

Remaining Analysis steps for the final project

Piyush Amitabh

Topics explored in this pdf_document

1. **Sexual Dimorphism and Foraging Behavior:** We will explore differences in foraging behavior between male and female bees and investigate whether they forage on different plant species.
 2. **Identification of Keystone Plant Species:** Using simulation models, we will assess the potential consequences of the extinction of different plant species, identifying the keystone plant species that are most critical for supporting bee populations. This analysis will involve simulating the removal of these plant species from the ecosystem and evaluating the impact on bee species richness, abundance, and diversity.
 3. **Climate and Foraging Patterns:** By integrating climate data, we aim to understand how climatic factors influence bee foraging behavior and whether there are changes in foraging patterns in response to climate variability. This analysis will involve data from sources such as the National Climatic Data Center (NCDC) (<https://www.ncdc.noaa.gov/>) for climate data (Parmesan et al., 2003).
-

Import OBA datasets

```
getwd()

## [1] "/home/piyush/Dropbox/Academics/BI510L Data Sci in R/BI510L Data Sci in R_code/bi510l_project"
setwd("/home/piyush/Dropbox/Academics/BI510L Data Sci in R/BI510L Data Sci in R_code/bi510l_project")
oba_2018 <- read.csv("data/2018_occurrence.csv")
oba_2019 <- read.csv("data/2019_occurrence.csv")
cat("The colnames are identical:", identical(colnames(oba_2018), colnames(oba_2019)))

## The colnames are identical: TRUE

oba_2018_2019 <- rbind(oba_2018, oba_2019)
head(oba_2018_2019, 3)

##      gbifID accessRights
## 1 3033375346          NA
## 2 3033375345          NA
## 3 3033375344          NA
##
## 1 Best, L., C. Feuerborn, J. Holt, S. Kincaid, C.J. Marshall, A. Melathopoulos and S.V.J.Robinson. 20
## 2 Best, L., C. Feuerborn, J. Holt, S. Kincaid, C.J. Marshall, A. Melathopoulos and S.V.J.Robinson. 20
## 3 Best, L., C. Feuerborn, J. Holt, S. Kincaid, C.J. Marshall, A. Melathopoulos and S.V.J.Robinson. 20
##   language   license modified publisher references      rightsHolder
## 1      en CC_BY_4_0    <NA>      NA      NA Oregon State University
## 2      en CC_BY_4_0    <NA>      NA      NA Oregon State University
## 3      en CC_BY_4_0    <NA>      NA      NA Oregon State University
```

```

##      type
## 1 Dataset
## 2 Dataset
## 3 Dataset
##
##                                     institutionID
## 1 https://www.gbif.org/grscicoll/institution/f23b6ebc-db50-482b-a923-f26b5cd8d2d1
## 2 https://www.gbif.org/grscicoll/institution/f23b6ebc-db50-482b-a923-f26b5cd8d2d1
## 3 https://www.gbif.org/grscicoll/institution/f23b6ebc-db50-482b-a923-f26b5cd8d2d1
##
##                                     collectionID
## 1 https://www.gbif.org/grscicoll/institution/f23b6ebc-db50-482b-a923-f26b5cd8d2d1
## 2 https://www.gbif.org/grscicoll/institution/f23b6ebc-db50-482b-a923-f26b5cd8d2d1
## 3 https://www.gbif.org/grscicoll/institution/f23b6ebc-db50-482b-a923-f26b5cd8d2d1
##
##                                     datasetID institutionCode
## 1 DOI: http://dx.doi.org/10.5399/osu/cat_osac.5.1.4647      OSAC
## 2 DOI: http://dx.doi.org/10.5399/osu/cat_osac.5.1.4647      OSAC
## 3 DOI: http://dx.doi.org/10.5399/osu/cat_osac.5.1.4647      OSAC
##  collectionCode  datasetName ownerInstitutionCode      basisOfRecord
## 1              OBA-OSAC-2018              OSAC PRESERVED_SPECIMEN
## 2              OBA-OSAC-2018              OSAC PRESERVED_SPECIMEN
## 3              OBA-OSAC-2018              OSAC PRESERVED_SPECIMEN
##  informationWithheld dataGeneralizations dynamicProperties
## 1              <NA>              <NA>              NA
## 2              <NA>              <NA>              NA
## 3              <NA>              <NA>              NA
##
##                                     occurrenceID catalogNumber
## 1      http://osac.oregonstate.edu/OBS/OBA_Martin_Stein:18.019.001
## 2 http://osac.oregonstate.edu/OBS/OBA_Robert_Pederson:18.168.010
## 3      http://osac.oregonstate.edu/OBS/OBA_Carol_Horning:18.046.001
##  recordNumber      recordedBy recordedByID individualCount organismQuantity
## 1              NA      Martin Stein              NA              NA              NA
## 2              NA Robert Pederson              NA              NA              NA
## 3              NA      Carol Horning              NA              NA              NA
##  organismQuantityType      sex lifeStage reproductiveCondition behavior
## 1              NA FEMALE              NA              NA              NA
## 2              NA  MALE              NA              NA              NA
## 3              NA FEMALE              NA              NA              NA
##  establishmentMeans degreeOfEstablishment pathway
## 1              NA              NA      NA
## 2              NA              NA      NA
## 3              NA              NA      NA
##  georeferenceVerificationStatus occurrenceStatus preparations disposition
## 1              NA              PRESENT      NA not retained
## 2              NA              PRESENT      NA not retained
## 3              NA              PRESENT      NA not retained
##  associatedOccurrences associatedReferences associatedSequences
## 1              NA              NA              NA
## 2              NA              NA              NA
## 3              NA              NA              NA
##
##                                     associatedTaxa      otherCatalogNumbers
## 1 "foraging on" : "Vaccinium ovatum"      Martin_Stein:18.019.001
## 2                                     Robert_Pederson:18.168.010
## 3                                     Carol_Horning:18.046.001
##  occurrenceRemarks organismID organismName organismScope associatedOrganisms
## 1              NA              NA              NA              NA              NA

```

## 2	NA	NA	NA	NA	NA	NA
## 3	NA	NA	NA	NA	NA	NA
##	previousIdentifications	organismRemarks	materialSampleID	eventID		
## 1	NA	NA	NA	NA		
## 2	NA	NA	NA	NA		
## 3	NA	NA	NA	NA		
##	parentEventID	fieldNumber	eventDate	eventTime		
## 1	NA	Martin_Stein:18.019.001	2018-05-21T00:00:00	NA		
## 2	NA	Robert_Pederson:18.168.010	2018-08-19T00:00:00	NA		
## 3	NA	Carol_Horning:18.046.001	2018-07-15T00:00:00	NA		
##	startDayOfYear	endDayOfYear	year	month	day	verbatimEventDate
## 1	NA	NA	2018	5	21	5/21/18
## 2	NA	NA	2018	8	19	8/19/18
## 3	NA	NA	2018	7	15	7/15/18
##	samplingProtocol	sampleSizeValue	sampleSizeUnit	samplingEffort	fieldNotes	
## 1	aerial net	NA	NA	NA		
## 2	aerial net	NA	NA	NA		
## 3	aerial net	NA	NA	NA		
##	eventRemarks	locationID	higherGeographyID	higherGeography	continent	
## 1	NA	NA	NA	NA	NORTH_AMERICA	
## 2	NA	NA	NA	NA	NORTH_AMERICA	
## 3	NA	NA	NA	NA	NORTH_AMERICA	
##	waterBody	islandGroup	island	countryCode	stateProvince	county municipality
## 1	NA	NA	NA	US	Oregon	Lane
## 2	NA	NA	NA	US	Oregon	Deschutes
## 3	NA	NA	NA	US	Oregon	Wheeler
##		locality	verbatimLocality	verbatimElevation		
## 1		Yachats, 3 miles SE of	NA	NA		
## 2		Bend, Tumalo, Pinehurst Road.	NA	NA		
## 3		Mitchell	NA	NA		
##	verticalDatum	verbatimDepth	minimumDistanceAboveSurfaceInMeters			
## 1	NA	NA	NA			
## 2	NA	NA	NA			
## 3	NA	NA	NA			
##	maximumDistanceAboveSurfaceInMeters	locationAccordingTo	locationRemarks			
## 1	NA	NA	NA			
## 2	NA	NA	NA			
## 3	NA	NA	NA			
##	decimalLatitude	decimalLongitude	coordinateUncertaintyInMeters			
## 1	44.27412	-124.0569	NA			
## 2	44.14590	-121.3766	NA			
## 3	44.86700	-120.3970	NA			
##	coordinatePrecision	pointRadiusSpatialFit	verbatimCoordinateSystem			
## 1	NA	NA	NA			
## 2	NA	NA	NA			
## 3	NA	NA	NA			
##	verbatimSRS	footprintWKT	footprintSRS	footprintSpatialFit	georeferencedBy	
## 1	NA	NA	NA	NA	NA	
## 2	NA	NA	NA	NA	NA	
## 3	NA	NA	NA	NA	NA	
##	georeferencedDate	georeferenceProtocol	georeferenceSources			
## 1	NA	NA	NA			
## 2	NA	NA	NA			
## 3	NA	NA	NA			

##	georeferenceRemarks	geologicalContextID	earliestEonOrLowestEonothem
## 1	NA	NA	NA
## 2	NA	NA	NA
## 3	NA	NA	NA
##	latestEonOrHighestEonothem	earliestEraOrLowestErathem	
## 1	NA	NA	
## 2	NA	NA	
## 3	NA	NA	
##	latestEraOrHighestErathem	earliestPeriodOrLowestSystem	
## 1	NA	NA	
## 2	NA	NA	
## 3	NA	NA	
##	latestPeriodOrHighestSystem	earliestEpochOrLowestSeries	
## 1	NA	NA	
## 2	NA	NA	
## 3	NA	NA	
##	latestEpochOrHighestSeries	earliestAgeOrLowestStage	latestAgeOrHighestStage
## 1	NA	NA	NA
## 2	NA	NA	NA
## 3	NA	NA	NA
##	lowestBiostratigraphicZone	highestBiostratigraphicZone	
## 1	NA	NA	
## 2	NA	NA	
## 3	NA	NA	
##	lithostratigraphicTerms	group	formation member bed identificationID
## 1	NA	NA	NA NA NA NA
## 2	NA	NA	NA NA NA NA
## 3	NA	NA	NA NA NA NA
##	verbatimIdentification	identificationQualifier	typeStatus identifiedBy
## 1	NA	NA	NA Lincoln R. Best
## 2	NA	NA	NA Lincoln R. Best
## 3	NA	NA	NA Lincoln R. Best
##	identifiedByID	dateIdentified	identificationReferences
## 1	NA	2019-01-01T00:00:00	NA
## 2	NA	2019-01-01T00:00:00	NA
## 3	NA	2019-01-01T00:00:00	NA
##	identificationVerificationStatus	identificationRemarks	taxonID
## 1	NA		NA
## 2	NA		NA
## 3	NA		NA
##	scientificNameID	acceptedNameUsageID	parentNameUsageID originalNameUsageID
## 1	NA	NA	NA NA
## 2	NA	NA	NA NA
## 3	NA	NA	NA NA
##	nameAccordingToID	namePublishedInID	taxonConceptID
## 1	NA	NA	NA
## 2	NA	NA	NA
## 3	NA	NA	NA
##		scientificName	acceptedNameUsage parentNameUsage
## 1		Lasioglossum Curtis, 1833	NA NA
## 2		Megachile perihirta Cockerell, 1898	NA NA
## 3		Megachile perihirta Cockerell, 1898	NA NA
##	originalNameUsage	nameAccordingTo	namePublishedIn namePublishedInYear
## 1	NA	NA	NA NA

