

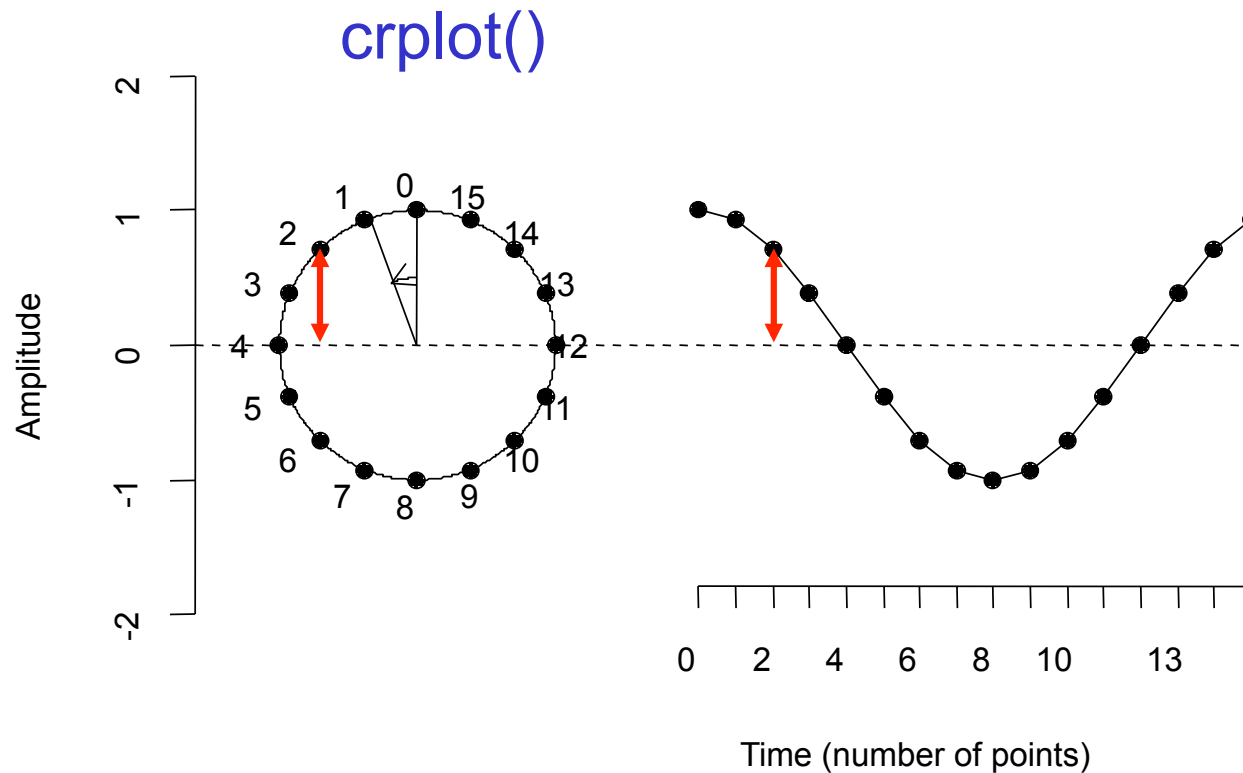
Spektrale Analysen in EMU-R: eine Einführung

Jonathan Harrington

1. Ein digitales Sinusoid
2. Fourier-Analyse
3. Ein Spektrum
4. Frequenz- und Zeitauflösung
5. Berechnung von Spektren mit Emu

1. Ein digitales Sinusoid

die Höhe über eine horizontale Linie eines Punktes, der sich in zeit-regelmäßigen Abständen (und daher mit konstanter Geschwindigkeit) im Kreis dreht.



Parameter eines digitalen Sinusoids

A: die Amplitude (Größe des Kreises)

k: die Anzahl der Schwingungen (Frequenz)

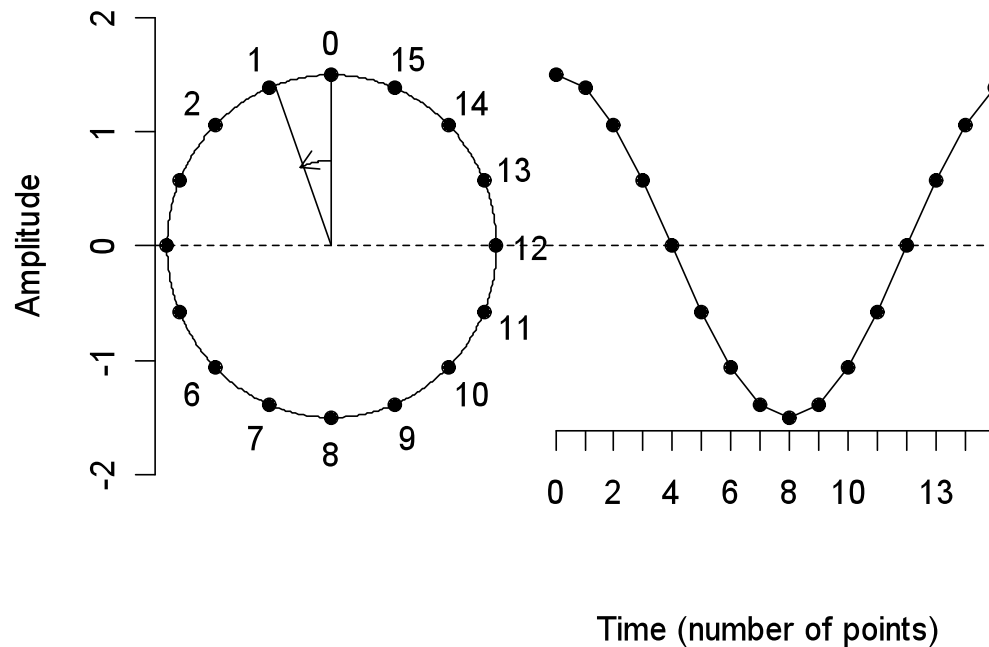
p: die Phase (wo beginnt der Punkt?)

N: aus wievielen digitalen Werten besteht der Sinusoid?

Höhere Amplitude

(Die Amplitude ist im Verhältnis zum Kreis-Radius)

`crplot(A=1.5)`

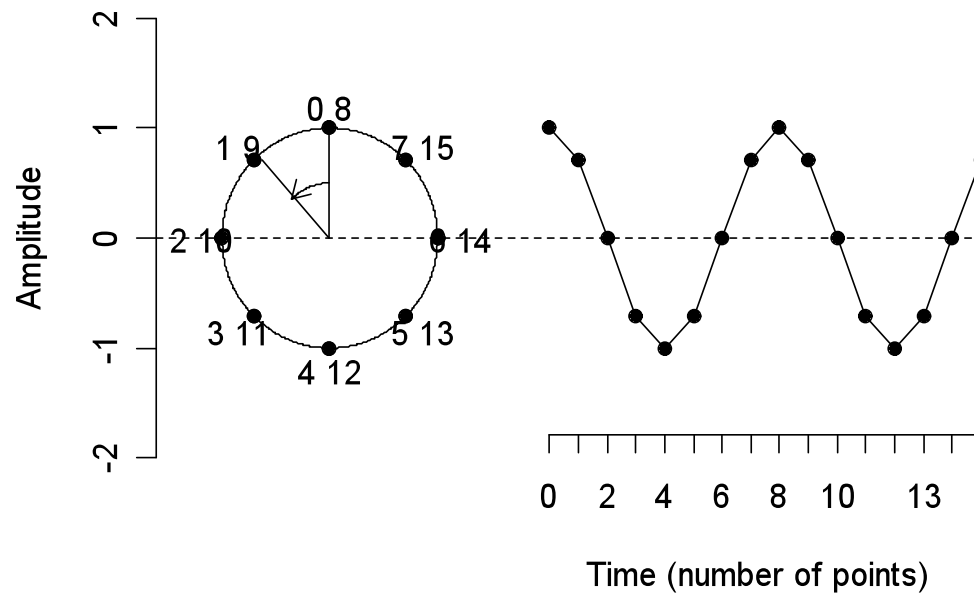


(Eine 16 Punkt digitale Cosinuswelle)

