Stats 101c Final Project Report

Team: YYX
group member: Ying He, Ying Luo, Xinru Fang
Instructed by: Prof. Gould

Abstract

The data set for the final project is from the last four years for all fire stations in LA. The goal is to predict the response time of all incident in the past four years. In this project, we should find the best model type and use it to predict the best results by the smallest MSE.

Data prepareation

From original data set, we use the lafdtraining data to build models without missing value NAs, and then use the test_without_response data to create a predicted response by using predictor values.

"row.id", "incident.ID", "year", "First in District", "Emergency Dispatch Code",
"Dispatch Sequence", "Dispatch Status", "Unit Type", "PPE level"
"Incident Create Time (GMT)", "elapsed_time"

Dealing with NAs

Before we trained our data, we first deleted all the missing values.
After we get our best model from training model, we used the mean of our prediction instead of those missing NAs in testing data to make a complete table with row.id and prediction.

External Data Collection:

"ave_response_seconds", "ave.arr", "ave.dis", "aveProcess", "aveTravel", "aveTurnout"
"timeOfDay", **"nDispatch"**, **"nPark"**, **"nUnitType"**

"Turnout.Time": standard turnout time for the fire station by PPE level
"hour": the hour of the Incident Creation Time
"inciCount": how many unit type appear at the incident
"ZIP", **"PopDensity**": each station's zip code and the area's population density
"percentFire", **"percentMed"** ,**"slow.seconds"**: the information of each stations at the map of the LATIMES
"zipGroup", **"unitGroup"**: group the zip and unit

"ave_response_seconds", "ave.arr", "ave.dis", "aveProcess", "aveTravel", "aveTurnout": average value of elasped time group by these related variables.

"Mean_ByHour", **"Mean_ByPPE"**, **"Mean_ByStatus"**, **"Mean_ByUnit"**, **"Mean_ByYear"**, **"Mean_ByZip"**: mean of elasped time group by these related variables.

"nDispatch", "nUnitType": the number of related variable per incident.
"nPark": the number of parks nearby each station.
"timeOfDay": the daily time.

We found and created these variables to improve the measurement of our prediction, because we thought these information are related to the elapsed time of fire stations. If we added more information into our model, the accuracy would be better.

However, after the best models we made from these variables, we have thought about the problem of overfitting. Thus, choosing which variables to predicting the response time is really important. We need to try many times to get our most important variables to predict response time.

BEST MODEL

-Algorithm: Gradient Boosting (package: xgboost)

We use Xgboost to predict the elapsed times, and our object is to minimize squared error. In this model, we removed the variables "row.id" "Emergency Dispatch Code", and "Incident Create Time (GMT)"from original data set.
First, We added all newly created variables from external data collection to training data. Unfortunately, those newly created variables didn't make the best model for our project, so we have concerned about the problem of overfitting. Thus, after we have tried hundreds of times for choosing important variables, we decided to using Xgboost to predict elapsed time on Dispatch Sequence, year, Unit Type, Dispatch Status, PPE level from original data set and newly created variable which is Mean_byStatus, Mean_byZip and ave.Travel. Predicting response time based on all of these 9 variables makes our best model and lowest MSE.

