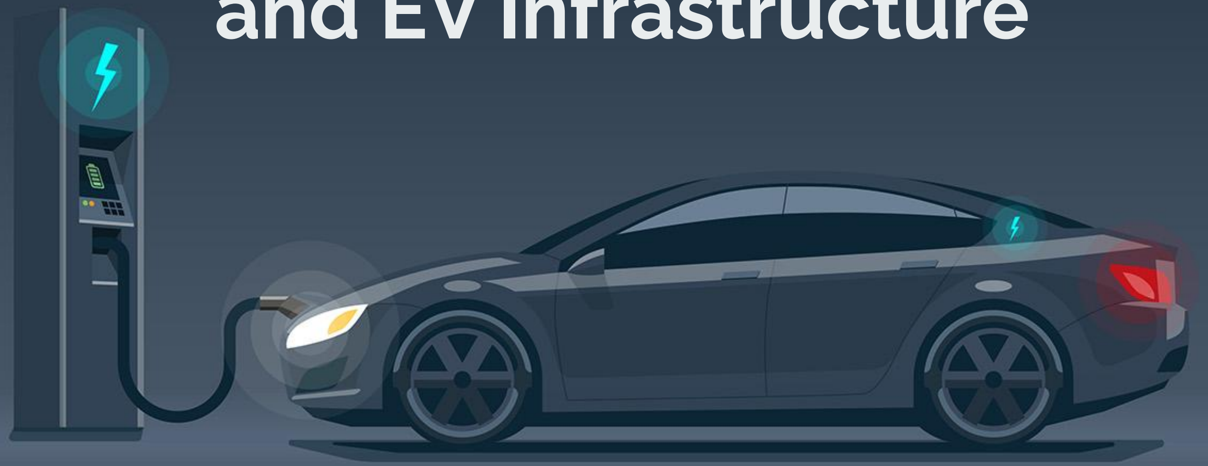
Modeling Electric Vehicle Registrations and EV Infrastructure



Duska, Emily, Lauren, Nuh

Table of Contents

Topic

We have chosen to evaluate the growth of electric vehicles nationwide and determine whether charging station infrastructure is on pace to meet demand.

Reason for Selection

1. About 27% of U.S. GHG emissions are attributed to transportation. This is the single biggest contributor.

2. Governments, both federal and state, are pushing policies to reduce the use of fossil-fuels.

3. Electric vehicle registrations have grown 320% nationwide over the last 6 years.

4. Electric vehicle infrastructure, such as charging stations, needs to be prioritized to meet this rising demand.

Project objective

Build machine learning models that can predict the number of electric vehicle registrations and charging stations by state.

Data Sources

- U.S. Department of Energy
  - Vehicle registration
  - Charging stations
- U.S. Census Bureau
  - Population
- Federal Reserve Bank of St. Louis
  - Median household income
  - Educational attainment

Vehicle Registrations

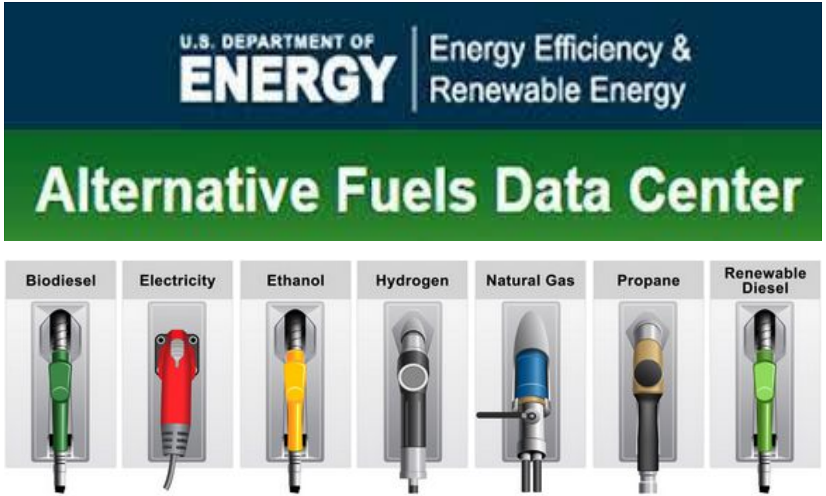
01

Source:

U.S. Department of Energy Alternative Fuels Data Center

Description:

Light-duty vehicle registration counts from 2016-2021 by state and fuel-type. Counts are rounded to the nearest 100 vehicles.



