

Code last run 2021-02-16.

Daily: Data as of January 28, 2021.

Neighbourhood: Data as of January 28, 2021.

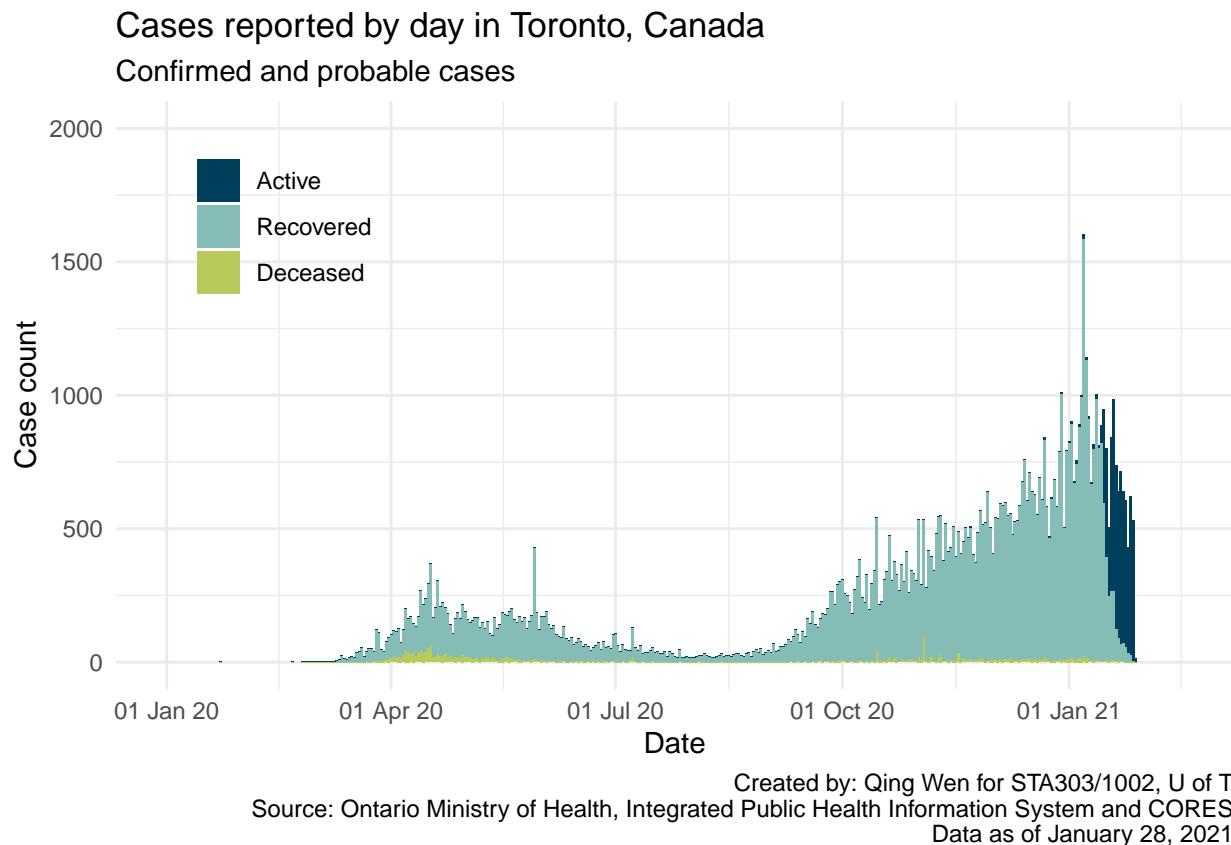
Task 1: Daily cases

Data wrangling

```
# to replace all NA values with 0 in the recovered, active, and deceased columns
# make sure the reported_date column is in date format
# tidy the data since variable does not have its own column
# change the value for status to sentence-case, and reorder its level
reported <- reported_raw %>%
  mutate_if(is.numeric, replace_na, replace = 0) %>%
  mutate(reported_date = date(reported_date)) %>%
  pivot_longer(-reported_date, names_to = "status", values_to = "total") %>%
  mutate(status = str_to_sentence(status)) %>%
  mutate(status = fct_relevel(status, "Active", after = 0)) %>%
  mutate(status = fct_relevel(status, "Recovered", after = 1))
```

