Seed Species Background Probability

Marco Baldo

2024-05-03

## Seed Species Background Probability - iLand BDV plot experiment

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

Here we will develop the workflow for the seed species background probability in the plot bdv experiment became within the Bottoms Up Cost Action project consorsium. We used the Czech Republic dataset JH1 that it is one of the most robust and rich in terms of data, dataset. Here we have forest structure metrics, plot/site information (i.g management history, age, bdv metrics, etc) and multi-taxonomic species informations (alpha diversity and species names without the abundance just occurrences). Particularly important and relevent for this study are Fungi, Lichens, Bryophytes, Birds, Saproxilic Beetles and Moths.

# externalSeedSpecies  
sp <- c("abal, acpl, acps, bepe, fasy, frex, lade, piab, pisy, potr, quro, soau, tico, ulgl")  
  
sp

## [1] "abal, acpl, acps, bepe, fasy, frex, lade, piab, pisy, potr, quro, soau, tico, ulgl"

## The script for the species proportion

library(ggplot2)  
library(GGally)

## Registered S3 method overwritten by 'GGally':  
## method from   
## +.gg ggplot2

library(cowplot)  
library(corrplot)

## corrplot 0.92 loaded

library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(fields)

## Loading required package: spam

## Spam version 2.10-0 (2023-10-23) is loaded.  
## Type 'help( Spam)' or 'demo( spam)' for a short introduction   
## and overview of this package.  
## Help for individual functions is also obtained by adding the  
## suffix '.spam' to the function name, e.g. 'help( chol.spam)'.

##   
## Attaching package: 'spam'

## The following objects are masked from 'package:base':  
##   
## backsolve, forwardsolve

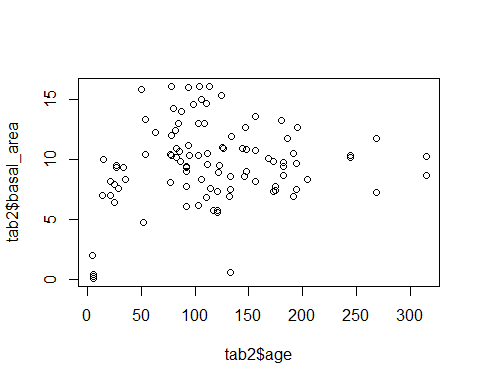
## Loading required package: viridisLite

##   
## Try help(fields) to get started.

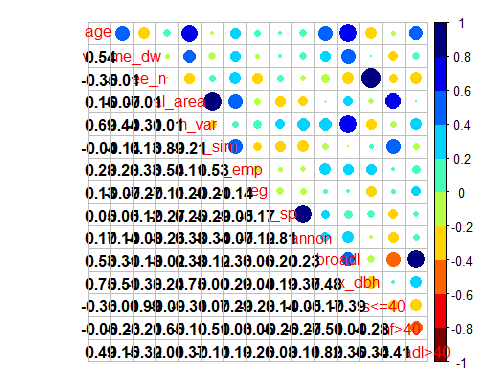
#install.packages("readxl")  
library(readxl)  
  
tab1 <- read\_xlsx("C:/iLand/2023/20230901\_Bottoms\_Up/Sources\_bottoms\_up/Jenik/final\_table\_imp/Bdv\_predictors\_clean\_correlation.xlsx")   
  
tab\_brow <-read.csv("I:/iLand/2022/20220604\_browsing\_first/variables.DB\_20220728.csv")  
  
tab2 <- read\_xlsx("C:/iLand/2023/20230901\_Bottoms\_Up/Sources\_bottoms\_up/Jenik/final\_table\_imp/Bdv\_predictors\_table\_final\_20231002.xlsx")  
  
#head(tab1)  
head(tab2)

## # A tibble: 6 × 18  
## plotID domin\_sp age volume\_dw tree\_n basal\_area dbh\_var lai\_sim lai\_emp  
## <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 L1\_03 larch 186 2.51 114 11.7 531. 3.36 3.35  
## 2 L1\_07 spruce 111 4.66 213 9.54 230. 4.07 2.22  
## 3 L1\_10 spruce 94 2.01 122 11.2 50.7 4.80 2.08  
## 4 L1\_13 beech 126 0.978 73 10.9 98.7 4.17 2.68  
## 5 L1\_17 EU 103 5.38 202 6.17 147. 1.92 2.43  
## 6 L1\_18 spruce 148 12.2 97 10.8 385. 3.88 2.28  
## # ℹ 9 more variables: reg <dbl>, n\_sp <dbl>, shannon <dbl>, ba\_broadl <dbl>,  
## # max\_dbh <dbl>, `trees<=40` <dbl>, `conf>40` <dbl>, `broadl>40` <dbl>,  
## # `ba\_broadl>40` <dbl>

