CSC110 Project Report:
A Halt on Life, but not Climate Change

Richard Shi

Tuesday, December 14, 2021

Introduction

Ever since the start of the Industrial Revolution in the mid-1700s, greenhouse gases concentrations began to rise (Intergovernmental Panel on Climate Change [IPCC], 2021). One such greenhouse gas, carbon dioxide (CO_2), is being added to the atmosphere far faster by human activities such as burning fossil fuels than any natural processes can remove them (Lindsey, 2020). This is a major concern because its abundance intensifies Earth's natural greenhouse effect: the trapping of heat in the atmosphere to keep the planet warm. Without drastic action, climate change threatens our longevity on this planet. Natural disasters such as wildfires, droughts, and tropical storms will increase in frequency and severity and rising sea levels, rising temperatures and changing precipitation patterns are a few among many possible consequences of climate change ("The Effects of Climate Change," 2021). These changes have major implications, including, but not limited to, the destruction of coastal cities and crippling the global food supply leading to food shortages and higher prices.

Although the pandemic has slowed down economic activity and, by extension, greenhouse gas emissions, global CO_2 levels are still on the rise (Lindsey, 2020) and will continue rising unless we dramatically decrease our greenhouse gas emissions. Not to mention, global temperatures are also continuing to rise. I remember reading a similar article headline which is what inspired me to relate my project to climate change. The headline serves to remind myself and other readers that climate change is still a thing and that a drop in emissions from the pandemic cannot reverse the effects of centuries of greenhouse gas emissions. Not to mention, there is a sense of urgency as if nothing is done to combat climate change, I (and others of this generation) will have to live with the consequences. This leads me to my research question: "**How well could a simple climate model predict real climate data influenced by the COVID-19 pandemic?**" I will be measuring the accuracy of this model by finding the percent error in the recorded and calculated temperature anomaly values.

Dataset Description

The following datasets are all .csv files. These are the datasets I'll be working with:

- This data set is called "Mauna Loa CO_2 annual mean data" which contains yearly CO_2 concentration readings from 1959-2020 from NOAA (Tans & Keeling, 2021). This data contains the year the reading was taken and the CO_2 concentration reading that year. These readings were taken at the Mauna Loa Observatory. I will be using the first two columns of data (columns labelled "year" and "mean."
- This data set is called "Mauna Loa CO_2 monthly mean data" which contains monthly CO_2 concentration readings from 1959-2020 from NOAA (Tans & Keeling, 2021). The data from this data set that we are focused on are the year and month the reading was taken and the average CO_2 reading that month. These readings were also taken at the Mauna Loa Observatory. I will be using the data from the columns labelled "year," "month," and "average." I will also only be using data from January 1959 (denoted by year: 1959 and month: 1 in the dataset) onwards to keep things consistent between all datasets.
- Monthly and annual global "temperature anomalies" from 1959-2020 from NOAA (National Oceanic and Atmospheric Administration, 2021). Temperature anomalies refers to the difference in temperature relative to a baseline value. In this case, the data was the difference in temperatures to the 20^{th} century average global temperature which is 13.9°C (Lindsey & Dahlman, 2021). I will be using all columns of data in these two

datasets. There are only two columns in this data set which are "year" (in the dataset for monthly global temperature anomalies, the month is concatenated with the year) and "value" (the temperature anomaly).

The first two datasets were taken from: https://gml.noaa.gov/ccgg/trends/data.html (licenses in the .csv files)
The last dataset was taken from: https://www.ncdc.noaa.gov/cag/global/time-series/globe/land/ann/9/1959-2021 (use is outlined here: https://www.noaa.gov/big-data-project-frequently-asked-questions)

Computational Overview

Description of major computations

- I have created two data classes, YearlyMetrics and MonthlyMetrics, to store the data from the data sets and the climate data generated in "extrapolate data" mode. These data classes have instance attributes for the year, CO_2 concentration levels, and the recorded and calculated temperature and temperature anomalies. The difference between the two is that the MonthlyMetrics data class also has an instance attribute for the month that data was taken.

