```
# Importing Seaborn and matplotlib packages
import seaborn as sns
import matplotlib.pyplot as plt
```

```
# Loading the car crashes dataset
df = sns.load_dataset('car_crashes')
```

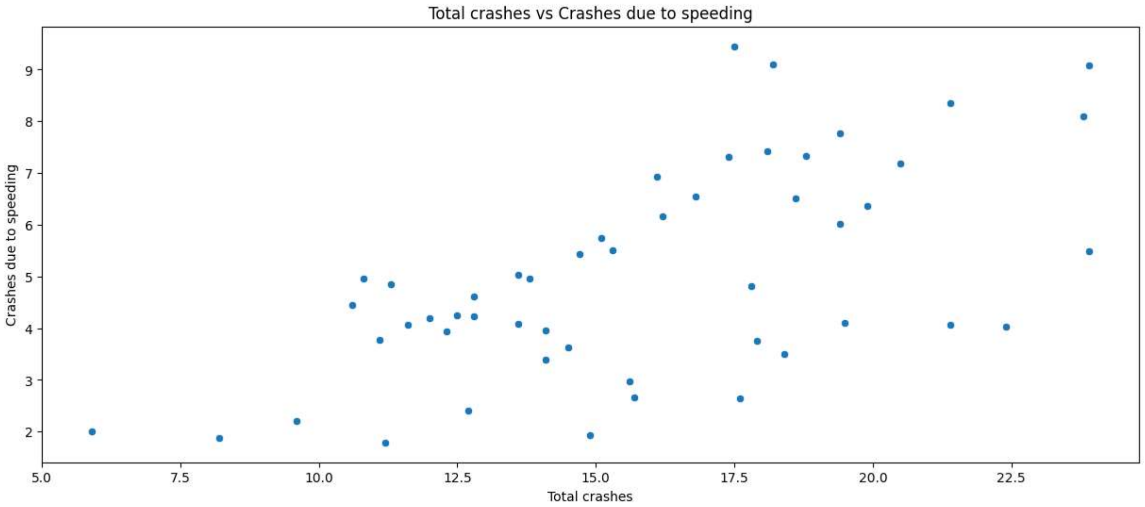
```
df.info()
```

```
<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()
```

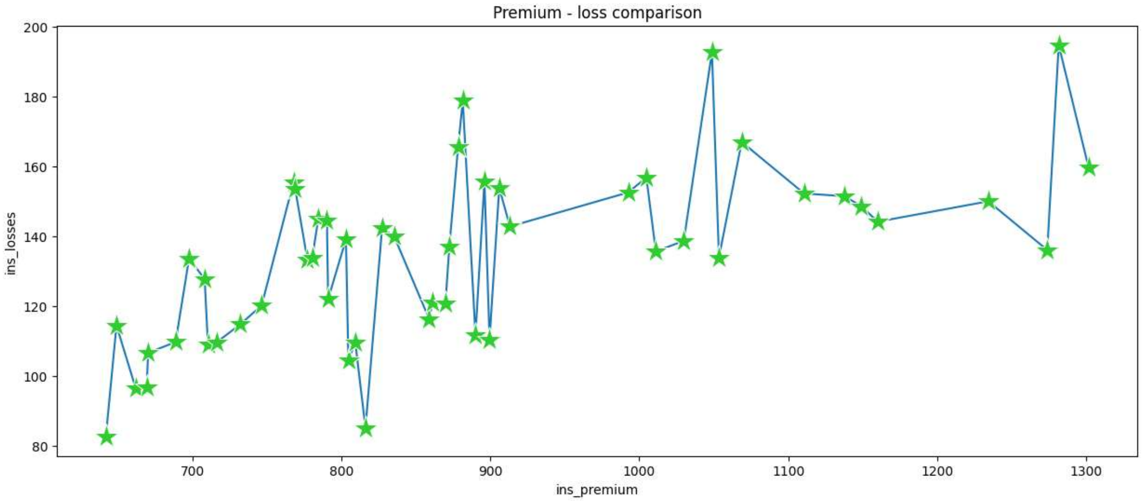
	total	speeding	alcohol	not_distracted	no_previous	ins_premium	ins_losses	abbrev
0	18.8	7.332	5.640	18.048	15.040	784.55	145.08	AL
1	18.1	7.421	4.525	16.290	17.014	1053.48	133.93	AK
2	18.6	6.510	5.208	15.624	17.856	899.47	110.35	AZ
3	22.4	4.032	5.824	21.056	21.280	827.34	142.39	AR
4	12.0	4.200	3.360	10.920	10.680	878.41	165.63	CA

```
# Scatter plot
plt.figure(figsize=(15, 6))
sns.scatterplot(x='total', y='speeding', data=df)