## 2	NA	NA	NA	NA	NA		
## 3	NA	NA	NA	NA	NA		
##	higherClassification	kingdom	phylum	class	order	family	
## 1	NA	Animalia	Arthropoda	Insecta	Hymenoptera	Halictidae	
## 2	NA	Animalia	Arthropoda	Insecta	Hymenoptera	Megachilidae	
## 3	NA	Animalia	Arthropoda	Insecta	Hymenoptera	Megachilidae	
##	subfamily	genus	genericName	subgenus	infragenericEpithet		
## 1	NA	Lasioglossum	Lasioglossum	NA	NA		
## 2	NA	Megachile	Megachile	NA	NA		
## 3	NA	Megachile	Megachile	NA	NA		
##	specificEpithet	infraspecificEpithet	cultivarEpithet	taxonRank			
## 1			NA	NA	GENUS		
## 2	perihirta		NA	NA	SPECIES		
## 3	perihirta		NA	NA	SPECIES		
##	verbatimTaxonRank	vernacularName	nomenclaturalCode	taxonomicStatus			
## 1	NA	NA	NA	ACCEPTED			
## 2	NA	NA	NA	ACCEPTED			
## 3	NA	NA	NA	ACCEPTED			
##	nomenclaturalStatus	taxonRemarks		datasetKey			
## 1	NA	NA	b2974853-6c41-4c63-a11b-7989e58a3ad4				
## 2	NA	NA	b2974853-6c41-4c63-a11b-7989e58a3ad4				
## 3	NA	NA	b2974853-6c41-4c63-a11b-7989e58a3ad4				
##	publishingCountry		lastInterpreted	elevation	elevationAccuracy	depth	
## 1	US	2023-09-07T10:01:31.129Z		NA	NA	NA	
## 2	US	2023-09-07T10:01:31.129Z		NA	NA	NA	
## 3	US	2023-09-07T10:01:31.129Z		NA	NA	NA	
##	depthAccuracy	distanceFromCentroidInMeters					
## 1	NA	NA					
## 2	NA	NA					
## 3	NA	NA					
##						issue	
## 1	GEODETTIC_DATUM_ASSUMED_WGS84;CONTINENT_DERIVED_FROM_COORDINATES;COLLECTION_MATCH_NONE						
## 2	GEODETTIC_DATUM_ASSUMED_WGS84;CONTINENT_DERIVED_FROM_COORDINATES;COLLECTION_MATCH_NONE						
## 3	GEODETTIC_DATUM_ASSUMED_WGS84;CONTINENT_DERIVED_FROM_COORDINATES;COLLECTION_MATCH_NONE						
##	mediaType	hasCoordinate	hasGeospatialIssues	taxonKey	acceptedTaxonKey		
## 1	NA	TRUE	FALSE	1353501	1353501		
## 2	NA	TRUE	FALSE	1335630	1335630		
## 3	NA	TRUE	FALSE	1335630	1335630		
##	kingdomKey	phylumKey	classKey	orderKey	familyKey	genusKey	subgenusKey
## 1	1	54	216	1457	7908	1353501	NA
## 2	1	54	216	1457	7911	1335011	NA
## 3	1	54	216	1457	7911	1335011	NA
##	speciesKey	species		acceptedScientificName			
## 1	NA			Lasioglossum Curtis, 1833			
## 2	1335630	Megachile perihirta	Megachile perihirta	Cockerell, 1898			
## 3	1335630	Megachile perihirta	Megachile perihirta	Cockerell, 1898			
##		verbatimScientificName	typifiedName	protocol			
## 1		Lasioglossum	NA	EML			
## 2	Megachile perihirta	Cockerell, 1898	NA	EML			
## 3	Megachile perihirta	Cockerell, 1898	NA	EML			
##		lastParsed		lastCrawled	repatriated		
## 1	2023-09-07T10:01:31.129Z	2023-01-05T07:44:36.605Z		FALSE			
## 2	2023-09-07T10:01:31.129Z	2023-01-05T07:44:36.605Z		FALSE			
## 3	2023-09-07T10:01:31.129Z	2023-01-05T07:44:36.605Z		FALSE			

```
##   relativeOrganismQuantity                                projectId
## 1                        NA DOI: https://doi.org/10.5399/osu/cat_osac.5.1.4647
## 2                        NA DOI: https://doi.org/10.5399/osu/cat_osac.5.1.4647
## 3                        NA DOI: https://doi.org/10.5399/osu/cat_osac.5.1.4647
##   level0Gid   level0Name level1Gid level1Name   level2Gid level2Name level3Gid
## 1      USA United States  USA.38_1   Oregon USA.38.20_1      Lane      NA
## 2      USA United States  USA.38_1   Oregon USA.38.9_1  Deschutes    NA
## 3      USA United States  USA.38_1   Oregon USA.38.35_1  Wheeler     NA
##   level3Name iucnRedListCategory eventType
## 1      NA              NE      NA
## 2      NA              NE      NA
## 3      NA              NE      NA
```

Clean dataset using regex

I will clean and update the associatedTaxa column which is supposed to have the plant name

1. Only keep the plant names in the string

```
sample(oba_2018_2019$associatedTaxa, size = 10)
```

```
## [1] ""
## [2] "foraging on : \"Hypochaeris radicata\""
## [3] "\"foraging on\" : \"Daucus carota\""
## [4] ""
## [5] ""
## [6] ""
## [7] ""
## [8] "foraging on : \"Acer circinatum\""
## [9] "foraging on : \"Asteraceae\""
## [10] "foraging on : \"Cucurbitaceae\""
```

*# using regex to select for the pattern to keep rather than the pattern to drop
noticed that the plant name is always bounded by the second double quotes ""*

Define the regex pattern

```
pattern <- '(.*)"(?<plant>.*)"$'
```

Use str_match() to extract the match

```
match <- str_match(oba_2018_2019$associatedTaxa, pattern)
plant_name_list <- match[, "plant"]
plant_name_list[is.na(plant_name_list)] <- ""
oba_2018_2019$associatedTaxa <- plant_name_list
sample(oba_2018_2019$associatedTaxa, size = 10)
```

```
## [1] ""                      "Ranunculaceae"        "Scandix pecten-veneris"
## [4] ""                      ""                      "Ceanothus"
## [7] "Malus"                 ""                      ""
## [10] "Claytonia"
```

2. Remove all weed names and keep only one plant name

```
cat("We have these many weeds:", sum(grepl("Weedy", oba_2018_2019$associatedTaxa)), "\n")
```

```
## We have these many weeds: 3
oba_2018_2019$associatedTaxa <- gsub("Weedy yellow comp.", "", oba_2018_2019$associatedTaxa)
oba_2018_2019$associatedTaxa <- gsub("Weedy", "", oba_2018_2019$associatedTaxa)
cat("Now we have these many weeds:", sum(grepl("Weedy", oba_2018_2019$associatedTaxa)), "\n")

## Now we have these many weeds: 0
# get rid of ;
cat("We have these many ';"', sum(grepl(";", oba_2018_2019$associatedTaxa)), "\n")

## We have these many ';' 11
oba_2018_2019$associatedTaxa[grepl(";", oba_2018_2019$associatedTaxa)] <- sapply(
  strsplit(
    oba_2018_2019$associatedTaxa[grepl(";", oba_2018_2019$associatedTaxa)], ";"
  ),
  function(x) x[1]
)
cat("We have these many ';"', sum(grepl(";", oba_2018_2019$associatedTaxa)), "\n")

## We have these many ';' 0
```

Add more information

1. Add plant resolution

We want to make a new column called plantResolution and assign it to “Family”, “Genus” or “Species” depending on the level of resolution associated taxa is resolved to.

```
oba_2018_2019$plantTaxaWordCount <- str_count(oba_2018_2019$associatedTaxa, "\\w+")
head(oba_2018_2019$associatedTaxa, 10)
```

```
## [1] "Vaccinium ovatum" "" "Whipplea modesta"
## [5] "Whipplea modesta" "" ""
## [9] "" "Horkelia fusca"
```

```
head(oba_2018_2019$plantTaxaWordCount, 10)
```

```
## [1] 2 0 0 2 2 0 0 0 0 2
```

```
summary(oba_2018_2019$plantTaxaWordCount)
```

```
##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.000  0.000   1.000   1.084  2.000   4.000
```

```
plant_resolution_list <- rep(NA, nrow(oba_2018_2019))
```

```
for (i in seq_along(plant_resolution_list)) {
  word_count <- oba_2018_2019$plantTaxaWordCount[i]
  plant_name <- oba_2018_2019$associatedTaxa[i]

  if (word_count == 1) { # can be genus or family
    if (grepl("aceae$", plant_name)) { # family
      plant_resolution_list[i] <- "family"
    } else {
      plant_resolution_list[i] <- "genus"
    }
  } else if (word_count > 1) { # it is species/subspecies
```

```

    plant_resolution_list[i] <- "species"
  }
}
oba_2018_2019$plantResolution <- plant_resolution_list
head(oba_2018_2019$associatedTaxa, 10)

## [1] "Vaccinium ovatum" "" "" "Whipplea modesta"
## [5] "Whipplea modesta" "" "" ""
## [9] "" "Horkelia fusca"
head(oba_2018_2019$plantTaxaWordCount, 10)

## [1] 2 0 0 2 2 0 0 0 0 2
head(oba_2018_2019$plantResolution, 10)

## [1] "species" NA NA "species" "species" NA NA
## [8] NA NA "species"

```

2. Add plant genus

Create a new column called plantGenus that is the genus if the associated taxa was resolved to species or genus, and NA if it was resolved to family.

```

plant_genus_list <- rep(NA, nrow(oba_2018_2019))

for (i in seq_along(plant_genus_list)) {
  plant_res <- oba_2018_2019$plantResolution[i]
  plant_name <- oba_2018_2019$associatedTaxa[i]

  if (!is.na(plant_res) && (plant_res == "genus")) {
    plant_genus_list[i] <- plant_name
  } else if (!is.na(plant_res) && (plant_res == "species")) {
    plant_genus_list[i] <- strsplit(plant_name, " ")[[1]][1]
  }
}

oba_2018_2019$plantGenus <- plant_genus_list
head(oba_2018_2019$associatedTaxa, 10)

## [1] "Vaccinium ovatum" "" "" "Whipplea modesta"
## [5] "Whipplea modesta" "" "" ""
## [9] "" "Horkelia fusca"
head(oba_2018_2019$plantTaxaWordCount, 10)

## [1] 2 0 0 2 2 0 0 0 0 2
head(oba_2018_2019$plantResolution, 10)

## [1] "species" NA NA "species" "species" NA NA
## [8] NA NA "species"
head(oba_2018_2019$plantGenus, 10)

## [1] "Vaccinium" NA NA "Whipplea" "Whipplea" NA
## [7] NA NA NA "Horkelia"

```


Save/Read the new dataframe as csv

Drop the columns that we are not going to use

```
col_to_save <- c(
  "gbifID", "genericName", "specificEpithet", "species", "sex",
  "taxonRank", "eventDate", "year", "month", "samplingProtocol",
  "associatedTaxa", "county", "decimalLatitude", "decimalLongitude",
  "plantResolution", "plantGenus"
)
oba_2018_2019 <- oba_2018_2019[, col_to_save]
colnames(oba_2018_2019)

## [1] "gbifID"          "genericName"      "specificEpithet"  "species"
## [5] "sex"             "taxonRank"        "eventDate"        "year"
## [9] "month"           "samplingProtocol" "associatedTaxa"    "county"
## [13] "decimalLatitude" "decimalLongitude" "plantResolution"  "plantGenus"
```

