Exploring COVID-19 data for Toronto, Canada

STA303/1002 Data exploration assessment

Information	Note
Name	Data exploration
Type	Type 1
Value	10%
Due	Friday, Feb 12 at 6:00 p.m. ET
Instructions and submission link	Instructions: https://q.utoronto.ca/courses/204826/assignments/415115 Submission: Via Crowdmark (link will be active once you receive an email from Crowdmark about this activity)
Late submission policy	For assessments in Type 1, late assessments will still be accepted through Crowdmark, but only if they are your first submission. You will lose 10 percentage points on the assessment, per day, with submissions accepted for up to 3 days after the due date. I.e., 72 hours after the initial due date.
Accommodations and extension policy	If you miss a type 1 assessment due to illness or a serious personal emergency, please complete **this form** within ONE week of the due date of the assignment. Upon receipt of your form, we will contact you via email within 3 business days to arrange an accommodation.

Introduction

For this assessment you will be working with the most up-to-date COVID data for the City of Toronto.

To complete this assessment, there are two documents you need to be aware of:
(1) this set of instructions (i.e., what you're currently reading), and
(2) `sta303_data-exploration_template.Rmd` a template you should save your on copy of.

The template will help you load the required data and has some helper functions for preparing your submission for Crowdmark.

In tasks 1 and 2, our aim is to create versions of the 'Cases by Day' and 'Cases by Outbreak Type and Week' graphs you can find on the Toronto COVID portal under 'Daily Status of Cases'.

In task 3 you will use data about Toronto's neighbourhoods, drawn once again from the Toronto COVID portal, this time under 'Neighbourhood Maps', as well as neighbourhood profile data from the 2016 census, from the OpenData Toronto portal.

The graphs in your final submission may look a little different from the ones in this document, as you will get the most up to date version of the data before submitting. That should be the only difference between your plots and the ones shown here.

Code last run 2021-01-20.
Daily: Data as of January 18, 2021.
Neighbourhood: Data as of January 17, 2021.

Task 1: Daily cases

Data wrangling

Prepare your data for visualization with the following data wrangling requirements:

- Start with `reported_raw`. (See the template code for the code to read the data in.)
- Your new wrangled dataset should be saved as an object called `reported`.
- Replace all NA values with 0 in the `recovered`, `active` and `deceased` columns. See the note, below.
- Make sure the `reported_date` column is in date format by explicitly overwriting it with a date version if itself. Use the `date()` function.
- Assess whether the data is currently tidy. If yes, proceed. If no, alter it to be tidy. Specifically, it needs to be in the correct format to be useful for creating the figure for this section.
- Note how "Recovered", "Active" and "Deceased" appear in the figure for this section. Make appropriate alterations to your data that makes sure these names are a) capitalized appropriately and b) will appear in the correct order in the legend of the figure for this section.

Note

Part marks will be given for any code that achieves this. For full marks, your code should have the following (filled in appropriately) as the first two lines:

```
reported <- <data> %>%
  <function from dplyr>(<function from base R that check if the column is numeric>,
  <function from tidyr that replaces NAs with what you specify>, replace=0)
```

(You don't need to use the same line breaks.)

Data visualization

Create a barchart of active, recovered and deceased cases by date. For full marks your graph must be exactly the same as the one below (except for the updated data) and must be created from the code you've written.

