rta-project

December 5, 2023

1 Introduction

Road traffic accidents are a significant global concern, causing countless injuries and fatalities each year. Addressing the issue of road safety requires proactive measures, including the use of advanced technologies like machine learning. The "Road Traffic Accident Machine Learning Classification Project" aims to leverage machine learning techniques to categorize and predict the outcomes of road traffic accidents, ultimately enhancing safety and response efforts.

1.1 Loading Libraries And Data

[ ]:
```
#importing libraries
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

[ ]:
```
#load and read the file
df=pd.read_csv("/content/RTA Dataset.csv")
df.head()
```

[ ]:
```
       Time Day_of_week Age_band_of_driver Sex_of_driver   Educational_level  \
0  17:02:00      Monday              18-30          Male   Above high school
1  17:02:00      Monday              31-50          Male  Junior high school
2  17:02:00      Monday              18-30          Male  Junior high school
3   1:06:00      Sunday              18-30          Male  Junior high school
4   1:06:00      Sunday              18-30          Male  Junior high school

  Vehicle_driver_relation Driving_experience      Type_of_vehicle  \
0                Employee             1-2yr           Automobile
1                Employee        Above 10yr  Public (> 45 seats)
2                Employee             1-2yr      Lorry (41?100Q)
3                Employee            5-10yr  Public (> 45 seats)
4                Employee             2-5yr                  NaN

  Owner_of_vehicle Service_year_of_vehicle  … Vehicle_movement  \
0            Owner               Above 10yr  …   Going straight
1            Owner                  5-10yrs  …   Going straight
```

```
2            Owner                     NaN  …   Going straight
3     Governmental                     NaN  …   Going straight
4            Owner                5-10yrs  …   Going straight

    Casualty_class Sex_of_casualty Age_band_of_casualty Casualty_severity  \
0               na              na                   na               na
1               na              na                   na               na
2  Driver or rider            Male                31-50                3
3       Pedestrian          Female                18-30                3
4               na              na                   na               na

  Work_of_casuality Fitness_of_casuality Pedestrian_movement  \
0               NaN                  NaN    Not a Pedestrian
1               NaN                  NaN    Not a Pedestrian
2            Driver                  NaN    Not a Pedestrian
3            Driver               Normal    Not a Pedestrian
4               NaN                  NaN    Not a Pedestrian

           Cause_of_accident Accident_severity
0             Moving Backward     Slight Injury
1                  Overtaking     Slight Injury
2   Changing lane to the left    Serious Injury
3  Changing lane to the right     Slight Injury
4                  Overtaking     Slight Injury

[5 rows x 32 columns]
```

[ ]:
```
df.tail()
```

[ ]:
```
           Time Day_of_week Age_band_of_driver Sex_of_driver  \
12311  16:15:00   Wednesday              31-50          Male
12312  18:00:00      Sunday            Unknown          Male
12313  13:55:00      Sunday            Over 51          Male
12314  13:55:00      Sunday              18-30        Female
12315  13:55:00      Sunday              18-30          Male

        Educational_level Vehicle_driver_relation Driving_experience  \
12311                NaN               Employee             2-5yr
12312   Elementary school               Employee            5-10yr
12313  Junior high school               Employee            5-10yr
12314  Junior high school               Employee        Above 10yr
12315  Junior high school               Employee            5-10yr

       Type_of_vehicle Owner_of_vehicle Service_year_of_vehicle  …  \
12311   Lorry (11?40Q)            Owner                     NaN  …
12312       Automobile            Owner                     NaN  …
12313            Bajaj            Owner                  2-5yrs  …
```

```
12314  Lorry (41?100Q)            Owner                  2-5yrs  …
12315            Other            Owner                  2-5yrs  …

      Vehicle_movement   Casualty_class Sex_of_casualty Age_band_of_casualty  \
12311   Going straight              na              na                    na
12312            Other              na              na                    na
12313            Other  Driver or rider            Male                 31-50
12314            Other              na              na                    na
12315         Stopping       Pedestrian          Female                     5

      Casualty_severity Work_of_casuality Fitness_of_casuality  \
12311                na            Driver               Normal
12312                na            Driver               Normal
12313                 3            Driver               Normal
12314                na            Driver               Normal
12315                 3            Driver               Normal

                                  Pedestrian_movement  \
12311                                Not a Pedestrian
12312                                Not a Pedestrian
12313                                Not a Pedestrian
12314                                Not a Pedestrian
12315  Crossing from nearside - masked by parked or s…

                       Cause_of_accident Accident_severity
12311                     No distancing     Slight Injury
12312                     No distancing     Slight Injury
12313           Changing lane to the right    Serious Injury
12314  Driving under the influence of drugs     Slight Injury
12315           Changing lane to the right     Slight Injury

[5 rows x 32 columns]
```

[ ]:
```
#checking each columns
df.columns
```

[ ]:
```
Index(['Time', 'Day_of_week', 'Age_band_of_driver', 'Sex_of_driver',
       'Educational_level', 'Vehicle_driver_relation', 'Driving_experience',
       'Type_of_vehicle', 'Owner_of_vehicle', 'Service_year_of_vehicle',
       'Defect_of_vehicle', 'Area_accident_occured', 'Lanes_or_Medians',
       'Road_allignment', 'Types_of_Junction', 'Road_surface_type',
       'Road_surface_conditions', 'Light_conditions', 'Weather_conditions',
       'Type_of_collision', 'Number_of_vehicles_involved',
       'Number_of_casualties', 'Vehicle_movement', 'Casualty_class',
       'Sex_of_casualty', 'Age_band_of_casualty', 'Casualty_severity',
       'Work_of_casuality', 'Fitness_of_casuality', 'Pedestrian_movement',
       'Cause_of_accident', 'Accident_severity'],
```

