BDA Project Report

a. Ami Mandekar(am7031) and Sneha Rohra(sr6225) worked on the project together.

b. Introduction: We first understood the problem definition and what is expected out of us. We tried making sense of whatever data and code we are given with, and then tried to figure out the gaps and necessities that are required. We then went ahead one step at a time and tested the minor changes we made. All the data was then collected and worked on to get the desired output.

c. Background:

GPS is an accurate worldwide navigational and surveying facility based on the reception of signals from an array of orbiting satellites.
The "Doppler Effect," a shift in radio transmission that allowed scientists to follow the satellite during the Sputnik era, is where GPS first emerged. Midway through the 1960s, the US Navy experimented with satellite navigation to monitor its nuclear-armed submarine fleet.

Six satellites orbiting the poles allowed submarines to quickly and accurately determine their location by observing changes in the Doppler of the satellites.

GPS Services:
Both military and civilian users can employ GPS satellites. The civilian service is continuously and universally accessible to all users at no cost. The military service is open to the armed forces of the United States and its allies as well as authorized government organizations.

GPS Applications:
To support atmospheric and ionospheric sciences, geodesy, and geodynamics from monitoring sea levels and glacier melt to measuring the Earth's gravitational field - GPS is employed as a remote sensing instrument. Through policy advocacy for modernization improvements through the GPS requirements process, the National Space-based Positioning, Navigation, and Timing (PNT) Executive Committee, and the

National Space-based PNT Advisory Board, SCaN and NASA's Science Mission Directorate have teamed up to enhance the performance of the GPS constellation.

References:

- https://www.gps.gov/systems/gps/
- https://www.nasa.gov/directorates/heo/scan/communications/policy/what_is_gps

d. Ethical Analysis

Google has access to all of this information. How could this be misused?
Knowing someone's location data is a very private information. This can be misused in many ways. For example if a high profile person is sharing its location access with google a hacker might gain access to this information. This information can be bought terrorists/ radical groups to track down the person and harm him/her.
Also knowing there are a lot of patterns created from bulk data, for example a mall is overcrowded on weekends, highways are busy on fridays. Access to this information by terrorist can help them easily create panic situations.

Is it ethical for you to analyze this data from Dr. Kinsman?
This data was curated and given to us by Dr. Kinsman with a purpose. Using this data to learn about big data and working on the deliverables of the project is completely ethical because this data is not taken from google or any other platform unethically, without permission.
However, using this data to monitor Dr. Kinsman activities, or sharing this data with anyone else is unethical.

What ethical issues might you run into?
If the data from our machines gets somehow leaked and someone else misuses this data, then we might run into ethical issues.

e. Data Cleaning

- After globbing all of the Gps files from all the folders, we first take in all the fields depending on the type of protocol mentioned, i.e GPRMC or GPGGA fields. We take in the timestamp to be a datetime object, and have it converted to Hours, Mins, Seconds format.

- Then latitudes and longitudes are converted into degrees and minutes.This is also adjusted according to North and South direction.
- All this data is recorded in a gps_info dictionary and the key here is timestamp. It is updated every time a record is present in the dictionary and is added and a new record in case a key is not present.
- Now, we remove redundant points. How we do this is by checking the latitude and longitude values. We first iterate through the entire key list and have a group of two longitudes and latitudes to check(current point and next). We compare their values with precision upto 5 places to see if they are the same point and remove one of them. This will also remove redundancy in case the car is moving in a straight line, since every two pairs of latitudes and longitudes are checked.
- We also check for redundancy in cases of constant speed at a particular instance. In this case, if we are moving in a straight line, unnecessary points in the middle are removed.
- We also remove the entry if the difference in latitude or longitude is too high compared to the speed. This manages errors where a GPS reading is way off.

f. Problem Definition:
Here we are using different gps data points collected from actuators and sensors. The problem is to use this data and infer something meaningful out of it. This GPS data is used to see the tracks of the car's moments, the places it visits, its speed, where it is parked.
Using various meaningful attributes like latitude, longitude, variance, timestamp we are checking where Dr. Kinsman parked his car.

Strategy:
We are finding if at all the car is parked and if the adjacent datapoints will very small(almost zero) change in its latitude and longitude.

We are considering if the difference in latitude and longitude of adjacent points is less than 0.00001 then there is no moment in the car and it is parked. The time for which it is parked is the difference between their timestamp. If the next data point is for the same spot we update the car parking time. We are updating the parking time as long as the value of latitude and longitude do not change more than 0.00001. We store this latitude and longitude along with its parking time in a dictionary format. Since the latitude and longitude is what uniquely defines the stop. A string of combination of latitude and longitude is set as keys an attribute

of parking time is added as values. This entry is updated if the parking time for the same spot is more the next time.

The issue that we faced here was that when the car was parked in similar spot the next time but not on as close as their difference in latitude and longitude falls is less than 0.00001 this will be considered another stop. And since the key is a combination of latitude and longitude a new entry is made in dictionary. Because of this we are getting multiple stops close to each other where the car might be parked multiple times.