plt.xlabel('Total crashes')
plt.ylabel('Crashes due to speeding')
plt.title('Total crashes vs Crashes due to speeding')
plt.show()
```



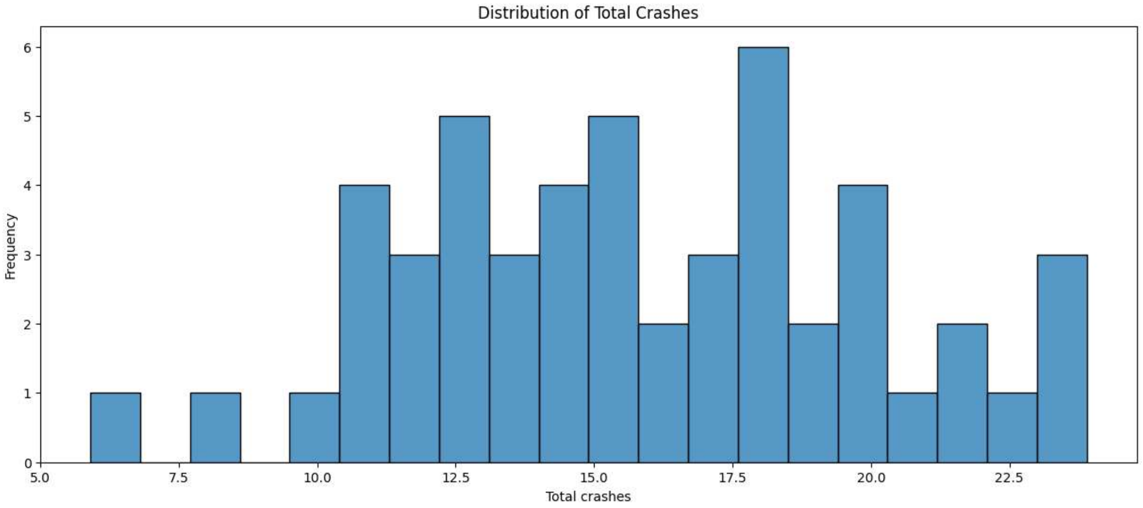
Inference -
The scatterplot compares Total car crashes with the crashes due to speeding,
The plot has a positive correlation and when there are more number of crashes, there are more number of speeding crashes

```
# Line plot
plt.figure(figsize=(15, 6))
sns.lineplot(x='ins_premium', y ='ins_losses', data=df, marker='*', markerfacecolor='limegreen', markersize=20)
plt.xlabel('ins_premium')
plt.ylabel('ins_losses')
plt.title('Premium - loss comparison')
plt.show()
```



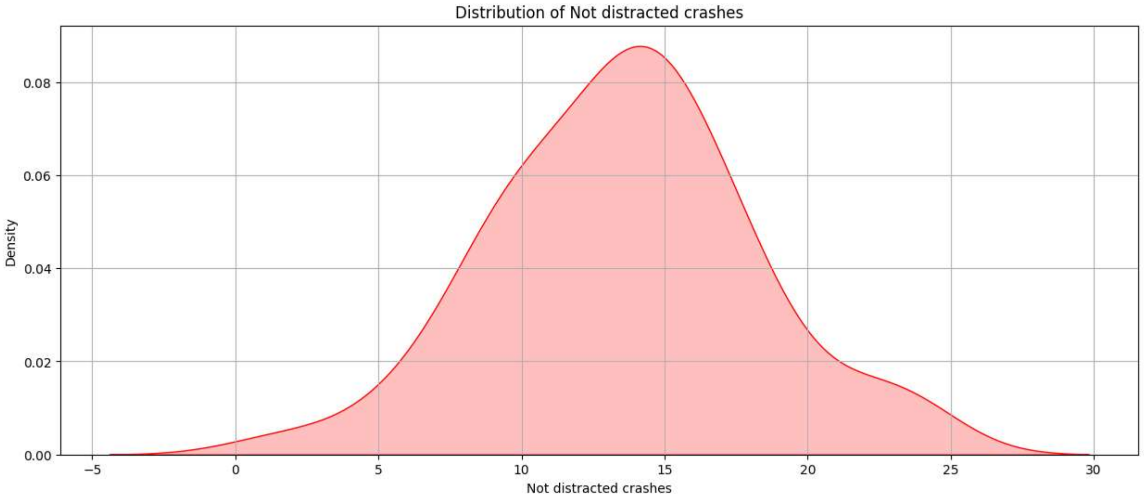
Inference-