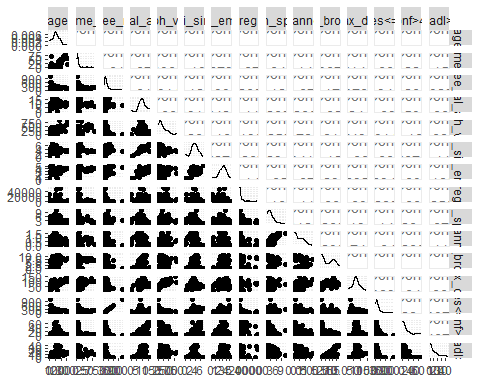
plot(tab2$age, tab2$basal\_area)



col4 <- colorRampPalette(c("#7F0000", "red", "#FF7F00", "yellow", "#7FFF7F", "cyan", "#007FFF", "blue", "#00007F"))  
  
  
#---------------------------------- just do the correlation plot all together  
a.num<-tab2[,3:17]  
  
# Look them all:  
par(mfrow = c(1, 1), pty="m", mar=c(3,3,3,3), oma=c(0,0,0,0))  
corrplot.mixed(cor(a.num),upper.col = col4(10),lower.col = "black", mar=c(0,0,0,0), tl.pos = "d")#, diag = "l")



#just do the correlation plot with the selected variables  
  
ggpairs(a.num)



# Second part - Calculate Shannon of every plot based on the species basal area  
  
library(vegan)

## Loading required package: permute

## Loading required package: lattice

## This is vegan 2.6-4

# Calculate the basal area per every tree based on their dbh  
  
raw\_data <- "C:/iLand/2023/20230901\_Bottoms\_Up/Sources\_bottoms\_up/Jenik/final\_table\_imp/Raw\_data\_structure\_CZ\_JH1\_final.xlsx"  
  
excel\_sheets(raw\_data)

## [1] "legend" "live\_dead\_trees"   
## [3] "standing\_lying\_deadwood"

raw\_data\_sp <- "live\_dead\_trees"  
  
data <- read\_excel(raw\_data,raw\_data\_sp, col\_names=T)

## New names:  
## • `` -> `...30`  
## • `` -> `...31`  
## • `` -> `...32`

new\_data <- data %>%   
 mutate(basal\_area = pi \* (treedb / 200)^2)  
  
print(new\_data)

## # A tibble: 15,901 × 33  
## siteID standID plotID treeID genus species treesp treedb treeht treevol  
## <chr> <chr> <chr> <chr> <chr> <chr> <chr> <dbl> <dbl> <chr>   
## 1 CZ\_JH1\_L1 C L1\_10 110S1 Picea abies Picea a… 44.2 26.8 1.7168…  
## 2 CZ\_JH1\_L1 C L1\_10 110S10 Picea abies Picea a… 35.5 27.3 1.1855…  
## 3 CZ\_JH1\_L1 C L1\_10 110S100 Picea abies Picea a… 32.2 25.8 0.9345…  
## 4 CZ\_JH1\_L1 C L1\_10 110S101 Picea abies Picea a… 47.5 31.3 2.3280…  
## 5 CZ\_JH1\_L1 C L1\_10 110S102 Picea abies Picea a… 41.5 30.7 1.7892…  
## 6 CZ\_JH1\_L1 C L1\_10 110S103 Picea abies Picea a… 35.5 30.4 1.3391…  
## 7 CZ\_JH1\_L1 C L1\_10 110S104 Picea abies Picea a… 29.5 28.1 0.8809…  
## 8 CZ\_JH1\_L1 C L1\_10 110S105 Picea abies Picea a… 31.2 28.2 0.9772…  
## 9 CZ\_JH1\_L1 C L1\_10 110S106 Picea abies Picea a… 28.5 27.9 0.8217…  
## 10 CZ\_JH1\_L1 C L1\_10 110S107 Picea abies Picea a… 28 29 0.8320…  
## # ℹ 15,891 more rows  
## # ℹ 23 more variables: cenver <chr>, azimuth <chr>, distan <chr>, coordx <chr>,  
## # coordy <chr>, crowndep <dbl>, crownrad1 <chr>, crownrad2 <chr>,  
## # crownrad3 <chr>, crownrad4 <chr>, treevit <dbl>, alive <dbl>, decsta <dbl>,  
## # plosiz <dbl>, weisiz <dbl>, TreMn1 <chr>, TreMn2 <chr>, TreMn3 <chr>,  
## # .. <lgl>, ...30 <lgl>, ...31 <lgl>, ...32 <dbl>, basal\_area <dbl>

