

Analyzing the Number of Toronto Marriage Licenses in 2023

Tam Ly

```
#### Preamble ####  
# Purpose: Obtain and prepare data on the number of Toronto marriage  
# licenses in 2023.  
# Author: Tam Ly  
# Email: annatn.ly@mail.utoronto.ca  
# Date: 16 Jan 24  
# Prerequisites: Need to know where to get Toronto marriage license  
# data.
```

We will use the R programming language (R Core Team 2021), and load the packages tidyverse (Wickham et al. 2019), janitor (Firke 2023), and opendatatoronto (Gelfand 2022). We will follow Chapter 2 and 3 of *Telling Stories with Data* (Alexander 2023).

```
#### Workplace setup ####  
install.packages("tidyverse")
```

Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.3'
(as 'lib' is unspecified)

```
install.packages("janitor")
```

Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.3'
(as 'lib' is unspecified)

```
install.packages("opendatatoronto")
```

Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.3'
(as 'lib' is unspecified)

```
library(tidyverse)
```

```
-- Attaching core tidyverse packages ----- tidyverse 2.0.0 --  
v dplyr      1.1.4      v readr      2.1.5  
v forcats    1.0.0      v stringr    1.5.1  
v ggplot2     3.4.4      v tibble     3.2.1  
v lubridate  1.9.3      v tidyr      1.3.0  
v purrr       1.0.2
```

```
-- Conflicts ----- tidyverse_conflicts() --  
x dplyr::filter() masks stats::filter()  
x dplyr::lag()     masks stats::lag()  
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become
```

```
library(janitor)
```

Attaching package: 'janitor'

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

chisq.test, fisher.test

```
library(opendatatoronto)
```

Plan

```
#### Plan ####
```

Simulate

```
#### Simulate ####
```

Acquire

```
#### Acquire ####
# Read in the data
toronto_marriages <-
  list_package_resources("e28bc818-43d5-43f7-b5d9-bdfb4eda5feb") |>
  filter(name == "Marriage Licence Statistics Data.csv") |>
  get_resource()

# Save the data
write_csv(
  x = toronto_marriages,
  file = "toronto_marriages.csv"
)

head(toronto_marriages)
```

```
# A tibble: 6 x 4
  X_id CIVIC_CENTRE MARRIAGE_LICENSES TIME_PERIOD
<int> <chr>          <int> <chr>
1 11101 ET              80 2011-01
2 11102 NY             136 2011-01
3 11103 SC             159 2011-01
4 11104 TO             367 2011-01
5 11105 ET             109 2011-02
6 11106 NY             150 2011-02
```

```
# Keep rows for year 2023 only
# Keep columns for month and number of marriage licenses
cleaned_toronto_marriages <-
  clean_names(toronto_marriages) |>
  slice(505:528) |>
  select(marriage_licenses, time_period)
```

```
head(cleaned_toronto_marriages)
```

```
# A tibble: 6 x 2
  marriage_licenses time_period
      <int> <chr>
1         149 2023-01
2         563 2023-01
3         156 2023-02
4         617 2023-02
5         215 2023-03
6         929 2023-03
```

```
# Change format of months from digits to name
```

```
cleaned_toronto_marriages <-
  cleaned_toronto_marriages |>
  mutate(
    time_period =
      case_match(
        time_period,
        "2023-01" ~ "January",
        "2023-02" ~ "February",
        "2023-03" ~ "March",
        "2023-04" ~ "April",
        "2023-05" ~ "May",
        "2023-06" ~ "June",
        "2023-07" ~ "July",
        "2023-08" ~ "August",
        "2023-09" ~ "September",
        "2023-10" ~ "October",
        "2023-11" ~ "November",
        "2023-12" ~ "December",
      )
  )
```

```
head(cleaned_toronto_marriages)
```

```
# A tibble: 6 x 2
  marriage_licenses time_period
      <int> <chr>
1         149 January
```

```

2           563 January
3           156 February
4           617 February
5           215 March
6           929 March

```

```

# Save the cleaned file
write_csv(
  x = cleaned_toronto_marriages,
  file = "cleaned_toronto_marriages.csv"
)

