dem Modell noch nicht bekannte Testdatensatz wird anschließend genutzt um das Modell zu testen. Die zu analysierenden Daten liegen als Audiodateien vor. Zusätzlich existiert zu jeder Audiodatei eine Textdatei in dem Anfangs- und Endpunkt jedes Phonems gespeichert war. Anschließend wurde noch eine Textdatei erstellt, die Anfangs- und Endzeitpunkt von verschiedenen Sprecher enthält. Die Sprecher wurden entsprechend Tabelle 3.1 kodiert. Zuerst erfolgt das Einlesen der Textdatei, in welcher die entsprechenden Pho-

Tabelle 3.1: Kodierung der einzelnden Sprecher

Kode	Sprecher
1	weiblicher Sprecher
2	männlicher Sprecher
3	männliche geplante Rede
4	weibliche Diskussionsteilnehmerin
5	männlicher Diskussionsteilnehmer
6	männliche geplante Rede
7	weibliche ungeplante Rede
8	weibliche geplante Rede

neme gespeichert sind. Hierbei werden in einem Vektor jeweils Anfangswert, Endwert und das entsprechende Phonem gespeichert. In Listing 3.1 wird beschrieben, wie durch eine Abfrage eine Matrix erstellt wird, die Anfangs- und Endwert der Phoneme a, i und u speichert. In Abhängigkeit des Phonems wird in der dritten Spalte ein Kode eingefügt, um das jeweilige Phonem nachträglich noch erkennen zu können.

```
datazw=cell(1,2);
for i=1:s
zw=cell2mat(datazw{1,2}(i));
    if (strcmp(zw,'a:')) %code for a is 1,green
        data(j,1)=datazw{1,1}(i)/1000;
        data(j,2)=datazw{1,1}(i+1)/1000;
        data(j,3)=1;
        j=j+1;
    end

    if (strcmp(zw,'u:')) %code for u is 2,blue
        data(j,1)=datazw{1,1}(i)/1000;
        data(j,2)=datazw{1,1}(i+1)/1000;
        data(j,3)=2;
        j=j+1;
    end


    if (strcmp(zw,'i:')) %code for i is 3,red
         data(j,1)=datazw{1,1}(i)/1000;
         data(j,2)=datazw{1,1}(i+1)/1000;
         data(j,3)=3;
         j=j+1;
    end
end
```

Listing 3.1: MATLAB Kode um Anfangs- und Endpunkt der Phoneme *a*, *i* und *u* in der Matrix data zu speichern

Danach werden die ersten beiden Formanten des entsprechenden Tonabschnitts berechnet. Hierfür wird mit dem Linear Predictive Polynomial gearbeitet, welches mit Hilfe des Burg-Algorithmus, wie in [Gray Jr and Wong, 1980] beschrieben, erlangt wird. Die Bestimmung der Formanten erfolgt dann über die numerische Berechnung der Nullstellen des Linear Predictive Polynomial mit dem QR-Algorithmus, wie in [Stoer et al., 1989] beschrieben[1]. Wie Listing 3.2 zeigt, wird zuerst das jeweilige Tonsegment aus der Audio-Datei extrahiert. Zur Berechnung des Linear Predictive Polynomial liefert MATLAB den Befehl *arburg*. Da die Nullstellen des Polynoms komplex sind, liegen die Nullstellen symmetrisch um den Nullpunkt. Somit ist es für den weiteren Verlauf nur nötig, die Nullstellen mit positivem Imaginärteil zu betrachten. Nach der Berechnung der Frequenz folgt die Auswahl der Formanten nach dem in [Snell and Milinazzo, 1993] genannten Kriterium, welches besagt, dass die Bandbreite eines Formanten kleiner als 400 Hz ist und die Frequenz größer als 400 Hz ist. Die jeweiligen Formanten werden dann in die vierte und fünfte

[1]Der genaue Algorithmus zur Berechnung der ersten beiden Formanten wurde bereits im Praxisprojekt implementiert [Kowsky, 2015]

Spalte der Datenmatrix *data* eingetragen, welche bereits in Listing 3.2 mit dem Anfangs- und Endzeitpunkt der Phoneme *a*, *i* und *u* gefüllt ist.

```
%extract segment in which the formant is included
I0 = round(data(i,1)/dt);
Iend = round(data(i,2)/dt);
x = ys(I0:Iend);
%get the linear predictive polynomial through the burg algorithm
A = arburg(x,12);
rts = roots(A);
rts = rts(imag(rts)>=0);
angz = atan2(imag(rts),real(rts));
[frqs,indices] = sort(angz.*(Fs/(2*pi)));
%get the bandwidth
bw = -1/2*(Fs/(2*pi))*log(abs(rts(indices)));
k = 1;
for p = 1:length(frqs)
    if (frqs(p) > 90 && bw(p) <400)
    formants(k) = frqs(p);
    k= k+1;
end

data(i,4)=formants(1);
data(i,5)=formants(2);
```

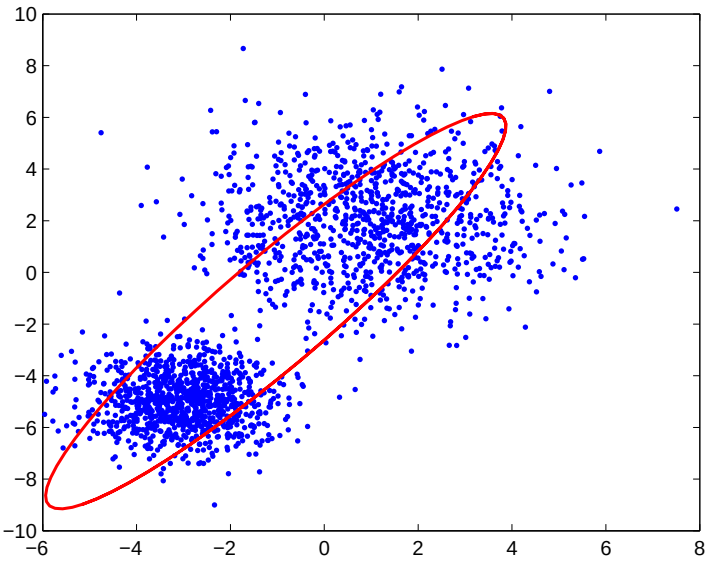
Listing 3.2: MATLAB Kode umd die einzelnen Formanten mittels des Linear Predictive Polynomial zu bestimmen [Mathworks, 2015a]

3.2 Modellierung der Daten durch GMM

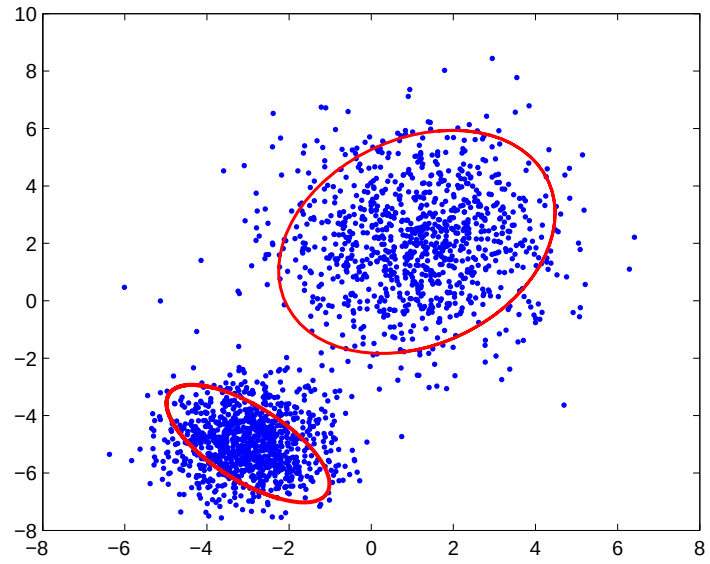
3.2.1 Die gemischte Gaußverteilung

In Kapitel 2.3 wurde bereits die multivariate Gaußverteilung beschrieben. Zur Modellierung von realen Daten ist eine einzige Gaußverteilung jedoch oft unzureichend. In Abbildung 3.1 wird zuerst versucht die Daten durch eine Gaußverteilung zu nähern, hier befinden sich im Zentrum der Daten nur sehr wenige Messwerte. Die an zwei Stellen konzentrierten Messwerte, werden somit durch eine Ellipse schlecht beschrieben. Eine bessere Näherung wird durch die Kombination von zwei Gaußverteilungen erreicht. So wird zur Beschreibung von Daten eine lineare Kombination von m verschiedenen mehrdimensionalen Gaußverteilungen gewählt:

$$G(x,\theta) = \sum_{i=1}^{m} \lambda_i \mathcal{N}(x, \mu_i, \Sigma_i) \tag{3.1}$$



(a) unpassende Modellierung der Daten mit einer Gaußverteilung



(b) Modellierung der Daten mit zwei Gaußverteilungen

Abbildung 3.1: Die Daten werden durch eine unterschiedliche Anzahl an Gaußverteilungen verschieden gut approximiert.

Hierbei ist $\theta = (\lambda_1, \mu_1, \Sigma_1, \ldots, \lambda_m, \mu_m, \Sigma_m)$, was die Parameter jeder einzelnen Gaußverteilung beschreibt. Der Parameter λ_i gibt den Einfluss einer einzelnen Verteilung zur Gesamtverteilung an. Damit $G(x, \theta)$ eine Wahrscheinlichkeitsdichte darstellt, muss $\lambda_i > 0$ und

$$\sum_{i=1}^{m} \lambda_i = 1 \tag{3.2}$$

gewählt werden. Ziel ist es nun Σ_i und μ_i bei einem gegebenen Lerndatensatz $X = \{x_1 \ldots x_n\}$ so zu wählen, dass die Daten bestmöglich modelliert werden.

