

# Making Individual Plot-Year Maps From Long-Term Data

Samantha L. Davis

June 25, 2015

## 1 Introduction

I need to make individual plot maps as they were in Year X (each of their measurement years, up to 7), which accurately represent the number of trees currently present in the plot, alongside their DBH measures if necessary. We have a few different types of trees that are present in the dataset.

1. Trees that were present before plot establishment as seedlings
2. Trees that were present before plot establishment as adults
3. Trees that were present before plot establishment as seedlings that died
4. Trees that were present before plot establishment as adults that died
5. Trees that appeared after plot establishment as seedlings that died
6. Trees that appeared after plot establishment as seedlings that survived
7. Seedlings that established and then died before DBH measure could be taken
8. Trees (saplings) that established then got too small for DBH before death

So, I think I need to do a loop for each of the 13,000 rows in `ssdAllTrees`. I should split them into groups, so that I know all NA's in DBH are expected because the tree is too small to measure, instead of it being measurements not taken yet. Also, let's switch the order of columns so that the xy columns are NOT last. That way, we have a definite number of columns before the dbh columns start.

```
> ## load library
> library(disperseR)
> ## switching column order
> ssdAllTrees <- cbind(ssdAllTrees[,c(1:6, 14:15, 7:13)])
> ##subsetting by years sampled
```

```

> sevenyears <- ssdAllTrees[ssdAllTrees$plot=="bellow" |
+                           ssdAllTrees$plot=="palate" |
+                           ssdAllTrees$plot=="reclusive" |
+                           ssdAllTrees$plot=="octane" |
+                           ssdAllTrees$plot=="artist",]
> sixyears <- ssdAllTrees[ssdAllTrees$plot=="trigger" |
+                          ssdAllTrees$plot=="rigid",]
> fiveyears <- ssdAllTrees[ssdAllTrees$plot=="crackers" |
+                           ssdAllTrees$plot=="trinity" |
+                           ssdAllTrees$plot=="realtor" |
+                           ssdAllTrees$plot=="sodium" |
+                           ssdAllTrees$plot=="distress" |
+                           ssdAllTrees$plot=="gravy" |
+                           ssdAllTrees$plot=="chestnut",]
> ## check to make sure all trees are accounted for
> (nrow(sevenyears) + nrow(sixyears) + nrow(fiveyears) == nrow(ssdAllTrees))

[1] TRUE

> ## delete the extraneous columns from sixyears and fiveyears
> fiveyears <- fiveyears[,-c(15,16)]
> sixyears <- sixyears[,-16]

```

Okay, so I now have three dataframes with DBH values that can be acted upon for each year. Let's get to writing this function. It should take a vector, which is a row of the dataframe. We know the identities of each of the first 9 columns for this vector, and the rest are to be dealt with in a loop. This function should output a data.frame that can be bound to a response data.frame. This response data.frame should be organized like so: Plot Name, Subplot, Tag, Species, Measure Year, Ingrowth Year, First Record Year, Death Year, x, y, DBH, Class

We need the first column, the plot, to look up sampling years. Columns 2 thru 4 just need to tag along, we don't need any information from them. If the ingrowth year is NA, that means that the tree was present in the plot before it was established. Now, in these plots, trees are checked yearly for establishment and death, but only measured every few years. What we know from ingrowth year, then, is the year that a seedling germinated and established. If ingrowth year is NOT NA, then we know the tree's entire life history, from the moment it was "born", and we can build the rows based on that.

If ingrowth year is NA, it gets a bit trickier. To deal with this, we can do a check. If there's no ingrowthyear, that means that the tree was alive before plot establishment. Therefore, we can make an assumption that if its first dbh value is NA, then it was too small to measure, e.g., a seedling. Otherwise, it's already a tree.