# write.csv(new\_data, "C:/iLand/2023/plot\_bottoms\_up/Jenik/Raw\_data\_with\_basal\_area.csv")  
  
# Visualize the data in a single plot  
  
chosen\_plot\_id <- "L5\_37"  
  
# Filter data for the chosen plot  
filtered\_data <- new\_data %>%  
 filter(plotID == chosen\_plot\_id)  
  
print(filtered\_data)

## # A tibble: 17 × 33  
## siteID standID plotID treeID genus species treesp treedb treeht treevol  
## <chr> <chr> <chr> <chr> <chr> <chr> <chr> <dbl> <dbl> <chr>   
## 1 CZ\_JH1\_L5 E L5\_37 537S10 Quercus petraea Querc… 6.25 5.8 NA   
## 2 CZ\_JH1\_L5 E L5\_37 537S11 Quercus petraea Querc… 7.25 6.8 7.6102…  
## 3 CZ\_JH1\_L5 E L5\_37 537S12 Quercus petraea Querc… 9.5 7.5 2.0925…  
## 4 CZ\_JH1\_L5 E L5\_37 537S14 Quercus petraea Querc… 7.5 6 8.4461…  
## 5 CZ\_JH1\_L5 E L5\_37 537S15 Quercus petraea Querc… 6 6.8 NA   
## 6 CZ\_JH1\_L5 E L5\_37 537S16 Quercus petraea Querc… 8 7.9 1.2587…  
## 7 CZ\_JH1\_L5 E L5\_37 537S17 Quercus petraea Querc… 6.25 6.5 NA   
## 8 CZ\_JH1\_L5 E L5\_37 537S18 Quercus petraea Querc… 6.75 7.4 NA   
## 9 CZ\_JH1\_L5 E L5\_37 537S19 Quercus petraea Querc… 7.25 7.2 7.8896…  
## 10 CZ\_JH1\_L5 E L5\_37 537S20 Quercus petraea Querc… 5.25 6.1 NA   
## 11 CZ\_JH1\_L5 E L5\_37 537S21 Quercus petraea Querc… 6.25 6.1 NA   
## 12 CZ\_JH1\_L5 E L5\_37 537S22 Quercus petraea Querc… 5.75 6.2 NA   
## 13 CZ\_JH1\_L5 E L5\_37 537S23 Quercus petraea Querc… 5.75 5.7 NA   
## 14 CZ\_JH1\_L5 E L5\_37 537S24 Quercus petraea Querc… 6.25 7.3 NA   
## 15 CZ\_JH1\_L5 E L5\_37 537S25 Quercus petraea Querc… 5.5 7.1 NA   
## 16 CZ\_JH1\_L5 E L5\_37 537S26 Pinus sylvest… Pinus… 6 3.5 NA   
## 17 CZ\_JH1\_L5 E L5\_37 537S9 Quercus petraea Querc… 7 6.5 6.6441…  
## # ℹ 23 more variables: cenver <chr>, azimuth <chr>, distan <chr>, coordx <chr>,  
## # coordy <chr>, crowndep <dbl>, crownrad1 <chr>, crownrad2 <chr>,  
## # crownrad3 <chr>, crownrad4 <chr>, treevit <dbl>, alive <dbl>, decsta <dbl>,  
## # plosiz <dbl>, weisiz <dbl>, TreMn1 <chr>, TreMn2 <chr>, TreMn3 <chr>,  
## # .. <lgl>, ...30 <lgl>, ...31 <lgl>, ...32 <dbl>, basal\_area <dbl>

#-------------------------------------------------------------------------------  
# Sum BA for every species in every plot  
  
# Group by plotID and species, then calculate the sum of basal areas  
summed\_tree\_areas <- new\_data %>%  
 group\_by(plotID, treesp) %>%  
 summarize(total\_basal\_area = sum(basal\_area))

## `summarise()` has grouped output by 'plotID'. You can override using the  
## `.groups` argument.

# Print the resulting dataframe  
print(summed\_tree\_areas)

## # A tibble: 301 × 3  
## # Groups: plotID [99]  
## plotID treesp total\_basal\_area  
## <chr> <chr> <dbl>  
## 1 L1\_03 Acer pseudoplatanus 0.00785  
## 2 L1\_03 Fagus sylvatica 6.04   
## 3 L1\_03 Larix decidua 5.62   
## 4 L1\_03 Populus tremula 0.00950  
## 5 L1\_03 Quercus robur 0.0104   
## 6 L1\_03 Sorbus aucuparia 0.0495   
## 7 L1\_07 Betula pendula 0.00650  
## 8 L1\_07 Fagus sylvatica 0.0113   
## 9 L1\_07 Picea abies 9.49   
## 10 L1\_07 Quercus robur 0.103   
## # ℹ 291 more rows

unique\_plots <- unique(summed\_tree\_areas$plotID) # alternative unique\_plots <- unique(CZ\_JH1[,"plotID"])  
print(unique\_plots) # 99 plots

