Analysis_birdsdata

Cheyenne, Kathina & Lina

4/7/2021

This document have been created to include all relevant parts in the analysis of our animal diversity project.

##Data preparation and data observation

The first code's chunk will be used to load all libraries needed to proceed.

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.0.5

## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.4     v purrr   0.3.4
## v tibble  3.1.2     v dplyr   1.0.7
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1

## Warning: package 'ggplot2' was built under R version 4.0.5

## Warning: package 'tibble' was built under R version 4.0.5

## Warning: package 'tidyr' was built under R version 4.0.5

## Warning: package 'forcats' was built under R version 4.0.5

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(vegan)
```

```
## Warning: package 'vegan' was built under R version 4.0.4

## Loading required package: permute

## Loading required package: lattice

## This is vegan 2.5-7
```

```
library(nlme)
```

```
## Warning: package 'nlme' was built under R version 4.0.5
```

```
##
## Attaching package: 'nlme'
```

```
## The following object is masked from 'package:dplyr':
##
##     collapse
```

```
library(MASS)
```

```
## Warning: package 'MASS' was built under R version 4.0.5
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
library(MuMIn)
```

```
## Warning: package 'MuMIn' was built under R version 4.0.5
```

In this second chunk, the data to be used will be loaded into a data frame and transferred into adequate data types.

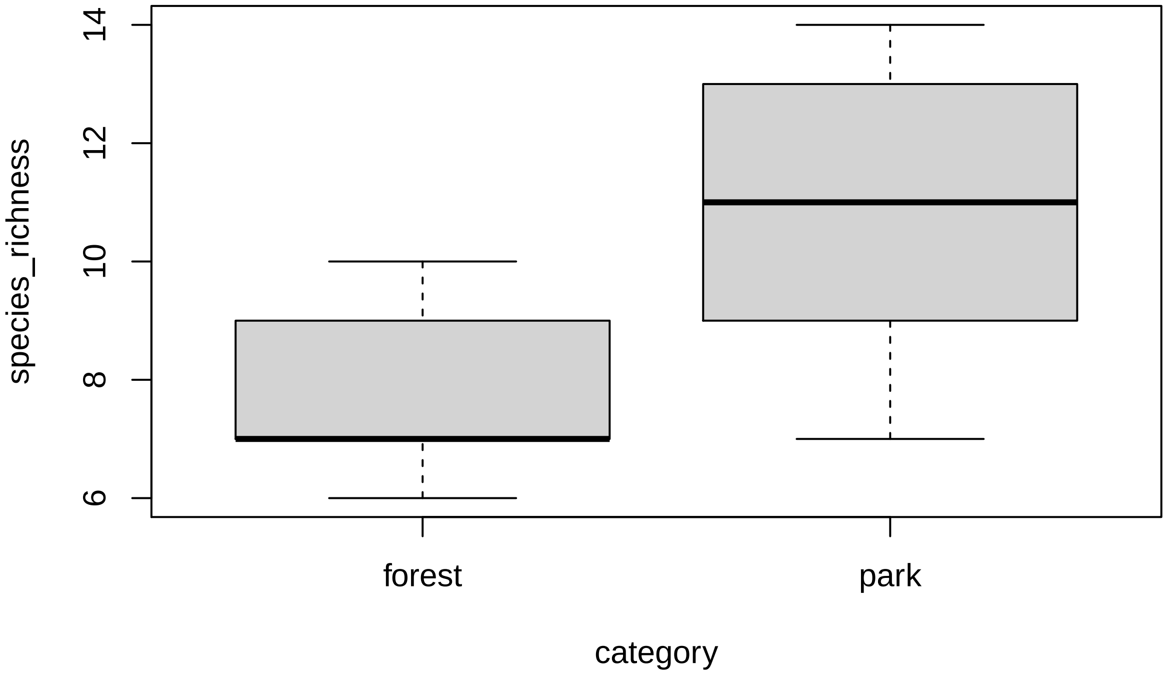
```
dat <- read.csv("data/birds_dataset.csv", sep=";")
dat$category <- as.factor(dat$category)
dat$site <- as.factor(dat$site)
```

Following, species richness, species abundance and rarefied richness of species will be calculated.

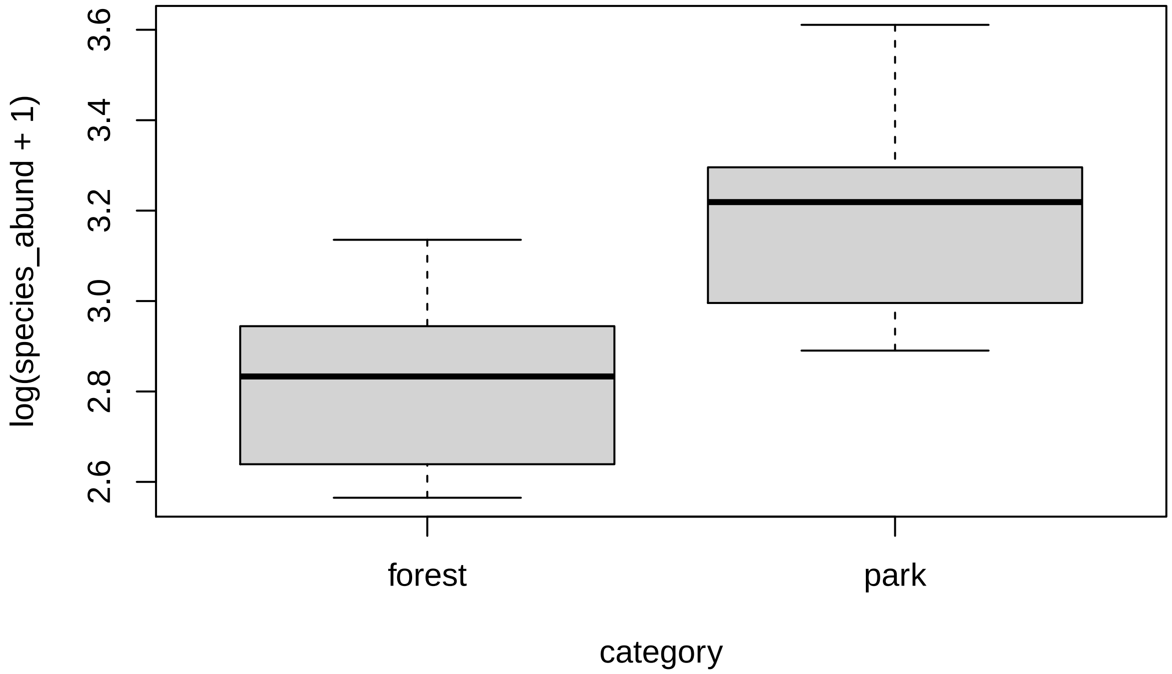
```
dat$species_richness <- specnumber(dat[,4:31]) #species richness
dat$species_abund <- rowSums(dat[,4:31]) #abundances
dat$rarefied_richness <- rarefy(dat[,4:31],min(dat$species_abund)) #rarefied richness based on the subs
```

After, data is observed comparing the plots from parks and forests.

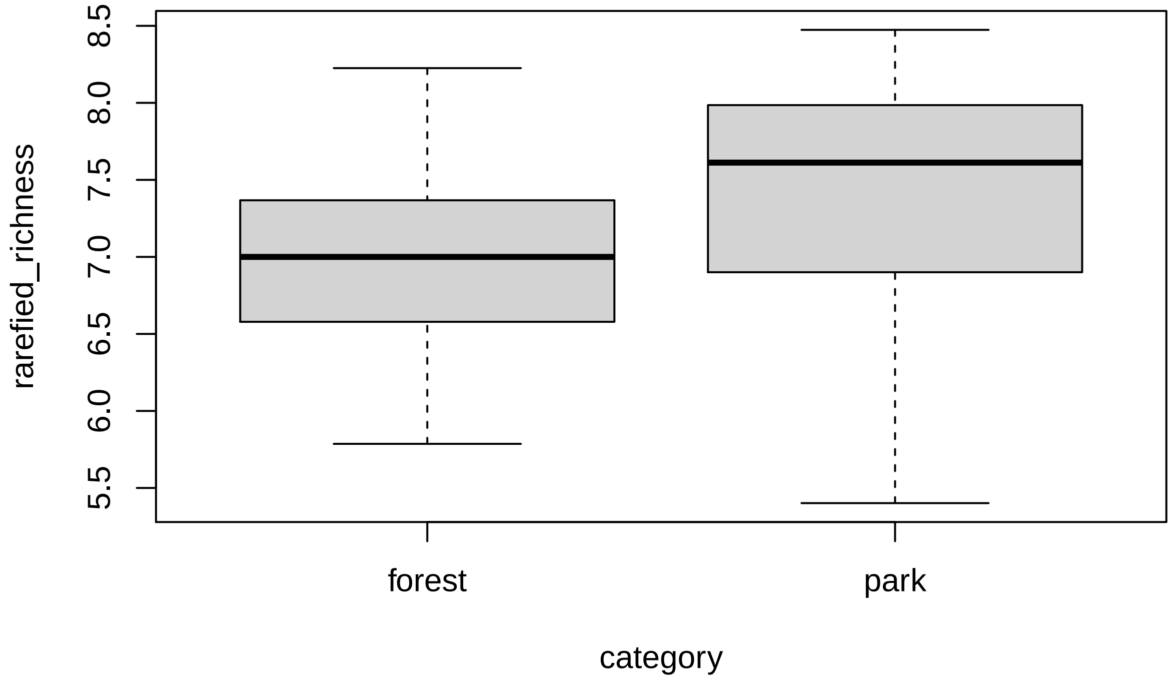
```
boxplot(species_richness~category, data=dat)
```



```
boxplot(log(species_abund+1)~category, data=dat)
```



```
boxplot(rarefied_richness~category, data=dat)
```



##Now let's begin with the data analysis

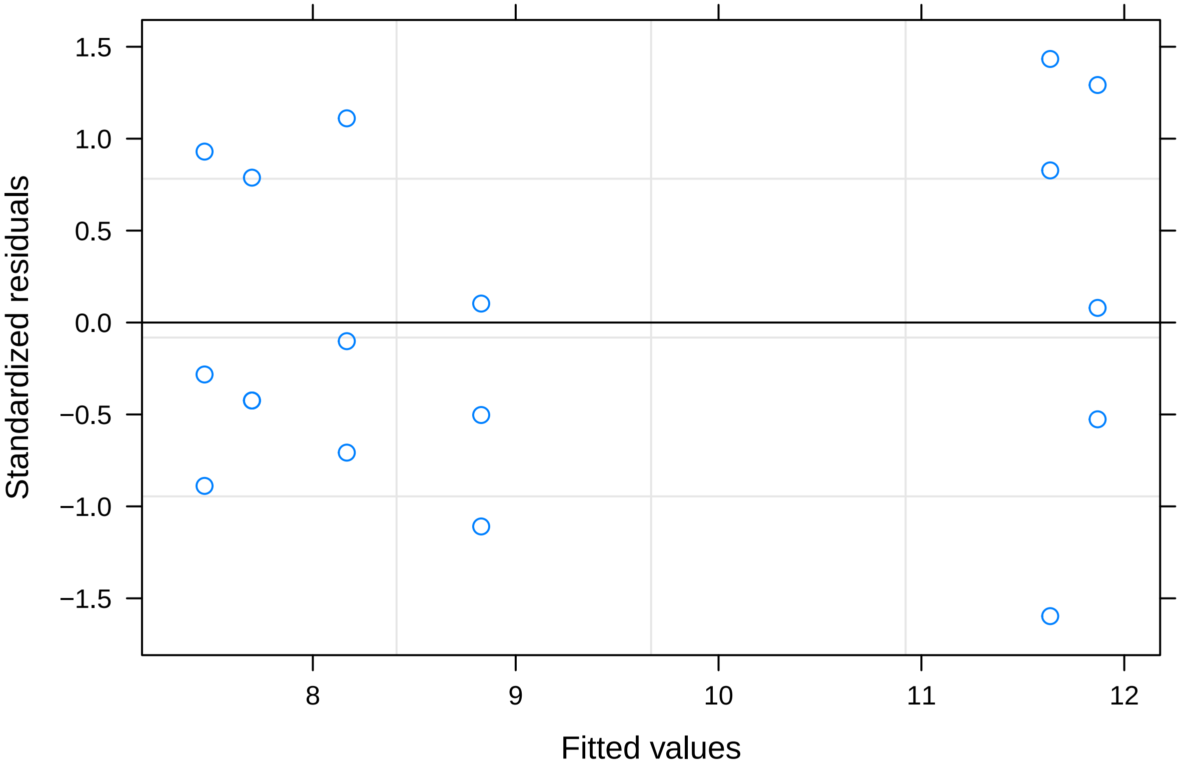
#2a linear models: to check for difference between habitat types with the implementation of mixed effect models.

```
mod1 <- lme(species_richness~category, random = (~1|site), data=dat) #model structure, random=... speci
summary(mod1) #model output - important is the "fixed effects" part. Here "forest" is hiding in the "In
```

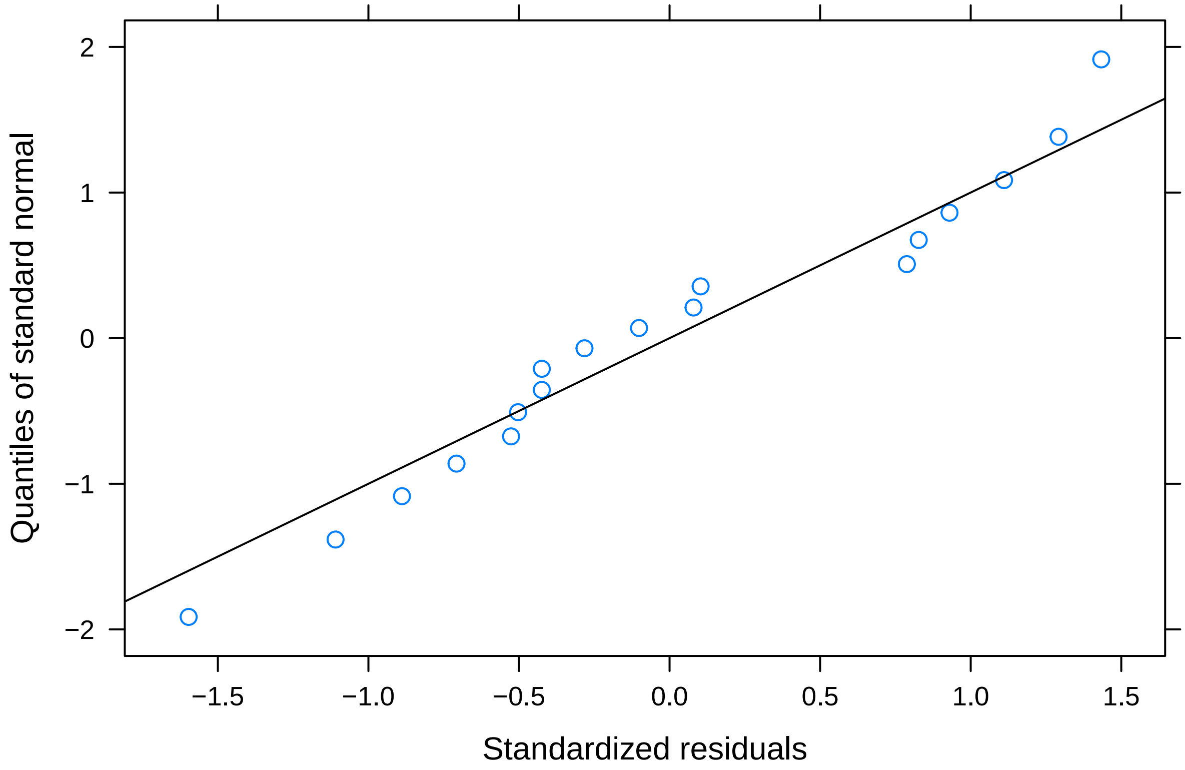
```
## Linear mixed-effects model fit by REML
##   Data: dat
##        AIC     BIC    logLik
##   78.65584 81.7462 -35.32792
##
## Random effects:
##  Formula: ~1 | site
##         (Intercept) Residual
## StdDev:    1.459325 1.649916
##
## Fixed effects:  species_richness ~ category
##                 Value Std.Error DF  t-value p-value
## (Intercept)  7.777778  1.006154 12 7.730207  0.0000
## categorypark 3.000000  1.422916  4 2.108346  0.1027
##  Correlation:
##              (Intr)
## categorypark -0.707
##
```

```
## Standardized Within-Group Residuals:
##        Min         Q1        Med         Q3        Max 
## -1.5969444 -0.5205776 -0.1919700  0.8175666  1.4335132 
##
## Number of Observations: 18
## Number of Groups: 6
```

```
plot(mod1) #check for homogeneity of variances (data points should have similar vertical spread along t
```



```
qqnorm(mod1, ~resid(.,type="p"), abline = c(0,1)) #check for normality of residuals (should not be comp
```



#2b linear models to include environmental variables and see the variation of the data with each of them.

