```
import pandas as pd
import matplotlib.pyplot as plt


import numpy as np


accidents=pd.read_csv("/content/accidents.csv")
vehicles=pd.read_csv("/content/vehicles.csv")


accidents.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 88624 entries, 0 to 88623
Data columns (total 32 columns):
 #   Column                                       Non-Null Count  Dtype  
---  ------                                       --------------  -----  
 0   Accident_Index                               88624 non-null  object 
 1   Location_Easting_OSGR                        88592 non-null  float64
 2   Location_Northing_OSGR                       88592 non-null  float64
 3   Longitude                                    88592 non-null  float64
 4   Latitude                                     88592 non-null  float64
 5   Police_Force                                 88623 non-null  float64
 6   Accident_Severity                            88623 non-null  float64
 7   Number_of_Vehicles                           88623 non-null  float64
 8   Number_of_Casualties                         88623 non-null  float64
 9   Date                                         88623 non-null  object 
 10  Day_of_Week                                  88623 non-null  float64
 11  Time                                         88622 non-null  object 
 12  Local_Authority_(District)                   88623 non-null  float64
 13  Local_Authority_(Highway)                    88623 non-null  object 
 14  1st_Road_Class                               88623 non-null  float64
 15  1st_Road_Number                              88623 non-null  float64
 16  Road_Type                                    88623 non-null  float64
 17  Speed_limit                                  88623 non-null  float64
 18  Junction_Detail                              88623 non-null  float64
 19  Junction_Control                             88623 non-null  float64
 20  2nd_Road_Class                               88623 non-null  float64
 21  2nd_Road_Number                              88623 non-null  float64
 22  Pedestrian_Crossing-Human_Control            88623 non-null  float64
 23  Pedestrian_Crossing-Physical_Facilities      88623 non-null  float64
 24  Light_Conditions                             88623 non-null  float64
 25  Weather_Conditions                           88623 non-null  float64
 26  Road_Surface_Conditions                      88623 non-null  float64
 27  Special_Conditions_at_Site                   88623 non-null  float64
 28  Carriageway_Hazards                          88623 non-null  float64
 29  Urban_or_Rural_Area                          88623 non-null  float64
 30  Did_Police_Officer_Attend_Scene_of_Accident  88623 non-null  float64
 31  LSOA_of_Accident_Location                    88512 non-null  object 
dtypes: float64(27), object(5)
memory usage: 21.6+ MB
```

```
vehicles.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 46070 entries, 0 to 46069
Data columns (total 22 columns):
 #   Column                              Non-Null Count  Dtype  
---  ------                              --------------  -----  
 0   Accident_Index                      46070 non-null  object 
 1   Vehicle_Reference                   46069 non-null  float64
 2   Vehicle_Type                        46069 non-null  float64
 3   Towing_and_Articulation             46069 non-null  float64
 4   Vehicle_Manoeuvre                   46069 non-null  float64
 5   Vehicle_Location-Restricted_Lane    46069 non-null  float64
 6   Junction_Location                   46069 non-null  float64
 7   Skidding_and_Overturning            46069 non-null  float64
 8   Hit_Object_in_Carriageway           46069 non-null  float64
 9   Vehicle_Leaving_Carriageway         46069 non-null  float64
 10  Hit_Object_off_Carriageway          46069 non-null  float64
 11  1st_Point_of_Impact                 46069 non-null  float64
 12  Was_Vehicle_Left_Hand_Drive?        46069 non-null  float64
 13  Journey_Purpose_of_Driver           46069 non-null  float64
 14  Sex_of_Driver                       46069 non-null  float64
 15  Age_of_Driver                       46069 non-null  float64
 16  Age_Band_of_Driver                  46069 non-null  float64
 17  Engine_Capacity_(CC)                46069 non-null  float64
 18  Propulsion_Code                     46069 non-null  float64
 19  Age_of_Vehicle                      46069 non-null  float64
 20  Driver_IMD_Decile                   46069 non-null  float64
 21  Driver_Home_Area_Type               46069 non-null  float64
dtypes: float64(21), object(1)
memory usage: 7.7+ MB
```

```
accidents["Accident_Severity"].unique()
#gives unique values in the column
```

```
array([ 2.,  3.,  1., nan])
```

```
#Combining two datasets
import pandas as pd
data=pd.merge(accidents,vehicles,on="Accident_Index",how='outer')
```

```
data.head()
```

