CSC110 Project Proposal: Analyzing and Comparing Carbon Emission Data in Regions of Canada from 1990-2018

Maureen Navera

December 2020

Introduction

As times are changing, the earth's climate is suffering at an alarming rate. Though there are many factors, both naturally occurring or caused by humans, that influence climate change, there is one universally known fact: carbon emission from greenhouse gasses are the main cause of the earth's alarming change in temperature (1). Greenhouse gasses trap heat within the earth's atmosphere, thus heating the earth as more and more are released due to human influence. We release these harmful chemicals into the atmosphere through burning fossil fuels and other non-renewable resources for energy. Industries utilize non-renewable resources due to their inexpensiveness and abundance; though these resources seem convenient, the repercussions from using them are starting to become more prominent and worrying.

According to the Government of Canada, in 2016, Canada's greenhouse gas emissions made up 1.5% of global greenhouse gas emissions. This was the result of a steady increase in emissions from 2005 to 2016 (2). As the impacts of climate change are becoming more and more prevalent, one would expect major industries to attempt to put an end to releasing as many carbon emissions as they do. The impact that different industries have on the increasing rate of carbon emission from Canada sparks curiosity and encourages us to ask **how carbon emissions produced by major industry categories in different regions of Canada have changed throughout several years**. With this information, we can compare and contrast the amount of carbon emissions being produced by several regions of Canada and to understand how these regions have increased or decreased the amount of carbon emissions released throughout the years. The observed trends will clearly indicate whether a region of Canada has been cautious of the changing climate and if they are working to put an end to it by having lower carbon emissions from its most prominent industries or contributing to it.

Dataset Description

The datasets used in this project are obtained from The Government of Canada's Official Greenhouse Gas Inventory called 'GHG_IPCC_Can_Prov_Terr' as a CSV file (2). This original dataset contains the annual data of many different chemical emissions along with the total amount of carbon dioxide equivalent (represented by 'CO_2eq') of those chemicals that have been produced by not just Canada as a whole, but every province and territory per industry. This original dataset contains information spanning from 1990 - 2018 and documents various chemical emissions from a number of industries for each province in Canada and Canada itself. Due to the size and complexity of this dataset, I have decided to only look at three industries for this project: Energy, Oil and Gas Extraction, and Transport. I have also decided to solely analyze just the carbon dioxide equivalent emissions (CO_2eq) of these industries instead of all of the other chemical emission information from the dataset. By breaking down the original dataset, I created 11 smaller datasets, each representing Canada's provinces and one representing Canada's data from the original.

To explain the new datasets in a more visual manner:

The new datasets all have the same structure and elements:

- Year - ranged from 1990 to 2018
- Region - the province (or Canada)
- Category ID

  - Three categories were selected out of 83 categories. The categories chosen:
    * 100 - Energy
    * 130 - Oil and Gas Extraction
    * 200 - Transport
- CO_2eq - the total amount of carbon dioxide equivalent emissions in metric tonnes

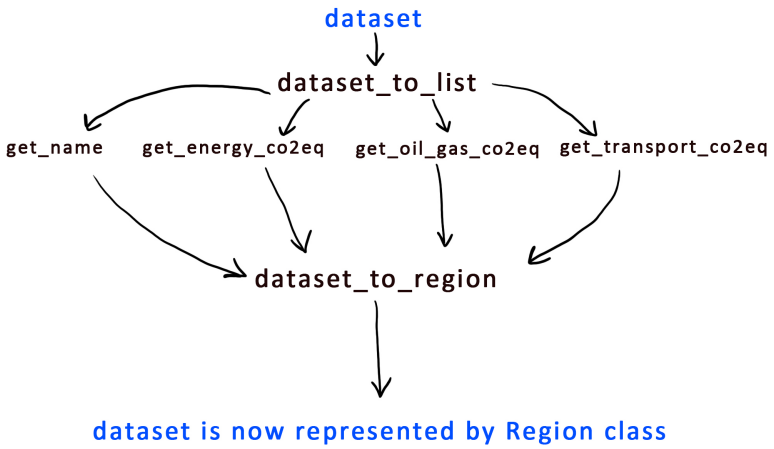
The following new datasets that have been extracted from the original are as follows, with 'given industries' referring to the energy, oil and gas extraction, and transport industries.:

Dataset Name	**Description**	**Format**
BC	Contains the CO_2eq data for British Columbia from years 1990-2018 for the given industries	.txt
AC	Contains the CO_2eq data for Alberta from years 1990-2018 for the given industries	.txt
MB	Contains the CO_2eq data for Manitoba from years 1990-2018 for the given industries	.txt
SK	Contains the CO_2eq data for Saskatchewan from years 1990-2018 for the given industries	.txt
ON	Contains the CO_2eq data for Ontario from years 1990-2018 for the given industries	.txt
QC	Contains the CO_2eq data for Quebec from years 1990-2018 for the given industries	.txt
NB	Contains the CO_2eq data for New Brunswick from years 1990-2018 for the given industries	.txt
NL	Contains the CO_2eq data for Newfoundland from years 1990-2018 for the given industries	.txt
NS	Contains the CO_2eq data for Nova Scotia from years 1990-2018 for the given industries	.txt
PE	Contains the CO_2eq data for P.E.I from years 1990-2018 for the given industries	.txt
CA	Contains the CO_2eq data for Canada from years 1990-2018 for the given industries	.txt

Computational Overview

In order to represent the complex datasets in my program, I decided to create a Region dataclass which stores the name of the region represented by that dataset, a list of floats that represent the CO_2eq emissions from the energy industry, a list of floats that represent the CO_2eq emissions from the oil and gas extraction industry, and a list of floats that represent the CO_2eq emissions from the transport industry. Therefore the region class has 4 instant atributes represented by the variable names : name, energy_co2eq, oil_gas_co2eq, and transport_co2eq. These lists representing CO_2eq are in order corresponding to the list of years.

The following algorithm is used to represent any dataset as a Region:



- dataset_to_list is a function that converts the .txt file into a list of strings using the split() method.
- get_name is a function which uses dataset_to_list as a helper function and returns the name of the region which the dataset represents

- get_energy_co2eq uses dataset_to_list as a helper function and returns a list of CO_2eq emission values for the energy industry of that region
- get_oil_gas_co2eq is a function which uses dataset_to_list as a helper function and returns a list of CO_2eq emission values for the oil and gas extraction industry of that region
- get_transport_co2eq is a function which uses dataset_to_list as a helper function and returns a list of CO_2eq emission values for the transport industry of that region
- dataset_to_region uses the lists from the three functions above and the string returned from get_name to represent that dataset as a Region.

In the create_graphs module, many functions work together to plot 3 different types of graphs:

