CSC110 Project Proposal: Pandemic impact on employment rates of Canadian Provinces and Territories

Lemeng Dai, Arthur Toshio Iliescu, Jiaxin Li, Nurul Maisarah Zulkefli

Tuesday, December 14, 2021

Problem Description and Research Question

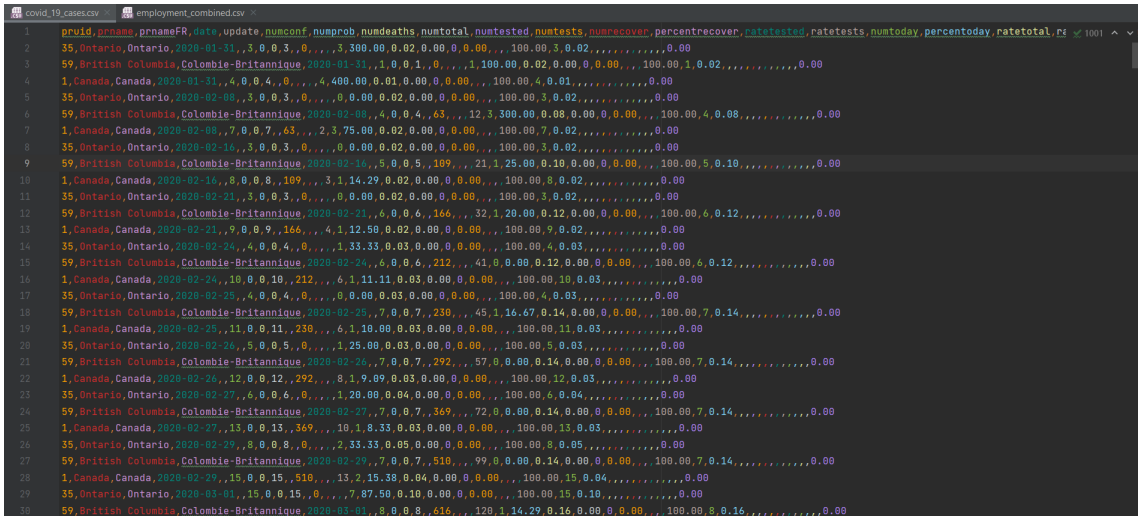
On January 5th of 2020, the World Health Organization publicly announced news of the covid-19 outbreak in Wuhan, Hubei Province, China(World Health Organization, 2020). Afterwards, the virus quickly spread across many countries and became a global pandemic. The fast spread of the virus has infected hundreds of millions and has caused 5 million deaths globally(The Visual and Data Journalism Team, 2021).

In response to the sudden pandemic, many governments have put restrictions and policies in place to best limit the spread of the virus(Ritchie et al., 2021). Although these restrictions are implemented to stop the spread of the pandemic, they can also cause issues with employment. In Canada, we experienced many lockdowns and restrictions(Hagan, 2021). This had negative effects on businesses and employees. In Canada, the number of those who have been unemployed for over 6 months has doubled to 389, 000 since February 2020(Press, 2021). Furthermore, many retail stores have remained closed, or are operating at a fraction of the capacity they would have pre-pandemic(Retail Council of Canada, 2021). To make up for the financial loss of the pandemic, or to satisfy restrictions and policies, many employers ended up laying off their employees. For example, according to Maclean's during the starting few months of the pandemic in Canada, Ford Motors Co ceased all of their production, temporarily laying off 6,900 workers(Maclean's, 2020). Many people found themselves without work, in a time where having a stable flow of income is more important than ever.

