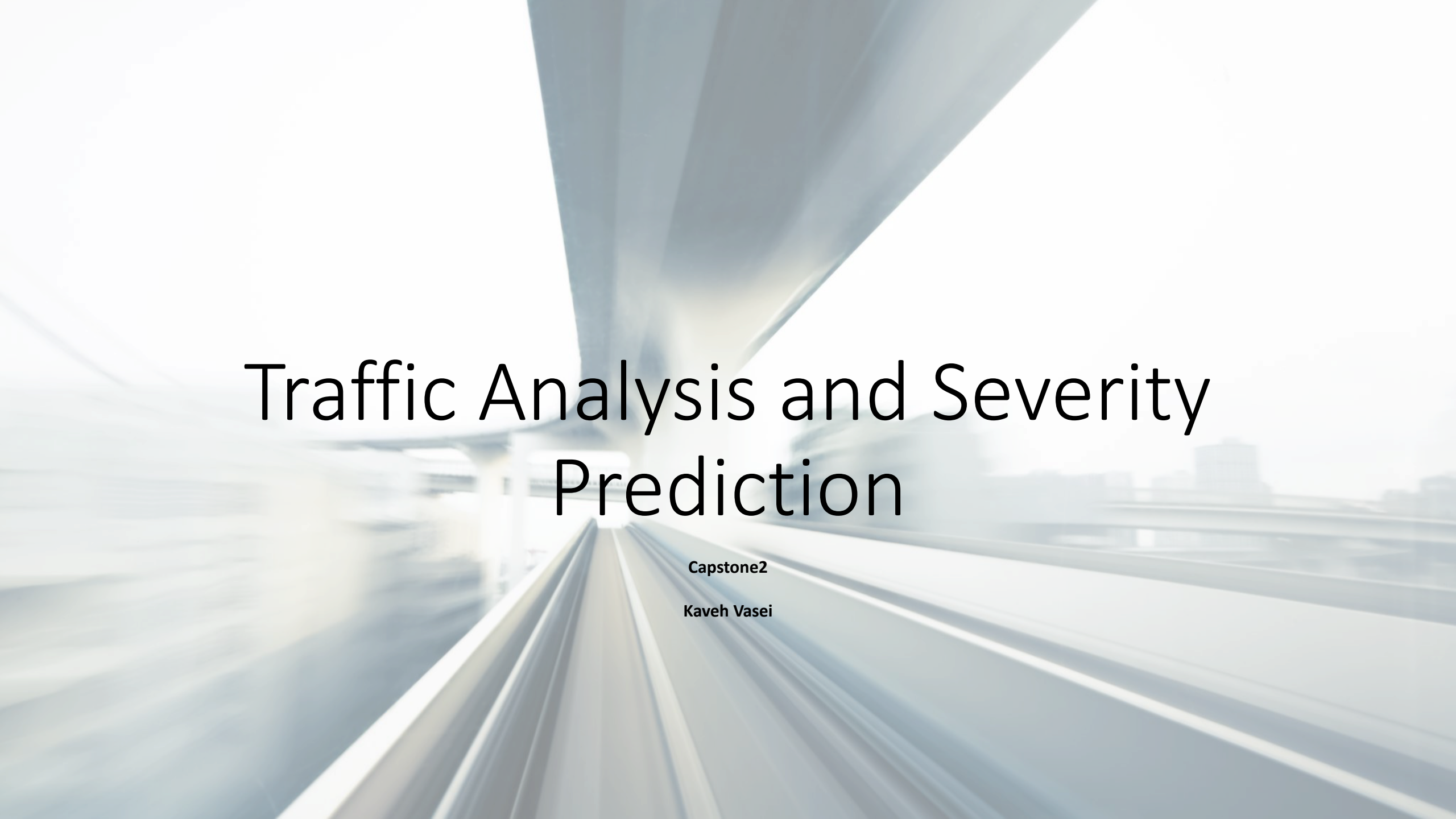
UK Road Safty :

Developing A Binary Classifier to Predict Accident Severity in UK

Kaveh Vasei

Springboard Cap2

Summer 2023



Traffic Analysis and Severity Prediction

Capstone2

Kaveh Vasei

Background

- The UK government collects and publishes (usually on an annual basis) detailed information about traffic accidents across the country. This information includes, but is not limited to, geographical locations, weather conditions, type of vehicles, number of casualties and vehicle maneuvers, making this a very interesting and comprehensive dataset for analysis and research.
- The full dataset for this project is available at : https://www.data.gov.uk/dataset/cb7ae6f0-4be6-4935-9277-47e5ce24a11f/road-safety-data

Importance of Analyzing Traffic Accidents

Accurately analyzing accidents can help governments to better the safety of their roads and highways.

Identifying high areas of accidents and high areas of accident severity can highlight areas of concern.

It can also be beneficial to insurance companies looking to change their rates in different areas.

Objective

- Develop machine learning model that can accurately classify accidents in UK as Major and Minor accidents.
- Accurately predict whether an accident will be classified as "minor" or "major".
- to identify high-risk areas, improve emergency response, and implement targeted interventions. (What features can cause serious accidents)
- Enhance road safety by making informed decisions and allocating resources effectively.

Major Accidents: Are the ones that was fatal or resulted on serious Hospitalization

Minor Accidents: Roadside medical help or just a quick emergency room visit

No fender Bender in Dataset (we always have at least one casualty)

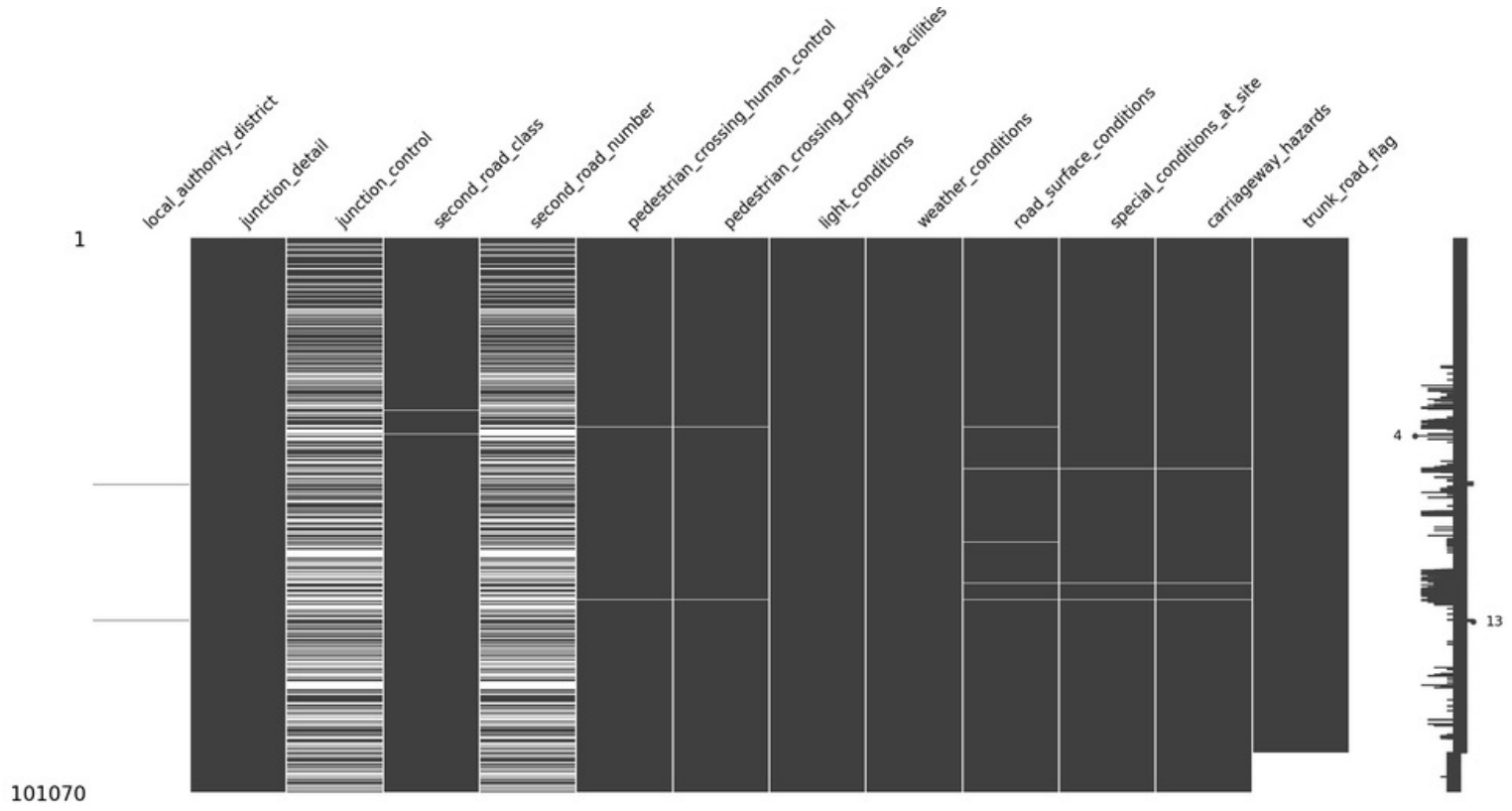
Data

- Accidents
  - main data set contains information about accident severity, weather, location, date, hour, day of week, road type…
- Vehicles
  - contains information about vehicle type, vehicle model, engine size, driver sex, driver age, car age, car status after accidents
- Casualty
  - contains information about casualty severity, age, sex social class, casualty type, pedestrian or car passenger…
- Data guide file
  - contains the text description of all variable code in the three files

