Assignment 3: Data Exploration

Sydney Williams

Spring 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 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.

```
# setting up my working directory
getwd()
```

```
## [1] "/home/guest/EDA_Spring2024"
```

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

# importing data sets
Neonics <- read.csv("./Data/Raw/Neonics.csv", stringsAsFactors = TRUE)
Litter <- read.csv("./Data/Raw/Litter.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.

Answer: We might be interested in the ecotoxicology of neonicotinoids on insects to determine and analyze the environmental effects of this insecticide on insects which in turn affect pollination and agriculture.

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: We might be interested in studying litter and woody debris that falls to the ground in forests to analyze the ecological dynamics of the litter and wood debris interacting with the ecosystems on the ground of the forest. This could help solve an environmental issue.

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. For Spatial Sampling design - Sampling occurs in the same locations over the lifetime of the Observatory. However, over time some sampling plots may become impossible to sample, due to disturbance or other local changes. 2. For Temporal Sampling design - Ground traps are sampled once per year. 3. . At sites with deciduous vegetation or limited access during winter months, litter sampling of elevated traps may be discontinued for up to 6 months during the dormant season.

Obtain basic summaries of your data (Neonics)

5. What are the dimensions of the dataset?

```
# using dimension function to find dimensions of dataset
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?

```
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
```

Answer: The most common effects are population and mortality. Population is of specific interest because studying them can help understand how neonicotoids affect the dynamics of specific species and ecosystems, where populations start to decrease. Mortality is important because it provides insights into the immediate lethal impact of neonicotoids.

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...]

```
# Using summary function to determine six most commonly studied species in dataset
species_summary <- summary(as.factor(Neonics$Species.Common.Name))
sorted_species_summary <- sort(species_summary, decreasing = TRUE)

# Extracting the top six species
top_six_species <- head(sorted_species_summary, 6)

# Printing the result
print(top_six_species)
```

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

Answer: The Honey Bee, Bumble Bee, Carniolan Honey Bee, Parasitic Wasp, and Buff Tailed Bumblebee are all important pollinators. They are of interest because they are important for pollination and agriculture and so if they are being killed, there needs to be analysis done on why to promote not using neonicotinoids on farms.

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?

```
# Finding what class Conc.1..Author is
class(Neonics$Conc.1..Author.)
```

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

Answer:The class of Conc.1..Author. column is a factor. It's not a numeric because there are some special characters in the column like '~', '/' and "NR".

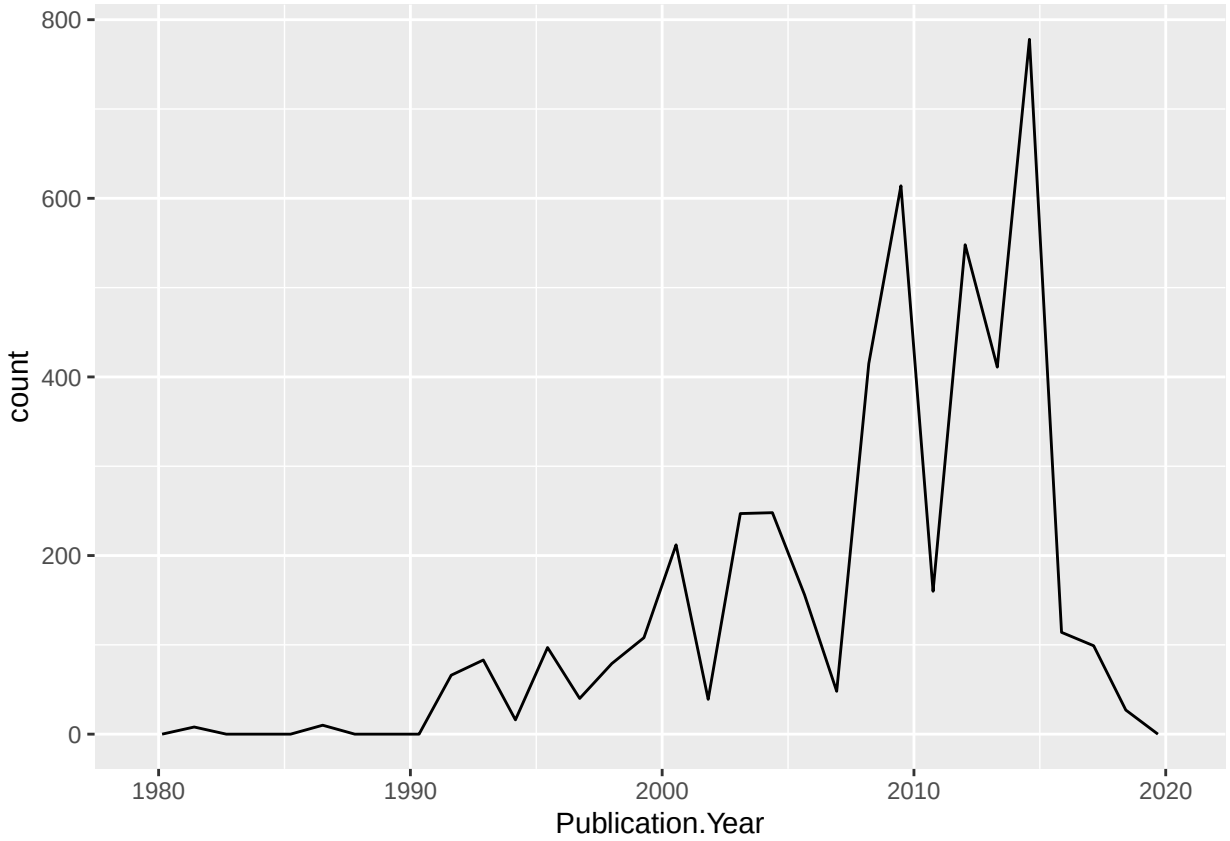
Explore your data graphically (Neonics)

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