Wrapping all of this into a giant for loop, we can expand our ssdAllTrees data.frame to a df better suited for dispersal analysis

```

> ## get values where ingrowthyear has a value, for testing
> samplerows <- head(ssdAllTrees[!is.na(ssdAllTrees$ingrowthyear),])
> ## add in rows that have no ingrowth year
> samplerows <- rbind(samplerows,
+                     head(ssdAllTrees[is.na(ssdAllTrees$ingrowthyear),]))
> ##add in rows that have no ingrowthyear but DO have a mortality year
> samplerows <- rbind(samplerows,
+                     head(ssdAllTrees[is.na(ssdAllTrees$ingrowthyear) &
+                                     !is.na(ssdAllTrees$mortalityyear),]))
> ## make the function
>
> processTreeRows <- function(samplerows, ssdPlotDesc, ssdAllTrees){
+ i <- 1
+ j <- 1
+
+   finalrows <- data.frame(plot="dummy",
+                             treeid="dummy",
+                             species="dummy",
+                             ingrowth="dummy",
+                             firstrec="dummy",
+                             deathyear="dummy",
+                             x="dummy",
+                             y="dummy",
+                             measyear="dummy",
+                             dbh="dummy",
+                             stage="dummy",
+                             stringsAsFactors=FALSE
+                             )
+   for(i in 1:nrow(samplerows)){
+     ##get plot measurement years
+     plotYears <- (ssdPlotDesc[ssdPlotDesc$plot==samplerows[i, "plot"], 5:11])
+     ## make a dataframe of dbh1-7 and year values to search later
+     rPlotYears <- as.data.frame(cbind(paste("dbh", 1:7, sep=""),
+                                       as.numeric(plotYears[1,])),
+                                stringsAsFactors=FALSE,
+                                na.string="NA")
+     ## clean it up a bit
+     colnames(rPlotYears) <- c("colname", "year")
+     ##get only years where there are acutally values
+     years <- plotYears[!is.na(plotYears)]
+
+     ## if there is an ingrowth year in this particular row...
+     if(!is.na(samplerows[i,"ingrowthyear"])){
+       ## remove any years that are less than that.
+       years <- years[years >= samplerows[i, "ingrowthyear"]]
+       ## remember that this row is a seedling at least once

```

```

+     class <- "seedling"
+   } else{
+     ## if it was NA, mark it as seedling
+     if(is.na(samplerows[i, "dbh1"])){
+       class <- "seedling"
+     } else{
+       class <- "tree"
+     } ## end else
+   } ##end else
+
+
+   ## is there a mortalityyear
+   if(!is.na(samplerows[i, "mortalityyear"])){
+     ## remove any years greater than mortality year
+     years <- years[years < samplerows[i, "mortalityyear"]]
+     ## remember that this tree died
+     ## this is currently unused, but we may need it later
+     dead <- TRUE
+   }
+
+   ##if a seedling died before it could be officially measured
+   if(length(years)==0){
+     addedrow <- NULL
+     addedrow <- data.frame(plot=samplerows[i, "plot"],
+                             treeid=samplerows[i, "treeid"],
+                             species=samplerows[i, "sppcode"],
+                             ingrowth=samplerows[i, "ingrowthyear"],
+                             firstrec=samplerows[i, "yearfirstrecorded"],
+                             deathyear=samplerows[i, "mortalityyear"],
+                             x=samplerows[i, "xcoord"],
+                             y=samplerows[i, "ycoord"],
+                             measyear=samplerows[i, "yearfirstrecorded"],
+                             dbh=NA,
+                             stage="seedling",
+                             stringsAsFactors=FALSE
+                             )
+
+     finalrows <- rbind(finalrows, addedrow)
+
+   } else{
+     ## so, for each year that we have measurements...
+     for(j in 1:length(years)){
+       ##find which dbh column to use
+       colname <- rPlotYears[rPlotYears$year==years[j], "colname"]
+       ## clean it up because R sucks
+       colname <- colname[!is.na(colname)]
+       ## select that dbh value

