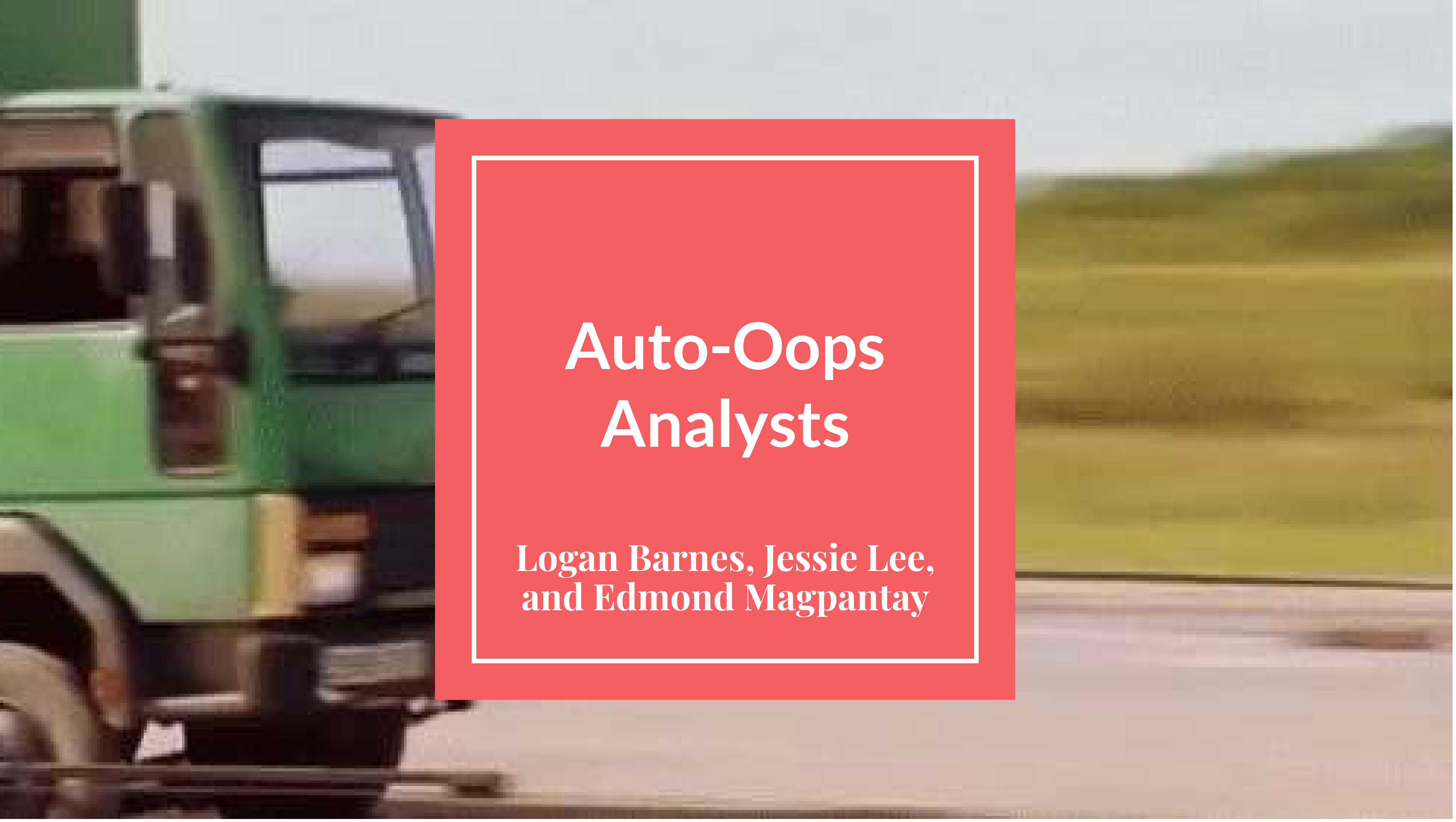


Auto-Oops Analysts

Logan Barnes, Jessie Lee,
and Edmond Magpantay

Theme

Part 1 (Project 3)

- Visualization on the Fatality Analysis Reporting System Data from the year 2021 by state based on:
    - Weather Conditions
    - Accident Types
    - Lighting Conditions

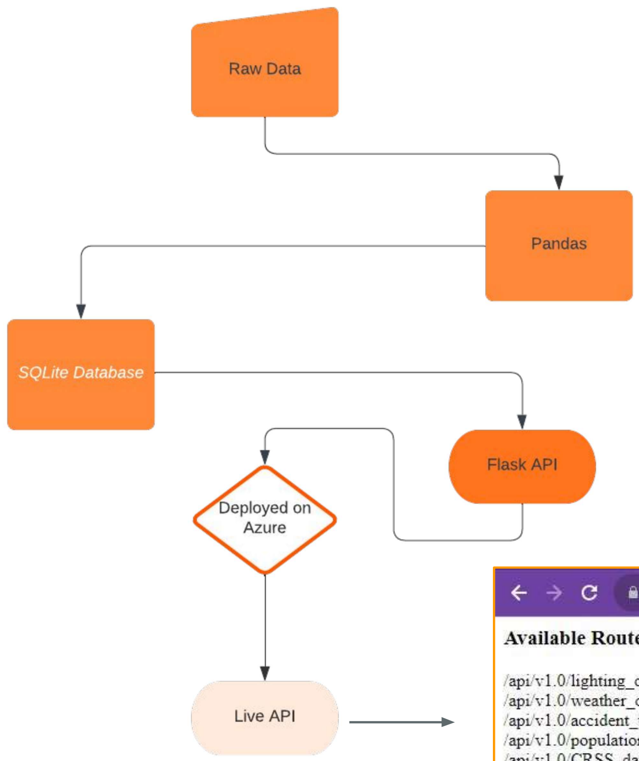
Part 2 (Project 4)

