**Inhalt**

[**R-Skripte für das Zusatzangebot** 2](#_Toc145398403)

[**PCA\_Beispielanalyse\_LSHBsp.R** 2](#_Toc145398404)

[**CA\_Beispielanalyse\_tab28.R** 4](#_Toc145398405)

[**NMDS\_Beispielanalyse\_tab28.R** 6](#_Toc145398406)

[**CCA\_Beispielanalyse\_varechem.R** 11](#_Toc145398407)

Zur Verwendung der R-Skripte:

Kopieren Sie sich das entsprechende R-Skript in die Zwischenablage, fügen es anschließend in R-Studio in ein leeres R-Skript ein und speichern es mit dem oben angegebenen Dateinamen ab. WICHTIG: der Dateipfad, der i.d.R. in den ersten Zeilen des R-Skriptes angegeben ist, muss auf Ihren angepasst sein, beachten Sie hierbei auch die richtige Zeichensetzung.

# **R-Skripte für das Zusatzangebot**

## **PCA\_Beispielanalyse\_LSHBsp.R**

# Ordination mit Beispieldaten - PCA

setwd("C://studium//statistik//Ordination") # Arbeitsverzeichnis setzen

library(vegan) # Paket vegan laden

# Einlesen der Beispieltabelle mit Pflanzenmerkmalen (Traits)

# csv-Tabellen (= comma separated values) können mit

# read.csv2() eingelesen werden

pft.bsp <- read.csv2("LHSBsp.csv", row.names = 1)

summary(pft.bsp) # Verteilungen der Variablen

# Vereinheitlichen der Wertebereiche der variablen (0-1)

pft.stand <- decostand(pft.bsp, method = "range")

summary(pft.stand) # Verteilungen der Variablen

# Berechnen der PCA mit der Funktion rda() aus vegan

pft.pca <- rda(pft.stand)

summary(pft.pca) # Ausgabe der Zusammenfassung

# Achse 1 erklärt 48,5%, Achse 2 erklärt 40 % der Varianz, in Summe 88%

# Ausgabe des Biplots

biplot(pft.pca)

######

# Klassifikation mit Ward's Methode

# Berechnen der Distanzmatrix (euklidische Distanzen)

pft.d.eu <- vegdist(pft.stand, method = "euclidean")

# Berechnen der hierarchischen KLassifikation

pft.ward <- hclust(pft.d.eu, method = "ward.D2")

# Ausgabe des Dendrogramms

plot(pft.ward, sub = "Ward's method", xlab = "")

# Einzeichnen von rahmen für 3 Cluster

rect.hclust(pft.ward, k= 3, border = "navy")

# Ableiten von Clusternummern für die Objekte

# (Reihenfolge entspricht nicht dem Dendrogramm)

pft.ward.nr <- cutree(pft.ward, k= 3)

#####

# Biplot mit Symbolfarben für die Cluster

# Ausgabe der Koordinaten der Arten in der PCA

# Da in diesem Fall die Arten die Objekte sind, wird display auf "sites" gestellt

koord.pft.pca.arten <- scores(pft.pca, display = "sites")

# Ausgabe der Koordinaten der Traits - diese sind hier die Eigenschaften

koord.pft.pca.traits <- scores(pft.pca, display = "species")

# Ausgabe eines Biplots mit kleinen Symbolen,

# Y-Achse wird in der Ausdehnung definiert,

# damit die Beschriftung für den Pfeil nach oben Platz hat

biplot(pft.pca, type = "p", ylim = c(-0.5,0.9),xlim = c(-1,1))

# einfügen der Symbole für die Arten mit Farbe nach CLuster

points(koord.pft.pca.arten[pft.ward.nr == 1,], pch = 21, bg = "red", cex = 2)

points(koord.pft.pca.arten[pft.ward.nr == 2,], pch = 21, bg = "royalblue", cex = 2)

points(koord.pft.pca.arten[pft.ward.nr == 3,], pch = 21, bg = "chartreuse", cex = 2)

# Beschriftung der Objekte - Ausrichtung links des Symbols (adj = 1)

text(koord.pft.pca.arten, adj = 1, row.names(pft.stand), cex = 0.8, col = "navy")

# Beschriftung der Pfeile (cex: Skalierungsfaktor)

text(koord.pft.pca.traits\*1.1, names(pft.stand), col = "red", cex = 0.8)

