

## Statistik 6 - Demoskript

(c) Juergen Dengler, 05.11.2018

```
#Mit Beispieldaten aus Wildi (2013)

library(dave)
sveg
str(sveg)
summary(sveg)
names(sveg)

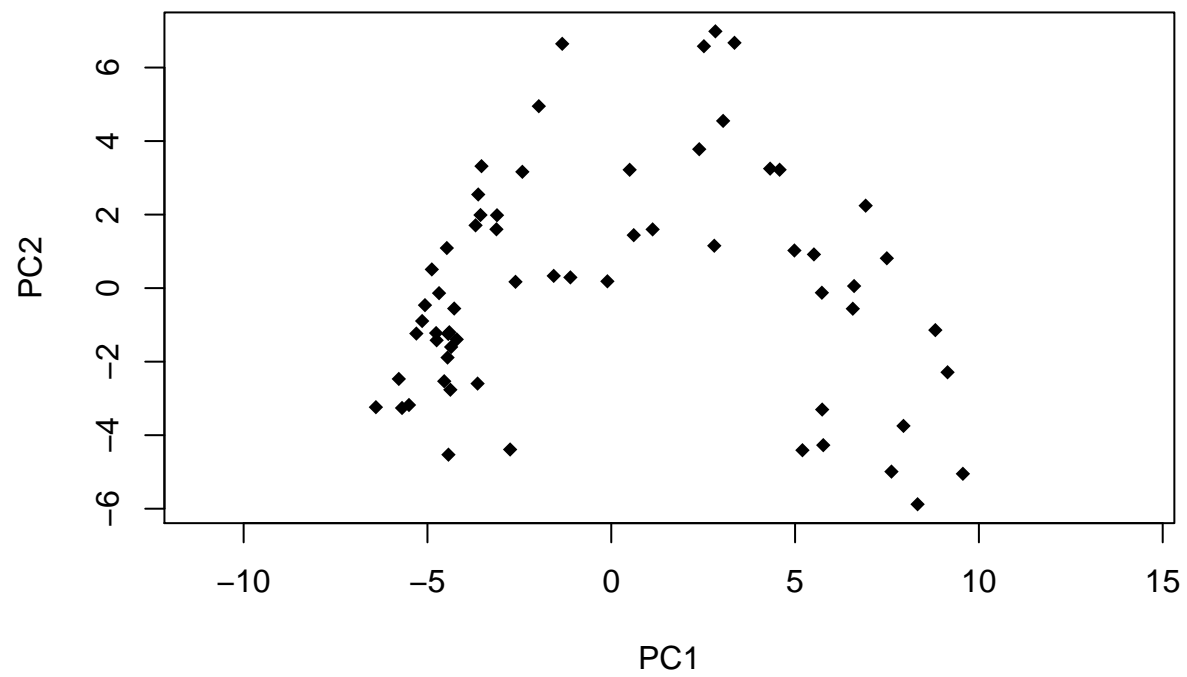
#PCA: Deckungen Wurzeltransformiert, cor=T erzwingt Nutzung der Korrelationsmatrix
o.pca<-pca(sveg^0.25,cor=T)

#Koordinaten im Ordinationsraum
o.pca$scores

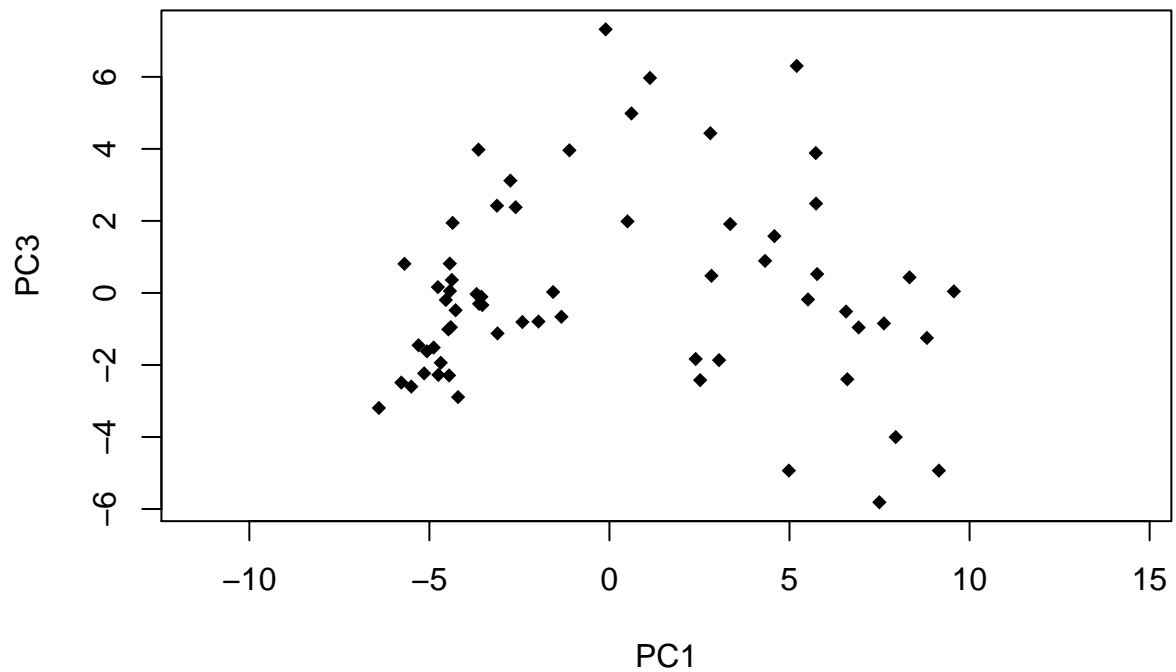
