

Geographic Analysis of Traffic
Accident Severity in Atlanta

Data Science and Analytics

Athens, GA
2022-2023
PID: 107301

Table of Contents

Introduction	3
Data Dictionary	4
Purpose	6
Methods	7
Results	8
Conclusion	20
Next Steps	21
References	22

Introduction

Every year in major cities across the United States, traffic safety becomes one of the most integral concerns for funding and new regulation. In the United States, about \$11.6 billion has been spent by the Fixing America's Surface Transportation Act (FAST Act) from 2016 to 2020. (City of Atlanta, 2018). Atlanta, one of the country's largest cities, is currently outgrowing its population center and has seen increased traffic accidents. This analysis seeks to utilize geographical and environmental factors as a basis to train a machine learning model measuring the severity of various car crashes. With the data retrieved from the city of Atlanta, the analysis iterates through the steps of data cleaning and exploratory data analysis and uses stochastic gradient descent to rate each accident's severity from a scale of 1 to 4. Atlanta traffic officials and first responders can use this data to make preliminary predictions on any accidents in the city and citizens can use this to travel safely and avoid areas of high accident severity or travel cautiously in certain road and weather conditions.

Data Dictionary

Column/Feature Title	Data Type	Description
Severity	int64	Rating of accident severity on a scale from 1 to 4
Start_Time	datetime64	Date and exact time accident occurred
End_Time	datetime64	Date and exact time accident 6 hours after the accident occurred
Start_Lat	float64	Geographical latitude accident occurred at
Start_Lng	float64	Geographical longitude accident occurred at
End_Lat	float64	Geographical latitude of vehicles after accident
End_Lng	float64	Geographical longitude of vehicles after accident
Distance(mi)	float64	Distance affected by accident-related vehicles during collision
Weather_Timestamp	datetime64	Time at which weather is measured
Temperature(F)	float64	Air temperature at the time of accident
Wind_Chill(F)	float64	Temperature accounting for wind at time of accident
Humidity(%)	float64	Humidity percentage at time of accident
Pressure(in)	float64	Pressure measurement at time of accident
Visibility(mi)	float64	Visibility rating out of 10 at time of accident
Wind_Speed(mph)	float64	Speed of wind in mph at time of accident
Precipitation(in)	float64	Inches of water-based precipitation fallen during the accident
Amenity	int64	Whether accident occurred near a public amenity or public grounds space or not
Crossing	int64	Whether accident occurred at a crosswalk or not
Give_Way	int64	Whether accident occurred at a yield stop or not

Junction	int64	Whether accident occurred at junction or not
No_Exit	int64	Whether accident occurred at a no exit space or not
Railway	int64	Whether accident occurred at railway or not
Station	int64	Whether accident occurred at a station or not
Stop	int64	Whether accident occurred at a stop sign or not
Traffic_Signal	int64	Whether accident occurred at a traffic signal light or not
Is_Calm	int64	Whether or not winds are calm
SideN	int64	Which side of the car accident primarily happened on, Left or Right
Time_of_Day	int64	Time of day at which accident occurred, Day or Night
Weather Variables: Derived from weather condition string categories		
Snowy	int64	Sub-variable for weather: if snowing
Tstorms	int64	Sub-variable for weather: if thunderstorms
Rain	int64	Sub-variable for weather: if raining
Cloudy	int64	Sub-variable for weather: if cloudy
Windy	int64	Sub-variable for weather: if windy
Clear	int64	Sub-variable for weather: if clear









Purpose

The city of Atlanta has a higher death rate for crashes than most other cities of its size and the national average, with 72% of fatalities and 42% of injuries occurring on only 6% of the roadways (Department of City Planning 2018). The city has also seen an increase from 1685 to 2581 annual crashes involving pedestrians and cyclists from the years 2006 to 2015 (Atlanta Regional Commission 2019). In the year 2020 alone, 1664 people died in car accidents, ranking the state of Georgia as the state with the 3rd highest number of crash-related deaths (Insurance Institute of Highway Safety 2022). Atlanta traffic accidents and accidents in general form “hotspots” in areas above 35 mph, with four or more lanes, at crosswalks, and in areas of poor lighting. The City of Atlanta has thus developed a zero-fatality plan known as “Vision Zero” to reduce all fatal and deadly crashes to zero by 2030. In order to adapt to the ever-changing scenarios occurring in each and every accident, the city must identify the most severe crashes and crash sites. By identifying key geographical aspects of severe crashes (intersections, traffic signals) in the Atlanta area and corroborating that data with environmental factors (weather, time of day), we seek to anticipate the importance and danger of certain streets.