  The above lineplot compares the insurance premium and insurance loss over the car crashes,
  From the plot it is clear that the maximum loss occurred for a insurance premium of 1300 and minimum loss is occurred for an insurance premium of 650 (approx.)

```
# Histplot
plt.figure(figsize=(15, 6))
sns.histplot(df["total"], bins=20)
plt.title("Distribution of Total Crashes")
plt.xlabel("Total crashes")
plt.ylabel("Frequency")
plt.show()
```



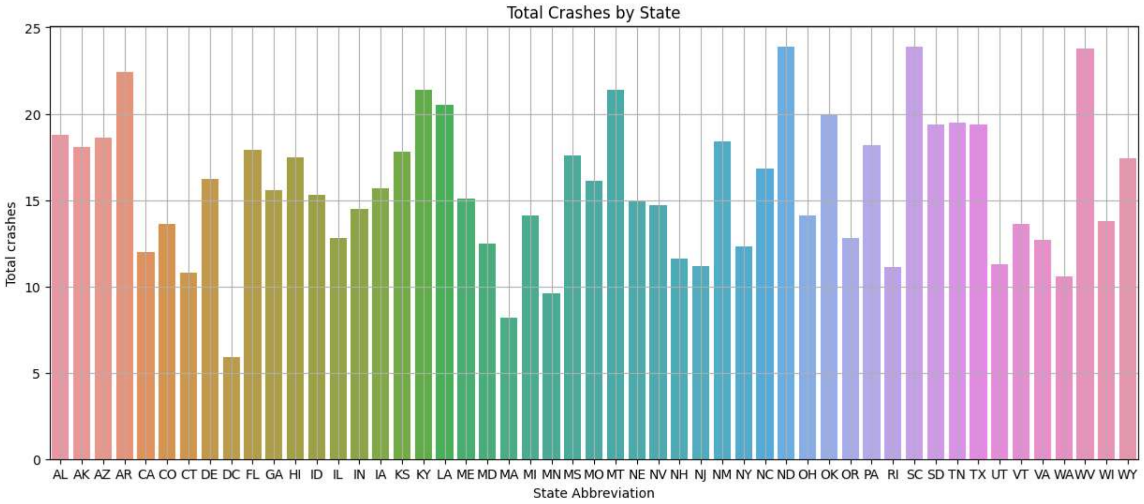
```
Inference -
  The histogram for the distribution of the total number of crashes shows a right-skewed distribution,
  with most states having a lower number of crashes, clustered around 5 to 10 crashes and 17.5 accidents has the maximum frequency
```

```
#Kernel density plot
plt.figure(figsize=(15, 6))