```
### MIXED EFFECTS MODEL WITH SPECIES RICHNESS

round(cor(dat[,32:45]),2) #check which predictor variables are strongly correlated (below -0.7 or above
```

```
##                  canopy_cover n_tree_spec n_tree_ind dbh_min dbh_mean dbh_max
## canopy_cover             1.00       -0.68       0.22   -0.43    -0.42   -0.33
## n_tree_spec             -0.68        1.00      -0.42    0.30     0.31    0.23
## n_tree_ind               0.22       -0.42       1.00   -0.12    -0.49   -0.52
## dbh_min                 -0.43        0.30      -0.12    1.00     0.68    0.26
## dbh_mean                -0.42        0.31      -0.49    0.68     1.00    0.83
## dbh_max                 -0.33        0.23      -0.52    0.26     0.83    1.00
## dbh_median              -0.29        0.29      -0.56    0.30     0.74    0.57
## dbh_sd                  -0.69        0.78      -0.63    0.09     0.36    0.47
## n_microhabitats          0.30       -0.42      -0.12   -0.06     0.27    0.14
## latitude                 0.30       -0.60       0.22   -0.23    -0.08    0.03
## longitude                0.71       -0.64       0.32   -0.57    -0.42   -0.19
## size                     0.64       -0.52       0.35   -0.33    -0.52   -0.52
## temperature             -0.02        0.03       0.44   -0.29    -0.26   -0.06
## species_richness        -0.60        0.28      -0.29    0.72     0.61    0.34
##                  dbh_median dbh_sd n_microhabitats latitude longitude  size
## canopy_cover          -0.29  -0.69            0.30     0.30      0.71  0.64
## n_tree_spec            0.29   0.78           -0.42    -0.60     -0.64 -0.52
## n_tree_ind            -0.56  -0.63           -0.12     0.22      0.32  0.35
## dbh_min                0.30   0.09           -0.06    -0.23     -0.57 -0.33
## dbh_mean               0.74   0.36            0.27    -0.08     -0.42 -0.52
## dbh_max                0.57   0.47            0.14     0.03     -0.19 -0.52
## dbh_median             1.00   0.44            0.49    -0.06     -0.39 -0.48
## dbh_sd                 0.44   1.00           -0.32    -0.40     -0.58 -0.61
## n_microhabitats        0.49  -0.32            1.00     0.30      0.21  0.22
## latitude              -0.06  -0.40            0.30     1.00      0.14 -0.09
## longitude             -0.39  -0.58            0.21     0.14      1.00  0.80
## size                  -0.48  -0.61            0.22    -0.09      0.80  1.00
## temperature           -0.38  -0.02           -0.45     0.05      0.23 -0.07
## species_richness       0.40   0.28            0.03    -0.01     -0.68 -0.45
##                  temperature species_richness
## canopy_cover           -0.02            -0.60
## n_tree_spec             0.03             0.28
## n_tree_ind              0.44            -0.29
## dbh_min                -0.29             0.72
## dbh_mean               -0.26             0.61
## dbh_max                -0.06             0.34
## dbh_median             -0.38             0.40
## dbh_sd                 -0.02             0.28
## n_microhabitats        -0.45             0.03
## latitude                0.05            -0.01
## longitude               0.23            -0.68
## size                   -0.07            -0.45
## temperature             1.00            -0.49
## species_richness       -0.49             1.00
```

```
mod1 <- lme(species_richness ~ category*size + canopy_cover + n_tree_spec + n_tree_ind + log(dbh_min) +
summary(mod1)
```

```
## Linear mixed-effects model fit by maximum likelihood
##   Data: dat
##        AIC      BIC    logLik
##   78.65435 89.33881 -27.32718
##
## Random effects:
##  Formula: ~1 | site
##          (Intercept) Residual
## StdDev: 2.708844e-05 1.104329
##
## Fixed effects:  species_richness ~ category * size + canopy_cover + n_tree_spec +      n_tree_ind + l
##                       Value Std.Error DF    t-value p-value
## (Intercept)       12.734568  5.578051  7  2.2829782  0.0564
## categorypark      -0.522334  2.725272  1 -0.1916630  0.8794
## size              -0.004117  0.011377  1 -0.3618820  0.7790
## canopy_cover      -0.043044  0.026424  7 -1.6289885  0.1473
## n_tree_spec       -0.255434  0.345011  7 -0.7403651  0.4832
## n_tree_ind         0.000272  0.155205  7  0.0017531  0.9987
## log(dbh_min)       1.494185  0.808408  7  1.8483053  0.1070
## n_microhabitats   -0.025560  0.083210  7 -0.3071698  0.7677
## temperature       -0.196110  0.166034  1 -1.1811413  0.4473
## categorypark:size  0.005771  0.060341  1  0.0956404  0.9393
##  Correlation:
##                   (Intr) ctgryp size   cnpy_c n_tr_s n_tr_n lg(d_) n_mcrh
## categorypark      -0.294
## size              -0.308  0.736
## canopy_cover      -0.569  0.348  0.026
## n_tree_spec       -0.420 -0.472 -0.325  0.128
## n_tree_ind        -0.469  0.179 -0.029  0.142  0.415
## log(dbh_min)      -0.452 -0.421 -0.125  0.023  0.388  0.079
## n_microhabitats   -0.285  0.068  0.051  0.079  0.023 -0.117 -0.160
## temperature       -0.333  0.206  0.332  0.154 -0.330 -0.510  0.085  0.478
## categorypark:size  0.209  0.078  0.018  0.165 -0.480 -0.512 -0.441  0.428
##                   tmprtr
## categorypark
## size
## canopy_cover
## n_tree_spec
## n_tree_ind
## log(dbh_min)
## n_microhabitats
## temperature
## categorypark:size  0.467
##
## Standardized Within-Group Residuals:
##        Min         Q1        Med         Q3        Max
## -1.6352996 -0.8620140  0.0303186  0.7301579  2.2153409
##
## Number of Observations: 18
## Number of Groups: 6
```

```
mod1.1 <- stepAIC(mod1) #model simplification based on AIC-value of the model
```

```
## Start:  AIC=78.65
## species_richness ~ category * size + canopy_cover + n_tree_spec +
##     n_tree_ind + log(dbh_min) + n_microhabitats + temperature
##
##                   Df    AIC
## - n_tree_ind       1 76.654
## - category:size    1 76.675
## - n_microhabitats  1 76.865
## - n_tree_spec      1 77.847
## <none>               78.654
## - temperature      1 79.548
## - canopy_cover     1 81.811
## - log(dbh_min)     1 83.055
##
## Step:  AIC=76.65
## species_richness ~ category + size + canopy_cover + n_tree_spec +
##     log(dbh_min) + n_microhabitats + temperature + category:size
##
##                   Df    AIC
## - category:size    1 74.683
## - n_microhabitats  1 74.868
## - n_tree_spec      1 76.088
## <none>               76.654
## - temperature      1 78.457
## - canopy_cover     1 79.904
## - log(dbh_min)     1 81.088
```

```
##
## Step:  AIC=74.68
## species_richness ~ category + size + canopy_cover + n_tree_spec +
##     log(dbh_min) + n_microhabitats + temperature
##
##                   Df    AIC
## - category         1 72.793
## - size             1 72.975
## - n_microhabitats  1 73.033
## - n_tree_spec      1 74.153
## <none>               74.683
## - temperature      1 76.935
## - canopy_cover     1 78.491
## - log(dbh_min)     1 80.929
##
## Step:  AIC=72.79
## species_richness ~ size + canopy_cover + n_tree_spec + log(dbh_min) +
##     n_microhabitats + temperature
##
##                   Df    AIC
## - size             1 70.992
## - n_microhabitats  1 71.139
## <none>               72.793
## - n_tree_spec      1 73.670
## - temperature      1 75.045
## - canopy_cover     1 76.638
## - log(dbh_min)     1 79.633
##
## Step:  AIC=70.99
## species_richness ~ canopy_cover + n_tree_spec + log(dbh_min) +
##     n_microhabitats + temperature
##
##                   Df    AIC
## - n_microhabitats  1 69.299
## <none>               70.992
## - n_tree_spec      1 71.688
## - temperature      1 73.048
## - canopy_cover     1 75.907
## - log(dbh_min)     1 78.993
##
## Step:  AIC=69.3
## species_richness ~ canopy_cover + n_tree_spec + log(dbh_min) +
##     temperature
##
##                Df    AIC
## <none>            69.299
## - n_tree_spec   1 69.688
## - temperature   1 71.200
## - canopy_cover  1 74.131
## - log(dbh_min)  1 77.308
```

```
summary(mod1.1) #final model which includes only the most important predictors
```

```
## Linear mixed-effects model fit by maximum likelihood
##   Data: dat
##        AIC      BIC    logLik
##   69.29947 75.53207 -27.64974
##
## Random effects:
##  Formula: ~1 | site
##         (Intercept) Residual
## StdDev: 2.762129e-05 1.124297
##
## Fixed effects:  species_richness ~ canopy_cover + n_tree_spec + log(dbh_min) +      temperature
##                  Value Std.Error DF   t-value p-value
## (Intercept)  11.366118  3.314519  9  3.429190  0.0075
## canopy_cover -0.044574  0.018196  9 -2.449668  0.0368
## n_tree_spec  -0.241219  0.177235  9 -1.361014  0.2066
## log(dbh_min)  1.542315  0.496008  9  3.109455  0.0125
## temperature  -0.160682  0.090601  4 -1.773506  0.1508
##  Correlation:
##              (Intr) cnpy_c n_tr_s lg(d_)
## canopy_cover -0.791
## n_tree_spec  -0.495  0.595
## log(dbh_min) -0.759  0.467 -0.013
## temperature  -0.681  0.241 -0.024  0.519
##
## Standardized Within-Group Residuals:
##         Min          Q1         Med          Q3         Max
## -1.67276372 -0.80102592 -0.08660587  0.68432847  2.15093491
##
## Number of Observations: 18
## Number of Groups: 6
```

```
mod1.2 <- update(mod1.1, ~.-n_tree_spec)
summary(mod1.2)
```

```
## Linear mixed-effects model fit by maximum likelihood
##   Data: dat
##        AIC      BIC    logLik
##   69.68834 75.03057 -28.84417
##
## Random effects:
##  Formula: ~1 | site
##         (Intercept) Residual
## StdDev:   0.1799486 1.188182
##
## Fixed effects:  species_richness ~ canopy_cover + log(dbh_min) + temperature
##                  Value Std.Error DF   t-value p-value
## (Intercept)   9.140053 2.9508347 10  3.097447  0.0113
## canopy_cover -0.029890 0.0150141 10 -1.990770  0.0745
## log(dbh_min)  1.532576 0.5067240 10  3.024479  0.0128
## temperature  -0.163764 0.0945977  4 -1.731167  0.1585
##  Correlation:
##              (Intr) cnpy_c lg(d_)
## canopy_cover -0.700
## log(dbh_min) -0.875  0.574
## temperature  -0.797  0.303  0.507
```

```
##
## Standardized Within-Group Residuals:
##         Min          Q1         Med          Q3         Max
## -1.47877950 -0.81708368  0.01400286  0.42067808  2.51962252
##
## Number of Observations: 18
## Number of Groups: 6
```

```
anova(mod1.1, mod1.2)
```

```
##        Model df      AIC      BIC    logLik   Test L.Ratio p-value
## mod1.1     1  7 69.29947 75.53207 -27.64973
## mod1.2     2  6 69.68834 75.03057 -28.84417 1 vs 2 2.38887  0.1222
```

```
mod1.3 <- update(mod1.2, ~.-temperature)
summary(mod1.3)
```

```
## Linear mixed-effects model fit by maximum likelihood
##   Data: dat
##        AIC      BIC    logLik
##   70.90548 75.35733 -30.45274
##
## Random effects:
##  Formula: ~1 | site
##         (Intercept) Residual
## StdDev:    0.550221 1.212528
##
## Fixed effects:  species_richness ~ canopy_cover + log(dbh_min)
##                   Value Std.Error DF   t-value p-value
## (Intercept)    5.562682 1.8467347 10  3.012172  0.0131
## canopy_cover -0.025139 0.0152642 10 -1.646896  0.1306
## log(dbh_min)  1.862918 0.4588731 10  4.059768  0.0023
##  Correlation:
##              (Intr) cnpy_c
## canopy_cover -0.756
## log(dbh_min) -0.882  0.427
##
## Standardized Within-Group Residuals:
##        Min         Q1        Med         Q3        Max
## -1.6480211 -0.6031499 -0.2136438  0.7787394  1.8252090
##
## Number of Observations: 18
## Number of Groups: 6
```

```
anova(mod1.2, mod1.3) #close to significcance effect with these variables (species increasing with incr
```

```
##        Model df      AIC      BIC    logLik   Test  L.Ratio p-value
## mod1.2     1  6 69.68834 75.03057 -28.84417
## mod1.3     2  5 70.90548 75.35733 -30.45274 1 vs 2 3.217136  0.0729
```

```
mod1.4 <- update(mod1.3, ~.-canopy_cover)
summary(mod1.4)
```

```
## Linear mixed-effects model fit by maximum likelihood
##   Data: dat
##        AIC      BIC    logLik
##   71.50383 75.06532 -31.75191
##
## Random effects:
##  Formula: ~1 | site
##          (Intercept) Residual
## StdDev: 0.0001335345  1.41207
##
## Fixed effects:  species_richness ~ log(dbh_min)
##                 Value Std.Error DF  t-value p-value
## (Intercept)  2.833014 1.1783306 11 2.404261  0.0350
## log(dbh_min) 2.342352 0.4085936 11 5.732719  0.0001
##  Correlation:
##              (Intr)
## log(dbh_min) -0.954
##
## Standardized Within-Group Residuals:
##         Min          Q1         Med          Q3         Max
## -2.06423425 -0.69965226  0.06165372  0.63232038  1.83834164
##
## Number of Observations: 18
## Number of Groups: 6
```

```
anova(mod1.3,mod1.4)
```

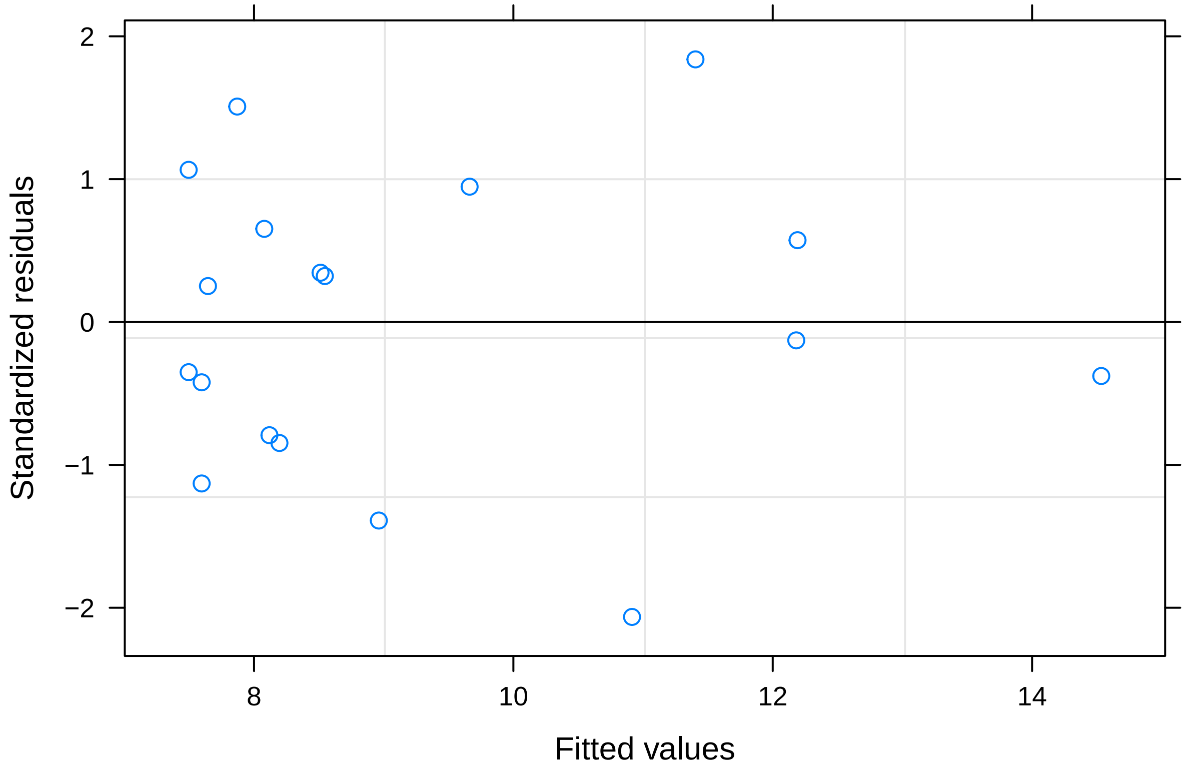
```
##        Model df      AIC      BIC    logLik   Test  L.Ratio p-value
## mod1.3     1  5 70.90548 75.35733 -30.45274
## mod1.4     2  4 71.50383 75.06532 -31.75191 1 vs 2 2.598353   0.107
```

```
#final model:
#lme(species_richness ~ log(dbh_min), random = (~1|site), data=dat, method="ML")
#https://jonlefcheck.net/2013/03/13/r2-for-linear-mixed-effects-models/
r.squaredGLMM(mod1.4)
```

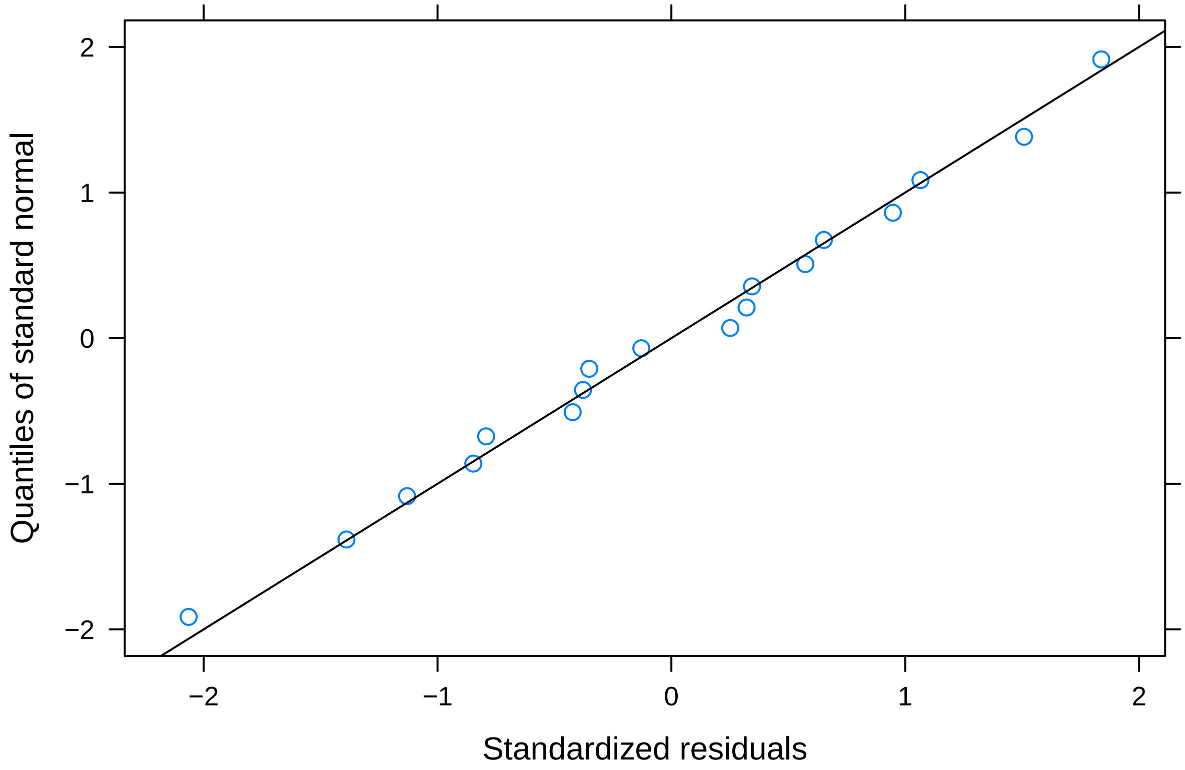
```
## Warning: 'r.squaredGLMM' now calculates a revised statistic. See the help page.
```

