Predicting Important Factors of NYC's Traffic Accidents

EL07: Syntax Error

1. Francis Nathan (U2120951H)
2. Matthew Dinata (U2120122G)
3. Muhammad Afiq (U2122587K)
4. Nigel Loh (U2122947F)

Introduction

Our Dataset

New York City Traffic Accidents from January - August 2020 classified into categories, such as CRASH DATE and TIME, CONTRIBUTING FACTOR VEHICLE, VEHICLE TYPE CODE, LATITUDE, LONGITUDE, etc.

Objective

Predict which factor most contributes to the number of persons injured in the traffic accident. This will help the government and police department in New York City set up suitable measurements and actionables to prevent more accidents.

Table of Contents

01

Introduction

Problems +
Objectives

02

Preparation

Data Cleaning

03

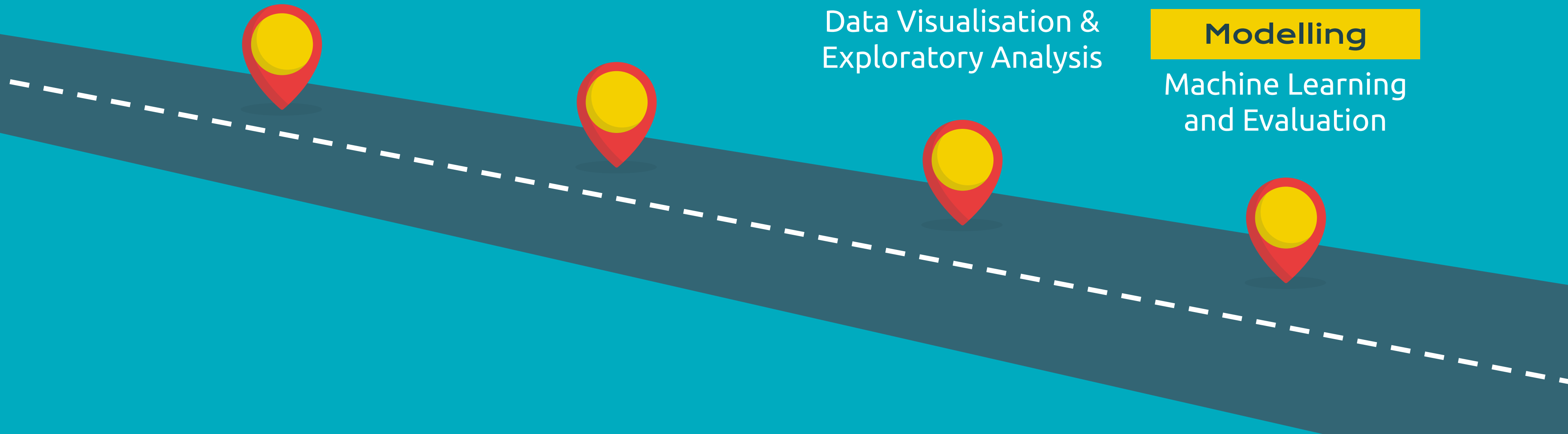
Visualisation

Data Visualisation &
Exploratory Analysis

04

Modelling

Machine Learning
and Evaluation



DATA PREPARATION

Data Preparation Contents

01

IMPORT

Importing packages and raw data

02

REMOVE COLUMNS

Dropping irrelevant columns

03

REMOVE NULL VALUES

Dropping NULL or ‘Unspecified values

04

DATA CONVERSION

Conversion of ‘CRASH DATE’ to Pandas DateTime

05

OVER-SAMPLING

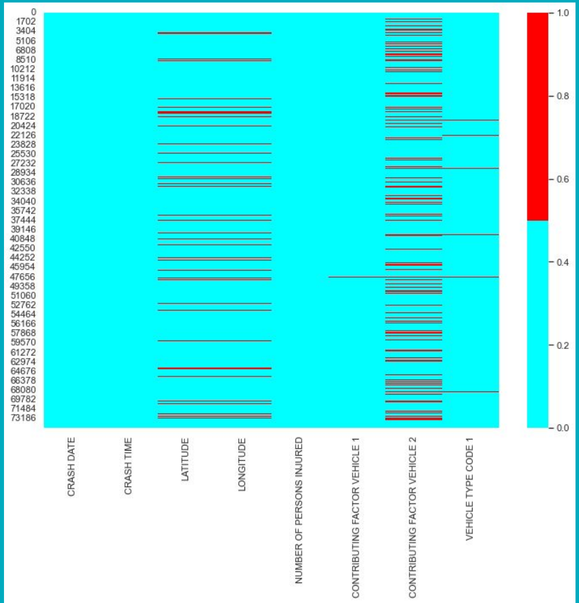
Balancing data using RandomOverSampler

06

REDUCE DATA

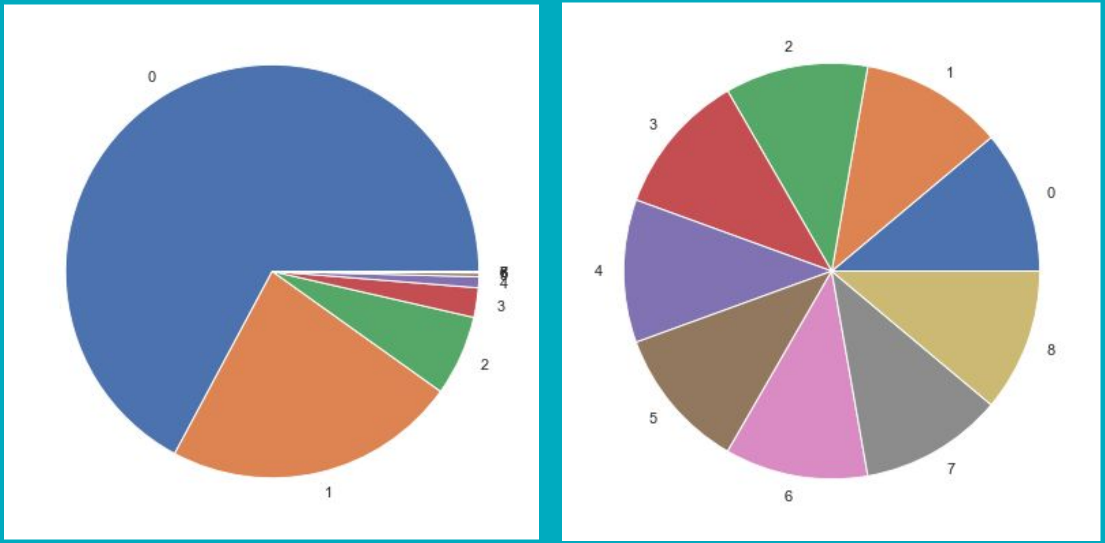
Removal of random data rows to reduce data size

REMOVING NULL VALUES



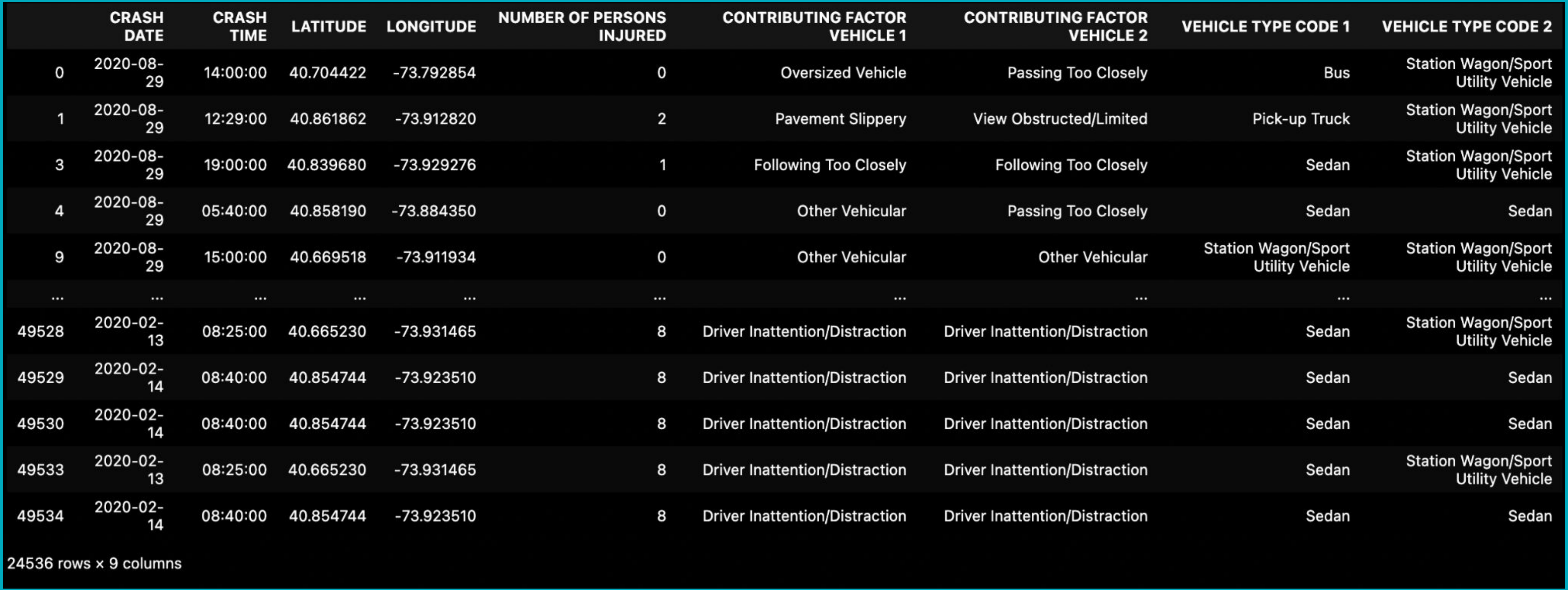
Based on the heatmap above, the color cyan indicates `No Missing Values` while the color red indicates `Missing Values`.

