HW 4

Brandon Hsu ph23497

You will submit this homework assignment as a pdf file on Gradescope.

For all questions, include the R commands/functions that you used to find your answer (show R chunk). Answers without supporting code will not receive credit. Write full sentences to describe your findings.

Part 1

Let's explore a dataset retrieved from the City of Austin data portal with the Austin 311 Public Data. The data was filtered to only include **complaints about coyotes** for the year of 2023:

```
# Upload the data from GitHub
coyotes <- read_csv("https://raw.githubusercontent.com/laylaguyot/datasets/main//coyotes_2023.csv")

# Take a quick look
head(coyotes)
```

```
## # A tibble: 6 x 22
##   ‘Service Request (SR) Number‘ ‘SR Description‘  ‘Method Received‘ ‘SR Status‘
##   <chr>                         <chr>              <chr>             <chr>
## 1 23-00001472                   Coyote Complaints Phone             Closed
## 2 23-00002238                   Coyote Complaints Phone             Closed
## 3 23-00002239                   Coyote Complaints Phone             Closed
## 4 23-00003607                   Coyote Complaints Phone             Closed
## 5 23-00003732                   Coyote Complaints Phone             Closed
## 6 23-00004999                   Coyote Complaints Phone             Closed
## # i 18 more variables: ‘Status Change Date‘ <chr>, ‘Created Date‘ <chr>,
## #   ‘Last Update Date‘ <chr>, ‘Close Date‘ <chr>, ‘SR Location‘ <chr>,
## #   ‘Street Number‘ <dbl>, ‘Street Name‘ <chr>, City <chr>, ‘Zip Code‘ <dbl>,
## #   County <chr>, ‘State Plane X Coordinate‘ <dbl>,
## #   ‘State Plane Y Coordinate‘ <dbl>, ‘Latitude Coordinate‘ <dbl>,
## #   ‘Longitude Coordinate‘ <dbl>, ‘(Latitude.Longitude)‘ <chr>,
## #   ‘Council District‘ <dbl>, ‘Map Page‘ <chr>, ‘Map Tile‘ <chr>
```

Question 1: (3 pts)

Look at the variables available in this dataset. How many variables are related to dates/times? How many variables are related to location?

4 variables are related to dates/ times. 12 variables are related to location.

To follow the workflow for data wrangling, you are going to wrangle the dataset below to create `coyotes_clean`:

- Keep only the necessary variables for answering the questions in this assignment.
- Ensure variable names are lowercase and use only one word.
- Convert date/time variables to proper R date/time formats.

Note: You will want to update this section as you work on this assignment to know which variables you will need to include and clean up!

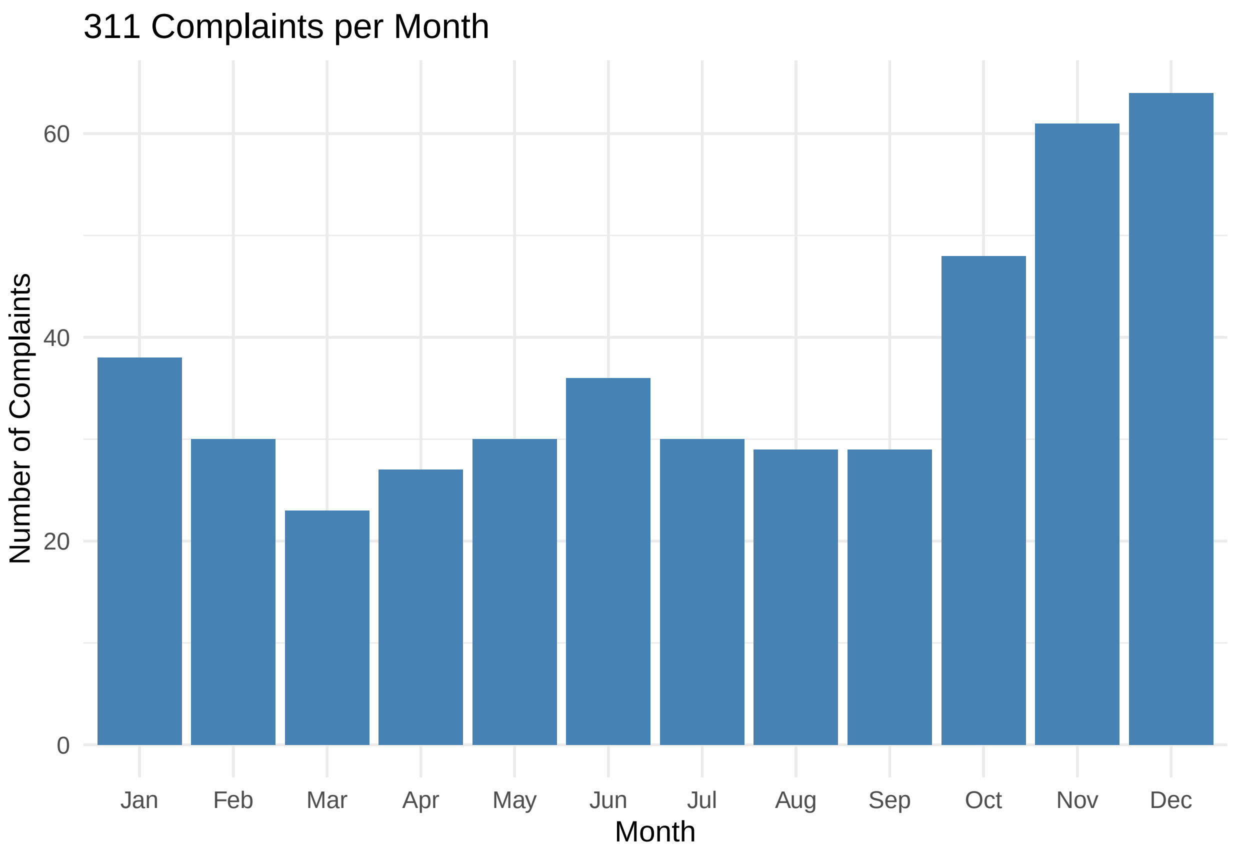
```
# creating coyotes_clean
coyotes_clean <- coyotes%>% select(-`Service Request (SR) Number`, -`SR Description`, -`Method Received

coyotes_clean <- setNames(coyotes_clean, c("change","created", "update", "close", "location", "streetnu
                                          "street", "city", "zip",
                                          "county", "latitude", "longitude","coordinate",
                                          "district"))
coyotes_clean$change <- mdy_hms(coyotes_clean$change)
coyotes_clean$created <- mdy_hms(coyotes_clean$created)
coyotes_clean$update <- mdy_hms(coyotes_clean$update)
coyotes_clean$close <- mdy_hms(coyotes_clean$close)
```

Don't modify `coyotes_clean` past this code chunk!

