Final Project Markdown

Carson Cherniss & Ainsley Gallagher

Data and Data Cleaning

Shiny App: https://carsonic.shinyapps.io/final_project_shiny_app/

GitHub: https://github.com/carsonic/Cherniss_Gallagher_Final_Project, https://github.com/AinsGalla/FinalProjectDS/tree/main

Dataset: https://wildlife.faa.gov/home

Research Question:

During what time of day are wildlife collisions most common?

Cleaning in Excel

Dataset over 130MB, too large to upload to GitHub, very slow loading into R. Removed 85 variables to reduce file size: `AIRPORT`, `AIRPORT_LATITUDE`, `AIRPORT_LONGITUDE`, `RUNWAY`, `FAAREGION`, `LOCATION`, `OPID`, `OPERATOR`, `REG`, `FLT`, `AMA`, `AMO`, `EMA`, `EMO`, `AC_CLASS`, `AC_MASS`, `TYPE_ENG`, `NUM_ENGS`, `ENG_1_POS`, `ENG_2_POS`, `ENG_3_POS`, `ENG_4_POS`, `PHASE_OF_FLIGHT`, `HEIGHT`, `SPEED`, `DISTANCE`, `AOS`, `COST_REPAIRS`, `COST_OTHER`, `COST_REPAIRS_IFL_ADJ`, `COST_OTHER_IFL_ADJ`, `INGESTED_OTHER`, `INDICATED_DAMAGE`, `DAMAGE_LEVEL`, `STR_RAD`, `DAM_RAD`, `STR_WINDSHLD`, `DAM_WINDSHLD`, `STR_NOSE`, `DAM_NOSE`, `STR_ENG1`, `DAM_ENG1`, `ING_Eng1`, `STR_ENG2`, `DAM_ENG2`, `ING_Eng2`, `STR_ENG3`, `DAM_ENG3`, `ING_Eng3`, `STR_ENG4`, `DAM_ENG4`, `ING_Eng4`, `STR_PROP`, `DAM_PROP`, `STR_WING_ROT`, `DAM_WING_ROT`, `STR_FUSE`, `DAM_FUSE`, `STR_LG`, `DAM_LG`, `STR_TAIL`, `DAM_TAIL`, `STR_LIGHTS`, `DAM_LIGHTS`, `STR_OTHER`, `DAM_OTHER`, `OTHER_SPECIFY`, `EFFECT`, `EFFECT_OTHER`, `BIRD_BAND_NUMBER`, `OUT_OF_RANGE_SPECIES`, `REMARKS`, `REMAINS_COLLECTED`, `REMAINS_SENT`, `WARNED`, `NUM_SEEN`, `ENROUTE_STATE`, `NR_INJURIES`, `NR_FATALATIES`, `COMMENTS`, `REPORTED_NAME`, `REPORTED_TITLE`, `SOURCE`, `PERSON`, `TRANSFER`.

Cleaning in R

```
# Load Packages
pacman::p_load(tidyverse, readxl, lubridate, janitor)
# Read the data
faa_data <- read_excel("Public.xlsx")
glimpse(faa_data)

# Parse INCIDENT_DATE and LUPDATE and TIME as dates
faa_data <- faa_data |>
  mutate(
    INCIDENT_DATE = as_date(INCIDENT_DATE),
    LUPDATE = as_date(LUPDATE),
    TIME = lubridate::hm(TIME)
  )

# Clean names using janitor package
faa_data <- faa_data |>
  janitor::clean_names()
```

I also downloaded the data as a csv file. The file was larger so I did not use it for most of the exploratory data analysis. However, I foudn out I needed it when I was trying to publish the shiny app and it would not work with the excel file. When using csv data, parsing the dates is not neccessary, they already show up as dates when you download the data. Attempting to parse time in particular causes the second graph to not render. Be careful using the csv version vs. the excel version.

