**TABULAR PLAYGROUND SERIES – MAR 2022**

The winning score was a single lgbm with no post-processing. First, I created a substantial amount of likelihood encodings and lag features. Unlike most public kernels, I used many other features than the medians. Second, I found the best subset of the features using Optuna.

I used to create both the lag features and the encodings. For the lagged features, I found means, variances, medians, minimums, maximums, and 1 interval shifts. I took this for every x-y-direction combination on both the day and weekday. I used both 3,5, and 10 day rolling windows and expanding windows.

The likelihood encodings took the minimus, maximums, medians, variances, and means for every xy and x-y-direction combination at all hour-minute combinations.

 I used Optuna to pick the best features. Specifically, I used trial.suggest\_categorical(feat\_in\_question, [True, False]), for each of the features I considered. After 300 trials, Optuna got very good at finding the best features.

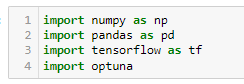
**DATASET**

To forecast twelve-hours of traffic flow in a major U.S. metropolitan area. Time, space, and directional features give you the chance to model interactions across a network of roadways.

**Files and Field Descriptions**

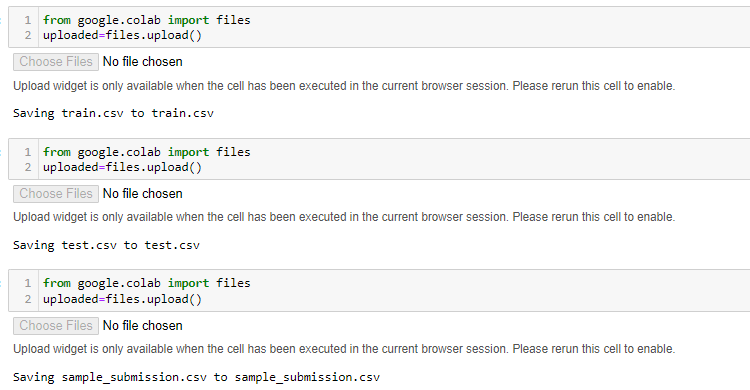
* **train.csv** - the training set, comprising measurements of traffic congestion across 65 roadways from April through September of 1991.
  + row\_id - a unique identifier for this instance
  + time - the 20-minute period in which each measurement was taken
  + x - the east-west midpoint coordinate of the roadway
  + y - the north-south midpoint coordinate of the roadway
  + direction - the direction of travel of the roadway. EB indicates "eastbound" travel, for example, while SW indicates a "southwest" direction of travel.
  + congestion - congestion levels for the roadway during each hour; the target. The congestion measurements have been normalized to the range 0 to 100.
* **test.csv** - the test set; you will make hourly predictions for roadways identified by a coordinate location and a direction of travel on the day of 1991-09-30.
* **sample\_submission.csv** - a sample submission file in the correct format.

**STEP 1: IMPORTING THE NECESSARY LIBRARIES**

****

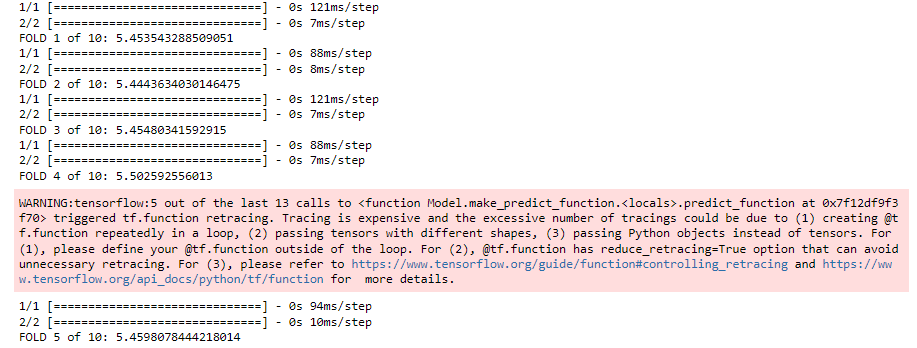
Optuna is an automatic hyperparameter optimization software framework, particularly designed for machine learning. It features an imperative, define-by-run style user API. Thanks to our define-by-run API, the code written with Optuna enjoys high modularity, and the user of Optuna can dynamically construct the search spaces for the hyperparameters.

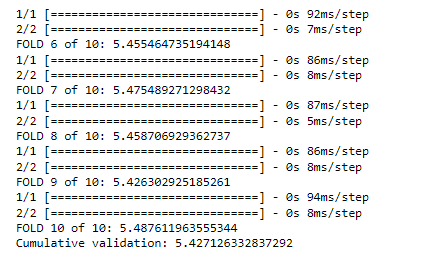
**STEP 2: DATA LOADING**

****

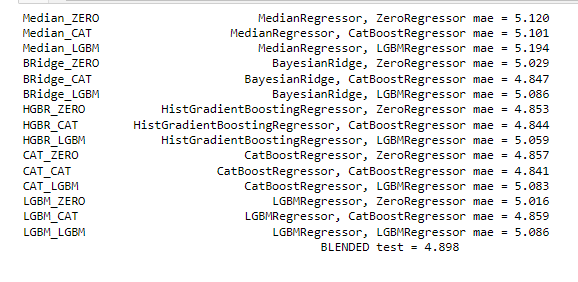
**STEP 3: FEATURE ENGINEERING**

****

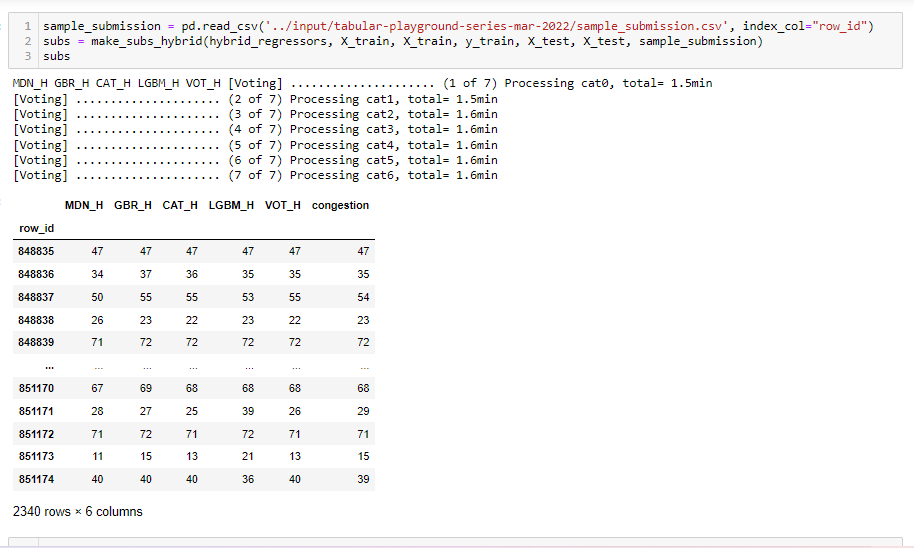
****

****

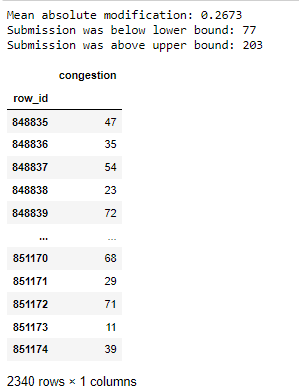
**STEP 4: COMPARE HYBRID REGRESSORS**



### **STEP 5: PREDICTIONS**



# **STEP 6: GENERALIZING THE SPECIAL VALUES**

****