Task 1 - Bring Your Own Data

DATASET:
US Accidents (2016 - 2023) (kaggle.com)

INTRODUCTION:
Road traffic accidents are the world's leading cause of death for individuals between the ages of one and twenty-nine. Throughout the world, cars, buses, trucks, motorcycles, pedestrians, animals, taxis, and other categoriesof travelers, share the roadways, contributing to economic and social development in many countries. Yet each year,many vehicles are involved in crashes that are responsible for millions of deaths and injuries. Globally, every year, about 1.25 million people are killed in motor vehicle crashes and approximately 50 million more are injured. Following current trends, about two million people could be expected to be killed in motor vehicle crashes each yearby 2030 [3].
Our project is based on a tabular dataset of US car accidents from 2016 to 2023. The dataset gives an insight into the traffic behaviors as well as the severity of these accidents. The dataset is highly significant as it provides the exact location and the effect of the accident which usually causes delays for other drivers on the road.

BACKGROUND:
Traffic accident prediction models are very useful tools in highway safety, given their potential for determining both the frequency of accident occurrence and the contributing factors that could then be addressed by transportation policies. Vehicular crash data can be used to model both the frequency of crash occurrence and the degree of crash severity. Crash frequency refers to the prediction of the number of crashes that would occur on a specific road segment or intersection in a time period. Crash severity methods generally explore the relationship between crash severity injury categories and contributing factors such as driver behavior, vehicle characteristics, roadway geometry, and road-environment conditions. Traffic accident-related fatalities and injuries can be prevented or at least minimized by a joint involvement from multiple sectors (i.e. transportation agencies, police, health departments, education institutions) that oversee road safety, vehicles, and the drivers themselves. Effective interventions include the design of safer infrastructure and incorporation of road safety features into land-use and transport planning; improvement of vehicle safety features; improvement of post-crash care for victims of road crashes, and improvement of driver behavior, such as setting and enforcement laws relating to key risk factors, and raising public awareness [3].

MOTIVATION:
We're motivated by a simple yet powerful belief: accidents aren't unavoidable, and we can prevent many of them by making informed choices. Most of the accidents happen in either good or bad weather. We can clarify the causes of road accidents and minimize the accident rate based on the findings of these data analyses. We aim to use the potential of data science and machine learning to create tools and strategies that help everyone i.e. authorities, drivers, and commuters make safer decisions on the road.

OBJECTIVES:
Traffic accidents are the greatest cause of death worldwide, taking the lives of millions of people each year due to their regularity. As a result, technology that predicts traffic accidents or accident-prone areas may be able to save lives. Nowadays, there is an increasing emphasis on traffic accident data mining and analysis, which can improve in-depth investigation and reduce traffic-related deaths [2].
The first objective of this project is to recognize key factors affecting the accident severity. The second one is to develop a model that can accurately predict accident severity. To be specific, for a given accident, without any detailed information about itself, like driver attributes or vehicle type, this model is supposed to be able to predict the likelihood of this accident being a severe one. The accident could be the one that just happened and still lack of detailed information, or a potential one predicted by other models

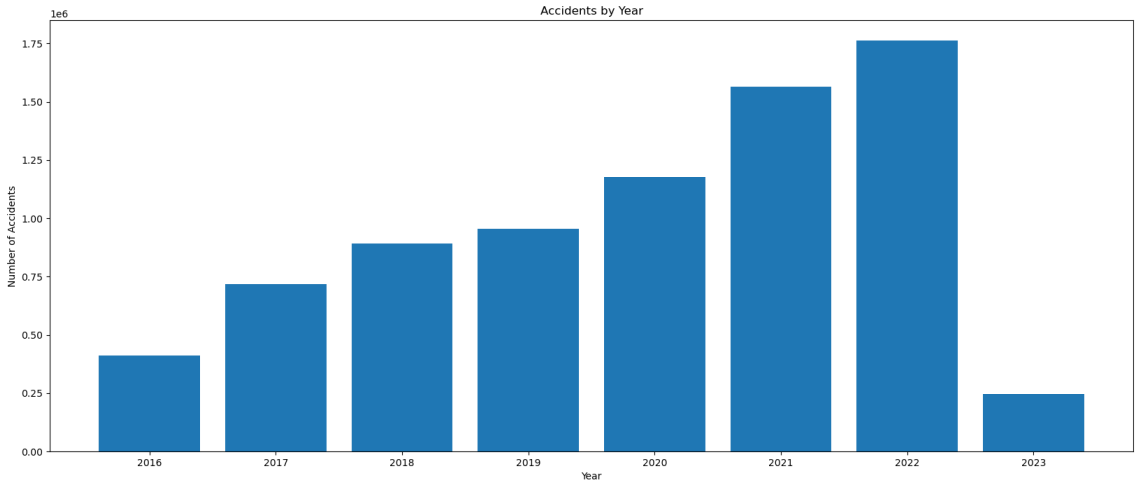
If we can crack the challenges presented by this dataset it has the potential that could lead to substantial improvements in road safety. It could lead to:

Reduced Accidents: We'll spot danger zones and times to target safety efforts where they're needed most.
Speedy Help: Predictive tools can help emergency crews arrive quicker, possibly saving lives.
Improved Infrastructure: City planners get the scoop on how to make roads safer and keep them in top shape.
Smart Travel: Commuters and drivers will have real-time accident risk info, making their journeys safer.

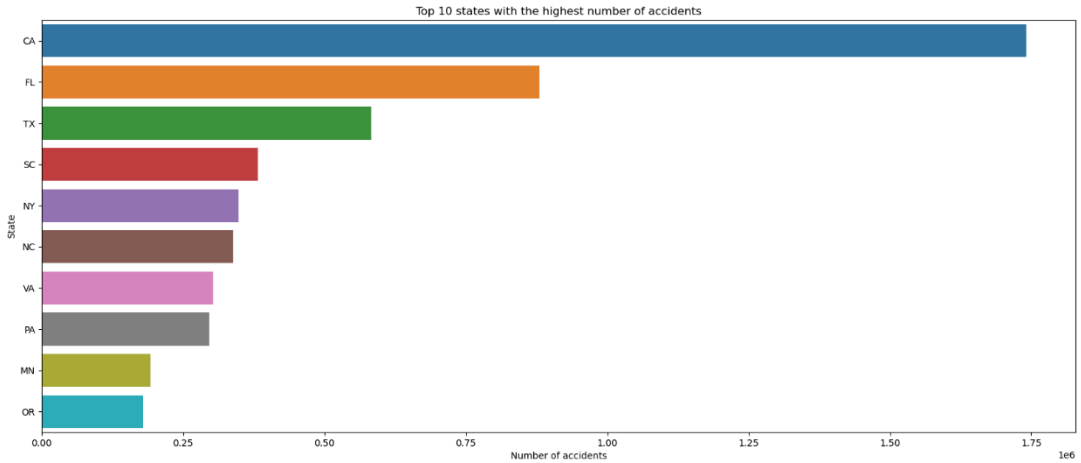
While the dataset promises immense value, we anticipate challenges related to data quality and the complexity of modeling accident prediction. Navigating these hurdles will be crucial to unlocking its full potential.

EDA:

Number Of Accidents by Year:

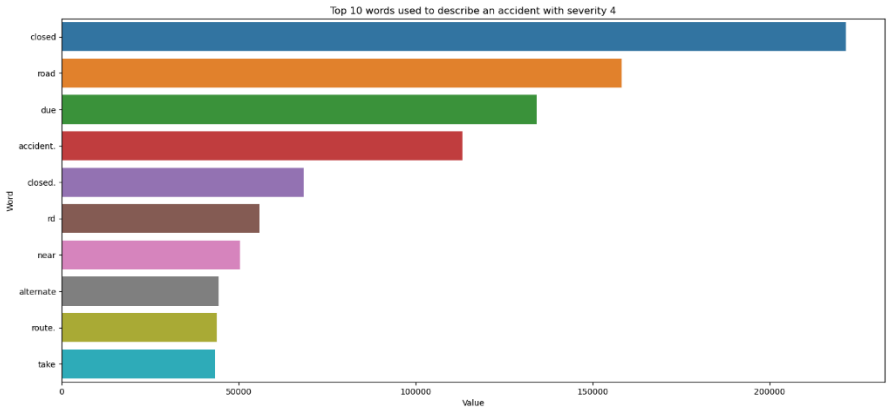


States With Highest Number Of Accidents:



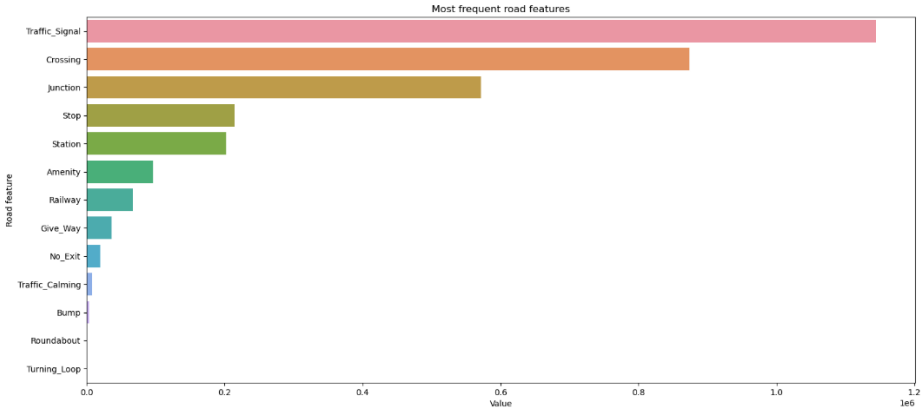
As noticed, California stands out with the highest number of accidents, approximately twice as many as the next state on the list.

Frequent Words in Description where Severity is 4:
We will analyze the most commonly occurring words in the "description" column of accidents with a severity level of 4. To do this, we'll exclude common English stopwords from the analysis.



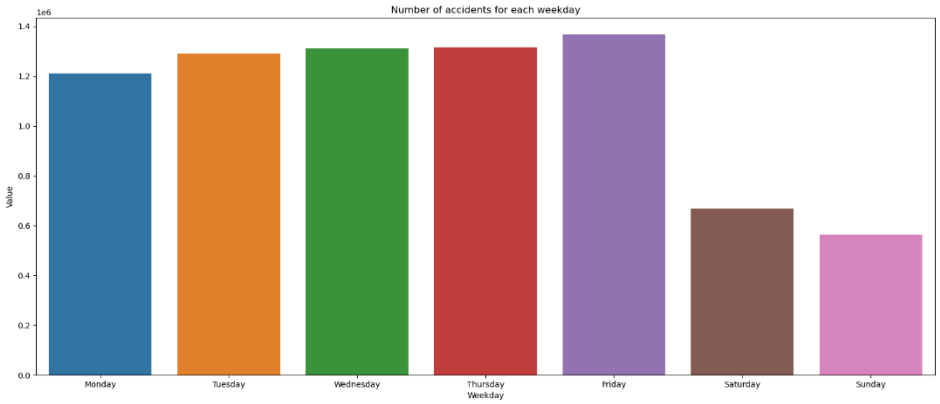
We can see that the most used word in the description is *closed*. Subsequent words are *accident*, *due* and *road.*

Most Frequent Road Features:



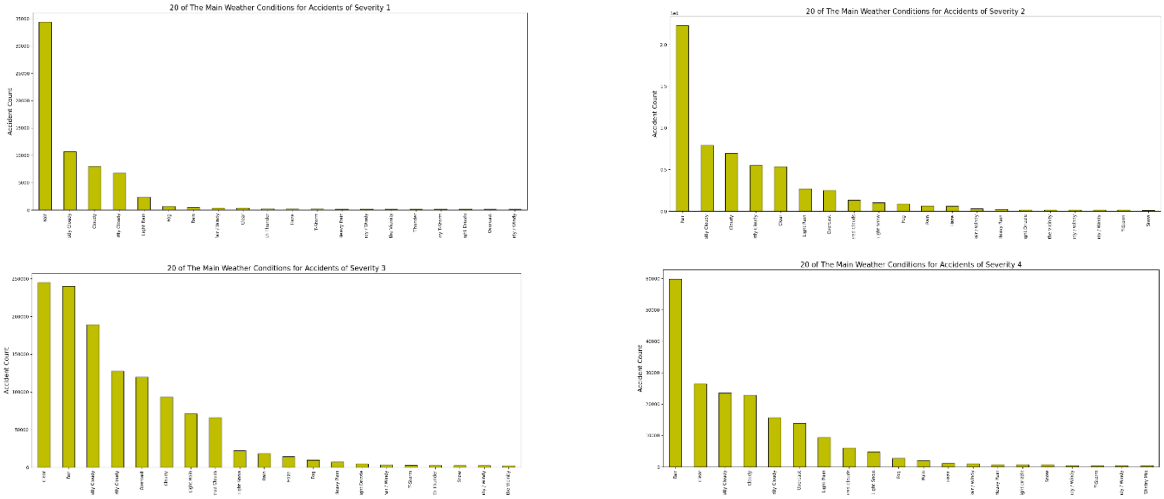
As observed, a significant number of accidents took place near a traffic signal, particularly in areas where junctions or crossings were located.