Alternative Fuels Data Center. (2021). *Vehicle Registration Counts by State* [data set]. https://afdc.energy.gov/vehicle-registration.

Charging Stations

02

Source:
U.S. Department of Energy Alternative Fuels Data Center

Description:
Alternative fuel stations, including electric, nationwide as of July 19, 2021. Information includes address, access, open date, etc.

Alternative Fuels Data Center. (2021). *Alternative Fuel Stations 2021* [data set]. https://catalog.data.gov/dataset/alternative-fueling-station-locations-422f2.

Population

03

Source:
U.S. Census Bureau, Population Division

Description:
Population estimates by state and region from 2010 to 2022. Estimates are based on the 2010 Census.

U.S. Census Bureau. Population Division. (2019). Annual Estimates of the Resident Population for the United States, Regions, States, and Puerto Rico [data set]. https://www.census.gov/data/datasets.html.

Median Household Income

04

Source:
Federal Reserve Bank of St. Louis

Description:
Annual median household income (dollars) by state and nationwide from 2016-2021.

Federal Reserve Bank of St. Louis (2021). *Median Household Income by State, Annual* [data set]. https://fred.stlouisfed.org/release/tables?rid=249&eid=259462.

Educational Attainment

05

Source:
Federal Reserve Bank of St. Louis

Description:
Percent of population 18 years old and over with a bachelor's degree or higher by state. Estimates are based on the American Community Survey.

Federal Reserve Bank of St. Louis. (2021). *Bachelor's Degree or Higher by State* [data set]. https://fred.stlouisfed.org/release/tables?rid=330&eid=391444#snid=391450.

Final Dataset

- US states and District of Columbia
- 2016 - 2021
- Non_ev_total includes:
    - Fossil-fuel vehicles
    - Other alternative fuel vehicles

ev_registration_gigas	
state	varchar
year	integer
electric_ev	integer
plug_in_hybrid_electric_phev	integer
charging_station_count	integer
income	float
education	float
population	float
non_ev_total	integer

Data Exploration

- Initial Dataset Inspection
- Correlation Analysis
- Feature Engineering

Initial Data Inspection

- Scrutinized structure & variables
- Evaluated data types, value ranges, missing/irregular data
- Preprocessed datasets for import
  - Renamed/removed columns
  - Altered data types
  - Removed commas in population, income, and education datasets

```python
# Define the table schemas and create the tables
Population_Columns = {
    'state': String,
    '2016': Integer,
    '2017': Integer,
    '2018': Integer,
    '2019': Integer,
    '2020': Integer,
    '2021': Integer,
    '2022': Integer,
}
create_table(engine, 'population', Population_Columns)

#Import and insert charging station data
df_population = pd.read_csv("population_estimates_2016-2022_fix.csv")
# Define the numeric columns that need commas removed
numeric_columns = [
    '2016',
    '2017',
    '2018',
    '2019',
    '2020',
    '2021',
    '2022'


]

# Remove commas from the numeric columns and convert them to integers
for col in numeric_columns:
    if df_population[col].dtype == object:
        df_population[col] = df_population[col].str.replace(',', '').astype(int)



insert_data(df_population, 'population', engine)
```