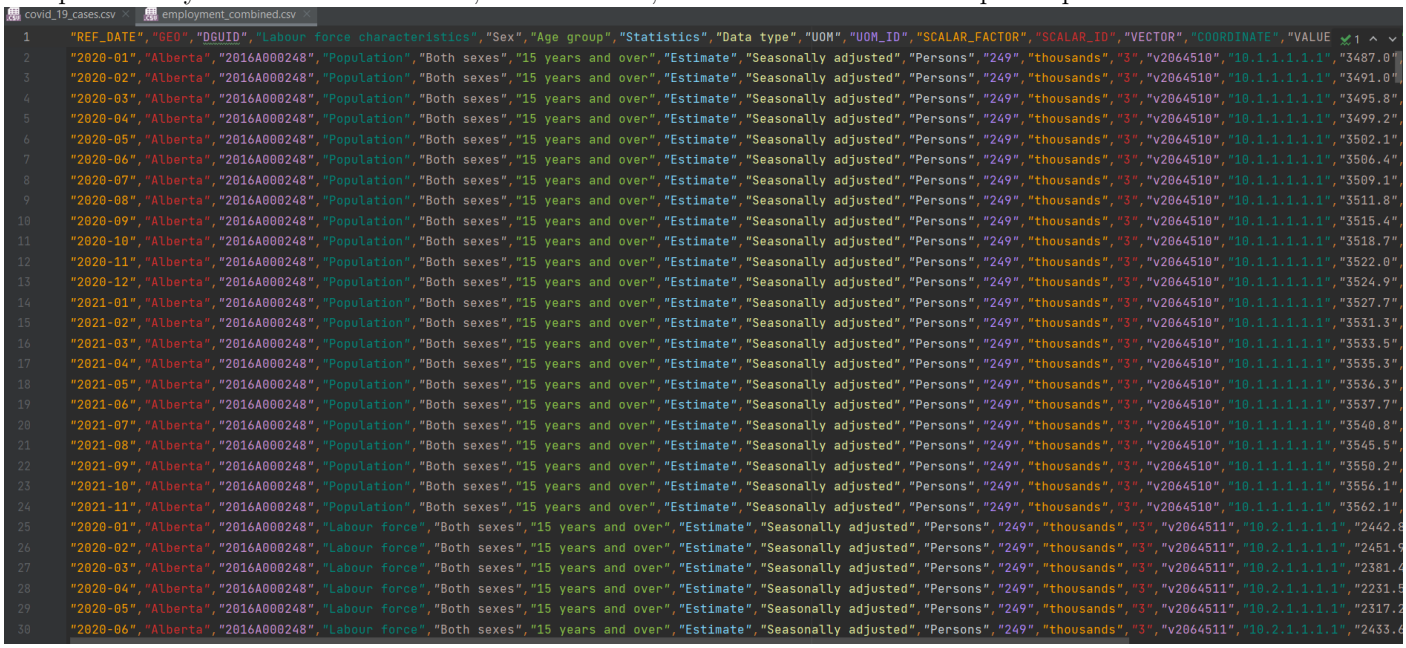
As people living in or staying in Canada, any situation that affects the entire country is of great concern. We want to show the effect of the number of monthly increased cases, the number of monthly increased tests, and case rate on the employment rate to highlight the Canadians affected as well as which Canadian provinces were hit the hardest. This brings us to our research question: **How has the pandemic affected the employment rate of each Province in Canada?**

Data set Description

Data set 1: COVID-19 cases by province: https://health-infobase.canada.ca/covid-19/epidemiological-summary-covid-19-cases.html?stat=num&measure=total&map=pt#a2
Source: government website (Canada, P. H. A. of. (2021, May 28).
Format: csv
Used columns: prname, date, numtoday, numtotal, numconf, numtests, ratetotal



Data set 2: Monthly Employment of Canada and each province: https://www150.statcan.gc.ca/t1/tbl1/en/tv.action?pid=1410028701&pickMembers%5B0%5D=1.1&pickMembers%5B1%5D=3.1&pickMembers%5B2%5D=4.1& pickMembers%5B3%5D=5.1&cubeTimeFrame.startMonth=01&cubeTimeFrame.startYear=2020&cubeTimeFrame.endMonth=09&cubeTimeFrame.endYear=2021&referencePeriods=20200101%2C20210901
Source: government website (Government of Canada, Statistics Canada. (2021, November 5).
Format: csv
Used columns: 'REF_DATE', 'GEO', "Labour force characteristics", "Data type", 'VALUE'
The picture only shows data for Canada, in the website, we can select data set for specific province.



