

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

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



	total	speeding	alcohol	not_distracted	no_previous	ins_premium	ins_losses
0	18.8	7.332	5.640	18.048	15.040	784.55	145.08
1	18.1	7.421	4.525	16.290	17.014	1053.48	133.93
2	18.6	6.510	5.208	15.624	17.856	899.47	110.35
3	22.4	4.032	5.824	21.056	21.280	827.34	142.39
4	12.0	4.200	3.360	10.920	10.680	878.41	165.63
5	13.6	5.032	3.808	10.744	12.920	835.50	139.91
6	10.8	4.968	3.888	9.396	8.856	1068.73	167.02
7	16.2	6.156	4.860	14.094	16.038	1137.87	151.48

```
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
```

```
18    20.5    7.175    6.765    14.965    20.090    1281.55    194.78
```

```
sns.barplot(x='abbrev', y='total', data=df)
```

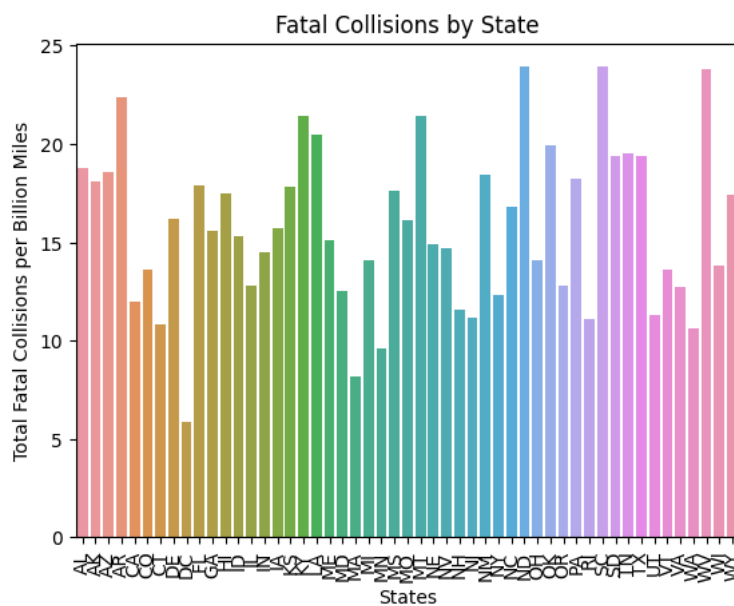
```
plt.xlabel('States')
```

```
plt.ylabel('Total Fatal Collisions per Billion Miles')
```

```
plt.title('Fatal Collisions by State')
```

```
plt.xticks(rotation=90)
```

```
plt.show()
```



Inference: ND has the most number of accidents

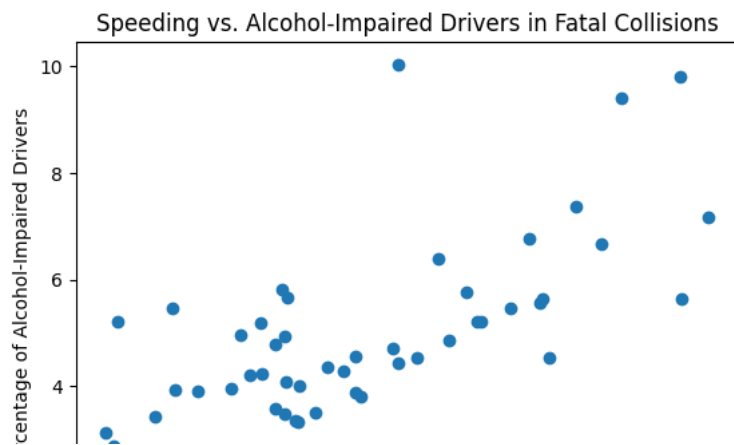
```
plt.scatter(df['speeding'], df['alcohol'])
```

```
plt.xlabel('Percentage of Speeding Drivers')
```

```
plt.ylabel('Percentage of Alcohol-Impaired Drivers')
```

