CDW Tally Analysis: D16

Cody Flagg, Courtney Meier

8th January 2017

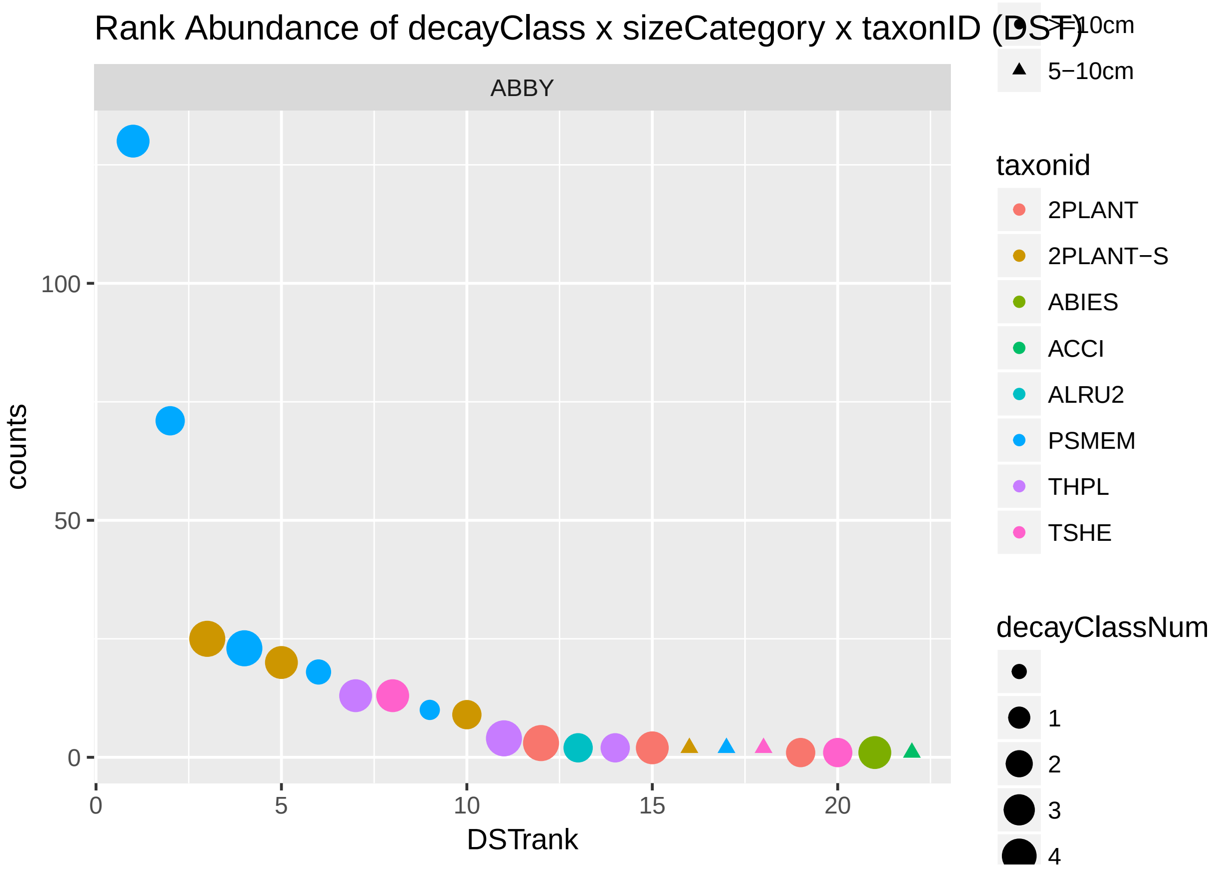
```
## Windows paths
# inpath <- "C:/Users/cflagg/Documents/GitHub/neonPlantSampling/cdw_tally_analysis/D07/"
# outpath <- inpath

##  macOS inputs
inpath <- "~/Documents/workDocuments/gitRepositories/neonPlantSampling/cdw_tally_analysis/fulcrumData_2
domain <- "D16"
outpath <- paste("~/Documents/workDocuments/gitRepositories/neonPlantSampling/cdw_tally_analysis/", doma


## import libraries
library(plyr)
library(dplyr)
library(ggplot2)

## import data
parent_cdw <- read.csv(paste(inpath,"tos_coarse_downed_wood_tally_prod.csv",sep=""), header=TRUE)
child_cdw <- read.csv(paste(inpath,"tos_coarse_downed_wood_tally_prod_per_plot_azimuth_log.csv",sep="")
```

```
## filter out parent fields that are unneeded
parent_cdw_fil <- dplyr::select(parent_cdw,
                            fulcrum_id, domainid, siteid, plotid_parent, tallydate, volumefactor, partic
## filter out child fields that are unneeded
child_cdw_fil <- dplyr::select(child_cdw,
                           fulcrum_parent_id, lidsazimuth, logdiameter, taxonid, decayclass, logid_inges

## join parent to child via parent::fulcrum_id to child::fulcrum_parent_id
cdw <- merge(x = parent_cdw_fil, y = child_cdw_fil, by.x = "fulcrum_id", by.y = "fulcrum_parent_id", all

## filter to the selected domain only
cdw <- dplyr::filter(cdw, domainid == domain)

## create diameter class factor
cdw$diameterClass <- ifelse(cdw$logdiameter >= 10, '>=10cm',
                            ifelse(cdw$logdiameter <5, "2-5cm", "5-10cm"))

## simplify decayclass to numeric value wrapped inside sapply(decayClassNum, "[[", 1) e.g. 'return first
cdw$decayClassNum <- sapply(stringr::str_split(cdw$decayclass, pattern = " "),"[[", 1)


## write file
write.csv(cdw, file = paste(outpath, paste0("merged_",domain,"_cdw_rawdata.csv", sep=""), sep = "/"))
```