Question 2: (1 pt)

Create a visualization to display the number of complaints per month using abbreviated month names. During which time of the year are 311 reports most likely to complain about coyotes?

```
# visualization of complaints per month
month_graph <- coyotes_clean %>%
  mutate(month = month(created, label = TRUE, abbr = TRUE))


complaints_per_month <- month_graph %>%
  group_by(month) %>%
  summarise(count = n())


ggplot(complaints_per_month, aes(x = month, y = count)) +
  geom_col(fill = "steelblue") +
  labs(title = "311 Complaints per Month",
       x = "Month",
       y = "Number of Complaints") +
  theme_minimal()
```

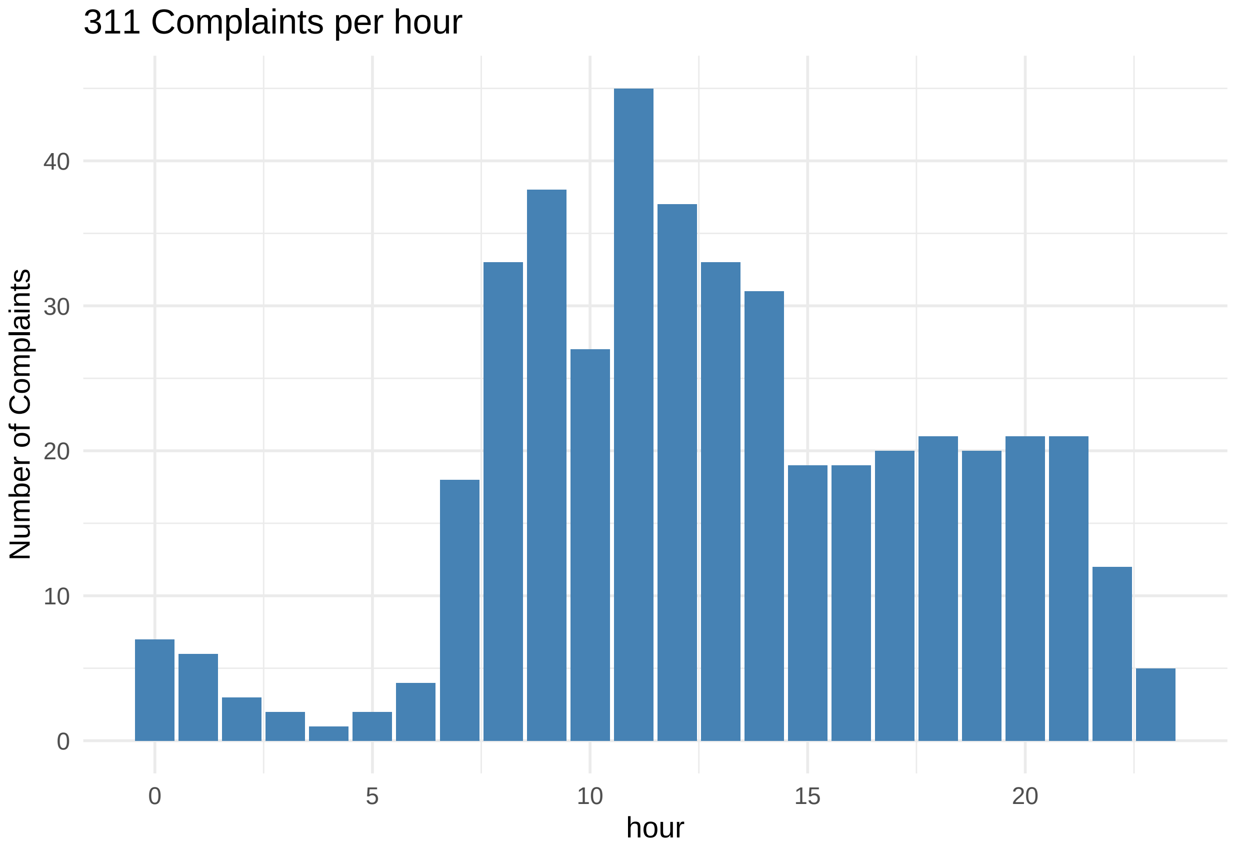


During winter there are the most number of complaints.

Question 3: (1 pt)

Create a visualization to display the number of complaints per time of the day. During which time of the day are 311 reports most likely to complain about coyotes? Does that make sense?

```
# visualization of number of complaints per time of the day
hour_graph <- coyotes_clean %>%
  mutate(hour = hour(created))

complaints_per_hour <- hour_graph %>%
  group_by(hour) %>%
  summarise(count = n())


ggplot(complaints_per_hour, aes(x = hour, y = count)) +
  geom_col(fill = "steelblue") +
  labs(title = "311 Complaints per hour",
       x = "hour",
       y = "Number of Complaints") +
  theme_minimal()
```



Around the 11th hour of the day there were the most number of complaints. This makes sense since most people are awake of at the time to file complaints.

