

CDW Tally Analysis: D13

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 <- "D13"
outpath <- paste("~/Documents/workDocuments/gitRepositories/neonPlantSampling/cdw_tally_analysis/", domain)

## 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, parti

## filter out child fields that are unneeded
child_cdw_fil <- dplyr::select(child_cdw,
                                fulcrum_parent_id, lidsazimuth, logdiameter, taxonid, decayclass, logid_ingest

## 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"))

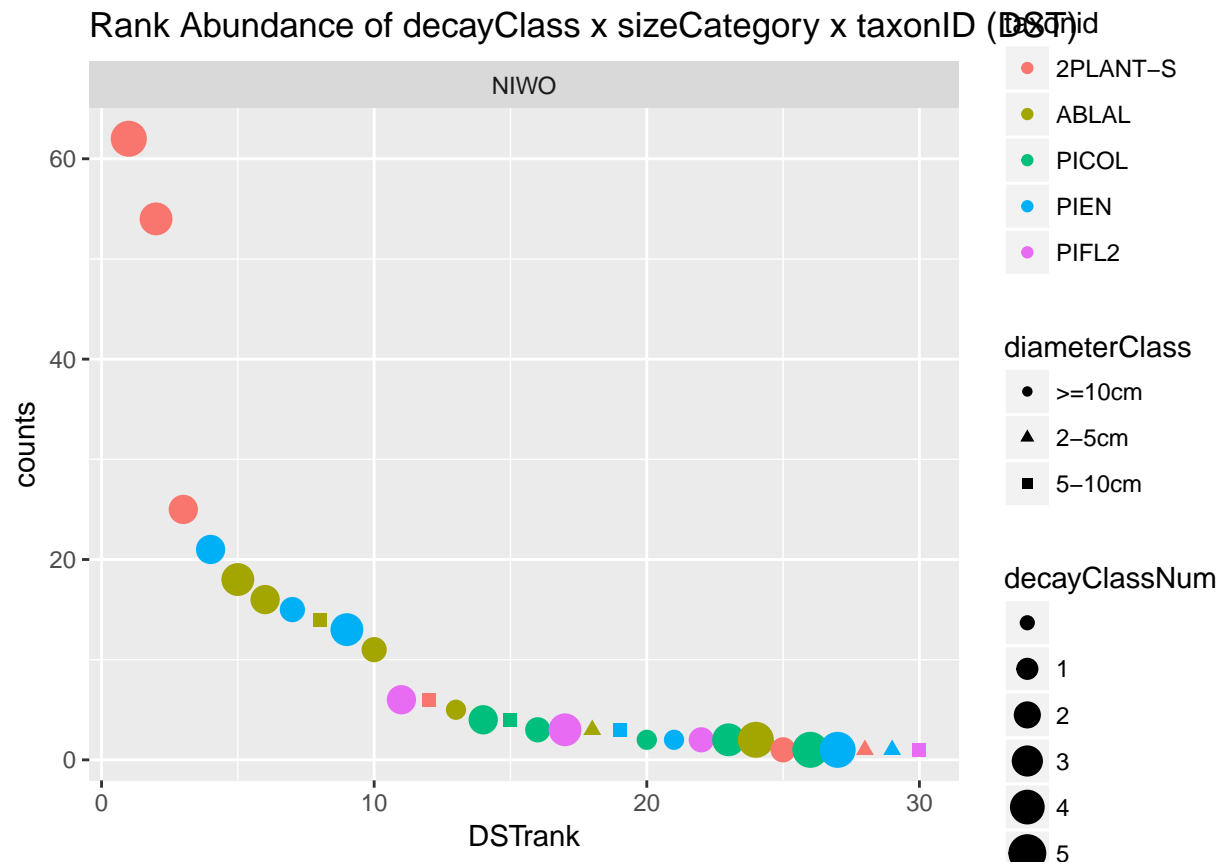
## simplify decayclass to numeric value wrapped inside sapply(decayClassNum, "[", 1) e.g. 'return first
cdw$decayClassNum <- sapply(stringr::str_split(cdw$decayclass, pattern = " "), "[", 1)

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

siteid	taxonid	decayClassNum	diameterClass	counts	totalLogs	relativeAbundance	cumulativeAbundance
NIWO	2PLANT-S	5	>=10cm	62	302	20.53	20.53
NIWO	2PLANT-S	4	>=10cm	54	302	17.88	38.41
NIWO	2PLANT-S	3	>=10cm	25	302	8.28	46.69

siteid	taxonid	decayClassNum	diameterClass	counts	totalLogs	relativeAbundance	cumulativeAbundance
NIWO	PIEN	3	>=10cm	21	302	6.95	53.64
NIWO	ABLAL	4	>=10cm	18	302	5.96	59.60
NIWO	ABLAL	3	>=10cm	16	302	5.30	64.90
NIWO	PIEN	2	>=10cm	15	302	4.97	69.87
NIWO	ABLAL		5-10cm	14	302	4.64	74.51
NIWO	PIEN	4	>=10cm	13	302	4.30	78.81
NIWO	ABLAL	2	>=10cm	11	302	3.64	82.45
NIWO	PIFL2	3	>=10cm	6	302	1.99	84.44
NIWO	2PLANT-S		5-10cm	6	302	1.99	86.43
NIWO	ABLAL	1	>=10cm	5	302	1.66	88.09
NIWO	PICOL	3	>=10cm	4	302	1.32	89.41
NIWO	PICOL		5-10cm	4	302	1.32	90.73
NIWO	PICOL	2	>=10cm	3	302	0.99	91.72
NIWO	PIFL2	4	>=10cm	3	302	0.99	92.71
NIWO	ABLAL		2-5cm	3	302	0.99	93.70
NIWO	PIEN		5-10cm	3	302	0.99	94.69
NIWO	PICOL	1	>=10cm	2	302	0.66	95.35
NIWO	PIEN	1	>=10cm	2	302	0.66	96.01
NIWO	PIFL2	2	>=10cm	2	302	0.66	96.67
NIWO	PICOL	4	>=10cm	2	302	0.66	97.33
NIWO	ABLAL	5	>=10cm	2	302	0.66	97.99
NIWO	2PLANT-S	2	>=10cm	1	302	0.33	98.32
NIWO	PICOL	5	>=10cm	1	302	0.33	98.65
NIWO	PIEN	5	>=10cm	1	302	0.33	98.98
NIWO	2PLANT-S		2-5cm	1	302	0.33	99.31
NIWO	PIEN		2-5cm	1	302	0.33	99.64
NIWO	PIFL2		5-10cm	1	302	0.33	99.97

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 <- "D13"
outpath <- paste("~/Documents/workDocuments/gitRepositories/neonPlantSampling/cdw_tally_analysis/", domain)

## 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, parti

## filter out child fields that are unneeded
```

```

child_cdw_fil <- dplyr::select(child_cdw,
                              fulcrum_parent_id, lidsazimuth, logdiameter, taxonid, decayclass, logid_inge

## 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 = decayClass

```