```

```

+         dbh <- as.numeric(samplerows[i, colname])
+         ## make sure we don't label a tree as a seedling more than once
+         if(j > 1 & class=="seedling"){
+             class <- "tree"
+         }
+         ## build row for new table.
+         addedrow <- NULL
+         addedrow <- data.frame(plot=samplerows[i, "plot"],
+                                treeid=samplerows[i, "treeid"],
+                                species=samplerows[i, "sppcode"],
+                                ingrowth=samplerows[i, "ingrowthyear"],
+                                firstrec=samplerows[i, "yearfirstrecorded"],
+                                deathyear=samplerows[i, "mortalityyear"],
+                                x=samplerows[i, "xcoord"],
+                                y=samplerows[i, "ycoord"],
+                                measyear=years[j],
+                                dbh=dbh,
+                                stage=class,
+                                stringsAsFactors=FALSE
+                                )
+         #print(addedrow)
+         ## do the rbind
+         finalrows <- rbind(finalrows, addedrow)
+     } ## end j loop
+ } ## end seedling loop
+ } ## end i loop
+ return(finalrows[-1,])
+ } ## end function
> ## use the function
> finalrows <- processTreeRows(samplerows, ssdPlotDesc, ssdAllTrees)
> head(finalrows)

```

	plot	treeid	species	ingrowth	firstrec	deathyear	x	y
2	crackers	71	PILA	2004	2004	<NA>	89893.515	48118.8599999999
3	crackers	71	PILA	2004	2004	<NA>	89893.515	48118.8599999999
4	crackers	72	PILA	2000	2000	<NA>	89893.072	48146.4199999999
5	crackers	72	PILA	2000	2000	<NA>	89893.072	48146.4199999999
6	crackers	72	PILA	2000	2000	<NA>	89893.072	48146.4199999999
7	crackers	73	PILA	1997	1997	<NA>	89900.691	48127.4100000001
	measyear	dbh	stage					
2	2007	0.8	seedling					
3	2012	1.3	tree					
4	2002	0.1	seedling					
5	2007	1.8	tree					
6	2012	2.7	tree					
7	1997	0.9	seedling					

```
> tail(finalrows)
```

	plot	treeid	species	ingrowth	firstrec	deathyear	x	y
55	crackers	26	CADE	<NA>	1992	2009	89901.767	48135.6699999999
56	crackers	26	CADE	<NA>	1992	2009	89901.767	48135.6699999999
57	crackers	26	CADE	<NA>	1992	2009	89901.767	48135.6699999999
58	crackers	26	CADE	<NA>	1992	2009	89901.767	48135.6699999999
59	crackers	27	CADE	<NA>	1992	1993	89894.712	48120.5699999998
60	crackers	34	CADE	<NA>	1992	1993	89888.256	48139.3199999998

  

	measyear	dbh	stage
55	1992	8.1	tree
56	1997	8.2	tree
57	2002	8.3	tree
58	2007	8.2	tree
59	1992	10.6	tree
60	1992	6.1	tree

```
> nrow(finalrows)
```

```
[1] 59
```

Ok, so the tests seem to work, in that the stage value seems to be correct in all cases; that we don't have records for trees before they were alive; and that we don't have records for trees after they were dead. Let's go ahead and do it to a larger dataframe.

```
> palate <- processTreeRows(ssdAllTrees[ssdAllTrees$plot=="palate",], ssdPlotDesc, ssdAllTre
> str(palate)
```

```
'data.frame':      277 obs. of  11 variables:
 $ plot      : chr  "palate" "palate" "palate" "palate" ...
 $ treeid    : chr  "7342" "7342" "7342" "7343" ...
 $ species   : chr  "PIMO" "PIMO" "PIMO" "PIMO" ...
 $ ingrowth  : chr  NA NA NA NA ...
 $ firstrec  : chr  "1996" "1996" "1996" "1993" ...
 $ deathyear: chr  "1998" "1998" "1998" "1998" ...
 $ x         : chr  "100211.022" "100211.022" "100211.022" "100211.09" ...
 $ y         : chr  "51357.0600000001" "51357.0600000001" "51357.0600000001" "51356.9500000000" ...
 $ measyear  : chr  "1983" "1988" "1993" "1983" ...
 $ dbh       : chr  NA NA NA NA ...
 $ stage     : chr  "seedling" "tree" "tree" "seedling" ...
```

```
> nrow(palate)
```

```
[1] 277
```

```
> head(palate)
```

```

      plot treeid species ingrowth firstrec deathyear      x      y
2 palate  7342    PIMO    <NA>    1996    1998 100211.022 51357.0600000001
3 palate  7342    PIMO    <NA>    1996    1998 100211.022 51357.0600000001
4 palate  7342    PIMO    <NA>    1996    1998 100211.022 51357.0600000001
5 palate  7343    PIMO    <NA>    1993    1998 100211.09  51356.9500000002
6 palate  7343    PIMO    <NA>    1993    1998 100211.09  51356.9500000002
7 palate  7343    PIMO    <NA>    1993    1998 100211.09  51356.9500000002
      measyear dbh stage
2      1983 <NA> seedling
3      1988 <NA>   tree
4      1993 <NA>   tree
5      1983 <NA> seedling
6      1988 <NA>   tree
7      1993 17.3   tree