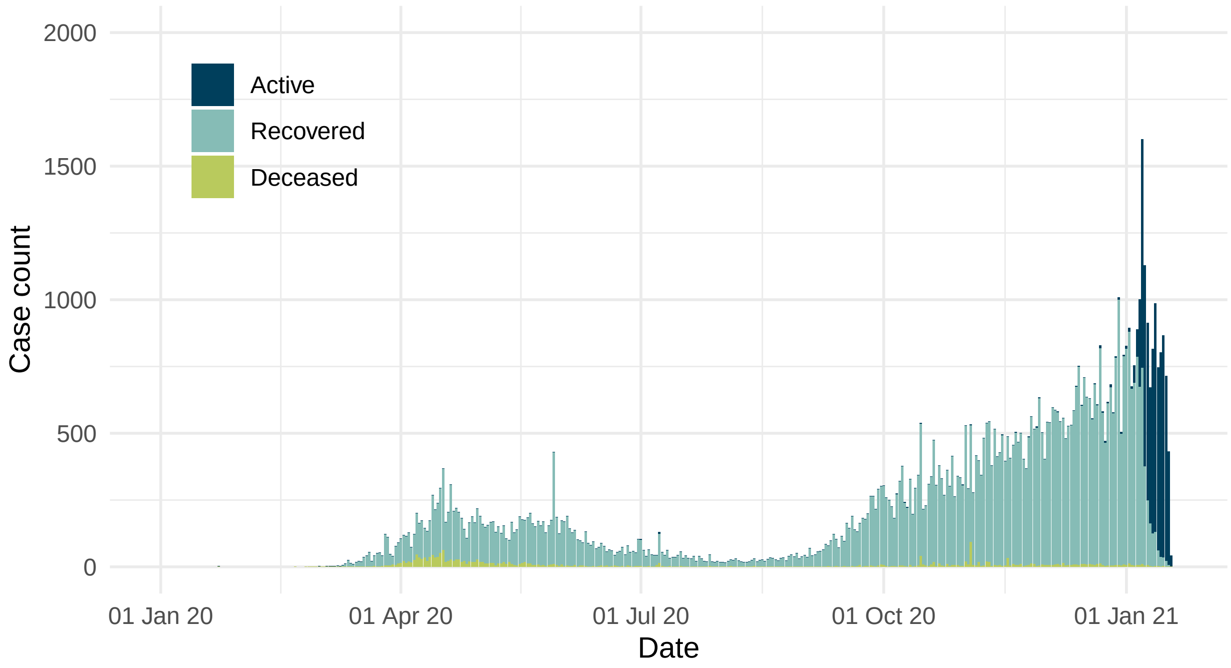
- The bar chart should be stacked, as opposed to side by side.
- The title should be: "Cases reported by day in Toronto, Canada".
- The subtitle should be: "Confirmed and probable cases".
- The axis labels must match my image (pay attention to capitalization).
- There should be small text below the graph that includes the following: "Created by: for STA303/1002, U of T" and "Source: Ontario Ministry of Health, Integrated Public Health Information System and CORES" and the date the data was last updated, set programmatically, not by hand. See tips 1, 2 and 3 below.
- The limits should be from January 1, 2020 to the present day, with the present day set programmatically, not by hand. See tip 4.
- Any warnings suppressed so they are not printed in your final document.
- No legend title.
- Legend position is set to `c(.15, .8)`.
- Active cases are coloured "#003F5C", recovered cases are coloured "#86BCB6", and deceased cases are coloured "#B9CA5D".

Notes

1. You can use \n to create a line break in text strings you are using in `ggplot`.
2. `date_daily[1,1]` will give you the data as a character string.
3. `str_c()` is useful for combining text and code to output a final text string.
4. `date("2020-01-01")` and `Sys.Date()` will be helpful.

Cases reported by day in Toronto, Canada

Confirmed and probable cases



Created by: <your name> for STA303/1002, U of T
Source: Ontario Ministry of Health, Integrated Public Health Information System and CORES
Data as of January 18, 2021

Task 2: Outbreak type

Data wrangling

- Start with `outbreak_raw`.
- Your new wrangled dataset should be saved as an object called `outbreak`.
- Make sure the `episode_week` column is in date format by explicitly overwriting it with a date version if itself. Use the `date()` function.
- Assess whether the data is currently tidy. If yes, proceed. If no, alter it to be tidy. Specifically, it needs to be in the correct format to be useful for creating the figure for this section.
- Note how "Sporadic" and "Outbreak associated" appear in the figure for this section. Make alterations to your data that make sure these names are a) correctly worded/capitalized and b) will appear in the correct order in the legend of the figure for this section.
- Create a new variable, `total_cases`, that indicates the total number of cases in the episode week, i.e., the sum of sporadic cases and outbreak associated cases.