Data visualization

```
reported %>%
  ggplot(aes(x = reported_date, y = total, fill = status)) +
  geom_bar(stat = "identity", width = 1) +
  scale_x_date(limits = c(date("2020-01-01"), Sys.Date()), date_labels = "%d %b %y") +
  scale_y_continuous(limits = c(0, 2000)) +
  theme_minimal() +
  labs(
    title = "Cases reported by day in Toronto, Canada",
    subtitle = "Confirmed and probable cases",
    x = "Date",
    y = "Case count",
    caption = str_c("Created by: Qing Wen for STA303/1002, U of T\nSource: Ontario Ministry of Health, Integrated Public Health Information System and CORES\nData as of January 28, 2021")
  ) +
  theme(legend.title = element_blank(), legend.position = c(0.15, 0.8)) +
  scale_fill_manual(values = c("#003F5C", "#86BCB6", "#B9CA5D"))
```



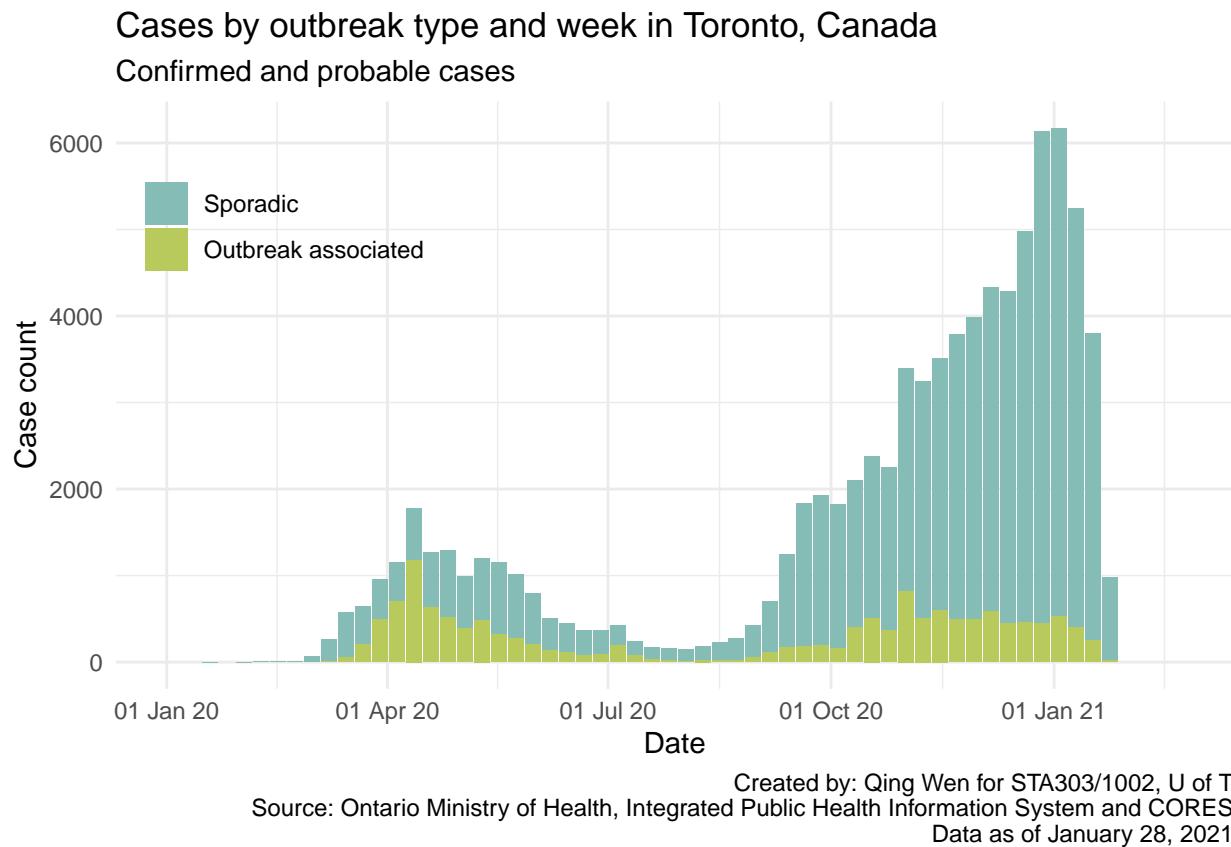
Task 2: Outbreak type

Data wrangling

```
outbreak <- outbreak_raw %>%
  mutate(episode_week = date(episode_week)) %>%
  mutate(outbreak_or_sporadic = str_replace_all(outbreak_or_sporadic, "OB Associated",
                                                "Outbreak associated")) %>%
  mutate(outbreak_or_sporadic = fct_relevel(outbreak_or_sporadic, "Sporadic", after = 0)) %>%
  group_by(episode_week) %>%
  mutate(total_cases = sum(cases))
```