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 109135 entries, 0 to 109134
Data columns (total 53 columns):
 #   Column                                       Non-Null Count   Dtype  
---  ------                                       --------------   -----  
 0   Accident_Index                               109135 non-null  object 
 1   Location_Easting_OSGR                        109102 non-null  float64
 2   Location_Northing_OSGR                       109102 non-null  float64
 3   Longitude                                    109102 non-null  float64
 4   Latitude                                     109102 non-null  float64
 5   Police_Force                                 109133 non-null  float64
 6   Accident_Severity                            109133 non-null  float64
 7   Number_of_Vehicles                           109133 non-null  float64
 8   Number_of_Casualties                         109133 non-null  float64
 9   Date                                         109133 non-null  object 
 10  Day_of_Week                                  109133 non-null  float64
 11  Time                                         109132 non-null  object 
 12  Local_Authority_(District)                   109133 non-null  float64
 13  Local_Authority_(Highway)                    109133 non-null  object 
 14  1st_Road_Class                               109133 non-null  float64
 15  1st_Road_Number                              109133 non-null  float64
 16  Road_Type                                    109133 non-null  float64
 17  Speed_limit                                  109133 non-null  float64
 18  Junction_Detail                              109133 non-null  float64
 19  Junction_Control                             109133 non-null  float64
 20  2nd_Road_Class                               109133 non-null  float64
 21  2nd_Road_Number                              109133 non-null  float64
 22  Pedestrian_Crossing-Human_Control            109133 non-null  float64
 23  Pedestrian_Crossing-Physical_Facilities      109133 non-null  float64
 24  Light_Conditions                             109133 non-null  float64
 25  Weather_Conditions                           109133 non-null  float64
 26  Road_Surface_Conditions                      109133 non-null  float64
 27  Special_Conditions_at_Site                   109133 non-null  float64
 28  Carriageway_Hazards                          109133 non-null  float64
 29  Urban_or_Rural_Area                          109133 non-null  float64
 30  Did_Police_Officer_Attend_Scene_of_Accident  109133 non-null  float64
 31  LSOA_of_Accident_Location                    108975 non-null  object 
 32  Vehicle_Reference                            46069 non-null   float64
 33  Vehicle_Type                                 46069 non-null   float64
 34  Towing_and_Articulation                      46069 non-null   float64
 35  Vehicle_Manoeuvre                            46069 non-null   float64
 36  Vehicle_Location-Restricted_Lane             46069 non-null   float64
 37  Junction_Location                            46069 non-null   float64
 38  Skidding_and_Overturning                     46069 non-null   float64
 39  Hit_Object_in_Carriageway                    46069 non-null   float64
 40  Vehicle_Leaving_Carriageway                  46069 non-null   float64
 41  Hit_Object_off_Carriageway                   46069 non-null   float64
 42  1st_Point_of_Impact                          46069 non-null   float64
 43  Was_Vehicle_Left_Hand_Drive?                 46069 non-null   float64
 44  Journey_Purpose_of_Driver                    46069 non-null   float64
 45  Sex_of_Driver                                46069 non-null   float64
 46  Age_of_Driver                                46069 non-null   float64
 47  Age_Band_of_Driver                           46069 non-null   float64
```

