MAGALITH DATA QUEST 2024

TEAM:
PAWAN262.WORK

OVERVIEW

OBJECTIVES

Develop a **Machine Learning** model to predict **Accident severity**, identify **key risk factors**, and recommend **Safety Improvements**.

Understanding Data
Identifying key features in Dataset & analysing problem

Model Building, Performance & Impact
Creating model and testing it on accuracy basis

- **Problem Analysis**
- **Dataset Creation**
- **Lazy Classifier**
- **Neural-Network Keras Classifier**
- **Hyperparameter Tuning & Accuracy**
- **Feature Importance**

EXECUTIVE SUMMARY OF MACHINE LEARNING MODEL

Data Cleaning & Labeling
Data Cleaning (handling null values, encoding categorical variables for visualization and analysis)

EDA
Exploratory Data Analysis (EDA) to visualize features using diverse plots and identify correlations via heatmaps

Feature Engineering
Feature engineering to create new features for robust model evaluation and increase the dimensionality of the data for training

Model Building & Evaluation
The processed data was utilized to train multiple classifiers, selecting the one with the best accuracy

Hyper-Parameter Tuning
Hyperparameter tuning was conducted on the selected classifier (Keras) to enhance performance

Identifying Important Features
Finally, important features were identified and visualized

MODEL PERFORMANCE OVERVIEW

ACCURACY MEASURES

- **Precision**
- **Recall**
- **F1 Score**

75%
Model run on 3 fold cross-validation achieving 75% accuracy

MODEL PIPELINE

Feature Engineering

- Features created such as Total Lane Width, Road Risk Index, Lane Utilization LCE, Is-sppeding & Speed Ratio
- Dimensionality Increases and Robust Data Correlation achieved

Normalisation, Log Transformation, K-means clustering, Cyclic Transformation (Sin Conversion) on features creating more correlated dataset

Random Forest | Keras | Decision Tree | Extra Trees

Lazy Classifier Algorithm

Lazy Classifier Algorithm Splits the entire data in Train and Test and run all the Machine Learning classifiers including RandomForest, DecissionTrees and so on and give the classification Report of each Algorithm

Keras Classifier

Building a neural network based Keras Classifier to get more acurate results and comparing them with other Classifiers

Hyper-Parameter Tuning

Keras Classifier optimized using Keras Tuner to set the optimum Parameters enhancing the accuracy of Model

3-Fold Cross Validation used to get more optmized results

Important Features

MODEL IMPLEMENTATION

*Please find the data model and implementation on Python here

NEURAL NETWORK ARCHITECTURE

First Dense Layer

A fully connected layer with N1 neurons was added. The layer computes:

$$z_1 = W_1 X_{\text{scaled}} + b_1$$

$$a_1 = \text{ReLU}(z_1), \text{ where } \text{ReLU}(x) = \max(0, x).$$

$$L_2 = \lambda \sum w^2, \text{ with } \lambda \text{ as the tunable penalty term.}$$

Dropout Layer

$$\text{Dropout Output} = a_1 \cdot \text{Bernoulli}(1-p)$$

dropout rates, and L2 regularization to control overfitting. Used relu for hidden layers and softmax for output to handle multi-class classification.

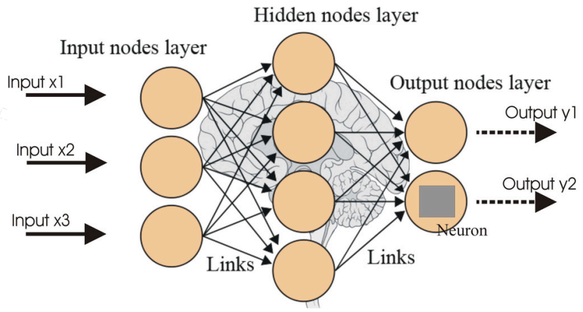
Output Layer

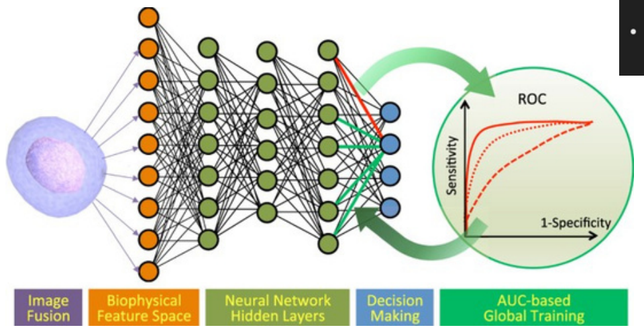
Output Layer: K-class softmax output:

$$y_{\text{pred}} = \text{softmax}(z) = \frac{e^{z_k}}{\sum_{j=1}^{K} e^{z_j}}, \; k \in \{1, 2, ..., K\}$$

Integrated Keras Tuner for optimizing layer units, dropout rates, L2 regularization, and optimizers (Adam, Nadam, RMSprop).

