Assignment 2

Data Visualisation

UMMALETI KUMAR

kumar.21bce9309@vitapstudent.ac.in

In [1]:

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

In [2]:

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

```
['anagrams', 'anscombe', 'attention', 'brain_networks', 'car_crashes', 'diamonds', 'dots', 'dowjones', 'exercise', 'fli
ghts', 'fmri', 'geyser', 'glue', 'healthexp', 'iris', 'mpg', 'penguins', 'planets', 'seaice', 'taxis', 'tips', 'titani
c']
```

In [3]:

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

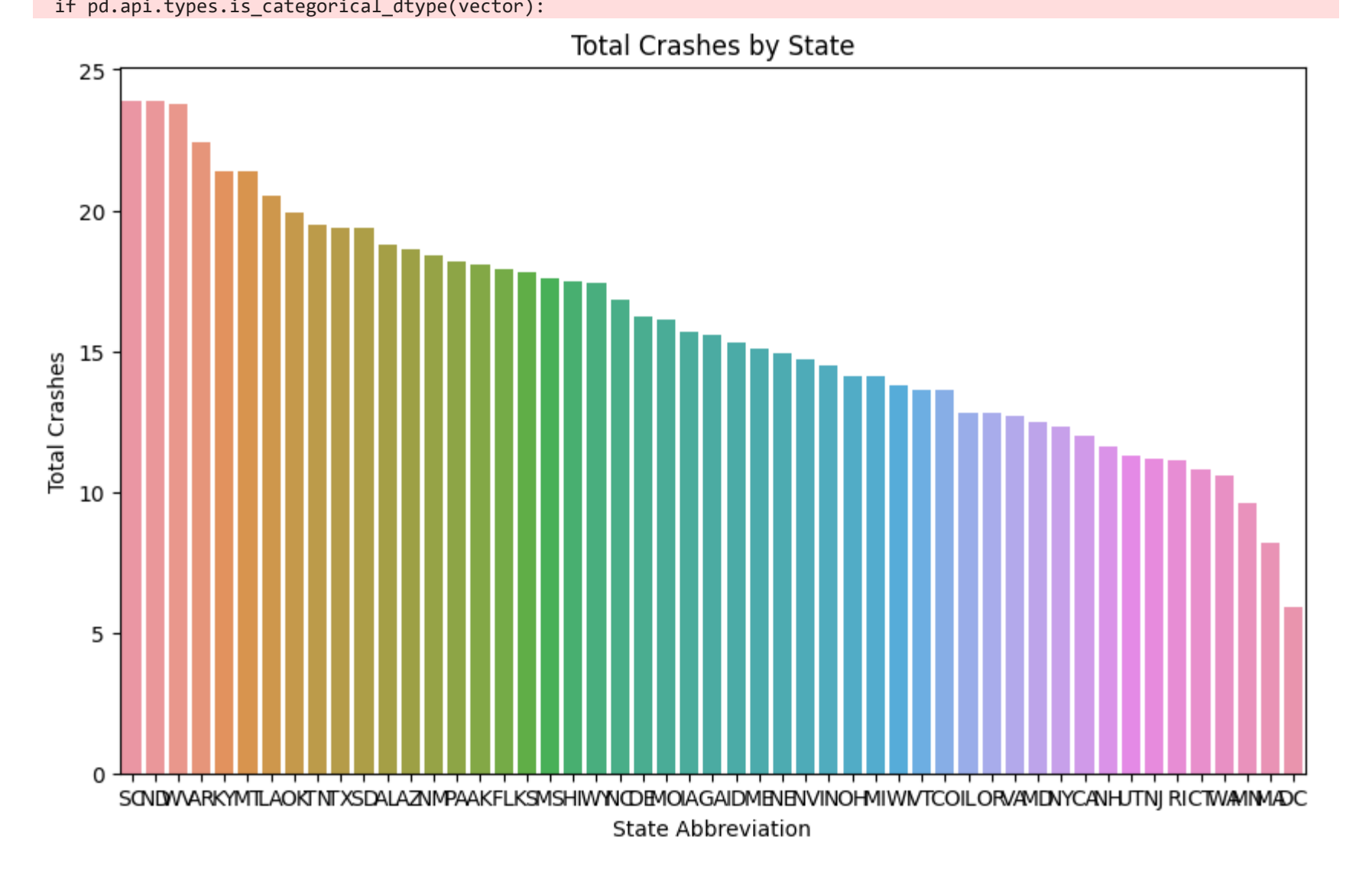
Out[3]:

	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
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 [5]:

```
plt.figure(figsize=(10, 6))
sns.barplot(x='abbrev', y='total', data=df.sort_values(by='total', ascending=False))
plt.xlabel('State Abbreviation')
plt.ylabel('Total Crashes')
plt.title('Total Crashes by State')
plt.show()
```

```
C:\Users\kumar\AppData\Local\Programs\Python\Python310\lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning: is_ca
tegorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\kumar\AppData\Local\Programs\Python\Python310\lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning: is_ca
tegorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\kumar\AppData\Local\Programs\Python\Python310\lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning: is_ca
tegorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
```