## **CA\_Beispielanalyse\_tab28.R**

# Beispielskript zur Korrespondenzanalyse M. Rudner 05.03.2023

setwd("C://studium//statistik//Ordination") # Arbeitsverzeichnis setzen

#Einlesen

library(readxl) # Paket readxl laden

tab28 <- read\_excel("tab28.xlsx", sheet = "veg\_t")

tab28 <- data.frame(tab28) # Umwandeln in Tabelle vom Typ data.frame

row.names(tab28) <- tab28[,1] # 1. Spalte als Zeilennamen setzen

tab28 <- tab28[,-1] # anschließend 1. Spalte löschen

summary(tab28) # Übersicht zu den Variablen

tab28[is.na(tab28)] <- 0 # Fehlende Werte durch 0 ersetzen

summary(tab28) # Übersicht zu den Variablen

library(vegan) # Paket vegan laden

# Prüfen, wieviele Arten nur einmal im Datensatz sind

sum(colSums(decostand(tab28, method = "pa")) < 2)

colSums(decostand(tab28, method = "pa")) < 2 # Anzeigen, welche Arten das sind

# Datensatz um diese Art(en) kürzen

tab28.2 <- tab28[,colSums(decostand(tab28, method = "pa")) > 1]

summary(tab28.2) # Ergebnis prüfen

# Prüfen, ob Aufnahmen sehr artenarm geworden sind

boxplot(rowSums(decostand(tab28.2, method = "pa")))

tab28.ca <- cca(sqrt(tab28.2))# CA mit wurzeltransformierten Daten

tab28.ca # kurze Übersicht

summary(tab28.ca) # ausführliche Ergebnisübersicht

plot(tab28.ca) # Biplot zur CA

plot(tab28.ca, display = "sites") # Plot nur mit Aufnahmen

# zusätzlich Arten eintragen ohne Überlappung

orditorp(tab28.ca, display = "species", col = "tomato")

# Plot mit Deckung der Buche im Symbol

fig1 <- ordiplot(tab28.ca, type = "p") # Biplot mit Symbolen

koord.ca.sites <- scores(tab28.ca, display = "sites") #Koordinaten der Aufnahmen

# Grüne Punktsymbole mit Größe nach Buchendeckung

points(koord.ca.sites, pch = 21, bg = "green",

 cex = sqrt(tab28.2$Fagus.sylvatica)/2)

# Einfügen einer Legende dazu

legend(-1.5,1.5, legend = c(25,50,80), pch = rep(21,3), pt.bg = rep("green",3),

 pt.cex = c(2.5,3.5,4.5), title = "Fagus sylvativca")

# Nachträgliches Einfügen eines Titels für die Abbildung

title(main = "Korrespondenzanalyse Wald 10er Datensatz")

# Plot mit Deckung des Wald-Bingelkrauts

fig1 <- ordiplot(tab28.ca, type = "p") # Biplot mit Symbolen

# Türkise Punktsymbole mit Größe nach Deckung des Bingelkrauts

points(koord.ca.sites, display = "sites", pch = 21, bg = "cyan",

 cex = sqrt(tab28.2$Mercurialis.perennis)\*1.5)

# Einfügen einer Legende dazu

legend(-1.5,1.5, legend = c(3,10), pch = rep(21,2), pt.bg = rep("cyan",2),

 pt.cex = c(2.6,4.7), title = "Mercurialis perennis")

# Nachträgliches Einfügen eines Titels für die Abbildung

title(main = "Korrespondenzanalyse Wald 10er Datensatz")

## **NMDS\_Beispielanalyse\_tab28.R**

# Beispielskript Ordination - NMDS M.Rudner 27.03.2023

setwd("C://studium//statistik//Ordination") #Arbeitsverzeichnis setzen

# Einlesen

library(readxl) # Paket readxl laden

# Vegetationstabelle einlesen

tab28 <- read\_excel("tab28.xlsx", sheet = "veg\_t")

tab28 <- data.frame(tab28) # umformatieren in data.frame

row.names(tab28) <- tab28[,1] # Zeilennamen zuweisen

tab28 <- tab28[,-1] # 1. Spalte löschen

summary(tab28) # Verteilung anzeigen

tab28[is.na(tab28)] <- 0 # fehlende Werte durch 0 ersetzen

summary(tab28) # prüfen

library(vegan) # Paket vegan laden

# wie viele Arten kommen nur 1x vor

sum(colSums(decostand(tab28, method = "pa")) < 2)

