

Parametrisierung von Spektra

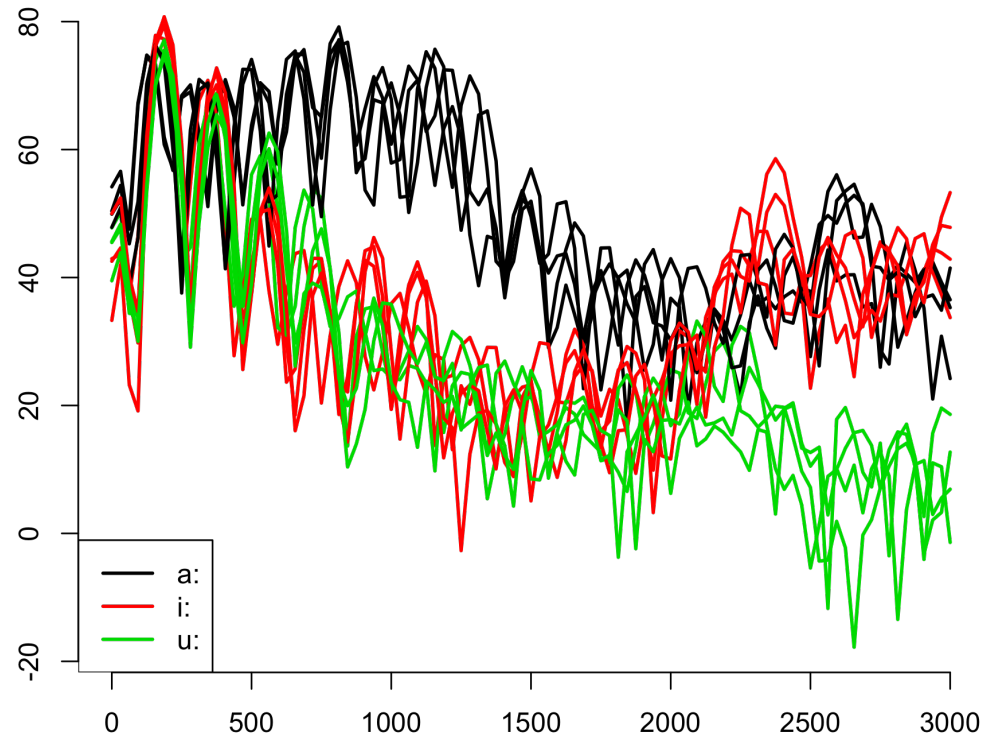
Energie-Summen

Spektrale Momemte

Spektrale Neigung

Jonathan Harrington

Das Ziel ist allen 3 Fällen ist, ein Spektrum, das sehr viele (z.B. 64, 256, 512..) Werte enthält, auf ein paar Parameter zu reduzieren.



Und zwar auf eine solche Weise, dass verschiedene phonetische Lautklassen voneinander differenziert werden können.