3.3 Anpassen des GMMs an die Trainingsdaten

3.3.1 Schätzen der Parameter durch den EM-Algorithmus

Sind die Daten durch eine Gaußmischverteilung approximiert, können die Parameter dieser einzelnen Verteilungen nicht direkt mit dem in Kapitel 2.4 beschriebenen Maximum-Likelihood-Schätzer bestimmt werden. Dies liegt darin begründet, dass die Parameter der einzelnen Gaußverteilungen von denselben Daten abhängen und somit nicht unabhängig sind. Daher behilft man sich mit dem Expectation-Maximization-Algorithmus, welcher auf der Einführung einer latenten Variable beruht. Eine latente Variable ist eine Variable, die nicht direkt beobachtet oder gemessen wird, sondern sich aus Berechnungen der Beobachtungen ergibt. Die grundsätzliche Struktur des Expectation-Maximization-Algorithmus beruht darauf, dass in einem ersten Schritt die latente Zufallsvariable ausgerechnet wird. In einem zweiten Schritt werden dann, unter Zuhilfenahme der latenten Zufallsvariable, mit dem Maximum-Likelihood-Schätzer, die zu schätzenden Werte neu bestimmt. Die Neubestimmung der Werte geschieht durch Maximierung der Log-Likelihoodfunktion. Unter der Annahme, dass jeder Testdatensatz zu einer der Komponenten der Mischverteilungen

gehört, kann die Wahrscheinlichkeit $P(k|n)$, mit der der Datensatz x_n zur Verteilung k gehört, geschrieben werden als

$$P(k|n) = \frac{P(n|k)P(k)}{P(n)} = \frac{\mathcal{N}(x_n, \mu_k, \Sigma_k)\lambda_k}{\sum_{j=1}^{m} \lambda_j \mathcal{N}(x_n, \mu_j, \Sigma_j)} \tag{3.3}$$

Nun soll in Iteration i zunächst die latente Variable $P^i(k|n)$ aus den in Schritt $i-1$ geschätzten Parametern bestimmt werden. Danach sollen aus der Variable $P^i(k|n)$ die Schätzer für $\theta^i = (\lambda_1^i, \mu_1^i, \Sigma_1^i, \ldots, \lambda_m^i, \mu_m^i, \Sigma_m^i)$ mit Hilfe der in Kapitel 2.4 beschriebenen Maximum-Likelihood-Methode berechnet werden. Zur Herleitung des Algorithmus wird zuerst die Likelihoodfunktion einer Gaußmischverteilung betrachtet. Diese ist gegeben durch:

$$L(\theta) := \prod_{n=1}^{N} \sum_{k=1}^{m} \lambda_k \mathcal{N}(x_n, \mu_k, \Sigma_k) \tag{3.4}$$

Mit Gleichung 3.4 kann somit die Log-Likelihoodfunktion definiert werden:

$$\mathcal{L}(\theta) := \log(L(\theta)) \tag{3.5}$$

$$= \log(\prod_{n=1}^{N} \sum_{k=1}^{m} \lambda_k \mathcal{N}(x_n, \mu_k, \Sigma_k) \tag{3.6}$$

$$= \sum_{n=1}^{N} \log \sum_{k=1}^{m} \underbrace{\lambda_k \mathcal{N}(x_n, \mu_k, \Sigma_k)}_{q(k,n)} \tag{3.7}$$

Im weiteren Verlauf wird der Term $\lambda_k \mathcal{N}(x_n, \mu_k, \Sigma_k)$ mit $q(k, n)$ bezeichnet. Um nun die latente Variable $P^i(x|k)$ einzuführen wird Gleichung 3.7 mit dem Faktor $1 = \frac{P^i(k|n)}{P^i(k|n)}$ multipliziert.

$$\mathcal{L}(\theta) = \sum_{n=1}^{N} \log \sum_{k=1}^{m} \frac{P^i(k|n)}{P^i(k|n)} q(k, n) \tag{3.8}$$

Um den Term aus Gleichung 3.7 für den iterativen Prozess abzuschätzen, wird die Eigenschaft genutzt, dass die Logarithmusfunktion konkav ist [Heuser, 2013]. Für konkave Funktionen kann die Jensensche Ungleichung angewendet werden. Diese besagt, dass eine konkave Funktion $f : \Re \to \Re$ mit einer diskreten Wahrscheinlichkeitsverteilung $\pi : \{1, 2 \ldots, n\} \to \Re$ und reellen Zahlen

$a_1, \ldots, a_n$ durch eine untere Schranke

$$f\left(\sum_{i=1}^{n} \pi(i) a_i\right) \geq \sum_{i=1}^{n} \pi(i) f(a_i) \tag{3.9}$$

abgeschätzt werden kann [Jensen, 1906]. Wird nun die Ungleichung 3.9 auf Gleichung 3.8 angewandt, erhält man die Abschätzung:

$$\mathcal{L}(\theta) = \sum_{n=1}^{N} \log\left(\sum_{k=1}^{m} \frac{P^i(k|n)}{P^i(k|n)} q(k,n)\right) \geq \sum_{n=1}^{N}\sum_{k=1}^{m} P^i(k|n) \log \frac{q(k,n)}{P^i(k|n)} =: b_i(\theta) \tag{3.10}$$

Nimmt b_i nun bei θ^{i+1} sein Maximum an, ist ein Parameter gefunden, für den $\mathcal{L}$ größer oder gleich ist, als bei der vorherigen Iteration. Nun kann b_i aus Gleichung 3.10 mithilfe der Logarithmusgesetze umgeschrieben werden:

$$b_i(\theta) = \sum_{n=1}^{N}\sum_{k=1}^{m} P^i(k|n) \log q(k,n) - \sum_{n=1}^{N}\sum_{k=1}^{m} P^i(k|n) P^i(k|n) \tag{3.11}$$

Da die zweite Doppelsumme unabhängig von θ ist, reicht es für die zu schätzenden Parameter, die erste Doppelsumme zu maximieren. Im Folgenden wird die erste Doppelsumme definiert als:

$$\beta_i(\theta) := \sum_{n=1}^{N}\sum_{k=1}^{m} P^i(k|n) \log q(k,n) \tag{3.12}$$

Zur Maximierung wird wie in Kapitel 2.4 bereits beschrieben vorgegangen. Im Folgenden soll der Schätzer für μ_k aus Gleichung 3.4 bestimmt werden. Zunächst wird dazu nach dem zu schätzenden Parameter abgeleitet:

$$\frac{\partial \beta_i(\theta)}{\partial \mu_k} = \sum_{n=1}^{N} P^i(k|n) \frac{\partial}{\partial \mu_k} \log q(k,n) \tag{3.13}$$

$$= \sum_{n=1}^{N} P^i(k|n) \left(\frac{1}{2} \frac{\partial}{\partial \mu_k} (x_n - \mu_k)^\top \Sigma_k^{-1} (x_n - \mu_k)\right) \tag{3.14}$$

$$= -\sum_{n=1}^{N} P^i(k|n) (x_n - \mu_k)^\top \Sigma_k^{-1} \tag{3.15}$$

Danach wird die Ableitung gleich null gesetzt und nach dem zu schätzenden Parameter aufgelöst:

$$\sum_{n=1}^{N} P^i(k|n)x_n = \mu_k \sum_{n=1}^{N} P^i(k|n) \tag{3.16}$$

$$\Rightarrow \hat{\mu}_k = \frac{1}{N_k} \sum_{n=1}^{N} P_i(k|n)x_n \tag{3.17}$$

Mit $N_k = \sum_{n=1}^{N} P^i(k|n)$
Das Schätzen von Σ aus Gleichung 3.4 verläuft analog. Da $q(k,n)$ abhängig von Σ^{-1} ist, wird hier nach Σ^{-1} abgeleitet:

$$\frac{\partial \beta_i(\theta)}{\partial \Sigma_k^{-1}} = \sum_{n=1}^{N} P^i(k|n) \frac{\partial \beta_i(\theta)}{\partial \Sigma_k^{-1}} \log q(k,n) \tag{3.18}$$

$$= \sum_{n=1}^{N} P^i(k|n) \frac{\partial}{\partial \Sigma_k^{-1}} \log \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \lambda_k \exp\big(-\frac{1}{2}(x_n - \hat{\mu}_k)^\top \Sigma^{-1}(x_n - \hat{\mu}_k)\big) \tag{3.19}$$

Mit den Logarithmusgesetzen wird daraus:

$$\frac{\partial \beta_i(\theta)}{\partial \Sigma_k^{-1}} = \sum_{n=1}^{N} P^i(k|n) \frac{\partial}{\partial \Sigma_k^{-1}} \big(\log \frac{1}{\sqrt{(2\pi)^d}} + \frac{1}{2}\log|\Sigma^{-1}| + \log(\lambda_k) - \tag{3.20}$$

$$\frac{1}{2}(x_n - \hat{\mu_k})^\top \Sigma^{-1}(x_n - \hat{\mu_k})\big) \tag{3.21}$$

$$= \sum_{n=1}^{N} P^i(k|n) \frac{\partial}{\partial \Sigma_k^{-1}} \big(\frac{1}{2}\log|\Sigma^{-1}| - \frac{1}{2}(x_n - \hat{\mu_k})^\top \Sigma^{-1}(x_n - \hat{\mu_k})\big) \tag{3.22}$$

Wie bereits in Kapitel 2.4 beschrieben, kann die Spur einer 1×1 Matrix als die Matrix selber geschrieben werden:

$$\frac{\partial \beta_i(\theta)}{\partial \Sigma_k^{-1}} = \sum_{n=1}^{N} P^i(k|n) \frac{\partial}{\partial \Sigma_k^{-1}} \frac{1}{2}\log|\Sigma^{-1}| - \frac{\partial \frac{1}{2}\beta_i(\theta)}{\partial \Sigma_k^{-1}} \mathrm{spur}\big((x_n - \hat{\mu_k})^\top \Sigma^{-1}(x_n - \hat{\mu_k})\big) \tag{3.23}$$

Mit $\frac{\partial}{\partial A}\log|A| = A^{-\top}$, $(AB)^\top = B^\top A^\top$ und $\frac{\partial}{\partial A}\mathrm{spur}(BA) = B^\top$ wird Gleichung 3.23 zu:

$$\frac{\partial \beta_i(\theta)}{\partial \Sigma_k^{-1}} = \frac{1}{2}\sum_{n=1}^{N} P^i(k|n)\hat{\Sigma}_k - \frac{1}{2}\sum_{n=1}^{N} P^i(k|n)(x_n - \hat{\mu}_k)(x_n - \hat{\mu}_k)^\top \tag{3.24}$$

Nun kann Gleichung 3.24 gleich null gesetzt werden und der Schätzer für Σ berechnet werden:

$$\frac{1}{2}\sum_{n=1}^{N} P^i(k|n)\hat{\Sigma}_k = \frac{1}{2}\sum_{n=1}^{N} P^i(k|n)(x_n - \hat{\mu}_k)(x_n - \hat{\mu}_k)^\top \tag{3.25}$$

$$\Rightarrow \hat{\Sigma}_k = \frac{1}{\hat{N}_k}\sum_{n=1}^{N} P^i(k|n)(x_n - \hat{\mu}_k)(x_n - \hat{\mu}_k)^\top \tag{3.26}$$

Im Anschluss werden die Parameter λ_k berechnet. Hierbei ist allerdings zu beachten, dass diese positiv sind und aufsummiert eins ergeben müssen. Damit dies gesichert ist, wird für alle $k \in \{1, \ldots, m\}$ der neue Parameter γ_k eingeführt, so dass

$$\lambda_k = \frac{e^{\gamma_k}}{\sum_{j=1}^{m} e^{\gamma_k}}. \tag{3.27}$$

Weiterhin wird die Ableitung von λ_j nach γ_k benötigt:

$$\frac{\partial \lambda_j}{\partial \gamma_k} = \begin{cases} \lambda_k - \lambda_k^2 & \text{wenn } j = k \\ -\lambda_j \lambda_k & \text{sonst} \end{cases} \tag{3.28}$$

