EP1162\_Prelim\_Data\_Screening

Hardy Griesbauer

18/02/2020

knitr::opts\_chunk$set(message=FALSE, warning = FALSE)  
library(tidyverse)  
  
# Load data  
load("../data/ep1162\_Data.RData")

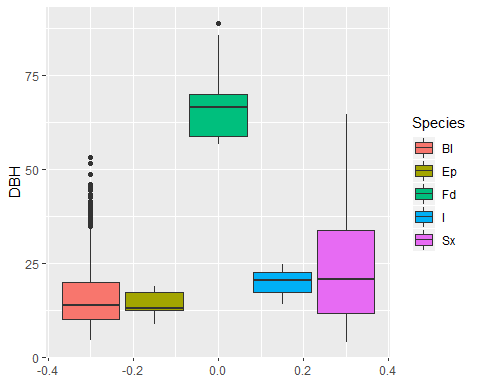
## Preliminary screening

I did some preliminary screening, to confirm data entry and detect any potential outliers. Some findings below.

## Outliers

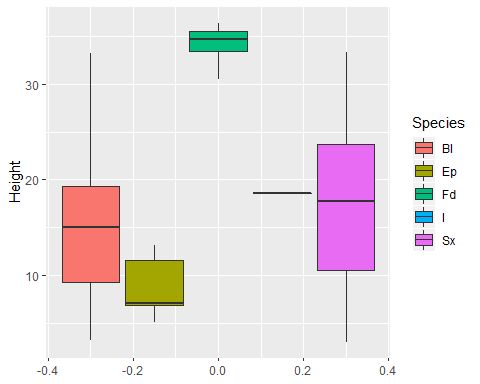
Quick boxplot to confirm DBH and height measurements

#### DBH



DBH measurements look reasonable, and no major outliers. Note that there is a tree with species recorded as “l”.

#### Height

 Again, measurements look reasonable here.

### Data entry

One tree has species recorded as “l”. I assume that it’s SAF, but might be worthwhile to verify with the field cards?

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Plot | TreeID | Species | Sector | Year | Status | DBH | Height | Comments | Conk | BlindConk | Scar | Fork.Crook | Frost.Crack | Mistletoe | R.Branch | DBT | Crown.Class | L.Crown |
| 16 | 314 | l | 2 | 1992 | Live | 14.2 | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| 16 | 314 | l | 2 | 1994 | Live | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| 16 | 314 | l | 2 | 1997 | Live | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| 16 | 314 | l | 2 | 2009 | Live | 20.6 | NA | Live | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA |
| 16 | 314 | l | 2 | 2019 | Live | 24.9 | 18.6 | scar in 1 | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA |

*ALRF staff confirmed that the species is “Bl”*, so we’ll change the species code accordingly in the next line of code:

dat<-  
 dat %>%   
 mutate(Species=replace(Species,Species=="l","Bl"))

There are also two trees recorded with id=631:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Plot | TreeID | Species | Sector | Year | Status | DBH | Height | Comments | Conk | BlindConk | Scar | Fork.Crook | Frost.Crack | Mistletoe | R.Branch | DBT | Crown.Class | L.Crown | n |
| 10 | 631 | Bl | 1 | 1992 | Live | NA | NA | Natural; AB; Snowpress | NA | NA | NA | NA | NA | NA | NA | NA | 4 | 7 | 2 |
| 10 | 631 | Bl | NA | 1992 | Live | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | 2 |
| 10 | 631 | Bl | 1 | 1994 | Live | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | 4 | 7 | 2 |
| 10 | 631 | Bl | NA | 1994 | Live | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | 2 |
| 10 | 631 | Bl | 1 | 1997 | Live | NA | NA | small leader, Dbh at 1.1m | NA | NA | NA | NA | NA | NA | NA | NA | 4 | 7 | 2 |
| 10 | 631 | Bl | NA | 1997 | Live | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | 2 |
| 10 | 631 | Bl | 1 | 2009 | Live | NA | NA | Live | NA | NA | NA | NA | NA | NA | NA | NA | 4 | 7 | 2 |
| 10 | 631 | Bl | NA | 2009 | Live | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | 2 |
| 10 | 631 | Bl | 1 | 2019 | Live | 7.7 | 6.1 | NA | NA | NA | NA | NA | NA | NA | NA | NA | 4 | 7 | 2 |
| 10 | 631 | Bl | NA | 2019 | Live | 7.7 | 6.1 | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | NA | 2 |

Looks like it is the same tree (exact same DBH and Height measured in 2019), so assume it’s safe to delete the duplicated entry?