```
##            R2m       R2c
## [1,] 0.6850223 0.6850223
```

```
plot(mod1.4) #check for homogeneity of variances (data points should have similar vertical spread along
```



```
qqnorm(mod1.4, ~resid(.,type="p"), abline=c(0,1)) #check for normality of residuals (should not be comp
```



```
#standard deviation, coefficient of variation (sd/mean) to make variation independent from out mean val

#dat$sddbh_mean <- sd() we can try as alternative
library(car)
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```
## The following object is masked from 'package:purrr':
##
##     some
```

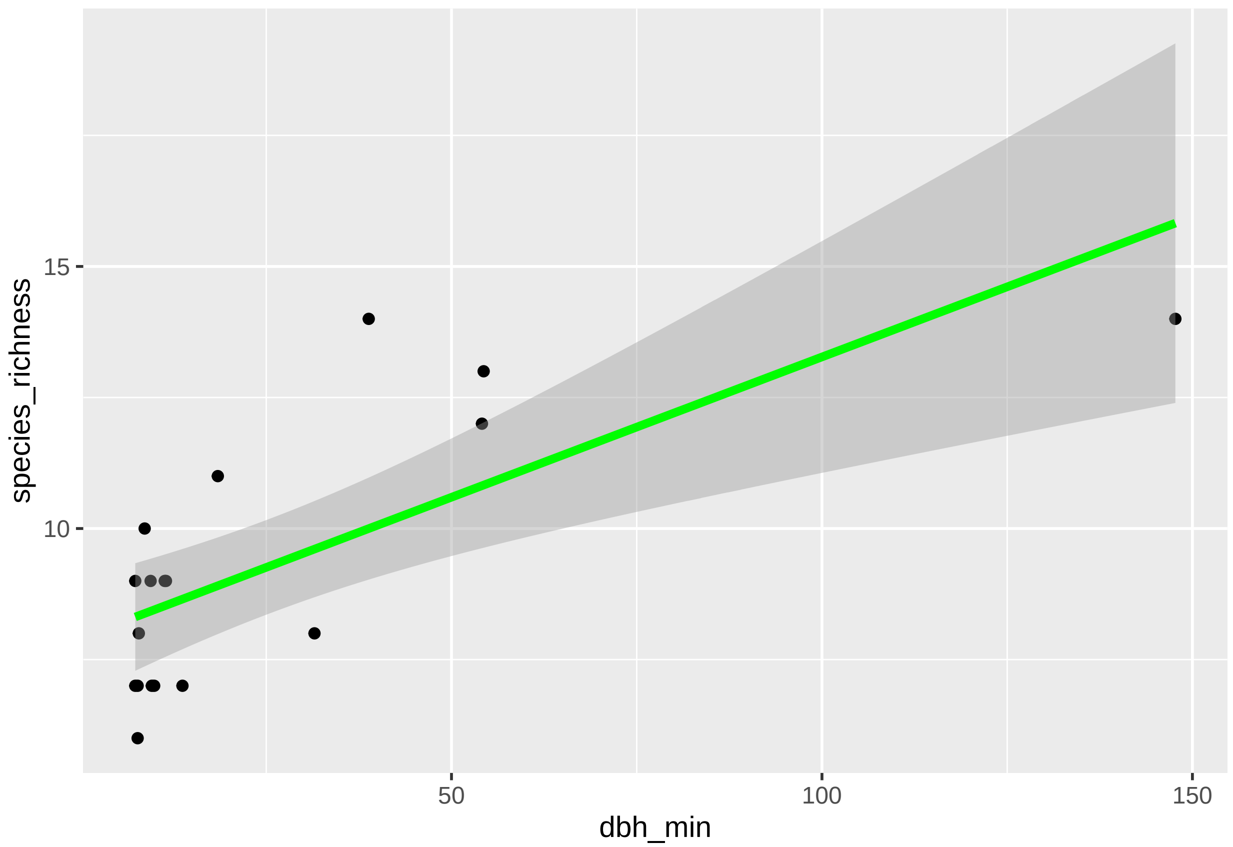
```
vif(mod1.2)
```

```
## canopy_cover log(dbh_min)  temperature
##     1.492654     1.824734     1.346892
```

```
ggplot(dat,aes(x = dbh_min, y = species_richness)) +
  geom_point() +
  geom_smooth(method = lm, se=TRUE, colour = 'green', size = 1.5, width = 2)
```

```
## Warning: Ignoring unknown parameters: width
```

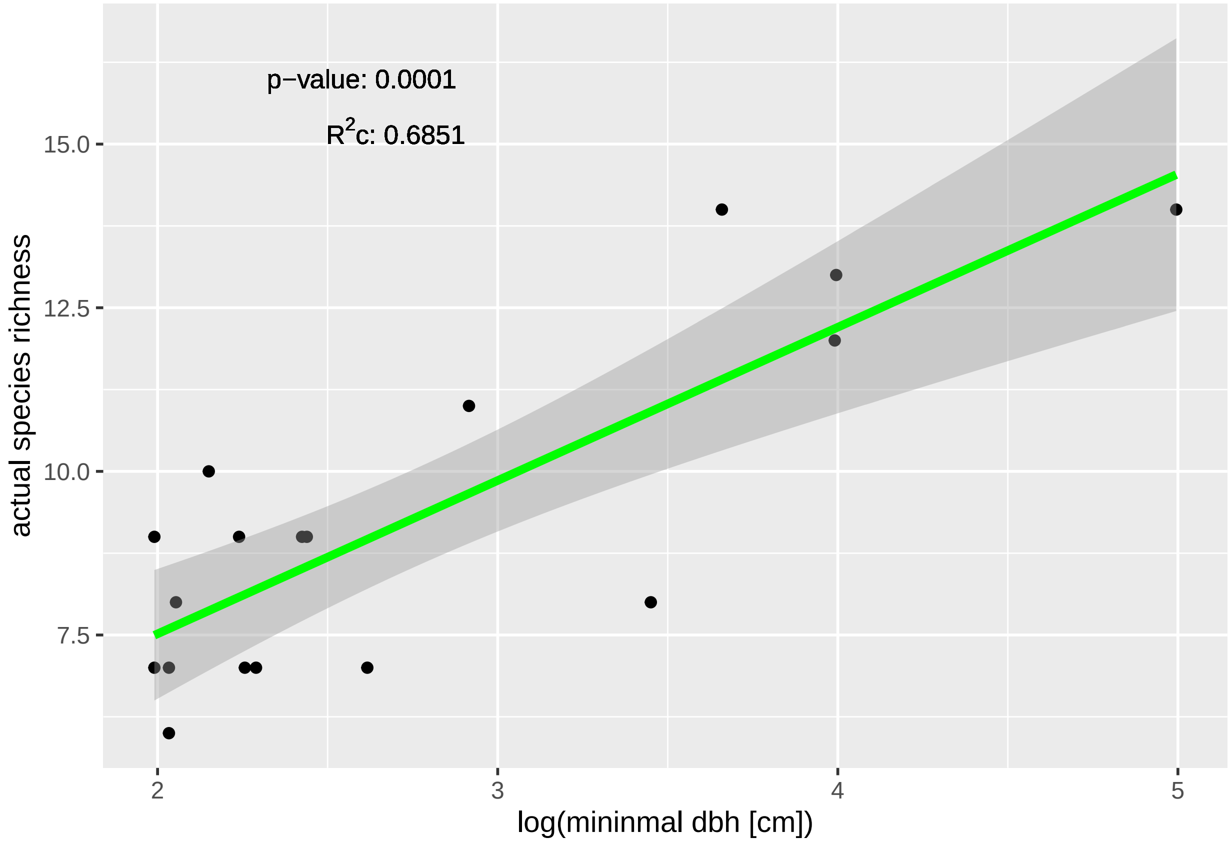
```
## `geom_smooth()` using formula 'y ~ x'
```



```
ggplot(dat, aes(x = log(dbh_min), y = species_richness))+
  geom_point() +
  geom_smooth(method = lm,se = TRUE, colour = 'green', size = 1.5) +
  geom_text(x=2.6, y=16, label="p-value: 0.0001", size=3.5) +
  geom_text(x=2.7,y=15.2,label=expression(paste("R"^2,"c: 0.6851")), size=3.5) +
  labs(x="log(mininmal dbh [cm])", y="actual species richness")
```

```
## `geom_smooth()` using formula 'y ~ x'
```

```
## Warning in is.na(x): is.na() aplicado a un objeto que no es (lista o vector) de
## tipo 'expression
```



```
### MIXED EFFECTS MODEL WITH BIRD ABUNDANCE
mod2.1 <- lme(species_abund ~ category*size + canopy_cover + n_tree_spec + n_tree_ind + log(dbh_min) +
summary(mod2.1)
```

```
## Linear mixed-effects model fit by maximum likelihood
##   Data: dat
##        AIC      BIC    logLik
##   109.115 119.7994 -42.55748
##
## Random effects:
##  Formula: ~1 | site
##          (Intercept) Residual
## StdDev: 5.556265e-05 2.573756
##
## Fixed effects:  species_abund ~ category * size + canopy_cover + n_tree_spec +      n_tree_ind + log
##                         Value Std.Error DF    t-value p-value
## (Intercept)         27.149524 13.000237  7  2.0883868  0.0751
## categorypark         0.107528  6.351534  1  0.0169294  0.9892
## size                -0.025030  0.026516  1 -0.9439715  0.5183
## canopy_cover        -0.111826  0.061584  7 -1.8158318  0.1123
## n_tree_spec         -0.326213  0.804084  7 -0.4056946  0.6971
## n_tree_ind           0.013514  0.361721  7  0.0373606  0.9712
## log(dbh_min)         2.523681  1.884081  7  1.3394760  0.2223
## n_microhabitats      0.282745  0.193930  7  1.4579710  0.1882
## temperature         -0.379197  0.386960  1 -0.9799367  0.5065
## categorypark:size -0.081267  0.140630  1 -0.5778744  0.6664
##  Correlation:
##                    (Intr) ctgryp size   cnpy_c n_tr_s n_tr_n lg(d_) n_mcrh
## categorypark      -0.294
## size              -0.308  0.736
## canopy_cover      -0.569  0.348  0.026
## n_tree_spec       -0.420 -0.472 -0.325  0.128
## n_tree_ind        -0.469  0.179 -0.029  0.142  0.415
## log(dbh_min)      -0.452 -0.421 -0.125  0.023  0.388  0.079
## n_microhabitats   -0.285  0.068  0.051  0.079  0.023 -0.117 -0.160
## temperature       -0.333  0.206  0.332  0.154 -0.330 -0.510  0.085  0.478
## categorypark:size  0.209  0.078  0.018  0.165 -0.480 -0.512 -0.441  0.428
##                    tmprtr
## categorypark
## size
## canopy_cover
## n_tree_spec
## n_tree_ind
## log(dbh_min)
## n_microhabitats
## temperature
## categorypark:size  0.467
##
## Standardized Within-Group Residuals:
##         Min          Q1         Med          Q3         Max
## -1.68676402 -0.60617909 -0.04981096  0.66364849  1.77306497
##
## Number of Observations: 18
## Number of Groups: 6
```

```
mod2.2 <- stepAIC(mod2.1) #model simplification based on AIC-value of the model
```

```
## Start:  AIC=109.11
## species_abund ~ category * size + canopy_cover + n_tree_spec +
##     n_tree_ind + log(dbh_min) + n_microhabitats + temperature
##
##                   Df    AIC
## - n_tree_ind       1 107.12
## - n_tree_spec      1 107.48
## - category:size    1 107.85
## <none>               109.11
## - temperature      1 109.16
## - log(dbh_min)     1 110.76
## - n_microhabitats  1 111.36
## - canopy_cover     1 113.33
##
## Step:  AIC=107.12
## species_abund ~ category + size + canopy_cover + n_tree_spec +
##     log(dbh_min) + n_microhabitats + temperature + category:size
##
##                   Df    AIC
## - n_tree_spec      1 105.59
## - category:size    1 106.05
## <none>               107.12
## - temperature      1 107.73
## - log(dbh_min)     1 108.77
## - n_microhabitats  1 109.43
## - canopy_cover     1 111.47
##
## Step:  AIC=105.59
## species_abund ~ category + size + canopy_cover + log(dbh_min) +
##     n_microhabitats + temperature + category:size
##
##                   Df    AIC
## - category:size    1 105.15
## <none>              105.59
## - temperature      1 106.51
## - n_microhabitats  1 108.02
## - log(dbh_min)     1 108.77
## - canopy_cover     1 109.63
##
## Step:  AIC=105.15
## species_abund ~ category + size + canopy_cover + log(dbh_min) +
##     n_microhabitats + temperature
##
##                   Df    AIC
## - category         1 103.40
## - temperature      1 105.13
## <none>              105.15
## - size             1 106.25
## - log(dbh_min)     1 106.86
## - canopy_cover     1 107.76
## - n_microhabitats  1 111.05
##
## Step:  AIC=103.4
## species_abund ~ size + canopy_cover + log(dbh_min) + n_microhabitats +
##     temperature
##
##                   Df    AIC
## - temperature      1 103.13
## <none>              103.40
## - log(dbh_min)     1 104.88
## - size             1 105.53
## - canopy_cover     1 106.22
## - n_microhabitats  1 109.93
##
## Step:  AIC=103.13
## species_abund ~ size + canopy_cover + log(dbh_min) + n_microhabitats
##
##                   Df    AIC
## <none>              103.13
## - size             1 104.18
## - canopy_cover     1 105.33
## - log(dbh_min)     1 108.31
## - n_microhabitats  1 112.79
```

```
summary(mod2.2) #final model which includes only the most important predictors
```

```
## Linear mixed-effects model fit by maximum likelihood
##   Data: dat
##        AIC      BIC    logLik
##   103.1345 109.3671 -44.56723
##
## Random effects:
##  Formula: ~1 | site
##         (Intercept) Residual
## StdDev: 6.954709e-05 2.877779
##
## Fixed effects:  species_abund ~ size + canopy_cover + log(dbh_min) + n_microhabitats
##                     Value Std.Error DF   t-value p-value
## (Intercept)     15.129106  4.497654  9  3.363777  0.0083
## size            -0.022182  0.014328  4 -1.548144  0.1965
## canopy_cover    -0.079938  0.043270  9 -1.847435  0.0977
## log(dbh_min)     2.864655  1.135269  9  2.523327  0.0326
## n_microhabitats  0.450631  0.129992  9  3.466596  0.0071
##  Correlation:
##                 (Intr) size   cnpy_c lg(d_)
## size            -0.119
## canopy_cover    -0.622 -0.479
## log(dbh_min)    -0.902  0.218  0.369
## n_microhabitats  0.106 -0.081 -0.274 -0.213
##
## Standardized Within-Group Residuals:
##        Min         Q1        Med         Q3        Max
## -1.4515950 -0.7063230 -0.1247196  0.8071915  1.7575253
##
## Number of Observations: 18
## Number of Groups: 6
```

```
mod2.3 <- update(mod2.2, ~.-size)
summary(mod2.3)
```

```
## Linear mixed-effects model fit by maximum likelihood
##   Data: dat
##        AIC      BIC    logLik
##   104.1802 109.5224 -46.09009
##
## Random effects:
##  Formula: ~1 | site
##         (Intercept) Residual
## StdDev: 0.0001524278 3.131845
##
## Fixed effects:  species_abund ~ canopy_cover + log(dbh_min) + n_microhabitats
##                     Value Std.Error DF   t-value p-value
## (Intercept)     14.298817  4.683029  9  3.053327  0.0137
## canopy_cover    -0.112004  0.039841  9 -2.811286  0.0203
## log(dbh_min)     3.248018  1.161888  9  2.795466  0.0209
## n_microhabitats  0.434242  0.135870  9  3.196014  0.0109
##  Correlation:
##                 (Intr) cnpy_c lg(d_)
## canopy_cover    -0.779
## log(dbh_min)    -0.904  0.552
## n_microhabitats  0.097 -0.358 -0.201
```

```
##
## Standardized Within-Group Residuals:
##        Min         Q1        Med         Q3        Max
## -1.8809748 -0.4536809 -0.1107584  0.9669037  1.8288680
##
## Number of Observations: 18
## Number of Groups: 6
```

