

Assessing the Rates of Crimes Against the Person in Toronto

Sofia Sellitto

2/3/23

Abstract

This data was free for use, courtesy of Gelfand (2022). We are using R Core Team (2022) for analyzing this data.

Loading Packages and Data

We import tidyverse by Wickham et al. (2019) for convenience because it contains many other packages which will be useful for plotting and data manipulations. We also use Firke (2021) for cleaning and Wickham et al. (2022) for manipulating data.

```
## loading appropriate packages
```

```
library(tidyverse)
```

```
-- Attaching packages ----- tidyverse 1.3.2 --
v ggplot2 3.4.0      v purrr  1.0.1
v tibble  3.1.8      v dplyr  1.0.10
v tidyr   1.2.1      v stringr 1.5.0
v readr   2.1.3      v forcats 0.5.2
-- Conflicts ----- tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
```

```
library(ggplot2)
library(knitr)
library(janitor)
```

Attaching package: 'janitor'

The following objects are masked from 'package:stats':

```
chisq.test, fisher.test
```

```
library(dplyr)
```

```
cleaned_data_dropped <- read.csv("~/Inputs/Tutorial 1/Scripts/cleaned_data.csv")
head(cleaned_data_dropped)
```

	reported_year	subtype	sex	count
1	2014	Assault	M	2
2	2014	Assault	M	1
3	2014	Assault	F	1
4	2014	Assault	F	5
5	2014	Assault	F	3
6	2014	Assault	F	1

```
## summing count of crimes and making them its own group
```

```
cleaned_data_dropped_grouped <- cleaned_data_dropped |>
  group_by(subtype, reported_year) |>
  summarise(across(count, sum))
```

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

```
cleaned_data_dropped_grouped
```

```
# A tibble: 32 x 3
# Groups:   subtype [4]
```

```

  subtype reported_year count
  <chr>      <int> <int>
1 Assault      2014 15141
2 Assault      2015 15900
3 Assault      2016 16563
4 Assault      2017 17126
5 Assault      2018 17834
6 Assault      2019 18886
7 Assault      2020 16439
8 Assault      2021 17325
9 Other        2014  4729
10 Other       2015  5613
# ... with 22 more rows

```

```
## calculating the mean number of crimes for each subcategory across 2014-2021
```

```

mean_data <- cleaned_data_dropped_grouped |>
  summarise(mean_count = mean(count))

```

```
mean_data
```

```

# A tibble: 4 x 2
  subtype      mean_count
  <chr>          <dbl>
1 Assault      16902.
2 Other         4533.
3 Robbery       3062.
4 Sexual Violation 2383.

```

```

knitr::kable(head(cleaned_data_dropped_grouped), "simple", caption = "The first 6 rows of
  col.names=c("Sub-Category of Crime", "Year", "Number of Victims"),
  align=c('l', 'c', 'c'))

```

Table 1: The first 6 rows of the data frame used to represent the number of victims according to category of crime in Toronto Across 2014 - 2021

Sub-Category of Crime	Year	Number of Victims
Assault	2014	15141
Assault	2015	15900

Sub-Category of Crime	Year	Number of Victims
Assault	2016	16563
Assault	2017	17126
Assault	2018	17834
Assault	2019	18886

```
## Creating a table to visually to display the mean number of crimes for each subcategory

knitr::kable(mean_data,"simple", caption = "Average Number of victims of crimes in Toronto",
              col.names=c("Category of Crime", "Average Number of Victims"),
              align=c('l', 'c'))
```

Table 2: Average Number of victims of crimes in Toronto across 2014-2021, grouped by category of crime

Category of Crime	Average Number of Victims
Assault	16901.750
Other	4532.750
Robbery	3062.500
Sexual Violation	2382.625

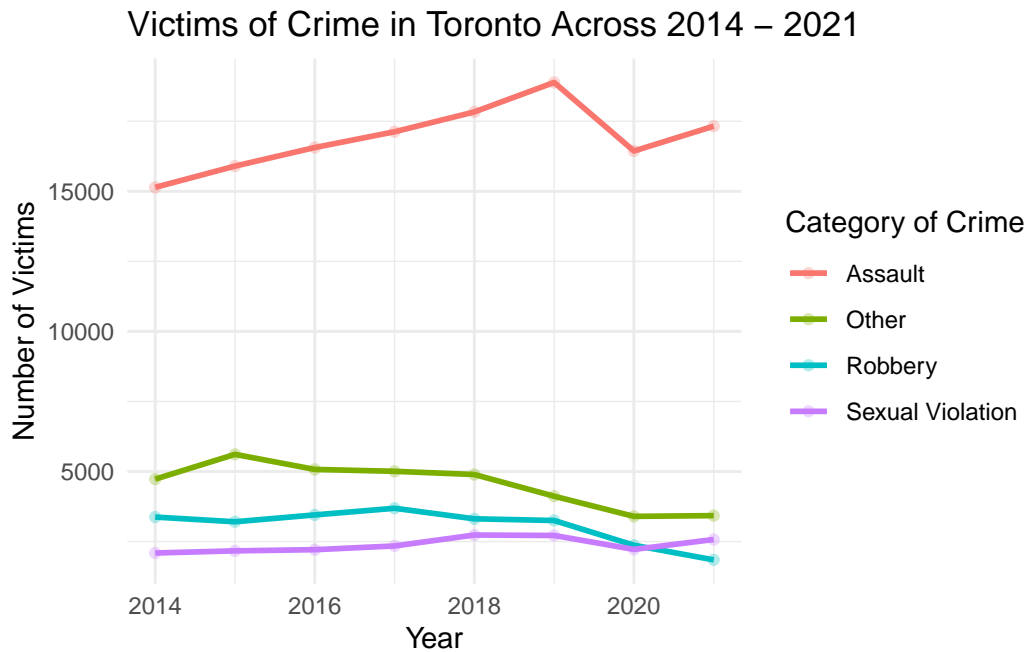
See (figure1?)

```
## creating a line graph that includes the number of victims according to category of crime

#| fig-cap: "Number of victims according to category of crime in Toronto Across 2014 - 2021"
#| label: figure1

cleaned_data_dropped_grouped |>
  ggplot(aes(x = reported_year, y = count, colour = subtype)) +
  geom_point(linewidth = 5, alpha = 0.3)+
  geom_line(linewidth = 1)+
  theme_minimal()+
  labs(
    title = "Victims of Crime in Toronto Across 2014 - 2021",
    x = "Year",
    y = "Number of Victims",
    colour = "Category of Crime" )
```

