Tree-mortality Data

Introduction

Holger Sennhenn-Reulen

Forest Growth Modeling & Computer Science,
Department of Growth and Yield,
Northwest German Forest Research Institute.

November 26, 2019

Abstract This document shows the steps I've performed in order to come to a data-set we can use for statistical tree-mortality modeling.

The data is openly available in a repository to the following publication (sorry for incorrect typeface of Polish letters):

> B. Brzeziecki, K. Bielak, L. Bolibok, S. Drozdowski, J. Zajaczkowski, H. Zybura (2019): *Structural and compositional dynamics of strictly protected woodland communities with silvicultural implications, using Bialowieza Forest as an example.* Annals of Forest Science 75(3).

Here is the link:

$$\texttt{https://bit.ly/2DfgBde}$$

to the blog post where you can find links to the published manuscript and also to the data repository.

Contents

1 Organise R Session

First we load all the packages we need, clean the global environment – if needed –, and have a look on the version numbers of R and packages:

```
library("plyr")
library("ggplot2")
rm(list = ls())
sessionInfo()
```

```
1  R version 3.6.1 (2019-07-05)
2  Platform: x86_64-pc-linux-gnu (64-bit)
3  Running under: Ubuntu 18.04.3 LTS
4
5  Matrix products: default
6  BLAS:   /usr/lib/x86_64-linux-gnu/blas/libblas.so.3.7.1
7  LAPACK: /usr/lib/x86_64-linux-gnu/lapack/liblapack.so.3.7.1
8
9  locale:
10  [1] LC_CTYPE=de_DE.UTF-8       LC_NUMERIC=C
11  [3] LC_TIME=de_DE.UTF-8        LC_COLLATE=de_DE.UTF-8
12  [5] LC_MONETARY=de_DE.UTF-8    LC_MESSAGES=de_DE.UTF-8
13  [7] LC_PAPER=de_DE.UTF-8       LC_NAME=C
14  [9] LC_ADDRESS=C               LC_TELEPHONE=C
15 [11] LC_MEASUREMENT=de_DE.UTF-8 LC_IDENTIFICATION=C
16
17  attached base packages:
18  [1] stats     graphics  grDevices utils     datasets  methods   base
19
20  other attached packages:
21  [1] ggplot2_3.2.1 plyr_1.8.4
22
23  loaded via a namespace (and not attached):
24   [1] Rcpp_1.0.2        withr_2.1.2       assertthat_0.2.1 dplyr_0.8.3
25   [5] crayon_1.3.4      R6_2.4.0          grid_3.6.1        gtable_0.3.0
26   [9] magrittr_1.5      scales_1.0.0      pillar_1.4.2      rlang_0.4.1
27  [13] lazyeval_0.2.2    tools_3.6.1       glue_1.3.1        purrr_0.3.2
28  [17] munsell_0.5.0     xfun_0.9          compiler_3.6.1    pkgconfig_2.0.3
29  [21] colorspace_1.4-1  tidyselect_0.2.5  knitr_1.25        tibble_2.1.3
```

2 Load data

Prior to loading the data, I saved the respective spreadsheets into separate csv-files:

```
dl <- read.csv("LIVE_TREES.csv", dec = ",")
di <- read.csv("INGROWTH_TREES.csv", dec = ",")
do <- read.csv("OUTGROWTH_TREES.csv", dec = ",")
```