```
anova(mod2.2, mod2.3)
```

```
##        Model df      AIC      BIC    logLik   Test  L.Ratio p-value
## mod2.2     1  7 103.1345 109.3671 -44.56723
## mod2.3     2  6 104.1802 109.5224 -46.09009 1 vs 2 3.045726  0.0809
```

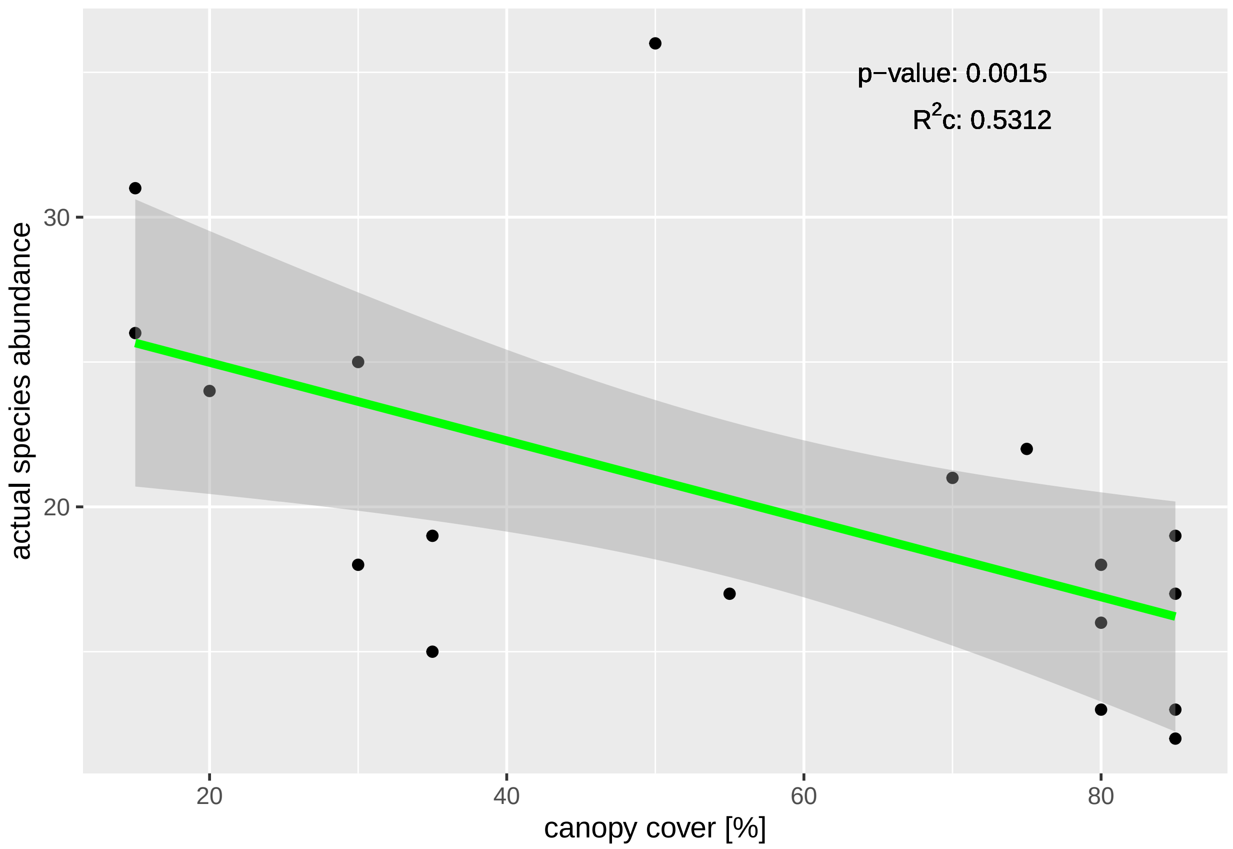
```
#mod2.4<-update(mod2.3, ~.-canopy_cover)
#summary(mod2.4)
#anova(mod2.3,mod2.4)

# plot model for abundance vs. canopy cover
mod_can<-lme(species_abund ~ canopy_cover, random = (~1|site), data=dat, method="ML")
r.squaredGLMM(mod_can)
```

```
##            R2m       R2c
## [1,] 0.3208035 0.5312365
```

```
ggplot(dat, aes(x = canopy_cover, y = species_abund))+
  geom_point() +
  geom_smooth(method = lm, se = TRUE, colour = 'green', size = 1.5) +
  geom_text(x=70, y=35, label="p-value: 0.0015", size=3.5) +
  geom_text(x=72,y=33.5, label=expression(paste("R"^2,"c: 0.5312")), size=3.5) +
  labs(x="canopy cover [%]", y="actual species abundance")
```

```
## `geom_smooth()` using formula 'y ~ x'
```

```
## Warning in is.na(x): is.na() aplicado a un objeto que no es (lista o vector) de
## tipo 'expression
```



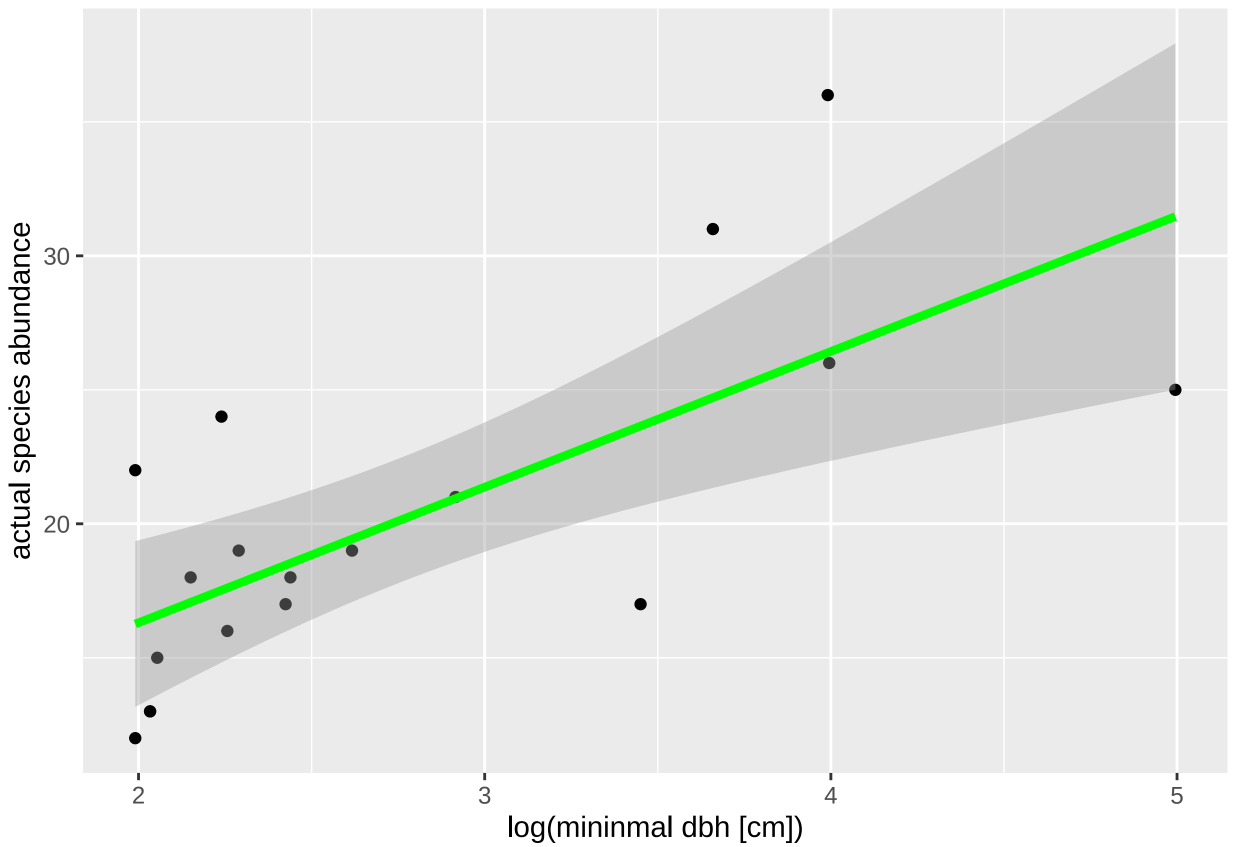
```
mod_dbh<-lme(species_abund ~ log(dbh_min), random = (~1|site), data=dat, method="ML")
r.squaredGLMM(mod_dbh)
```

```
##            R2m       R2c
## [1,] 0.506479 0.5225862
```

```
ggplot(dat, aes(x = log(dbh_min), y = species_abund))+
  geom_point() +
  geom_smooth(method = lm, se = TRUE, colour = 'green', size = 1.5) +
  geom_text(x=70, y=35, label="p-value: 0.0015", size=3.5) +
  geom_text(x=72,y=33.5, label=expression(paste("R"^2,"c: 0.5312")), size=3.5) +
  labs(x="log(mininmal dbh [cm])", y="actual species abundance")
```

```
## ‘geom_smooth()‘ using formula ’y ~ x’
```

```
## Warning in is.na(x): is.na() aplicado a un objeto que no es (lista o vector) de
## tipo ’expression
```



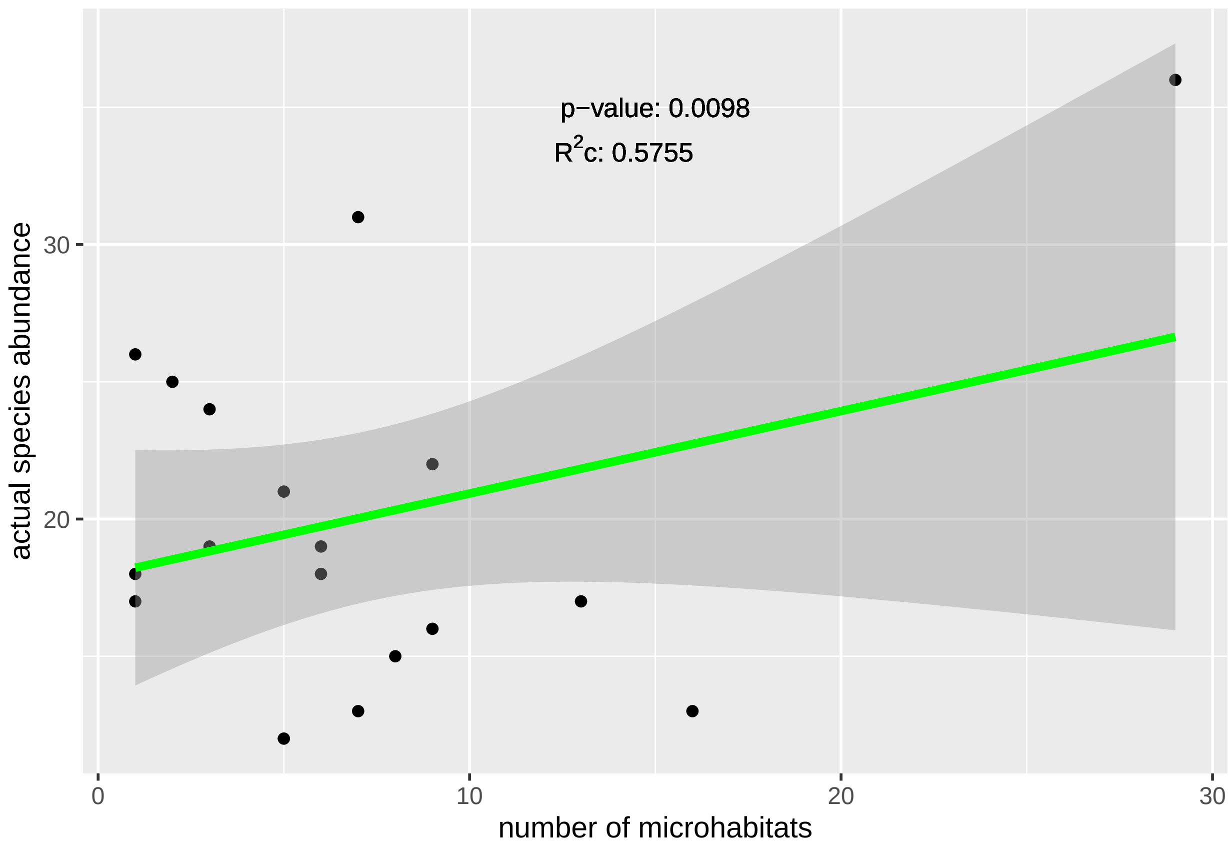
```
# plot model for abundance vs. microhabitats
mod_mic<-lme(species_abund ~ n_microhabitats, random = (~1|site), data=dat, method="ML")
r.squaredGLMM(mod_mic)
```

```
##            R2m       R2c
## [1,] 0.1355099 0.5754792
```

```
ggplot(dat, aes(x = n_microhabitats, y = species_abund))+
  geom_point() +
  geom_smooth(method = lm, se = TRUE, colour = 'green', size = 1.5) +
  geom_text(x=15, y=35, label="p-value: 0.0098", size=3.5) +
  geom_text(x=14.15,y=33.5, label=expression(paste("R"^2,"c: 0.5755")), size=3.5) +
  labs(x="number of microhabitats", y="actual species abundance")
```

```
## ‘geom_smooth()‘ using formula ’y ~ x’
```

```
## Warning in is.na(x): is.na() aplicado a un objeto que no es (lista o vector) de
## tipo ’expression
```



```
### MIXED EFFECTS MODEL WITH RAREFIED RICHNESS
mod3 <- lme(rarefied_richness ~ category*size + canopy_cover + n_tree_spec + n_tree_ind + log(dbh_min) +
summary(mod3)
```

```
## Linear mixed-effects model fit by maximum likelihood
##   Data: dat
##        AIC      BIC    logLik
##   58.56492 69.24938 -17.28246
##
## Random effects:
##  Formula: ~1 | site
##          (Intercept)  Residual
## StdDev: 2.442777e-05 0.6320405
##
## Fixed effects:  rarefied_richness ~ category * size + canopy_cover + n_tree_spec +      n_tree_ind +
##                       Value Std.Error DF    t-value p-value
## (Intercept)        8.419457  3.192485  7  2.6372736  0.0336
## categorypark      -0.269752  1.559754  1 -0.1729449  0.8910
## size              -0.000680  0.006511  1 -0.1044220  0.9338
## canopy_cover      -0.009324  0.015123  7 -0.6165055  0.5571
## n_tree_spec       -0.193305  0.197460  7 -0.9789587  0.3602
## n_tree_ind        -0.026853  0.088828  7 -0.3022983  0.7712
## log(dbh_min)       0.459643  0.462676  7  0.9934439  0.3536
## n_microhabitats   -0.007390  0.047624  7 -0.1551677  0.8811
## temperature       -0.019742  0.095026  1 -0.2077477  0.8696
## categorypark:size  0.017530  0.034535  1  0.5076063  0.7010
##  Correlation: 
##                   (Intr) ctgryp size   cnpy_c n_tr_s n_tr_n lg(d_) n_mcrh
## categorypark      -0.294                                                 
## size              -0.308  0.736                                          
## canopy_cover      -0.569  0.348  0.026                                   
## n_tree_spec       -0.420 -0.472 -0.325  0.128                            
## n_tree_ind        -0.469  0.179 -0.029  0.142  0.415                     
## log(dbh_min)      -0.452 -0.421 -0.125  0.023  0.388  0.079              
## n_microhabitats   -0.285  0.068  0.051  0.079  0.023 -0.117 -0.160       
## temperature       -0.333  0.206  0.332  0.154 -0.330 -0.510  0.085  0.478
## categorypark:size  0.209  0.078  0.018  0.165 -0.480 -0.512 -0.441  0.428
##                   tmprtr
## categorypark            
## size                    
## canopy_cover            
## n_tree_spec             
## n_tree_ind              
## log(dbh_min)            
## n_microhabitats         
## temperature             
## categorypark:size  0.467
## 
## Standardized Within-Group Residuals:
##        Min         Q1        Med         Q3        Max 
## -2.1905486 -0.3771982  0.1406606  0.6747755  1.9225911 
## 
## Number of Observations: 18
## Number of Groups: 6
```

```
mod3.1 <- stepAIC(mod3) #model simplification based on AIC-value of the model
```

```
## Start:  AIC=58.56
## rarefied_richness ~ category * size + canopy_cover + n_tree_spec + 
##     n_tree_ind + log(dbh_min) + n_microhabitats + temperature
## 
##                   Df    AIC
## - n_microhabitats  1 56.619
## - temperature      1 56.662
## - n_tree_ind       1 56.769
## - category:size    1 57.136
## - canopy_cover     1 57.400
## - n_tree_spec      1 58.543
## <none>               58.565
## - log(dbh_min)     1 58.659
## 
## Step:  AIC=56.62
## rarefied_richness ~ category + size + canopy_cover + n_tree_spec + 
##     n_tree_ind + log(dbh_min) + temperature + category:size
## 
##                 Df    AIC
## - temperature    1 54.671
## - n_tree_ind     1 54.851
## - canopy_cover   1 55.425
## - category:size  1 55.502
## - n_tree_spec    1 56.591
## <none>             56.619
## - log(dbh_min)   1 56.659
##
## Step:  AIC=54.67
## rarefied_richness ~ category + size + canopy_cover + n_tree_spec +
##     n_tree_ind + log(dbh_min) + category:size
##
##                 Df    AIC
## - n_tree_ind     1 53.159
## - canopy_cover   1 53.436
## - category:size  1 53.814
## <none>             54.671
## - log(dbh_min)   1 54.890
## - n_tree_spec    1 55.119
##
## Step:  AIC=53.16
## rarefied_richness ~ category + size + canopy_cover + n_tree_spec +
##     log(dbh_min) + category:size
##
##                 Df    AIC
## - canopy_cover   1 51.664
## - category:size  1 51.880
## <none>             53.159
## - n_tree_spec    1 53.213
## - log(dbh_min)   1 53.774
##
## Step:  AIC=51.66
## rarefied_richness ~ category + size + n_tree_spec + log(dbh_min) +
##     category:size
##
##                 Df    AIC
## - category:size  1 50.738
## - n_tree_spec    1 51.381
## <none>             51.664
## - log(dbh_min)   1 52.136
##
## Step:  AIC=50.74
## rarefied_richness ~ category + size + n_tree_spec + log(dbh_min)
##
##                Df    AIC
## - size          1 48.740
## - category      1 48.753
## - n_tree_spec   1 49.576
## <none>            50.738
## - log(dbh_min)  1 53.379
##
## Step:  AIC=48.74
## rarefied_richness ~ category + n_tree_spec + log(dbh_min)
##
##                Df    AIC
## - category      1 46.758
## - n_tree_spec   1 47.633
## <none>            48.740
## - log(dbh_min)  1 51.712
##
## Step:  AIC=46.76
## rarefied_richness ~ n_tree_spec + log(dbh_min)
##
##                 Df    AIC
## - n_tree_spec   1 46.091
## <none>            46.758
## - log(dbh_min)  1 52.240
##
## Step:  AIC=46.09
## rarefied_richness ~ log(dbh_min)
##
##                 Df    AIC
## <none>            46.091
## - log(dbh_min)  1 50.433
```

```
summary(mod3.1) #final model which includes only the most important predictors
```

