BC_Odes_w_Gibson_data

Nathan G. Earley

2025-12-17

Intro

We're trying to figure out where we should do the Southern Interior of BC's Dragonfly Count and Butterfly Count.

I'm not uploading the data to GitHub because they're fairly big files... let me know if you want them and I'll share.

Data in Collections

As a first step we're thinking of seeing the data available from the digitized Odonata holdings of the BC Entomology Collections from Gibson et al (2024; Can. Entomol. 156(e42): 1–16. doi:10.4039/tce.2024.38).

```
## Read in Gibson data
GibsonDF <- read.csv("data/GibsonData/Gibson2024CanEntSup002.csv")

## Remove rows with NA Lat Long data
GibsonDF <- GibsonDF %>%
  dplyr::filter(!is.na(Longitude), !is.na(Latitude))

## Restrict the datset to just BC records
GibsonBC <- GibsonDF %>%
  dplyr::filter(Prov_State == "British Columbia")
```

Now that we have the dataset, we want to look at which Christmas Bird Count circles in the South Okanagan-Similkameen have data, and how much. To do this we need to first make the circles. We'll do this based off of an approximate centroid of the circles (in a later itteration I'll get these from Birds Canada or Dave Bell). We'll do this for the Apex circle, the Vaseux circle, and the Oliver-Osoyoos circle.

```
## First we'll set the centroid of our count circles (Note: this is approximate)
ApexCentroid <- sf::st_sfc(
  sf::st_point(c(-119.911581, 49.308790)),  # lon, lat
  crs = 4326)
OliverCentroid <- sf::st_sfc(
  sf::st_point(c(-119.521697, 49.085022)),  # lon, lat
  crs = 4326)
VaseuxCentroid <- sf::st_sfc(
  sf::st_point(c(-119.592627, 49.308124)),  # lon, lat
  crs = 4326)

## Now we'll make sure that they're all projected to EPSG:3005 = BC Albers
ApexCentroid_proj <- sf::st_transform(ApexCentroid, 3005)
OliverCentroid_proj <- sf::st_transform(OliverCentroid, 3005)
VaseuxCentroid_proj <- sf::st_transform(VaseuxCentroid, 3005)
```

```
## Now that they're in BC Albers we can make the circle (12km radius)
ApexCBCcircle <- sf::st_buffer(ApexCentroid_proj, dist = 12000)
OliverCBCcircle <- sf::st_buffer(OliverCentroid_proj, dist = 12000)
VaseuxCBCcircle <- sf::st_buffer(VaseuxCentroid_proj, dist = 12000)

## Now lets reproject them to EPSG:4326
ApexCBCcircle_wgs84 <- sf::st_transform(ApexCBCcircle, 4326)
OliverCBCcircle_wgs84 <- sf::st_transform(OliverCBCcircle, 4326)
VaseuxCBCcircle_wgs84 <- sf::st_transform(VaseuxCBCcircle, 4326)

## Finally we need to make sure that the Gibson data are in the right projection
GibsonBC_sf <- GibsonBC %>%
  sf::st_as_sf(coords = c("Longitude", "Latitude"),
               crs = 4326,
               remove = FALSE)
```

Now that we've got that sorted we can subset the data in the Gibson dataset to those records within these circles.

```
## Filter the data to the circles
ApexRecords <- GibsonBC_sf %>%
  dplyr::filter(sf::st_intersects(., ApexCBCcircle_wgs84, sparse = FALSE)[, 1])
OliverRecords <- GibsonBC_sf %>%
  dplyr::filter(sf::st_intersects(., OliverCBCcircle_wgs84, sparse = FALSE)[, 1])
VaseuxRecords <- GibsonBC_sf %>%
  dplyr::filter(sf::st_intersects(., VaseuxCBCcircle_wgs84, sparse = FALSE)[, 1])
```