```
# load ggplot library
library(ggplot2)

# Using geom_freqpoly to generate a plot of the number of studies conducted by publication year
ggplot(Neonics) +
  geom_freqpoly(aes(x=Publication.Year))
```

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

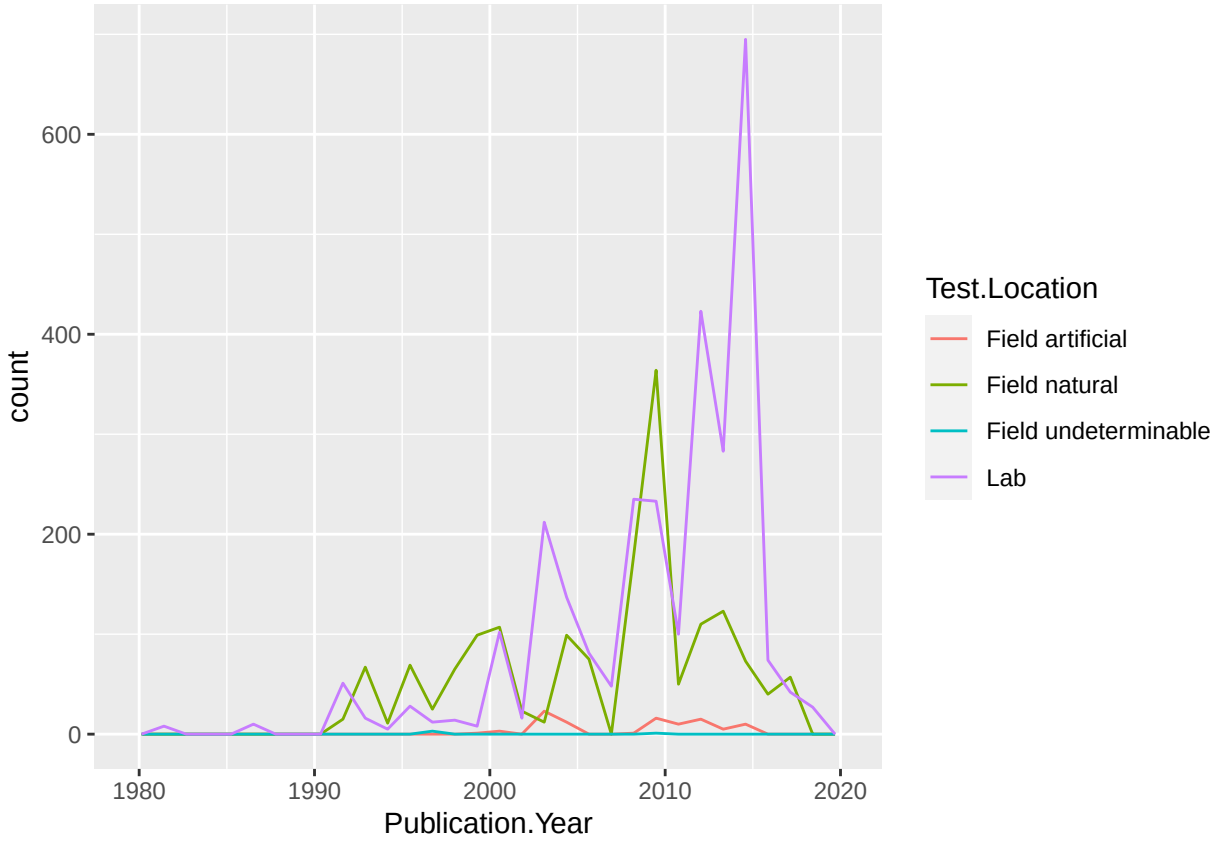


10. Reproduce the same graph but now add a color aesthetic so that different Test.Location are displayed as different colors.

```
# load ggplot2 library
library(ggplot2)

# Using geom_freqpoly to generate a plot of the number of studies conducted by publication year with co
ggplot(Neonics) +
   geom_freqpoly(aes(x = Publication.Year, color = Test.Location))
```

```
## `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?