NEURAL NETWORKS BASED KERAS CLASSIFIER





Hyperparameter Tuning:

- **Tunable Parameters:**
  - Layer units $N: N \in \{64, 96, \ldots, 256\}$
  - L2 regularization $\lambda \in \{0.01, 0.001, 0.0001\}$
  - Optimizers $(\text{Adam}, \text{Nadam}, \text{RMSprop})$
- Random Search Optimization: Objective function $J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} \log P(y_i|X_i;\theta)$, minimized for validation accuracy.

Training and Early Stopping:

- Loss Function: Sparse categorical cross-entropy for multi-class classification:

$$L = -\frac{1}{m}\sum_{i=1}^{m}\sum_{k=1}^{K} y_{i,k} \log y_{\text{pred},k}$$

- Early stopping criterion based on validation loss, monitoring:

$$\Delta L_{\text{val}} < \epsilon \text{ for } p \text{ epochs}$$

Feature Importance:

- Weight Importance: Extracted weights W_1 from the first dense layer. Feature importance calculated as:

$$I_j = \frac{1}{N}\sum_{i=1}^{N} |w_{ij}|$$

where w_{ij} is the weight of the j-th feature for the i-th neuron.

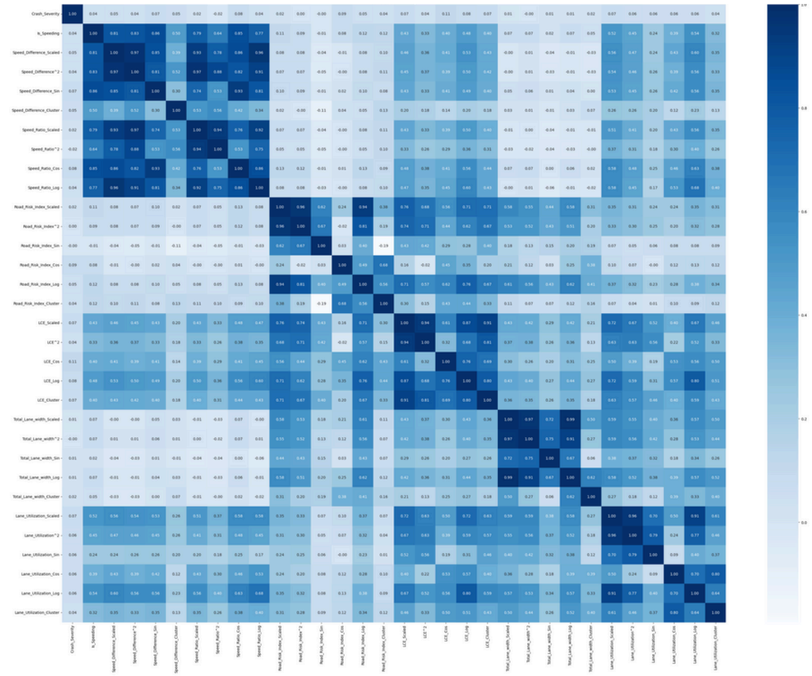
- Visualized using a bar plot for interpretability.

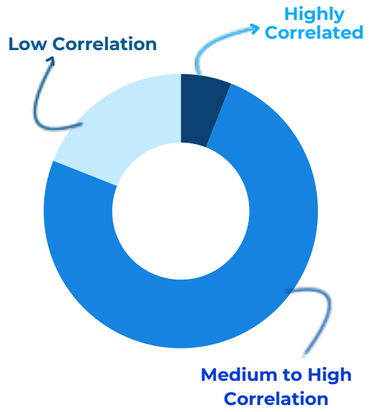
DATA PREPROCESSING & FEATURE ENGINEERING

```
[ ]: X['Is_Speeding']= (data_1['Vehicle_Speed']>data_1['Speed_Limit']).astype(int)

X['Speed_Ratio']= data_1['Vehicle_Speed']/data_1['Speed_Limit']

X['Speed_Difference']= data_1['Vehicle_Speed'] - data_1['Speed_Limit']

X['Road_Risk_Index']= data_1['Lane_Width']*data_1['Number_of_Lanes']*data_1['Road_Surface_Condition_severity']

X['LCE']= data_1['Vehicle_Speed']*data_1['Lane_Width']*data_1['Number_of_Lanes']*data_1['Road_Surface_Condition_severity']

X['Total_Lane_width']=data_1['Lane_Width']*data_1['Number_of_Lanes']

X['Lane_Utilization']=data_1['Vehicle_Speed']*data_1['Lane_Width']*data_1['Number_of_Lanes']
```

