Assignment 3: Data Exploration

Brianna Karson

Spring 2026

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. [NEW] 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 Canvas.
8. Initial here to acknowledge that you did not use AI in completing this assignment, except where expressly allowed: BK

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

TIP: If your code fails to knit, check: * That no `install.packages()` or `View()` commands exist in your code. * That you are not displaying the entire contents of a large dataframe in your code.

Set up your R session

1. Load necessary packages (tidyverse, here), check your current working directory and import 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: * Use the `here()` package in specifying the paths to your datasets * Include the appropriate subcommand to read in character based columns as factors

```
#1

#loading packages
library(tidyverse)
library(here)
library(lubridate)
getwd()
```

```
## [1] "/home/guest/EDE_Spring2026"
```

```
here()
```

```
## [1] "/home/guest/EDE_Spring2026"
```

```
#loading data
Neonics <- read.csv(
  file = here("./Data/Raw/ECOTOX_Neonicotinoids_Insects_raw.csv"),
    stringsAsFactors = TRUE)

Litter <- read.csv(
  file = here(
    "./Data/Raw/NEON_NIWO_Litter_massdata_2018-08_raw.csv"),
  stringsAsFactors = TRUE)
```

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. (AI is allowed here, but put answers in your own words.)

   Answer: It is useful to conduct analyses on neonicotinoids in order to understand how they may negatively affect non-target species such as pollinators, as well as the surrounding environment. Depending on their persistence in the environment, these chemicals may cause cumulative effects on the food chain that negatively impact the overall health and biodiversity of the system.

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. (AI is allowed here, but put answers in your own words.)

   Answer: Data surrounding forest litter and woody debris is useful to understand carbon and nutrient cycling. Understanding the composition of these materials informs scientific understanding of the biodiversity of the area, and can also inform fire management tactics.

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.Sampling occurs in randomly selected tower plots that contain woody vegetation >2m tall. 2. Within the plots, traps may be placed randomly or strategically, depending on observed vegetation. 3. Ground traps are only sampled 1x/year, but elevated traps may be sampled more frequently depending on forest type.

Obtain basic summaries of your data (Neonics)

5. What are the dimensions of the dataset?

```
#5 #dimensions of the dataset
dim(Neonics)
```

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

```
#4623 rows #30 columns
```

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

```
#6 #summarizing effects
neonics_effect <- summary(Neonics$Effect)
sort(neonics_effect, decreasing = TRUE)
```

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

Question: Which two effects stand out as the most studied? Can you guess why these effects might specifically be of interest?

Answer: The most commonly studied effects of neonics are population and mortality effects. The next most studied effects pertain to general behavior and feeding behavior.

7. Using the `summary` function, determine the six most commonly studied species in the dataset (common name).[TIP: Explore the help on the `summary()` function, in particular the `maxsum` argument...]

```
#7 #summarizing most studied species
summary(Neonics$Species.Common.Name, maxsum = 7)
```

```
##             Honey Bee        Parasitic Wasp Buff Tailed Bumblebee
##                   667                   285                   183
##   Carniolan Honey Bee            Bumble Bee      Italian Honeybee
##                   152                   140                   113
##               (Other)
##                  3083
```

Question: What do these species have in common? Why might they be of interest over other insects?

> Answer:The most commonly studied species are (in order of most to least studied), honey bee, parasitic wasp, buff tailed bumblebee, carniolan honey bee, bumble bee, and italian bumble bee. All of these species are pollinators, making them of crucial importance to crop growth. If neonics have a negative effect on the population or mortality rates of these species, there are likely to be negative impacts on agricultural yields as well.