```
      dtype='object')
```

[ ]:
```
#shape/ size of the data
df.shape
```

[ ]:
```
(12316, 32)
```

[ ]:
```
#checking the numerical statistics of the data
df.describe()
```

[ ]:

	Number_of_vehicles_involved	Number_of_casualties
count	12316.000000	12316.000000
mean	2.040679	1.548149
std	0.688790	1.007179
min	1.000000	1.000000
25%	2.000000	1.000000
50%	2.000000	1.000000
75%	2.000000	2.000000
max	7.000000	8.000000

[ ]:
```
#checking data types of each columns
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 12316 entries, 0 to 12315
Data columns (total 32 columns):
 #   Column                       Non-Null Count  Dtype
---  ------                       --------------  -----
 0   Time                         12316 non-null  object
 1   Day_of_week                  12316 non-null  object
 2   Age_band_of_driver           12316 non-null  object
 3   Sex_of_driver                12316 non-null  object
 4   Educational_level            11575 non-null  object
 5   Vehicle_driver_relation      11737 non-null  object
 6   Driving_experience           11487 non-null  object
 7   Type_of_vehicle              11366 non-null  object
 8   Owner_of_vehicle             11834 non-null  object
 9   Service_year_of_vehicle      8388 non-null   object
 10  Defect_of_vehicle            7889 non-null   object
 11  Area_accident_occured        12077 non-null  object
 12  Lanes_or_Medians             11931 non-null  object
 13  Road_allignment              12174 non-null  object
 14  Types_of_Junction            11429 non-null  object
 15  Road_surface_type            12144 non-null  object
 16  Road_surface_conditions      12316 non-null  object
 17  Light_conditions             12316 non-null  object
 18  Weather_conditions           12316 non-null  object
```

```
 19  Type_of_collision            12161 non-null  object
 20  Number_of_vehicles_involved  12316 non-null  int64
 21  Number_of_casualties         12316 non-null  int64
 22  Vehicle_movement             12008 non-null  object
 23  Casualty_class               12316 non-null  object
 24  Sex_of_casualty              12316 non-null  object
 25  Age_band_of_casualty         12316 non-null  object
 26  Casualty_severity            12316 non-null  object
 27  Work_of_casuality            9118 non-null   object
 28  Fitness_of_casuality         9681 non-null   object
 29  Pedestrian_movement          12316 non-null  object
 30  Cause_of_accident            12316 non-null  object
 31  Accident_severity            12316 non-null  object
dtypes: int64(2), object(30)
memory usage: 3.0+ MB
```

1.2 Exploratory Data Analysis

[ ]:
```
#finding duplicate values
df.duplicated().sum()
```

[ ]:
```
0
```

[ ]:
```
#Handling Missing Values
df.isna().sum()
```

[ ]:
```
Time                           0
Day_of_week                    0
Age_band_of_driver             0
Sex_of_driver                  0
Educational_level            741
Vehicle_driver_relation      579
Driving_experience           829
Type_of_vehicle              950
Owner_of_vehicle             482
Service_year_of_vehicle     3928
Defect_of_vehicle           4427
Area_accident_occured        239
Lanes_or_Medians             385
Road_allignment              142
Types_of_Junction            887
Road_surface_type            172
Road_surface_conditions        0
Light_conditions               0
Weather_conditions             0
Type_of_collision            155
Number_of_vehicles_involved    0
```

```
Number_of_casualties              0
Vehicle_movement                308
Casualty_class                    0
Sex_of_casualty                   0
Age_band_of_casualty              0
Casualty_severity                 0
Work_of_casuality              3198
Fitness_of_casuality           2635
Pedestrian_movement               0
Cause_of_accident                 0
Accident_severity                 0
dtype: int64
```

[ ]:
```
#dropping columns which has more than 2500 missing values and Time column
df.
 ↪drop(['Educational_level','Service_year_of_vehicle','Defect_of_vehicle','Work_of_casuality'
 ↪'Fitness_of_casuality','Time','Owner_of_vehicle', 'Type_of_vehicle',␣
 ↪'Road_surface_conditions',␣
 ↪'Pedestrian_movement','Casualty_severity','Educational_level','Day_of_week','Sex_of_driver'
 ↪axis = 1, inplace = True)
df.head()
```

[ ]:
```
  Age_band_of_driver Vehicle_driver_relation Driving_experience  \
0              18-30                Employee              1-2yr
1              31-50                Employee         Above 10yr
2              18-30                Employee              1-2yr
3              18-30                Employee             5-10yr
4              18-30                Employee              2-5yr

  Area_accident_occured   Lanes_or_Medians Types_of_Junction  \
0     Residential areas                NaN       No junction
1          Office areas  Undivided Two way       No junction
2    Recreational areas              other       No junction
3          Office areas              other           Y Shape
4      Industrial areas              other           Y Shape

  Road_surface_type       Light_conditions Weather_conditions  \
0     Asphalt roads              Daylight             Normal
1     Asphalt roads              Daylight             Normal
2     Asphalt roads              Daylight             Normal
3       Earth roads  Darkness - lights lit             Normal
4     Asphalt roads  Darkness - lights lit             Normal

                        Type_of_collision  Number_of_vehicles_involved  \
0  Collision with roadside-parked vehicles                            2
1           Vehicle with vehicle collision                            2
2          Collision with roadside objects                            2
```

```
3           Vehicle with vehicle collision                             2
4           Vehicle with vehicle collision                             2

   Number_of_casualties Vehicle_movement   Casualty_class  \
0                     2   Going straight              na
1                     2   Going straight              na
2                     2   Going straight  Driver or rider
3                     2   Going straight       Pedestrian
4                     2   Going straight              na