## [1] "L1\_03" "L1\_07" "L1\_10" "L1\_13" "L1\_17" "L1\_18" "L1\_22" "L1\_24" "L1\_26"  
## [10] "L1\_27" "L1\_31" "L1\_33" "L1\_34" "L1\_36" "L1\_38" "L1\_43" "L1\_44" "L1\_48"  
## [19] "L2\_01" "L2\_05" "L2\_06" "L2\_09" "L2\_13" "L2\_14" "L2\_16" "L2\_17" "L2\_18"  
## [28] "L2\_20" "L2\_26" "L2\_27" "L2\_30" "L2\_32" "L2\_33" "L2\_34" "L2\_38" "L2\_44"  
## [37] "L3\_02" "L3\_06" "L3\_09" "L3\_10" "L3\_12" "L3\_16" "L3\_18" "L3\_23" "L3\_24"  
## [46] "L3\_26" "L3\_27" "L3\_31" "L3\_32" "L3\_34" "L3\_37" "L3\_38" "L3\_41" "L3\_42"  
## [55] "L4\_02" "L4\_04" "L4\_06" "L4\_09" "L4\_10" "L4\_11" "L4\_16" "L4\_17" "L4\_18"  
## [64] "L4\_20" "L4\_23" "L4\_25" "L4\_26" "L4\_27" "L4\_30" "L4\_31" "L4\_32" "L4\_33"  
## [73] "L4\_39" "L4\_42" "L5\_01" "L5\_05" "L5\_12" "L5\_18" "L5\_19" "L5\_21" "L5\_25"  
## [82] "L5\_28" "L5\_32" "L5\_33" "L5\_37" "L5\_38" "L6\_01" "L6\_02" "L6\_03" "L6\_04"  
## [91] "L6\_05" "L6\_08" "L6\_10" "L6\_11" "L6\_14" "L6\_15" "L6\_17" "L6\_19" "L6\_21"

#write.csv(summed\_tree\_areas,"C:/iLand/2023/plot\_bottoms\_up/Jenik/summed\_tree\_areas.csv")  
  
# Calculate the Shannon diversity index  
shannon\_index <- summed\_tree\_areas %>%  
 group\_by(plotID) %>%  
 summarize(shannon = diversity(total\_basal\_area, base = exp(1)))  
  
# Print the resulting dataframe  
print(shannon\_index)

## # A tibble: 99 × 2  
## plotID shannon  
## <chr> <dbl>  
## 1 L1\_03 0.734   
## 2 L1\_07 0.0741  
## 3 L1\_10 0.281   
## 4 L1\_13 0   
## 5 L1\_17 1.23   
## 6 L1\_18 1.08   
## 7 L1\_22 0.691   
## 8 L1\_24 0.535   
## 9 L1\_26 0.952   
## 10 L1\_27 0.0796  
## # ℹ 89 more rows

#write.csv(shannon\_index,"C:/iLand/2023/plot\_bottoms\_up/Jenik/shannon\_index\_ba.csv")  
  
#-------------------------------------------------------------------------------  
# Calculate the number of trees between 10 and 40 cm  
library(readxl)  
library(writexl)  
  
# Specify the range for DBH  
dbh\_min <- 10  
dbh\_max <- 40  
  
# Calculate the number of trees in each plot with DBH between 10 and 40 cm  
result <- data %>%  
 filter(treedb >= dbh\_min, treedb <= dbh\_max) %>%  
 group\_by(plotID) %>%  
 summarise(tree\_10\_40 = n())  
  
# Generate a list of all unique plotIDs  
all\_plotIDs <- data %>%  
 distinct(plotID)  
  
# Left join the summarized data with the list of all unique plotIDs  
result <- left\_join(all\_plotIDs, result, by = "plotID")  
  
# Replace NA values with 0  
result[is.na(result)] <- 0  
  
# Print or use the result  
print(result)

## # A tibble: 99 × 2  
## plotID tree\_10\_40  
## <chr> <int>  
## 1 L1\_10 101  
## 2 L1\_13 29  
## 3 L1\_17 88  
## 4 L1\_18 54  
## 5 L1\_22 11  
## 6 L1\_24 125  
## 7 L1\_26 37  
## 8 L1\_27 125  
## 9 L1\_31 66  
## 10 L1\_33 94  
## # ℹ 89 more rows

# In this modified script, we first generate a list of all unique plotID values and then use a left join to combine it with the summarized data. This ensures that even the plots with zero trees in the specified DBH range are included in the result. Finally, we replace any NA values with 0 to represent the plots with no matching trees in the given DBH range.  
  
