Cornell Gorges

A Data-Driven Look at the Ithacan Gorges

Ming DeMers

September 10, 2023

Contents

Introduction

The gorges at Cornell University are one of the university's most iconic and scenic features. Nestled in the heart of the campus, Cornell's gorges are a series of deep and narrow ravines carved by the flow of water over thousands of years. These beautiful natural formations offer a tranquil escape from the hustle and bustle of campus life, providing students and visitors with picturesque spots for relaxation and exploration. While the gorges are captivating, they can be deceptively dangerous. Safety measures are enforced and visitors are urged to follow the guidelines.

The Gorge Stewards Program was established in 2014 to improve safety and enjoyment of the gorges, following the death of a student earlier that year. Gorge stewards walk the paths from May to September to provide information about trails, safety rules, natural history, activities, and swimming alternatives. Stewards also track visitor use, including unauthorized or illegal uses.

This project will explore how usage of the Gorges has fluctuated, how violations of gorge rules has decreased, and explain overall the changing landscape of the Gorges.

The Data

The data is sourced from the Gorge Stewards Program, courtesy of the Cornell Botanic Gardens and Cornell Outdoor Education (COE). It comprises of a table for each year recorded, with a number of variables detailing gorge usage, violations, weather and more.

Violations are when visitors do not abide by posted rules of the Gorge. These include swimming, diving, or being in certain parts of the stream, trespassing, illegal substance possessions, and more. The majority of violations involve individuals trying to swim - a deceptively dangerous activity, often following to the death of a person every few years. Since the establishment of the program, no deaths or rescues have been reported; and violations have steadily decreased as more people have visited the beautiful gorges.

Data Cleaning

```
clean_gorge <- function(df) {
  df <- df |>
  mutate(Date = as.Date(Date, format('%m/%d/%Y'))) |>
  drop_na(Date) |>
  mutate(High.Temperature = as.numeric(High.Temperature)) |>
  clean_names()

  colnames(df) <- gsub("number\\_of\\_", "", colnames(df))

  return(df)
}

# lapply(gorges, function)

gorge_2022_clean <- gorge_2022_raw |>
  clean_gorge()
```

```
## mutate: converted 'Date' from character to Date (1 new NA)

## drop_na: removed one row (1%), 108 rows remaining

## Warning: There was 1 warning in `.fun()`.
## i In argument: `High.Temperature = as.numeric(High.Temperature)`.
## Caused by warning:
## ! NAs introduced by coercion

## mutate: converted 'High.Temperature' from character to double (2 new NA)
```

```
gorge_2022_clean <- gorge_2022_clean |>
  mutate(date = replace(date, row_number() == 23, '2022-06-09')) |>
  mutate(steward_name = if_else(steward_name == "Philip",
                                "Phillip",
                                steward_name)) |>
  mutate(steward_name = str_replace_all(
    str_to_title(steward_name),
    " ",
    ""))
```

```
## mutate: changed one value (1%) of 'date' (0 new NA)

## mutate: changed one value (1%) of 'steward_name' (0 new NA)

## mutate: changed 9 values (8%) of 'steward_name' (0 new NA)
```

```
gorge_2022 <- gorge_2022_clean
```

We first import the sets and clean up the data best we can.

Gorge Statistics

We are interested in understanding the Gorges by the numbers. `Gorges` is a dataset that has all the years joined into one dataset. Due to format of the data changing, some granular detail is lost.

Join the Gorges

```
gorges_raw <- read.csv('data/gorges.csv')

gorges_clean <- gorges_raw |>
  drop_na(Date) |>
  drop_na(Number.of.observed.gorge.users) |>
  drop_na(total.Number.of.people.obseved.violating.a.rule) |>
  mutate(Date = as.Date(Date, format('%m/%d/%Y')),
         observed_gorge_users = as.numeric(Number.of.observed.gorge.users),
         observed_violations = as.numeric(
           total.Number.of.people.obseved.violating.a.rule),
         High.Temperature = as.numeric(High.Temperature)) |>
  clean_names()
```

```
## drop_na: no rows removed
## drop_na: no rows removed
## drop_na: no rows removed

## Warning: There were 3 warnings in `.fun()`.
## The first warning was:
## i In argument: `observed_gorge_users =
##   as.numeric(Number.of.observed.gorge.users)`.
## Caused by warning:
## ! NAs introduced by coercion
## i Run `dplyr::last_dplyr_warnings()` to see the 2 remaining warnings.

## mutate: converted 'Date' from character to Date (131 new NA)

##         converted 'High.Temperature' from character to double (84 new NA)

##         new variable 'observed_gorge_users' (double) with 327 unique values and 15% NA

##         new variable 'observed_violations' (double) with 41 unique values and 16% NA
```

```
gorges <- gorges_clean |>
  mutate(observed_gorge_users = ifelse(is.na(observed_gorge_users),
                                       0,
                                       observed_gorge_users),
         observed_violations = ifelse(is.na(observed_violations),
                                      0,
                                      observed_violations),
         high_temperature = ifelse(is.na(high_temperature),
                                   0,
                                   high_temperature)
         ) |>
  filter(year(date) > (2014))
```

```
## mutate: changed 84 values (8%) of 'high_temperature' (84 fewer NA)
##         changed 163 values (15%) of 'observed_gorge_users' (163 fewer NA)
##         changed 173 values (16%) of 'observed_violations' (173 fewer NA)
## filter: removed 132 rows (12%), 975 rows remaining
```

