

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 consortium. 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 relevant 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_predi
```

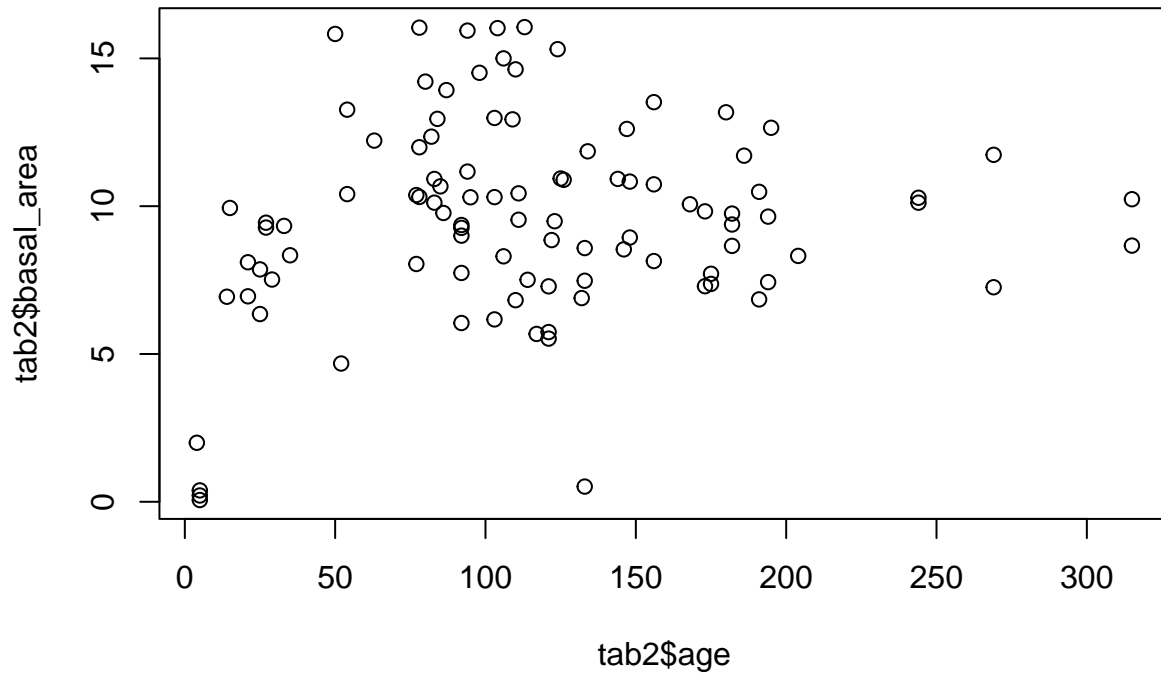
```
tab_brow <-read.csv("I:/iLand/2022/20220604_browsing_first/variables.DB_20220728.csv")
```

```
tab2 <- read_xlsx("C:/iLand/2023/20230901_Bottoms_Up/Sources_bottoms_up/Jenik/final_table_imp/Bdv_predi
```

```
#head(tab1)
head(tab2)
```

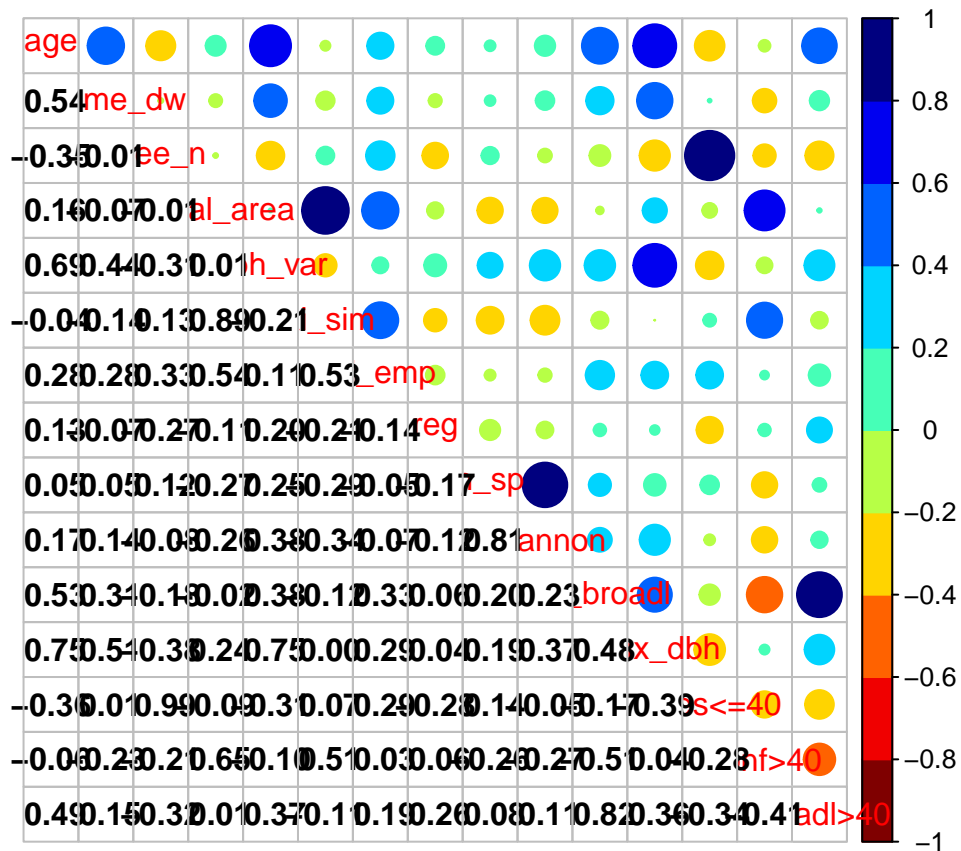
```
## # A tibble: 6 x 19
##   plotID domin_sp   age volume_dw tree_n basal_area dbh_var lai_sim lai_emp
##   <chr>   <chr>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
## 1 L1_03  larch     186     2.51    114     11.7    531.     3.36    3.35
## 2 L1_07  spruce    111     4.66    213     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 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>, MeanGAMAge <dbl>
```