Save

```
write.csv(oba_2018_2019, "data/cleaned_oba_2018_2019.csv", row.names = FALSE)
```

Read

```
oba_2018_2019 <- read.csv("data/cleaned_oba_2018_2019.csv")

# check if all values of column gbifID are unique
unique_gbifID <- length(unique(oba_2018_2019$gbifID))
total_gbifID <- nrow(oba_2018_2019)

if (unique_gbifID == total_gbifID) {
  print("All values of column 'gbifID' are unique.")
} else {
  print("Not all values of column 'gbifID' are unique.")
}
```

```
## [1] "All values of column 'gbifID' are unique."
```

This means we can use gbifID as a unique identifier

1. Sexual Dimorphism

```
# check the TOTAL PERCENTAGE of MALE vs FEMALE bees caught
total_male <- sum(oba_2018_2019$sex == "MALE", na.rm = TRUE)
total_female <- sum(oba_2018_2019$sex == "FEMALE", na.rm = TRUE)
total_bees <- nrow(oba_2018_2019)

percentage_male <- (total_male / total_bees) * 100
percentage_female <- (total_female / total_bees) * 100

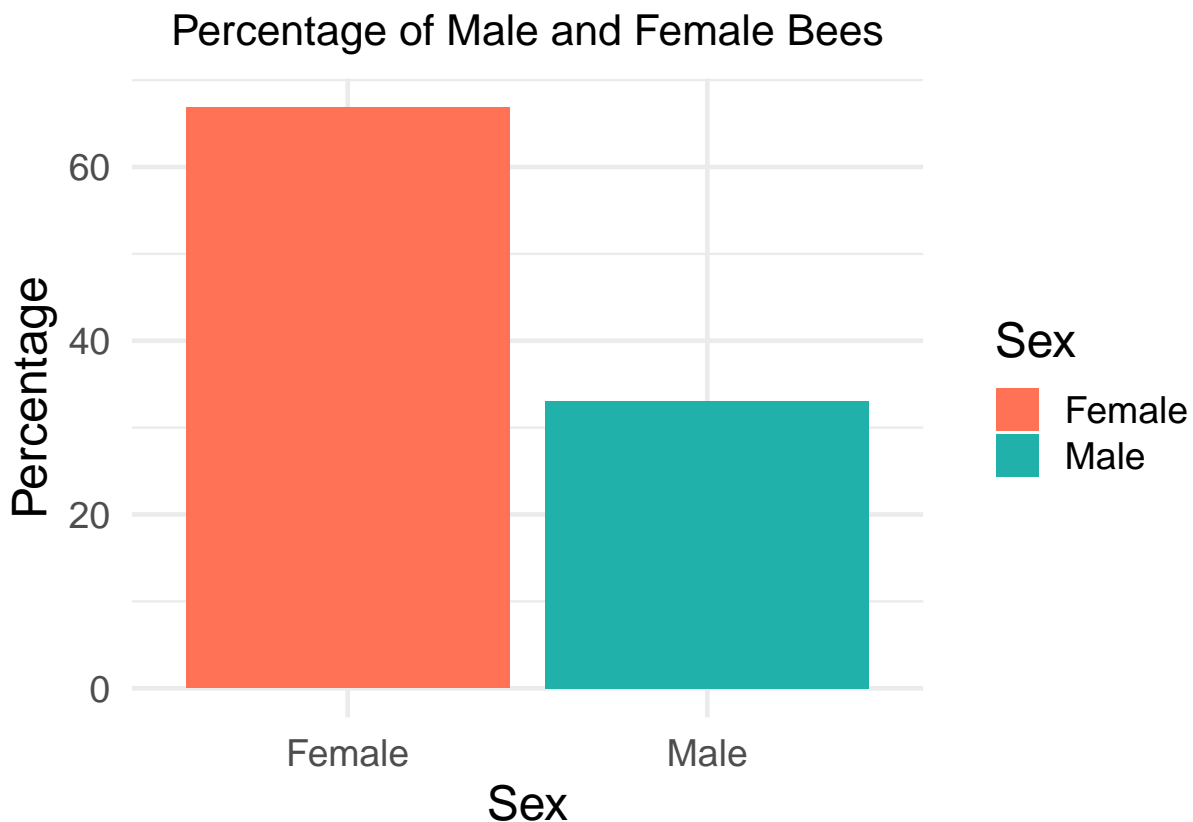
# Create a data frame for the plot
sex_per_df <- data.frame(
  Sex = c("Female", "Male"),
```

```

    Percentage = c(percentage_female, percentage_male)
  )

  # Create a bar plot with ggplot
  p_title <- "Percentage of Male and Female Bees"
  p <- ggplot(sex_per_df, aes(x = Sex, y = Percentage, fill = Sex)) +
    geom_bar(stat = "identity") +
    scale_fill_manual(values = c("coral1", "lightseagreen")) +
    labs(x = "Sex", y = "Percentage") +
    ggtitle(p_title) +
    theme_minimal(18) +
    theme(
      axis.text = element_text(size = 14),
      plot.title = element_text(hjust = 0.5, size = 16)
    )
  p

```



```

ggsave(glue("plots/{p_title}.png"), dpi = 300, plot = p)

## Saving 6.5 x 4.5 in image

# Calculate the percentage of bees caught using different sampling patterns
sampling_patterns <- unique(oba_2018_2019$samplingProtocol)
percentage_per_pattern <- sapply(sampling_patterns, function(pattern) {
  total_bees <- sum(oba_2018_2019$samplingProtocol == pattern, na.rm = TRUE)
  percentage <- (total_bees / nrow(oba_2018_2019)) * 100
  return(percentage)
})

```

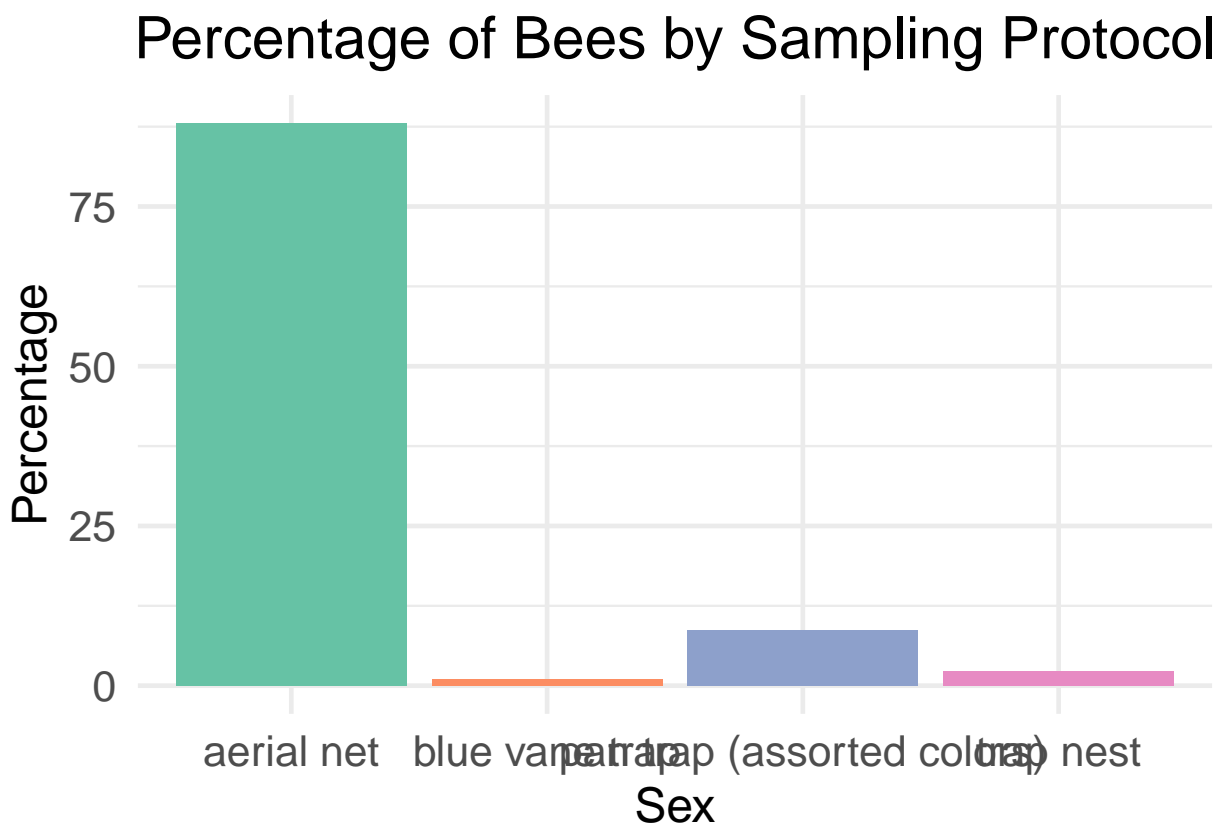
```

})

sampling_per_df <- data.frame(
  patterns = sampling_patterns,
  percentage = percentage_per_pattern
)

# Create a stacked bar plot with percentages
p_title <- "Percentage of Bees by Sampling Protocol"
p <- ggplot(sampling_per_df, aes(x = patterns, y = percentage, fill = patterns)) +
  geom_bar(stat = "identity", position = "stack", show.legend = F) +
  labs(x = "Sex", y = "Percentage", fill = "Sampling Protocol") +
  ggtitle(p_title) +
  theme_minimal(18) +
  theme(
    axis.text = element_text(size = 16),
    plot.title = element_text(hjust = 0.5)
  ) +
  scale_fill_brewer(palette = "Set2")
p

```



```

ggsave(glue("plots/{p_title}.png"), dpi = 300, plot = p)

