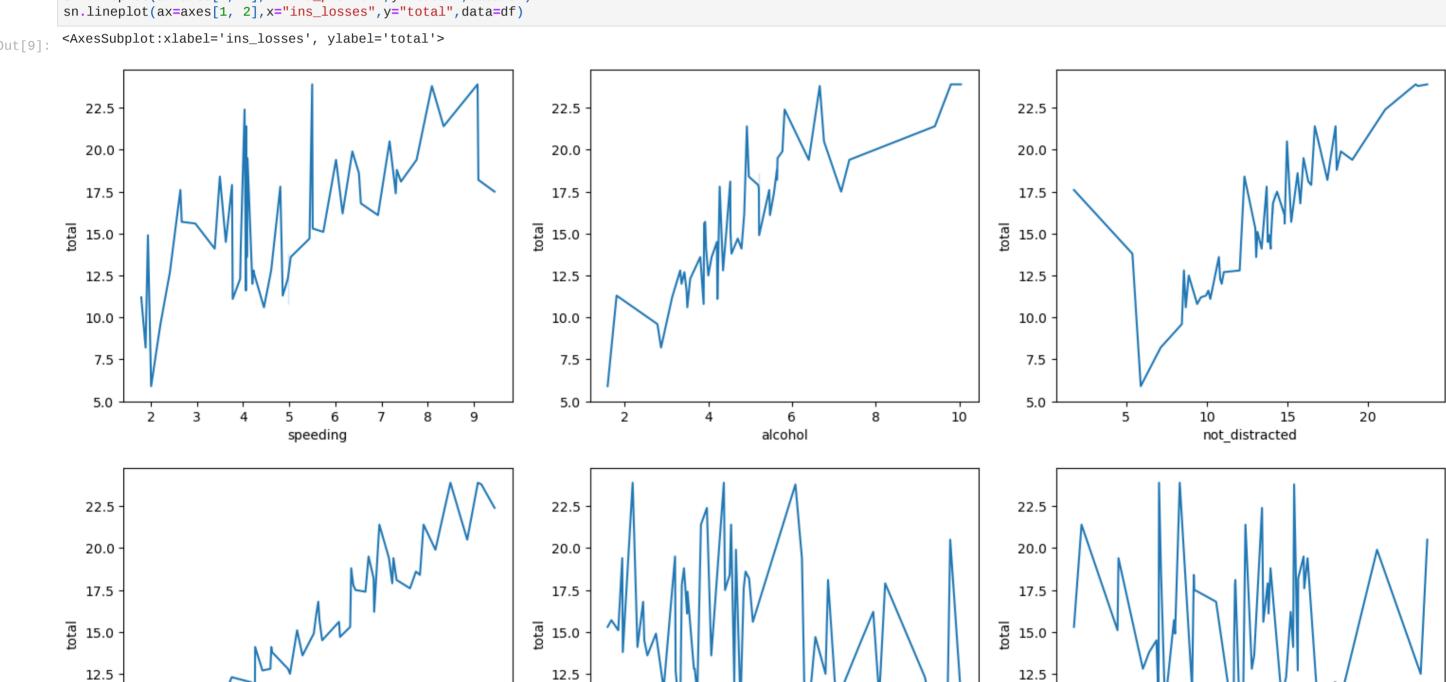
## 1.importing seaborn library import seaborn as sn from matplotlib import pyplot as plt loading the car\_crashes dataset which is present in seaborn datasets df=sn.load\_dataset('car\_crashes') df.head() Out[2]: total speeding alcohol not\_distracted no\_previous ins\_premium ins\_losses abbrev 5.640 18.048 15.040 145.08 $\mathsf{AL}$ **0** 18.8 7.332 784.55 **1** 18.1 7.421 4.525 16.290 17.014 1053.48 133.93 ΑK **2** 18.6 6.510 15.624 17.856 899.47 110.35 5.208 ΑZ **3** 22.4 4.032 5.824 21.056 21.280 827.34 142.39 AR **4** 12.0 4.200 3.360 10.920 10.680 878.41 165.63 CA getting information about the dataset In [3]: df.info() <class 'pandas.core.frame.DataFrame'> RangeIndex: 51 entries, 0 to 50 Data columns (total 8 columns): Non-Null Count Dtype Column total 51 non-null float64 0 speeding 51 non-null float64 1 2 alcohol 51 non-null float64 not\_distracted 51 non-null float64 3 51 non-null float64 4 no\_previous 51 non-null float64 5 ins\_premium ins\_losses float64 6 51 non-null abbrev 51 non-null object dtypes: float64(7), object(1) memory usage: 3.3+ KB In [4]: fig, axes = plt.subplots(2, 3,figsize=(18, 10)) sn.scatterplot(ax=axes[0, 0], x='total', y='speeding', data=df) sn.scatterplot(ax=axes[0,1], x='total', y='alcohol', data=df) sn.scatterplot(ax=axes[0,2],x='total',y='not\_distracted',data=df) sn.scatterplot(ax=axes[1,0], x='total', y='no\_previous', data=df) sn.scatterplot(ax=axes[1,1], x='total', y='ins\_premium', data=df) sn.scatterplot(ax=axes[1,2], x='total', y='ins\_losses', data=df) <AxesSubplot:xlabel='total', ylabel='ins\_losses'> 10 9 20 8 8 7 not\_distracted 10 speeding 2 alcohol 5.0 7.5 10.0 12.5 15.0 17.5 20.0 22.5 5.0 7.5 10.0 12.5 15.0 17.5 20.0 22.5 5.0 7.5 10.0 12.5 15.0 17.5 20.0 22.5 total total total 22 200 1300 20 180 1200 18 1100 160 16 no\_previous ins\_premium losses 1000 900 120 10 800 100 8 700 80 12.5 17.5 20.0 22.5 12.5 12.5 17.5 5.0 7.5 10.0 15.0 5.0 7.5 10.0 15.0 17.5 20.0 22.5 5.0 7.5 10.0 15.0 20.0 22.5 total total total inference from above plots we can observe that by increasing speed, alcohol consumption, not\_distracted rate and no previous accidents rates will increase the total rate of accidents. In [9]: fig, axes = plt.subplots(2, 3, figsize=(18, 10)) sn.lineplot(ax=axes[0, 0], x="speeding", y="total", data=df) sn.lineplot(ax=axes[0, 1], x="alcohol", y="total", data=df) sn.lineplot(ax=axes[0, 2], x="not\_distracted", y="total", data=df) sn.lineplot(ax=axes[1, 0], x="no\_previous", y="total", data=df) sn.lineplot(ax=axes[1, 1], x="ins\_premium", y="total", data=df) sn.lineplot(ax=axes[1, 2], x="ins\_losses", y="total", data=df) <AxesSubplot:xlabel='ins\_losses', ylabel='total'> Out[9]:



