cleanup-analysis-visualization

September 15, 2024

1 DATA CLEANUP, ANALYSIS & VISUALIZATION

1.0.1 Set up essential libraries for Data Analysis

[242]:

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

1.0.2 Import the Excel File into a Pandas DataFrame

[244]:

```
df = pd.read_excel('Hyderabad Traffic Monitoring System.xlsx')

df.head()
```

[244]:

```
            Timestamp    Location Direction  Vehicle_Count  Avg_Speed  \
0 2021-01-01 00:00:00  Location_A     South             93  43.794956
1 2021-01-01 01:00:00  Location_D     South              3  85.103826
2 2021-01-01 02:00:00  Location_B      East             98  51.720459
3 2021-01-01 03:00:00  Location_A     North             42  90.488256
4 2021-01-01 04:00:00  Location_D     North             77  66.501830

   Peak_Hour Weather_Condition  Visibility  Temperature   Humidity  \
0       True             Foggy    2.574827    33.503269  58.121361
1       True             Foggy    4.824104    29.303575  22.272571
2      False             Sunny    8.849263    24.816372  60.610620
3      False             Foggy    5.176548    24.389966  73.976233
4      False             Sunny    7.297831    32.654186  46.335860

   Wind_Speed  Accidents Roadwork Traffic_Signal_Status Congestion_Level  \
0    6.712738          1       No               Working              Low
1   10.441158          2       No           Not Working        Very High
2    5.674892          3      Yes               Working             High
3    1.500481          2       No               Working        Very High
4   14.403852          3      Yes           Not Working             High

   Duplicate_Column           Area
```

```
0                 93  Banjara Hills
1                  3       Ameerpet
2                 98       Begumpet
3                 42     Nallakunta
4                 77       Kondapur
```

1.0.3 Let's dive into data cleanup and transformation

Handling duplicate rows

[247]:
```
duplicates_count = df.duplicated(subset='Timestamp', keep='first').sum()

df.drop_duplicates(subset='Timestamp', keep='first', inplace=True)

print(f"Number of duplicate records in the dataset: {duplicates_count}")

print("\nUPDATED DATASET SAMPLE:\n")
df.head()
```

```
Number of duplicate records in the dataset: 10

UPDATED DATASET SAMPLE:
```

[247]:
```
            Timestamp    Location Direction  Vehicle_Count  Avg_Speed  \
0 2021-01-01 00:00:00  Location_A     South             93  43.794956
1 2021-01-01 01:00:00  Location_D     South              3  85.103826
2 2021-01-01 02:00:00  Location_B      East             98  51.720459
3 2021-01-01 03:00:00  Location_A     North             42  90.488256
4 2021-01-01 04:00:00  Location_D     North             77  66.501830

   Peak_Hour Weather_Condition  Visibility  Temperature   Humidity  \
0       True             Foggy    2.574827    33.503269  58.121361
1       True             Foggy    4.824104    29.303575  22.272571
2      False             Sunny    8.849263    24.816372  60.610620
3      False             Foggy    5.176548    24.389966  73.976233
4      False             Sunny    7.297831    32.654186  46.335860

   Wind_Speed  Accidents Roadwork Traffic_Signal_Status Congestion_Level  \
0    6.712738          1       No               Working              Low
1   10.441158          2       No           Not Working        Very High
2    5.674892          3      Yes               Working             High
3    1.500481          2       No               Working        Very High
4   14.403852          3      Yes           Not Working             High

   Duplicate_Column           Area
0                93  Banjara Hills
```

```
1                 3       Ameerpet
2                98       Begumpet
3                42     Nallakunta
4                77       Kondapur
```

Handling null values

[249]:
```
print(f"Total null values in the dataset: {df.isnull().sum().sum()}")

df.dropna(inplace=True)

print("\nUPDATED DATASET SAMPLE:\n")
df.head()
```

```
Total null values in the dataset: 0

UPDATED DATASET SAMPLE:

```

[249]:
```
            Timestamp    Location Direction  Vehicle_Count  Avg_Speed  \
0 2021-01-01 00:00:00  Location_A     South             93  43.794956
1 2021-01-01 01:00:00  Location_D     South              3  85.103826
2 2021-01-01 02:00:00  Location_B      East             98  51.720459
3 2021-01-01 03:00:00  Location_A     North             42  90.488256
4 2021-01-01 04:00:00  Location_D     North             77  66.501830

   Peak_Hour Weather_Condition  Visibility  Temperature   Humidity  \
0       True             Foggy    2.574827    33.503269  58.121361
1       True             Foggy    4.824104    29.303575  22.272571
2      False             Sunny    8.849263    24.816372  60.610620
3      False             Foggy    5.176548    24.389966  73.976233
4      False             Sunny    7.297831    32.654186  46.335860

   Wind_Speed  Accidents Roadwork Traffic_Signal_Status Congestion_Level  \
0    6.712738          1       No               Working              Low
1   10.441158          2       No           Not Working        Very High
2    5.674892          3      Yes               Working             High
3    1.500481          2       No               Working        Very High
4   14.403852          3      Yes           Not Working             High

   Duplicate_Column           Area
0                93  Banjara Hills
1                 3       Ameerpet
2                98       Begumpet
3                42     Nallakunta
4                77       Kondapur
```

Renaming the columns

[251]:

```
df.columns = [col.replace('_', ' ') for col in df.columns]

df.rename(columns={
    'Avg Speed': 'Average Speed (in km/h)',
    'Peak Hour': 'Peak Hour?',
    'Visibility': 'Visibility (in km)',
    'Temperature': 'Temperature (in °C)',
    'Humidity': 'Humidity (in %)',
    'Wind Speed': 'Wind Speed (in km/h)',
    'Roadwork': 'Roadwork?'
}, inplace=True)

print("\nUPDATED DATASET SAMPLE:\n")
df.head()
```

```
UPDATED DATASET SAMPLE:
```

[251]:

```
            Timestamp    Location Direction  Vehicle Count  \
0 2021-01-01 00:00:00  Location_A     South             93
1 2021-01-01 01:00:00  Location_D     South              3
2 2021-01-01 02:00:00  Location_B      East             98
3 2021-01-01 03:00:00  Location_A     North             42
4 2021-01-01 04:00:00  Location_D     North             77

   Average Speed (in km/h)  Peak Hour? Weather Condition  Visibility (in km)  \
0                43.794956        True             Foggy            2.574827
1                85.103826        True             Foggy            4.824104
2                51.720459       False             Sunny            8.849263
3                90.488256       False             Foggy            5.176548
4                66.501830       False             Sunny            7.297831

   Temperature (in °C)  Humidity (in %)  Wind Speed (in km/h)  Accidents  \
0            33.503269        58.121361              6.712738          1
1            29.303575        22.272571             10.441158          2
2            24.816372        60.610620              5.674892          3
3            24.389966        73.976233              1.500481          2
4            32.654186        46.335860             14.403852          3

  Roadwork? Traffic Signal Status Congestion Level  Duplicate Column  \
0        No               Working              Low                93
1        No           Not Working        Very High                 3
2       Yes               Working             High                98
3        No               Working        Very High                42
4       Yes           Not Working             High                77
```

```
            Area
0  Banjara Hills
1       Ameerpet
2       Begumpet
3     Nallakunta
4       Kondapur
```

Splitting 'Timestamp' column into 'Date' & 'Time' columns

```
[253]: df['Date'] = df['Timestamp'].dt.date

       df['Time'] = df['Timestamp'].dt.time

       print("\nUPDATED DATASET SAMPLE:\n")
       df.head()
```

```

UPDATED DATASET SAMPLE:

```

```
[253]:             Timestamp    Location Direction  Vehicle Count  \
       0 2021-01-01 00:00:00  Location_A     South             93
       1 2021-01-01 01:00:00  Location_D     South              3
       2 2021-01-01 02:00:00  Location_B      East             98
       3 2021-01-01 03:00:00  Location_A     North             42
       4 2021-01-01 04:00:00  Location_D     North             77

          Average Speed (in km/h)  Peak Hour? Weather Condition  Visibility (in km)  \
       0                43.794956        True             Foggy            2.574827
       1                85.103826        True             Foggy            4.824104
       2                51.720459       False             Sunny            8.849263
       3                90.488256       False             Foggy            5.176548
       4                66.501830       False             Sunny            7.297831

          Temperature (in °C)  Humidity (in %)  Wind Speed (in km/h)  Accidents  \
       0            33.503269        58.121361              6.712738          1
       1            29.303575        22.272571             10.441158          2
       2            24.816372        60.610620              5.674892          3
       3            24.389966        73.976233              1.500481          2
       4            32.654186        46.335860             14.403852          3

         Roadwork? Traffic Signal Status Congestion Level  Duplicate Column  \
       0        No               Working              Low                93
       1        No           Not Working        Very High                 3
       2       Yes               Working             High                98
       3        No               Working        Very High                42
       4       Yes           Not Working             High                77
```

```
            Area        Date      Time
0  Banjara Hills  2021-01-01  00:00:00
1       Ameerpet  2021-01-01  01:00:00
2       Begumpet  2021-01-01  02:00:00
3     Nallakunta  2021-01-01  03:00:00
4       Kondapur  2021-01-01  04:00:00
```

Refining the values of the 'Location' column

[255]:
```
df['Location'] = df['Location'].str.replace('Location_', '')

print("\nUPDATED DATASET SAMPLE:\n")
df.head()
```

```
UPDATED DATASET SAMPLE:
```

[255]:
```
            Timestamp Location Direction  Vehicle Count  \
0 2021-01-01 00:00:00        A     South             93
1 2021-01-01 01:00:00        D     South              3
2 2021-01-01 02:00:00        B      East             98
3 2021-01-01 03:00:00        A     North             42
4 2021-01-01 04:00:00        D     North             77

   Average Speed (in km/h)  Peak Hour? Weather Condition  Visibility (in km)  \
0                43.794956        True             Foggy            2.574827
1                85.103826        True             Foggy            4.824104
2                51.720459       False             Sunny            8.849263
3                90.488256       False             Foggy            5.176548
4                66.501830       False             Sunny            7.297831

   Temperature (in °C)  Humidity (in %)  Wind Speed (in km/h)  Accidents  \
0            33.503269        58.121361              6.712738          1
1            29.303575        22.272571             10.441158          2
2            24.816372        60.610620              5.674892          3
3            24.389966        73.976233              1.500481          2
4            32.654186        46.335860             14.403852          3

  Roadwork? Traffic Signal Status Congestion Level  Duplicate Column  \
0        No               Working              Low                93
1        No           Not Working        Very High                 3
2       Yes               Working             High                98
3        No               Working        Very High                42
4       Yes           Not Working             High                77
```

```
            Area        Date      Time
0  Banjara Hills  2021-01-01  00:00:00
1       Ameerpet  2021-01-01  01:00:00
2       Begumpet  2021-01-01  02:00:00
3     Nallakunta  2021-01-01  03:00:00
4       Kondapur  2021-01-01  04:00:00
```

Refining the values of 'Average Speed' column

[257]:

```
df['Average Speed (in km/h)'] = df['Average Speed (in km/h)'].round()

print("\nUPDATED DATASET SAMPLE:\n")
df.head()
```

```

UPDATED DATASET SAMPLE:

```

[257]:

```
            Timestamp Location Direction  Vehicle Count  \
0 2021-01-01 00:00:00        A     South             93
1 2021-01-01 01:00:00        D     South              3
2 2021-01-01 02:00:00        B      East             98
3 2021-01-01 03:00:00        A     North             42
4 2021-01-01 04:00:00        D     North             77

   Average Speed (in km/h)  Peak Hour? Weather Condition  Visibility (in km)  \
0                     44.0        True             Foggy            2.574827
1                     85.0        True             Foggy            4.824104
2                     52.0       False             Sunny            8.849263
3                     90.0       False             Foggy            5.176548
4                     67.0       False             Sunny            7.297831

   Temperature (in °C)  Humidity (in %)  Wind Speed (in km/h)  Accidents  \
0            33.503269        58.121361              6.712738          1
1            29.303575        22.272571             10.441158          2
2            24.816372        60.610620              5.674892          3
3            24.389966        73.976233              1.500481          2
4            32.654186        46.335860             14.403852          3

  Roadwork? Traffic Signal Status Congestion Level  Duplicate Column  \
0        No               Working              Low                93
1        No           Not Working        Very High                 3
2       Yes               Working             High                98
3        No               Working        Very High                42
4       Yes           Not Working             High                77

            Area        Date      Time
```

```
0  Banjara Hills  2021-01-01  00:00:00
1       Ameerpet  2021-01-01  01:00:00
2       Begumpet  2021-01-01  02:00:00
3     Nallakunta  2021-01-01  03:00:00
4       Kondapur  2021-01-01  04:00:00
```

Refining the values of ‘Peak Hour?’ column

[259]:
```
df['Peak Hour?'] = df['Peak Hour?'].replace({True: 'Yes', False: 'No'})

print("\nUPDATED DATASET SAMPLE:\n")
df.head()
```

```
UPDATED DATASET SAMPLE:
```

[259]:
```
            Timestamp Location Direction  Vehicle Count  \
0 2021-01-01 00:00:00        A     South             93
1 2021-01-01 01:00:00        D     South              3
2 2021-01-01 02:00:00        B      East             98
3 2021-01-01 03:00:00        A     North             42
4 2021-01-01 04:00:00        D     North             77

   Average Speed (in km/h) Peak Hour? Weather Condition  Visibility (in km)  \
0                     44.0        Yes             Foggy            2.574827
1                     85.0        Yes             Foggy            4.824104
2                     52.0         No             Sunny            8.849263
3                     90.0         No             Foggy            5.176548
4                     67.0         No             Sunny            7.297831

   Temperature (in °C)  Humidity (in %)  Wind Speed (in km/h)  Accidents  \
0            33.503269        58.121361              6.712738          1
1            29.303575        22.272571             10.441158          2
2            24.816372        60.610620              5.674892          3
3            24.389966        73.976233              1.500481          2
4            32.654186        46.335860             14.403852          3

  Roadwork? Traffic Signal Status Congestion Level  Duplicate Column  \
0        No               Working              Low                93
1        No           Not Working        Very High                 3
2       Yes               Working             High                98
3        No               Working        Very High                42
4       Yes           Not Working             High                77

           Area        Date      Time
0  Banjara Hills  2021-01-01  00:00:00
```

```
1       Ameerpet  2021-01-01  01:00:00
2       Begumpet  2021-01-01  02:00:00
3     Nallakunta  2021-01-01  03:00:00
4       Kondapur  2021-01-01  04:00:00
```

