Code last run 2021-02-13.
Daily: Data as of January 29, 2021.
Neighbourhood: Data as of January 28, 2021.

Task 1: Daily cases

Data wrangling

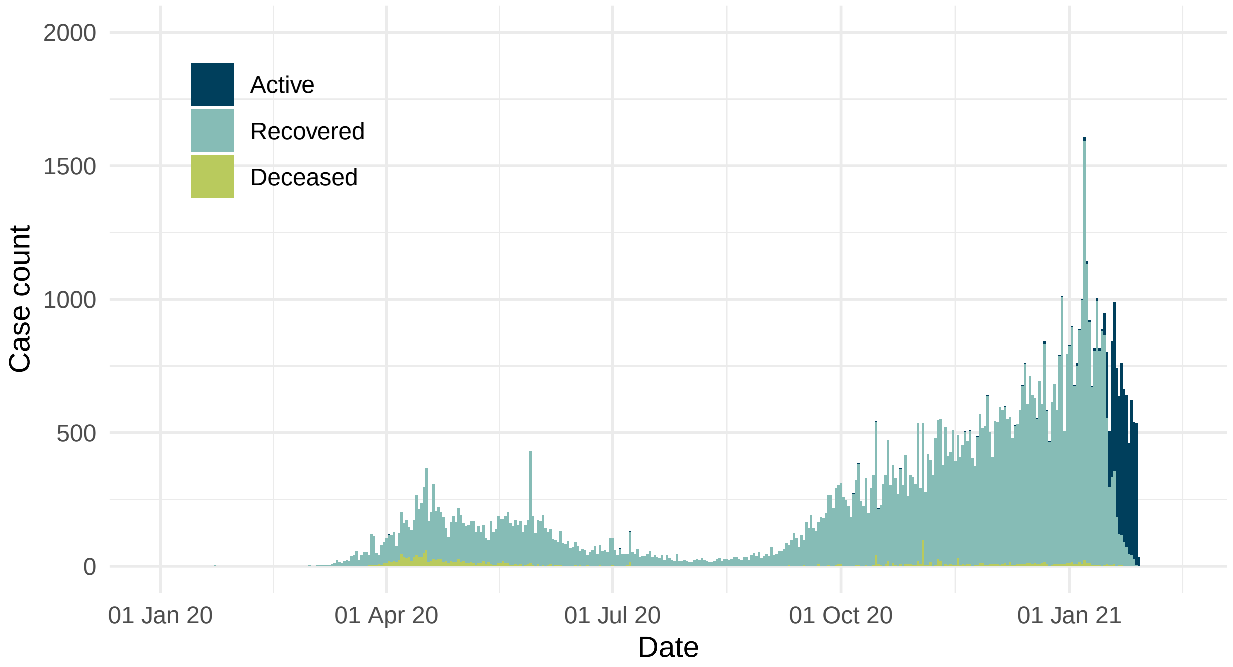
```r
reported <- reported_raw %>%
  mutate_if(is.numeric, replace_na, replace=0)

# reformat dates
reported$reported_date <- as.Date(reported$reported_date, "%Y-%m-%d")

# capitalize column names
colnames(reported)[2] <- "Recovered"
colnames(reported)[3] <- "Active"
colnames(reported)[4] <- "Deceased"

# move columns around
reported <- reported[c("reported_date", "Active", "Recovered", "Deceased")]

# need to make a new column called "Case Type", as data is not tidy
reported_long <- reported %>%
  pivot_longer(-reported_date, names_to = "case_type") %>%
  uncount(value)

# create factor levels (sort of cheating)
fac_levels <- c("Active", "Recovered", "Deceased")

# verify that reported_date is indeed in date format
# glimpse(reported)
```

Data visualization

```
reported_long %>%
  count(case_type, reported_date) %>%
  ggplot(aes(x = reported_date, y = n, fill = factor(case_type, levels = fac_levels))) +
  geom_bar(stat = "identity") +
  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: David Pham for STA303/1002, U of T\n",
                       "Source: Ontario Ministry of Health, Integrated Public Health Information System
                       date_daily[1,1])) +
  scale_x_date(limits = c(date("2020-01-01"), Sys.Date()), date_labels = "%d %b %y") +
  theme(legend.title = element_blank(), legend.position = c(0.15, 0.8)) +
  scale_y_continuous(limits = c(0, 2000), breaks = seq(0, 2000, by = 500)) +
  scale_fill_manual(values = c("#003F5C", "#86BCB6", "#B9CA5D"))
```