8. The `Conc.1..Author` column, which lists the concentration of the neonicitoid dose, should include numeric values. 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...]

```
#8 #determining class type
class(Neonics$Conc.1..Author.)
```

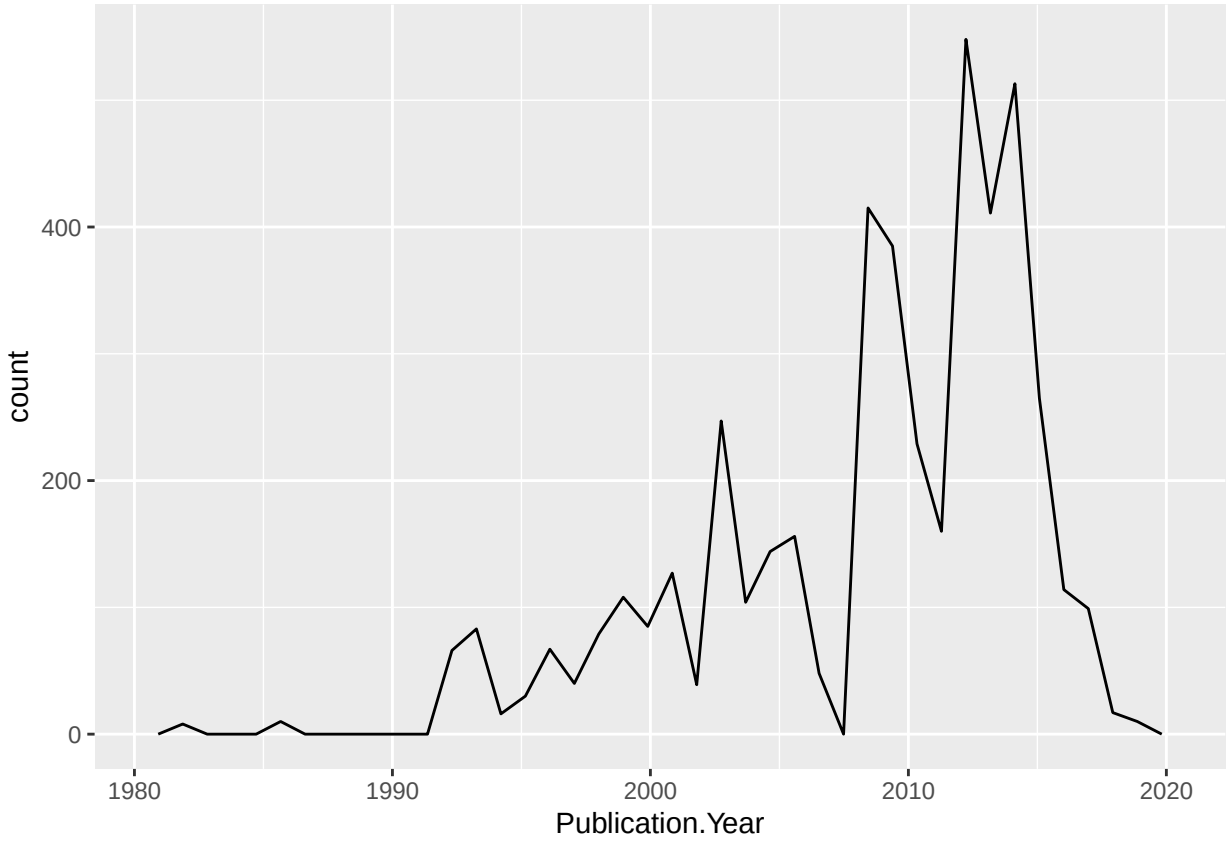
```
## [1] "factor"
```

> Answer: The Conc.1...Author column uses the factor data class. This is likely the case because many of the data values contain >,<, and / symbols that indicate a degree of certainty, or (in the case of the /) that the value is part of a concentration ratio. To preserve the data with these details intact, it must be listed as a factor rather than numeric variable.