```

```

## Saving 6.5 x 4.5 in image

```

```

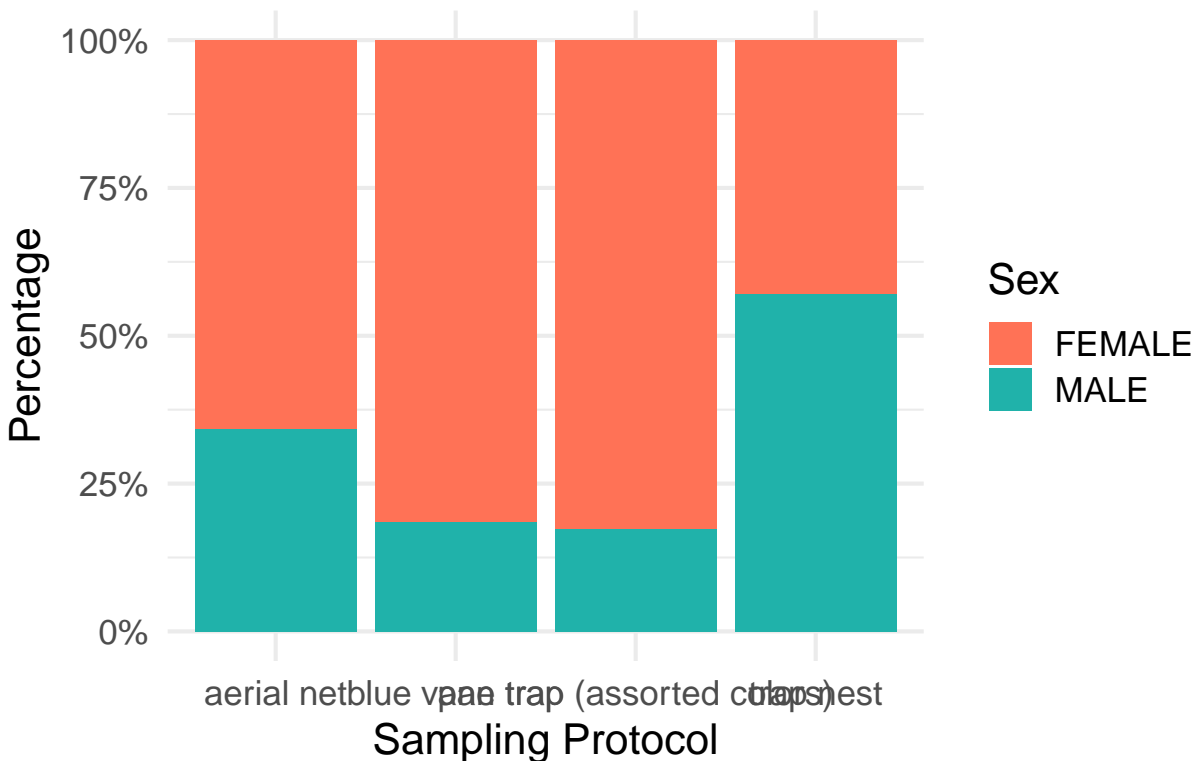
p_title <- "Percentage of Male and Female Bees by Sampling Protocol"
p <- oba_2018_2019 %>%
  filter(sex != "") %>%
  group_by(sex, samplingProtocol) %>%
  summarise(n_group = n()) %>%
  ungroup() %>%
  group_by(samplingProtocol) %>%
  mutate(
    n_sampling = sum(n_group),
    percentage = n_group / n_sampling
  ) %>%
  ggplot(aes(x = samplingProtocol, y = percentage, fill = sex)) +
  geom_bar(stat = "identity", position = "stack") +
  labs(x = "Sampling Protocol", y = "Percentage", fill = "Sex") +
  ggtitle(p_title) +
  scale_fill_manual(values = c("coral1", "lightseagreen")) +
  scale_y_continuous(labels = scales::percent_format()) +
  theme_minimal(16) +
  theme(
    axis.text = element_text(size = 13),
    plot.title = element_text(hjust = 0.5)
  )

```

`summarise()` has grouped output by 'sex'. You can override using the `groups`
argument.

p

Percentage of Male and Female Bees by Sampling Protocol



```
ggsave(glue("plots/{p_title}.png"), dpi = 300, plot = p)
```

```
## Saving 6.5 x 4.5 in image
```

```
# Generate the plot
```

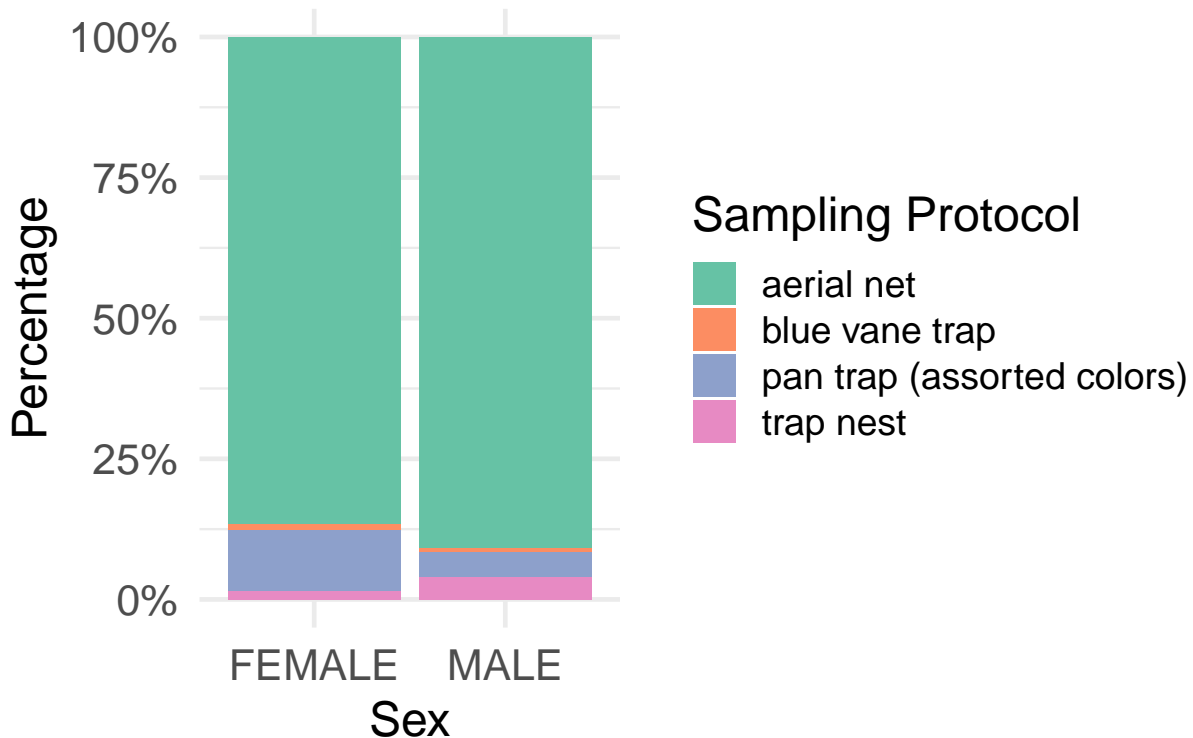
```
p_title <- "Percentage of Bees by Sampling Protocol_2"
```

```
p <- oba_2018_2019 %>%  
  filter(sex != "") %>%  
  group_by(sex, samplingProtocol) %>%  
  summarise(n_group = n()) %>%  
  ungroup() %>%  
  group_by(sex) %>%  
  mutate(  
    n_sampling = sum(n_group),  
    percentage = n_group / n_sampling  
  ) %>%  
  ggplot(aes(x = sex, y = percentage, fill = samplingProtocol)) +  
  geom_bar(stat = "identity", position = "stack") +  
  labs(x = "Sex", y = "Percentage", fill = "Sampling Protocol") +  
  ggtitle(p_title) +  
  scale_fill_brewer(palette = "Set2") +  
  scale_y_continuous(labels = scales::percent_format()) +  
  theme_minimal(18) +  
  theme(  
    axis.text = element_text(size = 16),  
    plot.title = element_text(hjust = 0.5)  
  )
```

```
## `summarise()` has grouped output by 'sex'. You can override using the `.groups`  
## argument.
```

```
p
```

ntage of Bees by Sampling Protocol_2



```
ggsave(glue("plots/{p_title}.png"), dpi = 300, plot = p)
```

```
## Saving 6.5 x 4.5 in image
```

2. Climate and Foraging Patterns

Lane county

```
# get climate data for lane county

# Read csv files
avg_temp <- read.csv("data/lane_county_avg_temp_2018_2019.csv", skip = 3, header = 4)
avg_ppt <- read.csv("data/lane_county_precipitation_2018_2019.csv", skip = 3, header = 4)

# rename columns
colnames(avg_temp)[colnames(avg_temp) == "Date"] <- "date"
colnames(avg_temp)[colnames(avg_temp) == "Value"] <- "temp_f"
colnames(avg_ppt)[colnames(avg_ppt) == "Date"] <- "date"
colnames(avg_ppt)[colnames(avg_ppt) == "Value"] <- "ppt_inch"

# add column 'year' and 'month' from the first four string character from the 'Date' column
avg_temp$year <- substr(avg_temp$date, 1, 4)
avg_temp$month <- substr(avg_temp$date, 5, 6)
avg_temp$year <- as.numeric(avg_temp$year)
avg_temp$month <- as.numeric(avg_temp$month)
avg_temp$temp_c <- (avg_temp$temp_f - 32) * (5 / 9) # add celsius
```

```
head(avg_temp)
```

```
##      date temp_f year month   temp_c
## 1 201801   42.0 2018     1  5.555556
## 2 201802   38.2 2018     2  3.444444
## 3 201803   41.3 2018     3  5.166667
## 4 201804   46.7 2018     4  8.166667
## 5 201805   55.0 2018     5 12.777778
## 6 201806   58.4 2018     6 14.666667
```

```
avg_ppt$year <- substr(avg_ppt$date, 1, 4)
avg_ppt$month <- substr(avg_ppt$date, 5, 6)
avg_ppt$year <- as.numeric(avg_ppt$year)
avg_ppt$month <- as.numeric(avg_ppt$month)
avg_ppt$ppt_cm <- avg_ppt$ppt_inch * 2.54 # add cm ppt
head(avg_ppt)
```

```
##      date ppt_inch year month  ppt_cm
## 1 201801     8.41 2018     1 21.3614
## 2 201802     4.73 2018     2 12.0142
## 3 201803     5.84 2018     3 14.8336
## 4 201804     6.55 2018     4 16.6370
## 5 201805     0.83 2018     5  2.1082
## 6 201806     1.86 2018     6  4.7244
```

```
lane_obo_2018_2019 <- obo_2018_2019 %>%
  filter(county == "Lane")
```

```
# find the count of total number of 'gbifID' by year and month in lane_obo_2018_2019
net_count_per_month <- lane_obo_2018_2019 %>%
  group_by(year, month) %>%
  summarise(count = n_distinct(gbifID))
```

```
## `summarise()` has grouped output by 'year'. You can override using the
## `.groups` argument.
```

```
species_count_per_month <- lane_obo_2018_2019 %>%
  filter(species != "") %>%
  group_by(year, month) %>%
  summarise(count = n_distinct(species))
```

```
## `summarise()` has grouped output by 'year'. You can override using the
## `.groups` argument.
```

```
# get and add ppt_cm information from avg_ppt df by year and month
net_count_per_month <- net_count_per_month %>%
  left_join(avg_ppt, by = c("year", "month")) %>%
  left_join(avg_temp, by = c("year", "month"))
```

```
species_count_per_month <- species_count_per_month %>%
  left_join(avg_ppt, by = c("year", "month")) %>%
  left_join(avg_temp, by = c("year", "month"))
```

```
head(net_count_per_month)
```

```
## # A tibble: 6 x 9
## # Groups:   year [1]
```

```
##   year month count date.x ppt_inch ppt_cm date.y temp_f temp_c
##   <dbl> <dbl> <int> <int>    <dbl>  <dbl> <int> <dbl> <dbl>
## 1  2018     4    42 201804     6.55 16.6   201804  46.7   8.17
## 2  2018     5    68 201805     0.83  2.11   201805   55   12.8
## 3  2018     6    55 201806     1.86  4.72   201806  58.4   14.7
## 4  2018     7    38 201807     0.01  0.0254 201807  67.1   19.5
## 5  2018     8    86 201808     0.13  0.330   201808  65.6   18.7
## 6  2018     9    21 201809     0.41  1.04   201809  58.6   14.8
```

