Assignment 3: Data Exploration

Victoria Thompson

Fall 2024

OVERVIEW

This exercise accompanies the lessons in Environmental Data Analytics on Data Exploration.

Directions

1. Rename this file `<FirstLast>_A03_DataExploration.Rmd` (replacing `<FirstLast>` with your first and last name).
2. Change "Student Name" on line 3 (above) with your name.
3. Work through the steps, **creating code and output** that fulfill each instruction.
4. Assign a useful **name to each code chunk** and include ample **comments** with your code.
5. Be sure to **answer the questions** in this assignment document.
6. When you have completed the assignment, **Knit** the text and code into a single PDF file.
7. After Knitting, submit the completed exercise (PDF file) to the dropbox in Canvas.

TIP: If your code extends past the page when knit, tidy your code by manually inserting line breaks.

TIP: If your code fails to knit, check that no `install.packages()` or `View()` commands exist in your code.

Set up your R session

1. Load necessary packages (tidyverse, lubridate, here), check your current working directory and upload two datasets: the ECOTOX neonicotinoid dataset (ECOTOX_Neonicotinoids_Insects_raw.csv) and the Niwot Ridge NEON dataset for litter and woody debris (NEON_NIWO_Litter_massdata_2018-08_raw.csv). Name these datasets "Neonics" and "Litter", respectively. Be sure to include the subcommand to read strings in as factors.

```
# load packages
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
```

```
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(lubridate)
library(here)
```

```
## here() starts at /home/guest/EDE_Fall2024
```

```
#used getwd() to check working directory

#assign names to datasets and read them in using read.csv; using
#stringsAsFactor subcommand to read strings in as factors

  Neonics <- read.csv(
  file = here('Data','Raw','ECOTOX_Neonicotinoids_Insects_raw.csv'),
  stringsAsFactors = T
)

  Litter <- read.csv(
  file = here('Data','Raw','NEON_NIWO_Litter_massdata_2018-08_raw.csv'),
  stringsAsFactors = T
)
```

Learn about your system

2. The neonicotinoid dataset was collected from the Environmental Protection Agency's ECOTOX Knowledgebase, a database for ecotoxicology research. Neonicotinoids are a class of insecticides used widely in agriculture. The dataset that has been pulled includes all studies published on insects. Why might we be interested in the ecotoxicology of neonicotinoids on insects? Feel free to do a brief internet search if you feel you need more background information.

   Answer: Neonicotinoids, like most other pesticides, can be effective in targeting specific, damaging pests. However, neonicotinoids have the risk of damaging non-target insects, like pollinators (ex. bees) and natural pest managers (ex. spiders). It is important to know both the intended and unintended effects of neonicotinoids on both target and non-target insects.

3. The Niwot Ridge litter and woody debris dataset was collected from the National Ecological Observatory Network, which collectively includes 81 aquatic and terrestrial sites across 20 ecoclimatic domains. 32 of these sites sample forest litter and woody debris, and we will focus on the Niwot Ridge long-term ecological research (LTER) station in Colorado. Why might we be interested in studying litter and woody debris that falls to the ground in forests? Feel free to do a brief internet search if you feel you need more background information.

   Answer: Leaf litter and other woody debris are critical components of nutrient cycling in forest ecosystems. As they decompose, they release essential nutrients back into the soil, supporting plant growth and maintaining ecosystem health. The amount of leaf litter is an important indicator of forest and soil health. Further, litter and woody debris provide habitat and food for a variety of wildlife; when I studied leaf litter at a previous job, it was in context of how it can be habitat for endangered salamanders. Finally, leaf litter is important in active forest management, as its presence in excess can usually indicate when a forest is overdue for a burn (natural or prescribed).

4. How is litter and woody debris sampled as part of the NEON network? Read the NEON_Litterfall_UserGuide.pdf document to learn more. List three pieces of salient information about the sampling methods here:

   Answer: 1. Data collection methods: Debris is sorted into functional groups before measuring. These groups include leaves, needles, twigs/branches, woody material, seeds, flowers, and other/unsorted. 2. Timing of sampling: Ground traps are sampled once a year. Elevated traps are sampled based on the type of vegetation at the site: in deciduous forests, they are sampled every two weeks during fall, while in evergreen forests, they are sampled every one to two months year-round. This means that not all sites are sampled at the same rate, which is something to note in the data. 3. Location of sampling: Sampling locations are selected randomly within existing plots for plant productivity research. Plots are either 40m x 40m or 20m by 20m. There are 1-4 trap pairs per plot, with each pair consisting of (1) a group trap and (2) an elevated trap. Depending on the vegetation in a plot, trap placement within plots may be either targeted or randomized.