Explore your data graphically (Neonics)

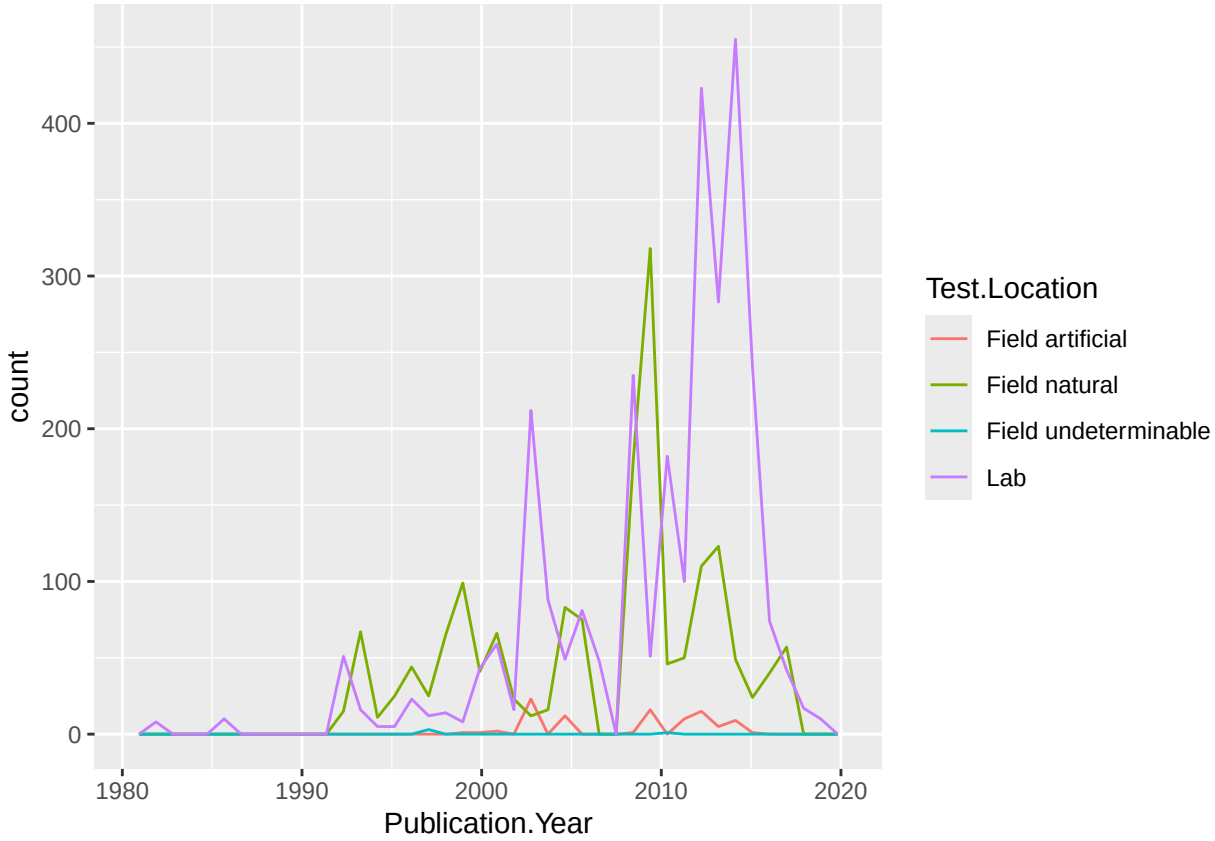
9. Using `geom_freqpoly`, generate a plot of the number of studies conducted by publication year.

```
#9 #plot of studies by year
ggplot(Neonics) +
  geom_freqpoly(aes(x = Publication.Year), bins=40)
```



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

```
#10 #plot of studies by year #adding color
ggplot(Neonics) +
  geom_freqpoly(aes(x = Publication.Year, color=Test.Location), bins=40)
```