Methods

1. [Kaggle](#): We utilized Kaggle's wide variety of open-source datasets to locate datasets regarding traffic congestion and accidents in the city of Atlanta. By sorting datasets on the number of usable quantitative factors, we compiled a dataset with information on severe accidents in Atlanta over the past 5 years that contained several valuable factors that could influence the severity of a given accident.
2. [Folium](#): Folium is a graphing module for Python that makes use of GeoJSON map data of the city of Atlanta to develop a heatmap visualizing the location of all the geotagged accidents in the dataset.
3. [Gradient Boosting Regressor](#): The Gradient Boosting Regressor ensemble is a Sci-Kit learn algorithm which consists of a group of individually weak regression trees connected together to build a strong learning algorithm that optimizes log-loss residual functions to build a classification model for the categorized output in this dataset with partly imputed data.
4. [Box Plots](#): Box Plots were used in for data engineering algorithms to impute values.
5. [Distribution Plots](#): Distribution plots displayed the spread of a categorized numerical variable and the count of each value of that variable.
6. [Joint Plots](#): Joint Plots not only use a Scatter Plot style dot format to plot individual data points and a secondary graph to show the general distributions of the X and Y variables in relation to each other.
7. [Histogram Plots](#): Histograms display the number of data entries with a variable in a certain range/bin and how that was affected by another variable.

Results

	Severity	ture(F)	Wind_Chill(F)	Humidity(%)	Pressure(in)	Visibility(mi)	Wind_Direction	Wind_Speed(mph)				
							Filter 11,855 recor...					
	1	4	98	-5	98	10	28	31	0	10		
	3	77		69	29.85	10	NE	8.1				
	2	48	46	66	28.97	10	WSW	5				
	2	53	53	86	29.23	10	CALM	0				
	2	53	53	86	29.23	10	CALM	0				
	2	68	68	100	28.93	5	VAR	6				
	2	38	38	76	29.24	10	CALM	0				
	2	43	43	68	29.59	10	CALM	0				
	2	68	68	100	28.93	5	VAR	6				
	2	68	68	100	28.93	5	VAR	6				
	2	67	67	90	29.04	9	E	6				
	3	55.9		100	30.06	0.5	East	8.1				
	2	48.9		97	30.27	4	East	8.1				
	2	64.9		56	30.09	10	NNW	13.8				

We began with [this public dataset](#) containing information on nearly twelve thousand traffic accidents in Atlanta and characteristics about the factors of the crash including temperature, location, weather conditions, road conditions, wind, and timing. The extraction of these factors is crucial to understanding the trend in accidents in Atlanta and mitigating the risk factors behind accidents, which can then be analyzed and used to predict the severity of a crash, a value on a scale from one to four.


```

df['Amenity'] = df['Amenity'].astype(int)
df['Crossing'] = df['Crossing'].astype(int)
df['Give_Way'] = df['Give_Way'].astype(int)
df['Junction'] = df['Junction'].astype(int)
df['No_Exit'] = df['No_Exit'].astype(int)
df['Railway'] = df['Railway'].astype(int)
df['Station'] = df['Station'].astype(int)
df['Stop'] = df['Stop'].astype(int)
df['Traffic_Signal'] = df['Traffic_Signal'].astype(int)
df['Start_Time'] = pd.to_datetime(df['Start_Time'])
df['End_Time'] = pd.to_datetime(df['End_Time'])
df['Weather_Timestamp'] = pd.to_datetime(df['Weather_Timestamp'])
df

print(df.dtypes)