  Age_band_of_casualty           Cause_of_accident Accident_severity
0                   na             Moving Backward     Slight Injury
1                   na                   Overtaking     Slight Injury
2                31-50   Changing lane to the left    Serious Injury
3                18-30  Changing lane to the right     Slight Injury
4                   na                   Overtaking     Slight Injury
```

[ ]:
```
#storing categorical column names to a new variable
category=['Vehicle_driver_relation','Driving_experience','Area_accident_occured','Lanes_or_Med
print(category)
```

```
['Vehicle_driver_relation', 'Driving_experience', 'Area_accident_occured',
'Lanes_or_Medians', 'Types_of_Junction', 'Road_surface_type',
'Type_of_collision', 'Vehicle_movement']
```

[ ]:
```
#for categorical values we can replace the null values with the Mode of it
for i in category:
    df[i].fillna(df[i].mode()[0],inplace=True)
```

[ ]:
```
#checking the current null values
df.isna().sum()
```

[ ]:
```
Age_band_of_driver             0
Vehicle_driver_relation        0
Driving_experience             0
Area_accident_occured          0
Lanes_or_Medians               0
Types_of_Junction              0
Road_surface_type              0
Light_conditions               0
Weather_conditions             0
Type_of_collision              0
Number_of_vehicles_involved    0
Number_of_casualties           0
Vehicle_movement               0
Casualty_class                 0
Age_band_of_casualty           0
```

```
Cause_of_accident             0
Accident_severity             0
dtype: int64
```

[ ]:
```
#Handling Categorical values
df.dtypes
```

[ ]:
```
Age_band_of_driver            object
Vehicle_driver_relation       object
Driving_experience            object
Area_accident_occured         object
Lanes_or_Medians              object
Types_of_Junction             object
Road_surface_type             object
Light_conditions              object
Weather_conditions            object
Type_of_collision             object
Number_of_vehicles_involved    int64
Number_of_casualties           int64
Vehicle_movement              object
Casualty_class                object
Age_band_of_casualty          object
Cause_of_accident             object
Accident_severity             object
dtype: object
```

[ ]:
```
#get_dummies
df1=pd.
  ↪get_dummies(df[['Age_band_of_driver','Vehicle_driver_relation','Driving_experience','Area_a
df1.head()
```

[ ]:
```
   Age_band_of_driver_31-50  Age_band_of_driver_Over 51  \
0                         0                           0
1                         1                           0
2                         0                           0
3                         0                           0
4                         0                           0

   Age_band_of_driver_Under 18  Age_band_of_driver_Unknown  \
0                            0                           0
1                            0                           0
2                            0                           0
3                            0                           0
4                            0                           0

   Vehicle_driver_relation_Other  Vehicle_driver_relation_Owner  \
0                              0                              0
```

```
1                                0                                0
2                                0                                0
3                                0                                0
4                                0                                0

   Vehicle_driver_relation_Unknown  Driving_experience_2-5yr  \
0                                0                         0
1                                0                         0
2                                0                         0
3                                0                         0
4                                0                         1

   Driving_experience_5-10yr  Driving_experience_Above 10yr  …  \
0                          0                              0  …
1                          0                              1  …
2                          0                              0  …
3                          1                              0  …
4                          0                              0  …

   Cause_of_accident_No distancing  \
0                                0
1                                0
2                                0
3                                0
4                                0

   Cause_of_accident_No priority to pedestrian  \
0                                            0
1                                            0
2                                            0
3                                            0
4                                            0

   Cause_of_accident_No priority to vehicle  Cause_of_accident_Other  \
0                                         0                       0
1                                         0                       0
2                                         0                       0
3                                         0                       0
4                                         0                       0

   Cause_of_accident_Overloading  Cause_of_accident_Overspeed  \
0                              0                            0
1                              0                            0
2                              0                            0
3                              0                            0
4                              0                            0
```

```
   Cause_of_accident_Overtaking  Cause_of_accident_Overturning  \
0                             0                              0
1                             1                              0
2                             0                              0
3                             0                              0
4                             1                              0

   Cause_of_accident_Turnover  Cause_of_accident_Unknown
0                           0                         0
1                           0                         0
2                           0                         0
3                           0                         0
4                           0                         0

[5 rows x 102 columns]
```

[ ]:
```
#concatinate dummy and old data frame
df2=pd.concat([df,df1],axis=1)
df2.head()
```

[ ]:
```
  Age_band_of_driver Vehicle_driver_relation Driving_experience  \
0              18-30               Employee              1-2yr
1              31-50               Employee         Above 10yr
2              18-30               Employee              1-2yr
3              18-30               Employee             5-10yr
4              18-30               Employee              2-5yr

  Area_accident_occured                                  Lanes_or_Medians  \
0     Residential areas  Two-way (divided with broken lines road marking)
1          Office areas                                 Undivided Two way
2    Recreational areas                                             other
3          Office areas                                             other
4      Industrial areas                                             other

  Types_of_Junction Road_surface_type       Light_conditions  \
0       No junction     Asphalt roads               Daylight
1       No junction     Asphalt roads               Daylight
2       No junction     Asphalt roads               Daylight
3           Y Shape       Earth roads  Darkness - lights lit
4           Y Shape     Asphalt roads  Darkness - lights lit

  Weather_conditions                       Type_of_collision  …  \
0             Normal  Collision with roadside-parked vehicles  …
1             Normal           Vehicle with vehicle collision  …
2             Normal          Collision with roadside objects  …
3             Normal           Vehicle with vehicle collision  …
4             Normal           Vehicle with vehicle collision  …
```

```
   Cause_of_accident_No distancing  \
0                                0
1                                0
2                                0
3                                0
4                                0

   Cause_of_accident_No priority to pedestrian  \
0                                            0
1                                            0
2                                            0
3                                            0
4                                            0

