Project 4: Geolocation

In this project you'll work in teams of 2 to create a system which predicts location (latitude and longitude) based on social media behavior.

The Story

In this project you will help the fictitious social network company MyFace+. Users on MyFace+ connect with one another and publish posts online. Despite the

infrastructure, development, and maintenance costs necessary to make MyFace+ run MyFace+ is completely free to all users. In order to keep the company afloat, executives at MyFace+ have decided to sell user data. Much of the user data that companies/governments are interested in purchasing (education level, political alignment, purchasing habits, etc.) is fairly straight forward to gather — MyFace+ can simply ask its users. Other data, like geographic location, can be gathered directly from the application without inconveniencing the user with annoying questions. Richer data, like what the user watches on TV and what ads the user mutes or listens to is something MyFace+ is very interested in. They have not yet entered this arena for fear of patent infringement (US20180167677), though.

By knowing a user's location, MyFace+ can ping nearby friends, suggest local deals, and alert authorities if the user is in danger. Some users, however, do not understand the value that MyFace+ provides by telling their Partners* where the user is currently residing. These users use 3rd party software to hide their location when they post. Focus groups have indicated that users cite "fear of political persecution", "isn't privacy still a thing?", and "I'm a reporter" as leading reasons for this choice.

Some users use their phone's setting to turn off GPS, or use a GPS-spoofing app. This is not a problem for MyFace+. Most users are connected to a wifi network and as long as some users on the same network have GPS enabled, MyFace+ knows the location of the network. Similarly, if a user is connected to a cell tower, MyFace+ can look up the location of the cell tower. Some users, however, utilize distributed encryption-based anonymity networks (think: Tor).

MyFace+ is interested in improving the MyFace+ experience for such users. Beta-testing attempts based on asking the user to manually report their location resulted in unreliable data collection for unknown reasons. As such, MyFace+ seeks to infer user geographic location based on posting behavior and social connections. MyFace+'s legal team points out that a possible (although not verifiably intentional) side effect of this plan is an increase in MyFace+'s revenue from selling user data.

You will be creating a system that predicts a user's most common location.

*"Partners" includes but is not limited to anyone willing to spend 0.005 USD per user profile — bulk discounts may apply.

Data

We have data for 57,562 users.

For all but 1,000 of them, you'll have the location the most commonly post from. Your task is to infer the locations of the remaining 1,000 users (we'll refer to these users as the test set).

Social Connections

The MyFace+ social network is undirected. This means that if user x is connected to user y, then user y is equally connected to user x. (Think of being "friends" with someone on Facebook as undirectional, whereas "following" a twit on Twitter would be directional.)
The file `graph.txt` holds 210k edges. Each is a single line containing two user ids. So the line:

```
1 52781
```

means that there is an edge from user `1` to user `52781`. Note that each of these 210k edges appears in each other. That is, if the line above is present then there is another line somewhere that's:

```
52781 1
```

Posts

For all users, we have the following information:

1. Hour1.
   The most frequent hour of the day that the user posts.
2. Hour2.
   The second most frequent hour of the day that the user posts.
3. Hour3.
   The third most frequent hour of the day that the user posts.
4. Posts.
   The total number of posts made by the user.

All above times are in UTC 24 hour time. Values are 00 to 23 (inclusive), or 25 to indicate a missing value. So, for example, if the user's most frequent hour of posting is 1:00PM UTC, then the value for Hour1 would be 13.

In addition, for the users not in the test set, we have latitude and longitude. To calculate `lat` and `lon`, we took the most frequent (latitude, longitude) pair from which the user posted. All latitude and longitude coordinates were truncated to 3 decimal places prior to computing the most frequent.
Note that this means that `lat` is not necessarily the most frequent latitude that the user posted from, but instead the pair (`lat`, `lon`) is the most frequent pair from which the user posted.

