Project Analysis

Zeerak Ahmed

11-23-2025

```r
library(tidyr)
```

```
## Warning: package 'tidyr' was built under R version 4.5.2
```

```r
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(ggplot2)
library(lubridate)
```

```
## Warning: package 'lubridate' was built under R version 4.5.2
```

```
##
## Attaching package: 'lubridate'
```

```
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

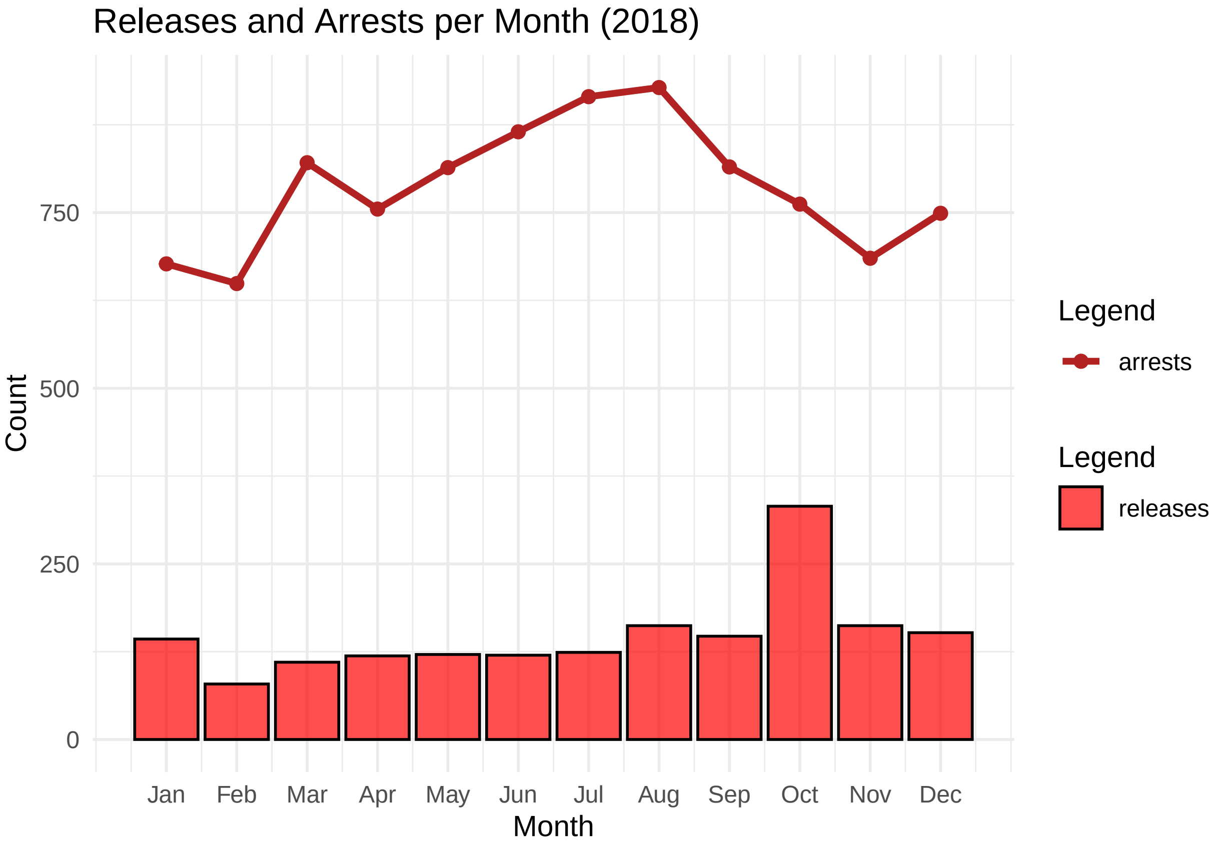
```r
horror_data <- read.csv("C:/Users/adeel/Downloads/archive (6)/horror_movies.csv")

# Extracts the year in the release_year column
horror_data[["year"]] <- as.integer(substr(horror_data[["release_date"]], 1, 4))

# Extracts the month in the release_year column
horror_data[["release_month"]] <- as.integer(substr(horror_data[["release_date"]], 6, 7))

######### Number of releases per month for each year ############
```

```r
# 2018
movies_2018_before <- horror_data %>%
  filter(year == 2018)

monthly_releases_2018 <- movies_2018_before %>%
      group_by(release_month) %>%
      summarise(count = n())

# 2019
movies_2019_before <- horror_data %>%
  filter(year == 2019)

monthly_releases_2019 <- movies_2019_before %>%
      group_by(release_month) %>%
      summarise(count = n())