```

Data Engineering Algorithm #1: Quantizing Boolean Variables

First, we need to clean up and quantize all of the data. Since this is a public dataset, most of the variables were boolean, string, or object data instead of numerical data that we could utilize in analysis. In order to convert the datatypes, we used the Pandas functions to convert booleans and timestamps to number values. However, for string-type data, the process was more complicated.

```

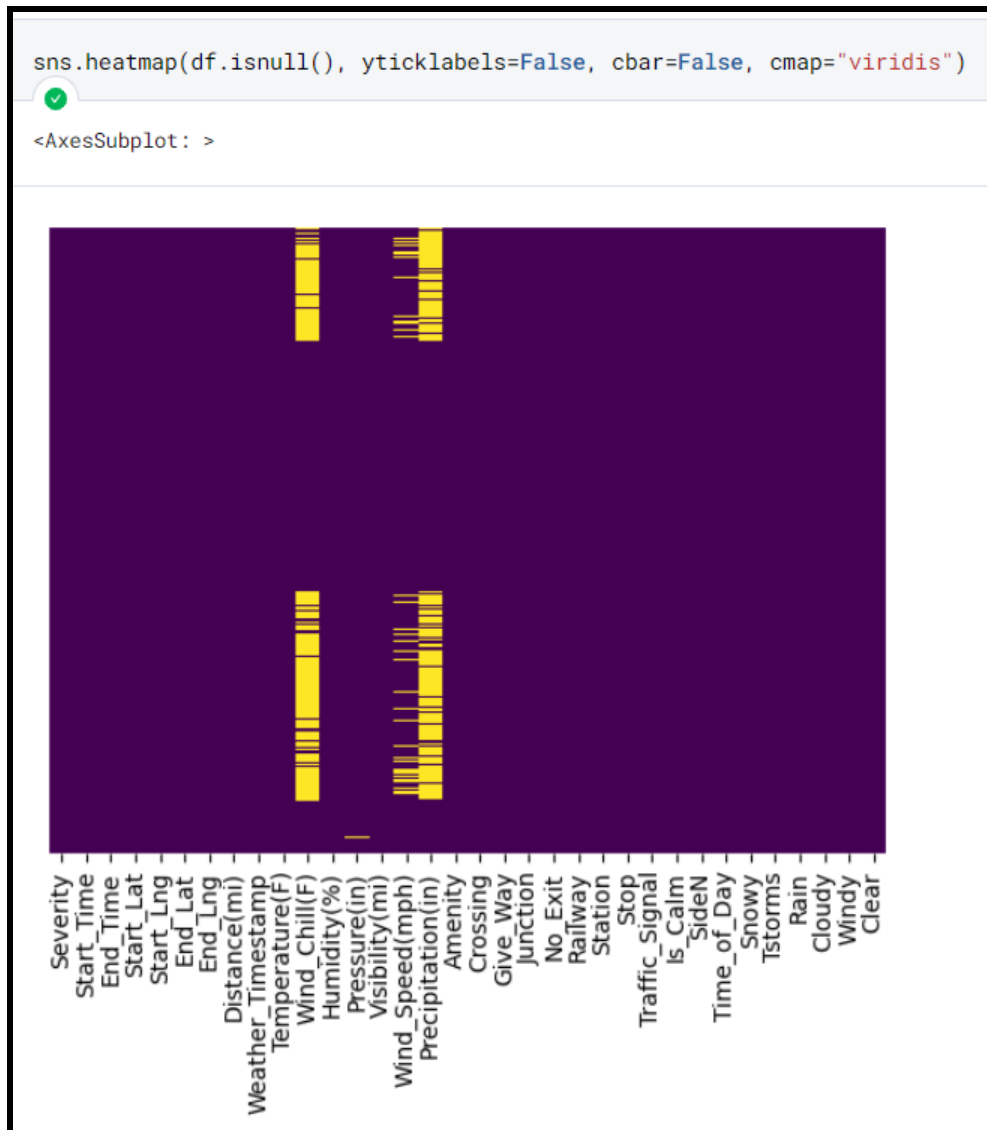
df['Snowy'] = df['Weather_Condition'].apply(lambda weather: 1 if weather == "Light Snow" or weather == "Wintry
df['Tstorms'] = df['Weather_Condition'].apply(lambda weather: 1 if weather == "Light Thunderstorms and Rain" or
df['Rain'] = df['Weather_Condition'].apply(lambda weather: 1 if weather == "Light Rain" or weather == "Rain" or
df['Cloudy'] = df['Weather_Condition'].apply(lambda weather: 1 if weather == "Mostly Cloudy" or weather == "Ove
df['Windy'] = df['Weather_Condition'].apply(lambda weather: 1 if weather == "Rain / Windy" or weather == "Light
df['Clear'] = df['Weather_Condition'].apply(lambda weather: 1 if weather == "Clear" or weather == "Fair" else 0

```

Data Engineering Algorithm #1 Part 2: Lambda Functions

To transform string-type data into quantitative data, we applied several lambda functions that created new integer-type columns. Since we need all data in the form of

zeros and ones for optimal training, specific weather conditions must be transcribed into different data values as there is no obvious numerical correlation between them. Each of the weather conditions was categorized into six main categories: snow, thunderstorms, rain, cloudy, windy, and clear. Based on the described weather condition for that data entry, ones and zeros were assigned for whether that condition was present or absent. This data engineering algorithm enabled the weather conditions to have a more significant impact than they would if they were considered to be thirty-nine individual types of weather conditions.