Let's have a look on the structure of those datasets:

```
str(dl)
1  'data.frame':        12299 obs. of  8 variables:
2  $ Community    : int  1 1 1 1 1 1 1 1 1 1 ...
3  $ Plot         : int  3 3 3 3 3 3 3 3 3 3 ...
4  $ Patch        : int  6 6 6 6 6 6 6 6 6 6 ...
5  $ Census       : int  1 1 1 1 1 1 1 1 1 1 ...
6  $ Species      : int  1 1 1 1 1 1 1 1 2 2 ...
7  $ DbhClassBottom: int  50 100 150 200 250 300 350 400 50 100 ...
8  $ Trees        : int  4 16 19 22 14 5 6 2 34 69 ...
9  $ BasalArea    : num  0.0245 0.2028 0.4593 0.8635 0.7942 ...
   str(di)
1  'data.frame':        1262 obs. of  8 variables:
2  $ Community    : int  1 1 1 1 1 1 1 1 1 1 ...
3  $ Plot         : int  3 3 3 3 3 3 3 3 3 3 ...
4  $ Patch        : int  6 6 6 6 6 6 6 6 6 6 ...
5  $ Census       : int  1 1 1 1 2 2 2 2 2 3 ...
6  $ Species      : int  2 2 4 4 2 4 4 5 5 2 ...
7  $ DbhClassBottom: int  50 100 50 100 50 50 100 50 100 50 ...
8  $ Trees        : int  2 1 35 2 6 19 2 7 2 1 ...
9  $ BasalArea    : num  0.00635 0.0141 0.12945 0.01651 0.02155 ...
   str(do)
1  'data.frame':        4311 obs. of  8 variables:
2  $ Community    : int  1 1 1 1 1 1 1 1 1 1 ...
3  $ Plot         : int  3 3 3 3 3 3 3 3 3 3 ...
4  $ Patch        : int  6 6 6 6 6 6 6 6 6 6 ...
5  $ Census       : int  1 1 1 1 1 1 1 1 1 1 ...
6  $ Species      : int  1 1 1 1 1 2 2 2 2 2 ...
7  $ DbhClassBottom: int  50 100 150 200 250 50 100 150 200 250 ...
8  $ Trees        : int  2 2 1 1 1 21 27 10 4 3 ...
9  $ BasalArea    : num  0.0138 0.0227 0.023 0.046 0.0507 ...
```

From the Readme-file in the data-repository, we get valuable information on how to link the three different datasets:

> The exact meaning of given census number differs between tables. In the case of LIVE_TREES table, census number indicates measurement year (1 for 1936, 2 for 1957, a so on to 7 for 2012). In the case of INGROWTH_TREES table, census #N indicates period between censuses N and N+1, during which trees from particular dbh class have reached or passed dbh threshold value for the first time. In other words, the census #N points to trees that were counted for the first time on N+1 census. Census number range in this table is 1-6.

In the case of OUTGROWTH_TREES table, census #N stands for period between censuses N and N+1, as well, but it should be read as the last census in which the tree had been alive, before it was found dead in N+1 census. Census number range in this table is 1-6, as well.

Let's have a check of those numbers in the next subsection.

2.1 Check: Do numbers match?

```
sdfl <- subset(dl, Community == 1 & Plot == 3 & Patch == 6 &
                Species == 2)[, c("Census", "DbhClassBottom", "Trees")]
sdfi <- subset(di, Community == 1 & Plot == 3 & Patch == 6 &
                Species == 2)[, c("Census", "DbhClassBottom", "Trees")]
sdfo <- subset(do, Community == 1 & Plot == 3 & Patch == 6 &
                Species == 2)[, c("Census", "DbhClassBottom", "Trees")]
ddl <- ddply(sdfl, c("Census"), summarise, n = sum(Trees))
ddi <- ddply(sdfi, c("Census"), summarise, n = sum(Trees))
ddo <- ddply(sdfo, c("Census"), summarise, n = sum(Trees))
ddl
```

```
  Census    n
1      1  186
2      2  123
3      3  100
4      4   77
5      5   59
6      6   51
7      7   40
```

```
diff(ddl$n)
```

```
[1] -63 -23 -23 -18  -8 -11
```

```
ddi
```

```
  Census n
1      1 3
2      2 6
3      3 1
4      5 2
5      6 1
```

```
ddo
```

```
  Census  n
1      1 66
2      2 29
3      3 24
4      4 18
5      5 10
6      6 12
```

Yes, they do; the differences in the number of tree in the live-tree dataset can be calculated by:

$$\Delta l = o - i$$

3 Merge live-tree and outgrowth-tree data

Caution: The next two commands take a loooong time:

```
ddl <- ddply(dl, c("Census", "Community", "Plot", "Patch", "Species", "DbhClassBottom"),
             summarise, .drop = FALSE,
             n = sum(Trees))
ddo <- ddply(do, c("Census", "Community", "Plot", "Patch", "Species", "DbhClassBottom"),
             summarise, .drop = FALSE,
             n = sum(Trees))
ddl$key <- paste(ddl$Census, ddl$Community, ddl$Plot, ddl$Patch,
                 ddl$Species, ddl$DbhClassBottom, sep = "_")
ddo$key <- paste(ddo$Census, ddo$Community, ddo$Plot, ddo$Patch,
                 ddo$Species, ddo$DbhClassBottom, sep = "_")
```

Merge those two data-sets:

```
d <- merge(ddl, ddo, by = "key")
rm(ddl, ddo)
```