Now we can do some quick summary stats to see how much data is in each dataset

```
## How many specimens in each circle
nrow(ApexRecords)
```

```
## [1] 18
```

```
nrow(OliverRecords)
```

```
## [1] 990
```

```
nrow(VaseuxRecords)
```

```
## [1] 804
```

```
## How many species (ItemName) in each circle
length(unique(ApexRecords$ItemName))
```

```
## [1] 4
```

```
length(unique(OliverRecords$ItemName))
```

```
## [1] 50
```

```
length(unique(VaseuxRecords$ItemName))
```

```
## [1] 49
```

```
## Now lets see how many of each species
ApexRecSppCounts <- ApexRecords %>%
  dplyr::group_by(ItemName) %>%
  dplyr::summarise(count = n(), .groups = "drop") %>%
  dplyr::arrange(dplyr::desc(count))
```

```
OliverRecSppCounts <- OliverRecords %>%
  dplyr::group_by(ItemName) %>%
  dplyr::summarise(count = n(), .groups = "drop") %>%
  dplyr::arrange(dplyr::desc(count))
VaseuxRecSppCounts <- VaseuxRecords %>%
  dplyr::group_by(ItemName) %>%
  dplyr::summarise(count = n(), .groups = "drop") %>%
  dplyr::arrange(dplyr::desc(count))

## What are the top 5 spp and how many records do they have?
head(sf::st_drop_geometry(ApexRecSppCounts))
```

```
## # A tibble: 4 x 2
##   ItemName                    count
##   <chr>                       <int>
## 1 Somatochlora semicircularis     6
## 2 Aeshna juncea                   5
## 3 Leucorrhinia hudsonica          5
## 4 Somatochlora albicincta         2
```

```
head(sf::st_drop_geometry(OliverRecSppCounts))
```

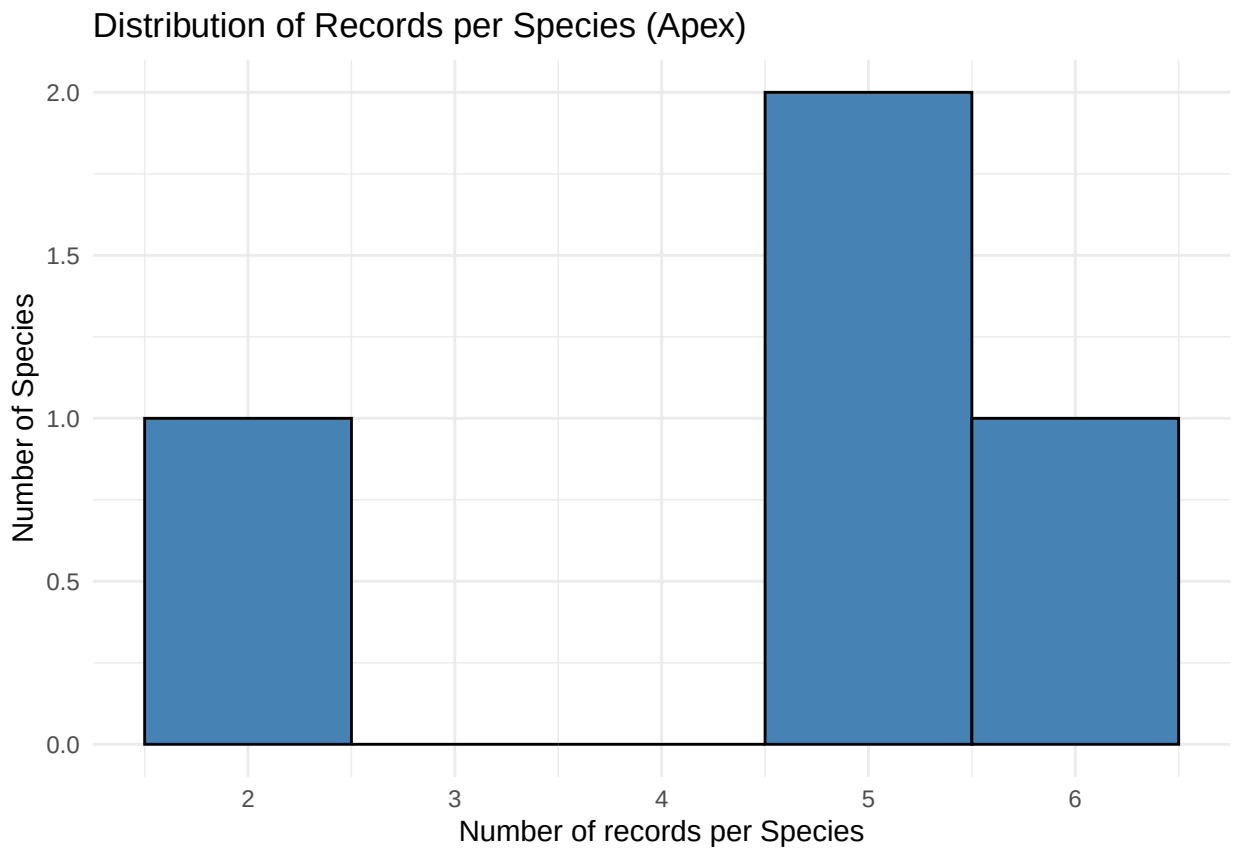
```
## # A tibble: 6 x 2
##   ItemName               count
##   <chr>                  <int>
## 1 Enallagma annexum        102
## 2 Enallagma ebrium          65
## 3 Sympetrum occidentale     63
## 4 Enallagma carunculatum    53
## 5 Ischnura cervula          51
## 6 Sympetrum costiferum      46
```

```
head(sf::st_drop_geometry(VaseuxRecSppCounts))
```

