Assignment 3: Data Exploration

Zhenghao Lin

Fall 2023

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 Sakai.

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. Check your working directory, load necessary packages (tidyverse, lubridate), 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.

```
#set up working directory
getwd()
```

```
## [1] "/Users/lzh/Desktop/EDE_Fall2023"
```

```
#Library Installation
#I encountered an error in `contrib.url()`:
#! trying to use CRAN without setting a mirror To let the pdf file successfully
#To successfully generate the pdf, I commented all the pacakge installation
#install.packages("tidyverse")
```

```
#install.packages("lubridate")
#install.packages("packagename")
#install.packages("ggplot2")
#install.packages("dplyr")

#library
library("tidyverse")
library("lubridate")
library("ggplot2")
library("dplyr")

#Import datasets
Neonics =
  read.csv("/Users/lzh/Desktop/EDE_Fall2023/Data/Raw/ECOTOX_Neonicotinoids_Insects_raw.csv", as.is = FA
Litter = read.csv("/Users/lzh/Desktop/EDE_Fall2023/Data/Raw/NEON_NIWO_Litter_massdata_2018-08_raw.csv",
```

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: As one of the most widely used insecticides, although neonicotinoid can help farmers to protect crops from pests, it can impose significantly negative impact to pollinators, human health, plants, soil,and water, and eventually hurt the entire ecosystem. Research on ecotoxicology of neoincotinoids can beneficial for EPA to assess and mitigate the risks of such insecticides and come up with suitable regulations.

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:Studying litter and woody debris that falls to the ground in forests can help us understand the biodiversity, carbon budgets, and nutrient cycling of the ecosystems. It's also a source of energy for aquatic ecosystems, providing habitat for terrestrial and aquatic organisms.

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. The dry weight of litterfall and fine woody debris collected from litter traps by plant functional type. 2. Litter and fine woody debris sampling is executed at terrestrial NEON sites that contain woody vegetation >2m tall. 3. Ground traps are sampled once per year.

Obtain basic summaries of your data (Neonics)

5. What are the dimensions of the dataset?

```r
#dimension of dataset Neonics
dim(Neonics)
```

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

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?

```r
#summary of the "Effect" column in dataset Neonics
summary(Neonics$Effect)
```

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

```r
# count the occurrences of each effect
effect_counts <- table(Neonics$Effect)

# most studied effect
most_studied_effect <- names(effect_counts)[which.max(effect_counts)]
most_studied_effect
```

```
## [1] "Population"
```

```r
max(effect_counts)
```

```
## [1] 1803
```

Answer: Based on the EPA official website, the ECOTOXicology Knowledgebase (ECOTOX) is a source for locating single chemical toxicity data for aquatic life, terrestrial plants and wildlife. What's more, the study objects, which are in the species.group sections, are all insects&spiders. By studying the effect of the chemical toxicity on the species's population, we can tell whether the existance of such chemical can cause harm to insects' population, not just to the pests but also to other insects, and how much harm can it do.

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: The `sort()` command can sort the output of the summary command...]

```r
#summary of column Species.Common.Name in dataset Neonics
species_summary <- summary(Neonics$Species.Common.Name)

#Sort the summary of Species.Common.Name in descending order
sorted_species_summary <- sort(species_summary, decreasing = TRUE)

#First six of the sorted Species.Common.Name
head(sorted_species_summary, 6)
```

```
##                (Other)             Honey Bee        Parasitic Wasp
##                    670                   667                   285
## Buff Tailed Bumblebee   Carniolan Honey Bee            Bumble Bee
##                    183                   152                   140
```

Answer: While the (other) options can includes various types of insects, most of other top-ranked insects in terms of the frequency in Species.Common.Name, they are all bees and wasps, which are pollinators that are beneficial to agriculture. The researchers in EPA want to study those pollinators because neonicotinoids have received significant attention due to concerns about their potential contribution to the decline in pollinator populations, particularly bees. Pollinators are essential for the reproduction of many plants, including many crops.

