Code last run 2021-02-15.
Daily: Data as of January 29, 2021.
Neighbourhood: Data as of January 28, 2021.

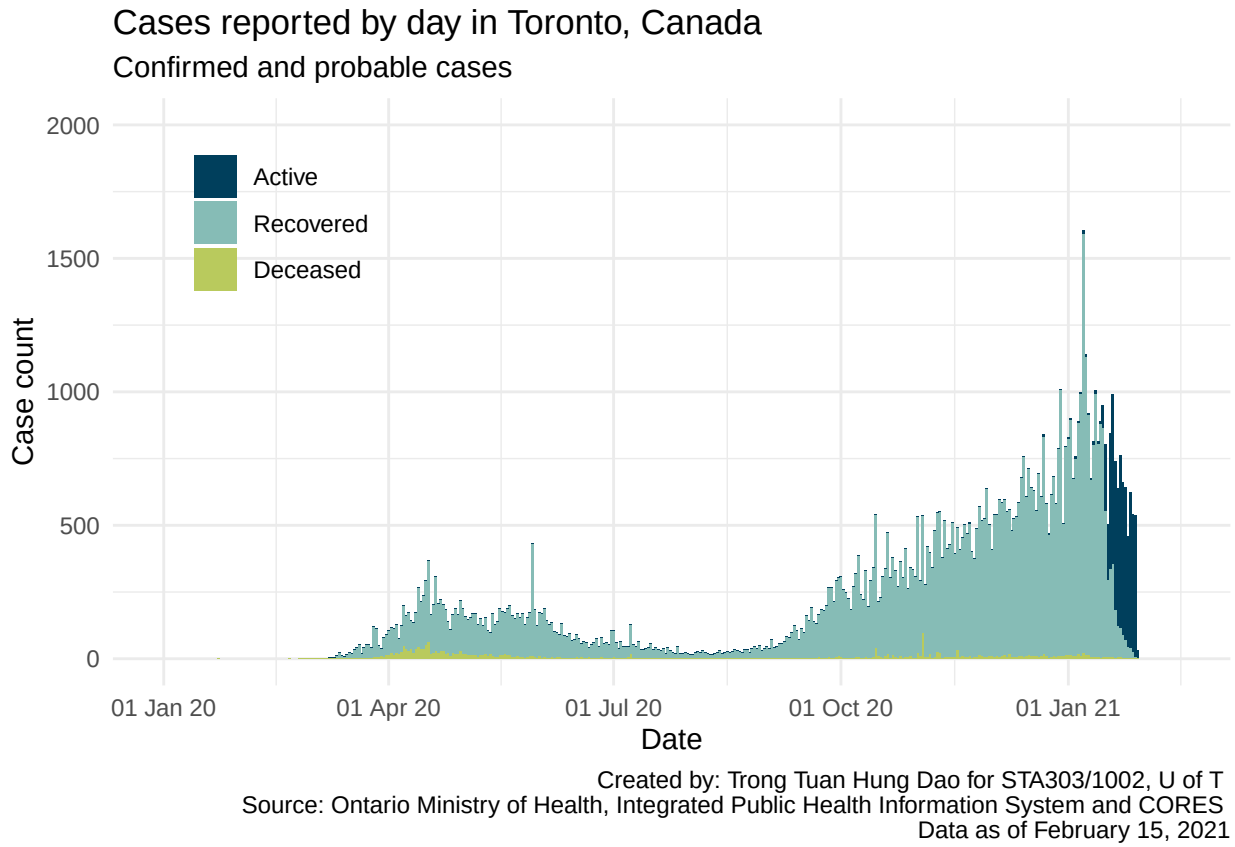
Task 1: Daily cases

Data wrangling

```
reported <- reported_raw %>%
  mutate_if(is.numeric, replace_na, replace=0)

reported$reported_date <- date(reported$reported_date)

reported <- reported %>%
  pivot_longer(-reported_date, names_to = 'Status', values_to = 'count')

reported$Status <- str_to_title(reported$Status)

reported <- reported %>%
  mutate(Status = fct_relevel(Status, 'Deceased', after = 2))
```