Using various transformations (for eg. Log Transformation, Clustering, Normalisation, Scaling, Sin Transformation, Binning) more robust dataset is created

Correlation Heatmap





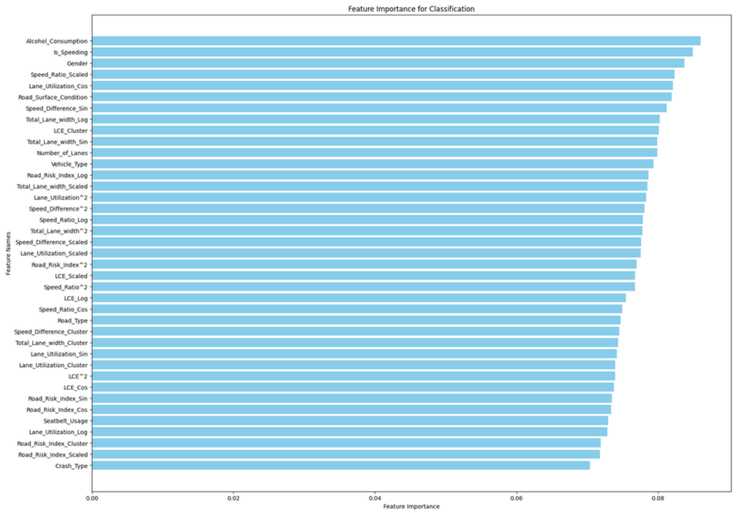
CORRELATION OVERVIEW

IDENTIFYING IMPORTANT FEATURES

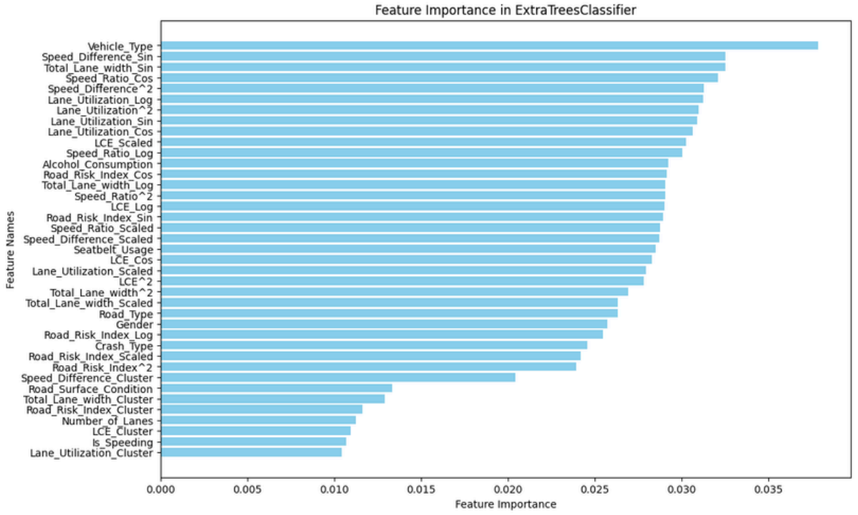
*Feature importance is in descending order from top to bottom

