

ASSIGNMENT 2

21BCE9197

HALEEMA SADHIA SYED

```
In [1]: import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
```

```
In [6]: sns.get_dataset_names()
```

```
Out[6]: ['anagrams',
'anscombe',
'attention',
'brain_networks',
'car_crashes',
'diamonds',
'dots',
'dowjones',
'exercise',
'flights',
'fmri',
'geyser',
'glue',
'healthexp',
'iris',
'mpg',
'penguins',
'planets',
'seaice',
'taxis',
'tips',
'titanic']
```

```
In [7]: sadhia = sns.load_dataset('car_crashes')  
sadhia
```

Out[7]:

	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
25	16.1	6.923	5.474	14.812	13.524	790.32	144.45	MO
26	21.4	8.346	9.416	17.976	18.190	816.21	85.15	MT
27	14.9	1.937	5.215	13.857	13.410	732.28	114.82	NE
28	14.7	5.439	4.704	13.965	14.553	1029.87	138.71	NV
29	11.6	4.060	3.480	10.092	9.628	746.54	120.21	NH
30	11.2	1.792	3.136	9.632	8.736	1301.52	159.85	NJ
31	18.4	3.496	4.968	12.328	18.032	869.85	120.75	NM
32	12.3	3.936	3.567	10.824	9.840	1234.31	150.01	NY
33	16.8	6.552	5.208	15.792	13.608	708.24	127.82	NC
34	23.9	5.497	10.038	23.661	20.554	688.75	109.72	ND
35	14.1	3.948	4.794	13.959	11.562	697.73	133.52	OH

	total	speeding	alcohol	not_distracted	no_previous	ins_premium	ins_losses	abbrev
36	19.9	6.368	5.771	18.308	18.706	881.51	178.86	OK
37	12.8	4.224	3.328	8.576	11.520	804.71	104.61	OR
38	18.2	9.100	5.642	17.472	16.016	905.99	153.86	PA
39	11.1	3.774	4.218	10.212	8.769	1148.99	148.58	RI
40	23.9	9.082	9.799	22.944	19.359	858.97	116.29	SC
41	19.4	6.014	6.402	19.012	16.684	669.31	96.87	SD
42	19.5	4.095	5.655	15.990	15.795	767.91	155.57	TN
43	19.4	7.760	7.372	17.654	16.878	1004.75	156.83	TX
44	11.3	4.859	1.808	9.944	10.848	809.38	109.48	UT
45	13.6	4.080	4.080	13.056	12.920	716.20	109.61	VT
46	12.7	2.413	3.429	11.049	11.176	768.95	153.72	VA
47	10.6	4.452	3.498	8.692	9.116	890.03	111.62	WA
48	23.8	8.092	6.664	23.086	20.706	992.61	152.56	WV
49	13.8	4.968	4.554	5.382	11.592	670.31	106.62	WI
50	17.4	7.308	5.568	14.094	15.660	791.14	122.04	WY

In [12]: `sadhia.describe()`

Out[12]:

	total	speeding	alcohol	not_distracted	no_previous	ins_premium	ins_losses
count	51.000000	51.000000	51.000000	51.000000	51.000000	51.000000	51.000000
mean	15.790196	4.998196	4.886784	13.573176	14.004882	886.957647	134.493137
std	4.122002	2.017747	1.729133	4.508977	3.764672	178.296285	24.835922
min	5.900000	1.792000	1.593000	1.760000	5.900000	641.960000	82.750000
25%	12.750000	3.766500	3.894000	10.478000	11.348000	768.430000	114.645000
50%	15.600000	4.608000	4.554000	13.857000	13.775000	858.970000	136.050000
75%	18.500000	6.439000	5.604000	16.140000	16.755000	1007.945000	151.870000
max	23.900000	9.450000	10.038000	23.661000	21.280000	1301.520000	194.780000

In [13]: `sadhia.tail()`

Out[13]:

	total	speeding	alcohol	not_distracted	no_previous	ins_premium	ins_losses	abbrev
46	12.7	2.413	3.429	11.049	11.176	768.95	153.72	VA
47	10.6	4.452	3.498	8.692	9.116	890.03	111.62	WA
48	23.8	8.092	6.664	23.086	20.706	992.61	152.56	WV
49	13.8	4.968	4.554	5.382	11.592	670.31	106.62	WI
50	17.4	7.308	5.568	14.094	15.660	791.14	122.04	WY

In [14]: `sadhia.head()`

Out[14]:

	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

In [9]: `sadhia.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
```

In [10]: `sadhia.info`

```
Out[10]: <bound method DataFrame.info of
no_previous  ins_premium  \
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
9      17.9      3.759      5.191      16.468      16.826      1160.13
10     15.6      2.964      3.900      14.820      14.508      913.15
11     17.5      9.450      7.175      14.350      15.225      861.18
12     15.3      5.508      4.437      13.005      14.994      641.96
13     12.8      4.608      4.352      12.032      12.288      803.11
14     14.5      3.625      4.205      13.775      13.775      710.46
15     15.7      2.669      3.925      15.229      13.659      649.06
16     17.8      4.806      4.272      13.706      15.130      780.45
17     21.4      4.066      4.922      16.692      16.264      872.51
18     20.5      7.175      6.765      14.965      20.090      1281.55
19     15.1      5.738      4.530      13.137      12.684      661.88
20     12.5      4.250      4.000      8.875      12.375      1048.78
21      8.2      1.886      2.870      7.134      6.560      1011.14
22     14.1      3.384      3.948      13.395      10.857      1110.61
23      9.6      2.208      2.784      8.448      8.448      777.18
24     17.6      2.640      5.456      1.760      17.600      896.07
25     16.1      6.923      5.474      14.812      13.524      790.32
26     21.4      8.346      9.416      17.976      18.190      816.21
27     14.9      1.937      5.215      13.857      13.410      732.28
28     14.7      5.439      4.704      13.965      14.553      1029.87
29     11.6      4.060      3.480      10.092      9.628      746.54
30     11.2      1.792      3.136      9.632      8.736      1301.52
31     18.4      3.496      4.968      12.328      18.032      869.85
32     12.3      3.936      3.567      10.824      9.840      1234.31
33     16.8      6.552      5.208      15.792      13.608      708.24
34     23.9      5.497      10.038      23.661      20.554      688.75
35     14.1      3.948      4.794      13.959      11.562      697.73
36     19.9      6.368      5.771      18.308      18.706      881.51
37     12.8      4.224      3.328      8.576      11.520      804.71
38     18.2      9.100      5.642      17.472      16.016      905.99
39     11.1      3.774      4.218      10.212      8.769      1148.99
40     23.9      9.082      9.799      22.944      19.359      858.97
41     19.4      6.014      6.402      19.012      16.684      669.31
42     19.5      4.095      5.655      15.990      15.795      767.91
43     19.4      7.760      7.372      17.654      16.878      1004.75
44     11.3      4.859      1.808      9.944      10.848      809.38
45     13.6      4.080      4.080      13.056      12.920      716.20
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
50     17.4      7.308      5.568      14.094      15.660      791.14
```