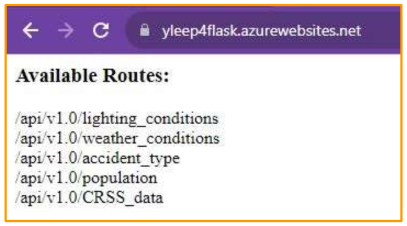
- Injury Prediction based on Crash Report Sampling System:
    - Interactive Machine Learning Module
    - Predicts injury severity based on user input

Coding Approach

- Jessie
  - Develop a Flask API for the new database.
  - Utilize machine learning to identify which features to retain in the database.
- Edmond
  - Create machine learning model using Tensorflow specifically for machine learning functionalities that will facilitate user interactions.
- Logan
  - Enhance the website with additional HTML components.
  - Integrate machine learning capabilities into the website.

Building the Database





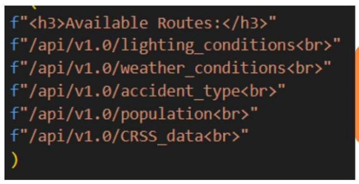
Available Routes:

/api/v1.0/lighting_conditions
/api/v1.0/weather_conditions
/api/v1.0/accident_type
/api/v1.0/population
/api/v1.0/CRSS_data

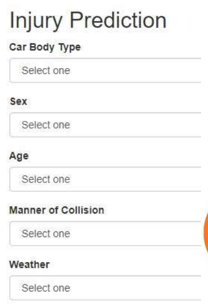
1. Features Check
   a. Load in the year 2021 data using Pandas, clean, scale, fit and train the data using Random Forest to identify the top 20 important features.
2. Processing the Data
   a. After discussing with Ed, we identified the features we will be needing for the Machine Learning.
   b. Write new table to existing database in DB Browser (SQLite).
   c. Read, clean and transform the csv data using Pandas and write it to the SQLite Database using SQLAlchemy.
3. FlaskAPI
   a. Update existing Flask API with new route pulling from the new table.
   b. Write requirements file for dependency installation.
   c. Deploy Flask API on to Azure.

Key Learnings:
1. Deploying via GitHub allows for quick and easy identification of any errors.
2. Requirement text is needed for Flask app deployment.
3. API routes must be https for Javascript to run correctly.

Learn Baby, Learn

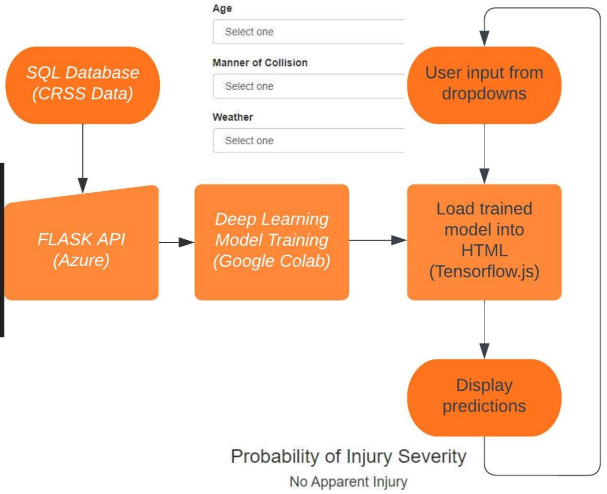


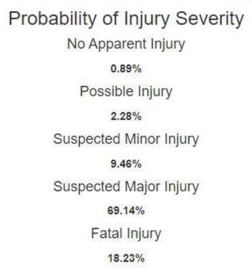
```
f"<h3>Available Routes:</h3>"
f"/api/v1.0/lighting_conditions<br>"
f"/api/v1.0/weather_conditions<br>"
f"/api/v1.0/accident_type<br>"
f"/api/v1.0/population<br>"
f"/api/v1.0/CRSS_data<br>"
)
```



Injury Prediction

Car Body Type	Select one
Sex	Select one
Age	Select one
Manner of Collision	Select one
Weather	Select one


SQL Database (CRSS Data)
FLASK API (Azure)
Deep Learning Model Training (Google Colab)
User input from dropdowns
Load trained model into HTML (Tensorflow.js)
Display predictions