Doppelte Frequenz

k = 2 Schwingungen pro 16 Punkte

`crplot(k=2)`

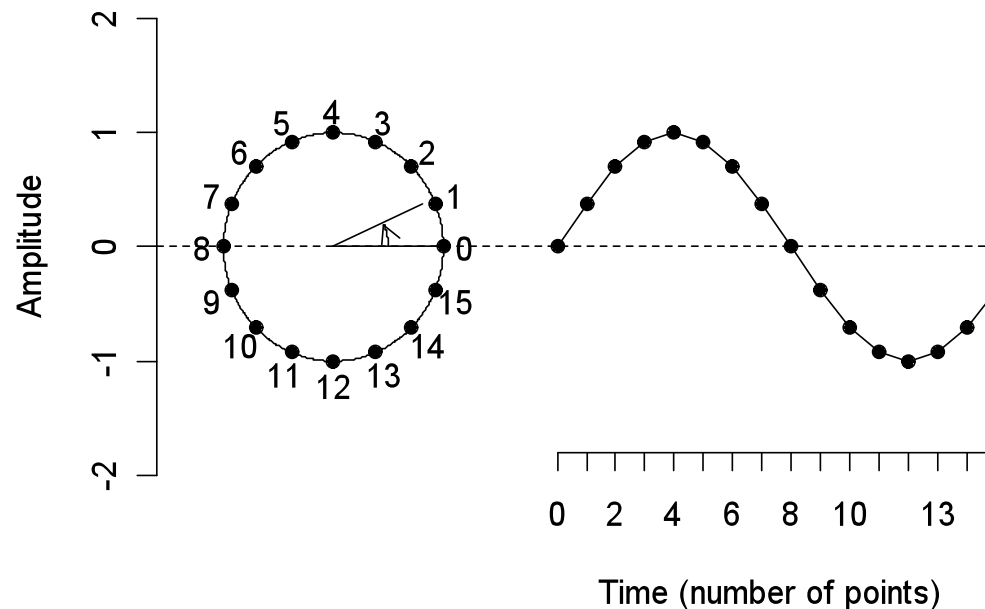


Phase

0 radian: der Punkt beginnt ganz oben

π radian: der Punkt beginnt ganz unten (= eine halbe Schwingung später)

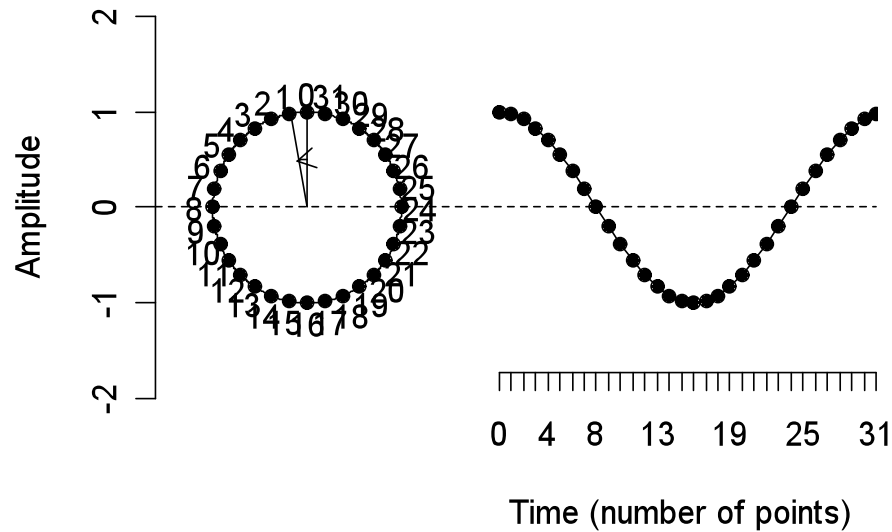
eine viertel Schwingung früher: `crplot(p=-pi/2)`



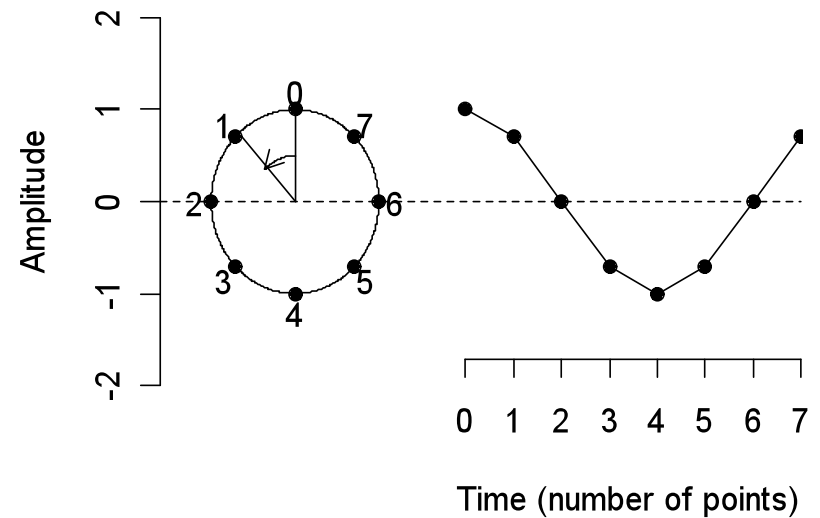
(Eine 16 Punkt digitale Sinuswelle)

Anzahl der digitalen Punkte

crplot(N=32)



crplot(N=8)



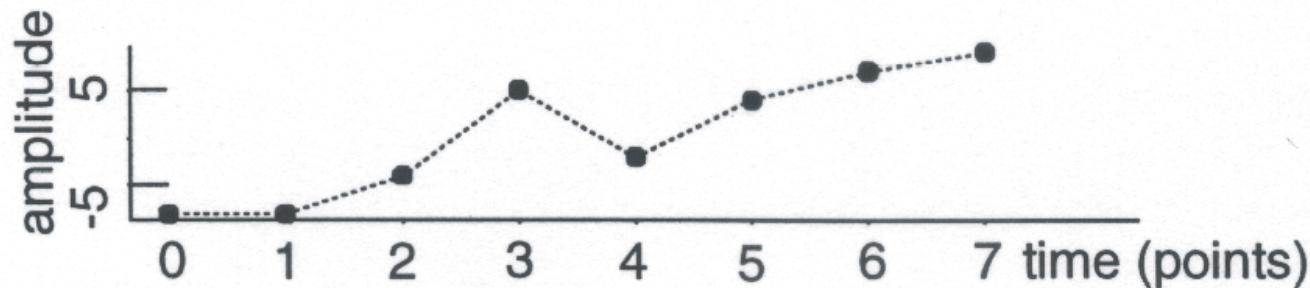
2. Die Fourier-Analyse

Die Zerlegung eines Signals in eine Reihenfolge von Sinusoiden zunehmender ganzer Frequenz-Intervallen, sodass wenn diese Sinusoiden summiert werden (= Fourier-Synthese), das Signal genau rekonstruiert wird.