Data Wrangling

```
card_acc = acc_df.nunique()
card_acc
```

```
accident_index                                101070
accident_year                                      1
accident_reference                            101070
longitude                                      99035
latitude                                       98036
police_force                                      44
accident_severity                                  3
number_of_vehicles                                13
number_of_casualties                              12
date                                             365
day_of_week                                        7
time                                            1440
local_authority_district                          14
local_authority_ons_district                     378
local_authority_highway                          206
first_road_class                                   6
first_road_number                               3099
road_type                                          6
speed_limit                                        6
junction_detail                                   10
second_road_class                                  7
pedestrian_crossing_human_control                  4
pedestrian_crossing_physical_facilities            7
light_conditions                                   5
weather_conditions                                 9
road_surface_conditions                            6
special_conditions_at_site                         9
carriageway_hazards                                7
urban_or_rural_area                                2
did_police_officer_attend_scene_of_accident        3
trunk_road_flag                                    3
lsoa_of_accident_location                      26581
dtype: int64
```

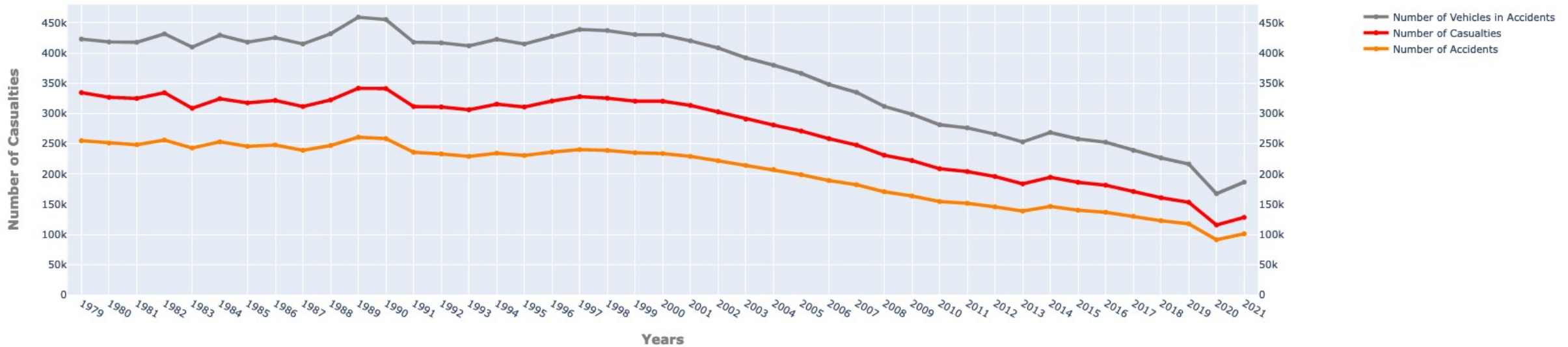


Mostly Nominal Categorical data type

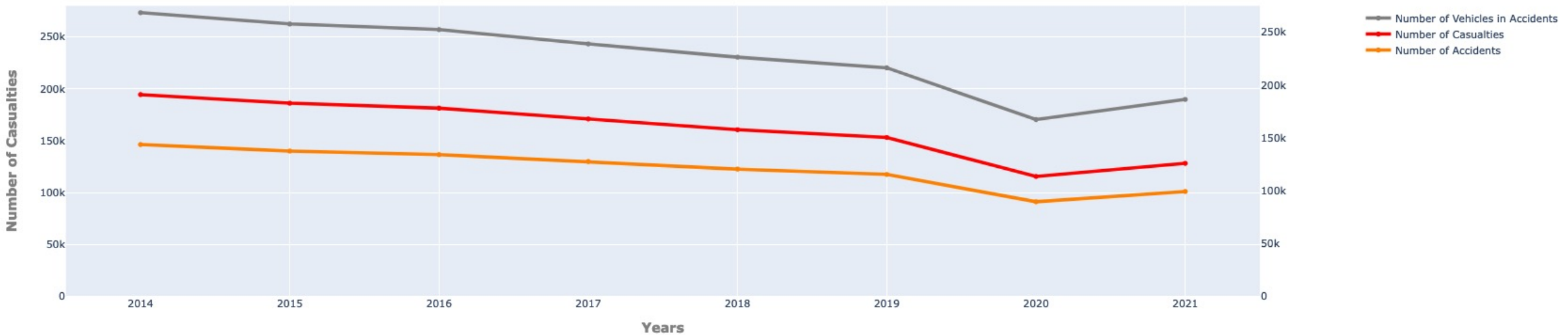
```
RangeIndex: 570199 entries, 0 to 570198
Data columns (total 28 columns):
 #   Column                           Non-Null Count   Dtype
---  ------                           --------------   -----
 0   accident_index                   570199 non-null  object
 1   accident_year                    570199 non-null  int64
 2   accident_reference               570199 non-null  object
 3   vehicle_reference                570199 non-null  int64
 4   vehicle_type                     570199 non-null  int64
 5   towing_and_articulation          570199 non-null  int64
 6   vehicle_manoeuvre                570199 non-null  int64
 7   vehicle_direction_from           570199 non-null  int64
 8   vehicle_direction_to             570199 non-null  int64
 9   vehicle_location_restricted_lane 570199 non-null  int64
 10  junction_location                570199 non-null  int64
 11  skidding_and_overturning         570199 non-null  int64
 12  hit_object_in_carriageway        570199 non-null  int64
 13  vehicle_leaving_carriageway      570199 non-null  int64
 14  hit_object_off_carriageway       570199 non-null  int64
 15  first_point_of_impact            570199 non-null  int64
 16  vehicle_left_hand_drive          570199 non-null  int64
 17  journey_purpose_of_driver        570199 non-null  int64
 18  sex_of_driver                    570199 non-null  int64
 19  age_of_driver                    570199 non-null  int64
 20  age_band_of_driver               570199 non-null  int64
 21  engine_capacity_cc               570199 non-null  int64
 22  propulsion_code                  570199 non-null  int64
 23  age_of_vehicle                   570199 non-null  int64
 24  generic_make_model               570199 non-null  object
 25  driver_imd_decile                570199 non-null  int64
 26  driver_home_area_type            570199 non-null  int64
 27  lsoa_of_driver                   570199 non-null  object
dtypes: int64(24), object(4)
memory usage: 121.8+ MB
```

```
 junction_location , with the following  11  features :
{0: 'Not at or within 20 metres of junction', 1: 'Approaching junction or waiting/parked at junction approach', 2: 'Cleared junction or waiting/parked at jun
[ 9  0  1  2  4  6  8  5  7  3 -1]
 0    231394
 1    126048
 8     86837
 2     31162
 9     30533
 6     23165
 4     16388
 5     12886
 3      9002
 7      1728
-1      1056
Name: junction_location, dtype: int64


 skidding_and_overturning , with the following  8  features :
{0: 'None', 1: 'Skidded', 2: 'Skidded and overturned', 3: 'Jackknifed', 4: 'Jackknifed and overturned', 5: 'Overturned', 9: 'unknown (self reported)', -1: 'D
[ 9  0  2  1  5 -1  4  3]
 0    470132
 9     40632
 1     33353
 5     13818
 2      9383
-1      2662
 3       148
 4        71
Name: skidding_and_overturning, dtype: int64


 hit_object_in_carriageway , with the following  14  features :
{0: 'None', 1: 'Previous accident', 2: 'Road works', 4: 'Parked vehicle', 5: 'Bridge (roof)', 6: 'Bridge (side)', 7: 'Bollard or refuge', 8: 'Open door of ve
[99  0 11  4 12 10  7  9  8  6 -1  2  1  5]
 0     499819
 99     39896
 4      10672
 10      8217
 7       3017
 11      2661
-1       2557
```