BAGGING CLASSIFIER	DECISION TREE CLASSIFIER	GRADIENT BOOSTING CLASSIFIER	RANDOM FOREST CLASSIFIER	EXTRA TREES CLASSIFIER	(NEURAL-NETWORKS) KERAS CLASSIFIER
Road_Risk_Index_Cos	Speed_Difference_Sin	Speed_Difference_Sin	Speed_Difference_Sin	Vehicle_Type	Alcohol_Consumption
LCE_Cos	Road_Risk_Index_Sin	Road_Risk_Index_Sin	Total_Lane_width_Sin	Speed_Difference_Sin	Is_Speeding
Total_Lane_width_Sin	Speed_Ratio_Cos	Total_Lane_width_Sin	Lane_Utilization_Sin	Total_Lane_width_Sin	Gender
Speed_Difference_Sin	Lane_Utilization_Sin	Lane_Utilization_Sin	Road_Risk_Index_Cos	Speed_Ratio_Cos	Speed_Ratio_Scaled
Lane_Utilization_Sin	LCE_Cos	Speed_Ratio_Cos	Speed_Ratio_Cos	Lane_Utilization_Log	Lane_Utilization_Cos
Vehicle_Type	Total_Lane_width_Sin	LCE_Cos	LCE_Scaled	LCE_Scaled	Road_Surface_Condition
Speed_Ratio_Scaled	Crash_Type	Vehicle_Type	Vehicle_Type	Alcohol_Consumption	Speed_Difference_Sin
Gender	Vehicle_Type	Alcohol_Consumption	Gender	Road_Risk_Index_Cos	Total_Lane_width_Log