Question 4: (1 pt)

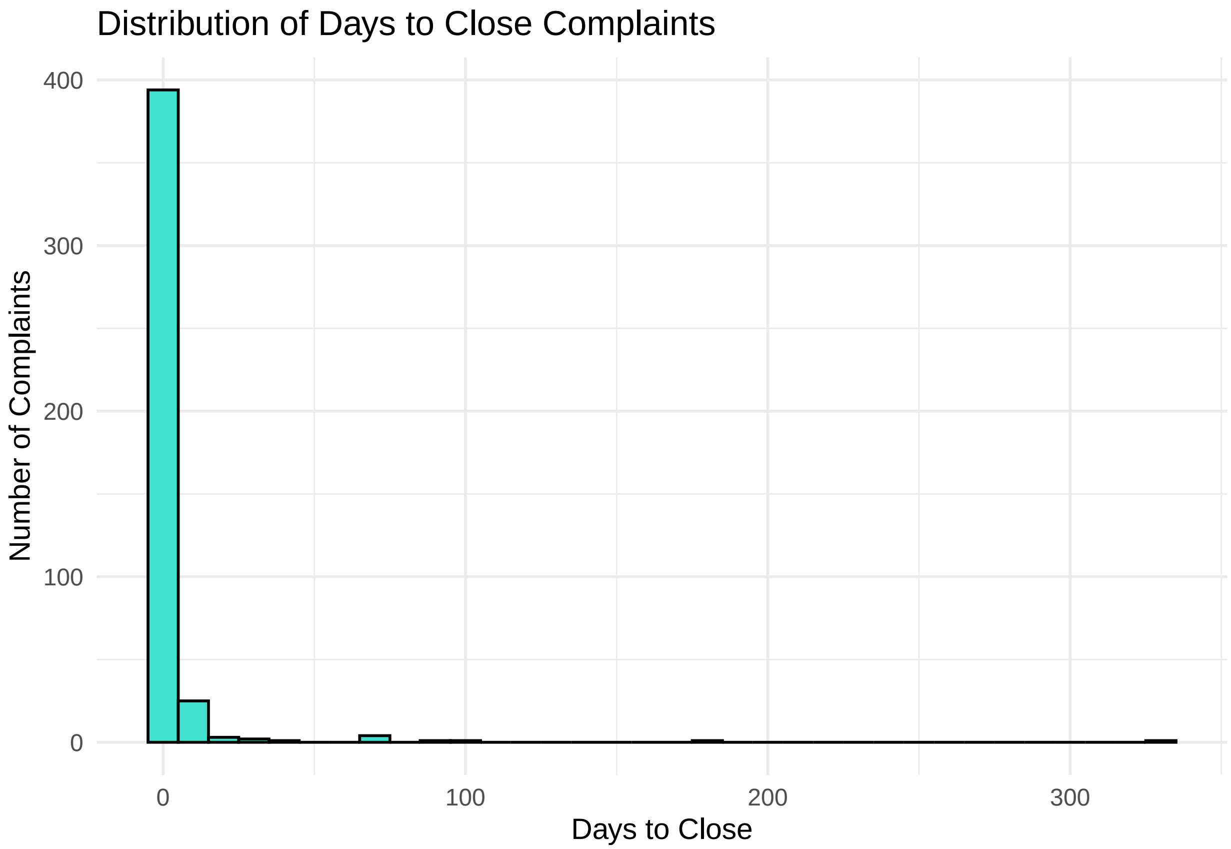
Create a visualization to display how many days it takes for a complaint to get closed after it was created. What is an average time for the complaint to be closed?

```
# visualization of how many days it takes for a complaint to get closed
close_time_graph <- coyotes_clean %>%
  mutate(close_time = as.numeric(difftime(close, created, units = "days")))

med <- median(close_time_graph$close_time, na.rm = TRUE)
print(med)
```

```
## [1] 1.669352
```

```
ggplot(close_time_graph, aes(x = close_time)) +
  geom_histogram(binwidth = 10, fill = "turquoise", color = "black") +
  labs(title = "Distribution of Days to Close Complaints",
       x = "Days to Close",
       y = "Number of Complaints") +
  theme_minimal()
```

```
## Warning: Removed 12 rows containing non-finite outside the scale range
## (‘stat_bin()‘).
```

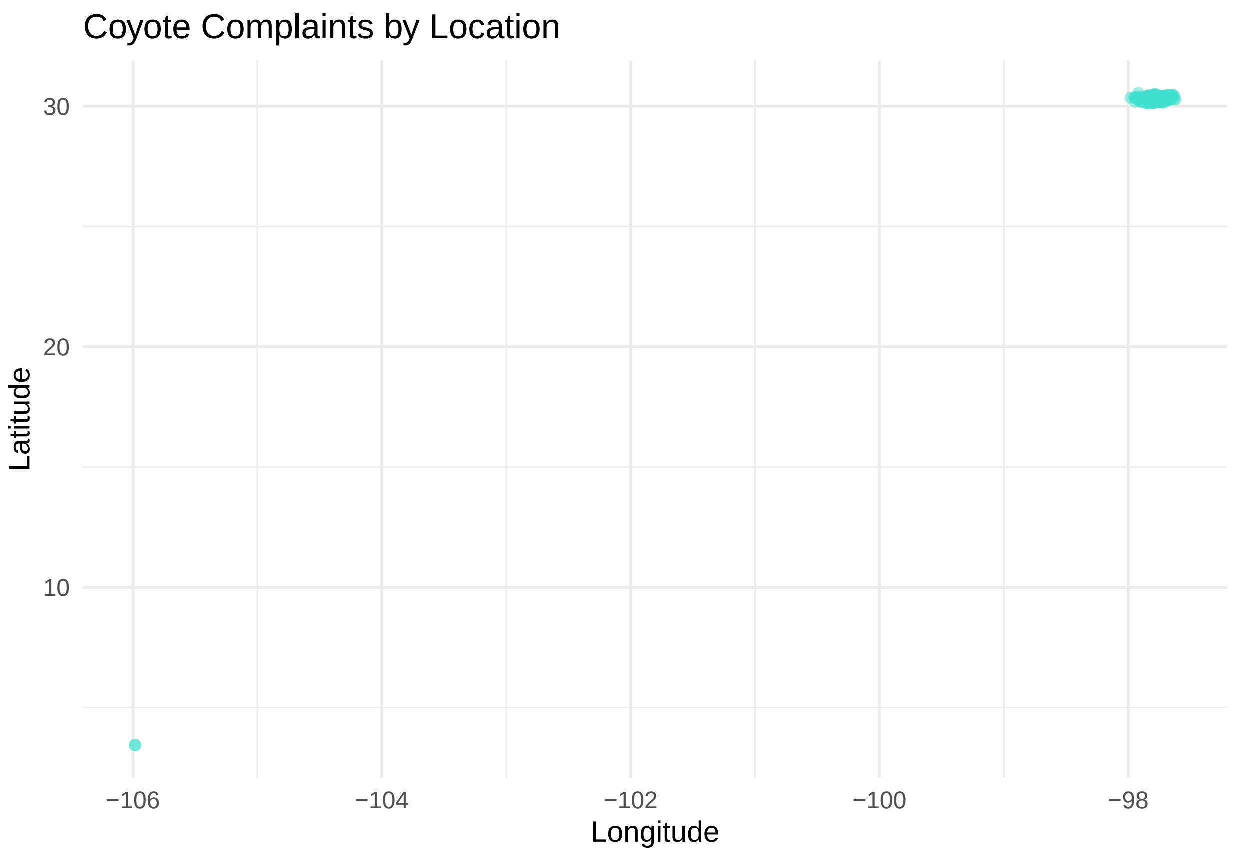


Half of the reports were closed within approxiamtely 1.67 days which is the median.

