In [ ]:
```python
import seaborn as sns
import matplotlib.pyplot as plt
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
```

In [ ]:
```python
car_crashes = sns.load_dataset("car_crashes")
car_crashes.dropna(inplace=True)
```

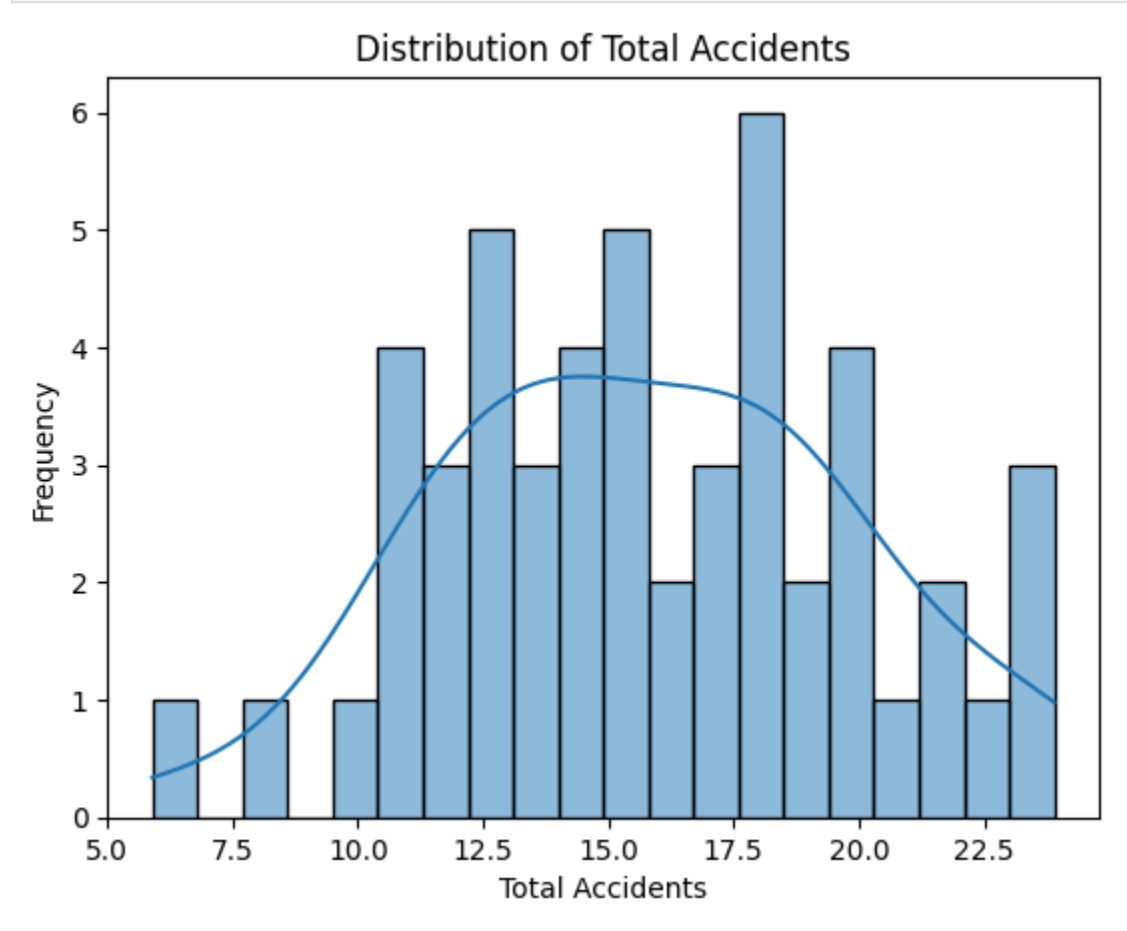
In [ ]:
```python
X = car_crashes.drop(columns=['total'])  # Independent variables
y = car_crashes['total']
```

In [ ]:
```python
le = LabelEncoder()