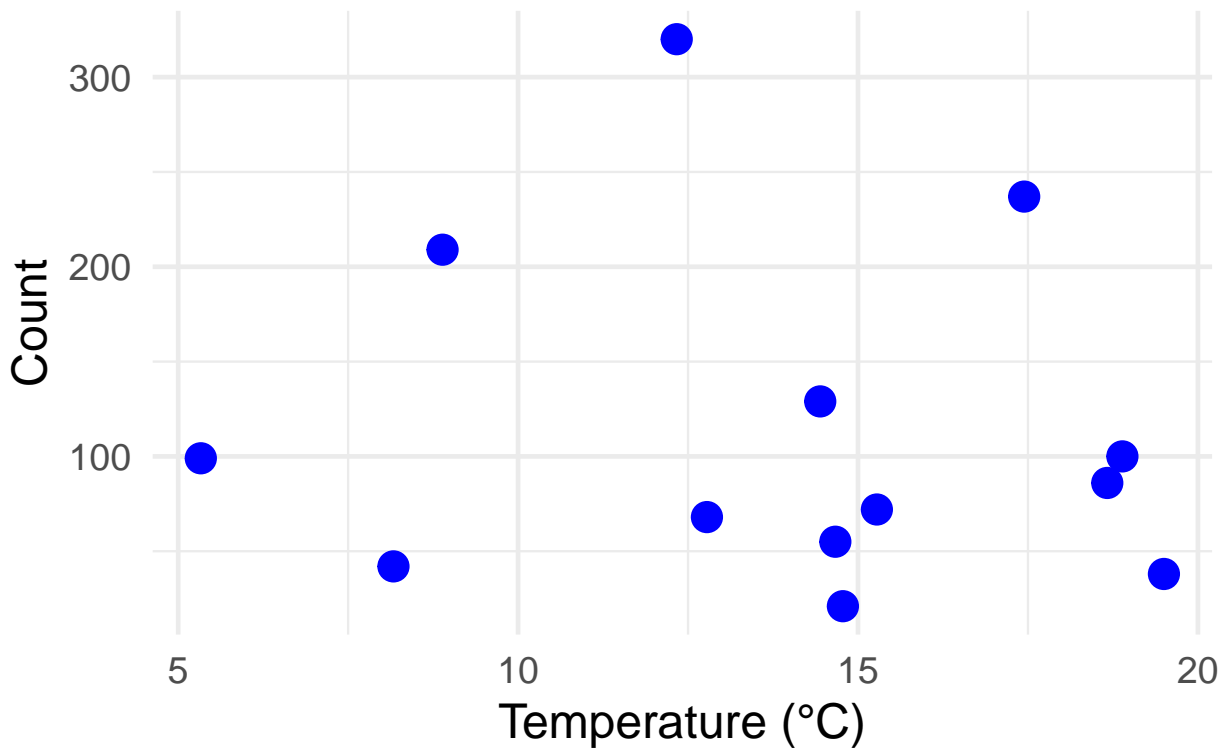
```
head(species_count_per_month)
```

```
## # A tibble: 6 x 9
## # Groups:   year [1]
##   year month count date.x ppt_inch ppt_cm date.y temp_f temp_c
##   <dbl> <dbl> <int> <int>    <dbl>  <dbl> <int> <dbl> <dbl>
## 1  2018     4     8 201804     6.55 16.6   201804  46.7   8.17
## 2  2018     5    14 201805     0.83  2.11   201805   55   12.8
## 3  2018     6    15 201806     1.86  4.72   201806  58.4   14.7
## 4  2018     7     8 201807     0.01  0.0254 201807  67.1   19.5
## 5  2018     8    21 201808     0.13  0.330   201808  65.6   18.7
## 6  2018     9    10 201809     0.41  1.04   201809  58.6   14.8
```

Plot wrt temp

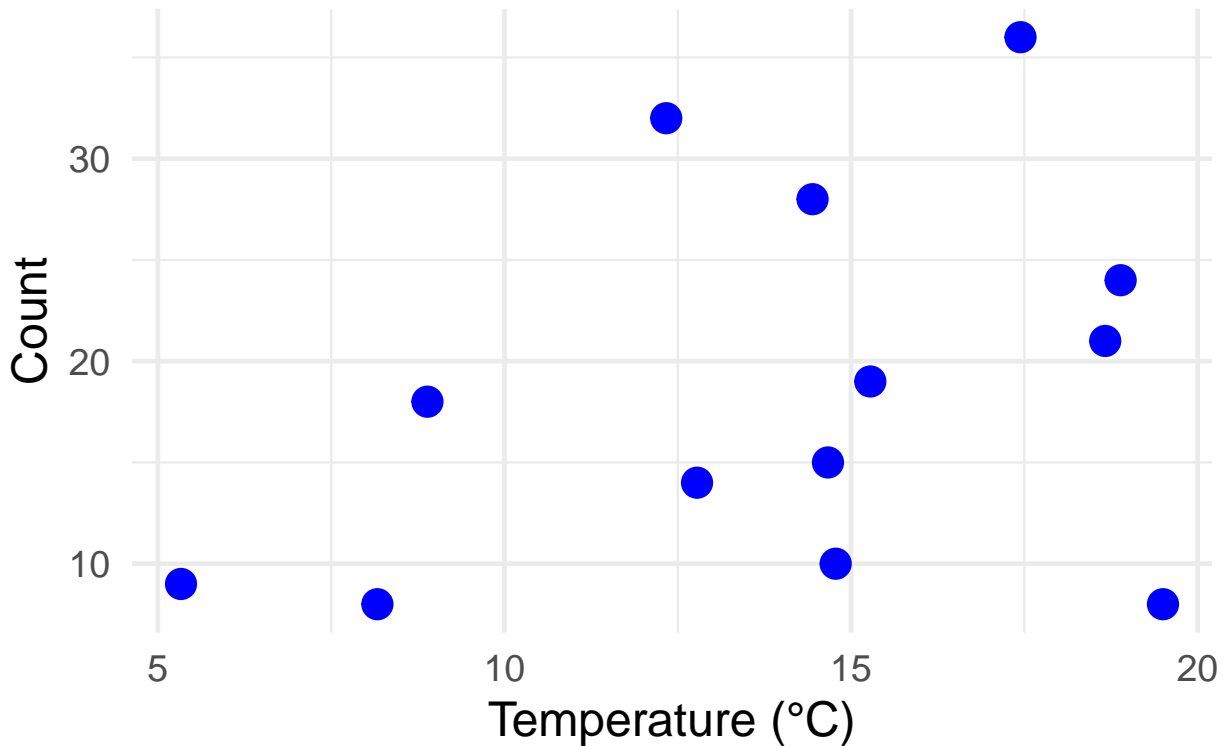
```
# Scatter plot
ggplot(net_count_per_month) +
  geom_point(aes(x = temp_c, y = count), size = 5, color = "blue") +
  labs(x = "Temperature (°C)", y = "Count") +
  ggtitle("Net Count vs Mean Temp") +
  theme_minimal(18) +
  theme(plot.title = element_text(hjust = 0.5))
```


Net Count vs Mean Temp



```
ggplot(species_count_per_month) +  
  geom_point(aes(x = temp_c, y = count), size = 5, color = "blue") +  
  labs(x = "Temperature (°C)", y = "Count") +  
  ggtitle("Species Count vs Mean Temp") +  
  theme_minimal(18) +  
  theme(plot.title = element_text(hjust = 0.5))
```

Species Count vs Mean Temp



```
lin_reg <- lm(species_count_per_month$count ~ species_count_per_month$temp_c)
lin_reg_slope <- summary(lin_reg)$coefficients[2]
lin_reg_pval <- summary(lin_reg)$coefficients[7]
summary(lin_reg)

##
## Call:
## lm(formula = species_count_per_month$count ~ species_count_per_month$temp_c)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.634  -4.143  -1.032   3.030  14.851
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      8.5498     8.7324   0.979   0.349
## species_count_per_month$temp_c  0.7223     0.5999   1.204   0.254
##
## Residual standard error: 9.1 on 11 degrees of freedom
## Multiple R-squared:  0.1164, Adjusted R-squared:  0.03613
## F-statistic: 1.45 on 1 and 11 DF,  p-value: 0.2538

print(lin_reg_slope)

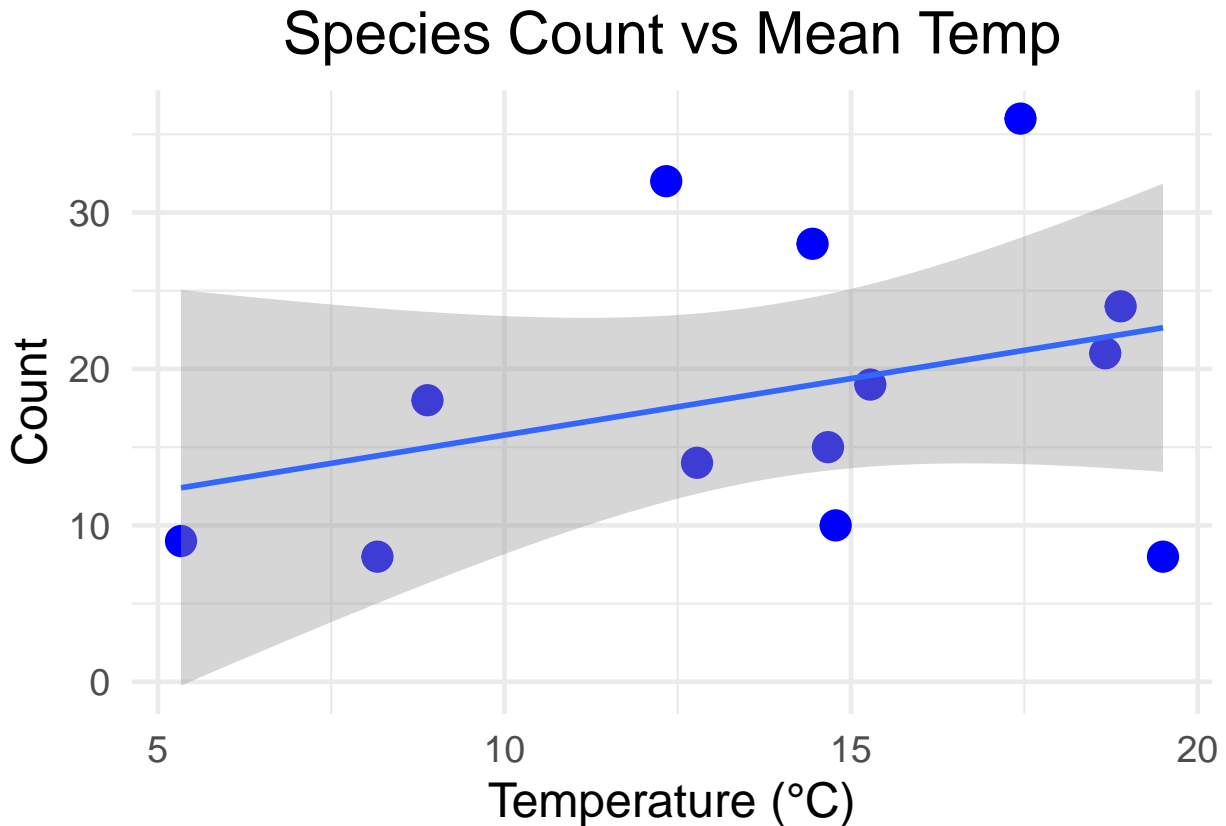
## [1] 0.7222794

print(lin_reg_pval)
```

```
## [1] 0.3485819
```

```
ggplot(species_count_per_month, aes(x = temp_c, y = count)) +  
  geom_point(size = 5, color = "blue") +  
  labs(x = "Temperature (°C)", y = "Count") +  
  ggtitle("Species Count vs Mean Temp") +  
  theme_minimal(18) +  
  geom_smooth(method = "lm", se = TRUE) +  
  theme(plot.title = element_text(hjust = 0.5))
```