Obtain basic summaries of your data (Neonics)

5. What are the dimensions of the dataset?

```
#dim() function retrieves dataset dimensions
dimension_Neonics <- dim(Neonics)
print(dimension_Neonics )
```

```
## [1] 4623   30
```

```
#4623 rows, 30 collumns
```

6. Using the `summary` function on the "Effect" column, determine the most common effects that are studied. Why might these effects specifically be of interest? [Tip: The `sort()` command is useful for listing the values in order of magnitude...]

```
Neonics_summary <- sort(summary(Neonics$Effect)) #summarizes column sorts in
#ascending order
Neonics_summary #prints the summary in the console
```

```
##       Hormone(s)        Histology       Physiology          Cell(s)
##                1                5                7                9
##     Biochemistry     Accumulation     Intoxication    Immunological
##               11               12               12               16
##       Morphology           Growth        Enzyme(s)         Genetics
##               22               38               62               82
##        Avoidance      Development     Reproduction Feeding behavior
##              102              136              197              255
##         Behavior        Mortality       Population
##              360             1493             1803
```

   Answer: The top 5 most common effects of Neonics studied are "Population" (1803), "Mortality" (1493), "Behavior" (360), "Feeding behavior" (255), and "Reproduction" (197). These effects may be specifically of interest because they demonstrate the changes in a species' "life history"- its pattern of survival (i.e. behavior, mortality) and reproduction (i.e. population, reproduction) events. Changes in life history are important in determining he success or failure of a species.

7. Using the `summary` function, determine the six most commonly studied species in the dataset (common name). What do these species have in common, and why might they be of interest over other insects? Feel free to do a brief internet search for more information if needed.[TIP: Explore the help on the `summary()` function, in particular the `maxsum` argument...]

```
Neonics_summary_species <- (summary(Neonics$Species.Common.Name, maxsum = 6))
#summarizes top 6 ("maxsum") species studied
Neonics_summary_species  #prints the summary in the console
```

```
##             Honey Bee        Parasitic Wasp Buff Tailed Bumblebee
##                   667                   285                   183
##   Carniolan Honey Bee            Bumble Bee               (Other)
##                   152                   140                  3196
```

Answer: The species most studied are honey bee (667), parasitic wasp (285), tailed bumblebee (183), california honey bee (152), and bumble bee (140). These species share the fact that they are all significant agricultural pollinators, especially the bumblebees. Because these insects, like the target pests, are likely to be present in agricultural fields, it is important to study the effect of neonics on them. Also, since these pollinators benefit crop growth, it is crucial to know if they are negatively affected, as that may impact crop yields. In short- pollinators are super important, and we need to make sure that these pesticides are not harming them.

8. Concentrations are always a numeric value. What is the class of `Conc.1..Author.` column in the dataset, and why is it not numeric? [Tip: Viewing the dataframe may be helpful...]

```
class(Neonics$Conc.1..Author.) #extract the class of 'Conc.1..Author' within
```

```
## [1] "factor"
```

```
# the Neonics dataframe
# the class is 'factor'
```