Spektra mit Default-Werten, berechnen, Sprachdatenbank `florian`

Segmentliste

```
v.s = emu.query("florian", "*", "phonetic=i:|u:|a:")
```

Label-Vektor

```
v.l = label(v.s)
```

Spektrale Trackdatei

```
v.dft = emu.track(v.s, "dft")
```

Spektra zum zeitlichen Mittelpunkt

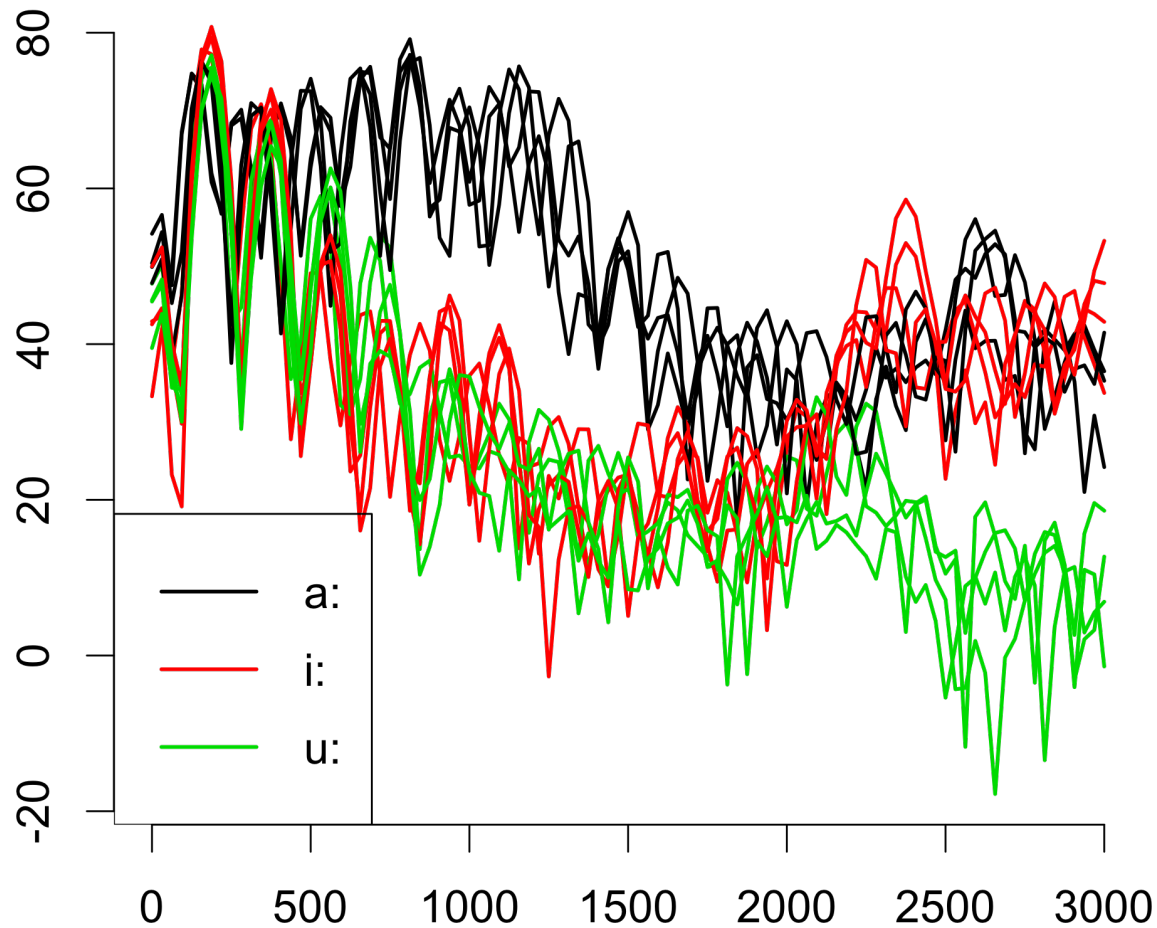
```
sp = dcut(v.dft, .5, prop=T)
```

Abbildung

```
plot(sp[,0:3000], v.l, xlab="Frequenz (Hz)", ylab="Intensitaet (dB)")
```

1. Energie-Mittelwert

2 Frequenz-Bereichen, in denen sich die Vokale unterschieden.



zB

2200-3000 Hz

noch eine?

fapply(): eine Funktion anwenden auf Spektra

dB Mittelwert 2200-3000 Hz vom ersten Segment?

```
mittel = mean(sp[1,2200:3000])
```

```
plot(sp[1,2200:3000])
```

```
abline(h = mittel)
```

Eine Funktion, `fun` auf alle Segmente einer

spektralen Matrix `m` andwenden `fapply(m, fun)`

dB-Mittelwert 2200-3000 Hz aller Segmente?

```
a = fapply(sp[,2200:3000], mean)
```