#Korrelationen der Variablen mit den Ordinationsachsen
o.pca$loadings

#Erklärte Varianz der Achsen
E<-o.pca$sdev^2/o.pca$totdev*100
E
E[1:5]

#PCA-Plot der Lage der Beobachtungen im Ordinationsraum
plot(o.pca$scores[,1],o.pca$scores[,2],type="n", asp=1, xlab="PC1", ylab="PC2")
points(o.pca$scores[,1],o.pca$scores[,2],pch=18)
```

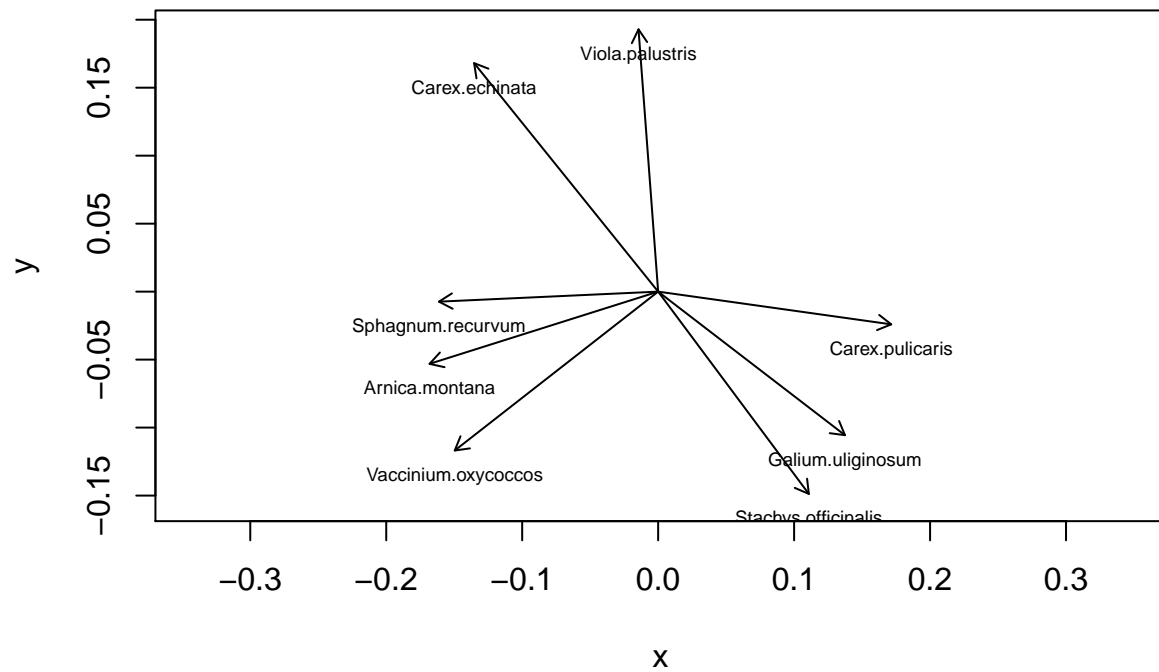


```
plot(o.pca$scores[,1],o.pca$scores[,3],type="n", asp=1, xlab="PC1", ylab="PC3")