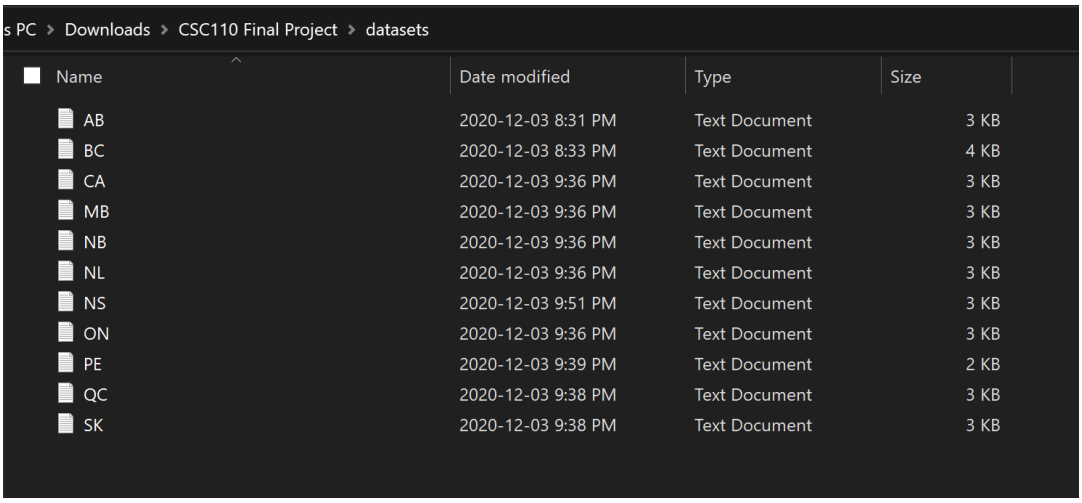
- plot_co2eq which plots the CO_2eq data per year using a plotly bar graph for the given region. This function takes in a Region class as its parameter and takes its (industry)_co2eq list attributes as the y values and its year attribute as the x value.
- plot_percent_change which plots the percent change of the CO_2eq data of the region after each year. This function displays the graph using plotly scatter plot with connected lines. This function's parameters include a Region class object and utilizes three helper functions:
  - energy_percent_change which returns a list of the percent change in the energy industry CO_2 emissions per year for the given region.
  - oil_gas_percent_change which returns a list of the percent change in the oil and gas extraction industry CO_2 emissions per year for the given region.
  - transport_percent_change which returns a list of the percent change in the transport industry CO_2 emissions per year for the given region.
- the three compare_¡industry¿ functions where ¡industry¿ is either energy, oil_gas, or transport. These functions display percent change data, much like plot_percent_change, from all of the regions (all datasets) using plotly scatter plot with connected lines. This function uses the same helper functions as plot_percent_change for the respective industry.

To make the program interactive, I have implemented a menu using the pygame-menu library. The first menu has buttons where users can choose to graph either the CO_2 emissions/year, the percent change in CO_2 emissions/year, or to compare Regions. Each button corresponds with the 3 types of graphs from the create_graphs module explained above.

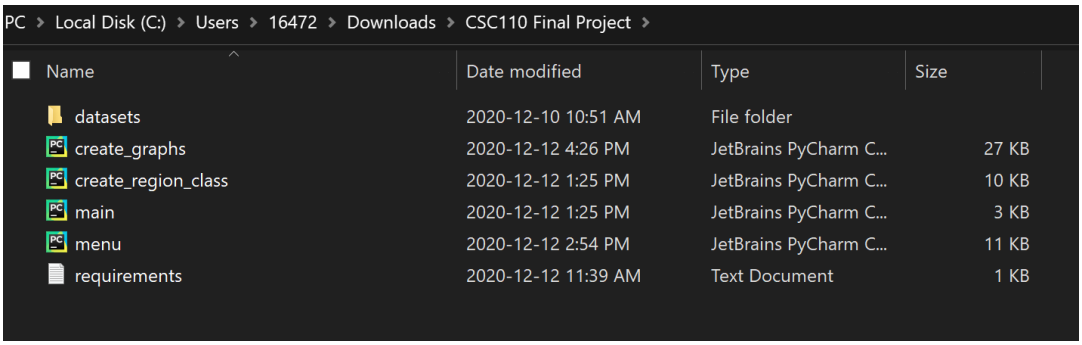
When the first two buttons are pressed, a menu displaying buttons representing regions will show up. Clicking on a region will display the chosen graph for that region. Using button widgets, new functions had to be made that displayed that specific graph per region. These functions can be found on the create_graphs module lines 48-220. The functions that had to be called when the third button was pressed which takes users to a menu with button widgets representing the three industries can be found on the create_graphs module lines 223-509.