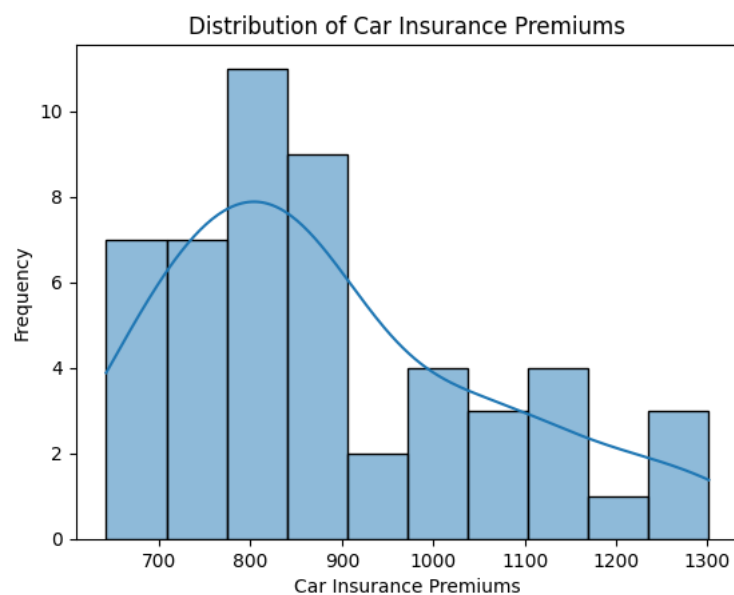
```
plt.title('Speeding vs. Alcohol-Impaired Drivers in Fatal Collisions')
```

```
plt.show()
```



Inference: Most drivers speeding within the range of 5-6% are alcohol impaired

```
sns.histplot(df['ins_premium'], bins=10, kde=True)
plt.xlabel('Car Insurance Premiums')
plt.ylabel('Frequency')
plt.title('Distribution of Car Insurance Premiums')
plt.show()
```



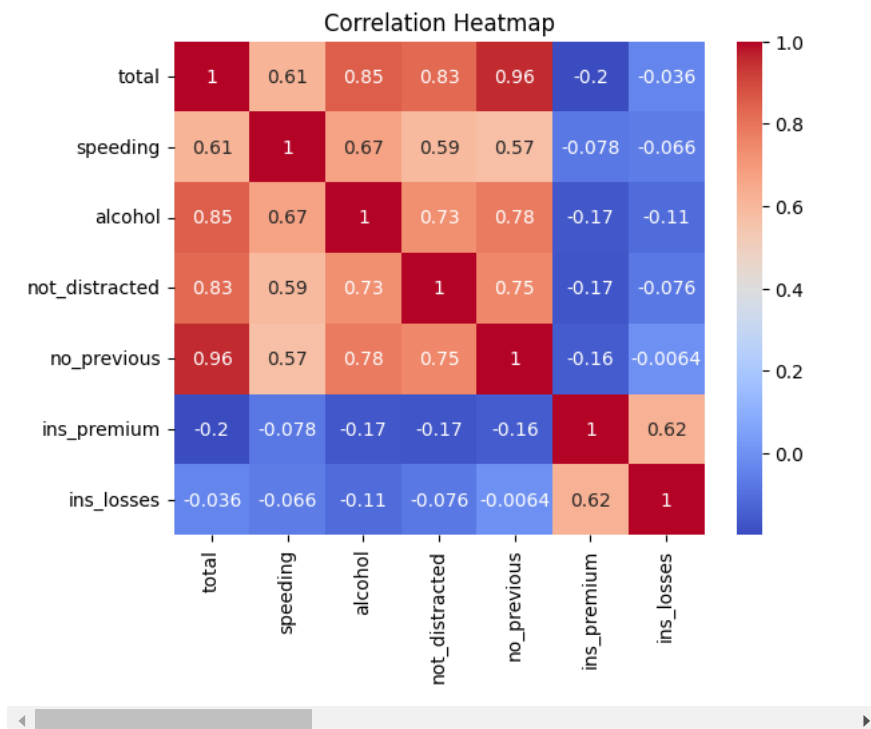
Inference: 800 car insurance were taken over 10 times.

```
sns.boxplot(y='ins_losses', data=df)
plt.ylabel('Insurance Losses')
plt.title('Distribution of Insurance Losses')
plt.show()
```