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?

```r
#class of column `Conc.1..Author.` in dataset Neonics
class(Neonics$Conc.1..Author.)
```

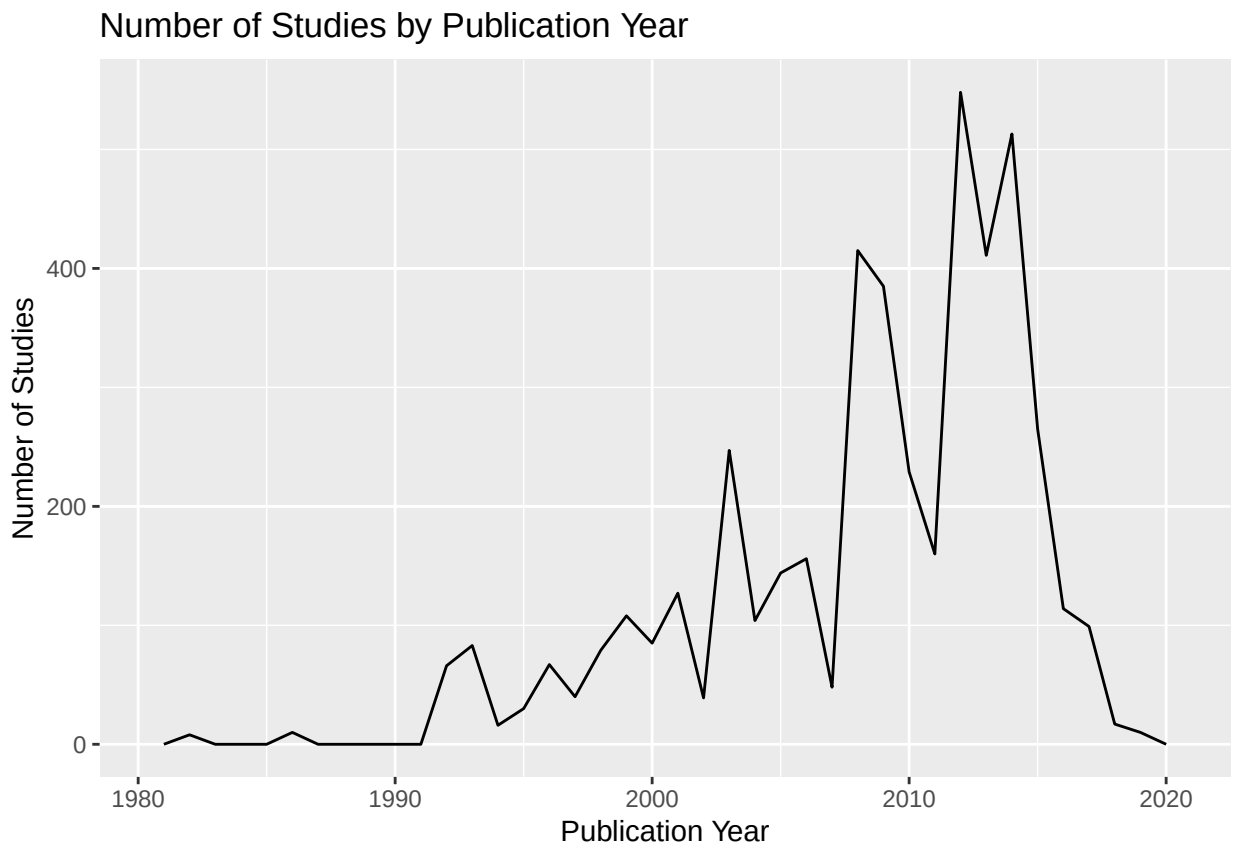
```
## [1] "factor"
```

Answer: The class of `Conc.1..Author.` is factor. It's factor because at the beginning of the dataset import, the question asked me to interpret the dataset as factor, so for the as.is part in the function read.csvm, I typed FALSE, which lead me to have factor as the return of class(Neonics$Conc.1..Author.).

