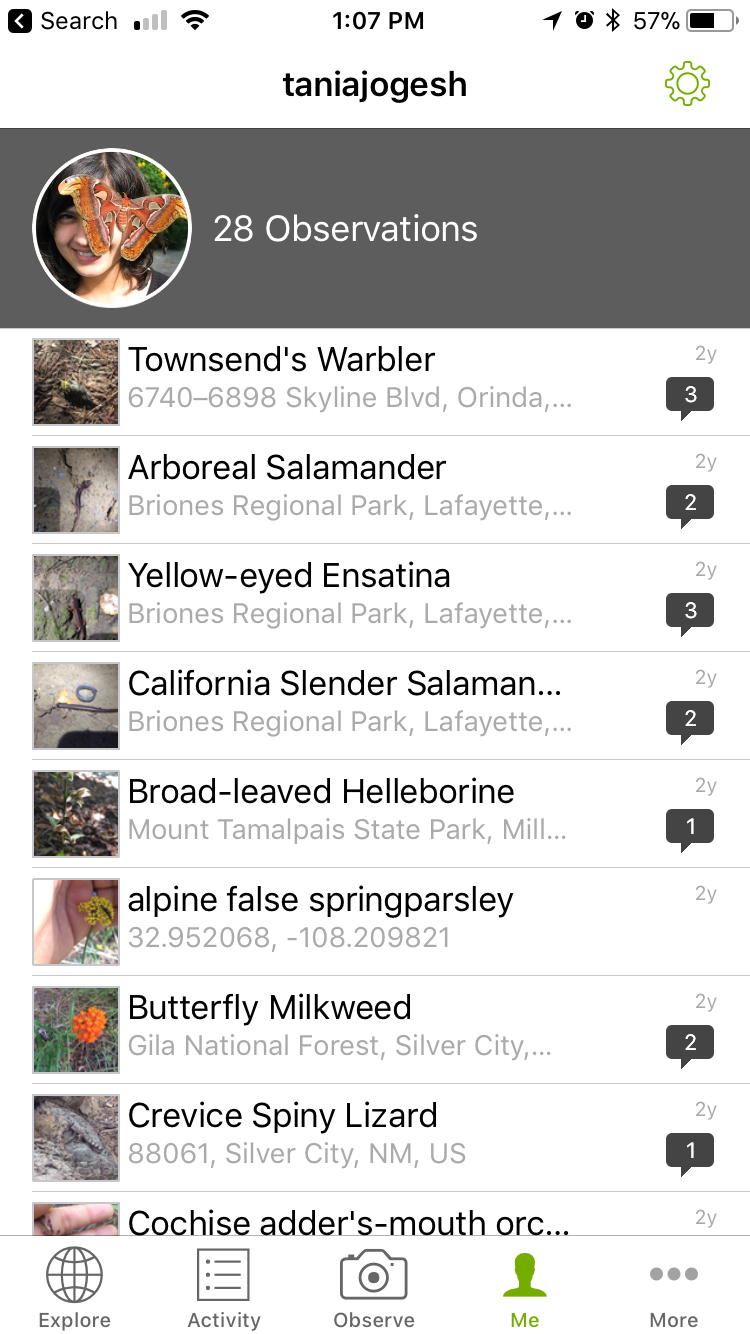
CAPSTONE 2

1. **What is the problem you want to solve?**

We share our plant with 20-30 billion species, and each individual species is a treasure trove of data. The food we eat, the plants in our gardens and even some of the most important medicines in human history are a product of other living species. Biodiversity is akin to a library that we are only beginning to understand and this library can hold the key to some of the most important problems in the future. It is then, imperative that we effectively document this data, to understand, which species share our planet and where there occur so we can catalog the valuable information around us.

iNaturalist is a mobile application run by the California Academy of Sciences that allows people to upload photos and information on species (plant and animals) they observe in their daily life. It is a fantastic application that relies of regular people, who look at birds in their garden or hike the National Parks of America to document and collect biodiversity data. However, not everyone is an expertly trained biologist, and some organisms are inherently difficult to identify.

Here, I develop a classifier that predicts if an identification posted via the Citizen Science Biodiversity app iNaturalist (https://www.inaturalist.org/) is likely to be correct. iNaturalist is a mobile application that allows users to upload photos and information on natural history observations (plant and animal sightings). The online community can subsequently verify observations uploaded by users. However, there are a lot of observations and not all plants/animals/fungi are easy to identify. Instead of relying solely on the community to verify an ID, my classifier will help iNaturalist decide which observations are more likely to be correct or misidentified.

1. **Who is your client and why do they care about this problem? In other words, what will your client DO or DECIDE based on your analysis that they wouldn’t have otherwise?**

My client is iNaturalist and they can use this classifier to determine which observations need additional expert identification and can tag those that are likely to be correct.

1. **What data are you going to use for this? How will you acquire this data?**

iNaturalist’s user data and observations are freely available to the public and can be queried and obtained via an API. A subset of these data include whether or not an observation was accurately classified by experts in the community so the data are already labeled for classification. I downloaded 10,000 records from their database (I can download more but am limited by computing power) for observations from 2016-2017.

1. **Approach to solving the problem**

* I acquired the data via an API in json format and I downloaded the data with each observation in a single json file. I did this to make it easier for spark to infer the schema and so that the data were imported as a single observation per row in the Spark Data Frame.
* I then selected the features of interest into a Spark Data Frame table. This step was necessary as the data is nested and the features of interest are numerical or string values that are stored as list arrays. I essentially used the query to flatten the data structure.
* Before exploring features associated with correct or incorrect identifications on iNaturalist, I plotted the most commonly observed species and the most commonly observed taxonomic groups.
* Using groupby, count and crosstab, I explored the relationship between all the features of interest and the accuracy of identification
* I obtained information on the location state via reverse geocoding and used this to plot observation accuracy on a map. State will be used as a feature in the machine learning models

**Feature engineering**

* The data contain a lot of categorical features. To deal with these I tested two approaches 1) labeled the data numerically 2) used a one hot encoder to get every category encoded.

**Machine learning classifiers**

* I tested three classifiers: 1.) Support Vector Machines 2.) Random Forest 3.) Gradient Boosting classifier
* For each of these I optimized hyper-parameters individually or with a grid search
* The Random Forest was the best model in terms of speed and accuracy