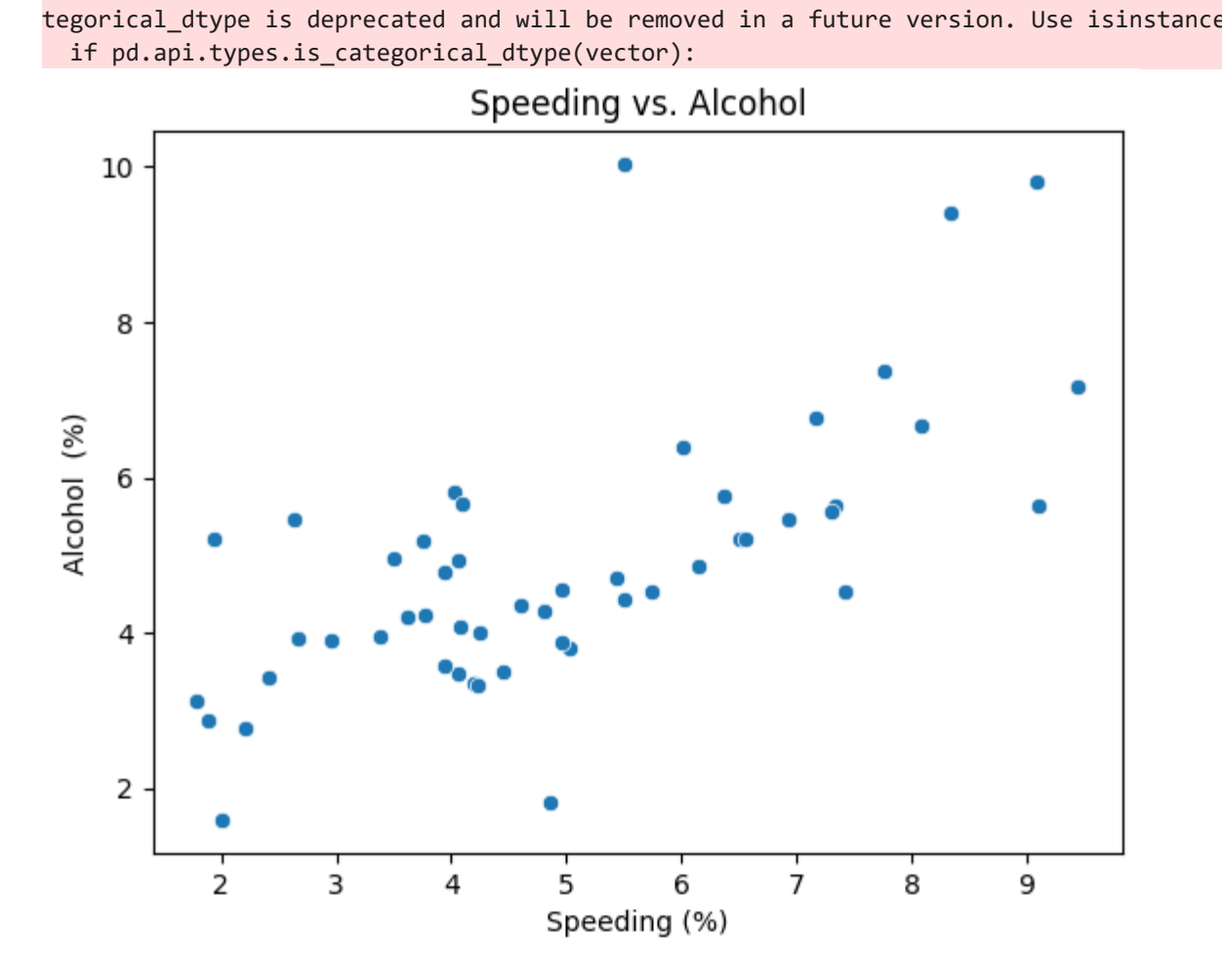


Here we can see that highest no.of crashes of 24 are in state SC ,ND And lowest no.of crashes of 6 in state DC

In [6]:

```
sns.scatterplot(x='speeding',y='alcohol',data=df)
plt.xlabel('Speeding (%)')
plt.ylabel('Alcohol  (%)')
plt.title('Speeding vs. Alcohol')
plt.show()
```

```
C:\Users\kumar\AppData\Local\Programs\Python\Python310\lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning: is_ca
tegorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\kumar\AppData\Local\Programs\Python\Python310\lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning: is_ca
tegorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
```

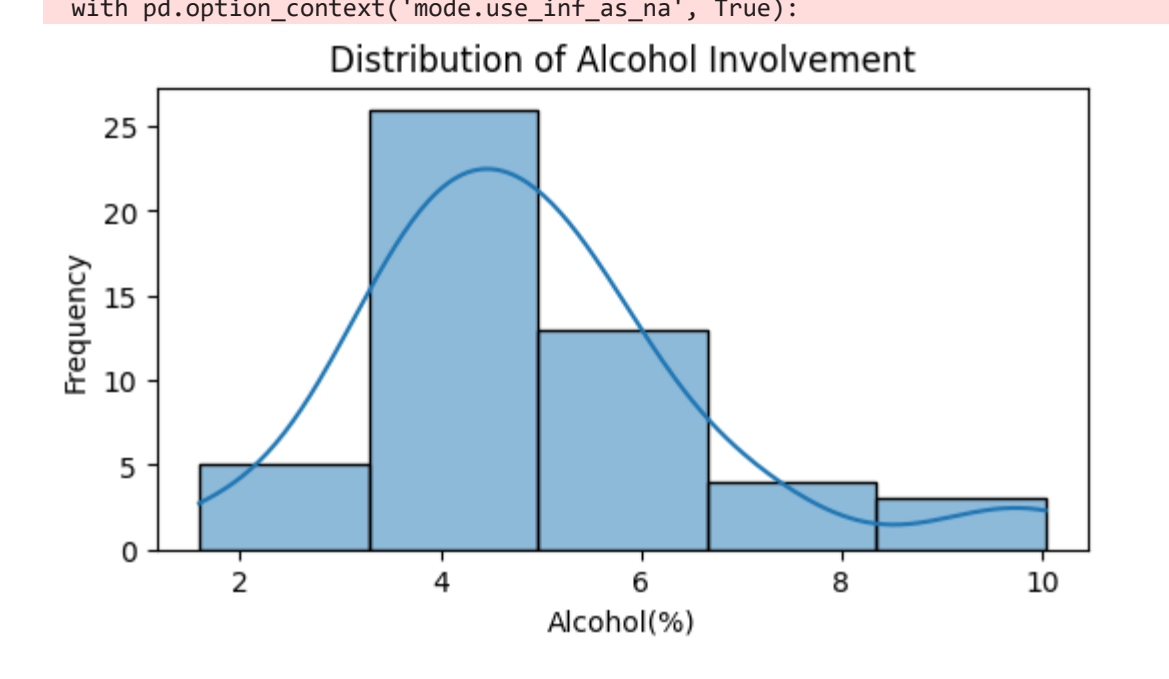


Here we can see there are more crashes at speeding and alcohol between 4-5 percent