  Cause_of_accident_No priority to vehicle Cause_of_accident_Other  \
0                                        0                       0
1                                        0                       0
2                                        0                       0
3                                        0                       0
4                                        0                       0

  Cause_of_accident_Overloading Cause_of_accident_Overspeed  \
0                             0                           0
1                             0                           0
2                             0                           0
3                             0                           0
4                             0                           0

  Cause_of_accident_Overtaking  Cause_of_accident_Overturning  \
0                            0                              0
1                            1                              0
2                            0                              0
3                            0                              0
4                            1                              0

   Cause_of_accident_Turnover  Cause_of_accident_Unknown
0                           0                         0
1                           0                         0
2                           0                         0
3                           0                         0
4                           0                         0

[5 rows x 119 columns]
```

[ ]:
```
#dropping dummied columns
```

```
df2.
  ↪drop(['Age_band_of_driver','Vehicle_driver_relation','Driving_experience','Area_accident_oc
  ↪= 1, inplace = True)
df2.head()
```

[ ]:

	Number_of_vehicles_involved	Number_of_casualties	Accident_severity	\
0	2	2	Slight Injury	
1	2	2	Slight Injury	
2	2	2	Serious Injury	
3	2	2	Slight Injury	
4	2	2	Slight Injury	

	Age_band_of_driver_31-50	Age_band_of_driver_Over 51	\
0	0	0	
1	1	0	
2	0	0	
3	0	0	
4	0	0	

	Age_band_of_driver_Under 18	Age_band_of_driver_Unknown	\
0	0	0	
1	0	0	
2	0	0	
3	0	0	
4	0	0	

	Vehicle_driver_relation_Other	Vehicle_driver_relation_Owner	\
0	0	0	
1	0	0	
2	0	0	
3	0	0	
4	0	0	

	Vehicle_driver_relation_Unknown	…	Cause_of_accident_No distancing	\
0	0	…	0	
1	0	…	0	
2	0	…	0	
3	0	…	0	
4	0	…	0	

	Cause_of_accident_No priority to pedestrian	\
0	0	
1	0	
2	0	
3	0	
4	0	

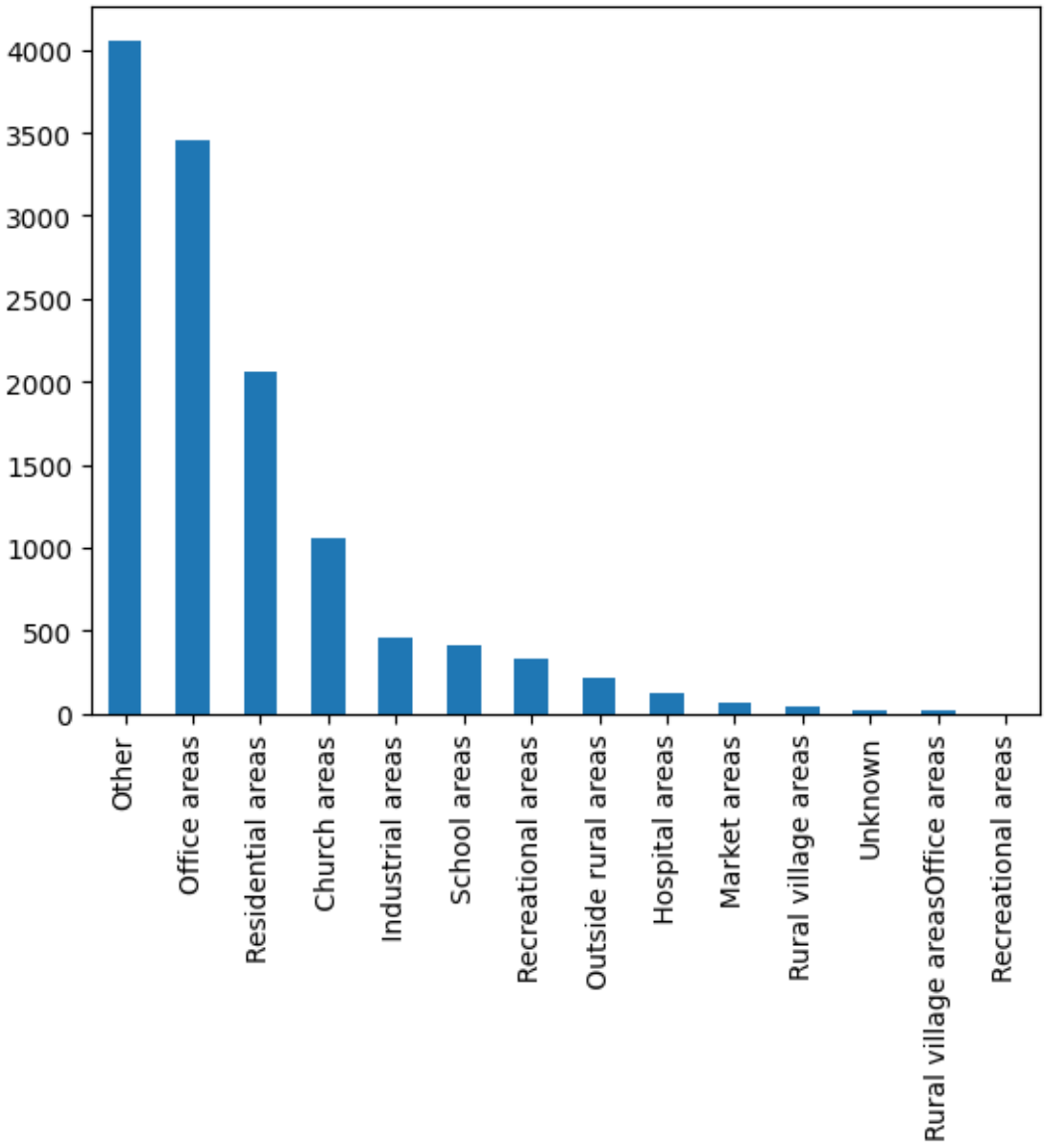
```
   Cause_of_accident_No priority to vehicle  Cause_of_accident_Other  \
0                                         0                       0
1                                         0                       0
2                                         0                       0
3                                         0                       0
4                                         0                       0