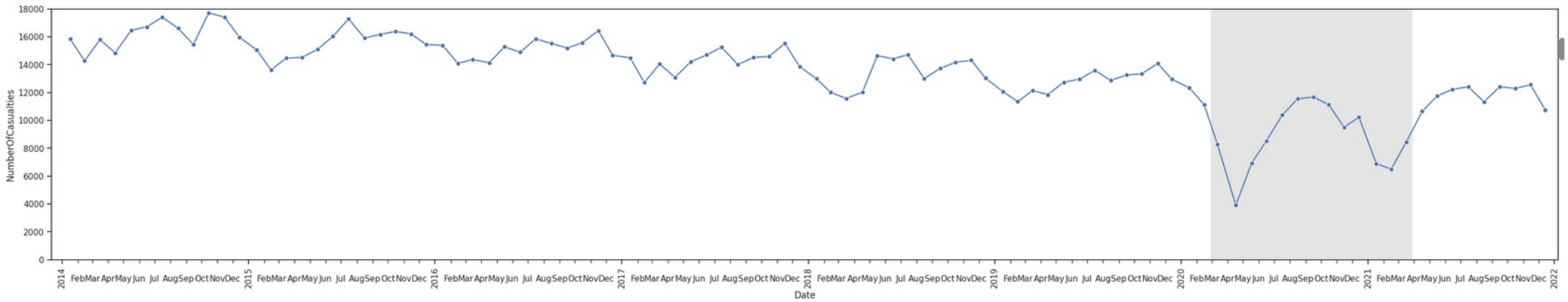
Records of Road Accidents (Casualties per year) in UK between 1979 - 2021

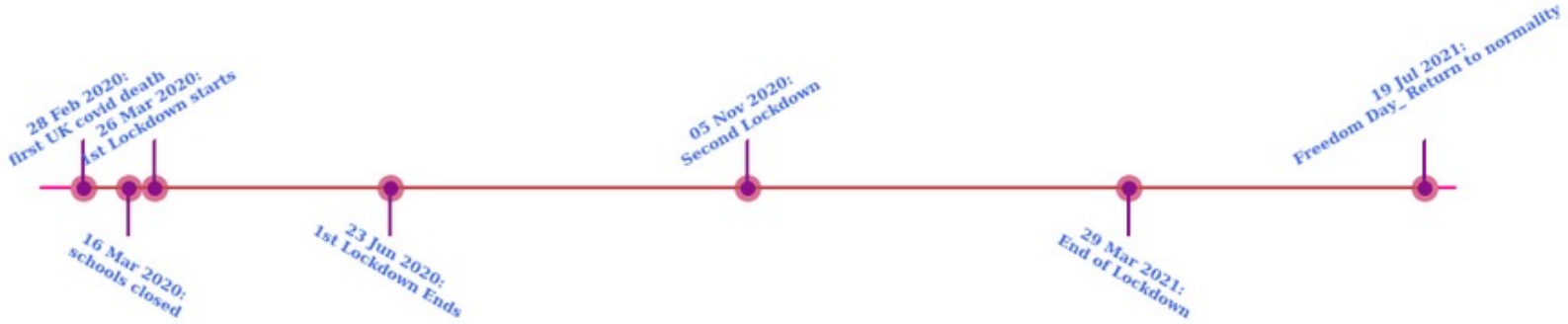


Records of Road Accidents (Casualties per year) in UK between 2014 - 2021

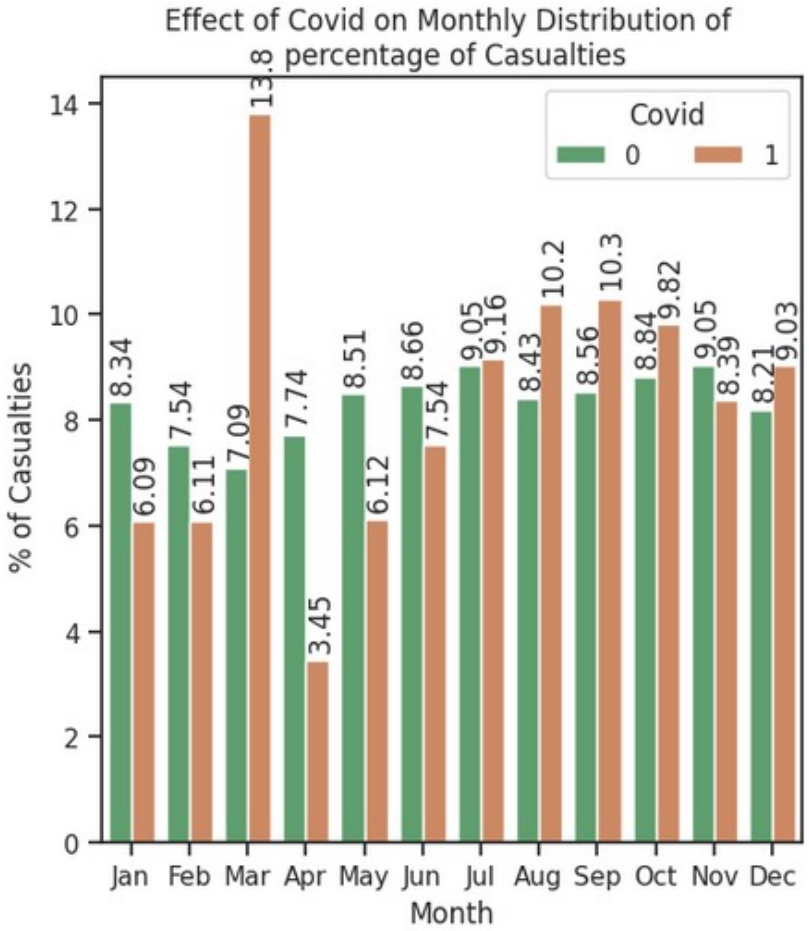


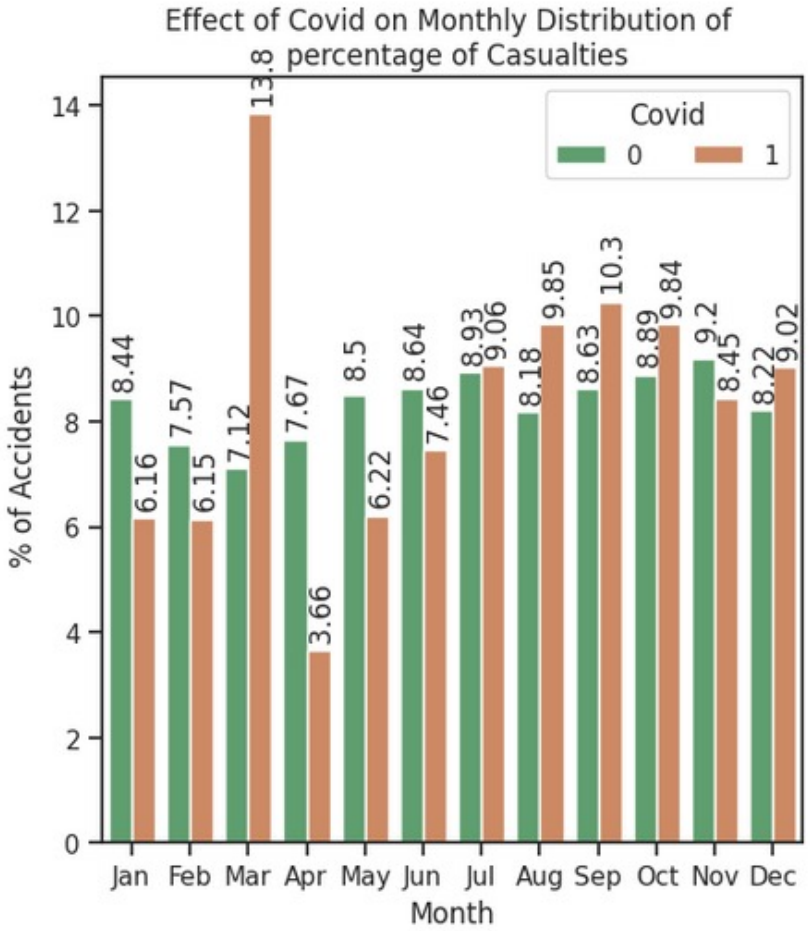
Covid


NumberOfCasualties
Date
18000
16000
14000
12000
10000
8000
6000
4000
2000
0
2014 Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
2015 Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
2016 Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
2017 Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
2018 Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
2019 Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
2020 Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
2021 Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
2022

