

## ▼ 8TH\_SEPT\_ASSIGNMENT

NAME:- B. PAVAN KUMAR

REG NO:- 21BCE8241

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>

## ▼ Steps:

- 1.import the necessary libraries
- 2.import the dataset
- 3.Handling null values
- 4.Seperate Dependent and independent variables
- 5.Encoding
- 6.splitting into training and testing set
- 7.Feature scaling

### ▼ 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

```
print(sns.get_dataset_names())
```

```
['anagrams', 'anscombe', 'attention', 'brain_networks', 'car_crashes', 'diamonds', 'dots', 'dowjones', 'exercise', 'flights', 'fmri', 'g
```



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

```
dataset
```

	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

```
sns.__version__
```

```
'0.12.2'
```

```
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

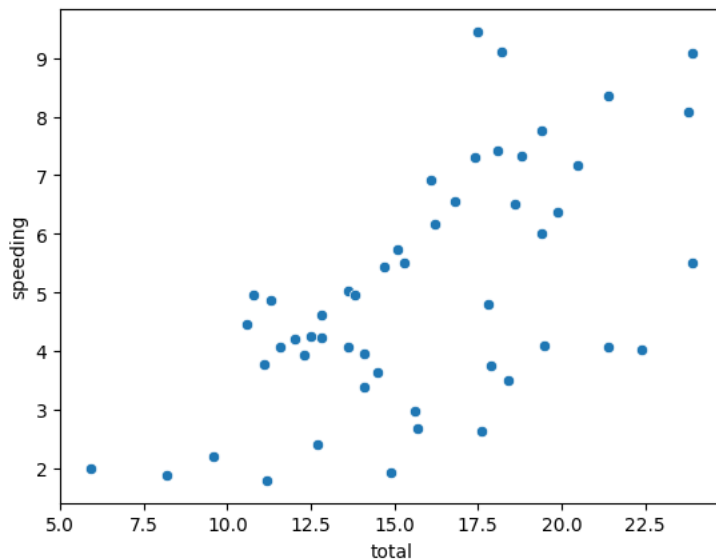
```

```
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

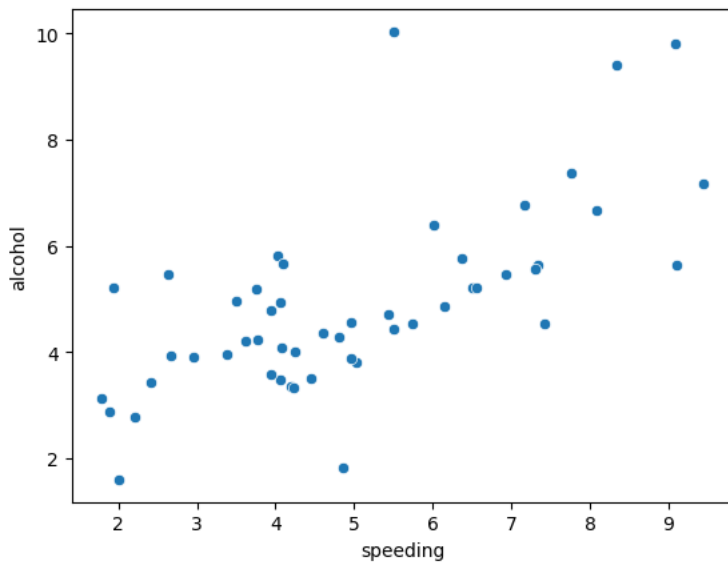
```
sns.scatterplot(x="total",y="speeding",data=df)
```

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



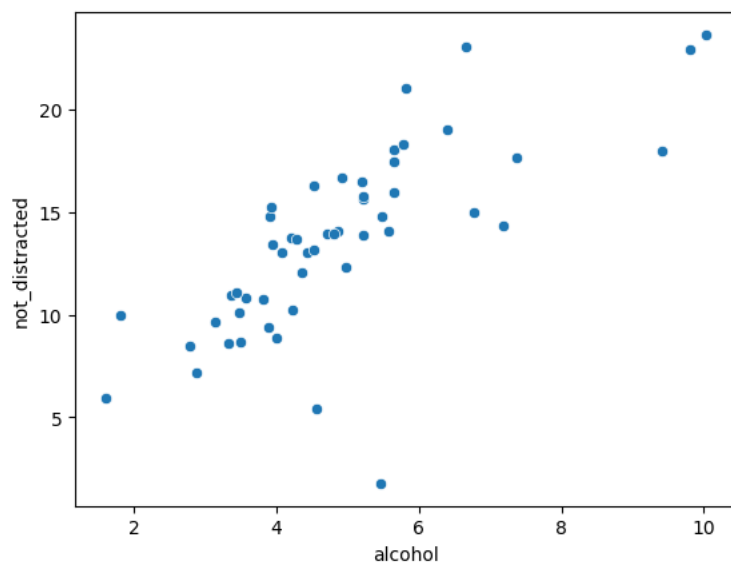
```
sns.scatterplot(x="speeding",y="alcohol",data=df)
```

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



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

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



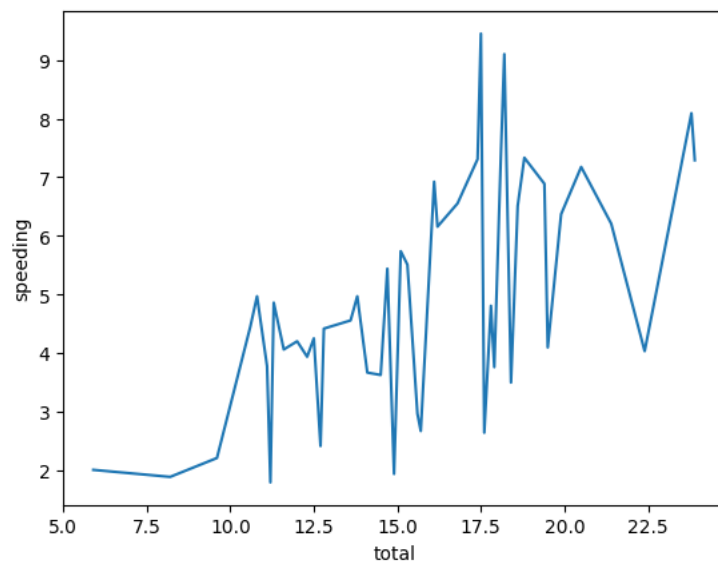
```
sns.lineplot(x="total",y="speeding",data=df,ci=None)
```