Data visualization

```
reported %>%
  ggplot(aes(x = reported_date, y = count, fill = as.factor(Status))) +
  geom_bar(stat = "identity") +
  scale_fill_manual(values = c('#003F5C', '#86BCB6', '#B9CA5D')) +
  scale_x_date(limits = c(date("2020-01-01"), Sys.Date()),
               labels = scales::date_format("%d %b %y")) +
  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: Trong Tuan Hung Dao for STA303/1002, U of T
                       Source: Ontario Ministry of Health, Integrated Public Health Information System a
                       Data as of ", format(Sys.Date(), "%B %d, %Y"))) +
  theme(legend.title = element_blank(), legend.position = c(0.15, 0.8)) +
  scale_y_continuous(limits = c(0, 2000))
```



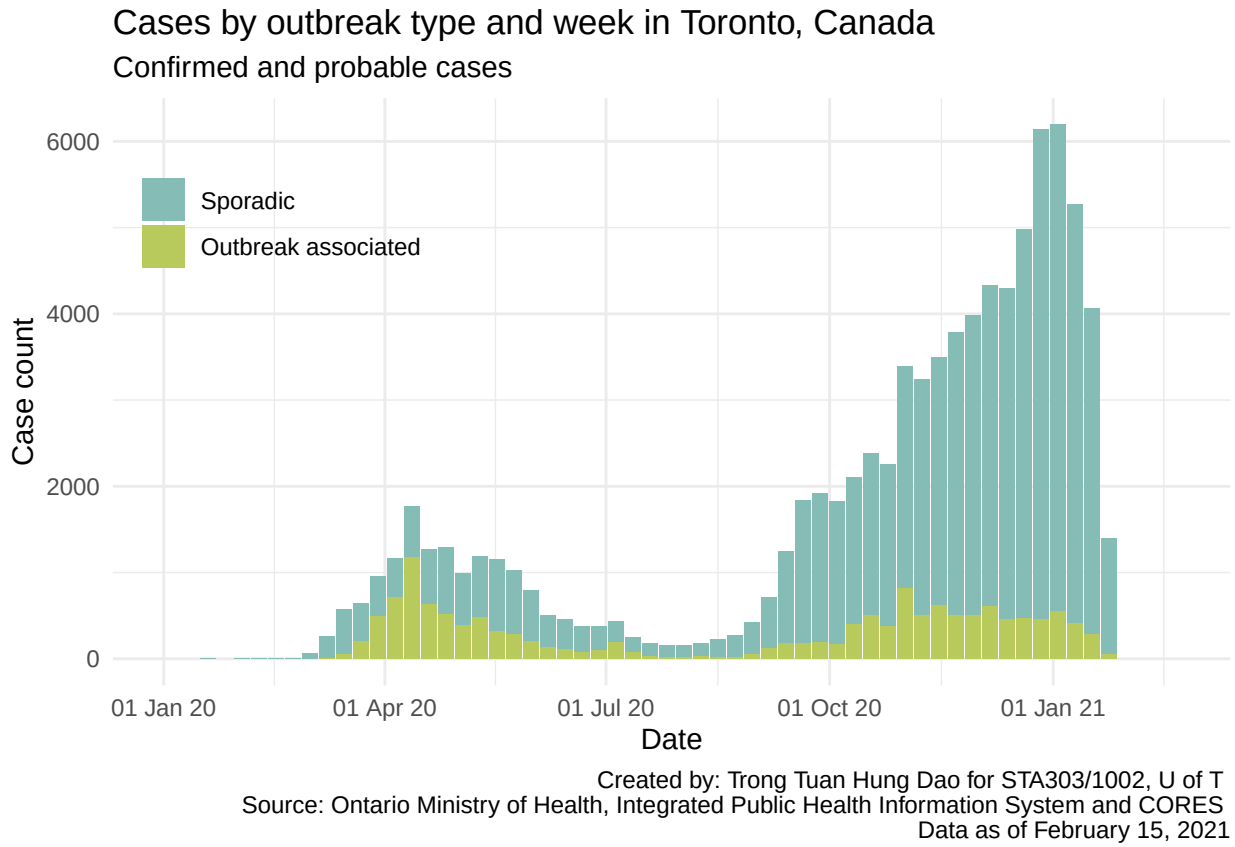
Task 2: Outbreak type

Data wrangling

```
outbreak <- outbreak_raw %>%
  mutate_if(is.numeric, replace_na, replace =0)

outbreak$episode_week <- date(outbreak$episode_week)

outbreak$outbreak_or_sporadic <- str_replace_all(outbreak$outbreak_or_sporadic,
                                                 'OB Associated', 'Outbreak associated')

outbreak <- outbreak %>%
  group_by(episode_week) %>%
  mutate(total_cases = sum(cases))

outbreak$outbreak_or_sporadic <- factor(outbreak$outbreak_or_sporadic)

outbreak <- outbreak %>%
  mutate(outbreak_or_sporadic = fct_relevel(outbreak_or_sporadic, 'Outbreak associated', after = 1))
```

Data visualization

```
outbreak %>%
  ggplot(aes(x = episode_week, y = cases, fill = as.factor(outbreak_or_sporadic))) +
  geom_bar(stat = "identity") +
  scale_fill_manual(values = c('#86BCB6', '#B9CA5D')) +
  scale_x_date(limits = c(date("2020-01-01"), Sys.Date() + 7),
               labels = scales::date_format("%d %b %y")) +
  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: Trong Tuan Hung Dao for STA303/1002, U of T
                       Source: Ontario Ministry of Health, Integrated Public Health Information System a
                       Data as of ", format(Sys.Date(), "%B %d, %Y"))) +
  theme(legend.title = element_blank(), legend.position = c(.15, .8)) +
  scale_y_continuous(limits = c(0, max(outbreak$total_cases)))
```



Task 3: Neighbourhoods

Data wrangling: part 1