Warning in `geom_point(linewidth = 5, alpha = 0.3)`: Ignoring unknown parameters: ``linewidth``



making a data frame that includes the variable sex so that I can call on it for a graph

```
sex_data <- cleaned_data_dropped |>
  group_by(subtype, sex, reported_year) |>
  summarise(across(count, sum))
```

``summarise()`` has grouped output by 'subtype', 'sex'. You can override using the ``.groups`` argument.

```
sex_data
```

```
# A tibble: 92 x 4
# Groups:   subtype, sex [12]
  subtype sex  reported_year count
  <chr>   <chr>          <int> <int>
1 Assault F           2014  7290
2 Assault F           2015  7502
```

```

3 Assault F          2016  7714
4 Assault F          2017  7880
5 Assault F          2018  8221
6 Assault F          2019  8679
7 Assault F          2020  7755
8 Assault F          2021  8097
9 Assault M          2014  7851
10 Assault M         2015  8395
# ... with 82 more rows

```

See (figure2?)

```

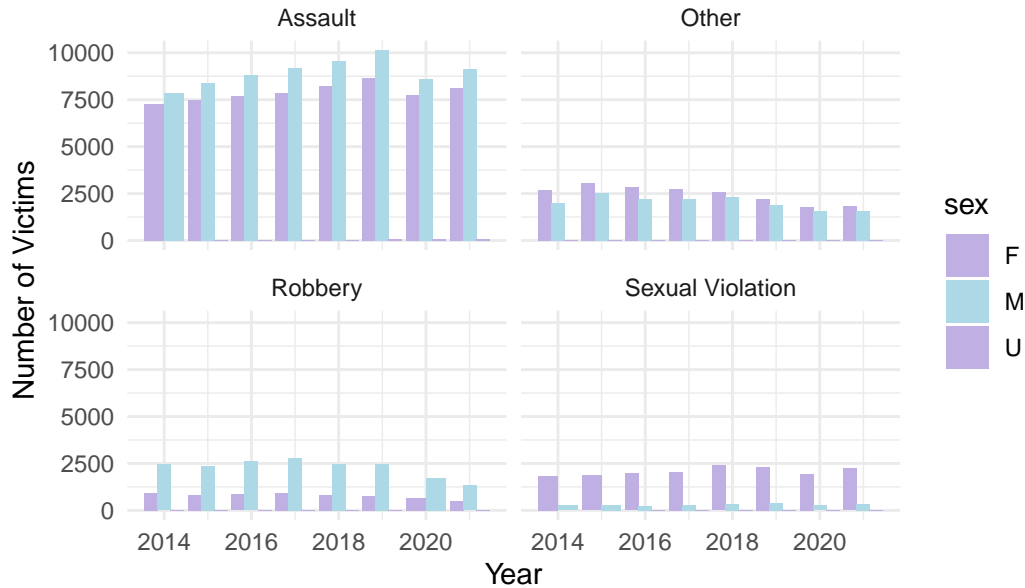
## calling on my data frame to create a bar plot of the sex of the victims across time

#| fig-cap: "How the victims of crimes varries across sex in Toronto from 2014-2021"
#| label: figure2

ggplot(sex_data, aes
  (x =reported_year,
    y =count,
    fill =sex))+
  geom_bar(position="dodge", stat="identity")+
  facet_wrap(~subtype)+
  scale_fill_manual(values=c("#COAFE2", "#ADD8E6", "#COAFE2"))+
  theme_minimal()+
  labs(
    title = "Victims of Crime in Toronto Categorized by Sex Across 2014 - 2021",
    x = "Year",
    y = "Number of Victims",
    legend = "Sex")

```

Victims of Crime in Toronto Categorized by Sex Across 2014



References

- Firke, Sam. 2021. *Janitor: Simple Tools for Examining and Cleaning Dirty Data*. <https://CRAN.R-project.org/package=janitor>.
- Gelfand, Sharla. 2022. *Opendatatoronto: Access the City of Toronto Open Data Portal*. <https://CRAN.R-project.org/package=opendatatoronto>.
- R Core Team. 2022. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. <https://www.R-project.org/>.
- Wickham, Hadley, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino McGowan, Romain François, Garrett Golemund, et al. 2019. "Welcome to the tidyverse." *Journal of Open Source Software* 4 (43): 1686. <https://doi.org/10.21105/joss.01686>.
- Wickham, Hadley, Romain François, Lionel Henry, and Kirill Müller. 2022. *Dplyr: A Grammar of Data Manipulation*. <https://CRAN.R-project.org/package=dplyr>.