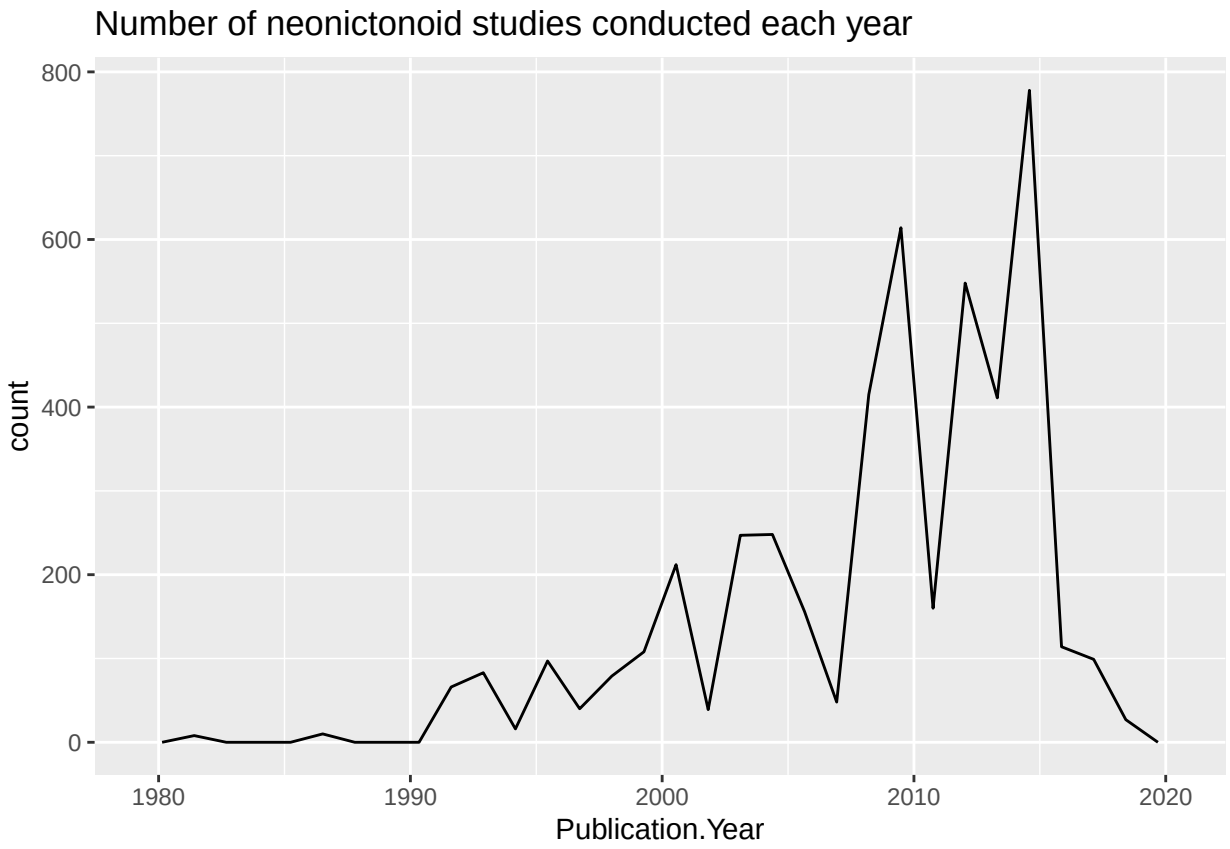
Answer: There are some non-number values stored in the 'Conc.1..Author' column. There are some "/" and "NR" values included, which makes it impossible for the data in this column to be stored in a numeric class.

Explore your data graphically (Neonics)

9. Using `geom_freqpoly`, generate a plot of the number of studies conducted by publication year.

```
#using gg plot to set up a plot with Publication.Year on the x axis and count on
#the y axis
ggplot(Neonics,aes(x=Publication.Year)) +
  geom_freqpoly() +
  ggtitle("Number of neonictonoid studies conducted each year")
```