```
<ipython-input-24-3a8ebbb7f915>:1: FutureWarning:
```

The `ci` parameter is deprecated. Use `errorbar=None` for the same effect.

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

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



```
sns.lineplot(x="alcohol",y="not_distracted",data=df,ci=None)
```

```
<ipython-input-25-46164a1eaf06>:1: FutureWarning:
```

The `ci` parameter is deprecated. Use `errorbar=None` for the same effect.

```
sns.lineplot(x="alcohol",y="not_distracted",data=df,ci=None)
<Axes: xlabel='alcohol', ylabel='not_distracted'>
```

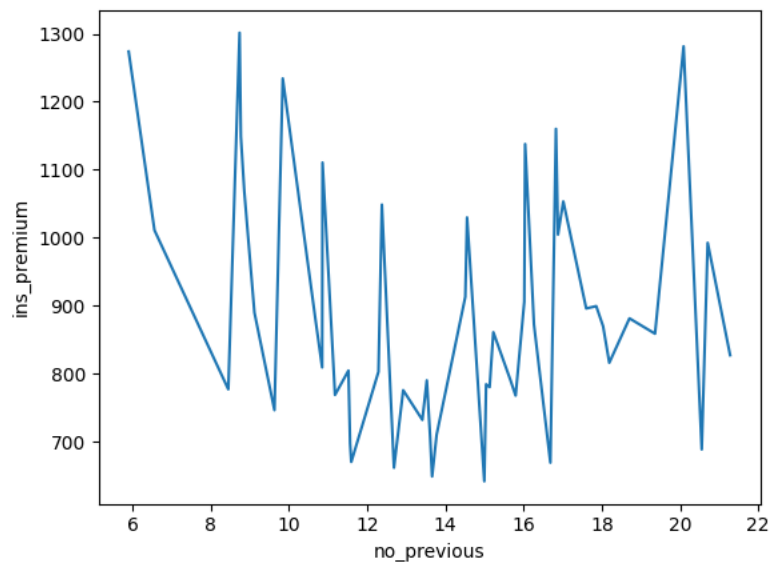


```
sns.lineplot(x="no_previous",y="ins_premium",data=df,ci=None)
```

```
<ipython-input-26-f3f326d16e1a>:1: FutureWarning:
```

The `ci` parameter is deprecated. Use `errorbar=None` for the same effect.

```
sns.lineplot(x="no_previous",y="ins_premium",data=df,ci=None)
<Axes: xlabel='no_previous', ylabel='ins_premium'>
```



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

```
<ipython-input-27-0d5ead9bfd1a>: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

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

```
<ipython-input-28-281d56044cde>:1: UserWarning:
```

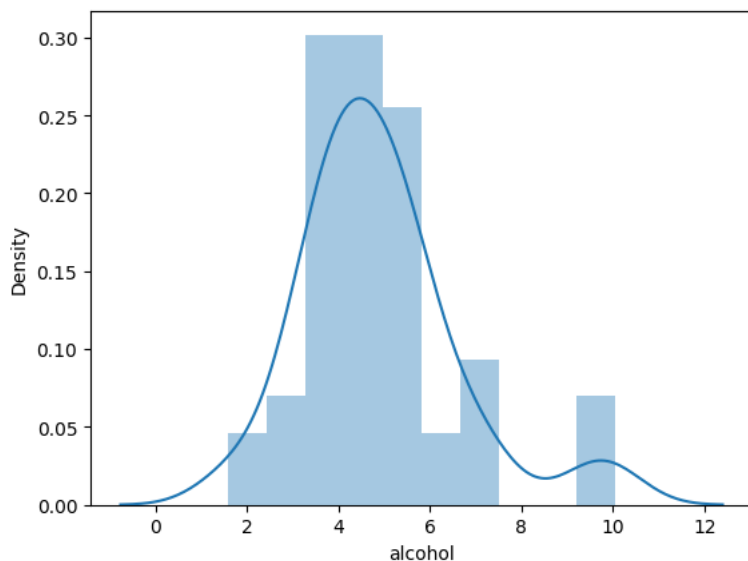
```
`distplot` is a deprecated function and will be removed in seaborn v0.14.0.
```