Number of Accidents by Weekday:



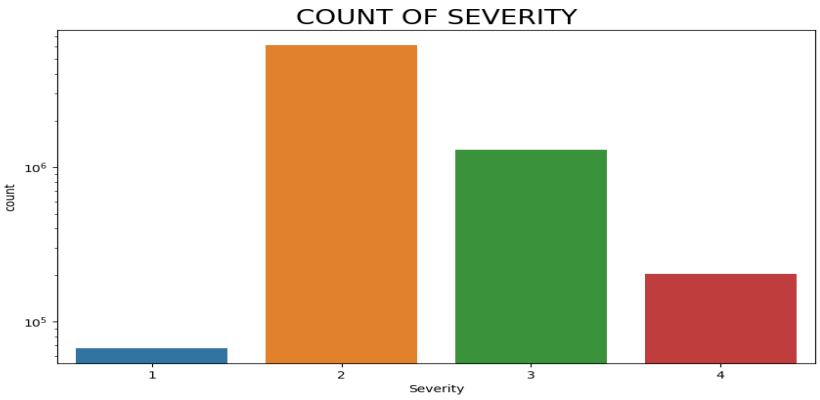
The days with the highest number of accidents are typically weekdays, while the frequency of accidents during the weekend is nearly half as much. This pattern could be attributed to the reduced traffic volume on weekends.

Weather Conditions for each Severity:



In most cases the weather condition is Fair / Clear.

Number of Accidents by Severity:

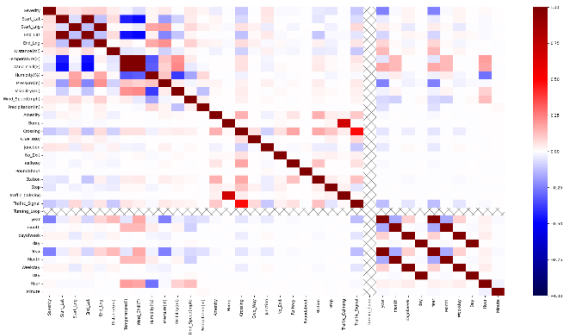


The number of accident with severity 2 is much higher.

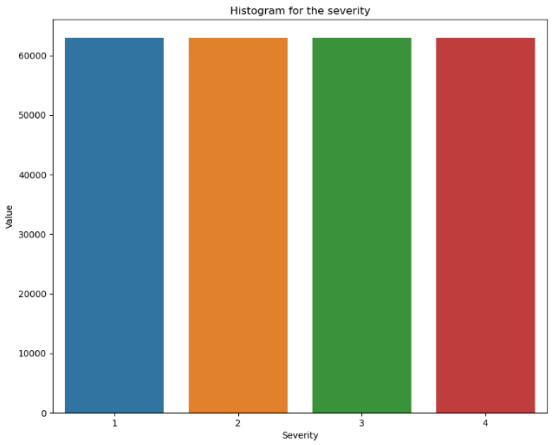
PREPROCESSING:

We prepare the dataset for further analysis, which includes tasks such as feature engineering and model development. This process encompasses actions like converting data types, extracting features, cleansing data, performing encoding, and achieving a balanced dataset.

1. Convert Start_Time to Datetime:
   X["Start_Time"] = pd.to_datetime(X["Start_Time"]): Convert the 'Start_Time' column to a datetime format.
2. Extract Date and Time Features:
   X["Year"] = X["Start_Time"].dt.year: Extract the year from the 'Start_Time' column.
   X["Month"] = X["Start_Time"].dt.month: Extract the month.
   X["Weekday"] = X["Start_Time"].dt.weekday: Extract the weekday (0 for Monday, 6 for Sunday).
   X["Day"] = X["Start_Time"].dt.day: Extract the day of the month.
   X["Hour"] = X["Start_Time"].dt.hour: Extract the hour of the day.
   X["Minute"] = X["Start_Time"].dt.minute: Extract the minute of the hour.
3. Correlation:
   Compute the correlation matrix of the entire dataset and plot it using a heatmap.