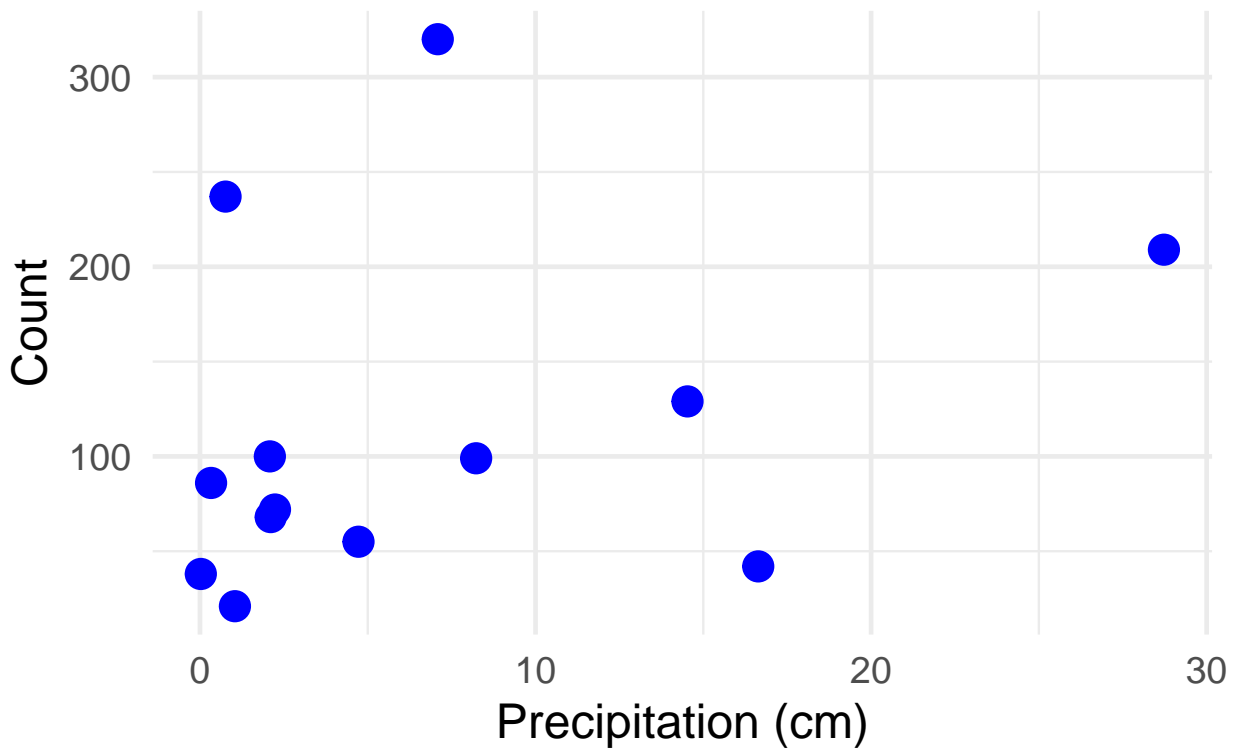
```
## `geom_smooth()` using formula = 'y ~ x'
```



Plot wrt ppt

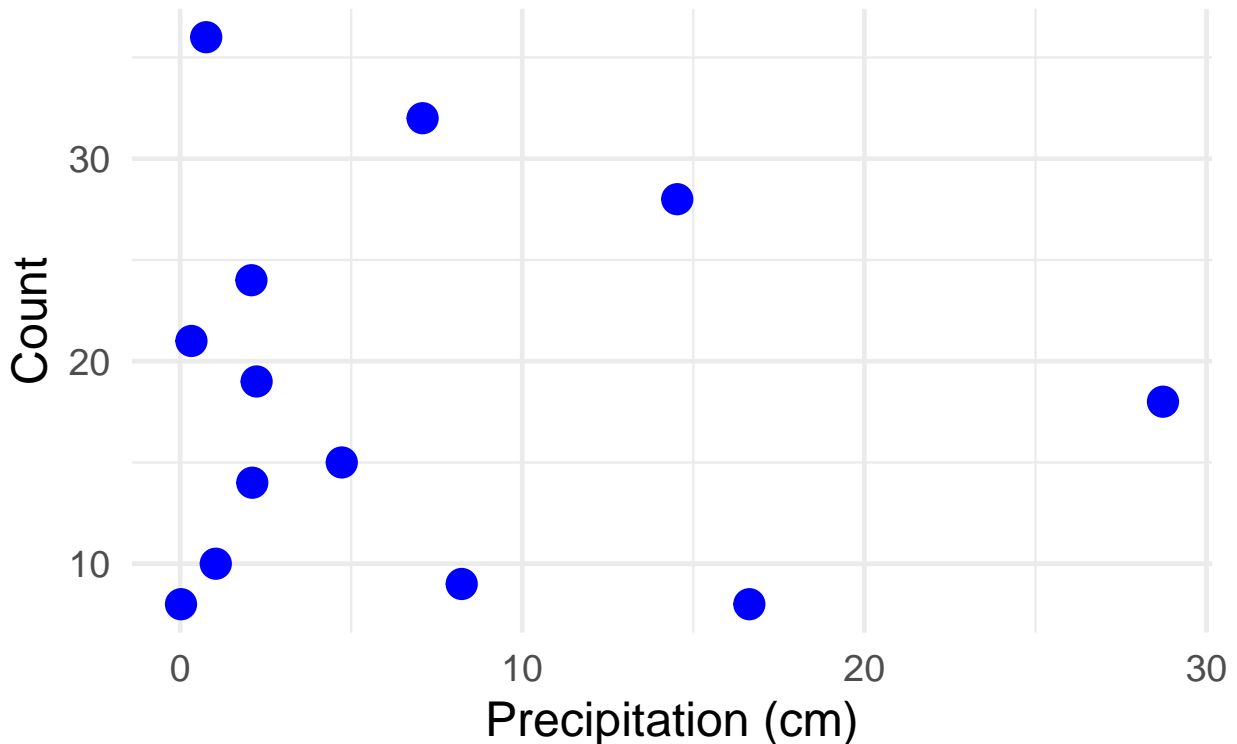
```
# Scatter plot  
ggplot(net_count_per_month) +  
  geom_point(aes(x = ppt_cm, y = count), size = 5, color = "blue") +  
  labs(x = "Precipitation (cm)", y = "Count") +  
  ggtitle("Net Count vs Mean Precipitation") +  
  theme_minimal(18) +  
  theme(plot.title = element_text(hjust = 0.5))
```

Net Count vs Mean Precipitation



```
ggplot(species_count_per_month) +  
  geom_point(aes(x = ppt_cm, y = count), size = 5, color = "blue") +  
  labs(x = "Precipitation (cm)", y = "Count") +  
  ggtitle("Species Count vs Mean Precipitation") +  
  theme_minimal(18) +  
  theme(plot.title = element_text(hjust = 0.5))
```

Species Count vs Mean Precipitation



```
lin_reg <- lm(net_count_per_month$count ~ species_count_per_month$ppt_cm)
lin_reg_slope <- summary(lin_reg)$coefficients[2]
lin_reg_pval <- summary(lin_reg)$coefficients[7]
summary(lin_reg)
```

```
##
## Call:
## lm(formula = net_count_per_month$count ~ species_count_per_month$ppt_cm)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -101.189  -52.249  -18.824    0.721   205.624
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      92.995     32.139   2.893  0.0146 *
## species_count_per_month$ppt_cm  3.017      3.023   0.998  0.3398
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 88.99 on 11 degrees of freedom
## Multiple R-squared:  0.08301,    Adjusted R-squared:  -0.0003575
## F-statistic: 0.9957 on 1 and 11 DF,  p-value: 0.3398
```

```
print(lin_reg_slope)
```

```
## [1] 3.01698
```

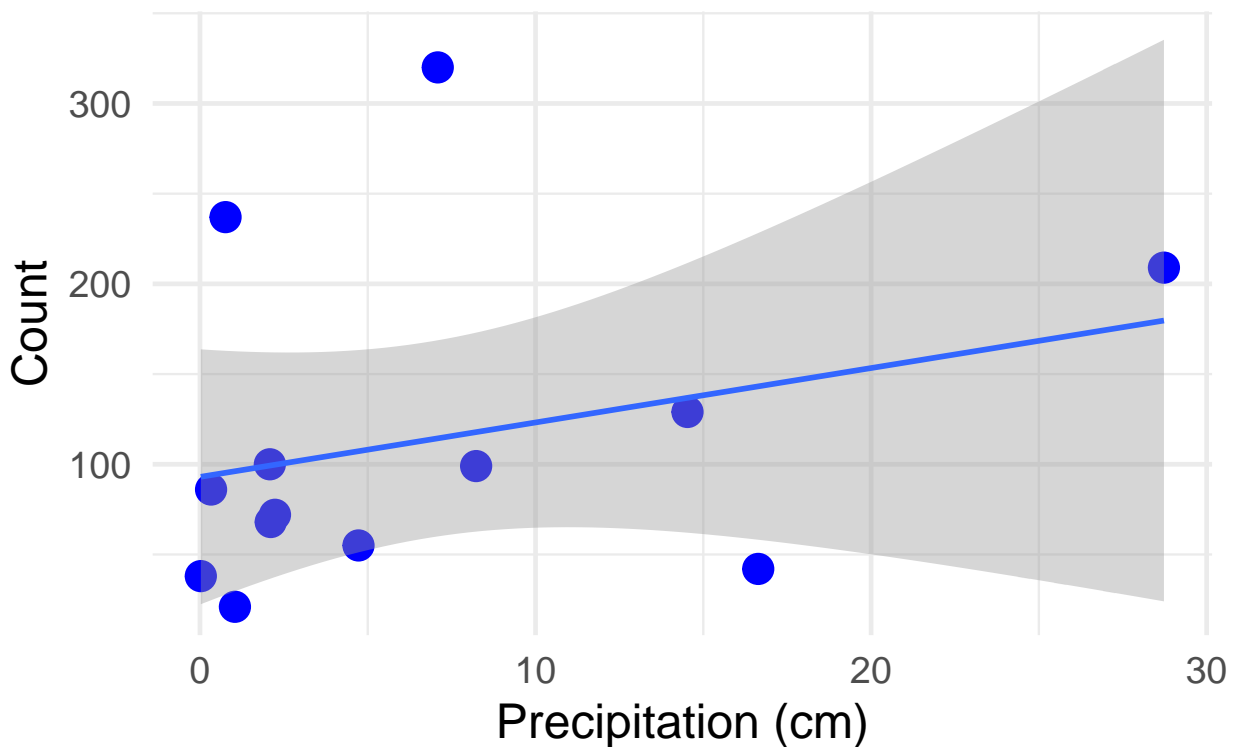
```
print(lin_reg_pval)
```

```
## [1] 0.01461421
```

```
ggplot(net_count_per_month, aes(x = ppt_cm, y = count)) +  
  geom_point(size = 5, color = "blue") +  
  labs(x = "Precipitation (cm)", y = "Count") +  
  ggtitle("Net Count vs Mean Temp") +  
  theme_minimal(18) +  
  geom_smooth(method = "lm", se = TRUE) +  
  theme(plot.title = element_text(hjust = 0.5))
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

Net Count vs Mean Temp



```
### Oregon
```

```
# get climate data for lane county
```

```
# Read csv files
```

```
avg_temp <- read.csv("data/oregon_avg_temp_2018_2019.csv", skip = 3, header = 4)
```

```
avg_ppt <- read.csv("data/oregon_precipitation_2018_2019.csv", skip = 3, header = 4)
```