X['abbrev'] = le.fit_transform(X['abbrev'])
```

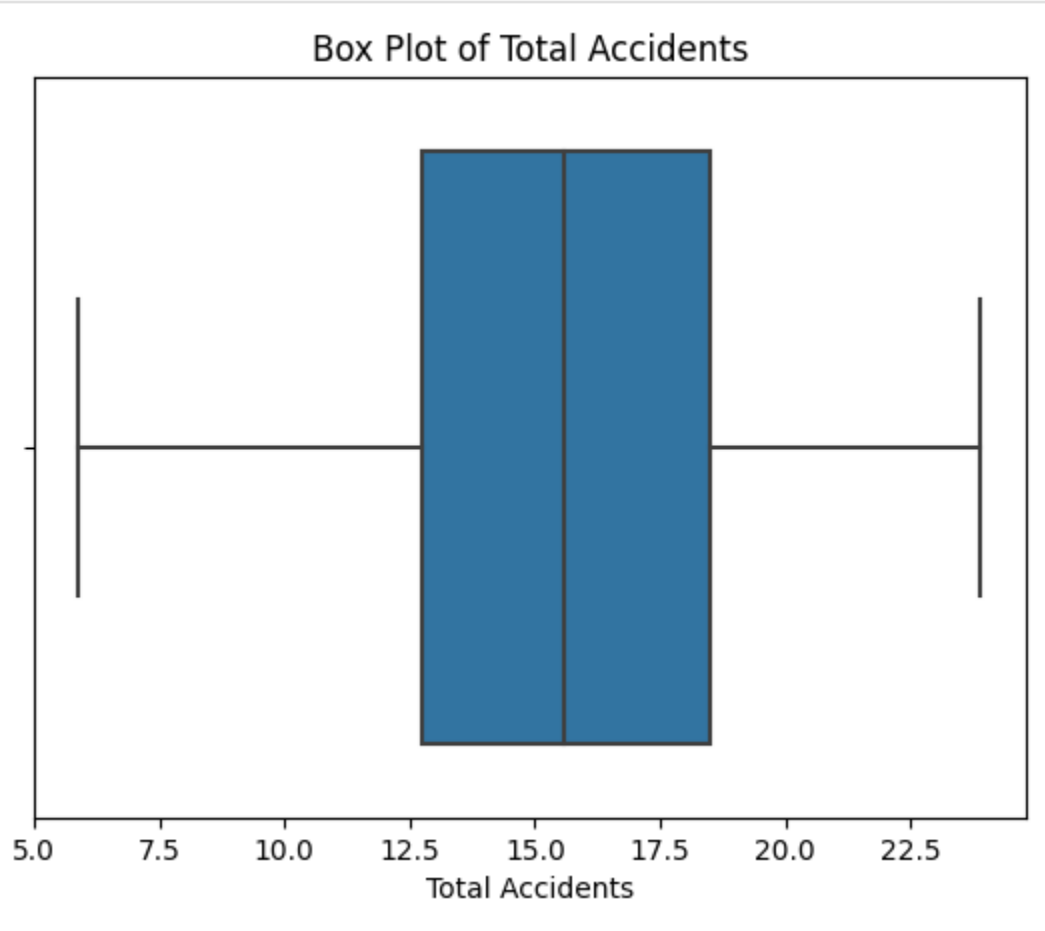
In [ ]:
```python
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

In [ ]:
```python
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
```

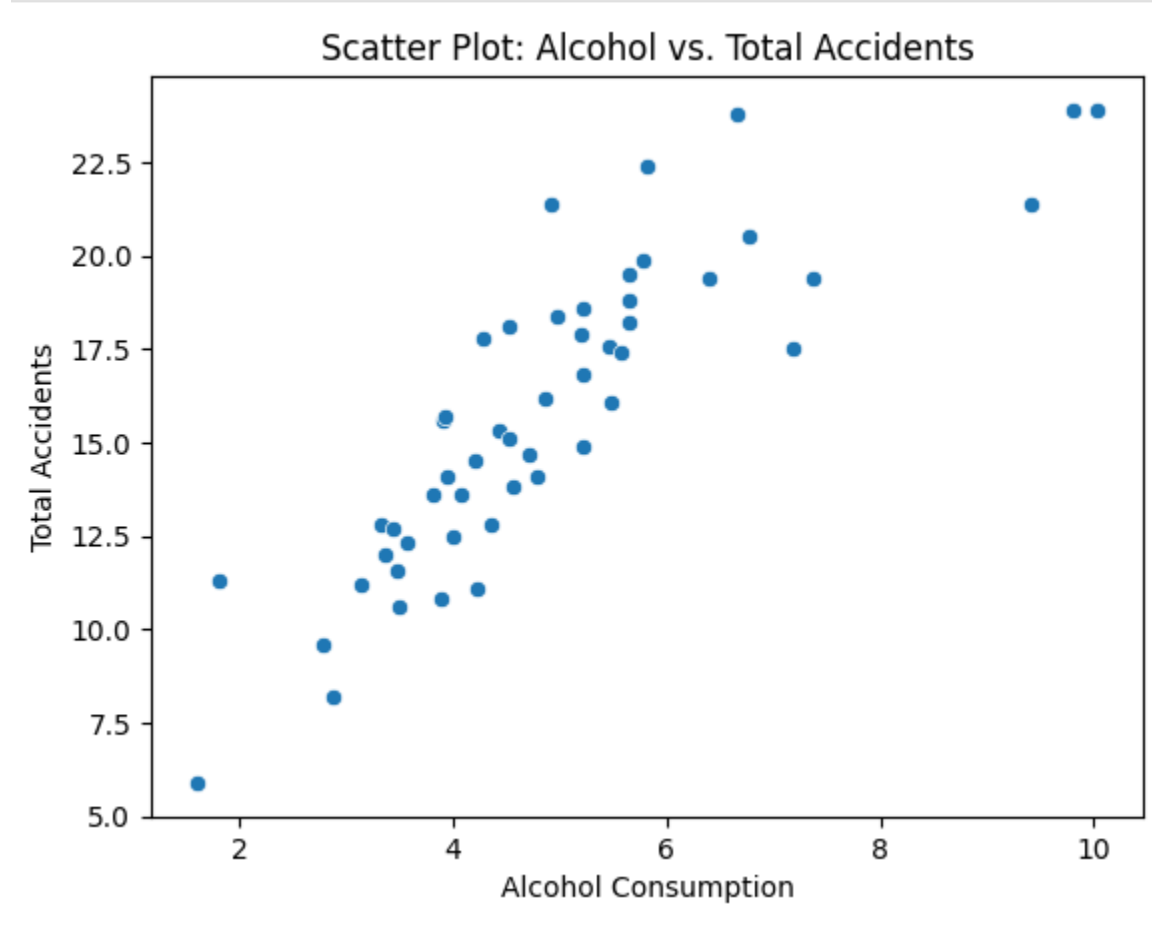
In [ ]:
```python
sns.histplot(car_crashes["total"], bins=20, kde=True)
plt.title("Distribution of Total Accidents")
plt.xlabel("Total Accidents")