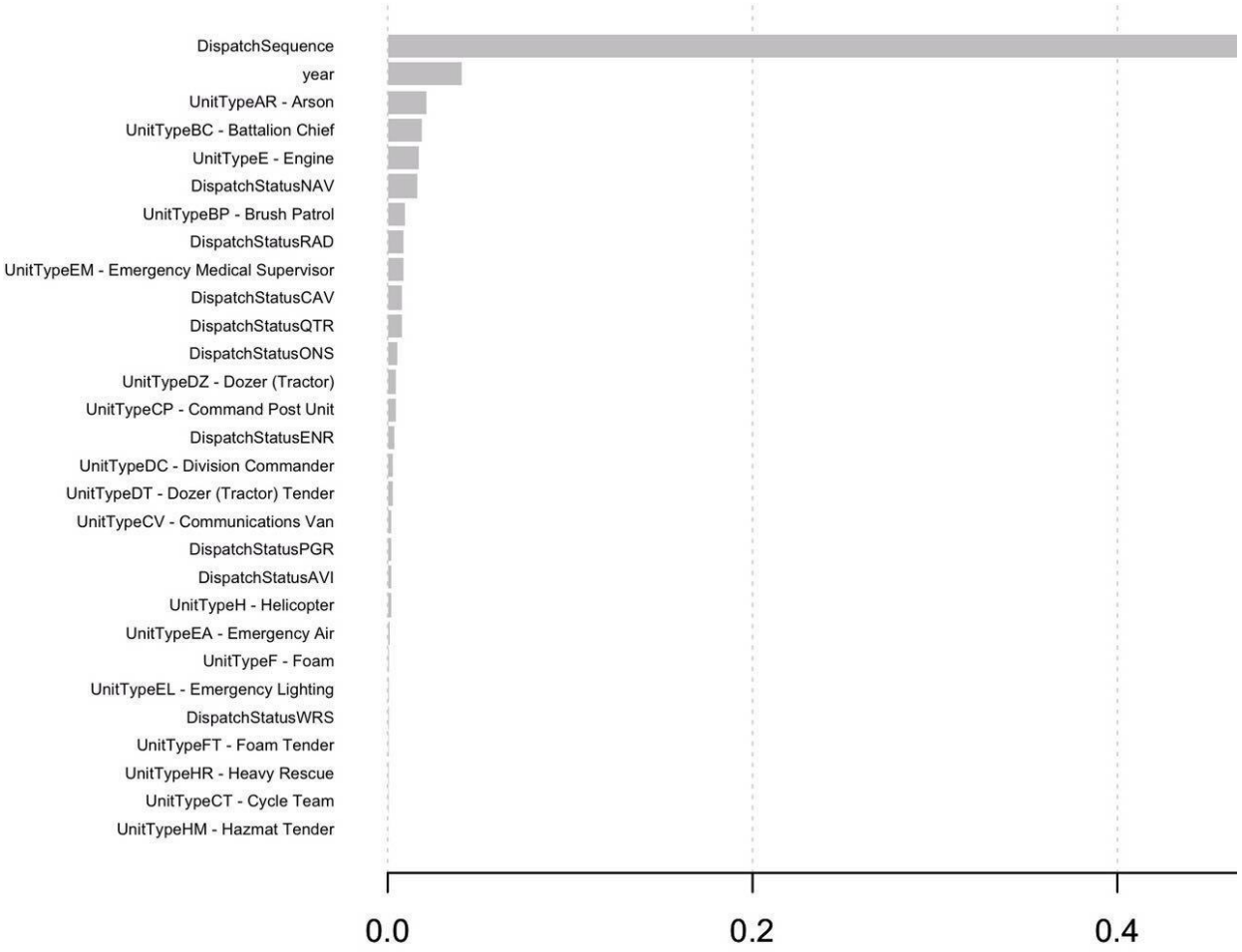
-**best MSE**: 1390489.67407

Parameter Tuning
-eta=0.15
-gamma=0.06
-max_dept=6
-min_child_weight=1
-nthread=4
-nrounds = 50

why it works well or why it failed to work well

The lafdtraining data contains 2774370 observations of 11 variables. Once we added all newly created variables, our training data would have 44 variables which is really big for running model. Xgboost provides parallelization of tree construction and distributed Computing for training very large models and very large dataset. It is really fast and works well on dominating structured datasets on classification and regression predictive modeling. Besides, The data set we use has many missing values. Gradient Xgboost algorithm can automatically handle missing data values and further boosted an already fitted model on new data.

Summary of Findings (Result with plot, Aalysis)



From the importance plot, we can see that the plot shows that Dispatch Sequence has the highest importance, and DispatchStatusHSP has the lowest importance. Thus, for predicting response time by using Xgboost, Dispatch Sequence is the most significant variable.