BALANCING DATA USING RANDOM OVER-SAMPLING



Comparison of the data before and after balancing using RandomOverSampler

REMOVAL OF RANDOM DATA ROWS



	CRASH DATE	CRASH TIME	LATITUDE	LONGITUDE	NUMBER OF PERSONS INJURED	CONTRIBUTING FACTOR VEHICLE 1	CONTRIBUTING FACTOR VEHICLE 2	VEHICLE TYPE CODE 1	VEHICLE TYPE CODE 2
0	2020-08-29	14:00:00	40.704422	-73.792854	0	Oversized Vehicle	Passing Too Closely	Bus	Station Wagon/Sport Utility Vehicle
1	2020-08-29	12:29:00	40.861862	-73.912820	2	Pavement Slippery	View Obstructed/Limited	Pick-up Truck	Station Wagon/Sport Utility Vehicle
3	2020-08-29	19:00:00	40.839680	-73.929276	1	Following Too Closely	Following Too Closely	Sedan	Station Wagon/Sport Utility Vehicle
4	2020-08-29	05:40:00	40.858190	-73.884350	0	Other Vehicular	Passing Too Closely	Sedan	Sedan
9	2020-08-29	15:00:00	40.669518	-73.911934	0	Other Vehicular	Other Vehicular	Station Wagon/Sport Utility Vehicle	Station Wagon/Sport Utility Vehicle
...	...	...	...	...	...	...	...	...	...
49528	2020-02-13	08:25:00	40.665230	-73.931465	8	Driver Inattention/Distraction	Driver Inattention/Distraction	Sedan	Station Wagon/Sport Utility Vehicle
49529	2020-02-14	08:40:00	40.854744	-73.923510	8	Driver Inattention/Distraction	Driver Inattention/Distraction	Sedan	Sedan
49530	2020-02-14	08:40:00	40.854744	-73.923510	8	Driver Inattention/Distraction	Driver Inattention/Distraction	Sedan	Sedan
49533	2020-02-13	08:25:00	40.665230	-73.931465	8	Driver Inattention/Distraction	Driver Inattention/Distraction	Sedan	Station Wagon/Sport Utility Vehicle
49534	2020-02-14	08:40:00	40.854744	-73.923510	8	Driver Inattention/Distraction	Driver Inattention/Distraction	Sedan	Sedan

24536 rows × 9 columns

Counter of 'NUMBER OF PERSONS INJURED':
{7: 2763, 2: 2753, 3: 2748, 4: 2748, 6: 2725, 0: 2722, 1: 2706, 5: 2699, 8: 2672}

DATA VISUALISATION

Data Visualisation Contents

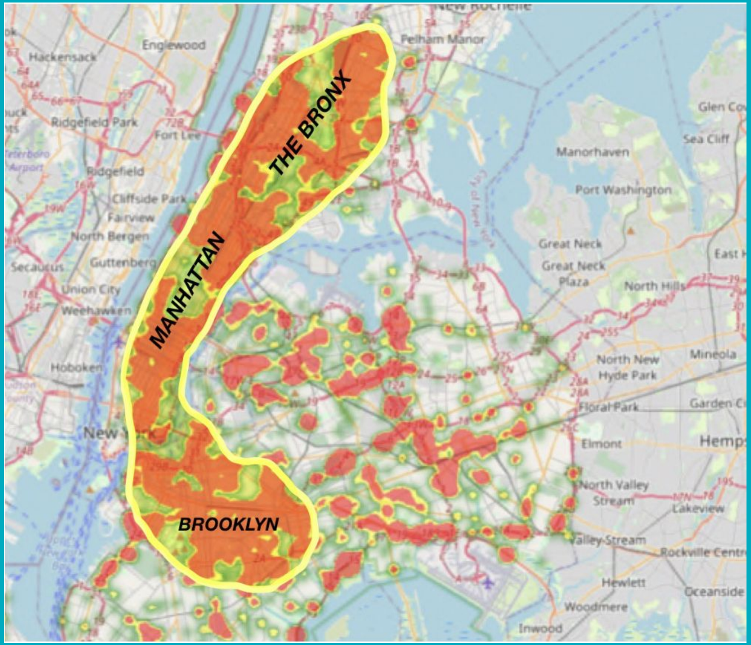
01 LOCATION COORDINATES

02 CRASH DATE AND TIME

03 CONTRIBUTING FACTOR VEHICLE

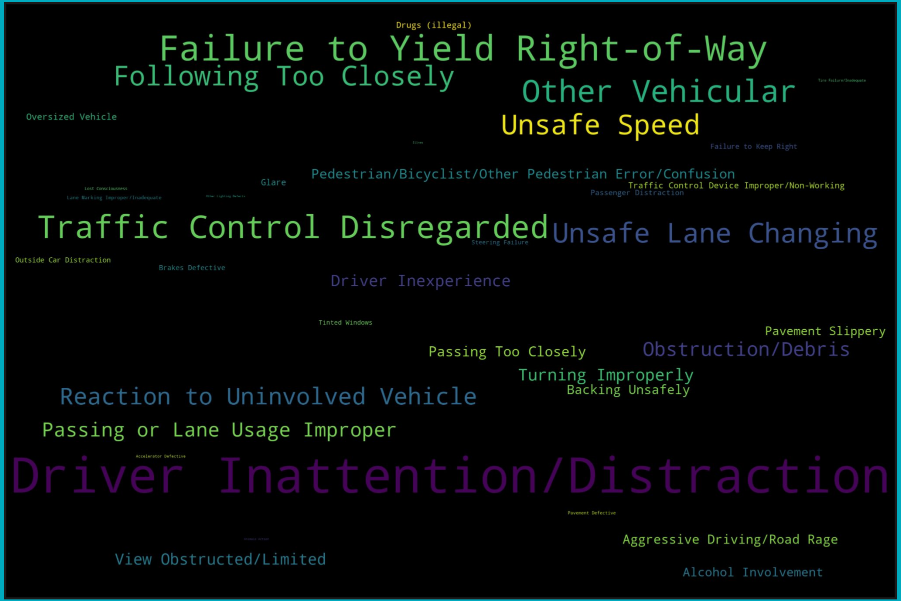
04 VEHICLE TYPE CODE

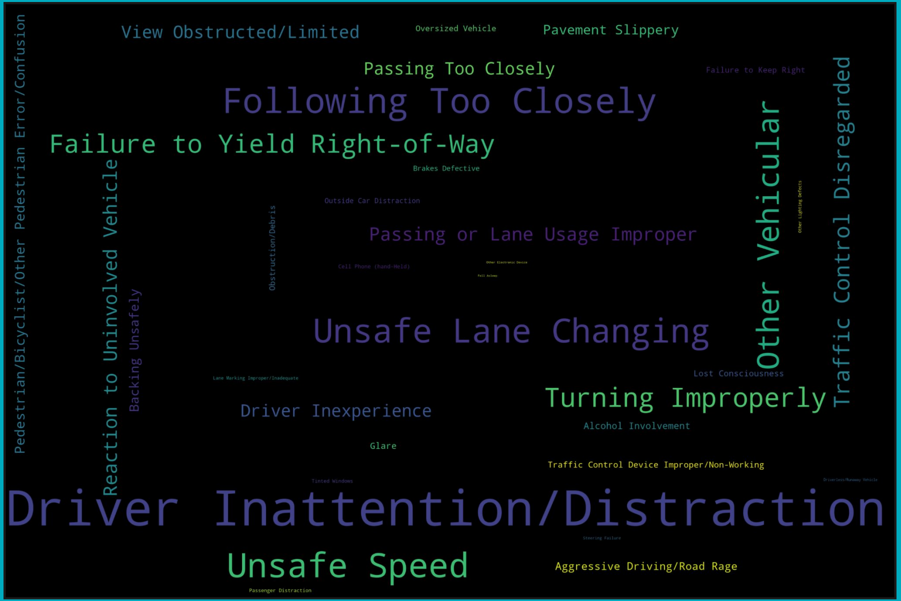
LOCATION COORDINATES Visualisation



The map indicates that most of the traffic accidents in New York City occurs in **Brooklyn**, **Manhattan**, and **The Bronx.**