siteid	taxonid	decayClassNum	diameterClass	counts	totalLogs	relativeAbundance	cumulativeAbundance
ABBY	PSMEM	4	>=10cm	130	355	36.62	36.62
ABBY	PSMEM	3	>=10cm	71	355	20.00	56.62
ABBY	2PLANT-S	5	>=10cm	25	355	7.04	63.66

siteid	taxonid	decayClassNum	diameterClass	counts	totalLogs	relativeAbundance	cumulativeAbundance
ABBY	PSMEM	5	>=10cm	23	355	6.48	70.14
ABBY	2PLANT-S	4	>=10cm	20	355	5.63	75.77
ABBY	PSMEM	2	>=10cm	18	355	5.07	80.84
ABBY	THPL	4	>=10cm	13	355	3.66	84.50
ABBY	TSHE	4	>=10cm	13	355	3.66	88.16
ABBY	PSMEM	1	>=10cm	10	355	2.82	90.98
ABBY	2PLANT-S	3	>=10cm	9	355	2.54	93.52
ABBY	THPL	5	>=10cm	4	355	1.13	94.65
ABBY	2PLANT	5	>=10cm	3	355	0.85	95.50
ABBY	ALRU2	3	>=10cm	2	355	0.56	96.06
ABBY	THPL	3	>=10cm	2	355	0.56	96.62
ABBY	2PLANT	4	>=10cm	2	355	0.56	97.18
ABBY	2PLANT-S		5-10cm	2	355	0.56	97.74
ABBY	PSMEM		5-10cm	2	355	0.56	98.30
ABBY	TSHE		5-10cm	2	355	0.56	98.86
ABBY	2PLANT	3	>=10cm	1	355	0.28	99.14
ABBY	TSHE	3	>=10cm	1	355	0.28	99.42
ABBY	ABIES	4	>=10cm	1	355	0.28	99.70
ABBY	ACCI		5-10cm	1	355	0.28	99.98

```
## Warning: Using size for a discrete variable is not advised.
```