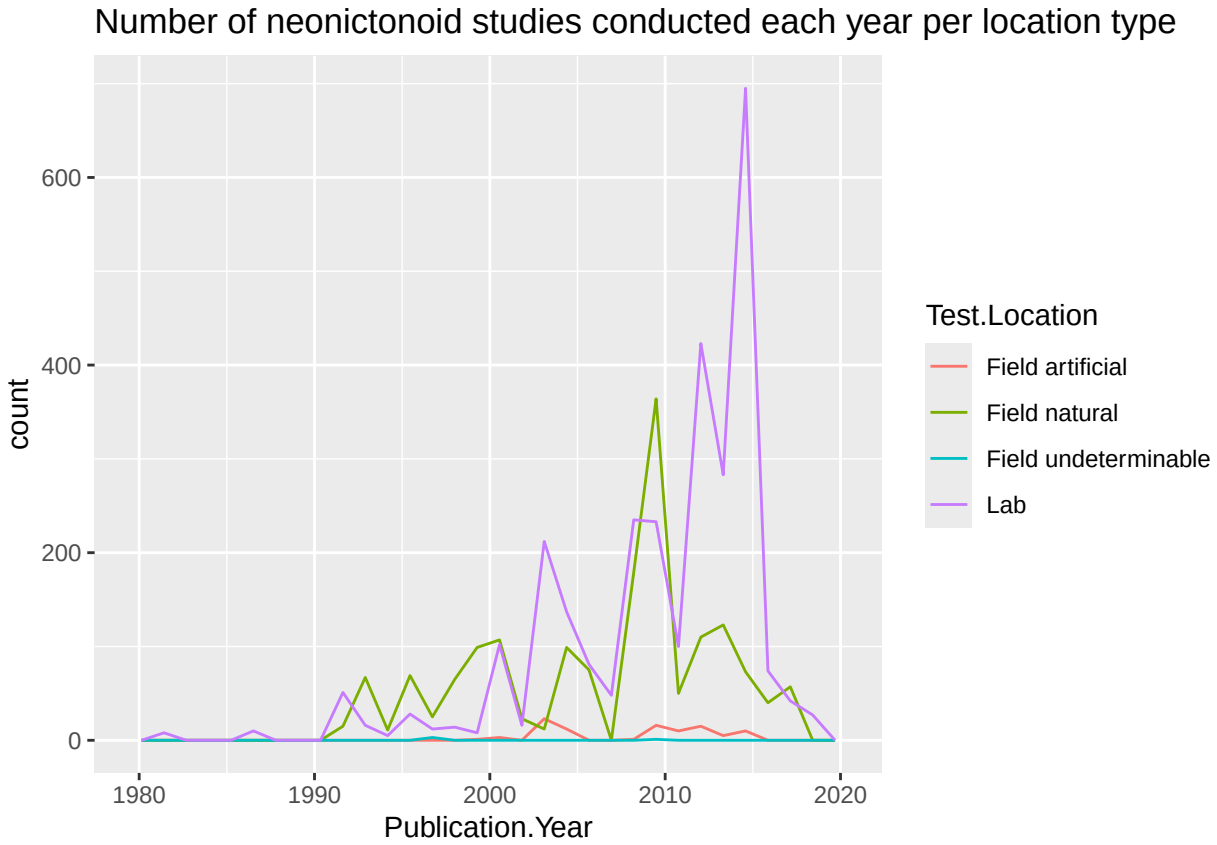
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