CONTRIBUTING VEHICLE TYPE Visualisation

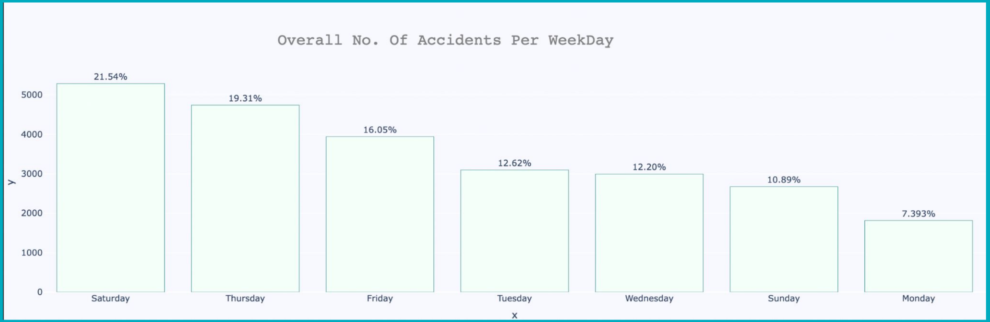


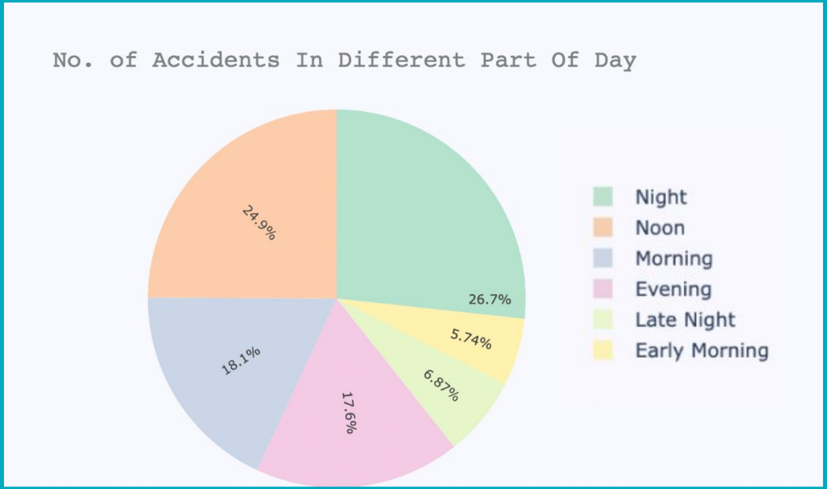


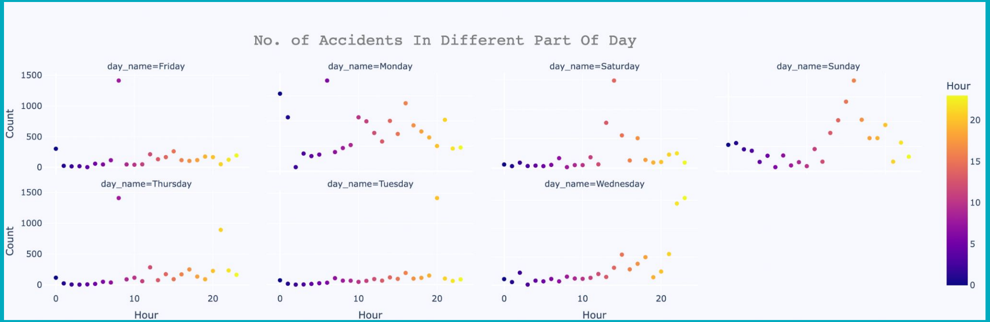
CONTRIBUTING VEHICLE TYPE 1 Word Cloud shows that most of the accidents are caused by Driver Inattention/Distraction

CONTRIBUTING VEHICLE TYPE 2 Word Cloud shows that most of the accidents are caused by Driver Inattention/Distraction

CRASH DATE and CRASH TIME Visualisation

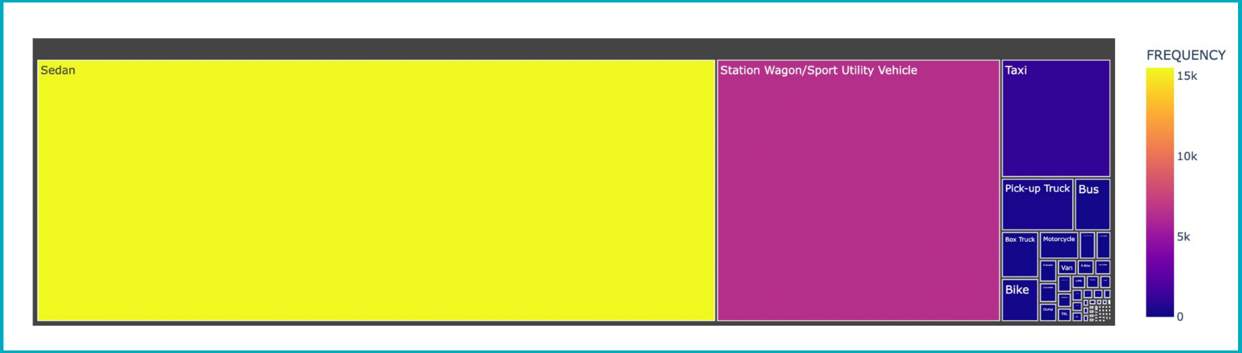




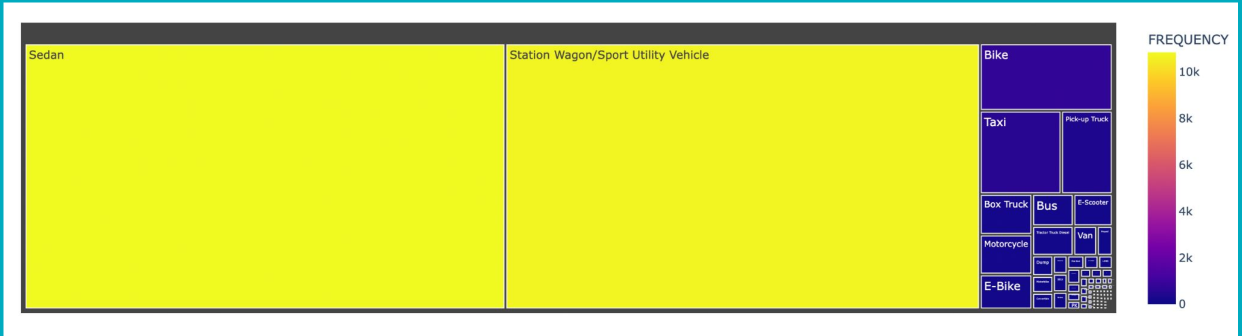


The charts indicate that most of the traffic accidents are at evening hours.