4. Attribute Selection:
   Create a list of features to be dropped and remove them from the dataset using X = X.drop(features_to_drop, axis=1).
5. Remove Duplicates:
   Calculate and print the number of rows.
   Remove duplicate records using X.drop_duplicates(inplace=True) and print the number of rows after removing duplicates.
6. Delete Records with Zero Values:
   Describe 'Pressure(in)' and 'Visibility(mi)' columns.
   Remove rows with zero values in these columns: X = X[X["Pressure(in)"] != 0] and X = X[X["Visibility(mi)"] != 0].
7. Weather_Condition:
   Perform data cleaning and grouping of weather conditions into categories.
8. Wind_Direction:
   Perform data cleaning for wind direction categories.
9. Fill Missing Values for Numeric Features:
   Find columns with missing values.
   Fill missing values in specific numeric features with their mean.
10. Delete Records with Missing Values for Categorical Features:
    Remove records with missing values in categorical features.
11. Balance Data:
    Balance the dataset by randomly sampling the same number of records for each severity level.



12. Feature Scaling:
    Normalize specific numerical features using Min-Max scaling.
13. Categorical Features:
    Cast selected columns to categorical data type.
14. Replace True and False with 1 and 0:
    Replace boolean values with 1 and 0.
15. One-Hot Encoding:
    Perform one-hot encoding on selected categorical features.

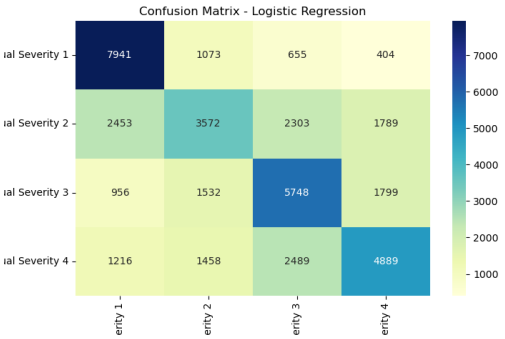
16. Binary Encoding for 'City':
    Apply binary encoding to the 'City' column and concatenate it with the dataset.

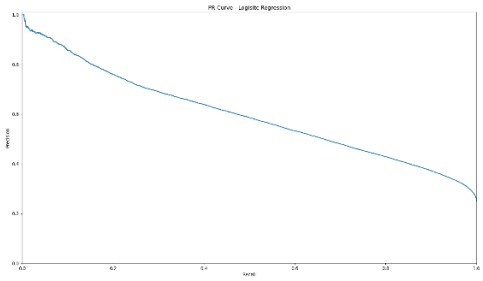
MODEL SELECTION, TRAINING, AND OPTIMIZATION:

A diverse set of machine learning models are selected, trained, and optimized to achieve the best possible performance on the dataset.These include models like Logistic Regression, Decision Tree, Random Forest, Support Vector Machine (SVM), and Naive Bayes classifiers.

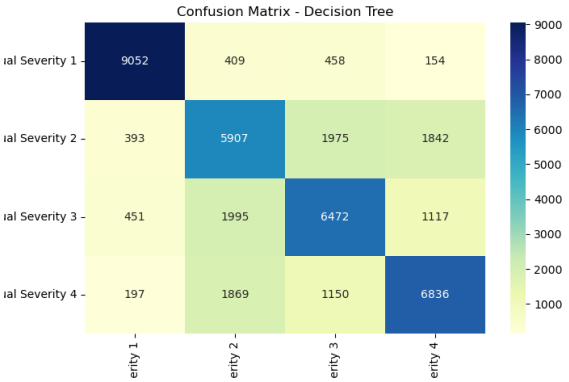
For Optimization, Hyperparameter tuning is performed using GridSearchCV on different models, Different sample sizes are tested, Various iterations of models are tested.