Data visualization

```
outbreak %>%
  ggplot(aes(x = episode_week, y = cases, fill = outbreak_or_sporadic)) +
  geom_bar(stat = "identity") +
  scale_x_date(labels = scales::date_format("%d %b %y"), limits = c(date("2020-01-01"),
  Sys.Date() + 7)) +
  scale_y_continuous(limits = c(0, max(outbreak$total_cases))) +
  theme_minimal() +
  labs(
    title = "Cases by outbreak type and week in Toronto, Canada",
    subtitle = "Confirmed and probable cases",
    x = "Date",
    y = "Case count",
    caption = str_c("Created by: Qing Wen for STA303/1002, U of T\nSource: Ontario Ministry of Health,",
      date_daily[1,1])
  ) +
  theme(legend.title = element_blank(), legend.position = c(0.15, 0.8)) +
  scale_fill_manual(values = c("#86BCB6", "#B9CA5D"))
```



Task 3: Neighbourhoods

Data wrangling: part 1

```
# find the relevant observation that records the percentage of 18-64 years-olds who are
# classified as low-income
# tidy the data to make sure each variable has its own column
# stores the percentages as numbers instead of character strings
income <- nbhood_profile %>%
  clean_names() %>%
  filter(id == "1143") %>%
  pivot_longer(-c(id, category, topic, data_source, characteristic),
              names_to = "neighbourhood_name", values_to = "percentage") %>%
  mutate(percentage = parse_number(percentage))
```

Data wrangling: part 2

```
income <- income %>%
  mutate(neighbourhood_name = ifelse(neighbourhood_name == "weston_pelham_park",
                                       "weston_pellam_park", neighbourhood_name))

nbhood_raw_1 <- nbhood_raw %>%
  mutate(neighbourhood_name = ifelse(neighbourhood_name == "Weston-Pelham Park",
                                       "Weston-Pellam Park", neighbourhood_name)) %>%
  mutate(neighbourhood_name = str_to_lower(str_replace_all(neighbourhood_name,
                                                          "[[:punct:]]", " "))) %>%
  mutate(neighbourhood_name = str_to_lower(str_replace_all(neighbourhood_name,
                                                          "\\\s\\s", " "))) %>%
  mutate(neighbourhood_name = str_replace_all(neighbourhood_name, "\\s$", "")) %>%
  mutate(neighbourhood_name = str_replace_all(neighbourhood_name, "\\s", "_"))

nbhoods_all <- nbhoods_shape_raw %>%
  clean_names() %>%
  mutate(neighbourhood_name = str_remove(area_name, "\\s\\\\(\\d+\\\\)$")) %>%
  mutate(neighbourhood_name = str_to_lower(str_replace_all(neighbourhood_name,
                                                          "[[:punct:]]", " "))) %>%
  mutate(neighbourhood_name = str_replace_all(neighbourhood_name, "\\s\\s", " ")) %>%
  mutate(neighbourhood_name = str_replace_all(neighbourhood_name, "\\s$", "")) %>%
  mutate(neighbourhood_name = str_replace_all(neighbourhood_name, "\\s", "_"))
  left_join(nbhood_raw_1, by = "neighbourhood_name") %>%
  left_join(income, by = "neighbourhood_name") %>%
  rename(rate_per_100000 = rate_per_100_000_people)
```