And only keep lines with observed trees in the respective diameter classes:

```
d <- d[d$n.x > 0.5, ]
d <- data.frame(Census = d$Census.x,
                Community = d$Community.x,
                Plot = d$Plot.x,
                Patch = d$Patch.x,
                Species = d$Species.x,
                DbhClassBottom = d$DbhClassBottom.x,
                n_all = d$n.x,
                n_dead = d$n.y)
d$key <- paste(d$Census, d$Community, d$Plot, d$Patch, d$Species,
               d$DbhClassBottom, sep = "_")
d$group_id <- paste(d$Community, d$Plot, d$Patch, d$Species,
                    d$DbhClassBottom, sep = "_")
d$group_id_coarse <- paste(d$Community, d$Plot, d$Patch,
                           d$Species, sep = "_")
```

4 Final data

This lead us to the final data-set:

```
str(d)
```

```
1  'data.frame':         10651 obs. of  11 variables:
2  $ Census        : int  1 1 1 1 1 1 1 1 1 1 ...
3  $ Community     : int  1 1 1 1 1 1 1 1 1 1 ...
4  $ Plot          : int  3 3 3 3 3 3 3 3 3 3 ...
5  $ Patch         : int  6 6 6 6 6 6 6 6 6 6 ...
6  $ Species       : int  1 1 1 1 1 1 1 1 2 2 ...
7  $ DbhClassBottom : int  100 150 200 250 300 350 400 50 100 150 ...
8  $ n_all         : int  16 19 22 14 5 6 2 4 69 46 ...
9  $ n_dead        : int  2 1 1 1 0 0 0 2 27 10 ...
10 $ key           : chr  "1_1_3_6_1_100" "1_1_3_6_1_150" "1_1_3_6_1_200" "1_1_3_6_1_250" ...
11 $ group_id      : chr  "1_3_6_1_100" "1_3_6_1_150" "1_3_6_1_200" "1_3_6_1_250" ...
12 $ group_id_coarse: chr  "1_3_6_1" "1_3_6_1" "1_3_6_1" "1_3_6_1" ...
```
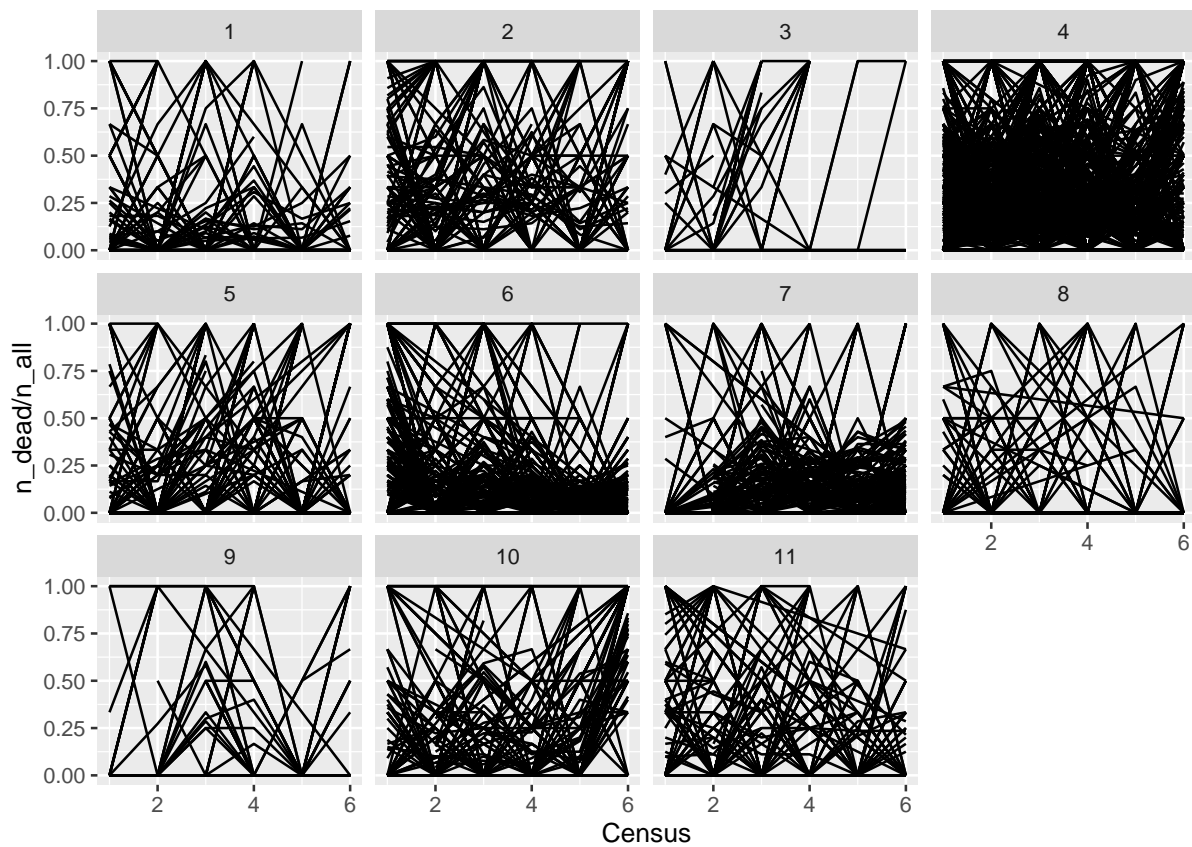
```
write.csv(d, file = "d_mortality.csv")
```

