Predictive Analysis of Crash Incidents in Hillsborough County

Bo Fethe

Group 123

March 2024

Hello everyone. My name is Bo and I am a Solutions Engineer at Hillsborough County. Today, I am going to share a project I worked on analyzing and predicting crash data in Hillsborough County.

Introduction

- Unintentional accidents were the #3 in leading cause of death in Florida during 2022 (Florida Department of Heath[1])
- International programs, like Vision Zero, help communities eliminate traffic fatalities and serious injuries by improving safety measures.
- Signal Four Analytics receives tabular and geospatial crash data from Florida Department of Highway Safety and Motor Vehicles (FLHSMV)
- **Q: How well can machine learning predict incapacitating events?**

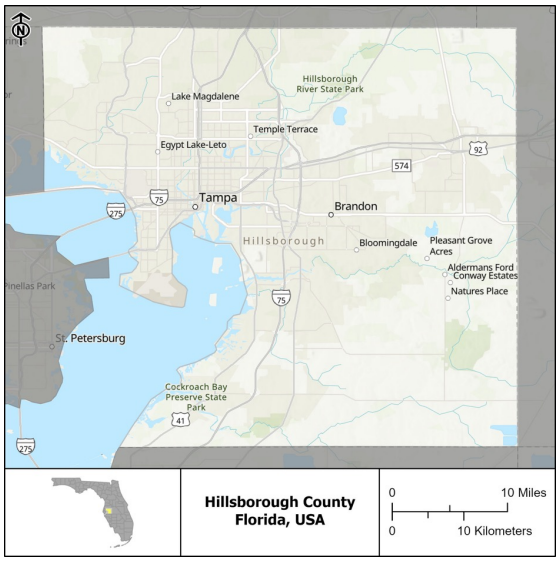
[1] https://www.flhealthcharts.gov/ChartsReports/rdPage.aspx?rdReport=ChartsProfiles.LeadingCausesOfDeathProfile

In 2022, the Florida Department of Heath showed Unintentional accidents were the state's 3rd highest cause of death. To improve public safety, international programs, such as Vision Zero, exist to help communities eliminate life-threatening events by implementing new roadway safety designs. The University of Florida's GeoPlan Center hosts and maintains a geospatial crash database called Signal Four Analytics which takes records from the Department of Highway Safety and Motor Vehicles and reporting officers. This raised the question of how well does machine learning predict these incapacitating events.

Study Area

- Hillsborough County, Florida, USA
- 3 incorporated cities
  - Tampa
  - Plant City
  - Temple Terrace
- Population
  - July 2023: **1,535,564** (U.S. Census Bureau[2])
  - 2050 Estimate: **2,017,294** (Plan Hillsborough[3])

[2] https://www.census.gov/quickfacts/fact/table/hillsboroughcountyflorida/PST045223
[3] https://planhillsborough.org/county-to-reach-2-million-residents-and-1-4-million-jobs-by-2050



The study area is Hillsborough County located on the coast of southwest Florida. The county has 3 incorporated cities: Tampa, Plant City, and Temple Terrace. In July of 2023, the population was 1.5M with an estimated growth to over 2M by 2050. As the regional density increases, more cars will be on the road which can lead to more accidents.

Data

- 2023 crash records from Signal Four Analytics
- 23/117 Signal Four columns + 2 generated columns selected for analysis = **25 columns**
- Crash Severity simplified into 2 classifications
  - Fatality or Serious Injury = **Incapacitating**
  - No Injury or Injury = **Non-incapacitating**
- Peak traffic hours
  - AM Peak (M-F; 07:00-10:00)
  - Mid Peak (M-F; 10:00-16:00)
  - PM Peak (M-F; 16:00-19:00)
  - Weekend Peak (Sa-Su; 06:00-20:00)
  - Off Peak for all other times

