

# Formant-Analysen von Vokalen (1)

Jonathan Harrington

- Ellipse-Darstellung ungespannter Vokale
- Ausreißer identifizieren
- Sprecher-Normalisierung

Literatur: siehe **C Vokale** der Seminar EMU-R  
Webseite.

# Die Sprechdaten

Ungespante [ɪ ε ɔ a] Vokale Sprecher 67 (M) und Sprecherin 68 (W)

Standard-Norddeutsche Sprecher, Vokale aus 100 gelesenen Sätzen.

```
data(package="emu")
```

|             |                          |                 |
|-------------|--------------------------|-----------------|
| vowlax      | Segmentliste             |                 |
| vowlax.fdat | Trackdatei F1-F4         |                 |
| vowlax.l    | Etikettierungen          | "E" "a" "I" "O" |
| ...         |                          |                 |
| vowlax.spkr | Sprecher-Etikettierungen | "67" "68"       |

|             |                                    |
|-------------|------------------------------------|
| vowlax      | Segmentliste                       |
| vowlax.fdat | Trackdatei F1-F4                   |
| vowlax.l    | Etikettierungen "E" "a" "I" "O"    |
| ...         |                                    |
| vowlax.spkr | Sprecher-Etikettierungen "67" "68" |

Formanten (F1-F4) zum zeitlichen Mittelpunkt

mid = dcut(vowlax.fdat, .5, prop=T)

Nur F1 und F2 mid = mid[,1:2]

# 1. Ellipse-Abbildungen

vowlex  
vowlex.fdat  
vowlex.l  
vowlex.spkr

Segmentliste  
Trackdatei F1-F4  
Etikettierungen "E" "a" "I" "O"  
Sprecher-Etikettierungen "67" "68"

```
mid = dcut(vowlex.fdat, .5, prop=T)
```

```
mid = mid[,1:2]
```

Logischer Vektor um Sprecher "67" zu identifizieren

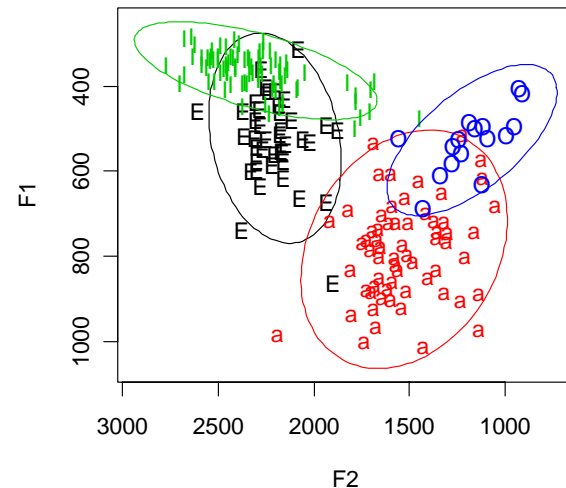
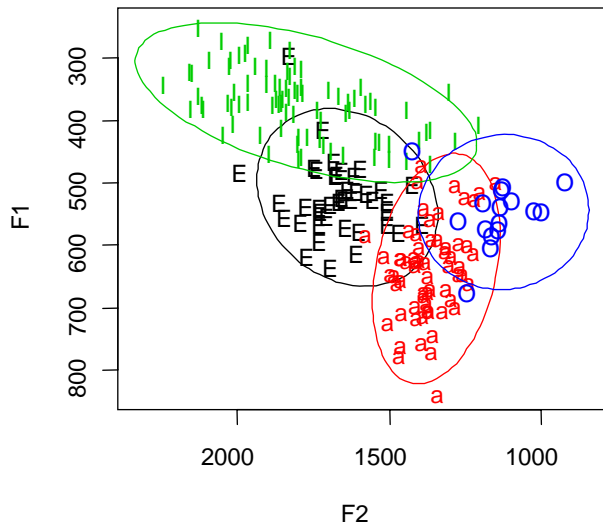
```
temp = vowlex.spkr=="67"
```

Ellipsen Sprecher 67

```
eplot(mid[temp,], vowlex.l[temp], dopoints=T,  
form=T)
```