5 Some illustrations

```
ggplot(aes(y = n_all, x = Census, group = group_id), data = d) +
  geom_line() + facet_wrap(~ Species)
```



```
ggplot(aes(y = n_all, x = Census, group = group_id), data = d) +
  geom_line() + facet_wrap(~ Species)
```
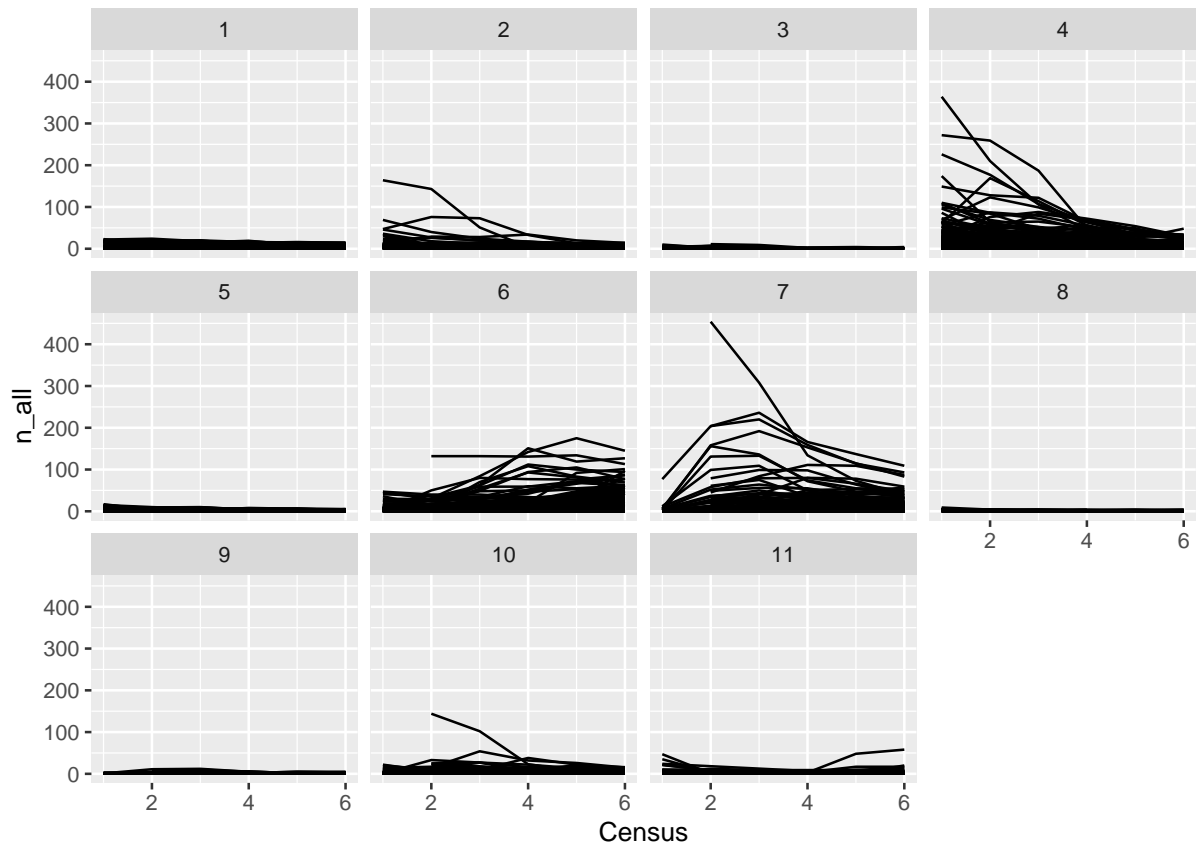
```
ggplot(aes(y = n_dead/n_all, x = Census, group = group_id), data = d) +
  geom_line() + facet_wrap(~ Species)
```

```
dd <- ddply(d, c("DbhClassBottom"), summarise,
            mm = mean(n_dead/n_all),
            n = sum(n_all))
par(mar = c(4, 4, 4, 0) + 0.1)
plot(d$DbhClassBottom + 25, d$n_dead/d$n_all, pch = 16, col = "grey",