```
ins_losses abbrev
0      145.08      AL
1      133.93      AK
```

2	110.35	AZ
3	142.39	AR
4	165.63	CA
5	139.91	CO
6	167.02	CT
7	151.48	DE
8	136.05	DC
9	144.18	FL
10	142.80	GA
11	120.92	HI
12	82.75	ID
13	139.15	IL
14	108.92	IN
15	114.47	IA
16	133.80	KS
17	137.13	KY
18	194.78	LA
19	96.57	ME
20	192.70	MD
21	135.63	MA
22	152.26	MI
23	133.35	MN
24	155.77	MS
25	144.45	MO
26	85.15	MT
27	114.82	NE
28	138.71	NV
29	120.21	NH
30	159.85	NJ
31	120.75	NM
32	150.01	NY
33	127.82	NC
34	109.72	ND
35	133.52	OH
36	178.86	OK
37	104.61	OR
38	153.86	PA
39	148.58	RI
40	116.29	SC
41	96.87	SD
42	155.57	TN
43	156.83	TX
44	109.48	UT
45	109.61	VT
46	153.72	VA
47	111.62	WA
48	152.56	WV
49	106.62	WI
50	122.04	WY >


```
In [17]: sadhia.isnull().sum()
```

```
Out[17]: total          0
         speeding       0
         alcohol        0
         not_distracted 0
         no_previous     0
         ins_premium     0
         ins_losses      0
         abbrev         0
         dtype: int64
```

```
In [19]: sadhia.isna().sum()
```

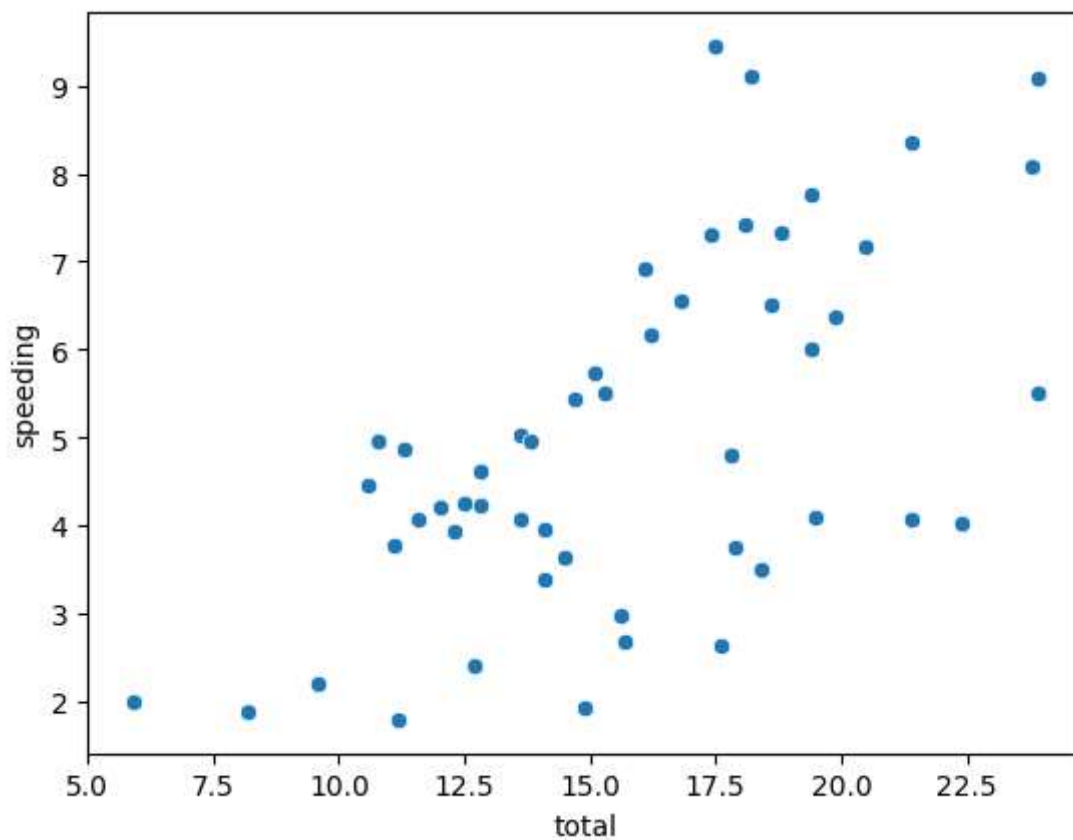
```
Out[19]: total          0
         speeding       0
         alcohol        0
         not_distracted 0
         no_previous     0
         ins_premium     0
         ins_losses      0
         abbrev         0
         dtype: int64
```

```
In [20]: sadhia.isnull().any()
```

```
Out[20]: total          False
         speeding       False
         alcohol        False
         not_distracted False
         no_previous     False
         ins_premium     False
         ins_losses      False
         abbrev         False
         dtype: bool
```

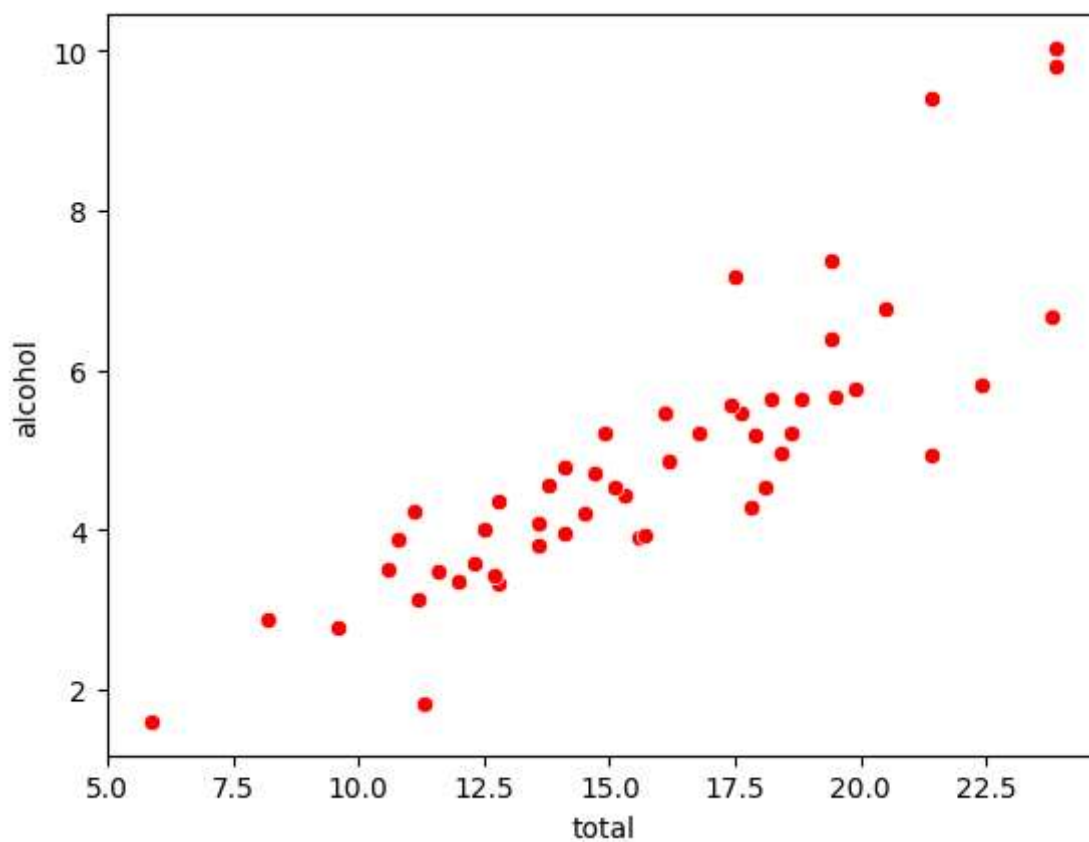
```
In [40]: #SCATTERPLOT
sns.scatterplot(x="total", y="speeding", data=sadhia)
#Inference: Above plot shows that as speeding increases total crashes is also
```