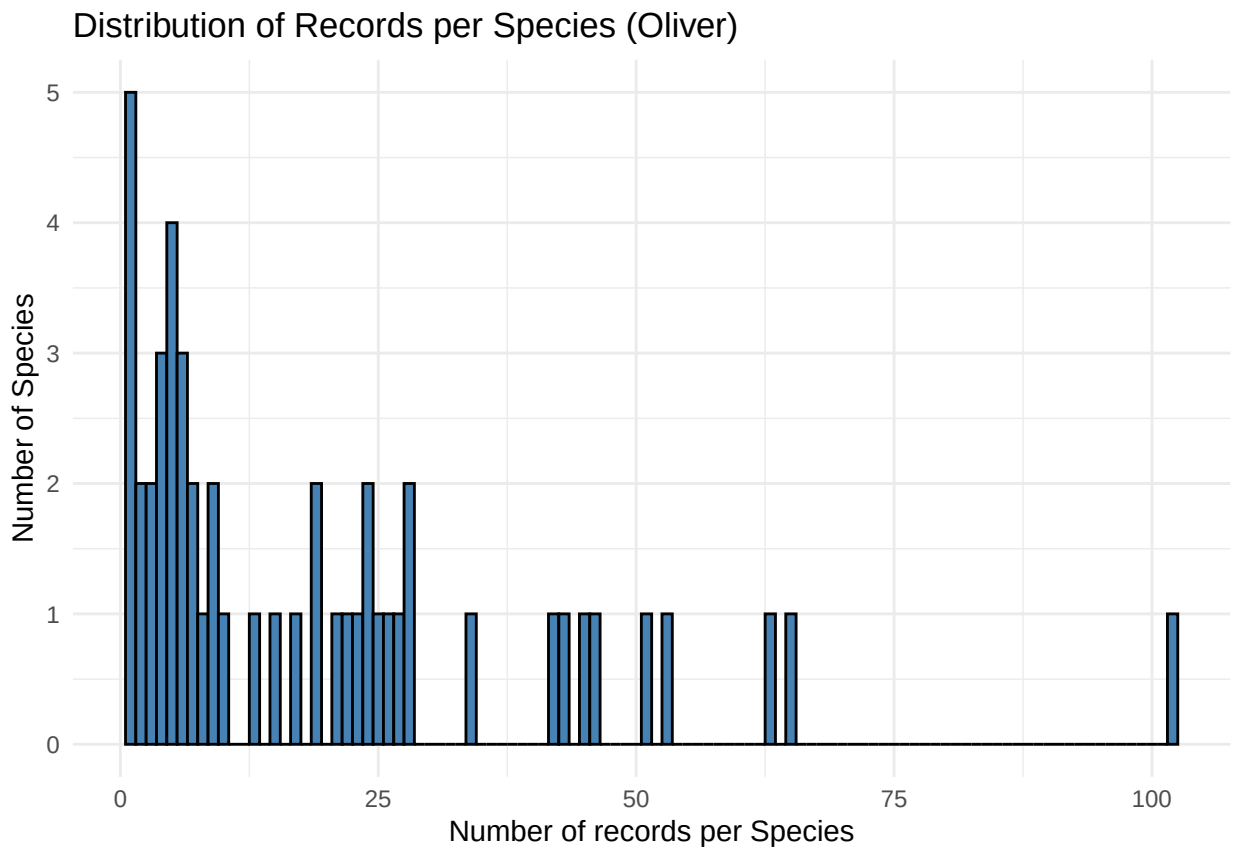
```
## # A tibble: 6 x 2
##   ItemName                count
##   <chr>                   <int>
## 1 Ischnura cervula           73
## 2 Amphiagrion abbreviatum    61
## 3 Enallagma boreale          49
## 4 Enallagma annexum          39
## 5 Lestes disjunctus          38
## 6 Sympetrum obtrusum         37
```