Beziehung zwischen Signal-Länge und die Anzahl der Sinusoiden

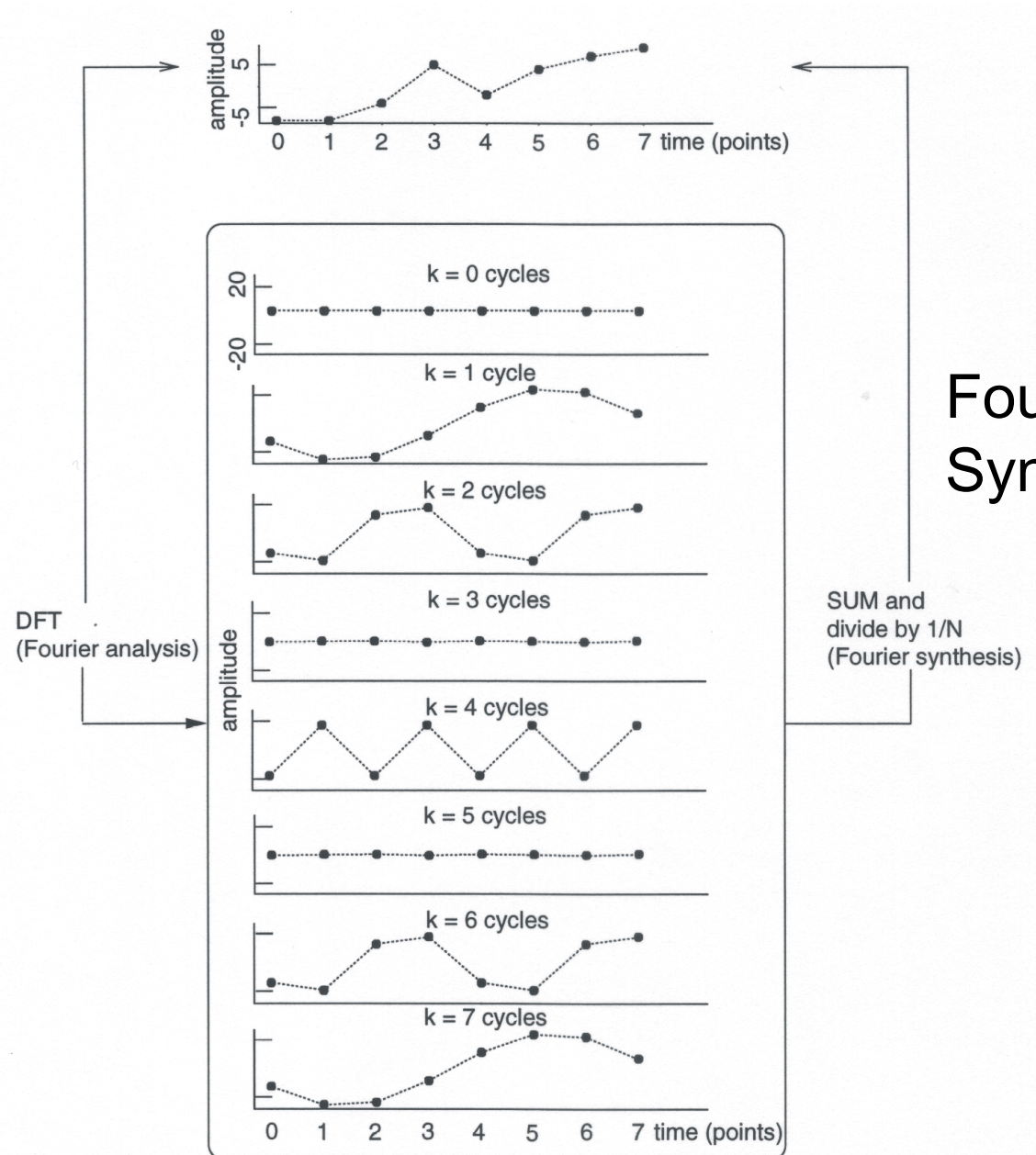
Wenn eine Fourier-Analyse auf ein N-Punkt Signal angewandt wird, dann wird immer das Signal in N Sinusoiden mit Frequenzen $k = 0, 1, 2, \dots, N-1$ Schwingungen zerlegt.

zB wollen wir eine Fourier-Analyse auf dieses 8-Punkt Signal anwenden.



Dann wissen wir schon, dass das Ergebnis davon 8 Sinusoiden sein wird, mit Frequenzen 0, 1, 2, 3, 4, 5, 6, 7 Schwingungen. Die Fourier-Analyse berechnet die Amplituden und die Phasen davon (und auf eine solche Weise, dass wenn wir die 8 Sinusoiden summieren, das obige 8-Punkt Signal genau rekonstruiert wird).

Fourier-Analyse



Die Faltung

Alle Sinusoiden mit einer Frequenz größer als $(N/2)$ Schwingungen sind Kopien (= werden gefaltet auf) Sinusoiden mit niedrigeren Frequenzen.

k

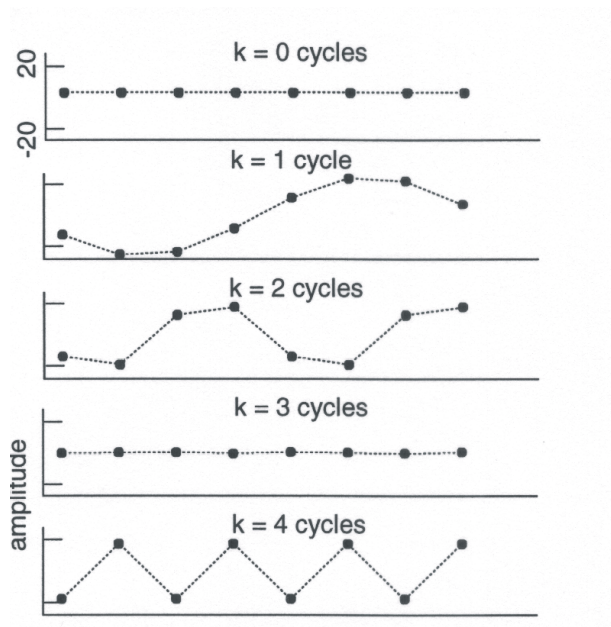
0

1

2

3

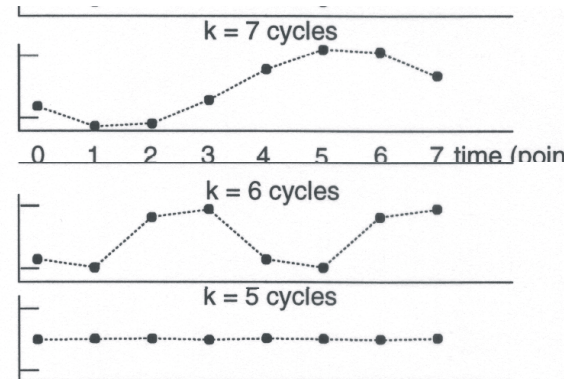
4



=

=

=



k

7

6

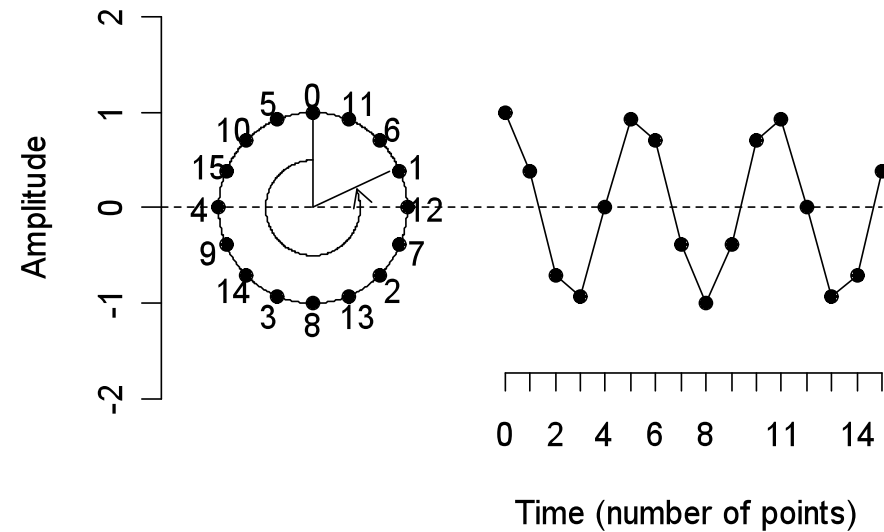
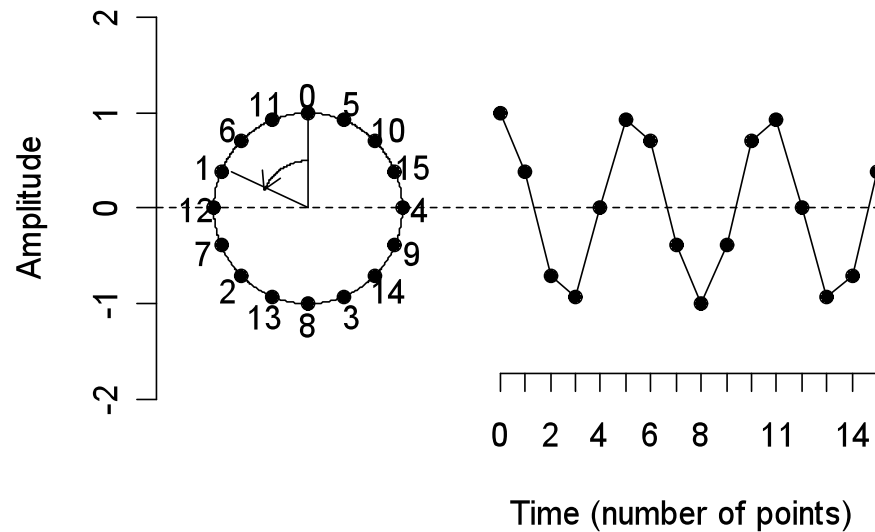
5

Die Faltung

`crplot(k=3, N=16)`

`crplot(k=13, N=16)`

verursachen dasselbe Sinusoid...

