Wesley Kwong
PBHLTH 244 Big Data: A Public Health Perspective

Project 3: Parkinson

Abstract

The Predicting Parkinson's Disease Progression with Smartphone Data competition by The Michael J. Fox Foundation for Parkinson's Research presented Kaggle users with a wealth of smartphone data collected from 9 patients diagnosed with Parkinson's and 7 seven patients that were not diagnosed with Parkinson's. The goal was to classify patients on whether they have Parkinson's or not and/or quantify Parkinson's symptoms to measure disease progression. To predict Parkinson's status, I trained a stochastic gradient descent model using acceleration data and achieved a training accuracy of 99.999% (11,283,411 training rows) and a test accuracy of 100% (1,253,713 testing rows). I also used the GPS data to investigate where are these patients by building an interactive html map. I discovered that a majority of the patients are located in Massachusetts and around the East Coast. A couple of patients are in California. Patients were also briefly in Ohio as well as Canada.

Data Processing

Smartphone data was collected from 9 patients diagnosed with Parkinson's and 7 seven patients that were not diagnosed with Parkinson's. The data provided for this project was provided in the Kaggle's competition Predicting Parkinson's Disease Progression with Smartphone Data by The Michael J. Fox Foundation for Parkinson's Research. The class provided a data pipeline to process the raw data into a mjff.Rdata file which contained acceleration, audio, battery, compass, and GPS data frames. Each data frame had a column for Parkinson's status. I exported these data frames into separate csv files so I could utilize Python packages to analyze and model the data.

While there were only a few people used in this study, the amount of data pooled together was extremely large since each individual patient had their data recorded multiple times. I wanted to narrow down my search to identify which data frame had the best set of features to predict Parkinson's status. I first removed the battery data frame since I felt the amount of stored energy in the phone was not relevant for prediction. Then, I randomly sampled 1,000 rows (with 50% having Parkinson's) from the remaining data frames for exploratory data analysis. I first plotted histograms and violin plots, separated by Parkinson's status, to identify potential features for my model but I did not see any significant distribution difference. I also lacked domain knowledge to identify potentially informative features. So, I decided to use the randomly sampled data and perform logistic regression with 5-fold cross validation to identify which data frame was the most informative. With one-third of the data selected for testing, the best model was trained on the acceleration data (Table 1). This was exciting since one of the hallmarks of Parkinson's disease is slow movement.

Table 1: Logistic Regression with 5-Fold Cross Validation Trained on Sampled Data

Data Frame	Model Accuracy
Acceleration	74.8%
Audio	50.3%
Compass	61.5%
GPS	52.7%

Using the randomly sampled data from the acceleration data frame, I then investigated which classifier to best model my data. I settled on the stochastic gradient descent (SGD) classifier. While SGD performed poorly with an accuracy of 62.1% compared to logistic regression accuracy of 74.8%, the SGD model allows me to continuously improve my existing SGD model with new data. This would allow me to train my model on big data relatively quickly since I could divide the training data into chunks and train my model one chunk at a time. Whereas in the logistic regression model, I could only train it on the entire data set.

Next, I looked into which features best predicted Parkinson's in the acceleration data frame. Since I was working with big data, I felt that having 26 features was unnecessary and would prove to be a massive computational burden. With the lack of domain knowledge, I used the FeatureSelector package to identify the best features. This package trains a gradient boosting model on the randomly sampled acceleration data to weight the features' importance for classification. I settled on using the top 5 features for the SGD model: number of samples, mean acceleration in the x, y, and z axis, and the time difference from last recording point.

To begin analysis on the whole acceleration data frame, I imported the data by dividing the data into 1 million chunks. Then with each chunk, I selected my 5 features and the Parkinson's outcome feature. Then, I removed any rows that had missing information. Pooling the chunks together, I had a total of 12,537,124 rows of acceleration data with 65.0% of my rows were Parkinson's positive. This was good to see since 56.3% of the patients were Parkinson's positive so they had a roughly similar distribution. Then, I standardized the data to ensure the mean of each feature was 0 and had a standard deviation of 1. This standardization would allow the SGD model to converge in gradient descent more rapidly and reduce the computational burden.