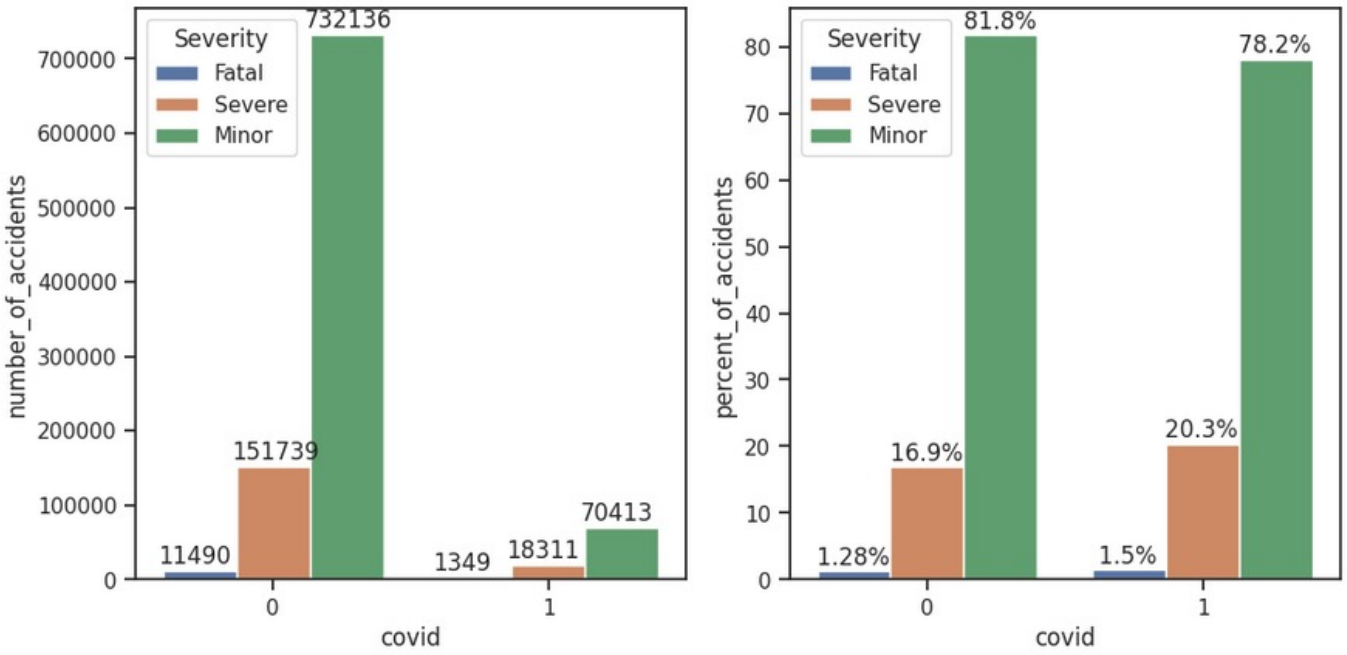

28 Feb 2020:
first UK covid death
26 Mar 2020:
1st Lockdown starts
16 Mar 2020:
schools closed
23 Jun 2020:
1st Lockdown Ends
05 Nov 2020:
Second Lockdown
29 Mar 2021:
End of Lockdown
19 Jul 2021:
Freedom Day_ Return to normality

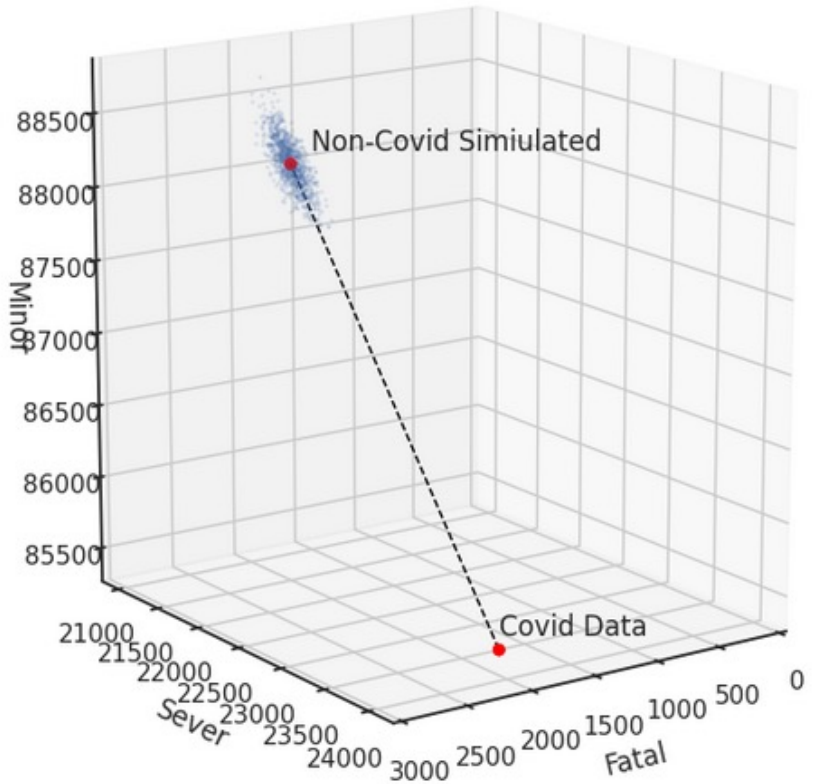
EDR


Effect of Covid on Monthly Distribution of percentage of Casualties
Covid
0
1
% of Casualties
Month
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
8.34 6.09
7.54 6.11
7.09 13.8
7.74 3.45
8.51 6.12
8.66 7.54
9.05 9.16
8.43 10.2
8.56 10.3
8.84 9.82
9.05 8.39
8.21 9.03

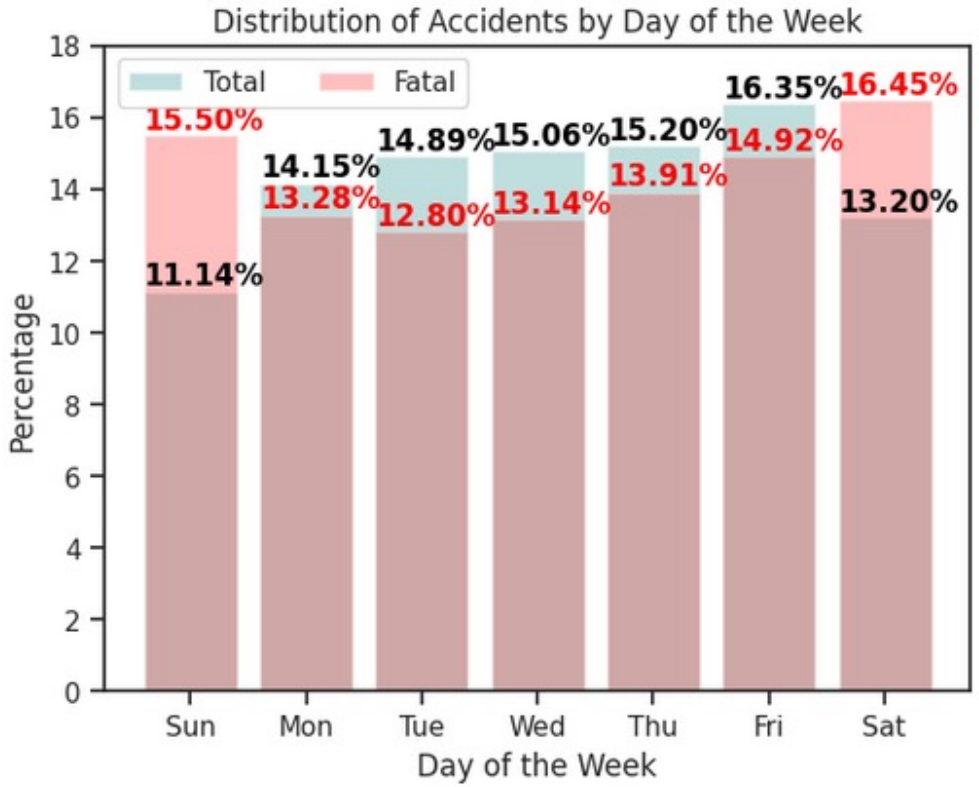

Effect of Covid on Monthly Distribution of percentage of Casualties
Covid
0
1
% of Accidents
Month
Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
8.44 6.16
7.57 6.15
7.12 13.8
7.67 3.66
8.5 6.22
8.64 7.46
8.93 9.06
8.18 9.85
8.63 10.3
8.89 9.84
9.2 8.45
8.22 9.02

