Finding Heavy Traffic Indicators on I-94

Introduction

The purpose of this project is to analyze westbound traffic on the I-94 Interstate highway. The data used is provided by John Hogue and can be found on the UCI Machine Learning Repository

The dataset contains the hourly traffic volume for MN DoT ATR station 301 located between Minneapolis and St Paul, MN. The dataset contains features pertaining to holidays and weather in addition to the traffic volume.

Exploratory Data Analysis

In [1]:
```
# Import packages
import pandas as pd
import matplotlib.pyplot as plt
```

In [3]:
```
# Read in dataset
i_94_df = pd.read_csv("Metro_Interstate_Traffic_Volume.csv.gz")
display(i_94_df)
```

	holiday	temp	rain_1h	snow_1h	clouds_all	weather_main	weather_descripti
0	NaN	288.28	0.0	0.0	40	Clouds	scattered clou
1	NaN	289.36	0.0	0.0	75	Clouds	broken clou
2	NaN	289.58	0.0	0.0	90	Clouds	overcast clou
3	NaN	290.13	0.0	0.0	90	Clouds	overcast clou
4	NaN	291.14	0.0	0.0	75	Clouds	broken clou
...	...	...	...	...	...	...	
48199	NaN	283.45	0.0	0.0	75	Clouds	broken clou
48200	NaN	282.76	0.0	0.0	90	Clouds	overcast clou
48201	NaN	282.73	0.0	0.0	90	Thunderstorm	proximi thunderstor
48202	NaN	282.09	0.0	0.0	90	Clouds	overcast clou
48203	NaN	282.12	0.0	0.0	90	Clouds	overcast clou

48204 rows × 9 columns

In [10]:
```
def stats_overview(df, col):
    stat_table = df[[col]].describe()  # Doesn't include median
    median = df[[col]].median()
    stat_table.loc["median"] = median  # Add median to stat_table
    return stat_table


display(stats_overview(i_94_df, "traffic_volume"))
```

	traffic_volume
count	48204.000000
mean	3259.818355
std	1986.860670
min	0.000000
25%	1193.000000
50%	3380.000000
75%	4933.000000
max	7280.000000
median	3380.000000

Let's separate the dataset into night and day to see if there are any differences in the traffic volume statistics.

In [13]:
```
# Transform 'date_time to datetime'
i_94_df["date_time"] = pd.to_datetime(i_94_df["date_time"])

# Separate data into day and night
day_df = i_94_df.copy()[
    (i_94_df["date_time"].dt.hour >= 7) & (i_94_df["date_time"].dt.hour < 19
]

night_df = i_94_df.copy()[
    (i_94_df["date_time"].dt.hour >= 19) | (i_94_df["date_time"].dt.hour < 7
]

display("Daytime traffic volume", stats_overview(day_df, "traffic_volume"))
display("Nighttime traffic volume", stats_overview(night_df, "traffic_volume
```

'Daytime traffic volume'

	traffic_volume
count	23877.000000
mean	4762.047452
std	1174.546482
min	0.000000
25%	4252.000000
50%	4820.000000
75%	5559.000000
max	7280.000000
median	4820.000000

'Nighttime traffic volume'

	traffic_volume
count	24327.000000
mean	1785.377441
std	1441.951197
min	0.000000
25%	530.000000
50%	1287.000000
75%	2819.000000
max	6386.000000
median	1287.000000

Data Visualization

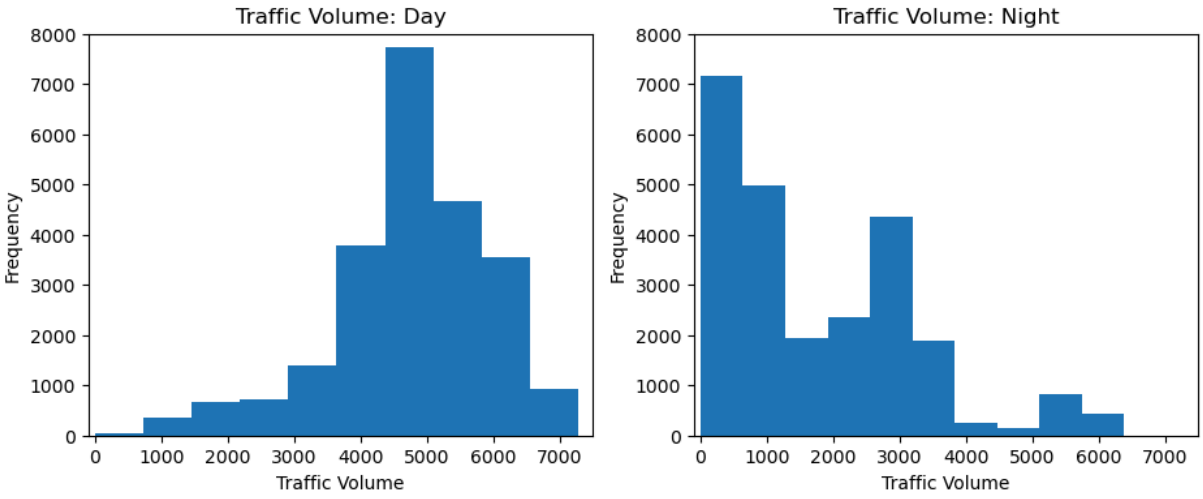
We will make a side-by-side histogram which will allow us to visualize the traffic volume in the day vs night.

In [14]:
```
plt.figure(figsize=(11, 4))

plt.subplot(1, 2, 1)
plt.hist(day_df["traffic_volume"])
plt.xlim(-100, 7500)
plt.ylim(0, 8000)
plt.title("Traffic Volume: Day")
plt.xlabel("Traffic Volume")
plt.ylabel("Frequency")

plt.subplot(1, 2, 2)
plt.hist(night_df["traffic_volume"])
plt.xlim(-100, 7500)
plt.ylim(0, 8000)
plt.title("Traffic Volume: Night")
plt.xlabel("Traffic Volume")
plt.ylabel("Frequency")

plt.show()
```



Conclusion

The distribution of the traffic volume during the day is left skewed; indicating that most of traffic volume values are high. The distribution of the nighttime data is right skewed; indicating mostly low traffic volume values.

Further analysis of this dataset could be performed to help find indicators of heavy traffic. Since traffic volumes are low in the evening, it would be reasonable to only analyze the daytime data.

Possible features to explore further are:

- Time indicators: Year, Month, Day, Hour
- Weather indicators