points(o.pca$scores[,1],o.pca$scores[,3],pch=18)
```



```
#Subjektive Auswahl von Arten zur Darstellung
sel.sp <- c(3,11,23,39,46,72,77,96)
snames <- names(sveg[,sel.sp])
snames

#PCA-Plot der Korrelationen der Variablen (hier Arten) mit den Achsen (h)
x <- o.pca$loadings[,1]
y<-o.pca$loadings[,2]
plot(x,y,type="n",asp=1)
arrows(0,0,x[sel.sp],y[sel.sp],length=0.08)
text(x[sel.sp],y[sel.sp],snames,pos=1,cex=0.6)
```

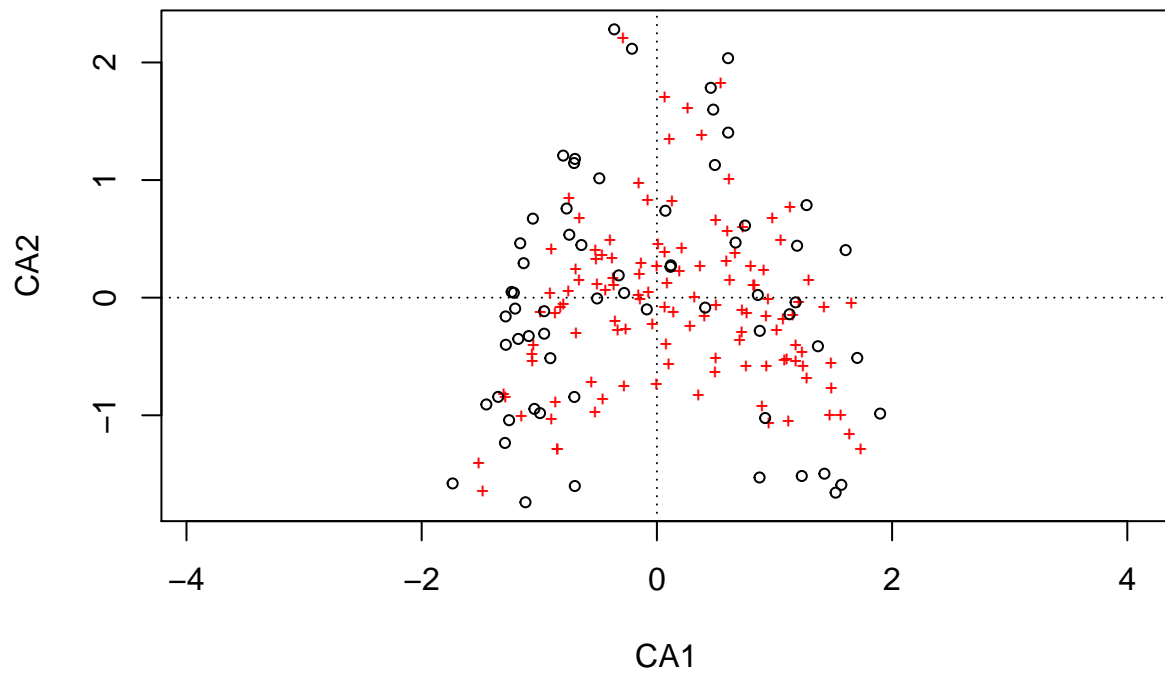


```
#Erklärte Varianz der Achsen in Prozent (sdev ist die Wurzel daraus)
E<-o.pca$sdev^2/o.pca$totdev*100
E
E[1:5]

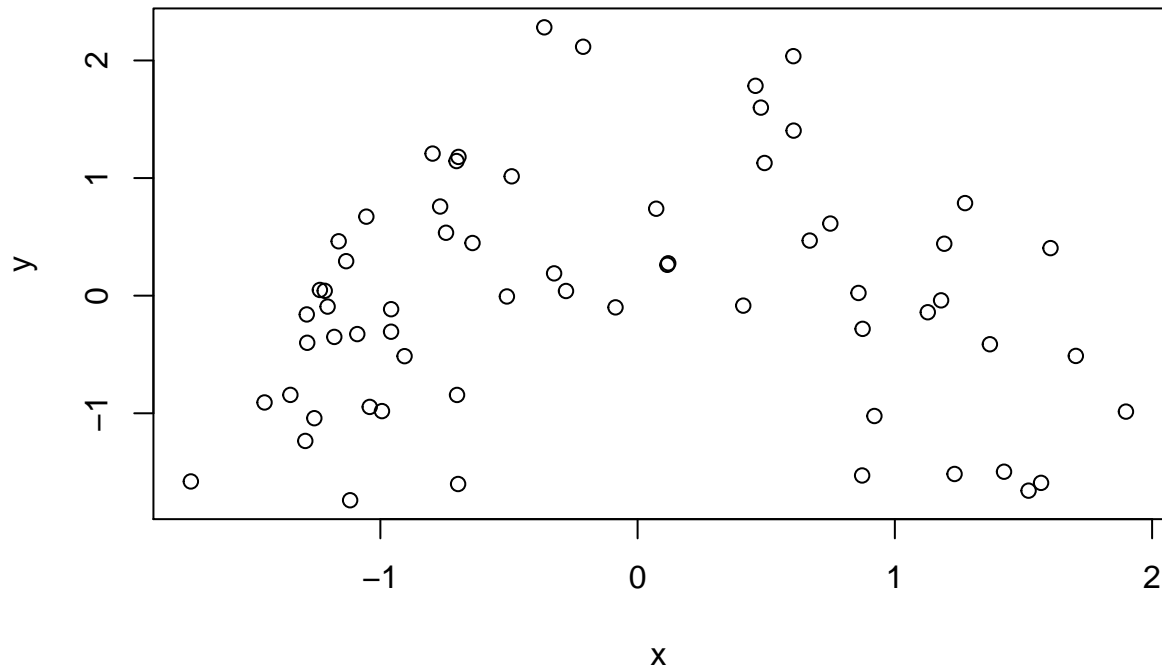
#CA

library(vegan)

o.ca<-cca(sveg^0.5)
#Arten (o) und Communities (+) plotten
plot(o.ca)
```



```
#Nur Arten plotten  
x<-o.ca$CA$u[,1];y<-o.ca$CA$u[,2]  
plot(x,y)
```



```