Yearly Stats

```
gorges_tbl <- gorges |>
  group_by(year(date)) |>
  summarise(shifts_completed = n(),
            visitors = sum(observed_gorge_users),
            violations = sum(observed_violations),
            # stewards = length(unique(steward_name)),
            avg_temp = mean(high_temperature[high_temperature != 0])) |>
  mutate(year = `year(date)`)
```

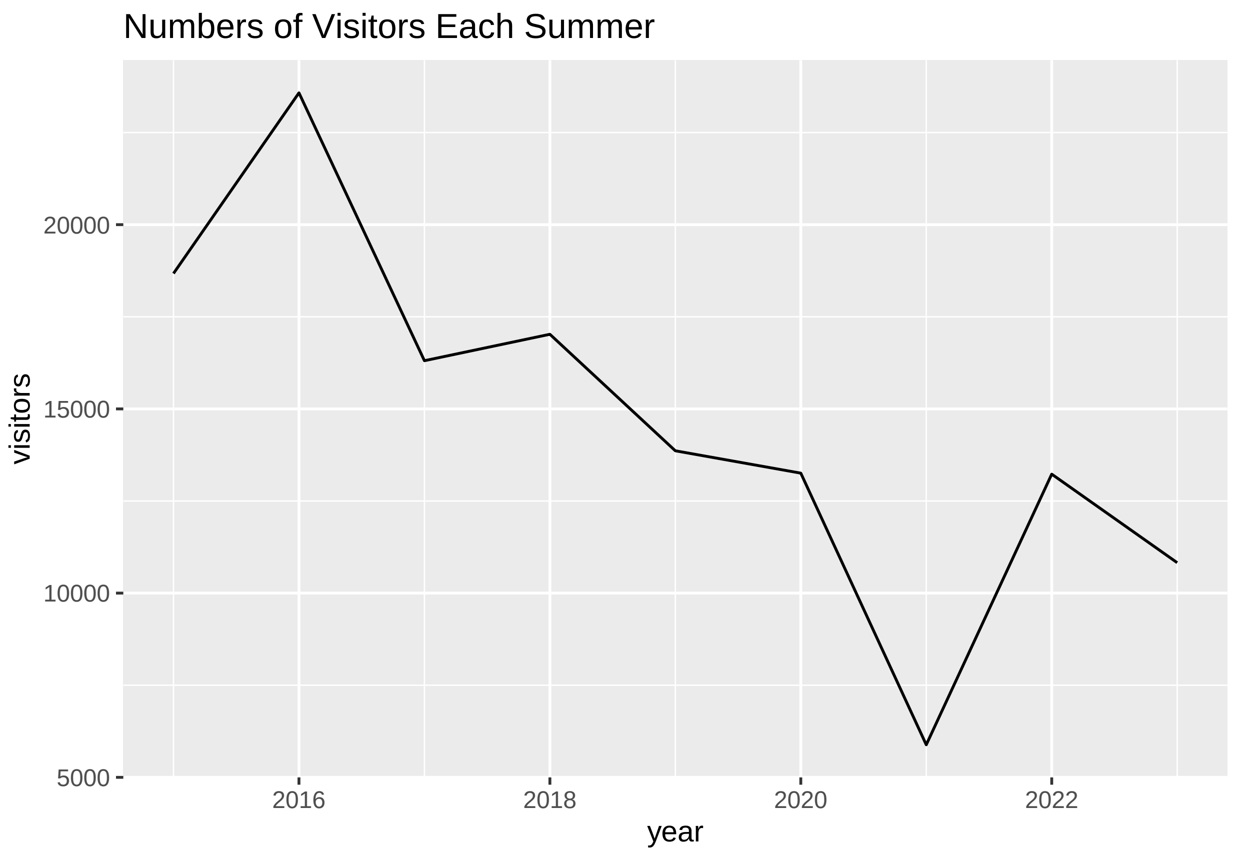
```
## group_by: one grouping variable (year(date))
## summarise: now 9 rows and 5 columns, ungrouped
## mutate: new variable 'year' (double) with 9 unique values and 0% NA
```

```
gorges_tbl
```

```
## # A tibble: 9 x 6
##   `year(date)` shifts_completed visitors violations avg_temp  year
##          <dbl>            <int>    <dbl>      <dbl>    <dbl> <dbl>
## 1         2015              124    18675        407     77.9  2015
## 2         2016               99    23578        807     82.3  2016
## 3         2017              107    16308        638     74.8  2017
## 4         2018               99    17025        283     74.8  2018
## 5         2019               91    13863         69     78.0  2019
## 6         2020              154    13256        958     81.7  2020
## 7         2021               96     5883        173     79.4  2021
## 8         2022              107    13229        562     81.2  2022
## 9         2023               98    10826        273     77.2  2023
```

