Assignment 8 th september

1.Take car crashes dataset from seaborn library

2.load the dataset

3.data visualiation

4.Inference is must for each and every graph

5.Submit it by wednesday in html format

Feedback - https://forms.gle/7vFfvANDVfvDxxW28

## 1.import the necessary libraries

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

## 2.import the dataset

```
dataset=pd.read_csv("car_crashes.csv")
```

```
dataset.head(10)
```

| | total | speeding | alcohol | not_distracted | no_previous | ins_premium | ins |
|---|---|---|---|---|---|---|---|
| **0** | 18.8 | 7.332 | 5.640 | 18.048 | 15.040 | 784.55 | |
| **1** | 18.1 | 7.421 | 4.525 | 16.290 | 17.014 | 1053.48 | |
| **2** | 18.6 | 6.510 | 5.208 | 15.624 | 17.856 | 899.47 | |
| **3** | 22.4 | 4.032 | 5.824 | 21.056 | 21.280 | 827.34 | |
| **4** | 12.0 | 4.200 | 3.360 | 10.920 | 10.680 | 878.41 | |
| **5** | 13.6 | 5.032 | 3.808 | 10.744 | 12.920 | 835.50 | |
| **6** | 10.8 | 4.968 | 3.888 | 9.396 | 8.856 | 1068.73 | |
| **7** | 16.2 | 6.156 | 4.860 | 14.094 | 16.038 | 1137.87 | |
| **8** | 5.9 | 2.006 | 1.593 | 5.900 | 5.900 | 1273.89 | |

```
dataset.tail()
```

| | total | speeding | alcohol | not_distracted | no_previous | ins_premium | in |
|---|---|---|---|---|---|---|---|
| **46** | 12.7 | 2.413 | 3.429 | 11.049 | 11.176 | 768.95 | |
| **47** | 10.6 | 4.452 | 3.498 | 8.692 | 9.116 | 890.03 | |
| **48** | 23.8 | 8.092 | 6.664 | 23.086 | 20.706 | 992.61 | |
| **49** | 13.8 | 4.968 | 4.554 | 5.382 | 11.592 | 670.31 | |

```
dataset.shape
```

```
(51, 8)
```

```
dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 51 entries, 0 to 50
Data columns (total 8 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   total           51 non-null     float64
 1   speeding        51 non-null     float64
 2   alcohol         51 non-null     float64
 3   not_distracted  51 non-null     float64
 4   no_previous     51 non-null     float64
```

```
 5   ins_premium     51 non-null     float64
 6   ins_losses      51 non-null     float64
 7   abbrev          51 non-null     object
dtypes: float64(7), object(1)
memory usage: 3.3+ KB
```

```
dataset.describe()
```

| | total | speeding | alcohol | not_distracted | no_previous | ins_pr |
|---|---|---|---|---|---|---|
| **count** | 51.000000 | 51.000000 | 51.000000 | 51.000000 | 51.000000 | 51.0 |
| **mean** | 15.790196 | 4.998196 | 4.886784 | 13.573176 | 14.004882 | 886.9 |
| **std** | 4.122002 | 2.017747 | 1.729133 | 4.508977 | 3.764672 | 178.2 |
| **min** | 5.900000 | 1.792000 | 1.593000 | 1.760000 | 5.900000 | 641.9 |
| **25%** | 12.750000 | 3.766500 | 3.894000 | 10.478000 | 11.348000 | 768.4 |
| **50%** | 15.600000 | 4.608000 | 4.554000 | 13.857000 | 13.775000 | 858.9 |
| **75%** | 18.500000 | 6.439000 | 5.604000 | 16.140000 | 16.755000 | 1007.9 |

