



```
import seaborn as sns
```

```
df = sns.load_dataset('car_crashes')  
df
```



	total	speeding	alcohol	not_distracted	no_previous	ins_premium	ins_losses	abbrev	
0	18.8	7.332	5.640	18.048	15.040	784.55	145.08	AL	
1	18.1	7.421	4.525	16.290	17.014	1053.48	133.93	AK	
2	18.6	6.510	5.208	15.624	17.856	899.47	110.35	AZ	
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	
5	13.6	5.032	3.808	10.744	12.920	835.50	139.91	CO	
6	10.8	4.968	3.888	9.396	8.856	1068.73	167.02	CT	
7	16.2	6.156	4.860	14.094	16.038	1137.87	151.48	DE	
8	5.9	2.006	1.593	5.900	5.900	1273.89	136.05	DC	
9	17.9	3.759	5.191	16.468	16.826	1160.13	144.18	FL	
10	15.6	2.964	3.900	14.820	14.508	913.15	142.80	GA	
11	17.5	9.450	7.175	14.350	15.225	861.18	120.92	HI	
12	15.3	5.508	4.437	13.005	14.994	641.96	82.75	ID	
13	12.8	4.608	4.352	12.032	12.288	803.11	139.15	IL	
14	14.5	3.625	4.205	13.775	13.775	710.46	108.92	IN	
15	15.7	2.669	3.925	15.229	13.659	649.06	114.47	IA	
16	17.8	4.806	4.272	13.706	15.130	780.45	133.80	KS	
17	21.4	4.066	4.922	16.692	16.264	872.51	137.13	KY	
18	20.5	7.175	6.765	14.965	20.090	1281.55	194.78	LA	
19	15.1	5.738	4.530	13.137	12.684	661.88	96.57	ME	
20	12.5	4.250	4.000	8.875	12.375	1048.78	192.70	MD	
21	8.2	1.886	2.870	7.134	6.560	1011.14	135.63	MA	
22	14.1	3.384	3.948	13.395	10.857	1110.61	152.26	MI	
23	9.6	2.208	2.784	8.448	8.448	777.18	133.35	MN	
24	17.6	2.640	5.456	1.760	17.600	896.07	155.77	MS	

Inference : Loaded car crashes dataset from seaborn library

26	21.4	8.346	9.416	17.976	18.190	816.21	85.15	MT
----	------	-------	-------	--------	--------	--------	-------	----



df.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
```

Inference : Info about df. There are 51 non-null data in all the columns. All the columns has data in float data type except for abbrev, it has object data type

```
39    11.1    3.774    4.218    10.212    8.769    1148.99    148.58    RI
df.head(5)
```

	total	speeding	alcohol	not_distracted	no_previous	ins_premium	ins_losses	abbrev	
0	18.8	7.332	5.640	18.048	15.040	784.55	145.08	AL	
1	18.1	7.421	4.525	16.290	17.014	1053.48	133.93	AK	
2	18.6	6.510	5.208	15.624	17.856	899.47	110.35	AZ	
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	
5	10.0	4.402	3.750	8.032	9.110	690.00	111.02	CO	

Inference: Top 5 elements of data

```
49    13.8    4.968    4.554    5.382    11.592    670.31    106.62    WI
sns.scatterplot(x="total",y="speeding",data=df)
```

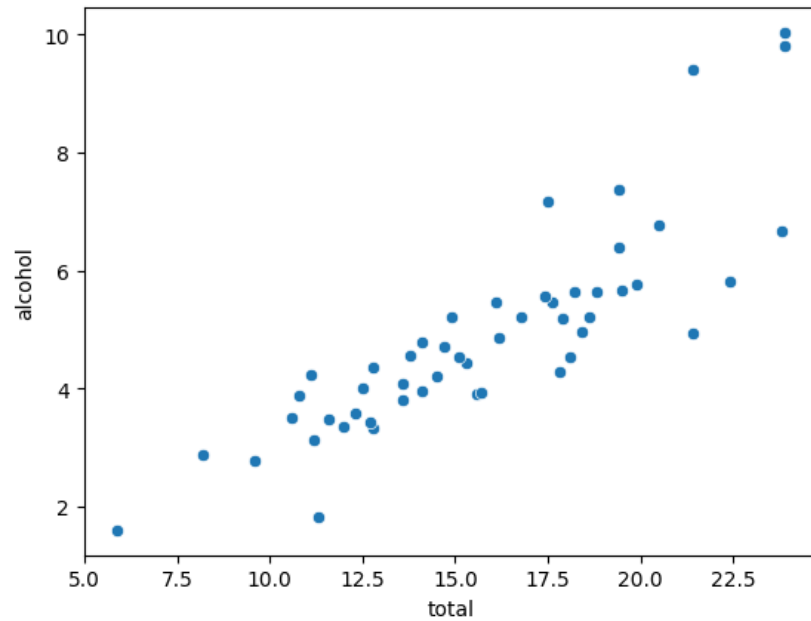
```
<Axes: xlabel='total', ylabel='speeding'>
```



Inference: From the plot we can say that as speeding increases car crashes is also increasing