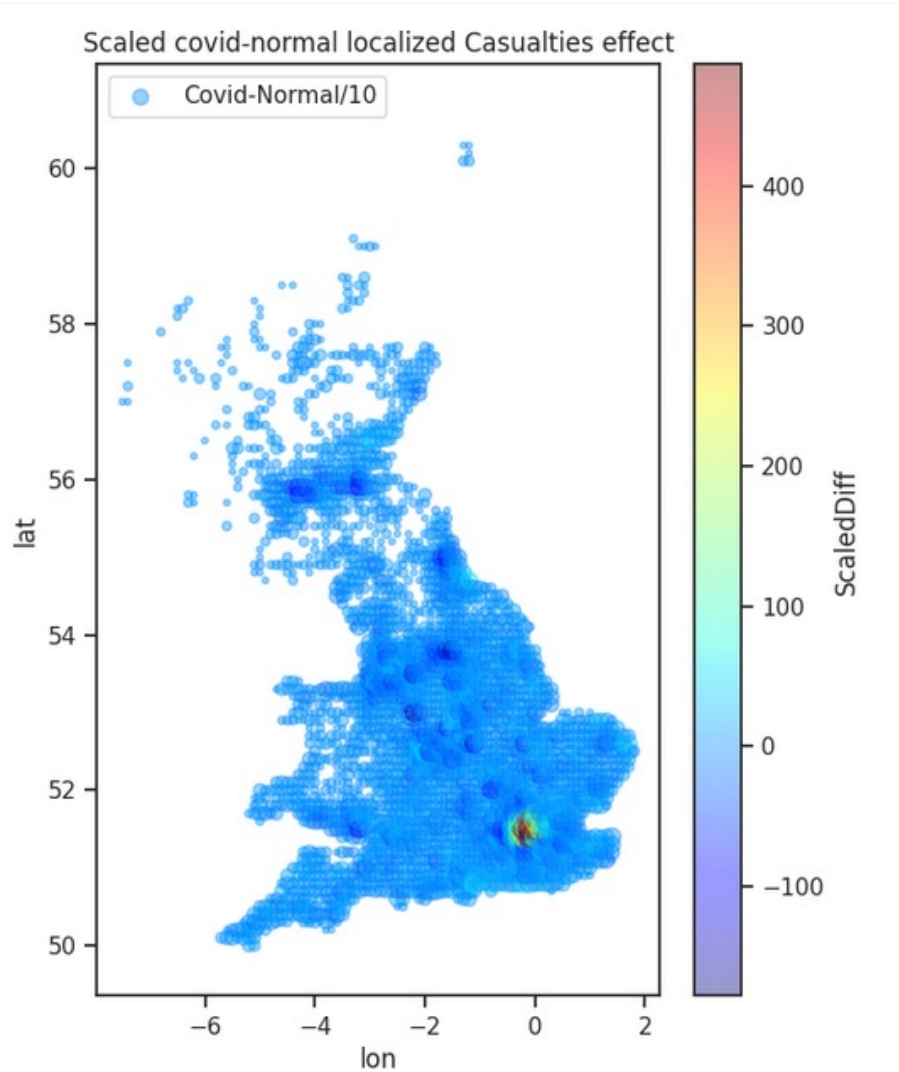
Effect of covid on target value





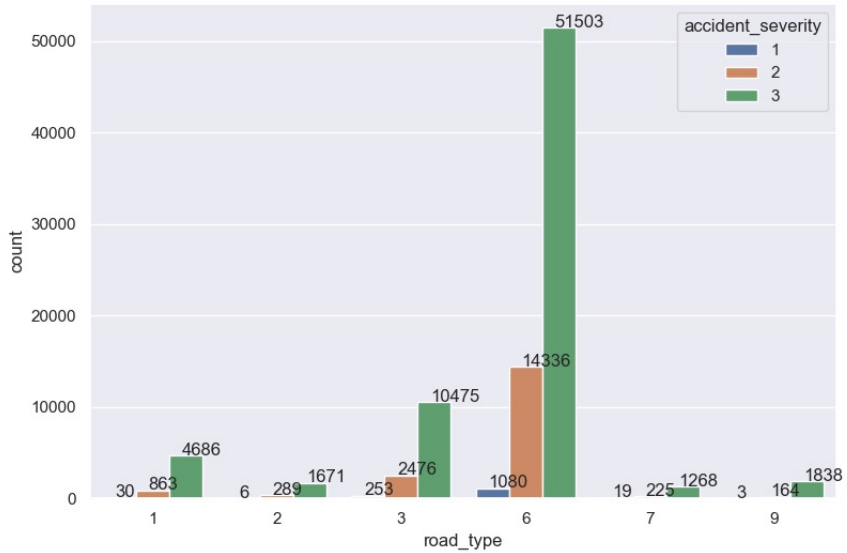
We are 20 sigma away from simiulated non covid samples. It is definetly statistically significant

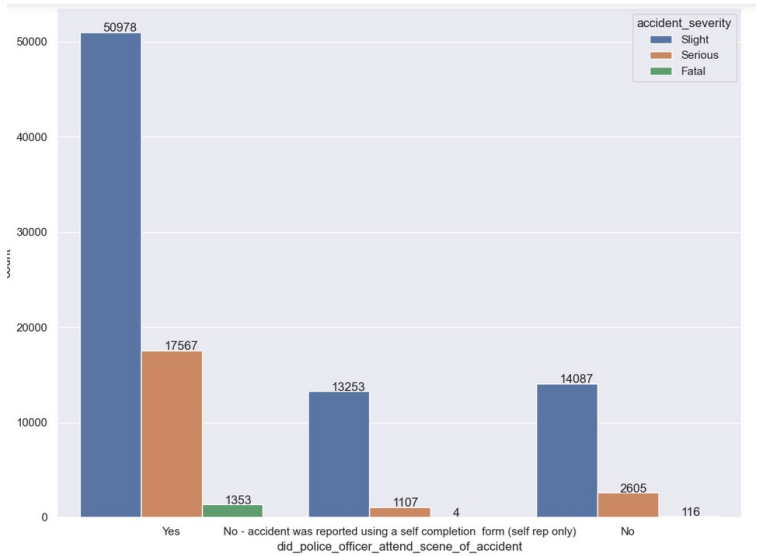


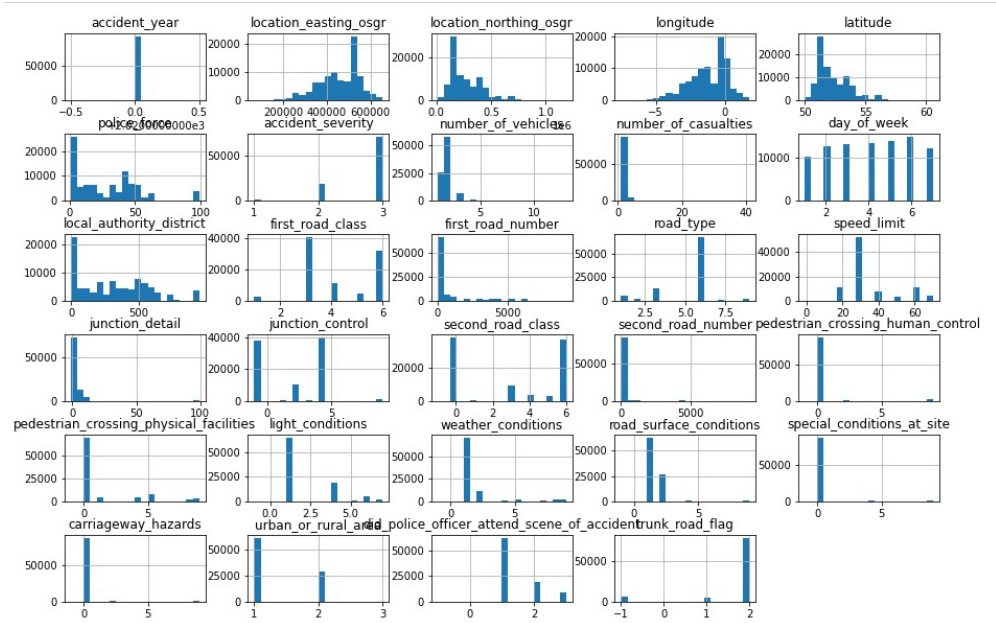


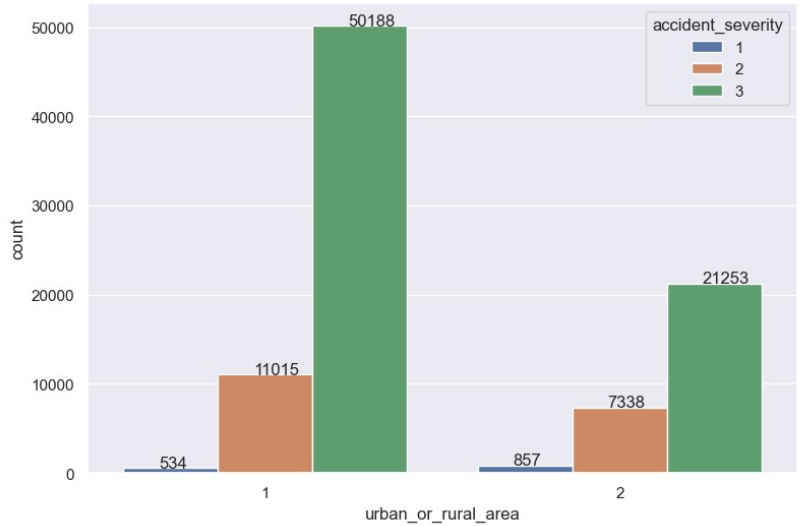
Majority of accidents occurred in 30 speed limit zones. It would have been beneficial to have actual data on the speeds of the vehicles involved or at least if they were speeding.