Exploratory Data Analysis

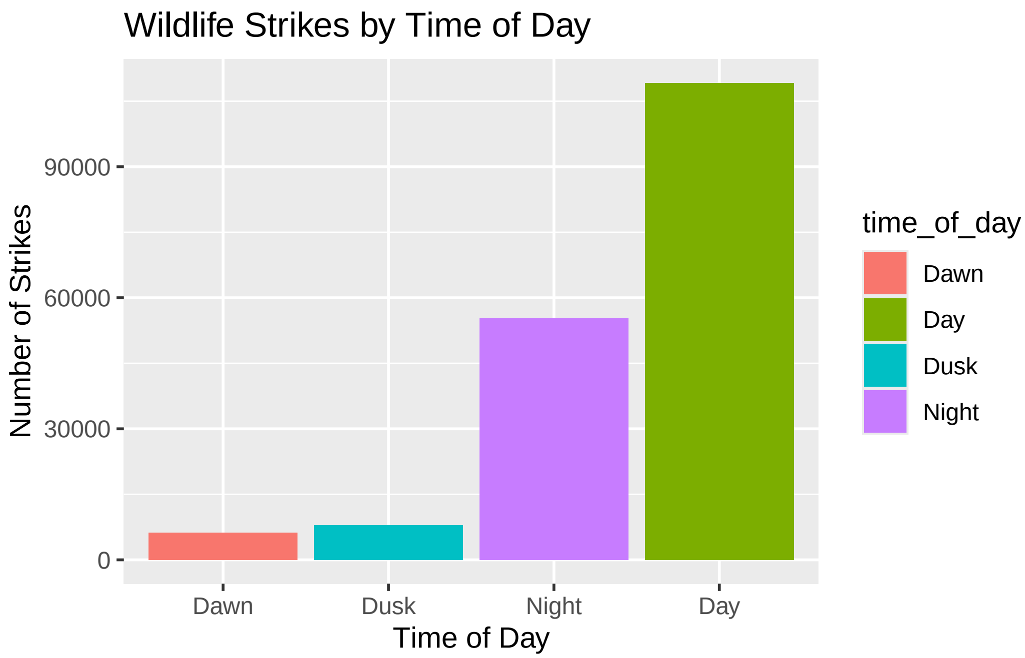
We created data visualizations and conducted a statistical test to investigate our hypothesis.

Data Visualization

First graph is a bar chart showing how the count of wildlife strikes is distributed by the `time_of_day` variable.

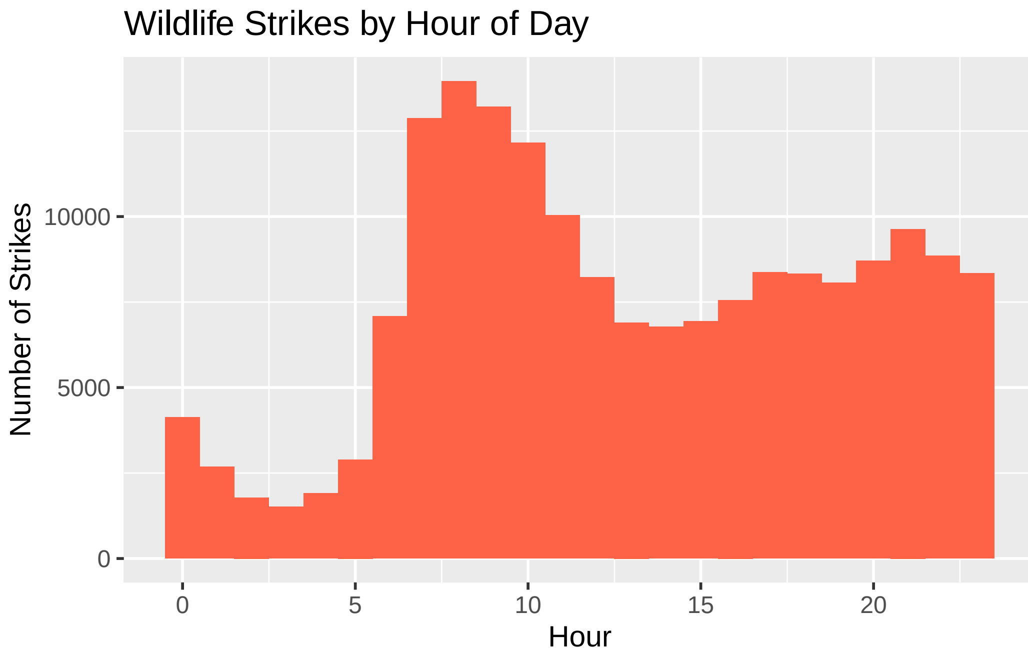
```
# Graph 1: Bar chart showing time_of_day variable distribution
faa_data |>
  filter(!is.na(time_of_day)) |>
  count(time_of_day) |>
```

```r
  ggplot(aes(x = fct_reorder(time_of_day, n), y = n, fill = time_of_day)) +
  geom_col() +
  labs(
    title = "Wildlife Strikes by Time of Day",
    x = "Time of Day",
    y = "Number of Strikes"
  )
```