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

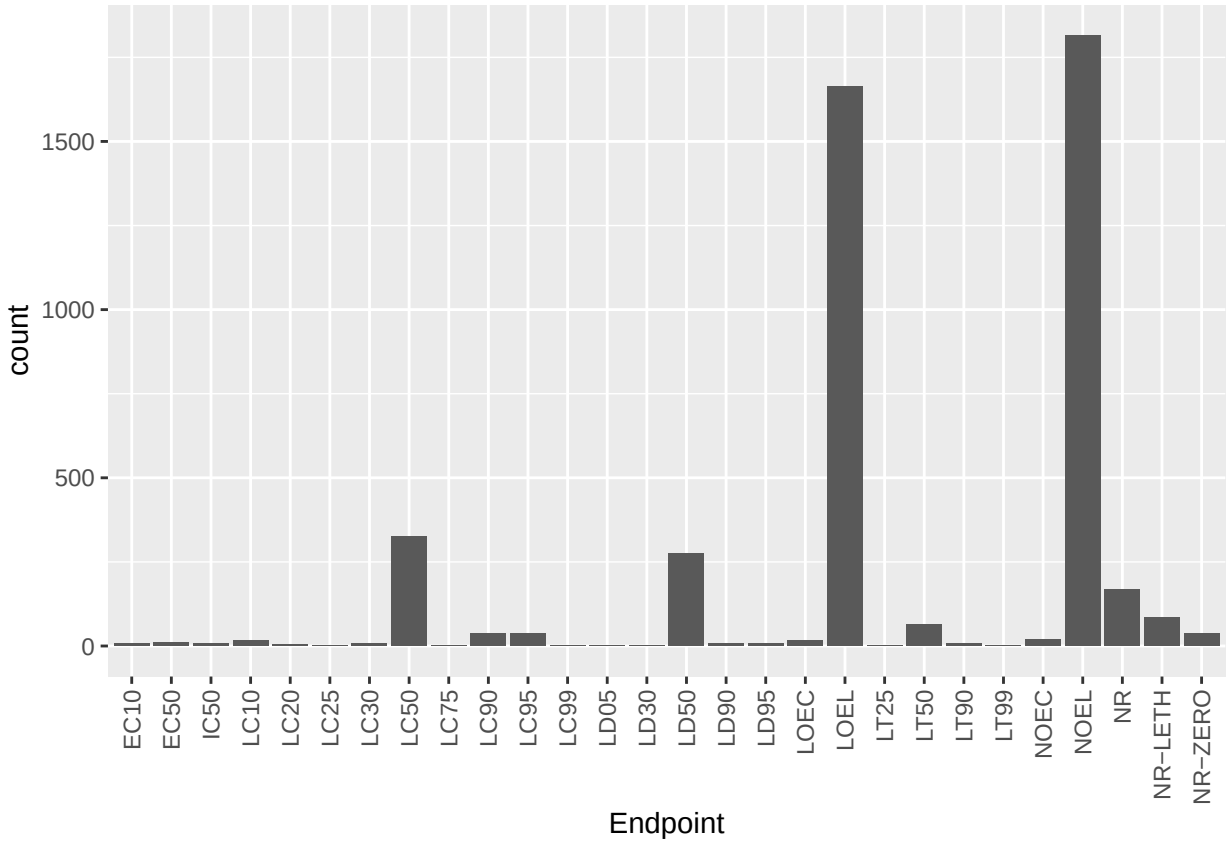
> Answer: Lab testing locations have become the most common test locations since the early 2000s. Previously, Field natural test locations were more popular, and they remain the second most popular testing location as of 2020.

11. Create a bar graph of Endpoint counts.

[**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…]

```
#11 #endpoint bar graph

ggplot(data = Neonics, aes(x = Endpoint)) + geom_bar() +
       theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



What are the two most common end points, and how are they defined? Consult the ECOTOX_CodeAppendix (p.721) for more information.

> Answer: The two most common end points are LOEL and NOEL. LOEL is used in the terrestrial database, and is defined as the "lowest observable effect level". In other words, it is the lowest concentration that has been observed to produce results that are significantly different from control results. NOEL is also used in the terrestrial database, and is the "no observable effect level," referring to the highest observable concentration value that produces responses not significantly different from control results.

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.

```
#12

#determining class
class(Litter$collectDate) #class is factor
```

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

```
#changing to a date
Litter$collectDate <- ymd(Litter$collectDate)
class(Litter$collectDate) #check that it worked
```

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

```
#using unique function
unique(Litter$collectDate)
```

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

```
    #litter was collected 08/02/2018 and 08/30/2018
```

13. Using the `unique` function, list the different `plotIDs` sampled at Niwot Ridge.