# 2020
movies_2020_before <- horror_data %>%
  filter(year == 2020)

monthly_releases_2020 <- movies_2020_before %>%
      group_by(release_month) %>%
      summarise(count = n())

# 2021
movies_2021_before <- horror_data %>%
  filter(year == 2021)

monthly_releases_2021 <- movies_2021_before %>%
      group_by(release_month) %>%
      summarise(count = n())

###########################################################################
# Reads in Dallas crime rates data set
police_arrests <- read.csv("C:/Users/adeel/Downloads/archive (5)/Police_Arrests.csv")

# Removes duplicate rows
police_arrests <- police_arrests %>%
  distinct()

# Rename Arrest Year column to match year column in Horror data set
police_arrests <- police_arrests %>%
  rename(year = Arrest.Year)

# Converts Arrest Data column to a date object
police_arrests[["Arrest.Date"]] <- as.Date(police_arrests[["Arrest.Date"]],
                                       format = "%m/%d/%y")


# Extracts the month in the Arrest Date column
police_arrests[["arrest_month"]] <- as.integer(substr(police_arrests[["Arrest.Date"]], 6, 7))

# Counts arrest per year from 2014-2022
```

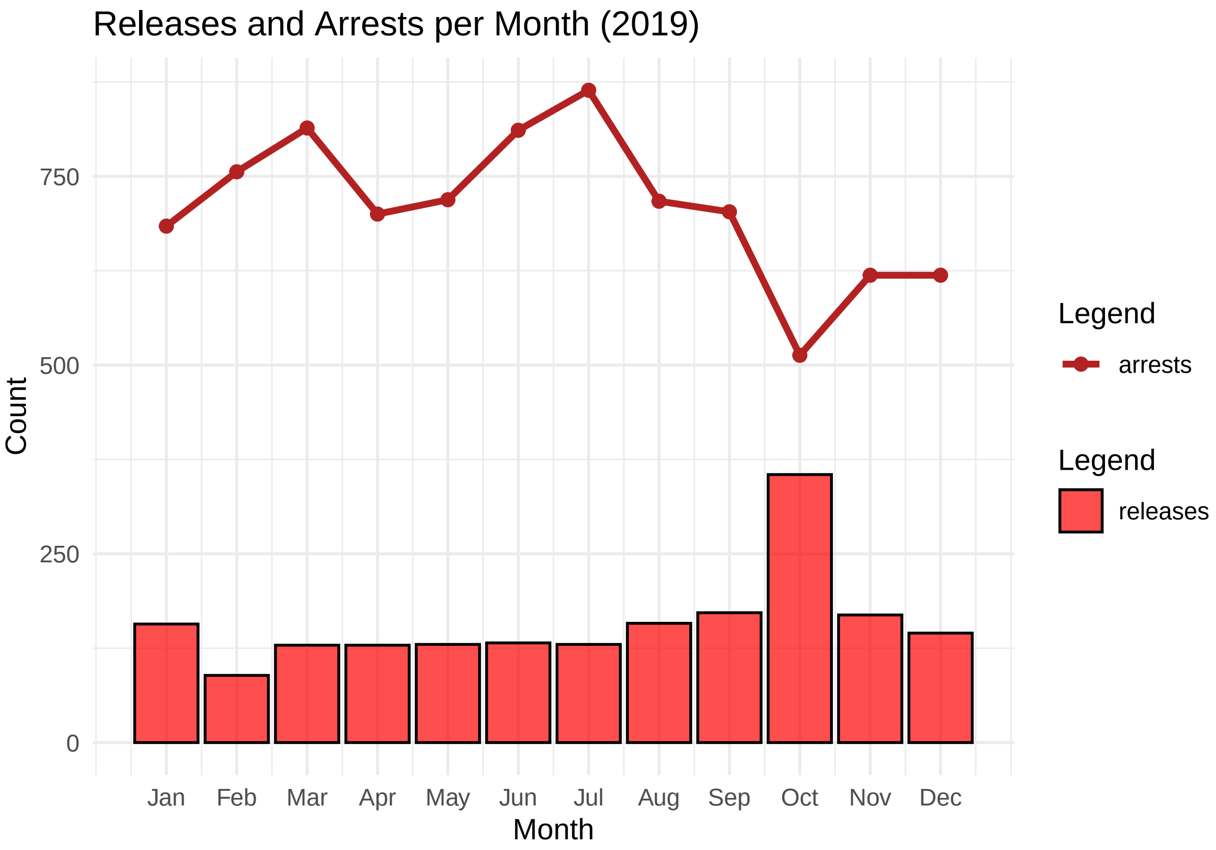
```r
arrests_before <- police_arrests %>%
      group_by(year) %>%
      summarise(count = n())