Data wrangling: part 3

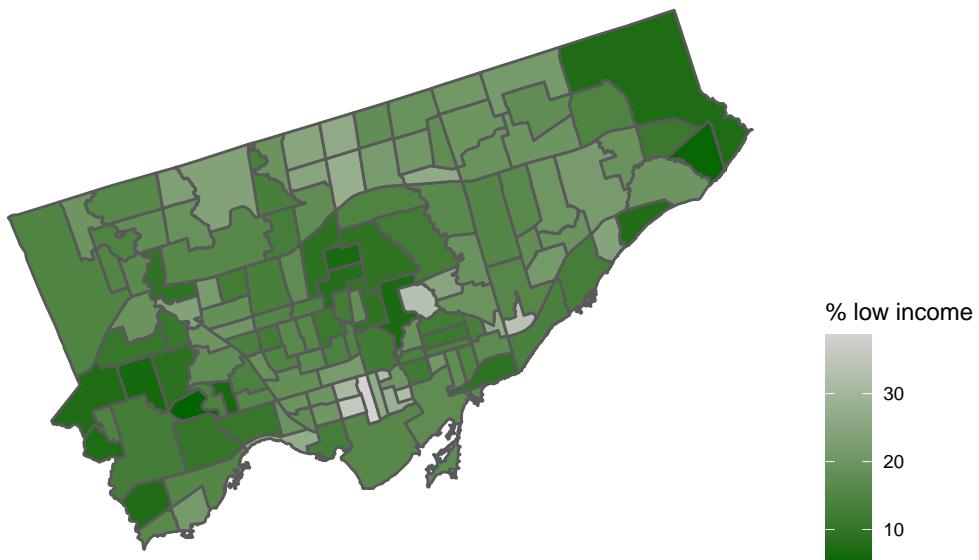
```
nbhoods_final <- nbhoods_all %>%
  mutate(med_inc = median(percentage)) %>%
  mutate(med_rate = median(rate_per_100000)) %>%
  mutate(nbhood_type = case_when(
    percentage >= med_inc & rate_per_100000 >= med_rate ~
      "Higher low income rate, higher case rate",
    percentage >= med_inc & rate_per_100000 < med_rate ~
```

```
"Higher low income rate, lower case rate",
percentage < med_inc & rate_per_100000 >= med_rate ~
"Lower low income rate, higher case rate",
percentage < med_inc & rate_per_100000 < med_rate ~
"Lower low income rate, lower case rate"
))
```

Data visualization

```
ggplot(data = nbhoods_final) +  
  geom_sf(aes(fill = percentage)) +  
  labs(  
    title = "Percentage of 18 to 64 year olds living in a low income family (2015)",  
    subtitle = "Neighbourhoods of Toronto, Canada",  
    caption = str_c("Created by: Qing Wen for STA303/1002, U of T\nSource: Census Profile 98-316-X2016001",  
                  date_daily[1, 1]))  
  ) +  
  theme_map() +  
  theme(legend.position = "right") +  
  scale_fill_gradient(name = "% low income", low = "darkgreen", high = "lightgrey")
```

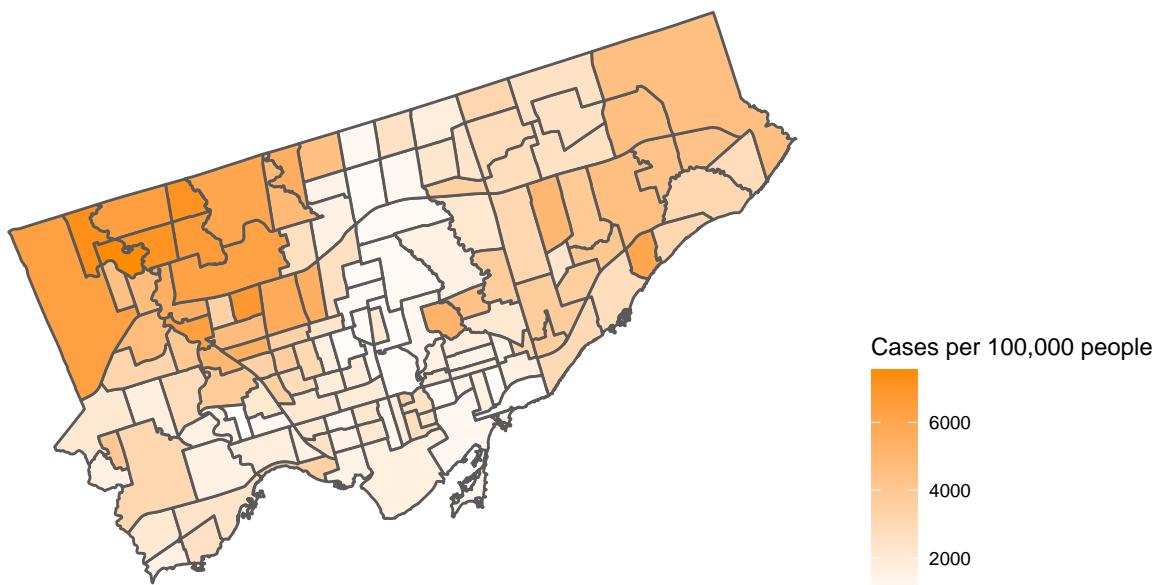
Percentage of 18 to 64 year olds living in a low income family (2015)
Neighbourhoods of Toronto, Canada