dB-Mittelwert aller Segmente vom anderen gewählten Frequenzbereich?

```
b =
```

Ellipse-Abbildung in diesem Raum

```
beide = cbind(a, b)
```

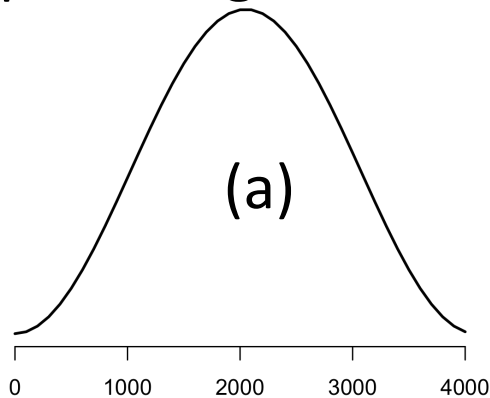
```
eplot(beide, v.l, dopoints=T)
```

m_1 : erstes spektrales Moment (spektrales Gewichtsschwerpunkt)

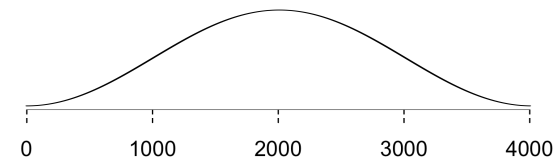
Je mehr sich die Energie in höheren Frequenzen konzentriert, umso höher m_1 (in Hz gemessen).

(a) m_1 ist ca. 2000 Hz (die Energie ist in den Frequenzen gleich verteilt)

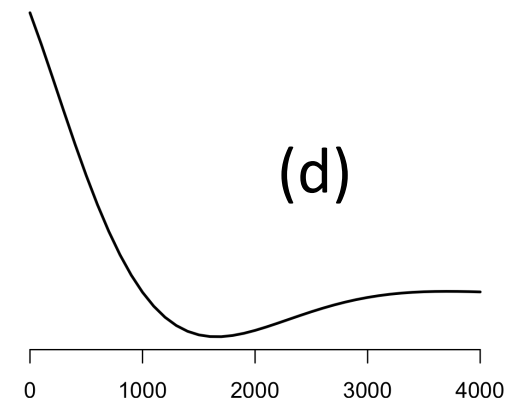
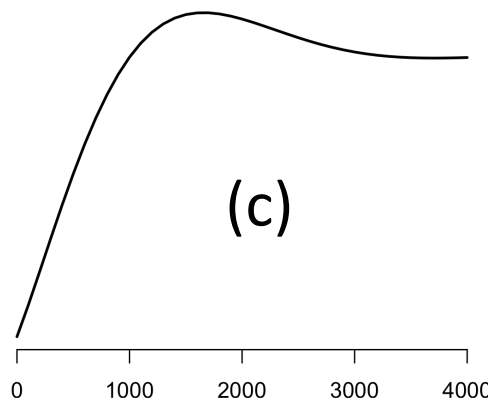
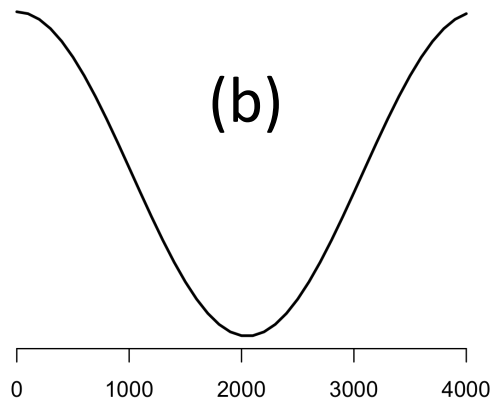
m_1 wird nicht von der dB-Skalierung beeinflusst



(b): m_1 wie für (a)

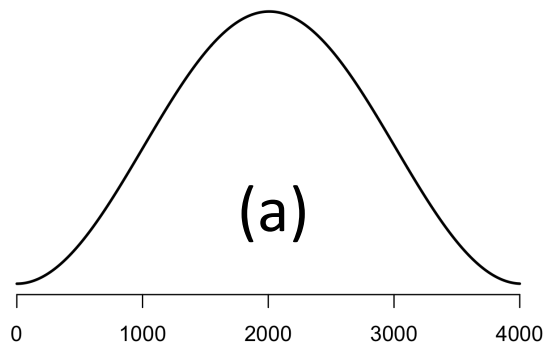


m_1 gleich/höher/tiefer im Vgl. zu (a)?