Question 5: (2 pts)

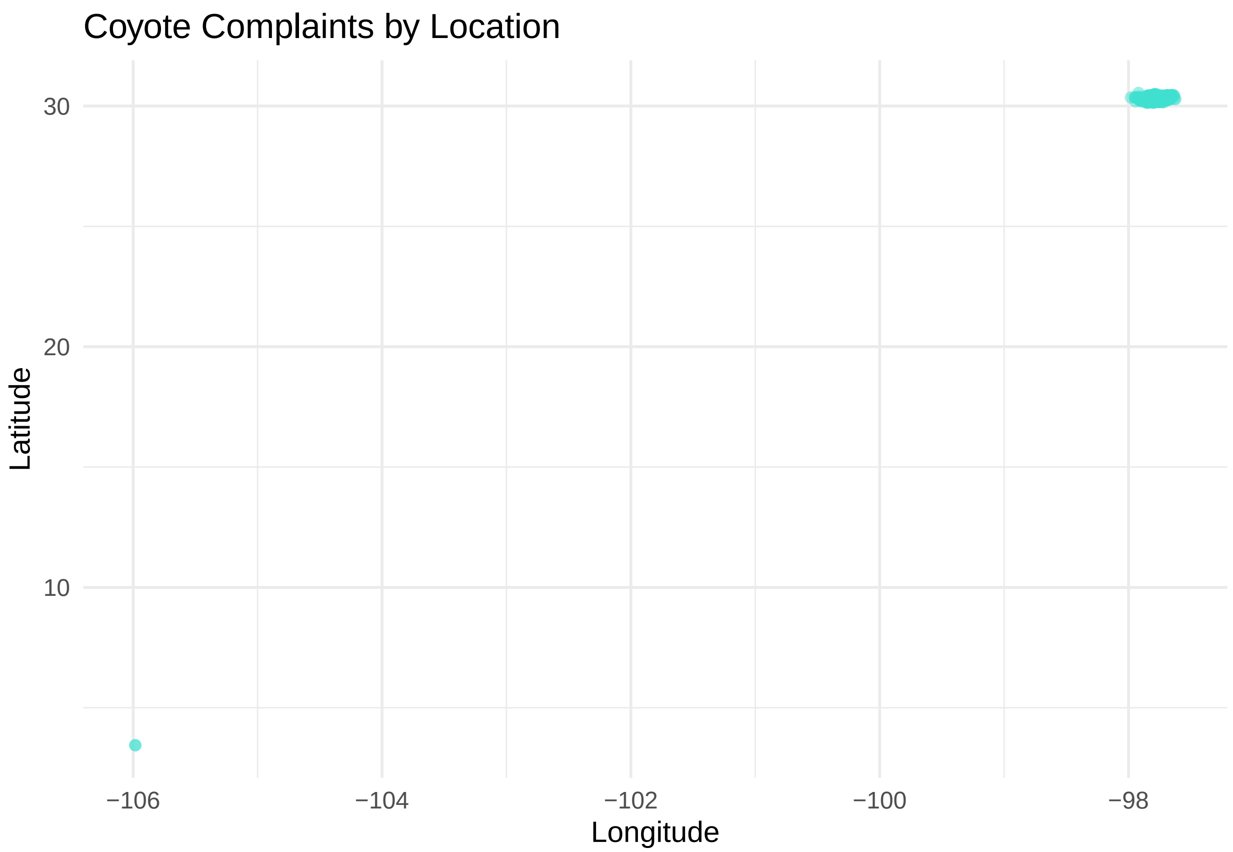
Create a visualization to display the location of the complaints given their latitude and longitude.

```
# location of the complaints given their latitude and logitude
ggplot(coyotes_clean, aes(x = longitude, y = latitude)) +
  geom_point(color = "turquoise", alpha = 0.5) +
  labs(title = "Coyote Complaints by Location",
       x = "Longitude",
       y = "Latitude") +
  theme_minimal()
```



You should notice some suspicious values for latitude/longitude. Investigate `coyotes` with all original data for invalid latitude and/or longitude values. What do you think about the location of these invalid values?