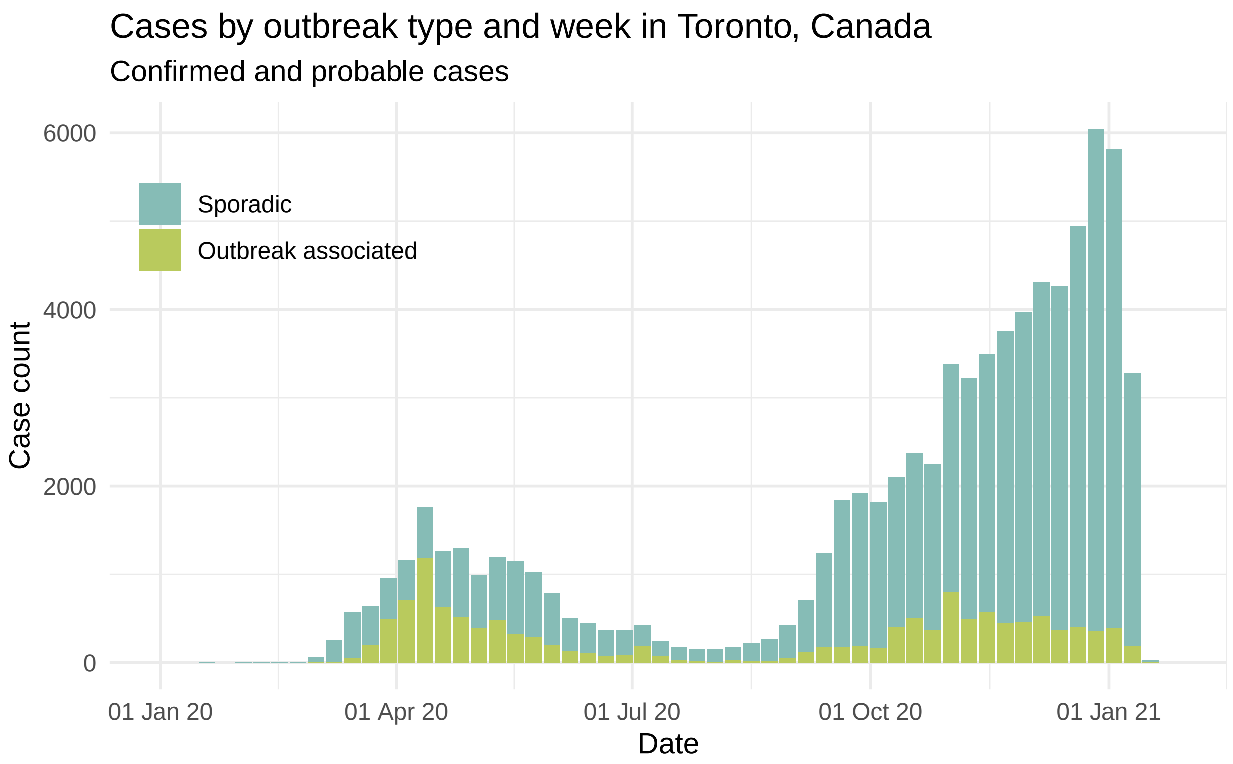
Data visualization

Create a barchart of cases by outbreak type and week. For full marks your graph must be exactly the same as the one below (except for the updated data) and must be created from the code you've written.

- The bar chart should be stacked, as opposed to side by side.
- The title should be: "Cases by outbreak type and week in Toronto, Canada".
- The subtitle should be: "Confirmed and probable cases".
- The axis labels must match my image (pay attention to capitalization).
- There should be small text below the graph that includes the following: "Created by: for STA303/1002, U of T" and "Source: Ontario Ministry of Health, Integrated Public Health Information System and CORES" and the date the data was last updated, set programmatically, not by hand. See Notes 1, 2 and 3 below.
- The x-axis labels should be formatted as day, month, year in the form "01 Jan 20". Complete the code in Note 4 to achieve this.
- The limits of the x-axis should be from January 1, 2020 to the present day + 7 days, with the present day set programmatically, not by hand. See tips 4 and 5.
- The limits of the y-axis should be from 0 up to the maximum of the `total_cases` variable you made. Set this programmatically, not by hand.
- Any warnings suppressed so they are not printed in your final document.
- No legend title.
- Legend position is set to c(.15, .8).
- Sporadic outbreak are coloured "#86BCB6", and outbreak associated cases are coloured "#B9CA5D".

Notes

1. You can use \n to create a line break in text strings you are using in `ggplot`.
2. `date_daily[1,1]` will give you the data as a character string.
3. `str_c()` is useful for combining text and code to output a final text string.
4. Complete and use this code: `scale_x_date(labels = scales::date_format("%d %b %y"), limits = <set appropriate limits>)`.
   5.`date("2020-01-01")` and `Sys.Date()+7` will be helpful.



Cases by outbreak type and week in Toronto, Canada

Confirmed and probable cases

Created by: <your name> for STA303/1002, U of T
Source: Ontario Ministry of Health, Integrated Public Health Information System and CORES
Data as of January 18, 2021

Task 3: Neighbourhoods

Data wrangling: part 1

Our goal here is to prepare a dataset that has the percentage of 18 to 64 years-olds who are classified as low income in each neighbourhood.