Cases reported by day in Toronto, Canada

Confirmed and probable cases



Created by: David Pham for STA303/1002, U of T
Source: Ontario Ministry of Health, Integrated Public Health Information System and CORES
Data as of January 29, 2021

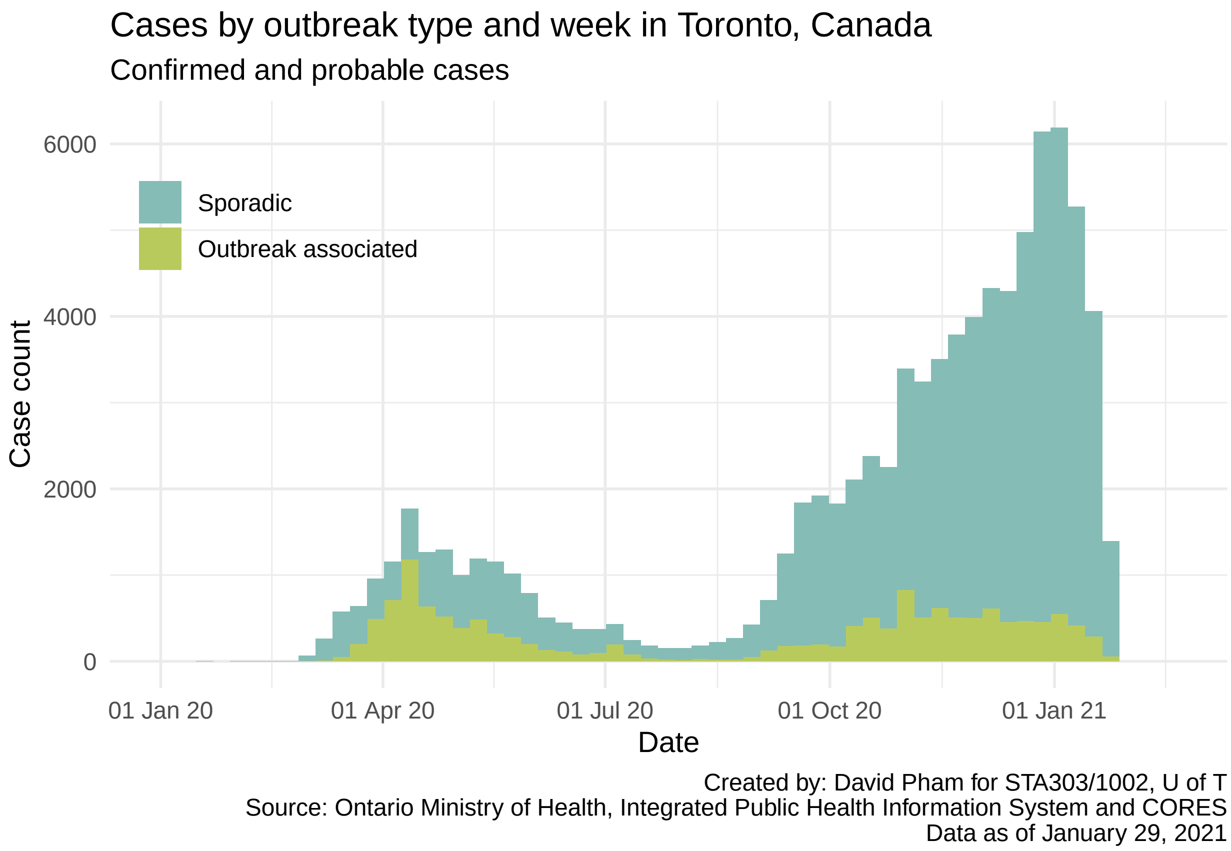
Task 2: Outbreak type

Data wrangling

```r
# create total_cases variable and fix wording for outbreak type
outbreak <- outbreak_raw %>%
  mutate(outbreak_or_sporadic = str_replace(outbreak_or_sporadic, "OB A", "Outbreak a")) %>%
  group_by(episode_week) %>%
  mutate(total_cases = sum(cases))

# reformat dates
outbreak$episode_week <- as.Date(outbreak$episode_week, "%Y-%m-%d")

# verify that episode_week is indeed in date format
# glimpse(outbreak)

# create factors (sort of cheating)
fac_levels <- c("Sporadic", "Outbreak associated")
```