10.0

7.5

5.0

80

100

120

140

ins\_losses

160

180

200

warnings.warn(msg, FutureWarning)
C:\Users\Bhagya Sri\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please a dapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).

warnings.warn(msg, FutureWarning)
C:\Users\Bhagya Sri\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please a

10.0

7.5

5.0

700

dapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).

dapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).

dapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).

dapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).

800

from the above graphs we can see that line plots of total vs speed, alcohol, not-distracted, no-previous accidents. we can observe that at some points the slope is increasing and at some points it is decreasing.

900

C:\Users\Bhagya Sri\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please a

C:\Users\Bhagya Sri\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please a

C:\Users\Bhagya Sri\anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please a

ins\_premium

1000 1100

1200

1300

<AxesSubplot:xlabel='no\_previous', ylabel='Density'> Out[16]: 0.30 0.08 0.20 0.25 0.07 0.06 0.20 0.15 Density 0.04 Density 0.15 Density 0.10 0.03 0.10 0.02 0.05 0.05 0.01 0.00 0.00 0.00 10 15 20 25 8 10 12 10 12 6 alcohol total speeding 0.12 1.0 0.10 0.10 0.8 0.08 0.08 Density 90.00 0.6 Density 90 90 0.4 0.04 0.04 0.2 0.02 0.02 0.00 0.00 0.0 20 20 10 15 25 30 10 15 25 0.0 0.2 0.4 0.6 0.8 1.0 5

no\_previous

-0.036011

-0.065928

-0.112547

-0.075970

-0.006359 0.623116

1.0

0.8

- 0.6

-0.199702

-0.077675

-0.170612

-0.174856

-0.156895

1.000000

-0.036

-0.078 -0.066

-0.17 -0.11

ins\_losses -0.036011 -0.065928 -0.112547 -0.075970 -0.006359 0.623116 1.000000

from the above correlation table we can say that total and no previous accidents are strongly related. speeding and alcohol are strongly related. alcohol and total are strongly related. not\_distracted and total and no\_previous

0.83

0.59

0.73

0.57

0.78

we can see that all the values are normally distributed because all the graphs are in bell shaped curve.

0.827560

0.588010

0.732816

1.000000

0.747307

-0.174856

alcohol not\_distracted no\_previous ins\_premium ins\_losses

0.956179

0.571976

0.783520

0.747307

1.000000

-0.156895

not\_distracted

0.852613

0.669719

1.000000

0.85

0.67

0.61

0.67

total speeding

0.669719

0.827560 0.588010 0.732816

total 1.000000 0.611548

0.852613

**no\_previous** 0.956179 0.571976

0.611548 1.000000

ins\_premium -0.199702 -0.077675 -0.170612

sn.heatmap(corr, annot=True, cmap="YlGnBu")

0.61

0.85

In [18]:

Out[18]:

Out[20]:

corr=df.corr()

speeding

alcohol

and total are related.

<AxesSubplot:>

total

speeding -

alcohol -

not\_distracted

10.0

7.5

5.0

10

sn.distplot(df["not\_distracted"], ax=axes[1,0])
sn.distplot(df["no\_previous"], ax=axes[1,1])

In [16]: fig, axes = plt.subplots(2, 3,figsize=(18, 10))
 sn.distplot(df["total"], ax=axes[0,0])
 sn.distplot(df["speeding"], ax=axes[0,1])
 sn.distplot(df["alcohol"], ax=axes[0,2])

warnings.warn(msg, FutureWarning)

warnings.warn(msg, FutureWarning)

warnings.warn(msg, FutureWarning)

12

14

no\_previous

16

18

20

22

not_distracted	- 0.83	0.59	0.73	1	0.75	-0.17	-0.076
no_previous	- 0.96	0.57	0.78	0.75	1	-0.16	-0.0064
ins_premium	0.2	-0.078	-0.17	-0.17	-0.16	1	0.62
ins_losses	0.036	-0.066	-0.11	-0.076	-0.0064	0.62	1
	total -	speeding -	alcohol -	stracted -	previous -	premium -	s_losses -