plt.ylabel("Frequency")
plt.show()
```



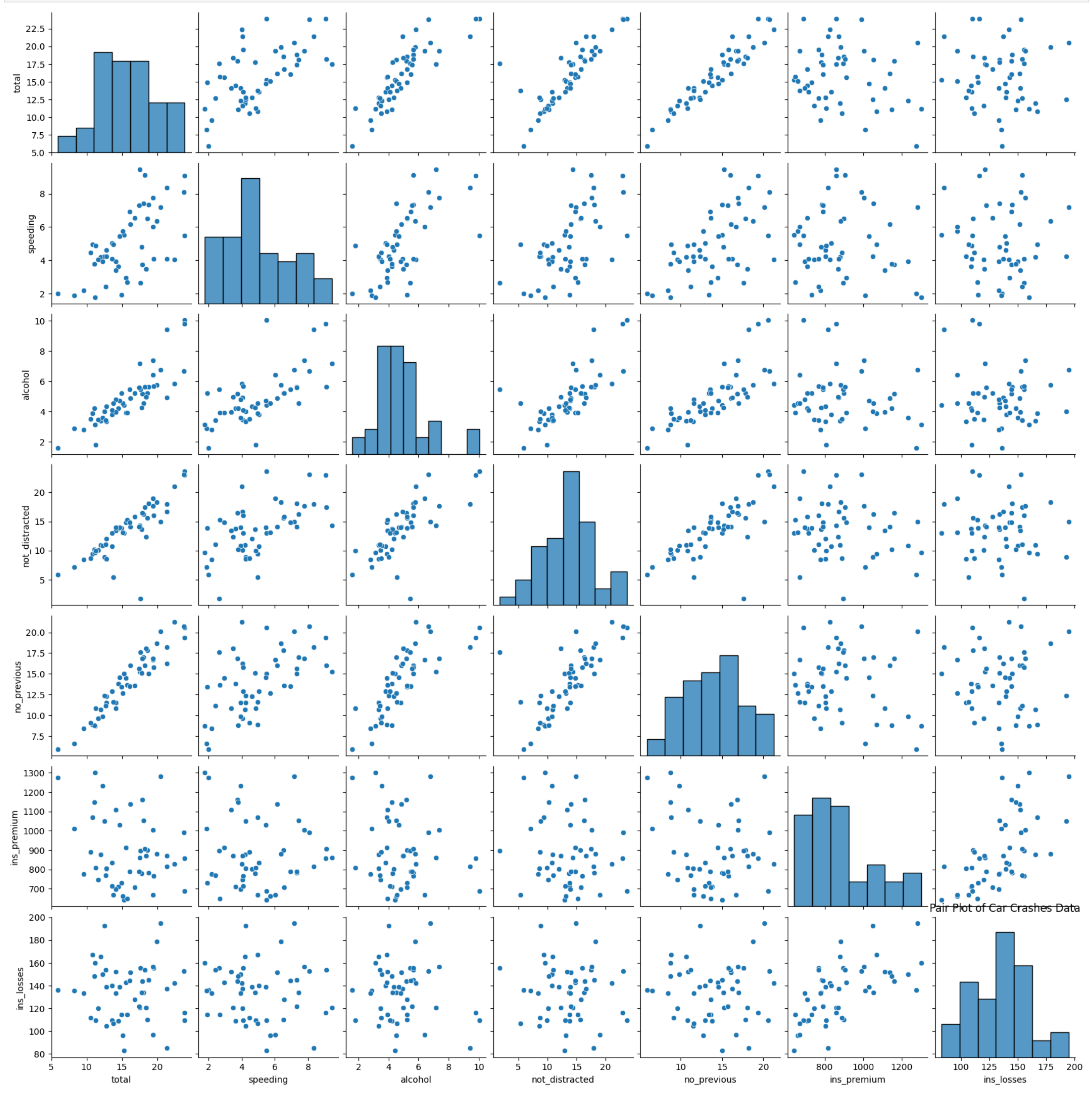
In [ ]:
```python
sns.boxplot(x="total", data=car_crashes)
plt.title("Box Plot of Total Accidents")
plt.xlabel("Total Accidents")
plt.show()
```



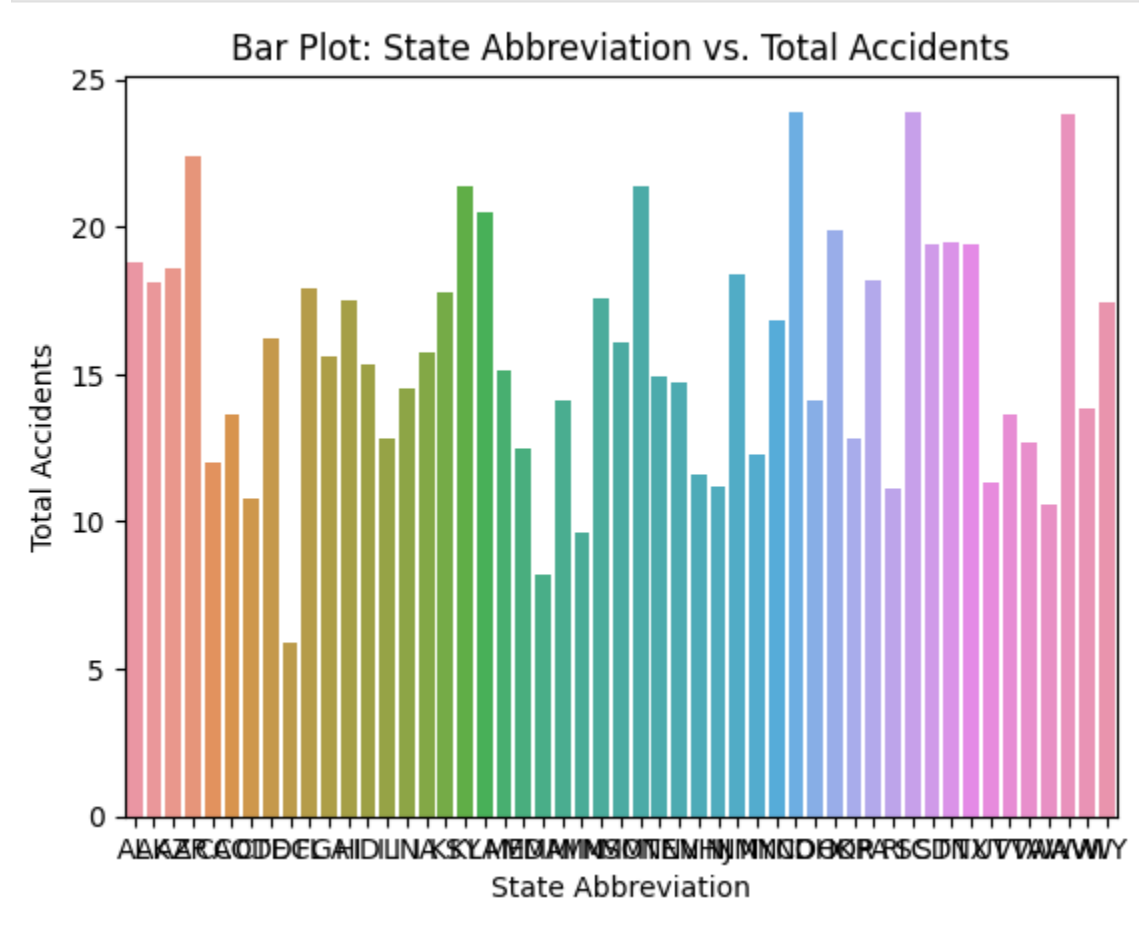
In [ ]:
```python
sns.scatterplot(x="alcohol", y="total", data=car_crashes)
plt.title("Scatter Plot: Alcohol vs. Total Accidents")
plt.xlabel("Alcohol Consumption")