```

```
> tail(palate)
```

```

      plot treeid species ingrowth firstrec deathyear      x      y
273 palate  7398    PIMO    <NA>    1998    <NA> 100201.144 51445.8500000001
274 palate  7398    PIMO    <NA>    1998    <NA> 100201.144 51445.8500000001
275 palate  7398    PIMO    <NA>    1998    <NA> 100201.144 51445.8500000001
276 palate  7398    PIMO    <NA>    1998    <NA> 100201.144 51445.8500000001
277 palate  7398    PIMO    <NA>    1998    <NA> 100201.144 51445.8500000001
278 palate  7398    PIMO    <NA>    1998    <NA> 100201.144 51445.8500000001
      measyear dbh stage
273      1988 <NA>   tree
274      1993 <NA>   tree
275      1998    1   tree
276      2003  2.2   tree
277      2008  2.1   tree
278      2013  2.1   tree

```

The run time on this is not great, so we need to do it outside of the Sweave document. I'll save it all as an R object for the package. The R-object is entitled "expandedTrees" and can be loaded directly after the package loads. Keep in mind that the messy x/y coordinates are just because of a conversion to character, and I cleaned those up in the final dataset. I also converted everything back to numeric where necessary.

```
> head(expandedTrees)
```

```

      plot treeid species ingrowth firstrec deathyear      x      y measyear dbh
2 bellow  7093    PIMO    2003    2003      NA 100733.1 51405.46    2003 1.2
3 bellow  7093    PIMO    2003    2003      NA 100733.1 51405.46    2008 2.1
4 bellow  7093    PIMO    2003    2003      NA 100733.1 51405.46    2013 2.7
5 bellow  7094    PIMO    2004    2004      NA 100732.7 51405.07    2008 2.1
6 bellow  7094    PIMO    2004    2004      NA 100732.7 51405.07    2013 3.8

```

```

7 bellow 7095 PIMO 2004 2004 NA 100733.2 51405.02 2008 2.4
  stage
2 seedling
3 tree
4 tree
5 seedling
6 tree
7 seedling

```

```
> str(expandedTrees)
```

```

'data.frame':      51493 obs. of  11 variables:
 $ plot      : chr  "bellow" "bellow" "bellow" "bellow" ...
 $ treeid    : num  7093 7093 7093 7094 7094 ...
 $ species   : chr  "PIMO" "PIMO" "PIMO" "PIMO" ...
 $ ingrowth  : num  2003 2003 2003 2004 2004 ...
 $ firstrec  : num  2003 2003 2003 2004 2004 ...
 $ deathyear : num  NA NA NA NA NA NA NA NA NA NA ...
 $ x         : num  100733 100733 100733 100733 100733 ...
 $ y         : num  51405 51405 51405 51405 51405 ...
 $ measyear  : num  2003 2008 2013 2008 2013 ...
 $ dbh       : num  1.2 2.1 2.7 2.1 3.8 2.4 4.1 8.1 8.1 8.2 ...
 $ stage     : chr  "seedling" "tree" "tree" "seedling" ...

```

I'm not going to incorporate the script above into the package itself, because the type of code that an individual needs to do to get their plots into the appropriate format will differ based on what they start with. In other words, I don't think the monstrosity of code above is really reusable. It will, however, serve as a record of how I generated the `expandedTrees` table.