```
## Linear mixed-effects model fit by maximum likelihood
##   Data: dat
##        AIC      BIC    logLik
##   46.09053 49.65202 -19.04527
##
## Random effects:
##  Formula: ~1 | site
##         (Intercept)  Residual
## StdDev:   0.3015021 0.6402719
##
## Fixed effects:  rarefied_richness ~ log(dbh_min)
##                  Value Std.Error DF  t-value p-value
## (Intercept)  5.551859 0.6230627 11 8.910594  0.0000
## log(dbh_min) 0.567502 0.2136461 11 2.656270  0.0223
##  Correlation:
##              (Intr)
## log(dbh_min) -0.943
##
## Standardized Within-Group Residuals:
##         Min          Q1         Med          Q3         Max
## -1.78828290 -0.39209673  0.05314066  0.66857616  1.81864161
##
## Number of Observations: 18
## Number of Groups: 6
```

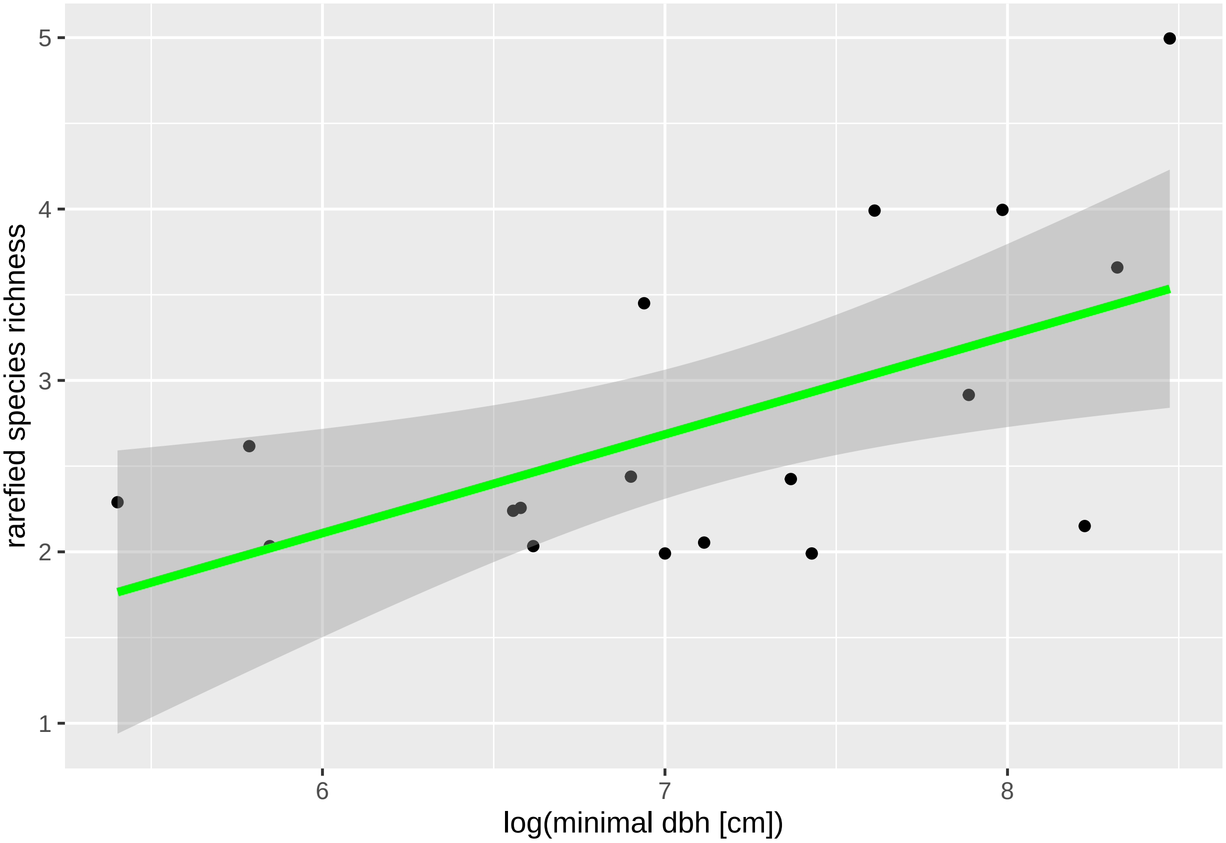
```
# plot model for rarefied richness vs. min dbh
mod_dbh<-lme(rarefied_richness ~ log(dbh_min), random = (~1|site), data=dat, method="ML")
r.squaredGLMM(mod3)
```

```
##            R2m       R2c
## [1,] 0.4834111 0.4834111
```

```r
ggplot(dat, aes(x = rarefied_richness, y = log(dbh_min)))+
  geom_point() +
  geom_smooth(method = lm, se = TRUE, colour = 'green', size = 1.5) +
  geom_text(x=70, y=35, label="p-value: 0.0223", size=3.5) +
  geom_text(x=72,y=33.5, label=expression(paste("R"^2,"c: 0.4573")), size=3.5) +
  labs(x="log(minimal dbh [cm])", y="rarefied species richness")
```

```
## `geom_smooth()` using formula 'y ~ x'
```

```
## Warning in is.na(x): is.na() aplicado a un objeto que no es (lista o vector) de
## tipo 'expression
```



```r
#plot(log(dbh_min)~rarefied_richness, data=dat, ylim=c(0,5))
#mod<-lm(rarefied_richness ~ log(dbh_min), data=dat, poly(degree = 2))
```

##2c Ordination with NMDS (to look for differences in species composition)

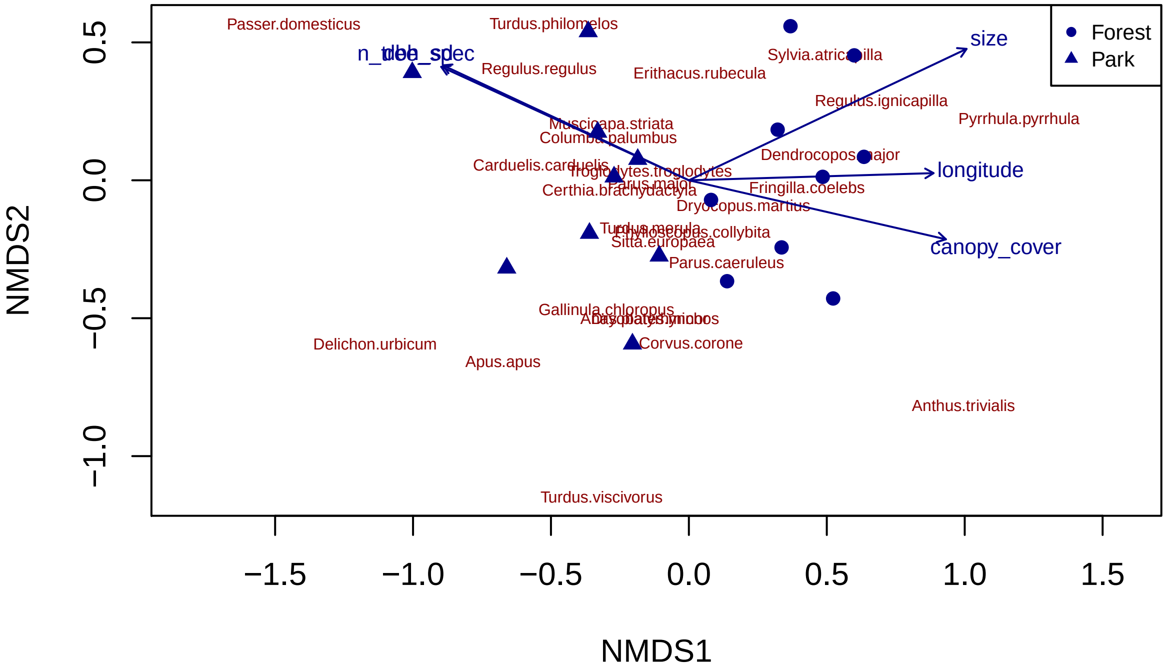
```r
nmd1 <- metaMDS(dat[,4:31], distance="horn", k=2) #NMDS analysis based on Morisita-Horn-Index as a diss
```

```
## Run 0 stress 0.1663514
## Run 1 stress 0.1663514
## ... New best solution
## ... Procrustes: rmse 2.616946e-05  max resid 4.446755e-05
## ... Similar to previous best
## Run 2 stress 0.1748461
## Run 3 stress 0.1777238
## Run 4 stress 0.1707767
## Run 5 stress 0.1707767
## Run 6 stress 0.1895839
## Run 7 stress 0.1707768
## Run 8 stress 0.174846
## Run 9 stress 0.1828469
## Run 10 stress 0.1663514
## ... New best solution
## ... Procrustes: rmse 3.64117e-05  max resid 0.0001229996
## ... Similar to previous best
## Run 11 stress 0.1663514
## ... New best solution
## ... Procrustes: rmse 2.981953e-05  max resid 9.738643e-05
## ... Similar to previous best
## Run 12 stress 0.174846
## Run 13 stress 0.1777238
## Run 14 stress 0.1707768
## Run 15 stress 0.1663514
## ... Procrustes: rmse 3.599549e-05  max resid 0.0001181214
## ... Similar to previous best
## Run 16 stress 0.1663514
## ... Procrustes: rmse 5.118595e-06  max resid 1.711481e-05
## ... Similar to previous best
## Run 17 stress 0.174846
## Run 18 stress 0.1663514
## ... Procrustes: rmse 3.882942e-06  max resid 1.207071e-05
## ... Similar to previous best
## Run 19 stress 0.174846
## Run 20 stress 0.1707767
## *** Solution reached
```

```
# orditkplot(nmd1, display = "species", col = "darkred", fill = NA, border = NA, cex = 0.6)
ordiplot(nmd1, choices = c(1, 2), type = "n") # ylim = c(-0.75, 0.5), xlim = c(-1.25, 1.3))
ordilabel(nmd1, display = "species", col = "darkred", fill = NA, border = NA, cex = 0.5)
points(nmd1, pch=c(16, 17)[as.numeric(as.factor(dat$category))], col = "darkblue") #add sampling points
legend("topright", pch = c(16, 17), c("Forest","Park"), col = "darkblue", cex = 0.7) #add legend

ef <- envfit(nmd1, dat[,32:44]) #check for correlation of dissimilarity gradients with environmental va
ef #results
```

```
##
## ***VECTORS
##
##                    NMDS1    NMDS2     r2 Pr(>r)
## canopy_cover     0.97466 -0.22368 0.5449  0.001 ***
## n_tree_spec     -0.90912  0.41653 0.5815  0.001 ***
## n_tree_ind       0.98935 -0.14553 0.1254  0.369
## dbh_min         -0.95993 -0.28023 0.0958  0.505
## dbh_mean        -0.71932 -0.69468 0.3251  0.053 .
## dbh_max         -0.75684 -0.65361 0.3293  0.056 .
## dbh_median      -0.64466 -0.76447 0.2933  0.065 .
## dbh_sd           -0.90591  0.42346 0.5773  0.003 **
## n_microhabitats   0.48732 -0.87322 0.2377  0.125
## latitude          0.66166 -0.74980 0.2294  0.155
## longitude         0.99957  0.02925 0.4695  0.008 **
## size              0.90391  0.42772 0.7395  0.001 ***
## temperature      -0.58667  0.80982 0.0484  0.682
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Permutation: free
## Number of permutations: 999
```

```
plot(ef, p.max=0.05, col = "darkblue", cex = 0.7) #add significant environmental variables to the NMDS
```



Now the species accumulation curve is shown.

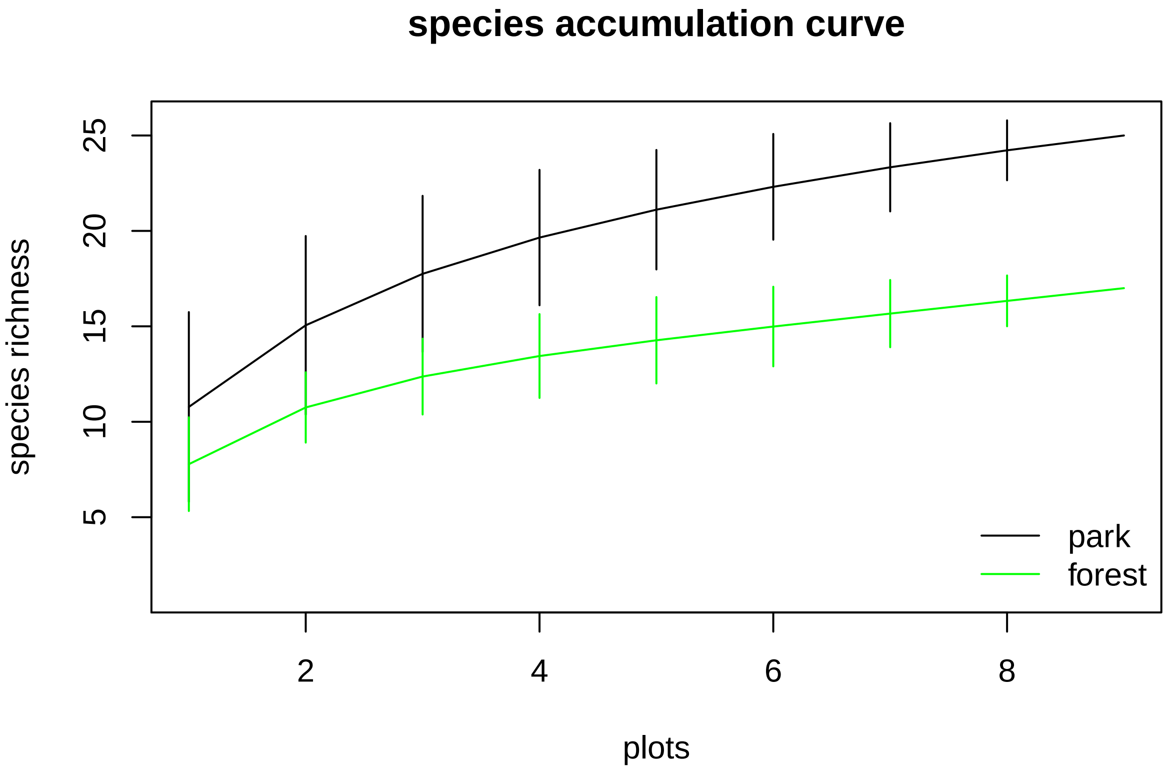
```
SAC_park <- specaccum(subset(dat[,4:31], dat$category == "park"))
```

```
## Warning in cor(x > 0): the standard deviation is zero
```

```
SAC_fore <- specaccum(subset(dat[,4:31], dat$category == "forest"))
```

```
## Warning in cor(x > 0): the standard deviation is zero
```

```
plot(SAC_park, xlab = "plots", ylab = "species richness", main="species accumulation curve")

plot(SAC_fore, xlab = "plots", ylab = "species richness", main="species accumulation curve", col="green
legend("bottomright", legend = c("park","forest"), col = c("black","green"), lwd=1, bty = "n")
```



Analysis of bird's diversity and the variables measured from these two types of ecosystems.

```
parks <- read.csv("data/parks.csv", sep=";")
forest <- read.csv("data/forest.csv", sep=";")
alpha <- specnumber(dat[,4:31]) # or use the binary site-species matrix
gamma <- ncol(dat[,colSums(dat[,4:31])>0])

##Lande's index  (beta) diversity
gamma - mean(alpha)
```

```
## [1] 37.72222
```

```
##Whittaker's index

#gamma/mean(alpha)


#For parks

alphap <- specnumber(parks[,4:31])
```

```
gammap <- ncol(parks[,colSums(parks[,4:31]) > 0])

##Lande's index
gammap - mean(alphap)
```

```
## [1] 28.22222
```

```
##Whittaker's index

#gammap/mean(alphap)

#For forest
alphaf <- specnumber(forest[,4:31])

gammaf <- ncol(parks[,colSums(parks[,4:31]) > 0])

##Lande's index
gammaf - mean(alphaf)
```

```
## [1] 31.22222
```

```
##Whittaker's index

#gammaf/mean(alphaf)
```

The number of shared and unique species for a given for the two plots combine and separated.

```
beta_virt <- betadiver(dat[,4:31], method = NA)
# a
beta_virt$a
```

```
##     1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17
## 2   7                                                
## 3   7  9                                             
## 4   4  5  4                                          
## 5   6  6  6  5                                       
## 6   5  5  7  4  5                                    
## 7   4  5  5  3  3  2                                 
## 8   5  6  7  4  4  6  4                              
## 9   5  6  6  3  4  4  4  6                           
## 10  4  5  5  4  5  5  2  5  4                        
## 11  7  8  8  6  7  6  5  7  6  7                     
## 12  5  6  7  4  6  6  3  6  5  6  7                  
## 13  4  6  7  3  4  4  3  5  5  4  5  5               
## 14  5  7  7  4  4  5  4  7  6  5  7  5  6            
## 15  5  5  6  3  4  4  2  4  4  4  5  5  5  5         
## 16  7  6  8  4  6  7  3  5  4  4  6  6  4  4  4      
## 17  9  8  9  5  6  7  5  7  7  6  9  6  5  7  6  8   
## 18  7  7  8  6  5  7  4  7  6  5  7  6  5  6  6  8 11
```

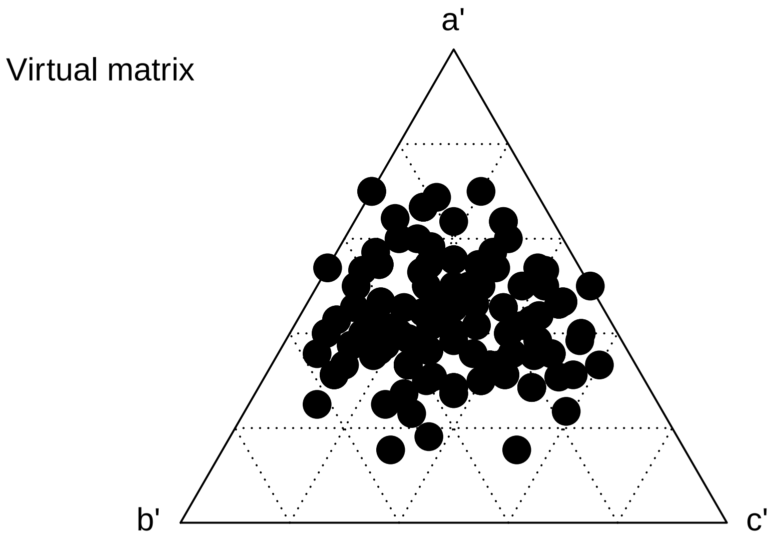
```
beta_virt$b
```