# Analysis will only be 2018-2022
police_arrests <- police_arrests %>%
  filter(year >= 2018 & year <= 2021)

# Arrests per month in 2018
arrests_2018 <- police_arrests %>%
  filter(year == 2018)

monthly_arrests_2018 <- arrests_2018 %>%
  group_by(arrest_month) %>%
  summarise(count = n())

# Arrests per month in 2019
arrests_2019 <- police_arrests %>%
  filter(year == 2019)

monthly_arrests_2019 <- arrests_2019 %>%
  group_by(arrest_month) %>%
  summarise(count = n())

# Arrests per month in 2020
arrests_2020 <- police_arrests %>%
  filter(year == 2020)

monthly_arrests_2020 <- arrests_2020 %>%
  group_by(arrest_month) %>%
  summarise(count = n())

# Arrests per month in 2021
arrests_2021 <- police_arrests %>%
  filter(year == 2021)

monthly_arrests_2021 <- arrests_2021 %>%
  group_by(arrest_month) %>%
  summarise(count = n())

########################################################################
# Number of movie releases per month compared to number of arrests per month

# 2018
# Combine the two data sets by month
combined_2018 <- monthly_releases_2018 %>%
  rename(month = release_month, releases = count) %>%
  inner_join(
    monthly_arrests_2018 %>% rename(month = arrest_month, arrests = count),
    by = "month") %>%
  pivot_longer(cols = c(releases, arrests), names_to = "type",
               values_to = "count")
```

```r
# Overlay the plots
ggplot(combined_2018, aes(x = month)) +
  # Bars only for releases
  geom_col(data = combined_2018 %>% filter(type == "releases"),
           aes(y = count, fill = type), color = "black", alpha = 0.7) +
  # Line + points only for arrests
  geom_line(data = combined_2018 %>% filter(type == "arrests"),
            aes(y = count, color = type), size = 1.2) +
  geom_point(data = combined_2018 %>% filter(type == "arrests"),
             aes(y = count, color = type), size = 2) +
  labs(title = "Releases and Arrests per Month (2018)", x = "Month",
       y = "Count", fill = "Legend", color = "Legend") +
  scale_x_continuous(breaks = 1:12,
                     labels = c("Jan", "Feb", "Mar", "Apr", "May", "Jun",
                                "Jul", "Aug", "Sep", "Oct", "Nov",
                                "Dec")) +
  scale_fill_manual(values = c("releases" = "red")) +
  scale_color_manual(values = c("arrests" = "firebrick")) +
  theme_minimal()
```

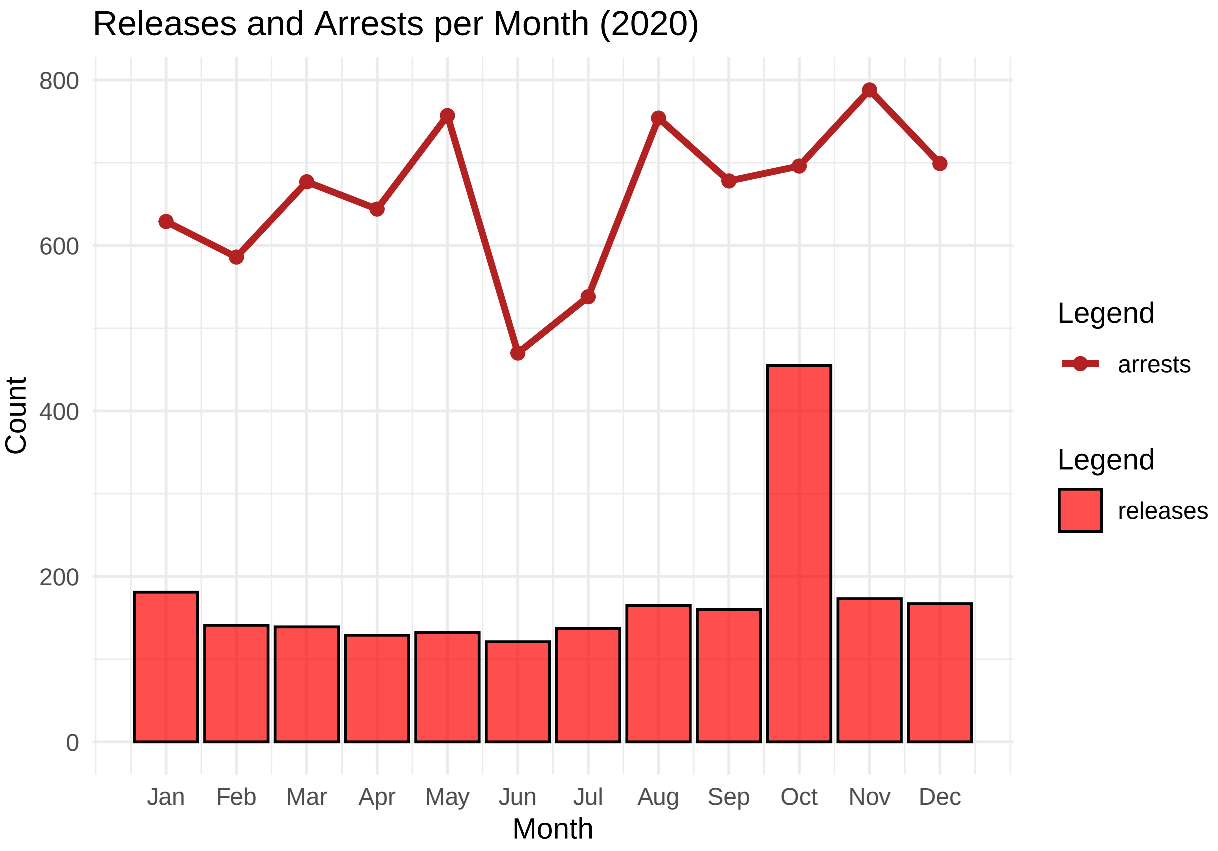
```
## Warning: Using ‘size‘ aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use ‘linewidth‘ instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```