```
 48  Engine_Capacity_(CC)                    46069 non-null   float64
 49  Propulsion_Code                         46069 non-null   float64
 50  Age_of_Vehicle                          46069 non-null   float64
 51  Driver_IMD_Decile                       46069 non-null   float64
 52  Driver_Home_Area_Type                   46069 non-null   float64
```

```
Description of variables:
    Accident_Severity:1-Fatal,2-Serious,3-Slight
    Road_Type:1-Roundabout,2-One way street,3-Dual carriageway,
        6-Single carriageway,7-Slip road,9-Unknown
    Road_Surface_Conditions:1-Dry,2-Wet or damp,3-Snow,4-Frost or ice,
       5-Flood over 3cm. deep
    Skidding_and_Overturning:0-None,1-Skidded,2-Skidded and overturned,
       5-Overturned
    Weather_Conditions:1-Fine no high winds,2-Raining no high winds,
         3-Snowing no high winds,4-Fine + high winds,5-Raining + high winds,
         6-Snowing + high winds,7-Fog or mist,8-Other,9-Unknown
```

Description of variables: Accident_Severity:1-Fatal,2-Serious,3-Slight Road_Type:1-Roundabout,2-One way street,3-Dual carriageway, 6-Single carriageway,7-Slip road,9-Unknown Road_Surface_Conditions:1-Dry,2-Wet or damp,3-Snow,4-Frost or ice, 5-Flood over 3cm. deep Skidding_and_Overturning:0-None,1-Skidded,2-Skidded and overturned, 5-Overturned Weather_Conditions:1-Fine no high winds,2-Raining no high winds, 3-Snowing no high winds,4-Fine + high winds,5-Raining + high winds, 6-Snowing + high winds,7-Fog or mist,8-Other,9-Unknown

+ Code + Text

```
#Filling the missing values by using dropna() method.
#Our dataset contains null values in the format -1 and NAN
for column in data.columns:
    data=(data[data[column]!=-1])
data.dropna(inplace=True)
```

```
data.isnull().sum()
#We can see that all columns have no null values.
```

```
Accident_Index                          0
Location_Easting_OSGR                   0
Location_Northing_OSGR                  0
Longitude                               0
Latitude                                0
Police_Force                            0
Accident_Severity                       0
Number_of_Vehicles                      0
Number_of_Casualties                    0
Date                                    0
Day_of_Week                             0
Time                                    0
Local_Authority_(District)              0
Local_Authority_(Highway)               0
1st_Road_Class                          0
1st_Road_Number                         0
Road_Type                               0
Speed_limit                             0
```