Download and Running Instructions

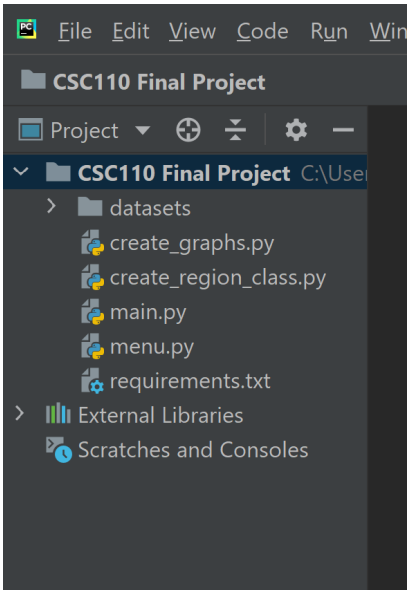
- Download all .py files and requirements.txt from MarkUs and move them all to one folder. If that does not work, use this link.
- Download all datasets from markus (11 of them) (Since they have been filtered and each of them are fairly small, I have uploaded them to MarkUs). Create a new folder called 'datasets' and move all of the downloaded datasets to this folder. It should look like this:



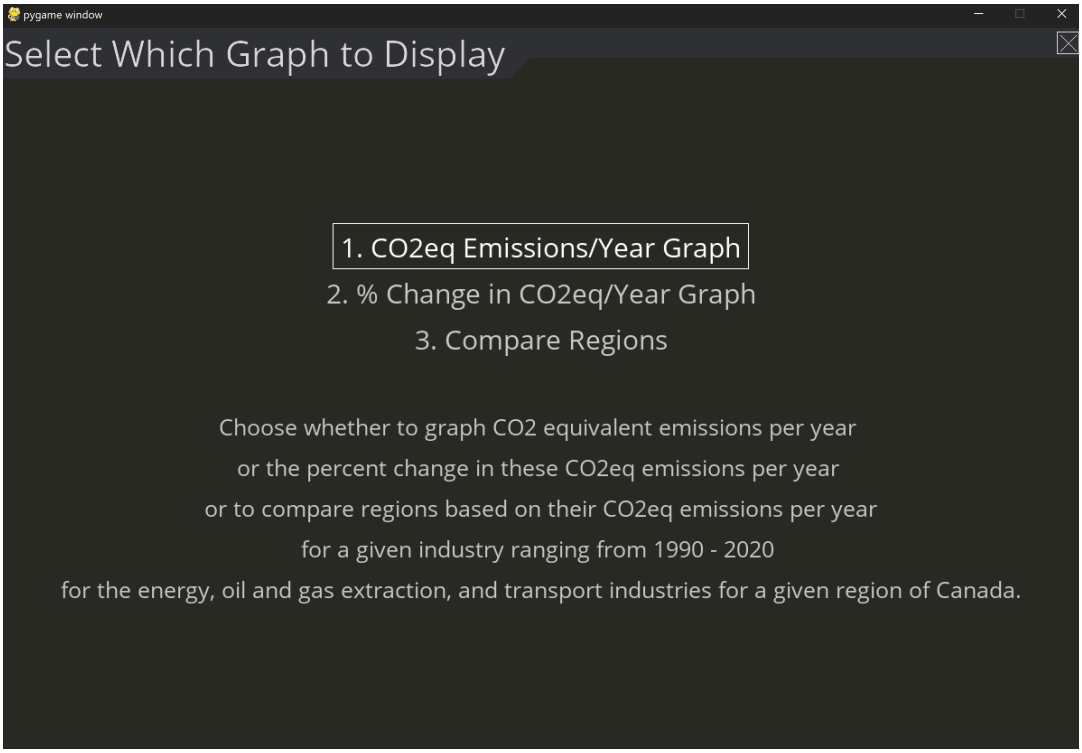
- Move this folder to the folder where you saved the .py files from MarkUs. It should look something like this:



- Now, open the folder with all of the .py files in pycharm. the left side of your screen should look like this:



- Mark this folder as the Sources Root by right clicking ¿ Mark Directory as ¿ Sources Root
- Next, install all libraries listed in the requirements.txt file
- Open main.py and read the introduction at the beginning of the module, then run it. You will be able to interact with the main menu.