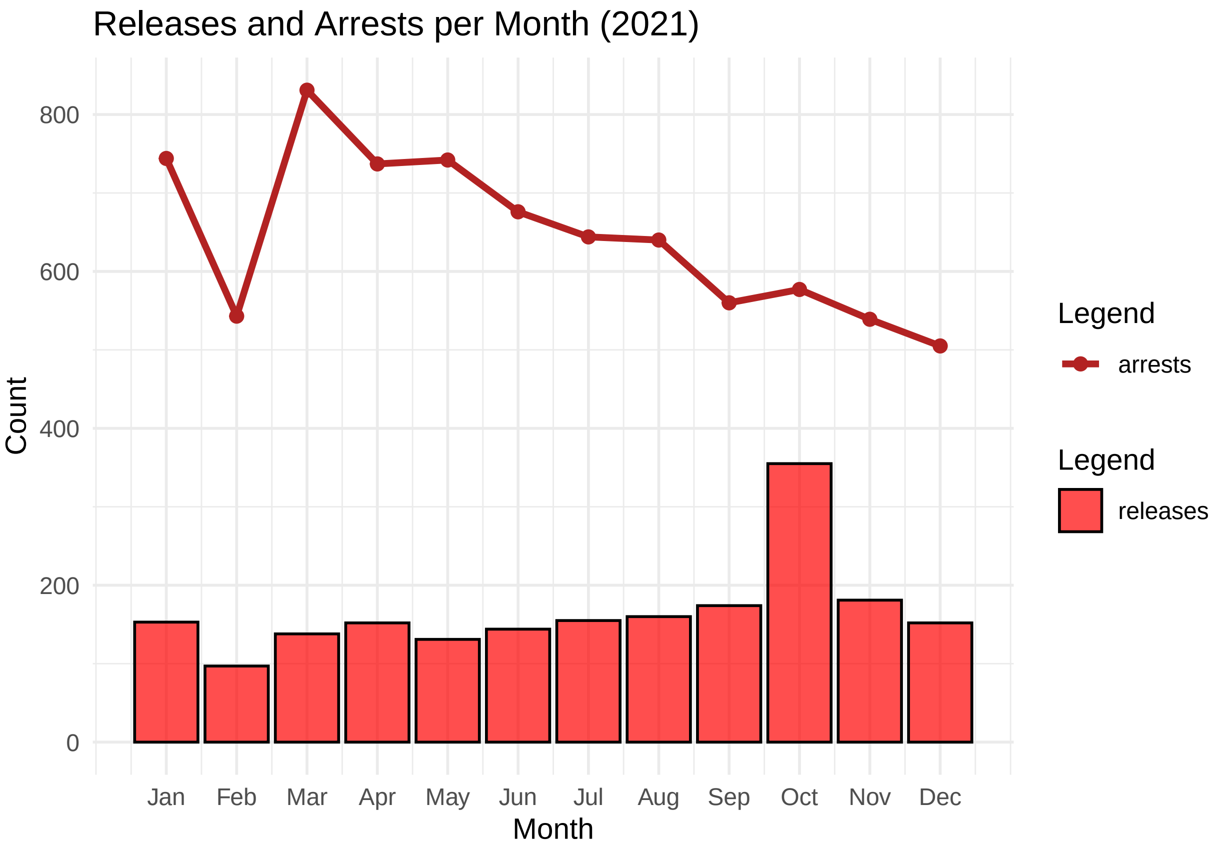
```r
# 2019
combined_2019 <- monthly_releases_2019 %>%
  rename(month = release_month, releases = count) %>%
  inner_join(
    monthly_arrests_2019 %>% rename(month = arrest_month, arrests = count),
    by = "month") %>%
  pivot_longer(cols = c(releases, arrests), names_to = "type",
               values_to = "count")

ggplot(combined_2019, aes(x = month)) +
  geom_col(data = combined_2019 %>% filter(type == "releases"),
           aes(y = count, fill = type), color = "black", alpha = 0.7) +
  geom_line(data = combined_2019 %>% filter(type == "arrests"),
            aes(y = count, color = type), size = 1.2) +
  geom_point(data = combined_2019 %>% filter(type == "arrests"),
             aes(y = count, color = type), size = 2) +
  labs(title = "Releases and Arrests per Month (2019)", x = "Month",
       y = "Count", fill = "Legend", color = "Legend") +
  scale_x_continuous(breaks = 1:12,
                     labels = c("Jan", "Feb", "Mar", "Apr", "May", "Jun",
                                "Jul", "Aug", "Sep", "Oct", "Nov",
                                "Dec")) +
  scale_fill_manual(values = c("releases" = "red")) +
  scale_color_manual(values = c("arrests" = "firebrick")) +