```
##     1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17
## 2   5                                                
## 3   7  5                                             
## 4   3  2  3                                          
## 5   1  1  1  2                                       
## 6   4  4  2  5  4                                    
## 7   2  1  1  3  3  4                                 
## 8   4  3  2  5  5  3  5                              
## 9   2  1  1  4  3  3  3  1                           
## 10  3  2  2  3  2  2  5  2  3                        
## 11  3  2  2  4  3  4  5  3  4  3                     
## 12  3  2  1  4  2  2  5  2  3  2  1                  
## 13  3  1  0  4  3  3  4  2  2  3  2  2               
## 14  3  1  1  4  4  3  4  1  2  3  1  3  2            
## 15  4  4  3  6  5  5  7  5  5  5  4  4  4  4         
## 16  2  3  1  5  3  2  6  4  5  5  3  3  5  5  5      
## 17  4  5  4  8  7  6  8  6  6  7  4  7  8  6  7  5   
## 18  7  7  6  8  9  7 10  7  8  9  7  8  9  8  8  6  3
```

```
beta_virt$c
```

```
##     1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17
## 2   4                                                
## 3   4  3                                             
## 4   7  7 10                                          
## 5   5  6  8  2                                       
## 6   6  7  7  3  2                                    
## 7   7  7  9  4  4  7                                 
## 8   6  6  7  3  3  3  2                              
## 9   6  6  8  4  3  5  2  3                           
## 10  7  7  9  3  2  4  4  4  3                        
## 11  4  4  6  1  0  3  1  2  1  0                     
## 12  6  6  7  3  1  3  3  3  2  1  3                  
## 13  7  6  7  4  3  5  3  4  2  3  5  3               
## 14  6  5  7  3  3  4  2  2  1  2  3  3  1            
## 15  6  7  8  4  3  5  4  5  3  3  5  3  2  3         
## 16  4  6  6  3  1  2  3  4  3  3  4  2  3  4  5      
## 17  2  4  5  2  1  2  1  2  0  1  1  2  2  1  3  1   
## 18  4  5  6  1  2  2  2  2  1  2  3  2  2  2  3  1  2
```

```
plot(betadiver(dat[,4:31], method=NA), pch = 16, cex = 2,)
legend("topleft", legend = "Virtual matrix", bty = "n")
```



```
beta_virtp <- betadiver(parks, method = NA)
# a
beta_virtp$a
```

```
##    1  2  3  4  5  6  7  8
## 2 23
## 3 23 25
## 4 20 22 23
## 5 21 23 23 22
## 6 21 21 22 21 21
## 7 23 22 24 20 20 20
## 8 25 24 25 21 23 22 24
## 9 23 23 24 21 22 22 24 27
```

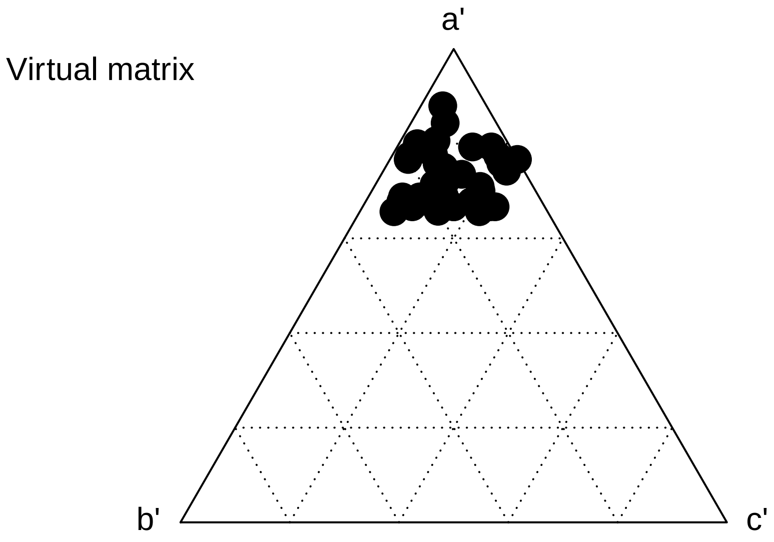
```
beta_virtp$b
```

```
##   1 2 3 4 5 6 7 8
## 2 5
## 3 7 5
## 4 3 1 0
## 5 3 1 1 2
## 6 4 4 3 4 4
## 7 2 3 1 5 5 5
## 8 4 5 4 8 6 7 5
## 9 7 7 6 9 8 8 6 3
```

```
beta_virtp$c
```

```
##   1 2 3 4 5 6 7 8
## 2 4
## 3 4 3
## 4 7 6 7
## 5 6 5 7 1
## 6 6 7 8 2 3
## 7 4 6 6 3 4 5
## 8 2 4 5 2 1 3 1
## 9 4 5 6 2 2 3 1 2
```

```
plot(betadiver(parks, method=NA), pch = 16, cex = 2,)
legend("topleft", legend = "Virtual matrix", bty = "n")
```

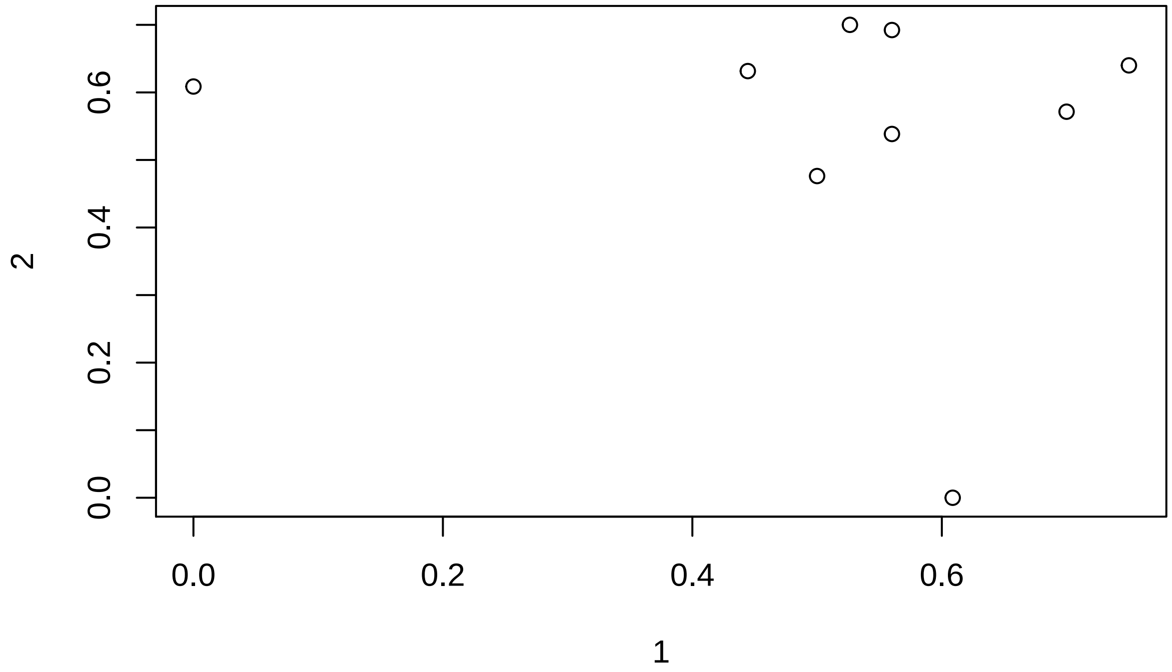


```
#vennd iagram <- as alternative to show the overlap of shared species
```

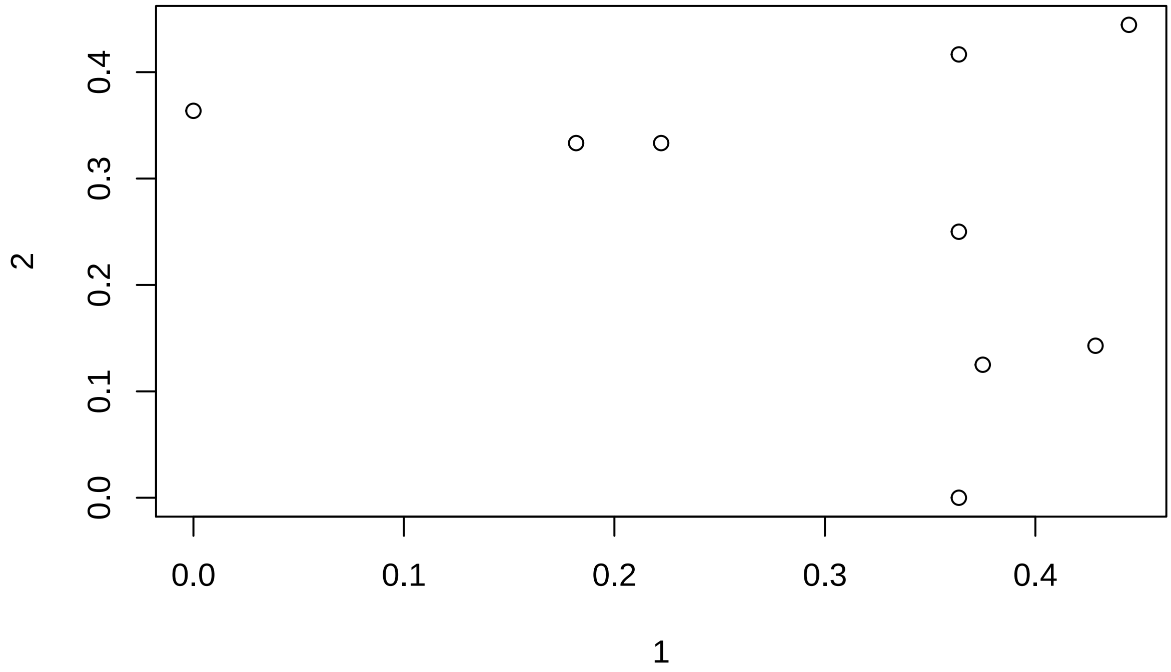
We can see how nice heterogeneity is present between the two ecosystems

Now the similarity between plots by Sorensen, Simpson and Jaccard

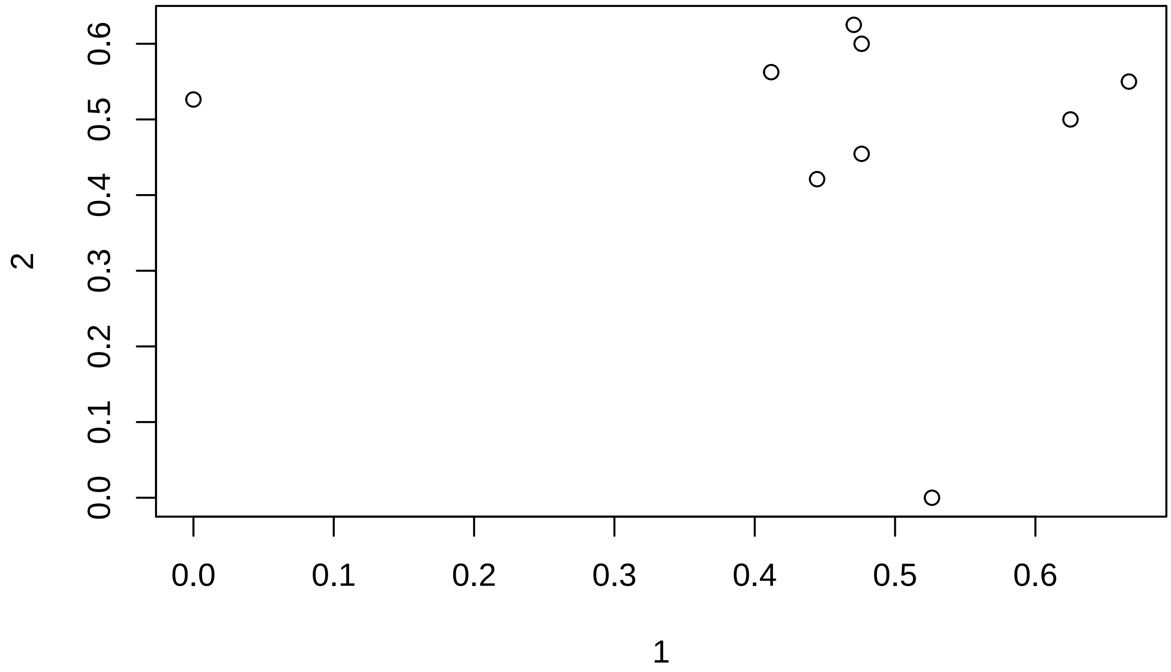
```
#Sorensen similarity
sorparks <-  betadiver(parks[,4:31], method = "sor")
sorensen <-  as.matrix(sorparks)[,]
plot(sorensen)
```



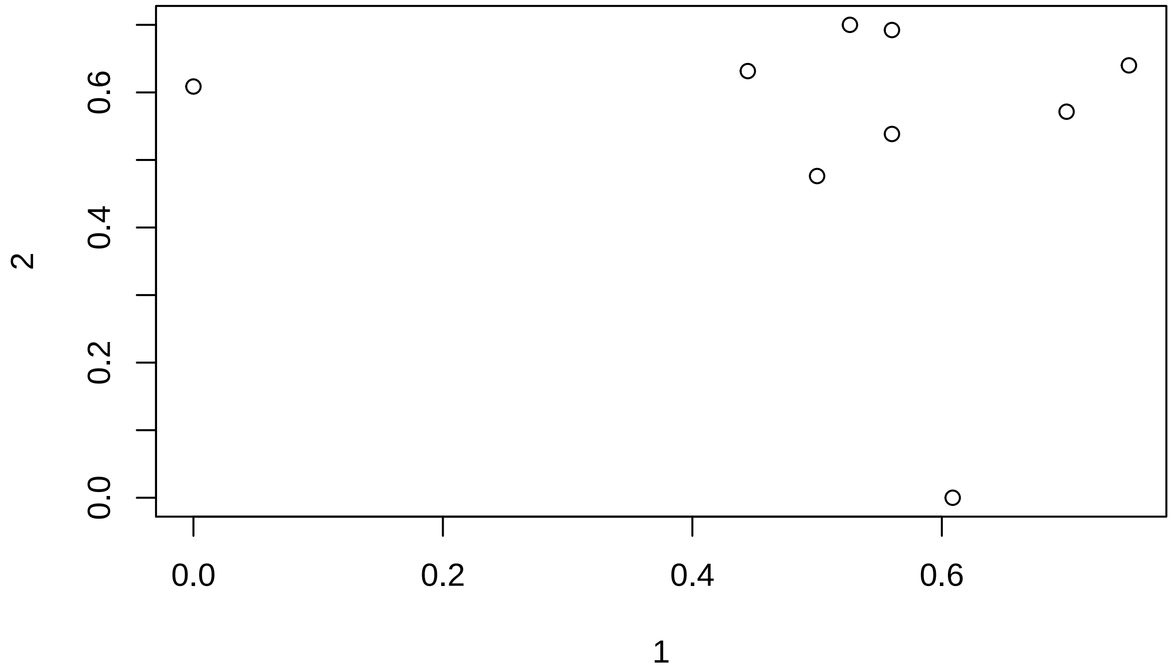
```
#Simpson similarity
simpark<- betadiver(parks[,4:31], method = "sim")
simpsonpark <- as.matrix(simpark)[,]
plot(simpsonpark)
```



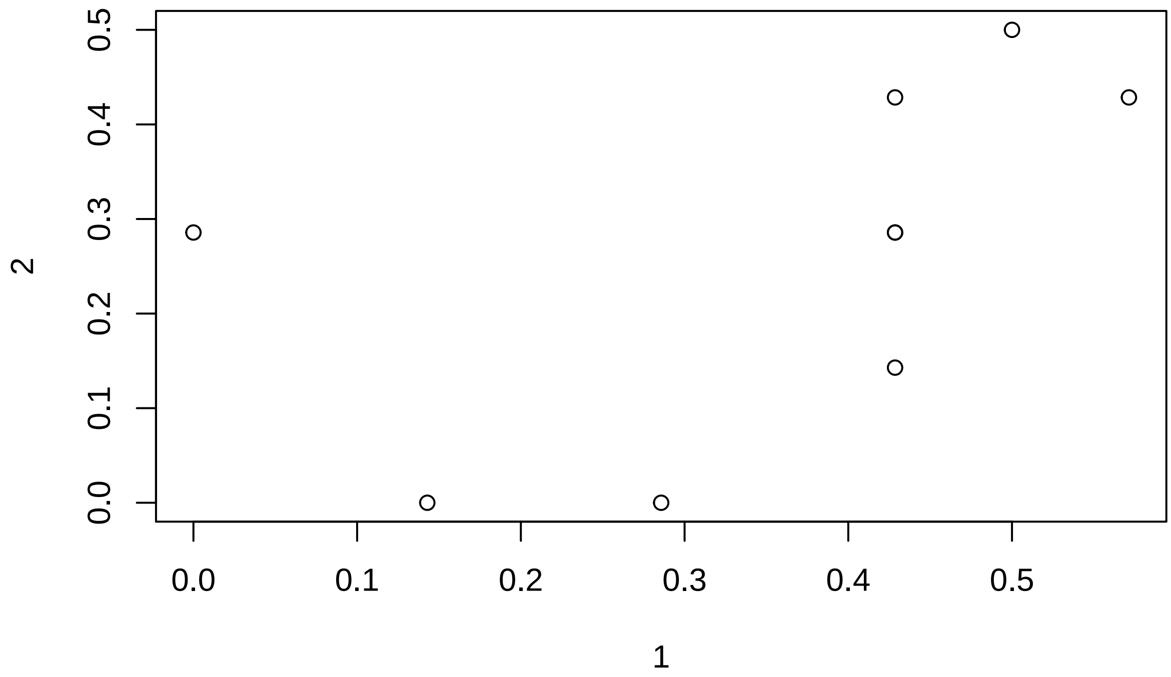
```
# Jaccard similarity
jparks <- betadiver(parks[,1:31], method = "j")
plot(as.matrix(jparks)[,])
```