Somit kann nun die Ableitung von β_i nach γ_k berechnet werden:

$$\frac{\partial \beta_i}{\partial \gamma_k} = \sum_{n=1}^{N}\sum_{j=1}^{m} P^i(j|n)\frac{\partial}{\partial \lambda_k}\log q(j,n) \tag{3.29}$$

$$= \sum_{n=1}^{N}\sum_{j=1}^{m} P^i(j|n)\frac{\partial}{\partial \lambda_k}\log \lambda_j \tag{3.30}$$

$$= \sum_{n=1}^{N}\Big(P^i(k|n)\frac{1}{\lambda_k}(\lambda_k - \lambda_k^2) - \sum_{j\neq k} P^i(j|n)\frac{1}{\lambda_j}\lambda_j\lambda_k\Big) \tag{3.31}$$

$$= \sum_{n=1}^{N}\Big(P^i(k|n) - P^i(k|n)\lambda_k - \sum_{j\neq k} P^i(j|n)\lambda_k\Big) \tag{3.32}$$

$$= \sum_{n=1}^{N}\Big(P^i(k|n) - \sum_{j=1}^{m} P^i(j|n)\lambda_k\Big) \tag{3.33}$$

$$= \sum_{n=1}^{N}(P^i(k|n) - \lambda_k) \tag{3.34}$$

$$\tag{3.35}$$

Somit erhält man als Schätzer $\hat{\lambda}_k$ für λ_k:

$$\hat{\lambda}_k = \frac{1}{N} \sum_{n=1}^{N} P^i(k|n) \tag{3.36}$$

3.3.2 Zuordnen neuer Datensätze

Zur Zuordnung eines neuen Datensatzes $M_i = \{(x_{1i}, y_{1i}) \dots (x_{ni}, y_{ni})\}$
$1 \leq i \leq N$ bei gegebenen gelernten Modellen $G_1, \dots G_k$, $G_i = \sum_{i=0}^{K} \lambda_i \mathcal{N}(\mu_i, \Sigma_i)$ wird jedes Modell an allen im Testdatensatz vorhandenen Punkten ausgewertet. Diese Auswertungen werden miteinander multipliziert. Hierbei wird eine Wahrscheinlichkeit für ein bestimmtes Modell berechnet:

$$P_i = \prod_{p=0}^{n} \sum_{k=0}^{K} \lambda_k \mathcal{N}((x_{ni}, y_{ni}), \mu_k, \Sigma_k) \tag{3.37}$$

Der neue Datensatz M_{neu} wird dann der Klasse zugeordnet, dessen Modell die höchste Wahrscheinlichkeit hat. Somit gilt:
Ordne den Datensatz M_{neu} der Klasse M_k für die gilt:

$$k = \arg\max\{P_i | 1 \leq i \leq N\} \tag{3.38}$$

In der Praxis wird aus numerischen Gründen oft der Logarithmus der Summe der einzelnen Elemente verwendet, wie in Listing 3.3 beschrieben.

```
fit=0;
zw=0;
for(k=1:1:(size(speaker,1)))
  for i=1:1:gaussian
   zw= zw+ (obj.PComponents(i)*mvnpdf( [speaker(k,1) speaker(k,2) ] , obj.mu(i,:),obj.Sigma(:,:,i) )  );
  end
  fit=fit+log(zw);
end
```

Listing 3.3: Berechnen der Wahrscheinlichkeit für ein einzelnes Modell.

3.4 Modellierung durch Flächeninhalt und Winkel

In Abschnitt 3.2 wurde das Modell direkt an die errechneten Formanten angepasst. Im Folgenden soll eine weitere Methode zur Modellierung erläutert werden. Wie in Abschnitt 2.2 bereits beschrieben, kann die menschliche Stimme durch das vokale Dreieck charakterisiert werden. Das vokale Dreieck ergibt sich durch Verbinden der Mittelpunkte der ersten beiden Formanten der Phoneme *a*, *i* und *u*. Dieses vokale Dreieck wird nun durch seinen Flächeninhalt, sowie seine drei Winkel charakterisiert. Im Rahmen der Lernphase wurde in einem ersten Versuch jedem Audiosegment ein Tupel aus Winkel und Flächeninhalt zugeordnet. Anschließend wird die Gaußmischverteilung auf die, durch Flächeninhalt und Winkel gegebenen Punkte angepasst. Weiterhin wird in einem weiteren Schritt die Gaußmischverteilung auf die durch die ersten beiden Winkel erzeugten Punkte angepasst.

3.5 Der nächste-Nachbar-Klassifikator

Bei der bisher in Kapitel 3.2 beschriebenen Methode wurde aus einem Lerndatensatz ein Modell entwickelt, an das andere Daten angepasst wurden. In diesem Abschnitt soll hingegen die nächste-Nachbar-Methode thematisiert werden, welche gänzlich ohne Modelle arbeitet. Die Klassifikation erfolgt über einen Merkmalsvektor $\vec{x} \in \Re^n$. Der Merkmalsvektor fasst verschiedene numerische Kennwerte eines Objektes zusammen. Zur Klassifikation eines neuen Objektes, werden die Distanz[2] des zu klassifizierenden Merkmalsvektors zu den Merkmalsvektoren aller anderen Klassen berechnet. Der Vektor wird derjenigen Klasse k zugeordnet, zu der der Vektor den geringsten Abstand hat. Somit gilt bei gegebenen Musterklassen M_i $1 \leq i \leq N$ mit gegebenen Merkmalsvektoren $\vec{m_i}$ für einen unbekannten Vektor $\vec{x}$:
Ordne $\vec{x_k}$ der Klasse M_k zu, für die gilt:

$$k = \arg\min\{d(x, m_i) | 1 \leq i \leq N\} \tag{3.39}$$

[2]Hierbei sind unterschiedliche Normen möglich. Es bieten sich beispielsweise die Euklidnorm oder Manhattanmetrik an.

Im vorliegenden Fall sind die Daten allerdings als Mengen von Tupeln $M_i = \{(x_{1i}, y_{1i}) \dots (x_{ni}, y_{ni})\}$ gegeben (Abbildung 3.2).

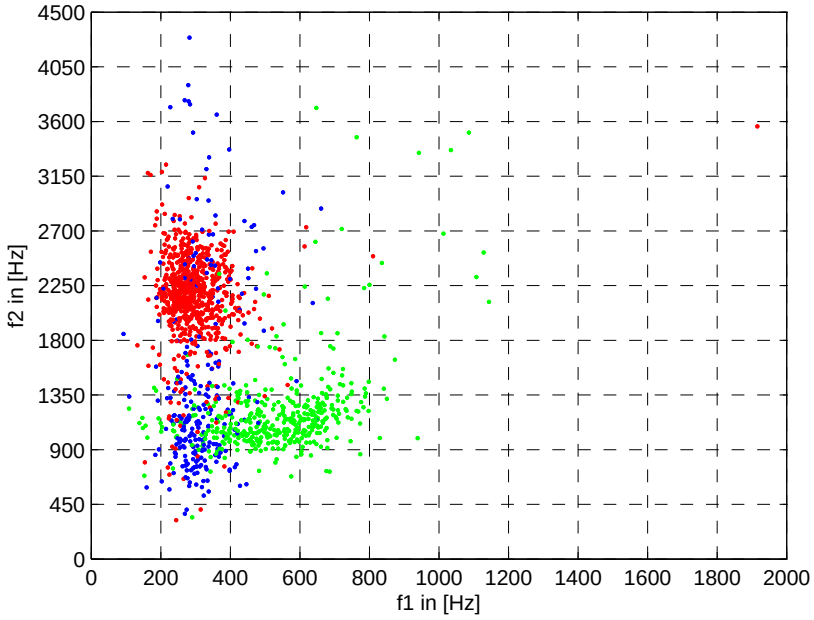


Abbildung 3.2: Erzeugung des Merkmalsvektors am Beispiel der Audio-Daten eines weiblichen Sprechers. Erster und zweiter Formant der Audio-Daten eines weiblichen Sprechers sind gegeneinander aufgetragen. Hierbei sind die Formanten der Vokale a (grün), i (rot) und u (blau) farbig dargestellt. Weiterhin wurde ein 10×10 Gitternetz über die Grafik gelegt um aus der zweidimensionalen Grafik einen Merkmalsvektor zu erzeugen.

Um hier einen Merkmalsvektor zu definieren, wird ein Histogramm zur Hilfe genommen (Abbildung 3.3).

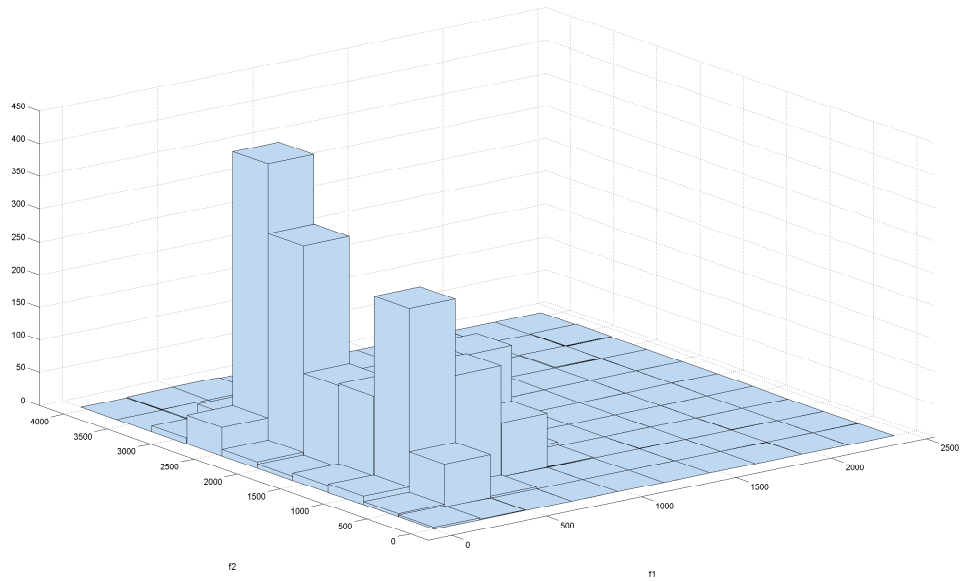


Abbildung 3.3: Aus den Daten aus Abbildung 3.2 wurde ein Histogramm erzeugt. Hieraus kann durch die Anzahl der Punkte in einem Feld der Merkmalsvektor erzeugt werden.

MATLAB liefert hierzu den Befehl $hist3(X, nbins)$[Mathworks, 2015b], welcher eine zweidimensionale Graphik in $n \times n$ gleich große Felder unterteilt. Im Anschluss wird die Anzahl der Punkte in den einzelnen Feldern gezählt und durch den Befehl $reshape(A, sz)$ in Vektorform gebracht. Anschließend wird der Vektor normiert. Zur Berechnung der Euklidnorm eines Vektors liefert MATLAB den Befehl $norm(v)$ [Mathworks, 2015c]. Die Normierung skaliert dabei jeden Vektor auf die Länge eins. Somit werden die verschiedenen Vektoren untereinander vergleichbar.

```
1 hist_diskussion=hist3([diskussion_maennl(:,2),diskussion_maennl(:,3)],...
  [10 10]);
3 vhist_diskussion=reshape(hist_diskussion,numel(hist_diskussion),1)...
  ./norm(reshape(hist_diskussion,numel(hist_diskussion),1));
```

Listing 3.4: Die zweidimensionale Grafik wird in $n \times n$ große Felder unterteilt, um anschließend den Merkmalsvektor zu erzeugen.