  theme_minimal()
```



```r
# 2020
combined_2020 <- monthly_releases_2020 %>%
  rename(month = release_month, releases = count) %>%
  inner_join(
    monthly_arrests_2020 %>% rename(month = arrest_month, arrests = count),
    by = "month") %>%
  pivot_longer(cols = c(releases, arrests), names_to = "type",
               values_to = "count")

ggplot(combined_2020, aes(x = month)) +
  geom_col(data = combined_2020 %>% filter(type == "releases"),
           aes(y = count, fill = type), color = "black", alpha = 0.7) +
  geom_line(data = combined_2020 %>% filter(type == "arrests"),
            aes(y = count, color = type), size = 1.2) +
  geom_point(data = combined_2020 %>% filter(type == "arrests"),
             aes(y = count, color = type), size = 2) +
  labs(title = "Releases and Arrests per Month (2020)", x = "Month",
       y = "Count", fill = "Legend", color = "Legend") +
  scale_x_continuous(breaks = 1:12,
                     labels = c("Jan", "Feb", "Mar", "Apr", "May", "Jun",
                                "Jul", "Aug", "Sep", "Oct", "Nov",
                                "Dec")) +
  scale_fill_manual(values = c("releases" = "red")) +
  scale_color_manual(values = c("arrests" = "firebrick")) +
  theme_minimal()
```