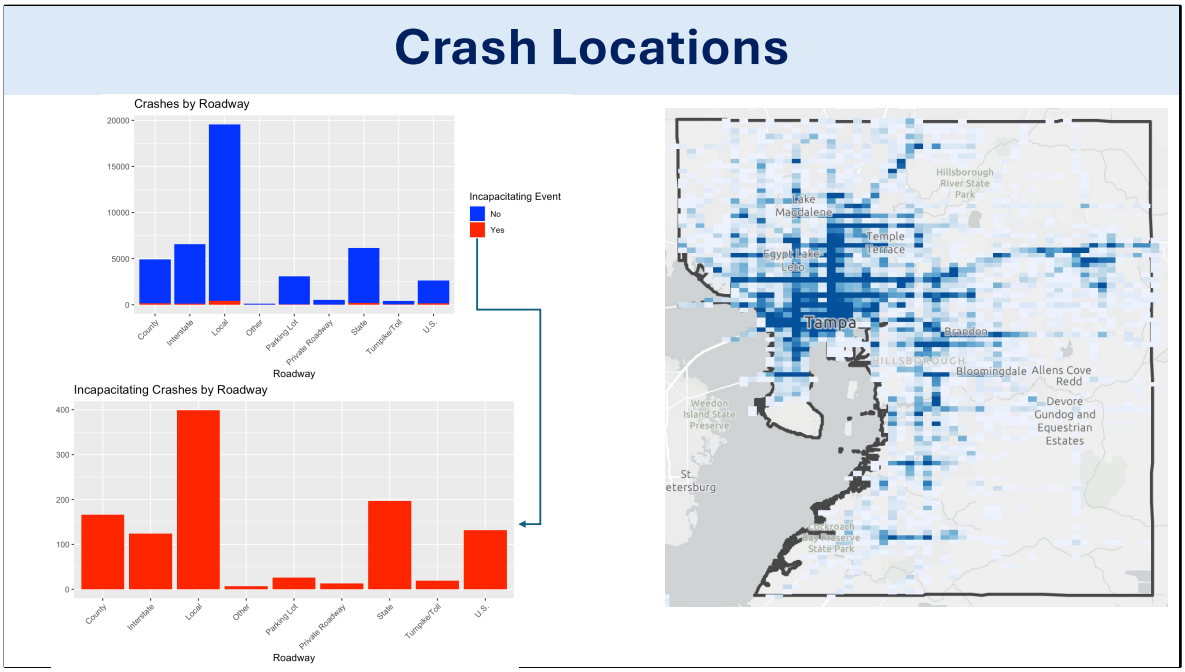
For the analysis, all 2023 crash events were exported. By default, Signal Four has 117 predictors consisting of a mix of categories, quantities, dates, spatial, and unstructured values. To simplify the the data for analysis, 23 columns from Signal Four were selected with the addition of 2 generated columns for a total of 25 attributes. Crash Severity was classified into a binary repones with fatal or serious injuries represented as Incapacitating and no injury and minor injuries as Non-Incapacitating. The date and time of the events were transformed into 5 peak traffic period bins for AM, Mid, PM, Weekend, and Off peak.

Attributes

Name	Description
INCAPACITATING_FLAG	Binary response whether the crash severity resulted in a serious injury/fatality or not derived from S4_CRASH_TYPE_SIMPLIFIED
PEAK	Peak traffic period derived from CRASH_DATE_AND_TIME
TOTAL_NUMBER_OF_VEHICLES	Count of all vehicles involved in the crash
TOTAL_NUMBER_OF_PERSONS	Count of all persons involved in the crash (drivers, passengers, and non-motorists)
RURAL_OR_URBAN	Check if the traffic crash occurred inside the corporate limits of the city
ROAD_SYSTEM_IDENTIFER	This classification is used to identify the primary road system on which the traffic crash occurred
ROAD_SURFACE_CONDITION	This classification is used to identify the surface condition of the street, road or highway at the time of the traffic crash
S4_CRASH_TYPE_SIMPLIFIED	Crash type simplified for practitioners who desire less detailed crash types
S4_DAY_OR_NIGHT	To identify if the crash happened in the daytime or nighttime
S4_IS_AGGRESSIVE_DRIVING	To identify the presence of aggressive driving in the crash
S4_IS_ALCOHOL_RELATED	To identify the presence of alcohol by the driver in the crash
S4_IS_CMV_INVOLVED	To identify the presence of a commercial motor vehicle is involved in the crash
S4_IS_DISTRACTED	To identify the presence of driving distraction in the crash
S4_IS_DRUG_RELATED	To identify if the crash is drug related due to refusal of drug test or positive drug test of driver
S4_IS_HIT_AND_RUN	To identify if the crash is hit and run related
S4_IS_INTERSECTION_RELATED	To identify if the crash is intersection related
S4_IS_LANE_DEPARTURE_RELATED	To identify the presence of lane departure in the crash
S4_IS_SPEEDING_RELATED	To identify if the crash is speed related
S4_TRAILER_COUNT	Number of trailers in the crash
S4_MOTORCYCLE_COUNT	Number of motorcycles in the crash
S4_MOPED_COUNT	Number of mopeds in the crash
S4_BICYCLIST_COUNT	Number of bicyclists in the crash
S4_AGING_DRIVER_COUNT	Number of aging drivers in the crash who are 65 or older
S4_TEENAGER_DRIVER_COUNT	The number of drivers involved in the crash whose age at time of crash is between 15 and 19
S4_UNRESTRAINED_COUNT	Number of motor vehicle occupants not using restraint system(s) at time of crash