Refining the values of 'Weather Condition' column

[261]:

```
import numpy as np
import datetime

def replace_weather_condition(row):
    time = row['Time']

    if isinstance(time, str):
        time = datetime.datetime.strptime(time, '%H:%M:%S').time()

    weather = row['Weather Condition']
    weather_conditions = [condition for condition in df['Weather Condition'].
 ↪unique() if condition != 'Sunny']

    if datetime.time(18, 0, 0) <= time <= datetime.time(23, 59, 59) or datetime.
 ↪time(0, 0, 0) <= time <= datetime.time(8, 0, 0):
        if weather == 'Sunny':
            return np.random.choice(weather_conditions)

    return weather

df['Weather Condition'] = df.apply(replace_weather_condition, axis=1)

print("\nUPDATED DATASET SAMPLE:\n")
df.head()
```

```
UPDATED DATASET SAMPLE:
```

[261]:

```
            Timestamp Location Direction  Vehicle Count  \
0 2021-01-01 00:00:00        A     South             93
1 2021-01-01 01:00:00        D     South              3
2 2021-01-01 02:00:00        B      East             98
3 2021-01-01 03:00:00        A     North             42
4 2021-01-01 04:00:00        D     North             77

   Average Speed (in km/h) Peak Hour? Weather Condition  Visibility (in km)  \
0                     44.0        Yes             Foggy            2.574827
1                     85.0        Yes             Foggy            4.824104
2                     52.0         No            Cloudy            8.849263
```

```
3                     90.0         No             Foggy            5.176548
4                     67.0         No             Windy            7.297831

   Temperature (in °C)  Humidity (in %)  Wind Speed (in km/h)  Accidents  \
0            33.503269        58.121361              6.712738          1
1            29.303575        22.272571             10.441158          2
2            24.816372        60.610620              5.674892          3
3            24.389966        73.976233              1.500481          2
4            32.654186        46.335860             14.403852          3

  Roadwork? Traffic Signal Status Congestion Level  Duplicate Column  \
0        No               Working              Low                93
1        No           Not Working        Very High                 3
2       Yes               Working             High                98
3        No               Working        Very High                42
4       Yes           Not Working             High                77

            Area        Date      Time
0  Banjara Hills  2021-01-01  00:00:00
1       Ameerpet  2021-01-01  01:00:00
2       Begumpet  2021-01-01  02:00:00
3     Nallakunta  2021-01-01  03:00:00
4       Kondapur  2021-01-01  04:00:00
```

Refining the values of ‘Visibility’ column

[263]:

```
df['Visibility (in km)'] = df['Visibility (in km)'].round()

print("\nUPDATED DATASET SAMPLE:\n")
df.head()
```

```

UPDATED DATASET SAMPLE:

```

[263]:

```
            Timestamp Location Direction  Vehicle Count  \
0 2021-01-01 00:00:00        A     South             93
1 2021-01-01 01:00:00        D     South              3
2 2021-01-01 02:00:00        B      East             98
3 2021-01-01 03:00:00        A     North             42
4 2021-01-01 04:00:00        D     North             77

   Average Speed (in km/h) Peak Hour? Weather Condition  Visibility (in km)  \
0                     44.0        Yes             Foggy                 3.0
1                     85.0        Yes             Foggy                 5.0
2                     52.0         No            Cloudy                 9.0
3                     90.0         No             Foggy                 5.0
```

```
4                     67.0         No             Windy                 7.0

   Temperature (in °C)  Humidity (in %)  Wind Speed (in km/h)  Accidents  \
0            33.503269        58.121361              6.712738          1
1            29.303575        22.272571             10.441158          2
2            24.816372        60.610620              5.674892          3
3            24.389966        73.976233              1.500481          2
4            32.654186        46.335860             14.403852          3

  Roadwork? Traffic Signal Status Congestion Level  Duplicate Column  \
0        No               Working              Low                93
1        No           Not Working        Very High                 3
2       Yes               Working             High                98
3        No               Working        Very High                42
4       Yes           Not Working             High                77

            Area        Date      Time
0  Banjara Hills  2021-01-01  00:00:00
1       Ameerpet  2021-01-01  01:00:00
2       Begumpet  2021-01-01  02:00:00
3     Nallakunta  2021-01-01  03:00:00
4       Kondapur  2021-01-01  04:00:00
```

Refining the values of 'Temperature' column

[265]:
```
df['Temperature (in °C)'] = df['Temperature (in °C)'].round()

print("\nUPDATED DATASET SAMPLE:\n")
df.head()
```

```
UPDATED DATASET SAMPLE:
```

[265]:
```
            Timestamp Location Direction  Vehicle Count  \
0 2021-01-01 00:00:00        A     South             93
1 2021-01-01 01:00:00        D     South              3
2 2021-01-01 02:00:00        B      East             98
3 2021-01-01 03:00:00        A     North             42
4 2021-01-01 04:00:00        D     North             77

   Average Speed (in km/h) Peak Hour? Weather Condition  Visibility (in km)  \
0                     44.0        Yes             Foggy                 3.0
1                     85.0        Yes             Foggy                 5.0
2                     52.0         No            Cloudy                 9.0
3                     90.0         No             Foggy                 5.0
4                     67.0         No             Windy                 7.0
```

```
   Temperature (in °C)  Humidity (in %)  Wind Speed (in km/h)  Accidents  \
0                 34.0        58.121361              6.712738          1
1                 29.0        22.272571             10.441158          2
2                 25.0        60.610620              5.674892          3
3                 24.0        73.976233              1.500481          2
4                 33.0        46.335860             14.403852          3

  Roadwork? Traffic Signal Status Congestion Level  Duplicate Column  \
0        No               Working              Low                93
1        No           Not Working        Very High                 3
2       Yes               Working             High                98
3        No               Working        Very High                42
4       Yes           Not Working             High                77

            Area        Date      Time
0  Banjara Hills  2021-01-01  00:00:00
1       Ameerpet  2021-01-01  01:00:00
2       Begumpet  2021-01-01  02:00:00
3     Nallakunta  2021-01-01  03:00:00
4       Kondapur  2021-01-01  04:00:00
```

Refining the values of 'Humidity' column

[267]:
```python
df['Humidity (in %)'] = df['Humidity (in %)'].round()

print("\nUPDATED DATASET SAMPLE:\n")
df.head()
```

```

UPDATED DATASET SAMPLE:

```

[267]:
```
            Timestamp Location Direction  Vehicle Count  \
0 2021-01-01 00:00:00        A     South             93
1 2021-01-01 01:00:00        D     South              3
2 2021-01-01 02:00:00        B      East             98
3 2021-01-01 03:00:00        A     North             42
4 2021-01-01 04:00:00        D     North             77

   Average Speed (in km/h) Peak Hour? Weather Condition  Visibility (in km)  \
0                     44.0        Yes             Foggy                 3.0
1                     85.0        Yes             Foggy                 5.0
2                     52.0         No            Cloudy                 9.0
3                     90.0         No             Foggy                 5.0
4                     67.0         No             Windy                 7.0
```

```
   Temperature (in °C)  Humidity (in %)  Wind Speed (in km/h)  Accidents  \
0                 34.0             58.0              6.712738          1
1                 29.0             22.0             10.441158          2
2                 25.0             61.0              5.674892          3
3                 24.0             74.0              1.500481          2
4                 33.0             46.0             14.403852          3

  Roadwork? Traffic Signal Status Congestion Level  Duplicate Column  \
0        No               Working              Low                93
1        No           Not Working        Very High                 3
2       Yes               Working             High                98
3        No               Working        Very High                42
4       Yes           Not Working             High                77

            Area        Date      Time
0  Banjara Hills  2021-01-01  00:00:00
1       Ameerpet  2021-01-01  01:00:00
2       Begumpet  2021-01-01  02:00:00
3     Nallakunta  2021-01-01  03:00:00
4       Kondapur  2021-01-01  04:00:00
```

Refining the values of 'Wind Speed' column

```
[269]: df['Wind Speed (in km/h)'] = df['Wind Speed (in km/h)'].round()

       print("\nUPDATED DATASET SAMPLE:\n")
       df.head()
```

```
UPDATED DATASET SAMPLE:
```

```
[269]:             Timestamp Location Direction  Vehicle Count  \
       0 2021-01-01 00:00:00        A     South             93
       1 2021-01-01 01:00:00        D     South              3
       2 2021-01-01 02:00:00        B      East             98
       3 2021-01-01 03:00:00        A     North             42
       4 2021-01-01 04:00:00        D     North             77

          Average Speed (in km/h) Peak Hour? Weather Condition  Visibility (in km)  \
       0                     44.0        Yes             Foggy                 3.0
       1                     85.0        Yes             Foggy                 5.0
       2                     52.0         No            Cloudy                 9.0
       3                     90.0         No             Foggy                 5.0
       4                     67.0         No             Windy                 7.0

          Temperature (in °C)  Humidity (in %)  Wind Speed (in km/h)  Accidents  \
       0                 34.0             58.0                   7.0          1
       1                 29.0             22.0                  10.0          2
       2                 25.0             61.0                   6.0          3
       3                 24.0             74.0                   2.0          2
       4                 33.0             46.0                  14.0          3

         Roadwork? Traffic Signal Status Congestion Level  Duplicate Column  \
       0        No               Working              Low                93
       1        No           Not Working        Very High                 3
       2       Yes               Working             High                98
       3        No               Working        Very High                42
       4       Yes           Not Working             High                77

                   Area        Date      Time
       0  Banjara Hills  2021-01-01  00:00:00
       1       Ameerpet  2021-01-01  01:00:00
       2       Begumpet  2021-01-01  02:00:00
       3     Nallakunta  2021-01-01  03:00:00
       4       Kondapur  2021-01-01  04:00:00
```

Modifying the ‘Accidents’ column

[271]:

```python
def categorize_accidents(accidents):
    if accidents == 0:
        return 'None'
    elif accidents == 1:
        return 'Low'
    elif accidents == 2:
        return 'Moderate'
    elif accidents == 3:
        return 'High'
    else:
        return 'Unknown'

df['Accident Level'] = df['Accidents'].apply(categorize_accidents)

df.drop(columns=['Accidents'], inplace=True)

print("\nUPDATED DATASET SAMPLE:\n")
df.head()
```

```

UPDATED DATASET SAMPLE:

```

[271]:

```
            Timestamp Location Direction  Vehicle Count  \
0 2021-01-01 00:00:00        A     South             93
```

```
1 2021-01-01 01:00:00        D     South              3
2 2021-01-01 02:00:00        B      East             98
3 2021-01-01 03:00:00        A     North             42
4 2021-01-01 04:00:00        D     North             77

   Average Speed (in km/h) Peak Hour? Weather Condition  Visibility (in km)  \
0                     44.0        Yes             Foggy                 3.0
1                     85.0        Yes             Foggy                 5.0
2                     52.0         No            Cloudy                 9.0
3                     90.0         No             Foggy                 5.0
4                     67.0         No             Windy                 7.0

   Temperature (in °C)  Humidity (in %)  Wind Speed (in km/h) Roadwork?  \
0                 34.0             58.0                   7.0        No
1                 29.0             22.0                  10.0        No
2                 25.0             61.0                   6.0       Yes
3                 24.0             74.0                   2.0        No
4                 33.0             46.0                  14.0       Yes

  Traffic Signal Status Congestion Level  Duplicate Column           Area  \
0               Working              Low                93  Banjara Hills
1           Not Working        Very High                 3       Ameerpet
2               Working             High                98       Begumpet
3               Working        Very High                42     Nallakunta
4           Not Working             High                77       Kondapur

         Date      Time Accident Level
0  2021-01-01  00:00:00            Low
1  2021-01-01  01:00:00       Moderate
2  2021-01-01  02:00:00           High
3  2021-01-01  03:00:00       Moderate
4  2021-01-01  04:00:00           High
```

Refining the values of 'Congestion Level' column

[273]:
```
def categorize_traffic_volume(vehicle_count):
    if 0 <= vehicle_count <= 15:
        return 'Minimal'
    elif 16 <= vehicle_count <= 35:
        return 'Low'
    elif 36 <= vehicle_count <= 60:
        return 'Moderate'
    elif 61 <= vehicle_count <= 85:
        return 'High'
    elif 86 <= vehicle_count <= 100:
        return 'Extreme'
    else:
        return 'Out of Range'

df['Congestion Level'] = ''

df['Congestion Level'] = df['Vehicle Count'].apply(categorize_traffic_volume)

print("UPDATED DATASET SAMPLE:\n")
df.head()
```

```
UPDATED DATASET SAMPLE:
```

```
[273]:
             Timestamp Location Direction  Vehicle Count  \
0 2021-01-01 00:00:00        A     South             93   
1 2021-01-01 01:00:00        D     South              3   
2 2021-01-01 02:00:00        B      East             98   
3 2021-01-01 03:00:00        A     North             42   
4 2021-01-01 04:00:00        D     North             77   

   Average Speed (in km/h) Peak Hour? Weather Condition  Visibility (in km)  \
0                     44.0        Yes             Foggy                 3.0   
1                     85.0        Yes             Foggy                 5.0   
2                     52.0         No            Cloudy                 9.0   
3                     90.0         No             Foggy                 5.0   
4                     67.0         No             Windy                 7.0   

   Temperature (in °C)  Humidity (in %)  Wind Speed (in km/h) Roadwork?  \
0                 34.0             58.0                   7.0        No   
1                 29.0             22.0                  10.0        No   
2                 25.0             61.0                   6.0       Yes   
3                 24.0             74.0                   2.0        No   
4                 33.0             46.0                  14.0       Yes   

  Traffic Signal Status Congestion Level  Duplicate Column           Area  \
0               Working          Extreme                93  Banjara Hills   
1           Not Working          Minimal                 3       Ameerpet   
2               Working          Extreme                98       Begumpet   
3               Working         Moderate                42     Nallakunta   
4           Not Working             High                77       Kondapur   

         Date      Time Accident Level  
0  2021-01-01  00:00:00            Low  
1  2021-01-01  01:00:00       Moderate  
2  2021-01-01  02:00:00           High  
3  2021-01-01  03:00:00       Moderate  
4  2021-01-01  04:00:00           High  
```

Removing the column 'Duplicate Column' from the dataset

[275]:
```
df.drop('Duplicate Column', axis=1, inplace=True)

print("\nUPDATED DATASET SAMPLE:\n")
df.head()
```

```
UPDATED DATASET SAMPLE:
```