```
Junction_Detail                                0
Junction_Control                               0
2nd_Road_Class                                 0
2nd_Road_Number                                0
Pedestrian_Crossing-Human_Control              0
Pedestrian_Crossing-Physical_Facilities        0
Light_Conditions                               0
Weather_Conditions                             0
Road_Surface_Conditions                        0
Special_Conditions_at_Site                     0
Carriageway_Hazards                            0
Urban_or_Rural_Area                            0
Did_Police_Officer_Attend_Scene_of_Accident    0
LSOA_of_Accident_Location                      0
Vehicle_Reference                              0
Vehicle_Type                                   0
Towing_and_Articulation                        0
Vehicle_Manoeuvre                              0
Vehicle_Location-Restricted_Lane               0
Junction_Location                              0
Skidding_and_Overturning                       0
Hit_Object_in_Carriageway                      0
Vehicle_Leaving_Carriageway                    0
Hit_Object_off_Carriageway                     0
1st_Point_of_Impact                            0
Was_Vehicle_Left_Hand_Drive?                   0
Journey_Purpose_of_Driver                      0
Sex_of_Driver                                  0
Age_of_Driver                                  0
Age_Band_of_Driver                             0
Engine_Capacity_(CC)                           0
Propulsion_Code                                0
Age_of_Vehicle                                 0
Driver_IMD_Decile                              0
Driver_Home_Area_Type                          0
dtype: int64
```

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 14027 entries, 9 to 46068
Data columns (total 53 columns):
 #   Column                                      Non-Null Count  Dtype  
---  ------                                      --------------  -----  
 0   Accident_Index                              14027 non-null  object 
 1   Location_Easting_OSGR                       14027 non-null  float64
 2   Location_Northing_OSGR                      14027 non-null  float64
 3   Longitude                                   14027 non-null  float64
 4   Latitude                                    14027 non-null  float64
 5   Police_Force                                14027 non-null  float64
 6   Accident_Severity                           14027 non-null  float64
 7   Number_of_Vehicles                          14027 non-null  float64
 8   Number_of_Casualties                        14027 non-null  float64
 9   Date                                        14027 non-null  object 
 10  Day_of_Week                                 14027 non-null  float64
```

```
 11  Time                                          14027 non-null  object
 12  Local_Authority_(District)                    14027 non-null  float64
 13  Local_Authority_(Highway)                     14027 non-null  object
 14  1st_Road_Class                                14027 non-null  float64
 15  1st_Road_Number                               14027 non-null  float64
 16  Road_Type                                     14027 non-null  float64
 17  Speed_limit                                   14027 non-null  float64
 18  Junction_Detail                               14027 non-null  float64
 19  Junction_Control                              14027 non-null  float64
 20  2nd_Road_Class                                14027 non-null  float64
 21  2nd_Road_Number                               14027 non-null  float64
 22  Pedestrian_Crossing-Human_Control             14027 non-null  float64
 23  Pedestrian_Crossing-Physical_Facilities       14027 non-null  float64
 24  Light_Conditions                              14027 non-null  float64
 25  Weather_Conditions                            14027 non-null  float64
 26  Road_Surface_Conditions                       14027 non-null  float64
 27  Special_Conditions_at_Site                    14027 non-null  float64
 28  Carriageway_Hazards                           14027 non-null  float64
 29  Urban_or_Rural_Area                           14027 non-null  float64
 30  Did_Police_Officer_Attend_Scene_of_Accident   14027 non-null  float64
 31  LSOA_of_Accident_Location                     14027 non-null  object
 32  Vehicle_Reference                             14027 non-null  float64
 33  Vehicle_Type                                  14027 non-null  float64
 34  Towing_and_Articulation                       14027 non-null  float64
 35  Vehicle_Manoeuvre                             14027 non-null  float64
 36  Vehicle_Location-Restricted_Lane              14027 non-null  float64
 37  Junction_Location                             14027 non-null  float64
 38  Skidding_and_Overturning                      14027 non-null  float64
 39  Hit_Object_in_Carriageway                     14027 non-null  float64
 40  Vehicle_Leaving_Carriageway                   14027 non-null  float64
 41  Hit_Object_off_Carriageway                    14027 non-null  float64
 42  1st_Point_of_Impact                           14027 non-null  float64
 43  Was_Vehicle_Left_Hand_Drive?                  14027 non-null  float64
 44  Journey_Purpose_of_Driver                     14027 non-null  float64
 45  Sex_of_Driver                                 14027 non-null  float64
 46  Age_of_Driver                                 14027 non-null  float64
 47  Age_Band_of_Driver                            14027 non-null  float64
 48  Engine_Capacity_(CC)                          14027 non-null  float64
 49  Propulsion_Code                               14027 non-null  float64
 50  Age_of_Vehicle                                14027 non-null  float64
 51  Driver_IMD_Decile                             14027 non-null  float64
 52  Driver_Home_Area_Type                         14027 non-null  float64
```

```
data['Accident_Severity'].unique()
#The column 'Accident_Severity' has 3 unique values:1,2,3 indicating Fatal,Severe and Slight Accident
```

```
array([3., 2., 1.])
```

```
data['Road_Type'].unique()
```

```
array([6., 2., 3., 1., 9., 7.])
```

```
data['Road_Surface_Conditions'].unique()
```

```
array([1., 2., 4., 3., 5.])
```

```
data['Skidding_and_Overturning'].unique()
```

```
array([0., 1., 5., 2.])
```

```
import pandas as pd
data.to_csv("Accidents_data.csv")
```

```
data['Accident_Severity'].unique()
```

```
array([3., 2., 1.])
```

```
data['Weather_Conditions'].unique()
```

```
array([1., 2., 3., 8., 4., 5., 9., 7., 6.])
```

```
data[(data['Accident_Severity']!=1)].count()
```

```
Accident_Index                                 13957
Location_Easting_OSGR                          13957
Location_Northing_OSGR                         13957
Longitude                                      13957
Latitude                                       13957
Police_Force                                   13957
Accident_Severity                              13957
Number_of_Vehicles                             13957
Number_of_Casualties                           13957
Date                                           13957
Day_of_Week                                    13957
Time                                           13957
Local_Authority_(District)                     13957
Local_Authority_(Highway)                      13957
1st_Road_Class                                 13957
1st_Road_Number                                13957
Road_Type                                      13957
Speed_limit                                    13957
Junction_Detail                                13957
Junction_Control                               13957
2nd_Road_Class                                 13957
2nd_Road_Number                                13957
Pedestrian_Crossing-Human_Control              13957
Pedestrian_Crossing-Physical_Facilities        13957
Light_Conditions                               13957
Weather_Conditions                             13957
Road_Surface_Conditions                        13957
Special_Conditions_at_Site                     13957
Carriageway_Hazards                            13957
Urban_or_Rural_Area                            13957
Did_Police_Officer_Attend_Scene_of_Accident    13957
LSOA_of_Accident_Location                      13957
Vehicle_Reference                              13957
Vehicle_Type                                   13957
Towing_and_Articulation                        13957
```

```
Vehicle_Manoeuvre                          13957
Vehicle_Location-Restricted_Lane           13957
Junction_Location                          13957
Skidding_and_Overturning                   13957
Hit_Object_in_Carriageway                  13957
Vehicle_Leaving_Carriageway                13957
Hit_Object_off_Carriageway                 13957
1st_Point_of_Impact                        13957
Was_Vehicle_Left_Hand_Drive?               13957
Journey_Purpose_of_Driver                  13957
Sex_of_Driver                              13957
Age_of_Driver                              13957
Age_Band_of_Driver                         13957
Engine_Capacity_(CC)                       13957
Propulsion_Code                            13957
Age_of_Vehicle                             13957
Driver_IMD_Decile                          13957
Driver_Home_Area_Type                      13957
dtype: int64
```