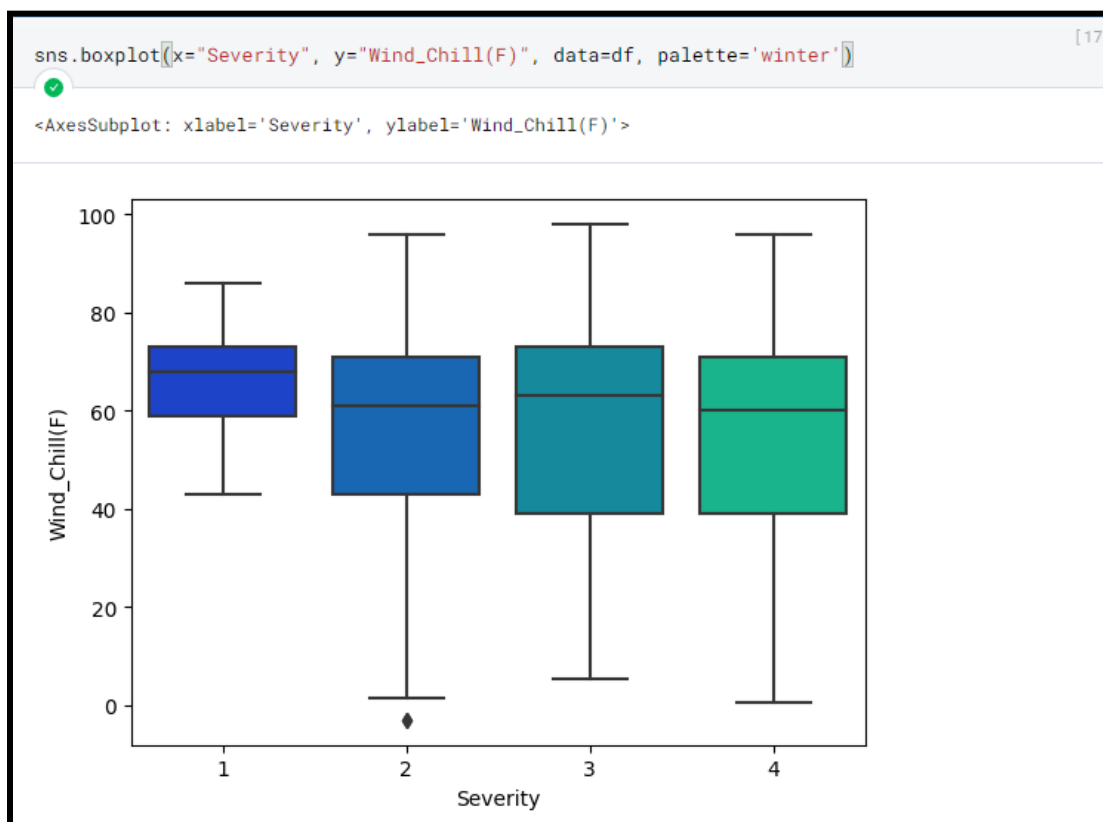


In order to process and analyze this data, we had to clean up the data with further processing with our second data engineering algorithm. As typical with public datasets, there is a lot of missing or NaN data, which required us to filter out this data in certain columns. This graph is a heatmap showing where the most missing data in the dataset is located, with the most being in the "Wind_Chill" and "Precipitation" categories.

```
# impute wind chill function
def impute_wc(cols):
    windchill = cols[0]
    severity = cols[1]

    if pd.isnull(windchill):
        if severity == 1:
            return 68
        elif severity == 2:
            return 61
        elif severity == 3:
            return 63
        else:
            return 60
    else:
        return windchill

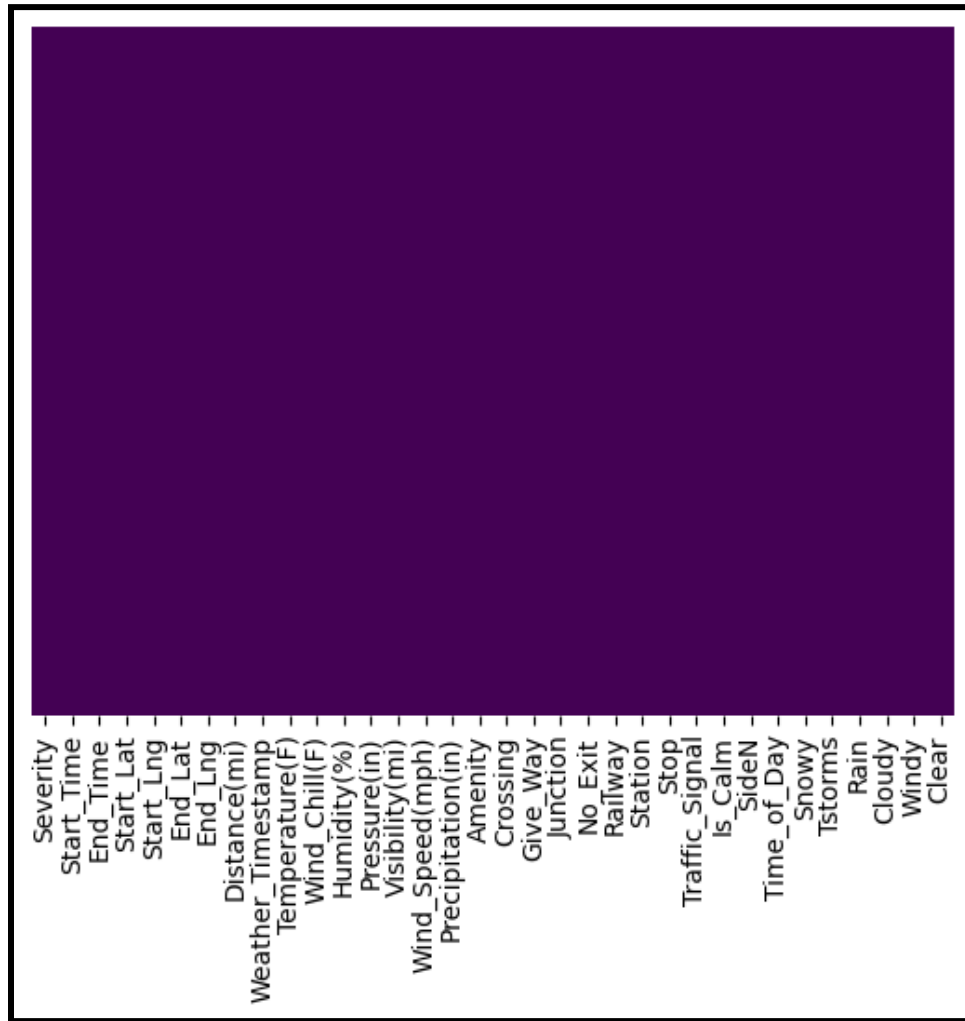
# applying the impute function
df['Wind_Chill(F)'] = df[['Wind_Chill(F)', 'Severity']].apply(impute_wc, axis=1)
```