In [7]:

```
plt.figure(figsize=(6, 3))
sns.histplot(df['alcohol'], bins=5, kde=True)
plt.xlabel('Alcohol(%)')
plt.ylabel('Frequency')
plt.title('Distribution of Alcohol Involvement')
plt.show()
```

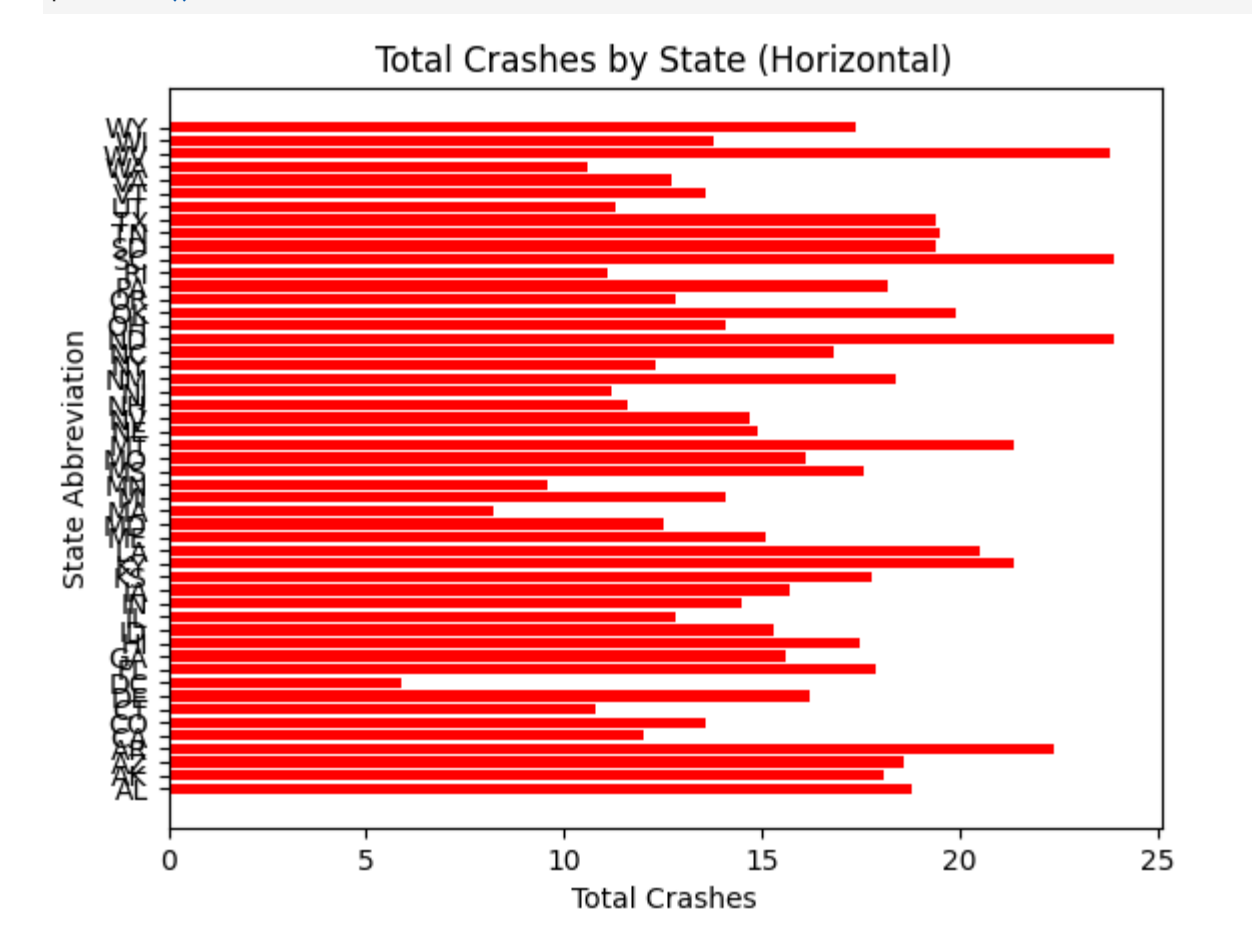
```
C:\Users\kumar\AppData\Local\Programs\Python\Python310\lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning: is_ca
tegorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\kumar\AppData\Local\Programs\Python\Python310\lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_i
nf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating inste
ad.
  with pd.option_context('mode.use_inf_as_na', True):
```



Here we can see in the histogram that alcohol percentage is mostly between 4-5 percent

In [8]:

```
plt.barh(df['abbrev'], df['total'],color="red")
plt.xlabel('Total Crashes')
plt.ylabel('State Abbreviation')
plt.title('Total Crashes by State (Horizontal)')
plt.show()
```