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["alcohol"])  
<Axes: xlabel='alcohol', ylabel='Density'>
```

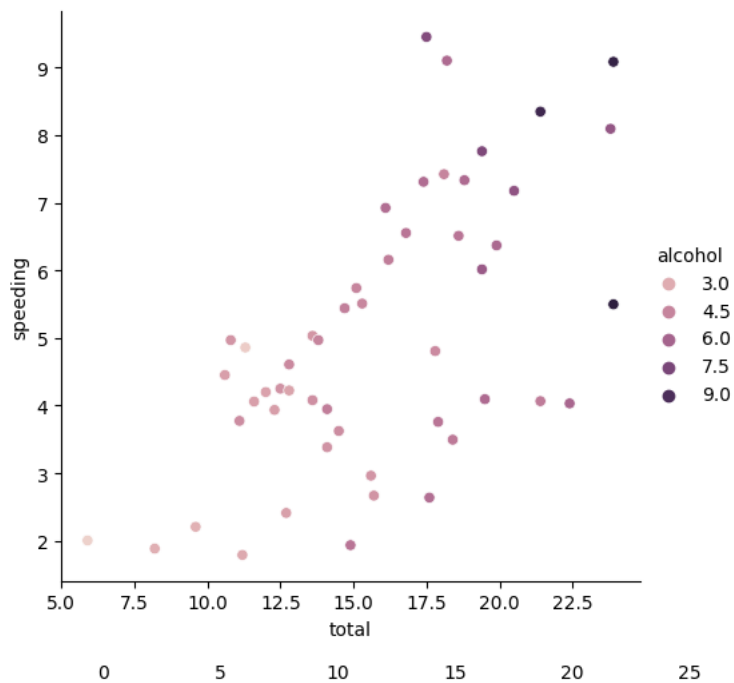


```
sns.distplot(df["no_previous"])
```

```
<ipython-input-29-f4c3808e7f5a>:1: UserWarning:
```

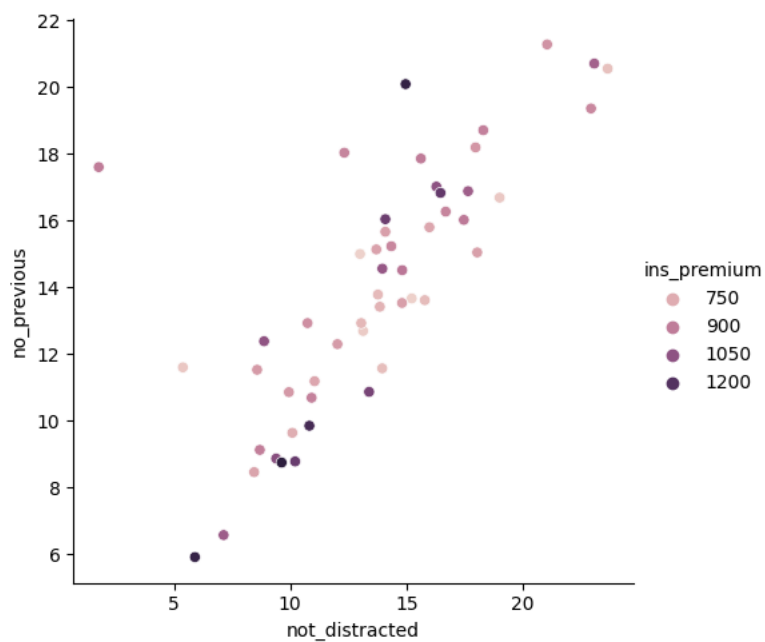
```
sns.relplot(x="total",y="speeding",data=df,hue="alcohol")
```

```
<seaborn.axisgrid.FacetGrid at 0x7af54fd4da50>
```

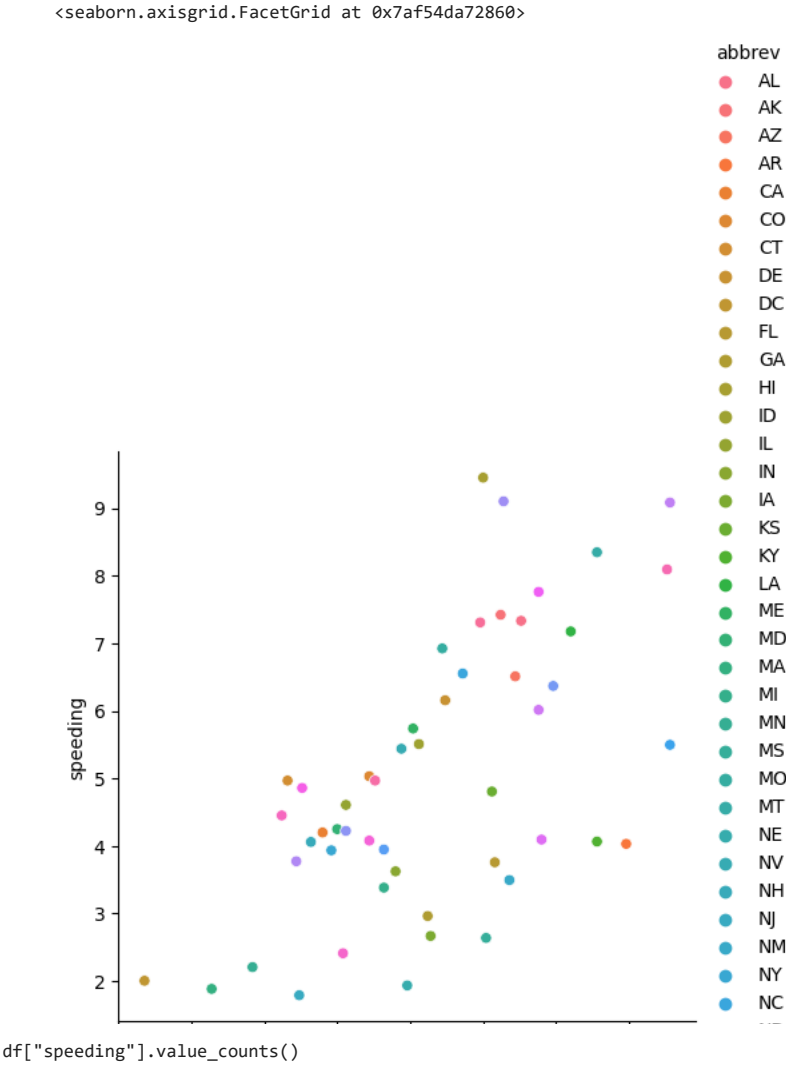