```
Out[40]: <Axes: xlabel='total', ylabel='speeding'>
```



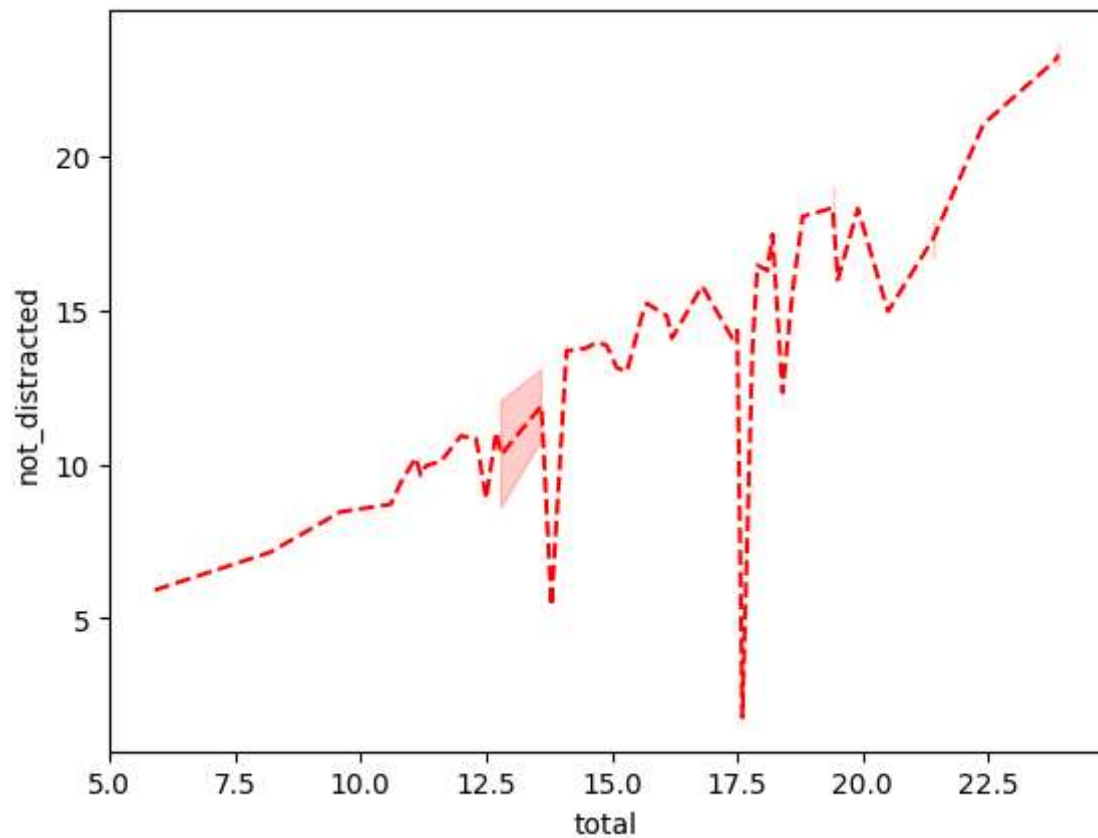
```
In [41]: sns.scatterplot(x="total", y="alcohol", data=sadhia, color="r")  
#Inference: Above plot shows that as alcohol increases total crashes is also i
```

```
Out[41]: <Axes: xlabel='total', ylabel='alcohol'>
```



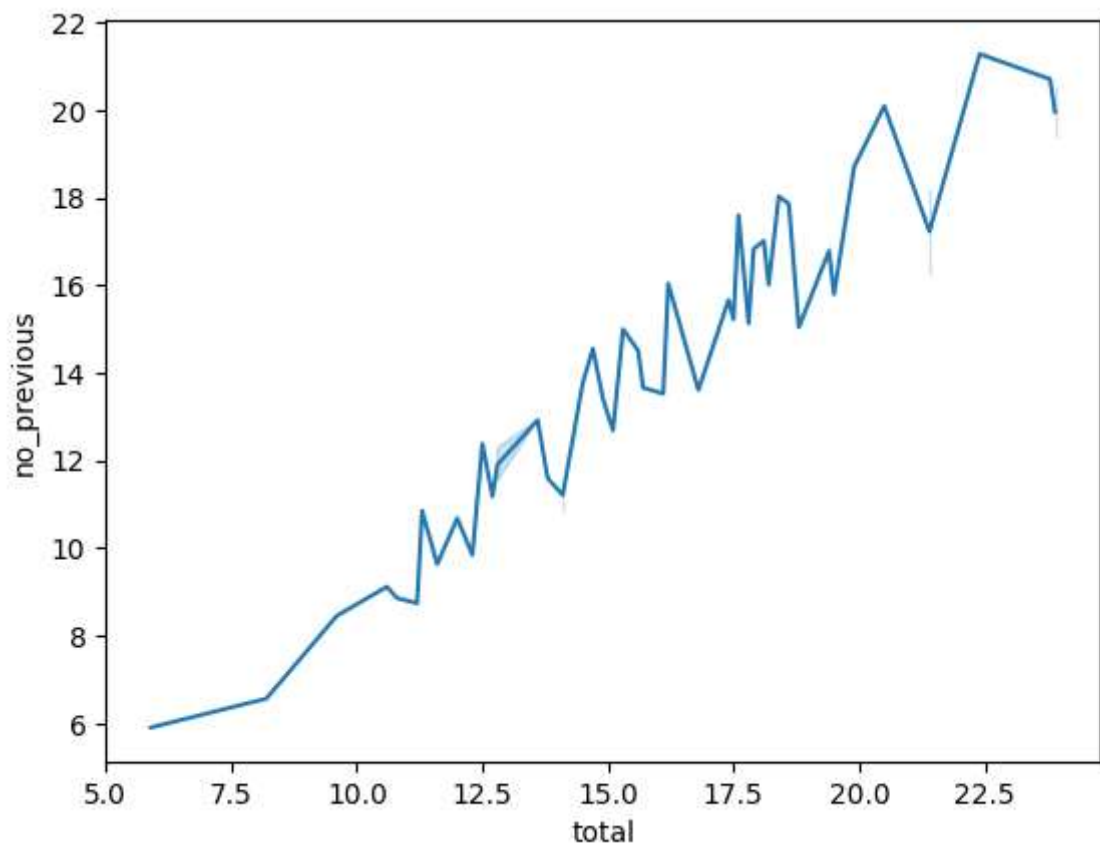
```
In [71]: sns.lineplot(x="total", y="not_distracted", data=sadhia, linestyle="--", color=
#Inference: Above plot shows that as not_distracted increases total crashes is
```