How many species have 1 specimen, 2 specimens, etc?

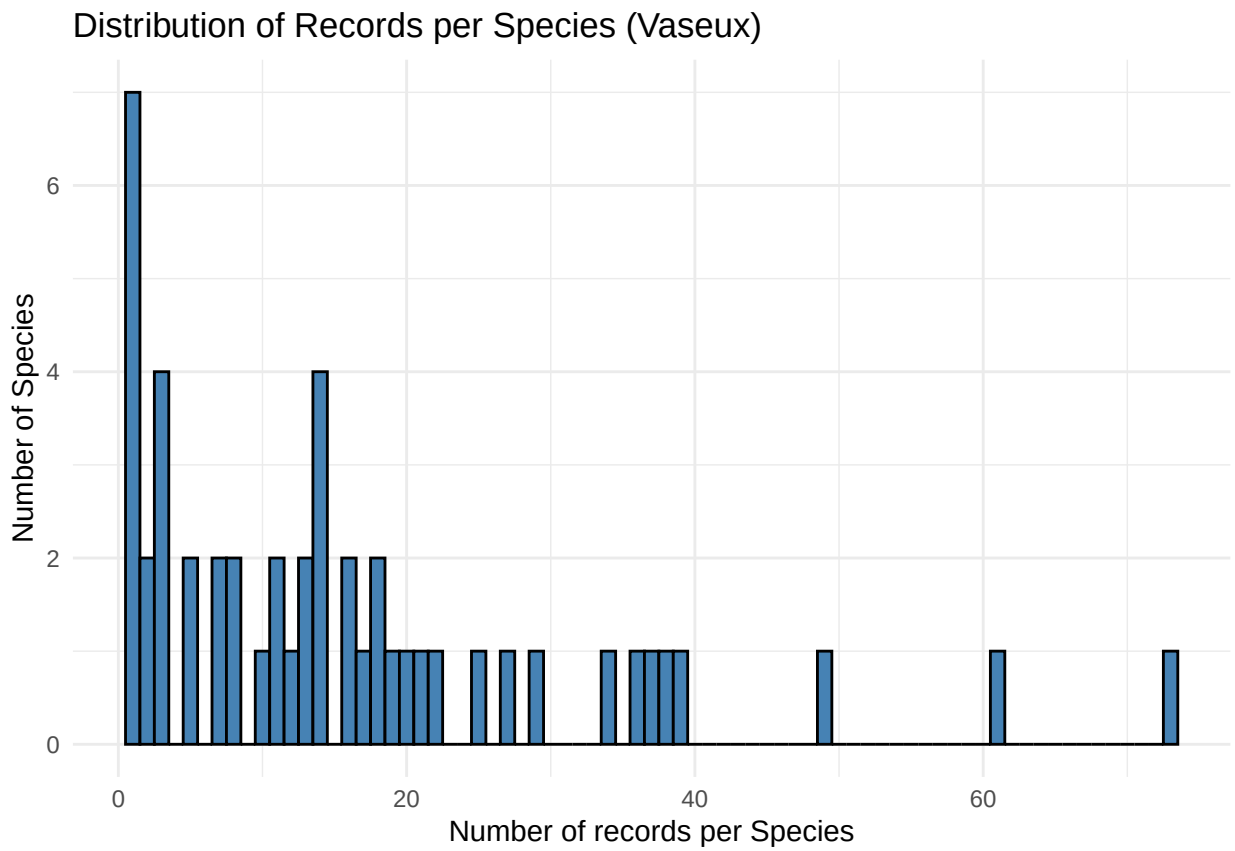
```
ggplot(ApexRecSppCounts, aes(x = count)) +
  geom_histogram(binwidth = 1, fill = "steelblue", color = "black") +
  labs(
    x = "Number of records per Species",
    y = "Number of Species",
    title = "Distribution of Records per Species (Apex)"
  ) +
  theme_minimal()
```




```
ggplot(OliverRecSppCounts, aes(x = count)) +
  geom_histogram(binwidth = 1, fill = "steelblue", color = "black") +
  labs(
    x = "Number of records per Species",
    y = "Number of Species",
    title = "Distribution of Records per Species (Oliver)"
  ) +
  theme_minimal()
```




```
ggplot(VaseuxRecSppCounts, aes(x = count)) +
  geom_histogram(binwidth = 1, fill = "steelblue", color = "black") +
  labs(
    x = "Number of records per Species",
    y = "Number of Species",
    title = "Distribution of Records per Species (Vaseux)"
  ) +
  theme_minimal()
```



