INTRODUCATION:

A data center is a facility that centralizes an organization's shared IT operations and equipment for the purposes of storing, processing, and disseminating data and applications. Data centers are vital to the continuity of daily operations as they house an organizations most critical and proprietary assets. Choosing the right data center location to house virtual infrastructure and data can be crucial to avoiding the debilitating costs of unplanned downtime.

Data Centers consumes a significant amount of energy. They contribute to at least two percentage of global electricity usage. Since power represents up to 70% of the total operating costs of the data centers, many enterprise users and colocation operators focus their site selection on lower-cost power options. Renewable power sources like Solar panels are becoming increasingly viable and economical to help make companies less dependent on power suppliers. Data centers also emit a significant amount of CO2. The IT industry is responsible for approximately 2% of global CO2 emission, which is on par with the aviation industry.

It is inherently more difficult and more expensive to keep servers and other hardware cool and running smoothly when data centers are located in warmer climates. Any data center located in high temperature environments can face an enormous drain on overall efficiency because of the massive energy consumption required to keep things cool.

Additionally, being close to end users is always an important data center location criterion, mainly for latency reasons. By reducing the distance between the data center and its end users, the data center can provide faster response times. More populated regions have additional advantages like obtaining power sources at reasonable prices, high network bandwidth connected to the network/internet, hire IT staff, etc.

NO2 can represent how actively people live and work. A place with a higher NO2 level is a more favorable place to build a data center because that means there are more people to sell the cloud services. As you can see here, NASA measures NO2 on the Earth surface with the Ozone Monitoring Instrument (OMI) equipped on the Aura satellite. The full set of raw NO2 data is available on this page.

PROJECT GOAL

The project will identify a set of reasonable data center locations with global satellite using the NASA's Prediction of Worldwide renewable Energy Resources (POWER) web service: https://power.larc.nasa.gov/

The following parameters are considered in the project:

1. Irradiance (available sunshine to generate solar power)
   The higher, the better in terms of building a data center because the customers would pay less for electricity bills as they can generate more solar power.
2. Temperature
   The lower, the better in terms of building a data center because the customers would pay less for cooling a data center and servers as it can be ventilated with colder air.
3. NO2 level
   The higher, the better in terms of building a data center because the customers can sell their cloud services to more nearby people and hire more nearby people.

IMPLEMENTATION:

Fetch irradiance and temperature for given cities

The project downloads the data about "available sunshine" (solar irradiance), temperature, and average NO2 at a given location from the NASA POWER service. The input to the program is either a GPS coordinate (decimal latitude-longitude coordinate) stored in inputFolder/lat-lon.xlsx file or a mailing address (in string, such as "Boston, USA") stored in inputFolder/city-name.xlsx file using the naming conventions as:
City name, State code: for US cities
City name, Country name: for non-US cities
The program will read the city-name file and fetch the corresponding latitude and longitude for each city. When the program reads a lat-lon file, perform reverse-geocoding to obtain a city name (and state code or country name) for each lat-lon pair. Both the results are then concatenated into a data frame which will now consist of all the cities with City Name, State Code/Country Name, Latitude, and Longitude.

NASA POWER offers a REST API for data downloads. The program will contact NASA POWER server using REST API to get irradiance and temperature data for each city in the data frame with the following parameters:

- All Sky Surface Shortwave Downward Irradiance: to get the annual average in the unit of kW-hr/m^2/day.
- Temperature at 2 meters: to get the annual average in the unit of C.
- Temperature at 2 Meters Maximum: to get the annual maximum in the unit of C.
- Temperature at 2 Meters Minimum: to get the annual minimum in the unit of C.

For each parameter, the program will fetch the data for the recent 5 years (2016-2020) and calculate the average. The average of irradiance, average temperature, maximum temperature, and minimum temperature values are then appended to the data frame for each city.