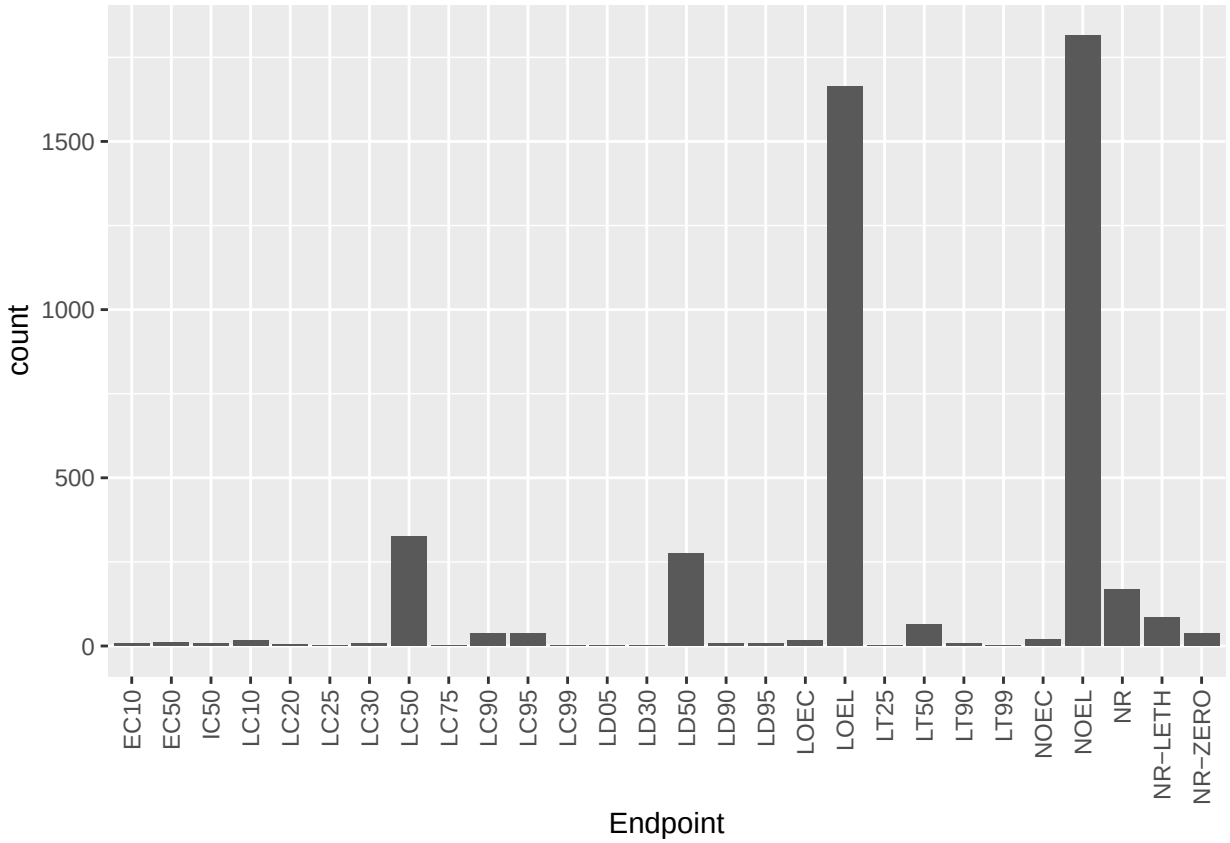
Answer: The lab and the natural field are the most common test locations where between 1990-2000 and 2005-2010 natural field was most common while between 2000-2005 and 2010-2015 was the lab.

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...]

```
# Uploading ggplot2 library
library(ggplot2)

# Create a bar graph of Endpoint counts
ggplot(Neonics) +
  geom_bar(aes(x = Endpoint)) +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1))
```




```
# Finding two most common end points using summary function
endpoint_summary <- summary(Neonics$Endpoint)
top_two_endpoints <- head(sort(endpoint_summary, decreasing = TRUE), 2)

# Printing top two endpoints
print(top_two_endpoints)
```

```
## NOEL LOEL
## 1816 1664
```

Answer: The top two endpoints are defined as NOEL at 1816 and LOEL at 1664.

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.

```
# Checking the class of collectDate
class(Litter$collectDate)
```

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

```
# If it's not a date, convert it to a date
Litter$collectDate <- as.Date(Litter$collectDate, format = "%Y-%m-%d")

# Confirm the new class of the variable
class(Litter$collectDate)
```

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

```r
# Use unique to determine which dates litter was sampled in August 2018
august_2018_dates <- unique(Litter$collectDate[format(Litter$collectDate, "%Y-%m") == "2018-08"])

# Print the unique dates in August 2018
print(august_2018_dates)
```