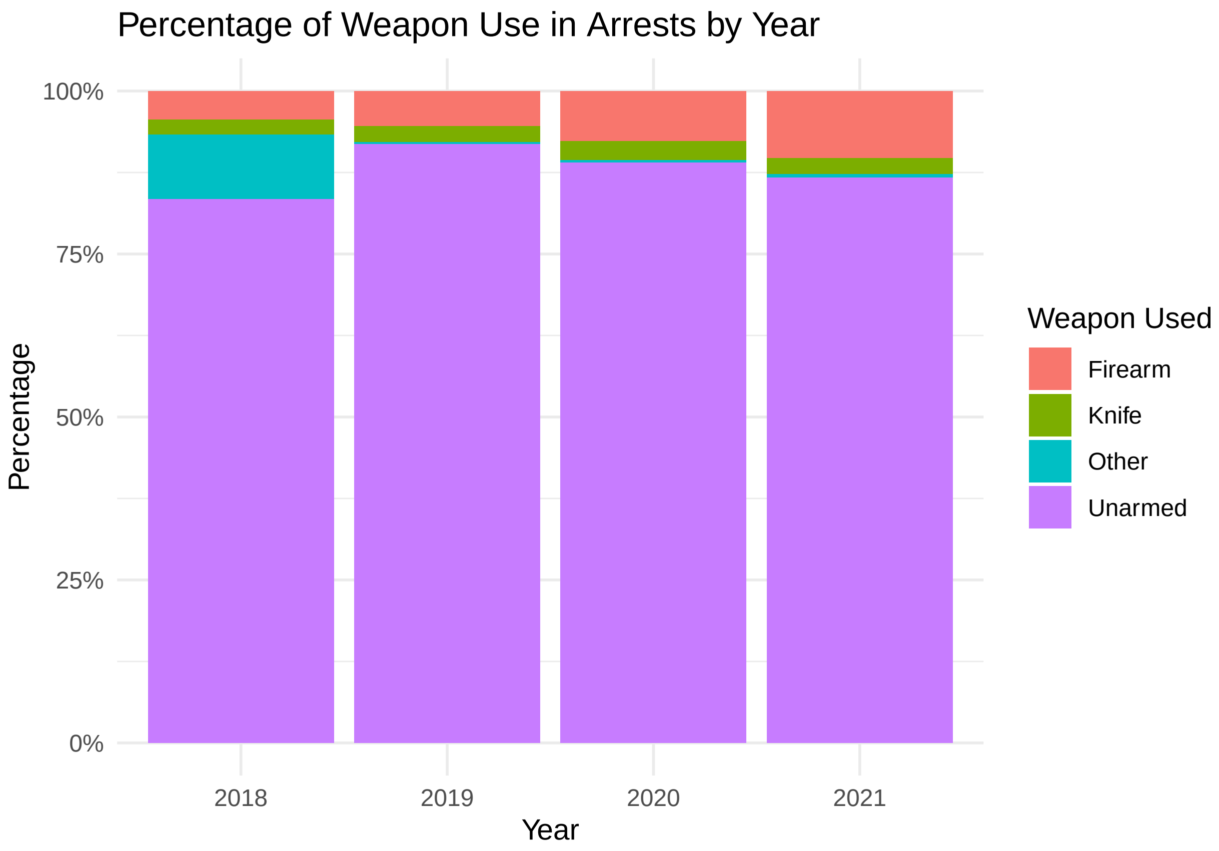
```r
# 2021
combined_2021 <- monthly_releases_2021 %>%
  rename(month = release_month, releases = count) %>%
  inner_join(
    monthly_arrests_2021 %>% rename(month = arrest_month, arrests = count),
    by = "month") %>%
  pivot_longer(cols = c(releases, arrests), names_to = "type",
               values_to = "count")

ggplot(combined_2021, aes(x = month)) +
  geom_col(data = combined_2021 %>% filter(type == "releases"),
           aes(y = count, fill = type), color = "black", alpha = 0.7) +
  geom_line(data = combined_2021 %>% filter(type == "arrests"),
            aes(y = count, color = type), size = 1.2) +
  geom_point(data = combined_2021 %>% filter(type == "arrests"),
             aes(y = count, color = type), size = 2) +
  labs(title = "Releases and Arrests per Month (2021)", x = "Month",
       y = "Count", fill = "Legend", color = "Legend") +
  scale_x_continuous(breaks = 1:12,
                     labels = c("Jan", "Feb", "Mar", "Apr", "May", "Jun",
                                "Jul", "Aug", "Sep", "Oct", "Nov",
                                "Dec")) +
  scale_fill_manual(values = c("releases" = "red")) +
  scale_color_manual(values = c("arrests" = "firebrick")) +
  theme_minimal()
```



```r
##########################################################################
# Read in Weapon data set
Weapon_Data <- read.csv("C:/Users/adeel/Downloads/Weapon_Data (1).csv")

# Remove empty rows
weapon_simplified <- na.omit(Weapon_Data)

# Mutate the Arrest Weapon column and organize into categories
police_arrests <- police_arrests %>%
  mutate(Arrest.Weapon = case_when(
  Arrest.Weapon %in% c("33", "Firearm (Type Not Stated)", "Gun", "Handgun",
                       "Other Firearm", "Shotgun") ~ "Firearm",
  Arrest.Weapon %in% c("Knife - Buthcer", "Knife - Other",
                       "Knife - Pocket", "Missle/Arrow",
                       "Stabbing Instrument") ~ "Knife",
  Arrest.Weapon %in% c("Unarmed") ~ "Unarmed",
  TRUE ~ "Other"))

# Create a data set to store percentage of weapon usage in crimes per year
percentage_data <- police_arrests %>%
      group_by(year, Arrest.Weapon) %>%
      summarise(Count = n()) %>%
      mutate(Percentage = Count / sum(Count))
```