```
Out[71]: <Axes: xlabel='total', ylabel='not_distracted'>
```



```
In [69]: #LINE PLOT
sns.lineplot(x="total", y="no_previous",data=sadhia)
#Inference: Above graph it is no of drivers increases with percetage in fatal
```

Out[69]: <Axes: xlabel='total', ylabel='no_previous'>



```
In [72]: #DISTRIBUTION PLOT
sns.distplot(sadhia["ins_premium"])
#Inference: Above graph it is true that the speeding is high at 300 to 900
```

<ipython-input-72-e1f31e266ffe>:2: UserWarning:

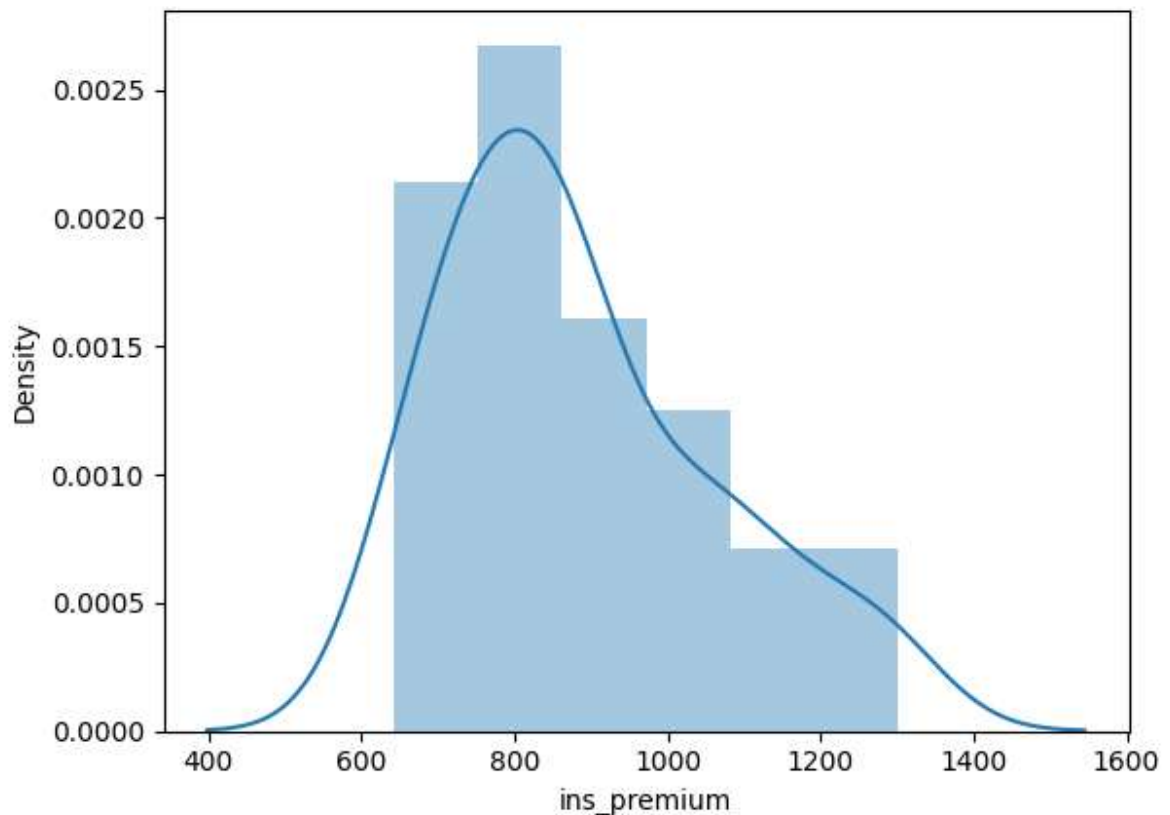
`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