Fetch Average NO2 for given cities

The project uses NASA's global NO2 measurement data. The CSV to download the data that contains NO2 measurements at many (globally distributed) cities from January 2015 to this month is available on:
https://so2.gsfc.nasa.gov/no2/pix/time_series/OMNO2_Timeseries_AllCities.csv.
The Python program parses thorough this CSV file and computes the daily average of (median) NO2 for each city. The CSV file uses approximately 2,500 rows for each city with each row for one day NO2 measurement. So, we extend the program to calculate the average NO2 value for each city listed by NASA which is approximately 250 cities, let's call them *nasa_cities*. As we require the NO2 values for the cities mentioned in the inputFolder, let's call then *input_cities*. We traverse the list of *nasa_cities*, retrieve the NO2 level of each city from the *input_cities*. If the city is not included in the *nasa_cities*, we find the closest available city using Geopy and append to the data frame of the corresponding city. The program then creates a new outputFolder/irradiance_temp_no2_data.xlsx file and put the following fetched data: City Name, State Code/Country Name, Latitude, Longitude, Irradiance, Avg temp, Max temp, Min temp, Average NO2.

Hierarchical Clustering and Non-Hierarchical Clustering

The program performs clustering using hierarchical and non-hierarchical clustering algorithms on the resulting dataset to analyze which cities are more similar with each other and which ones are less similar with each other in terms of irradiance, temperature and NO2 level.

The program uses scipy's implementation of hierarchical clustering because it can show a clustering result as a dendrogram. For the dendrogram linkage, we used metric='euclidean'. However, the irradiance and avg temp have very different data ranges which is not good in terms of using Euclidean distance metric and hence the program normalized the data and then run hierarchical clustering with normalized irradiance and normalized avg temp values. The data is then clustered into 6 clusters using fcluster() and is visualized with 2D and 3D scatter plots using Plotly and Pandas.
The program uses k-means implementation of non-hierarchical clustering and is visualized with 2D and 3D scatter plots using Plotly and Pandas.

Pareto comparison

The program uses data in the dataset to compare and rank cities - candidate locations for data centers in terms of irradiance, temperature and NO2 level using Pareto comparison or domination-based ranking by calculating the domination count and plot graphs after calculating the cumulative for each of the three parameters.

As for visualization, the program plots individual cities in a 3-dimensional scatter chart. Its 3 axes will represent irradiance, avg temp and NO2 level. Besides a 3-dimensional scatter chart the program also plots a scatter matrix chart, a polar scatter chart and a polar line chart.

Bubble Map Visualization

The program uses the data from the dataset to show a global map and a US Cities only map of each city's irradiance, average temperature and NO2 level on a bubble map based on its latitude and longitude. The program adjusts the circle's diameter based on its irradiance level (A bigger circle means a higher irradiance level), Average temperature level (A bigger circle means a cooler the temperature) and NO2 level (A bigger circle means a higher NO2 level). The program also has a zoom-in/out feature on the map. The color scheme is categorized based on top 10% cities, top 10-20% cities, and 20%+ cities.

The program calculates the following value (DHI) for each candidate city:
Datacenter Hotspot Index (DHI) = [a * (normalized irradiance) + b * (normalized avg temp) + c * (normalized NO2)] / 3
Where,

- Normalized irradiance = (Irradiance at a city in question) / [(Max irradiance among all cities) - (Min irradiance among all cities)]
- Normalized avg temp = 1 - T
    - T = [(Avg temp at a city in question) - (Min avg temp among all cities)] / [(Max avg temp among all cities) - (Min avg temp among all cities)]
- Normalized NO2 = [(NO2 at a city in question) - (Min NO2 among all cities)] / [(Max NO2 among all cities) - (Min NO2 among all cities)]
- by default, a = b = c = 1.0

Irradiance, temp and NO2 are normalized because their value scales are different. Normalized temp is calculated as (1-T) because, the lower the temp is, the better in terms of building data centers. Higher numbers are computed for a city with lower temps.
For normalized temp and NO2, the numerator looks like [(NO2 at a city in question) - (Min NO2 among all cities)] because temp and NO2 values can be negative.

The program then uses the DHI data to show a global map and a US Cities only map of each city's DHI level on a bubble map based on its latitude and longitude. The program adjusts the circle's diameter based on its DHI value; A bigger circle means a higher DHI value. The bubble maps also have a zoom-in/out feature. The color scheme is categorized based on top 10% cities, top 10-20% cities, and 20%+ cities.