plt.ylabel("Total Accidents")
plt.show()
```



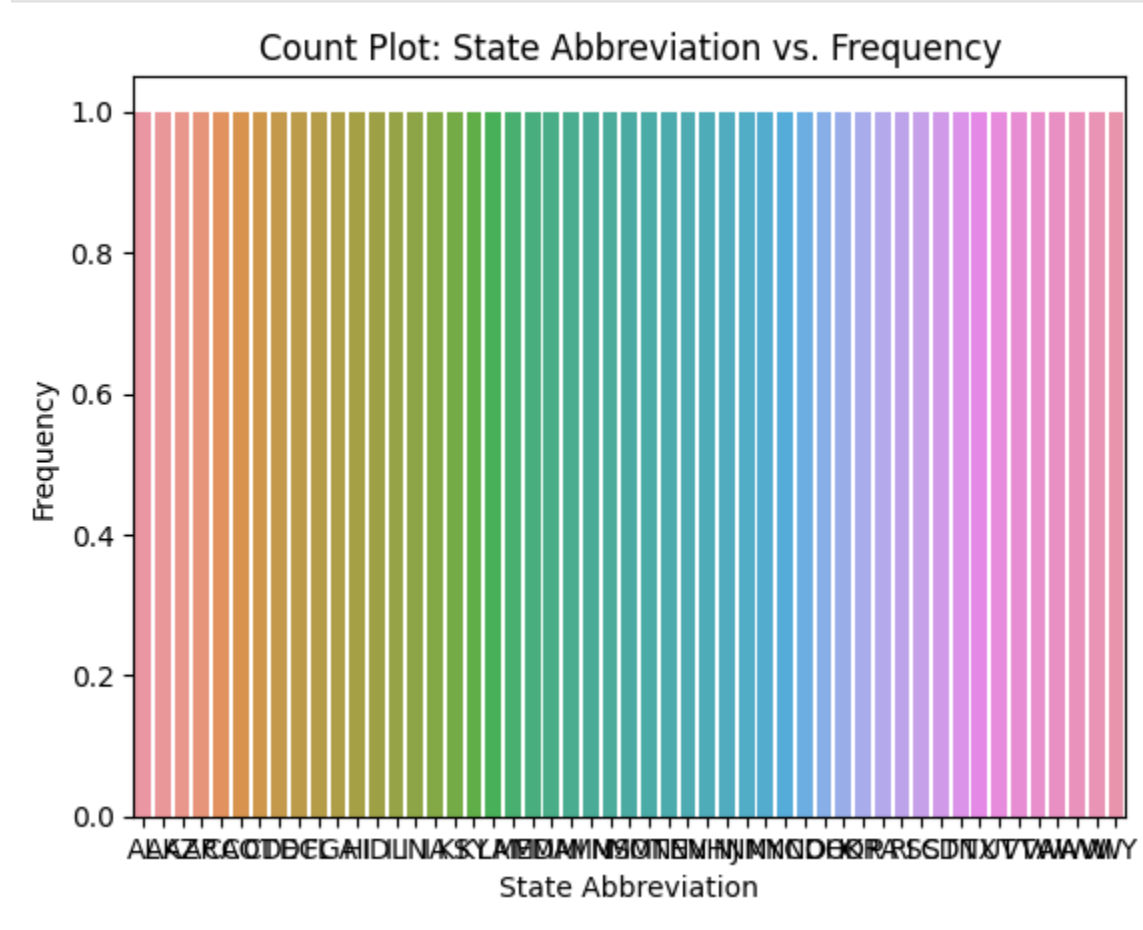
In [ ]:
```python
sns.pairplot(car_crashes)
plt.title("Pair Plot of Car Crashes Data")
plt.show()
```



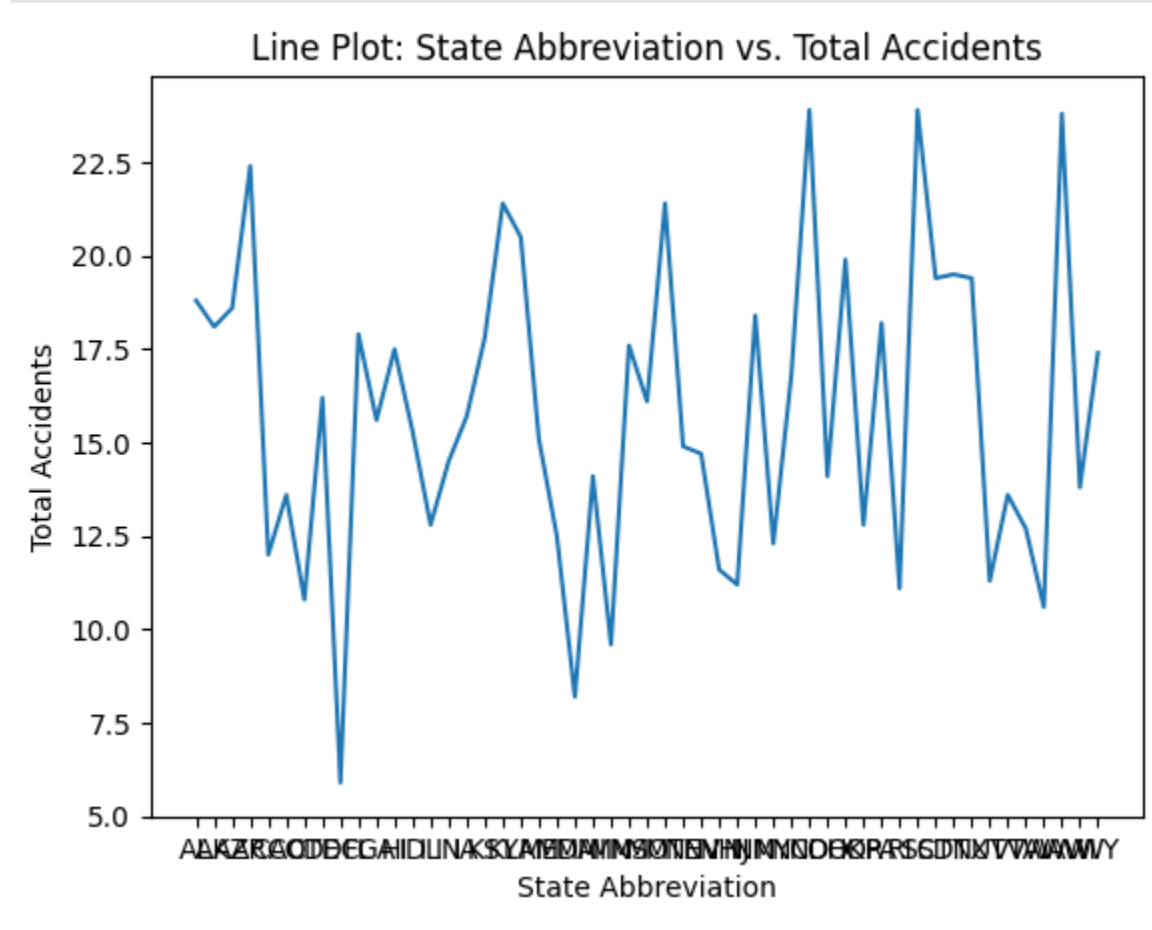
In [ ]:
```python
sns.barplot(x="abbrev", y="total", data=car_crashes)
plt.title("Bar Plot: State Abbreviation vs. Total Accidents")
plt.xlabel("State Abbreviation")