The file `posts_train.txt` contains information about when users post. The first line is a header, specifying the order and meaning of the columns. Note that each line contains the userid (the same userid used in `graph.txt`) that that line is describing. For example, consider the first few lines of the file:

```
Id,Hour1,Hour2,Hour3,Lat,Lon,Posts
1,18,19,20,28.6,77.2,38
2,13,07,08,18.490,73.912,13
```

Line 1 is the header.
Line 2 tells us that user `1` most often posts at hour `18`, their second most common posting hour is `19`, their third most is `10`. They most often post from location (28.6, 77.2), and have posted 38 times in total.

Null Island:
Note that some users may have successfully/accidentally disabled location metadata in their posts. These will show up as `lat` being 0.0 and `lon` being 0.0. (As an aside, this is referred to as "Null Island": https://en.wikipedia.org/wiki/Null_Island, and often causes problems when people perform analysis on geo data.)

Test Data
In `posts_test.txt` you'll find the data for the 1,000 users in the test set.
Here we have the same data as for the training set, but with `lat` and `lon` missing. Your task is to infer these values.

How to Submit your Predictions

This project is run as a Kaggle competition.
The link to the competition is here: https://cosc247f18.page.link/P4_Competition
Please do not share this link with anybody, as we do not want internet randos joining in on the fun.

You'll submit a file in the format of `submission-example.txt`. This file has a header (the first line, which is `Id,Lat,Lon`). After that, each line should contain three values: the userid you're predicting, the predicted latitude, the predicted longitude. For example, the submission:

```
Id,Lat,Lon
18,35.699,139.578
65,40.804,-73.951
```

Predicts that user `18` is at location `(lat = 35.699, lon=139.578)`.
To make your submission, upload such a file to the Kaggle competition cite.
Note: **you may submit up to five times per day.**
(This number may increase on the last day or two.)

Performance Measure

Your score, given a set of predictions, will be the Root Mean Squared Error (RMSE) of your predictions:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y^{(i)} - \hat{y}^{(i)})^2}$$

~~This is computed for your Latitude predictions, and separately for your Longitude predictions, and then summed.~~
[Edit: Dec 6] The above is not correct. Kaggle actually computes the RMSE as follows:

$$\text{RMSE} = \sqrt{\frac{1}{2n}\sum_{i=1}^{n}(\text{lat}^{(i)} - \hat{\text{lat}}^{(i)})^2 + (\text{lon}^{(i)} - \hat{\text{lon}}^{(i)})^2}$$

$$= \frac{1}{\sqrt{2n}}\sqrt{\sum_{i=1}^{n}(\text{lat}^{(i)} - \hat{\text{lat}}^{(i)})^2 + (\text{lon}^{(i)} - \hat{\text{lon}}^{(i)})^2}$$

Note: This isn't the perfect measure for these sorts of predictions. However, it's a simplification designed to make your life easier. You're encouraged to think about what would be a better measure of loss if you were to actually deploy a system that predicts user latitude and longitude.

What Else You'll Submit

In addition to your predictions, which you'll submit on Kaggle, you'll submit the following on Moodle. Note: Each **individual** should submit. It's fine if you submit identical files as your project partner, though. If you and your project partner submit different files, only one submission (chosen arbitrarily) will be graded.

1. Your team document.
   Submit the following under the "Kaggle Registration for P4" assignment on Moodle.
   This is a text file named `team.txt` that must contain three lines.
   The first is your kaggle team name. The next line is for you, and the third is for your partner.
   Each line must contain: your email and your kaggle username separated by commas.
   So for example, you might submit:

```
The Sunrise Above A Mountain Upon Which A Single Oak Tree Grow
s
salfeld@amherst.edu, salfeld
prohaxor@n/a.com, sysadmin
```

2. Your code.
   Submit this under the "Project 4" assignment on Moodle.
   Submit a single .zip archive that contains all your code.
   In addition, it should contain a file `README.txt` that describes your code, and how to run it.
3. Your writeup.
   Submit this under the "Project 4" assignment on Moodle.
   Submit a file `team_name.pdf` (with `team_name` replaced with your team name).
   Details below.
4. `Readme.txt`