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

Answer: The dates litter was sampled was 2018-08-02 and 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`?

```r
#Using unique function to create data frame for how many plots were sampled at Niwot Ridge
unique_plots <- unique(Litter$uid)

# Print the unique plots
print(unique_plots)
```

```
##   [1] 7f065fec-bcb2-4af9-b742-8e520fab7f6e 88df210b-1445-4c3f-b19e-5dabd9305c6e
##   [3] 7f3c549c-1dfa-43bf-a485-c7c2bcb31fd6 97806ab5-42d2-49c0-8463-db48cd5eab12
##   [5] 9d7c89f5-85f8-47b6-b415-1ae208580e6f 6ca7a3e8-4d9e-4062-91a0-845f23b5b925
##   [7] a0f02718-2a8e-4f02-beaa-edac27ab1b74 500eb7f8-1881-4a10-bd41-cce84f3b3c47
##   [9] aa0ce5fb-6c8f-42cb-a325-f8c6ab214cff a588a308-b670-4f07-8040-6980d6cfdb72
##  [11] 9df0737f-67f3-4d29-a1ec-8eab4ebc2726 53ec9ef3-bd18-4712-9517-4132649cafe7
##  [13] 57f5c94c-1655-4ea8-a492-64a660c26803 65134dbe-0a9d-446c-a600-4740f396c201
##  [15] be43eacf-16e0-4f2b-b928-2bbf0de2f3c1 836b268d-5e2f-4781-8457-b7b622d13ccd
##  [17] 0fc3a175-47a1-4bd3-9158-96d0ec3815f9 c9bb4c46-d98f-45de-9f17-8a2c608dfe79
##  [19] 4e6bbdd4-3151-4a05-8b77-f5757b11531b ebf1432e-c43e-48c1-ad32-ae4ce423808b
##  [21] 9feeb756-46f9-4bf0-8e94-f2e856728889 edbee742-9d18-4c23-a097-d695a23a4e30
##  [23] 1537c343-14f2-4a75-b91d-c827dd529b55 e101681f-57df-44ca-8d24-b14496813e8c
##  [25] 07780a1e-8af9-4b8a-bb9b-be8add15a1e0 4bca72cc-6f04-480b-95c9-4f55345f32bd
##  [27] b0be64dc-fb65-41e6-b9fa-30201c94606b 6856b517-6d05-403c-893a-3dd8a7b30bff
##  [29] ba9800b5-b01d-4ad3-87fb-1e512c8dc17d f1a1cf1e-1f74-4500-81e3-d179dabed35c
##  [31] acf36093-4706-4dcb-be8c-d4d3a845548f 1475c9b3-a732-4617-bffa-406b072d382e
##  [33] 0f34060c-fc8a-4c8c-bd71-5836e9bbfb05 c1b97ed7-ff4e-4982-9e61-a41d0ab8cbbd
##  [35] f7577092-93be-4a42-9157-f2ee2b12318f 99709f0e-3989-412e-a80d-6987d2ac54e9
##  [37] 4920d35f-624a-45cc-9c75-dac8f9f1d9f8 a1afcbb7-add9-4dd5-8feb-1b0a5e295fed
##  [39] 9cf0463e-c60b-4619-8658-2ed071ae3dcd 73a932ba-e4c5-4ca7-9f19-8d34ef1dea5a
##  [41] a94addfa-17fc-47cb-8d69-4af3903c8bec 51b709df-af0d-441c-8835-b4bf2251ac17
##  [43] cb0eb445-e514-468e-bcad-b6b4ae52ccba f7188915-7307-4a91-b71c-7e3ff38f7d0b
##  [45] c5b62b0f-e753-40e0-8cf3-e78d8a2c6c8a 85a503a8-6817-4513-8a64-d780842d6947
##  [47] 1b049f51-fbda-4b62-83fb-652da4308f5a 3f0a9383-16f4-4197-808c-55ac449b952d
##  [49] 25fff36f-f181-4f62-8529-b419227909d2 ce1f0639-26a8-4a90-9df5-39549bfa412b
##  [51] 028eea3d-5c20-4afc-bb7e-a05bab305152 89f98b92-bbc5-4a43-a852-46db48f6b16f
##  [53] fc47bdf8-99aa-4289-9158-6ebe5b4ccb06 88ae9d88-44fd-4ef3-ba99-bd5c0590b507
##  [55] 7dd99eca-b6ef-42f7-8ce1-672c1d4626a5 0cbcd7ab-3995-49c8-8a36-6361dee82bc6
##  [57] 2a87c5aa-60ab-4ba1-afe5-24e0b52aa7d8 491fba9a-a682-4f7c-ac22-5b01b759f734
##  [59] ba4d7a74-4570-4317-bea1-69a81b8083bf