## HEATMAP

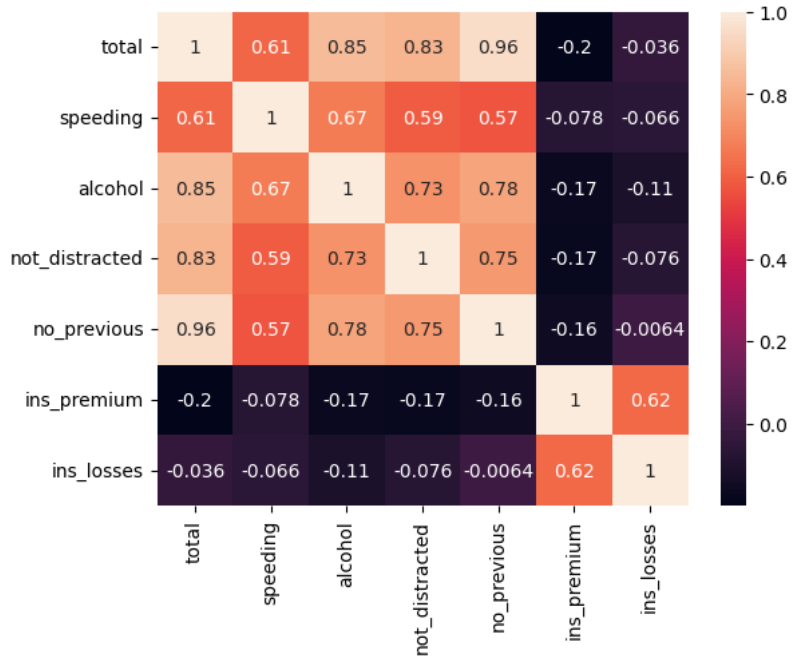
```
corr=dataset.corr()
corr
```

```
<ipython-input-152-f22ca9e9dc13>:1: FutureWarning: The default value of numeric_only
  corr=dataset.corr()
```

| | total | speeding | alcohol | not_distracted | no_previous | ins_premium |
|---|---|---|---|---|---|---|
| **total** | 1.000000 | 0.611548 | 0.852613 | 0.827560 | 0.956179 | -0.199702 |
| **speeding** | 0.611548 | 1.000000 | 0.669719 | 0.588010 | 0.571976 | -0.077675 |
| **alcohol** | 0.852613 | 0.669719 | 1.000000 | 0.732816 | 0.783520 | -0.170612 |
| **not_distracted** | 0.827560 | 0.588010 | 0.732816 | 1.000000 | 0.747307 | -0.174856 |
| **no_previous** | 0.956179 | 0.571976 | 0.783520 | 0.747307 | 1.000000 | -0.156895 |
| **ins_premium** | -0.199702 | -0.077675 | -0.170612 | -0.174856 | -0.156895 | 1.000000 |
| **ins_losses** | -0.036011 | -0.065928 | -0.112547 | -0.075970 | -0.006359 | 0.623116 |

```
#plt.subplots(figsize=(10,10))
sns.heatmap(corr,annot=True)
```

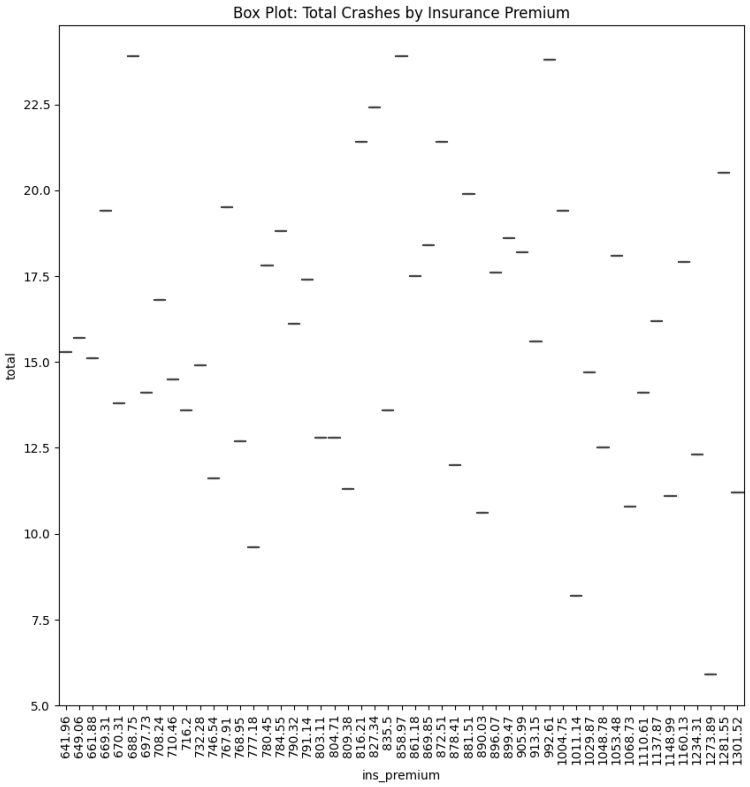
```
<Axes: >
```



.