To overcome this issue we added the combination of longitude and latitude as key but taking the values upto only 4 decimal places. This help reduce the problem a bit but there are still a couple of points in close proximity in areas where car is parked multiple times.

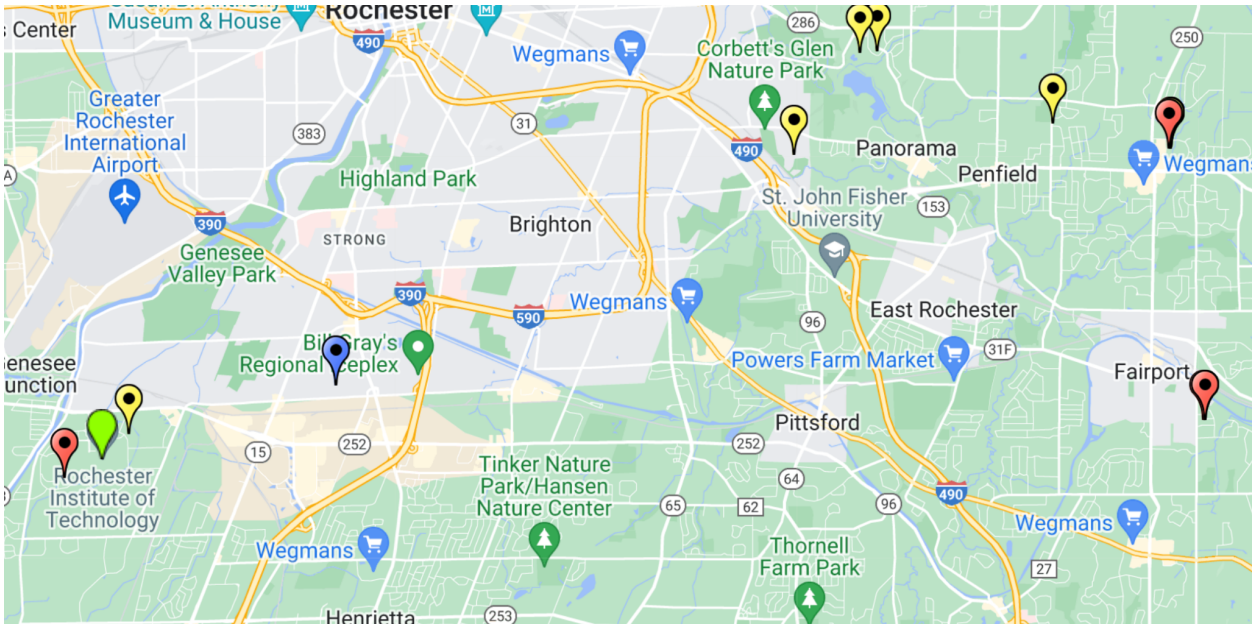
g. Implementation details:
We went through the code to understand what we already have. We started making small changes to find out the missing pieces. We first tested the code to check if it runs for a single gps text file. Once that was done, we went ahead to collect all the data at once.

Data Collection:
We first parsed all files at once from all folders and wrote it to a global text file, and then let all the preprocessing happen on this global file with data from all the months. Instead of doing it this way, we could definitely make use of threads so that this process is a bit faster.

We took inspiration from the remove_redundant_points method to figure out the latitude and longitude of a pair of points.We used this and the timestamp associated with these to figure out the stops. We made use of a dictionary, where latitude and longitude of a point is a key. If we find the same point with a higher parking time (difference in timestamp), we are breaking the ties by considering the stop with higher parking time.

We used the delta method from the datetime library to find out the time difference between two given points. We made use of the seconds attribute in datetime delta. This gives us all the time differences in seconds.

i. Currently, we have 84 stops according to the implementation we have done. We have marked them according to the colors. We found very few red stops in our implementation



j. Conclusion

We took inspiration from each of the assignments we worked on. We took the idea from one of our initial assignments to use threads and glob.glob to read multiple directories in a faster manner.

We learnt a lot about data cleaning from each of our assignments and extended the knowledge here to identify what data might be useful. We followed the pointers mentioned in the problem description and the existing code provided to us and successfully found the gaps. This was a great learning experience as we were not working on any testing data but actual data collected from actual gps

sensors. This made us familiar with the different device issues that might create a need for different types of data cleanings.

This also made us think about the ethics involved in working with the real data and what factors we need to consider even when working on educational projects.

It was really fascinating to work with the kml files and see our result on the google maps! It was fun to get that out on google earth too.

Any data present on the internet can be used to build products that help commercialize the businesses. This location data is already used by google, other platforms to suggest coffee shops, gas stations, restaurants whilst traveling. A person's stops can tell a lot about a person, for example if a person's car is parked a lot at a school, it's likely they have kids, and hence its good to show them school supplies advertisements. If a person stops a lot at vet, pet shops. It's good to show them pet food advertisements.