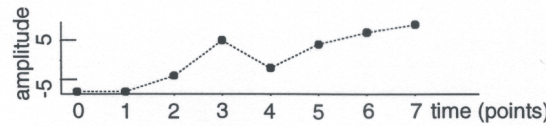


3. Ein (Amplitude) Spektrum

ist eine Abbildung der Amplitude als Funktion der Frequenz für alle Sinusoiden bis zur und inklusive der Faltung-Frequenz ($k = N/2$)

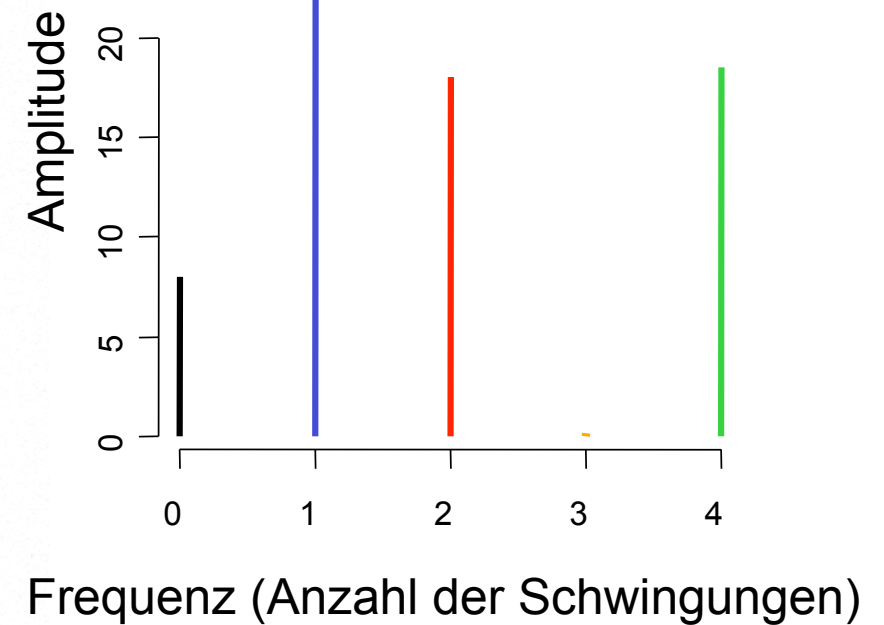
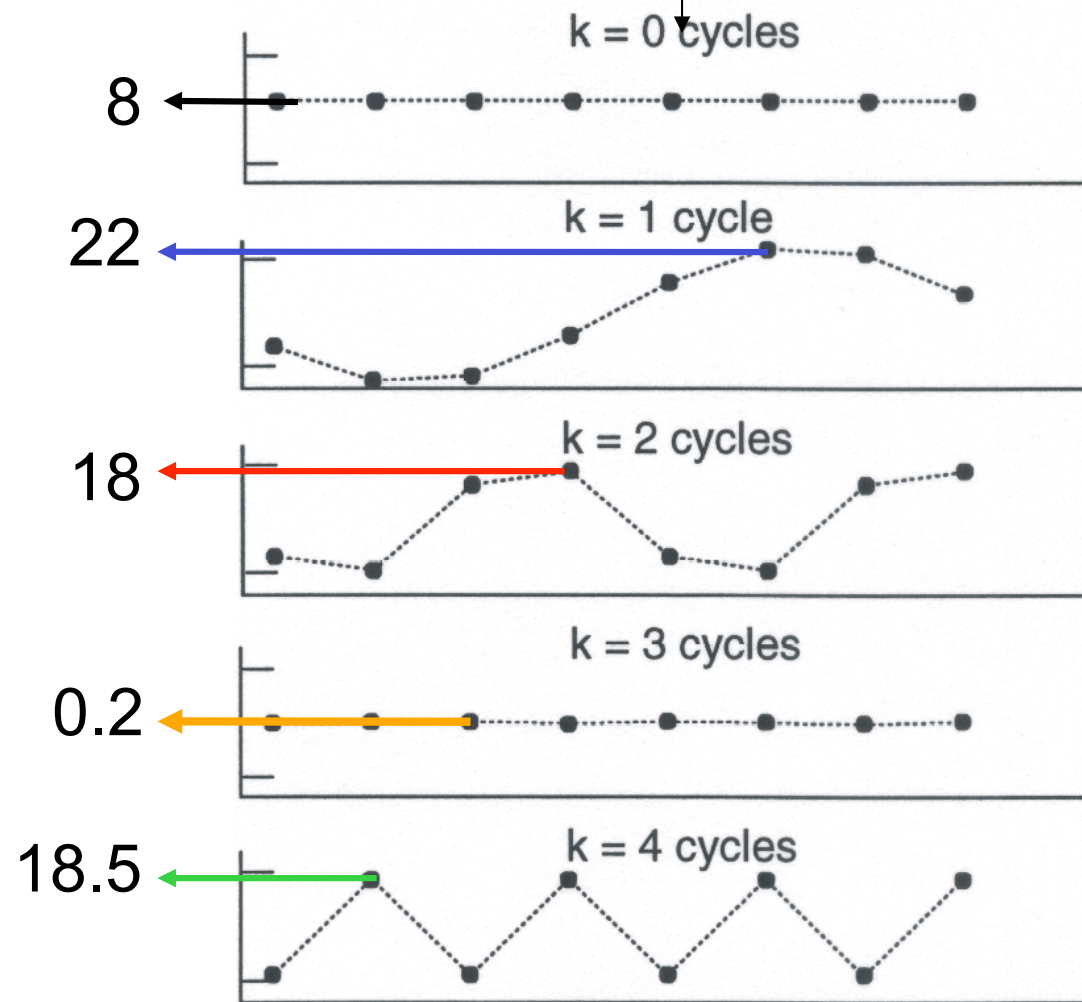
daher werden für das 8-Punkt-Signal nach der Fourier-Analyse in einem Spektrum die Amplituden der Sinusoiden mit Frequenzen 0, 1, 2, 3, 4 ($= N/2$) Schwingungen abgebildet

Zeitsignal



Fourier-Analyse

Spektrum



Schwingungen in Hertz (Hz) umrechnen

Die Umsetzung der Frequenzachse in Hz ist von der Abtastrate des Signals, **fs**, abhängig.

$$\mathbf{1. \text{ Frequenz (Hz) = Schwingungen} \times fs/N}$$

zB bei der Fourier-Analyse eines 8-Punkt-Signals bekommen wir Sinusoiden mit Schwingungen 0, 1, 2, 3, 4 (bis zur Faltung-Frequenz)

Bei $fs = 16000$ Hz entsprechen diese Schwingungen

0 Hz, 2000 Hz, 4000 Hz, 6000 Hz, 8000 Hz

$$\begin{array}{c} \updownarrow \\ = 3 \times 16000/8 \end{array}$$

Frequenz- und Zeitauflösung

1. Frequenz (Hz) = Schwingungen $\times f_s/N$
2. **der Abstand zwischen Spektralkomponenten = f_s/N Hz** (wegen 1.)
3. **die Anzahl der Spektralkomponente bis zur Faltung = $N/2 + 1$**

zB $f_s = 16000$ Hz, Zeisignal hat $N = 8$ Punkte

$N/2 + 1 = 5$ Spektralkomponente mit einem jeweiligen Frequenzabstand von $16000/8 = 2000$ Hz.

wie man hier gesehen hat...