EDR

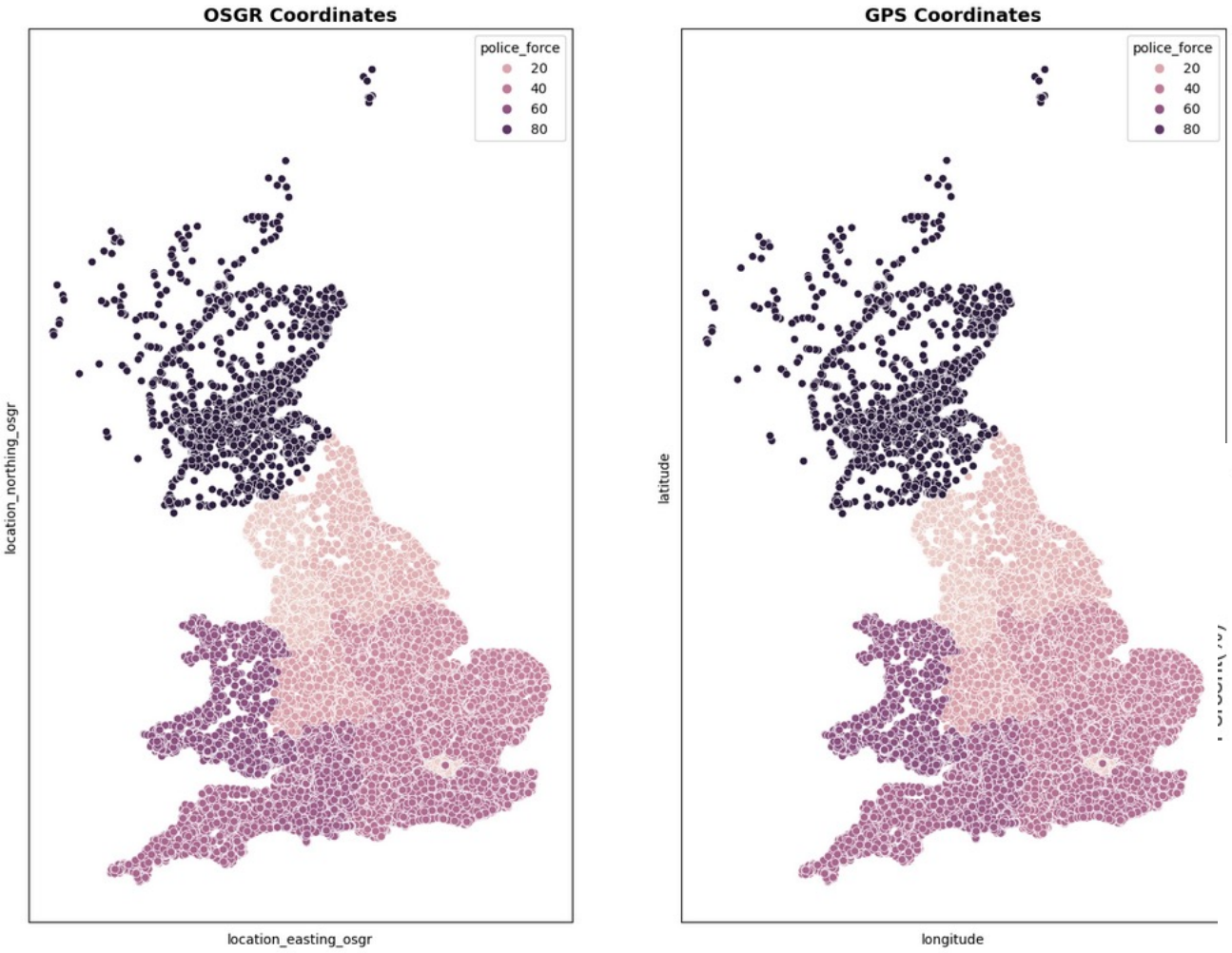

accident_severity
1
2
3
count
road_type
30
863
4686
6
289
1671
253
2476
10475
1080
14336
51503
19
225
1268
3
164
1838


accident_severity
Slight
Serious
Fatal
50978
17567
1353
13253
1107
4
14087
2605
116
Yes
No - accident was reported using a self completion form (self rep only)
No
did_police_officer_attend_scene_of_accident

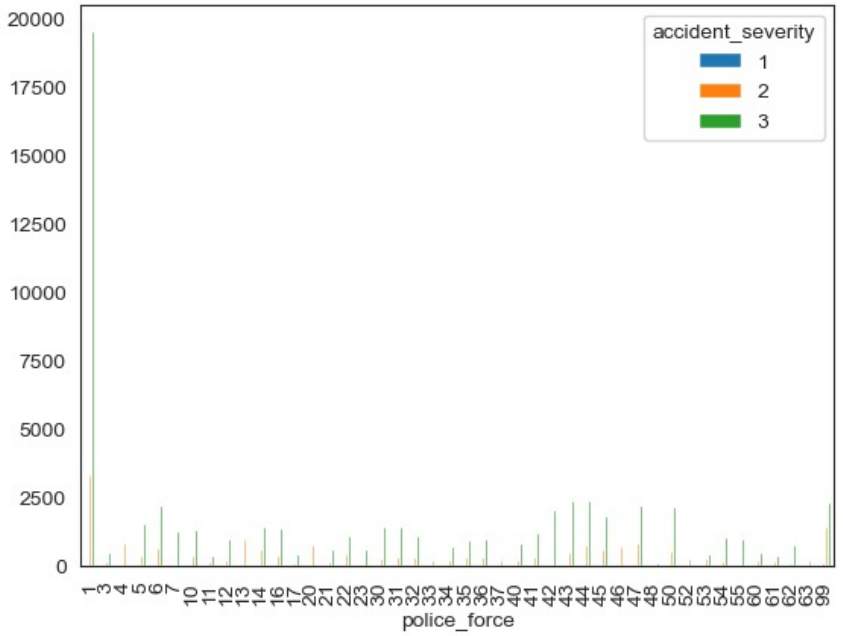

accident_year
location_easting_osgr
location_northing_osgr
longitude
latitude
police_force
accident_severity
number_of_vehicles
number_of_casualties
day_of_week
local_authority_district
first_road_class
first_road_number
road_type
speed_limit
junction_detail
junction_control
second_road_class
second_road_number
pedestrian_crossing_human_control
pedestrian_crossing_physical_facilities
light_conditions
weather_conditions
road_surface_conditions
special_conditions_at_site
carriageway_hazards
urban_or_rural_area
did_police_officer_attend_scene_of_accident
trunk_road_flag

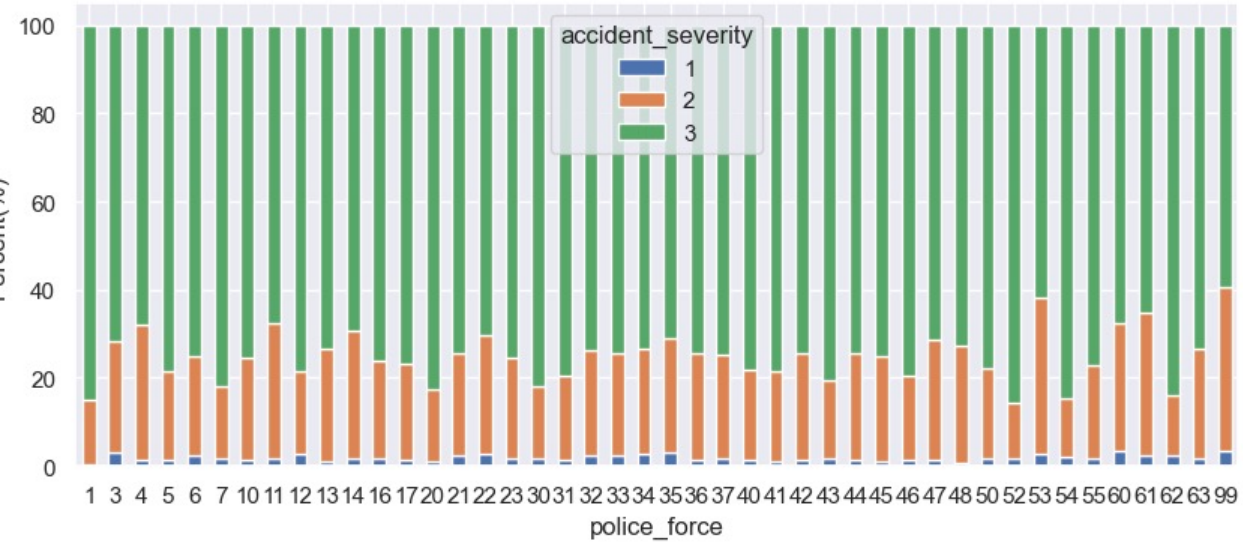

accident_severity
1
2
3
count
urban_or_rural_area
534
11015
50188
857
7338
21253

EDR


OSGR Coordinates
police_force
20
40
60
80
location_northing_osgr
location_easting_osgr