plt.ylabel("Total Accidents")
plt.show()
```



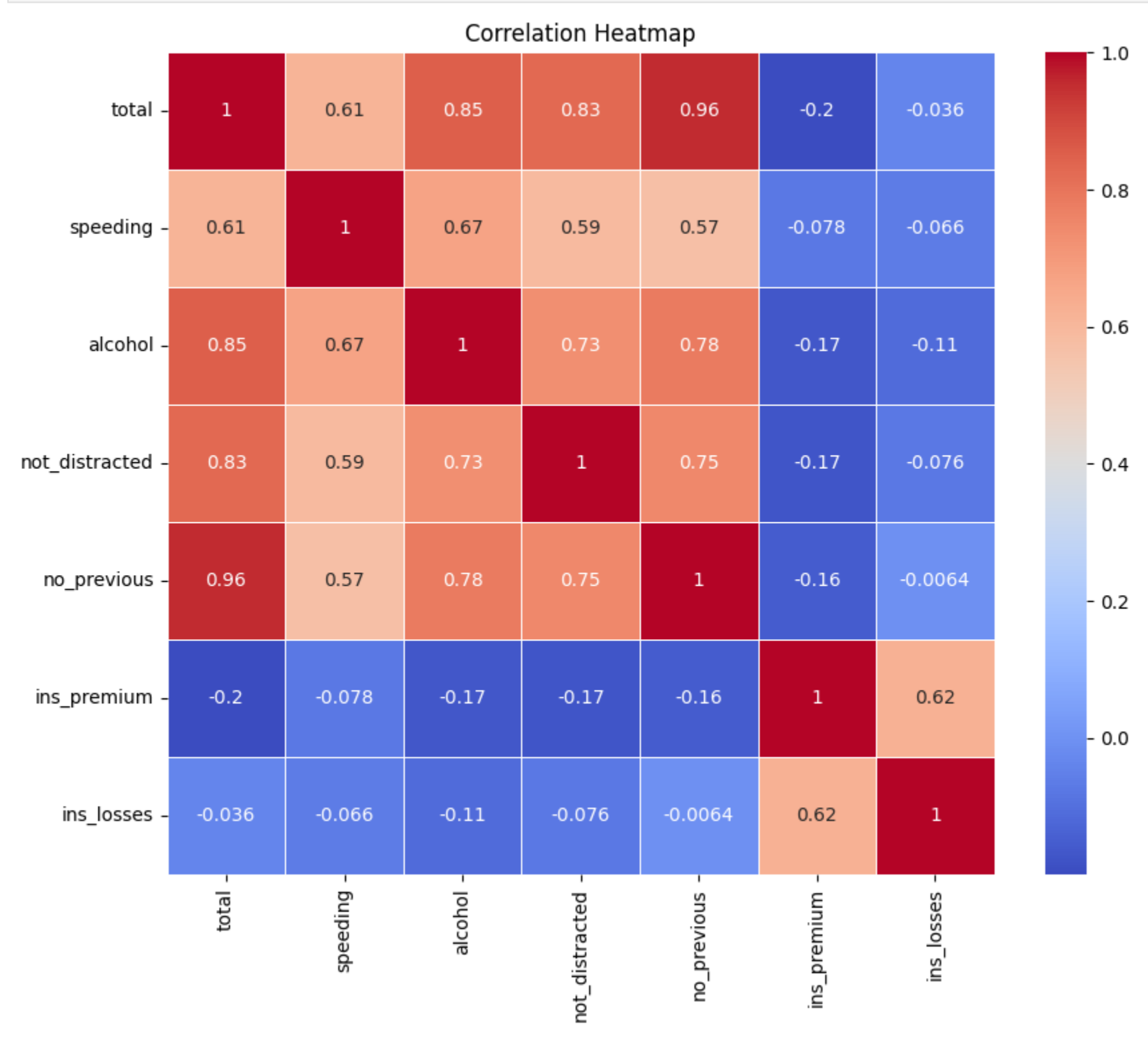
In [ ]:
```python
sns.countplot(x="abbrev", data=car_crashes)
plt.title("Count Plot: State Abbreviation vs. Frequency")
plt.xlabel("State Abbreviation")
plt.ylabel("Frequency")
plt.show()
```



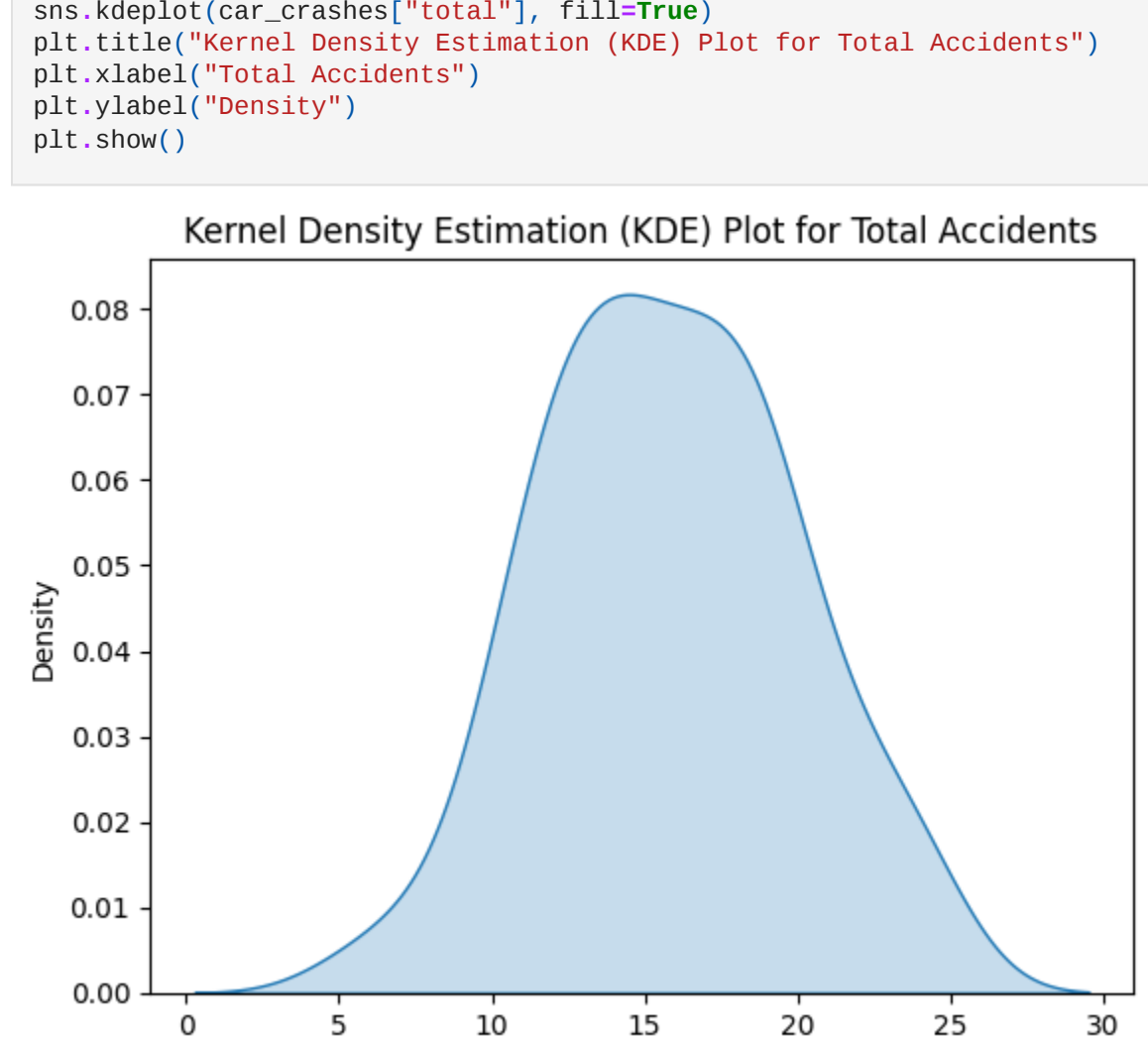
In [ ]:
```python
sns.lineplot(x="abbrev", y="total", data=car_crashes)
plt.title("Line Plot: State Abbreviation vs. Total Accidents")
