```
# Import necessary libraries
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

```
from google.colab import files
uploaded = files.upload()
```

Choose Files No file chosen Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

Saving uk_accident.csv to uk_accident.csv

```
#Load the dataset
df = pd.read_csv('/content/uk_accident.csv')
```

```
df.head()
```

	accident_index	speed_limit	light_conditions	weather_conditions	road_surface_condi
0	2.02E+12	30	darkness	other	wet o
1	2.02E+12	30	darkness	fine	
2	2.02E+12	40	daylight	fine	
3	2.02E+12	40	daylight	fine	
4	2.02E+12	30	daylight	fine	

```
# Display basic information about the dataset
print("Dataset Overview:")
print(df.info())
print("\nFirst 5 rows of the dataset:")
print(df.head())
```

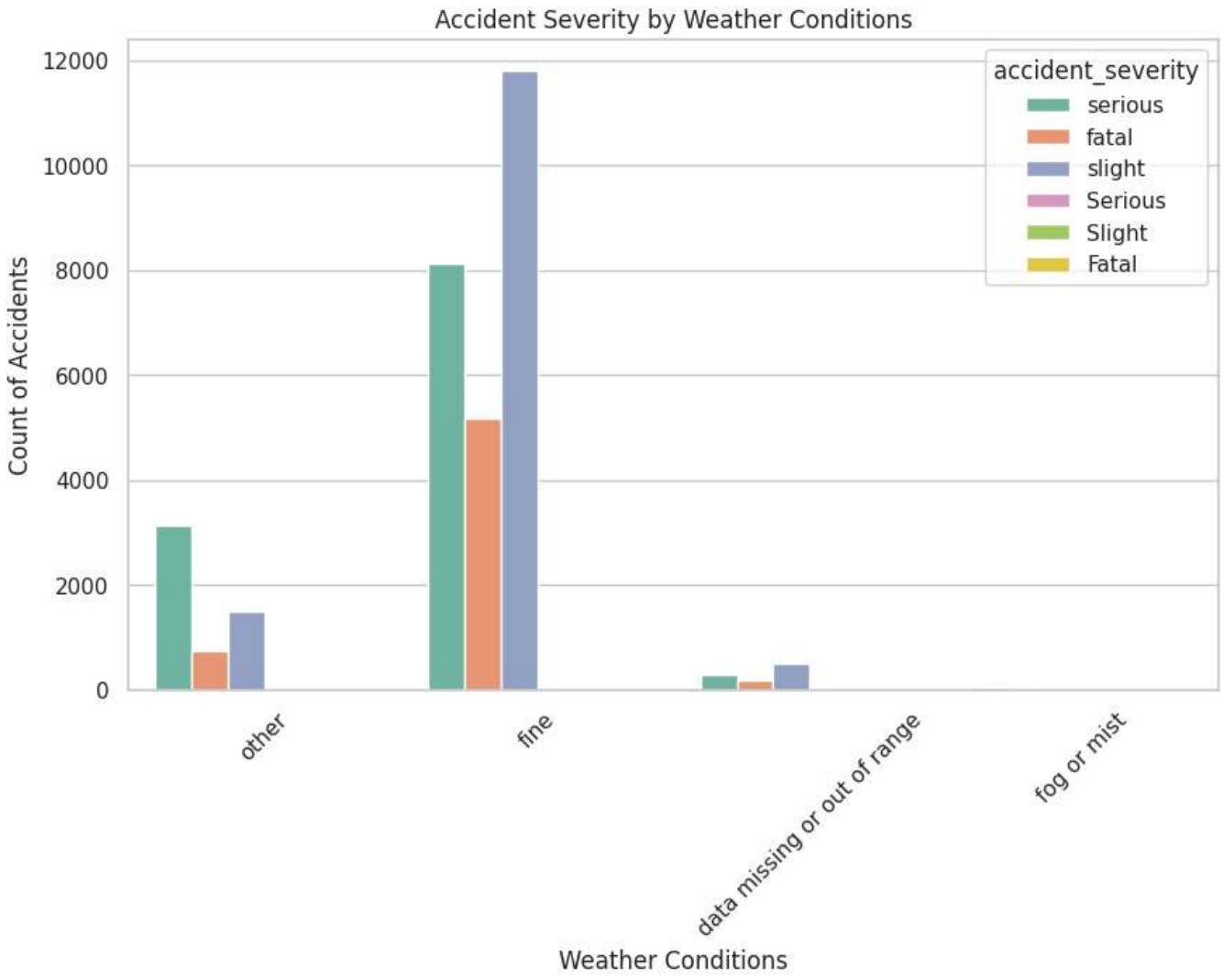
```
 9   hit_object_off_carriageway   31647 non-null  object
 10  first_point_of_impact        31647 non-null  object
 11  sex_of_driver                31647 non-null  object
 12  age_of_oldest_driver         31647 non-null  float64
 13  accident_severity            31647 non-null  object
dtypes: float64(1), int64(1), object(12)
memory usage: 3.4+ MB
None