cbf183ba-6177-4afc-88d6-328f37fd57d4
##  [61] 77a0a09f-c819-4e54-b322-0529fa585d02 e5bbc4fc-92d5-4fab-b151-3e9655678e65
##  [63] 0a6cae78-ea42-4e68-98c6-9d929068a38a 80263145-05ab-4b6c-93d3-b058fd56a044
##  [65] fe503f47-15a6-497f-b7dc-b865099d0faa 76676d6a-bdd4-4764-b56f-1e8abd242d62
##  [67] 3eb148f7-219b-43ba-9d39-7c9ea4c6f569 fbc280eb-cd64-41d6-bb82-616d9b11a8a5
##  [69] 63867744-5cd5-4c61-96f1-e6522ea3ef55 ea74be18-c9ce-4708-8ad6-513be0e66a22
##  [71] 3933adbb-6a03-4a7b-b87f-74af1fd92b50 c6a43776-e89f-463b-b27a-fa7b5de8a334
##  [73] b209072a-dc98-480b-b41c-1da05d97a137 9812f8f1-25bf-4b29-8a51-5762e99b7578
```

```
##  [75] d2c18392-2022-4984-86e1-290749d371bc 324775a3-4799-4496-b545-8770724212ed
##  [77] 47c666c6-577b-4de5-90d1-972eb7dd7820 3195d37b-860c-400e-ab26-3cf08f034563
##  [79] aa7ef4c5-da6d-4455-8761-730dd4135191 38e221c3-5011-4d73-aa99-8127154ddd0c
##  [81] f8ef9082-9281-4c65-862c-f2696da58e2a b6582d1e-b9c3-4a0d-bb37-aac749b1642e
##  [83] 3e567fbb-9616-444f-9d13-da894718ecf1 8c02f879-d03e-4903-9ca8-5d4dbcacac57
##  [85] d1dc46e9-052d-4638-bdfd-840a9dc51f44 33aa6853-b3fd-4321-b8f0-9aa144867d6b
##  [87] 78c8dd41-483f-4f1d-9d35-0b775d0901f2 9a6bc315-d122-49a6-9817-8288703b1277
##  [89] 250f0c64-4927-4999-aae6-0d58b1dd7cbf 11d4f1e8-a7d2-4bb6-b25c-ad8296689ba5
##  [91] 1ec020ef-5c48-4b39-b6a7-f94f6a739987 ca636d0d-d049-4e76-be36-4a355a107b6a
##  [93] 84f9566d-5364-4af7-9981-b94a494dc892 89a7b052-f348-4967-b79b-bccfb428d44f
##  [95] 61d57643-995d-482b-ba4a-2fa58d064555 6785dc11-9504-4fd0-9bbe-9bef31f51218
##  [97] 0ae1c621-387e-42a9-bcf3-7ad1c9b97ab4 be20875b-99a3-452e-9102-8a80d59fe527
##  [99] e923388d-bcb1-40ad-b48b-514951f98a94 f67211cc-cfdf-446b-a470-34801aed6539
## [101] 30b7312e-690a-41ed-9aa2-4510769172db 68becebd-7288-4060-86a4-d0d8bfe8967b
## [103] 81c5d213-3ed3-44c6-a250-6365b405aaab a8f15620-bac0-4c39-8d1d-3351d5647165
## [105] 94c59c93-b569-495f-adca-9f711a2a6eb3 94b4a3e9-bb28-48ee-9098-fb99a22f82aa
## [107] b481266c-37f9-462b-b810-51984d506c8d 51a2740a-009c-4262-be1f-b8142eebabfc
## [109] 65bbf249-f8e4-419a-b5fd-0b597900d074 a6f6ad8c-3de1-4723-81f0-0e11b98c5b02
## [111] 06789d7b-b742-41d9-8556-79d23c193dc0 bc63b722-e358-486c-9505-9b0bf85dfef4
## [113] 81014f97-1cda-49f0-adfc-52b93890bba2 14b12019-d75f-47e7-a9b9-933a63701168
## [115] 5fb584f4-e59e-488d-8337-8495e43f3fc0 1868228c-b789-4ed1-a688-d6b19fcdcf31
## [117] 74cde2d3-9540-4012-bc2a-341b5385d59e c13adcbc-da15-4a50-a2ee-fcc81c3722cf
## [119] 1ec4b7ae-7690-48b8-8524-ee1e1ab18992 1ac9e884-e1f8-4138-919b-d295cfa1a215
## [121] 7ccd74d3-fee9-4ff9-8fdb-6aaa11ae857b 36f4f5c4-4a49-43f7-bb4f-4290361e5674
## [123] f52fb766-633a-4141-bd66-fb13dbfdbd0a dd4fb81c-682e-47b5-b698-2186bc1e01be
## [125] e79a0db0-a9da-47bb-9cce-fd50084e1edc 32bd2f37-1274-4c59-95f6-2c7a7c04c814