Computational Overview

Part1: covid19_cases.py

1. A dataclass called `CasesData` is created.
2. A function called `load_data` is written to extract the data we will use from data sets (as mentioned in Data set Description).
3. The function called `cases_per_month` allows us to calculate the total number of increased COVID-19 cases in a given month of a given year in a given province of Canada.
4. We used `cases_per_month` as a helper function to write `cases_per_month_province` which allows us to get a dictionary matching the month in a calendar year to the number of increased cases for the given province.
5. We used `cases_per_month_province` as a helper function to write `total_cases_per_year` which returns a dictionary in which provinces are mapped to the years and months to the total number of new cases of the respective month.
6. A function called `return_data_cases` is written to run `cases_per_month_province` with our data set as the input for all provinces in Canada in a given period of time (from January 2020 to November 2021). This function will be used to do predictions based on our data set.
7. The function called `total_tests_per_month` allows us to calculate the total number of tests tested in a given month of a given year in a given province of Canada.
8. We used `total_tests_per_month` as a helper function to write `per_month_province_test` which allows us to get a dictionary matching the month in a calendar year to the number of tests tested for the given province.
9. We used `per_month_province_test` as a helper function to write `total_tests_per_years` which returns a dictionary in which provinces are mapped to the years and months to the total number of tests tested of the respective month.
10. A function called `return_data_tests` is written to run `total_tests_per_years` with our data set as the input for all provinces in Canada in a given period of time (from January 2020 to November 2021). This function will be used to do predictions based on our data set.
11. The function called `average_rate_per_month` allows us to calculate the average of rate (total number of cases in a province divided by the population of the province times 100000) in a given month of a given year in a given province of Canada.
12. We used `average_rate_per_month` as a helper function to write `rate_per_month_province` which allows us to get a dictionary matching the month in a calendar year to the average rate for the given province.
13. We used `rate_per_month_province` as a helper function to write `rate_per_years` which returns a dictionary in which provinces are mapped to the years and months to the average rate of the respective month.
14. A function called `return_data_rate` is written to run `rate_per_years` with our data set as the input for all provinces in Canada in a given period of time (from January 2020 to November 2021). This function will be used to do predictions based on our data set.
15. Function `cases_map_provinces` is created to return a dictionary that maps 'provinces' to the list of given provinces and maps 'cases' to the number of total new cases in the given month of the given year.
16. Function `return_case_for_map` is a short-cut that helps us to call `cases_map_provinces` with our data set file as inputs and the list of all provinces in Canada as variable provinces. This function will be used to generate DataFrame in the visualization.py.

Part 2: employment_rate.py

1. A dataclass called `EmploymentData` is created.
2. A function called `load_data_employment` is written to extract the data we will use from data sets (as mentioned in Data set Description).
3. The function called `population_num` allows us to find the population of people in thousands of the province during set month and year.
4. The function `employment_num` is created to help us to find the number of people employed in thousands of the province during set month and year.
5. A function called `employment_rate_per_month` is created by using `population_num` and `employment_num` as a helper function which helps us to calculate the number of employment rate for set month and year of the province.
6. A function called `employment_rate_to_date` is created by using `employment_rate_per_month` as a helper function. This function creates a dictionary mapping each date in months and years to the employment rate of province.
7. `employment_rate_total` function is created by using a helper function `employment_rate_to_date` that allows us to create a dictionary that maps each province to another dictionary with date mapped to employment rate for that province.
8. By creating `employment_rate_total`, it helps us to create another function called `employment_rate_month_province` create a dictionary mapping province to its employment rate in the given month of the given year.
9. Function `return_emp_for_map` is created by using the helper function `employment_rate_month_province` and `load_data_employment` that helps us to create a dictionary mapping province to its employment rate in the given month of the given year.It is a short-cut that helps us to call `employment_rate_month_province` with our data set file as inputs and the list of all provinces in Canada as variable provinces. This function will be used to generate DataFrame in the visualization.py.