First 5 rows of the dataset:
  accident_index  speed_limit light_conditions weather_conditions  \
0       2.02E+12           30         darkness              other
1       2.02E+12           30         darkness               fine
2       2.02E+12           40         daylight               fine
3       2.02E+12           40         daylight               fine
4       2.02E+12           30         daylight               fine

  road_surface_conditions      vehicle_type  \
0             wet or damp  at least one van
1                     dry         only cars
2                     dry         only cars
3                     dry         only cars
4                     dry         only cars

                       junction_location    skidding_and_overturning  \
0      at or within 20 metres of junction  no skidding or overturning
1      at or within 20 metres of junction  no skidding or overturning
2      at or within 20 metres of junction  no skidding or overturning
3  not at or within 20 metres of junction  no skidding or overturning
4  not at or within 20 metres of junction  no skidding or overturning

               vehicle_leaving_carriageway  \
0                  none leaving carriageway
1  at least one vehicle leaving carriageway
2                  none leaving carriageway
3                  none leaving carriageway
4                  none leaving carriageway

          hit_object_off_carriageway  \
0                  none hit an object
1  at least one vehicle hit an object
```

```
3              47.234038           serious
4              39.000000            slight
```

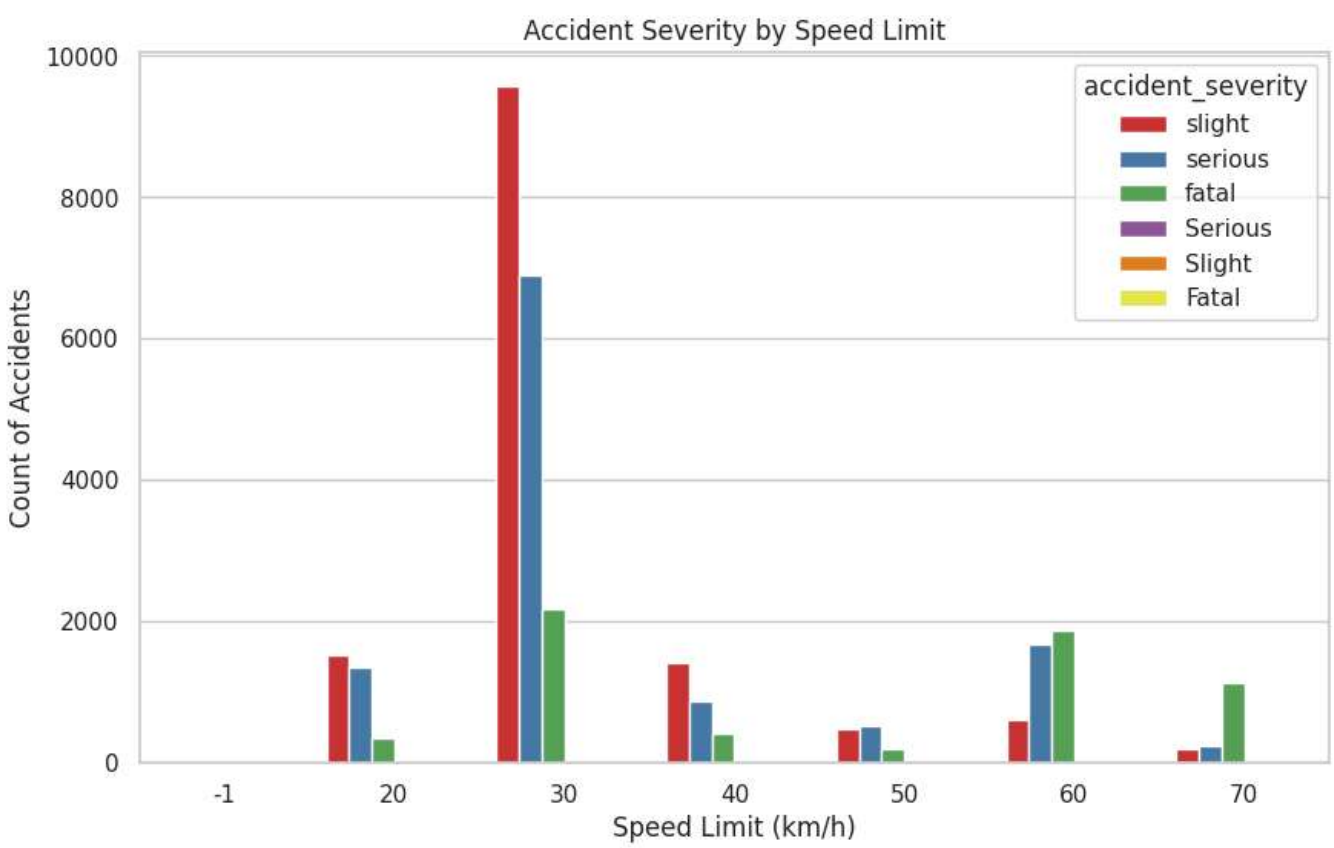
```
# Check for missing values
print("\nMissing Values in Each Column:")
print(df.isnull().sum())
```