```
#Sorensen similarity
sorforest <-  betadiver(forest[,4:31], method = "sor")
sorensenforest <-  as.matrix(sorforest)[,]
plot(sorensen)
```



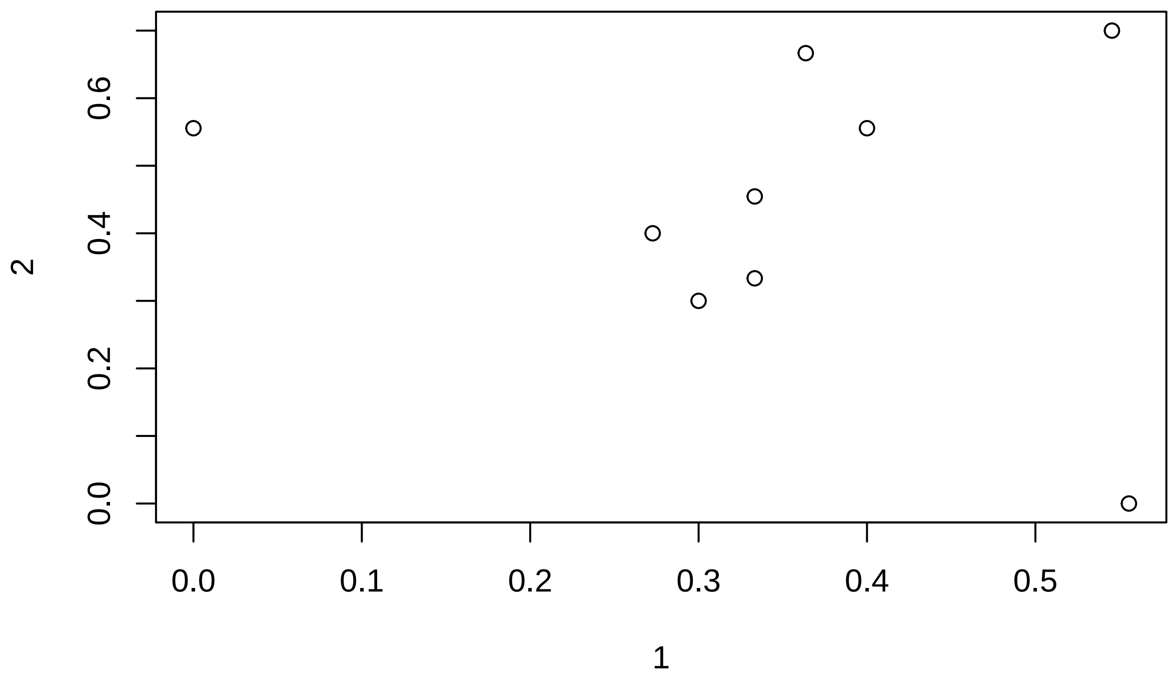
```
#Simpson similarity
simf <- betadiver(forest[,4:31], method = "sim")
simpson <- as.matrix(simf)[,]
plot(simpson)
```



```
# Jaccard similarity

jforest <- betadiver(forest[,4:31], method = "j")
as.matrix(jforest)[,]
```

```
##           1         2         3         4         5         6         7
## 1 0.0000000 0.5555556 0.3333333 0.3000000 0.3333333 0.2727273 0.4000000
## 2 0.5555556 0.0000000 0.4545455 0.3000000 0.3333333 0.4000000 0.5555556
## 3 0.3333333 0.4545455 0.0000000 0.1538462 0.5000000 0.3333333 0.4545455
## 4 0.3000000 0.3000000 0.1538462 0.0000000 0.3636364 0.4444444 0.1818182
## 5 0.3333333 0.3333333 0.5000000 0.3636364 0.0000000 0.6000000 0.4545455
## 6 0.2727273 0.4000000 0.3333333 0.4444444 0.6000000 0.0000000 0.4000000
## 7 0.4000000 0.5555556 0.4545455 0.1818182 0.4545455 0.4000000 0.0000000
## 8 0.5454545 0.7000000 0.4615385 0.4545455 0.5833333 0.5454545 0.7000000
## 9 0.3636364 0.6666667 0.5454545 0.2727273 0.5454545 0.5000000 0.6666667
##           8         9
## 1 0.5454545 0.3636364
## 2 0.7000000 0.6666667
## 3 0.4615385 0.5454545
## 4 0.4545455 0.2727273
## 5 0.5833333 0.5454545
## 6 0.5454545 0.5000000
## 7 0.7000000 0.6666667
## 8 0.0000000 0.6363636
## 9 0.6363636 0.0000000
```

```
plot(as.matrix(jforest)[,])
```



```
#calculate mean of these similarities indices and compare between sites
```

RANK abundance curve <- takes the most abundance species and plot it against species abundance rank

```
#library(wesanderson)
#library("RColorBrewer")

par(mfrow = c(1, 1))
plot1_1 <- data.frame(sp = colnames(dat[,4:31]),
                      ab = as.numeric(dat[1,4:31]))

plot1_1 <- plot1_1[which(plot1_1$ab!=0),]
dim(plot1_1)
```

```
## [1] 11  2
```

```
# Add rank of species in the first community
plot1_1$rank <- rank(-plot1_1$ab, ties.method = "random")
# Ordering data before plotting
plot1_1 <- plot1_1[order(plot1_1$rank), ]
# Plot
plot(plot1_1$rank, plot1_1$ab, type = "b",
     col = "coral", pch = 16, lwd = 1,
     main = "RAD",
     xlab = "Rank", ylab = "Abundances")



plot1_2 <- data.frame(sp = colnames(dat[,4:31]),
                       ab = as.numeric(dat[2,4:31]))


plot1_2 <- plot1_2[which(plot1_2$ab!=0),]
dim(plot1_2)
```

```
## [1] 12  2
```

```
plot1_2$rank <- rank(-plot1_2$ab, ties.method = "random")

plot1_2 <- plot1_2[order(plot1_2$rank), ]
points(plot1_2$rank, plot1_2$ab, type = "both",   col = "coral2", pch = 16)
```

```
## Warning in plot.xy(xy.coords(x, y), type = type, ...): gráfico de tipo 'both' va
## a ser truncado al primer carácter
```

```
plot1_3 <- data.frame(sp = colnames(dat[,4:31]),
                       ab = as.numeric(dat[3,4:31]))

plot1_3 <- plot1_3[which(plot1_3$ab!=0),]
dim(plot1_3)
```

```
## [1] 14  2
```

```
plot1_3$rank <- rank(-plot1_3$ab, ties.method = "random")
plot1_3 <- plot1_3[order(plot1_3$rank), ]
points(plot1_3$rank, plot1_3$ab, type = "b",
     col = "coral3", pch = 16, lwd = 1)


plot2_1 <- data.frame(sp = colnames(dat[,4:31]),
                       ab = as.numeric(dat[4,4:31]))
plot2_1 <- plot2_1[which(plot2_1$ab!=0),]
dim(plot2_1)
```

```
## [1] 7 2
```

```
plot2_1$rank <- rank(-plot2_1$ab, ties.method = "random")
plot2_1 <- plot2_1[order(plot2_1$rank), ]
points(plot2_1$rank, plot2_1$ab, type = "b",
     col = "aquamarine", pch = 16, lwd = 1)
```

```r
plot2_2 <- data.frame(sp = colnames(dat[,4:31]),
                       ab = as.numeric(dat[5,4:31]))
plot2_2 <- plot2_2[which(plot2_2$ab!=0),]
dim(plot2_2)
```

```
## [1] 7 2
```

```r
plot2_2$rank <- rank(-plot2_2$ab, ties.method = "random")
plot2_2 <- plot2_2[order(plot2_2$rank), ]
points(plot2_2$rank, plot2_2$ab, type = "b",
     col = "aquamarine3", pch = 16, lwd = 1)

plot2_3 <- data.frame(sp = colnames(dat[,4:31]),
                       ab = as.numeric(dat[6,4:31]))
plot2_3 <- plot2_3[which(plot2_3$ab!=0),]
dim(plot2_3)
```

```
## [1] 9 2
```

```r
plot2_3$rank <- rank(-plot2_3$ab, ties.method = "random")
plot2_3 <- plot2_3[order(plot2_3$rank), ]
points(plot2_3$rank, plot2_3$ab, type = "b",
     col = "aquamarine4", pch = 16, lwd = 1)

plot3_1 <- data.frame(sp = colnames(dat[,4:31]),
                       ab = as.numeric(dat[7,4:31]))
plot3_1 <- plot3_1[which(plot3_1$ab!=0),]
dim(plot3_1)
```

```
## [1] 6 2
```

```r
plot3_1$rank <- rank(-plot3_1$ab, ties.method = "random")
plot3_1 <- plot3_1[order(plot3_1$rank), ]
points(plot3_1$rank, plot3_1$ab, type = "b",
     col = "antiquewhite", pch = 16, lwd = 1)

plot3_2 <- data.frame(sp = colnames(dat[,4:31]),
                       ab = as.numeric(dat[8,4:31]))
plot3_2 <- plot3_2[which(plot3_1$ab!=0),]
dim(plot3_2)
```

```
## [1] 6 2
```

```r
plot3_2$rank <- rank(-plot3_2$ab, ties.method = "random")
plot3_2 <- plot3_2[order(plot3_2$rank), ]
points(plot3_2$rank, plot3_2$ab, type = "b",
     col = "antiquewhite3", pch = 16, lwd = 1)


plot3_3 <- data.frame(sp = colnames(dat[,4:31]),
                      ab = as.numeric(dat[9,4:31]))
plot3_3 <- plot3_3[which(plot3_1$ab!=0),]
dim(plot3_3)
```

```
## [1] 6 2
```

```
plot3_3$rank <- rank(-plot3_3$ab, ties.method = "random")
plot3_3 <- plot3_3[order(plot3_3$rank), ]
points(plot3_3$rank, plot3_3$ab, type = "b",
     col = "antiquewhite4", pch = 16, lwd = 1)


plot4_1 <- data.frame(sp = colnames(dat[,4:31]),
                      ab = as.numeric(dat[10,4:31]))
plot4_1 <- plot4_1[which(plot4_1$ab!=0),]
dim(plot4_1)
```

```
## [1] 7 2
```

```
plot4_1$rank <- rank(-plot4_1$ab, ties.method = "random")
plot4_1 <- plot4_1[order(plot4_1$rank), ]
points(plot4_1$rank, plot4_1$ab, type = "b",
     col = "brown1", pch = 16, lwd = 1)

plot4_2 <- data.frame(sp = colnames(dat[,4:31]),
                      ab = as.numeric(dat[11,4:31]))
plot4_2 <- plot4_2[which(plot4_2$ab!=0),]
dim(plot4_2)
```

```
## [1] 10  2
```

```
plot4_2$rank <- rank(-plot4_2$ab, ties.method = "random")
plot4_2 <- plot4_2[order(plot4_1$rank), ]
points(plot4_2$rank, plot4_2$ab, type = "b",
     col = "brown4", pch = 16, lwd = 1)

plot4_3 <- data.frame(sp = colnames(dat[,4:31]),
                      ab = as.numeric(dat[12,4:31]))
plot4_3 <- plot4_3[which(plot4_3$ab!=0),]
dim(plot4_3)
```

```
## [1] 8 2
```

```
plot4_3$rank <- rank(-plot4_3$ab, ties.method = "random")
plot4_3 <- plot4_3[order(plot4_3$rank), ]
points(plot4_3$rank, plot4_3$ab, type = "b",
     col = "brown2", pch = 16, lwd = 1)


plot5_1 <- data.frame(sp = colnames(dat[,4:31]),
                      ab = as.numeric(dat[13,4:31]))
plot5_1 <- plot5_1[which(plot5_1$ab!=0),]
dim(plot5_1)
```

```
## [1] 7 2
```

```
plot5_1$rank <- rank(-plot5_1$ab, ties.method = "random")
plot5_1 <- plot5_1[order(plot5_1$rank), ]
points(plot5_1$rank, plot5_1$ab, type = "b",
     col = "chartreuse", pch = 16, lwd = 1)


plot5_2 <- data.frame(sp = colnames(dat[,4:31]),
                      ab = as.numeric(dat[14,4:31]))
plot5_2 <- plot5_2[which(plot5_2$ab!=0),]
dim(plot5_2)
```

```
## [1] 8 2
```

```
plot5_2$rank <- rank(-plot5_2$ab, ties.method = "random")
plot5_2 <- plot5_2[order(plot5_2$rank), ]
points(plot5_2$rank, plot5_2$ab, type = "b",
     col = "chartreuse3", pch = 16, lwd = 1)

plot5_3 <- data.frame(sp = colnames(dat[,4:31]),
                      ab = as.numeric(dat[15,4:31]))
plot5_3 <- plot5_3[which(plot5_3$ab!=0),]
dim(plot5_3)
```

```
## [1] 9 2
```

```
plot5_3$rank <- rank(-plot5_3$ab, ties.method = "random")
plot5_3 <- plot5_3[order(plot5_3$rank), ]
points(plot5_3$rank, plot5_3$ab, type = "b",
     col = "chartreuse4", pch = 16, lwd = 1)


plot6_1 <- data.frame(sp = colnames(dat[,4:31]),
                      ab = as.numeric(dat[16,4:31]))
plot6_1 <- plot6_1[which(plot6_1$ab!=0),]
dim(plot6_1)
```

```
## [1] 9 2
```

```
plot6_1$rank <- rank(-plot6_1$ab, ties.method = "random")
plot6_1 <- plot6_1[order(plot6_1$rank), ]
points(plot6_1$rank, plot6_1$ab, type = "b",
     col = "cadetblue", pch = 16, lwd = 1)
```

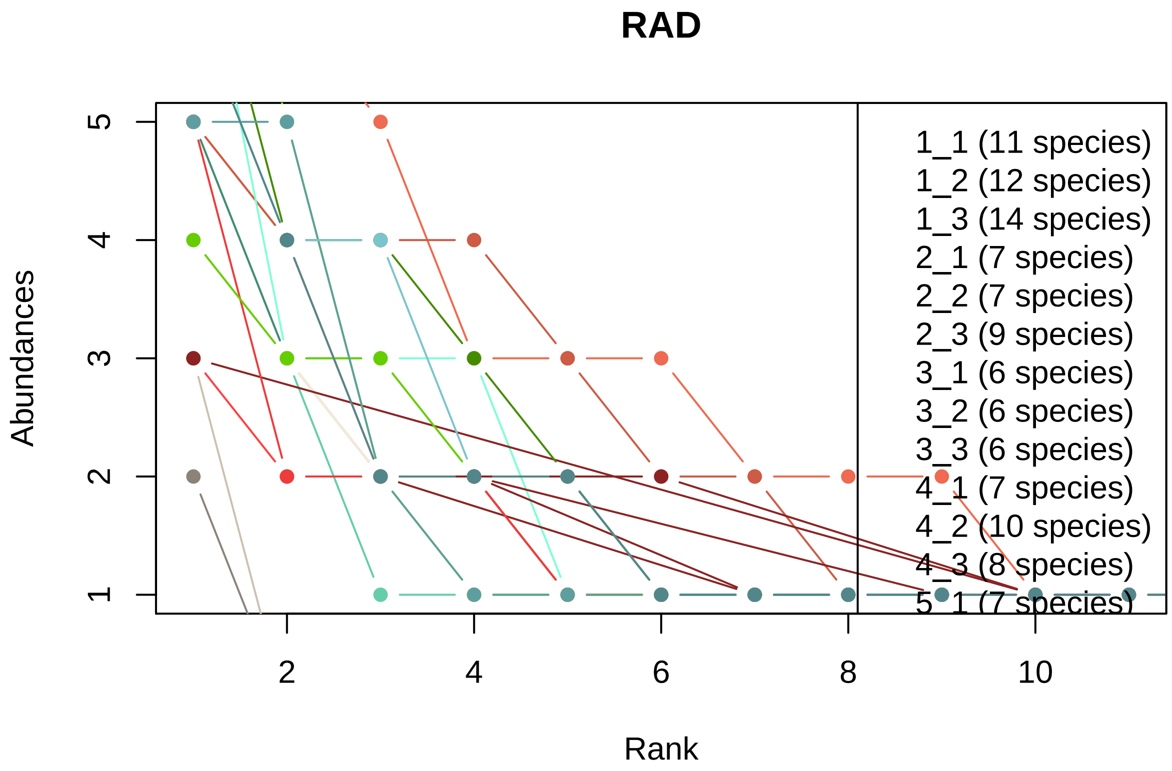
```
plot6_2 <- data.frame(sp = colnames(dat[,4:31]),
                       ab = as.numeric(dat[17,4:31]))
plot6_2 <- plot6_2[which(plot6_2$ab!=0),]
dim(plot6_2)
```

```
## [1] 13  2
```

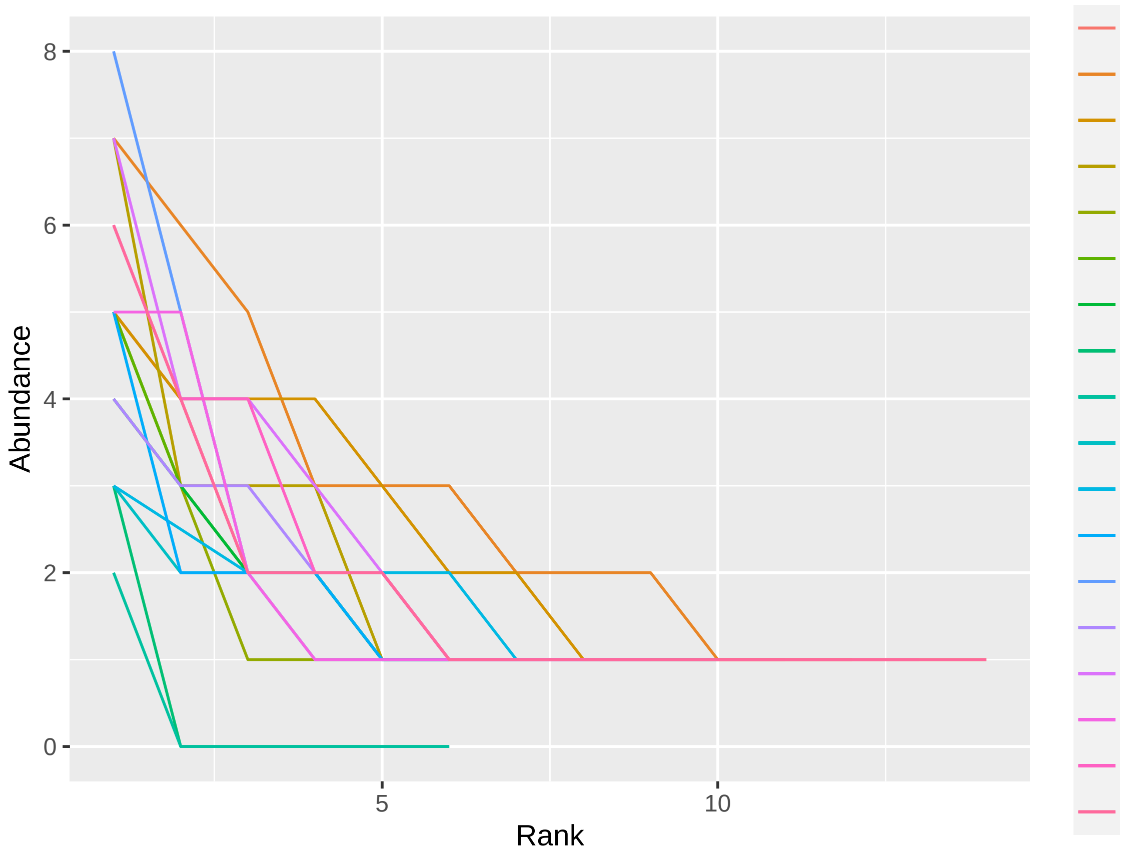
```
plot6_2$rank <- rank(-plot6_2$ab, ties.method = "random")
plot6_2 <- plot6_2[order(plot6_2$rank), ]
points(plot6_2$rank, plot6_2$ab, type = "b",
     col = "cadetblue3", pch = 16, lwd = 1)


plot6_3 <- data.frame(sp = colnames(dat[,4:31]),
                       ab = as.numeric(dat[18,4:31]))
plot6_3 <- plot6_3[which(plot6_3$ab!=0),]
dim(plot6_3)
```