0 Hz, 2000 Hz, 4000 Hz, 6000 Hz, 8000 Hz


$$= 3 \times 16000/8$$

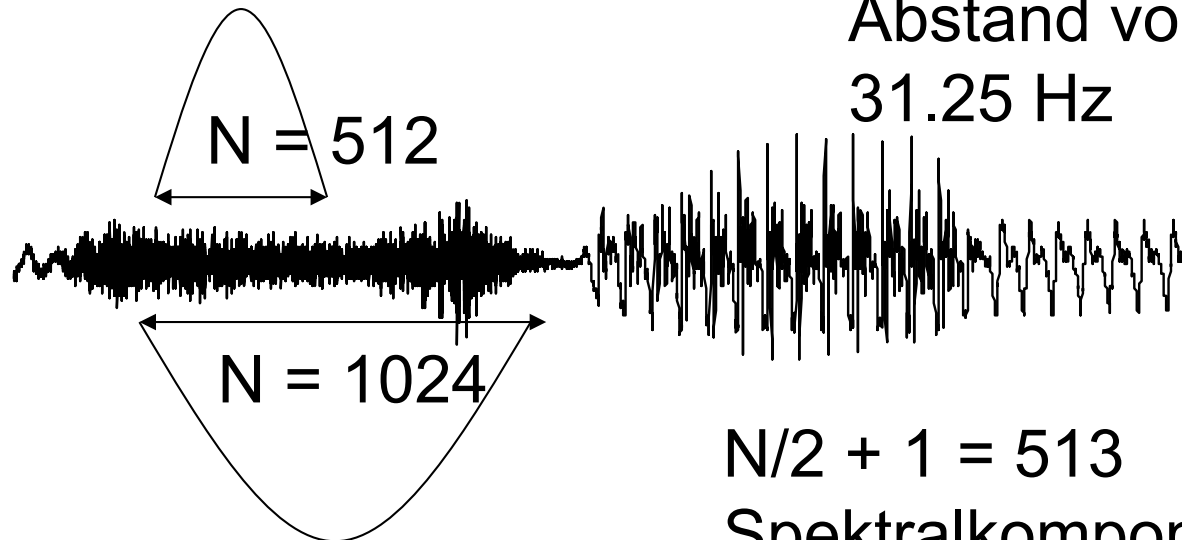
2. der Abstand zwischen
Spektralkomponenten = f_s/N Hz (wegen 1.)

**4. Daher, je größer N (also je grober die
Zeitauflösung), umso feiner/detaillierter das
Spektrum...**

Frequenz und Zeitauflösung

4. Daher, je größer N (also je grober die Zeitauflösung), umso feiner/detaillierter das Spektrum...

$f_s = 16000 \text{ Hz}$



$$N/2 + 1 = 257$$

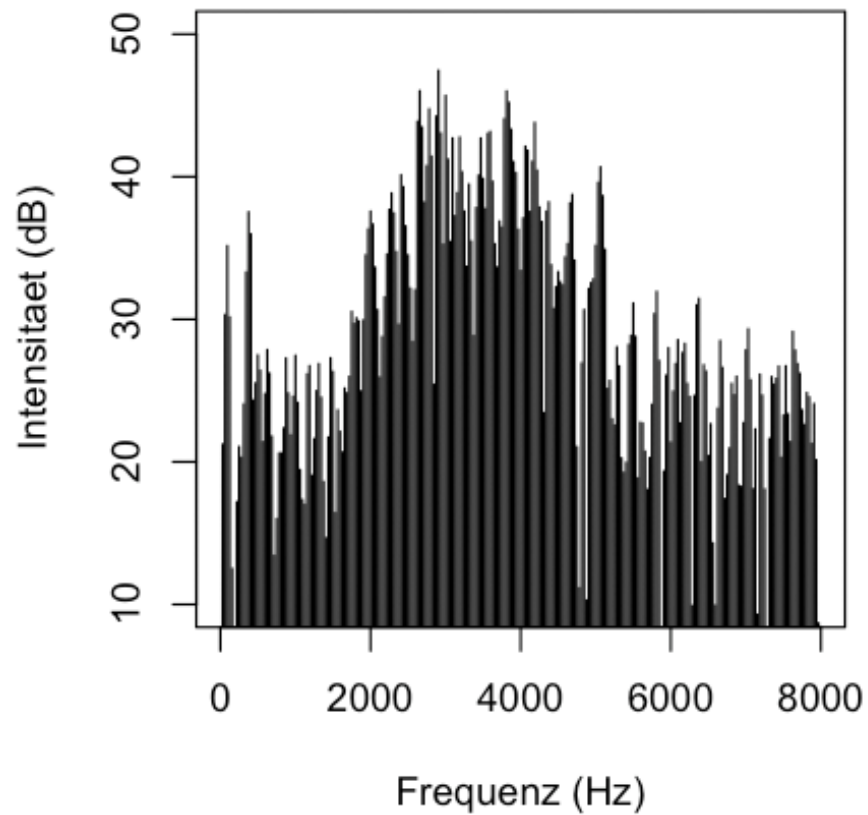
Spektralkomponente zwischen
0 und 8 kHz mit einem
Abstand von $16000/512 =$
31.25 Hz

$$N/2 + 1 = 513$$

Spektralkomponente zwischen 0
und 8 kHz mit einem Abstand
von $16000/1024 = 15.625 \text{ Hz}$

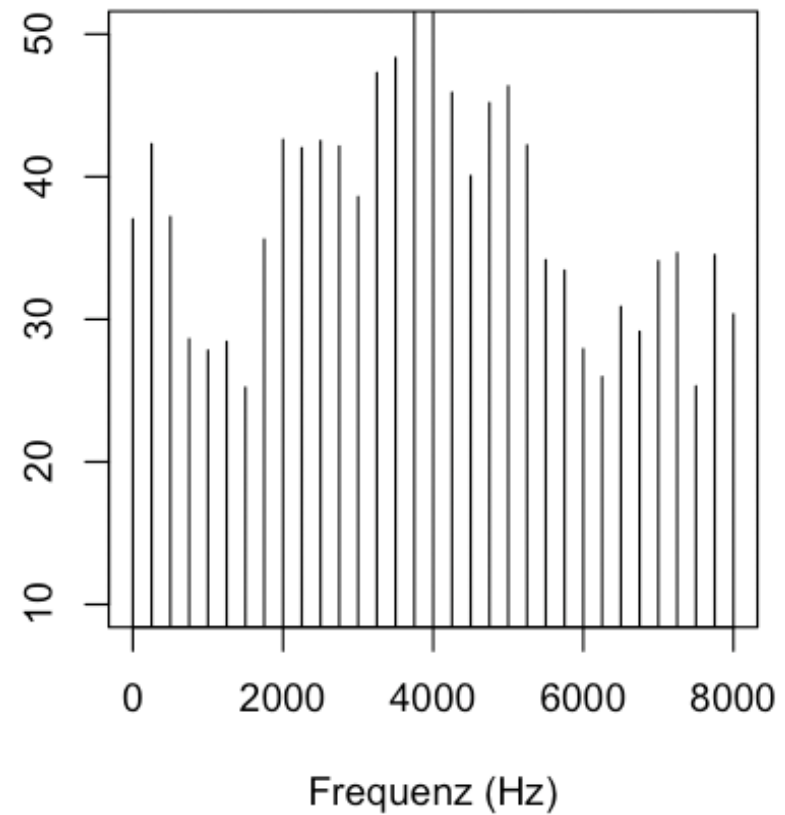
$f_s = 16000 \text{ Hz}$, $N = 512$

Frequenzabstand
= 31.25 Hz



$f_s = 16000 \text{ Hz}$, $N = 64$

Frequenzabstand =
 $16000/64 = 250 \text{ Hz}$



Zusammenfassung

- Bei einer Fourier-Analyse werden N aufeinanderfolgende digitale Werte eines Zeitsignals in N spektrale Werte umgewandelt.
- Dauer in ms eines N -Punkt-Fensters: $N/f_{\text{s}_{\text{kHz}}}$, wo $f_{\text{s}_{\text{kHz}}}$ die Abtastrate in kHz ist. zB 256 Punkte bei 10 kHz = 25.6 ms.
- Von den N -spektralen Werten behalten wir diejenigen bis zur und inkl. der Faltung-Frequenz.
- Das sind $(N/2) + 1$ spektrale Komponente **zwischen 0 und $f_{\text{s}}/2$ Hz** mit einem Frequenzabstand von f_{s}/N

Spektra in dbemu und EMU-R

Das Ziel: Spektra von [ç, x] ('ich' vs. 'ach')
miteinander vergleichen der timetable Datenbank

Wie müssten sich die Frikative voneinander im
wesentlichen spektral unterscheiden?

Warum dbnorm?

Die Amplituden-Werte von Spektren sind in **Decibel**.

Decibel sind aber Logarithmen, und um den Durchschnitt von Logarithmen zu bekommen, müssen sie zuerst in Anti-**Logarithmen** (eine Potenz hoch 10) umgerechnet werden.