```
Missing Values in Each Column:
accident_index                0
speed_limit                   0
light_conditions              0
weather_conditions            0
road_surface_conditions       0
vehicle_type                  0
junction_location             0
skidding_and_overturning      0
vehicle_leaving_carriageway   0
hit_object_off_carriageway    0
first_point_of_impact         0
sex_of_driver                 0
age_of_oldest_driver          0
accident_severity             0
dtype: int64
```

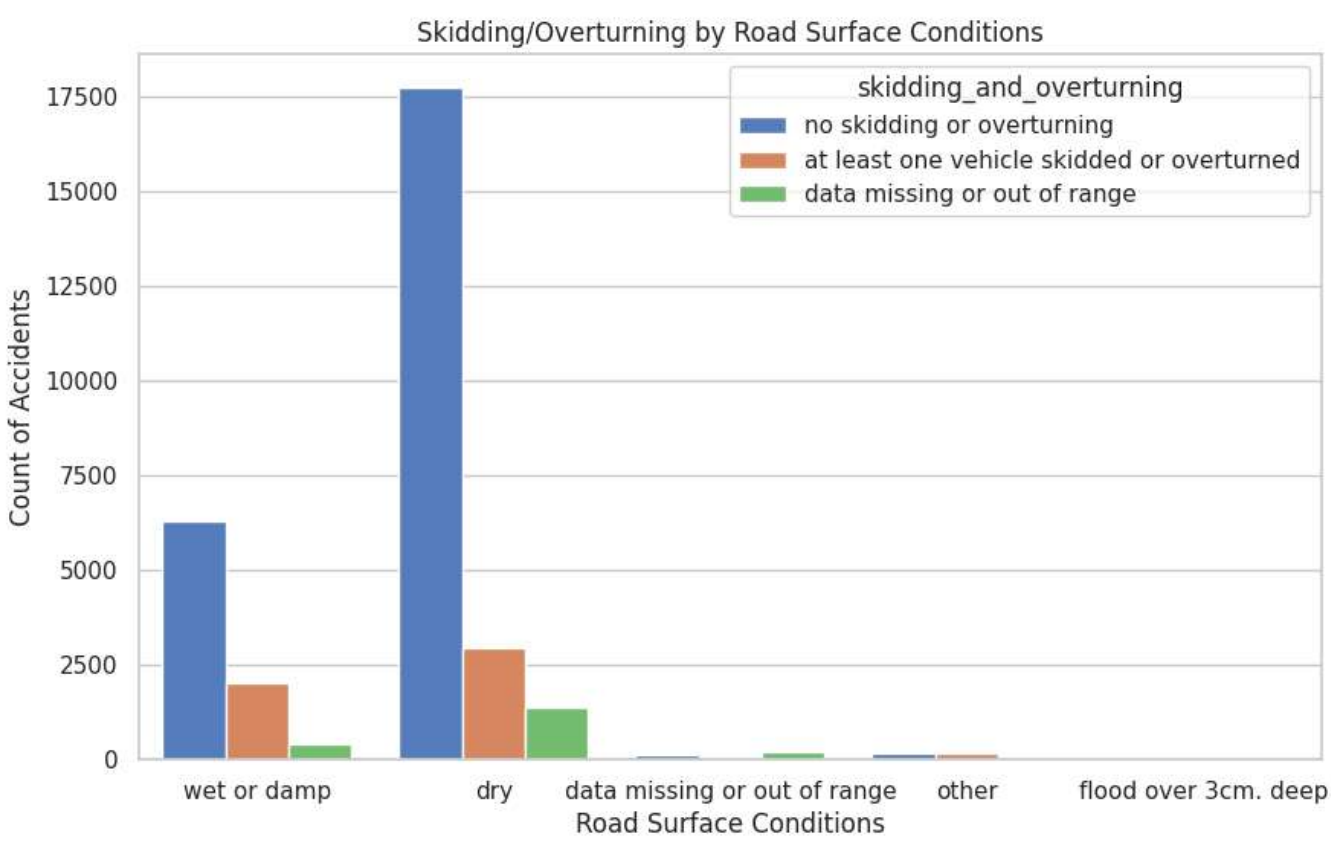
```
# Set up a seaborn style for plots
sns.set(style="whitegrid")