```
sns.relplot(x="not_distracted",y="no_previous",data=df,hue="ins_premium")
```

```
<seaborn.axisgrid.FacetGrid at 0x7af54dbcff10>
```



```
sns.relplot(x="total",y="speeding",data=df,hue="abbrev")
```



4.968	2
7.332	1
9.100	1
5.439	1
4.060	1
1.792	1
3.496	1
3.936	1
6.552	1
5.497	1
3.948	1
6.368	1
4.224	1
3.774	1
8.346	1
9.082	1
6.014	1
4.095	1
7.760	1
4.859	1
4.080	1
2.413	1
4.452	1
8.092	1
1.937	1
6.923	1
7.421	1
2.640	1
6.510	1
4.032	1
4.200	1
5.032	1
6.156	1
2.006	1
3.759	1
2.964	1
9.450	1



```

5.508    1
4.608    1
3.625    1
2.669    1
4.806    1
4.066    1
7.175    1
5.738    1
4.250    1
1.886    1
3.384    1
2.208    1
7.308    1
Name: speeding, dtype: int64

```

```
df["not_distracted"].value_counts()
```

```

14.094    2
18.048    1
17.472    1
13.965    1
10.092    1
9.632     1
12.328    1
10.824    1
15.792    1
23.661    1
13.959    1
18.308    1
8.576     1
10.212    1
17.976    1
22.944    1
19.012    1
15.990    1
17.654    1
9.944     1
13.056    1
11.049    1
8.692     1
23.086    1
13.857    1
14.812    1
16.290    1
1.760     1
15.624    1
21.056    1
10.920    1
10.744    1
9.396     1
5.900     1
16.468    1
14.820    1
14.350    1
13.005    1
12.032    1
13.775    1
15.229    1
13.706    1
16.692    1
14.965    1
13.137    1
8.875     1
7.134     1
13.395    1
8.448     1
5.382     1
Name: not_distracted, dtype: int64

```

```
df["no_previous"].value_counts()
```

```

12.920    2
15.040    1
16.016    1
14.553    1
9.628     1
8.736     1
18.032    1
9.840     1
13.608    1
20.554    1
11.562    1

```

```

18.706 1
11.520 1
8.769 1
18.190 1
19.359 1
16.684 1
15.795 1
16.878 1
10.848 1
11.176 1
9.116 1
20.706 1
11.592 1
13.410 1
13.524 1
17.014 1
17.600 1
17.856 1
21.280 1
10.680 1
8.856 1
16.038 1
5.900 1
16.826 1
14.508 1
15.225 1
14.994 1
12.288 1
13.775 1
13.659 1
15.130 1
16.264 1
20.090 1
12.684 1
12.375 1
6.560 1
10.857 1
8.448 1
15.660 1
Name: no_previous, dtype: int64

```

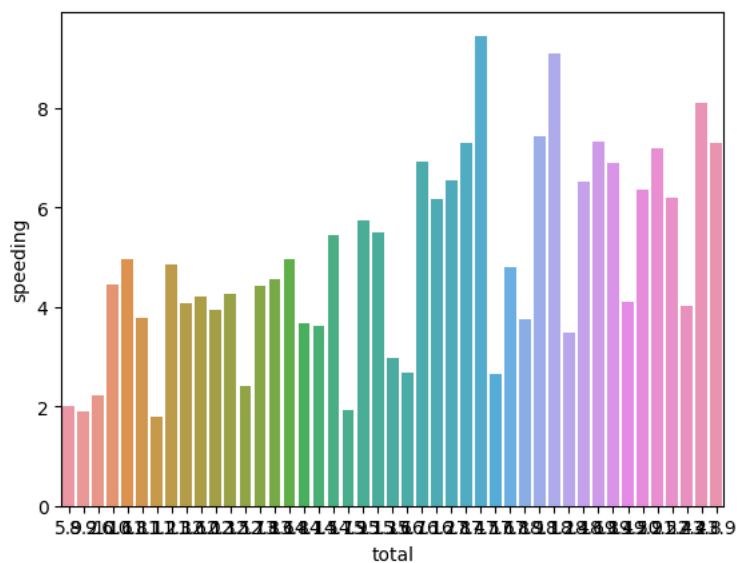
```
sns.barplot(data=df,x="total",y="speeding",ci=None)
```

```
<ipython-input-55-45580ba4c45b>:1: FutureWarning:
```

The `ci` parameter is deprecated. Use `errorbar=None` for the same effect.