```
income <- nbhood_profile %>%
  filter(Characteristic == '  18 to 64 years (%)') %>%
  filter(`_id` == 1143) %>%
  pivot_longer(-c(`_id`, Category, Topic, `Data Source`, Characteristic), names_to = 'neighbourhood_name
  mutate(percentage = parse_number(percentage))
```

Data wrangling: part 2

```
nbhoods_all <- nbhoods_shape_raw %>%
  mutate(neighbourhood_name = str_remove(AREA_NAME, '\\d+')) %>%
  mutate(neighbourhood_name = str_remove(neighbourhood_name, '\\(')) %>%
  mutate(neighbourhood_name = str_remove(neighbourhood_name, '\\)')) %>%
  mutate(neighbourhood_name = str_remove(neighbourhood_name, '\\s$'))

nbhood_raw <- nbhood_raw %>%
  mutate(neighbourhood_name = str_remove(neighbourhood_name, '\\d+')) %>%
  mutate(neighbourhood_name = str_remove(neighbourhood_name, '\\(')) %>%
  mutate(neighbourhood_name = str_remove(neighbourhood_name, '\\)')) %>%
  mutate(neighbourhood_name = str_remove(neighbourhood_name, '\\s$')) %>%
  mutate(neighbourhood_name = str_replace(neighbourhood_name, 'Mimico includes Humber Bay Shores', 'Mim


nbhoods_all <- nbhoods_all %>%
  mutate(neighbourhood_name = str_replace(neighbourhood_name, 'Weston-Pellam Park', 'Weston-Pelham Park
  mutate(neighbourhood_name = str_replace(neighbourhood_name, 'North St.James Town', 'North St. James To
  mutate(neighbourhood_name = str_replace(neighbourhood_name, 'Cabbagetown-South St.James Town', 'Cabbag
  mutate(neighbourhood_name = str_replace(neighbourhood_name, filter(nbhoods_all, `_id` == 10873)$neighb
  mutate(neighbourhood_name = str_remove(neighbourhood_name, '\\(')) %>%
  mutate(neighbourhood_name = str_remove(neighbourhood_name, '\\)')) %>%
  left_join(nbhood_raw, by = 'neighbourhood_name')

income <- income %>%
  mutate(neighbourhood_name = str_remove(neighbourhood_name, '\\d+')) %>%
  mutate(neighbourhood_name = str_remove(neighbourhood_name, '\\(')) %>%
  mutate(neighbourhood_name = str_remove(neighbourhood_name, '\\)')) %>%
  mutate(neighbourhood_name = str_remove(neighbourhood_name, '\\s$')) %>%
  mutate(neighbourhood_name = str_replace(neighbourhood_name, 'Mimico includes Humber Bay Shores', 'Mim

nbhoods_all <- nbhoods_all %>%
  left_join(income, by = 'neighbourhood_name')
```