Anschließend werden die Vektoren über den euklidischen Abstand miteinander verglichen. Wenn $\vec{x}$ und $\vec{y}$ Vektoren $\in \Re^n$ sind, ist der euklidische Abstand definiert als [Heuser, 2013]:

$$d(x, y) = \sqrt{(\sum_{i=1}^{n}(x_i - y_i)^2)} \tag{3.40}$$

Der neue Vektor wird dann der Klasse zugeordnet, zu der der euklidische Abstand am geringsten ist.

3.6 Beurteilung der Klassifikationsgüte und Zuordnen neuer Datensätze

Zur Beurteilung der Klassifikationsgüte werden die Begriffe *accuracy* und *confusion matrix* eingeführt. Die *confusion matrix* ist eine Matrix $A \in \Re^{k \times k}$, mit der Anzahl der Klassen k, wobei der Eintrag $A_{i,j}$ für $1 \leq i, j \leq k$ angibt, wie oft die Klasse i als Klasse j erkannt wird. Die *confusion matrix* eines guten Klassifikators hat somit hohe Einträge auf der Hauptdiagonalen ($i = j$). Die *accuracy* beschreibt, den Anteil der Elemente, die richtig klassifiziert wurden und berechnet sich wie folgt:

$$\text{acc} = \frac{\sum \text{ richtig klassifizierte Elemente}}{\sum \text{alle Elemente}} \tag{3.41}$$

Die Klassifizierung kann auf verschiedene Arten erfolgen. Zum einen können alle Wahrscheinlichkeiten parallel berechnet werden um danach den neuen Datensatz der Klasse mit der höchsten Wahrscheinlichkeit zuzuordnen. Diese Art der Klassifikation nennt sich flache Klassifikation. Zum anderen kann eine hierarchische Klassifikation wie in Abbildung 3.4 illustriert erfolgen. Bei der hierarchischen Klassifikation werden, anstelle von einer Entscheidung, viele Einzelfallentscheidungen getroffen. So wird zuerst entschieden, ob der Sprecher weiblich oder männlich ist. Danach wird zwischen ungeplanter und geplanter männlicher und weiblicher Rede unterschieden. Erst danach wird der Datensatz in der dritten Hierarchiestufe der endgültigen Klasse zugeordnet.

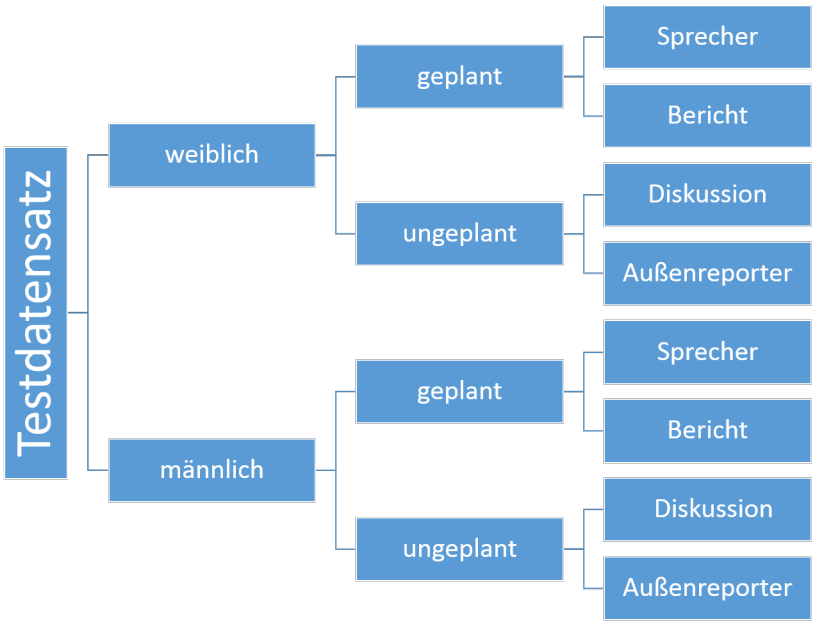


Abbildung 3.4: Hierarchie nach der die Entscheidung über die Klasse getroffen wird

4 Ergebnisse

Im Folgenden sollen die Ergebnisse beschrieben werden, die mit den in Abschnitt 3.2 beschriebenen Methoden erzeugt wurden. Hierbei lag ein Audiodatensatz vor, welcher für die Anwendung in einen Lerndatensatz, bestehend aus 70% der Dateien, und einen Testdatensatz, bestehend aus den restlichen 30% der Dateien, aufgeteilt wurde. In einem ersten Schritt wurden mit dem in Kapitel 3.1 beschriebenen Algorithmus die ersten beiden Formanten der Phoneme *a*, *i* und *u* der Daten bestimmt. Danach wurde mithilfe eines Histogramms der normierte Merkmalsvektor gebildet, um den in Abschnitt 4.1 beschriebenen nächste-Nachbar-Klassifikator anzuwenden. Für die hierarchische Klassifikation mit Gaußmischverteilungen wurden anschließend auf die Daten eine Gaußmischverteilung angepasst. Die Grafik 4.1 zeigt, wie die Gaußmischverteilungen

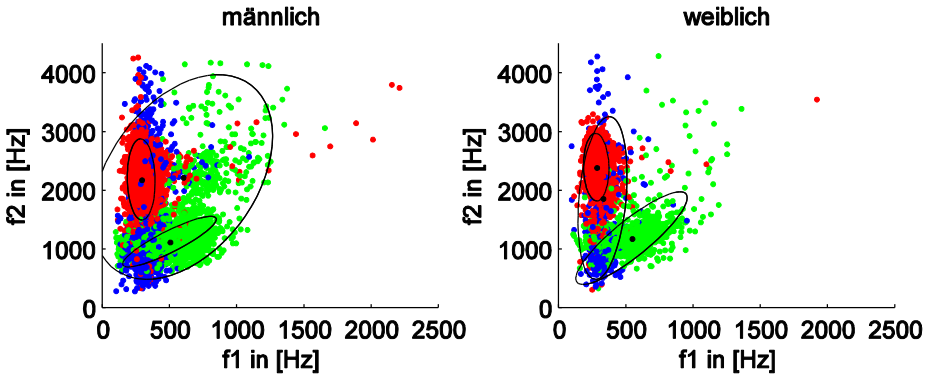


Abbildung 4.1: Anpassen der Gaußmischverteilung an die ersten beiden Formanten von weiblicher und männlicher Rede. Hierbei sind die Formanten der Vokale *a* (grün), *i* (rot) und *u* (blau) farbig dargestellt.

zunächst an die ersten beiden Formanten des weiblichen und des männlichen Sprechers angepasst wurden. In Grafik 4.2 wurden die Gaußmischverteilungen an die ersten beiden Formanten von ungeplanter und geplanter Rede eines weiblichen Sprechers und ungeplanter und geplanter Rede eines männlichen Sprechers angepasst. Aus Grafik 4.3 wird deutlich, wie die

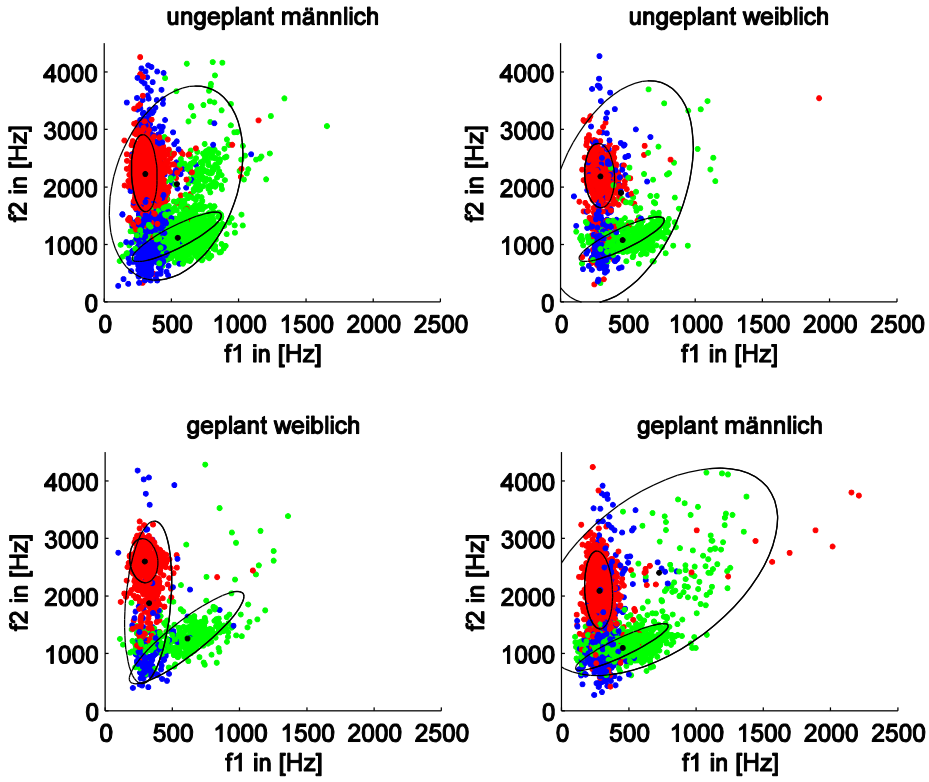


Abbildung 4.2: Anpassen der Gaußmischverteilung an die ersten beiden Formanten von ungeplanter Rede und geplanter Rede von weiblichen und männlichen Sprechern.

Gaußmischverteilungen an die endgültigen Klassen angepasst werden.

In einem weiteren Schritt wurde die Gaußmischverteilung an die Winkel des vokalen Dreiecks angepasst, wie in Abschnitt 3.4 beschrieben. Da auch hier eine hierarchische Klassifikation vorgenommen wurde, mussten die Modelle zunächst an den männlichen und den weiblichen Sprecher angepasst werden, wie in Grafik 4.4 ersichtlich. Danach wurden in Grafik 4.5 die Gaußmischverteilungen an die ersten beiden Winkel von ungeplanter und geplanter Rede von weiblichen und männlichen Sprechern angepasst. In einem abschließenden Schritt wurden die Gaußverteilungen an die in Abschnitt 3.1 definierten Klassen angepasst (Grafik 4.6). In einer weiteren hierarchischen Klassifizierung wurde eine Gaußmischverteilung an die Kombination aus Flächeninhalt und Winkel von weiblichem und männlichem Sprecher angepasst (Grafik 4.7).Danach folgte wieder die Unterscheidung zwischen männlicher und weiblicher geplanter und ungeplanter Rede, wie in Grafik 4.8 zu sehen, und das Anpassen an die endgültigen Klassen wie in Grafik 4.9 zu sehen.