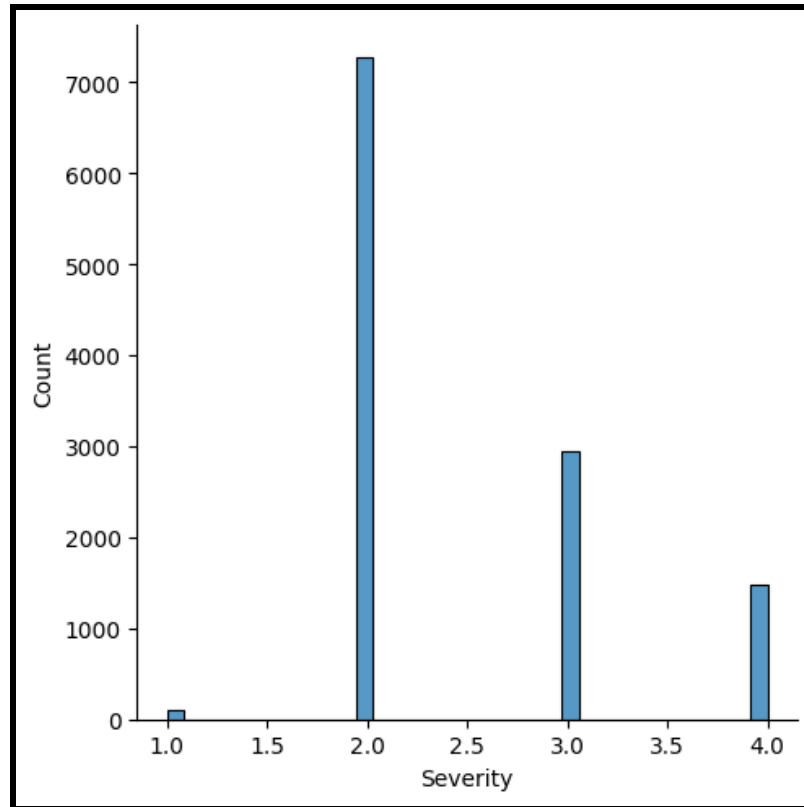
Data Engineering Algorithm #2: Imputing Missing Values

This impute function takes the median value from the boxplot comparing severity and another variable and then iterates through every row of processable data, looking for

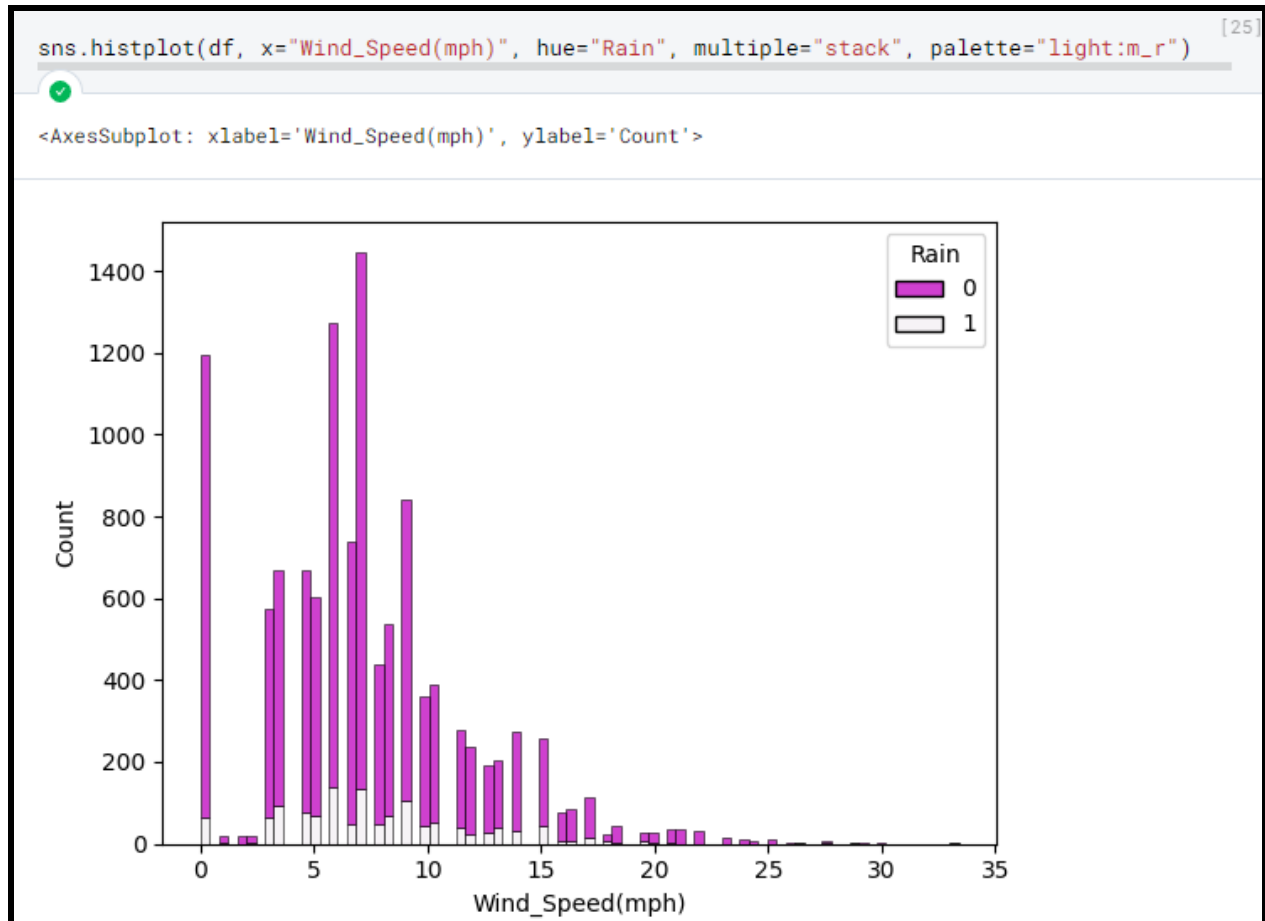
“NaN” or null values in the given data column. Based on the severity of the data entry, the median value of the variable for that given severity is imputed for the missing value to remove all the missing data on each of these variables.