   Cause_of_accident_Overloading  Cause_of_accident_Overspeed  \
0                              0                            0
1                              0                            0
2                              0                            0
3                              0                            0
4                              0                            0

   Cause_of_accident_Overtaking  Cause_of_accident_Overturning  \
0                             0                              0
1                             1                              0
2                             0                              0
3                             0                              0
4                             1                              0

   Cause_of_accident_Turnover  Cause_of_accident_Unknown
0                           0                         0
1                           0                         0
2                           0                         0
3                           0                         0
4                           0                         0

[5 rows x 105 columns]
```

1.3 Data Visualization

BAR CHART

[ ]:
```
df["Area_accident_occured"].value_counts().plot(kind='bar')
```

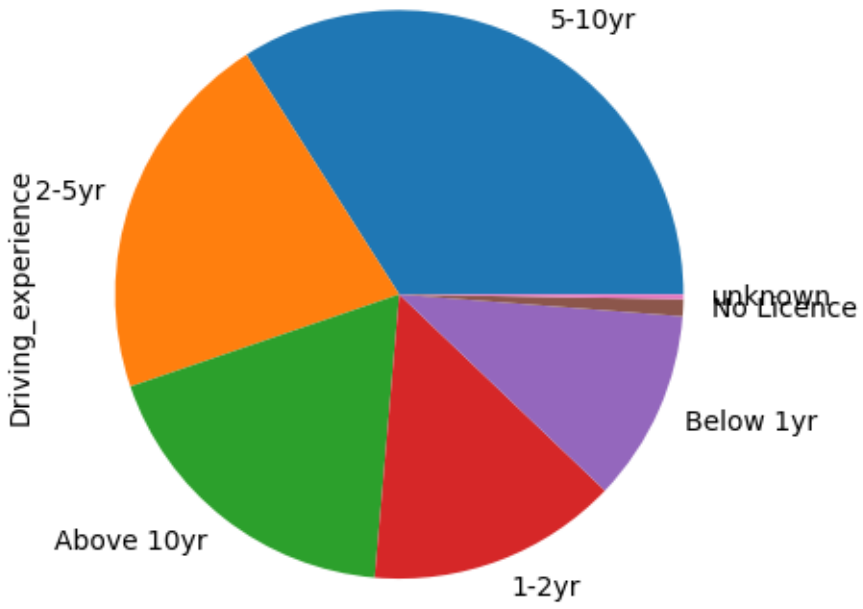
[ ]:
```
<Axes: >
```



PIE CHART

[ ]:
```
df["Driving_experience"].value_counts().plot(kind='pie')
```

[ ]:
```
<Axes: ylabel='Driving_experience'>
```



[ ]:
```
#checking the correlation between numerical columns
df.corr()
```

```
<ipython-input-20-c7435214f394>:2: FutureWarning: The default value of
numeric_only in DataFrame.corr is deprecated. In a future version, it will
default to False. Select only valid columns or specify the value of numeric_only
to silence this warning.
  df.corr()
```

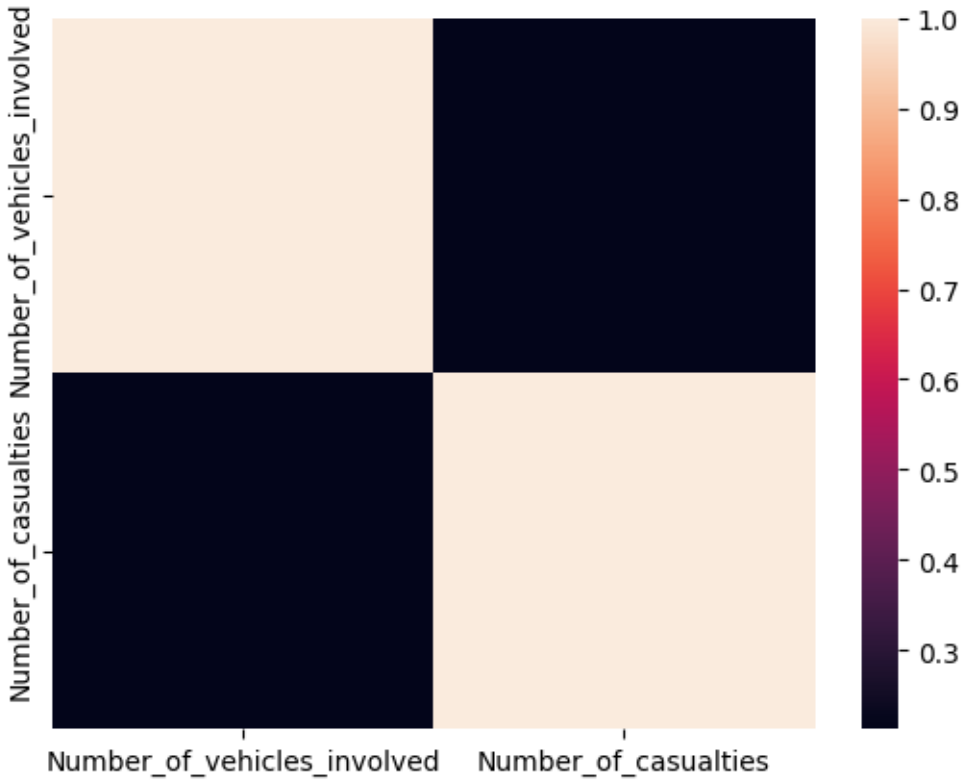
[ ]:

	Number_of_vehicles_involved	Number_of_casualties
Number_of_vehicles_involved	1.000000	0.213427
Number_of_casualties	0.213427	1.000000

[ ]:
```
#plotting the correlation using heatmap
sns.heatmap(df.corr())
```

