A history of reported hate crimes in Toronto

Increased frequency poses risk to diversity

AUTHOR
Marzia Zaidi

PUBLISHED
September 26, 2024

ABSTRACT
Upon studying the history of reported hate crimes in Toronto, we observe that there has been a sharp spike in 2023. We also observe that most of the hate crimes happen during the day, i.e. in broad daylight. Finally, we see that while most of the crims are reported almost instantaneously, there are some that take up to a day to repor.

1 Introduction

You can and should cross-reference sections and sub-sections. We use @citeR and @rohan.

The remainder of this paper is structured as follows. Section 2....

2 Data

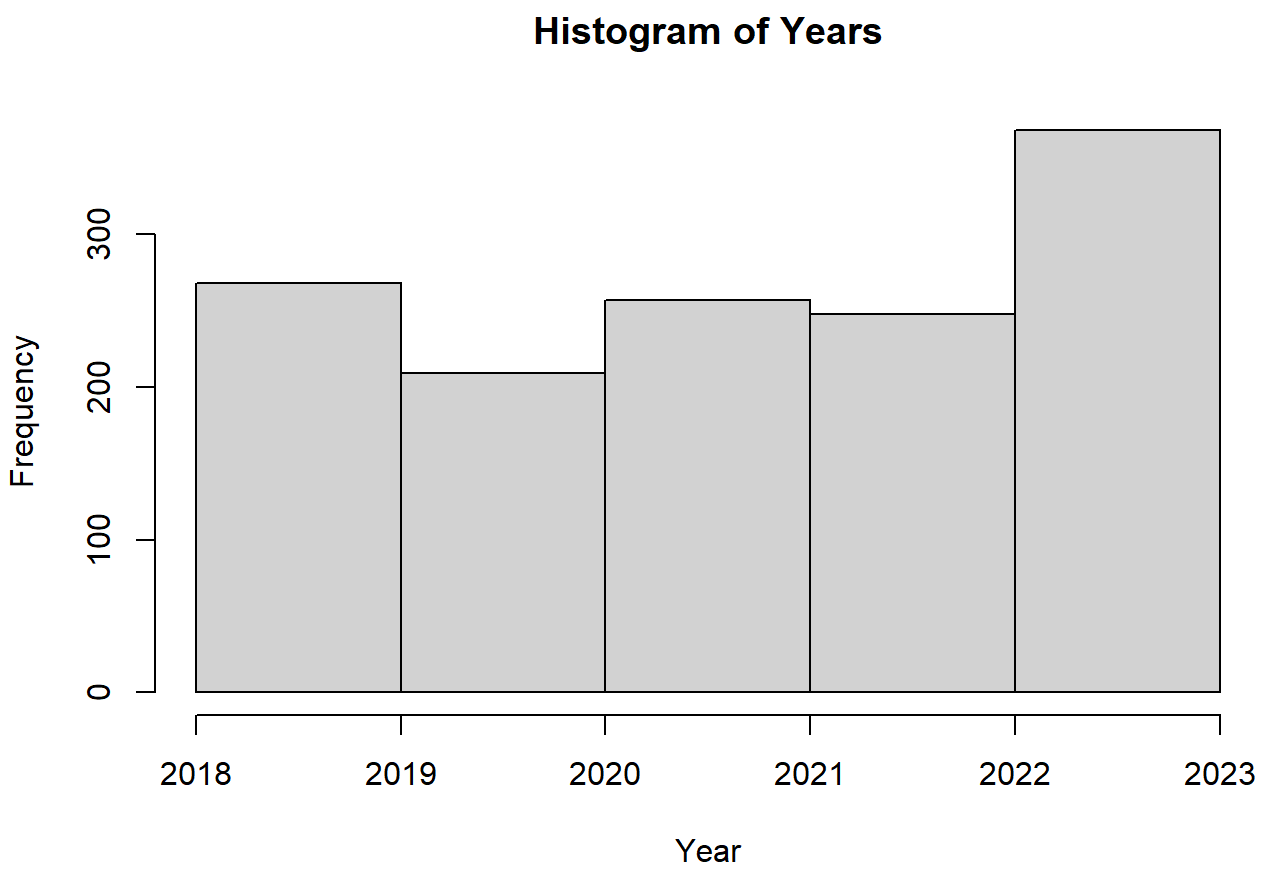
Some of our data is of penguins (**?@fig-bills**), from @palmerpenguins.

```
library(opendatatoronto)
```

```
Warning: package 'opendatatoronto' was built under R version 4.3.3
```

```
library(tidyverse)
library(dplyr)
nameOfDataSet <- "hate-crimes-open-data"
#### Download data ####
package <- show_package(nameOfDataSet)

# get all resources for this package
resources <- list_package_resources(nameOfDataSet)

# identify datastore resources; by default, Toronto Open Data sets datastore resource format to CS

datastore_resources <- filter(resources, tolower(format) %in% c('csv', 'geojson'))

# load the first datastore resource as a sample
data <- filter(datastore_resources, row_number()==1) %>% get_resource()

# first, we analyze if the number of crimes have increased over the years

# Convert the string years to integers
years_numeric <- as.integer(data$REPORTED_YEAR)
```

```r
# Create a histogram of the years
hist(years_numeric, breaks = 7, col = "lightgrey",
     main = "Histogram of Years",
     xlab = "Year",
     ylab = "Frequency")
```



