TASK 04

Analyze traffic accident data to identify patterns related to road conditions, weather, and time of day. Visualize accident hotspots and contributing factors.

Accident Dataset

Explorer (Ctrl+Shift+E)

C: > Users > gaura > Desktop > crf > intership > 123skillcraft > task4 > SCT_DS_4.ipynb > Loading Libraries and Data > #importing libraries

Generate | Code | Markdown | Run All | Clear All Outputs | Outline | ... | Select Kernel

TASK 4 - BY Gaurav Singh Yadav

Task 4 :- Analyze traffic accident data to identify patterns related to road conditions, weather, and time of day. Visualize accident hotspots and contributing factors.

Loading Libraries and Data

```
#importing libraries
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings('ignore')
```

[35] Python

```
#load and read the file
df=pd.read_csv("RTA Dataset.csv")
df.head()
```

[36] Python

SCT_DS_4.ipynb

C: > Users > gaura > Desktop > crf > intership > 123skillcraft > task4 > SCT_DS_4.ipynb > Loading Libraries and Data > #shape/ size of the data

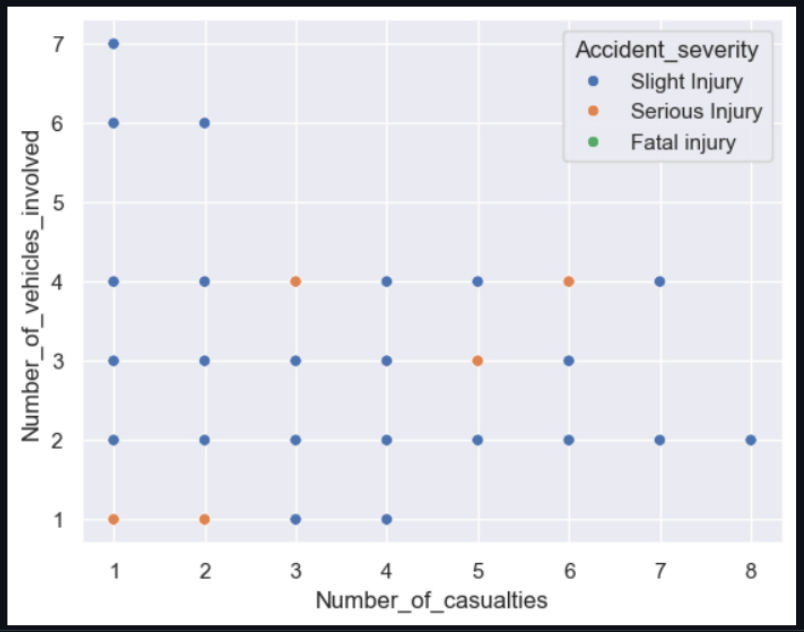
Generate | Code | Markdown | Run All | Clear All Outputs | Outline | Select Kernel

Data Visualization

```
#plotting relationship between Number_of_casualties and Number_of_vehicles_involved
sns.scatterplot(x=df['Number_of_casualties'], y=df['Number_of_vehicles_involved'], hue=df['Accident_severity'])
```

[121] Python

```
<Axes: xlabel='Number_of_casualties', ylabel='Number_of_vehicles_involved'>
```