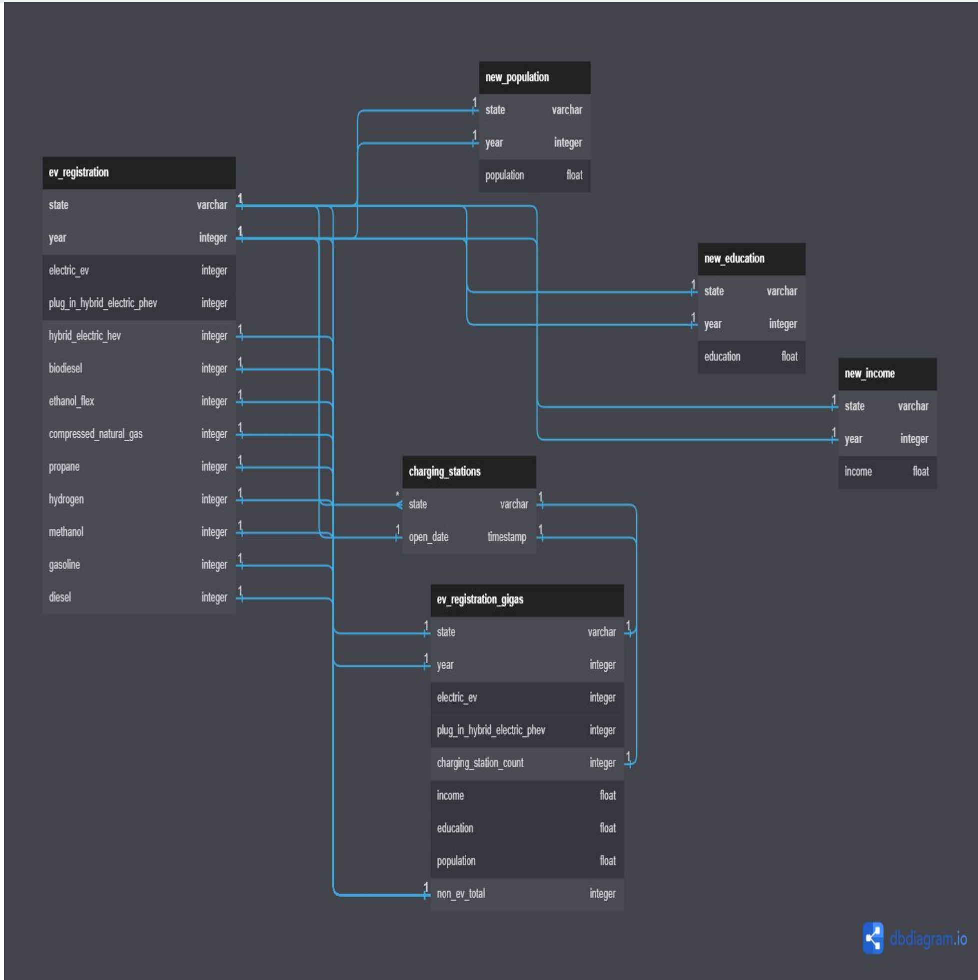
Correlation Analysis

- Conducted correlation analysis to assess linear relationships between continuous variables
- Focused on year and state as the continuous variables within our datasets.
- Restructured our Population, Income, and Education datasets for easier joining with EV registration data.
- Utilized the open_date column in our charging station dataset to create a count by year column.
- Performed four left joins to create a comprehensive table with all the desired data.
- Used Year and State as primary keys for joining the tables in pgAdmin.

```sql
CREATE TABLE ev_registration_gigas AS
SELECT
    ev.state,
    ev.year,
    ev.electric_ev,
        ev.plug_in_hybrid_electric_phev,
    cs.charging_station_count,
    ni.income,
    ne.education,
    np.population
FROM ev_registration ev
LEFT JOIN (
    SELECT state, year, SUM(count) OVER (PARTITION BY state ORDER BY year) AS charging_station_count
    FROM (
        SELECT state, EXTRACT(year FROM open_date)::integer AS year, COUNT(*) AS count
        FROM charging_stations
        GROUP BY state, EXTRACT(year FROM open_date)
    ) sub_cs
) cs ON ev.state = cs.state AND ev.year = cs.year
LEFT JOIN new_income ni ON ev.state = ni.state AND ev.year = ni.year::integer
LEFT JOIN new_education ne ON ev.state = ne.state AND ev.year = ne.year::integer
LEFT JOIN new_population np ON ev.state = np.state AND ev.year = np.year::integer
ORDER BY ev.state, ev.year;



ALTER TABLE ev_registration_gigas
ADD COLUMN non_ev_total INTEGER;


UPDATE ev_registration_gigas
SET non_ev_total = er.hybrid_electric_hev + er.biodiesel + er.ethanol_flex + er.compressed_natural_gas + er.propane + er.hydrogen + er.methanol + er.gasoline + er.diesel
FROM ev_registration er
WHERE ev_registration_gigas.state = er.state AND ev_registration_gigas.year = er.year;
```

Feature Engineering

- Final table used for ML models & visualization dashboard.
- Dashboard: US map (50 states), EV & charging station counts per year.
- ML models: Population, income, education, EV total & charging station count used as variables.
    - Goal: Predict EV registration growth & charging station expansion.



Questions to Answer

What Can the Data Tell Us?

By exploring, modeling, and visualizing the data previously described, we hope to answer the following questions:

01 | What are the adoption rates of EVs by state and nationwide?