4.1 Klassifikation mit dem nächste Nachbar Klassifikator

Die Klassifikation mit dem nächste-Nachbar-Klassifikator erfolgte zum einen über eine hierarchische Klassifikation, zum anderen über eine flache Klassifikation, wie bereits in Kapitel 3.6 beschrieben. Bei der flachen Klassifikation wurde folgende *confusion matrix* erzeugt:

$$\begin{bmatrix} 3 & 1 & 5 & 1 & 1 & 0 & 6 & 4 \\ 7 & 0 & 15 & 7 & 0 & 0 & 9 & 21 \\ 5 & 0 & 3 & 3 & 1 & 0 & 6 & 3 \\ 5 & 2 & 5 & 4 & 2 & 0 & 11 & 2 \\ 6 & 4 & 10 & 0 & 0 & 0 & 19 & 10 \\ 2 & 0 & 5 & 4 & 1 & 2 & 4 & 6 \\ 8 & 3 & 2 & 1 & 0 & 0 & 2 & 2 \\ 2 & 1 & 7 & 2 & 0 & 0 & 3 & 0 \end{bmatrix} \tag{4.1}$$

Dies entspricht einer *accuracy* von $0,0588$. Somit wurden $5,88\%$ der Datensätze richtig erkannt. Dies ist eine geringere Anzahl, als beim einfachen Raten

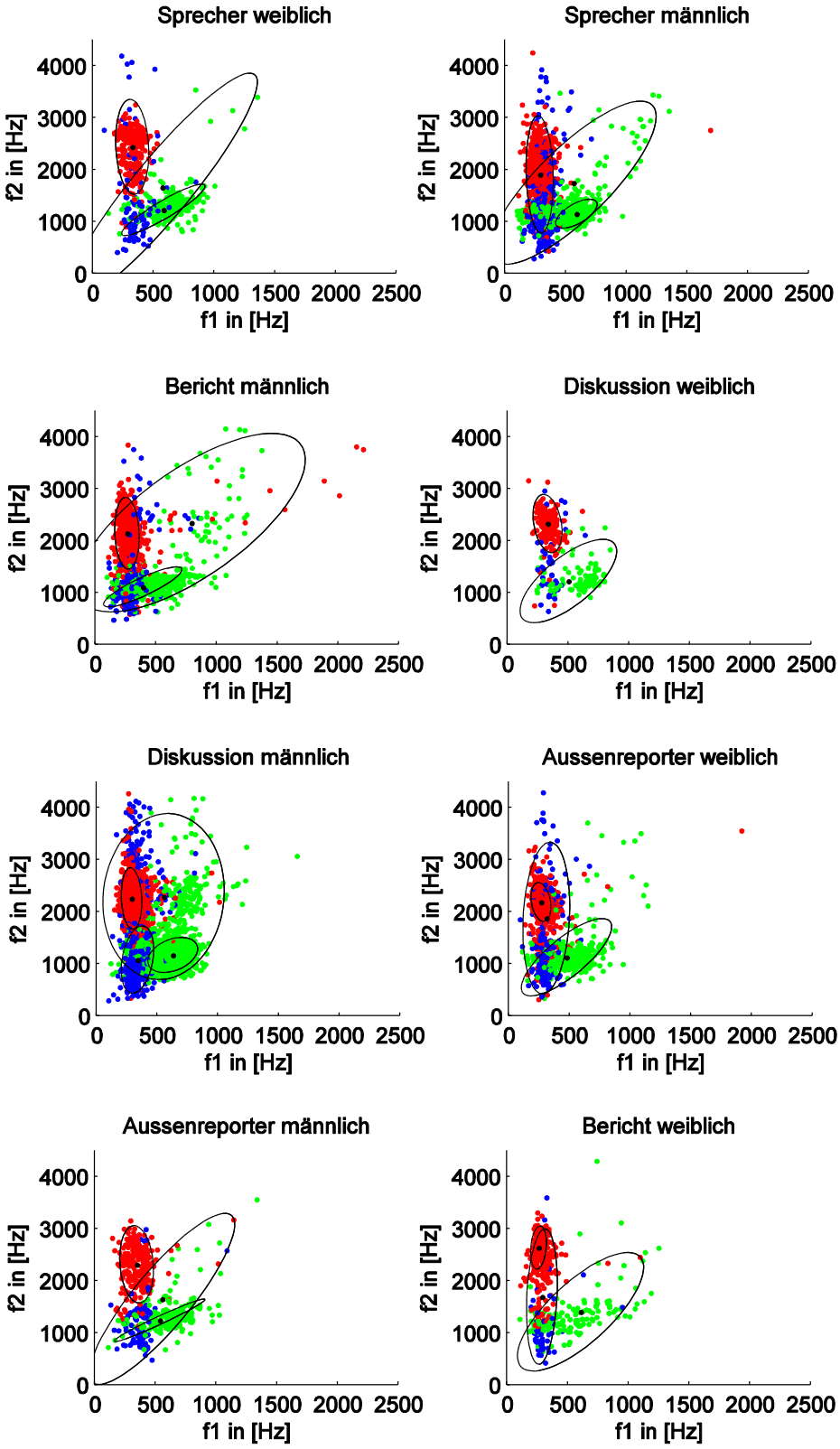


Abbildung 4.3: Anpassen der Gaußmischverteilung an die endgültigen Klassen.

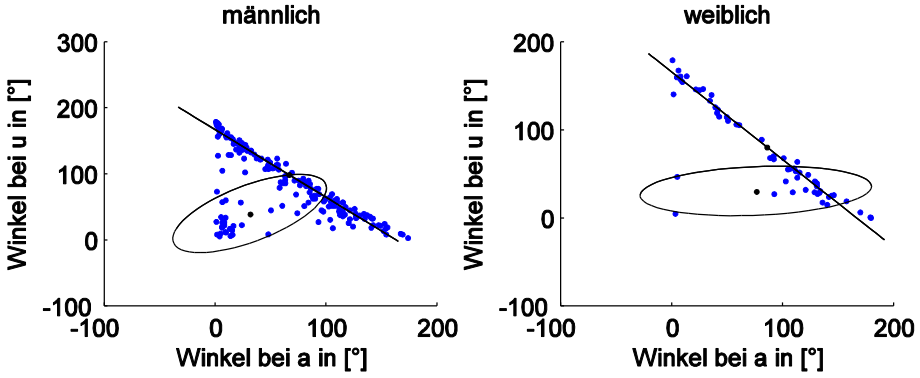


Abbildung 4.4: Anpassen der Gaußmischverteilungen an die beiden Winkel des vokalen Dreiecks von männlichem und weiblichem Sprecher.

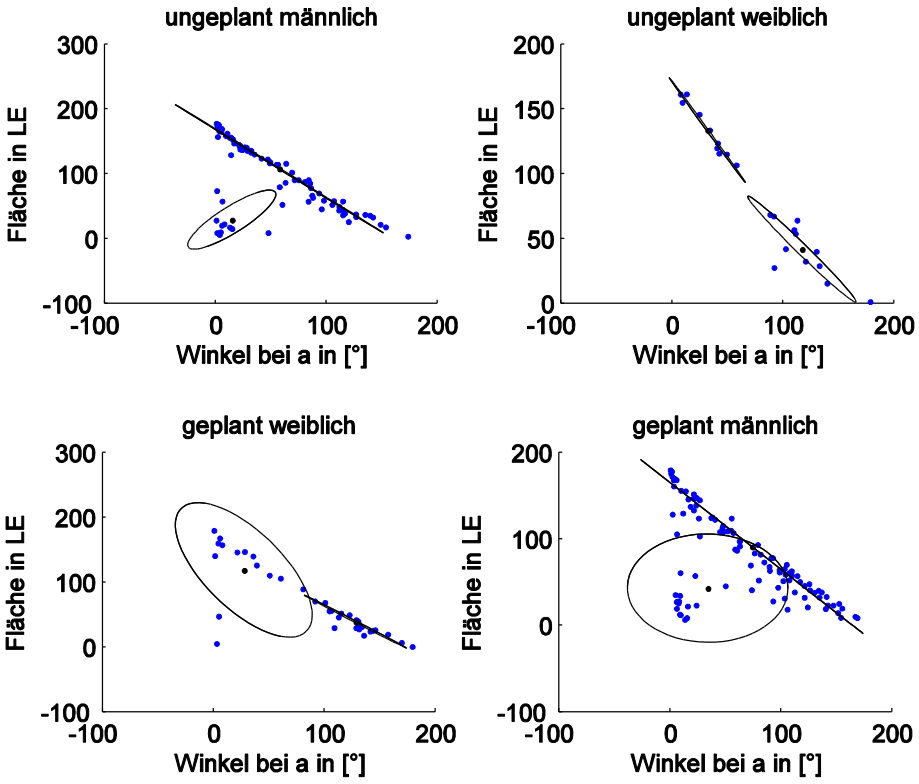


Abbildung 4.5: Anpassen der Gaußmischverteilung an die beiden Winkel des vokalen Dreiecks von geplanter und ungeplanter männlicher und weiblicher Rede.

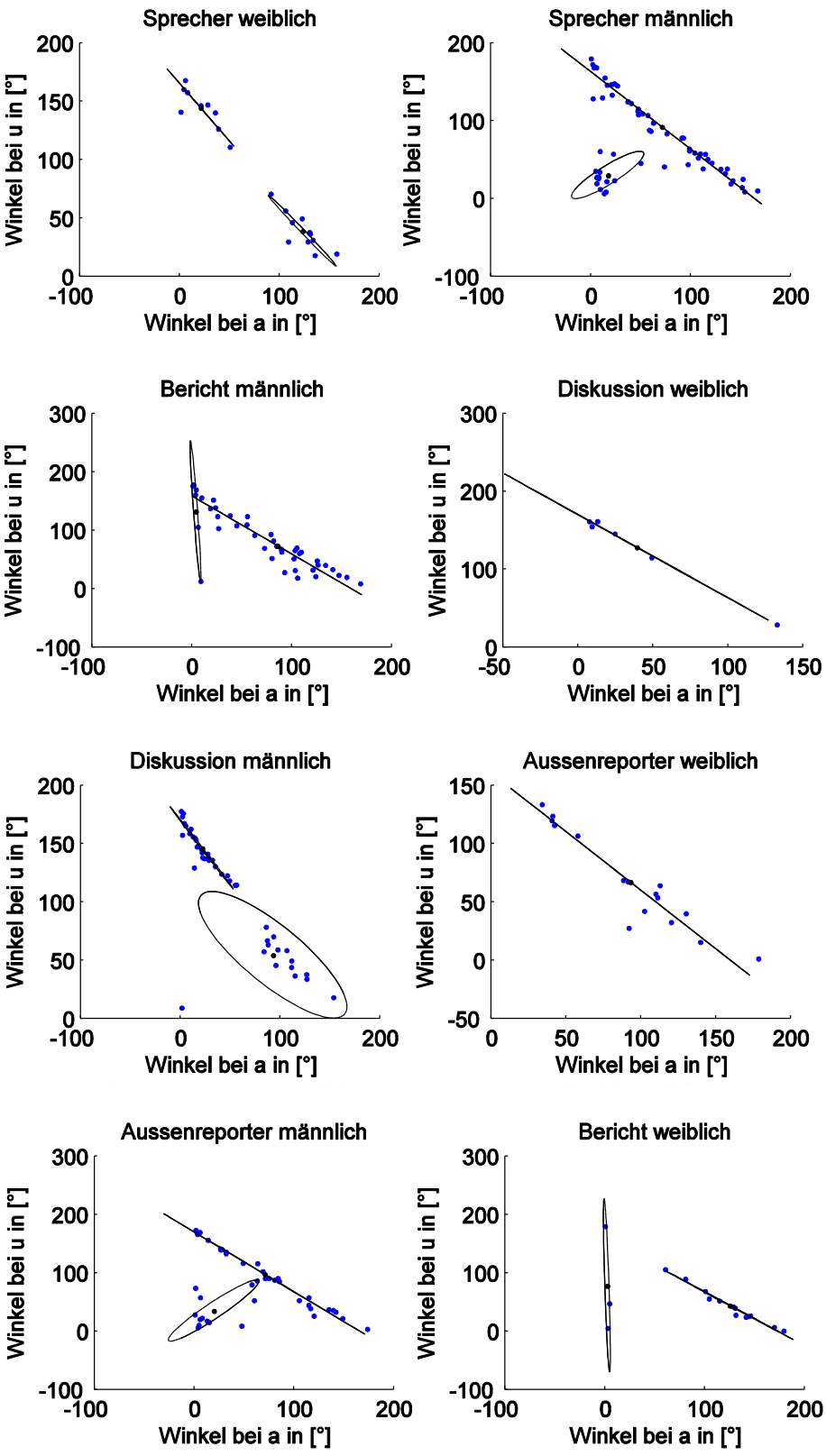


Abbildung 4.6: Anpassen der Gaußmischverteilung an die Winkel des vokalen Dreiecks der endgültigen Klassen.