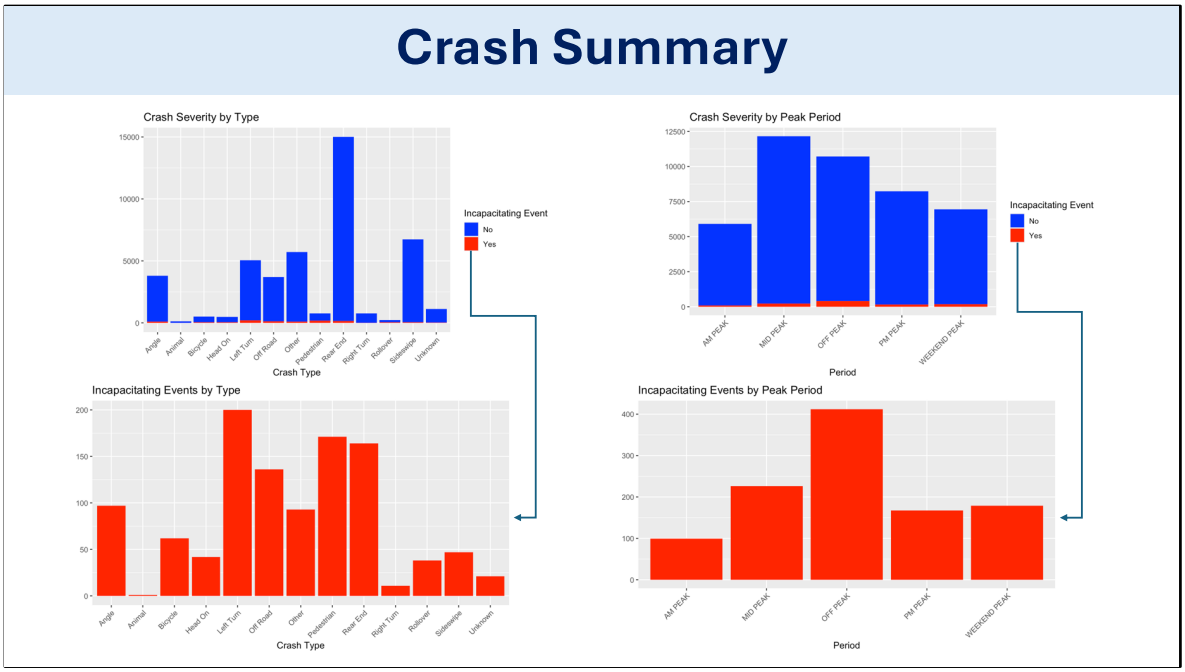
Above are the columns used for this analysis along with their descriptions taken from the data dictionary. The top row, Incapacitating flag, represents the response variable used for training and prediction.

Crash Locations



The crash locations in the raw data are shown on this slide. The bin aggregate map, shows that the majority of the crashes occur within city limits with linear tangents along the highways heading east towards Orlando and north towards Gainesville. A significant portion of all crashes occurred on local roadways with almost 3x more events than the next leading roadway system. This trend stays relatively true when considering only incapacitating events with the proportion dropping closer to 2x more than the next leading system. What this means is traffic safety improvements are especially important within the City.