```
sns.scatterplot(x="total",y="alcohol",data=df)
```

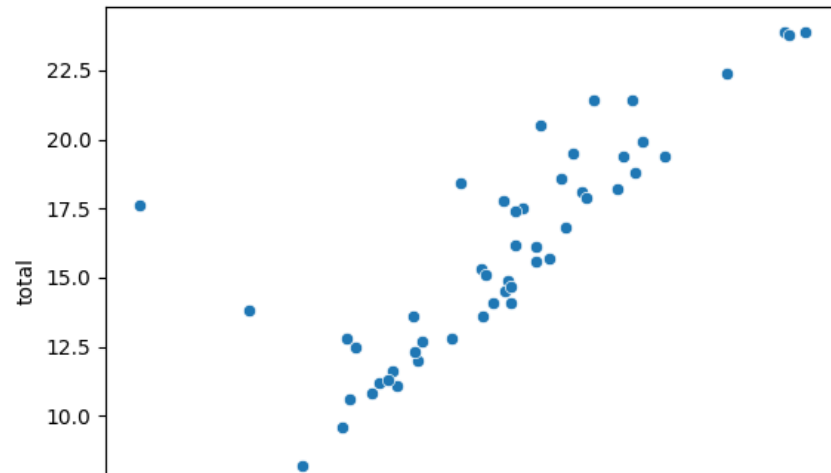
```
<Axes: xlabel='total', ylabel='alcohol'>
```



Inference: From the plot we can say that as alcohol increases car crashes is also increasing

```
sns.scatterplot(x="not_distracted",y="total",data=df)
```

```
<Axes: xlabel='not_distracted', ylabel='total'>
```



Inference: From the plot we can say that as not distracted increases car crashes is also increasing

```
sns.lineplot(x="no_previous",y="total",data=df,ci=None)
```

```
<ipython-input-41-c663563d3ba0>:1: FutureWarning:
```

Inference: From the line plot we can say that as no previous increases car crashes is also increasing

```
sns.linelplot(x= no_previous ,y= total ,data=dt,ci=None)
```

```
sns.distplot(df["ins_premium"])
```

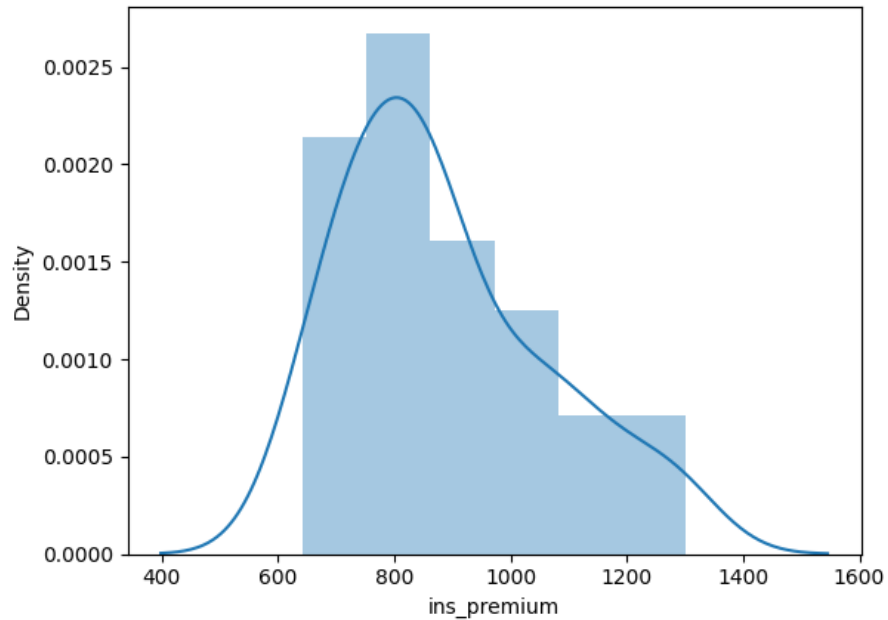
```
<ipython-input-42-8677a75b2d6c>:1: UserWarning:
```

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

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

For a guide to updating your code to use the new functions, please see <https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751>