Sprecherin 68

Mit !temp statt temp



# Mehrere Abbildungen gleichzeitig

Eine Reihe, zwei Spalten

```
par(mfrow=c(1,2))
```

```
eplot(mid[temp,], vowlax.l[temp], dopoints=T, form=T)
```

```
eplot(mid[!temp,], vowlax.l[!temp], dopoints=T, form=T)
```

wieder auf eine Reihe x eine Spalte setzen

```
par(mfrow=c(1,1))
```

# Info zu `eplot()`

`args(eplot)`

Eine Ellipse für alle Daten

```
eplot(mid[temp,])
```

`help(eplot)`

Eine Ellipse pro Kategorie

```
eplot(mid[temp,], vowelax.l[temp])
```

Mit Mittelpunkt-Beschriftung

```
eplot(mid[temp,], vowelax.l[temp], centroid=T)
```

`dopoints`: eine entsprechende Etikettierung pro Wert

```
eplot(mid[temp,], vowelax.l[temp], dopoints=T)
```

Achsen drehen (nur für F1 x F2 Abbildungen)

```
eplot(mid[temp,], vowelax.l[temp], centroid=T, form=T)
```

Keine Ellipsen

```
eplot(mid[temp,], vowlax.l[temp], centroid=T, doellipse=F)
```

Farben variieren

Keine Farben

```
eplot(mid[temp,], vowlax.l[temp], centroid=T, col=F)
```

```
eplot(mid[temp,1:2], vowlax.l[temp], centroid=T,  
col=c("pink", "red", "blue", "green"))
```

Linientyp

```
eplot(mid[temp,], vowlax.l[temp], centroid=T, linetype=T)
```

## Achsen Beschriftung

```
eplot(mid[temp,], vowlax.l[temp], form=T, main="Vokale",  
xlab="F2 (Hz)", ylab="F1 (Hz)")
```

## Bereiche setzen

```
x = c(500, 3000)
```

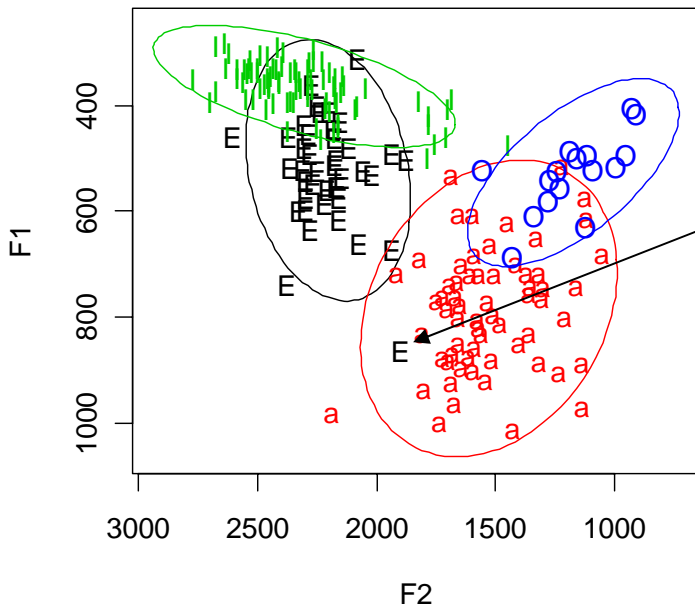
```
y = c(0, 1000)
```

```
eplot(mid[temp,], vowlax.l[temp], form=T, xlim=x, ylim=y)
```

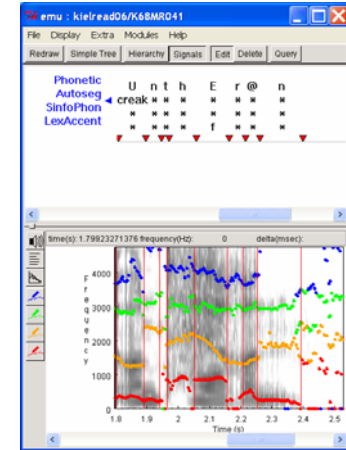


## 2. Ausreißer identifizieren

Vokale des weiblichen Sprechers  
Formanten zum zeitlichen Mittelpunkt



Ein Spektrogramm  
dieses Vokales in  
Emulabel darstellen.