10. Reproduce the same graph but now add a color aesthetic so that different Test.Location are displayed as different colors.

```
#using color function within the aesthetic ("aes") command. Changing color based
#on the value for Test.Location.
ggplot(Neonics, aes(x = Publication.Year, color = Test.Location)) +
  geom_freqpoly() +
  ggtitle("Number of neonictonoid studies conducted each year per location type")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



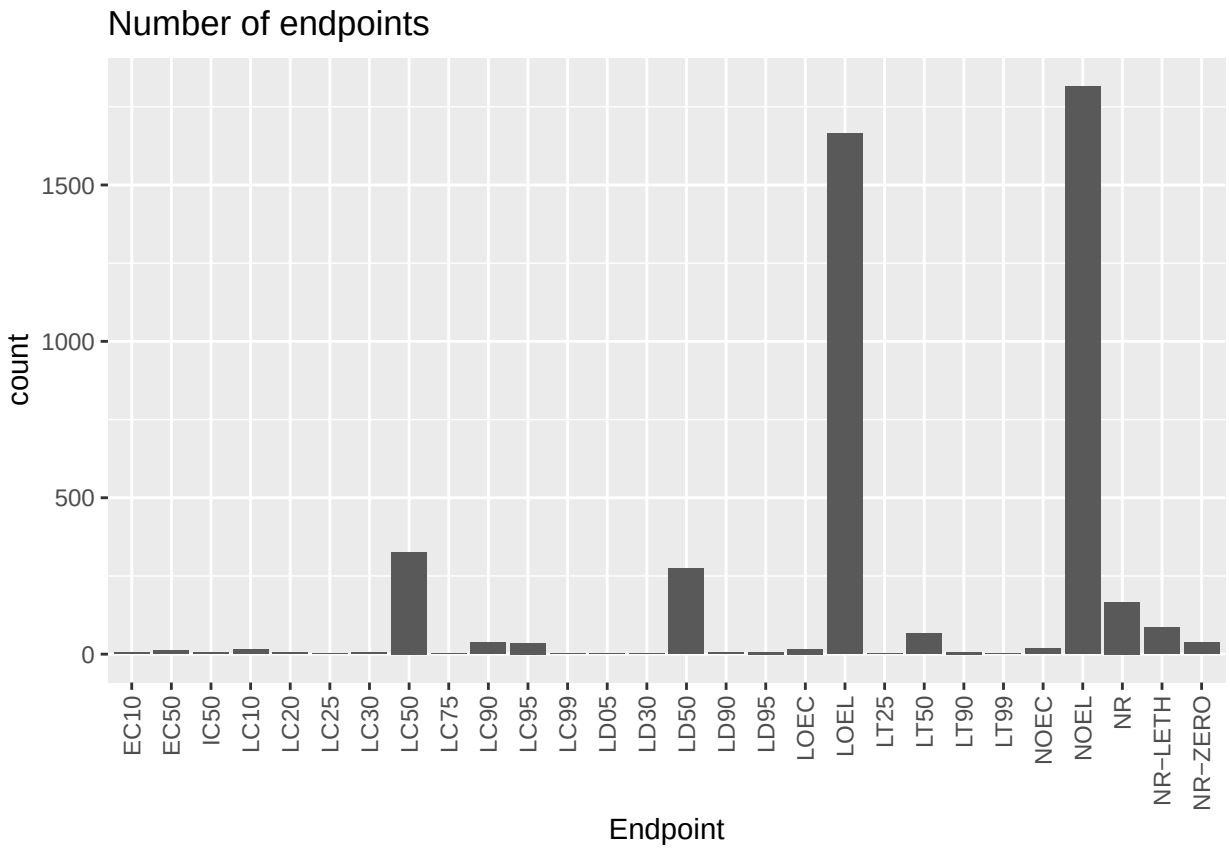
Interpret this graph. What are the most common test locations, and do they differ over time?

> Answer: The most common test locations are in the lab and in natural fields. There is variation in terms of which is higher than the other in a given year. Both increase over time, but after 2010, the number of lab-based studies is consistently and significantly higher than those in the natural fields. After a high point around 2015, the overall number of tests drops, with lab and natural studies evening out as they approach 2020.

11. Create a bar graph of Endpoint counts. What are the two most common end points, and how are they defined? Consult the ECOTOX_CodeAppendix for more information.

[**TIP**: Add `theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))` to the end of your plot command to rotate and align the X-axis labels...]

```
ggplot(Neonics, aes(x = Endpoint)) +
  geom_bar() +
  ggtitle("Number of endpoints") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



Answer: The most common end points are NOEL and LOEL. NOEL stands for "No-observable-effect-level", where the highest concentration of toxin does not produce a significant difference producing effects not significantly different from a control. LOEL stands for "lowest-observable-effect-level", which is when the lowest dose of toxin produces effects that are significantly different from the controls. In this way, these two endpoints are nearly opposites in terms of their scientific implications.

Explore your data (Litter)

12. Determine the class of collectDate. Is it a date? If not, change to a date and confirm the new class of the variable. Using the `unique` function, determine which dates litter was sampled in August 2018.

```
class(Litter$collectDate)
```

```
## [1] "factor"
```

```
# class is currently a factor

Litter$collectDate <- ymd(Litter$collectDate) # using the lubridate function for
#"year - month- day"
class(Litter$collectDate)
```

```
## [1] "Date"
```

```
# class is now a date

unique(Litter$collectDate)
```

```
## [1] "2018-08-02" "2018-08-30"
```

```
# data was collected on the 2nd and 30th of August
```

