Q0:

1. Team Name: Boolnuts

2. Name, email address, and kaggle username for both team members:
- Will Friedrichs, wfriedrichs20@amhest.edu, Will Friedrichs (Kaggle Name)
- Ooga Nam, ynam20@amherst.edu, Yongwook Nam (Kaggle Name)

3. Public leaderboard Kaggle score: 27.588

Q1: Detailed system description

Our system uses a neural network to make predictions, more specifically a Multi-Layer Perceptron (MLP) regressor, optimized by the Adaptive Moment Estimation (adam). This regressor takes in seven features for each user:

1. The most frequent hour of the day the user posts (0 to 24)
2. The second most frequent hour of the day the user posts (0 to 24)
3. The third most frequent hour of the day the user posts (0 to 24)
4. The total number of posts made by the user
5. The median of the latitude coordinates of the user's friends
6. The median of the longitude coordinates of the user's friends
7. Median number of total posts of the user's friends

Features 1-3, regarding hours when users post, is useful particularly for longitude, as different users live in different time zones and post accordingly. Features 4 and 7 would likely be useful in that some countries or parts of the world likely have different patterns of social media use, and a higher total number of posts could correspond to higher-income locations such developed countries in the "global North" (US, Canada, Europe). Features 5 and 6 are particularly helpful in determining the location of users because users are likely to have friends located close to them. We decided not to use the medians of the users' friends' data, such as most, second most, and third most frequent posting hour so that we would avoid the overfitting problems we would likely encounter by using too many variables.

Because the original training and testing data did not include features 5, 6, or 7, we added this information in our code. First, our program adds two empty columns to our training set and test set arrays "X_tr" and "X_te". Then, it loops through each row of "X_tr". At each iteration, it calls a method that uses the data from "graph.txt" to determine its friends, and then based on the IDs of those friends, finds the median of the friends' longitudes, latitudes, and number of posts, finally adding those pieces of data to the last three indexes of the 'i'th row of "X_tr". A similar loop adds the medians of friends' longitudes, latitudes, and number of posts to each row of "X_te". The averages of latitude and longitude do not include friends with locations on "null island" (where latitude and longitude are unknown and thus listed as '0'); if a user has no friends, or the user's only friends are on null island, a random but reasonable guess at latitude and longitude is added for features 5 and 6 (the data would be skewed further if these values were left at zero).

The data from X_tr and y_tr (y_tr being made up of training set latitude and longitude data) is then used in the MLP regressor. MLP is a type of neural network that uses backpropagation (meaning the error gradient for neurons in the network is calculated from the last layer to the first). In MLP, there are one or more "hidden layers" of nodes that do not activate.

We used GridSearch to experiment with different parameters, including the type of solver ("adam" or Stochastic Gradient Descent), the number of hidden layers (80, 90, or 100), and the maximum number of iterations that the MLP regressor would run (200, 300, 400, or 500). GridSearch yielded "adam" as the more successful solver, 80 as the optimal number of hidden layers, and 200 as the best maximum number of iterations.

Q2: Our plan to improve our system given additional time

Given more time, we would explore different types of learners to fit the data. In our current project, we experimented with K Nearest Neighbors and Decision Trees before eventually deciding on the MLP regressor, but in the future many other options would be available. Among these are Support Vector Machines, Bayesian regression, and a priori learners. Once we find a learner that seems to perform optimally, we would use various ensemble meta-learners to maximize the chosen learner's performance. For example, the ensemble method known as boosting involves partitioning the data into smaller samples and learning "weaker" learners whose results are aggregated to form a prediction. Doing so can help mitigate the impact that data set variance has on our eventual prediction.

Additionally, we can pre-process the data to our advantage. While we used a min-max scaler ranging from 0-1, other pre-processing methods exist (such as standard scaling). Furthermore, we synthesized features into the training set (specifically the latitude and longitude coordinates of friends). However, we can always synthesize the training set with the remaining 3 friend features one at a time, and see which features/combinations thereof best predict a given user's location. It could be the case that these synthesized features improve performance even more in a lifted feature space (for instance top posting hours could have a quadratic relationship with a given user's location), so we could also experiment with a variety of lifting functions and see how our loss is affected.

Q3: Additional useful data from MyFace+

Assuming location data from wifi networks and cell towers is already utilized to find location data of users by MyFace+, additional helpful resources would include the residence information set by users in their profiles, login times, and searches for users on other social media sites to find location information there.

Residence information is available on user profiles for many social media sites—many facebook users, for example, have this information publicly available. If MyFace+ has not implemented this feature, opportunities to collect more accurate location data for users would be an excellent reason to add it (in addition to enhanced user experience).

Login times would be helpful for the same reasons that the most, second-most, and third-most frequent hour of the day users made posts are helpful for determining information. Users are more likely to use social media in the afternoon and evening on weekdays, and these peaks vary based on timezone. Knowing the timezone in which a user most likely resides helps MyFace+ determine the longitude data of its users.

If a MyFace+ user attempts to make posts without divulging location data, there are other sources of location information available. Most social media users use more than one social media site, and public information is often available via a quick google search. MyFace+ could make use of Internet-searching algorithms, which could search for location or residence data on LinkedIn, Facebook, Twitter, and Instagram.

MyFace+ also keeps track of the links its users click on. If a user clicks on a link related to a specific location, for example some clickbait advertisement-article titled "this Amherst, MA company is disrupting a $200 billion industry," it's likely that the user's location is near Amherst MA.

These pieces of information could be used as additional features to be added for each user. Comparing the performance of models using different combinations of features and different weights used for the different features would highlight the parameters that would better predict user location.