Data visualization

```
outbreak %>%
  ggplot(aes(x = episode_week, y = cases, fill = factor(outbreak_or_sporadic, levels = fac_levels))) +
  geom_bar(stat = "identity", width = 7) +
  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: David Pham for STA303/1002, U of T\n",
                       "Source: Ontario Ministry of Health, Integrated Public Health Information System
                       date_daily[1,1])) +
  scale_x_date(labels = scales::date_format("%d %b %y"),
               limits = c(date("2020-01-01"), Sys.Date()+7)) +
  theme(legend.title = element_blank(), legend.position = c(0.15, 0.8)) +
  scale_y_continuous(limits = c(0, max(outbreak$total_cases)),
                     breaks = seq(0, max(outbreak$total_cases), by = 2000)) +
  scale_fill_manual(values = c("#86BCB6", "#B9CA5D"))
```



Cases by outbreak type and week in Toronto, Canada

Confirmed and probable cases

Created by: David Pham for STA303/1002, U of T
Source: Ontario Ministry of Health, Integrated Public Health Information System and CORES
Data as of January 29, 2021

Task 3: Neighbourhoods

Data wrangling: part 1

```
# filter for the correct row (we'll use LICO-AT)
income <- nbhood_profile %>%
  filter(nbhood_profile$Category == "Income", nbhood_profile$Topic == "Low income in 2015",
         nbhood_profile$Characteristic == "  18 to 64 years (%)", nbhood_profile$'_id' == 1143)

# delete unnecessary variables
income <- select(income, c(-'_id', -Category, -Topic, -Characteristic, -"Data Source"))

# make data tidy so that we only have neighbourhood names and % of low income 18 to 64 year olds
income <- income %>%
  pivot_longer(cols = everything(), names_to = "neighbourhood_name", values_to = "Percentage")

# change value types in percentages column to doubles
income$Percentage <- parse_number(income$Percentage)

# glimpse(income)
```

Data wrangling: part 2

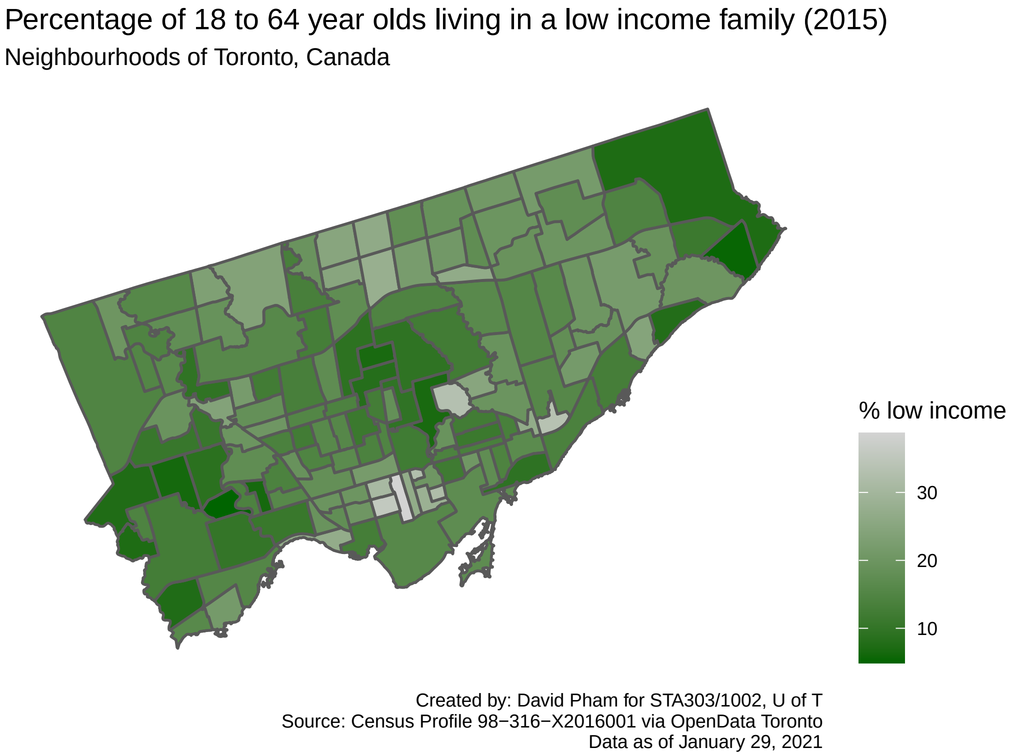
```
# make neighbourhood_name variable, mutating AREA_NAME
nbhoods_all <- nbhoods_shape_raw %>%
  mutate(neighbourhood_name = str_replace_all(string = nbhoods_shape_raw$AREA_NAME,
                                              pattern = "\\s\\(\\d+\\)$",
                                              replacement = ""))

# after a bit of observing, it appears that three neighbourhoods have been spelt incorrectly in nbhoods_
# we'll edit the city names in 'neighbourhood_name' so that we don't NA values when merging datasets.
nbhoods_all$neighbourhood_name[127] <- "Cabbagetown-South St. James Town"
nbhoods_all$neighbourhood_name[76] <- "Weston-Pelham Park"
nbhoods_all$neighbourhood_name[54] <- "North St. James Town"

# left join nbhood_raw and income datasets to nbhoods_all by 'neighbourhood_name'
nbhoods_all <- left_join(nbhoods_all, nbhood_raw, by = "neighbourhood_name") %>%
  left_join(., income, by = "neighbourhood_name")

# rename rate_per_100_000_people to rate_per_100000
colnames(nbhoods_all)[21] <- "rate_per_100000"
```