## [127] 72d1615a-c544-4165-9bdc-dfafa6914a76 aa743782-0a16-4ae8-9891-8c82ee443fc0
## [129] b57cc043-c38c-44ab-9b74-722a5a6bef98 ff27be98-6c8e-440b-8bc6-6b2aae7414d9
## [131] cd691903-631f-40bf-9e89-f895e6e81ca0 480726a3-d83f-4144-a35d-ab986c85512c
## [133] f0e67fb7-03a9-477a-af89-43e1b4f80a8b 1de997a7-2d93-4d99-950d-b374cc71d64d
## [135] 1fb74156-86e2-4b59-a8ab-ff0a1dcd4e45 86301cd8-7886-421c-aaad-56f49a09d9c7
## [137] e61dcc1c-13b6-4b3e-b5b7-ca845eb2a661 d89bb8fd-6cae-4089-9d08-091a608c21a3
## [139] c33d2042-6a5b-4c47-8f7f-e516f1781539 652a84e7-5004-465f-afd5-c42a5690c7c8
## [141] 8b4b0878-e627-44a6-97d7-be404cc3c1f3 86071f09-1d00-44b8-a6d5-506d0fdc0571
## [143] 368a8fb4-4955-4547-833a-3113f8e0a37a 0b274782-8e52-4f6a-bb17-36daa821f929
## [145] 3edbccc1-9e9c-4af9-8ff4-89f05ca76309 abcfac6e-f18e-422b-82e7-26680263d098
## [147] 8301d028-dff9-4927-a898-e305352d4867 4a0a0228-b65f-43fc-893d-8b09408fe851
## [149] 63cb6b0a-d92c-4628-a66a-30fca548598a e11be8c9-5bae-4b59-ae17-73d6361d13c6
## [151] 894b404a-36b3-4ac1-b174-04fca02ae9c8 ca411347-6a76-46a7-a649-1d4c8437ae6e
## [153] 6bafe7ec-7b7a-4fb7-b5e4-8c416631dbf0 8983b717-6a35-4990-98e1-662d19bc50a4
## [155] 647d3e0c-5479-4dc6-80ff-a421e58d4892 58b99e74-2267-4f04-99a5-1d5850502a7b
## [157] 1ace6e31-6078-413a-9ccf-97ab249f2469 f1a11408-0c9d-4071-813e-3f03d71a98d7
## [159] f96bad2d-73b0-4319-82be-d8a180d0ef72 28e788d5-7b1f-4873-b173-79582bdc73b4
## [161] e4a1d2cd-0eb4-4e7c-8dab-925ee15e7c97 c847a531-666b-4271-9675-b3e6a4a9ebb4
## [163] 729390b7-45a6-4b78-a568-ac5b2d01fd6d a424b04b-bdd7-4432-96be-1c4f7618c5a3
## [165] 3803299c-3849-4efe-8b58-1944a97dbbf1 424d28fc-f70d-4e33-b540-89d1dcfe61aa
## [167] a06569ed-afbc-4cb5-9a62-c3d03ed10f0c 50ebc822-1a19-4741-81ca-93ce060c8381
## [169] efeba585-efea-4fda-9b26-5c47c2725f8d 55afd7c2-ebf7-4581-a4c2-76af701a13da
## [171] e5e3eb9e-5813-448d-8b62-160d50634251 62b2bb98-cf97-4444-ba3e-b608c799e378
## [173] 1c833228-0664-4237-abed-ecfbe4fc14f8 dfd5b756-bfed-457f-af58-fbcc88d67690
## [175] adee8e06-a895-4eb8-9dfd-baaf7198efbc b4b0d964-8f8a-499c-b741-bf370d598fcc
## [177] a839c806-7344-4727-b36f-24a109589729 e48b40e8-f16c-4dd6-bea9-7c64efe27202
## [179] f6aaf2c1-9555-41ca-9101-5eaea74d6639 7cda3549-f9e6-4f46-8f5c-f16406a52b50
## [181] 68e8292f-b86a-4efb-88d1-7820c853fe15 89f1d431-0743-4504-9e5e-be3b39c44875
```

```
## [183] ebeec5a0-815d-4f3d-a94f-759cca792b11 d91a07ab-0da7-4182-9e61-a04d01612f83
## [185] cc4285fd-d7cf-40b1-9f67-27aa04b502c3 93f8312d-c181-4613-80af-4d081b29bf0d
## [187] 5b7c6e0e-40c8-4bc6-b509-a760cbe1a5e4 6de90fcf-901c-44c1-88b9-424c92df8c06
## 188 Levels: 028eea3d-5c20-4afc-bb7e-a05bab305152 ...
```

```r
# Determine how many plots were sampled
num_plots <- length(unique_plots)

# Printing number of plots sampled
print(num_plots)
```