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> (<https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751>)

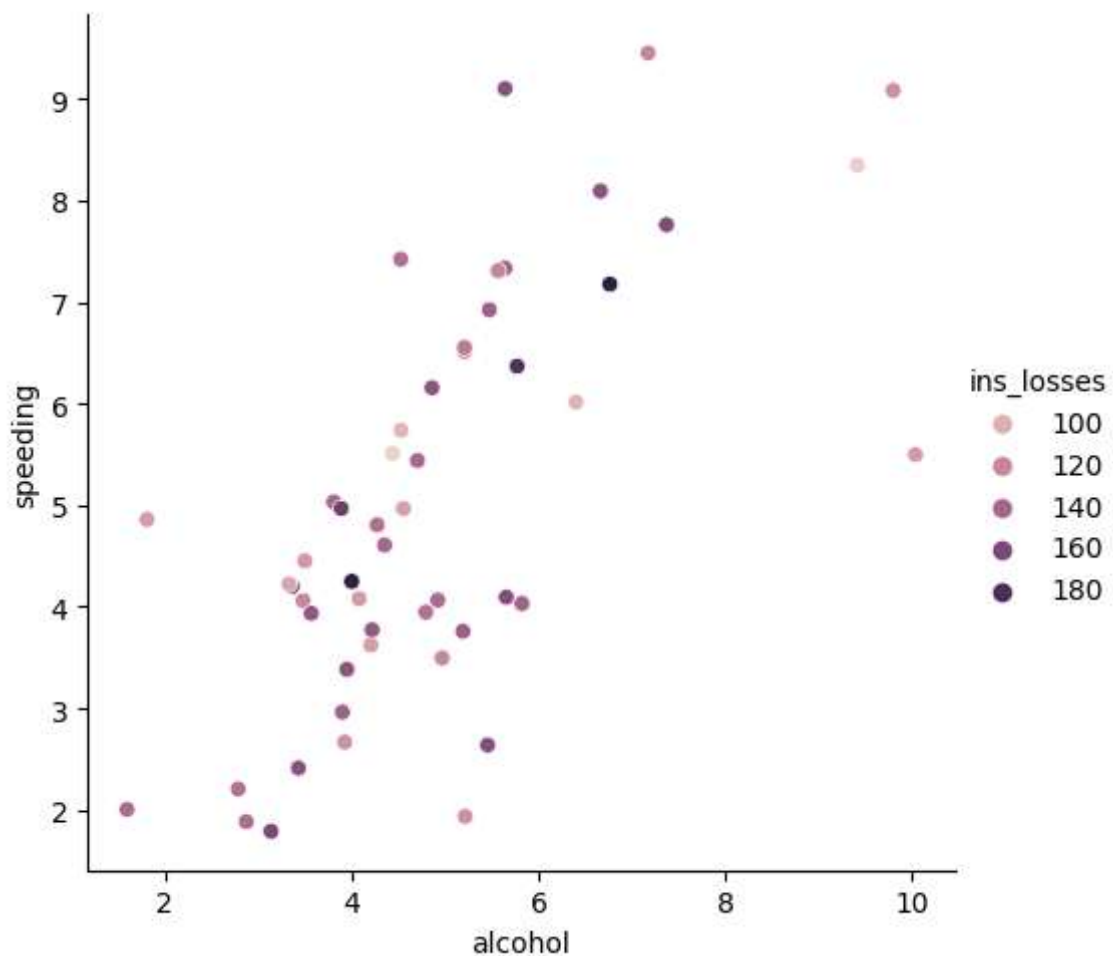
```
sns.distplot(sadhia["ins_premium"])
```

Out[72]: <Axes: xlabel='ins_premium', ylabel='Density'>



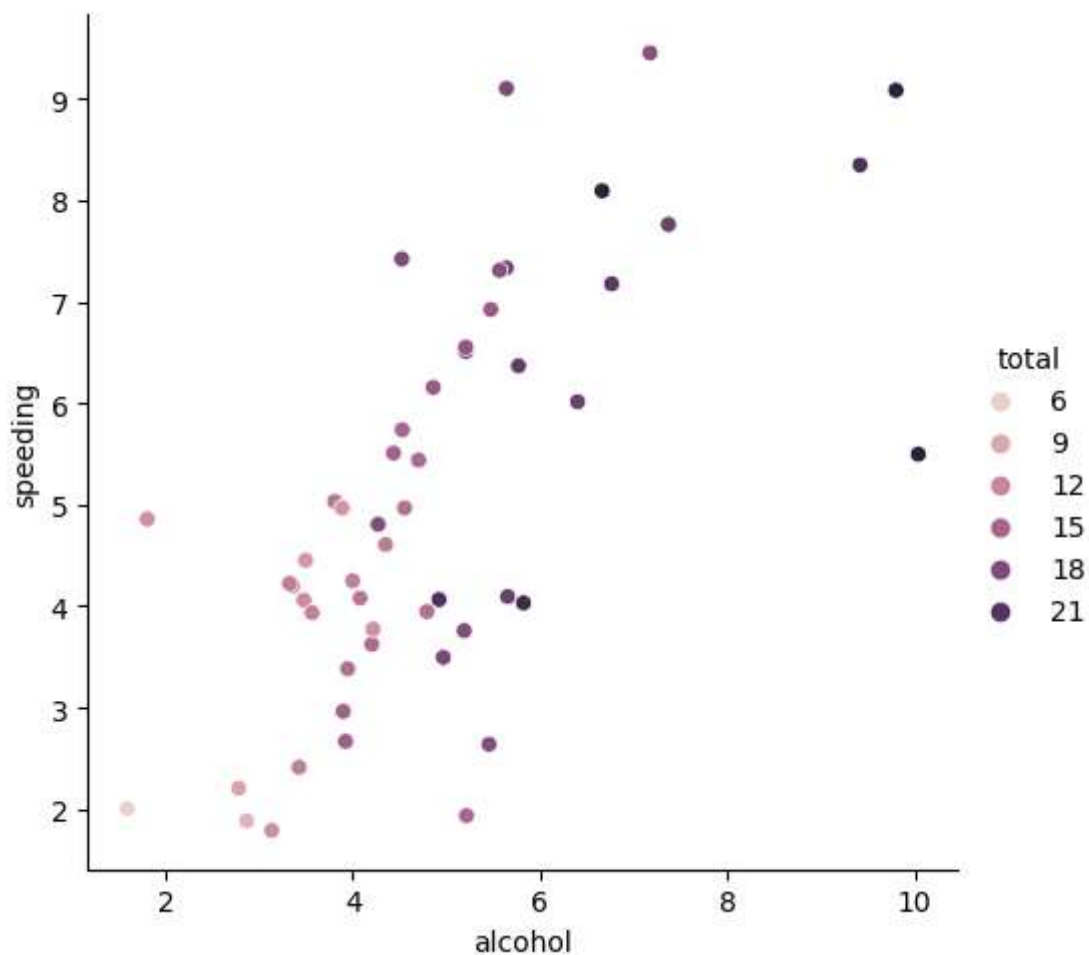
```
In [46]: #Relplot
sns.relplot(x="alcohol", y="speeding", data=sadhia , hue="ins_losses")
#Inference: Alcohol consumption is directly proportional to speeding
```

Out[46]: <seaborn.axisgrid.FacetGrid at 0x7fea6dd9c5b0>

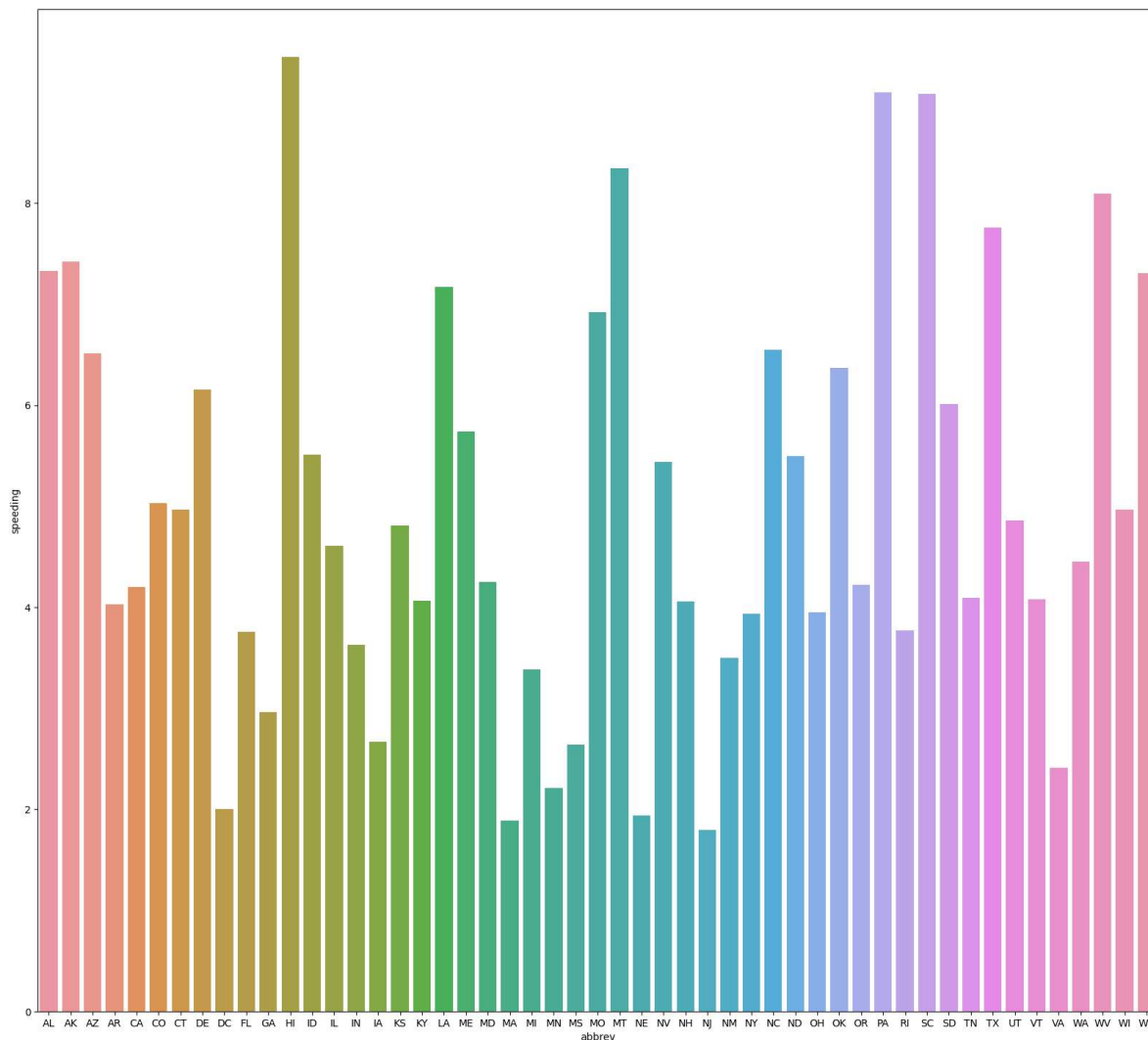