13. Using the `unique` function, determine how many different plots were sampled at Niwot Ridge. How is the information obtained from `unique` different from that obtained from `summary`?

```
unique (Litter$plotID)
```

```
##  [1] NIWO_061 NIWO_064 NIWO_067 NIWO_040 NIWO_041 NIWO_063 NIWO_047 NIWO_051
##  [9] NIWO_058 NIWO_046 NIWO_062 NIWO_057
## 12 Levels: NIWO_040 NIWO_041 NIWO_046 NIWO_047 NIWO_051 NIWO_057 ... NIWO_067
```

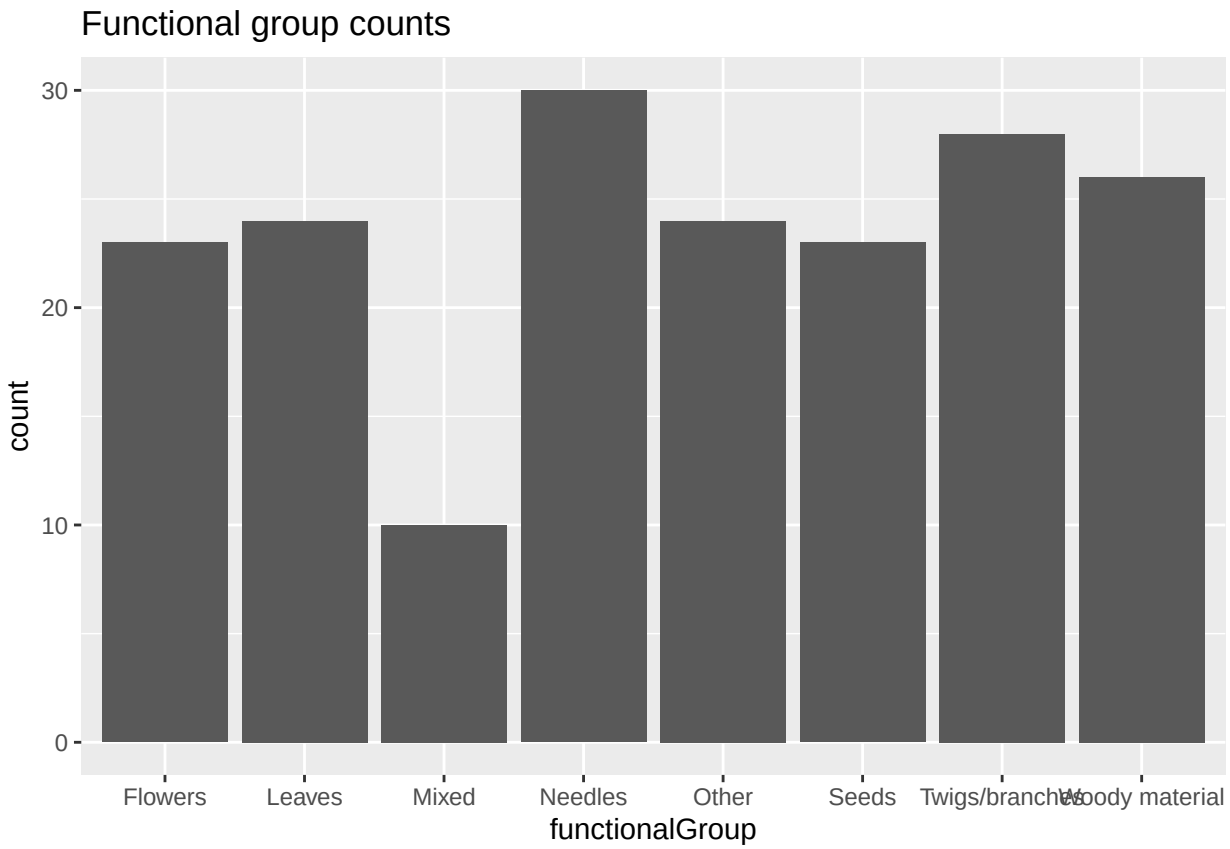
```
summary (Litter$plotID)
```

```
## NIWO_040 NIWO_041 NIWO_046 NIWO_047 NIWO_051 NIWO_057 NIWO_058 NIWO_061
##       20       19       18       15       14        8       16       17
## NIWO_062 NIWO_063 NIWO_064 NIWO_067
##       14       14       16       17
```

Answer: "Unique" gives the character name of each unique plot. "Summary" gives the same, but also the number of times each unique value appears in the data.