# welche Arten kommen nur 1 x vor?

colSums(decostand(tab28, method = "pa")) < 2

tab28.sqrt <- sqrt(tab28) # Vegetationstabelle wurzeltransformieren

summary(tab28.sqrt) # Ergebnis prüfen

# NMDS berechnen mit 2 Dimensionen

tab28.NMDS <- metaMDS(tab28.sqrt, k = 2)

plot(tab28.NMDS) # Ordinationsdiagramm anzeigen - nur Symbole

plot(tab28.NMDS, type = "t") # Ord.diagramm mit Text: unübersichtlich

# prüfen, welche Anzahl an Dimensionen gut geeignet ist

w28.nmds1 <- metaMDS(tab28.sqrt, k = 1)

w28.nmds2 <- metaMDS(tab28.sqrt, k = 2)

w28.nmds3 <- metaMDS(tab28.sqrt, k = 3)

w28.nmds4 <- metaMDS(tab28.sqrt, k = 4)

w28.nmds5 <- metaMDS(tab28.sqrt, k = 5)

dimensionen <- 1:5

stress.dim <- c(w28.nmds1$stress, w28.nmds2$stress, w28.nmds3$stress,

 w28.nmds4$stress, w28.nmds5$stress)

plot(stress.dim ~ dimensionen, type = "b")

# Ordinationsdiagramm nach und nach aufbauen

plot(tab28.NMDS, type = "n") # leeres Ordinationsdiagramm

# Namen der Aufnahmen ohne Überlappung

orditorp(tab28.NMDS, display = "sites", col = "navy")

cnam <- make.cepnames(names(tab28)) # abgekürzte Artnamen erzeugen

# kurze Artnamen ohne Überlappung

orditorp(tab28.NMDS, display = "species", labels = cnam,

 col = "tomato")

# Überlagerung mit Deckungsangaben von Waldbaumarten

# Koordinaten der Aufnamen

k.w28.nmds.sit <- scores(tab28.NMDS, display = "sites")

plot(tab28.NMDS, type = "n") # leeres Ord.diagramm

# Aufnahmen einzeichnen

points(k.w28.nmds.sit, pch = 21, bg = "grey80", cex = 0.5)

# Deckung der Buche einzeichnen

points(k.w28.nmds.sit, pch = 21, bg = "chartreuse",

 cex = tab28.sqrt$Fagus.sylvatica/2)

points(k.w28.nmds.sit, pch = 21, bg = "sandybrown",

 cex = tab28.sqrt$Quercus.petraea/2) # dto. Traubeneiche

points(k.w28.nmds.sit, pch = 21, bg = "purple",

 cex = tab28.sqrt$Fraxinus.excelsior /2) # dto. Gem. Esche

# Legende einfügen

legend(1,1, legend = c("Fag sylv", "Fra exce", "Que petr", " 1: 3%", " 3: 37%"," 5: 87%"),

 pch =rep(21,6), pt.bg = c("chartreuse", "purple", "sandybrown", rep("grey80",3)),

 pt.cex = c(1,1,1,0.86,3.04,4.66), cex = 0.8)

# Umweltdaten einlesen

tab28.env <- read\_excel("tab28.xlsx", sheet = "env\_t")

summary(tab28.env) # Verteilungen anzeigen

ordisurf(tab28.NMDS, tab28.env$`Hangneigung (Grad)`, bubble = 5)

# Exposition aufteilen

tab28.env$`Exposition (Grad)`

NSexp28 <- cos((tab28.env$`Exposition (Grad)`/180) \* pi) # Nord-Süd

OWexp28 <- sin((tab28.env$`Exposition (Grad)`/180) \* pi) # Ost-West

ordisurf(tab28.NMDS, NSexp28) # Überlagerungsdiagramm

ordisurf(tab28.NMDS, OWexp28) # Überlagerungsdiagramm

# Auswahlvektoren zu Himmelsrichtungen anlegen

tab28.Nord <- logical(nrow(tab28.env))

tab28.Nord[tab28.env$`Exposition (Grad)` < 25] <- TRUE

tab28.Nord[tab28.env$`Exposition (Grad)` > 335] <- TRUE

tab28.Nordost <- logical(nrow(tab28.env))

tab28.Nordost[tab28.env$`Exposition (Grad)` > 24 &

 tab28.env$`Exposition (Grad)` < 67] <- TRUE

tab28.Ost <- logical(nrow(tab28.env))

tab28.Ost[tab28.env$`Exposition (Grad)` > 66 &

 tab28.env$`Exposition (Grad)` < 115] <- TRUE

tab28.SOst <- logical(nrow(tab28.env))

tab28.SOst[tab28.env$`Exposition (Grad)` > 114 &

 tab28.env$`Exposition (Grad)`< 155] <- TRUE

tab28.Sued <- logical(nrow(tab28.env))

tab28.Sued[tab28.env$`Exposition (Grad)` > 154 &

 tab28.env$`Exposition (Grad)` < 205] <- TRUE

tab28.SWest <- logical(nrow(tab28.env))

tab28.SWest[tab28.env$`Exposition (Grad)` > 204 &

 tab28.env$`Exposition (Grad)` < 245] <-TRUE

tab28.West <- logical(nrow(tab28.env))