```
In [47]: sns.relplot(x="alcohol", y="speeding", data=sadhia , hue="total")  
#Inference: we can say that as speeding increases alcohol is also increasing
```

```
Out[47]: <seaborn.axisgrid.FacetGrid at 0x7fea6d332c20>
```

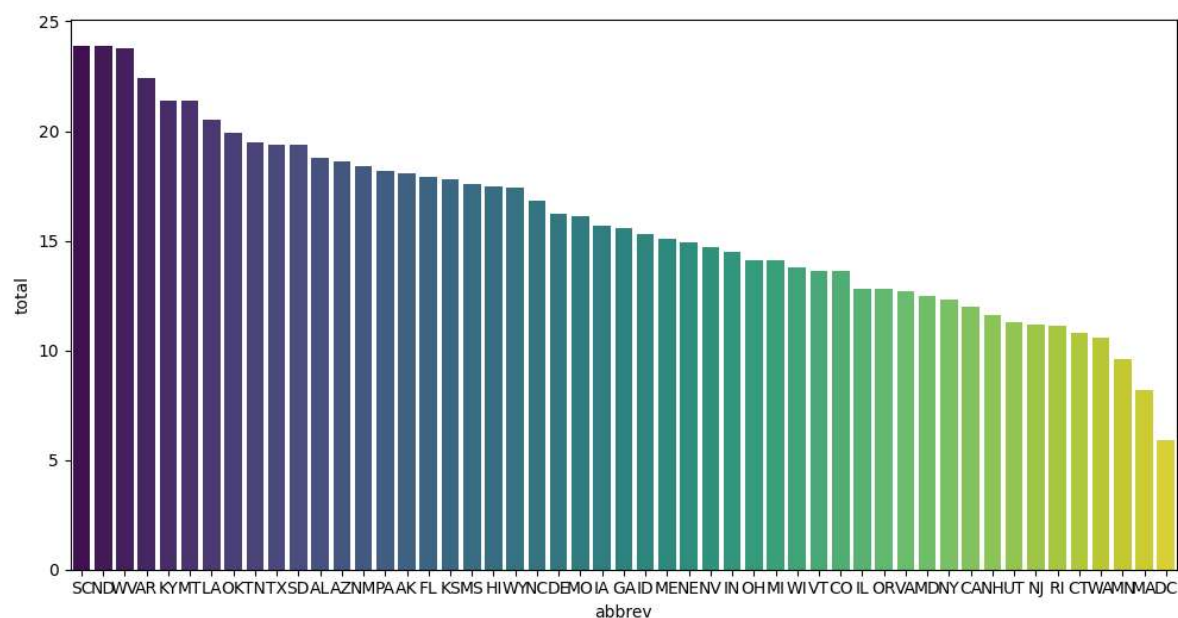



```
In [50]: #BAR PLOT
plt.figure(figsize=(20, 18))
sns.barplot(x='abbrev', y='speeding', data=sadhia)
plt.show()
#Inference: ND has the total number of heighest collisions
```



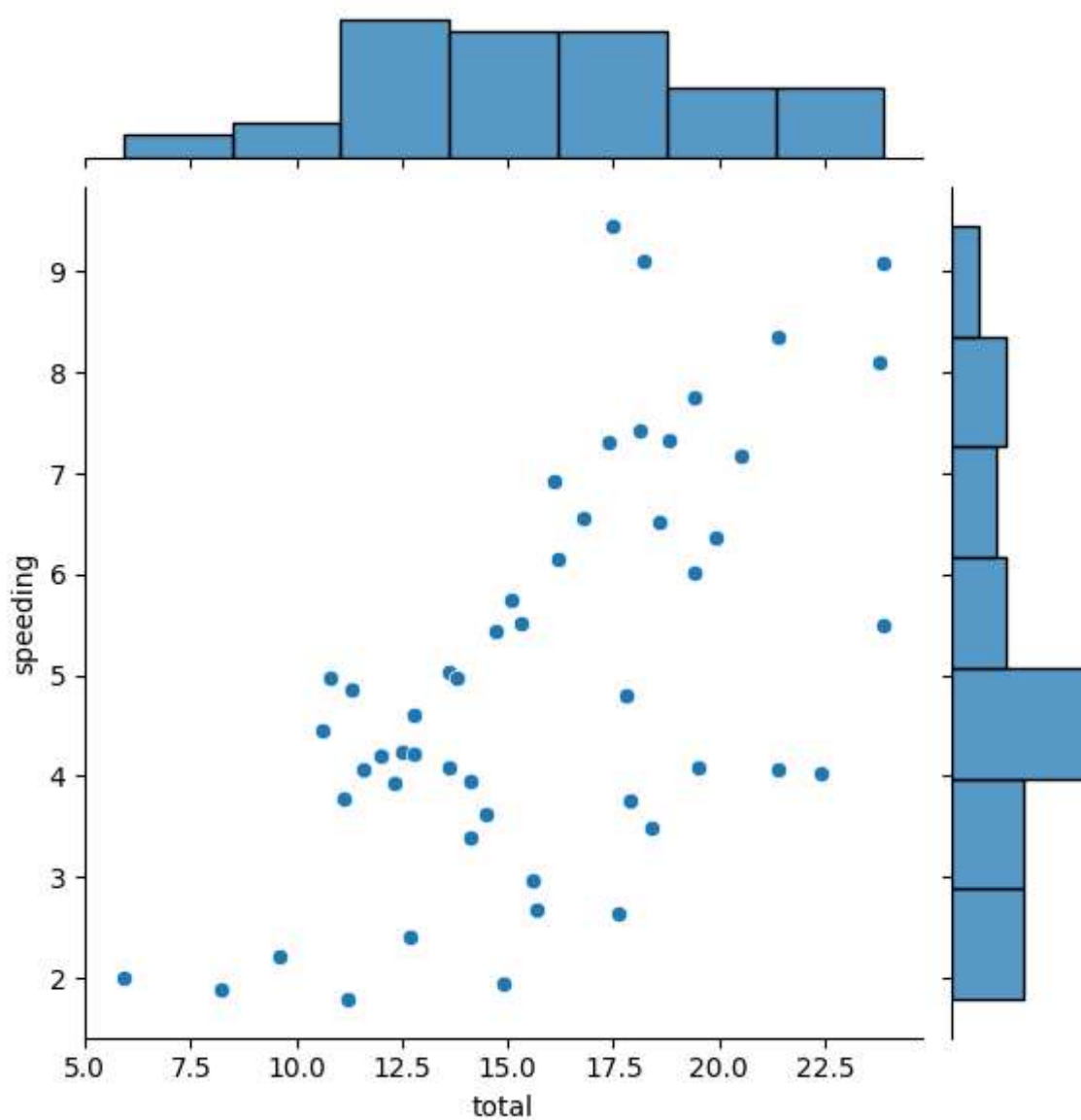
```
In [56]: df= df.sort_values(by='total', ascending=False)
plt.figure(figsize=(12, 6))
sns.barplot(x='abbrev', y='total', data=sadhia, palette='viridis')
#Inference: we can say that as abbrev 'SC' have higher total crashes among the
```