- The data from the data sets was processed and aggregated into lists of YearlyMetrics and MonthlyMetrics objects by the load_yearly_data and load_monthly_data functions from load_data.py. It used the csv library to help organize each row of data into the list and it was just a matter of instantiating data class objects with values from each row in the csv file. The data generated from the "extrapolate data" mode was handled by the extrapolate_data function in calculations.py. It utilizes two methods calculate_concentration and calculate_temperature which employ a formula to calculate the CO_2 concentration and the temperature (the formulas will be discussed in the point below). Then, the extrapolate_data method instantiate YearlyMetrics objects with year, co2, temp, and temp_anomaly instance attributes. As a design choice, rather than storing the temp and temp_anomaly values in calculated_temp and calculated_temp_anomaly, respectively I chose to store them in temp and temp_anomaly. I did so because it wouldn't make much of a difference and it allowed for functions in visualization.py to work for both sets of data (recorded and extrapolated) and reduce the complexity of the code.

- The calculate_concentration function was created to generate CO_2 concentration data and used as a parameter in the calculate_temperature function as well. The calculate_concentration function employs a formula to calculate the CO_2 concentration which is taking 45% of a given amount of carbon emissions (in gigatons, GtC), dividing the result by 2.3, and then adding it to the previous year's CO_2 concentration (University Corporation for Atmospheric Research). Initial values were taken from the co2_annmean_mlo.csv file and correspond to data from 2020.

- The calculate_temperature function uses the following formula to calculate the temperature: $T = T_0 + S * \log_2 \frac{C}{C_0})$, where T_0 and C_0 are the known temperature and CO2 concentration at some reference time, S is the climate sensitivity factor, and C is the new CO2 concentration (University for Atmospheric Research). This function also calculated the "temperature anomaly" which is the difference in temperature to some baseline temperature value which is the 20^{th} century global average temperature of 13.9°C (Lindsey & Dahlman, 2021).

- Conversely, the global temperature was calculated by simply adding the temperature anomaly to that baseline value of 13.9°C. This was done when instantiating YearlyMetrics and MonthlyMetrics objects in the load_yearly_data and load_monthly_data functions in load_data.py.

- For reporting the accuracy of the model, I've calculated the percent error for each entry of data with the following formula: $\%\ error = \ln e^{\text{calculated - recorded}} \times 100\%$. This is done in the calculate_error function in calculations.py which returns a mapping of the date of the data entry to the percent error. I also calculated the average percent error in calculate_average_percent_error by dividing the sum of all the percent errors with the number of entries.

Explanation of Reporting Results

My program will display the data in the data set in an interactive way by creating a GUI where the user can change parameters to output a graph of the data. The program outputs two graphs, one is the time series of CO_2 concentration and the temperature while the other is a small section of the data showing the temperature anomaly over a smaller period of time. Note: I used the data for 1959 and 1959-01 as initial values for calculating the temperature anomalies for the yearly and monthly data so this is why the second graph excludes those dates. The user can also check the accuracy of the model by checking the percent error of the climate model's calculations to the data in the data sets. The program shows a time series of the percent error of the model for a small portion of the data that is chosen by the user. The average percent error is also outputted to the Python console.

Explanation of the Use of New Libraries

My program uses tkinter to create an interactive GUI to display the climate data (see the run_visualization function in visualization.py). It has various widgets such as tk.Scale and tk.Button which allow the user to change various parameters to change the outputted graph. In particular, the tk.Scale widget was very useful since it allowed me to create many sliders to change parameters such as the climate sensitivity factor, how many years to extrapolate data, and the amount of annual carbon emissions. I preferred using these sliders over getting user input because I didn't have to worry about constraining the user to follow rules for formatting their input.

I've also used plotly to generate these graphs. I used plotly.subplots.make_subplots to create a graph that has two y-axes—one for CO_2 concentration and one for temperature (see plot_climate_data in visualization.py). I also used plotly.graphing_objects.Bar to create a double bar graph to compare temperature anomaly values (see plot_compared_data).