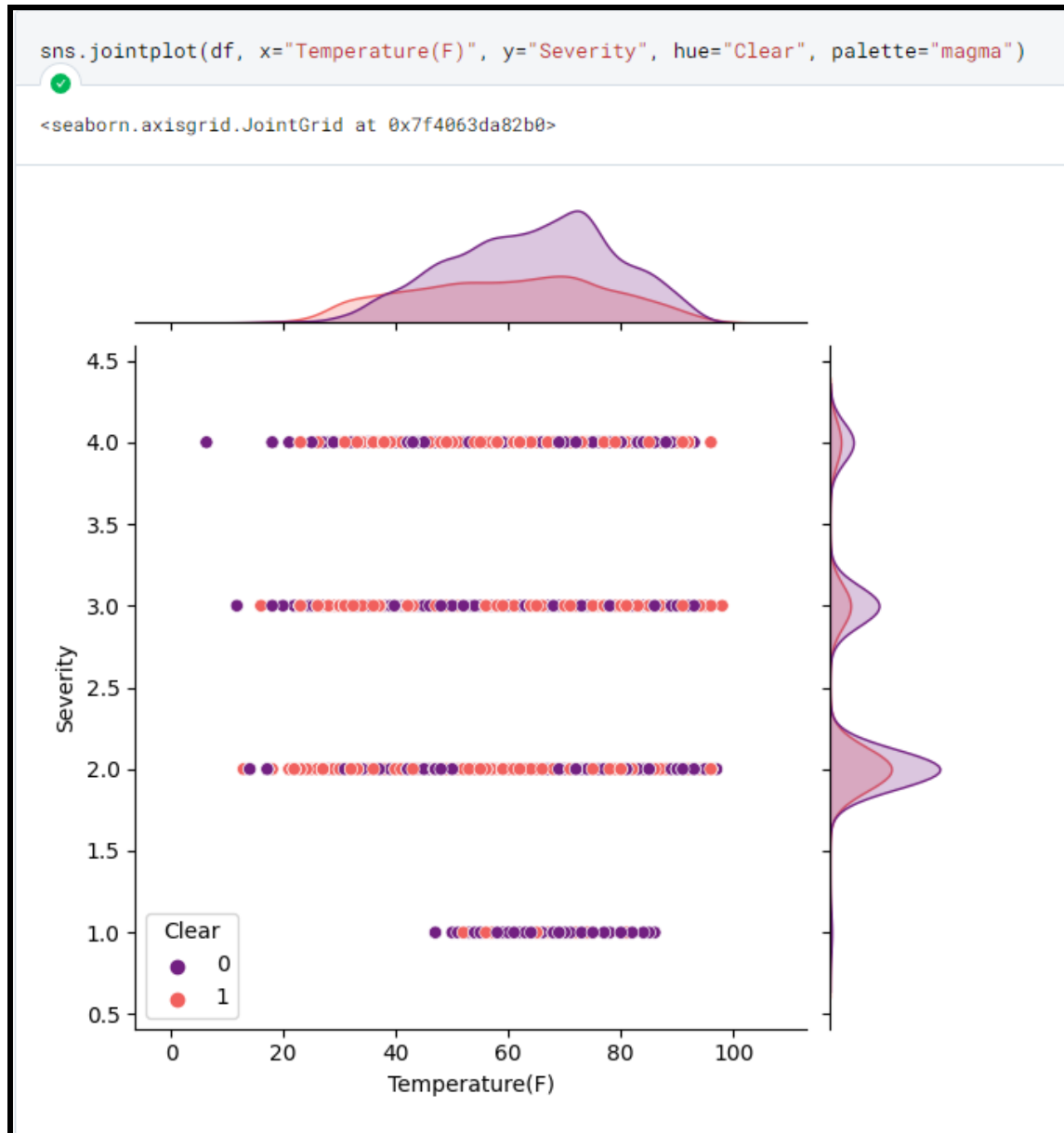
Wrapping up the data processing and engineering, we checked to ensure that there was no more missing data in the new dataframe and that we had all the necessary features extracted and ready for exploratory data analysis.



Starting off with exploratory data analysis, we made a distribution plot of the Severity across all the datasets to illustrate the count of each of the four ratings of severity. From this plot, we can note that the most common accidents are on a severity level 2. This distribution plot also showcases the vast amount of data collected in this dataset regarding accidents as there are over 11500 rows of available data for analysis, which improves our desired statistic as the data is more likely to be representative of all the accidents in the metro Atlanta area.