Part 3: visualization.py

1. Downloading a `GeoJSON` file containing the geometry information for CA provinces ("GeoJSON", n.d.).
2. The GeoJSON file is uploaded to our group Github repository and urlopen module is imported to open the URL of the GeoJson file. Then, the file is stored in a variable called provinces.
3. `choropleth_map_emp` is a function that returns a choropleth map containing information about the employment rate in provinces of Canada in the given month of the given year in explorer. In its function body, pandas.DataFrame is called to transform the dict into DataFrame type("pandas.DataFrame", n.d.). `plotly.express` from the module `plotly` is used to generate the choropleth map("Choropleth Maps in Python", n.d.).
4. `fig.update_geos` is called to limit the visible range of map to provinces of Canada.
5. The implementation of `choropleth_map_cases` is similar to `choropleth_map_emp`.

Part 4: prediction.py

1. Get a dictionary in which the provinces mapped to the date mapped to the total covid 19 cases per month.
2. Get a dictionary mapping each province to another dictionary with date mapped to employment rate for that province.
3. Using the dictionary in the last step, calculate the differences in employment rate between each pair of consecutive months. Then create a dictionary in which the provinces mapped to the date mapped to the the differences in employment rate between each pair of consecutive months.
4. Transform the covid 19 cases dictionary and differences in employment rate dictionary into DataFrame type.

5. Transform the two dataframes into one dictionary that maps province name to a dataframe that contains this province's monthly covid 19 cases and the difference in employment rate.

6. Using `pd.DataFrame` from pandas library to transform dictionary of data into DataFrame type ("pandas.DataFrame", n.d.).

7. Then build model using the dictionary in the last step. Using `sklearn.model_selection.train_test_split` from `scikit-learn` to choose 80% the data of each province to train the model and 20% of the data to test the model later on ("sklearn.model_selection.train_test_split", n.d.).

8. Using `plotly.subplots.make_subplots` and `plotly.graph_objects.Scatter` to draw a scatter diagram with the number of covid 19 cases as independent variable and employment rate as dependent variable for each province("Scatter Plots in Python", n.d.).

9. Using `scikit-learn` to find the model that fits mapping for each province. Used `linear_model.LinearRegression` which is linear model and `sklearn.tree.DecisionTreeRegressor` which is non-linear model. ("Supervised learning: Predicting an output variable from high-dimensional observations", n.d.).

10. Using `model.predict` from `scikit-learn` to perform the model on the 20% testing data and get the predicted results to compare it with the actual data using `model.score` from `scikit-learn` to get the score of how well the model fits ("Supervised learning: Predicting an output variable from high-dimensional observations", n.d.).

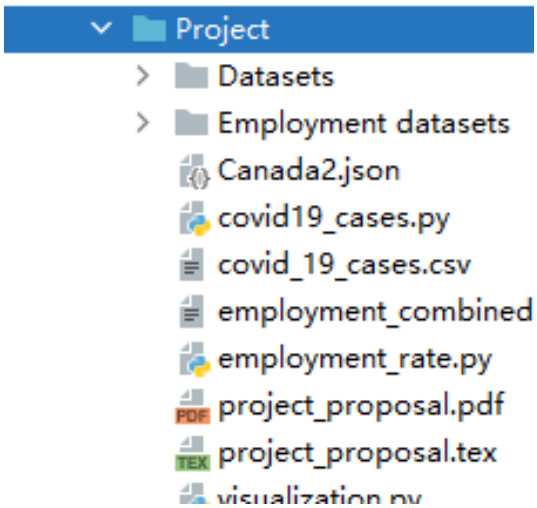
11. Also used `sklearn.metrics.mean_squared_error` to get the mean squared error of the predicted result and the actual testing data.

12. Return a dictionary that maps the province name to a tuple that contains the score and and the mean squared error.

13. Do the same thing for the number of covid 19 tests issued and rate of total covid 19 cases to the total population.

Instructions

Download the requirements.txt we submitted on Markus and you can download all the modules that have been used in our project through this file.
For the data sets save the file in the same folder as the code, marking the folder as route directory. Below is an example of how the data sets would be saved.