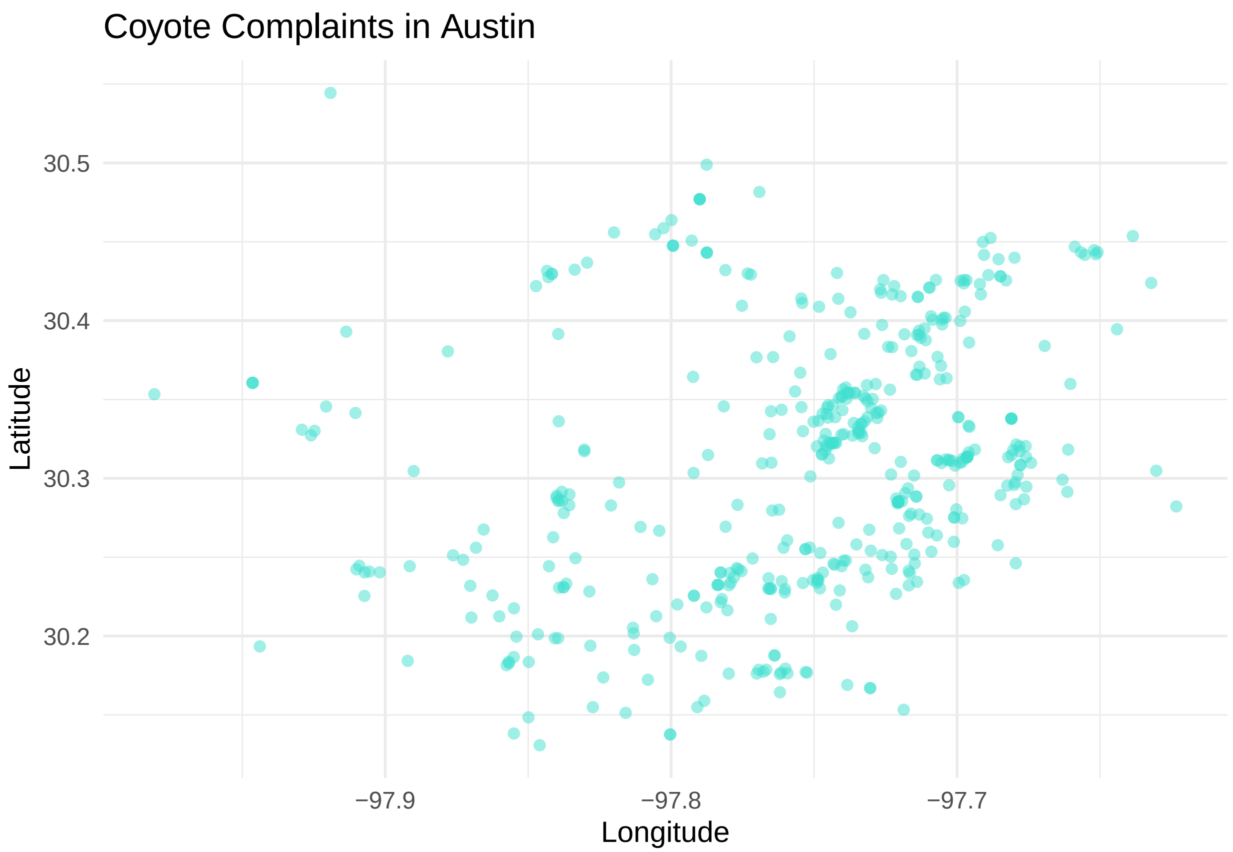
```r
# investigating coyotes dataset
ggplot(coyotes, aes(x = `Longitude Coordinate`, y = `Latitude Coordinate`)) +
  geom_point(color = "turquoise", alpha = 0.5) +
  labs(title = "Coyote Complaints by Location",
       x = "Longitude",
       y = "Latitude") +
  theme_minimal()
```



Based on the graph displayed there seems to be at least one outlier that has a longitude value around -106 and latitude value less than 5. This appears to be a misreported case as it is not within the Austin area. The value should be removed to make a more accurate graph.

Redo the visualization ignoring the invalid values for latitude and longitude and represent again the distribution of the reports in the Austin area. Are all 311 reports coming from the same location?

```
# redo of visualization
corrected_values <-coyotes_clean %>%
  filter(latitude >= 20 &  longitude >= -98.0)

ggplot(corrected_values, aes(x = longitude, y = latitude)) +
  geom_point(color = "turquoise", alpha = 0.5) +
  labs(title = "Coyote Complaints in Austin",
       x = "Longitude",
       y = "Latitude") +
  theme_minimal()
```



Almost all the 311 reports fall within Austin's longitude and latitude values. The longitude values range from -97.5 to -98. The latitutde values range from around 30 to 30.5.

Question 6: (3 pts)

Let's take a different perspective to analyze the location of the reports by ZIP code. First, you should note that there is one missing value for ZIP code in `coyotes`:

```
# find missing value of zip
summary(coyotes$`Zip Code`)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   78641   78723   78741   78734   78752   78759       1
```

```
view(coyotes)
missing_zip <- coyotes_clean %>%
  filter(is.na(zip))

print(missing_zip)
```

```
## # A tibble: 1 x 14
##   change              created             update
##   <dttm>              <dttm>              <dttm>
## 1 2023-12-13 12:36:32 2023-12-10 05:44:50 2023-12-13 12:36:32
## # i 11 more variables: close <dttm>, location <chr>, streetnum <dbl>,
## #   street <chr>, city <chr>, zip <dbl>, county <chr>, latitude <dbl>,
## #   longitude <dbl>, coordinate <chr>, district <dbl>
```

However, another variable contains the information about the actual ZIP code! Replace the missing value with the appropriate `Zip Code` in `coyotes_clean`. Now, summarize the number of 311 reports complaining about coyotes per ZIP code with `nb_reports` and save the resulting table as `coyotes_summary`:

```
# replace missing value of zip with appropriate zip cod, creating coyotes_summary
coyotes_clean <- coyotes_clean %>%
  mutate(zip = ifelse(is.na(zip), "78701", zip))

coyotes_summary <- coyotes_clean %>%
  group_by(zip) %>%
  summarise(nb_reports = n())

print(coyotes_summary)
```

```
## # A tibble: 41 x 2
##    zip   nb_reports
##    <chr>      <int>
##  1 78641          1
##  2 78652          2
##  3 78660          8
##  4 78701          2
##  5 78702         18
##  6 78703          6
##  7 78704         40
##  8 78717          6
##  9 78721          4
## 10 78722         17
## # i 31 more rows
```

What if we want to represent the ZIP codes on a map? We need what we call a shapefile: this type of file provides information about the borders of a location entity such as ZIP codes, districts, counties, states, countries, ... We can represent the borders with the geometry of a multipolygon. We will need a new package, `sf`, to manipulate shapefiles:

```
# Run `install.packages("sf")` in your console before loading this package
library(sf)
```