```

Explore

```

#### Explore ####
# Read in the cleaned data
cleaned_toronto_marriages <-
  read_csv(
    file = "cleaned_toronto_marriages.csv",
    show_col_types = FALSE
  )

# Sum up the number of marriages per month
cleaned_toronto_marriages <-
  cleaned_toronto_marriages |>
  group_by(time_period) |>
  summarize(number_marriages = sum(marriage_licenses))

head(cleaned_toronto_marriages)

```

```

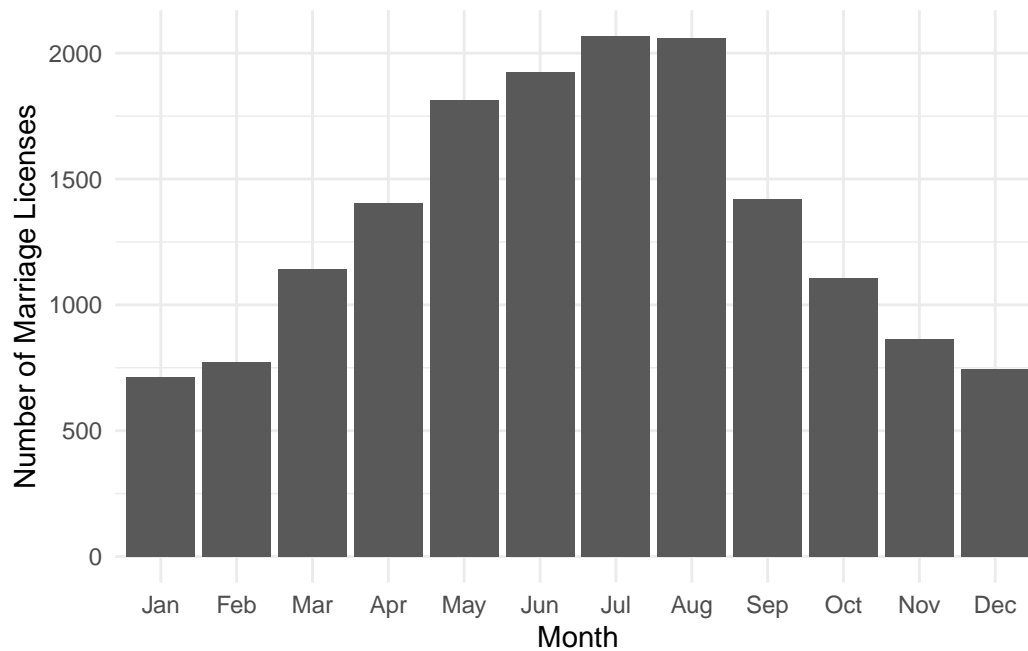
# A tibble: 6 x 2
  time_period number_marriages
  <chr>         <dbl>
1 April         1404
2 August        2061
3 December       746
4 February       773
5 January        712
6 July          2067

```

```
# Arrange the months in chronological order
cleaned_toronto_marriages |>
  mutate(month = factor(time_period, levels = month.name)) |>
  arrange(month) |>
  select(month, number_marriages)
```

```
# A tibble: 12 x 2
  month      number_marriages
  <fct>          <dbl>
1 January           712
2 February          773
3 March            1144
4 April             1404
5 May              1812
6 June             1923
7 July             2067
8 August            2061
9 September        1422
10 October         1107
11 November         866
12 December         746
```

```
# Build a bar graph
cleaned_toronto_marriages |>
  mutate(month = factor(time_period, levels = month.name, labels
= month.abb)) |>
  ggplot(aes(x = month, y = number_marriages)) +
  geom_col() +
  theme_minimal() +
  labs(x = "Month", y = "Number of Marriage Licenses")
```



Share

Share

References

- Alexander, Rohan. 2023. *Telling Stories with Data*. Chapman; Hall/CRC. <https://tellingstorieswithdata.com>.
- Firke, Sam. 2023. *Janitor: Simple Tools for Examining and Cleaning Dirty Data*. <https://github.com/sfirke/janitor>.
- Gelfand, Sharla. 2022. *Opendatatoronto: Access the City of Toronto Open Data Portal*. <https://sharlagelfand.github.io/opendatatoronto/>.
- R Core Team. 2021. *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>.