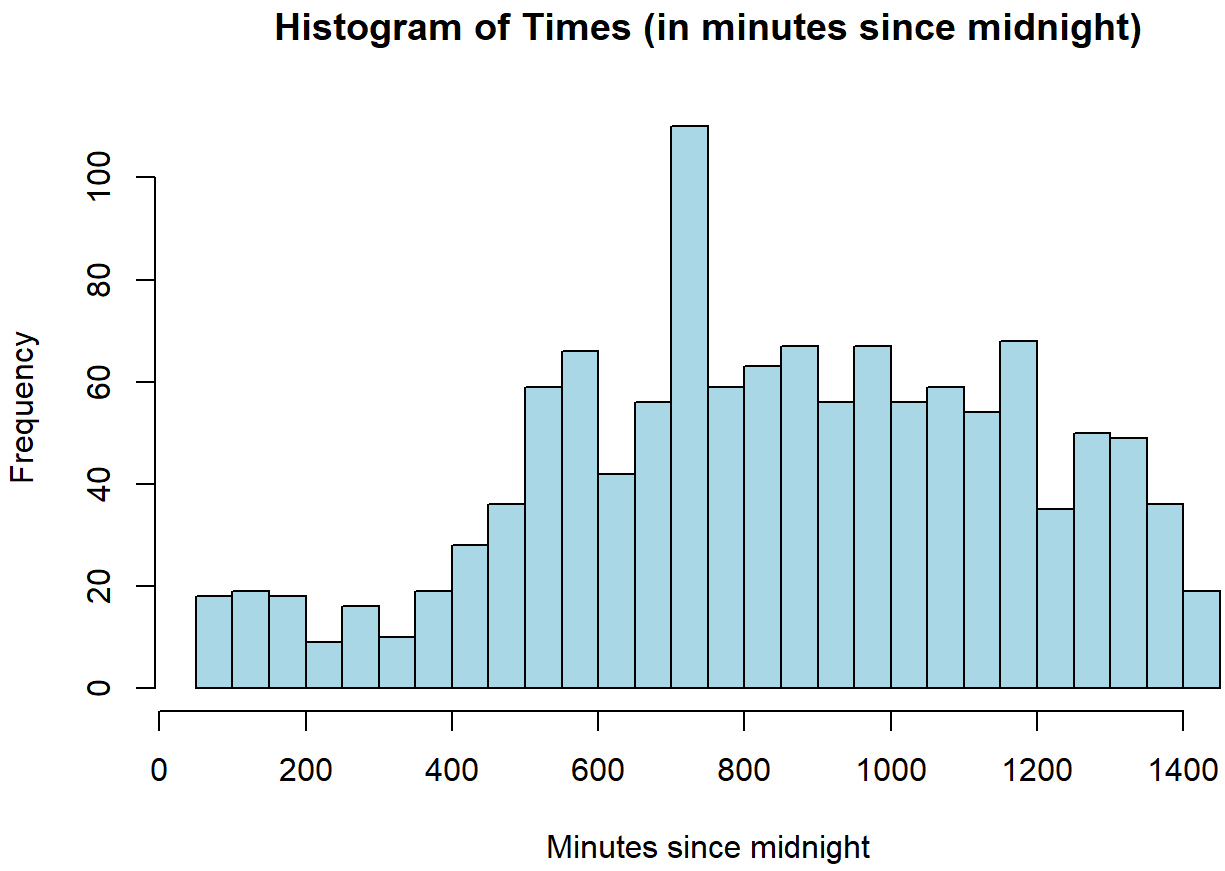

```r
# Looking at this, we can see that the trend was mostly non existent till 2022, and then in 2023,
```

Talk more about it.