vowlax  
vowlax.fdat  
vowlax.l  
vowlax.spkr

Segmentliste  
Trackdatei F1-F4  
Etikettierungen "E" "a" "I" "O"  
Sprecher-Etikettierungen "67" "68"

```
mid = dcut(vowlax.fdat, .5, prop=T)
```

Logischer Vektor: F1 ist mehr als 800 und der Vokal ist "E" und die Sprecherin ist "68"

```
temp = mid[,1] > 800 & vowlax.l == "E" & vowlax.spkr == "68"
```

Äußerung identifizieren: `vowlax[temp,]` oder `utt(vowlax[temp,])`

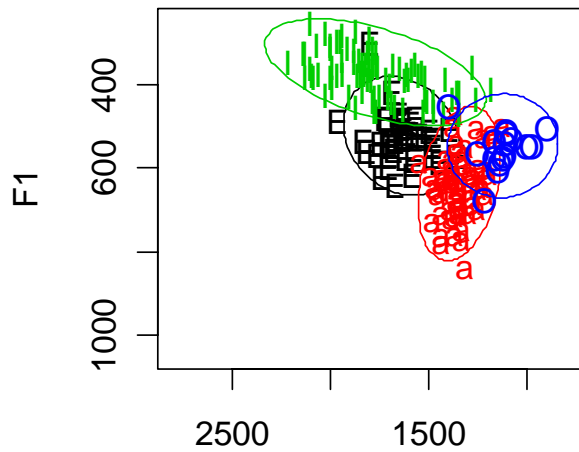
### 3. Sprechernormalisierung

Frauen haben kürzere Vokaltrakte, daher eine Erhöhung der Formanten für den selben phonetischen Vokal. Bei Kindern ist der Vokaltrakt noch kürzer und daher sind die Formanten noch höher.

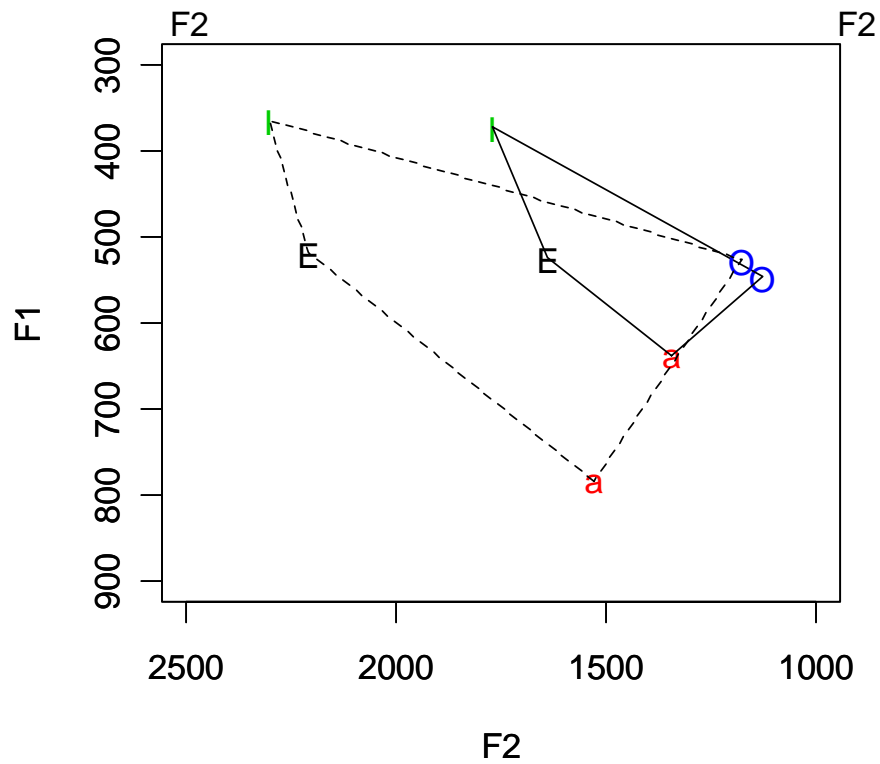
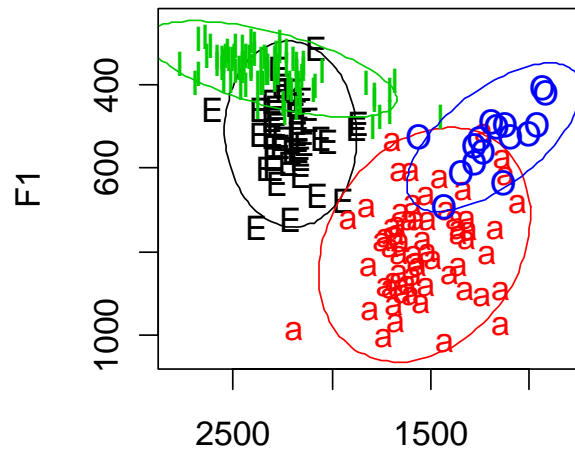
Dies führt zu **Vokalüberlappung**. zB die Formanten eines männlichen [a] haben oft denselben Wert wie diejenigen eines [o] Vokals von einem Kind.