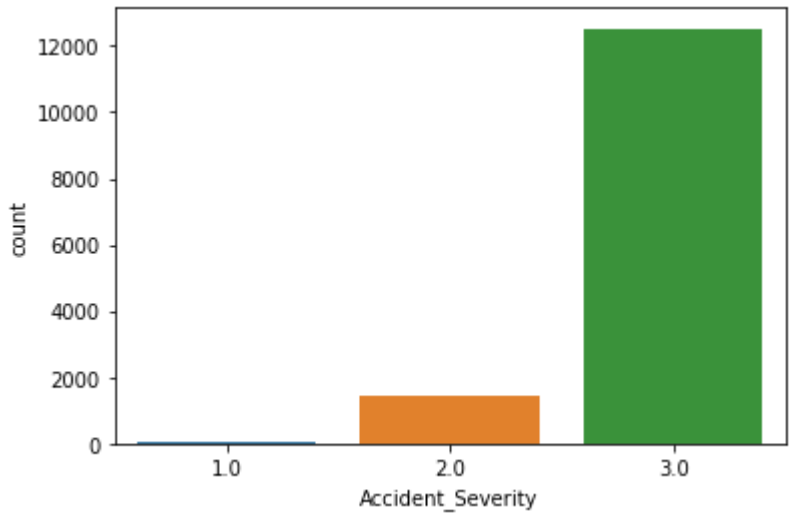
Visualization:

How many accidents are severe to the most?

```
import seaborn as sns
```

```
sns.countplot(data['Accident_Severity'])
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pass th
  FutureWarning
<matplotlib.axes._subplots.AxesSubplot at 0x7f94081ed110>
```



From the plot,we can see that most of the accidents are slight only.