Created by: Qing Wen for STA303/1002, U of T
Source: Census Profile 98-316-X2016001 via OpenData Toronto
Data as of January 28, 2021

```
ggplot(data = nbhoods_final) +  
  geom_sf(aes(fill = rate_per_100000)) +  
  labs(  
    title = "COVID-19 cases per 100,000, by neighbourhood in Toronto, Canada",  
    caption = str_c("Created by: Qing Wen for STA303/1002, U of T\nSource: Ontario Ministry of Health,  
    date_daily[1, 1])  
) +  
  theme_map() +  
  theme(legend.position = "right") +  
  scale_fill_gradient(name = "Cases per 100,000 people", low = "white", high = "darkorange")
```

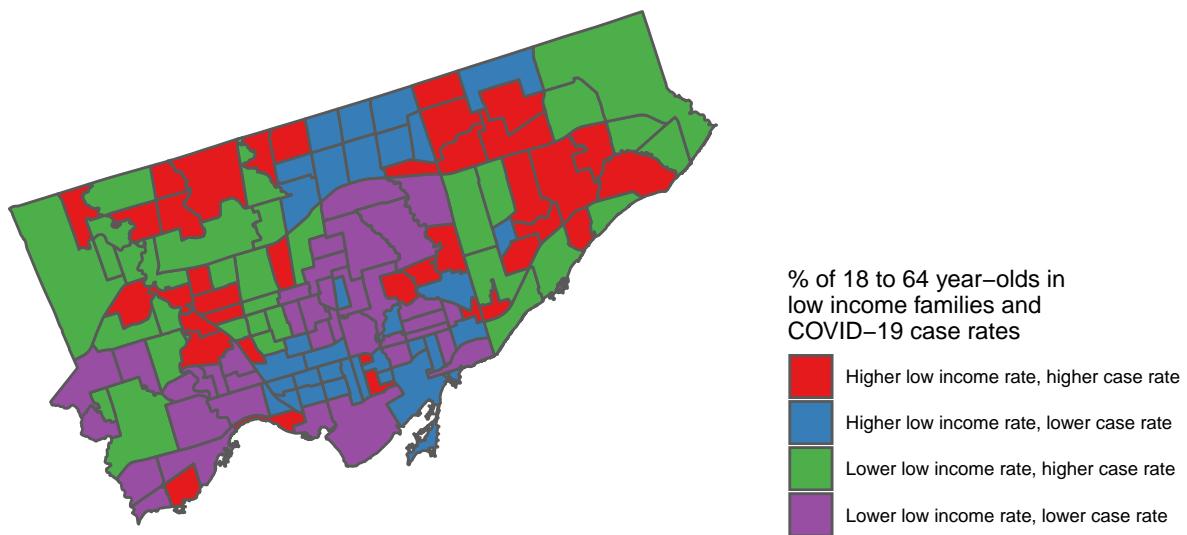
COVID-19 cases per 100,000, by neighbourhood in Toronto, Canada



Created by: Qing Wen for STA303/1002, U of T
Source: Ontario Ministry of Health, Integrated Public Health Information System and CORES
Data as of January 28, 2021

```
ggplot(data = nbhoods_final) +  
  geom_sf(aes(fill = nbhood_type)) +  
  labs(  
    title = "COVID-19 cases and low income status by neighbourhood in Toronto, Canada",  
    caption = str_c("Created by: Qing Wen for STA303/1002, U of T\nIncome data source: Census Profile 98-316-X2016001 via OpenData Toronto  
    COVID data source: Ontario Ministry of Health, Integrated Public  
    Health Information System and CORES  
    Data as of January 28, 2021")  
) +  
  theme_map() +  
  theme(legend.position = "right") +  
  scale_fill_brewer(name = "% of 18 to 64 year-olds in\nlow income families and\nCOVID-19 case rates",  
    palette = "Set1")
```

COVID-19 cases and low income status by neighbourhood in Toronto, Canada



Created by: Qing Wen for STA303/1002, U of T
Income data source: Census Profile 98-316-X2016001 via OpenData Toronto
COVID data source: Ontario Ministry of Health, Integrated Public
Health Information System and CORES
Data as of January 28, 2021