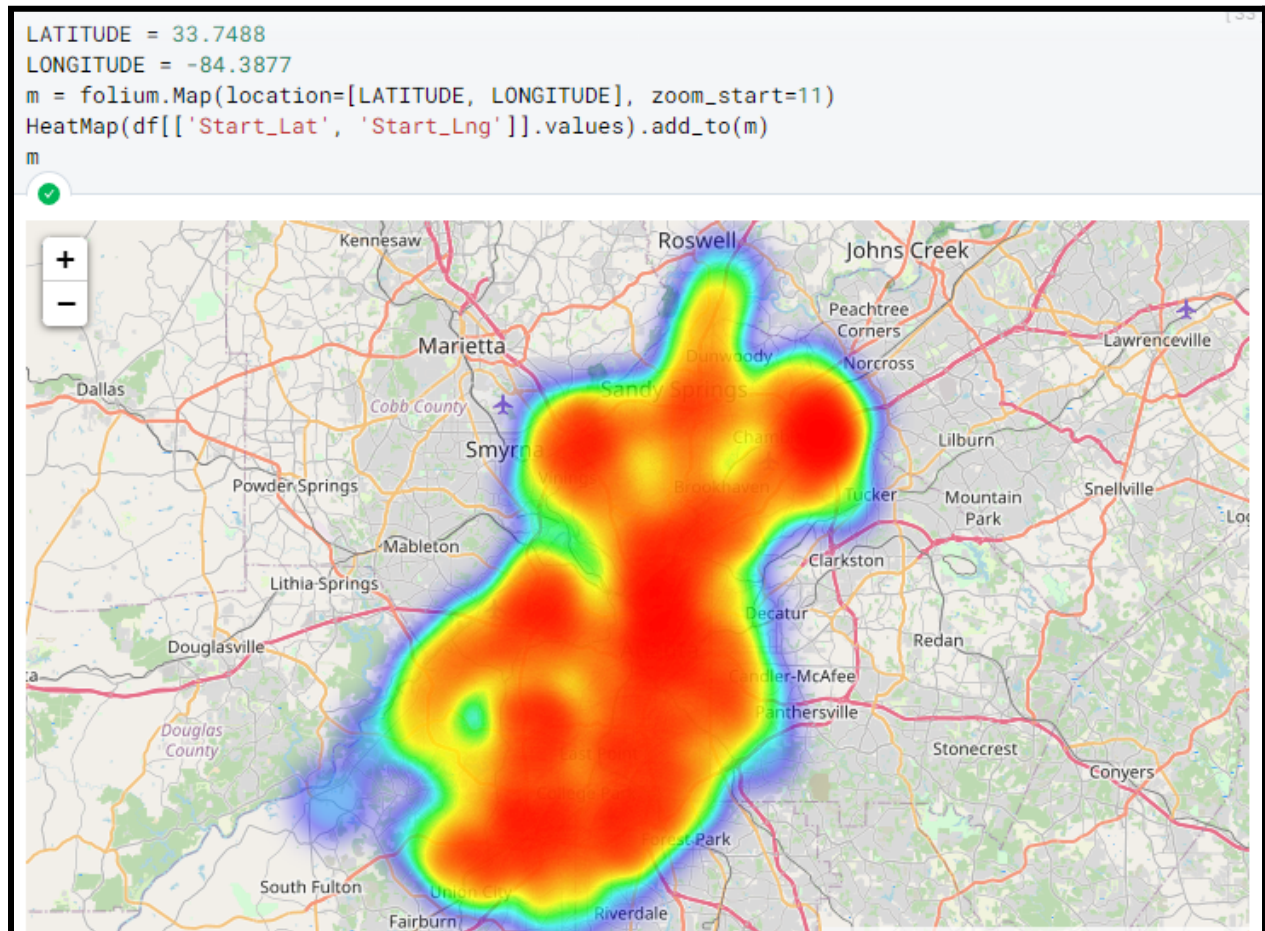


Next, we created a histogram of the number of accidents that occurred in that given range of wind speeds and added a hue of whether it was raining or not. The graph shows a skew to the right, meaning a wider distribution of accidents occurring at higher wind speeds. It has a peak of around 7.5 mph, which demonstrates that there are a lot of accidents that happen in the obstruction of the wind. Additionally, the rain hue shows us that a lot of accidents in the rain, which are shown with the white, occur at lower wind speeds, showing an inverse relationship between the rain and wind speed on the number of accidents that happen.



Following the histogram, we wanted to examine direct relationships between variables and the severity. We made a series of joint plots including this one, which shows the correlation between temperature and clear weather. This graph demonstrates that the clear weather didn't really have an effect on whether or not the crash was extremely severe or not, however, the temperature did as the more extreme temperatures have a higher concentration of severe crashes and the more middle-road temperatures between

60 and 70 degrees had less severe crashes, especially for the severity level 1 crashes. There were almost no severity level 1 crashes at extreme temperature levels.