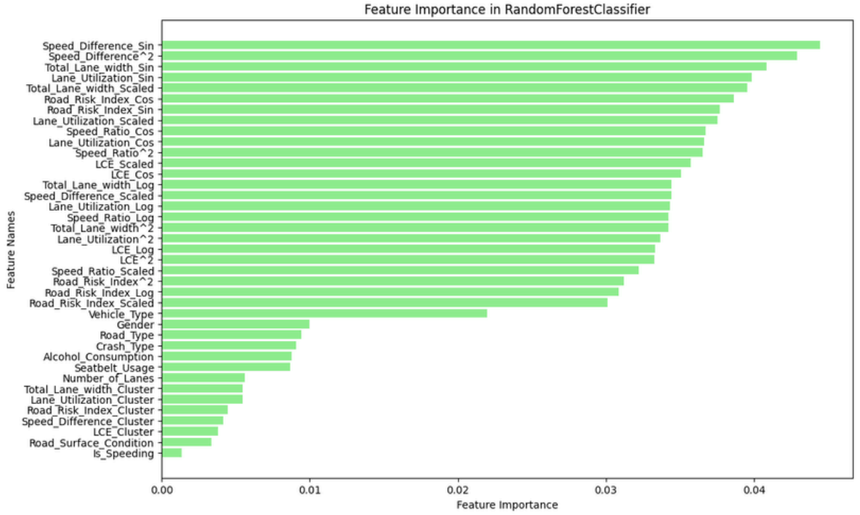
PLOT OF FEATURE IMPORTANCE



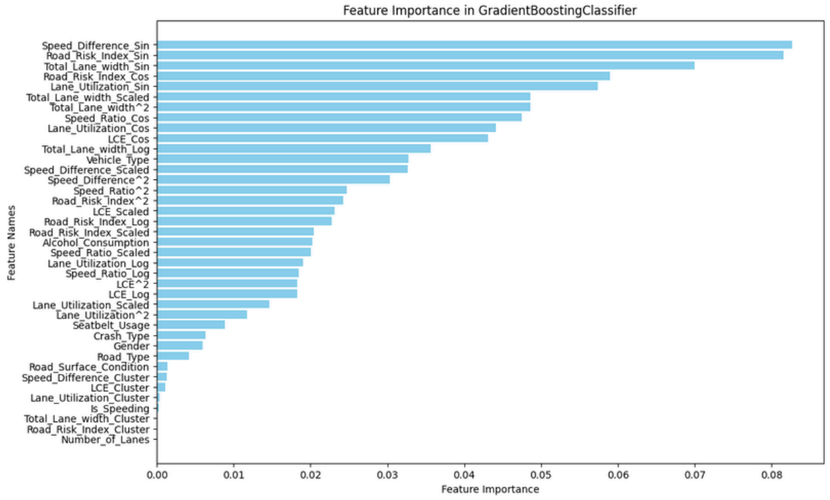
Keras Classifier



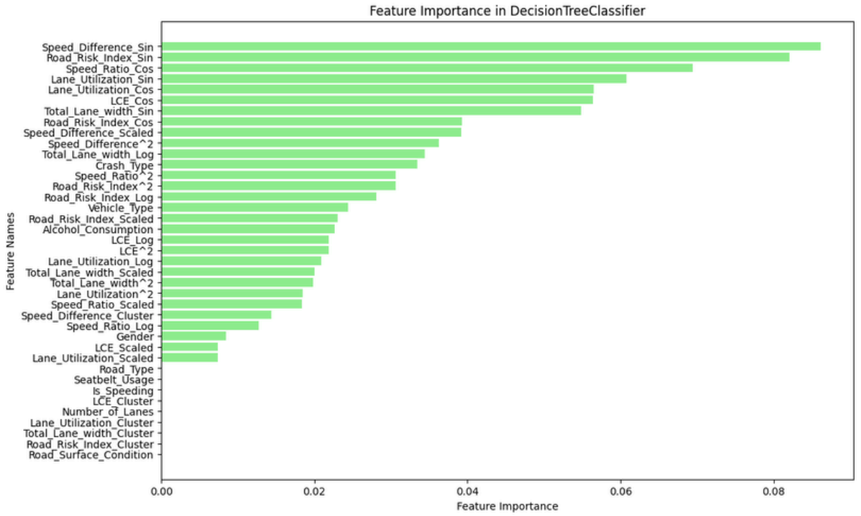
ExtraTrees Classifier



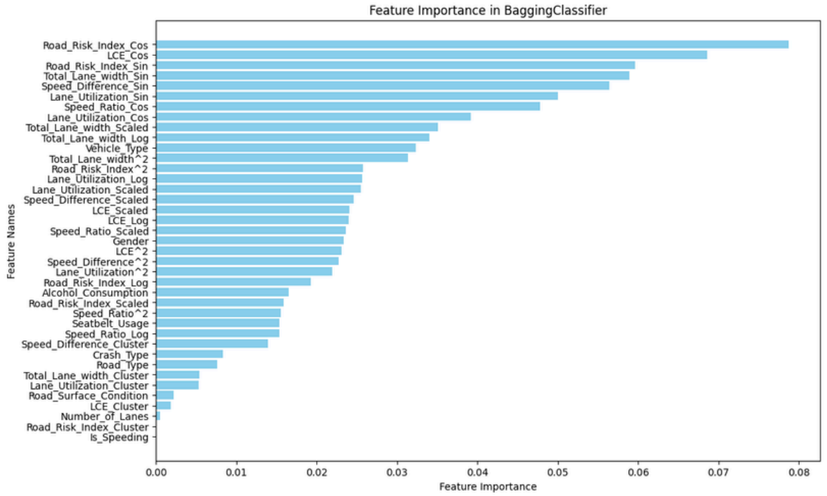
RandomForest Classifier



GradientBoosting Classifier



DecisionTree Classifier



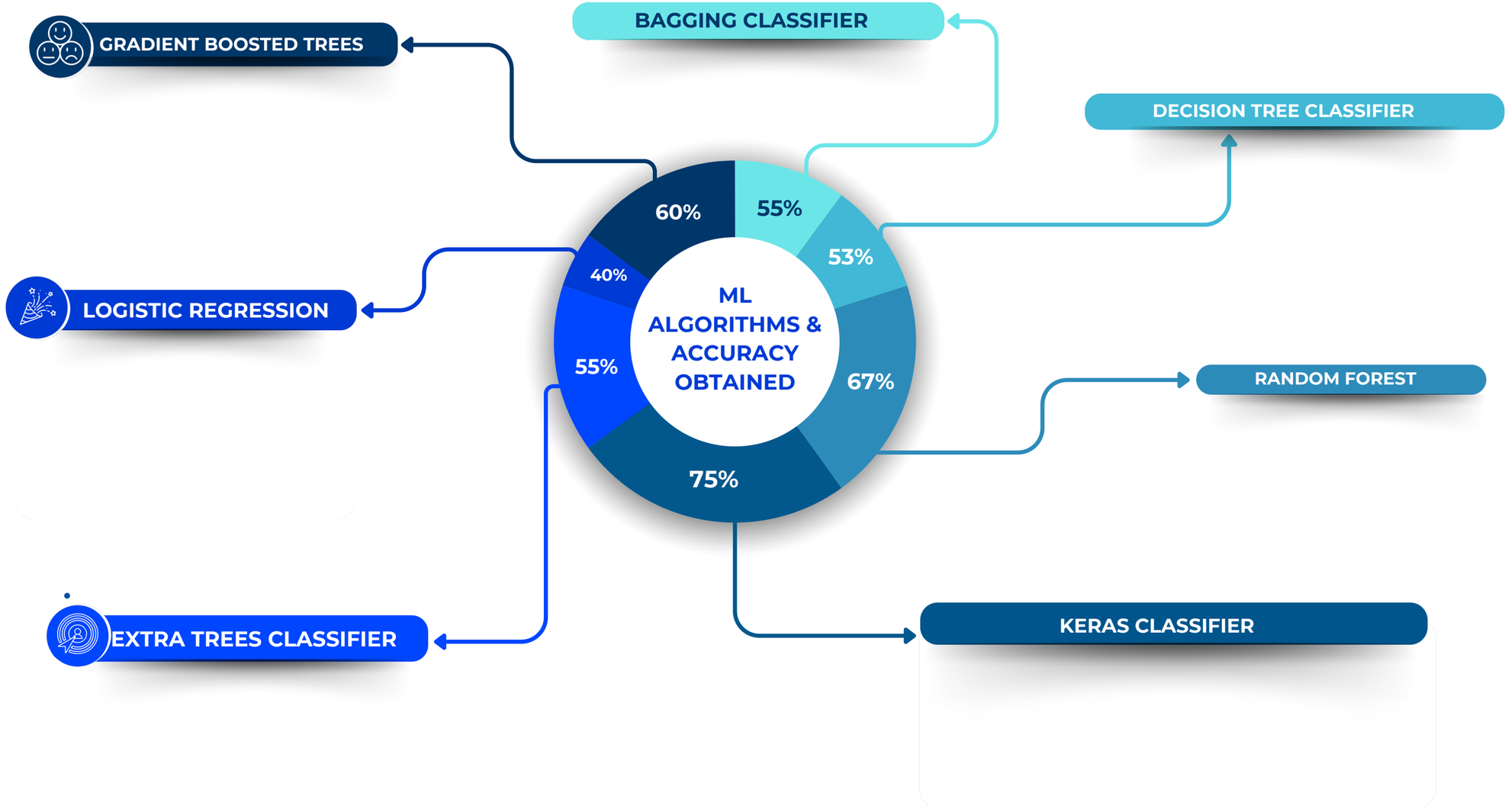
Bagging Classifier

OVERALL ANALYSIS

A COMPREHENSIVE ANALYSIS OF MODEL PERFORMANCE AND LIST OF IMPORTANT FEATURES

Features With Decreasing Weightage

1. Speed Difference
2. Road Risk Index
3. Total Lane Width
4. Road Surface Condition
5. Alcohol Consumption
6. Lane Utilisation
7. LCE & Vehicle Type



ROAD IMPROVEMENT & SAFETY MEASURES

SAFETY RECOMMENDATION

Monitoring Alcohol Consumption

- Install breathalyzer checkpoints at key locations to test drivers for alcohol levels.
- Use AI-powered cameras to detect driving behavior linked to alcohol impairment.
- Implement stricter penalties and fines for violations detected during monitoring.
- Raise awareness about the dangers of drunk driving through campaigns.

Speed Monitoring

- Deploy speed cameras at critical locations to identify vehicles exceeding speed limits
- Integrate speed data into automated traffic management systems for quick action.
- Implement variable speed limits based on traffic density and road conditions.

Speed Limit Setup