Okay so now we know what's going on in the BC collections.

Data on iNaturalist

Lets look at iNat to see how many records there are in the iNat dataset for Odonata. This dataset is much younger but will likely have more recent data. The data we'll use were downloaded from iNaturalist through the Export function on 2025-12-16. Note, some of these records are obscured... we'll start by not removing them but keep that in mind.

Now we'll read in the iNat data.

```
## Read in the data
iNatDF <- read.csv("data/iNatData/iNatOdesBC2025DEC16.csv")

## Keep only reseach grade obs
iNatRG <- iNatDF %>%
  dplyr::filter(quality_grade == "research")

## Keep only CC0, CC-BY, CC-BY-NC records so you don't go to jail for 1B years
iNatRG_CC <- iNatRG %>%
  dplyr::filter(license %in% c("CC0", "CC-BY", "CC-BY-NC"))

## Remove read.csv2()## Remove all data with big (>5km) or NA accuracies
iNatRG_CC_acc <- iNatRG_CC %>%
  dplyr::filter(positional_accuracy < 5000)

## Change the Lat and long columns to Lat and Long
```

```r
iNatRG_CC_acc <- iNatRG_CC_acc %>%
  dplyr::rename("Longitude" = "longitude") %>%
  dplyr::rename("Latitude" = "latitude")

## Finally we need to make sure that the Gibson data are in the right projection
iNat_sf <- iNatRG_CC_acc %>%
  sf::st_as_sf(coords = c("Longitude", "Latitude"),
               crs = 4326,
               remove = FALSE)
```

Now that we have the iNat data read in and cleaned lets filter to theose same circles as before.

```r
## Filter the data to the circles
ApexObs <- iNat_sf %>%
  dplyr::filter(sf::st_intersects(., ApexCBCcircle_wgs84, sparse = FALSE)[, 1])
OliverObs <- iNat_sf %>%
  dplyr::filter(sf::st_intersects(., OliverCBCcircle_wgs84, sparse = FALSE)[, 1])
VaseuxObs <- iNat_sf %>%
  dplyr::filter(sf::st_intersects(., VaseuxCBCcircle_wgs84, sparse = FALSE)[, 1])
```

Now we can do some quick summary stats to see how much data is in each dataset

```r
## How many specimens in each circle
nrow(ApexObs)
```

```
## [1] 2
```

```r
nrow(OliverObs)
```

```
## [1] 385
```

```r
nrow(VaseuxObs)
```

```
## [1] 447
```

```r
## How many species (scientific_name) in each circle
length(unique(ApexObs$scientific_name))
```

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

```r
length(unique(OliverObs$scientific_name))
```

```
## [1] 45
```

```r
length(unique(VaseuxObs$scientific_name))
```

```
## [1] 44
```

```r
## Now lets see how many of each species
ApexObsSppCounts <- ApexObs %>%
  dplyr::group_by(scientific_name) %>%
  dplyr::summarise(count = n(), .groups = "drop") %>%
  dplyr::arrange(dplyr::desc(count))
OliverObsSppCounts <- OliverObs %>%
  dplyr::group_by(scientific_name) %>%
  dplyr::summarise(count = n(), .groups = "drop") %>%
  dplyr::arrange(dplyr::desc(count))
VaseuxObsSppCounts <- VaseuxObs %>%
  dplyr::group_by(scientific_name) %>%
  dplyr::summarise(count = n(), .groups = "drop") %>%
```

```
  dplyr::arrange(dplyr::desc(count))

## What are the top 5 spp and how many records do they have?
head(sf::st_drop_geometry(ApexObsSppCounts))
```

```
## # A tibble: 1 x 2
##   scientific_name    count
##   <chr>              <int>
## 1 Libellula forensis     2
```

```
head(sf::st_drop_geometry(OliverObsSppCounts))
```