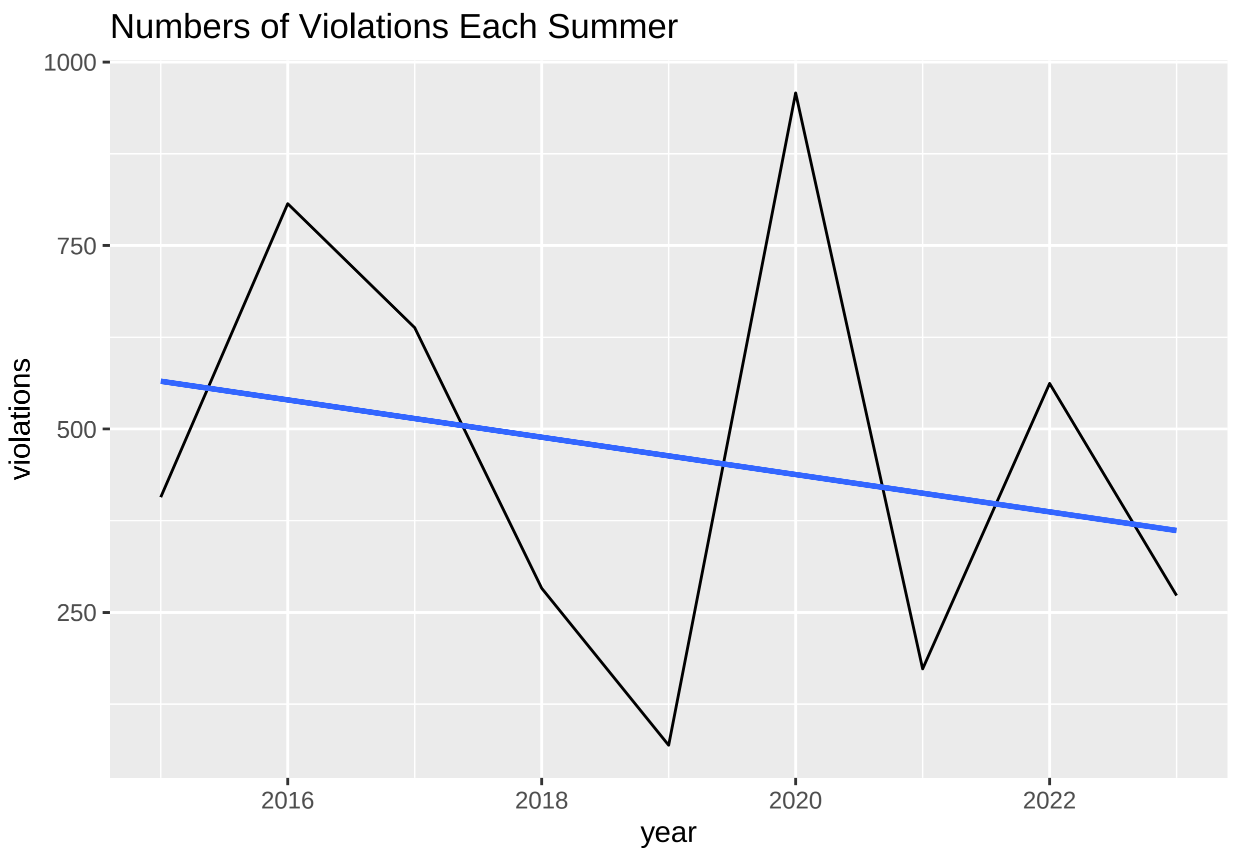
We see some interesting points. Overall, it seems that the around 14,000 visitors experience the Gorges every year, and around 12 stewards patrol every Summer. The average temperature is a high 70s, and around 100 shifts are completed every year. Notably are the years 2018, 2019, 2020, and 2021.

```
gorges_tbl |>
  ggplot(aes(x = year, y = visitors)) +
  geom_line() +
  labs(
    title = "Numbers of Visitors Each Summer"
  )
```

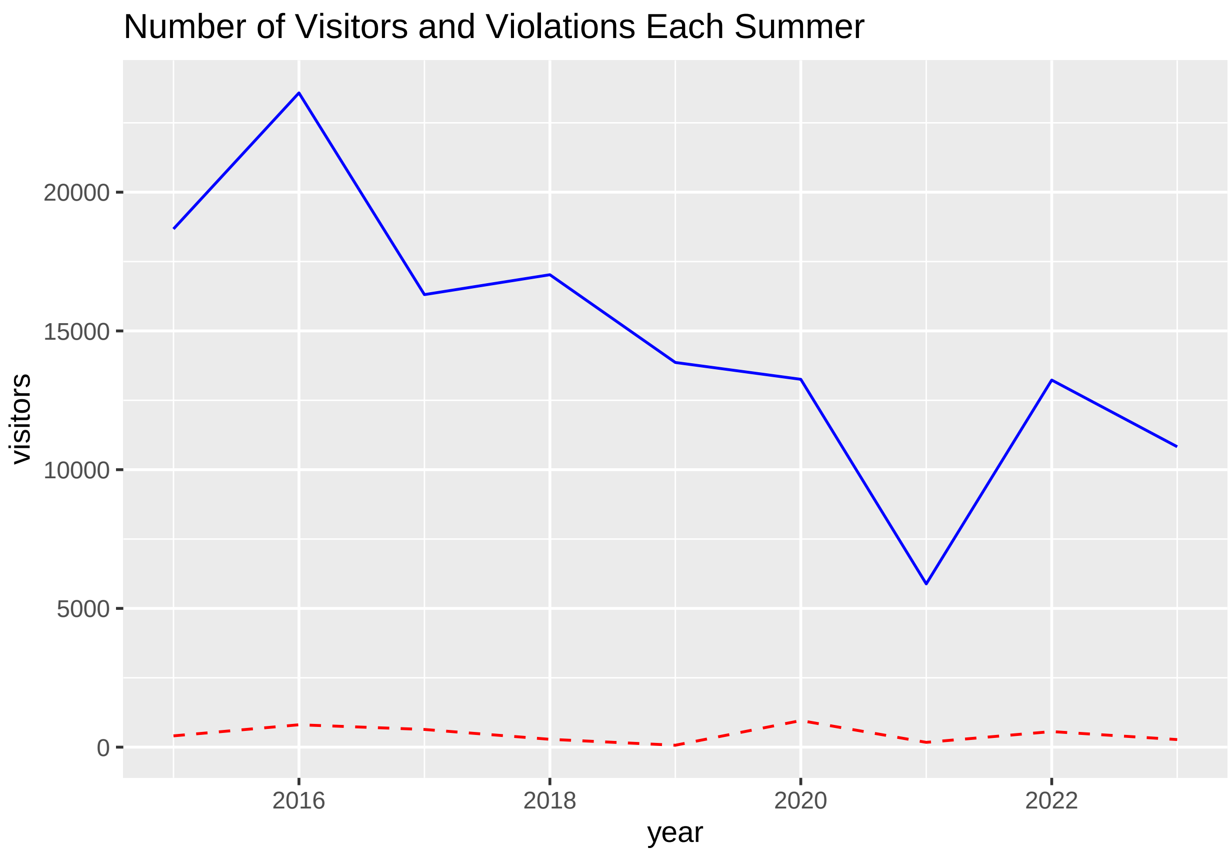


```
gorges_tbl |>
  ggplot(aes(x = year, y = violations)) +
  geom_line() +
  geom_smooth(method = lm, se = F) +
  labs(
    title = "Numbers of Violations Each Summer"
  )
```