Note: This file must be a plaintext `.txt` document. No `.pdf`, `.doc`, `.docx`, `.rtf`, `.ps` or other filetypes will be accepted.
In it, fill out the following template by replacing the strings surrounded by `<` and `>` appropriately.

```
Resources used:

  <List all resources you used here (e.g., specific websites, the textbook, who you talked to in the class>

Time spent on assignment: <roughly the number of hours you spent on the assignment>

On a scale from -2 to 2:

  How hard was the assignment?

  <-2 being super easy, 2 being super hard, 0 being just right.>

  How much did you learn from the assignment?

  <-2 being learned nothing, 2 being leared far more than you expected, 0 being what you would expect from an assignment of this magnitude>

  How much did you enjoy the assignment?

  <-2 being you hated having to do it, 2 being you would have done it even if it was not assigned, 0 being that in the middle.>

Additional notes: <Anything else you'd like me to know when grading it>
```

Note: How much time you spent on the assignment, and your answers to the questions will not affect your grade. This data gathering is for my own purposes and is mostly for use in future semesters. Your responses will be kept anonymous from your classmates, although I may release aggregate statistics.

The Writeup

You'll turn in a PDF where you answer the following questions.
Note that this may enter your portfolio when you go off and apply for jobs or grad school. And/or people may ask you about it during interviews. In addition, if you ever ask me for a letter of reference, I'm going to go back and look at this document. So,

it's in your best interest to accurately record what you did in enough detail to remind yourself (or me) how awesome a job you did on this project.

Q0:

1. What is your team name?
2. What is your name, email address, and kaggle username (for both team members)?
3. What is your score on the public leaderboard?

Q1:

How does your system work?
Describe in detail how your predictor makes its predictions.
Also describe why you chose what you did.

This should be a maximum of 2 pages (US Letter, 9pt font, single space, normal margins).

Q2.

If you had this same data, but an extra six months you could dedicate to building the best prediction system you could, what would you do?

This should be 0.5 to 1 page.

Q3.

If you could get more data from MyFace+ to help infer user location, what would you want and how would you use it?
Focus on data that a company like MyFace+ probably already collects anyway (e.g., login times, post content, number of seconds the user has each individual post on screen (yes, everybody collects this), links clicked, etc.).

This should be 0.5 to 1 page.

Some Notes and Logistics:

Special thanks to Prof Ben Rubinstein at the University of Melbourne for the data sets.
Smiley face in MyFace+ logo from:
https://github.com/emojione/emojione/graphs/contributors.

Obtaining Files

All data files mentioned above are available from the Kaggle Competition site.

Multiple Submissions

You can submit on Kaggle up to 5 times per day.
A "day" begins at 7:00PM Eastern Time (Midnight UTC).
On Kaggle, you can select which of your past submissions you want evaluated for your final score. Note: this may **not** be the one that performs best on the leaderboard — see below.

Score from Leaderboard

The leaderboard maintains **two** scores. The public score, and the private score. Each is computed on 500 examples in the test set. What you can see is the public score. What you'll be graded on is the private score.

Note: You will be graded based on your **private** score. **Not** your public score. If you reverse engineer the exact target values by making many (strategic) submissions and seeing how your public score changes, you might be able to get an extremely good public score. However, your private score would likely be horrible.

In other words: **Try not to overfit to the public leaderboard test set.**

Data is often Messy

Are you sure all users exist in the dataset?
Are you sure all users have ≥ 1 connection?

Are you sure there are no missing values?
Are you sure there are no outliers?
These and other assumptions can often get you into trouble when working with data.
Check your assumptions.

That said, this particular dataset is actually pretty clean compared to what you'd normally encounter in the wild. But still, be careful.