```
## [1] 14  2
```

```
plot6_3$rank <- rank(-plot6_3$ab, ties.method = "random")
plot6_3 <- plot6_3[order(plot6_3$rank), ]
points(plot6_3$rank, plot6_3$ab, type = "b",
     col = "cadetblue4", pch = 16, lwd = 1)


legend( "topright",legend=c("1_1 (11 species)", "1_2 (12 species)", "1_3 (14 species)", "2_1 (7 species
```



```
datplot <- bind_rows(plot1_1,plot1_2,plot1_3)
plot6_2$plot <- "plot6_2"

#plot1 <- cbind(plot1_1[,1:3], plot1_2[,1:3], plot1_3[,1:3])


ggplot()+
  geom_line(aes(rank, ab, colour="plot1_1"), data=plot1_1)+
geom_line(aes(rank, ab, colour="plot1_2"), data=plot1_2)+
geom_line(aes(rank, ab, colour="plot1_3"), data=plot1_3)+
geom_line(aes(rank, ab, colour="plot2_1"), data=plot2_1)+
geom_line(aes(rank, ab, colour="plot2_2"), data=plot2_2)+
geom_line(aes(rank, ab, colour="plot2_3"), data=plot2_3)+
geom_line(aes(rank, ab, colour="plot3_1"), data=plot3_1)+
geom_line(aes(rank, ab, colour="plot3_2"), data=plot3_2)+
geom_line(aes(rank, ab, colour="plot3_3"), data=plot3_3)+
geom_line(aes(rank, ab, colour="plot4_1"), data=plot4_1)+
geom_line(aes(rank, ab, colour="plot4_2"), data=plot4_2)+
geom_line(aes(rank, ab, colour="plot4_3"), data=plot4_3)+
geom_line(aes(rank, ab, colour="plot5_1"), data=plot5_1)+
geom_line(aes(rank, ab, colour="plot5_2"), data=plot5_2)+
geom_line(aes(rank, ab, colour="plot5_3"), data=plot5_3)+
geom_line(aes(rank, ab, colour="plot6_1"), data=plot6_1)+
geom_line(aes(rank, ab, colour="plot6_2"), data=plot6_2)+
geom_line(aes(rank, ab, colour="plot6_3"), data=plot6_3)+
  labs(y = "Abundance", x = "Rank", color = "")
```



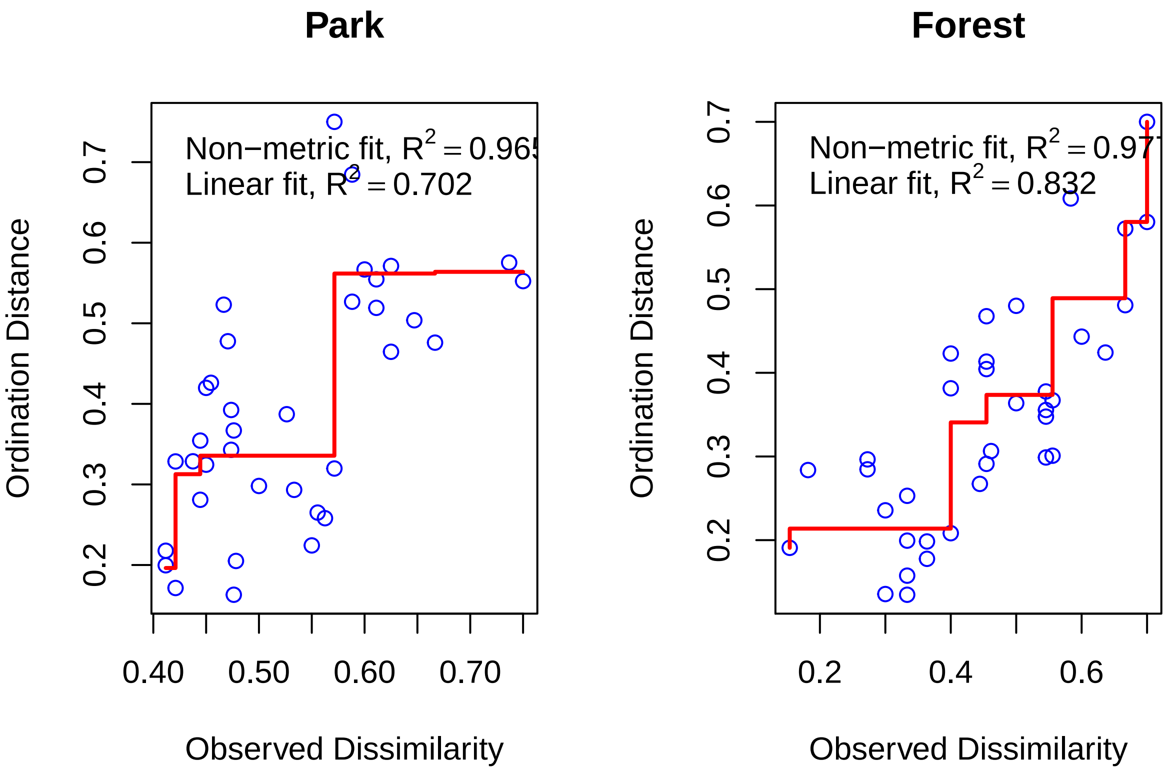
```
datp_NMDS <- metaMDS(jparks, k = 2)
```

```
## Run 0 stress 0.1880834
## Run 1 stress 0.2328249
## Run 2 stress 0.2193518
## Run 3 stress 0.2209639
## Run 4 stress 0.1958146
## Run 5 stress 0.2209639
## Run 6 stress 0.2411254
## Run 7 stress 0.1892095
## Run 8 stress 0.2271776
## Run 9 stress 0.259949
## Run 10 stress 0.2318105
## Run 11 stress 0.2292171
## Run 12 stress 0.2177675
## Run 13 stress 0.1892095
## Run 14 stress 0.2328249
## Run 15 stress 0.2496823
## Run 16 stress 0.1892095
## Run 17 stress 0.2273543
## Run 18 stress 0.2193518
## Run 19 stress 0.2469389
## Run 20 stress 0.2387428
## *** No convergence -- monoMDS stopping criteria:
##     20: stress ratio > sratmax
```

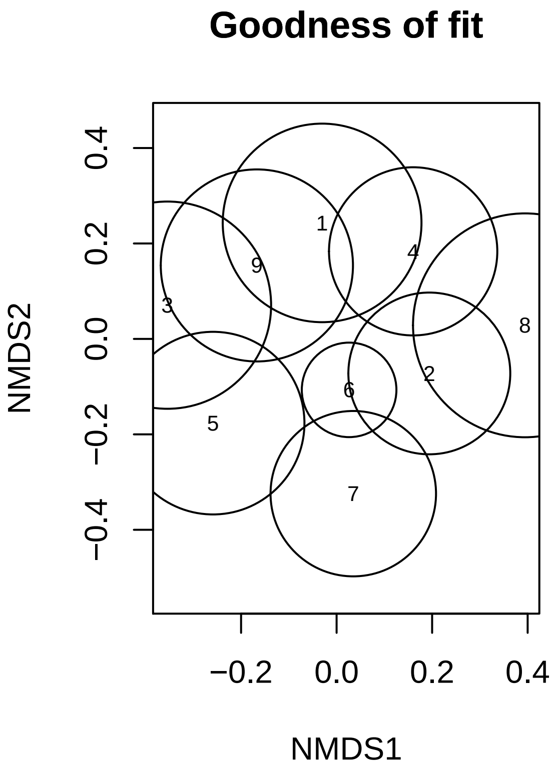
```r
datf_NMDS <- metaMDS(jforest, k=2)
```

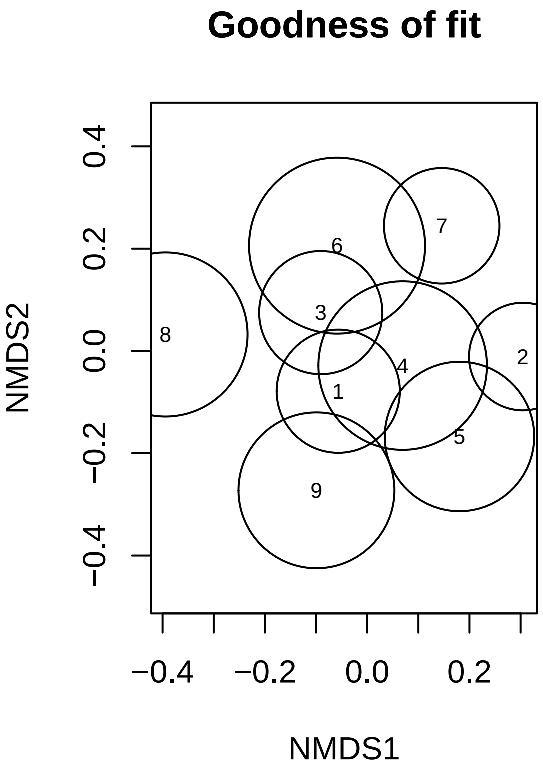
```
## Run 0 stress 0.174371
## Run 1 stress 0.1996688
## Run 2 stress 0.2466973
## Run 3 stress 0.1746113
## ... Procrustes: rmse 0.1819392  max resid 0.3055167
## Run 4 stress 0.1707359
## ... New best solution
## ... Procrustes: rmse 0.1487575  max resid 0.2460855
## Run 5 stress 0.1506224
## ... New best solution
## ... Procrustes: rmse 0.1694343  max resid 0.3442798
## Run 6 stress 0.2140076
## Run 7 stress 0.1783598
## Run 8 stress 0.2006613
## Run 9 stress 0.1506225
## ... Procrustes: rmse 0.00011449  max resid 0.0002215034
## ... Similar to previous best
## Run 10 stress 0.1794955
## Run 11 stress 0.1627734
## Run 12 stress 0.1587959
## Run 13 stress 0.1794955
## Run 14 stress 0.2145148
## Run 15 stress 0.212146
## Run 16 stress 0.2154229
## Run 17 stress 0.1644235
## Run 18 stress 0.1587959
## Run 19 stress 0.1738179
## Run 20 stress 0.1627734
## *** Solution reached
```

```r
par(mfrow = c(1, 2))
stressplot(datp_NMDS, main = "Park")
stressplot(datf_NMDS, main = "Forest")
```

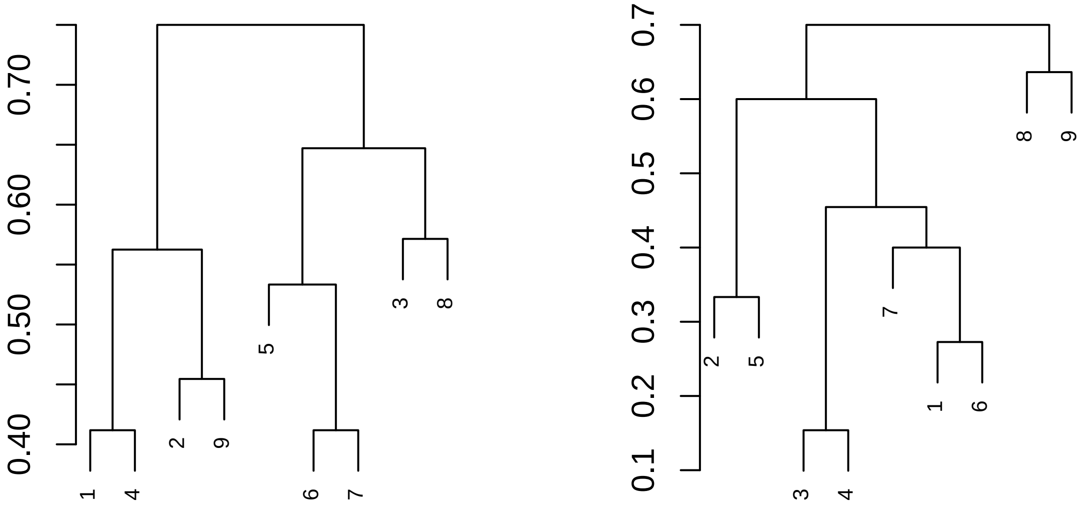


```
par(mfrow = c(1, 2))
plot(datp_NMDS, display = "sites", type = "t", main = "Goodness of fit")
points(datp_NMDS, display = "sites", cex = goodness(datp_NMDS)*200)
title(main = "Goodness of fit")

plot(datf_NMDS, display = "sites", type = "t", main = "Goodness of fit")
points(datf_NMDS, display = "sites", cex = goodness(datf_NMDS)*200)
title(main = "Goodness of fit")
```

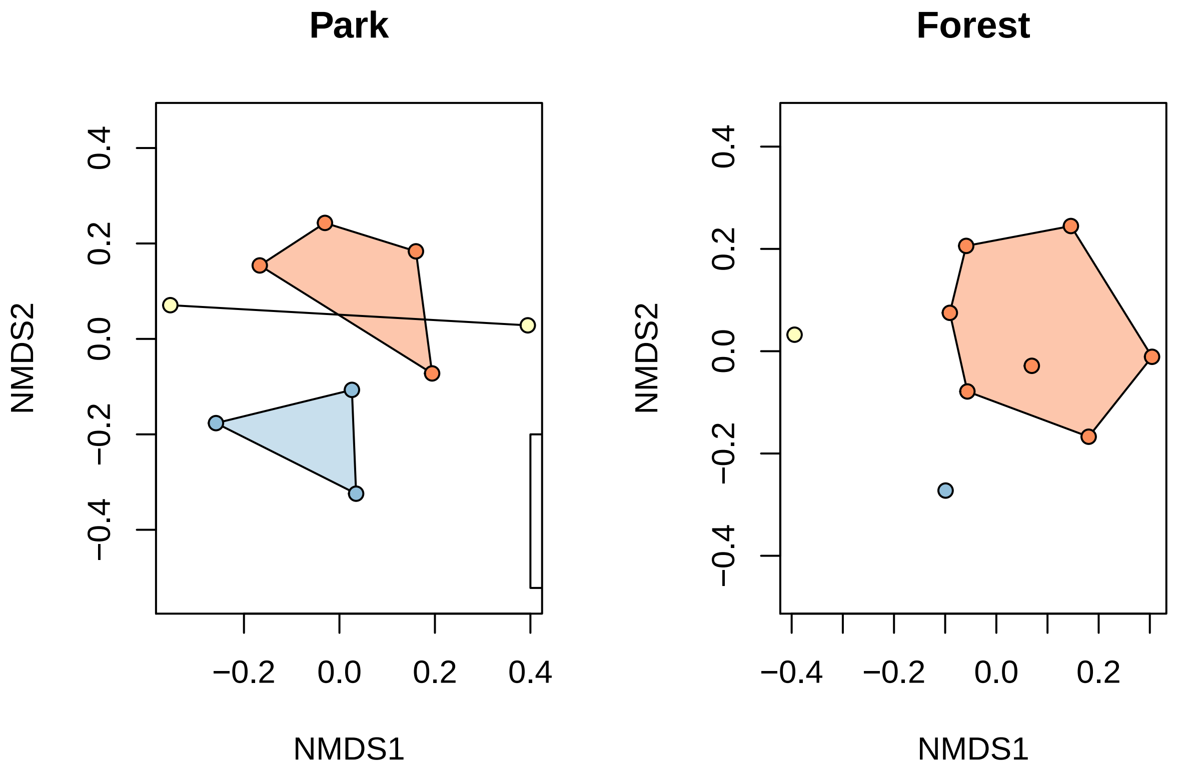




```
datpclust <- hclust(jparks, method = "complete")
plot(datpclust, xlab = "", cex = 0.7, ann = FALSE)
abline(h = 0.78, lwd = 2, col = "black", lty = 2)

datfclust <- hclust(jforest, method = "complete")
plot(datfclust, xlab = "", cex = 0.7, ann = FALSE)
abline(h = 0.78, lwd = 2, col = "black", lty = 2)
```



```
datpclust_3 <- cutree(datpclust, k = 3)

plot(datp_NMDS, display = "sites", type = "n", main = "Park")
ordihull(datp_NMDS, groups = datpclust_3,
         draw = "polygon", col = c("#fc8d59", "#ffffbf", "#91bfdb"),
         label = FALSE)
points(datp_NMDS, display = "sites", pch = 21, col = "black",
       bg = c("#fc8d59", "#ffffbf", "#91bfdb")[datpclust_3])
legend(x = 0.4, y = -0.2, legend = c(paste0("Cl. ", seq(1:3))),
       col = c("#fc8d59", "#ffffbf", "#91bfdb"), pch = 16)

datfclust_3 <- cutree(datfclust, k = 3)

plot(datf_NMDS, display = "sites", type = "n", main = "Forest")
ordihull(datf_NMDS, groups = datfclust_3,
         draw = "polygon", col = c("#fc8d59", "#ffffbf", "#91bfdb"),
         label = FALSE)
points(datf_NMDS, display = "sites", pch = 21, col = "black",
       bg = c("#fc8d59", "#ffffbf", "#91bfdb")[datfclust_3])
legend(x = 0.4, y = -0.2, legend = c(paste0("Cl. ", seq(1:3))),
       col = c("#fc8d59", "#ffffbf", "#91bfdb"), pch = 16)
```


Park
Forest
NMDS2
NMDS1
0.4
0.2
0.0
−0.2
−0.4
−0.2
0.0
0.2
0.4
−0.4
−0.2
0.0
0.2