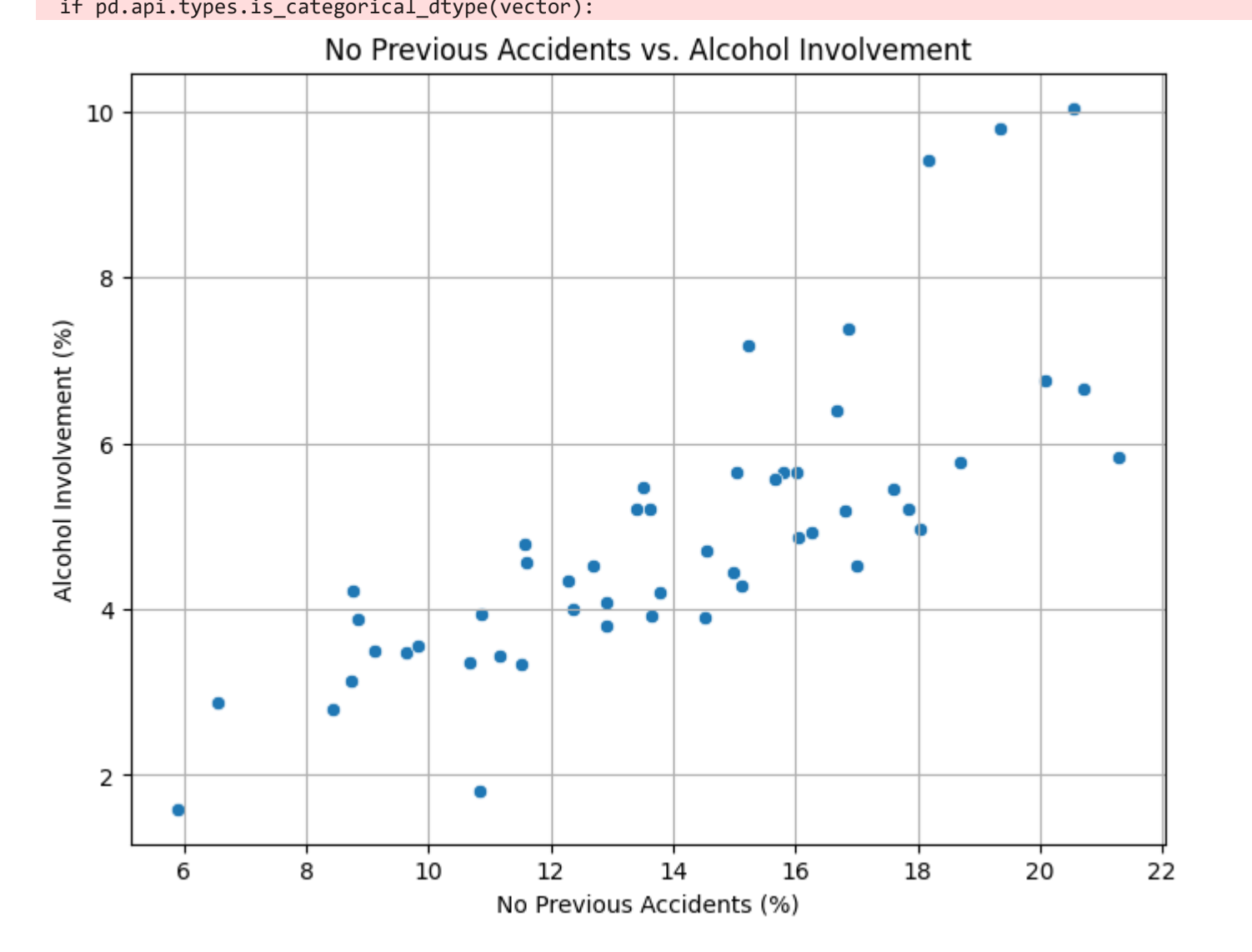


Here we can see that highest no.of crashes of 24 are in state SC And lowest no.of crashes of 6 in state DC

In [9]:

```
x = df['no_previous']
y = df['alcohol']

plt.figure(figsize=(8, 6))
sns.scatterplot(x=x, y=y)
plt.xlabel('No Previous Accidents (%)')
plt.ylabel('Alcohol Involvement (%)')
plt.title('No Previous Accidents vs. Alcohol Involvement')
plt.grid(True)
plt.show()
```

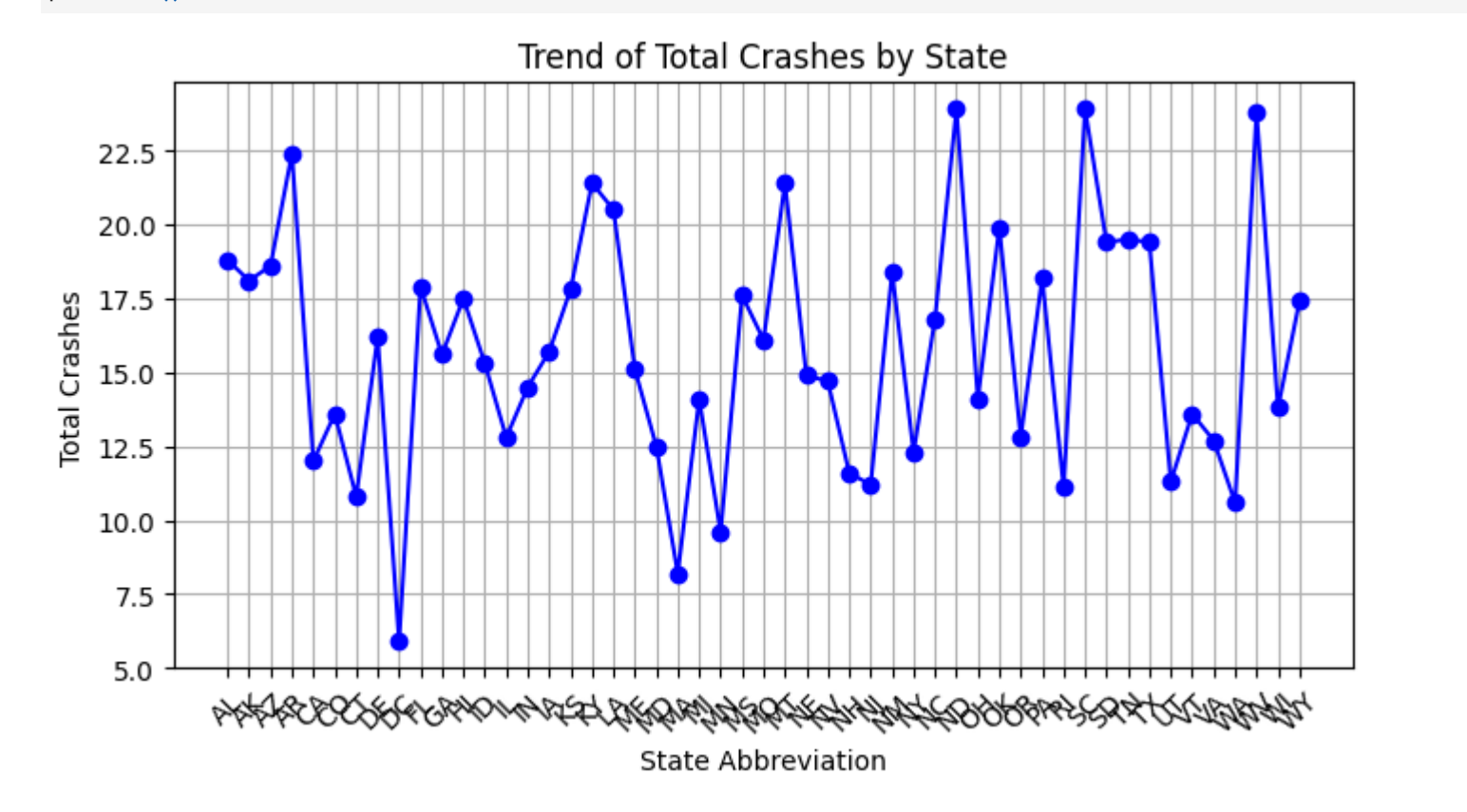
```
C:\Users\kumar\AppData\Local\Programs\Python\Python310\lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning: is_ca
tegorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\kumar\AppData\Local\Programs\Python\Python310\lib\site-packages\seaborn\_oldcore.py:1498: FutureWarning: is_ca
tegorical_dtype is deprecated and will be removed in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
```