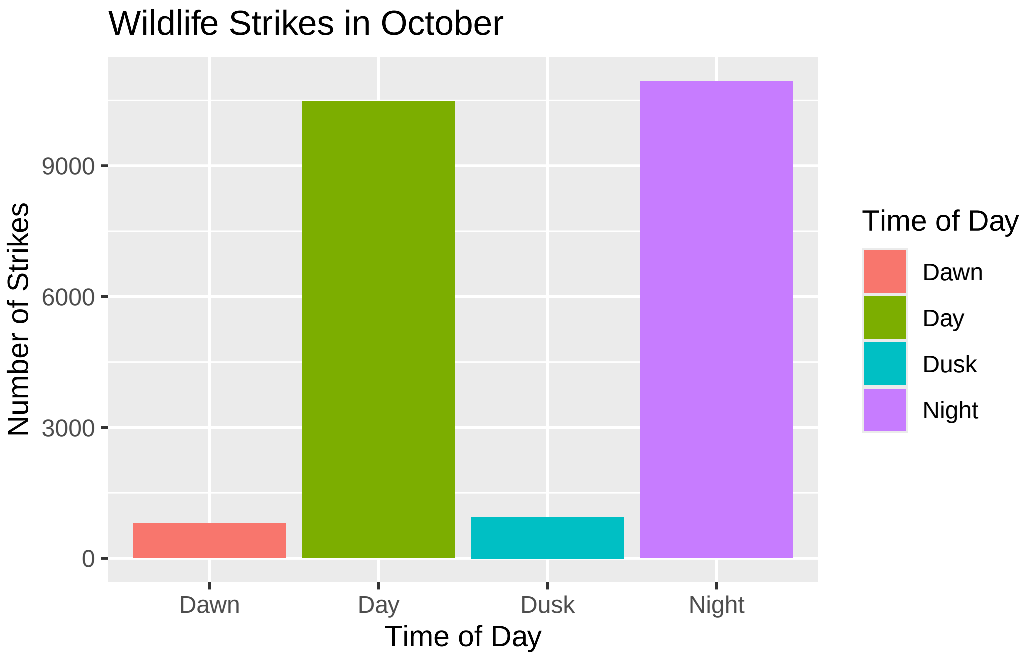
Second graph shows the count but instead uses the `time` variable. This shows more accurately the distribution of strikes throughout the day, since we do not have a codebook for this data and do not know what hours mean `day` or `night` or `dawn` or `dusk`.

```r
# Graph 2: Bar chart showing time variable distribution
faa_data |>
  filter(!is.na(time)) |>
  mutate(hour = hour(time)) |>
  ggplot(aes(x = hour)) +
  geom_histogram(binwidth = 1, fill = "tomato") +
  labs(title = "Wildlife Strikes by Hour of Day",
       x = "Hour",
       y = "Number of Strikes")
```