02 | What is the ratio of charging stations to EVs by state and nationwide?

03 | Does population, income, and education accurately predict EV adoption?

04 | What does our model predict for 2021 EV adoption and charging stations?

05 | Which states have the appropriate charging station infrastructure?

Analysis

- Variable selection
- Model selection
- Model training
- Model Evaluation

Variable & Model Selection

Variables Analyzed: Year, income, education, population, non EV total (independent variables)

EV total, Charging station count (dependent variables)

Model Selection: We chose Linear Regression after evaluating various machine learning models, due to its optimal performance and interpretability for our dataset.

```
df = df1.drop(['state', 'electric_ev', 'plug_in_hybrid_electric_phev', 'charging_station_count'], axis= 1)
print(df)
```

```
     year  income  education  population  non_ev_total  ev_total
0    2021   76918      0.307     1904314       1875700      6000
1    2021   79253      0.371    12686469       9913300     54800
2    2021   70190      0.289     6813532       6018200     17900
3    2021   72429      0.305     3197689       3043800      7300
4    2021   75979      0.354     2937922       2565100      7800
..    ...     ...        ...         ...           ...       ...
301  2016   80268      0.381     8410106       7331000      7300
302  2016   87648      0.351     7294771       6073200     21300
303  2016   46836      0.208     1831023       1502400       400
304  2016   69943      0.295     5772628       5310200      5200
305  2016   71052      0.271      584215        591200       200

[306 rows x 6 columns]
```

```
# creating train and test sets
#X_train, X_test, y_train, y_test = train_test_split(
#    X, y, shuffle = True)

train_index = X.year==2021
X_train = X.loc[~train_index]
X_test = X.loc[train_index]
y_train = y.loc[~train_index]
y_test = y.loc[train_index]
```

```
# creating a regression model
model = LinearRegression()
```

Model Training

- Independent variables: year, income, education, population, non-EV total
- Targets: EV total and Charging station count
- Models Linear Regression
- Outcome: Generated predictions

```
X = df.drop('ev_total',axis= 1)
y = df['ev_total']
print(X)
print(y)
```

```
     year  income  education  population  non_ev_total
0    2021   76918      0.307     1904314       1875700
1    2021   79253      0.371    12686469       9913300
2    2021   70190      0.289     6813532       6018200
3    2021   72429      0.305     3197689       3043800
4    2021   75979      0.354     2937922       2565100
..    ...     ...        ...         ...           ...
301  2016   80268      0.381     8410106       7331000
302  2016   87648      0.351     7294771       6073200
303  2016   46836      0.208     1831023       1502400
304  2016   69943      0.295     5772628       5310200
305  2016   71052      0.271      584215        591200

[306 rows x 5 columns]
0       6000
1      54800
2      17900
3       7300
4       7800
       ...
301     7300
302    21300
303      400
304     5200
305      200
Name: ev_total, Length: 306, dtype: int64
```

Model Evaluation

We assessed the performance of our models using appropriate evaluation metrics, such as mean squared error and mean absolute error.

Charging Stations

```
# model evaluation
print(
    'mean_squared_error : ', mean_squared_error(y_test, predictions))
print(
    'mean_absolute_error : ', mean_absolute_error(y_test, predictions))
```

```
mean_squared_error :  4709182.604144309
mean_absolute_error :  821.7638589782542
```

EV's

```
# model evaluation
print(
    'mean_squared_error : ', mean_squared_error(y_test, predictions))
print(
    'mean_absolute_error : ', mean_absolute_error(y_test, predictions))
```

```
mean_squared_error :  6252853592.8166275
mean_absolute_error :  27925.482282963163
```

Technologies & Tools

Technologies & Tools

Database

Modeling

Visualization

Results of Our Analysis

Charging Stations

```
# fitting the model
model.fit(X_train,y_train)
training_score = model.score(X_train, y_train)
testing_score = model.score(X_test, y_test)

print(f"Training Score: {training_score}")
print(f"Testing Score: {testing_score}")
```

```
Training Score: 0.8749101272942907
Testing Score: 0.6260192606608335
```

EV's

```
# fitting the model
model.fit(X_train,y_train)
training_score = model.score(X_train, y_train)
testing_score = model.score(X_test, y_test)

print(f"Training Score: {training_score}")
print(f"Testing Score: {testing_score}")
```

```
Training Score: 0.5894207936808619
Testing Score: 0.5775005368657995
```