1. We attached a github link to the raw data of the downloaded data set because the statcan150 website sometimes goes down for unknown durations.
The first data set for the covid-19 cases data can be found at: https://raw.githubusercontent.com/LemengDai/Final_Project/main/covid_19_cases.csv or alternatively: https://health-infobase.canada.ca/covid-19/epidemiological-summary-covid-19-cases.html? stat=num&measure=total&map=pt#a2.
The second github link to the raw data can be found at: https://raw.githubusercontent.com/LemengDai/Final_Project/main/employment_combined.csv or alternatively: https://www150.statcan.gc.ca/t1/tbl1/en/

tv.action?pid=1410028701&pickMembers%5B0%5D=1.1&pickMembers%5B1%5D=3.1&pickMembers%5B2%5D=4.1&pickMembers %5B3%5D=5.1&cubeTimeFrame.startMonth=01&cubeTimeFrame.startYear=2020&cubeTimeFrame.endMonth=09&cubeTimeFrame.endYear=2021&referencePeriods=20200101%2C20210901.

2. Next after downloading the data sets and saving them in the same file as the main code, you may run the main.py file.

3. After runing the main.py file, for the line `visualization.main()`, you are expected to see 8 pages open in your browser. Each two of them is a pair of choropleth maps. One of them shows the number of new COVID-19 cases of a given month in a given year for provinces in Canada. The other one shows that employment rate of a given month in a given year for provinces in Canada. On the left top of each page, there is a title that tells what is showing in the choropleth map. You can try other combination of month and year by calling `visualization.choropleth_map_cases(month, year)` or `visualization.choropleth_map_emp(month, year)` in the Python Console. Note that month is limited to one of [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12] and year is limited to one of [2020, 2021]. Note that these functions does not work when month is 12 and year is 2021 because we do not have data for December 2021 yet.

4. After running the main.py file, for the line `prediction.main()` you will see three graphs in the browser, each shows the relationship between the number of covid 19 cases and the difference of employment rate in Canadian provinces, the relationship between the number of covid 19 tests issued and the difference of employment rate in Canadian provinces, and the relationship between the rate of total COVID-19 cases to the total population and the difference of employment rate in Canadian provinces. It also returns a dictionary that contains three dictionaries mapping the province name to a tuple that contains the score and and the mean squared error of the model for each of the three relationships.

Changes to computational plan

Compared to our computational plan in our proposal, our computational overview is more detailed. In the proposal, we only gave a vague idea of what our variables meant rather than specify which variables in the data sets we were going to use. Not all steps in our computational plan were done in our actual program. For example, we did not calculate the monthly average of the newly diagnosed cases per day. Rather, we used the total number of newly diagnosed cases. This is because taking the average number will not make a big difference when doing the linear regression with another variable as the employment rate.

In our computational plan we also did not mention using the rate of covid-19 cases or the covid-19 tests. Initially we had only thought we were going to compare the covid-19 cases to the employment rates in each province. While working on the project however, we decided it would be better to add more comparisons and thus we came up with those two additional comparisons.

The functions in our computational plan were also slightly different. For instance, in part one of our computational plan we had said we were going to have two functions, one to derive the data for a specific month/day and one that adds up the number of cases in each day and returns the mean total number of cases in each month. However as mentioned earlier, we changed the monthly average to new cases per day. For each type of data(covid-19 cases, tests, and rate of tests) we ended up having three functions each for a total of 9 functions to gather the data. The three functions were similar for all three types of data. The first function gathered the data per month, the second function mapped the date of the data to the data, and the third function mapped the province to its respective dates mapped to data. An example of the output of the third function looks like: `{'Ontario': {(2020, 1): 40}}`. Finally for each type of data we had a function that would gather the data called return_data. In total each type of data had three functions for gathering data and one additional function to return the data using the other three functions as helper functions.

We used different `plotly` methods to plot the subplots in the prediction.py file. At first, we did not thought of making subplots, so instead used `plotly.express.scatter`. But since we don't want many webpages of graphs, we decided to put them all in one graph. So we used `plotly.make_subplots` and `plotly.graph_objects.Scatter`. Moreover, we did not plan to calculate the mean squared error between the predicted value and the actual value. But we found out that mean squared value is frequently used to indicate whether a model fits well. So we decided

to include it. We used `sklearn.metrics.mean_squared_value`.

