Code last run 2021-02-21.
Daily: Data as of February 19, 2021.
Neighbourhood: Data as of February 18, 2021.

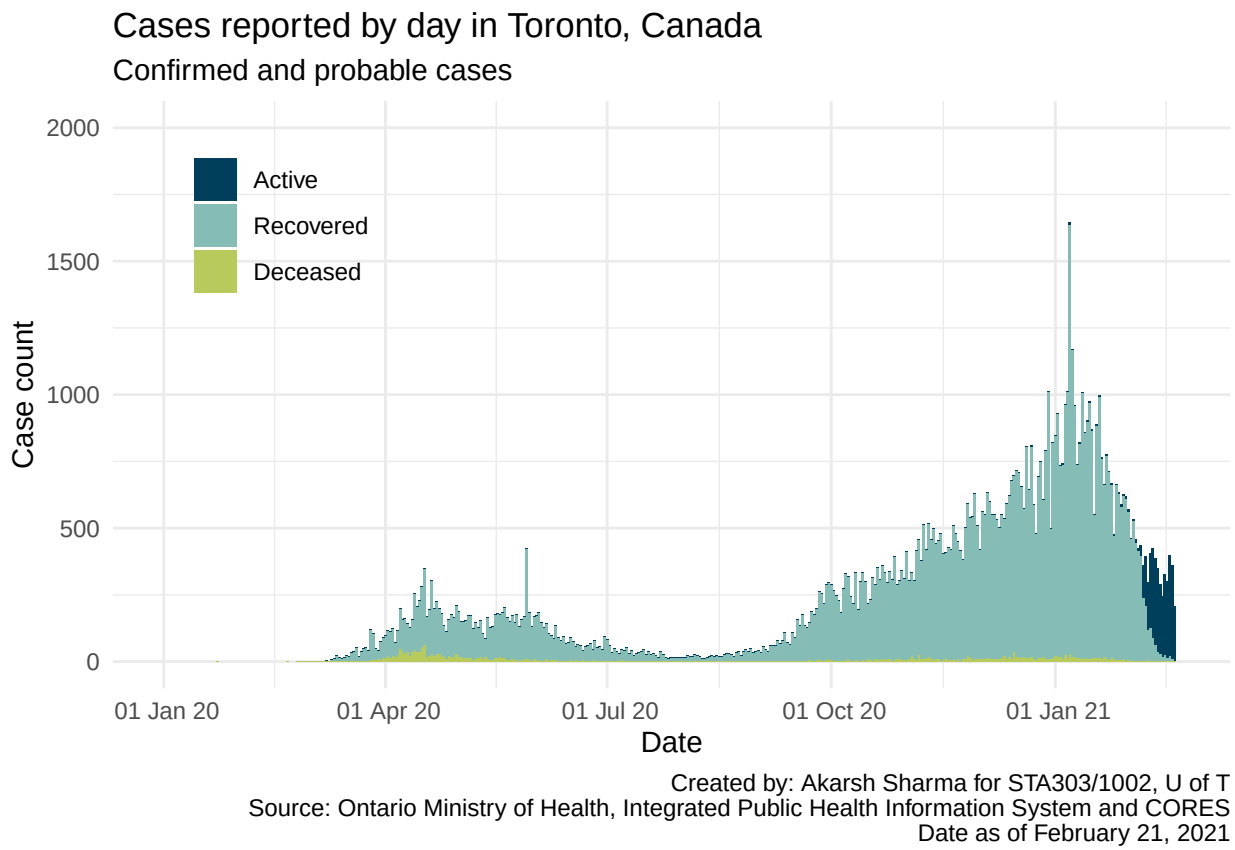
Task 1: Daily cases

Data wrangling

```
# Changing the NAs to 0s
reported <- reported_raw %>% mutate_if(is.numeric, replace_na, replace = 0)

# Changing the "reported_date" column to correct date format
reported <- reported %>% mutate(reported_date=as.Date(reported_date, format = "%Y.%m.%d"))

# Making the data tidy
reported <- reported %>% pivot_longer(-c(reported_date), names_to =
            "Case_Status", values_to = "Number_of_cases")

# Capitalize the "Case_status" column properly
reported <- reported %>%
  mutate(Case_Status = case_when(
    str_detect(Case_Status, "recovered") ~ "Recovered",
    str_detect(Case_Status, "active") ~ "Active",
    str_detect(Case_Status, "deceased") ~ "Deceased"))

# Making sure the case status appears in the correct order in the legend
reported$Case_Status <- factor(reported$Case_Status, levels = c("Active", "Recovered", "Deceased"))
```