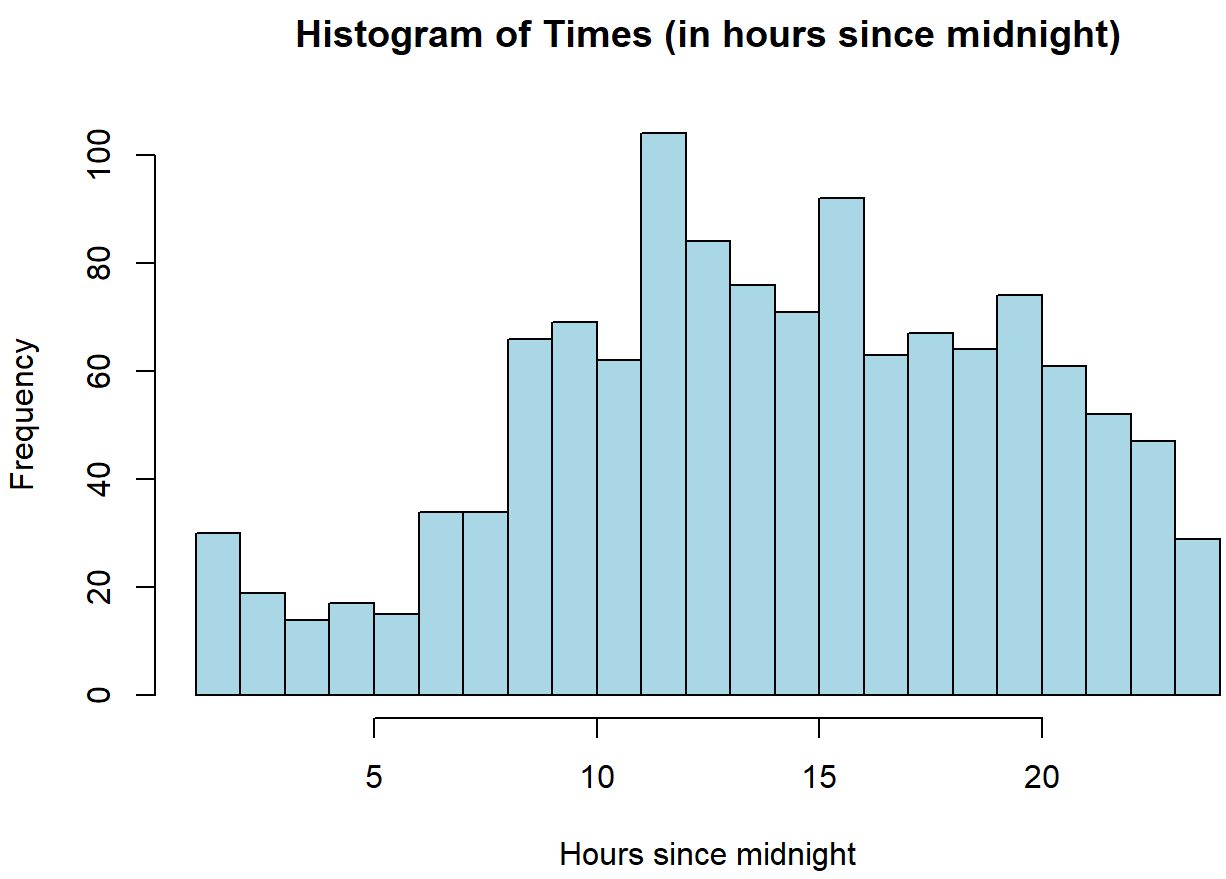
And also planes (**?@fig-planes**). (You can change the height and width, but don't worry about doing that until you have finished every other aspect of the paper - Quarto will try to make it look nice and the defaults usually work well once you have enough text.)

```r
# we want to analyze if the crimes happened at night
# Here is the vector of time integers
times <- data$OCCURRENCE_TIME

# Pad integers to ensure four digits (HHMM format)
times_padded <- sprintf("%04s", times)

# Extract hours and minutes
```

```r
hours <- as.integer(substr(times_padded, 1, 2))  # First two characters are the hours
minutes <- as.integer(substr(times_padded, 3, 4))  # Last two characters are the minutes

# Convert to total minutes since midnight
total_minutes <- hours * 60 + minutes


# Create a histogram of the total minutes since midnight
hist(total_minutes, breaks = 24, col = "lightblue",
     main = "Histogram of Times (in minutes since midnight)",
     xlab = "Minutes since midnight",
     ylab = "Frequency")
```