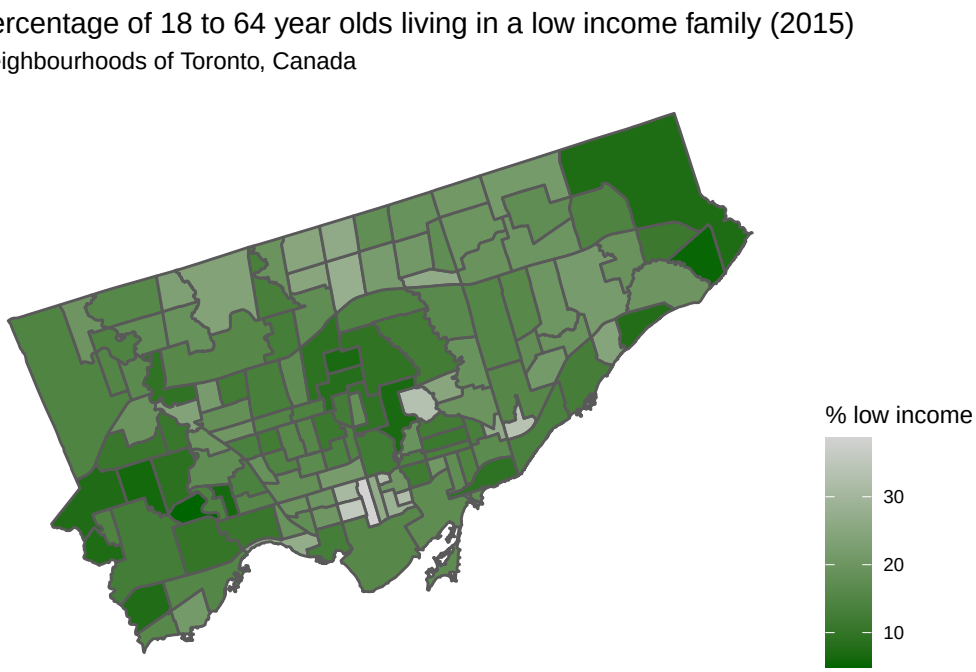
Data wrangling: part 3

```
nbhoods_final <- nbhoods_all %>%
  mutate(med_inc = median(nbhoods_all$percentage)) %>%
  mutate(med_rate = median(nbhoods_all$rate_per_100_000_people)) %>%
  mutate(nbhood_type = ifelse(percentage >= med_inc & rate_per_100_000_people >= med_rate,
                              'Higher low income rate, higher case rate',
                              ifelse(percentage >= med_inc & rate_per_100_000_people < med_rate,
```