```
sns.distplot(df["ins_premium"])
<Axes: xlabel='ins_premium', ylabel='Density'>
```



Inference : The ins_premium is ranging from 800 to 900

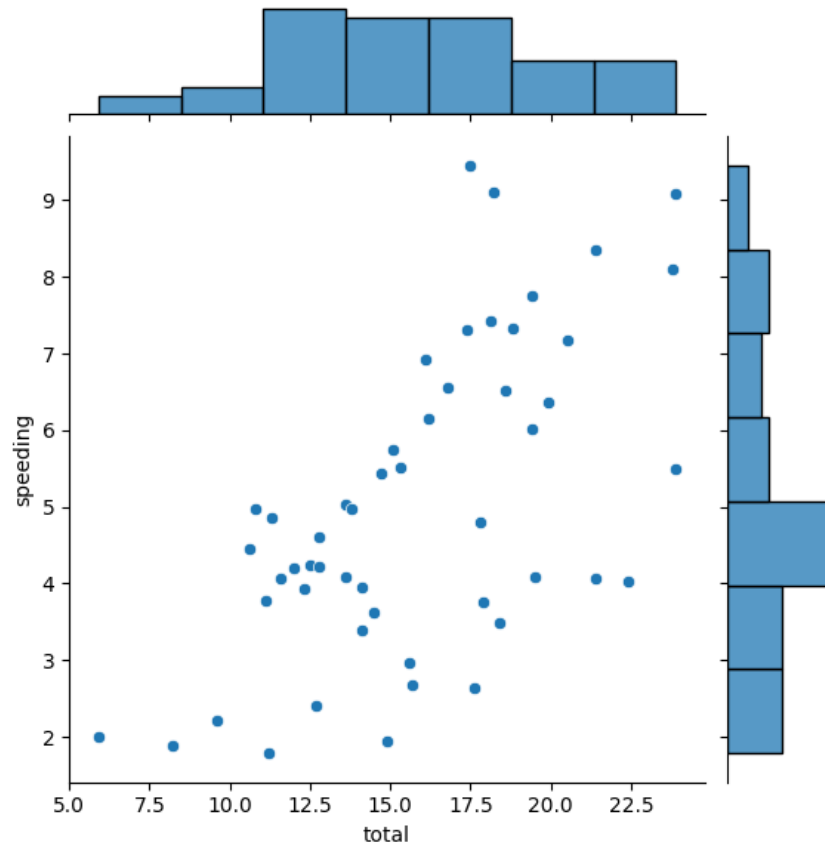
```
df.head(5)
```

	total	speeding	alcohol	not_distracted	no_previous	ins_premium	ins_losses	abbrev
0	18.8	7.332	5.640	18.048	15.040	784.55	145.08	AL
1	18.1	7.421	4.525	16.290	17.014	1053.48	133.93	AK
2	18.6	6.510	5.208	15.624	17.856	899.47	110.35	AZ
3	22.4	4.022	5.824	21.056	21.280	827.34	142.30	AR



```
sns.jointplot(x="total",y="speeding",data=df)
```

<seaborn.axisgrid.JointGrid at 0x79333b6ea9b0>



Inference: Upper histogram is univariate analysis of total and right histogram is univariate analysis of speeding. And relationship between total and speeding is bivariate analysis

```
corr=df.corr()
corr
```

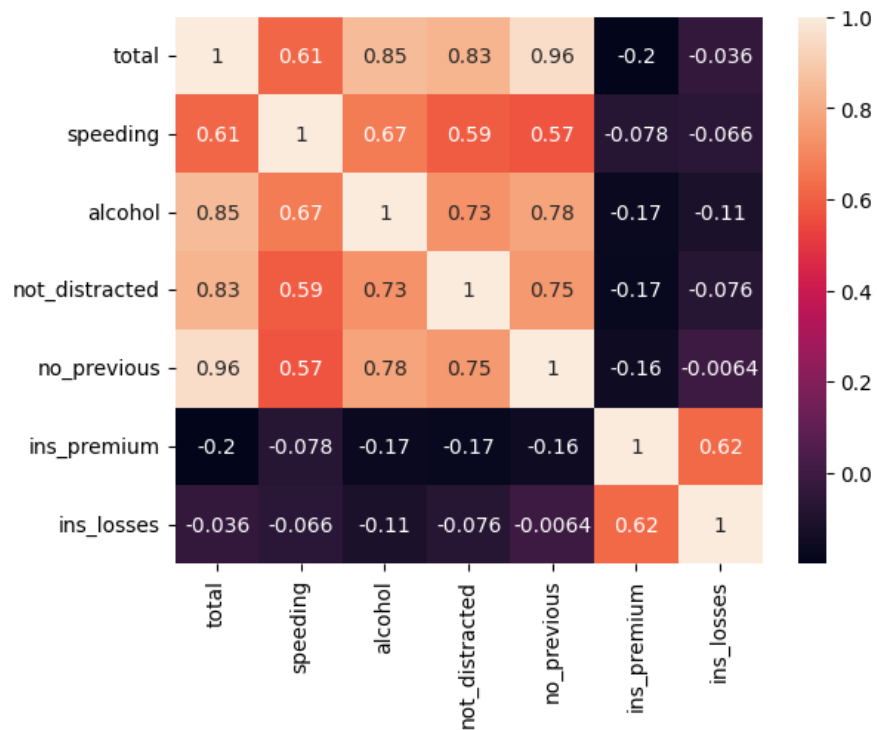
<ipython-input-45-7d5195e2bf4d>:1: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future version, it will default to 'ignore'. To silence this warning, you can explicitly specify 'numeric_only=True'.

```
corr=df.corr()
```

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

```
sns.heatmap(corr,annot=True)
```

<Axes: >



Inference: Less than 0.5 is highly correlated and More than 0.5 is less correlated

[Colab paid products](#) - [Cancel contracts here](#)

✓ 0s completed at 1:17 AM