- Diese Umrechnung in Anti-Logarithmen konvertiert die logarithmische Decibel oder Bel Skala in eine **lineare Kraft Skala**
- Die Berechnung (Durchschnitt usw.) erfolgt dann in der Kraft-Skala.
- Dann werden diese Berechnungen wieder in dB konvertiert.

Logarithmische dB-Werte

60 dB

70 dB



lineare Kraft-Werte

10^6

10^7



Berechnungen
durchführen

$$(10^6 + 10^7) / 2$$

$$= 5500000$$



Logarithmische dB-Werte

$$10 * \log(5500000, \text{base}=10)$$

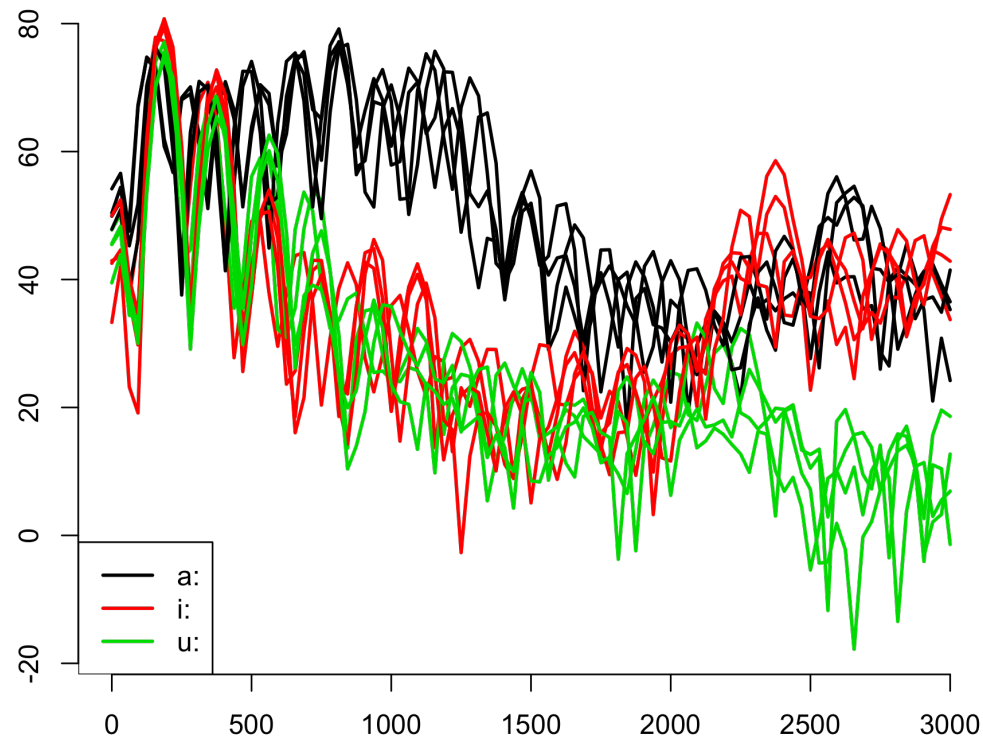
$$[1] 67.40363$$

(Der Mittelwert von 60 dB und 70 dB = 67.4 dB)

Parametrisierung von Spektra

Jonathan Harrington

Das Ziel 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

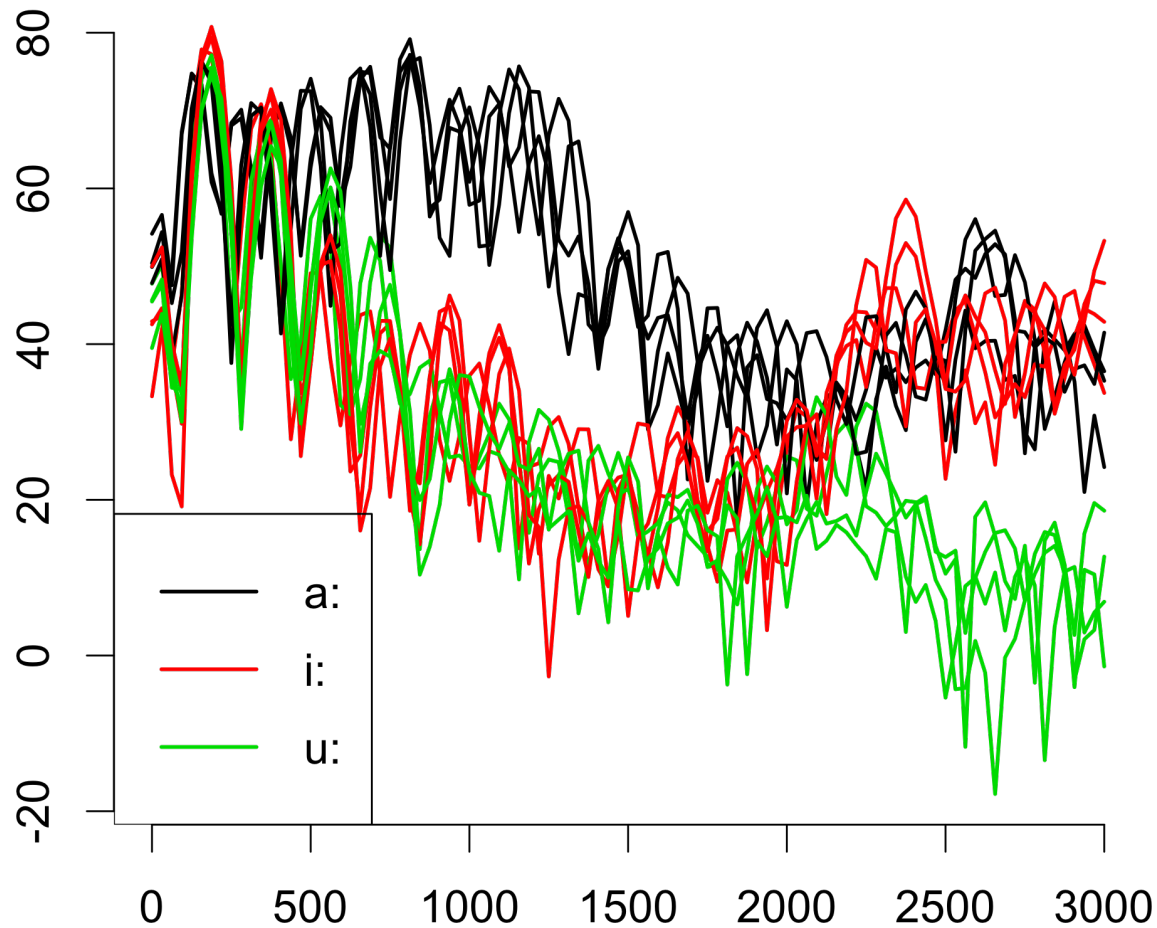
```
sp = dcut(v.dft, .5, prop=T)
```

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?