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



```
gorges_tbl |>
  ggplot(aes(x = year)) +
  geom_line(aes(y = visitors), color = "blue", linetype = "solid") +
  geom_line(aes(y = violations), color = "red", linetype = "dashed") +
  labs(
    title = "Number of Visitors and Violations Each Summer")
```



Interestingly, we see the amount of visitors spikes in 2016, and falls to it lowest in 2021. 2020 shows only slightly lesser numbers than previous years, and 2022 shows membership returned near fully. It's hard to determine why this drop occured - the COVID-19 Pandemic would be a likely explanation, but that doesn't seem to account for 2020's regular number.

Over the 8 years of the program, violations overall have a downward trend, which is a good outcome of the Stewards Program. However, the decline is not steady, and there's a noticeable spike in 2020 and 2022. Again, this could be explained by the COVID pandemic: one of the few activities during the pandemic was going outside, and perhaps individuals did not follow rules as they were unaware of them or more careless during that year.

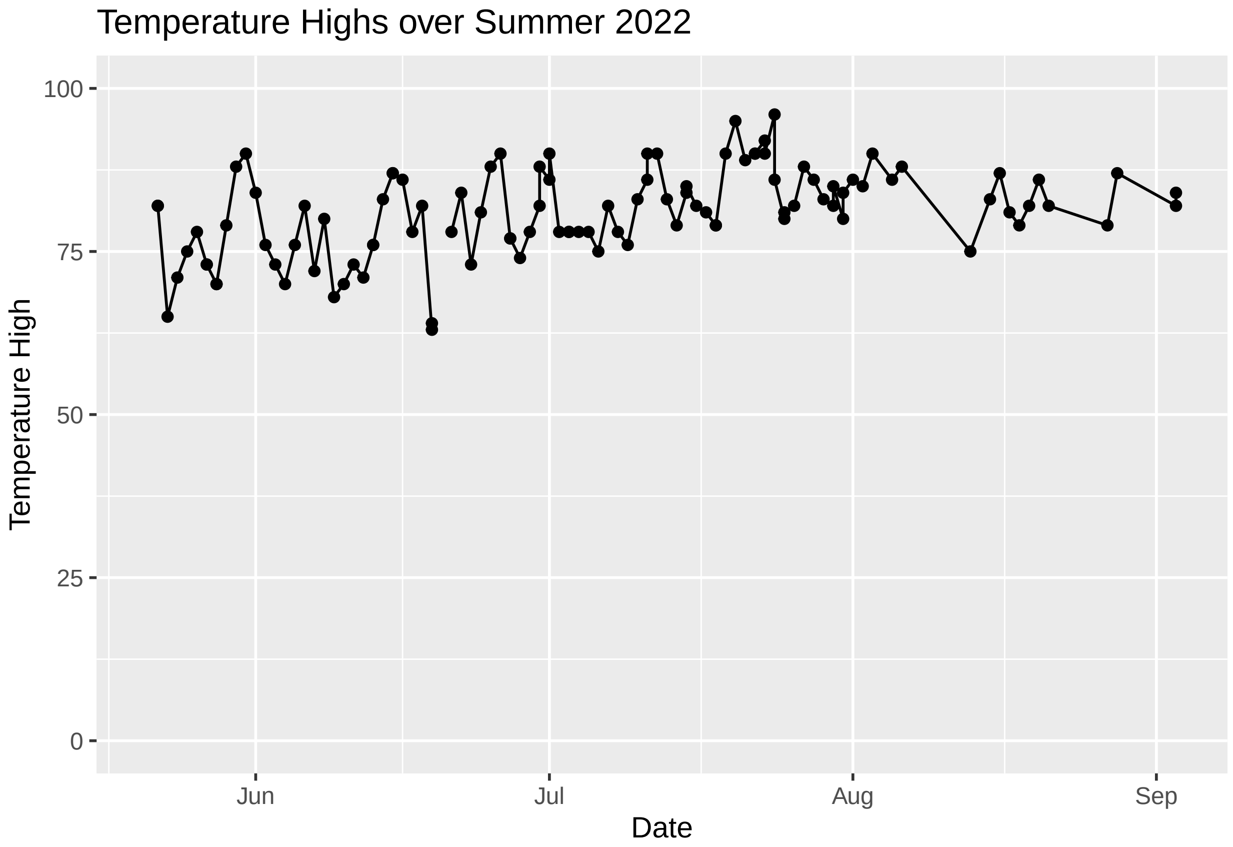
A Look at 2022

We now choose to look at a specific year. We select 2022 as it has high visitor numbers and is more recent.

```
gorge_2022 |>
  ggplot(aes(x = date, y = high_temperature)) +
  geom_point() + ylim(0, 100) + geom_line() +
  labs(
    title = 'Temperature Highs over Summer 2022',
    x = "Date",
    y = 'Temperature High'
  )
```

```
## Warning: Removed 2 rows containing missing values (`geom_point()`).
```