Peterson & Barney (1952): Vokale von 76 Männern, Frauen, und Kindern.

# Sprecher 67



# Sprecher 68



# Vokalnormalisierung

- wie wird derselbe phonetische Vokal trotz dieser sprecherbedingten Unterschiede wahrgenommen?
- Wie können sprecherbedingte Merkmale im akustischen Signal entfernt werden?

# Normalisierung

Die Trennung phonetischer und sprecherbedingter Eigenschaften

intrinsisch

Wird auf einzelne Vokale durchgeführt, **ohne die anderen Vokale desselben Sprechers zu berücksichtigen**

extrinsisch

Die Normalisierung eines Vokals braucht **eine Stichprobe** von Vokalen desselben Sprechers (zB [i a u])

# Die extrinsische Normalisierung

Joos (1948)

Die Vokale eines Sprechers werden im Bezug zu seinem/ihrer [i u a] wahrgenommen. Daher ist die Entfernung von zB [ɛ] zu den Eckvokalen ca. dieselbe für verschiedene Sprecher.

Ladefoged & Broadbent (1957)

Synthese von 'Please say the word bVt', in dem F1 variiert wurde.

Je höher F1, umso mehr verschiebt sich die Wahrnehmung von 'bet' nach 'bit'

Dagegen Verbrugge et al (1976, JASA). Hörer identifizieren Vokale genau, auch wenn sie den Sprecher zum ersten Mal hören.

## Extrinsische Normalisierung (Lobanov, 1971)

(Standard-Normalisierung)

Normalisierte Formantwerte = (Formantwerte –  $m$ )/ $s$

$m$  und  $s$  sind der Durchschnitt und die Standardabweichung eines Formanten (zB F1) eines Sprechers

| <b>F1</b> | <b>F1-m</b> | <b>(F1-m) / s</b> |
|-----------|-------------|-------------------|
| 220       | -214        | -1.11             |
| 550       | 116         | 0.60              |
| 300       | -134        | -0.69             |
| 700       | 266         | 1.38              |
| 400       | -34         | -0.18             |

$m = 434$  Hz,  $s = 193.1$  Hz

$f = c(220, 550, 300, 700, 400)$

**R Befehl für (f-m)/s**

$(f - \text{mean}(f))/\text{sd}(f)$

In eine **Funktion** packen – damit sie auf beliebige Formant-Daten angewendet werden kann...