```
<ipython-input-21-0f2d154e87b4>:2: FutureWarning: The default value of
numeric_only in DataFrame.corr is deprecated. In a future version, it will
default to False. Select only valid columns or specify the value of numeric_only
to silence this warning.
  sns.heatmap(df.corr())
```

[ ]: <Axes: >



1.3.1 Feauture Selection

[ ]:
```
#import chi2 test
from sklearn.feature_selection import SelectKBest,chi2
x=df2.drop(['Accident_severity'],axis=1)
y=df2[['Accident_severity']]
chi=SelectKBest(chi2,k=80)
best=chi.fit_transform(x,y)
best.shape
```

[ ]:
```
(12316, 80)
```

[ ]:
```
x_nm=chi.get_support(indices=True)
print(df2.columns[x_nm])
```

```
Index(['Number_of_vehicles_involved', 'Number_of_casualties',
       'Accident_severity', 'Age_band_of_driver_31-50',
       'Age_band_of_driver_Over 51', 'Age_band_of_driver_Under 18',
       'Age_band_of_driver_Unknown', 'Vehicle_driver_relation_Other',
       'Vehicle_driver_relation_Owner', 'Vehicle_driver_relation_Unknown',
       'Driving_experience_Above 10yr', 'Driving_experience_Below 1yr',
       'Driving_experience_No Licence', 'Driving_experience_unknown',
```

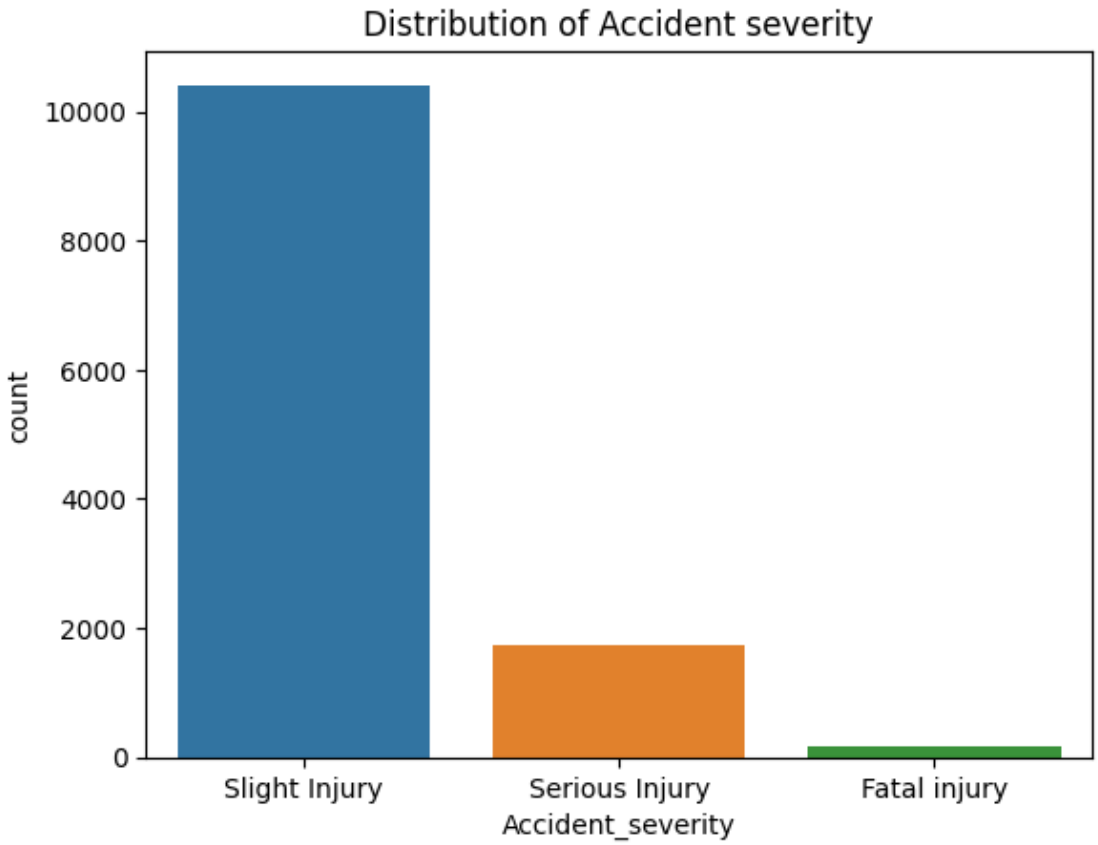
```
 'Area_accident_occured_  Recreational areas',
 'Area_accident_occured_ Church areas',
 'Area_accident_occured_ Industrial areas',
 'Area_accident_occured_ Outside rural areas',
 'Area_accident_occured_Office areas', 'Area_accident_occured_Other',
 'Area_accident_occured_Recreational areas',
 'Area_accident_occured_Residential areas',
 'Area_accident_occured_Rural village areas',
 'Area_accident_occured_School areas', 'Lanes_or_Medians_One way',
 'Lanes_or_Medians_Two-way (divided with solid lines road marking)',
 'Lanes_or_Medians_Undivided Two way', 'Lanes_or_Medians_other',
 'Types_of_Junction_No junction', 'Types_of_Junction_O Shape',
 'Types_of_Junction_Other', 'Types_of_Junction_Unknown',
 'Types_of_Junction_Y Shape',
 'Road_surface_type_Asphalt roads with some distress',
 'Road_surface_type_Earth roads', 'Road_surface_type_Gravel roads',
 'Road_surface_type_Other', 'Light_conditions_Darkness - lights unlit',
 'Light_conditions_Darkness - no lighting',
 'Weather_conditions_Fog or mist', 'Weather_conditions_Normal',
 'Weather_conditions_Other', 'Weather_conditions_Raining',
 'Weather_conditions_Raining and Windy', 'Weather_conditions_Snow',
 'Weather_conditions_Unknown', 'Weather_conditions_Windy',
 'Type_of_collision_Collision with pedestrians',
 'Type_of_collision_Collision with roadside objects',