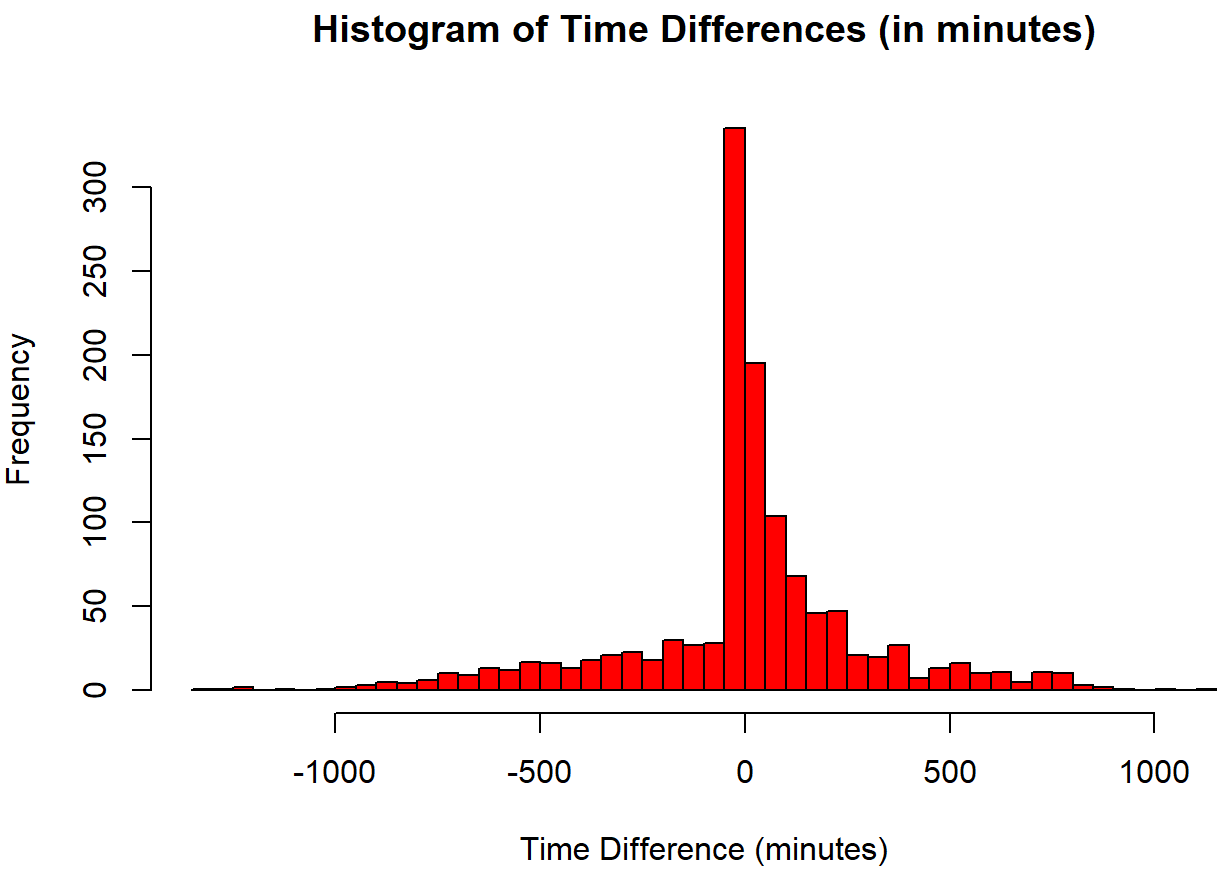

```r
# Convert total minutes to hours (fractional)
total_hours <- total_minutes / 60

# Create a histogram of the total hours since midnight
hist(total_hours, breaks = 24, col = "lightblue",
     main = "Histogram of Times (in hours since midnight)",
     xlab = "Hours since midnight",
     ylab = "Frequency")
```