plt.xlabel("State Abbreviation")
plt.ylabel("Total Accidents")
plt.show()
```



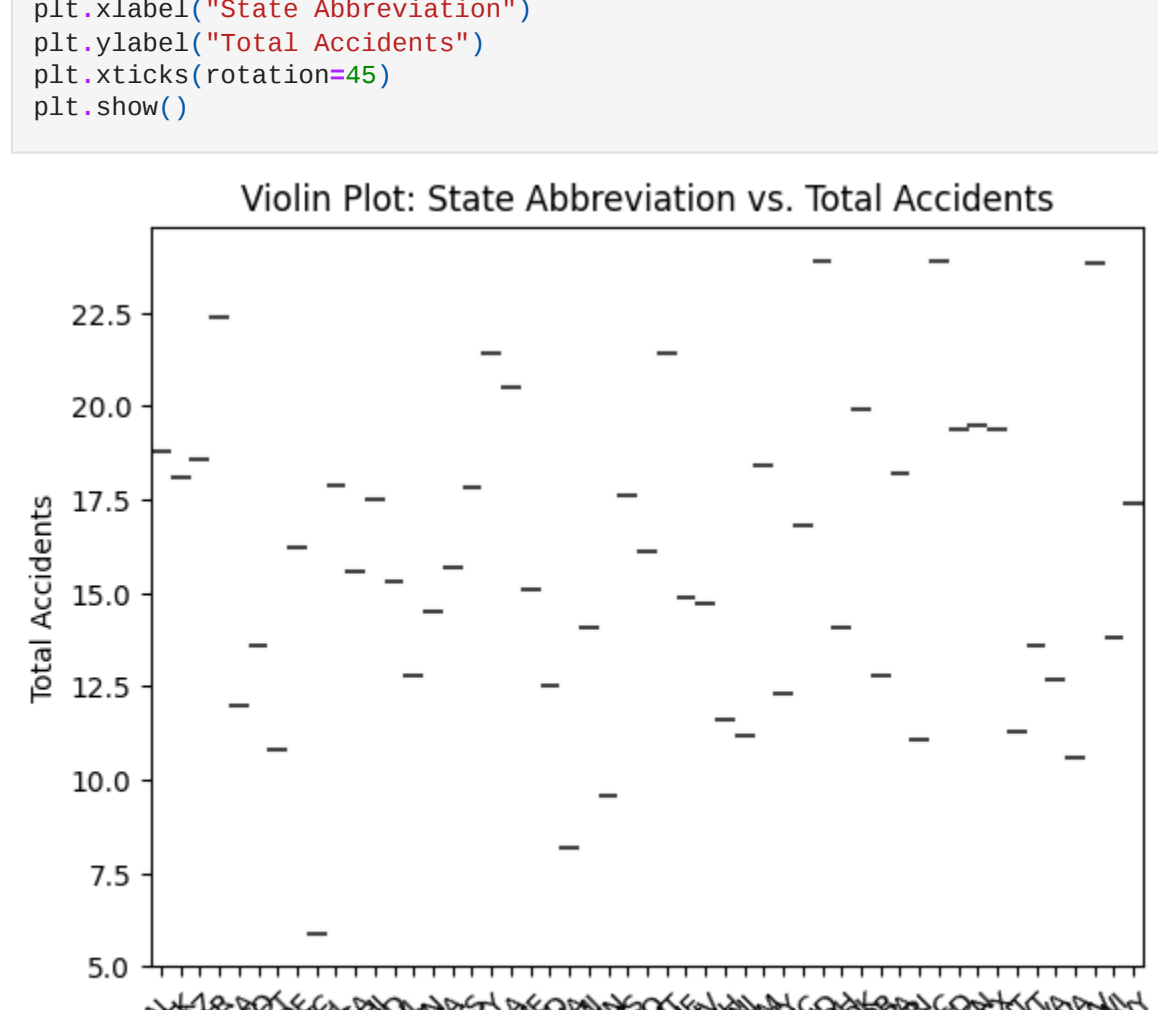
In [ ]:
```python
corr_matrix = car_crashes.corr(numeric_only=True)
plt.figure(figsize=(10, 8))
sns.heatmap(corr_matrix, annot=True, cmap='coolwarm', linewidths=.5)
plt.title("Correlation Heatmap")
plt.show()
```



In [ ]:
```python
sns.kdeplot(car_crashes["total"], fill=True)
plt.title("Kernel Density Estimation (KDE) Plot for Total Accidents")
plt.xlabel("Total Accidents")
plt.ylabel("Density")
plt.show()
```



In [ ]:
```python
sns.violinplot(x="abbrev", y="total", data=car_crashes)
plt.title("Violin Plot: State Abbreviation vs. Total Accidents")
plt.xlabel("State Abbreviation")
plt.ylabel("Total Accidents")
plt.xticks(rotation=45)
plt.show()
```



In [ ]:
```python
model = LinearRegression()
```

In [ ]:
```python
model.fit(X_train_scaled, y_train)
```

Out[ ]:
```
▾ LinearRegression
LinearRegression()
```

In [ ]:
```python
y_pred = model.predict(X_test_scaled)
```

In [ ]:
```python
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)
```

In [ ]:
```python
print("Mean Squared Error:", mse)
print("R-squared:", r2)
```

```
Mean Squared Error: 2.1306910265217325
R-squared: 0.8821908178536237
```