Explore your data graphically (Neonics)

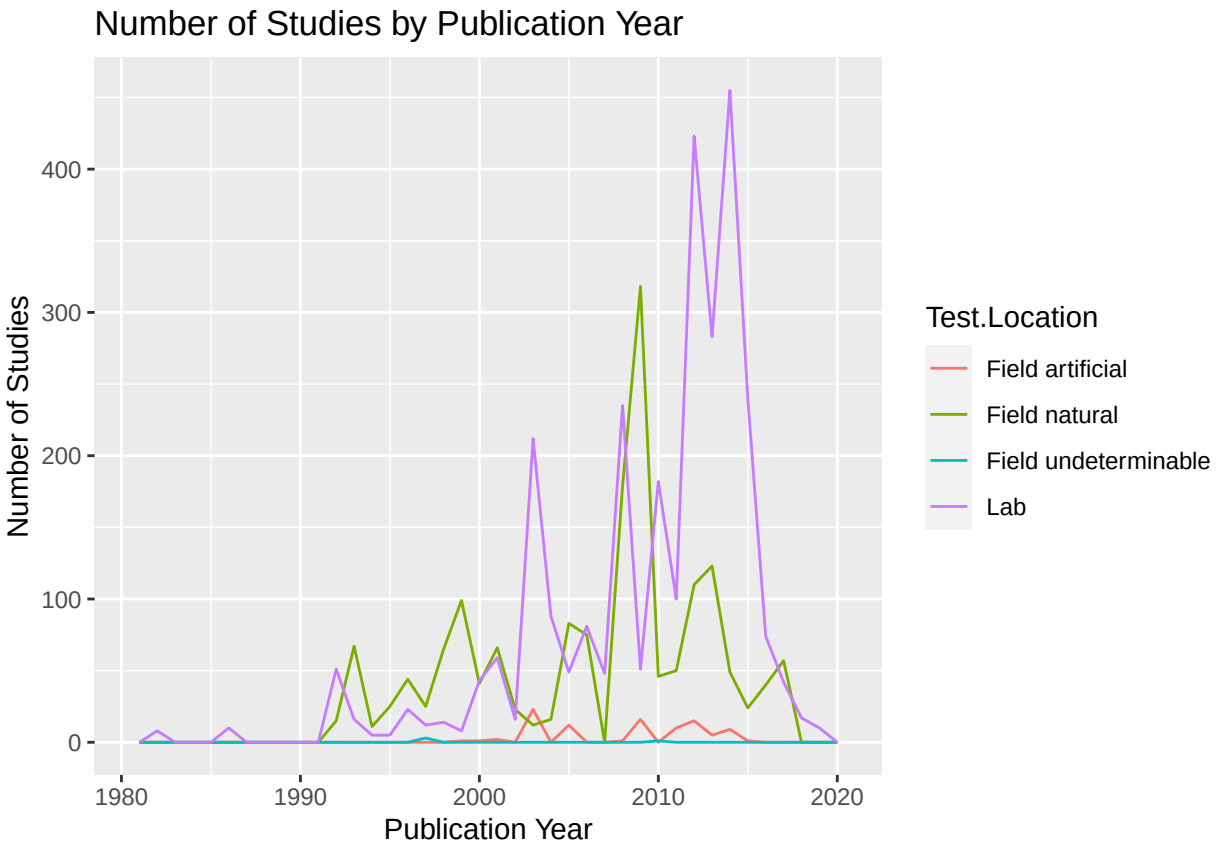
9. Using `geom_freqpoly`, generate a plot of the number of studies conducted by publication year.

```r
#plot of the number of studies conducted by publication year
PubYr <- ggplot(Neonics, aes(x = Publication.Year)) +
  geom_freqpoly(binwidth = 1) +
  labs(title = "Number of Studies by Publication Year",
       x = "Publication Year",
       y = "Number of Studies")
PubYr
```



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

```
#plot of the number of studies conducted by publication year coloring in different test locations
PubYr_Location <- ggplot(Neonics, aes(x = Publication.Year, color = Test.Location)) +
  geom_freqpoly(binwidth = 1) +
  labs(title = "Number of Studies by Publication Year",
       x = "Publication Year",
       y = "Number of Studies")
PubYr_Location
```



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

> Answer: There are four test locations: Field artificial, Field natural, Field undetermindable, and Lab. The most common test location is Lab. Except Field undetermindable, all other three test locations do differ over time.

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.