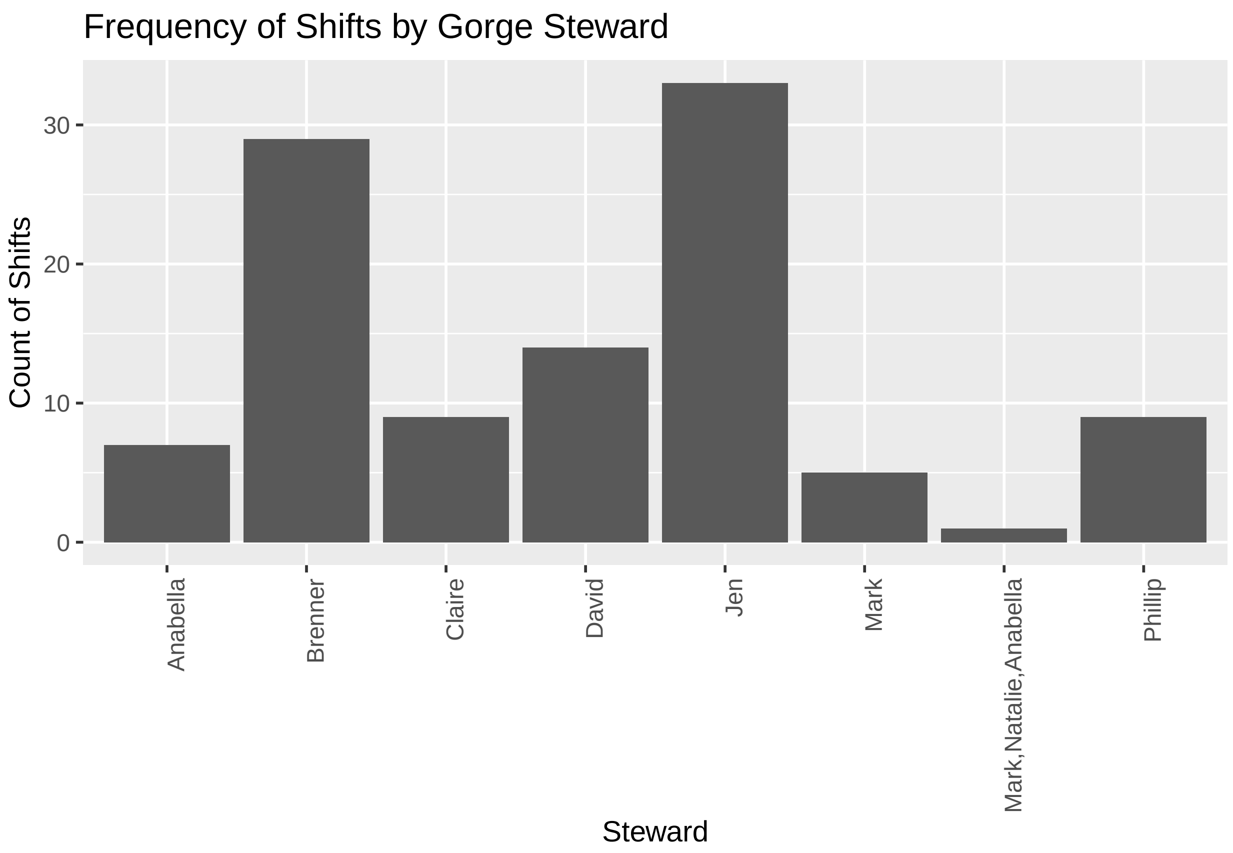
```
## Warning: Removed 1 row containing missing values (`geom_line()`).
```



```
gorge_2022 |>
  filter(steward_name != '') |>
  ggplot(aes(x = steward_name)) +
  geom_histogram(stat = 'Count') +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  labs(
    title = 'Frequency of Shifts by Gorge Steward',
    x = "Steward",
    y = 'Count of Shifts'
  )
```