GPS Coordinates
police_force
20
40
60
80
latitude
longitude


accident_severity
1
2
3
20000
17500
15000
12500
10000
7500
5000
2500
0
police_force


accident_severity
1
2
3
100
80
60
40
20
0
police_force

Feature engineering & Data PreProcessing

- Relation of many to one from veh_df to acc_df
  - (groupby, aggregate)

- **Train Vlidation and test sets (stratify)**
  - 15% test
  - 15% of rest validation
  - Rest train ~200k
- OHE (one hot encode nominals)

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 278143 entries, 2 to 308585
Data columns (total 40 columns):
 #   Column                                      Non-Null Count   Dtype
---  ------                                      --------------   -----
 0   accident_year                               278143 non-null  int32
 1   longitude                                   278143 non-null  float64
 2   latitude                                    278143 non-null  float64
 3   police_force                                278143 non-null  uint8
 4   accident_severity                           278143 non-null  uint8
 5   number_of_vehicles                          278143 non-null  uint8
 6   number_of_casualties                        278143 non-null  uint8
 7   day_of_week                                 278143 non-null  uint8
 8   first_road_class                            278143 non-null  uint8
 9   road_type                                   278143 non-null  uint8
 10  speed_limit                                 278143 non-null  uint8
 11  junction_detail                             278143 non-null  uint8
 12  pedestrian_crossing_human_control           278143 non-null  uint8
 13  pedestrian_crossing_physical_facilities     278143 non-null  uint8
 14  light_conditions                            278143 non-null  uint8
 15  weather_conditions                          278143 non-null  uint8
 16  road_surface_conditions                     278143 non-null  uint8
 17  special_conditions_at_site                  278143 non-null  uint8
 18  carriageway_hazards                         278143 non-null  uint8
 19  urban_or_rural_area                         278143 non-null  uint8
 20  did_police_officer_attend_scene_of_accident 278143 non-null  uint8
 21  trunk_road_flag                             278143 non-null  uint8
 22  hour                                        278143 non-null  uint8
 23  weekend                                     278143 non-null  uint8
 24  month                                       278143 non-null  uint8
 25  covid                                       278143 non-null  uint8
 26  accident_level                              278143 non-null  uint8
 27  time_of_day                                 278143 non-null  uint8
 28  smallest_veh                                278143 non-null  uint8
 29  biggest_vel                                 278143 non-null  uint8
 30  towing                                      278143 non-null  uint8
 31  restricted                                  278143 non-null  uint8
 32  junction_location                           278143 non-null  uint8
 33  skidding                                    278143 non-null  uint8
 34  jacknifed                                   278143 non-null  uint8
 35  overturned                                  278143 non-null  uint8
 36  xcenterguard                                278143 non-null  uint8
 37  rebounded                                   278143 non-null  uint8
 38  male_drivers                                278143 non-null  uint8
 39  female_drivers                              278143 non-null  uint8
dtypes: float64(2), int32(1), uint8(37)
memory usage: 17.2 MB
```

Imbalance

- Using algorithms with ability to define class weights
- Undersampling
- Oversampling (SMOTE)

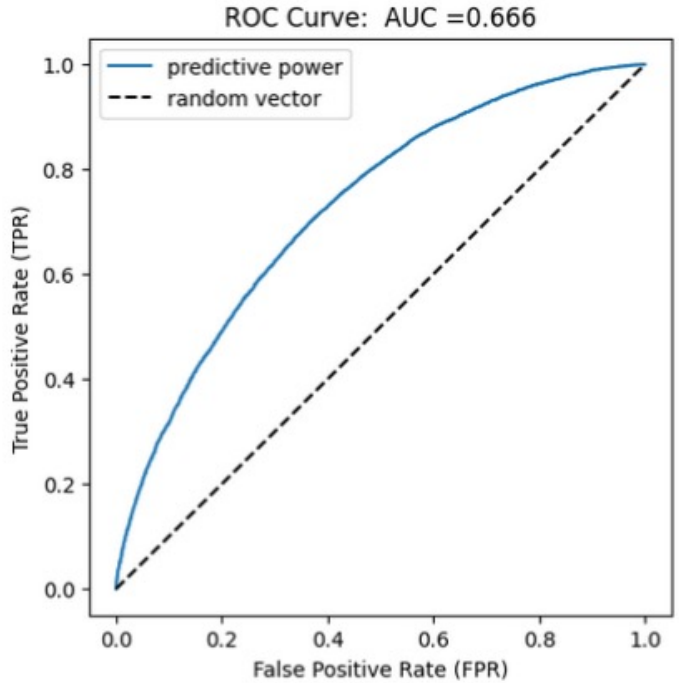
```
imbalance ratio:  3.35 to 1  ,  with  22.97 % being Major accidents and the rest Minor
0    214242
1     63901
Name: accident_level, dtype: int64
```

Models

- Random Forest Classifier
- Balanced Random Forest
- XGBOOST
- Naïve-Base
- Logistic regression

```python
# Define the F1 score scorer for the 'Major' class
def f1_score_positive(y_true, y_pred):
    return f1_score(y_true, y_pred, pos_label=1)

scorer = make_scorer(f1_score, pos_label=1)
f1_scorer = make_scorer(f1_score_positive)
scorers = {
    'f1_score': f1_scorer, # make_scorer(f1_score),
    'precision': make_scorer(precision_score),
    'recall': make_scorer(recall_score),
    'accuracy': make_scorer(accuracy_score),
    'auc': make_scorer(roc_auc_score),
    'auprc': make_scorer(average_precision_score),
}
# ...
cv1 = StratifiedKFold(n_splits=5, random_state=82, shuffle=True)



import xgboost as xgb
# Create the XGBoost classifier
xgb_model = xgb.XGBClassifier()

# Define the parameter grid for hyperparameter tuning
parameters_xgb = {
    'n_estimators': [200],
    'max_depth': [8,10],
    'learning_rate': [0.1,0.05],
    'scale_pos_weight': [3.5,4],
    'subsample': [0.7],
    'colsample_bytree': [.9]
}