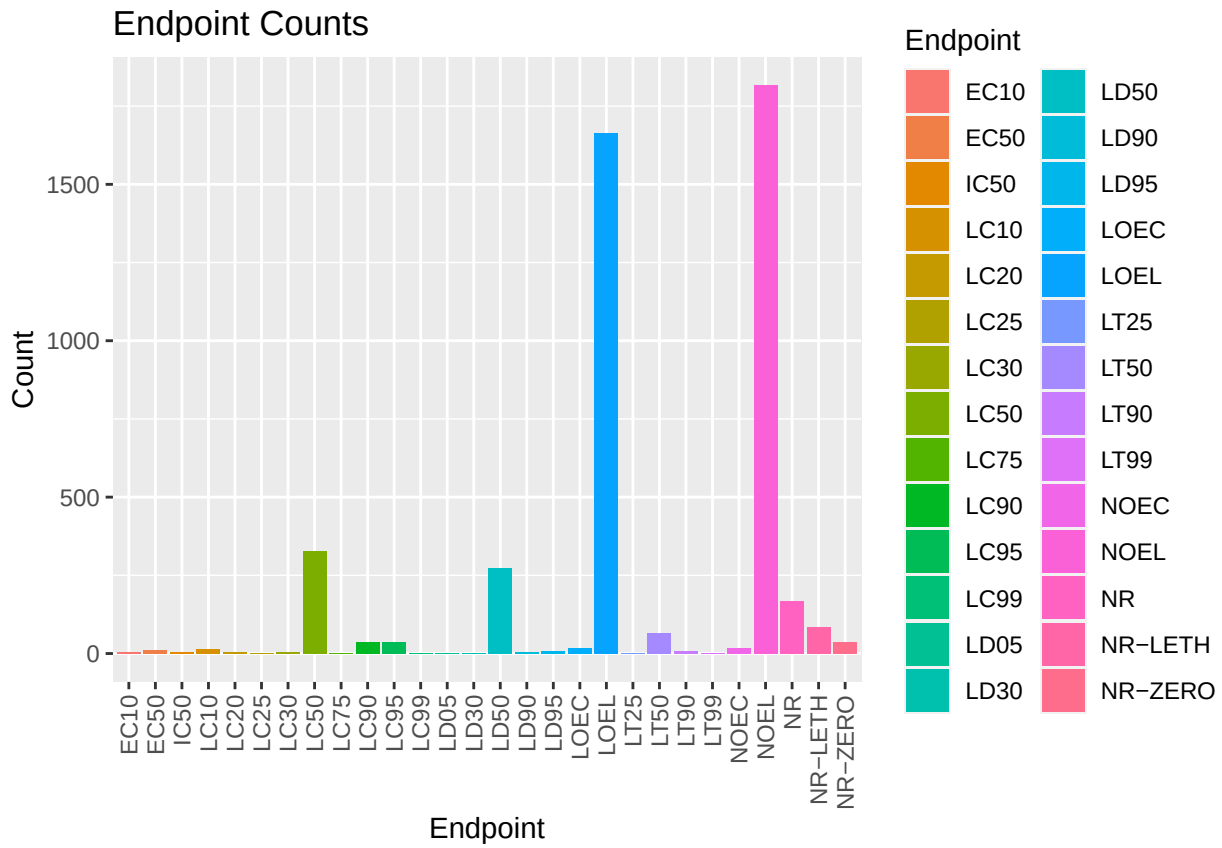
[**TIP**: Add `theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))` to the end of your plot command to rotate and align the X-axis labels...]

```
#dataset for Endpoint with endpoint and count as the only two columns
Endpoint <- Neonics %>% group_by(Endpoint) %>% summarize(count = n())
Endpoint
```

```
## # A tibble: 28 x 2
##    Endpoint count
##    <fct>    <int>
##  1 EC10         6
##  2 EC50        11
##  3 IC50         6
##  4 LC10        15
##  5 LC20         5
##  6 LC25         1
```

```
##  7 LC30          6
##  8 LC50        327
##  9 LC75          1
## 10 LC90         37
## # i 18 more rows
```

```
#bar graph of Endpoint counts.
endpoint_bar <- ggplot(Endpoint, aes(x = Endpoint, y = count, fill = Endpoint)) +
  geom_bar(stat = "identity") +
  labs(title = "Endpoint Counts",
       x = "Endpoint",
       y = "Count") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1))
endpoint_bar
```



Answer: The most two common end points are NOEL(count = 1816) and LOEL(count = 1664). NOEL is No-observable-effect-level: highest dose (concentration) producing effects not significantly different from responses of controls according to author's reported statistical test (NOEAL/NOEC) LOEL is Lowest-observable-effect-level: lowest dose (concentration) producing effects that were significantly different (as reported by authors) from responses of controls (LOEAL/LOEC)