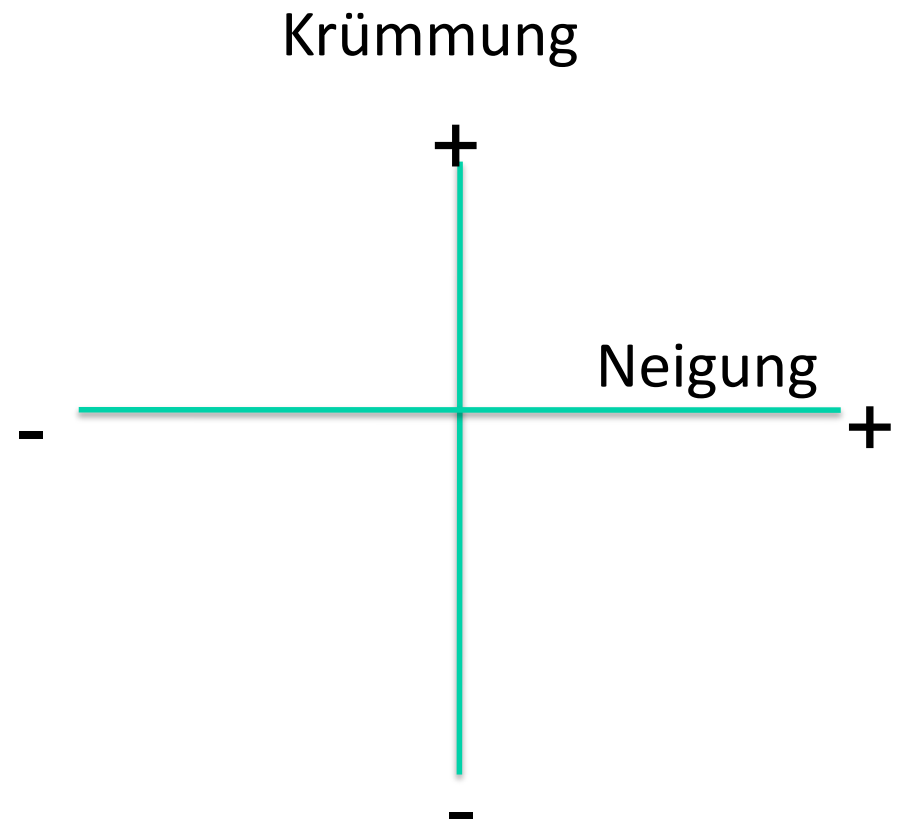
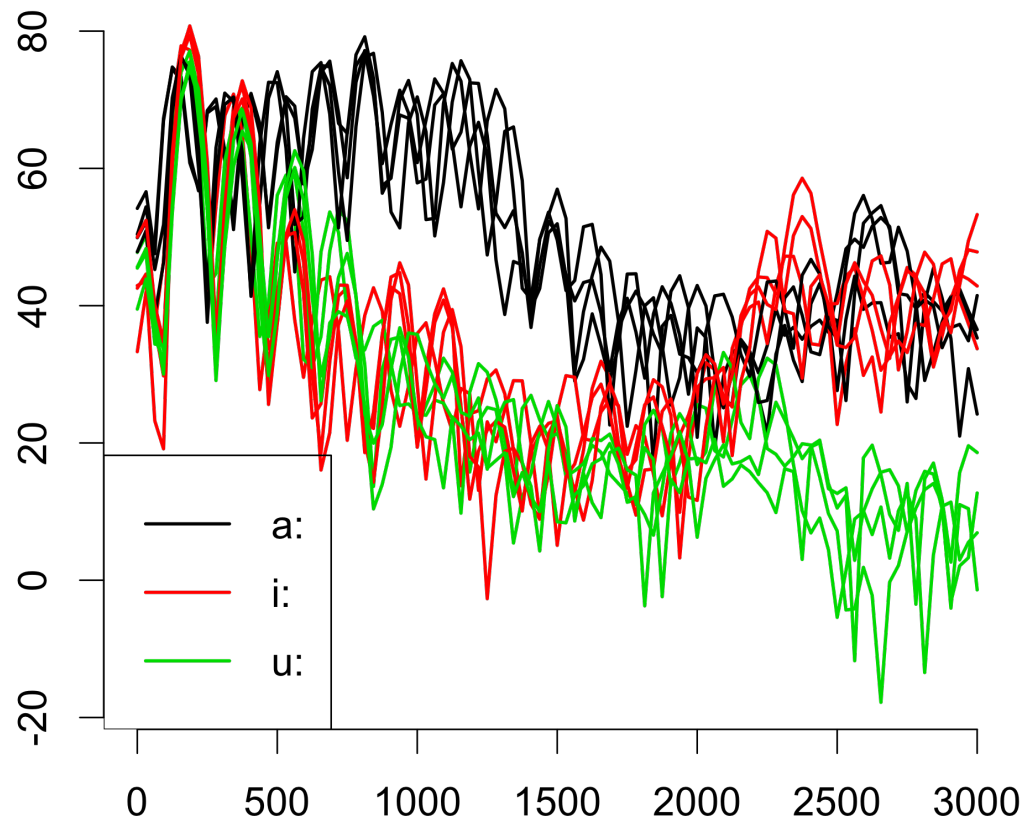
```
b =
```

Ellipse-Abbildung in diesem Raum

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

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

Die spektrale Neigung und Krümmung



Die spektrale Neigung und Krümmung

Neigung und Krümmung des Spektrums 0-3000 Hz
vom ersten Segment?

```
p =          dct( sp[1,0:3000], 2 )
```

Neigung und Krümmung des Spektrums 0-3000 Hz aller
Segmente? `fapply(trackdatei, fun, Argumente)` verwenden

```
p =          fapply(sp[,0:3000], dct, 2)
```

Ellipse-Abbildung im Raum Neigung x Krümmung

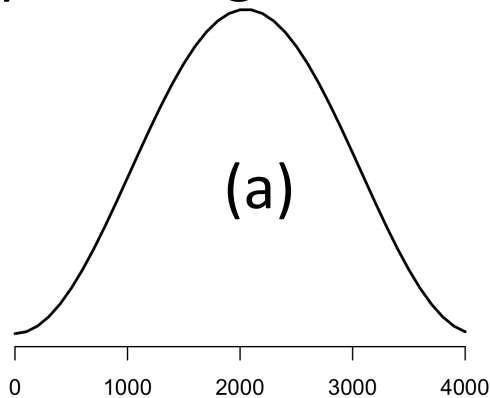
```
eplot(p[,2:3], v.l, dopoints=T)
```

m_1 : erstes spektrales Moment (spektrales Gewichtsschwerpunkt)

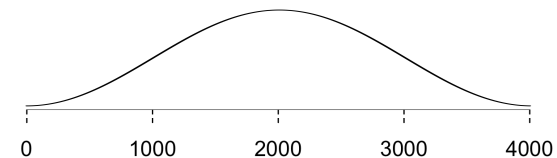
Je mehr sich die Energie in höheren Frequenzen konzentriert, umso höher m_1 (in Hz gemessen).

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

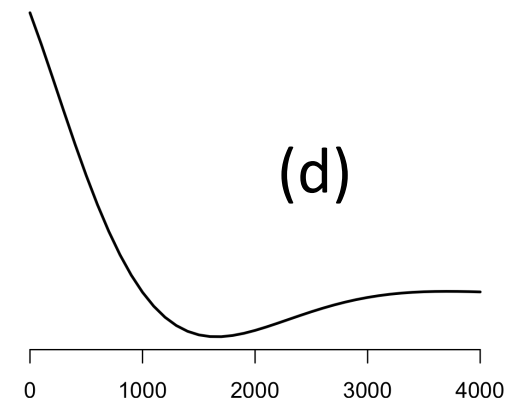
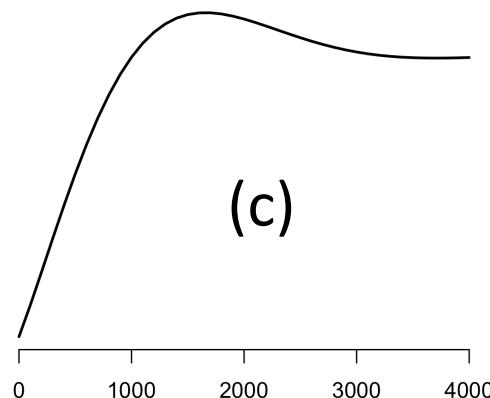
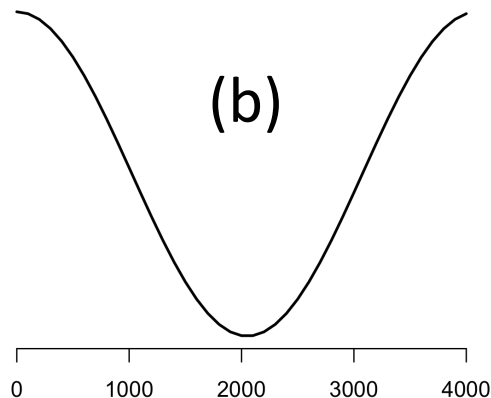
m_1 wird nicht von der dB-Skalierung beeinflusst



(b): m_1 wie für (a)

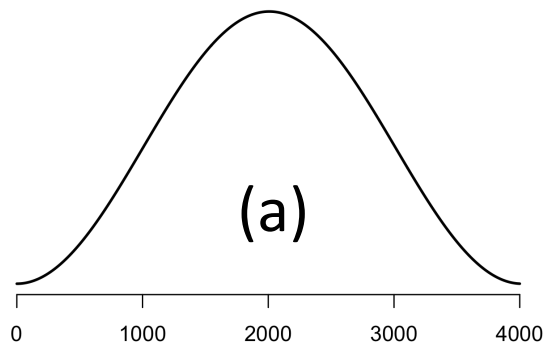


m_1 gleich/höher/tiefer im Vgl. zu (a)?