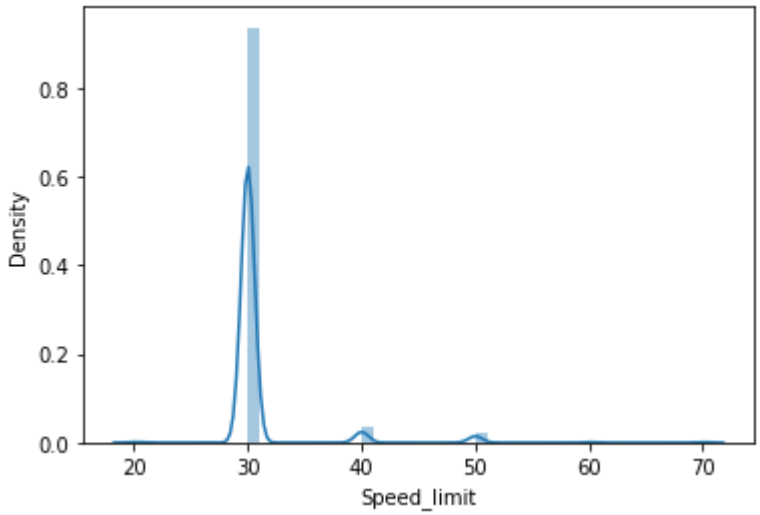
```
data[data['Accident_Severity']==1].count()
```

```
Accident_Index                                 70
Location_Easting_OSGR                          70
Location_Northing_OSGR                         70
Longitude                                      70
Latitude                                       70
Police_Force                                   70
Accident_Severity                              70
Number_of_Vehicles                             70
Number_of_Casualties                           70
Date                                           70
Day_of_Week                                    70
Time                                           70
Local_Authority_(District)                     70
Local_Authority_(Highway)                      70
1st_Road_Class                                 70
1st_Road_Number                                70
Road_Type                                      70
Speed_limit                                    70
Junction_Detail                                70
Junction_Control                               70
2nd_Road_Class                                 70
2nd_Road_Number                                70
Pedestrian_Crossing-Human_Control              70
Pedestrian_Crossing-Physical_Facilities        70
Light_Conditions                               70
Weather_Conditions                             70
Road_Surface_Conditions                        70
Special_Conditions_at_Site                     70
Carriageway_Hazards                            70
Urban_or_Rural_Area                            70
Did_Police_Officer_Attend_Scene_of_Accident    70
LSOA_of_Accident_Location                      70
Vehicle_Reference                              70
Vehicle_Type                                   70
Towing_and_Articulation                        70
Vehicle_Manoeuvre                              70
Vehicle_Location-Restricted_Lane               70
Junction_Location                              70
Skidding_and_Overturning                       70
Hit_Object_in_Carriageway                      70
Vehicle_Leaving_Carriageway                    70
Hit_Object_off_Carriageway                     70
1st_Point_of_Impact                            70
Was_Vehicle_Left_Hand_Drive?                   70
Journey_Purpose_of_Driver                      70
Sex_of_Driver                                  70
Age_of_Driver                                  70
Age_Band_of_Driver                             70
Engine_Capacity_(CC)                           70
Propulsion_Code                                70
Age_of_Vehicle                                 70
Driver_IMD_Decile                              70
Driver_Home_Area_Type                          70
dtype: int64
```