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 of Litter$collectDate before converting to Date
class(Litter$collectDate)
```

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

```
#Convert the class of Litter$collectDate to Date
Litter$collectDate <- as.Date(Litter$collectDate, format = "%Y-%m-%d")

#determine dates when litter was sampled in August 2018
unique(Litter$collectDate)
```

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

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

```
#Use Unique function to get number of plots that were sampled at Niwot Ridge
length(unique(Litter$uid))
```

```
## [1] 188
```

```
#Use summary function to get number of plots that were sampled at Niwot Ridge
summary(Litter)
```

```
##                                     uid                    namedLocation
##  028eea3d-5c20-4afc-bb7e-a05bab305152:  1   NIWO_040.basePlot.ltr:20
##  06789d7b-b742-41d9-8556-79d23c193dc0:  1   NIWO_041.basePlot.ltr:19
##  07780a1e-8af9-4b8a-bb9b-be8add15a1e0:  1   NIWO_046.basePlot.ltr:18
##  0a6cae78-ea42-4e68-98c6-9d929068a38a:  1   NIWO_061.basePlot.ltr:17
##  0ae1c621-387e-42a9-bcf3-7ad1c9b97ab4:  1   NIWO_067.basePlot.ltr:17
##  0b274782-8e52-4f6a-bb17-36daa821f929:  1   NIWO_058.basePlot.ltr:16
##  (Other)                              :182   (Other)              :81
##  domainID   siteID         plotID            trapID        weighDate
##  D13:188   NIWO:188   NIWO_040:20   NIWO_040_205:20   2018-08-06:91
##                       NIWO_041:19   NIWO_041_059:19   2018-09-05:97
##                       NIWO_046:18   NIWO_046_155:18
##                       NIWO_061:17   NIWO_061_169:17
##                       NIWO_067:17   NIWO_067_017:17
##                       NIWO_058:16   NIWO_058_101:16
##                       (Other) :81   (Other)     :81
##        setDate    collectDate                  ovenStartDate
##  2018-07-05:91   Min.   :2018-08-02   2018-08-02T21:00Z:91
##  2018-08-02:97   1st Qu.:2018-08-02   2018-08-30T22:30Z:97
##                  Median :2018-08-30
##                  Mean   :2018-08-16
##                  3rd Qu.:2018-08-30
```