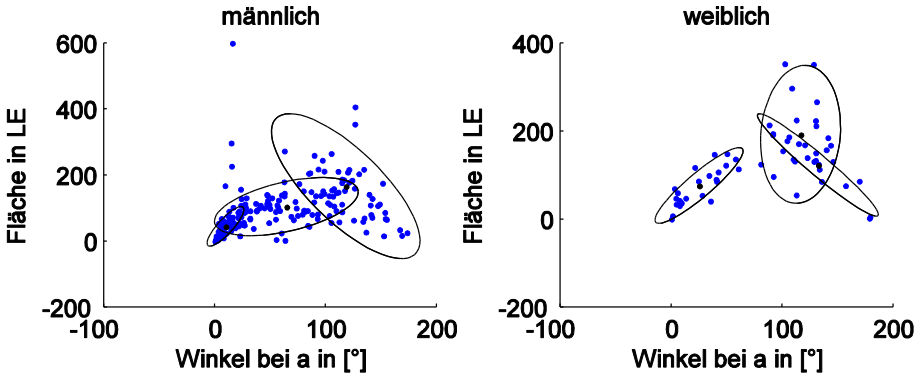


Abbildung 4.7: Anpassen der Gaußmischverteilung an die beiden Winkel der ungeplanten Rede und geplanten Rede eines weiblichen und männlichen Sprechers.

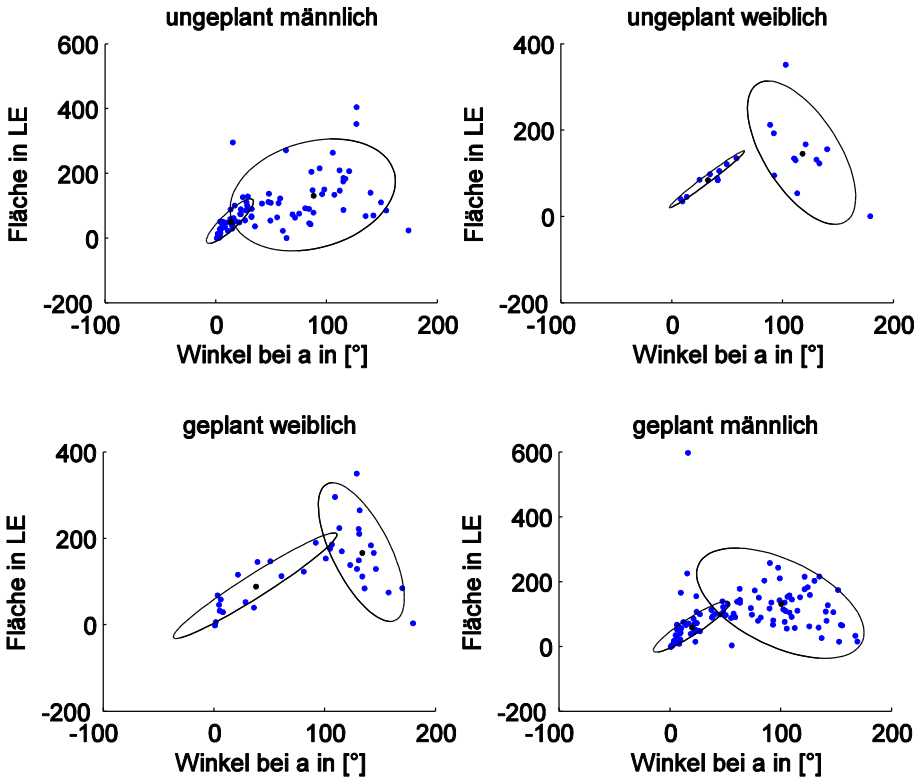


Abbildung 4.8: Anpassen der Gaußmischverteilung an die beiden Winkel der ungeplanten Rede und geplanten Rede eines weiblichen und männlichen Sprechers.

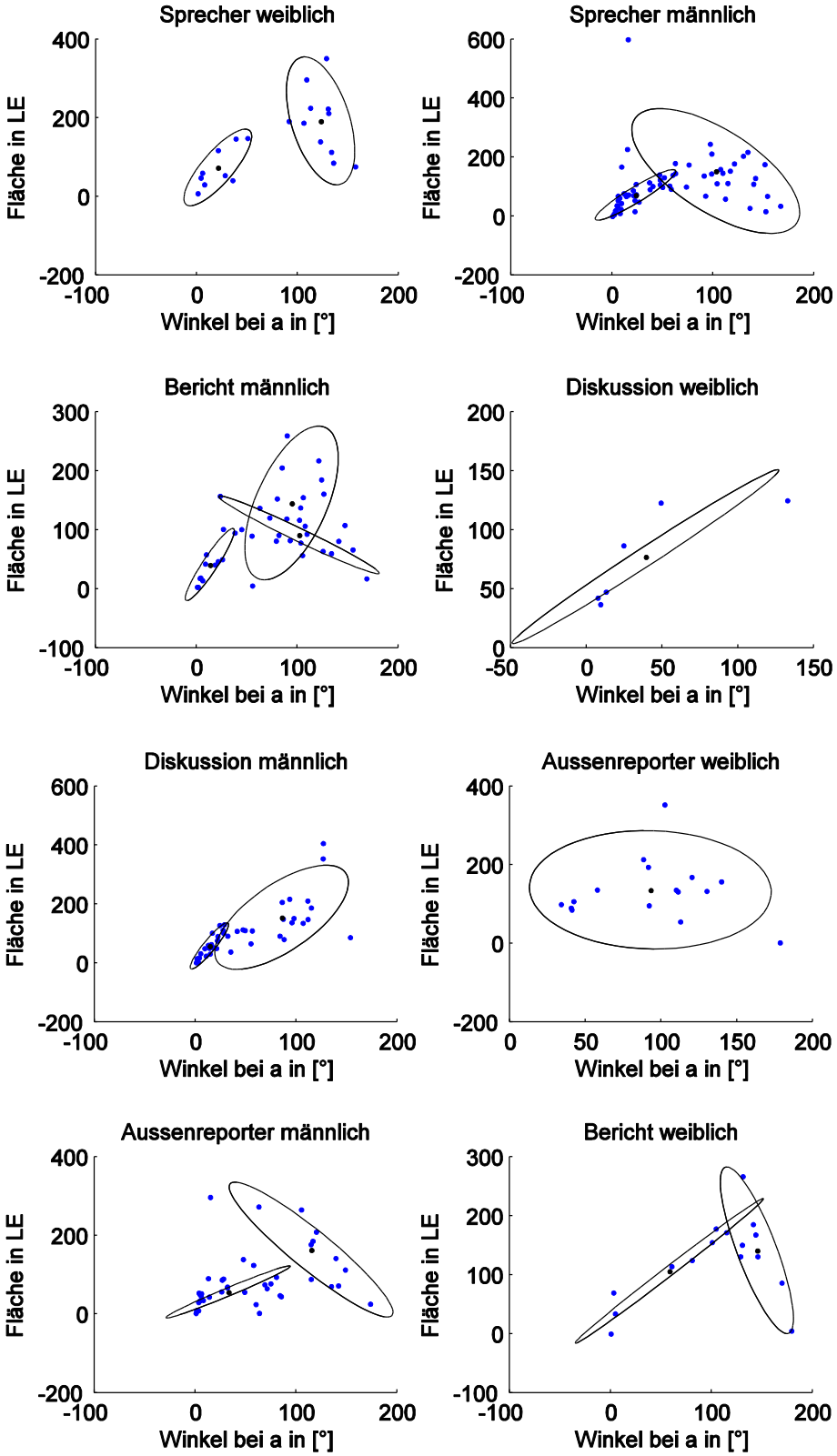


Abbildung 4.9: Anpassen der Gaußmischverteilung an die beiden Winkel der ungeplanten Rede und geplanten Rede eines weiblichen und männlichen Sprechers.

erreicht worden wäre. Bei der hierarchischen Klassifikation wurden die Datensätze besser erkannt. Bei der Unterscheidung zwischen Mann und Frau wurde zunächst folgenden *confusion matrix* erzeugt:

$$\begin{bmatrix} 56 & 29 \\ 112 & 41 \end{bmatrix} \tag{4.2}$$

Dies entspricht einer *accuracy* von $0,4076$. Hierbei entspricht Klasse 1 dem Erkennen als Frau und Klasse 2 dem Erkennen als Mann. In einem weiteren Schritt wurde zwischen weiblicher geplanter Rede (Klasse 1), weiblicher ungeplanter Rede (Klasse 2), männlicher geplanter Rede (Klasse 3) und männlicher ungeplanter Rede (Klasse 4) unterschieden. Hierbei wurde folgende *confusion matrix* erhalten:

$$\begin{bmatrix} 9 & 14 & 12 & 1 \\ 36 & 18 & 25 & 1 \\ 17 & 12 & 18 & 2 \\ 24 & 19 & 27 & 3 \end{bmatrix} \tag{4.3}$$

Dies entspricht einer *accuracy* von $0,2017$. Also einem richtigen Erkennen von $20,17\%$ der Datensätze. In der endgültigen Klassifikation wurde die folgende *confusion matrix* mit einer *accuracy* von $0,1362$ erreicht:

$$\begin{bmatrix} 4 & 1 & 1 & 4 & 1 & 4 & 3 & 3 \\ 4 & 5 & 5 & 10 & 4 & 10 & 1 & 18 \\ 2 & 4 & 4 & 3 & 2 & 3 & 2 & 1 \\ 3 & 1 & 3 & 6 & 6 & 7 & 4 & 1 \\ 5 & 1 & 7 & 4 & 8 & 4 & 6 & 14 \\ 1 & 0 & 2 & 4 & 3 & 4 & 4 & 5 \\ 5 & 2 & 1 & 7 & 1 & 1 & 1 & 0 \\ 0 & 2 & 6 & 3 & 4 & 0 & 0 & 0 \end{bmatrix} \tag{4.4}$$

Somit wurden hierbei $13,62\%$ der Datensätze richtig erkannt, was eine deutliche Verbesserung gegenüber der flachen Klassifizierung ist, jedoch nur minimal besser als eine zufällige Zuordnung von den Klassen wäre.