m_2 : zweites spektrales Moment (spektrale Varianz)

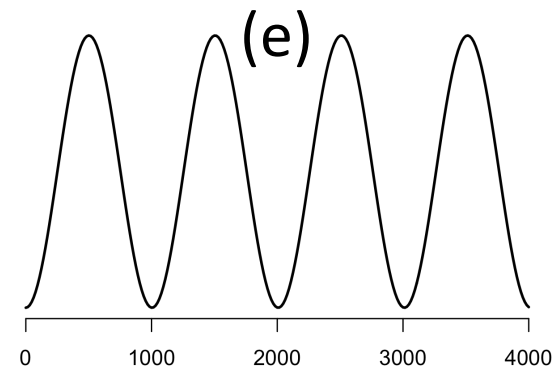
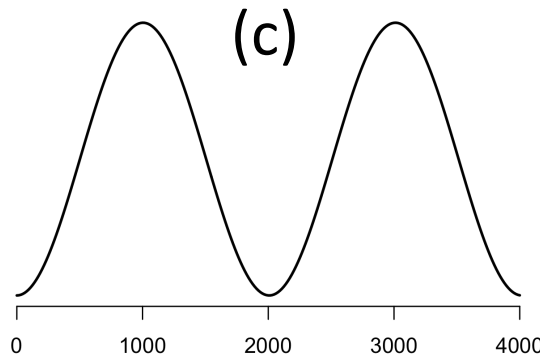
Je verteilter die Energie im Spektrum, umso höher m_2 (in Hz^2)



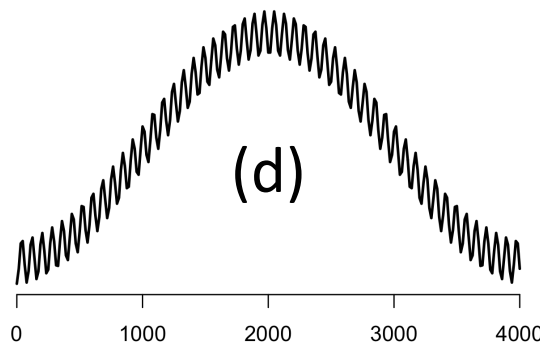
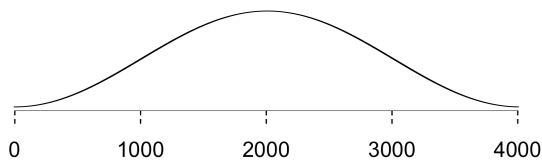
m_1 von c-f im Vgl. zu (a)?

m_2 von (c, d) > (a) m_2 von (e) im Vgl. zu (c)?

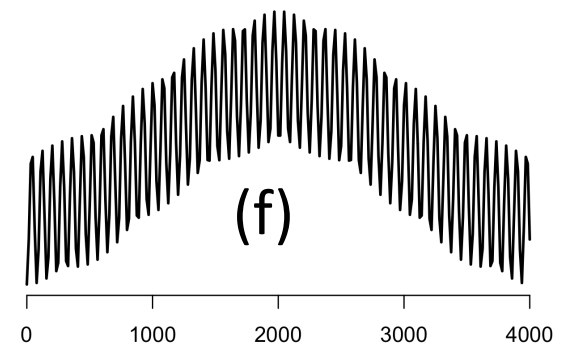
m_2 wird nicht von
der dB-Skalierung
beeinflusst