```
sns.barplot(data=df,x="total",y="speeding",ci=None)
```

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

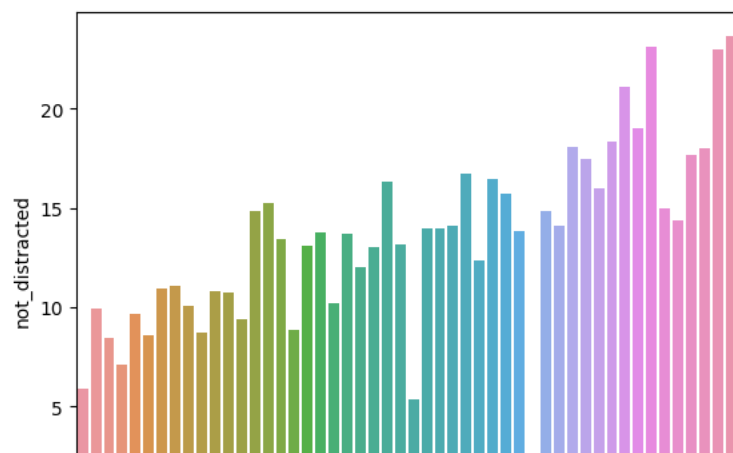


```
sns.barplot(data=df,x="alcohol",y="not_distracted",ci=None)
```

```
<ipython-input-39-c836539ef2b1>:1: FutureWarning:
```

The `ci` parameter is deprecated. Use `errorbar=None` for the same effect.

```
sns.barplot(data=df,x="alcohol",y="not_distracted",ci=None)
<Axes: xlabel='alcohol', ylabel='not_distracted'>
```

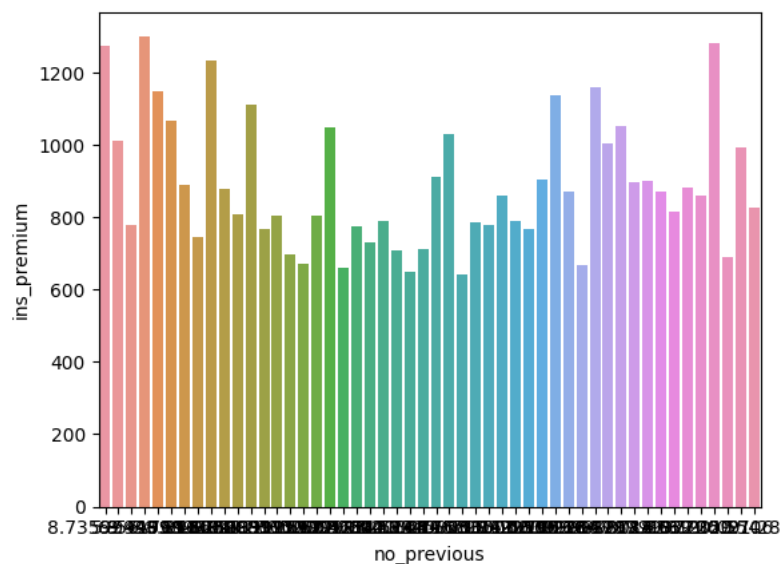