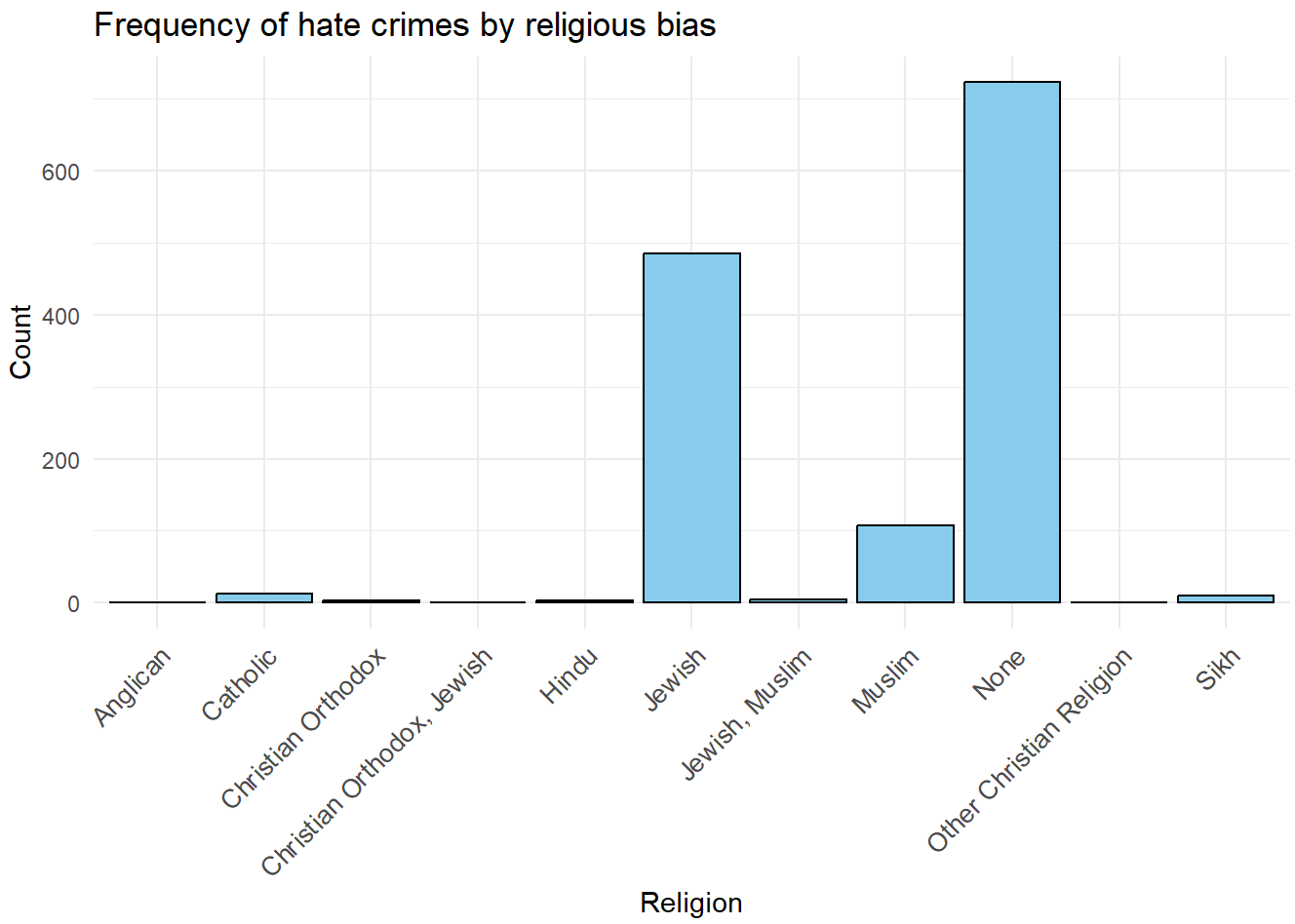

```
# we see most of these have been crimes during the day
```

```
# analyze the time between a crime happening and the report
# Function to convert string integers to total minutes since midnight
convert_to_minutes <- function(time_strings) {
  # Pad the time strings to ensure four digits (HHMM format)
  times_padded <- sprintf("%04s", time_strings)

  # Extract hours and minutes
  hours <- as.integer(substr(times_padded, 1, 2))  # First two characters are hours
  minutes <- as.integer(substr(times_padded, 3, 4))  # Last two characters are minutes

  # Convert to total minutes since midnight
  total_minutes <- hours * 60 + minutes
  return(total_minutes)
}

# Convert both columns to minutes since midnight
time_start_minutes <- convert_to_minutes(data$OCCURRENCE_TIME)
time_end_minutes <- convert_to_minutes(data$REPORTED_TIME)


# Calculate time differences (end time - start time)
```