# Save the result to a new Excel file  
# write\_xlsx(result, "result\_file.xlsx")  
  
  
#-------------------------------------------------------------------------------  
# Calculate the basal area only of the broadleave with a dbh > 40cm  
# Define the conditions for filtering  
dbh\_condition\_2 <- new\_data$treedb > 40  
  
# To define the species to be removed  
unique\_sp <- unique(new\_data$treesp) # alternative unique\_plots <- unique(CZ\_JH1[,"plotID"])  
print(unique\_sp) # in 99 plots

## [1] "Picea abies" "Betula pendula" "Pinus sylvestris"   
## [4] "Fagus sylvatica" "Fraxinus excelsior" "Sorbus aucuparia"   
## [7] "Larix decidua" "Acer pseudoplatanus" "Tilia cordata"   
## [10] "Acer platanoides" "Abies alba" "Ulmus glabra"   
## [13] "Sambucus racemosa" "Sambucus nigra" "Quercus robur"   
## [16] "Populus alba" "Ulmus minor" "Salix caprea"   
## [19] "Carpinus betulus" "Pinus nigra" "Quercus petraea"   
## [22] "Sorbus torminalis" "Acer campestre" "Quercus rubra"   
## [25] "Pinus strobus" "Populus tremula" "Acer pseudoplatatnus"

species\_to\_remove <- c("Picea abies", "Pinus sylvestris", "Larix decidua",  
 "Abies alba","Pinus nigra","Pinus strobus")  
  
# Use subset to filter the dataframe  
filtered\_df <- subset(new\_data, dbh\_condition\_2 & !treesp %in% species\_to\_remove)  
  
# Print the filtered dataframe  
print(filtered\_df)

## # A tibble: 973 × 33  
## siteID standID plotID treeID genus species treesp treedb treeht treevol  
## <chr> <chr> <chr> <chr> <chr> <chr> <chr> <dbl> <dbl> <chr>   
## 1 CZ\_JH1\_L1 B L1\_13 113S1 Fagus sylvatica Fagus … 52.5 31.6 3.5402…  
## 2 CZ\_JH1\_L1 B L1\_13 113S10 Fagus sylvatica Fagus … 42.8 30.2 2.2018…  
## 3 CZ\_JH1\_L1 B L1\_13 113S11 Fagus sylvatica Fagus … 46.2 35.2 2.9957…  
## 4 CZ\_JH1\_L1 B L1\_13 113S12 Fagus sylvatica Fagus … 43.5 31 2.3360…  
## 5 CZ\_JH1\_L1 B L1\_13 113S13 Fagus sylvatica Fagus … 48.5 35.5 3.3481…  
## 6 CZ\_JH1\_L1 B L1\_13 113S16 Fagus sylvatica Fagus … 41.8 36.2 2.4886…  
## 7 CZ\_JH1\_L1 B L1\_13 113S17 Fagus sylvatica Fagus … 48.5 32 3.0302…  
## 8 CZ\_JH1\_L1 B L1\_13 113S20 Fagus sylvatica Fagus … 49.2 36 3.4983…  
## 9 CZ\_JH1\_L1 B L1\_13 113S21 Fagus sylvatica Fagus … 51.2 32.2 3.3   
## 10 CZ\_JH1\_L1 B L1\_13 113S23 Fagus sylvatica Fagus … 60.8 34.8 5.3126…  
## # ℹ 963 more rows  
## # ℹ 23 more variables: cenver <chr>, azimuth <chr>, distan <chr>, coordx <chr>,  
## # coordy <chr>, crowndep <dbl>, crownrad1 <chr>, crownrad2 <chr>,  
## # crownrad3 <chr>, crownrad4 <chr>, treevit <dbl>, alive <dbl>, decsta <dbl>,  
## # plosiz <dbl>, weisiz <dbl>, TreMn1 <chr>, TreMn2 <chr>, TreMn3 <chr>,  
## # .. <lgl>, ...30 <lgl>, ...31 <lgl>, ...32 <dbl>, basal\_area <dbl>

# write.csv(filtered\_df, "C:/iLand/2023/plot\_bottoms\_up/Jenik/hhh.csv")  
  
# Now sum the basal area   
# Group by plotID and species, then calculate the sum of basal areas  
  
summed\_tree\_areas\_sp <- filtered\_df %>%  
 group\_by(plotID, treesp) %>%  
 summarize(total\_basal\_area = sum(basal\_area))