Linear Regression Equations

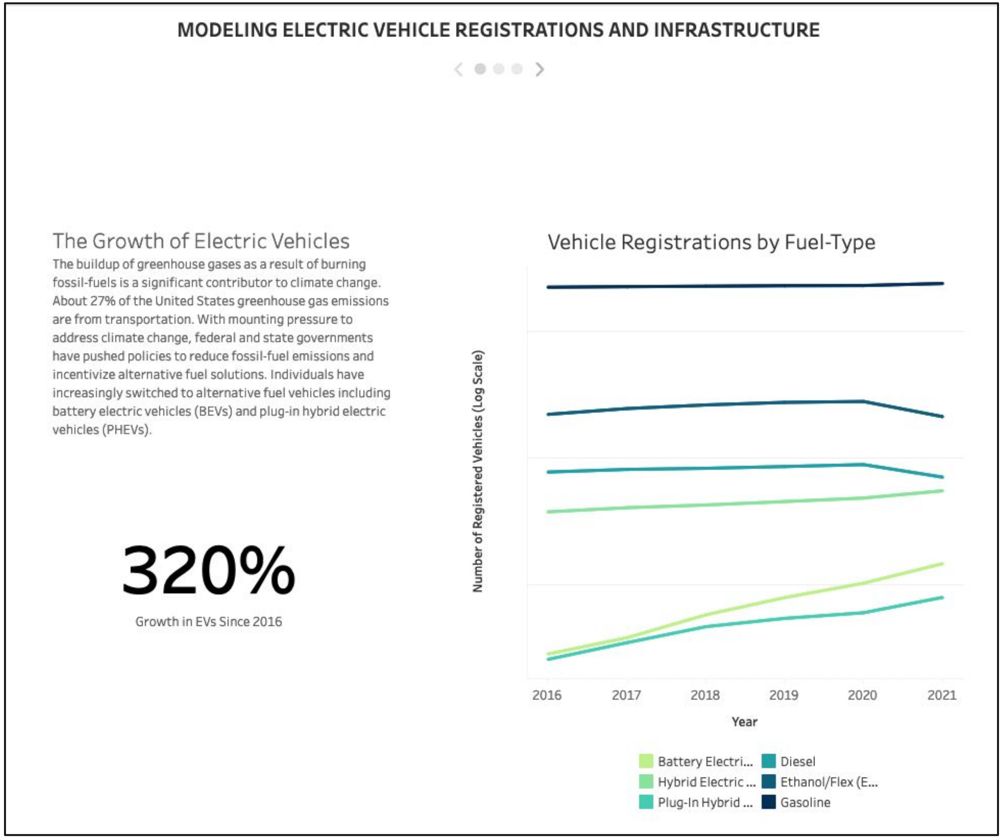
Electric Vehicle Model

```
Intercept:  -7846442.181098354
Coefficients:
 [('year', 3844.559881461876),
  ('income', 1.0594712361829406),
  ('education', -50107.46557777868),
  ('population', -0.011333359628165666),
  ('non_ev_total', 0.023256862853493293)]
```

Charging Station Model

```
Intercept:  -193235.480352949
Coefficients:
 [('year', 95.69215042620789),
  ('income', -0.0008107494534003849),
  ('education', 1120.800782229163),
  ('population', 7.335480147190854e-05),
  ('non_ev_total', -4.97690388281733e-05),
  ('ev_total', 0.010543792141677756)]
```

[Tableau Dashboard]



Recommendations

To gain deeper insights into EV adoption and charging station growth in the US, we suggest further analysis.

- Study impact of local/state policies & incentives on EV adoption & infrastructure.
- Compare urban vs. rural areas in EV adoption & infrastructure challenges.
- Evaluate environmental & economic benefits of EV adoption.

Things to do With More Time

Given additional time, we could broaden our analysis and enhance the precision of our forecasts

1. Incorporate extra data, including vehicle types and regional elements, which may affect EV adoption and charging station needs.
2. Refine ML model with more algorithms for 2030 predictions, considering tech trends, policies, and consumer preferences.
3. Analyze EV infrastructure costs vs. economic & environmental benefits.
4. Assess EV adoption's impact on electrical grid & potential upgrades.

Thank you.
Questions?