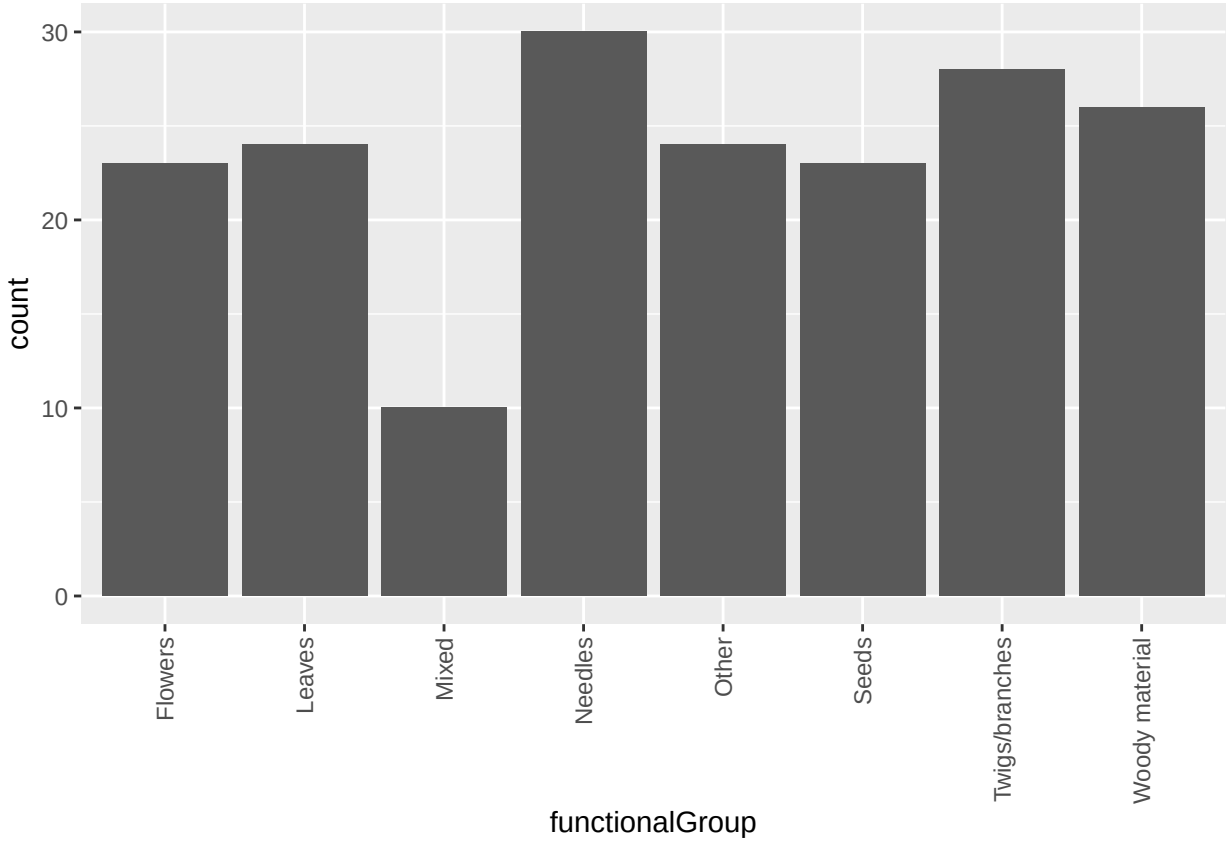
```
## [1] 188
```

Answer: There were 188 plots.This information is different from unique than it is from summary as a unique function tells you the unique values in a specific column or vector while a summary gives you a summary of statistics for a numeric vector.

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.

```r
# Using ggplot2 library
library(ggplot2)

# Create a bar graph of functionalGroup counts
ggplot(Litter) +
  geom_bar(aes(x = functionalGroup)) +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1))
```