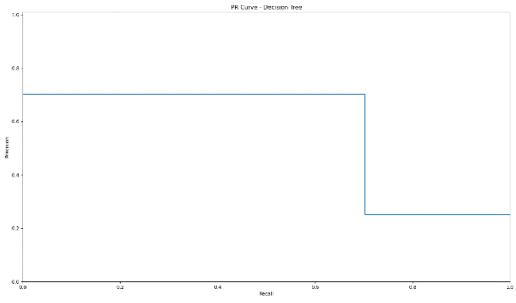
- Logistic Regression:
    - A Logistic Regression model is created with the LogisticRegression class from scikit-learn.
    - Hyperparameter optimization is performed using GridSearchCV. This involves searching over different solver options ("newton-cg", "sag", "saga") to find the best configuration.
    - The model is trained and validated on the split training and validation datasets (X_train and X_validate) to assess performance.
    - Classification reports are generated to display performance metrics like precision, recall, F1-score, etc.
    - A confusion matrix is visualized to illustrate the agreement between predicted and actual classes.
    - A Precision-Recall curve is plotted to evaluate the model's ability to trade off precision and recall.

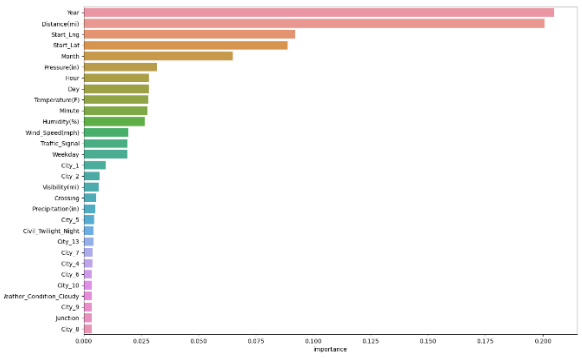




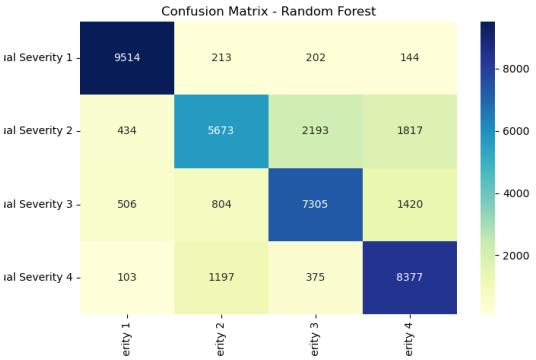
- Decision Tree:
    - A Decision Tree classifier is created with the DecisionTreeClassifier class from scikit-learn.
    - GridSearchCV is employed to optimize hyperparameters (criterion and max depth).
    - Training and validation scores are printed to evaluate how well the model fits the data.
    - Feature importances are calculated and the top 30 features are visualized.
    - Classification reports and confusion matrices are generated for performance evaluation.
    - A Precision-Recall curve is plotted to examine the precision-recall trade-off.

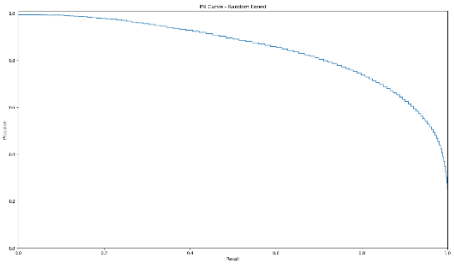




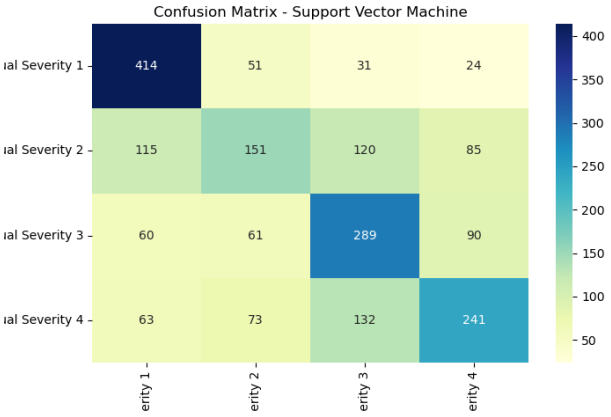


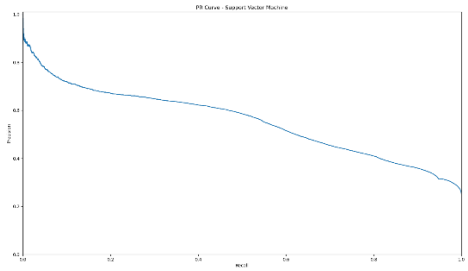
- Random Forest:
  - A Random Forest classifier is created using the RandomForestClassifier class from scikit-learn.
  - Hyperparameter optimization is performed using GridSearchCV, searching over the number of estimators (n_estimators) and the maximum depth of the trees (max_depth).
  - The model is trained and validated, and training and validation scores are printed.
  - Classification reports, confusion matrices, and Precision-Recall curves are generated to evaluate model performance.