Inference: From this we can see that 70 persons from our data have been met with most severe accident which endangered their life.

```
sns.distplot(data['Speed_limit'])
```

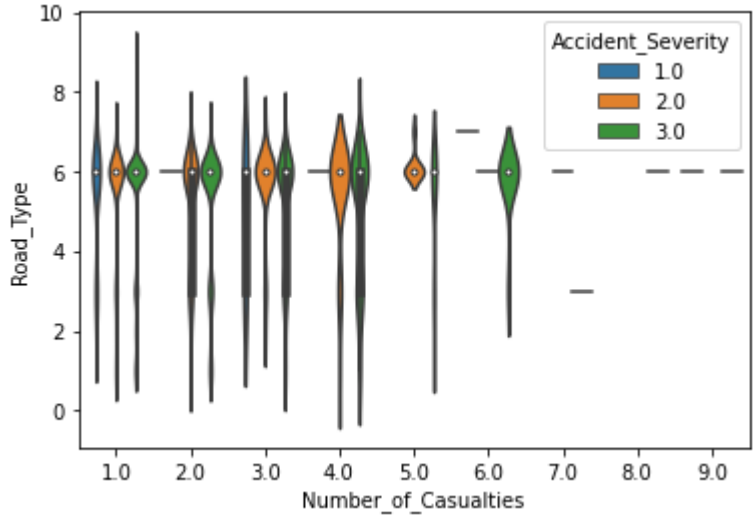
```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning: `di
  warnings.warn(msg, FutureWarning)
<matplotlib.axes._subplots.AxesSubplot at 0x7f94074f4d10>
```



In our dataset,the Spped limit is distributed in the range of 20-70.

```
sns.violinplot(data['Number_of_Casualties'],data['Road_Type'],hue=data['Accident_Severity'])
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pass th
  FutureWarning
<matplotlib.axes._subplots.AxesSubplot at 0x7f9400ad2750>
```



Identifying dependent and independent features:

```
y=data['Accident_Severity']
x=data[['Longitude','Latitude','Number_of_Vehicles','Number_of_Casualties','Road_Type','Speed_limit','Weather_Conditions','Road_Surface_Condition
```

Splitting the dataset into train and test:

```
from sklearn.model_selection import train_test_split
```

```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=101)
```

Use Ensemble learning VotingClassifier technique:

```
#Use ensemble learning:
from sklearn.ensemble import VotingClassifier
from sklearn.linear_model import LogisticRegression
model1=LogisticRegression(max_iter=100)
from sklearn.tree import DecisionTreeClassifier
model2=DecisionTreeClassifier(random_state=101)
from sklearn import svm
#Create a svm Classifier
model3 = svm.SVC(kernel='linear') # Linear Kernel
model=VotingClassifier(estimators=[('log',model1),('dt',model2),('svm',model3)],voting='hard')
model.fit(x_train,y_train)
pred=model.predict(x_test)
```

```
/usr/local/lib/python3.7/dist-packages/sklearn/linear_model/_logistic.py:818: Convergenc
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG,
```

```
import pandas as pd
pred=pd.DataFrame(pred)
pred
```

	0
0	3.0
1	3.0
2	3.0
3	3.0
4	3.0

Evaluating our model:

2801 3.0

```
from sklearn.metrics import accuracy_score
accuracy_score(y_test,pred)
```

```
0.8970064148253742
```

2804 3.0

Inference: We get accuracy to be 90% approximately => our model works very well.

2806 rows × 1 columns

Saving the trained model:

```
import pickle


filename="accident_severity_model.pkl"
pickle.dump(model,open(filename,'wb'))
```

Loading the model:

```
loaded_model=pickle.load(open('accident_severity_model.pkl','rb'))
```

```
loaded_model
```

```
VotingClassifier(estimators=[('log', LogisticRegression()),
                             ('dt', DecisionTreeClassifier(random_state=101)),
                             ('svm', SVC(kernel='linear'))])
```