Analysis

I implemented a test train split where the test size was 10% of my data (11,283,411 for training and 1,253,713 for testing). Since the data was very large, I felt that the traditional 70% training and 30% testing was not necessary here. 10% allowed me to still have a sizeable testing sample. Since the training data was still very large, I split my training data equally into 10 minibatches (1,128,342 rows). Then, I trained my SGD model on each minibatch one at a time. I was surprised by the very high accuracy on the first minibatch since the SGD did not perform well on the 1,000 randomly sampled data. Overall, the SGD model accuracy improves slightly with each minibatch and the training accuracy averages to 99.999%. Minibatch 10 shows that while the model is very accurate, it isn't completely perfect (Table 2). I then applied my trained SGD model onto the testing data and received 100% accuracy.

Table 2: SGD Model Training Accuracy on Minibatch Training Data

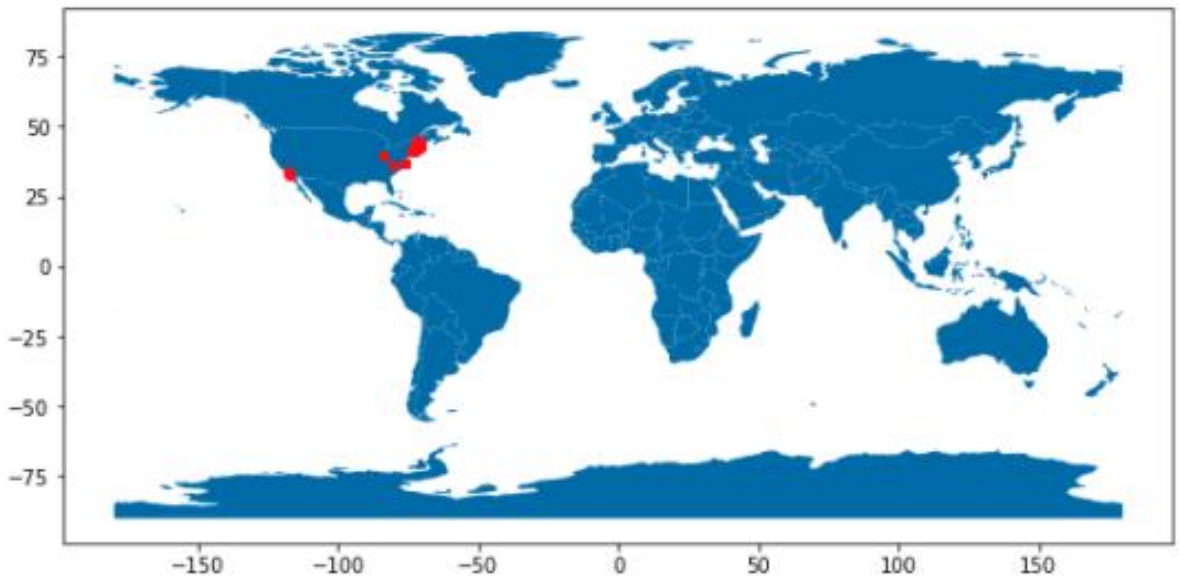
Minibatch Number	Training Accuracy
1	99.99185529978479%
2	99.99820089864669%
3	99.99887445383315%
4	99.99992909945405%
5	99.99997341229527%

6	100.0%
7	100.0%
8	100.0%
9	100.0%
10	99.99997341229527%
Total	99.99888065763093

Question of My Own

Since GPS data was provided, I wanted to investigate where these patients came from by plotting them on a map. I randomly sampled 1,000 rows from the GPS data frame that did not have any missing values. Then, I used the geopandas package to use the latitude and longitude to plot each row as a point on a world map (Figure 1). The map indicates that all the patients came from the US. It is interesting to note that patients either came from the East Coast or the West Coast instead of just one area.

Figure 1: GPS Locations of Patients on a World Map



To get a closer look at where these points are exactly, I then used the gpxpy and folium packages to generate an interactive html map with each point specified. Most of the East Coast GPS points are located in Massachusetts, followed by New Hampshire. Other states on the East Coast that have points are New York, Connecticut, Rhode Island, Virginia, and North Carolina. On the West Coast, there are a couple of points in Los Angeles and San Diego of California. There is also one point in Ohio. Interestingly, there are 3 points in Canada near Montreal. It is likely a patient crossed the US-Canadian border and walked around for a bit.

Reference

https://www.kaggle.com/c/predicting-parkinson-s-disease-progression-with-smartphone-data/data