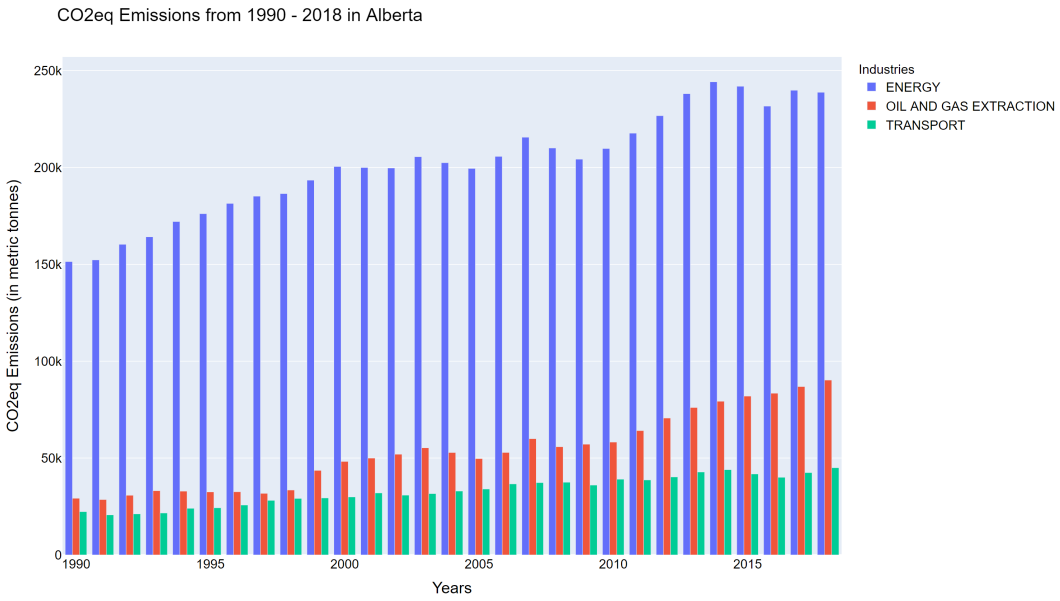


Navigating the Menu

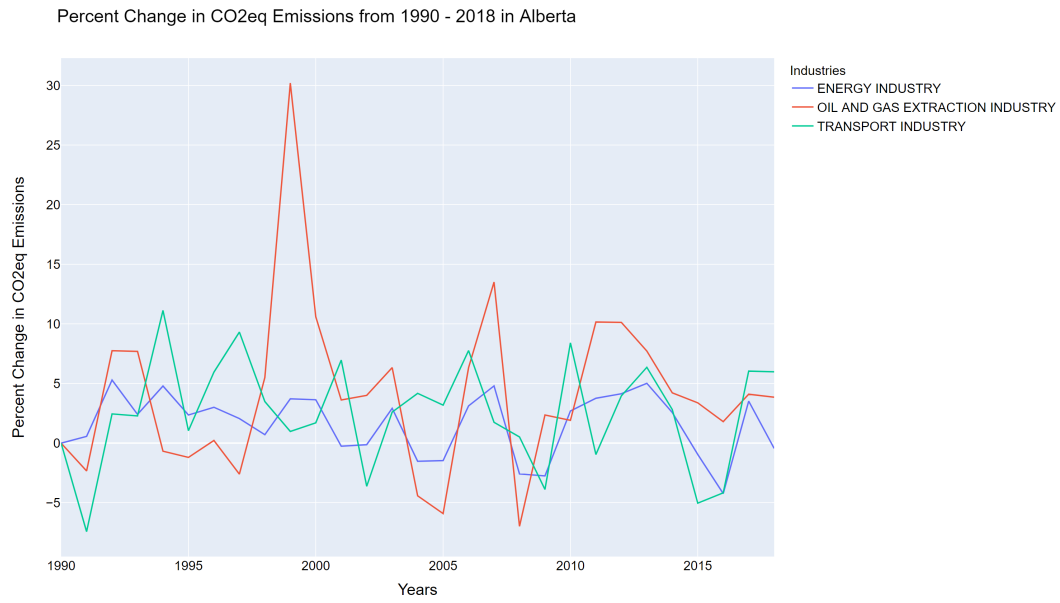
The main menu contains three buttons: one that displays graphs that represent CO_2eq Emissions per year, one that displays the percent change in CO_2eq Emissions per year, and one that will compare regions' percent change in CO_2eq Emissions per year. Clicking either of the first buttons will send the user to the following menu:

The buttons on this screen are labeled as the regions looked at in this project. Clicking one will display a graph in your browser which uses plotly. If you clicked on the 'CO2eq Emissions/Year' button, clicking on a region will

display a graph which represents that region's CO_2eq Emission data from the years 1990-2018 using a plotly bar graph. As an example, here is what gets displayed on my browser when clicking on Alberta:

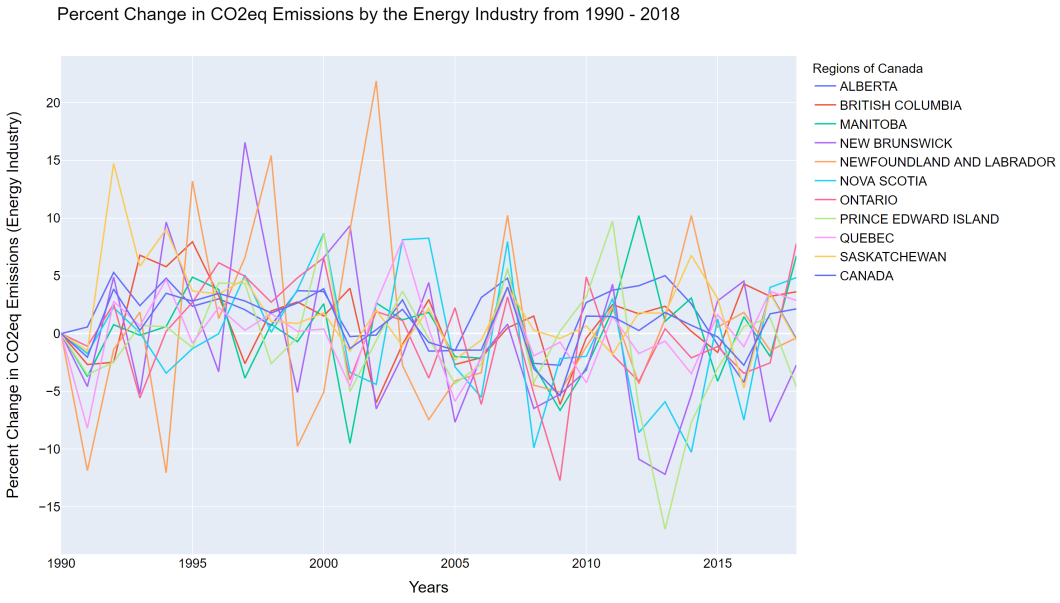


If the '% Change in CO2eq/Year' button is pressed, clicking on a region will display a graph which represents that region's percent change in CO_2eq Emission data from the years 1990-2018 using a plotly scatter graph with the plotly library go.Scatter. As an example, here is what gets displayed on my browser when clicking on Alberta:



Now, if the third button on the main menu, 'Compare Regions', is clicked, another menu will pop up asking you to choose which industry to compare regions with:

Clicking on an industry will open a graph which represents all of the regions percent change in CO_2eq Emissions for that industry. This graph will allow users to compare regions with each other based on how much percentage their carbon emissions have increased or decreased throughout the years. As an example, here is what should be displayed after pressing 'ENERGY':



Though it looks complicated at first, plotly allows users to choose which data to display by simply clicking on the name of the region on the legend to hide/display it.
This program can display multiple different graphs, so feel free to explore through the menus!