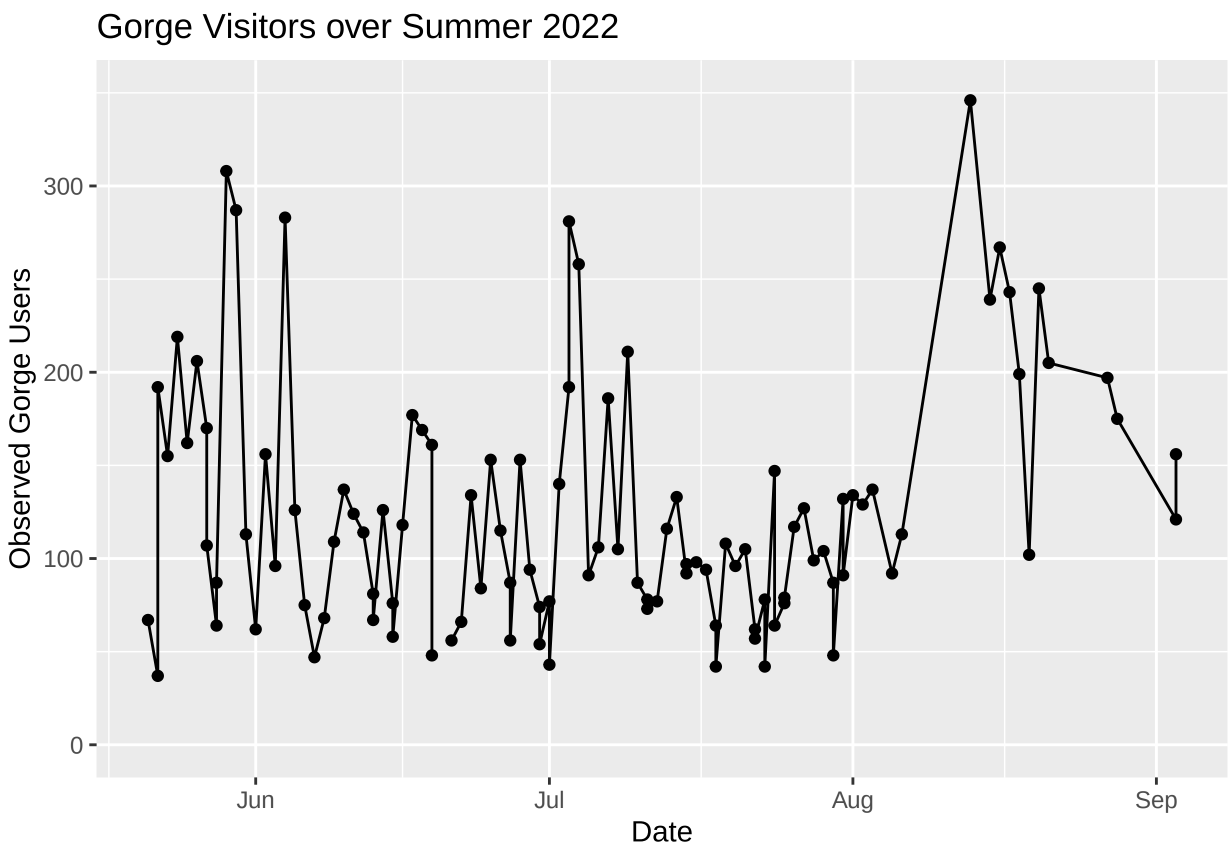
```
## filter: removed one row (1%), 107 rows remaining
```

```
## Warning in geom_histogram(stat = "Count"): Ignoring unknown parameters:
## `binwidth`, `bins`, and `pad`
```

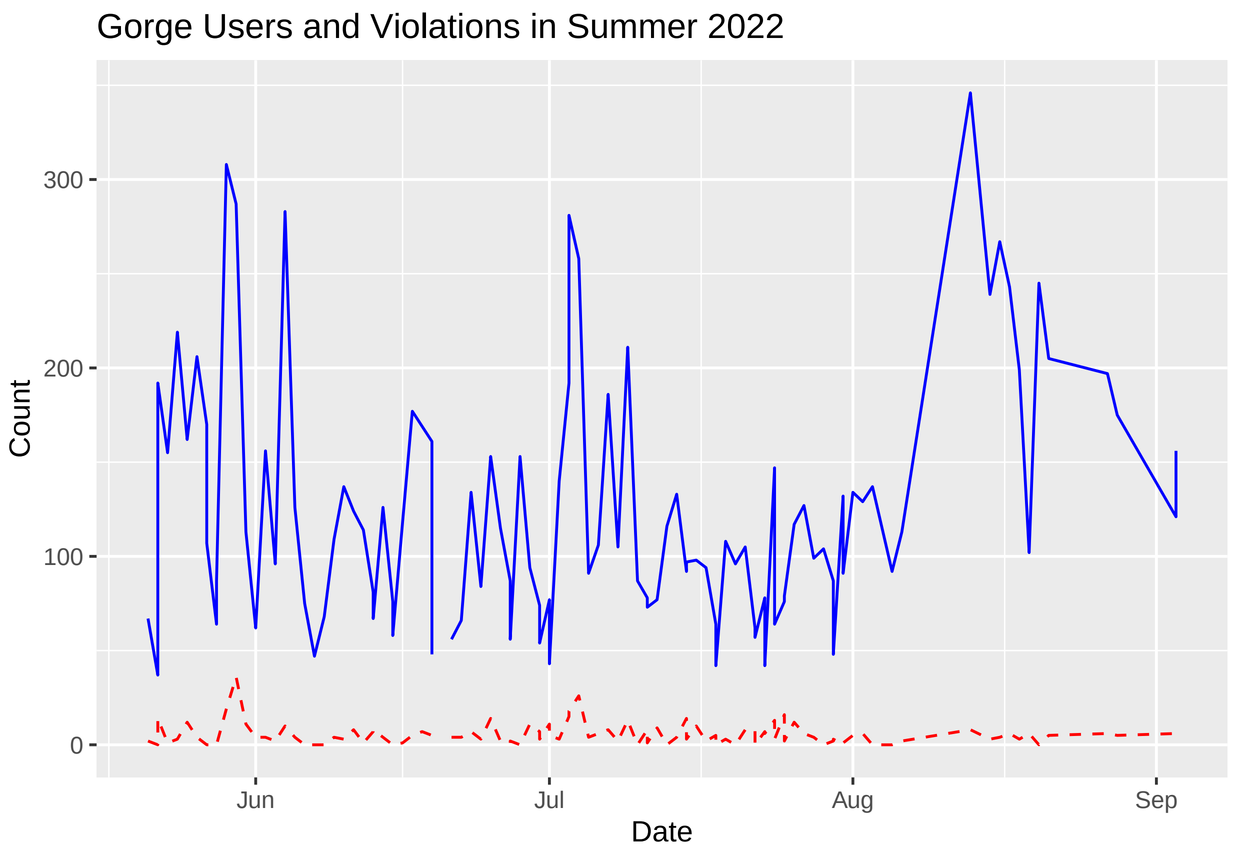


```
gorge_2022 |>
  ggplot(aes(x = date, y = observed_gorge_users)) +
  geom_point() + geom_line() + ylim(0, 350) +
  labs(
    title = 'Gorge Visitors over Summer 2022',
    x = 'Date',
    y = 'Observed Gorge Users'
  )
```

```
## Warning: Removed 1 rows containing missing values (`geom_point()`).
```



```
gorge_2022 |>
  ggplot(aes(x = date)) +
  geom_line(aes(y = observed_gorge_users), color = "blue", linetype = "solid") +
  geom_line(aes(y = observed_violations), color = "red", linetype = "dashed") +
  labs(title = "Gorge Users and Violations in Summer 2022",
       x = "Date",
       y = "Count")
```