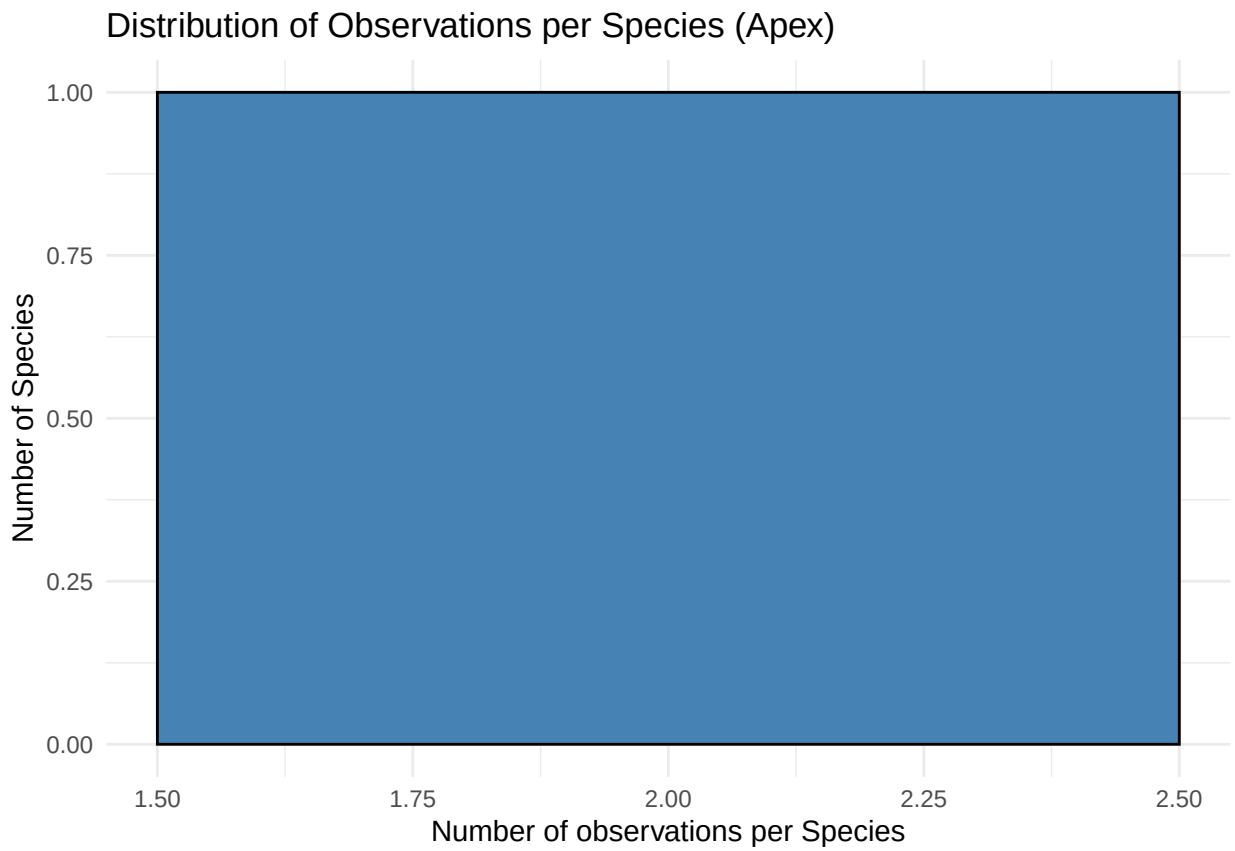
```
## # A tibble: 6 x 2
##   scientific_name     count
##   <chr>               <int>
## 1 Argia emma             56
## 2 Libellula forensis     41
## 3 Libellula pulchella    30
## 4 Sympetrum corruptum    23
## 5 Ischnura cervula       18
## 6 Lestes congener        16
```

```
head(sf::st_drop_geometry(VaseuxObsSppCounts))
```