#  Visualize the relationship between weather conditions and accident severity
plt.figure(figsize=(10, 6))
sns.countplot(data=df, x='weather_conditions', hue='accident_severity', palette='Set2')
plt.title('Accident Severity by Weather Conditions')
plt.xlabel('Weather Conditions')
plt.ylabel('Count of Accidents')
plt.xticks(rotation=45)
plt.show()
```



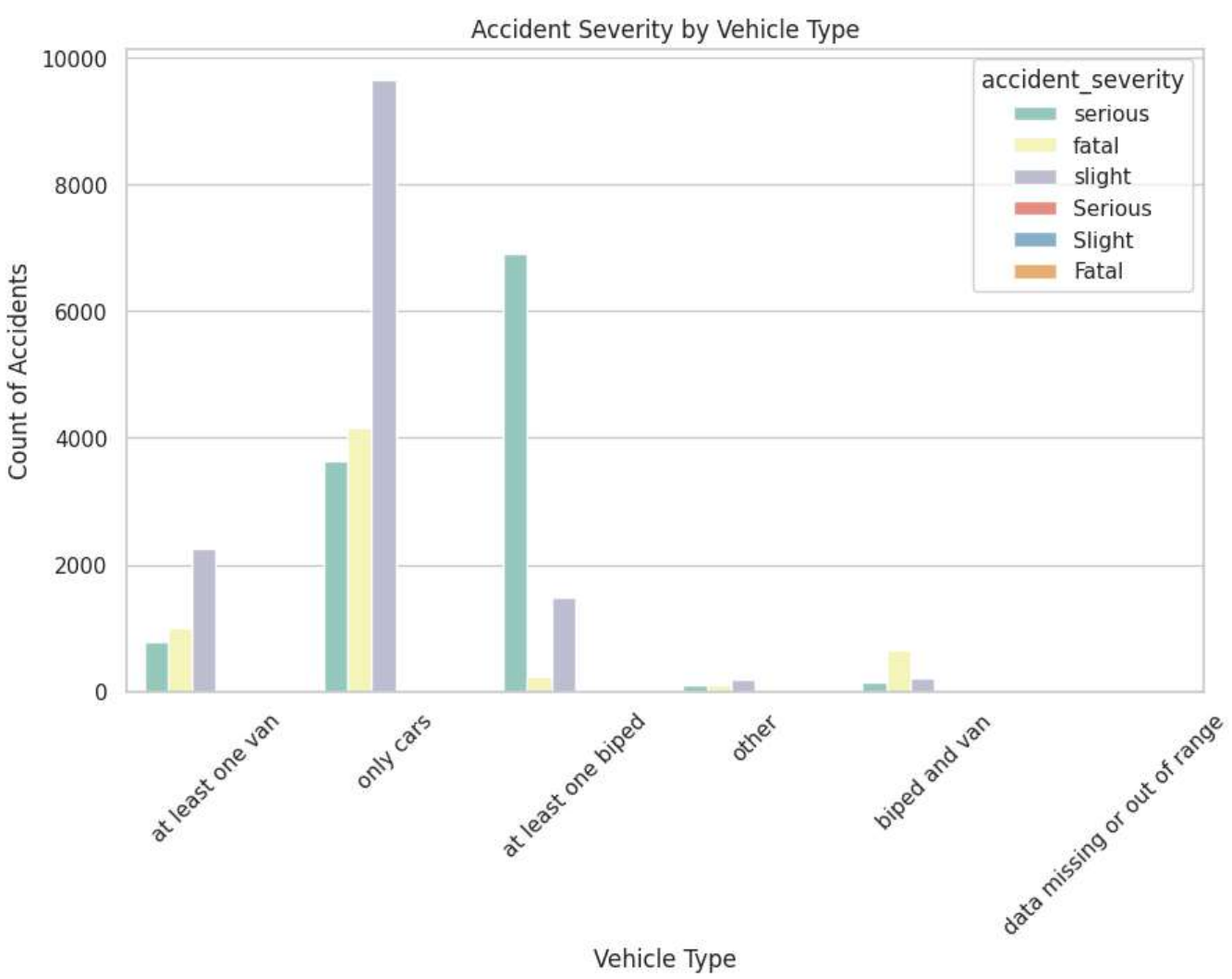