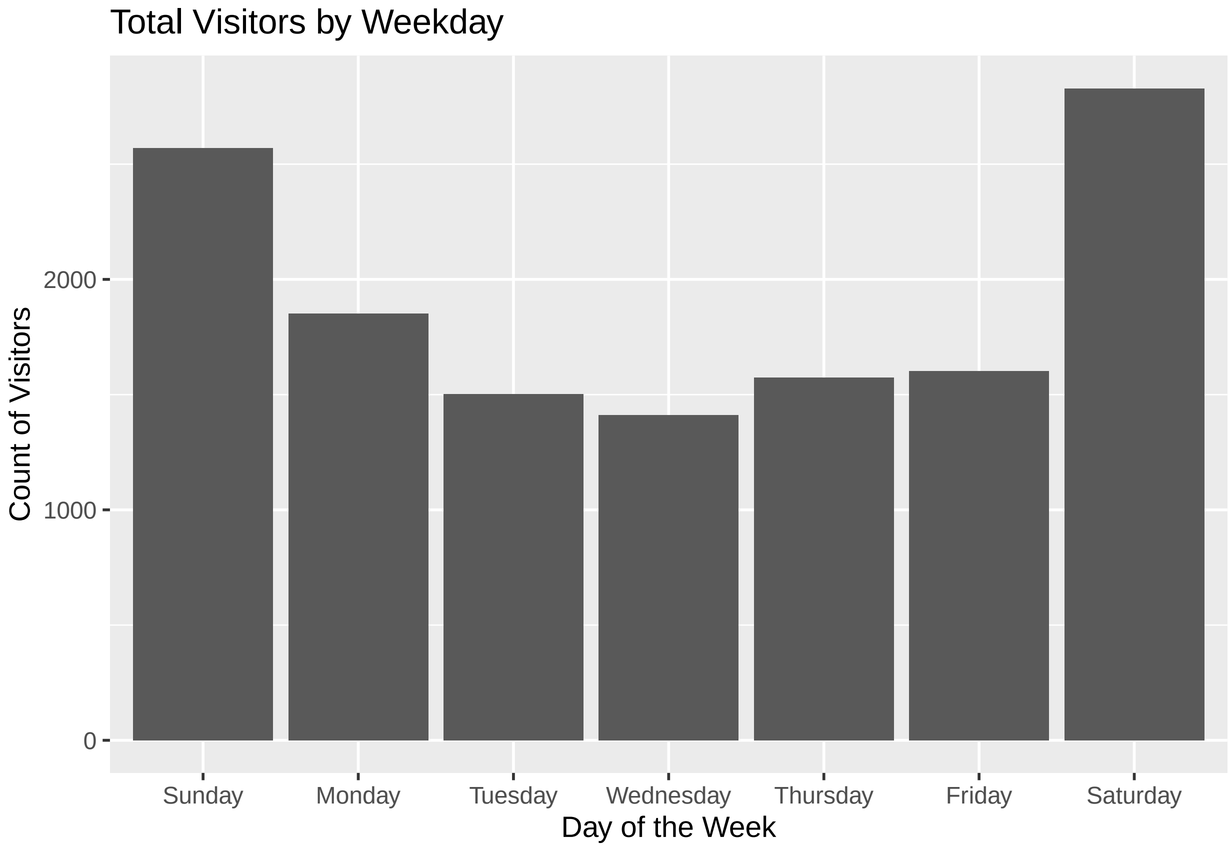
```
gorge_2022 |>
  drop_na(observed_gorge_users) |>
  mutate(dayofweek = weekdays(date)) |>
  group_by(dayofweek) |>
  summarize(weekday_count = sum(observed_gorge_users)) |>
  ggplot(aes(dayofweek, weekday_count)) +
  geom_col() +
  scale_x_discrete(limits = c("Sunday",
                              "Monday",
                              "Tuesday",
                              "Wednesday",
                              "Thursday",
                              "Friday",
                              "Saturday")) +
  labs(
    title = "Total Visitors by Weekday",
    x = 'Day of the Week',
    y = "Count of Visitors"
  )
```

```
## drop_na: removed one row (1%), 107 rows remaining

## mutate: new variable 'dayofweek' (character) with 7 unique values and 0% NA

## group_by: one grouping variable (dayofweek)

## summarize: now 7 rows and 2 columns, ungrouped
```



In the Summer of 2022, we find that some 6 stewards walked the Gorges, counting 13,229 visitors and over 500 violations. The temperature hovers around 80 degrees Fahrenheit and amount of visitors peaks at above 300, but usually hovers around 115.

The peaks are early June, July 4th, and late August. This makes sense as early June is when students finished with finals can explore campus a bit before they leave, and those who stay in Ithaca look for some adventure. July 4th is of course a time when many will want to enjoy nature with family, and late August is when students return, many excited to visit the famous gorges. The peak continues though early-September (when the Gorge Stewarding program pauses), as there are simply more students on campus.