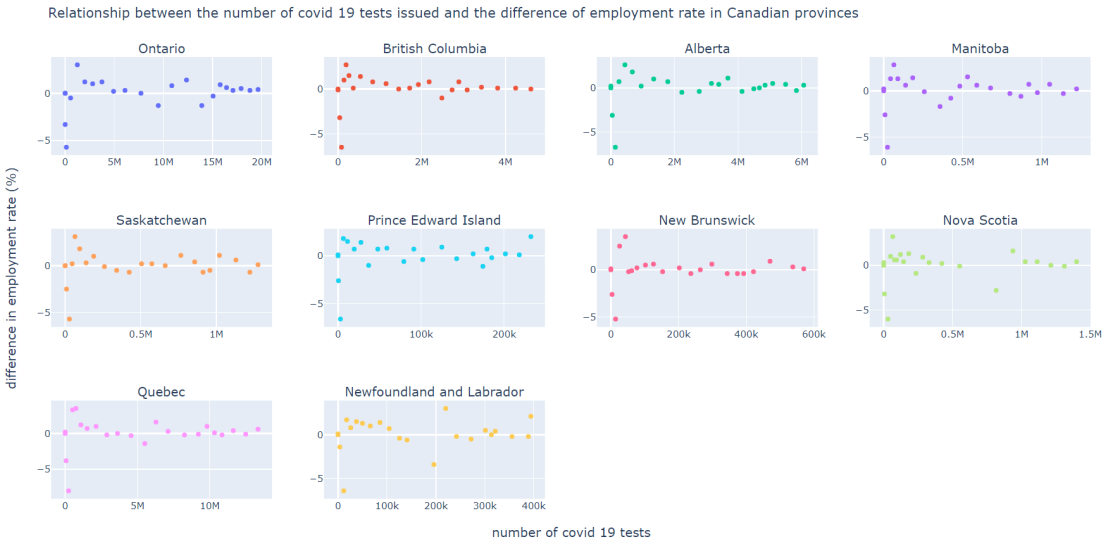
Discussion

In our program we looked at the number of cases, the number of tests, and the rate of cases to analyze the correlation between the three and the employment rate in each province. At the start of the project we unanimously thought that there would be a correlation between the three variables and the employment rate.

We tried to build both linear and non-linear models to fit the relationships between covid-19 cases, tests, and rates of tests, to employment rate in each province. However, both give very large mean squared errors and poor scores, indicating that it's hard to find a simple model that describes the correlation. The program returned the results in the following graphs:

Number of covid 19 cases and difference of employment rate

Number of covid 19 tests and difference of employment rate



Rate of total covid 19 cases and difference of employment rate

For the first graph. When the first initial cases started to appear there is a noticeable decrease in the employment rate, however as the pandemic progressed and the cases increased further this correlation did not hold. Although if we were to isolate the period of time to be the start of the pandemic meaning the first half of 2020, then we would be able to find that as the number of cases increased from 0 the employment rate did decrease. However, this trend was only seen in the first few months of the data for 2020. After looking at this data as well as referring to our research done for the proposal, this data perfectly aligns with how it should as there were many restrictions at the start of the pandemic dropping the employment rate, but as the pandemic continued these restrictions were slowly lifted allowing the employment rate to return.

In the second and third set of graphs we saw a similar result as the first. Both maintained a correlation only for the starting few months of 2020 when the pandemic began.

From the information we can conclude that all three maintained a correlation at the beginning of the pandemic. However, when considering if any of the three had a strong correlation throughout the pandemic the answer to our research question would be no.

There were a few obstacles related to the data sets which we encountered. First the Statistics Canada website was unreliable as it would periodically be unavailable for access. Also, one of our data sets contained the data for covid 19 cases in all Provinces and Territories, however our other dataset on employment rate only recorded data from Provinces. Thus, we had to make a change in our project to look at the only the Provinces of Canada. Another issue we faced is that in our data sets some of the data columns would be left blank, presumably because none was required for that specific section. However, when writing code to convert the string data into integers or floats, these empty data entries would be considered empty string. This would cause an error when we were trying to write our `load_data` function. Finally we faced issues when we were writing our functions to calculate the data from the data sets. Because both data sets only recorded data up until November we had to omit the month of November for the year 2021 in each function that took data from the data set.