Here we can see that Accidents percentage is less when the alcohol consumption is less

In [10]:

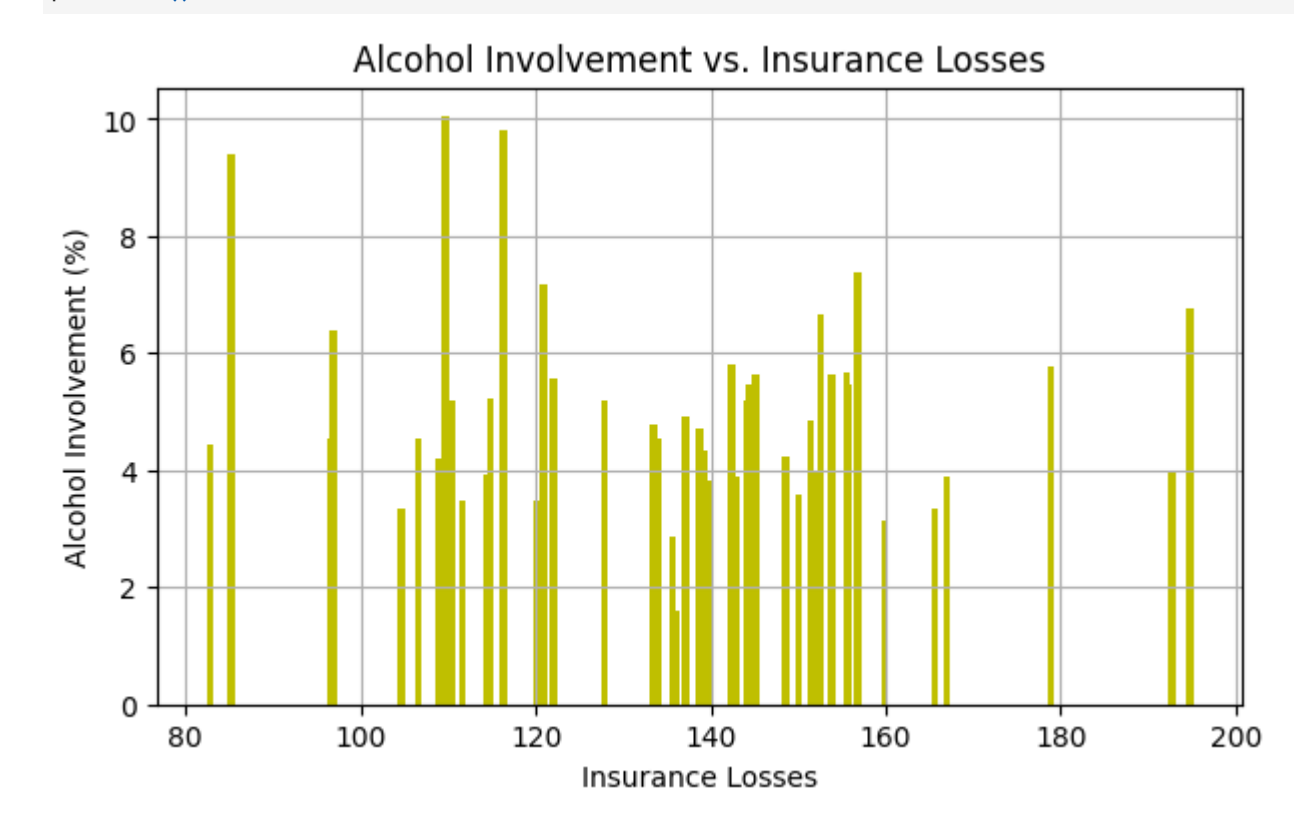
```
x = df['abbrev']
y = df['total']

# Create a line plot
plt.figure(figsize=(8, 4))
plt.plot(x, y, marker='o', linestyle='-', color='b')
plt.xlabel('State Abbreviation')
plt.ylabel('Total Crashes')
plt.title('Trend of Total Crashes by State')
plt.xticks(rotation=45)
plt.grid(True)
plt.show()
```



This is the lineplot between total crashes and States

In [15]:

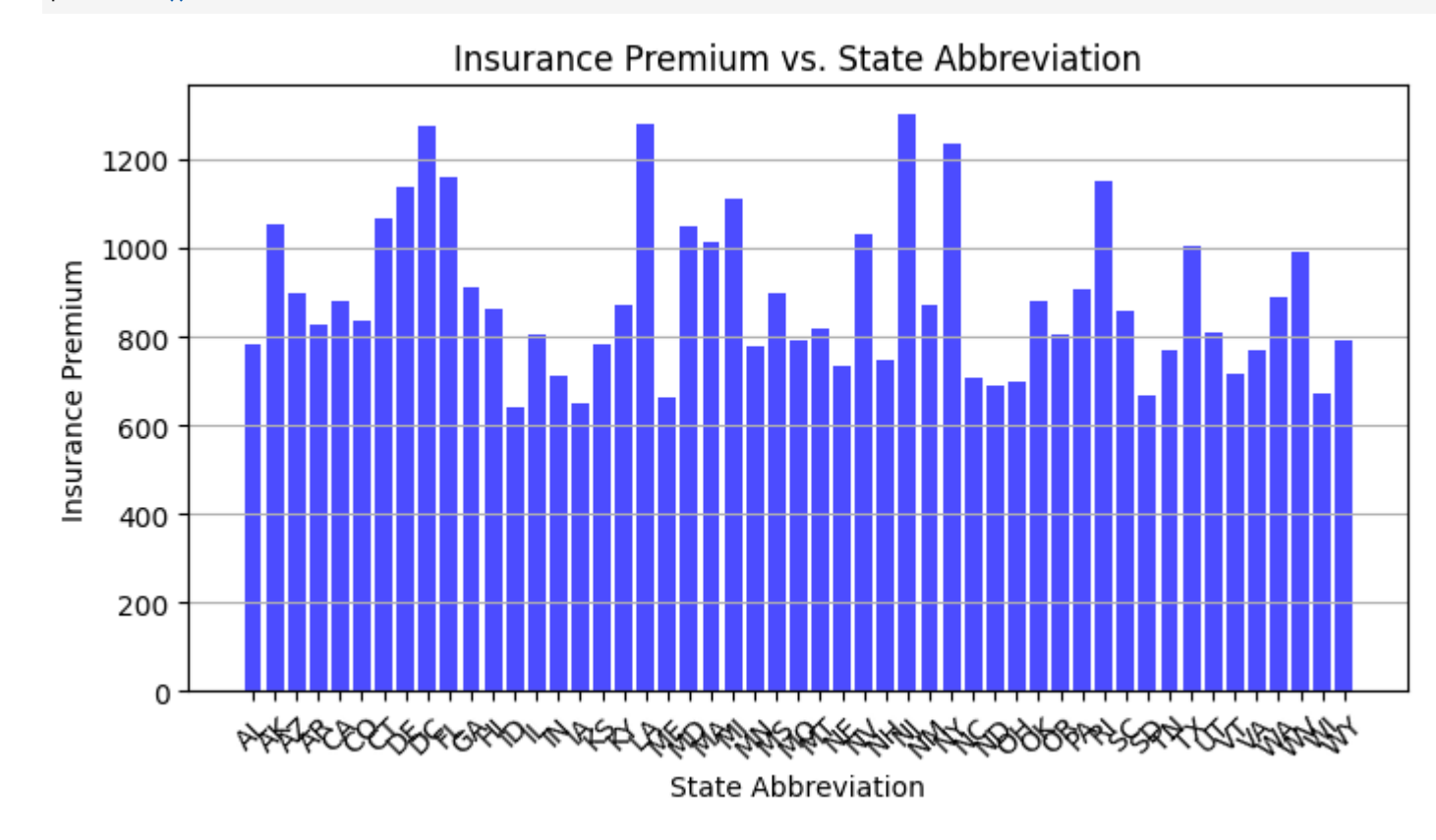
```
plt.figure(figsize=(7, 4))
plt.bar( df['ins_losses'],df['alcohol'], color='y')
plt.ylabel('Alcohol Involvement (%)')
plt.xlabel('Insurance Losses')
plt.title('Alcohol Involvement vs. Insurance Losses')
plt.grid(True)
plt.show()
```



Insurance loss is more for the Alcohol consumption

In [16]:

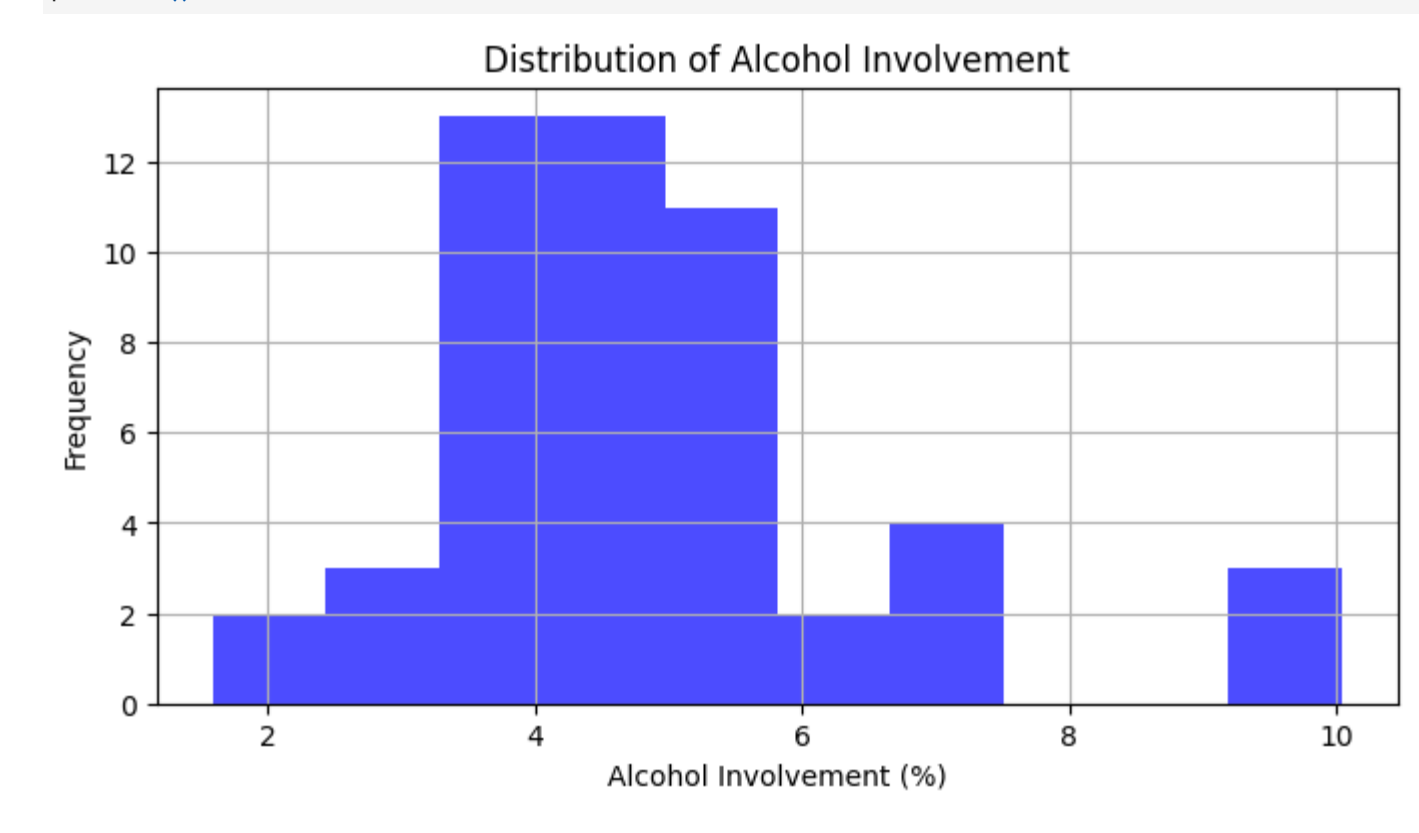
```
plt.figure(figsize=(8, 4))
plt.bar(df['abbrev'], df['ins_premium'], color='b', alpha=0.7)
plt.xlabel('State Abbreviation')
plt.ylabel('Insurance Premium')
plt.title('Insurance Premium vs. State Abbreviation')
plt.xticks(rotation=45)
plt.grid(axis='y')
plt.show()
```