#Anteilige Varianz, die durch die ersten beiden Achsen erklärt wird
o.ca$CA$eig[1:2]/sum(o.ca$CA$eig)

#DCA

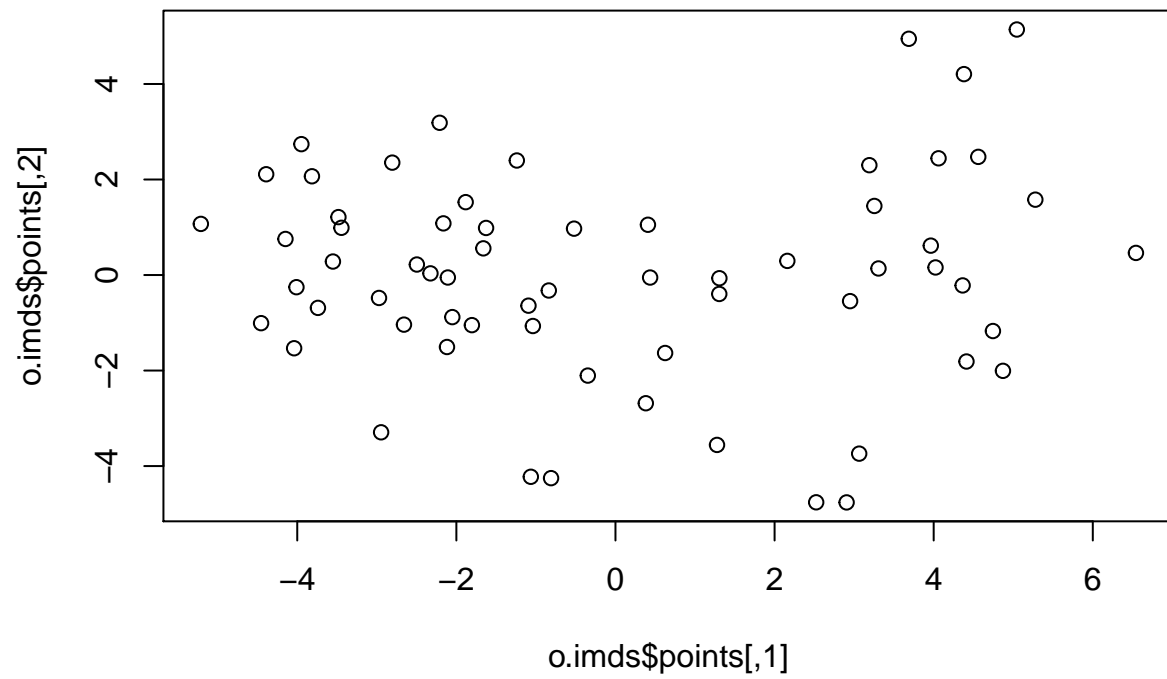
o.dca <- decorana(sveg,mk=10)
#plot(o.dca$rproj, asp=1)

o.dca2 <- decorana(sveg,mk=100)
#plot(o.dca2$rproj, asp=1)

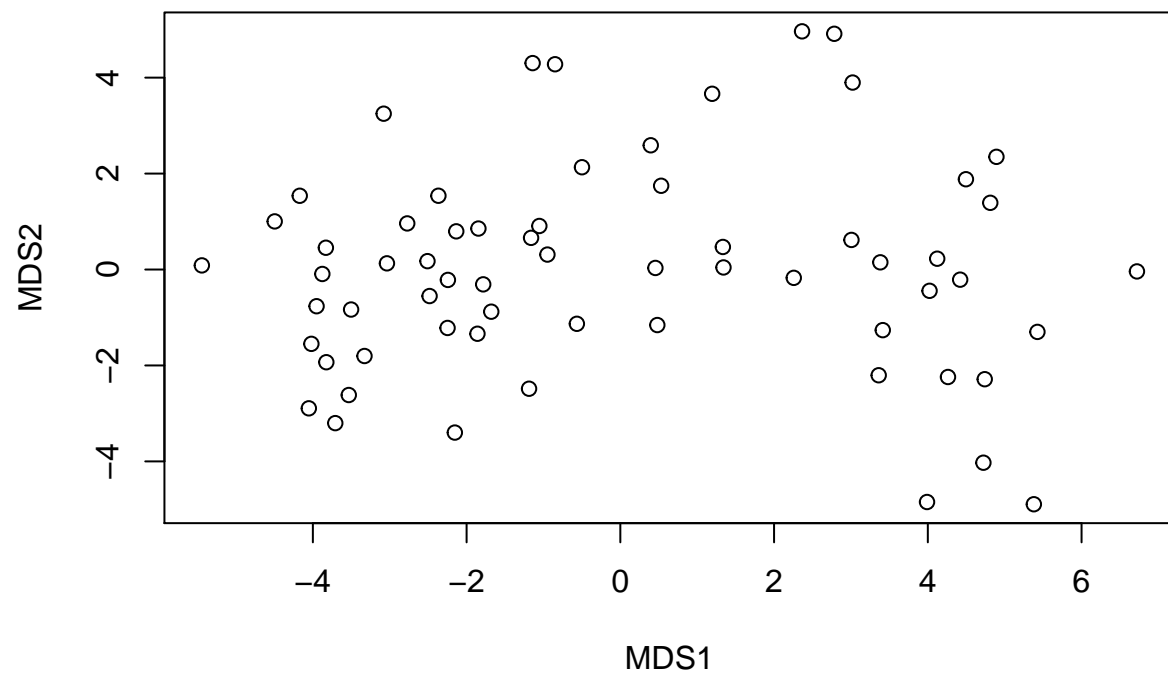
#NMDS

#Distanzmatrix als Start erzeugen
c
mde <-vegdist(sveg,method="euclidean")

#Zwei verschiedene NMDS-Methoden