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

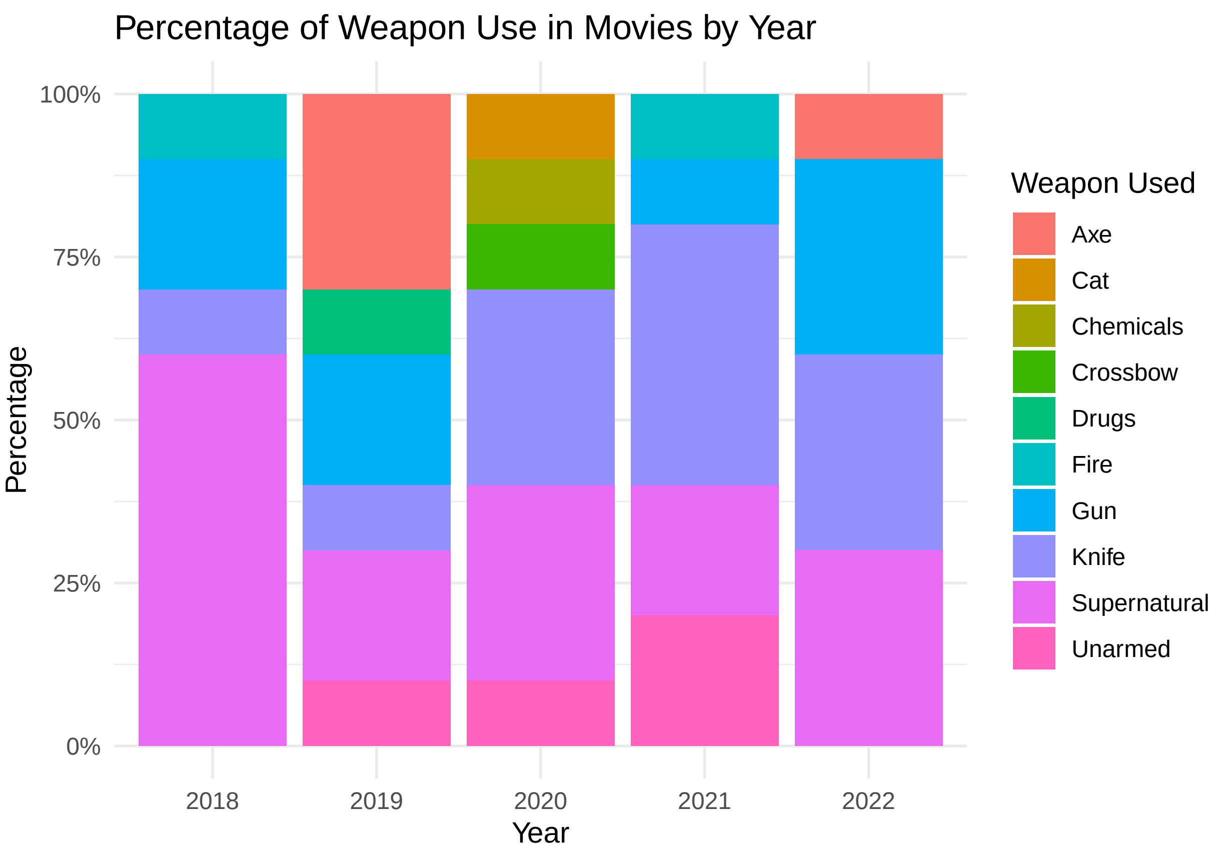
```r
# Plot the percentage of weapons used in crimes per year
ggplot(percentage_data, aes(x = factor(year), y = Percentage,
                            fill = Arrest.Weapon)) +
      geom_col(position = "fill") +
      scale_y_continuous(labels = scales::percent) +
      labs(
        title = "Percentage of Weapon Use in Arrests by Year",
        x = "Year",
        y = "Percentage",
        fill = "Weapon Used"
      ) +
      theme_minimal()
```



```r
# Create a data set to store the percentage of weapons used in movies
percent <- weapon_simplified %>%
  group_by(Year, Weapon) %>%
  summarise(Count = n()) %>%
  mutate(Percentage = Count / sum(Count))
```

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

```r
# Plot the percentage of weapons used in movies per year
ggplot(percent, aes(x = factor(Year), y = Percentage,
                              fill = Weapon)) +
      geom_col(position = "fill") +
      scale_y_continuous(labels = scales::percent) +
      labs(
        title = "Percentage of Weapon Use in Movies by Year",
        x = "Year",
        y = "Percentage",
        fill = "Weapon Used"
      ) +
      theme_minimal()
```



```r
########################################################################
# Linear Regression Model for crime VS movie releases per month per year

# Change the year and month columns to match across the data sets
horror_data <- horror_data %>%
  mutate(
    year = year(release_date),
    month = month(release_date)
  )

police_arrests <- police_arrests %>%
  rename(month = arrest_month)

# Movies by year-month
movies_by_month <- horror_data %>%
  group_by(year, month) %>%
  summarize(num_movies = n())
```