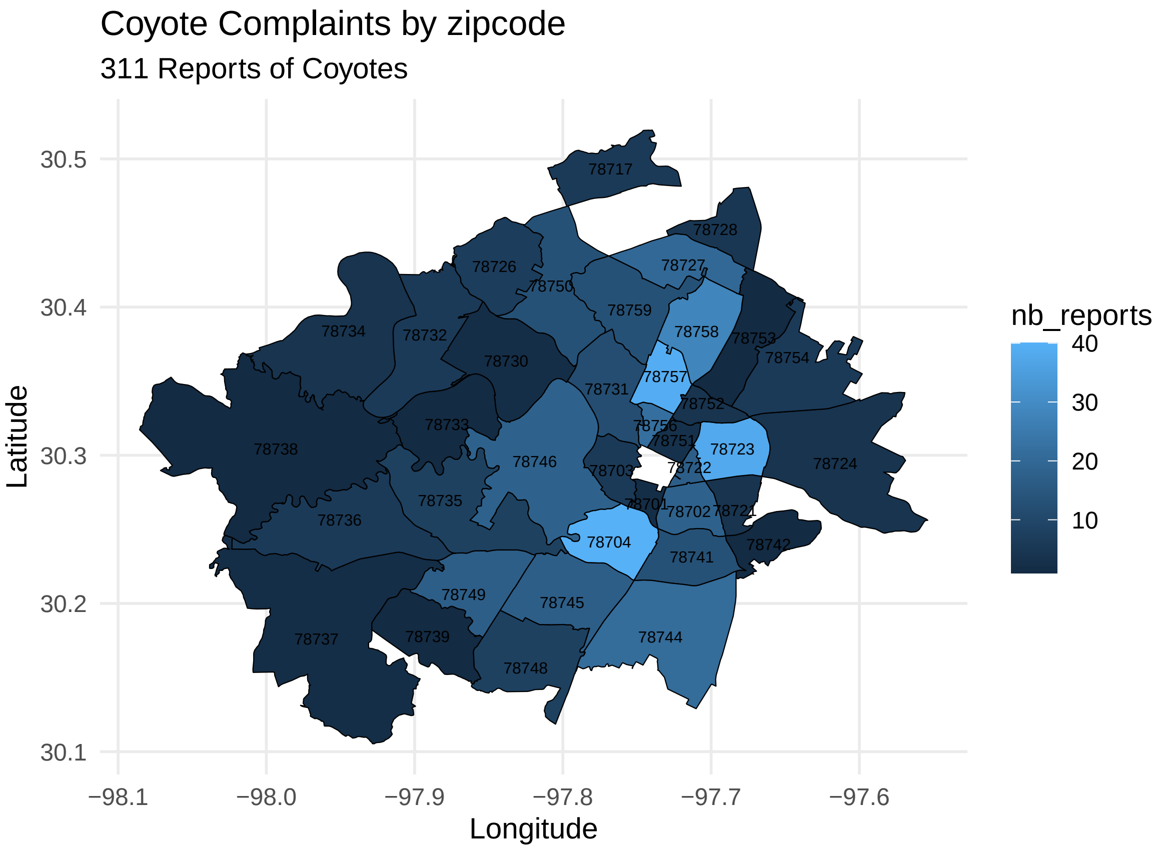
```
## Warning: package 'sf' was built under R version 4.3.3
```

We can upload the shapefile for ZIP codes from the data portal:

```
# Import the shapefile from the portal
zipcodes <- read_csv("https://data.austintexas.gov/resource/49ja-3mqz.csv") |>
  # Define the geometry in R format
  mutate(geometry = st_as_sfc(the_geom))
```

Join `coyotes_summary`, containing the number of 311 reports for coyote complaints per ZIP code, with the `zipcodes` shapefile. Then we make a map of ZIP codes in Austin! Comment each line of code to describe what it does. Improve the plot with labels and use colors that align with standard conventions. Which area seems to have more 311 reports for coyote complaints? Does that make sense?

```
zipcodes <- zipcodes %>%
  mutate(zipcode = as.character(zipcode))

zipcodes |>
  #Using the zipcodes dataset
  inner_join(coyotes_summary, by = c("zipcode" = "zip")) |>
  # This line of code joins coyotes_summary with the zipcodes shapefile
  ggplot() +
  # Make the plot using ggplot2
  geom_sf(aes(geometry = geometry, fill = nb_reports), color = "black") +
  # geom_sf plots spatial data using zipcodes, fill represents the number of complaints, while color ou
  geom_sf_text(aes(geometry = geometry, label = zipcode), size = 2, color = "black") +
  # this adds the zipcode labels while customizing text size and color
  labs(title = "Coyote Complaints by zipcode",
       subtitle = "311 Reports of Coyotes",
       x = "Longitude", y = "Latitude") +
  theme_minimal()
```



```
  #this adds the x and y axis labels and title
```

Areas 78704, 78757, and 78723 have more 311 reports. This makes sense since these areas have both urban and suburban areas where human-wildlife interactions occur at a higher frequency, resulting in a higher number of 311 reports.

Question 7: (2 pts)

We can do a similar analysis with the council districts! First, you should note that there are many missing values for the Council District in `coyotes`. Find which ZIP codes did not have a corresponding district. Look up some of these ZIP codes. Why does it make sense that the district is missing?

```
# anaylysis with council districts
missing_district <- coyotes %>%
  filter(is.na(`Council District`))
print(missing_district)
```

```
## # A tibble: 47 x 22
##    ‘Service Request (SR) Number‘ ‘SR Description‘  ‘Method Received‘ ‘SR Status‘
##    <chr>                         <chr>             <chr>             <chr>
##  1 23-00014813                   Coyote Complaints Phone             Closed
##  2 23-00019717                   Coyote Complaints Phone             Closed
##  3 23-00024961                   Coyote Complaints Phone             Closed
##  4 23-00025935                   Coyote Complaints Phone             Closed
##  5 23-00029656                   Coyote Complaints Phone             Closed
##  6 23-00106706                   Coyote Complaints Phone             Closed
##  7 23-00116300                   Coyote Complaints Phone             Closed
##  8 23-00118989                   Coyote Complaints Phone             Closed
##  9 23-00162171                   Coyote Complaints Phone             Closed
## 10 23-00167720                   Coyote Complaints Phone             Closed
## # i 37 more rows
## # i 18 more variables: ‘Status Change Date‘ <chr>, ‘Created Date‘ <chr>,
## #   ‘Last Update Date‘ <chr>, ‘Close Date‘ <chr>, ‘SR Location‘ <chr>,
## #   ‘Street Number‘ <dbl>, ‘Street Name‘ <chr>, City <chr>, ‘Zip Code‘ <dbl>,
## #   County <chr>, ‘State Plane X Coordinate‘ <dbl>,
## #   ‘State Plane Y Coordinate‘ <dbl>, ‘Latitude Coordinate‘ <dbl>,
## #   ‘Longitude Coordinate‘ <dbl>, ‘(Latitude.Longitude)‘ <chr>, ...
```