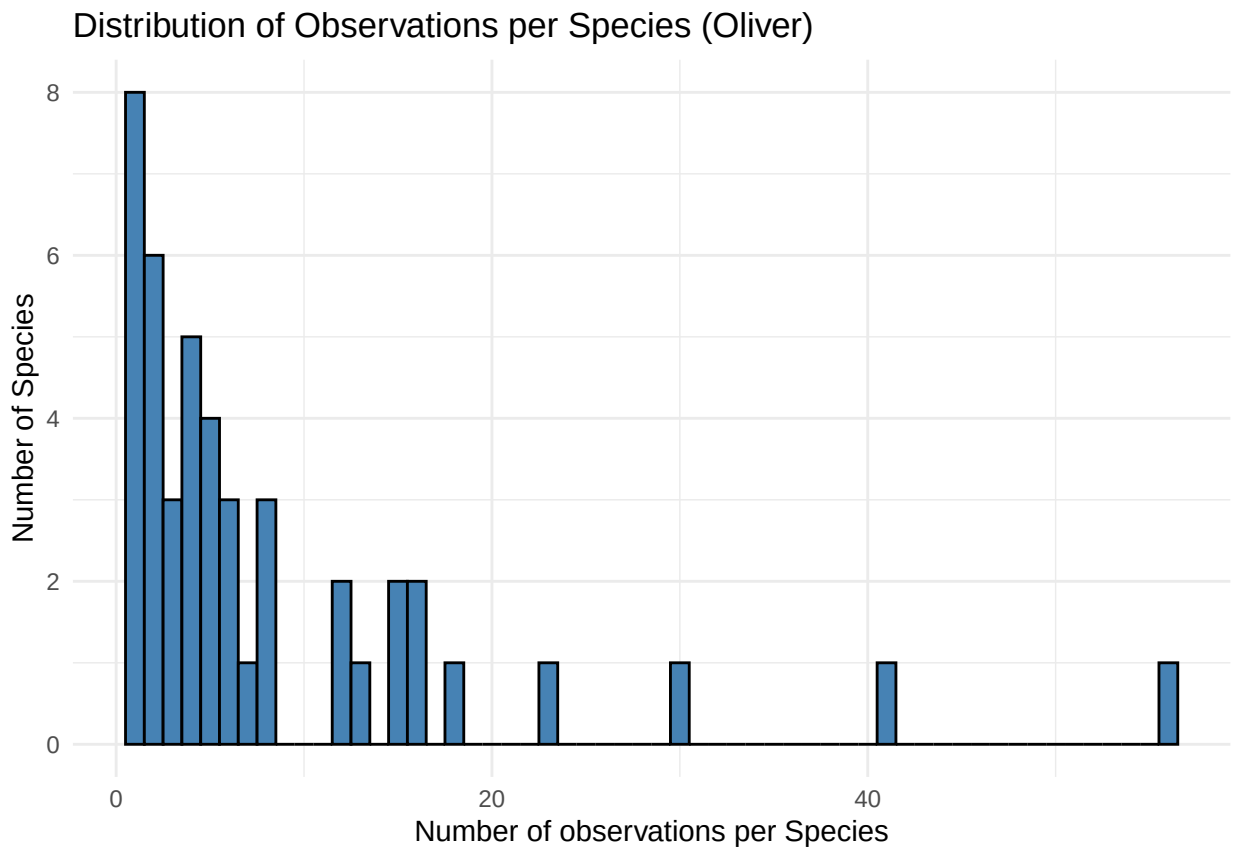
```
## # A tibble: 6 x 2
##   scientific_name          count
##   <chr>                    <int>
## 1 Libellula forensis          67
## 2 Rhionaeschna californica    42
## 3 Libellula quadrimaculata    33
## 4 Ischnura cervula            31
## 5 Aeshna palmata              24
## 6 Argia emma                  23
```

How many species have 1 specimen, 2 specimens, etc?