[275]:
```
            Timestamp Location Direction  Vehicle Count  \
0 2021-01-01 00:00:00        A     South             93   
1 2021-01-01 01:00:00        D     South              3   
2 2021-01-01 02:00:00        B      East             98   
3 2021-01-01 03:00:00        A     North             42   
4 2021-01-01 04:00:00        D     North             77   

   Average Speed (in km/h) Peak Hour? Weather Condition  Visibility (in km)  \
0                     44.0        Yes             Foggy                 3.0   
1                     85.0        Yes             Foggy                 5.0   
2                     52.0         No            Cloudy                 9.0   
3                     90.0         No             Foggy                 5.0   
4                     67.0         No             Windy                 7.0   

   Temperature (in °C)  Humidity (in %)  Wind Speed (in km/h) Roadwork?  \
0                 34.0             58.0                   7.0        No   
1                 29.0             22.0                  10.0        No   
2                 25.0             61.0                   6.0       Yes   
3                 24.0             74.0                   2.0        No   
4                 33.0             46.0                  14.0       Yes   

  Traffic Signal Status Congestion Level          Area        Date      Time  \
0               Working          Extreme  Banjara Hills  2021-01-01  00:00:00   
1           Not Working          Minimal       Ameerpet  2021-01-01  01:00:00   
2               Working          Extreme       Begumpet  2021-01-01  02:00:00   
3               Working         Moderate     Nallakunta  2021-01-01  03:00:00   
4           Not Working             High       Kondapur  2021-01-01  04:00:00   

  Accident Level
0            Low
1       Moderate
2           High
3       Moderate
4           High
```

Added two dummy rows to the dataset for enhanced analysis

[277]:

```
new_rows = pd.DataFrame({
    'Timestamp': ['2021-01-03 22:00:00', '2021-01-03 23:00:00'],
    'Date': ['2021-01-03', '2021-01-03'],
    'Time': ['22:00:00', '23:00:00'],
    'Location': ['B', 'B'],
    'Direction': ['North', 'South'],
    'Vehicle Count': [48, 54],
    'Average Speed (in km/h)': [64, 55],
    'Peak Hour?': ['No', 'Yes'],
    'Weather Condition': ['Cloudy', 'Foggy'],
    'Visibility (in km)': [8.0, 5.0],
    'Temperature (in °C)': [30.0, 28.0],
    'Humidity (in %)': [65, 72],
    'Wind Speed (in km/h)': [10.0, 12.0],
    'Accident Level': ['Low', 'High'],
    'Roadwork?': ['No', 'Yes'],
    'Traffic Signal Status': ['Working', 'Not Working'],
    'Congestion Level': ['Medium', 'High'],
    'Area': ['Jubilee Hills', 'Banjara Hills']
})

df = pd.concat([df, new_rows], ignore_index=True)

print("\nUPDATED DATASET:\n")
df
```

```
UPDATED DATASET:
```

[277]:

```
              Timestamp Location Direction  Vehicle Count  \
0   2021-01-01 00:00:00        A     South             93
1   2021-01-01 01:00:00        D     South              3
2   2021-01-01 02:00:00        B      East             98
3   2021-01-01 03:00:00        A     North             42
4   2021-01-01 04:00:00        D     North             77
..                  …      …       …           …
67  2021-01-03 19:00:00        A     North              4
68  2021-01-03 20:00:00        A      East             67
69  2021-01-03 21:00:00        A     North             11
70  2021-01-03 22:00:00        B     North             48
71  2021-01-03 23:00:00        B     South             54

    Average Speed (in km/h) Peak Hour? Weather Condition  Visibility (in km)  \
0                      44.0        Yes             Foggy                 3.0
1                      85.0        Yes             Foggy                 5.0
2                      52.0         No            Cloudy                 9.0
```

```
3                      90.0         No             Foggy                 5.0
4                      67.0         No             Windy                 7.0
..                      …        …              …                 …
67                     33.0         No             Rainy                 9.0
68                     70.0        Yes             Windy                 2.0
69                     66.0        Yes             Rainy                 8.0
70                     64.0         No            Cloudy                 8.0
71                     55.0        Yes             Foggy                 5.0

    Temperature (in °C)  Humidity (in %)  Wind Speed (in km/h) Roadwork?  \
0                  34.0             58.0                   7.0        No   
1                  29.0             22.0                  10.0        No   
2                  25.0             61.0                   6.0       Yes   
3                  24.0             74.0                   2.0        No   
4                  33.0             46.0                  14.0       Yes   
..                  …              …                  …       …   
67                 25.0             70.0                   0.0        No   
68                 25.0             63.0                  14.0        No   
69                 28.0             67.0                   8.0        No   
70                 30.0             65.0                  10.0        No   
71                 28.0             72.0                  12.0       Yes   

   Traffic Signal Status Congestion Level           Area        Date  \
0                Working          Extreme  Banjara Hills  2021-01-01   
1            Not Working          Minimal       Ameerpet  2021-01-01   
2                Working          Extreme       Begumpet  2021-01-01   
3                Working         Moderate     Nallakunta  2021-01-01   
4            Not Working             High       Kondapur  2021-01-01   
..                   …              …           …         …   
67               Working          Minimal       Ameerpet  2021-01-03   
68           Not Working             High       Begumpet  2021-01-03   
69               Working          Minimal     Gachibowli  2021-01-03   
70               Working           Medium  Jubilee Hills  2021-01-03   
71           Not Working             High  Banjara Hills  2021-01-03   

        Time Accident Level
0   00:00:00            Low
1   01:00:00       Moderate
2   02:00:00           High
3   03:00:00       Moderate
4   04:00:00           High
..       …            …
67  19:00:00           High
68  20:00:00           High
69  21:00:00       Moderate
70  22:00:00            Low
71  23:00:00           High
```

```
[72 rows x 18 columns]
```

Adding custom index to the dataset

[279]:
```
traffic_update_id = [f'TUPD{i+1:03d}' for i in range(len(df))]

df['Traffic Update ID'] = traffic_update_id

df.set_index('Traffic Update ID', inplace=True)

print("\nUPDATED DATASET SAMPLE:\n")
df.head()
```

```
UPDATED DATASET SAMPLE:
```

[279]:
```
                             Timestamp Location Direction  Vehicle Count  \
Traffic Update ID
TUPD001            2021-01-01 00:00:00        A     South             93
TUPD002            2021-01-01 01:00:00        D     South              3
TUPD003            2021-01-01 02:00:00        B      East             98
TUPD004            2021-01-01 03:00:00        A     North             42
TUPD005            2021-01-01 04:00:00        D     North             77

                   Average Speed (in km/h) Peak Hour? Weather Condition  \
Traffic Update ID
TUPD001                               44.0        Yes             Foggy
TUPD002                               85.0        Yes             Foggy
TUPD003                               52.0         No            Cloudy
TUPD004                               90.0         No             Foggy
TUPD005                               67.0         No             Windy

                   Visibility (in km)  Temperature (in °C)  Humidity (in %)  \
Traffic Update ID
TUPD001                           3.0                 34.0             58.0
TUPD002                           5.0                 29.0             22.0
TUPD003                           9.0                 25.0             61.0
TUPD004                           5.0                 24.0             74.0
TUPD005                           7.0                 33.0             46.0

                   Wind Speed (in km/h) Roadwork? Traffic Signal Status  \
Traffic Update ID
TUPD001                             7.0        No               Working
TUPD002                            10.0        No           Not Working
TUPD003                             6.0       Yes               Working
```

```
TUPD004                             2.0        No              Working
TUPD005                            14.0       Yes          Not Working

                  Congestion Level           Area        Date      Time  \
Traffic Update ID
TUPD001                    Extreme  Banjara Hills  2021-01-01  00:00:00
TUPD002                    Minimal       Ameerpet  2021-01-01  01:00:00
TUPD003                    Extreme       Begumpet  2021-01-01  02:00:00
TUPD004                   Moderate     Nallakunta  2021-01-01  03:00:00
TUPD005                       High       Kondapur  2021-01-01  04:00:00

                  Accident Level
Traffic Update ID
TUPD001                      Low
TUPD002                 Moderate
TUPD003                     High
TUPD004                 Moderate
TUPD005                     High
```

Adding a new column 'Speed Level'

```
[281]: def categorize_average_speed(speed):
           if 0 <= speed <= 20:
               return 'Minimal'
           elif 21 <= speed <= 40:
               return 'Low'
           elif 41 <= speed <= 60:
               return 'Moderate'
           elif 61 <= speed <= 80:
               return 'High'
           elif 81 <= speed <= 100:
               return 'Extreme'
           else:
               return 'Out of Range'

       df['Speed Level'] = df['Average Speed (in km/h)'].
        ↪apply(categorize_average_speed)

       print("\nUPDATED DATASET SAMPLE:\n")
       df.head()
```

```
UPDATED DATASET SAMPLE:
```

```
[281]:                             Timestamp Location Direction  Vehicle Count  \
       Traffic Update ID
```

```
TUPD001            2021-01-01 00:00:00        A     South              93
TUPD002            2021-01-01 01:00:00        D     South               3
TUPD003            2021-01-01 02:00:00        B      East              98
TUPD004            2021-01-01 03:00:00        A     North              42
TUPD005            2021-01-01 04:00:00        D     North              77

                   Average Speed (in km/h) Peak Hour? Weather Condition  \
Traffic Update ID
TUPD001                               44.0        Yes             Foggy
TUPD002                               85.0        Yes             Foggy
TUPD003                               52.0         No            Cloudy
TUPD004                               90.0         No             Foggy
TUPD005                               67.0         No             Windy

                   Visibility (in km)  Temperature (in °C)  Humidity (in %)  \
Traffic Update ID
TUPD001                           3.0                 34.0             58.0
TUPD002                           5.0                 29.0             22.0
TUPD003                           9.0                 25.0             61.0
TUPD004                           5.0                 24.0             74.0
TUPD005                           7.0                 33.0             46.0

                   Wind Speed (in km/h) Roadwork? Traffic Signal Status  \
Traffic Update ID
TUPD001                             7.0        No               Working
TUPD002                            10.0        No           Not Working
TUPD003                             6.0       Yes               Working
TUPD004                             2.0        No               Working
TUPD005                            14.0       Yes           Not Working

                  Congestion Level           Area        Date      Time  \
Traffic Update ID
TUPD001                    Extreme  Banjara Hills  2021-01-01  00:00:00
TUPD002                    Minimal       Ameerpet  2021-01-01  01:00:00
TUPD003                    Extreme       Begumpet  2021-01-01  02:00:00
TUPD004                   Moderate     Nallakunta  2021-01-01  03:00:00
TUPD005                       High       Kondapur  2021-01-01  04:00:00

                  Accident Level Speed Level
Traffic Update ID
TUPD001                      Low    Moderate
TUPD002                 Moderate     Extreme
TUPD003                     High    Moderate
TUPD004                 Moderate     Extreme
TUPD005                     High        High
```

Adding a new column 'Visibility Level'

[283]:

```
def categorize_visibility(visibility):
    if 0 <= visibility <= 3:
        return 'Low'
    elif 4 <= visibility <= 7:
        return 'Moderate'
    elif 8 <= visibility <= 10:
        return 'High'
    else:
        return 'Out of Range'

df['Visibility Level'] = df['Visibility (in km)'].apply(categorize_visibility)

print("\nUPDATED DATASET SAMPLE:\n")
df.head()
```

```
UPDATED DATASET SAMPLE:
```

[283]:

```
                             Timestamp Location Direction  Vehicle Count  \
Traffic Update ID
TUPD001            2021-01-01 00:00:00        A     South             93
TUPD002            2021-01-01 01:00:00        D     South              3
TUPD003            2021-01-01 02:00:00        B      East             98
TUPD004            2021-01-01 03:00:00        A     North             42
TUPD005            2021-01-01 04:00:00        D     North             77

                   Average Speed (in km/h) Peak Hour? Weather Condition  \
Traffic Update ID
TUPD001                               44.0        Yes             Foggy
TUPD002                               85.0        Yes             Foggy
TUPD003                               52.0         No            Cloudy
TUPD004                               90.0         No             Foggy
TUPD005                               67.0         No             Windy

                   Visibility (in km)  Temperature (in °C)  Humidity (in %)  \
Traffic Update ID
TUPD001                           3.0                 34.0             58.0
TUPD002                           5.0                 29.0             22.0
TUPD003                           9.0                 25.0             61.0
TUPD004                           5.0                 24.0             74.0
TUPD005                           7.0                 33.0             46.0

                   Wind Speed (in km/h) Roadwork? Traffic Signal Status  \
Traffic Update ID
TUPD001                             7.0        No               Working
TUPD002                            10.0        No           Not Working
```

```
TUPD003                             6.0       Yes               Working
TUPD004                             2.0        No               Working
TUPD005                            14.0       Yes           Not Working

                  Congestion Level           Area        Date      Time  \
Traffic Update ID
TUPD001                    Extreme  Banjara Hills  2021-01-01  00:00:00
TUPD002                    Minimal       Ameerpet  2021-01-01  01:00:00
TUPD003                    Extreme       Begumpet  2021-01-01  02:00:00
TUPD004                   Moderate     Nallakunta  2021-01-01  03:00:00
TUPD005                       High       Kondapur  2021-01-01  04:00:00

                  Accident Level Speed Level Visibility Level
Traffic Update ID
TUPD001                      Low    Moderate              Low
TUPD002                 Moderate     Extreme         Moderate
TUPD003                     High    Moderate             High
TUPD004                 Moderate     Extreme         Moderate
TUPD005                     High        High         Moderate
```

Adding a new column 'Temperature Level'

[285]:

```python
def categorize_temperature(temp):
    if 15 <= temp <= 22:
        return 'Cool'
    elif 23 <= temp <= 28:
        return 'Moderate'
    elif 29 <= temp <= 34:
        return 'Warm'
    else:
        return 'Warm'

df['Temperature Level'] = df['Temperature (in °C)'].
 ↪apply(categorize_temperature)

print("\nUPDATED DATASET SAMPLE:\n")
df.head()
```

```
UPDATED DATASET SAMPLE:
```

[285]:

```
                             Timestamp Location Direction  Vehicle Count  \
Traffic Update ID
TUPD001            2021-01-01 00:00:00        A     South             93
TUPD002            2021-01-01 01:00:00        D     South              3
TUPD003            2021-01-01 02:00:00        B      East             98
```

```
TUPD004            2021-01-01 03:00:00        A     North             42
TUPD005            2021-01-01 04:00:00        D     North             77

                   Average Speed (in km/h) Peak Hour? Weather Condition  \
Traffic Update ID
TUPD001                              44.0        Yes             Foggy
TUPD002                              85.0        Yes             Foggy
TUPD003                              52.0         No            Cloudy
TUPD004                              90.0         No             Foggy
TUPD005                              67.0         No             Windy

                   Visibility (in km)  Temperature (in °C)  Humidity (in %)  \
Traffic Update ID
TUPD001                           3.0                 34.0             58.0
TUPD002                           5.0                 29.0             22.0
TUPD003                           9.0                 25.0             61.0
TUPD004                           5.0                 24.0             74.0
TUPD005                           7.0                 33.0             46.0

                   …  Roadwork? Traffic Signal Status Congestion Level  \
Traffic Update ID  …
TUPD001            …         No               Working          Extreme
TUPD002            …         No           Not Working          Minimal
TUPD003            …        Yes               Working          Extreme
TUPD004            …         No               Working         Moderate
TUPD005            …        Yes           Not Working             High

                            Area        Date      Time Accident Level  \
Traffic Update ID
TUPD001            Banjara Hills  2021-01-01  00:00:00            Low
TUPD002                 Ameerpet  2021-01-01  01:00:00       Moderate
TUPD003                 Begumpet  2021-01-01  02:00:00           High
TUPD004               Nallakunta  2021-01-01  03:00:00       Moderate
TUPD005                 Kondapur  2021-01-01  04:00:00           High

                  Speed Level Visibility Level Temperature Level
Traffic Update ID
TUPD001              Moderate              Low              Warm
TUPD002               Extreme         Moderate              Warm
TUPD003              Moderate             High          Moderate
TUPD004               Extreme         Moderate          Moderate
TUPD005                  High         Moderate              Warm

[5 rows x 21 columns]
```