     las = 1, ylab = "Mortality", xlab = "Dbh [mm]", bty = "n")
for (i in 1:nrow(dd)) {
  lines(dd$DbhClassBottom[i] + c(0, 50),
        rep(dd$mm[i], 2),
        col = rgb(0.4, 0.2, 0.2), lwd = 2)
}
axis(3, at = dd$DbhClassBottom + 25, lab = dd$n, las = 2)
mtext(side = 3, text = "Number of live trees at beginning of Census interval",
      line = 3)
```
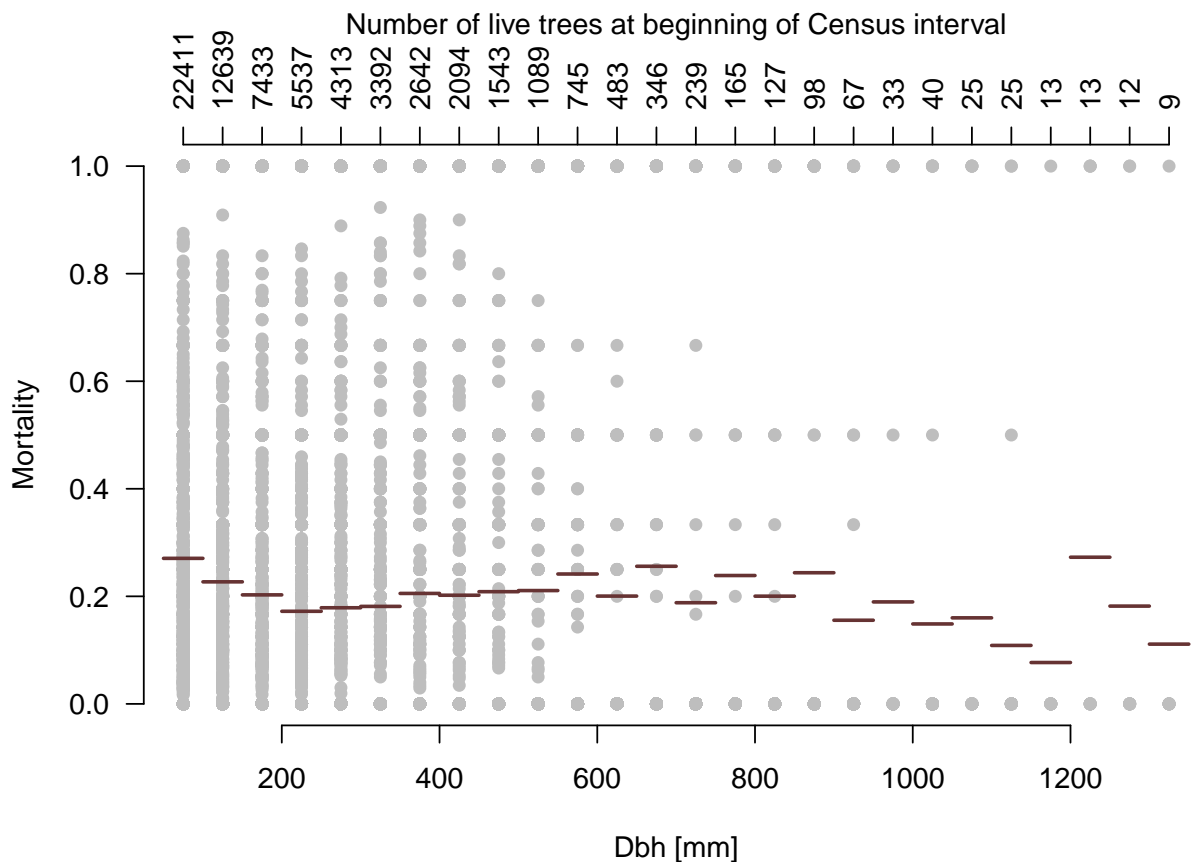

We see a higher mortality in trees with very low and higher physical age, ie. dbh.
How can we model this adequately? We will answer this question later during this course.