

```
# Wenn nicht bereits gemacht...
library(emu)
library(lattice)
# Um Trackdateien aus einer Webseite einzulesen
library(RCurl)
source(file.path(pfadu, "readtrackfromurl.R"))
# Für Ellipsen
library(latticeExtra)
source(file.path(pfadu, "ellipsefun.R"))
# ggf. für Hann/Hamming Fenster
source(file.path(pfadu, "hwin.R"))

##### DATEN EINLESEN
# Segment list (derived from the dfgvowels database)
g.s = read.emusegs(file.path(paste(pfadu, "dfg", sep="/"), "g.s.txt"))
gc.l = scan(file.path(paste(pfadu, "dfg", sep="/"), "gc.l.txt"), what="")
g.fm = readtrackfromurl("g.fm.txt",file.path(paste(pfadu, "dfg", sep="/")) , pfad)
# dauert 1-2 Minuten
g.dft = readtrackfromurl("g.dft.txt",file.path(paste(pfadu, "dfg", sep="/")) , pfad)

#####

# Berechnung von Energie-Summen in einer Filterbank von n Filtern
# Die Frequenzen: int = seq(0, to = max(trackfreq(specdaten)), length = N + 2)
# links = int[1:N]; rechts = int[3:(N+2)]

#####
# Hauptkomponentenanalyse: prcomp(d)
# p = prcomp(d)
# plot(p); summary(p); cor(p$x)
# das Ergebnis für PCA-1 : sum(d[1,] * p$rotation[,1])
```