Adding a new column ‘Humidity Level’

[287]:

```
def categorize_humidity(humidity):
    if 20 <= humidity <= 40:
        return 'Low'
    elif 41 <= humidity <= 60:
        return 'Moderate'
    elif 61 <= humidity <= 90:
        return 'High'
    else:
        return 'Out of Range'

df['Humidity Level'] = df['Humidity (in %)'].apply(categorize_humidity)

print("\nUPDATED DATASET SAMPLE:\n")
df.head()
```

```
UPDATED DATASET SAMPLE:
```

[287]:

```
                             Timestamp Location Direction  Vehicle Count  \
Traffic Update ID
TUPD001            2021-01-01 00:00:00        A     South             93
TUPD002            2021-01-01 01:00:00        D     South              3
TUPD003            2021-01-01 02:00:00        B      East             98
TUPD004            2021-01-01 03:00:00        A     North             42
TUPD005            2021-01-01 04:00:00        D     North             77

                   Average Speed (in km/h) Peak Hour? Weather Condition  \
Traffic Update ID
TUPD001                               44.0        Yes             Foggy
TUPD002                               85.0        Yes             Foggy
TUPD003                               52.0         No            Cloudy
TUPD004                               90.0         No             Foggy
TUPD005                               67.0         No             Windy

                   Visibility (in km)  Temperature (in °C)  Humidity (in %)  \
Traffic Update ID
TUPD001                           3.0                 34.0             58.0
TUPD002                           5.0                 29.0             22.0
TUPD003                           9.0                 25.0             61.0
TUPD004                           5.0                 24.0             74.0
TUPD005                           7.0                 33.0             46.0

                   …  Traffic Signal Status Congestion Level           Area  \
Traffic Update ID  …
TUPD001            …                Working          Extreme  Banjara Hills
TUPD002            …            Not Working          Minimal       Ameerpet
```

```
TUPD003            …                Working          Extreme       Begumpet
TUPD004            …                Working         Moderate     Nallakunta
TUPD005            …            Not Working             High       Kondapur

                         Date      Time Accident Level Speed Level   \
Traffic Update ID
TUPD001            2021-01-01  00:00:00            Low    Moderate
TUPD002            2021-01-01  01:00:00       Moderate     Extreme
TUPD003            2021-01-01  02:00:00           High    Moderate
TUPD004            2021-01-01  03:00:00       Moderate     Extreme
TUPD005            2021-01-01  04:00:00           High        High

                  Visibility Level Temperature Level Humidity Level
Traffic Update ID
TUPD001                        Low              Warm       Moderate
TUPD002                   Moderate              Warm            Low
TUPD003                       High          Moderate           High
TUPD004                   Moderate          Moderate           High
TUPD005                   Moderate              Warm       Moderate

[5 rows x 22 columns]
```

Adding a new column 'Wind Speed Level'

[289]:

```python
def categorize_wind_speed(speed):
    if 0 <= speed <= 5:
        return 'Light'
    elif 6 <= speed <= 10:
        return 'Moderate'
    elif 11 <= speed <= 15:
        return 'Strong'
    else:
        return 'Out of Range'

df['Wind Speed Level'] = df['Wind Speed (in km/h)'].apply(categorize_wind_speed)

print("\nUPDATED DATASET SAMPLE:\n")
df.head()
```

```
UPDATED DATASET SAMPLE:
```

[289]:

```
                             Timestamp Location Direction  Vehicle Count   \
Traffic Update ID
TUPD001            2021-01-01 00:00:00        A     South             93
TUPD002            2021-01-01 01:00:00        D     South              3
```

```
TUPD003            2021-01-01 02:00:00        B      East             98
TUPD004            2021-01-01 03:00:00        A     North             42
TUPD005            2021-01-01 04:00:00        D     North             77

                   Average Speed (in km/h) Peak Hour? Weather Condition  \
Traffic Update ID
TUPD001                               44.0        Yes             Foggy
TUPD002                               85.0        Yes             Foggy
TUPD003                               52.0         No            Cloudy
TUPD004                               90.0         No             Foggy
TUPD005                               67.0         No             Windy

                   Visibility (in km)  Temperature (in °C)  Humidity (in %)  \
Traffic Update ID
TUPD001                           3.0                 34.0             58.0
TUPD002                           5.0                 29.0             22.0
TUPD003                           9.0                 25.0             61.0
TUPD004                           5.0                 24.0             74.0
TUPD005                           7.0                 33.0             46.0

                   …  Congestion Level           Area        Date      Time  \
Traffic Update ID  …
TUPD001            …           Extreme  Banjara Hills  2021-01-01  00:00:00
TUPD002            …           Minimal       Ameerpet  2021-01-01  01:00:00
TUPD003            …           Extreme       Begumpet  2021-01-01  02:00:00
TUPD004            …          Moderate     Nallakunta  2021-01-01  03:00:00
TUPD005            …              High       Kondapur  2021-01-01  04:00:00

                  Accident Level Speed Level Visibility Level  \
Traffic Update ID
TUPD001                      Low    Moderate              Low
TUPD002                 Moderate     Extreme         Moderate
TUPD003                     High    Moderate             High
TUPD004                 Moderate     Extreme         Moderate
TUPD005                     High        High         Moderate

                  Temperature Level Humidity Level Wind Speed Level
Traffic Update ID
TUPD001                        Warm       Moderate         Moderate
TUPD002                        Warm            Low         Moderate
TUPD003                    Moderate           High         Moderate
TUPD004                    Moderate           High            Light
TUPD005                        Warm       Moderate           Strong

[5 rows x 23 columns]
```

Making final modifications to the dataset

[291]:

```
df['Timestamp'] = pd.to_datetime(df['Timestamp'], errors='coerce')
df.sort_values(by='Timestamp', ascending=True, inplace=True)

column_order = [
    'Timestamp', 'Date', 'Time', 'Area', 'Location', 'Direction',
    'Vehicle Count', 'Congestion Level', 'Average Speed (in km/h)',
    'Speed Level', 'Peak Hour?', 'Weather Condition', 'Visibility (in km)',
    'Visibility Level', 'Temperature (in °C)', 'Temperature Level', 'Humidity␣
 ↪(in %)',
    'Humidity Level', 'Wind Speed (in km/h)', 'Wind Speed Level',
    'Roadwork?', 'Traffic Signal Status', 'Accident Level'
]

df = df[column_order]

print("\nCOLUMN & DATATYPE DETAILS:\n")
df.info()

print("\nUPDATED DATASET SAMPLE:\n")
df.head()
```

```
COLUMN & DATATYPE DETAILS:

<class 'pandas.core.frame.DataFrame'>
Index: 72 entries, TUPD001 to TUPD072
Data columns (total 23 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   Timestamp                72 non-null     datetime64[ns]
 1   Date                     72 non-null     object
 2   Time                     72 non-null     object
 3   Area                     72 non-null     object
 4   Location                 72 non-null     object
 5   Direction                72 non-null     object
 6   Vehicle Count            72 non-null     int64
 7   Congestion Level         72 non-null     object
 8   Average Speed (in km/h)  72 non-null     float64
 9   Speed Level              72 non-null     object
 10  Peak Hour?               72 non-null     object
 11  Weather Condition        72 non-null     object
 12  Visibility (in km)       72 non-null     float64
 13  Visibility Level         72 non-null     object
 14  Temperature (in °C)      72 non-null     float64
 15  Temperature Level        72 non-null     object
 16  Humidity (in %)          72 non-null     float64
 17  Humidity Level           72 non-null     object
 18  Wind Speed (in km/h)     72 non-null     float64
 19  Wind Speed Level         72 non-null     object
 20  Roadwork?                72 non-null     object
 21  Traffic Signal Status    72 non-null     object
 22  Accident Level           72 non-null     object
dtypes: datetime64[ns](1), float64(5), int64(1), object(16)
memory usage: 13.5+ KB

UPDATED DATASET SAMPLE:
```

[291]:

```
                            Timestamp        Date      Time           Area  \
Traffic Update ID
TUPD001           2021-01-01 00:00:00  2021-01-01  00:00:00  Banjara Hills
TUPD002           2021-01-01 01:00:00  2021-01-01  01:00:00       Ameerpet
TUPD003           2021-01-01 02:00:00  2021-01-01  02:00:00       Begumpet
TUPD004           2021-01-01 03:00:00  2021-01-01  03:00:00     Nallakunta
TUPD005           2021-01-01 04:00:00  2021-01-01  04:00:00       Kondapur

                  Location Direction  Vehicle Count Congestion Level  \
Traffic Update ID
TUPD001                  A     South             93          Extreme
TUPD002                  D     South              3          Minimal
TUPD003                  B      East             98          Extreme
TUPD004                  A     North             42         Moderate
TUPD005                  D     North             77             High

                   Average Speed (in km/h) Speed Level  … Visibility Level  \
Traffic Update ID                                       …
TUPD001                               44.0    Moderate  …              Low
TUPD002                               85.0     Extreme  …         Moderate
TUPD003                               52.0    Moderate  …             High
TUPD004                               90.0     Extreme  …         Moderate
TUPD005                               67.0        High  …         Moderate

                  Temperature (in °C)  Temperature Level Humidity (in %)  \
Traffic Update ID
TUPD001                          34.0               Warm            58.0
TUPD002                          29.0               Warm            22.0
TUPD003                          25.0           Moderate            61.0
TUPD004                          24.0           Moderate            74.0
TUPD005                          33.0               Warm            46.0

                   Humidity Level Wind Speed (in km/h)  Wind Speed Level  \
Traffic Update ID
TUPD001                  Moderate                  7.0          Moderate
TUPD002                       Low                 10.0          Moderate
```

```
TUPD003                      High                 6.0          Moderate
TUPD004                      High                 2.0             Light
TUPD005                  Moderate                14.0            Strong

                  Roadwork?  Traffic Signal Status Accident Level
Traffic Update ID
TUPD001                  No                Working            Low
TUPD002                  No            Not Working       Moderate
TUPD003                 Yes                Working           High
TUPD004                  No                Working       Moderate
TUPD005                 Yes            Not Working           High

[5 rows x 23 columns]
```

Saving the DataFrame to a new Excel file

[293]:
```
df.to_excel('Hyderabad Traffic Monitoring System_Updated.xlsx', index=True)
```

1.0.4 Let's get started with statistical analysis and visualistions on the dataset

Comparing Traffic Volumes Across Different Locations

- **Statistical Analysis**

[297]:
```
from scipy.stats import levene, f_oneway, kruskal

locations = df['Area'].unique()
volume_groups = [df[df['Area'] == loc]['Vehicle Count'] for loc in locations]

levene_result = levene(*volume_groups)
print("LEVENE'S TEST FOR HOMOGENEITY OF VARIANCES")
print(f"P-value: {levene_result.pvalue:.4f}")
print("Conclusion: Levene's Test confirms equal variances across groups. ANOVA␣
  ↪assumptions are satisfied.\n")

anova_result = f_oneway(*volume_groups)
print("ANOVA FOR TRAFFIC VOLUME ACROSS LOCATIONS")
print(f"P-value: {anova_result.pvalue:.4f}")
if anova_result.pvalue < 0.05:
    print("Conclusion: ANOVA indicates significant variation in average speed␣
  ↪across areas. Let's perform the Kruskal-Wallis test.")
    kruskal_result = kruskal(*volume_groups)
    print("\nKRUSKAL-WALLIS TEST")
    print(f"P-value: {kruskal_result.pvalue:.4f}")
    if kruskal_result.pvalue < 0.05:
        print("Conclusion: Kruskal-Wallis test indicates significant variation␣
  ↪in traffic volume across areas (non-parametric).")
    else:
        print("Conclusion: Kruskal-Wallis test indicates no significant␣
  ↪variation in traffic volume across areas (non-parametric).")
else:
    print("Conclusion: ANOVA indicates no significant variation in traffic␣
  ↪volume across areas. Traffic volume is consistent across areas.\n")
    print("KRUSKAL-WALLIS TEST")
    print("Conclusion: Not performed as ANOVA results are reliable and do not␣
  ↪show significant variation.")
```

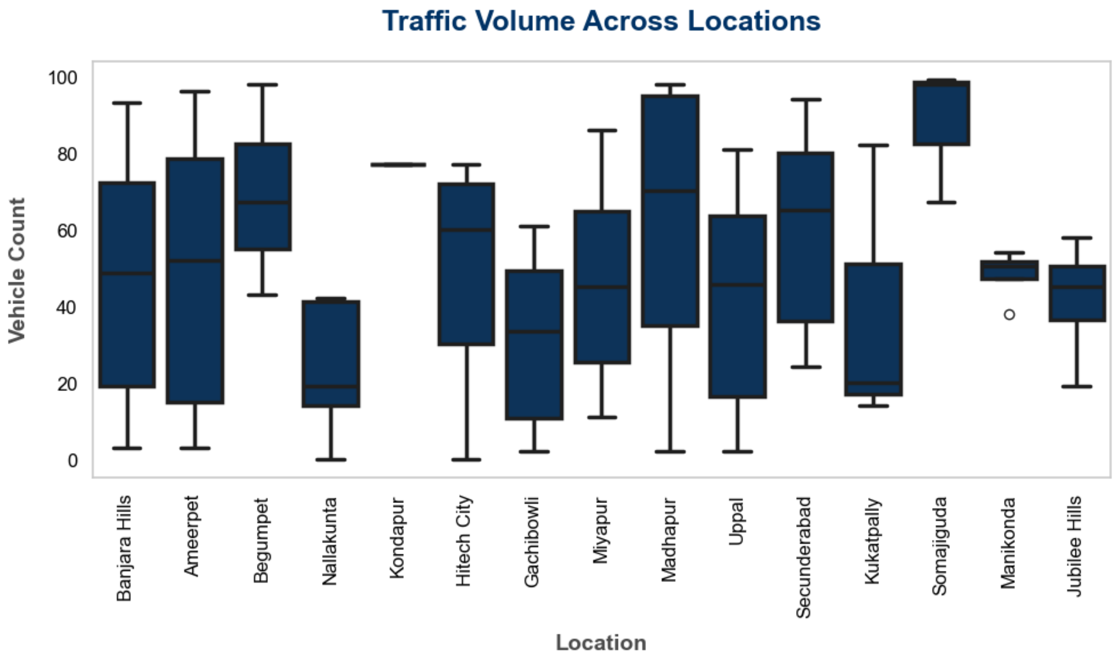
```
LEVENE'S TEST FOR HOMOGENEITY OF VARIANCES
P-value: 0.2563
Conclusion: Levene's Test confirms equal variances across groups. ANOVA
assumptions are satisfied.

ANOVA FOR TRAFFIC VOLUME ACROSS LOCATIONS
P-value: 0.3462
Conclusion: ANOVA indicates no significant variation in traffic volume across
areas. Traffic volume is consistent across areas.

KRUSKAL-WALLIS TEST
Conclusion: Not performed as ANOVA results are reliable and do not show
significant variation.
```