*Confirmed that this tree was duplicated by mistake. We will remove one entry.*

dat<-  
 dat %>%   
 filter(!(Plot==10 & TreeID==631 & Sector=="NA")) # remove entry

There are also quite a few trees with their ID recorded as “New”. Not an issue from an analysis point of view,

### Decreasing DBH

I also screened trees that had a smaller DBH measurements than a previous measurement at some point. Below are some examples:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Plot | TreeID | Species | Sector | Year | Status | DBH |
| 3 | 279 | Sx | 1 | 1992 | Live | 43.5 |
| 3 | 279 | Sx | 1 | 1994 | Live | 43.3 |
| 3 | 279 | Sx | 1 | 1997 | Live | 43.2 |
| 3 | 279 | Sx | 1 | 2009 | Live | 43.5 |
| 3 | 279 | Sx | 1 | 2019 | Dead | 43.5 |
| 3 | 294 | Bl | 2 | 1992 | Live | 20.5 |
| 3 | 294 | Bl | 2 | 1994 | Live | 20.3 |
| 3 | 294 | Bl | 2 | 1997 | Live | 20.9 |
| 3 | 294 | Bl | 2 | 2009 | Live | 22.0 |
| 3 | 294 | Bl | 2 | 2019 | Live | 23.2 |
| 3 | 441 | Bl | 3 | 1992 | Live | 10.5 |
| 3 | 441 | Bl | 3 | 1994 | Live | 9.1 |
| 3 | 441 | Bl | 3 | 1997 | Live | 9.7 |
| 3 | 441 | Bl | 3 | 2009 | Live | 11.0 |
| 3 | 441 | Bl | 3 | 2019 | Live | 14.0 |
| 3 | 457 | Sx | 3 | 1992 | Live | 41.0 |
| 3 | 457 | Sx | 3 | 1994 | Live | 41.2 |
| 3 | 457 | Sx | 3 | 1997 | Live | 41.4 |
| 3 | 457 | Sx | 3 | 2009 | Live | 42.2 |
| 3 | 457 | Sx | 3 | 2019 | Dead | 41.7 |

In most cases, the differences are very small, and likely can be atrributed to slight measurement differences between years.

In total, there are 31 trees with this situation. I have the treeID saved and can provide them if we want to double check these trees.

## Trees without monotonically increasing DBH measurements

After discussion, we will handle trees without monotonically increasing DBH measurements as follows:

* Where DBH decreases occur in dead trees or during periods where trees died, we will treat these decreases as likely due to wood shrinkage/rot associated with mortality.
* Where DBH decreases occur during periods where trees were live, we will treat these decreases as a measurement error. We will use linear interpolation to correct these errors.

Code below.

### Trees with decreasing DBH over dead periods

# Step one: identify periods where DBH decrease occured during a period when tree waslive  
  
dbhDecrease %>%   
 filter(Status=="Dead") %>% # filter trees with decreasing DBH that had dead status at any point  
 left\_join(dat,by=c("Plot","TreeID")) %>% # join with dataset  
 dplyr::select(Plot,TreeID,Year=Year.y,Status=Status.y,DBH=DBH.y)

## # A tibble: 45 x 5  
## Plot TreeID Year Status DBH  
## <fct> <fct> <fct> <fct> <dbl>  
## 1 3 279 1992 Live 43.5  
## 2 3 279 1994 Live 43.3  
## 3 3 279 1997 Live 43.2  
## 4 3 279 2009 Live 43.5  
## 5 3 279 2019 Dead 43.5  
## 6 3 457 1992 Live 41.0  
## 7 3 457 1994 Live 41.2  
## 8 3 457 1997 Live 41.4  
## 9 3 457 2009 Live 42.2  
## 10 3 457 2019 Dead 41.7  
## # ... with 35 more rows

A visual inspection reveals that the following trees had DBH decreases during periods where the tree was dead:

* Plot 3, Tree 457
* Plot 10, Tree 409

These trees should be filtered from further adjustment.