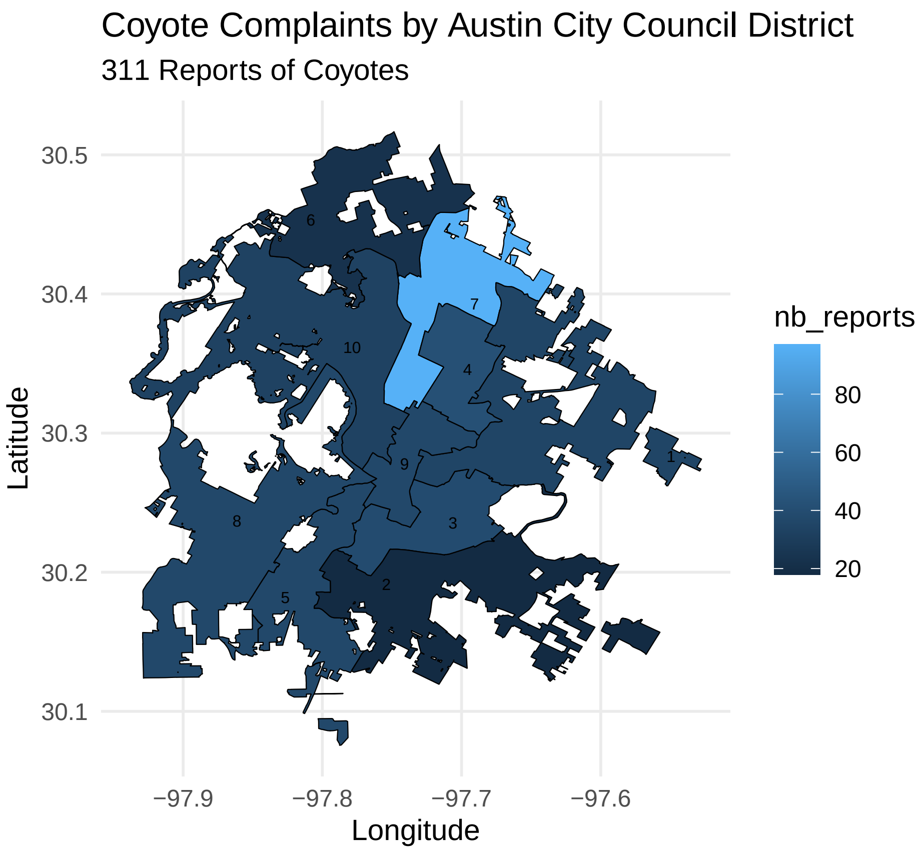
It makes sense that some of these zip codes that indicate Travis County have missing districts since some of these zip codes of Travis County are not entirely part of or outside of Austin.

We can upload the shapefile for districts from the data portal:

```
# Import the shapefile from the portal
districts <- read_csv("https://data.austintexas.gov/resource/w3v2-cj58.csv") |>
  # Define the geometry in R format
  mutate(geometry = st_as_sfc(the_geom))
```

Use the `districts` shapefile to make a map of the number of 311 reports complaining about coyotes per council district. Which area seems to have more 311 reports for coyote complaints? Does that make sense?

```
# making map of the number of 311 reports per council district
coyotes_district <- coyotes_clean %>%
  filter(!is.na(district)) %>%
  group_by(district) %>%
  summarise(nb_reports = n()) %>%
  mutate(district = as.character(district))


districts <- districts %>%
  mutate(district_number = as.character(district_number))

districts |>

  inner_join(coyotes_district, by = c("district_number" = "district")) |>

  ggplot() +

  geom_sf(aes(geometry = geometry, fill = nb_reports), color = "black") +

  geom_sf_text(aes(geometry = geometry, label = district_number), size = 2, color = "black") +
  labs(title = "Coyote Complaints by Austin City Council District",
       subtitle = "311 Reports of Coyotes",
       x = "Longitude", y = "Latitude") +
  theme_minimal()
```



District number 7 seems to have the most amount 311 reports. There are a lot of key green spaces such as walnut creek metropolitan park in that areah, providing ideal habitats for coyotes.

Question 8: (2 pts)

We can do a similar analysis with the counties! First, you should note that there is one missing value for county in `coyotes`:

```r
# find missing county
missing_county <- coyotes %>%
  filter(is.na(`County`))
print(missing_county)
```

```
## # A tibble: 1 x 22
##   ‘Service Request (SR) Number‘ ‘SR Description‘   ‘Method Received‘ ‘SR Status‘
##   <chr>                         <chr>              <chr>             <chr>
## 1 23-00542102                   Animal Protection~ Phone             Closed
## # i 18 more variables: ‘Status Change Date‘ <chr>, ‘Created Date‘ <chr>,
## #   ‘Last Update Date‘ <chr>, ‘Close Date‘ <chr>, ‘SR Location‘ <chr>,
## #   ‘Street Number‘ <dbl>, ‘Street Name‘ <chr>, City <chr>, ‘Zip Code‘ <dbl>,
## #   County <chr>, ‘State Plane X Coordinate‘ <dbl>,
## #   ‘State Plane Y Coordinate‘ <dbl>, ‘Latitude Coordinate‘ <dbl>,
## #   ‘Longitude Coordinate‘ <dbl>, ‘(Latitude.Longitude)‘ <chr>,
## #   ‘Council District‘ <dbl>, ‘Map Page‘ <chr>, ‘Map Tile‘ <chr>
```

However, another variable can indicate for which county that complaint was reported! Replace the missing value with the appropriate county name in `coyotes_clean`.