```
plot(tab2$age, tab2$basal_area)
```



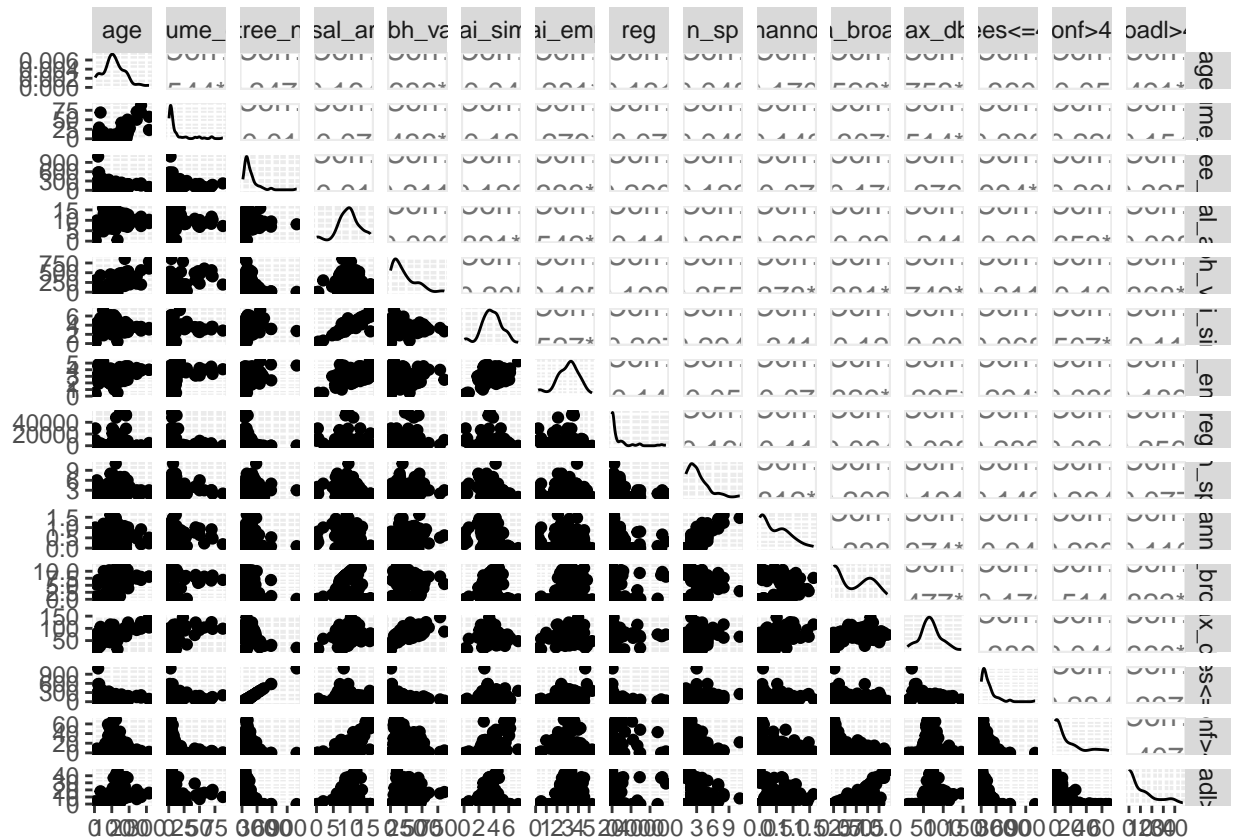
```
col4 <- colorRampPalette(c("#7F0000", "red", "#FF7F00", "yellow", "#7FFF7F", "cyan", "#007FFF", "blue",
#----- just do the correlation plot all together
a.num<-tab2[,3:17]

# 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")#, dia.
```



#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_structur
```

```
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 x 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 x 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 x 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 x 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 x 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
```

```
# 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 x 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 x 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 x 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 x 20
##   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 11 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>, MeanGAMAge <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_concis

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_predi

#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_struct

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 x 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")
```

```
# Create 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 x 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

```