```
#13 #Unique plot IDs at Niwot Ridge
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
```

```
    #lists 12 unique values: NIWO_061, NIWO_064, NIWO_067, NIWO_040, NIWO_041, NIWO_063, NIWO_047,NIWO_

summary(Litter$plotID) #also lists how many observations occur at each level
```

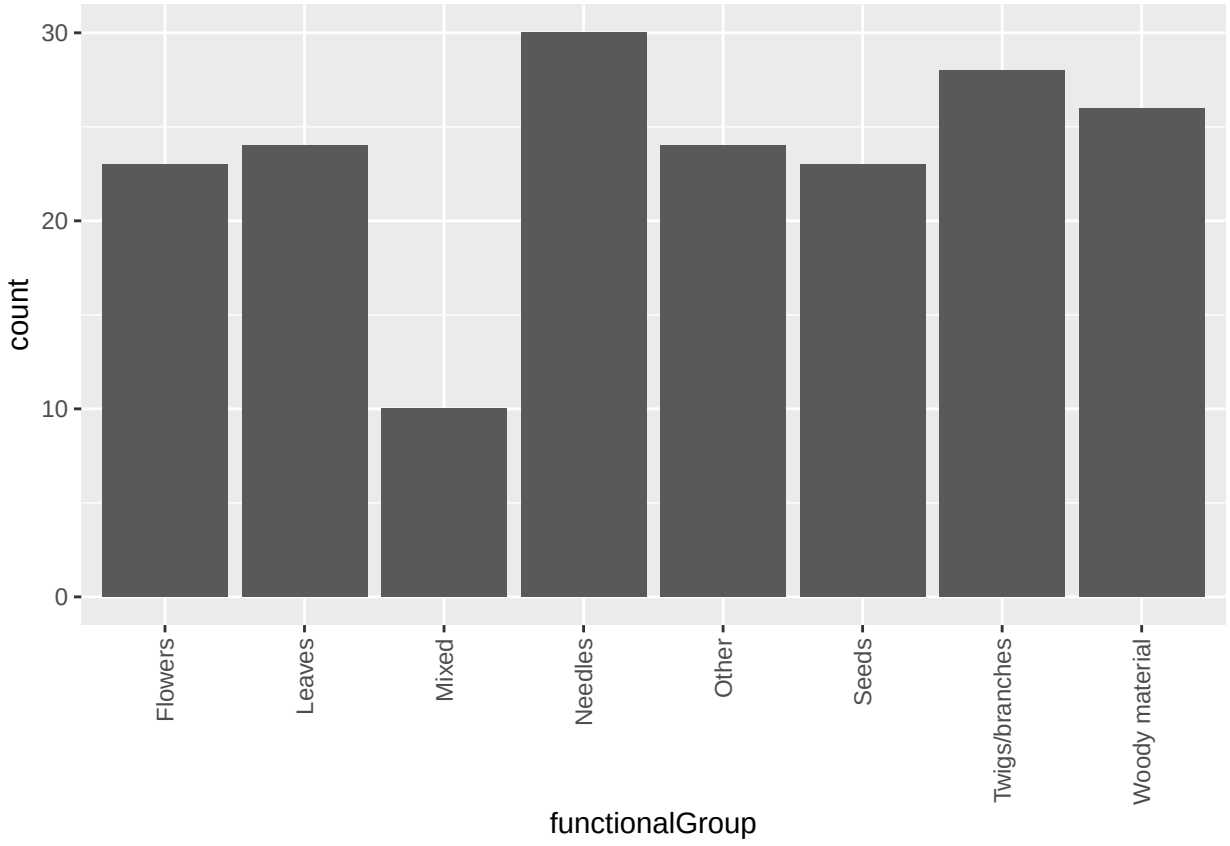
```
## NIWO_040 NIWO_041 NIWO_046 NIWO_047 NIWO_051 NIWO_057 NIWO_058 NIWO_061
##       20       19       18       15       14        8       16       17
## NIWO_062 NIWO_063 NIWO_064 NIWO_067
##       14       14       16       17
```