```
sns.barplot(data=df,x="no_previous",y="ins_premium",ci=None)
```

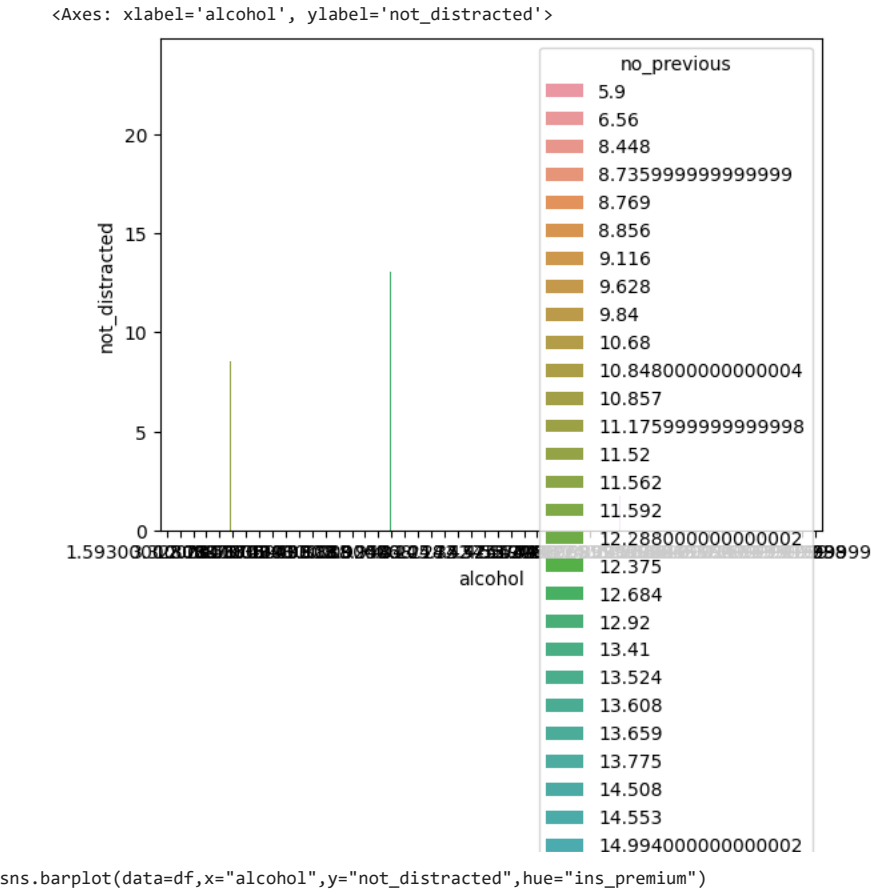
```
<ipython-input-40-560e315f829c>:1: FutureWarning:
```

The `ci` parameter is deprecated. Use `errorbar=None` for the same effect.

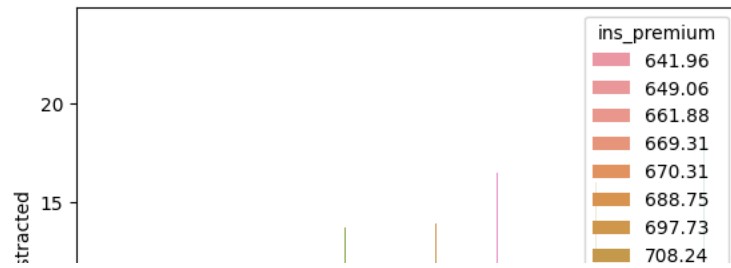
```
sns.barplot(data=df,x="no_previous",y="ins_premium",ci=None)
<Axes: xlabel='no_previous', ylabel='ins_premium'>
```