```
Out[56]: <Axes: xlabel='abbrev', ylabel='total'>
```



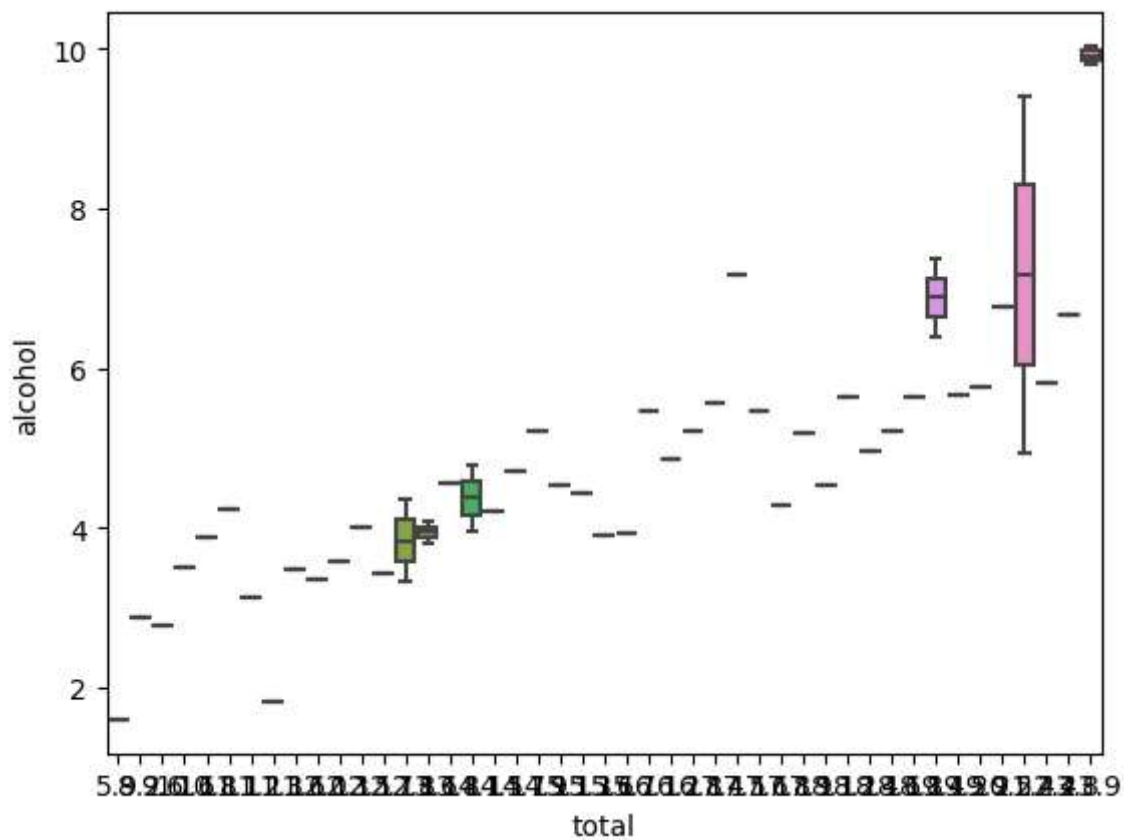
```
In [57]: #JOIN PLOT
sns.jointplot(x="total",y="speeding",data=sadhia)
#Inference: we can say that as speeding and total crashes are increased at (1
```

```
Out[57]: <seaborn.axisgrid.JointGrid at 0x7fea6cbdb400>
```



```
In [59]: #BOX PLOT
sns.boxplot(x="total",y="alcohol",data=sadhia)
#Inference: we can say that as alcohol increases total crashes is also increases
```