# Create the GridSearchCV object
xgb_clf = GridSearchCV(xgb_model, param_grid=parameters_xgb, cv=cv1,
                       scoring=scorers, refit='f1_score', n_jobs=-1, verbose=2)
```

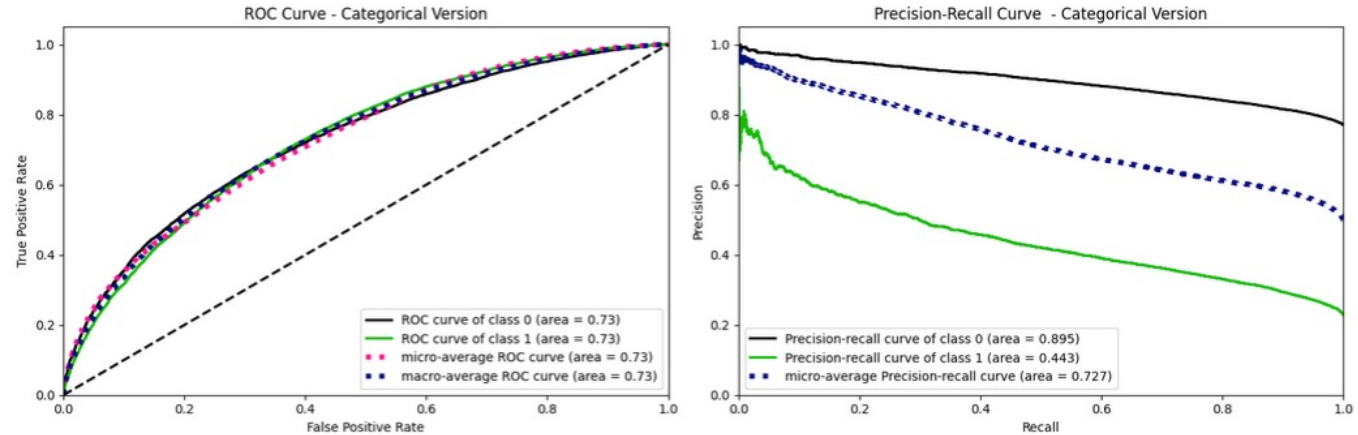
- Cross validation
- Hyper parameter tuning:
  - Grid search
  - Random search
- Various imbalance approach
- The best models where RFC and XGB very similar results after tuning



	precision	recall	f1-score	support
0	0.87	0.65	0.74	27316
1	0.37	0.68	0.48	8148
accuracy			0.66	35464
macro avg	0.62	0.67	0.61	35464
weighted avg	0.76	0.66	0.68	35464

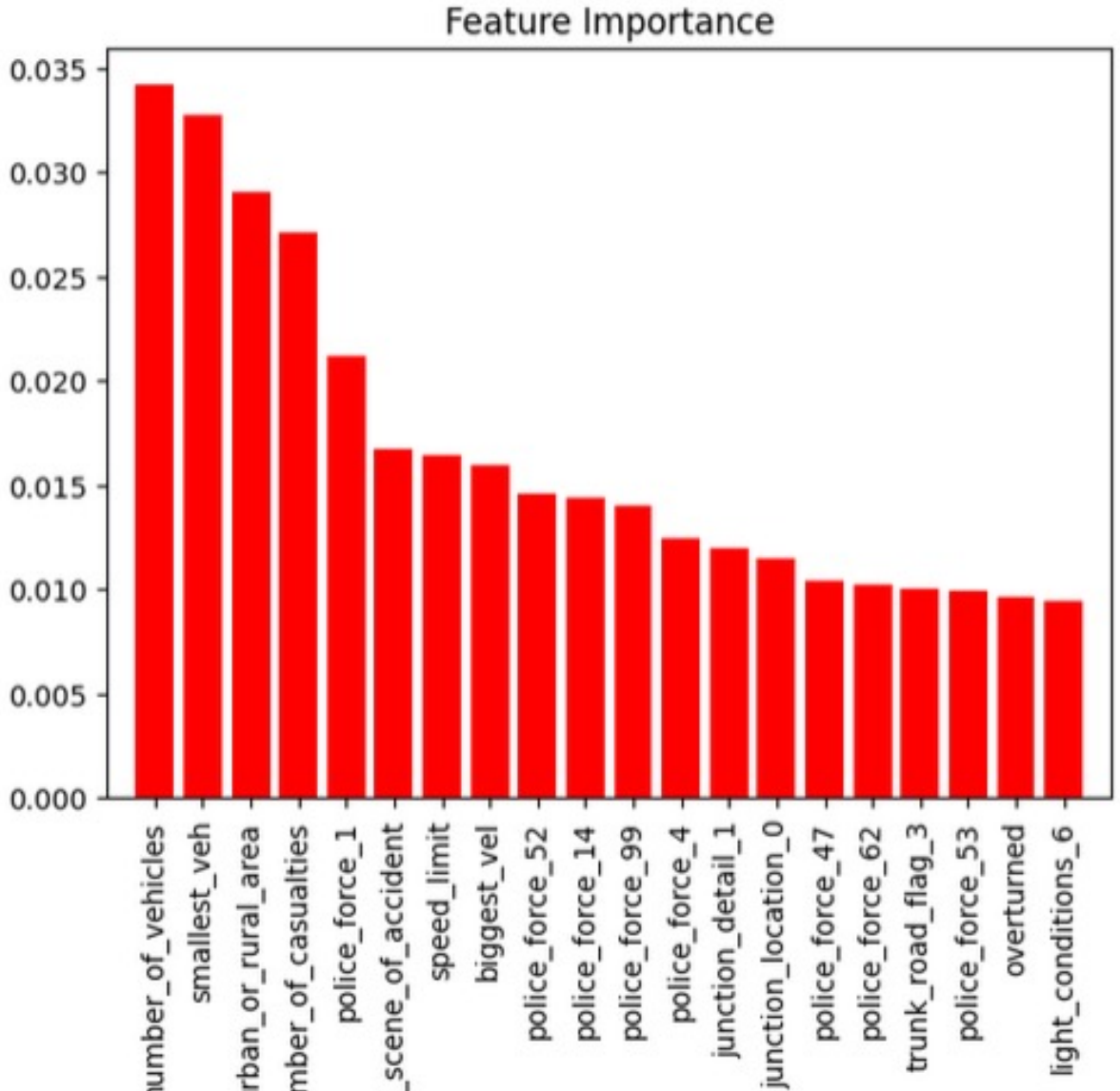
	predicted \n Minor	predicted \n Major
Actual Minor_Acc	17724	9592
Actual Major_Acc	2589	5559

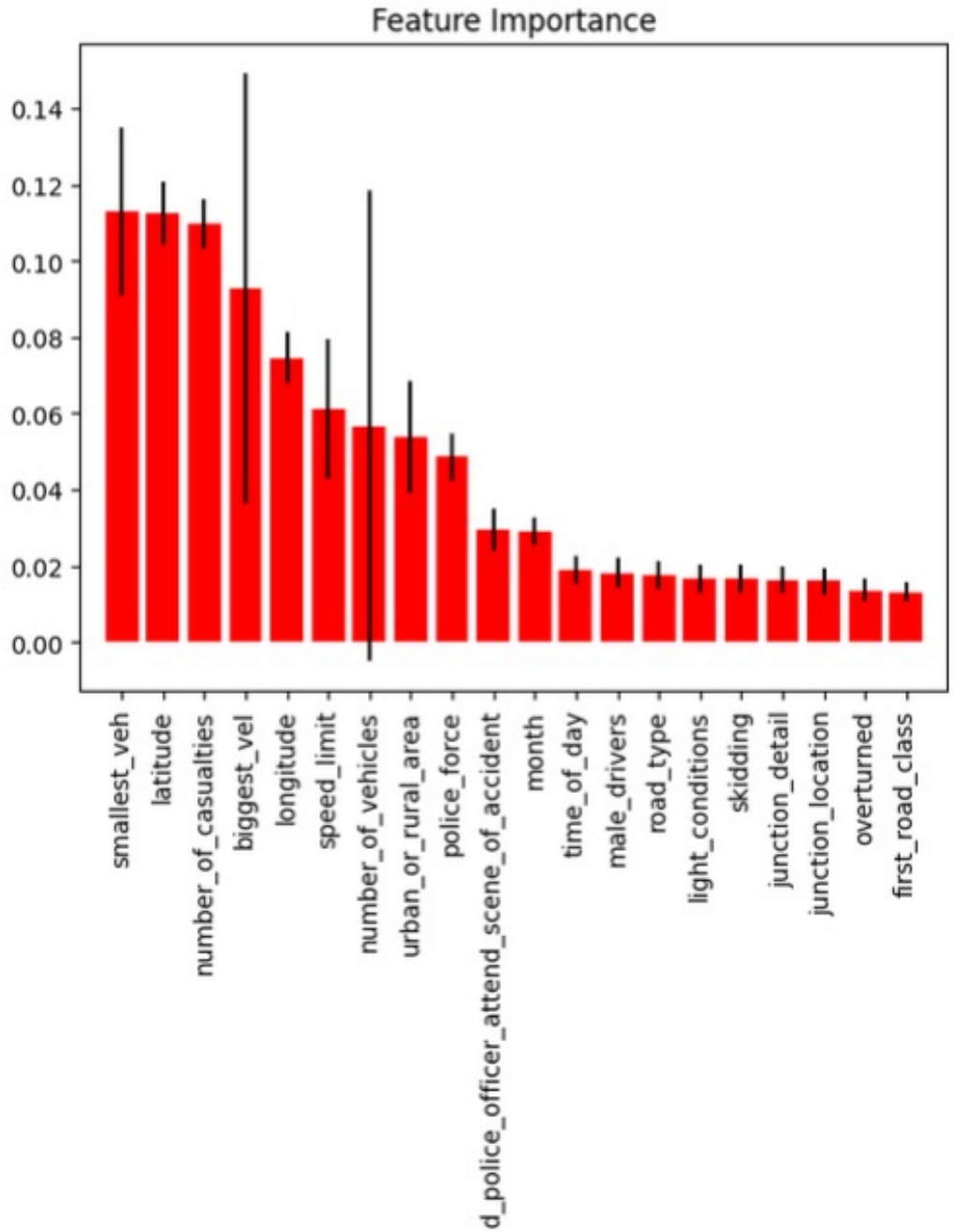
```
[ ] ROC_PRC(xgb_best, X_test, y_test)
```




```
AUC score:  0.7285192905476249        AUPRC score:  0.4427014384122914
(0.7285192905476249, 0.4427014384122914)
```

```
skplt.estimators.plot_feature_importances(xgb_best, feature_names= list(X_train.columns))
plt.xticks(rotation=90)
```


Feature Importance
0.035
0.030
0.025
0.020
0.015
0.010
0.005
0.000
number_of_vehicles
smallest_veh
urban_or_rural_area
number_of_casualties
police_force_1
ttend_scene_of_accident
speed_limit
biggest_vel
police_force_52
police_force_14
police_force_99
police_force_4
junction_detail_1
junction_location_0
police_force_47
police_force_62
trunk_road_flag_3
police_force_53
overturned
light_conditions_6


Feature Importance
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
smallest_veh
latitude
number_of_casualties
biggest_vel
longitude
speed_limit
number_of_vehicles
urban_or_rural_area
police_force
d_police_officer_attend_scene_of_accident
month
time_of_day
male_drivers
road_type
light_conditions
skidding
junction_detail
junction_location
overturned
first_road_class

Next Steps:

- Maybe instead of projecting veh_df to acc_df it is better to run the model for each vehicle involved in the accident and then chose the severity of the accident and urgency of sending resources to accident.

- Using more data that is going to be soon released to decrease the effect of covid

- Using a neural network as well to find the patterns