 'Type_of_collision_Collision with roadside-parked vehicles',
 'Type_of_collision_Rollover', 'Type_of_collision_Unknown',
 'Type_of_collision_Vehicle with vehicle collision',
 'Vehicle_movement_Other', 'Vehicle_movement_Parked',
 'Vehicle_movement_Stopping', 'Vehicle_movement_Turnover',
 'Vehicle_movement_Unknown', 'Vehicle_movement_Waiting to go',
 'Casualty_class_Passenger', 'Casualty_class_Pedestrian',
 'Casualty_class_na', 'Age_band_of_casualty_31-50',
 'Age_band_of_casualty_5', 'Age_band_of_casualty_Over 51',
 'Age_band_of_casualty_Under 18',
 'Cause_of_accident_Changing lane to the right',
 'Cause_of_accident_Driving at high speed',
 'Cause_of_accident_Driving carelessly',
 'Cause_of_accident_Driving under the influence of drugs',
 'Cause_of_accident_Getting off the vehicle improperly',
 'Cause_of_accident_Improper parking',
 'Cause_of_accident_Moving Backward', 'Cause_of_accident_No distancing',
 'Cause_of_accident_No priority to pedestrian',
 'Cause_of_accident_Other', 'Cause_of_accident_Overloading',
 'Cause_of_accident_Overspeed', 'Cause_of_accident_Overturning',
 'Cause_of_accident_Turnover'],
dtype='object')
```

[ ]:
```
#Distribution of Accident severity
df['Accident_severity'].value_counts()
```

[ ]:
```
Slight Injury     10415
Serious Injury     1743
Fatal injury        158
Name: Accident_severity, dtype: int64
```

[ ]:
```
#plotting count plot using seaborn
sns.countplot(x = df2['Accident_severity'])
plt.title('Distribution of Accident severity')
```

[ ]:
```
Text(0.5, 1.0, 'Distribution of Accident severity')
```



###Oversampling

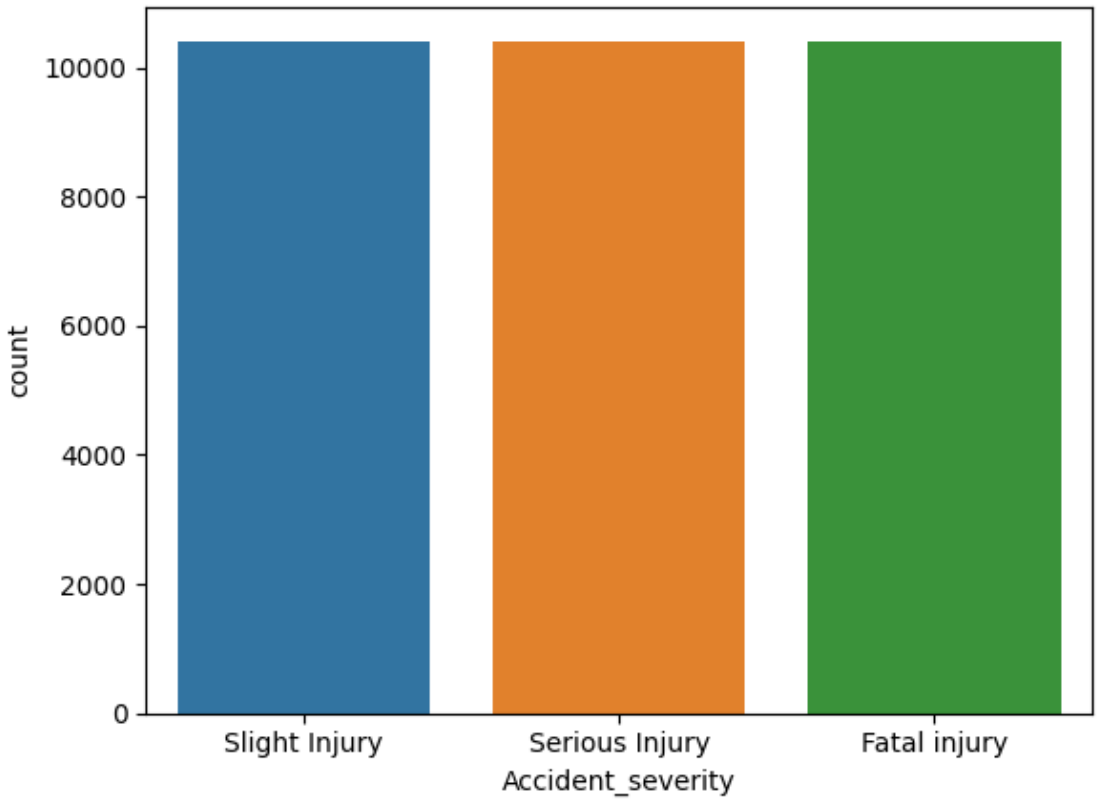
[ ]:
```
#importing SMOTE
from imblearn.over_sampling import SMOTE
oversample=SMOTE()
xo,yo=oversample.fit_resample(best,y)
```

[ ]:
```
#checking the oversampling output
y1=pd.DataFrame(yo)
y1.value_counts()
```

[ ]:
```
Accident_severity
Fatal injury          10415
Serious Injury        10415
Slight Injury         10415
dtype: int64
```

[ ]:
```
sns.countplot(x=yo['Accident_severity'])
```

[ ]:
```
<Axes: xlabel='Accident_severity', ylabel='count'>
```



##Data splitting

[ ]:
```
#converting data to training data and testing data
from sklearn.model_selection import train_test_split
#splitting 70% of the data to training data and 30% of data to testing data
x_train,x_test,y_train,y_test=train_test_split(xo,yo,test_size=0.
 ↪30,random_state=42)
```

[ ]:
```
print(x_train.shape,x_test.shape,y_train.shape,y_test.shape)
```