```
sns.barplot(data=df,x="alcohol",y="not_distracted",hue="no_previous")
```

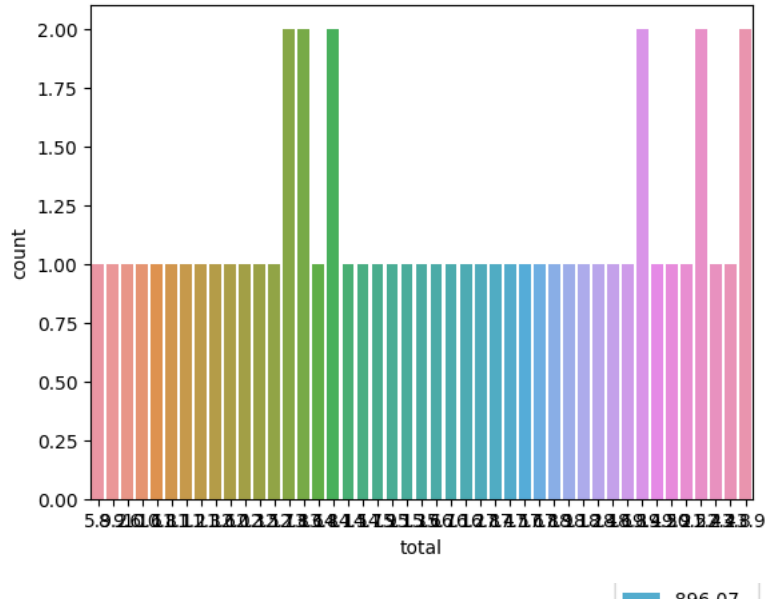


<Axes: xlabel='alcohol', ylabel='not\_distracted'>



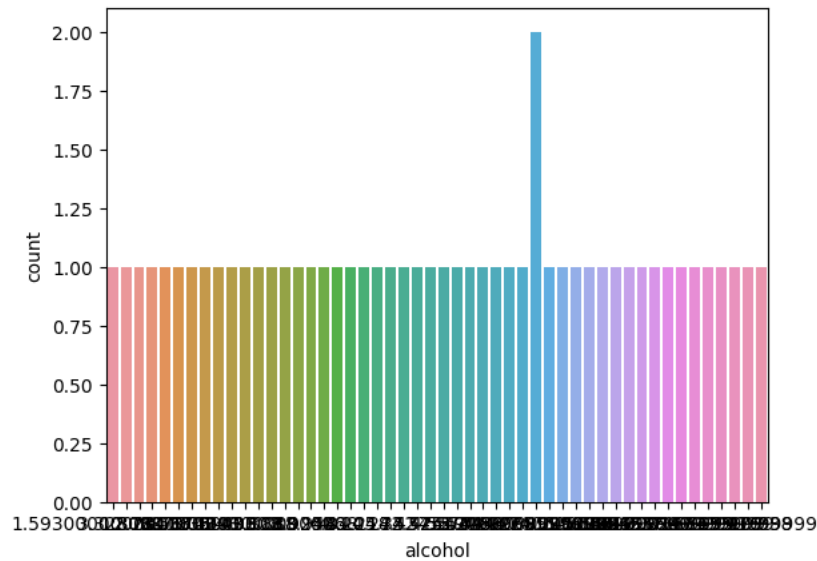
sns.countplot(x="total",data=df)

<Axes: xlabel='total', ylabel='count'>



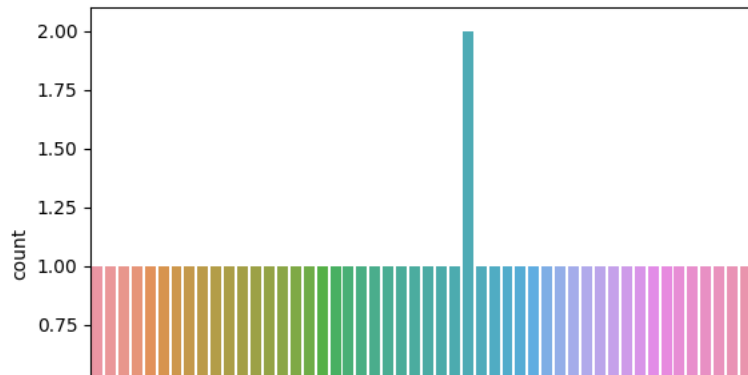
sns.countplot(x="alcohol",data=df)

<Axes: xlabel='alcohol', ylabel='count'>



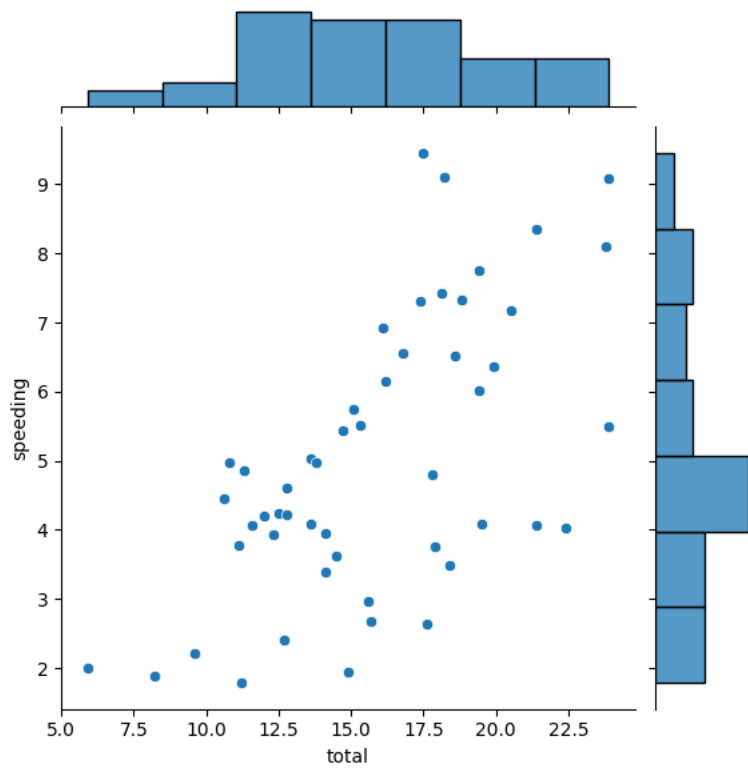
sns.countplot(x="not\_distracted",data=df)

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



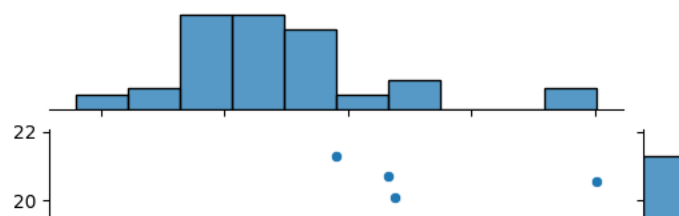
```
sns.jointplot(x="total",y="speeding",data=df)
```

```
<seaborn.axisgrid.JointGrid at 0x7af544aa2440>
```



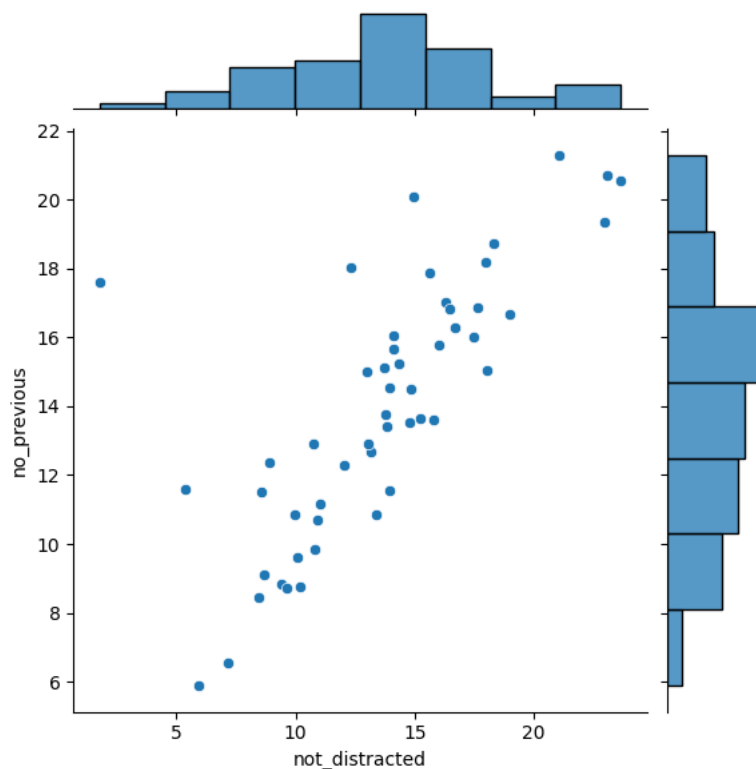
```
sns.jointplot(x="alcohol",y="no_previous",data=df)
```