tab28.West[tab28.env$`Exposition (Grad)` > 244 &

 tab28.env$`Exposition (Grad)` < 295] <- TRUE

tab28.NWest <- logical(nrow(tab28.env))

tab28.NWest[tab28.env$`Exposition (Grad)` > 294 &

 tab28.env$`Exposition (Grad)` < 336] <- TRUE

# Himelsrichtung als Farbe und Hangneigung als Punktgröße einzeichnen

farben.exp8 <- c("blue","cyan","green","yellow","orange","red",

 "orchid3", "purple") # Definition einer Farbreihe

plot(tab28.NMDS, type = "n") # leeres Ord.diagramm

points(k.w28.nmds.sit[tab28.Nord,], pch = 21, bg = farben.exp8[1],

 cex = sqrt(tab28.env$`Hangneigung (Grad)`[tab28.Nord]))

points(k.w28.nmds.sit[tab28.Nordost,], pch = 21, bg = farben.exp8[2],

 cex = sqrt(tab28.env$`Hangneigung (Grad)`[tab28.Nordost]))

points(k.w28.nmds.sit[tab28.Ost,], pch = 21, bg = farben.exp8[3],

 cex = sqrt(tab28.env$`Hangneigung (Grad)`[tab28.Ost]))

points(k.w28.nmds.sit[tab28.SOst,], pch = 21, bg = farben.exp8[4],

 cex = sqrt(tab28.env$`Hangneigung (Grad)`[tab28.SOst]))

points(k.w28.nmds.sit[tab28.Sued,], pch = 21, bg = farben.exp8[5],

 cex = sqrt(tab28.env$`Hangneigung (Grad)`[tab28.Sued]))

points(k.w28.nmds.sit[tab28.SWest,], pch = 21, bg = farben.exp8[6],

 cex = sqrt(tab28.env$`Hangneigung (Grad)`[tab28.SWest]))

points(k.w28.nmds.sit[tab28.West,], pch = 21, bg = farben.exp8[7],

 cex = sqrt(tab28.env$`Hangneigung (Grad)`[tab28.West]))

points(k.w28.nmds.sit[tab28.NWest,], pch = 21, bg = farben.exp8[8],

 cex = sqrt(tab28.env$`Hangneigung (Grad)`[tab28.NWest]))

legend(1,1.2,legend = c("Nord","Nordost","Ost","Südost","Süd","Südwest",

 "West", "Nordwest", " 2°", " 5°", "10°","20°"), pch = 21,

 pt.bg = c(farben.exp8,rep("grey80",4)), pt.cex = c(rep(1,8),1,1.4,2.2,3.1,4.5), cex = 0.8 )

title(main = "NMDS tab28: Exposition und Hangneigung")

## **CCA\_Beispielanalyse\_varechem.R**

library(vegan) # Paket vegan laden

data(varespec, varechem) # Datensätze Rentierweiden laden

summary(varechem) # Verteilungen der Umweltvariablen

summary(varespec) # Verteilungen der Artmächtigkeiten

# kanonische Korrespondenzanalyse mit 3 ausgewählten Umweltvariablen

mod <- cca(varespec ~ Al + P + K, varechem)

anova(mod) # Prüfen, ob Gesamtmodell signifikant ist

anova(mod, by="term") # Prüfen, ob Variablen signifikant sind

# Vergleich der Rest"varianzen" des Modells und eines vollen Modells

anova(mod, cca(varespec ~ ., varechem))

# Die F-Statistik ist nicht signifikant: Beide Modelle sind gleich gut.

# Das Hinzufügen von weiteren Umweltvariablen würde also nichts bringen

summary(mod) # Zusammenfassung des Modells

plot(mod) # Triplot zeichnen - schwer lesbar

# Komponenten einzeln in Triplot einzeichnen (ohne Überlagerung der Texte)

ordiplot(mod, type = "n") # Zeichnen eines leeren Plots

orditorp(mod, display = "sites", col = "navy") # Zeichnen der Aufnahmen

orditorp(mod, display = "species", col = "darkred") # Zeichnen der Arten

# Zeichnen der Pfeile für die Umweltvariablen

 Pfeile <- data.frame(mod$CCA$biplot) # Koordinaten der Pfeilspitzen

 # Zeichnen der drei Pfeile mit for-Schleife,

 # Pfeile werden auf doppelte Länge skaliert

 for (i in 1:3){

 arrows(0,0,Pfeile[i,1]\*2,Pfeile[i,2]\*2, length = 0.1, col = "blue")

 }

 # Beschriftung der Pfeile

 text(Pfeile[,1:2]\*2.1, row.names(Pfeile), col = "blue", cex = 0.9)