Data visualization

```
reported %>%
  ggplot(aes(x = reported_date, y = Number_of_cases, fill = Case_Status)) +
  geom_bar(stat = "identity") +
  scale_y_continuous(limits = c(0,2000)) +
  scale_x_date(limits = c(as.Date("2020-01-01"), as.Date(Sys.Date())), date_labels = " %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: Akarsh Sharma for STA303/1002, U of T\n",
"Source: Ontario Ministry of Health, Integrated Public Health Information System and CORES\n",
"Date as of ", format(Sys.Date(), format = "%B %d, %Y"))) +
  theme(legend.title = element_blank(), legend.position = c(0.15, 0.8)) +
scale_fill_manual(values = c("#003F5C", "#86BCB6", "#B9CA5D"))
```



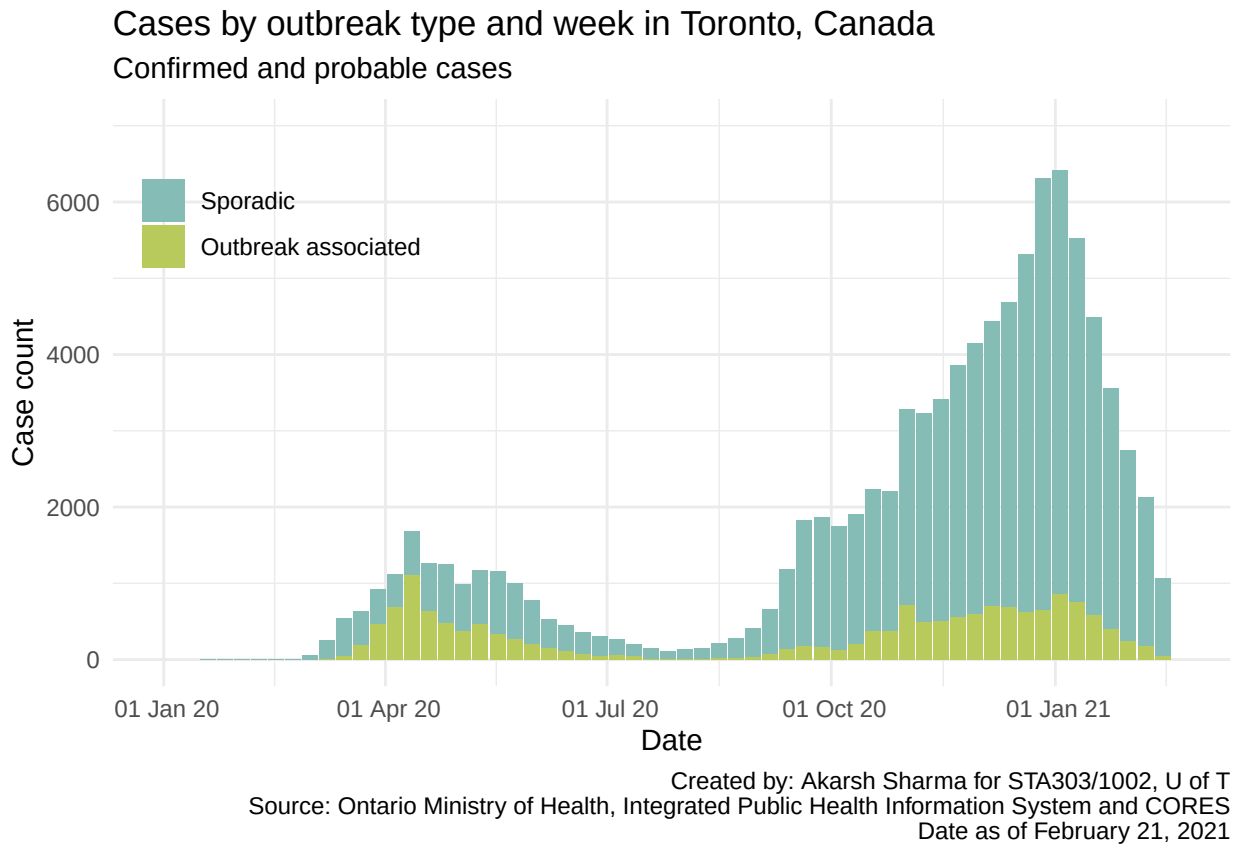
Task 2: Outbreak type

Data wrangling

```
outbreak <- outbreak_raw

# Changing the episode week to proper date column format
outbreak <- outbreak %>% mutate(episode_week = as.Date(episode_week, format = "%Y.%m.%d"))

# Proper capitalizing of outbreak or sporadic
outbreak <- outbreak %>% mutate(outbreak_or_sporadic = case_when(
    str_detect(outbreak_or_sporadic, "Sporadic") ~ "Sporadic",
    str_detect(outbreak_or_sporadic, "OB Associated") ~ "Outbreak associated"))

# Making sure legend names appear correctly
outbreak$outbreak_or_sporadic <- factor(outbreak$outbreak_or_sporadic,
                    levels =c("Sporadic","Outbreak associated"))

# New variable total_cases
outbreak <- outbreak %>% 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_y_continuous(limits = c(0,7000)) +
  scale_x_date(limits = c(as.Date("2020-01-01"), as.Date(Sys.Date())), date_labels = " %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: Akarsh Sharma for STA303/1002, U of T\n",
  "Source: Ontario Ministry of Health, Integrated Public Health Information System and CORES\n",
  "Date as of ", format(Sys.Date(), format = "%B %d, %Y"))) +
  theme(legend.title = element_blank(), legend.position = c(.15, .8)) +
  scale_fill_manual(values = c("#86BCB6", "#B9CA5D"))
```