```
##                  Max.   :2018-08-30
##
##             ovenEndDate                       fieldSampleID
##  2018-08-06T18:02Z:91   NEON.LTR.NIWO041059.20180830: 11
##  2018-09-05T19:30Z:97   NEON.LTR.NIWO040205.20180802: 10
##                         NEON.LTR.NIWO040205.20180830: 10
##                         NEON.LTR.NIWO046155.20180802: 10
##                         NEON.LTR.NIWO058101.20180802:  9
##                         NEON.LTR.NIWO061169.20180802:  9
##                         (Other)                     :129
##                            massSampleID      samplingProtocolVersion
##  NEON.LTR.NIWO040205.20180802.MXT:  2   NEON.DOC.001710vE:188
##  NEON.LTR.NIWO040205.20180802.NDL:  2
##  NEON.LTR.NIWO040205.20180830.MXT:  2
##  NEON.LTR.NIWO040205.20180830.NDL:  2
##  NEON.LTR.NIWO041059.20180830.MXT:  2
##  NEON.LTR.NIWO041059.20180830.NDL:  2
##  (Other)                          :176
##        functionalGroup    dryMass       qaDryMass remarks
##  Needles       :30     Min.   :0.0000   N:168     Mode:logical
##  Twigs/branches:28     1st Qu.:0.0000   Y: 20     NA's:188
##  Woody material:26     Median :0.0050
##  Leaves        :24     Mean   :0.6115
##  Other         :24     3rd Qu.:0.3200
##  Flowers       :23     Max.   :8.6300
##  (Other)       :33
##                        measuredBy
##  kstyers@battelleecology.org:91
##  szrillo@battelleecology.org:97
##
##
##
##
##
```

Answer: For unique, Here I used length function on top of the unique function to find out how many plots were sampled at Niwot Ridge. If I simply type unique(Litter$uid), it will only give me every unique experiment. And I got 188 for the number of plots For summary, I used this function on the entire dataset. Although the information of the summary of entire dataset is a lot, I'm still able to find the number of plots sampled at Niwot Ridge under different sections. The number I got here is also 188.

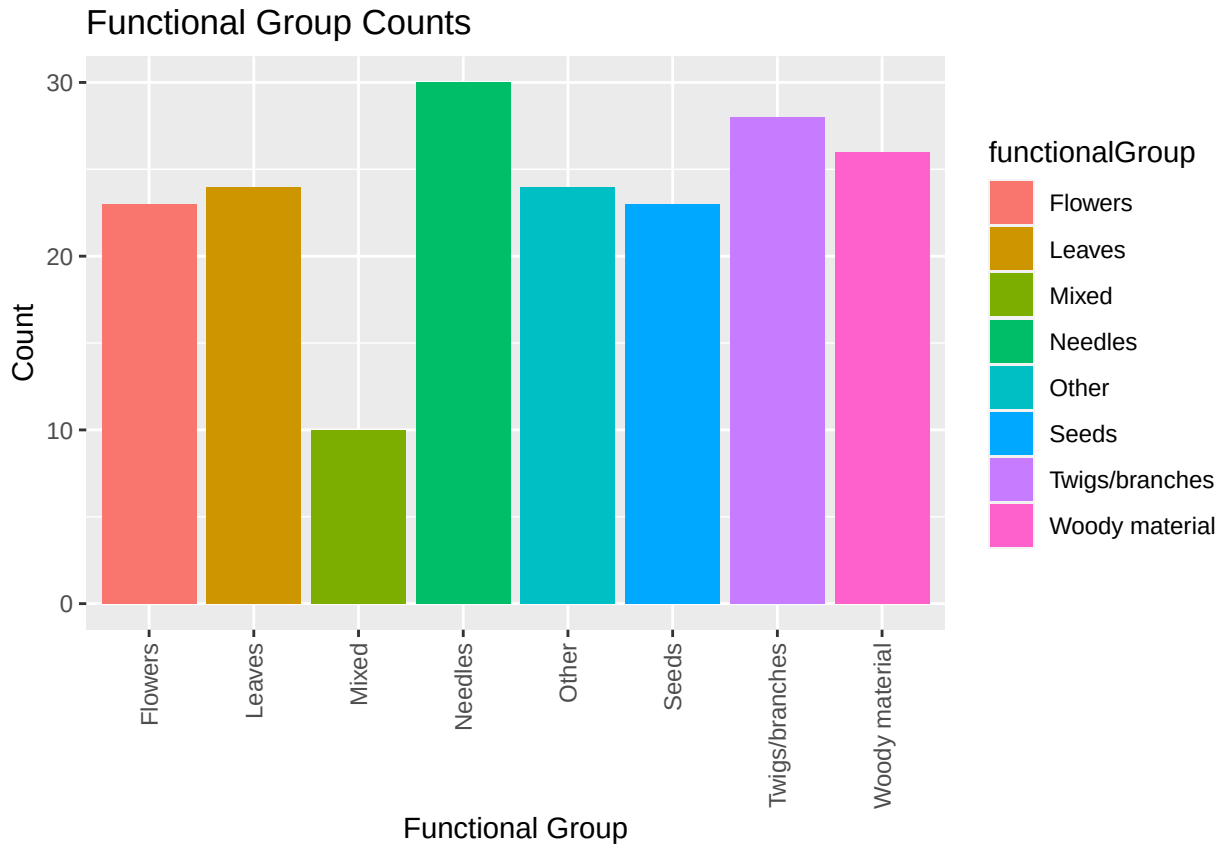
14. Create a bar graph of functionalGroup counts. This shows you what type of litter is collected at the Niwot Ridge sites. Notice that litter types are fairly equally distributed across the Niwot Ridge sites.