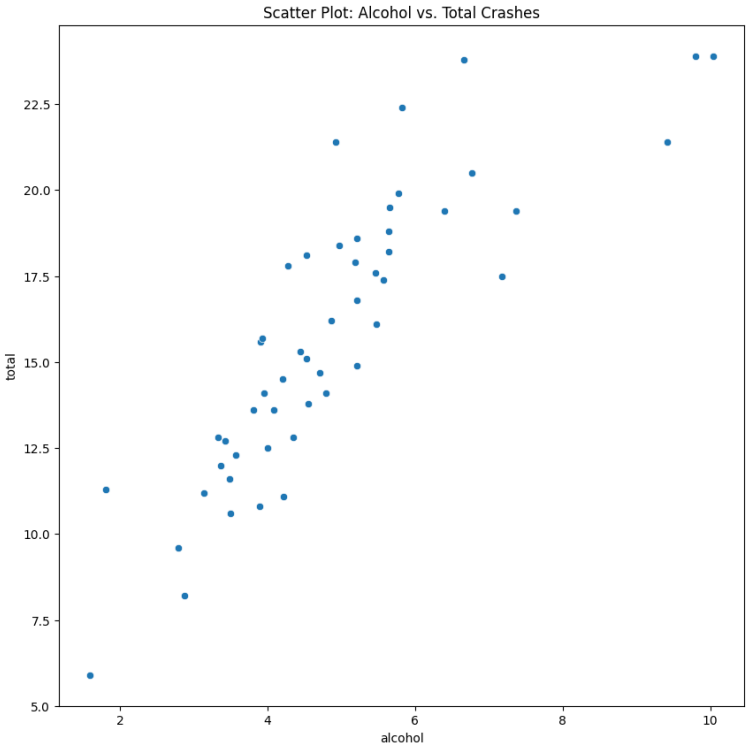
## BOX-PLOT

```
# Box plot of "total" crashes by "ins_premium"
plt.subplots(figsize=(10,10))
sns.boxplot(data=dataset, x="ins_premium", y="total")
plt.title("Box Plot: Total Crashes by Insurance Premium")
plt.xticks(rotation=90)
plt.show()
```



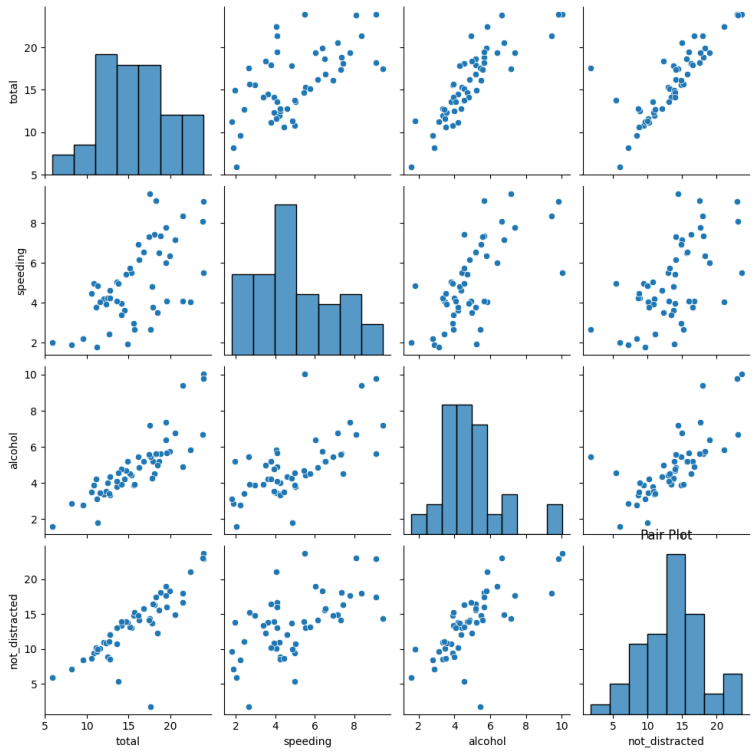

## SCATTER PLOT

```
# Scatter plot between "alcohol" and "total"
plt.subplots(figsize=(10, 10))
sns.scatterplot(data=dataset, x="alcohol", y="total")
plt.title("Scatter Plot: Alcohol vs. Total Crashes")
plt.show()
```




## PAIR PLOT

```
# Pair plot for selected columns
sns.pairplot(dataset[['total', 'speeding', 'alcohol', 'not_distracted']])
plt.title("Pair Plot")
plt.show()
```




✓ 7s completed at 12:04 AM