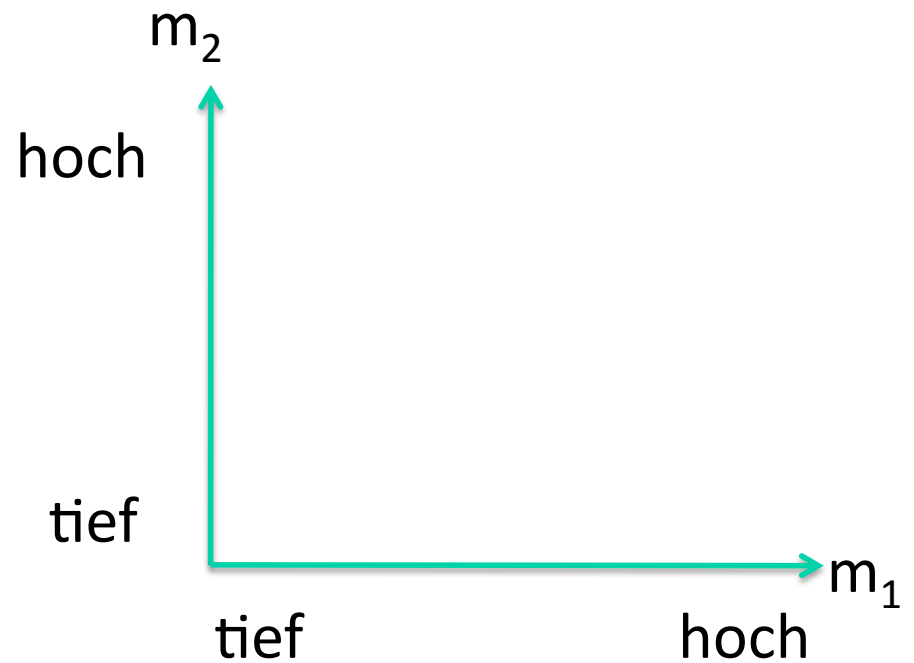
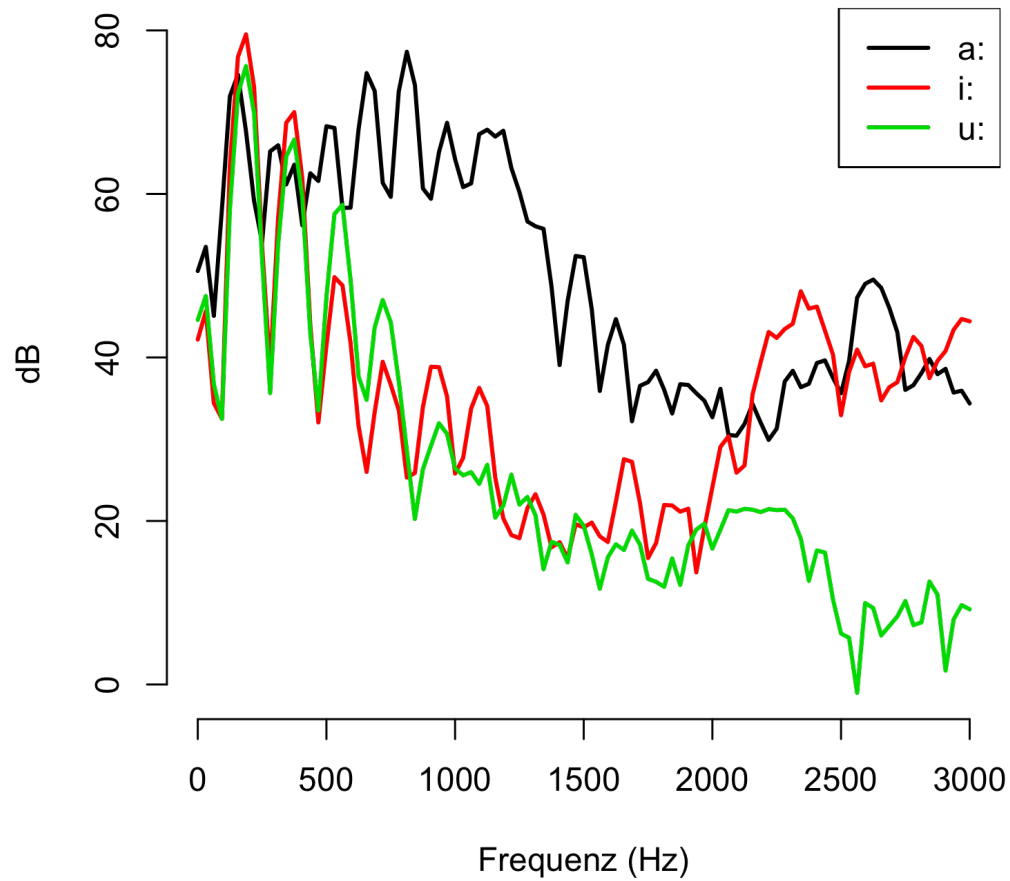
(b): m_2 wie für (a)



m_2 von (f) im Vgl. zu (d)?