```
#  Visualize the impact of speed limits on accident severity
plt.figure(figsize=(10, 6))
sns.countplot(data=df, x='speed_limit', hue='accident_severity', palette='Set1')
plt.title('Accident Severity by Speed Limit')
plt.xlabel('Speed Limit (km/h)')
plt.ylabel('Count of Accidents')
plt.show()
```




```
#  Visualize the correlation between road surface conditions and skidding/overturning
plt.figure(figsize=(10, 6))
sns.countplot(data=df, x='road_surface_conditions', hue='skidding_and_overturning', palette=
plt.title('Skidding/Overturning by Road Surface Conditions')
plt.xlabel('Road Surface Conditions')
plt.ylabel('Count of Accidents')
plt.show()
```

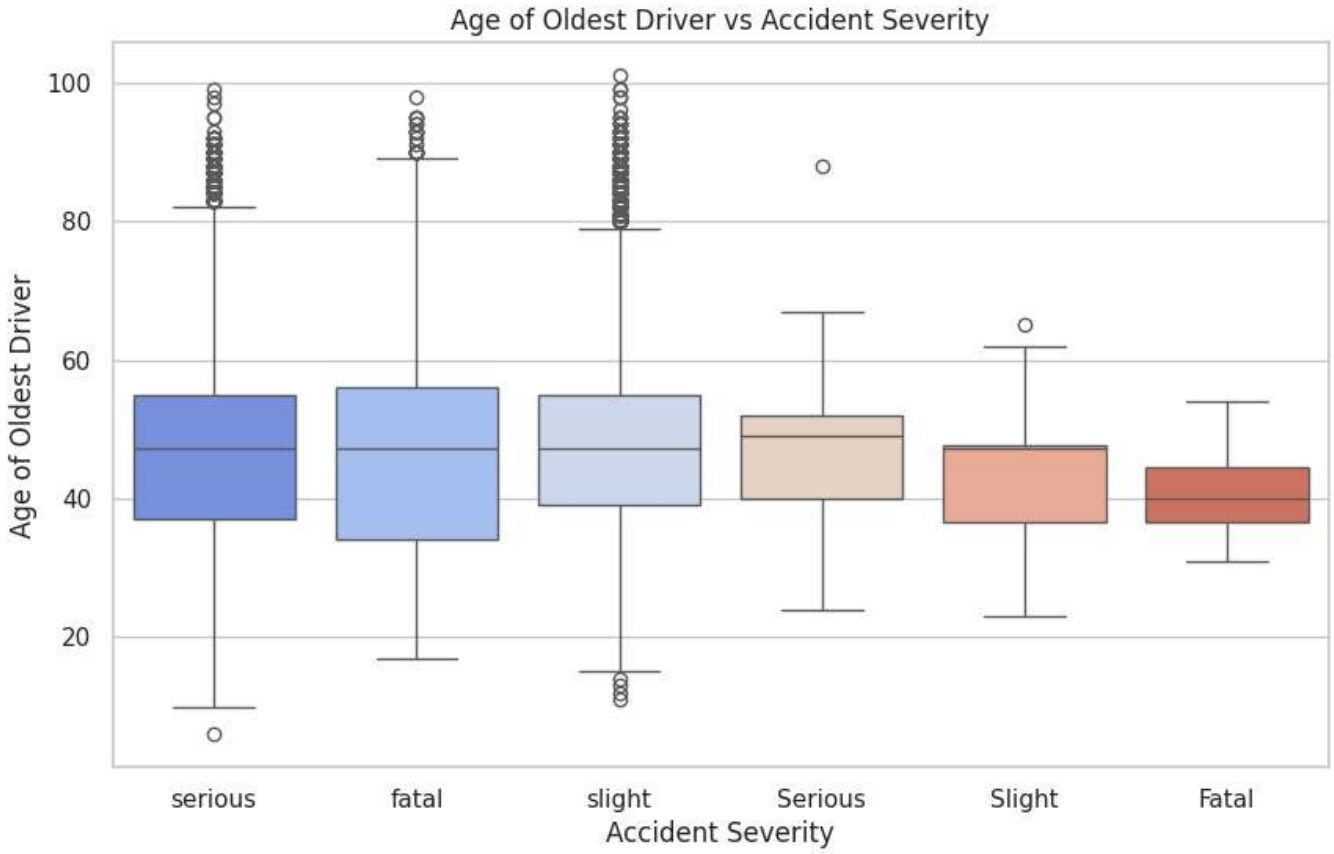