## `summarise()` has grouped output by 'plotID'. You can override using the  
## `.groups` argument.

# Print the resulting dataframe  
print(summed\_tree\_areas\_sp)

## # A tibble: 84 × 3  
## # Groups: plotID [57]  
## plotID treesp total\_basal\_area  
## <chr> <chr> <dbl>  
## 1 L1\_03 Fagus sylvatica 4.72   
## 2 L1\_13 Fagus sylvatica 8.45   
## 3 L1\_17 Betula pendula 1.13   
## 4 L1\_17 Fraxinus excelsior 1.04   
## 5 L1\_18 Fagus sylvatica 7.14   
## 6 L1\_22 Acer platanoides 0.350  
## 7 L1\_22 Acer pseudoplatanus 0.129  
## 8 L1\_22 Fagus sylvatica 7.88   
## 9 L1\_22 Tilia cordata 1.43   
## 10 L1\_24 Acer platanoides 0.139  
## # ℹ 74 more rows

#write.csv(summed\_tree\_areas\_sp, "C:/iLand/2023/plot\_bottoms\_up/Jenik/summed\_tree\_areas\_sp\_over40dbh.csv")  
  
# Do the same but only one value for each plot  
summed\_tree\_areas\_plot <- filtered\_df %>%  
 group\_by(plotID) %>%  
 summarize(total\_basal\_area = sum(basal\_area))  
  
# Print the resulting dataframe  
print(summed\_tree\_areas\_plot)

## # A tibble: 57 × 2  
## plotID total\_basal\_area  
## <chr> <dbl>  
## 1 L1\_03 4.72   
## 2 L1\_13 8.45   
## 3 L1\_17 2.16   
## 4 L1\_18 7.14   
## 5 L1\_22 9.79   
## 6 L1\_24 3.29   
## 7 L1\_26 4.77   
## 8 L1\_33 1.32   
## 9 L1\_34 5.86   
## 10 L1\_44 0.229  
## # ℹ 47 more rows

#write.csv(summed\_tree\_areas\_plot, "C:/iLand/2023/plot\_bottoms\_up/Jenik/summed\_tree\_ba\_broadl\_over40dbh.csv")  
  
#-------------------------------------------------------------------------------  
# Clean and harmonize 2 dataframes in the way to have the same plotID and n of rows  
  
# Filter df1 to keep only rows with plotID values that exist in df2  
tab2\_broadl <- tab2 %>%  
 filter(plotID %in% summed\_tree\_areas\_plot$plotID)  
  
# Use left\_join to merge the harmonized dataframes based on plotID  
merged\_df <- left\_join(tab2\_broadl, summed\_tree\_areas\_plot, by = "plotID")  
  
# Print the merged dataframe  
print(merged\_df)

## # A tibble: 57 × 19  
## plotID domin\_sp age volume\_dw tree\_n basal\_area dbh\_var lai\_sim lai\_emp  
## <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 L1\_03 larch 186 2.51 114 11.7 531. 3.36 3.35  
## 2 L1\_13 beech 126 0.978 73 10.9 98.7 4.17 2.68  
## 3 L1\_17 EU 103 5.38 202 6.17 147. 1.92 2.43  
## 4 L1\_18 spruce 148 12.2 97 10.8 385. 3.88 2.28  
## 5 L1\_22 beech 191 12.7 48 10.5 286. 3.80 2.46  
## 6 L1\_24 beech 204 27.8 158 8.32 201. 3.32 3.60  
## 7 L1\_26 beech 191 50.3 83 6.85 463. 2.51 3.14  
## 8 L1\_33 spruce 121 6.47 231 5.74 151. 2.06 3.17  
## 9 L1\_34 beech 173 3.48 91 9.83 435. 3.43 2.30  
## 10 L1\_44 spruce 95 2.41 108 10.3 47.1 4.32 1.85  
## # ℹ 47 more rows  
## # ℹ 10 more variables: reg <dbl>, n\_sp <dbl>, shannon <dbl>, ba\_broadl <dbl>,  
## # max\_dbh <dbl>, `trees<=40` <dbl>, `conf>40` <dbl>, `broadl>40` <dbl>,  
## # `ba\_broadl>40` <dbl>, total\_basal\_area <dbl>

#write.csv(merged\_df, "C:/iLand/2023/plot\_bottoms\_up/Jenik/Bdv\_predictors\_table\_final\_broadl\_40.csv")

# For the original script go through this path:

C:/Users/baldo/Documents/GitHub/Bottoms-Up/STSM/BDV\_tables/table\_edit\_workflow/forest\_structure\_corr\_concise\_20231018.R