Code

```
## Windows paths
# inpath <- "C:/Users/cflagg/Documents/GitHub/neonPlantSampling/cdw_tally_analysis/D07/"
# outpath <- inpath

##  macOS inputs
inpath <- "~/Documents/workDocuments/gitRepositories/neonPlantSampling/cdw_tally_analysis/fulcrumData_2
domain <- "D16"
outpath <- paste("~/Documents/workDocuments/gitRepositories/neonPlantSampling/cdw_tally_analysis/", doma


## import libraries
library(plyr)
library(dplyr)
library(ggplot2)

## import data
parent_cdw <- read.csv(paste(inpath,"tos_coarse_downed_wood_tally_prod.csv",sep=""), header=TRUE)
child_cdw <- read.csv(paste(inpath,"tos_coarse_downed_wood_tally_prod_per_plot_azimuth_log.csv",sep="")
```

```
## filter out parent fields that are unneeded
parent_cdw_fil <- dplyr::select(parent_cdw,
                            fulcrum_id, domainid, siteid, plotid_parent, tallydate, volumefactor, partic
## filter out child fields that are unneeded
child_cdw_fil <- dplyr::select(child_cdw,
                           fulcrum_parent_id, lidsazimuth, logdiameter, taxonid, decayclass, logid_inges

## join parent to child via parent::fulcrum_id to child::fulcrum_parent_id
cdw <- merge(x = parent_cdw_fil, y = child_cdw_fil, by.x = "fulcrum_id", by.y = "fulcrum_parent_id", all

## filter to the selected domain only
cdw <- dplyr::filter(cdw, domainid == domain)

## create diameter class factor
cdw$diameterClass <- ifelse(cdw$logdiameter >= 10, '>=10cm',
                            ifelse(cdw$logdiameter <5, "2-5cm", "5-10cm"))

## simplify decayclass to numeric value wrapped inside sapply(decayClassNum, "[[", 1) e.g. 'return first
cdw$decayClassNum <- sapply(stringr::str_split(cdw$decayclass, pattern = " "),"[[", 1)


## write file
write.csv(cdw, file = paste(outpath, paste0("merged_",domain,"_cdw_rawdata.csv", sep=""), sep = "/"))
```

```
#names(cdw)
## table 1 -- count of taxonIDs, report top 10?
## order ascending output by totalTaxa length() will count length of a vector

## Filter out 'blank' transects where TTP = NO
cdw_fil <- dplyr::filter(cdw, targettaxapresent == "Y")
t1 <- ddply(cdw_fil, ~siteid+taxonid+decayClassNum+diameterClass, summarize, counts = length(taxonid))

# an alternative to ddply call, returns similar result but output column name is defaulted
# plyr::count(cdw, c("siteid","taxonid", "decayClassNum", "diameterClass"))

## Sort rows by siteID then by descending number of counts
t1 <- dplyr::arrange(t1, siteid, desc(counts), diameterClass, decayClassNum, taxonid)

## Calculate relative abundances
t1$totalLogs <- nrow(cdw_fil)
t1$relativeAbundance <- round((t1$counts/t1$totalLogs)*100,2)
t1$cumulativeAbundance <- round(cumsum(t1$relativeAbundance),2)

## determine raw ranking by count of DST combo within siteid
t1 <- plyr::ddply(t1, ~siteid, mutate, DSTrank = as.numeric(paste(1:length(counts))), DSTcombo = paste0
t2 <- ddply(cdw, ~siteid+taxonid, summarize, counts = length(taxonid))

##  Determine required sample size per DST combo, and cumulative sample size
t1$sampledDiskNum <- ifelse(t1$diameterClass == ">=10cm", 10,5)
t1$cumulativeDiskNum <- cumsum(t1$sampledDiskNum)
```

```
## species abundance/total abundance
ggplot(data = t1, aes(x = DSTrank, y = counts, color = taxonid, shape = diameterClass, size = decayClas
```