Distribution of Insurance Losses

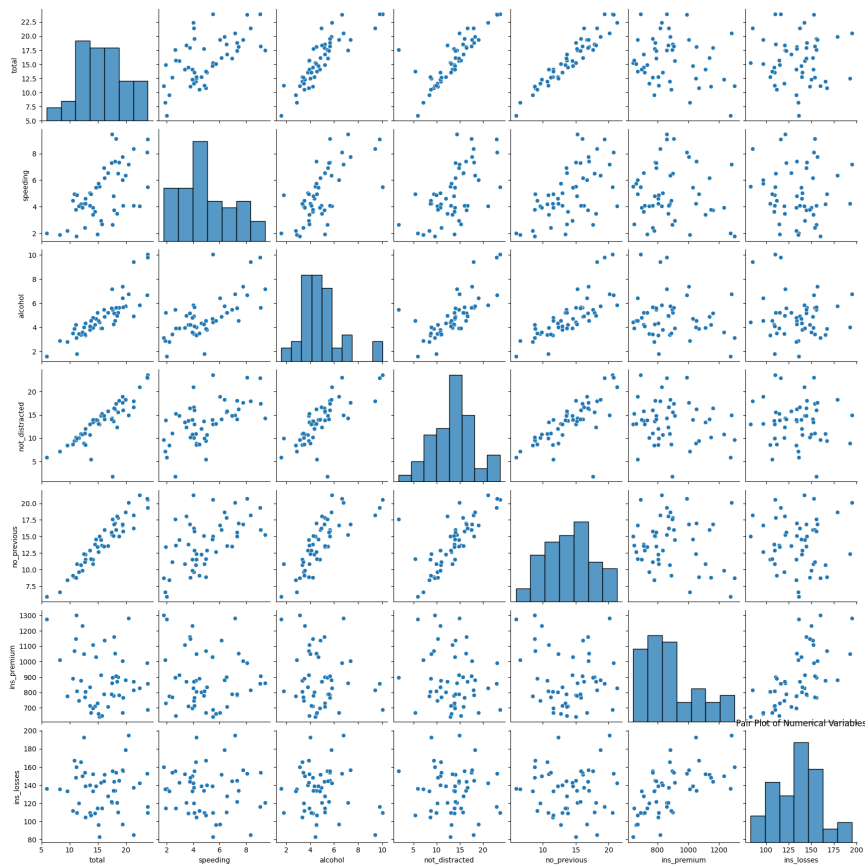
```
correlation_matrix = df.corr()
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm')
plt.title('Correlation Heatmap')
plt.show()
```

<ipython-input-8-182fd031f822>:1: FutureWarning: The default value of numeric_only in
correlation_matrix = df.corr()



Inference: Heatmap showing the dependency of different factors.

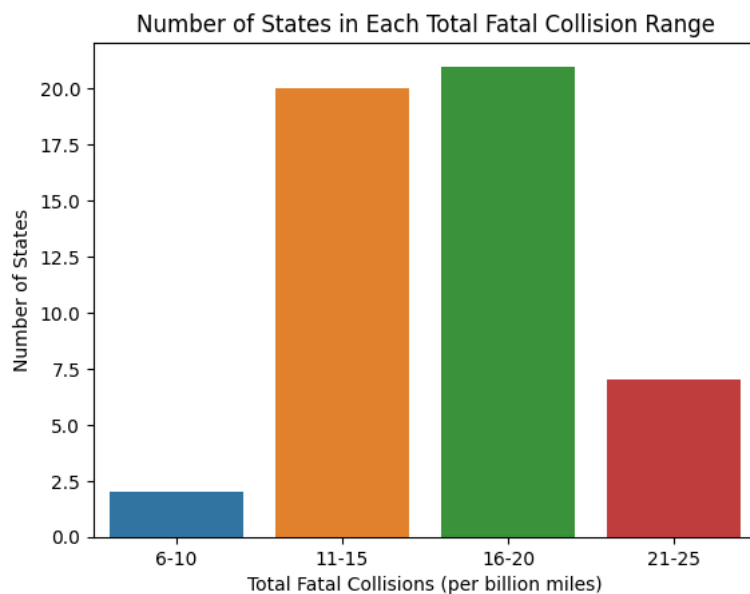
```
sns.pairplot(df[['total', 'speeding', 'alcohol', 'not_distracted', 'no_previous', 'ins_premium', 'ins_losses']])
plt.title('Pair Plot of Numerical Variables')
plt.show()
```