Task 3: Neighbourhoods

Data wrangling: part 1

```
# Filtering the nbhood_profile dataset
income <- nbhood_profile

income <- filter(as.data.frame(nbhood_profile), nbhood_profile$Topic %in% c("Low income in 2015"))
income <- filter(as.data.frame(nbhood_profile),
      nbhood_profile$Characteristic %in% c("  18 to 64 years (%)"))
income <- filter(as.data.frame(nbhood_profile), nbhood_profile$`_id` %in% c(1143))

# Making sure the data is tidy
income <- income %>% pivot_longer(-c(`_id`, Category, Topic, `Data Source`,
Characteristic), names_to = "Neighbourhoods", values_to = "Percentage of 18 to 64 year olds")
```

Data wrangling: part 2

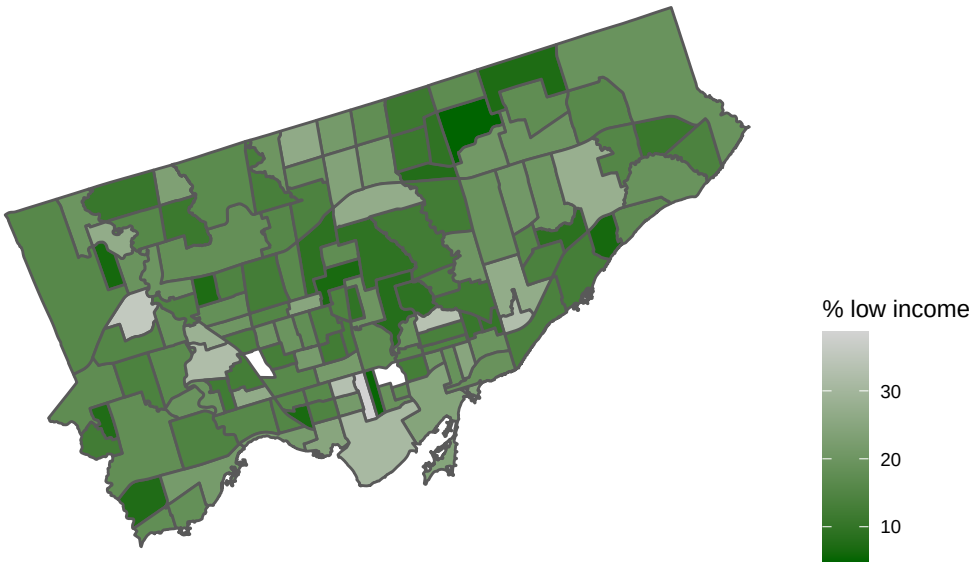
```
nbhoods_all <- nbhoods_shape_raw
nbhoods_all <- nbhoods_all %>%
  mutate(neighbourhood_name = AREA_NAME) %>% mutate(
    neighbourhood_name = str_remove(neighbourhood_name, "\\s\\(\\d+\\)$"))

# Removing NA row from nbhood_raw
nbhood_raw <- nbhood_raw %>% filter(!is.na(neighbourhood_id))
# Removing "City of Toronto" row from income dataset
income <- income %>% filter(Neighbourhoods != "City of Toronto")
# Merging the datasets to get required columns
nbhoods_all <- full_join(nbhoods_all, nbhood_raw)
nbhoods_all <- merge(nbhoods_all, income, by.x = "neighbourhood_name", by.y = "Neighbourhoods")
# Renaming rate_per_100_000_people
nbhoods_all <- nbhoods_all %>% rename(rate_per_100000 = rate_per_100_000_people)
nbhoods_all <- nbhoods_all %>% mutate(`Percentage of 18 to 64 year olds`=
parse_number(as.character(`Percentage of 18 to 64 year olds`)))
```