- Start with the `nbhood_profile` dataset.
- Your new wrangled dataset should be saved as an object called `income`.
- Filter to only include the row(s) that are relevant, i.e. the row(s) that give the percentage of 18 to 64 year-olds who are classified as low income.
- Assess whether the data is currently tidy. If yes, proceed. If no, alter it to be tidy.
- Make sure that the percentages are stored as numbers, not character strings. The function `parse_number()` may be of use to you for this.

Data wrangling: part 2

Our goal here is to use the `income` data from part 1 and merge it with the `nbhoods_shape_raw` data, that will allow us to draw a map of Toronto, with the neighbourhoods.

- Start with `nbhoods_shape_raw`.
- Your new wrangled dataset should be saved as an object called `nbhoods_all`.
- Each row of this dataset represents a neighbourhood of Toronto. Use the `AREA_NAME` variable to create a new variable called `neighbourhood_name` that removes the number in parentheses (and the space before it) to get a clean neighbourhood name. See note 1 below.
- Merge appropriately so that you have the cases per 100,000 people by neighbourhood and the percentage of 18 to 64 years-olds who are classified as being low income.
- Ensure the neighbourhoods are correctly matched. (Hint: Toronto has 140 neighbourhoods.) See note 2 below.
- Assess whether the data is currently tidy. If yes, proceed. If no, alter it to be tidy. Specifically, it needs to be in the correct format to be useful for creating the figure for this section.
- Rename your case rate variable to `rate_per_100000` if it isn't already.

Notes

1. `"\\s"` matches the space character, `"\\("` and `"\\)"` matches parentheses, and `"\\d"` by itself will match any of the digits between 0 and 9 once, and the `+` will match as many of that type as exist in a row, so "`\\d+`" would match a string of digits of any length. Putting a `$` at the end says we want this to match this pattern at the end of our character string.
2. 'City of Toronto' is not a neighbourhood, but may appear in your data, depending on how you merge. Merge and/or filter appropriately so it is not included.

Data wrangling: part 3

Our goal here is to use the `nbhood_all` from part 2 and create a new variable that indicates

- Create a new dataset called `nbhoods_final` from `nbhoods_all`.
- Create two new variables `med_inc` and `med_rate` that are the median percentage of 18 to 64 year-olds who are classified as low income and the median case rate per 100,000 people, across Toronto's 140 neighbourhoods, respectively.

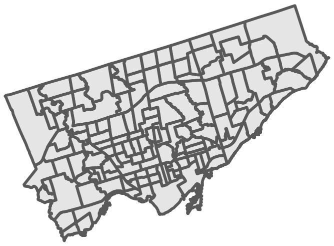
- Create a new variable called `nbhood_type` that takes the following values under the following conditions:
  - "*Higher low income rate, higher case rate*" for neighbourhoods where the percentage of 18 to 64 year-olds who are low income is **greater than or equal to** the median percentage for Toronto neighbourhoods and the number of cases per 100,000 people is also **greater than or equal to** the median.
  - "*Higher low income rate, lower case rate*" for neighbourhoods where the percentage of 18 to 64 year-olds who are low income is **greater than or equal to** the median percentage for Toronto neighbourhoods and the number of cases per 100,000 people is **lower than** the median for Toronto neighbourhoods.
  - "*Lower low income rate, higher case rate*" for neighbourhoods where the percentage of 18 to 64 year-olds who are low income is **less than** the median percentage for Toronto neighbourhoods and the number of cases per 100,000 people is also **greater than or equal to** the median.
  - "*Lower low income rate, higher case rate*" for neighbourhoods where the percentage of 18 to 64 year-olds who are low income is **less than** the median percentage for Toronto neighbourhoods and the number of cases per 100,000 people is also **lower than** the median for Toronto neighbourhoods.

Data visualization

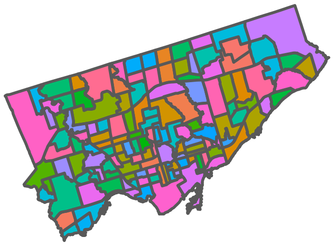
Create three maps:
1. One showing the percentage of 18 to 64 year-olds that are classified as low income in each of Toronto's 140 neighbourhoods.
2. One showing the number of COVID-19 cases per 100,000 people in each neighbourhood.
3. One showing each neighbourhood coloured by its case rate and low income combination (`nbhood_type`).