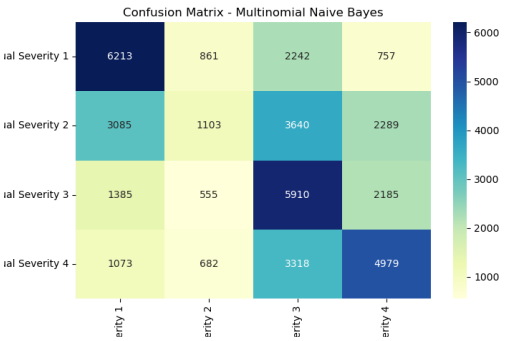


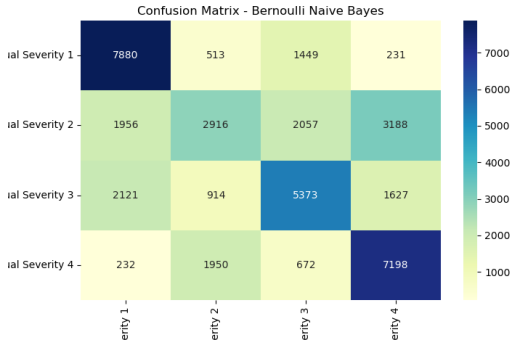
- Support Vector Machine:
  - Support Vector Machine (SVM) models are created and optimized using GridSearchCV. The code tests different kernel functions ("linear", "rbf", "sigmoid", and "poly") and hyperparameters.
  - Training and validation scores are printed for models tested with different kernel functions.
  - Classification reports and confusion matrices are generated for performance evaluation.
  - Precision-Recall curves are plotted for models with different kernel functions, considering two sample sizes (5,000 and 10,000).

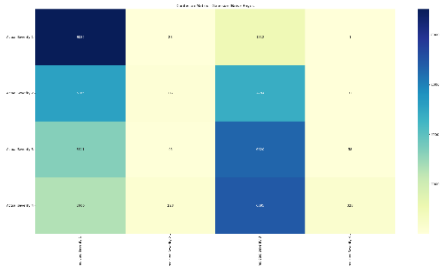


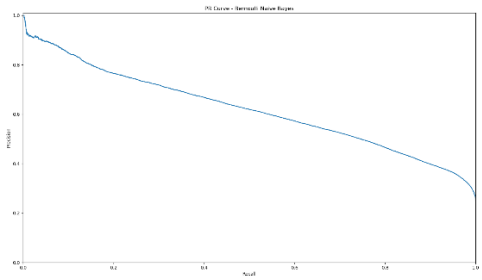


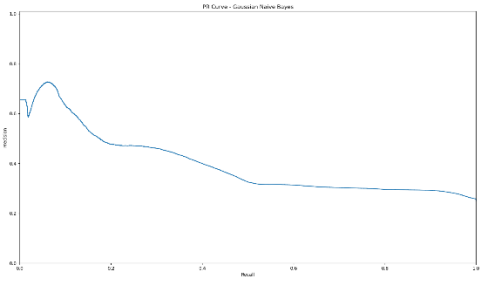
- Naive Bayes Models:
    - Three variants of Naive Bayes classifiers (Gaussian, Multinomial, and Bernoulli) are evaluated.
    - Each variant is created and trained, and training and validation scores are calculated.
    - Classification reports and confusion matrices are generated to assess the performance of each Naive Bayes variant.
    - Precision-Recall curves are plotted for each Naive Bayes variant.