*Note that the echo = FALSE parameter was added to the code chunk to prevent printing of the R code that generated the plot.*

#-------------------------------------------------------------------------------  
# Create the proportion for the species seed background probability   
  
tab2 <- read\_xlsx("C:/iLand/2023/20230901\_Bottoms\_Up/Sources\_bottoms\_up/Jenik/final\_table\_imp/Bdv\_predictors\_table\_final\_20231002.xlsx")  
  
#head(tab1)  
#head(tab1)  
  
# import the tree data  
  
raw\_data <- "C:/iLand/2023/20230901\_Bottoms\_Up/Sources\_bottoms\_up/Jenik/final\_table\_imp/Raw\_data\_structure\_CZ\_JH1\_final.xlsx"  
  
excel\_sheets(raw\_data)

## [1] "legend" "live\_dead\_trees"   
## [3] "standing\_lying\_deadwood"

raw\_data\_sp <- "live\_dead\_trees"  
  
data <- read\_excel(raw\_data,raw\_data\_sp, col\_names=T)

## New names:  
## • `` -> `...30`  
## • `` -> `...31`  
## • `` -> `...32`

print(data)

## # A tibble: 15,901 × 32  
## siteID standID plotID treeID genus species treesp treedb treeht treevol  
## <chr> <chr> <chr> <chr> <chr> <chr> <chr> <dbl> <dbl> <chr>   
## 1 CZ\_JH1\_L1 C L1\_10 110S1 Picea abies Picea a… 44.2 26.8 1.7168…  
## 2 CZ\_JH1\_L1 C L1\_10 110S10 Picea abies Picea a… 35.5 27.3 1.1855…  
## 3 CZ\_JH1\_L1 C L1\_10 110S100 Picea abies Picea a… 32.2 25.8 0.9345…  
## 4 CZ\_JH1\_L1 C L1\_10 110S101 Picea abies Picea a… 47.5 31.3 2.3280…  
## 5 CZ\_JH1\_L1 C L1\_10 110S102 Picea abies Picea a… 41.5 30.7 1.7892…  
## 6 CZ\_JH1\_L1 C L1\_10 110S103 Picea abies Picea a… 35.5 30.4 1.3391…  
## 7 CZ\_JH1\_L1 C L1\_10 110S104 Picea abies Picea a… 29.5 28.1 0.8809…  
## 8 CZ\_JH1\_L1 C L1\_10 110S105 Picea abies Picea a… 31.2 28.2 0.9772…  
## 9 CZ\_JH1\_L1 C L1\_10 110S106 Picea abies Picea a… 28.5 27.9 0.8217…  
## 10 CZ\_JH1\_L1 C L1\_10 110S107 Picea abies Picea a… 28 29 0.8320…  
## # ℹ 15,891 more rows  
## # ℹ 22 more variables: cenver <chr>, azimuth <chr>, distan <chr>, coordx <chr>,  
## # coordy <chr>, crowndep <dbl>, crownrad1 <chr>, crownrad2 <chr>,  
## # crownrad3 <chr>, crownrad4 <chr>, treevit <dbl>, alive <dbl>, decsta <dbl>,  
## # plosiz <dbl>, weisiz <dbl>, TreMn1 <chr>, TreMn2 <chr>, TreMn3 <chr>,  
## # .. <lgl>, ...30 <lgl>, ...31 <lgl>, ...32 <dbl>

# write.csv(new\_data, "C:/iLand/2023/plot\_bottoms\_up/Jenik/Raw\_data\_with\_basal\_area.csv")  
  
# Crete the species proportion per site  
  
library(dplyr)  
library(knitr)  
  
tree\_data <- data %>%  
 mutate(Site = sub("\_[0-9]+", "", plotID),  
 Plot = sub("([A-Z]+)[0-9]+\_[0-9]+", "\\1\\2", plotID))  
  
# Count the number of trees for each species within each site  
site\_species\_counts <- tree\_data %>%  
 group\_by(Site, treesp) %>%  
 summarise(TreeCount = n())

## `summarise()` has grouped output by 'Site'. You can override using the  
## `.groups` argument.

# Calculate species proportions within each site  
site\_species\_proportions <- site\_species\_counts %>%  
 group\_by(Site) %>%  
 mutate(SpeciesProportion = TreeCount / sum(TreeCount))  
  