VEHICLE TYPE CODE Visualisation



The treemap shows that Sedan is the most common type of vehicle in the accidents for VEHICLE TYPE CODE 1

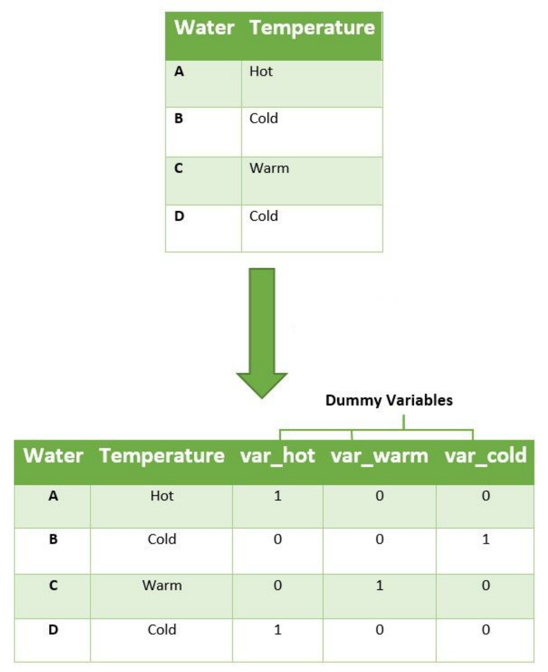


The treemap shows that Sedan is also the most common type of vehicle in the accidents for VEHICLE TYPE CODE 2

DATA MODELLING & MACHINE LEARNING

ONE-HOT ENCODING

PD.GET_DUMMIES

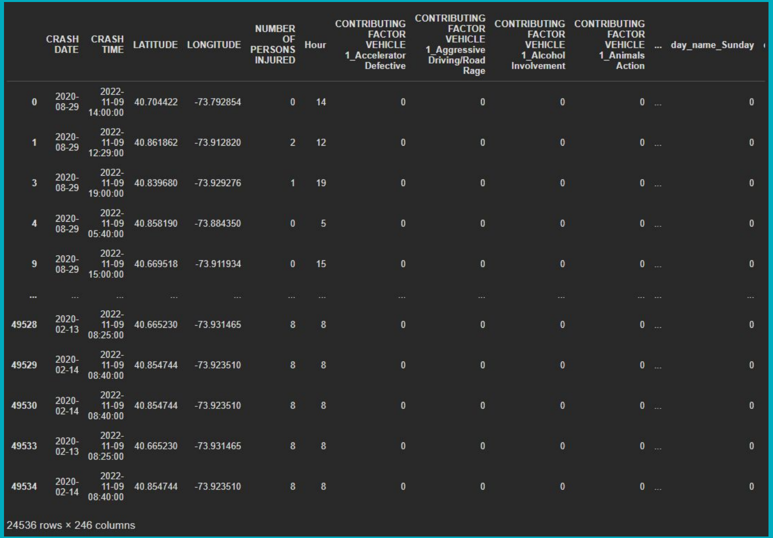


Water	Temperature
A	Hot
B	Cold
C	Warm
D	Cold

Dummy Variables

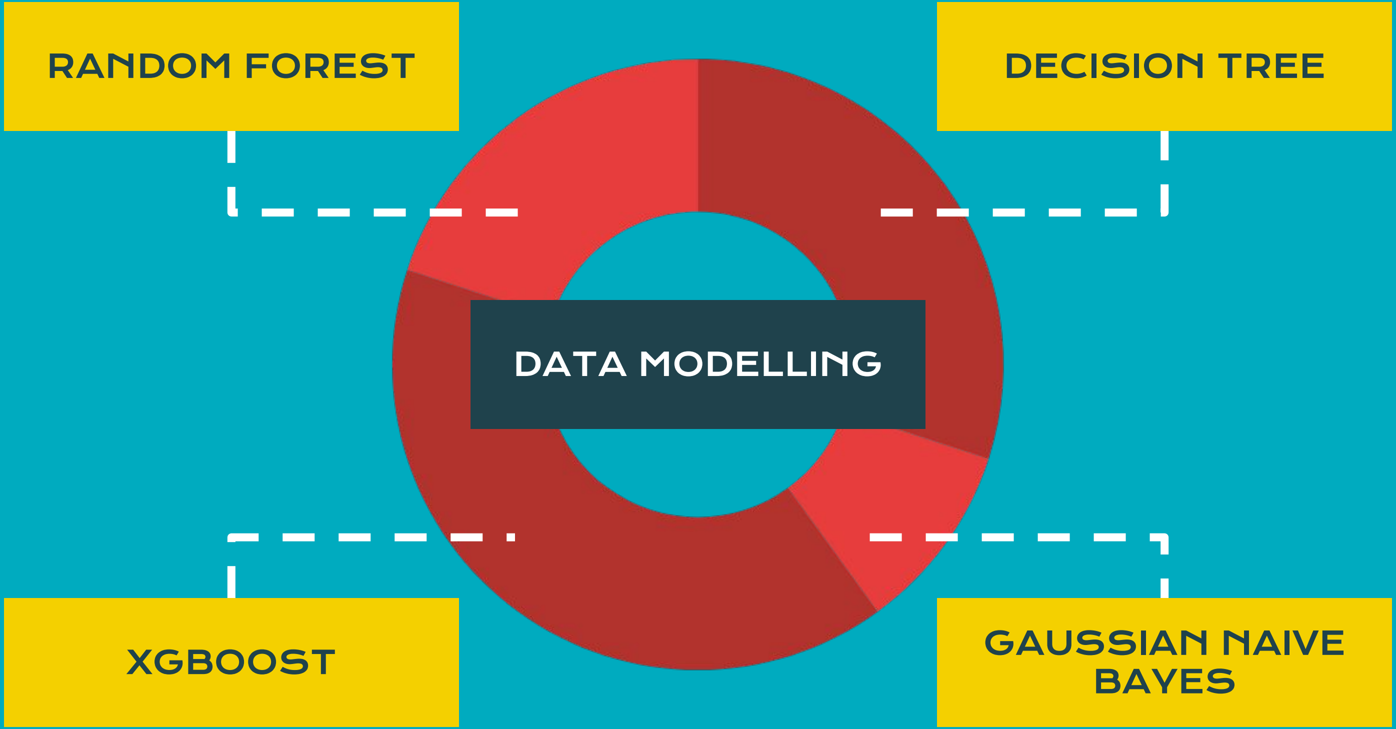
Water	Temperature	var_hot	var_warm	var_cold
A	Hot	1	0	0
B	Cold	0	0	1
C	Warm	0	1	0
D	Cold	1	0	0

Encoded DataFrame



	CRASH DATE	CRASH TIME	LATITUDE	LONGITUDE	NUMBER OF PERSONS INJURED	Hour	CONTRIBUTING FACTOR VEHICLE 1_Accelerator Defective	CONTRIBUTING FACTOR VEHICLE 1_Aggressive Driving/Road Rage	CONTRIBUTING FACTOR VEHICLE 1_Alcohol Involvement	CONTRIBUTING FACTOR VEHICLE 1_Animals Action	...	day_name_Sunday
0	2020-08-29	2022-11-09 14:00:00	40.704422	-73.792854	0	14	0	0	0	0	...	0
1	2020-08-29	2022-11-09 12:29:00	40.861862	-73.912820	2	12	0	0	0	0	...	0
3	2020-08-29	2022-11-09 19:00:00	40.839680	-73.929276	1	19	0	0	0	0	...	0
4	2020-08-29	2022-11-09 05:40:00	40.858190	-73.884350	0	5	0	0	0	0	...	0
9	2020-08-29	2022-11-09 15:00:00	40.669518	-73.911934	0	15	0	0	0	0	...	0
...	...	...	...	...	...	...	...	...	...	...	...	...
49528	2020-02-13	2022-11-09 08:25:00	40.665230	-73.931465	8	8	0	0	0	0	...	0
49529	2020-02-14	2022-11-09 08:40:00	40.854744	-73.923510	8	8	0	0	0	0	...	0
49530	2020-02-14	2022-11-09 08:40:00	40.854744	-73.923510	8	8	0	0	0	0	...	0
49533	2020-02-13	2022-11-09 08:25:00	40.665230	-73.931465	8	8	0	0	0	0	...	0
49534	2020-02-14	2022-11-09 08:40:00	40.854744	-73.923510	8	8	0	0	0	0	...	0