Inference: Joint plot

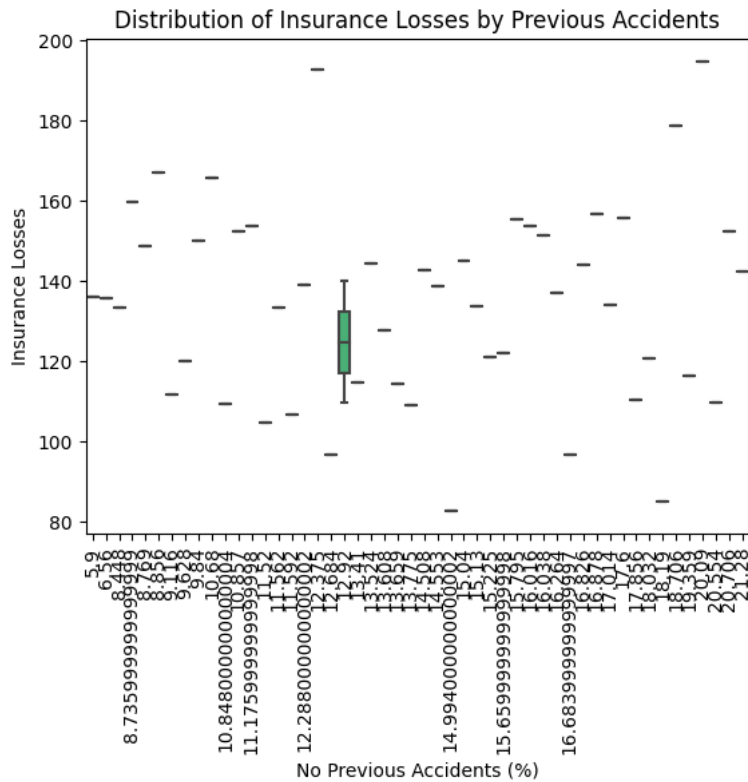
```
bins = [5.900, 10, 15, 20, 25]
labels = ['6-10', '11-15', '16-20', '21-25']
df['total_bins'] = pd.cut(df['total'], bins=bins, labels=labels)

sns.countplot(x='total_bins', data=df)
plt.xlabel('Total Fatal Collisions (per billion miles)')
plt.ylabel('Number of States')
plt.title('Number of States in Each Total Fatal Collision Range')
plt.show()
```