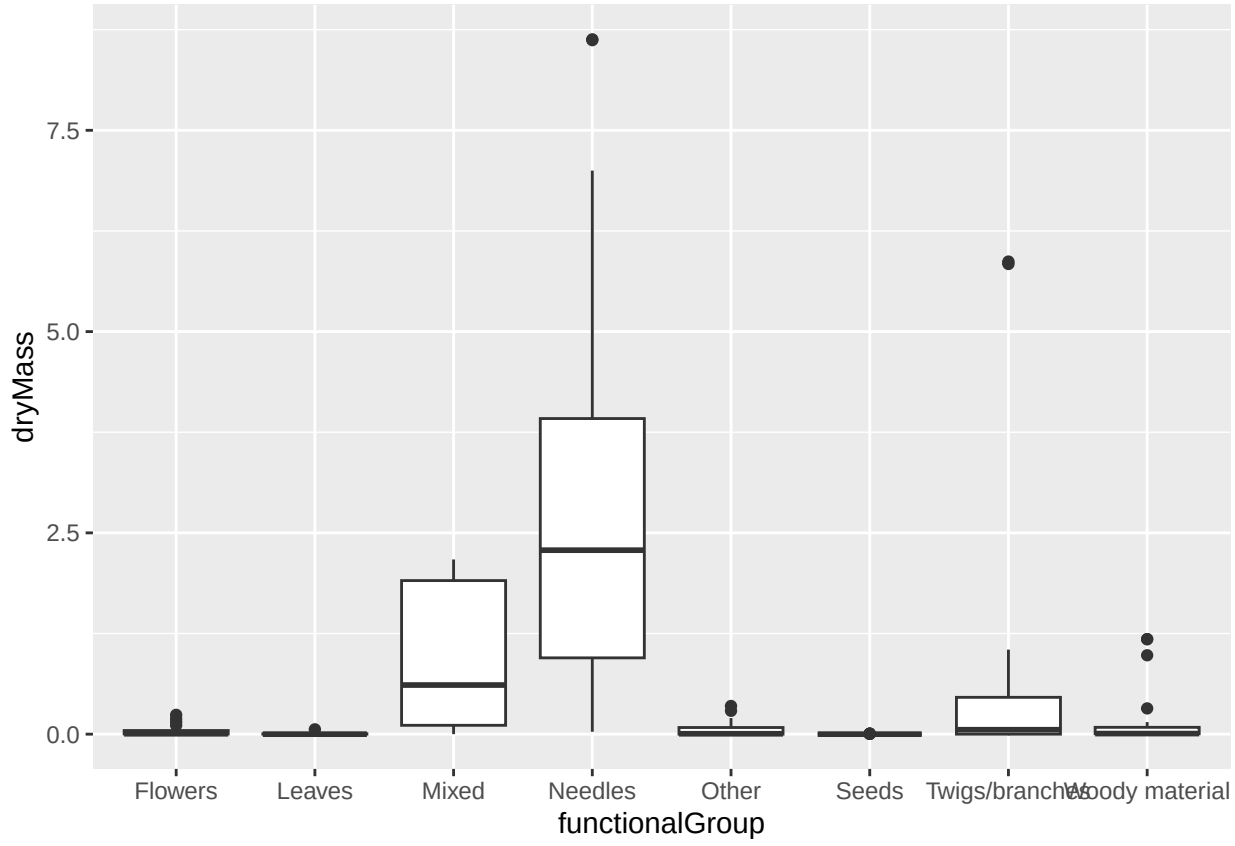
14. Create a bar graph of functionalGroup counts. This shows you what type of litter is collected at the Niwot Ridge sites. Notice that litter types are fairly equally distributed across the Niwot Ridge sites.

```
ggplot(Litter, aes(x = functionalGroup)) +
  geom_bar() +
  ggtitle("Functional group counts")
```