24536 rows × 246 columns



DATA MODELLING

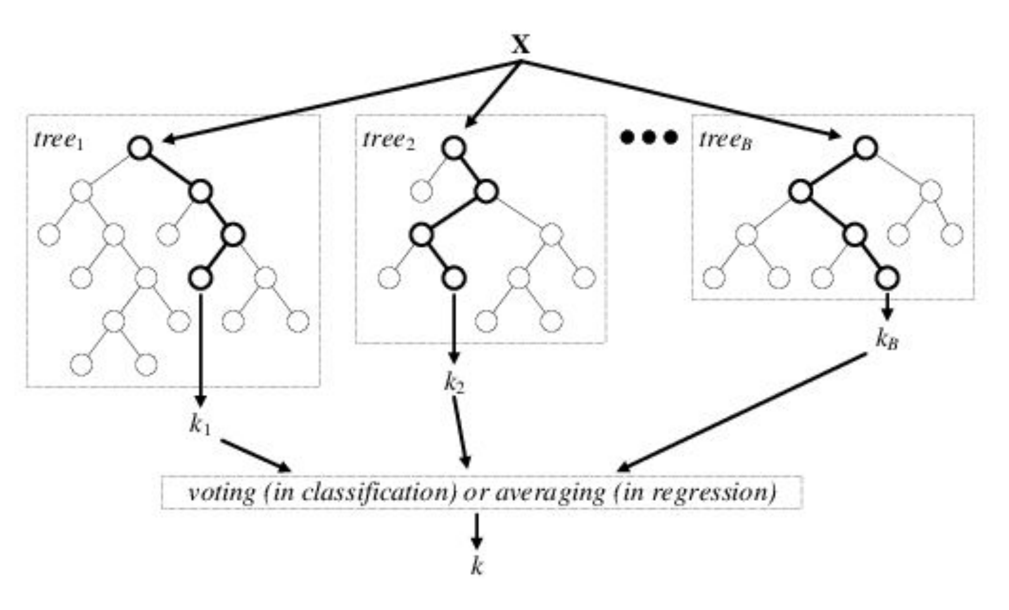
RANDOM FOREST

DECISION TREE

XGBOOST

GAUSSIAN NAIVE BAYES

RANDOM FOREST CLASSIFICATION



BOOTSTRAP AGGREGATION

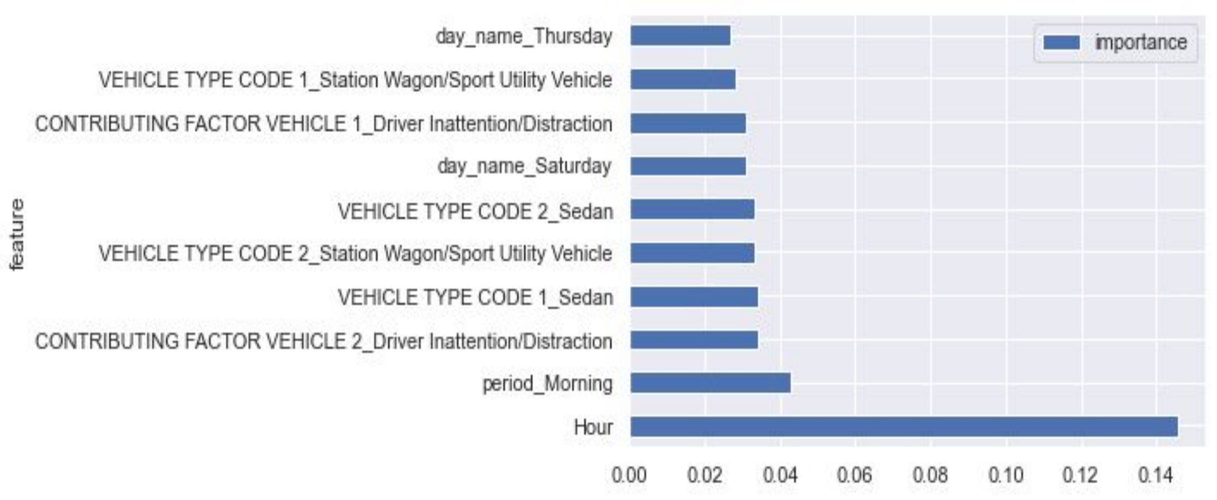
Bagging is an ensemble algorithm that fits multiple models on different subsets of a training dataset, then combines the predictions from all models.

N_ESTIMATORS **100**

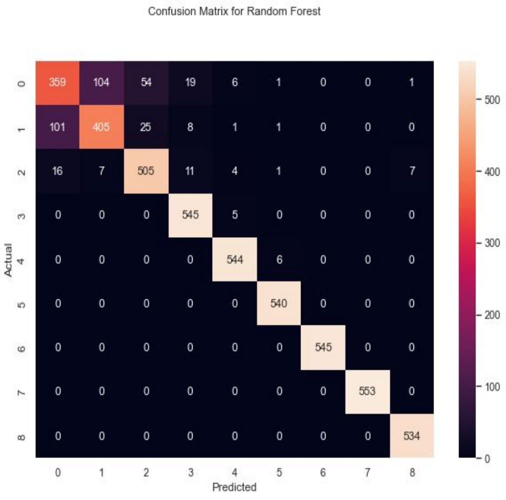
We use 100 different decision trees for optimality.

RANDOM FOREST CLASSIFICATION

FEATURE IMPORTANCES



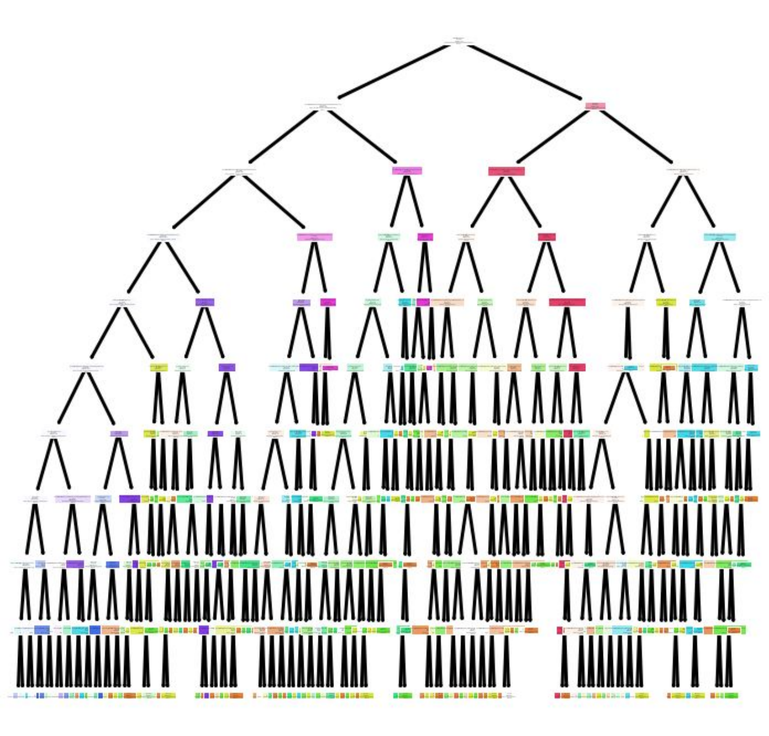
CONFUSION MATRIX

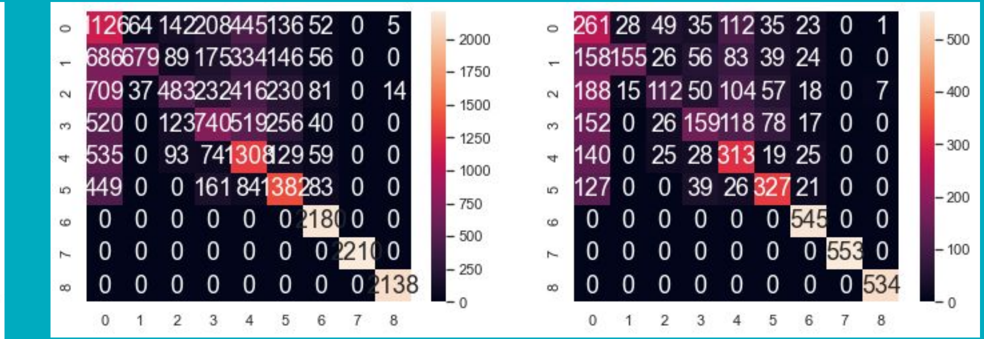


Actual \ Predicted	0	1	2	3	4	5	6	7	8
0	359	104	54	19	6	1	0	0	1
1	101	405	25	8	1	1	0	0	0
2	16	7	505	11	4	1	0	0	7
3	0	0	0	545	5	0	0	0	0
4	0	0	0	0	544	6	0	0	0
5	0	0	0	0	0	540	0	0	0
6	0	0	0	0	0	0	545	0	0
7	0	0	0	0	0	0	0	553	0
8	0	0	0	0	0	0	0	0	534

Accuracy	:	*0.9229828850855746*
Balanced Accuracy Score	:	*0.9227839528287662*
Precision	:	*0.9229828850855746*
Recall	:	*0.9229828850855746*