- **Visualization**

[470]:

```
sns.set_theme(style="whitegrid")

plot_color = '#003366'
title_color = '#4B4B4B'

plt.figure(figsize=(10, 6))

sns.boxplot(x='Area', y='Vehicle Count', data=df, color=plot_color, linewidth=2.
  ↪5)

plt.title('Traffic Volume Across Locations', fontsize=18, color=plot_color,␣
  ↪weight='bold', pad=20)
plt.xlabel('Location', fontsize=14, color=title_color, weight='bold',␣
  ↪labelpad=10)
plt.ylabel('Vehicle Count', fontsize=14, color=title_color, weight='bold',␣
  ↪labelpad=10)

plt.xticks(rotation=90, fontsize=12, color='black', weight='medium')
plt.yticks(fontsize=12, color='black', weight='medium')

plt.grid(False)
```

```python
plt.tight_layout()
plt.show()
```



Comparing Average Speed Across Different Locations

- Statistical Analysis

[302]:
```python
from scipy.stats import levene, f_oneway, kruskal

locations = df['Area'].unique()
speed_groups = [df[df['Area'] == loc]['Average Speed (in km/h)'] for loc in locations]

levene_result = levene(*speed_groups)
print("LEVENE'S TEST FOR HOMOGENEITY OF VARIANCES")
print(f"P-value: {levene_result.pvalue:.4f}")
if levene_result.pvalue < 0.05:
    print("Conclusion: Levene's Test indicates unequal variances across groups. ANOVA assumptions may not be satisfied.\n")
else:
    print("Conclusion: Levene's Test confirms equal variances across groups. ANOVA assumptions are satisfied.\n")

anova_result = f_oneway(*speed_groups)
print("ANOVA FOR AVERAGE SPEED ACROSS LOCATIONS")
print(f"P-value: {anova_result.pvalue:.4f}")
```

```
if anova_result.pvalue < 0.05:
    print("Conclusion: ANOVA indicates significant variation in average speed␣
 ↪across areas. Let's perform the Kruskal-Wallis test.")

    kruskal_result = kruskal(*speed_groups)
    print("\nKRUSKAL-WALLIS TEST")
    print(f"P-value: {kruskal_result.pvalue:.4f}")
    if kruskal_result.pvalue < 0.05:
        print("Conclusion: Kruskal-Wallis Test indicates significant variation␣
 ↪in average speed across areas (non-parametric).")
    else:
        print("Conclusion: Kruskal-Wallis Test indicates no significant␣
 ↪variation in average speed across areas (non-parametric).")
else:
    print("Conclusion: ANOVA indicates no significant variation in average␣
 ↪speed across areas. Average speed is consistent across areas.\n")
    print("KRUSKAL-WALLIS TEST")
    print("Conclusion: Not performed as ANOVA results are reliable and do not␣
 ↪show significant variation.")
```

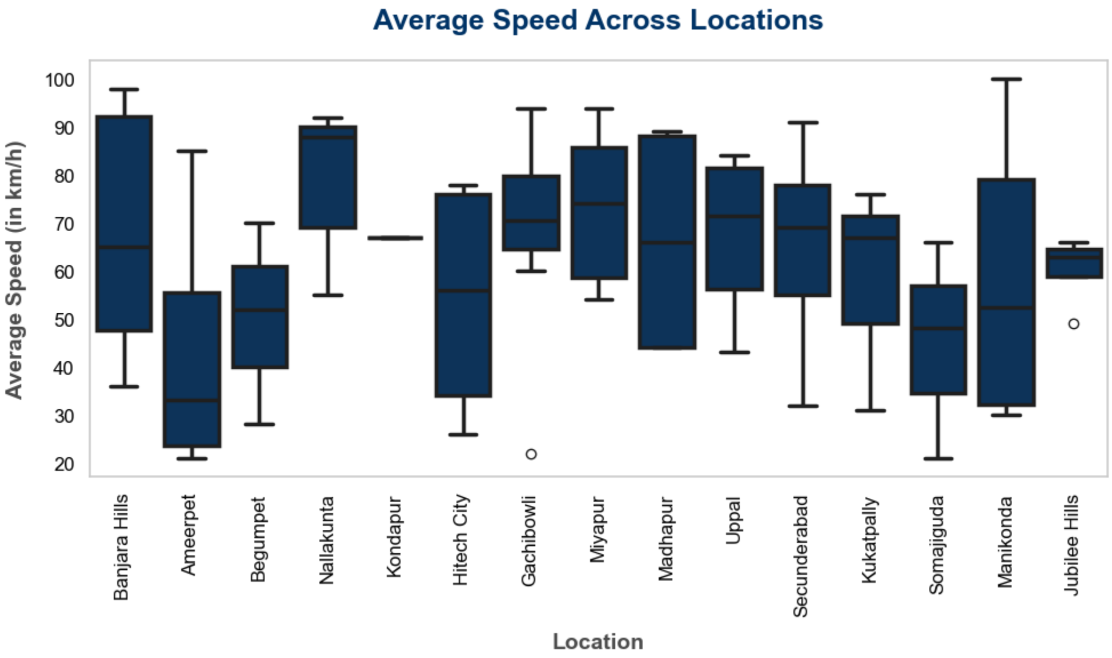
```
LEVENE'S TEST FOR HOMOGENEITY OF VARIANCES
P-value: 0.6057
Conclusion: Levene's Test confirms equal variances across groups. ANOVA
assumptions are satisfied.

ANOVA FOR AVERAGE SPEED ACROSS LOCATIONS
P-value: 0.4014
Conclusion: ANOVA indicates no significant variation in average speed across
areas. Average speed is consistent across areas.

KRUSKAL-WALLIS TEST
Conclusion: Not performed as ANOVA results are reliable and do not show
significant variation.
```

- **Visualization**

[472]:

```
plot_color = '#003366'
title_color = '#4B4B4B'

plt.figure(figsize=(10, 6))

sns.boxplot(x='Area', y='Average Speed (in km/h)', data=df, color=plot_color,␣
 ↪linewidth=2.5)

plt.title('Average Speed Across Locations', fontsize=18, color=plot_color,␣
 ↪weight='bold', pad=20)
```

```
plt.xlabel('Location', fontsize=14, color=title_color, weight='bold',␣
 ↪labelpad=10)
plt.ylabel('Average Speed (in km/h)', fontsize=14, color=title_color,␣
 ↪weight='bold', labelpad=10)

plt.xticks(rotation=90, fontsize=12, color='black', weight='medium')
plt.yticks(fontsize=12, color='black', weight='medium')

plt.grid(False)

plt.tight_layout()
plt.show()
```



Association Between Traffic Volume and Average Speed

- **Statistical Analysis**

[307]:
```
from scipy.stats import pearsonr, spearmanr

traffic_volume = df['Vehicle Count']
average_speed = df['Average Speed (in km/h)']

pearson_corr, pearson_pvalue = pearsonr(traffic_volume, average_speed)
print("PEARSON CORRELATION COEFFICIENT")
print(f"Correlation Coefficient: {pearson_corr:.4f}")
print(f"P-value: {pearson_pvalue:.4f}")
```

```
if pearson_pvalue < 0.05:
    print("Conclusion: Pearson Correlation indicates a significant linear␣
 ↪relationship between traffic volume and average speed.")
else:
    print("Conclusion: Pearson Correlation indicates no significant linear␣
 ↪relationship between traffic volume and average speed.\n")

spearman_corr, spearman_pvalue = spearmanr(traffic_volume, average_speed)
print("\nSPEARMAN RANK CORRELATION")
print(f"Correlation Coefficient: {spearman_corr:.4f}")
print(f"P-value: {spearman_pvalue:.4f}")
if spearman_pvalue < 0.05:
    print("Conclusion: Spearman Rank Correlation indicates a significant␣
 ↪monotonic relationship between traffic volume and average speed.")
else:
    print("Conclusion: Spearman Rank Correlation indicates no significant␣
 ↪monotonic relationship between traffic volume and average speed.")
```

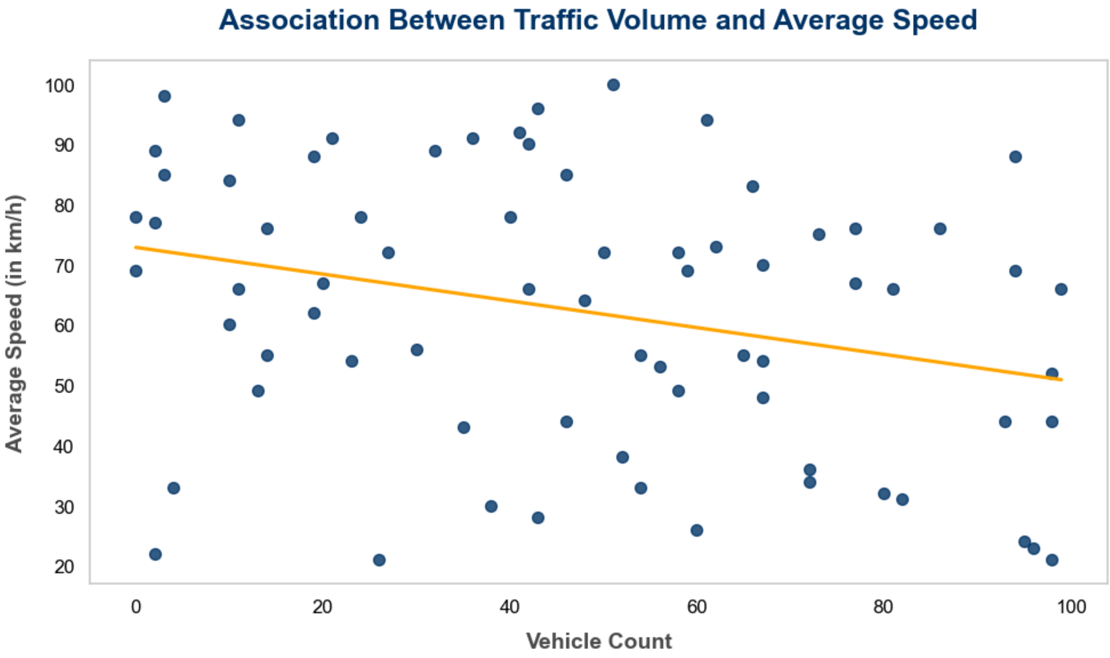
```
PEARSON CORRELATION COEFFICIENT
Correlation Coefficient: -0.2970
P-value: 0.0113
Conclusion: Pearson Correlation indicates a significant linear relationship
between traffic volume and average speed.

SPEARMAN RANK CORRELATION
Correlation Coefficient: -0.3077
P-value: 0.0086
Conclusion: Spearman Rank Correlation indicates a significant monotonic
relationship between traffic volume and average speed.
```

- **Visualization**

[474]:

```
plot_color = '#003366'
title_color = '#4B4B4B'

plt.figure(figsize=(10, 6))

sns.regplot(x='Vehicle Count', y='Average Speed (in km/h)', data=df,␣
 ↪scatter_kws={'color': plot_color, 's': 50}, line_kws={'color': 'orange'},␣
 ↪ci=None)

plt.title('Association Between Traffic Volume and Average Speed', fontsize=18,␣
 ↪color=plot_color, weight='bold', pad=20)
plt.xlabel('Vehicle Count', fontsize=14, color=title_color, weight='bold',␣
 ↪labelpad=10)
plt.ylabel('Average Speed (in km/h)', fontsize=14, color=title_color,␣
 ↪weight='bold', labelpad=10)
```

```
plt.xticks(fontsize=12, color='black', weight='medium')
plt.yticks(fontsize=12, color='black', weight='medium')

plt.grid(False)

plt.tight_layout()
plt.show()
```



Categorical Analysis of Congestion Levels Across Areas

- **Statistical Analysis**

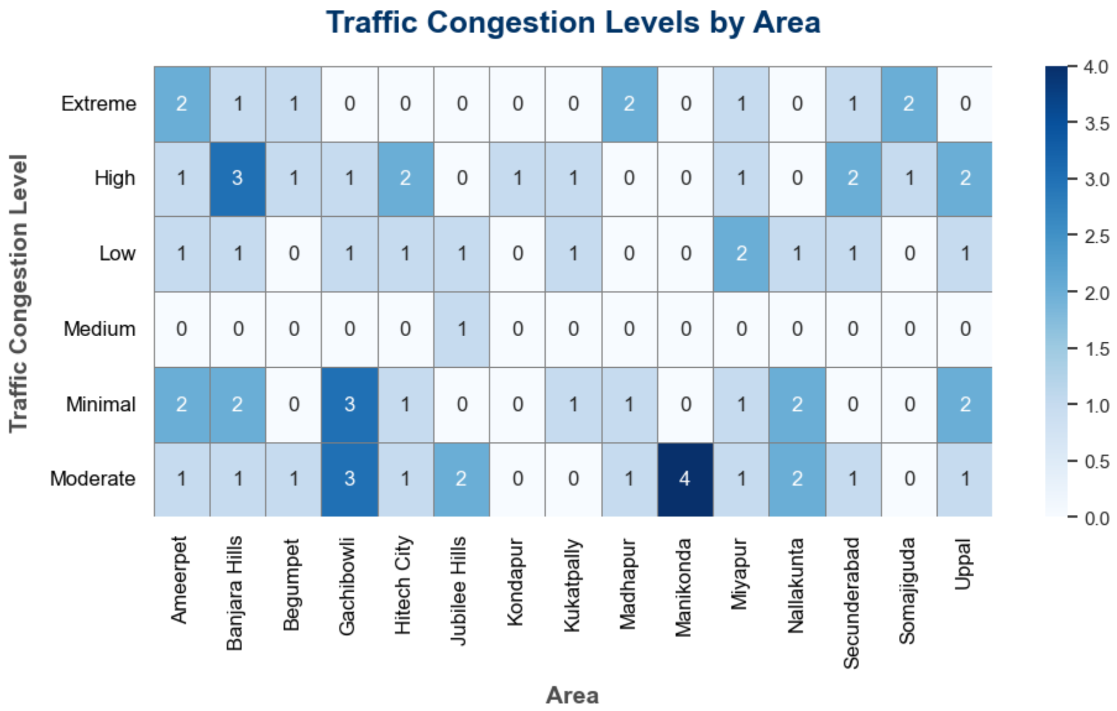
[312]:
```
import pandas as pd
from scipy.stats import chi2_contingency

contingency_table = pd.crosstab(df['Congestion Level'], df['Area'])

chi2_stat, p_value, dof, expected = chi2_contingency(contingency_table)

print("CHI-SQUARE TEST OF INDEPENDENCE")
print(f"Chi2 Statistic: {chi2_stat:.4f}")
print(f"P-value: {p_value:.4f}")
print(f"Degrees of Freedom: {dof}")

if p_value < 0.05:
    print("Conclusion: Chi-Square Test indicates a significant association␣
  ↪between traffic congestion levels and areas.")
else:
    print("Conclusion: Chi-Square Test indicates no significant association␣
  ↪between traffic congestion levels and areas.")
```

```
CHI-SQUARE TEST OF INDEPENDENCE
Chi2 Statistic: 69.2330
P-value: 0.5034
Degrees of Freedom: 70
Conclusion: Chi-Square Test indicates no significant association between traffic
congestion levels and areas.
```