4.2 Kombination von Winkel und Fläche

Bei der hierarchischen Klassifikation mit dem Tupel bestehend aus Winkel und Fläche wird im ersten Schritt eine *accuracy* von $0,6053$ erreicht. Dies entspricht dem richtigen Erkennen von $60,56\%$ der Datensätze.

$$\begin{bmatrix} 11 & 13 \\ 2 & 12 \end{bmatrix} \tag{4.5}$$

Bei der Unterscheidung von weiblicher und männlicher geplanter und ungeplanter Rede wird eine *accuracy* von $0,2632$ erreicht.

$$\begin{bmatrix} 5 & 3 & 2 & 3 \\ 0 & 1 & 0 & 0 \\ 4 & 1 & 0 & 6 \\ 1 & 7 & 1 & 4 \end{bmatrix} \tag{4.6}$$

Im letzten Klassifikationsschritt wird eine abschließende, endgültige *accuracy* von $0,1842$ erreicht, mit folgender *confusion matrix.*

$$\begin{bmatrix} 2 & 0 & 3 & 1 & 3 & 0 & 1 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 4 & 0 & 0 & 0 \\ 1 & 2 & 5 & 0 & 4 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & 0 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.7}$$

Somit wurden insgesamt $18,42\%$ der Datensätze bei der hierarchischen Klassifikation mit einem Tupel bestehend aus Winkel und Fläche richtig klassifiziert. Im Gegensatz zu der in Abschnitt 3.5 genannten Klassifikation mit dem nächste-Nachbar-Klassifikator wurde somit eine Verbesserung um $4,8\%$ erreicht.

4.3 Kombination von zwei Winkeln

Bei dem Modell mit zwei Winkeln wird im ersten Schritt eine *accuracy* von $0,6579$ erreicht. Dies entspricht einem richtigen Erkennen von $65,79\%$ der Datensätze.

$$\begin{bmatrix} 14 & 10 \\ 3 & 11 \end{bmatrix} \tag{4.8}$$

Im zweiten Schritt wird bei der Unterscheidung zwischen weiblicher und männlicher geplanter und ungeplanter Rede eine *accuracy* von $0,4211$ erreicht.

$$\begin{bmatrix} 6 & 1 & 1 & 5 \\ 0 & 0 & 0 & 1 \\ 3 & 0 & 4 & 4 \\ 1 & 4 & 2 & 6 \end{bmatrix} \tag{4.9}$$

Im letzten Schritt wird eine *accuracy* von $0,2368$ erreicht. Dies entspricht einer Verbesserung um $10,06\%$ gegenüber der Klassifikation mit dem nächste-Nachbar-Klassifikator und eine Verbesserung um $5,26\%$ gegenüber dem Modell mit Dreiecksfläche und Winkel.

$$\begin{bmatrix} 2 & 0 & 1 & 1 & 4 & 1 & 0 & 4 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 3 & 0 & 2 & 0 \\ 1 & 0 & 4 & 2 & 5 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.10}$$

4.4 Klassifikation mit 2 Formanten

Bei der hierarchischen Klassifikation im Modell mit 2 Formanten wurde zunächst beim Unterscheiden von weiblichen und männlichen Sprechern eine *accuracy* von $0,8319$ erreicht.

$$\begin{bmatrix} 65 & 20 \\ 20 & 133 \end{bmatrix} \tag{4.11}$$

In der anschließenden Unterscheidung zwischen weiblicher und männlicher geplanter und ungeplanter Rede wurde eine *accuracy* von $0,4706$ erreicht.

$$\begin{bmatrix} 29 & 2 & 2 & 3 \\ 2 & 42 & 7 & 29 \\ 28 & 6 & 6 & 9 \\ 5 & 27 & 6 & 35 \end{bmatrix} \tag{4.12}$$

Bei der endgültigen Entscheidung wurde eine *accuracy* von $0,3655$ bei acht verschiedenen Klassen (siehe Abschnitt 3.2) erreicht.

$$\begin{bmatrix} 14 & 2 & 0 & 1 & 1 & 1 & 0 & 2 \\ 1 & 24 & 6 & 1 & 22 & 0 & 4 & 1 \\ 0 & 3 & 9 & 0 & 7 & 0 & 2 & 0 \\ 11 & 2 & 2 & 1 & 6 & 0 & 2 & 7 \\ 1 & 19 & 0 & 0 & 25 & 0 & 2 & 2 \\ 0 & 5 & 3 & 1 & 9 & 1 & 3 & 2 \\ 8 & 1 & 1 & 1 & 2 & 1 & 2 & 2 \\ 2 & 0 & 0 & 0 & 1 & 0 & 1 & 11 \end{bmatrix} \tag{4.13}$$

Dies entspricht einem richtigen Erkennen von $36,55\%$ der Datensätze bei acht vorher definierten Klassen. Somit wurde eine Verbesserung von $22,93\%$ gegenüber der Klassifikation mit dem nächste-Nachbar-Klassifikator, eine Verbesserung von $18,13\%$ gegenüber dem Modell bestehend aus Dreiecksfläche und Winkel und eine Verbesserung von 12.87% gegenüber dem Modell mit 2 Winkeln, erreicht.

5 Zusammenfassung und Ausblick

Die quantitative Analyse von Sprache spielt beispielsweise bei der Sprecheridentifikation sowie bei einer Vielzahl von medizinischen Anwendungen eine bedeutende Rolle. Ziel dieser Arbeit war die Sprecherklassifikation mit Hilfe der im menschlichen Vokalraum am meisten verstärkten Frequenzen, welche auch als die ersten beiden Formanten bezeichnet werden. Zur Bestimmung der Formanten wurde die von Snell und Milinazzo beschriebene Methode des Suchens der Nullstellen des Linear Predictive Polynomial genutzt [Snell and Milinazzo, 1993]. Hierbei wurde der Burg-Algorithmus verwendet um die Übertragungsfunktion $H(z)$ vorherzusagen. Die Übertragungsfunktion hat die Form $H(z) = \frac{1}{A(z)}$ und wird, wie in [Snell and Milinazzo, 1993] beschrieben, All-Pole-Model genannt, da sie keine Nullstellen und nur Pole besitzt. Die Formanten repräsentieren gerade die Maxima des Frequenzgangs. Wie bereits im Praxisbericht [Kowsky, 2015] hergeleitet, nimmt der Betrag des Frequenzgangs an den Polstellen sein Maximum ein. Somit ist es nötig die Nullstellen des Polynoms $A(z)$ numerisch zu bestimmen. Hierzu wurde die Begleitmatrix des normierten Polynoms verwendet. Dabei wurde ausgenutzt, dass die Eigenwerte der Begleitmatrix gerade die Nullstellen des normierten Polynoms darstellen. Somit konnte zur Bestimmung der Eigenwerte der QR-Algorithmus Anwendung finden. Unter Anwendung des QR-Algorithmus wird eine Matrixfolge $A_1 = A$, $A_2 = Q_1^{-1}A_1Q_1, \ldots$, $A_{n+1} = Q_n^{-1}A_nQ_n$ rekursiv definiert, welche gegen eine Dreiecksmatrix konvergiert. Die Eigenwerte der oberen Dreiecksmatrix befinden sich auf der Hauptdiagonalen.
Zur Konstruktion des vokalen Dreiecks wurden jeweils die Mittelwerte der Formanten der Eckpunkte a, i und u benutzt. Verbindet man die jeweiligen Punkte wird ein Dreieck aufgespannt, welches unterschiedliche Charakteristika aufweist. So lassen sich hier Flächeninhalt und Winkel bestimmen.
Zur Klassifikation von Sprechern wurde von verschiedenen Methoden des maschinellen Lernens Gebrauch gemacht. Hierbei wurde zum einen die nächste-Nachbar-Methode und zum anderen die Klassifikation mit Hilfe von Gaußmischverteilungen angewandt. Bei der nächste-Nachbar-Methode

(siehe Abschnitt 3.5) wurde über die im Koordinatensystem aufgetragenen Formanten ein $n \times n$ großes Gitternetz gelegt. Danach wurde die Anzahl der Punkte innerhalb eines Feldes gezählt. Aus der jeweiligen Anzahl der Punkte ergab sich der Merkmalsvektor. Somit konnte jede Klasse durch einen Merkmalsvektor repräsentiert werden. Um einen neuen Testdatensatz zuzuordnen, wurde weiterhin auch der Merkmalsvektor des Testdatensatzes bestimmt. Danach wurde der euklidische Abstand zu den Merkmalsvektoren der einzelnen Klassen berechnet. Der Testdatensatz wurde der Klasse zugeordnet, zu der der euklidische Abstand am geringsten war.
Zur Klassifikation mit Hilfe von Gaußmischverteilungen (siehe Abschnitt 3.2) wurde an die verschiedenen Merkmale eine unterschiedliche Anzahl von Gaußverteilungen angepasst. Zum Schätzen der Parameter der Gaußmischverteilung wurde von dem EM-Algorithmus Gebrauch gemacht. Als Merkmale, zum Anpassen der Gaußmischverteilungen, wurden die ersten beiden Formanten, die beiden Winkel des vokalen Dreiecks und eine Kombination aus Dreiecksfläche und Winkel benutzt. Zur Klassifikation eines Testdatensatzes wurden die einzelnen gelernten Modelle an den Punkten des Testdatensatzes ausgewertet. Die resultierenden Wahrscheinlichkeiten wurden miteinander multipliziert. Dabei wird die Verbundwahrscheinlichkeit für eine einzelne Klasse erhalten.
Tabelle 5.1 fasst die durch alle Verfahren erlangten Ergebnisse zusammen. Zunächst wurde eine flache Klassifikation durchgeführt. Dies bedeutet, dass