m_2 : zweites spektrales Moment (spektrale Varianz)

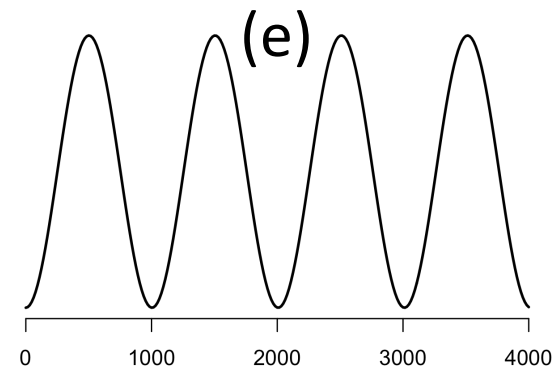
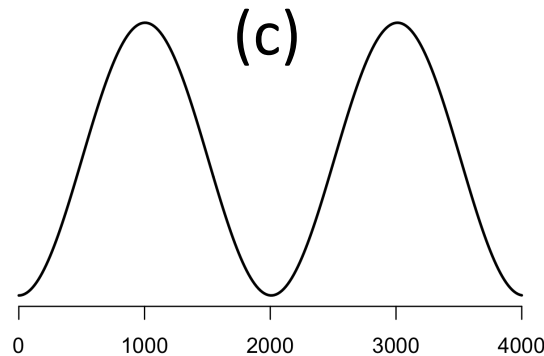
Je verteilter die Energie im Spektrum, umso höher m_2 (in Hz^2)



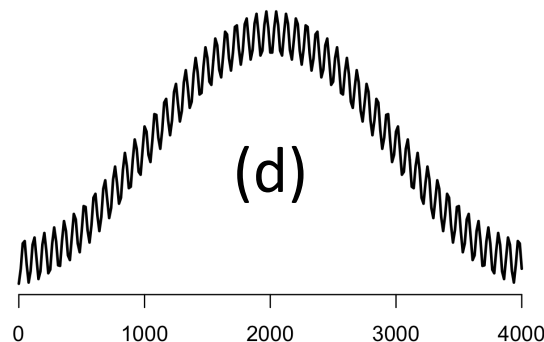
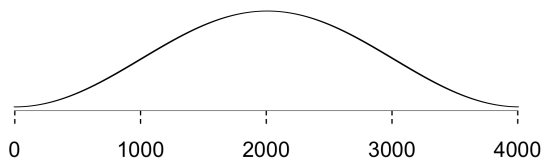
m_1 von c-f im Vgl. zu (a)?

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

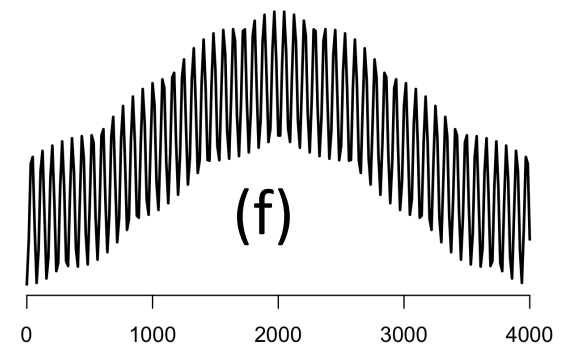
m_2 wird nicht von
der dB-Skalierung
beeinflusst



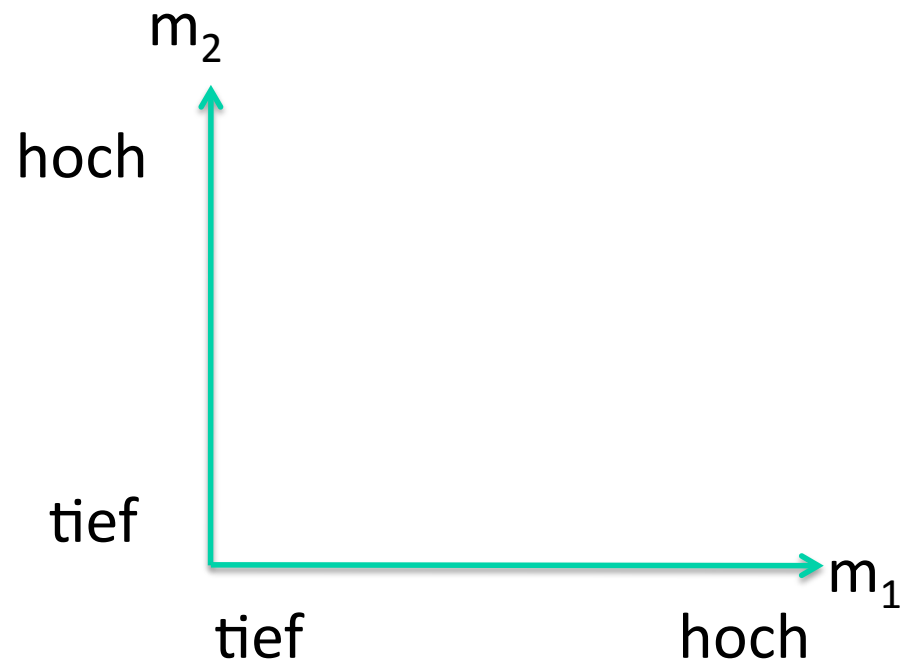
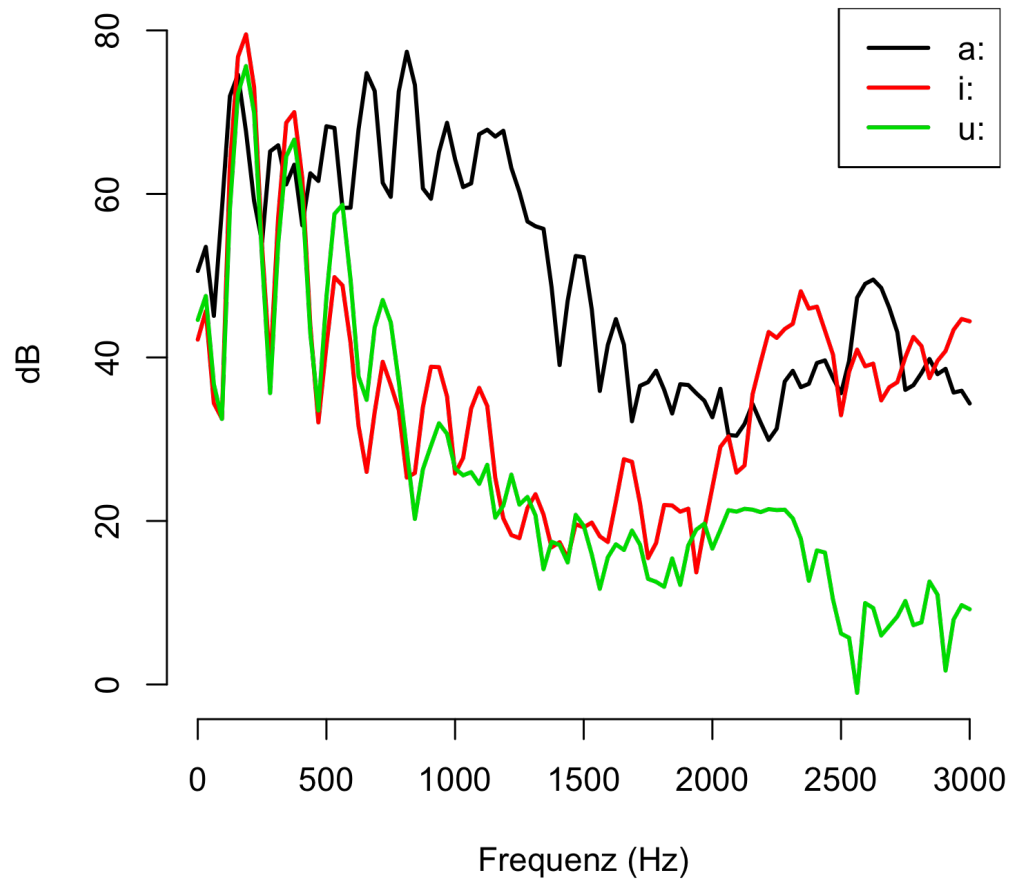
(b): m_2 wie für (a)



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)
```