```
#build a functionalGroup table
functionalGroup <- Litter %>%
  group_by(functionalGroup) %>%
  summarize(count = n())
functionalGroup
```

```
## # A tibble: 8 x 2
##   functionalGroup count
```

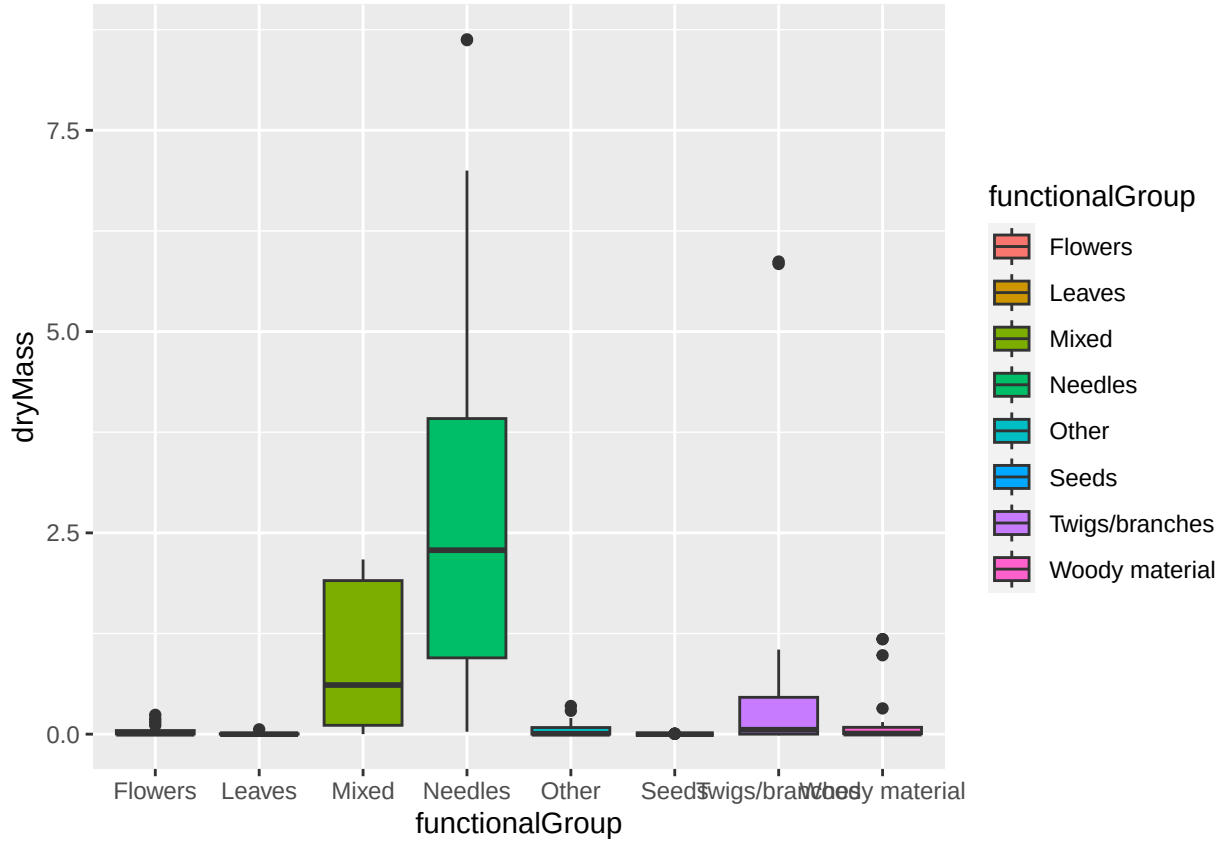
```
##   <fct>           <int>
## 1 Flowers            23
## 2 Leaves             24
## 3 Mixed              10
## 4 Needles            30
## 5 Other              24
## 6 Seeds              23
## 7 Twigs/branches     28
## 8 Woody material     26
```

```r
#creat a bar graph of Endpoint counts.
fg_bar <- ggplot(functionalGroup, aes(x = functionalGroup, y = count, fill = functionalGroup)) +
  geom_bar(stat = "identity") +
  labs(title = "Functional Group Counts",
       x = "Functional Group",
       y = "Count") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1))
fg_bar
```



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