Data wrangling: part 3

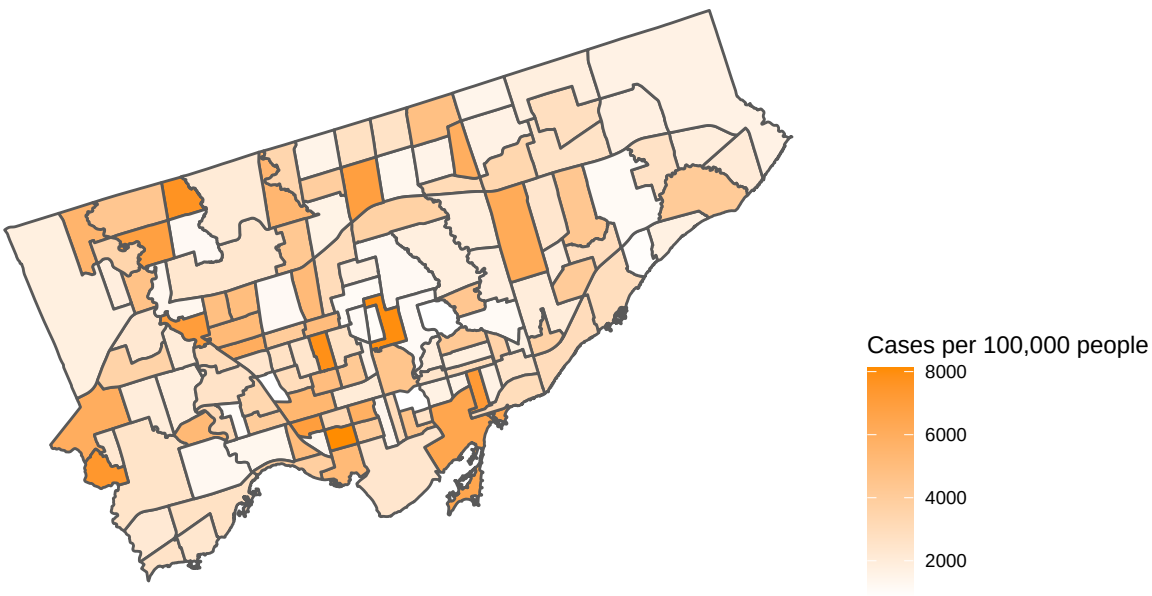
```
# New variables med_inc and med_rate
nbhoods_final <- nbhoods_all %>% mutate(
  med_inc = median(`Percentage of 18 to 64 year olds`)) %>% mutate(med_rate = median(rate_per_100000))

# New variable nbhood_type
nbhoods_final <- nbhoods_final %>% mutate(nbhood_type = case_when(
  `Percentage of 18 to 64 year olds`>= med_inc & rate_per_100000 >= med_rate ~
    "Higher low income rate, higher case rate",
  `Percentage of 18 to 64 year olds`>=med_inc & rate_per_100000 < med_rate ~
    "Higher low income rate, lower case rate",
  `Percentage of 18 to 64 year olds`< med_inc & rate_per_100000>= med_rate ~
    "Lower low income rate, higher case rate",
  `Percentage of 18 to 64 year olds` < med_inc & rate_per_100000 < med_rate ~
    "Lower low income rate, lower case rate"
))
```

Data visualization

```
ggplot(data = nbhoods_final) +
geom_sf(aes(fill = nbhoods_final$`Percentage of 18 to 64 year olds`)) +
theme_map() +
theme(legend.position = "none") +
scale_fill_gradient(name="% low income", low = "darkgreen", high = "lightgrey") +
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: Akarsh Sharma for STA303/1002, U of T\n", "Source:  Census Profile 9
  "Date as of ", format(Sys.Date(), format = "%B %d, %Y"))) + theme(legend.position = "right")
```



```
ggplot(data = nbhoods_final) +
geom_sf(aes(fill = nbhoods_final$rate_per_100000)) +
theme_map() +
theme(legend.position = "none") +
scale_fill_gradient(name="Cases per 100,000 people", low = "white", high = "darkorange") +
labs(title = "COVID-19 cases per 100,000, by neighbourhood in Toronto, Canada",
       caption = str_c("Created by: Akarsh Sharma for STA303/1002, U of T\n",
"Source: Ontario Ministry of Health, Integrated Public Health Information System and CORES\n",
                       "Date as of ", format(Sys.Date(), format = "%B %d, %Y"))) +
  theme(legend.position = "right")
```



```
ggplot(data = nbhoods_final) +
geom_sf(aes(fill = nbhood_type)) +
theme_map() +
theme(legend.position = "none") +
scale_fill_brewer(name = "% of 18 to 64 year-olds in\n low income families and\n COVID-19 case rates",
                  palette = "Set1") +

 labs(title = "COVID-19 cases per 100,000, by neighbourhood in Toronto, Canada",
      caption = str_c("Created by: Akarsh Sharma for STA303/1002, U of T\n",
      "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\n",
      "Date as of ", format(Sys.Date(), format = "%B %d, %Y"))) + theme(legend.position = "right")
```