API 222 Prediction Competition

Machine Learning and Big Data Analytics

Due Before 11:45 am on November 22, 2019

Under the Kennedy School Academic Code, this assignment is a Type II assignment. You are encouraged to work in a study group, but must submit your own hand- or type-written solutions. It is not acceptable to work on one electronic document as a group and submit identical, or nearly identical versions.

1 Overview

The U.S. Department of Homeland Security (DHS)'s annual budget is approximately $40 billion. A substantial portion of these funds are allocated to implement decisions to make our country and its citizens safe from terrorism. These decisions are important and also challenging, in part because of the uncertainty in the outcomes of the potential terrorist attack. It is critical to make decisions on how much resource should be allocated, because spending too much resource in the low-risk event can reduce the valuable resource that could be spent on other high-risk events. Thus, predicting (a) which potential attack is likely to be successful and (b) the magnitude can help the decision-makers.

The Global Terrorism Database (GTD) is an open-source database including information on domestic and international terrorist attacks around the world from 1970 through 2018, and now includes more than 190,000 cases. For each event, information is available on the date and location of the incident, the weapons used and nature of the target, the number of casualties, and–when identifiable–the group or individual responsible. You can find out more about the data and the codebook for more information about each variable.

2 Assignment

Your objective is to develop a model that accurately predicts whether the terror attack is successful (`success`) and how many U.S. citizens are killed (`nkillus`). The data defines **successful attack** as the following: *"Success of a terrorist strike is defined according to the tangible effects of the attack. Success is not judged in terms of the larger goals of the perpetrators. For example, a bomb that exploded in a building would be counted as a success even if it did not succeed in bringing the building down or inducing government repression."* You can read more about the way the data defines variables in the codebook.

Your score for predicting `success` will be determined using the macro F_1 Score on a holdout set of data. Your score for predicting `nkillus` will be determined using the mean squared error on the holdout (i.e., test).

The macro F_1 score is an aggregate measure for classification errors. It calculates two F_1 scores (because the target can take two unique values) and then averages them. For example, the first F_1 score for success/failure, will treat success as positive and failure as negative. The second F_1

score will treat failure as positive and success as negative. The F_1 score for a binary classification problem is defined as:

$$F_1 = \frac{2 \cdot \text{ true positives}}{2 \cdot \text{ true positives} + \text{ false negatives } + \text{ false positives}} \tag{1}$$

Professor Saghafian will announce in class the first, second, and third place students who have been able to achieve the highest macro F_1 scores and the mean squared error using a Machine Learning algorithm. Your score will also be determined by an accompanying write up that clearly explains the process you went through of choosing a model and describes your final approach.

You will be required to submit working code (we must be able to run it by changing only one file path line), a CSV file of predictions for a hold out set of data that will be released 48 hours before the submission deadline, and a write-up between 1.5 and 2.5 pages single spaced, size 12 font Times New Roman, 1 inch margins. **Your write-up should be geared toward a member of the U.S. Department of Homeland Security staff who has some familiarity with Machine Learning but who is not an expert.** The goal of the written portion of this assignment is to get you familiar with explaining the process of model selection to a broad audience in a clear way. This will be an important skill in facilitating the adoption of high-performing yet new or unfamiliar methods in the types of organizations where many of you will work after graduation.

3 Prediction Competition Rules

You may not use any data other than the data provided by the course instructors in developing your model. **Anyone who uses any additional data will receive zero credit for the assignment and faces possible disciplinary charges.** However, you may do whatever you like with the data provided, such as generating new features through interactions, non-linear transformations, etc.

4 Grading

This competition is worth 15% of your course grade.[1] Therefore, the assignment will be worth 15 points, which will be broken down into three evenly weighted components (e.g. 5 points each):

1. The write-up, which has
   (a) A thorough description of the process you took to arrive at your final model
   (b) A clear description of your final model, including any data manipulation or feature engineering
   (c) A discussion of your approach as it pertains to algorithmic bias and transparency. This section should contain some numbers illustrating how your model performance varies along salient characteristics, such as demographic and geographic characteristics. It should also concretely discuss the trade-offs your model makes between predictive performance and interpretability / transparency.
2. Clean code that:
   (a) Trains your model

[1] Please note that the points you received from problem sets and the midterm will be scaled such that they will consist of 15% and 25% of the final grades, respectively.

(b) Produces a CSV of predictions for the holdout data

The teaching staff must be able to successfully run the code by changing only one line of the file path. **Code that we cannot run without further edits will receive at most 1 of the 5 possible points.**

3. A CSV file of predictions. We will order students in terms of predictive accuracy on the holdout data.

   (a) Students in the top one-fifth of the class on this measure will receive 5 out of 5 points on this component.

   (b) Students in the second-to-top one-fifth will receive 4 out of 5 points on this component.

   ⋮

   (c) Students in the bottom fifth will receive 1 out of 5 points on this component.

   We will provide you with a sample submission CSV, which will have two columns:

   (a) `Id` - An ID column that maps to the holdout data released 48 hours before the submission deadline

   (b) `Prediction_1` - A column that contains your predicted success for each event in the holdout sample

   (c) `Prediction_2` - A column that contains your predicted number of deaths of U.S. citizens for each event in the holdout sample

   You should submit a CSV file with the same three columns, and those columns should be named `Id`, `Prediction_1`, and `Prediction_2`. The filename should be `lastname_prediction.csv`, where you replace `lastname` with your actual last name (for example Prof. Saghafian's file would be `saghafian_prediction.csv`). **Any submissions that do not include these three columns (with the correct column names) or have the wrong file name will receive zero points on this section.**

5 Data Description

Number of Observations	181,691
Number of Predictors	135
ID Variables	`eventid`
Response Variables	`success`, `nkillus`

6 Useful Hints

- Use the following commands to load the data:

```
setwd("insert your working directory path")
library(data.table)
terror_data <- data.frame(fread("terror_data.csv"))
```

- `eventid` is the ID variable designated to each event. It likely does not have any predictive information.

- Many of the variables are missing. Make sure to address them.

- Make sure the variables are in the right format (numeric vs. categorical). Some variables (e.g., `imonth`, `iyear`) can be interpreted as either. Think about what makes more sense.
- Some variables (`summary`) have a long text entry. You would probably need to remove them. You are welcome to do text analysis if you would like, but this is beyond the scope of our class.
- Some variables contain the same information but in different formats. For example, `country` and `country_txt` contain the same information.
- We do not recommend including all (or too many) variables from the beginning. Including all variables, especially a lot of categorical variables with a lot of categories, can make your R run forever or crash. One possible approach is to start with a reasonable model with numeric and/or binary variables and increasingly explore the other categorical variables. Another approach is to run your model in a small subsample of the data first.
- Again, it will make your life easier to start with a simple model.
- You need to think about overfitting and the trade-off for the prediction in the test data, since that is what your model will be evaluated on.
- Finally, note that CAs and the TF can help if you face difficulties. Please utilize their office hours if they can be of any help.