Changes from Proposal

Initially, I planned on only displaying two graphs: one representing Canada's CO_2eq emissions per year and one representing the change in carbon emission from only 1990 to 2018 (no years in between). The comments suggested

that I add some complexity to my project, and possibly graph data for the provinces instead of just Canada, or graph data of the different chemical emissions and not just CO_2eq emissions. I decided to graph data based on the provinces' CO_2eq emissions. I believe that this data would allow users of the program to analyze and compare each of the provinces' data and allowed me to add an interactive element to my program: the menu. Since I decided to complicate my original idea, my computational plan had also changed quite a bit. Instead of creating three dictionaries for the three industries all representing Canada, I created a Region dataclass that stored this data in lists as this would be more efficient. I also added another set of graphs that could be displayed which compares all of the regions' percent change in CO_2eq emissions for a given industry in one graph so that comparing region data is easier. Plotly also allows users to choose which region data to display.

Discussion

I believe that the results of my program allows users to answer the research question: How have carbon emission produced by regions of Canada changed throughout several years? The program gives users the option to graph the percent change in CO_2eq emissions by a chosen region for the energy, oil and gas, and transport industries. This data visually depicts the change in carbon emission production through the years 1990-2018, which answers the original research question. This data helps users analyze the rate at which a region of Canada has been increasing or decreasing their CO_2eq emissions after a substantial number of years and is telling whether a region has been working to decrease their carbon footprint. With the 'Compare Regions' button on the menu, users can compare the percent change in CO_2eq emissions for any of the three industries for all of the regions that were looked at in this project. These graphs make it easy for users to see how different regions fare against each other when it comes to improving their carbon footprint throughout the years.

There were many limitations when dealing with the datasets, algorithms, and libraries. Some limitations of the program include the data not being specific enough, as the original dataset contained not just CO_2eq data, but a number of chemical emission data for 83 different industries in Canada, not just the three industries that were analyzed in this program. The program only visualized a small subset of the dataset and could have possibly left out data that could be substantial in further answering the research question. The pygame menu library also offered limitations to the project, as I tried to implement a slider for the user to choose which type of graph to display, however I had trouble sending values of the slider to other functions to graph the selected graph. Instead, I decided to use buttons for the user to interact with which made the code much longer. Moreover, I decided to only limit the regions being represented in this project to just the provinces and Canada. This is because the data which represented carbon emissions in Canada's territories was much lower and less substantial than that of the provinces, for all 83 industries and chemical emissions.

The next step for further exploration could possibly be creating a program which analyzes the change in all regions of Canada's emissions for all of the 83 industries and for all of the chemical emissions that were looked at in the original dataset, not just the subsets that were used for this project. This new program could utilize a pygame menu much like the one used in the original, with more options and a more optimized design for the screen which can fit all of the data the program would display concisely. This program would also contain a Region dataclass, much like the original project, with much more instance attributes representing CO_2eq and other chemical emission data as lists for the 83 industries. Ideally, to make things easier to navigate, users should be able to have an option in which chemical emission data to display, for which industries, and for which regions on the menu instead of clicking on the variables on the plotly browser to hide or show data.

References

(1) February 23, 2017 Melissa Denchak. "Global Climate Change: What You Need to Know." *NRDC*, 30 Sept. 2020, www.nrdc.org/stories/global-climate-change-what-you-need-know.

(2) *Global Greenhouse Gas Emissions.* Environment and Climate Change, Government of Canada, 15 Apr. 2020, www.canada.ca/en/environment-climate-change/services/environmental-indicators/global-greenhouse-gas-emissions.html. Accessed 5 November 2020.

(3) *Canada's Official Greenhouse Gas Inventory.* Environment and Climate Change Canada, Government of Canada, 24 September 2016, open.canada.ca/data/en/dataset/779c7bcf-4982-47eb-af1b-a33618a05e5b, Accessed 5

November 2020.

"Plotly Python Graphing Library." *Plotly*, plotly.com/python/.

"Pygame-Menu." Pygame, Pablo Pizarro R, pygame-menu.readthedocs.io/en/latest/.