Spektrale Momente einschätzen



Spektrale Momente in Emu-R

Die spektralen Momente im Bereich 0-3000 Hz vom ersten Segment?

```
p = moments( sp[1,0:3000], minval=T )
```

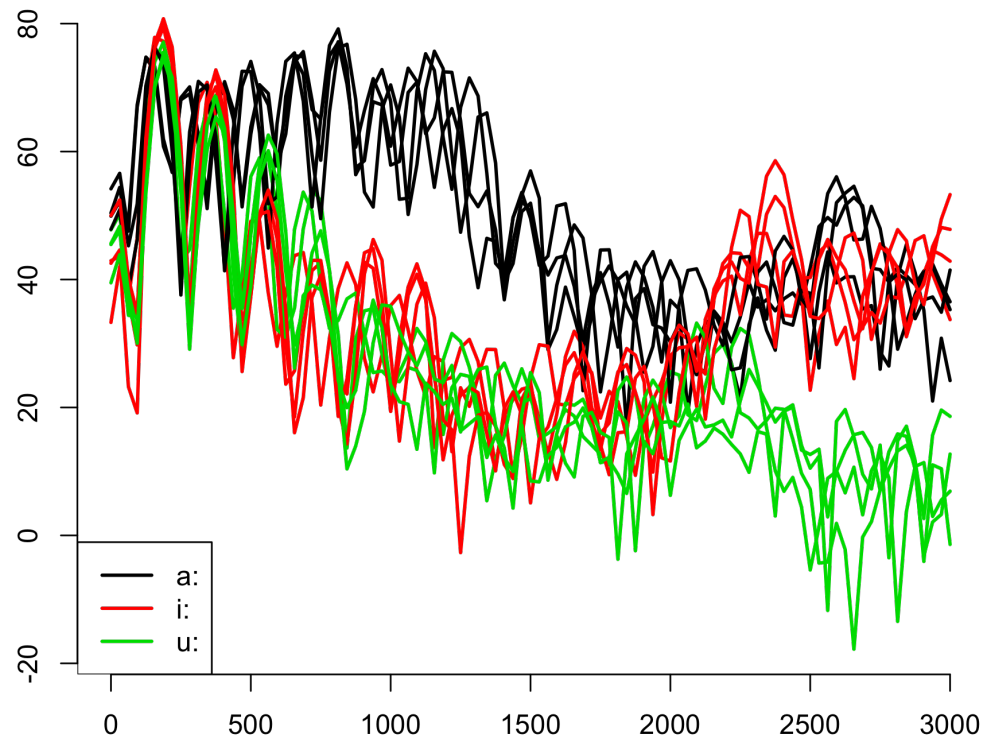
Die spektralen Momente im Bereich 0-3000 Hz aller Segmente?

```
p = fapply(sp[,0:3000], moments, minval=T)
```

Ellipse-Abbildung im Raum $m_1 \times m_2$

```
eplot(p[,1:2], v.l, dopoints=T)
```

Lineare Neigung 1500-3000 Hz



positiv, flach, oder negativ?

a: flach

i: +

u: -

Lineare Neigung 1500-3000 Hz

Vom ersten Segment

```
lm(sp[1,1500:3000] ~ trackfreq(sp[1,1500:3000]))$coef
```

Funktion, um die lineare Neigung zu berechnen:

```
spec = function(specdaten)
{
  lm(specdaten ~ trackfreq(specdaten))$coef
}
```

Funktion auf alle Segmente anwenden

```
erg = fapply(sp[,1500:3000], spec)
```

Boxplot der Neigungen als Funktion der Vokal-Kategorie

```
boxplot(erg[,2] ~ v.l)
```