```
#  Visualize accidents by vehicle type and severity
plt.figure(figsize=(10, 6))
sns.countplot(data=df, x='vehicle_type', hue='accident_severity', palette='Set3')
plt.title('Accident Severity by Vehicle Type')
plt.xlabel('Vehicle Type')
plt.ylabel('Count of Accidents')
plt.xticks(rotation=45)
plt.show()
```

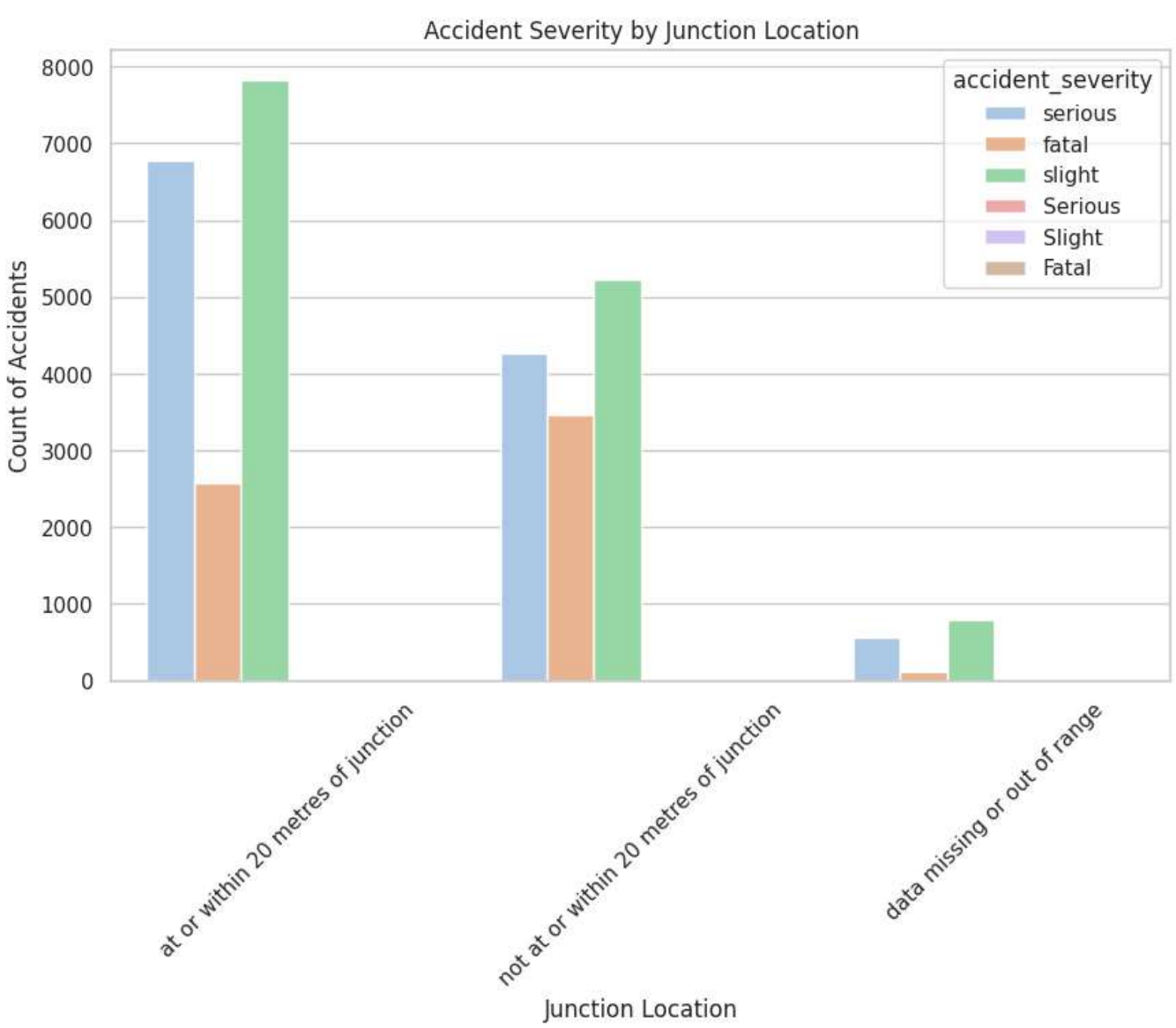



```
#  Age of oldest driver vs accident severity
plt.figure(figsize=(10, 6))
sns.boxplot(data=df, x='accident_severity', y='age_of_oldest_driver', palette='coolwarm')
plt.title('Age of Oldest Driver vs Accident Severity')
plt.xlabel('Accident Severity')
plt.ylabel('Age of Oldest Driver')
plt.show()
```

```
<ipython-input-26-a2bc2b28b015>:3: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0.

  sns.boxplot(data=df, x='accident_severity', y='age_of_oldest_driver', palette='coolwar
```




```
#  Accident severity at different junction types
plt.figure(figsize=(10, 6))
sns.countplot(data=df, x='junction_location', hue='accident_severity', palette='pastel')
plt.title('Accident Severity by Junction Location')
plt.xlabel('Junction Location')
plt.ylabel('Count of Accidents')
plt.xticks(rotation=45)
plt.show()
```




```
#  Distribution of accident severity
plt.figure(figsize=(8, 5))
sns.countplot(data=df, x='accident_severity', palette='viridis')
plt.title('Distribution of Accident Severity')
plt.xlabel('Accident Severity')
plt.ylabel('Count of Accidents')
plt.show()
```

```
<ipython-input-30-897f32e327e0>:3: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0.

  sns.countplot(data=df, x='accident_severity', palette='viridis')
```