Tabelle 5.1: Vergleich der *accuracy* der einzelnen Methoden

Methode	accuracy
nächste Nachbar	0,0588
nächste Nachbar hierarchisch	0,1362
GMM mit Dreiecksfläche und Winkel	0,1842
GMM mit zwei Winkeln	0,2368
GMM mit den ersten beiden Formanten	0,3655

ein neuer Testdatensatz sofort derjenigen Klasse zugeordnet wird, zu der der entsprechende Merkmalsvektor den geringsten Abstand hat. Bei einer Modellierung mit Gaußmischverteilungen, wird der Testdatensatz derjenigen Klasse zugeordnet, die die höchste Verbundwahrscheinlichkeit besitzt. Hierbei wird somit nur eine Einzelfallunterscheidung getroffen. Die nächste-Nachbar-Methode lieferte hierbei eine sehr geringe *accuracy* von $0,0588$. Bei der hierarchischen Klassifikation werden hingegen an Stelle einer Einzelfallentscheidung mehrere Teilentscheidungen getroffen. So wurde hier erst zwischen männlichem und weiblichem Sprecher unterschieden. Danach wurde die Entscheidung

zwischen weiblicher und männlicher geplanter und ungeplanter Rede getroffen. Erst danach wurde der Datensatz seiner endgültigen Klasse zugeordnet. Hierbei wurde bei der nächste-Nachbar-Methode eine *accuracy* von $0,1362$ erreicht. Dies entspricht einer Verbesserung von $7,74\%$. Vergleiche mit der Literatur [Maimon and Rokach, 2005], [Greiner et al., 1997] haben ergeben, dass die hierarchische Klassifikation grundsätzlich eine höhere *accuracy* liefert. Somit wurden bei den weiteren in Tabelle 5.1 genannten Modellierungen mit Dreiecksfläche und einem Winkel, mit zwei Formanten und mit zwei Winkeln sofort die *accuracy* der hierarchischen Klassifikation miteinander verglichen. Die flache Klassifikation mit dem nächste-Nachbar-Klassifikator erkennt $5,88\%$ der Daten richtig. Somit wird hier eine geringe Anzahl von Daten richtig klassifiziert, als beim zufälligen Zuordnen von Klassen, wobei durchschnittlich $12,5\%$ der Daten richtig klassifiziert werden würden. Die hierarchische Klassifikation mit dem nächste-Nachbar-Klassifikator ist mit einer *accuracy* von $0,1362$ nur geringfügig besser als das zufällige Zuordnen von Klassen.
Eine Verbesserung von $4,8\%$ gegenüber der hierarchischen Klassifikation mit dem nächste-Nachbar-Klassifikator wurde durch Anpassen einer Gaußmischverteilung auf die Kombination von Dreiecksfläche und Winkel erreicht. Hier wurde eine *accuracy* von $0,1842$ erreicht. Eine weitere Verbesserung wurde im Modell mit zwei Winkeln erreicht. Hier wurden bei acht verschiedenen Klassen $23,68\%$ der Daten richtig erkannt. Dies ist eine Verbesserung um $17,8\%$ gegenüber der flachen nächste-Nachbar-Klassifikation und eine Verbesserung von $11,18\%$ gegenüber dem zufälligen Zuordnen von Klassen. Die höchste *accuracy* wurde mit dem Modell der Gaußmischverteilung mit zwei Formanten erreicht. Im Gegensatz zur flachen Klassifikation lieferte dies eine Verbesserung um $30,67\%$. Bei dem Modell mit zwei Formanten und acht verschiedenen Klassen wurden $36,55\%$ der Daten richtig zugeordnet. Gegenüber der zufälligen Zuordnung von Klassen ist dies eine Verbesserung um $24,05\%$.
Ziel der Arbeit war die Unterscheidung von einzelnen Personengruppen. Eine Unterscheidung von männlichen und weiblichen Sprechern wurde in $83,18\%$ der Fälle richtig durchgeführt. Die Unterscheidung zwischen weiblicher und männlicher geplanter und ungeplanter Rede wurde in $47,06\%$ der Fälle richtig durchgeführt. Dies ist gegenüber einer zufälligen Zuordnung von 4 Klassen eine Verbesserung um $25,06\%$.
Jedoch kann die endgültige Zuordnung in Klassen noch verbessert werden. Hierbei kann von weiteren Algorithmen des überwachten Lernens Gebrauch gemacht werden. Ein möglicher Klassifikator wäre hierbei eine

Support Vector Machine (SVM). Die SVM unterteilt die Daten mit einer Hyperebene linear so, dass um die Daten ein möglichst breiter Rand ohne Objekte bleibt [Bishop, 2006]. Da im Datenraum jeder Punkt durch einen Vektor repräsentiert wird, wird der Abstand der zu der Hyperebene am nächsten liegenden Vektoren maximiert. Die Maximierung des Abstandes dient dazu, neue Daten möglichst gut klassifizieren zu können. In der Praxis sind viele Daten oft nicht linear trennbar. Beispielsweise können hier Punkte der einen Klasse innerhalb der Punktewolke einer anderen Klasse liegen. Aus diesem Grund wird, wie durch [Ben-Hur and Weston, 2010] beschrieben, ein Fehlerterm eingeführt. Oft können die Daten jedoch nicht ohne Missklassifikation voneinander getrennt werden. In diesem Fall werden die Daten in einen höher dimensionalen Vektorraum transformiert, in welchem dann eine lineare Hyperebene konstruiert werden kann. Zur Konstruktion der Hyperebene wird mit einer Kernfunktion gearbeitet. Diese erlauben die Berechnung von der Hyperebene ohne die explizite Berechnung der Transformation in einen höher dimensionalen Raum, wie in [Schölkopf et al., 1998] beschrieben. Da die SVM die Daten zunächst linear, durch eine Ebene trennt, ist auch hier eine hierarchische Klassifikation notwendig.
Neben einem Austausch des Klassifikators könnte auch eine Nutzung weiterer Features eine Verbesserung bringen. Das in Abschnitt 3.4 vorgestellte Verfahren könnte um zusätzliche Features, wie beispielsweise den ersten Formanten erweitert werden.

Das Ziel der vorliegenden Arbeit war die Unterscheidung von Sprechern. Eine geschlechterspezifische Unterscheidung ist mit einer Genauigkeit von $83,18\%$ somit gelungen. Jedoch wäre ein Ansatz mit weiteren Lernalgorithmen, wie beispielsweise der SVM empfehlenswert.

Danksagung

Die vorliegende Arbeit entstand am Fraunhofer Institut für intelligente Analyse- und Informationssysteme. An dieser Stelle möchte ich allen meinen herzlichen Dank aussprechen, die mich bei dieser Arbeit unterstützt haben.
Ein besonderes Dankeschön geht dabei an meinen Betreuer Dr. Rolf Bardeli für die interessante Aufgabenstellung die wegweisenden und konstruktiven Ideen und die hervorragende Betreuung während der Anfertigung der Arbeit. Zudem möchte ich mich bei der kompletten Abteilung für die hilfreichen Anmerkungen und Kritik bedanken.
Ganz herzlich bedanke ich mich bei meinen Betreuern Frau Prof. Dr. Babette Dellen und Herr Prof. Dr. Markus Neuhäuser für die hervorragende Betreuung am Campus und die wissenschaftliche Unterstützung.
Weiterhin bedanke ich mich bei meinen Eltern Petra Nitschke-Kowsky und Hartmut Kowsky sowie meiner Schwester Jana für die moralische Unterstützung und Ermutigung während des gesamten Studiums.

Erklärung

Ich, Lea Charlotte Sophie Kowsky, Matrikel-Nr. 52 11 56, versichere hiermit, dass ich meine Bachelorarbeit mit dem Thema

Sprecherklassifizierung anhand des vokalen Dreiecks

selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe, wobei ich alle wörtlichen und sinngemäßen Zitate als solche gekennzeichnet habe. Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt und auch nicht veröfentlicht.

Remagen, den 14. Dezember 2015

Lea Charlotte Sophie Kowsky

Literaturverzeichnis

[Ben-Hur and Weston, 2010] Ben-Hur, A. and Weston, J. (2010). A user's guide to support vector machines. In *Data mining techniques for the life sciences*, pages 223–239. Springer. 5

[Bishop, 2006] Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer. 5

[Bosch, 2006] Bosch, K. (2006). *Elementare Einführung in die Wahrscheinlichkeitsrechnung*. Springer. 2.3, 2.3

[Darby et al., 1984] Darby, J. K., Simmons, N., and Berger, P. A. (1984). Speech and voice parameters of depression: A pilot study. *Journal of Communication Disorders*, 17(2):75–85. 1

[Gray Jr and Wong, 1980] Gray Jr, A. H. and Wong, D. Y. (1980). The burg algorithm for lpc speech analysis/synthesis. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 28(6):609–615. 3.1

[Greiner et al., 1997] Greiner, R., Grove, A., and Schuurmans, D. (1997). On learning hierarchical classifications. *In ResearchIndex; The NECI Scientifc Literature Digital Libraray [Online].. Citado em*, 32:34–40. 5

[Heuser, 2013] Heuser, H. (2013). *Lehrbuch der Analysis*. Springer-Verlag. 3.3.1, 3.5

[Jänich, 2013] Jänich, K. (2013). *Lineare Algebra*. Springer-Verlag. 2.3

[Jensen, 1906] Jensen, J. L. W. V. (1906). Sur les fonctions convexes et les inégalités entre les valeurs moyennes. *Acta Mathematica*, 30(1):175–193. 3.3.1

[Kowsky, 2015] Kowsky, L. C. S. (2015). *Sprecher Klassifikation anhand des vokalen Dreiecks*. 1, 5

[Maimon and Rokach, 2005] Maimon, O. and Rokach, L. (2005). *Data mining and knowledge discovery handbook*, volume 2. Springer. 5

[Mathworks, 2015a] Mathworks, T. (2015a). Formant Estimation with LPC Coefficients. (document), 22

[Mathworks, 2015b] Mathworks, T. (2015b). Reshape An Array. 3.5

[Mathworks, 2015c] Mathworks, T. (2015c). Vector and Matrix norm. 3.5

[McRae et al., 2002] McRae, P. A., Tjaden, K., and Schoonings, B. (2002). Acoustic and perceptual consequences of articulatory rate change in parkinson disease. *Journal of Speech, Language, and Hearing Research*, 45(1):35–50. 1

[Scherer et al., 2015] Scherer, S., Morency, L.-P., Gratch, J., and Pestian, J. (2015). Reduced vowel space is a robust indicator of psychological distress: A cross-corpus analysis. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 4789–4793. IEEE. 1, 2.2

[Schnaitter, 2015] Schnaitter, J. (2015). Die menschliche Stimme. (document), 2.1

[Schölkopf et al., 1998] Schölkopf, B., Smola, A., and Müller, K.-R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural computation*, 10(5):1299–1319. 5

[Shannon, 2001] Shannon, C. E. (2001). A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(1):3–55. 2.2

[Snell and Milinazzo, 1993] Snell, R. C. and Milinazzo, F. (1993). Formant location from lpc analysis data. *IEEE Transactions on Speech and Audio Processing*, 1(2):129–134. 1, 2.2, 3.1, 5

[Stoer et al., 1989] Stoer, J., Bauer, F. L., and Bulirsch, R. (1989). *Numerische Mathematik*. Springer. 3.1

[Unbehauen, 2002] Unbehauen, R. (2002). *Systemtheorie 1: Allgemeine Grundlagen, Signale und lineare Systeme im Zeit-und Frequenzbereich*. Walter de Gruyter. 2.2

[von Grünigen, 2001] von Grünigen, D. C. (2001). *Digitale Signalverarbeitung*. Fachbuchverlag Leipzig im Carl-Hanser-Verlag. 2.2, 2.2