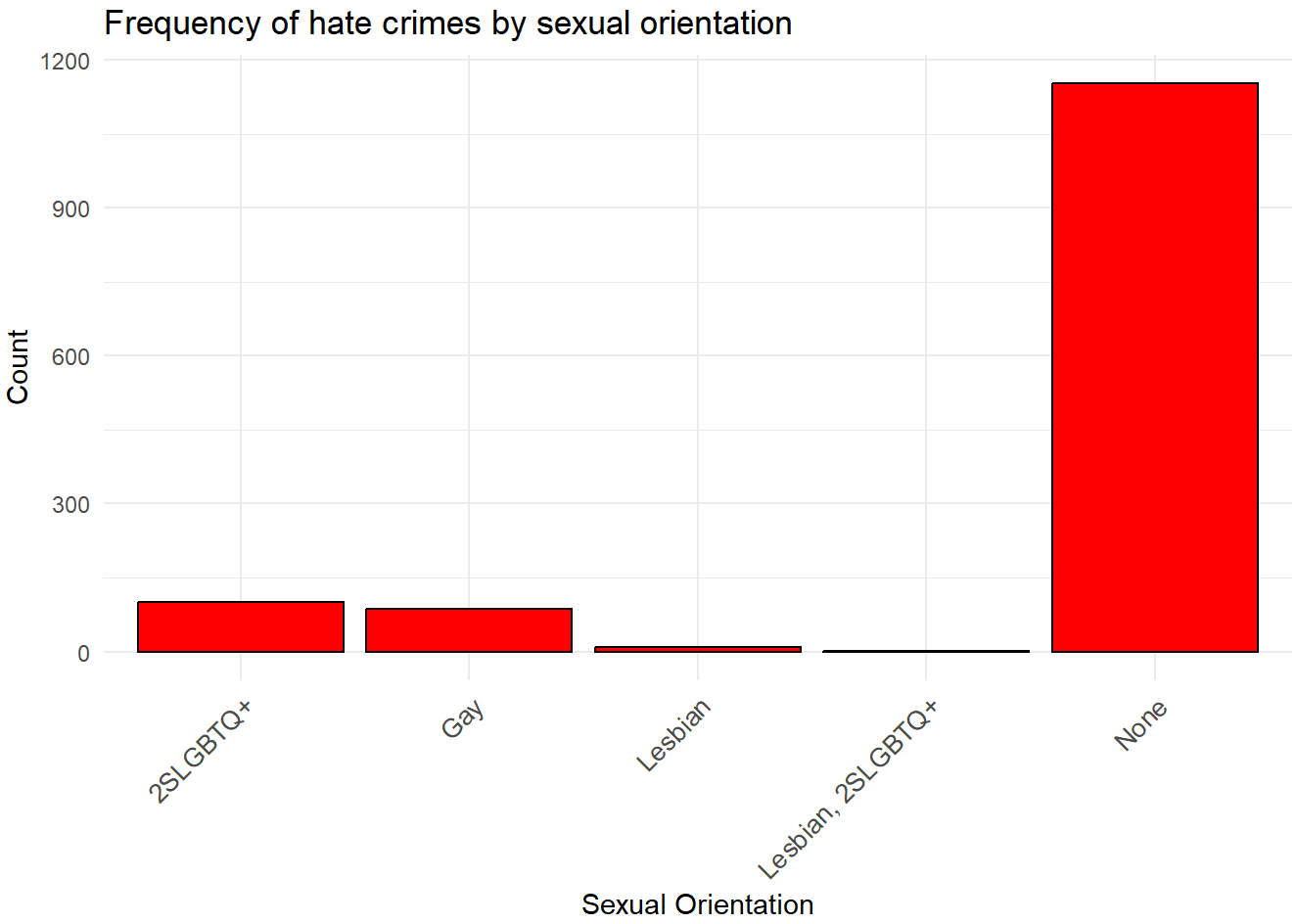
```r
time_differences <- time_end_minutes - time_start_minutes

# Create a histogram of the time differences
hist(time_differences, breaks = 50, col = "red",
     main = "Histogram of Time Differences (in minutes)",
     xlab = "Time Difference (minutes)",
     ylab = "Frequency")
```