sns.kdeplot(df["not_distracted"], fill = True,color='red',cbar=True)
plt.title("Distribution of Not distracted crashes")
plt.xlabel("Not distracted crashes")
plt.ylabel("Density")
plt.grid()
plt.show()
```



```
Inference-
  The above kernel density plot shows the density levels of notdistracted crashes of the data,
  The maximum density 0.09 is obtained for 13 not distracted crashes and there is high density for 10-20 number of crashes
```

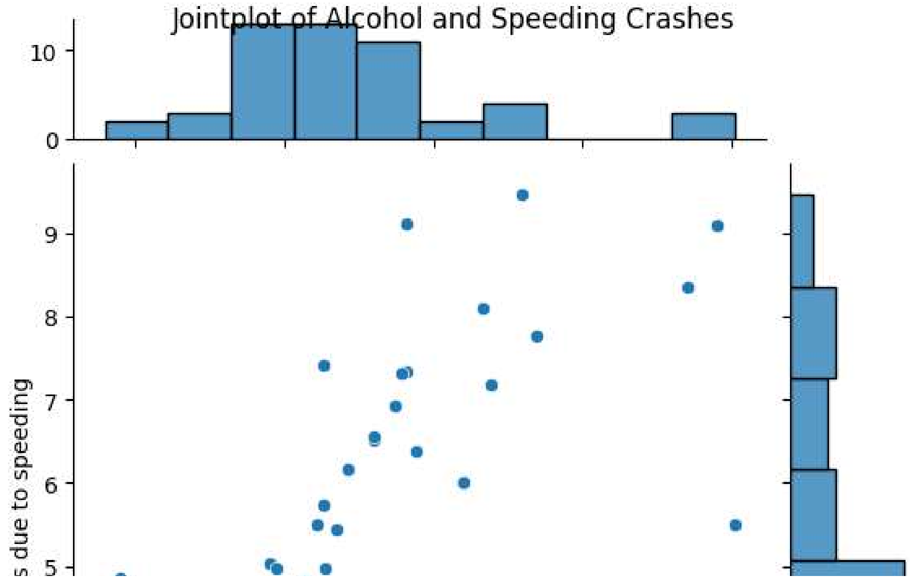
```
#Barplot
plt.figure(figsize=(15, 6))
sns.barplot(x="abbrev", y="total", data=df, orient='v')
plt.title("Total Crashes by State")
plt.xlabel("State Abbreviation")
plt.ylabel("Total crashes")
plt.grid()
plt.show()
```



```
Inference-
 The barplot shows the total number of accidents by state,