- **Visualization**

[314]:

```
contingency_table = pd.crosstab(df['Congestion Level'], df['Area'])

plot_color = '#003366'
title_color = '#4B4B4B'

plt.figure(figsize=(10, 6))

sns.heatmap(contingency_table, annot=True, cmap='Blues', fmt='d', linewidths=0.
  ↪5, linecolor='gray')

plt.title('Traffic Congestion Levels by Area', fontsize=18, color=plot_color,␣
  ↪weight='bold', pad=20)
plt.xlabel('Area', fontsize=14, color=title_color, weight='bold', labelpad=10)
plt.ylabel('Traffic Congestion Level', fontsize=14, color=title_color,␣
  ↪weight='bold', labelpad=10)

plt.xticks(rotation=90, fontsize=12, color='black', weight='medium')
plt.yticks(rotation=0, fontsize=12, color='black', weight='medium')

plt.tight_layout()
plt.show()
```



Categorical Analysis of Speed Levels Across Areas

- **Statistical Analysis**

```
[317]: import pandas as pd
       from scipy.stats import chi2_contingency

       contingency_table = pd.crosstab(df['Speed Level'], df['Area'])

       chi2_stat, p_value, dof, expected = chi2_contingency(contingency_table)

       print("CHI-SQUARE TEST OF INDEPENDENCE")
       print(f"Chi2 Statistic: {chi2_stat:.4f}")
       print(f"P-value: {p_value:.4f}")
       print(f"Degrees of Freedom: {dof}")

       if p_value < 0.05:
           print("Conclusion: Chi-Square Test indicates a significant association␣
         ↪between traffic speed levels and areas.")
       else:
           print("Conclusion: Chi-Square Test indicates no significant association␣
         ↪between traffic speed levels and areas.")
```

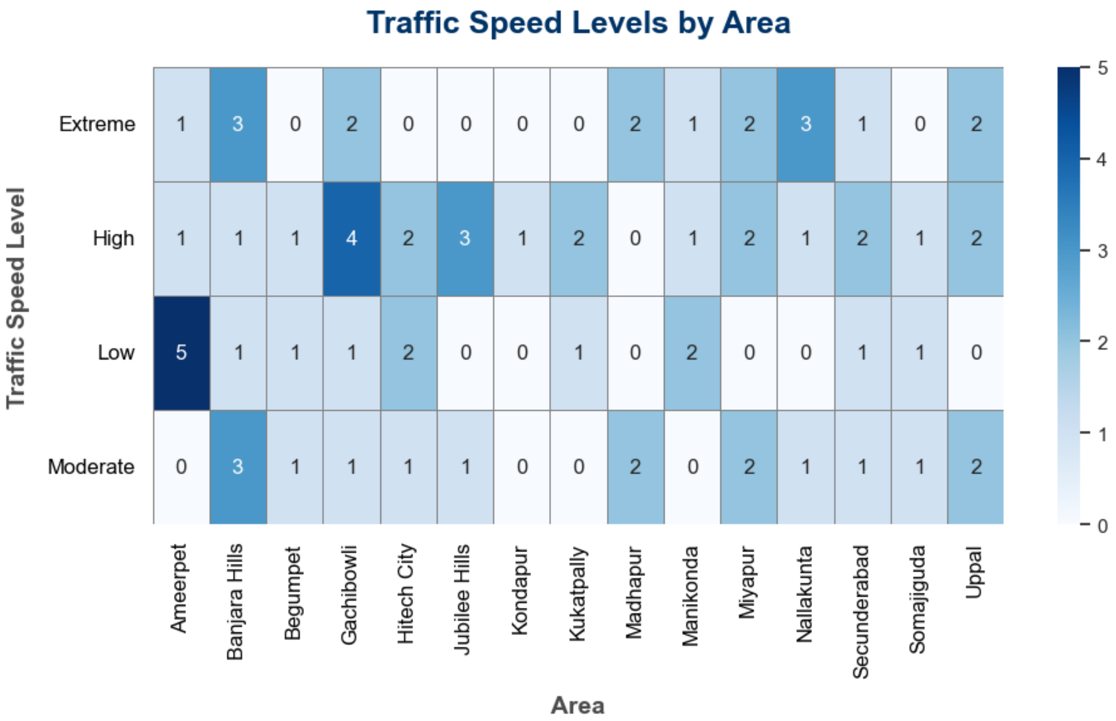
```
CHI-SQUARE TEST OF INDEPENDENCE
Chi2 Statistic: 43.2956
```

```
P-value: 0.4158
Degrees of Freedom: 42
Conclusion: Chi-Square Test indicates no significant association between traffic
speed levels and areas.
```

- **Visualization**

[319]:

```
contingency_table = pd.crosstab(df['Speed Level'], df['Area'])

plot_color = '#003366'
title_color = '#4B4B4B'

plt.figure(figsize=(10, 6))

sns.heatmap(contingency_table, annot=True, cmap='Blues', fmt='d', linewidths=0.
  ↪5, linecolor='gray')

plt.title('Traffic Speed Levels by Area', fontsize=18, color=plot_color,␣
  ↪weight='bold', pad=20)
plt.xlabel('Area', fontsize=14, color=title_color, weight='bold', labelpad=10)
plt.ylabel('Traffic Speed Level', fontsize=14, color=title_color,␣
  ↪weight='bold', labelpad=10)

plt.xticks(rotation=90, fontsize=12, color='black', weight='medium')
plt.yticks(rotation=0, fontsize=12, color='black', weight='medium')

plt.tight_layout()
plt.show()
```



Comparing Traffic Volume Between Peak and Non-Peak Hours

- Statistical Analysis

[322]:
```
from scipy.stats import ttest_ind, f_oneway, kruskal
import pandas as pd

# Define time periods and separate the data into groups
time_periods = df['Peak Hour?'].unique()
volume_groups = [df[df['Peak Hour?'] == period]['Vehicle Count'] for period in␣
 ↪time_periods]

# Statistical Analysis
if len(time_periods) == 2:
    # T-Test for comparing two time periods
    ttest_result = ttest_ind(volume_groups[0], volume_groups[1])
    print("T-TEST FOR TIME-BASED VARIATIONS IN TRAFFIC VOLUME")
    print(f"P-value: {ttest_result.pvalue:.4f}")
    if ttest_result.pvalue < 0.05:
        print("Conclusion: T-Test indicates significant variation in traffic␣
 ↪volume with respect to the peak hours.")
    else:
        print("Conclusion: T-Test indicates no significant variation in traffic␣
 ↪volume with respect to the peak hours.\n")
```

```
elif len(time_periods) > 2:
    # ANOVA for more than two time periods
    anova_result = f_oneway(*volume_groups)
    print("ANOVA FOR TIME-BASED VARIATIONS IN TRAFFIC VOLUME")
    print(f"P-value: {anova_result.pvalue:.4f}")
    if anova_result.pvalue < 0.05:
        print("Conclusion: ANOVA indicates significant variation in traffic␣
 ↪volume with respect to the peak hours. Let's perform the Kruskal-Wallis test.
 ↪")
        kruskal_result = kruskal(*volume_groups)
        print("\nKRUSKAL-WALLIS TEST")
        print(f"P-value: {kruskal_result.pvalue:.4f}")
        if kruskal_result.pvalue < 0.05:
            print("Conclusion: Kruskal-Wallis Test indicates significant␣
 ↪variation in traffic volume with respect to the peak hours (non-parametric).
 ↪")
        else:
            print("Conclusion: Kruskal-Wallis Test indicates no significant␣
 ↪variation in traffic volume with respect to the peak hours (non-parametric).
 ↪")
    else:
        print("Conclusion: ANOVA indicates no significant variation in traffic␣
 ↪volume with respect to the peak hours. Traffic volume is consistent␣
 ↪regardless.\n")
        print("KRUSKAL-WALLIS TEST")
        print("Conclusion: Not performed as ANOVA results are reliable and do␣
 ↪not show significant variation.")
else:
    print("Error: Not enough time periods for statistical analysis.")
```

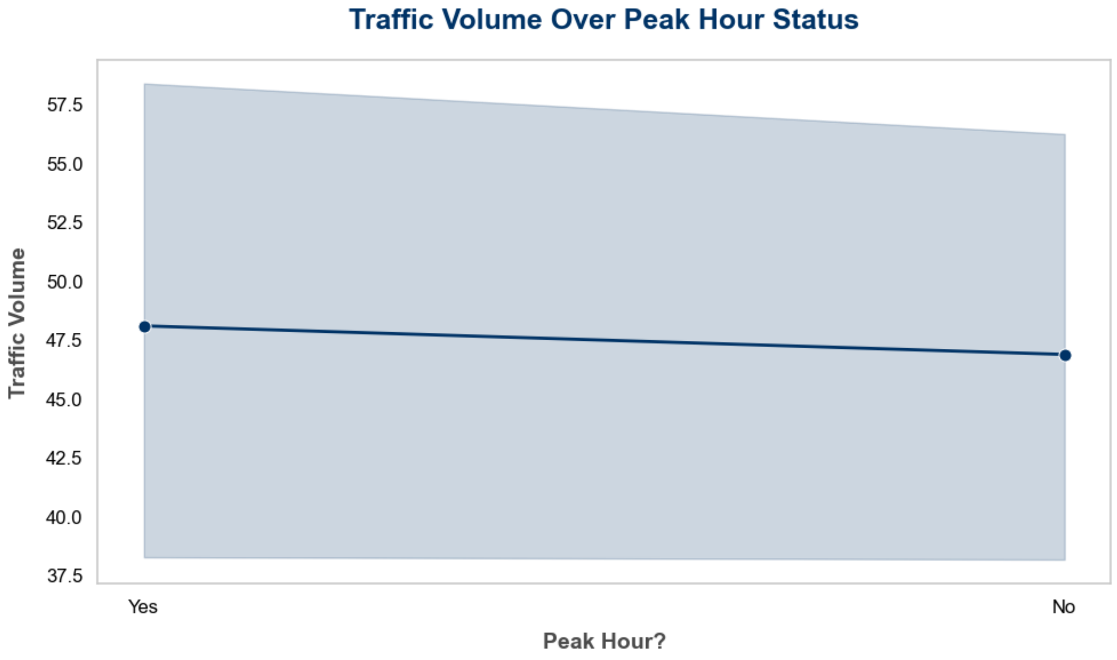
```
T-TEST FOR TIME-BASED VARIATIONS IN TRAFFIC VOLUME
P-value: 0.8666
Conclusion: T-Test indicates no significant variation in traffic volume with
respect to the peak hours.
```

- **Visualization**

[476]:

```
plot_color = '#003366'
line_color = '#003366'
title_color = '#4B4B4B'
marker_color = '#003366'

plt.figure(figsize=(10, 6))

sns.lineplot(x='Peak Hour?', y='Vehicle Count', data=df, color=line_color,␣
 ↪linewidth=2, marker='o', markersize=8, markerfacecolor=marker_color)
```

```
plt.title('Traffic Volume Over Peak Hour Status', fontsize=18,␣
  ↪color=plot_color, weight='bold', pad=20)
plt.xlabel('Peak Hour?', fontsize=14, color=title_color, weight='bold',␣
  ↪labelpad=10)
plt.ylabel('Traffic Volume', fontsize=14, color=title_color, weight='bold',␣
  ↪labelpad=10)

plt.xticks(rotation=0, fontsize=12, color='black', weight='medium')
plt.yticks(fontsize=12, color='black', weight='medium')

plt.grid(False)

plt.tight_layout()
plt.show()
```