```
# rename columns
```

```
colnames(avg_temp)[colnames(avg_temp) == "Date"] <- "date"
```

```
colnames(avg_temp)[colnames(avg_temp) == "Value"] <- "temp_f"
```

```
colnames(avg_ppt)[colnames(avg_ppt) == "Date"] <- "date"
```

```
colnames(avg_ppt)[colnames(avg_ppt) == "Value"] <- "ppt_inch"
```

```

# add column 'year' and 'month' from the first four string character from the 'Date' column
avg_temp$year <- substr(avg_temp$date, 1, 4)
avg_temp$month <- substr(avg_temp$date, 5, 6)
avg_temp$year <- as.numeric(avg_temp$year)
avg_temp$month <- as.numeric(avg_temp$month)
avg_temp$temp_c <- (avg_temp$temp_f - 32) * (5 / 9) # add celsius
head(avg_temp)

```

```

##      date temp_f year month    temp_c
## 1 201801   37.5 2018     1  3.055556
## 2 201802   33.2 2018     2  0.666667
## 3 201803   38.4 2018     3  3.555556
## 4 201804   45.0 2018     4  7.222222
## 5 201805   55.4 2018     5 13.000000
## 6 201806   58.5 2018     6 14.722222

```

```

avg_ppt$year <- substr(avg_ppt$date, 1, 4)
avg_ppt$month <- substr(avg_ppt$date, 5, 6)
avg_ppt$year <- as.numeric(avg_ppt$year)
avg_ppt$month <- as.numeric(avg_ppt$month)
avg_ppt$ppt_cm <- avg_ppt$ppt_inch * 2.54 # add cm ppt
head(avg_ppt)

```

```

##      date ppt_inch year month ppt_cm
## 1 201801    4.14 2018     1 10.5156
## 2 201802    2.29 2018     2  5.8166
## 3 201803    3.42 2018     3  8.6868
## 4 201804    3.26 2018     4  8.2804
## 5 201805    1.27 2018     5  3.2258
## 6 201806    1.13 2018     6  2.8702

```

```

# find the count of total number of 'gbifID' by year and month in oba_2018_2019
net_count_per_month <- oba_2018_2019 %>%
  group_by(year, month) %>%
  summarise(count = n_distinct(gbifID))

```

```

## `summarise()` has grouped output by 'year'. You can override using the
## `.groups` argument.

```

```

species_count_per_month <- oba_2018_2019 %>%
  filter(species != "") %>%
  group_by(year, month) %>%
  summarise(count = n_distinct(species))

```

```

## `summarise()` has grouped output by 'year'. You can override using the
## `.groups` argument.

```

```

# get and add ppt_cm information from avg_ppt df by year and month
net_count_per_month <- net_count_per_month %>%
  left_join(avg_ppt, by = c("year", "month")) %>%
  left_join(avg_temp, by = c("year", "month"))

```

```

species_count_per_month <- species_count_per_month %>%
  left_join(avg_ppt, by = c("year", "month")) %>%
  left_join(avg_temp, by = c("year", "month"))

```

```
head(net_count_per_month)
```

```
## # A tibble: 6 x 9
## # Groups:   year [1]
##   year month count date.x ppt_inch ppt_cm date.y temp_f temp_c
##   <dbl> <dbl> <int> <int>   <dbl> <dbl> <int> <dbl> <dbl>
## 1  2017     3     1    NA      NA     NA     NA     NA     NA
## 2  2017     5    31    NA      NA     NA     NA     NA     NA
## 3  2017     6    14    NA      NA     NA     NA     NA     NA
## 4  2017     7    70    NA      NA     NA     NA     NA     NA
## 5  2017     8    70    NA      NA     NA     NA     NA     NA
## 6  2017     9    43    NA      NA     NA     NA     NA     NA
```

```
head(species_count_per_month)
```

```
## # A tibble: 6 x 9
## # Groups:   year [1]
##   year month count date.x ppt_inch ppt_cm date.y temp_f temp_c
##   <dbl> <dbl> <int> <int>   <dbl> <dbl> <int> <dbl> <dbl>
## 1  2017     5     4    NA      NA     NA     NA     NA     NA
## 2  2017     6     7    NA      NA     NA     NA     NA     NA
## 3  2017     7    15    NA      NA     NA     NA     NA     NA
## 4  2017     8    20    NA      NA     NA     NA     NA     NA
## 5  2017     9    12    NA      NA     NA     NA     NA     NA
## 6  2017    10     2    NA      NA     NA     NA     NA     NA
```

Plot wrt temp

```
# Scatter plot
lin_reg <- lm(net_count_per_month$count ~ net_count_per_month$temp_c)
# lin_reg_slope <- summary(lin_reg)$coefficients[2]
# lin_reg_pval <- summary(lin_reg)$coefficients[7]
summary(lin_reg)

##
## Call:
## lm(formula = net_count_per_month$count ~ net_count_per_month$temp_c)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1516.5  -975.3  -422.3   278.2  3616.4
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -264.18     724.20  -0.365  0.71977
## net_count_per_month$temp_c    196.07     58.28   3.364  0.00368 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1528 on 17 degrees of freedom
## (8 observations deleted due to missingness)
## Multiple R-squared:  0.3997, Adjusted R-squared:  0.3644
## F-statistic: 11.32 on 1 and 17 DF,  p-value: 0.003681

p_title = "Oregon - Net Count vs Mean Monthly Temperature"
p <- ggplot(net_count_per_month, aes(x = temp_c, y = count)) +
```



```
geom_point(size = 5, color = "#ff0000") +
labs(x = "Temperature (°C)", y = "Total #Bees") +
ggtitle(p_title) +
theme_minimal(18) +
geom_smooth(method = "lm", se = TRUE, color="#ff0000") +
theme(plot.title = element_text(hjust = 0.5))
```

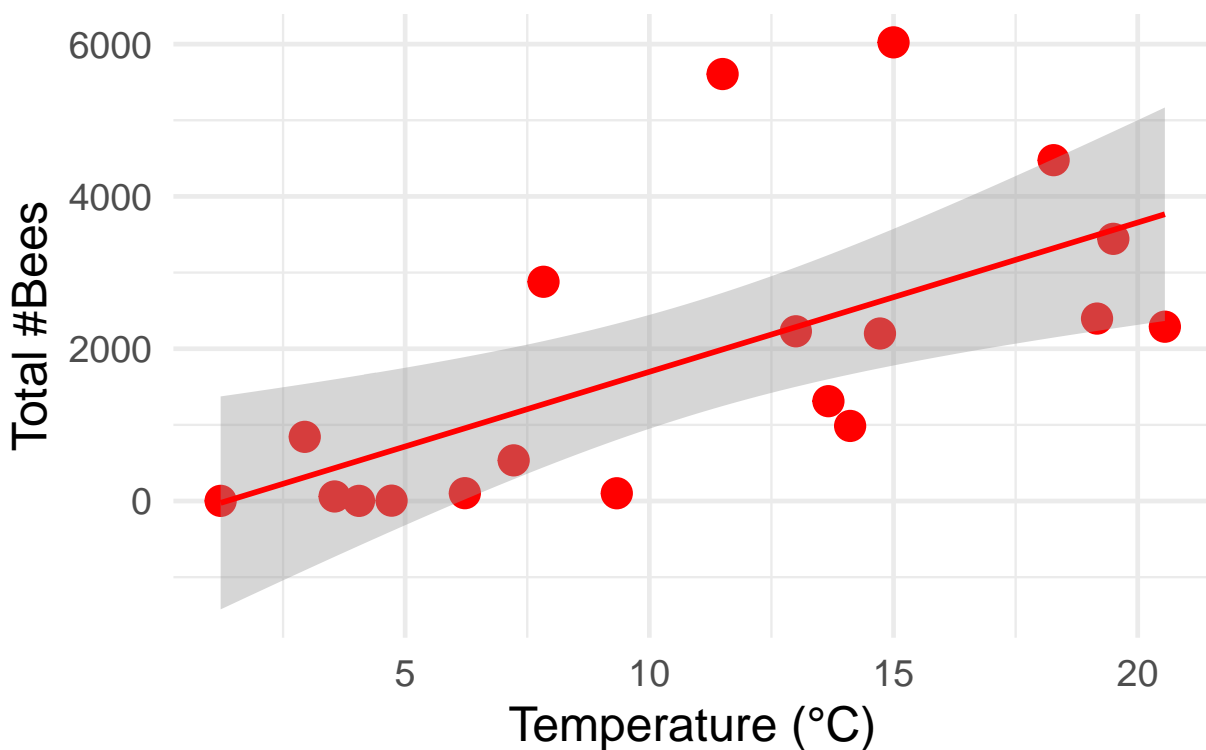
p

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 8 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 8 rows containing missing values (`geom_point()`).
```

Oregon – Net Count vs Mean Monthly Temperature



```
ggsave(glue("plots/{p_title}.png"), dpi = 300, plot = p)
```

```
## Saving 6.5 x 4.5 in image
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 8 rows containing non-finite values (`stat_smooth()`).
```

```
## Removed 8 rows containing missing values (`geom_point()`).
```

```
lin_reg <- lm(species_count_per_month$count ~ species_count_per_month$temp_c)
summary(lin_reg)
```

```
##
```

```
## Call:
```

```
## lm(formula = species_count_per_month$count ~ species_count_per_month$temp_c)
```

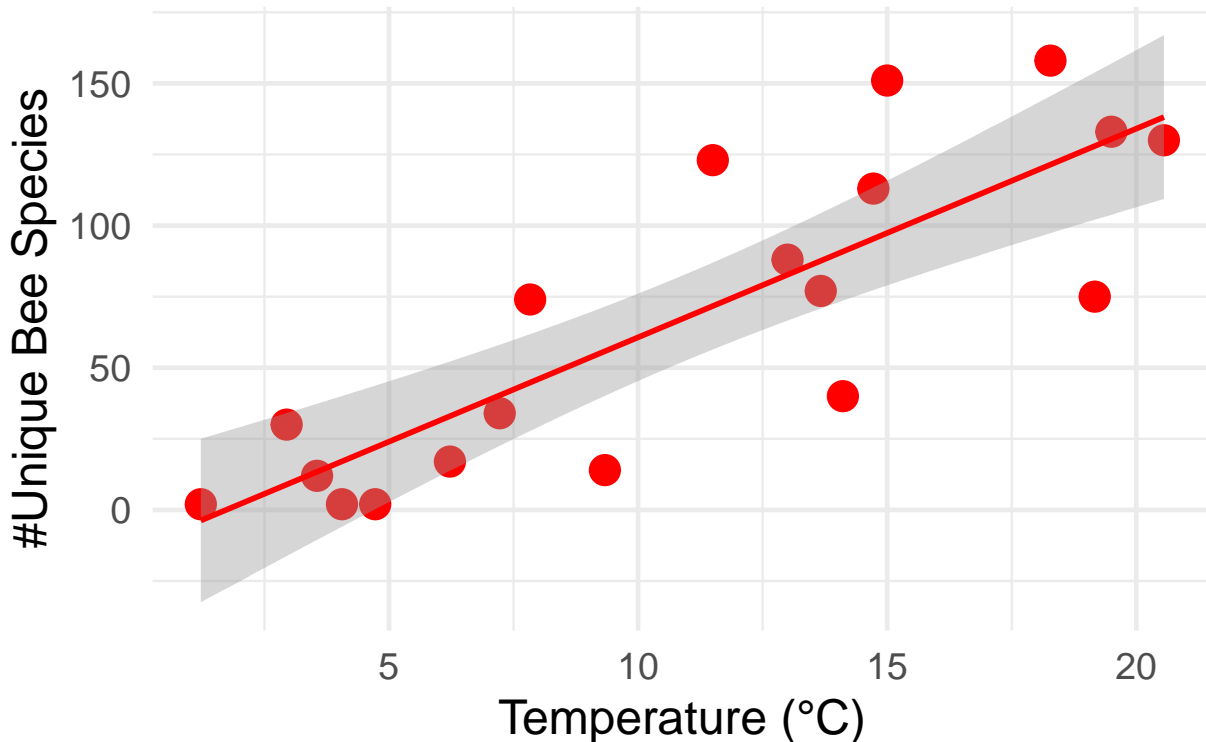
```
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -52.952 -15.521  -1.403   19.370   53.621
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -12.687     14.883  -0.852    0.406
## species_count_per_month$temp_c     7.338      1.198   6.126 1.12e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 31.4 on 17 degrees of freedom
## (7 observations deleted due to missingness)
## Multiple R-squared:  0.6883, Adjusted R-squared:  0.6699
## F-statistic: 37.53 on 1 and 17 DF,  p-value: 1.119e-05

p_title <- "Oregon - Species Count vs Mean Monthly Temperature"
p <- ggplot(species_count_per_month, aes(x = temp_c, y = count)) +
  geom_point(size = 5, color = "#ff0000") +
  labs(x = "Temperature (°C)", y = "#Unique Bee Species") +
  ggtitle(p_title) +
  theme_minimal(18) +
  geom_smooth(method = "lm", se = TRUE, color = "#ff0000") +
  theme(plot.title = element_text(hjust = 0.5))
p

## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 7 rows containing non-finite values (`stat_smooth()`).
## Warning: Removed 7 rows containing missing values (`geom_point()`).
```