```
Out[59]: <Axes: xlabel='total', ylabel='alcohol'>
```

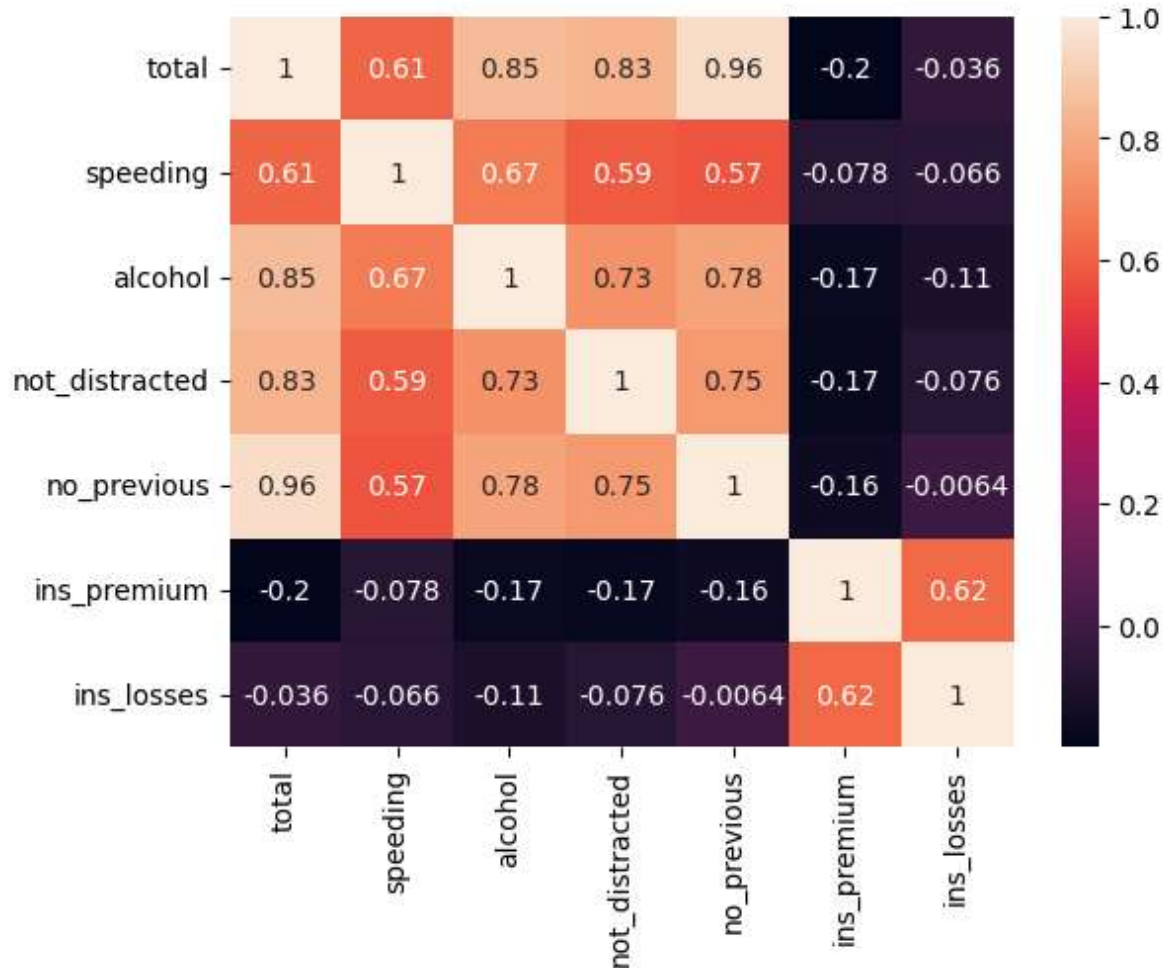


```
In [63]: corr=sadhia.corr()
corr
sns.heatmap(corr,annot=True)
#Inference: we can say that >0.5 is highly correlated <0.5 is less correlated
```

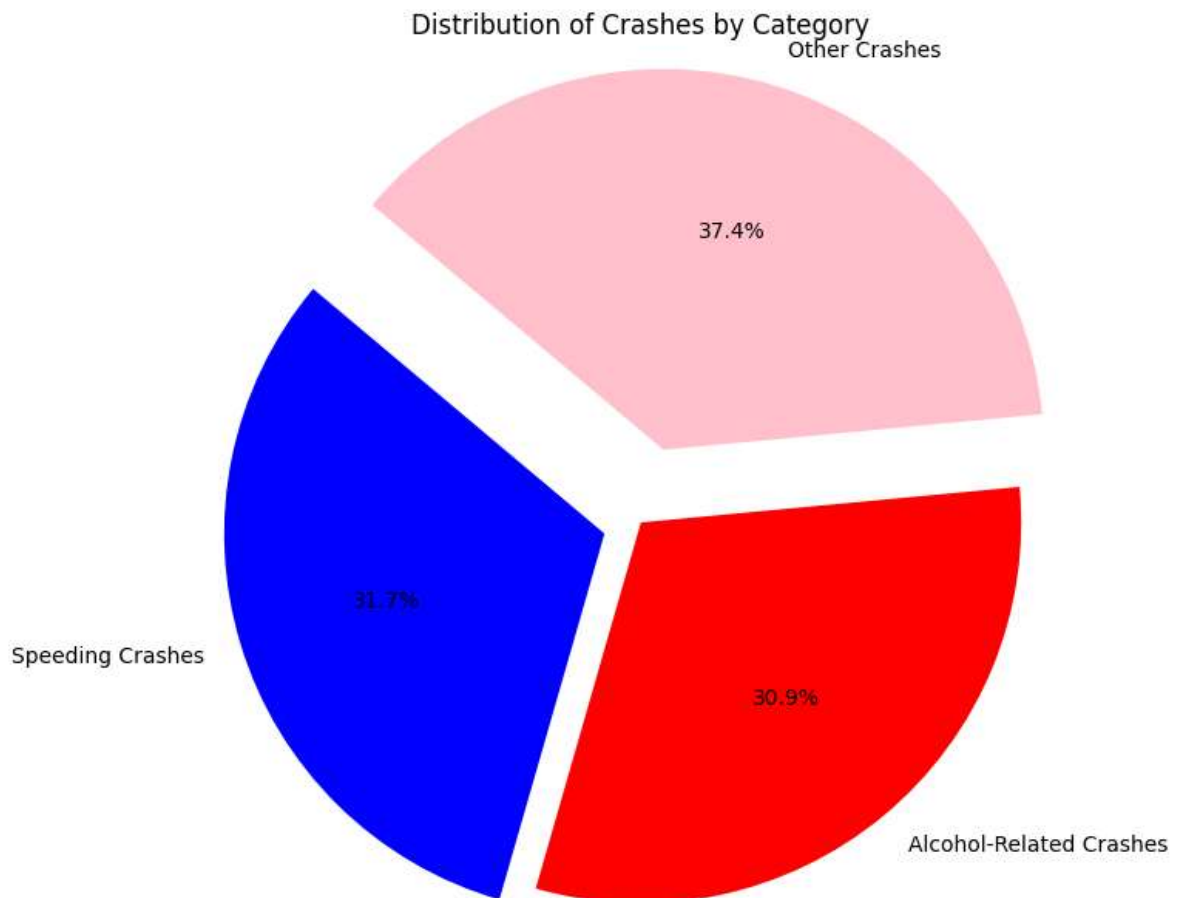
<ipython-input-63-3895bddc49d9>:1: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future version, it will default to False. Select only valid columns or specify the value of numeric_only to silence this warning.

```
corr=sadhia.corr()
```

Out[63]: <Axes: >



```
In [68]: tsc = sadhia['speeding'].sum()
tac = sadhia['alcohol'].sum()
toc = sadhia['total'].sum() - (tsc + tac)
labels = ['Speeding Crashes', 'Alcohol-Related Crashes', 'Other Crashes']
sizes = [tsc, tac, toc]
colors = ['blue', 'red', 'pink']
Explode=[0.1,0,0.2]
plt.figure(figsize=(8, 8))
plt.pie(sizes, labels=labels, colors=colors, autopct='%1.1f%%', startangle=140)
plt.title('Distribution of Crashes by Category')
plt.show()
#Inference: we can say that as other reason crashes greater than the Alcohol r
#Speeding crashes slightly greater than the Alcohol related Crashes. Here i ex
```



```
In [73]: #BAR PLOT
sns.barplot(sadhia['alcohol'])
plt.show()
#Inference: we can say that alcohol is just below 5
```