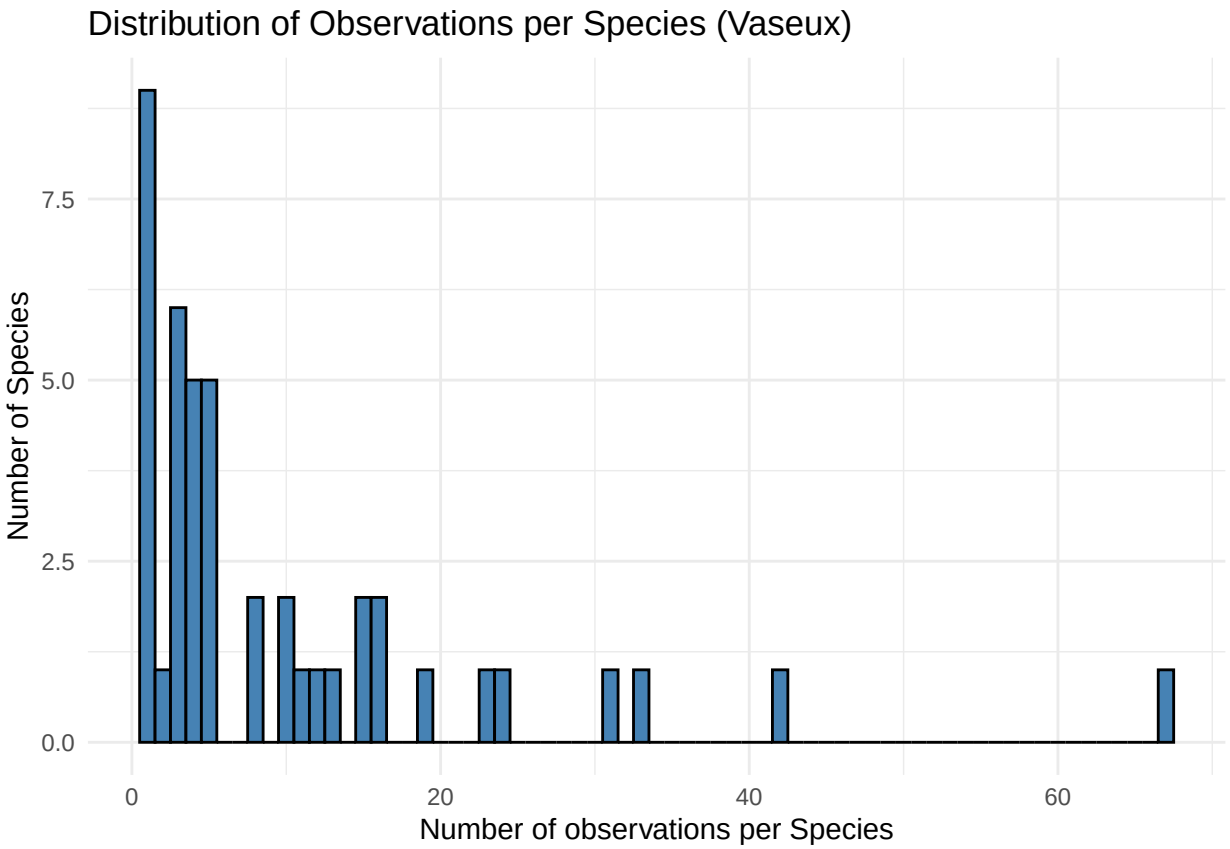
```
ggplot(ApexObsSppCounts, aes(x = count)) +
  geom_histogram(binwidth = 1, fill = "steelblue", color = "black") +
  labs(
    x = "Number of observations per Species",
    y = "Number of Species",
    title = "Distribution of Observations per Species (Apex)"
  ) +
  theme_minimal()
```




```
ggplot(OliverObsSppCounts, aes(x = count)) +
  geom_histogram(binwidth = 1, fill = "steelblue", color = "black") +
  labs(
    x = "Number of observations per Species",
    y = "Number of Species",
    title = "Distribution of Observations per Species (Oliver)"
  ) +
  theme_minimal()
```




```
ggplot(VaseuxObsSppCounts, aes(x = count)) +
  geom_histogram(binwidth = 1, fill = "steelblue", color = "black") +
  labs(
    x = "Number of observations per Species",
    y = "Number of Species",
    title = "Distribution of Observations per Species (Vaseux)"
  ) +
  theme_minimal()
```



Okay, now that we've done that we can see that the two datasets are (at least) superficially similar. Let's do some comparisons.