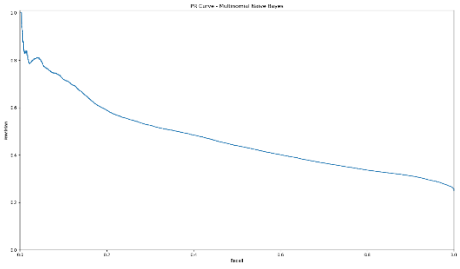






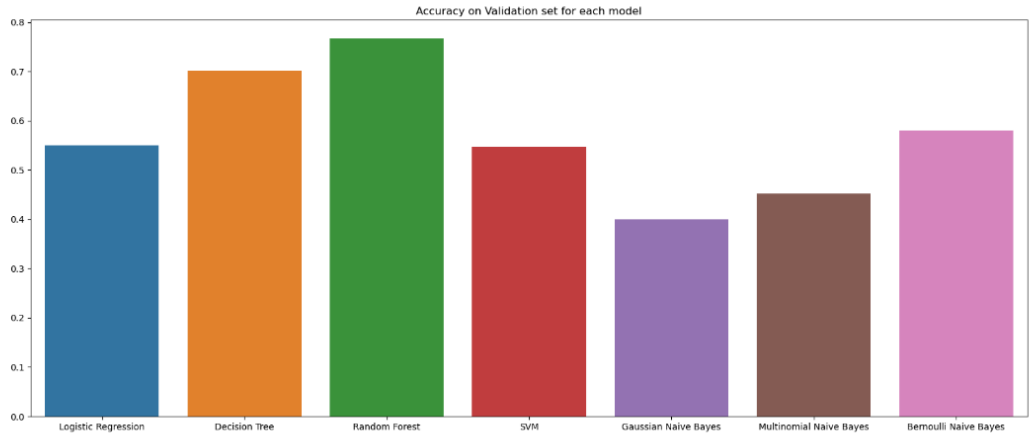


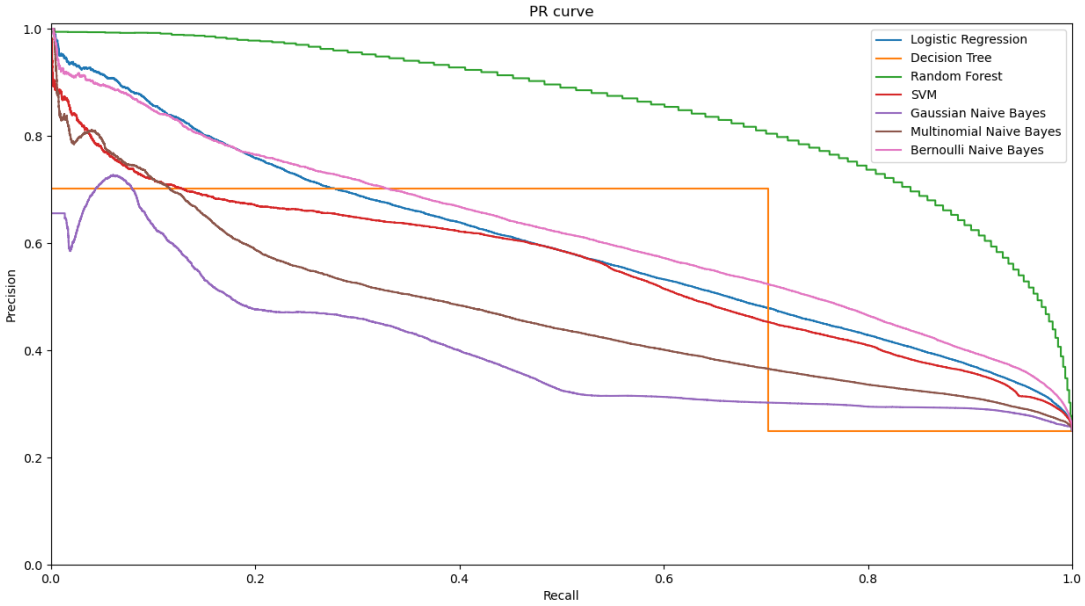




MODEL EVALUATION:

We've employed performance metrics like accuracy, precision, recall, and F1 score to evaluate various machine learning models. These metrics help us assess and compare how well the models are performing.





REFERENCES:

1. "Motor vehicle - Introduction - Injury Facts," *Injury Facts*, Apr. 18, 2023. https://injuryfacts.nsc.org/motor-vehicle/overview/introduction/
2. Reddy, S. S., Chao, Y. L., Kotikalapudi, L. P., & Ceesay, E. (2022). Accident analysis and severity prediction of road accidents in the United States using machine learning algorithms. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9247097
3. Abdulhafedh, A. (2017). Road crash prediction models: different statistical modeling approaches. *Journal of Transportation Technologies*, *07*(02), 190–205. https://doi.org/10.4236/jtts.2017.72014