# Filter out trees with DBH decrease with dead periods  
dbhDecrease<-  
 dbhDecrease %>%   
 filter(!(Plot==3 & TreeID==457)) %>%   
 filter(!(Plot==10 & TreeID==409))

### Trees with decreasing DBH over live periods

We will make adjustments on a tree by tree basis.

First, reshape the dbhDecrease dataframe to make it easier to see DBH by year, then save it to a .csv file for manual editing.

*I set the following chunk not to run, because we only want to do this once*

dbhEdit1<-  
 dbhDecrease%>%   
 dplyr::select(-Status) %>% # Don't need this column anymore  
 pivot\_wider(names\_from="Year",values\_from="DBH") %>%   
 write.csv(file="dbh\_for\_edit.csv",row.names=FALSE)

One thing to note: the following trees had fallen as of 2019, which means we should not interpolate their 2019 values:

dbhDecrease %>%   
 filter(Year==2019 & is.na(DBH))

## # A tibble: 2 x 7  
## Plot TreeID Species Sector Year Status DBH  
## <fct> <fct> <fct> <fct> <fct> <fct> <dbl>  
## 1 6 111 Bl 3 2019 Live NA  
## 2 10 626 Sx 4 2019 Live NA

dbhEdit2<-  
 readr::read\_csv(here::here("data-raw","dbh\_for\_edit.csv"))  
  
# save this file so we can go back to it  
save(dbhEdit2,file=paste("editedDBH",Sys.Date(),"RData",sep="."))

Also, there are trees that were <7.5cm DBH during the 1992 or 1994 measurements. DBH values for those trees/periods should not be interpolated either.

For values that shouldn’t be interpolated, I entered ‘DoNotReplace’ in the corresponding cell.

Values to be replaced by interpolation had ‘Interpolate’ entered into their cell.

# Load tree data with values to replace  
load(here::here("data-raw","editedDBH.2020-02-27.RData"))  
  
# Show data with replacements  
dbhEdit2

## # A tibble: 29 x 9  
## Plot TreeID Species Sector `1992` `1994` `1997` `2009` `2019`   
## <dbl> <dbl> <chr> <dbl> <chr> <chr> <chr> <chr> <chr>   
## 1 3 279 Sx 1 Interpol~ 43.3 Interpo~ 43.5 43.5   
## 2 3 294 Bl 2 20.5 Interpol~ 20.9 22 23.2   
## 3 3 441 Bl 3 Interpol~ 9.1 9.7 11 14   
## 4 3 459 Sx 3 34.9 Interpol~ 35 35.1 Interpo~  
## 5 5 11 Bl 4 Interpol~ 8.6 9.4 13 15.3   
## 6 6 322 Bl 1 Interpol~ 35 36.5 41 44.7   
## 7 6 219 Bl 2 7.9 8.1 Interpo~ 9.2 9.2   
## 8 6 111 Bl 3 Interpol~ Interpol~ 14.8 14.8 DoNotRe~  
## 9 6 332 Bl 3 21.3 Interpol~ 22 24.9 26.3   
## 10 6 120 Bl 4 26.1 Interpol~ 26.4 28.5 30.2   
## # ... with 19 more rows

Now we can interpolate the missing values

library(imputeTS)  
library(padr)  
library(lubridate)  
library(Hmisc)  
  