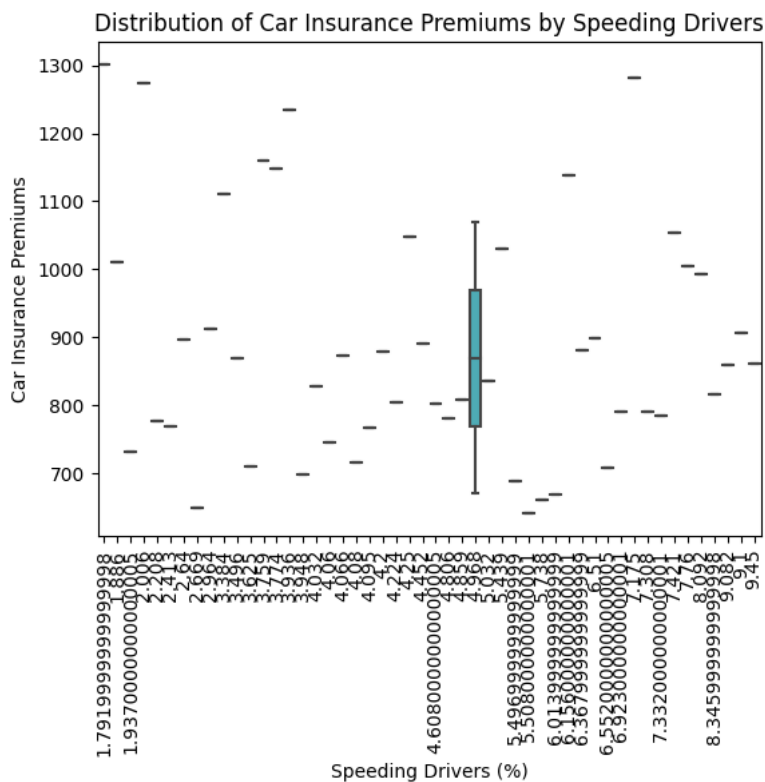


Inference: around 20 states have 16-20 fatal collisions.

```
sns.boxplot(x='no_previous', y='ins_losses', data=df)
plt.xlabel('No Previous Accidents (%)')
plt.ylabel('Insurance Losses')
plt.title('Distribution of Insurance Losses by Previous Accidents')
plt.xticks(rotation=90)
plt.show()
```

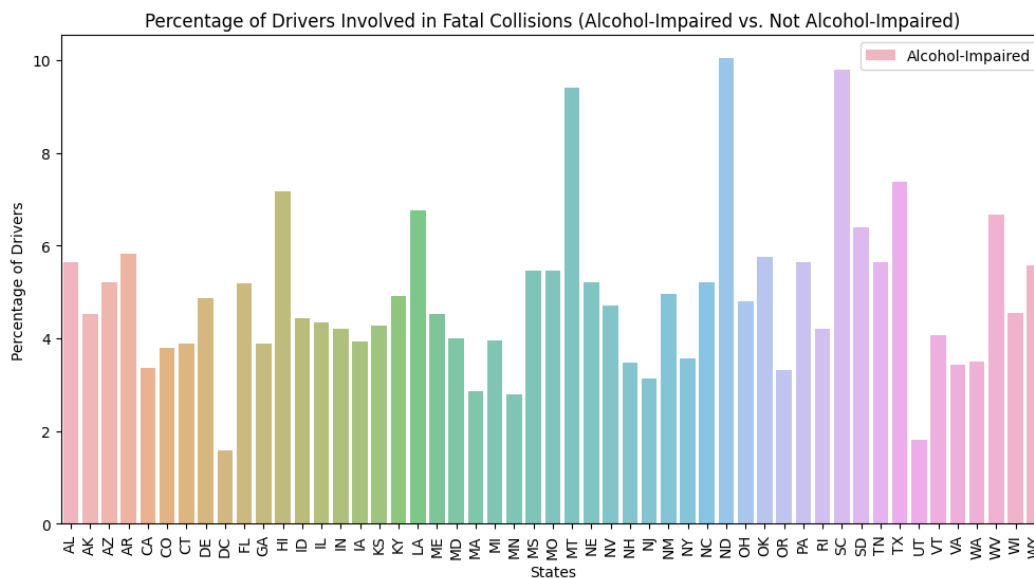


```
sns.boxplot(x='speeding', y='ins_premium', data=df)
plt.xlabel('Speeding Drivers (%)')
plt.ylabel('Car Insurance Premiums')
plt.title('Distribution of Car Insurance Premiums by Speeding Drivers')
plt.xticks(rotation=90)
plt.show()
```



```
plt.figure(figsize=(12, 6))
sns.barplot(x='abbrev', y='alcohol', data=df, label='Alcohol-Impaired', alpha=0.7)

plt.xlabel('States')
plt.ylabel('Percentage of Drivers')
plt.title('Percentage of Drivers Involved in Fatal Collisions (Alcohol-Impaired vs. Not Alcohol-Impaired)')
plt.xticks(rotation=90)
plt.legend()
plt.show()
```



Inference: ND is has the most fatal accidents by alcochol impaired drivers.