# Funktionen schreiben

## Mit default

```
probe <- function(x, y)
{
# summiert x und y
x + y
}
```

```
probe(20, 30)
```

```
probe <- function(x=20, y=30)
{
# summiert x und y
x + y
}
```

```
probe()
```

# Funktionen schreiben

Das **Ergebnis** einer Funktion ist die letzte Zeile ohne =

```
probe <- function(x=20, y=30)
```

```
{
```

```
# summiert x und y
```

```
etwas = x + y
```

```
etwas
```

```
}
```

```
probe()
```

```
f = c(220, 550, 300, 700, 400)
```

## R Befehl für (f-m)/s

```
(f - mean(f))/sd(f)
```

In eine Funktion `snorm(f)` packen – damit sie auf beliebige Formant-Daten angewendet werden kann...

```
snorm <- function(f)
{
  (f - mean(f))/sd(f)
}
```

```
mat = cbind(c(10, 20, 30), c(5, 8, 8))
```

```
mat
```

|      | [ ,1] | [ ,2] |   |
|------|-------|-------|---|
| [1,] | 10    | 5     | Durchschnitt Spalte 1: mean(mat[,1])                                      |
| [2,] | 20    | 8     | Durchschnitt Spalte 2: mean(mat[,2])                                      |
| [3,] | 30    | 8     | Durchschnitt beider Spalten gleichzeitig:<br>apply(mat, 2, mean) [1] 20 7 |

Bedeutung von

```
apply(mat, 1, mean)
```

Durchschnitt der  
Reihenwerte

```
apply(mat, 2, sd)
```

Standardabweichung der  
Spaltenwerte

Wie kann ich unsere `snorm()` Funktion auf beide  
Spalten von `mat` anwenden?

```
apply(mat, 2, snorm)
```

|             |                                    |                                     |
|-------------|------------------------------------|-------------------------------------|
| vowlax      | Segmentliste                       |                                     |
| vowlax.fdat | Trackdatei F1-F4                   | mid = dcut(vowlax.fdat, .5, prop=T) |
| vowlax.l    | Etikettierungen "E" "a" "I" "O"    |                                     |
| vowlax.spkr | Sprecher-Etikettierungen "67" "68" | mid = mid[,1:2]                     |

Normalisierte F1 und F2 Werte zum zeitlichen Mittelpunkt,  
Sprecher 67?

Logischer Vektor

```
temp = vowlax.spkr=="67"
```

Normalisierte Werte, Sprecher 67

```
fnorm.67 = apply(mid[temp,1:2], 2, snorm)
```

Normalisierte Werte Sprecher 68

```
fnorm.68 = apply(mid[!temp,1:2], 2, snorm)
```

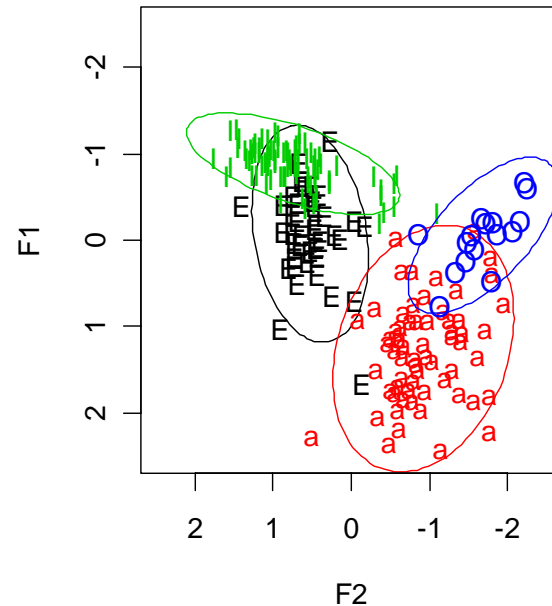
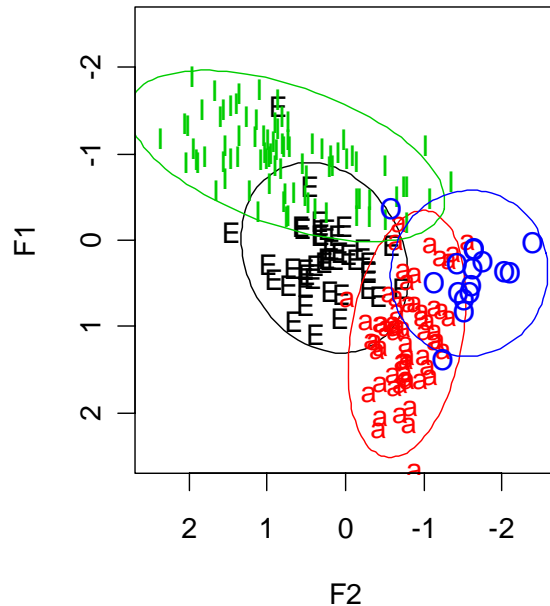
# Sprecher 67 (links), Sprecherin 68 (rechts) normalisierte F1 x F2

```
xlim = ylim = c(-2.5, 2.5)
```

```
par(mfrow=c(1,2))
```

```
eplot(fnorm.67, vowlax.l[temp], dopoints=T, form=T, xlim=xlim, ylim=ylim)
```

```
eplot(fnorm.68, vowlax.l[!temp], dopoints=T, form=T, xlim=xlim, ylim=ylim)
```