 North Dakota, South Carolina & West Virginia have the highest number of accidents followed by Arkansas.
```

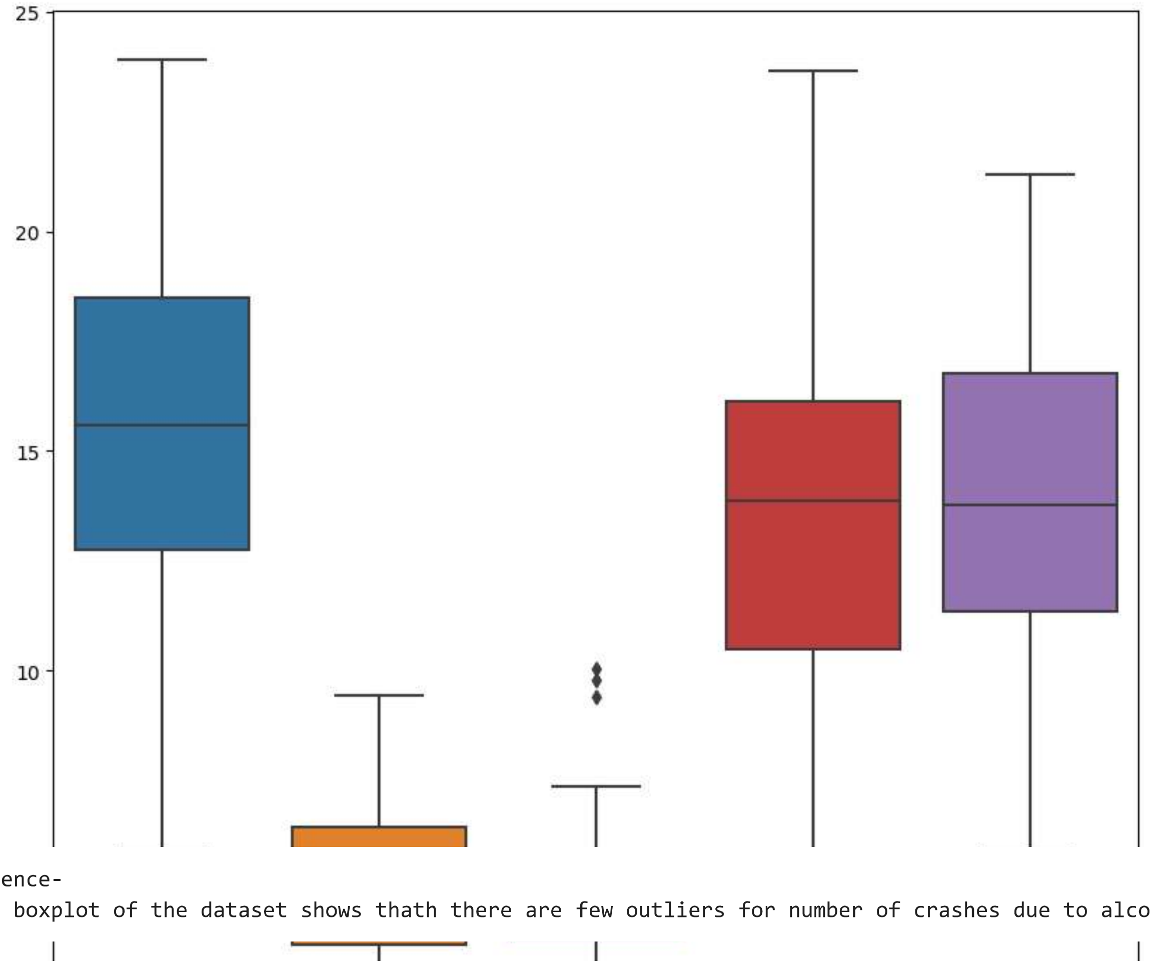
```
#Jointplot
plt.figure(figsize=(15, 15))
sns.jointplot(x="alcohol", y="speeding", data=df, kind="scatter",marginal_ticks=True)
plt.suptitle("Jointplot of Alcohol and Speeding Crashes")
plt.xlabel("Crashes due to alcohol")
plt.ylabel("Crashes due to speeding")
plt.show()
```

```
<Figure size 1500x1500 with 0 Axes>
```



```
Inference-
 The jointplot visually represents the relationship between alcohol and speeding crashes.
 It shows that as alcohol-related accidents increase, speeding-related accidents also tend to increase.
```

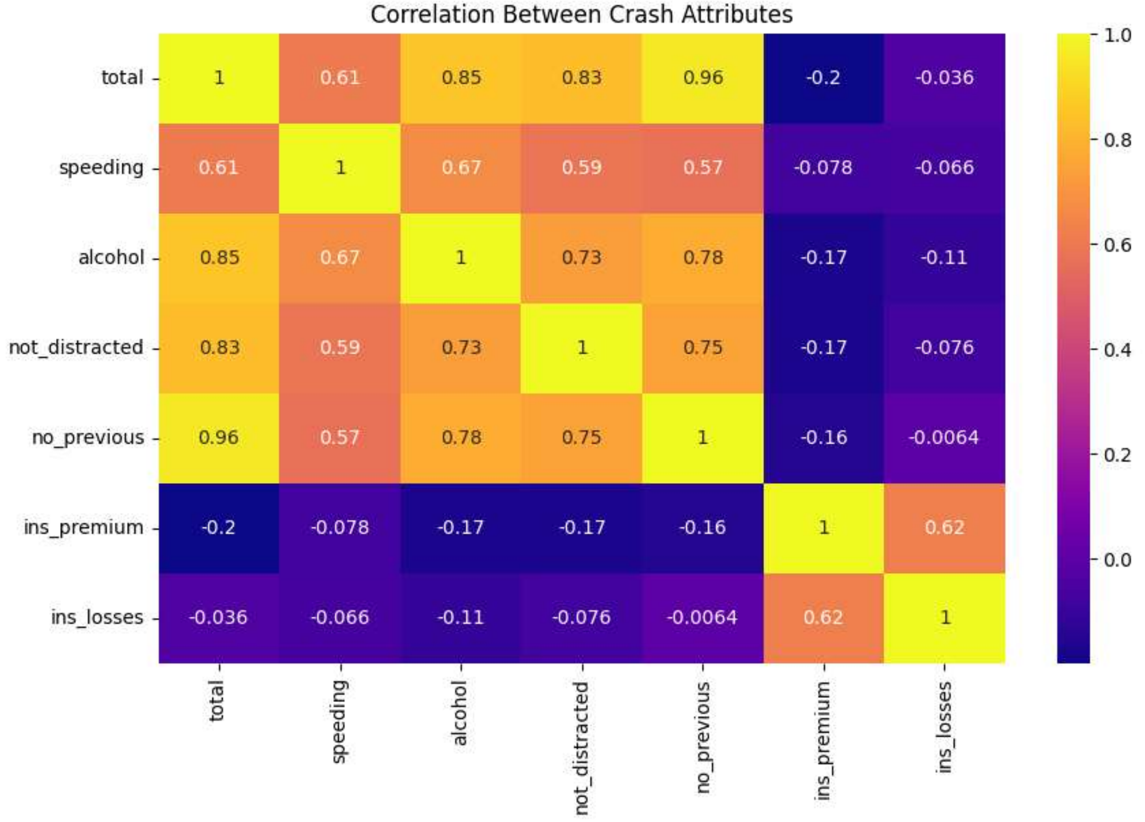
```
#Boxplot
plt.figure(figsize=(10, 10))