```
<seaborn.axisgrid.JointGrid at 0x7af544aa3910>
```



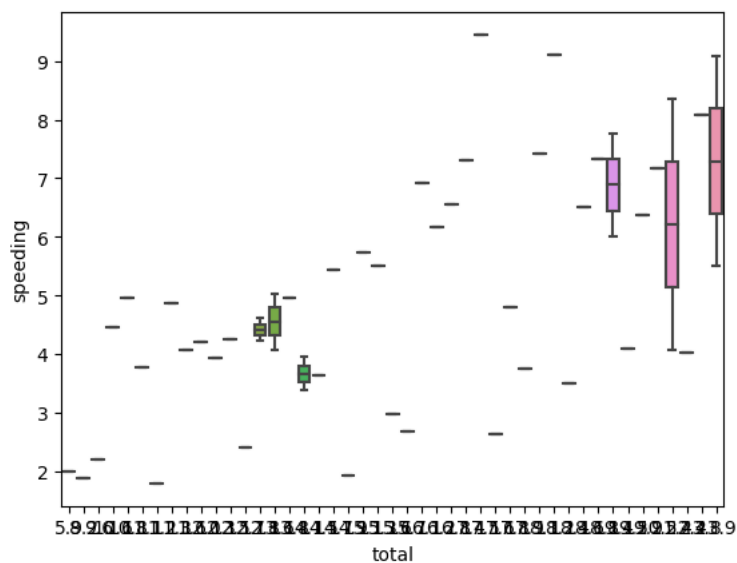
```
sns.jointplot(x="not_distracted",y="no_previous",data=df)
```

```
<seaborn.axisgrid.JointGrid at 0x7af544957df0>
```



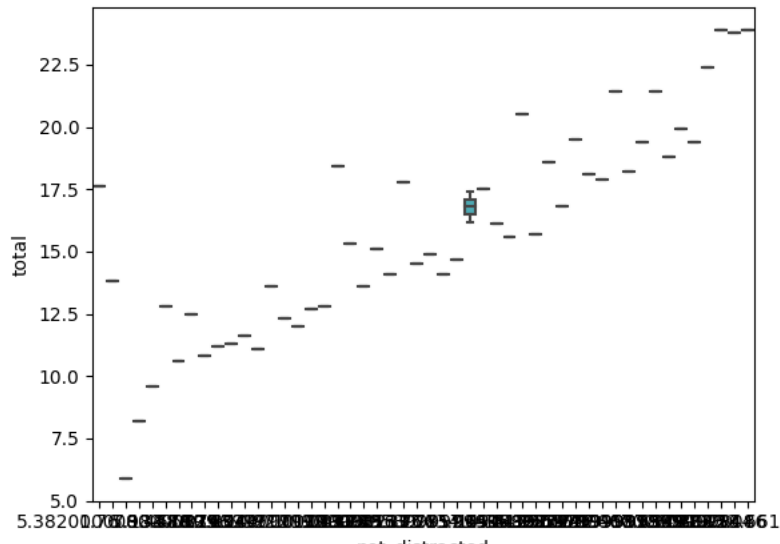
```
sns.boxplot(x="total",y="speeding",data=df)
```

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



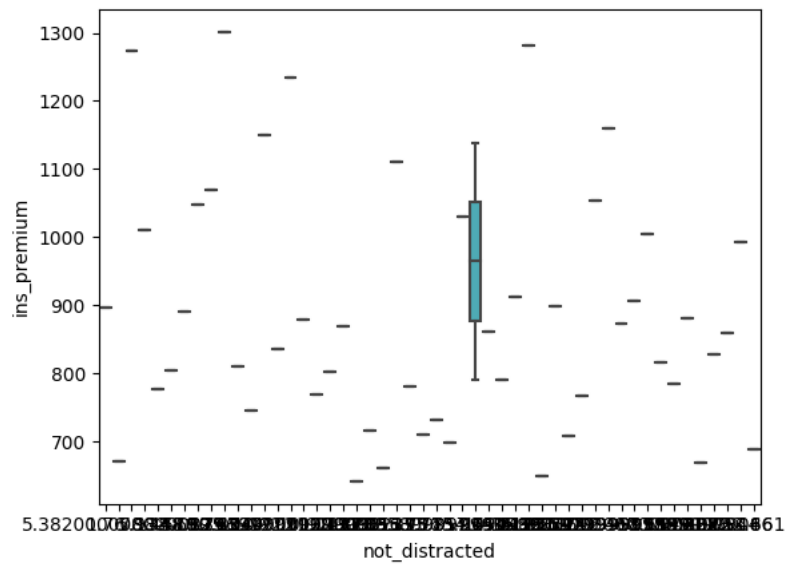
```
sns.boxplot(x="not_distracted",y="total",data=df)
```

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



```
sns.boxplot(x="not_distracted",y="ins_premium",data=df)
```

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



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