We can upload the shapefile for counties from the data portal:

```r
# Import the shapefile from the portal
counties <- read_csv("https://data.austintexas.gov/resource/tnsq-nquk.csv") |>
  # Define the geometry in R format
  mutate(geometry = st_as_sfc(the_geom))

head(counties)
```

```
## # A tibble: 6 x 11
##   the_geom     statefp countyfp countyns affgeoid geoid name   lsad  aland awater
##   <chr>          <dbl>    <dbl>    <dbl> <chr>    <dbl> <chr> <dbl>  <dbl>  <dbl>
## 1 MULTIPOLYG~       48      421  1383996 0500000~ 48421 Sher~     6 2.39e9 4.29e5
## 2 MULTIPOLYG~       48      493  1384032 0500000~ 48493 Wils~     6 2.08e9 1.21e7
## 3 MULTIPOLYG~       48      115  1383843 0500000~ 48115 Daws~     6 2.33e9 4.72e6
## 4 MULTIPOLYG~       48       69  1383820 0500000~ 48069 Cast~     6 2.32e9 1.26e7
## 5 MULTIPOLYG~       48      279  1383926 0500000~ 48279 Lamb      6 2.63e9 3.97e6
## 6 MULTIPOLYG~       48      385  1383978 0500000~ 48385 Real      6 1.81e9 2.35e6
## # i 1 more variable: geometry <MULTIPOLYGON>
```

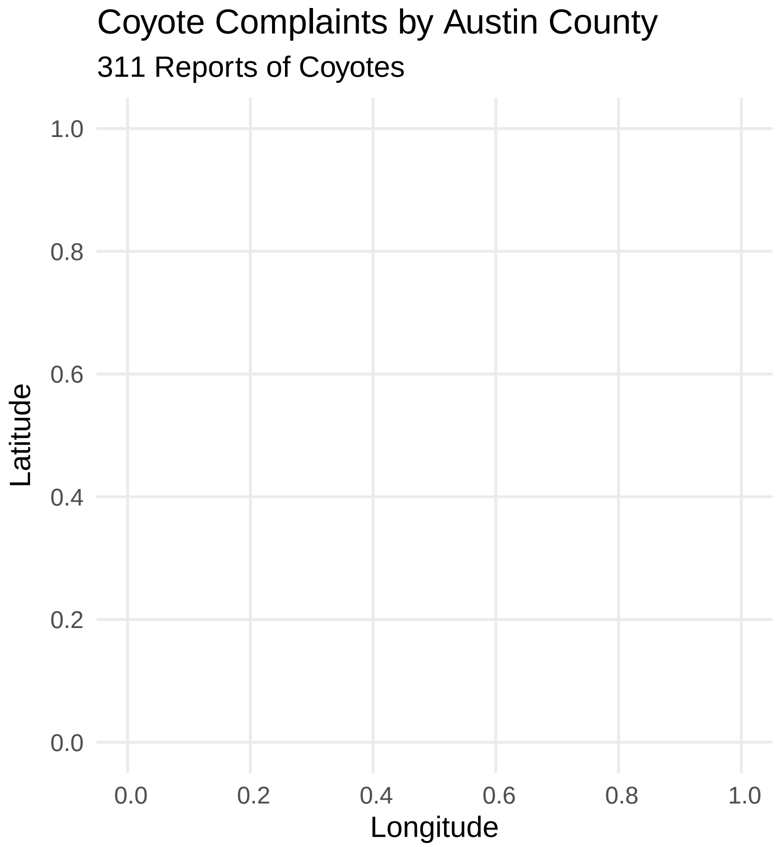
```
view(counties)

coyotes_clean <- coyotes_clean %>%
  mutate(county = ifelse(is.na(county), "TRAVIS", county))
```

Use the `counties` shapefile to make a map of the number of 311 reports complaining about coyotes per county. Which area seems to have more 311 reports for coyote complaints? Does that make sense?

```
# making map of the 311 reports about coyotes per county

coyotes_county <- coyotes_clean %>%
  group_by(county) %>%
  summarise(nb_reports = n())

print(coyotes_county)
```

```
## # A tibble: 2 x 2
##   county     nb_reports
##   <chr>           <int>
## 1 TRAVIS            428
## 2 WILLIAMSON         17
```

```
counties |>
  inner_join(coyotes_county, by = c("name" = "county")) |>
  ggplot() +
  geom_sf(aes(geometry = geometry, fill = nb_reports), color = "black") +
  geom_sf_text(aes(geometry = geometry, label = name), size = 2, color = "black") +
  labs(title = "Coyote Complaints by Austin County",
       subtitle = "311 Reports of Coyotes",
       x = "Longitude", y = "Latitude") +
  theme_minimal()
```



There are no matching counties from the shapefile county and coyotes_clean. Therefore, a blank map was produced.

Part 2

In this part, we are focusing on ethical considerations. As mentioned above, the City of Austin collects and publishes 311 service request data, including complaint details, locations, and timestamps. While this data is useful for analysis, it can also raise ethical concerns.

Question 9: (2 pts)

What are some potential risks of publishing location-based complaint data?

Publishing location-based complaint data can pose privacy risks by potentially revealing the identity of individuals who report complaints, especially in low-density areas. Additionally, it may lead to unintended consequences such as property devaluation or social stigma for neighborhoods with high complaint rates, as they may be perceived as problematic areas. Lastly, the data could be misused for discriminatory practices, influencing real estate decisions or resource allocation in a way that disproportionately affects certain communities.

Question 10: (2 pts)

Suppose the City of Austin uses 311 complaint data about coyotes to allocate resources for wildlife control, responding more quickly to areas with more reports. Suggest a way to reduce bias and ensure fair resources across all neighborhoods.

YTo reduce bias and ensure fair resource allocation, the City of Austin could normalize complaint data by population density and land use to avoid disproportionately prioritizing areas with higher reporting rates simply due to higher population levels. Additionally, proactive monitoring strategies, such as wildlife surveys and community outreach programs, can help identify coyote activity in areas with historically lower reporting rates to ensure an equitable distribution of resources..

Formatting: (1 pt)

Throughout the semester, the following components are expected for formatting all assignments:

- **code comments**: the code in the assignment is commented. Comments should explain the purpose of the code and help clarify different steps (see class worksheets for example of meaningful comments).
- **knitted file**: the R file is knitted so we can see the code and the outputs. You can knit into pdf directly or into html. Once it knits in html, click on `Open in Browser` at the top left of the window pops out. Print your html file into pdf from your browser.
- **acknowledgements**: any external resources used to complete the assignment are cited. If AI was used, the prompts are shared and commented.
- **page selection**: each page of the assignment is selected and matched to the corresponding questions on Gradescope.

Any issue? Ask your instructional team!