PROJECT MILESTONE #1: Due by 11:55 PM CT on Monday, February 21, 2022

NAME: **Omkar Mehta (omehta2)**

Important Note #1: Your plans for your project will likely evolve as you progress on the project, especially your processing and analysis techniques. "BIG" changes to what you submit should optimally be run past me, to ensure you are still on a feasible track. I will give you a mechanism for doing this.

Important Note #2: Please be sure you are answering each question in its entirety – many questions have multiple sub-questions.

Important Note #3: To make both our lives easier, please answer directly in this document after each question or sub-question – make your answer a different color. Thanks!

- **(1) What data do you want to work with for this project? This can be, but does not have to be, the dataset you identified in your homework assignment**

  - A. Please include the link, and a few sentence description of your dataset. Include:
    - a discussion of whether your data is spatial or not
    - what the file format is (e.g., text/CSV, binary like Netcdf, other format, etc.)

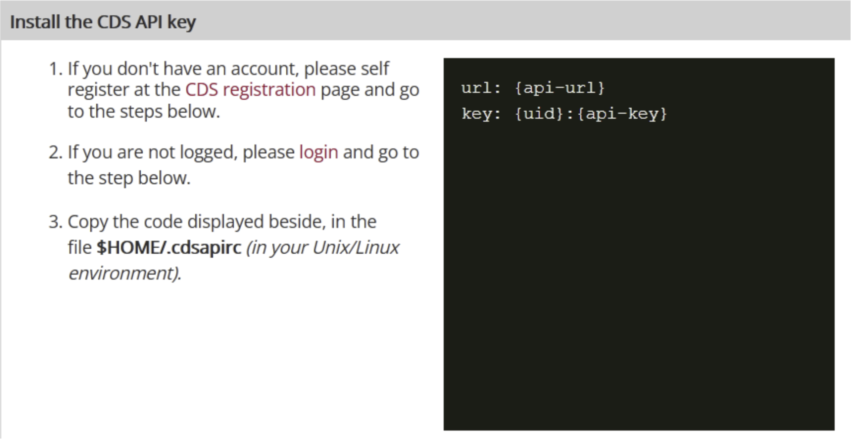
  ERA5 dataset
  - I will be downloading ERA5 monthly averaged data for various parameters like Temperature of air at 2m above the surface, Total precipitation, etc.) from this website. ERA5's land data documentation can be found here.
  - Complete Era5 dataset covers the period from 1950 and continues to be extended forward in near real-time. ERA5 is produced using 4D-Var data assimilation and model forecasts.
  - The data are archived in the ECMWF data archive (MARS) and a pertinent sub-set of the data, interpolated to a regular latitude/longitude grid, has been copied to the C3S Climate Data Store (CDS) disks.

  Dataset Information:
  - The dataset is available in GRIB and netCDF file formats, out of which I will be using netCDF file format.
  - The data is spatial and temporal.
    - Temporal frequency:
      - I could select the day-time, months, years at which I want the data.
    - Spatial Grid:
      - The ERA5-Land HRES dataset has been produced at a resolution of 9 km, (~0.08°) and in a (octahedral) reduced Gaussian grid (represented as TCo1279).

  - B. Describe in a few sentences how you will access this data.

  1. Make an account with the Climate Data Store / ECMWF at this website. https://cds.climate.copernicus.eu/cdsapp#!/home
  2. I need to install the python api to download the data. Here are some basic instructions. I can do pip install cdsapi. https://cds.climate.copernicus.eu/api-how-to

3. Once I make my account, visit the api-how-to above and there should be a field that looks like this.



Install the CDS API key

1. If you don't have an account, please self register at the CDS registration page and go to the steps below.
2. If you are not logged, please login and go to the step below.
3. Copy the code displayed beside, in the file **$HOME/.cdsapirc** *(in your Unix/Linux environment).*

```
url: {api-url}
key: {uid}:{api-key}
```

- When I login, the right fields will be populated with my api key. I just have to create a file according to the instructions.

4. Finally, what years do I want your dataset cover? (I need to discuss this with you). The most popular and comprehensive dataset is ERA5, https://cds.climate.copernicus.eu/cdsapp#!/dataset/reanalysis-era5-single-levels-monthly-means?tab=form
5. I can select the Total Precipitation variable here and build a custom subset. The highest resolution is 1 hour, but if I want 6 hours I can mark the check boxes every 6 hours starting from 0:00
6. And then at the bottom I can copy and paste the code it gives me in the "Show API request". I run this as a python script and it'll automatically make a request to ECMWF, put me into a queue, and start the download.
7. Once the dataset is downloaded, I will store it in Google Drive so that I can perform analysis, visualization and model-training on Google Colab.

I will only consider the 9-10 days of July, 2021, as the whole dataset is very large. I might change the month, depending on the visualizations. The model training would be done on the subset of the data, especially Maharashtra, India where heavy rainfall takes place.

- C. What is the size (in mb or gb) of this dataset (or the subset of the dataset you're using?)
    - The dataset would be around 500 mb for each parameter that I decide to use in the final project.
    - The subset of the data would be smaller size like 200 mb.

- D. Do you anticipate having issues storing this dataset where you're running Python? *Keep in mind, there do exist ways to remotely access some kinds of data and use it in your code without having to download it – whether this will be possible for yours specifically depends on a variety of factors*

        - Yes. I will be using Google Drive to store the data. All of my analysis would be done on Google Colab.

    - E. Please verify to me that this dataset is “new” to you (you have not worked with it significantly before).
        - I had heard of this data but never got to use it for whatever I had learned so far.