After exploratory data analysis, we designed this Folium heatmap that represents the density of severe accidents that occurred across the metropolitan Atlanta area. As the color gets warmer on a spectrum from blue to red, the number of accidents and the severity of those accidents increases. From this map, we can deduce that the majority of accidents in Atlanta occur along major highways and major roads, specifically along I-75, I-85, and I-20. The bulk of the accidents in the red area occurs on the roads around the airport and as well around the Midtown area near the Georgia Institute of Technology.

```

import sklearn as skl
from sklearn.ensemble import GradientBoostingRegressor

features = list(df.columns)
features.remove('Severity')
target = ['Severity']

xdf = df.dropna()

X = xdf[features]

def convert_timestamp_column(df, timestamp_column):
    df[timestamp_column + '_epoch'] = df[timestamp_column].apply(lambda x: x.timestamp() if not pd.isnull(x) else 0)
    df[timestamp_column + '_epoch_milliseconds'] = df[timestamp_column + '_epoch'] * 1000
    df[timestamp_column + '_week'] = df[timestamp_column].dt.week
    df[timestamp_column + '_month'] = df[timestamp_column].dt.month
    df[timestamp_column + '_year'] = df[timestamp_column].dt.year
    df.drop(columns=[timestamp_column], inplace=True)

for ts_col in [x for x in X.columns if X[x].dtype == '<M8[ns]']:
    convert_timestamp_column(X, ts_col)

cats = [x for x in X.columns if X[x].dtype == 'O']
X[cats] = X[cats].astype('category').apply(lambda x: x.cat.codes)
Y = xdf[target]

model = GradientBoostingRegressor()
model.fit(X, Y)

acc = model.score(X, Y)
print(acc)

plt.bar(X.columns[np.argsort(model.feature_importances_)], model.feature_importances_)
plt.xticks(rotation=90)

```

/tmp/ipykernel_88/747729917.py:15: FutureWarning: Series.dt.weekofyear and Series.dt.week have been deprecated. Please use Series.dt.isocalendar().week
 df[timestamp_column + '_week'] = df[timestamp_column].dt.week
 /shared-libs/python3.9/py/lib/python3.9/site-packages/sklearn/ensemble/_gb.py:570: DataConversionWarning: A column-vector y was passed when you used fit: a 1D array was expected.
 y = column_or_1d(y, warn=True)
 0.87594708806543671

For our actual analysis, we chose a Gradient Boosting Regressor algorithm as it would allow us to see the specific impact of certain features on a gradient scale and it is also a regressor classification model that we can use to classify the output into the severity scale from 1 to 4. The gradient boosting regression model determined the importance of each feature as well as split the data into training and testing and tested the model on itself. This gradient regressor rendered an accuracy of 87.5%, which is above the threshold of typical regression models. If an Atlanta infrastructure professional was to

Conclusion

The Gradient Boosting Regressor illustrated where the Atlanta Department of Transportation should focus its efforts on mitigating traffic in certain types of locations and in certain conditions. In order to achieve the goals the city of Atlanta set, first responders and traffic officials, need to identify the most dangerous places in existing and future traffic ways. Anyone driving in the metropolitan Atlanta area can now locate where accidents are most prone to occur, which are major highway roads and crosswalk intersections, and also the city of Atlanta can modify its FAST Act plan for traffic, avoiding unnecessary spending and wasting resources on unfocused efforts, and accidents in the city by understanding the causes and impacts of said causes on crashes. The 88% accuracy on the model further demonstrates the capabilities of this data analysis and the crucial applications of the regression model. By being able to predict whether a crash is going to be fatal or not, which is denoted by a higher severity level, the safety of the city would be stronger and people would be less likely to die in crashes as now there is a better understanding of which specific factors would affect it the most. Our analysis can be most beneficial to understanding traffic accidents in Atlanta and to helping the city pursue its goals of decreasing the number of severe accidents to zero, thus achieving its "Vision Zero" plan for zero fatalities.

Next Steps

In future data analysis, we would like to look at a more holistic view of the infrastructure in the city and see how the traffic accidents and their severities would affect certain aspects of the infrastructure including budgets for certain departments, the way the government treats different accidents, and the way first responders are positioned and ready to go to any given location of a crash. We would like to apply the analysis from here about crashes to other risk factors in society and improve the infrastructure of the city of Atlanta. In order to achieve these next steps, we would need new features to examine the specifics of these geolocated accidents. Newly collected data would help with being able to see further about road conditions in specific places, distractions or other factors inside the car, and more metadata about the crashes including the types and the number of involved vehicles. Using this data, first responders can foster even better changes in the way they respond and react to crashes to promote the safety of the drivers on the congested roads of Atlanta, Georgia.

References

Atlanta Regional Commission. (2019, March). Safe Streets for Walking and Bicycling.

Atlanta Regional Commission.

<https://cdn.atlantaregional.org/wp-content/uploads/arc-safe-streets-webview-rev-mar19-1.pdf>.

City of Atlanta. (2018). Safer Streets 2018. *AtlantaGA*.

<https://www.atlantaga.gov/home/showpublisheddocument?id=48105>.

Department of City Planning. (2018). Atlanta's Transportation Plan Final Report. *AtlantaGA*

<https://w>

www.atlantaga.gov/home/showpublisheddocument/48083/637353603332230000.

Insurance Institute for Highway Safety, Highway Loss Data Institute. (2022). *Fatality Facts*

2020 General statistics: State by state. [Data Set]. Insurance Institute for Highway Safety.

<http://www.iihs.org/iihs/topics/t/general-statistics/fatalityfacts/state-by-state-overview>