Use the following 'starter code' to help you. You can add a `fill` command within an `aes()` command in `geom_sf()` (simple feature geometry) to colour each neighbourhood. These example maps should NOT appear in your final submission.

```
# Basic code
ggplot(data = nbhoods_final) +
  geom_sf() +
  theme_map()
```



```
# Example with each neighbourhood coloured
ggplot(data = nbhoods_final) +
  geom_sf(aes(fill = neighbourhood_name)) +
  theme_map() +
  theme(legend.position = "none")
```

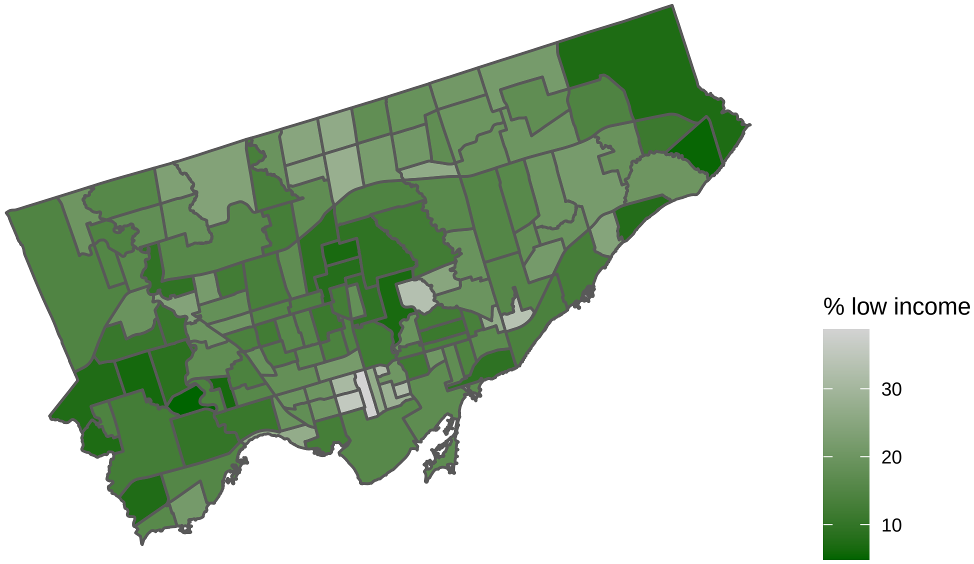


Notes

- For the first figure, you can set the colour with the following line of code: `scale_fill_gradient(name= "% low income", low = "darkgreen", high = "lightgrey")`.
- For the second figure, you can use similar code to the above, but the colours should go from `white` (low) to `darkorgange` (high).
- For the third figure, use the palette called 'Set1' from R colour brewer. (Hint: `scale_fill_brewer()`).

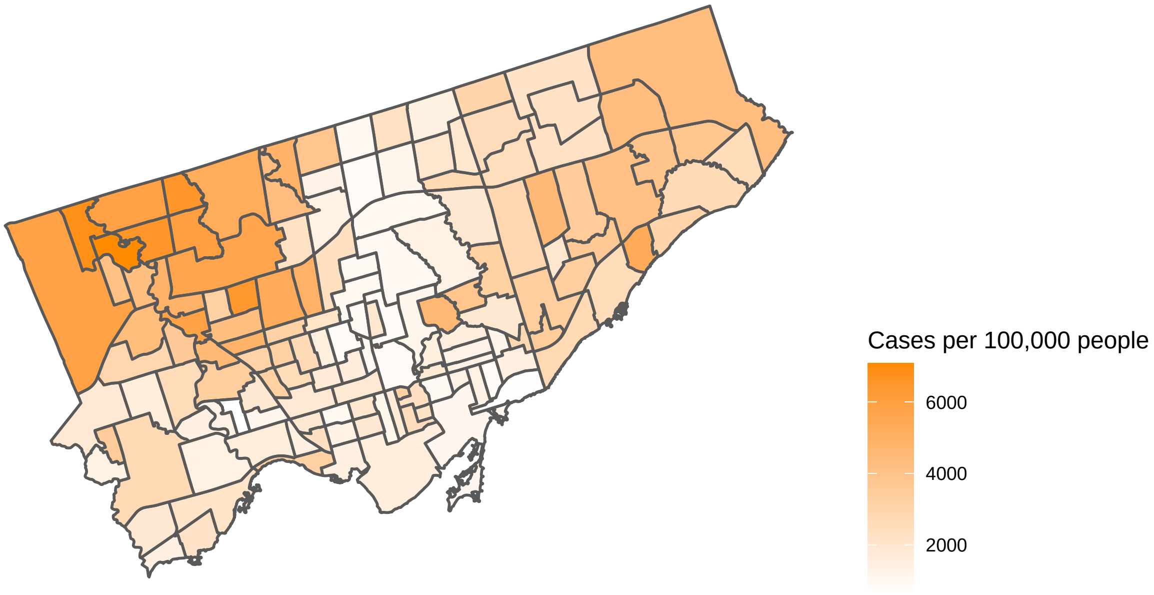
Percentage of 18 to 64 year olds living in a low income family (2015)