DECISION TREE CLASSIFICATION

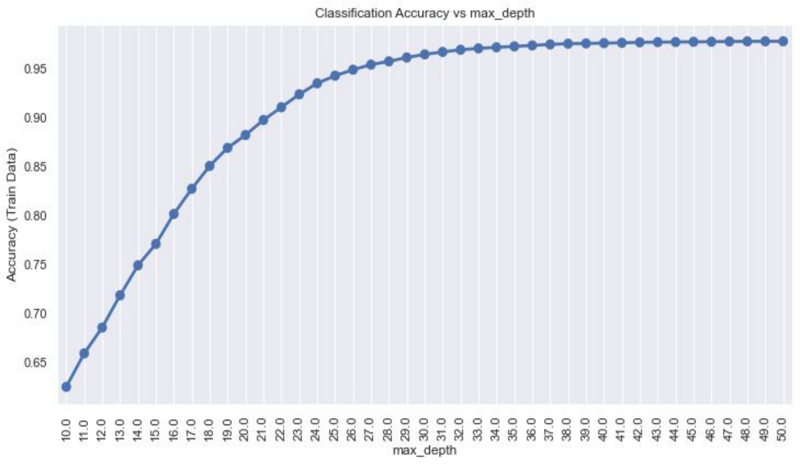


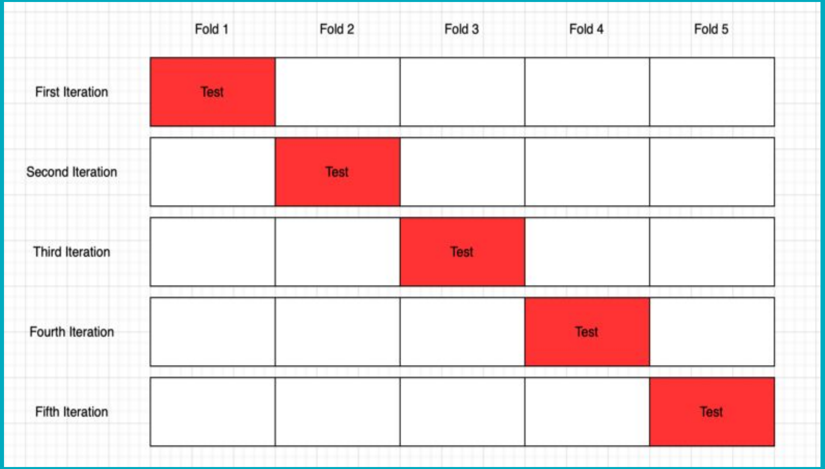


ACCURACY FOR TRAIN	0.6239046260
ACCURACY FOR TEST	0.6028932355

We need to optimise/tune the parameters!

Finding the Optimal Max_Depth using 5-Fold Cross Validation





	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
First Iteration	Test				
Second Iteration		Test			
Third Iteration			Test		
Fourth Iteration				Test	
Fifth Iteration					Test

GridSearchCV

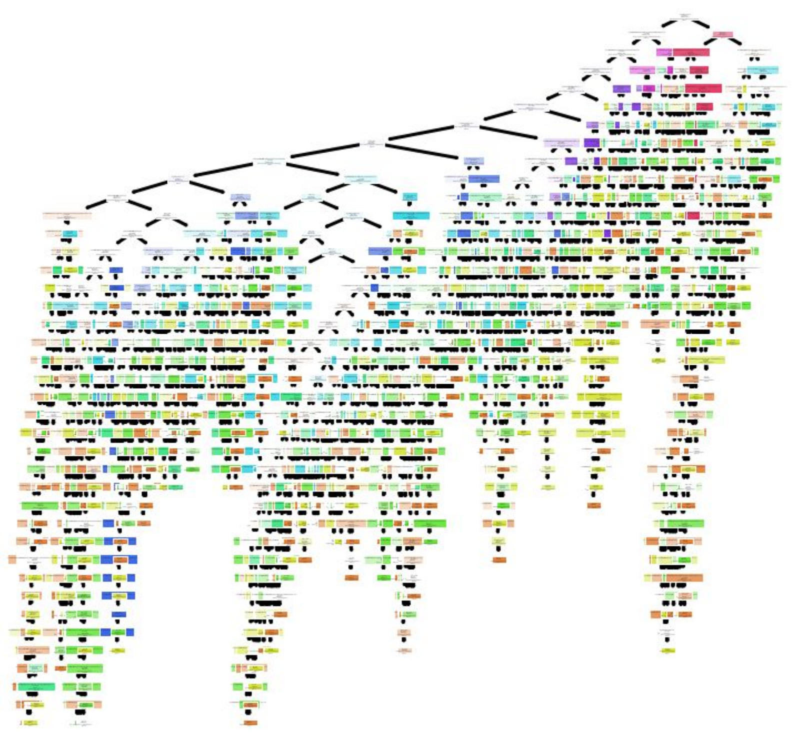
Grid search is a process that searches exhaustively through a manually specified subset of the hyperparameter space of the targeted algorithm.

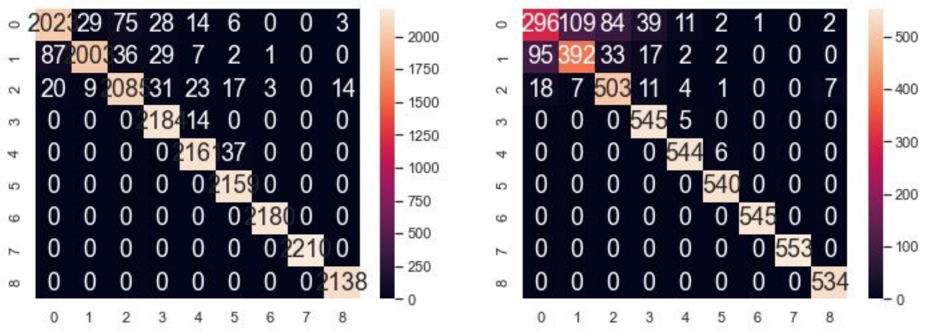
StratifiedKFold

5

KFold divides all the samples in groups of samples. The fold left out is used for test. We use 5-fold cross validation.

DECISION TREE CLASSIFICATION





	0	1	2	3	4	5	6	7	8
0	2023	29	75	28	14	6	0	0	3
1	87	2003	36	29	7	2	1	0	0
2	20	9	2085	31	23	17	3	0	14
3	0	0	0	2184	14	0	0	0	0
4	0	0	0	0	2161	37	0	0	0
5	0	0	0	0	0	2159	0	0	0
6	0	0	0	0	0	0	2180	0	0
7	0	0	0	0	0	0	0	2210	0
8	0	0	0	0	0	0	0	0	2138

	0	1	2	3	4	5	6	7	8
0	296	109	84	39	11	2	1	0	2
1	95	392	33	17	2	2	0	0	0
2	18	7	503	11	4	1	0	0	7
3	0	0	0	545	5	0	0	0	0
4	0	0	0	0	544	6	0	0	0
5	0	0	0	0	0	540	0	0	0
6	0	0	0	0	0	0	545	0	0
7	0	0	0	0	0	0	0	553	0
8	0	0	0	0	0	0	0	0	534

ACCURACY FOR TRAIN	0.97529040147
ACCURACY FOR TEST	0.9070904645

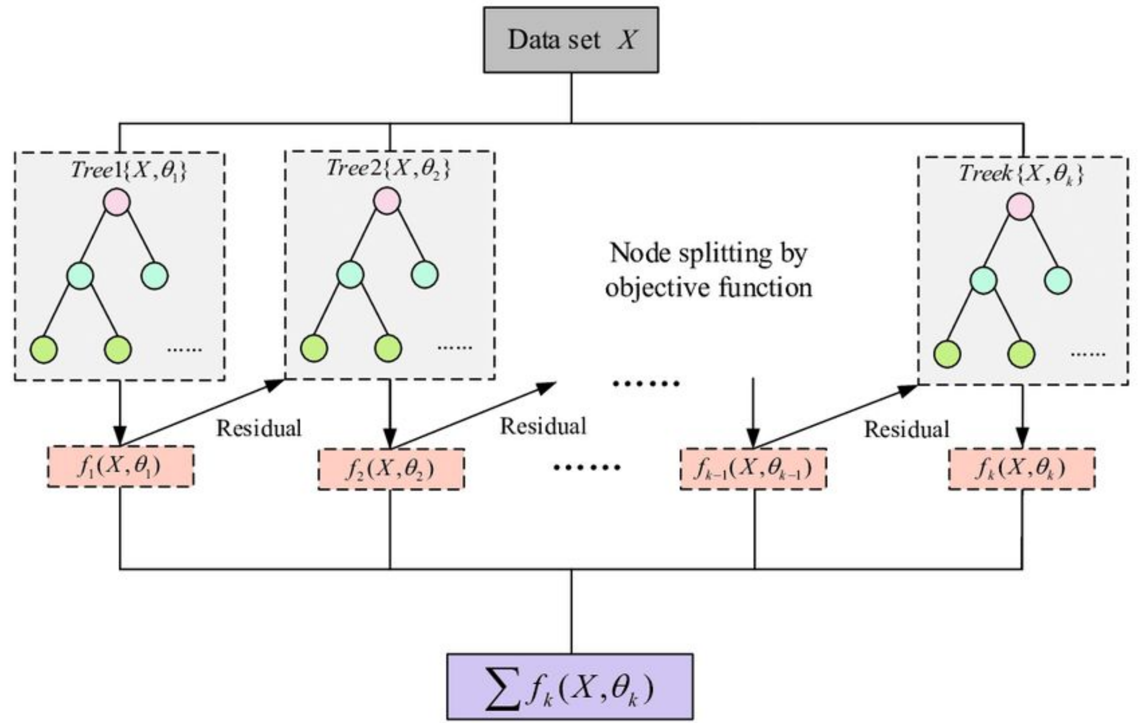
The most optimal max_depth used is 39.