```
(21871, 80) (9374, 80) (21871, 1) (9374, 1)
```

1.4 Model Creation

[ ]:
```
# implimenting algorithms to create a best model (knn,naive bayes,svm,decision␣
 ↪tree and random forest)

from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import MultinomialNB
from sklearn.svm import SVC
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
knn=KNeighborsClassifier(n_neighbors=5)
nb=MultinomialNB()
svm=SVC()
dec=DecisionTreeClassifier()
rf=RandomForestClassifier(n_estimators=10)
lst_model=[knn,nb,svm,dec,rf]
```

[ ]:
```
from sklearn.metrics import␣
 ↪confusion_matrix,accuracy_score,classification_report
for i in lst_model:
  print(i)
  i.fit(xo,yo)
  y_pred=i.predict(x_test)
 ␣
 ↪print('**********************************************************************')
  print(accuracy_score(y_test,y_pred))
 ␣
 ↪print('**********************************************************************')
  print(confusion_matrix(y_test,y_pred))
 ␣
 ↪print("**********************************************************************")
  print(classification_report(y_test,y_pred))
```

```
KNeighborsClassifier()
```

```
/usr/local/lib/python3.10/dist-
packages/sklearn/neighbors/_classification.py:215: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape
of y to (n_samples,), for example using ravel().
  return self._fit(X, y)
```

```
**********************************************************************
0.8310219756774055
**********************************************************************
```

```
[[3062   63    1]
 [ 148 2935   61]
 [ 332  979 1793]]
**********************************************************************
               precision    recall  f1-score   support

  Fatal injury       0.86      0.98      0.92      3126
Serious Injury       0.74      0.93      0.82      3144
 Slight Injury       0.97      0.58      0.72      3104

      accuracy                           0.83      9374
     macro avg       0.86      0.83      0.82      9374
  weighted avg       0.86      0.83      0.82      9374

MultinomialNB()
**********************************************************************
0.6120119479411137
**********************************************************************

/usr/local/lib/python3.10/dist-packages/sklearn/utils/validation.py:1143:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)

[[2229  677  220]
 [ 769 1590  785]
 [ 331  855 1918]]
**********************************************************************
               precision    recall  f1-score   support

  Fatal injury       0.67      0.71      0.69      3126
Serious Injury       0.51      0.51      0.51      3144
 Slight Injury       0.66      0.62      0.64      3104

      accuracy                           0.61      9374
     macro avg       0.61      0.61      0.61      9374
  weighted avg       0.61      0.61      0.61      9374

SVC()

/usr/local/lib/python3.10/dist-packages/sklearn/utils/validation.py:1143:
DataConversionWarning: A column-vector y was passed when a 1d array was
expected. Please change the shape of y to (n_samples, ), for example using
ravel().
  y = column_or_1d(y, warn=True)

**********************************************************************
0.8296351610838489
**********************************************************************
```

```
[[2943  158   25]
 [ 336 2273  535]
 [  83  460 2561]]
**********************************************************************
               precision    recall  f1-score   support

  Fatal injury       0.88      0.94      0.91      3126
Serious Injury       0.79      0.72      0.75      3144
 Slight Injury       0.82      0.83      0.82      3104

      accuracy                           0.83      9374
     macro avg       0.83      0.83      0.83      9374
  weighted avg       0.83      0.83      0.83      9374

DecisionTreeClassifier()
**********************************************************************
0.9789844250053339
**********************************************************************
[[3109   17    0]
 [  80 3054   10]
 [  18   72 3014]]
**********************************************************************
               precision    recall  f1-score   support

  Fatal injury       0.97      0.99      0.98      3126
Serious Injury       0.97      0.97      0.97      3144
 Slight Injury       1.00      0.97      0.98      3104

      accuracy                           0.98      9374
     macro avg       0.98      0.98      0.98      9374
  weighted avg       0.98      0.98      0.98      9374

RandomForestClassifier(n_estimators=10)

<ipython-input-32-a62b6640b181>:4: DataConversionWarning: A column-vector y was
passed when a 1d array was expected. Please change the shape of y to
(n_samples,), for example using ravel().
  i.fit(xo,yo)

**********************************************************************
0.9730104544484744
**********************************************************************
[[3109   17    0]
 [  82 3031   31]
 [  19  104 2981]]
**********************************************************************
               precision    recall  f1-score   support

  Fatal injury       0.97      0.99      0.98      3126
```

Serious Injury	0.96	0.96	0.96	3144
Slight Injury	0.99	0.96	0.97	3104
accuracy			0.97	9374
macro avg	0.97	0.97	0.97	9374
weighted avg	0.97	0.97	0.97	9374

2 Conclusion

Among the various machine learning algorithms examined, decision trees and random forests consistently exhibited superior performance in classifying accident severity. These models effectively leveraged the dataset's rich features to make highly accurate predictions. Incoparating behavioral analysis to predicts driver actions and identify risky behaviour patters leading to accidents. These future devolapments can contribute to more accurate and effective accident classification models, Ultimate aim of this project is helping to improve road safety