Data wrangling: part 3

```r
# create med_inc, med_rate and nbhood_type variables
nbhoods_final <- nbhoods_all %>%
  mutate(med_inc = median(Percentage, na.rm = TRUE), med_rate = median(rate_per_100000, na.rm = TRUE),
         nbhood_type = ifelse(Percentage >= med_inc & rate_per_100000 >= med_rate,
                              "Higher low income rate, higher case rate",
                              ifelse(Percentage >= med_inc & rate_per_100000 < med_rate,
                                     "Higher low income rate, lower case rate",
                                     ifelse(Percentage < med_inc & rate_per_100000 >= med_rate,
                                            "Lower low income rate, higher case rate",
                                            ifelse(Percentage < med_inc & rate_per_100000 < med_rate,
                                                   "Lower low income rate, lower case rate", NA)))))
```

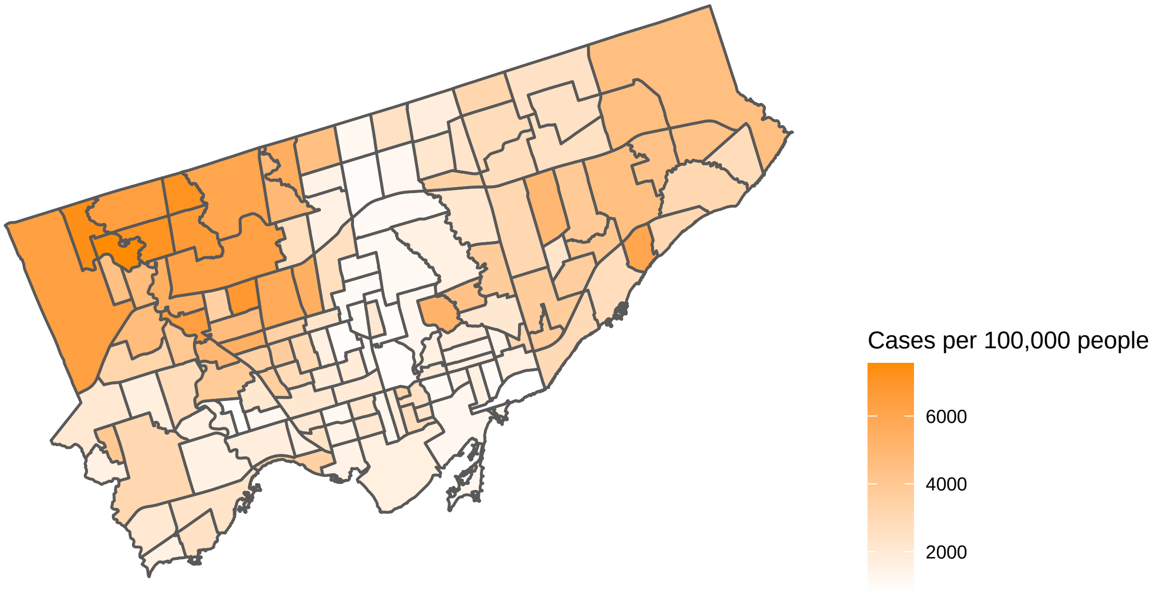
Data visualization

```
ggplot(data = nbhoods_final) +
  geom_sf(aes(fill = Percentage)) +
  theme_map() +
  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: David Pham for STA303/1002, U of T\n",
                        "Source: Census Profile 98-316-X2016001 via OpenData Toronto\n",
                        date_daily[1,1])) +
  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
% low income
30
20
10
Created by: David Pham for STA303/1002, U of T
Source: Census Profile 98−316−X2016001 via OpenData Toronto
Data as of January 29, 2021