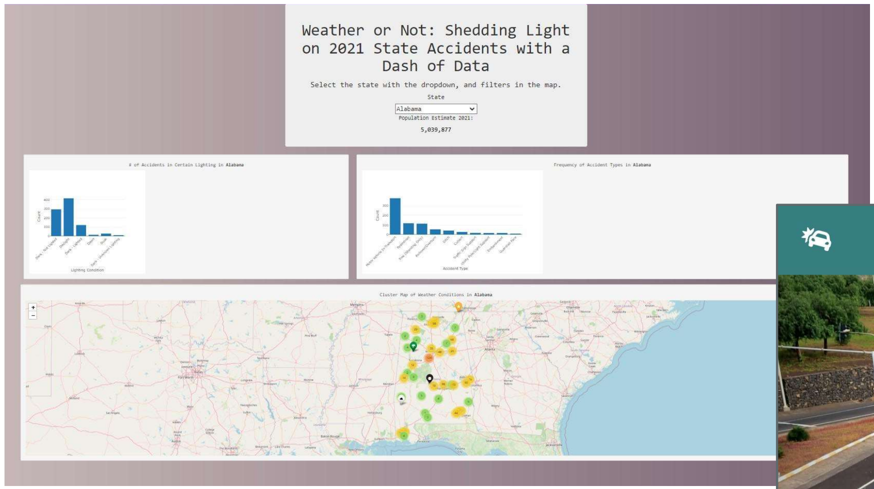


Probability of Injury Severity

```
No Apparent Injury
0.89%
Possible Injury
2.28%
Suspected Minor Injury
9.46%
Suspected Major Injury
69.14%
Fatal Injury
18.23%
```

1. Preprocess Data
    a. From Jessie’s Flask API on Azure, pulled data and cleaned up, like binning ages, creating ordinal data, dropping null rows, one hot encode because of multiclass.
2. Train the model
    a. Using Google Colab, used Keras tuner, luck and praying to create a viable model
        i. LeakyReLU, relu, elu
        ii. Softmax
        iii. So many epochs
3. Save and load the model to HTML
    a. Used tensorflow.js to load the model.json
    b. Pull the dropdown selected elements’ values to create an input.
    c. Model produces probabilities of the 5 classes, which are displayed on the page.
4. Notes
    a. Over and undersampling are necessary if data is unbalanced.
    b. For multiple classes, softmax, categorical_crossentropy, and one-hot encoding
    c. For training, make what you are able to ordinal data, easier for training.

Final Visualization



Weather or Not: Shedding Light on 2021 State Accidents with a Dash of Data

Select the state with the dropdown, and filters in the map.

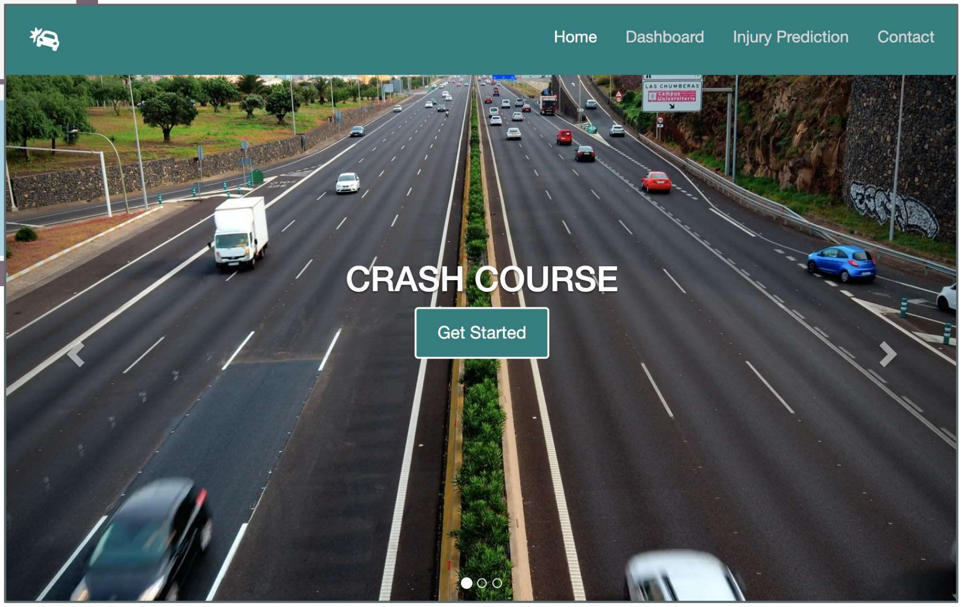
State: Alabama

Population Estimate 2021: 5,039,877

# of Accidents in Certain Lighting in Alabama
Frequency of Accident Types in Alabama
Cluster Map of Weather Conditions in Alabama

Old...

New!



Learnings

- The CRSS data (Crash Reporting Sampling System) did not have geographical data available.
- Since CRSS data is a sample, it may not be truly reflective of all the reported traffic accidents.
- We only have data on accidents that occured and have no insight on how that exists in proportion to actual traffic on the road.
- Train using more columns and rows to get a better trained model (current accuracy is around 65%).

Thank you!

Let us know if you have any questions! ☺️