Comparing Average Speed Between Peak and Non-Peak Hours

- **Statistical Analysis**

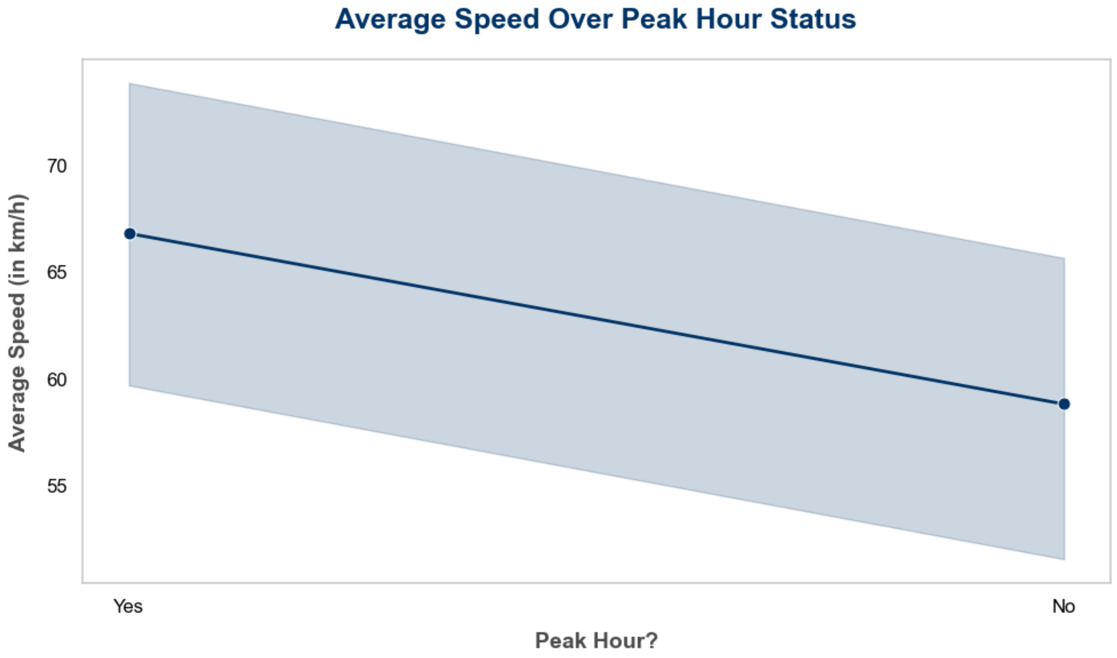
[327]:
```
from scipy.stats import ttest_ind, f_oneway, kruskal
import pandas as pd

time_periods = df['Peak Hour?'].unique()
volume_groups = [df[df['Peak Hour?'] == period]['Average Speed (in km/h)'] for␣
  ↪period in time_periods]
```

```
if len(time_periods) == 2:
    ttest_result = ttest_ind(volume_groups[0], volume_groups[1])
    print("T-TEST FOR TIME-BASED VARIATIONS IN AVERAGE SPEED")
    print(f"P-value: {ttest_result.pvalue:.4f}")
    if ttest_result.pvalue < 0.05:
        print("Conclusion: T-Test indicates significant variation in average␣
 ↪speed with respect to the peak hours.")
    else:
        print("Conclusion: T-Test indicates no significant variation in average␣
 ↪speed with respect to the peak hours.\n")

elif len(time_periods) > 2:
    anova_result = f_oneway(*volume_groups)
    print("ANOVA FOR TIME-BASED VARIATIONS IN AVERAGE SPEED")
    print(f"P-value: {anova_result.pvalue:.4f}")
    if anova_result.pvalue < 0.05:
        print("Conclusion: ANOVA indicates significant variation in average␣
 ↪speed with respect to the peak hours. Let's perform the Kruskal-Wallis test.
 ↪")
        kruskal_result = kruskal(*volume_groups)
        print("\nKRUSKAL-WALLIS TEST")
        print(f"P-value: {kruskal_result.pvalue:.4f}")
        if kruskal_result.pvalue < 0.05:
            print("Conclusion: Kruskal-Wallis Test indicates significant␣
 ↪variation in average speed with respect to the peak hours (non-parametric).")
        else:
            print("Conclusion: Kruskal-Wallis Test indicates no significant␣
 ↪variation in average speed with respect to the peak hours (non-parametric).")
    else:
        print("Conclusion: ANOVA indicates no significant variation in average␣
 ↪speed with respect to the peak hours. Traffic volume is consistent␣
 ↪regardless.\n")
        print("KRUSKAL-WALLIS TEST")
        print("Conclusion: Not performed as ANOVA results are reliable and do␣
 ↪not show significant variation.")
else:
    print("Error: Not enough time periods for statistical analysis.")
```

```
T-TEST FOR TIME-BASED VARIATIONS IN AVERAGE SPEED
P-value: 0.1345
Conclusion: T-Test indicates no significant variation in average speed with
respect to the peak hours.
```

- **Visualization**

```
[478]: plot_color = '#003366'  # Navy blue
       line_color = '#003366'  # Navy blue
       title_color = '#4B4B4B'  # Dark gray
       marker_color = '#003366'  # Navy blue

       plt.figure(figsize=(10, 6))

       sns.lineplot(x='Peak Hour?', y='Average Speed (in km/h)', data=df,␣
         ↪color=line_color, linewidth=2, marker='o', markersize=8,␣
         ↪markerfacecolor=marker_color)

       plt.title('Average Speed Over Peak Hour Status', fontsize=18, color=plot_color,␣
         ↪weight='bold', pad=20)
       plt.xlabel('Peak Hour?', fontsize=14, color=title_color, weight='bold',␣
         ↪labelpad=10)
       plt.ylabel('Average Speed (in km/h)', fontsize=14, color=title_color,␣
         ↪weight='bold', labelpad=10)

       plt.xticks(rotation=0, fontsize=12, color='black', weight='medium')
       plt.yticks(fontsize=12, color='black', weight='medium')

       plt.grid(False)

       plt.tight_layout()
       plt.show()
```



Categorical Analysis of the Association Between Weather Condition and Accident Levels

- **Statistical Analysis**

[459]:

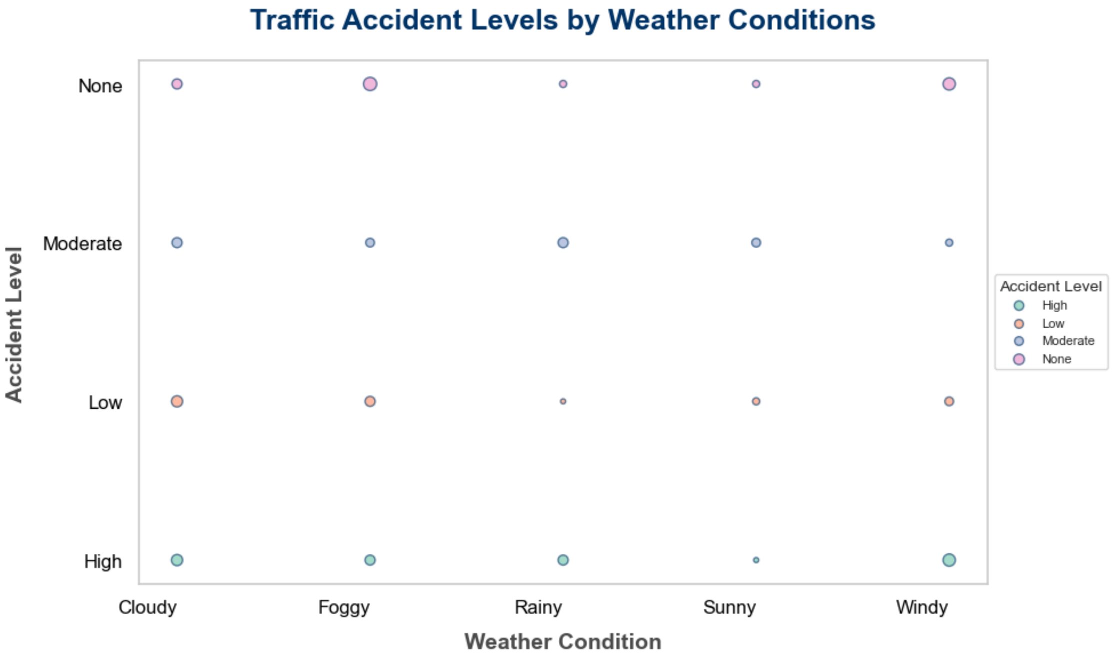
```
import pandas as pd
from scipy.stats import chi2_contingency

contingency_table = pd.crosstab(df['Weather Condition'], df['Accident Level'])

chi2_stat, p_value, dof, expected = chi2_contingency(contingency_table)
d
print("CHI-SQUARE TEST OF INDEPENDENCE")
print(f"Chi2 Statistic: {chi2_stat:.4f}")
print(f"P-value: {p_value:.4f}")
print(f"Degrees of Freedom: {dof}")

if p_value < 0.05:
    print("Conclusion: Chi-Square Test indicates a significant association␣
  ↪between weather conditions and accident levels.")
else:
    print("Conclusion: Chi-Square Test indicates no significant association␣
  ↪between weather conditions and accident levels.")
```

```
CHI-SQUARE TEST OF INDEPENDENCE
Chi2 Statistic: 7.3400
P-value: 0.8343
Degrees of Freedom: 12
Conclusion: Chi-Square Test indicates no significant association between weather
conditions and accident levels.
```

- **Visualization**

[480]:

```
contingency_table = pd.crosstab(df['Weather Condition'], df['Accident Level'])

plt.figure(figsize=(10, 6))

base_color = '#003366'
bubble_colors = plt.get_cmap('Set2').colors

for i, level in enumerate(contingency_table.columns):
    plt.scatter(
        x=contingency_table.index,
        y=[level] * len(contingency_table.index),
        s=contingency_table[level] * 10,  # Bubble size proportional to count
        alpha=0.6,
        color=bubble_colors[i],  # Assign color from the Set2 colormap
        edgecolor=base_color,
        label=level
    )

plt.title('Traffic Accident Levels by Weather Conditions', fontsize=18, color=base_color, weight='bold', pad=20)
plt.xlabel('Weather Condition', fontsize=14, color='#4B4B4B', weight='bold', labelpad=10)
plt.ylabel('Accident Level', fontsize=14, color='#4B4B4B', weight='bold', labelpad=10)

plt.xticks(rotation=0, ha='right', fontsize=12, color='black', weight='medium')
plt.yticks(fontsize=12, color='black', weight='medium')

plt.legend(title='Accident Level', title_fontsize='10', fontsize='8', loc='center left', bbox_to_anchor=(1, 0.5))

plt.grid(False)

plt.tight_layout()
plt.show()
```



Categorical Analysis of the Association Between Temperature Levels and Accident Levels

- Statistical Analysis

[332]:

```
import pandas as pd
from scipy.stats import chi2_contingency

contingency_table = pd.crosstab(df['Temperature Level'], df['Accident Level'])

chi2_stat, p_value, dof, expected = chi2_contingency(contingency_table)

print("CHI-SQUARE TEST OF INDEPENDENCE")
print(f"Chi2 Statistic: {chi2_stat:.4f}")
print(f"P-value: {p_value:.4f}")
print(f"Degrees of Freedom: {dof}")

if p_value < 0.05:
    print("Conclusion: Chi-Square Test indicates a significant association␣
 ↪between temperature levels and accident levels.")
else:
    print("Conclusion: Chi-Square Test indicates no significant association␣
 ↪between temperature levels and accident levels.")
```

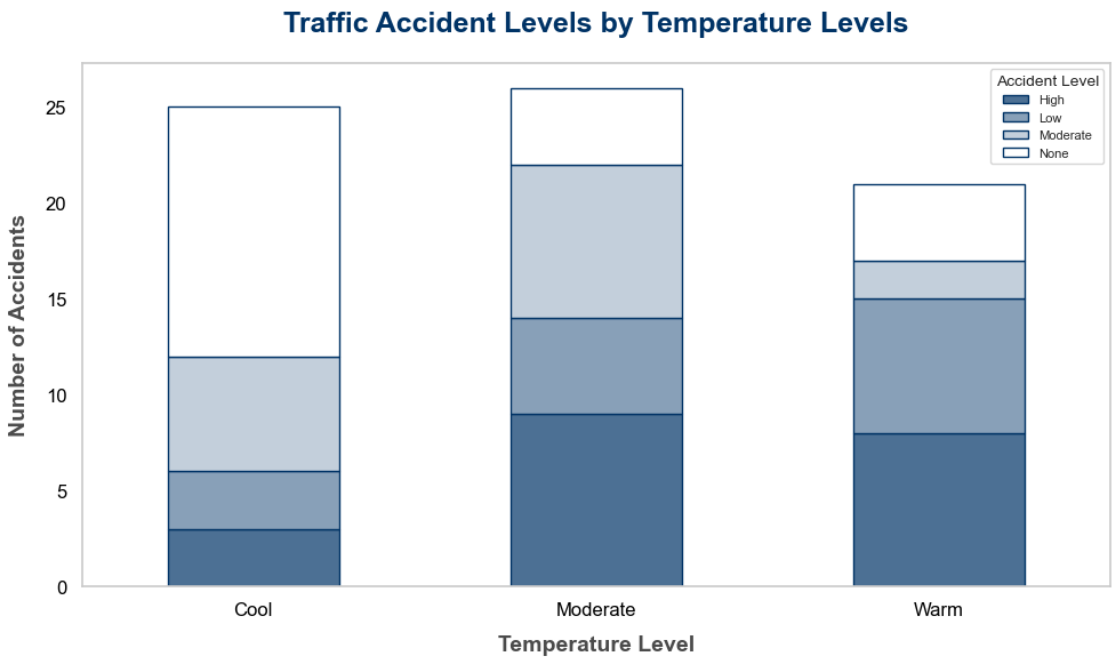
```
CHI-SQUARE TEST OF INDEPENDENCE
Chi2 Statistic: 15.3395
P-value: 0.0178
Degrees of Freedom: 6
Conclusion: Chi-Square Test indicates a significant association between
temperature levels and accident levels.
```

- **Visualization**

[334]:

```
base_color = '#003366'

def generate_shades(color, num_shades):
    base_rgb = mcolors.hex2color(color)
    shades = []
    for i in range(num_shades):
        # Calculate a lighter shade
        lightness = 0.3 + 0.7 * i / (num_shades - 1)  # Range from 0.3 to 1
        shade_rgb = [base_rgb[0] + (1 - base_rgb[0]) * lightness,
                     base_rgb[1] + (1 - base_rgb[1]) * lightness,
                     base_rgb[2] + (1 - base_rgb[2]) * lightness]
        shades.append(mcolors.to_hex(shade_rgb))
    return shades

num_accident_levels = len(df['Accident Level'].unique())
bar_colors = generate_shades(base_color, num_accident_levels)

contingency_table = pd.crosstab(df['Temperature Level'], df['Accident Level'])
```

```
ax = contingency_table.plot(kind='bar', stacked=True, figsize=(10, 6),␣
 ↪color=bar_colors, edgecolor='#003366')
plt.title('Traffic Accident Levels by Temperature Levels', fontsize=18,␣
 ↪color='#003366', weight='bold', pad=20)
plt.xlabel('Temperature Level', fontsize=14, color='#4B4B4B', weight='bold',␣
 ↪labelpad=10)
plt.ylabel('Number of Accidents', fontsize=14, color='#4B4B4B', weight='bold',␣
 ↪labelpad=10)
plt.xticks(rotation=0, fontsize=12, color='black', weight='medium')
plt.yticks(fontsize=12, color='black', weight='medium')

plt.legend(title='Accident Level', title_fontsize='10', fontsize='8',␣
 ↪loc='upper right')

plt.grid(False)

plt.tight_layout()
plt.show()
```



Categorical Analysis of the Association Between Visibility Levels and Accident Levels

- Statistical Analysis

[337]:
```
import pandas as pd
from scipy.stats import chi2_contingency
```

```
contingency_table = pd.crosstab(df['Visibility Level'], df['Accident Level'])

chi2_stat, p_value, dof, expected = chi2_contingency(contingency_table)

print("CHI-SQUARE TEST OF INDEPENDENCE")
print(f"Chi2 Statistic: {chi2_stat:.4f}")
print(f"P-value: {p_value:.4f}")
print(f"Degrees of Freedom: {dof}")

if p_value < 0.05:
    print("Conclusion: Chi-Square Test indicates a significant association␣
 ↪between visibility levels and accident levels.")
else:
    print("Conclusion: Chi-Square Test indicates no significant association␣
 ↪between visibility levels and accident levels.")
```