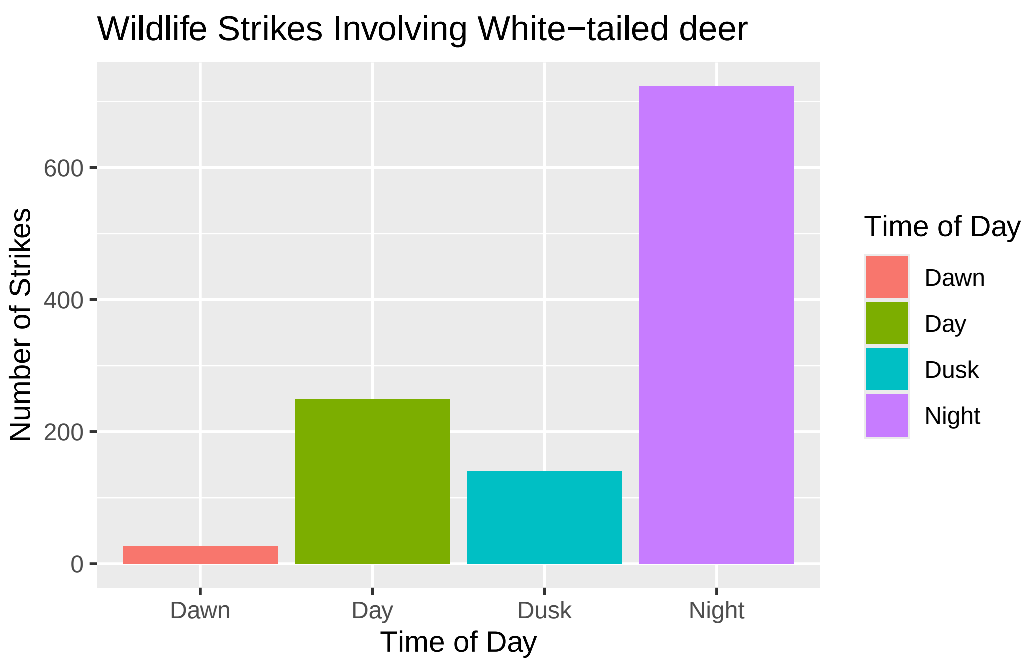


The additional 2 graphs will allow us to add an interactive element to the shiny app. The code will change slightly when used in the app, but it will allow us to see the `time_of_day` variable bar graph but broken down by month.

```
# Graph 3: Monthly plot
selected_month <- "October"

monthly_plot <- faa_data |>
  filter(!is.na(time_of_day)) |>
  mutate(month = lubridate::month(incident_date, label = TRUE, abbr = FALSE)) |>
  filter(month == selected_month) |>
  count(time_of_day) |>
  ggplot(aes(x = time_of_day, y = n, fill = time_of_day)) +
  geom_col() +
  labs(
    title = paste("Wildlife Strikes in", selected_month),
    x = "Time of Day",
    y = "Number of Strikes",
    fill = "Time of Day"
  )

monthly_plot
```




```r
# Graph 4:
selected_species <- "White-tailed deer"

species_plot <- faa_data |>
  filter(!is.na(time_of_day), species == selected_species) |>
  count(time_of_day) |>
  ggplot(aes(x = time_of_day, y = n, fill = time_of_day)) +
  geom_col() +
  labs(
    title = paste("Wildlife Strikes Involving", selected_species),
    x = "Time of Day",
    y = "Number of Strikes",
    fill = "Time of Day"
  )

species_plot
```



Statistical Test

To formally test whether wildlife strikes are evenly distributed across times of day, we conducted a Chi-square goodness-of-fit test.

Statistical Hypthesis:

Null hypothesis (H_0):
Wildlife strikes are evenly distributed across time-of-day categories. **Alternative hypothesis (H_1):**
Wildlife strikes are not evenly distributed across time-of-day categories.