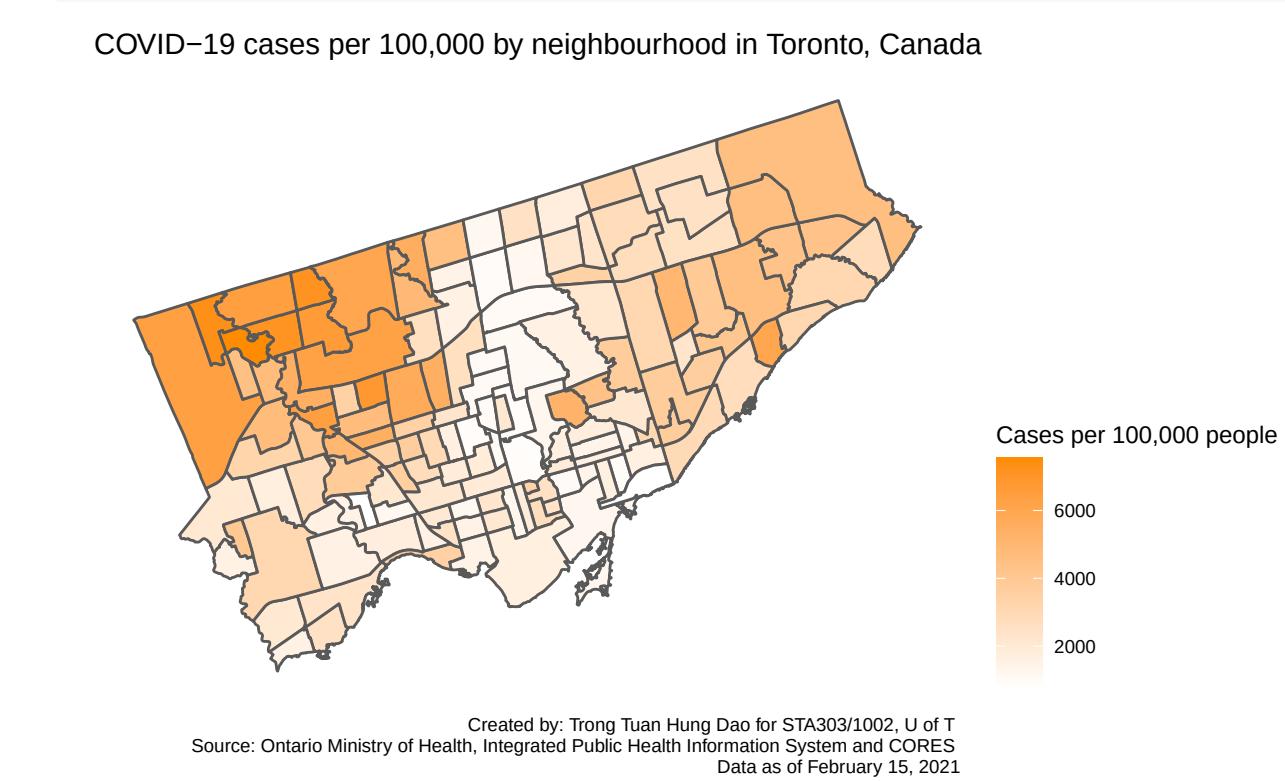
```
                                    'Higher low income rate, lower case rate',
                                    ifelse(percentage < med_inc & rate_per_100_000_people >= med_rate,
                                           'Lower low income rate, higher case rate','Lower low income
```

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: Trong Tuan Hung Dao for STA303/1002, U of T
                        Source: Census Profile 98-316-X2016001 via OpenData Toronto
                        Data as of ", format(Sys.Date(), "%B %d, %Y"))) +
  scale_fill_gradient(name='% low income', low = 'darkgreen', high = 'lightgray') +
  theme_map() +
  theme(legend.position = 'right')
```



```
ggplot(data = nbhoods_final) +
  geom_sf(aes(fill=rate_per_100_000_people)) +
  labs(title = 'COVID-19 cases per 100,000 by neighbourhood in Toronto, Canada',
       caption = str_c("Created by: Trong Tuan Hung Dao for STA303/1002, U of T
                       Source: Ontario Ministry of Health, Integrated Public Health Information System a
                       Data as of ", format(Sys.Date(), "%B %d, %Y"))) +
  scale_fill_gradient(name='Cases per 100,000 people', low = 'white', high = 'darkorange') +
  theme_map() +
  theme(legend.position = 'right')
```



```
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: Trong Tuan Hung Dao 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 ", format(Sys.Date(), "%B %d, %Y"))) +
  scale_fill_brewer(name='% of 18 to 64 year-olds
                    in low income families and
                    COVID-19 case rates', palette = 'Set1') +
  theme_map() +
  theme(legend.position = 'right')
```