XGBOOST CLASSIFICATION

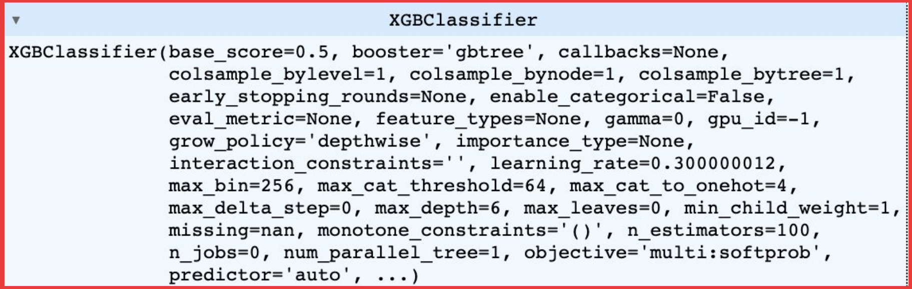
XGBoost, which stands for Extreme Gradient Boosting, is a scalable, distributed gradient-boosted decision tree (GBDT) machine learning library.

ONE TREE CREATED AT A TIME

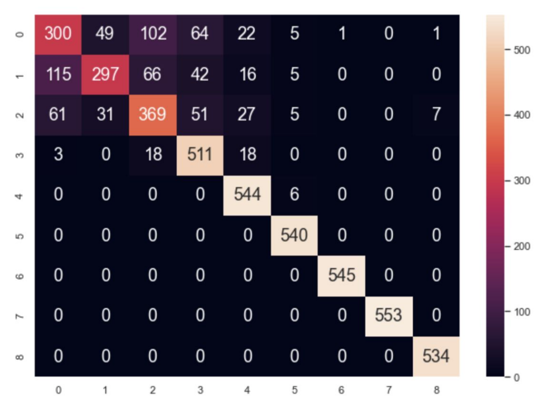
ITERATION TO REDUCE ERRORS



XGBOOST CLASSIFICATION



```
XGBClassifier(base_score=0.5, booster='gbtree', callbacks=None,
              colsample_bylevel=1, colsample_bynode=1, colsample_bytree=1,
              early_stopping_rounds=None, enable_categorical=False,
              eval_metric=None, feature_types=None, gamma=0, gpu_id=-1,
              grow_policy='depthwise', importance_type=None,
              interaction_constraints='', learning_rate=0.300000012,
              max_bin=256, max_cat_threshold=64, max_cat_to_onehot=4,
              max_delta_step=0, max_depth=6, max_leaves=0, min_child_weight=1,
              missing=nan, monotone_constraints='()', n_estimators=100,
              n_jobs=0, num_parallel_tree=1, objective='multi:softprob',
              predictor='auto', ...)
```



	0	1	2	3	4	5	6	7	8
0	300	49	102	64	22	5	1	0	1
1	115	297	66	42	16	5	0	0	0
2	61	31	369	51	27	5	0	0	7
3	3	0	18	511	18	0	0	0	0
4	0	0	0	0	544	6	0	0	0
5	0	0	0	0	0	540	0	0	0
6	0	0	0	0	0	0	545	0	0
7	0	0	0	0	0	0	0	553	0
8	0	0	0	0	0	0	0	0	534

A

N_ESTIMATORS

Based on the classification, the model requires to run 100 times to learn the data.

B

MAX_DEPTH

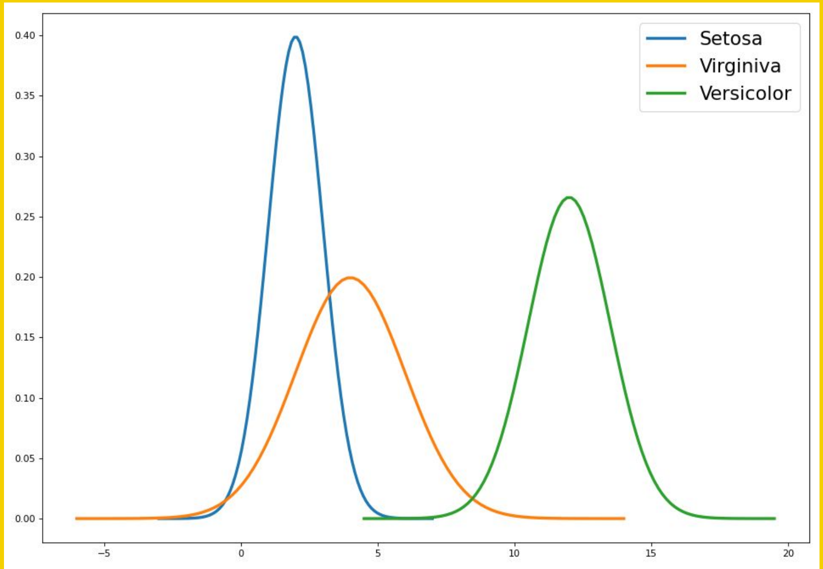
Maxed depth we chose is as default, 6, to prevent overfit of data

C

TEST ACCURACY

Accuracy of model turned out positive with approximately 85.4% success rate

GAUSSIAN NAIVE BAYES

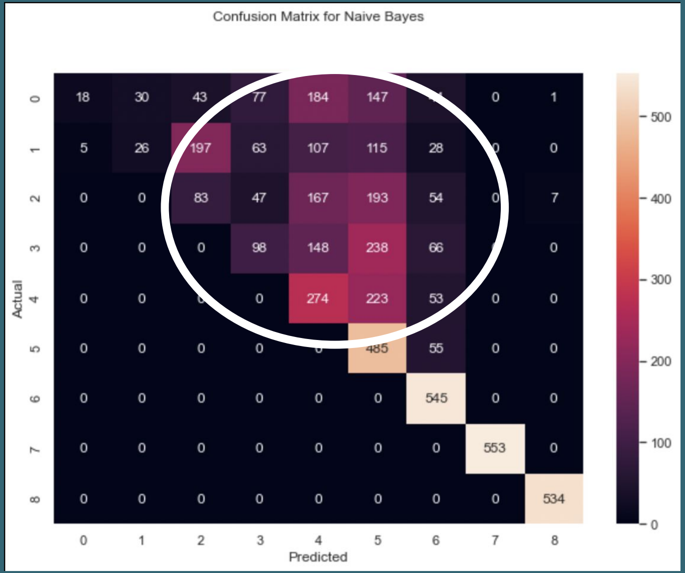


Naive Bayes is a classification algorithm for binary (two-class) and multi-class classification problems. The technique is easiest to understand when described using binary or categorical input values.

Assumes each class follows a Gaussian Distribution

Assumes that the features are independent

GAUSSIAN NAIVE BAYES



Confusion Matrix for Naive Bayes

Actual \ Predicted	0	1	2	3	4	5	6	7	8
0	18	30	43	77	184	147	[illegible]	0	1
1	5	26	197	63	107	115	28	0	0
2	0	0	83	47	167	193	54	0	7
3	0	0	0	98	148	238	66	0	0
4	0	0	0	0	274	223	53	0	0
5	0	0	0	0	0	485	55	0	0
6	0	0	0	0	0	0	545	0	0
7	0	0	0	0	0	0	0	553	0
8	0	0	0	0	0	0	0	0	534

BALANCED ACCURACY

0.5340327087

WHY IS IT SO LOW?