Region – Species Count vs Mean Monthly Temperature



```
ggsave(glue("plots/{p_title}.png"), dpi = 300, plot = p)
```

```
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'

## Warning: Removed 7 rows containing non-finite values (`stat_smooth()`).
## Removed 7 rows containing missing values (`geom_point()`).
```

Plot wrt ppt

```
lin_reg <- lm(net_count_per_month$count ~ net_count_per_month$ppt_cm)
# lin_reg_slope <- summary(lin_reg)$coefficients[2]
# lin_reg_pval <- summary(lin_reg)$coefficients[7]
summary(lin_reg)
```

```
##
## Call:
## lm(formula = net_count_per_month$count ~ net_count_per_month$ppt_cm)
##
## Residuals:
```

##	Min	1Q	Median	3Q	Max
##	-2145.4	-1265.1	-377.3	585.1	4077.4

```
##
## Coefficients:
```

##		Estimate	Std. Error	t value	Pr(> t)
##	(Intercept)	2693.7	727.9	3.700	0.00178 **
##	net_count_per_month\$ppt_cm	-179.7	128.1	-1.403	0.17865

```
## ---
```

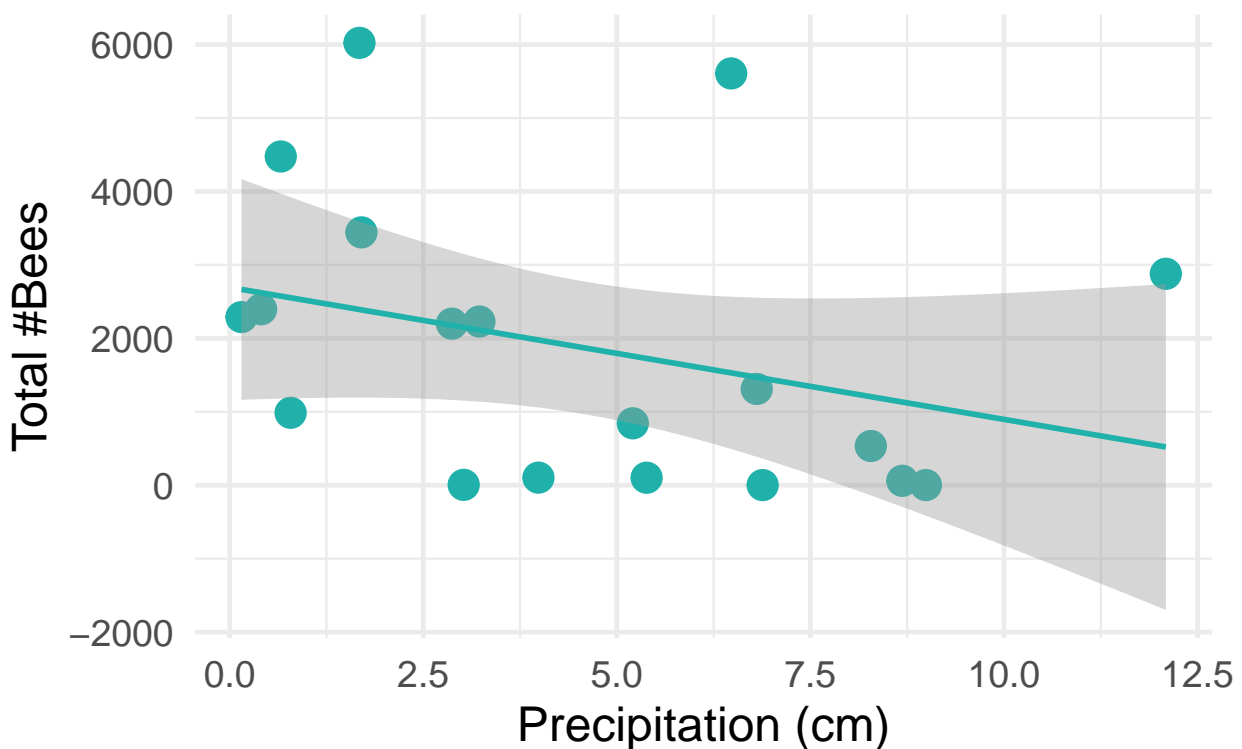
```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1867 on 17 degrees of freedom
## (8 observations deleted due to missingness)
## Multiple R-squared:  0.1038, Adjusted R-squared:  0.05104
## F-statistic: 1.968 on 1 and 17 DF,  p-value: 0.1786
```

```
p_title <- "Oregon - Net Count vs Mean Monthly Precipitation"
p <- ggplot(net_count_per_month, aes(x = ppt_cm, y = count)) +
  geom_point(size = 5, color = "lightseagreen") +
  labs(x = "Precipitation (cm)", y = "Total #Bees") +
  ggtitle(p_title) +
  theme_minimal(18) +
  geom_smooth(method = "lm", se = TRUE, color = "lightseagreen") +
  theme(plot.title = element_text(hjust = 0.5))
```

p

```
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 8 rows containing non-finite values (`stat_smooth()`).
## Warning: Removed 8 rows containing missing values (`geom_point()`).
```

Oregon – Net Count vs Mean Monthly Precipit



```
ggsave(glue("plots/{p_title}.png"), dpi = 300, plot = p)
```

```
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 8 rows containing non-finite values (`stat_smooth()`).
```

```

## Removed 8 rows containing missing values (`geom_point()`).
lin_reg <- lm(species_count_per_month$count ~ species_count_per_month$ppt_cm)
summary(lin_reg)

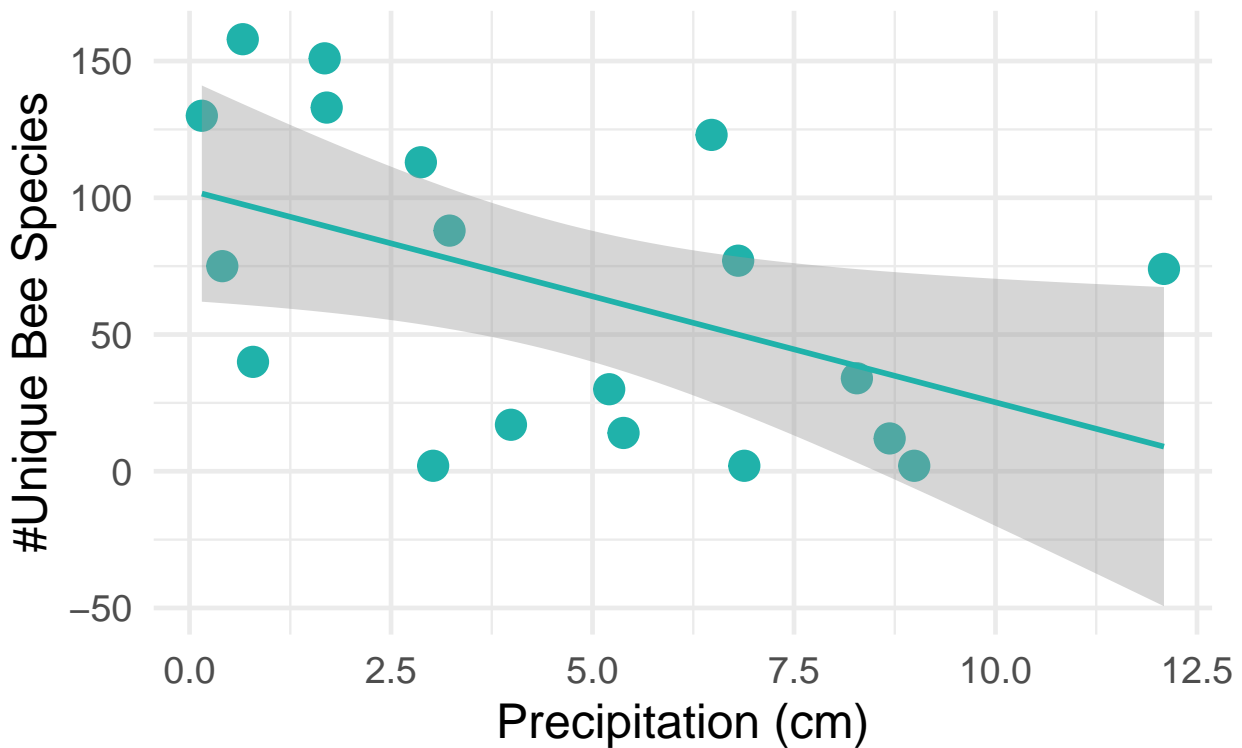
##
## Call:
## lm(formula = species_count_per_month$count ~ species_count_per_month$ppt_cm)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -77.292 -39.670  -4.536   37.999   70.485
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      102.721      19.151   5.364 5.16e-05 ***
## species_count_per_month$ppt_cm    -7.751       3.370  -2.300  0.0344 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 49.11 on 17 degrees of freedom
## (7 observations deleted due to missingness)
## Multiple R-squared:  0.2373, Adjusted R-squared:  0.1924
## F-statistic:  5.29 on 1 and 17 DF,  p-value: 0.03439

p_title <- "Oregon - Species Count vs Mean Monthly Precipitation"
p <- ggplot(species_count_per_month, aes(x = ppt_cm, y = count)) +
  geom_point(size = 5, color = "lightseagreen") +
  labs(x = "Precipitation (cm)", y = "#Unique Bee Species") +
  ggtitle(p_title) +
  theme_minimal(18) +
  geom_smooth(method = "lm", se = TRUE, color = "lightseagreen") +
  theme(plot.title = element_text(hjust = 0.5))
p

## `geom_smooth()` using formula = 'y ~ x'
## Warning: Removed 7 rows containing non-finite values (`stat_smooth()`).
## Warning: Removed 7 rows containing missing values (`geom_point()`).

```

Oregon – Species Count vs Mean Monthly Precip



```
ggsave(glue("plots/{p_title}.png"), dpi = 300, plot = p)
```

```
## Saving 6.5 x 4.5 in image
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 7 rows containing non-finite values (`stat_smooth()`).
```

```
## Removed 7 rows containing missing values (`geom_point()`).
```