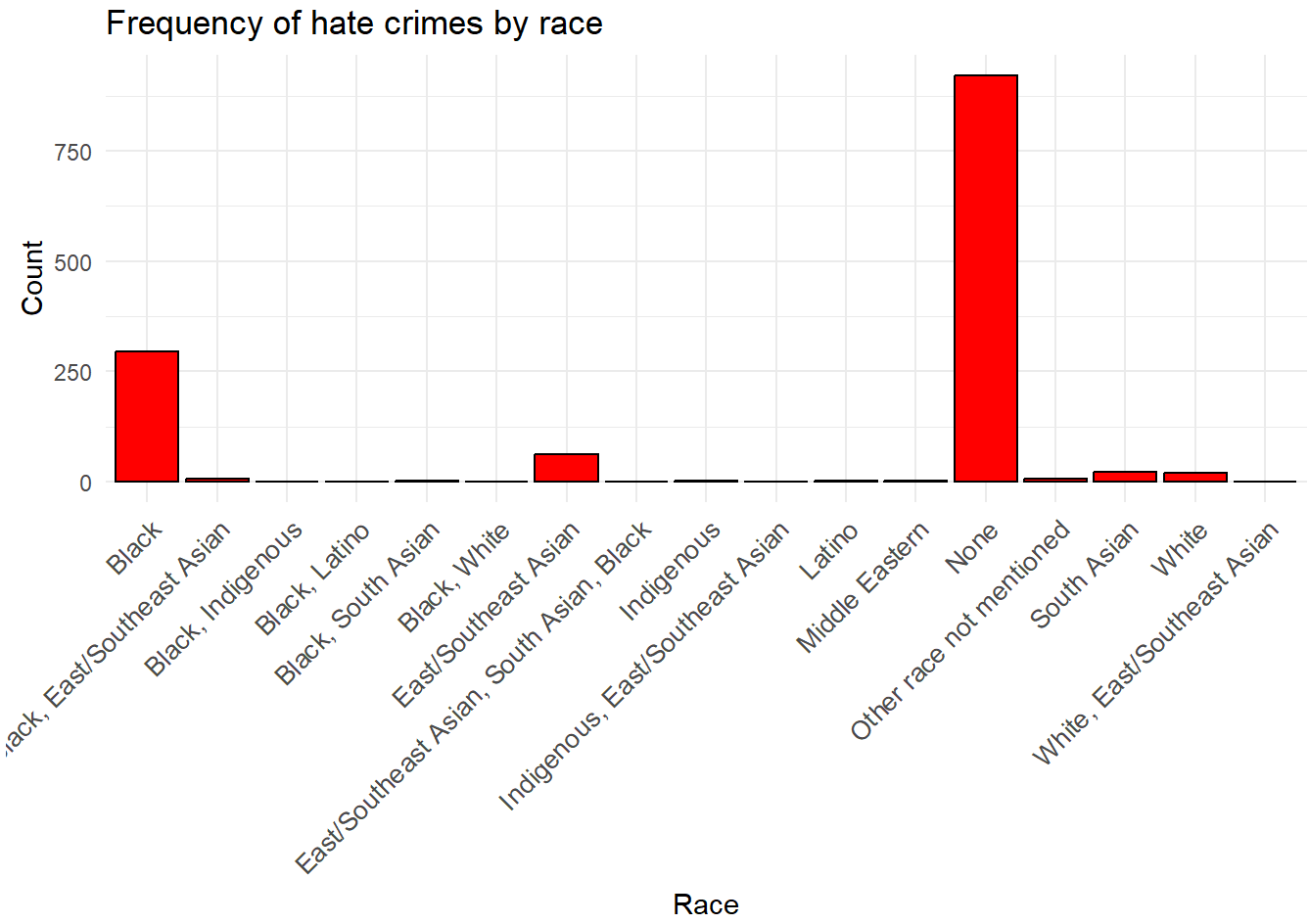

```r
# most of the crimes were rerported very quickly, with some being reported the next day (correspor

# this is a histogram of crimes by religion
# Install and load necessary packages
library(ggplot2)

# Create the bar plot
ggplot(data, aes(x = RELIGION_BIAS)) +
  geom_bar(fill = "skyblue", color = "black") +
  labs(title = "Frequency of hate crimes by religious bias", x = "Religion", y = "Count") +
  theme_minimal() + theme(axis.text.x = element_text(angle = 45, hjust = 1, size = 10))
```




```
# this is a histogram of crimes by sexual orientation

# Create the bar plot
ggplot(data, aes(x = SEXUAL_ORIENTATION_BIAS)) +
  geom_bar(fill = "red", color = "black") +
  labs(title = "Frequency of hate crimes by sexual orientation", x = "Sexual Orientation", y = "C
  theme_minimal() + theme(axis.text.x = element_text(angle = 45, hjust = 1, size = 10))
```




```
# this is a histogram of crimes by race

# Create the bar plot
ggplot(data, aes(x = RACE_BIAS
                 )) +
  geom_bar(fill = "red", color = "black") +
  labs(title = "Frequency of hate crimes by race", x = "Race", y = "Count") +
  theme_minimal() + theme(axis.text.x = element_text(angle = 45, hjust = 1, size = 10))
```



Talk way more about it.

3 Model

The goal of our modelling strategy is to show that the number of crimes has increased over the years. We use a linear regression model to prove this

3.1 Model set-up

3.1.1 Model justification

4 Results

Our results are summarized in

```
data$OCCURRENCE_YEAR <- as.integer(data$OCCURRENCE_YEAR)
frequency_table <- table(data$OCCURRENCE_YEAR)
print(frequency_table)
```