sns.boxplot(df.iloc[:,0:5])
plt.show()
```



```
Inference-
  The boxplot of the dataset shows thath there are few outliers for number of crashes due to alcohol and not_ditracted crashes
```

```
#Heatmap
correlation_matrix = df.corr()
plt.figure(figsize=(10, 6))
sns.heatmap(correlation_matrix, annot=True, cmap="plasma")
plt.title("Correlation Between Crash Attributes")
plt.show()
```

```
<ipython-input-15-5cc733a074a5>:2: FutureWarning: The default value of numeric_only in DataF
  correlation_matrix = df.corr()
```



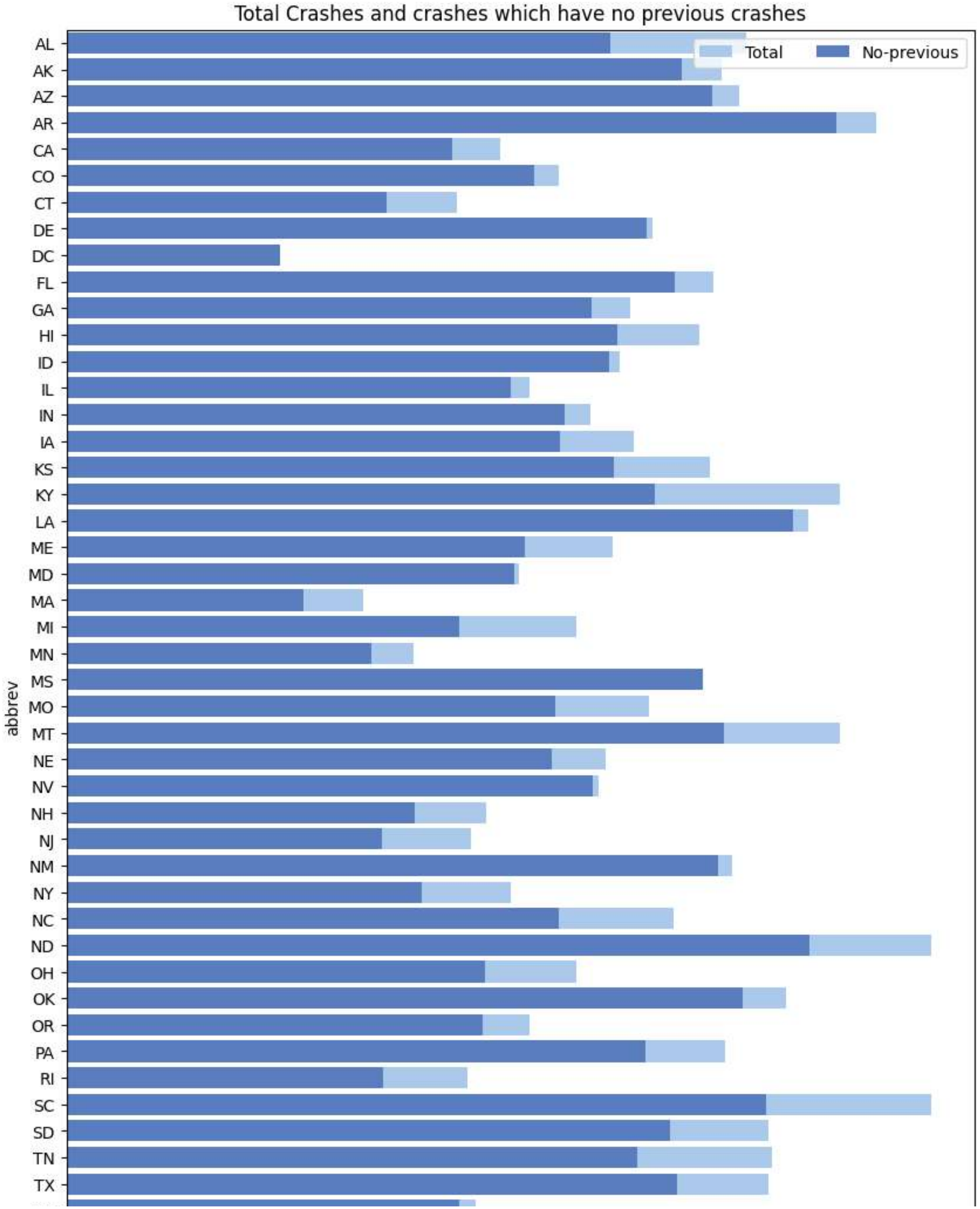
```
Inference-
 The heatmap illustrating the correlation between crashes attributes shows that there is a positive correlation between some attributes
```

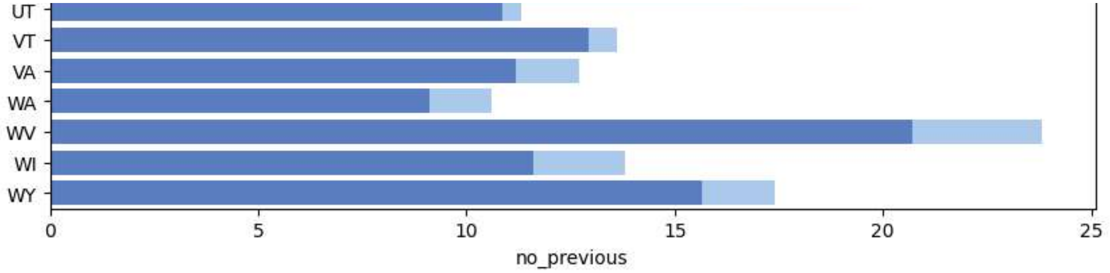