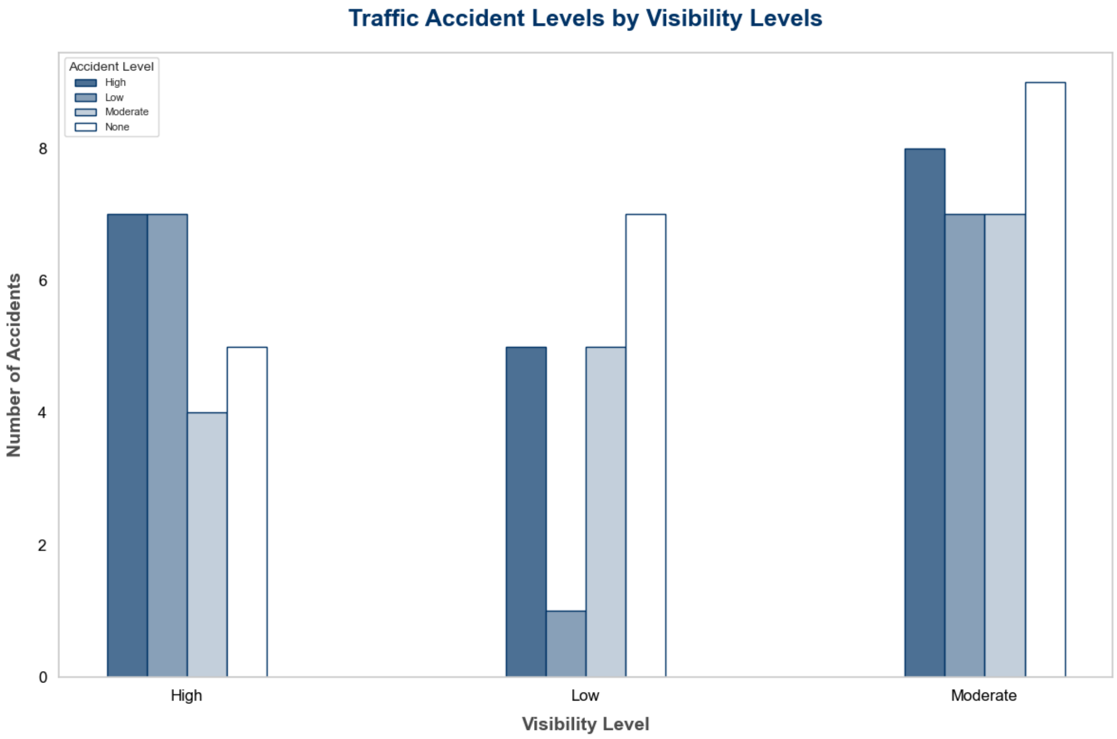
```
CHI-SQUARE TEST OF INDEPENDENCE
Chi2 Statistic: 4.6936
P-value: 0.5837
Degrees of Freedom: 6
Conclusion: Chi-Square Test indicates no significant association between
visibility levels and accident levels.
```

- **Visualization**

[435]:

```
base_color = '#003366'

def generate_shades(color, num_shades):
    base_rgb = mcolors.hex2color(color)
    shades = []
    for i in range(num_shades):
        lightness = 0.3 + 0.7 * i / (num_shades - 1)  # Range from 0.3 to 1
        shade_rgb = [base_rgb[0] + (1 - base_rgb[0]) * lightness,
                     base_rgb[1] + (1 - base_rgb[1]) * lightness,
                     base_rgb[2] + (1 - base_rgb[2]) * lightness]
        shades.append(mcolors.to_hex(shade_rgb))
    return shades

num_accident_levels = len(df['Accident Level'].unique())
bar_colors = generate_shades(base_color, num_accident_levels)

contingency_table = pd.crosstab(df['Visibility Level'], df['Accident Level'])

fig, ax = plt.subplots(figsize=(12, 8))

width = 0.1
x = range(len(contingency_table.index))
```

```
for i, (accident_level, color) in enumerate(zip(contingency_table.columns, bar_colors)):
    ax.bar([p + width * i for p in x], contingency_table[accident_level], width=width, label=accident_level, color=color, edgecolor='#003366')

ax.set_title('Traffic Accident Levels by Visibility Levels', fontsize=18, color='#003366', weight='bold', pad=20)
ax.set_xlabel('Visibility Level', fontsize=14, color='#4B4B4B', weight='bold', labelpad=10)
ax.set_ylabel('Number of Accidents', fontsize=14, color='#4B4B4B', weight='bold', labelpad=10)
ax.set_xticks([p + width * (num_accident_levels / 2 - 0.5) for p in x])
ax.set_xticklabels(contingency_table.index, fontsize=12, color='black', fontweight='medium')
ax.tick_params(axis='y', labelsize=12, colors='black', labelcolor='black')

ax.legend(title='Accident Level', title_fontsize='10', fontsize='8', loc='upper left')

ax.grid(False)

plt.tight_layout()
plt.show()
```



Categorical Analysis of the Association Between Humidity Levels and Accident Levels

- **Statistical Analysis**

[341]:

```
import pandas as pd
from scipy.stats import chi2_contingency

contingency_table = pd.crosstab(df['Humidity Level'], df['Accident Level'])

chi2_stat, p_value, dof, expected = chi2_contingency(contingency_table)

print("CHI-SQUARE TEST OF INDEPENDENCE")
print(f"Chi2 Statistic: {chi2_stat:.4f}")
print(f"P-value: {p_value:.4f}")
print(f"Degrees of Freedom: {dof}")

if p_value < 0.05:
    print("Conclusion: Chi-Square Test indicates a significant association␣
  ↪between humidity levels and accident levels.")
else:
    print("Conclusion: Chi-Square Test indicates no significant association␣
  ↪between humidity levels and accident levels.")
```

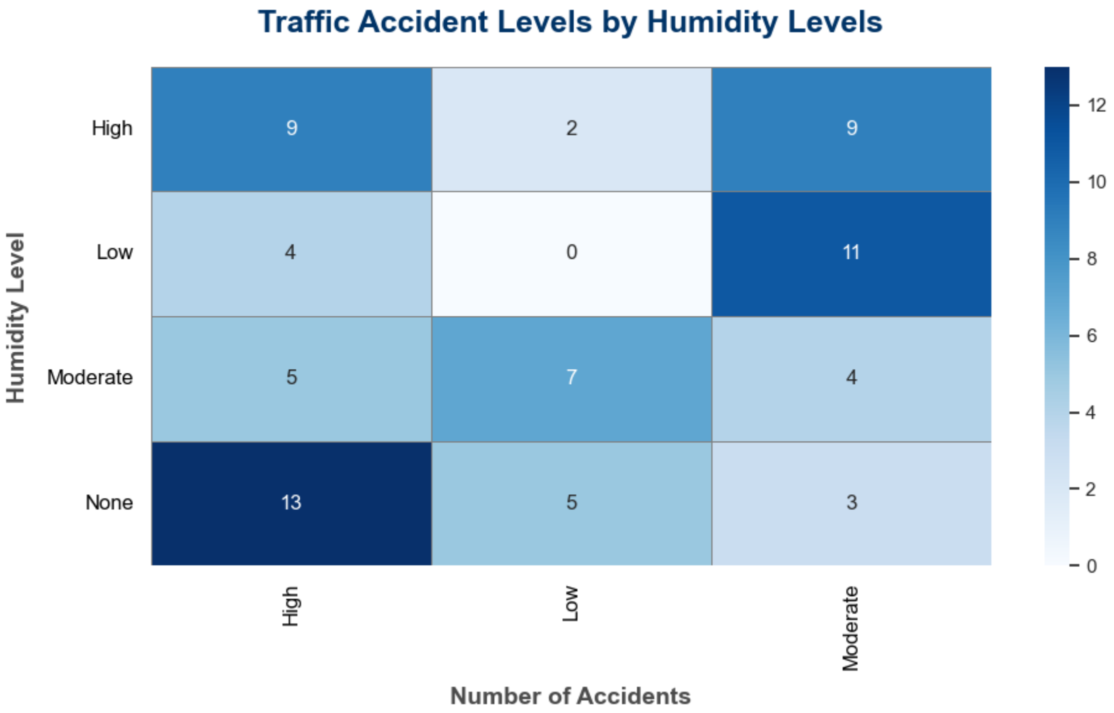
```
CHI-SQUARE TEST OF INDEPENDENCE
Chi2 Statistic: 21.2258
P-value: 0.0017
Degrees of Freedom: 6
Conclusion: Chi-Square Test indicates a significant association between humidity
levels and accident levels.
```

- **Visualization**

[465]:

```
contingency_table = pd.crosstab(df['Accident Level'], df['Humidity Level'])

plot_color = '#003366'
title_color = '#4B4B4B'

plt.figure(figsize=(10, 6))

sns.heatmap(contingency_table, annot=True, cmap='Blues', fmt='d', linewidths=0.
  ↪5, linecolor='gray')

plt.title('Traffic Accident Levels by Humidity Levels', fontsize=18,␣
  ↪color=plot_color, weight='bold', pad=20)
plt.xlabel('Number of Accidents', fontsize=14, color=title_color,␣
  ↪weight='bold', labelpad=10)
plt.ylabel('Humidity Level', fontsize=14, color=title_color, weight='bold',␣
  ↪labelpad=10)
```

```
plt.xticks(rotation=90, fontsize=12, color='black', weight='medium')
plt.yticks(rotation=0, fontsize=12, color='black', weight='medium')

plt.tight_layout()
plt.show()
```



Categorical Analysis of the Association Between Wind Speed Levels and Accident Levels

- Statistical Analysis

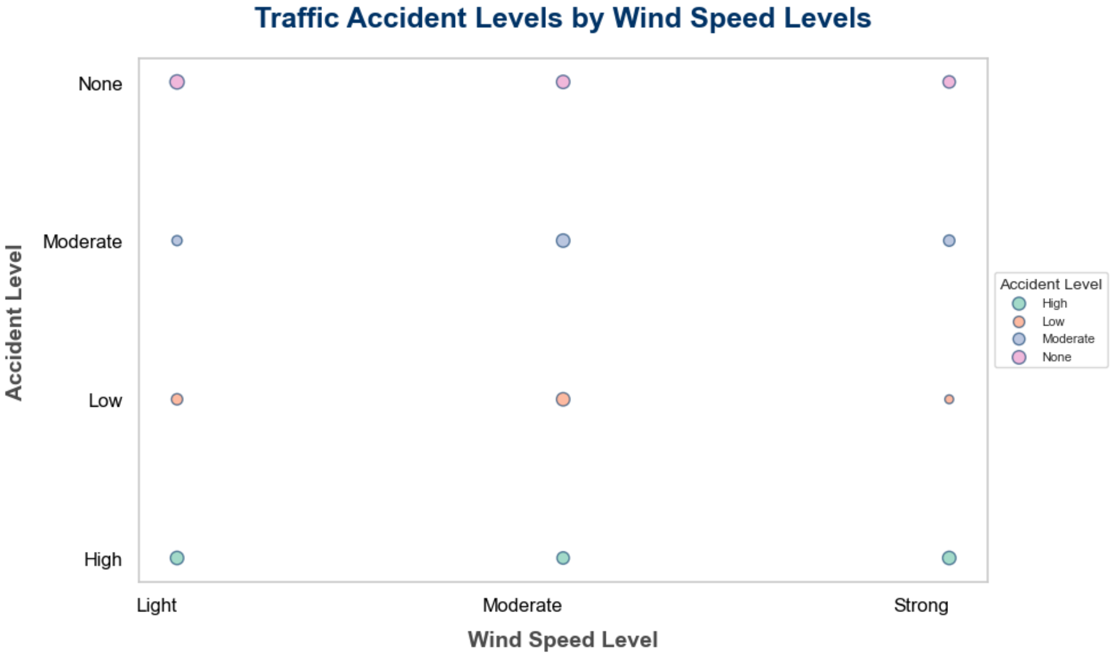
[345]:
```
import pandas as pd
from scipy.stats import chi2_contingency

contingency_table = pd.crosstab(df['Wind Speed Level'], df['Accident Level'])

chi2_stat, p_value, dof, expected = chi2_contingency(contingency_table)

print("CHI-SQUARE TEST OF INDEPENDENCE")
print(f"Chi2 Statistic: {chi2_stat:.4f}")
print(f"P-value: {p_value:.4f}")
print(f"Degrees of Freedom: {dof}")

if p_value < 0.05:
    print("Conclusion: Chi-Square Test indicates a significant association␣
 ↪between wind speed levels and accident levels.")
else:
    print("Conclusion: Chi-Square Test indicates no significant association␣
 ↪between wind speed levels and accident levels.")
```

```
CHI-SQUARE TEST OF INDEPENDENCE
Chi2 Statistic: 2.0847
P-value: 0.9117
Degrees of Freedom: 6
Conclusion: Chi-Square Test indicates no significant association between wind
speed levels and accident levels.
```

- **Visualization**

[482]:

```
contingency_table = pd.crosstab(df['Wind Speed Level'], df['Accident Level'])

plt.figure(figsize=(10, 6))

base_color = '#003366'
bubble_colors = plt.get_cmap('Set2').colors

for i, level in enumerate(contingency_table.columns):
    plt.scatter(
        x=contingency_table.index,
        y=[level] * len(contingency_table.index),
        s=contingency_table[level] * 10,  # Bubble size proportional to count
        alpha=0.6,
        color=bubble_colors[i],  # Assign color from the Set2 colormap
        edgecolor=base_color,
        label=level
    )

plt.title('Traffic Accident Levels by Wind Speed Levels', fontsize=18,␣
 ↪color=base_color, weight='bold', pad=20)
plt.xlabel('Wind Speed Level', fontsize=14, color='#4B4B4B', weight='bold',␣
 ↪labelpad=10)
plt.ylabel('Accident Level', fontsize=14, color='#4B4B4B', weight='bold',␣
 ↪labelpad=10)

plt.xticks(rotation=0, ha='right', fontsize=12, color='black', weight='medium')
plt.yticks(fontsize=12, color='black', weight='medium')

plt.legend(title='Accident Level', title_fontsize='10', fontsize='8',␣
 ↪loc='center left', bbox_to_anchor=(1, 0.5))

plt.grid(False)
plt.tight_layout()
plt.show()
```


Traffic Accident Levels by Wind Speed Levels
Accident Level
None
Moderate
Low
High
Light
Moderate
Strong
Wind Speed Level
Accident Level
High
Low
Moderate
None