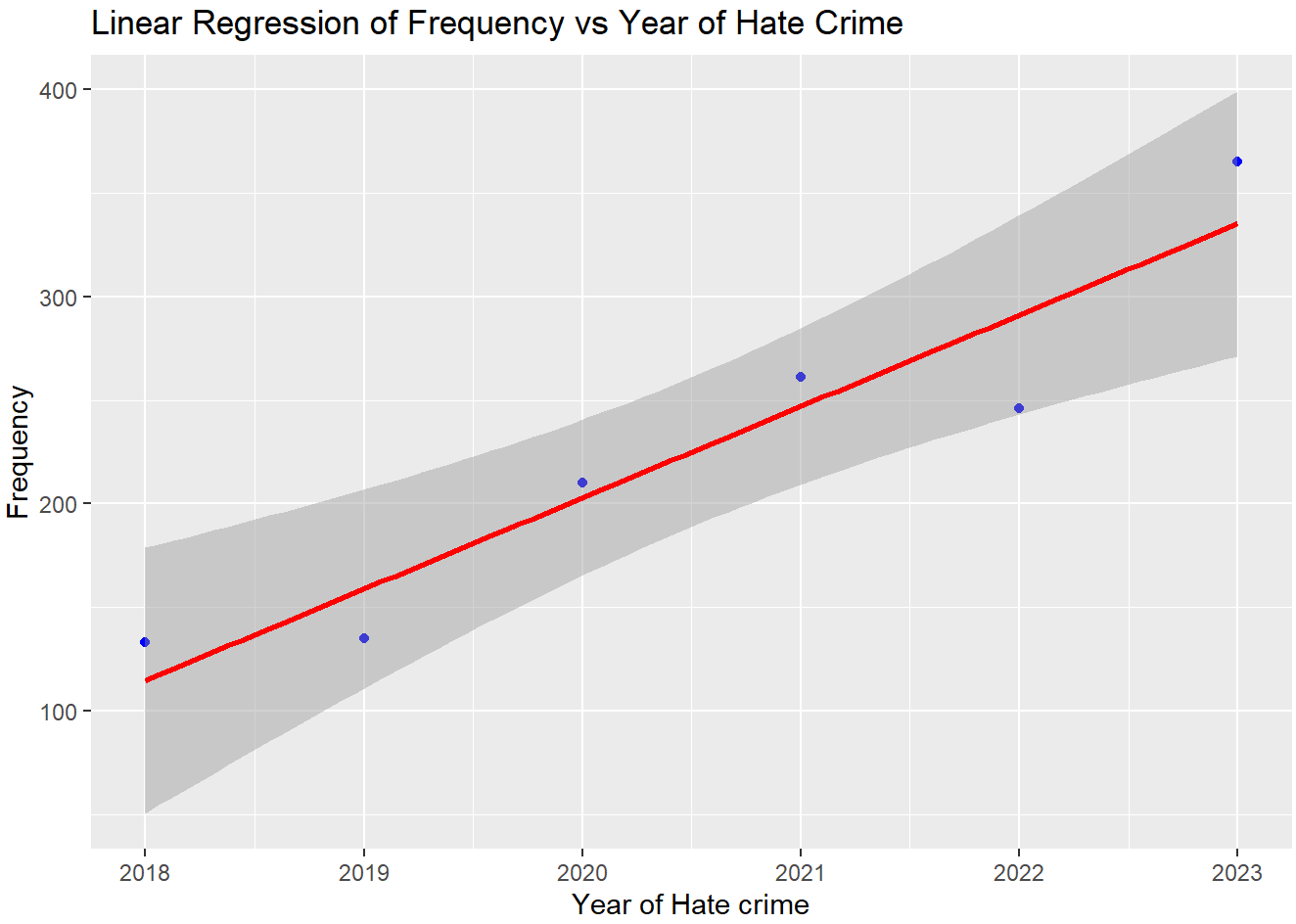
```
2018 2019 2020 2021 2022 2023
 133  135  210  261  246  365
```

```r
frequency_df <- as.data.frame(frequency_table)
colnames(frequency_df) <- c("Year_of_hate_crime", "frequency")
frequency_df$Year_of_hate_crime <- as.numeric(as.character(frequency_df$Year_of_hate_crime))
# Fit the linear regression model
model <- lm(frequency ~ Year_of_hate_crime, data = frequency_df)

# Print the summary of the model
summary(model)
```

```
Call:
lm(formula = frequency ~ Year_of_hate_crime, data = frequency_df)

Residuals:
      1       2       3       4       5       6
 18.286 -23.829   7.057  13.943 -45.171  29.714

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)        -88907.914  15406.181  -5.771  0.00448 **
Year_of_hate_crime     44.114      7.625   5.786  0.00443 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 31.9 on 4 degrees of freedom
Multiple R-squared:  0.8933,    Adjusted R-squared:  0.8666
F-statistic: 33.47 on 1 and 4 DF,  p-value: 0.004435
```

```r
# Load ggplot2
library(ggplot2)
ggplot(frequency_df, aes(x = Year_of_hate_crime, y = frequency)) +
  geom_point(color = "blue") +
  geom_smooth(method = "lm", color = "red") +
  labs(title = "Linear Regression of Frequency vs Year of Hate Crime",
       x = "Year of Hate crime",
       y = "Frequency")
```

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


Linear Regression of Frequency vs Year of Hate Crime
Frequency
400
300
200
100
2018
2019
2020
2021
2022
2023
Year of Hate crime