library(MASS)
set.seed(1) #macht man, wenn man bei einer Wiederholung exakt die gleichen Ergebnisse will
o.imds<-isoMDS(mde,k=2)
set.seed(1)
o.mmds<-metaMDS(mde,k=2)
plot(o.imds$points)
```

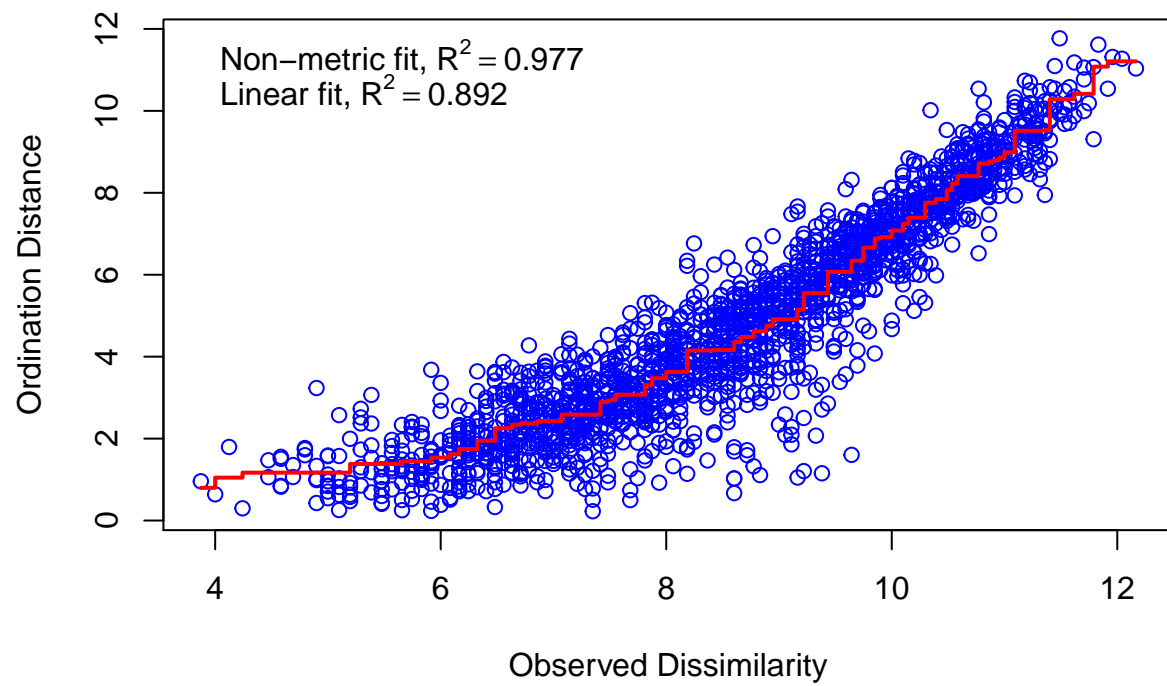


```
plot(o.mmds$points)
```



*#Stress = S = Abweichung der zweidimensionalen NMDS-Lösung von der originalen Distanzmatrix*  
`stressplot(o.imds,mde)`





```
stressplot(o.mmds,mde)
```