Neighbourhoods of Toronto, Canada



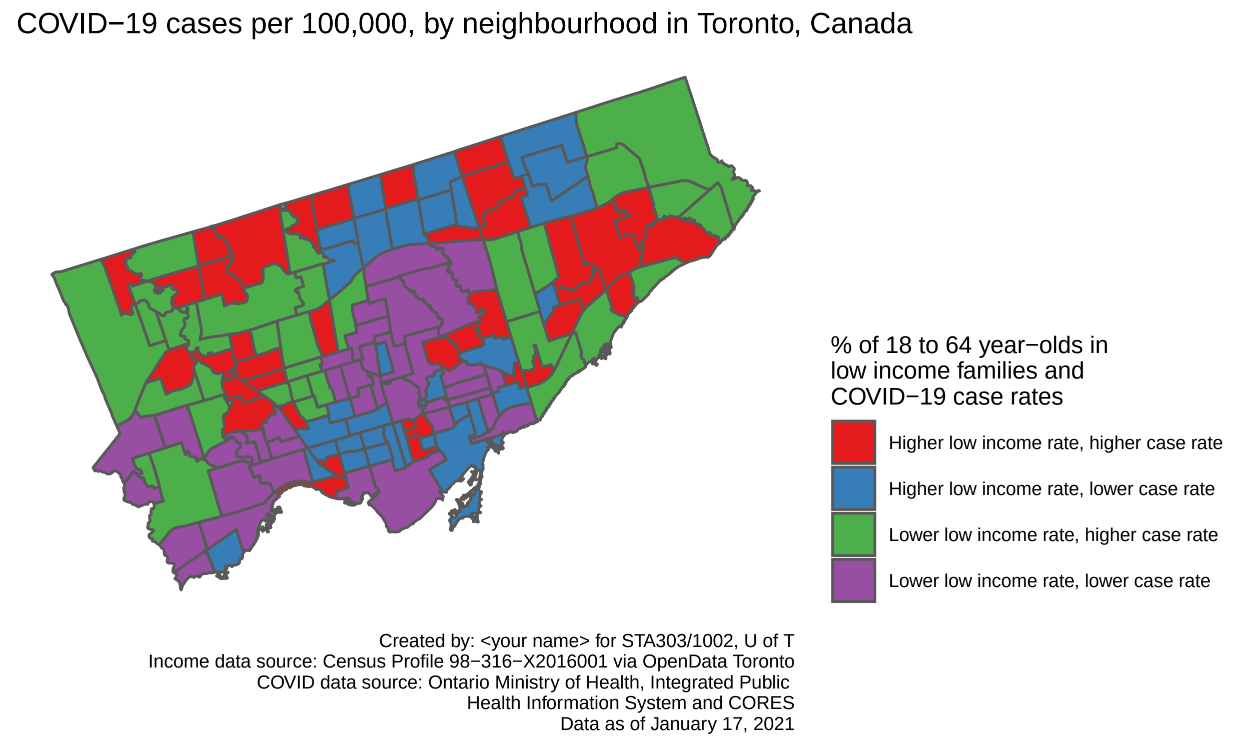
Created by: <your name> for STA303/1002, U of T
Source: Census Profile 98−316−X2016001 via OpenData Toronto
Data as of January 17, 2021

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



Created by: <your name> for STA303/1002, U of T
Source: Ontario Ministry of Health, Integrated Public Health Information System and CORES
Data as of January 17, 2021

COVID−19 cases per 100,000, 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: <your name> 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 17, 2021