- **(2) What do you hope to investigate with this data – i.e., what are your current objectives?**

    - A. In your answer to this question, please be sure to discuss which variables you think might be of greatest interest/relevance
        - Your objectives likely will evolve and that’s fine! This is just based on your initial physical intuition of the data.
        - Perform Exploratory Data Analysis and Visualisation of the datasets.
        - Get statistical information like mean, median, std, iqr etc fro each dataset.
        - Check for missing values, NaN values. Handle missing values.
        - Predict the future 2m temperature using Deep Learning Models like LSTM and Neural ODE.
        - Create time series and find similar time series (Deep Time Series Clustering).
        - For the model, the variables would in fact be the time-lag that we introduce in time series analysis. So, I would consider 10 time steps in the past to predict in the future time step.

FYI: So you are aware, one of your initial steps in your project will be exploratory data analysis, where you’ll examine basic characteristics of your data, do some cleaning, and produce numerical and visual statistical summaries of your data. This will guide you in refining your objectives!

- **(3) Based on what you know about your data so far, what at least TWO types of processing techniques might you use on your data and why? Doing quality-control/cleaning on your data is required, and therefore is a given, so it needs to be TWO techniques in addition to quality-control/cleaning.** *You are not limited to the below list. For the purposes of this project, any technique that prepares your data for more advanced statistical analysis will likely count as processing.*

    - Options include, but are not limited to:
        - *Smoothing* (reducing noise to highlight the signal)
        - *Interpolation* (estimating data where there isn’t currently data [within the bounds of current data])
        - *Grouping* (sub-grouping data by certain conditions to allow for more meaningful analyses completed on the sub-groups)
        - *Masking/Filtering* (identifying data that meet a certain condition for reasons other than quality-control)
        - *Merging* (combining datasets together carefully)
        - *Transformation* (changing the statistical distribution of your data)

        - Interpolation part, because the spatial resolution of the dataset is 0.1x0.1 degree, which I can increase it to 0.05x0.05 degree.
        - Transformation, because for deep networks, the data needs to standardized.
        - Merging (combining datasets together carefully), because some of the visualizations that I've thought of, include the values like 2m temperature and precipitation to be shown on the map plot, using plotly.
        - Masking/Filtering, because the data has many NaN values.

- **(4) Based on what you know so far, what type(s) of advanced statistical analysis technique(s) might you use on your data? Your choice will depend in large part on what your objectives of your project are, yes?**

  Note: It is required that you evaluate correlations and basic statistics on (relevant subsets) of your data.

  You must choose, apply, and evaluate at least one advanced statistical analysis technique on your data, in your final project that you turn in…it is likely you will play around with multiple techniques along the way! Your options for this are:

    - (1) Developing and applying a regression model to your data. Regressing modeling is one of the most commonly-used machine learning techniques, and in this class you will learn, in-depth, how to run and evaluate ordinary, linear least-squares regression models.

      FYI: you would use a regression model to investigate **relationships** between variables and/or **make predictions**

      Depending on the type of data you're analyzing, and other factors, OLS linear regression modeling may not be the most suitable technique, so here are some additional options (*caveat: the below types of regression, I likely will not formally teach in this class in-depth; potentially limited support from me/your peers if you go down this route, but I can try to help!*):

        - Poisson regression
        - Logistic regression
        - Non-linear regression

    - (2) Developing and applying another type of machine learning model - e.g., clustering or classification (a solid broad overview of the uses of each can be found here: https://jakevdp.github.io/PythonDataScienceHandbook/05.01-what-is-machine-learning.html)

      Caveat: I formally teach a bit about these other machine learning techniques in this class, conceptually, and if time permits I'll also present simple examples of running them in

Python. So – potentially limited support from me/your peers if you go down this route, but I can try to help!

- (3) Running a hypothesis test (great overview here: https://towardsdatascience.com/hypothesis-tests-explained-8a070636bd28), for example to evaluate if certain characteristics of two distributions of data are statistically significantly different. I do plan on covering hypothesis tests to some degree this semester, including how to run basic, common tests in Python.

- (4) Using an 'advanced' statistical tool not outlined in (1), (2), or (3). *Potentially limited support from me/your peers if you go down this route, but I can try to help!*

If you're looking to challenge yourself in this class, you are encouraged to use this aspect of the project to do so!

- Since the data is sequential, I wish to perform time series forecasting using LSTM, Neural ODE to predict the next temperature/precipitation.
- We use Euclidean Distance for finding the similarity between two vectors (K-means clustering). But, time series will have lags of 90 or sometimes 180 degrees. Hence, euclidean distance won't be a proper measure to find the similarity. Instead, we could use something like Time Series Clustering or Deep Multivariate Time Series EmbeddingClustering via Attentive-Gated Autoencoder for finding similar time series. This will help us find out similar trajectories in different grids.
- **(5.) Will you use additional, field-specific packages for this project? If so, list them.**
  - I will use xarray and pandas.
  - For models, I will use PyTorch and TensorFlow.

- **(6.) If you foresee using a field-specific package in place of Pandas/Xarray (which are the two primary packages you'll learn in this class to read in data, do some processing and visualization, etc.), please justify how your use of a field-specific package(s) will still demonstrate your grasp of relevant skills learned in this class.**
  - *Pandas*:
  https://pandas.pydata.org/pandas-docs/stable/getting_started/intro_tutorials/01_table_oriented.html
  - *Xarray:* http://xarray.pydata.org/en/stable/why-xarray.html

  - Not applicable. I will be using xarray and pandas. These two libraries are sufficient for processing, visualization, and modeling.

- **(7.) What challenges do you anticipate you might encounter in this project?**

- Since the dataset is huge, deep learning models and clustering tasks might take a long time. I might have to use High-RAM GPU for training the model.