Observed Counts:

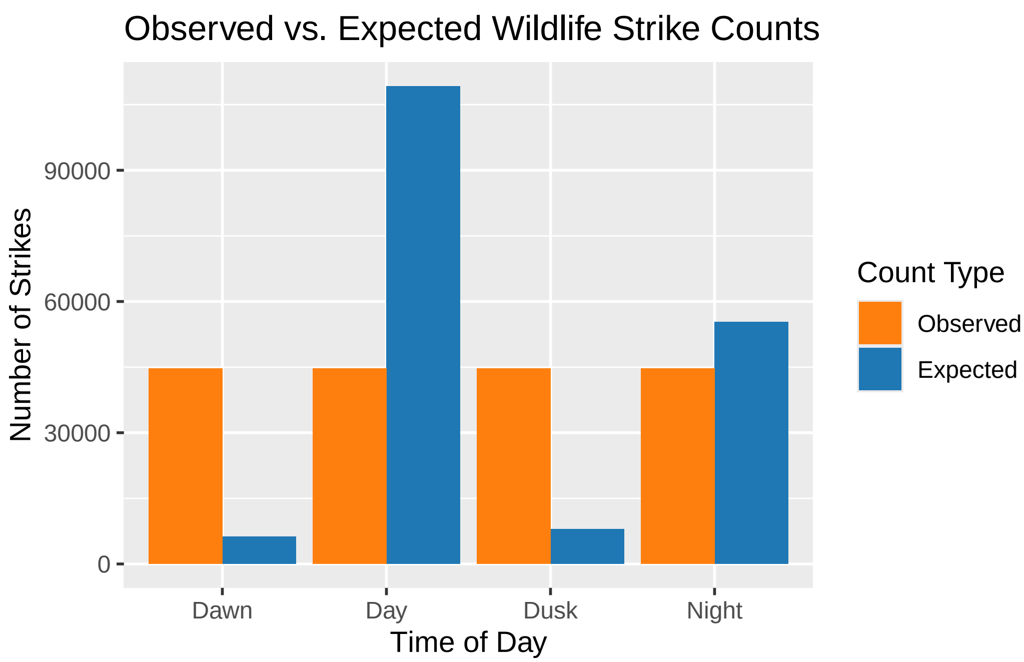
```r
strike_counts <- faa_data |>
  filter(!is.na(time_of_day)) |>
  count(time_of_day)

print(strike_counts)
```

```
# A tibble: 4 x 2
  time_of_day     n
  <chr>       <int>
1 Dawn         6203
```

```
2 Day         109228
3 Dusk          7913
4 Night        55275
```

Visualization (We can compare the actual and expected counts visually before we do the statistical test):

```
chi_data <- faa_data |>
  filter(!is.na(time_of_day)) |>
  count(time_of_day) |>
  arrange(time_of_day)

chi_test <- chisq.test(chi_data$n)

chi_data <- chi_data |>
  mutate(expected = chi_test$expected)

plot_data <- chi_data |>
  pivot_longer(cols = c(n, expected), names_to = "type", values_to = "count")

# Plot observed vs expected
ggplot(plot_data, aes(x = time_of_day, y = count, fill = type)) +
  geom_col(position = "dodge") +
  scale_fill_manual(
    values = c("n" = "#1f77b4", "expected" = "#ff7f0e"),
    labels = c("Observed", "Expected")
  ) +
  labs(
    title = "Observed vs. Expected Wildlife Strike Counts",
    x = "Time of Day",
    y = "Number of Strikes",
    fill = "Count Type"
  )
```



Chi-square test:

```
chisq_test <- chisq.test(strike_counts$n)
chisq_test
```

```
    Chi-squared test for given probabilities

data:  strike_counts$n
X-squared = 159244, df = 3, p-value < 2.2e-16
```