- Determine speed limits based on factors like Road Risk Index and LCE.
- Implement dynamic speed limits that adjust to weather, traffic, and road conditions.
- Use real-time data from IoT sensors to optimize speed limit recommendations.
- Install visible and interactive signboards to communicate speed limits effectively.

Vehicle Type Analysis

- Categorize accident data by vehicle type (e.g., two-wheelers, heavy vehicles).
- Deploy targeted safety measures, such as separate lanes for heavy vehicles.
- Monitor and analyze vehicle-specific behavior to design tailored safety strategies.

ROAD CONDITION IMPROVEMENTS

Reducing Bias

Fairlearn Toolkit: Mitigates bias by applying fairness constraints such as demographic parity, equal opportunity

EXISTING ROAD PARAMETERS

- **Accurately calculate Lane Condition Efficiency (LCE) and Road Risk Index to evaluate road parameters.**
- **Use them to set speed limits and safety instructions.**

ROAD PLANING PARAMETERS

- **Use factors suggested by us, like the number of lanes, total lane width, and lane utilization.**
- **Take these factors into account while planning new road designs.**

Road Safety Signs

Warning boards should be set up on harsh road conditions such as wet and icy roads.

These boards should highlight potential danger to enhance rider awareness.

The boards should also advise safe riding speeds to prevent accidents.

Thank you!