# Extrinsische Normalisierung (Nearey, 1978)

Normalisierte Formantwerte =  $\text{Log. (Formantwerte)} - k$

$k$  ist ein sprechabhängiger Konstant

# Normalisierung laut Nearey

$$f1 = c(220, 550, 300, 700, 400)$$

$$f2 = c(2000, 1800, 1900, 1500, 600)$$

1. Sprecherbedingter konstant,  $k$ , berechnen

(a) F1 und F2 in Logarithmen umwandeln

$$\text{logf1} = \log(f1)$$

$$\text{logf2} = \log(f2)$$

(b) Durchschnitt von log. F1 berechnen

$$\text{mlogf1} = \text{mean}(\text{logf1})$$

(c) Durchschnitt von Log F2 berechnen

$$\text{mlogf2} = \text{mean}(\text{logf2})$$

$k$  ist der Durchschnitt von (a) und (b)

$$k = \text{mean}(c(\text{mlogf1}, \text{mlogf2}))$$



$$f1 = c(220, 550, 300, 700, 400)$$

$$f2 = c(2000, 1800, 1900, 1500, 600)$$

Normalisierter F1 ist der Logarithmus davon minus k

$$\text{normf1} = \log(f1) - k$$

Ebenfalls für den normalisierten F2

$$\text{normf2} = \log(f2) - k$$

Funktion schreiben, nearey(x) um Nearey-Normalisierung auf eine 2-spaltige Matrix, x, (von F1 und F2 Werten) anzuwenden.

```
> form = cbind(f1, f2)
```

```
> form
```

|      | f1  | f2   |
|------|-----|------|
| [1,] | 220 | 2000 |
| [2,] | 550 | 1800 |
| [3,] | 300 | 1900 |
| [4,] | 700 | 1500 |
| [5,] | 400 | 600  |

```
> nearey(form)
```

|      | f1          | f2         |
|------|-------------|------------|
| [1,] | -1.23698013 | 0.9702948  |
| [2,] | -0.32068940 | 0.8649343  |
| [3,] | -0.92682520 | 0.9190015  |
| [4,] | -0.07952734 | 0.6826127  |
| [5,] | -0.63914313 | -0.2336780 |

```
nearey <- function(x)
# x ist eine Matrix
{
# (a) Log-Werte der Matrix
logmat = log(x)

# Vektor der Durchschnitte deren Spaltenwerte
logmat.m = apply(logmat, 2, mean)

# (b) Sprecherabhängiger konstant
k = mean(logmat.m)

# (a) – (b)
logmat – k
}
```

# Abbildung F1 x F2, Nearey-normalisierte Daten, Sprecher 67 (links), Sprecherin 68 (rechts)

```
temp = vowlax.spkr=="67"
```

```
n67= nearey(vowlax.fdat.5[temp,1:2])
```

```
n68 = nearey(vowlax.fdat.5[!temp,1:2])
```

```
ylim = c(-1.5, 0.2)
```

```
xlim = c(-.2, 1.2)
```

```
eplot(n67, vowlax.l[temp], dopoints=T, form=T, xlim=xlim, ylim=ylim, main="67")
```

```
eplot(n68, vowlax.l[!temp], dopoints=T, form=T, xlim=xlim, ylim=ylim, main="68")
```