Crash Summary



For crash types for all events, accidents involving rear ends were the most common, but a slightly different trend shows when considering only incapacitating events. Taking a left turn over on-coming traffic was the leading crash type, and pedestrians being on the roadside, rear ends, and drivers running off the road were the other leading causes of incapacitating events. As far as peak traffic hours go, the mid-day rush between 10a-4p has the most events and the off peak between M-F 7p-7a and Sa-Su 8p-6a has the most incapacitating events, which could be related to lack of visibility or alcohol.

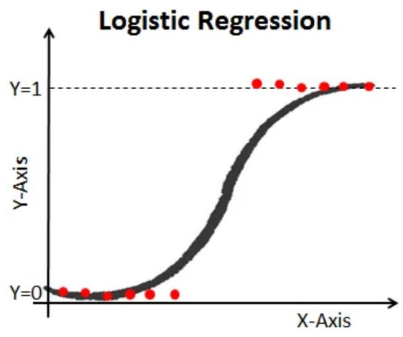
Methodology

- Split data into training and testing subsets
- Handle unbalanced data
  - 1,083 Incapacitating (*2.5%)* and 42,877 Non-incapacitating (*97.5%)*
  - Randomly Over Sampling Examples (ROSE) balances data by synthetically sampling training data
  - ROSE(INCAPACITATING_FLAG~., crashtrain, N=2000)
- 3 models used and compared testing error ($\frac{TP + TN}{TP + TN + FP + FN}$) and specificity ($\frac{TN}{TN + FP}$)

To get started, a subset of the data was reserved for testing while the majority was used for training. The severity of crash data is unbalanced with about 2.5% classified as incapacitating. While this adds a challenge when training a model to successfully make predictions, this is a good problem to have. When I am on the road, I want the odds of my safety strongly skewed in my favor of not experiencing a life-threatening event. Techniques to account for this include under-sampling the majority class, over-sampling the minority class, or synthetically sampling both classes using either ROSE or SMOTE. The code shown here is an example using ROSE to ensure the size of each class is 2,000. 3 different models were trained and the testing error and specificity were compared to measure performance.

Model - Logit

- Logistic Regression
  - A probabilistic classification model used for binary classification
  - Suitable for scenarios when you need to understand probability of the response variable
  - Stepwise variable section via Akaike information criterion (AIC)
  - step(glm(INCAPACITATING_FLAG~ ., data = crashtrain_balanced), trace = F)
  - Selected 16 predictors

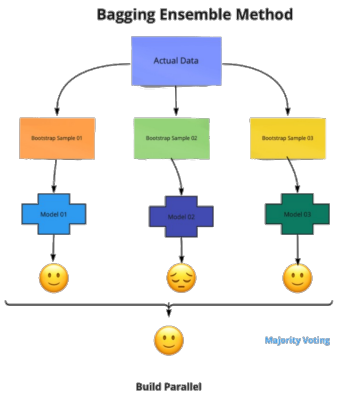


TOAL_NUMBER_OF_VEHICLES	S4_CRASH_TYPE_SIMPLIFIED	S4_IS_DRUG_RELATED	S4_MOTORCYCLE_COUNT
TOTAL_NUMBER_OF_PERSONS	S4_IS_AGGRESSIVE_DRIVING	S4_IS_HIT_AND_RUN	S4_MOPED_COUNT
RURAL_OR_URBAN	S4_IS_ALCOHOL_RELATED	S4_IS_INTERSECTION_RELATED	S4_AGING_DRIVER_COUNT
ROAD_SYSTEM_IDENTIFER	S4_IS_CMV_INVOLVED	S4_IS_LANE_DEPARTURE_RELATED	S4_UNRESTRAINED_COUNT

Logistic regression is a binary classification model, which is suitable when you want to understand the probability of an event occurring. Variable selection can simplify the model by only using variables that contribute to the model's performance by comparing the AIC values between each iteration. This can be done taking a forward approach and adding variables, a backward approach and removing variables, or a stepwise approach that is a combination of both forward and backwards. One method is the stepwise approach which both adds and removes variables from the model using the step() function in R. This results in 16 variables shown in the table on the bottom of this slide.