```
# Stacked barchart
f, ax = plt.subplots(figsize=(10, 15))

# Plot the total crashes
sns.set_color_codes("pastel")
sns.barplot(x="total", y="abbrev", data=df ,label="Total", color="b")

# Plot the crashes who are not involved in previous crashes
sns.set_color_codes("muted")
sns.barplot(x="no_previous", y="abbrev", data=df, label="No-previous", color="b")
plt.title("Total Crashes and crashes which have no previous crashes")
ax.legend(ncol=2, loc="upper right", frameon=True)
```

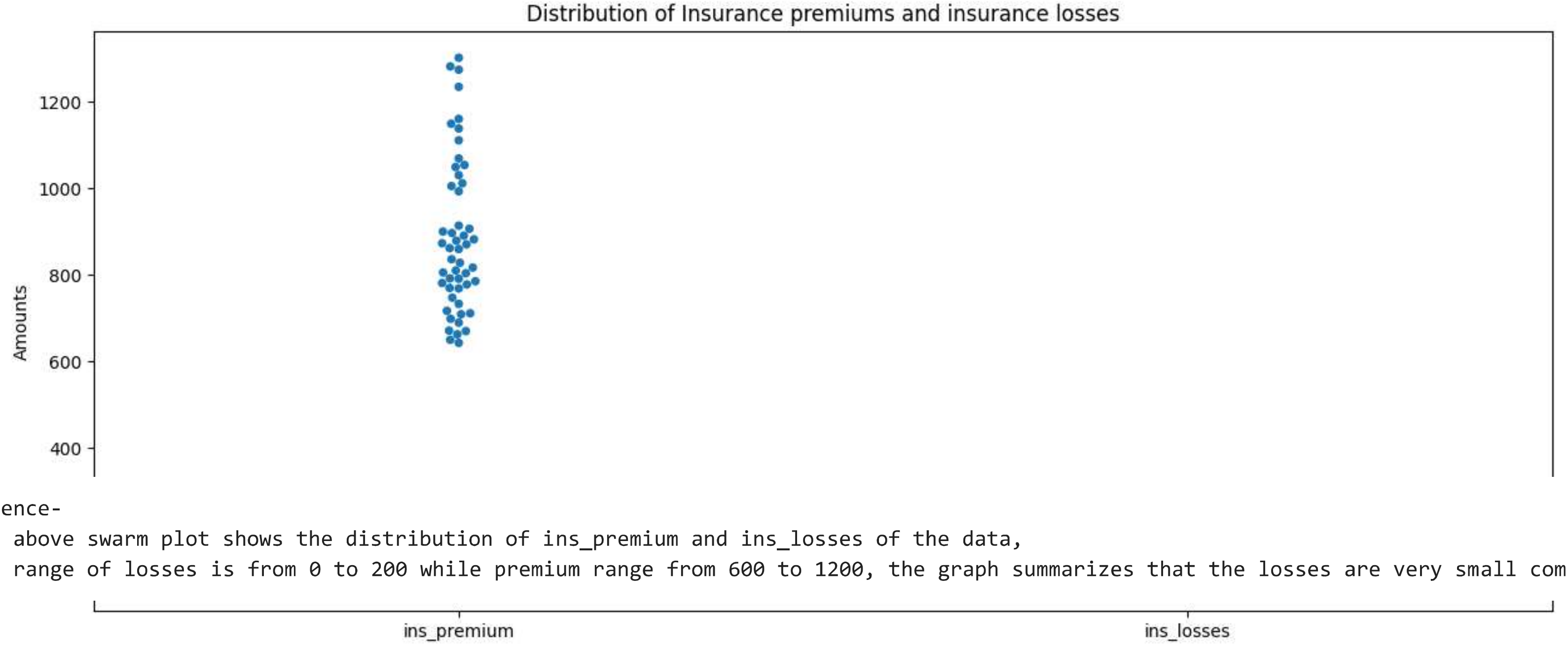
```
<matplotlib.legend.Legend at 0x791a8df5c040>
```


Total Crashes and crashes which have no previous crashes
Total
No-previous
abbrev
AL
AK
AZ
AR
CA
CO
CT
DE
DC
FL
GA
HI
ID
IL
IN
IA
KS
KY
LA
ME
MD
MA
MI
MN
MS
MO
MT
NE
NV
NH
NJ
NM
NY
NC
ND
OH
OK
OR
PA
RI
SC
SD
TN
TX



```
Inference-
  The plot contains the bars of both total number of crashes and crashes with no previous crashes,
  It is clear that the total number of crashes is directly proportional to No previous crashes and
  More crashes that occur are having no previous crash history
```

```
# Swarm plot
plt.figure(figsize=(15, 6))
sns.swarmplot(df.iloc[:,5:7])
plt.title("Distribution of Insurance premiums and insurance losses")
plt.ylabel("Amounts")
plt.show()
```



```
Inference-
  The above swarm plot shows the distribution of ins_premium and ins_losses of the data,
  the range of losses is from 0 to 200 while premium range from 600 to 1200, the graph summarizes that the losses are very small compared to the premium
```