```
ggplot(data = nbhoods_final) +
  geom_sf(aes(fill = rate_per_100000)) +
  theme_map() +
  labs(title = "COVID-19 cases per 100,000, by neighbourhood in Toronto, Canada",
       caption = str_c("Created by: David Pham for STA303/1002, U of T\n",
                       "Source: Ontario Ministry of Health, Integrated Public Health Information System
                       date_daily[1,1])) +
  theme(legend.position = 'right') +
  scale_fill_gradient(name = "Cases per 100,000 people", low = "white", high = "darkorange")
```

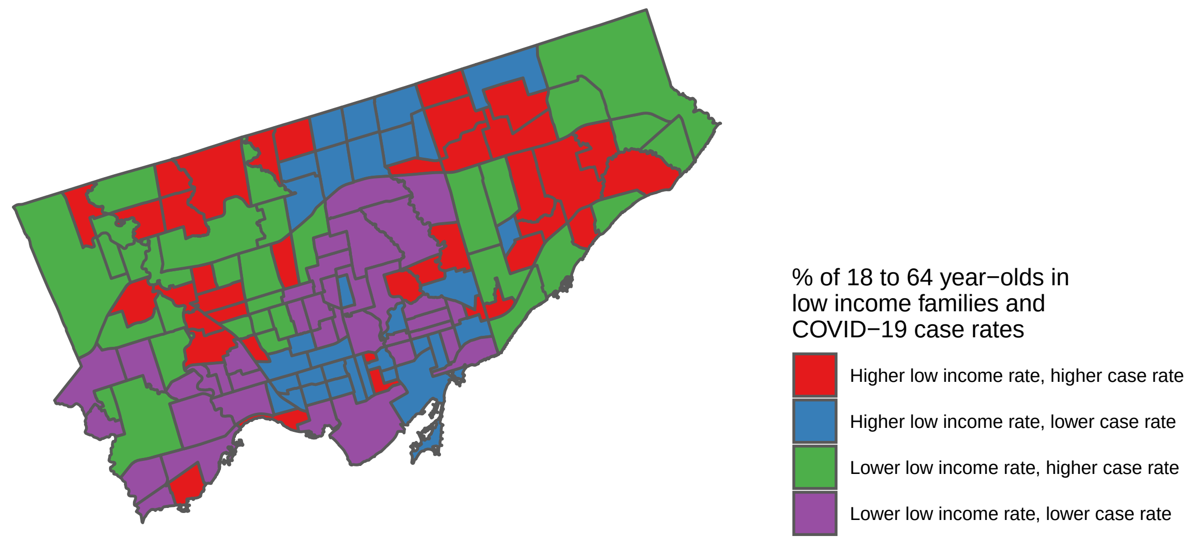
COVID−19 cases per 100,000, by neighbourhood in Toronto, Canada



Created by: David Pham for STA303/1002, U of T
Source: Ontario Ministry of Health, Integrated Public Health Information System and CORES
Data as of January 29, 2021

```r
ggplot(data = nbhoods_final) +
  geom_sf(aes(fill = nbhood_type)) +
  theme_map() +
  labs(title = "COVID-19 cases and low-income status by neighbourhood in Toronto, Canada",
       fill = str_c("% of 18 to 64 year-olds in\n",
                    "low income families and\n",
                    "COVID-19 case rates"),
       caption = str_c("Created by: David Pham for STA303/1002, U of T\n",
                       "Income data source: Census Profile 98-316-X2016001 via OpenData Toronto\n",
                       "COVID data source: Ontario Ministry of Health, Integrated Public\n",
                       "Health Information System and CORES\n",
                       date_daily[1,1])) +
  theme(legend.position = 'right') +
  scale_fill_brewer(palette = 'Set1')
```

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


% of 18 to 64 year−olds in
low income families and
COVID−19 case rates
🟥 Higher low income rate, higher case rate
🟦 Higher low income rate, lower case rate
🟩 Lower low income rate, higher case rate
🟪 Lower low income rate, lower case rate

Created by: David Pham 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 29, 2021