For further exploration we could look more into data on government restrictions related to jobs and when they were implemented. This would probably give us a more clear and consistent correlation with the employment rate for each province. We could also examine the relationship between the covid-19 cases, tests, and rate of cases compared to the restrictions which were implemented.

In conclusion the answer we received from our data was that there was no direct correlation when comparing covid-19 cases, tests, and rates of tests, to employment rate in each province. The only relations we saw were limited to the start of the pandemic in the first quarter of 2020. Although the results were different from our expectations, now looking back with research on restrictions we can come to understand why this came to be.

References

"Archived: Who Timeline - Covid-19." World Health Organization, World Health Organization, 27 Apr. 2020, https://www.who.int/news/item/27-04-2020-who-timeline—covid-19.

"Choropleth Maps in Python". (n.d.). Retrieved November 04, 2021, from https://plotly.com/python/choropleth-maps/

"Covid-19 Requirements for Retailers by Region." Retail Council of Canada, Retail Council of Canada, 27 Oct. 2021, https://www.retailcouncil.org/coronavirus-info-for-retailers/provincial-covid-19-resources-and-updates/.

"Covid-19: Canada Layoff Tracker." Macleans.ca, St. Joseph Communications, 30 Apr. 2020, https://www.macleans.ca/economy/covid-19-canada-layoff-tracker/.

"GeoJSON". (n.d.). Retrieved November 04, 2021, from https://doc.arcgis.com/en/arcgis-online/reference/geojson.htm

Hagan, Shelly. "Canada Suffers Second Labour Market Setback Ahead of Reopening - BNN Bloomberg." BNN, Bell Media, 4 June 2021, https://www.bnnbloomberg.ca/canada-suffers-second-labour-market-setback-ahead-of-reopening-1.1612591.

"pandas.DataFrame". (n.d.). Retrieved November 04, 2021, from https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.html?highlight=dataframe#pandas-dataframe

Press, Jordan. "Canada Adds 157,000 Jobs in September, Returning to Pre-Pandemic Levels." CTVNews, Bell Media, 8 Oct. 2021, https://www.ctvnews.ca/business/canada-adds-157-000-jobs-in-september-returning-to-pre-pandemic-levels-1.5616099.

Ritchie, Hannah, et al. "Coronavirus Pandemic (COVID-19)." Our World in Data, Global Change Data Lab, 3 Nov. 2021, https://ourworldindata.org/policy-responses-covid#workplaces-closures.

"Scatter Plots in Python". (n.d.). Retrieved November 04, 2021, from https://plotly.com/python/line-and-scatter/

"sklearn.model_selection.train_test_split". (n.d.). Retrieved November 04, 2021, from https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html

"Supervised learning: Predicting an output variable from high-dimensional observations". (n.d.). Retrieved November 04, 2021, from https://scikit-learn.org/stable/tutorial/statistical_inference/supervised_learning.html

The Visual and Data Journalism Team. "Covid Map: Coronavirus Cases, Deaths, Vaccinations by Country." BBC News, BBC, 1 Nov. 2021, https://www.bbc.com/news/world-51235105

Covid-19 Daily Epidemiology update. Canada.ca. Retrieved November 5, 2021, from https://health-infobase.canada.ca/covid-19/epidemiological-summary-covid-19-cases.html? stat=num&measure=total&map=pt#a2.

Statistics Canada. Table 14-10-0287-01 Labour force characteristics, monthly, seasonally adjusted and trend-cycle, last 5 months. Retrieved November 5, 2021 https://doi.org/10.25318/1410028701-eng

Labour force characteristics, monthly, seasonally adjusted and trend-cycle, last 5 months. , from https://www150.statcan.gc.ca/t1/tbl1/en/tv.action?pid=1410028701&pickMembers%5B0%5D=1.1&pickMembers%5B1%5D=3.1&pickMembers%5B2%5D=4.1&pickMembers%5B3%5D=5.1&cubeTimeFrame.startMonth=01&cubeTimeFrame.startYear=2020&cubeTimeFrame.endMonth=09&cubeTimeFrame.endYear=2021&referencePeriods=20200101%2C20210901.