# Print the species proportions for all sites  
print(site\_species\_proportions)

## # A tibble: 70 × 4  
## # Groups: Site [6]  
## Site treesp TreeCount SpeciesProportion  
## <chr> <chr> <int> <dbl>  
## 1 L1 Abies alba 7 0.00245   
## 2 L1 Acer platanoides 15 0.00524   
## 3 L1 Acer pseudoplatanus 272 0.0950   
## 4 L1 Acer pseudoplatatnus 1 0.000349  
## 5 L1 Betula pendula 21 0.00734   
## 6 L1 Fagus sylvatica 499 0.174   
## 7 L1 Fraxinus excelsior 214 0.0748   
## 8 L1 Larix decidua 394 0.138   
## 9 L1 Picea abies 1348 0.471   
## 10 L1 Pinus sylvestris 13 0.00454   
## # ℹ 60 more rows

# Visualize species proportions for all sites  
for (site in unique(site\_species\_proportions$Site)) {  
 # Filter the data for the specific site  
 site\_proportions <- filter(site\_species\_proportions, Site == site)  
   
 # Extract the species names and proportions for the current site  
 species\_names <- site\_proportions$treesp  
 species\_proportions <- site\_proportions$SpeciesProportion  
   
 # Print the species names and proportions  
 cat("Species Proportions in Site", site, ":\n")  
 for (i in 1:length(species\_names)) {  
 cat(species\_names[i], ": ", species\_proportions[i], "\n")  
 }  
   
 # Create a data frame for the table  
 site\_table <- data.frame(Species = species\_names, Proportion = species\_proportions)  
   
 # Display the table in R Markdown  
 kable(site\_table, format = "markdown")  
}

## Species Proportions in Site L1 :  
## Abies alba : 0.002445842   
## Acer platanoides : 0.00524109   
## Acer pseudoplatanus : 0.09503843   
## Acer pseudoplatatnus : 0.000349406   
## Betula pendula : 0.007337526   
## Fagus sylvatica : 0.1743536   
## Fraxinus excelsior : 0.07477289   
## Larix decidua : 0.137666   
## Picea abies : 0.4709993   
## Pinus sylvestris : 0.004542278   
## Populus alba : 0.000349406   
## Populus tremula : 0.001048218   
## Quercus robur : 0.001048218   
## Sambucus nigra : 0.001048218   
## Sambucus racemosa : 0.000698812   
## Sorbus aucuparia : 0.00174703   
## Tilia cordata : 0.01327743   
## Ulmus glabra : 0.00698812   
## Ulmus minor : 0.001048218

## Species Proportions in Site L2 :  
## Acer platanoides : 0.000435161   
## Acer pseudoplatanus : 0.001305483   
## Fagus sylvatica : 0.5739774   
## Picea abies : 0.4225413   
## Sorbus aucuparia : 0.000870322   
## Ulmus minor : 0.000870322   
## Species Proportions in Site L3 :  
## Abies alba : 0.01972318   
## Acer pseudoplatanus : 0.002422145   
## Betula pendula : 0.001038062   
## Fagus sylvatica : 0.6384083   
## Picea abies : 0.3384083   
## Species Proportions in Site L4 :  
## Abies alba : 0.03033626   
## Acer platanoides : 0.0003654971   
## Acer pseudoplatanus : 0.006578947   
## Betula pendula : 0.00877193   
## Fagus sylvatica : 0.3837719   
## Fraxinus excelsior : 0.009137427   
## Larix decidua : 0.001461988   
## Picea abies : 0.5277778   
## Pinus sylvestris : 0.0007309942   
## Quercus robur : 0.02704678   
## Salix caprea : 0.0003654971   
## Sorbus aucuparia : 0.0007309942   
## Tilia cordata : 0.002923977   
## Species Proportions in Site L5 :  
## Acer campestre : 0.0004145937   
## Acer pseudoplatanus : 0.00787728   
## Betula pendula : 0.04767828   
## Carpinus betulus : 0.1119403   
## Fagus sylvatica : 0.005804312   
## Fraxinus excelsior : 0.0004145937   
## Larix decidua : 0.006633499   
## Picea abies : 0.4365672   
## Pinus nigra : 0.0004145937   
## Pinus sylvestris : 0.2993367   
## Quercus petraea : 0.0331675   
## Quercus robur : 0.04643449   
## Salix caprea : 0.001658375   
## Sorbus torminalis : 0.0008291874   
## Ulmus glabra : 0.0004145937   
## Ulmus minor : 0.0004145937   
## Species Proportions in Site L6 :  
## Betula pendula : 0.01960784   
## Carpinus betulus : 0.001479837   
## Larix decidua : 0.009248983   
## Picea abies : 0.05697373   
## Pinus strobus : 0.04550499   
## Pinus sylvestris : 0.5123936   
## Quercus petraea : 0.1172771   
## Quercus robur : 0.18165   
## Quercus rubra : 0.0003699593   
## Sorbus aucuparia : 0.001109878   
## Tilia cordata : 0.05438402