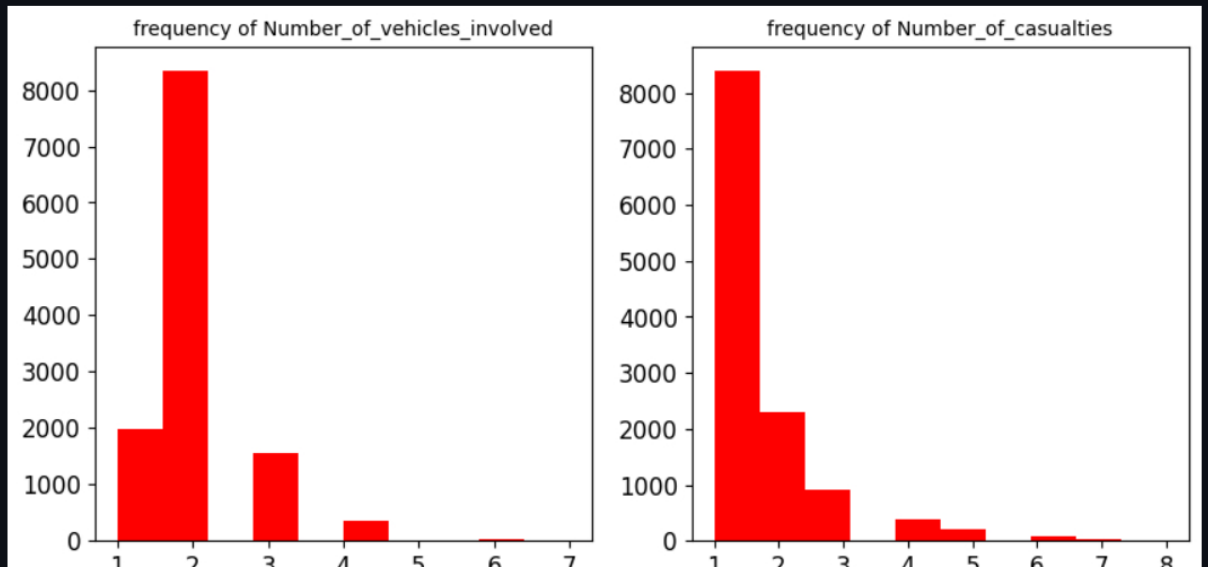
SCT_DS_4.ipynb

C: > Users > gaura > Desktop > crf > intership > 123skillcraft > task4 > SCT_DS_4.ipynb > Loading Libraries and Data > #shape/ size of the data

Generate | Code | Markdown | Run All | Clear All Outputs | Outline | Select Kernel

```python
#distribution for numerical columns
plt.figure(figsize=(10,10))
plotnumber = 1
for i in numerical:
    if plotnumber <= df.shape[1]:
        ax1 = plt.subplot(2,2,plotnumber)
        plt.hist(df[i],color='red')
        plt.xticks(fontsize=12)
        plt.yticks(fontsize=12)
        plt.title('frequency of '+i, fontsize=10)
    plotnumber +=1
```

[63] Python



SCT_DS_4.ipynb

C: > Users > gaura > Desktop > crf > intership > 123skillcraft > task4 > SCT_DS_4.ipynb > Prediction > #KNN model alg

Generate | Code | Markdown | Run All | Clear All Outputs | Outline | ... | Select Kernel

KNN Model Creation

Prediction

```python
#KNN model alg
from sklearn.neighbors import KNeighborsClassifier
model_KNN=KNeighborsClassifier(n_neighbors=5)
model_KNN.fit(x_train,y_train)
```

[108] Python

KNeighborsClassifier

▼ Parameters

n_neighbors	5
weights	'uniform'
algorithm	'auto'
leaf_size	30
p	2
metric	'minkowski'
metric_params	None
n_jobs	None

SCT_DS_4.ipynb

C: > Users > gaura > Desktop > crf > intership > 123skillcraft > task4 > SCT_DS_4.ipynb > Prediction > #KNN model alg

Generate | Code | Markdown | Run All | Clear All Outputs | Outline | ... | Select Kernel

Checking Accuracy, Classification Report, Confusion Matrix

```
[116]
from sklearn.metrics import confusion_matrix, classification_report, accuracy_score, ConfusionMatrixDisplay
```

Python

Classification Report

```
[117]
report_KNN = classification_report(y_test,y_pred)
print(report_KNN)
```

Python

```
                precision    recall  f1-score   support

   Fatal injury       0.79      1.00      0.88      3126
Serious Injury       0.64      0.90      0.75      3144
 Slight Injury       0.96      0.30      0.46      3104

      accuracy                           0.74      9374
     macro avg       0.80      0.73      0.70      9374
  weighted avg       0.80      0.74      0.70      9374
```

Accuracy Score

```
accuracy_KNN = accuracy_score(y_test,y_pred)
```