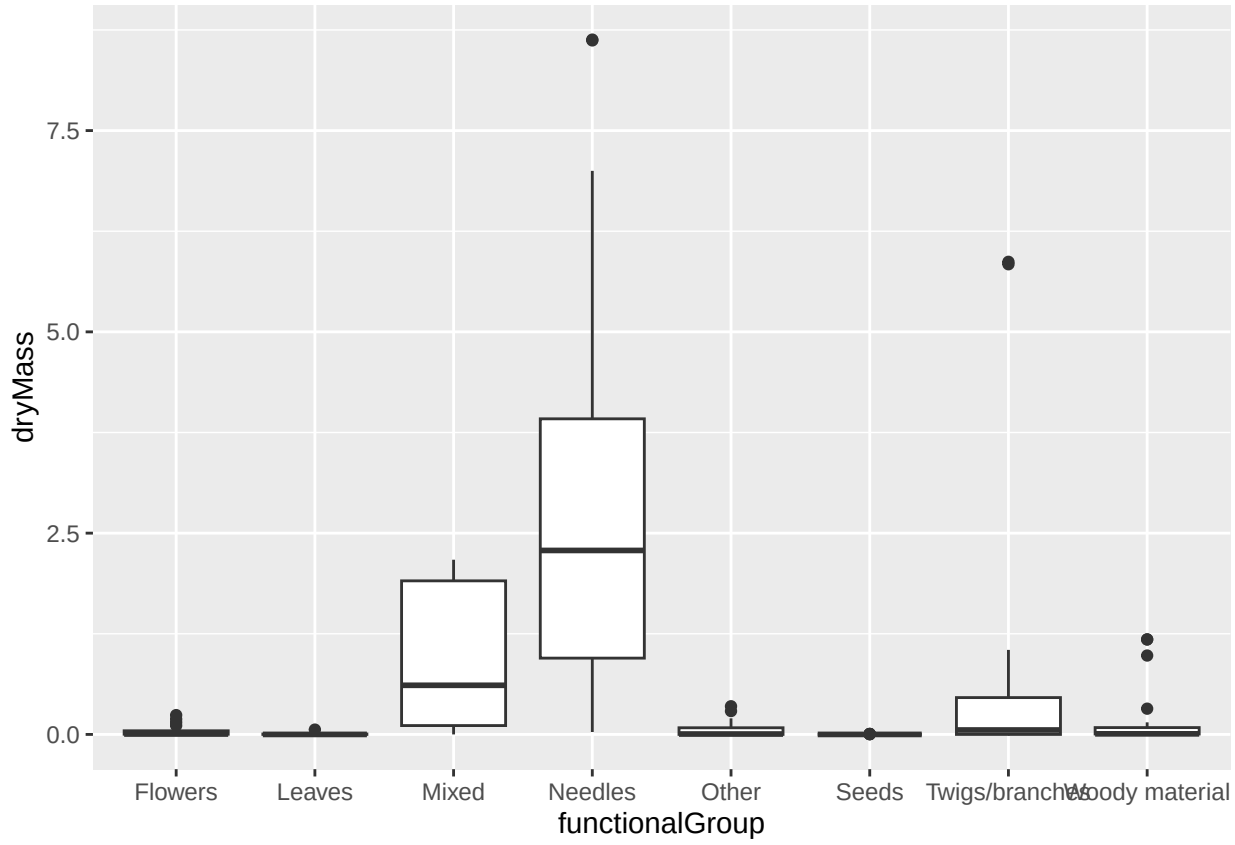


15. Using `geom_boxplot` and `geom_violin`, create a boxplot and a violin plot of dryMass by functional-

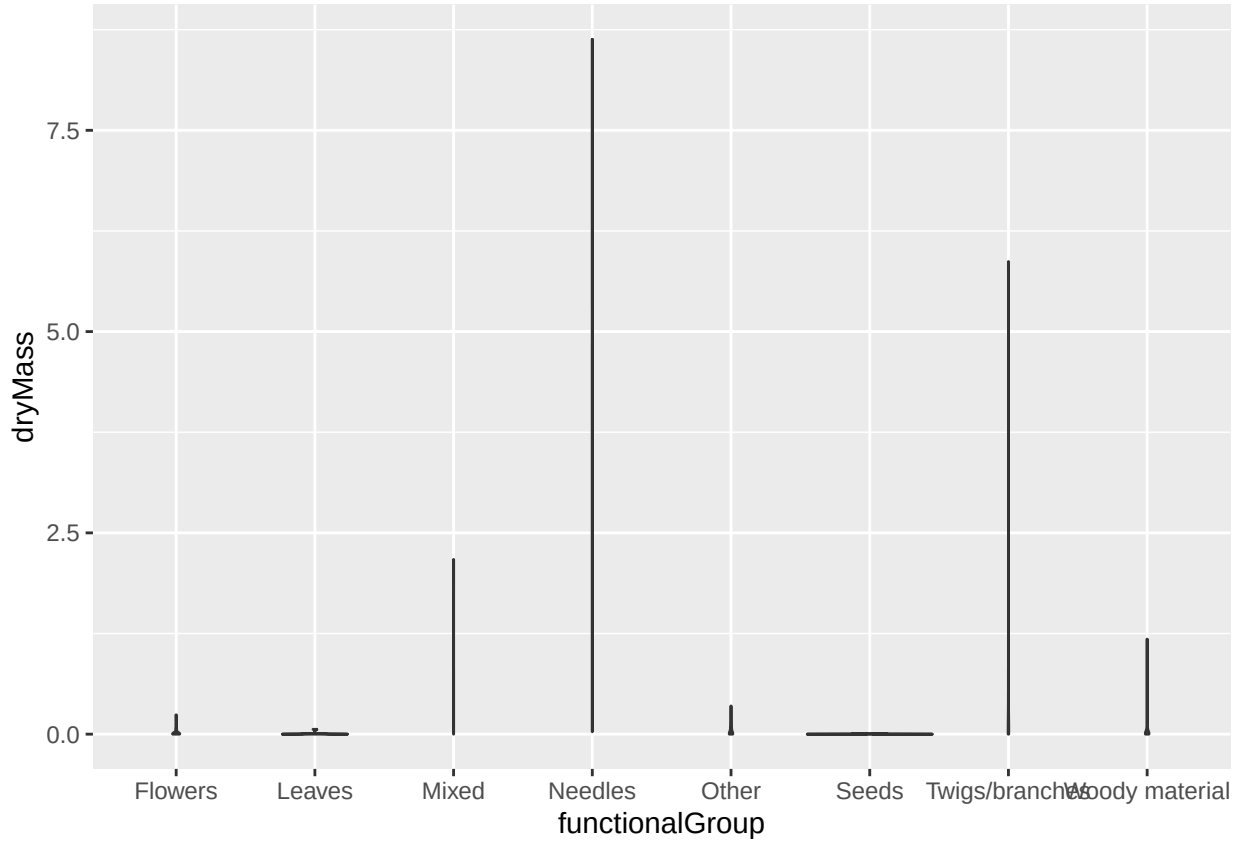
Group.

```
# Using ggplot2 library
library(ggplot2)

# Create a boxplot and a violin plot of dryMass by functionalGroup
ggplot(Litter) +
  geom_boxplot(aes(x = functionalGroup, y = dryMass))
```




```
ggplot(Litter) +
  geom_violin(aes(x=functionalGroup, y=dryMass))
```



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

> Answer: The box plot is a more effective visualization option than the violin plot in this case because it is simplier to identify the mean, mix, max and median of drmymass for each functional group.

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

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