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

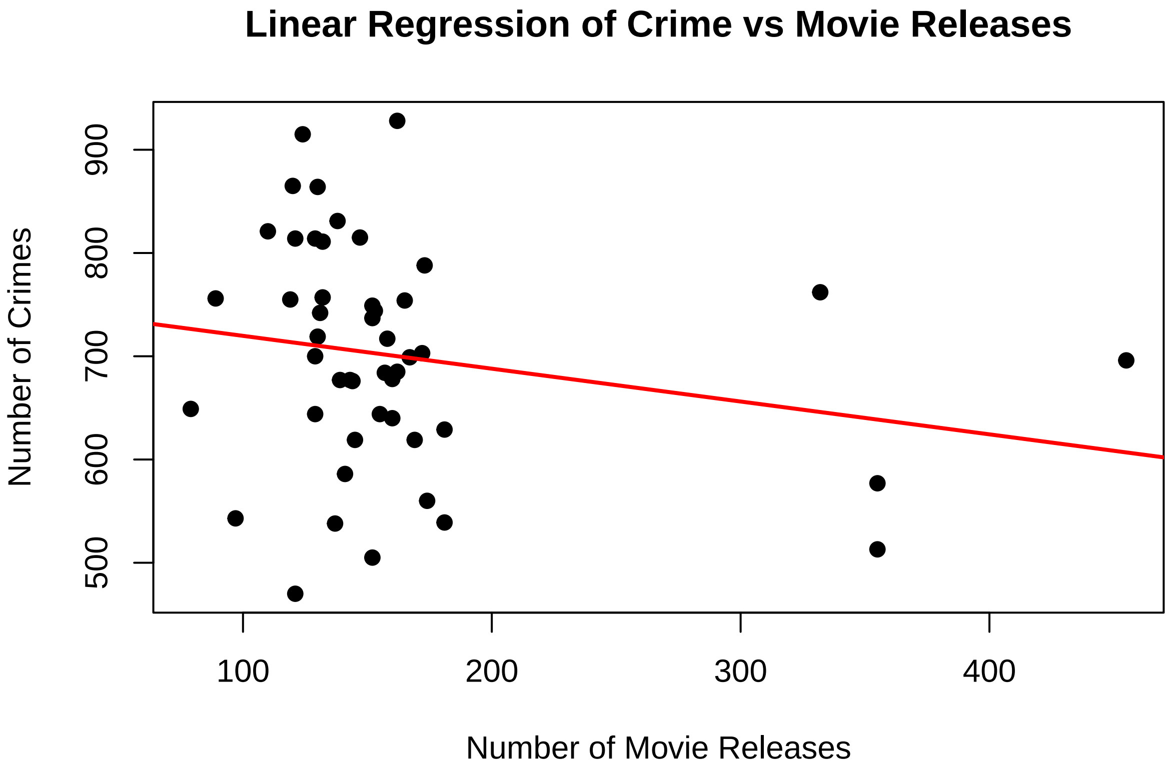
```r
# Crimes by year-month
crimes_by_month <- police_arrests %>%
  group_by(year, month) %>%
  summarize(num_crimes = n())
```

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

```r
# Join the data sets
merged_data <- inner_join(
  movies_by_month,
  crimes_by_month,
  by = c("year", "month")
)

# Plot the regression
model <- lm(num_crimes ~ num_movies, data = merged_data)
summary(model)
```

```
##
## Call:
## lm(formula = num_crimes ~ num_movies, data = merged_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -243.086  -65.389    3.338   89.709  227.953
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 751.5644    39.2910  19.128   <2e-16 ***
## num_movies   -0.3180     0.2235  -1.423    0.161
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 107.2 on 46 degrees of freedom
## Multiple R-squared:  0.04216,    Adjusted R-squared:  0.02134
## F-statistic: 2.025 on 1 and 46 DF,  p-value: 0.1615
```

```r
plot(merged_data$num_movies, merged_data$num_crimes,
     main = "Linear Regression of Crime vs Movie Releases",
     xlab = "Number of Movie Releases",
     ylab = "Number of Crimes",
     pch = 19)

abline(model, col = "red", lwd = 2)
```


Linear Regression of Crime vs Movie Releases
Number of Crimes
500
600
700
800
900
100
200
300
400
Number of Movie Releases