Instructions for the Program

Obtaining data sets

To the grader: there are many ways to obtain the data sets for this program.

1. I've submitted the .csv files on MarkUs. The file names of the data sets are (in no particular order): co2_annmean_mlo.csv, co2_mm_mlo.csv, annual_temp_anomalies.csv, and monthly_temp_anomalies.csv.
2. You can download the data sets from the following links (but ensure that their file names match):
   - https://gml.noaa.gov/webdata/ccgg/trends/co2/co2_annmean_mlo.txt (make sure this is named co2_annmean_mlo.csv)
   - https://gml.noaa.gov/webdata/ccgg/trends/co2/co2_mm_mlo.txt (make sure this is named co2_mm_mlo.csv)
   - https://www.ncdc.noaa.gov/cag/global/time-series/globe/land/all/12/1959-2021 (make sure this is named monthly_temp_anomalies.csv)
   - https://www.ncdc.noaa.gov/cag/global/time-series/globe/land/ann/9/1959-2021 (make sure this is named annual_temp_anomalies.csv)

Once you've obtained the data sets, ensure that they are in the same directory as the other Python files (i.e. in the same folder).

Running the Program

When you execute main.py in the Python console, a window should pop up.

- The first slider is for choosing to display data from the yearly or monthly climate data sets. Adjusting to the slider to the left will display monthly data while sliding to the right will display yearly climate data.
- The next slider is for adjusting the climate sensitivity factor. This will affect the calculated temperature and calculated temperature anomaly values.
- Next is a toggle button for extrapolating data. It is defaulted to off but clicking it will toggle on "extrapolate data" mode. This reveals two new sliders, one for adjusting how many years you wish to extrapolate the data for and the other for adjusting how many gigatons of carbon is emitted annually. The extrapolate data mode only extrapolates annual data.
- The next button shows the accuracy of the model by prompting the user for an input date in the Python console. The format is YYYY-MM (year-month) so 1959-02, 1959-2, and 1959-12 are acceptable inputs. Ensure that

the input is within the range stated in the Python console. Also, the input for month should be between 1 and 12. A graph will appear showing a time series of the percentage error in temperature anomalies for 12 consecutive data entries, starting from the date you inputted.

- The last button will output two graphs (if not in "extrapolate data" mode), but will first prompt the user for an input date. The rules from the previous point still apply here. Then, the program will produce two graphs, one comparing the calculated and recorded temperature anomaly values and the other showing the CO_2 concentration and temperature change over all entries in the data set. If toggled on "extrapolate data" mode, then it will only output one graph which is the CO_2 concentration and temperature change over the specified time interval.

Description of Changes

- Rather than graphing expected vs actual temperature values, I chose to graph expected vs actual temperature anomaly values instead. I first implemented the original idea, but the graphs looked boring since you couldn't really see the difference between the two given the scale and magnitude of the temperatures compared to their differences.
- I also gave the user the ability to show which section of data to display for comparing temperature anomaly values and for checking the model's accuracy. Originally, I planned to just display all entries but the amount of entries made the graphs too clustered and made them difficult to read.
- On the proposal feedback, my TA asked how I was going to report the accuracy of the model so I implemented a button which outputs a graph of the percent error in temperature anomaly values and then prints the average percent error for that run in the Python console.