Violations seem to largely follow the peaks of visitors - notiable in June and July 4th. Overall, however, violations seem t decrease over time, and very few are recorded in late-August.

Finally, we see that Saturday and Sunday, the weekends, are the most popular days to visit the gorges, while Wednesday is the least popular.

Possible Correlations

We are now interested in asking - do certain factors correlate and/or cause higher amount of visitors or violators?

Correlation Matrix

We first construct a correlation matrix

```
gorge_2020_raw |>
  select(High.Temperature, observed_gorge_users, observed_violations) |>
  drop_na() |>
  cor(method = "pearson", use = "complete.obs")
```

```
## select: dropped 26 variables (Date, Patrol.Time, Steward.name, Weather, Number.of.observed.gorge.use
```

```
## drop_na: removed 108 rows (59%), 75 rows remaining
```

```
##                       High.Temperature observed_gorge_users observed_violations
## High.Temperature            1.00000000           0.09964287           0.1918108
## observed_gorge_users        0.09964287           1.00000000           0.4875066
## observed_violations         0.19181080           0.48750661           1.0000000
```

```
gorge_2021_raw |>
  select(High.Temperature, observed_gorge_users, observed_violations) |>
  cor(method = "pearson", use = "complete.obs")
```

```
## select: dropped 14 variables (Date, Patrol.Time, Steward.name, Weather, Number.of.observed.gorge.use
```

```
##                       High.Temperature observed_gorge_users observed_violations
## High.Temperature             1.0000000           0.18983626          0.16621664
## observed_gorge_users         0.1898363           1.00000000          0.06606966
## observed_violations          0.1662166           0.06606966          1.00000000
```

```
gorge_2022 |>
  select(high_temperature, observed_gorge_users, observed_violations) |>
  cor(method = "pearson", use = "complete.obs")
```

```
## select: dropped 12 variables (date, patrol_time, steward_name, weather, person_interactions_alternat
```

```
##                       high_temperature observed_gorge_users observed_violations
## high_temperature            1.00000000           -0.1696086          0.09303063
## observed_gorge_users       -0.16960856            1.0000000          0.46832471
## observed_violations         0.09303063            0.4683247          1.00000000
```

```
gorge_2023_raw |>
  select(High.Temperature, observed_gorge_users, observed_violations) |>
  cor(method = "pearson", use = "complete.obs")
```

```
## select: dropped 17 variables (Date, Patrol.Time, Steward.name, Weather, Number.of.observed.gorge.use
```

```
##                       High.Temperature observed_gorge_users observed_violations
## High.Temperature            1.00000000           0.04219599           0.2496043
## observed_gorge_users        0.04219599           1.00000000           0.4116002
## observed_violations         0.24960427           0.41160024           1.0000000
```

Interestingly, we see that the highest correlations are observed users to violations, and violations to temperature. We see that gorge users tend to correlate with violations by almost 0.50.

On average the r-squared value for the past 4 years is 0.5134591. 2023 has a correlation of 0.25.

However, temperature seems to have to no correlation with gorge visitors. Thus, high temperatures may not dissuade visiting the gorge, but it might cause more violations, This makes sense as hotter days mean visitors may be more likely to try swimming in the gorge or seeking out shady but restricted areas. Higher temperatures may also mean people are more irrate or less willing to abide by posted signs.

Future Work

There is still some interesting questions to be answered in this dataset. For example, how might certain variables predict number of violations? How are violations and interactions described?

Text Analysis

We are also given descriptions of each shift. We are interested in the most often used words and the general sentiment of each interaciton. Futher text analysis would be required to achieve this. The currently work is insufficient for results.

```
library(stringr)

common_prepositions <- c("this", "is", "a", "an", "the", "and", "with", "than", "of", "in", "on", "at",

most_freq_words <- function(df, column, top_n, ignore_words) {
  df %>%
    pull(!!column) %>%
    str_split("\\s+") %>%
    unlist() %>%
    tolower() %>%
    str_replace_all("[[:punct:]]$", "") %>%  # Remove trailing punctuation marks
    str_remove("\\d+") %>%
    setdiff(common_prepositions) %>%
    table() %>%
    as.data.frame() %>%
    arrange(desc(Freq))
}



interact_words <- most_freq_words(gorge_2022, 'description_of_interactions', 50)
```

Machine Learning

Some work in ML could also be done. This would include using some variables, such as temperature, number of visitors, the date, or Steward, to possibly predict amount of violators that day. Any model would likely be weak, but it would be an interesting experiment to train a classification model.

Conclusion

The Gorges of Ithaca are a beautiful natural landmark that attracts thousands of Visitors every Summer. The Gorges Program ensure the gorges are safe and enjoyable for everyone. As a part of their job, Gorge Stewards track certain data points that we analyze. Unsurprisingly, the most popular times to visit the Gorges are towards the beginning or end of the academic year, and on the Fourth of July. Overall number of visitors has increased, while violations have significantly decreased since the inception of the program. COVID did not seem to have too dramatic of an effect; although 2021 had the lowest visitor count by far.

It makes sense that as there are more visitors in the gorges, there will be more violations. Interestingly, however, it is also possible that high temperatures cause more violations.

There's more analysis that can be conducted, especially in terms of text analysis and model training.

The data reinforces the success of the Gorge Stewards Program and its positive effect on the Gorges. The program has evolved since its founding in 2012 (e.g. stewards used to offer tours, picking up trash) but its mission has remained the same: to protect and enjoy the beauty of the gorges.