Models - RF

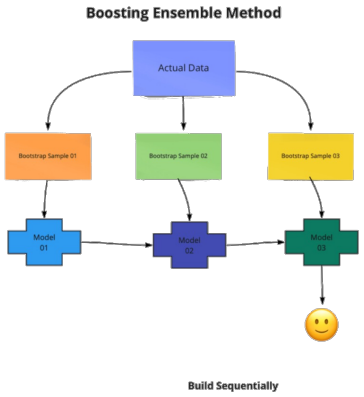
- Random Forest
  - An ensemble **B**ootstrap **agg**regat**ing** (bagging) model that combines many decision trees
  - Suitable for handling large datasets with high dimensionality and complexity
  - randomForest(as.factor(INCAPACITATING_FLAG)~., crashtrain_balanced, proximity=T, importance=T)



The next 2 models are called ensemble methods because they combine the results of multiple models into 1 result. Random forests are a bagging algorithm that bootstraps many independent decision trees and aggregates the results by taking the average or mode. This is suitable for handling large datasets with high dimensionality and can be done in R using the randomForest() function.

Models - GBM

- Generalized Boosting Regression
  - An ensemble gradient boosting model similar to random forest but iteratively trains weak learners to improve model performance.
  - Use cross-validation to find the optimal # of iterations
  - Suitable when you can tolerate longer training times
  - gbm(INCAPACITATING_FLAG~., data=crashtrain_balanced, distribution = 'bernoulli', n.trees = 5000, shrinkage = 0.01, interaction.depth = 1, cv.folds = 10)



Another ensemble algorithm is boosting, which is similar to random forest that iteratively builds decision trees, but instead of aggregating independent trees, it uses the previous results to train weak learners and improve each iteration. To find the optimal # of iterations for predicting, you can use cross-validation. This can improve accuracy and can be done in R using the gbm() function, but it also increases time spent in computation.

Results

Logit	N	Y
N	3480	38
Y	795	83
Test Error	0.190	
Specificity	0.686	

RF	N	Y
N	4164	91
Y	111	30
Test Error	0.046	
Specificity	0.240	

GBM	N	Y
N	4273	117
Y	2	4
Test Error	0.027	
Specificity	0.033	

Testing Error: Proportion of errors overall. **Low = good**
Specificity: Proportion of correctly IDed negative results. **High = good**

Now for the results. The confusion matrices for each model are shown here with the diagonal showing the amount of correctly identified events. For an overall approach, the boosting model had the lowest testing error, meaning when considering all values for all classes, it performed the best. But there are cases where incorrect negative IDs can be costly, such as life-threatening events. This is where specificity becomes important. While the boosting model had the lowest test error, it really struggled with predicting true incapacitating events and got 4 correct and 117 wrong. Logistic regression had the worst testing error but outperformed the other 2 algorithms when predicting true incapacitating events.

Conclusion

- **Q: How well can machine learning predict incapacitating events?**
  - Different interpretations
    - Best for overall accuracy: Generalized Boosting Regression
    - Best for predicting incapacitating events: Logistic Regression
- Dataset challenges
  - Unbalanced data
  - Missing/wrong data

Back to the question of how well machine learning can predict incapacitating events. Can it? In short - yes. But it's important to consider how you're measuring efficiency. Overall, boosting was the preferred method for predicting crashes, but struggled with life-threatening events. Logistic regression did better at predicting those life-threatening events, but overall didn't do perform the best. This could be due to the handling of unbalanced data, or the lack of stronger predictors. For example: speeding involvement was incorporated as a binary classification, but going 5 mph over is very different than going 50 mph over. Additionally, in situations with costly consequences such as interacting with law enforcement, information may be withheld or fabricated for self protection.

More Details

Scan or **click** below for more resources about this study

Explore the data using an interactive map on ArcGIS Online

View the code and data used on GitHub

Questions?

This concludes the presentation. For more details, scan or click the links shown on this slide to view the data in an interactive map in ArcGIS Online, or view the code used for this analysis in the project's GitHub repository. Each site also has a reference to the other, so as long as you view one, you can get to the other. Thank you, and please let me know if you have any questions.