Discussion of Results

- I think that the results of these computations did help answer the research question. The graphs showing the percent error and the calculated vs recorded temperature anomaly values show that this model doesn't do a good job at predicting real climate data. The percent error wasn't even consistent and kept fluctuating. Not to mention, comparing the temperature anomaly values realized that the model would often overestimate and underestimate the actual temperature anomaly values. For example, in late March/early April 2020, most of the world began implementing lockdown measures ("Coronavirus: The world in lockdown"). This would intuitively lead to less carbon emissions and a reduction in global temperature. However, in the model, it was consistently overestimating the temperature anomaly value from March-July 2020.
- Some limitations in this model is that it doesn't consider the impact of other greenhouse gases and how they contribute to increasing global temperatures. Not to mention, the model also doesn't consider all of the processes that remove greenhouse gases from the atmosphere. A shortcoming in the program is that the "extrapolate data" mode was not as exciting as I had hoped. No matter the input, the graphs all have the same shape. The CO_2 concentrations always grow linearly and the temperature always grows logarithmically. In addition, when coding the GUI and the callback functions, it was extremely challenging trying to have certain features hide/reveal others. For example, when toggling the "extrapolate data" mode, it reveals two new sliders. However, when implementing this, the callback functions in my solution required so many parameters that it become hard to keep track of.
- Some steps to improving this program could first be finding a better climate model. In this program, a very simple one was used just so I could get something working, but a more accurate model is required for more accurate results. Such model could account for a wider variety of greenhouse gases and be more adaptive to a change in human activities. This could be implemented into my program by altering the formula used to calculate the temperature. In addition, rather than using the same amount of carbon emissions annually for a given number of years, it would be better if it was more dynamic and allowed the user to change the amount of emissions annually (would also solve the problem of boring extrapolated graphs). However, all of this would naturally increase the complexity of the program as we would need much more data (e.g. data for other greenhouse gases) and more code for processing, aggregating, and manipulating that data for producing graphs.

In conclusion, we can say that this simple climate model is not very good at predicting real climate trends. It fails to consider other contributors to increasing temperatures and is not dynamic enough for extrapolating future climate data. Further work is required to create a more accurate model and a better program. An ideal model will need to take into account all possible (or at least most) contributors to rising temperatures. Perhaps better libraries are required for graphing the data to enable the user to change parameters in the model and see the change reflected in the graph in real time.

References

Inc., P. T. (2015). Collaborative data science. Montreal, QC: Plotly Technologies Inc. Retrieved from https://plot.ly

"Coronavirus: The World in Lockdown in Maps and Charts." BBC News, BBC, 6 Apr. 2020, https://www.bbc.com/news/world-52103747.

Intergovernmental Panel on Climate Change. (2021). *Climate Change 2021: The Physical Science Basis.* https://www.ipcc.ch/report/ar6/wg1/downloads/report/IPCC_AR6_WGI_Full_Report.pdf.

Lindsey, Rebecca. (2020, August 14). *Climate Change: Atmospheric Carbon Dioxide.* NOAA Climate.gov. https://www.climate.gov/news-features/understanding-climate/climate-change-atmospheric-carbon-dioxide.

Lindsey, Rebecca., & Dahlman, Luann. (2021, March 15). *Climate Change: Global Temperature.* NOAA Climate.gov. https://www.climate.gov/news-features/understanding-climate/climate-change-global-temperature.

Lundh, F. (1999). An introduction to tkinter. *URL: Www.Pythonware.Com/Library/Tkinter/Introduction/Index. Htm.*

National Oceanic and Atmospheric Administration (2021). *Climate at a Glance: Global Time Series.* [Data set]. National Centers for Environmental Information. https://www.ncdc.noaa.gov/cag/global/time-series/globe/land/all/12/1959-2021

National Oceanic and Atmospheric Administration (2021). *Climate at a Glance: Global Time Series.* [Data set]. National Centers for Environmental Information. https://www.ncdc.noaa.gov/cag/global/time-series/globe/land/ann/9/1959-2021

Tans, Pieter., & Keeling, Ralph. (2021). *Mauna Loa CO_2 annual mean data.* [Data set]. Global Monitoring Laboratory. https://gml.noaa.gov/webdata/ccgg/trends/co2/co2_annmean_mlo.txt

Tans, Pieter., & Keeling, Ralph. (2021). *Mauna Loa CO_2 annual mean growth rates.* [Data set]. Global Monitoring Laboratory. https://gml.noaa.gov/webdata/ccgg/trends/co2/co2_mm_mlo.txt

The Effects of Climate Change. (2021, August 26). NASA: Climate Change and Global Warming. Retrieved from https://climate.nasa.gov/effects/

University Corporation for Atmospheric Research. (n.d.). *The Very Simple Climate Model Activity.* UCAR Center for Science Education. https://scied.ucar.edu/activity/very-simple-climate-model-activity