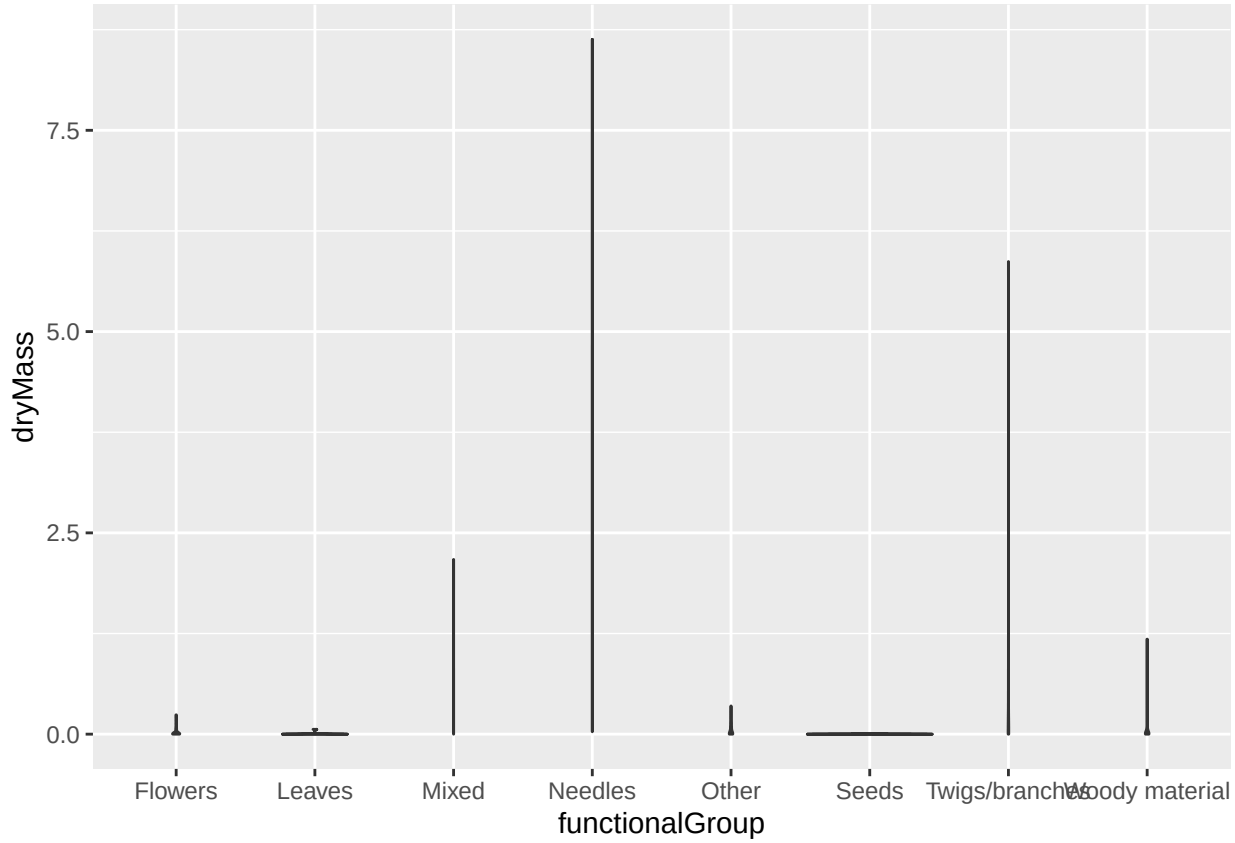


15. Using `geom_boxplot` and `geom_violin`, create a boxplot and a violin plot of dryMass by functionalGroup.

```
# dry mass boxplot
ggplot(Litter) +
  geom_boxplot(aes(x = functionalGroup, y = dryMass))
```




```
# dry mass violin plot
ggplot(Litter) +
  geom_violin(aes(x = functionalGroup, dryMass),
              draw_quantiles = c(0.25, 0.5, 0.75))
```



Why is the boxplot a more effective visualization option than the violin plot in this case?

> Answer: A boxplot is more effective in this case because it very clearly demarcates outliers, as well as where the majority of the data lies for each group. The violin plot does not provide any useful data, as the widths lack meaning in isolation and the outliers are not clearly shown.

What type(s) of litter tend to have the highest biomass at these sites?

> Answer: Needles, mixed litter, and twigs/branches have the highest biomass at these sites.