How is the information obtained from `unique` different from that obtained from `summary`?

> Answer: The 'unique' function provides a list of each unique level within this factor variable. The 'summary' function also lists each unique level, but additionally provides a count of how many observations occur of each level.

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.

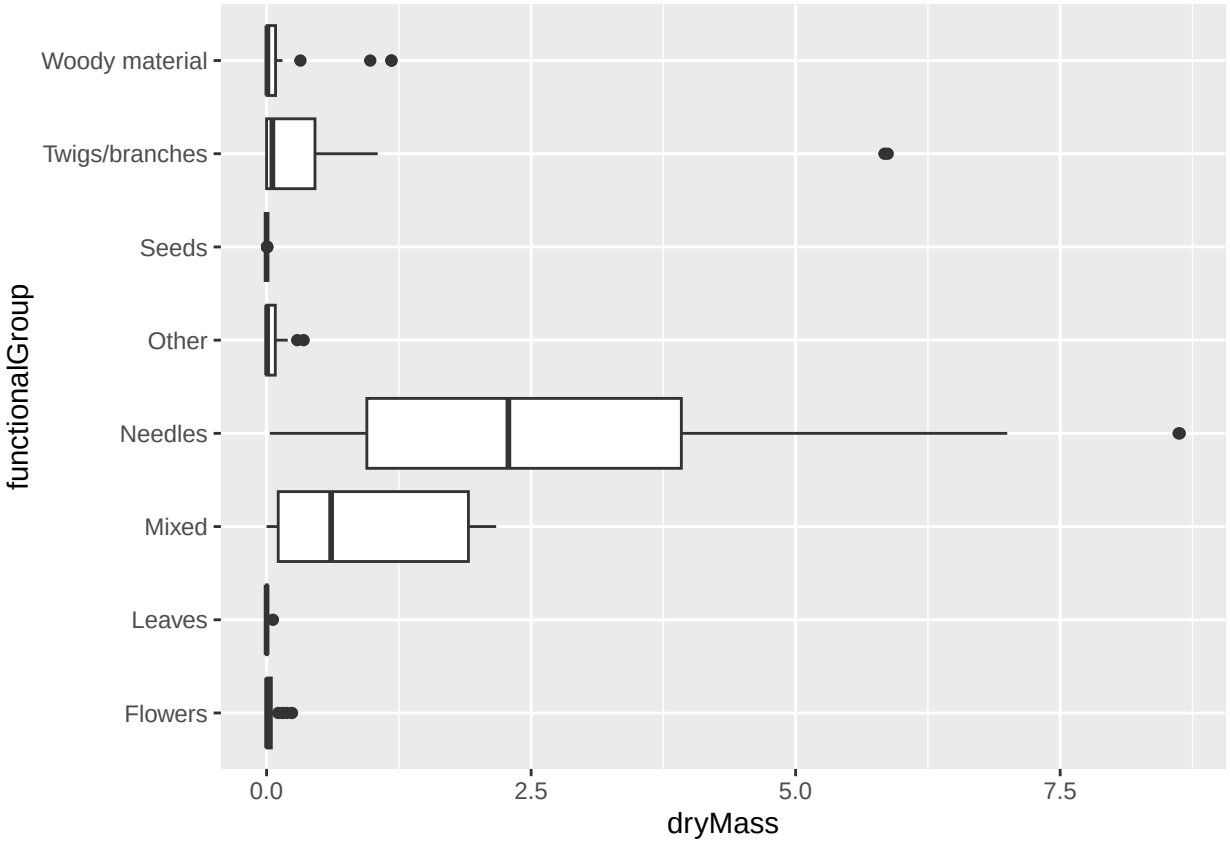
```
#14 #bar group of functionalGroup count
ggplot(data = Litter, aes(x = functionalGroup)) + geom_bar() +
       theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```



