Municipal EV Charging Dashboard

Personal Practice Project | Data Analysis Portfolio

Overview

This project focuses on analyzing electric vehicle (EV) charging station data to generate clear, meaningful insights that help improve municipal planning and support the growing demand for sustainable transportation. The analysis covers the full lifecycle of EV charging activities from driver usage to energy trends allowing stakeholders to understand how charging stations are being used, when demand is highest, and which locations perform best.

By transforming raw charging session data into interactive dashboards, this project helps city officials, planners, and energy managers make informed decisions regarding infrastructure expansion, station maintenance, and resource allocation. It also provides a strong foundation for future forecasting and optimization as EV adoption continues to rise.

Data Source

The dataset used for this project was obtained from **Data.gov – U.S. Government Open Data**, a public platform that provides free access to high-quality, government-generated datasets. The raw EV charging records include details such as charging locations, energy consumption, charging duration, session timestamps, and session statuses.

This open-data source was selected because it offers:

- Reliable and publicly verified information
- Comprehensive session-level data suitable for detailed analysis
- Transparent documentation and metadata
- Accessibility for reproducibility and future enhancements

Using an official open government dataset ensures that the analysis is grounded in real-world EV

charging behavior and reflects actual infrastructure usage patterns. **Tools Used**

- **Python** — for data cleaning, preprocessing, and handling large datasets • **Excel / Power Query** — for data transformation and data visualization

Key Insights 1. Number of Total Drivers

I calculated the total unique drivers using the charging stations. This metric shows user adoption and helps identify demand for additional charging points.

2. Total Energy Provided

I calculated the total unique drivers using the charging stations. This metric shows user adoption and helps identify demand for additional charging points.

3. Peak Hours Identification

Using session timestamps, I identified the busiest hours of charging activity. This helps determine **when demand is highest**, supporting scheduling, maintenance planning, and potential station expansion.

4. Total Used Energy by Month

I aggregated monthly energy consumption to show usage trends over time. This allows stakeholders to easily see patterns, seasonality, growth in EV usage, and overall station performance.

5. Performance by Location (Duration & Energy)

I compared each charging location based on **average charging duration** and **total energy delivered**. This helps identify which stations are performing well, which ones are under-utilized, and where potential upgrades or maintenance may be needed.

5. Total Session Status

I analyzed the overall status of charging sessions (e.g., Disconnected, Paid, Roaming). This gives a quick view of station reliability, user behavior, and possible operational issues.

Insights Summary

The analysis showed steady EV charging activity, with clear peak hours and growing monthly energy usage. Some locations handled significantly higher demand, while session status data highlighted areas where station reliability could be improved.

Conclusion

This dashboard turns raw Data.gov data into simple, actionable insights. It helps cities understand usage patterns, identify high-demand locations, and make smarter decisions for improving and expanding EV charging infrastructure.

Python Code for Data Cleaning:

df = drop_duplicates()

- Remove duplicate rows to ensure each record is unique.

df[‘Date’] = pd.to_datatime(df[‘Date’])

- Convert the 'Date' column to datetime format for proper date-based analysis.

df[‘Location Name’] = df[‘Location Name’].replace(‘Old Name’ , ’New Name’)

- Standardize location names by replacing incorrect or outdated names.

df[‘Location Code’, ‘Location Name’] = df[‘Location Name’].str.split(‘-‘,n=1,expand = True)

- Split a combined column into separate 'Location Code' and 'Location Name' columns.

df[‘Column Name’] = df[‘Column Name’].map(‘{:.2f}’.format)

- Format numeric values to 2 decimal places for consistency.

df[‘Country’] = df[‘Country’].fillna(‘USA’)

- Fill missing country values with a default value ('USA') to avoid nulls.

Excel Functions & Methods Used:

=A1 + I2

- Combining Dates and Time

=HOUR(Q2)

- Extracts the hour from a time value in cell Q2.

=TEXT(Q2, "ddd")

- Converts a date into the day of the week (e.g., Mon, Tue, Wed).

Custom Dates

- Applies a custom date format to display dates in a specific style.