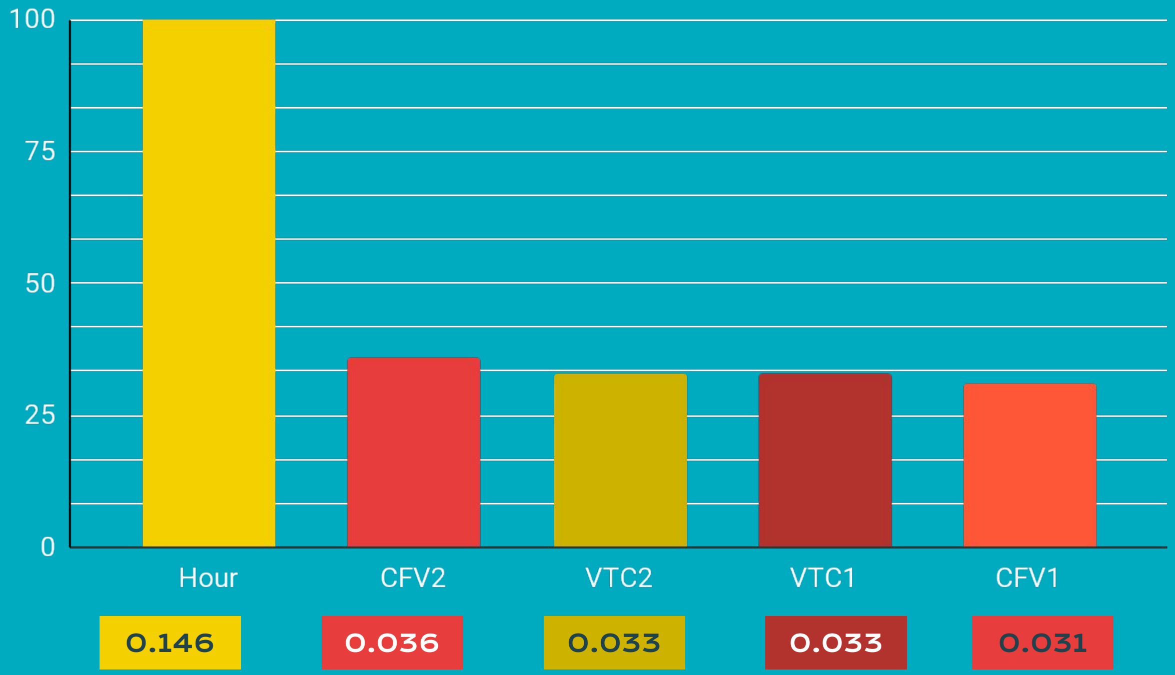
The Gaussian Naive Bayes assumes the dataset to be distributed normally.

In which our case, it is not.

CLOSING

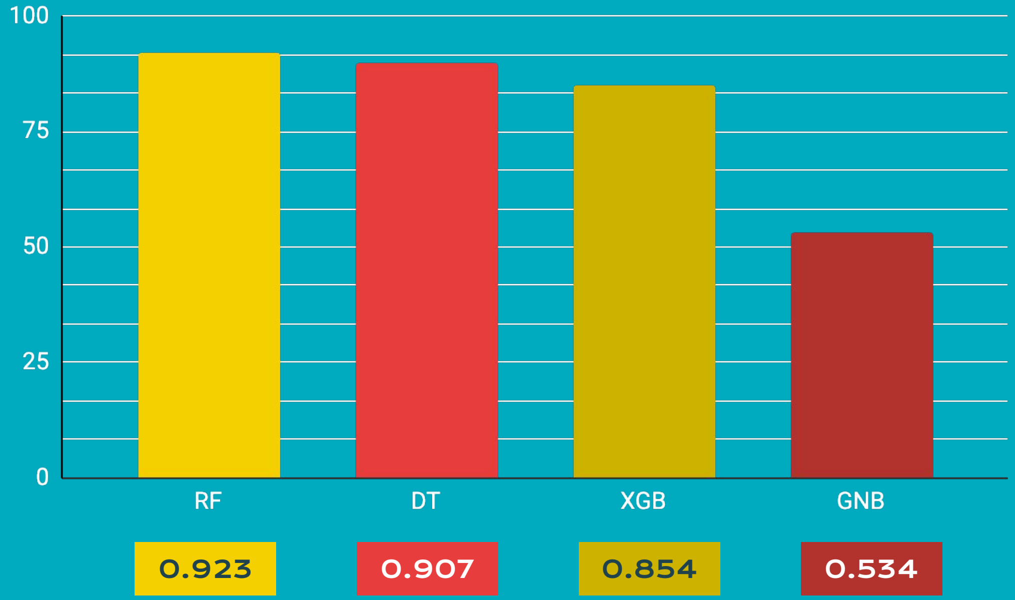
CONCLUSION

Top 5 Most Important Features Predicting Number of Persons Injured

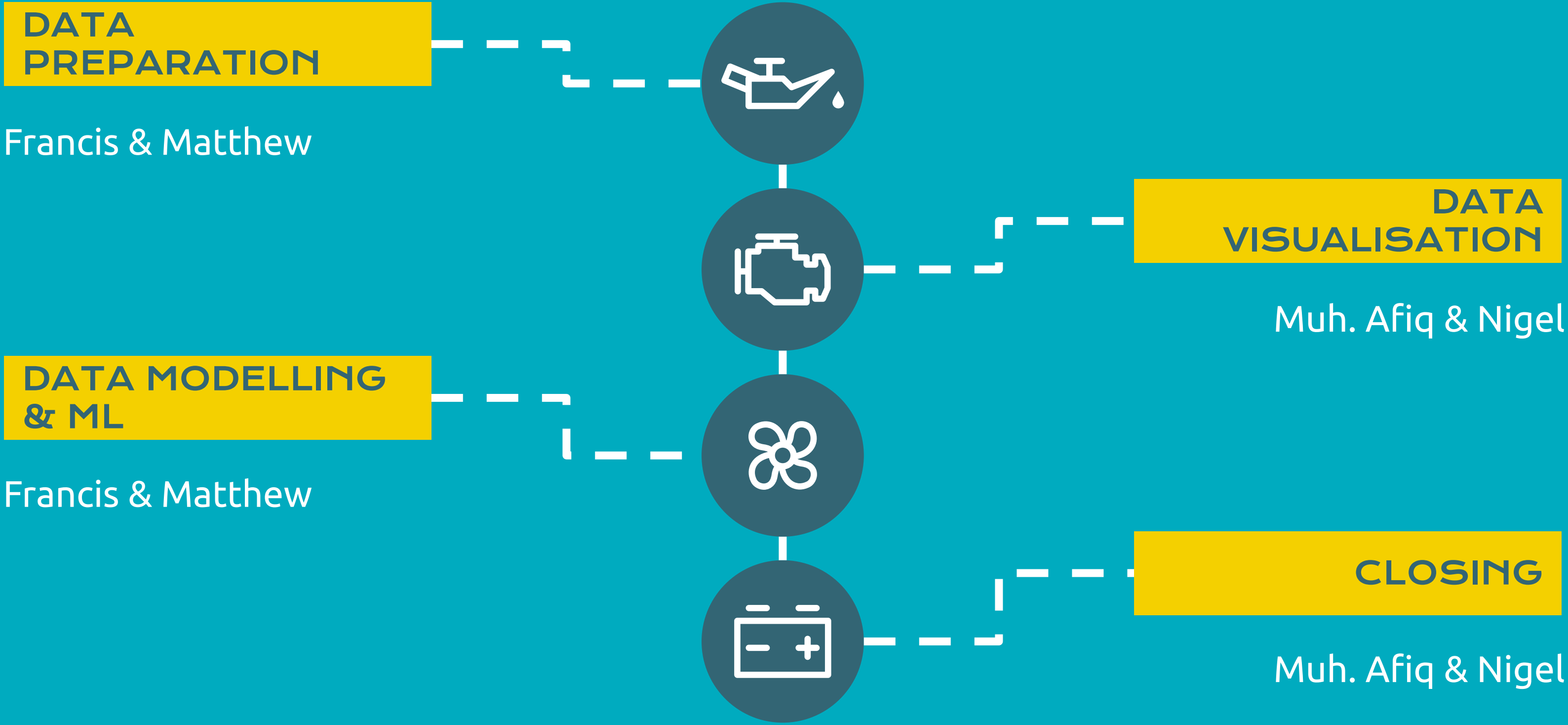


CONCLUSION

Most Suitable Machine Learning Models



Work Contribution



DATA PREPARATION

Francis & Matthew

DATA VISUALISATION

Muh. Afiq & Nigel

DATA MODELLING & ML

Francis & Matthew

CLOSING

Muh. Afiq & Nigel

THANK YOU

References

- https://www.sciencedirect.com/topics/mathematics/grid-search
- https://towardsdatascience.com/ensemble-methods-bagging-boosting-and-stacking-c9214a10a205
- https://towardsdatascience.com/gaussian-naive-bayes-4d2895d139a
- https://www.nvidia.com/en-us/glossary/data-science/xgboost/#:~:text=XGBoost%2C%20which%20stands%20for%20Extreme,%2C%20classification%2C%20and%20ranking%20problems.
- https://xgboost.readthedocs.io/en/stable/parameter.html