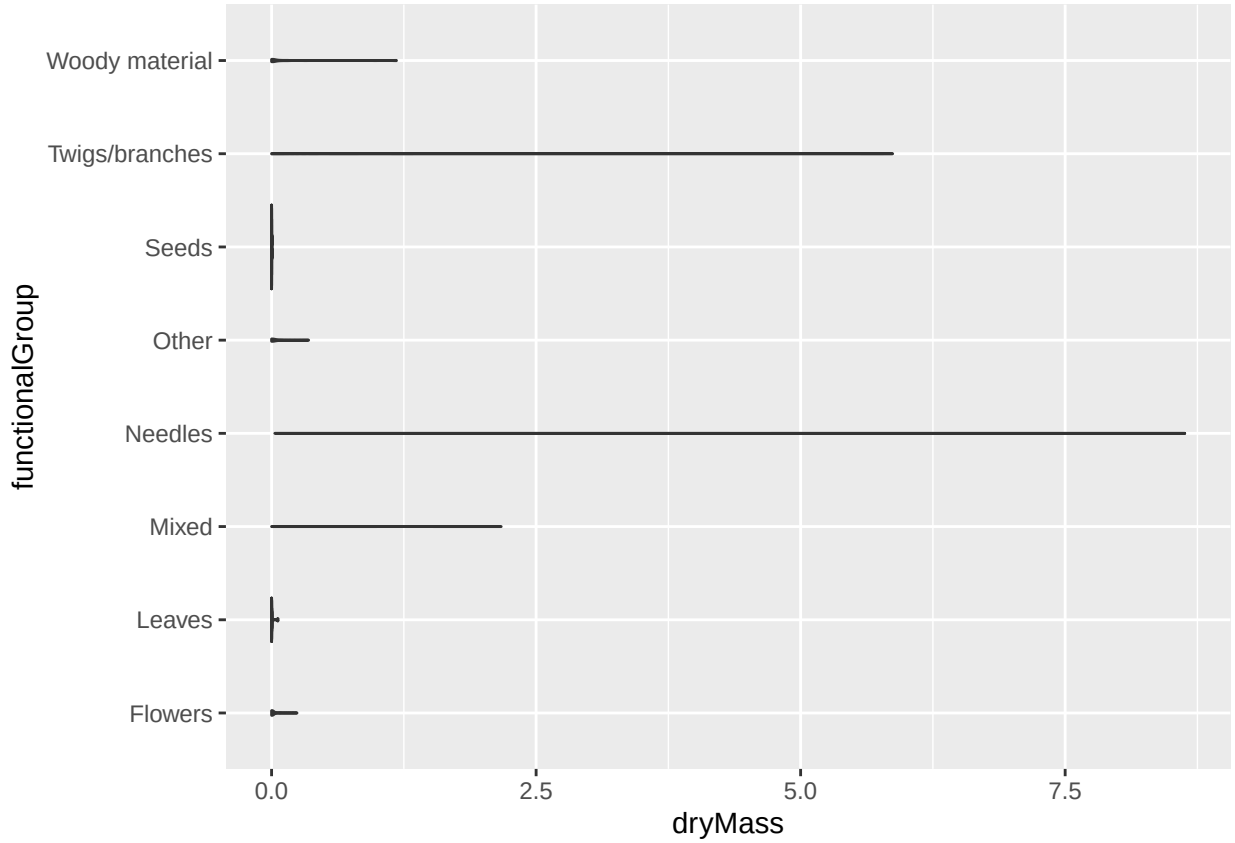
15. Using `geom_boxplot` and `geom_violin`, create a boxplot and a violin plot of `dryMass` by `functionalGroup`.

```
#15

#box plot
ggplot(data = Litter, aes(x = dryMass, y = functionalGroup)) + geom_boxplot()
```




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




```
#context to answer next question
summary(Litter$functionalGroup)
```

```
##        Flowers         Leaves          Mixed        Needles          Other
##             23             24             10             30             24
##          Seeds Twigs/branches Woody material
##             23             28             26
```

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

> Answer: In this case, the box plot is a more effective visualization option than the violin plot because of the distribution and variance of the data. Since there is limited dryMass data for each functional group (30 or less observations) taken on only two days, it makes sense that there is not much variation in the distribution of the data to produce useful violin plot.

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

> Answer: The needle and mixed litter types tend to have the highest biomass at these sites.