From this we can say that almost many states have insurance premium for more than 600 people The highest percentage is for state NI And the lowest percentage is for the state TD

In [17]:

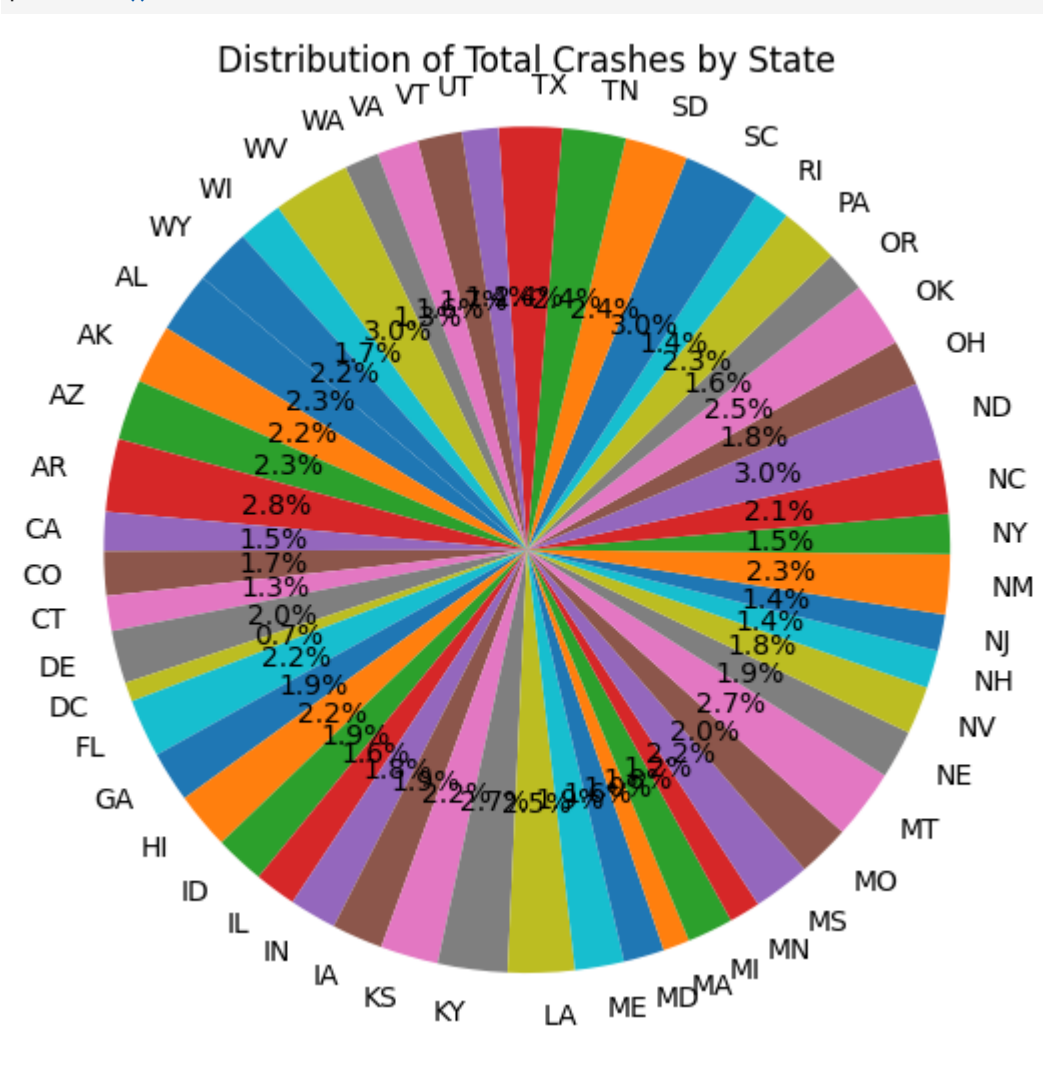
```
plt.figure(figsize=(8, 4))
plt.hist(df['alcohol'], bins=10, color='b', alpha=0.7)
plt.xlabel('Alcohol Involvement (%)')
plt.ylabel('Frequency')
plt.title('Distribution of Alcohol Involvement')
plt.grid(True)
plt.show()
```



The maximum alcohol consumption is between 4-5

In [18]:

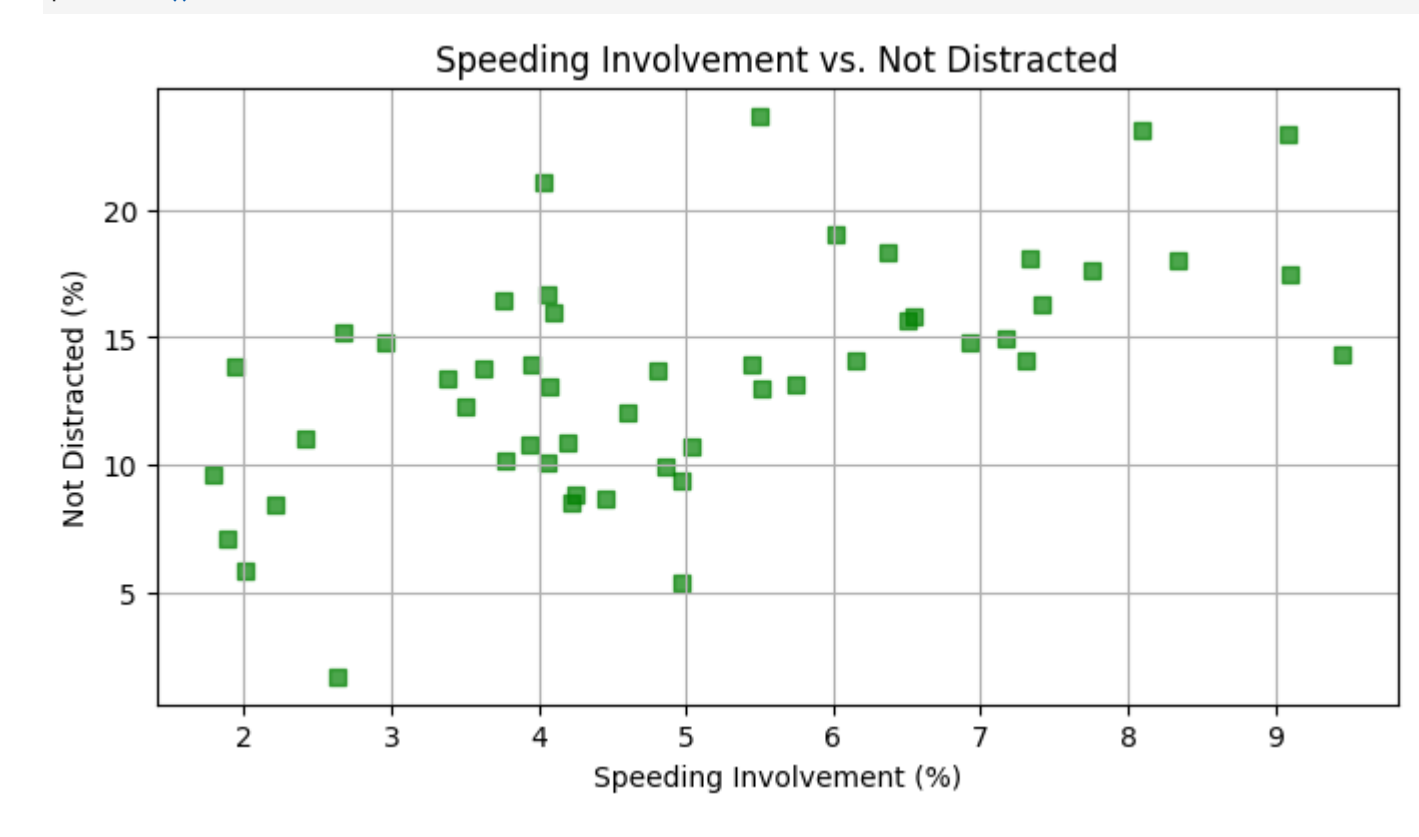
```
plt.figure(figsize=(6, 6))
plt.pie(df['total'], labels=df['abbrev'], autopct='%1.1f%%', startangle=140)
plt.axis('equal')
plt.title('Distribution of Total Crashes by State')
plt.show()
```



The states with maximum percentage of crashes are WV , SC with 3.0% The states with minimum percentage of crashes are DC with 0.7%

In [19]:

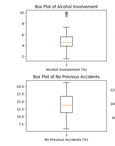
```
plt.figure(figsize=(8, 4))
plt.scatter(df['speeding'], df['not_distracted'], marker='s', color='g', alpha=0.7)
plt.xlabel('Speeding Involvement (%)')
plt.ylabel('Not Distracted (%)')
plt.title('Speeding Involvement vs. Not Distracted')
plt.grid(True)
plt.show()
```

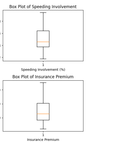


We can see that people with maintaining less speed have high chances of not to distract

In [20]:

```
plt.figure(figsize=(8, 6))

# Box Plot 1: Alcohol Involvement
plt.subplot(2,2,1)
plt.boxplot(df['alcohol'])
plt.xlabel('Alcohol Involvement (%)')
plt.title('Box Plot of Alcohol Involvement')

# Box Plot 2: Speeding Involvement
plt.subplot(2,2,2)
plt.boxplot(df['speeding'])
plt.xlabel('Speeding Involvement (%)')
plt.title('Box Plot of Speeding Involvement')

# Box Plot 3: No Previous Accidents
plt.subplot(2,2,3)
plt.boxplot(df['no_previous'])
plt.xlabel('No Previous Accidents (%)')
plt.title('Box Plot of No Previous Accidents')

# Box Plot 4: Insurance Premium
plt.subplot(2,2,4)
plt.boxplot(df['ins_premium'])
plt.xlabel('Insurance Premium')
plt.title('Box Plot of Insurance Premium')

# Adjust spacing between subplots
plt.tight_layout()

# Show the subplots
plt.show()
```





Here we can see that for ALCOHOL minimum percentage is : 1.59 1st Quadratile: 3.89 Median: 4.56 3rd Quadratile: 5.60 Maximum : 10.03

for Speed minimum percentage is : 1.79 1st Quadratile : 3.76 Median : 4.60 3rd Quadratile: 6.40 Maximum : 9.45

for No previous Accidents minimum percentage is : 5.24 1st Quadratile : 11.34 Median : 13.77 3rd Quadratile : 16.75 Maximum 21.28

for Insurance premium Minimum 641.96 1st Quartile : 768.40 Median : 858.97 3rd Quartile : 1007.94 Maximum 1307.52