# Create a vector of years that are not to be interpolated  
DoNotReplace<-   
 dbhEdit2 %>%   
 pivot\_longer(cols=`1992`:`2019`,names\_to="Year",values\_to="DBH") %>%   
 mutate(TreeID=factor(TreeID)) %>%   
 mutate(Year=as.numeric(Year)) %>%   
 filter(DBH=="DoNotReplace")  
  
# Interpolation  
dbhEdit3<-  
 dbhEdit2 %>%   
 pivot\_longer(cols=`1992`:`2019`,names\_to="Year",values\_to="DBH") %>%   
 mutate(DBH=as.numeric(DBH)) %>%   
 mutate(Year=paste("01-01",Year,sep="-")) %>%   
 mutate(Year=dmy(Year)) %>%   
 mutate(TreeID=factor(TreeID)) %>%   
 group\_by(TreeID) %>%   
 pad() %>%   
 ungroup() %>%   
 mutate(Year=year(Year)) %>%   
 as.data.frame()  
  
# Define function to predict DBH values  
dbhInterp<-function() {  
   
 YearList=data.frame(Year=c(1992,1994,1997,2009,2019))   
   
 x<-  
 dbhEdit3 %>%   
 lm(DBH~Year,data=.) %>%   
 predict(.,newdata=YearList)  
   
 return(data.frame(Year=YearList$Year,DBH.pred=x))  
   
}  
  
# Run function by TreeID  
dbhPred1<-by(dbhEdit3, dbhEdit3[,"TreeID"],  
 function(x) predict(lm(DBH ~ Year, data = x),new=data.frame(Year=c(1992,1994,1997,2009,2019))),simplify=F)  
  
# Bring datasets together   
dbhPred<-data.frame(TreeID=names(dbhPred1),do.call(rbind.data.frame,dbhPred1)) %>% # unpack output from by() function  
 setNames(c("TreeID",c(1992,1994,1997,2009,2019))) %>% # rename columns  
 pivot\_longer(cols=`1992`:`2019`,names\_to="Year",values\_to = "DBH.Pred") %>% # pivot longer   
 mutate(Year=as.numeric(Year)) %>%   
   
 # Join with dbhEdit3  
 left\_join(dbhEdit3,by=c("TreeID","Year")) %>%   
   
 # Join with doNot Replace  
 left\_join(DoNotReplace,by=c("TreeID","Year")) %>%   
   
 # Formatting  
 dplyr::select(Plot=Plot.x,TreeID,Year,DBH=DBH.x,DBH.Pred,DoNotRep=DBH.y) %>%   
 mutate\_at(vars(DBH.Pred),round,1) %>% # round DBH.Pred  
   
 # Find and replace  
 mutate(DBH.New=DBH) %>%   
 mutate(DBH.New=coalesce(DBH.New,DBH.Pred)) %>%   
 mutate(DBH.New=replace(DBH.New,!is.na(DoNotRep),"NA")) %>%   
   
 # Set new DBH column  
 dplyr::select(Plot:Year,DBH=DBH.New) %>%   
 mutate(DBH=as.numeric(DBH))

Now let’s test to make sure that all DBH are now monotonic.

library(MonoInc)  
  
x<-  
 dbhPred %>%   
 drop\_na(DBH)   
   
 mono.x<- by(x$DBH,x$TreeID,monotonic)  
  
 # One tree still has decreasing DBH  
 dbhPred %>%   
 filter(TreeID==as.numeric(names(which(mono.x==FALSE))))

## # A tibble: 5 x 4  
## Plot TreeID Year DBH  
## <dbl> <chr> <dbl> <dbl>  
## 1 12 31 1992 22.4  
## 2 12 31 1994 22.2  
## 3 12 31 1997 22.6  
## 4 12 31 2009 25.7  
## 5 12 31 2019 30.2

1994 value for this tree is less than 1992. This is because the linear function predicts values below the 1992 value. Let’s manually replace the 1994 value with interpolation between 1992 and 1997 values.

dbhPred[dbhPred$TreeID==31 & dbhPred$Year==1994,"DBH"]=22.5