```r
# Create a boxplot
ggplot(Litter, aes(x = functionalGroup, y = dryMass, fill = functionalGroup)) +
  geom_boxplot()
```



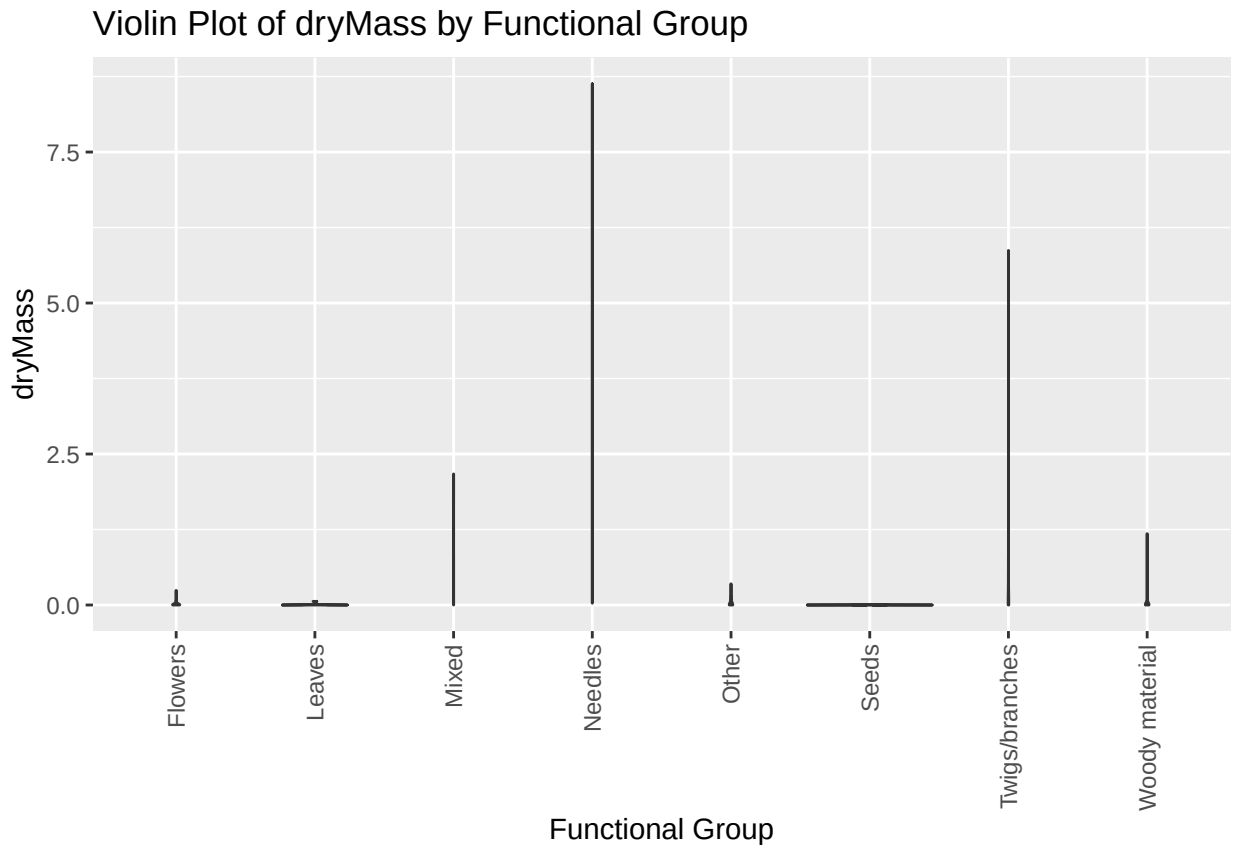

```
  labs(title = "Boxplot Plot of dryMass by Functional Group",
       x = "Functional Group",
       y = "dryMass") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1))
```

```
## NULL
```

```
# Create a violin plot
ggplot(Litter, aes(x = Litter$functionalGroup, y = Litter$dryMass)) +
  geom_violin() +
  labs(title = "Violin Plot of dryMass by Functional Group",
       x = "Functional Group",
       y = "dryMass") +
   theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1))
```

```
## Warning: Use of `Litter$functionalGroup` is discouraged.
## i Use `functionalGroup` instead.
```

```
## Warning: Use of `Litter$dryMass` is discouraged.
## i Use `dryMass` instead.
```



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

> Answer: Here if we look at the violin plot, since the database here may not be large enough, there is no obvious violin shape displayed to let the audience see the distribution of the data. What's more, the audience cannot also see whether there are outliers for each functionalGroup categories. A boxplot in this case can do all of that.

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

> Answer: The Needles category seems to have the highest drymass